Oracle® Database
データベース・アップグレード・ガイド
19c

F16191-19(原本部品番号:E96252-57)

2023年9月

タイトルおよび著作権情報

Oracle Databaseデータベース・アップグレード・ガイド, 19c

F16191-19

Copyright © 1996, 2023, Oracle and/or its affiliates.

原本著者: Douglas Williams

原本協力者: Drew Adams、Pablo Sainz Albanez、Frederick Alvarez、Yasin Baskan、Subhransu Basu、Rae Burns、Rhonda Day、Mike Dietrich、Joseph Errede、Daniel Overby Hansen、Yuan Hao、Subrahmanyam Kodavaluru、Jai Krisnani、Cindy Lim、Valarie Moore、Byron Motta、Satish Panchumarthy、Geetha Ravi、Carol Tagliaferri、Hector Vieyra

目次

- 例の一覧
- 図一覧
- 表一覧
- タイトルおよび著作権情報
- はじめに
  - 対象読者
  - ドキュメントのアクセシビリティ
  - ダイバーシティ&インクルージョン
  - Java Accessibilityを実装するためのJava Access Bridgeのセットアップ
  - 関連ドキュメント
  - 表記規則
- 1 Oracle Databaseのアップグレードの概要
  - Oracle Databaseのアップグレード・ツールおよびアップグレード処理の概要
    - アップグレードおよび移行という用語の定義
    - アップグレードおよびデータ移行方法と処理
    - 直接のアップグレードをサポートするOracle Databaseのリリース
    - Oracle Databaseのアップグレードに関する最新情報の入手先
  - Oracle Databaseのアップグレード処理の主なステップ
  - Oracle Databaseリリース間の互換性および相互運用性
    - Oracle Databaseのリリース番号について
    - 『Oracle Databaseアップグレード・ガイド』のリリース番号を参照する際の規則に関する説明
    - Oracle Databaseの互換性の概要
      - Oracle Databaseの互換性の理解
      - Oracle DatabaseでのCOMPATIBLE初期化パラメータ設定のタイミング
      - Oracle DatabaseのCOMPATIBLE初期化パラメータ
      - Oracle DatabaseのCOMPATIBLE初期化パラメータの値
      - Oracle Databaseのアップグレードでのダウングレードおよび互換性について
      - Oracle DatabaseでのCOMPATIBLE初期化パラメータの動作
      - Oracle Databaseの互換性レベルの確認
    - Oracle Databaseのアップグレードの相互運用性の概要
    - 無効なスキーマ・オブジェクトおよびデータベース・アップグレードについて
    - Oracle OLAPデータ・セキュリティ・ポリシーのアップグレードについて
  - 複数のOracleリリースの実行について
    - 異なるコンピュータ上の複数のOracleホームに存在するデータベース
    - 同じコンピュータ上の複数のOracleホームに存在するデータベース
    - Optimal Flexible Architecture標準について
    - 複数のOracleホームのサポートについて
  - アップグレード中のデータベース変換について
    - アップグレード中のデータベース変換の概要
    - スタンバイ・データベースを使用したアップグレードについて
    - Oracle GoldenGateを使用したOracle Databaseのアップグレード・ステップの概要

- Oracle DatabaseのStandard EditionからEnterprise Editionへの移行
- Oracle DatabaseのEnterprise EditionからStandard Editionへの移行
- Oracle Database Express Edition (Oracle Database XE)からOracle Databaseへの移行
- 新しいOracle Databaseリリース用のプラットフォームのアップグレードについて
  - オペレーティング・システムのアップグレードについて
  - 異なるオペレーティング・システムにデータを転送するオプション
- イメージ・ベースのOracle Databaseのインストールについて

- 2 Oracle Databaseのアップグレードの準備
  - Oracle Databaseのアップグレードを準備する作業
    - Oracle Databaseの新機能の理解
    - AutoUpgradeでのアップグレード前情報チェック
    - 非推奨となったかサポートが終了した機能の確認
    - Oracle Databaseのアップグレード方法の選択
      - AutoUpgradeユーティリティでOracle Databaseをアップグレードする方法
      - 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のアップグレード時のデータの移行でエクスポート/インポートする方法
        - アップグレードしたOracle Databaseでのエクスポート/インポートの影響
        - Oracle Databaseのデータ移行でのエクスポート/インポートの利点
        - エクスポート/インポートを使用したデータ移行の時間要件
      - 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
    - アップグレード時のOracleホームの新しい場所の選択
    - Oracle Databaseのアップグレードでのテスト計画の開発
      - アップグレード・テスト
      - 最小テスト
      - アップグレード後の機能テスト
      - 高可用性テスト
      - アプリケーションの互換性を確保する統合テスト
      - アップグレードしたOracle Databaseのパフォーマンス・テスト
        - データベース・リプレイおよびパフォーマンス・テスト
        - SQLパフォーマンス・アナライザ
        - SQL計画管理
          - SQL計画管理を実行する理由
          - カーソル・キャッシュからのSQL管理ベースのバルク・ロード
          - SQLチューニング・セット(STS)を使用したSQL管理ベースのバルク・ロード
          - ステージング表からの既存のSQL計画ベースラインの展開
      - Oracle Databaseのアップグレードでのボリューム/ロード・ストレス・テスト
      - Oracle Databaseアップグレード計画のテスト計画のガイドライン
    - スキーマ専用アカウントおよびEXPIREDパスワード・アカウントのアップグレード
    - ダウングレードおよびリカバリ・オプションを保持するためのファイルのバックアップ
      - 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 Data Guard Brokerの構成ファイルおよびダウングレード
        - ブローカ構成のエクスポート
    - AutoUpgradeでのアップグレード前情報チェック
    - 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 RAC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リリースの更新および要件
        - アップグレードおよび透過的データ暗号化
        - Oracle Databaseのアップグレード時のOracle Net Servicesに関する推奨事項
        - ORAPWDを使用したパスワード・ファイルの作成または移行
        - 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
        - STIGおよびCISプロファイルのリソースおよびパスワード・パラメータの更新
        - プロファイル・スクリプト(glogin.sqlおよびlogin.sql)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の高可用性オプション
            - 監査表の事前アップグレードおよびアーカイブの要件
        - Oracle Database Standard Editionでの高可用性のオプション
            - Standard Edition Oracle RACまたはOracle RAC One Nodeのアップグレードの準備
            - Oracle DatabaseでStandard Edition High Availabilityを使用するための要件
        - 統合監査証跡へのオペレーティング・システムの監査レコードの移動
        - 非CDBのアップグレードとOracle GoldenGate
        - AutoUpgradeを使用する前の大規模データベースのバックアップ
    - Oracle Data Guardを使用したデータベースのローリング・アップグレードの準備
    - アップグレードのための新しいOracleホームの準備
    - Windows上のOracleホームを準備するための前提条件
    - AutoUpgradeを使用したアップグレード前チェックの実行
        - AutoUpgradeユーティリティのシステム・チェックについて
        - 分析モードを使用したAutoUpgradeの事前チェックの実行の例
        - アップグレード・チェックの概要ファイルの確認
        - CDBでAutoUpgradeの事前チェックを実行するための構成ファイルの作成
        - 以前のリリースのOracleデータベースでのAutoUpgrade修正の実行
    - Oracle Databaseのアップグレード処理のテスト
        - 優先度リスト・エミュレーションを使用したアップグレードのテストの例
        - Oracle Call Interface (OCI)およびプリコンパイラ・アプリケーションのアップグレード
    - Oracle Label SecurityおよびOracle Database Vaultを使用しているデータベースのアップグレードの要件
        - 監査表の事前アップグレードおよびアーカイブの要件
        - Oracle Database Vaultを使用したデータベースのアップグレードの準備
        - Oracle Database VaultおよびOracle Databaseリリース11.2のアップグレード
    - アップグレードのためのOracle Databaseのバックアップ
- 3 Oracle DatabaseアップグレードでのAutoUpgradeの使用

- Oracle Database AutoUpgradeについて
- AutoUpgradeの使用方法の例
  - 同じサーバー上にソースおよびターゲットのデータベース・ホームがあるAutoUpgrade(通常)
  - 異なるサーバー上にソースおよびターゲット・データベース・ホームがあるAutoUpgrade
- AutoUpgradeメッセージおよびプロセスの説明用語
  - AutoUpgradeジョブIDの概要
  - AutoUpgradeステージの概要
  - AutoUpgradeステージの操作および状態の概要
- AutoUpgrade処理モードについて
  - AutoUpgrade処理モードを実行するための準備
  - AutoUpgrade分析処理モードについて
  - AutoUpgrade修正処理モードについて
  - AutoUpgradeデプロイ処理モードについて
  - AutoUpgradeアップグレード処理モードについて
- AutoUpgradeワークフローおよびステージの理解
- AutoUpgradeを使用した非CDBからPDBへのアップグレードの理解
- AutoUpgradeを使用した切断/接続アップグレードの理解
- AutoUpgradeのコマンドライン・パラメータおよびオプション
  - AutoUpgradeコマンドライン構文
  - debug
  - clear_recovery_data
  - config
  - config_values
  - console
  - create_sample_file
  - error_code
  - listchecks
  - load_password
  - mode
  - noconsole
  - preupgrade
  - settings
  - version
  - restore
  - restore_on_fail
  - zip
- AutoUpgradeユーティリティ構成ファイル
  - AutoUpgrade構成ファイルのローカルに変更可能なグローバル・パラメータ
    - defer_standby_log_shipping
    - dictionary_stats_after
    - dictionary_stats_before
    - drop_grp_after_upgrade
    - enable_local_undo

    - fixed_stats_before
    - manage_network_files
    - remove_underscore_parameters
    - restoration
    - target_base
    - target_home
    - target_version
- AutoUpgrade構成ファイルのローカル・パラメータ
    - add_after_upgrade_pfile
    - add_during_upgrade_pfile
    - after_action
    - before_action
    - catctl_options
    - checklist
    - close_source
    - del_after_upgrade_pfile
    - del_during_upgrade_pfile
    - drop_win_src_service
    - env
    - ignore_errors
    - keep_source_pdb
    - log_dir
    - manage_standbys_clause
    - pdbs
    - raise_compatible
    - remove_rac_config
    - remove_underscore_parameters
    - replay
    - restoration
    - revert_after_action
    - revert_before_action
    - run_hcheck
    - run_utlrp
    - sid
    - skip_tde_key_import
    - source_base
    - source_dblink
    - source_home
    - source_ldap_admin_dir
    - source_tns_admin_dir
    - start_time
    - target_base
    - target_cdb

        - target_pdb_copy_option=file_name_convert
        - target_pdb_name
        - target_ldap_admin_dir
        - target_tns_admin_dir
        - timezone_upg
        - tune_setting
        - upgrade_node
    - AutoUpgradeユーザー構成ファイルのグローバル・パラメータ
        - add_after_upgrade_pfile
        - add_during_upgrade_pfile
        - after_action
        - autoupg_log_dir
        - before_action
        - catctl_options
        - del_after_upgrade_pfile
        - del_during_upgrade_pfile
        - drop_grp_after_upgrade
        - keystore
        - raise_compatible
        - replay
        - target_base
        - target_home
        - target_version
        - upgradexml
- AutoUpgradeおよびOracle Databaseの構成オプション
    - 非CDBからPDBへのアップグレードのガイドラインおよび例
    - Oracle Grid Infrastructure管理構成のAutoUpgradeプロセス・フロー
    - AutoUpgradeを使用したアップグレードのためのOracle RACの要件
    - AutoUpgradeを使用したOracle RACのアップグレードの準備
    - AutoUpgradeパッチ適用
        - AutoUpgradeによるAutoUpgradeパッチ適用の実行方法
        - AutoUpgradeパッチ適用の構成ファイルおよびログ・ファイル
    - AutoUpgradeとOracle Data Guard
        - AutoUpgradeによるOracle Data Guardのアップグレードの実行方法
        - Oracle Data Guardのアップグレードのために AutoUpgradeで実行されるステップ
        - プライマリ・データベースのアップグレード後のステップ
    - 高速デプロイ・オプションを使用してAutoUpgradeを実行する方法
    - 暗号化されたPDBの切断/接続アップグレードの実行方法
    - 暗号化されたPDBの非CDBからPDBへの変換を実行する方法
- AutoUpgrade構成ファイルの例
    - AutoUpgradeの構成ファイルの作成
    - AutoUpgradeを使用したアップグレード時のTDEウォレット・ストアの場所の更新
    - 2つのデータベース・エントリを持つAutoUpgrade構成ファイル

        - AutoUpgrade構成ファイル・エントリを使用したアップグレードの標準化
        - 一連のPDBの増分アップグレード用のAutoUpgrade構成ファイル
        - スクリプト・ジョブまたはバッチ・ジョブでのAutoUpgradeの実行方法
        - AutoUpgradeによる切断/接続再配置アップグレード
        - AutoUpgrade構成ファイルを使用した修正およびチェックの無視
        - AutoUpgradeを使用したカスタム・スクリプトの実行
    - AutoUpgrade内部設定構成ファイル
    - AutoUpgradeログ・ファイルの構造
    - AutoUpgradeのための完全なデプロイの有効化
    - AutoUpgradeコンソールの使用例
    - AutoUpgradeの既知の制限
        - AutoUpgradeおよびディスク領域の問題
    - AutoUpgradeによるデータベース変更の適切な管理
    - デフォルト修正のオーバーライド方法
    - ローカル構成ファイルのパラメータ修正チェックリストの例
    - AutoUpgradeおよびMicrosoft WindowsのACLとCLI
- 4 パラレル・アップグレード・ユーティリティを使用したOracle Databaseの手動アップグレード
    - パラレル・アップグレード・ユーティリティを使用した手動アップグレード
        - 「Oracle Databaseのパラレル・アップグレード・ユーティリティについて(CATCTL.PLおよびDBUPGRADE)」
        - パラレル・アップグレード・ユーティリティを実行するための一般的なステップ
        - パラレル・アップグレード・ユーティリティ(catctl.pl)のパラメータ
        - パラレル・アップグレード・ユーティリティの使用例
    - マルチテナント・アーキテクチャのOracle Databaseの手動アップグレード・シナリオ
        - Oracle Multitenant Oracle Databaseのアップグレードについて
        - マルチテナント・アップグレードを使用したプロキシPDBのアップグレードの調整
        - マルチテナント・コンテナOracle Database (CDB)の手動でのアップグレード
        - 優先度リスト付きのパラレル・アップグレード・ユーティリティを使用したPDBのアップグレードについて
        - 優先度リスト、包含リストおよび除外リストを使用したPDBのアップグレードについて
        - マルチテナント・アーキテクチャのパラレル・アップグレード
            - プラガブル・データベース(PDB)のパラレル・アップグレードについて
            - マルチテナント・コンテナ・データベースのパラレル・アップグレード
        - 切断/接続を使用したマルチテナント・アーキテクチャの順次アップグレード
            - プラガブル・データベース(PDB)の順次アップグレードについて
            - 以前のリリースのPDBの以前のリリースのCDBからの切断
            - 以前のリリースのPDBの新しいリリースのCDBへの接続
            - 以前のリリースのPDBの新しいリリースへのアップグレード
            - PDBのアップグレードに包含リストまたは除外リストを使用する
    - Windowsでの非CDBのOracle Database の手動アップグレードの例
        - コマンドライン・ユーティリティを使用した、Windowsでの非CDBのアップグレードの準備
        - コマンドライン・ユーティリティを使用した、Windowsでの非CDBの手動アップグレード
    - DbupgradeスクリプトおよびOracle Databaseのcatupgrd.sqlについて
    - データベースのトランスポートおよびアップグレードについて(フル・トランスポータブル・エクスポート/インポート)

- 非CDB Oracle Databaseのアップグレード・シナリオ
    - 最新のAutoUpgradeユーティリティのダウンロード
    - PDBプラグインを使用したPDBとしての非CDBの採用について
    - PDBとしての非CDBの適用
    - マルチテナント環境におけるOracle Label Security統合
    - 非CDBアーキテクチャのOracle Databaseの手動アップグレード
    - フリート・パッチ適用およびプロビジョニングを使用したOracle Databaseのアップグレード
- Oracle Databaseのアップグレードの再実行
    - Oracle Databaseのアップグレードの再実行について
    - アップグレード(catctl.pl)・スクリプトを使用したアップグレードの再実行
    - マルチテナント・データベース(CDB)のアップグレード再実行のオプション
        - CDBのアップグレード全体の再実行
        - 指定したPDBでのみのアップグレードの再実行
        - 他のPDBがオンラインの間にアップグレードを再実行する
        - CDBまたはPDBを指定するための包含リストを使用したアップグレードの再実行
    - -pを使用して失敗したアップグレードを特定のフェーズから再開する
        - 失敗したフェーズのCDBログ・ファイルの表示
        - 失敗したフェーズからのマルチテナント・アップグレードの特定および再開の手順
- 5 Oracle Databaseのアップグレードのトラブルシューティング
    - Oracle Databaseのアップグレード・モードでの起動について
    - 異なるORACLE_HOME所有者でのDBUAの実行
    - 無効なオブジェクトの警告およびDBAレジストリ・エラー
    - 無効なオブジェクトおよび時期尚早なアップグレード後ツールの使用
    - Oracle Databaseアップグレード・スクリプトの終了エラーの解決
    - Oracle Databaseのアップグレード中におけるリソース制限エラーの原因のトラブルシューティング
    - Oracle DatabaseでのSQL*Plusエディション・セッションの起動エラーの解決
    - エラーORA-00020 utlrp.sqlの実行中に処理の最大数が超過する
    - 「ORA-28365: ウォレットがオープンしていません」エラーの修正
    - ビューCDB_JAVA_POLICYの問題の解決
    - サーバーの再起動後のアップグレードの継続(ADVM/ACFSドライバ・エラー)
    - コンポーネント・ステータスおよびアップグレード
        - アップグレード後の状態ツールを使用したコンポーネント・ステータスの理解
        - OPTION OFFコンポーネントのステータスおよびアップグレード
        - アップグレード・サマリー・レポートの例
    - Standard Editionの初期データベースおよびステータスがOPTION OFFのコンポーネント
    - アップグレード後のOracle ASMパスワード・ファイルの場所の調整
    - プラガブル・データベースのアップグレードでの「警告: XDBは現在無効です」エラーの修正
    - ORA-27248: 「sys.dra_reevaluate_open_failuresが実行されています」の修正
    - Datapatchのみが失敗した場合の失敗したアップグレードの修正
- 6 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
    - データベース内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 RACクラスタ・メンバーのアップグレードでのリスナー・ファイルの場所の更新
    - 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・アップグレードについて
    - 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 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
    - 以前のリリースのOracle APEXの削除
    - 読取り専用OracleホームでのDEMOディレクトリの置換
    - 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
    - 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
    - アップグレード後修正スクリプトの実行
    - 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
    - Oracle Grid Infrastructure、Oracle ASMおよびOracle Clusterwareの理解
    - Oracle Grid InfrastructureのインストールとアップグレードおよびOracle ASM
    - 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
    - LONGデータ型からLOBデータ型への表の移行
    - アップグレードしたOracle Databaseの統合監査の使用への移行
        - Oracle Databaseでの統合監査の移行プロセスの理解
        - Oracle Databaseでの統合監査への移行
        - 統合監査への移行後の古い監査レコードの管理について
        - 統合監査機能の削除

- 統合監査を使用しない場合のドキュメント参照の取得
- 再構築対象のOracle Text索引の識別
- DBMS_SCHEDULERジョブの削除および再作成
- アップグレード後の統合監査レコードの転送
- アップグレード後の監査レコードの転送について
- アップグレード後の監査レコードの転送
- アップグレードした本番Oracle Databaseのテストについて
- Oracle RACデータベースのアップグレード後の推奨作業
- Oracle ASMのアップグレード後の推奨作業
- ASMディスク・グループでの共有パスワード・ファイルの作成
- Oracle ASMパスワードのリセットによる大/小文字区別の強制
- Oracle ASMとOracle Databaseのディスク・グループの互換性の拡張
- Oracle ASMの優先読取りの障害グループの設定
- Oracle Database Express Editionのアップグレード後の推奨作業
- 手動によるOracle Databaseのアップグレード後にのみ行う作業
- Oracle付属アカウントのパスワードの変更
- サーバー・パラメータ・ファイルへの初期化パラメータ・ファイルの移行
- Oracle Textファイルを特定し、新しいOracleホームにコピーする
- Oracle Clusterware構成のアップグレード
- 新しいリリース用の初期化パラメータ・ファイルの調整
- アップグレード後のCOMPATIBLE初期化パラメータの設定
- アップグレード後のTNSNAMES.ORAおよびLISTENERパラメータの調整
- アップグレード後のOracle RACでのCLUSTER_DATABASE初期化パラメータの設定
- Oracle Databaseをアップグレードする場合のリリースの更新および要件
- 7 Oracle Databaseのアップグレード後のアプリケーションのアップグレード
- 新しいOracle Databaseリリースでのアプリケーションのアップグレードの概要
- Oracle Databaseの異なるリリースでのアプリケーションの互換性の問題
- Oracle Databaseでのソフトウェアのアップグレードおよびクライアントとサーバー構成
- Oracle Databaseで使用可能なクライアントおよびサーバーの構成
- Oracle Databaseのクライアントおよびサーバー・ソフトウェアのソフトウェア・アップグレードのタイプ
- Oracle Databaseクライアントまたはサーバー・ソフトウェアをアップグレードする場合のアプリケーションの互換性規則
- Oracle Databaseサーバー・ソフトウェアのアップグレードに対する規則
- クライアント環境を変更しない場合、再リンクは不要
- Oracle Databaseサーバーの新/旧リリースでアプリケーションの実行が可能
- Oracle Databaseクライアント・ソフトウェアのアップグレード
- イメージベースのOracle Databaseクライアント・インストールについて
- アプリケーションと新しいライブラリとのリンクについて
- 静的にリンクされたアプリケーションを常に再リンクさせる必要がある
- 動的にリンクされたアプリケーションの再リンクについて
- Oracle DatabaseでのプリコンパイラおよびOCIアプリケーションのアップグレードについて
- スキーマ専用アカウントおよびEXPIREDパスワード・アカウントのアップグレード
- OracleプリコンパイラおよびOCIアプリケーションのアップグレード・オプションについて

- 選択肢1: アプリケーションを変更しない
- 選択肢2: 新しいソフトウェアを使用してアプリケーションをプリコンパイルまたはコンパイルする
- オプション3: アプリケーション・コードを変更して、新しいOracle Databaseの機能を使用する
- OracleプリコンパイラおよびOCIアプリケーションの開発環境の変更
    - プリコンパイラ・アプリケーションの変更
    - OCIアプリケーションの変更
- Oracle DatabaseのアップグレードのSQL*PlusスクリプトおよびPL/SQLのアップグレード
- Oracle FormsまたはOracle Developerアプリケーションのアップグレードについて

- 8 以前のリリースへのOracle Databaseのダウングレード
    - Oracle Databaseのダウングレードがサポートされているリリース
    - プライマリでスタンバイ・データベースをダウングレードする準備
    - Oracle Databaseのダウングレード時のCOMPATIBLEパラメータの確認
    - Oracle Databaseのダウングレード前の全体バックアップの実行
    - Oracle Databaseをダウングレードする前に必要なステップの実行
    - スクリプトを使用したCDBまたは非CDBのOracle Databaseのダウングレード
        - Dbdowngradeを使用したCDBおよび非CDBのOracle Databaseのダウングレード
        - catdwgrd.sqlを使用したCDBまたは非CDBのOracle Databaseの手動ダウングレード
    - 単一のプラガブルOracle Database (PDB)のダウングレード
    - Oracle Application Expressが含まれるPDBのダウングレード
    - Oracle Databaseをダウングレードする場合のダウングレード後の作業
        - ダウングレード後のリリース更新およびその他のパッチの再適用
        - Oracle Databaseのダウングレード後のOracle RACの再有効化
        - Oracle Databaseのダウングレード後のOracle Database Vaultの再有効化
        - Oracle Clusterwareの構成のリストア
        - Oracle Databaseのダウングレード後のOracle Enterprise Managerのリストア
            - ダウングレード後のOracle Enterprise Managerのリストアの要件
            - EMCAの実行によるダウングレード後のOracle Enterprise Managerのリストア
                - Oracle ASMがない単一インスタンスのOracle Databaseでのemcaの実行
                - Oracle ASMがないOracle RACデータベースでのEMCAの実行
                - 単一インスタンスのOracle ASMインスタンスでのEMCAの実行
                - Oracle RACインスタンス上のOracle ASMでのemcaの実行
                - Oracle ASMがある単一インスタンスのOracle Databaseでのemcaの実行
                - Oracle RACデータベースおよびOracle ASMインスタンスでのemcaの実行
            - Enterprise Manager Database Controlをリストアするためのemdwgrdユーティリティの実行
        - 以前のリリースへのOracle APEXのリストア
        - ダウングレード後のディクショナリ統計の収集
        - ダウングレード後の修正したオブジェクト統計の再収集
        - ダウングレード後の失効しているCBO統計の再収集
        - ダウングレード後のレジストリ・コンポーネントの有効性の確認
    - Oracle Databaseのダウングレードのトラブルシューティング
        - catdwgrd.sqlスクリプトを使用したOracle Databaseコンポーネントのダウングレード中のエラー
        - アップグレードの成功または失敗後のOracle Grid Infrastructure (Oracle Restart)のダウン

グレード
- Oracle Messaging Gatewayを使用したデータベースのダウングレード・エラー
- 9 Oracle Data Pumpを使用したデータの移行
  - データの移行でのデータ・ポンプおよびエクスポート/インポートの概要
  - アップグレード前のOracle Data Pumpによるデータの移行
  - ネットワーク・リンクを使用したOracle Database全体のインポート
  - Oracle Databaseのダウングレード時のOracle Data Pumpの要件
- 10 Oracle Databaseの変更、サポート終了および非推奨
  - 非推奨およびサポート対象外のステータスについて
  - サポート終了および非推奨のリリース通知日
  - ORADiffツールを使用したリリース変更の検索
  - Oracle Database 23cの動作の変更、サポート終了および非推奨
    - Oracle Database 23cアップグレード計画の動作の変更
      - Oracle Spatialおよび非推奨オブジェクト
      - AutoUpgradeのREST API
      - XML JSON検索索引の拡張機能
      - SQL/JSONファンクションJSON_VALUEとブール型のJSON値
      - アップグレード前のFIPS内の非AESアルゴリズムからの移行
      - 読取り専用のOracleホームについて
      - SYSDATEおよびSYSTIMESTAMPがPDBタイム・ゾーンを反映
      - 4バンド・ラスター・ブロックでのOracle Spatial GeoRaster JPEG圧縮
    - Oracle Database 23cでサポートが終了した機能
      - ODP.NET OracleConfiguration.DirectoryTypeプロパティおよび.NET構成ファイルのDIRECTORY_TYPE設定のサポート終了
      - 元のエクスポート・ユーティリティ(EXP)のサポート終了
      - Oracle用MySQLクライアント・ライブラリ・ドライバのサポート終了
      - ACFSUTIL REPL REVERSEのサポート終了
      - クラスタ・ドメイン - ドメイン・サービス・クラスタのサポート終了
      - 適応しきい値機能DBSNMPのDBSNMPパッケージのサポート終了
      - ポリシー管理型のデータベース・デプロイメントのサポート終了
      - エンタープライズ・ユーザー・セキュリティ・ユーザー移行ユーティリティのサポート終了
      - Oracle Enterprise Manager Database Expressのサポート終了
      - Oracle Wallet Manager (OWM)のサポート終了
      - RASADMのサポート終了
      - Oracle Label Securityのパラメータおよびファンクションのサポート終了
      - Oracle Internet DirectoryとOracle Label Securityのサポート終了
      - RADIUSユーザーへの管理権限の付与のサポート終了
      - Transparent Data Encryption PKIキーのサポート終了
      - GOSTおよびSEED TDE暗号化アルゴリズムのサポート終了
      - Oracle Database 10Gパスワード検証機能のサポート終了
      - Transport Layer Securityバージョン1.0および1.1のサポート終了
      - Unix Crypt (MD5crypt)パスワード検証機能のサポート終了
      - FIPS強度80の暗号化のサポート終了

    - Diffie-Hellman匿名暗号のサポート終了
    - Oracle Database Extensions for .NETのサポート終了
    - Quality of Service Managementのサポート終了
    - 従来の監査のサポート終了
    - config.shのサポート終了
    - OLS表のLABELS列のサポート終了
    - 32ビットのOracle Databaseクライアントのサポート終了
    - Oracle Sharding高可用性のためのOracle GoldenGateレプリケーションのサポート終了
    - グリッド・インフラストラクチャ管理リポジトリ(GIMR)のサポート終了
    - OpatchおよびOPatchAutoを使用したアウトオブプレース・パッチ適用のサポート終了
    - データ・リカバリ・アドバイザ(DRA)のサポート終了
    - DBUAおよび手動アップグレード方法のサポート終了
    - Oracle OLAPのサポート終了
- Oracle Database 23cでサポートが終了したパラメータ
    - SYS_CONTEXT USERENVのEXTERNAL_NAMEパラメータのサポート終了
    - サービス属性値SESSION_STATE_CONSISTENCY = STATICのサポート終了
    - Oracle Label Securityのパラメータおよびファンクションのサポート終了
    - ENCRYPTION_WALLET_LOCATIONパラメータのサポート終了
    - ADD_SSLV3_TO_DEFAULT SQLNET.ORAパラメータ(およびSSLv3)のサポート終了
- Oracle Database 23cで非推奨となった機能
    - PROXY_ONLY_CONNECTの非推奨
    - Oracle Database 11gのSHA-1検証機能の非推奨
    - RFC-2138に基づくRADIUS APIの非推奨
    - エンタープライズ・ユーザー・セキュリティ(EUS)の非推奨
    - 自動ログイン・ウォレット・バージョン6の非推奨
    - WALLET_LOCATIONパラメータの非推奨
    - Fusion Middlewareに対するRAS中間層セッションのサポートの非推奨
    - Oracle Data Provider for .NET管理対象外ドライバの非推奨
    - GOSTおよびSEEDアルゴリズムの非推奨
    - Oracle Persistent Memoryの非推奨
    - Oracle Memory Speed File SystemでのPMEMサポートは非推奨
    - 部分DN一致のサービス名およびサーバー専用の証明書チェックの非推奨
    - mkstoreコマンドライン・ユーティリティの非推奨
    - ネットワーク・データ・モデル(NDM) XML APIは非推奨
    - SDO_GEOR_ADMINパッケージ内の2つのサブプログラムが非推奨
    - SDO_GEORパッケージ内のSDO_GEOR.importFromおよびSDO_GEOR.exportToサブプログラムは非推奨
    - Oracle Data GuardのSWITCHOVERコマンドのWAITオプション
    - DBMS_RESULT_CACHEファンクション名の非推奨
    - Oracle ACFSスナップショットの再マスタリングの非推奨
    - Oracle ACFS圧縮の非推奨

        - Oracle Virtual DirectoryとReal Application Securityの非推奨
        - Oracle TextのBIG_IO属性の非推奨
        - Oracle TextのASYNCHRONOUS属性の非推奨
        - Oracle Text CTXCAT索引タイプの非推奨
        - Oracle XML DBリポジトリの非推奨
        - DBMS_XMLGENの非推奨
        - DBMS_XMLSTOREの非推奨
        - 非構造化XML索引の非推奨
        - DBMS_HANG_MANAGERパッケージの非推奨
        - 停止時間ゼロのアップグレード(ZDU)の非推奨
        - グリッド・ネーミング・サービス(GNS)の非推奨
        - 従来の監査パッケージおよびファンクションの非推奨
    - Oracle Database 23cで非推奨になったビュー
        - V$DATABASEビューのV$DATABASE.FS_FAILOVER列の非推奨
        - V$PQ_SLAVEビューの非推奨
        - V$FS_FAILOVER_STATSビューの非推奨
        - DBA_HANG_MANAGER_PARAMETERSデータ・ディクショナリ・ビューの非推奨
        - V$RECOVERY_SLAVEビューの非推奨
    - Oracle Database 23cで非推奨になったパラメータ
        - ENCRYPT_NEW_TABLESPACESの非推奨
        - ALLOW_MD5_CERTSおよびALLOW_SHA1_CERTS sqlnet.oraパラメータの非推奨
        - MY_WALLET_DIRECTORY接続文字列の非推奨
        - FIPSパラメータの非推奨
        - ONE_STEP_PLUGIN_FOR_PDB_WITH_TDE初期化パラメータの非推奨
        - 従来の監査初期化パラメータの非推奨
        - PRE_PAGE_SGA初期化パラメータの非推奨
- Oracle Database 21cの動作の変更、サポート終了および非推奨
    - Oracle Database 21cアップグレード計画の動作の変更
        - Oracle Database 21cの読取り専用Oracle Homesについて
        - Oracle Database 21cでのみのマルチテナントのアップグレード
        - ロジカル・スタンバイと新しいデータ型
        - HRサンプル・スキーマの再配置
        - PDBでのDRCPの管理
        - Oracle Advanced Cluster File System (Oracle ACFS)の名前の変更
        - Windows認証でNTLMがデフォルトで使用されなくなった
    - Oracle Database 21cでサポートが終了した機能
        - DBMS_OBFUSCATION_TOOLKITパッケージのサポート終了
        - 複数のXML Database (XDB)機能のサポート終了
        - DBMS_LOB.LOADFROMFILEおよびLOBバッファリングのサポート終了
        - Oracle Data Guard Brokerプロパティおよびロジカル・スタンバイのサポート終了
        - DBMS_CRYPTO_TOOLKIT_TYPESおよびDBMS_CRYPTO_TOOLKITのサポート終了

    - 非CDB Oracle Databaseのサポート終了
    - クラスタ・ドメインのメンバー・クラスタのサポート終了
    - Unicode照合アルゴリズム(UCA) 6.1照合のサポート終了
    - Microsoft WindowsでのACFSの使用のサポート終了
    - Oracle ACFSセキュリティ(Vault)およびACFS監査のサポート終了
    - メンバー・クラスタでのOracle ACFSの使用(ACFSリモート)のサポート終了
    - SolarisおよびWindowsでのACFS暗号化のサポート終了
    - ACFSレプリケーションREPV1のサポート終了
    - Oracle Clusterwareとベンダー・クラスタウェアの統合のサポート終了
    - VERIFY_FUNCTIONおよびVERIFY_FUNCTION_11Gのサポート終了
    - 非推奨となったOracle Database Vaultロールのサポート終了
    - 匿名RC4暗号スイートのサポート終了
    - Adobe FlashベースのOracle Enterprise Manager Expressのサポート終了
    - インテリジェント・データ配置(IDP)のサポート終了
    - XML DB Content Connectorのサポート終了
    - DBMS_XMLSAVEのサポート終了
    - DBMS_XMLQUERYのサポート終了
    - FIPS保護およびプロセス強度0のサポート終了
    - PDBフラット・ファイル・ディクショナリ・ダンプのサポート終了
    - Oracle Fail Safeのサポート終了
- Oracle Database 21cでサポートが終了したビュー
    - V$OBJECT_USAGEビューのサポート終了
- Oracle Database 21cのサポートが終了した初期化パラメータ
    - UNIFIED_AUDIT_SGA_QUEUE_SIZEのサポート終了
    - パスワードのIGNORECASEパラメータのサポート終了
    - DISABLE_DIRECTORY_LINK_CHECKのサポート終了
    - REMOTE_OS_AUTHENTパラメータのサポート終了
    - SEC_CASE_SENSITIVE_LOGONのサポート終了
    - CLUSTER_DATABASE_INSTANCESパラメータのサポート終了
- Oracle Database 21cで非推奨となった機能
    - AUTO OPTIMIZEフレームワークの非推奨
    - CTXFILTERCACHE問合せ演算子の非推奨
    - ポリシー管理型データベースの非推奨
    - 従来の監査の非推奨
    - 古い暗号化アルゴリズムの非推奨
    - クラスタ・ドメインの非推奨 - ドメイン・サービス・クラスタ
    - エンタープライズ・ユーザー・セキュリティ(EUS)ユーザー移行ユーティリティの非推奨
    - ロジカル・スタンバイと新しいデータ型
    - シャード・キューの非推奨
    - Oracle用MySQLクライアント・ライブラリ・ドライバの非推奨
    - TLS 1.0および1.1トランスポート層セキュリティの非推奨
    - Unix Crypt (またはMD5crypt)パスワード検証機能の非推奨
    - ODP.NET OracleConfiguration.DirectoryTypeプロパティの非推奨

        - 脆弱な暗号化キー強度の非推奨
        - 適応しきい値機能のDBSNMPパッケージの非推奨
        - Oracle Sharding高可用性のためのOracle GoldenGateレプリケーションの非推奨
        - アウトバウンドTLS接続での匿名暗号スイートの非推奨
        - KERBEROS5PREアダプタの非推奨
        - Oracle Wallet Managerの非推奨
        - Oracle Enterprise Manager Database Expressの非推奨
        - サービス属性値SESSION_STATE_CONSISTENCY = STATICの非推奨
        - SQLNETおよびDBMS_CRYPTOに対するSHA-1の使用の非推奨
        - ACFSUTIL REPL REVERSEの非推奨
        - Oracle OLAPの非推奨
        - Oracle Database Extensions for .NETの非推奨
        - リポジトリ・イベントの非推奨
        - Quality of Service Managementの非推奨
        - グリッド・インフラストラクチャ管理リポジトリの非推奨
    - Oracle Database 21cで非推奨になったビュー
        - 従来の監査ビューの非推奨
    - Oracle Database 21cで非推奨になったパラメータ
        - 従来の監査初期化パラメータの非推奨
- Oracle Database 19cの動作の変更、サポート終了および非推奨
    - Oracle Database 19cアップグレード計画の動作の変更
        - リリース更新(RU)に含まれるすべてのタイム・ゾーン・ファイル(DST)
        - Oracle Data Guardのプロパティ管理の変更点
        - 高速ホーム・プロビジョニング(RHP)の名前の変更
        - OCRおよび投票ディスクの直接ファイル配置の再サポート
        - グリッド・インフラストラクチャ管理リポジトリのオプションのインストール
        - DBMS_JOBのサポート
        - Standard Edition High Availabilityについて
        - DBMS_OPTIM_BUNDLEによる「インストールされているが無効になっている」モジュールのバグ修正の管理
        - Windows認証でNTLMがデフォルトで使用されなくなった
    - Oracle Database 19cでサポートが終了した機能
        - Oracle Data Provider for .NETの昇格可能なトランザクション設定のサポート終了
        - Oracle Multimediaのサポート終了
        - LogMinerのCONTINUOUS_MINE機能のサポート終了
        - 拡張データ型サポート(EDS)のサポート終了
        - Data Guard BrokerのMaxConnectionsプロパティのサポート終了
        - フレックス・クラスタ・アーキテクチャでのリーフ・ノードのサポート終了
        - Oracle Streamsのサポート終了
        - PRODUCT_USER_PROFILE表のサポート終了
        - Standard Edition 2 (SE2)データベース・エディション用のOracle Real Application Clustersのサポート終了
    - Oracle Database 19cでサポートが終了したパラメータ

        - EXAFUSION_ENABLED初期化パラメータのサポート終了
        - LOG_ARCHIVE_DEST_nのMAX_CONNECTIONS属性のサポート終了
        - O7_DICTIONARY_ACCESSのサポート終了
        - OPTIMIZE_PROGRESS_TABLEパラメータのサポート終了
    - Oracle Database 19cで非推奨となった機能
        - URL_DATASTOREテキスト型の非推奨
        - FILE_DATASTORE型の非推奨
        - Oracle Data Guard Brokerの非推奨となったプロパティ
        - Oracle Data Guardのロジカル・スタンバイ・プロパティの非推奨となった機能
        - コマンドラインのASMCMD PWCREATEの非推奨
        - Addnodeスクリプトの非推奨
        - clone.plスクリプトの非推奨
        - Oracle Fail Safeの非推奨
        - GDSCTLオペレーティング・システム・コマンドライン・パスワード・リセットの非推奨
        - Oracle Enterprise Manager Expressの非推奨
        - DV_REALM_OWNERロールの非推奨
        - DV_REALM_RESOURCEロールの非推奨
        - DV_PUBLICロールの非推奨
        - Oracle ACFSレプリケーション・プロトコルREPV1の非推奨
        - Oracle OLAPの非推奨
        - SolarisおよびWindowsでのOracle ACFS暗号化の非推奨
        - WindowsでのOracle ACFSの使用の非推奨
        - Oracle ACFSセキュリティ(Vault)およびACFS監査の非推奨
        - メンバー・クラスタでのOracle ACFSの使用(ACFSリモート)の非推奨
        - クラスタ・ドメインの非推奨 - メンバー・クラスタ
        - Oracle Clusterwareとベンダー・クラスタウェアの統合の非推奨
        - Oracle Clusterwareを使用したブラック・ボックス仮想マシン管理の非推奨
        - OPatchおよびOPatchAutoアプトオブプレース・パッチ適用の非推奨
        - データ・リカバリ・アドバイザ(DRA)の非推奨
        - DBUAおよび手動アップグレードの非推奨
    - Oracle Database 19cで非推奨となった初期化パラメータ
        - 非推奨となったCLUSTER_DATABASE_INSTANCES初期化パラメータ
        - SQLNET.ENCRYPTION_WALLET_LOCATIONパラメータの非推奨
        - SERVICE_NAMES初期化パラメータの非推奨
- Oracle Database 18cの動作の変更、サポート終了および非推奨
    - Oracle Database 18cアップグレード計画の動作の変更
        - 簡素化されたイメージベースのOracle Databaseのインストール
        - 64文字を超えるJSONキー名の索引付けのサポート
        - 既存のデータベースのアップグレードにかわるイメージ・インストール
        - RPMベースのOracle Databaseのインストールについて
        - Oracle Text索引のトークン制限
        - /ALL/USER/DBA_ユーザー・ビューおよびPL/SQL外部ライブラリの変更
        - シンボリック・リンクおよびUTL_FILE

        - DBCAを使用したリスナーの直接登録の非推奨
        - INIT.ORA内のUNIFORM_LOG_TIMESTAMP_FORMATの変更
    - Oracle Database 18cでサポートが終了した機能
        - Oracle Administration Assistant for Windowsのサポート終了
        - Oracle Multimedia DICOMのサポート対象外機能
        - Oracle Multimedia Javaクライアント・クラスのサポート終了
        - Oracle XML DBのサポート対象外機能
        - ODP.NET管理対象ドライバの分散トランザクションのDLLのサポート終了
        - Data Guard Broker DGMGRL ALTER構文のサポート終了
        - Microsoft WindowsシステムでのCRSUSERのサポート終了
    - Oracle Database 18cのサポートが終了した初期化パラメータ
        - STANDBY_ARCHIVE_DEST初期化パラメータのサポート終了
        - UTL_FILE_DIR初期化パラメータのサポート終了
    - Oracle Streamsの最終リリース
    - Oracle Database 18cで非推奨となった機能
        - Data GuardのMAX_CONNECTIONS属性の非推奨化
        - 拡張データ型サポート(EDS)の非推奨化
        - DBMS_DATA_MININGパッケージのGET_*ファンクションの非推奨化
        - パッケージDBMS_XMLQUERYの非推奨化
        - パッケージDBMS_XMLSAVEの非推奨化
        - Oracle Label Securityビューで非推奨となった列
        - NUMBERを使用したTrueまたはFalseのJSON戻り値の非推奨化
        - Oracle TextのMAIL_FILTERの非推奨
        - asmcmd showversionオプションの非推奨
        - Oracle TextのNEWS_SECTION_GROUPの非推奨
        - Oracle Net ServicesのSDPのサポートの非推奨化
        - フレックス・クラスタ(ハブ/リーフ)アーキテクチャの非推奨
        - PRODUCT_USER_PROFILE表の非推奨
        - Oracle Multimediaの非推奨
- Oracle Database 12cリリース2 (12.2)の動作の変更、サポート終了および非推奨
    - Oracle Database 12cリリース2 (12.2)での動作の変更
        - Oracle Database 12cリリース2 (12.2)での初期化パラメータのデフォルトの変更
        - Database Upgrade Assistant (DBUA)の機能拡張と変更点
        - Oracle Data Guard Brokerおよびローリング・アップグレードの拡張機能
        - Oracle DatabaseのデフォルトのSGA権限の変更について
        - ネットワーク・アクセス制御リストおよびOracle Database 12cへのアップグレード
        - パラレル・アップグレード・ユーティリティのバッチ・スクリプト
        - 統合監査のAUDIT_ADMINおよびAUDIT_VIEWERロールの変更
        - Oracleバッチ更新のバッチ・サイズ設定の無効化
        - Oracle管理タイプに依存する表のアップグレードについて
        - パスワードの大/小文字の区別およびORA-1017無効なユーザー名またはパスワード
        - Oracle Fleet Patching and Provisioningを使用したOracle Grid Infrastructureのデプロイについて

        - Zero Data Loss Recovery Applianceリリース12.1のバックアップに関する使用上の制限
        - クライアントおよびフォアグラウンド・サーバーのプロセス・メモリーの変更
    - Oracle Database 12cリリース2 (12.2)でサポートが終了した機能
        - アドバンスト・レプリケーションのサポート終了
        - OCRおよび投票ファイルのファイル・システムへの直接配置のサポート終了
        - JPublisherのサポート終了
        - preupgrd.sqlおよびutluppkg.sqlのサポート終了
        - トランザクション・ガード用Oracle Data Provider for .NET APIのサポート終了
        - Oracle Database 12cリリース2 (12.2)でのビューのサポート終了
        - Oracle Database内のSQLJのサポート
        - 一部のXML DB機能のサポート終了
    - Oracle Database 12cリリース2 (12.2)でサポートが終了した初期化パラメータ
    - Oracle Database 12cリリース2 (12.2)で非推奨となった機能
        - ALTER TYPE REPLACEの非推奨
        - configToolAllCommandsスクリプトの非推奨
        - DBMS_DEBUGパッケージの非推奨
        - インテリジェント・データ配置(IDC)の非推奨
        - CONTINUOUS_MINEオプションの非推奨
        - 非CDBアーキテクチャの非推奨
        - Oracle Administration Assistant for Windowsの非推奨
        - Oracle Data Provider for .NETのPromotableTransaction設定の非推奨
        - oracle.jdbc.OracleConnection.unwrap()の非推奨
        - oracle.jdbc.rowsetパッケージの非推奨
        - oracle.sql.DatumWithConnectionクラスの非推奨
        - Oracle Multimedia Java APIの非推奨
        - Oracle MultimediaのDICOMのサポートの非推奨
        - Multimedia SQL/MM Still Image標準サポートの非推奨
        - Unicode照合アルゴリズム(UCA) 6.1照合の廃止
        - UNIFIED_AUDIT_SGA_QUEUE_SIZEの非推奨
        - VERIFY_FUNCTIONおよびVERIFY_FUNCTION_11Gの非推奨
        - V$MANAGED_STANDBYの非推奨
        - 一部のXML DB関数の非推奨
        - Oracle XML Databaseの非推奨となった機能
    - Oracle Database 12cリリース2 (12.2)で非推奨となった初期化パラメータ
- Oracle Database 12cリリース1 (12.1)の動作の変更、サポート終了および非推奨
    - Oracle Database 12cリリース1 (12.1)の動作の変更
        - PARALLEL=NOなしで実行されたcatupgrd.sqlに関連するエラー
        - スタンドアロン削除ツールの変更点
        - セキュリティ監査機能の変更点
        - SQLNET.ALLOWED_LOGON_VERSIONパラメータが設定されていないシステムのアップグレード
        - Oracle Warehouse Builder (OWB)はOracle Databaseでインストールされない

    - Oracle Databaseリリース10.2または11.1およびOCFSとRAWデバイスのアップグレードについて
    - VARCHAR2、NVARCHAR2およびRAWデータ型の変更
    - RESOURCE_LIMITパラメータのデフォルトの変更
    - Oracle XML DBは必須でアンインストールできない
    - Oracle RACのデフォルトで有効になっているDirect NFS
- Oracle Database 12cリリース1 (12.1)でサポートが終了した機能
    - RAW記憶域デバイスのサポート終了
    - テキスト索引のALTER INDEX OPTIMIZEのサポート終了
    - CLEANUP_ORACLE_BASEプロパティのサポート終了
    - Oracle Enterprise Manager Database Controlのサポート終了
    - Secure Sockets Layer (SSL)の暗号スイートのサポート終了
    - データベースのRules Manager (RUL)およびExpression Filter (EXF)のサポート終了
    - OCFS用のcluvfy comp cfsのサポート終了
    - チェンジ・データ・キャプチャのサポート終了
    - Oracle Data Miningのサポートが終了した機能
    - 暗黙的接続キャッシュのサポート終了
    - Oracle Data Miningのアップグレードでの ABNモデルのサポート終了
    - OLAPカタログ(AMD)のサポート終了
    - OracleグローバリゼーションのCSSCANおよびCSALTERのサポート終了
    - Oracle Netの接続プーリングのサポート終了
    - Oracle Net Listenerパスワードのサポート終了
    - Oracle Namesのサポート終了
    - Oracle Net ServicesのOracle Names制御ユーティリティのサポート終了
    - Oracle TextおよびOracle XML DBでのCTXXPATHのサポート終了
    - Oracle Net ServicesのSQLNET.KERBEROS5_CONF_MITパラメータのサポート終了
    - テキスト索引のALTER INDEX OPTIMIZEのサポート終了
    - テキスト索引のSYNC [MEMORY memsize]のサポート終了
    - Microsoft Windowsプラットフォームでのサポートが終了した機能
        - WindowsでのOracle COM Automationのサポート終了
        - Oracle Objects for OLEのサポート終了
        - Oracle Counters for Windows Performance Monitorのサポート終了
        - WindowsでのOracle Cluster File System (OCFS)のサポート終了
- Oracle Database 12cリリース2 (12.2)で非推奨となった機能
    - 非CDBアーキテクチャの非推奨
    - catupgrd.sqlスクリプトの非推奨およびパラレル・アップグレード・ユーティリティの導入
    - 非推奨となったDELETE_CATALOG_ROLE
    - Oracle Label Securityで非推奨となった機能とパラメータ
    - 非推奨となったOracle Database VaultのAPI
    - 非推奨となったOracle Database Vaultのデフォルトのレルム
    - 非推奨となったOracle Database Vaultのデフォルトのルール・セット

- NTLMプロトコルを使用したWindows NTS認証の非推奨
- 透過的データ暗号化に対する公開キー・インフラストラクチャの非推奨
- Oracle Data Guard Brokerの非推奨となった機能
- 非推奨となったEndToEndMetrics関連のAPI
- Oracle Restartの非推奨
- QOSCTLサービスのクオリティ(QoS)コマンドの-checkpasswdの非推奨
- 非推奨となったOracle Net Services用のNT LAN Manager (NTLM)プロトコル
- Oracle Call Interfaceの非推奨となった機能
- Oracle Streamsの非推奨
- Oracle Data Provider for .NETの非推奨のプログラミング・インタフェース
- 非推奨となったOracle Databaseのセマンティク・テクノロジのVPDサポート
- Oracle Databaseのセマンティク・テクノロジのバージョン対応モデルのサポート
- 非推奨となったアドバンスト・レプリケーション
- 1文字SRVCTL CLIオプションの非推奨
- クラスタ管理での管理ユーザーの格納されたリストの非推奨
- サーバー内部SQLJの非推奨
- 非推奨となったOracleバッチ更新
- 非推奨となったEndToEndMetrics関連のAPI
- 非推奨となったストアド・アウトライン
- 非推奨となったoracle.sqlパッケージの具象クラス
- 非推奨となったdefineColumnTypeメソッド
- 非推奨となったCONNECTION_PROPERTY_STREAM_CHUNK_SIZEプロパティ
- Oracle Data Pumpエクスポート・ユーティリティの機能
- Oracle Databaseのセマンティク・テクノロジのバージョン対応モデルのサポート
- 非推奨となったdefineColumnTypeメソッド
- 非推奨となったOracle Net Services用のNT LAN Manager (NTLM)プロトコル
- Oracle Call Interfaceの非推奨となった機能
- クラスタ管理での管理ユーザーの格納されたリストの非推奨
- -cleanupOBaseの非推奨
- Oracle XML Databaseの非推奨となった機能
- 非推奨となったアドバンスト・レプリケーション
- Oracle Database 12cリリース1 (12.1)で非推奨となったOracle MultimediaのDICOM
- JPublisherの非推奨

- Oracle Database 12cリリース1 (12.1)で非推奨となった初期化パラメータ
  - 非推奨となったFILE_MAPPING初期化パラメータ
  - 非推奨となったRDBMS_SERVER_DN初期化パラメータ
  - IGNORECASEおよびSEC_CASE_SENSITIVE_LOGONの非推奨
  - SQLNET.ALLOWED_LOGON_VERSIONパラメータの非推奨
  - LOG_ARCHIVE_LOCAL_FIRST初期化パラメータのサポート終了
- Oracle Database 12cリリース1 (12.1)で非推奨となったビュー

- A AutoUpgrade REST API
  - AutoUpgrade REST APIの概要

        - AutoUpgrade REST APIについて
    - スタート・ガイド
        - cURLのインストール
        - AutoUpgrade REST APIを設定して使用する方法
        - AutoUpgrade REST API認証権限を使用する方法
    - AutoUpgradeのREST API
        - 新しいAutoUpgradeタスクの作成
        - AutoUpgradeタスクの進行状況レポート・リクエスト
        - AutoUpgradeタスクのコンソール・リクエスト
        - AutoUpgradeのタスク詳細リクエスト
        - AutoUpgradeタスクのタスク・リスト・リクエスト
        - AutoUpgradeタスクのログまたはログ・リスト
        - AutoUpgradeのタスク・リクエストのステータス・レポート
- B Oracle Databaseアップグレード・ユーティリティ
    - Oracle Databaseのアップグレード用スクリプト
    - アップグレード前情報ツールおよびAutoUpgradeのアップグレード前処理
        - アップグレード前情報ツールについて
        - アップグレード前情報ツールによって生成されるアップグレード前スクリプト
        - アップグレード前情報ツールによって生成されるアップグレード後スクリプト
        - 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(preupgrade.jar)のコマンド
        - アップグレード前情報ツールの出力
        - 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
            - 以前のOracle Databaseリリースからのパスワードが指定されたデータベース・リンクについて
            - TIMESTAMP WITH TIME ZONEデータ型に対するOracle Databaseの警告について
- C Oracle Database Upgrade Assistant (DBUA)を使用したアップグレード
    - DBUAを使用するための要件
    - アップグレード時のDBUAの停止について
    - DBUAによるOracle Databaseのアップグレードの処理方法
    - DBUAによって起動されるアップグレード・スクリプト
    - 既存のOracleホームからのデータベースの移動
    - 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
    - サイレント・モードのDBUAを使用したOracle Databaseのアップグレード
        - サイレント・モードでのDBUAの実行
        - アクティブ・モードおよびサイレント・モードでのDBUAのコマンドライン構文
- D 異なるORACLE_HOME所有者でのDBUAの実行
- 索引

例の一覧

- 2-1　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
- 2-2　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
- 2-3　SEC_CASE_SENSITIVE_LOGON SetがFALSEになっているものがないことの確認
- 2-4　分析モードでのAutoUpgradeの使用によるOracle Database 12c非CDBシステムのチェック
- 2-5　CDBおよびPDBのAutoUpgrade構成ファイル
- 2-6　優先度リスト・エミュレーションを使用したパラレル・アップグレード・ユーティリティの実行の例
- 2-7　Oracle Database Vaultを使用するデータベースのAutoUpgradeプロシージャ
- 3-1　新しい場所へのデータベースの移動後のターゲット・ホームでのAutoUpgradeの実行
- 3-2　非CDBからPDBへの変換のためのAutoUpgrade構成ファイル
- 3-3　AutoUpgrade切断/接続のアップグレード用構成ファイル
- 3-4　マルチテナント・アーキテクチャを使用した非CDBからOracle Database 19cへのアップグレードおよび変換
- 3-5　様々なパッチ適用シナリオ用のAutoUpgrade構成ファイル
- 3-6　AutoUpgradeパッチ適用のサマリー・ログ・ファイル
- 3-7　REDO転送サービスのステータスの確認
- 3-8　適用ラグおよびトランスポート・ラグの確認
- 3-9　AutoUpgradeを使用した切断/接続アップグレードによる暗号化されたPDBのアップグレード
- 3-10　TDEを使用したデータベースのアップグレードおよび非CDBからPDBへの変換
- 3-11
- 3-12　データ・ファイルに対してリフレッシュ率を使用しない非CDBおよびPDBの切断/接続再配置の構成ファイル
- 3-13　データ・ファイルに対してデプロイメント遅延およびリフレッシュ率を使用した非CDBおよびPDBの切断/接続再配置の構成ファイル
- 3-14　PDBの切断/接続再配置のための単純な構成ファイル
- 3-15　構成ファイルのbefore_actionおよびafter_actionの使用
- 3-16　構成ファイルのrevert_before_actionおよびrevert_after_actionの使用
- 3-17　AutoUpgradeを使用する前のアーカイブ・ロギングおよび高速リカバリ領域(FRA)の設定
- 3-18　パスワード・ファイルとセキュリティ・パスワード・ファイルの更新
- 3-19　透過的データ暗号化およびAutoUpgrade
- 3-20　AutoUpgradeコンソールを有効および無効にする方法の例
- 3-21　アップグレードの実行中のディスク領域エラー
- 3-22　アップグレード完了後のディスク領域エラー
- 3-23　分析モードでのAutoUpgradeユーティリティの起動
- 3-24　構成ファイルの新規チェックリストの作成
- 3-25　checklist.cfgの検索および編集
- 4-1　CDBおよび非CDBでのパラメータを使用したパラレル・アップグレード・ユーティリティの実行
- 4-2　パラレル・アップグレード・ユーティリティを使用した複数のプラガブル・データベース(PDB)に対するパラレル・アップグレードの実行
- 4-3　完全なPDBアップグレード優先度の指定
- 4-4　PDBの優先度サブセットの指定およびデフォルト処理による他のPDBのアップグレード
- 4-5　PDBの優先度サブセットの指定および包含リストによる他のPDBのアップグレード
- 4-6　PDBの優先度サブセットの指定および除外リストによるCDB$ROOTの除外

- 4-7　CATCTL_LISTONLYを使用した除外リストの指定
- 4-8　CON_ID値を使用した優先度リストの指定
- 4-9　デフォルトを使用したマルチテナント・アーキテクチャのアップグレードの例(パラメータ・セットなし)
- 4-10　64個のPDBのパラレル・アップグレード・プロセッサおよび4個のパラレルSQLプロセスを使用したマルチテナント・アーキテクチャのアップグレードの例
- 4-11　20個のPDBのパラレル・アップグレード・プロセッサおよび2個のパラレルSQLプロセスを使用したマルチテナント・アーキテクチャのアップグレードの例
- 4-12　10個のPDBのパラレル・アップグレード・プロセッサおよび4個のパラレルSQLプロセスを使用したマルチテナント・アーキテクチャのアップグレードの例
- 4-13　ユーザー環境変数のリセット
- 4-14　データベースへの接続のテスト
- 4-15　Oracle Enterprise Managerの手動での削除(em_present)
- 4-16　OLAPカタログの手動での削除(amd_exists)
- 4-17　ADMINISTER DATABASE TRIGGER権限の収集(trgowner_no_admndbtrg)
- 4-18　マテリアライズド・ビューの更新(mv_refresh)
- 4-19　Oracle Application Expressのアップグレード(apex_upgrade_msg)
- 4-20　コマンドライン・コマンドを使用したデータベース・サービスの停止
- 4-21　Microsoft Windows PowerShellスクリプトを使用したデータベース・サービスの停止
- 4-22　以前のリリースのOracleホームからのデータベース・サービスの削除
- 4-23　以前のリリースのOracleホームのリスナーの停止
- 4-24　新しいOracleホームへの環境変数の設定
- 4-25　新しいOracleホームへのPFILEのコピー、および新しいOracle Databaseバイナリを使用した新規サービスの作成
- 4-26　データベースのアップグレードの開始
- 4-27　アップグレード後チェックの実行
- 4-28　再開オプションを使用したアップグレードの再実行
- 4-29　除外リストを使用したアップグレードの再実行
- 4-30　再開オプションを使用したPDBでのアップグレードの再実行
- 4-31　除外リストを使用したPDBでのアップグレードの再実行
- 4-32　包含リストを使用したPDBでのアップグレードの再実行
- 5-1　アップグレード後の状態ツールのアップグレード・サマリー・レポート
- 6-1　dbupgdiag.sqlスクリプトの実行
- 6-2　DBA_REGISTRYでのSQL問合せの実行
- 6-3　無効なオブジェクトをチェックするSQL問合せの実行
- 6-4　残っている無効なオブジェクトの数
- 6-5　再コンパイルされたオブジェクトの数
- 6-6　エラー付きで再コンパイルされたオブジェクトの数
- 6-7　TNSNAMES.ORAおよびSQLNET.ORAを新しいデフォルト・ネットワーク管理ディレクトリにコピー
- 6-8　TNS_ADMIN環境変数の設定
- 6-9　以前のリリースのDemoディレクトリのコピーおよび読取り専用Oracleホームでのデモンストレーションのリフレッシュ
- 例6-10　非CDB Oracle Databaseのアップグレード後修正結果のスプーリングの例
- 例6-11　Linux、UNIXおよびWindowsシステムでcatcon.plを使用してアップグレード後修正を実行する方法

の例
- 8-1 CDB_REGISTRYビューを使用したCDBでのレジストリの確認
- 8-2 DBA_REGISTRYビューを使用した非CDBまたはPDBでのレジストリの確認
- 8-3 ORA-06512によるORA-20001エラー
- 10-1 UTL_FILEおよびシンボリック・リンクによるエラー・メッセージの例
- A-1 AutoUpgradeタスク・リクエストの発行
- A-2 問題のあるリクエスト後のAutoUpgradeジョブのレスポンス
- A-3 taskidを使用したAutoUpgradeジョブの再実行
- A-4 タスクの進行状況レポートのGETリクエストの発行
- A-5 タスクIDが無効なタスク進行状況レポート・リクエスト
- A-6 タスク・コンソールGETリクエストの発行
- A-7 タスクIDが無効なコンソール・リクエストのレスポンス・ヘッダー
- A-8 タスク詳細GETリクエストの発行
- A-9 taskidが無効なタスク詳細リクエスト
- A-10 タスク・リストのGETリクエストの発行
- A-11 タスク・ログのGETリクエストの発行
- A-12 タスクIDが無効なログ・リクエストのレスポンス・ヘッダー
- A-13 タスクIDログ・リストのレスポンス本文
- A-14 nameパラメータを使用した特定のログのリクエスト
- A-15 タスク・ステータスGETリクエストの発行
- A-16 taskidが無効なタスク・ステータスGETリクエストの発行
- B-1 ソースOracleホームの非CDBの例
- B-2 ソースOracleホームのCDB
- C-1 DBUAを使用するアップグレードのデータベースの選択
- C-2 非対話型(サイレント)オプションを使用したDBUAによるアップグレードのデータベースの選択
- C-3 非対話(「サイレント」)モードでのDBUAの実行のユース・ケース

図一覧

- 1-1　Oracle Databaseのアップグレード・ステップのワークフロー
- 1-2　Oracle Databaseのリリース番号の例
- 2-1　アップグレード・チェックの概要ファイルの例
- 3-1　AutoUpgradeを使用した非CDBの変換およびアップグレード
- 3-2　ソースからターゲットへの切断/接続アップグレード
- 4-1　DBMS_PDB.DESCRIBEプロシージャを使用した非CDBの接続

表一覧

- 1-1　Oracle Database 19cのアップグレード・パスの例
- 1-2　COMPATIBLE初期化パラメータ
- 1-3　アップグレード中のデータベース移行およびアップグレードの技術方法
- 3-1　Oracle Database AutoUpgradeユーティリティのグローバル構成パラメータ
- 3-2　AutoUpgradeの内部設定構成ファイル・パラメータ
- 4-1　パラレル・アップグレード・ユーティリティ(catctl.pl)のパラメータ
- 6-1　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
- 8-1　ダウングレードがサポートされているリリースおよびエディション
- 10-1　Oracle Data Guardのプロパティ名の変更
- 10-2　Oracle Label Securityビューで非推奨となった列
- B-1　アップグレード・スクリプト、アップグレード後スクリプトおよびダウングレード・スクリプト
- B-2　タイムゾーン・ファイルのバージョンの修正を行う場合の選択肢

はじめに

次のトピックでは、アップグレードの計画および手順の内容の範囲に関する情報について提供します。

このマニュアルでは、Oracle Databaseのアップグレードの計画と実行のプロセスについて説明します。また、このマニュアルでは、互換性、アプリケーションのアップグレード、および初期化パラメータの変更やデータ・ディクショナリの変更などの新しいOracle Databaseリリースにおける重要な変更に関する情報も提供します。

また、『*Oracle Database*アップグレード・ガイド』では、Oracle Database(Standard Editionともいう)およびOracle Database Enterprise Edition製品の様々な機能についても説明します。Oracle DatabaseおよびOracle Database Enterprise Editionには、同等の基本機能があります。ただし、Oracle Database Enterprise Editionでのみ使用可能な拡張機能もあります。これらの一部はオプションです。たとえば、アプリケーション・フェイルオーバーを使用するには、Oracle Real Application Clustersを備えたEnterprise Editionが必要です。

- 対象読者
- ドキュメントのアクセシビリティ
- ダイバーシティ&インクルージョン
- Javaアクセシビリティを実装するためのJava Access Bridgeの設定
  Microsoft Windowsシステムのアシスティブ・テクノロジがJava Accessibility APIを使用できるようにJava Access Bridgeをインストールします。
- 関連ドキュメント
- 表記規則

対象読者

『Oracle Databaseアップグレード・ガイド』は、Oracle Databaseのアップグレードの計画および実行を担当する、データベース管理者(DBA)、アプリケーション開発者、セキュリティ管理者、システム・オペレータを対象としています。

このマニュアルを使用するには、次の情報について詳しく理解しておく必要があります。

- リレーショナル・データベースの概要
- ご使用のOracle Databaseのリリース
- ご使用のオペレーティング・システム環境

親トピック: はじめに

ドキュメントのアクセシビリティ

オラクル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

サポートをご契約のお客様には、My Oracle Supportを通して電子支援サービスを提供しています。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

ダイバーシティ&インクルージョン

Oracleはダイバーシティ&インクルージョンに積極的に取り組んでいます。Oracleは、ソート・リーダーシップと革新性を高める社員の多様性を尊重し、その価値を重んじています。従業員、お客様、パートナー様にポジティブな影響をもたらすインクルーシブな文化を醸成する私たちのイニシアティブの一環として、製品やドキュメントからインセンシティブな用語を取り除くように努めています。また、Oracle製品および業界標準が進化する中、お客様の既存の技術との互換性を維持する必要性およびサービスの継続性確保の要求にも留意しています。このような技術的な制限により、当社のインセンシティブな用語を削除する取組みは継続中であり、時間と皆様のご協力が必要となります。

親トピック: はじめに

Java Accessibilityを実装するためのJava Access Bridgeのセットアップ

Microsoft Windowsシステムの支援テクノロジがJava Accessibility APIを使用できるように、Java Access Bridgeをインストールします。

Java Access Bridgeは、Java Accessibility APIを実装するJavaアプリケーションおよびアプレットをMicrosoft Windowsシステム上のユーザー補助テクノロジから可視にするためのテクノロジです。

Java Access Bridgeの使用に必要な支援テクノロジの最低限サポートされるバージョンの詳細は、*Java Platform, Standard Edition*アクセシビリティ・ガイドを参照してください。インストール手順とテスト手順、およびJava Access Bridgeの使用方法についてもこのガイドを参照してください。

関連項目

- Java Platform, Standard Edition Javaアクセシビリティ・ガイド

親トピック: はじめに

関連ドキュメント

追加情報については、次のドキュメント・リストを参照してください。

- このマニュアルで使用される概念および用語の概要については、『Oracle Database概要』を参照してください。
- Oracle Databaseの管理の詳細は、『Oracle Database管理者ガイド』を参照してください
- このリリースの新機能の詳細は、『Oracle Database新機能ガイド』を参照してください
- Oracle DatabaseのSQLコマンドおよびファンクションの詳細は、『Oracle Database SQL言語リファレンス』を参照してください
- Oracle Databaseにバンドルされるユーティリティの詳細は、『Oracle Databaseユーティリティ』を参照してください
- Oracle Net Servicesの詳細は、『Oracle Database Net Services管理者ガイド』を参照してください。

このガイドでは、Oracle Databaseのインストール時に「基本インストール」オプションを選択した場合に、デフォルトでインストールされるサンプル・スキーマを例として多用しています。これらのスキーマがどのように作成されているか、およびその使用方法については、次のガイドを参照してください。

Oracle Databaseサンプル・スキーマ

親トピック: はじめに

表記規則

このドキュメントでは、次のテキスト表記規則が使用されます:

表記規則	意味
boldface	太字体は、アクションに関連付けられたグラフィカル・ユーザー・インタフェース要素や、本文または用語集で定義されている用語を示します。
italic	イタリック体は、ブック・タイトル、強調、またはユーザーが特定の値を指定するプレースホルダー変数を示します。
`monospace`	等幅体は、段落内のコマンド、URL、サンプル内のコード、画面に表示されるテキスト、またはユーザーが入力するテキストを示します。

親トピック: はじめに

1 Oracle Databaseのアップグレードの概要

ご使用のデータベース環境に応じたアップグレード・オプションと計画、およびOracle Databaseのアップグレード処理を自動化するツールの配列が提供されています。

- Oracle Databaseのアップグレード・ツールおよびアップグレード処理の概要
  Oracle Databaseの用語、ツールおよびプロセスを理解するには、これらのトピックを参照してください。
- Oracle Databaseのアップグレード処理の主なステップ
  Oracle Databaseのアップグレードは、6つの主なステップで構成されます。
- Oracle Databaseリリース間の互換性および相互運用性
  Oracle Databaseのリリースの違いによって発生する可能性のある互換性と相互運用性の問題を理解および回避する方法について学習します。
- 複数のOracleリリースの実行について
  Optimal Flexible Architecture (OFA)を使用して複数のリリースを実行します。
- アップグレード中のデータベース変換について
  Oracle Databaseをアップグレードする際に選択する最適なパスを決定するには、これらのトピックを参照してください。
- 新しいOracle Databaseリリース用のプラットフォームのアップグレードについて
  新しいOracle Databaseリリース用にオペレーティング・システムまたはハードウェアをアップグレードする場合、これらのトピックを参照してください。
- イメージベースのOracle Databaseインストールについて
  イメージベースのインストールを理解して、Oracle Databaseソフトウェアのインストールおよび構成を簡略化します。

Oracle Databaseのアップグレード・ツールおよびアップグレード処理の概要

Oracle Databaseの用語、ツールおよびプロセスを理解するには、これらのトピックを参照してください。

- アップグレードおよび移行という用語の定義
  アップグレードと移行は、異なるタイプのデータベース変更です。
- アップグレードおよびデータ移行方法と処理
  オラクル社では、アップグレード処理を自動化し、アップグレードの効率的な完了を支援する機能および製品を提供しています。
- 直接のアップグレードをサポートするOracle Databaseのリリース
  最新のOracle Databaseリリースに直接アップグレードするためにサポートされるオプションを参照してください。
- Oracle Databaseのアップグレードに関する最新情報の入手先
  このドキュメント以外に、オラクル社では、そのサポート・サイトおよび`preupgrade`パラメータを使用したAutoUpgradeユーティリティを介してアップグレードに関する情報を提供しています。

親トピック: Oracle Databaseのアップグレードの概要

アップグレードおよび移行という用語の定義

アップグレードと移行は、異なるタイプのデータベース変更です。

アップグレードによって、既存のOracle Database環境(インストール済コンポーネントおよび関連するアプリケーション)が新しいリリースのOracle Database環境に変換されます。データベースのデータ・ディクショナリは、新しいリリースにアップグレードされます。アップグレードによってユーザー・データが直接影響を受けることはありません。アップグレード中に、データへのアクセス、データの変更または移動が行われることはありません。

データを移行することは、1つのOracle Databaseから、データの移行または移動用に事前に作成された別のデータベースにデータを移動することです。データを移行するのは、新しいハードウェアやオペレーティング・システム・プラットフォーム、または新しい文字セットにデータベース環境を移動する必要がある場合です。最新リリースにアップグレードすることは、移行には含まれません。アップグレード処理は、データの移行後に個別に行われます。

『*Oracle Database*アップグレード・ガイド』のアップグレード・ステップは、特に指定されていないかぎり、すべてのオペレーティング・システムに適用されます。オペレーティング・システムによっては、追加のアップグレード・ステップを実行する必要があります。

関連項目

- Oracle Databaseのインストレーション・ガイド
- Oracle Databaseユーティリティ

親トピック: Oracle Databaseのアップグレード・ツールおよびアップグレード処理の概要

アップグレードおよびデータ移行方法と処理

オラクル社では、アップグレード処理を自動化し、アップグレードの効率的な完了を支援する機能および製品を提供しています。

Oracle Databaseでは、データベースを新しいリリースにアップグレードまたは移行するための次の方法がサポートされています。

- AutoUpgradeユーティリティ

  アップグレード前の問題を識別し、アップグレードをデプロイし、アップグレード後の処理を実行し、アップグレードしたOracle Databaseを起動します。

- Database Upgrade Assistant(DBUA)

  Graphical User Interfaceによる指示に従ってデータベースをアップグレードできます。DBUAはインストール中にOracle Universal Installerから起動することもできますが、後からいつでもスタンドアロン・ツールとしてDBUAを起動することができます。

- パラレル・アップグレード・ユーティリティや他のコマンドライン・ユーティリティを使用した手動アップグレード

  シェル・スクリプトを使用してアップグレードを実行できます。

- Oracle Data Pumpを使用したデータの移行

  エクスポートおよびインポート・ユーティリティが提供されます。Oracle Data Pumpでは、データベースの全体または一部をエクスポートし、それを新しいリリースのOracle Databaseにインポートできます。Oracle Data Pumpのエクスポート/インポートは、データベースを変更することなく、データベース内のデータのサブセットをコピーできます。

- CREATE TABLE AS SQL文

  データベースから新しいOracle Databaseリリースにデータを移行します。この方法を使用すると、データベースを変更せずにデータのサブセットをコピーできます。

- 優先度に従ってPDBをグループ化およびアップグレードするための優先度リストを使用したCDBおよびPDBのアップグレード。

  優先度リストを使用してアップグレードを実行し、アップグレード実行時にそのリストを呼び出すには、`-L`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(`dbupgrade`または`catctl.pl`)を実行します。

- スタンバイ・データベースの同期化、アップグレード、およびOracle GoldenGateを使用したアップグレード後のデータベースの同期化(停止時間ゼロ・オプション)

- フリート・パッチ適用およびプロビジョニング(FPP)を使用してデータベースをアップグレードします。

  フリート・パッチ適用およびプロビジョニング(FPP)アップグレード(以前の高速ホーム・プロビジョニング)では、新しいOracle Databaseインストールを完了します。データベースで使用する標準運用環境(SOE)に従ってデータベースをテスト、修正した後に、FPPゴールド・イメージを作成します。DBAはそのゴールド・イメージのインスタンスを、アップグレード対象の以前のリリースのデータベースがあるサーバーに配置します。これらのゴールド・イメージを配置した後に、DBAは単一の`rhpctl`コマンドを実行して、ファイルの移動、構成変更の実行、そして新しいバイナリを使用するのに必要なその他のステップを実行します。高速ホーム・プロビジョニングの詳細は、『*Oracle Clusterware*管理およびデプロイメント・ガイド』を参照してください。

関連項目

- Oracle Clusterware管理およびデプロイメント・ガイド

親トピック: Oracle Databaseのアップグレード・ツールおよびアップグレード処理の概要

直接のアップグレードをサポートするOracle Databaseのリリース

最新のOracle Databaseリリースに直接アップグレードするためにサポートされるオプションを参照してください。

次の各リリースから新しいリリースへは、直接、アップグレードできます。

- 11.2.0.4
- 12.1.0.2
- 12.2.0.1
- 18

最新のOracle Databaseリリースにアップグレードするために必要なパスは、現行のデータベースのリリース番号によって異なります。

現行のOracle Databaseが11.2.0.4より前のリリースの場合は、Oracle Databaseの現行のリリースから最新リリースに直接アップグレードすることはできません。この場合、Oracle Database 19cにアップグレードする前に、中間リリースにアップグレードする必要があります。

直接のアップグレードを実行できない場合、直接のアップグレードがサポートされる最新のリリースへのアップグレードを実行します。

ノート:

複数のステップで行われるアップグレードでは、現行のリリースにアップグレードするために 2 回のアップグレードを実施する必要がある場合、最初にターゲットのアップグレード・リリースへの直接アップグレードがサポートされている中間アップグレード・リリースへのアップグレードを完了します。次に、ターゲットのアップグレード・リリースのアップグレードを完了します。

たとえば、アップグレード元になるデータベースで Oracle Database 10g を実行している場合に Oracle Database 19c にアップグレードするには、次のステップに従います。

1. 『*Oracle Database* アップグレード・ガイド *12c* リリース *1 (12.1)*』の指示(12.1.0.2 向けのアップグレード前スクリプトの実行を含む)に従って、リリース 10.2.0.5 からリリース 12.1.0.2 (ターミナル・パッチセットのリリース)へアップグレードします。

2. Oracle Database 12c リリース 1 (12.1.0.2)を直接 Oracle Database 19c にアップグレードします。Oracle Database 19c 向けのアップグレード前スクリプトの実行を含め、このマニュアル『*Oracle Database* アップグレード・ガイド』の指示に従ってください。

次の表に、Oracle Databaseのリリースごとに必要なアップグレード・パスを示します。Oracle Database 18cに完全にアップグレードする前に、ご使用のデータベースのアップグレード・パスおよび指定されたドキュメントを使用して中間アップグレードを実行します。

表1-1 Oracle Database 19cのアップグレード・パスの例

現行リリース	アップグレード・オプション
18 (すべてのリリース)、12.2.0.1、12.1.0.2、11.2.0.4	直接のアップグレードがサポートされています。現行の『Oracle Database アップグレード・ガイド』(このマニュアル)を使用してアップグレードを実行します。
12.1.0.1 11.2.0.1, 11.2.0.2, 11.2.0.3 11.1.0.6, 11.1.0.7 10.2 以前のリリース	Oracle Database 19c への直接のアップグレードはサポートされていません。 解決策: 現行のリリースに直接アップグレードできる Oracle Database の中間リリースにアップグレードします。このリリースでの直接のアップグレードがサポートされない Oracle Database リリースを、直接のアップグレードがサポートされる Oracle Database の中間リリースにアップグレードします。 Oracle Database の中間リリースにアップグレードする際には、中間リリース向けのアップグレード前スクリプトの実行を含め、中間リリースのドキュメントの指示に従ってください。Oracle Database の中間リリースへのアップグレードが完了したら、中間リリースのデータベースを現行の Oracle Database リリースにアップグレードできます。 この制限は、Oracle Data Pump エクスポート/インポートを使用して新しいリリースにデータを移行する場合には適用されません。 たとえば: ● リリース 11.2.0.2 または 11.1.0.7 からアップグレードする場合は、Oracle Database 11g リリース 2 (11.2.0.4)のターミナル・パッチセットのリリースにアップグレードする必要があります。これは、Oracle Database19c への直接アップグレードでサポートされます。 ● リリース 10.2.0.2、10.2.0.3、10.2.0.4、10.2.0.5 または 10.1.0.5 からアップグレードする場合は、先にリリース 11.2 または 12.1 にアップグレードする必要があります ● リリース 9.2.0.8 からアップグレードする場合は、先に Oracle Database の一連の中間リリースにアップグレードする必要があります。 リリース 9.2.0.8 からリリース 11.2.0.4 へのアップグレード。次に、リリース 11.2 から 19c にアップグレードします ノート: 常に、直接アップグレードできる最新の中間リリースに更新してください。実際は、ここに示した例とは異なる場合があります。

親トピック: Oracle Databaseのアップグレード・ツールおよびアップグレード処理の概要

Oracle Databaseのアップグレードに関する最新情報の入手先

このドキュメントに加えて、アップグレードに関する情報が、サポート・サイトおよびpreupgradeパラメータを使用したAutoUpgradeユーティリティを介して提供されます。

My Oracle Supportは、サポートWebサイトを介して、アップグレード前の要件、アップグレード・プロセス、アップグレード後のタスク、互換性および相互運用性に関する最新の更新、ディスカッションおよびベスト・プラクティスを提供します。

アップグレードを開始する前に、My Oracle Supportで入手可能なAutoUpgradeユーティリティの最新バージョンをダウンロードすることをお薦めします。最新バージョンには、アップグレードのためにシステムを準備し、アップグレードを正常に完了するために提供される最新のチェックおよびテストが含まれています。アップグレードは、AutoUpgradeを使用して直接実行できます。AutoUpgradeを使用してアップグレードを実行しない場合、潜在的なアップグレードの問題を識別するには、analyzeモードでpreupgradeパラメータを使用してAutoUpgradeを実行する必要があります。このAutoUpgradeオプションは、以前のリリースで使用されていた可能性があるアップグレード前情報ツールに置き換わります。

Mike Dietrichのブログ・サイト「Upgrade Your Database Now」、Daniel Overby Hansenのサイト「Databases Are Fun」では、Oracleのアップグレードおよび移行製品のマネージャから最新のインサイトが得られます。その意見と推奨事項は、Oracleの顧客とのエクスペリエンスと対話に基づいており、時間と労力を節約できます。これらを参照することをお薦めします。

この文書では、入手可能な最新情報を提供しようと努めています。改善方法のご提案、またはよい記述や不適切な記述の箇所に関するご提案があれば、各HTMLページの下部にある「このページは役に立ちましたか?」リンクにご自由にコメントをお寄せください。オラクル社のミッションは、お客様の成功を支援することです。この文書の作成者は、お客様のご意見を重視しています。お客様の作業を容易にし、アップグレードが成功するために何かできることがあれば、ぜひお知らせください。

関連項目

- My Oracle Supportノート2485457.1
- 今すぐデータベースをアップグレード
- データベースの活用

親トピック: Oracle Databaseのアップグレード・ツールおよびアップグレード処理の概要

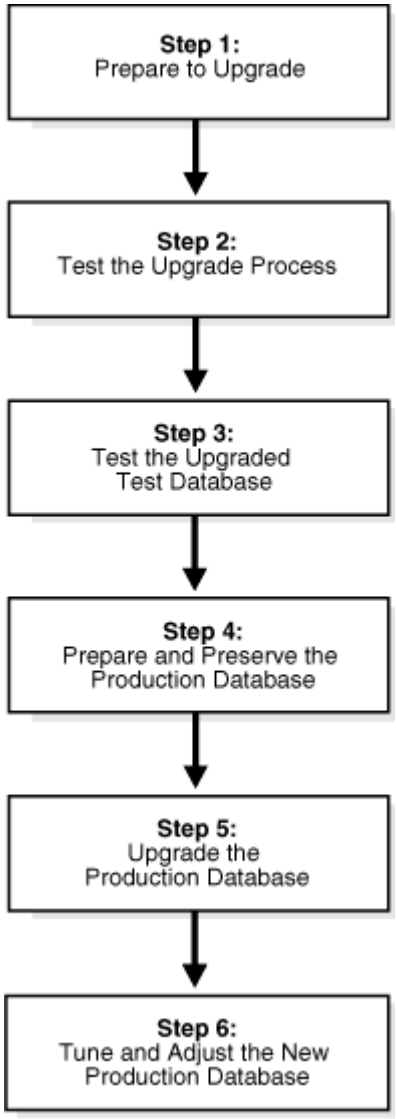
Oracle Databaseのアップグレード処理の主なステップ

Oracle Databaseのアップグレードは、6つの主なステップで構成されます。

アップグレード・ステップのワークフロー

次の図に、アップグレード処理で実行する主な手順の概要を示します。

図1-1 Oracle Databaseのアップグレード・ステップのワークフロー



ステップ1: Oracle Databaseをアップグレードするための準備

- 新しいOracle Databaseのリリースの機能を理解します。
- 新しいリリースへのアップグレード・パスを決定します。
- アップグレード方法を選択します。
- 新しいリリースのOracleホーム・ディレクトリを選択します。
- テスト計画を作成します。
- バックアップ計画を準備します。
- アップグレード前の推奨事項に従います。

- アップグレード前修正スクリプトを実行するか、手動でアップグレード前のシステム更新を実行します。

ノート:

アップグレード中、新しいリリースをテストしている間、既存のリリースを本番環境として使用するために複数のリリースのデータベース・ソフトウェアを実行することを検討してください。

新しい Oracle Database リリースのソフトウェアのみのインストールを完了することを検討してください。ソフトウェアのみのインストールでは、Oracle Database ソフトウェアをインストールしますが、インストール・プロセスの一環としてデータベースは作成しません。

ステップ2: Oracle Databaseのアップグレード処理のテスト

- テスト・データベースを使用して、アップグレードのテストを実行します。本番データベースに影響を与えないテスト用に作成された環境でテスト・アップグレードを実行します。テスト環境が存在するサーバーは、本番環境と可能なかぎり同じ状態にすることをお薦めします。たとえば、サーバーで同じオペレーティング・システムを使用するのみでなく、同じパッケージを使用した同じパッチ・レベルを実行し、本番システム構成の他の詳細も一致させることをお薦めします。

ステップ3: アップグレードしたテスト用Oracle Databaseのテスト

- Oracle Databaseの新しいリリースにアップグレードしたテスト・データベースに対して、ステップ1で計画したテストを実行します。
- 結果を確認し、テストでの相違点を記録します。
- 発見した相違点を解決する方法を調べて、その解決方法を実装します。
- アップグレードのテストが問題なく完了し、必要なアプリケーションで正常に動作するまで、必要に応じてステップ1、ステップ2およびステップ3の最初の部分を繰り返します。
- 相違点をテストし、潜在的なサポートの質問を特定するため、SQL計画管理を実行します。SQL計画管理には、次のステップが含まれます。
  1. アップグレードの前に、以前のリリースのOracle Databaseのベースラインと計画を取得し、それらの計画を格納します。

     計画をステージング表に格納してから、それらの表に対してデータ・ポンプ・エクスポート・ユーティリティexpdpを実行することをお薦めします。
  2. アップグレードの後に、パフォーマンスの低下などの問題が発生した場合、以前のリリースのOracle Databaseから取得した計画に基づいて適切であることがわかっている古い計画を適用(ロード/承認/展開)します。

参照:

- SQL計画管理の詳細は、『Oracle Database SQLチューニング・ガイド』を参照してください
- ドキュメント1948958.1 SQL計画管理(SPM)による問題を避けるために11.2.0.3で検討するパッチ
- ドキュメント2034706.1 SQL計画管理(SPM)による問題を避けるために11.2.0.4で検討するパッチ
- ドキュメント2035897.1 SQL計画管理(SPM)による問題を避けるために12.1.0.1からのアップグレード時に検討す

るパッチ

ステップ4: 本番Oracle Databaseの準備および保存

既存の本番データベースをアップグレードする前に、次の作業を完了します。

- Oracle Databaseの新しいリリースへのアップグレードが正しく行われるように、現行の本番データベースを準備します。
- 本番データベースのバックアップ処理とアップグレード処理に必要な停止時間をスケジューリングします。
- 現行の本番データベースのバックアップを作成します。

  システムに大規模な変更を行う前に、必ずフォールバック計画を実装しておくことをお薦めします。フォールバック計画には、次の準備を含めることをお薦めします。

  - バックアップ計画をテストし、これが機能することを確認します。
  - バックアップ計画が必要な場合は、メンテナンス・ウィンドウ中に適用するために必要な時間を計画します。
  - アップグレードの準備でプラン・スタビリティ・チェックを実行するには、SQL計画管理を実行します。支援が必要な場合、サービス・リクエストを送信します。

ノート:

新しいバージョンのオプティマイザをインストールするデータベースのアップグレードでは、通常、ほんのわずかな SQL 文に対して計画の変更が発生します。

ほとんどの計画変更において、パフォーマンスの向上が見られるか、または変化がありません。ただし、一部の計画の変更は、パフォーマンスの低下の原因となることがあります。SQL 計画ベースラインにより、アップグレードがもたらす低下の可能性を最小限に抑えられます。

アップグレード時には、データベースにより計画ベースラインからの計画のみが使用されます。データベースは現在のベースラインに含まれない新しい計画を保持領域に入れ、後でそれらを評価し、ベースラインに含まれる現在の計画よりも少ないリソースを使用するかどうかを判断します。それらの計画のパフォーマンスがより優れている場合は、それらの計画をベースラインに昇格させます。そうでない場合は昇格させません。

参照:

Oracle Database SQLチューニング・ガイド

ステップ5: 本番Oracle Databaseのアップグレード

- Oracle Databaseの新しいリリースに本番データベースをアップグレードします。
- アップグレードの後、本番データベースの全体バックアップおよびその他のアップグレード後の処理を行います。

ステップ6: 新しい本番Oracle Databaseのチューニングおよび調整

- 新しいリリース用に新しい本番データベースをチューニングします。通常、新しい本番Oracle Databaseのパフォーマンスは、アップグレード前のデータベースと同等以上になります。
- 新しいOracle Databaseリリースのどの機能を使用するかを決定し、それに応じてアプリケーションを更新します。

- 必要に応じて、新しいデータベース管理の手順を作成します。
- アップグレードしたデータベースで使用する必要があるすべてのアプリケーションがテストされ、適切に動作するまでは、本番のOracle Databaseリリースを新しいリリースにアップグレードしないでください。

関連項目

- https://support.oracle.com/epmos/faces/DocumentDisplay?cmd=show&type=NOT&id=1948958.1
- https://support.oracle.com/epmos/faces/DocumentDisplay?cmd=show&type=NOT&id=2034706.1
- https://support.oracle.com/epmos/faces/DocumentDisplay?cmd=show&type=NOT&id=2035897.1

親トピック: Oracle Databaseのアップグレードの概要

Oracle Databaseリリース間の互換性および相互運用性

Oracle Databaseのリリースの違いによって発生する可能性のある互換性と相互運用性の問題を理解および回避する方法について学習します。

Oracle Databaseのリリースには相違点があり、これによって互換性と相互運用性の問題が生じる可能性があります。これらの違いは、一般的なデータベース管理と既存のアプリケーションの両方に影響する可能性があります。

- Oracle Databaseのリリース番号について
  Oracle Databaseリリースは、リリース情報を示す5つの数値セグメントによって分類されます。
- 『Oracle Databaseアップグレード・ガイド』のリリース番号を参照する際の規則に関する説明
  参照して文がどのように各リリースに適用されるかについて理解してください。
- Oracle Databaseの互換性の概要
  アップグレードする前に、アップグレード計画の一環として以前のリリースのOracle Databaseと新しいOracle Databaseリリースの互換性を確認してください。
- Oracle Databaseのアップグレードの相互運用性の概要
  Oracle Databaseにおける相互運用性とは、異なるリリースのOracle Database間での通信および連動が分散環境において可能である状態を指します。
- 無効なスキーマ・オブジェクトおよびデータベース・アップグレードについて
  `utlrp.sql`を実行して、アップグレード・テスト計画の一部として無効なオブジェクトを検証します。
- Oracle OLAPデータ・セキュリティ・ポリシーのアップグレードについて
  セキュリティ・ポリシーは、Oracle Database 12c以降では異なる方法で格納されます。既存のOracle Databaseが11gリリース11.2の場合、セキュリティ・ロールを削除します。

親トピック: Oracle Databaseのアップグレードの概要

Oracle Databaseのリリース番号について

Oracle Databaseリリースは、リリース情報を示す5つの数値セグメントによって分類されます。

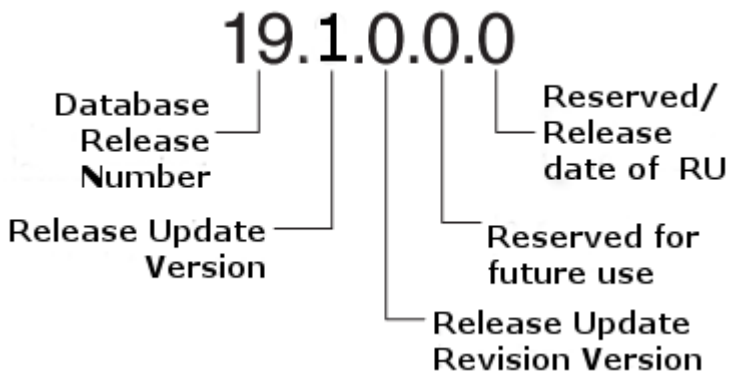
> ノート:
>
> 四半期ごとの更新は、リリース更新(Update または RU)およびリリース更新リビジョン(Revision または RUR)の形式で提供されています。パッチ・セットまたはバンドル・パッチ・セットはリリースされなくなりました。詳細は、My Oracle Support のノート 2285040.1 を参照してください。

Oracle Databaseリリースは、`version`リリースおよび`version_full`リリースとしてリリースされます。

`version`リリースは、メジャー・リリース・バージョン`.0.0.0.0`の形式で指定されます。メジャー・リリース・バージョンは、Oracle Databaseのバージョンがリリースされた最初の年の最後の2桁に基づいています。たとえば、2018年に最初にリリースされたOracle Databaseバージョンのメジャー・リリース・バージョンは18で、その`version`リリースは`18.0.0.0.0`になります。

`version_full`リリースは`version`リリースの更新で、メジャー・リリース・バージョン、四半期ごとのリリース更新バージョン(Update)、および四半期ごとのリリース更新リビジョン(Revision)に基づいて指定されます。`version_full`リリースは、次の例に示すように、ピリオドで区切られた5つの数値セグメントで分類されます。

図1-2 Oracle Databaseのリリース番号の例



- 1番目の数字: この数字はメジャー・リリース・バージョンを示します。これは、Oracle Databaseのバージョンがリリースされた最初の年の最後の2桁でもあります。
- 2番目の数字: この数字はリリース更新バージョン(UpdateまたはRU)を示します。
- 3番目の数字: この数字はリリース更新リビジョン・バージョン(RevisionまたはRUR)を示します。
- 4番目の数字: この数字は将来使用するために予約されています。現在は常に0に設定されています。
- 5番目の数字: 最初の3つのフィールドのみが一般的に使用されますが、5番目のフィールドには19.7.0.0.200414など、リリース更新(RU)のリリース日を冗長に明確にする数値を表示できます。

> ノート:
>
> 最初の 3 つの数字は、主に Oracle Database のリリースを識別します。

注意:

アップグレードを開始する前およびダウングレードを開始する前に、ターゲット・データベースに最新のリリース更新を適用することをお薦めします。可能な場合は、ソース環境にパッチが適用されていることも確認します。リリースの更新は累積されます。Oracle Grid Infrastructure 環境も更新する場合は、Oracle Database Oracle ホームを更新する前に、必ず最新のリリース更新を Grid 環境に適用します。更新の詳細は、My Oracle Support ノート 2118136.2 を参照してください。

関連項目

- My Oracle Supportノート2285040.1
- My Oracle Supportノート2118136.2

親トピック: Oracle Databaseリリース間の互換性および相互運用性

『Oracle Databaseアップグレード・ガイド』のリリース番号を参照する際の規則に関する説明

参照して文がどのように各リリースに適用されるかについて理解してください。

『Oracle Databaseアップグレード・ガイド』にデータベースのメジャー・リリース番号に関する説明が記載されている場合、その説明は、そのデータベースのメジャー・リリースに含まれるすべてのリリースに適用されます。

同様に、『Oracle Databaseアップグレード・ガイド』にメンテナンス・リリースに関する説明が記載されている場合、その説明は、そのメンテナンス・リリースに含まれるすべてのコンポーネント固有のリリースおよびプラットフォーム固有のリリースに適用されます。Oracle Database 12cに関する説明は、このリリースに含まれるすべてのコンポーネント固有のリリースおよびプラットフォーム固有のリリースに適用されます。Oracle Database 11gリリース2 (11.2)に関する説明は、リリース11.2.0.2、リリース11.2.0.3およびOracle Database 11gリリース2 (11.2)に含まれるその他すべてのプラットフォーム固有のリリースに適用されます。

親トピック: Oracle Databaseリリース間の互換性および相互運用性

Oracle Databaseの互換性の概要

アップグレードする前に、アップグレード計画の一環として以前のリリースのOracle Databaseと新しいOracle Databaseリリースの互換性を確認してください。

- Oracle Databaseの互換性の理解
  新しい機能に以前のリリースとの互換性がない場合、データベースの互換性が問題の原因であることがあります。
- Oracle DatabaseでのCOMPATIBLE初期化パラメータ設定のタイミング
  COMPATIBLEパラメータの増加は、アップグレードしたデータベースのテストが完了した後にのみ行うことをお薦めします。
- Oracle DatabaseのCOMPATIBLE初期化パラメータ
  参照して非CDBおよびマルチテナント・アーキテクチャ・コンテナのCOMPATIBLE初期化パラメータの設定方法を理解してください。
- Oracle DatabaseのCOMPATIBLE初期化パラメータの値
  Oracle Database 19cの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確認してください。
- Oracle Databaseのアップグレードでのダウングレードおよび互換性について
  Oracle Database 19cにアップグレードする前に、COMPATIBLE初期化パラメータを11.2.0以上に設定する必要があります。
- Oracle DatabaseでのCOMPATIBLE初期化パラメータの動作
  COMPATIBLE初期化パラメータによって、リリースの互換性に基づいてOracle Database機能を有効化または無効化できます
- Oracle Databaseの互換性レベルの確認
  このSQL問合せを使用して、データベースに設定されているCOMPATIBLE初期化パラメータの値を確認します。

親トピック: Oracle Databaseリリース間の互換性および相互運用性

Oracle Databaseの互換性の理解

新しい機能に以前のリリースとの互換性がない場合、データベースの互換性が問題の原因であることがあります。

異なるリリースのOracle Databaseソフトウェアで同じ機能がサポートされており、その機能が同様に動作する場合は、リリースが異なってもデータベースには互換性があります。新しいリリースのOracle Databaseへアップグレードする場合、一部の新機能が原因で以前のリリースとの互換性が失われることがあります。

アップグレードしたデータベースで以前のリリースとの互換性が失われるのは、次の場合です。

- 新機能によってディスクに格納されるデータ(データ・ディクショナリの変更を含む)が、以前のリリースでは処理できない場合。
- 既存の機能の動作が、以前の環境と新しい環境とで異なる場合。

親トピック: Oracle Databaseの互換性の概要

Oracle DatabaseでのCOMPATIBLE初期化パラメータ設定のタイミング

COMPATIBLEパラメータの増加は、アップグレードしたデータベースのテストが完了した後にのみ行うことをお薦めします。

アップグレードが完了したら、COMPATIBLE初期化パラメータの設定をOracle Databaseの新リリース用に最大レベルに増やすことができます。ただし、COMPATIBLEパラメータを増やすと、後でデータベースをダウングレードできません。

Oracle DatabaseのCOMPATIBLE初期化パラメータ

非CDBおよびマルチテナント・アーキテクチャ・コンテナのCOMPATIBLE初期化パラメータの設定方法を確認してください。

Oracle Databaseでは、COMPATIBLE初期化パラメータを使用して、データベースの互換性を制御できます。

COMPATIBLE初期化パラメータの理解

Oracle Database 19cでは、COMPATIBLE初期化パラメータがパラメータ・ファイルで設定されていない場合、`COMPATIBLE`パラメータ値はデフォルトの`19.0.0`になります。COMPATIBLE初期化パラメータを`19.0.0`に設定しない場合、アップグレードしたデータベースがOracle Database 19c機能にとって必要なCOMPATIBILITY設定で実行されていないため、Oracle Database 19cの新機能を使用することはできません。

内部SQL識別子およびスキーマ・オブジェクト名の制限における変更により、`COMPATIBLE`パラメータ設定を増やすと、データベースがデータベース・サーバーで直接または間接的にJavaストアド・プロシージャを実行するプロセスに対して初期遅延が発生する可能性があります。Oracle Databaseの`COMPATIBLE`パラメータを12.2.0より前の`COMPATIBLE`設定から12.2.0以降に増やし、データベースがJavaストアド・プロシージャを実行すると、データベースは"名前変換"操作を実行します。この操作は、完了するまでに数分間かかることがあります。互換性パラメータを増やした後に、データベースが初めてJavaストアド・プロシージャを実行するときは、この遅延を予期しておく必要があります。データベースによる名前変換の実行において、この遅延は最初のJavaプロセス・コール時にのみ発生します。

ノート:

- Oracle Database 19c にアップグレードする前に、`COMPATIBLE` 初期化パラメータを Oracle Database 19c の最小設定である `11.2.0` 以上に設定する必要があります。
- COMPATIBLE パラメータは、ピリオドで区切られた 3 つ以上の 10 進数である必要があります。たとえば:

```
SQL> ALTER SYSTEM SET COMPATIBLE = '19.0.0' SCOPE=SPFILE;
```

- `COMPATIBLE` パラメータの増加は、アップグレードしたデータベースを完全にテストした後にのみ行うことをお薦めします。
- `COMPATIBLE` パラメータを増やした後は、データベースをダウングレードすることも、リストア・ポイントにフラッシュバックすることもできません。

注意:

Oracle Database リリース 11.2 からアップグレードする場合、COMPATIBLE の値を 11.2.0 以上に設定する必要があります。これは、アップグレード時に行う必要があります。COMPATIBLE 初期化パラメータ値を増加した後に以前の互換性レベルにダウングレードすることはできないため、アップグレードの準備が完了するまで、この変更は行わないでください。

参照:

初期化パラメータの管理の詳細は、『Oracle Database管理者ガイド』を参照してください

マルチテナント・アーキテクチャでのCOMPATIBLEパラメータ設定のルール

コンテナ・データベース(CDB)のCOMPATIBLEパラメータは、そのコンテナ・データベースに接続されているプラガブル・データベース(PDB)のCOMPATIBLEパラメータ設定に影響します。PDBをCDBに接続したときに発生する次のシナリオを確認してください。

- PDBのCOMPATIBLEがCDB$ROOT COMPATIBLEパラメータの設定と同じ。

  結果: PDB COMPATIBLEパラメータ設定への変更はありません。

- PDB COMPATIBLEがCDB$ROOT COMPATIBLEパラメータ設定より低い。

  結果: PDB COMPATIBLEパラメータはCDB$ROOTと同じCOMPATIBLEパラメータ設定に、自動的に増やされます。PDBに接続した後では、以前のリリースにPDBをダウングレードできません。

- PDB COMPATIBLEがCDB$ROOT COMPATIBLEパラメータ設定より高い。

  結果: PDBを接続できません。CDB$ROOTと同等以下のCOMPATIBLEパラメータを持つPDBだけが、CDBに接続できます。

親トピック: Oracle Databaseの互換性の概要

Oracle DatabaseのCOMPATIBLE初期化パラメータの値

Oracle Database 19cの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確認してください。

COMPATIBLEパラメータのデフォルト値と最小値

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。次の表に、直接アップグレードがサポートされている以前のリリースと、Oracle Database 19cのCOMPATIBLEパラメータのデフォルト値と最小値を示します。

表1-2 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
Oracle Database 19c	19.0.0	11.2.0
Oracle Database 12c リリース 2 (12.2)	12.2.0	11.2.0
Oracle Database 12c リリース 1(12.1)	12.0.0	11.0.0
Oracle Database 11g リリース 2(11.2)	11.2.0	10.0.0

親トピック: Oracle Databaseの互換性の概要

Oracle Databaseのアップグレードでのダウングレードおよび互換性について

Oracle Database 19cにアップグレードする前に、COMPATIBLE初期化パラメータを11.2.0以上に設定する必要があり

ます。

Oracle Database 18cにアップグレード後、COMPATIBLE初期化パラメータを新しいリリースのリリース番号と一致するように設定できます。このように設定すると、新しいリリースのすべての機能を使用できますが、以前のリリースにダウングレードすることはできません。COMPATIBLE初期化パラメータがより小さい値に設定されている場合、Oracle Database 18cの機能のサブセットのみを使用できます。

注意:

COMPATIBLE パラメータを増やすと、データベースをダウングレードできません。

関連項目

- 以前のリリースへのOracle Databaseのダウングレード

親トピック: Oracle Databaseの互換性の概要

Oracle DatabaseでのCOMPATIBLE初期化パラメータの動作

COMPATIBLE初期化パラメータによって、リリースの互換性に基づいてOracle Database機能を有効化または無効化できます

COMPATIBLE初期化パラメータは、次のように動作します。

- COMPATIBLE初期化パラメータによって、機能の使用を有効化または無効化することで、既存のアプリケーションによるデータの使用を保護するために役立ちます。

  COMPATIBLE初期化パラメータを11.2.0に設定してOracle Database 12cデータベースを実行すると、データベース・ソフトウェアによって、Oracle Databaseリリース11gリリース2 (11.2)と互換性のあるデータベース構造がディスクに生成されます。Oracle Databaseのより新しいリリースに含まれる機能の使用を試み、データベースがCOMPATIBLE初期化パラメータとの互換性を失うと、エラーが発生します。ただし、Oracle Databaseリリース11gリリース2と互換性のない変更がディスクに作成されない新機能は、有効化されます。

- 使用するCOMPATIBLE初期化パラメータ設定との互換性をデータベースが失う変更がデータベースに行われると、データベースは起動せず、初期化はエラーで終了します。この場合、COMPATIBLE初期化パラメータを、データベースに対する適切な値に設定する必要があります。

参照:

データベース構造の詳細は、『Oracle Database概要』を参照してください。

親トピック: Oracle Databaseの互換性の概要

Oracle Databaseの互換性レベルの確認

このSQL問合せを使用して、データベースに設定されているCOMPATIBLE初期化パラメータの値を確認します。

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

親トピック: Oracle Databaseの互換性の概要

Oracle Databaseのアップグレードの相互運用性の概要

Oracle Databaseにおける相互運用性とは、異なるリリースのOracle Database間での通信および連動が分散環境において可能である状態を指します。

分散データベース・システムには異なるリリースのOracle Databaseを配置することができ、サポート対象すべてのリリースのOracle Databaseを分散データベース・システムに参加させることができます。ただし、分散データベースと連動するアプリケーションでは、システム内の各ノードで使用できる機能やファンクションとも相互運用できる必要があります。

新しいOracle Databaseリリースの最小要件によって一部またはすべてのホストのオペレーティング・システムをアップグレードする必要がある場合があるため、特にローリング・アップグレード時には、異なるオペレーティング・システムおよびオペレーティング・システム・バージョン間での相互運用性が問題の原因となる可能性があります。そのため、Oracle Databaseのアップグレードを開始する前に、ローリング・アップグレード中のシステムのすべての中間的なアップグレード状態において、ドライバ、ネットワークおよびストレージに互換性があることを確認する必要があります。

ノート:

『*Oracle Database* アップグレード・ガイド』では、異なるリリースの Oracle Database 間で実行されるアップグレードおよびダウングレードについて説明しているため、相互運用性の定義は Oracle Database リリース用です。他の Oracle のマニュアルでは、さらに幅広い定義で相互運用性という用語を使用している場合があります。たとえば、異なるハードウェア・プラットフォームおよびオペレーティング・システム間の通信を相互運用性という場合があります。

詳細は、My Oracle Support ノート ID 207303.1「Client / Server / Interoperability Support Between Different Oracle Versions」を参照してください。

関連項目

- https://support.oracle.com/rs?type=doc&id=207303.1

親トピック: Oracle Databaseリリース間の互換性および相互運用性

無効なスキーマ・オブジェクトおよびデータベース・アップグレードについて

`utlrp.sql`を実行して、アップグレード・テスト計画の一部として無効なオブジェクトを検証します。

データベースのアップグレード後、リリース変更によって、アップグレードしたデータベースで無効なスキーマ・オブジェクトが発生することがあります。通常、無効なオブジェクトは、アクセス時または実行時に自己修復が行われます。ただし、ユーザーがこれらの無効なオブジェクトにアクセスする前に、無効なオブジェクトと必要な依存性に関する問題を解決できるように、パッチ適用およびアップグレード手順の一環としてデータベースの無効なオブジェクトを再コンパイルすることをお薦めします。

オブジェクトの検証は、Oracle Databaseのデータ定義言語(DDL)の文をチェックする操作です。これらの文は、データベース構造またはスキーマを定義するために使用されます。DDL文の検証は、完了するまでに時間がかかることがあります。次に、オブジェクトの検証時間に影響する可能性のある一般的な要因のリストを示します。

- 無効なオブジェクトの数
- CPUタイプ
- プロセッサ速度
- システム負荷
- 使用可能な物理メモリー

`utlrp.sql`コマンドは、パッケージ、プロシージャおよび型を含め、無効な状態のすべてのオブジェクトを再コンパイルします。これは、`$ORACLE_HOME/rdbms/admin`ディレクトリにあります。`utlrp.sql`スクリプトは、使用可能なCPUの数(パラメータcpu_countで指定)にCPUごとのスレッドの数(パラメータparallel_threads_per_cpuで指定)を乗じた結果に基づいて、シリアル再コンパイルまたはパラレル再コンパイルで自動的に実行されます。Oracle Real Application Clustersシステム(Oracle RAC)では、すべてのOracle RACノード全体でパラレル・スレッドの数が追加されます。

コマンドは、SYSユーザーとして、またはSYSDBAシステム権限が付与された別のユーザー・アカウントとして実行します。

以前のリリースのOracle Databaseで`utlrp.sql`コマンドを実行して、データベース内に存在する無効なオブジェクトを再コンパイルすることをお薦めします。特に、SYSおよびSYSTEMユーザー・スキーマの無効なオブジェクトを必ず更新してください。アップグレードの一部として再コンパイル時間の計画を含むことができるように、アップグレードのテスト中に、アップグレード・テスト計画の一環としてアップグレードしたOracle Databaseで`utlrp.sql` を実行してください再コンパイル時間は、データベースの無効なオブジェクトの数に比例します。アップグレードの結果、無効なオブジェクトが大量に発生した場合、`utlrp.sql`を実行するために相当な時間がかかることがあります。

親トピック: Oracle Databaseリリース間の互換性および相互運用性

Oracle OLAPデータ・セキュリティ・ポリシーのアップグレードについて

セキュリティ・ポリシーは、Oracle Database 12c以上では異なる方法で格納されます。既存のOracle Databaseが11gリリース11.2の場合、セキュリティ・ロールを削除します。

Oracle Database 12cでは、Oracle OLAPは、Oracle Database 11gリリースで使用されていたExtensible Data Security (XDS)のかわりに、Oracle Real Application Security (ORAS)を使用して、OLAPデータ・セキュリティ・ポリシーを格納します。

Oracle Databaseをリリース11gから新しいOracle Databaseリリースにアップグレードする場合、XDSデータ・セキュリティ・ポリシーはすべて自動的にORASに変換されます。

ノート:

リリース 11g の Oracle Database インスタンスで定義されたデータ・セキュリティ・ロールは、自動的に ORAS に変換されません。Oracle Database リリース 11g を最新の Oracle Database リリースにアップグレードする前に、11g データベースで定義されたすべてのデータ・セキュリティ・ロールを削除する必要があります。アップグレード後に、新しいリリースの Oracle Database Analytic Workspace Manager を使用して、データ・セキュリティ・ロールを再び定義できます。

11g のデータ・セキュリティ・ロールを削除せずに、11g データベースをアップグレードすると、新しい Oracle Database リリースでは、データ・セキュリティ・ロールを含むすべてのデータ・セキュリティ・ポリシーが無効になります。

関連項目

- Oracle OLAPユーザーズ・ガイド

親トピック: Oracle Databaseリリース間の互換性および相互運用性

複数のOracleリリースの実行について

Optimal Flexible Architecture (OFA)を使用して複数のリリースを実行します。

Optimal Flexible Architecture (OFA)は、効率的で信頼できるOracle DatabaseおよびOracle Grid Infrastructureのデプロイメントのための構成ガイドラインのセットです。Oracle DatabaseインストールのOFAアーキテクチャ標準に準拠するようにすべてのOracleソフトウェア・インストールをデプロイすることをお薦めします。OFA標準に準拠することで、インストール環境の管理が簡単になり、Oracleサポートからの迅速な支援を容易に受けることができます。

OFAには、次のメリットがあります。

- ディスク上の大量の複雑なソフトウェアおよびデータを構成し、デバイスのボトルネックやパフォーマンスの低下を避けることができる
- ソフトウェアやデータのバックアップ機能など、データ破損の危険性のある日常業務を簡単にする。
- 複数のOracle Databaseの管理を簡略化する
- データ・ディクショナリにおける空き領域の断片化を排除し、その他の断片化を特定し、リソース競合を最小限に抑える
- 効果的なエンタープライズ・データ管理計画をデプロイするためにデータベース管理者を支援する

現在、OFA規格を使用していない場合、OFA規格への切替えには、ディレクトリ構造の修正とデータベース・ファイルの再配置が伴います。

OFAの詳細は、お使いのオペレーティング・システム個別のOracleのドキュメントを参照してください。制御ファイルと一時ファイルの管理の詳細は、『*Oracle Database*管理者ガイド』を参照してください。

- 異なるコンピュータ上の複数のOracleホームに存在するデータベース
  Oracle Databaseクライアントを使用して複数のリリースに接続する場合に参照してください。
- 同じコンピュータ上の複数のOracleホームに存在するデータベース
  Oracle Database 19cを実行している同じコンピュータ上に以前のリリースのOracle Databaseをインストールすると、クライアント接続の問題が発生する可能性があります。
- Optimal Flexible Architecture標準について
  Oracle Optimal Flexible Architecture (OFA)ルールは、異なるユーザーが所有する異なるバージョンの複数のデータベースが共存できるように、データベース・ソフトウェアを編成してデータベースを構成する際に役立ちます。
- 複数のOracleホームのサポートについて
  Oracle Databaseでは、複数のOracleホームがサポートされます。このリリース以前のソフトウェアを、同じシステムの異なるOracleホーム・ディレクトリに複数回インストールできます。

関連項目

- Oracle Database管理者ガイド

親トピック: Oracle Databaseのアップグレードの概要

異なるコンピュータ上の複数のOracleホームに存在するデータベース

Oracle Databaseクライアントを使用して複数のリリースに接続する場合に参照してください。

異なるコンピュータ上のOracleホームにOracle Database 12cおよびOracle Database 11gデータベースをインストールし、いずれかまたはすべてのデータベースに接続するOracle Database 11gクライアントがあるOracle Database 12cをインス

トールできます。ただし、最新リリースを最初にインストールし、その後に以前のリリースを古い順にインストールする必要があります。古い順にインストールすることで、Oracleインベントリが破損しないように、各インストールでOracleインベントリを検出し、確実にそのインストールを登録できます。

親トピック: 複数のOracleリリースの実行について

同じコンピュータ上の複数のOracleホームに存在するデータベース

Oracle Database 19cを実行している同じコンピュータ上に以前のリリースのOracle Databaseをインストールすると、クライアント接続の問題が発生する可能性があります。

リリースが異なるクライアントとサーバーの組合せを実行すると、相互運用性の問題が発生する可能性があります。Oracle Database 19cを実行中の同じコンピュータに以前のリリースのOracle Databaseをインストールし、クライアントから以前のリリースのデータベースに接続できない場合があります。たとえば、同じコンピュータ上の複数の(異なる) OracleホームにOracle Database 11g、Oracle Database 12c、Oracle Database 18cおよびOracle Database 19cデータベースをインストールし、Oracle Database 11c、Oracle Database 12c、Oracle Database 18cおよびOracle Database 19cクライアントから、このコンピュータ上のいずれかまたはすべてのデータベースに接続することはできません。1つのシステムに複数のリリースを組み合せて使用できます。

My Oracle SupportのMy Oracle Supportノート207303.1「異なるOracleバージョン間でのクライアント/サーバー/相互運用性のサポート」から相互運用性の互換性およびサポート対象の構成に関する最新情報を取得することをお薦めします。

関連項目

- 異なるOracleバージョンのクライアント/サーバーの相互運用性サポート・マトリックス(ドキュメントID 207303.1)

親トピック: 複数のOracleリリースの実行について

Optimal Flexible Architecture標準について

Oracle Optimal Flexible Architecture (OFA)ルールは、異なるユーザーが所有する異なるバージョンの複数のデータベースが共存できるように、データベース・ソフトウェアを編成してデータベースを構成する際に役立ちます。

以前のOracle Databaseリリースでは、OFAルールは、断片化を分離して競合を最小限に抑えることで、最適なシステム・パフォーマンスを実現していました。現行のリリースでは、OFAルールは、一貫性のあるデータベースの管理およびサポートを実現し、データベースの拡張または追加(あるいは他のハードウェアの追加)を簡略化します。

デフォルトで、Oracle Universal Installerによって、OFAルールに準拠する権限付きでOracle Databaseコンポーネントがディレクトリの場所に配置されます。OFAのガイドラインに従ってすべてのOracleコンポーネントを構成することをお薦めします。

OFAのデフォルトを受け入れることをお薦めします。データベースが大規模な場合や、複数データベースの使用を計画している場合は特に、OFAルールに従うことは有用です。

> ノート:
>
> OFA により、インシデントを適切に収集するための自動診断リポジトリ(ADR)の診断データが含まれる ORACLE_BASE の識別が支援されます。

親トピック: 複数のOracleリリースの実行について

複数のOracleホームのサポートについて

Oracle Databaseは、複数のOracleホームをサポートします。このリリース以前のソフトウェアを、同じシステムの異なるOracleホーム・ディレクトリに複数回インストールできます。

マウント・ポイント名を慎重に選択することで、Oracleソフトウェアの管理が簡単になります。Optimal Flexible Architecture (OFA)ルールに従って複数のOracleホームを構成することには、次のメリットがあります。

- このリリース以前のソフトウェアを、同じシステムの異なるOracleホーム・ディレクトリに複数回インストールできます。ただし、Oracle Databaseのあるリリースの製品を別のリリースのOracleホーム・ディレクトリにはインストールできません。たとえば、既存のOracle 18cのOracleホーム・ディレクトリにOracle Database 19cのソフトウェアをインストールすることはできません。
- 所有するユーザーおよびバージョンの異なる複数のデータベースの共存。
- 新しいOracle Databaseリリースは、以前のリリースのOracle Databaseとは異なる新しいOracleホームにインストールする必要があります。

  1つのOracleホームに複数のリリースをインストールすることはできません。Optimal Flexible Architecture (OFA)のガイドラインに従って、リリースごとに異なるOracle DatabaseのOracleホームを作成することをお薦めします。
- 本番環境では、Oracle Databaseサーバー・ソフトウェアのリリースは、最初の4桁全体(メジャー、メンテナンスおよびパッチ・リリース番号)がOracle Databaseディクショナリのリリースと同じである必要があります。
- より新しいOracle Databaseリリースは、より古いOracle Databaseリリースにアクセスできます。ただし、このアクセスは、アップグレード専用です。たとえば、Oracle Database 19cは、Oracle Database 18cデータベースがアップグレード・モードで起動された場合、その18cにアクセスできます。
- Oracle Database Clientは、Oracle Databaseと同じリリース・レベルであれば、同じOracle Databaseホームにインストールできます。たとえば、Oracle Database Client 12.2.0.1を既存のOracle Database 12.2.0.1ホームにインストールすることはできますが、Oracle Database Client 12.2.0.1を既存のOracle Database 12.1.0.2ホームにインストールすることはできません。クライアントのインストール前にパッチ・セットを適用した場合は、再度パッチ・セットを適用する必要があります。
- 構造化されたディレクトリおよびファイルの配置、一貫性のあるデータベース・ファイルのネーミングによるデータベース管理の簡略化。
- データベース管理者がOracleホーム・ディレクトリを追加、移動または削除する際にログイン・ホーム・ディレクトリに危険が及ばない。
- ソフトウェアのアップグレードを、本番データベースが配置されているOracleホームから分離したディレクトリのOracleホームでテストできます。

親トピック: 複数のOracleリリースの実行について

アップグレード中のデータベース変換について

Oracle Databaseをアップグレードする際に選択する最適なパスを決定するには、これらのトピックを参照してください。

- アップグレード中のデータベース変換の概要
  アップグレード中に非CDBをマルチテナント・アーキテクチャのOracle Databaseに変換する方法は2つあり、使用できるテクノロジも複数あります。
- スタンバイ・データベースを使用したアップグレードについて
  Active Oracle Data GuardまたはOracle Enterprise Manager Cloud Controlを使用してデータベースのローリング・アップグレードを実行できます。
- Oracle GoldenGateを使用したOracle Databaseのアップグレード・ステップの概要
  Oracle GoldenGateを使用してOracle Databaseをアップグレードする方法を大まかに理解するには、次のステップを参照してください。
- Oracle DatabaseのStandard EditionからEnterprise Editionへの移行
  Oracle Database Standard EditionからOracle Database Enterprise Editionに移行するには、これらのオプションを確認します
- Oracle DatabaseのEnterprise EditionからStandard Editionへの移行
  Enterprise EditionからStandard Editionへの変換では、エクスポート・ユーティリティを使用したデータのエクスポートおよびインポートが必要です。
- Oracle Database Express Edition (Oracle Database XE)からOracle Databaseへの移行
  Oracle Database Express EditionからOracle Database Enterprise Editionにアップグレードした後に、現行のOracle Databaseリリースにアップグレードする必要があります。

親トピック: Oracle Databaseのアップグレードの概要

Oracle GoldenGateを使用したOracle Databaseのアップグレード・ステップの概要

Oracle GoldenGateを使用してOracle Databaseをアップグレードする方法を大まかに理解するには、次のステップを参照してください。

Oracle GoldenGateを使用した新しいOracle Databaseリリースへのアップグレードは、次のステップで構成されています。

1. 以前のリリースのOracle Databaseソフトウェアを実行中のターゲット・データベースを、既存のデータベースのバックアップを使用して設定します。
2. ターゲット・データベースを新しいOracle Databaseリリースにアップグレードします。
3. ターゲット・データベースを本番データベースと同期化します。
4. アクティブ・モードまたはライブ・モードで環境をテストします。
5. アプリケーションをターゲット・データベースに切り替えます。
6. 新しいOracle Databaseリリースで使用する予定の、デフォルトでインストールされ無効になっていた機能を有効にするなど、ターゲット・データベースで新しいリリースの包括的なテストを実行します。
7. 本番データベースを新しいOracle Databaseリリースにアップグレードします。

ノート:

この手順の詳細は、Oracle GoldenGate のドキュメントを参照してください。

関連項目

- Oracle GoldenGateのドキュメント
- データベースのアップグレードのテスト

親トピック: アップグレード中のデータベース変換について

アップグレード中のデータベース変換の概要

アップグレード中に非CDBをマルチテナント・アーキテクチャのOracle Databaseに変換する方法は2つあり、使用できるテクノロジも複数あります。

Oracle Database 19cは、非CDBのOracleデータベースを非CDBアーキテクチャにアップグレードする最終リリースです。非CDBのOracleデータベース・アーキテクチャから、マルチテナント・アーキテクチャを使用してデプロイされたOracleデータベースに移行およびアップグレードするためのアップグレード・オプションを確認してください。

Oracle Databaseの非CDBアーキテクチャからマルチテナント・アーキテクチャへの手動移行およびアップグレードのオプション

マルチテナント・アーキテクチャを使用するOracle Databaseコンテナ・データベース(CDB)およびプラガブル・データベース(PDB)に非CDBを手動で移行およびアップグレードできる方法は2つあります。

- アップグレード前に非CDBをPDBに変換します。

  このオプションでは、非CDB Oracle Databaseリリースを同じリリースのCDBに接続します。(たとえば、非CDBのOracle Databaseリリース18cをOracle Database 18cリリースのCDBに接続します)。PDBへの非CDB Oracle Databaseの変換を実行します。次に、CDB全体とそのPDBをOracle Database 19cにアップグレードします。

- 非CDBを接続してアップグレードし、アップグレード後にPDBへの非CDBの変換を実行します。

  このオプションでは、非CDB Oracle DatabaseリリースをOracle Database 19cのCDBに接続します。接続された非CDB Oracle DatabaseをOracle Database 19cにアップグレードします。次に、PDBへの非CDB Oracle Databaseの変換を実行します。

オプションのアップグレードの技術方法

次の表に、アップグレードの変換に使用できる方法(可用性の問題に対する参照を含む)を示します。また、各アップグレード方法を実行する方法を記述したドキュメントへの参照も提供します。

表1-3 アップグレード中のデータベース移行およびアップグレードの技術方法

方法	説明	参照先
Oracle Database の AutoUpgrade	AutoUpgrade ツールを使用して、データベースのアップグレードを自動化します。	「Oracle Database アップグレードでの AutoUpgrade の使用」のトピックを参照してください
Oracle Data Guard 一時スタンバイ(フィジカル・スタンバイ)・データベース	既存のフィジカル・スタンバイ・データベースを一時的にロジカル・スタンバイ・データベースに変換して、データベースのアップグレードを実行してから、再度フィジカル・スタンバイに変換します。	「スタンバイ・データベースを使用したアップグレードについて」のトピックを参照してください
パラレル・アップグレード・ユーティリティ	パラレル・アップグレード・ユーティリティを使用して、Oracle データベースの手動アップグレードを実行します。Oracle Database が非 CDB の場合は、データベースを PDB に変換する必要があります	「DBUA またはパラレル・アップグレード・ユーティリティを使用した Oracle Database のアップグレード」内のトピックを参照してください
Oracle GoldenGate による	Oracle GoldenGate をソフトウェア・アップグレード	ユースケースについては、Oracle

方法	説明	参照先
停止時間ゼロのアップグレードのための本番データベースおよびスタンバイ・データベースの同期	および Oracle Database のデータ移行の手順と組み合せて使用し、アップグレード中に可用性を維持する同期化アプローチを実行します。 ● RMAN リストアおよびアップグレードを使用し、既存のバックアップを使用して以前のリリースのソフトウェアを実行するスタンバイ・データベースを設定します ● スタンバイ・データベースを新しい Oracle Database リリースにアップグレードします ● データベース全体を移行し、次のツールを使用してスタンバイ・データベースを本番データベースに同期します。 ● Oracle Data Pump ● トランスポータブル表領域(TTS) ● CREATE TABLE AS SELECT (CTIS)による新しい表の作成と指定した問合せからの行の移入。 ● INSERT AS SELECT (IAS)によるパーティション化されていない表の作成 ● データ・ロード/アンロードを使用して、新しいデータベース・リリースにデータをロードし、古いデータベース・リリースからデータをアンロードします	GoldenGate のドキュメントおよび関連する Oracle Database のドキュメントを参照してください。
Oracle Enterprise Manager Cloud Control	Oracle Database 12c 以降のリリースでデータベース・アップグレードを実行するための Cloud Control サポートが提供されています。このオプションでは Enterprise Manager Lifecycle Management Pack の購入が必要です。	Oracle Enterprise Manager Cloud Control のオンライン・ヘルプを参照してください

ノート:

Oracle Grid Infrastructure (Oracle Clusterware および Oracle Automatic Storage

Management)のアップグレードは、Oracle Database のアップグレードの前に個別に実行します。Oracle Database インストールをアップグレードする前に、Oracle Grid Infrastructure のアップグレードを完了する必要があります。Oracle Database とともにインストールされる他の機能には、追加のアップグレード要件が存在することがあります。

関連項目

- ご使用のプラットフォーム用の『Oracle Grid Infrastructureインストレーション・ガイド』
- Oracle GoldenGateのドキュメント

親トピック: アップグレード中のデータベース変換について

スタンバイ・データベースを使用したアップグレードについて

Active Oracle Data GuardまたはOracle Enterprise Manager Cloud Controlを使用して、データベースのローリング・アップグレードを実行できます。

DBMS_ROLLING PL/SQLパッケージにより、Oracle Data Guard構成内のデータベース・ソフトウェアをローリング方式でアップグレードできます。Active Data Guardを使用したローリング・アップグレードでは、Oracle Data Guardのフィジカル・スタンバイ・データベースおよびSQL Applyプロセスを使用します。Data Guardを使用したローリング・アップグレードは、Oracle Database 12cリリース1 (12.1)以上のOracle Databaseリリースでサポートされています。

Oracle Database 12cリリース2 (12.2)以上のリリースでは、DBMS_ROLLING PL/SQLパッケージを使用してローリング・アップグレードを実行する場合、ブローカを無効にする必要がなくなりました。また、ブローカによって、ローリング・アップグレードの適用時期がレポートされ、そのステータスが追跡されるようになりました。ステータス情報は、DGMGRLコマンドのSHOW CONFIGURATIONおよびSHOW DATABASEの出力に表示されます。

Oracle Enterprise Manager Cloud Controlでは、Data Guard構成でデータベースのローリング・アップグレードを実行するための複数のオプションが提供されます。手順は、Cloud Control内のオンライン・ヘルプに記載されています。

参照:

- Oracle Data Guard Broker構成におけるアップグレードおよびダウングレードの詳細は、『Oracle Data Guard Broker』を参照してください
- DBMS_ROLLINGを使用したローリング・アップグレードの実行については、『Oracle Data Guard概要および管理』を参照してください。

親トピック: アップグレード中のデータベース変換について

Oracle DatabaseのStandard EditionからEnterprise Editionへの移行

次のオプションを確認して、Oracle Database Standard EditionからOracle Database Enterprise Editionに移行します。

新しいOracle Databaseリリースよりも古いリリースのOracle Database Standard Editionがある場合は、次のいずれかのオプションを選択して、それをStandard EditionのリリースからOracle Database Enterprise Editionに変更できます。

- 通常のアップグレード手順を実行する。

  新しいOracleホームにOracle Enterprise Editionソフトウェアをインストールし、「Oracle Databaseのアップグレード」の章で説明されているように、通常のアップグレード手順を実行します。Standard Edition構成とEnterprise Edition構成のデータ・ディクショナリは同じです。Standard EditionとEnterprise Editionの違いは、サーバーの実行可能ファイルで使用できるオプションにあります。

- 同じOracleホームを使用して、インプレース・アップグレードを実行する。

  新しいリリースのOracle Databaseよりも古いリリースのStandard Editionがある場合で、同じOracleホームを使用してインプレース・アップグレードを実行する場合は、まず、Standard Edition Databaseをアップグレードする必要があります。アップグレードの完了後に、ここに記載されている手順を使用して、Oracle Database Enterprise Editionソフトウェアをインストールし、Oracle Database Enterprise Editionに移行します。

注意:

この手順を実行すると Oracle Standard Edition ソフトウェアがアンインストールされます。その結果、Oracle ホーム下、および高速リカバリ領域(FRA)下にあるデータベース・ファイルが削除されます。この手順を実行する前に、現在の Oracle ホーム下にあるデータベース・ファイルをバックアップしてください。

1. Oracle Standard Editionサーバー・ソフトウェアとOracle Enterprise Editionサーバー・ソフトウェアのリリース番号が同じかどうかを確認します。
2. データベースを停止します。
3. オペレーティング・システムがWindowsの場合、`OracleServiceSID`(`SID`はインスタンス名)を含むすべてのOracleサービスを停止します。
4. 保持する必要のある現行のOracleホーム下のデータベース・ファイルをすべてバックアップします。
5. Standard Editionサーバー・ソフトウェアを削除します。

注意:

このステップを実行すると、Oracle ホーム下に存在するすべての既存のデータベース・ファイルが削除されます。

Oracleホームから削除ツールを実行します。この削除ツールは、インストール後に別のコマンド(`deinstall`)として

Oracleホーム・ディレクトリから使用できます。これは`ORACLE_HOME/deinstall`にあります。

Microsoft WindowsでOracleホームを削除するには、次の構文を使用します。

```
setup.exe -deinstall -home path_of_Oracle_home_to_be_deinstalled
```

LinuxおよびUnixでOracleホームを削除するには、次の構文を使用します。

```
$ ./runInstaller -deinstall -home path_of_Oracle_home_to_be_deinstalled
```

> ノート:
>
> 削除ツールはデータベースのインストール・メディアに統合されています。削除ツールは、`-deinstall`および`-home`オプションを指定し、Linux および Unix では `runInstaller` を、Windows では `setup.exe` を使用して、Oracle Database、Oracle Database Client または Oracle Grid Infrastructure インストール・メディアのベース・ディレクトリから実行できます。

6. Oracle Enterprise Editionサーバー・ソフトウェアをインストールします。

   削除したStandard Editionで使用されていたのと同じOracleホーム、または新しいOracleホームを選択します。インストール中は、必ずEnterprise Editionを選択します。プロンプトが表示されたら、「**ソフトウェアのみ**」を選択します。

7. 既存のデータベースが存在する場合、`ORACLE_SID`をこの既存のデータベースに設定します。

   既存のデータベースがMicrosoft Windows上にある場合、`ORADIM`ユーティリティを使用してデータベース・サービスを再作成する必要があります。

8. データベースを起動します。

関連項目

- Oracle Databaseのアップグレード

親トピック: アップグレード中のデータベース変換について

Oracle DatabaseのEnterprise EditionからStandard Editionへの移行

Enterprise EditionからStandard Editionへの変換では、エクスポート・ユーティリティを使用したデータのエクスポートおよびインポートが必要です。

Enterprise EditionデータベースからStandard Editionデータベースへの変換を適切に行うには、エクスポート/インポート操作を実行する必要があります。Standard Editionソフトウェアだけをインストールした場合は、一部のデータ・ディクショナリ・オブジェクトが無効になります。データベースを保守するときに、これらの無効なオブジェクトにより問題が発生します。

エクスポート/インポート操作では、SYSスキーマ・オブジェクトはエクスポートされないため、Enterprise Edition固有のデータ・ディクショナリ・オブジェクトは対象となりません。Standard Editionデータベースへのインポート後に必要な作業は、Enterprise Edition機能に関連するユーザー・スキーマの削除のみです。

親トピック: アップグレード中のデータベース変換について

Oracle Database Express Edition (Oracle Database XE)からOracle Databaseへの移行

Oracle Database Express EditionからOracle Database Enterprise Editionにアップグレードした後に、現行のOracle Databaseリリースにアップグレードする必要があります。

Oracle Database Express Edition (Oracle Database XE)は、Oracle Databaseのエントリレベルのエディションです。

Oracle Database 11gリリース2 (11.2) Express Edition (Oracle Database XE)をOracle Database 12cリリース2以降のリリースにアップグレードするには、最初にOracle Database XEからOracle Database 12cリリース1 (12.1.0.2) Enterprise Editionにアップグレードしてから、以降のOracle Database Enterprise Editionリリースにアップグレードする必要があります。

詳細は、「Oracle Database Express Edition (XE)」Oracleオンライン・フォーラムを参照してください。

http://forums.oracle.com

親トピック: アップグレード中のデータベース変換について

新しいOracle Databaseリリース用のプラットフォームのアップグレードについて

新しいOracle Databaseリリース用にオペレーティング・システムまたはハードウェアをアップグレードする場合、これらのトピックを参照してください。

- オペレーティング・システムのアップグレードについて
  新しいリリースのオペレーティング・システム要件を確認し、必要に応じて、Oracle Databaseをアップグレードする前にオペレーティング・システムをアップグレードします。
- 異なるオペレーティング・システムにデータを転送するオプション
  クロス・プラットフォーム・アップグレードを実行する場合、これらの制限およびガイドラインを参照してください。

親トピック: Oracle Databaseのアップグレードの概要

オペレーティング・システムのアップグレードについて

新しいリリースのオペレーティング・システム要件を確認し、必要に応じて、Oracle Databaseをアップグレードする前にオペレーティング・システムをアップグレードします。

新しいリリースのOracleソフトウェアにアップグレードする場合、オペレーティング・システム要件が変更されている可能性があります。必要に応じて、Oracle Databaseをアップグレードする前にオペレーティング・システムをアップグレードします。

参照:

- サポートされているオペレーティング・システムのリストを取得するには、ご使用のプラットフォームの『Oracle Database インストレーション・ガイド』を参照してください
- オペレーティング・システムのアップグレードを実行する方法については、ご使用のオペレーティング・システム固有のマニュアルを参照してください。

親トピック: 新しいOracle Databaseリリース用のプラットフォームのアップグレードについて

異なるオペレーティング・システムにデータを転送するオプション

クロス・プラットフォーム・アップグレードを実行する場合、これらの制限およびガイドラインを参照してください。

DBUAを使用している場合またはOracle Databaseを手動でアップグレードしている場合は、オペレーティング・システム間でデータベースのデータは直接、移行または転送できません。たとえば、DBUAを使用して、Solaris上のOracle Databaseのデータを Windows上のOracle 12cデータベースには移行できません。使用しているオペレーティング・システム・プラットフォーム固有の手順に従う必要があります。

プラットフォーム間でのデータの転送がサポートされているプラットフォームを確認するには、SQL*Plusを使用して次の問合せを実行します。

```
SELECT * FROM V$TRANSPORTABLE_PLATFORM ORDER BY PLATFORM_NAME;
```

ノート:

ソース・プラットフォームとターゲット・プラットフォームでエンディアンが異なる場合、RMAN CONVERT DATABASE は使用できません。この処理では、ソースとターゲット・プラットフォームの両方が同じエンディアン値である必要があります。使用可能なオプションは、データ・ポンプ・レプリケーション、データ・ポンプ・エクスポート/インポートまたはトランスポータブル表領域、と RMAN CONVERT TABLESPACE の使用です。プラットフォームでエンディアンが同じ場合は、変換は必要なく、データは同じプラットフォーム上にある場合と同様に転送できます。

参照:

- プラットフォーム間でのデータ転送の詳細は、『Oracle Database管理者ガイド』 を参照してください
- RMAN CONVERT DATABASEコマンドおよびRMAN CONVERT TABLESPACE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: 新しいOracle Databaseリリース用のプラットフォームのアップグレードについて

イメージベースのOracle Databaseインストールについて

イメージベースのインストールを理解して、Oracle Databaseソフトウェアのインストールおよび構成を簡略化します。

Oracle Databaseをインストールするには、新規Oracleホームを作成し、新しく作成したOracleホームにイメージ・ファイルを抽出し、設定ウィザードを実行して、Oracle Database製品を登録します。

イメージベースのインストールを使用すると、単一インスタンスおよびクラスタ構成用のOracle Databaseをインストールおよびアップグレードできます。

このインストール機能によりインストール・プロセスが効率的になり、大規模なカスタム・デプロイメントの自動化がサポートされます。また、ベースリリース・ソフトウェアに対して必要なリリース更新(Update)またはリリース更新リビジョン(Revision)を適用した後に、このインストール方法を使用してカスタマイズ済イメージをデプロイすることもできます。

ノート:

Oracle Database ホームを配置するディレクトリにイメージ・ソフトウェア(`db_home.zip`)を抽出してから、Oracle Database 設定ウィザードを実行し、Oracle Database のインストールおよび構成を開始する必要があります。作成した Oracle ホームのディレクトリ・パスが Oracle Optimal Flexible Architecture の推奨事項に準拠することをお薦めします。

親トピック: Oracle Databaseのアップグレードの概要

2 Oracle Databaseのアップグレードの準備

アップグレードを正常に終了できるように、アップグレード前タスクとチェックを実行します。

この章では、アップグレードの計画、データ収集、テスト、アップグレード用の新しいOracleソフトウェアのインストール、パラレル・アップグレード・ユーティリティを使用したアップグレードの実行、他のチェックおよび作業の実行など、アップグレード前の作業に関する情報および手順を提供します。

- Oracle Databaseのアップグレードを準備する作業
  アップグレードを準備するには、これらの作業を実行します。
- AutoUpgradeを使用したアップグレード前情報のチェック
  Oracle Databaseをアップグレードする前に完了する必要があるタスクのチェックリストを取得するには、`analyze`モードでAutoUpgradeユーティリティ(`autoupgrade.jar`)を実行します。
- 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ことで、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のインストールが支援されます。
- Oracle RAC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ことで、Oracle RAC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のインストールが支援されます。
- 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、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 Data Guardを使用したデータベースのローリング・アップグレードの準備
  Oracle Data Guardを使用してローリング・アップグレードを実行する場合、アップグレードを開始する前に、Data Guard Broker構成ファイルを移動する必要があります。
- 新しいOracleホームのアップグレードの準備
  新しい場所で新しいOracleホームを準備するには、構成ファイルを移動する必要があるかどうか、またはその他の作業を実行する必要があるかどうかを確認します。
- Windows上のOracleホームを準備するための前提条件
  Microsoft WindowsプラットフォームでOracle Databaseをアップグレードする前に、現在のシステムがこれらの要件を満たしている必要があります。
- AutoUpgradeを使用したアップグレード前チェックの実行
  AutoUpgradeユーティリティは、Oracleによって提供されるJava JARファイルで、アップグレードが正常に完了したことを確認するために役立ちます。
- Oracle Databaseのアップグレード処理のテスト
  Oracle Databaseのアップグレードのテスト計画には、これらのテスト手順が含まれる必要があります。
- Oracle Label SecurityおよびOracle Database Vaultを使用しているデータベースのアップグレードの要件
  Oracle Label SecurityまたはOracle Database Vaultを使用しているデータベースのアップグレードを開始する前に、これらの作業を完了する必要があります。
- アップグレード前のOracle Databaseのバックアップ
  この手順を使用して、アップグレードを試みる前に既存のOracle Databaseをバックアップします。

Oracle Databaseのアップグレードを準備する作業

アップグレードを準備するには、これらの作業を実行します。

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薦めします。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薦めします。

- Oracle Databaseの新機能の理解
  アップグレード処理を計画する前に、新しいOracle Databaseリリースの機能を理解します。
- AutoUpgradeを使用したアップグレード前情報のチェック
  Oracle Databaseをアップグレードする前に完了する必要があるタスクのチェックリストを取得するには、`analyze`モードでAutoUpgradeユーティリティ(`autoupgrade.jar`)を実行します。
- 非推奨となったかサポートが終了した機能の確認
  アップグレードする前に、アップグレード計画で非推奨となった機能またはサポートが終了した機能に注意する必要があるかどうかを確認します。
- Oracle Databaseのアップグレード方法の選択
  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。
- アップグレード時のOracleホームの新しい場所の選択
  データベースをアップグレードまたはパッチ適用したら、新しいOracleホームを新しい場所にインストールする必要があります(アウトオブプレース・アップグレードまたはパッチ適用)。
- Oracle Databaseのアップグレードでのテスト計画の開発
  アップグレード処理のすべてのステージを検証するために慎重に設計された一連のテストを作成する方法を理解するには、これらのトピックを参照してください。
- スキーマ専用アカウントおよびEXPIREDパスワード・アカウントのアップグレード
  アップグレードを開始する前に、パスワードが`EXPIRED`ステータスであり、そのアカウントが`LOCKED`ステータスになっている場合にデフォルトのOracle Databaseアカウントにパスワード認証を使用するかどうかを判断します
- ダウングレードおよびリカバリ・オプションを保持するためのファイルのバックアップ
  必要に応じて、アップグレードの問題からリカバリして以前のリリースにダウングレードできるように、データベースおよび特定のファイルのバックアップ計画の実装をお薦めします。

親トピック: Oracle Databaseのアップグレードの準備

Oracle Databaseの新機能の理解

アップグレード処理を計画する前に、新しいOracle Databaseリリースの機能を理解します。

Oracle Databaseのリリース間の相違点については、『*Oracle Database*新機能ガイド』を参照してください。また、特定のコンポーネントの新機能については、Oracle Databaseのドキュメント・ライブラリにある固有のガイドを参照してください。たとえば、Oracle Real Application Clustersの変更点については、『*Oracle Real Application Clusters*管理およびデプロイメント・ガイド』を参照してください。

ノート:

Oracle Database で使用可能な機能を活用する方法をを学習するには、Oracle Database のトレーニング・クラスを受講することが最適です。詳細は、次を参照してください。

http://education.oracle.com/

関連項目

- Oracle Database新機能ガイド

親トピック: Oracle Databaseのアップグレードを準備する作業

AutoUpgradeでのアップグレード前情報チェック

Oracle Databaseをアップグレードする前に完了する必要があるタスクのチェックリストを取得するには、`analyze`モードでAutoUpgradeユーティリティ(`autoupgrade.jar`)を実行します。

Oracleでは、Oracle Databaseをアップグレードする前に、AutoUpgradeの最新リリースをダウンロードして`-analyze`モードで実行することをお薦めします。AutoUpgradeでは、アップグレードを開始する前に対処する必要がある問題を識別できます。場合によっては、一部の問題を解決できるスクリプトをAutoUpgradeで生成することもできます。

> ヒント:
>
> Mike Dietrich のアップグレードに関するブログで、アップグレードの準備に役立つヒントおよび推奨事項を確認することを検討してください。Oracle Database 19c の場合、My Oracle Support のノート Oracle Database 19c の重要な推奨個別パッチ(ドキュメント ID 555.1)を確認して、Oracle Database 19c の最も重要なパッチ・バンドルに関する情報を取得します。

関連項目

- My Oracle Support AutoUpgradeツール(ドキュメントID 2485457.1)
- My Oracle Support Oracle Database 19cの重要な推奨個別パッチ(ドキュメントID 555.1)
- "Upgrade your Database – NOW!" Mike DietrichのOracle Databaseアップグレード・ブログ

親トピック: Oracle Databaseのアップグレードを準備する作業

非推奨となったかサポートが終了した機能の確認

アップグレードする前に、アップグレード計画で非推奨となった機能またはサポートが終了した機能に注意する必要があるかどうかを確認します。

オラクル社では、すべてのリリースで機能、ビューおよびパラメータのサポートを変更または削除し、データベースの他の機能の中核となる管理容易性および機能性を向上させることに集中できるようにしています。そのため、アップグレード計画の一環として、新しいリリースで非推奨となったまたはサポートが終了したとして記載されている機能のリストを確認し、これらの変更がアプリケーションで重要かどうか判断することをお薦めします。

削除が予定されている機能には、次の2つのカテゴリがあります。

- **非推奨となった**機能とは、拡張はされませんが、Oracle Databaseのこのリリースの有効期間中は引き続きサポートされる機能です。
- **サポート対象外**機能とは、その機能に関連するバグの修正によるサポートを受けられなくなった機能です。多くの場合、機能を使用するために必要なコードを削除することがオラクル社では選択できます。非推奨となった機能は、次のOracle Databaseリリースではサポート終了となる可能性があります。

機能が非推奨であることが判明した場合は、実施可能になり次第、その機能の使用を停止することをお薦めします。非推奨となった機能からの移行の計画は、それらが非推奨となった時点で開始してください。

親トピック: Oracle Databaseのアップグレードを準備する作業

Oracle Databaseのアップグレード方法の選択

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。

- AutoUpgradeユーティリティでOracle Databaseをアップグレードする方法
  AutoUpgradeユーティリティは、アップグレード前に問題を識別し、アップグレード前とアップグレード後のアクションを実行し、アップグレードをデプロイし、アップグレード後のアクションを実行して、アップグレードされたOracle Databaseを起動します。
- 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のアップグレード時のデータの移行でエクスポート/インポートする方法
  Oracle Data Pumpを使用してデータのエクスポートとインポートを実行できます。
- 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。

親トピック: Oracle Databaseのアップグレードを準備する作業

AutoUpgradeユーティリティでOracle Databaseをアップグレードする方法

AutoUpgradeユーティリティは、アップグレード前に問題を識別し、アップグレード前とアップグレード後のアクションを実行し、アップグレードをデプロイし、アップグレード後のアクションを実行して、アップグレードされたOracle Databaseを起動します。

My Oracle Supportドキュメント2485457.1から最新バージョンのAutoUpgradeユーティリティをダウンロードし、`autoupgrade.jar`を使用してアップグレードを準備およびデプロイすることをお薦めします。AutoUpgradeユーティリティは、アップグレードの開始前、アップグレードのデプロイメント時、およびアップグレード後のチェックと構成の移行時、アップグレード・プロセスを自動化するように設計されています。新しいOracle Databaseリリースのバイナリをダウンロードした後にAutoUpgradeを使用し、新しいリリースのOracleホームを設定します。AutoUpgradeを使用するときに、各データベース・デプロイメントで必要に応じてカスタマイズされた単一の構成ファイルを使用して、複数のOracle Databaseデプロイメントを同時にアップグレードできます。Oracle Database 21c以降、既存のターゲット・リリースのCDBがある場合、AutoUpgradeを使用して、アップグレード中に非CDBのOracle Databaseをターゲット・リリースのCDBのPDBに変換できるようになりました。

AutoUpgradeでは、既存のターゲット・リリースのCDBがある場合、AutoUpgradeを使用して、アップグレード中に非CDBのOracle Databaseをターゲット・リリースのCDB上のPDBに変換できます。

AutoUpgradeバージョン22.1以降、Oracleでは、Oracle REST Data Services (ORDS)またはOracle Cloud Infrastructure (OCI) REST APIを使用してSSHを介してリモートでアップグレードを実行できるREST APIを提供しています。ORDSデータベースAPIは、Oracle REST Data Servicesに埋め込まれているデータベース管理およびモニタリングREST APIです。OCI REST APIは、OCI署名バージョン1のセキュリティ・ポリシーを使用するようにRESTアダプタ接続を構成することで有効化されます。これらの機能を使用し、HTTPSまたはHTTPを介してAutoUpgradeアップグレードをリモートで実行できます。

ソース・データベースの最小COMPATIBLEパラメータ設定は12.2.0以上である必要があります。COMPATIBLE設定が低いバージョンの場合、変換およびアップグレード・プロセス中にCOMPATIBLEが12.2.0に設定されます。変換中、元のデータファイルは保持されます。これらは、新しいPDBを作成するためにコピーされません。AutoUpgradeでアップグレードを実行できるようにするには、AutoUpgrade構成ファイルを編集して、AutoUpgradeパラメータtarget_versionをターゲットCDBリリースに設定し、target_cdbを使用して、アップグレードしたデータベースが配置されるCDBを識別します。変換およびアップグレード・プロセスでは、AutoUpgradeはその情報を使用してターゲットCDBへのアップグレードを完了します。

注意:

変換およびアップグレードを完了するために AutoUpgrade を実行する前に、ソース・データベースの完全なバックアップを作成し、アップグレードの完全なテストを完了することをお薦めします。AutoUpgrade がこのプロシージャを開始した後、非 CDB の Oracle Database 状態にロールバックするオプションはありません。

親トピック: Oracle Databaseのアップグレード方法の選択

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。

Oracle Databaseのアップグレードには、AutoUpgradeユーティリティを使用することをお薦めします。ただし、Oracle Database 19cでも、引き続きDBUAを使用してマルチテナント・アーキテクチャのコンテナ・データベース(CDB)およびプラガブル・データベース(PDB)をアップグレードできます。

DBUAは、アップグレード前パラメータ(autoupgrade.jar -preupgrade -mode)を使用してAutoUpgrradeを起動します。これにより、一部の構成設定がアップグレードに必要な値に修正されます。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のアップグレード方法の選択

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

- AutoUpgrade (autoupgrade.jar -mode analyze)を使用してデータベースを分析します。

  AutoUpgrade分析(analyze)処理モードでは、データベースをチェックしてアップグレードの準備が完了しているかどうかを確認します。分析モードでAutoUpgradeを実行すると、AutoUpgradeではデータベースからのデータの読取りのみを実行し、データベースに対する更新は実行しません。通常の営業時間に分析モードを使用してAutoUpgradeを実行できます。ターゲット・リリースのOracle Databaseホームを設定する前に、ソースOracle Databaseホームで分析モードでAutoUpgradeを実行できます。

- 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。

親トピック: Oracle Databaseのアップグレード方法の選択

Oracle Databaseのアップグレード時のデータの移行でエクスポート/インポートする方法

Oracle Data Pumpを使用してデータのエクスポートとインポートを実行できます。

トピック:

- アップグレードしたOracle Databaseでのエクスポート/インポートの影響
  エクスポート/インポートによるデータ移行のメリットを理解するには、このトピックを参照してください。
- Oracle Databaseのデータ移行でのエクスポート/インポートの利点
  Oracle Databaseのアップグレード時にエクスポート/インポートを使用してデータを移行すると、パフォーマンスを向上できるメリットがあります。
- エクスポート/インポートを使用したデータ移行の時間要件
  Oracle Data Pumpを使用したデータ移行にかかる時間を理解します。

親トピック: Oracle Databaseのアップグレード方法の選択

アップグレードしたOracle Databaseでのエクスポート/インポートの影響

エクスポート/インポートによるデータ移行のメリットを理解するには、このトピックを参照してください。

エクスポート/インポートによるデータ移行方法では、現行のデータベースが変更されないため、データベースはアップグレード処理を通して常に使用可能な状態です。ただし、一貫性のあるデータベースのスナップショットが必要な場合(データの整合性保持またはその他の目的のため)、データベースは、制限モードで実行するか、またはエクスポート手順の実行中は変更禁止にする必要があります。現行のデータベースを使用可能な状態にしておくことができるため、たとえば、既存の本番データベースを実行しながら、新しくアップグレードしたOracle Databaseのデータベースの作成とエクスポート/インポートを同時に行うことができます。このアップグレード処理中にデータベースの完全な一貫性を維持するには、新しくアップグレードしたOracle Databaseのデータベースにも同じ変更を加えないかぎり、データベースのデータを変更できないようにします。

最も重要なことは、エクスポート/インポート操作の結果として、新しいデータベースが作成されることです。現行のターゲット・データベースには最終的に指定した移行済のデータのコピーが含まれますが、アップグレードしたデータベースは、元のソース・データベースとは異なる方法で運用される可能性があります。エクスポート/インポートにより、データベースと同一のコピーが作成されても、他の要因により、想定外のパフォーマンス問題が発生する可能性があります。(ディスクのデータ配列、調整パラメータの未設定など)。

親トピック: Oracle Databaseのアップグレード時のデータの移行でエクスポート/インポートする方法

Oracle Databaseのデータ移行でのエクスポート/インポートの利点

Oracle Databaseのアップグレード時にエクスポート/インポートを使用してデータを移行すると、パフォーマンスを向上できるメリットがあります。

エクスポート/インポートを使用してデータを移行すると、次のメリットがあります。

- データの断片化を解消できます。インポートされたデータを圧縮することによって、パフォーマンスを向上できます。

- データベースを再構築できます。新しい表領域を作成したり、既存の表、表領域、またはインポートしたデータを移入するパーティションを変更することができます。
- 完全に新しいデータベースが作成されるため、Oracle Databaseの新旧リリースの比較テストを行うことができます。
- 指定されたデータベース・オブジェクトまたはユーザーをコピーできます。オブジェクト、ユーザーおよびその他の必要な項目のみをインポートすると、本番データのサブセットにのみ新規ソフトウェアのテスト環境を確立する場合に役に立ちます。データ・ポンプ・エクスポート/インポートでは、データのサブセット化機能を柔軟に使用できます。
- バックアップ・アーカイブとして機能します。データベースの全体エクスポートを現行のデータベースのアーカイブとして使用できます。
- 以前のリリースのデータベースが配置されているプラットフォームとは異なるオペレーティング・システムまたはハードウェア・プラットフォームにアップグレード後のデータベースを構築できます。
- ネットワークベースのデータ・ポンプ・インポートによって、以前のリリースのOracle Databaseのネットワーク全体に新しいリリースのOracle Databaseを直接ロードできます。ネットワークベースのデータ・ポンプ・インポートを使用することで、介在するダンプ・ファイルを使用する必要がなくなります。

親トピック: Oracle Databaseのアップグレード時のデータの移行でエクスポート/インポートする方法

エクスポート/インポートを使用したデータ移行の時間要件

Oracle Data Pumpを使用したデータ移行にかかる時間を理解します。

エクスポート/インポートを使用したOracle Data Pumpの使用によるOracle Database全体の移行には、特にDBUAを使用する場合または手動アップグレードを実行する場合と比較して、時間が長くかかる場合があります。Oracle Data Pumpを使用してデータを移行する場合、ピーク時以外に移行をスケジュールするか、アップグレード時に現在のデータベースに対して行われる変更を新しいデータベースに伝播するようにプロビジョニングします。

親トピック: Oracle Databaseのアップグレード時のデータの移行でエクスポート/インポートする方法

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。

Oracle Databaseのアップグレードには、AutoUpgradeユーティリティを使用することをお薦めします。ただし、Oracle Database 19cでも、引き続きDBUAを使用してマルチテナント・アーキテクチャのコンテナ・データベース(CDB)およびプラガブル・データベース(PDB)をアップグレードできます。

DBUAは、アップグレード前パラメータ(autoupgrade.jar -preupgrade -mode)を使用してAutoUpgrradeを起動します。これにより、一部の構成設定がアップグレードに必要な値に修正されます。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のアップグレード方法の選択

アップグレード時のOracleホームの新しい場所の選択

データベースをアップグレードまたはパッチ適用したら、新しいOracleホームを新しい場所にインストールする必要があります(アウトオブプレース・アップグレードまたはパッチ適用)。

AutoUpgradeでは、アウトオブプレース・アップグレードおよびパッチ適用が実行されます。つまり、アップグレードまたはパッチ適用済のOracleホームは新しいOracleホームに置かれます。以前は、インプレース・アップグレードまたはパッチ適用によって、アップグレードまたはパッチが既存のOracleホームに適用されていました。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。別のインストール場所を使用することで、本番環境を完全に置き換える前に、アウトオブプレースのOracleホーム・データベースでアップグレードまたはパッチ適用プロセスをテストできます。

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、アップグレードはアウトオブプレース・アップグレードである必要があります。つまり、新しいOracleホームをインストールするには新しい場所が必要です。

関連項目

- データベースとGIのパッチ適用を高速化する方法

親トピック: Oracle Databaseのアップグレードを準備する作業

Oracle Databaseのアップグレードでのテスト計画の開発

アップグレード処理のすべてのステージを検証するために慎重に設計された一連のテストを作成する方法を理解するには、これらのトピックを参照してください。

データベースとアプリケーションの厳密なテストを実行することをお薦めします。テストを正常に実行して完了すると、本番データベースのアップグレード処理を確実に理解して、アップグレード処理を予測可能な状態で正常に実行できます。本番データベースをアップグレードする前に、可能なかぎり十分なテストを実行することを強くお薦めします。完全で反復可能なテスト処理は、非常に重要です。

手動でテストを実行するか、テストを支援するユーティリティ(たとえば、データベース・リプレイやSQLパフォーマンス・アナライザなどのOracle Real Application Testing機能)を使用するかを選択できます。どちらの場合でも、実行するテストのタイプは同じです。

テスト計画には次のタイプのテストを含める必要があります。

- アップグレード・テスト
  Oracle Databaseを新しいリリースにアップグレードする場合、アップグレード計画を作成、テストおよび検証することをお薦めします。
- 最小テスト
  新しいリリースのテスト環境でアプリケーションの最小テストを実行すると、アプリケーションの起動または呼出しに関する問題を検出できます。
- アップグレード後の機能テスト
  アップグレードが完了した後に、アップグレードしたOracle Databaseの機能テストを実行します。
- 高可用性テスト
  サービス・レベル契約を継続して満たせるようにするには、アップグレードしたOracle Databaseシステムで高可用性テストの実行を計画します。
- アプリケーションの互換性を確保する統合テスト
  Oracle Databaseの統合テストでは、システムのコンポーネント間で相互作用を調査します。
- アップグレードしたOracle Databaseのパフォーマンス・テスト
  以前のリリースとアップグレードしたOracle Database間でパフォーマンス・テストの比較を計画します。
- Oracle Databaseのアップグレードでのボリューム/ロード・ストレス・テスト
  アップグレードされたOracle Database全体のボリューム/ロード・ストレス・テストを大きいボリュームとロードの状態で実行するには、データベース・リプレイを使用します。
- Oracle Databaseアップグレード計画のテスト計画のガイドライン
  現行のデータベースおよび新しいOracle Databaseリリースにアップグレードしたテスト・データベースに対して、計画したテストを実行します。

親トピック: Oracle Databaseのアップグレードを準備する作業

アップグレード・テスト

Oracle Databaseを新しいリリースにアップグレードする場合、アップグレード計画を作成、テストおよび検証することをお薦めします。

Oracle Databaseのアップグレード・テストでは、現行のOracle Databaseソフトウェアから新しいOracle Databaseリリースへのアップグレード・パスの計画とテストを実行します。Oracle Database Upgrade Assistant (DBUA)を使用するか、

手動アップグレードを実行するか、またはAutoUpgradeユーティリティを使用するかに関係なく、アップグレードを計画およびテストすることをお薦めします。Oracle Data Pumpエクスポート/インポートなどのデータ移行方法または他のデータコピー方法を使用する場合も、計画およびテストが適用されます。選択するアップグレードまたはデータ移行方法に関係なく、変更を計画、テスト、および検証する必要があります。

親トピック: Oracle Databaseのアップグレードでのテスト計画の開発

最小テスト

新しいリリースのテスト環境でアプリケーションの最小テストを実行すると、アプリケーションの起動または呼出しに関する問題を検出できます。

Oracle Databaseの最小テストでは、現行のデータベースからアプリケーションの全部または一部を新しいデータベースに移動し、データベースの新機能を使用可能にしないでアプリケーションを実行します。最小テストでは、実際の本番環境で発生する問題が明らかにならない可能性があります。ただし、アプリケーションの起動または呼出しに関する問題がある場合は、すぐに検出されます。

親トピック: Oracle Databaseのアップグレードでのテスト計画の開発

アップグレード後の機能テスト

アップグレードが完了した後に、アップグレードしたOracle Databaseの機能テストを実行します。

Oracle Databaseの機能テストとは、システムの新機能と既存機能をアップグレード後にテストする一連のテストです。機能テストには、すべてのデータベース、ネットワーキングおよびアプリケーション・コンポーネントのテストが含まれます。機能テストの目的は、システムの各コンポーネントがアップグレード前と同様に機能し、新機能が正常に動作していることを検証することです。

親トピック: Oracle Databaseのアップグレードでのテスト計画の開発

高可用性テスト

サービス・レベル契約を継続して満たせるようにするには、アップグレードしたOracle Databaseシステムで高可用性テストの実行を計画します。

Oracle Databaseの高可用性テストでは、アップグレードしたデータベース・システムが次のリカバリ・ビジネス要件を満たしていることを確認します。

- リカバリ時間目標(RTO)
- リカバリ・ポイント目標(RPO)

高可用性テストでは、次のテスト手順をお薦めします。

- ストレス・テスト中にノードまたはインスタンス障害を作成します。ノードまたはインスタンス障害は、Oracle RACのリカバリ機能を評価するために役立ちます。
- アップグレードしたデータベースの停止時間を最小化できるように、フォールバック計画および手順をテストします。
- データベースのパフォーマンスおよび安定性を確認し、パフォーマンスの問題を解決します。パフォーマンスの問題を解決すれば、割り当てた時間内にアップグレード・プロセスが確実に実行できるようになります。

親トピック: Oracle Databaseのアップグレードでのテスト計画の開発

アプリケーションの互換性を確保する統合テスト

Oracle Databaseの統合テストでは、システムのコンポーネント間で相互作用をテストします。

統合テストの一部として次のテストを実行することをお薦めします。

- Pro*C/C++アプリケーションとアップグレードしたデータベースとの互換性を確認するには、アップグレードしたOracle Databaseを使用してPro*C/C++アプリケーション・クライアントをテストします
- グラフィカル・ユーザー・インタフェースをテストします。
- データベースと直接的または間接的に相互作用するすべてのアプリケーションをテストします。アップグレードされたOracle Databaseのデータ型やデータ・ディクショナリのデータなどのわずかな変更(データ・ディクショナリの追加行、オブジェクト型の変更など)が、アップグレードされたOracle Databaseインスタンスに直接接続されていないフロントエンド・アプリケーションにも影響する可能性があります。
- コンポーネント間のOracle NetまたはOracle Net Services接続のテストおよびストレス・テストを行います。

関連項目

- C++アプリケーション
- Oracle Net Servicesのアップグレードに関する考慮事項

親トピック: Oracle Databaseのアップグレードでのテスト計画の開発

アップグレードしたOracle Databaseのパフォーマンス・テスト

以前のリリースとアップグレードしたOracle Database間でパフォーマンス・テストの比較を計画します。

アップグレードしたOracle Databaseのパフォーマンス・テストでは、新しいデータベースでの様々なSQL文のパフォーマンスを、現行データベースでの同じ文のパフォーマンスと比較します。アップグレードする前に、現在のOracle Databaseリリースでのアプリケーションのパフォーマンス・プロファイルを分析します。特に、アプリケーションがデータベース・サーバーに対して実行するコールを分析して理解します。

本番システムと同じストレージ、データおよびその他の特性を持つテスト・システムを設定することをお薦めします。

- データベース・リプレイとパフォーマンス・テスト
  データベース・リプレイ機能を使用して、本番データベースを実際にアップグレードする前に、ユーザーの本番ワークロード上でデータベースをアップグレードする現実的なテストを実行できます。
- SQLパフォーマンス・アナライザ
  SQLパフォーマンス・アナライザを使用して、システムの変更がSQLワークロードに及ぼす影響を予測します。
- SQL計画管理
  パフォーマンスの低下を避けるためにアップグレード後にSQL計画管理を実行する方法を理解するには、このトピック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テスト計画の開発

データベース・リプレイおよびパフォーマンス・テスト

新しいデータベース・リプレイ機能を使用して、本番データベースを実際にアップグレードする前に、ユーザーの本番ワークロード上でデータベースをアップグレードする現実的なテストを実行できます。

データベース・リプレイ機能は、本番システム上で実際のデータベースのワークロードを取得して、これをテスト・システム上でリプレ

イします。また、データベース・リプレイでは、潜在的な問題(エラーの発生やパフォーマンスの相違など)を示す分析とレポート作成も提供されます。さらに、自動データベース診断モニター、自動ワークロード・リポジトリ(AWR)およびアクティブ・セッション履歴など、定期的なEnterprise Managerパフォーマンスの監視およびレポート作成ツールをすべて使用して、問題に対処します。

> ノート:
>
> データベース内のストアド・プロシージャのロジックを変更できます。ただし、アプリケーション・ロジックを実装しているストアド PL/SQL プロシージャでは、アップグレード前と同じインタフェースを維持する必要があります。アプリケーションのストアド・プロシージャがアップグレードによって影響を受ける場合は、ワークロードをリプレイできない場合があります。同じインタフェースでデータベース・リプレイ・ツールを使用すると、サーバー内の新しいアプリケーション・ロジックがアップグレード後にも期待どおりに動作するかどうかを確認でき、有効な診断が行えます。

参照:

- ワークロードの取得およびリプレイ方法の詳細は、『Oracle Database Testingガイド』を参照してください。
- 自動ワークロード・リポジトリの詳細は、『Oracle Databaseパフォーマンス・チューニング・ガイド』を参照してください。

親トピック: アップグレードしたOracle Databaseのパフォーマンス・テスト

SQLパフォーマンス・アナライザ

SQLパフォーマンス・アナライザを使用して、システムの変更がSQLワークロードに及ぼす影響を予測します。

SQLパフォーマンス・アナライザによって、SQLワークロードへのアップグレードの影響を評価できます。SQLパフォーマンス・アナライザはアップグレードにより影響を受けるSQL文を特定することにより、起こりうる問題を検出します。次に、アップグレード前とアップグレード後のSQLワークロードのパフォーマンスの相違を測定します。この分析により、ユーザーはSQLパフォーマンスに対する全体的な影響を評価できます。そうすれば、SQLワークロードの変更がユーザーに影響を与える前に、変更の結果の悪影響を避けるための対策を講じることができます。

参照:

潜在的なデータベース変更の分析にSQLパフォーマンス・アナライザを使用した詳細および例は、『Oracle Database Testingガイド』を参照してください。

親トピック: アップグレードしたOracle Databaseのパフォーマンス・テスト

SQL計画管理

パフォーマンスの低下を避けるためにアップグレード後にSQL計画管理を実行する方法を理解するには、このトピックを参照してください。

新しいバージョンのオプティマイザをインストールするデータベースのアップグレードでは、通常、ほんのわずかなSQL文に対して計画の変更が発生します。ほとんどの計画の変更においてパフォーマンスに変化や向上は見られません。ただし、ある一部の計画変更では、パフォーマンスが低下する場合があります。SQL計画の管理を使用すると、SQL計画の情報を取得、選択および改良するためのコンポーネントが用意されているため、SQL文の実行計画に対する突然の変更によるパフォーマンスの低下を回避できます。SQL計画管理は、一定期間におけるSQL文の実行計画を記録および評価する予防的メカニズムであり、反復的な

使用により効率的であることが証明されている既存の計画セットで構成されたSQL計画のベースラインを構築します。SQL計画管理では、システムで発生している変更に関係なく、対応するSQL文のパフォーマンスを保持するためにSQL計画のベースラインを使用します。

SQL計画管理を使用すると、オプティマイザによって実行計画が自動的に管理され、既知の計画または検証済の計画のみが使用されます。SQL計画管理は、SQL文に新しい計画を見つけると、データベースが新しい計画の互換性を確認するか、現在の計画よりも高いパフォーマンスを検出するまで、この計画を使用しません。SQL計画管理を現行の実行計画でシードすると、それらの計画は各文のSQL計画のベースラインになります。オプティマイザは、アップグレード後にこれらの計画を使用します。異なる計画でパフォーマンスが向上する可能性があるとOracle Database 12cオプティマイザが判断した場合でも、新しい計画は検証のためにキューに格納され、現行の計画と同等かそれ以上のパフォーマンスを実現できる計画であることが確認されるまでは、使用されません。

SQL管理ベース(SMB)を実行計画にシードまたは移入するには、複数の方法があります。

カーソル・キャッシュからのSQL管理ベースのバルク・ロード

実行計画またはSQL計画ベースラインのカーソル・キャッシュからのバルク・ロードは、データベースをOracle Database 11gからOracle Databaseの最新リリースにアップグレードする場合に役立ちます。カーソル・キャッシュは共有SQL領域で、バルク・ロードされるSQL計画は、自動的にそのまま使用され、SQL計画ベースラインとして既存または新規の計画履歴に追加されます。

1. Oracle Databaseのソース・リリースでは、`DBMS_SPM.LOAD_PLAN_FROM_CURSOR_CACHE`プロシージャまたはOracle Enterprise Managerを使用して、カーソル・キャッシュ内の実行計画をすべてSQL管理ベースにロードします。
2. データベースをアップグレードします。

参照:

共有SQL領域からPL/SQLまたはOracle Enterprise Managerを使用して計画をロードする方法の詳細は、『Oracle Database SQLチューニング・ガイド』を参照してください。

SQLチューニング・セット(STS)を使用したSQL管理ベースのバルク・ロード

実行計画またはSQL計画のベースラインのバルク・ロードは、SQLチューニング・セットを使用して実行できる場合があります。これは、カーソル・キャッシュから直接ロードするため、または自動ワークロード・リポジトリから履歴計画をロードするためのSQL Management Base (SMB)がないOracle Database 10gからアップグレードを行う場合に役立ちます。

1. Oracle Databaseのソース・リリースで、各SQL文の実行計画を含むSTSを作成します。
2. STSをステージング表にロードし、ステージング表をダンプ・ファイルにエクスポートします。
3. ステージング表をダンプ・ファイルからOracleの新しいリリースにインポートし、STSをアンロードします。
4. Oracle Enterprise Managerまたは`DBMS_SPM.LOAD_PLANS_FROM_SQLSET`を使用して、実行計画をSQL管理ベースにロードします。

参照:

実行計画またはSQL計画のベースラインのバルク・ロードの詳細な手順は、『Oracle Database SQLチューニング・ガイド』を参照してください。

ステージング表からの既存のSQL計画ベースラインの展開

すべての重要なSQL問合せをOracle Databaseのテスト環境でテストおよびチューニングしてから、そのSQL実行計画をOracle Databaseの本番環境に移動できます。または、アップグレード前Oracle Databaseの本番環境からSQL問合せの計画を実行してから、アップグレード後の本番環境にそれらを移動できます。

1. Oracle Database 12cのテスト・システムで、すべてのテストとチューニングを完了してから、`DBMS_SPM.LOAD_PLAN_FROM_CURSOR_CACHE`プロシージャまたはEnterprise Managerを使用して、カーソル・キャッシュ内の実行計画をすべてSQL管理ベースにロードします。
2. `DBMS_SPM.CREATE_STGTAB_BASELINE`プロシージャを使用して、ステージング表を作成します。
3. `DBMS_SPM.PACK_STGTAB_BASELINE`ファンクションを使用して、ステップ1で作成したSQL計画ベースラインをステージング表に格納します。
4. データ・ポンプを使用して、ステージング表をフラット・ファイルにエクスポートします。
5. このフラット・ファイルをターゲット・システムに転送します。
6. データ・ポンプを使用して、フラット・ファイルからステージング表をインポートします。
7. `DBMS_SPM.UNPACK_STGTAB_BASELINE`ファンクションを使用して、ステージング表からターゲット・システムのSQL管理ベースにSQL計画ベースラインを展開します。

- SQL計画管理を実行する理由
  Oracle Databaseのアップグレード後のパフォーマンスの低下を防ぐために、SQL計画管理を実行します。
- カーソル・キャッシュからのSQL管理ベースのバルク・ロード
  カーソル・キャッシュからの実行計画またはSQL計画ベースラインのバルク・ロードは、以前のリリースをOracle Databaseの最新リリースにアップグレードする場合に役立ちます。
- SQLチューニング・セット(STS)を使用したSQL管理ベースのバルク・ロード
  実行計画またはSQL計画ベースラインのバルク・ロードは、自動ワークロード・リポジトリから履歴計画をロードする場合に役立ちます。
- ステージング表からの既存のSQL計画ベースラインの展開
  `DBMS_SPM.LOAD_PLAN_FROM_CURSOR_CACHE`を使用してテスト用に移行可能なステージング表を作成することにより、重要なSQL問合せおよび実行計画をテストします。

参照:

- ステージング表からの計画のロードの詳細は、『Oracle Database SQLチューニング・ガイド』を参照してください。
- データ・ポンプの使用については、『Oracle Databaseユーティリティ』を参照してください。

親トピック: アップグレードしたOracle Databaseのパフォーマンス・テスト

SQL計画管理を実行する理由

Oracle Databaseのアップグレード後のパフォーマンスの低下を防ぐために、SQL計画管理を実行します。

新しいバージョンのオプティマイザをインストールするOracle Databaseのアップグレードでは、通常、ごく一部のSQL文に計画の変更が発生します。ほとんどの計画の変更においてパフォーマンスに変化や向上は見られません。ただし、ある一部の計画変更は、パフォーマンスが低下する原因となる場合があります。SQL計画の管理を使用すると、SQL計画の情報を取得、選択および改良するためのコンポーネントが用意されているため、SQL文の実行計画に対する突然の変更によるパフォーマンスの低下

を回避できます。SQL計画管理は、一定期間におけるSQL文の実行計画を記録および評価する予防的メカニズムであり、反復的な使用により効率的であることが証明されている既存の計画セットで構成されたSQL計画のベースラインを構築します。SQL計画管理では、システムで発生している変更に関係なく、対応するSQL文のパフォーマンスを保持するためにSQL計画のベースラインを使用します。

SQL計画管理を使用すると、オプティマイザによって実行計画が自動的に管理され、既知の計画または検証済の計画のみが使用されます。SQL計画管理は、SQL文に新しい計画を見つけると、データベースが新しい計画の互換性を確認するか、現在の計画よりも高いパフォーマンスを検出するまで、この計画を使用しません。SQL計画管理を現行の実行計画でシードすると、それらの計画は各文のSQL計画のベースラインになります。オプティマイザは、アップグレード後にこれらの計画を使用します。アップグレードされたOracle Databaseオプティマイザが、異なる計画でパフォーマンスが向上する可能性があると判断した場合、新しい計画は検証のためにキューに入れられます。新しい計画は、現在の計画と同等かそれよりも高いパフォーマンスが得られると確認されるまで使用されません。

親トピック: SQL計画管理

カーソル・キャッシュからのSQL管理ベースのバルク・ロード

実行計画またはSQL計画ベースラインのカーソル・キャッシュからのバルク・ロードは、以前のリリースをOracle Databaseの最新リリースにアップグレードする場合に役立ちます。

カーソル・キャッシュは、共有SQL領域です。バルク・ロードされるSQL計画は、自動的にそのまま使用され、SQL計画ベースラインとして既存または新規の計画履歴に追加されます。

1. Oracle Databaseのソース・リリースでは、`DBMS_SPM.LOAD_PLAN_FROM_CURSOR_CACHE`プロシージャまたはOracle Enterprise Managerを使用して、カーソル・キャッシュ内の実行計画をすべてSQL管理ベースにロードします。
2. データベースをアップグレードします。

関連項目

- 共有SQL領域からの計画のロード

親トピック: SQL計画管理

SQLチューニング・セット(STS)を使用したSQL管理ベースのバルク・ロード

実行計画またはSQL計画ベースラインのバルク・ロードは、自動ワークロード・リポジトリから履歴計画をロードする場合に役立ちます。

実行計画またはSQL計画のベースラインのバルク・ロードは、SQLチューニング・セットを使用して実行できる場合があります。これは、以前のOracle Database自動ワークロード・リポジトリから履歴計画をロードする場合に役立ちます。

1. Oracle Databaseのソース・リリースで、各SQL文の実行計画を含むSTSを作成します。
2. STSをステージング表にロードし、ステージング表をダンプ・ファイルにエクスポートします。
3. ステージング表をダンプ・ファイルからOracleの新しいリリースにインポートし、STSをアンロードします。
4. Oracle Enterprise Managerまたは`DBMS_SPM.LOAD_PLANS_FROM_SQLSET`を使用して、実行計画をSQL管理ベースにロードします。

関連項目

- SQLチューニング・セットからの計画のロード

親トピック: SQL計画管理

ステージング表からの既存のSQL計画ベースラインの展開

DBMS_SPM.LOAD_PLAN_FROM_CURSOR_CACHEを使用してテスト用に移行可能なステージング表を作成することにより、重要なSQL問合せおよび実行計画をテストします。

すべての重要なSQL問合せをOracle Databaseのテスト環境でテストおよびチューニングしてから、そのSQL実行計画をOracle Databaseの本番環境に移動できます。または、アップグレード前Oracle Databaseの本番環境からSQL問合せの計画を実行してから、アップグレード後の本番環境にそれらを移動できます。

1. 新しいOracle Databaseリリースのテスト・システムで、すべてのテストとチューニングを完了してから、DBMS_SPM.LOAD_PLAN_FROM_CURSOR_CACHEプロシージャまたはEnterprise Managerを使用して、カーソル・キャッシュ内の実行計画をすべてSQL管理ベースにロードします。
2. DBMS_SPM.CREATE_STGTAB_BASELINEプロシージャを使用して、ステージング表を作成します。
3. DBMS_SPM.PACK_STGTAB_BASELINEファンクションを使用して、ステップ1で作成したSQL計画ベースラインをステージング表に格納します。
4. Oracle Data Pumpを使用して、ステージング表をフラット・ファイルにエクスポートします。
5. このフラット・ファイルをターゲット・システムに転送します。
6. Oracle Data Pumpを使用して、フラット・ファイルからステージング表をインポートします。
7. DBMS_SPM.UNPACK_STGTAB_BASELINEファンクションを使用して、ステージング表からターゲット・システムのSQL管理ベースにSQL計画ベースラインを展開します。

関連項目

- ステージング表からの計画のロード
- Oracle Data Pumpの概要

親トピック: SQL計画管理

Oracle Databaseのアップグレードでのボリューム/ロード・ストレス・テスト

アップグレードされたOracle Database全体のボリューム/ロード・ストレス・テストを大きいボリュームとロードの状態で実行するには、データベース・リプレイを使用します。

Oracleリプレイは、アップグレードされたOracle Databaseリリースを本番に移行する前に、ロードの問題を明らかにするために役立ちます。**ボリューム**とは、操作されるデータの量を表します。**ロード**とは、システム上の同時要求のレベルを表します。したがって、実際の本番システムのボリュームとロードを取得してリプレイするときに、アップグレードしたOracle Databaseのロードをエミュレートし、様々なボリュームとロードでのパフォーマンスを観察できます。

ボリューム/ロード・ストレス・テストは非常に重要です。しかしそれは一般に見過ごされています。アップグレードの後、一部の顧客がどのようなボリューム/ロード・ストレス・テストも実行しないことがわかりました。そのかわりに、ビジネス・アプリケーションを特に考慮しているわけではないベンチマークが広く利用されています。ベンチマークは重要です。アプリケーションのベンチマークを実行することをお薦めします。ベンチマークは機能、パフォーマンス、統合に関連した問題を明らかにします。ただし、ベンチマークを使用してもボリューム/ロード・ストレス・テストを置き換えることはできません。

ロード・テストには、新しいOracle Databaseリリースに対して、同じデータおよびインフラストラクチャの環境を使用してアプリケーションのロードを実行することが含まれます。ロード・テストを実行する際に、アプリケーションに、新しいエラーや、本番環境で

発生すると考えられるロード条件下でのパフォーマンスの問題などの問題が発生しないことを確認します。多くの場合、問題は特定のロード条件でのみ明らかになるため、通常、機能テストでは見つかりません。データベース・リプレイ機能はそうしたロード・テストに最適です。データベース・リプレイでは、本番環境のシステム・ワークロードを取得し、テスト・システムでそれを同じようにリプレイできます。

関連項目

- データベース・リプレイの概要

親トピック: Oracle Databaseのアップグレードでのテスト計画の開発

Oracle Databaseアップグレード計画のテスト計画のガイドライン

現行のデータベースおよび新しいOracle Databaseリリースへアップグレードされたテスト・データベースに対して、計画したテストを実行します。

- テスト結果を比較し、相違点を記録します。
- 問題が解決されるまで、必要な回数、テスト・アップグレードを繰り返します。

新しいOracle Databaseで既存のアプリケーションが正常に動作するかを確認するために、この新しくアップグレードされたテスト・データベースをテストします。

- 利用可能なOracle Databaseの機能を追加して、拡張された機能および新しい機能についてテストします。
- アプリケーションが現行のデータベースの場合と同様に動作することを確認します。

参照:

データベースのアップグレードのテストの詳細は、『Oracle Database Testing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テスト計画の開発

スキーマ専用アカウントおよびEXPIREDパスワード・アカウントのアップグレード

アップグレードを開始する前に、パスワードがEXPIREDステータスであり、そのアカウントがLOCKEDステータスになっている場合にデフォルトのOracle Databaseアカウントにパスワード認証を使用するかどうかを判断します

Oracle Database 19c以降のリリースへのアップグレード中に、アップグレード前にパスワードがリセットされておらず(EXPIREDステータスに設定され)、さらにLOCKEDステータスにも設定されているデフォルトのOracleアカウントは、アップグレードの完了後にNO AUTHENTICATIONに設定されます。

この新機能により、スキーマ専用アカウントに変更されたデフォルトのアカウントは、パスワード認証に使用できなくなります。この機能の利点は、管理者がこれらのOracle Database提供のスキーマのパスワードを定期的にローテーションする必要がなくなったことです。この機能により、攻撃者がデフォルトのパスワードを使用してこれらのアカウントにハッキングするセキュリティ・リスクも削減されます。

アップグレード時にこれらのOracleアカウントがスキーマ専用アカウントに設定されないようにする場合、アップグレードの開始前にアカウントに有効な強力なパスワードを設定するか、アップグレード後にこれらのアカウントに有効な強力なパスワードを設定するか、アップグレードされたOracle Databaseにログインする前にアカウントのロックを解除する必要があります。

アップグレード後、管理者はスキーマ専用アカウントのパスワード認証を有効にすることもできます。ただし、セキュリティを強化するために、これらのアカウントをスキーマ専用のアカウントとして保持することをお薦めします。

関連項目

- 『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準備する作業

ダウングレードおよびリカバリ・オプションを保持するためのファイルのバックアップ

必要に応じて、アップグレードの問題からリカバリして以前のリリースにダウングレードできるように、データベースおよび特定のファイルのバックアップ計画の実装をお薦めします。

- 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
  アップグレードの成功を保証する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 Data Guard Brokerの構成ファイルおよびダウングレード
  Oracle Database 19c以上のリリースへのアップグレードにより、以前のリリースにダウングレードする機能を保持するためにData Guard Broker構成ファイルをバックアップする必要があります。
- ブローカ構成のエクスポート
  `EXPORT CONFIGURATION`コマンドを使用して、ブローカ構成ファイルに含まれるメタデータをテキスト・ファイルにエクスポートします。

親トピック: Oracle Databaseのアップグレードを準備する作業

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。

Oracle Database Enterprise Editionの場合、主なフォールバック・メカニズムはフラッシュバック・データベースです。ただし、フラッシュバック・データベースを使用して切断/接続アップグレードを元に戻すことはできません。切断/接続アップグレードの場合は、RMANバックアップなどの他のフォールバック計画を使用します。

AutoUpgradeを使用する場合は、AutoUpgrade構成ファイル内で`target_pdb_copy_option=file_name_convert`を指定することをお薦めします。ここで、`file_name_convert`は、データ・ファイルに接頭辞として付ける変換パターンです。これを行うと、AutoUpgradeは、データベースを接続する前にデータ・ファイルのコピーを作成するようにデータベースに指示します。この方法の使用を選択すると、元のデータベースをフォールバックとして使用できます。ただし、データ・ファイルのコピーを作成する場合はアップグレードにさらにディスク領域と時間が必要になることに留意してください。

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。
- バックアップが正しく動作するようにテストされているか。
- 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)。
- 障害時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・アドバンスト・ユーザーズ・ガイドを参照してください。

また、ダウングレードの準備を確実に行うには、ダウングレードに関する章を確認し、リリースの準備に必要な準備ステップを完了してください。

関連項目

- データベースのバックアップ
- フラッシュバック・データベースおよびリストア・ポイントの使用

親トピック: ダウングレードおよびリカバリ・オプションを保持するためのファイルのバックアップ

Oracle Data Guard Brokerの構成ファイルおよびダウングレード

Oracle Database 19c以上のリリースへのアップグレードにより、以前のリリースにダウングレードする機能を保持するためにData Guard Broker構成ファイルをバックアップする必要があります。

Oracle Database 19cより前のリリースでは、Oracle Data Guard Brokerプロパティにマップされる Oracle Database設定はOracle Data Guard Broker構成ファイルに保持され、DGMGRLコマンドライン・インタフェースを使用して変更できます。ただし、Oracle Database 19c以上では、これらのデータベース設定はブローカ構成ファイルに格納されなくなりました。この変更の結果、DGMGRLを使用して引き続きこれらのプロパティを変更できますが、変更した値はOracle Data Guard Broker構成ファイルに格納されなくなります。かわりに、DGMGRLコマンドで、Oracle Data Guard Brokerプロパティがマップされる Oracle Database初期化パラメータまたはデータベース設定を直接変更します。

プロパティ設定の管理方法に対するこの変更のために、Oracle Data Guard Brokerを使用する場合、アップグレードを開始する前に、以前のリリースのOracle Data Guard Broker構成ファイルを安全なバックアップの場所にエクスポートすることをお薦めします。アップグレード前にOracle Data Guard Broker構成ファイルをバックアップしない場合、アップグレード後は、以前のリリースにダウングレードできなくなり、以前にOracle Data Guardに選択したプロパティ・オプションを保持できなくなります。

親トピック: ダウングレードおよびリカバリ・オプションを保持するためのファイルのバックアップ

ブローカ構成のエクスポート

EXPORT CONFIGURATIONコマンドを使用して、ブローカ構成ファイルに含まれるメタデータをテキスト・ファイルにエクスポートします。

ブローカ構成ファイルが格納されるディレクトリは、Oracleサーバー・プロセスからアクセスできる必要があります。

1. プライマリ・データベースに接続します。

```
DGMGRL> CONNECT sysdg@North_Sales.example.com;
Password: password
Connected to "North_Sales"
Connected as SYSDG.
```

2. ブローカ構成をエクスポートします。

   次のコマンドは、ブローカ構成をエクスポートし、トレース・ディレクトリ内のmyconfig.txtという名前のファイルに格納します。

```
DGMGRL> EXPORT CONFIGURATION TO 'myconfig.txt';
Succeeded.
```

親トピック: ダウングレードおよびリカバリ・オプションを保持するためのファイルのバックアップ

AutoUpgradeでのアップグレード前情報チェック

Oracle Databaseをアップグレードする前に完了する必要があるタスクのチェックリストを取得するには、`analyze`モードでAutoUpgradeユーティリティ(`autoupgrade.jar`)を実行します。

Oracleでは、Oracle Databaseをアップグレードする前に、AutoUpgradeの最新リリースをダウンロードして`-analyze`モードで実行することをお薦めします。AutoUpgradeでは、アップグレードを開始する前に対処する必要がある問題を識別できます。場合によっては、一部の問題を解決できるスクリプトをAutoUpgradeで生成することもできます。

> ヒント:
>
> Mike Dietrich のアップグレードに関するブログで、アップグレードの準備に役立つヒントおよび推奨事項を確認することを検討してください。Oracle Database 19c の場合、My Oracle Support のノート Oracle Database 19c の重要な推奨個別パッチ(ドキュメント ID 555.1)を確認して、Oracle Database 19c の最も重要なパッチ・バンドルに関する情報を取得します。

関連項目

- My Oracle Support AutoUpgradeツール(ドキュメントID 2485457.1)
- My Oracle Support Oracle Database 19cの重要な推奨個別パッチ(ドキュメントID 555.1)
- Upgrade your Database – NOW! Mike Dietrich's Oracle Database Upgrade Blog

親トピック: Oracle Databaseのアップグレードの準備

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ことで、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のインストールが支援されます。

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:

1. Oracleのオペレーティング・システム固有のドキュメントに従って、Oracle Databaseソフトウェアのインストールを準備します。
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. 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: Oracle Databaseのアップグレードの準備

Oracle RAC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ことで、Oracle RAC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のインストールが支援されます。

> ノート:
>
> 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。

Oracle RACデータベースをアップグレードする場合は、記述されている順番で次のステップを実行する必要があります。

1. Oracle Clusterwareのアップグレード:
   a. ご使用のオペレーティング・システムの『Oracle Grid Infrastructureインストレーション・ガイド』の記載に従って、最初にOracle Clusterwareをアップグレードします。
   b. プロンプトが表示されたら、別のターミナル・セッションを開き、`root`でログインして`root.sh`を実行します。
2. Oracle Clusterwareをアップグレードした後、Oracleのオペレーティング・システム固有のドキュメントに従って、Oracle Databaseソフトウェアのインストールを準備します。
3. Oracle Universal Installerを起動し、ソフトウェアをインストールします。

   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。
4. `preupgrade`パラメータを指定してAutoUpgradeを`analyze`モードで実行します。AutoUpgradeでは、多くの問題を自動的に修正し、生成される`prefixups`ファイルに他の問題をリストできます。これは、アップグレード前に手動で修正できます。
5. デプロイ・モードでAutoUpgradeを実行します。

関連項目

- 使用しているプラットフォームの『Oracle Clusterwareインストレーション・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リリースの更新および要件
  アップグレードを開始する前に、新しいリリースのOracleホームを最新のリリース更新(Update)に更新します。
- アップグレードおよび透過的データ暗号化
  TDEを使用してデータベースをアップグレードするには、`-load_password`コマンドライン・オプションを使用するか、外部パスワード・ストアを指定して、AutoUpgradeにTDEパスワードを指定します。
- Oracle Databaseのアップグレード時のOracle Net Servicesに関する推奨事項
  リスナーが新しいリリースのOracleホームで実行されていることを確認する必要があります。
- ORAPWDを使用したパスワード・ファイルの作成または移行
  REMOTE_LOGIN_PASSWORDFILEが設定されている場合に参照してください。
- 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)以降のリリースでは、排他モード認証プロトコルが使用されます。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。
- STIGおよびCISプロファイルのリソースおよびパスワード・パラメータの更新
  Oracle Database 21c以降、アップグレードでは、既存のSTIGプロファイルの更新、およびアップグレードの一環としてのCISプロファイルのインストールを含むOracle推奨プロファイルが構成されます。
- プロファイル・スクリプト(glogin.sqlおよびlogin.sql)の確認
  すべてのアップグレード方法で、プロファイル・スクリプトを使用せずにアップグレードを実行することをお薦めします。
- 読取り専用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。
- Oracle Databaseの高可用性オプション
  Standard Edition High Availability、Oracle Restart、Oracle Real Application Clusters (Oracle RAC)およびOracle RAC One Nodeを使用して、Oracle Databaseで使用可能な高可用性オプションを確認します。
- Oracle Database Standard Editionでの高可用性のオプション
  Oracle Database 19c以降のリリースでOracle Database Standard Editionの高可用性を可能にするために、Standard Edition High Availabilityの使用方法について学習します。
- 統合監査証跡へのオペレーティング・システムの監査レコードの移動
  スピルオーバー監査ファイルに書き込まれた監査レコードは、統合監査証跡データベース表に移動できます。
- 非CDBのアップグレードとOracle GoldenGate
  Oracle GoldenGateがデプロイされている非CDB Oracle Databaseをアップグレードする場合は、Oracle GoldenGateを停止し、マルチテナント・アーキテクチャ用に変換およびアップグレードした後で再構成する必要があります。
- AutoUpgradeを使用する前の非常に大規模なデータベースのバックアップ

大規模なデータベースで部分オフライン・バックアップを使用する場合は、データベースをダウングレードする必要がある場合に停止時間を最小限に抑えるために、表領域をチェックし、リカバリに必要なすべての表領域がバックアップされていることを確認します。

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リリースの更新および要件

アップグレードを開始する前に、新しいリリースのOracleホームを最新のリリース更新(Update)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。

アップグレードを開始する前に、新しいリリースのOracleホームを最新の四半期リリース更新(Update)に更新することをお薦めします。

My Oracle Supportで、最新更新の入手方法に加え、ライフサイクル管理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。たとえば:

- My Oracle Supportノート555.1のOracle Database 19cの重要な推奨個別パッチには、Oracle Database 19cの特定の重要度のパッチのリストが含まれています。
- My Oracle Supportノート2118136.2には、環境に必要な更新、リビジョン、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・リリースを選択する際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薦めします。
- 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。

関連項目

- Oracle Database 19cの重要な推奨個別パッチ(ドキュメントID 555.1)
- My Oracle Supportノート2118136.2
- My Oracle Supportノート1227443.1

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

アップグレードおよび透過的データ暗号化

TDEを使用してデータベースをアップグレードするには、`-load_password`コマンドライン・オプションを使用するか、外部パスワード・ストアを指定して、AutoUpgradeにTDEパスワードを指定します。

AutoUpgradeバージョン22.1以降では、アップグレード中にコマンドラインでTransparent Data Encryption (TDE)パスワードを指定して、ソース・キーストアにアクセスし、選択した場所のターゲット・システム上の新しい外部キーストアをAutoUpgradeで作成することも、TDEパスワードを含む既存のセキュアな外部パスワード・ストア(SEPS)にAutoUpgradeがアクセスするように指定することもできます。

パスワード初期化およびストレージを使用したコマンドラインでのTDEパスワードの指定

アップグレードするデータベースのTDEパスワードがある場合は、コマンドラインでこれらのパスワードをAutoUpgradeに指定できます。AutoUpgradeでは、AutoUpgradeによって生成および保守される外部キー・マネージャを作成します。この構成では、AutoUpgradeによって、TDE対応データベースの無人操作または自動操作がサポートされます。アップグレードが実行される

と、AutoUpgradeは、キーストア・パスワードを要求せずに各ソース・データベース・キーストアをオープンし、ターゲット・データベースをTDE外部キーストアに登録してキー管理を行うことで、ターゲット・データベースを自動的に起動できます。

1. AutoUpgradeを実行する前に、TDEを使用するデータベースで使用する構成ファイルにグローバル・パラメータ`global.keystore`を追加し、アップグレードされたデータベース用に作成するキーストアの場所へのセキュアなパスを指定します。このパスは、キーストアがログ・ファイルの場所に存在しないように、AutoUpgradeで指定した他のファイル・パスと異なっている必要があります。
2. デプロイ・モードでAutoUpgradeを実行する場合は、`-config`コマンドライン・パラメータおよび`-load_password`パラメータを使用して実行する必要があります。
3. AutoUpgradeがデータベースのアップグレードを開始する前に、AutoUpgradeによって、TDEを使用する構成ファイルで指定された各データベースにTDEパスワードを指定するよう求められます。これらのパスワードは、ソース・リリースのTDEキーストアへのアクセス、および新しいターゲット外部キーストアへのTDEパスワードの書込みにのみ使用されます。パスワードは、アップグレード中にSQL*Plus実行計画に書き込まれません。AutoUpgradeでTDEパスワードが必要なくなった場合、これらのパスワードはメモリーから削除されます。パスワードのログ・レコードは保持されません。

パスワードの初期化およびストレージ・オプションの次のセキュリティ機能に注意してください。

- AutoUpgradeがデプロイを開始するときにTDEパスワードを指定します。これらは構成ファイルには含まれていません。
- AutoUpgradeでは、TDEキーストアを含む構成ファイルで指定された各ソース・データベースにTDEパスワードを指定するように求められます。
- AutoUpgradeでは、アップグレード中にAutoUpgradeによって書き込まれたファイルにパスワード・ロギングを実行しません。かわりに、AutoUpgradeは、デプロイ中に`load_password`コマンドライン・オプションが使用されたことを記録します。
- AutoUpgradeを実行すると、コマンドラインで入力されたTDEパスワードをセキュアなJava KeyStoreオブジェクトに配置します。
- Oracle DatabaseキーストアのTDEパスワードがアクセスに使用され、ターゲット・データベースがキー管理のためにTDE外部キーストアに登録されると、AutoUpgradeはパスワードを含むJava KeyStoreオブジェクトをクリアして、これらのパスワードがメモリーに存在しないようにします。
- AutoUpgradeでは、アップグレード中にAutoUpgradeによって生成されるzipファイルにキーストア・ファイルを含めません。
- アップグレードが非CDBまたは切断/接続PDBアップグレードの場合、AutoUpgradeで作成された、非CDBからPDBまたは切断/接続アップグレードを実行するデータベース用のXMLマニフェスト・ファイルには、TDE暗号化キーが含まれています。このファイルは、AutoUpgradeによって生成されたzipファイルでも除外されます。

AUTO LOGINキーストアを使用したTDEパスワードの指定

既存の外部キーストアを使用してTDEのパスワードをAutoUpgradeに指定する場合は、DBAの介入を必要とせずにデータベースを停止および再起動できるように、`AUTO LOGIN`キーストアを1回かぎり設定する必要があります。

既存のものにOracle Wallet値が指定されていることを確認するには、次のコマンドを入力します。

```
SQL> show parameter WALLET_ROOT;
```

AutoUpgrade 22.1以降では、`sqlnet.ora file`を新しいOracleリリースにコピーする必要がなくなりました。セキュアな外部パスワード・キーストアを使用する場合は、`WALLET_ROOT`静的初期化パラメータおよび`TDE_CONFIGURATION`動的初期化パラメータを使用することをお薦めします。

AutoUpgrade構成ファイルで、アップグレード中にキーストアの場所を変更するように指定できます。または、アップグレード後にこのタスクを手動で完了することもできます。いずれの場合も、アップグレードを開始する前に、キーストアから最新のバックアッ

プが作成されていることを確認してください。

このタスクを手動で完了する場合は、アップグレード後、キーストアを構成しデータの暗号化を開始する前に、作成する予定のキーストアの場所とタイプを指定するために、WALLET_ROOTパラメータとTDE_CONFIGURATIONパラメータを使用して1回かぎりの構成を実行する必要があります。

WALLET_ROOTパラメータは、キーストア・ディレクトリの場所を指定します。WALLET_ROOTを設定する前に、キーストアを格納するために使用できる既存のディレクトリがあることを確認してください。(通常、このディレクトリはwalletと呼ばれます。)

TDE_CONFIGURATIONパラメータは、キーストアのタイプ(ソフトウェア・キーストア、ハードウェア・キーストアまたはOracle Key Vaultキーストア)を指定します。TDE_CONFIGURATIONパラメータを省略した場合、Oracle Databaseはsqlnet.oraファイル設定を使用します。TDE_CONFIGURATIONを使用してキーストアのタイプを設定した後で、キーストアを作成すると、Oracle Databaseはキーストア・タイプのWALLET_ROOTの場所内にディレクトリを作成します。たとえば、透過的データ暗号化キーストアのためにTDE_CONFIGURATIONをFILEに設定した場合は、Oracle Databaseによってウォレット・ディレクトリ内にtde (小文字)というディレクトリが作成されます。キーストア・タイプ間で移行する場合は、最初にTDE_CONFIGURATIONパラメータに使用するキーストア・タイプを設定し、ADMINISTER KEY MANAGEMENT文を使用して移行を実行する必要があります。たとえば、ハードウェア・セキュリティ・モジュール(HSM)キーストアからTDEキーストアに移行できます。

V$ENCRYPTION_WALLET動的ビューのKEYSTORE_MODE列は、統一モードまたは分離モードが構成されているかどうかを示します。

> ノート:
>
> 以前のリリースでは、キーストア・ディレクトリの場所を定義するためにSQLNET.ENCRYPTION_WALLET_LOCATION パラメータが使用されていました。このパラメータは現在非推奨になっています。かわりに、WALLET_ROOT 静的初期化パラメータおよび TDE_CONFIGURATION 動的初期化パラメータを使用することをお薦めします。AutoUpgrade ユーティリティを使用して、アップグレード中にこの更新を自動的に実行できます。

関連項目

- キーストアおよびマスター暗号化キーの管理
- AutoUpgradeを使用したアップグレード時のTDEウォレット・ストアの場所の更新

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

リスナーが新しいリリースのOracleホームで実行されていることを確認する必要があります。

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、アップグレードを開始する前に、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。現在のリスナーには、以前のOracle Databaseとの下位互換性があります。

Oracle Real Application Clusters Oracle DatabaseまたはOracle Database 12cより古いリリースをアップグレードする場合は、次の追加情報を確認してください。

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。

Oracle Database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`

これらのパラメータは、Oracle Database 12cで導入されたもので、以前のリリースではサポートされていません。

関連項目

- 『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

ORAPWDを使用したパスワード・ファイルの作成または移行

REMOTE_LOGIN_PASSWORDFILEが設定されている場合に参照してください。

REMOTE_LOGIN_PASSWORDFILE初期化パラメータが`EXCLUSIVE`に設定されている場合、`ORAPWD`を使用してパスワード・ファイルを作成または移行します。Oracle Database 12c以上のリリースでは、既存のデータベースからパスワード・ファイルを移行するために、`ORAPWD`に新しいオプションが提供されています。

Oracle Database 12cリリース2 (12.2)以上では、REMOTE_LOGIN_PASSWORDFILEが`SHARED`に設定されている場合、アップグレード前チェックの検証で警告が出ます。次のいずれかのオプションを選択して、この問題を修正します。

- `REMOTE_LOGIN_PASSWORDFILE = NONE`を設定して、パスワード・ファイル・ベースの認証を完全に無効にします
- `REMOTE_LOGIN_PASSWORDFILE = EXCLUSIVE`を設定して、パスワード・ファイル・ベースの認証を制限します

参照:

パスワード・ファイルの作成および移行の詳細は、『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

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)以降のリリースでは、排他モード認証プロトコルが使用されます。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12C)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以降のリリースにアップグレードした後は、大/小文字を区別しない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。このように変更されたのは、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。クライアントを認証するパスワード・バージョンなしで、サーバーはそのまま構成されます。

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薦めします。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薦めします。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12C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。たとえば: SQLNET.ALLOWED_LOGON_VERSION_SERVER=11

関連項目

- 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2-1 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
```

追加の操作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すして、)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2-2 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。たとえば:

```
SQLNET.ALLOWED_LOGON_VERSION_SERVER=11
```

この変更を行った後で、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。システムは許容モードで実行中のため、10Gパスワード・バージョンは引き続き存在します。
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O5L_NP機能フラグがあることを確認します。

> ノート:
>
> 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 O5L_NP 機能があります。他のクライアントでは CPUOct2012 パッチを適用して O5L_NP 機能を取得する必要があります。
>
> O5L_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O5L_NP機能が導入されたら、次のプロシージャを使用して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。たとえば:

   ```
   SEC_CASE_SENSITIVE_LOGON = TRUE
   ```

   b. SQLNET.ALLOWED_LOGON_VERSION_SERVERパラメータをサーバーのSQLNET.ORAファイルから削除するか、サーバーのSQLNET.ORAファイルのSQLNET.ALLOWED_LOGON_VERSION_SERVERの値を12に戻して、排他モードに設定し直します。たとえば:

   ```
   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2-3 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.
```

ノート:

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。

関連項目

- 『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

STIGおよびCISプロファイルのリソースおよびパスワード・パラメータの更新

Oracle Database 21c以降、アップグレードでは、既存のSTIGプロファイルの更新、およびアップグレードの一環としてのCISプロファイルのインストールを含むOracle推奨プロファイルが構成されます。

プロファイルとは、ユーザーに適用される属性の集合です。それらの属性を共有する複数ユーザーの中の任意のユーザーに関する単一の参照になります。

Oracle Databaseのアップグレード中に、米国国防総省(DISA)セキュリティ技術導入ガイド(STIG)のベースラインで指定されている最新のシステム構成ベースラインに従って、Oracle提供プロファイルORA_STIG_PROFILEユーザー・プロファイルが更新されます。この更新により、ORA_STIG_PROFILEユーザー・プロファイルで以前に設定した可能性のあるパスワードおよびリソース制限が上書きされます。また、新しいプロファイルORA_CIS_PROFILEが追加され、ソフトウェア・リリース時にOracleで使用可能な最新のCenter of Internet Security (CIS)ベースライン更新に準拠します。これら2つのプロファイルは、**Oracle推奨プロファイル**に指定されています。これらのプロファイルはSTIGおよびCISベースラインに基づいているため、標準のDEFAULTプロファイルとは異なります。

プロファイルORA_STIG_PROFILEおよびORA_CIS_PROFILEはLOCALプロファイルとして作成され、CONTAINER=CURRENT句が使用されます。ただし、Oracleが提供するプロファイルのセキュリティを強化するために、SYSユーザーのみがこれらのファイルを変更する権限を持っています。

ORA_STIG_PROFILEに関連付けられているユーザーがいる場合、これらのユーザーの次のパラメータはアップグレード後に厳密になります。

- PASSWORD_LIFE_TIME。35に変更されます。
- PASSWORD_REUSE_TIME。175に変更されます。
- PASSWORD_GRACE_TIME。0に変更されます。

Oracle推奨プロファイルの使用の詳細は、*Oracle Database*セキュリティ・ガイドを参照してください。

関連項目

- プロファイルによるリソースの管理

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

プロファイル・スクリプト(glogin.sqlおよびlogin.sql)の確認

すべてのアップグレード方法で、プロファイル・スクリプトを使用せずにアップグレードを実行することをお薦めします。

プロファイル・スクリプト(glogin.sqlおよびlogin.sql)の内容によっては、これらのスクリプトがOracle Databaseのアップグレードを妨げるリスクがあり、「UPG-1400 アップグレードに失敗しました」エラー、「catconで予期しないエラーが発生しました」または「ORA-04023: オブジェクトSYS.STANDARDを検証または認可できませんでした」が発生する可能性があります。アップグレードを開始する前に、ターゲットのOracleホームから(Oracleホームの/sqlplus/adminにある)サイト・プロファイル・スクリプト(glogin.sql)を削除することをお薦めします。また、現在のディレクトリまたは環境変数SQLPATHを使用して指定されたディレクトリに、ユーザー・プロファイル・スクリプトが定義されていないことを確認します。

関連項目

- アップグレードおよびプロファイル・スクリプト

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パラレル・アップグレード・ユーティリティを使用します。

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/cfgtoologs/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` (`pdbname`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.
```

参照:

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の高可用性オプション

Standard Edition High Availability、Oracle Restart、Oracle Real Application Clusters (Oracle RAC)およびOracle RAC One Nodeを使用して、Oracle Databaseで使用可能な高可用性オプションを確認します。

Oracle Databaseで使用可能な高可用性オプションの概要は次のとおりです。

Standard Edition High Availability

- クラスタベースのアクティブ/パッシブOracle Databaseフェイルオーバー・ソリューション
- シングル・インスタンスのStandard Edition Oracle Database用に設計されています
- Oracle Database 19cリリース更新(RU) 19.7以降で使用できます
- Oracle Grid Infrastructure 19c RU 19.7以降をスタンドアロン・クラスタとしてインストールする必要があります

Oracle Restart

- Oracle Databaseインスタンスの再起動のみの機能およびスタンドアロン・サーバー・デプロイメント用のOracle Automatic Storage Management (Oracle ASM)の基礎
- シングル・インスタンスのOracle Database用

- スタンドアロン・サーバー(クラスタなし)用のOracle Grid Infrastructureが必要です

Oracle Real Application Clusters (Oracle RAC) One Node

- クラスタベースのアクティブ/パッシブOracle Databaseフェイルオーバーおよびオンライン・データベース再配置ソリューションを提供します
- Oracle RAC対応のOracle Databaseで使用できます
- 通常の操作では、Oracle RAC対応のOracle Databaseの1つのインスタンスのみが実行されています
- オンライン方式で、クラスタ内の別のサーバーにアクティブなインスタンスを再配置できます。アクティブなインスタンスを再配置するために、宛先サーバー上の2番目のインスタンスを一時的に起動して、ワークロードを再配置できます
- ローリング・アップグレード(パッチ・セット、データベースおよびオペレーティング・システム)をサポートします
- アプリケーション・コンティニュイティをサポートします
- Oracle Grid Infrastructureをスタンドアロン・クラスタとしてインストールする必要があります

Oracle Real Application Clusters(Oracle RAC)

- アクティブ/アクティブOracle Database高可用性およびスケーラビリティのソリューションを提供します
- 複数のサーバーが同じOracle Databaseで同時トランザクションを実行できます
- 高可用性を実現します。他のインスタンスやサーバーが稼働したままであるため、データベース・インスタンスまたはサーバーの障害によってデータベース・サービス全体が中断されることはありません
- ローリング・アップグレード(パッチ・セット、データベースおよびオペレーティング・システム)をサポートします
- アプリケーション・コンティニュイティをサポートします
- Oracle Grid Infrastructureをスタンドアロン・クラスタとしてインストールする必要があります

- 監査表の事前アップグレードおよびアーカイブの要件
  Oracle Label SecurityおよびOracle Database Vaultを使用している12.1より前のOracle Databaseリリースでは、アップグレードの前にOLS事前処理スクリプトを実行する必要があります。

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

監査表の事前アップグレードおよびアーカイブの要件

Oracle Label SecurityおよびOracle Database Vaultを使用している12.1より前のOracle Databaseリリースでは、アップグレードの前にOLS事前処理スクリプトを実行する必要があります。

Oracle Label Security (OLS)およびOracle Database Vaultを使用するOracle Databaseリリース12.1より前のデータベースをアップグレードする場合は、最初にOLS事前処理スクリプト(`olspreupgrade.sql`)を実行して`aud$`表のコンテンツを処理する必要があります。OLSアップグレードでは、`aud$`表が、SYSTEMスキーマからSYSスキーマに移動されます。`olspreupgrade.sql`スクリプトは、この移動に必要な事前処理スクリプトです。

> 注意:
>
> Oracle Label Security および Oracle Database Vault を使用する Oracle Database リリース 12.1 より前のデータベースをアップグレードする前には、`olspreupgrade.sql` スクリプトを実行する必要があります。Oracle Database リリース 12.1 にアップグレード後に、データベースのパッチまたはアップグレードをするための OLS 事前処理の手順を実行する必要はありません。

`olspreupgrade.sql`スクリプトでは、SYSスキーマに一時表PREUPG_AUD$が作成され、`SYSTEM.aud$`レコードが

SYS.PREUPG_AUD$に移動されます。安全対策として、olspreupgrade.sqlスクリプトを実行する前に監査証跡をアーカイブすることをお薦めします。Oracle Label Securityがデータベースにインストールされていて、以前のリリースからアップグレードする場合は、アップグレードの前にOLS事前処理スクリプトを実行する必要があります。

参照:

監査証跡のアーカイブの詳細は、『Oracle Databaseセキュリティ・ガイド』を参照してください

OLS事前処理スクリプトの詳細は、Oracle Label Security管理者ガイドを参照してください。

親トピック: Oracle Databaseの高可用性オプション

Oracle Database Standard Editionでの高可用性のオプション

Oracle Database 19c以降のリリースでOracle Database Standard Editionの高可用性を可能にするために、Standard Edition High Availabilityの使用方法について学習します。

- Standard Edition Oracle RACまたはOracle RAC One Nodeのアップグレードの準備
  Standard Edition Oracle Real Application Clusters (Oracle RAC)からの移行後に高可用性を維持するために、Standard Edition High Availabilityを使用できます。
- Oracle DatabaseでStandard Edition High Availabilityを使用するための要件
  Standard Edition High Availabilityを使用するには、次の構成要件に従ってOracle Database Standard Edition 2をデプロイします。

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

Standard Edition Oracle RACまたはOracle RAC One Nodeのアップグレードの準備

Standard Edition Oracle Real Application Clusters (Oracle RAC)からの移行後に高可用性を維持するために、Standard Edition High Availabilityを使用できます。

Oracle Database 19cのリリース以降、Oracle Database Standard Edition 2ではOracle RACはサポートされません。Oracle Database Standard Editionの高可用性のニーズを引き続き満たすために、Standard Edition High Availabilityが導入されています。

親トピック: Oracle Database Standard Editionでの高可用性のオプション

Oracle DatabaseでStandard Edition High Availabilityを使用するための要件

Standard Edition High Availabilityを使用するには、次の構成要件に従ってOracle Database Standard Edition 2をデプロイします。

- データベースは、スタンドアロン・クラスタ用のOracle Grid Infrastructureを実行するクラスタで作成され、そのデータベース・ファイルはOracle Automatic Storage Management (Oracle ASM)またはOracle Automatic Storage Management Cluster File System (Oracle ACFS)に配置されます。
- Database Configuration Assistantを使用する場合は、Standard Edition High Availability用に構成するOracle Database Standard Edition 2データベースの作成時にリスナーを作成しないでください。
- 単一クライアント・アクセス名(SCAN)リスナーに、データベースをリモート・リスナーとして、ノード・リスナーをローカル・リス

ナーとして登録します。

- データベース・サービスを作成します。アプリケーションまたはデータベース・クライアントをデータベースに接続する際は、デフォルトのデータベース・サービスではなくこのサービスを使用します。
- サーバー・パラメータ・ファイル(spfile)およびパスワード・ファイルがOracle ASMまたはOracle ACFSにあることを確認します。データベースの作成または構成時にspfileおよびパスワード・ファイルがローカル・ファイル・システムに配置された場合は、それらのファイルをOracle ASMまたはOracle ACFSに移動します。

満たす必要がある追加の要件は、データベースのインストールに関するドキュメントを参照してください。

関連項目

- Oracle Databaseインストレーション・ガイドfor Linux

親トピック: Oracle Database Standard Editionでの高可用性のオプション

統合監査証跡へのオペレーティング・システムの監査レコードの移動

スピルオーバー監査ファイルに書き込まれた監査レコードは、統合監査証跡データベース表に移動できます。

データベースが書込み可能でない場合(データベースのマウント中など)、データベースがクローズされている場合、または読取り専用の場合は、Oracle Databaseによって、監査レコードがこれらの外部ファイルに書き込まれます。これらの外部ファイルのデフォルトの場所は、$ORACLE_BASE/audit/$ORACLE_SIDディレクトリです。

DBMS_AUDIT_MGMT.LOAD_UNIFIED_AUDIT_FILESプロシージャを実行して、データベースにファイルをロードできます。多くのオペレーティング・システム監査レコードを外部ファイルに移動している場合は、パフォーマンスが影響を受ける場合があるので注意してください。

データベースが書込み可能な場合に、これらのファイルの監査レコードをAUDSYSスキーマ監査表に移動するには:

1. AUDIT_ADMINロールを割り当てられているユーザーとして、CDBルートにログインします。
   現在のリリースまたはOracle Databaseにアップグレードする前に、アップグレード・プロセス中にオペレーティング・システムの過剰ファイルが失われないようにするために、CDBルートから
   DBMS_AUDIT_MGMT.LOAD_UNIFIED_AUDIT_FILESプロシージャを実行する必要があります。
   たとえば:

```
CONNECT c##aud_admin
Enter password: password
Connected.
```

2. データベースがオープンで、書込み可能であることを確認します。

   データベースがオープンされており書込み可能かどうかを確認するには、V$DATABASEビューを問い合せます。

```
SELECT NAME, OPEN_MODE FROM V$DATABASE;
NAME            OPEN_MODE
--------------- ----------
HRPDB           READ WRITE
```

   show pdbsコマンドを実行して、現在のインスタンスに関連付けられているPDBについて情報を見つけることができます。

3. DBMS_AUDIT_MGMT.LOAD_UNIFIED_AUDIT_FILESプロシージャを実行します。

```
EXEC DBMS_AUDIT_MGMT.LOAD_UNIFIED_AUDIT_FILES;
```

4. 個々のPDB監査レコードをロードする場合は、各PDBにログインして

DBMS_AUDIT_MGMT.LOAD_UNIFIED_AUDIT_FILESプロシージャを再実行します。

監査レコードはAUDSYSスキーマ監査表に即座にロードされ、$ORACLE_BASE/audit/$ORACLE_SIDディレクトリから削除されます。

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

非CDBのアップグレードとOracle GoldenGate

Oracle GoldenGateがデプロイされている非CDB Oracle Databaseをアップグレードする場合は、Oracle GoldenGateを停止し、マルチテナント・アーキテクチャ用に変換およびアップグレードした後で再構成する必要があります。

アップグレードする非CDB Oracle DatabaseでOracle GoldenGateを使用している場合、ソースの非CDB Oracle Databaseをマルチテナント・アーキテクチャに変換およびアップグレードする前に、Oracle GoldenGateプロセスを停止して削除し、マルチテナント・アーキテクチャ用に変換およびアップグレードした後で再構成する必要があります。必要なプロセスの概要は次のとおりです。

1. ソースOracle DatabaseでOracle GoldenGateユーザーを削除します。
2. Oracle GoldenGateプロセスがOracle GoldenGate証跡内のすべての現行DMLデータおよびDDLデータの処理を終了し、プロセスがエンド・オブ・ファイル(EOF)になるまで待機します。
3. ソース・データベースですべてのOracle GoldenGateプロセスを停止します。
4. ソース非CDB Oracle Databaseからターゲット・リリースのCDB上のターゲットOracle Databaseへの変換およびアップグレードを完了します。
5. データベースを再起動します。
6. また、データベースを以前のリリースから新しいメジャー・リリース・ファミリ(たとえば、Oracle Database 12.1からOracle Database 19c (Oracle Database 12.2ファミリのターミナル・パッチ・セット))にアップグレードする場合は、Oracle Database 19cでサポートされている新しいバージョンのOracle GoldenGateをインストールする必要があります。Oracle DatabaseとOracle GoldenGateの両方を同時にアップグレードする場合は、まずデータベースをアップグレードする必要があります。

データベースの変換およびアップグレードが完了した後に、Oracle GoldenGate Extractユーザー用の新しい資格証明を作成できます。新しい資格証明を使用して、ターゲットCDBのアップグレード済Oracle Database PDB用に新しいExtractプロセスとExtractポンプおよび配布サービスを作成し、新しく作成したプロセスを起動できます。アップグレード後にこれらの手順を完了する方法の詳細は、Oracle GoldenGateのドキュメントを参照してください。

関連項目

- Oracle GoldenGate資格証明の確立
- マルチテナント・コンテナ・データベースでのOracle GoldenGateの構成

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

AutoUpgradeを使用する前の大規模データベースのバックアップ

大規模なデータベースで部分オフライン・バックアップを使用する場合は、データベースをダウングレードする必要がある場合に停止時間を最小限に抑えるために、表領域をチェックし、リカバリに必要なすべての表領域がバックアップされていることを確認します。

バックアップ・オプションとして部分オフライン・バックアップを選択したデータベースのアップグレードにAutoUpgradeユーティリティを使用している場合は、アップグレードに必要なすべての表領域がREAD WRITEモードであることを確認し、バックアップに必要な

すべての表領域を特定した後にのみ、AutoUpgradeを実行する前に、リカバリに必要な表領域のOFFLINEバックアップを取得する前に、必要なすべての表のステータスを変更します。

このガイドラインの理由は、次のとおりです。

AutoUpgradeの操作中に、SYSTEM、SYSAUXおよびUNDO以外の他の表領域をアップグレードのためにREAD WRITEステータスで保持する必要がある場合があります。アップグレード中にこの要件が発生する理由には、次のものがあります。

- ディクショナリ・オブジェクトを含む表領域
- Oracle管理ユーザーのデフォルト表領域である表領域
- データベースのデフォルト表領域である表領域

AutoUpgradeは、アップグレードに必要なすべての表領域がREAD WRITEステータスであるかどうかを検出します。

アップグレードのためにREAD WRITEモードに変更する必要がある表領域がある場合は、次のようにします。

- PRECHECKS処理モードでは、AutoUpgradeはREAD ONLYステータスの表を問題として検出します。
- FIXUP処理モードでは、AutoUpgradeは自動修正を実行して、READ WRITEモードである必要があるREAD ONLYモードで検出された表領域をREAD WRITEモードに更新します。

AutoUpgradeがREAD ONLYモードからREAD WRITEモードに変更するアップグレードに必要な表領域があり、これらの表領域がAutoUpgradeの起動前にバックアップに含まれていなかった場合、リカバリ計画にリスクがあります。バックアップがリカバリに有効であることを確認するには、バックアップが必要な表領域を確認した後にのみ、OFFLINEバックアップを作成する必要があります。

部分オフライン・バックアップに、アップグレード中に変更されたすべての表領域のバックアップが含まれていることを確認するには、次の手順を実行します。

1. SYSTEM、SYSUXおよびUNDO以外のすべての表領域と、READ WRITEに存在する必要があることがわかっている表領域をREAD ONLYモードにします。
2. ピボット・ユーザーに対して次の問合せを実行します。

```
(SELECT username
FROM dba_users
WHERE user_id in (
  SELECT schema# FROM sys.registry$
    WHERE namespace = 'SERVER'
    UNION
    SELECT schema# FROM sys.registry$schemas
    WHERE namespace = 'SERVER'
    UNION
    SELECT user# FROM sys.user$
    WHERE type#=1 AND bitand(spare1,256)=256))
  SELECT tablespace_name
  FROM dba_tablespaces
  WHERE status <>'ONLINE' and tablespace_name IN
  (
  SELECT property_value
  FROM database_properties
    WHERE property_name = 'DEFAULT_PERMANENT_TABLESPACE'
    UNION
    SELECT default_tablespace
    FROM dba_users
    WHERE username IN (SELECT username FROM pivot_users)
    UNION
    SELECT tablespace_name
    FROM dba_segments
    WHERE owner IN (SELECT username FROM pivot_users)
    UNION
    SELECT t.name
```

```
    FROM modeltab$ m,  ts$ t, sys_objects s
    WHERE m.obj#=s.object_id and s.ts_number=t.ts#
    )'
```

次に実行するステップは、問合せの結果によって異なります。

- 問合せで行が戻されない場合は、SYSTEM、SYSAUXおよびUNDOと、特にREAD WRITEに存在することがわかっている表のバックアップで、部分オフライン・バックアップを完了するのに十分であることを意味します。
- 問合せで表領域の行が戻される場合、部分オフライン・バックアップを完了するには、これらの追加の表領域をREAD WRITEモードにする必要があります。

3. 必要なすべての表領域の識別およびREAD WRITEモードへの設定が完了したら、それらの表領域の部分オフライン・バックアップを取得します。また、SYSTEM、SYSAUXおよびUNDO.redoログ、制御ファイル、および必要に応じてリストア/リカバリ手順に関連するその他のファイルもバックアップします。
4. ANALYZEモードでAutoUpgradeを実行します。出力を確認し、READ WRITEに配置する必要があるREAD ONLYとしてレポートされた追加の表領域がAutoUpgradeによって識別されないことを確認します。

(オプション)この手順の結果をここで入力します。

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 Data Guardを使用したデータベースのローリング・アップグレードの準備

Oracle Data Guardを使用してローリング・アップグレードを実行する場合、アップグレードを開始する前に、Data Guard Broker構成ファイルを移動する必要があります。

DB_BROKER_CONFIGファイルのデフォルトの場所は、以前のリリースのOracle Database Oracleホームのdbsディレクトリ内です。Oracle Data Guardを使用してデータベース・インスタンスのローリング・アップグレードを実行する場合、DG_BROKER_CONFIGファイルを前のリリースのOracleホーム外のマウント・ポイント場所に移動する必要があります。また、DG_BROKER_CONFIG_FILEnパラメータでは、必ずその場所を以前のリリースのOracleホーム内の場所のかわりに指定してください。データベースのアップグレード中にリスナーを移行しないでください。アップグレードが完了した後、リスナーを停止し、データベースを停止し、以前のソースのOracle Databaseリリース環境から新しいOracle Databaseリリース環境にlistener.oraおよびtnsnames.oraをコピーして、リスナーおよびデータベースを起動します。

アップグレード開始前の作業

ローリング・アップグレード中にDB_BROKER_CONFIGファイルにアクセスできるようにするには、アップグレードを開始する前に、次の作業を完了する必要があります

1. アップグレードを開始する前に、ストレージにOracle Automatic Storage Management (Oracle ASM)を使用していない場合は、Oracle Data GuardファイルDG_BROKER_CONFIG_FILE1およびDG_BROKER_CONFIG_FILE2を、ソースまたはターゲットのOracle Database OracleホームのOracleホーム・パス以外の場所にあるサーバー上の別のマウント・ポイントに設定します。

> ノート:
>
> Oracle Database 21c より前のデフォルトの ORACLE_HOME レイアウトでは、ORACLE_HOME、ORACLE_BASE_HOME および ORACLE_BASE_CONFIG が 1 つの場所にまとめられていました。Oracle Database 21c 以降、使用可能な構成は読取り専用の ORACLE_HOME であり、ORACLE_BASE_HOME と ORACLE_BASE_CONFIG は ORACLE_HOME とは別に配置されています。以前にフォルダ dbs にあった Oracle Data Guard ファイルなどのファイルは、現在は ORACLE_BASE_CONFIG/dbs にあります。

2. 以前のリリースのOracleホームから新しいOracle Databaseリリースへのアップグレードを正常に完了します。

アップグレード中の作業

アップグレード中に、リスナーを移行しないでください。

AutoUpgradeを使用してアップグレードを完了することをお薦めします。次を参照してください。

AutoUpgradeとOracle Data Guard

アップグレード完了後の作業

1. 新しいリリースのOracle Databaseのリスナーを停止します。
2. 新しいリリースのOracle Databaseを停止します。
3. 以前のリリースのOracle Databaseから新しいリリースのOracle Databaseにlistener.oraおよび

`tnsnames.ora`ファイルをコピーします。

4. リスナーと新しいリリースのOracle Databaseを起動します

Data Guard Broker構成ファイルの移動の詳細は、『*Oracle Data Guard Broker*』を参照してください。

関連項目

- ブローカ構成ファイルの名前の変更

親トピック: Oracle Databaseのアップグレードの準備

アップグレードのための新しいOracleホームの準備

新しい場所で新しいOracleホームを準備するには、構成ファイルを移動する必要があるかどうか、またはその他の作業を実行する必要があるかどうかを確認します。

アップグレードするデータベースをバックアップした後、読取り専用Oracleホームを使用していない場合は、新しい場所に新しいOracleホームを準備し、新しいOracle Databaseリリース用のソフトウェアを新しい場所にインストールします。

1. (手動アップグレードのみ)アップグレード対象のデータベースのOracleホームから新しいリリースのOracle Database Oracleホーム(ターゲット)に構成ファイルをコピーします。AutoUpgradeユーティリティ、DBUAまたは読取り専用Oracleホームを使用している場合は、構成ファイルが自動的にコピーされるため、このステップを無視できます。

   構成ファイルを新しいOracleホームにコピーするには、次の手順を使用します。

   a. パラメータ・ファイルが古い環境のOracleホームに存在する場合は、新しいOracleホームへコピーします。デフォルトでは、パラメータ・ファイルを検索する場所は、LinuxまたはUnixプラットフォームの場合は`ORACLE_HOME/dbs`ディレクトリで、Windowsオペレーティング・システムの場合は`ORACLE_HOME¥database`ディレクトリです。アップグレードが終了したら、パラメータ・ファイルは、古い環境のOracleホームを除き、どこにあってもかまいません。

   > ノート:
   >
   > 必要に応じて、サーバー・パラメータ・ファイル(`SPFILE`)からテキスト初期化パラメータ・ファイル(`PFILE`)を作成して、初期化パラメータを編集できるようにしてください。

   b. Oracle ASMインスタンス内にパラメータ・ファイルがある場合は、次のコマンドを使用してパラメータ・ファイルをバックアップします。

   ```
   CREATE pfile FROM spfile;
   ```

   次のコマンドを使用してパラメータ・ファイルを作成することもできます。ここで、`/path/to/pfile/`は新しいOracleホームへのパス、`pfile_name`はパラメータ・ファイルの名前です。

   ```
   create pfile[='/path/to/pfile/pfile_name.ora/
   ```

   `SPFILE`がOracle ASMにある環境のデータベースをダウングレードする必要がある場合は、ダウングレードする前にパラメータ・ファイルをリストアする必要があります。

   c. パラメータ・ファイルが`IFILE` (インクルード・ファイル)エントリまたは`SPFILE` (サーバー・パラメータ・ファイル)エントリのいずれかを含むテキストベースの初期化パラメータ・ファイルであり、`IFILE`または`SPFILE`エントリ内に指定されたファイルが以前のリリースの環境のOracleホームに存在する場合、`IFILE`または`SPFILE`エントリで指定されているファイルを新しいOracleホームへコピーします。`IFILE`エントリまたは`SPFILE`エントリ内に指定されたファイルには、追加の初期化パラメータがあります。

   d. 古い環境のOracleホームに格納されたパスワード・ファイルがある場合は、そのパスワード・ファイルを新しいOracleホームに移動またはコピーします。

   パスワード・ファイルの名前と位置は、オペレーティング・システムによって異なります。パスワード・ファイルは次の場所にあります。ここで、`SID`はOracleインスタンスのIDです。

   - LinuxまたはUnixプラットフォーム: デフォルトのパスワード・ファイルは`orapw SID`です。ディレクトリ

ORACLE_HOME/dbsにあります。

- Microsoft Windowsオペレーティング・システム: デフォルトのパスワード・ファイルはpwdSID.oraです。ディレクトリORACLE_HOME¥databaseにあります。

2. 次のステップを実行して、新しいOracle Databaseリリースのパラメータ・ファイルを調整します。

   a. サポートが終了した初期化パラメータを削除して、非推奨になった初期化パラメータを調整します。新しいリリースには、サポートが終了したパラメータや非推奨となったパラメータがあります。新しいOracle Databaseインスタンスの起動に使用されるパラメータ・ファイルから、サポートが終了したパラメータをすべて削除します。サポートが終了したパラメータが原因で、新しいOracle Databaseリリースでエラーが発生する可能性があります。また、新しいリリースで構文が変更になったパラメータも変更します。

      analyzeモードで-preupgradeパラメータを指定してAutoUpgradeを実行すると、生成されたupgrade.xmlファイルで検出された非推奨のパラメータおよびサポートが終了したパラメータが表示されます。

      初期化パラメータの値は、upgrade.xmlに示されている最小値以上に調整します。

      パラメータ・ファイルのすべてのパス名が完全に指定されていることを確認してください。パラメータ・ファイルには相対パス名を使用しないでください。

   b. パラメータ・ファイルにIFILEエントリが含まれている場合は、パラメータ・ファイル内のIFILEエントリを変更します。IFILEエントリは、ステップ1で指定した新しい場所のテキスト初期化パラメータ・ファイルを指している必要があります。また、ステップ1でパラメータ・ファイルを編集したときと同じ方法で、IFILEエントリに指定されているファイルを編集します。

   c. クラスタ・データベースをアップグレードする場合は、必要に応じて、SPFILEまたはinitORACLE_SID.oraファイルを変更できます。

   これらのパラメータ・ファイルを調整した後で、変更したすべてのファイルを必ず保存してください。

3. (手動アップグレードのみ) AutoUpgradeまたはリプレイ・アップグレードを使用せずにクラスタ・データベースをアップグレードする場合は、データベース・インスタンスをクラスタから手動で切り離す必要があります。CLUSTER_DATABASE初期化パラメータをfalseに設定します。アップグレード後に、この初期化パラメータの設定をtrueに戻す必要があります。DBUAを使用している場合は、アシスタントによってこの作業が自動的に行われます。

親トピック: Oracle Databaseのアップグレードの準備

Windows上のOracleホームを準備するための前提条件

Microsoft WindowsプラットフォームでOracle Databaseをアップグレードする前に、現在のシステムがこれらの要件を満たしている必要があります。

セキュリティ上の理由により、他のOracleホームのOracleホーム・ユーザーとして構成された異なるMicrosoft Windowsユーザー・アカウントでは、同一のOracleベースを共有することはできません。

- データベース・アップグレードは、ソースと宛先Oracleホームの両方で同じWindowsユーザー・アカウントがOracleホームのユーザーとして使用された場合にサポートされます。
- データベース・アップグレードは、アップグレード対象のデータベースのOracleホームでWindows組込みアカウントを使用する場合にサポートされます。Windows上のOracleホームのユーザーの組込みアカウント・オプションは、Oracle Database 12cより前のリリース(リリース11.2以下)でのみサポートされていました。
- Oracleホームのユーザーは、自身のOracleベースおよびOracleホームの外部にあるファイルにはアクセスできない可能性があります。その場合、アップグレード中に別のOracleベースを選択すると、Oracle Databaseサービスが古いOracleベースのファイルにアクセスできなくなる可能性があります。データベースのアップグレードにDBUAを使用すると、Oracleホームのユーザーは、自身のOracleベースおよびOracleホームの外部にあるファイルにアクセスできます。

  手動でのアップグレードまたは古いOracleベースのカスタム・ファイル(ウォレット、構成ファイルおよびその他カスタム・ファイルなど)を使用する前に、これらの外部ファイルに対してOracleホームのユーザーにアクセス権を付与するか、新しいOracleベースにこれらのファイルをコピーする必要があります。

参照:

Windows上のデータベース管理の詳細は、Oracle Databaseプラットフォーム・ガイドfor Microsoft Windowsを参照してください。

親トピック: Oracle Databaseのアップグレードの準備

AutoUpgradeを使用したアップグレード前チェックの実行

AutoUpgradeユーティリティは、Oracleによって提供されるJava JARファイルで、アップグレードが正常に完了したことを確認するために役立ちます。

- AutoUpgradeユーティリティのシステム・チェックについて
  アップグレードが成功したことを確認するために、分析モードでAutoUpgradeユーティリティを使用してシステムを確認することをお薦めします。
- 分析モードを使用したAutoUpgradeの事前チェックの実行の例
  アップグレード前にAutoUpgradeを使用して非CDBのOracleデータベースをチェックする方法を確認するには、この例を使用して手順を理解してください。
- アップグレード・チェックの概要ファイルの確認
  AutoUpgradeアップグレード・チェックの概要ファイルを使用してアップグレードの準備をする方法を学習します。
- AutoUpgradeの事前チェックをCDBで実行するための構成ファイルの作成
  AutoUpgradeがアップグレードの事前チェックを実行して、マルチテナント・アーキテクチャのOracleデータベースにPDBを含めたり除外する方法をどのようにして制御できるかを確認します。
- 以前のリリースのOracleデータベースでのAutoUpgrade修正の実行
  この例を使用して、システムに分析モードで生成されるAutoUpgrade修正の実行方法を確認します。

親トピック: Oracle Databaseのアップグレードの準備

AutoUpgradeユーティリティのシステム・チェックについて

アップグレードが成功したことを確認するために、分析モードでAutoUpgradeユーティリティを使用してシステムを確認することをお薦めします。

アップグレードにAutoUpgradeユーティリティを使用するには、最初に分析モードでAutoUpgradeを実行する必要があります。

> ノート:
>
> AutoUpgrade ユーティリティには Java 8 が必要です。正しい Java バージョンが使用可能であることを確認するには、ターゲットの Oracle データベースの Oracle ホームにある Java バージョンを使用して、AutoUpgrade を実行することをお薦めします。Oracle ホームの Java バージョンへのパスは、`Oracle-home/jdk/bin/java` で、`Oracle-home` はターゲットの Oracle データベースの Oracle ホームです。Oracle ホームに Java 8 があることを確認するには、次のコマンドを実行します。
>
> ```
> $ORACLE_HOME/jdk/bin/java -version
> java version "1.8.0_211"
> Java(TM) SE Runtime Environment (build 1.8.0_211-b12)
> Java HotSpot(TM) 64-Bit Server VM (build 25.211-b12, mixed mode)
> ```

AutoUpgradeと分析モード

AutoUpgradeユーティリティには、アップグレード中に発生する可能性がある多くの問題を回避するのに役立つ広範なシステム・チェックが含まれています。このユーティリティは、新しいOracle Databaseバイナリ・ホームにあります。ただし、最新の更新を取得するには、My Oracle Supportドキュメント2485457.1.から最新バージョンのツールをダウンロードすることをお薦めします。ダウンロードしたファイルは、任意のディレクトリに配置できます。データベースを以前のリリースで実行している間に、

Analyzeを実行して新しいリリースへのアップグレードの準備状況を確認するには、target_versionパラメータを使用して、構成ファイルにターゲット・バージョンを手動で指定する必要があります。たとえば: upg1.target_version=19。

分析モードでAutoUpgradeを実行すると、AutoUpgradeではデータベースからのデータの読取りのみを実行し、データベースに対する更新は実行しません。通常の営業時間に分析モードを使用してAutoUpgradeを実行できます。ターゲット・リリースのOracle Databaseホームを設定する前に、ソースOracle Databaseホームで分析モードでAutoUpgradeを実行できます。

関連項目

- My Oracle Supportノート2485457.1

親トピック: AutoUpgradeを使用したアップグレード前チェックの実行

分析モードを使用したAutoUpgradeの事前チェックの実行の例

アップグレード前にAutoUpgradeを使用して非CDBのOracleデータベースをチェックする方法を確認するには、この例を使用して手順を理解してください。

AutoUpgradeを使用するには、Java 8がインストールされている必要があります。Oracle Databaseリリース12.1 (12.1.0.2)または新しいOracleホームには、デフォルトで有効なJavaバージョンがあります。次の構文を使用して分析モードでAutoUpgradeを起動します。ここで、Oracle_homeはOracleホーム・ディレクトリまたはOracleホームに設定された環境変数で、yourconfig.txtは構成ファイルです。

```
java -jar autoupgrade.jar -config yourconfig.txt -mode analyze
```

AutoUpgradeの実行中に進行状況の概要を取得する場合は、LinuxおよびUnixシステムでlsjコマンドを入力できます。すべてのチェックが完了すると、ツールは、システムの準備状況に関するレポートを提供するアップグレード前修正HTMLファイルを書き込んで、ジョブ実行のステータスを画面に表示し、終了します。すべてのジョブが「正常に終了しました」とリストされている場合は、先に進んでデータベースをアップグレードできることを意味します。ただし、アップグレードを開始する前に従う必要があるお薦めの事項があるかどうかを確認するために、アップグレード前修正HTMLファイルを参照することをお薦めします。いずれかのジョブが「失敗」とリストされている場合は、アップグレードの開始を妨げるエラーがあることを意味します。

例2-4 分析モードでのAutoUpgradeの使用によるOracle Database 12c非CDBシステムのチェック

この例では、AutoUpgradeを/tmpフォルダにダウンロードして、非CDBのOracle Database 12cリリース2 (12.2)システムから新しいOracle Databaseリリースにアップグレードするための構成ファイルを使用してAutoUpgradeを実行しています。

```
java -jar /tmp/autoupgrade.jar -config 122-to-new.txt -mode analyze
```

このチェックのために呼び出される構成ファイル(122-to.new.txt)は次のとおりです。

```
#12.2-to-19c config file
#
global.autoupg_log_dir=/home/oracle/autoupgrade
upg1.source_home=/u01/app.oracle/product/12.2.0.1
upg1.target_home=/u01/app/oracle/product/19
upg1.sid=dbsales
upg1.start_time=now
upg1.log_dir=/home/oracle/autoupgrade/dbsales
upg1.upgrade_node=localhost
```

pdbsパラメータが指定されていないことに注意してください。チェックする特定のPDBを示し、CDB上の他のPDBを除外する場合にのみ、pdbsパラメータを指定する必要があります。log_dirに指定する値は、AutoUpgradeがアップグレード前修正HTMLファイルを配置する場所です。ファイルは次の形式を使用して書き込まれます。ここで、log-pathはログ・ファイルに指

定するパス、sidはOracle Databaseシステム識別子で、job-numberはAutoUpgradeジョブ番号です。

```
/log-path/sid/job-number/prechecks
```

AutoUpgradeを実行すると、出力は次のように表示されます。

```
AutoUpgrade tool launched with default options
+--------------------------------+
| Starting AutoUpgrade execution |
+--------------------------------+
1 databases will be analyzed
Type 'help' to list console commands
upg>
.
.
.
Job 100 completed
------------------- Final Summary -------------------
Number of databases            [1]
Jobs finished successfuly      [1]
Jobs failed                    [0]
Jobs pending                   [0]
------------- JOBS FINISHED SUCCESSFULLY -------------
Job 100 FOR DB12
```

この場合、構成ファイルは、ログ・ファイルを/home/oracle/autoupgradeパスに配置するように指定しています。Oracle Databaseシステム識別子(SID)はCDB1で、AutoUpgradeジョブは100であるため、このジョブのアップグレード・チェックの概要ファイルは、パス/home/oracle/autoupgrade/dbsales/100/prechecksに置かれています。

アップグレード・チェックの概要ファイルを確認し、アップグレードを続行する前にレポートされたエラーを修正します。AutoUpgradeを修正モードで実行すると、多くのエラーを修正できます。

CDBでは、同じプロシージャおよび同じ構成ファイルを使用できます。

> ノート:
>
> AutoUpgrade を CDB に対して分析または修正モードで実行すると、AutoUpgrade は CDB 内のすべての PDB を開き、アクションを完了します。PDB が閉じている場合は、AutoUpgrade が PDB を開いて、分析または修正の完了後に OPEN 状態のままにします。PDB を閉じたままにしてチェックや修正を実行しない場合は、構成ファイル・パラメータ pdb を使用して AutoUpgrade でチェックする PDB をリストすることにより、特定の PDB のみがチェックまたは修正されるように指定できます。

親トピック: AutoUpgradeを使用したアップグレード前チェックの実行

アップグレード・チェックの概要ファイルの確認

AutoUpgradeアップグレード・チェックの概要ファイルを使用してアップグレードの準備をする方法を学習します。

AutoUpgradeの分析モードが完了すると、構成ファイルのlog_dirパラメータで指定したディレクトリにファイルが配置されます。ファイルは次の形式を使用して書き込まれます。ここで、log-pathはログ・ファイルに指定するパス、sidはOracle Databaseシステム識別子(SID)で、job-numberはAutoUpgradeジョブ番号です。

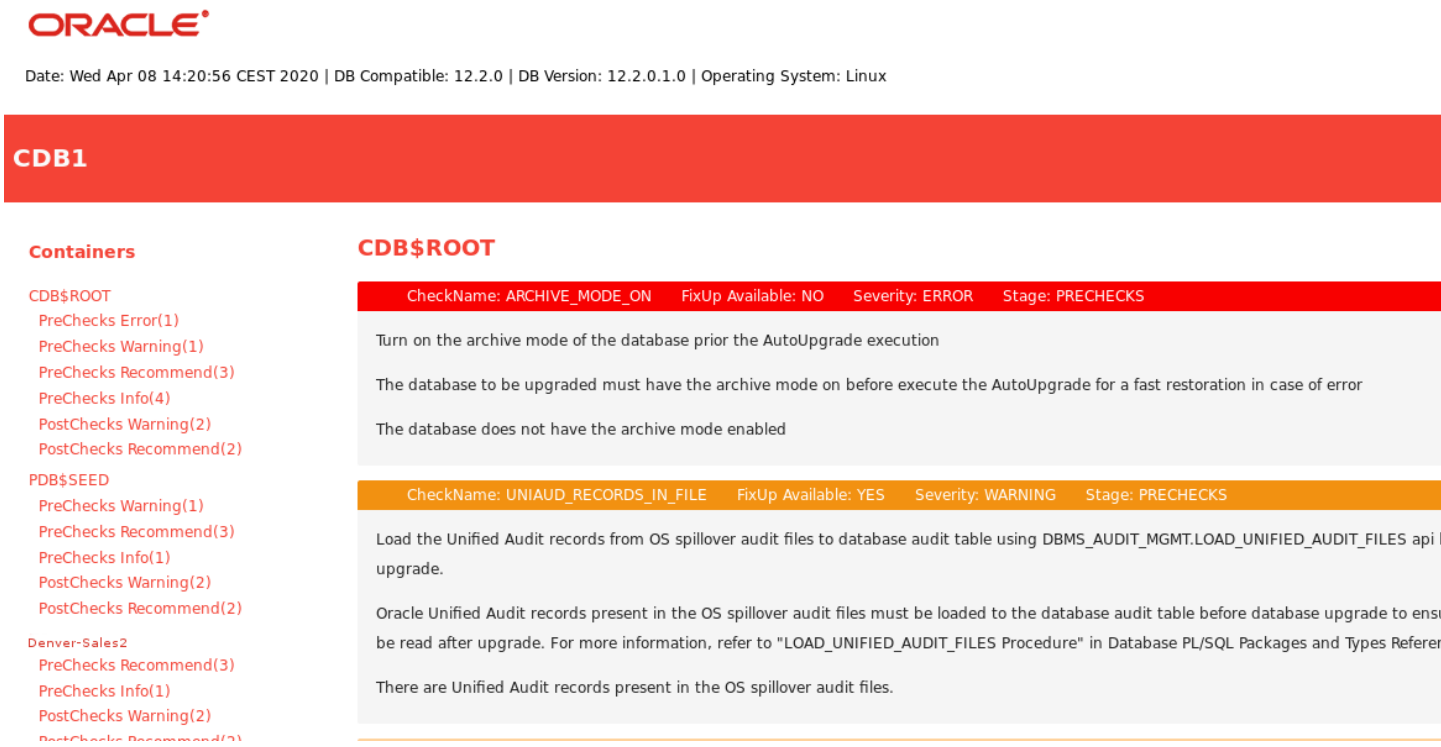
```
/log-path/sid/job-number/prechecks
```

たとえば、ログ・ディレクトリが/home/oracle/autoupgrade/DB12として指定され、データベースSIDがDB12で、ジョブ番号が100の場合は、次のようになります。

```
/home/oracle/autoupgrade/DB12/DB12/100/prechecks
```

アップグレード・チェックの概要ファイルを確認し、アップグレードを続行する前にレポートされたエラーを修正します。AutoUpgradeを修正モードで実行し、多くのエラーを修正することもできます。

図2-1 アップグレード・チェックの概要ファイルの例



結果ファイルの例では、ファイルのトピック領域にCDB$ROOTの2つのレポート・メッセージが表示され、チェックの名前、修正モードでAutoUpgradeを使用して実行可能な修正があるかどうか、問題の重大度、および実行されているAutoUpgradeのステージが示されます。

親トピック: AutoUpgradeを使用したアップグレード前チェックの実行

CDBでAutoUpgradeの事前チェックを実行するための構成ファイルの作成

AutoUpgradeがアップグレードの事前チェックを実行して、マルチテナント・アーキテクチャのOracleデータベースにPDBを含めたり除外する方法をどのようにして制御できるかを確認します。

マルチテナント・コンテナ・データベース(CDB)およびプラガブル・データベース(PDB)で構成されたOracle Databaseサーバーをチェックするには、非CDBのOracleデータベースで使用するのと同じプロシージャおよび構成ファイルを使用できます。分析または修正モードの実行中にAutoUpgradeがチェックを実行すると、CDB内のすべてのPDBが開かれます。AutoUpgradeをCDB上で実行し、PDBが閉じている場合、AutoUpgradeはPDBを開き、分析チェックや修正アクションの後、AutoUpgradeはPDBを開いたままにします。

チェックのために開くPDBを管理して、一部のPDBを閉じたままにできるようにするには、構成ファイルのオプションpdbsを使用して、チェックするPDBのみを含むリストを指定できます。チェックするPDBのリストを指定すると、AutoUpgradeによって、CDB$ROOT、PDB$SEEDおよびリストに指定したすべてのPDBがチェックされます。この例では、denver-sales2という名前のPDBが指定されています。

例2-5 CDBおよびPDBのAutoUpgrade構成ファイル

次の例は、denver-sales2という名前のPDBのみを開いて分析することを指定しています。

```
global.autoupg_log_dir=/home/oracle/autoupgrade
upg1.source_home=/u01/app/oracle/product/12.2.0
upg1.target_home=/u01/app/oracle/product/19
upg1.sid=CDB1
upg1.start_time=now
upg1.log_dir=/home/oracle/autoupgrade/CDB1
upg1.pdbs=denver-sales2
```

親トピック: AutoUpgradeを使用したアップグレード前チェックの実行

以前のリリースのOracleデータベースでのAutoUpgrade修正の実行

この例を使用して、システムに分析モードで生成されるAutoUpgrade修正の実行方法を確認します。

AutoUpgradeを修正モードで実行すると、AutoUpgradeは分析モードでも実行するチェックを実行します。これらのチェックが完了した後、アップグレードを開始する前に、AutoUpgradeによって、新しいリリースに必要なすべての自動修正が実行されます。データベースを新しいリリースに移動する予定がある場合は、修正モードを使用してデータベースをアップグレードする準備をします。

> 注意:
>
> 修正モードで AutoUpgrade を実行する前に、分析モードで AutoUpgrade を個別に実行することをお薦めします。修正モードでは、ソース・データベースを変更できます。

アップグレードの準備の一環として、アップグレード中にエラーの原因となるような状態をソース・データベースで修正する必要がある場合には、修正モードでAutoUpgradeを実行することでソース・データベースに対して自動修正が実行されます。修正モードでのAutoUpgradeの実行は別のシステムに移動する際に実行するステップであるため、保証付きリストア・ポイントは作成されません。このモードは通常の営業時間外で実行することをお薦めします。

Java 8がソースOracleホームにある場合、次の構文を使用して修正モードでAutoUpgradeを起動します。ここで、Oracle_homeはOracleホーム・ディレクトリまたはOracleホームに設定された環境変数で、yourconfig.txtは構成ファイルです。

```
$ java -jar autoupgrade.jar -config yourconfig.txt -mode fixup
```

親トピック: AutoUpgradeを使用したアップグレード前チェックの実行

Oracle Databaseのアップグレード処理のテスト

Oracle Databaseのアップグレードのテスト計画には、これらのテスト手順が含まれる必要があります。

アップグレード前の処理、アップグレード処理およびアップグレード後の処理のすべてをテストするデータベース環境の完全な作業用コピーを作成することをお薦めします。

現行の本番Oracle Databaseに影響を与えないテスト環境を作成できます。たとえば、Oracle Data Guardでは、フィジカル・スタンバイ・データベースおよびスナップショット・スタンバイ・データベースの作成が可能です。

テスト環境は、選択するアップグレード方法によって異なります。

- DBUAを使用する場合または手動アップグレードを実行する場合は、現行の本番データベースのテスト・バージョンを作成します。
- データ・ポンプ・エクスポート/インポートを使用する場合は、現行の本番データベースのサブセットを使用して、段階的にエクスポートおよびインポートします。

テスト環境を使用してデータベースのアップグレードを行います。ベスト・プラクティスは、ダウンサイズしたコピーまたはテスト・データに対してではなく、アップグレードするデータベースの正確なコピーに対してアップグレード処理のテストを実行することです。正確なコピーを実現できない場合は、テスト環境に移動する代表的なデータのサブセットを慎重に選択し、そのデータでアップグレードをテストします。

- 優先度リスト・エミュレーションを使用したアップグレードのテストの例
  マルチテナント・アーキテクチャのOracle Databaseにパラレル・アップグレード・ユーティリティを使用してアップグレード・エミュレーションを実行し、アップグレードを実行する前に優先度リストや他のパラメータ設定をテストできます。
- Oracle Call Interface (OCI)およびプリコンパイラ・アプリケーションのアップグレード
  Oracle Databaseの新しいリリースで使用するすべてのOracle Call Interface (OCI)およびプリコンパイラ・アプリケーションをアップグレードします。

参照:

- データベースのアップグレードのテストの詳細は、『Oracle Database Testingガイド』を参照してください。
- データ・ポンプのエクスポートおよびインポート・ユーティリティの詳細は、『Oracle Databaseユーティリティ』を参照してください。
- フィジカル・スタンバイ・データベースおよびスナップショット・スタンバイ・データベースの詳細は、『Oracle Data Guard概要および管理』を参照してください。

親トピック: Oracle Databaseのアップグレードの準備

優先度リスト・エミュレーションを使用したアップグレードのテストの例

マルチテナント・アーキテクチャのOracle Databaseにパラレル・アップグレード・ユーティリティを使用してアップグレード・エミュレーションを実行し、アップグレードを実行する前に優先度リストや他のパラメータ設定をテストできます。

マルチテナント・アーキテクチャのOracle DatabaseシステムのOracle Database 12cリリース2 (12.2)以上では、優先度リストを使用して、特定のPDBをアップグレードまたは除外したり、特定のアップグレード優先度順を設定できます。優先度エミュレーションを使用してパラレル・アップグレード・ユーティリティを実行することで、実際にアップグレードを実行せずに優先度リストを

テストできます。パラレル・アップグレード・ユーティリティのエミュレーション機能を使用して、優先度リストを使用するアップグレード計画をテストします。

アップグレード・エミュレーション・テストの準備

エミュレーションを実行する前に、ソースおよびターゲットのアップグレードの場所を設定して、実際のアップグレードを準備する場合と同じようにデータベースを準備する必要があります。アップグレードは実際には発生しませんが、パラレル・アップグレード・ユーティリティによって、実際のアップグレードがどのように実行されるかを示したログ・ファイルが生成されます。

ノート:

-E パラメータを使用してエミュレーション・モードでパラレル・アップグレード・ユーティリティを実行することで、優先度リストがどのように実行されるかをテストしたり、他のアップグレード・パラメータの選択がアップグレード中にどのように実行されるかをテストできます。たとえば、-n および-N パラメータを使用して行うリソース割当ての選択がどのように実行されるかについて、詳細情報を取得するアップグレード・エミュレーションを実行できます。

優先度リスト・エミュレーションを実行するための構文

パラレル・アップグレード・ユーティリティで通常使用するパラメータ設定をどれでも使用できます。ただし、優先度リストを作成する必要があり、パラレル・アップグレード・ユーティリティを-Eパラメータとともに実行してアップグレード・エミュレーションを実行するように設定するときに、-Lパラメータを使用してそのリストを呼び出す必要があります。

次に、優先度リスト・エミュレーションを使用してパラレル・アップグレード・ユーティリティを実行するために最低限必要な構文の例を示します(priority_list_nameは優先度リスト・ファイルの名前)。

```
catctl -E -L priority_list_name catupgrd.sql
```

例2-6 優先度リスト・エミュレーションを使用したパラレル・アップグレード・ユーティリティの実行の例

次の例では、plist.txtという名前のこの優先度リストを使用します。

```
1,CDB$ROOT
2,PDB$SEED
3,CDB1_PDB2
4,CDB1_PDB4
4,CDB1_PDB3
5,CDB1_PDB5
5,CDB1_PDB1
```

次のコマンドでは、この優先度リストを呼び出してパラレル・エミュレーションを実行します。

```
$ORACLE_HOME/perl/bin/perl catctl.pl -L plist.txt -E -n 4 -N 2 catupgrd.sql
```

このコマンドでは、次のパラメータ設定を使用します。

- -Eでは、パラレル・アップグレード・ユーティリティによってアップグレード手順をエミュレーション・モードで実行するように指定します。
- -n 4では、アップグレードで4つのプロセスを割り当ててパラレル・アップグレード操作を実行するように指定します。
- -N 2では、アップグレードで2つのSQLプロセッサを実行してPDBをアップグレードするように指定します。同時実行されるPDBアップグレードの最大数は、-Nの値で割られた-nの値になるため、このアップグレードでは、最大2つの同時PDBアップグレードが実行されます。

- -Lでは、アップグレード優先度を設定するためにコマンドで読み取る優先度リストを指定します。

パラレル・アップグレード・ユーティリティでエミュレートされたアップグレードを実行すると、実際のアップグレード中に表示されるものと同じ出力が画面に表示されます。

アップグレード・エミュレーションが完了すると、ログ・ファイル(catctl_prority_run.list)が生成されますが、これはデフォルトのログ・ディレクトリに格納されるか、-lパラメータで指定したログ・ディレクトリの場所に格納されます。この例では、他のログ・ディレクトリを指定せず、CDB1というコンテナ・データベースでアップグレードを実行するため、出力は、パスOracle_base/cfgtoollogs/CDB1/runに配置されます(Oracle_baseはアップグレードを実行するユーザーのOracleベースで、CDB1はアップグレードを実行するコンテナ・データベース(CDB)の名前です)。

ログ・ファイルcatctl_priority_run.lstにはアップグレード・エミュレーション中にアップグレードがどのような優先順位で実行されたかのリストが表示されます。また、パラレル・アップグレード・ユーティリティによるPDBアップグレードのグループ化の方法を示します。実際にアップグレードを実行する前に、グループ化と優先度が含まれた優先実行を調べることができます。アップグレードによって生成されるログ・ファイルは、アップグレードが完了した後の画面にも表示されます。

アップグレード・ログの最後に、CDB$ROOTが最初にアップグレードされることが示されます。CDB$ROOTのアップグレードの完了後、パラレル・アップグレード・ユーティリティでは、優先度リストの優先度設定に従って、次のPDBの同時アップグレードが実行されます。

1. PDB$SEEDおよびCDB1_PDB2。出力ログは、PDB$SEEDの場合、pdb_seedとして指定されているログ識別子(ログID)で、CDB_1PDB2の場合、ログID mayapdb2で生成されます
2. CDB1_PDB3およびCDB1_PDB4。ログIDは、mayapdb3およびmayapdb4として指定されます
3. CDB1_PDB5およびCDB1_PDB6。ログIDは、mayapdb5およびmayapdb6として指定されます
4. CDB1_PDB1。ログIDは、mayapdb1として指定されます。

関連項目

- 「Oracle Databaseのパラレル・アップグレード・ユーティリティについて(CATCTL.PLおよびDBUPGRADE)」
- パラレル・アップグレード・ユーティリティ(catctl.pl)のパラメータ

親トピック: Oracle Databaseのアップグレード処理のテスト

Oracle Call Interface (OCI)およびプリコンパイラ・アプリケーションのアップグレード

Oracle Databaseの新しいリリースで使用するすべてのOracle Call Interface (OCI)およびプリコンパイラ・アプリケーションをアップグレードします。

現行の本番データベースをアップグレードする前に、これらのアプリケーションをテスト・データベースでテストすることをお薦めします。

関連項目

- Oracle DatabaseでのプリコンパイラおよびOCIアプリケーションのアップグレードについて

親トピック: Oracle Databaseのアップグレード処理のテスト

Oracle Label SecurityおよびOracle Database Vaultを使用しているデータベースのアップグレードの要件

Oracle Label SecurityまたはOracle Database Vaultを使用しているデータベースのアップグレードを開始する前に、これらの作業を完了する必要があります。

- 監査表の事前アップグレードおよびアーカイブの要件
  Oracle Label SecurityおよびOracle Database Vaultを使用している12.1より前のOracle Databaseリリースでは、アップグレードの前にOLS事前処理スクリプトを実行する必要があります。
- Oracle Database Vaultを使用したデータベースのアップグレードの準備
  アップグレードする予定のOracle DatabaseでOracle Database Vaultが使用される場合は、アップグレードを開始する前にOracle Database Vaultを無効にする必要があります。
- Oracle Database VaultおよびOracle Databaseリリース11.2のアップグレード
  アップグレード対象の以前のリリースにOracle Label Securityがインストールされている場合は、SYSにDV_PATCH_ADMINロールを付与します。

親トピック: Oracle Databaseのアップグレードの準備

監査表の事前アップグレードおよびアーカイブの要件

Oracle Label SecurityおよびOracle Database Vaultを使用している12.1より前のOracle Databaseリリースでは、アップグレードの前にOLS事前処理スクリプトを実行する必要があります。

Oracle Label Security (OLS)およびOracle Database Vaultを使用するOracle Databaseリリース12.1より前のデータベースをアップグレードする場合は、最初にOLS事前処理スクリプト(`olspreupgrade.sql`)を実行して`aud$`表のコンテンツを処理する必要があります。OLSアップグレードでは、`aud$`表が、`SYSTEM`スキーマから`SYS`スキーマに移動されます。`olspreupgrade.sql`スクリプトは、この移動に必要な事前処理スクリプトです。

> 注意:
>
> Oracle Label Security および Oracle Database Vault を使用する Oracle Database リリース 12.1 より前のデータベースをアップグレードする前には、`olspreupgrade.sql` スクリプトを実行する必要があります。Oracle Database リリース 12.1 にアップグレード後に、データベースのパッチまたはアップグレードをするための OLS 事前処理の手順を実行する必要はありません。

`olspreupgrade.sql`スクリプトでは、`SYS`スキーマに一時表`PREUPG_AUD$`が作成され、`SYSTEM.aud$`レコードが`SYS.PREUPG_AUD$`に移動されます。安全対策として、`olspreupgrade.sql`スクリプトを実行する前に監査証跡をアーカイブすることをお薦めします。Oracle Label Securityがデータベースにインストールされていて、以前のリリースからアップグレードする場合は、アップグレードの前にOLS事前処理スクリプトを実行する必要があります。

参照:

監査証跡のアーカイブの詳細は、『Oracle Databaseセキュリティ・ガイド』を参照してください

OLS事前処理スクリプトの詳細は、Oracle Label Security管理者ガイドを参照してください。

親トピック: Oracle Label SecurityおよびOracle Database Vaultを使用しているデータベースのアップグレードの要件

Oracle Database Vaultを使用したデータベースのアップグレードの準備

アップグレードする予定のOracle DatabaseでOracle Database Vaultが使用される場合は、アップグレードを開始する前にOracle Database Vaultを無効にする必要があります。

アップグレード・プロセス中に、ソースOracle DatabaseでOracle Database Vaultが使用されている場合は、アップグレードを開始する前にまずOracle Database Vaultを無効にする必要があります

使用できるオプションは2つあります。

1. 手動プロシージャを使用します。`CDB$ROOT` に共通のDatabase Vault (DV)管理者としてログオンし、`DV_PATCH_ADMIN`ロールをSYSに付与するか、ログインしてすべてのコンテナでOracle Database Vaultを無効にします。手順は、アップグレード・シナリオによって少し異なります。この手順は、My Oracle Supportの「Database Vaultを使用したデータベースのアップグレードの要件(ドキュメントID 2757126.1)」に記載されています。
2. 最新のAutoUpgrade Jarファイルをダウンロードし、ここで説明されている手順を実行します。

どちらのオプションでも、分析モードでAutoUpgradeを実行すると、Oracle Database Vaultが有効であることが検出され、Oracle Database Vaultの前提条件およびアップグレードが満たされていることを確認する必要があることがレポートに示されます。

例2-7 Oracle Database Vaultを使用するデータベースのAutoUpgradeプロシージャ

AutoUpgradeを使用しており、データベースがOracle Database Vaultで構成されている場合、アップグレード手順は次のとおりです:

1. Oracle Database Vaultを無効にします。
2. 新しいOracle Databaseリリースをインストールします。
3. 最新のAutoUpgrade JARファイルをMy Oracle Supportノート2485457.1からダウンロードし、新しいOracle Databaseリリースのパス`Oracle_home/rdbms/admin`でAutoUpgrade JARファイルを置き換えます。
4. AutoUpgradeユーティリティ(またはDatabase Upgrade Assistant)を実行し、アップグレードを完了します。
5. アップグレードされたOracle DatabaseでOracle Database Vaultを有効にします。

関連項目

- Oracle Database Vaultの無効化および有効化
- Database Vaultを使用したデータベースのアップグレードの要件(ドキュメントID 2757126.1)
- AutoUpgradeツール(ドキュメントID 2485457.1)

親トピック: Oracle Label SecurityおよびOracle Database Vaultを使用しているデータベースのアップグレードの要件

Oracle Database VaultおよびOracle Databaseリリース11.2のアップグレード

アップグレード対象の以前のリリースにOracle Label Securityがインストールされている場合は、SYSにDV_PATCH_ADMINロールを付与します。

Oracle Database Vaultがリリース11.2データベースにインストールされていない場合は、この項のステップ2、3、6および7を

スキップできます。

アップグレード前にリリース11.2データベースでOLS事前処理スクリプトを実行するには:

1. 新しくインストールしたOracleホームからアップグレードするデータベースのOracleホームに次のスクリプトをコピーします。
   - `ORACLE_HOME/rdbms/admin/olspreupgrade.sql`
   - `ORACLE_HOME/rdbms/admin/emremove.sql`
   - `ORACLE_HOME/rdbms/admin/catnoamd.sql`
2. SQL*Plusを起動し、アップグレードするデータベースにDVOWNERとして接続します。
3. 次の文を実行します。

```
SQL> GRANT DV_PATCH_ADMIN to SYS;
```

4. システム・プロンプトで、SYS as SYSDBAで接続します。

```
CONNECT SYS AS SYSDBA
```

5. Data Vaultに対して事前処理スクリプトを実行します

```
ORACLE_HOME/rdbms/admin/olspreupgrade.sql
ORACLE_HOME/rdbms/admin/emremove.sql
ORACLE_HOME/rdbms/admin/catnoamd.sql
```

   事前処理スクリプトの実行中も、データベース上でアプリケーションを継続して実行できます。

6. `olspreupgrade.sql`の実行が正常に完了したら、SQL*Plusを起動してデータベースにDVOWNERとして接続します。
7. 次のSQL文を実行します。

```
SQL> REVOKE DV_PATCH_ADMIN from SYS;
```

親トピック: Oracle Label SecurityおよびOracle Database Vaultを使用しているデータベースのアップグレードの要件

アップグレードのためのOracle Databaseのバックアップ

この手順を使用して、アップグレードを試みる前に既存のOracle Databaseをバックアップします。

注意:

Oracle ソフトウェアを変更する前に、Oracle ソフトウェアおよびデータベースのバックアップを作成することをお薦めします。Microsoft Windows オペレーティング・システム上で動作する Oracle ソフトウェアでは、Windows レジストリもバックアップする必要があります。Microsoft Windows では、レジストリのバックアップがないと、アップグレードが失敗して以前のソフトウェア・インストールに戻す場合に、Oracle ソフトウェアを動作状態にリストアできません。

データベースを正常に停止する前に、preupgradeパラメータを使用してAutoUpgradeを実行する必要があります。停止時間を最小限に抑えるために、オンライン・バックアップを実行するか、保証付きリストア・ポイントを作成できます。

1. 次のように、Oracle RMANにサインオンします。

```
rman "target / nocatalog"
```

2. 次のRMANコマンドを実行します。

```
RUN
{
    ALLOCATE CHANNEL chan_name TYPE DISK;
    sql 'ALTER SYSTEM ARCHIVE LOG CURRENT';
    BACKUP DATABASE FORMAT '/tmp/db%U' TAG before_upgrade
    PLUS ARCHIVELOG FORMAT '/tmp/arch%U' TAG before_upgrade;
    BACKUP CURRENT CONTROLFILE FORMAT '/tmp/ctl%U' TAG before_upgrade;
}
```

注意:

このバックアップ中に他の RMAN バックアップが実行されないようにする必要があります。別の RMAN コマンドによってアーカイブ・ログがバックアップおよび削除された場合、他の RMAN コマンドによってアーカイブ・ログが削除されているため、このバックアップはリカバリできない可能性があります。

関連項目

- オンライン・バックアップおよびバックアップ・モードについて
- フラッシュバック・データベースおよびリストア・ポイントの使用
- データベースのバックアップ

親トピック: Oracle Databaseのアップグレードの準備

3 Oracle DatabaseアップグレードでのAutoUpgradeの使用

AutoUpgradeを使用してアップグレード・タスクを簡略化する方法について学習します。

- Oracle Database AutoUpgradeについて
  AutoUpgradeユーティリティは、アップグレードの開始前、アップグレードのデプロイ中、およびアップグレード後のチェックと構成の移行中の両方で、アップグレード・プロセスを自動化するように設計されています。
- AutoUpgradeの使用方法の例
  アップグレードをガイドするには、アップグレードのユースケースに一致するAutoUpgradeワークフローの例を使用します。
- 自動アップグレード・メッセージおよびプロセスの説明用語
  アップグレード・チェックおよび操作の処理方法を理解するために、ユーティリティの実行時に生成されるAutoUpgradeユーティリティ・メッセージについて学習します。
- AutoUpgrade処理モードについて
  4つのAutoUpgrade処理モード(分析、修正、デプロイおよびアップグレード)は、AutoUpgradeの実行時に実行するアクションを示します。
- AutoUpgradeワークフローおよびステージの理解
  AutoUpgradeワークフローは、一般的なアップグレード・プロセスの各ステップを自動化します。実行されるステージは、選択した処理モードによって異なります。
- AutoUpgradeを使用した非CDBからPDBへのアップグレードの理解
  1回の操作で非CDBを新しいCDBのPDBにアップグレードおよび変換するか、非CDBデータベースを既存のCDBのPDBにアップグレードしてから変換できます。
- AutoUpgradeを使用した切断/接続のアップグレードの理解
  AutoUpgradeでは、以前のリリースのソース・コンテナ・データベース(CDB)からプラガブル・データベース(PDB)を切断し、新しいリリースのターゲットCDBに接続して、PDBをターゲットCDBリリースにアップグレードするために必要なすべてのステップを完了できます。
- AutoUpgradeコマンドライン・パラメータおよびオプション
  AutoUpgradeパラメータを確認し、Oracle Databaseのアップグレードのユース・ケースのパラメータおよびオプションを選択します。
- AutoUpgradeユーティリティ構成ファイル
  AutoUpgrade構成ファイルには、Oracle Databaseのアップグレードを実行するために必要なすべての情報が含まれています。
- AutoUpgradeおよびOracle Databaseの構成オプション
  AutoUpgradeを実行すると、データベースのタイプ(Oracle Database、Oracle ASMを備えたOracle DatabaseスタンドアロンまたはOracle RAC)が判別され、そのタイプのデータベースのアップグレードが実行されます
- AutoUpgrade構成ファイルの例
  次の例を使用して、アップグレード時に様々な構成アクションを実行するために独自のAutoUpgrade構成ファイルを変更する方法を理解します。
- AutoUpgrade内部設定構成ファイル
  内部構成設定では、AutoUpgradeの実行方法を制御します。
- AutoUpgradeログ・ファイルの構造
  AutoUpgradeユーティリティは、ジョブ・ステータスおよび構成ファイルを含むログ・ファイルの構造を生成します

- AutoUpgradeのための完全なデプロイの有効化
  保証付きリストア・ポイント(GRP)を有効にしてアップグレードをフラッシュバックできるようにするには、アーカイブ・ロギングを設定し、AutoUpgradeを有効にしてアップグレードを完了できるように他のタスクを完了する必要があります。
- AutoUpgradeコンソールの使用例
  AutoUpgradeコンソールには、AutoUpgradeジョブの進行状況を監視する一連のコマンドが用意されています。コンソールは、AutoUpgradeユーティリティの実行時にデフォルトで起動され、パラメータconsoleおよびnoconsoleによって有効または無効になります。
- AutoUpgradeの既知の制限
  アップグレードで問題が発生した場合は、既知の制限を確認して解決策を見つけてください。
- AutoUpgradeによるデータベース変更の適切な管理
  AutoUpgradeは強力なユーティリティであるため、責任を持って使用する必要があります。データベースを危険にさらすような方法でAutoUpgradeを使用することを確認し、回避してください。
- デフォルト修正のオーバーライド方法
  RUNFIX列エントリを使用すると、修正の無効化がセキュリティまたはOracleポリシーに違反する場合を除き、自動修正を無効にできます。
- ローカル構成ファイルのパラメータ修正チェックリストの例
  アップグレード時に個々のデータベースに対して特定の修正を含めるか除外する場合は、ローカル構成ファイル・チェックリストを使用します。
- AutoUpgradeおよびMicrosoft WindowsのACLとCLI
  Microsoft WindowsシステムでAutoUpgradeを実行する場合、アクセス制御リスト(ACL)およびコマンドライン・インタフェース(CLI)を使用した追加のベスト・プラクティスをお薦めします。

Oracle Database AutoUpgradeについて

AutoUpgradeユーティリティは、アップグレードの開始前、アップグレードのデプロイメント時、およびアップグレード後のチェックと構成の移行時、アップグレード・プロセスを自動化するように設計されています

アップグレードを実行するときは、My Oracle Supportドキュメント2485457.1から最新バージョンのAutoUpgradeユーティリティをダウンロードし、`autoupgrade.jar`を使用してアップグレードを準備およびデプロイすることをお薦めします。新しいOracle Databaseリリースのバイナリをダウンロードした後にAutoUpgradeを使用し、新しいリリースのOracleホームを設定します。AutoUpgradeを使用するときに、各データベース・デプロイメントで必要に応じてカスタマイズされた単一の構成ファイルを使用して、複数のOracle Databaseデプロイメントを同時にアップグレードできます。

The `autoupgrade.jar`ファイルは、デフォルトでOracleホームに存在します(`Oracle_home/rdbms/admin`)。ただし、AutoUpgradeを使用する前に、最新のAutoUpgradeバージョンをダウンロードすることをお薦めします。AutoUpgradeは各リリース・アップデート(RU)ごとに含まれていますが、最新のAutoUpgradeバージョンは、常にMy Oracle Supportドキュメント2485457.1.から入手できます。

ノート:

AutoUpgrade は、Oracle Database Enterprise Edition および Oracle Database Standard Edition で使用できます。他の Oracle Database エディションでは使用できません。

問題の防止: 分析モードおよび修正モード

アップグレードの前に、修正が必要な問題を識別できるように、分析モードではAutoUpgradeユーティリティによってデータベースの読取り専用分析が実行されます。ユーティリティは、通常のデータベース操作中に実行できます。修正モードでは、AutoUpgradeユーティリティによって、手動介入を必要とする修正とアップグレード・デプロイメント・フェーズ中にAutoUpgradeユーティリティで実行できる修正の両方を検出および識別します。

アップグレードの簡略化: デプロイおよびアップグレード・モード

デプロイ・フェーズで、AutoUpgradeユーティリティは構成ファイルに指定したデータベースを変更します。これにより、アップグレード時に独自のカスタム・スクリプトをコールしてデータベースを構成できます。多くの場合、AutoUpgradeユーティリティはアップグレード・プロセス中にデータベースに対して自動修正を実行できるため、手動介入は必要ありません。

アップグレード後のチェックおよび修正のデプロイおよびアップグレード

デプロイまたはアップグレード・モードのいずれかでアップグレードが完了すると、AutoUpgradeはアップグレード後のチェックを実行します。これは、AutoUpgrade構成ファイルに指定する構成手順に従って、アップグレードした各データベースでカスタム・スクリプトを実行できるようにするプロセスを提供し、アップグレード後のプロセスの一部としてアップグレード後の自動修正を実行することもできます。デプロイ・モードでは、AutoUpgradeではアップグレードが成功したことも確認し、`sqlnet.ora`、`tnsname.ora`および`listener.ora`などのデータベース・ファイルをソース・ホームからターゲット・ホームにコピーします。これらのアクションが完了すると、アップグレードされたOracle Databaseリリースが新しいOracleホームで起動されます。

Oracle LiveLabsでハンズオン・ラボを使用してAutoUpgradeを試す

Oracle LiveLabsでは、無料のラボ環境でOracleテクノロジを試すことができます。Daniel Overby Hansenが、Oracle LiveLabsを使用して独自のデモ・ラボ環境をOracle Cloud環境に設定する方法を紹介します。ぜひ参照してください。

Oracle LiveLabsでハンズオン・ラボを使用してAutoUpgradeを試す

関連項目

- My Oracle Supportのドキュメント2485457.1
- Oracle LiveLabs

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

AutoUpgradeの使用例

アップグレードをガイドするには、アップグレードのユース・ケースに一致するAutoUpgradeワークフローの例を使用します。

次の例は一般的なワークフロー順序で示されます。構成ファイルを使用し、noconsoleパラメータを指定してスクリプトを実行する方法は、「AutoUpgradeコンソールの使用方法」の例を参照してください。

- 同一サーバー上にソースおよびターゲットのデータベース・ホームがあるAutoUpgrade (通常)
  Oracle Databaseのソースおよびターゲットの Oracleホームが同じ物理サーバーにインストールされている場合、この例を使用します。
- 異なるサーバー上にソースおよびターゲット・データベース・ホームがあるAutoUpgrade
  Oracle Databaseのソースおよびターゲットの Oracleホームが異なる物理サーバーにあるときは、両方のサーバーでタスクを完了する必要があります。

関連項目

- AutoUpgradeコンソールの使用例

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

同じサーバー上にソースおよびターゲットのデータベース・ホームがあるAutoUpgrade(通常)

Oracle Databaseのソースおよびターゲットの Oracleホームが同じ物理サーバーにインストールされている場合、この例を使用します。

コンテキスト: ソース・ホームとターゲット・ホームが同じサーバー上にあります。

分析を開始するには、次のコマンドを入力します。

```
java -jar autoupgrade.jar -config config.txt -mode analyze
```

このコマンドは、レポートを生成し、コマンドが検出したエラー状態を示します。エラー状態を確認してください。

アップグレードのデプロイメントを開始するには、次のコマンドを入力します。

```
java -jar autoupgrade.jar -config config.txt -mode deploy
```

親トピック: AutoUpgradeの使用方法の例

異なるサーバー上にソースおよびターゲット・データベース・ホームがあるAutoUpgrade

Oracle Databaseのソースおよびターゲットの Oracleホームが異なる物理サーバーにあるときは、両方のサーバーでタスクを完了する必要があります。

コンテキスト: ソースおよびターゲットの Oracleホームが異なる物理サーバー上にあります。

分析を開始するには、次のコマンドを入力します。

```
java -jar autoupgrade.jar -config config.txt -mode analyze
```

このコマンドは、レポートを生成し、コマンドが検出したエラー状態を示します。エラー状態を確認してください。

ソースとターゲットのOracle Database Oracleホームは異なるサーバー上にあるため、ソース・サーバーで修正を実行し、ターゲット・サーバーでアップグレードを実行します。

1. ソース・サーバーで修正を実行します。

```
java -jar autoupgrade.jar -config config.txt -mode fixups
```

2. タスクを実行してソースOracle Databaseをソース・サーバーからターゲット・サーバーに移動します。
3. ターゲット・サーバー上で、アップグレード・モードでデータベースを起動し、upgradeモードでAutoUpgradeを実行します。

```
java -jar autoupgrade.jar -config config.txt -mode upgrade
```

親トピック: AutoUpgradeの使用方法の例

AutoUpgradeメッセージおよびプロセスの説明用語

アップグレード・チェックおよび操作の処理方法を理解するために、ユーティリティの実行時に生成されるAutoUpgradeユーティリティ・メッセージについて学習します。

- AutoUpgradeジョブIDの概要
  AutoUpgrade**ジョブ**は、ジョブ識別子(`jobid`)で識別される、アップグレードに関連付けられた作業の単位です。
- AutoUpgradeステージの概要
  AutoUpgradeユーティリティ・ジョブは**ステージ**と呼ばれる一連のフェーズを経由し、そこで特定のアクションが実行されます。
- AutoUpgradeステージの操作および状態の概要
  AutoUpgradeでは、**操作**はステージ中に実行されたアクションを表し、**状態**はステージ操作のステータスを示します。

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

AutoUpgradeジョブIDの概要

AutoUpgrade**ジョブ**は、ジョブ識別子(`jobid`)で識別される、アップグレードに関連付けられた作業の単位です。

ジョブは、AutoUpgradeによって実行される一連のアクションを表します。各ジョブには、目的を遂行するための一連のステージがあります。ジョブは、`jobid`と呼ばれる一意の正の整数で識別されます。新しいAutoUpgradeジョブはそれぞれ、AutoUpgradeユーティリティの構成ファイルにあるデータベースごとに新しいジョブID (`jobid`)を生成します。`jobid`を持つ起動済のデータベースがシステム上に存在し、未完了であることが自動アップグレードによって検出された場合、AutoUpgradeは、この既存の`jobid`を**再開**操作として識別します。再開操作では、前回のAutoUpgrade実行時に完了しなかった、`jobid`によって識別されるジョブのステージが、停止した時点から続行されます。

親トピック: AutoUpgradeメッセージおよびプロセスの説明用語

AutoUpgradeステージの概要

AutoUpgradeユーティリティ・ジョブは**ステージ**と呼ばれる一連のフェーズを経由し、そこで特定のアクションが実行されます。

ステージ中に発生するアクションは、AutoUpgradeに対して選択した**処理モード**(**分析**、**修正**、**デプロイ**および**アップグレード**)によって定義されます。

AutoUpgradeには次のステージがあります。

- SETUP: AutoUpgradeユーティリティ・ジョブ・マネージャがジョブを起動する準備の一部として作成する初期ステージ。
- PREUPGRADE: 十分な使用可能ディスク領域があるかどうかを判定するためのチェックなど、AutoUpgradeで現在のシステム構成に基づいてシステムのチェックを実行して、アップグレードの準備状況を判定するステージ。
- PRECHECKS: AutoUpgradeがソースOracleデータベースを分析して、データベースがアップグレード要件を満たしているかどうかを判定するステージ。
- GRP: 保証付きリストア・ポイント(GRP)。AutoUpgradeでアップグレード・プロセスの開始前に作成されます。このオプションは、Oracle Database Enterprise Editionのリリースにのみ使用可能です。Oracle Database Standard Editionには使用できません。AutoUpgradeではデフォルトでGRPが作成されますが、アップグレードを開始する前にバックアップを実行することをお薦めします。
- PREFIXUPS: アップグレードを開始する前に、AutoUpgradeがアップグレード前の修正を実行するステージ。たとえ

ば、これは、AutoUpgradeがソースのOracleホームでディクショナリ統計を収集するステージです。

- DRAIN: AutoUpgradeがデータベースを停止する段階。
- DBUPGRADE: AutoUpgradeがアップグレードを実行し、アップグレードの完了後に見つかった無効なオブジェクトを収集するステージ。
- POSTCHECKS: アップグレード後の修正を開始する前に、AutoUpgradeがターゲットのOracleホーム(アップグレードされたOracleデータベース)でチェックを実行するステージ。
- POSTFIXUPS: AutoUpgradeがアップグレード後の修正処理(タイムゾーンのアップグレードなど)を実行するステージ。
- POSTUPGRADE: AutoUpgradeが、ソースのOracleホーム構成ファイル(`tnsnames.ora`、`sqlnet.ora`およびその他のファイル)をターゲットのOracleホームにコピーまたはマージするステージ。

親トピック: AutoUpgradeメッセージおよびプロセスの説明用語

AutoUpgradeステージの操作および状態の概要

AutoUpgradeでは、**操作**はステージ中に実行されたアクションを表し、**状態**はステージ操作のステータスを示します。

操作メッセージの理解

操作メッセージは、AutoUpgrade状態で発生している内容を説明する内部フェーズ・メッセージです。操作メッセージには、次の2種類があります。

PREPARING: AutoUpgradeステージを完了する準備として、AutoUpgradeインスタンスが作成、初期化またはコールされます。これは情報メッセージです。このメッセージが表示された場合、実行するアクションはありません。

EXECUTING: AutoUpgradeはステージのメイン・ワークフロー実行の処理中です。これは情報メッセージです。実行するアクションはありません。

状態メッセージの理解

状態メッセージは、メッセージが表示されるステージの現在のワークフローのステータスを示します。次の4つの状態メッセージがあります。

- ABORTED: AutoUpgradeはユーザー・リクエストに応答してステージ・ワークフローの実行を停止します。
- ERROR: ステージ・ワークフローの実行中にエラーが発生しました。エラーの原因を確認します。
- FINISHED: AutoUpgradeはステージのワークフローを正常に完了しました。
- RUNNING: AutoUpgradeはステージ・ワークフローを実行しています。

親トピック: AutoUpgradeメッセージおよびプロセスの説明用語

AutoUpgrade処理モードについて

4つのAutoUpgrade処理モード(分析、修正、デプロイおよびアップグレード)は、AutoUpgradeの実行時に実行するアクションを示します。

- AutoUpgrade処理モードを実行するための準備
  AutoUpgrade処理モードを実行する前に、準備を完了している必要があります。
- AutoUpgrade分析処理モードについて
  AutoUpgrade分析(`analyze`)処理モードでは、データベースをチェックしてアップグレードの準備が完了しているかどうかを確認します。
- AutoUpgrade修正処理モードについて
  AutoUpgrade修正(`fixups`)処理モードでは、データベースを分析し、アップグレードを実行する前に修正する必要がある項目の修正を実行します。
- AutoUpgradeデプロイ処理モードについて
  AutoUpgradeデプロイ(`deploy`)処理モードでは、データベースの実際のアップグレードが実行され、保留中のすべての修正が実行されます。
- AutoUpgradeアップグレード処理モードについて
  AutoUpgradeアップグレード(`upgrade`)処理モードでは、ソースまたはターゲットのOracleホームをアップグレードできます。

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

AutoUpgrade処理モードを実行するための準備

AutoUpgrade処理モードを実行する前に、準備を完了している必要があります。

AutoUpgrade処理モードを使用する前に、次の要件を満たしていることを確認してください。

- ユーザー構成ファイルが作成されていること。
- ソースOracle Databaseリリースは、元のOracleホームで稼働しています。AutoUpgradeの再起動の場合は、アップグレード・フローのフェーズに対応するOracleホームでデータベースを起動する必要があります。
- データベースが実行されているサーバーが、サーバー・ホスト・ファイル(たとえば`/etc/hosts`)またはドメイン・ネーム・サーバー(DNS)に登録されていること。

  ターゲット・データベースが配置されているサーバーにログインしていて、データベースが`localhost`で実行されている場合、またはAutoUpgradeが実行されている場合は、AutoUpgradeの`config`ファイルからhostnameパラメータを削除します。

- コンテナ・データベース(CDB)で、プラガブル・データベース(PDB)のサブセットをアップグレードする場合、アップグレードを実行するPDBはオープンされ、AutoUpgradeローカル・パラメータ`pdbs`を使用してユーザー構成ファイルで構成されます。PDBのリストを指定しない場合、AutoUpgradeはCDB上のすべてのPDBをアップグレードします。
- AutoUpgrade jarファイル(`autoupgrade.jar`)がダウンロード済または使用可能であり、Java 8配布を使用して実行できること。
- AutoUpgradeをバッチまたはスクリプトで実行する場合は、コマンドで`noconsole`パラメータを使用してAutoUpgradeをコールしておきます。

Oracle Database 19c (19.3)以降のターゲットのOracleホームでは、デフォルトで`autoupgrade.jar`ファイルが存在し

ます。ただし、AutoUpgradeを使用する前に、My Oracle Supportドキュメント2485457.1から入手可能な最新バージョンをダウンロードすることをお薦めします。

関連項目

- My Oracle SupportのドキュメントMy Oracle Supportドキュメント2485457.1

親トピック: AutoUpgrade処理モードについて

AutoUpgrade分析処理モードについて

AutoUpgrade分析(analyze)処理モードでは、データベースをチェックしてアップグレードの準備が完了しているかどうかを確認します。

分析モードでAutoUpgradeを実行すると、AutoUpgradeではデータベースからのデータの読取りのみを実行し、データベースに対する更新は実行しません。通常の営業時間に分析モードを使用してAutoUpgradeを実行できます。ターゲット・リリースのOracle Databaseホームを設定する前に、ソースOracle Databaseホームで分析モードでAutoUpgradeを実行できます。

次の構文を使用して、分析モードでAutoUpgradeを起動します。ここで、Java-8-homeはJava 8ディストリビューションの場所、またはJava 8ホームに設定された環境変数であり、path/yourconfig.txtは構成ファイルのパスとファイル名です。

```
Java-8-home/bin/java -jar autoupgrade.jar -config /path/yourconfig.txt -mode analyze
```

たとえば、rdbms/adminの下の新しいリリースのOracleホームに最新のAutoUpgradeリリースをコピーし、そのホームの環境変数を21CHOMEに設定して、ディレクトリ/scriptsの下のOracleユーザー・ホームに構成ファイルをコピーし、21config.cfgという名前を付けた場合、次のコマンドを入力します。

```
java -jar $21CHOME/rdbms/admin/autoupgrade.jar -config /scratch/scripts/21config.cfg
-mode analyze -mode analyze
```

Oracle Databaseリリース12.2 (12.2.0.1)または新しいOracleホームには、デフォルトで有効なJavaバージョンがあります。

AutoUpgrade分析モードでは2つの出力ファイルが生成され、チェックするデータベースのシステム識別子(SID)の名前が指定されます。

- SID.html: Webブラウザを使用してこのファイルを表示します。
- SID_preupgrade.log: テキスト・エディタを使用してこのファイルを表示します。

各レポートはアップグレード・エラーを示します。自動修正スクリプトを実行するか手動修正を実行してこれらを訂正しなければエラーが発生します。エラーが発生した場合、ユーザー・ログ・ファイル、およびstatus.jsonファイルにもレポートされます。

分析モードでは、パスcfgtoollogs/upgrade/auto/statusにステータス・ディレクトリも生成されます。このディレクトリには、分析が成功したか失敗したかを示すファイルが含まれています。このディレクトリには、2つのJSONファイルstatus.jsonおよびprogress.jsonがあります。

- status.json : アップグレードの最終ステータスを含む高レベルのステータスJSONファイル。
- progress.json: 構成ファイルのかわりに実行されるすべてのアップグレードの現在の進行状況を含むJSONファイル。エラーが発生した場合、AutoUpgradeを実行しているユーザーのログ・ファイルおよびstatus.jsonファイルにレポートされます。

ターゲット・データベースのOracleホームがサーバーで使用できない場合、構成ファイル内でソースのOracleホーム・パラメータを

同じパスに設定し、AutoUpgrade分析処理モードを実行できるようにする必要があります。たとえば:

```
#
# Source Home
#
sales3.source_home=d:¥app¥oracle¥product¥12.2.0¥dbhome_1
#
# Target Oracle Home
#
sales3.target_home=d:¥app¥oracle¥product¥21.0.0¥dbhome_1
```

以前のリリースのAutoUpgradeでは、`target_home`を設定する必要がありました。AutoUpgradeの以降のリリースでは、この制限は分析モードと修正モードの両方に対して導入されていました。

親トピック: AutoUpgrade処理モードについて

AutoUpgrade修正処理モードについて

AutoUpgrade修正(`fixups`)処理モードでは、データベースを分析し、アップグレードを実行する前に修正する必要がある項目の修正を実行します。

AutoUpgradeを修正モードで実行すると、AutoUpgradeは分析モードでも実行するチェックを実行します。これらのチェックが完了した後、アップグレードを開始する前に、AutoUpgradeによって、修正に必要なすべての自動修正が実行されます。データベースを別のプラットフォームに移動する場合、修正モードを使用すると、データベースがアップグレード用に準備されます。

> 注意:
>
> 修正モードで AutoUpgrade を実行する前に、分析モードで AutoUpgrade を個別に実行することをお薦めします。修正モードでは、ソース・データベースを変更できます。

アップグレードの準備の一環として、アップグレード中にエラーの原因となるような状態をソース・データベースで修正する必要がある場合には、修正モードでAutoUpgradeを実行することでソース・データベースに対して自動修正が実行されます。修正モードでのAutoUpgradeの実行は別のシステムに移動する際に実行するステップであるため、保証付きリストア・ポイントは作成されません。このモードは通常の営業時間外で実行することをお薦めします。

次の構文を使用して、修正モードでAutoUpgradeを開始します。`Java-8-home`は、Java 8配布の場所、またはJava 8ホームに設定されている環境変数です。

```
Java-8-home/bin/java -jar autoupgrade.jar -config yourconfig.txt -mode fixups
```

Java 8がソースOracleホームにある場合、次の構文を使用して修正モードでAutoUpgradeを開始します。ここで、`Oracle_home`はOracleホーム・ディレクトリまたはOracleホームに設定された環境変数で、`yourconfig.txt`は構成ファイルです。

```
Oracle_home/jdk8/bin/java -jar autoupgrade.jar -config yourconfig.txt -mode fixups
```

AutoUpgradeは修正モードで実行されると、分析モードで実行する場合と同じ事前チェックを実行して開始されます。次に、アップグレードの準備でソース・データベースで自動修正を実行し、修正操作の成功または失敗を示す高レベルのステータス・ファイルを生成します。エラーが発生した場合、エラーはAutoUpgradeを実行しているユーザーのログ・ファイルにレポートされます。

注意:

修正モードの AutoUpgrade では、保証付きリストア・ポイントは作成されません。修正モードで AutoUpgrade を開始する前に、データベースがバックアップされていることを確認してください。

親トピック: AutoUpgrade処理モードについて

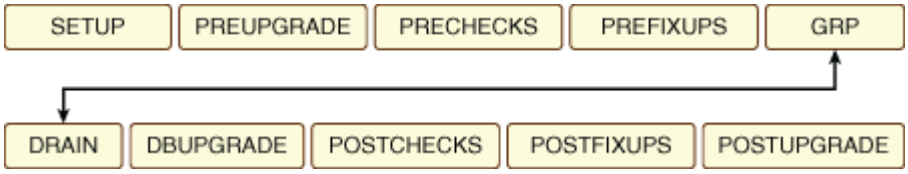
AutoUpgradeデプロイ処理モードについて

AutoUpgradeデプロイ(deploy)処理モードでは、データベースの実際のアップグレードが実行され、保留中のすべての修正が実行されます。

デプロイを実行する前に、ターゲットのOracleホームが事前にインストールされている必要があります。また、バックアップ計画を準備してから、AutoUpgradeスクリプトの一部としてバックアップ計画を実行する必要があります。

次の構文を使用して修正モードでAutoUpgradeを起動します。ここで、Oracle_homeはOracleホーム・ディレクトリまたはOracleホームに設定された環境変数で、yourconfig.txtは構成ファイルです。

```
Oracle_home/jdk8/bin/java -jar autoupgrade.jar -config yourconfig.txt -mode deploy
```



AutoUpgradeをデプロイ・モードで実行すると、AutoUpgradeは、アップグレード前のソース・データベース分析からアップグレード後のチェックまで、データベース上ですべてのアップグレード操作を実行します。各操作は、次の操作の準備を行います。エラーが発生した場合、操作は停止します。エラーはすべて、関連するログ・ファイルおよびコンソール(有効な場合)に記録されます。操作の成功または失敗を示す、操作ごとの高レベルのステータス・ファイルが生成されます。まだ保留中の修正がある場合(たとえば、最初に分析および修正モードでAutoUpgradeを実行せずにデプロイ・モードでAutoUpgradeを実行する場合)、AutoUpgradeがデプロイ・モードで修正を完了することがあります。

親トピック: AutoUpgrade処理モードについて

AutoUpgradeアップグレード処理モードについて

AutoUpgradeアップグレード(upgrade)処理モードでは、ソースまたはターゲットのOracleホームをアップグレードできます。

アップグレード・モードを使用すると、アップグレードを次の2つの部分に分けることができます。

1. (強く推奨)Oracleホームで実行されているソース・データベースで事前修正モードを実行します。
2. (オプション)ソース・データベースを別のシステムの新しいOracleホームに移動します。
3. アップグレード・モードを使用してデータベースのアップグレードを実行します。

ノート:

ソース Oracle ホームで実行すると、AutoUpgrade は PRECHECKS ステージおよび PREFIXUPS ステージをスキップした直後にアップグレードの処理を開始します。他のすべてのステージ(POSTUPGRADE を

除く)は、通常 DEPLOY 中に実行されます。

アップグレード・モードを使用するには、AutoUpgradeをアップグレード・モードで実行する前に、ソースまたはターゲットOracleホームでデータベースが稼働している必要があります。このオプションは、AutoUpgradeデプロイ・モードを使用できないように、Oracle Databaseを元のソース・システムから別のシステムに移動したときに特に役立ちます。

このプロシージャは、新しいOracleホームの場所のデータベースでアップグレード、アップグレード後のチェックおよび事後修正の操作を実行します。

ターゲットOracleホームでのAutoUpgradeアップグレード・モード

ターゲットOracleホームでデータベースがオープンしている場合、プロセス・フローは次のようになります。

AutoUpgradeがアップグレード・モードで実行されると、AutoUpgradeスクリプトを実行するユーザーのログ・ファイルにエラーが記録されます。操作の成功または失敗を示す、操作ごとの高レベルのステータス・ファイルが生成されます。

ノート:

アップグレード・モードで AutoUpgrade を実行すると、アップグレード後の操作は実行されないため、これらのステップを個別に完了する必要があります。たとえば、次のアップグレード後操作は実行されません。

- ネットワーク・ファイルのコピー(`tnsnames.ora`、`sqlnet.ora`、`listener.ora` およびその他のリスナー・ファイル、LDAP ファイル、`oranfstab`)
- アップグレード中に作成された保証付きリストア・ポイント(GRP)の削除
- Oracle Real Application Clusters データベースの最終再起動

ソースOracleホームでのAutoUpgradeアップグレード・モード

ソースOracleホームでデータベースがオープンしている場合、アップグレード・ホームで実行されるステージは、アップグレード・モードでAutoUpgradeを起動する前にソースOracleホームで修正が実行されているかどうかによって異なります。

- ソースOracleホームで修正がすでに実行されている場合は、事前チェックおよび事前修正を除く、一般的なデプロイ・モードのすべてのステージが実行されます。

  このオプションは、事前チェックおよび事前修正を個別に実行できる場合に使用します。これは、AutoUpgradeは、アップグレード中には事前チェックおよび事前修正ステージの実行をバイパスし、停止時間を減らすためです。

- ソースOracleホームでの修正が過去3日以内に実行されていない場合、アップグレード・モードにそれらのステージが含まれます。その結果、ソースOracleホームでAutoUpgradeをアップグレード・モードで実行することは、デプロイ・モードでAutoUpgradeを実行することとまったく同じです。これは、事前チェック・ステージおよび事前修正ステージはアップグレード・モードの一部として実行されるためです。

例3-1 新しい場所へのデータベースの移動後のターゲット・ホームでのAutoUpgradeの実行

ここで、`dbname`はデータベースの名前で、次のステップを使用してAutoUpgradeを実行します。

- AutoUpgradeを事前修正モードで実行した場合:
  a. デフォルト名(`initSID.ora`)を使用して、`during_upgrade_pfile_dbname.ora`ファイルをターゲットOracleホームのデフォルトの場所にコピーします。

during_upgrade_pfile_dbname.oraファイルは、AutoUpgradeの実行に使用されるログ・パスのtempディレクトリにあります。

b. (オプション) SQL*Plusに接続し、tempディレクトリのduring_upgrade_pfile_dbname.oraを使用してSPFILEを作成できます。たとえば:

```
SQL> create spfile from
pfile='/u01/autoupgrade/au21/CDBUP/temp/during_upgrade_pfile_cdbupg
.ora';
```

- AutoUpgradeを事前修正モードで実行しなかった場合:
  a. 初期化ファイル(init.oraまたはspfileSID.ora)をソースOracleホームからターゲットOracleホームの場所にコピーします。

2. 次の構文を使用してアップグレード・モードでAutoUpgradeを実行します。ここで、Oracle_homeはOracleホーム・ディレクトリ・パスまたはOracleホームに設定された環境変数で、yourconfig.txtは構成ファイルです。

```
Oracle_home/jdk8/bin/java -jar autoupgrade.jar -config yourconfig.txt -mode
upgrade
```

このコマンドでは、データベースでアップグレード操作を実行します。

親トピック: AutoUpgrade処理モードについて

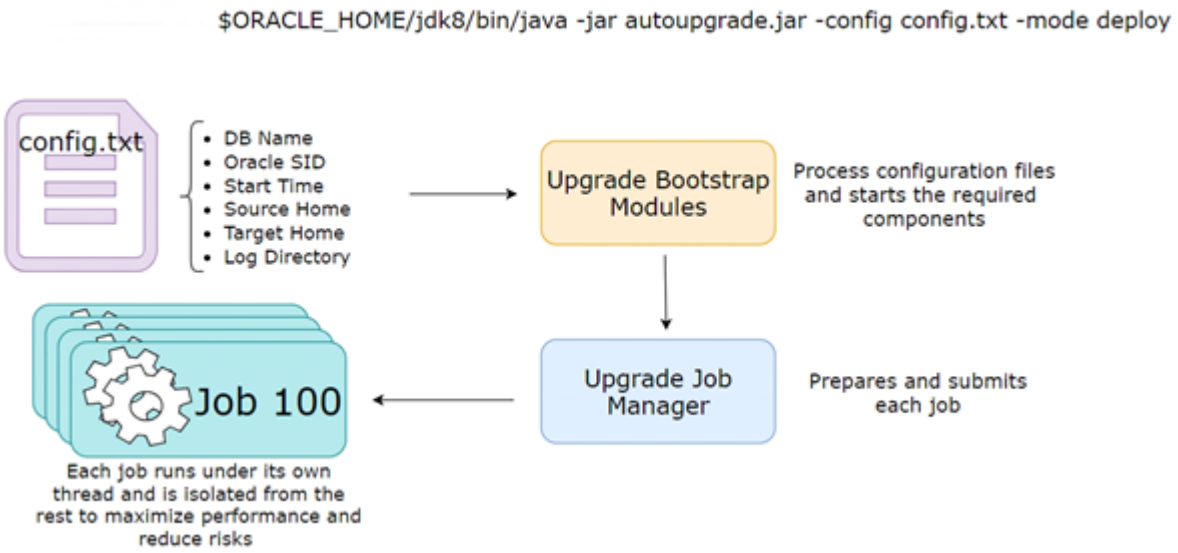
AutoUpgradeワークフローおよびステージの理解

AutoUpgradeワークフローは、一般的なアップグレード・プロセスの各ステップを自動化します。実行されるステージは、選択した処理モードによって異なります。

AutoUpgradeは、できるだけ人間が介入しないでアップグレードを実行できるように設計されています。AutoUpgradeを起動すると、コマンドで指定した構成ファイルがAutoUpgradeジョブ・マネージャに渡されます。ジョブ・マネージャは、選択した処理モードに必要なジョブを作成し、モードに必要なデータ構造をディスパッチャに渡します。ディスパッチャは、個々のタスクを実行する下位レベルのモジュールを起動します。

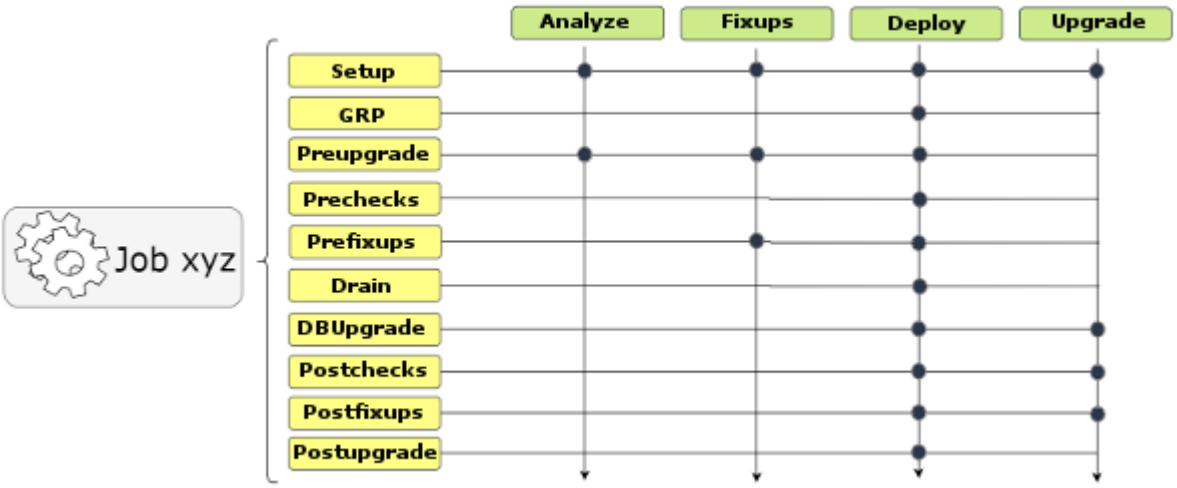
AutoUpgrade処理モードのワークフロー処理

AutoUpgradeがワークフロー・モードを処理する方法を理解するには、次の図を確認してください。これは、デプロイ処理モードの処理方法を示しています。



AutoUpgrade処理モードのステージ

アップグレード・ジョブに対してAutoUpgradeが実行されるステージは、選択した処理モードによって異なります。



AutoUpgradeには、次の4つのモードがあります。各モードでは、AutoUpgradeのステップが順に実行されます。各モードのステップの違いに注意してください

- **分析**モード: 設定、事前チェック。
- **修正**モード: 設定、事前チェックおよび事前修正。
- **デプロイ**・モード: 設定、保証付きリストア・ポイント(GRP)、アップグレード前処理、事前チェック、事前修正、排出、DB(データベース)のアップグレード、事後チェック、事後修正およびアップグレード後処理。アップグレード前(アップグレード前のステージ)またはアップグレード後(アップグレード後ステージ)、あるいはアップグレードの前後に、独自のスクリプトを実行できます。
- **アップグレード**・モード: 設定、DB(データベース)のアップグレード、事後チェックおよび事後修正。

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

AutoUpgradeを使用した非CDBからPDBへのアップグレードの理解

1回の操作で非CDBを新しいCDBのPDBにアップグレードおよび変換するか、非CDBデータベースを既存のCDBのPDBにアップグレードしてから変換できます。

Oracle Database 21cへのすべてのアップグレードで、マルチテナント・アーキテクチャを使用する必要があります。非CDB Oracle Databaseアーキテクチャの使用はサポートされなくなりました。非CDBアーキテクチャからPDBにデータベースを移行する場合、マルチテナント・ライセンスを必要とせずに、コンテナ・データベース(CDB)で最大3つのユーザー構成可能なPDBを取得します。4つ以上のPDBを構成することを選択する場合は、マルチテナント・ライセンスが必要です。

AutoUpgradeユーティリティの非CDBからPDBへの機能には、マルチテナント・アーキテクチャへのアップグレードおよび変換時に以前のリリースの非CDB Oracle Databaseをアップグレードする方法を制御する柔軟なオプションが用意されています。Oracle Database 21c以降、既存のターゲット・リリースのCDBがある場合、AutoUpgradeを使用して、アップグレード中に非CDBのOracle Databaseをターゲット・リリースのCDBのPDBに変換できるようになりました。非CDBからPDBへのアップグレードおよび変換を実行するには、非CDBに関する情報をAutoUpgrade構成ファイルに指定します。必要に応じて、非CDBのOracle Databaseをソース・リリースのPDBに変換し、PDBをターゲット・リリースのCDBに接続することもできます。この場合、PDBを接続すると、アップグレードが実行されます。

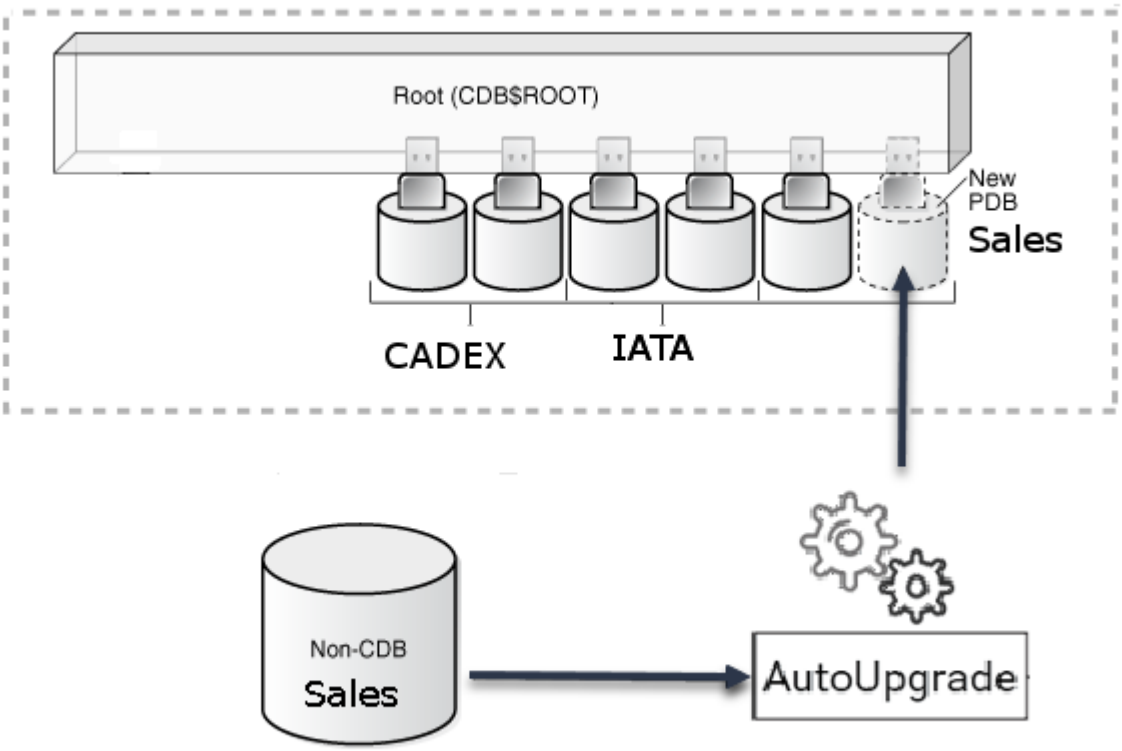
アップグレード後、データベース・リスナーおよびローカル・ネーミング・パラメータ(`tnsnames.ora`ファイル)を構成する必要があります。

注意:

変換およびアップグレードを完了するために AutoUpgrade を実行する前に、ソース・データベースの完全なバックアップを作成し、アップグレードの完全なテストを完了することをお薦めします。AutoUpgrade がこのプロシージャを開始した後、非 CDB の Oracle Database 状態にロールバックするオプションはありません。

図3-1 AutoUpgradeを使用した非CDBの変換およびアップグレード

この図では、Salesという非CDB Oracle Databaseが変換され、SalesというPDBにアップグレードされます。



ソース非CDBおよびターゲットCDBの要件

PDBへのアップグレードおよび変換を実行するためのソース非CDBおよびターゲットCDBの要件は、次のとおりです。

- AutoUpgradeを使用してアップグレードを実行する前に、ターゲットCDBを作成する必要があります。
- 非CDBから作成されたPDBは、ソース非CDB名を引き続き使用する必要があります。データベースの名前は変更できません。
- ソースとターゲットの両方に同じOracle Databaseオプションのセットが構成されていること。
- ソースCDBとターゲットCDBのエンディアン形式が同じであること。
- ソースCDBとターゲットCDBに、互換性のある文字セットと各国語文字セットがあること。
- ソース非CDBのOracle Databaseリリースおよびオペレーティング・システム・プラットフォームでは、ターゲットCDBリリースへの直接のアップグレードがサポートされている必要があります。
- ソースCDBおよびターゲットCDBに対してオペレーティング・システム認証が有効になっていること

ソース・データベースの最小`COMPATIBLE`パラメータ設定は12.2.0以上である必要があります。`COMPATIBLE`設定が低いバージョンの場合、変換およびアップグレード・プロセス中に`COMPATIBLE`が12.2.0に設定されます。変換中、元のデータファイルは保持されます。これらは、新しいPDBを作成するためにコピーされません。AutoUpgradeでアップグレードを実行できるようにするには、AutoUpgrade構成ファイルを編集して、AutoUpgradeパラメータ`target_version`をターゲットCDBリリースに設定し、`target_cdb`を使用して、アップグレードしたデータベースが配置されるCDBを識別します。変換およびアップグレード・プロセスでは、AutoUpgradeはその情報を使用してターゲットCDBへのアップグレードを完了します。

例3-2 非CDBからPDBへの変換のためのAutoUpgrade構成ファイル

非CDBからPDBオプションを使用するには、AutoUpgrade構成ファイルでパラメータ`target_cdb`を設定する必要があります。`target_cdb`パラメータの値は、非CDBのOracleデータベースを接続するコンテナ・データベースのOracleシステム識別子(SID)を定義します。たとえば:

```
global.autoupg_log_dir=/home/oracle/autoupg
upg1.sid=s12201
upg1.source_home=/u01/product/12.2.0/dbhome_1
upg1.log_dir=/home/oracle/autoupg
upg1.target_home=/u01/product/20.1.0/dbhome_1
upg1.target_base=/u01
upg1.target_version=19.1.0
upg1.target_cdb=cdb19x
```

関連項目

- Oracle Database製品で許可される機能、オプションおよび管理パック

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

AutoUpgradeを使用した切断/接続アップグレードの理解

AutoUpgradeでは、以前のリリースのソース・コンテナ・データベース(CDB)からプラガブル・データベース(PDB)を切断し、新しいリリースのターゲットCDBに接続して、PDBをターゲットCDBリリースにアップグレードするために必要なすべてのステップを完了できます。

アップグレードの構成方法に応じて、AutoUpgradeを使用した切断/接続PDBアップグレードには2つのワークフローがあります。

- 1つ以上のプラガブル・データベースを1つのソースCDBから切断し、新しいリリースのターゲットCDBに接続します。
- 異なるソースCDBから複数のプラガブル・データベースを切断し、新しいリリースのターゲットCDBに接続します。

また、切断/接続操作の場合、AutoUpgradeでは、ソースPDBからターゲットPDBへのPDBのデフォルト状態の移動がサポートされるようになりました。ソースPDBで`alter pluggable database save state`を設定すると、その状態がターゲットPDBに転送されるため、CDB$ROOTがオープンされるとPDBが自動的にオープンされます。

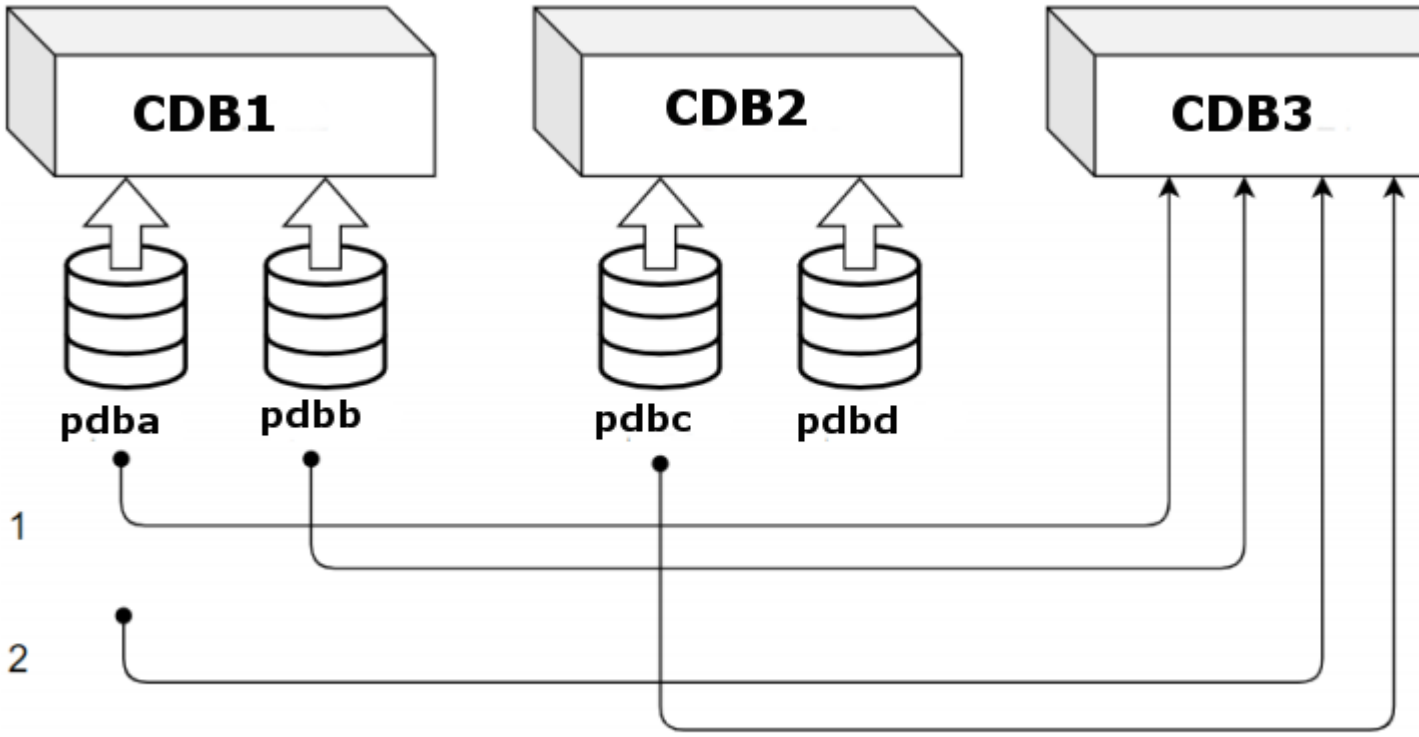
注意:

データベースに対するその他の変更と同様に、AutoUpgrade を実行して変換およびアップグレードを完了する前に、予期しないデータ損失を防ぐために信頼性のあるバックアップ計画を実装することをお薦めします。AutoUpgrade がこの手順を開始した後、切断/接続 PDB アップグレードをロールバックするオプションはありません。また、フラッシュバック・データベースは、PDB 変換では動作せず、元に戻せません。バックアップは唯一のフォールバック計画です。

次の図は、この単一PDBの場合の切断/接続操作を示しています。

1. ソースOracle Databaseが1つとターゲット・リリースのOracle Databaseが1つあります。この段階で、構成ファイルを作成し、AutoUpgradeを分析モード(`autoupgrade.jar -mode analyze`)で実行してアップグレードの準備状況を確認し、レポートされた問題を修正します。
2. AutoUpgradeをデプロイ・モード(autoupgrade.jar -mode deploy)で実行します。AutoUpgradeは、構成ファイルに指定された情報を使用してPDBをターゲット・リリースに移動し、PDBを接続します。
3. AutoUpgradeは、事前修正を実行した後、PDBをターゲット・リリースにアップグレードします。

図3-2 ソースからターゲットへの切断/接続アップグレード



ソースCDBとターゲットCDBの要件

切断/接続アップグレードを実行するには、ソースCDBとターゲットCDBが次の条件を満たしている必要があります。

- 切断/接続アップグレードを開始する前に、ターゲット・リリースCDBを作成し、CDBをオープンしていること。
- ソースCDBとターゲットCDBのエンディアン形式が同じであること。
- ターゲット・リリースCDB用に構成されたOracle Databaseコンポーネントのセットに、ソースCDBで使用可能なすべてのコンポーネントが含まれていること。
- ソースCDBとターゲットCDBに、互換性のある文字セットと各国語文字セットがあること。
- ターゲットCDBリリースへの直接のアップグレードでは、ソースCDBリリースがサポートされている必要があります。
- ソースCDBおよびターゲットCDBに対して外部認証(オペレーティング・システム認証)が有効になっていること
- ソースCDBのOracle APEXインストール・タイプは、ターゲットCDBのインストール・タイプと一致する必要があります。
- CDBに接続する非CDB Oracle Databaseには、既存の保証付きリストア・ポイント(GRP)を含めないでください。

ノート:

AutoUpgrade 22 以降の更新では、AutoUpgrade を使用して Oracle Data Guard 構成に接続できるようになりました。AutoUpgrade は、STANDBYS=NONE 句を使用して PDB を作成します。アップグレード後、データ・ファイルをスタンバイ・データベースにリカバリすることでスタンバイを再確立できます。

切断/接続アップグレードの機能

切断/接続アップグレードを選択すると、AutoUpgrade構成ファイルの構成方法に応じて、AutoUpgradeを使用してアップグレード時に次のオプションを実行できます。

- ソースCDBにあるPDB名を保持するか、PDB名を変更できます。
- 古いファイルをすべて保持しながら、データ・ファイルのコピーをターゲットCDBに作成できます。
- データファイルをターゲットの場所にコピーしてから、ソースCDBで古いファイルを削除できます。
- 1つのPDBを処理することも、包含リストにリンクして多数のPDBを1つのアップグレード手順で処理することもできます。処理できるPDBの数の制限は、サーバー制限とCDB上のPDBの制限のみです。

例3-3 切断/接続アップグレード用のAutoUpgrade構成ファイル

PDB切断/接続アップグレード・オプションを使用するには、AutoUpgrade構成ファイル内で次の値を指定する必要があります。

- ソースCDBのシステム識別子パラメータ(パラメータ`sid`)。
- ターゲットCDB (パラメータ`target_cdb`)。
- ソースCDB内のPDBの名前、およびそれを変換する場合はターゲット変換名。

たとえば、ソースCDBが`CDB122`、ターゲットCDBが`cdb21x`、ソースCDB内のPDBの名前が`pdb2`、ターゲットCDBに必要なPDBの変換名が`depsales`の場合は次のようになります。

```
global.autoupg log_dir=/home/oracle/autoupg
upg1.sid=CDB122
upg1.source_home=/u01/app/oracle/product/12.2.0/dbhome_1
upg1.target_home=/u01/app/oracle/product/19.1.0/dbhome_1
upg1.target_cdb=cdb21x
upg1.pdbs=pdb_2
upg1.target_pdb_name.pdb_2=depsales
upg1.target_pdb_copy_option.pdb_2=file_name_convert=('pdb_2','depsales')
```

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

AutoUpgradeのコマンドライン・パラメータおよびオプション

AutoUpgradeパラメータを確認し、Oracle Databaseのアップグレードのユース・ケースのパラメータおよびオプションを選択します。

パラメータは、`java -jar autoupgrade.jar`コマンドとともに使用します。

- AutoUpgradeコマンドライン構文
  AutoUpgradeを使用してアップグレードを実行する方法を確認するには、構文および実行時のユースケースを確認してください。
- debug
  AutoUpgradeパラメータ`debug`により、AutoUpgradeデバッグ・メッセージ機能をオンにします。これは、正しくないAutoUpgradeジョブ構文の修正に役立ちます。
- clear_recovery_data
  AutoUpgradeパラメータ`clear_recovery_data`により、リカバリ・チェックポイントを削除します。これにより、AutoUpgradeは、指定されたデータベースまたはすべてのデータベースで次回ツールが起動されたときに、新しく起動されます。
- config
  AutoUpgradeパラメータ`config`により、アップグレードするデータベースについて情報を提供するために使用する、構成ファイルを特定します。
- config_values
  AutoUpgradeパラメータ`config_values`を使用すると、テキスト構成ファイルと同様の内容で、システムに関する入力値を指定できます。このパラメータは、`config`パラメータと組み合せて使用できます。
- console
  AutoUpgradeパラメータ`console`により、AutoUpgradeコンソールをオンにし、AutoUpgradeジョブの進行状況を監視するための一連のコマンドが提供されるようにします。
- create_sample_file
  AutoUpgradeパラメータ`create_sample_file`により、構成ファイルまたは設定ファイルを生成します。これらのファイルを編集して、本番構成ファイルまたはAutoUpgrade用設定ファイルを作成します。
- error_code
  AutoUpgradeパラメータ`error_code`により、AutoUpgradeエラーのエラー・コードが表示されるようにします。
- listchecks
  AutoUpgradeパラメータ`listchecks`は、アップグレードのすべてのアップグレード・チェックのリストを提示するか、特定のチェックが指定された場合は、指定されたチェックに関する詳細を提示します。
- load_password
  AutoUpgradeパラメータ`load_password`を使用すると、AutoUpgradeのキーストアにパスワードを安全に入力できます。
- mode
  AutoUpgradeパラメータ`mode`の値により、AutoUpgradeの実行モードを設定します。
- noconsole
  AutoUpgradeパラメータ`noconsole`により、AutoUpgradeコンソールをオフにし、構成ファイル情報のみを使用してAutoUpgradeが実行されるようにします。
- preupgrade
  AutoUpgradeパラメータ`preupgrade`により、データベース・チェックと、ほとんどの問題をアップグレード開始前に修

正するアップグレード前修正と、ほとんどの問題をアップグレード完了後に修正するアップグレード後修正を実行します。

- settings
  AutoUpgradeパラメータ`settings`により、AutoUpgradeユーティリティのカスタム・ランタイム構成の提供に使用する構成ファイルを特定します。
- version
  AutoUpgradeパラメータ`version`により、`autoupgrade.jar`ファイルの現在のビルドを端末画面に出力します。
- restore
  AutoUpgradeパラメータ`restore`により、指定したAutoUpgradeジョブのシステムレベル・リストアを実行します。
- restore_on_fail
  AutoUpgradeパラメータ`restore_on_fail`により、デプロイメント中に失敗したジョブが自動的にリストアされるようにします。
- zip
  AutoUpgradeパラメータ`zip`により、AutoUpgradeサービス・リクエストのファイリングに必要なログ・ファイルのzipファイルを作成します。

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

AutoUpgradeコマンドライン構文

AutoUpgradeを使用してアップグレードを実行する方法を確認するには、構文および実行時のユースケースを確認してください。

前提条件

- ソース環境にJava Development Kit (JDK) 8以降がインストールされている必要があります。

  JDK 8は、Oracle Database 12cリリース2 (12.2)以降のリリースごとにインストールされます。12.2より前のリリースでは、ターゲットOracle DatabaseのJavaリリースを使用してAutoUpgradeを実行するか、ソース・データベース・サーバーにJDK 8をインストールする必要があります。

- AutoUpgradeユーティリティを使用したOracle Databaseのアップグレードは、Oracle Databaseの手動アップグレードに適用されるアップグレード・ルールと同じルールに従います。ソースOracle Databaseのリリースがアップグレードでサポートされていることを確認してください。

非CDBからPDBへの変換およびアップグレードでは、AutoUpgradeは、次の条件が満たされた場合に、アップグレードと変換の両方を自動的に完了することがあります。

- ターゲット・リリースCDBが存在する必要があります。
- ターゲットCDBシステム識別子が`target_cdb`であるAutoUpgrade構成ファイルで、次の構文を使用してローカル・パラメータ`target_cdb`を設定する必要があります。

  `target_cdb=target_cdb`。たとえば:

  ```
  target_cdb=cdb1
  ```

- `target_cdb`値は、非CDBを接続するCDBのOracle SIDです。

ファイル・パス

AutoUpgradeユーティリティは、新しいリリースのOracle DatabaseホームにあるJava JARファイルです。

`Oracle_home/rdbms/admin/autoupgrade.jar`

My Oracle Supportから最新のAutoUpgrade JARファイルを取得することをお薦めします。JARファイルおよびJARファイルのデプロイメント手順は、My Oracle Supportノート2485457.1から入手できます

構文

AutoUpgradeコマンドの構文では、大文字と小文字が区別されます。コマンドを小文字で入力します。

```
java -jar autoupgrade.jar [options]
```

複数のオプションを連結できます。

AutoUpgradeの実行タイプ1 (基本)のパラメータ

AutoUpgradeの実行タイプ1 (基本)のパラメータおよびオプションにより、アップグレードの準備の開始点が提供されます。

パラメータ	説明
`-version`	AutoUpgrade のバージョンを表示します。
`-help`	AutoUpgrade 構文のヘルプ・ファイルを表示します。
`-create_sample_file [settings | config config-file-name]`	AutoUpgrade のサンプル構成ファイルを作成します。オプションの詳細は、`create_sample_file` パラメータのトピックを参照してください。

AutoUpgradeの実行タイプ2 (コア)のパラメータ

AutoUpgradeの実行タイプ2 (コア)のパラメータおよびオプションは、ほとんどのアップグレード・シナリオで重要なアップグレード機能を提供します。

パラメータ	説明
`-config [config_path | -config_values config_values]`	アップグレードするデータベースに関する情報を提供するために使用する構成ファイルを識別します。オプションの詳細は、`config` パラメータのトピックを参照してください。
`-mode [analyze|fixups|deploy|upgrade|postfixups]`	AutoUpgrade を実行するモードを設定します。オプションの詳細は、`mode` パラメータのトピックを参照してください。
`-restore -jobs job#`	指定する AutoUpgrade ジョブのシステムレベルのリストアを実行します
`-restore_on_fail`	設定されている場合、ジョブが失敗すると、データベースは自動的にリストアされます。PDB のエラーは致命的とはみなされず、`CDB$ROOT` または非 CDB のエラーのみが致命的とみなされます。
`-console`	コンソールを有効にして AutoUpgrade を開始します。

パラメータ	説明
-noconsole	コンソールを無効にして AutoUpgrade を開始します。
-debug	デバッグ・メッセージを有効にします。
-clear_recovery_data [-jobs job#,job#,...]	リカバリ情報を削除します。これにより、すべてのデータベースまたは-jobs で指定されたカンマ区切りリストのデータベースで AutoUpgrade が最初から開始されます。オプションの詳細は、clear_recovery_data パラメータのトピックを参照してください。
-restore -jobs job#,job#,...	指定されたジョブのシステムレベルのリストアを実行します。データベースは保証付きリストア・ポイント(GRP)にフラッシュ・バックされます。このコマンドを実行する前に、GRP が AutoUpgrade によってすでに作成されている必要があります。オプションの詳細は、clear_recovery_data パラメータのトピックを参照してください。
-zip [-sid sid] [-d dir]	AutoUpgrade サービス・リクエストのファイリングに必要なログ・ファイルを圧縮します。オプションの詳細は、zip パラメータのトピックを参照してください。

AutoUpgradeの実行タイプ3 (追加)のパラメータ

AutoUpgradeの実行タイプ3 (追加)のパラメータおよびオプションは、アップグレードの失敗したポイントからの再開や特定の修正の実行など、特定のアップグレード・シナリオに役立ちます。

パラメータ	説明
-debug	デバッグ・メッセージを有効にします。
-error_code	AutoUpgrade エラー・コードが表示されます。
-help	AutoUpgrade 構文のヘルプ・ファイルを表示します。
-mode [analyze|fixups|postfixups]	AutoUpgrade を実行するモードを設定します。オプションの詳細は、mode パラメータのトピックを参照してください。
-load_password	AutoUpgrade が必要とするパスワードを AutoUpgrade のキーストアに安全に入力できるようにします。
-preupgrade preupgrade_options options	データベース・チェックおよびほとんどの問題をアップグレードの開始前に修正するアップグレード前修正と、ほとんどの問題をアップグレードの完了後に修正するアップグレード後修正を実行します。オプションの詳細は、preupgrade パラメータのトピックを参照してください。

関連項目

- My Oracle Supportノート2485457.1

- 直接のアップグレードをサポートするOracle Databaseのリリース

親トピック: AutoUpgradeのコマンドライン・パラメータおよびオプション

debug

AutoUpgradeパラメータdebugは、AutoUpgradeデバッグ・メッセージ機能をオンにします。これは、障害のあるAutoUpgradeジョブ構文の修正に役立ちます。

プロパティ	説明
パラメータ・タイプ	文字列
構文	`autoupgrade.jar -parameter -debug`

説明

AutoUpgradeのdebugパラメータは、AutoUpgradeコマンド構文の修正に役立つデバッグ・メッセージをオンにします。

使用上のノート

debugパラメータは、他のAutoUpgradeパラメータと組み合せて使用します。

親トピック: AutoUpgradeのコマンドライン・パラメータおよびオプション

clear_recovery_data

AutoUpgradeパラメータclear_recovery_dataは、リカバリ・チェックポイントを削除します。これにより、AutoUpgradeは、指定されたデータベースまたはすべてのデータベースで次回ツールが起動されたときに、新しく起動します。

プロパティ	説明
パラメータ・タイプ	文字列
構文	`clear_recovery_data [-jobs job_numbers]` ここで、 `job_numbers` は、クリアするジョブのカンマ区切りリストです

説明

AutoUpgradeのclear_recovery_dataパラメータは、リカバリ情報を削除します。これにより、指定したデータベースまたはすべてのデータベースでAutoUpgradeが最初から開始されます。

使用上のノート

データベースを手動でリストアし、新しいアップグレードを試行した後に使用します。ジョブのリストが指定されていない場合は、デフォルトですべてのジョブ・メタデータが削除されます。メタデータを削除しても、ログ・ファイルは削除されず、ジョブ識別子(jobid)カウンタもリセットされません。各ジョブの進行状況を追跡するために使用されるAutoUpgradeファイルのみが削除されます。

例

次の例は、問題が発生した後にclear_recovery_dataオプションを使用して修正し、AutoUpgradeを再度実行する方法を示しています。

AutoUpgradeをデプロイ・モードで実行します

```
java -jar autoupgrade.jar -config config.cfg -mode deploy
```

ただし、アップグレード中に問題が発生しました。AutoUpgradeを停止してデータベースをリストアし、データベースを変更して問題を修正します。AutoUpgradeプロシージャを再実行して現在のジョブ状態情報をクリアするには、以前に実行したジョブに関連付けられているジョブ番号を指定します。ジョブ番号を指定すると、AutoUpgradeはその特定のジョブの状態情報のみを削除します。残りのジョブは変更されません。

```
java -jar autoupgrade.jar -config config.cfg -mode analyze -clear_recovery_data -jobs
100
```

ノート:

ジョブ ID 番号は、ジョブをデータベースに関連付けます。間違ったジョブ ID を入力すると、AutoUpgrade は間違ったジョブを最初から再開します。

分析結果に問題はないため、再度デプロイ・オプションを実行します。

```
java -jar autoupgrade.jar -config config.cfg -mode deploy
```

-clear_recovery_dataパラメータを指定してautoupgrade.jar -configを実行すると、AutoUpgradeによって状態ファイルのみが削除されます。以前生成されたログ・ファイルは無視されるため、さらに参照するためにログ・ファイルを保持できます。-clear_recovery_dataパラメータを指定してAutoUpgradeを実行すると、最新のjobid情報も保持されるため、AutoUpgradeが次のジョブに対して作成するjobidは、次の順番のIDとなります。AutoUpgradeがjobid状態を保持することで、同じログ・ファイル内に前のAutoUpgradeジョブのログ出力を混在させないようにすることができます。

次に、clear_recovery_dataパラメータの実行方法のその他の例を示します。

```
java -jar autoupgrade.jar -config config.cfg -clear_recovery_data
java -jar autoupgrade.jar -config config.cfg -clear_recovery_data -jobs 111,222
```

親トピック: AutoUpgradeのコマンドライン・パラメータおよびオプション

config

AutoUpgradeパラメータconfigは、アップグレードするデータベースに関する情報を提供するために使用する構成ファイルを識別します。

プロパティ	説明
パラメータ・タイプ	文字列
構文	-config [configfile
デフォルト値	なし

説明

configパラメータは構成ファイル名を指定します。これは次の3つの引数を取ります。

- 構成プロパティ名
- (オプション) config-fileで表される構成ファイルへのパス

  パラメータ-load_passwordとともに使用すると、グローバル構成ファイル・パラメータglobal.keystoreがデータベースの構成ファイルに設定されている場合、AutoUpgradeは、そのパラメータで指定された場所にTransparent Data Encryption (TDE)パスワードのキーストアも作成します。

例

処理モードdeployで、myconfig.cfgという名前の構成ファイルを指定してAutoUpgradeを実行します。

```
java -jar autoupgrade.jar -config myconfig.cfg -mode deploy
```

親トピック: AutoUpgradeのコマンドライン・パラメータおよびオプション

config_values

AutoUpgradeパラメータconfig_valuesを使用すると、テキスト構成ファイルと同じシステムに関する入力値を指定できます。このパラメータは、configパラメータと組み合せて使用できます。

プロパティ	説明
パラメータ・タイプ	文字列。
構文	-config_values [config-parameter1=value*,config-parameter2=value,...]
デフォルト値	なし。

説明

config_valuesパラメータを使用すると、使用しない場合は構成ファイルを指定する必要があるデータベース・パス、インスタンスおよびターゲット・リリースに関する値をAutoUpgradeコマンドラインを介して指定できます。その後、AutoUpgradeによってユーティリティの実行時に構成ファイルが作成されます。config_valuesを使用することで、構成ファイルなしにAutoUpgradeを実行できます。

config_valuesオプションは、複数のデータベース・アップグレードをサポートできるカンマ区切りリストです。各データベース構成は、それぞれのデータベースを識別するためにアスタリスク(*)で区切られます。グローバル・エントリの名前には、global接頭辞が含まれている必要があります。たとえば:

```
global.autoupg_log_dir=/u01/app/oracle/cfgtoollogs/upgradelogs/
```

ローカル・エントリには、名前のみが含まれている必要があります。

```
target_home=/u01/app/oracle/product/21.0.0.0/dbhome_1
```

ロギング・ディレクトリは、次の方法で解決されます。

- ケース: グローバルのautoupg_log_dirが指定されていない場合。

  config_fileパラメータがAutoUpgradeに渡されない場合、ローカル・ディレクトリがグローバル・ログ・ディレクトリとして使用されます。config_fileパラメータがAutoUpgradeに渡されない場合、グローバル・ログ・ディレクトリのデ

フォルトはJava一時ディレクトリになります。

- UNIXおよびLinuxシステム: `/tmp/autoupgrade`
- Microsoft Windows: `C:¥Users¥name¥AppData¥Local¥Temp¥autoupgrade`
- `autoupgradeYYYYMMMHHMMSS.cfg`という名前の構成ファイルが作成されます。ここで、`YYYY`は年、`MMM`は月、`HH`は時、`MM`は分、`SS`は秒です。

- ケース: グローバルの`autoupg_log_dir`が指定されている場合。

  `config_file`パラメータがディレクトリをAutoUpgradeに渡さない場合、AutoUpgradeは、パラメータで指定されたAutoUpgradeログ・ディレクトリに構成ファイルを作成します。`config_file`パラメータがディレクトリをAutoUpgradeに渡さない場合、構成ファイルはグローバル・ログ・ディレクトリ下に作成されます。既存の構成ファイル名を指定すると、AutoUpgradeは、接尾辞`YYYYMMMHHMMMMSS.cfg` (`YYYY`は年、`MMM`は月、`HH`は時間、`MM`は分、`SS`は秒)を使用して、既存の構成ファイルの名前を変更します。たとえば、2020年4月29日の午前8時30分4秒に、構成ファイル`¥tmp¥autoupgrade.cfg`がすでに存在する場合、ファイル名`-config_file ¥tmp¥autoupgrade.cfg`をAutoUpgradeに渡すと、既存のファイルの名前は`¥tmp¥autoupgrade.cfg20200429083004`に変更されます。その後、AutoUpgradeは新しい構成ファイル`¥tmp¥autoupgrade.cfg`を作成します。

`-config_values`パラメータを使用し、AutoUpgradeコマンドを実行するユーザー・アカウントに次のオペレーティング・システム環境変数が設定されている場合、AutoUpgradeでは、これらの変数に定義されているパスを取得します。

- `ORACLE_HOME` - ソースOracleホームのOracleホーム・パス。
- `ORACLE_TARGET_HOME` - ターゲットOracleホーム・パス。
  - LinuxおよびUnix: `export ORACLE_TARGET_HOME`コマンドと同等です。例: `export ORACLE_TARGET_HOME=/u01/app/oracle/product/19.0.0/`
  - Microsoft Windows: `SET ORACLE_TARGET_HOME`コマンドと同等です。例: `SET ORACLE_TARGET_HOME=C:¥oracle¥19`
- `ORACLE_SID` - Oracle Databaseシステム識別子(SID)。
  - LinuxおよびUnix: オペレーティング・システムのシェル・コマンド`export ORACLE_SID`を使用して設定します。たとえば: `export ORACLE_SID=sales`
  - Microsoft Windows: オペレーティング・システムのシェル・コマンド`SET ORACLE_SID`を使用して設定します。たとえば: `SET ORACLE_SID=sales`
- `ORACLE_TARGET_VERSION` - 新しいOracleホームのターゲット・リリース。このオペレーティング・システム環境変数は、ターゲットOracleホームが存在しない場合、またはターゲット・ホームがOracle Database 18cより前のリリースのときに設定する必要があります。
  - LinuxおよびUNIX: `export ORACLE_TARGET_VERSION`を使用して設定します。たとえば、Oracle Database 19cの場合は、次のように入力します。

    ```
    export ORACLE_TARGET_VERSION=19.1
    ```

    Oracle Database 21c:の場合

    ```
    export ORACLE_TARGET_VERSION=21.1
    ```

  - Microsoft Windows: `SET ORACLE_TARGET_VERSION`を使用して設定します。

    たとえば、Oracle Database 19cの場合は、次のように入力します。

    ```
    SET ORACLE_TARGET_VERSION=19.1
    ```

    たとえば、Oracle Database 21cの場合は、次のように入力します。

```
SET ORACLE_TARGET_VERSION=21.1
```

構成ファイルのかわりにconfig_valuesパラメータを使用し、AutoUpgradeを実行するユーザー・アカウントにこれらのオペレーティング・システム環境変数を設定していない場合は、config_valuesを使用して少なくともこの4つの値を引数として指定する必要があります。

例: 既存の構成ファイルを使用したAutoUpgradeの実行

シナリオ: config_valuesを使用して既存の構成ファイルでAutoUpgradeを実行。次のコマンド構文では、myconfig.cfgファイルが作成されているローカル・ディレクトリからglobal.autoupg_log_dirを作成します。このコマンドの結果、global.autoupg_log_dirの場所は/dirに設定されます。

```
java –jar autoupgrade.jar –config /dir/myconfig.cfg –config_values
        “source_home=/srcdir, target_home=/trgdir, sid=sales” –mode
        deploy
```

構成ファイルmyconfigは、パス/dirに作成され、次のエントリが含まれます。

```
global.autoupg_log_dir=/dir
autoupgrade1.source_home=/srcdir
autoupgrade1.target_home=/trgdir
autoupgrade1.sid=sales
```

例: –config_valuesの値を指定しないAutoUpgradeの実行

分析、修正、アップグレードまたはデプロイの各モードでは、AutoUpgradeで実行する必要があるユーザー環境値を設定し、これらの値を–config_valuesの引数として渡さない場合、AutoUpgradeはデフォルトでサーバーに設定されているユーザー環境変数を使用します。

この仕組みを理解するために、ユーザーoracleとして、次の環境変数が設定されており、ターゲット・バージョンがOracle Database 21cのAutoUpgradeを実行するとします。

- ORACLE_HOMEは、/u01/app/oracle/product/12.2.0.1/dbhome_1に設定されています
- ORACLE_TARGET_HOMEは、/u01/app/oracle/product/19.0.0/dbhome_1に設定されています
- ORACLE_SIDは、salesに設定されています
- ORACLE_TARGET_VERSIONは、19.1に設定されています

ここで、2020年9月30日の午前11時45分15分に次のコマンドを実行したとします。

```
[Wed Sep 30 11:45:15] oracle@example:~$ java –jar autoupgrade.jar –config_values –
mode analyze
```

ログ・ディレクトリが指定されていないため、AutoUpgradeはデフォルトで、実行用の構成ファイルを一時ディレクトリに書き込みます。AutoUpgradeが作成した構成ファイルは、/tmp/autoupgrade/autoupgrade20200501114515.cfgファイルとしてパス/tmp/autoupgradeに存在し、次のエントリが含まれます。

```
global.autoupg_log_dir=/tmp/autoupgrade
# Value from environmental variable ORACLE_HOME
autoupgrade1.source_home=/u02/app/oracle/122
# Value from environmental variable ORACLE_TARGET_HOME
autoupgrade1.target_home=/scratch/oracle/19
# Value from environmental variable ORACLE_SID
autoupgrade1.sid=sales
# Value from environmental variable ORACLE_TARGET_VERSION
autoupgrade1.target_version=19.3
```

このオプションを使用すると、AutoUpgradeを使用して、アップグレードに関する広範な詳細を指定しなくても単一のデータ

ベース・アップグレードを処理できます。

例: 複数のデータベースに関する-config_valuesエントリを使用したAutoUpgradeの実行

このシナリオでは、*を使用して各データベースの値を区切り、複数のデータベースに関する-config_valuesエントリを指定して、ターゲット・リリースがOracle Database 21cのAutoUpgradeを実行します。

```
 java -jar autoupgrade.jar -config /tmp/auto.cfg -config_values
"global.autoupg_log_dir=/scratch/upglogs,source_home=/scratch/122,target_home=/scratc
h/21,sid=sales,*,source_home=/scratch/18,target_home=/scratch/21,sid=employees"
```

構成ファイルは、/tmp/auto.cfgとしてディレクトリ/tmpに作成され、次のエントリが含まれます。

```
global.autoupg_log_dir=/scratch/upglogs
autoupgrade1.source_home=/scratch/19
autoupgrade1.target_home=/scratch/21
autoupgrade1.sid=sales
autoupgrade2.source_home=/scratch/19
autoupgrade2.target_home=/scratch/21
autoupgrade2.sid=employees
```

親トピック: AutoUpgradeのコマンドライン・パラメータおよびオプション

console

AutoUpgradeパラメータconsoleは、AutoUpgradeコンソールをオンにし、AutoUpgradeジョブの進行状況を監視する一連のコマンドを提供します。

プロパティ	説明
パラメータ・タイプ	文字列
構文	autoupgrade.jar -config your-file -mode your-mode

説明

アップグレードを監視するには、AutoUpgradeパラメータconsoleを使用してコンソールを実行し、アップグレード・ジョブのステータスを監視します。

AutoUpgradeコンソールは、デフォルトでAutoUpgradeコマンドを使用して起動されます。オプション-console|-noconsoleを使用して、AutoUpgradeコンソールを再度有効または無効にできます

-noconsoleオプションを使用する場合、AutoUpgradeは構成ファイルの設定のみを使用して実行され、コンソールでの入力は必要ありません。複数のデータベースを分析する場合など、AutoUpgrade用のスクリプトを作成する場合は、noconsoleオプションを使用します。AutoUpgradeジョブが終了した後、分析モード・ログの出力を確認して、構成スクリプトに含まれる各データベースのアップグレードに必要な内容を確認できます。

ノート:

AutoUpgrade のインスタンスは必要な数を起動できますが、各インスタンスで一意のグローバル・ロギング・ディレクトリ(global.autoupg_log_dir)を使用する必要があります。グローバル・ロギング・ディレクトリが 1 つのみの

場合、起動できるインスタンスは 1 つのみです。

使用上のノート

コンソールを起動すると、コンソール内のオプションを使用できます。

コンソール・オプション	説明
-exit	クローズしてコンソールを終了します。実行中のジョブがある場合、それらは停止されます。
-help	コンソール・コマンド・ヘルプを表示します
-lsj [(-r|-f|-p|-e)-a number] -n number	数値 number で指定したジョブ数まで、ステータス別にジョブをリストします。次のフラグを使用できます。 -f: (オプション)終了したジョブでフィルタします。 -r: :(オプション)実行中ジョブでフィルタします。 -e: (オプション)エラーのあるジョブでフィルタします。 -p: (オプション)準備中のジョブでフィルタします。 -a number: (オプション)整数値(number)で指定された秒数の後にコマンドを繰り返します。 -n number: (必須)整数値で指定される、表示するジョブの数。
-lsr	リストア・キューを表示します。
-lsa	停止するジョブのキューを表示します。
-tasks	実行中のタスクを表示します。
-clear	端末表示をクリアします。
-resume -job number [-ignore_errors=ORA-number,ORA-number]	実行していた前のジョブから再起動します。ジョブを数値(number)で指定します。 ignore_errors オプション: このフラグはオプションです。パッチ適用またはアップグレード・ステージの処理中に再開エラーが発生した場合は、ステージの失敗としてレポートされます。AutoUpgrade は、アップグレード後の操作に進みませ

コンソール・オプション	説明
	ん。エラーを無視してアップグレード後の操作を続行できる場合もあります。特定のエラーがパッチ適用またはアップグレードに影響しないと思われる場合は、ignore_errors オプションを使用して、無視するエラーを指定し、アップグレード後の操作を続行できるようにすることができます。エラー番号は、カンマ区切りのエラー・リストで指定されます。例: -resume -job 444 -ignore_errors=ORA-48101,ORA-00001
-status [-job number | -c dbname| -config| -a number]	特定のジョブのステータスを、フラグで指定したレスポンスとともにリストします。 フラグ: -job number: 数値で指定された特定のジョブに関する情報を表示します。 -c dbname: 指定する特定のデータベース名(dbname)に関する詳細情報を表示します(可能な場合)。 -config number 指定したジョブの構成情報を表示します。 -a number: (オプション)整数値(number)で指定された秒数の後にコマンドを繰り返します。
-restore [-job (0-9)|-all_failed]	整数値 number で指定された AutoUpgrade ジョブでデータベースをアップグレードの開始前の状態にリストアします。 all_failed オプションを指定して実行すると、アップグレードが開始する前に、失敗したすべてのジョブが前の状態にリストアされます。
-logs	すべてのログ・ファイルの場所を表示します。
-abort -job number	入力した数値(number)で指定されたジョブを停止します。
-h[ist][/number]	コンソール・コマンドライン履歴を表示し、実行するフラットに応じてコマンドを再度実行するオプションを取得します。

コンソール・オプション	説明
	フラグ: / 最後のコマンドを再度実行します。 / number 指定したコマンドライン番号で指定された履歴ログ内のコマンドを実行します。

親トピック: AutoUpgradeのコマンドライン・パラメータおよびオプション

create_sample_file

AutoUpgradeパラメータcreate_sample_fileは、構成ファイルまたは設定ファイルを生成します。これらのファイルを編集して、本番構成ファイルまたはAutoUpgrade用設定ファイルを作成します。

プロパティ	説明
パラメータ・タイプ	文字列
構文	-create_sample_file config [filename] [full|unplug|noncdbtopdb] | settings [filename]
デフォルト値	create_sample_file config の場合、コマンドにファイル名を付加すると、サンプル構成ファイルが指定した名前で作成されます。出力ファイル名を指定しない場合、サンプル構成ファイルは sample_config.cfg という名前で作成されます。 次のオプションのいずれかを使用して、AutoUpgrade 構成ファイルのタイプを指定する句を追加できます。 ● full: 完全なオプションの AutoUpgrade 構成ファイル ● unplug: PDB の切断/接続アップグレードのオプションを含む AutoUpgrade 構成ファイル。 ● noncdbtopdb: 非 CDB から PDB へのアップグレード用のオプションを含む AutoUpgrade 構成ファイル。 settings 句を追加すると、内部設定構成ファイルが生成されます。デフォルトのファイル名をそのまま使用することも、カスタム名を指定することもできます。

使用上のノート

create_sample_fileパラメータはオプションです。他のパラメータと一緒に使用することはできません。このパラメータを指定する場合、settingsまたはconfig句のいずれかが必要です。

settings: sample_autoupg.cfgという名前または指定した名前でAutoUpgradeの**設定**ファイルが生成されます。

config: sample_config.cfgという名前または指定した名前でAutoUpgradeの**構成**ファイルが生成されます。

これらのサンプル・ファイルのいずれかを生成すると、ファイルを変更してAutoUpgradeユーティリティによるアップグレードの実行方法を制御できます。

- config: 構成モード・タイプのテンプレート・アップグレード構成ファイルを生成します。AutoUpgradeは、現在のフォルダにsample_config.cfgという名前または指定された名前のファイルを生成します
- settings: AutoUpgradeは、現在のフォルダにsample_autoupg.cfgという名前または指定された名前のファイルを生成します。

configとsettingsのいずれのオプションの場合も、拡張子が.cfgのデフォルトのファイル名が生成されます。ただし、ファイルが有効な(プレーン・テキスト)ファイルであれば、AutoUpgradeでは拡張子なしでも、拡張子を指定してもファイルを読み取ることができます。デフォルトの拡張子は、これらのファイルを構成ファイルとして識別する際に便利です。

サンプル構成ファイルの生成は、AutoUpgradeを使用するための標準的な準備で行われます。サンプル構成ファイルの構成ファイル・パラメータをカスタマイズすると、そのファイルを本番設定およびアップグレード用構成ファイルとして使用できます。

注意:

設定ファイルは、AutoUpgrade の内部設定を上書きするために使用されます。ほとんどのユース・ケースでは、サンプル設定ファイルの生成は必須ではありません。慎重に使用してください。

例

config句でcreate_sample_fileパラメータを実行する例:

```
[oracle@example ~]$ java -jar autoupgrade.jar -create_sample_file config
Created sample configuration file /home/oracle/sample_config.cfg
```

出力構成ファイル名を指定するconfig句でcreate_sample_fileパラメータを実行する例:

```
[oracle@example ~]$ java -jar autoupgrade.jar -create_sample_file config sales01
Created sample configuration file /home/oracle/sales01.cfg
```

settings句でcreate_sample_fileパラメータを実行する例:

```
oracle@example ~]$ java -jar autoupgrade.jar -create_sample_file settings
Created sample settings file /home/oracle/sample_autoupg.cfg
```

出力構成ファイル名を指定するsettings句でcreate_sample_fileパラメータを実行する例:

```
oracle@example ~]$ java -jar autoupgrade.jar -create_sample_file settings
testsetting.test
Created sample settings file /home/oracle/testsetting.test
```

親トピック: AutoUpgradeのコマンドライン・パラメータおよびオプション

error_code

AutoUpgradeパラメータerror_codeは、AutoUpgradeエラーのエラー・コードを示します。

プロパティ	説明
パラメータ・タイプ	文字列
構文	`-error_code [errorcode]`
デフォルト値	エラー・コードを指定しないと、すべての AutoUpgrade エラー・コードがコンソールに表示されます。 エラー・コードを指定すると、指定のエラー・コードに関する情報がコンソールに表示されます。

例

指定なしで入力すると、すべてのエラー・コードの説明が生成されます。

```
$ java -jar autoupgrade.jar  -error_code
ERROR1000.ERROR  = UPG-1000
ERROR1000.CAUSE  = It was not possible to create the data file where the jobsTable is
being written or there was a problem during the writing, it might be thrown due to a
permission error or a busy resource scenario
ERROR1001.ERROR  = UPG-1001
ERROR1001.CAUSE  = There was a problem reading the state file perhaps there was
corruption writing the file and in the next write it might be fixed
ERROR1002.ERROR  = UPG-1002
ERROR1002.CAUSE  = Error deserializing the object for rerun, review log for any
errors
.
.
.
```

特定のエラー・コードを入力すると、指定したエラーの出力が提示されます。たとえば:

```
java -jar autoupgrade.jar -error_code UPG-3010
```

このコマンドにより、次の出力が生成されます。

```
ERROR3010.ERROR  = UPG-3010
ERROR3010.CAUSE  = Error running approot_to_pdb.sql script
```

次に、もう一つの例を示します。

```
$ java -jar autoupgrade.jar  -error_code UPG-1400
```

このコマンドにより、次の出力が生成されます。

```
ERROR1400.ERROR  = UPG-1400
ERROR1400.CAUSE  = Database upgrade failed with errors
```

親トピック: AutoUpgradeのコマンドライン・パラメータおよびオプション

listchecks

AutoUpgradeパラメータ`listchecks`は、アップグレードのすべてのアップグレード・チェックのリストを提示するか、特定のチェックが指定された場合は、指定されたチェックに関する詳細を提示します。

プロパティ	説明
パラメータ・タイプ	文字列
構文	`-listchecks [checkname]`
デフォルト値	なし。特定のチェックが指定されていない場合は、すべての AutoUpgrade チェックのリストが提示されます。

例

指定しないで入力すると、Autoupgrade `listchecks`パラメータにより、AutoUpgradeがアップグレードに対して実行するすべてのチェックの説明が生成されます。

```
$ java -jar autoupgrade.jar  -listchecks
Check : AMD_EXISTS
        Description : Starting with Oracle Database 12c, the OLAP Catalog (OLAP AMD)
is desupported and will be automatically marked as OPTION OFF during the database
upgra
de if present. Oracle recommends removing OLAP Catalog (OLAP AMD) before database
upgrade.  This step can be manually performed before the upgrade to reduce downtime.
        Fixup Action : Remove OLAP Catalog by running the {1} SQL script
$ORACLE_HOME/olap/admin/catnoamd.sql script.
        Severity : WARNING
        Fixup Stage : PRE
        Min Version(inclusive) Check applies : NONE
        Max Version(exclusive) Check applies : NONE
        Check Introduced Version : NONE
        Check Removed Version : NONE
        Manual Fixup or Automatic : AUTO
        AutoUpgrade Only : NO
        Run for Datapatch : NO
Check : APEX_MANUAL_UPGRADE
        Description : Starting with Oracle Database Release 18, APEX is not upgraded
automatically as part of the database upgrade. Refer to My Oracle Support Note
1088970.
1 for information about APEX installation and upgrades. Refer to MOS Note 1344948.1
for the minimum APEX version supported for your target database release. Unsupported
ver
sions of APEX will be in an INVALID state when its database dependencies are not in
sync with the upgraded database.
        Fixup Action : Upgrade Oracle Application Express (APEX) manually before or
after the database upgrade.
        Severity : WARNING
        Fixup Stage : PRE
        Min Version(inclusive) Check applies : NONE
        Max Version(exclusive) Check applies : NONE
        Check Introduced Version : 18
        Check Removed Version : NONE
        Manual Fixup or Automatic : MANUAL
        AutoUpgrade Only : NO
        Run for Datapatch : NO
Check : APEX_PATCH
        Description : The APEX patching process is not performed by the {1} Oracle
database upgrade.  The APEX version upgrade only ensures that the APEX version is
upgrade
d to version {3} and does not guarantee the version is brought all the way to the
patched level {2}.  If a PDB from this CDB is unplugged and plugged into another ROOT,
the
.
.
.
```

チェックを指定して入力すると、`listchecks`により、指定したチェックのチェックに関する詳細が提示されます。

```
$ java -jar autoupgrade.jar  -listchecks XDB_RESOURCE_TYPE
Check : XDB_RESOURCE_TYPE
        Description : Direct access to either TYPE XDB.XDB$RESOURCE_T or TABLE
XDB.XDB$RESOURCE is restricted to Oracle internal code only.
        Fixup Action : Please contact Oracle Support to resolve the problem.
        Severity : ERROR
        Fixup Stage : PRE
        Min Version(inclusive) Check applies : 11.1
        Max Version(exclusive) Check applies : NONE
        Check Introduced Version : NONE
        Check Removed Version : NONE
        Manual Fixup or Automatic : MANUAL
        AutoUpgrade Only : NO
        Run for Datapatch : NO
```

親トピック: AutoUpgradeのコマンドライン・パラメータおよびオプション

load_password

AutoUpgradeパラメータload_passwordを使用すると、AutoUpgradeのキーストアにパスワードを安全に入力できます。分析モードでAutoUpgradeを実行すると、必要なパスワードが通知され、AutoUpgradeキーストアにロードできます。AutoUpgrade 22.1以降では、後のデータベース・アップグレードの一環として透過的データ暗号化(TDE)キーストアのパスワードを入力できます。

プロパティ	説明
パラメータ・タイプ	文字列
構文	-load_password
デフォルト値	なし。AutoUpgrade では、対話型プロンプトで値の入力を求められます。

説明

アップグレードに必要なパスワードを提供するために、load_passwordパラメータを使用できます。このパラメータは、-configパラメータとともに使用する必要があります。引数はありません。かわりに、キーストアに必要な情報を提供できる特定のコマンドを使用して対話型プロンプトを起動します。

AutoUpgrade 22.1以降、キーストア外部パスワード・ストアがなく、TDEを使用する場合、load_passwordを使用してアップグレードに必要なTDEパスワードをロードできます。

ノート:

- キーストア外部パスワード・ストアをデータベースで構成する場合、AutoUpgrade はキーストア外部パスワード・ストアの存在を検出し、TDE キーストア・パスワードを手動で入力するのではなく、この外部パスワード・ストアを使用します。ただし、すべてのデータベースがキーストア外部パスワード・ストアを使用して構成されている場合、引き続き global.keystore の定義が必要になりることもあります。
- 場合によっては、AutoUpgrade に、他の機密情報を書き込むための AutoUpgrade キーストアへのアクセス権が必要になります。たとえば、AutoUpgrade は、ADMINISTER KEY MANAGEMENT EXPORT KEYS および ADMINISTRATER KEY MANAGEMENT IMPORT KEYS コマンドによって使用される転送シークレット(パスフレーズ)を AutoUpgrade に書き込むことができます。

キーストア外部パスワード・ストアの詳細は、『*Oracle Database Advanced Security* ガイド』を参照してください。

`-mode analyze`を使用してAutoUpgradeを実行すると、AutoUpgradeは、構成ファイルでアップグレードに指定されたデータベースに必要なパスワードを検出し、アップグレード前のサマリー・レポートにリストします。アップグレードの前に、`-load_parameter`を使用してデータベースのパスワードを入力できます。これらのパスワードは、`global.keystore`で指定された場所にあるAutoUpgrade独自のキーストアに安全に格納されます。パスワードは、ソース・データベースのTDEキーストアにアクセスし、ターゲット・キーストアにTDEパスワードを書き込む場合にのみ使用されます。

注意:

AutoUpgrade を指定して `global.keystore` で作成するディレクトリにはソフトウェア・キーストアが含まれているため、TDE キーストア・ファイルで使用する場合と同じセキュリティのベスト・プラクティスを使用して保護する必要があります。 コマンドラインで `-load_parameter` オプションを指定して AutoUpgrade を実行すると、AutoUpgrade では対話型コンソールを起動してパスワード・オプションを構成できます。AutoUpgrade 22.1 以降では、次の構成オプションを使用できます。

- `add ORACLE_SID [-pdb isolated-pdb]`

  指定されたOracleシステム識別子(`ORACLE_SID`).のTDEパスワードを追加します

  パスワードを必要とする分離PDBがある場合、構成ファイルでオプションの`-PDB`パラメータを使用して、パスワードを指定する分離PDBを指定します。CDBルートおよびすべてのPDBが統一モードで構成されている場合、統一モードでは、キーストアがCDBルートと構成されているすべてのPDBの間で共有されるため、`-pdb`パラメータは必要ありません。

- `delete ORACLE_SID [-pdb isolated-pdb]`

  指定されたOracleシステム識別子(`ORACLE_SID`)のロード済パスワードを削除します。

  削除する分離PDBのパスワードをロードしていた場合、構成ファイルでオプションの`-pdb pdbname`パラメータを使用して、パスワードを削除する分離PDBの名前を指定します。CDBルートおよびすべてのPDBが統一モードで構成されている場合、`-pdb`パラメータは必要ありません。

- `list`

  各Oracle DatabaseをOracleシステム識別子(`ORACLE_SID`)別にリストし、各データベースの詳細を提供し、暗号化されたデータベース・アップグレードを実行するために追加のアクションが必要かどうかを示します。AutoUpgradeは、構成ファイルにリストされたどのデータベースにも保留中のアクションがない場合にのみデプロイ・モードを開始します。データベースをアップグレードする前にアクションが必要な場合、AutoUpgradeチェック`tde_passwords_required`が事前チェック・ステージ中に失敗します。

- `group`

  処理中のパスワード・グループを変更します。AutoUpgrade 22.1では、透過的データ暗号化(TDE)が唯一のグループ・オプションです。

- `save`

  キーストアを`global.keystore`ディレクトリで指定された場所に保存します。自動ログイン・キーストアがまだ有効化されていない場合、キーストアを自動ログイン・キーストアに変換するかどうかを入力するよう求められます。

- `help`

  使用可能なすべての`load_password`コマンドをリストします。

- exit

  load_password対話型コンソールを終了します。キーストアが変更され、まだ保存されていない場合は、終了する前にキーストアを保存するかどうかを確認するよう求められます。

アップグレード中に、AutoUpgradeがソース・データベース・キーストアにアクセスできるように、AutoUpgradeではメモリーの暗号化されたパスワード配列にパスワードを格納します。アップグレード中、SQL*Plusスクリプトにパスワードは書き込まれません。AutoUpgradeでパスワードが必要なくなった場合、これらのパスワードはメモリーから削除されます。パスワードのログ・レコードは保持されません。

例

AutoUpgradeキーストアに追加されたTDEキーストアのパスワード

AutoUpgradeを実行して、myconfig.cfgという名前の構成ファイルを使用して、TDEキーストアのパスワードをAutoUpgrade独自のキーストアに追加します(-load_passwordは、TDEキーストアのパスワードが必要なデータベースのTDEパスワードの入力を求めるために使用されます)。

```
java -jar autoupgrade.jar -config myconfig.cfg -load_password
```

TDEパスワードがロードされた後、分析または構成モードでAutoUpgradeを実行できます。

```
java -jar autoupgrade.jar -config myfile.cfg -mode deploy
```

AutoUpgradeは、独自のキーストアのTDEパスワード・ファイルを使用してソース・データベースのTDEキーストアにアクセスし、ターゲット・データベースのキーストアにTDEパスワードを書き込みます。

複数のソース・データベースのアップグレードで使用される複数のロード・パスワード・オプション

次の例では、すべてのload_passwordコマンド・オプションを使用して、ソース・データベースdb12201、cdb122およびdb19xから共通ターゲットCDB cdb19xにTDEパスワードをロードします。

```
$ java -jar autoupgrade.jar -config config.cfg -load_password
Processing config file ...
Starting AutoUpgrade Password Loader - Type help for available options
Creating new keystore - Password required
Enter password:
Enter password again:
Keystore was successfully created
TDE> add cdb19x
Enter your secret/Password:
Re-enter your secret/Password:
TDE> add cdb122
Enter your secret/Password:
Re-enter your secret/Password:
TDE> add db12201
Enter your secret/Password:
Re-enter your secret/Password:
TDE> add db19x
Enter your secret/Password:
Re-enter your secret/Password:
TDE> delete cdb19x
TDE> list
+----------+----------------+------------------+-----------+------------------+
|ORACLE_SID|Action Required | TDE Password     |SEPS Status|Active Wallet Type|
+----------+----------------+------------------+-----------+------------------+
|    cdb122|                |          Verified|   Inactive|               Any|
|    cdb19x|Add TDE password|No password loaded|   Inactive|               Any|
|   db12201|                |          Verified|   Inactive|        Auto-login|
|     db19x|                |          Verified|   Inactive|               Any|
+----------+----------------+------------------+-----------+------------------+
TDE> help
```

```
The following options are available
-----------------------------------
1 add
2 delete
3 list
4 group
5 save
6 help
7 exit
TDE> save
Convert the keystore to auto-login [YES|NO] ? YES
TDE> exit
AutoUpgrade Password Loader finished - Exiting AutoUpgrade
```

AutoUpgradeキーストアに追加された分離PDBのTDEキーストアのパスワードの追加

プラガブル・データベース(PDB)に独自のキーストアがある分離モードでは、PDBは独自のキーストアを別個に作成および管理ができます。分離モードのPDBの場合、AutoUpgradeパスワード・ローダーを起動し、構文`add Oracle_SID -pdb pdbname`を使用します(`Oracle_SID`はCDBルートの名前、`pdbname`は分離PDBの名前です)。

たとえば、CDBルートが`cdb19x`で、分離PDB名が`iso`の場合:

```
TDE> add cdb19x -pdb iso
Enter your secret/Password:
Re-enter your secret/Password:
```

関連項目

- 統一モードの構成について
- 外部キーストアの構成について
- 外部キー・マネージャへの自動オープン接続の構成

親トピック: AutoUpgradeのコマンドライン・パラメータおよびオプション

mode

AutoUpgradeパラメータ`mode`の値は、AutoUpgradeの実行モードを設定します。

プロパティ	説明
パラメータ・タイプ	文字列
構文	`-mode = [analyze|fixups|deploy|upgrade|postfixups]`
デフォルト値	なし。次のいずれかのオプションを選択します。 ● `analyze`: ソース Oracle ホームでアップグレード準備状況チェックを実行します。 ● `fixups`: アップグレード準備状況チェックおよびアップグレード前の修正を実行しますが、アップグレードは実行しません。 ● `deploy`: データベースのアップグレードを最初から最後

プロパティ	説明
	まで実行します。 ● upgrade: データベースのアップグレードおよびアップグレード後のアクションを実行します。このモードを開始する前に、ターゲット Oracle ホームのデータベースが稼働している必要があります。 ● postfixups: ターゲット Oracle ホームでデータベースの事後修正を実行します。

例

```
java -jar autoupgrade.jar -config config.cfg -mode analyze
java -jar autoupgrade.jar -config config.cfg -mode deploy
java -jar autoupgrade.jar -preupgrade "target_version=21" -mode fixups
```

親トピック: AutoUpgradeのコマンドライン・パラメータおよびオプション

noconsole

AutoUpgradeパラメータnoconsoleは、AutoUpgradeコンソールをオフにし、構成ファイル情報のみを使用してAutoUpgradeを実行します。

プロパティ	説明
パラメータ・タイプ	文字列
構文	-noconsole

説明

noconsoleオプションを使用する場合、AutoUpgradeは構成ファイルの設定のみを使用して実行され、コンソールでの入力は必要ありません。AutoUpgradeをバッチ・フローの一部として実行する場合や、複数のデータベースを分析する場合などにスクリプトで実行する場合は、noconsoleオプションを使用します。AutoUpgradeジョブが終了した後、分析モード・ログの出力を確認して、構成スクリプトに含まれる各データベースのアップグレードに必要な内容を確認できます。

ノート:

特定の構成ファイルに関連付けられている AutoUpgrade インスタンスは、一度に 1 つのみ実行できます。

使用上のノート

この例では、AutoUpgradeはnoconsoleモードで構成ファイルを使用して分析モードで実行されます。

```
java -jar autoupgrade.jar -config autoupgrade.cfg -mode analyze -noconsole
```

親トピック: AutoUpgradeのコマンドライン・パラメータおよびオプション

preupgrade

AutoUpgradeパラメータpreupgradeは、データベース・チェックおよびほとんどの問題をアップグレードの開始前に修正するアップグレード前修正と、ほとんどの問題をアップグレードの完了後に修正するアップグレード後修正を実行します。

プロパティ	説明
パラメータ・タイプ	文字列
構文	-preupgrade preupgrade_options -mode [analyze|fixups|postfixups]
デフォルト値	分析

説明

AutoUpgradeの-preupgrade.jar句は、以前のリリースで手動のアップグレード前情報ツール(preupgrade.jar)によって以前に実行された関数に置き換わります。-mode句は、次の3つの値のいずれかを取ります。

- analyze: アップグレードの準備状況をシステムで確認します。
- fixups: アップグレードの準備として、ソースOracle Databaseリリースで必要に応じて修正を実行します
- postfixups: アップグレードの完了後にターゲットのOracle Databaseリリースで修正を実行します。

-modeの値が指定されていない場合、-preupgradeパラメータはデフォルトでanalyzeモードになります。

使用上のノート

アップグレード前情報ツール(preupgrade.jar)で以前に使用可能にした機能と同じ機能を取得する場合にのみ、preupgrade句を使用します。ほとんどのアップグレード・シナリオでは、このパラメータを使用する必要はありません。

-preupgradeパラメータにはpreupgrade_optionsが必要です。これは、カンマ区切りのオプションと値のペアのリストを次の形式で指定します。option1=value1,option2=value2,…

引数

- target_version=release-number: アップグレード先のリリースであるターゲットのOracle Databaseリリース・バージョンを指定します。

  この引数の値は、analyzeおよびfixupsモードで必要です。ただし、ターゲット・リリースはtarget_homeから導出できます。したがって、analyzeおよびfixupsモードでは、target_versionまたはtarget_homeのいずれかを指定する必要があります。target_versionの値は、12.2以上のリリース値である必要があります。

- target_home=[target-path|env-variable]: アップグレード先のターゲット・リリースのOracle Databaseホームの場所を指定します(Oracleホーム・パスまたはオペレーティング・システム・パス変数のいずれか)。

  postfixupsモードを選択した場合、この引数は必須です。postfixupsモードを選択し、ターゲット・ホーム・パスを指定しない場合、デフォルト値は、AutoUpgradeを実行しているユーザーのOracleホーム・セットのOracleホーム環境変数(LinuxおよびUnixシステムの場合は$ORACLE_HOME、Microsoft Windowsシステムの場合は%ORACLE_HOME%)によって指定されます。

- oh=[source-path|env-variable]: アップグレード元のソース・リリースのOracle Databaseホームの場所を指定します(Oracleホーム・パスまたはオペレーティング・システム・パス変数のいずれか)。

  analyzeまたはfixupsモードを選択する場合、この引数は必須です。analyzeまたはfixupsモードを選択し、

ソース・ホーム・パスを指定しない場合、デフォルト値は、AutoUpgradeを実行しているユーザーのOracleホーム・セットのOracleホーム環境変数(LinuxおよびUnixシステムでは$ORACLE_HOME、Microsoft Windowsシステムでは%ORACLE_HOME%)によって指定されます。

- sid=system-identifier: アップグレードするソース・データベースのOracleシステム識別子を指定します。この引数は、analyzeまたはfixupsモードでは必須です。analyzeまたはfixupsモードのいずれかを選択し、システム識別子を指定しない場合、デフォルト値は、AutoUpgradeを実行しているユーザーのOracleホーム・セットのOracleホーム環境変数(LinuxおよびUnixシステムでは$ORACLE_SID、Microsoft Windowsシステムでは%ORACLE_SID%)によって指定されます。
- dir=path: 出力を特定のディレクトリに送ります。dir引数で出力ディレクトリを指定しなかった場合は、システムの一時ディレクトリに配置されているautoupgradeというフォルダに出力されます。通常、このディレクトリは次のいずれかの場所にあります。
  - LinuxまたはUnix: /tmpまたは/var/tmp。
  - Microsoft Windows: C:¥WINNT¥TEMP
- inclusion_list=list: 処理に含めるコンテナ・データベース(CDB)内のプラガブル・データベース(PDB)のリストを指定します。処理するPDBのスペース区切りリストを次のいずれかの形式で指定します。pdb1、pdb2およびpdb3は処理するPDBです。
  - pdb1 pdb2 pdb3
  - (pdb1 pdb2 pdb3)

  PDBのリストを指定しない場合、CDB内のすべてのPDBが処理されます。
- exclusion_list=list: 処理対象から除外するコンテナ・データベース(CDB)内のプラガブル・データベース(PDB)のリストを指定します。処理するPDBのスペース区切りリストを次のいずれかの形式で指定します。pdb1、pdb2およびpdb3は処理するPDBです。
  - pdb1 pdb2 pdb3
  - (pdb1 pdb2 pdb3)

  PDBのリストを指定しない場合、CDB内のすべてのPDBが処理されます。
- user=username: AutoUpgradeユーティリティがOracle Databaseへの接続に使用するOracle Databaseユーザー名を指定します。ユーザーが指定されている場合、AutoUpgradeはコマンドラインでユーザー名パスワードの入力を求めます。ユーザー名が指定されていない場合、AutoUpgradeはOracle Database接続にオペレーティング・システム認証を使用します。

モード

- analyze(デフォルト値): 指定したターゲット・リリース引数に適用されるすべてのアップグレード前チェックを使用して、分析モードで自動アップグレードを実行します。モードを指定しない場合、AutoUpgradeはデフォルトでanalyzeになります。
- fixups: アップグレード前に修正する必要があるソース・データベースで、AutoUpgrade Analyzeのアップグレード前チェックによって報告されたすべての問題に対してアップグレード前修正(使用可能な場合)を実行します。すべてのチェックが実行されます。

  修正結果は、ファイルupgrade.xmlにレポートされます。このファイルはパスlog_dir/db_name/jobnumber/prefixups/prefixups.xmlに配置されます。ここで、log_dirはdir引数を使用して指定するログ・ディレクトリ、db_nameはソース・データベースの名前、jobnumberは自動アップグレード・ジョブ番号です。

- postfixups: アップグレードの完了後に修正する必要があるアップグレード済データベースで、AutoUpgrade Analyzeのアップグレード前チェックによって報告されたすべての問題に対してアップグレード後修正(使用可能な場合)を実行します。

  修正後の結果は、ファイルpostfixups.xmlに報告されます。このファイルはパス log_dir/db_name/jobnumber/postfixupsに配置されます。ここで、log_dirはdir引数を使用して指定したログ・ディレクトリ、db_nameはソース・データベースの名前、jobnumberはAutoUpgradeジョブ番号です。

例

analyzeモードを使用してpreupgrade句を指定し、ターゲット・リリースがOracle Database 19cであることを指定してAutoUpgradeを実行します。

```
java -jar autoupgrade.jar -preupgrade "target_version=19" -mode analyze
```

fixupsモードを使用してpreupgrade句を指定し、ターゲット・リリースがOracle Database 19cであることを指定してAutoUpgradeを実行します。

```
java -jar autoupgrade.jar -preupgrade "target_version=19" -mode fixups
```

postfixupsモードを使用してpreupgrade句でAutoUpgradeを実行し、ターゲットのOracleホームがパス C:¥app¥oracle¥product¥19.0.0¥dbhome_1に存在することを指定します。

```
java -jar autoupgrade.jar -preupgrade
"target_home=C:¥app¥oracle¥product¥19.0.0¥dbhome_1" -mode postfixups
```

モードを指定せずにpreupgrade句を指定してAutoUpgradeを実行し、ターゲット・リリースがOracle Database 19cであることを指定します。この場合、使用されるモードはデフォルト・モードのanalyzeです。

```
java -jar autoupgrade.jar -preupgrade "target_version=19"
```

親トピック: AutoUpgradeのコマンドライン・パラメータおよびオプション

settings

AutoUpgradeパラメータsettingsは、AutoUpgradeユーティリティのカスタム・ランタイム構成の提供に使用する構成ファイルを識別します。

プロパティ	説明
パラメータ・タイプ	文字列
構文	-settings my-custom-advanced-settings
デフォルト値	該当なし

説明

settingsパラメータには、名前の必須引数およびカスタム設定で変更した設定構成ファイルへのパスがあります。settingsパラメータは単独では使用できませんが、AutoUpgradeが処理モードを実行する方法を変更する構成入力ファイルとして使用できます。

使用上のノート

このパラメータは、詳細パラメータです。ほとんどのアップグレード・シナリオでは、内部AutoUpgradeパラメータ設定を変更する

必要はありません。

例

この例では、settingsでmy_custom_advanced_settings.cfgという設定入力ファイルを指定しています。

```
java -jar autoupgrade.jar -settings my_custom_advanced_settings.cfg -config
config.cfg -mode deploy
```

親トピック: AutoUpgradeのコマンドライン・パラメータおよびオプション

version

AutoUpgradeパラメータ`version`は、`autoupgrade.jar`ファイルの現在のビルドを端末の画面に出力します。

プロパティ	説明
パラメータ・タイプ	文字列
構文	-version
デフォルト値	該当なし。

説明

サーバー上のautoupgrade.jarユーティリティのバージョンを確認するには、このオプション・パラメータを使用します。

使用上のノート

コマンド例:

```
java -jar autoupgrade.jar -version
```

出力例:

```
[oracle@example ~]$ java -jar autoupgrade.jar -version
build.version 22.1.220304
build.date 2022/03/04 13:29:34 -0500
build.hash 29007da
build.hash_date 2022/03/04 12:48:36 -0500
build.supported_target_versions 12.2,18,19,21
build.type production
```

親トピック: AutoUpgradeのコマンドライン・パラメータおよびオプション

restore

AutoUpgradeパラメータ`restore`は、指定したAutoUpgradeジョブのシステム・レベルのリストアを実行します。

プロパティ	説明
パラメータ・タイプ	文字列
構文	`-restore -jobs job#,job#`
デフォルト値	該当なし。

説明

このオプション・パラメータは、カンマ区切りのジョブ番号のリストを使用して、指定したジョブのシステムレベルのリストアを指定する場合に使用します。指定したアップグレード・ジョブのデータベースは、保証付きリストア・ポイント(GRP)にフラッシュバックされます。このコマンドを実行する前に、GRPがAutoUpgradeによって作成されている必要があります。

例

```
java -jar autoupgrade.jar -config config.cfg -restore -jobs 111
java -jar autoupgrade.jar -config config.cfg -restore -jobs 111,222 -console
java -jar autoupgrade.jar -config config.cfg -restore -jobs 111,222 -noconsole
```

親トピック: AutoUpgradeのコマンドライン・パラメータおよびオプション

restore_on_fail

AutoUpgradeパラメータ`restore_on_fail`は、デプロイメント中に失敗したジョブを自動的にリストアします。

プロパティ	説明
パラメータ・タイプ	文字列
構文	`-restore_on_fail`
デフォルト値	該当なし。

説明

このオプション・パラメータは、アップグレード・デプロイメント中に失敗したジョブをAutoUpgradeでリストアすることを指定する場合に使用します。

例

```
java -jar autoupgrade.jar -config config.cfg -mode deploy -restore_on_fail
```

親トピック: AutoUpgradeのコマンドライン・パラメータおよびオプション

zip

AutoUpgradeパラメータ`zip`は、AutoUpgradeサービス・リクエストのファイリングに必要なログ・ファイルのzipファイルを作成します。

プロパティ	説明
パラメータ・タイプ	文字列
構文	`-zip [-sid sid] [-d dir]`
デフォルト値	該当なし。

説明

このオプション・パラメータは、サービス・リクエストのオブジェクトであるジョブのログ・ファイルを含むOracleサポートに送信できるzipファイルを作成する場合に使用します。`-sid`句を使用して、ログ・ファイルを送信するデータベースのシステム識別子(SID)のカンマ区切りリストを指定できます。SID値が定義されていない場合は、AutoUpgradeによって、構成ファイルに指定されているすべてのデータベースのzipファイルが作成されます。`-d` 句を使用して、特定の出力ディレクトリを指定します。ディレクトリが指定されていない場合、コマンドの実行元の現在のディレクトリがzipファイルの出力に使用されます。

ノート:

`-zip` 句を使用する場合は、`-mode` 句は使用できません。

例

```
java -jar autoupgrade.jar -config yourconfig.cfg -zip
java -jar autoupgrade.jar -config yourconfig.cfg -zip -sid sales1,sales2 -d
/scratch/upgrd
```

親トピック: AutoUpgradeのコマンドライン・パラメータおよびオプション

AutoUpgradeユーティリティ構成ファイル

AutoUpgrade構成ファイルには、Oracle Databaseのアップグレードを実行するために必要なすべての情報が含まれています。

AutoUpgrade処理モードを使用するには、アップグレードするデータベース用にAutoUpgrade構成ファイルを作成する必要があります。

AutoUpgrade構成ファイルには、**グローバル**および**ローカル**構成パラメータが含まれています。デフォルトでは、グローバル・パラメータは構成ファイルによって扱われるすべてのデータベースに適用されます。特定のデータベースに対して指定した場合、ローカル構成パラメータは構成ファイルで指定されたグローバル・パラメータをオーバーライドします。

- AutoUpgrade構成ファイルのローカルに変更可能なグローバル・パラメータ
  AutoUpgradeの必須構成パラメータは、すべてのアップグレードに対してグローバルに設定することも、ローカルに設定することもできます。
- AutoUpgrade構成ファイルのローカル・パラメータ
  AutoUpgradeユーティリティのアップグレード用に特定のOracle Databaseの情報を構成するには、AutoUpgradeローカル・パラメータに情報を指定します。
- AutoUpgradeユーザー構成ファイルのグローバル・パラメータ
  構成ファイルで扱われるすべてのOracle Databaseアップグレードのパラメータに対するデフォルトの動作を指定するには、オプションのAutoUpgradeグローバル・パラメータを使用できます。

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

AutoUpgrade構成ファイルのローカルに変更可能なグローバル・パラメータ

AutoUpgradeの必須構成パラメータは、すべてのアップグレードに対してグローバルに設定することも、ローカルに設定することもできます。

使用上のノート

AutoUpgradeの必須パラメータをローカルに変更可能なグローバル・パラメータとしてグローバルに設定した場合、これらのパラメータは、特定のアップグレードに設定されたローカル・パラメータによってオーバーライドできるため、AutoUpgradeジョブの処理をより適切に制御できます。

ローカルに変更可能なグローバル・パラメータでは、接頭辞`global`を使用して、必須パラメータの値をAutoUpgrade構成ファイルのすべてのジョブのグローバル・パラメータとして設定できますが、ローカル・ジョブ接頭辞と同じパラメータを指定して、グローバル値を同じ構成ファイル内の特定のジョブの別の値に再設定できます。ローカルで変更可能なグローバル・パラメータのみを、各AutoUpgradeジョブのローカル・パラメータとして設定することもできます。

> ノート:
>
> これらのパラメータは、My Oracle Support からダウンロードできる AutoUpgrade の最新バージョンで使用できます。

ローカルで変更可能なグローバル・パラメータがグローバル接頭辞とローカル・ジョブ接頭辞の両方で設定されている場合、ローカルに変更されたパラメータ値は、パラメータで使用する接頭辞によって識別されるジョブのグローバル・パラメータ値をオーバーライドします。

たとえば、`global.target_home`の場合、使用する構文は`global.target_home=Global target Oracle home`および`database.target_home=local target Oracle home`の形式になります。

例

AutoUpgrade構成ファイルでは、必須パラメータ`target_home`が1つのOracleホーム・パスにグローバルに設定されます。ただし、構成ファイルでは、同じパラメータが別のOracleホーム・パスに対してローカルに設定されます。AutoUpgradeは構成ファイル内のジョブを処理すると、接頭辞`upgrade3`で定義されたジョブの`target_home`にローカルに定義されたパスを使用し、グローバル・パラメータ設定をオーバーライドします。

```
global.target_home=/u01/app/oracle/21.0.0/dbhome01
upgrade3.target_home=/u03/app/oracle3/12.2.0.1/dbhome3
```

- defer_standby_log_shipping
  プライマリ・データベースから任意のスタンバイ・データベースへのログの送信を遅延させます。すべてのログ・アーカイブの宛先(`log_archive_dest_n`)が遅延に設定されます。
- dictionary_stats_after
  (オプション)アップグレードの完了後に、AutoUpgradeがターゲット・データベースのデータ・ディクショナリ統計を収集するように指定します。
- dictionary_stats_before
  (オプション)アップグレードを開始する前に、AutoUpgradeがソース・データベースのデータ・ディクショナリ統計を収集するように指定します。

- drop_grp_after_upgrade
  データベースのアップグレード後に、保証付きリストア・ポイント(GRP)を削除します。
- enable_local_undo
  CDBのアップグレードの場合、CDB$ROOTのアップグレード前にLOCAL UNDOを有効にする必要があるかどうかを指定します。
- fixed_stats_before
  (オプション)アップグレードを開始する前に、AutoUpgradeがソース・データベースの固定オブジェクト統計を収集するように指定します。
- manage_network_files
  アップグレード中にネットワーク・ファイルを処理するかどうかを指定します。
- remove_underscore_parameters
  構成ファイル内のすべてのOracle Databaseについて、アップグレード中およびアップグレード後にPFILEファイルからアンダースコア(非表示)パラメータを削除します。
- restoration
  (Enterprise Editionでのみ使用可能)データベース・リストアの保証付きリストア・ポイント(GRP)を生成します。
- target_base
  ターゲットOracleホームのターゲットORACLE_BASEパスを指定します。
- target_home
  (ターゲット・ホームがシステム上にない場合で、アップグレード・モードおよびデプロイ・モードの場合は必須です。分析および修正モードの場合はオプションです。)ターゲットOracleホーム(ORACLE_HOME)パスを指定します。
- target_version
  (ターゲットOracleホームがシステム上にないか、リリース12.2の場合は必須) AutoUpgradeでアップグレードを実行するターゲット・リリース・バージョンを指定します。

親トピック: AutoUpgradeユーティリティ構成ファイル

defer_standby_log_shipping

プライマリ・データベースから任意のスタンバイ・データベースへのログの送信を遅延させます。すべてのログ・アーカイブの宛先(log_archive_dest_n)が遅延に設定されます。

使用上のノート

デフォルトでは、ログ送信はアップグレードの一部として実行されます。Autoupgradeによりログ送信が遅延すると、ログ送信が遅延され、アップグレードが正常に完了した後にプライマリ・データベースからセカンダリ・データベースへの送信ログを再度有効にする必要があることが通知されます。

> ノート:
>
> この構成ファイル・パラメータは、スタンバイ・データベースのみならず、Oracle Zero Data Loss Recovery Appliance (ZDLRA)リアルタイム・ログ転送や Oracle GoldenGate ダウンストリーム取得など、プライマリ・データベースから REDO を受信するすべての製品またはサービスに影響します。

オプション

[yes | no]

デフォルト値はnoです

デフォルトはnoです(ログ出送信は遅延されません)。デフォルトをYesに変更すると、ログ送信は遅延されるため、アップグレード後に手動で再度有効にする必要があります。

例

```
defer_standby_log_shipping=yes
```

親トピック: AutoUpgrade構成ファイルのローカルに変更可能なグローバル・パラメータ

dictionary_stats_after

(オプション)アップグレードの完了後に、AutoUpgradeがターゲット・データベースのデータ・ディクショナリ統計を収集するように指定します。

使用上のノート

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薦めします。yesを指定すると、AutoUpgradeはアップグレードの完了後にディクショナリ統計を収集します。

オプション

[yes | no]

デフォルト値はYesです。

例

```
global.dictionary_stats_after=yes

sales.dictionary_stats_after=yes
```

親トピック: AutoUpgrade構成ファイルのローカルに変更可能なグローバル・パラメータ

dictionary_stats_before

(オプション)アップグレードを開始する前に、AutoUpgradeがソース・データベースのデータ・ディクショナリ統計を収集するように指定します。

使用上のノート

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薦めします。yesを指定すると、AutoUpgradeはアップグレードを開始する前にディクショナリ統計を収集します。

オプション

[yes | no]

デフォルト値はYesです。

例

```
global.dictionary_stats_before=yes

sales.dictionary_stats_before=yes
```

親トピック: AutoUpgrade構成ファイルのローカルに変更可能なグローバル・パラメータ

drop_grp_after_upgrade

データベースのアップグレード後に、保証付きリストア・ポイント(GRP)を削除します。

使用上のノート

このオプションを選択した場合、アップグレードが正常に完了した後にGRPが削除されます。

オプション

`[yes | no]`

デフォルト値はnoです。

例

```
global.drop_grp_after_upgrade=yes

sales.drop_grp_after_upgrade=yes
```

親トピック: AutoUpgrade構成ファイルのローカルに変更可能なグローバル・パラメータ

enable_local_undo

CDBのアップグレードの場合、CDB$ROOTのアップグレード前にLOCAL UNDOを有効にする必要があるかどうかを指定します。

使用上のノート

このオプションを選択した場合、AutoUpgradeはアップグレード前に次の文を実行します: `ALTER DATABASE LOCAL UNDO ON;`。

ローカルUNDOを最初に有効にすると、PDB$SEED内のUNDO表領域のサイズは、CDB$ROOT内のUNDO表領域のサイズの係数として決定されます。デフォルトは、UNDO表領域サイズの30%です。CDB内の他のすべてのPDBは、PDB$SEEDからこのプロパティを継承します。新しいUNDO表領域を割り当てるのに十分な領域があることを確認します。

オプション

`[yes | no]`

デフォルト値はnoです。

例

```
enable_local_undo=yes
```

親トピック: AutoUpgrade構成ファイルのローカルに変更可能なグローバル・パラメータ

fixed_stats_before

(オプション)アップグレードを開始する前に、AutoUpgradeがソース・データベースの固定オブジェクトの統計を収集するように指定します。

使用上のノート

アップグレードの前に、固定オブジェクトの統計を再収集することをお薦めします。

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。

オプション

[yes | no]

デフォルト値はYesです。

例

```
global.fixed_stats_before=yes

sales.fixed_stats_before=yes
```

親トピック: AutoUpgrade構成ファイルのローカルに変更可能なグローバル・パラメータ

manage_network_files

アップグレード中にネットワーク・ファイルを処理するかどうかを指定します。

使用上のノート

このオプションを選択すると、AutoUpgradeは、指定したオプションに応じてネットワーク・ファイルを処理します。

oranfstab、ldap.ora、tnsnames.ora、sqlnet.oraおよびlistener.oraの各ネットワーク・ファイルが処理されます。

オプション

[FULL|SKIP|IGNORE_READ_ONLY]

- FULL: (デフォルト)ネットワーク・ファイルのターゲットOracleホームへのコピーおよびマージ中に発生したすべての例外を呼び出します。
- SKIP: アップグレード後にネットワーク・ファイルを処理しません。
- IGNORE_READ_ONLY: ネットワーク・ファイルをコピーおよびマージしようとしましたが、ターゲット・ファイルが読取り専用の場合、アップグレード中に例外は発生しません。

例

```
manage_network_files=ignore_read_only
```

親トピック: AutoUpgrade構成ファイルのローカルに変更可能なグローバル・パラメータ

remove_underscore_parameters

構成ファイル内のすべてのOracle Databaseについて、アップグレード時およびアップグレード後にPFILEファイルからアンダースコア(非表示)パラメータを削除します。

使用上のノート

アンダースコアのパラメータは、Oracleサポートの指示がある場合のみ使用できます。

オプション

[yes | no]

デフォルト値はnoです。

例

```
global.remove_underscore_parameters=yes
```

親トピック: AutoUpgrade構成ファイルのローカルに変更可能なグローバル・パラメータ

restoration

(Enterprise Editionでのみ使用可能)データベース・リストアの保証付きリストア・ポイント(GRP)を生成します。

使用上のノート

このオプションは、データベース・バックアップおよびデータベースのリストアをDBAによって手動で実行する必要があるかどうかを決定します。

Standard Editionはフラッシュバック・データベースをサポートしていないため、このオプションはStandard Editionでは使用できません。データベースがStandard EditionのOracle Databaseの場合、別個のフォールバック・メカニズムを持つようにする必要があります。

オプション

`[yes | no]`

デフォルト値はyes。

例

```
global.restoration=no
```

親トピック: AutoUpgrade構成ファイルのローカルに変更可能なグローバル・パラメータ

target_base

ターゲットOracleホームのターゲットORACLE_BASEパスを指定します。

例

```
global.target_base=/u01/app/oracle
sales4.target_base=/u04/app/oracle4
```

親トピック: AutoUpgrade構成ファイルのローカルに変更可能なグローバル・パラメータ

target_home

(ターゲット・ホームがシステム上にない場合で、アップグレード・モードおよびデプロイ・モードの場合は必須です。分析および修正モードの場合はオプションです。)ターゲットOracleホーム(ORACLE_HOME)パスを指定します。

使用上のノート

AutoUpgradeでは、このパラメータで指定したリリース・バージョン情報を使用して、アップグレード対象のターゲットのOracle Databaseリリースに対して正しいチェックおよび修正が使用されるようにします。このパラメータの形式は、有効なOracleバージョンのピリオド区切りの値です。

このオプションは、ターゲット・ホームがシステムに存在しないか、ターゲット・ホームが12.2リリースの場合のみ必要です。それ以外の場合、AutoUpgradeはターゲット・リリース値を導出できます。

オプション

有効な値

- 12.2
- 18
- 19
- 21

例

```
global.target_version=19
employees.target_version=12.2
```

親トピック: AutoUpgrade構成ファイルのローカルに変更可能なグローバル・パラメータ

target_version

(ターゲットOracleホームがシステム上にないか、リリース12.2の場合は必須) AutoUpgradeでアップグレードを実行するターゲット・リリース・バージョンを指定します。

使用上のノート

AutoUpgradeでは、このパラメータで指定したリリース・バージョン情報を使用して、アップグレード対象のターゲットのOracle Databaseリリースに対して正しいチェックおよび修正が使用されるようにします。このパラメータの形式は、有効なOracleバージョンのピリオド区切りの値です。

このオプションは、ターゲット・ホームがシステムに存在しないか、ターゲット・ホームが12.2リリースの場合のみ必要です。それ以外の場合、AutoUpgradeはターゲット・リリース値を導出できます。

オプション

有効な値

- 12.2
- 18
- 19
- 21

例

```
global.target_version=19
employees.target_version=12.2
```

親トピック: AutoUpgrade構成ファイルのローカルに変更可能なグローバル・パラメータ

AutoUpgrade構成ファイルのローカル・パラメータ

AutoUpgradeユーティリティのアップグレード用に特定のOracle Databaseの情報を構成するには、AutoUpgradeローカル・パラメータに情報を指定します。

使用上のノート

ローカル・パラメータは、AutoUpgrade構成ファイルで設定されたグローバル・パラメータよりも優先されます。ローカルで設定する必要があるか、ローカルで変更可能なグローバル・パラメータとして設定する必要があるローカル・パラメータは(「必須」)で示されます。すべてのローカル・パラメータは接頭辞を使用します(例: 特定のデータベースまたはアップグレードを識別するために定義する値で識別)。接頭辞は、構成ファイルでパラメータが適用される特定のアップグレード・ジョブを識別します。

例: 構成ファイルの最初のアップグレードのパラメータのセットでは、接頭辞salesが使用され、構成ファイル内の次のアップグレードのパラメータのセットでは、接頭辞employeesが使用されます。

```
sales.source_home=/u01/app/oracle/12.2/dbhome1
.
.
.
employees.sid=salescdb
employees.source_home-/03/app/oracle/21/dbhome1
```

- add_after_upgrade_pfile
  (オプション)アップグレード後にパラメータを追加するPFILEのパスおよびファイル名を指定します。
- add_during_upgrade_pfile
  (オプション)アップグレード中にパラメータを追加するPFILEのパスおよびファイル名を指定します。
- after_action
  (オプション)deployモードでは、接頭辞アドレスで識別されるデータベースのデプロイ・ジョブの完了後に実行するカスタム・アクションを指定します。
- before_action
  (オプション) deployモードでは、接頭辞で指定された特定のデータベース・ジョブのアップグレード・ジョブの起動前に実行するカスタム・アクションを指定します。すべてのアップグレード・ジョブの前にスクリプトを実行する場合は、ローカル・パラメータ (global.before_action)を使用してそのスクリプトを指定します
- catctl_options
  (オプション) AutoUpgradeがcatctl.plに送信してデフォルトの動作をオーバーライドするために選択できるcatctl.plオプションのセットを1つ以上指定します。
- checklist
  (オプション)ポリシーまたはセキュリティ上の問題のために自動的に実装しない修正など、AutoUpgradeによって実行される修正のデフォルト・リストをオーバーライドするために使用できるチェックリストへのパスを指定します。
- close_source
  (オプション) AutoUpgradeが切断再配置アップグレードを開始する直前に、ソース非CDBまたはソースPDBをクローズします。
- del_after_upgrade_pfile
  (オプション)アップグレード後にパラメータを削除するPFILEのパスおよびファイル名を指定します。
- del_during_upgrade_pfile
  (オプション)アップグレード中にパラメータを削除するPFILEのパスおよびファイル名を指定します。
- drop_win_src_service

(オプション) Microsoft Windowsでのアップグレードの場合、アップグレード後にソースOracle DatabaseのWindowsオペレーティング・システム・サービスを削除するかどうかを指定します。

- env
  (オプション) ORACLE_SID、ORACLE_HOME、ORACLE_BASEおよびTNS_ADMINを除く、オペレーティング・システムに設定されているカスタム・オペレーティング・システム環境変数を指定します。
- ignore_errors
  (オプション)アップグレードまたはパッチ適用プロセス中にAutoUpgradeで無視する特定のOracleエラーのカンマ区切りリストを指定できます。
- keep_source_pdb
  (オプション)切断/接続アップグレード操作のソースPDBが、ソースCDBから削除されるのではなく、閉じた状態に保持されるかどうかを指定します。
- log_dir
  (オプション)パラメータの接頭辞で識別されるアップグレード・ジョブに含まれるデータベース・セットにあるデータベース・アップグレード用に生成されるログ・ファイルの場所を設定します。
- manage_standbys_clause
  (オプション) DB_UNIQUE_NAMEで特定したスタンバイOracle Data Guardスタンバイ・データベースをAutoUpgradeプラグイン・アップグレードから除外して、スタンバイ・データベース・ファイルを再利用できるかどうかを指定します。
- pdbs
  (オプション)アップグレードを実行するPDBのリストを設定します。このパラメータは、マルチテナント・アーキテクチャ(CDB)データベースのアップグレードにのみ適用されます。非CDBデータベースを接続およびアップグレードしている場合、このパラメータは無視されます。
- raise_compatible
  (オプション)アップグレードが正常に完了した後、Oracle DatabaseのCOMPATIBLE初期化パラメータをターゲット・リリースのデフォルト値まで増やします。
- remove_rac_config
  (オプション)ターゲットCDBホームへの正常な変換後にソースOracleホームのクラスタウェアから非CDB Oracle RACデータベースを削除するか、ソース・データベースを変更せずに残すかを指定します。
- remove_underscore_parameters
  (オプション)構成ファイル内のすべてのOracle Databaseについて、アップグレード中およびアップグレード後にPFILEファイルからアンダースコア(非表示)パラメータを削除します。
- replay
  (オプション)リプレイを使用してデータベースをアップグレードするかどうかを指定します。
- restoration
  (オプション)データベースのリストアに保証付きリストア・ポイント(GRP)を生成します。
- revert_after_action
  (オプション)接頭辞によって指定された特定のデータベース・ジョブに対してシステムのリストアが完了し、データベースが稼働した後にオペレーティング・システムで実行するカスタム・アクションを指定します。
- revert_before_action
  (オプション)接頭辞によって指定された特定のデータベース・ジョブに対してシステムのリストアが完了し、データベースが稼働する前にオペレーティング・システムで実行するカスタム・アクションを指定します。
- run_hcheck
  (オプション)アップグレード前のチェックの一部としてOracle Dictionary Health Checksを実行して、データベース・

ディクショナリの不整合を特定するかどうかを指定します。

- run_utlrp
  (オプション)アップグレード後の処理の一部として、utlrp.sqlの1つのバージョンの実行を有効または無効にして、Oracle管理スキーマの無効なオブジェクトのみを再コンパイルします。
- sid
  (必須)アップグレードするデータベースのOracleシステム識別子(SID)を示します。
- skip_tde_key_import
  (オプション) yesに設定すると、アップグレードは実行されますが、ソース・データベースKeyStoreのターゲット・データベースへのインポートはスキップされ、エラーは発生しません。
- source_base
  (オプション)ソースOracleホームのソースORACLE_BASEパスを指定します。
- source_dblink
  (オプション)切断/接続再配置(ホット・クローン)アップグレード用に設定されたデータベース・リンクを指定します。
- source_home
  (分析モード、修正モードおよびデプロイ・モードの場合は必須です。アップグレード・モードの場合はオプションです。)アップグレードするデータベースの現在のOracleホーム(ORACLE_HOME)。
- source_ldap_admin_dir
  (オプション)ソース・データベース・ホームのLDAP_ADMINディレクトリへのパスを指定します。
- source_tns_admin_dir
  (オプション)ソース・データベース・ホームのTNS_ADMINディレクトリへのパスを指定します。
- start_time
  (オプション)アップグレード・ジョブを実行する将来の起動時間を設定します。このパラメータは、アップグレード・ジョブをスケジュールしてサーバーの負荷を分散し、複数のジョブが即時に起動されないようにするために使用します。
- target_base
  (オプション)ターゲットOracleホームのターゲットORACLE_BASEパスを指定します。
- target_cdb
  (オプション)非CDB Oracleデータベースが接続されるターゲットCDBのSIDを指定します。このパラメータは、非CDBのOracleデータベースをアップグレードおよび変換する場合に必須です。
- target_pdb_copy_option=file_name_convert
  (オプション)非CDBデータベースをPDBに変換する場合、または別のソースCDBの既存のPDBを指定のターゲットCDBのPDBに変換する場合にAutoUpgradeが実行する、プラガブル・データベースを作成する文で使用されるfile_name_convertオプションを指定します。
- target_pdb_name
  (オプション)ターゲットCDBに接続された後に、非CDBのソースOracleデータベースに割り当てる名前を指定します。
- target_ldap_admin_dir
  (オプション)ターゲット・データベース・ホームのLDAP_ADMINディレクトリへのパスを指定します。
- target_tns_admin_dir
  (オプション)ターゲット・データベース・ホームのTNS_ADMINディレクトリへのパスを指定します。
- timezone_upg
  (オプション) AutoUpgradeプロセスの一環としてタイムゾーンのアップグレードの実行を有効または無効にします。
- tune_setting
  (オプション)指定したワークフロー・オプションに応じて、実行時にAutoUpgradeの動作を変更する特別なワークフローを有効にします。

- upgrade_node
  (オプション)現在のユーザー構成が有効なノードを指定します。デフォルト値は`localhost`です。

親トピック: AutoUpgradeユーティリティ構成ファイル

add_after_upgrade_pfile

(オプション)アップグレード後にパラメータを追加する`PFILE`のパスおよびファイル名を指定します。

例

```
sales3.add_after_upgrade_pfile=/path/to/my/pfile_add.ora
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

add_during_upgrade_pfile

(オプション)アップグレード時にパラメータを追加するPFILEのパスおよびファイル名を指定します。

例

```
sales3.add_during_upgrade_pfile=/path/to/my/newpfile.ora
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

after_action

(オプション) `deploy`モードでは、接頭辞アドレスで識別されるデータベースのデプロイ・ジョブの完了後に実行するカスタム・アクションを指定します。

使用上のノート

使用するスクリプトは`name.ext` (たとえば、`myscript.sh`)の形式である必要があります。これにより、AutoUpgradeは実行するスクリプトのタイプを識別できます。許可される拡張子オプションは、次のとおりです。

- UNIXシェル(`.sh`)
- Microsoft Windowsバッチ(`.bat`、`.cmd`)
- Microsoft Windows PowerShell (`.ps1`)
- Oracle SQLファイル(`.sql`)。接頭辞によって指定されたローカル操作のみ。

デフォルトでは、スクリプトが失敗すると、AutoUpgradeが引き続き実行されます。オペレーティング・システムがスクリプトの失敗を検出した場合、`Y`フラグを使用してAutoUpgradeが停止するように指定します。スクリプトが`0`とは異なるステータスで終了すると、失敗して完了したとみなされます。

グローバルの`after_action`パラメータとは対照的に、ローカルの`after_action`パラメータはSQLスクリプトを指定でき、非CDBのOracleホーム上のターゲットOracleデータベース・バイナリまたは`CDB$ROOT`を使用してデータベースで実行されます。追加のコンテナ固有のアクションを実行する場合、コード内でこれらを設定する必要があります。より複雑なシナリオでは、シェルでコンテナ固有のアクションを実行できます。

スクリプトの出力が取得され、ファイルに格納されます。`stdout`と`stderr`の両方が取得されます。ファイルは、特定のデータベースまたはジョブに一致するディレクトリの`postupgrade`サブディレクトリに格納されます。

次の環境変数はスクリプトを実行するシェルに設定されます。

- ORACLE_SID
- ORACLE_UNQNAME
- ORACLE_BASE
- ORACLE_HOME
- TNS_ADMIN

例

AutoUpgradeが処理を開始した後に、指定したスクリプトを実行します。スクリプトが失敗した場合にAutoUpgradeを停止するには、次のようにYフラグを設定します。

```
sales2.after_action=/user/path/script.sh Y
```

AutoUpgradeがデプロイ・オプションの処理を開始した後、指定したスクリプトを実行します。スクリプトが失敗した場合に実行を継続するには、次のようにAutoUpgradeを設定します。

```
sales3.after_action=/user/path/script.sh
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

before_action

(オプション) deployモードでは、接頭辞で指定された特定のデータベース・ジョブのアップグレード・ジョブの起動前に実行するカスタム・アクションを指定します。すべてのアップグレード・ジョブの前にスクリプトを実行する場合は、ローカル・パラメータ(global.before_action)を使用してそのスクリプトを指定します

使用上のノート

使用するスクリプトはname.ext (たとえば、myscript.sh)の形式である必要があります。これにより、AutoUpgradeは実行するスクリプトのタイプを識別できます。許可される拡張子オプションは、次のとおりです。

- UNIXシェル(.sh)
- Microsoft Windowsバッチ(.bat、.cmd)
- Microsoft Windows PowerShell (.ps1)
- Oracle SQLファイル(.sql)。接頭辞によって指定されたローカル操作の場合のみ。

デフォルトでは、スクリプトが失敗すると、AutoUpgradeが引き続き実行されます。オペレーティング・システムがスクリプトの失敗を検出した場合、Yフラグを使用してAutoUpgradeが停止するように指定します。スクリプトが0とは異なるステータスで終了すると、失敗して完了したとみなされます。

グローバルbefore_actionパラメータとは対照的に、ローカルbefore_actionパラメータはSQLスクリプトを指定でき、これは前のリリースのOracle Databaseバイナリを使用して、ソース・データベースのOracleホームのデータベースで実行されます。スクリプトは非CDBのOracleホームまたはCDB$ROOTで実行されます。追加のコンテナ固有のアクションを実行する場合、コード内でこれらを設定する必要があります。より複雑なシナリオでは、シェルでコンテナ固有のアクションを実行できます。

スクリプトの出力が取得され、ファイルに格納されます。stdoutとstderrの両方が取得されます。ファイルは、特定のデータベースまたはジョブに一致するディレクトリのpreupgradeサブディレクトリに格納されます。

次の環境変数はスクリプトを実行するシェルに設定されます。

- ORACLE_SID
- ORACLE_UNQNAME

- ORACLE_BASE
- ORACLE_HOME
- TNS_ADMIN

例

AutoUpgradeがdeployモードの処理を開始する前に、指定したスクリプトを実行します。スクリプトが失敗した場合にAutoUpgradeを停止するには、次のようにYフラグを設定します。

```
sales.before_action=/user/path/script.sh Y
```

AutoUpgradeが処理を開始する前に、指定したスクリプトを実行します。スクリプトが失敗した場合実行を継続するには、次のようにAutoUpgradeを設定します。

```
sales4.before_action=/user/path/script.sh
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

catctl_options

(オプション) AutoUpgradeがcatctl.plに送信してデフォルトの動作をオーバーライドするために選択できるcatctl.plオプションのセットを1つ以上指定します。

使用上のノート

使用可能なcatctl.plオプション:

- -n パラレル操作に使用するプロセス数。リプレイ・アップグレードの場合、アップグレードに使用されるパラレル・プロセスの数は、CPU_COUNTを4で割った値にデフォルト設定されます。クラシック・アップグレードの場合、CDB$ROOTのデフォルトは8です。
- -N PDBのアップグレード時に使用するプロセッサの数。リプレイ・アップグレードの場合、アップグレードに使用されるパラレル・プロセスの数は、CPU_COUNTを4で割った値にデフォルト設定されます。クラシック・アップグレードの場合、デフォルトは2です
- -T オフラインのユーザー・スキーマベースの表領域を取得します。
- -z catcon.plの本番デバッグ情報を有効にします。

例

```
sales4.catctl_options=-n 24 -N 4
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

checklist

(オプション)ポリシーまたはセキュリティ上の問題のために自動的に実装しない修正など、AutoUpgradeによって実行される修正のデフォルト・リストをオーバーライドするために使用できるチェックリストへのパスを指定します。

使用上のノート

他のAutoUpgradeモードでこのパラメータを使用するには、analyzeモードでAutoUpgradeを実行する必要があります。AutoUpgradeで分析が終了すると、事前チェック・ディレクトリ(dbname_checklist.cfg)の下のデータベース名によって識別されるチェックリスト・ファイルを検索できます。ファイルを手動で更新して、AutoUpgradeでバイパスする修正を除外し、ファイルを新しい名前で保存します。AutoUpgradeを再度実行すると、作成したチェックリスト・ファイルを指すパラメータを指定し、個々のデータベースに対して実行された修正を変更できます。チェックリスト・ファイル・パスを指定しない場合、アップグレード

時に実行される修正のセットは、指定したデプロイ・モードで作成される最新バージョンのチェックリスト・ファイルです。

例

```
sales.checklist=/u01/app/oracle/upgrade-jobs/salesdb_checklist.cfg
```

前述の例では、`salesdb_checklist.cfg`はデータベース`salesdb`のチェックリスト構成ファイルです。

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

close_source

(オプション) AutoUpgradeが切断再配置アップグレードを開始する直前に、ソース非CDBまたはソースPDBをクローズします。

使用上のノート

切断再配置操作中に、`close_source`が`yes` (デフォルト)に設定されている場合、AutoUpgradeは、アップグレードを開始する直前にソース非CDBまたはソースPDBをクローズします。さらに、Oracle Real Application ClustersまたはOracle Grid Infrastructure (CRS)サービスが非CDBソース用に構成されている場合、アップグレードを開始する前に無効化されます。

このパラメータは、ソース・データベースとターゲット・データベースの両方が同じシステム上にある場合にのみ使用できます。異なるシステム上にある場合、ソース非CDBまたはPDBはAutoUpgradeがアクセスできないためクローズできません。

オプション

```
[yes | no]
```

デフォルト値は`yes`。

例

```
sales3.close_source=yes
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

del_after_upgrade_pfile

(オプション)アップグレード後にパラメータを削除する`PFILE`のパスおよびファイル名を指定します。

例

```
sales3.del_after_upgrade_pfile=/path/to/my/pfile_del.ora
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

del_during_upgrade_pfile

(オプション)アップグレード時にパラメータを削除するPFILEのパスおよびファイル名を指定します。

例

```
sales3.del_during_upgrade_pfile=/path/to/my/oldpfile.ora
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

drop_win_src_service

(オプション) Microsoft Windowsでのアップグレードの場合、アップグレード後にソースOracle DatabaseのWindowsオペレーティング・システム・サービスを削除するかどうかを指定します。

使用上のノート

デフォルトでは、Microsoft Windowsオペレーティング・システムでのOracle Databaseアップグレードの場合、AutoUpgradeによってWindows Oracle Databaseサービスが停止され、アップグレードが完了すると、サービスはそのまま残ります。サービスを停止したままにしておくと、データベースのMicrosoft Windowsサービスを再作成しなくても、データベースをソースOracleホームにリストアできます。ただし、アップグレードが正常に完了した後、ソース・データベースのMicrosoft Windowsサービスを自動的に削除することを選択できます。noが指定されているか、値が指定されていない場合、サービスはソース上で停止されますが、アップグレード後にそのまま残されます。

オプション

```
[yes | no]
```

デフォルト値はnoです。

例

```
upg1.drop_win_src_service=yes
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

env

(オプション) ORACLE_SID、ORACLE_HOME、ORACLE_BASEおよびTNS_ADMINを除く、オペレーティング・システムに設定されているカスタム・オペレーティング・システム環境変数を指定します。

使用上のノート

このパラメータを使用して、データベースsqlnet.oraファイルに示されている環境設定(Oracleウォレットに使用されるセキュア・ソケット・レイヤー暗号スイートなど)を指定します。複数設定するには、カンマで区切ります。

構文:

```
prefix=VARIABLE1=value1 [, VARIABLE2=value2, ...]
```

例

PDB sales2の場合、WALLET_LOCATIONの値はカスタム環境変数を使用して設定されるとします。

```
WALLET_LOCATION=
  (SOURCE=
    (METHOD=file)
    (METHOD_DATA=(DIRECTORY=/databases/wallets/$CUSTOM_ENV1/$CUSTOM_ENV2))
```

その場合、AutoUpgradeでそれらのカスタム環境変数の値を認識するには、envパラメータ(dir1は環境変数CUSTOM_ENV1によって指定されるパス、dir2はCUSTOM_ENV2によって指定されるパス)を使用して指定する必要があります。

```
sales2.env=CUSTOM_ENV1=dir1,CUSTOM_ENV2=dir2
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

ignore_errors

(オプション)アップグレードまたはパッチ適用プロセス中にAutoUpgradeで無視する特定のOracleエラーのカンマ区切りリストを指定できます。

使用上のノート

このパラメータを構成ファイルに追加すると、指定したアップグレード接頭辞のアップグレード中に、指定したエラー番号は無視されます。

例

```
sales3.ignore_errors=ORA-48181,ORA-00001
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

keep_source_pdb

(オプション)切断/接続アップグレード操作のソースPDBが、ソースCDBから削除されるのではなく、閉じた状態に保持されるかどうかを指定します。

使用上のノート

デフォルトでは、ソースPDBはアップグレード・プロセス中にソースCDBから削除されます。`keep_source_pdb`がYESに設定されている場合、ソースPDBは以前のリリース・システムから削除されません。パラメータ`target_pdb_copy_option`でコピー・オプションが指定されている場合のみ、パラメータをYESに設定できます。コピー・オプションを使用しない場合、PDBを削除する必要があるため、このパラメータは無視されます。コピーがない場合、既存のデータファイルは単一のCDBでのみ使用できます。

オプション

```
[yes | no]
```

デフォルト値はnoです。

例

```
sales1.keep_source_pdb=yes
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

log_dir

(オプション)パラメータの接頭辞で識別されるアップグレード・ジョブに含まれるデータベース・セットにあるデータベース・アップグレード用に生成されるログ・ファイルの場所を設定します。

使用上のノート

設定すると、AutoUpgradeは、指定したローカル・ログ・ファイル・パスに基づいて階層ディレクトリを作成します。たとえば、ジョブ識別子の接頭辞は`sales`で、`log_dir`は`upgrade-jobs`として識別され、`stage1`、`stage2`および`stagen`はアップグレードのステージを表します。

```
/u01/app/oracle/upgrade-jobs
                                        /temp/
                                        /sales/
                                        /sales/stage1
```

```
                                        /sales/stage2
                                        /sales/stagen
```

チルダ(~)など、パスにワイルド・カードを使用することはできません。完全なパスを使用する必要があります。

例

```
salesdb.log_dir=/u01/app/oracle/upgrade-jobs
```

デフォルトでは、グローバル構成ファイル・パラメータglobal.autoupg_log_dirが指定され、log_dirを指定しない場合、global.autoupg_log_dirで指定されたパスがデフォルトになります。

global.autoupg_log_dirもlog_dirも指定されていない場合、デフォルトでは、ログ・ファイルは構成ファイルに含めるデータベースのorabaseユーティリティで示される場所に配置されます。その場合、デフォルトのログ・ディレクトリはパスORACLE_BASE/cfgtoollogs/autoupgradeにあります。

構成ファイルに含まれるすべてのデータベースについてorabaseユーティリティが失敗した場合、ログ・ファイルの場所は、AutoUpgradeを実行しているユーザーのtempディレクトリに基づきます。

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

manage_standbys_clause

(オプション) DB_UNIQUE_NAMEで特定したスタンバイOracle Data Guardスタンバイ・データベースをAutoUpgradeプラグインのアップグレードから除外して、スタンバイ・データベース・ファイルを再利用できるようにするかどうかを指定します。

使用上のノート

スタンバイ・データベースを使用してデータベース構成をアップグレードする前に、潜在的な問題を減らすために、スタンバイ・データベースでAutoUpgradeを分析モードで実行することをお薦めします。

オプション

次の構文では、pdb-nameは、切断/接続アップグレードでターゲットCDBにアップグレードするソースPDBのDB_UNIQUE_NAMEです。

```
manage_standbys_clause=STANDBYS=[STANDBYS=NONE|STANDBYS=ALL|STANDBYS=ALL EXCEPT
('pdb-name', 'pdb-name', ...)|STANDBYS=('pdb-name', 'pdb-name', ...)]
```

デフォルト値はNONEです。

例

次の例では、Oracle Data Guardスタンバイのメンバーである非CDBまたはプラガブル・データベースは、AutoUpgradeプラグイン・アップグレードから除外されません。

```
upg2.sid=cdb1
upg2.pdbs=*
upg2.target_cdb=cdb21x
upg2.source_home=/source/18x
upg2.target_home=/target/21x
upg2.manage_standbys_clause=standbys=none
```

次の例では、すべてのスタンバイ・データベースでのデータ・ファイルへのREDO適用が、すべてのAutoUpgradeプラグイン・アップグレードで遅延されます。

```
upg3.sid=cdb2
```

```
upg3.pdbs=*
upg3.target_cdb=cdb21x
upg3.source_home=/source/18x
upg3.target_home=/target/21x
upg3.manage_standbys_clause=standbys=all
```

次の例では、AutoUpgradeプラグイン・アップグレード中に、データ・ファイルへのREDO適用がPDB cdb3_stby_1およびcdb3_stby_2を除いたすべてのスタンバイPDBで遅延されます。

```
upg4.sid=cdb3
upg4.pdbs=*
upg4.target_cdb=cdb21x
upg4.source_home=/source/12.2x
upg4.target_home=/target/21x
upg4.manage_standbys_clause=standbys=all except ('cdb3_stby_1','cdb3_stby_2')
```

次の例では、AutoUpgradeプラグイン・アップグレード中に、データ・ファイルへのREDO適用がスタンバイPDB cdb4_stby1でのみ遅延されます。

```
upg4.sid=cdb4
upg4.pdbs=*
upg4.target_cdb=cdb21x
upg4.source_home=/source/12.2x
upg4.target_home=/target/21x
upg4.manage_standbys_clause=standbys=('cdb4_stby_1')
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

pdbs

(オプション)アップグレードを実行するPDBのリストを設定します。このパラメータは、マルチテナント・アーキテクチャ(CDB)データベースのアップグレードにのみ適用されます。非CDBデータベースを接続およびアップグレードしている場合、このパラメータは無視されます。

使用上のノート

PDBリストはカンマ区切りです。リストには、PDB名またはアスタリスク(*)のいずれかを含めることができます。これは、AutoUpgradeの実行時にCDBでオープンしているすべてのPDBをアップグレードすることを示します。パラメータが指定されていない場合、デフォルト値は*です。

ANALYZEモードで実行している場合、AutoUpgradeはマウント状態のPDBを無視します。

FIXUPS、DEPLOYまたはUPGRADEモードで実行している場合、AutoUpgradeは実行モードに応じて、マウント状態のPDBを読取り/書込みモード、アップグレード・モードまたはその両方で開きます。

例

```
sales1.pdbs=pdb1, pdb2, pdbn
   upgr1.pdbs=*
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

raise_compatible

(オプション)アップグレードが正常に完了した後、Oracle DatabaseのCOMPATIBLE初期化パラメータをターゲット・リリースの

デフォルト値まで増やします。

使用上のノート

オプション:

- Y: COMPATIBLEパラメータ設定をターゲット・リリースまで増やします
- N: COMPATIBLEパラメータ設定をターゲット・リリースまで増やしません

デフォルトはNです。

注意:

- COMPATIBLE パラメータを増やした後、データベースのダウングレードはできません。
- COMPATIBLE パラメータを現在のリリース・レベルに設定するのは、アップグレードしたデータベースを完全にテストした後にのみ行うことをお薦めします。
- autoupgrade コマンドライン・パラメータ restore に使用する値に関係なく、構成ファイル・パラメータ raise_compatible の値を yes に設定した場合は、アップグレードを開始する前に、作成した保証付きリストア・ポイントを手動で削除する必要があります。アップグレードが正常に完了すると、アップグレードを開始する前に AutoUpgrade によって作成された保証付きリストア・ポイントは自動的に削除されます。AutoUpgrade によって POSTUPGRADE ステージが開始されると、データベースをリストアする方法はありません。

例

```
sales1.raise_compatible=yes
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

remove_rac_config

(オプション)ターゲットCDBホームへの正常な変換後にソースOracleホームのクラスタウェアから非CDB Oracle RACデータベースを削除するか、ソース・データベースを変更せずに残すかを指定します。

使用上のノート

デフォルトでは、非CDBのソースOracle RACデータベース構成は、アップグレード・プロセス中にCDBに移行されると、ソースOracle Grid Infrastructureから削除されます。remove_rac_configがnoに設定されている場合、ソースOracle RACデータベースは以前のリリースの非CDBシステムから削除されません。

オプション

```
[yes | no]
```

デフォルト値はyes。

例

```
upg1.remove_rac_config=no
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

remove_underscore_parameters

(オプション)構成ファイル内のすべてのOracle Databaseについて、アップグレード中およびアップグレード後にPFILEファイルからアンダースコア(非表示)パラメータを削除します。

使用上のノート

アンダースコアのパラメータは、Oracleサポートの指示がある場合のみ使用できます。

オプション

[yes | no]

デフォルト値はnoです。

例

```
sales1.remove_underscore_parameters=yes
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

replay

(オプション)リプレイを使用してデータベースをアップグレードするかどうかを指定します。

使用上のノート

デフォルトでは、AutoUpgradeはクラシック・アップグレードを実行してデータベースをアップグレードします。

オプション

[yes | no]

デフォルト値はnoです。

例

```
upg1.replay=yes
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

restoration

(オプション)データベースのリストアに保証付きリストア・ポイント(GRP)を生成します。

使用上のノート

restoration=noを設定する場合は、データベースのバックアップおよびリストアを手動で実行する必要があります。このオプションは、NOARCHIVELOGモードで動作するデータベース、およびOracle Flashback Technology機能のフラッシュバック・データベースをサポートしていないStandard EditionおよびStandard Edition 2データベースに使用します。パラメータを指定しない場合、デフォルト値(yes)が使用され、保証付きリストア・ポイントが作成されます。

オプション

[yes | no]

デフォルト値はyes。

例

```
sales1.restoration=no
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

revert_after_action

(オプション)接頭辞によって指定された特定のデータベース・ジョブに対してシステムのリストアが完了し、データベースが稼働した後にオペレーティング・システムで実行するカスタム・アクションを指定します。

使用上のノート

`revert_after_action`で指定するアクションは、リストア・プロセスが完了し、データベースが稼働した後、ターゲットのOracleホーム・バイナリで実行されます。

実行するように指定するスクリプトは`name.ext` (例: `myscript.sh`)の形式である必要があります。これにより、AutoUpgradeは実行するスクリプトのタイプを識別できます。許可される拡張子オプションは、次のとおりです。

- UNIXシェル(`.sh`)
- Microsoft Windowsバッチ(`.bat`、`.cmd`)
- Microsoft Windows PowerShell (`.ps1`)
- Oracle SQLスクリプト(`.sql`)。`revert_before_action`パラメータ接頭辞で指定されたデータベースのローカル操作の場合。

オプション

失敗時に停止します: `[Y|N]`。デフォルトはNです。

デフォルトでは、指定したスクリプトが失敗すると、AutoUpgradeは引き続き実行されます(`N`)。スクリプトが失敗した場合にAutoUpgradeが停止するように指定するには、`Y`フラグを使用します。スクリプトが`0`以外のステータスでオペレーティング・システムで実行を終了した場合、AutoUpgradeはスクリプトを失敗として識別します。

例

AutoUpgradeがリストアの処理を完了した後に、オペレーティング・システムで指定したスクリプトを実行します。スクリプトが失敗した場合にAutoUpgradeを停止するには、`Y`フラグを設定します。

```
sales3.revert_after_action =/user/path/script.sh Y
```

AutoUpgradeがリストアの処理を完了した後に、オペレーティング・システムで指定したスクリプトを実行します。フラグを指定しないと、デフォルトの失敗時に停止するオプションはNであるため、スクリプトが失敗した場合、AutoUpgradeは引き続き実行されます。

```
sales3.revert_after_action =/user/path/script.sh
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

revert_before_action

(オプション)接頭辞によって指定された特定のデータベース・ジョブに対してシステムのリストアが完了し、データベースが稼働する前にオペレーティング・システムで実行するカスタム・アクションを指定します。

使用上のノート

`revert_before_action`で指定するアクションは、データベースのリストアが開始されてデータベースが稼働する前に、ター

ゲットのOracleホーム・バイナリで実行されます。

実行するように指定するスクリプトはname.ext (例: myscript.sh)の形式である必要があります。これにより、AutoUpgradeは実行するスクリプトのタイプを識別できます。許可される拡張子オプションは、次のとおりです。

- UNIXシェル(.sh)
- Microsoft Windowsバッチ(.bat、.cmd)
- Microsoft Windows PowerShell (.ps1)
- Oracle SQLスクリプト(.sql)。revert_before_actionパラメータ接頭辞で指定されたデータベースのローカル操作の場合。

オプション

失敗時に停止します: [Y|N]。デフォルトはNです。

デフォルトでは、指定したスクリプトが失敗すると、AutoUpgradeは引き続き実行されます(N)。スクリプトが失敗した場合にAutoUpgradeが停止するように指定するには、Yフラグを使用します。スクリプトが0以外のステータスでオペレーティング・システムで実行を終了した場合、AutoUpgradeはスクリプトを失敗として識別します。

例

AutoUpgradeがリストアを開始する前に、オペレーティング・システムで指定するスクリプトを実行します。スクリプトが失敗した場合にAutoUpgradeを停止するには、Yフラグを設定します。

```
sales3.revert_before_action =/user/path/script.sh Y
```

AutoUpgradeがリストアを開始する前に、オペレーティング・システムで指定したスクリプトを実行します。フラグを指定しないと、デフォルトの失敗時に停止するオプションはNであるため、スクリプトが失敗した場合、AutoUpgradeは引き続き実行されます。

```
sales3.revert_before_action =/user/path/script.sh
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

run_hcheck

(オプション)アップグレード前のチェックの一部としてOracle Dictionary Health Checksを実行して、データベース・ディクショナリの不整合を特定するかどうかを指定します。

使用上のノート

データベース・ディクショナリの不整合を特定しやすくするために、AutoUpgradeがアップグレード前のチェックの一部として、ソース・データベースでDBMS_HCHECK PL/SQLパッケージを実行するように指定できます。設定した場合、AutoUpgrade run_hcheckパラメータにより、AutoUpgradeがデータベース・ディクショナリでOracle Dictionary Health Checksの配列全体を実行するか、または最もクリティカルなチェック・セットのみを実行するかを、アップグレード・ソース・データベースごとに指定できます。チェックによってデータベース・ディクショナリの潜在的またはクリティカルな問題が検出された場合、アップグレードの開始が回避されます。

Oracle Dictionary Health Checkの結果は、AutoUpgrade事前チェック・ディレクトリの下にdbname_healthcheck_result.logという形式で格納されます。dbnameは、チェックが実行されたデータベースの名前です。Oracle Dictionary Health Check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』のDBMS_HCHECKパッケージ・ドキュメントを参照してください。

オプション

`[full| critical]`

パラメータが設定されていない場合、デフォルトではDBMS_HCHECKを実行しません。

例

```
sales1.run_hcheck=full
sales2.run_hcheck=critical
```

関連項目

- DBMS_HCHECK

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

run_utlrp

(オプション)アップグレード後の処理の一部として、utlrp.sqlの1つのバージョンの実行を有効または無効にして、Oracle管理スキーマの無効なオブジェクトのみを再コンパイルします。

使用上のノート

utlrpユーティリティは、データベース・アップグレード時に無効になるすべてのデータ・ディクショナリおよびユーザー・オブジェクトを再コンパイルします。run_utlrp=noを設定した場合、または無効なユーザー・オブジェクトも再コンパイルする必要がある場合は、AutoUpgradeでアップグレードした後に、このユーティリティを使用して無効なオブジェクトを再コンパイルすることをお薦めします。

オプション

`[yes | no]`

デフォルト値はyes。

例

```
prefix.run_utlrp=yes
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

sid

(必須)アップグレードするデータベースのOracleシステム識別子(SID)を示します。

例

```
sales1.sid=salesdb
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

skip_tde_key_import

(オプション) yesに設定すると、アップグレードは実行されますが、ソース・データベースKeyStoreのターゲット・データベースへのインポートはスキップされ、エラーは発生しません。

使用上のノート

ノート:

このパラメータは不要になったため、非推奨です。今後の AutoUpgrade リリースで削除される可能性があります。このパラメータを使用するかわりに、-load_password コマンドライン・オプションを使用して TDE パスワードを AutoUpgrade のキーストアに追加するか、TDE パスワードを Secure External Password Store (SEPS) に追加することをお薦めします。

このオプションは、非CDBからPDBへの操作および切断/接続操作に使用できます。ソース・データベースKeyStoreをターゲット・データベースへのインポートをスキップする場合、AutoUpgradeではPDBをアップグレード・モードでオープンしたままにするため、キーを手動でインポートできます。キーをインポートした後、通常モードでデータベースを再起動する必要があります。

オプション

[yes | no]

デフォルト値はnoです。

例

```
sales1.skip_tde_key_import=yes
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

source_base

(オプション)ソースOracleホームのソースORACLE_BASEパスを指定します。

例

```
source_base=/u01/app/oracle
sales4.source_base=/u04/app/oracle4
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

source_dblink

(オプション)切断/接続再配置(ホット・クローン)アップグレード用に設定されたデータベース・リンクを指定します。

使用上のノート

非CDBまたはPDBに対して切断/接続再配置アップグレードを設定するには、まずソース・データベースとターゲット・データベースの場所間にデータベース・リンクを設定する必要があります。次に、source_dblinkパラメータを使用して、そのデータベース・リンクをAutoUpgradeに渡します。データベース・リンクに関連付けられているソース・データベース名は、source_dblink.パラメータの接尾辞として識別します。データベースがデータベース・リンクからリフレッシュされる時間値を秒単位で指定することもできます。

ノート:

このオプションは、Oracle Database 12.1.0.2 以降のソース・データベース・リリースで使用できます。target_pdb_copy_optionパラメータを使用すると、source_dblinkパラメータがアクティブになります。source_dblinkを使用する場合は、file_name_convertパラメータの値も指定して、ファイル名を変換するように指定するか、ファイル名を変換しないように指定する必要があります。file_name_convertがnoneに設定されている場合、

データベース・ファイルを配置する場所を指定するためにdb_create_file_destも設定する必要があります。

また、ソース・データベースのデータベース・リンクでターゲット・データベースが更新される頻度を指定するリフレッシュ間隔を秒単位で設定することもできます。start_timeパラメータとともにリフレッシュ間隔を使用すると、ターゲットの場所に対してソース・データベースをリフレッシュするように維持できます。リフレッシュ率が指定されていない場合、ソース・データベースは1回のみクローニングされ、リフレッシュは行われません。リフレッシュ率が指定されているが、start_timeパラメータを使用して将来の開始時間を指定しない場合、リフレッシュ間隔値は無視され、データベースは1回のみクローニングされます。

オプション

- (必須)接尾辞として指定されたソース・データベース名。
- (必須)作成したデータベース・リンクの名前。
- (オプション)ソース・データベースからのターゲット・データベースへのリフレッシュ率(秒)。リフレッシュ率を指定する場合は、通常、start_timeパラメータを使用して将来の開始時間も指定します。
- (オプション) CLONE_ONLY。このオプションを追加すると、作成されるPDBはリフレッシュされないクローンになり、クローン操作が完了した直後にアップグレードが開始するように指定されます。このオプションは、ソースがOracle Database 12.1 (リリース12.1.0.2)の場合に必要です。

例

次の例では、2つのデータベース・リンクが作成されます。

- pdbxcdb18x_link。pdbxという名前のPDBソース・データベースに作成されます。

```
CREATE DATABASE LINK pdbxcdb18x_link CONNECT TO remote-user IDENTIFIED BY
password
USING'(DESCRIPTION =(ADDRESS = (PROTOCOL = TCP)(HOST
GRANT CREATE SESSION, CREATE PLUGGABLE DATABASE, SELECT_CATALOG_ROLE TO remote-
user;
GRANT READ ON sys.enc$ TO remote-user;
```

- db18x_link。db18xという名前の非CDBソース・データベースに作成されます。

```
CREATE DATABASE LINK db18x_link CONNECT TO remote-user IDENTIFIED BY password
USING'(DESCRIPTION =(ADDRESS = (PROTOCOL = TCP)(HOST = db-node1)(PORT = 1521))
(CONNECT_DATA = (SERVICE_NAME = db18x)))';
```

AutoUpgrade構成ファイルでは、データベース・リンクに関連付けられたデータベース名は、その名前をsource_dblinkの接尾辞として使用して指定します(PDBソース・データベースの場合、接尾辞はpdbxで、非CDBソース・データベースの場合、接尾辞はdb18x)。

次の例では、source_dblinkを使用して、ソース・データベースpdbxのdblinkを指定します。時間間隔が指定されていないため、AutoUpgradeの起動直後にPDBアップグレード・デプロイメントが開始されます。

```
upg1.source_dblink.pdbx=pdbxcdb18x
```

同じ構成ファイルを使用して、AutoUpgradeは、AutoUpgradeがコマンドラインから起動された1時間40分後に、db18xという名前のデータベースのアップグレードを開始します。AutoUpgradeが起動されてからstart_timeで指定されたデプロイメント時間までの間、クローニングされたターゲット・データベースはソースから20秒ごとにリフレッシュされます。

```
upg1.source_dblink.db18x=db18x_link 20
upg1.start_time=+1h40m
```

次の例では、ソース・データベースdb18xがターゲットPDB db18x_linkにクローニングされ、そのソース・データベースが正常にクローニングされた直後にアップグレードが開始されます。

```
upg1.source_dblink.db18x=db18x_link CLONE_ONLY
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

source_home

(分析モード、修正モードおよびデプロイ・モードの場合は必須です。アップグレード・モードの場合はオプションです。)アップグレードするデータベースの現在のOracleホーム(ORACLE_HOME)。

使用上のノート

upgradeモードの場合、ソース・ホームとターゲット・ホームの値は同じパスにできます。

例

```
sales2.source_home=/path/to/my/source/oracle/home
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

source_ldap_admin_dir

(オプション)ソース・データベース・ホームのLDAP_ADMINディレクトリへのパスを指定します。

使用上のノート

WindowsではLDAP_ADMIN環境変数がレジストリ内で設定されているため、このパラメータはMicrosoft Windowsには影響しません。

例

```
sales1.source_ldap_admin_dir=/u01/app/oracle/12.2/dbhome01/ldap/admin
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

source_tns_admin_dir

(オプション)ソース・データベース・ホームのTNS_ADMINディレクトリへのパスを指定します。

使用上のノート

WindowsではTNS_ADMIN環境変数がレジストリ内で設定されているため、このパラメータはMicrosoft Windowsには影響しません。

例

```
sales1.source_tns_admin_dir=/u01/app/oracle/12.2/dbhome01/network/admin
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

start_time

(オプション)アップグレード・ジョブを実行する将来の起動時間を設定します。このパラメータは、アップグレード・ジョブをスケジュールしてサーバーの負荷を分散し、複数のジョブが即時に起動されないようにするために使用します。

使用上のノート

値は、now (すぐに開始)の形式にするか、英語日付書式の形式DD/MM/YYYYまたはMM/DD/YYYYにする必要があります。ここで、MMは月、DDは日、YYYYは年です。値を設定しない場合、デフォルトはnowです。

設定できる値:

```
now
30/12/2019 15:30:00
01/11/2020 01:30:15
2/5/2020 3:30:50
```

start_time値をnowに設定して複数のジョブが起動されると、AutoUpgradeではシステムで使用可能なリソースに基づいて起動時間をスケジュールします。ジョブの起動時間は数分単位でずれる場合があります。

日付要素または時刻要素に間違ったデリミネータを使用している値、または次のような間違った日付書式または時間書式を使用している値は無効です。

```
30-12-2019 15:30:00
01/11/2020 3:30:15pm
2020/06/01 01:30:15
```

例

```
sales1.start_time=now
sales2.start_time=07/11/2020 01:30:15
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

target_base

(オプション)ターゲットOracleホームのターゲットORACLE_BASEパスを指定します。

例

```
target_base=/u01/app/oracle
sales4.target_base=/u04/app/oracle4
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

target_cdb

(オプション)非CDB Oracle Databaseが接続されるターゲットCDBのSIDを指定します。このパラメータは、非CDBのOracleデータベースをアップグレードおよび変換する場合に必須です。

例

```
emp.target_cdb=salescdb
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

target_pdb_copy_option=file_name_convert

(オプション)非CDBデータベースをPDBに変換する場合、または別のソースCDBの既存のPDBを指定のターゲットCDBのPDBに変換する場合にAutoUpgradeが実行する、プラガブル・データベースを作成する文で使用されるfile_name_convertオプションを指定します。

使用上のノート

注意:

target_pdb_copy_option を指定すると、AutoUpgrade は必要に応じてリカバリを管理できます。target_pdb_copy_option が設定されておらず、デフォルトの nocopy オプションが使用されている場合、デフォルトの PDB はリカバリされません。必ず、ソース PDB をバックアップしてください。

このオプションは、ターゲットCDB内にプラガブル・データベースを作成する場合にのみ使用します。このパラメータを指定しない場合、パラメータのデフォルト値はNOCOPYであり、ソース・データベースの既存のデータ・ファイルが再利用されます。このパラメータを指定する場合は、ソース・データベース名またはPDB名を指定するパラメータに接尾辞を追加(target_pdb_copy_option.suffix)し、次のいずれかのオプションとともにfile_name_convert=を指定する必要があります。

- ソース・ファイル名(f)およびターゲット置換ファイル名(r)を指定する('f', 'r')か、NONEを指定します
- リフレッシュ可能なクローン・データベースを作成する場合は、ソース・データベース名またはPDB名を指定するパラメータに接尾辞を追加します(target_pdb_copy_option.suffix)

ターゲットCDBで、ASMを使用している場合、またはパラメータDB_CREATE_FILE_DESTまたはPDB_FILE_NAME_CONVERTを設定し、ターゲットCDBでこれらのパラメータを置換ファイル名に対して有効にする場合は、値prefix.target_pdb_copy_option.source-db-name-or-pdb=file_name_convert=NONEを設定します。

ターゲットCDBで変換時に1つ以上のデータ・ファイル名を変更する場合、構文prefix.target_pdb_copy_option.source-db-name-or-pdb=('f1', 'r1', 'f2', 'r2', . . .)を使用して、パラメータの値を入力してソース・データベース名またはPDB(接尾辞として指定)、変更するソース・ファイル名および既存のファイルのコピー先となるターゲット・ファイル名を指定します(f1は、ソースの最初のファイル名パターン、r1はターゲットCDBの最初の置換ファイル名パターン、f2はソースの2番目のファイル名パターン、r2はターゲットCDBの2番目の置換ファイル名パターン、となっていきます)。

構文

```
prefix.target_pdb_copy_option.source-db-name-or-pdb=file_name_convert=('f1', 'r1',
'f2', 'r2', 'f3', 'r3'...)
```

例

この例では、AutoUpgradeは、接頭辞文字列upg1および接尾辞salesを付けて指定されたデータベースの変換時に既存のデータファイルをコピーし、ファイルパス文字列およびファイル名/old/path/pdb_2をファイルパス文字列およびファイル名/new/path/depsalesで置換します。

```
upg1.target_pdb_copy_option.sales=file_name_convert=('/old/path/pdb_2',
'/new/path/depsales')
```

target_pdb_copy_optionsource-db-name-or-pdb=file_name_convertを使用してOMFファイルを変換するには、ターゲットOracleホームがOracle Database 19cリリース更新6以降(19.6.0)またはOracle Database 18cリリース更新10以降(18.10.0)である必要があります。

この例では、Oracle ASMに格納されているデータ・ファイルの内、Oracle管理ファイルとして格納されていないデータ・ファイルを+DATA/dbname/salesから+DATA/dbname/depsalesにコピーするように、パラメータが構成されています。

```
upg1.target_pdb_copy_option.sales=file_name_convert=('+DATA/dbname/sales',
'+DATA/dbname/depsales')
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

target_pdb_name

(オプション)ターゲットCDBに接続された後に、非CDBのソースOracle Databaseに割り当てる名前を指定します。

使用上のノート

このパラメータは省略可能です。これは、非CDBのOracle DatabaseをPDBにアップグレードして変換する場合、またはPDBをソース・リリースCDBから切断し、ターゲット・リリースCDBにアップグレードするために接続する場合に使用されます。

既存の非CDBデータベースをアップグレードしてターゲットCDBのPDBに変換する場合、`target_cdb`パラメータはターゲットCDBを指定するため必須です。ターゲットCDBでのPDBの作成方法を決定する場合は、オプションのパラメータ`target_pdb_name`および`target_pdb_copy_option`を使用して、ターゲットCDBでのPDBの作成方法を指定できます。ただし、いずれのオプション・パラメータも使用しない場合は、ソースCDBの完全アップグレードが実行されます。

非CDBをPDBに変換する場合のターゲットPDBのデフォルト名は、非CDB Oracle Databaseのデータベースの一意の名前を使用します。CDBへの接続時に非CDBの既存の名前とは異なる名前を指定するには、`target_pdb_name`を使用して新しい名前を設定します。さらに、リフレッシュ可能クローン・データベースを作成する場合は、ソース・データベース名またはPDB名を指定する接尾辞をパラメータに追加します(`target_name.suffix`)

例

次の例では、ソース非CDBデータベースは`emp19`です。target_pdb_nameパラメータを使用して、ターゲットCDBデータベースの名前を`emp23pdb`に変更します。

```
upg.target_pdb_name=emp23pdb
```

リフレッシュ可能クローンの場合は、クローンのソース・データベースを示す接頭辞を追加します。この例では、ソース・コンテナ・データベースは`db122b`で、`pdb1`を`db122b`からターゲット・コンテナ・データベース`db19`にクローニングしています。接尾辞`pdb1`は、`target_pdb_name`と`source_dblink`の両方の識別子として使用されます。`pdb1`接尾辞識別子は、ソース`pdb1`からターゲットPDB `PLUG122`にデータを移動するために使用されるターゲットPDB名とdblinkの両方を関連付けます。

```
global.autoupg_log_dir=/tmp/logs
upg1.source_home=/u01/app/oracle/122
upg1.target_home=/u01/app/oracle/19
upg1.sid=db122b
upg1.target_cdb=db19
upg1.pdbs=pdb1
upg1.target_pdb_name.pdb1=PLUG122
upg1.target_pdb_copy_option.pdb1=file_name_convert=('/u01/app/oracle/oradata/db122b/p
db1', '/u01/app/oracle/plug/pdb122b')
upg1.source_dblink.pdb1=pdbxcdb122x_link
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

target_ldap_admin_dir

(オプション)ターゲット・データベース・ホームの`LDAP_ADMIN`ディレクトリへのパスを指定します。

例

```
sales1.target_ldap_admin_dir=/u01/app/oracle/19/dbhome01/ldap/admin
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

target_tns_admin_dir

(オプション)ターゲット・データベース・ホームのTNS_ADMINディレクトリへのパスを指定します。

例

```
sales1.target_tns_admin_dir=/u01/app/oracle/19/dbhome01/network/admin
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

timezone_upg

(オプション)AutoUpgradeプロセスの一部としてタイムゾーンのアップグレードの実行を有効または無効にします。

使用上のノート

データ整合性を維持するには、データベースのアップグレード時にタイムゾーン・ファイル(DST)の設定をアップグレードすることをお薦めします。特に、timestamp with time zone表の列など、タイムゾーンに依存するデータがある場合、タイムゾーンをアップグレードします。この設定は、チェックリスト・ファイルで修正を上書きすることで無効にできることに注意してください。

AutoUpgrade構成ファイルでタイムゾーン・ファイルのアップグレードを明示的に無効にする場合は、このタスクをアップグレード計画の一環として実行するか、後で実行することをお薦めします。

オプション

[yes | no]

デフォルト値は、アップグレードの場合はyes、パッチ適用の場合はnoです。

例

```
sales1.timezone_upg=no
```

ノート:

RU 19.18 以降のパッチをデータベースに適用すると、更新済のタイムゾーン・ファイルがデフォルトで Oracle ホームにインストールされます。パッチが適用された Oracle ホームの Database Configuration Assistant (DBCA) で作成される新しいデータベースは最新のタイム・ゾーン・ファイルを使用して作成されます。

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

tune_setting

(オプション)指定したワークフロー・オプションに応じて、実行時にAutoUpgradeの動作を変更する特別なワークフローを有効にします。

使用上のノート

tune_settingパラメータを使用すると、アップグレード・ステップ、またはAutoUpgrade構成ファイルのパラメータ接頭辞で指定されたコンテナ・データベースまたはプラガブル・データベース(CDBまたはPDB)によって指定されたアップグレードの処理に割り当てられたリソースを微調整できます。この機能は、デフォルトのAutoUpgrade値がシステム要件に十分でないか、デフォルト以外のAutoUpgradeオプションを有効にする場合に、一部のアップグレードに役立ちます。

構文

```
prefix.tune_setting=option[, option, option, ...]
```

次のリストから、必要なAutoUpgradeランタイム・チューニングを提供するtune_settingオプションを選択します。複数のチューニング・オプションをtune_settingパラメータと組み合せるには、カンマ区切りを使用します。例:

```
sales3.tune_setting=proactive_fixups=true,query_hint_parallel=8,utlrp_threads_per_pdb
=8
```

ノート:

1 つの tune_setting エントリに複数のパラメータを連結できます

オプション	説明
active_nodes_limit	Oracle Real Application Clusters データベースの分散アップグレード中に使用するアクティブなクラスタ・メンバー・ノードの新しい合計を設定します。デフォルトは 2 です。指定した数がクラスタ・メンバー・ノードの最大数以上である場合、すべてのノードが取得されます。 sales3.tune_setting=active_nodes_limit=4
distributed_upgrade	AutoUpgrade が**分散アップグレード**を実行することを指定します。分散アップグレードでは、Oracle Clusterware クラスタ・メンバー・ノードのリソースを利用して、クラスタ上の PDB のアップグレードをより迅速に実行します。2 つ以上のノードの Oracle RAC クラスタ内の CDB をアップグレードする場合、このオプションを使用します。このオプションを選択すると、proactive_fixups オプションもデフォルトで有効になります。例: sales3.tune_setting=proactive_fixups=true,distributed_upgrade=true
make_pdbs_available	**事前修正**モードを使用するときに、事後修正の完了後、接頭辞で指定された PDB を読取り/書込みモードおよび非制限モードでオープンします。このオプションを使用すると、アップグレードの完了直後に接頭辞で指定された PDB がサービスで使用可能になり、他の PDB が引き続きアップグレードされます。これは大規模なフリート・アップグレード・デプロイメントに役立ちます。 注意: このオプションを選択すると、指定した PDB はユーザーからのサービス・リクエストを受け入れ、他の PDB は引き続きアップ

オプション	説明
	グレードされます。サービス・リクエストに対する PDB のレスポンス時間、および進行中の PDB アップグレードに必要な時間は、それぞれ影響を受ける可能性があります。 例: `sales3.tune_setting=make_pdbs_available=true`
`proactive_fixups`	アップグレードの最後のステージとして PDB がアップグレードされる、事前修正モードを有効にします。PDB の数がデータベースで定義されている CPU 数を 2 で割った数より多い場合、このチューニング・オプションを選択すると、アップグレードが速くなります。例: `sales3.tune_setting=proactive_fixups=true` 注意: CPU の数が PDB の数より多い場合、この設定を変更すると、パフォーマンスが向上しない可能性があります。
`query_hint_parallel`	接頭辞で指定された PDB の問合せ中に表領域からデータを収集するコードにパラレル・スレッド指定を指定します。これにより、接頭辞で指定された PDB にパラレル・スレッド数を増減して割り当てることができます。例: `sales3.tune_setting=query_hint_parallel=8` このオプションを選択すると、AutoUpgrade によってシステム・リソースの消費量が増える可能性があります。
`utlrp_threads_per_pdb`	CDB 内の無効なオブジェクトの再コンパイルによって生成されるデフォルトの最大スレッド数を上書きし、指定したスレッド数を使用します。例: `sales3.tune_setting=utlrp_threads_per_pdb=8` 注意: 指定されたスレッド数がシステムで使用可能なスレッド数を超

オプション	説明
	えると、パフォーマンスが低下する可能性があります。
utlrp_pdb_in_parallel	デフォルトの最大同時再コンパイル・スレッド数を、指定した数に上書きします。このオプションを使用して、無効なオブジェクトの再コンパイルのデフォルトの最大同時プロセス数を上書きします。例: sales3.tune_setting=utlrp_pdbs_in_parallel=2 注意: 各 PDB プロセスでは、utlrp_threads_per_pdb で指定された数と同じ数のスレッドがシステムで必要です。

例

次の例では、接頭辞sales3で指定されたデータベースのアップグレードは、Oracle Real Application ClustersのOracle Databaseインスタンスです。tune_settingパラメータを使用して、Oracle Grid Infrastructureクラスタ内の複数のCDB間でアップグレード・ロードを分散する設定distributed_upgradeを使用するように、これらのデータベース・インスタンスを設定します。

```
sales3.tune_setting=proactive_fixups=true,distributed_upgrade=true
```

次の例では、接頭辞sales3で指定されたデータベースのアップグレードは、複数のtune_settingパラメータ・オプションでチューニングされます。

```
sales3.tune_setting=proactive_fixups=true,query_hint_parallel=8,utlrp_threads_per_pdb
=8
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

upgrade_node

(オプション)現在のユーザー構成が有効なノードを指定します。デフォルト値はlocalhostです。

使用上のノート

このパラメータの目的は、AutoUpgradeで使用する構成ファイルにリストされているデータベースがAutoUpgradeで処理されないようにすることです。ここで、upgrade_nodeパラメータの値は現在のホスト名に対応していません。AutoUpgradeをリモートで実行することはできません。キーワードlocalhostをワイルド・カードとして使用して、ローカル・ホストのデータベースを処理するように指定できます。

ユース・ケース:

構成ファイルconfig.cfgには、10個のデータベースが含まれています。そのデータベースのうちの5つは、upgrade_nodeの値がdenver01に設定されています。残りの5つは、upgrade_nodeの値がdenver02に設定されています。AutoUpgradeが構成ファイルconfig.cfgを使用してサーバーdenver01で実行される場合、AutoUpgradeはupgrade_nodeがdenver01に設定されているデータベースのみを処理します。upgrade_nodeがdenver02に設定さ

れているデータベースは無視されます。hostnameユーティリティは、アップグレード・ノードを解決するために使用される値を識別します

例

```
hostname
denver02
sales1.upgrade_node=denver01
```

親トピック: AutoUpgrade構成ファイルのローカル・パラメータ

AutoUpgradeユーザー構成ファイルのグローバル・パラメータ

構成ファイルで扱われるすべてのOracle Databaseアップグレードのパラメータに対するデフォルトの動作を指定するには、オプションのAutoUpgradeグローバル・パラメータを使用できます。

使用上のノート

すべてのグローバル・パラメータはオプションです。すべてのグローバル・パラメータは接頭辞`global`を使用します。

`add_after_upgrade_pfile`および`del_during_upgrade_pfile`グローバルおよびローカルの`PFILE`パラメータ操作は、次の階層順序で実行されます。

1. グローバル・アクション
   a. グローバルの削除
   b. グローバルの追加
2. ローカル・アクション
   a. ローカルの削除
   b. ローカルの追加

表3-1 Oracle Database AutoUpgradeユーティリティのグローバル構成パラメータ

パラメータ	説明
`add_after_upgrade_pfile`	(オプション) `PFILE` をアップグレードした後にパラメータを追加する `PFILE` のパスおよびファイル名を指定します。この指定は、ユーザー構成ファイル内のすべてのデータベースに適用されます。 例: `global.add_after_upgrade_pfile=/path/to/my/add_after.ora`
`add_during_upgrade_pfile`	(オプション) `PFILE` のアップグレード中にパラメータを追加する `PFILE` のパスおよびファイル名を指定します。この指定は、ユーザー構成ファイル内のすべてのデータベースに適用されます。 `global.add_during_upgrade_pfile=/path/to/my/add_during.ora`
`after_action`	(オプション)すべてのアップグレード・ジョブが正常に終了した後に実行するカスタム・ユーザー・スクリプトのパスおよびファイル名を指定します。使用するスクリプトは `name.ext` (たとえば、`myscript.sh`)の形式である必要があります。これにより、AutoUpgrade は実行するスクリプトのタイプを識別できます。許可される拡張子オプションは、次のとおりです。 ● UNIX シェル(`.sh`)

パラメータ	説明

	● Microsoft Windows バッチ(`.bat`、`.cmd`)
	● Microsoft Windows PowerShell (`.ps1`)
	
	デフォルトでは、スクリプトが失敗すると、AutoUpgrade が引き続き実行されます。オペレーティング・システムがスクリプトの失敗を検出した場合、Y フラグを使用して AutoUpgrade が停止するように指定します。スクリプトが `0` とは異なるステータスで終了すると、失敗して完了したとみなされます。
	
	例:
	
	スクリプトが失敗した場合、AutoUpgrade を停止します。
	
	`global.after_action=/path/to/my/script.sh Y`
	
	スクリプトが失敗した場合、AutoUpgrade を続行します。
	
	`global.after_action=/path/to/my/script.sh`

`autoupg_log_dir`	(オプション) AutoUpgrade が使用するログ・ファイルとグローバル・モジュールに属する一時ファイルの場所を設定します。
	
	例:
	
	`global.autoupg_log_dir=/path/to/my/global/log/dir`
	
	AutoUpgrade 19.7 以降、特定の接頭辞のログ・ディレクトリにある `userconfig` ファイルに、異なるログ・ディレクトリ・パスを構成できます。
	
	`global.autoupg_log_dir=/path/to/my/global/log/dir`
	`myprefix.log_dir=global.auto_log_dir:different/path`
	
	この構文が使用されると、ログ・ファイルおよび一時ファイルは、接頭辞 `myprefix` で識別されるデータベースの次のパスに配置されます。
	
	`/path/to/my/global/log/dir/different/path`
	
	このパラメータをパスに設定しない場合、デフォルトでは、ログ・ファイルは構成ファイルに含めるデータベースの `orabase` ユーティリティで示される場所に配置されます。その場合、デフォルトのログ・ディレクトリはパス `ORACLE_BASE/cfgtoollogs/autoupgrade` にあります。
	
	構成ファイルに含まれるすべてのデータベースについて `orabase` ユーティリティが失敗した場合、ログ・ファイルの場所は、`AutoUpgrade` を実行しているユーザーの `temp` ディレクトリに基づきます。

パラメータ	説明
before_action	(オプション) アップグレード・ジョブを開始する前にすべてのアップグレードに対して実行するカスタム・ユーザー・スクリプトを指定します。使用するスクリプトは name.ext (たとえば、myscript.sh)の形式である必要があります。これにより、AutoUpgrade は実行するスクリプトのタイプを識別できます。特定のアップグレード・ジョブの前にスクリプトを実行する場合は、ローカル・パラメータ (local.before_action)を使用してそのスクリプトを指定します 許可される拡張子オプションは、次のとおりです。 ● UNIX シェル(.sh) ● Microsoft Windows バッチ(.bat、.cmd) ● Microsoft Windows PowerShell (.ps1) デフォルトでは、スクリプトが失敗すると、AutoUpgrade が引き続き実行されます。オペレーティング・システムがスクリプトの失敗を検出した場合、Y フラグを使用して AutoUpgrade が停止するように指定します。スクリプトが 0 とは異なるステータスで終了すると、失敗して完了したとみなされます。 例: スクリプトが失敗した場合、AutoUpgrade を停止します。 global.before_action=/path/to/my/script.sh Y スクリプトが失敗した場合、AutoUpgrade を続行します。 global.before_action=/path/to/my/script.sh
catctl_options	(オプション) AutoUpgrade が catctl.pl に送信してデフォルトの動作をオーバーライドするために選択できる catctl.pl オプションのセットを 1 つ以上指定します。オプションの詳細は、この表の最後にリンクされている「パラレル・アップグレード・ユーティリティ(catctl.pl)のパラメータ」を参照してください。 使用可能な catctl.pl オプション: ● -n パラレル操作に使用するプロセス数。非 CDB および CDB$ROOT のアップグレードの場合、デフォルトのプロセス数は 8 です。 ● -N PDB のアップグレード時に使用する SQL プロセッサの数。デフォルトのプロセス数は 2 です。 ● -t クラシック・アップグレードで SQL を実行し、デフォルトのリプレイ・アップグレード方法を上書き

パラメータ	説明
	● -T オフラインのユーザー・スキーマベースの表領域を取得します。 ● -z catcon.pm の本番デバッグ情報を有効にします。 例: global.catctl_options=-t -n 24 -N 4
del_after_upgrade_pfile	(オプション) PFILE のアップグレード後にパラメータを削除する PFILE のパスおよびファイル名を指定します。この指定は、ユーザー構成ファイル内のすべてのデータベースに適用されます。 例: global.del_after_upgrade_pfile=/path/to/my/del_after.ora
del_during_upgrade_pfile	(オプション) PFILE のアップグレード中にパラメータを削除する PFILE のパスおよびファイル名を指定します。この指定は、ユーザー構成ファイル内のすべてのデータベースに適用されます。 例: global.del_during_upgrade_pfile=/path/to/my/del_during.ora
drop_grp_after_upgrade	(オプション)データベースのアップグレード後に、保証付きリストア・ポイント(GRP)を削除します。このオプションを選択した場合、アップグレードが正常に完了した後に GRP が削除されます。raise_compatible を yes に設定する場合は、パラメータ drop_grp_after_upgrade も yes に設定する必要があります。 オプション: [yes | no] デフォルト値は no です。 例: global.drop_grp_after_upgrade=yes
target_base	(オプション)ターゲット Oracle ホームのターゲット ORACLE_BASE パスを指定します。このパラメータを使用する必要があるのは、まれな場合のみです。 例: global.target_base=/u01/app/oracle sales4.target_base=/u04/app/oracle4

パラメータ	説明
raise_compatible	(オプション)アップグレードが正常に完了した後、compatible パラメータをターゲット・リリースのデフォルト値まで増やします。 オプション: [yes | no] デフォルト値は no です。 注意: ● COMPATIBLE パラメータを増やした後、データベースのダウングレードはできません。 ● COMPATIBLE パラメータを現在のリリース・レベルに設定するのは、アップグレードしたデータベースを完全にテストした後にのみ行うことをお薦めします。 ● autoupgrade コマンドライン・パラメータ restore に使用する値に関係なく、構成ファイル・パラメータ raise_compatible の値を yes に設定した場合は、アップグレードを開始する前に、作成した保証付きリストア・ポイントを手動で削除する必要があります。アップグレードが正常に完了すると、アップグレードを開始する前に AutoUpgrade によって作成された保証付きリストア・ポイントは自動的に削除されます。AutoUpgrade によって POSTUPGRADE ステージが開始されると、データベースをリストアする方法はありません。 ● raise_compatible を yes に設定する場合は、パラメータ drop_grp_after_upgrade も yes に設定する必要があります。 例: global.raise_compatible=yes
target_home	(分析および修正モードの場合はオプションです。アップグレードおよびデプロイ・モードの場合は必須です。)構成ファイルで指定されたすべてのデータベースのグローバル・ターゲット・ホームを設定します。同じ target_home を複数回指定しないようにするには、このオプションを使用します。このパラメータはローカルで上書きできます。 例: global.target_home=/target/Oracle/home
target_version	(オプション)AutoUpgrade でアップグレードを実行するターゲット・リリース・バージョンを指定します。AutoUpgrade では、このパラメータで指定したリリース・バージョン情報を使用して、アップグレード対象のターゲットの Oracle Database リリースに対して正しいチェックおよび修正が使用されるようにします。このパラメータの形式は、有

パラメータ	説明

	効な Oracle バージョンのピリオド区切りの値です。
	
	有効な値
	
	● 12.2
	● 18
	● 19
	
	このオプションは、ターゲット・ホームがシステムに存在しないか、ターゲット・ホームが 12.2 リリースの場合のみ必要です。それ以外の場合、AutoUpgrade はターゲット・リリース値を導出できます。
	
	例:
	
	`global.target_version=18`
	`employees.target_version=12.2`

`upgradexml`	(オプション) `upgrade.xml` ファイルを生成します。オプション: `[yes | no]` `upgrade.xml` は、XML パラメータを指定したときにアップグレード前パッケージが生成した以前のリリースのファイルと同等です。このファイルは、分析モード(`mode -analyze`)中に作成されます。これは、AutoUpgrade ログ・ファイル用に定義された事前チェック・ディレクトリに生成されます。 例: `global.upgradexml=yes`

- add_after_upgrade_pfile
  (オプション) `PFILE`をアップグレードした後にパラメータを追加する`PFILE`のパスおよびファイル名を指定します。
- add_during_upgrade_pfile
  (オプション) `PFILE`のアップグレード中にパラメータを追加する`PFILE`のパスおよびファイル名を指定します。
- after_action
  (オプション)すべてのアップグレード・ジョブが正常に終了した後に実行するカスタム・ユーザー・スクリプトのパスおよびファイル名を指定します。
- autoupg_log_dir
  (オプション) ログ・ファイルと、AutoUpgradeが使用するグローバル・モジュールに属する一時ファイルの場所を設定します。
- before_action
  (オプション)アップグレード・ジョブを開始する前にすべてのアップグレードに対して実行するカスタム・ユーザー・スクリプトを指定します。
- catctl_options
  (オプション) AutoUpgradeが`catctl.pl`に送信してデフォルトの動作をオーバーライドするために選択できる`catctl.pl`オプションのセットを1つ以上指定します。
- del_after_upgrade_pfile

(オプション) PFILEのアップグレード後にパラメータを削除するPFILEのパスおよびファイル名を指定します。

- del_during_upgrade_pfile
  (オプション) PFILEのアップグレード中にパラメータを削除するPFILEのパスおよびファイル名を指定します。
- drop_grp_after_upgrade
  (オプション)データベースのアップグレード後に、保証付きリストア・ポイント(GRP)を削除します。
- keystore
  (オプション)AutoUpgradeがパスワードを格納するために排他的に使用する専用ソフトウェア・キーストアの場所と、その他の機密情報を指定します。
- raise_compatible
  (オプション)アップグレードが正常に完了した後、compatibleパラメータをターゲット・リリースのデフォルト値まで増やします。
- replay
  (オプション)リプレイを使用してデータベースをアップグレードするかどうかを指定します。
- target_base
  (オプション)ターゲットOracleホームのターゲットORACLE_BASEパスを指定します。
- target_home
  (分析および修正モードの場合はオプションです。アップグレードおよびデプロイ・モードの場合は必須です。)構成ファイルで指定されたすべてのデータベースのグローバル・ターゲット・ホームを設定します。
- target_version
  (オプション) AutoUpgradeでアップグレードを実行するターゲット・リリース・バージョンを指定します。
- upgradexml
  (オプション) upgrade.xmlファイルを生成します。

関連項目

- パラレル・アップグレード・ユーティリティ(catctl.pl)のパラメータ

親トピック: AutoUpgradeユーティリティ構成ファイル

add_after_upgrade_pfile

(オプション) PFILEをアップグレードした後にパラメータを追加するPFILEのパスおよびファイル名を指定します。

使用上のノート

この指定は、ユーザー構成ファイル内のすべてのデータベースに適用されます。

例

```
global.add_after_upgrade_pfile=/path/to/my/add_after.ora
```

親トピック: AutoUpgradeユーザー構成ファイルのグローバル・パラメータ

add_during_upgrade_pfile

(オプション) PFILEのアップグレード中にパラメータを追加するPFILEのパスおよびファイル名を指定します。

使用上のノート

この指定は、ユーザー構成ファイル内のすべてのデータベースに適用されます。

例

```
global.add_during_upgrade_pfile=/path/to/my/add_during.ora
```

親トピック: AutoUpgradeユーザー構成ファイルのグローバル・パラメータ

after_action

(オプション)すべてのアップグレード・ジョブが正常に終了した後に実行するカスタム・ユーザー・スクリプトのパスおよびファイル名を指定します。

使用上のノート

使用するスクリプトはname.ext (たとえば、myscript.sh)の形式である必要があります。これにより、AutoUpgradeは実行するスクリプトのタイプを識別できます。許可される拡張子オプションは、次のとおりです。

- UNIXシェル(.sh)
- Microsoft Windowsバッチ(.bat、.cmd)
- Microsoft Windows PowerShell (.ps1)

デフォルトでは、スクリプトが失敗すると、AutoUpgradeが引き続き実行されます。オペレーティング・システムがスクリプトの失敗を検出した場合、Yフラグを使用してAutoUpgradeが停止するように指定します。スクリプトが0とは異なるステータスで終了すると、失敗して完了したとみなされます。

スクリプトの出力が取得され、ファイルに格納されます。stdoutとstderrの両方が取得されます。ファイルは、特定のデータベースまたはジョブに一致するディレクトリのpostupgradeサブディレクトリに格納されます。

次の環境変数はスクリプトを実行するシェルに設定されます。

- ORACLE_SID
- ORACLE_UNQNAME
- ORACLE_BASE
- ORACLE_HOME
- TNS_ADMIN

例

スクリプトが失敗した場合、AutoUpgradeを停止します。

```
global.after_action=/path/to/my/script.sh Y
```

スクリプトが失敗した場合、AutoUpgradeを続行します。

```
global.after_action=/path/to/my/script.sh
```

親トピック: AutoUpgradeユーザー構成ファイルのグローバル・パラメータ

autoupg_log_dir

(オプション) AutoUpgradeが使用するログ・ファイルとグローバル・モジュールに属する一時ファイルの場所を設定します。

使用上のノート

特定の接頭辞のログ・ディレクトリにあるuserconfigファイルに、異なるログ・ディレクトリ・パスを構成できます

このパラメータをパスに設定しない場合、デフォルトでは、ログ・ファイルは構成ファイルに含めるデータベースのorabaseユーティリ

ティで示される場所に配置されます。その場合、デフォルトのログ・ディレクトリはパス
ORACLE_BASE/cfgtoollogs/autoupgradeにあります。

構成ファイルに含まれるすべてのデータベースについてorabaseユーティリティが失敗した場合、ログ・ファイルの場所は、AutoUpgradeを実行しているユーザーのtempディレクトリに基づきます。

例

```
global.autoupg_log_dir=/path/to/my/global/log/dir
```

特定の接頭辞のログ・ディレクトリにあるuserconfigファイルに、異なるログ・ディレクトリ・パスを構成します

```
global.autoupg_log_dir=/path/to/my/global/log/dir
myprefix.log_dir=global.auto_log_dir:different/path
```

この構文が使用されると、ログ・ファイルおよび一時ファイルは、接頭辞myprefixで識別されるデータベースの次のパスに配置されます。

```
/path/to/my/global/log/dir/different/path
```

親トピック: AutoUpgradeユーザー構成ファイルのグローバル・パラメータ

before_action

(オプション) アップグレード・ジョブを開始する前にすべてのアップグレードに対して実行するカスタム・ユーザー・スクリプトを指定します。

使用上のノート

使用するスクリプトはname.ext (たとえば、myscript.sh)の形式である必要があります。これにより、AutoUpgradeは実行するスクリプトのタイプを識別できます。特定のアップグレード・ジョブの前にスクリプトを実行する場合は、ローカル・パラメータ(local.before_action)を使用してそのスクリプトを指定します

許可される拡張子オプションは、次のとおりです。

- UNIXシェル(.sh)
- Microsoft Windowsバッチ(.bat、.cmd)
- Microsoft Windows PowerShell (.ps1)

デフォルトでは、スクリプトが失敗すると、AutoUpgradeが引き続き実行されます。オペレーティング・システムがスクリプトの失敗を検出した場合、Yフラグを使用してAutoUpgradeが停止するように指定します。スクリプトが0とは異なるステータスで終了すると、失敗して完了したとみなされます。

スクリプトの出力が取得され、ファイルに格納されます。stdoutとstderrの両方が取得されます。ファイルは、特定のデータベースまたはジョブに一致するディレクトリのpreupgradeサブディレクトリに格納されます。

次の環境変数はスクリプトを実行するシェルに設定されます。

- ORACLE_SID
- ORACLE_UNQNAME
- ORACLE_BASE
- ORACLE_HOME
- TNS_ADMIN

例

スクリプトが失敗した場合、AutoUpgradeを停止します。

```
global.before_action=/path/to/my/script.sh Y
```

スクリプトが失敗した場合、AutoUpgradeを続行します。

```
global.before_action=/path/to/my/script.sh
```

親トピック: AutoUpgradeユーザー構成ファイルのグローバル・パラメータ

catctl_options

(オプション) AutoUpgradeが`catctl.pl`に送信してデフォルトの動作をオーバーライドするために選択できる`catctl.pl`オプションのセットを1つ以上指定します。

使用上のノート

オプションの詳細は、「パラレル・アップグレード・ユーティリティ(`catctl.pl`)のパラメータ」を参照してください。

オプション

使用可能な`catctl.pl`オプション:

- `-n` パラレル操作に使用するプロセス数。リプレイ・アップグレードの場合、アップグレードに使用されるパラレル・プロセスの数は、`CPU_COUNT`を4で割った値にデフォルト設定されます。クラシック・アップグレードの場合、`CDB$ROOT`のデフォルトは8です。
- `-N` PDBのアップグレード時に使用するプロセッサの数。リプレイ・アップグレードの場合、アップグレードに使用されるパラレル・プロセスの数は、`CPU_COUNT`を4で割った値にデフォルト設定されます。クラシック・アップグレードの場合、デフォルトは2です
- `-T` オフラインのユーザー・スキーマベースの表領域を取得します。
- `-z` `catcon.pm`の本番デバッグ情報を有効にします。

例

```
global.catctl_options=-n 24 -N 4
```

関連項目

- パラレル・アップグレード・ユーティリティ(catctl.pl)のパラメータ
- catcon.plの構文およびパラメータ

親トピック: AutoUpgradeユーザー構成ファイルのグローバル・パラメータ

del_after_upgrade_pfile

(オプション) `PFILE`のアップグレード後にパラメータを削除する`PFILE`のパスおよびファイル名を指定します。

使用上のノート

この指定は、ユーザー構成ファイル内のすべてのデータベースに適用されます。

例

```
global.del_after_upgrade_pfile=/path/to/my/del_after.ora
```

親トピック: AutoUpgradeユーザー構成ファイルのグローバル・パラメータ

del_during_upgrade_pfile

(オプション) PFILEのアップグレード中にパラメータを削除するPFILEのパスおよびファイル名を指定します。

使用上のノート

この指定は、ユーザー構成ファイル内のすべてのデータベースに適用されます。

例

```
global.del_during_upgrade_pfile=/path/to/my/del_during.ora
```

親トピック: AutoUpgradeユーザー構成ファイルのグローバル・パラメータ

drop_grp_after_upgrade

(オプション)データベースのアップグレード後に、保証付きリストア・ポイント(GRP)を削除します。

使用上のノート

このオプションを選択した場合、アップグレードが正常に完了した後にGRPが削除されます。raise_compatibleをyesに設定する場合は、パラメータdrop_grp_after_upgradeもyesに設定する必要があります。

オプション

[yes | no]

デフォルト値はnoです。

例

```
global.drop_grp_after_upgrade=yes
```

親トピック: AutoUpgradeユーザー構成ファイルのグローバル・パラメータ

keystore

(オプション)AutoUpgradeがパスワードを格納するために排他的に使用する専用ソフトウェア・キーストアの場所と、その他の機密情報を指定します。

使用上のノート

キーストア・パラメータを使用して、AutoUpgradeが排他的に使用する専用ソフトウェア・キーストアを作成する場所を指定できます。

AutoUpgradeキーストアには、ファイルewallet.p12が格納されます(データベースで使用される他の種類のキーストアと同様)。ファイルは、TDEプロンプトでsaveコマンドを使用すると作成されます。自動ログイン・キーストアの生成を選択すると、ファイルcwallet.ssoも作成されます。自動ログイン・キーストアがある場合、AutoUpgradeの起動時にキーストア・パスワードの入力を求められません。

AutoUpgradeによって生成されたキーストアには機密情報が格納され、キーストアを初めて使用するときに選択したパスワードで保護されます。キーストアを変更するたびに、パスワードを指定する必要があります。AutoUpgradeの自動ログイン・キーストアを作成することにした場合を除き、AutoUpgradeを起動し、AutoUpgradeがキーストアからの情報を要求するたびに、キーストア・パスワードを指定する必要があります。

注意:

global.keystore で指定するディレクトリにはソフトウェア・キーストアが含まれているため、他のすべての高度にセキュアなキーストア・ファイルと同じセキュリティのベスト・プラクティスを使用して保護する必要があります。

例

次の例では、ORACLE_SIDを、キーストアを使用するデータベースのシステム識別子に置き換えます。

```
global.keystore=/etc/oracle/keystores/ORACLE_SID/autoupgrade
```

親トピック: AutoUpgradeユーザー構成ファイルのグローバル・パラメータ

raise_compatible

(オプション)アップグレードが正常に完了した後、compatibleパラメータをターゲット・リリースのデフォルト値まで増やします。

使用上のノート

このオプションを選択した場合、アップグレードが正常に完了した後にGRPが削除されます。raise_compatibleをyesに設定する場合は、パラメータdrop_grp_after_upgradeもyesに設定する必要があります。

注意:

- COMPATIBLEパラメータを増やした後、データベースのダウングレードはできません。
- COMPATIBLEパラメータを現在のリリース・レベルに設定するのは、アップグレードしたデータベースを完全にテストした後にのみ行うことをお薦めします。
- autoupgradeコマンドライン・パラメータrestoreに使用する値に関係なく、構成ファイル・パラメータraise_compatibleの値をyesに設定した場合は、アップグレードを開始する前に、作成した保証付きリストア・ポイントを手動で削除する必要があります。アップグレードが正常に完了すると、アップグレードを開始する前にAutoUpgradeによって作成された保証付きリストア・ポイントは自動的に削除されます。AutoUpgradeによってPOSTUPGRADEステージが開始されると、データベースをリストアする方法はありません。
- raise_compatibleをyesに設定する場合は、パラメータdrop_grp_after_upgradeもyesに設定する必要があります。

オプション

[yes | no]

デフォルト値はnoです。

例

```
global.raise_compatible=yes
```

親トピック: AutoUpgradeユーザー構成ファイルのグローバル・パラメータ

replay

(オプション)リプレイを使用してデータベースをアップグレードするかどうかを指定します。

使用上のノート

デフォルトでは、AutoUpgradeはクラシック・アップグレードを実行してデータベースをアップグレードします。

オプション

[yes | no]

デフォルト値はnoです。

例

```
global.replay=yes
```

親トピック: AutoUpgradeユーザー構成ファイルのグローバル・パラメータ

target_base

(オプション)ターゲットOracleホームのターゲットORACLE_BASEパスを指定します。

使用上のノート

このパラメータを使用する必要があるのは、まれな場合のみです。

例

```
global.target_base=/u01/app/oracle
sales4.target_base=/u04/app/oracle4
```

親トピック: AutoUpgradeユーザー構成ファイルのグローバル・パラメータ

target_home

(分析および修正モードの場合はオプションです。アップグレードおよびデプロイ・モードの場合は必須です。)構成ファイルで指定されたすべてのデータベースのグローバル・ターゲット・ホームを設定します。

使用上のノート

同じtarget_homeを複数回指定しないようにするには、このオプションを使用します。このパラメータはローカルで上書きできます。

例

```
global.target_home=/target/Oracle/home
```

親トピック: AutoUpgradeユーザー構成ファイルのグローバル・パラメータ

target_version

(オプション)AutoUpgradeでアップグレードを実行するターゲット・リリース・バージョンを指定します。

使用上のノート

AutoUpgradeでは、このパラメータで指定したリリース・バージョン情報を使用して、アップグレード対象のターゲットのOracle Databaseリリースに対して正しいチェックおよび修正が使用されるようにします。このパラメータの形式は、有効なOracleバージョンのピリオド区切りの値です。

有効な値

- 12.2

- 18
- 19
- 21

このオプションは、ターゲット・ホームがシステムに存在しないか、ターゲット・ホームが12.2リリースの場合のみ必要です。それ以外の場合、AutoUpgradeはターゲット・リリース値を導出できます。

例

```
global.target_version=19
employees.target_version=12.2
```

親トピック: AutoUpgradeユーザー構成ファイルのグローバル・パラメータ

upgradexml

(オプション) `upgrade.xml`ファイルを生成します。

使用上のノート

生成された`upgrade.xml`は、XMLパラメータを指定したときにアップグレード前パッケージが生成した以前のリリースのファイルと同等のものです。このファイルは、分析モード(`mode -analyze`)中に作成されます。これは、AutoUpgradeログ・ファイル用に定義された事前チェック・ディレクトリに生成されます。

オプション

`[yes | no]`

デフォルト値はnoです。

例

```
global.upgradexml=yes
```

親トピック: AutoUpgradeユーザー構成ファイルのグローバル・パラメータ

AutoUpgradeおよびOracle Databaseの構成オプション

AutoUpgradeを実行すると、データベースのタイプ(Oracle Database、Oracle ASMを備えたOracle Databaseスタンドアロンまたは Oracle RAC)が判別され、そのタイプのデータベースのアップグレードが実行されます

- 非CDBからPDBへのアップグレードのガイドラインおよび例
  変換前に、データファイルおよびデータベースをバックアップし、ソースのOracle Databaseリリースに関するガイドラインに従います。
- Oracle Grid Infrastructure管理構成のAutoUpgradeプロセス・フロー
  AutoUpgradeがOracle RAC、Oracle RAC One NodeまたはOracle Restartを検出すると、すべてのOracle RACインスタンスに必要なアップグレード・ステップの実行に進みます。
- AutoUpgradeを使用したアップグレードのためのOracle RAC要件
  AutoUpgradeがOracle Real Application Clusters (Oracle RAC)またはOracle RAC One Nodeデータベースをアップグレードできるかどうかを判断するには、ユースケース要件を確認します。
- AutoUpgradeを使用したOracle RACのアップグレードの準備
  アップグレード前に収集する必要がある情報およびその他のアップグレード準備のガイドラインを確認します。
- AutoUpgradeパッチ適用
  AutoUpgradeは、アウトオブプレース・パッチ適用を使用して、Oracle Databaseリリースにパッチを適用し、新しいリリース更新または月次推奨パッチにすることができます。
- AutoUpgradeとOracle Data Guard
  My Oracle SupportからダウンロードするAutoUpgradeユーティリティにより、Oracle Data Guard用に構成されたプライマリおよびセカンダリのデータベースのアップグレード・プロセスを簡略化できます。
- 高速デプロイ・オプションを使用してAutoUpgradeを実行する方法
  停止時間を最小限に抑えるには、高速デプロイ・オプションを使用してAutoUpgradeを実行することでデータベースをアップグレードできます。
- 暗号化されたPDBの切断/接続アップグレードの実行方法
  AutoUpgradeユーティリティを使用して、暗号化されたPDBの切断/接続アップグレードを実行する方法について説明します。
- 暗号化されたPDBの非CDBからPDBへの変換を実行する方法
  AutoUpgrade 22.1以降の更新では、AutoUpgradeにより、Transparent Data Encryption (TDE)を使用するOracle Databaseのアップグレードおよび変換が簡略化されます。

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

非CDBからPDBへのアップグレードのガイドラインおよび例

変換前に、データファイルおよびデータベースをバックアップし、ソースのOracle Databaseリリースに関するガイドラインに従います。

変換中にデータが失われないようにするために、AutoUpgradeを使用して非CDBのアップグレードおよび変換を実行する前に、アップグレード計画に時間をかけてバックアップ戦略を実装することをお薦めします

アップグレード計画のガイドライン

非CDBからPDBへの変換とアップグレードのプロセスはリカバリできません。適切なアップグレードと変換、および予期しない停止時間を減らすために、分析フェーズ中に見つかったエラー状態に対処することをお薦めします。

構成ファイル内で`target_pdb_copy_option`を使用してデータ・ファイルのコピーを作成する場合は、既存のデータベースをバックアップとして使用できます。これは安全なオプションですが、さらに時間とディスク領域が必要になります。AutoUpgrade構成ファイルの`target_pdb_copy_option`を設定していない場合、データベース変換は、既存のデータベース・ファイルで使用されているファイルの場所およびファイル名と同じものを使用します。潜在的なデータ損失を防ぐには、データをバックアップし、AutoUpgradeを起動する前にファイル配置計画を検討するようにします。

非CDBからマルチテナント・アーキテクチャへのGRPとアップグレード

- アップグレード中に、AutoUpgradeは、AutoUpgradeデプロイ・ワークフローのアップグレード・ステージのコンテキストでのみ使用可能な保証付きリストア・ポイント(GRP)を作成します。潜在的なデータ損失に備えて、AutoUpgradeを起動する前にバックアップ計画を実装する必要があります。
- 非CDBからマルチテナント・アーキテクチャへのデータベース変換は、AutoUpgradeの排出ステージで実行されます。このステージが完了した後、AutoUpgradeによって作成されたGRPは削除され、AutoUpgradeの`restore`コマンドを使用してデータベースをリストアすることはできません。以前の非CDBのOracleデータベース・リリースへのリカバリが必要な場合は、データベースを手動でリカバリする準備が必要です。

例3-4 マルチテナント・アーキテクチャを使用した非CDBからOracle Database 19cへのアップグレードおよび変換

デプロイ変換およびアップグレード・ワークフロー中に、AutoUpgradeはGRPを作成し、事前修正ステージを実行します。事前修正ステージ完了までのデプロイ・ワークフローのいずれかの部分が失敗した場合、AutoUpgradeはデプロイメントの開始時に作成されたGRPにデータベースをリストアできます。

ただし、事前修正ステージが完了すると、アップグレードされたデータベースはターゲット・リリースのOracle Databaseコンテナ・データベース(CDB)に接続されて変換が完了します。非CDBがCDBに接続されるとすぐに、GRPは有効でなくなり削除されます。

接続の間になんらかの問題が発生し、構成ファイル内で`target_pdb_copy_option`を使用してデータ・ファイルのコピーを作成しなかった場合、AutoUpgradeはデータベースをリカバリおよびリストアできないことに留意してください。その場合は、データベースを手動でリストアする必要があります。

親トピック: AutoUpgradeおよびOracle Databaseの構成オプション

Oracle Grid Infrastructure管理構成のAutoUpgradeプロセス・フロー

AutoUpgradeがOracle RAC、Oracle RAC One NodeまたはOracle Restartを検出すると、すべてのOracle RACインスタンスに必要なアップグレード・ステップの実行に進みます。

AutoUpgradeを起動すると、OracleデータベースがOracle Grid Infrastructureで構成されるときに、Oracle Real Application Clusters (Oracle RAC)のクラスタ・メンバー・ノードのメンバー、Oracle RAC One Node構成、またはスタンドアロン・サーバー(Oracle Restart)構成のためのOracle Grid Infrastructureのいずれかとして検出されます。

ノート:

このアップグレード・オプションを選択すると、AutoUpgrade がデータベース・インスタンスおよびシステム構成のアップグレードを完了している間に、クラスタ・データベースの停止時間が必要になります。Oracle Enterprise Manager を使用する場合は、アップグレード後に再構成する必要があります。

AutoUpgradeは、Oracle DatabaseがOracle Clusterwareリソースであることを検出すると、次のステップを順番に実行

します。

1. AutoUpgradeは、データベース、またはOracle RACデータベースのすべてのインスタンスを停止します。
2. AutoUpgradeは、Oracle RAC、Oracle RAC One NodeまたはOracle Restartの各サービスを無効にします。
3. Oracle ClusterwareリソースがOracle RACの場合、AutoUpgradeは、Oracle Clusterware内のすべてのOracle RACデータベース・クラスタ・メンバー・ノードのクラスタ・メンバーシップを無効にします。
4. AutoUpgradeは、Oracle Databaseインスタンスを起動します。
   - インスタンスがOracle RACクラスタ・メンバーである場合、アップグレード・モードでローカルのOracle Databaseインスタンスを起動し、クラスタ・パラメータをFALSEに設定します。
   - インスタンスが単一インスタンスのOracle Databaseの場合は、アップグレード・モードでインスタンスを起動します。
5. AutoUpgradeは、ローカルのOracle Database Oracleホーム・バイナリを新しいOracle Databaseリリース・バイナリにアップグレードします。
6. AutoUpgradeは、ローカルのOracle Databaseホームから`srvctl upgrade database`を実行し、Oracle RACの場合はOracle RACサービスの構成を新しいリリースにアップグレードします。
7. AutoUpgradeは、`srvctl enable database`を使用して、データベースのOracle Grid Infrastructureサービスを有効にします。Oracle RACの場合は、アップグレードされたOracle RACデータベースをクラスタ・メンバー・ノードとしてOracle RACクラスタに追加します。
8. AutoUpgradeは、更新されたパラメータと、リリース・アップグレードの影響を受けない環境に対して以前に設定したパラメータ・オプションを使用して、サーバー・パラメータ・ファイル(SPFILE)を再作成します。
9. Oracle DatabaseがOracle RACクラスタのメンバーであった場合、AutoUpgradeは、すべてのクラスタ・メンバーがアップグレードされてクラスタに追加され、各クラスタ・メンバー・ノードでSPFILEが再作成されるまで、他の各クラスタ・メンバー・ノードでこのプロセスを繰り返します。
10. AutoUpgradeは、Oracle Databaseを起動します。Oracle RACの場合は、クラスタ上のOracle Real Application Clustersのすべてのインスタンスを起動します。

ノート:

スタンドアロン・サーバー(Oracle Restart、Oracle RAC One Node または Oracle RAC Database)の Oracle Grid Infrastructure で AutoUpgrade を起動する前に、アップグレード先の Oracle Database リリースと同じリリースまたはそれより新しいリリースに Oracle Grid Infrastructure をアップグレードする必要があります。

親トピック: AutoUpgradeおよびOracle Databaseの構成オプション

AutoUpgradeを使用したアップグレードのためのOracle RACの要件

AutoUpgradeでOracle Real Application Clusters (Oracle RAC)またはOracle RAC One Nodeデータベースをアップグレードできるかどうかを判断するには、ユースケース要件を確認します。

Oracle RACデータベースでAutoUpgradeを使用するための要件

AutoUpgradeを使用して、Oracle RACまたはOracle Real Application Clusters One Nodeシステムのアップグレードを実行できます。ただし、システムが次の要件のすべてを満たしている必要があります。

- LinuxまたはUNIXベースのシステムである必要があります。Microsoft Windowsシステムはサポートされていません。

- 新しいOracle Databaseリリースにアップグレードするためのアップグレード要件を満たしている必要があります。
- サーバー制御(SRVCTL)ユーティリティを使用して登録および管理する必要があります。

データベース管理者がAutoUpgradeを使用するために必要なタスク

データベース管理者は、次のタスクを完了する必要があります。

- アップグレードによる問題によるデータ損失を防ぐために、適切なバックアップ計画を作成します。
- 必要に応じて、ローカルの`tnsnames.ora`リスナーとSCANリスナーの両方に対して、リスナーおよびTransparent Network Substrate (TNS)ファイルを構成します。
- Oracle Walletの証明書および管理(必要な場合)を構成し、自動ログインを構成します。

関連項目

- AutoUpgradeのための完全なデプロイの有効化

親トピック: AutoUpgradeおよびOracle Databaseの構成オプション

AutoUpgradeを使用したOracle RACのアップグレードの準備

アップグレード前に収集する必要がある情報およびその他のアップグレード準備のガイドラインを確認します。

Oracle Real Application Clusters (Oracle RAC)のアップグレードにAutoUpgradeを使用するには、Oracle Automatic Storage Management (Oracle ASM)もアップグレードされるため、アップグレードの前に必要な情報を収集し、アップグレード中に情報を指定する準備が整っていることを確認します。

AutoUpgradeおよびOracle RACのスコープ制限

- AutoUpgradeは、Oracle Grid InfrastructureのOracle Clusterwareコンポーネントのアップグレードを実行しません。AutoUpgradeを起動してOracle RACデータベースをアップグレードする前に、新しいリリースへのOracle Grid Infrastructureのアップグレードを正常に完了する必要があります。

AutoUpgradeを使用したアップグレード前のファイル・システムの準備

AutoUpgradeは、Oracle ASMで共有される`PFILE`ファイルおよび`SPFILE`ファイルを識別できます。AutoUpgradeは、SPFILEをアップグレードの一部として再作成します。Oracle ASMを使用してクラスタ上でファイルを共有する場合、この手順を完了する必要はありません。

親トピック: AutoUpgradeおよびOracle Databaseの構成オプション

AutoUpgradeパッチ適用

AutoUpgradeは、アウトオブプレース・パッチ適用を使用して、Oracle Databaseリリースにパッチを適用し、新しいリリース更新または月次推奨パッチにすることができます。

AutoUpgradeパッチ適用により、リリース更新(RU)および月次推奨パッチ(MRP)を適用できます。このAutoUpgradeプロシージャを使用すると、事前チェック、再開、リストアおよびOracle RAC管理のAutoUpgrade機能を利用して、複数のRUまたはMRP適用を1つのシームレスな操作で実行できます。AutoUpgradeパッチ適用により、企業へのOracle Database更新のデプロイを簡略化できます。

- AutoUpgradeによるAutoUpgradeパッチ適用の実行方法
  AutoUpgradeパッチ適用により、AutoUpgradeアップグレード・プロセスがパッチ適用まで拡張され、1つのコマンドを使用して複数のデータベースに対してアウトオブプレース・パッチ適用を実行できるようになります。

- AutoUpgradeパッチ適用の構成ファイルおよびログ・ファイル
  AutoUpgradeパッチ適用の構成ファイルおよびログ・ファイルの例を見てみましょう。

親トピック: AutoUpgradeおよびOracle Databaseの構成オプション

AutoUpgradeによるAutoUpgradeパッチ適用の実行方法

AutoUpgradeパッチ適用により、AutoUpgradeアップグレード・プロセスがパッチ適用まで拡張され、1つのコマンドを使用して複数のデータベースに対してアウトオブプレース・パッチ適用を実行できるようになります。

AutoUpgradeの最新リリースでは、AutoUpgradeパッチ適用プロシージャは、アプトオブプレース・パッチ適用を使用して、リリース更新(RU)、月次推奨パッチ(MRP)および個別パッチに対して実行できます。AutoUpgradeを使用して以前のRUまたはMRPからパッチを適用すると、AutoUpgradeの簡潔性、信頼性およびリカバリ性がパッチ適用プロセスにまで拡張されます。その結果、パッチ適用は実行しやすくなり、パッチのデプロイ中に発生する可能性のある問題からのリカバリも簡単になります。

パッチ適用を実行するためにAutoUpgradeの構成ファイルに設定する必要のある追加のパラメータまたはオプションはありません。ソースおよびターゲットのOracleホームを指定するだけで、AutoUpgradeはデータベースに変更を適用します。ソースとターゲットのOracleホームが同じOracle Databaseリリースである場合(19.11から19.13など)、AutoUpgradeは操作をRUまたはMRPパッチ操作として識別します。この機能は個別パッチにも適用されます。ターゲットOracleホームに適用されたRUまたはMRPがソースOracleホームのRUまたはMRPより新しい場合は、この方法を使用して、必要に応じてターゲットOracleホームに個別パッチを適用できます。

AutoUpgradeはパッチ操作を実行する際、Datapatchを使用してOracle Real Application Clusters (Oracle RAC)データベースなどのデータベースにRUまたはMRPを適用します。パッチ適用のプロセスでは、完全なリリース・アップグレードと同様に、保証付きリストア・ポイントの作成など、既存のAutoUpgradeオプションおよび操作を利用します。パッチ適用プロセス中、データベースはソースOracleホームで停止され、ターゲットOracleホームで再起動されます。RUまたはMRPの更新は、Datapatchを使用してアップグレード・モードで実行されます。

AutoUpgradeパッチ適用の利点

- AutoUpgradeを使用すると、構成ファイルに指定されたすべてのデータベースのパッチ適用が1回の操作でできるようになります。
- 再開機能は、AutoUpgradeにすでに組み込まれています。
- リストア機能は、旧リリースのOracleホームにロールバックできるように提供されています。
  - パッチ適用プロセス中に生成された保証付きリストア・ポイント(GRP)を使用してリストアします。
  - GRPが存在しない場合は、Datapatchのロールバック機能でもリストアを実行できます。AutoUpgradeは、GRPが存在しないことを検出すると、自動的にDatapatchを使用してリストアを実行します。Datapatchのロールバック・リストアを有効にするには、リストア構成オプションをnoに設定します。例: `sales1.restoration=no`。
- Oracle RAC管理は、AutoUpgradeで自動的に提供されます。
- AutoUpgradeのエラー管理レポート機能は、パッチ適用にまで拡張されています。
- AutoUpgradeのJSONステータスおよび進捗レポートは、パッチ適用にまで拡張されています。
- AutoUpgradeは、Datapatch JSONステータス・ファイルを使用して各プロセスの成否を判断し、プロセスの完了時にその結果をレポートします。
- トラブルシューティングおよびエラー・トリアージを簡略化するために、AutoUpgradeには、AutoUpgradeによって実行されるすべてのアクションおよびアクションが失敗した理由を特定する広範なトレース・ロギングが用意されています。

AutoUpgradeパッチ適用でサポートされている機能

AutoUpgradeパッチ適用には、次の機能が用意されています。

- Transparent Data Encryption (TDE)により暗号化されたデータベースのパッチ適用。
- ホット・クローニング/再配置。非CDBから、またはリモート・ホスト上のPDBからPDBを作成できるようになります。
- 事前修正。PDBにパッチが適用されるとすぐにPDBに対して事後修正を実行します。
- Oracle Real Application Clusters (Oracle RAC)を使用した分散データベース・アップグレード。これにより、複数のノード間でパッチ適用ワークロードを分散できます。
- ソースのOracleホーム構成ファイル(`tnsnames.ora`、`sqlnet.ora`およびその他のファイル)をターゲットのOracleホームにコピーまたはマージすることによる構成管理。
- パッチの準備が整っていることを確認するための、パッチ適用プロシージャを開始する前のデータベースの分析。
- 停止時間を短縮するための、パッチ適用前の本番での修正の実行。その後、`-mode upgrade`を使用して修正の実行をバイパスし、データベースへのパッチ適用に直接進むことができます。
- エンドツーエンドのパッチ適用を実現するための、すべてのパッチ適用タスクのデプロイ・モードでの実行。
- パッチ適用プロセスの拡張レポート。これにより、`datapatch_summary.log`レポートを使用してエラーをより簡単に診断できるようになります。
- DatapatchサマリーJSONファイルに含まれるDatapatchログ・ファイルは、次のような適用またはロールバックの操作ログの出力ファイルにコピーされます。
  - 適用ログの形式(dbnameはデータベースの名前): `applydatapatchlogfiles#dbname.log`
  - ロールバック・ログの形式(dbnameはデータベースの名前): `rollbackdatapatchlogfiles#dbname.log`
- Datapatch出力は、次の操作ログに記録されます。
  - 適用ログの形式(dbnameはデータベースの名前): `applyautoupgrade#dbname.log`
  - ロールバック・ログの形式(dbnameはデータベースの名前): `rollbackautoupgrade#dbname.log`
- Datapatch JSON出力は、次の操作ログに記録されます。
  - 適用ログの形式(dbnameはデータベースの名前): `applydatapatchsummary#dbname.log`
  - ロールバック・ログの形式(dbnameはデータベースの名前): `rollbackdatapatchsummary#dbname.log`
- AutoUpgradeによって生成されたすべての関連パッチ適用ログ・ファイルのストレージは、`dbupgrade`ディレクトリに格納されます。
- 標準のDatapatchログ・ファイルは、`Oracle-base/cfgtoollogs/sqlpatch`に格納されます(`Oracle-base`は、AutoUpgradeを実行しているユーザー・アカウントのOracleベース・ディレクトリです)。
- AutoUpgrade 23.1以降、パッチ適用はMicrosoft Windowsのデータベースでサポートされています。

AutoUpgradeパッチ適用を使用するための要件および制限事項

- AutoUpgradeパッチ適用を使用するには停止時間が必要です。
  - パッチ操作が完了すると、データベースは再起動されます。
  - パッチ操作中、デフォルトでは、すべてのPDBでパッチ適用プロセス全体が完了するまで、CDBおよびすべてのPDBは使用できません。
  - パッチが正常に適用されたらすぐにPDBを使用できるようにするには、必ず構成ファイルで`tune_settings`のオプション`make_pdbs_avaliable`を`true`に設定します。例: `sales3.tune_setting=proactive_fixups=true,make_pdbs_available=true`
- AutoUpgradeは、追加のパッチが適用されたターゲットRUまたはMRPをサポートできます。ただし、ソースOracle DatabaseのRUまたはMRPは、ターゲットのOracle DatabaseのRUまたはMRPよりも古い必要があります。
- AutoUpgradeは、OPatchコマンドを実行しません。そのため、AutoUpgradeを起動する前に、ターゲットOracleホームにパッチを完全に適用する必要があります。

- AutoUpgradeパッチ適用を使用してRUおよびMRPをターゲット・ホームに対して実行する前に、ターゲットOracle Databaseホームのインストールおよび構成が完了している必要があります。データベースに適用された後にターゲットRUにパッチが適用された場合、AutoUpgradeパッチ適用を使用できなくなります。かわりに、Datapatchを使用して更新を適用する必要があります。
- Oracle Database (単一インスタンスまたはOracle RAC)のローリング・パッチの実行はサポートされていません。単一インスタンスまたはOracle RAC Oracle DatabaseでAutoUpgradeパッチ適用を使用すると、AutoUpgradeのOracle RAC管理によって、すべてのノードでデータベースが停止されます。
- ターゲットOracle Database RUは、少なくともOracle Database 19c, RU 19.3以降のリリースである必要があります。AutoUpgradeパッチ適用を使用して、Oracle Databaseの旧リリースに対してRUまたはMRPを実行することはできません。
- Oracle Data Guardフィジカル・スタンバイまたはOracle Data Guardロジカル・スタンバイおよびローリング・アップグレードを使用したデータベースに対するAutoUpgradeパッチ適用の影響は、Datapatchを使用した場合と同じです。AutoUpgradeパッチ適用をプロシージャに含めることができます。
- データベースを別のターゲット・システムに移動する場合、AutoUpgradeパッチ適用を`-mode upgrade`オプションとともに使用できます。ただし、データベースが別のシステムに移動されると、AutoUpgradeはリストアを実行できません。その場合、パッチ適用のアップグレード・オプションは、データベースのアップグレードに適用されるのと同じルールに従います。
- このドキュメントでは、アップグレードを実行しているAutoUpgradeについて説明します。AutoUpgradeがパッチ適用を実行する場合、機能的にはパッチ適用はアップグレードと似ています。したがって、このガイドまたは図でのアップグレードについての言及は、AutoUpgradeパッチ適用によるアップグレードとパッチ適用の両方に適用されます。
- AutoUpgradeパッチ適用によるパッチ適用の実行では、アップグレードでのみサポートされている`preupgrade`モードを除くすべてのAutoUpgrade `-mode`オプションがサポートされています。
- 再配置可能なPDBへのパッチ適用またはアップグレードの場合、構成ファイルにはパッチ・クローンPDBとアップグレード済クローンPDBを混在させることはできません。クローンPDBの構成は、すべてアップグレード・クローンまたはすべてパッチ適用済クローンのいずれかである必要があります。
- AutoUpgradeパッチ適用では、`catctl_options`設定の`-n number` (`number`はパラレル操作に使用するプロセスの数)が1つのみサポートされます。`catctl_options=-n`設定を使用すると、パッチ適用プロセス中に同時に実行するPDBの合計数を制御できます。デフォルトは、`CPU_COUNT`を2で割った値です。たとえば、`CPU_COUNT`が24に設定されている場合、デフォルトでは、パッチ適用プロセス中に12個のPDB (Datapatchインスタンス)を同時に実行できます。

  デフォルトを上書きするには、`prefix.catctl_options`を構成ファイルに追加し、実行する同時Datapatchインスタンスの数の値を指定します。たとえば、デフォルトをオーバーライドして、接頭辞`sales`を付けて指定されたパッチ操作に対して6個のPDB (Datapatchインスタンス)を実行するようにAutoUpgradeを構成するには、次の行を構成ファイルに追加します。

  ```
  sales.catctl_options=-n 6
  ```

> 注意:
>
> 19.13 以前のリリース更新(RU)の場合、パッチ適用操作に `catctl_options` パラメータを設定することをお薦めします。
>
> 19.13 以前の RU では、各 Datapatch インスタンスに割り当てられる SQL*Plus プロセッサのデフォルト

値は、AutoUpgrade が起動する Datapatch インスタンスごとに CPU_COUNT に 2 を掛けた数(CPU_COUNT*2)のプロセッサです。このデフォルトの SQL*Plus プロセッサ割当てにより、すぐにシステムが過負荷状態になる可能性があります。パッチ操作に割り当てられるシステム・リソースを制限するには、同時に実行される Datapatch インスタンスの数を制限することが唯一の選択肢です。RU 19.13 以降では、Datapatch のインスタンスごとに 1 つの SQL*Plus プロセスのみが起動されます。この変更により、AutoUpgrade は大量のシステム・リソースを消費することなく、より多くの PDB をパラレルで実行できます。

AutoUpgradeパッチ適用のセキュリティ特性

- AutoUpgradeパッチ適用を実行しているユーザーには、データベースにログインしてパッチ適用操作を実行するためにSYSDBAシステム権限が必要です。
- アップグレードに適用されるのと同じセキュリティ・ルールが、AutoUpgradeパッチ適用でも適用されます。

AutoUpgradeパッチ適用のパフォーマンス特性

- RUまたはMRPのデプロイ速度は、CDB内のPDBの数およびリリース更新またはリリース更新リビジョンの変更内容によって左右されます。ソースRUまたはMRPパッチからの変更数が比較的少ない場合は、パッチのデプロイメントは速くなります。パッチに多数の変更がある場合は、パッチの適用にさらに時間が必要となります。
- AutoUpgradeパッチ適用には多数の追加の自動化プロシージャが組み込まれているため、リリース更新のデプロイメントは、手動でDatapatchを実行する場合よりもわずかに遅くなります。たとえば、AutoUpgradeパッチ適用には、無効なオブジェクトの再コンパイルおよびターゲット・システムでのOracle RAC管理の構成が自動的に組み込まれています。

AutoUpgradeパッチ適用でリリース更新がロールバックされた場合の処理

- AutoUpgradeは、パッチ適用プロセス中に生成された保証付きリストア・ポイント(GRP)を使用して、旧リリースのOracleホームにロールバックします。
- GRPが作成されていない場合、AutoUpgradeは自動的にDatapatchをコールして変更をロールバックします。

親トピック: AutoUpgradeパッチ適用

AutoUpgradeパッチ適用の構成ファイルおよびログ・ファイル

AutoUpgradeパッチ適用の構成ファイルおよびログ・ファイルの例を見てみましょう。

AutoUpgradeパッチ適用の構成ファイルは、アップグレード用のAutoUpgrade構成ファイルと基本的に同じです。ただし、指定されたパラメータを使用してアップグレードを実行するかわりに、AutoUpgradeはソースOracle Databaseパッチ・リリースからターゲットOracle Databaseパッチ・リリースにパッチ操作を実行し、プロシージャの一部としてDatapatchを実行します。

例3-5 様々なパッチ適用シナリオ用のAutoUpgrade構成ファイル

次の構成ファイルの例では、次のOracle Databasesに、リリース更新(RU) 11から19c RU 13にパッチが適用されたOracle Database 19cからパッチが適用されます。

- 19.11.0から19.13.0にパッチが適用され、接頭辞patch1が指定された非CDBデータベース
- 19.11から19.13にパッチが適用され、接頭辞patch2が指定されたOracle Database 19c CDB
- 切断/接続によってパッチが適用され、接頭辞patch3が指定された暗号化されたPDB
- パッチが適用されてCDBに変換され、接頭辞patch4が指定された非CDB
- パッチ適用操作中に再配置されるPDBが含まれる、接頭辞patch5が指定されたCDB
- 接頭辞patch6が指定されたOracle Real Application Clusters (Oracle RAC)データベース

- 接頭辞patch7が指定された、分散クラスタ構成内のOracle RACデータベース

```
#
# Global log directory for patch logs
#
global.autoupg_log_dir=/databases/patchlogs
#
# Non-CDB patch to Non-CDB patch, source and target home
#
patch1.sid=db19
patch1.source_home=/databases/ee/product/1911/dbhome_1
patch1.target_home=/databases/ee/product/1913/dbhome_2
#
#
# CDB patch, Source and Target home
#
patch2.sid=cdb19
patch2.source_home=/databases/ee/product/1911/dbhome_1
patch2.target_home=/databases/ee/product/1913/dbhome_2
#
# Unplug-Plug with KeyStore
#
global.keystore=/databases/tde
patch3.sid=cdb19
patch3.source_home=/databases/ee/product/1911/dbhome_1
patch3.target_home=/databases/ee/product/1913/dbhome_2
patch3.target_cdb=cdb1913
patch3.target_pdb_name=sales
patch3.target_pdb_copy_option=file_name_convert=('/databases/ee/oradata/CDB19/sales',
'/databases/ee/oradata/CDB1913/sales')
#
# Non-CDB to CDB, Source and Target home
#
patch4.sid=db19
patch4.source_home=/databases/ee/product/1911/dbhome_1
patch4.target_home=/databases/ee/product/1913/dbhome_2
patch4.target_cdb=cdb1913
patch4.target_pdb_name=emp
patch4.target_pdb_copy_option=file_name_convert=('/databases/ee/oradata/DB19',
'/databases/ee/oradata/CDB1913/emp')
#
# Patch relocate DB
#
patch5.sid=cdb19
patch5.source_home=/databases/ee/product/1911/dbhome_1
patch5.target_home=/databases/ee/product/1913/dbhome_2
patch5.target_cdb=cdb1913
patch5.pdbs=cars
patch5.target_pdb_copy_option.cars=file_name_convert=('/databases/ee/oradata/CDB19/ca
rs', '/databases/ee/oradata/CDB1913/cars')
patch5.source_dblink.cars=db19_link
#
# Oracle RAC
#
patch6.sid=rac1
patch6.source_home=/databases/ee/product/1911/dbhome_1
patch6.target_home=/databases/ee/product/1913/dbhome_2
#
# Distributed Oracle RAC with proactive fixups
#
patch7.sid=rac2
patch7.source_home=/databases/ee/product/1911/dbhome_1
patch7.target_home=/databases/ee/product/1913/dbhome_2
patch7.tune_setting=proactive_fixups=true,distributed_upgrade=true
```

例3-6 AutoUpgradeパッチ適用のサマリー・ログ・ファイル

このパッチ適用サマリー・レポート・ファイルでは、AutoUpgradeによってCDBおよびPDBにパッチがどのように適用されたかを確認できます。

```
*******************************************************************************
                 Datapatch Apply Summary Report for CDB$ROOT
        Return code        = 0 SUCCESS
        Failure reason     = null
        Total time         = 161.721805095673
        Install patches    = 1
        Database Open      = SUCCESS
        Invocation Log     =
/databases/cfgtoollogs/sqlpatch/sqlpatch_17781_2022_05_18_10_57_58/sqlpatch_invocatio
n.log
        Bootstrap Required = 1
        Bootstrap Status   = SUCCESS
        Bootstrap Log      =
/databases/cfgtoollogs/sqlpatch/sqlpatch_17781_2022_05_18_10_57_58/bootstrap1_CDB19X_
CDBROOT.log
        Total patches      = 1
        Patch Key          = 33192793-24462514
        Mode               = apply
        Status             = SUCCESS
        Patch Log File     =
/databases/cfgtoollogs/sqlpatch/33192793/24462514/33192793_apply_CDB19X_CDBROOT_2022M
ay18_10_58_06.log
        RU Log File        =
/databases/cfgtoollogs/sqlpatch/33192793/24462514/33192793_ru_apply_CDB19X_CDBROOT_20
22May18_10_58_05.log
        RU Errors          = N/A
*******************************************************************************
                 Datapatch Apply Summary Report for PDBX
        Return code        = 0 SUCCESS
        Failure reason     = null
        Total time         = 123.969398021698
        Install patches    = 1
        Database Open      = SUCCESS
        Invocation Log     =
/databases/cfgtoollogs/sqlpatch/sqlpatch_18416_2022_05_18_11_01_40/sqlpatch_invocatio
n.log
        Bootstrap Required = 1
        Bootstrap Status   = SUCCESS
        Bootstrap Log      =
/databases/cfgtoollogs/sqlpatch/sqlpatch_18416_2022_05_18_11_01_40/bootstrap1_CDB19X_
PDBX.log
        Total patches      = 1
        Patch Key          = 33192793-24462514
        Mode               = apply
        Status             = SUCCESS
        Patch Log File     =
/databases/cfgtoollogs/sqlpatch/33192793/24462514/33192793_apply_CDB19X_PDBX_2022May1
8_11_01_55.log
        RU Log File        =
/databases/cfgtoollogs/sqlpatch/33192793/24462514/33192793_ru_apply_CDB19X_PDBX_2022M
ay18_11_01_55.log
        RU Errors          = N/A
*******************************************************************************
                 Datapatch Apply Summary Report for PDB$SEED
        Return code        = 0 SUCCESS
        Failure reason     = null
        Total time         = 124.234117984772
        Install patches    = 1
        Database Open      = SUCCESS
        Invocation Log     =
/databases/cfgtoollogs/sqlpatch/sqlpatch_18406_2022_05_18_11_01_40/sqlpatch_invocatio
n.log
```

```
        Bootstrap Required = 1
        Bootstrap Status   = SUCCESS
        Bootstrap Log      =
/databases/cfgtoollogs/sqlpatch/sqlpatch_18406_2022_05_18_11_01_40/bootstrap1_CDB19X_
PDBSEED.log
        Total patches      = 1
        Patch Key          = 33192793-24462514
        Mode               = apply
        Status             = SUCCESS
        Patch Log File     =
/databases/cfgtoollogs/sqlpatch/33192793/24462514/33192793_apply_CDB19X_PDBSEED_2022M
ay18_11_01_55.log
        RU Log File        =
/databases/cfgtoollogs/sqlpatch/33192793/24462514/33192793_ru_apply_CDB19X_PDBSEED_20
22May18_11_01_55.log
        RU Errors          = N/A
```

親トピック: AutoUpgradeパッチ適用

AutoUpgradeとOracle Data Guard

My Oracle SupportからダウンロードするAutoUpgradeユーティリティにより、Oracle Data Guard用に構成されたプライマリおよびセカンダリのデータベースのアップグレード・プロセスを簡略化できます。

- AutoUpgradeによるOracle Data Guardのアップグレードの実行方法
  AutoUpgradeは、Oracle Data Guard構成を検出し、プライマリ・データベース用に構成されたスタンバイ・データベースへのログの送信を遅延させることができます。
- Oracle Data GuardのアップグレードのためにAutoUpgradeで実行されるステップ
  AutoUpgradeで実行されるステップは、スタンバイ・データベースが手動で管理されているか、Data Guard Brokerを使用して管理されているかによって異なります。
- プライマリ・データベースのアップグレード後のステップ
  Oracle Data Guardのアップグレードでは、プライマリ・データベースをアップグレードした後に、次の手順を実行する必要があります。

親トピック: AutoUpgradeおよびOracle Databaseの構成オプション

AutoUpgradeによるOracle Data Guardのアップグレードの実行方法

AutoUpgradeは、Oracle Data Guard構成を検出し、プライマリ・データベース用に構成されたスタンバイ・データベースへのログの送信を遅延させることができます。

AutoUpgradeでは、Oracle Data Guardデプロイメントの有無と、そのデプロイメントが手動で構成されているのか、Data Guard Brokerを使用してOracle Data Guard構成を管理およびモニターしているのかが自動的に検出されます。

構成ファイルでパラメータ`defer_standby_log_shipping`をno (デフォルト)に設定すると、AutoUpgradeでは、Oracle Data Guardが手動で構成されている場合とData Guard Brokerを使用してOracle Data Guardが構成されている場合の両方で、構成済のスタンバイ・データベースへのログ送信を遅延させることができます。

データベースとOracle Data GuardのAutoUpgradeによるアップグレード前の準備

アップグレードを開始する前に、プライマリ・データベースのアップグレード中に障害が発生した場合またはプライマリ・データベースをソースOracleホームに戻す必要がある場合に備えて、スタンバイ・データベースが保護されており、リカバリ可能であることを確認します。

親トピック: AutoUpgradeとOracle Data Guard

Oracle Data Guardのアップグレードのために AutoUpgradeで実行されるステップ

AutoUpgradeで実行されるステップは、スタンバイ・データベースが手動で管理されているか、Data Guard Brokerを使用して管理されているかによって異なります。

Oracle Data Guardが手動またはData Guard Brokerを使用して管理されるOracle Data Guardの以前のリリース(**ソース**)のデータベースの場合、スタンバイ・データベースへのログ送信を管理するために、AutoUpgrade構成ファイルで `defer_standby_log_shipping=yes` (デフォルトはno)を設定できます。ただし、AutoUpgradeで実行される特定のアクションは、スタンバイ・データベースの管理方法によって異なります。

> ノート:
>
> 手動または Data Guard Broker を使用して管理されるスタンバイ・データベースの場合、アップグレードの完了後、プライマリ・データベースでのアップグレードが正常に終了した後に各スタンバイ・アーカイブ・ログの宛先で `ENABLE DATABASE database-name;`を実行し、REDO ログ適用を使用してスタンバイ・データベースをアップグレードするために必要なステップをすべて実行する必要があります。

手動管理のOracle Data Guardスタンバイ・データベース

直接アップグレードがサポートされているOracle Data Guardスタンバイ・データベースの場合、AutoUpgradeは、手動管理スタンバイ・データベースとData Guard Broker管理スタンバイ・データベースの両方のアップグレード・プロセスを開始する前に、すべての`VALID`および`ENABLED`スタンバイ・アーカイブ・ログの宛先を`DEFER`モードにします。

Data Guard Broker管理のOracle Data Guardスタンバイ・データベース

Data Guard Brokerを使用して管理されるOracle Data Guardスタンバイ・データベースとの直接アップグレードがサポートされているOracle Databaseリリースの場合、AutoUpgradeは次のアクションを完了します。

- プライマリ・データベースの状態が、Data Guard Brokerを使用して構成されたすべてのスタンバイ・データベースに対して`TRANSPORT-OFF`に設定されます。
- Data Guard Brokerのファイルが、ソースOracleホームからターゲットOracleホームにコピーされます。

> ノート:
>
> Data Guard Broker ファイルが Oracle ホームの外部にある場合、ファイルは見つからず、コピーされません。

親トピック: AutoUpgradeとOracle Data Guard

プライマリ・データベースのアップグレード後のステップ

Oracle Data Guardのアップグレードでは、プライマリ・データベースをアップグレードした後に、次の手順を実行する必要があります。

- プライマリ・データベースでREDO転送が有効になっていることを確認し、アップグレードがスタンバイ・データベースに適用されるようにします。
- アーカイブが適用され、ギャップが最小限であることを確認します。適用ラグおよびトランスポート・ラグは5分以内にすることをお薦めします。

例3-7 REDO転送サービスのステータスの確認

プライマリ・データベースのREDO転送サービスのステータスを確認するには、Data Guard Brokerコマンドライン・インタフェース

(DGMGRL)で監視可能なプロパティLogXptStatusを使用します。たとえば:

```
DGMGRL> SHOW DATABASE 'sales1' 'LogXptStatus' ;
```

例3-8 適用ラグおよびトランスポート・ラグの確認

アーカイブが適用されていることを確認し、適用ラグおよびトランスポート・ラグが5分以内であることを確認するには、プライマリ・データベースにログインし、次のようなSQL問合せを発行します。

```
[oracle]$ sqlplus / as sysdba
SYS@sales1>
SET LINESIZE 200
COL VALUE FOR A30
SELECT NAME,VALUE,TIME_COMPUTED,DATUM_TIME FROM V$DATAGUARD_STATS WHERE NAME LIKE
'%lag';
```

結果は次のようになります。

```
NAME VALUE TIME_COMPUTED DATUM_TIME
-------------- -------------- --------------------- --------------------
transport lag +00 00:00:00 timestamp timestamp
apply lag +00 00:01:07 timestamp timestamp
```

関連項目

- 使用例13: Data Guard構成の監視

親トピック: AutoUpgradeとOracle Data Guard

高速デプロイ・オプションを使用してAutoUpgradeを実行する方法

停止時間を最小限に抑えるには、高速デプロイ・オプションを使用してAutoUpgradeを実行することでデータベースをアップグレードできます。

AutoUpgrade 21.2以降、アプリケーションで停止時間を最小限に抑える必要がある場合、高速デプロイを使用することで停止時間を短くしてアップグレードできるようになりました。高速デプロイ・オプションを使用すると、データベースがオンライン中でも事前チェックおよび事前修正を実行できます。ソース・データベースで修正を実行した後、デプロイ・モードでAutoUpgradeを実行し、事前チェックおよび事前修正のステージをスキップできるため、実際のアップグレードのみに停止時間が必要となります。

ノート:

標準の分析モードとデプロイ・モードを使用して AutoUpgrade を実行することをお薦めします。高速デプロイ方法を使用する場合は、アップグレード前の分析モードで AutoUpgrade を実行した後、かつアップグレード・モードで AutoUpgrade を実行する前の期間中に新しい問題が発生する可能性があるというリスクがわずかにあることに注意してください。これが問題を引き起こす可能性があります。そのリスクを評価し、それに応じて予防措置を講じてください。

1. AutoUpgrade構成ファイルを作成し、ソース・システムおよびターゲット・システムに関する情報とアップグレード・プリファレンスを指定します。次の各ステップでは、そのファイル名はmyconfig.cfgです。
2. 分析モードを使用してデータベースを分析します。

   ```
   - java -jar autoupgrade.jar -config myconfig.cfg -mode analyze
   ```

3. 修正モードを使用してアップグレード前の修正を実行します。

   ```
   - java -jar autoupgrade.jar -config myconfig.cfg -mode fixups
   ```

4. アップグレード・モードを使用してデータベースをアップグレードします。

```
– java -jar autoupgrade.jar -config myconfig.cfg -mode upgrade
```

このコマンドを実行すると、アップグレードされるデータベースがターゲットのOracleホームでUPGRADEモードでオープンされるため、データベースが停止することがあります。

親トピック: AutoUpgradeおよびOracle Databaseの構成オプション

暗号化されたPDBの切断/接続アップグレードの実行方法

AutoUpgradeユーティリティを使用して、暗号化されたPDBの切断/接続アップグレードを実行する方法について説明します。

注意:

AutoUpgrade が `global.keystore` を使用して作成するディレクトリを指定する場合は、ソフトウェア・キーストアが格納されることに注意してください。TDE キーストア・ファイルとともに使用する場合と同じセキュリティのベスト・プラクティスを使用して保護する必要があります。

暗号化されたPDBのアップグレードを実行するには、AutoUpgradeでTDEキーストアのパスワードを独自のセキュア・キーストアにロードする必要があります。パスワードをロードするには、構成ファイルでAutoUpgradeグローバル構成ファイル・パラメータkeystoreを設定し、コマンドライン・パラメータload_passwordを使用してAutoUpgradeを実行します。このパラメータは、-configパラメータとともに使用する必要があります。引数はありません。かわりに、キーストアに必要な情報を提供できる特定のコマンドを使用して対話型プロンプトを起動します。AutoUpgradeは、アップグレード中に安全にロードされたパスワードを格納し、必要に応じてこれらのパスワードを使用します。

ノート:

データベースが Oracle Secure External Password Store (SEPS)を使用しており、TDE キーストア・パスワードが要求されると、AutoUpgrade は `IDENTIFIED BY EXTERNAL STORE` 句を使用するため、パスワードを AutoUpgrade パスワード・キーストアにロードする必要はありません。ただし、すべてのデータベースがセキュアな外部パスワード・ストアを使用して構成されている場合は、引き続き構成ファイルで `global.keystore` を定義する必要があります。

AutoUpgrade キーストアは、Oracle Database が使用する他のキーストアと似ています。自動ログイン・キーストアとして作成することもできます。たとえば、外部キーストアが `ewallet.p12` の場合、AutoUpgrade は、`ewallet.p12` キーストアを開くために使用される自動ログイン・キーストア `cwallet.sso` を作成します。

暗号化されたPDBのアップグレードを開始する前に、次を満たす必要があります。

- AutoUpgradeはバージョン22.2以降である必要があります。常に最新バージョンのAutoUpgradeをダウンロードして実行することをお薦めします。
- 外部TDEキーストアを使用する必要があります
- ソースおよびターゲットのCDBの外部キーストア・パスワードを使用可能にしておく必要があります。

例3-9 AutoUpgradeを使用した切断/接続アップグレードによる暗号化されたPDBのアップグレード

次の例では、Transparent Data Encryption (TDE)を使用しているOracle Database 12cリリース2 (12.2) PDBが、

アップグレード中にTDEキーのセキュア・ストアを指定するためにAutoUpgrade構成ファイル・キーストア・パラメータとともにAutoUpgrade load_passwordコマンド・オプションを使用して、Oracle Database 19cにアップグレードされます。

1. 最新バージョンのAutoUpgradeがあることを確認します。

```
$ java -jar autoupgrade.jar -version
```

AutoUpgradeバージョンが以前のリリースの場合は、My Oracle Support AutoUpgradeツール(ドキュメントID 2485457.1)から最新バージョンのAutoUpgradeをダウンロードします

2. AutoUpgrade構成ファイルを作成します。この例では、パス/u01/app/oracle/admin/autoupgrade/keystoreの、Oracleベース・ディレクトリ下のadminフォルダに指定されているキーストアへのパスを使用して、構成ファイルPDB1.cfgが作成されます。ソースCDBはCDB1で、ターゲットCDBはCDB2です。

```
global.autoupg_log_dir=/u01/app/oracle/cfgtoollogs/autoupgrade
global.keystore=/u01/app/oracle/admin/autoupgrade/keystore
upg1.log_dir=/u01/app/oracle/cfgtoollogs/autoupgrade/DB12
upg1.source_home=/u01/app/oracle/product/12.2.0/dbhome_1
upg1.target_home=/u01/app/oracle/product/19.1.0/dbhome_1
upg1.sid=CDB1
upg.pdbs=PDB1
upg1.target_cdb=CDB2
```

ノート:

キーストアの場所を指定せず、以前に AutoUpgrade キーストアが作成されていない場合は、AutoUpgrade によって AutoUpgrade キーストアが作成されます。

3. 分析モードでAutoUpgradeを実行します。

```
$ java -jar autoupgrade.jar -config PDB1.cfg -mode analyze
```

サマリー・レポートは、アップグレードを開始する前にTDEキーストア・パスワードが必要であることを示しています(TDE_PASSWORDS_REQURED)。

```
[Stage Name]    PRECHECKS
[Status]        FAILURE
[Start Time]    2022-03-29 07:58:52
[Duration]
[Log Directory] /u01/app/oracle/cfgtoollogs/autoupgrade/PDB1/CDB1/100/prechecks
[Detail]
/u01/app/oracle/cfgtoollogs/autoupgrade/PDB1/CDB1/100/prechecks/cdb1_preupgrade
.log
                Check failed for PDB1, manual intervention needed for the below
checks
                [TDE_PASSWORDS_REQUIRED]
```

4. TDEキーストア・パスワードをAutoUpgradeキーストアに追加します。

```
$ java -jar autoupgrade.jar -config PDB1.cfg -load_password
```

-load_passwordコマンド・オプションを指定してAutoUpgradeを初めて実行すると、TDEパスワードを安全に格納できるAutoUpgradeキーストアのパスワードを作成するよう求められます。

```
Starting AutoUpgrade Password Loader - Type help for available options
```

```
Creating new keystore - Password required
Enter password:
Enter password again:
Keystore was successfully created
```

SEPSキーストアを使用しない場合、AutoUpgradeにより、構成ファイルのsource_homeで指定されたデータベースのTDEキーストア・パスワードをAutoUpgrade独自のキーストアに追加するよう求められますが、この場合、そのデータベースがTDEキーストア・パスワードを必要とします。これは、構成ファイルの例で指定したものです。

次の例では、ソースおよびターゲットのTDEキーストア・パスワードがロードされます。

```
TDE> ADD CDB1
Enter your secret/Password:
Re-enter your secret/Password:
TDE> ADD CDB2
Enter your secret/Password:
Re-enter your secret/Password:
```

パスワードがロードされたことを確認します。

```
TDE> LIST
+---------------+-----------------+-----------------+--------------+-----------
-----------+
|ORACLE_SID     |Action Required  |TDE Password     |SEPS Status   |Active
Wallet Type    |
+---------------+-----------------+-----------------+--------------+-----------
-----------+
| CDB1          |                 | Verified        | Inactive     | Auto-login
|
| CDB2          |                 | Verified        | Inactive     | Any
|
+---------------+-----------------+-----------------+--------------+-----------
-----------+
```

Action Required列が空の場合、アップグレードにTDEパスワードを使用できます。

ソースCDBまたはターゲットCDBでSEPSキーストアを使用する場合、AutoUpgradeはTDEパスワードのソースとしてSEPSキーストアを自動的に検出します。AutoUpgradeはIDENTIFIED BY EXTERNAL STORE句を使用するので、TDEキーストア・パスワードをAutoUpgradeキーストアにロードする必要がありません。ここでも、TDE LISTコマンドで空のAction Required列が表示されます。

```
TDE> LIST
+--------------+-----------------+-------------------------+--------------+---
------------------+
|ORACLE_SID    |Action Required    | TDE Password           |SEPS Status
|Active Wallet Type   |
+--------------+-------------------+------------------------+--------------+---
------------------+
| CDB1         |                   |No password loaded      | Verified     |
Any                 |
| CDB2         |                   |No password loaded      | Verified     |
Any                 |
+--------------+-------------------+------------------------+--------------+---
------------------+
```

同じ構成ファイルで両方のオプションを使用できます。たとえば、ソース非CDBデータベースにSEPSキーストアを使用せずに、ターゲットCDBにSEPSキーストアを使用する場合、必要なのは、ソース非CDBのパスワードをロードすることのみです。

5. TDEパスワードをAutoUpgradeキーストアに保存します。(オプション): この例では、キーストアを自動ログイン・キーストアに変換します。

```
TDE> save
Convert the keystore to auto-login [YES|NO] ? YES
TDE> exit
```

6. PDBを再度分析します。

```
$ java -jar autoupgrade.jar -config PDB1.cfg -mode analyze
```

レポートをレビューし、すべての問題が解決されたことを確認します。

7. アップグレードおよび変換を開始します。

```
$ java -jar autoupgrade.jar -config PDB1.cfg -mode deploy
```

8. AutoUpgradeは、アップグレードを完了するために必要に応じてTDEパスワードを使用して、ターゲットOracle DatabaseでPDB1のアップグレードを開始します。この例では、AutoUpgrade自動ログイン・キーストアを作成し、独自のセキュア・キーストアを使用してTDEパスワードにアクセスするようにAutoUpgradeを構成したため、アップグレード中にTDEパスワードの指定を求められることはありません。

関連項目

- パスワード資格証明用の安全性の高い外部パスワード・ストアの管理
- Upgrading an Encrypted PDB
- AutoUpgradeツール(ドキュメントID 2485457.1)

親トピック: AutoUpgradeおよびOracle Databaseの構成オプション

暗号化されたPDBの非CDBからPDBへの変換を実行する方法

AutoUpgrade 22.1以降の更新では、AutoUpgradeにより、Transparent Data Encryption (TDE)を使用するOracle Databaseのアップグレードおよび変換が簡略化されます。

アップグレード・プロセスを簡略化しながらアップグレード中に機密情報を保護するには、AutoUpgradeグローバル構成ファイル・パラメータ`keystore`およびAutoUpgradeコマンドライン・パラメータ`load_password`を使用して、TDEパスワードをAutoUpgradeキーストアに安全にロードし、必要に応じてそのパスワードを使用します。AutoUpgradeキーストアを使用できるようにするには、`global.keystore`を使用して、AutoUpgrade構成ファイルで外部パスワード・ストアの場所を指定します。

ノート:

AutoUpgrade キーストア・パスを指定する場合、そのパスは、AutoUpgrade で指定した他のファイル・パスとは異なる必要があるので、キーストアはどのログ・ファイルの場所にも存在しません。AutoUpgrade キーストア・パス・ディレクトリが存在しない場合、AutoUpgrade によって自動的に作成されます。

データベースが Oracle Secure External Password Store (SEPS)を使用しており、TDE キーストア・パスワードが要求されると、AutoUpgrade は `IDENTIFIED BY EXTERNAL STORE` 句を使用するため、パスワードを AutoUpgrade パスワード・キーストアにロードする必要はありません。ただし、すべてのデータベースがセキュアな外部パスワード・ストアを使用して構成されている場合は、引き続き構成ファイルで `global.keystore` を定義する必要があります。

例3-10 TDEを使用したデータベースのアップグレードおよび非CDBからPDBへの変換

次の例では、Transparent Data Encryption (TDE)を使用しているOracle Database 12cリリース2 (12.2)非CDB

データベースが、PDB変換およびアップグレード中にTDEキーのセキュア・ストアを指定するためにAutoUpgrade構成ファイル・キーストア・パラメータとともにAutoUpgrade `load_password`コマンド・オプションを使用して、Oracle Database 19cにアップグレードされ、PDBに変換されます。

1. 最新バージョンのAutoUpgradeがあることを確認します。

```
$ java -jar autoupgrade.jar -version
```

AutoUpgradeバージョンが以前のリリースの場合は、My Oracle Support AutoUpgradeツール(ドキュメントID 2485457.1)から最新バージョンのAutoUpgradeをダウンロードします

2. AutoUpgrade構成ファイルを作成します。この例では、キーストアへのパスが、パス `/u01/app/oracle/admin/autoupgrade/keystore`の、Oracleベース・ディレクトリ下のadminフォルダに指定されています。

```
global.autoupg_log_dir=/u01/app/oracle/cfgtoollogs/autoupgrade
global.keystore=/u01/app/oracle/admin/autoupgrade/keystore
upg1.log_dir=/u01/app/oracle/cfgtoollogs/autoupgrade/DB12
upg1.source_home=/u01/app/oracle/product/12.2.0
upg1.target_home=/u01/app/oracle/product/19.1.0
upg1.sid=DB12
upg1.target_cdb=CDB2
```

3. 分析モードでAutoUpgradeを実行します。

```
$ java -jar autoupgrade.jar -config DB12.cfg -mode analyze
```

サマリー・レポートは、アップグレードを開始する前に追加のアップグレード前タスクを完了する必要がないことを示しています。

```
[Stage Name]    PRECHECKS
[Status]        SUCCESS
[Start Time]    2022-03-30 10:28:38
[Duration]
[Log Directory] /u01/app/oracle/cfgtoollogs/autoupgrade/DB12/DB12/100/prechecks
[Detail]
/u01/app/oracle/cfgtoollogs/autoupgrade/DB12/DB12/100/prechecks/db12_preupgrade
.log
                                  Check passed and no manual intervention needed
```

`TDE_PASSWORDS_REQUIRED`チェックが失敗した場合は、TDEパスワードをロードします。

```
$ java -jar autoupgrade.jar -config DB12.cfg -load_password
```

`-load_password`コマンド・オプションを指定してAutoUpgradeを初めて実行すると、AutoUpgradeでアップグレードの実行にTDEパスワードが必要な場合は、TDEパスワードを安全に格納できるAutoUpgradeキーストアのパスワードを作成するよう求められます。

```
Starting AutoUpgrade Password Loader - Type help for available options
Creating new keystore - Password required
Enter password:
Enter password again:
Keystore was successfully created
```

SEPSキーストアを使用しない場合、AutoUpgradeにより、構成ファイルの`source_home`で指定されたデータベースのTDEキーストア・パスワードをAutoUpgrade独自のキーストアに追加するよう求められますが、この場合、そのデータベースがTDEキーストア・パスワードを必要とします。これは、構成ファイルの例で指定したものです。

次の例では、ソースおよびターゲットのTDEキーストア・パスワードがロードされます。

```
TDE> ADD DB12
Enter your secret/Password:
Re-enter your secret/Password:
TDE> ADD CDB2
Enter your secret/Password:
Re-enter your secret/Password:
```

4. TDE構成を確認するには、-load_passwordコマンド・パラメータを使用してAutoUpgradeを再度実行できます。今回は、パスワードがすでにロードされているため、TDEコンソール・プロンプトが表示されます。TDEコンソールのlistコマンドを実行して、TDE構成を確認します。

```
$ java -jar autoupgrade.jar -config DB12.cfg -load_password

TDE> list
+----------+---------------+------------------+-----------+------------------+
|ORACLE_SID|Action Required|   TDE Password   |SEPS Status|Active Wallet Type|
+----------+---------------+------------------+-----------+------------------+
|      CDB2|               |No password loaded|   Verified|               Any|
|      DB12|               |No password loaded|    Unknown|        Auto-login|
+----------+---------------+------------------+-----------+------------------+
```

listコマンドの表出力では、Action Required列は空です。この結果から、TDEキーストア・パスワードが指定されたことが確認されます。SEPS Status列で、AutoUpgradeはターゲットOracle Database CDB2でのパスワード・ストレージ・アクセスをチェックしたことをレポートし、パスワードが機能していることを確認しています。パスワードをチェックするTDEコンソール機能はOracle Database 19cで追加された機能であるため、AutoUpgradeはOracle Database 12cリリース2 (12.2)でTDEのパスワード・チェックを確認できないので、Unknownという結果が想定されます。

ソースCDBまたはターゲットCDBでSEPSキーストアを使用する場合、AutoUpgradeはTDEパスワードのソースとしてSEPSキーストアを自動的に検出します。AutoUpgradeはIDENTIFIED BY EXTERNAL STORE句を使用するので、TDEキーストア・パスワードをAutoUpgradeキーストアにロードする必要がありません。ここでも、TDE LISTコマンドで空のAction Required列が表示されます。

同じ構成ファイルで両方のオプションを使用できます。たとえば、ソース非CDBデータベースにSEPSキーストアを使用せずに、ターゲットCDBにSEPSキーストアを使用する場合、必要なのは、ソース非CDBのパスワードをロードすることのみです。

5. アップグレードおよび変換を開始します。

```
$ java -jar autoupgrade.jar -config DB12.cfg -mode deploy
```

6. AutoUpgradeは、アップグレードを完了するために必要に応じてTDEパスワードを使用して、ソースの非CDB Oracle DatabaseのアップグレードおよびターゲットOracle DatabaseのPDBへの変換を開始します。

注意:

他の Oracle Database キーストアと同様に、AutoUpgrade キーストア・ファイルを保護します。

- 制限されたファイル・システム権限の適用
- アクセスの監査

- キーストアのバックアップ

関連項目

- パスワード資格証明用の安全性の高い外部パスワード・ストアの管理
- AutoUpgrade and Secure External Password Store Enables Complete Automation
- AutoUpgradeツール(ドキュメントID 2485457.1)

親トピック: AutoUpgradeおよびOracle Databaseの構成オプション

AutoUpgrade構成ファイルの例

次の例を使用して、アップグレード時に様々な構成アクションを実行するために独自のAutoUpgrade構成ファイルを変更する方法を理解します。

- AutoUpgradeの構成ファイルの作成
  AutoUpgradeを使用してアップグレードを完了するには、まず、新しいリリースのOracleホームからAutoUpgradeで構成ファイルを作成します。
- AutoUpgradeを使用したアップグレード時のTDEウォレット・ストアの場所の更新
  AutoUpgrade構成ファイルのパラメータを使用して、アップグレード時に透過的データ暗号化(TDE)ウォレット・ストアを更新する方法について説明します。
- 2つのデータベース・エントリを持つAutoUpgrade構成ファイル
  構成ファイルで複数のデータベースのアップグレード・オプションを指定する方法について説明します。
- AutoUpgrade構成ファイル・エントリを使用したアップグレードの標準化
  AutoUpgradeを使用してアップグレード時にデータベース構成の標準化を強制する方法について説明します。
- 一連のPDBの増分アップグレード用のAutoUpgrade構成ファイル
  ソースCDBの他のPDBに影響を与えることなく、AutoUpgradeを使用してPDBのサブセットを選択的にアップグレードする方法について説明します。
- スクリプト・ジョブまたはバッチ・ジョブでのAutoUpgradeの実行方法
  `noconsole`パラメータを使用してAutoUpgradeを呼び出し、非対話型モードで独自のスクリプトでAutoUpgradeを実行する方法について学習します。
- AutoUpgradeによる切断/接続再配置アップグレード
  切断/接続再配置機能(ホット・クローニング・アップグレードとも呼ばれる)を使用して、アップグレード前に特定の期間にリフレッシュ可能なPDBを作成する方法を確認します。
- AutoUpgrade構成ファイルを使用した修正およびチェックの無視
  データベースのチェックおよび修正ステップ全体をスキップするには、修正チェックリスト・ファイルから修正の`runfix`フラグを読み取り、その修正のフラグを`YES`から`SKIP`に設定するようにAutoUpgradeに指示します。
- AutoUpgradeを使用したカスタム・スクリプトの実行
  デプロイ・プロセスの一環としてAutoUpgradeを使用して独自のスクリプトを実行する方法について学習します。

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

AutoUpgradeの構成ファイルの作成

AutoUpgradeを使用してアップグレードを完了するには、まず、新しいリリースのOracleホームからAutoUpgradeで構成ファイルを作成します。

次の例では、パラメータ`sample_config_file`を使用してAutoUpgradeユーティリティが実行されます。このパラメータは、AutoUpgradeを実行しているユーザーのホームで構成ファイルを生成します。このファイルを編集すると、アップグレードの環境パスと設定およびアップグレード・プリファレンスを提供できます。構成ファイル(`config`)を生成するには、`sample_config_file`パラメータを使用して新しいリリースのOracle DatabaseホームからAutoUpgradeを実行し、出力ファイル名を指定します。

ノート:

AutoUpgrade は定期的に更新されます。その他の例、および新規コマンドライン・パラメータやオプション、新規ま

たは拡張構成ファイルのパラメータなど、最新の AutoUpgrade リリースの詳細は、アップグレードするリリースの『Oracle Database アップグレード・ガイド』を参照してください。My Oracle Support ノート「AutoUpgrade ツール(ドキュメント ID 2485457.1)」も参照してください。このノートには、最新の AutoUpgrade の更新に関する情報が含まれています。

この例では、ユーザーoracleが以前のリリースのOracleホームの場所に移動します。この例ではOracle Database 19cです。

```
cd /u01/app/oracle/product/19.0.0/
```

次に、OracleユーザーがOracle Database 23cのOracleホームからAutoUpgradeを起動し、ユーザー・ホーム・ディレクトリ/home/oracleに構成ファイルを作成します。

```
java -jar /u01/app/oracle/product/23/rdbms/admin/autoupgrade.jar -create_sample_file
config
Created sample configuration file /home/oracle/sample_config.cfg
```

構成ファイルを作成したら、優先テキスト・エディタでファイルを開き、環境に必要なパラメータ設定を変更します。

```
cd /
vi sample_config.cfg
```

関連項目

- Oracle Databaseドキュメント
- AutoUpgradeツール(ドキュメントID 2485457.1)

親トピック: AutoUpgrade構成ファイルの例

AutoUpgradeを使用したアップグレード時のTDEウォレット・ストアの場所の更新

AutoUpgrade構成ファイルのパラメータを使用して、アップグレード時に透過的データ暗号化(TDE)ウォレット・ストアを更新する方法について説明します。

以前のリリースでは、Oracle WalletをTDEとともに使用した場合、非推奨のsqlnet.oraのパラメータSQLNET.ENCRYPTION_WALLET_LOCATIONを使用して、既存のキーストア・ディレクトリの場所を指定していました。Oracle Database 19c以降のリリースでは、データベース初期化パラメータ・ファイル(PFILE)のWALLET_ROOTシステム・パラメータを使用してキーストアの場所を指定する必要があります。実行する必要がある操作は、ソースのOracle Databaseリリースの構成方法によって異なります。

- ソースのOracle DatabaseリリースでWALLET_ROOTがすでに設定されている場合、AutoUpgradeによって生成されるパラメータ・ファイルでは、アップグレード時にソース・データベースからWALLET_ROOTシステム・パラメータを自動的に取得し、そのパラメータをターゲット・データベースのパラメータ・ファイルで使用します。
- ソースのOracle DatabaseリリースでWALLET_ROOTが設定されていない場合は、アップグレード時にAutoUpgradeを使用してそのタスクを完了できます。

1. 使用するディレクトリのWALLET_ROOT初期化パラメータ値を使用してオペレーティング・システムにテキスト・ファイルを作成し、必要なキーストアのタイプを作成するための構成オプションをTDE_CONFIGURATION動的初期化パラメータに指定します。たとえば、透過的データ暗号化ソフトウェア・キーストアにFILEを使用するようにTDE_CONFIGURATIONを構成すると、Oracle DatabaseによってWALLET_ROOT/tde (小文字)にソフトウェア・キーストアが作成されます。

2. AutoUpgrade構成ファイルで、AutoUpgrade構成ファイルのパラメータadd_during_upgrade_pfileおよびadd_after_upgrade_pfileを使用して、オペレーティング・システムのそのファイルを参照し、アップグレード時にWALLET_ROOTおよびTDE_CONFIGURATIONを設定します。

たとえば、WALLET_ROOTでパス/u01/app/oracle/admin/hr/walletを使用し、透過的データ暗号化でソフトウェア・キーストアをWALLET_ROOT/tdeに格納する場合、次の行が含まれるtde-upgradeというテキスト・ファイルを作成できます。

```
WALLET_ROOT=/u01/app/oracle/admin/hr/wallet
tde_configuration="KEYSTORE_CONFIGURATION=FILE"
```

その後、AutoUpgradeでこれらのパラメータをAutoUpgrade構成ファイルに設定するように指定できます。たとえば、アップグレード中およびアップグレード後に透過的データ暗号化キーストアを設定するには、AutoUpgrade操作の一環として、ローカル構成ファイルに次の行を追加してそのテキスト・ファイルを呼び出します。

```
#
#       Example local pfile configuration entries
upg1.add_after_upgrade_pfile=/usr/home/oracle/tde-upgrade
upg1.add_during_upgrade_pfile=/usr/home/oracle/tde-upgrade
```

関連項目

- 透過的データ暗号化の構成の仕組み

親トピック: AutoUpgrade構成ファイルの例

2つのデータベース・エントリを持つAutoUpgrade構成ファイル

構成ファイルで複数のデータベースのアップグレード・オプションを指定する方法について説明します。

この例は、2つのデータベースのアップグレードを指定するAutoUpgrade構成ファイルです。構成ファイルは、AutoUpgradeが次のアクションを実行することを指定します。

データベース1

- Oracle Database 12cリリース2 (12.2) CDBのインプレース・データベース・アップグレード。ソースおよびターゲットのOracleホームは、同じシステム識別子(sid=HR1)を持つ同じサーバー・ハードウェア上の同じOracleベース・ディレクトリ(Oracle Databaseインストール所有者oracle (/u01/app/oracle/)のデータベース・ホーム・ディレクトリ)を使用します。
- アップグレード中に、CDBのすべてのPDBがアップグレードされます(pdbs=*)
- アップグレードはただちに開始されます(start_time=now)。
- データベース・アップグレード・ログは、パス/database/logs/hr (log_dir=/database/logs/hr)に送信されます。
- タイムゾーンのアップグレードは、すべてのコンテナで実行されます(timezone_upg=yes)

データベース2

- Oracle Database 18c CDBのインプレース・データベース・アップグレード。ソースおよびターゲットのOracleホームは、同じOracleベース・ディレクトリ(同じシステム識別子(sid=SALES1)を持つ同じサーバー・ハードウェア上のOracle Databaseインストール所有者oracle (/u01/app/oracle/)のデータベース・ホーム・ディレクトリ)を使用します。
- アップグレードはただちに開始されます(start_time=now)。
- データベース・アップグレード・ログは、パス/database/logs/sales (log_dir=/database/logs/sales)に送信されます。

- タイムゾーンのアップグレードは、コンテナ(timezone_upg=no)では実行されません。

両方のデータベースの場合:

- パラメータupgrade_nodeでは、ホスト名に割り当てられた別名ではなく、実際のシステム・ホスト名(nodename-1)を指定します。(キーワードlocalhostを使用して、現在のシステムを参照することもできます。)
- グローバルAutoUpgradeログ・ファイル(ジョブ・マネージャ・ログとも呼ばれる)は、パス/database/jobmgr(autoupg_log_dir=/database/jobmgr)の下に配置されます。

```
#
# Global logging directory pertains to all jobs
#
global.autoupg_log_dir=/database/jobmgr
#
# Database 1
#
upg1.source_home=/u01/app/oracle/product/12.2.0.2/dbhome_1
upg1.target_home=/u01/app/oracle/product/19.0.0/dbhome_1
upg1.sid=HR1
upg1.start_time=now
upg1.pdbs=*
upg1.log_dir=/database/logs/hr
upg1.upgrade_node=nodename1
upg1.run_utlrp=yes
upg1.timezone_upg=yes
upg1.target_version=21
#
# Database 2
#
upg2.source_home=/u01/app/oracle/product/18.0.0/dbhome_1
upg2.target_home=/u01/app/oracle/product/19.0.0/dbhome_1
upg2.sid=SALES1
upg2.start_time=now
upg2.log_dir=/database/logs/sales
upg2.upgrade_node=nodename1
upg2.timezone_upg=no
upg2.target_version=21
```

Oracle Databaseアップグレードおよび移行のYouTubeチャネルで、Mike Dietrichによって示されている、同様のアップグレード・シナリオ*Oracle Database AutoUpgrade 19c - Upgrading 2 databases in parallel* (11:57)を参照できます。

関連項目

- Oracle Database AutoUpgrade 19c - 2つのデータベースのパラレルなアップグレード

親トピック: AutoUpgrade構成ファイルの例

AutoUpgrade構成ファイル・エントリを使用したアップグレードの標準化

AutoUpgradeを使用してアップグレード時にデータベース構成の標準化を強制する方法について説明します。

次の構成ファイルでは、AutoUpgrade構成ファイル・エントリを使用してデータベース構成を標準化する方法を確認できます。グローバルPFILEエントリは、構成ファイル内のすべてのデータベースに適用されます。ローカルPFILEエントリは、構成ファイルの特定のデータベースにのみ適用されます。これらのPFILE値の構文は、PFILE構成と同じOracleルールに従います。

```
#
#    Example global pfile configuration entries
#
global.del_during_upgrade_pfile=/database/pfiles/global_during_delinit.ora
global.add_during_upgrade_pfile=/database/pfiles/global_during_addinit.ora
```

```
global.del_after_upgrade_pfile=/database/pfiles/global_after_delinit.ora
global.add_after_upgrade_pfile=/database/pfiles/global_after_addinit.ora
#
#      Example local pfile configuration entries
#
upg1.del_during_upgrade_pfile=/database/pfiles/hr_during_delinit.ora
upg1.add_during_upgrade_pfile=/database/pfiles/hr_during_addinit.ora
upg1.del_after_upgrade_pfile=/database/pfiles/hr_after_delinit.ora
upg1.add_after_upgrade_pfile=/database/pfiles/hr_after_addinit.ora
```

AutoUpgradeプロセス時に、ファイルduring_upgrade_pfile_dbname.oraおよびafter_upgrade_pfile_dbname.oraの両方が作成されます。これらのファイルは、アップグレード時およびアップグレード後にデータベースを起動する際に使用されます。アップグレード時またはアップグレード後にシステム・パラメータを変更する場合は、両方のファイルを変更できます。

最初にグローバルのPFILEエントリが適用され、次にジョブ接頭辞upglで指定されるローカルのPFILEエントリが適用されます。これら2つの構成ファイル内で、パラメータdel_upgrade_pfileのエントリが最初に適用され、その後にパラメータadd_upgrade_pfileのエントリが続きます。これらのPFILE構成エントリのパラメータは、PFILEのターゲットに応じてPFILE during_upgrade_pfile_dbname.oraまたはPFILE after_upgrade_pfile_dbname.oraのいずれかに直接適用されます。

処置:

- del_during_upgrade_pfile during_upgrade_pfile_dbname.oraからエントリを削除します
- add_during_upgrade_pfile during_upgrade_pfile_dbname.oraにエントリを追加します。
- del_after_upgrade_pfile after_upgrade_pfile_dbname.oraからエントリを削除します
- add_after_upgrade_pfile after_upgrade_pfile_dbname.ora.にエントリを追加します。

パラメータdel_during_upgrade_pfileおよびdel_after_upgrade_pfileによって参照されるファイルには、各行に1つのデータベース・パラメータがあります。行全体がパラメータ名の一部であるため、パラメータに接頭辞を追加できません。次に例を示します。

```
#
# global.del_during_upgrade_pfile
#
processes
*.open_cursors
```

この構成設定の結果、構成ファイルに示されている各データベースのPFILEからprocessesパラメータへのすべての参照が削除されますが、open_cursorsパラメータへの参照は削除されません。接頭辞を持つopen_cursorsのインスタンスのみが削除されます。ただし、PFILEから削除されるパラメータには、接頭辞が付いたすべてのパラメータが含まれています。たとえば、*.processesおよびinstance_name.processesは、この構文を使用して両方とも削除されます。

パラメータadd_during_upgrade_pfileおよびadd_after_upgrade_pfileによって参照されるファイルには、書式parameter=valueで各行に1つのパラメータが示されます。PFILEからエントリを削除する場合は、値フィールドを空のままにできます。パラメータの先頭に*.またはinstancename.がある場合、それらの参照は、変更されたPFILEには追加されません。既存のパラメータの値を更新するには、最初にそのパラメータを削除する必要があります。その後、必要な値のパラメータを追加できます。次に例を示します。

```
#
# global.add_during_upgrade_pfile
#
processes=400
*.open_cursors=250
```

このグローバル構成ファイル・エントリにより、構成ファイルにリストされた各データベースのPFILEに次のエントリが追加されます。

```
processes=400
open_cursors=250
```

パラメータafter_upgrade_pfile_dbnameは、アップグレード後のプロセスでデータベースSPFILEを作成するために使用されます。

親トピック: AutoUpgrade構成ファイルの例

一連のPDBの増分アップグレード用のAutoUpgrade構成ファイル

ソースCDBの他のPDBに影響を与えることなく、AutoUpgradeを使用してPDBのサブセットを選択的にアップグレードする方法について説明します。

このシナリオでは、ソースCDB内の他のPDBをアップグレードせずに2つの特定のPDBをアップグレードします。増分アップグレードを実行するには、指定したPDBを以前のリリースのCDBから切断してターゲット・リリースのCDBに接続した後、ターゲットCDB上の以前のリリースのPDBをアップグレードするように、構成ファイルでAutoUpgradeに指示します。切断、接続およびアップグレードするPDBをこのように選択すると、以前のリリースのCDBでPDBの増分アップグレードを実行してダウンタイムを短縮することができます。

次の構成ファイルでは、CDB CDB122をソースCDBとして識別します。ソースCDBには10のPDB (PDB1からPDB10)がありますが、アップグレードされるのはPDB1およびPDB2のみです。アップグレード時に、PDB2というPDBの名前がDEPSALESに変更され、PDB2のデータベース・ファイル名がDEPSALESに変更されます。

```
global.autoupg_log_dir=/home/oracle/autoupg
upg1.sid=CDB122
upg1.source_home=/u03/app/oracle/product/12.2.0/dbhome_1
upg1.target_home=/u01/app/oracle/product/21.0.0/dbhome_1
upg1.target_cdb=CDB21C
upg1.pdbs=PDB2, PDB1
upg1.target_pdb_name.PDB2=DEPSALES
upg1.target_pdb_name.PDB1=EMPLOYEES
upg1.target_pdb_copy_option.PDB2=file_name_convert=('PDB2','DEPSALES')
```

この構成ファイルでは、次の操作をAutoUpgradeに指示します。

- ホーム/u03/app/oracle/product/12.2.0/dbhome_1のソースOracle Database CDB122からPDBを選択します
- PDB PDB2およびPDB1をターゲットのOracle Database 21c Oracleホーム /u01/app/oracle/product/21.0.0/dbhome_1にアップグレードします
- PDB2の名前をDEPSALESに変更し、新しいファイル名DEPSALESを使用してPDB2のファイルをコピーします。
- PDB1の名前をEMPLOYEESに変更します。

親トピック: AutoUpgrade構成ファイルの例

スクリプト・ジョブまたはバッチ・ジョブでのAutoUpgradeの実行方法

noconsoleパラメータを使用してAutoUpgradeを呼び出し、非対話型モードで独自のスクリプトでAutoUpgradeを実行する方法について学習します。

デフォルトでは、AutoUpgradeはコンソール・モードで実行されるため、システムでAutoUpgradeジョブを実行しているときに、これらのジョブの特定の側面を監視するコマンドを実行できます。

ノート:

特定の構成ファイルに関連付けられている AutoUpgrade インスタンスは、一度に 1 つのみ実行できます。

例3-11

この例では、構成ファイルautoupgrade.cfgで指定された設定を使用し、noconsoleパラメータを使用してコンソールをオフにして、AutoUpgradeをデプロイ・モードで実行します。

```
java -jar autoupgrade.jar -config autoupgrade.cfg -mode deploy -noconsole
```

noconsoleモードを使用すると、ユーザー入力の要件がなくなるため、このコマンドをスクリプトに入力すると、構成ファイルで指定したアップグレードを実行できます。

親トピック: AutoUpgrade構成ファイルの例

AutoUpgradeによる切断/接続再配置アップグレード

切断/接続再配置機能(ホット・クローニング・アップグレードとも呼ばれる)を使用して、アップグレード前に、特定の期間にリフレッシュ可能なPDBを作成する方法を確認します。

切断/接続再配置アップグレードを、他のPDBまたは非CDBデータベースからクローンを作成する方法として使用できます。ソース・データベースが非CDBである場合、AutoUpgradeはアップグレード処理の一環として非CDBをアップグレードし、変換します。この機能は、ソース・リリースとしてOracle Database 12cリリース1 (12.1.0.2)以降のリリースと互換性があります。AutoUgradeバージョン22.5以降、AutoUpgradeは、少なくともOracle Database 12cリリース1 (12.1.0.2)のソースを使用した切断/再配置操作をサポートしています。12.1.0.2ソースが使用されている場合、デプロイ操作ではソースを読取り専用モードにし、ターゲットCDBをローカルUNDOモードにしないようにする必要があります。source_dblink parameterを指定する場合、CLONE_ONLYオプションも必要です。Oracle Databaseリリース11g (11.2)は、AutoUpgradeホット・クローニング機能のソース・バージョンとして互換性がありません。

切断/接続再配置アップグレード方法を使用するには、次の構成を完了する必要があります。

- クローニングするデータベースに、CREATE SESSION、CREATE PLUGGABLE DATABASE、SELECT_CATALOG_ROLEおよびGRANT READ ON sys.enc$権限が付与されているユーザーが存在する必要があります。
- PDBを再配置するCDBで、クローニングするデータベースを指すCDBにデータベース・リンク(@dblink)を作成する必要があります。
- ターゲットCDBは、ソース・データベースが構成されているすべてのコンポーネントを構成しておく必要があります。そうしないと、アップグレード・プロセスは失敗します。

AutoUpgrade構成ファイルに入力する必要があるため、データベース・リンクをメモします。

次の例では、非CDBとPDBの両方について、プロセス内の各ステップを実行します。次のシナリオは、2つのホストで構成されます。

- db-node1: ステップ1が実行されるソース・データベースのホストの場所。
- db-node2: ステップ2が実行されるターゲット・データベースのホストの場所。

ノート:

この例では 2 つの別々のホストを使用しますが、この手順は単一のホストで実行することもできます。

1. CREATE SESSION、CREATE PLUGGABLE DATABASEおよびSELECT_CATALOG_ROLE権限を、クローニングするデータベース上でユーザーに付与します。また、SYSで、ユーザーにGRANT READ ON sys.enc$も付与します。この例では、ユーザーcloneがこれらの権限を使用して作成されます

   - 非CDB:

     ```
     alter system set local_listener='(ADDRESS=(PROTOCOL=TCP)(HOST=db-node1)(PORT=1521))';
     ALTER SYSTEM REGISTER;
     CREATE USER clone IDENTIFIED BY some-password;
     GRANT CREATE SESSION, CREATE PLUGGABLE DATABASE, SELECT_CATALOG_ROLE TO clone;
     GRANT READ ON sys.enc$ TO user;
     ```

   - PDB:

     ```
     alter system set local_listener='(ADDRESS=(PROTOCOL=TCP)(HOST=db-node1)(PORT=1521))';
     ALTER SYSTEM REGISTER;
     CREATE USER c##clone IDENTIFIED BY some-password CONTAINER=ALL;
     GRANT CREATE SESSION, CREATE PLUGGABLE DATABASE, SELECT_CATALOG_ROLE TO c##clone CONTAINER=ALL;
     GRANT READ ON sys.enc$ TO c##clone CONTAINER=ALL;
     ```

     PDBでは、特定のPDB名を指定して、ユーザーを作成する範囲をCDB上の単一のPDBに制限することもできます。たとえば:

     ```
     CREATE USER c##clone IDENTIFIED BY some-password CONTAINER=PDBX;
     ```

2. クローニングするソース・システム上のソース・データベースを指すデータベース・リンクを作成して、PDBのクローニング先のCDBを構成します。

   次の例では、データベース・リンクがソース・データベースdb18xを指しています。

   - 非CDB

     次の例では、データベース・リンクがソース・データベースdb18xを指しています。

     ```
     CREATE DATABASE LINK db18x_link
     CONNECT TO clone IDENTIFIED BY some-password
     USING'(DESCRIPTION =(ADDRESS = (PROTOCOL = TCP)(HOST = db-node1)(PORT = 1521))(CONNECT_DATA = (SERVICE_NAME = db18x)))';
     ```

   - PDB

     次の例では、データベース・リンクはCDB cdb18xのソースPDB pdbxを指しています。

     ```
     CREATE DATABASE LINK pdbxcdb18x_link
     CONNECT TO c##clone IDENTIFIED BY some-password
     USING'(DESCRIPTION =(ADDRESS = (PROTOCOL = TCP)(HOST = db-node1) (PORT = 1521))(CONNECT_DATA = (SERVICE_NAME = pdbx)))';
     ```

3. AutoUpgrade構成ファイルを、クローニングするデータベースの識別に必要な情報で更新します。

   クローニングする各データベースは、ファイル名を変換するためにAutoUpgradeローカル・パラメータtarget_pdb_copy_optionを設定する必要があります。

ノート:

ターゲット CDB で、パラメータ DB_CREATE_FILE_DEST または PDB_FILE_NAME_CONVERT を設定し、ターゲット CDB でこれらのパラメータを有効にする場合は、値 prefix.target_pdb_copy_option=file_name_convert=NONE を設定します。ただし、構成ファイルまたはターゲット CDB のいずれかでファイル名とパスを指定しないと、アップグレードは失敗します。

各データベースのローカル・パラメータの接頭辞を使用すると、同じ構成ファイルで非CDBおよびPDBのアップグレードを組み合せて、非CDBとPDBの両方を同じターゲットCDBに再配置できます。次に、非CDBデータベースdb18xがアップグレード用に構成され、PDB pdbxもアップグレード用に構成される構成ファイルの関連部分の例を示します。この場合、db18xのファイル名は、先頭に付加される名前をDB18xからdb18xに変更するように設定します。ファイル名は次のようになります。

```
upg1.sid=db18x
upg1.source_home=/source/18x
upg1.target_home=/target/19x
upg1.target_cdb=cdb19x
upg2.sid=cdb18x
upg2.pdbs=pdbx
upg2.target_cdb=cdb19x
upg2.source_home=/source/18x
upg2.target_home=/target/19x
```

各データベースにはtarget_pdb_copy_optionが設定されています。たとえば、非CDBファイル名は、先頭に付加される文字列がDB18Xからdb18xに変換されます。

```
upg1.target_pdb_copy_option.db18x=file_name_convert=('DB18X', 'db18x')
```

各データベースでは、ローカル・パラメータsource_dblinkをターゲットCDBで作成したデータベース・リンクに設定する必要があります。

```
upg1.source_dblink.db18x=db18x_link
```

一貫性を向上させるために、ローカル・パラメータstart_timeをsource_dblinkと組み合せてオプションのリフレッシュ率(秒)とともに使用し、データのリフレッシュ率を設定し、ジョブの開始時間をスケジュールすることもできます。次の結合したパラメータはAutoUpgradeを開始しますが、アップグレードのデプロイメントに対してAutoUpgradeの開始時から1時間40分の遅延を設定し、ソース・データベース・ファイルからターゲット・データベース・ファイルへのリフレッシュ率を20秒ごととして設定します。

```
upg1.source_dblink.db18x=db18x_link 20
upg1.start_time=+1h40m
```

source_dblinkにリフレッシュ率を設定しない場合、クローニングされたデータベースのデータベース・ファイルは、リフレッシュなしで1回のみクローニングされます。start_timeが設定されていない(アップグレードの処理に遅延がない)場合、source_dblinkリフレッシュ率値は無視されます。

4. 切断/接続再配置デプロイメントを開始し、このタスク用に作成した構成ファイルを指定します。

```
java -jar autoupgrade.jar -config config.cfg -mode deploy
```

例3-12 データ・ファイルに対してリフレッシュ率を使用しない非CDBおよびPDBの切断/接続再配置の構成ファイル

```
global.autoupg_log_dir=/home/oracle/xupg
```

```
#database1
upg1.sid=cdb18x
upg1.target_cdb=cdb19x
upg1.source_home=/databases/ee/product/18x/dbhome_1
upg1.target_home=/databases/ee/product/19x/dbhome_1
upg1.pdbs=pdbx
upg1.target_pdb_name.pdbx=pdbxr
upg1.target_pdb_copy_option.pdbx=file_name_convert=('pdbx', 'pdbxr')
upg1.source_dblink.pdbx=pdbxcdb18x_link
#database2
upg2.sid=db18x
upg2.target_cdb=cdb19x
upg2.source_home=/databases/ee/product/18x/dbhome_1
upg2.target_home=/databases/ee/product/19x/dbhome_1
upg2.target_pdb_copy_option.db18x=file_name_convert=('DB18X', 'db18x')
upg2.source_dblink.db18x=db18x_link
upg2.target_pdb_name.db18x=db18x
```

例3-13 データ・ファイルに対してデプロイメント遅延およびリフレッシュ率を使用した非CDBおよびPDBの切断/接続再配置の構成ファイル

```
global.autoupg_log_dir=/home/oracle/xupg
#database1
upg1.sid=cdb18x
upg1.target_cdb=cdb19x
upg1.source_home=/databases/ee/product/18x/dbhome_1
upg1.target_home=/databases/ee/product/19x/dbhome_1
upg1.pdbs=pdbx
upg1.target_pdb_name.pdbx=pdbxr
upg1.target_pdb_copy_option.pdbx=file_name_convert=('pdbx', 'pdbxr')
upg1.source_dblink.pdbx=pdbxcdb18x_link 600
upg1.start_time=+3h
#database2
upg2.sid=db18x
upg2.target_cdb=cdb19x
upg2.source_home=/databases/ee/product/18x/dbhome_1
upg2.target_home=/databases/ee/product/19x/dbhome_1
upg2.target_pdb_copy_option.db18x=file_name_convert=('DB18X', 'db18x')
upg2.source_dblink.db18x=db18x_link 900
upg2.target_pdb_name.db18x=db18x
upg2.start_time=+3h30m
```

例3-14 PDBの切断/接続再配置のための単純な構成ファイル

```
global.autoupg_log_dir=/home/oracle/xupg #database1
upg1.sid=cdb18x
upg1.target_cdb=cdb19x
upg1.source_home=/databases/ee/product/18x/dbhome_1
upg1.target_home=/databases/ee/product/19x/dbhome_1
upg1.pdbs=pdbx
upg1.target_pdb_copy_option.pdbx=file_name_convert=('pdbx', 'pdbxr')
upg1.source_dblink.pdbx=pdbxcdb18x_link
```

切断/接続アップグレードのその他の例は、Mike DietrichのブログUpgrade Your Database Now!"およびDaniel Overby HansenのブログDatabases are Fun"を参照してください。これらのサイトでは、新しいユースケースの例を定期的に紹介しています。次に2つの例を付記します。

関連項目

- Unplug/Plug Upgrade with AutoUpgrade
- Upgrading an Encrypted PDB

親トピック: AutoUpgrade構成ファイルの例

AutoUpgrade構成ファイルを使用した修正およびチェックの無視

データベースのチェックおよび修正ステップ全体をスキップするには、修正チェックリスト・ファイルから修正のrunfixフラグを読み取り、その修正のフラグをYESからSKIPに設定するようにAutoUpgradeに指示します。

AutoUpgradeがアップグレード用に自動的に実行する修正のデフォルト・リストをオーバーライドするには、ローカル構成ファイル・パラメータchecklistを使用して、AutoUpgradeに修正ステップの既存のチェックリストをロードします。ローカル・パラメータで指定するチェックリストでは、次のように事前チェック修正を設定できます。

- YES (デフォルト): チェックを実行し、修正も実行します
- NO: チェックを実行しますが、修正は実行しません。
- _SKIP: チェックを実行せず、修正も実行しません。

構成ファイルでは、ローカル・パラメータchecklistを使用して、AutoUpgradeに既存のチェックリスト・ファイルを指定します。

```
global.autoupg_log_dir=/home/oracle/autoupg

upg1.sid=db12204
upg1.source_home=/databases/ee/product/12.2.0/dbhome_1
upg1.target_home=/databases/ee/product/21.1.0/dbhome_1
upg1.checklist=/home/oracle/autoupg/db12204/100/prechecks/db11204_checklist.cfg
```

修正チェックリスト・ファイルで、DICTIONARY_STATS修正のrunfixフラグは、このステップをスキップするように設定されています。

```
[SID] [db21.1.0]
==========================================
[container] [db21.1.0]
==========================================
[checkname] EXISTENCE_OF_DATAPUMP_AQ_TABLES
[stage] PRECHECKS
[fixup_available] YES
[runfix] YES
[severity] WARNING
-----------------------------------------------------
[checkname] DICTIONARY_STATS
[stage] PRECHECKS
[fixup_available] YES
[runfix] _SKIP
[severity] RECOMMEND
-----------------------------------------------------
```

親トピック: AutoUpgrade構成ファイルの例

AutoUpgradeを使用したカスタム・スクリプトの実行

デプロイ・プロセスの一環としてAutoUpgradeを使用して独自のスクリプトを実行する方法について学習します。

4つのパラメータを構成ファイルに構成すると、カスタム・スクリプトを実行できます。

- before_action

  データベースが稼働しているときにアップグレード前のステージで、データベースをアップグレードする前に、ソースOracleホームでカスタム・スクリプトを実行します。

- after_action

  構成ファイルで設定されたオプションに応じて、アップグレード後のステージまたはPDBアップグレードのステージで、データベースのアップグレード後にターゲットOracleホームでカスタム・スクリプトを実行します。

- `revert_before_action`

  データベースがリストアされる前に、ターゲットOracleホームでカスタム・スクリプトを実行します(通常はデータベースが稼働している場合)。アップグレード・プロセスの性質上、スクリプトの起動時にデータベースを停止できることに注意してください。このスクリプトを使用して、`before_action`カスタム・スクリプトによって実行されるアクションを元に戻すことができます。

- `revert_after_action`

  データベースがリストアされ、データベースが稼働した後に、ソースOracleホームでカスタム・スクリプトを実行します。このスクリプトを使用して、`after_action`カスタム・スクリプトによって実行されるアクションを元に戻すことができます。

これらのパラメータの詳細は、「AutoUpgradeユーティリティ構成ファイル」のパラメータの説明を参照してください。

例3-15 構成ファイルの`before_action`および`after_action`の使用

次の構成ファイル`config.cfg`の例では、ローカル・パラメータ`before_action`を構成ファイルで使用して、リストア前にパス`/path/to/a/`でスクリプト`script1.sh`を実行し、ローカル・パラメータ`after_action`を使用して、アップグレード後にパス`/path/to/a/`でスクリプト`script2.sh`を実行します。

構成ファイル`config.cfg`の内容:

```
global.autoupg_log_dir=/home/oracle/autoupg
upg1.sid=db12204
upg1.source_home=/databases/ee/product/12.2.0/dbhome_1
upg1.target_home=/databases/ee/product/19.0.0/dbhome_1
upg1.before_action=/path/to/a/script1.sh
upg1.after_action=/path/to/a/script2.sh
```

構成ファイルは、コマンドラインからAutoUpgradeによってコールされます。

```
java -jar autoupgrade.jar -config config.cfg -restore -jobs 100,101,...
```

これらのスクリプトはアップグレード前およびアップグレード後のステージで実行されるため、スクリプトの実行前または実行後に、GRPへのリストアが行われます。スクリプトは実際のGRPプロセス自体には影響せず、これにより、スクリプト・アクションがデータベースに影響を与える可能性がある場合に、再実行がより安全になります。

例3-16 構成ファイルの`revert_before_action`および`revert_after_action`の使用

次の構成ファイル`config.cfg`の例では、ローカル・パラメータ`revert_before_action`を構成ファイルで使用して、リストア前にパス`/path/to/a/`でスクリプト`script3.sh`を実行し、ローカル・パラメータ`revert_after_action`を使用してアップグレード後にパス`/path/to/a/`でスクリプト`script4.sh`を実行します。

```
global.autoupg_log_dir=/home/oracle/autoupg
upg1.sid=db12204
upg1.source_home=/databases/ee/product/12.2.0/dbhome_1
upg1.target_home=/databases/ee/product/19.0.0/dbhome_1
upg1.revert_before_action=/path/to/a/script3.sh
upg1.revert_after_action=/path/to/a/script4.sh
```

再度、構成ファイルを使用して、リストア・ジョブを実行するようにAutoUpgradeに指示します。

```
java -jar autoupgrade.jar -config config.cfg -restore -jobs 100,101,...
```

この場合、スクリプトはデータベースの稼働時にターゲットのOracle Databaseバイナリで実行されます。`revert_before_action`によってコールされる`script3.sh`スクリプトは、データベースが稼働しているとき、GRPへのリストアを開始する前に実行され、`revert_after_action`によってコールされる`script4.sh`スクリプトは、以前のリリースへのリストアが完了し、データベースが元のOracleホームで稼働した後、データベースで実行されます。

関連項目

- AutoUpgradeユーティリティ構成ファイル

親トピック: AutoUpgrade構成ファイルの例

AutoUpgrade内部設定構成ファイル

内部構成設定では、AutoUpgradeの実行方法を制御します。

使用上のノート

これらの構成設定は、参照専用として提供されています。通常、これらのパラメータは使用しないでください。

表3-2 AutoUpgrade用の内部設定構成ファイルのパラメータ

パラメータ	デフォルト	説明
`heartbeatHeartbeatSleep`	1	各ジョブ・ハートビート間の待機する分数。
`heartbeatHeartbeatRetries`	10	失敗したジョブ・ハートビート後に再試行する回数。
`lsjNoConsoleTimer`	30	`noconsole` の `lsj` コマンドの実行時間間隔(秒)。間隔は 7 から 1200 秒の間にする必要があります。
`shutdownJobWaitTime`	10	スケジュール済のアップグレード時に実行中ジョブがジョブ・キューで終了するまで待機する分数。
`systemChecksAbort_timer`	60	ジョブが自動的に終了するのをシステムが確認するまで待機する分数。
`systemChecksOracleHomeReqSpac`	6g	十分な最小ディスク領域(GB)のシステム・チェック。(g は必須です。)
`systemChecksMinCpuIdlePct`	10	システム上にある CPU リソースの残りの使用可能な割合がアップグレードの完了に不十分であることを示す警告アラートしきい値の割合。
`systemChecksMinFreeMemPct`	5	システム・ランダム・アクセス・メモリー(RAM)の残りの使用可能な割合がアップグレードを完了するために不十分であることを示す警告アラートしきい値の割合。
`systemChecksMinFreeSwapPct`	5	システム・スワップ領域メモリーの残りの使用可能な割合がアップグレードを完了するために不十分であることを示す警告アラートしきい値の割

パラメータ	デフォルト	説明
		合。
dbPreCheckAbortTimer	60	データベースのアップグレード前チェックが自動的に終了するまで待機する分数。
dbUpgradeDurationTimer	180	データベースのアップグレード・ジョブがアップグレードの進行状況の追加の監視を開始するまで待機する分数。
dbUpgradeWakeupTimer	3	データベースのアップグレード・ジョブがアップグレードの監視を再開するまで待機する分数。
dbUpgradeAbortTimer	1440	データベースのアップグレードが自動的に終了するまで待機する分数。
dbUpgradeFatalErrors	ORA-00600, ORA-07445	どのアップグレード内部エラーによって、自動的にデータベースのアップグレード後リストアが保証付きリストア・ポイントに戻されるかを識別します。エントリはカンマ区切りです。
dbPostUpgradeAbortTimer	60	アップグレード後のジョブが自動的に終了するまで待機する分数。
dbGrpAbortTimer	3	保証付きリストア・ポイント・ジョブが自動的に終了するまで待機する分数

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

AutoUpgradeログ・ファイルの構造

AutoUpgradeユーティリティは、ジョブ・ステータスおよび構成ファイルを含むログ・ファイルの構造を生成します

AutoUpgradeログ・ファイルのベース・パス

AutoUpgradeログ・ファイルのパスは、グローバル・パラメータautoupg_log_dirを使用して設定されます。デフォルトでは、グローバル・パラメータには次の定義があります。

```
global.autoupg_log_dir=/database/jobmgr
```

AutoUpgrade構成およびステータス・ファイルのパスは、global.autoupg_log_dirで設定するディレクトリ・パスに対する相対パスです。

/cfgtoollogs/upgrade/auto

自動構成ツールのログ・ディレクトリ(/cfgtoollogs/upgrade/auto)には、AutoUpgradeジョブ・マネージャで処理される各ジョブに関する特定の情報を提供する3つのトレース・ログ・ファイルが含まれています。

- autoupgrade.log: ジョブの実行中に発生した問題を識別するジョブの詳細なログを提供します。
- autoupgrade_usr.log: 読みやすいように書式設定されたジョブ情報。
- autoupgrade_err.log: ジョブの実行時に発生する予期しない例外のレポート。

ジョブの開始または停止時に問題が発生した場合、これらのログ・ファイルの情報を使用して、問題の原因を特定できます。

/config_files

config_filesディレクトリには、AutoUpgrade内部ランタイム構成およびグローバル一時ファイルが含まれています。

/status

/statusディレクトリには、JSONジョブ・ステータス・ファイルが含まれています。これには2つのディレクトリがあります。

- status.json: このディレクトリには、JSONファイル形式で完了したすべてのジョブの最終ジョブ・ステータスが含まれています。
- progress.json: このディレクトリには、JSONファイル形式で現在実行中のすべてのジョブの進行状況が含まれています。

ディレクトリ内の各モジュールには、実行された操作のステータス・ファイルが含まれています。モジュールの形式は次のようになります。接頭辞dbnameはデータベース名、operationは実行されたアップグレード操作、接尾辞statusはその操作の完了ステータスです。

```
dbname_operation-name.status
```

その操作の成功または失敗は、接尾辞によって示されます。その操作の実行が成功したことを示す.successか、その操作が失敗したことを示す.failureのいずれかです。たとえば、次のモジュール名は、データベースsalesでの事前チェック操作が正常に実行されたことを示します。

```
sales_prechecks.success
```

操作モジュール名には、次のいずれかを指定できます。

- preupgrade: カスタム・スクリプトを実行できるアップグレード前のステージ。
- prechecks: アップグレードの開始前に完了したアップグレード・チェック。
- grp: 保証付きリストア・ポイント(Oracle Flashback Technologyを使用)

- prefixups: アップグレードを開始する前にアップグレード前の修正を実行します
- drain: アップグレードを開始する前に既存のジョブが完了または移行されるステージ。
- dbupgrade: アップグレードが実行されるステージ。
- postchecks: アップグレードの完了後にアップグレード後のチェックが実行されるステージ。
- postfixups: アップグレード後の修正が実行されるステージ。
- postupgrade: カスタムのアップグレード後スクリプトを実行できるステージ。

個々のジョブおよびデータベース・ログ・ファイル・ディレクトリ

AutoUpgradeディスパッチャによって起動される各ジョブには、そのジョブ識別子の接頭辞を持つディレクトリが与えられます。ジョブ・ディレクトリの内部では、ジョブ内の各データベースにパス/database/logs/sidにログ・ディレクトリが提供されます。ここで、sidはデータベースのシステム識別子です。たとえば、構成ファイルで識別されたジョブがsales1で、データベース・システム識別子がsalesの場合、データベースsalesのログ・ファイルのパスは次のようになります。

```
sales.log_dir=/database/logs/sales1
```

ログ・ディレクトリには、そのデータベースに対して実行されたすべてのタスクに関するすべての関連ログ・ファイルが含まれています。デフォルトでは、SIDで識別されるディレクトリが/database/logsディレクトリの下に作成されます。この方法で構成ファイルを設定することを選択した場合、各データベース・ジョブに個別のログ・ディレクトリを設定できます。

/#### (ジョブ番号)

個々のジョブ実行は、/####という形式の実行番号で識別されるサブディレクトリに配置されます。ここで、####は実行番号を表します。たとえば、0004となります。ジョブ実行番号ディレクトリには、次のログ・ファイルが含まれます。

- autoupgrade_err.log: ジョブの実行中に発生する予期しない例外をレポートします。
- autoupgradeYYYYMMDD.log: AutoUpgradeトレース・ログ・ファイル。ジョブの実行中に発生した問題を識別するジョブの詳細なログを提供します。変数YYYYMMDDは、ジョブの年、月および日を表します。
- autoupgrade_YYYYMMDD_user.log: AutoUpgradeジョブのステータス・ファイルで、読みやすいように書式設定されています。変数YYYYYYMMDDは、ジョブの年、月および日を表します。

/preupgrade

アップグレード前ディレクトリ(/preupgrade)には、次のファイルとログ・ファイルが含まれています。

- prechecks_databasename.log: トレース・ログ・ファイル。このファイルは、アップグレード前のジョブ・ステージ中に発生した問題の識別に役立つ詳細なログを提供します。変数databasenameは、チェックされたデータベースの名前です。
- databasename_preupgrade.html: データベースのステータスに関するHTMLレポート。変数databasenameは、チェックされたデータベースの名前です。
- databasename_preupgrade.html: データベースのステータスに関するテキスト・レポート。変数databasenameは、チェックされたデータベースの名前です。

/dbupgrade

データベース・アップグレード・ディレクトリ(/dbupgrade)には、データベース・アップグレードに関連付けられたすべてのログ・ファイルが含まれています。

- autoupgradeYYYYMMDDHHMISCdbname.log: アップグレードが実行された日付および並列度を示すデータベース名で識別されるソース・データベースのログ・ファイル。形式は次のとおりです。
  - YYYY: 年
  - MM: 月

    - DD: 日
    - HH: 時
    - MI: 分
    - SC: 秒
    - dbname: データベース名。dbnameはデータベース名です。
- catupgradeYYYYMMDDHHMISCdbnameN.log: アップグレードが実行された日付で識別されるソース・データベースのログ・ファイル。

  形式は次のとおりです。

    - YYYY: 年
    - MM: 月
    - DD: 日
    - HH: 時
    - MI: 分
    - SC: 秒
    - dbnameN: データベース名。dbnameはデータベース名で、Nは並列度を示します。CDB ROOTおよび非CDBデータベースの場合は0...3で、PDBの場合は0...1です。

/temp

一時AutoUpgradeファイル(/temp)。このディレクトリには、アップグレード時に使用されるPFILEなどのファイルを含めることができます。

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

AutoUpgradeのための完全なデプロイの有効化

保証付きリストア・ポイント(GRP)を有効にしてアップグレードをフラッシュバックできるようにするには、アーカイブ・ロギングを設定し、AutoUpgradeを有効にしてアップグレードを完了できるように他のタスクを完了する必要があります。

AutoUpgradeで新しいリリースのOracle Databaseの完全なデプロイを実行できるようにするには、次の条件を満たす必要があります。

- データベースには、高速リカバリ領域(FRA)の適切な構成が必要です。具体的には、`DB_RECOVERY_FILE_DEST`および`DB_RECOVERY_FILE_DEST_SIZE`を設定し、適切なサイズにする必要があります。
- ソースOracle Databaseは、`ARCHIVELOG`モードで実行する必要があります。

> ノート:
>
> AutoUpgrade では、デプロイ処理モードで保証付きリストア・ポイント(GRP)を作成します。以前に定義された保証付きリストア・ポイントは必要ありません。

例3-17 AutoUpgradeを使用する前のアーカイブ・ロギングおよび高速リカバリ領域(FRA)の設定

次の例では、`your-directory-or-diskgroup`は、リカバリ領域が配置されているディレクトリ・パスまたはディスク・グループです。`DB_RECOVERY_FILE_DEST_SIZE`の値は50GBに指定されますが、リカバリ領域に必要な値を使用する必要があります。

```
sqlplus / as sysdba
shutdown immediate;
startup mount;
alter system set db_recovery_file_dest_size = 50g scope=both sid='*';
alter system set db_recovery_file_dest ='your-directory-or-diskgroup' scope=both
sid='*';
alter database archivelog;
alter database open;
alter pluggable database all open;
```

例3-18 パスワード・ファイルとセキュリティ・パスワード・ファイルの更新

アップグレード中に、AutoUpgradeユーティリティはソースOracleデータベースのOracleホームからターゲットのOracleデータベースのOracleホームにパスワード・ファイルをコピーします。ただし、コピーされたパスワード・ファイルは以前のリリースのパスワード・ファイル・バージョンを保持します。パスワード・ファイルを再生成して、新しいリリース・パスワード・ファイル・バージョンに更新することをお薦めします。

例3-19 透過的データ暗号化およびAutoUpgrade

AutoUpgradeユーティリティで透過的データ暗号化キーストアのコピーに必要な権限を取得できるようにするには、AutoUpgradeがターゲット・リリースのOracleホームにコピーできるように、これらのキーストアに対する自動ログインを有効にする必要があります。自動ログインを有効にしない場合、AutoUpgradeはアップグレードを完了できません。

また、AutoUpgradeでは、`TNS_ADMIN`環境変数を使用してOracle Net管理ディレクトリの場所が検索されます。`TNS_ADMIN`環境変数が定義されていない場合、ネットワーク管理ディレクトリのパスはデフォルトでパス`Oracle_home/network/admin`になります。別のパスを指定する必要がある場合は、ローカル構成パラメータ`source_tns_admin_dir`を使用して構成ファイルにパスを指定し、必要に応じて`target_tns_admin_dir`を使用し

てターゲット・パスを指定します。

既存のキーストアに対するADMINISTRATOR KEY MANAGEMENTまたはSYSKM権限を使用して、自動ログイン・ソフトウェア・キーストアまたはローカル自動ログイン・ソフトウェア・キーストアを有効にします。たとえば、パスワード保護されたキーストアの自動ログイン・ソフトウェア・キーストアを /etc/ORACLE/WALLETS/orclディレクトリに作成するには、次のようにします。

```
ADMINISTER KEY MANAGEMENT CREATE AUTO_LOGIN KEYSTORE
FROM KEYSTORE '/etc/ORACLE/WALLETS/tde'
IDENTIFIED BY password;
keystore altered.
```

ノート:

AutoUpgrade が透過的データ暗号化キーストアのコピーを完了した後、自動ログインを無効にして、以前のリリースのセキュリティがリストアされるようにします。

関連項目

- 自動ログイン・ソフトウェア・キーストアまたはローカル自動ログイン・ソフトウェア・キーストアの作成
- ローカル・ネーミング・パラメータの概要
- 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
- DB_RECOVERY_FILE_DEST

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

AutoUpgradeコンソールの使用例

AutoUpgradeコンソールには、AutoUpgradeジョブの進行状況を監視する一連のコマンドが用意されています。コンソールは、AutoUpgradeユーティリティの実行時にデフォルトで起動され、パラメータ`console`および`noconsole`によって有効または無効になります。

コンソール・モードでは、AutoUpgradeコンソールにより、システムでAutoUpgradeジョブを実行しているときに、これらのジョブの特定の側面を監視するコマンドを実行できます。

> ノート:
>
> AutoUpgrade コンソールが完了前に終了した場合、実行中のジョブは停止し、スケジュールされたジョブは起動しません。このため、AutoUpgrade のすべてのジョブが完了するまではコンソールを終了したり AutoUpgrade プロセスを停止しないでください。

例3-20 AutoUpgradeコンソールを有効および無効にする方法の例

> ノート:
>
> 特定の構成ファイルに関連付けられている AutoUpgrade インスタンスは、一度に 1 つのみ実行できます。

この例では、AutoUpgradeは`noconsole`モードで構成ファイルを使用して分析モードで実行されます。

```
java -jar autoupgrade.jar -config autoupgrade.cfg -mode analyze -noconsole
```

`noconsole`モードを使用すると、ユーザー入力が不要になるため、スクリプトの使用に適しています。

この例では、AutoUpgradeは分析モードで実行され、`-console`オプションを指定しているのでコンソールは再度有効になります。

```
java -jar autoupgrade.jar -config autoupgrade.cfg -mode analyze -console
```

コンソール・ユーザーの入力が再開されます。

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

AutoUpgradeの既知の制限

アップグレードで問題が発生した場合は、既知の制限を確認して解決策を見つけてください。

- AutoUpgradeおよびディスク領域の問題
  AutoUpgradeの実行中にディスク領域が不足した場合は、ここで説明する解決策を確認して適用します。

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

AutoUpgradeおよびディスク領域の問題

AutoUpgradeの実行中にディスク領域が不足した場合は、ここで説明する解決策を確認して適用します。

ディスク領域の問題を回避するために役立ちます。Oracleは、必要なディスク領域を計算するためのチェックを追加しました。ログ・サイズ指定を確認するには、AutoUpgrade分析レポートを確認します。アップグレード中にデータベースのディスク領域が不足した場合は、データベースに必要なディスク領域を解放し、必要に応じてジョブを再開します。

例3-21 アップグレード実行時のディスク領域エラー

1. 現在実行中のプロファイルを停止します

```
abort -job 100
```

2. アップグレードを再開できるようにディスク領域を解放します。
3. 停止したアップグレードを再開します

```
resume -job 100
```

例3-22 アップグレード完了後のディスク領域エラー

AutoUpgradeがディスク領域エラーで完了した場合は、次の手順を実行します。

1. アップグレードを再開できるようにディスク領域を解放します。
2. AutoUpgradeを再起動して、残りの問題を完了させます。

   たとえば、構成ファイル名が`yourconfigfile.cfg`の場合:

```
java -jar autoupgrade.jar -mode deploy -config yourconfigfile.cfg
```

親トピック: AutoUpgradeの既知の制限

AutoUpgradeデータベース変更の適切な管理

AutoUpgradeは強力なユーティリティであり、責任を持って使用する必要があります。データベースを危険にさらすような方法でAutoUpgradeを使用することを確認し、回避してください。

次に、AutoUpgradeの不適切な使用と、これらのエラーによって発生する問題の回避方法を一覧で示します。

デプロイ中にAutoUpgradeの再開機能を使用不可にする

問題の説明: `-clean_recovery_data`オプションを使用すると、AutoUpgradeでデータベースを再開またはリストアできなくなります。

回避策: データベースのバックアップ・コピーからリストアします。

原因: AutoUpgradeをデプロイ・モードで実行してから、修正完了後の任意の段階でその実行を中断し、AutoUpgradeのデプロイ・コマンドの正常な再開および完了の前にオプション`clear_recovery_data`を実行しました。たとえば:

```
java -jar autoupgrade.jar -config config.cfg -mode deploy
Ctrl+C //sample interruption
java -jar autoupgrade.jar -config config.cfg -clear_recovery_data
java -jar autoupgrade.jar -config config.cfg -mode deploy
```

デプロイ中またはデプロイ後にAutoUpgradeのグローバル・ログ・ディレクトリを変更する

問題の説明: デプロイ・コマンドの実行中または実行後にグローバル・ディレクトリを変更すると、AutoUpgradeユーティリティで、保留中の作業を再開できなくなります。

回避策: データベースのバックアップ・コピーからリストアします。

原因: AutoUpgradeのグローバル・ログ・ディレクトリには、その操作の状態を追跡するためにAutoUpgradeユーティリティで使用されるファイルも含まれています。このツールをデプロイ・モードで実行し、デプロイ操作を停止し、グローバル・ログ・ディレクトリを名前変更または削除した場合は、AutoUpgradeユーティリティで、デプロイ操作が停止されたときの状態を判別できません。その結果、AutoUpgradeを再起動すると、アップグレード操作が最初から開始され、初期GRPが上書きされます。そのGRPを使用して元のデータベースをリストアすることはできません。

AUTOLOGINで設定されていない資格証明でのキーストアの使用

問題の説明: ツールを実行し、キーストアの資格証明が構成されていましたが、AutoUpgradeがデータベースにログインできません。

回避策: 自動ログイン・キーストアを作成し、自動ログインが有効な状態でデータベースを構成します。

このリリースの時点。AutoUpgradeは、データベースへの自動ログイン用に構成されていないかぎり、キーストア資格証明の使用をサポートしません。

関連項目

- キーストア・パスワードを必要とする操作の実行

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

デフォルト修正のオーバーライド方法

RUNFIX列エントリを使用すると、修正の無効化がセキュリティまたはOracleポリシーに違反する場合を除き、自動修正を無効にできます。

AutoUpgradeプロシージャの一部であるデフォルトの修正は、分析処理モード・ステージ中に生成されます。生成された修正リストを変更して自動修正を無効にすると、独自の修正を実行できます。

ステップの順序は次のとおりです。

1. 分析モードでAutoUpgradeユーティリティを実行します
2. 分析モードで生成される`sid_checklist.cfg`ファイルを開いて編集し、手動で実行する必要がある修正が自動的に実行されないようにします。
3. 最新のAutoUpgrade構成ファイルの自動修正を抑制するジョブのローカル・パラメータ・リストで、`sid.checklist`パラメータを探します。`sid`は、自動修正を抑制するデータベースのシステム識別子(SID)です。デフォルトでは、AutoUpgradeは最新の生成されたファイルを使用します。別の構成ファイルを指すようにするには、このパラメータを編集し、編集した`checklist.cfg`ファイルへのパスを指定します。

ノート:

AutoUpgrade の再開では、常に最新の `sid_checklist.config` ファイルが使用されます。

たとえば、生成された AutoUpgrade 構成ファイルが 2 つある場合(`/logdir/100/sid_checklist.cfg` および`/logdir/101/sid_checklist.cfg`)は、使用する構成ファイルへの直接パスを指定するか、最新のファイル(この場合は`/logdir/101/`の `sid_checklist.cfg` ファイル)を編集する必要があります。

検出した問題を手動で修正し、AutoUpgrade が別のチェックで修正ファイルを使用するようにするとします。AutoUpgrade に別のファイルを使用するように指示する場合は、`prefix.checklist` パラメータを使用して `checklist.cfg` ファイルにファイル・パスを直接指定できます。`prefix` はデータベースの識別子です。たとえば、`prefix.checklist=logdir/repress-standard/sid_checklist.cfg` です

AutoUpgrade ユーティリティでエラー・レベルのデータベース状態が見つかり、これに対して使用可能な修正がない場合、または使用可能な修正を手動で無効にした場合、エラー状態のデータベースを含む AutoUpgrade ジョブが停止します。

この手順を利用するには、次の例を使用します。

例3-23 分析モードでのAutoUpgradeユーティリティの起動

```
java -jar autoupgrade.jar -config config.cfg –mode analyze
```

例3-24 構成ファイルの新規チェックリストの作成

このシナリオでは、リリース18cへのアップグレード準備として、Oracle Database 11gリリース2 (11.2)のデータベース・ホームで、リリース更新8を使用してAutoUpgradeチェックを実行しています。構成ファイル`config.cfg`から開始します。

```
global.autoupg_log_dir=/home/oracle/autoupg
upg1.sid=db11204
upg1.source_home=/databases/ee/product/11.2.0/dbhome_1
```

```
upg1.target_home=/databases/ee/product/18x/dbhome_1
upg1.target_base=/databases
upg1.target_version=18.8.0
```

次のステップを完了します。

1. コマンド`java -jar autoupgrade.jar -config config.cfg -mode analyze`を実行します

   このコマンドは、次のパスにチェックリスト・ファイルを生成します。

   ```
   /home/oracle/autoupg/db11204/100/prechecks/db11204_checklist.cfg
   ```

2. チェックリスト・ファイルを別の場所に移動します。たとえば:

   ```
   oracle@example: $ cd $PRECHECKS
   oracle@example: $ pwd
   /home/oracle/autoupg/db11204/100/prechecks
   oracle@example: $ mv ./db11204_checklist.cfg /tmp
   ```

3. テキスト・エディタを使用してファイルを開き、チェック`AMD_EXISTS`と`EM_PRESENT`を探します。

   たとえば:

   ```

   [SID] [db11204]
   ==========================================
   [container] [db11204]
   ==========================================
   [checkname] AMD_EXISTS
   [stage] PRECHECKS
   [fixup_available] YES
   [runfix] YES
   [severity] WARNING
   -----------------------------------------------------
   [checkname] DICTIONARY_STATS
   [stage] PRECHECKS
   [fixup_available] YES
   [runfix] YES
   [severity] RECOMMEND
   -----------------------------------------------------
   [checkname] EM_PRESENT
   [stage] PRECHECKS
   [fixup_available] YES
   [runfix] YES
   [severity] WARNING
   -----------------------------------------------------
   ...
   [truncated]
   ```

4. チェック`AMD_EXISTS`および`EM_PRESENT`の値を`yes`から`no`に変更します。

   たとえば

   ```

   [SID] [db11204]
   ==========================================
   [container] [db11204]
   ==========================================
   [checkname] AMD_EXISTS
   [stage] PRECHECKS
   [fixup_available] YES
   [runfix] NO
   [severity] WARNING
   -----------------------------------------------------
   [checkname] DICTIONARY_STATS
   ```

```
[stage] PRECHECKS
[fixup_available] YES
[runfix] YES
[severity] RECOMMEND
-----------------------------------------------------
[checkname] EM_PRESENT
[stage] PRECHECKS
[fixup_available] YES
[runfix] NO
[severity] WARNING
-----------------------------------------------------
...
[truncated]
```

両方のパラメータでfixup_available値がYESであることに注意してください。つまり、使用可能な修正が存在し、それを実行しないように選択することを意味します。使用可能な修正がない場合、runfixの値はN/Aです。

5. パラメータ・チェックリストのパスを更新して、AutoUpgradeが構成ファイルを検索する場所を変更します。これを実行するには、編集したファイルが存在するディレクトリへのチェックリスト・ポインタを含むエントリを構成ファイルに追加します。たとえば:

```
global.autoupg_log_dir=/home/oracle/autoupg
upg1.sid=db11204
upg1.source_home=/databases/ee/product/11.2.0/dbhome_1
upg1.target_home=/databases/ee/product/18x/dbhome_1
upg1.target_base=/databases
upg1.target_version=18.8.0
upg1.checklist=/home/oracle/db11204_checklist.cfg
```

6. 編集および移動した構成ファイルを使用して修正を実行します。たとえば:

```
java -jar autoupgrade.jar -config config.cfg -mode fixups
```

自動アップグレードでは、/home/oracleにある構成ファイルを使用してAutoUpgradeユーティリティを実行します。

例3-25 checklist.cfgの検索および編集

分析モードでは、ファイル名がchecklist.cfgの修正ファイルが生成されます。ファイルに移動します。DATABASE_LOGS_DIRはデータベースのAutoUpgradeのlog_dirパラメータに設定された値で、job-idはAutoUpgradeユーティリティによって生成されるジョブ識別子を参照し、sidは自動修正を抑制するデータベースのシステム識別子です。

```
DATABASE_LOGS_DIR/job-id/prechecks/sid_checklist.cfg
```

テキスト・エディタを使用してchecklist.cfgファイルを開きます。次に、SIDがDB11204のデータベースのchecklist.cfgファイルの例を示します。

```
[dbname]          [DB11204]
==========================================
[container]          [DB11204]
==========================================
[checkname]          AMD_EXISTS
[stage]              PRECHECKS
[fixup_available]    YES
[runfix]             YES
[severity]           WARNING
-----------------------------------------------------
[checkname]          DEPEND_USR_TABLES
[stage]              POSTCHECKS
[fixup_available]    YES
[runfix]             YES
```

```
[severity]            WARNING
-----------------------------------------------------
[checkname]           DICTIONARY_STATS
[stage]               PRECHECKS
[fixup_available]     YES
[runfix]              YES
[severity]            RECOMMEND
-----------------------------------------------------
[checkname]           EM_PRESENT
[stage]               PRECHECKS
[fixup_available]     YES
[runfix]              YES
[severity]            WARNING
-----------------------------------------------------
.
.
.
```

ファイルには階層構造があります。これはデータベース名、およびchecklist.cfgのエントリが適用されるコンテナ名で始まります。ファイルには、自動的に適用される一連の修正チェックが含まれています。各エントリには、次の5つの関連値パラメータがあります。

パラメータ	説明
[checkname]	データベース・チェックの名前
[stage]	チェックが実行される AutoUpgrade ステージ。事前チェックまたは事後チェックが可能です。
[fixup available]	自動修正の可用性。このパラメータ値は、YES(自動修正が可能)または NO(自動修正は不可)のいずれかです。
runfix	修正の実行ステータス。このパラメータは、次のいずれかの値を使用します。 YES: 修正を実行します。 NO: 修正を実行しません。
severity	自動修正が対応する問題の重大度のクラス。

手動で実行する修正ごとに、[runfix]パラメータ値をYESからNOに変更します。

関連項目

- My Oracle SupportドキュメントID 2380601.1「データベースのアップグレード前ツールのチェック・リスト」

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

ローカル構成ファイルのパラメータ修正チェックリストの例

アップグレード時に個々のデータベースに対して特定の修正を含めるか除外する場合は、ローカル構成ファイル・チェックリストを使用します。

この例では、sales4_checklist.cfgという名前のローカル・チェックリスト・ファイルがあり、データベースsales4のアップグレード前修正チェックリストが提供されます。ファイルの一部には、次の設定が含まれています。

```
[checkname] DICTIONARY_STATS
[stage] PRECHECKS
[fixup_available] YES
[runfix] YES
[severity] RECOMMEND
```

DICTIONARY_STATSに対するデフォルトの修正を変更し、チェックのrunfixオプションをYESからNOに変更することで、データベースsales4の修正の実行を除外できます。

```
[checkname] DICTIONARY_STATS
[stage] PRECHECKS
[fixup_available] YES
[runfix] NO
[severity] RECOMMEND
```

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

AutoUpgradeおよびMicrosoft WindowsのACLとCLI

Microsoft WindowsシステムでAutoUpgradeを実行する場合、アクセス制御リスト(ACL)およびコマンドライン・インタフェース(CLI)を使用した追加のベスト・プラクティスをお薦めします。

AutoUpgradeおよびアクセス制御リスト(ACL)

WindowsシステムでAutoUpgradeを使用する場合、Windowsアクセス制御リストを使用するための自動ツールの設定は簡単ではありません。次のベスト・プラクティス手順を実行することをお薦めします。

- 各ターゲット・データベースに対する権限、およびこれらの権限がデータベースのディレクトリ(Oracleベース・ディレクトリ、`oraInventory`ファイルなど)にどのように関連しているかを確認します。
- NTFSファイル・システムのインストール後の構成タスクについては、『*Oracle Database*管理者リファレンス』の項を参照してください。
- Windows PowerShellに関するMicrosoftのドキュメントを確認してください。
- `ORA_DBA`、`ORA_HOME_USERS`および`ORA_ASM`グループの権限を確認してください。`ORA_DBA`グループでは、Oracle DatabaseへのSYSDBA権限のみを提供します。Oracle ASM管理権限は、グループASM `DISKGROUPS`のメンバーによって制御されます。
- 多数の管理タスクを実行するための`ORA_DBA`グループ・メンバー権限は、`POSIX`システムで使用可能な権限と比較して制限されています。AutoUpgradeを期待どおりに実行できるようにするには、Microsoft Windowsサーバーの管理者権限を持つユーザー・アカウントで、トレース、リスナー、構成などのOracleベース・ディレクトリ要素も管理することをお薦めします。
- AutoUpgradeは、管理権限を持つコマンド・コンソール(`CMD`)を使用して実行する必要があります。そのコンソールは、Oracleインストール・ユーザーまたは同様の権限を持つユーザーとして開く必要があります。
- My Oracle Supportノート1529702.1および1595375.1を参照してください。

AutoUpgradeでは、データベースで実行されているサービスとともに次の手順を使用します。

- AutoUpgradeは、ソース・データベースのサービスを停止し、ターゲット・データベースのOracleホームに一時サービスを作成します。

  リストアが必要な場合、ターゲット内のサービスが削除され、ソース内のサービスが再起動されます。

- デプロイ・モードの処理が正常に完了すると、ターゲット内のサービスが削除されます。この時点では、アップグレードしたOracle Databaseに対するDBAが`ORADIM`を使用してサービスを作成します。AutoUpgradeではORADIMサービスを作成する権限を取得するためのパスワードがないため、このサービスを手動で作成する必要があります。

AutoUpgradeおよびWindowsコマンドライン・インタフェース

Windows上のコマンドライン・インタフェースでは、戻り文字がコンソールに送信されるのを待機している間に、アプリケーションが応答を停止する可能性があります。この動作は、AutoUpgradeユーティリティに影響を与える可能性があります。原因は、QEMに関連する既知のMicrosoft Windowsコンソール・ウィンドウ特性(クイック編集モード)にあります。クイック編集コンソール・モードを無効にした場合でも、アプリケーションではこの動作が発生することがあります。

プログラムが応答を待機しないようにするには、アプリケーションの起動後に[Enter]キーを数回押します。これを行うと、端末の応答を待機せずにアプリケーションを続行するために必要な端末入力が提供されます。

関連項目

- Oracle Database管理者リファレンスfor Microsoft Windows

- My Oracle Supportノート1529702.1
- My Oracle Supportノート1595375.1

親トピック: Oracle DatabaseアップグレードでのAutoUpgradeの使用

4 パラレル・アップグレード・ユーティリティを使用したOracle Databaseの手動アップグレード

パラレル・アップグレード・ユーティリティのコマンドライン・オプションを使用して手動でアップグレードできます。

Oracle Database 19c以降、Database Upgrade Assistant (DBUA)はAutoUpgradeユーティリティに置き換えられています。

> 注意:
>
> 古い Oracle Database ソフトウェアを保持している場合、アップグレードしたデータベースを古い Oracle Database ソフトウェアを使用して起動しないでください。新しい Oracle Database インストールの実行可能ファイルでのみデータベースを起動してください。

- パラレル・アップグレード・ユーティリティを使用した手動アップグレード
  手動で実行および管理するスクリプトを使用してアップグレードを実行するには、パラレル・アップグレード・ユーティリティ(`catctl.pl`)を使用できます。
- マルチテナント・アーキテクチャのOracle Databaseの手動アップグレード・シナリオ
  手動アップグレードの準備のために、マルチテナント・アーキテクチャでデプロイされたOracle Databaseの手動アップグレード・シナリオおよび手順を参照してください。
- Windowsでの非CDBのOracle Databaseの手動アップグレードの例
  次の例では、Oracle Database 11gリリース2 (11.2.0.4)をOracle Database 19cにアップグレードするために実行するアップグレード前チェック、アップグレード、アップグレード後のチェックのステップを示しています。
- Dbupgradeスクリプトおよび以前のリリースのOracle Databaseのcatupgrd.sqlについて
  `catupgrd.sql`スクリプトの機能は、パラレル・アップグレード・ユーティリティ`catctl.pl`と、`dbupgrade`および`dbupgrade.cmd`スクリプトによって置き換えられています。
- データベースのトランスポートおよびアップグレードについて(フル・トランスポータブル・エクスポート/インポート)
  データの転送には、ファイルベース・モードまたは非ファイルベース・モードを使用できます。
- 非CDB Oracle Databaseのアップグレード・シナリオ
  非CDB Oracle Databaseのアップグレード・シナリオおよび手順を理解するには、これらのトピックを参照してください
- Oracle Databaseのアップグレードの再実行
  これらのオプションを使用してアップグレードを再実行します。

パラレル・アップグレード・ユーティリティを使用した手動アップグレード

手動で実行および管理するスクリプトを使用してアップグレードを実行するには、パラレル・アップグレード・ユーティリティ(`catctl.pl`)を使用できます。

- Oracle Databaseのパラレル・アップグレード・ユーティリティについて(CATCTL.PLおよびDBUPGRADE)
  パラレル・アップグレード・ユーティリティ(`catctl.pl`、および`dbupgrade` スクリプト)によって、特定の順序でアップグレードを実行する必要のないコンポーネントを同時にアップグレードできます。
- パラレル・アップグレード・ユーティリティを実行するための一般的なステップ
  参照してOracle Databaseのパラレル・アップグレード・ユーティリティの使用方法の概要を把握してください。
- パラレル・アップグレード・ユーティリティ(catctl.pl)のパラメータ
  パラレル・アップグレード・ユーティリティ(`catctl.pl`)の実行方法を制御します。これらの引数を使用して`dbupgrade`シェル・コマンドを実行することもできます。
- パラレル・アップグレード・ユーティリティの使用例
  この例を使用して、パラレル・アップグレード・ユーティリティを手動で実行してアップグレードを実行する方法を理解してください。

親トピック: パラレル・アップグレード・ユーティリティを使用したOracle Databaseの手動アップグレード

Oracle Databaseのパラレル・アップグレード・ユーティリティについて(CATCTL.PLおよびDBUPGRADE)

パラレル・アップグレード・ユーティリティ(`catctl.pl`、および`dbupgrade` スクリプト)によって、特定の順序でアップグレードを実行する必要のないコンポーネントを同時にアップグレードできます。

Oracle Database 12cリリース1 (12.1)には、パラレル・アップグレード・ユーティリティ`catctl.pl`が導入されています。このユーティリティでは、データベース・ディクショナリをパラレルにロードし、データベースをアップグレードするために複数のSQLプロセスを使用することで、アップグレードの実行にかかる合計時間を短縮します。コンポーネントのパラレル・アップグレードを実行することで、CPU容量を十分に活用できます。手動によるアップグレードとDatabase Upgrade Assistant (DBUA)を使用して実行するアップグレードの両方でアップグレード処理を簡易化するための改善を引き続き行います。DBUAおよび手動アップグレード手順では、新しいパラレル・アップグレード・ユーティリティを活用します。

シェル・コマンド、`dbupgrade`を実行できます。このコマンドは`catctl.pl`をPerlから実行する必要がなく、コマンド・ラインから開始します。

`dbupgrade`シェル・コマンドはファイル・パス`$ORACLE_HOME/bin` (LinuxおよびUNIXの場合)、および`%ORACLE_HOME%¥bin` (Windowsの場合)にあります。`catctl.pl`に有効なコマンド引数をシェル・コマンドに指定できます。コマンドは新しいOracleホーム・パスから直接実行するか、またはユーザー環境変数を設定してそのファイル・パスをポイントします。

たとえば:

デフォルト値による実行:

```
$ ./dbupgrade
```

/tmpに配置されたログ・ディレクトリを指定して実行:

```
$ ./dbupgrade -l /tmp
```

優先度リストを使用して、パラレル・アップグレード・ユーティリティを実行することもできます。たとえば:

```
$ ./dbupgrade -L priority_list_name
```

優先度リストを使用する場合は、アップグレードでPDBの特定のリストを含めたり除外できます。

また、優先度エミュレーションを使用してパラレル・アップグレード・ユーティリティを実行し、実際にアップグレードを実行せずに優先度リストの読取り方法と実行方法を確認することもできます。たとえば:

```
$ ./dbupgrade -E
```

関連項目

- 優先度リスト・エミュレーションを使用したアップグレードのテストの例

親トピック: パラレル・アップグレード・ユーティリティを使用した手動アップグレード

パラレル・アップグレード・ユーティリティを実行するための一般的なステップ

参照してOracle Databaseのパラレル・アップグレード・ユーティリティの使用方法の概要を把握してください。

パラレル・アップグレード・ユーティリティ(catctl.pl、これはシェル・コマンドdbupgradeを使用して起動します)はデータ・ディクショナリとコンポーネントを並行でロードします。並行でロードすると全体的なアップグレード時間が短縮されます。パラレル・アップグレード・ユーティリティを実行する前に、アップグレード前に通常行うデータベースのバックアップ手順に従います。また、前提条件として、preupgrade句を使用してAutoUpgradeを実行し、アップグレードを続行する前にデータベース管理者が対処する必要がある問題を特定する必要があります。

パラレル・アップグレード・ユーティリティを使用してデータベースをアップグレードするための一般的なステップは次のとおりです。

1. 現行のデータベースをバックアップします。
2. 新しいリリースのOracle Databaseソフトウェアをインストールします。
3. ソース・データベースでpreupgradeパラメータを指定してAutoUpgradeを実行し、AutoUpgradeで修正されない問題を修正します。
4. 現在のデータベースを停止します。
5. 新しいリリースのデータベース・ソフトウェアにアクセスするように新しいOracleホーム環境を設定し、ディレクトリORACLE_HOME/rdbms/adminからSQL*Plusを起動します。
6. SYSDBAシステム権限を使用してユーザー・アカウントにログインし、アップグレードするデータベースに接続します。

   ```
   CONNECT / AS SYSDBA
   ```

7. データベースをアップグレード・モードで起動します。構成タイプに応じたコマンドを使用します。

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

> ノート:
>
> UPGRADE キーワードは、アップグレード用に環境を準備する操作を実行します。

起動コマンドでPFILEオプションを使用して、初期化パラメータ・ファイルの場所を指定する必要があります。

データベースをアップグレード・モードで起動すると、catctl.plスクリプトを実行するまでは、固定ビューへの問合せのみ、エラーにならずに実行できます。catctl.plを実行するまでは、PL/SQLの使用を試みたり、他のビューに問合せを実行しようとすると、エラーが返されます。

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ノートにとり、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。

8. SQL*Plusを終了します。
9. 新しいOracleホームからパラレル・アップグレード・ユーティリティを実行します。

   ユーティリティはシェル・コマンド(LinuxおよびUnixではdbupgrade、Microsoft Windowsの場合はdbupgrade.cmd)として、またはPerlコマンド(catctl.pl)として実行できます。

   たとえば、LinuxおよびUnixの場合は次のようにします。

```
cd $ORACLE_HOME/bin
./dbupgrade
```

   たとえば、Microsoft Windowsでは次のように指定します。

```
cd %ORACLE_HOME%¥bin
dbupgrade
```

   パラレル・アップグレード・ユーティリティによって、アップグレード処理が開始されます。

> ノート:
>
> パラレル・アップグレード・ユーティリティはアップグレードを実行するために他のファイルを使用します。Linux および Unix システムでは、これらのファイルには catconst.pm、catcom.pm、sqlpatch、sqlpatch.pl または sqlpatch.pm および orahome が含まれます。Windows システムでは、これらのファイルには orahome.exe が含まれます。これらのファイルを変更または削除しないでください。

関連項目

- 起動時における初期化パラメータの指定

親トピック: パラレル・アップグレード・ユーティリティを使用した手動アップグレード

パラレル・アップグレード・ユーティリティ(catctl.pl)のパラメータ

パラレル・アップグレード・ユーティリティ(catctl.pl)の実行方法を制御します。これらの引数を使用してdbupgradeシェル・コマンドを実行することもできます。

> ノート:
>
> シェル・コマンド・ユーティリティ dbupgrade は catctl.pl を開始します。dbupgrade ユーティリティは ORACLE_HOME/bin ディレクトリにあります。コマンド・プロンプトでシェル・コマンド・ユーティリティを使用して、パラレル・アップグレード・ユーティリティを起動できます。デフォルト値を使用してユーティリティを実行するか、catctl.pl

の入力パラメータを使用してパラレル・アップグレード・ユーティリティの引数を指定できます。

表4-1 パラレル・アップグレード・ユーティリティ(catctl.pl)のパラメータ

パラメータ	説明

-c	アップグレードする PDB の空白区切りの包含リストを指定します。たとえば、PDB1、PDB2、PDB3 および PDB4 のある Oracle Multitenant デプロイメントで、PDB1 と PDB2 を含めながら、指定しない PDB を除外できます。PDB1 と PDB2 はアップグレードされますが、PDB3 と PDB4 はアップグレードされません。
	
	Linux および UNIX(一重引用符を使用):
	
	`-c 'PDB1 PDB2'`
	
	Windows (二重引用符を使用):
	
	`-c "PDB1 PDB2"`

-C	アップグレードする PDB の空白区切りの除外リストを指定します。たとえば、PDB1、PDB2、PDB3 および PDB4 のある Oracle Multitenant デプロイメントで、除外リストを使用して PDB1 と PDB2 を除外しながら、指定しない PDB を含めることができます。PDB1 と PDB2 はアップグレードされませんが、PDB3 と PDB4 はアップグレードされます。
	
	Linux および UNIX(一重引用符を使用):
	
	`-C 'PDB1 PDB2'`
	
	Windows (二重引用符を使用):
	
	`-C "PDB1 PDB2"`
	
	ノート: `-c` と `-C` は相互に排他的です。
	
	`-C 'CATCTL_LISTONLY'`は、パラレル・アップグレード・ユーティリティで優先度リストの PDB のみを処理するように指定するオプションです。このオプションを `-L` パラメータとともに使用して、リストを指定します。

-d	処理するファイルが含まれているディレクトリの場所を指定します。
-e	スクリプトの実行中に echo OFF に設定します。デフォルトは echo ON です。
-E	アップグレード・エミュレーションを実行できます。 `-E` パラメータを使用してエミュレーション・モードでパラレル・アップグレード・ユーティリティを実行することで、優先度リストがどのように実行されるかをテストしたり、他のアップグレード・パラメータの選択がアップグレード中にどのように実行されるかをテストできます。たとえば、`-n` および `-N` パラメータを使用して

パラメータ	説明
	行うリソース割当ての選択がどのように実行されるかについて、詳細情報を取得するアップグレード・エミュレーションを実行できます。 アップグレード・エミュレーションを実行するには、パラレル・アップグレード・ユーティリティを実行する前にすべてのアップグレードの準備を完了してから、-E を使用してコマンドを実行します。 -E パラメータ付きでパラレル・アップグレード・ユーティリティを実行し、-L パラメータを使用してコマンドの一部として優先度リストを呼び出すと、ユーティリティによって catctl_priority_run.lst ファイルにアップグレード順序が書き込まれます。このリストは、-l パラメータで指定するファイル・パスに配置されるか、または別の出力ファイル・パスを指定しない場合はデフォルトのログ・ファイル領域に配置されます。
-F	以前のアップグレード・エラーを強制的にクリーンアップします。 非 CDB データベースでは、-F パラメータのみが必要です。CDB では、空白区切りの包含リスト(-c で指定)でこのオプションを使用します。
-i	スプール・ログ・ファイルの作成時に使用する識別子を指定します。
-l	スプール・ログ・ファイルで使用するディレクトリの場所を指定します。 デフォルトの場所は Oracle_base/cfgtoollogs/dbname/upgradedatetime です。date および time 文字列の書式は YYYYMMDDHHMMSC です。YYYY は年、MM は月、DD は日、HH は時間、MM は分、SC は秒を表します。 /admin ディレクトリにはログ・ファイルを書き込まないことを強くお薦めします。
-L	Oracle Database のアップグレード中に優先度リストを使用して PDB をアップグレードし、優先度リスト名を指定します。優先度リストによって、アップグレード中にデータベースの優先度ステータスが更新されます。この優先度リストは、将来のアップグレードでも維持されます。 デフォルトで、CDB$ROOT および PDB$SEED データベースは、常に最初に処理されます。これらは、優先度リストに追加されていなくても、最初に処理されます。優先度リストに含まれるすべての PDB は、優先度リストに含まれない PDB より先に処理されます。
-M	PDB のアップグレード中に CDB$ROOT を UPGRADE モードで維持します。 非 CDB では、このパラメータは無視されます。 CDB のアップグレードでこのパラメータ設定を使用すると、CDB とそのすべての PDB がアップグレード・モードになり、合計アップグレード時間が短縮できます。ただし、CDB とそのすべての PDB がアップ

パラメータ	説明
	グレードされるまで、どの PDB も起動できません。 デフォルトでは、-M パラメータ設定を使用しない場合、CDB$ROOT がアップグレードされて標準モードで再起動され、標準のバックグラウンド・プロセスが起動されます。それぞれの PDB のアップグレード中に、他の PDB のアップグレードが進行中に PDB をオンラインにすることができます。
-n	パラレル操作で使用する処理の数を指定します。 非 CDB の場合: -n パラメータはデータベースをアップグレードするときに使用する SQL プロセスの数を指定します。 マルチテナント・アーキテクチャ・データベース(CDB)の場合: 同時にアップグレードされる PDB の数は -n パラメータの値により制御されます。複数の PDB アップグレードが一緒に処理されます。Oracle Database 12c 以上では、マルチテナント・アーキテクチャ・データベースのデフォルト値は、システムの CPU の数です。24 に等しい cpu_count は、-n のデフォルト値である 24 と同じです。 -n パラメータの値: 非 CDB の場合: -n の最大値は 8 です。最小値は 1 です。デフォルト値は 4 です。 マルチテナント・アーキテクチャ・データベース(CDB)の場合: -n の最大値に制限はありません。最小値は 4 です。同時実行される PDB アップグレードの最大数は、-N の値で割られた-n の値になります。
-N	データベースをアップグレードする時に使用する SQL 処理の数を指定します。 非 CDB では、このパラメータは無視されます。 CDB では、最大値は 8 です。最小値は 1 です。デフォルト値は 2 です。
-p	指定したフェーズから再開します。アップグレードを再実行する場合、正常に完了しているフェーズは再開されません。
-P	指定したフェーズから停止します。
-R	失敗したフェーズからアップグレードを再開します。実行されていないアップグレード・フェーズだけが再実行されるように、-R パラメータを指定することで、失敗した時点からアップグレードを再開できます。
-s	セッションを初期化する SQL スクリプトを指定します。
-S	パラレルではなく、シリアル・アップグレードを指定します。

パラメータ	説明
	Oracle Database 12.2 以降では、-S オプションを使用した catupgrd.sql はサポートされなくなりました。
-T	ユーザー・スキーマベースの表領域をオフラインにします。
-u	ユーザー名を指定し、パスワードを要求します。
-y	フェーズのみを表示します。
-z	catcon.pm の本番デバッグ情報を有効にします。
-Z	catctl.pl のデバッグ・トレース情報を戻します。 たとえば、1 に設定する場合、-Z 1 を入力します。

親トピック: パラレル・アップグレード・ユーティリティを使用した手動アップグレード

パラレル・アップグレード・ユーティリティの使用例

この例を使用して、パラレル・アップグレード・ユーティリティを手動で実行してアップグレードを実行する方法を理解してください。

パラレル・アップグレード・ユーティリティ(catctl.pl)は、AutoUpgradeおよびDBUAと統合されています。catctl.pl Perlスクリプトでは、クラシック・アップグレードを使用してCDB$ROOTをアップグレードします。パラレル・アップグレード・ユーティリティは、コマンドライン・スクリプトdbupgradeを使用して手動で実行することもできます。コマンドライン・パラメータを使用してパラレル・アップグレード・ユーティリティを実行し、アップグレードの実行方法を指定します。たとえば、パラレル操作を使用せずにシリアル・モードでユーティリティを実行するには、-n 1オプションを指定します。

例4-1 CDBおよび非CDBでのパラメータを使用したパラレル・アップグレード・ユーティリティの実行

パラレル・アップグレード・ユーティリティの実行時に-n 4オプションを使用すると、アップグレード処理によってcatupgrd0.log、catupgrd1.log、catupgrd2.logおよびcatupgrd3.logが作成されます。すべてのcatupgrd#.logファイルをチェックして、アップグレードが成功したことを確認してください。アップグレードに失敗し、問題を修正してパラレル・アップグレード・ユーティリティを再実行した場合、以前のログ・ファイルは上書きされます(-lパラメータを使用して別のログ・ディレクトリを指定した場合を除く)。

たとえば:

```
cd $ORACLE_HOME/bin
dbupgrade -n 4 -l $ORACLE_HOME/diagnostics
```

例4-2 パラレル・アップグレード・ユーティリティを使用した複数のプラガブル・データベース(PDB)に対するパラレル・アップグレードの実行

これらの例では、パラメータ設定によって、パラレル・アップグレード・ユーティリティによる複数のPDBに対するアップグレードの実行方法がどのように変化するかを示します。

ノート:

アップグレード計画では、次のことに注意してください。

- CDB$ROOT は、SQL プロセスの最小値が 4 に、最大値が 8 にデフォルトで設定されます
- -N のデフォルト値は 2 です。
- PDB$SEED は、アップグレード・サイクルでは常に 1 つの PDB としてカウントされます
- パラレル・アップグレード・ユーティリティのパラメータ-n のデフォルトは、CPU_COUNT パラメータの値です

次の例のシステムは、CPU_COUNTの値が24であるOracle Multitenant Oracle Databaseシステムです。

パラメータ-nおよび-Nの値を指定せずに(つまり、-Nのデフォルト値2を受け入れ、-nのデフォルト値をCPU_COUNTパラメータの値24として受け入れます)、パラレル・アップグレード・ユーティリティを実行します。次のパラレル処理が発生します。

- 12のPDBがパラレルにアップグレードされます(CPU_COUNTを2で割る)
- 1つのPDBに対して2つのパラレル処理が実行されます

-nの値を64に、-Nを4に指定します。次のパラレル処理が発生します。

- 16のPDBがまとめてアップグレードされます(64を4で割る)
- 1つのPDBに対して4つのパラレル処理が実行されます

-nの値を20に、-Nを2に指定します。次のパラレル処理が発生します。

- 10のPDBがまとめてアップグレードされます(20を2で割る)
- 1つのPDBに対して2つのパラレル処理が実行されます

-nの値を10に、-Nを4に指定します。次のパラレル処理が発生します。

- 2つのPDBがまとめてアップグレードされます(10を4で割って切捨て)。
- 1つのPDBに対して4つのパラレル処理が実行されます

-nの値を指定せずに(つまり、-nのデフォルト値(CPU_COUNTパラメータの値)を受け入れます)、-Nの値を1と指定します。次のパラレル処理が発生します。

- 24のPDBがまとめてアップグレードされます(CPU_COUNT値を1で割る)
- 1つのPDBに対して1つの処理が実行されます

-nの値を20に指定し、-Nの値を指定しません(つまり、-Nのデフォルト値である2を受け入れます)。次のパラレル処理が発生します。

- 10のPDBがまとめてアップグレードされます(20を2で割る)
- 1つのPDBに対して2つのパラレル処理が実行されます

親トピック: パラレル・アップグレード・ユーティリティを使用した手動アップグレード

マルチテナント・アーキテクチャのOracle Databaseの手動アップグレード・シナリオ

手動アップグレードの準備のために、マルチテナント・アーキテクチャでデプロイされたOracle Databaseの手動アップグレード・シナリオおよび手順を参照してください。

Oracle Database 12c以上では、マルチテナント・アーキテクチャによって、プラガブル・データベース(PDB)を含むマルチテナント・コンテナ・データベース(CDB)を使用したOracle Databaseデプロイメントが可能になります。Oracle Database 12cリリース1 (12.1.0.1)より前のすべてのOracle Databaseリリースでは、非CDBアーキテクチャを使用します。

> 注意:
>
> COMPATIBLE 初期化パラメータを 12.1.0.2 以上のリリースに設定した後は、データベースをダウングレードできません。プラガブル・データベース(PDB)のダウングレードは、互換性が 12.1.0.1 に設定されている場合にのみ可能です。それでもダウングレードが制限される可能性があります。
>
> アップグレードまたはダウングレードを開始する前に、ソース・データベースとターゲット・データベースを最新のリリース更新またはリリース更新リビジョンにアップグレードすることをお薦めします。

- Oracle Multitenant Oracle Databaseのアップグレードについて
  マルチテナント・アーキテクチャにインストールされたOracle Databaseをパラレルに、または順番にアップグレードできます。
- マルチテナント・アップグレードを使用したプロキシPDBのアップグレードの調整
  プロキシPDBとPDBターゲットのバージョンが同じになるようにCDBのアップグレードを調整します。
- マルチテナント・コンテナOracle Database (CDB)の手動でのアップグレード
  この項のステップでは、コマンドライン・プロシージャを使用してCDBを手動でアップグレードするためのステップを示します。
- 優先度リスト付きのパラレル・アップグレード・ユーティリティを使用したPDBのアップグレードについて
  Oracle Database 12.2以降では、あるPDBのセットを他のPDBより先にアップグレードするように優先度リストを使用してPDBをアップグレードできます(このアップグレード優先度は変更できます)。
- 優先度リスト、包含リストおよび除外リストを使用したPDBのアップグレードについて
  プラガブル・データベース(PDB)のアップグレード方法を制御するには、優先度リストを含む包含リストおよび除外リストを使用できます。
- マルチテナント・アーキテクチャのパラレル・アップグレード
  この方法を使用して、コンテナ・データベース(CDB)をアップグレードしてから複数のプラガブル・データベース(PDB)をパラレルにアップグレードして、マルチテナント・アーキテクチャOracle Databaseリリース(Oracle Database 12cリリース1 (12.1.0.1)以降)をアップグレードします。
- 切断/接続を使用したマルチテナント・アーキテクチャの順次アップグレード
  以前のリリースのマルチテナント・コンテナ・データベース(CDB) (Oracle Databaseリリース12c (12.1.0.1)以降)に存在するプラガブル・データベース(PDB)をアップグレードするには、以前のリリースのCDBからPDBを切断し、そのPDBを新しいリリースのCDBに接続できます。

関連項目

- Oracle Multitenant管理者ガイド

親トピック: パラレル・アップグレード・ユーティリティを使用したOracle Databaseの手動アップグレード

Oracle Multitenant Oracle Databaseのアップグレードについて

マルチテナント・アーキテクチャにインストールされたOracle Databaseをパラレルに、または順番にアップグレードできます。

Oracle Database 12cリリース1 (12.1)以上では、コンテナ・データベース(CDB)のプラガブル・データベース(PDB)の作成および管理ができるマルチテナント・アーキテクチャが提供されます。マルチテナント・アーキテクチャ・システムをアップグレードするには、Oracle Database Upgrade Assistant (DBUA)を使用するか、またはパラレル・アップグレード・ユーティリティを使用して手動アップグレードを実行します。

マルチテナント・アーキテクチャを使用するOracle Databaseをアップグレードする場合、次の2つの方法があります。

- パラレル。この方法では、CDBをアップグレードする1つのアップグレード操作を実行してから、複数のPDBをパラレルにアップグレードします。
- 順次。この方法では、新しいリリースのCDBをインストールし、PDBを準備して以前のリリースのCDBから切断し、新しいリリースのCDBに接続して、各PDBのアップグレードを完了します。

次の項では、各アップグレード方法の概要について説明します。

Oracle Multitenantのパラレル・アップグレード

パラレルの方法では、パラレル処理と可用性の程度を設定するパラメータを使用して、パラレル・アップグレード・ユーティリティ(`catctl.pl`)で最初に`CDB$ROOT`をアップグレードします。

- `-n`パラメータは、アップグレードを実行するパラレル処理の数を定義します(最大8)。
- `-M`パラメータは、アップグレード全体を通じて`CDB$ROOT`を`UPGRADE`モードのままにするか、CDBのアップグレードの完了後にアクセス可能にするかを指定します。`-M`パラメータを使用してアップグレードを実行しない場合、`CDB$ROOT`のアップグレードが完了すると、PDBは、各PDBのアップグレードが完了すると同時にアクセスできるようになります。`–M`パラメータを使用してアップグレードを実行する場合、`CDB$ROOT`は`UPGRADE`モードのままとなり、PDBは、すべてのPDBのアップグレードが完了するまで使用できません。

Oracle Multitenantの順次アップグレード

順次の方法では、新しいリリースのマルチテナント・アーキテクチャCDBをインストールします。次に、以前のリリースのマルチテナント・アーキテクチャCDBで、SQLコマンドを発行してアップグレード前スクリプトを実行し、アップグレードする1つ以上のPDBを準備してからそれらを停止します。その後、PDBを切断し、それらを新しいリリースのマルチテナント・アーキテクチャCDBに接続して、PDBごとに順番にアップグレードを完了します。

親トピック: マルチテナント・アーキテクチャのOracle Databaseの手動アップグレード・シナリオ

マルチテナント・アップグレードを使用したプロキシPDBのアップグレードの調整

プロキシPDBとPDBターゲットのバージョンが同じになるようにCDBのアップグレードを調整します。

アップグレード時には、プロキシPDBをアップグレードしても、それに対応するターゲットPDBはアップグレードされません。ターゲットPDBのアップグレードは個別に実行する必要があります。

親トピック: マルチテナント・アーキテクチャのOracle Databaseの手動アップグレード・シナリオ

マルチテナント・コンテナOracle Database (CDB)の手動でのアップグレード

この項のステップでは、コマンドライン・プロシージャを使用してCDBを手動でアップグレードするためのステップを示します。

この手順を使用する前に、次のステップを完了しておく必要があります。

- 新しいリリースのOracle Databaseソフトウェアのインストール
- 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

Oracle Database 12cでは、Oracle Databaseをプラガブル・データベースを含むマルチテナント・コンテナ・データベース(CDB)として機能させることができるマルチテナント・アーキテクチャが導入されています。DBUAを使用してCDBをアップグレードできますが、その処理で、CDBに連結されているすべてのプラガブル・データベースを同時にアップグレードします。または、Oracle Databaseのソフトウェアをインストールして新しいOracleホームを準備した後に、手動のコマンドライン・アップグレードに進むことができます。

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。
2. ソース・データベースをバックアップします。
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;
   ```

   CLUSTER_DATABASEパラメータを変更した後にデータベースを再起動します。
5. データベースを停止します。(構文は、非CDBとCDBで同じです。)

   ```
   SQL> SHUTDOWN IMMEDIATE
   ```

   ノート:

   PDB を閉じるには、CDB ルートから次のように指定できます: `alter pluggable database PDBname close`

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`OracleService SID`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。

      自分の`SID`が`ORCL`の場合、`SID`を自分のSIDに置き換えて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。

たとえば:

```
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ホーム・ユーザー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コマンドは、このユーザー・アカウントのパスワードを要求します。ORADIMを使用して、他のオプションを指定できます。

この例で、SIDの値がORCL、password (SYSPWD)の値がTWxy5791、ユーザーの最大数(MAXUSERS)の値が10、およびOracleホーム・パスがC:¥ORACLE¥PRODUCT¥19.1.0¥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¥19.1.0¥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。ログ・ファイルには、マルチテナント・データベース内のPDBの名前が含まれています。

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Oracle Database 12cのOracleホームを指していること。
   c. 次の環境変数がOracle Database 12cの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。

ノート:

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ノードでこれらのチェックを実行してください。

参照:

ご使用のオペレーティング・システムでのその他の重要な環境変数の設定に関する詳細は、Oracle DatabaseおよびOracle Clusterwareのインストレーション・ガイドを参照してください

8. 新しいOracle DatabaseリリースのOracleホームの所有者として、システムにログインします。
9. 新しいOracleホームのパスOracle_home/rdbms/adminディレクトリからSQL*Plusを起動します。

たとえば:

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

Windowsプラットフォームでは、SQL*Plusにアクセスするには、ディレクトリを%ORACLE_HOME%/binに変更します

10. 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
```

11. アップグレード・モードでCDBを開始します。

```
SQL> startup upgrade
```

12. SQL*Plusで次のコマンドを発行して、インスタンスを起動します。

```
SQL> alter pluggable database all open 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ノートにとり、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。

ノート:

UPGRADE モードでデータベースを起動すると、次のことが実行されます。

- 新しいバージョンの Oracle Database インスタンスを使用してデータベースが起動されます
- ログインが SYSDBA に限定されます
- システム・トリガーが無効になります
- アップグレード用にデータベースを準備する追加操作が実行されます

13. SQL*Plusを終了したら、次のステップに進みます。

たとえば:

```
SQL> EXIT
```

14. CDB全体をアップグレードするには、パラレル・アップグレード・ユーティリティ(catctl.pl)を新しいOracleホームから実行します。パラレル・アップグレード・ユーティリティでは、停止時間を短縮するパラレル・アップグレード・オプションが提供されます。コマンドライン・スクリプトdbupgradeを使用して、新しいOracleホームからコマンドを実行できます。

たとえば:

Linuxの場合:

```
cd $ORACLE_HOME/bin
./dbupgrade
```

Windowsの場合:

```
cd %ORACLE_HOME%¥bin
dbupgrade
```

ノート:

- -d オプションを使用して、処理対象のファイルが含まれるディレクトリを指定します。-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。
- CDB 全体をアップグレードしており、CDB$ROOT にエラーがある場合、アップグレードは強制終了されます。

15. CDB内のPDBのサブセットをアップグレードするために、包含リストまたは除外リストのいずれかを指定します。

a. LinuxまたはUNIXシステム用の次の例では、包含リストを使用してPDB1のみをアップグレードします。

```
cd $ORACLE_HOME/bin
./dbupgrade -c 'PDB1'
```

b. Windowsシステム用の次の例では、除外リストを使用してPDB1を除くCDB内のすべてをアップグレードします。

```
cd $ORACLE_HOME¥bin
dbupgrade -C "PDB1"
```

ノート:

個別の PDB は、以前のリリースの CDB から切断し、それよりも新しいリリースの CDB に接続することによりアップグレードできます。

Windows の場合、包含オプション(-c)または除外オプション(-C)を指定して dbupgrade コマンドを実行するときに、CDB ルート名および PDB シード名を引用符で囲んでオプションを指定する必要があります。

たとえば:

```
... -C "CDB$ROOT PDB$SEED"
```

16. CDBでは、SYSDBAとしてCDBにログインし、コマンドalter pluggable database all openを実行してデータベースでコードを再コンパイルできるようにします。たとえば:

```
$ sqlplus / as sysdba
SQL> alter pluggable database all open;
```

17. ログ・ファイル・ベース(-b)コマンドを使用してcatcon.plを実行し、ログ・ファイル・ベースをutlrpに設定します。このコマンドはutlrp.sqlを起動し、残っているストアドPL/SQLおよびJavaコードを再コンパイルします。

たとえば:

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

コマンドで-b utlrpが指定されているため、再コンパイルの結果はログ・ファイルutlrp0.logに格納されます。

18. catconユーティリティを使用してpostupgrade_fixups.sqlを実行し、ログ・ファイル・ベースをpostupgrade_fixupsに設定します。

たとえば:

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

19. catconユーティリティを使用してutlusts.sqlを実行し、ログ・ファイル・ベースをutlu19sに設定します。このコマンドは、すべての問題が修正されたことを検証します。

    たとえば、CDBでは次のようになります。

    ```
    $ORACLE_HOME/perl/bin/perl catcon.pl -n 1 -e -b utlu19s -d '''.''' utlusts.sql
    ```

    コマンドで-b utlu19sが指定されているため、アップグレードの結果はログ・ファイルutlu19s0.logに格納されます。

    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、VALIDとしてマークされていないコンポーネントを表示した場合や、実行したSQLスクリプト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。

20. 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・ゾーンを更新することもできます。タイムゾーンを更新する場合は、CDB$ROOTとPDBの両方でタイムゾーンを更新する必要があります。

21. SQL*Plusを終了します。

    たとえば:

    ```
    SQL> EXIT
    ```

22. (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
    ```

    この例で、db-unique-nameは(インスタンス名ではなく)割り当てられた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ていた短いCLIオプションに加えて、GNUスタイルの長いオプションもサポートされます。

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。

参照:

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

関連項目

- Oracle Database管理者ガイド

親トピック: マルチテナント・アーキテクチャのOracle Databaseの手動アップグレード・シナリオ

優先度リスト付きのパラレル・アップグレード・ユーティリティを使用したPDBのアップグレードについて

Oracle Database 12.2以降では、あるPDBのセットを他のPDBより先にアップグレードするように優先度リストを使用してPDBをアップグレードできます(このアップグレード優先度は変更できます)。

優先度リストにより、その優先度に従ってPDBをグループ化してアップグレードできます。優先度リストは、アップグレード優先度の順序(数値の優先度グループ別のPDB)を定義するカンマ区切りリストが含まれたテキスト・ファイルです。優先度リストを使用してアップグレードを実行し、アップグレード実行時にそのリストを呼び出すには、`-L`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(`dbupgrade`、`dbupgrade.cmd`または`catctl.pl`)を実行します。

次の書式を使用してリストを作成します。この書式例で、変数numeralは数値で、pdbxはPDBの名前です。

```
Number, Pdb
   numeral,pdb1,pdb2,pdb3
   numeral,pdb4
   numeral,pdb5,pdb6,pdb7,pdb8
   .
   .
   .
```

数値はPDBの優先度を表します。

PDBの優先度は次のとおりです。

1. CDB$ROOT: 優先度1。コンテナ・データベースを最初にアップグレードすることは、必須の優先度です。コンテナ・データベースのアップグレードの優先度を変更することはできません。CDB$ROOTは、常に最初に処理されます。
2. PDB$SEED: 優先度1。PDBシード・データベースをアップグレードすることは、必須の優先度です。PDBシードのアップグレードの優先度を変更することはできません。PDB$SEEDは、常にCDB$ROOTの後に、PDBアップグレードの最初のバッチとともにアップグレードされます。
3. 優先度リスト1のPDB: 優先度1は、ユーザー選択による最高の優先度です。これらのPDBは、CDB$ROOTの後で、PDB$SEED PDBがアップグレードされるバッチで2番目にアップグレードされます。
4. 優先度リスト2のPDB: 優先度2は、2番目に高い優先度のPDBセットです。これらのPDBは、優先度1のPDBの後にアップグレードされます。
5. 優先度リスト3のPDB: 優先度3は、3番目に高い優先度のPDBセットです。これらのPDBは、優先度2のPDBの後にアップグレードされます。
6. 優先度リスト4のPDB: 優先度4は、4番目に高い優先度のPDBセットです。これらのPDBは、優先度3のPDBの後にアップグレードされます。
7. 優先度リスト5のPDB: 優先度5は、5番目に高い優先度のPDBセットです。これらのPDBは、優先度4のPDBの後

にアップグレードされます。

8. 優先度リスト6のPDB: 優先度6は、6番目に高い優先度のPDBセットです。これらのPDBは、優先度5のPDBの後にアップグレードされます。

パラレル・アップグレード・ユーティリティを実行する場合、次の処理ルールが適用されます。

- CDB$ROOTおよびPDB$SEEDは、優先度リストに存在しない場合でも、常に最初に処理されます。
- 優先度リストに含まれるすべてのPDBは、優先度の順序で処理されます
- 優先度リストに指定されていないPDBは、優先度リストに指定されているPDBの後に処理されます。

たとえば:

```
Number,Pdb
1,sales1,region2,receivables1
2,sales2
3,dss1,region3,region2,dss2,dss3
```

優先度リストを使用してパラレル・アップグレード・ユーティリティを実行するには、次の構文を使用します。

```
dbupgrade -L priority_list_name
```

たとえば、パラレル・アップグレード・ユーティリティのバッチ・コマンドと`My122Upgrade`という名前の優先度リストを使用して、Windowsシステムでパラレル・アップグレード・ユーティリティを実行するには、次のコマンドを入力します。

```
C:>¥u01¥app¥19.1.0¥db_home1¥rdbms¥admin¥dbupgrade -L MyUpgrade
```

アップグレード優先度を設定する優先度リストを使用したアップグレードを完了した後、それらのPDB優先度の状態は、各PDB用にCDBで維持されます。次回のアップグレードでは、以前のアップグレードでPDB用に設定された優先度が考慮されます。

PDBのアップグレード優先度の状態を変更するには、次のSQLコマンド構文を使用します(`PDBName`は変更するアップグレード優先度が設定されたPDBの名前で、`PDBPriorityNumber`は割り当てる新しい優先度の値です)。

```
SQL> alter session set container = CDB$ROOT
SQL> alter pluggable database PDBName upgrade priorityPDBPriorityNumber
```

たとえば:

```
SQL> alter session set container = CDB$ROOT
SQL> alter pluggable database region2 upgrade priority 2
```

この例によって、前の例でアップグレード優先度1に設定されたregion 2という名前のPDBは、アップグレード優先度2に変更されます。

親トピック: マルチテナント・アーキテクチャのOracle Databaseの手動アップグレード・シナリオ

優先度リスト、包含リストおよび除外リストを使用したPDBのアップグレードについて

プラガブル・データベース(PDB)のアップグレード方法を制御するには、優先度リストを含む包含リストおよび除外リストを使用できます。

Oracle Database 12cリリース1 (12.1.0.2)以降のリリースでは、アップグレードがPDBに適用される順序を指定するために適した方法は、`alter pluggable database`を`upgrade priority`とともに使用してソースOracle Databaseで優先度を設定することです(`pdbname`はPDB、`number`はアップグレード用に割り当てる優先度)。

```
alter pluggable database pdbname upgrade priority number
```

たとえば:

```
alter pluggable database CDB1_PDB3 upgrade priority 2
```

ただし、ここで説明されている手順を使用して、アップグレード時に優先度リストを指定することもできます。

アップグレード処理およびリスト

次の用語は、アップグレードのリスト処理のタイプを示します。

- **優先度リスト**: リスト内のPDBのアップグレード優先度を指定するカンマ区切りリスト。
- **包含リスト**: アップグレードするPDBを指定するカンマ区切りリスト。これらのリストのPDBは、優先度リストに指定されたPDBの後にアップグレードされます。
- **除外リスト**: アップグレードしないPDBを指定するカンマ区切りリスト。

包含リストと除外リストは、次のように使用できます。

- アップグレードで独自にPDBのセットを含めたり除外することができます
- 優先度リストと組み合せて、PDBのアップグレード順やアップグレードから除外するPDBを詳細に指定できます。

  包含リストを優先度リストと組み合せて使用すると、包含リストに指定されたPDBは、優先度リストで割り当てられている優先度の値に従ってアップグレードされます。包含リストに指定されているが、優先度リストに指定されていないPDBは、優先度リストのすべてのPDBがアップグレードされた後にアップグレードされます。

  除外リストを優先度リストと組み合せて使用すると、除外リストに指定されたPDBは、アップグレードされません。

ノート:

Linux や Unix の vi や Microsoft Windows のメモ帳などのプレーン・テキスト・エディタを使用して優先度リストを作成します。

このトピックの例では、cpu_count の値は 2 です。

デフォルト処理を使用するアップグレード優先度

デフォルト処理は、リストを使用してコンテナ・データベース(CDB)内のPDBをアップグレードする方法を指定しない場合に実行されるアップグレード処理です。

デフォルト処理では、CDB$ROOTがアップグレードされ、次にPDB$SEEDがアップグレードされます。設定した並列度に応じて、PDB$SEEDでは1つ以上のPDBを並行で更新できます。アップグレードが完了し、アップグレード・プロセッサが使用可能になると同時にPDBがアップグレードされます。

後続の例では、CDBおよびPDBで次のマルチテナント構成を使用します。

```
CDB$ROOT
PDB$SEED
CDB1_PDB1
CDB1_PDB2
CDB1_PDB3
CDB1_PDB4
CDB1_PDB5
```

デフォルト処理では、アップグレードするPDBやアップグレードから除外するPDBのプリファレンスは指定しません。デフォルト処理では、CDB$ROOTが最初にアップグレードされ、アップグレードされるPDBの最初のグループでPDB$SEEDが更新されます。

例4-3 完全なPDBアップグレード優先度の指定

次の優先度リストの例では、すべてのPDBの優先度がリストで設定されます。

```
1,CDB$ROOT
1,PDB$SEED
1,CDB1_PDB1
1,CDB1_PDB2
2,CDB1_PDB3
2,CDB1_PDB4
3,CDB1_PDB5
```

次に、優先度順にPDBをグループ化する同じリストの別の記述方法を示します。

```
1,CDB$ROOT
1,PDB$SEED
1,CDB1_PDB1,CDB1_PDB2
2,CDB1_PDB3,CDB1_PDB4
3,CDB1_PDB5
```

前述の例で、優先度1にリストされているPDBは、CDB1_PDB1およびCDB1_PDB2です。これらのPDBは、CDB1_PDB3およびCDB1_PDB4より先にアップグレードされます。

次に、コンテナIDの値(CON_ID)を使用して優先度順を設定する、同じリストの別の記述方法を示します。

```
1,CDB$ROOT
1,PDB$SEED
1,3,4
2,5,6
3,7
```

前述の例で、優先度1にリストされているPDBは、CDB1_PDB1 (CON_ID 3で指定)およびCDB1_PDB2 (CON_ID 4で指定)です。これらのPDBは、CDB1_PDB3 (CON_ID 5)およびCDB1_PDB4 (CON_ID 6)より先にアップグレードされます。

CON_IDメソッドを使用して優先度を指定する場合、最初の数値で、PDBのグループの優先度を指定します。2番目の数値で、その優先度グループに含まれるPDB (CON_ID)の数値を指定します。CDB$ROOTは、常に最初に更新され、PDB$SEEDは、常に最初のアップグレード優先度グループで更新されます。

これらの例は、すべて次の特性を持つ優先度リストのアップグレードを示します。

- 除外処理: なし
- 包含処理: なし
- デフォルト処理: なし

アップグレード順序は、次の順番で実行されます。

1. CDB$ROOT
2. PDB$SEED、CDB1_PDB1
3. CDB1_PDB2、CDB1_PDB3
4. CDB1_PDB4、CDB1_PDB5

例4-4 PDBの優先度サブセットの指定およびデフォルト処理による他のPDBのアップグレード

次の例では、アップグレード用にPDBのサブセットを指定するpriority.lstという優先度リストを指定します。

```
catctl -L priority.lst catupgrd.sql
1,CDB$ROOT
1,PDB$SEED
1,CDB1_PDB1,CDB1_PDB2
```

この例は、次の特性を持つ優先度リストのアップグレードを示します。

- 除外処理: なし
- 包含処理: なし
- デフォルト処理: CDB1_PDB3、CDB1_PDB4、CDB1_PDB5

アップグレード順序は、次の順番で実行されます。

1. CDB$ROOT
2. PDB$SEED、CDB1_PDB1
3. CDB1_PDB2、CDB1_PDB3
4. CDB1_PDB4、CDB1_PDB5

例4-5 PDBの優先度サブセットの指定および包含リストによる他のPDBのアップグレード

次の例では、アップグレード用にPDBの優先度サブセットを指定するpriority.lstという優先度リストを指定します:

```
catctl -L priority.lst -c 'CDB1_PDB2 CDB1_PDB4 CDB1_PDB5' catupgrd.sql
```

このコマンドは、次の優先度リストを参照します。

```
1,CDB$ROOT
1,PDB$SEED
1,CDB1_PDB2,CDB1_PDB4
2.CDB1_PDB5
```

この例は、次の特性を持つ優先度リストのアップグレードを示します。

- 除外処理: なし
- 包含処理: CDB1_PDB2、CDB1_PDB4、CDB1_PDB5
- デフォルト処理: なし

アップグレード順序は、次の順番で実行されます。

1. CDB1_PDB2、CDB1_PDB4
2. CDB1_PDB5

パラレル・アップグレード・ユーティリティは包含リストにあるPDBだけを、優先度リストの順に処理します。

例4-6 PDBの優先度サブセットの指定および除外リストによるCDB$ROOTの除外

次の例では、priority.lstという優先度リストを使用してcatctlを実行します。このコマンドを-Cオプション付きで実行するため、アップグレードからCDB$ROOTが除外されます。

```
catctl -L priority.lst -C 'CDB$ROOT' catupgrd.sql
```

優先度リストは次のとおりです。

```
1,CDB$ROOT
```

```
1,PDB$SEED
1,CDB1_PDB1,CDB1_PDB2
2,CDB1_PDB3,CDB1_PDB4
3,CDB1_PDB5
```

アップグレードは、アップグレード優先度を指定する優先度リストを使用して処理されます。

- 包含処理: なし
- 除外処理: CDB$ROOT
- 優先度処理: PDB$SEED、CDB1_PDB1、CDB1_PDB2、CDB1_PDB3、CDB1_PDB4、CDB1_PDB5

CDB$ROOTは除外されるため、優先度処理はシフトします。アップグレード順序は、次の順番で実行されます。

1. PDB$SEED、CDB_PDB1
2. CDB_PDB2、CDB_PDB3
3. CDB1_PDB4、CDB1_PDB5

例4-7 CATCTL_LISTONLYを使用した除外リストの指定

次の例では、アップグレード用にPDBのサブセットを指定するpriority.lstという優先度リストを指定します。CATCTL_LISTONLYオプションを指定すると、優先度リストに含まれないPDBはアップグレードから除外されます。

```
catctl -L priority.lst -C 'CATCTL_LISTONLY' catupgrd.sql
```

優先度リスト:

```
1,CDB$ROOT
1,PDB$SEED
1,CDB1_PDB1,CDB1_PDB2
2,CDB1_PDB3
3,CDB1_PDB5
```

- 除外処理: CATCTL_LISTONLY (包含優先度リストのみを処理)
- 包含処理: なし
- デフォルト処理: なし

アップグレード順序は、次の順番で実行されます。

1. CDB$ROOT
2. PDB$SEED、CDB1_PDB1、CDB1_PDB2
3. CDB1_PDB3、CDB1_PDB5

ノート:

除外リストにキーワード CATCTL_LISTONLY を指定することで、優先度リストは包含優先度リストになります。リストの PDB のみが処理されます。このシナリオではデフォルト処理は発生しないため、この例では CDB1_PDB4 は処理されません。

例4-8 CON_ID値を使用した優先度リストの指定

次の例では、アップグレード用にPDBのサブセットを指定するpriority.lstという優先度リストを指定します。

```
catctl -L priority.lst -C 'CATCTL_LISTONLY' catupgrd.sql
```

アップグレード順序は、優先度リストの優先度値によって決まります。-Lパラメータによってコールされるリスト(priority.lst)で、アップグレード優先度値に続く数値は、PDB名に関連付けられたCON_ID値です。

```
1,3,4
2,5,CDB1_PDB4
3,7
```

前述のリスト例で、CON_ID値とPDB名を混在させて使用できることに注意してください。

優先度1にリストされているPDBは、CDB1_PDB1 (CON_ID 3で指定)およびCDB1_PDB2 (CON_ID 4で指定)です。これらのPDBは、CDB1_PDB3 (CON_ID 5)、CDB1_PDB4 (名前で指定)およびCDB1_PDB5 (CON_ID 7)より先にアップグレードされます。

- 除外処理: -C CATCTL_LISTONLY (包含優先度リストのPDBのみを処理)
- 除外処理: なし
- 包含処理: priority.lstで指定
- デフォルト処理: CDB$ROOT、PDB$SEED

アップグレード順序は、PDBに関連付けられたCON_ID番号を使用する優先度リストによって決まります。

1. CDB$ROOT
2. PDB$SEED、CDB1_PDB1
3. CDB1_PDB2、CDB1_PDB3
4. CDB1_PDB4、CDB1_PDB5

ノート:

この例は、CON_ID メソッドを使用して PDB を指定する方法を示しており、優先度リストから CDB$ROOT および PDB$SEED を省略しています。CDB$ROOT および PDB$SEED は、デフォルト処理を使用して処理されます。

親トピック: マルチテナント・アーキテクチャのOracle Databaseの手動アップグレード・シナリオ

マルチテナント・アーキテクチャのパラレル・アップグレード

この方法を使用して、コンテナ・データベース(CDB)をアップグレードしてから複数のプラガブル・データベース(PDB)をパラレルにアップグレードして、マルチテナント・アーキテクチャOracle Databaseリリース(Oracle Database 12cリリース1 (12.1.0.1)以降)をアップグレードします。

- プラガブル・データベース(PDB)のパラレル・アップグレードについて
  パラレルの方法を使用して、コンテナ・データベース(CDB)をアップグレードしてからすぐにパラレルSQLプロセッサを使用してPDBをアップグレードできます。
- マルチテナント・コンテナ・データベースのパラレル・アップグレード
  この方法を使用して、1回のアップグレード操作でCDB$ROOT、PDB$SEEDおよびCDB内のすべてのPDBをアップグレードします。

親トピック: マルチテナント・アーキテクチャのOracle Databaseの手動アップグレード・シナリオ

プラガブル・データベース(PDB)のパラレル・アップグレードについて

パラレルの方法を使用して、コンテナ・データベース(CDB)をアップグレードしてからすぐにパラレルSQLプロセッサを使用してPDBをアップグレードできます。

CDBには、0 (ゼロ)、1またはそれ以上のPDBを含めることができます。デフォルトでは、パラレル・アップグレード・ユーティリティ(`catctl.pl`)により、CDBおよび同じアップグレード・ウィンドウ内のそのすべてのPDBが更新されます。パラレル・アップグレード・ユーティリティでは、コンピュータ処理装置(CPU)の数を使用して、同時にアップグレードするPDBの最大数を決定します。リプレイ・アップグレードを使用したアップグレードの場合(デフォルト)、パラレルにアップグレードされるPDBの数は、パラレルSQLプロセス数(`-n`オプション)をパラレルPDB SQLプロセス数(`-N`オプション)で割って2で割ることで決定されます。クラシック・アップグレードの場合、同時に実行されるPDBアップグレードの最大数は、パラレルSQLプロセス数(`-n`オプション)をパラレルPDB SQLプロセス数(`-N`オプション)で割った、パラレルにアップグレードされるPDBの数です。

ノート:

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

プラガブル・データベースのアップグレード構文

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

- `-M`では、`CDB$ROOT`をアップグレード・モードで維持するかどうか、またはそれをアップグレードの完了時に使用可能にするかどうかを指定します。
  - `-M`パラメータを使用してパラレル・アップグレード・ユーティリティを実行する場合、アップグレードでは、`CDB$ROOT`とそのすべてのPDBがアップグレード・モードになり、合計アップグレード時間を短縮できます。ただし、CDBとそのすべてのPDBがアップグレードされるまで、どのPDBも起動できません。
  - `-M`パラメータを使用してパラレル・アップグレード・ユーティリティを実行しない場合、CDB$ROOTがアップグレードされて標準モードで再起動され、標準のバックグラウンド・プロセスが起動されます。アップグレードが成功した後、`CDB$ROOT`のみが読取り/書込みモードで開かれます。PDBはすべて`MOUNT`モードのままになります。それぞれのPDBのアップグレード中に、他のPDBのアップグレードが進行中でも各PDBをオンラインにできます。
- `-n`では、PDBのパラレル・アップグレード・プロセッサの数を指定します。

  `-n`の値を指定しない場合、`-n`のデフォルトは`CPU_COUNT`値になります。

  `-n`の値を指定する場合、その値を使用してパラレルSQLプロセスの数が決定されます。最大値の制限はありません。最小値は4です。
- `-N`では、PDBのアップグレード時に使用するSQLプロセッサの数を指定します。最大値は8です。最小値は1です。`-N`の値を指定しない場合、デフォルト値は2になります。

  クラシック・アップグレードの場合、同時に実行されるPDBアップグレードの最大数は、クラシック・アップグレードの場合は`-n`の値で割られた`-N`の値になり、リプレイ・アップグレードのデフォルトの場合は-nで割られた値を2で割った値になります。
- `-t`リプレイ・アップグレードのデフォルトを使用するかわりに、AutoUpgradeを使用してクラシック・アップグレードを使用することを指定します。

次に、PDBのパラレル・アップグレードの方法で使用する手順の概要を示します。

1. バックアップ計画が完了していることを確認します。
2. preupgrade句を使用してAutoUpgradeを実行し、アップグレードを開始する前に修正する必要がある問題があるかどうかを確認します。レポートされた問題をすべて修正します。

   たとえば:

   ```
   java -jar autoupgrade.jar -preupgrade
   "dir=/tmp,oh=/u01/app/product/12.2.0/dbhome_1,sid=db122,target_version=21" -
   mode analyze
   ```

3. AutoUpgradeユーティリティを実行します。次のアップグレードが順番に実行されます。
   a. サイクル1: CDB$ROOTが新しいOracleリリースにアップグレードされます
   b. サイクル2からサイクルx: -nを使用して指定したパラメータ設定により決定されるアップグレードのサイクル数で、x: PDB$SEEDおよびPDBがパラレルにアップグレードされます。
4. アップグレード後のステップを完了します。

例4-9 デフォルトを使用したマルチテナント・アーキテクチャのアップグレードの例(パラメータ・セットなし)

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

結果:

リプレイ・アップグレードの場合、6つのPDBがパラレルにアップグレードされます([CPU_COUNTを2で割る]、2で割る、または12を2で割る)。PDBごとに2個のパラレルSQLプロセスが割り当てられます。

従来のアップグレードの場合、12のPDBがパラレルにアップグレードされます(CPU_COUNTを2で割るか、24を2で割る)。PDBごとに2個のパラレルSQLプロセスが割り当てられます。

例4-10 64個のPDBのパラレル・アップグレード・プロセッサおよび4個のパラレルSQLプロセスを使用したマルチテナント・アーキテクチャのアップグレードの例

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

結果:

リプレイ・アップグレードの場合、8つのPDBがパラレルにアップグレードされます([64を4で割る]/2)。PDBごとに4個のパラレルSQLプロセスが存在します。

従来のアップグレードの場合、16のPDBがパラレルにアップグレードされます(64を4で割る)。PDBごとに4個のパラレルSQLプロセスが存在します。

例4-11 20個のPDBのパラレル・アップグレード・プロセッサおよび2個のパラレルSQLプロセスを使用したマルチテナント・アーキテクチャのアップグレードの例

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

結果:

リプレイ・アップグレードの場合、5つのPDBがパラレルにアップグレードされます([20を2で割る]、2で割る)。PDBごとに2個のパ

ラレルSQLプロセスが存在します。

クラシック・アップグレードの場合、10のPDBがパラレルにアップグレードされます(20を2で割る)。PDBごとに2つのパラレルSQLプロセスがあります。

例4-12 10個のPDBのパラレル・アップグレード・プロセッサおよび4個のパラレルSQLプロセスを使用したマルチテナント・アーキテクチャのアップグレードの例

このシナリオでは、オプション`-n 10`を指定して、パラレルPDBアップグレード・プロセッサの値を10に設定します。オプション`-N 4`を指定して、パラレルSQLプロセッサの値を2に指定します。

結果:

リプレイ・アップグレードの場合、1つのPDBがアップグレードされます([10を4で割る]、2で割る)。PDBには4つのパラレルSQLプロセスがあります。

クラシック・アップグレードの場合、2つのPDBがパラレルにアップグレードされます(10を4で割る)。PDBごとに4個のパラレルSQLプロセスが存在します。

親トピック: マルチテナント・アーキテクチャのパラレル・アップグレード

マルチテナント・コンテナ・データベースのパラレル・アップグレード

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

AutoUpgradeユーティリティを使用してアップグレードを完了しない場合、またはリプレイ・アップグレードを使用する場合は、停止時間をスケジュールできる場合は、このアプローチを使用することをお薦めします。この手順を使用すると、サーバーの使用可能なプロセッサ(CPU)に応じてマルチテナント・アーキテクチャ・コンテナ・データベースのすべてのPDBがパラレルにアップグレードされます。これはアップグレードの直接的な手順で、メンテナンスが簡単になります。

ノート:

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

注意:

- 構成の変更を開始する前に、既存のデータベースのバックアップを必ず作成してください。
- compatible 初期化パラメータを設定した後は、データベースをダウングレードできません。
- アップグレードを開始する前には、ソースおよびターゲット・データベースを最新のリリース更新(RU)またはリリース更新リビジョン(RUR)にアップグレードし、ダウングレードを開始する前には最新のリリース更新にアップグレードすることをお薦めします。

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

   たとえば:

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

3. アップグレードの準備状況を確認するには、preupgradeパラメータを使用してAutoUpgradeを実行し、dirオプションを使用して出力ログ・ディレクトリを指定します。

```
java -jar autoupgrade.jar -preupgrade
"dir=/tmp,oh=/u01/app/product/12.2.0/dbhome_1,sid=db122,target_version=21" -
mode analyze
```

4. ログ・ディレクトリのupgrade.xmlファイルを確認します。

   マルチテナント・アーキテクチャのOracleデータベースでは、fixupsモードでpreupgradeパラメータを使用してAutoUpgradeを実行すると、AutoUpgradeの実行時に開いていたすべてのコンテナで修正が実行されます。これらのスクリプトによって、AutoUpgradeで識別されるいくつかの問題が解決されます。

   upgrade.xmlファイルで特定されたその他のアップグレード前タスクを完了します。

5. (条件付き) Oracle RACデータベースの場合は、SQL*Plusを使用してクラスタ・データベース初期化パラメータをFALSEに設定します。

   たとえば:

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

6. srvctlを使用して、古いOracleホームのデータベースを停止します。

   たとえば、db_unique_nameはデータベース名です。

```
cd $ORACLE_HOME
pwd
/u01/app/oracle/19.0.0/dbhome_1
srvctl stop database -d db_unique_name
srvctl disable database -d db_unique_name
```

7. PFILEまたはSPFILEを古いOracleホームから新しいOracleホームにコピーします
8. SQL*Plusと接続:

```
sqlplus / as sysdba
```

9. CDB$ROOTインスタンスをアップグレード・モードに移行します。

```
STARTUP UPGRADE
```

10. すべてのPDBをアップグレード・モードに移行します。

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

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

```
SHOW PDBS
```

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

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

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

13. パラレル・アップグレード・ユーティリティ(catctl.pl、シェル・コマンドdbupgradeを使用)を実行し、アップ

グレードを開始します(-dでディレクトリの場所を指定します)。

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

Oracle Database 21c以降、デフォルトでは、dbupgradeスクリプトはリプレイ・アップグレードをコールします。これにより、並列度はCPUの数を4で割った数に設定されます。パラレル・アップグレードされるPDBの数は、常にレガシー・アップグレードで以前に使用された値の半分です。64個のCPUを持つサーバーでは、64を4で割った数に等しい16個のPDBがパラレルにアップグレードされます。

> ノート:
>
> クラシック・アップグレードを使用してアップグレードを実行する場合は、-t オプションを使用して dbupgrade を起動します。たとえば:
>
> ```
> dbupgrade -t -d $ORACLE_HOME/rdbms/admin
> ```
>
> AutoUpgrade を使用する場合は、AutoUpgrade スクリプトの指示を参照してください。

14. upg_summary.logを確認してアップグレードが成功したことを確認します。必要に応じて他のログを確認します。
15. SQL*Plusを使用してすべてのPDBをオープンし、データベースを再コンパイルできるようにします。

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

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

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

17. preupgradeパラメータを指定してAutoUpgradeをpostfixupsモードで実行します。AutoUpgradeはすべてのデータベース・チェックを実行し、その結果に基づいて自動的に修正を実行します。

たとえば:

```
java -jar autoupgrade.jar -preupgrade
"dir=/tmp,oh=u01/app/product/12.2.0/dbhome_1,sid=db122,target_home=/databases/p
roduct/19c/dbhome_1" -mode postfixups
```

18. utlusts.sqlを実行して、アップグレードに関する問題がないことを確認します。

たとえば、catconユーティリティを使用してutlusts.sqlを実行するCDBでは、フラグ-bを使用してアップグレードのログ・ファイル・ベースをutlu19sに設定します。

```
$ORACLE_HOME/perl/bin/perl catcon.pl -n 1 -e -b utlu19s -d '''.''' utlusts.sql
```

コマンドで-b utlu19sが指定されているため、アップグレードの結果はログ・ファイルutlu19s0.logに格納されます。

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

19. (条件付き) Oracle RAC環境の場合のみ、次のコマンドを入力して、CLUSTER_DATABASEの初期化パラメータ値をTRUEに設定します。

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

20. Oracle Databaseを起動します。dbnameはデータベースの名前です。

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

21. SQL*Plusを使用してデータベースを停止します。

```
SHUTDOWN IMMEDIATE
```

22. srvctlを使用して、データベースを起動します。

```
srvctl start database -db db-unique-name
```

これで、データベースがアップグレードされました。

注意:

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

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

関連項目

- データファイルおよび一時ファイルの管理

親トピック: マルチテナント・アーキテクチャのパラレル・アップグレード

切断/接続を使用したマルチテナント・アーキテクチャの順次アップグレード

以前のリリースのマルチテナント・コンテナ・データベース(CDB) (Oracle Databaseリリース12c (12.1.0.1)以降)に存在するプラガブル・データベース(PDB)をアップグレードするには、以前のリリースのCDBからPDBを切断し、PDBを新しいリリースのCDBに接続できます。

- プラガブル・データベース(PDB)の順次アップグレードの概要
  パラレル・アップグレード・ユーティリティを使用してPDBを手動でアップグレードするには、以前のリリースのCDBからPDBを切断し、そのPDBを新しいリリースのCDBに接続してから新しいリリースにアップグレードします。
- 以前のリリースのPDBの以前のリリースのCDBからの切断
  PDBのアップグレードを準備するには、この手順を使用して以前のリリースのCDBからPDBを切断します。
- 以前のリリースのPDBの新しいリリースのCDBへの接続
  以前のリリースのCDBから新しいリリースのCDBにPDBを接続するには、CREATE PLUGGABLE DATABASEコマンドを使用します。
- 以前のリリースのPDBの新しいリリースへのアップグレード

UPGRADEモードでのPDBのオープンでは、パラレル・アップグレード・ユーティリティを使用して、以前のリリースのPDBをCDBのリリース・レベルにアップグレードします。

- PDBのアップグレードに包含リストまたは除外リストを使用する
  以前のリリースのPDBのサブセットをアップグレードする場合は、包含リストまたは除外リストを使用して、新しいリリース・レベルのCDBまたはPDBを再度アップグレードしないようにします。

親トピック: マルチテナント・アーキテクチャのOracle Databaseの手動アップグレード・シナリオ

プラガブル・データベース(PDB)の順次アップグレードについて

パラレル・アップグレード・ユーティリティを使用してPDBを手動でアップグレードするには、以前のリリースのCDBからPDBを切断し、そのPDBを新しいリリースのCDBに接続してから新しいリリースにアップグレードします。

PDBのアップグレードに使用できるオプションが複数あります。CDBには、0 (ゼロ)、1またはそれ以上のプラガブル・データベース(PDB)を含めることができます。新しいOracle Databaseリリースをインストールするか、またはCDB (CDB$R00T)をアップグレードした後に、CDB上のすべてのPDBをアップグレードせずに1つ以上のPDBをアップグレードできます。

サービス・デリバリのニーズを満たすアップグレード計画を選択できます。たとえば、AutoUpgradeユーティリティを使用してPDBをアップグレードしたり、手動のパラレル・アップグレード・ユーティリティを使用してPDBを個別にアップグレードしたり、包含リストまたは除外リストを使用してアップグレードすることができます。CDBとすべてのPDBをアップグレードすることも(パラレル手動アップグレード)、CDBをアップグレードしてから包含リストや除外リストを使用して個別にまたはセットでPDBを順次アップグレードすることもできます。データベース・アップグレード・ユーティリティ(DBUA)も引き続き使用できます。ただし、Oracle Databaseのアップグレードには、AutoUpgradeユーティリティを使用することをお薦めします。

パラレル・アップグレード・ユーティリティを使用してアップグレードを実行し、手動の切断/接続アップグレードを実行することにした場合、パラレル・アップグレード・ユーティリティを使用したPDBの順次アップグレードに必要なステップの概要は次のとおりです。

1. 以前のリリースのPDBを以前のリリースのCDBから切断します。
2. CDBからPDBを削除します。
3. 以前のリリースのPDBを新しいリリースのCDBに接続します。
4. 以前のリリースのPDBを新しいリリースにアップグレードします。

パラレル・アップグレード・ユーティリティを使用して手動でアップグレードすることにした場合は、リストを使用してPDBのアップグレードを管理できます。

- 優先度リスト: PDBのアップグレード順序を設定します
- 包含リスト: 優先度リストに指定されたPDBがアップグレードされた後にアップグレードするPDBのセットを指定できます
- 除外リスト: アップグレードしないPDBのセットを指定できます

ノート:

PDB は、バックアップしないとリカバリできません。CDB を作成して PDB に接続する方法でアップグレードした後は、必ず PDB をバックアップしてください。

関連項目

- 『Oracle Databaseバックアップおよびリカバリ・アドバンスト・ユーザーズ・ガイド』
- Oracle Database管理者ガイド

以前のリリースのPDBの以前のリリースのCDBからの切断

PDBのアップグレードを準備するには、この手順を使用して以前のリリースのCDBからPDBを切断します。

1. データベースのアップグレード準備が整っているかどうかを確認するには、preupgradeパラメータを指定してAutoUpgradeをanalyzeモードで実行します。たとえば、データベースsalespdbがOracleホーム/u01/app/oracle/product/12.2.0/dbhome1にある場合、Oracle Database 19cにアップグレードする準備状況を確認します。

   a. setenv ORACLE_HOME /u01/app/oracle/product/12.2.0/dbhome1を実行します。
   b. setenv ORACLE_SID salespdbを実行します。
   c. java -jar autoupgrade.jar -preupgrade "target_version=19,dir=/autoupgrade/test/log" –mode fixupsを実行します。
   d. ディレクトリ/autoupgrade/test/log/salespdb/prefixupsの下のprefixups.xmlを確認します。
2. 自動的に修正できなかったAutoUpgradeで検出された問題を修正します。
3. 切断するPDBをクローズします。

   たとえば、PDB salespdbをクローズするには、次のコマンドを使用します。

   ```
   SQL> ALTER PLUGGABLE DATABASE salespdb CLOSE;
   ```

4. もう一度CDB$ROOTにログインします。

   ```
   CONNECT / AS SYSDBA
   SQL> ALTER SESSION SET CONTAINER=CDB$ROOT;
   ```

5. 次のSQLコマンド構文を使用して以前のリリースのPDBを切断します(pdbはPDBの名前で、pathはPDB XMLファイルの場所です)。

   ```
   ALTER PLUGGABLE DATABASE pdb UNPLUG INTO 'path/pdb.xml';
   ```

   たとえば、pdbの名前がsalespdbで、パスが/home/oracle/salespdb.xmlの場合:

   ```
   SQL> ALTER PLUGGABLE DATABASE salespdb UNPLUG INTO '/home/oracle/salespdb.xml';
   ```

   コマンドが完了すると、次のレスポンスが表示されます。

   ```
   Pluggable database altered
   ```

6. プラガブル・データベースsalespdbは削除しますが、データ・ファイルは保持します。

   CDBビューに残っている情報をクリーン・アップし、将来の問題を回避するために、この手順の後にsalespdbを削除することをお薦めします。ベスト・プラクティス・ガイドラインとして、まず対象CDBのPDBをバックアップし、その後にソースでDROPコマンドを発行します。

注意:

DROP コマンドによりバックアップ・ファイルが削除されるため、元の CDB から PDB を削除した後は、先に

行ったバックアップを使用して元に戻すことはできません。

プラガブル・データベースを削除するには、次のコマンドを入力します。

```
SQL> DROP PLUGGABLE DATABASE salespdb KEEP DATAFILES;
```

7. 終了します。

親トピック: 切断/接続を使用したマルチテナント・アーキテクチャの順次アップグレード

以前のリリースのPDBの新しいリリースのCDBへの接続

以前のリリースのCDBから新しいリリースのCDBにPDBを接続するには、CREATE PLUGGABLE DATABASEコマンドを使用します。

次の手順例では、Oracle管理ファイルを使用しているときのPDBへの接続方法を示しています。PDBの接続の詳細は、『*Oracle Database*管理者ガイド』を参照してください。

1. 新しいリリースのCDBに接続します。
2. 次のSQLコマンドを使用して以前のリリースのPDBに接続します(pdbはPDBの名前で、pathはPDB XMLファイルが存在するパスです)。

   ```
   CREATE PLUGGABLE DATABASE pdb USING 'path/pdb.xml';
   ```

   たとえば:

   ```
   SQL> CREATE PLUGGABLE DATABASE salespdb USING '/home/oracle/salespdb.xml';
   ```

   コマンドが完了すると、次のレスポンスが表示されます。

   ```
   Pluggable database created.
   ```

ノート:

以前のリリースの PDB に接続するときには、PDB は制限モードです。アップグレードの場合にのみ、PDB をオープンできます。

関連項目

- Oracle Database管理者ガイド

親トピック: 切断/接続を使用したマルチテナント・アーキテクチャの順次アップグレード

以前のリリースのPDBの新しいリリースへのアップグレード

UPGRADEモードでのPDBのオープンでは、パラレル・アップグレード・ユーティリティを使用して、以前のリリースのPDBをCDBのリリース・レベルにアップグレードします。

1. 必要に応じて、アップグレードするPDBに切り替えます。たとえば、PDB salespdbに切り替えるには、次のコマンドを入力します。

   ```
   SQL> ALTER SESSION SET CONTAINER=salespdb;
   ```

2. PDBをUPGRADEモードで開きます。

```
SQL> ALTER PLUGGABLE DATABASE OPEN UPGRADE;
```

3. パラレル・アップグレード・ユーティリティ・コマンド(catctl.pl、またはシェル・ユーティリティdbupgrade)を使用してPDBをアップグレードします。

   PDBをアップグレードする場合、パラレル・アップグレード・ユーティリティで通常使用するコマンドを使用します。ただし、アップグレードするPDBを指定するためにオプション-c PDBnameも追加します。salespdbという名前のPDBを使用する次の例に示すとおり、PDBの名前を大文字にします。

```
$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catctl.pl -d  ¥
$ORACLE_HOME/rdbms/admin -c 'salespdb' -l $ORACLE_BASE catupgrd.sql
```

4. 結果を確認します。

   ログのデフォルトのファイル・パスはOracle_base/cfgtoollogs/dbname/upgradedatetimeです。Oracle_baseはOracleベース・パス、dbnameはデータベース名、upgradedatetimeはアップグレードの日時です。日時の文字列の書式はYYYYMMDDHHMMSCで、YYYYは年、MMは月、DDは日、HHは時間、MMは分、SCは秒を表します。

   たとえば:

```
$ORACLE_BASE/cfgtoollogs/salespdb/upgrade20181015120001/upg_summary.log
```

5. アップグレード後修正を実行し、databaseUseのINVALIDオブジェクトを再コンパイルするには、ユーティリティcatcon.plを使用してスクリプトpostupgrade_fixups.sqlを実行します。

```
$ORACLE_HOME/perl/bin/perl catcon.pl –c 'salespdb' -n 1 -e -b postfixups -d
'''.''' /tmp/cfgtoollogs/salespdb/preupgrade/postupgrade_fixups.sql
```

6. $ORACLE_HOME/rdbms/adminディレクトリから、ユーティリティcatcon.plを使用してutlrp.sqlを実行します。

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

   このスクリプトは、データベース内のINVALIDオブジェクトを再コンパイルし、現在のディレクトリにcomp0.logという名前のログ・ファイルを配置します。

親トピック: 切断/接続を使用したマルチテナント・アーキテクチャの順次アップグレード

PDBのアップグレードに包含リストまたは除外リストを使用する

以前のリリースのPDBのサブセットをアップグレードする場合は、包含リストまたは除外リストを使用して、新しいリリース・レベルのCDBまたはPDBを再度アップグレードしないようにします。

アップグレードするコンテナを記録し、包含リストまたは除外リストを使用して、これらのコンテナが連続バルク・アップグレードから除外されるようにします。アップグレードしたコンテナを連続バルク・アップグレードから除外することで、アップグレードが必要なPDBでのみアップグレードが実行されることが保証されます。再アップグレードを回避することでアップグレード時間全体が短縮され、不要なダウン時間を避けることになります。

たとえば、マルチテナント・アーキテクチャ・デプロイメントを使用してOracle Databaseをインストールした場合は、コンテナCDB$ROOT、PDB$SEED、およびCDBが作成されたときに作成された他のPDBは、新しいリリースのマルチテナント・アーキテクチャの一部です。CDBおよび、同時に一連のPDBを新しいリリースにアップグレードした場合は、CDBコンテナも、アップグレードしたPDBも再度アップグレードする必要はありません。

どちらの場合でも、以前のリリースのPDBに接続し、それらをアップグレードするときに、除外リストか包含リストを使用してPDBをアップグレードします。

- アップグレードするPDBセットだけを指定するには、包含リストを使用します。
- すでにアップグレードしたCDBおよびPDBを除外するには、除外リストを使用します。

アップグレード範囲を制限するための包含リストも除外リストも使用しない場合、パラレル・アップグレード・ユーティリティ(`catctl.pl`)は、アップグレードが必要なPDBだけではなく、CDB全体をアップグレードしようとします。アップグレード処理中、システムには不要なダウンタイムがあります。包含リスト・オプションと除外リスト・オプションは相互に排他的です。

親トピック: 切断/接続を使用したマルチテナント・アーキテクチャの順次アップグレード

Windowsでの非CDBのOracle Databaseの手動アップグレードの例

次の例では、Oracle Database 11gリリース2 (11.2.0.4)をOracle Database 19cにアップグレードするために実行するアップグレード前チェック、アップグレード、アップグレード後のチェックのステップを示しています。

これらのトピックの個々の例は、Microsoft Windowsプラットフォームでのアップグレードを示しています。これらではWindows環境固有のツールが使用されます。ただし、これらの例では手動インストールの概要を提供しています。

- コマンドライン・ユーティリティを使用した、Windowsでの非CDBのアップグレードの準備
  非CDB Oracle Database 11gリリース2 (11.2.0.4)データベースを非CDB Oracle Database 12c以上のリリースにアップグレードする前に、必要なアップグレード前のステップを実行します。
- コマンドライン・ユーティリティを使用した、Windowsでの非CDBの手動アップグレード
  これらの例は、非CDBのOracle Database 11gリリース2 (11.2.0.4)を非CDBのOracle Databaseにアップグレードするステップを示しています。

親トピック: パラレル・アップグレード・ユーティリティを使用したOracle Databaseの手動アップグレード

コマンドライン・ユーティリティを使用した、Windowsでの非CDBのアップグレードの準備

非CDB Oracle Database 11gリリース2 (11.2.0.4)データベースを非CDB Oracle Database 12c以上のリリースにアップグレードする前に、必要なアップグレード前のステップを実行します。

この手順では、Microsoft Windowsで非CDBのOracle Databaseから現行リリースの非CDBのOracle Databaseに手動アップグレードを実行するために、コマンドライン・ユーティリティを使用してどのようにシステムを準備するかの例を示しています。この例ではアップグレードするOracle DatabaseはOracle Database 11gリリース2 (11.2.0.4)です。

ORA_DBAグループのメンバーとしてログインします。新しいリリースのOracle Databaseバイナリをサーバーにインストールすることをお薦めします。その手順を実行するには、Windows用の『Oracle Databaseインストレーション・ガイド』を参照してください。

アップグレード前作業は次のステップで構成されます。

1. Oracleインストール所有者ユーザー・アカウント(Oracleユーザー)のユーザー環境変数を新しいリリースのOracleホームから11.2.0.4のOracleホームに再設定します。
2. 11.2.0.4 Oracle Databaseへの接続をテストします。
3. 次のコマンドを実行します。

   ```
   autoupgrade.jar -mode analyze
   ```

4. `autoupgrade.jar`レポートを確認します。
5. データベースをバックアップします。

これらの例では次のシステム、パス、ユーザーを使用します。

Oracle Database 11gリリース2(11.2.0.4)

- Oracleインストレーション所有者ユーザー・アカウント: `oracle1`、ORA_DBAグループのメンバーです。
- Oracleホーム: `C:¥app¥oracle¥product¥11.2.0¥dbhome_1`

Oracle Database 19c

- Oracleインストレーション所有者ユーザー・アカウント: `oracle2`、ORA_DBAグループのメンバーです。
- Oracleホーム: `C:¥app¥oracle1¥product¥19¥dbhome_1`

例4-13 ユーザー環境変数のリセット

この例は、Oracle Database 19cインストールで設定された環境変数の値の変更方法を示しています。

```
C:¥app¥oracle1¥product¥19¥dbhome_1¥rdbms¥admin>set
ORACLE_HOME=C:¥app¥oracle2¥product¥11.2.0¥dbhome_1
C:¥app¥oracle1¥product¥19¥dbhome_1¥rdbms¥admin>set ORACLE_SID=orcl11
```

例4-14 データベースへの接続のテスト

この例では、データベースへの接続のテストして、アップグレードするリリース11.2.0.4データベースに接続していることを確認する方法を示します。

```
C:¥app¥oracle1¥product¥19¥dbhome_1¥rdbms¥admin>sqlplus / as sysdba
        SQL*Plus: Release 11.2.0.4.0 Production on Tue Dec 11 12:14:49 2018
        Copyright (c) 1982, 2011, Oracle.  All rights reserved.
        Connected to:
        Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - 64bit Production
        With the Partitioning, OLAP, Data Mining and Real Application Testing options
        SQL> select version from v$instance;
        VERSION
        -----------------
        11.2.0.4.0
        SQL>
```

例4-15 Oracle Enterprise Managerの手動での削除(em_present)

`autoupgrade.jar -mode analyze`ログ・ファイルは、手動でOracle Enterprise Managerを削除する手順を提供しています(`em_present`)。

```
+ Remove the EM repository.

                  - Copy the rdbms/admin/emremove.sql script from the target
19.0.0.0.0
                  ORACLE_HOME into the source 11.2.0.4.0 ORACLE_HOME.

                  Step 1: If database control is configured, stop EM Database Control,
                  using the following command

                    $> emctl stop dbconsole

                  Step 2: Connect to the database using the SYS account AS SYSDBA

                    SET ECHO ON;
                    SET SERVEROUTPUT ON;
                   @emremove.sql
                 ...
                 ...
                 ...

                 Execution:

                 Step 1 - Stop dbconsole:
```

```
                        C:¥app¥oracle1¥product¥19¥dbhome_1¥rdbms¥admin> emctl stop
dbconsole
                        Oracle Enterprise Manager 11g Database Control Release
11.2.0.4.0
                        Copyright (c) 1996, 2011 Oracle Corporation.  All rights
reserved.
                        https://db01.example.com:1158/em/console/aboutApplication
                        The OracleDBConsoleorcl11 service is stopping...........
                        The OracleDBConsoleorcl11 service was stopped successfully.

                Step 2 - Remove em (provide credentials if needed)
                        ...
                        ...
                        ...
                        Dropping synonym : SETEMVIEWUSERCONTEXT ...
                        Dropping synonym : SMP_EMD_AVAIL_OBJ ...
                        Dropping synonym : SMP_EMD_DELETE_REC_ARRAY ...
                        Dropping synonym : SMP_EMD_INTEGER_ARRAY ...
                        Dropping synonym : SMP_EMD_INTEGER_ARRAY_ARRAY ...
                        Dropping synonym : SMP_EMD_NVPAIR ...
                        Dropping synonym : SMP_EMD_NVPAIR_ARRAY ...
                        Dropping synonym : SMP_EMD_STRING_ARRAY ...
                        Dropping synonym : SMP_EMD_STRING_ARRAY_ARRAY ...
                        Dropping synonym : SMP_EMD_TARGET_OBJ ...
                        Dropping synonym : SMP_EMD_TARGET_OBJ_ARRAY ...
                        Finished phase 5
                        Starting phase 6 : Dropping Oracle Enterprise Manager related
other roles ...
                        Finished phase 6
                        The Oracle Enterprise Manager related schemas and objects are
dropped.
                        Do the manual steps to studown the DB Control if not done
before running this
                        script and then delete the DB Control configuration files
                        PL/SQL procedure successfully completed.
                        SQL>
```

例4-16 OLAPカタログの手動での削除(amd_exists)

preupgrade_fixups.sqlログ・ファイルは、手動でOLAPカタログを削除するよう、指示しています(amd_exists)。

```
 + Remove OLAP Catalog by running the 11.2.0.4.0 SQL script
     $ORACLE_HOME/olap/admin/catnoamd.sql script.

     The OLAP Catalog component, AMD, exists in the database.

         sqlplus / as sysdba @%ORACLE_HOME%¥olap¥admin¥catnoamd.sql
                ...
                ...
                ...

                Synonym dropped.
                User dropped.
                Role dropped.
                SQL>
```

例4-17 ADMINISTER DATABASE TRIGGER権限の収集(trgowner_no_admndbtrg)

autoupgrade.jar -mode analyzeレポートは、ADMINISTER DATABASE TRIGGER権限(trgowner_no_admndbtrg)を付与する手順を提供しています。

```
+ Directly grant ADMINISTER DATABASE TRIGGER privilege to the owner of the
                 trigger or drop and re-create the trigger with a user that was
granted
                 directly with such. You can list those triggers using "SELECT OWNER,
```

```
                  TRIGGER_NAME FROM DBA_TRIGGERS WHERE BASE_OBJECT_TYPE='DATABASE' AND
                  OWNER NOT IN (SELECT GRANTEE FROM DBA_SYS_PRIVS WHERE
                  PRIVILEGE='ADMINISTER DATABASE TRIGGER')"

                         SQL> SELECT OWNER,
                           2       TRIGGER_NAME FROM DBA_TRIGGERS WHERE
BASE_OBJECT_TYPE='DATABASE' AND
                           3       OWNER NOT IN (SELECT GRANTEE FROM DBA_SYS_PRIVS
WHERE
                           4       PRIVILEGE='ADMINISTER DATABASE TRIGGER');
                         OWNER                          TRIGGER_NAME
                         ------------------------------ ------------------------------
                         MDSYS                          SDO_DROP_USER
                         MDSYS                          SDO_ST_SYN_CREATE
                         MDSYS                          SDO_TOPO_DROP_FTBL
                         MDSYS                          SDO_GEOR_BDDL_TRIGGER
                         MDSYS                          SDO_GEOR_ADDL_TRIGGER
                         MDSYS                          SDO_NETWORK_DROP_USER
                         6 rows selected.
                         SQL> grant ADMINISTER DATABASE TRIGGER to MDSYS;
                         Grant succeeded.
```

例4-18 マテリアライズド・ビューの更新(mv_refresh)

`autoupgrade.jar -mode analyze`ログ・ファイルは、マテリアライズド・ビューを更新する手順を提供しています(`mv_refresh`)。

```
+ Please make sure that all the MVs are refreshed and sys.sumdelta$
                  becomes empty before doing upgrade, unless you have strong business
                  reasons not to do so. You can use dbms_mview.refresh() to refresh
the
                  MVs except those stale ones  to be kept due to business need. If
there
                  are any stale MVs depending on changes in sys.sumdelta$, do not
truncate
                  it, because doing so will cause wrong results after refresh.

                         SQL> declare
                           2  num_failures integer(3) :=0;
                           3  begin
                           4  DBMS_MVIEW.REFRESH_ALL_MVIEWS(num_failures,'C','', TRUE,
FALSE);
                           5  end;
                           6  /
                         PL/SQL procedure successfully completed.
                         SQL> select count(1) from sumdelta$;
                           COUNT(1)
                         ----------
                                         0
                         SQL>
```

例4-19 Oracle Application Expressのアップグレード(apex_upgrade_msg)

`autoupgrade.jar -mode analyze`レポートは、Oracle Application Expressをアップグレードできることを推奨しています(`apex_upgrade_msg`)。このアップグレードを実行すると、Oracle Databaseのアップグレードに必要なダウンタイムを短縮できます。ログ・ファイルには、このアップグレードの完了方法に関する詳細が示されています。

```
+ Consider upgrading APEX manually, before the database upgrade.

                  The database contains APEX version 3.2.1.00.12 and will need to be
                  upgraded to at least version 5.0.4.00.11.

                  To reduce database upgrade time, you can upgrade APEX manually
before
```

```
                 the database upgrade.  Refer to My Oracle Support Note 1088970.1 for
                 information on APEX installation upgrades.

                Download: http://www.oracle.com/technetwork/developer-
tools/apex/downloads/index.html provide credentials.

                Then go to: http://www.oracle.com/technetwork/developer-
tools/apex/application-express/upgrade-apex-for-xe-154969.html and

https://docs.oracle.com/cd/E59726_01/install.50/e39144/toc.htm to guide you to do the
upgrade.

                Once downloaded, please go to the directory and unzip the files, run
the apex upgrade from there.

                ...
                ...
                        Completing registration process. 12:07:41
                        Validating installation.  12:07:41
                        ...Starting validation 12:07:41
                        ...Database user "SYS", database schema "APEX_050000", user#
"90" 12:07:41
                        ...272 packages
                        ...265 package bodies
                        ...465 tables
                        ...8 functions
                        ...16 procedures
                        ...4 sequences
                        ...497 triggers
                        ...1582 indexes
                        ...255 views
                        ...0 libraries
                        ...14 types
                        ...5 type bodies
                        ...0 operators
                        ...0 index types
                        ...Begin key object existence check 12:07:53
                        ...Completed key object existence check 12:07:54
                        ...Setting DBMS Registry 12:07:54
                        ...Setting DBMS Registry Complete 12:07:54
                        ...Exiting validate 12:07:54
                        PL/SQL procedure successfully completed.
                        timing for: Validate Installation
                        Elapsed: 00:00:13.00
                        Session altered.
                        timing for: Complete Installation
                        Elapsed: 00:18:40.49

                ...
                ...
```

注意:

アップグレード前ステップが完了したら、アップグレードを続行する前にデータベースをバックアップすることをお薦めします。

関連項目

- アップグレードのためのOracle Databaseのバックアップ

親トピック: Windowsでの非CDBのOracle Databaseの手動アップグレードの例

コマンドライン・ユーティリティを使用した、Windowsでの非CDBの手動アップグレード

これらの例は、非CDBのOracle Database 11gリリース2 (11.2.0.4)を非CDBのOracle Databaseにアップグレードするステップを示しています。

アップグレード前ステップを実行した後で、非CDBのOracle Databaseを新しいリリースの非CDBのOracle Databaseにアップグレードできます。アップグレードを開始する前に、データベース・サービスを停止する必要があります。データベース・サービスは、コマンドライン・コマンドを使用するか、またはMicrosoft Windows PowerShellスクリプトを使用して停止できます。

アップグレードを完了するステップの順序は次のとおりです。

1. コマンドライン・コマンドまたはPowerShellを使用して、データベース・サービスを停止します。
2. 以前のリリースのOracleホームからデータベース・サービスを削除します。
3. 以前のリリースのOracleホームのリスナーを停止します。
4. 環境変数を新しいOracleホームに設定します。
5. `tnsnames.ora`、`listener.ora`、パスワード・ファイル、ウォレット、その他のデータベース・ファイルを新しいOracleホームにコピーします。
6. PFILEを新しいOracle Database Oracleホームにコピーし、OracleホームでOracle Databaseバイナリを使用して新規サービスを作成します。(この例で、PFILEは新しいリリースと互換性があると想定しています。お使いのPFILEが新しいリリースと互換性がない場合もあります。)
7. データベースのアップグレードを開始します。
8. アップグレード後のステップを完了します。

例4-20 コマンドライン・コマンドを使用したデータベース・サービスの停止

1. サービス名がわからない場合は、サービス名を特定します。

```
c:¥apex¥apex>sc query type= service | find /i "orcl11"
                                  SERVICE_NAME: OracleServiceORCL11
                                  DISPLAY_NAME: OracleServiceORCL11
```

2. サービス名を使用して、サービスのステータスを調べます。

```
c:¥apex¥apex>sc query  OracleServiceORCL11
                                  SERVICE_NAME: OracleServiceORCL11
                                          TYPE               : 10
WIN32_OWN_PROCESS
                                          STATE              : 4  RUNNING

(STOPPABLE, PAUSABLE, ACCEPTS_SHUTDOWN)
                                          WIN32_EXIT_CODE    : 0  (0x0)
                                          SERVICE_EXIT_CODE  : 0  (0x0)
                                          CHECKPOINT         : 0x0
                                          WAIT_HINT          : 0x0
```

3. サービスを停止します

```
c:¥apex¥apex>sc stop  OracleServiceORCL11
                                  SERVICE_NAME: OracleServiceORCL11
                                          TYPE               : 10
WIN32_OWN_PROCESS
                                          STATE              : 3  STOP_PENDING
```

```
(STOPPABLE, PAUSABLE, ACCEPTS_SHUTDOWN)
                                        WIN32_EXIT_CODE    : 0  (0x0)
                                        SERVICE_EXIT_CODE  : 0  (0x0)
                                        CHECKPOINT         : 0x5
                                        WAIT_HINT          : 0x15f90
```

4. 数分間待ってからステータスを確認します。

```
c:¥apex¥apex>sc query  OracleServiceORCL11
                                SERVICE_NAME: OracleServiceORCL11
                                        TYPE               : 10
WIN32_OWN_PROCESS
                                        STATE              : 1  STOPPED
                                        WIN32_EXIT_CODE    : 0  (0x0)
                                        SERVICE_EXIT_CODE  : 0  (0x0)
                                        CHECKPOINT         : 0x0
                                        WAIT_HINT          : 0x0
```

サービスが停止したら、サービスの削除を続行できます。

例4-21 Microsoft Windows PowerShellスクリプトを使用したデータベース・サービスの停止

1. サービスのステータスをチェックします。

```
PS C:¥app¥oraclbm¥cfgtoollogs¥orcl¥preupgrade> Get-Service | Where-Object
{$_.displayName.Contains("ORCL11")}
                                Status   Name               DisplayName
                                ------   ----               -----------
                                Stopped  OracleJobSchedu...
OracleJobSchedulerORCL11
                                Running  OracleServiceOR... OracleServiceORCL11
                                Running  OracleVssWriter... Oracle ORCL11 VSS
Writer Service
```

2. サービスを停止します。

```
PS C:¥app¥oraclbm¥cfgtoollogs¥orcl¥preupgrade> Get-Service | Where-Object
{$_.displayName.Contains("ORCL11")} | Stop-Ser
                                vice
                                WARNING: Waiting for service
'OracleServiceORCL11 (OracleServiceORCL11)' to stop...
                                WARNING: Waiting for service
'OracleServiceORCL11 (OracleServiceORCL11)' to stop...
                                WARNING: Waiting for service
'OracleServiceORCL11 (OracleServiceORCL11)' to stop...
                                WARNING: Waiting for service
'OracleServiceORCL11 (OracleServiceORCL11)' to stop...
                                WARNING: Waiting for service
'OracleServiceORCL11 (OracleServiceORCL11)' to stop...
                                WARNING: Waiting for service
'OracleServiceORCL11 (OracleServiceORCL11)' to stop...
                                WARNING: Waiting for service
'OracleServiceORCL11 (OracleServiceORCL11)' to stop...
                                WARNING: Waiting for service
'OracleServiceORCL11 (OracleServiceORCL11)' to stop...
                                WARNING: Waiting for service
'OracleServiceORCL11 (OracleServiceORCL11)' to stop...
                                WARNING: Waiting for service
'OracleServiceORCL11 (OracleServiceORCL11)' to stop...
```

サービスが停止したら、データベース・サービスの削除を続行できます。

例4-22 以前のリリースのOracleホームからのデータベース・サービスの削除

Windows上のOracle Database 11gリリース2 (11.2.0.4)の場合は、以前のリリースのOracleホームのORADIMを使用します。ORADIMはWindowsサービスの管理に使用できるWindows固有のユーティリティです。

```
c:¥apex¥apex>oradim -delete -sid orcl11
                        Unable to stop service, OS Error = 1062
                        Instance deleted.
```

例4-23 以前のリリースのOracleホームのリスナーの停止

```
c:¥apex¥apex>lsnrctl status
                        LSNRCTL for 64-bit Windows: Version 11.2.0.4.0 - Production
on 13-JUL-2016 13:58:52
                        Copyright (c) 1991, 2012, Oracle.  All rights reserved.
                        Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC1522)))
                        STATUS of the LISTENER
                        ------------------------
                        Alias                     LISTENER
                        Version                   TNSLSNR for 64-bit Windows: Version
11.2.0.4.0 - Production
                        Start Date                05-JUL-2018 10:01:30
                        Uptime                    8 days 3 hr. 57 min. 23 sec
                        Trace Level               off
                        Security                  ON: Local OS Authentication
                        SNMP                      OFF
                        Listener Parameter File
C:¥app¥oracle2¥product¥11.2.0¥dbhome_1¥network¥admin¥listener.ora
                        Listener Log File
C:¥app¥oracle2¥diag¥tnslsnr¥slc01auu¥listener¥alert¥log.xml
                        Listening Endpoints Summary...

(DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(PIPENAME=¥¥.¥pipe¥EXTPROC1522ipc)))

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=db01.example.com)(PORT=1522)))
                        Services Summary...
                        Service "CLRExtProc" has 1 instance(s).
                          Instance "CLRExtProc", status UNKNOWN, has 1 handler(s) for
this service...
                        The command completed successfully
c:¥apex¥apex>lsnrctl stop
                        LSNRCTL for 64-bit Windows: Version 11.2.0.4.0 - Production
on 13-JUL-2018 13:59:00
                        Copyright (c) 1991, 2012, Oracle.  All rights reserved.
                        Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC1522)))
                        The command completed successfully
```

例4-24 新しいOracleホームへの環境変数の設定

```
c:¥apex¥apex>set ORACLE_HOME=C:¥app¥oracle1¥product¥18.1.0¥dbhome_1
c:¥apex¥apex>SET PATH=%ORACLE_HOME%¥BIN;%PATH%
c:¥apex¥apex>SET ORACLE_SID=ORCL11
```

例4-25 新しいOracleホームへのPFILEのコピー、および新しいOracle Databaseバイナリを使用した新規サービスの作成

この例ではWindows固有のORADIMユーティリティを使用して新規データベース・サービスを作成する方法を示しています。この例では、以前のリリースのPFILEは新しいリリースと互換性があります。

```
c:¥apex¥apex>C:¥app¥oracle1¥product¥18.1.0¥dbhome_1¥bin¥oradim -new -sid orcl11 -
syspwd
My-sys-password -maxusers 1000 -startmode auto -pfile
C:¥app¥oracle1¥product¥18.1.0¥dbhome_1¥database¥initorcl11.ora
                        Enter password for Oracle service user:
```

```
                          Instance created.
```

例4-26 データベースのアップグレードの開始

パラレル・アップグレード・ユーティリティを使用してアップグレードを開始します。

> ノート:
>
> PFILE にあるパスのフォルダ・シンボリック・リンク(DIAGNOSTIC_DEST など)を変更する必要がある場合があります。その場
> 示されます。
>
> ```
> ORA-48173: error checking directory existence during ADR initialization [C:¥app¥orac
> ORA-00205: error in identifying control file, check alert log for more info
> ```
>
> これらのエラーが表示された場合は、以前のリリースの Oracle ホーム、および継承関係を持つすべてのフォルダに対する権限
> ベースがアクセスするすべてのアクセス制御リスト(ACL)(ウォレット・ファイルなど)にも、この変更を加えます。以前のリリースの O
> 場所として使用している場合、このステップは特に必要です。

組込みアカウントを使用している場合、サービスは`LocalSystem`として実行されます。仮想アカウントを使用している場合は、サービスはWindows仮想アカウントを使用して実行され、そのアカウントの名前はサービス名に由来します。Oracle Database 19cバイナリの所有者が仮想アカウントである場合は、Oracleホーム・ユーザーではなく、その仮想アカウントをグループ`ORA_Homename_SVCAACCTS`に追加する必要があります。

Oracle Database 19cにはシェル・スクリプト`dbupgrade`が含まれています。このシェル・スクリプトはパラレル・アップグレード・ユーティリティ(`catctl.pl`)を呼び出すので、アップグレードをコマンドとして実行できます。この例はOracle Databaseリリース11.2.0.4からリリース19cへのアップグレードを示しています。使用しているシステムでアップグレードにかかる時間は、次の例の結果により異なる場合があります。

```
C:¥app¥oracle1¥product¥19.1.0¥dbhome_1¥bin>dbupgrade
                          ...
                          ...
                          ...
                                  C:¥app¥oracle1¥product¥19.1.0¥dbhome_1¥bin>REM Batch
file to execute catctl.pl
                                  Argument list for
[C:¥app¥oracle1¥product¥19¥dbhome_1¥rdbms¥admin¥catctl.pl]
                                  Run in                c = 0
                                  Do not run in         C = 0
                                  Input Directory       d = 0
                                  Echo OFF              e = 1
                                  Simulate              E = 0
                                  Forced cleanup        F = 0
                                  Log Id                i = 0
                                  Child Process         I = 0
                                  Log Dir               l = c:¥temp2
                                  Priority List Name    L = 0
                                  Upgrade Mode active   M = 0
                                  SQL Process Count     n = 4
                                  SQL PDB Process Count N = 0
                                  Open Mode Normal      o = 0
                                  Start Phase           p = 0
                                  End Phase             P = 0
                                  Reverse Order         r = 0
                                  AutoUpgrade Resume    R = 0
                                  Script                s = 0
                                  Serial Run            S = 0
                                  RO User Tablespaces   T = 0
```

```
                                Display Phases        y = 0
                                Debug catcon.pm       z = 0
                                Debug catctl.pl       Z = 0
                                catctl.pl VERSION: [19.0.0.0.0]
                    STATUS: [production]
                    BUILD: [RDBMS_MAIN_WINDOWS.X64_160624]

C:¥app¥oracle1¥product¥19.1.0¥dbhome_1¥rdbms¥admin¥orahome.exe =
[C:¥app¥oracle1¥product¥19.1.0¥dbhome_1]

C:¥app¥oracle1¥product¥19.1.0¥dbhome_1¥bin¥orabasehome.exe =
[C:¥app¥oracle1¥product¥189.1.0¥dbhome_1]
                                catctlGetOrabase =
[C:¥app¥oracle1¥product¥19.1.0¥dbhome_1]
                                Analyzing file
C:¥app¥oracle1¥product¥19.1.0¥dbhome_1¥rdbms¥admin¥catupgrd.sql
                                Log file directory = [c:¥temp2]
                                catcon: ALL catcon-related output will be written to
[c:¥temp2/catupgrd_catcon_3252.lst]
                                catcon: See [c:¥temp2/catupgrd*.log] files for output
generated by scripts
                                catcon: See [c:¥temp2/catupgrd_*.lst] files for spool
files, if any
                                Number of Cpus        = 2
                                Database Name         = orcl11
                                DataBase Version      = 11.2.0.4.0
                                Parallel SQL Process Count            = 4
                                Components in [orcl11]
                                        Installed [APEX APS CATALOG CATJAVA CATPROC
CONTEXT JAVAVM ORDIM OWM SDO XDB XML XOQ]
                                Not Installed [DV EM MGW ODM OLS RAC WK]
                                ----------------------------------------------------
-
                                Phases [0-109]         Start Time:[2018_12_13
15:49:40]
                                ----------------------------------------------------
-
                                ***********   Executing Change Scripts   ***********
                                Serial   Phase #:0    [orcl11] Files:1
                        ...
                        ...
                        ...
                                ----------------------------------------------------
-
                                Phases [0-109]         End Time:[2018_12_13 16:53:27]
                                ----------------------------------------------------
-
                                Grand Total Time: 3830s
                                 LOG FILES: (c:¥temp2¥catupgrd*.log)
                                Upgrade Summary Report Located in:
                                c:¥temp2¥upg_summary.log
                                Grand Total Upgrade Time:    [0d:1h:3m:50s]
```

例4-27 アップグレード後チェックの実行

アップグレードしたOracle Databaseを起動し、バージョンとステータスを確認します。

```
C:¥app¥oracle1¥product¥19.1.0¥dbhome_1¥bin>sqlplus / as sysdba
                                SQL*Plus: Release 19.1.0 Production on Thu Dec 13
09:38:50 201
                                Copyright (c) 1982, 2019, Oracle.  All rights
reserved.
                                Connected to an idle instance.
                                SQL> startup
                                ORACLE instance started.
                                Total System Global Area 1828716544 bytes
```

```
Fixed Size                  8740096 bytes
Variable Size             503317248 bytes
Database Buffers         1308622848 bytes
Redo Buffers                8036352 bytes
Database mounted.
Database opened.
SQL> select open_mode from v$database;
OPEN_MODE
--------------------
READ WRITE
SQL> select version from v$instance;
VERSION
-----------------
19.0.0.0.0
SQL>
```

アップグレードが完了したら、アップグレード後チェックを実行し、オブジェクトが有効であること、そして、残っている問題がないことを確認します。

親トピック: Windowsでの非CDBのOracle Databaseの手動アップグレードの例

Dbupgradeスクリプトおよび以前のリリースのOracle Databaseのcatupgrd.sqlについて

catupgrd.sqlスクリプトの機能は、パラレル・アップグレード・ユーティリティcatctl.plと、dbupgradeおよびdbupgrade.cmdスクリプトによって置き換えられています。

以前のリリースのOracle Databaseでは、catupgrd.sqlアップグレード・ユーティリティによってアップグレードを処理していました。Oracle Database 12cリリース1 (12.1)以上では、このスクリプトは、catctl.plパラレル・アップグレード・ユーティリティと、そのコマンドライン・スクリプトのdbupgradeによって置き換えられています。パラレル・アップグレード・ユーティリティでは、パラレル処理モードとシリアル・モードの両方が提供されます。

dbupgradeスクリプトはcatctl.plを呼び出し、データ・ディクショナリ表を作成、変更します。アップグレード・スクリプトはcatctl.plも呼び出して、新しいOracle Database 12cデータベースの次のデータベース・コンポーネントをアップグレードまたはインストールします。

- Oracle Database Catalog Views
- Oracle Database Packages and Types
- JServer JAVA Virtual Machine
- Oracle Database Java Packages
- Oracle XDK
- Real Application Clusters
- Oracle Workspace Manager
- Oracle Multimedia
- Oracle XML Database
- OLAP Analytic Workspace
- Oracle OLAP API
- Oracle Text
- Oracle Spatial and Graph
- Oracle Data Mining
- Oracle Label Security
- メッセージ・ゲートウェイ
- Oracle Database Vault

親トピック: パラレル・アップグレード・ユーティリティを使用したOracle Databaseの手動アップグレード

データベースのトランスポートおよびアップグレードについて(フル・トランスポータブル・エクスポート/インポート)

データの転送には、ファイルベース・モードまたは非ファイルベース・モードを使用できます。

Oracle Data Pumpのフル・トランスポータブル・エクスポート/インポート機能には、2つのオプションがあります。

- ファイルベースのOracle Data Pumpエクスポート/インポートの使用
- 非ファイルベースのネットワーク・モードのOracle Data Pumpインポートの使用

参照:

- エクスポート・ダンプ・ファイルを使用したデータベースのトランスポートの詳細は、『Oracle Database管理者ガイド』を参照してください
- ネットワークを介したデータベースのトランスポートの手順は、『Oracle Database管理者ガイド』を参照してください

親トピック: パラレル・アップグレード・ユーティリティを使用したOracle Databaseの手動アップグレード

非CDB Oracle Databaseのアップグレード・シナリオ

非CDB Oracle Databaseのアップグレード・シナリオおよび手順を理解するには、これらのトピックを参照してください

ノート:

Oracle Database 12c リリース 1 (12.1)以上では、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。マルチテナント・アーキテクチャを使用してデプロイされた Oracle Database は、デフォルトの構成オプションです。Oracle Database 12c リリース 1 (12.1.0.1)より前のすべての Oracle Database リリースでは、非 CDB アーキテクチャを使用します。

注意:

COMPATIBLE 初期化パラメータを 12.1.0.2 に設定した後は、データベースをダウングレードできません。プラガブル・データベース(PDB)のダウングレードは、互換性が 12.1.0.1 に設定されている場合にのみ可能です。ダウングレードには、追加の制限事項があります。

アップグレードまたはダウングレードを開始する前に、ソース・データベースとターゲット・データベースを最新の四半期リリース更新(Update)、リリース更新リビジョン(Revision)、バンドル・パッチまたはパッチ・セット更新(BP または PSU)にアップグレードすることをお薦めします。

- 最新のAutoUpgradeユーティリティのダウンロード
  スムーズなアップグレードを確実にするためにチェックに対する最新の更新を取得するには、最新のAutoUpgradeユーティリティをダウンロードします。
- PDBプラグインを使用した非CDBのPDBとしての採用について
  非CDBをPDBとして手動で採用するには、非CDBを記述するXMLファイルを生成し、`DBMS_PDB.DESCRIBE`プロシージャを使用します。その後、切断されたPDBを接続するのと同じように、非CDBに接続します。
- PDBとしての非CDBの適用
  `DBMS_PDB.DESCRIBE`プロシージャを使用して非CDBをPDBに適用(移動)できます。
- マルチテナント環境におけるOracle Label Security統合
  Oracle Label Securityは、マルチテナント環境で使用できます。
- 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。
- フリート・パッチ適用およびプロビジョニングを使用したOracle Databaseのアップグレード
  Oracle Database 12cリリース2 (12.2)以上のリリースでは、フリート・パッチ適用およびプロビジョニングを使用して以前のリリースのOracle Databaseをアップグレードできるようになりました。

親トピック: パラレル・アップグレード・ユーティリティを使用したOracle Databaseの手動アップグレード

最新のAutoUpgradeユーティリティのダウンロード

スムーズなアップグレードを確実にするためにチェックに対する最新の更新を取得するには、最新のAutoUpgradeユーティリティをダウンロードします。

AutoUpgradeユーティリティは、最初のソフトウェア・リリース後も引き続き開発されています。My Oracle Supportノート2485457.1から最新バージョンを常にダウンロードすることをお薦めします。

関連項目

- My Oracle Supportノート2485457.1

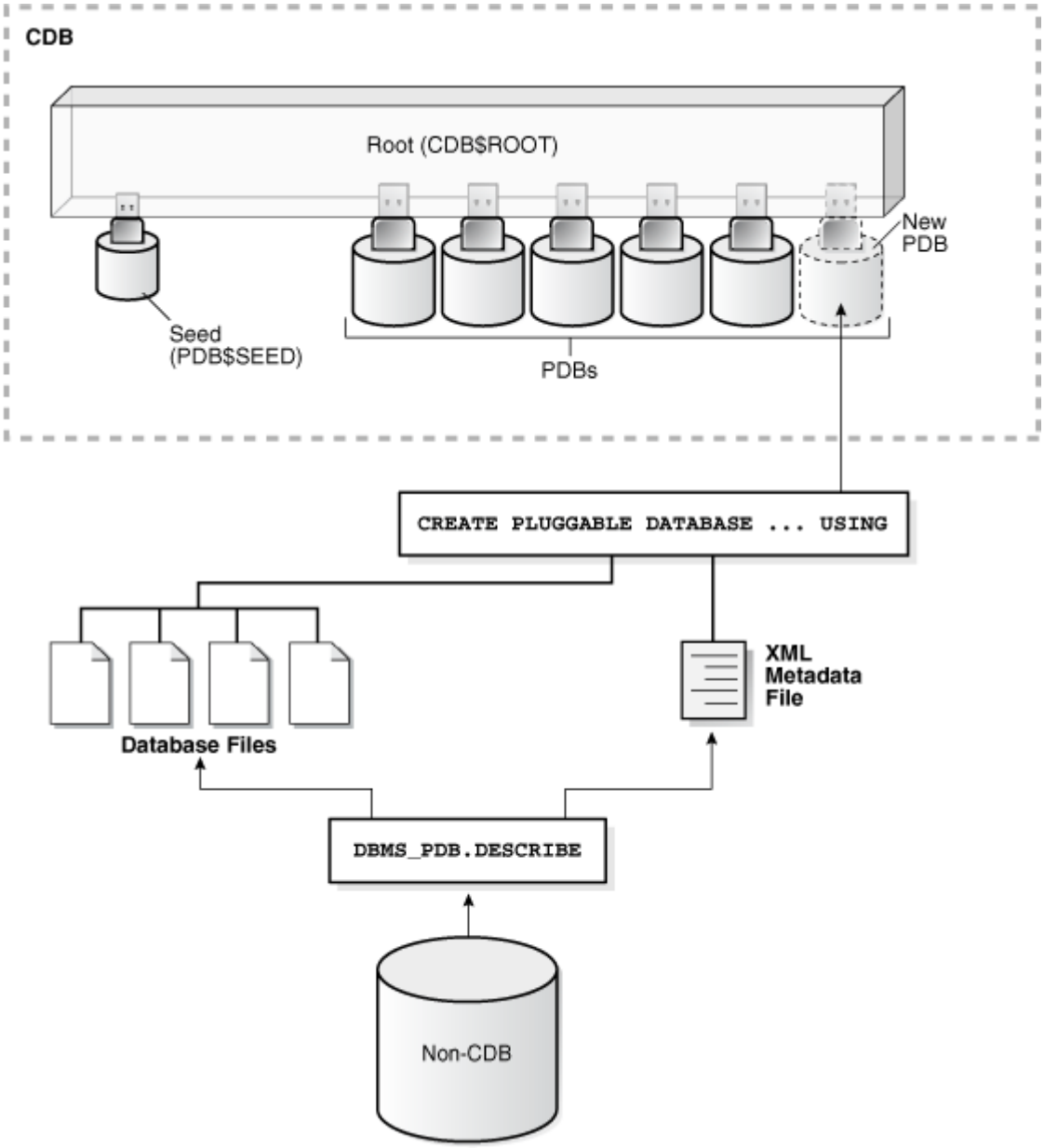
親トピック: 非CDB Oracle Databaseのアップグレード・シナリオ

PDBプラグインを使用したPDBとしての非CDBの採用について

非CDBをPDBとして手動で採用するには、非CDBを記述するXMLファイルを生成し、`DBMS_PDB.DESCRIBE`プロシージャを使用します。その後、切断されたPDBを接続するのと同じように、非CDBに接続します。

非CDBをPDBに自動的に採用およびアップグレードする取得リプレイ・メソッドを使用しないことを選択した場合、非CDBを記述し非CDBをPDBに採用する手動の手順を使用できます。`CREATE PLUGGABLE DATABASE ... USING`文を使用してPDBを作成します。非CDBがCDBに接続されると、それは新しいPDBですが、`ORACLE_HOME/rdbms/admin/noncdb_to_pdb.sql`スクリプトを使用してデータディクショナリが変換されるまで使用できません。

図4-1 DBMS_PDB.DESCRIBEプロシージャを使用した非CDBの接続



アプリケーション・コンテナに新しいアプリケーションPDBを作成するのと同じ手法を使用できます。

親トピック: 非CDB Oracle Databaseのアップグレード・シナリオ

PDBとしての非CDBの適用

DBMS_PDB.DESCRIBEプロシージャを使用して非CDBをPDBに適用(移動)できます。

この手順により、非CDBのOracle DatabaseをCDB上のPDBに更新できます。この手順を使用するには、最初にCDBを使用して新しいOracle Databaseリリースをインストールする必要があります。

1. CDBが存在しない場合は作成します。
2. 非CDBがトランザクション上一貫した状態にあることを確認してください。
3. 非CDBを読取り専用モードにします。
4. 非CDBに接続し、DBMS_PDB.DESCRIBEプロシージャを実行して、非CDBを記述するXMLファイルを作成します。

   現行ユーザーには、SYSDBA管理権限が必要です。ユーザーは、接続時にAS SYSDBAを使用して権限を行使する必要があります。

たとえば、ncdb.xmlという名前のXMLファイルを/disk1/oracleディレクトリに生成し、次のプロシージャを実行します。

```
BEGIN
  DBMS_PDB.DESCRIBE(
    pdb_descr_file => '/disk1/oracle/ncdb.xml');
END;
/
```

このプロシージャが正常に完了した後、XMLファイルおよび非CDBデータベース・ファイルを使用して、非CDBをCDBに接続できます。

5. DBMS_PDB.CHECK_PLUG_COMPATIBILITYファンクションを実行して、非CDBがCDBと互換性があるかどうかを確認します。

   このファンクションを実行する場合は、次のパラメータを設定します。

   - pdb_descr_file - パラメータをXMLファイルへのフルパスに設定します。
   - pdb_name - 新しいPDBの名前を指定します。このパラメータを省略した場合は、XMLファイル内のPDB名が使用されます。

   たとえば、/disk1/oracle/ncdb.xmlファイルで記述されている非CDBが現在のCDBと互換性があるかどうかを確認するには、次のPL/SQLブロックを実行します。

```
SET SERVEROUTPUT ON
DECLARE
  compatible CONSTANT VARCHAR2(3) :=
    CASE DBMS_PDB.CHECK_PLUG_COMPATIBILITY(
           pdb_descr_file => '/disk1/oracle/ncdb.xml',
           pdb_name       => 'NCDB')
    WHEN TRUE THEN 'YES'
    ELSE 'NO'
END;
BEGIN
  DBMS_OUTPUT.PUT_LINE(compatible);
END;
/
```

   出力がYESの場合は非CDBに互換性があり、次のステップに進むことができます。出力がNOの場合は非CDBに互換性がなく、PDB_PLUG_IN_VIOLATIONSビューをチェックして、互換性がない理由を確認できます。続行する前に、すべての違反を修正する必要があります。たとえば、バージョンまたはパッチの不一致は、アップグレードまたはデータパッチのユーティリティを実行して解決する必要があります。違反を修正してから、DBMS_PDB.CHECK_PLUG_COMPATIBILITYを再度実行して、非CDBがCDBと互換性があることを確認します。

6. 非CDBを停止します。

7. 非CDBを接続します。

   たとえば、次のSQL文は非CDBをプラグインし、そのファイルを新しい場所にコピーして、非CDBからtbs3ユーザー表領域のみを含めます。

```
CREATE PLUGGABLE DATABASE ncdb USING '/disk1/oracle/ncdb.xml'
  COPY
  FILE_NAME_CONVERT = ('/disk1/oracle/dbs/', '/disk2/oracle/ncdb/')
  USER_TABLESPACES=('tbs3');
```

   違反がない場合は、新しいPDBをオープンしないでください。次のステップでオープンします。

USER_TABLESPACES句を使用すると、非CDB内の複数のテナントに使用されたデータを異なるPDBに分けることができます。この句を指定したCREATE PLUGGABLE DATABASE文を複数使用して、非CDBに存在する他の表領域のデータを含む別のPDBを作成できます。

8. ORACLE_HOME/rdbms/admin/noncdb_to_pdb.sqlスクリプトを実行します。このスクリプトは、PDBを初めてオープンする前に実行する必要があります。

   PDBが非CDBでない場合、noncdb_to_pdb.sqlスクリプトを実行する必要はありません。noncdb_to_pdb.sqlスクリプトを実行するには、次のステップを実行します。

   - PDBにアクセスします。

     現在のユーザーにはSYSDBA管理権限が必要であり、その権限は共通で付与されているか、またはPDBでローカルに付与されている必要があります。ユーザーは、接続時にAS SYSDBAを使用して権限を行使する必要があります。

   - noncdb_to_pdb.sqlスクリプトを次のように実行します。

     ```
     @$ORACLE_HOME/rdbms/admin/noncdb_to_pdb.sql
     ```

   スクリプトによって、PDBのオープン、変更の実行、および変更の完了時にPDBのクローズが行われます。

9. 新しいPDBを読取り/書込みモードでオープンします。

   Oracle Databaseで新しいPDBをCDBに統合するには、PDBを読取り/書込みモードでオープンする必要があります。PDBを読取り専用モードでオープンしようとすると、エラーが返されます。PDBが読取り/書込みモードでオープンされると、そのステータスはNORMALになります。

10. PDBをバックアップします。

    PDBは、バックアップしないとリカバリできません。

> ノート:
>
> PDB の作成中にエラーが返される場合は、作成される PDB が UNUSABLE 状態である可能性があります。PDB の状態を確認するには、CDB_PDBS または DBA_PDBS ビューを問い合せます。PDB 作成のエラーについてさらに学習するには、アラート・ログを確認してください。使用できない PDB は削除するしかありません。使用できない PDB が作成される可能性があるので、同じ名前の PDB を作成しようとする前に、使用できない PDB を削除する必要があります。

親トピック: 非CDB Oracle Databaseのアップグレード・シナリオ

マルチテナント環境におけるOracle Label Security統合

Oracle Label Securityは、マルチテナント環境で使用できます。

マルチテナント環境では、プラガブル・データベース(PDB)をマルチテナント・コンテナ・データベース(CDB)やアプリケーション・コンテナにプラグインしたり、これらからプラグアウトできます。

次の点に注意してください。

- 各PDBには、ポリシー、ラベル、ユーザー認証などの独自のOracle Label Securityメタデータがあります。LBACSYSスキーマは共通ユーザー・スキーマです。

- PDBをCDBに接続する前に、データベースにOracle Label Securityがインストールされていない場合は、データベースに対して`$ORACLE_HOME/rdbms/admin/catols.sql`スクリプトを実行して、ラベルベースのフレームワーク、データ・ディクショナリ、データ型およびパッケージをインストールします。このスクリプトは、LBACSYSアカウントを作成します。
- Oracle Label Securityのポリシーは個々のPDBに格納されるので、PDBごとに個別のポリシーを作成できます。PDBに対して定義されたポリシーは、PDBに含まれるローカル表およびスキーマ・オブジェクトに対して強制できます。
- 単一のCDBに、それぞれOracle Label Securityで構成されている複数のPDBがある場合があります。
- Oracle Label SecurityポリシーをCDBルートやアプリケーション・ルートに作成することはできません。
- ローカルのOracle Label Securityポリシーを共通のCDBオブジェクトまたは共通のアプリケーション・オブジェクトに対して強制することはできません。
- Oracle Label Securityポリシー・ラベルおよび権限を、プラガブル・データベース内の共通のユーザーおよびアプリケーション共通ユーザーに割り当てることはできません。
- Oracle Label Securityの権限を、プラガブル・データベース内の共通のプロシージャまたはファンクション、およびアプリケーション共通のプロシージャまたはファンクションに割り当てることはできません。
- Oracle Label SecurityをOracle Internet Directoryを使用して構成する場合は、CDBに含まれるすべてのPDBで同じ構成を使用する必要があります。PDBからDBA_OLS_STATUSデータ・ディクショナリ・ビューを次のように問い合せて、データベースがOracle Internet Directoryに対して構成されているかどうかを判断できます。

  ```
  SELECT STATUS FROM DBA_OLS_STATUS WHERE NAME = 'OLS_DIRECTORY_STATUS';
  ```

  TRUEが返される場合、Oracle Label Securityではインターネット・ディレクトリが有効になっています。それ以外の場合はFALSEを返します。

関連項目

- 『Oracle Databaseセキュリティ・ガイド』

親トピック: 非CDB Oracle Databaseのアップグレード・シナリオ

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。

> ノート:
>
> 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。
- preupgradeパラメータを指定してAutoUpgradeを実行します。

ステップ:

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。たとえば:

```
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。

ノート:

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。

ノート:

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。

   たとえば:

```
$ 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ノートにとり、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。

ノート:

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。

たとえば:

```
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。

Oracle_home/binにあるdbupgradeシェル・コマンドを使用し、パラレル・アップグレード・ユーティリティをコマンドライン・シェル・コマンドとして実行できます。PATH環境変数を設定してOracle_home/binを含めると、コマンドを直接コマンドラインから実行できます。たとえば:

```
$ dbupgrade
```

ノート:

- 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際に、-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できます。

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。たとえば:

```
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、バッファがクリアされて、その他のハウスキーピング・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。

ノート:

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。

たとえば:

```
$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ときに生成されます。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。たとえば:

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

ノート:

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/cfgtoollogs/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。

たとえば:

```
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

たとえば:

```
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。

参照:

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: 非CDB Oracle Databaseのアップグレード・シナリオ

フリート・パッチ適用およびプロビジョニングを使用したOracle Databaseのアップグレード

Oracle Database 12cリリース2 (12.2)以上のリリースでは、フリート・パッチ適用およびプロビジョニングを使用して以前のリリースのOracle Databaseをアップグレードできるようになりました。

フリート・パッチ適用およびプロビジョニング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 Databaseリリースのコピーを作成し、コマンド`rhpctl upgrade database`を使用して以前のリリースのOracle Databaseをアップグレードします。このアップグレードはアウトオブプレース・アップグレードです。アップグレードが完了した後、リスナーと他の初期化変数が新しいOracleホームを指すように設定されます。フリート・パッチ適用およびプロビジョニング・イメージの作成方法の詳細は、『*Oracle Clusterware*管理およびデプロイメント・ガイド』を参照してください。

このステップの概要を使用することで、フリート・パッチとプロビジョニングを使用して以前のOracle Databaseリリースをアップグレードする方法を理解してください。

1. 新しいOracle Databaseリリースをインストールします。
2. 標準運用環境(SOE)の仕様に合わせてデータベースへのパッチ適用、テスト、構成を行います。
3. SOEリリースOracle Databaseホームからフリート・パッチ適用およびプロビジョニングのゴールド・イメージを作成します。
4. アップグレード対象のデータベースがあるサーバー上で新しいOracle Grid Infrastructureリリースへのアップグレードを実行します。このアップグレードはフリート・パッチ適用およびプロビジョニングを使用して実行できます。(ノート: ご使用のOracle Grid Infrastrauctureソフトウェアは常にOracle Databaseソフトウェアと同じか、それより新しいリリースである必要があります。)
5. アップグレード対象の以前のリリースのOracle Databaseがあるサーバーに新しいリリースのOracle Databaseフリート・パッチ適用およびプロビジョニングのゴールド・イメージのコピーをデプロイします。
6. フリート・パッチ適用およびプロビジョニング・コマンド`rhpctl upgrade database`を実行します。このコマンドは、新しいリリースのフリート・パッチ適用およびプロビジョニングのゴールド・イメージを使用して、以前のリリースのデータベースをアップグレードします。新しいリリースのOracle Databaseゴールド・イメージを使用してプロビジョニングされたサーバー上の1つ、複数、またはすべての以前のリリースのOracle Databaseインスタンスをアップグレードできます。

関連項目

- フリート・パッチ適用およびプロビジョニング

親トピック: 非CDB Oracle Databaseのアップグレード・シナリオ

Oracle Databaseのアップグレードの再実行

これらのオプションを使用してアップグレードを再実行します。

- Oracle Databaseのアップグレードの再実行について
  Oracle Databaseのアップグレードを再実行または再開始するために、Oracleでは失敗フェーズを含め、ここにリストされた機能を提供しています。
- アップグレード・スクリプト(catctl.pl)を使用したアップグレードの再実行
  アップグレードの問題を修正してから、`catctl.pl`スクリプトまたは`dbupgrade`シェル・コマンドを使用して、アップグレードを再実行できます。
- マルチテナント・データベース(CDB)のアップグレード再実行のオプション
  Oracle Database 12cリリース2 (12.2)のマルチテナント・データベース・アーキテクチャ(CDB)でアップグレードを再実行する場合、4つのオプションを使用できます。
- -pを使用して失敗したアップグレードを特定のフェーズから再開する
  このオプションを使用して、エラーの修正後にアップグレードを完了します。

親トピック: パラレル・アップグレード・ユーティリティを使用したOracle Databaseの手動アップグレード

Oracle Databaseのアップグレードの再実行について

Oracleでは失敗フェーズを含め、Oracle Databaseのアップグレードを再実行または再開始するために、ここにリストされた機能を提供しています。

パラレル・アップグレード・ユーティリティおよび再開始または再起動

Database Upgrade Assistant (DBUA)またはパラレル・アップグレード・ユーティリティ(`catctl.pl`)スクリプトを使用して、Oracle Databaseアップグレード・フェーズを再実行または再開始できます。また、最初の試行でアップグレードに失敗したPDBにコマンドを実行して、アップグレードを完了することもできます

パラレル・アップグレード・ユーティリティの再開オプション

Oracle Database 12cリリース2 (12.2)には、パラレル・アップグレード・ユーティリティの新しい再開オプションが含まれています。このオプションはCDBと非CDBの両方に使用できます。アップグレードを再実行または再開始する際に、失敗したフェーズや完了しなかったフェーズを特定する必要はありません。再開オプション(`-R`)を使用してパラレル・アップグレード・ユーティリティを使用する場合は、正常に完了しなかった以前のアップグレードのフェーズを、このユーティリティが自動的に検出します。その後、アップグレードが完了するように、パラレル・アップグレード・ユーティリティは正常に完了しなかったそれらのフェーズだけを再実行または再開始します。すでに正常に完了したステップを飛ばすことによって、アップグレードの再実行にかかる時間を減らします。

再開オプションを使用するには、-Rパラメータを使用してパラレル・アップグレード・ユーティリティを実行します。たとえば:

```
$ORACLE_HOME/perl/bin/perl catctl.pl -L plist.txt -n 4 -N 2 -R -l
$ORACLE_HOME/cfgtoollogs catupgrd.sql
```

アップグレードで失敗したのがどのフェーズであったかにかかわらず、アップグレード全体をいつでも再実行できます。失敗したフェーズだけではなく、アップグレード全体の再実行を計画している場合は、再開(`-R`)オプションを使用しないでパラレル・アップグレード・ユーティリティを実行します。

親トピック: Oracle Databaseのアップグレードの再実行

アップグレード(catctl.pl)・スクリプトを使用したアップグレードの再実行

アップグレードの問題を修正してから、catctl.plスクリプトまたはdbupgradeシェル・コマンドを使用して、アップグレードを再実行できます。

> ノート:
>
> 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。

1. データベースを停止します。非CDBとCDBでは、構文は同じです。

```
SQL> SHUTDOWN IMMEDIATE
```

2. UPGRADEモードでデータベースを再起動します。

   非CDBの場合:

```
SQL> STARTUP UPGRADE
```

   CDBの場合:

```
SQL> STARTUP UPGRADE
SQL> alter pluggable database all open upgrade;
```

3. パラレル・アップグレード・ユーティリティ(catctl.pl、またはdbupgradeシェル・コマンド)を再実行します。

   何度でも必要な回数だけパラレル・アップグレード・ユーティリティを再実行できます。

   CDBには再開オプション(-R)を使用して、パラレル・アップグレード・ユーティリティを再実行できます。スクリプトは失敗したフェーズから再開されます。

   たとえば:

```
$ORACLE_HOME/perl/bin/perl catctl.pl -n 4 -R -l $ORACLE_HOME/cfgtoollogs
catupgrd.sql
```

   また、アップグレード実行する1つ以上の特定のPDBの名前を指定することもできます。

   たとえば、次のコマンドはcdb1_pdb1という名前のPDBに対してアップグレードを再実行します。

```
$ORACLE_HOME/perl/bin/perl catctl.pl -n 4 -R -l $ORACLE_HOME/cfgtoollogs -c
'cdb1_pdb1' catupgrd.sql
```

   同じコマンドを実行するのにdbupgradeシェル・コマンドを使用できます。

```
dbupgrade -n 4 -R -l $ORACLE_HOME/diagnostics
dbupgrade -n 4 -R -l $ORACLE_HOME/diagnostics -c 'cdb1_pdb1'
```

4. utlusts.sql (アップグレード後の状態ツール)を実行します(これにより、スプール・ログにアップグレードのステータスのサマリーが表示されます)。utlusts.sqlは、アップグレードの完了する前または後にいつでも実行できますが、アップグレード中は実行できません。

   非CDBの場合:

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

CDBの場合:

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

utlusts.sqlスクリプトがエラーを返したり、VALIDではないコンポーネントを表示した場合や、最新のリリースではない場合、トラブルシューティングの手順に従って詳細を確認してください。

5. utlrp.sqlを実行して、残りのすべてのストアドPL/SQLおよびJavaコードを再コンパイルします。

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

スクリプトによって、ログ・ファイルutlrp0.logが生成され、そこに再コンパイルの結果が示されます。

次のSQLコマンドを使用して、予期されるすべてのパッケージおよびクラスが有効であることを確認します。

単一のPDB (この例ではcdb1_pdb1)で、次のように標準モードでPDBを開きます。

```
alter pluggable database cdb1_pdb1 open;
```

catcon.plを実行して、PDB内の utlrp.sqlを起動し、残りのすべてのストアドPL/SQLおよびJavaコードを再コンパイルします。次の構文を使用します。

```
$ORACLE_HOME/perl/bin/perl catcon.pl -n 1 -e -b utlrp -d '''.''' -c 'cdb1_pdb1'
 utlrp.sql
```

非CDBの場合:

```
SQL> SELECT count(*) FROM dba_invalid_objects;
SQL> SELECT distinct object_name FROM dba_invalid_objects;
```

CDB全体の場合:

```
SQL> ALTER SESSION SET CONTAINER = "CDB$ROOT"
SQL> SELECT count(*) FROM dba_invalid_objects;
SQL> SELECT distinct object_name FROM dba_invalid_objects;
SQL> ALTER SESSION SET CONTAINER = "PDB$SEED"
SQL> SELECT count(*) FROM dba_invalid_objects;
SQL> SELECT distinct object_name FROM dba_invalid_objects;
SQL> ALTER SESSION SET CONTAINER = "cdb1_pdb1"
SQL> SELECT count(*) FROM dba_invalid_objects;
SQL> SELECT distinct object_name FROM dba_invalid_objects;
```

6. utlusts.sqlを再度実行して、すべての問題が修正されたことを確認します。

非CDBの場合:

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

CDBの場合:

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

7. SQL*Plusを終了します。

8. リリース11.2からクラスタ・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(db-unique-nameは、(インスタンス名ではなく)割り当てられたデータベース名であり、Oracle_home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 upgrade database -d db-unique-name -o Oracle_home
```

これで、データベースがアップグレードされました。Oracle Databaseのアップグレード後の作業を完了する準備が整いました。

関連項目

- マルチテナント・データベース(CDB)のアップグレードの再実行のオプション
- Oracle Databaseのアップグレードのトラブルシューティング
- Oracle Databaseのアップグレード後の作業

親トピック: Oracle Databaseのアップグレードの再実行

マルチテナント・データベース(CDB)のアップグレード再実行のオプション

Oracle Database 12cリリース2 (12.2).のマルチテナント・データベース・アーキテクチャ(CDB)でアップグレードを再実行する場合、4つのオプションを使用できます。

- CDBのアップグレード全体の再実行
  最初のアップグレードの試行中に複数の異なる問題が発生した場合、この手順を使用してアップグレード全体を再実行します。
- 指定したPDBのみのアップグレードの再実行
  「再開」オプション(-R)または「除外リスト」オプション(-C)のいずれかを指定してパラレル・アップグレード・ユーティリティを実行することにより、指定したマルチテナント・コンテナでアップグレードを再実行できます。
- 他のPDBがオンラインの間にアップグレードを再実行する
  パラレル・アップグレード・ユーティリティの再開オプションを使用して、または包含リストか除外リストを使用してオンラインのPDBを明示的に包含または除外して、PDBのアップグレードを再実行できます。
- CDBまたはPDBを指定するための包含リストを使用したアップグレードの再実行
  包含リストを使用してPDBでアップグレードを再実行する場合、この例をモデルとして使用してください。

親トピック: Oracle Databaseのアップグレードの再実行

CDBのアップグレード全体の再実行

最初のアップグレードの試行中に複数の異なる問題が発生した場合、この手順を使用してアップグレード全体を再実行します。

この例では、最初のアップグレードの試行中に発生した問題(共有プールの不足など)の修正後に、CDB$ROOT、PDB$SEEDおよびすべてのPDBでアップグレードを再度実行する方法について示します。

1. 再度アップグレードを開始します。たとえば:

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

2. パラレル・アップグレード・ユーティリティ(catctl.pl、またはdbupgradeシェル・スクリプト)を実行します。たとえば:

```
cd $ORACLE_HOME/bin/
./dbupgrade -d $ORACLE_HOME/rdbms/admin -l $ORACLE_HOME/rdbms/log
```

   CDB$ROOT、PDB$SEEDおよびCDB内のすべてのPDBを含むすべてのコンテナでアップグレードが再実行されます。

親トピック: マルチテナント・データベース(CDB)のアップグレード再実行のオプション

指定したPDBでのみのアップグレードの再実行

再開オプション(-R)または除外リスト・オプション(-C)のいずれかを指定してパラレル・アップグレード・ユーティリティを実行することにより、指定したマルチテナント・コンテナでアップグレードを再実行できます。

次の両方の例で、マルチテナント・コンテナ・データベースには5つのPDBが含まれています。CDB1_PDB1とCDB1_PDB2はアップグレード・エラーによって失敗しましたが、それ以外のすべてのアップグレードは正常に実行されています。これらの2つのコンテナでアップグレードを実行するには、マルチテナント・データベース全体を停止し、アップグレードするPDBのみを再起動します。

ノート:

パラレル・アップグレード・ユーティリティのパラメータでは大/小文字が区別されます。

例4-28 再開オプションを使用したアップグレードの再実行

パラレル・アップグレード・ユーティリティの再開パラメータ・オプション-Rを使用して、1つ以上のマルチコンテナ(CDB)のみのアップグレードを再実行できます。

次の例で、アップグレード・スクリプトはCDB1_PDB1およびCDB1_PDB2コンテナのみで実行することを検出しています。

1. マルチテナント・データベースを停止し、データベースをアップグレード・モードで起動してから、アップグレードが完了していないPDBを起動します。たとえば:

```
SQL> shutdown immediate;
      startup upgrade;
      alter pluggable database CDB1_PDB1 open upgrade;
      alter pluggable database CDB1_PDB2 open upgrade;
```

2. CDBとPDBのステータスを表示します。

```
 SQL> show pdbs

       CON_ID CON_NAME                        OPEN MODE  RESTRICTED
       ---------- ------------------------------ ---------- ----------
           2 PDB$SEED                        MIGRATE    YES
           3 CDB1_PDB1                       MIGRATE    YES
           4 CDB1_PDB2                       MIGRATE    YES
           5 CDB1_PDB3                       MOUNTED
           6 CDB1_PDB4                       MOUNTED
           7 CDB1_PDB5                       MOUNTED
```

3. アップグレードを再実行します。アップグレードはCDB$ROOTとPDB$SEEDが正常にアップグレードされたことを、前回のアップグレード・ログから自動的に検出します。アップグレードはCDB$ROOTとPDB$SEEDを無視して、CDB1_PDB1とCDB_PDB2だけで実行されます。次のコマンド例は、Linux/UNIXシステム用です。

```
cd $ORACLE_HOME/bin
./dbupgrade -d $ORACLE_HOME/rdbms/admin -l $ORACLE_HOME/cfgtoollogs -R
```

パラレル・アップグレード・ユーティリティはCDB1_PDB1とCDB1_PDB2でアップグレードを完了します。

例4-29 除外リストを使用したアップグレードの再実行

除外リストには、アップグレードしないコンテナが含まれています。除外リストでは、パラレル・アップグレード・ユーティリティの-Cパラメータ・オプションを使用します。ディレクトリをOracle_home/rdbms/admin/に変更し、catctl.plを使用してPerlでユーティリティを実行することにより、パラレル・アップグレード・ユーティリティを実行するか、または、ディレクトリをOracle_home/binに変更してコマンドライン・スクリプトdbupgrade -Cを実行することにより、パラレル・アップグレード・ユーティリティを実行します。この方法は、アップグレードを再実行するPDBが多い場合に役立ちます。

次の例では、除外リストを指定して、アップグレードを実行する必要のないコンテナでの実行からアップグレード・スクリプトを除外します。

1. マルチテナント・データベースを停止し、データベースをアップグレード・モードで起動してから、アップグレードが完了していないPDBを起動します。たとえば:

```
SQL> shutdown immediate;
      startup upgrade;
      alter pluggable database CDB1_PDB1 open upgrade;
      alter pluggable database CDB1_PDB2 open upgrade;
```

2. CDBとPDBのステータスを表示します。

```
 SQL> show pdbs

       CON_ID CON_NAME                        OPEN MODE  RESTRICTED
       ---------- ------------------------------ ---------- ----------
           2 PDB$SEED                       MIGRATE    YES
           3 CDB1_PDB1                      MIGRATE    YES
           4 CDB1_PDB2                      MIGRATE    YES
           5 CDB1_PDB3                      MOUNTED
           6 CDB1_PDB4                      MOUNTED
           7 CDB1_PDB5                      MOUNTED
```

3. 単一引用符で指定するスペース区切りの除外リストでCDB$ROOTとPDB$SEEDを除外して、アップグレードを再実行します。次のコマンド例は、Linux/UNIXシステム用です。

```
$ORACLE_HOME/bin/dbupgrade -d $ORACLE_HOME/rdbms/admin -l
$ORACLE_HOME/rdbms/log -C 'CDB$ROOT PDB$SEED'
```

アップグレードはCDB1_PDB1およびCDB1_PDB2で再実行し、完了します。

ノート:

Windows の場合は、二重引用符を使用して除外リストを指定し、-C オプションの除外リストを指定する必要があります。たとえば:

```
... -C "CDB$ROOT PDB$SEED"
```

親トピック: マルチテナント・データベース(CDB)のアップグレード再実行のオプション

他のPDBがオンラインの間にアップグレードを再実行する

パラレル・アップグレード・ユーティリティの再開オプションを使用して、または包含リストか除外リストを使用してオンラインのPDBを明示的に包含または除外して、PDBのアップグレードを再実行できます。

PDBでアップグレードを実行するモデルとして次の例を使用します。ここでは、他のPDBがオープンしている間に一部のPDBでアップグレードを再実行します。

この例で、コンテナCDB1_PDB1およびCDB1_PDB2はアップグレードに失敗していますが、コンテナCDB1_PDB3、CDB1_PDB4およびCDB1_PDB5は成功しています。

CDB$ROOTを通常モードで起動します。CDB1_PDB3、CDB1_PDB4およびCDB1_PDB5のコンテナがオンラインであることがわかります。CDB1_PDB3、CDB1_PDB4およびCDB1_PDB5のアップグレード・ログを確認し、これらのコンテナをオンラインに移行します。

例4-30 再開オプションを使用したPDBでのアップグレードの再実行

次の例はパラレル・アップグレード・ユーティリティの再開オプションを使用して、CDB1_PDB1とCDB1_PDB2のアップグレードを実

行する方法を示しています。再開オプションにより、すでにアップグレードしたPDBは除外されます。

1. 標準モードでCDB$ROOTを起動し、アップグレード・モードでCDB1_PDB1およびCDB1_PDB2をオープンします。CDB1_PDB3、CDB1_PDB4、CDB1_PDB5は標準モードです。たとえば:

```
SQL> startup;
      alter pluggable database CDB1_PDB1 open upgrade;
      alter pluggable database CDB1_PDB2 open upgrade;
      alter pluggable database cdb1_pdb3 open;
      alter pluggable database cdb1_pdb4 open;
      alter pluggable database cdb1_pdb5 open;
```

2. SQLコマンドshow pdbsを使用して、PDBのステータスを表示します。たとえば:

```
SQL> show pdbs
        CON_ID CON_NAME                        OPEN MODE  RESTRICTED
        ---------- ------------------------------ ---------- ----------
             2 PDB$SEED                       READ ONLY  NO
             3 CDB1_PDB1                      MIGRATE    YES
             4 CDB1_PDB2                      MIGRATE    YES
             5 CDB1_PDB3                      READ WRITE NO
             6 CDB1_PDB4                      READ WRITE NO
             7 CDB1_PDB5                      READ WRITE NO
```

3. アップグレードからCDB$ROOTを除外し、パラレル・アップグレード・ユーティリティのコマンドライン・スクリプトdbupgradeを使用して、アップグレードを再実行します。

```
cd $ORACLE_HOME/bin
./dbupgrade -d $ORACLE_HOME/rdbms/admin/ -l $ORACLE_HOME/cfgtoollogs -R
```

再開オプションを使用してアップグレードを実行するため、パラレル・アップグレード・ユーティリティはログをチェックし、CDB1でアップグレードされていないコンテナはCDB1_PDB1とCDB1_PDB2の2つのみであることを突き止めます。アップグレード・スクリプトはその2つのPDBでのみ、実行されます。PDB$SEED、CDB$ROOT、CDB_PDB3、CDB_PDB4およびCDB_PDB5でアップグレードが再実行されることはありません。

例4-31 除外リストを使用したPDBでのアップグレードの再実行

次の例はアップグレード・スクリプトを実行しないPDBを除外するために除外リストを使用して、CDB1_PDB1とCDB1_PDB2のアップグレードを実行する方法を示しています。

1. 標準モードでCDB$ROOTを起動し、アップグレード・モードでCDB1_PDB1およびCDB1_PDB2をオープンします。CDB1_PDB3、CDB1_PDB4、CDB1_PDB5は標準モードです。たとえば:

```
SQL> startup;
      alter pluggable database CDB1_PDB1 open upgrade;
      alter pluggable database CDB1_PDB2 open upgrade;
      alter pluggable database cdb1_pdb3 open;
      alter pluggable database cdb1_pdb4 open;
      alter pluggable database cdb1_pdb5 open;
```

2. SQLコマンドshow pdbsを使用して、PDBのステータスを表示します。たとえば:

```
SQL> show pdbs
        CON_ID CON_NAME                        OPEN MODE  RESTRICTED
        ---------- ------------------------------ ---------- ----------
             2 PDB$SEED                       READ ONLY  NO
             3 CDB1_PDB1                      MIGRATE    YES
             4 CDB1_PDB2                      MIGRATE    YES
             5 CDB1_PDB3                      READ WRITE NO
             6 CDB1_PDB4                      READ WRITE NO
             7 CDB1_PDB5                      READ WRITE NO
```

3. アップグレードからCDB$ROOTを除外し、パラレル・アップグレード・ユーティリティのコマンドライン・スクリプトdbupgradeを使用して、アップグレードを再実行します。

```
cd $ORACLE_HOME/bin
./dbupgrade -d $ORACLE_HOME/rdbms/admin -l $ORACLE_HOME/cfgtoollogs -R -C
'CDB$ROOT'
```

再開オプション(-R)を指定してパラレル・アップグレード・ユーティリティが実行され、CDB1_PDB1と

CDB1_PDB2はアップグレードが完了していないことがログからわかります。除外オプション(-C)を指定してパラレル・アップグレード・ユーティリティを実行し、CDB$ROOTが除外されるように指定しているため、アップグレード・スクリプトも明示的にCDB$ROOTでの実行から除外されます。

Windowsで除外オプション(-C)を付けてパラレル・アップグレード・ユーティリティを実行する場合は、CDB root名とPDBシード名を二重引用符で囲んで-Cオプションのターゲットを指定する必要があります。たとえば:

```
. . . -C "CDB$ROOT PDB$SEED"
```

4. アップグレードはCDB1_PDB1およびCDB1_PDB2で再実行し、完了します。

親トピック: マルチテナント・データベース(CDB)のアップグレード再実行のオプション

CDBまたはPDBを指定するための包含リストを使用したアップグレードの再実行

包含リストを使用してPDBでアップグレードを再実行する場合、この例をモデルとして使用してください。

包含リストを使用して、アップグレードを再実行するCDBおよびPDBのリストを指定し、包含リストに存在しないノードを除外できます。包含オプション(-c)の後に、アップグレードするコンテナを単一引用符で囲んだスペース区切りリストを指定して、パラレル・アップグレード・ユーティリティを実行することにより、包含リストを指定します。

この例で、コンテナCDB1_PDB1およびCDB1_PDB2はアップグレードに失敗していますが、コンテナCDB1_PDB3、CDB1_PDB4およびCDB1_PDB5は成功しています。CDB$ROOTを通常モードで起動します。CDB1_PDB3、CDB1_PDB4およびCDB1_PDB5のコンテナがオンラインであることがわかります。CDB1_PDB3、CDB1_PDB4およびCDB1_PDB5のアップグレード・ログを確認し、これらのコンテナをオンラインに移行します。

例4-32 包含リストを使用したPDBでのアップグレードの再実行

たとえば:

1. 標準モードでCDB$ROOTを起動し、アップグレード・モードでCDB1_PDB1およびCDB1_PDB2をオープンします。CDB1_PDB3、CDB1_PDB4、CDB1_PDB5は標準モードです。たとえば:

```
SQL> startup;
      alter pluggable database CDB1_PDB1 open upgrade;
      alter pluggable database CDB1_PDB2 open upgrade;
      alter pluggable database cdb1_pdb3 open;
      alter pluggable database cdb1_pdb4 open;
      alter pluggable database cdb1_pdb5 open;
```

2. SQLコマンドshow pdbsを使用して、PDBのステータスを表示します。たとえば:

```
SQL> show pdbs
        CON_ID CON_NAME                        OPEN MODE  RESTRICTED
        ---------- ------------------------------ ---------- ----------
             2 PDB$SEED                       READ ONLY  NO
             3 CDB1_PDB1                      MIGRATE    YES
             4 CDB1_PDB2                      MIGRATE    YES
             5 CDB1_PDB3                      READ WRITE NO
             6 CDB1_PDB4                      READ WRITE NO
```

```
             7 CDB1_PDB5                          READ WRITE NO
```

3. 包含(-c)オプションの後に、アップグレード対象を単一引用符で囲んで指定したスペース区切りの包含リストを付けて、パラレル・アップグレード・ユーティリティを再実行します。このオプションにより、包含リストで指定したPDBに対してのみアップグレードが実行されます。

   たとえば:

   ```
   cd $ORACLE_HOME/bin
   ./dbupgrade -d $ORACLE_HOME/rdbms/admin -l ORACLE_HOME/cfgtoollogs -R -c
   'CDB1_PDB1 CDB1_PDB2'
   ```

   Windowsで包含オプションを付けてパラレル・アップグレード・ユーティリティを実行する場合は、包含リストでアップグレード対象を二重引用符で囲んで-cオプションのターゲットを指定する必要があります。たとえば:

   ```
   . . . -C "CDB1_PDB1 CDB1_PDB2"
   ```

4. アップグレードはCDB1_PDB1およびCDB1_PDB2で再実行し、完了します。

親トピック: マルチテナント・データベース(CDB)のアップグレード再実行のオプション

-pを使用して失敗したアップグレードを特定のフェーズから再開する

このオプションを使用して、エラーの修正後にアップグレードを完了します。

パラレル・アップグレード・ユーティリティ(`catctl.pl`、またはシェル・スクリプトの`dbupgrade`あるいは`dbupgrade.cmd`)を、`-p`オプションを使用して実行すると、アップグレードを再実行し、すでに正常に実行されているアップグレード・フェーズをスキップできます。失敗したフェーズの修正をテストするために、1つのフェーズに対してアップグレードを再実行することもできます。

再開するフェーズの番号を確認するには、アップグレード・ログで最初のエラーが発生した場所とフェーズを特定します。その後、エラーの原因を修正して、その修正をテストするか、アップグレードを再実行して完了できます。

- 失敗したフェーズのCDBログ・ファイルの表示
  ログ・ファイルの場所を特定し、CDBとPDBのログ・ファイルを確認します。
- 失敗したフェーズからのマルチテナント・アップグレードの特定および再開の手順
  失敗したフェーズからマルチテナント・アップグレードを再開するには、まずエラーが発生したPDBを特定し、そのエラーに関する適切なログ・ファイルを検索します。

親トピック: Oracle Databaseのアップグレードの再実行

失敗したフェーズのCDBログ・ファイルの表示

ログ・ファイルの場所を特定し、CDBとPDBのログ・ファイルを確認します

自動診断リポジトリ(ADR)とアップグレード・スクリプトによって作成される診断ログ・ファイルの場所は、環境変数およびパラメータ設定に応じて異なります。

ログ・ファイルのパスを設定するには、パラレル・アップグレード・ユーティリティ(`catctl`)の実行時に`-l`オプションを設定してログ・ファイル・パスを定義します。

CDB$ROOT (CDB)のログ・ファイルの範囲は、`catupgrd0...catupgrd7.log`です。プラガブル・データベース(PDB)のログ・ファイルは、PDBコンテナ名(`dbname`)によって識別され、その範囲は`catupgrdpdbname0...catupgrdpdbname7.log`です。

親トピック: -pを使用して失敗したアップグレードを特定のフェーズから再開する

失敗したフェーズからのマルチテナント・アップグレードの特定および再開の手順

失敗したフェーズからマルチテナント・アップグレードを再開するには、まずエラーが発生したPDBを特定し、そのエラーに関する適切なログ・ファイルを検索します。

マルチテナント・アップグレードのエラーの原因のPDBを特定するには、アップグレード・サマリー・レポートを確認するか、catupgrd0.logを参照します。このログ・ファイルの最後に、アップグレード・サマリー・レポートが記載されています。
次の手順に従い、各ログ・ファイルをチェックしてエラーを探します。

1. エラーのあるログ・ファイルを特定します。

   たとえば:

   LinuxおよびUnix

```
$ grep -i 'error at line' catupgrd*.log
```

Windows

```
C:¥> find /I "error at line" catupgrd*.log
```

grepまたはfindコマンドを使用すると、エラーが発見されたログ・ファイルのファイル名が表示されます。

2. エラーのあるログ・ファイルをチェックし、最初のエラーが発生した場所を特定します。任意のエディタを使用して、ログ・ファイルを表示します。最初の文字列「error at line」を検索します。文字列を発見したら、エラーからさかのぼって、PHASE_TIME___STARTを検索します(逆方向検索)。

   たとえば:

   ```
   PHASE_TIME___START 15 15-01-16 08:49:41
   ```

   PHASE_TIME___STARTの後の数字が、エラーの発生したフェーズ番号です。この例では、フェーズ番号は15です。

   各ログ・ファイルにエラーがあります。文字列PHASE_TIME___STARTの検索を繰り返して、すべてのログ・ファイルでエラーのあるフェーズ番号を特定し、最も小さいフェーズ番号が含まれるログ・ファイルを特定します。

   最も小さいフェーズ番号が含まれるログ・ファイルが、再開フェーズ番号(アップグレードを再開する起点となるフェーズ番号)です。

   たとえば:

   ```
   catupgrd0.log error occurred in phase 15:
   PHASE_TIME___START 15 15-01-16 08:49:41
   catupgrd1.log error occurred in phase 19:
   PHASE_TIME___START 19 15-01-16 08:50:01
   ```

   この例では、フェーズ番号15から再開します。すべてのログ・ファイルを検証して最初のエラーを探し、アップグレードを再開するフェーズを特定してください。

3. パラレル・アップグレード・ユーティリティ(catctl.pl、LinuxおよびUnixの場合はdbupgradeを使用して、Windowsの場合はdbupgrade.cmdを使用してコマンドラインから実行可能)の実行ディレクトリに変更し、失敗したフェーズからアップグレードを再開します。-pフラグを使用して、いずれかのフェーズからアップグレードを再開することを示し、再開フェーズ番号を指定します。マルチテナント・データベースでは、-c 'PDBname'という構文を通じて-cフラグも使用します(PDBnameは障害が発生したPDBの名前)。

   たとえば:

   LinuxまたはUNIXシステムの非CDB Oracle Databaseの場合:

   ```
   cd $ORACLE_HOME/bin
   dbupgrade -p 15
   ```

   WindowsシステムのマルチテナントOracle Database (CDB)のPDBの場合:

   ```
   cd $ORACLE_HOME/bin
   dbupgrade -p 15 -c 'PDB1'
   ```

   いずれの例でも、アップグレードは、-pフラグを使用して特定されたフェーズ15から再開されます。マルチテナントの例では、エラーの発生したPDBは、-cフラグで特定されます。
   これらの例で、アップグレードはフェーズ15から再開され、最後までアップグレードが実行されます。

4. (オプション) -Pフラグを使用して停止フェーズを指定することで、エラーを含むフェーズを実行することもできます。停止フェーズを使用すると、エラーが発生したフェーズのみを再実行できます。アップグレード全体を実行することなく、エラー

が発生したフェーズのみで実行することで、エラーが修正されたかどうかを迅速に特定できます。

たとえば、Perlスクリプトのパラレル・アップグレード・ユーティリティのコマンド・オプションを使用する場合、次のようになります。

```
cd $ORACLE_HOME/rdbms/admin
$ORACLE_HOME/perl/bin/perl catctl.pl -p 15 -P 15 -c 'PDB1'
```

エラーの発生したフェーズでエラーが修正されたことを確認したら、その後のフェーズのアップグレードを中止できます。

たとえば、マルチテナント・データベースPDB1のアップグレードのフェーズ15で発生したエラーが修正されたことを確認したら、LinuxおよびUnixシステムで次のコマンドを使用してフェーズ16からアップグレードを継続できます。

```
cd $ORACLE_HOME/bin dbupgrade -p 16 -c 'PDB1'
```

親トピック: -pを使用して失敗したアップグレードを特定のフェーズから再開する

5 Oracle Databaseのアップグレードのトラブルシューティング

これらのトラブルシューティングのヒントを使用して、データベースのアップグレード中に発生する可能性のあるエラーまたは問題に対処します。

また、関連リンクも参照して、発生したエラーに関連する可能性のある、このリリースに影響する変更を理解し、エラーの解決後にアップグレードを再実行する方法について確認してください。

- Oracle Databaseのアップグレード・モードでの起動について
  Oracle Databaseをアップグレード・モードで起動する場合、固定ビューに対する問合せのみを実行できます。他のビューまたはPL/SQLを実行しようとしても、エラーが発生します。
- 異なるORACLE_HOME所有者でのDBUAの実行
  Oracle Databaseホームが異なるオペレーティング・システム・ユーザー・アカウントによって所有されているか、「`upgrade.xml`が見つかりません」エラーが発生した場合、このトピックを参照してください。
- 無効なオブジェクトの警告およびDBAレジストリ・エラー
  アップグレードを開始する前に、アップグレード前情報ツール(`preupgrd.jar`)を実行することをお薦めします。
- 無効なオブジェクトおよび時期尚早なアップグレード後ツールの使用
  アップグレードが完了するまで、新しいOracle Databaseリリースに対してアップグレード後の状態ツール(`utlusts.sql`)を実行しないでください。
- Oracle Databaseアップグレード・スクリプトの終了エラーの解決
  ORA-00942、ORA-00904またはORA-01722エラーが発生した場合、この項を参照してください。
- Oracle Databaseのアップグレード中におけるリソース制限エラーの原因のトラブルシューティング
  リソース制限エラーを意味する`ORA-01650`、`ORA-01651`、`ORA-01652`、`ORA-01653`、`ORA-01654`、`ORA-01655`、`ORA-0431`、`ORA-01562`、`ORA-19815`などのエラーが発生した場合、この項を参照してください。
- Oracle DatabaseでのSQL*Plusエディション・セッションの起動エラーの解決
  この項を使用して、SP2-1540:「Oracle Databaseはエディション・セッションでは起動できません。」を理解および解決します。
- エラーORA-00020 utlrp.sqlの実行中に処理の最大数が超過する
  このエラーは、使用するOracle構成にリコンパイルに必要な数のプロセスがないことを示します。
- 「ORA-28365: ウォレットがオープンしていません」エラーの修正
  透過的暗号化(TDE)によるOracleウォレットを使用しており、Database Upgrade Assistant (DBUA)を使用してデータベースをアップグレードする場合は、ORA-28365「ウォレットがオープンしていません」エラーが発生する可能性があります。
- ビューCDB_JAVA_POLICYの問題の解決
  ビューCDB_JAVA_POLICYが無効になった場合は、この手順を使用します。
- サーバーの再起動後のアップグレードの継続(ADVM/ACFSドライバ・エラー)
  Windowsプラットフォームでは、アップグレード中にサーバーが再起動した場合に、ADVMドライバまたはACFSドライバ関連のエラーが発生する可能性があります。
- コンポーネント・ステータスおよびアップグレード
  コンポーネント・ステータス設定は、インストール済のコンポーネント、およびそれらのコンポーネントがアップグレード対象としてサポートされるかどうかの両方の要因に左右されます。
- Standard Editionの初期データベースおよびステータスがOPTION OFFのコンポーネント

Oracle Database 18c (18.1)以降、すべてのOPTION OFFコンポーネントは新しいリリースにアップグレードされますが、Oracle Database Standard Edition (SE)では、これらのオプションはOPTION OFFのまま無効化されます。

- アップグレード後のOracle ASMパスワード・ファイルの場所の調整
  Oracle Grid Infrastructureのアップグレード後にOracle ASMの新しいパスワード・ファイルを作成する必要があります。
- プラガブル・データベースのアップグレードでの「警告: XDBは現在無効です」エラーの修正
  プラガブル・データベース(PDB)のアップグレード時に「警告: XDBは現在無効です、無効なオブジェクトが見つかりました」というエラーが発生した場合、このトピックを参照してください。
- ORA-27248: 「sys.dra_reevaluate_open_failuresが実行されています」の修正
  この手順を使用して、アップグレードをブロックしているDRA_REEVALUATE_OPEN_FAILURESジョブを特定します。
- Datapatchのみが失敗した場合の失敗したアップグレードの修正
  アップグレード中にdatapatchが失敗した場合にのみ、datapatchを直接再実行します。

関連項目

- Oracle Databaseのアップグレードの再実行

Oracle Databaseのアップグレード・モードでの起動について

Oracle Databaseをアップグレード・モードで起動する場合、固定ビューに対する問合せのみを実行できます。他のビューまたはPL/SQLを実行しようとしても、エラーが発生します。

データベースがアップグレード・モードで起動すると、固定ビューの問合せのみがエラーなしで実行されます。この制限は、パラレル・アップグレード・ユーティリティ(catctl.pl)を直接実行するか、dbupgradeスクリプトを使用して間接的に実行するまで適用されます。アップグレード・スクリプトを実行する前に、他のビューでPL/SQLを使用するか、他のビューに対して問合せを実行するとエラーが戻されます。この項で説明するエラーを受け取った場合は、SHUTDOWN ABORTコマンドを発行してデータベースを停止し、問題を修正します。

新しいOracle Databaseリリースを起動しようとすると、次のエラー・リストが発生する場合があります。これらのエラーの一部はアラート・ログに記録され、セッションには表示されません。

- ORA-00401: パラメータcompatibleの値はこのリリースではサポートされません。

  COMPATIBLE初期化パラメータが11.2.0未満に設定されている場合。
- ORA-39701: データベースをUPGRADEまたはDOWNGRADE用にEXCLUSIVEでマウントしてください

  CLUSTER_DATABASE初期化パラメータがFALSEではなく、TRUEに設定されている場合。
- ORA-39700: データベースは、UPGRADEオプションを使用してオープンしてください

  UPGRADEキーワードを指定しないでSTARTUPコマンドを発行した場合。
- Ora-00704: ブートストラップ障害

  パス変数が以前のリリースのOracleホームを指している可能性がある場合。
- ORA-00336: ログ・ファイルのサイズxxxxが最小値8192より小さくなっています

  REDOログのサイズが4MB未満の場合。

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ノートにとり、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。

関連項目

- https://support.oracle.com/rs?type=doc&id=1349722.1

親トピック: Oracle Databaseのアップグレードのトラブルシューティング

異なるORACLE_HOME所有者でのDBUAの実行

Oracle Databaseホームが異なるオペレーティング・システム・ユーザー・アカウントによって所有されているか、「upgrade.xmlが見つかりません」エラーが発生した場合、このトピックを参照してください。

デフォルトのDBUAアップグレードでは、ソースOracleホームとターゲットOracleホームの両方が同じユーザーによって所有されていると想定しています。各Oracleホームが同じユーザーに所有されていない場合、データベース・ファイルの権限を変更し、DBUAに追加のパラメータを渡す必要があります。これを行わない場合、アップグレード中にDBUAの「前提条件チェック」ページに「upgrade.xmlが見つかりません」エラーがレポートされます。このエラーが修正されるまで、アップグレードを継続することはできません。

- すべてのOracle Databaseインストール所有者は、プライマリ・グループとしてOINSTALLグループ(またはOracleインベントリ・グループ)に指定したグループを保持している必要があります。すべてのデータベース・ファイル(データファイル、REDOファイル、制御ファイル、アーカイブ・ログの保存先、リカバリ領域、SPFILEおよびパスワード・ファイル)が新しいターゲット・リリースとソース(以前の)リリースの両方のバイナリ所有者によって読取りおよび書込み可能であることを確認します。そうでない場合は、各インストール所有者がそのプライマリ・グループと同じグループを持っていることを確認し、OINSTALLグループのメンバーが以前のリリースとそれ以降のリリースのすべてのOracle Databaseファイルおよびディレクトリに対する読取り/書込みアクセス権を持っていることを確認します。
- -logdirコマンドライン・オプションを指定してDBUAを実行し、新しいリリースと以前のリリースの両方のバイナリ所有者が書き込むことのできるディレクトリを指定します。たとえば、/tmpのように入力します。DBUAは、logdirパラメータで指定されたディレクトリを使用して、アップグレード前情報ツールからの出力を格納し、アップグレード中に生成されたDBUAログ・ファイルを格納します。以前のリリースのOracle Databaseインスタンスのアップグレード前情報ツールは、以前のリリースのOracle Databaseインストール所有者のユーザー・アカウントとして実行します。

  たとえば:

  ```
  dbua -logdir /tmp
  ```

親トピック: Oracle Databaseのアップグレードのトラブルシューティング

無効なオブジェクトの警告およびDBAレジストリ・エラー

アップグレードを開始する前に、アップグレード前情報ツール(preupgrd.jar)を実行することを強くお薦めします。

アップグレード前情報ツールでは、無効なSYSおよびSYSTEMオブジェクト、さらに他の無効なオブジェクトが識別されます。utlrp.sqlを使用して、無効なオブジェクトを再コンパイルします。アップグレード前にこれを行わないと、システム内で、アップグレード前に無効であったオブジェクトと、アップグレード後に無効になったオブジェクトを判別するのが難しくなります。

関連項目

- アップグレード前情報ツールおよびAutoUpgradeのアップグレード前処理

親トピック: Oracle Databaseのアップグレードのトラブルシューティング

無効なオブジェクトおよび時期尚早なアップグレード後ツールの使用

アップグレードが完了するまで、新しいOracle Databaseリリースに対してアップグレード後の状態ツール(`utlusts.sql`)を実行しないでください。

アップグレード処理が完了し、`utlrp.sql`を実行した後でのみ、アップグレード後の状態ツールを実行することをお薦めします。`@utlrp.sql`を実行する前にアップグレード後の状態ツールが実行された場合、ツールの出力に正確な最終的コンポーネントの状態値が表示されないことがあります。`utlrp.sql`を実行する前にツールが実行された場合、コンポーネントの状態値が最終的な状態を正確に反映しないことがあります。最終的なコンポーネント状態は、`utlrp.sql`を実行した後でのみ特定できます。

親トピック: Oracle Databaseのアップグレードのトラブルシューティング

Oracle Databaseアップグレード・スクリプトの終了エラーの解決

ORA-00942、ORA-00904またはORA-01722エラーが発生した場合、この項を参照してください。

アップグレードの開始前に`preupgrade`パラメータを使用してAutoUpgradeを実行しなかった場合、`catctl.pl`および`catupgrd.sql`スクリプトは次のようなエラーで終了します。

```
ORA-00942: table or view does not exist
ORA-00904: "TZ_VERSION": invalid identifier
ORA-01722: invalid number
```

次のいずれかのエラーが表示された場合は、この手順を使用して問題を解決します。

1. `SHUTDOWN ABORT`コマンドを入力し、コマンドが実行を完了するまで待機します。
2. 元のOracleホーム・ディレクトリに戻ります
3. preupgradeパラメータを使用してAutoUpgradeユーティリティを実行し、`upgrade.xml`ファイルで報告された問題を修正します。

関連項目

- Oracle Databaseのアップグレード前情報ツールの使用

親トピック: Oracle Databaseのアップグレードのトラブルシューティング

Oracle Databaseのアップグレード中におけるリソース制限エラーの原因のトラブルシューティング

リソース制限エラーを意味するORA-01650、ORA-01651、ORA-01652、ORA-01653、ORA-01654、ORA-01655、ORA-0431、ORA-01562、ORA-19815などのエラーが発生した場合、この項を参照してください。

アップグレード中にリソースが不足した場合は、リソースの割当てを増やします。リソースの割当てを増やした後、`SHUTDOWN ABORT`でインスタンスを停止し、`UPGRADE`モードでインスタンスを再起動してから`catupgrd.sql`スクリプトを再実行します。パラレル・アップグレード・ユーティリティ(`catctl.pl`)を使用してアップグレードを実行する場合、失敗した時点から自動的にアップグレードを再開するには、`-R`オプションを指定してユーティリティを実行します。問題を修正すると、AutoUpgradeが自動的にアップグレードを再開します。

通常、Oracle Databaseの新しいリリース用に増やす必要があるリソースは、次のとおりです。

- SYSTEM表領域およびSYSAUX表領域

  SYSTEM表領域のサイズが不十分な場合、通常は次のエラー・メッセージが表示されます。

  ```
  ORA-01650: unable to extend rollback segment string by string in tablespace
  string
  ORA-01651: unable to extend save undo segment by string for tablespace string
  ORA-01652: unable to extend temp segment by string in tablespace string
  ORA-01653: unable to extend table string.string by string in tablespace string
  ORA-01654: unable to extend index string.string by string in tablespace string
  ORA-01655: unable to extend cluster string.string by string in tablespace
  string
  ```

  これらのエラーを回避するには、SYSTEMおよびSYSAUX表領域にAUTOEXTEND ON MAXSIZE UNLIMITEDを設定します。

- 共有メモリー

  場合によっては、より大きな共有メモリー・プール・サイズが必要です。増やす必要がある共有メモリーの初期化パラメータが、次の形式でエラー・メッセージに示されます。

  ```
  ORA-04031: unable to allocate string bytes of shared memory
  ("string","string","string","string")
  ```

  参照:

  手動の共有メモリー管理の使用の詳細は、『Oracle Database管理者ガイド』を参照してください。

- ロールバック・セグメント/UNDO表領域

  ロールバック・セグメントを使用している場合、アップグレード・スクリプトの実行中は、1つの大きい(100MBの)パブリック・ロールバック・セグメントをオンラインにする必要があります。小さいパブリック・ロールバック・セグメントは、アップグレード中はオフラインにする必要があります。ロールバック・セグメントのサイズが不十分な場合、通常は次のエラーが発生します。

  ```
  ORA-01562: failed to extend rollback segment number string
  ```

  UNDO表領域を使用している場合は、400MB以上であることを確認してください。

- 高速リカバリ領域

  高速リカバリ領域を使用していてアップグレード中にこの領域が一杯になった場合は、アラート・ログに次のエラーが表示され、その後に問題から復旧するためのアドバイスが表示されます。

  ```
  ORA-19815: WARNING: db_recovery_file_dest_size of string bytes is 98.99%
  used, and has string remaining bytes available.
  ```

  問題の根本原因を特定し、アップグレードを続行するための適切な対処を行います。アップグレード中にこの問題が発生しないようにするには、アップグレードを開始する前に、高速リカバリ領域内の使用可能な領域を増やします。

親トピック: Oracle Databaseのアップグレードのトラブルシューティング

Oracle DatabaseでのSQL*Plusエディション・セッションの起動エラーの解決

この項を使用して、SP2-1540:「Oracle Databaseはエディション・セッションでは起動できません。」を理解および解決しま

す。

SQL*Plusでアップグレード・スクリプトまたはコマンドを実行してEDITIONパラメータを設定すると、それ以降Oracle Databaseを適切に起動できません。データベースを起動しようとすると、次のエラーが発生します。

```
SP2-1540: "Oracle Database cannot startup in an Edition session"
```

この問題を回避するには、このパラメータが変更される`catugrd.sql`またはSQL*Plusセッションを実行した後に、そのSQL*Plusセッションを終了し、別のセッションでそのインスタンスを再起動します。

親トピック: Oracle Databaseのアップグレードのトラブルシューティング

エラーORA-00020 utlrp.sqlの実行中に処理の最大数が超過する

このエラーは、使用するOracle構成にリコンパイルに必要な数のプロセスがないことを示します。

PROCESSESパラメータの設定の詳細は、Oracleのマニュアルを参照してください。

参照:

『Oracle Databaseリファレンス』

親トピック: Oracle Databaseのアップグレードのトラブルシューティング

「ORA-28365: ウォレットがオープンしていません」エラーの修正

透過的暗号化(TDE)によるOracleウォレットを使用しており、Database Upgrade Assistant (DBUA)を使用してデータベースをアップグレードする場合は、ORA-28365「ウォレットが開いていません」エラーが発生する可能性があります。

この問題を回避するには、アップグレード前に次の作業を行います。

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`ewallet.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。

   たとえば:

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

4. 通常どおり、アップグレードを開始します。

親トピック: Oracle Databaseのアップグレードのトラブルシューティング

ビューCDB_JAVA_POLICYの問題の解決

ビューCDB_JAVA_POLICYが無効になった場合は、この手順を使用します。

Oracle Database 12cリリース2 (12.2)以降のリリースへのアップグレード後、またはリリース12.2以降のリリースから12.1へのダウングレード後に、CDB_JAVA_POLICYビューに関する問題が発生することがあります。通常は機能する方法でビューを使用しているときにCDB_JAVA_POLICYが無効になるか、またはエラーが発生することがあります。エラーが発生した場合はSYSとして接続し、次のコマンドを実行します。

非CDB:

```
alter session set "_ORACLE_SCRIPT"=true;
exec CDBView.create_cdbview(false,'SYS','dba_java_policy','CDB_java_policy');
grant select on SYS.CDB_java_policy to select_catalog_role
/
create or replace public synonym CDB_java_policy for SYS.CDB_java_policy
/
```

マルチテナント・アーキテクチャ・システム:

同じこれらのコマンドを実行しますが、まずCDB$ROOTで、次にCDBの他のコンテナで実行します。

親トピック: Oracle Databaseのアップグレードのトラブルシューティング

サーバーの再起動後のアップグレードの継続(ADVM/ACFSドライバ・エラー)

Windowsプラットフォームでは、アップグレード中にサーバーが再起動した場合に、ADVMドライバまたはACFSドライバ関連のエラーが発生する可能性があります。

サーバーがアップグレード中に再起動した場合、次のいずれかのエラー・メッセージが表示されることがあります。

```
ACFS-9427: Failed to unload ADVM/ACFS drivers. A system reboot is recommended

ACFS-9428 Failed to load ADVM/ACFS drivers. A system reboot is recommended.
```

- 原因

  ADVMドライバとACFSドライバがまだ使用中です。システムを再起動して新しいドライバを起動する必要があります。

- 処置

  次の手順に記載されているステップを完了します。

最初のノード(アップグレードを開始したノード)以外のノードで、次の手順を実行します。

1. エラーが発生したノードを再起動します。
2. そのノード上で、rootスクリプトを再度実行します。

最初のノード(アップグレードを開始したノード)で、次の手順を実行します。

1. クラスタ内の他のすべてのノードでアップグレードを完了します。
2. 最初のノードを再起動します。
3. 最初のノード上で、rootスクリプトを再度実行します。
4. アップグレードを完了するには、rootでログインし、パス`Grid_home/cfgtoollogs/configToolAllCommands`にあるスクリプト`configToolAllCommands`を実行します。

参照:

クラスタのアップグレードの問題のトラブルシューティングの詳細は、使用しているオペレーティング・システムの『Oracle Grid Infrastructureインストレーション・ガイド』を参照してください。

親トピック: Oracle Databaseのアップグレードのトラブルシューティング

コンポーネント・ステータスおよびアップグレード

コンポーネント・ステータス設定は、インストール済のコンポーネント、およびそれらのコンポーネントがアップグレード対象としてサポートされるかどうかの両方の要因に左右されます。

トピック:

- アップグレード後の状態ツールを使用したコンポーネント・ステータスの理解
  アップグレード後の状態ツール(`utlusts.sql`)は、アップグレードの完了後にデータベース・コンポーネント・ステータスをレポートします。
- OPTION OFFコンポーネントのステータスおよびアップグレード
  `OPTION OFF`コンポーネントのアップグレード・ステータスは、ターゲット・リリースでコンポーネントがサポートされるかどうか、およびコンポーネントをアップグレードの一環としてアップグレードする必要があるかどうかの両方の要因に左右されます。
- アップグレード・サマリー・レポートの例
  アップグレード・サマリー・レポートでは、コンポーネントのアップグレード・ステータスに関する情報が提供されます。

親トピック: Oracle Databaseのアップグレードのトラブルシューティング

アップグレード後の状態ツールを使用したコンポーネント・ステータスの理解

アップグレード後の状態ツール(`utlusts.sql`)は、アップグレードの完了後にデータベース・コンポーネント・ステータスをレポートします。

アップグレード後の状態ツール(`utlusts.sql`)は、アップグレード後に、または`utlrp.sql`で無効なオブジェクトを再コンパイルした後にいつでも実行できます。

次のリストでは、アップグレード後の状態ツールによってレポートされるステータス値を簡単に説明します。

- `INVALID`

  アップグレードが完了したときに、コンポーネントの一部のオブジェクトが無効な状態のまま残されました。コンポーネント・アップグレードのログ・ファイルにエラーが見つからない場合、スクリプト`utlrp.sql`を実行します。このスクリプトを実行することで、アップグレード全体を再実行せずに無効なコンポーネントのステータスを`VALID`に変更できる可能性があります。`utlrp.sql`を実行した後に、`DBA_REGISTRY`ビューを確認します。
- `VALID`

  コンポーネントは有効で、エラーがありません。
- `LOADING`

  コンポーネントはロード中です。
- `LOADED`

  コンポーネントは正常にロードが終了しました。
- `UPGRADING`

  コンポーネントはアップグレードの処理中です。
- `UPGRADED`

  コンポーネントはアップグレードが完了し、エラーがありません。
- `DOWNGRADING`

  コンポーネントはダウングレードの処理中です。

- DOWNGRADED

  コンポーネントはダウングレードが完了し、エラーがありません。

- REMOVING

  コンポーネントは削除の処理中です。

- REMOVED

  データベースから削除されたため、コンポーネントはアップグレードされませんでした。

- OPTION OFF

  コンポーネントに必要なサーバー・オプションがインストールされていないか、またはサーバーとリンクされていません。V$OPTIONビューおよびインストール・ログを確認します。コンポーネントをインストールするか、サーバーを必要なオプションと再リンクしてから、パラレル・アップグレード・ユーティリティを再実行します。

- NO SCRIPT

  コンポーネント・アップグレード・スクリプトが$ORACLE_HOMEに見つかりませんでした。インストール・ログを確認し、コンポーネント・ソフトウェアをインストールしてから、パラレル・アップグレード・ユーティリティを再実行します。

ノート:

パラレル・アップグレード・ユーティリティ(catctl.pl)を、コマンドライン・スクリプト dbupgrade を使用して、サポートされているすべてのプラットフォームで実行できます。

親トピック: コンポーネント・ステータスおよびアップグレード

OPTION OFFコンポーネントのステータスおよびアップグレード

OPTION OFFコンポーネントのアップグレード・ステータスは、ターゲット・リリースでコンポーネントがサポートされるかどうか、およびコンポーネントをアップグレードの一環としてアップグレードする必要があるかどうかの両方の要因に左右されます。

OPTION OFFコンポーネントがアップグレードされるかどうかは3つのケースに分かれます。

ステータスがOPTION OFFのサポート対象外コンポーネント

データベースにステータスがOPTION OFFのコンポーネントが存在し、そのコンポーネントがターゲット・リリースへのデータベース・アップグレードの対象としてサポートされない場合、該当するコンポーネントはアップグレードされません。アップグレード後、そのバージョンおよびステータスは変更されません。

ステータスがOPTION OFFのサポート対象コンポーネント

データベースにステータスがOPTION OFFのコンポーネントが存在し、そのコンポーネントがターゲット・リリースへのデータベース・アップグレードの対象としてサポートされる場合、該当するコンポーネントはアップグレードされます。アップグレード後、コンポーネントのバージョンはターゲット・リリースのバージョンと一致します。このコンポーネントのステータスは、UPGRADED (正常なアップグレード)またはINVALID (エラー)のいずれかになります。アップグレード対象のすべてのコンポーネントのステータスがUPGRADEDになるまで、必要に応じてアップグレードを再実行してください。次に、utlrp.sqlを実行します。アップグレード前のコンポーネントのステータスがOPTION OFFであった場合、アップグレード後、コンパイルと検証が正常に終了したら、ステータスがOPTION OFFに戻ります。

アップグレードを求める必須オプションが設定されたサポート対象コンポーネント

必須オプションを持つコンポーネントは、すべてアップグレードする必要があります。該当するコンポーネントは次のとおりです。

- RAC
- SDO
- APS
- XOQ

アップグレードが必要なコンポーネントは、ステータスがOPTION OFFの標準のサポート対象コンポーネントの場合と同じ手順に従ってアップグレードされます

親トピック: コンポーネント・ステータスおよびアップグレード

アップグレード・サマリー・レポートの例

アップグレード・サマリー・レポートでは、コンポーネントのアップグレード・ステータスに関する情報が提供されます。

アップグレードが完了すると、Oracle Database 12cリリース2以上では、アップグレード・ユーティリティ・スクリプト(たとえば、Oracle Database 19の場合はutlusts.sql)によって、次の書込み順序を使用してレポートのコピーが書き込まれます。

1. ORACLE_BASE/cfgtoollogs/db_unique_name/upgradeYYMMDDHHMMSC/upg_summary.rpt
2. $ORACLE_HOME/rdbms/log/upg_summary.rpt
3. LinuxおよびUNIX:

   ```
   /tmp/upg_summary.rpt
   ```

   Windowsの場合:

   ```
   ¥TEMP¥upg_summary.rpt
   ```

ユーティリティ・スクリプトは、最初のディレクトリ・スキームにレポートを作成できない場合、引き続き2番目の場所、3番目の場所の順に書込みを試みます。これらのどのパスにも書き込むことができない場合、アップグレード・サマリー・レポートは書き込まれません。Oracle Database 18cの例を示します。

例5-1 アップグレード後の状態ツールのアップグレード・サマリー・レポート

```
Oracle Database Release 18 Post-Upgrade Status Tool    05-27-2018 23:41:0
Component                                Current          Full     Elapsed Time
Name                                     Status           Version  HH:MM:SS
Oracle Server                           UPGRADED      18.3.0.0.0  00:18:21
JServer JAVA Virtual Machine            UPGRADED      18.3.0.0.0  00:04:31
Oracle XDK                              UPGRADED      18.3.0.0.0  00:00:51
Oracle Database Java Packages           UPGRADED      18.3.0.0.0  00:00:22
OLAP Analytic Workspace                 UPGRADED      18.3.0.0.0  00:00:51
Oracle Label Security                   UPGRADED      18.3.0.0.0  00:00:20
Oracle Database Vault                   UPGRADED      18.3.0.0.0  00:00:40
Oracle Text                             UPGRADED      18.3.0.0.0  00:01:37
Oracle Workspace Manager                UPGRADED      18.3.0.0.0  00:01:10
Oracle Real Application Clusters        UPGRADED      18.3.0.0.0  00:00:00
Oracle XML Database                     UPGRADED      18.3.0.0.0  00:02:17
Oracle Multimedia                       UPGRADED      18.3.0.0.0  00:03:01
Spatial                                 UPGRADED      18.3.0.0.0  00:06:23
Oracle OLAP API                         UPGRADED      18.3.0.0.0  00:00:22
Upgrade Datapatch                                                 00:00:10
Final Actions                                                     00:00:20
Post Upgrade                                                      00:00:05
Post Upgrade Datapatch                                            00:00:04
Total Upgrade Time: 00:41:45
```

親トピック: コンポーネント・ステータスおよびアップグレード

Standard Editionの初期データベースおよびステータスがOPTION OFFのコンポーネント

Oracle Database 18c (18.1)以降、すべての`OPTION OFF`コンポーネントは新しいリリースにアップグレードされますが、Oracle Database Standard Edition (SE)では、これらのオプションはOPTION OFFのまま無効化されます。

Oracle Database Standard Edition (SE)初期データベースをアップグレードする場合、初期データベースに含まれないコンポーネントはオンになり、アップグレードされます。`utlrp.sql`を実行すると、サーバーでオンになっておらず、SEに含まれていないオプションは、`DBA_REGISTRY`ビューで`OPTION OFF`にリセットされます。

親トピック: Oracle Databaseのアップグレードのトラブルシューティング

アップグレード後のOracle ASMパスワード・ファイルの場所の調整

Oracle Grid Infrastructureのアップグレード後にOracle ASMの新しいパスワード・ファイルを作成する必要があります。

Grid Infrastructureのアップグレード後、`srvctl config asm`を実行すると、コマンド出力にはOracle ASMパスワード・ファイルの場所は表示されません。パスワード・ファイルの場所は、新しいOracle ASMディスク・グループに自動的には渡されません。SRVCTLがアップグレード後のパスワード・ファイルの場所を特定できるようにするには、ディスク・グループの互換性の設定を拡張し、ディスク・グループにPWFILEを作成する必要があります。SRVCTLは、共有PWFILEの構成済の場所を報告します。

参照:

ディスク・グループでの共有パスワード・ファイルの管理の詳細は、『Oracle Automatic Storage Management管理者ガイド』を参照してください

親トピック: Oracle Databaseのアップグレードのトラブルシューティング

プラガブル・データベースのアップグレードでの「警告: XDBは現在無効です」エラーの修正

プラガブル・データベース(PDB)のアップグレード時に「警告: XDBは現在無効です、無効なオブジェクトが見つかりました」というエラーが発生した場合、このトピックを参照してください。

Oracle Database 12cリリース1 (12.1)のプラガブル・データベース(PDB)をOracle Database 12cリリース2 (12.2)以上のマルチテナント・コンテナ・データベース(CDB)に接続する場合に、XMLオブジェクト・エラーが発生する可能性があります。

オブジェクト・リレーショナルXMLスキーマにシステム生成の名前を登録することで、共通オブジェクト(`dba_objects`に`sharing='METADATA LINK'`を含むオブジェクト)が作成されます。これらの共通タイプは、オブジェクト・リレーショナル記憶域に複数のORDSYSスキーマを登録することで作成されます。

これらの共通オブジェクトの名前はシステムによって生成されますが、リリース12.1で生成される名前は、リリース12.2以上でそれらのオブジェクトに使用される名前と異なる場合があります。これらの名前変更の可能性により、リリース12.1のオブジェクト・タイプは、リリース12.2以上のCDBルートの共通タイプと一致しないことがあります。

この問題を解決するには、次の手順を使用します。

1. ターゲットCDBルートの`PDB_PLUG_IN_VIOLATIONS`ビューを問い合せ、`'GetTypeDDL'`を含むアクションが存

在するかどうかを確認します
'GetTypeDDL'アクションが見つかった場合、アップグレードしたPDBに共通オブジェクトの問題があります。

2. ターゲットPDBでPL/SQLパッケージSET SERVEROUTPUT ONおよびexec xdb.DBMS_XMLSTORAGE_MANAGE.GetTypeDDLを実行し、ユーザー指定のSQLスクリプト(script1.sqlなど)を生成します。
3. 12.1のソースCDBで、ステップ2で作成したスクリプト(script1.sqlなど)を実行し、エラーが発生した共通タイプごとにタイプ作成スクリプトを取得します
4. これらの作成スクリプトを含む別のユーザー指定のSQLスクリプト(script2.sqlなど)を生成します。
5. 12.1のソースCDBで作成したスクリプト(script2.sqlなど)をターゲットPDBで実行します。

リリース12.1のソースCDBのタイプ作成スクリプトから生成されたスクリプトによって、ターゲットPDBにこれらすべてのオブジェクトが生成されます。通常、ターゲットPDBでこれらの共通オブジェクトを使用できるようにすることで、無効なXDBエラーは解消します。

親トピック: Oracle Databaseのアップグレードのトラブルシューティング

ORA-27248:「sys.dra_reevaluate_open_failuresが実行されています」の修正

この手順を使用して、アップグレードをブロックしているDRA_REEVALUATE_OPEN_FAILURESジョブを特定します。

アップグレード時に、DBUAがエラーORA-27248: 「sys.dra_reevaluate_open_failuresが実行されています」で失敗した場合、アップグレードの障害の原因となるジョブDRA_REEVALUATE_OPEN_FAILURESが実行されています。アップグレードを継続する前に、このジョブが停止していることを確認してください。

ジョブ定義で、ALLOW_RUNS_IN_RESTRICTED_MODEがTRUEに設定され、ジョブ所有者が制限モード中にログインできる場合、データベースが制限モードやアップグレード・モードでもそのジョブを実行できます。このパラメータのデフォルトの設定はFALSEです。

次の問合せを使用して、実行中のジョブの状態を確認してください。

```
SQL> select OBJECT_NAME, Owner, OBJECT_TYPE from dba_objects whereobject_name like
'%DRA_REEVA%';
```

親トピック: Oracle Databaseのアップグレードのトラブルシューティング

Datapatchのみが失敗した場合の失敗したアップグレードの修正

アップグレード中にdatapatchが失敗した場合にのみ、datapatchを直接再実行します。

Datapatchスクリプトはシェル・スクリプトです。一部のパッチ適用操作では、ORA-20001などのエラーのため、最終的なアップグレード後のパッチ適用は実行されないことがあります。Datapatchスクリプトのみが失敗した場合は、この問題を修正するためにアップグレードを再実行する必要はありません。かわりに、datapatchスクリプトを直接実行します。

失敗したdatapatchを修正するには、Oracleユーザーとしてログインし、この手順を完了します。

1. アップグレードされたOracleホームの中にあるOpatchディレクトリに変更します。

```
$ cd $ORACLE_HOME/OPatch
```

2. datapatchコマンドを実行します。

LinuxおよびUNIXの場合:

```
./datapatch -verbose
```

Microsoft Windowsの場合:

```
datapatch -verbose
```

親トピック: Oracle Databaseのアップグレードのトラブルシューティング

6 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。

> ノート:
>
> AutoUpgrade を使用してアップグレードを完了すると、これらのタスクの多くはアップグレードの一部として自動的に実行されます。

- 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。
- Oracleデータ・ディクショナリの現在の状態の表示方法
  アップグレードと移行の診断のためにOracleデータ・ディクショナリの状態を確認するには、次の3つの方法のいずれかを使用します。
- 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薦めします。特に明記されている場合を除き、これらの方法はすべてのタイプのアップグレードにお薦めします。
- Oracle RACデータベースのアップグレード後の推奨作業
  接続用にSCANまたはノード・リスナーを使用するようにクライアントを構成するかどうかを決定します。
- Oracle ASMのアップグレード後の推奨作業
  Oracle ASMのアップグレード後に、Oracle ASMパスワードのリセットおよびディスク・グループの構成などの作業を行うことをお薦めします。
- Oracle Database Express Editionのアップグレード後の推奨作業
  DBCAを使用するか、手動スクリプトを実行して、Oracle Databaseに追加のコンポーネントをインストールします。
- 手動によるOracle Databaseのアップグレード後にのみ行う作業
  DBUAを使用せずに手動でデータベースをアップグレードした場合、アップグレードの完了後に、ここに記載されている作業を実行する必要があります。
- Oracle Databaseをアップグレードする場合のリリースの更新および要件
  アップグレードを開始する前に、新しいリリースのOracleホームを最新のリリース更新(Update)に更新します。

Oracle Databaseをアップグレードする場合のリリースの更新および要件

アップグレードを開始する前に、新しいリリースのOracleホームを最新のリリース更新(Update)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。

アップグレードを開始する前に、新しいリリースのOracleホームを最新の四半期リリース更新(Update)に更新することをお薦めします。

My Oracle Supportで、最新更新の入手方法に加え、ライフサイクル管理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。たとえば:

- My Oracle Supportノート555.1のOracle Database 19cの重要な推奨個別パッチには、Oracle Database 19cの特定の重要度のパッチのリストが含まれています。
- My Oracle Supportノート2118136.2には、環境に必要な更新、リビジョン、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・リリースを選択する際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薦めします。
- 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。

関連項目

- Oracle Database 19cの重要な推奨個別パッチ(ドキュメントID 555.1)
- My Oracle Supportノート2118136.2
- My Oracle Supportノート1227443.1

親トピック: 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

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、パス`$ORACLE_HOME/rdbms/admin`にあります。このツールは、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

アップグレードと移行の診断のためにOracleデータ・ディクショナリの状態を確認するには、次の3つの方法のいずれかを使用します。

例6-1 dbupgdiag.sqlスクリプトの実行

dbupgdiag.sqlスクリプトは、アップグレードの前または後のいずれかで、データベースのステータスに関する診断情報を収集します。My Oracle Supportノート556610からスクリプトをダウンロードし、データベースのSYSユーザーとしてスクリプトを実行します。このスクリプトにより、db_upg_diag_sid_timestamp.logという名前のログ・ファイルに、読みやすい形式で診断情報が生成されます。sidはデータベースのOracleシステム識別子で、timestampはファイルが生成された時刻です。

たとえば、/u01/dbupgdiag-scriptディレクトリにスクリプトをダウンロードして配置する場合は次のとおりです。

```
/u01/dbupdiag-script/ $ sqlplus / as sysdba
sql> alter session set nls_language='American';
sql> @dbupgdiag.sql
sql> exit
```

スクリプトは、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。スクリプトの詳細は、My Oracle Supportノート556610.1の指示および出力サンプル・ファイルを参照してください。

例6-2 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_full,status
from dba_registry order by comp_id;
```

例6-3 無効なオブジェクトをチェックするSQL問合せの実行

無効なオブジェクトをチェックするには、次の例のようにSQL問合せを実行します。

1. データベース内のすべての無効なオブジェクトをリストします。

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

2. データベース内の無効なOracle管理オブジェクトをリストします。

   ```
   SQL> select owner, object_name, object_type from sys.dba_invalid_objects where
   oracle_maintained='Y';
   ```

   データベースをアップグレードし、utlrp.sqlを実行した後は、すべてのOracle管理オブジェクトが有効である必要があります。

3. データベース内の無効なアプリケーション・オブジェクトをリストします。

   ```
   SQL> select owner, object_name, object_type from sys.dba_invalid_objects where
   oracle_maintained='N';
   ```

関連項目

- 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。

> ノート:
>
> この必須タスクのリストは、AutoUpgrade を使用してアップグレードを実行したという前提に基づいています。AutoUpgrade では、手動で実行する必要があるタスク以外の多数のタスクを自動的に完了します。

- 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
  必要なオペレーティング・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します。
- データベース内の無効なオブジェクトの再コンパイル
  データベースをインストール、パッチ適用またはアップグレードした後、再コンパイル・ドライバ・スクリプトを使用して、CDBおよびPDB上の無効なオブジェクトを再コンパイルします。
- 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 RACクラスタ・メンバーのアップグレードでのリスナー・ファイルの場所の更新
  共有Oracleベース・ホームを使用せず、Oracle Real Application Clustersメンバー・ノードであるオラクル Databaseインスタンスでアップグレードを実行する場合は、リスナー・パスを更新する必要があります。
- 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ますが、これは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・アップグレードについて
  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にAutoUpgradeアップグレード前レポートによってタイムゾーン・ファイルのアップグレードを指示され、このタスクを完了するようにAutoUpgradeを設定しない場合は、サポートされている方法のいずれか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 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以降のリリースのOracle XML DBのポートは構成されません。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 Textナレッジ・ベースに対してユーザーが拡張した項目は、Oracle Databaseアップグレード後に再生成する必要があります。
- 以前のリリースのOracle APEXの削除
  無効なオブジェクトを回避するには、Application Express (APEX) 19.1より前のOracle APEXリリースを削除します。
- 読取り専用OracleホームでのDEMOディレクトリの置換
  読取り専用Oracleホームをアップグレードした後、以前のリリースのOracle Databaseのdemoディレクトリをコピーし、読取り専用Oracleホームのdemoディレクトリを新しいリリースのdemoディレクトリに置き換えます。
- 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ングレイン・アクセス制御が含まれています。
- 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。
- 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

必要なオペレーティング・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します。

通常、オペレーティング・システムの環境変数は、プロファイルおよびシェル・スクリプトで設定します。次のOracleユーザー環境変数が新しい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

以前のリリースのOracleホームを参照する他の環境変数(LD_LIBRARY_PATHなど)を探します。通常、環境変数内のすべての古いOracleホームを新しいOracleホーム・パスに置き換える必要があります。

関連項目

- ステップ2: 必要な環境変数が設定されていることの確認

親トピック: Oracle Databaseのアップグレード後に必要な作業

データベース内の無効なオブジェクトの再コンパイル

データベースをインストール、パッチ適用またはアップグレードした後、再コンパイル・ドライバ・スクリプトを使用して、CDBおよび

PDB上の無効なオブジェクトを再コンパイルします。

Oracleには、再コンパイル・スクリプトutlrp.sql、utlprp.sqlおよびutlprpom.sqlが用意されています。これらのスクリプトは、Oracle_home/rdbms/adminディレクトリにあります。

> ノート:
>
> AutoUpgrade 23.1 以降、AutoUpgrade ユーティリティを実行すると、AutoUpgrade によって utlprpom.sql スクリプトが実行され、utlrp.sql は実行されません。Oracle Database 12c リリース 2 (12.2.0.1)以降のリリースへのアップグレードに AutoUpgrade を使用すると、AutoUpgrade は Oracle 管理スキーマが所有する無効なオブジェクトのみを再コンパイルします。データベースのアップグレードではユーザー・オブジェクトにアクセスする必要がないため、AutoUpgrade は無効なオブジェクトを再コンパイルするときにこのポリシーを維持します。

データベースのインストール後、次の手順ですべての無効なオブジェクトを再コンパイルします。

1. ディレクトリをOracle_home/rdbms/adminに変更します。たとえば

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

2. Oracleホームのcatcon.plスクリプトを使用して、utlrp.sqlを実行します。たとえば:

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

この使用例では次の点に注意してください。

- --nパラメータ: 1に設定されているため、各PDBの再コンパイルは順番に実行されます。
- --eパラメータ: エコーをオンにします。
- --bパラメータ: ログ・ファイルのベース名を設定します。utlrpに設定されています。

PDBのシリアル再コンパイルが完了するまでの時間の遅延を予期してください。アップグレードするPDBの数によっては、再コンパイルは、アップグレード・スクリプトの完了に要する時間を大幅に超えて延長される可能性があります。

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。

データベースにパッチを適用またはアップグレードした後、無効なOracle所有オブジェクトおよびユーザー所有オブジェクトを再コンパイルするには複数のアプローチがあります。

utlrp.sqlまたはutlprp.sqlを使用して、すべての無効なオブジェクト(Oracleとユーザー・スキーマの両方で無効なオブジェクト)を再コンパイルします。

時間的な要因があり、無効なオブジェクトのタイプが主にアプリケーション所有のものである場合、Oracle所有の無効なオブジェクトを最初に再コンパイルし、アプリケーション所有の無効なオブジェクトの再コンパイルを後回しにできます。Oracleスキーマで無効なオブジェクトを再コンパイルするには、utlprpom.sqlを使用します。残りの無効なオブジェクトを再コンパイルするには、utlrp.sqlまたはutlprp.sqlを使用します。

ノート:

utlprp.sql または utlprpom.sql を使用する場合、どちらのスクリプトでも、スクリプトで使用する並列度を定義するか、使用するパラレル再コンパイル・ジョブの数を決定する必要があることに注意してください。

このスクリプトでは次の構文を使用します。baseはログ・ファイルに指定するベース名、Nは再コンパイル・ジョブをパラレルで実行するPDBの数(並列度)、script.sqlは使用するよう選択したOracle再コンパイル・スクリプト、Pはパラレルで実行するPDBの数です。

```
$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl -b base -d
$ORACLE_HOME/rdbms/admin
          -n N -l /tmp script.sql '--pP'
```

ログ・ファイルのベース名recompを使用してCDBで再コンパイルを実行し、並列度の設定はPDBコンテナごとに3件のジョブとし、使用するよう選択するスクリプトはutlprp.sqlで、一度に最大10個のPDBにわたって再コンパイルするとします。この場合、再コンパイル操作の実行に使用する構文は次のようになります。

```
$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl -b recomp -d
$ORACLE_HOME/rdbms/admin -n 10 -l /tmp utlprp.sql '--p3'
```

関連項目

- catcon.plの構文およびパラメータ

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。

ノート:

AutoUpgrade ユーティリティを使用してアップグレードした場合、AutoUpgrade がアップグレード中に自動的にこのタスクを処理します。このタスクを実行する必要はありません。

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薦めします。SQL問合せを実行してスクリプトを監視できます。

例6-4 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6-5 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6-6 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 RACクラスタ・メンバーのアップグレードでのリスナー・ファイルの場所の更新

共有Oracleベース・ホームを使用せず、Oracle Real Application Clustersメンバー・ノードであるOracle Databaseインスタンスでアップグレードを実行する場合は、リスナー・パスを更新する必要があります。

共有Oracleベース・ホームを使用せず、Oracle Real Application Clustersクラスタ・メンバー・ノードであるOracle Databaseインスタンスでアップグレードを実行する場合に、Database Configuration Assistant (DBCA)を使用しないと、すべてのリモートtnsnames.oraファイルを手動で更新する必要があります。ファイルが更新されていない場合、次のエラーが発生します。

```
TNS-03505: Failed to resolve name
```

Oracle Database 21c以降、デフォルトのネットワーク管理ディレクトリは、ローカルのOracleホームの以前のデフォルトであるOracle_home/network (/u01/app/oracle/product/19.1.0/dbhome_1/networkなど)から新しい場所に変更されます。新しいデフォルトの場所は、パスORACLE_BASE/homes/HOME_NAME/network/adminの共有Oracleベース・ホームです。たとえば、/u01/app/oracle/homes/OraDB20Home1/network/adminは特定のOracle Real Application Clusters (Oracle RAC)インスタンスのOracleホームで、読取り専用Oracleホームのデフォルト・パスにあります。そのファイル・パスは、ローカルのtnsnames.oraおよびsqlnet.oraファイルのデフォルトの場所です。

アップグレード中に、Net Configuration Assistant (NetCAまたはnetca)によって、ローカル・ホスト上のネットワーク・リスナー・ファイルtnsnames.oraおよびsqlnet.oraの場所が更新されます。ただし、NetCAは、すべてのクラスタ・メンバー・ノードのインスタンスのネットワーク・リスナー・ファイルの場所を更新するわけではありません。すべてのクラスタ・メンバー・ノードで、アップグレードされたクラスタ・メンバー・ノード・インスタンスに対するサービス・リクエストを解決できるようにするには、次のいずれかを実行する必要があります。

- 各クラスタ・メンバー・ノードで、tnsnames.oraファイルとsqlnet.oraファイルを既存の場所からORACLE_BASE/homes/HOME_NAME/network/admin/に手動でコピーします。
- アップグレードされたクラスタ・メンバー・ノードの環境変数を、既存のネットワーク管理ファイルの場所を指すように設定します。

p

回避策を確認するには: ----------tnsnames.oraおよびsqlnet.oraを$ORACLE_BASE/homes/OraDB20Home1/network/admin/sqlnet.oraにコピーまたはTNS_ADMIN=$ORACLE_HOME/network/adminをエクスポート

例6-7 TNSNAMES.ORAおよびSQLNET.ORAを新しいデフォルト・ネットワーク管理ディレクトリにコピー

この例では、既存のtnsnames.oraおよびsqlnet.oraファイルを、各クラスタ・メンバー・ノード上の新しいデフォルトのネットワークの場所にコピーします。

1. Oracleインストール所有者としてログインし、以前のリリースのネットワーク管理ディレクトリにディレクトリを変更します。

たとえば:

```
cd $ORACLE_HOME/network
```

2. セキュア・コピー(scp)を使用して、リスナー・ファイルを別のクラスタ・メンバー・ノード上のデフォルトのネットワーク・ディレクトリの場所にコピーします。たとえば、racnode2は、リスナー・ファイルをコピーするクラスタ・メンバー・ノードです。

```
cd $OLD_ORACLE_HOME/network
scp tnsnames.ora sqlnet.ora ¥
oracle@racnode2:/u01/app/oracle/homes/OraDB20Home1/network/admin/
```

例6-8 TNS_ADMIN環境変数の設定

アップグレードされたノードで、Oracleユーザーとしてログインし、既存のリスナー・ファイルの場所を指すようにTNS_ADMINの環境変数を設定します。たとえば:

```
/home/oracle oracle> $ export TNS_ADMIN=$ORACLE_HOME/network/admin
```

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す。

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インストレーション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(Update)およびリリース更新リビジョン(Revision)に関する最新の推奨事項を定期的に入手してください。

関連項目

- 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、データベースのアップグレードにAutoUpgradeを使用しない場合は、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。AutoUpgradeまたはDBUAを使用してアップグレードを実行する場合、これらのユーティリティは自動的に`oratab`を新しいOracleホームを指します。ただし、アップグレードに使用する方法にかかわらず、クライアント・スクリプトを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

- ステップ2: オペレーティング・システムの環境変数の設定
- My Oracle Supportノート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ますが、これは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、AutoUpgradeを使用する場合は、無効なオブジェクトに関するAutoUpgradeレポートを確認します。リプレイ・アップグレードを使用する場合は、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。

関連項目

- 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。

> ノート:
>
> AutoUpgrade ユーティリティを使用してアップグレードした場合、AutoUpgrade がアップグレード中に自動的にこのタスクを処理します。このタスクを実行する必要はありません。

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・モードで開いているときに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。たとえば:

```
CONNNECT 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。

参照:

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。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、この変更が行われました。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。

参照:

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・アップグレードについて

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。

参照:

- RMANバックアップおよびリポジトリ・レコードのメンテナンス
- リカバリ・カタログのアップグレード
- My Oracle Support RMAN互換性マトリックス(ドキュメントID 73431.1)

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にAutoUpgradeアップグレード前レポートによってタイムゾーン・ファイルのアップグレードを指示され、このタスクを完了するようにAutoUpgradeを設定しない場合は、サポートされている方法のいずれか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。

デフォルトでは、AutoUpgradeはデータベースのタイム・ゾーンを最新の使用可能なレベルに変更します。タイム・ゾーンをアップグレードしない場合は、AutoUpgrade構成ファイル内のローカル・パラメータtimezone_upgを明示的にnoに設定する必要があります。たとえば:

```
upg1.timezone_upg=no
```

> ノート:
>
> AutoUpgrade 構成ファイルでタイムゾーン・ファイルのアップグレードを明示的に無効にする場合は、このタスクをアップグレード計画の一環として実行するか、後で実行することをお薦めします。

関連項目

- 日時およびタイム・ゾーン・パラメータと環境変数
- Primary Note DST FAQ : Updated DST Transitions and New Time Zones in Oracle RDBMS and OJVM Time Zone File Patches (ドキュメントID 412160.1)

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。

参照:

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 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)は、Oracle Database 12c以降のリリースでOracle XML DB用のポートを構成しません。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薦めします。

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. Use `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);
   ```

   ノート:

   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`を問い合せます。

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Oracle Textが提供するナレッジ・ベースのインストール

Oracle Textナレッジ・ベースに対してユーザーが拡張した項目は、Oracle Databaseアップグレード後に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。

Oracle Textが提供するナレッジ・ベースは新しいOracle Databaseの付属製品の一部ですが、アップグレード後すぐに使用することはできません。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。

参照:

- 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インストレーション・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

以前のリリースのOracle APEXの削除

無効なオブジェクトを回避するには、Application Express (APEX) 19.1より前のOracle APEXリリースを削除します。

パッケージDBMS_OBFUSCATION_TOOLKITはOracle Database 11gリリース2 (11.2)で非推奨になりました。Oracle Database 21c以降、DBMS_OBFUSCATION_TOOLKITはサポート対象外になり、DBMS_CRYPTに置き換えられました。Oracle Database 21cより前のリリースでは、Oracle APEXは継続してDBMS_OBFUSCATION_TOOLKITに依存していました。APEXリリース19.2を含むOracle Database 21cでは、この依存性が削除されました。

Oracle Databaseのアップグレード中、APEXはOracle Databaseリリースに付属のリリースに自動的にアップグレードされません。アップグレード前に、preupgradeパラメータを使用してAutoUpgradeを実行すると、出力upgrade.xmlファイルに、APEX 19.1.0.00.15より前のAPEXリリースをアップグレードする必要があることが示されます。これは、以前のリリースがDBMS_OBFUSCATION_TOOLKITに依存しているためです。

INVALIDオブジェクトを回避するには、Oracle Databaseをアップグレードする前または後のいずれかで、APEXを少なくともOracle Database 21cに付属するバージョンにアップグレードしてから、APEXの以前のリリースを削除する必要があります。

1. 新しいOracle DatabaseのOracleホームにログインします
2. Oracle APEXを、Oracleホーム(APEX 19.2)に同梱されているOracle Application Express 19.2以上、またはそれ以降のバージョンが使用可能な場合は、それを使用してOracle Application Expressをアップグレードします
3. 以前のリリースのAPEXユーザーを削除します。

   たとえば:

   ```
   drop user APEX_050000 cascade;
   drop user APEX_040200 cascade;
   drop user APEX_030100 cascade;
   ```

4. 以前のリリースのAPEX SYSが所有するオブジェクトを削除します。

   たとえば:

   ```
   drop package SYS.WWV_DBMS_SQL;
   ```

> ノート:
>
> Oracle APEX 18c (18.1)以降では、SYS.WWV_DBMS_SQL オブジェクトには Oracle APEX リリース・スキーマが追加されています。
>
> たとえば:
>
> ```
> SYS.WWV_DBMS_SQL_APEX_180100
> ```

親トピック: Oracle Databaseのアップグレード後に必要な作業

読取り専用OracleホームでのDEMOディレクトリの置換

読取り専用Oracleホームをアップグレードした後、以前のリリースのOracle Databaseのdemoディレクトリをコピーし、読取り専用Oracleホームのdemoディレクトリを新しいリリースのdemoディレクトリに置き換えます。

Oracle Database 18c以上のリリースでは、ファイル・パスOracle_home/rdbms/demoに製品デモンストレーション・ディレクトリが含まれています。これらのディレクトリには、各Oracle Databaseリリースのオプションと機能に固有の例と製品デモンストレーションが含まれています。これらの一部は、Oracle Database Examplesをインストールすると、アップグレード後に追加できます。以前のリリースでは、以前のリリースのデモンストレーション・ファイルを使用してダウンロードして作業した場合、2つの

問題があります。新しいリリースでレビューおよびテストするために以前のリリースの作業を保存する必要があること、および新しいリリースに固有のデモンストレーションのリフレッシュを取得する必要があることです。

Oracleホームをアップグレードした後、新しいデモンストレーション・ファイルで実行する他の作業をダウンロードして実行したら、古いデモンストレーション・ファイルをリフレッシュできます。

例6-9 以前のリリースのDemoディレクトリのコピーおよび読取り専用Oracleホームでのデモンストレーションのリフレッシュ

アップグレード後に、この手順を使用して、以前のdemoディレクトリの作業を読取り専用Oracleホームに保存し、以前のリリースのdemoディレクトリを新しいリリースのdemoディレクトリで置き換えます。

1. Oracleソフトウェア所有者ユーザー(oracle)としてログインします。
2. rdbms/demoディレクトリが読取り専用Oracleホームにコピーされているかどうかを確認します。

   この例では、環境変数ORACLE_BASE_HOMEが読取り専用Oracleホームのパスとして定義されています。

   LinuxおよびUNIXプラットフォーム:

   ```
   $ ls -l -d $ORACLE_BASE_HOME/rdbms/demo
   /u01/app/oracle/product/19.0.0/dbhome_1/rdbms/demo
   ```

   Microsoft Windowsプラットフォーム

   ```
   ls -l -d %ORACLE_BASE_HOME%¥rdbms¥demo
   %ORACLE_BASE_HOME%¥rdbms¥demo
   ```

3. ディレクトリを読取り専用Oracleホームに変更し、コピーを作成します。ここで、demo.old_release18は以前のリリースのデモンストレーション・ファイルに指定する名前です。

   ```
   cd $ORACLE_BASE_HOME/rdbms
   mv demo demo.old_release18
   ```

4. 新しいdemoディレクトリをアップグレードされたOracleホームから読取り専用Oracleホームにコピーします。

   この例では、環境変数ORACLE_HOMEが新しいリリースのOracleホームとして定義されています。

   LinuxおよびUNIX:

   ```
   cp -r $ORACLE_HOME/rdbms/demo demo
   ```

   Microsoft Windows

   ```
   xcopy c:¥%ORACLE_HOME%¥rdbms¥demo c:%ORACLE_BASE_HOME%¥rdbms¥demo /E
   ```

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ングレイン・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。

参照:

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

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
  Oracle Database 12cリリース2 (12.2.0.1)以降のリリースにアップグレードするには、ルート・コンテナで`DV_PATCH_ADMIN`ロールをSYSに共通に付与してアップグレード後に取り消すか、アップグレード後にOracle Database Vaultを無効にして再度有効にします。
- 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のアップグレード後に必要な作業

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

Oracle Database 12cリリース2 (12.2.0.1)以降のリリースにアップグレードするには、ルート・コンテナで`DV_PATCH_ADMIN`ロールをSYSに共通に付与してアップグレード後に取り消すか、アップグレード後にOracle Database Vaultを無効にして再度有効にします。

Oracle Database Vaultが有効になっており、CDB全体をアップグレードする場合は、次のいずれかの方法を使用します。

- CDBアップグレード方法1: `DV_OWNER`ロールを持つ共通ユーザーとしてルート・コンテナにログインしてから、`GRANT DV_PATCH_ADMIN TO SYS CONTAINER=ALL`文を発行することで、`DV_PATCH_ADMIN`をユーザーSYSに一時的に付与します。Oracle Database Vaultの制御は、アップグレード前と同じ状態になります。アップグレードが完了した後は、`DV_OWNER`ユーザーとしてルート・コンテナにログインし、`REVOKE DV_PATCH_ADMIN FROM SYS CONTAINER=ALL`文を発行してSYSから`DV_PATCH_ADMIN`ロールを取り消します。
- CDBアップグレード方法2: `DV_OWNER`ロールを持つユーザーとして各コンテナにログインしてから、`DBMS_MACADM.DISABLE_DV`プロシージャを実行します。最初にPDBでOracle Database Vaultを無効にしてから、最後にルート・コンテナでOracle Database Vaultを無効にする必要があります。PDBを1つのみアップグレードしている場合は、そのPDBでのみOracle Database Vaultを無効にできます。アップグレードが完了した後は、`DV_OWNER`ユーザーとして各コンテナにログインしてから`DVSYS.DBMS_MACADM.ENABLE_DV`プロシージャを実行することで、Oracle Database Vaultを有効にできます。Oracle Database Vaultを有効にする順序は、最初にルート・コンテナで、その後にPDBで有効にする必要があります。PDBは任意の順序で有効にできますが、最初にルート・コンテナを有効にする必要があります。

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。

ノート:

この手順は、非 CDB アップグレードにも適用されます

関連項目

- Oracle Database - 12.2.0.1以降のデータベース・パッチ配信方法の概要(ドキュメントID 2337415.1)
- Oracle Database Vaultの無効化および有効化Oracle Database Vault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じ強化ステータスになります。

表6-1 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 以下	12.1	可	無効。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 以下	12.2、18.1 以上	いいえ	無効。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 以上	12.2、18.1 以上	いいえ	Database Vault はアップグレード前に用意したものと同じ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

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以上では、`SQLNET.ALLOWED_LOGON_VERSION`パラメータのデフォルト値がOracle Database 12c (12.2)の11からOracle Database 18c以上のリリースの12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、リリース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以上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。

参照:

- 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』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薦めします。特に明記されている場合を除き、これらの方法はすべてのタイプのアップグレードにお薦めします。

- Databaseのバックアップ
  少なくともレベル1のバックアップを実行するか、時間が許せばレベル0のバックアップを実行することをお薦めします。
- アップグレード後修正スクリプトの実行
  CDBおよび非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ためには、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薦めします。
- 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。
- 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。
- Oracle Grid Infrastructure、Oracle ASMおよびOracle Clusterwareの理解
  Oracle ClusterwareとOracle Automatic Storage Management (Oracle ASM)は、両方ともOracle Grid Infrastructureインストールの一部です。
- Oracle Grid InfrastructureのインストールとアップグレードおよびOracle ASM
  Oracle ASMは、Oracle Grid Infrastructureとともにインストールされます。
- 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、アップグレードしたOracle Databaseの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。
- 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。
- アップグレードしたOracle Databaseの統合監査の使用への移行
  従来の監査は非推奨です。統合監査の全機能を使用するには、手動で統合監査に移行する必要があります。

- 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。
- DBMS_SCHEDULERジョブの削除および再作成
  以前のリリースからのアップグレード後にDBMS_SCHEDULERジョブが機能しなくなった場合は、ジョブを削除して再作成します。
- アップグレード後の統合監査レコードの転送
  アップグレードし、統合監査に移行した後で高いパフォーマンスを得る方法を理解するためには、これらのトピックを確認します
- アップグレードした本番Oracle Databaseのテストについて
  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

少なくともレベル1のバックアップを実行するか、時間が許せばレベル0のバックアップを実行することをお薦めします。

関連項目

- データベースのバックアップ

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

アップグレード後修正スクリプトの実行

CDBおよび非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ためには、この手順を確認します。

> ノート:
>
> AutoUpgrade ユーティリティを使用してアップグレードした場合、AutoUpgrade は固定オブジェクトの統計をアップグレード中に自動的に更新します。このタスクを実行する必要はありません。

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。プラガブル・データベース(PDB)を持つコンテナ・データベース(CDB)と非CDBデータベースの両方で、アップグレード後修正スクリプトを実行することによって、一般的な警告、エラーおよび推奨事項に関する情報が提供されます。

`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いなくても、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
- CDB: 2つの異なるスクリプト・セット:
  1. postupgrade_fixups.sql: すべてのPDB用の統合スクリプト
  2. 複数のpostupgrade_fixups_pdbname.sqlスクリプト。pdbnameは、スクリプト生成の対象となるPDBの名前です。個々のスクリプトを特定のPDBで実行します。

例6-10 非CDB Oracle Databaseのアップグレード後修正結果のスプーリングの例

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> SPOOL postupgrade.log
SQL> @postupgrade_fixups.sql
SQL> SPOOL OFF
```

スクリプト結果のログ・ファイルへのスプーリングをオフにします。

```
SQL> SPOOL OFF
```

例6-11 Linux、UNIXおよびWindowsシステムでcatcon.plを使用してアップグレード後修正を実行する方法の例

この項の例では、catconコマンドを使用して、CDBデータベースのすべてのコンテナでpostupgrade_fixups.sqlを実行しています。このコマンドを実行する前に、UNIXおよびLinux環境に対してオペレーティング・システム環境変数ORACLE_HOMEおよびORACLE_SIDが設定されていて、同等の設定がWindows環境に対しても行われていることを確認する必要があります。次の例では、sales1はターゲット・データベースの一意の名前です。

LinuxおよびUNIX環境:

```
$ORACLE_HOME/perl/bin/perl -I$ORACLE_HOME/perl/lib -I$ORACLE_HOME/rdbms/admin
$ORACLE_HOME/rdbms/admin/catcon.pl -l /home/oracle/here/ -b postupg
/cfgtoollogs/sales1/preupgrade/postupgrade_fixups.sql
```

Windows環境:

```
%ORACLE_HOME%¥perl¥bin¥perl –I%ORACLE_HOME%¥perl¥lib –
I%ORACLE_HOME%¥rdbms¥admin %ORACLE_HOME%¥rdbms¥admin¥catcon.pl -l c:¥tmp¥logdir¥ -b
postupg c:¥cfgtoollogs¥sales1¥preupgrade¥postupgrade_fixups.sql
```

このWindowsの例では、コマンド・オプション-lを使用して、ファイル・パスc:/tmp/logdir/の下にログを作成し、アップグレード後スクリプトの出力をそのディレクトリに配置しています。オプション-bを使用して、出力ファイルに接頭辞postupgを設定しています。

ノート:

catcon のパラメータは、これらの例に示されている順序で入力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

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薦めします。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。

ノート:

AutoUpgrade ユーティリティを使用してアップグレードを完了した場合は、このタスクを完了する必要はありません。AutoUpgrade ユーティリティによって自動的に完了します。

- 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薦めします。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薦めします

  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。

関連項目

- 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薦めします。

ノート:

パフォーマンス・チューニングのための最も正しい固定オブジェクト統計を提供するために、システムが代表的なワークロードを実行している時点で、ベースライン統計を収集することをお薦めします。役立つ結果を得るために、アップグレードの直後には DBMS_STATS.GATHER_FIXED_OBJECTS_STATS を実行しないでください。

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す。これは、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で、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。

データベースで代表的なワークロードが実行されていることを確認し、DBMS_STATS.GATHER_FIXED_OBJECTS_STATS PL/SQLプロシージャを使用して固定オブジェクト統計を収集します。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$表は一時的な性質を持つため、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薦めします。

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;
```

関連項目

- データベース統計の収集

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薦めします。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。

参照:

- 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・ビューに問い合せて、ユーザー・アカウント用に構成されたパスワード・バージョンすべてのリストを出すことができます。

たとえば:

```
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のときに、Oracle Database 12cリリース12.1で再設定されました。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当初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、その時点でデフォルトのSQLNET.ALLOWED_LOGON_VERSION設定の8を使用していました。大/小文字を区別しないこと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/小文字が区別されるようになりました。
- 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たものです。

- ユーザーblake: このアカウントはまだOracle Database 10gパスワード・バージョンを使用しています。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パスワードを削除します。次の手順で、10Gパスワード・バージョンを持つユーザーのパスワードを再設定するには、ログインが許可される前のクライアントの機能レベル要件を制御する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、それらのユーザーがログインしてパスワードを変更し、10Gパスワード・バージョンに加えてより新しいパスワード・バージョンを生成できるようになります。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。そうすれば、パスワードにもう10G,が含まれず、より安全な11Gおよび12Cパスワード・バージョンだけを持つように、ユーザーはもう一度パスワードを再設定できます。

1. DBA_USERSビューに問い合せて、10Gパスワード・バージョンだけを使用するユーザーを見つけます。

```
SELECT USERNAMEFROM DBA_USERS
WHERE ( PASSWORD_VERSIONS = '10G '
OR PASSWORD_VERSIONS = '10G HTTP ')
AND USERNAME <> 'ANONYMOUS';
```

2. 次のようにして、データベースが排他モードで実行されないように構成します。
   a. sqlnet.oraファイルのSQLNET.ALLOWED_LOGON_VERSION_SERVER設定を、デフォルトよりも許容度が高くなるように編集します。たとえば:

   ```
   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。

   11Gまたは12C、あるいはその両方のパスワード・バージョンを持つユーザーではなく、10Gパスワード・バージョンだけを持つユーザーを期限切れにする必要があります。

   たとえば:

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

4. 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、欠落している11Gおよび12Cパスワード・バージョンを生成します。データベースは許容モードで実行中のため、10Gパスワード・バージョンは引き続き存在します。

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O5L_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O5L_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O5L_NP機能が導入されたら、次のようにしてサーバーのセキュリティを元の排他モードに設定します。
   a. インスタンス初期化ファイルからSEC_CASE_SENSITIVE_LOGONパラメータを削除するか、またはSEC_CASE_SENSITIVE_LOGONをTRUEに設定します。

   ```
   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ファイルでSQLNET.ALLOWED_LOGON_VERSION_SERVERの値を12に設定しなおして、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・バージョンを持っているアカウントを見つけます。

```
SELECT USERNAME FROM DBA_USERS
WHERE PASSWORD_VERSIONS LIKE '%10G%'
AND USERNAME <> 'ANONYMOUS';
```

8. まだ10Gパスワード・バージョンを持っ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。そうすると、データベースはそのアカウントに11Gおよび12Cパスワード・バージョンだけを生成します。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・バージョンはもう生成されません。

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';
```

   この問合せで返された結果が0件だった場合は、10Gパスワード・バージョンを持つユーザー・アカウントがないことを意味します。したがって、データベースは以前のリリースよりもよりセキュアなモードで実行されています。

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 Grid Infrastructure、Oracle ASMおよびOracle Clusterwareの理解

Oracle ClusterwareとOracle Automatic Storage Management (Oracle ASM)は、両方ともOracle Grid Infrastructureインストールの一部です。

Oracle Grid Infrastructureは、単一サーバーにインストールした場合、Oracle ASMとともにOracle Restartインストールとしてデプロイされます。クラスタにインストールした場合は、Oracle ASMとともにOracle Clusterwareインストールとしてデプロイされます。

Oracle Restartは、単一インスタンス環境におけるOracle Databaseの可用性を向上します。Oracle Restartをインストールし、データベース、リスナー、Oracle ASMインスタンスなどのOracle Databaseソフトウェア・スタックの一部が一時的に失敗した場合、失敗したコンポーネントはOracle Restartによって自動的に再起動されます。また、これらのすべてのコンポーネントは、データベース・ホスト・コンピュータが再起動されるときに、Oracle Restartによって起動されます。コンポーネントは、コンポーネント間の依存性を考慮して適切な順序で起動されます。

Oracle Clusterwareは、単一サーバーを単一システムとして連携するようにクラスタ化できるポータブル・クラスタ・ソフトウェアです。Oracle RACに必須なインフラストラクチャも提供します。また、Oracle Clusterwareでは、クラスタ内の任意のOracleアプリケーションまたはその他のアプリケーションを保護できます。いずれの場合でも、Oracle Clusterwareはそれらの

システムにおけるインテリジェント機能としてクラスタ・ノード間の必要な連携を確実にします。

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 Grid Infrastructureのインストールとアップグレードおよび Oracle ASM

Oracle ASMは、Oracle Grid Infrastructureとともにインストールされます。

以前のリリースまでは、Oracle ASMは、Oracle Databaseインストールの一部としてインストールされていました。Oracle Databaseリリース11.2以上では、Oracle ASMは、Grid Infrastructureコンポーネントのインストール時にインストールされます。Oracle ASMは、Oracle ClusterwareとOracleホームを共有します。

参照:

Oracleホーム、ロール割当て済のシステム権限グループ、様々なインストール・ソフトウェア所有者ユーザー、および他の変更の詳細は、使用しているプラットフォームの『Oracle Grid Infrastructureインストレーション・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

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。

参照:

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シージャの再確認を計画し、必要に応じて変更します。

新しいOracle Database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シージャを再確認し、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、アップグレードしたOracle Databaseの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 Database 18c以上のリリースでは、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に役立ちます。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

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。

参照:

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

アップグレードしたOracle Databaseの統合監査の使用への移行

従来の監査は非推奨です。統合監査の全機能を使用するには、手動で統合監査に移行する必要があります。

統合監査と従来の監査(混合モード)は、Oracle Database 12c以降デフォルトの監査モードでした。混合モードの監査は、統合監査に慣れるためと、従来の監査から移行するために提供されていました。Oracle Database 21cで従来の監査が非推奨になったため、統合監査に移行することをお薦めします。『Oracle Databaseセキュリティ・ガイド』の移行手順を参照してください。

- Oracle Databaseでの統合監査の移行プロセスの理解
  アップグレードしたデータベースで使用する監査ポリシーを決定します。
- Oracle Databaseでの統合監査への移行
  マルチテナント・コンテナ(CDB)データベースでこの手順を使用して、統合監査に移行します。
- 統合監査への移行後の古い監査レコードの管理について
  統合監査証跡を使用する準備として古い監査証跡を確認、アーカイブおよび削除します。
- 統合監査機能の削除
  この手順を使用して、統合監査を削除し、混合モードの監査を使用します。
- 統合監査を使用しない場合のドキュメント参照の取得
  ここにリストされているドキュメントにアクセスして、非統合監査の使用方法に関する構成情報を取得できます。

関連項目

- 統合監査の移行による各監査機能への影響

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 Databaseでの統合監査の移行プロセスの理解

アップグレードしたデータベースで使用する監査ポリシーを決定します。

デフォルトでは、アップグレードされたデータベースでは、統合監査は有効化されていません。以前のリリースからOracle Database 12cアップグレードした場合は、データベースでは以前のリリースで使用していたものと同じ監査機能が使用されています。新しく作成されたデータベースでは、統合監査の混合モードの方式がデフォルトで有効化されます。統合監査への移行が完了すると、従来の監査は無効化され、新しい監査レコードが統合監査証跡に書き込まれます。

監査ポリシーを有効化して使用方法を構成するには、次のように1つの方法を選択します。

- 純統合監査機能を使用。

  統合監査に移行して、統合監査の全機能を使用します。統合監査への移行手順が完了すると、新しい監査ポリシーを作成および有効化でき、事前定義された監査ポリシーも使用できます。これらのポリシーの監査レコードは、統合監査証跡に書き込まれます。以前の監査証跡および監査レコードは保持されますが、新しい監査レコードは以前の監査証跡には書き込まれません。

  ノート:

  以前のリリースの監査構成は、統合監査システムでは効果がありません。統合監査証跡内では、統合監査ポリシーのみが監査レコードを生成します。

- 混合モードの監査機能を使用。

  混合モードの監査機能は、従来の監査機能と統合監査機能の両方を同時に実行でき、新しいデータベースとアップグレードされたデータベースの両方に適用されます。混合モードの統合監査機能は、1つ以上の事前定義された統合監査ポリシーを有効化すると使用可能になります。これらのポリシーの監査レコードは、統合監査証跡に書き込まれます。Oracle Databaseの以前のリリースでの監査構成も使用可能で、この構成の監査レコードは以前の監査証跡に書き込まれます。純統合監査機能を使用することに決定した場合、それに移行できます。

  ノート:

  データベースが書込み不可な場合、監査レコードは、`$ORACLE_BASE/audit/$ORACLE_SID` ディレクトリにある新しい形式のオペレーティング・システム・ファイルに書き込まれます。

  参照:

  - 事前定義された監査ポリシーの詳細は、『Oracle Databaseセキュリティ・ガイド』を参照してください。
  - ora_SecureConfig監査ポリシーの詳細は、『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: アップグレードしたOracle Databaseの統合監査の使用への移行

Oracle Databaseでの統合監査への移行

マルチテナント・コンテナ・データベース(CDB)でこの手順を使用して、統合監査に移行します。

CDB環境で、`root`で次の手順を実行します。この手順によって、`root`CDBとそれに関連するPDBの両方が統合監査に移

行されます。

1. SYSDBA権限を持つユーザーSYSとしてSQL*Plusにログインします。

```
sqlplus sys as sysdba
Enter password: password
```

プラガブル・データベース環境では、このログインにより、rootに接続されます。

2. 次の問合せを使用して、使用しているOracle Databaseが統合監査に移行したかどうかを確認します。

```
SQL> SELECT VALUE FROM V$OPTION WHERE PARAMETER = 'Unified Auditing';
```

VALUE列の出力がTRUEの場合、データベースで統合監査がすでに有効化されています。以前の監査レコードの管理に進むことができます。出力がFALSEの場合は、この手順の残りのステップを完了してください。

3. データベースを停止します。単一インスタンス環境では、SQL*Plusから次のコマンドを入力します。

```
SQL> SHUTDOWN IMMEDIATE
SQL> EXIT
```

Windowsシステムでは、Oracleサービスを停止します。

```
net stop OracleService%ORACLE_SID%
```

Oracle RACインストールの場合は、次のようにして、各データベース・インスタンスを停止します。

```
srvctl stop database -db db_name
```

4. リスナーを停止します。(Oracle RACおよびOracle Grid Infrastructureリスナーでは、リスナーを停止する必要はありません。)

```
lsnrctl stop listener_name
```

lsnrctl statusコマンドを実行すると、リスナーの名前を検索できます。名前はAlias設定で示されます。

5. ディレクトリ$ORACLE_HOME/rdbms/libに移動します。
6. Oracleユーザーの統合監査を有効にします。
   - LinuxおよびUNIX

     ```
     make -f ins_rdbms.mk uniaud_on ioracle ORACLE_HOME=$ORACLE_HOME
     ```

   - Windows

     ファイル名を%ORACLE_HOME%/bin/orauniaud12.dll.dblから%ORACLE_HOME%/bin/orauniaud12.dllに変更します。

> ノート:
>
> Oracle ホームを共有していない Oracle RAC データベースの場合、各クラスタ・メンバー・ノードでこのステップを繰り返して、各クラスタ・ノードのローカル ORACLE_HOME 内のバイナリが更新されるようにする必要があります。

7. リスナーを再起動します。

```
lsnrctl start listener_name
```

8. データベースを再起動します。

   SQL*Plusにログインしてから、STARTUPコマンドを入力します。

```
sqlplus sys as sysoper
Enter password: password
SQL> STARTUP
```

   Windowsシステムでは、Oracleサービスを起動します。

```
net start OracleService%ORACLE_SID%
```

   Oracle RACインストールの場合は、各データベース・インスタンスを起動します。

```
srvctl start database -db db_name
```

親トピック: アップグレードしたOracle Databaseの統合監査の使用への移行

統合監査への移行後の古い監査レコードの管理について

統合監査証跡を使用する準備として古い監査証跡を確認、アーカイブおよび削除します。

統合監査を使用するためのOracle Databaseの移行手順が完了すると、データベースで使用していた以前の監査レコードは以前の監査証跡に保持されます。これらの監査レコードをアーカイブして、監査証跡を削除できます。統合監査が使用可能な場合、新しい監査レコードは統合監査証跡に書き込まれます。

参照:

- 『Oracle Databaseセキュリティ・ガイド』の監査証跡のアーカイブに関する説明
- 『Oracle Databaseセキュリティ・ガイド』の監査証跡レコードの削除に関する説明

親トピック: アップグレードしたOracle Databaseの統合監査の使用への移行

統合監査機能の削除

この手順を使用して、統合監査を削除し、混合モードの監査を使用します。

データベースで統合監査を使用できるようにした後で、統合監査を使用しないことに決定した場合、この手順を使用して、統合監査機能を削除できます。この場合、データベースでは混合モードの監査機能が使用されます。

1. データベースを停止します。

```
sqlplus sys as sysoper
Enter password: password
SQL> SHUTDOWN IMMEDIATE
SQL> EXIT
```

   Windowsシステムでは、Oracleサービスを停止します。

```
net stop OracleService%ORACLE_SID%
```

   Oracle RACインストールの場合は、次のようにして、各データベース・インスタンスを停止します。

```
srvctl stop database -db db_name
```

2. `$ORACLE_HOME/rdbms/lib`ディレクトリに移動します。

3. 統合監査実行可能ファイルを無効にします。

   - Unix: 次のコマンドを実行します。

     ```
     make -f ins_rdbms.mk uniaud_off ioracle ORACLE_HOME=$ORACLE_HOME
     ```

   - Microsoft Windows: `%ORACLE_HOME%/bin/orauniaud12.dll`ファイルの名前を`%ORACLE_HOME%/bin/orauniaud12.dll.dbl`に変更します。

4. データベースを再起動します。

   ```
   sqlplus sys as sysoper
   Enter password: password
   SQL> STARTUP
   SQL> EXIT
   ```

   Microsoft Windowsシステムでは、Oracleサービスを再度起動します。

   ```
   net start OracleService%ORACLE_SID%
   ```

   Oracle RACインストールの場合は、次の構文を使用して、各データベース・インスタンスを起動します。

   ```
   srvctl start database -db db_name
   ```

親トピック: アップグレードしたOracle Databaseの統合監査の使用への移行

統合監査を使用しない場合のドキュメント参照の取得

ここにリストされているドキュメントにアクセスして、非統合監査の使用方法に関する構成情報を取得できます。

新しいOracle Databaseリリースへのアップグレード後に、統合監査に変更しないことを選択した場合、OracleドキュメントおよびOracle Technology Networkで従来の非統合監査に関する情報が提供されています。

- *Oracle Database*セキュリティ・ガイド: このガイドは、監査を構成する際の主要な情報源です。このマニュアルのOracle Databaseリリース11gバージョンを使用する必要があります。このガイドにアクセスするには:

  1. 次のOracle Technology Networkの`docs.oracle.com`サイトで、データベース・ページにアクセスします。

     https://docs.oracle.com/en/database/index.html

  2. 「Oracle Database」を選択します。
  3. 「ダウンロード」ページで、「ドキュメント」タブを選択します。
  4. リリース・リスト・フィールドで「Earlier Releases」を選択し、「Oracle Database 11g Release 2 (11.2)」を選択します。
  5. Oracle Database 11gリリース2 (11.2)の「Documentation」ページから、「All Books」リンクを選択して、ドキュメント・セットの刊行物を表示します。
  6. 「セキュリティ・ガイド」を検索します。
  7. このガイドの「HTML」または「PDF」リンクのいずれかを選択します。

親トピック: アップグレードしたOracle Databaseの統合監査の使用への移行

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

DBMS_SCHEDULERジョブの削除および再作成

以前のリリースからのアップグレード後にDBMS_SCHEDULERジョブが機能しなくなった場合は、ジョブを削除して再作成します。

アップグレード後にDBMS_SCHEDULERジョブが機能しないことに気づいたら、それらのジョブを削除して再作成してください。この問題は、アップグレード処理からの問題報告がなく、システム・オブジェクトが有効な場合でも発生する可能性があります。

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

アップグレード後の統合監査レコードの転送

統合監査をアップグレードし、移行した後で高いパフォーマンスを得る方法を理解するためには、これらのトピックを確認します。

- アップグレード後の監査レコードの転送について
  統合監査レコードをOracle Database 12cリリース12.1から、新しいOracle DatabaseリリースのAUDSYSスキーマ下の新しいリレーショナル表に転送すると、統合監査証跡の読取りパフォーマンスが向上します。
- アップグレード後の統合監査レコードの転送
  DBMS_AUDIT_MGMT.TRANSFER_UNIFIED_AUDIT_RECORDS PL/SQLプロシージャを使用して、統合監査レコードをAUDSYSの新しいリレーショナル表に転送できます。

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

アップグレード後の監査レコードの転送について

統合監査レコードをOracle Database 12cリリース12.1から、新しいOracle DatabaseリリースのAUDSYSスキーマ下の新しいリレーショナル表に転送すると、統合監査証跡の読取りパフォーマンスが向上します。

Oracle Database 12cリリース2以降、統合監査レコードは直接、AUDSYSスキーマにある新しい内部リレーショナル表に書き込まれます。Oracle Database 12cリリース12.1では、統合監査レコードは共通ロギング・インフラストラクチャ(CLI) SGAキューに書き込まれていました。そのリリースの統合監査に移行した場合は、より高い読取りパフォーマンスを得るために、そのリリースの統合監査レコードを新しいOracle Databaseリリースの内部表に転送できます。この転送を実行することは必須ではありませんが、統合監査証跡の読取りパフォーマンスを向上させるため、転送をお薦めします。これは1回かぎりの操作です。アップグレード後に生成されたすべての新しい統合監査レコードはこの新しい表に書き込まれます。この表は、読取り専用の表です。

この表のメタデータまたはデータを変更しようとすると、それらはすべて強制的に監査されます。

新しいOracle Databaseリリースにアップグレードした後に、以前のリリースのUNIFIED_AUDIT_TRAILに統合監査レコードが存在している場合は、この転送プロシージャを使用してそれらを新しい内部リレーショナル表に転送し、統合監査証跡の読取りパフォーマンスを高めることを検討してください。

SYSスキーマの場合と同様、SELECT ANY TABLEシステム権限がある場合はAUDSYSスキーマの問合せはできません。さらに、この表はユーザーがSELECT ANY DICTIONARYシステム権限か、この内部表に対する明示的なSELECT権限を持っていないかぎり、ALL_TABLESデータ・ビューでスキーマ・オブジェクトとしてリストされません。データベースが読取り/書込みモードでオープンするまで、監査レコードはオペレーティング・システムの過剰ファイルに書き込まれます(.bin形式)。ただし、データベースが読取り/書込みモードでオープンした後に、DBMS_AUDIT_MGMT.LOAD_UNIFIED_AUDIT_FILESプロシージャを使用してこれらのオペレーティング・システム・ファイルの監査レコードを内部リレーショナル表に転送することはできません。

親トピック: アップグレード後の統合監査レコードの転送

アップグレード後の監査レコードの転送

DBMS_AUDIT_MGMT.TRANSFER_UNIFIED_AUDIT_RECORDS PL/SQLプロシージャを使用して、統合監査レコードをAUDSYSの新しいリレーショナル表に転送できます。

1. AUDIT_ADMINロールを付与されたユーザーとして、データベース・インスタンスにログインします。

   たとえば、非マルチテナント環境では、次のとおりです。

   ```
   sqlplus sec_admin
   Enter password: password
   ```

   マルチテナント環境では、rootとして接続します。

   ```
   sqlplus c##sec_admin@root
   Enter password: password
   ```

   このプロシージャの実行はPDBだけではなくrootでも実行できます。これは、UNIFIED_AUDIT_TRAILビューがコンテナ別であるからです。さらに、転送手順もコンテナ別です。つまり、rootからの転送を実行してもPDBの統合監査証跡にある統合監査レコードには影響がありません。

2. マルチテナント環境ではDBA_PDB_HISTORYビューに問い合せて、監査レコードの転送元のコンテナ固有のCLI表に関連付けられた正しいGUIDを見つけます。

   たとえば:

   ```
   SQL> SELECT PDB_NAME, PDB_GUID FROM DBA_PDB_HISTORY;
   PDB_NAME  PDB_GUID
   --------  --------------------------------
   HR_PDB    33D96CA7862D53DFE0534DC0E40A7C9B
   ...
   ```

3. マルチテナント環境の、監査レコードの転送を行うコンテナに接続します。

   現在接続しているコンテナとは別のコンテナでは転送操作を実行できません。

4. DBMS_AUDIT_MGMT.TRANSFER_UNIFIED_AUDIT_RECORDSプロシージャを実行します。
   たとえば:

   ```
   SQL> EXEC DBMS_AUDIT_MGMT.TRANSFER_UNIFIED_AUDIT_RECORDS;
   PL/SQL procedure successfully completed.
   ```

または、PDB GUIDを指定するために次を実行します。

```
SQL> EXEC DBMS_AUDIT_MGMT.TRANSFER_UNIFIED_AUDIT_RECORDS
('33D96CA7862D53DFE0534DC0E40A7C9B');
PL/SQL procedure successfully completed.
```

5. データベースが読取り/書込みモードでオープンしている場合は、DBMS_AUDIT_MGMT.LOAD_UNIFIED_AUDIT_FILESプロシージャを実行します。

   データベースが読取り/書込みモードでオープンするまで、監査レコードはオペレーティング・システム(OS)のファイルに書き込まれます。DBMS_AUDIT_MGMT.LOAD_UNIFIED_AUDIT_FILESプロシージャは、そのファイルに存在する統合監査レコードをデータベース表に移動します。OS過剰ファイルに存在する統合監査レコードは、V$UNIFIED_AUDIT_TRAIL動的ビューの問合せにより見つけることができます。

   たとえば、このプロシージャをHR_PDBコンテナの監査レコードに実行する場合は、まず、そのPDBに接続する必要があります。

```
SQL> CONNECT sec_admin@HR_PDB
Enter password: password
SQL> EXEC DBMS_AUDIT_MGMT.LOAD_UNIFIED_AUDIT_FILES;
PL/SQL procedure successfully completed.
```

6. UNIFIED_AUDIT_TRAILデータ・ディクショナリに問い合せて、レコードが正しく転送されたことを確認します。

   問合せUNIFIED_AUDIT_TRAILを行うことを強くお薦めします。監査レコードが正常に転送されたら、問合せUNIFIED_AUDIT_TRAILを行う必要があります。これは、V$UNIFIED_AUDIT_TRAIL動的ビューの問合せではOSの過剰ファイルにある監査レコードしか表示されないからです。

親トピック: アップグレード後の統合監査レコードの転送

アップグレードした本番Oracle Databaseのテストについて

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・データベースを新しいOracle Databaseリリースにアップグレードしてテストした場合、新しいOracle Database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。

新しいOracle Database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

Oracle RACデータベースのアップグレード後の推奨作業

接続用にSCANまたはノード・リスナーを使用するようにクライアントを構成するかどうかを決定します。

Oracle Real Application Clusters 12cでは、単一クライアント・アクセス名(SCAN)が使用されます。SCANは、パブリック・ネットワークで3つのIPアドレスに解決される単一の名前です。11.2より前のリリースのOracle RACデータベースをアップグレードすると、Oracle RACデータベースがリモート・リスナーとしてSCANリスナーに登録されます。Oracle RACデータベースは、引き続きすべてのノード・リスナーにも登録されます。SCANリスナーには様々なメリットがあります。たとえば、クライアントを1回で構成してから、クライアントの接続構成を変更せずにクラスタを対象としてノードを追加または削除できます。

SCANを使用するようにクライアントを構成することも、クラスタ・メンバー・ノードで構成されているリスナーを使用し続けることもできます。SCANを使用するようにすべてのクライアント接続を移行する場合は、REMOTE_LISTENERSパラメータからノード・リスナーを削除できます。ただし、ノード・リスナーのみがデータベースの専用サーバーを作成できるため、ノード・リスナー自体を削除することはできません。

参照:

単一クライアント・アクセス名(SCAN)の詳細は、『Oracle Clusterware管理およびデプロイメント・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作業

Oracle ASMのアップグレード後の推奨作業

Oracle ASMのアップグレード後に、Oracle ASMパスワードのリセットおよびディスク・グループの構成などの作業を行うことをお薦めします。

- ASMディスク・グループでの共有パスワード・ファイルの作成
  COMPATIBLE.ASMディスク・グループ属性を拡張する場合は、共有パスワード・ファイルを作成します。
- Oracle ASMパスワードのリセットによる大/小文字区別の強制
  パスワードの大/小文字区別の強制を活用するには、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。
- Oracle ASMとOracle Databaseのディスク・グループの互換性の拡張
  ソフトウェア・バージョンをまたいでOracle DatabaseとOracle ASMのディスク・グループの互換性設定を拡張できます。
- Oracle ASMの優先読取りの障害グループの設定
  Oracle ASM管理者は、一部のディスクを読取りI/O操作のための優先読取りディスクとして指定できます。

関連項目

- 新機能の適宜追加
- 必要な新しい管理手順の作成

親トピック: Oracle Databaseのアップグレード後の作業

ASMディスク・グループでの共有パスワード・ファイルの作成

COMPATIBLE.ASMディスク・グループ属性を拡張する場合は、共有パスワード・ファイルを作成します。

COMPATIBLE.ASMディスク・グループ属性を12.1以降に拡張した場合、ASMディスク・グループに共有パスワード・ファイルを作成する必要があります。

参照:

ディスク・グループでの共有パスワード・ファイルの管理の詳細は、『Oracle Automatic Storage Management管理者ガイド』を参照してください

親トピック: Oracle ASMのアップグレード後の推奨作業

Oracle ASMパスワードのリセットによる大/小文字区別の強制

パスワードの大/小文字区別の強制を活用するには、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。

Oracle Database 11gリリース1 (11.1)より前のリリースでは、パスワードで大文字と小文字は区別されません。パスワードでの大/小文字の区別を強制できます。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。

新しいOracle ASMインスタンスの場合は、追加の作業や追加の管理要件はありません。アップグレードしたOracle ASMのインスタンスの場合は、各ユーザーのパスワードをALTER USER文でリセットする必要があります。

ノート:

Oracle Database 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。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 ASMのアップグレード後の推奨作業

Oracle ASMとOracle Databaseのディスク・グループの互換性の拡張

ソフトウェア・バージョンをまたいでOracle DatabaseとOracle ASMのディスク・グループの互換性設定を拡張できます。

注意:

COMPATIBLE.RDBMS 属性を拡張した場合、前の設定に戻ることはできません。COMPATIBLE.RDBMS 属性を拡張する前に、ディスク・グループを使用するすべてのデータベースの COMPATIBLE 初期化パラメータの値が少なくとも COMPATIBLE.RDBMS の新しい設定に設定されていることを確認します。

互換性を拡張することにより、新しいリリースのみで使用可能な新機能が有効になります。ただし、こうすることで、ソフトウェアの古いリリースではディスク・グループが非互換になります。ディスクの互換性を拡張する操作は、元に戻すことができません。

compatible.rdbmsおよびcompatible.asm属性を使用して、データベース・インスタンスおよびOracle ASMインスタンスからディスク・グループにアクセスするのに必要な最小ソフトウェア・リリースをそれぞれ指定します。たとえば、次のALTER DISKGROUP文によって、ディスク・グループasmdg2のOracle ASMの互換性が拡張されます。

```
ALTER DISKGROUP asmdg2 SET ATTRIBUTE 'compatible.asm' = '12.2'
```

この場合、ディスク・グループを管理できるのはリリース12.2以上のOracle ASMソフトウェアのみですが、データベース・クライアントはリリース11.2以上であれば、このディスク・グループを使用できます。

親トピック: Oracle ASMのアップグレード後の推奨作業

Oracle ASMの優先読取りの障害グループの設定

Oracle ASM管理者は、一部のディスクを読取りI/O操作のための優先読取りディスクとして指定できます。

ASM管理者がOracle ASMの優先読取りの障害グループを定義すると、Oracle ASMは、常にプライマリ・コピーを読み取るのではなく、最も近い優先読取りディスクに存在するエクステントから読み取ることができます。

参照:

- 拡張クラスタへの障害グループの設定については、『Oracle Clusterware管理およびデプロイメント・ガイド』を参照してください。
- Oracle ASMの優先読取りの障害グループ、および新しいASM_PREFERRED_READ_FAILURE_GROUPS初期化パラメータを指定してクラスタ内の各ノードに優先読取りディスクを含む障害グループ名をリストする方法の詳細は、『Oracle Automatic Storage Management管理者ガイド』を参照してください

- ASM_PREFERRED_READ_FAILURE_GROUPS初期化パラメータについては、『Oracle Databaseリファレンス』を参照してください。

親トピック: Oracle ASMのアップグレード後の推奨作業

Oracle Database Express Editionのアップグレード後の推奨作業

DBCAを使用するか、手動スクリプトを実行して、Oracle Databaseに追加のコンポーネントをインストールします。

Oracle Database Expressのデータベースに含まれているのは、Oracle Database Standard EditionまたはOracle Database Enterprise Editionのデータベースで使用できるコンポーネントのサブセットのみです。新しいOracle Databaseリリースにアップグレードした後、Database Configuration Assistant (DBCA)または手動スクリプトを使用して追加のコンポーネントをデータベースにインストールできます。

親トピック: Oracle Databaseのアップグレード後の作業

手動によるOracle Databaseのアップグレード後にのみ行う作業

DBUAを使用せずに手動でデータベースをアップグレードした場合、アップグレードの完了後に、ここに記載されている作業を実行する必要があります。

- Oracle付属アカウントのパスワードの変更
  これらの作業を実行して、新しいOracleユーザー・アカウントを保護することをお薦めします。
- サーバー・パラメータ・ファイルへの初期化パラメータ・ファイルの移行
  現在、従来の初期化パラメータ・ファイルを使用している場合、この手順を使用して、サーバー・パラメータ・ファイルに移行します。
- Oracle Textファイルを特定し、新しいOracleホームにコピーする
  Oracle Textをアップグレードするには、この手順を使用して必要なファイルを特定し、それを既存のOracleホームから新しいリリースのOracleホームにコピーします。このタスクはOracle Databaseをアップグレードした後で実行します。
- Oracle Clusterware構成のアップグレード
  Oracle Clusterwareを使用している場合は、データベースのOracle Clusterwareキーをアップグレードする必要があります。
- 新しいリリース用の初期化パラメータ・ファイルの調整
  アップグレード後に初期化パラメータをチェックするには、これらのトピックを参照してください。
- アップグレード後のOracle RACでのCLUSTER_DATABASE初期化パラメータの設定
  Oracle RACデータベース・インスタンスを手動でアップグレードする場合は、CLUSTER_DATABASE初期化パラメータを変更して、ノードを新しいリリースのクラスタに再度参加させる必要があります。

親トピック: Oracle Databaseのアップグレード後の作業

Oracle付属アカウントのパスワードの変更

これらの作業を実行して、新しいOracleユーザー・アカウントを保護することをお薦めします。

アップグレード元のリリースによっては、データベースに新しいOracleユーザー・アカウントが含まれる場合があります。SYSおよびSYSTEM以外のすべてのOracle付属アカウントは、ロックしてパスワードを期限切れにし、アカウントのロックを解除したときに新しいパスワードが要求されるようにすることをお薦めします。

ノート:

Oracle Database 12c 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。

参照:

パスワード要件については、『Oracle Databaseセキュリティ・ガイド』を参照してください

次のSQL文を発行して、すべてのアカウントの状態を確認できます。

```
SQL> SELECT username, account_status
```

```
        FROM dba_users
        ORDER BY username;
```

次のSQL文を発行して、パスワードをロックまたは期限切れにします。

```
SQL> ALTER USER username PASSWORD EXPIRE ACCOUNT LOCK;
```

親トピック: 手動によるOracle Databaseのアップグレード後にのみ行う作業

サーバー・パラメータ・ファイルへの初期化パラメータ・ファイルの移行

現在、従来の初期化パラメータ・ファイルを使用している場合、この手順を使用して、サーバー・パラメータ・ファイルに移行します。

1. 初期化パラメータ・ファイルがクライアント・コンピュータ上にある場合は、クライアント・コンピュータからサーバー・コンピュータに転送します。
2. `CREATE SPFILE`文を使用して、サーバー・パラメータ・ファイルを作成します。この文は、初期化パラメータ・ファイルを読み取り、サーバー・パラメータ・ファイルを作成します。`CREATE SPFILE`文を発行するために、データベースを起動する必要はありません。
3. 新しく作成されたサーバー・パラメータ・ファイルを使用して、インスタンスを起動します。

ノート:

Oracle Real Application Clusters (Oracle RAC)を使用している場合は、すべてのインスタンス固有の初期化パラメータ・ファイルを単一の初期化パラメータ・ファイルに結合する必要があります。クラスタ・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ために必要な各手順を完了してください。

関連項目

- Oracle RACでの初期化パラメータ・ファイルの概要

親トピック: 手動によるOracle Databaseのアップグレード後にのみ行う作業

Oracle Textファイルを特定し、新しいOracleホームにコピーする

Oracle Textをアップグレードするには、この手順を使用して必要なファイルを特定し、それを既存のOracleホームから新しいリリースのOracleホームにコピーします。このタスクはOracle Databaseをアップグレードした後で実行します。

一部のOracle Text機能は、ユーザーが構成したOracleホーム下のファイルを使用します。新しいOracle Databaseリリースへの手動アップグレードの後、または、Oracleホームを変更するすべてのプロセスの後に、これらのファイルを手動で特定し、移動する必要があります。これらのファイルにはユーザー・フィルタ、メール・フィルタ構成ファイルおよびナレッジ・ベース拡張ファイルなどが含まれます。これらのファイルを特定したら、既存のOracleホームから新しいOracleホームにコピーします。

必要なファイルを特定し、それを既存のOracleホームから新しいリリースのOracleホームにコピーするには:

1. アップグレードしたデータベースのSYS、SYSTEMまたはCTXSYSシステム権限でログインします。
2. アップグレードしたデータベースのOracleホームの下で、`$ORACLE_HOME/ctx/admin/ctx_oh_files.sql` SQLスクリプトを実行します。

   たとえば:

   ```
   sqlplus / as sysdba
   ```

```
connected
SQL> @?/ctx/admin/ctx_oh_files
```

3. `ctx_oh_files.sql`コマンドの出力を確認し、それらのファイルを新しいOracleホームにコピーします。

関連項目

- 『Oracle Textアプリケーション開発者ガイド』

親トピック: 手動によるOracle Databaseのアップグレード後にのみ行う作業

Oracle Clusterware構成のアップグレード

Oracle Clusterwareを使用している場合は、データベースのOracle Clusterwareキーをアップグレードする必要があります。

Oracle Database 12cの`srvctl`を実行してデータベースをアップグレードします。たとえば:

```
ORACLE_HOME/bin/srvctl upgrade database -db name -o ORACLE_HOME
```

関連項目

- Oracle Real Application Clusters管理およびデプロイメント・ガイド

親トピック: 手動によるOracle Databaseのアップグレード後にのみ行う作業

新しいリリース用の初期化パラメータ・ファイルの調整

アップグレード後に初期化パラメータをチェックするには、これらのトピックを参照してください。

Oracle Databaseの新しいリリースごとに新しい初期化パラメータが導入され、非推奨となったり、サポートが終了した初期化パラメータもあります。ご使用のシステムに有効な可能性のある新しい初期化パラメータを使用するために、これらの変更に対してパラメータ・ファイルを調整する必要があります。また、DBUAを使用しないで手動アップグレードを実行した場合、`tnsnames.ora`ファイルには新しい構成情報と設定が自動的に移入されません。したがって、手動で`tnsnames.ora`を更新し、`local_listener`および`remote_listener`パラメータ参照を調節する必要があります(これらを解決する必要がある場合)。

- アップグレード後のCOMPATIBLE初期化パラメータの設定
  テストの後に、COMPATIBLE初期化パラメータを、新しいデータベースに必要な互換性レベルに設定できます。
- アップグレード後のTNSNAMES.ORAおよびLISTENERパラメータの調整
  手動アップグレードの実行後、`tnsnames.ora`の`local_listener`および`remote_listener`を解決する必要がある場合、それらのパラメータを手動で調整する必要があります。

参照:

- 新しい初期化パラメータのリストおよび各パラメータの詳細は、Oracle Databaseリファレンスの「このリリースでの変更」を参照してください

親トピック: 手動によるOracle Databaseのアップグレード後にのみ行う作業

アップグレード後のCOMPATIBLE初期化パラメータの設定

テストの後に、COMPATIBLE初期化パラメータを、新しいデータベースに必要な互換性レベルに設定できます。

COMPATIBLE初期化パラメータは、ご使用のデータベースの互換レベルを制御します。使用しているデータベースをダウングレードする機能をもう必要としないことが確実な場合にのみ、COMPATIBLE初期化パラメータをより高い値に設定してください。

1. COMPATIBLE初期化パラメータ値を増やす前に、データベースのバックアップを取ります(オプション)。

   COMPATIBLE初期化パラメータを高くすると、現在のデータベースが以前のリリースのOracle Databaseとの互換性を失う可能性があります。バックアップによって、必要時に以前のリリースに確実に戻ることができます。

2. サーバー・パラメータ・ファイルを使用している場合は、次のステップを実行します。

   a. COMPATIBLE初期化パラメータの値を設定または変更するには、サーバーのパラメータ・ファイルを更新します。

      たとえば、COMPATIBLE初期化パラメータを`12.2.0`に設定するには、次の文を入力します。

      ```
      SQL> ALTER SYSTEM SET COMPATIBLE = '12.2.0' SCOPE=SPFILE;
      ```

   b. インスタンスを停止し、再起動します。

3. 初期化パラメータ・ファイルを使用している場合は、次のステップを実行します。

   a. インスタンスが実行中の場合は、停止します。

      たとえば:

      ```
      SQL> SHUTDOWN IMMEDIATE
      ```

   b. COMPATIBLE初期化パラメータの値を設定または変更するには、初期化パラメータ・ファイルを編集します。

      たとえば、COMPATIBLE初期化パラメータを`for Oracle Database release 12.2`に設定するには、初期化パラメータ・ファイルに次を入力します。

      ```
      COMPATIBLE = 12.2.0
      ```

   c. `STARTUP`を使用してインスタンスを起動します。

ノート:

ASM ディスク・グループを使用している場合、ディスク・グループの互換性属性は、`init.ora` のデータベース互換性パラメータの値以下である必要があります。

参照:

バックアップの実行の詳細は、『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: 新しいリリース用の初期化パラメータ・ファイルの調整

アップグレード後のTNSNAMES.ORAおよびLISTENERパラメータの調整

手動アップグレードの実行後、`tnsnames.ora`の`local_listener`および`remote_listener`を解決する必要がある場合、それらのパラメータを手動で調整する必要があります。

Oracleの自動アップグレード・ユーティリティは、自動アップグレード中にネットワーク・ネーミングおよびリスナーに対する変更を実行します。ただし、手動アップグレードでは、`tnsnames.ora`もリスナーも変更されません。

関連項目

- インストール後のtnsnames.oraファイル構成

親トピック: 新しいリリース用の初期化パラメータ・ファイルの調整

アップグレード後のOracle RACでのCLUSTER_DATABASE初期化パラメータの設定

Oracle RACデータベース・インスタンスを手動でアップグレードする場合は、CLUSTER_DATABASE初期化パラメータを変更して、ノードを新しいリリースのクラスタに再度参加させる必要があります。

クラスタ・メンバー・ノードのアップグレードでは、クラスタ・データベースをアップグレードする前にCLUSTER_DATABASE初期化パラメータを`false`に設定します。

アップグレードの完了後に、このパラメータを`true`に設定して、ノードを新しいリリースのクラスタに再度参加させることができるようにする必要があります。

> ノート:
>
> Database Upgrade Assistant (DBUA)を使用してアップグレードを実行する場合は、DBUA によってこの作業が自動的に行われます。

親トピック: 手動によるOracle Databaseのアップグレード後にのみ行う作業

7 Oracle Databaseのアップグレード後のアプリケーションのアップグレード

新機能を最大限に利用する場合は、新しいリリースで実行されているアプリケーションをアップグレードする必要があります。

新しいリリースのOracle Databaseにアップグレードすると、多くの新機能および拡張機能を使用できます。各アプリケーションのアップグレード計画のガイダンスについては、これらのトピックを参照してください。

- 新しいOracle Databaseリリースでのアプリケーションのアップグレードの概要
  Oracle Databaseの新しいリリースで使用可能になった機能を使用しない既存のアプリケーションは、変更する必要がありません。
- Oracle Databaseの異なるリリースでのアプリケーションの互換性の問題
  Oracle Databaseの異なるリリース間で、アプリケーションに影響を与える可能性のある互換性の問題が発生する場合があります。
- Oracle Databaseでのソフトウェアのアップグレードおよびクライアントとサーバー構成
  実行するソフトウェアのアップグレードのタイプとクライアントおよびサーバー構成に応じて、プリコンパイラおよびOracle Call Interface (OCI)アプリケーションのアップグレードのオプションを理解するには、これらのトピックを使用してください。
- Oracle Databaseクライアントまたはサーバー・ソフトウェアをアップグレードする場合のアプリケーションの互換性規則
  Oracle Databaseクライアントまたはサーバー・ソフトウェアのアップグレード時に、互換性規則が適用されます。
- Oracle DatabaseでのプリコンパイラおよびOCIアプリケーションのアップグレードについて
  プリコンパイラおよびOracle Call Interface (OCI)アプリケーションをアップグレードする場合、この情報を参照してください。
- スキーマ専用アカウントおよびEXPIREDパスワード・アカウントのアップグレード
  アップグレードを開始する前に、パスワードがEXPIREDステータスであり、そのアカウントがLOCKEDステータスになっている場合にデフォルトのOracle Databaseアカウントにパスワード認証を使用するかどうかを判断します
- OracleプリコンパイラおよびOCIアプリケーションのアップグレード・オプションについて
  Oracle Databaseの新しいリリース上で実行するプリコンパイラおよびOracle Call Interface (OCI)アプリケーションのアップグレード用に、いくつかのオプションが提供されています。
- Oracle Databaseのアップグレード後のSQL*PlusスクリプトおよびPL/SQLのアップグレード
  新しいOracle Databaseリリースの機能およびファンクションを使用するには、新しいOracle Databaseリリースの構文を使用するように既存のSQLスクリプトを変更する必要があります。
- Oracle FormsまたはOracle Developerアプリケーションのアップグレードについて
  Oracle FormsおよびOracle Developerの新機能を確認して、それらの機能の恩恵を受けるアプリケーションがあるかどうか調べます。

スキーマ専用アカウントおよびEXPIREDパスワード・アカウントのアップグレード

アップグレードを開始する前に、パスワードがEXPIREDステータスであり、そのアカウントがLOCKEDステータスになっている場合にデフォルトのOracle Databaseアカウントにパスワード認証を使用するかどうかを判断します

Oracle Database 19c以降のリリースへのアップグレード中に、アップグレード前にパスワードがリセットされておらず(EXPIREDステータスに設定され)、さらにLOCKEDステータスにも設定されているデフォルトのOracleアカウントは、アップグレードの完了後にNO AUTHENTICATIONに設定されます。

この新機能により、スキーマ専用アカウントに変更されたデフォルトのアカウントは、パスワード認証に使用できなくなります。この

機能の利点は、管理者がこれらのOracle Database提供のスキーマのパスワードを定期的にローテーションする必要がなくなったことです。この機能により、攻撃者がデフォルトのパスワードを使用してこれらのアカウントにハッキングするセキュリティ・リスクも削減されます。

アップグレード時にこれらのOracleアカウントがスキーマ専用アカウントに設定されないようにする場合、アップグレードの開始前にアカウントに有効な強力なパスワードを設定するか、アップグレード後にこれらのアカウントに有効な強力なパスワードを設定するか、アップグレードされたOracle Databaseにログインする前にアカウントのロックを解除する必要があります。

アップグレード後、管理者はスキーマ専用アカウントのパスワード認証を有効にすることもできます。ただし、セキュリティを強化するために、これらのアカウントをスキーマ専用のアカウントとして保持することをお薦めします。

関連項目

- 『Oracle Databaseセキュリティ・ガイド』

親トピック: Oracle Databaseのアップグレード後のアプリケーションのアップグレード

新しいOracle Databaseリリースでのアプリケーションのアップグレードの概要

Oracle Databaseの新しいリリースで使用可能になった機能を使用しない既存のアプリケーションは、変更する必要がありません。

新しいOracle Databaseの新しいリリースで実行される既存のアプリケーションは以前のリリースと同様に機能し、同等以上のパフォーマンスを得ることができます。

Oracle Databaseの新しいリリースにアップグレードすると、多くの新機能および拡張機能を使用できます。これらの変更によって、追加の機能とファンクションが提供される一方、パフォーマンスが向上するものもあります。アプリケーションをアップグレードする前に、これらの新機能を確認して完全にテストし、どの機能を使用するかを判断する必要があります。

関連項目

- データベース機能およびライセンス・アプリケーション

親トピック: Oracle Databaseのアップグレード後のアプリケーションのアップグレード

Oracle Databaseの異なるリリースでのアプリケーションの互換性の問題

Oracle Databaseの異なるリリース間で、アプリケーションに影響を与える可能性のある互換性の問題が発生する場合があります。

Oracle Databaseリリース間の相違点が原因で、互換性の問題が発生する可能性があります。また、Oracle Databaseのリリースが新しくなるたびに、Oracleの新しい予約語が追加されたり、初期化パラメータやデータ・ディクショナリが変更される場合があります。詳細は、このドキュメントの関連トピックを参照してください。

Oracle Databaseソフトウェアを新しいリリースにアップグレードする場合は、アプリケーションでOracleにより予約された語が使用されていないか、アプリケーションにデータベースの初期化パラメータとの互換性があるか、アプリケーションにデータベースのデータ・ディクショナリとの互換性があるかを確認してください。

Oracle Databaseの新しいリリースは、特定のオペレーティング・システム・リリースまたはパッチ・セットでのみサポートされる場合があることにも注意してください。以前のリリースのOracle Databaseと組み合せて使用できたオペレーティング・システム・リリースおよびパッチ・セットは、現行のリリースでは使用できないことがあります。oracleソフトウェアをインストールしてアップグレードを実行する前に、オペレーティング・システム要件を確認してください。さらに、一部の機能を使用できるように、システムで追加のパッチ・セットまたはカーネルの追加が必要になる場合があります。

関連項目

- Oracle SQLの予約語とキーワード

親トピック: Oracle Databaseのアップグレード後のアプリケーションのアップグレード

Oracle Databaseでのソフトウェアのアップグレードおよびクライアントとサーバー構成

実行するソフトウェアのアップグレードのタイプとクライアントおよびサーバー構成に応じて、プリコンパイラおよびOracle Call Interface (OCI)アプリケーションのアップグレードのオプションを理解するには、これらのトピックを使用してください。

- Oracle Databaseで使用可能なクライアントおよびサーバーの構成
  プリコンパイラおよびOCIアプリケーションを実行するためのクライアント/サーバー構成を選択します。
- Oracle Databaseのクライアントおよびサーバー・ソフトウェアのソフトウェア・アップグレードのタイプ
  参照してリリースがメジャー・リリースまたはメンテナンス・リリースのどちらであるかを特定してください。

親トピック: Oracle Databaseのアップグレード後のアプリケーションのアップグレード

Oracle Databaseで使用可能なクライアントおよびサーバーの構成

プリコンパイラおよびOCIアプリケーションを実行するためのクライアント/サーバー構成を選択します。

プリコンパイラおよびOCIアプリケーションはクライアント/サーバー環境のクライアントで実行されます(ここでのサーバーは、Oracle Databaseサーバーです)。ご使用の環境で、次の1つ以上のクライアント/サーバー構成を使用できます。

異なるコンピュータ上に存在するOracle Databaseクライアントおよびサーバー

クライアント・ソフトウェアおよびサーバー・ソフトウェアが異なるコンピュータにあり、それぞれがネットワークで接続されています。クライアント環境とサーバー環境は独立しています。

同じコンピュータ上の異なるOracleロケーションに存在するOracle Databaseクライアントおよびサーバー

クライアント・ソフトウェアおよびサーバー・ソフトウェアは同じコンピュータ上にありますが、異なるOracleホーム・ディレクトリにインストールされています。クライアント環境とサーバー環境は独立しています。

同じOracleロケーション上に存在するOracle Databaseクライアントおよびサーバー

クライアント・ソフトウェアおよびサーバー・ソフトウェアが同じコンピュータ上の同じOracleホームにインストールされています。この場合、サーバー・ソフトウェアをアップグレードすると、クライアント・ソフトウェアもアップグレードされます。

参照:

クライアント/サーバー環境の詳細は、『Oracle Database概要』を参照してください

親トピック: Oracle Databaseでのソフトウェアのアップグレードおよびクライアントとサーバー構成

Oracle Databaseのクライアントおよびサーバー・ソフトウェアのソフトウェア・アップグレードのタイプ

参照してリリースがメジャー・リリースまたはメンテナンス・リリースのどちらであるかを特定してください。

Oracle Databaseのクライアントおよびサーバー・ソフトウェアでは、Oracle Databaseのメジャー・リリースとOracle Databaseのメンテナンス・リリースの2つのタイプのアップグレードを行うことができます。

Oracle Databaseのメジャー・リリースのアップグレード

メジャー・リリースでは、アップグレードでリリース番号の最初の数字が変更されます。たとえば、Oracle Database 11gからOracle Database 12cへのアップグレードは、データベースのメジャー・リリースのアップグレードです。

Oracle Databaseのメンテナンス・リリースのアップグレード

メンテナンス・リリースでは、アップグレードでリリース番号の2番目の数字が変更されます。たとえば、Oracle Database 12cリリース1 (12.1)からOracle Database 12gリリース2 (12.2)へのアップグレードは、データベースのメンテナンス・リリースのアップグレードです。ただし、メンテナンス・リリースでは新機能が導入されることがあります。

親トピック: Oracle Databaseでのソフトウェアのアップグレードおよびクライアントとサーバー構成

Oracle Databaseクライアントまたはサーバー・ソフトウェアをアップグレードする場合のアプリケーションの互換性規則

Oracle Databaseクライアントまたはサーバー・ソフトウェアのアップグレード時に、互換性規則が適用されます。

互換性規則は、実行するソフトウェア・アップグレードのタイプおよびクライアント/サーバー構成のタイプに基づきます。

> ノート:
>
> この項では、「ソフトウェアのアップグレードおよびクライアントとサーバー構成」で説明する用語を使用しています。

- Oracle Databaseサーバー・ソフトウェアのアップグレードに対する規則
  Oracle Databaseのサーバー・ソフトウェアのアップグレード時には、データベース環境に応じて様々な規則が適用されます。
- Oracle Databaseクライアント・ソフトウェアのアップグレード
  サーバー・ソフトウェアとクライアント・ソフトウェアのリリース番号を同じにしておくと、アプリケーションの最大の安定性を得ることができます。

親トピック: Oracle Databaseのアップグレード後のアプリケーションのアップグレード

Oracle Databaseサーバー・ソフトウェアのアップグレードに対する規則

Oracle Databaseのサーバー・ソフトウェアのアップグレード時には、データベース環境に応じて様々な規則が適用されます。

- クライアント環境を変更しない場合、再リンクは不要
  アップグレード後にアプリケーションを再リンクする必要があるかどうかを判断するには、これらのシナリオを参照してください。
- Oracle Databaseサーバーの新/旧リリースでアプリケーションの実行が可能
  データベース・サーバーに対してプリコンパイラまたはOCIアプリケーションを実行する場合、クライアント・ソフトウェア・リリース以上のリリースのデータベース・サーバー・ソフトウェアを使用することをお薦めします。

親トピック: Oracle Databaseクライアントまたはサーバー・ソフトウェアをアップグレードする場合のアプリケーションの互換性規則

クライアント環境を変更しない場合、再リンクは不要

アップグレード後にアプリケーションを再リンクする必要があるかどうかを判断するには、これらのシナリオを参照してください。

クライアントとサーバーが異なるコンピュータにあるか、または同じコンピュータの異なるOracleホーム・ディレクトリにあるときに、クライアント・ソフトウェアを変更しないでOracle Databaseサーバー・ソフトウェアをアップグレードする場合は、アプリケーションをプリコンパイル、コンパイルまたは再リンクする必要はありません。

この一連のシナリオでは、Oracle Databaseを使用するクライアント・ソフトウェアは、サーバー・ソフトウェアとは異なる場所にあり、クライアント・ソフトウェアはアップグレードの直接の影響を受けずに動作し続けます。

ただし、アプリケーションがOracle Databaseサーバーと同じOracleホームを使用している場合は、サーバーをアップグレードするとクライアント・ソフトウェアもアップグレードされます。この場合、Oracle Databaseクライアント・ソフトウェアのアップグレードの規則に従う必要があります。

ノート:

両方のバイナリで同じ Oracle ホームを使用している場合、新しいプリコンパイラまたは OCI クライアント・ソフトウェアはインストールせずに、Oracle Database サーバー・ソフトウェアをアップグレードできます。この場合、クライアント・ソフトウェアはアップグレードされません。ただし、この構成はお薦めしません。

親トピック: Oracle Databaseサーバー・ソフトウェアのアップグレードに対する規則

Oracle Databaseサーバーの新/旧リリースでアプリケーションの実行が可能

データベース・サーバーに対してプリコンパイラまたはOCIアプリケーションを実行する場合、クライアント・ソフトウェア・リリース以上のリリースのデータベース・サーバー・ソフトウェアを使用することをお薦めします。

この推奨構成は、厳密に必要とされるものではありません。

たとえば、クライアント・ソフトウェアがOracle 12cリリース2 (12.2.0.1)の場合に、サーバーに対してクライアント上でプリコンパイラ・アプリケーションを実行する場合、そのサーバー・ソフトウェアはOracle 12cリリース2 (12.2)以降であることをお薦めします。

親トピック: Oracle Databaseサーバー・ソフトウェアのアップグレードに対する規則

Oracle Databaseクライアント・ソフトウェアのアップグレード

サーバー・ソフトウェアとクライアント・ソフトウェアのリリース番号を同じにしておくと、アプリケーションの最大の安定性を得ることができます。

この情報を使用して、Oracle Databaseクライアントのインストールを計画してください。アプリケーションのリンク方法に応じて、Oracle Databaseのクライアント・ソフトウェアのアップグレード時に適用される規則が異なります。

ご使用のクライアント・ソフトウェアをアップグレードして、現行のサーバー・ソフトウェアと一致させることをお薦めします。たとえば、Oracle Databaseを新しいOracle Databaseリリースにアップグレードする場合、Oracle Databaseクライアント・ソフトウェアを新しいリリースにアップグレードすることもお薦めします。最新のOracle Databaseクライアント・ソフトウェアでは、新しいリリースでのみ使用できる追加機能やパフォーマンス向上が提供される場合があります。

- イメージベースのOracle Databaseクライアント・インストールについて
  Oracle Database 19c以上では、イメージベースのインストールによってOracle Databaseクライアント・ソフトウェアのインストールや構成を簡単に行うことができます。
- アプリケーションと新しいライブラリとのリンクについて
  プリコンパイラ・アプリケーションおよびOracle Call Interface (OCI)によって生成されたコードを、サーバー・リリース以上のリリースのクライアント・ライブラリにリンクできます。
- 静的にリンクされたアプリケーションを常に再リンクさせる必要がある
  静的にリンクされたコードは、アップグレードされたORACLE_HOMEのエラー・メッセージと非互換になる可能性があります。
- 動的にリンクされたアプリケーションの再リンクについて
  Oracle Database 10gリリース1 (10.1)以上のリリースから動的にリンクされたOCIアプリケーションは、現行のリリースと上位互換性があります。

親トピック: Oracle Databaseクライアントまたはサーバー・ソフトウェアをアップグレードする場合のアプリケーションの互換性規則

イメージベースのOracle Databaseクライアント・インストールについて

Oracle Database 19c以上では、イメージベースのインストールによってOracle Databaseクライアント・ソフトウェアのインストールや構成を簡単に行うことができます。

Oracle Databaseクライアントをインストールするには、新規Oracleホームを作成し、新しく作成したOracleホームにイメージ・ファイルを抽出し、設定ウィザードを実行して、Oracle Database製品を登録します。

Oracle Databaseホームを配置するディレクトリにイメージ・ソフトウェア(`client_home.zip`)を抽出してから、クライアント設定ウィザードを実行し、Oracle Databaseクライアントのインストールおよび構成を開始する必要があります。作成したOracleホームのディレクトリ・パスがOracle Optimal Flexible Architectureの推奨事項に準拠することをお薦めします。

イメージベースのインストールを使用すると、管理者インストール・タイプのOracle Databaseクライアント32ビットおよび64ビット構成をインストールできます。

Oracle DatabaseおよびOracle Grid Infrastructureのイメージ・ファイルのインストールと同様に、Oracle Databaseクライアントのイメージ・インストールではOracle Databaseクライアントのインストールが簡略化され、ベスト・プラクティスのデプロイメントが保証されます。Oracle Databaseクライアントのインストール・バイナリは、イメージ以外のzipファイルとして従来の形式で引き続き使用できます。

親トピック: Oracle Databaseクライアント・ソフトウェアのアップグレード

アプリケーションと新しいライブラリとのリンクについて

プリコンパイラ・アプリケーションおよびOracle Call Interface (OCI)によって生成されたコードを、サーバー・リリース以上のリリースのクライアント・ライブラリにリンクできます。

使用するOCIランタイム・ライブラリは、アプリケーションの開発に使用されたOCIライブラリのリリース以上のリリースである必要があります。

親トピック: Oracle Databaseクライアント・ソフトウェアのアップグレード

静的にリンクされたアプリケーションを常に再リンクさせる必要がある

静的にリンクされたコードは、アップグレードされたORACLE_HOMEのエラー・メッセージと非互換になる可能性があります。

静的にリンクされたOCIアプリケーションは、メジャー・リリースとマイナー・リリースの両方で再リンクする必要があります。静的にリンクされたOracleクライアントサイド・ライブラリ・コードは、アップグレードされたORACLE_HOMEのエラー・メッセージと非互換になる可能性があります。たとえば、エラー・メッセージが追加パラメータで更新されると、それは静的にリンクされたコードとは非互換になります。

親トピック: Oracle Databaseクライアント・ソフトウェアのアップグレード

動的にリンクされたアプリケーションの再リンクについて

Oracle Database 10gリリース1 (10.1)以上のリリースから動的にリンクされたOCIアプリケーションは、現行のリリースと上位互換性があります。

Oracleクライアントサイド動的ライブラリは、そのライブラリの前のバージョンと上位互換性があります。Oracle Universal Installerによって前のバージョンのライブラリへのシンボリック・リンクが作成され、このシンボリック・リンクが現行バージョンのライブラリに解決されます。したがって、前のバージョンのOracleクライアントサイド動的ライブラリと動的にリンクされたアプリケーションは、現行バージョンのOracleクライアントサイド・ライブラリと再リンクしなくても動作します。

ノート:

アプリケーションがランタイム・ライブラリの検索パス(Linux での-rpath など)にリンクされていると、リンクされているバージョンの Oracle クライアントサイド・ライブラリを使用して引き続き実行される場合があります。現行バージョンの Oracle クライアントサイド・ライブラリを使用して実行するには、アプリケーションを再リンクする必要があります。

アプリケーションが遅延オプション(静的にリンクされているアプリケーションなど)にリンクされている場合は、再リンクする必要があります。

Oracle Database 10g リリース 1 (10.1)より前のリリースのアプリケーションを使用している場合は、再リンクする必要があります。

親トピック: Oracle Databaseクライアント・ソフトウェアのアップグレード

Oracle DatabaseでのプリコンパイラおよびOCIアプリケーションのアップグレードについて

プリコンパイラおよびOracle Call Interface (OCI)アプリケーションをアップグレードする場合、この情報を参照してください。

プリコンパイラおよびOracle Call Interfaceのアップグレードのテストは、次のステップで構成されます。

1. 本番環境をアップグレードする前に、テスト環境を作成します。
2. テスト環境に、アップグレードされたアプリケーションおよび新しいOracle Databaseソフトウェアを含めてください。
3. テスト環境で、アプリケーションの現実的なテストが行われることを確認します。

関連項目

- 『Pro*C/C++プログラマーズ・ガイド』
- Oracle Call Interfaceプログラマーズ・ガイド¥

親トピック: Oracle Databaseのアップグレード後のアプリケーションのアップグレード

OracleプリコンパイラおよびOCIアプリケーションのアップグレード・オプションについて

Oracle Databaseの新しいリリース上で実行するプリコンパイラおよびOracle Call Interface (OCI)アプリケーションのアップグレード用に、いくつかのオプションが提供されています。

アップグレードのオプションは、難易度が高く、考えられるメリットが大きくなる順序に記述されています。つまり、選択肢1は最も簡単ですが、考えられるメリットは最も小さくなり、選択肢3は最も難しい方法ですが、考えられるメリットは最も大きくなります。

- オプション1: アプリケーションを変更しない
  アプリケーションおよびその環境を変更しません。
- オプション2: 新しいソフトウェアを使用してアプリケーションをプリコンパイルまたはコンパイルする
  APIが非推奨になったか、または変更されている場合は、アプリケーション・コードを変更する必要があります。
- オプション3: アプリケーション・コードを変更して、新しいOracle Databaseの機能を使用する
  アプリケーションのコードを変更して、新しいOracle Databaseの機能を利用します。
- OracleプリコンパイラおよびOCIアプリケーションの開発環境の変更
  使用する新機能を決定したら、これらの機能を使用するためにアプリケーションのコードを変更します。

親トピック: Oracle Databaseのアップグレード後のアプリケーションのアップグレード

オプション1: アプリケーションを変更しない

アプリケーションおよびその環境を変更しません。

アプリケーションを再リンク、プリコンパイルまたはコンパイルせず、また、アプリケーション・コードを変更しません。アプリケーションは、新しいOracle Databaseリリースで引き続き動作します。この選択肢では、アプリケーションのOracleホーム環境をアップグレードしないでください。アプリケーションは変更しなくても、新しいリリースのOracle Databaseサーバーで引き続き使用できます。この選択肢の主なメリットは、単純で簡単であることです。さらに、この選択肢では、クライアント・コンピュータをアップグレードする必要がないため、管理の必要性が最も少なくなります。多くのクライアント・コンピュータがある場合は、すべてのコンピュータのアップグレードに費やす管理コストを削減することが大変重要になります。

このオプションの主なデメリットは、Oracle Databaseの新しいリリースで使用可能な機能をアプリケーションで使用できないことです。また、期待できる新しいOracle Databaseリリースのパフォーマンス上のすべてのメリットをアプリケーションで活用することができません。

親トピック: OracleプリコンパイラおよびOCIアプリケーションのアップグレード・オプションについて

オプション2: 新しいソフトウェアを使用してアプリケーションをプリコンパイルまたはコンパイルする

APIが非推奨になったか、または変更されている場合は、アプリケーション・コードを変更する必要があります。

Oracle Databaseの新しいリリースを使用して、プリコンパイルまたはコンパイルし、次にアプリケーションを再リンクします。新しいリリースのOracle Databaseソフトウェアにアップグレードする場合、APIが非推奨になったか変更されたために必要になったコード変更を行った後、新しいソフトウェアを使用してアプリケーションをプリコンパイルまたはコンパイルする必要があります。

この選択肢では、各クライアント・コンピュータに新しいOracle Databaseクライアント・ソフトウェアをインストールする必要があり

ます。クライアントの数にかかわらず、必要なアプリケーションのプリコンパイルまたはコンパイルおよび再リンクは1回のみです。

再コンパイルすることによって、アプリケーション・コードの構文チェックを実行します。新しいOracle Databaseソフトウェアを使用してプリコンパイルまたはコンパイルすると、Oracleソフトウェアの以前のリリースでは検出されなかったアプリケーション・コードのいくつかの問題が、検出される場合があります。新しいソフトウェアを使用したプリコンパイルまたはコンパイルは、以前に検出されなかったアプリケーション・コードの問題の検出および解決に役立ちます。

また、アプリケーションは新しいOracle Databaseリリースで動作するため、再コンパイルすることによって、アプリケーションの最大の安定性を得ることができます。さらに、使用可能な最新のツール製品および機能を使用して、ご使用の環境で新しい開発を行うことができます。さらに、再コンパイルおよび再リンクの実行後のみ、新しいOracleソフトウェアで得られるパフォーマンスの向上を得ることができます。

親トピック: OracleプリコンパイラおよびOCIアプリケーションのアップグレード・オプションについて

オプション3: アプリケーション・コードを変更して、新しいOracle Databaseの機能を使用する

アプリケーションのコードを変更して、新しいOracle Databaseの機能を利用します。

新しいOracle Databaseリリースの新機能を使用するために、アプリケーション・コードを変更します。コードをプリコンパイルまたはコンパイルし、再リンクさせます。この選択肢は最も難しい方法ですが、考えられるメリットも最大です。この選択肢では、「オプション2: 新しいソフトウェアを使用してアプリケーションをプリコンパイルまたはコンパイルする」で説明したすべてのメリットを実現できますまた、アプリケーションを変更することにより、Oracle Databaseの新しいリリースで得られるパフォーマンスおよびスケーラビリティのメリットを活用できます。さらに、新しいリリースでのみ使用できる新機能を、アプリケーションに追加することもできます。使用する機能の実装方法については、ご使用の開発環境のOracleマニュアルを参照してください。

参照:

この新しいOracle Databaseリリースの機能の詳細は、Oracle Database新機能ガイドを参照してください

親トピック: OracleプリコンパイラおよびOCIアプリケーションのアップグレード・オプションについて

OracleプリコンパイラおよびOCIアプリケーションの開発環境の変更

使用する新機能を決定したら、これらの機能を使用するためにアプリケーションのコードを変更します。

ご使用の開発環境に基づいて、次の項の該当する指示に従ってください。

- プリコンパイラ・アプリケーションの変更
  これらのステップを完了して、新しいOracle Databaseリリースの機能を使用するためにプリコンパイラ・アプリケーションを変更します。
- OCIアプリケーションの変更
  新しいOracle Databaseリリースの新機能を使用するには、新しいOracle DatabaseリリースのOCIコールでアプリケーションを再コンパイルする必要があります。

親トピック: OracleプリコンパイラおよびOCIアプリケーションのアップグレード・オプションについて

プリコンパイラ・アプリケーションの変更

これらのステップを完了して、新しいOracle Databaseリリースの機能を使用するためにプリコンパイラ・アプリケーションを変更します。

新しいOracle Databaseリリースの新機能を使用するには、既存のアプリケーションに新しいコードを追加し、アプリケーションを再コンパイルする必要があります。

1. 新しい機能のコードを既存のアプリケーションに取り込みます。
2. Oracleプリコンパイラを使用して、各アプリケーションをプリコンパイルします。
3. 各アプリケーションをコンパイルします。
4. プリコンパイラに付属する新しいOracle Databaseリリースのランタイム・ライブラリ(`SQLLIB`)を、各アプリケーションと再リンクさせます。

親トピック: OracleプリコンパイラおよびOCIアプリケーションの開発環境の変更

OCIアプリケーションの変更

新しいOracle Databaseリリースの新機能を使用するには、新しいOracle DatabaseリリースのOCIコールでアプリケーションを再コンパイルする必要があります。

1. 新しいOracle DatabaseリリースのOCIコールを既存のアプリケーションに取り込みます
2. アプリケーションをコンパイルします。
3. アプリケーションを新しいOracle Databaseリリースのランタイム・ライブラリと再リンクします。

親トピック: OracleプリコンパイラおよびOCIアプリケーションの開発環境の変更

Oracle Databaseのアップグレード後のSQL*PlusスクリプトおよびPL/SQLのアップグレード

新しいOracle Databaseリリースの機能およびファンクションを使用するには、新しいOracle Databaseリリースの構文を使用するように既存のSQLスクリプトを変更する必要があります。

既存のSQLスクリプトで新しいOracle Databaseリリースの機能およびファンクションを使用しない場合、それらは新しいOracle Databaseリリースで変更されずに実行されるため、変更は不要です。

新しいOracle Databaseリリースではエラー確認機能が強化されているため、実行時ではなくコンパイル時にエラーを識別できる場合があります。

親トピック: Oracle Databaseのアップグレード後のアプリケーションのアップグレード

Oracle FormsまたはOracle Developerアプリケーションのアップグレードについて

Oracle FormsおよびOracle Developerの新機能を確認して、それらの機能の恩恵を受けるアプリケーションがあるかどうか調べます。

Oracle Database 12cでは『*Oracle Database*開発ガイド』の名前が『*Oracle Database*アドバンスト・アプリケーション開発者ガイド』に変更されました。アプリケーションの開発手順の変更に関する情報、およびアプリケーション開発に影響を与えるこのOracle Databaseリリースの新機能については、その出版物と新機能ガイドを確認してください。

親トピック: Oracle Databaseのアップグレード後のアプリケーションのアップグレード

8 以前のリリースへのOracle Databaseのダウングレード

Oracle Databaseのサポートされているリリースでは、最後にアップグレードしたリリースにデータベースをダウングレードできます。

たとえば、Oracle Database 12cリリース1 (12.1.0.2)を19cにアップグレードし、compatible初期化パラメータを19.1.0に変更しなかった場合、リリース12.1.0.2にダウングレードできます。

- Oracle Databaseのダウングレードがサポートされているリリース
  データベースがアップグレードされる前の元のOracle Databaseのリリースに基づいて、メジャー・リリースとリリース更新またはパッチ・セット・リリースの両方をダウングレードできます。
- プライマリでのスタンバイ・データベースのダウングレードの準備
  Oracle Data Guardスタンバイ・データベースを使用している場合は、スタンバイ・データベースをダウングレードするために使用できる手順をプライマリ・データベースで確認します
- Oracle Databaseのダウングレード時のCOMPATIBLEパラメータの確認
  アップグレード後に`COMPATIBLE`が変更された場合、Oracle Databaseをダウングレードできなくなります。
- Oracle Databaseのダウングレード前の全体バックアップの実行
  サポートされている以前のリリースにダウングレードする前に、新しいOracle Databaseリリースの完全バックアップを実行することをお薦めします。
- Oracle Databaseをダウングレードする前に必要なステップの実行
  アップグレード前の以前のリリースにOracle Databaseをダウングレードする前に、ここに説明されている必要な準備ステップを完了します。
- スクリプトを使用したCDBまたは非CDBのOracle Databaseのダウングレード
  ダウングレードを自動化するために、Oracle Database 19c以降、`dbdowngrade`ユーティリティ・スクリプトが提供されています。以前のリリースと同様に、必要に応じて`catdwgrd.sql`を手動で実行することもできます。
- 単一のプラガブルOracle Database (PDB)のダウングレード
  Oracle Databaseをダウングレードする場合、CDB全体をダウングレードせずに1つのPDBをダウングレードできます。
- Oracle Application Expressが含まれるPDBのダウングレード
  このプロシージャを使用すると、Oracle Application Expressが含まれるPDBをダウングレードするときにINVALID OBJECTS OWNED BY APEX_050000エラーを回避できます。
- Oracle Databaseのダウングレード後のタスク
  Oracle Databaseのリリースをダウングレードした後、互換性、コンポーネントおよびサポートされるプロトコルに影響する変更により、追加のタスクを完了する必要がある場合があります。
- Oracle Databaseのダウングレードのトラブルシューティング
  このトラブルシューティング情報を使用して、Oracle Databaseのダウングレード時に発生する可能性のある問題に対処します。

Oracle Databaseのダウングレードがサポートされているリリース

データベースがアップグレードされる前の元のOracle Databaseのリリースに基づいて、メジャー・リリースとリリース更新またはパッチ・セット・リリースの両方をダウングレードできます。

ダウングレードがサポートされているリリース

非CDB Oracle Databaseは、Oracle Database 19cからOracle Database 18c、Oracle Database 12cおよびOracle Database 11gリリース11.2.0.4にダウングレードできます。

PDBまたはCDBはOracle Database 18c、Oracle Database 12cリリース12.2およびOracle Database 12cリリース12.1.0.2にダウングレードできます。

> ノート:
>
> 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。

次の表に、ダウングレードがサポートされているリリースに関する追加情報を示します。この表を使用する際には、「Oracle Databaseをダウングレードするときの非互換性の確認」で互換性についても確認してください。

表8-1 ダウングレードがサポートされているリリースおよびエディション

Oracle Database のリリースまたはエディション	ダウングレード(可/不可)	ノート
18	可	現時点では追加情報はありません。
12.2.0.1 および 12.1.0.2	可	COMPATIBLE 初期化パラメータを 12.1.0.2 に設定した後は、データベースをダウングレードできません。 非 CDB は、19 から 18、または 12.2 から 12.1.0.2、または 11.2.0.4 (サポートされているすべてのアップグレード・リリース)にダウングレードできます。 CDB をダウングレードする前に、または PDB を切断してダウングレードする前に、最新のリリース更新、リリース・リビジョン、バンドル・パッチまたはパッチ・セット更新(BP または PSU)をインストールします。パッチは、My Oracle Support でダウンロードできます。My Oracle Support ノート 756671.1 を参照して、最新のパッチ・セットを取得してください。 新しい Oracle Database リリースの最小の互換性設定より前のリリースにダウングレードすることはできません。
Oracle Enterprise Manager	不可	サポートされている以前のリリースにダウングレードする場合、Oracle Enterprise Manager の制御を再構成する必要があります。 アップグレードを開始する前に、`emdwgrd` ユーティリティを使用して、ダウングレード後に Oracle Enterprise Manager Database Control (DB Control)をリストアできるように、DB Control ファイルおよびデータを保存する必要があります。
Oracle Database	不可	Oracle Database Express Edition からアップグレードされたデータ

Oracle Database のリリースまたはエディション	ダウングレード(可/不可)	ノート
Express Edition		ベースをダウングレードすることはできません。

ダウングレード前に確認する推奨事項

サポートされている以前のリリースでは、Oracle Databaseのダウングレードについて、次の推奨事項があります。

- マルチテナント・アーキテクチャは、マルチテナント・コンテナ・データベース(CDB)およびプラガブル・データベース(PDB)用のアーキテクチャ機能を提供します。マルチテナント・アーキテクチャにアップグレードし、このリリースへアップグレードした後にcompatible初期化パラメータを最大レベルに設定した場合は、アップグレード後にデータベースをダウングレードできなくなります。
- このリリースでは、SELECTに加えて、新しいオブジェクト権限READが含まれています。アップグレード後、このオブジェクト権限の次の影響に注意してください。
  - SELECTおよびREADオブジェクト権限を持っている場合、READ権限は削除されます。
  - 前にREADオブジェクト権限のみを持っていた場合、READオブジェクト権限はSELECTオブジェクト権限に変換されます。

  READおよびSELECTオブジェクト権限の詳細は、『Oracle Databaseセキュリティ・ガイド』を参照してください。
- アップグレードするデータベースにOracle XML DBがインストールされていない場合、ダウングレード中にOracle XML DBはアンインストールされます。たとえば、Oracle Database 11gリリース2 (11.2)とともにOracle XML DBをインストールしなかった場合、XML DBはOracle Database 12cとともにインストールされます。データベースをダウンロードする場合は、ダウングレードの一部としてOracle XML DBがアンインストールされます。Oracle XML DBはOracle Database 12cリリース1 (12.1)以降のリリースでは、デフォルトで組み込まれています。
- Oracle Database 12cへのアップグレード時に、Database (DB) Controlリポジトリは削除されます。以前のリリースにダウングレードする場合、Database (DB) Controlを再構成してダウングレード後も使用できるようにする必要があります。
- Oracle Enterprise Managerのダウングレードはサポートされていません。サポートされている以前のリリースにダウングレードする場合、Oracle Enterprise Managerの制御を再構成する必要があります。

関連項目

- https://support.oracle.com/rs?type=doc&id=756671.1
- 『Oracle Databaseセキュリティ・ガイド』

親トピック: 以前のリリースへのOracle Databaseのダウングレード

プライマリでスタンバイ・データベースをダウングレードする準備

Oracle Data Guardスタンバイ・データベースを使用している場合は、スタンバイ・データベースをダウングレードするために使用できる手順をプライマリ・データベースで確認します

スタンバイ・データベースを中断することなく、データベースをダウングレードできます。Oracle Data Guard構成にフィジカル・スタンバイ・データベースまたはロジカル・スタンバイ・データベースが存在する場合にOracle Databaseをダウングレードするには、

*Oracle Data Guard*概要および管理のOracle Data Guard構成でのデータベースのパッチ適用、アップグレードおよびダウングレードの説明に従って、ダウングレード・シナリオの手順を使用します

関連項目

- Oracle Data Guard構成におけるデータベースのパッチ適用、アップグレードおよびダウングレード

親トピック: 以前のリリースへのOracle Databaseのダウングレード

Oracle Databaseのダウングレード時のCOMPATIBLEパラメータの確認

アップグレード後にCOMPATIBLEが変更された場合、Oracle Databaseをダウングレードできなくなります。

COMPATIBLEパラメータを更新して、Oracle Databaseリリースの互換性レベルを現在のリリースに設定した場合は、以前のリリースにダウングレードできません。この問題は、新しいリリースではデータ・ディクショナリが変更されており、ダウングレードを妨げるその他の機能変更がある可能性もあるために起こります。

ダウングレードする前にデータベースのCOMPATIBLEパラメータ設定を確認するには、次のコマンドを入力します。

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

ノート:

Oracle ASM ディスク・グループの場合、アップグレード後に compatible.asm パラメータをアップグレードされたリリース値に変更した場合、前のリリースにダウングレードすると、Oracle ASM ディスク・グループをマウントできません。compatible.asm の値は、ディスク・グループをマウントできる Oracle ASM の最低限のリリースを設定します。

ダウングレードの一部として、ダウングレードされたリリース・レベルに新しいディスク・グループを作成し、ダウングレードされた互換性のある ASM ディスク・グループにデータをリストアする必要があります。

親トピック: 以前のリリースへのOracle Databaseのダウングレード

Oracle Databaseのダウングレード前の全体バックアップの実行

サポートされている以前のリリースにダウングレードする前に、新しいOracle Databaseリリースの完全バックアップを実行することをお薦めします。

レベル0の完全バックアップを完了する時間がない場合は、少なくともレベル1のバックアップを完了してください。

関連項目

- Oracle Databaseバックアップおよびリカバリ・ユーザーズ・ガイドのデータベースのバックアップ

親トピック: 以前のリリースへのOracle Databaseのダウングレード

Oracle Databaseをダウングレードする前に必要なステップの実行

アップグレード前の以前のリリースにOracle Databaseをダウングレードする前に、ここに説明されている必要な準備ステップを完了します。

ダウングレードを開始する前に、データベース・リリース間の非互換性を解決しておく必要があります。たとえば、状況に応じて、ダウングレードを開始する前に、データベースのコンポーネントを無効化する必要があるかどうかを判断します。

1. データベースでOracle Database Vaultを有効化している場合は、データベースをダウングレードする前にOracle Database Vaultを無効化します。

   DBA_DV_STATUSを使用して、Oracle Database Vaultが有効になっているかどうかを確認します。

   ```
   SQL> SELECT * FROM DBA_DV_STATUS;
   ```

   出力がTRUEの場合は、Oracle Database Vaultが有効化されているため、無効化する必要があります。

   マルチテナント・アーキテクチャのOracle Databaseシステムで、CDB$ROOTにCDB_DV_STATUSを使用して、CDBに接続されているすべてのPDBのOracle Database Vaultステータスを検出します。

   ```
   SQL> SELECT * FROM CDB_DV_STATUS;
   ```

2. 統合監査が有効化されている場合は、統合監査証跡をバックアップおよび消去できます。

   a. 統合監査レコードが存在するかどうかを確認します。

      ```
      SQL> SELECT COUNT(*) FROM UNIFIED_AUDIT_TRAIL;
      ```

   b. 既存の監査データを表にバックアップします。たとえば:

      ```
      SQL> CREATE TABLE UA_DATA AS (SELECT * FROM UNIFIED_AUDIT_TRAIL);
      ```

   c. 監査証跡をクリーンアップします。

      ```
      EXEC DBMS_AUDIT_MGMT.CLEAN_AUDIT_TRAIL(audit_trail_type =>
      DBMS_AUDIT_MGMT.AUDIT_TRAIL_UNIFIED, use_last_arch_timestamp => FALSE);
      ```

3. ダウングレードする前に、データベースで使用しているタイム・ゾーン・ファイルのバージョンがダウングレード対象のデータベースのOracleホームに含まれていることを確認します。

   データベースで現在使用されているタイムゾーン・ファイルのバージョンを確認するには、次のコマンドを使用してV$TIMEZONE_FILEを問い合せます。

   ```
   SQL> select * from  V$TIMEZONE_FILE;
   ```

   たとえば:

   問合せにより列V$TIMEZONE_FILE.FILENAMEにtimezlrg_20.datが返されたら、対象のOracleホームにそのファイルが存在するかどうかを確認します。

   LinuxおよびUnix

   ```
   $ORACLE_HOME/oracore/zoneinfo/timezlrg_20.dat
   ```

   Windows

   ```
   %ORACLE_HOME%¥oracore¥zoneinfo¥timezlrg_20.dat
   ```

   必要なタイム・ゾーン・ファイルが対象のOracleホームにない場合は次のいずれかを実行します。

   a. 現在のバージョンのタイム・ゾーン・ファイルをパッチとしてインストールしており、パッチ番号を覚えている場合は、その同じパッチ番号を使用して、My Oracle Support Webサイトから対象リリースに対応するタイム・ゾーン・ファイルをダウンロードします。

b. My Oracle Support Webサイトのパッチ検索機能を使用して、正しいパッチを見つけてください。入力する検索基準は次のとおりです。「製品」に「Oracle Database」、「リリース」にターゲット・リリース、そして「説明に次を含む」に「DST」

c. My Oracle Support Webサイトでパッチが見つからない場合は、Oracle Supportでサービス・リクエストを記録します。

必要なパッチを見つけてダウンロードしたら、対象のOracleホームにインストールします。

4. 固定オブジェクトに基づいてオブジェクトを作成した場合は、これらのオブジェクトを削除して、ORA-00600エラーの発生を回避します。ダウングレード後にこれらのオブジェクトを再作成できます。

5. データベースにOracle Enterprise Managerが構成されている場合は、Enterprise Managerユーザーを削除します。

```
DROP USER sysman CASCADE;
```

ノート:

Enterprise Manager ユーザーを削除すると、MGMT*シノニムが無効になる場合があります。ダウングレードしたデータベースで Oracle Enterprise Manager の制御を使用するには、Oracle Enterprise Manager を再構成する必要があります。

関連項目

- Oracle Databaseグローバリゼーション・サポート・ガイド

親トピック: 以前のリリースへのOracle Databaseのダウングレード

スクリプトを使用したCDBまたは非CDBのOracle Databaseのダウングレード

ダウングレードを自動化するために、Oracle Database 19c以降、dbdowngradeユーティリティ・スクリプトが提供されています。以前のリリースと同様に、必要に応じてcatdwgrd.sqlを手動で実行することもできます。

- Dbdowngradeを使用したCDBおよび非CDBのOracle Databaseのダウングレード
  以前のデータベース・リリースにダウングレードするには、ダウングレード・スクリプトdbdowngradeを実行することをお薦めします。
- catdwgrd.sqlを使用したCDBまたは非CDBのOracle Databaseの手動ダウングレード
  手動スクリプトを実行する必要があるか、過度のスレッドの問題を回避する場合、手動でcatdwgrd.sqlスクリプトを実行して、サポートされているメジャー・リリースまたは以前のリリース更新にOracle Databaseをダウングレードできます。

親トピック: 以前のリリースへのOracle Databaseのダウングレード

Dbdowngradeを使用したCDBおよび非CDBのOracle Databaseのダウングレード

以前のデータベース・リリースにダウングレードするには、ダウングレード・スクリプトdbdowngradeを実行することをお薦めします。

Oracle Database 19c以上では、ダウングレード・ユーティリティ・スクリプトdbdowngradeが提供されています。dbdowngradeユーティリティを使用すると、ダウングレードに適切な値が設定され、ダウングレードの開始方法が簡略化されま

す。具体的には、catcon.plへの基礎となるコールが推奨値を使用することが保証されるため、過剰なスレッドが生成されるために発生するエラーが削減されます。この機能は、マルチテナント・アーキテクチャ(CDB)データベースのダウングレードに特に役立ちます。ダウングレードに使用されるリソースの数を制御する場合は、以前のリリースと同様にcatdwgrd.sqlスクリプトを手動で実行できます。以前のデータベース・リリースにダウングレードすると、その以前のリリースに対するリリース更新を適用できます。

dbdowngradeシェル・コマンドはファイル・パス$ORACLE_HOME/bin (LinuxおよびUNIXの場合)、および%ORACLE_HOME%¥bin (Windowsベースのシステムの場合)にあります。CDBをダウングレードする場合、スクリプトに引数として包含リストを指定できます。

マルチテナント・アーキテクチャ・データベース(CDB)をダウングレードする場合、dbdowngradeスクリプトには、包含リストを使用するかどうかに応じて2つの動作があります。

- 包含リストがない場合。ダウングレードは、CDBで開いているすべてのコンテナ(PDBおよびCDB)で実行されます。

  CDB全体をダウングレードする場合は、包含リストなしでダウングレードを実行します。このシナリオでは開いているコンテナがすべてダウングレードされます。dbdowngradeスクリプトを開始する前に、CDB内のすべてのPDBを手動で開く必要があります。

- 包含リストがある場合。ダウングレードは包含リスト内のPDBでのみ実行され、ダウングレード操作中にCDB$ROOTはダウングレードされません。

  包含リストにリストされているPDBのセットのみをダウングレードする場合は、包含リストでダウングレードを実行します。このシナリオでは、切断および接続のアップグレードを使用する場合、包含リストにリストされているPDBのセットのみがダウングレードされます。包含リストにないCDBおよびPDBは、その後のリリースではアップグレードされたままになります。

前提条件:

- Oracle Database 19cからOracle Database 18c、Oracle Database 12.2またはOracle Database 12.1にダウングレードする場合、マルチテナント・コンテナ・データベース(CDB)またはCDB内の1つのプラガブル・データベース(PDB)にあるすべてのデータベースをダウングレードできます。Oracle Database 12cより前のOracle Databaseリリースでは、マルチテナント・アーキテクチャを使用しませんでした。
- 包含リストなしでダウングレードする場合、dbdowngradeスクリプトを実行する前にすべてのPDBコンテナを開く必要があります。

1. Oracle DatabaseのOracleホーム・ディレクトリの所有者として、システムにログインします。
2. ORACLE_HOME環境変数を、ダウングレードするアップグレード済Oracle DatabaseリリースのOracleホームに設定します。
3. ORACLE_SID環境変数を、ダウングレードするOracle Databaseのシステム識別子(SID)に設定します。

> ノート:
>
> クラスタ・データベースをダウングレードしている場合は、データベースを完全に停止し、CLUSTER_DATABASE 初期化パラメータを FALSE に設定します。ダウングレードが完了した後、このパラメータの設定を TRUE に戻します。

4. SQL*Plusを使用して、SYSDBA権限を持つユーザーとして、ダウングレードするデータベース・インスタンスに接続します。

```
sqlplus sys as sysdba
```

```
Enter password: password
```

5. 使用するOracle Databaseインスタンス・タイプに応じて、SQL*Plusで次のコマンドを発行し、ダウングレード・モードでインスタンスを起動します。起動中のエラーの場合、PFILEオプションを使用して初期化パラメータ・ファイルの場所を指定するように求められる可能性があります。
   - 非CDBインスタンス:

     ```
     SQL> startup downgrade pfile=pfile_name
     ```

   - CDBインスタンス:

     ```
     SQL> startup downgrade pfile=pfile_name
     SQL> alter pluggable database all open downgrade;
     ```

   初期化パラメータ・ファイルPFILEの場所を指定します。

   参照:

   初期化パラメータの起動時の指定および初期化パラメータ・ファイルの詳細は、『Oracle Database管理者ガイド』を参照してください

6. デフォルト値または包含リストを使用してdbdowngradeスクリプトを起動します。

   包含リストを使用する場合は、CDB$ROOTを包含リストに含めないでください。

   たとえば:

   デフォルト値による実行

   LinuxおよびUnix

   ```
   $cd $ORACLE_HOME/bin
   $./dbdowngrade
   ```

   Microsoft Windows

   ```
   $cd %ORACLE_HOME%¥bin
   $dbdowngrade.cmd
   ```

   CDBの包含リストによる実行

   LinuxおよびUnix

   ```
   $cd $ORACLE_HOME/bin
   $./dbdowngrade -c 'PDB1 PDB2 PDBN'
   ```

   Microsoft Windows

   ```
   $cd %ORACLE_HOME%¥bin
   $dbdowngrade.cmd -c "PDB1 PDB2 PDBN"
   ```

   dbdowngradeスクリプト(catdwgrd.sqlを起動)の実行が終了した後、上位バージョンのOracleホームでデータベースを停止します。

7. 次の環境変数がダウングレード先のリリースのディレクトリを指定するように変更します。
   - ORACLE_HOME
   - PATH

   また、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、ダウングレードされた

Oracleホームを指していることを確認します。

ダウングレードを終了するには、下位バージョンのOracleホームでダウングレード再ロードを実行します。

8. 下位バージョンのOracleホームにあるデータベースをアップグレード・モードで起動します。非CDBデータベースで`catrelod.sql`を実行するか、または`catcon.pl`を使用してCDBデータベースで`catrelod.sql`を実行します。
   - 非CDBの場合:

     ```
     SQL> $ORACLE_HOME/rdbms/admin/catrelod.sql
     ```

   - CDBの場合:

     ```
      $ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl 1 -e -b
     catrelod -d $ORACLE_HOME/rdbms/admin catrelod.sql
     ```

   このコマンドにより、ダウングレードしたデータベース内の各データベース・コンポーネントに該当するバージョンが再ロードされます。

`dbdowngrade`スクリプトの実行結果として、ユーティリティは`catdwgrd`および`catcon.pl`を実行します。これらのスクリプトは、ダウングレードするリリースの推奨値を使用してダウングレードを実行します。ダウングレード・スクリプトによって生成されるログ・ファイルは、次の3つのオプションのいずれかで見つかった最初のディレクトリの下に順番に配置されます。

- `orabasehome`コマンドで識別されるOracleベース・ホーム
- `orabase`コマンドで識別されるOracleベース・ホーム
- `oracle_home`コマンドで識別されるOracleホーム

たとえば:

```
$ $ORACLE_HOME/bin/orabasehome
/u01/app/oracle/product/19.0.0/dbhome_1
```

この例では、$ORACLE_BASEディレクトリは`/u01/app/oracle/product/19.0.0/dbhome_1`で、ログは`/u01/app/oracle/product/19.0.0/dbhome_1/cfgtoollogs/downgrade`にあります。ディレクトリでは、ログ・ファイルの先頭に文字列`catdwgrd`が付きます。

ノート:

- 読取り/書込みの Oracle ホーム: コマンド `orabaseconfig` および `orabasehome` は、両方とも ORACLE_HOME の環境設定を返します。
- 読取り専用 Oracle ホーム:コマンド: `orabaseconfig` は、パス`$ORACLE_BASE/homes` 内の Oracle ベースの読取り専用パス構成を返します。

親トピック: スクリプトを使用したCDBまたは非CDBのOracle Databaseのダウングレード

catdwgrd.sqlを使用したCDBまたは非CDBのOracle Databaseの手動ダウングレード

手動スクリプトを実行する必要があるか、過度のスレッドの問題を回避する場合、手動で`catdwgrd.sql`スクリプトを実行して、サポートされているメジャー・リリースまたは以前のリリース更新にOracle Databaseをダウングレードできます。

Oracle Database 19cからOracle Database 18c、Oracle Database 12.2またはOracle Database 12.1にダウ

ングレードする場合、マルチテナント・コンテナ・データベース(CDB)またはCDB内の1つのプラガブル・データベース(PDB)にあるすべてのデータベースをダウングレードできます。Oracle Database 12cより前のOracle Databaseリリースでは、マルチテナント・アーキテクチャを使用しませんでした。

> ノート:
>
> Oracle Database 12c リリース 1 (12.1)以上では、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。

1. Oracle DatabaseのOracleホーム・ディレクトリの所有者として、システムにログインします。
2. ORACLE_HOME環境変数を、アップグレードしたOracle DatabaseリリースのOracleホームに設定します
3. ORACLE_SID環境変数を、ダウングレードするOracle Databaseのシステム識別子(SID)に設定します。
4. システム・プロンプトで、ORACLE_HOME/rdbms/adminディレクトリへ移動します(ORACLE_HOMEはシステムのOracleホームです)。

   > ノート:
   >
   > クラスタ・データベースをダウングレードしている場合は、データベースを完全に停止し、CLUSTER_DATABASE 初期化パラメータを FALSE に設定します。ダウングレードが完了した後、このパラメータの設定を TRUE に戻します。

5. SQL*Plusを使用して、SYSDBA権限を持つユーザーとして、ダウングレードするデータベース・インスタンスに接続します。

   ```
   sqlplus sys as sysdba
   Enter password: password
   ```

6. 使用するOracle Databaseインスタンス・タイプに応じて、SQL*Plusで次のコマンドを発行し、ダウングレード・モードでインスタンスを起動します。起動中のエラーの場合、PFILEオプションを使用して初期化パラメータ・ファイルの場所を指定するように求められる可能性があります。
   - 非CDBインスタンス:

     ```
     SQL> startup downgrade pfile=pfile_name
     ```

   - CDBインスタンス:

     ```
     SQL> startup downgrade pfile=pfile_name
     SQL> alter pluggable database all open downgrade;
     ```

   初期化パラメータ・ファイルPFILEの場所を指定します。

   参照:

   初期化パラメータの起動時の指定および初期化パラメータ・ファイルの詳細は、『Oracle Database管理者ガイド』を参照してください

7. (推奨)非CDBをダウングレードする場合は、変更および問題を追跡できるよう、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することをお薦めします。

CDBをダウングレードしている場合、次のステップを実行する必要はありません。CDBは自動的に出力を catcon_logsにスプールします。

CDB以外の場合は、次のコマンドを入力して結果をログ・ファイルにスプールします(downgrade.logはログ・ファイルの名前です)。

```
set echo on termout on serveroutput on timing on
SQL> SPOOL downgrade.log
```

8. 構成に応じて、次のコマンドを使用してダウングレードを開始します。
   - 非CDB:

     ```
     SQL> @catdwgrd.sql
     ```

   - CDB:

     ```
     $ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl -d
     $ORACLE_HOME/rdbms/admin -e -b catdwgrd -l output directory -r
     catdwgrd.sql
     ```

CDBの例で、catdwgrd.sqlはcatcon.plを使用してコンテナに対して実行されます。catcon.plユーティリティを使用してコマンドを実行するには、まずPerlを起動します。-dパラメータは、catcon.plにcatdwgrdの入手先を指定します。-lパラメータは、ログ・ファイルの出力ディレクトリを指定します(rdbms/adminディレクトリに書き込むかわりとなります)。-rパラメータを指定すると、catdwgrdはまずPDBで実行され、次にCDB_ROOTで実行されます。

CDBをダウングレードするときには-rパラメータを使用してcatdwgrdを実行してください。–rパラメータはスクリプトが実行されるデフォルトの順番を変更し、まずすべてのPDBでスクリプトが実行され、その後にCDB_ROOTで実行されます。

ノート:

- Oracle Database リリースに付属する catdwgrd.sql スクリプトのバージョンを使用します。
- CDB をダウングレードするときには-r パラメータを使用して catdwgrd を実行してください。
- Oracle Database リリース環境で catdwgrd.sql を実行します。
- catdwgrd.sql スクリプトは Oracle Database のデータベース内のすべてのコンポーネントを、アップグレード前のリリースにダウングレードします。ダウングレードはアップグレード前のサポートされているメジャー・リリースか、またはアップグレード前のパッチ・リリースのいずれかです。

マルチテナント環境のデータベースにダウングレードしており、catdwgrd.sqlコマンドでエラーが発生した場合は、エラー・メッセージを確認します。続行する前に、CDB$ROOTまたはPDBにどのような問題があるかを調べます。「Oracle Databaseのダウングレードのトラブルシューティング」の項を確認します。エラーで指摘された問題を修正します。エラーを解決した後、構文catcon.pl -c 'cdb,pdb' -rを使用して、catcon.plユーティリティを指定したcatdgwrd.sqを再実行します。

注意:

コンポーネントのダウングレードが失敗すると、ORA-39709 エラーが表示されます。Oracle Database データ・ディレクトリをダウングレードせずに SQL*Plus セッションが終了します。すべてのコンポーネントは、Oracle Database のデータ・ディクショナリをダウングレードする前に、正常にダウングレードする必要があります。catdwgrd.sql スクリプトを再実行する前に、問題を特定して修正します。

9. 非CDBの場合のみ、スプールをオンにしていた場合は、スクリプト結果のログ・ファイルへのスプーリングをオフにします。

```
SQL> SPOOL OFF
```

スプール・ファイルを確認して、ダウングレード中にエラーが発生しなかったかどうかを検証します。ステップ8で、スプール・ファイル名をdowngrade.logとしました。このファイルで検出された問題は、すべて修正します。必要に応じて、ダウングレード・スクリプトを再実行します。

ノート:

ダウングレード・スクリプトの最初の実行時から結果を保存します。ダウングレード・スクリプトを再実行する前に、downgrade.log ファイルの名前を変更し、スクリプトを再実行したときにファイルが上書きされないようにします。

10. インスタンスを停止します。

```
SQL> SHUTDOWN IMMEDIATE
```

11. SQL*Plusを終了します。
12. 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、次の環境変数がダウングレード先のリリースのディレクトリを指定するように変更してください。
    - LinuxおよびUNIXシステムの場合

      次の環境変数がダウングレード先のリリースのディレクトリを指定するように変更します。

      - ORACLE_HOME
      - PATH

      また、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、ダウングレードされたOracleホームを指していることを確認します。

      参照:

      ご使用のオペレーティング・システムでのその他の重要な環境変数の設定に関する詳細は、ご使用のオペレーティング・システムの『Oracle Databaseインストレーション・ガイド』を参照してください

    - Microsoft Windowsシステム

      - Oracle Database 12cデータベースのすべてのOracleサービス(OracleServiceSID Oracleサービスなど)を停止します(SIDはインスタンスの名前)。

        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
        ```

参照:

WindowsでのOracleサービスの停止の詳細は、『Oracle Database管理者リファレンスfor Microsoft Windows』を参照してください

- コマンド・プロンプトでコマンドORADIMを発行して、Oracleサービスを削除します。

  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
```

  コマンド・プロンプトでコマンドORADIMを使用して、ダウングレードするデータベースのOracleサービスを作成します。

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

  ORADIMの構文には次の変数が含まれます。

変数	説明
SID	ダウングレードするデータベースの SID と同じシステム識別子(SID)名です。
PASSWORD	データベース・インスタンスのパスワードです。このパスワードは SYSDBA 権限で接続ユーザー用のパスワードです。-INTPWD オプションは必須ではありません。パスワー力を求められた場合は、この Windows プラットフォームの標準のユーザー・アカウンスワードを使用します。
USERS	SYSDBA 権限および SYSOPER 権限を付与できるユーザーの最大数です。
ORACLE_HOME	ダウングレード先のデータベースの Oracle ホーム・ディレクトリです。-PFILE オプシ使用して、(Oracle ホーム・ディレクトリがマウントされているドライブを含む)フルパス指定する必要があります。 初期化パラメータの起動時の指定および初期化パラメータ・ファイルの詳細は、『*Oracle Database* 管理者ガイド』を参照してください。

  たとえば、SIDがORCL、PASSWORDがTWxy5791、USERSの最大数が10、ORACLE_HOMEディレクトリがC:¥ORANTの場合、次のコマンドを入力します。

```
C:¥> ORADIM -NEW -SID ORCL -INTPWD TWxy5791 -MAXUSERS 10
     -STARTMODE AUTO -PFILE C:¥ORANT¥DATABASE¥INITORCL.ORA
```

ノート:

ORADIM コマンドによって、Oracle ホームのユーザー・アカウントのパスワードの入力が求められます。ORADIM を使用して、他のオプションを指定できます。

データベースをダウングレードする際に、Windows レジストリの設定を変更する必要はありません。必要な変更は、すべて ORADIM ユーティリティによって自動的に行われます。

参照:

ORADIMを使用したOracle Databaseインスタンスの管理の詳細は、Oracle Database管理者のリファレンスfor Microsoft Windowsを参照してください

13. ダウングレード先のリリースの構成ファイル(パラメータ・ファイル、パスワード・ファイルなど)をリストアします。

データベースがOracle RACデータベースの場合は、次のコマンドを実行して、データベースを単一インスタンス・モードに戻します。

```
SET CLUSTER_DATABASE=FALSE
```

ノート:

クラスタ・データベースをダウングレードしている場合は、このクラスタ・データベースのインスタンスが構成されているすべてのノードでこのステップを実行してください。初期化パラメータ CLUSTER_DATABASE の値を FALSE に設定します。ダウングレードが完了した後、この初期化パラメータの設定を TRUE に戻します。

参照:

Oracle RACでの初期化パラメータの使用の詳細は、『Oracle Real Application Clusters管理およびデプロイメント・ガイド』を参照してください

14. システム・プロンプトで、ダウンロード先である以前のリリースのOracleホーム・ディレクトリのadminディレクトリに変更します。(ORACLE_HOME/rdbms/admin。ORACLE_HOMEは以前のリリースのOracleホームのパスです。)

15. SQL*Plusを開始して、SYSDBA権限を持つユーザーとして、データベース・インスタンスに接続します。

- 非CDBの場合:

```
SQL> CONNECT / AS SYSDBA
SQL> STARTUP UPGRADE
```

- CDBの場合:

```
connect / as sysdba
startup upgrade;
alter pluggable database all open upgrade;
```

16. (オプション) 非CDBの場合、変更および問題を追跡するために、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。このステップはCDBには必要ありません。

```
SQL> SPOOL reload.log
```

17. 非CDBデータベースでcatrelod.sqlを実行するか、またはcatcon.plを使用してCDBデータベースでcatrelod.sqlを実行します。

- 非CDBの場合:

```
SQL> $ORACLE_HOME/rdbms/admin/catrelod.sql
```

- CDBの場合:

```
 $ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl -n 1 -e -b
catrelod -d $ORACLE_HOME/rdbms/admin catrelod.sql
```

このコマンドにより、ダウングレードしたデータベース内の各データベース・コンポーネントに該当するバージョンが再ロードされます。

18. 非CDBでスプーリングをオンにした場合、スクリプト結果のログ・ファイルへのスプーリングをオフにします。

```
SQL> SPOOL OFF
```

スプール・ファイルをチェックし、パッケージとプロシージャのコンパイルが成功したかどうかを確認します。このログ・ファイルで見つけた問題を処理し、必要に応じて適切なスクリプトを再実行します。

19. 通常の操作のために、インスタンスを停止し、再起動します。

```
SQL> SHUTDOWN IMMEDIATE
SQL> STARTUP
```

オプションPFILEを使用して、初期化パラメータ・ファイルの場所を指定するように求められる場合があります。

参照:

初期化パラメータの起動時の指定および初期化パラメータ・ファイルの詳細は、『Oracle Database管理者ガイド』を参照してください

20. Oracle Label Securityを使用するようにデータベースを構成した場合、このステップを完了します。Oracle Label Securityを使用するようにデータベースを構成しなかった場合、次のステップに進みます。
    - `olstrig.sql`スクリプトを、Oracle Database 12cのOracleホームから、データベースをダウングレードするリリース番号のOracleホームにコピーします。
    - ダウングレード・リリースのOracleホームから`olstrig.sql`を実行し、Oracle Label Securityのポリシーを適用して、表のDMLトリガーを再作成します。

```
SQL> @olstrig.sql
```

21. (オプション) 非CDBの場合、変更および問題を追跡するために、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。このステップはCDBには必要ありません。

例:

```
SQL> SPOOL utlrp.log
```

22. `utlrp.sql`スクリプトを実行して、残りのすべてのストアドPL/SQLおよびJavaコードを再コンパイルします。使用している構成に対応する手順を使用します。
    - 非CDB:

```
SQL> $ORACLE_HOME/rdbms/admin/utlrp.sql
```

- CDB:

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

utlrp.sqlスクリプトは、以前INVALID状態だったパッケージ、プロシージャ、型などのすべての既存のPL/SQLモジュールを、再コンパイルします。ログ・ファイルutlrp0.logが生成されます。そのログ・ファイルには再コンパイルの結果がリストされます。

23. utlrp.sqlを実行するときに非CDBのスプーリングをオフにする場合は、ログ・ファイルへのスクリプトの結果のスプールをオフにします。

```
SQL> SPOOL OFF
```

スプール・ファイルをチェックし、パッケージとプロシージャのコンパイルが成功したかどうかを確認します。このログ・ファイルで検出された問題は、すべて修正します。必要に応じて、該当するスクリプトを再実行します。

24. SQL*Plusを終了します。

このプロシージャの完了時に、データベースがダウングレードされます。

関連項目

- Oracle Databaseのダウングレードのトラブルシューティング
- Oracle Database管理者ガイド

親トピック: スクリプトを使用したCDBまたは非CDBのOracle Databaseのダウングレード

単一のプラガブルOracle Database (PDB)のダウングレード

Oracle Databaseをダウングレードする場合、CDB全体をダウングレードせずに1つのPDBをダウングレードできます。

Oracle Database 12cリリース2 (12.2)より新しいOracle Databaseリリースでは、個々のPDBをダウングレードできます。たとえば、アップグレードしたCDBからPDBを切断し、PDBをダウングレードした後、これを以前のリリースのCDBに接続するか、そのPDBデータベースをスタンドアロン・データベースに変換することができます。

PDBのダウングレード

この手順の例では、PDBをリリース12.1.0.2にダウングレードします。

1. PDBをDOWNGRADEモードで起動します。これを行うとき、CDBは標準モードになります。

```
SQL> alter pluggable database CDB1_PDB1 open downgrade;
```

2. dbdowngradeユーティリティを使用するか、catcon.plを使用してcatdwgrdを手動で実行して、PDBをダウングレードします。

   これらの各オプションでは、ダウングレードするPDBはPDB1です。

   - dbdowngradeユーティリティを使用してダウングレードします。

     次のようにdbdowngradeスクリプトを使用してPDBをダウングレードします。

```
cd $ORACLE_HOME/bin
./dbdowngrade -c 'PDB1'
```

   - catcon.pを使用して、catdwgrdによって手動でダウングレードします。

     次のようにcatdwgrdを実行します。

```
$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl -d
```

```
 $ORACLE_HOME/rdbms/admin -n 1 -l <output directory> -e -b catdwgrd -c
'PDB1'
 catdwgrd.sql
```

例では、catdwgrdはcatcon.plとともに実行されます。-dパラメータは、catcon.plにcatdwgrdの入手先を指定します。-lパラメータは、ログ・ファイルの出力ディレクトリを指定します(rdbms/adminディレクトリに書き込むかわりとなります)。2つのスクリプトを同時に一緒に実行するには、-rパラメータを使用する必要があります。

ダウングレードのログ・ファイルは、Oracleベース・ホーム(コマンドorabasehomeまたはorabaseで識別されるOracleベースか、コマンドoracle_homeで識別されるOracleホーム)にこの順番で配置されます。

3. PDBをクローズします。

PDBのCDBからの切断

このステップでは、ダウングレードされたPDBをリリース12.2.0.1のCDBから切断します。

1. アップグレードしたCDBに接続します。
2. 切断するPDBをクローズします。

```
SQL> alter pluggable database PDB1 close;
```

3. ダウングレードした12.1.0.2 PDBを切断し、変数pathシステムのパスに置き換えます。

```
SQL> alter pluggable database PDB1 unplug into 'path/pdb1.xml';
```

切断が完了すると、次のレスポンスが表示されます。

```
Pluggable database altered
```

ダウングレードされた12.1.0.2 PDBの接続

このステップでは、ダウングレードされた12.1.0.2のPDBを12.1.0.2のCDBに接続します。これを行うには、このCDBでPDBを作成する必要があります。次の例は、PDB1というプラガブル・データベースの作成方法を示しています。

1. 12.1.0.2のCDBに接続します。
2. 12.1.0.2のPDBを接続します。

```
SQL> create pluggable database PDB1 using 'path/pdb1.xml';
```

このコマンドを実行すると、「プラガブル・データベースが作成されました。」が返されます。

3. PDBをアップグレード・モードで開きます。

```
SQL> alter pluggable database PDB1 open upgrade;
```

4. PDBに接続します。

```
SQL> alter session set container=PDB1;
```

5. PDBでcatrelodを実行します。

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

catrelod.sqlスクリプトによって、ダウングレードしたデータベース内の各データベース・コンポーネントに該当するバージョンが再ロードされます。

6. PDBでutlrpを実行します。

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

utlrp.sqlスクリプトは、以前INVALID状態だったパッケージ、プロシージャ、型などのすべての既存のPL/SQLモジュールを、再コンパイルします。

親トピック: 以前のリリースへのOracle Databaseのダウングレード

Oracle Application Expressが含まれるPDBのダウングレード

このプロシージャを使用すると、Oracle Application Expressが含まれるPDBをダウングレードするときにINVALID OBJECTS OWNED BY APEX_050000エラーを回避できます。

PDBを以前のリリースにダウングレードした後、次のようなSQL文を入力してOracle Application Expressユーザーを削除します。

```
$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl -b drop_apex5
-c 'PDB1' -- --x'drop user apex_050000 cascade'
```

この例では、PDB名は「PDB1」です。

親トピック: 以前のリリースへのOracle Databaseのダウングレード

Oracle Databaseをダウングレードする場合のダウングレード後の作業

Oracle Databaseのリリースをダウングレードした後、互換性、コンポーネントおよびサポートされるプロトコルに影響する変更により、追加のタスクを完了する必要がある場合があります。

- ダウングレード後のリリース更新およびその他のパッチの再適用
  アップグレード後かつダウングレード前に、元のOracleホームに新しいパッチをインストールしていた場合は、ダウングレードが実行されて、catrelod.sqlが正常に完了したら、インストールしたパッチが適用されていることを確認します。
- Oracle Databaseのダウングレード後のOracle RACの再有効化
  ダウングレード後、Oracle Real Application Clusters (Oracle RAC)を再有効化できます。
- Oracle Databaseのダウングレード後のOracle Database Vaultの再有効化
  ダウングレード時にOracle Database Vaultを無効化するよう指示があった場合、これを実行する必要があります。
- Oracle Clusterwareの構成のリストア
  構成をリストアするには、アップグレード前のリリースをリストアする必要があります。
- Oracle Databaseのダウングレード後のOracle Enterprise Managerのリストア
  この項で説明されているリストア作業が必要になるのは、ダウングレードを実行する際に、Oracle Enterprise Managerがホストで構成されている場合のみです。
- 以前のリリースへのOracle APEXのリストア
  Oracle Databaseのアップグレードと同時にOracle APEX (以前のOracle Application Express)もアップグレードした場合、ダウングレード後に以前のリリースに戻すステップを実行する必要があります。
- ダウングレード後のディクショナリ統計の収集
  ダウンロード後に高いパフォーマンスを保証するためには、次のプロシージャを使用してディクショナリ統計を収集します。
- ダウングレード後の修正したオブジェクト統計の再収集
  ダウングレード後に、Oracle Databaseで代表的なワークロードを実行し、修正されたオブジェクト統計を再収集します。

- ダウングレード後の失効しているCBO統計の再収集
  Oracle Databaseのダウングレードが完了した後に、Oracle Cost-Based Optimizer (CBO)統計を再収集することをお薦めします。
- ダウングレード後のレジストリ・コンポーネントの有効性の確認
  レジストリ・コンポーネントの有効性を確認し、無効なコンポーネントを特定します。

親トピック: 以前のリリースへのOracle Databaseのダウングレード

ダウングレード後のリリース更新およびその他のパッチの再適用

アップグレード後かつダウングレード前に、元のOracleホームに新しいパッチをインストールしていた場合は、ダウングレードが実行されて、`catrelod.sql`が正常に完了したら、インストールしたパッチが適用されていることを確認します。

新しいパッチをインストールした場合、`datapatch`ツールを実行して、ダウングレードしたデータベースにこれらのパッチを適用します。アップグレード後にOracleホームのバイナリおよびファイルを変更しなかった場合、`catrelod.sql`を実行した後に`datapatch`を実行する必要はありません。ただし、新しいパッチがインストールされているかどうか疑問がある場合は、`datapatch`を実行します。パッチがデータベースに適用されていることを確認するために必要な回数だけ`datapatch`を実行できない安全上の懸念はありません。

親トピック: Oracle Databaseのダウングレードのダウングレード後の作業

Oracle Databaseのダウングレード後のOracle RACの再有効化

ダウングレード後に、Oracle Real Application Clusters (Oracle RAC)を再度有効にできます。

ダウングレードに`CLUSTER_DATABASE=FALSE`を設定したOracle RACデータベースでのダウングレードでは、`CLUSTER_DATABASE=TRUE`を再度設定し、RACクラスタ内のすべてのインスタンスから開始できるようになりました。

親トピック: Oracle Databaseのダウングレードのダウングレード後の作業

Oracle Databaseのダウングレード後のOracle Database Vaultの再有効化

ダウングレード時にOracle Database Vaultを無効化するよう指示があった場合、これを実行する必要があります。

Oracle Database Vaultを使用している場合、ダウングレードする前にこれを無効にするように指示されている可能性があります。ダウングレード後にOracle Database Vaultを使用するには、登録して再度有効にする必要があります。

関連項目

- Oracle Database Vaultの登録

親トピック: Oracle Databaseのダウングレードのダウングレード後の作業

Oracle Clusterwareの構成のリストア

構成をリストアするには、アップグレード前のリリースをリストアする必要があります。

Oracle Clusterware構成をOracle Clusterwareのアップグレード前の状態にリストアできます。新しいOracle Databaseのアップグレード処理中またはアップグレード処理後に実行した構成変更は削除され、リカバリできません。

親トピック: Oracle Databaseのダウングレードのダウングレード後の作業

Oracle Databaseのダウングレード後のOracle Enterprise Managerのリストア

この項で説明されているリストア作業が必要になるのは、ダウングレードを実行する際に、Oracle Enterprise Managerがホストで構成されている場合のみです。

Oracle Enterprise Managerをリストアするには、最初にOracle Enterprise Manager Configuration Assistant (EMCA)を実行してから、emdwgrdユーティリティを実行します。

- ダウングレード後のOracle Enterprise Managerのリストアの要件
  12.1より前のリリースへのダウングレード後にOracle Enterprise Managerをリストアできるようにするには、アップグレード前にこれらの要件を満たす必要があります。
- EMCAの実行によるダウングレード後のOracle Enterprise Managerのリストア
  これらのトピックを参照して、ダウングレード後にOracle Enterprise Managerをリストアするためのリストア・シナリオを選択します。
- Enterprise Manager Database Controlをリストアするためのemdwgrdユーティリティの実行
  Oracle Enterprise Manager Database Controlおよびデータをリストアするには、emca -restoreの実行後にemdwgrdユーティリティを使用します。

親トピック: Oracle Databaseのダウングレードのダウングレード後の作業

ダウングレード後のOracle Enterprise Managerのリストアの要件

12.1より前のリリースへのダウングレード後にOracle Enterprise Managerをリストアできるようにするには、アップグレード前にこれらの要件を満たす必要があります。

emca -restoreを使用してOracle Enterprise Managerを以前の状態にリストアするには、次の要件を満たす必要があります。

- アップグレードの前に、Oracle Enterprise Managerの構成ファイルおよびデータのバックアップを保存していること
- この手順で新しいOracle Databaseリリースのホームに存在するemcaバイナリを実行すること

Oracle Clusterwareシステムで、Oracle RACデータベースのOracle Enterprise Managerをリストアするには、emca -restoreを実行する前にsrvctlを使用してデータベースを登録しておく必要があります。emca -restoreは、データベースのダウングレード先となる以前のOracle DatabaseリリースのORACLE_HOME/binディレクトリから実行する必要があります。

適切なオプションを指定してemca -restoreコマンドを実行し、Oracle Enterprise Manager Database ControlまたはOracle Enterprise Manager Grid Controlを古いOracleホームにリストアします。

ダウングレードするデータベースが単一インスタンス・データベースか、Oracle RACデータベースか、Oracle ASMデータベースかによって、emcaの異なるオプションを指定します。

関連項目

- Oracle Clusterware管理およびデプロイメント・ガイド

親トピック: Oracle Databaseのダウングレード後のOracle Enterprise Managerのリストア

EMCAの実行によるダウングレード後のOracle Enterprise Managerのリストア

これらのトピックを参照して、ダウングレード後にOracle Enterprise Managerをリストアするためのリストア・シナリオを選択します。

- Oracle ASMがない単一インスタンスのOracle Databaseでのemcaの実行
  Enterprise Manager Configuration Assistant (emca)を使用してデータベースを管理します。
- Oracle ASMがないOracle RACデータベースでのEMCAの実行
  Enterprise Manager Configuration Assistant (emca)を使用してデータベースを管理します。
- 単一インスタンスのOracle ASMインスタンスでのEMCAの実行
  Enterprise Manager Configuration Assistant (emca)を使用してデータベースとストレージを管理します。
- Oracle RACインスタンス上のOracle ASMでのemcaの実行
  Enterprise Manager Configuration Assistant (emca)を使用してデータベースとストレージを管理します。
- Oracle ASMがある単一インスタンスのOracle Databaseでのemcaの実行
  Enterprise Manager Configuration Assistant (emca)を使用してデータベースとストレージを管理します。
- Oracle RACデータベースおよびOracle ASMインスタンスでのemcaの実行
  Enterprise Manager Configuration Assistant (emca)を使用してデータベースとストレージを管理します。

親トピック: Oracle Databaseのダウングレード後のOracle Enterprise Managerのリストア

Oracle ASMがない単一インスタンスのOracle Databaseでのemcaの実行

Enterprise Manager Configuration Assistant (emca)を使用してデータベースを管理します。

このコマンドを使用してEnterprise Manager Configuration Assistantを実行します。

```
ORACLE_HOME/bin/emca -restore db
```

次の情報の入力を求められます。

- リストアするデータベースのOracleホーム
- データベースのSID
- リスナー・ポート番号

親トピック: EMCAの実行によるダウングレード後のOracle Enterprise Managerのリストア

Oracle ASMがないOracle RACデータベースでのEMCAの実行

Enterprise Manager Configuration Assistant (emca)を使用してデータベースを管理します。

この手順を使用してEnterprise Manager Configuration Assistantを実行します。

```
ORACLE_HOME/bin/emca -restore db -cluster
```

次の情報の入力を求められます。

- リストアするデータベースのOracleホーム
- データベースの一意の名前
- リスナー・ポート番号

親トピック: EMCAの実行によるダウングレード後のOracle Enterprise Managerのリストア

単一インスタンスのOracle ASMインスタンスでのEMCAの実行

Enterprise Manager Configuration Assistant (emca)を使用してデータベースとストレージを管理します。

このコマンドを使用してEnterprise Manager Configuration Assistantを実行します。

```
ORACLE_HOME/bin/emca -restore asm
```

次の情報の入力を求められます。

- リストアするデータベースのOracleホーム
- Oracle ASMポート
- Oracle ASM SID

親トピック: EMCAの実行によるダウングレード後のOracle Enterprise Managerのリストア

Oracle RACインスタンス上のOracle ASMでのemcaの実行

Enterprise Manager Configuration Assistant (emca)を使用してデータベースとストレージを管理します。

このコマンドを使用してEnterprise Manager Configuration Assistantを実行します。

```
ORACLE_HOME/bin/emca -restore asm -cluster
```

次の情報の入力を求められます。

- リストアするデータベースのOracleホーム
- Oracle ASMポート

親トピック: EMCAの実行によるダウングレード後のOracle Enterprise Managerのリストア

Oracle ASMがある単一インスタンスのOracle Databaseでのemcaの実行

Enterprise Manager Configuration Assistant (emca)を使用してデータベースとストレージを管理します。

このコマンドを使用してEnterprise Manager Configuration Assistantを実行します。

```
ORACLE_HOME/bin/emca -restore db_asm
```

次の情報の入力を求められます。

- リストアするOracle DatabaseのOracleホーム
- データベースのSID
- リスナー・ポート番号
- Oracle ASMポート
- Oracle ASMホーム
- Oracle ASM SID [+ASM]

親トピック: EMCAの実行によるダウングレード後のOracle Enterprise Managerのリストア

Oracle RACデータベースおよびOracle ASMインスタンスでのemcaの実行

Enterprise Manager Configuration Assistant (emca)を使用してデータベースとストレージを管理します。

このコマンドを使用してEnterprise Manager Configuration Assistantを実行します。

```
ORACLE_HOME/bin/emca -restore db_asm -cluster
```

次の情報の入力を求められます。

- リストアするデータベースのOracleホーム
- データベースの一意の名前

- リスナー・ポート番号
- Oracle ASMポート
- Oracle ASM Oracleホーム
- Oracle ASM SID [+ASM]

emcaの出力は、指定するオプションおよびプロンプトで入力する値によって様々です。Oracle RAC環境では、Oracle RACのすべてのクラスタ・メンバー・ノードでこのステップを繰り返す必要があります。

ここで、Oracle Enterprise Manager Database Controlおよびデータをリストアするためにemdwgrdユーティリティを実行する必要があります。

親トピック: EMCAの実行によるダウングレード後のOracle Enterprise Managerのリストア

Enterprise Manager Database Controlをリストアするためのemdwgrdユーティリティの実行

Oracle Enterprise Manager Database Controlおよびデータをリストアするには、emca -restoreの実行後にemdwgrdユーティリティを使用します。

emdwgrdを使用するには、次の操作を実行する必要があります。

- ORACLE_HOMEおよび他の環境変数を、アップグレードが実行されたOracleホームを指すように設定します。
- 新しいOracle DatabaseリリースのOracleホームからemdwgrdユーティリティを実行します。

次の手順は、LinuxおよびUNIX用です。これをWindowsで実行する場合は、emdwgrdをemdwgrd.batに置き換えてください。

1. ORACLE_HOMEを、データベースがアップグレードされる前のOracleホームに設定します。
2. ORACLE_SIDを、アップグレード後にダウングレードされたデータベースのSIDに設定します。
3. データベースがアップグレードされる前のOracleホームを指すように、PATH、LD_LIBRARY_PATHおよびSHLIB_PATHを設定します。
4. 新しいOracle DatabaseリリースのOracleホームに移動します。

   ```
   cd $ORACLE_HOME/bin
   ```

5. 次のいずれかの手順を使用してemdwgrdを実行します。

   a. 単一インスタンス・データベースの場合は、次のコマンドを実行します。SIDはアップグレード後にダウングレードされたデータベースのSIDで、save_directoryは、Database Controlのファイルおよびデータの保存時に選択した記憶域の場所へのパスです。

   ```
   emdwgrd -restore -sid SID -path save_directory -tempTablespace TEMP
   ```

   b. Oracle RACデータベースの場合、クラスタ・ノード間のリモート・コピーが必要です。構成済のリモート・コピーを示す環境変数を定義します。たとえば:

   ```
   setenv EM_REMCP /usr/bin/scp
   ```

   その後、次のオプションを使用してemdwgrd -restoreを実行します。

   ```
   emdwgrd -restore -tempTablespace TEMP -cluster -sid SID_OldHome -path
   save_directory
   ```

Oracleホームが共有デバイス上にある場合、-sharedをemdwgrdコマンドのオプションに追加します。

6. emdwgrdにより要求された場合は、SYSおよびSYSMANパスワードを入力します。

7. emdwgrdが終了すると、Oracle Enterprise Manager Database Controlは古いOracleホームにダウングレードされます。

親トピック: Oracle DatabaseのダウングレードのOracle Enterprise Managerのリストア

以前のリリースへのOracle APEXのリストア

Oracle Databaseのアップグレードと同時にOracle APEX (以前のOracle Application Express)もアップグレードした場合、ダウングレード後に以前のリリースに戻すステップを実行する必要があります。

データベースのダウングレード後にOracle APEXのダウングレードを完了するには、『*Oracle Application Express*インストレーション・ガイド』にリストされたすべてのステップを実行して、お使いのOracle APEXのリリースを以前のリリースに戻します。戻すためのステップは、お使いのアーキテクチャが非CDBか、またはマルチテナント・アーキテクチャ(CDB) Oracle Databaseかによって異なります。

ノート:

データベースをアップグレードしたとき、同時に Oracle APEX をアップグレードした場合、実行する必要があるのはこのステップのみです。

関連項目

- Oracle Application Expressインストレーション・ガイド

親トピック: Oracle DatabaseのダウングレードのダウングレードのDowngrade後の作業

ダウングレード後のディクショナリ統計の収集

ダウンロード後に高いパフォーマンスを保証するためには、次のプロシージャを使用してディクショナリ統計を収集します。

ダウングレードしたリリースのデータ・ディクショナリ表の統計が収集されるように、データベースのダウングレード後にディクショナリ統計を収集することをお薦めします。

ノート:

ダウングレード・プロセスの後、データ・ディクショナリにはダウングレード済のディクショナリに保持されている変更が存在する可能性があることに注意してください。これらの変更は重要ではありません。ダウングレード済のデータ・ディクショナリは、以前のリリースのデータ・ディクショナリと機能的に同等です。

- 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薦めします。たとえば、次のSQL文を入力します。

```
SQL> EXEC DBMS_STATS.GATHER_DICTIONARY_STATS;
```

- CDB (マルチテナント・アーキテクチャ) Oracle Database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薦めします。

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。

関連項目

- 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のダウングレードのダウングレード後の作業

ダウングレード後の修正したオブジェクト統計の再収集

ダウングレード後に、Oracle Databaseで代表的なワークロードを実行し、修正されたオブジェクト統計を再収集します。

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$を通じて定義されます。ダウングレード後に、修正したオブジェクト統計を再収集して、リストアしたデータベースのオプティマイザが正しい実行計画を生成できるようにします。こうした実行計画により、データベースのパフォーマンスを改善できます。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、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/spfileおよび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。

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;
```

参照:

GATHER_FIXED_OBJECTS_STATSプロシージャの使用の詳細は、『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のダウングレードのダウングレード後の作業

ダウングレード後の失効しているCBO統計の再収集

Oracle Databaseのダウングレードが完了した後に、Oracle Cost-Based Optimizer (CBO)統計を再収集することをお薦めします。

Oracle Databaseをアップグレードし、新しいCBO統計を収集すると、アップグレードしたデータベースは新しいデータベース統計を持ちます。アップグレードしたデータベースは新しいヒストグラム・タイプも含むことができます。このため、データベースをダウングレードしたときには、新しいリリースで収集した統計は前のリリースとは異なる可能性があります。この問題はデータ・ディクショナリ表と一般のユーザー表の両方に当てはまります。

GATHER_DATABASE_STATSか、または通常、ディクショナリおよびアプリケーション・スキーマで失効している統計を更新す

るのに使用する収集コマンドを使用して、失効している統計を再収集します。

たとえば:

- 非CDBのOracle Databaseの場合: 統計を再収集するには、GATHER_DATABASE_STATSプロシージャを、オプション'GATHER STALE'を指定して使用することをお薦めします。たとえば:

  ```
  SQL> execute dbms_stats.gather_database_stats(options=>'GATHER STALE');
  ```

- CDB (マルチテナント・アーキテクチャ) Oracle Databaseの場合: マルチテナント・アーキテクチャ全体にわたるデータ・ディクショナリ統計を再収集するには、catconの使用をお薦めします。

  コンテナ・データベースのすべてのPDBの失効している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_database_stats(options=>'GATHER
  STALE')"
  ```

  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_database_stats(options=>'GATHER STALE')"
  ```

  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。

関連項目

- 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のダウングレードのダウングレード後の作業

ダウングレード後のレジストリ・コンポーネントの有効性の確認

レジストリ・コンポーネントの有効性を確認し、無効なコンポーネントを特定します。

ダウングレード後に、データベース内のコンポーネントの状態を確認します。完全なCDBをダウングレードする場合は、CDB_REGISTRYビューを使用できます。DBA_REGISTRYを使用して個々のPDBを確認することもできます。すべてのコンポーネントがVALIDまたはOPTION OFFのいずれかであることを確認します。

例8-1 CDB_REGISTRYビューを使用したCDBでのレジストリの確認

```
set line 200
set pages 1000
col COMP_ID format a8
col COMP_NAME format a34
col SCHEMA format a12
col STATUS format a10
col VERSION format a12
col CON_ID format 99
select CON_ID, COMP_ID, comp_name, schema, status, version from CDB_REGISTRY order by
1,2;
```

例8-2 DBA_REGISTRYビューを使用した非CDBまたはPDBでのレジストリの確認

```
set line 200
set pages 1000
col COMP_ID format a8
```

```
col COMP_NAME format a34
col SCHEMA format a12
col STATUS format a10
col VERSION format a12
col CON_ID format 99
select CON_ID, COMP_ID, comp_name, schema, status, version from DBA_REGISTRY order by
1,2;
```

関連項目

- Upgrade Your Database Now!のDatabase Migration from non-CDB to PDB – The Component Pitfall
- CHECK_COMPONENTS.SQL Script - Displays installed components from DBA_REGISTRY

親トピック: Oracle Databaseのダウングレードのダウングレード後の作業

Oracle Databaseのダウングレードのトラブルシューティング

このトラブルシューティング情報を使用して、Oracle Databaseのダウングレード時に発生する可能性のある問題に対処します。

この項には、ダウングレード時に発生する可能性のある既知のエラーと、それらのエラーに対処するための回避策が含まれます。

- catdwgrd.sqlスクリプトを使用したOracle Databaseコンポーネントのダウングレード中のエラー
  この項を使用して、ダウングレード中に`catdwgrd.sql`スクリプトを実行するときに発生するエラー(`ORA-20001: ダウングレードを続行できませんなど`)のトラブルシューティングを行います。
- アップグレードの成功または失敗後のOracle Grid Infrastructure (Oracle Restart)のダウングレード
  Oracle Restartをダウングレードするには、Oracle Grid Infrastructureを構成解除してから再インストールする必要があります。その後、データベースおよびサービスを戻すことができます。
- Oracle Messaging Gatewayを使用したデータベースのダウングレード・エラー
  Oracle Messaging Gatewayを使用して構成されたデータベースをダウングレードすると、`ORA-02303`エラーが発生することがあります。

親トピック: 以前のリリースへのOracle Databaseのダウングレード

catdwgrd.sqlスクリプトを使用したOracle Databaseコンポーネントのダウングレード中のエラー

この項を使用して、ダウングレード中に`catdwgrd.sql`スクリプトを実行するときに発生するエラー(`ORA-20001: ダウングレードを続行できませんなど`)のトラブルシューティングを行います。

`catdwgrd.sql`スクリプトはOracle Databaseのデータベース内のすべてのコンポーネントを、アップグレード前の当初のメジャー・リリースにダウングレードします。このスクリプトはデータ・ディクショナリをダウングレードする前に実行する必要があります。スクリプトの実行時に何か問題が発生した場合は、問題の原因を修正してからスクリプトを再実行します。

発生する可能性のあるエラーには、「`ORA-39709: コンポーネントのダウングレードが不完全です; stringのダウングレードは中断されました`」および「`ORA-06512`」があります。これらのエラーが発生した場合、ダウングレードを続行できません。

- 原因: データ・ディクショナリのダウングレード前にダウングレードする必要のある1つ以上のコンポーネントがダウングレードされていません。
- 処置: ログ・ファイルを確認して、`catdwgrd.sql`スクリプトが停止し、ダウングレードが中断する前に、どのようなエラーが発生したかを判断します。

これらの例を確認して、この問題の修正方法を理解します。

エラーには一般に、ダウングレードの完了を妨げている問題を修正するために何をする必要があるかが説明されています。エラー・メッセージの説明に従ってください。エラーの原因を修正した後に、catdwgrd.sqlスクリプトを再実行します。

たとえば、チェックによりCDB$ROOTのダウングレード中にCDBのダウングレードが失敗した場合は、エラー・メッセージの説明に従って、条件エラーを修正します。エラーを修正した後に、catcon.plを指定してcatdwgrd.sqlを再実行します。-cオプションを使用して、包含リスト'CDB$ROOT PDB1'でコマンドを実行します。-rオプションを使用して、コマンドを最初にPDBで、次にCDB$ROOTで実行します。たとえば:

```
$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl -d
$ORACLE_HOME/rdbms/admin -e -b catdwgrd -l /scratch/rac/downgradeLogs -c 'CDB$ROOT,
PDB1, PDB2' -r catdwgrd.sql
```

例8-3 ORA-06512によるORA-20001エラー

ダウングレードが中止されました。ログ・ファイルを確認したところ、catdwgrd.sqlがこのエラーで終了したことがわかりました。

```
DECLARE * ERROR at line 1: ORA-20001: Downgrade cannot proceed -
Unified Audit Trail data exists. Please clean up the data first
using DBMS_AUDIT_MGMT.CLEAN_AUDIT_TRAIL. ORA-06512: at line 65
ORA-06512: at line 42
```

CDB$ROOTおよびすべてのPDBで統合監査証跡を削除する必要があります。

たとえば:

1. 統合監査証跡の存在を確認します。

```
SQL> SELECT COUNT(*) FROM UNIFIED_AUDIT_TRAIL;
COUNT(*)
----------
      4538
```

2. CDB上で監査証跡を削除します。

   たとえば、この場合監査証跡タイプはDBMS_AUDIT.MGMT.AUDITになります。

```
EXEC DBMS_AUDIT_MGMT.CLEAN_AUDIT_TRAIL DBMS_AUDIT_MGMT.AUDIT
```

3. CDB$ROOTでcatdwngrd.sqlを実行します。PDBに依然として統合監査データが存在する場合、スクリプトはORA20001で失敗します。

```
 62        execute immediate
 63        'select count(*) from audsys.'||'"'||tab_name||'"' into no_rows;
 64
 65        -- If audit trail has some data, raise the application error
 66        IF no_rows > 0 THEN
 67          RAISE_APPLICATION_ERROR(-20001, ErrMsg);
 68        END IF;
 69      END IF;
 70    END IF;
 71  EXCEPTION
 72    WHEN NO_DATA_FOUND THEN
 73     NULL;
 74    WHEN OTHERS THEN
 75     RAISE;
 76  END;
 77  /
DECLARE
*
ERROR at line 1:
```

```
ORA-20001: Downgrade cannot proceed - Unified Audit Trail data exists.Please
clean up the data first using DBMS_AUDIT_MGMT.CLEAN_AUDIT_TRAIL.
ORA-06512: at line 75
```

4. 個々のPDBに接続し、統合監査証跡があるかどうかを確認します。すべてのPDBの統合監査証跡をクリアします。たとえば、PDB1という名前のPDBに統合監査証跡がある場合、次のようになります。

```
ALTER SESSION SET container = PDB1;
SQL> SELECT COUNT(*) FROM UNIFIED_AUDIT_TRAIL;
 COUNT(*)
----------
      1330
```

5. 統合監査証跡を特定します。

```
SQL> CREATE TABLE UA_DATA AS (SELECT * FROM UNIFIED_AUDIT_TRAIL);
```

6. 監査証跡を削除します。

   この例では、監査証跡タイプはDBMS_AUDIT_MGMT.AAUDIT_TRAIL_UNIFIEDであり、USE_LAST_ARCH_TIMESTAMP値はFALSEに設定されているため、最終アーカイブ・タイムスタンプを考慮せずにすべての監査レコードが削除され、CONTAINER値がDBMS_AUDIT_MGMT.CONTAINER_ALLに設定されているため、すべてのPDB上の監査レコードが削除されます。

```
BEGIN
  DBMS_AUDIT_MGMT.CLEAN_AUDIT_TRAIL(
   AUDIT_TRAIL_TYPE            =>  DBMS_AUDIT_MGMT.AUDIT_TRAIL_UNIFIED,
   USE_LAST_ARCH_TIMESTAMP     =>  FALSE,
   CONTAINER                   =>  DBMS_AUDIT_MGMT.CONTAINER_ALL
END;
/
```

7. PDBおよびCDBレベルでcatdwngrd.sqlを再実行します。たとえば:

```
$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl -c
'CDB$ROOT,PDB1' -d $ORACLE_HOME/rdbms/admin -e -b catdwgrd -l
/u01/oracle/product/19.0.0/downgrade_logs -r catdwgrd.sql
```

8. CDBおよびPDBでスクリプトが正常に完了してORA-20001エラーがログに表示されなくなるまで、監査証跡を検索して削除するプロセスを繰り返し、catdwgrd.sqlを実行します

関連項目

- 『Oracle Databaseセキュリティ・ガイド』

親トピック: Oracle Databaseのダウングレードのトラブルシューティング

アップグレードの成功または失敗後のOracle Grid Infrastructure (Oracle Restart)のダウングレード

Oracle Restartをダウングレードするには、Oracle Grid Infrastructureを構成解除してから再インストールする必要があります。その後、データベースおよびサービスを戻すことができます。

関連項目

- https://support.oracle.com/rs?type=doc&id=1364412.1
- Oracle Grid Infrastructureのインストレーション・ガイド

親トピック: Oracle Databaseのダウングレードのトラブルシューティング

Oracle Messaging Gatewayを使用したデータベースのダウングレード・エラー

Oracle Messaging Gatewayを使用して構成されたデータベースをダウングレードすると、ORA-02303エラーが発生することがあります。

Oracle Messaging Gatewayオブジェクトを含むOracle Databaseをダウングレードすると、次のエラーが発生することがあります。

```
ORA-02303: cannot drop or replace a type with type or table dependents
```

- 原因 catrelod.sqlスクリプトで、以前のOracle Databaseリリースとは異なるタイプのOracle Messaging Gatewayオブジェクトを再ロードしようとしています。
- 処置 処置は必要ありません。このエラーは無視できます。

親トピック: Oracle Databaseのダウングレードのトラブルシューティング

9 Oracle Data Pumpを使用したデータの移行

Oracle Data Pumpのエクスポートおよびインポート・ユーティリティを使用して、あるデータベースから別のデータベースにデータを移行します。

Oracle Data Pumpでは、高パフォーマンスのエクスポート(expdp)およびインポート(impdp)ユーティリティが提供されています。これらのユーティリティによってOracle Databaseのアップグレードが容易になります。

- データの移行でのデータ・ポンプおよびエクスポート/インポートの概要
  Oracleでは、あるOracle Databaseインスタンスから別のOracle Databaseインスタンスにデータを移行(移動)するためのOracle Data Pumpのエクスポートおよびインポートが提供されています。
- アップグレード前のOracle Data Pumpによるデータの移行
  このOracle Data Pumpの手順を使用して、新しいOracle Databaseソフトウェアをインストールする前にソース・データベースからデータをエクスポートします。次に、アップグレードしたターゲット・データベースにデータをインポートします。
- ネットワーク・リンクを使用したOracle Database全体のインポート
  このデータベースのエクスポート/インポート方法は、Oracle Data Pumpによる移行の代替方法で、異なるストレージ・システムに移行する場合に役立ちます。
- Oracle Databaseのダウングレード時のOracle Data Pumpの要件
  Oracle Data Pumpを使用して下位互換性のあるダンプ・ファイルを取得できます。

参照:

データ・ポンプおよびエクスポート/インポート・ユーティリティの詳細は、『Oracle Databaseユーティリティ』を参照してください。

データの移行でのデータ・ポンプおよびエクスポート/インポートの概要

Oracleでは、あるOracle Databaseインスタンスから別のOracle Databaseインスタンスにデータを移行(移動)するためのOracle Data Pumpのエクスポートおよびインポートが提供されています。

Oracle Data Pumpを使用してデータを移行すると、次のメリットがあります。

- オブジェクトおよびオブジェクトの種類に基づいてエクスポートおよびインポートされたメタデータを、INCLUDEおよびEXCLUDEパラメータを使用してフィルタ処理できます。
- データベースの一部を様々なモード(全体データベース・モード、スキーマ・モード、表モード、表領域モード、トランスポータブル表領域モードなど)でアンロード/ロードできます。
- インポート操作中のパーティション表の処理方法を、PARTITION_OPTIONSパラメータを使用して指定できます。
- 全種類のデータ型がサポートされます。

Oracle Database 18cでは、UTL_FIL_DIRはサポートされなくなりました。UTL_FILEパッケージを使用した機能によるシンボリック・リンクへのアクセスは、新しいOracle Databaseリリースでは失敗します。この問題を回避するには、シンボリック・リンクが含まれないようにディレクトリ・オブジェクトとファイル名を変更する必要があります。このサポート終了は、Oracle Data Pump、BFILEおよび外部表を含む(ただし、これに制限されない)、シンボリック・リンクを使用する以前のリリースの機能に影響する可能性があります。アップグレード後に影響を受ける機能を使用しようとして、その機能でシンボリック・リンクが使用された場合、ORA-29283: 無効なファイル操作です。 : パスがsymlinkをトラバースしていますが発生します。

関連項目

- Oracle Databaseのアップグレード時のデータの移行でエクスポート/インポートする方法

参照:

データ・ポンプ・エクスポートおよびインポートの概要は、『Oracle Databaseユーティリティ』を参照してください。

親トピック: Oracle Data Pumpを使用したデータの移行

アップグレード前のOracle Data Pumpによるデータの移行

このOracle Data Pumpの手順を使用して、新しいOracle Databaseソフトウェアをインストールする前にソース・データベースからデータをエクスポートします。次に、アップグレードしたターゲット・データベースにデータをインポートします。

1. 新しいOracle Databaseソフトウェアをインストールし、最新のリリース更新(RU)またはリリース更新リビジョン(RUR)に更新されていることを確認します。Oracle Databaseのインストール・ステップは、ご使用のオペレーティング・システム固有のOracleマニュアルを参照してください。
2. 現行データベースに付属のエクスポート・ユーティリティを使用して、現行データベースのデータをエクスポートします。

   エクスポートの実行中と実行後は、現行データベースは更新不可である必要があります。エクスポート実行後にユーザーが現行データベースを更新できるようにする場合は、現行データベースで行われた変更を、インポート完了後に新しいデータベースにコピーするためのプロシージャを設定し、その後で現行データベースを使用可能にする必要があります。

   エクスポートしたデータベースを更新できる一貫性維持点を取得するには、次のいずれかのオプションを使用します。

   - パラメータ・ファイルでFLASHBACK_TIME=SYSTIMESTAMPを設定して、エクスポートする表のデータから取得するイメージが、エクスポートするすべての表の同じ単一の時点における表のデータのコミット済状態を表すようにします。
   - FLASHBACK_SCNを使用して、エクスポートがフラッシュバック問合せユーティリティを有効化するために使用できる特定のシステム変更番号(SCN)を選択します。

   フラッシュバック・オプションを使用すると、UNDOの使用量および保存が増加します。
3. 新しいデータベースを作成します。新しいデータベースが同じサーバー上にあり、それに現行データベースと同じ名前を付ける場合は、新しいデータベースを作成する前に現行データベースを停止します。

   (オプション)ソース・データベースとは異なる記憶域パラメータを指定できます。新しいデータベースに表領域、ユーザーおよび表を事前に作成し、記憶域パラメータを変更することにより、ディスク領域の使用状況を改善できます。SQL*Plusを使用して表を事前に作成する場合は、元のデータベースとの互換モードで実行するか、またはインポート中に発生する特定のデータ定義の変換を許可するようにします。
4. SQL*Plusを新しいOracle Database環境で起動し、Oracle Databaseインスタンスを起動します。

   たとえば:

   ```
   $ SQLPLUS / AS SYSDBA
   SQL> STARTUP
   ```
5. 事前作成された項目がある場合、データ・ポンプ・インポートでTABLE_EXISTS_ACTION=APPENDオプションを指定します。

ノート:

新しいデータベースをソース・データベースと同じサーバー上に作成し、ソース・データベースのデータファイルを上書きしない場合は、事前に表領域を作成し、データ・ポンプ・インポートで次のパラメータ値を設定する必要があります。

- データ・ポンプ・インポートで REUSE_DATAFILES=N を指定

  (オプション) REMAP_DATAFILE、 REMAP_TABLESPACE および REMAP_TABLE オプションを使用して、ダンプ・ファイル・セットのエクスポート・データベース名に対する参照を、インポート・データベースの競合しない新しい名前に再マップできます。

- オリジナルのインポート・ユーティリティで DESTROY=N を指定

6. 新しいデータベースのインポート・ユーティリティを使用して、現在のデータベースからエクスポートされたオブジェクトをインポートします。情報メッセージやエラー・メッセージをインポート・セッションからファイルに保存するには、次のパラメータを使用します。
   - データ・ポンプ・インポートでLOGFILEパラメータを指定
   - オリジナルのインポート・ユーティリティでLOGパラメータを指定
7. インポート作業が完了したら、正常に終了した特定のオブジェクトのインポートに関する情報を、インポート・ログ・ファイルで確認します。失敗したものがある場合は、失敗したオブジェクトに関する情報を確認します。
8. その他のインポート方法を使用するか、データベース・オブジェクトを作成するSQLスクリプトを使用して不完全なインポートをクリーン・アップします(または完全に新規のインポートを開始します)。

ノート:

データ・ポンプ・エクスポートまたはインポートのジョブでリカバリ不能なエラーが発生した場合は、エラーの原因となった条件を修正した後でジョブを再起動できます。ジョブは、エラーが発生した箇所から自動的に続行されます。

9. エクスポートの実行後に現行データベースに変更を加える場合は、ユーザーに使用可能にする前に、それらの変更が新しいデータベースに伝播されるようにします。この手順のステップ1を参照してください。
10. 第4章「Oracle Databaseのアップグレード後の作業」の記載に従って、必要なアップグレード後の作業を完了します。

関連項目

- Oracle Databaseユーティリティ

親トピック: Oracle Data Pumpを使用したデータの移行

ネットワーク・リンクを使用したOracle Database全体のインポート

このデータベースのエクスポート/インポート方法は、Oracle Data Pumpによる移行の代替方法で、異なるストレージ・システ

ムに移行する場合に役立ちます。

データベース・リンクを設定し、データ・ポンプ・インポート・ユーティリティ(impdp)を使用してソース・データベースから宛先データベースへのデータベース全体のインポートを実行します。この方法を使用してデータを移行すると、ダンプ・ファイルには書き込まれないため、ダンプ・ファイルをコピーする必要はありません。この方法は、異なるストレージ・システムを使用している場合に特に役立ちます。ただし、Oracle Databaseのエクスポート時に以前のリリースのOracle Data Pumpソフトウェアによって適用された制限内で作業を行う必要があります。

ノート:

相互運用性のエラーを回避するため、アップグレードを開始する前に、アップグレードするデータベースに適切なパッチ・セットが適用されていることを確認します。

1. ソース・データベースでエクスポートを実行するユーザーにDATAPUMP_EXP_FULL_DATABASEロールがあることを確認します。

   データベース・リンクの作成時に、このエクスポートを実行するユーザーを指定する必要があります。

2. インポート先データベースでインポートを実行するユーザーにDATAPUMP_IMP_FULL_DATABASEロールがあることを確認します。

3. ソース・データベースとインポート先データベースとの間にデータベース・リンクを作成してテストします。

4. 次のコマンド構文を使用してデータ・ポンプ・エクスポートを開始します(import_userはインポートを実行するユーザーのユーザー名で、db_linkはエクスポートを実行するユーザーが所有するデータベース・リンクの名前です)。

```
impdp import_user NETWORK_LINK=db_link FULL=Y NOLOGFILE=Y;
```

   ノート:

   インポート・データベースでこのコマンドを実行すると、エクスポート Oracle Database でデータ・ポンプ・エクスポート操作(expdp)が暗黙的にトリガーされます。

5. インポート操作のログ・ファイルが、DATA_PUMP_DIRディレクトリに書き込まれます。このディレクトリの場所は、次のコマンドを実行して検出できます。

```
SQL> select * from dba_directories where DIRECTORY_NAME like 'DATA_PUMP_DIR';
```

   このコマンドを実行する場合、XDBリポジトリは、データベース全体のエクスポートおよびインポート操作では移動されないことに注意してください。LONG列およびNESTED列も移動されません。ただし、ユーザーが作成したXMLスキーマは移動されます。

ノート:

インポート操作によって、新しい宛先サーバーでユーザーが再作成されます。dba_users 表のユーザーの作成日には、実際のインポート日が示されます。有効期限は、creation_date に password_life_time 列を加えた値に更新されます。宛先データベースの dba_users 表の作成日は、ソース・データベースの dba_users

表の値とは異なります。

関連項目

- アップグレード前のOracle Data Pumpによるデータの移行

参照:

最新のパッチ・セットを取得するには、http://support.oracle.comを参照してください

親トピック: Oracle Data Pumpを使用したデータの移行

Oracle Databaseのダウングレード時のOracle Data Pumpの要件

Oracle Data Pumpを使用して下位互換性のあるダンプ・ファイルを取得できます。

ダウングレード処理でOracle Data Pumpを使用する場合、ダウングレード先として可能なOracle Databaseリリースは、最大でもダウングレード元のリリースより1つ前のリリースです。

次のOracle Data Pumpエクスポートの方法のいずれかを使用して、下位互換性のあるダンプ・ファイルを取得します。

- 現行のリリースのOracle Databaseホームに含まれるOracle Data Pumpエクスポート・ユーティリティを使用し、VERSIONパラメータをダウングレードする以前のターゲットのリリースに設定します。

  Oracle Data Pumpインポートは、現行のOracle Databaseリリースより後のOracle Data Pumpのバージョンで作成されたダンプ・ファイル・セットを読み取ることができません(ただし、それらのダンプ・ファイル・セットが、ターゲット・データベースのリリース番号に設定されたVERSIONパラメータを使用して作成されている場合を除く)。そのため、ダウングレードを行う際の最適な方法は、VERSIONパラメータをダウングレードするターゲット・データベースのリリース番号に設定して、Oracle Data Pumpエクスポートを使用することです。

- `NETWORK_LINK`パラメータと`VERSION`パラメータを使用して、`DOWNGRADE`でOracle Data Pumpエクスポート・ユーティリティを使用します。

  ダウングレード対象のデータベースの互換性レベルが、作成したエクスポート・ダンプ・ファイル・セットのバージョンよりも古い場合でも、エクスポート元のOracle Databaseの互換性レベルが2つのメジャー・リリース番号の間であればデータベース・リンクを介してデータを転送できます。たとえば、一方のデータベースがOracle Database 18cの場合、もう一方のデータベースは12cリリース1 (12.1)または12cリリース2 (12.2)である必要があります。データ・ポンプ・エクスポートをこのように使用することで、以前のダンプ・ファイル・エクスポート中に`VERSION`が互換性のない値に設定されていても回復できます。

ノート:

新しいリリースの Oracle Database をインストールした後で、その新しいリリースの機能が使用できるように互換性レベルを上げた場合、それらの新機能を使用するオブジェクトはダウングレードできません。ダウングレード後にそれらのオブジェクトをインポートしようとして、Oracle Data Pump インポート・ユーティリティが以降のリリースでのみ使用可能な機能を使用するオブジェクトの再作成を試行した場合は、エラーが発生します。

参照:

VERSIONパラメータの使用方法の詳細は、Oracle Databaseユーティリティを参照してください

親トピック: Oracle Data Pumpを使用したデータの移行

10 Oracle Databaseの変更、サポート終了および非推奨

データベースのパッチ適用およびアップグレードに関する長期的な計画の作成を支援するために、Oracleでは、複数のリリースにわたる変更、非推奨およびサポート終了のリストが提供されます。

このドキュメントは、Oracleがリリースに関する情報を反映して定期的に更新されるため、後日移行する予定のリリースの変更を計画できます。たとえば、Oracle Database 19cにアップグレードする場合、長期計画では、以降のリリースでサポートが終了した機能や、以降のリリースで非推奨の通知がある機能を認識しておくと便利です。また、セキュリティ・プロトコルまたはデフォルトのパラメータ設定が以降のリリースで変更されていることを認識しておくと役立つことがあります。これにより、これらの非推奨またはサポート終了が有効になっているリリースにアップグレードする前に、現在のリリースでその変更の準備を開始できます。

- 非推奨およびサポート対象外のステータスについて
  Oracle Databaseリリースでは、新機能に加えて、機能を変更、非推奨またはサポート終了して、データベースのアップグレード動作の変更を導入できます
- サポート終了および非推奨のリリース通知日
  サポート終了または非推奨の通知が公開された日付は、トピックの日付スタンプを使用して識別できます。
- ORADiffツールを使用したリリース変更の検索
  パラメータのデフォルト、ユーザー、ロール、およびデータベース・リリース間のその他の変更内容を検索するために、OracleではOracle Databaseリリース差分ユーティリティ(ORADiff)を使用することをお薦めします。
- Oracle Database 23cの動作の変更、サポート終了および非推奨
  Oracle Database 23cリリースの変更の詳細を確認してください。
- Oracle Database 21cの動作の変更、サポート終了および非推奨
  Oracle Database 21cリリースの変更の詳細を確認してください。
- Oracle Database 19cの動作の変更、サポート終了および非推奨
  Oracle Database 19cリリースの変更の詳細を確認してください。
- Oracle Database 18cの動作の変更、サポート終了および非推奨
  Oracle Database 18cリリースの変更の詳細を確認してください。
- Oracle Database 12cリリース2 (12.2)の動作の変更、サポート終了および非推奨
  Oracle Database 12cリリース2 (12.2)の変更の詳細を確認してください。
- Oracle Database 12cリリース1 (12.1)の動作の変更、サポート終了および非推奨
  Oracle Database 12cリリース1 (12.1)の変更の詳細を確認してください。

非推奨およびサポート対象外のステータスについて

Oracle Databaseリリースでは、新機能に加えて、機能を変更、非推奨またはサポート終了して、データベースのアップグレード動作の変更を導入できます

非推奨およびサポート終了の意味は、次のとおりです。

- **非推奨**機能とは、拡張はされませんが、非推奨通知が公開されるOracle Databaseリリースの存続期間中は引き続きサポートされる機能です。
- **サポート終了**機能とは、サポート終了通知が公開されているOracle Databaseリリースのその機能に関連するバグの修正によってサポートされなくなった機能です。多くの場合、機能を使用するために必要なコードを削除することがオラクル社では選択できます。非推奨となった機能は、次のOracle Databaseリリースではサポート終了となる可能性があります。

親トピック: Oracle Databaseの変更、サポート終了および非推奨

サポート終了および非推奨のリリース通知日

トピックの日付スタンプを使用して、サポート終了または非推奨通知が公開された日付を識別できます。

2023年9月

この文の上のテキストは日付スタンプです。このトピックは、アップグレード・ガイドのために2023年9月の更新で追加されたことを示しています。2023年8月以降、Oracleはこのタイムスタンプをサポート終了および非推奨のトピックに追加しています。

親トピック: Oracle Databaseの変更、サポート終了および非推奨

ORADiffツールを使用したリリース変更の検索

パラメータのデフォルト、ユーザー、ロール、およびデータベース・リリース間のその他の変更内容を検索するために、OracleではOracle Databaseリリース差分ユーティリティ(ORADiff)を使用することをお薦めします。

2023年8月

Oracle Databaseリリース差分ユーティリティ(ORADiff)は、2つのOracle Databaseリリース、またはリリース更新を比較できるOracleツールです。パラメータ、権限、ユーザーおよびロール、予約語、パッチ修正、データベース・ホーム、オブジェクトなどの差異がレポートされます。特定のリリースのこれらのアイテムのインベントリを取得するためにも使用できます。ORAdiffデータは、四半期ごとに新しいリリース更新がリリースされたときにリフレッシュされます。また、ソース・データベースのリリースから削除または変更されたアイテムをターゲット・データベースのリリースに識別するために使用できるレポートを生成することもできます。ぜひ参照してください。

ORADiffを使用するには、Oracleアカウントのシングル・サインオン(SSO)ユーザーでログインする必要があります。

関連項目

- ORADiff
- ORAdiffがライブ - 2つのOracle Databaseリリースとパッチ・リリースを比較します。
- ORAdiff - 2つのOracle Databaseリリースの違いを見つけます

親トピック: Oracle Databaseの変更、サポート終了および非推奨

Oracle Database 23cの動作の変更、サポート終了および非推奨

Oracle Database 23cリリースの変更の説明を確認してください。

- Oracle Database 23cアップグレード計画の動作の変更
  これらの選択した動作変更を確認して、Oracle Database 23cへのアップグレードの計画に役立ててください
- Oracle Database 23cでサポートが終了した機能
  アップグレード計画の一環として、このOracle Databaseリリースでサポートが終了した機能を確認してください。
- Oracle Database 23cでサポートが終了したパラメータ
  アップグレード計画の一環として、このOracle Databaseリリース以降、サポートされなくなった初期化パラメータを確認してください。
- Oracle Database 23cで非推奨となった機能
  アップグレード計画の一環として、このOracle Databaseリリースで非推奨となった機能を確認し、アプリケーション戦略に対する代替方法を確認してください。

- Oracle Database 23cで非推奨になったビュー
  アップグレード計画の一環として、このOracle Databaseリリースから非推奨になったビューを確認してください。
- Oracle Database 23cで非推奨になったパラメータ
  アップグレード計画の一環として、ここにリストされている、このOracle Databaseリリース以降で非推奨になる初期化パラメータを確認してください。

親トピック: Oracle Databaseの変更、サポート終了および非推奨

Oracle Database 23cアップグレード計画の動作の変更

Oracle Database 23cへのアップグレードの計画に役立つこれらの優れた動作変更を確認してください

動作の変更には、パラメータまたは機能のデフォルトの変更、製品名の変更、および注意が必要なデータベース構成に対するその他の変更が含まれる場合があります。

- Oracle Spatialおよび非推奨オブジェクト
  以前にメディア・オブジェクトに対してOracle MultimediaまたはOracle Locatorを使用した場合は、Oracle Database 23cへのアップグレード時にOracle Spatialがインストールされて、ORDIMおよびLCTRコンポーネントがレジストリから削除されます。
- AutoUpgradeのREST API
  Oracle DatabaseのアップグレードにAutoUpgradeユーティリティを安全かつ確実にリモート使用するために、AutoUpgradeでREST API (ORDSおよびOCI)が提供されるようになりました。
- XML JSON検索索引の拡張機能
  Oracle Database 23c以降では、JSON検索の拡張機能がアップグレード計画の要因となる場合があります。
- SQL/JSONファンクションJSON_VALUEとブール型のJSON値
  Oracle Database 23c以降では、データ型BOOLEANがOracle SQLに追加されます。この機能はSQL/JSON演算子を拡張して、BOOLEAN値を返すか、入力として受け入れることを可能にします。
- アップグレード前のFIPSの非AESアルゴリズムからの移行
  Transparent Data Encryption (TDE)を使用する場合、Oracle Database 23cへのアップグレードを開始する前に、ソース・データベースをAES暗号化に移行する必要があります。
- 読取り専用Oracle Homesについて
  Oracle Database 23cでは、Oracleホームはデフォルトで読取り/書込みモードで使用可能です。ただし、ソフトウェアのみのOracle Databaseインストールを実行した後で、読取り専用モードでOracleホームを構成することを選択できます。
- SYSDATEおよびSYSTIMESTAMPがPDBタイム・ゾーンを反映
  Oracle Database 23c以降では、SYSDATEおよびSYSTIMESTAMPを、個々のデータベースのタイム・ゾーン設定に従ってデータベースごとに個別に管理できます。
- 4バンド・ラスター・ブロックでのOracle Spatial GeoRaster JPEG圧縮
  Oracle Database 23c以降では、Oracle Spatial GeoRasterオブジェクトのJPEG圧縮は、1バンドおよび3バンド・ラスター・ブロックにのみ適用できます。

親トピック: Oracle Database23cの動作の変更、サポート終了および非推奨

Oracle Spatialおよび非推奨オブジェクト

以前にメディア・オブジェクトに対してOracle MultimediaまたはOracle Locatorを使用した場合は、Oracle Database 23cへのアップグレード時にOracle Spatialがインストールされて、ORDIMおよびLCTRコンポーネントがレジストリから削除されます。

- Oracle Database 21cへのアップグレードでは、MDSYSがデータベースに存在したが、Oracle Spatial (SDO)は存在しなかった場合に、Oracle Locator (LCTR)がデータベース・レジストリにインストールされました。
- Oracle Database 23cへのアップグレード時には、Oracle Multimedia (ORDIM)が削除されます。LCTRがデータベース・レジストリに存在する場合は、それも削除されます。MDSYSスキーマのLocatorオブジェクトがデータベースに存在するがSDOは存在しない場合は、SDOがインストールされます。
- Oracle Database 23cからダウングレードした場合、ダウングレードの一環としてORDIMはリストアされません。
- Oracle Database 23cからOracle Database 21cにダウングレードする場合は、LCTRが再インストールされます。
- データベース・レジストリ内にORDIMやLCTRがある下位バージョンの非CDBやPDBを、SDOがインストールされていないターゲットの23c CDBにプラグインする場合は、プラグインの前にOracle Database 23cのCDB$ROOTにSDOをインストールします。

関連項目

- Oracle Spatialのインストール、移行、互換性およびアップグレード
- 『Oracle Spatial and Graph開発者ガイド』のこのリリースでの変更点
- 注意: ORDIMが存在するがSDOがない場合、21cへのアップグレードが失敗します
- My Oracle SupportドキュメントID 2555923.1

親トピック: Oracle Database 23cアップグレード計画の動作の変更

AutoUpgradeのREST API

Oracle DatabaseのアップグレードにAutoUpgradeユーティリティを安全かつ確実にリモート使用するために、AutoUpgradeでREST API (ORDSおよびOCI)が提供されるようになりました。

Oracle REST Data Services (ORDS)データベースAPIは、Oracle REST Data Servicesに埋め込まれているデータベース管理およびモニタリングREST APIです。Oracle Cloud Infrastructure (OCI) REST APIは、OCI署名バージョン1のセキュリティ・ポリシーを使用するようにRESTアダプタ接続を構成することで有効化されます。これらの機能を使用し、SSHを介してAutoUpgradeアップグレードをリモートで実行できます。AutoUpgrade 22以降のバージョンでは、AutoUpgrade REST APIの使用がサポートされます。

関連項目

- https://docs.oracle.com/en/database/oracle/oracle-database/19/dbrst/

親トピック: Oracle Database 23cアップグレード計画の動作の変更

XML JSON検索索引の拡張機能

Oracle Database 23c以降では、JSON検索の拡張機能がアップグレード計画の要因となる場合があります。

アップグレード・テストが完了した後、Oracle Database 23cで導入されたXML検索索引の拡張機能を利用するために、COMPATIBLEパラメータを23.0.0.0に更新してからOracle Text索引およびJSON検索索引の再作成を計画することが必要になることがあります。ただし、既存のXQuery全文索引の機能は変更されません。この索引は、COMPATIBLE設定の更新後も引き続き使用できます。

COMPATIBLEパラメータ設定を更新した後は、既存のJSON検索索引が使用可能なすべてのパスを含むものとして処理されます。再作成しない既存のJSON検索索引は引き続き使用できます。パス対応JSON検索索引がOracle Database 21cで導入されたため、パス対応を有効にするためのJSON検索索引へのアップグレード変更は、Oracle Database 21cへのアップグレードの一部としてすでに処理されています。Oracle Database 23cで導入されたXML検索索引およびINCLUDEパス句の機能では、データベース互換性の設定を23.0.0.0.0に設定することが必須になります。

以前のリリースで使用されていた索引付け構文は引き続き使用できます。ただし、既存の索引付け機能用にOracle Database 23cで導入された構文は、互換性の設定を変更しなくても使用可能になります。ただし、JSON検索索引の場合、VALUEモードおよびパスのサブセット化機能は、JSON型列に索引を作成した後にのみ使用できます。

COMPATIBLEを23.0.0.0.0にアップグレードすると、次のシナリオでのデータベースのダウングレードが防止されます。

- XML検索索引が存在する場合
- パスのサブセット化を使用したXMLまたはJSON検索索引が存在する場合。

親トピック: Oracle Database 23cアップグレード計画の動作の変更

SQL/JSONファンクションJSON_VALUEとブール型のJSON値

Oracle Database 23c以降では、Oracle SQLにデータ型BOOLEANが追加されます。この機能はSQL/JSON演算子を拡張して、BOOLEAN値を返すか、入力として受け入れることを可能にします。

COMPATIBLEパラメータを23.0.0.0に更新した後に、SQL/JSONパス式項目メソッドboolean()およびbooleanOnly()がSQL BOOLEAN値を返すようになりました。つまり、json_valueのセマンティクスを持つ問合せでは、項目メソッドによって生成される値はデフォルトでBOOLEAN型になります。この値は、BOOLEAN型のjson_value RETURNING句によって制御されたかのように処理されます。

以前のリリースでは、これらのメソッドはVARCHAR2(20)値'true'または'false'を返しました。VARCHAR2値を(互換性のためなどで)取得する必要がある場合は、SQL関数to_charで値をラップできます。

関連項目

- Oracle Database JSON開発者ガイドSQL/JSONファンクションJSON_VALUEとブール型のJSON値の使用方法

親トピック: Oracle Database 23cアップグレード計画の動作の変更

アップグレード前のFIPS内の非AESアルゴリズムからの移行

Transparent Data Encryption (TDE)を使用する場合は、Oracle Database 23cへのアップグレードを開始する前に、ソース・データベースをAES暗号化に移行する必要があります。

Oracle Database 23cでは、Federal Information Processing Standard (FIPS)用に構成されたTransparent Data Encryption (TDE)を使用する場合、Advanced Encryption Standard (AES)暗号AES-128、AES-192およびAES-256のみが許可されます。ソースOracle DatabaseがFIPSモード用に構成されており、他のアルゴリズムを使用して列または表領域を暗号化している場合、アップグレード前にAESを使用して列および表領域のキーを再作成する必要があります。

ソース・データベースをアップグレードし、暗号化にサポートされていないアルゴリズムを使用している場合、アップグレードされたデータベースは起動に失敗するか、データベースが表領域キーを復号化できないため、暗号化された表領域を使用できません。その場合、Oracleでは、データベースをダウングレードし、暗号化キーをサポートされているAES暗号にアップグレードしてから、アップグレードを再開することをお薦めします。

親トピック: Oracle Database 23cアップグレード計画の動作の変更

読取り専用Oracleホームについて

Oracle Database 23cでは、Oracleホームはデフォルトで読取り/書込みモードで使用可能です。ただし、ソフトウェアのみのOracle Databaseインストールを実行した後で、読取り専用モードでOracleホームを構成することを選択できます。

読取り専用のOracleホームになることで、インストールと構成が別々に実装され、それによりプロビジョニングが簡素化されます。

Oracleホームが読取り専用の場合、すべての構成データおよびログ・ファイルがその読取り専用Oracleホーム以外の場所に格納されます。

従来のORACLE_BASEおよびORACLE_HOMEディレクトリとは別に、ORACLE_HOMEに格納されていたファイルは次のディレクトリに格納されます。

- ORACLE_BASE_HOME
- ORACLE_BASE_CONFIG

ノート:

この機能は、データベース管理者によるシステム・パフォーマンスの監視、診断および調整の方法に影響を及ぼすものではありません。

親トピック: Oracle Database 23cアップグレード計画の動作の変更

SYSDATEおよびSYSTIMESTAMPがPDBタイム・ゾーンを反映

Oracle Database 23c以降では、SYSDATEおよびSYSTIMESTAMPを、個々のデータベースのタイム・ゾーン設定に従ってデータベースごとに個別に管理できます。

マルチテナントの適用性と透過性を高め、独立したデータベースをさらに統合するために、プラガブル・データベース(PDB)をコンテナ・データベース(CDB)のデータベースに対して個別に管理できるようになりました。ユーザーに表示されるすべての操作および内部機能(Oracle SchedulerやOracle Flashbackテクノロジなど)がこの設定に従います。

以前のリリースでは、データベース・ホストのオペレーティング・システム・レベルに従って、SYSDATEおよびSYSTIMESTAMP設定が一元管理されていました。この更新により、Oracle Multitenantを使用してOracle Databaseで複数のプラガブル・データベースを自己完結型データベースとして統合できるため、リソース使用率とデータベース管理が改善されます。アップグレード後に初期化パラメータを変更しない場合、デフォルトのままになり、すべてのプラガブル・データベースはオペレーティング・システムからシステム時間を継承します。

関連項目

- SYSDATE
- SYSTIMESTAMP

親トピック: Oracle Database 23cアップグレード計画の動作の変更

4バンド・ラスター・ブロックでのOracle Spatial GeoRaster JPEG圧縮

Oracle Database 23c以降、Oracle Spatial GeoRasterオブジェクトのJPEG圧縮は、1バンドおよび3バンド・ラスター・ブロックにのみ適用できます。

4バンド・ラスター・ブロックを持つGeoRasterオブジェクトがある場合は、JPEG圧縮を適用する前に、オブジェクトを1バンドまたは3バンド・ラスター・ブロックに再ロックする必要があります。以前のリリースの4バンド・ラスター・ブロックでJPEG圧縮を使用するGeoRasterオブジェクトがある場合は、Oracle Database 23cにアップグレードする前に、それらを1バンドまたは3バンド・ラスター・ブロックに再ロックする必要があります。

関連項目

- My Oracle SupportドキュメントID 2960620.1

Oracle Database 23cでサポートが終了した機能

アップグレード計画の一環として、このOracle Databaseリリースでサポートが終了した機能を確認してください。

- ODP.NET OracleConfiguration.DirectoryTypeプロパティおよび.NET構成ファイルのDIRECTORY_TYPE設定のサポート終了
  `OracleConfiguration`クラスの一部であるOracle Data Provider for .NET `DirectoryType`プロパティは、Oracle Database 23cでサポートが終了しました。
- 元のエクスポート・ユーティリティ(EXP)のサポート終了
  元のOracle Databaseのエクスポート(exp)ユーティリティは、Oracle Database 23cでサポートが終了しました。
- Oracle用MySQLクライアント・ライブラリ・ドライバのサポート終了
  Oracle用MySQLクライアント・ライブラリ・ドライバは、Oracle Database 23cでサポートが終了しました。
- ACFSUTIL REPL REVERSEのサポート終了
  `acfsutil repl reverse`コマンドは、Oracle Database 23cでサポートが終了しました。かわりに`repl failover`または`repl switchover`を使用します。
- クラスタ・ドメイン - ドメイン・サービス・クラスタのサポート終了
  Oracle Grid Infrastructure 23c以降、Oracleクラスタ・ドメイン・アーキテクチャに含まれるドメイン・サービス・クラスタ(DSC)はサポートされなくなりました。
- 適応しきい値機能DBSNMPのDBSNMPパッケージのサポート終了
  適応しきい値機能に関連付けられたDBSNMP PL/SQLパッケージは、Oracle Database 23cでサポートが終了しました。
- ポリシー管理型のデータベース・デプロイメントのサポート終了
  ポリシー管理型のデータベース・デプロイメント・オプションは、Oracle Database 23cでサポートが終了しました。
- エンタープライズ・ユーザー・セキュリティ・ユーザー移行ユーティリティのサポート終了
  Database 23c以降、エンタープライズ・ユーザー・セキュリティ(EUS)ユーザー移行ユーティリティ(UMU)はサポートされなくなりました。
- Oracle Enterprise Manager Database Expressのサポート終了
  Oracle Enterprise Manager Database Express (EM Express)は、Oracle Databaseリリース23cでサポートが終了しました。
- Oracle Wallet Manager (OWM)のサポート終了
  Oracle Database 23c以降、Oracle Wallet Manager (OWM)はサポートされなくなりました。
- RASADMのサポート終了
  Real Application Security GUI管理ツール(RASADM)は、Oracle Database 23cでサポートが終了しました。
- Oracle Label Securityのパラメータおよびファンクションのサポート終了
  Oracle Database 12cで非推奨となったOracle Label Security (OLS)のパラメータおよびファンクションは、Oracle Database 23cでサポートが終了しました。
- Oracle Internet DirectoryとOracle Label Securityのサポート終了
  Oracle Internet Directory (OID)とOracle Label Security (OLS)の統合は、Oracle Database 23cでサポートが終了しました。
- RADIUSユーザーへの管理権限の付与のサポート終了
  Oracle Database 23c以降、レガシーRADIUS APIを使用してデータベースへの認証を行うユーザーには、管理権限が付与されなくなりました。
- Transparent Data Encryption PKIキーのサポート終了

Transparent Data Encryption (TDE)公開キー・インフラストラクチャ(PKI)キーはOracle Database 23cでサポートが終了しました

- GOSTおよびSEED TDE暗号化アルゴリズムのサポート終了
  Oracle Database 23c以降、GOSTおよびSEEDアルゴリズムのTransparent Data Encryption (TDE)暗号化ライブラリはサポートされなくなり、削除されました。GOSTおよびSEED復号化ライブラリは非推奨です。どちらも、HP Itaniumプラットフォームでは削除されます。
- Oracle Database 10Gパスワード検証機能のサポート終了
  Oracle Database 23c以降では、10Gデータベース・パスワード検証機能はサポートされなくなりました。
- Transport Layer Securityバージョン1.0および1.1のサポート終了
  Database 23c以降では、Transport Layer Securityプロトコル・バージョン1.0および1.1の使用はサポートされなくなりました。
- Unix Crypt (MD5crypt)パスワード検証機能のサポート終了
  Unix Crypt (`MD5crypt`)パスワード検証機能アルゴリズムは、Oracle Database 23cのサーバーおよびクライアントではサポートされなくなりました。
- FIPS強度80の暗号化のサポート終了
  FIPS暗号強度80は、Oracle Database 23cのリリースではサポートされなくなりました。かわりに、FIPS暗号強度112を使用してください。
- Diffie-Hellman匿名暗号のサポート終了
  Oracle Database 23cでは、アウトバウンド接続とデータベース・クライアント/サーバー接続の両方について、Diffie-Hellman匿名暗号(DH anon)の使用はサポートされなくなりました。
- Oracle Database Extensions for .NETのサポート終了
  Oracle Database Extensions for .NETはサポートされなくなりました。.NETコードを中間層に配置するか、外部プロシージャ機能を使用するか、PL/SQLまたはJavaを使用してコードをリライトすることをお薦めします。
- Quality of Service Managementのサポート終了
  Oracle Databaseリリース23c以降では、Oracle Quality of Service Management (QoSMまたはQoS Management)はサポートされなくなりました。
- 従来の監査のサポート終了
  Oracle Database 23cでは、従来の監査はサポートされなくなりました。統合監査を使用することをお薦めします。
- config.shのサポート終了
  Oracle Database 23c以降では、Oracle Grid Infrastructure構成ウィザード・ツール(`config.sh`)のサポートが終了しました。かわりに`gridSetup.sh`を使用してください。
- OLS表のLABELS列のサポート終了
  ALL_SA_USER_LABELS表およびDBA_SA_USER_LABELS OLS表のLABELS列は、Oracle Database 23cでサポートされなくなりました。
- 32ビットのOracle Databaseクライアントのサポート終了
  32ビットのOracle Databaseクライアントは、Oracle Database 23cでサポートされなくなりました。
- Oracle Sharding高可用性のためのOracle GoldenGateレプリケーションのサポート終了
  Oracle Shardingのシャード・レベルの高可用性のためのOracle GoldenGateレプリケーションの使用が、Oracle Database 23cでサポートされなくなりました。
- Grid Infrastructure管理リポジトリ(GIMR)のサポート終了
  Oracle Database 23c以降、Grid Infrastructure管理リポジトリ(GIMR)の使用はサポートされなくなりました。
- OPatchおよびOPatchAutoアウトオブプレース・パッチ適用のサポート終了
  アウトオブプレース・パッチ適用でのOpatchおよびOPatchAutoの使用は、Oracle Database 23cでサポートされ

なくなりました。

- データ・リカバリ・アドバイザ(DRA)のサポート終了
  Oracle Database 23c以降、データ・リカバリ・アドバイザ(DRA)機能はサポートされなくなりました。
- DBUAおよび手動アップグレード方法のサポート終了
  Oracleでは、データベースのアップグレードにAutoUpgradeを使用することをお薦めします。
- Oracle OLAPのサポート終了
  分析ワークスペース、OLAP DMLプログラミング言語およびOLAP Java APIは、Oracle Database 23cではサポートされなくなりました。

親トピック: Oracle Database23cの動作の変更、サポート終了および非推奨

ODP.NET OracleConfiguration.DirectoryTypeプロパティおよび.NET構成ファイルのDIRECTORY_TYPE設定のサポート終了

`OracleConfiguration`クラスの一部であるOracle Data Provider for .NET `DirectoryType`プロパティは、Oracle Database 23cでサポートが終了しました。

日付: 2023年4月

.NET構成ファイルの`DIRECTORY_SERVER_TYPE`設定によって、`DIRECTORY_TYPE`設定が置換されます。`OracleConfiguration DirectoryServerType`プロパティは、`DirectoryType`プロパティを置き換えます。これらのプロパティはすべて同一の機能を持ちます。開発者は、`DirectoryServerType`プロパティを使用して移行することをお薦めします。`DirectoryServerType`の名前は、同等の機能を提供する`DIRECTORY_SERVER_TYPE`よりも、`ldap.ora`パラメータに合わせて調整されています。

親トピック: Oracle Database 23cでサポートが終了した機能

元のエクスポート・ユーティリティ(EXP)のサポート終了

元のOracle Databaseのエクスポート(`exp`)ユーティリティは、Oracle Database 23cでサポートが終了しました。

日付: 2023年4月

Oracle Data Pumpエクスポート(`expdp`)を使用することをお薦めします。

親トピック: Oracle Database 23cでサポートが終了した機能

Oracle用MySQLクライアント・ライブラリ・ドライバのサポート終了

Oracle用MySQLクライアント・ライブラリ・ドライバはOracle Database 23cでサポートが終了しました。

日付: 2023年4月

MySQLクライアント・ライブラリ・ドライバ`liboramysql`は、Oracle Database 21cで非推奨になりました。現在は、サポートが終了しています。これに替わる機能はありません。このサポート終了は、`liboramysql`を使用してデータベースに接続する以前のOracle Databaseクライアント・リリースの機能に影響を与えることはありません。ただし、これらのクライアントで使用できる機能が最終的に制限される可能性があります。

親トピック: Oracle Database 23cでサポートが終了した機能

ACFSUTIL REPL REVERSEのサポート終了

`acfsutil repl reverse`コマンドは、Oracle Database 23cでサポートが終了しました。かわりに`repl failover`または`repl switchover`を使用します。

日付: 2023年4月

Oracle Automatic Cluster File System (ACFS)コマンド・ユーティリティ`acfsutil`には、コマンド`repl failover`および`repl switchover`が含まれます。これらのコマンドは、`acfsutil repl reverse`のすべての機能を含め、さらに多くの機能を提供します。このため、`acfsutil repl reverse`コマンドはサポートされなくなりました。

親トピック: Oracle Database 23cでサポートが終了した機能

クラスタ・ドメイン - ドメイン・サービス・クラスタのサポート終了

Oracle Grid Infrastructure 23c以降、Oracleクラスタ・ドメイン・アーキテクチャに含まれるドメイン・サービス・クラスタ(DSC)はサポートされなくなりました。

日付: 2023年4月

Oracleクラスタ・ドメインは、ドメイン・サービス・クラスタ(DSC)とメンバー・クラスタで構成されます。メンバー・クラスタは、Oracle Grid Infrastructure 19cで非推奨になりました。DSCは、引き続き本番クラスタにサービスを提供できます。ただし、それらのサービスの大部分ではホスティングにDSCを必要としなくなったため、DSCのインストールはOracle Database 23cでサポートが終了しました。該当する場合、以前にDSCでホストされていたサービスに、任意のクラスタまたはシステムを使用することをお薦めします。Oracleでは、代替システムで各サービスを使用できるようになるまで、共有サービスをホストするためのDSCのサポートを継続します。

親トピック: Oracle Database 23cでサポートが終了した機能

適応しきい値機能DBSNMPのDBSNMPパッケージのサポート終了

適応しきい値機能に関連付けられたDBSNMP PL/SQLパッケージは、Oracle Database 23cでサポートが終了しました。

日付: 2023年4月

Oracle Enterprise Manager Cloud Control 13.5以降、すべてのOracle Databaseターゲットのデータベース・サーバー適応しきい値およびベースライン・メトリックしきい値のすべての機能が削除されています。この機能をサポートするデータベース・サーバー側パッケージDBSNMP.BSLN、DBSNMP.BSLN_INTERNALは使用できなくなります。DBSNMP.MGMT_RESPONSEは、引き続きサポートされます。

この非推奨およびサポート終了の詳細は、My Oracle Support 2697846.1を参照してください

親トピック: Oracle Database 23cでサポートが終了した機能

ポリシー管理型のデータベース・デプロイメントのサポート終了

ポリシー管理型のデータベース・デプロイメント・オプションは、Oracle Database 23cでサポートが終了しました。

日付: 2023年4月

ポリシー管理データベースはOracle Database 21cで非推奨となり、管理者管理データベース・デプロイメントはポリシー管理データベースと同様の機能で拡張されました。ポリシー管理データベースによって提供される自動化と管理者管理データベースの一貫性を組み合せることで、データベース管理者のデータベース管理タスクの簡素化を目指しました。この組合せ型のデータベース・デプロイメントでは、デプロイメント時に特定のスタイルを選択する必要なく、データベースの起動順序をランク付けおよび定義するオプションを提供するなど、両方のメリットが提供されます。

親トピック: Oracle Database 23cでサポートが終了した機能

エンタープライズ・ユーザー・セキュリティ・ユーザー移行ユーティリティのサポート終了

Database 23c以降、エンタープライズ・ユーザー・セキュリティ(EUS)ユーザー移行ユーティリティ(UMU)はサポートされなくなりました。

日付: 2023年4月

回避策はありません。

親トピック: Oracle Database 23cでサポートが終了した機能

Oracle Enterprise Manager Database Expressのサポート終了

Oracle Enterprise Manager Database Express (EM Express)は、Oracle Databaseリリース23cでサポートが終了しました。

日付: 2023年4月

EM Expressは、Oracle Database内に構築されるWebベースのデータベース管理ツールです。これは、キー・パフォーマンス管理および基本的なデータベース管理機能をサポートしています。EM ExpressはOracle Database 21cで非推奨になりました。EM Expressの機能の多くは、Oracle Cloud Infrastructure (OCI) Database Managementサービス、Oracle Enterprise Manager Cloud ControlまたはOracle SQL Developerで使用できるようになりました。

EM Expressのかわりに、OCI Database Managementサービス、Oracle Enterprise Manager Cloud Control、Oracle SQL Developer WebまたはOracle SQL Developerデスクトップ製品から、要件およびデプロイメント・タイプ(クラウド、オンプレミスまたはハイブリッド)に適合するツールを選択することをお薦めします。

親トピック: Oracle Database 23cでサポートが終了した機能

Oracle Wallet Manager (OWM)のサポート終了

Oracle Database 23c以降、Oracle Wallet Manager (OWM)はサポートされなくなりました。

日付: 2023年4月

`orapki`コマンドライン・ツールを使用してOWMを置き換えることをお薦めします。

親トピック: Oracle Database 23cでサポートが終了した機能

RASADMのサポート終了

Real Application Security GUI管理ツール(RASADM)は、Oracle Database 23cでサポートが終了しました。

日付: 2023年4月

RASADMは、Oracle Database 23cでの使用がサポートされなくなりました。RASADMのすべての機能を含むRAS PL/SQL APIを使用することをお薦めします。

親トピック: Oracle Database 23cでサポートが終了した機能

Oracle Label Securityのパラメータおよびファンクションのサポート終了

Oracle Database 12cで非推奨となったOracle Label Security (OLS)のパラメータおよびファンクションは、Oracle Database 23cでサポートが終了しました。

日付: 2023年4月

以前に非推奨になったOLSのパラメータおよびファンクションには、使用できる代替パラメータがあります。

次のOLSファンクションはサポートが終了しました。

Least_UBOUND。かわりにOLS_GREATEST_LBOUNDを使用します。
LUBD。かわりにOLS_GLBDを使用します。
DOMINATES。かわりにOLS_DOMINATESスタンドアロン・ファンクションを使用してください。
DOM。かわりにOLS_STRICTLY_DOMINATESスタンドアロン・ファンクションを使用してください。
STRICTLY_DOMINATES。かわりにOLS_STRICTLY_DOMINATESスタンドアロン・ファンクションを使用してください。
S_DOM。かわりにOLS_STRICTLY_DOMINATESスタンドアロン・ファンクションを使用してください。
DOMINATED_BY。かわりにOLS_DOMINATED_BYスタンドアロン・ファンクションを使用してください。
DOM_BY。かわりにOLS_DOMINATED_BYスタンドアロン・ファンクションを使用してください。
STRICTLY_DOMINATED_BY。かわりにOLS_STRICTLY_DOMINATED_BYスタンドアロン・ファンクションを使用してください。
S_DOM_BY。かわりにOLS_STRICTLY_DOMINATED_BYスタンドアロン・ファンクションを使用してください。

また、OLSスタンドアロン・ファンクションSA_UTL.DOMINATESブール・データ型もサポートされなくなります。NUMBERデータ型を使用するSA_UTL.DOMINATESファンクションは非推奨ではありません。

代替方法の詳細は、『Oracle Label Security管理者ガイド』の「非推奨となった機能」を参照してください。

親トピック: Oracle Database 23cでサポートが終了した機能

Oracle Internet DirectoryとOracle Label Securityのサポート終了

Oracle Internet Directory (OID)とOracle Label Security (OLS)の統合は、Oracle Database 23cでサポートが終了しました。

日付: 2023年4月

OLSポリシーおよびラベルを格納するためのOIDの使用は、Oracle Database 23cでサポートが終了しました。この機能の代替は計画されていません。この機能を使用している場合は、この情報をコピーして保存するカスタム・メソッドを作成する必要があります。

親トピック: Oracle Database 23cでサポートが終了した機能

RADIUSユーザーへの管理権限の付与のサポート終了

Oracle Database 23c以降、レガシーRADIUS APIを使用してデータベースへの認証を行うユーザーには、管理権限が付与されなくなりました。

日付: 2023年4月

以前のリリースでは、RADIUS APIを使用して認証するユーザーには、SYSDBAやSYSBACKUPなどの管理権限を付与できました。Oracle Database 23cでは、最新の標準を使用する新しいRADIUS APIが導入されています。ユーザーに管理権限を付与するには、データベースへのデータベース接続で新しいRADIUS APIが使用されていること、およびOracle Database 23cクライアントを使用してOracle Database 23cサーバーに接続していることを確認します。

親トピック: Oracle Database 23cでサポートが終了した機能

Transparent Data Encryption PKIキーのサポート終了

Transparent Data Encryption (TDE)公開キー・インフラストラクチャ(PKI)キーはOracle Database 23cでサポートが終了しました

日付: 2023年4月

TDE PKIキーの非推奨は、Oracle Database 12cで発表されました。かわりに現在のTDEキーのいずれかを使用することをお薦めします。Oracle Database 23cをアップグレードする前に、ドキュメントに従って新しいTDEキーでデータベースのキー更新を行います。TDEで使用されるPKIキーがアップグレード・スクリプトによって検出された場合、スクリプトは停止し、アップグレードの前にTDEマスター・キーのキー更新を行うように通知されます。

関連項目

- キーストアのTDEマスター暗号化キーの設定またはキー更新

親トピック: Oracle Database 23cでサポートが終了した機能

GOSTおよびSEED TDE暗号化アルゴリズムのサポート終了

Oracle Database 23c以降、GOSTおよびSEEDアルゴリズムのTransparent Data Encryption (TDE)暗号化ライブラリはサポートされなくなり、削除されました。GOSTおよびSEED復号化ライブラリは非推奨です。どちらも、HP Itaniumプラットフォームでは削除されます。

日付: 2023年4月

GOST 28147-89はロシア政府によって非推奨とされ、SEEDは韓国政府によって非推奨とされました。韓国政府が承認したTDE暗号化が必要である場合は、かわりにARIAを使用します。GOST 28147-89を使用している場合は、サポートされている別のTDEアルゴリズムを使用して、復号化および暗号化する必要があります。GOST 28147-89およびSEEDの復号化アルゴリズムはOracle Database 23cに含まれていますが、非推奨であり、GOST暗号化アルゴリズムはOracle Database 23cでサポートが終了しました。TDE暗号化にGOSTまたはSEEDを使用している場合は、Oracle Database 23cにアップグレードする前に、別のアルゴリズムで復号化および暗号化することをお薦めします。ただし、HP Itaniumプラットフォームを除き、Oracle Database 23cではGOSTおよびSEED復号化ライブラリを使用できるため、アップグレード後に復号化することもできます。

親トピック: Oracle Database 23cでサポートが終了した機能

Oracle Database 10Gパスワード検証機能のサポート終了

Oracle Database 23c以降、10Gデータベース・パスワード検証機能はサポートされなくなりました。

日付: 2023年4月

Oracle Database 10gのデータベース・パスワード検証機能`10G`は、Oracle Database 23cでサポートされなくなったか、使用できなくなりました。Oracle Database 10Gデータベース・パスワード検証機能を識別する方法、および最新の最も安全なデータベース・パスワード検証機能の暗号化を使用するようにデータベース・ユーザーを更新する方法の詳細は、『データベース・アップグレード・ガイド』のインストール前の章を参照してください。

10G検証機能に関連付けられたパラメータは、Oracle Database 21cでサポートされなくなりました。

- `IGNORECASE`
- `SEC_CASE_SENSITIVE_LOGON`

Oracle Database 12cリリース2 (12.2)以降では、11Gおよび12Cの検証機能がデフォルトのパスワード検証機能でした。次の関連パラメータ値もサポートされなくなります。

- `SQLNET.ALLOWED_LOGON_VERSION_SERVER`値の8、9、10および11が削除されました(12と12aは残っています)。
- `SQLNET.ALLOWED_LOGON_VERSION_CLIENT`値の11が削除されました(12と12aは残っています)。

また、次の点にも注意してください。

- 12c検証機能を使用する10gおよび11gのパッチ適用されたクライアントは引き続き動作します。
- Database Net Servicesリファレンスにリストされているクライアント機能が変更されます。
- 03L、04Lおよび05Lクライアント機能がクライアント機能表から削除されます。

親トピック: Oracle Database 23cでサポートが終了した機能

Transport Layer Securityバージョン1.0および1.1のサポート終了

Database 23c以降、Transport Layer Securityプロトコル・バージョン1.0および1.1の使用はサポートされなくなりました。

日付: 2023年4月

データベース・クライアントおよびサーバーは最も安全なプロトコルおよび暗号アルゴリズムの使用をネゴシエートするため、ほとんどの場合、この変更による影響はありません。ただし、TLS 1.0または1.1が指定されている場合は、これを削除してデータベース・サーバーおよびクライアントが最も安全なプロトコルを選択できるようにするか、プロトコルにTLS 1.2またはTLS 1.3、あるいはその両方を指定する必要があります。最も安全な最新のプロトコルを使用することをお薦めします。そのプロトコルはTLS 1.3で、Oracle Database 23cで導入されています。

親トピック: Oracle Database 23cでサポートが終了した機能

Unix Crypt (MD5crypt)パスワード検証機能のサポート終了

Unix Crypt (`MD5crypt`)パスワード検証機能アルゴリズムは、Oracle Database 23cのサーバーおよびクライアントではサポートされなくなりました。

日付: 2023年4月

Oracle Internet Directory (OID)のユーザーを使用するエンタープライズ・ユーザー・セキュリティ(EUS)のお客様は、Unix Cryptによって、OIDによって、またはOIDへの移行前にオペレーティング・システムによって生成された、古い安全性の低いパスワード検証機能を使用できる可能性があります。パスワードをハッシュする現在の方法に比べ、Unix Cryptは安全性の低いアルゴリズムです。Oracle Databaseでは、それらの古いパスワード検証機能を使用してEUSまたはOIDユーザーを認証できなくなりました。新しいセキュアなハッシュ・アルゴリズムを使用して、OIDのパスワードを今すぐリセットすることをお薦めします。

親トピック: Oracle Database 23cでサポートが終了した機能

FIPS強度80の暗号化のサポート終了

FIPS暗号強度80は、Oracle Database 23cのリリースではサポートされなくなりました。かわりに、FIPS暗号強度112を使用してください。

日付: 2023年4月

FIPS暗号強度80は、RSA、Diffie-Hellman (DH)およびDigital Signature Algorithm (DSA)のキー長1024に相当します。Oracle Database 21cでのFIPSの使用では、低い強度0 (RSA/DH/DSAのキー長512)がサポートされなくなりました。Oracle Database 23cでサポートされているFIPS暗号強度は112 (RSA/DH/DSAのキー長2048)のみです。暗号強度80と0は、FIPS以外では引き続き使用できます。FIPS暗号強度112はFIPS暗号強度80よりもずっと強力であり、Oracle Database 23cではその長いキー長が使用されるため、FIPS 112を使用することをお薦めします。

親トピック: Oracle Database 23cでサポートが終了した機能

Diffie-Hellman匿名暗号のサポート終了

Oracle Database 23cでは、アウトバウンド接続とデータベース・クライアント/サーバー接続の両方について、Diffie-Hellman匿名暗号(DH anon)の使用はサポートされなくなりました。

日付: 2023年4月

DH anon暗号の削除により、Oracle Database接続のセキュリティが向上します。

Oracle Database 23cでは、次の3つの暗号はサポートされなくなりました。

- `TLS_DH_ANON_WITH_AES_256_GCM_SHA384`
- `TLS_DH_ANON_WITH_AES_128_GCM_SHA256`
- `SSL_DH_ANON_WITH_3DES_EDE_CBC_SHA`

エンタープライズ・ユーザー・セキュリティ(EUS)のお客様の場合は、OID接続でサーバーベースのTLS認証を使用していることを確認する必要があります。データベース・クライアント/サーバー接続の場合は、サーバーとクライアントを、可能なかぎり強力な接続になるようネゴシエートできるようにする必要があります。DH anon暗号を指定した場合は、その暗号を削除する必要と、データベース・サーバーが1方向TLS接続に対して認証されるようにする必要があります。

親トピック: Oracle Database 23cでサポートが終了した機能

Oracle Database Extensions for .NETのサポート終了

Oracle Database Extensions for .NETはサポートされなくなりました。.NETコードを中間層に配置するか、外部プロシージャ機能を使用するか、PL/SQLまたはJavaを使用してコードをリライトすることをお薦めします。

日付: 2023年4月

Oracle Database Extensions for .NETは、Microsoft Windows上のOracle Databaseの機能であり、.NETによって管理される言語(C#など)で記述されたストアド・プロシージャおよびファンクションが使用できるようになります。

Oracle Databaseは、Oracleデータベース・プロセスの外にある外部プロセス内でMicrosoft Common Language Runtime(CLR)をホストします。アプリケーション開発者は、C#やVB.NETなど、任意の.NET準拠言語を使用してストアド・プロシージャおよびファンクションを記述し、これらの.NETストアド・プロシージャを他のPL/SQLまたはJavaストアド・プロシージャと同様にデータベースで使用できます。.NETストアド・プロシージャは、PL/SQLパッケージ、プロシージャ、ファンクションおよびSQL文からのトリガー、またはPL/SQLプロシージャやファンクションをコール可能な場所ならどこからでもコールできます。

次の移行オプションがあります。

- 中間層への.NETコード(アセンブリ)の移動
- 外部プロシージャ機能を使用した、外部プロセスのロードおよび.NETアセンブリの実行
- PL/SQLまたはJavaを使用したストアド・プロシージャのリライト

親トピック: Oracle Database 23cでサポートが終了した機能

Quality of Service Managementのサポート終了

Oracle Databaseリリース23c以降では、Oracle Quality of Service Management (QoSMまたはQoS Management)はサポートされなくなりました。

日付: 2023年4月

Oracle QoSMでは、事前定義されたポリシーに基づいてシステム構成を調整し、必要なパフォーマンス・レベルでアプリケーションを継続的に実行できるようにすることで、システム全体のワークロード管理が自動化されます。アプリケーションおよびデータベースはますます、QoSMのリソース管理機能の一部を提供するシステムにデプロイされるようになってきています。同時に、OracleのAutonomous Health Frameworkは、システムとその関連コンポーネントの健全性および運用能力に影響を与える事象や状態を緩和するための推奨事項を調整したり提供するように強化されています。このため、QoSMおよびそのメモリー・ガー

ド機能は、Oracle Database 23cではサポートされなくなりました。

親トピック: Oracle Database 23cでサポートが終了した機能

従来の監査のサポート終了

Oracle Database 23cでは、従来の監査はサポートされなくなりました。統合監査を使用することをお薦めします。

日付: 2023年4月

Oracle Database 23c以降では、統合監査が、Oracle Database監査を実行するための方法です。統合監査では、選択的かつ効果的な監査をより柔軟に実行できます。これは、企業にとって重要なアクティビティに重点を置くために役立ちます。統合監査には、単一のセキュアな統合証跡、監査選択性に関する条件ポリシー、および簡潔性に関する事前構成済のデフォルト・ポリシーがあります。セキュリティとコンプライアンスの向上のために、統合監査の使用をお薦めします。

関連項目

- 従来の監査のサポート終了の処理

親トピック: Oracle Database 23cでサポートが終了した機能

config.shのサポート終了

Oracle Database 23c以降では、Oracle Grid Infrastructure構成ウィザード・ツール(`config.sh`)はサポートされなくなりました。かわりに`gridSetup.sh`を使用してください。

日付: 2023年4月

イメージベースのOracle Grid Infrastructureインストールの導入により、`config.sh`機能は廃止されました。`gridSetup.sh`スクリプトでは、config.shで以前使用されていたレスポンス・ファイル機能がサポートされています。

親トピック: Oracle Database 23cでサポートが終了した機能

OLS表のLABELS列のサポート終了

ALL_SA_USER_LABELSおよびDBA_SA_USER_LABELS OLS表のLABELS列は、Oracle Database 23cでサポートされなくなりました。

日付: 2023年4月

LABELS列は、他の列で使用可能な情報と重複するため、Oracle Label Security (OLS)に対しては以前に非推奨になっていました。この変更により、参照する列を選択する必要がなくなったため、これらの表の使いやすさが向上します。

親トピック: Oracle Database 23cでサポートが終了した機能

32ビットのOracle Databaseクライアントのサポート終了

32ビットのOracle Databaseクライアントは、Oracle Database 23cでサポートされなくなりました。

日付: 2023年4月

Oracleは、32ビットのOracle Databaseクライアントの開発を中止しました。64ビットのOracle Databaseクライアントを使用することをお薦めします。32ビット・アプリケーションを使用している場合は、「異なるOracleバージョンのクライアント/サーバーの相互運用性サポート・マトリックス(ドキュメントID 207303.1)」、およびリリース・サポート・ライフサイクル(My Oracle Support 742060.1)に従って、以前の32ビットOracle Databaseクライアントを引き続き使用できます。Thin JDBCや管理対象ODP.NETなどのThinまたはManagedデータ・クライアント・アクセス・ドライバを使用することもできます。

関連項目

- 異なるOracleバージョンのクライアント/サーバーの相互運用性サポート・マトリックス(ドキュメントID 207303.1)
- 現在のデータベース・リリースのリリース・スケジュール(ドキュメントID 742060.1)

親トピック: Oracle Database 23cでサポートが終了した機能

Oracle Sharding高可用性のためのOracle GoldenGateレプリケーションのサポート終了

Oracle Shardingのシャード・レベルの高可用性のためのOracle GoldenGateレプリケーションの使用が、Oracle Database 23cでサポートされなくなりました。

日付: 2023年4月

親トピック: Oracle Database 23cでサポートが終了した機能

グリッド・インフラストラクチャ管理リポジトリ(GIMR)のサポート終了

Oracle Database 23c以降、グリッド・インフラストラクチャ管理リポジトリ(GIMR)の使用はサポートされなくなりました。

日付: 2023年7月

Oracleは、別のリポジトリを使用してメタデータを格納しています。GIMRの使用はOracle Database 23cではサポートされていません。この移行は、Oracle Autonomous Health Framework (AHF)、Cluster Health Monitor (CHM)、Oracle Cluster Health Advisor (CHA)、Oracle Fleet Patching and Provisioning (FPP)など、GIMR (GIMRクライアント)を使用する機能に影響を与えないよう計画されています。

親トピック: Oracle Database 23cでサポートが終了した機能

OpatchおよびOPatchAutoを使用したアウトオブプレース・パッチ適用のサポート終了

アウトオブプレース・パッチ適用でのOpatchおよびOPatchAutoの使用は、Oracle Database 23cでサポートされなくなりました。

日付: 2023年7月

Oracle Database 23cでのパッチ適用の場合、Oracleでは、ゴールド・イメージを使用してアウトオブプレース・パッチ適用を使用して、四半期ごとのリリース更新(RU)を適用することをお薦めします。Oracleでは、アウトオブプレース・パッチ適用のためのOracle Database 23cでのOPatchおよびOPatchAutoの使用をサポートしていません。このサポート終了は、インプレース・パッチ適用には影響しません。インプレース・パッチ適用を実行する場合、OPatchおよびOPatchAutoはこの目的に引き続き使用できます。Oracle Database 23c Free - 開発者リリースは、RUでのパッチ適用をサポートしていません。

親トピック: Oracle Database 23cでサポートが終了した機能

データ・リカバリ・アドバイザ(DRA)のサポート終了

Oracle Database 23c以降、データ・リカバリ・アドバイザ(DRA)機能はサポートされなくなりました。

日付: 2023年7月

DRAのサポート終了には、`LIST FAILURE`、`ADVISE FAILURE`、`REPAIR FAILURE`および`CHANGE FAILURE`というOracle Recovery Manager (RMAN)コマンドのサポート終了が含まれています。データベース管理者はこれらのコマンドにアクセスできなくなります。DRAに代わる機能はありません。

親トピック: Oracle Database 23cでサポートが終了した機能

DBUAおよび手動アップグレード方法のサポート終了

Oracleでは、データベースのアップグレードにAutoUpgradeを使用することをお薦めします。

日付: 2023年7月

Oracle Databaseのアップグレードの信頼性とサポートを改善するために、Oracleでは、Database Upgrade Assistant (DBUA)、およびパラレル・アップグレード・ユーティリティ(`catctl.pl`)とデータベース・アップグレード・スクリプト`dbupgrade`および`dbupgrade.cmd`を使用する手動アップグレードのサポートを終了しています。Oracle Databaseアップグレードで AutoUpgradeを使用します。廃止されたアップグレード方法を置き換えることで、OracleはAutoUpgradeユーティリティの機能と管理性の改善と拡張に集中できます。

親トピック: Oracle Database 23cでサポートが終了した機能

Oracle OLAPのサポート終了

分析ワークスペース、OLAP DMLプログラミング言語およびOLAP Java APIは、Oracle Database 23cではサポートされなくなりました。

日付: 2023年7月

高度な分析機能が必要な新しいアプリケーションの場合、予測およびwhat-if分析のために分析ビュー(Oracle Databaseの機能)またはOracle Essbaseを検討することをお薦めします。Oracle分析ビューは、あらゆるOracle Databaseエディションの機能です。アプリケーションでディメンション問合せおよびレポート・アプリケーションにOLAPを使用する場合は、Oracle分析ビューをOLAPの代替として検討することをお薦めします。分析ビューは、既存のデータベース表およびビューに格納されているデータの分析問合せを迅速に効率的に作成する方法を提供します。Oracle分析ビューを使用すると、"キューブ・ビルド/更新"プロセスを必要とせずに、ディメンション問合せモデルおよびサポート・メタデータを取得できます。キューブ・ビルド/更新プロセスを省くことで、スケーラビリティ制約(モデルの複雑さとデータ量)が緩和され、データ準備パイプラインが簡素化され、データ待機時間が短縮または省かれます。

親トピック: Oracle Database 23cでサポートが終了した機能

Oracle Database 23cでサポートが終了したパラメータ

アップグレード計画の一環として、このOracle Databaseリリース以降、サポートされなくなった初期化パラメータを確認してください。

- SYS_CONTEXT USERENVでのEXTERNAL_NAMEパラメータのサポート終了
  SYS_CONTEXT USERENVのEXTERNAL_NAMEパラメータは、Oracle Database 23cでサポートが終了しました
- サービス属性値SESSION_STATE_CONSISTENCY = STATICのサポート終了
  SESSION_STATE_CONSISTENCY = STATICが指定されたセッション属性値FAILOVER_TYPE = TRANSACTIONは、サポートされているサービス属性の組合せではなくなりました。
- Oracle Label Securityのパラメータおよびファンクションのサポート終了
  Oracle Database 12cで非推奨となったOracle Label Security (OLS)のパラメータおよびファンクションは、Oracle Database 23cでサポートが終了しました。
- ENCRYPTION_WALLET_LOCATIONパラメータのサポート終了
  Oracle Database 23c以降では、パラメータ`ENCRYPTION_WALLET_LOCATION`はサポートされなくなりました。
- ADD_SSLV3_TO_DEFAULT SQLNET.ORAパラメータ(およびSSLv3)のサポート終了
  Oracle Database 23c以降では、データベースのサーバーとクライアントの接続でSecure Socket Layer v3プロ

トコル(SSLv3)がサポートされなくなり、sqlnet.oraのパラメータADD_SSLV3_TO_DEFAULTが削除されました。

親トピック: Oracle Database23cの動作の変更、サポート終了および非推奨

SYS_CONTEXT USERENVのEXTERNAL_NAMEパラメータのサポート終了

SYS_CONTEXT USERENVのEXTERNAL_NAMEパラメータは、Oracle Database 23cでサポートが終了しました

日付: 2023年4月

EXTERNAL_NAMEパラメータは、Oracle Database 10gで非推奨になりました。2つの新しいパラメータAUTHENTICATED_IDENTITYおよびENTERPRISE_IDENTITYが詳細情報を提供します。Oracle Database 23cではEXTERNAL_NAMEが削除されます。

親トピック: Oracle Database 23cでサポートが終了したパラメータ

サービス属性値SESSION_STATE_CONSISTENCY = STATICのサポート終了

SESSION_STATE_CONSISTENCY = STATICが指定されたセッション属性値FAILOVER_TYPE = TRANSACTIONは、サポートされているサービス属性の組合せではなくなりました。

日付: 2023年4月

以前のリリースでは、サービス・パラメータsession_stateを使用し、透過的アプリケーション・コンティニュイティを使用してセッション状態を自動的に管理したり、session_stateをDYNAMICまたはSTATICに設定して管理できました。しかし、Oracle Database 23c以降、STATICオプションを使用できなくなりました。かわりに、次のいずれかのフェイルオーバー・オプションを使用します。

- FAILOVER_TYPE = AUTOとSESSION_STATE_CONSISTENCY = AUTOまたはHYBRID
- FAILOVER_TYPE = TRANSACTIONおよびSESSION_STATE_CONSISTENCY = DYNAMIC

これらの構成により、Oracle Databaseでのセッション状態トラッキングが強制され、セッション移行およびセッションのフェイルオーバー時にセッション状態が維持されます。

Oracle Database 21c以降は、RESET_STATE属性を設定して、リクエストの最後にアプリケーションにより設定されたセッション状態をクリアすることもできます。詳細は、RESET_STATEを参照してください。計画メンテナンスの場合、完了していないリクエストについてアプリケーション・コンティニュイティと組み合せて、Oracle Database接続プールからリクエストを排出することをお薦めします。

関連項目

- セッション状態の一貫性

親トピック: Oracle Database 23cでサポートが終了したパラメータ

Oracle Label Securityのパラメータおよびファンクションのサポート終了

Oracle Database 12cで非推奨となったOracle Label Security (OLS)のパラメータおよびファンクションは、Oracle Database 23cでサポートが終了しました。

日付: 2023年4月

以前に非推奨になったOLSのパラメータおよびファンクションには、使用できる代替パラメータがあります。

次のOLSファンクションはサポートが終了しました。

Least_UBOUND。かわりにOLS_GREATEST_LBOUNDを使用します。

LUBD。かわりにOLS_GLBDを使用します。
DOMINATES。かわりにOLS_DOMINATESスタンドアロン・ファンクションを使用してください。
DOM。かわりにOLS_STRICTLY_DOMINATESスタンドアロン・ファンクションを使用してください。
STRICTLY_DOMINATES。かわりにOLS_STRICTLY_DOMINATESスタンドアロン・ファンクションを使用してください。
S_DOM。かわりにOLS_STRICTLY_DOMINATESスタンドアロン・ファンクションを使用してください。
DOMINATED_BY。かわりにOLS_DOMINATED_BYスタンドアロン・ファンクションを使用してください。
DOM_BY。かわりにOLS_DOMINATED_BYスタンドアロン・ファンクションを使用してください。
STRICTLY_DOMINATED_BY。かわりにOLS_STRICTLY_DOMINATED_BYスタンドアロン・ファンクションを使用してください。
S_DOM_BY。かわりにOLS_STRICTLY_DOMINATED_BYスタンドアロン・ファンクションを使用してください。

また、OLSスタンドアロン・ファンクションSA_UTL.DOMINATESブール・データ型もサポートされなくなります。NUMBERデータ型を使用するSA_UTL.DOMINATESファンクションは非推奨ではありません。

代替方法の詳細は、『Oracle Label Security管理者ガイド』の「非推奨となった機能」を参照してください。

親トピック: Oracle Database 23cでサポートが終了したパラメータ

ENCRYPTION_WALLET_LOCATIONパラメータのサポート終了

Oracle Database 23c以降では、ENCRYPTION_WALLET_LOCATIONパラメータはサポートされなくなりました。

日付: 2023年4月

TDEウォレットを格納および取得するには、WALLET_ROOT構造(Oracle Database 18cで導入)を使用します。

> ノート:
>
> Transparent Data Encryption (TDE)が有効になっているが、WALLET_ROOT が構成されていない場合は、Oracle Database 23c へのアップグレードがブロックされます。TDE を使用したデータベースのアップグレードに対するこのブロックは、アップグレード後にデータベースをオープンできない可能性を回避するためです。

親トピック: Oracle Database 23cでサポートが終了したパラメータ

ADD_SSLV3_TO_DEFAULT SQLNET.ORAパラメータ(およびSSLv3)のサポート終了

Oracle Database 23c以降では、データベースのサーバーとクライアントの接続でSecure Socket Layer v3プロトコル(SSLv3)がサポートされなくなり、sqlnet.oraのパラメータADD_SSLV3_TO_DEFAULTが削除されました。

日付: 2023年4月

データベースのサーバーとクライアント間の接続を保護する上で、SSLv3はあまり安全ではないプロトコルです。SSLv3を使用するかわりに、サーバーとクライアント間で共通する最もセキュアなプロトコルを、データベースのサーバーとクライアントがネゴシエートできるようになります。Oracle Database 23cには、証明書ベースのネットワーク暗号化用にTLS 1.2およびTLS 1.3プロトコルが用意されています。

親トピック: Oracle Database 23cでサポートが終了したパラメータ

Oracle Database 23cで非推奨となった機能

アップグレード計画の一環として、このOracle Databaseリリースで非推奨となった機能を確認し、アプリケーション戦略に対する代替方法を確認してください。

- PROXY_ONLY_CONNECTの非推奨
  スキーマをPROXY_ONLY_CONNECTに設定する機能は、Oracle Database 23cでは非推奨です。
- Oracle Database 11gのSHA-1検証機能の非推奨
  Oracle Database 23c以降、Oracle Database 11gで導入されたSHA-1パスワード検証機能は非推奨です。
- RFC-2138に基づくRADIUS APIの非推奨
  Oracle Database 23c以降、Request for Comments (RFC) 2138に基づく古いRADIUS APIは非推奨です。
- エンタープライズ・ユーザー・セキュリティ(EUS)の非推奨
  エンタープライズ・ユーザー・セキュリティ(EUS)は、Oracle Database 23cで非推奨になりました。
- 自動ログイン・ウォレット・バージョン6の非推奨
  Oracle Database 23cでは、新しい自動ログイン・ウォレット・バージョン(7)が導入されました。Oracleローカル自動ログイン・ウォレットのバージョン6は非推奨です。
- WALLET_LOCATIONパラメータの非推奨
  パラメータWALLET_LOCATIONは、Oracle Databaseサーバー用のOracle Database 23cでの使用が非推奨になりました。Oracle Databaseクライアントでの使用は非推奨ではありません。
- Fusion Middlewareに対するRAS中間層セッションのサポートの非推奨
  Fusion Middlewareアプリケーション・セッション・サービスに対するOracle Real Application Security (RAS)中間層マスター・セッションのサポートは、Oracle Database 23cで非推奨になりました
- Oracle Data Provider for .NET管理対象外ドライバの非推奨
  Oracle Data Provider for .NET (ODP.NET)管理対象外ドライバは、Oracle Database 23cでは非推奨です。
- GOSTおよびSEEDアルゴリズムの非推奨
  Oracle Database 23c以降では、GOSTおよびSEEDアルゴリズムのTransparent Data Encryption (TDE)復号化ライブラリは非推奨になり、GOSTおよびSEEDへの暗号化はサポートされなくなります。
- Oracle Persistent Memoryの非推奨
  Oracle Persistent Memory Database (PMEM)は、IntelがOptane Persistent Memoryハードウェアの製造を中止したため、Oracle Database 23cで非推奨となりました。
- Oracle Memory Speed File SystemでのPMEMサポートは非推奨
  Oracle Memory Speed (OMS) File Systemは、Oracle Database 23cでIntel Optane Persistent Memoryでの使用が非推奨となりました。
- 部分DN一致のサービス名およびサーバー専用の証明書チェックの非推奨
  Oracle Database 23c以降では、(部分および完全)一致がデータベース・サーバー証明書に限定されなくなりました。リスナー証明書もチェックされ、部分DN一致の場合はSERVICE_NAMEパラメータが無視されます。
- mkstoreコマンドライン・ユーティリティの非推奨
  mkstoreウォレット管理コマンドライン・ツールは、Oracle Database 23cでは非推奨であり、将来のリリースで削除される可能性があります。
- ネットワーク・データ・モデル(NDM) XML APIは非推奨
  Oracle SpatialのNDM機能で使用されるネットワーク・データ・モデル(NDM) XML APIは、Oracle Databaseリリース23cでは非推奨となっています。
- SDO_GEOR_ADMINパッケージ内の2つのサブプログラムが非推奨
  SDO_GEOR_ADMINパッケージ内のGeoRasterサブプログラムSDO_GEOR_ADMIN.isUpgradeNeededおよびSDO_GEOR_ADMIN.upgradeGeoRasterは、データベース内のGeoRasterオブジェクトのメンテナンスに使用されるものであり、Oracle Databaseリリース23cでは非推奨となっています。

- SDO_GEORパッケージ内のSDO_GEOR.importFromおよびSDO_GEOR.exportToサブプログラムは非推奨
  SDO_GEORパッケージ内のGeoRasterサブプログラムSDO_GEOR.importFromおよびSDO_GEOR.exportToは、Oracle Databaseリリース23cでは非推奨となっています。
- Oracle Data GuardのSWITCHOVERコマンドのWAITオプション
  Oracle Database 23c以降では、Oracle Data GuardのSWITCHOVERコマンドのWAITオプションが非推奨になりました。
- DBMS_RESULT_CACHEファンクション名の非推奨
  Oracle Database 23cでは、複数のDBMS_RESULT_CACHEのファンクション名が変更されます。
- Oracle ACFSスナップショットの再マスタリングの非推奨
  Oracle Databaseリリース23c以降では、Oracle Advanced Cluster File System (ACFS)のスナップショットの再マスタリング機能が非推奨になりました。
- Oracle ACFS圧縮の非推奨
  Oracle Advanced Cluster File System (ACFS)のACFS圧縮機能は、Oracle Database 23cでは非推奨になりました。
- Oracle Virtual DirectoryとReal Application Securityの非推奨
  Oracle Virtual DirectoryとOracle Real Application Securityの使用は、Oracle Database 23cで非推奨になりました。
- Oracle TextのBIG_IO属性の非推奨
  CONTEXT索引タイプのBIG_IO属性は、Oracle Database 23cで非推奨になり、将来のリリースで無効化または削除される可能性があります。
- Oracle TextのASYNCHRONOUS属性の非推奨
  CONTEXT索引タイプのASYNCHRONOUS属性は、Oracle Database 23cで非推奨になり、将来のリリースで無視または削除される可能性があります。
- Oracle Text CTXCAT索引タイプの非推奨
  Oracle Text索引タイプCTXCATは、Oracle Database 23cでは非推奨です。索引タイプ自体とその演算子CTXCATは、将来のリリースで削除される可能性があります。
- Oracle XML DBリポジトリの非推奨
  Oracle XML DBリポジトリは、Oracle Database 23cでは非推奨です。
- DBMS_XMLGENの非推奨
  PL/SQLパッケージDBMS_XMLGENは、Oracle Database 23cで非推奨になりました。
- DBMS_XMLSTOREの非推奨
  PL/SQLパッケージDBMS_XMLSTOREは、Oracle Database 23cで非推奨になりました。
- 非構造化XML索引の非推奨
  非構造化XML索引は、Oracle Database 23cでは非推奨です。
- DBMS_HANG_MANAGERパッケージの非推奨
  DBMS_HANG_MANAGERパッケージは、Oracle Database 23cでは非推奨です。かわりにDBMS_BLOCKER_RESOLVERを使用します。
- 停止時間ゼロのアップグレード(ZDU)の非推奨
  Oracle Fleet Patching and Provisioning (FPP)の停止時間ゼロのアップグレード(ZDU)機能は、Oracle Database 23cでは非推奨です。
- グリッド・ネーミング・サービス(GNS)の非推奨
  Oracle Grid Infrastructureのグリッド・ネーミング・サービス(GNS)機能は、Oracle Database 23cでは非推奨です。

- 従来の監査パッケージおよびファンクションの非推奨
  従来の監査パッケージおよびファンクションは、Oracle Database 23cでは非推奨です。

親トピック: Oracle Database23cの動作の変更、サポート終了および非推奨

PROXY_ONLY_CONNECTの非推奨

スキーマをPROXY_ONLY_CONNECTに設定する機能は、Oracle Database 23cでは非推奨です。

日付: 2023年4月

スキーマのPROXY_ONLY_CONNECTへの設定はこのリリースで非推奨となり、さらに将来のリリースではサポートされなくなる可能性があります。PROXY_ONLY_CONNECTのかわりにSCHEMA ONLYユーザー・アカウントを使用することをお薦めします。スキーマ専用アカウントにはスキーマに関連付けられたパスワードがないため、スキーマに直接ログインすることはできず、パスワードをローテーションする必要はありません。

親トピック: Oracle Database 23cで非推奨となった機能

Oracle Database 11gのSHA-1検証機能の非推奨

Oracle Database 23c以降、Oracle Database 11gで導入されたSHA-1検証機能は非推奨です。

日付: 2023年4月

Oracle Database 12cで導入されたsalt付きの複数ラウンドSHA-512パスワード・ハッシュ("検証機能"とも呼ばれる)は、パスワードのセキュリティを強化します。データベースで11gの検証機能(11G)がまだ使用されている場合は、12c (12C)の最適化解除済PBKDF2ベースの検証機能にアップグレードできるように、リセットすることをお薦めします。

親トピック: Oracle Database 23cで非推奨となった機能

RFC-2138に基づくRADIUS APIの非推奨

Oracle Database 23c以降、Request for Comments (RFC) 2138に基づく古いRADIUS APIは非推奨です。

Oracle Database 23cでは、RFC 6613およびRFC 6614に基づく更新済RADIUS APIが導入されています。新しいRADIUS APIを使用するために、できるだけ早く移行を計画することをお薦めします。新しいAPIは、デフォルトで有効になっています。古いRADIUS APIに関連付けられたパラメータも非推奨です(SQLNET.RADIUS_ALTERNATE、SQLNET.RADIUS_ALTERNATE_PORT、SQLNET.RADIUS_AUTHENTICATIONおよびSQLNET.RADIUS_AUTHENTICATION_PORT)。古いRADIUS APIを使用するようにデフォルトを変更する方法の詳細は、RADIUS APIのドキュメントを参照してください。

親トピック: Oracle Database 23cで非推奨となった機能

エンタープライズ・ユーザー・セキュリティ(EUS)の非推奨

エンタープライズ・ユーザー・セキュリティ(EUS)は、Oracle Database 23cで非推奨になりました。

集中管理ユーザー(CMU)の使用に移行することをお薦めします。この機能を使用すると、エンタープライズ・ユーザー認証およびデータベースへの認可用のディレクトリ・サービスを介在させることなく、Microsoft Active Directoryに直接接続できます。Oracle Databaseがクラウドにある場合は、クラウド・アイデンティティ・プロバイダとの新しい統合の1つに移行することもできます。

親トピック: Oracle Database 23cで非推奨となった機能

自動ログイン・ウォレット・バージョン6の非推奨

Oracle Database 23cでは、新しい自動ログイン・ウォレット・バージョン(7)が導入されました。Oracleローカル自動ログイン・ウォレットのバージョン6は非推奨です。

日付: 2023年4月

ローカルの自動ログイン・ウォレットを更新するには、`orapki`を使用して変更します。

親トピック: Oracle Database 23cで非推奨となった機能

WALLET_LOCATIONパラメータの非推奨

パラメータ`WALLET_LOCATION`は、Oracle Databaseサーバー用のOracle Database 23cでの使用が非推奨になりました。Oracle Databaseクライアントでの使用は非推奨ではありません。

日付: 2023年4月

Oracle Databaseサーバーの場合、`WALLET_LOCATION`を使用するかわりに`WALLET_ROOT`システム・パラメータを使用することをお薦めします。

親トピック: Oracle Database 23cで非推奨となった機能

Fusion Middlewareに対するRAS中間層セッションのサポートの非推奨

Fusion Middlewareアプリケーション・セッション・サービスに対するOracle Real Application Security (RAS)中間層マスター・セッションのサポートは、Oracle Database 23cで非推奨になりました

日付: 2023年4月

このフィルタは、FMWアプリケーション・サーバーのRASセッションを自動的に作成します。Oracle Database 23c以降では、他のアプリケーション・サーバーの場合と同様に、今後のRASセッションを作成するにはコードを作成する必要があります。

親トピック: Oracle Database 23cで非推奨となった機能

Oracle Data Provider for .NET管理対象外ドライバの非推奨

Oracle Data Provider for .NET (ODP.NET)管理対象外ドライバは、Oracle Database 23cでは非推奨です。

日付: 2023年4月

ODP.NETは、Oracle DatabaseへのADO.NETベースのデータ・アクセスを提供します。Microsoft .NET Frameworkには、ODP.NET管理対象ドライバとODP.NET管理対象外ドライバという2つのプライマリOracleデータ・アクセス・ドライバがあります。Oracle Database 23cでは、ODP.NET管理対象ドライバは、同じアプリケーション・プログラミング・インタフェースおよび構成設定を持つODP.NET管理対象外ドライバで使用可能なすべての主要な機能をサポートします。既存の.NETアプリケーションの大部分では、管理対象外のODP.NETから管理対象のODP.NETへのコード移行は簡単です。既存の管理対象外のODP.NETアプリケーションをODP.NET管理対象ドライバに移行することをお薦めします。ODP.NET管理対象外ドライバは、将来のリリースでサポートされなくなる可能性があります。

ODP.NET管理対象ドライバは、NuGetパッケージングを介して消費できる、よりコンパクトで簡単なインストールです。管理対象外のODP.NETは、NuGetパッケージとして使用できません。複数の管理対象外ODP.NETデプロイメントよりも、同じマシン上の複数の管理対象ODP.NETデプロイメントを管理する方が容易です。これらのメリットにより、お客様の使用には管理対象外のODP.NETよりも管理対象のODP.NETが優先されます。管理対象外のODP.NETには管理対象のODP.NETに勝る利点がないため、管理対象外のODP.NETは非推奨になりました。

親トピック: Oracle Database 23cで非推奨となった機能

GOSTおよびSEEDアルゴリズムの非推奨

Oracle Database 23c以降では、GOSTおよびSEEDアルゴリズムのTransparent Data Encryption (TDE)復号化ライブラリは非推奨になり、GOSTおよびSEEDへの暗号化はサポートされなくなります。

日付: 2023年4月

GOST 28147-89はロシア政府によって非推奨とされ、SEEDは韓国政府によって非推奨とされました。韓国政府が承認したTDE暗号化が必要である場合は、かわりにARIAを使用します。GOST 28147-89を使用している場合は、サポートされている別のTDEアルゴリズムを使用して、復号化および暗号化する必要があります。GOST 28147-89およびSEEDの復号化アルゴリズムはOracle Database 23cに含まれていますが非推奨となり、GOST暗号化アルゴリズムはOracle Database 23cでサポートが終了しました。TDE暗号化にGOSTまたはSEEDを使用している場合は、Oracle Database 23cにアップグレードする前に、別のアルゴリズムで復号化および暗号化することをお薦めします。ただし、HP Itaniumプラットフォームを除き、Oracle Database 23cではGOSTおよびSEED復号化ライブラリを使用できるため、アップグレード後に復号化することもできます。

親トピック: Oracle Database 23cで非推奨となった機能

Oracle Persistent Memoryの非推奨

Oracle Persistent Memory Database (PMEM)は、IntelがOptane Persistent Memoryハードウェアの製造を中止したため、Oracle Database 23cで非推奨となりました。

日付: 2023年4月

Intelは、Optane Persistent Memory製品の製造中止を発表しました。そのため、Oracle Persistent Memory Databaseは非推奨となっています。

親トピック: Oracle Database 23cで非推奨となった機能

Oracle Memory Speed File SystemでのPMEMサポートは非推奨

Oracle Memory Speed (OMS) File Systemは、Oracle Database 23cでIntel Optane Persistent Memoryでの使用が非推奨となりました。

日付: 2023年4月

Intelは、Optane Persistent Memory製品の製造中止を発表しました。そのため、Oracle Memory Speed (OMS) File Systemの使用は非推奨となっています。

親トピック: Oracle Database 23cで非推奨となった機能

部分DN一致のサービス名およびサーバー専用の証明書チェックの非推奨

Oracle Database 23c以降では、(部分および完全)一致がデータベース・サーバー証明書に限定されなくなりました。リスナー証明書もチェックされ、部分DN一致の場合は`SERVICE_NAME`パラメータが無視されます。

日付: 2023年4月

識別名(DN)によるサーバー側の証明書の検証が、リスナー証明書とデータベース・サーバー証明書の両方がチェックされるように変更されました。以前のOracle Databaseリリースでは、データベース・サーバー証明書のみがチェックされていました。本番ではほとんどのケースで、リスナーとデータベースで同じ証明書が使用されます。異なる証明書が使用されている場合、DN一致では、SANまたはホスト名証明書情報での部分DN一致を可能にするために新しい証明書が必要になることがあります。リスナー証明書の確認に加えて、部分DN一致を使用する際には`SERVICE_NAME`パラメータが無視されて、ホスト名接続文字

列パラメータのみが証明書共通名(CN)およびサブジェクト代替名(SAN)フィールドに対してチェックされます。以前のリリースでの動作(ホスト名に加えてサービス名を使用し、データベース・サーバー証明書のみをチェックする)に戻すには、新しいパラメータ`SSL_ALLOW_WEAK_DN_MATCH=TRUE`を設定します。デフォルトは`FALSE`です。

親トピック: Oracle Database 23cで非推奨となった機能

mkstoreコマンドライン・ユーティリティの非推奨

`mkstore`ウォレット管理コマンドライン・ツールは、Oracle Database 23cでは非推奨であり、将来のリリースで削除される可能性があります。

日付: 2023年4月

ウォレットの管理には、`orapki`コマンドライン・ツールの使用をお薦めします。

親トピック: Oracle Database 23cで非推奨となった機能

ネットワーク・データ・モデル(NDM) XML APIは非推奨

Oracle SpatialのNDM機能で使用されるネットワーク・データ・モデル(NDM) XML APIは、Oracle Databaseリリース23cでは非推奨となっています。

日付: 2023年4月

カスタマイズしたネットワーク分析を効率的かつ安全に実装するために、Oracle SpatialではNDM XML APIを非推奨にしています。かわりに、Load On Demand (LOD) APIを使用します。LOD APIは、カスタマイズを簡単かつ安全に実装できるカスタマイズ・インタフェースを提供します。また、NDM縮約階層REST APIを使用して高パフォーマンス・ネットワーク分析を行うこともできます。

親トピック: Oracle Database 23cで非推奨となった機能

SDO_GEOR_ADMINパッケージ内の2つのサブプログラムが非推奨

`SDO_GEOR_ADMIN`パッケージ内のGeoRasterサブプログラム`SDO_GEOR_ADMIN.isUpgradeNeeded`および`SDO_GEOR_ADMIN.upgradeGeoRaster`は、データベース内のGeoRasterオブジェクトのメンテナンスに使用されるものであり、Oracle Databaseリリース23cでは非推奨となっています。

日付: 2023年4月

GeoRasterデータベースのサイズが大きくなると、`SDO_GEOR_ADMIN.isUpgradeNeeded`および`SDO_GEOR_ADMIN.upgradeGeoRaster`サブプログラムのパフォーマンスが効率的ではなくなります。Oracle Database内のGeoRasterオブジェクトとそのシステム・データのメンテナンスには、かわりに、`SDO_GEOR_ADMIN.checkSysdataEntries`や`SDO_GEOR_ADMIN.maintainSysdataEntries`などの代替サブプログラムと、その他のいくつかのサブプログラムを使用することをお薦めします。

GeoRasterデータベースのメンテナンスのために代替サブプログラムを実行する方法の詳細は、『*Oracle Spatial Spatial GeoRaster*開発者ガイド』を参照してください。

親トピック: Oracle Database 23cで非推奨となった機能

SDO_GEORパッケージ内のSDO_GEOR.importFromおよびSDO_GEOR.exportToサブプログラムは非推奨

`SDO_GEOR`パッケージ内のGeoRasterサブプログラム`SDO_GEOR.importFrom`および`SDO_GEOR.exportTo`は、

Oracle Databaseリリース23cでは非推奨となっています。

日付: 2023年4月

SDO_GEOR.importFromおよびSDO_GEOR.exportToサブプログラムでは、いくつかの一般的なイメージ・ファイル形式で格納された、非常に小さいイメージがサポートされています。これらは、パフォーマンスが低いため、多くの使用制限があり、Oracle Autonomous Databaseではサポートされていません。イメージおよびラスター・ファイルをロードおよびエクスポートするには、非推奨となったこれらのサブプログラムを、SDO_GEOR_GDAL.translateサブプログラム、GDAL、またはGDALベースのGeoRaster ETLなどの代替オプションに置き換えることをお薦めします。これらのオプションでは、多くの制限はなく、100種類を超える形式での様々な地理空間イメージがサポートされており、パフォーマンスとスケーラビリティが大幅に高くなります。

親トピック: Oracle Database 23cで非推奨となった機能

Oracle Data GuardのSWITCHOVERコマンドのWAITオプション

Oracle Database 23c以降では、Oracle Data GuardのSWITCHOVERコマンドのWAITオプションが非推奨になりました。

日付: 2023年4月

Oracle Database 21cでは、新しいOracle Data GuardのプロパティDrainTimeoutが導入されました。このプロパティにより、SWITCHOVERの発行時に、Oracle Clusterwareに登録されているデータベース・サービスのdrain_timeoutのデフォルトがオーバーライドされます。Oracle Data GuardのSWITCHOVERコマンドのWAITオプションが冗長になりました。新しいDrainTimeoutプロパティを使用することをお薦めします。

親トピック: Oracle Database 23cで非推奨となった機能

DBMS_RESULT_CACHEファンクション名の非推奨

Oracle Database 23cでは、複数のDBMS_RESULT_CACHEのファンクション名が変更されます。

日付: 2023年4月

次のファンクションおよびプロシージャが非推奨になりました:

- BLACK_LISTファンクション。BLOCK_LISTファンクションを使用してください。
- BLACK_LIST_ADDプロシージャ。BLOCK_LIST_ADDプロシージャを使用してください。
- BLACK_LIST_CLEARプロシージャ。BLOCK_LIST_CLEARプロシージャを使用してください
- BLACK_LIST_REMOVEプロシージャ。BLOCK_LIST_REMOVEプロシージャを使用してください
- OBJECT_BLACK_LISTファンクション。OBJECT_BLOCK_LISTファンクションを使用してください
- OBJECT_BLACK_LIST_ADDプロシージャ。OBJECT_BLOCK_LIST_ADDプロシージャを使用してください。
- OBJECT_BLACK_LIST_CLEARプロシージャ。OBJECT BLOCK LIST_CLEARプロシージャを使用してください。
- OBJECT_BLACK_LIST_REMOVEプロシージャ。OBJECT_BLOCK_LIST_REMOVEプロシージャを使用してください。

親トピック: Oracle Database 23cで非推奨となった機能

Oracle ACFSスナップショットの再マスタリングの非推奨

Oracle Databaseリリース23c以降では、Oracle Advanced Cluster File System (ACFS)のスナップショットの再マスタリング機能が非推奨になりました。

日付: 2023年4月

この機能は、将来のリリースでサポートされなくなり、代替機能もない可能性があります。採用数が少ない機能のサポートを終

了することにより、中核となるスケーリング、可用性および管理性の向上に集中できます。

親トピック: Oracle Database 23cで非推奨となった機能

Oracle ACFS圧縮の非推奨

Oracle Advanced Cluster File System (ACFS)のACFS圧縮機能は、Oracle Database 23cでは非推奨になりました。

日付: 2023年4月

この機能は、将来のリリースでサポートされなくなり、代替機能もない可能性があります。採用数が少ない機能のサポートを終了することにより、中核となるスケーリング、可用性および管理性の向上に集中できます。

親トピック: Oracle Database 23cで非推奨となった機能

Oracle Virtual DirectoryとReal Application Securityの非推奨

Oracle Virtual DirectoryとOracle Real Application Securityの使用は、Oracle Database 23cでは非推奨です。

日付: 2023年7月

OVDとOracle Real Application Securityの使用は、OVDが別の製品として更新されなくなるため非推奨です

親トピック: Oracle Database 23cで非推奨となった機能

Oracle TextのBIG_IO属性非推奨

`CONTEXT`索引タイプの`BIG_IO`属性は、Oracle Database 23cで非推奨になり、将来のリリースで無効化または削除される可能性があります。

日付: 2023年7月

Oracleでは、この値をデフォルト値Nに設定することをお薦めします。`BIG_IO`は、索引ポストの長さが4KBを超えた場合にシークのコストを削減するために導入されました。ただし、内部コードは比較的非効率であり、属性を新しい索引オプションと組み合せることはできません。シーク・コストは、ソリッド・ステート・ディスクまたは非揮発性メモリー・デバイス(NVMe)にはあまり関連せず、ポストがキャッシュされている場合、シーク・コストは関係ありません。したがって、この設定はほとんどの索引にとってほとんどメリットがありません。

親トピック: Oracle Database 23cで非推奨となった機能

Oracle TextのASYNCHRONOUS属性の非推奨

`CONTEXT`索引タイプの`ASYNCHRONOUS`属性は、Oracle Database 23cで非推奨になり、将来のリリースで無視または削除される可能性があります。

日付: 2023年7月

Oracleでは、将来のリリースでこの属性を無視または削除する可能性があります。Oracleでは、この値をデフォルト値Nに設定することをお薦めします。更新時の予期しない結果損失を回避するには、`SYNC (ON COMMIT)`または `SYNC(EVERY [time-period])`を短い期間で使用します。

`ASYNCHRONOUS`設定は、更新が"delete followed by insert"として実装され、削除が即時(コミット時)であり、挿入が索引同期中にのみ実行されるという事実の回避策として導入されました。ただし、この設定は他のいくつかの索引オプションと互換性がありません。Oracleでは、使用を中止することをお薦めします。

親トピック: Oracle Database 23cで非推奨となった機能

Oracle Text CTXCAT索引タイプの非推奨

Oracle Textの索引タイプCTXCATは、Oracle Database 23cでは非推奨です。索引タイプ自体とその演算子CTXCATは、将来のリリースで削除される可能性があります。

日付: 2023年7月

CONTEXT問合せの代替文法としてCTXCATおよびCTXCAT文法の使用はどちらも非推奨です。かわりに、Oracleでは、CONTEXT索引タイプを使用することをお薦めします。索引タイプは、トランザクション以外のすべての同じ機能を提供できます。CONTEXTでのトランザクションに近い動作は、SYNC(ON COMMIT)または短い期間(できればSYNC(EVERY [time-period]))を使用して実現できます。

CTXCATは、通常、索引のサイズが数メガバイトのときに導入されました。最近の大規模な索引は、CTXCATで管理が困難な場合があります。CTXCATへの索引セットの追加は、CONTEXT索引タイプでFILTER BY列とORDER BY列、またはSDATA列(あるいはその両方)を使用することで、より効率的に実行できます。したがって、CTXCATが適切な選択になることはほとんどありません。Oracleでは、より効率的なCONTEXT索引タイプを選択することをお薦めします。

親トピック: Oracle Database 23cで非推奨となった機能

Oracle XML DBリポジトリの非推奨

Oracle XML DBリポジトリは、Oracle Database 23cでは非推奨です。

日付: 2023年7月

Oracleでは、XML DBリポジトリで使用される機能を代替テクノロジに置き換えることをお薦めします。

親トピック: Oracle Database 23cで非推奨となった機能

DBMS_XMLGENの非推奨

PL/SQLパッケージDBMS_XMLGENは、Oracle Database 23cで非推奨になりました。

日付: 2023年7月

DBMS_XMLGENは、SQL問合せまたはPL/SQLを使用してXMLドキュメントを生成および変換するために提供される非標準のOracle独自のパッケージです。このパッケージは非推奨で、今後のリリースでサポートされなくなる可能性があります。Oracleでは、SQL/XML演算子を使用して、かわりにリレーショナル列からXMLを生成することをお薦めします。XMLドキュメントの生成および変更にANSI SQL/XML演算子を使用すると、XMLドキュメントを操作するための標準化された将来的な方法が提供されます。

親トピック: Oracle Database 23cで非推奨となった機能

DBMS_XMLSTOREの非推奨

PL/SQLパッケージDBMS_XMLSTOREは、Oracle Database 23cで非推奨になりました。

日付: 2023年7月

DBMS_XMLSTOREは、Oracle DatabaseでXMLデータを格納および操作できる非標準のOracle独自のパッケージです。このパッケージは非推奨で、今後のリリースでサポートされなくなる可能性があります。Oracleでは、通常のSQL DMLを使用し、標準のXQueryおよびSQL/XMLを使用してXMLデータを格納および管理することをお薦めします。標準機能を使用すると、XMLデータを格納および操作するための将来的にも有効な方法が提供されます。

親トピック: Oracle Database 23cで非推奨となった機能

非構造化XML索引の非推奨

非構造化XML索引は、Oracle Database 23cでは非推奨です。

日付: 2023年7月

非構造化XML索引は非推奨であり、XML検索索引に置き換えられます。XML検索索引または構造化XML索引を使用することをお薦めします。

親トピック: Oracle Database 23cで非推奨となった機能

DBMS_HANG_MANAGERパッケージの非推奨

DBMS_HANG_MANAGERパッケージは、Oracle Database 23cでは非推奨です。かわりにDBMS_BLOCKER_RESOLVERを使用します。

日付: 2023年7月

DBMS_HANG_MANAGERパッケージは、セッションの問題に対処するために一部の構成パラメータおよび制約を変更する方法を提供します。このパッケージはDBMS_BLOCKER_RESOLVERに置き換えられます。DBMS_HANG_MANAGERは、将来のリリースで削除される可能性があります。

親トピック: Oracle Database 23cで非推奨となった機能

停止時間ゼロのアップグレード(ZDU)の非推奨

Oracle Fleet Patching and Provisioning (FPP)の停止時間ゼロのアップグレード(ZDU)機能は、Oracle Database 23cで非推奨になりました。

日付: 2023年7月

ZDUでは、一時的なロジカル・スタンバイまたはOracle GoldenGateクラシックが使用されます。ただし、一時ロジカル・スタンバイは最新のデータ型をサポートしておらず、Oracle GoldenGateのクラシック実装は非推奨です。ZDUのかわりに、Oracleでは、DBMS_ROLLING PL/SQLパッケージと組み合せてFPPを使用することをお薦めします。新規Oracleホームのプロビジョニングおよびデプロイメントには FPPを使用でき、ローリング・アップグレードの効率的な方法としてDBMS_ROLLINGを使用できます。

DBMS_ROLLINGパッケージはアプリケーション・コンティニュイティと緊密に統合されており、データベース層の中断を最小限に抑え、ローリング・パッチ適用アクティビティをマスクするために役立ちます。

関連項目

- DBMS_ROLLINGを使用したローリング・アップグレードの実行

親トピック: Oracle Database 23cで非推奨となった機能

グリッド・ネーミング・サービス(GNS)の非推奨

Oracle Grid Infrastructureのグリッド・ネーミング・サービス(GNS)機能は、Oracle Database 23cでは非推奨です。

日付: 2023年7月

GNS仮想IPアドレスは、Oracle Clusterwareに含まれるマルチキャスト・ドメイン・ネーム・サービス(mDNS)を使用してOracle Grid Infrastructureが解決する、DNSで構成された静的IPアドレスです。ただし、GNSのプライマリ・ロールまたはセカンダリ・ロールを指定するロールを使用するオプションと、共有GNS (マルチクラスタGNS)を使用するオプションは、どちらもOracle Database 23cのデプロイメント・オプションとして終了しました。これらの機能を使用する既存の構成は、アップグレー

ド後も機能しますが、新しいGNSデプロイメントでは使用できません。Oracleでは、独自のDNSを使用してOracle Grid Infrastructureへのクライアント接続を構成することをお薦めします。

親トピック: Oracle Database 23cで非推奨となった機能

従来の監査パッケージおよびファンクションの非推奨

従来の監査パッケージおよびファンクションは、Oracle Database 23cでは非推奨です。

日付: 2023年9月

従来の監査のサポート終了により、従来の監査に関連付けられたPL/SQLパッケージおよびファンクションは非推奨になります。この非推奨には、パッケージおよびファンクション`INIT_CLEANUP`、`DEINIT_CLEANUP`および`IS_CLEANUP_INITIALIZED`が含まれます。これらのパッケージまたはファンクションは引き続きOracle Database 23cで動作しますが、従来の監査構成に対する追加または変更はできません。

親トピック: Oracle Database 23cで非推奨となった機能

Oracle Database 23cで非推奨になったビュー

アップグレード計画の一環として、このOracle Databaseリリースから非推奨になったビューを確認してください。

- V$DATABASEビューのV$DATABASE.FS_FAILOVER列の非推奨
  Oracle DATABASE 23c以降では、`V$DATABASE`ビューのすべての`V$DATABASE.FS_FAILOVER`列が非推奨になりました。`V$FAST_START_FAILOVER_CONFIG`ビューを使用してください。
- V$PQ_SLAVEビューの非推奨
  Oracle Database 23cでは、`V$PQ_SLAVE`ビューが`V$PQ_SERVER`で置き換えられます。
- V$FS_FAILOVER_STATSビューの非推奨
  Oracle Database 23cでは、`V$FS_FAILOVER_STATS`ビューが非推奨になり、`V$DG_BROKER_ROLE_CHANGE`ビューで置き換えられます。
- DBA_HANG_MANAGER_PARAMETERSデータ・ディクショナリ・ビューの非推奨
  `DBA_HANG_MANAGER_PARAMETERS`静的データ・ディクショナリ・ビューは、Oracle Database 23cでは非推奨です。かわりに`DBA_BLOCKER_RESOLVER_PARAMETERS`を使用します。
- V$RECOVERY_SLAVEビューの非推奨
  Oracle Database 23cでは、`V$RECOVERY_SLAVE`ビューが`V$RECOVERY_WORKER`に置き換えられます。

親トピック: Oracle Database23cの動作の変更、サポート終了および非推奨

V$DATABASEビューのV$DATABASE.FS_FAILOVER列の非推奨

Oracle DATABASE 23c以降では、`V$DATABASE`ビューのすべての`V$DATABASE.FS_FAILOVER`列が非推奨になりました。`V$FAST_START_FAILOVER_CONFIG`ビューを使用してください。

日付: 2023年4月

Oracle Data Guardのファスト・スタート・フェイルオーバー機能の構成詳細を含む`V$FAST_START_FAILOVER_CONFIG`ビューが導入されています。その結果、`V$DATABASE`ビューの`FS_FAILOVER_*`列が非推奨になりました。かわりに、`V$FAST_START_FAILOVER_CONFIG`ビューの対応する列を使用してください。

列の置換は次のとおりです:

- `V$DATABASE.FS_FAILOVER_MODE`:
  `V$FAST_START_FAILOVER_CONFIG.FAST_START_FAILOVER_MODE`で置換

- V$DATABASE.FS_FAILOVER_STATUS: V$FAST_START_FAILOVER_CONFIG.STATUSで置換
- V$DATABASE.FS_FAILOVER_CURRENT_TARGET: V$FAST_START_FAILOVER_CONFIG.CURRENT_TARGETで置換
- V$DATABASE.FS_FAILOVER_THRESHOLD: V$FAST_START_FAILOVER_CONFIG.THRESHOLDで置換
- V$DATABASE.FS_FAILOVER_OBSERVER_PRESENT: V$FAST_START_FAILOVER_CONFIG.OBSERVER_PRESENTで置換
- V$DATABASE.FS_FAILOVER_OBSERVER_HOST: V$FAST_START_FAILOVER_CONFIG.OBSERVER_HOST で置換

親トピック: Oracle Database 23cで非推奨になったビュー

V$PQ_SLAVEビューの非推奨

Oracle Database 23cでは、V$PQ_SLAVEビューがV$PQ_SERVERで置き換えられます。

日付: 2023年4月

V$PQ_SLAVEのかわりにV$PQ_SERVERを使用してください。

親トピック: Oracle Database 23cで非推奨になったビュー

V$FS_FAILOVER_STATSビューの非推奨

Oracle Database 23cでは、V$FS_FAILOVER_STATSビューが非推奨になり、V$DG_BROKER_ROLE_CHANGEビューで置き換えられます。

日付: 2023年4月

V$FS_FAILOVER_STATSビューには、最後に発生したファスト・スタート・フェイルオーバーに関する限られた情報が1行表示されていました。データはプライマリ・データベースでのみ使用でき、データベースを再起動するとなくなっていました。新しいV$DG_BROKER_ROLE_CHANGEビューには、Data Guard Broker構成全体のスイッチオーバー、フェイルオーバーおよび回復を含む、最近10回のロール変更に関する詳細情報が含まれており、非推奨のV$FS_FAILOVER_STATSビューよりも詳細な情報が提供されます。

親トピック: Oracle Database 23cで非推奨になったビュー

DBA_HANG_MANAGER_PARAMETERSデータ・ディクショナリ・ビューの非推奨

DBA_HANG_MANAGER_PARAMETERS静的データ・ディクショナリ・ビューは、Oracle Database 23cでは非推奨です。かわりにDBA_BLOCKER_RESOLVER_PARAMETERSを使用します。

日付: 2023年7月

DBA_HANG_MANAGER_PARAMETERS静的データ・ディクショナリ・ビューには、使用可能なユーザーチューニング可能なDBMS_BLOCKER_RESOLVERパラメータおよび値が表示されます(以前はDBA_HANG_MANAGER_PARAMETERSとして使用可能でした)。このビューは、DBA_BLOCKER_RESOLVER_PARAMETERSに置き換えられます。DBA_HANG_MANAGER_PARAMETERSは、将来のリリースで削除される可能性があります。

親トピック: Oracle Database 23cで非推奨になったビュー

V$RECOVERY_SLAVEビューの非推奨

Oracle Database 23cでは、V$RECOVERY_SLAVEビューがV$RECOVERY_WORKERに置き換えられます。

日付: 2023年8月

V$RECOVERY_SLAVEのかわりにV$RECOVERY_WORKERを使用します。

親トピック: Oracle Database 23cで非推奨になったビュー

Oracle Database 23cで非推奨になったパラメータ

アップグレード計画の一環として、ここにリストされている、このOracle Databaseリリース以降で非推奨になる初期化パラメータを確認してください。

データベース・リリースのすべての非推奨パラメータの現在のリストを取得するには、SQL*Plusで次の問合せを実行します。

```
SQL> SELECT name from v$parameter
         WHERE isdeprecated = 'TRUE' ORDER BY name;
```

- ENCRYPT_NEW_TABLESPACESの非推奨
  Oracle Database 23c以降では、ENCRYPT_NEW_TABLESPACES初期化パラメータが非推奨になりました。
- ALLOW_MD5_CERTSおよびALLOW_SHA1_CERTS sqlnet.oraパラメータの非推奨
  Oracle Database 23c以降、ALLOW_MD5_CERTSおよびALLOW_SHA1_CERTS sqlnet.oraパラメータは非推奨です。
- MY_WALLET_DIRECTORY接続文字列の非推奨
  接続文字列パラメータMY_WALLET_DIRECTORYは、Oracle Database 23cで非推奨になりました。
- FIPSパラメータの非推奨
  Oracle Database 23c以降では、FIPS_140に関連付けられた複数のパラメータが非推奨になりました。
- ONE_STEP_PLUGIN_FOR_PDB_WITH_TDE初期化パラメータの非推奨
  パラメータONE_STEP_PLUGIN_FOR_PDB_WITH_TDEは、Oracle Database 23cでは非推奨です。
- 従来の監査初期化パラメータの非推奨
  Oracle Database 23で従来の監査ポリシーを作成および変更するサポートが終了すると、従来の監査パラメータは非推奨になります。
- PRE_PAGE_SGA初期化パラメータの非推奨
  Oracleでは、PRE_PAGE_SGAは廃止されているため非推奨です。通常、パラメータを設定すると、パフォーマンスの利点はほとんどまたはまったく得られず、問題が発生する可能性があります。

親トピック: Oracle Database23cの動作の変更、サポート終了および非推奨

ENCRYPT_NEW_TABLESPACESの非推奨

Oracle Database 23c以降では、ENCRYPT_NEW_TABLESPACES初期化パラメータが非推奨になりました。

日付: 2023年4月

Oracle Database 23cで導入された初期化パラメータTABLESPACE_ENCRYPTIONを使用することをお薦めします。

親トピック: Oracle Database 23cで非推奨になったパラメータ

ALLOW_MD5_CERTSおよびALLOW_SHA1_CERTS sqlnet.oraパラメータの非推奨

Oracle Database 23c以降、ALLOW_MD5_CERTSおよびALLOW_SHA1_CERTS sqlnet.oraパラメータは非推奨です。

日付: 2023年4月

これらのパラメータのかわりに、Oracle Database 23cで導入されたALLOWED_WEAK_CERT_ALGORITHMS sqlnet.oraパラメータを使用します。

親トピック: Oracle Database 23cで非推奨になったパラメータ

MY_WALLET_DIRECTORY接続文字列の非推奨

接続文字列パラメータMY_WALLET_DIRECTORYは、Oracle Database 23cで非推奨になりました。

日付: 2023年4月

接続文字列でWALLET_LOCATIONを使用して、sqlnet.ora WALLET_LOCATION設定をオーバーライドすることをお薦めします。sqlnet.oraおよびtnsnames.oraで同じパラメータを使用できるように、接続文字列のWALLET_LOCATIONが更新されました。この変更により、覚えておく必要があるパラメータが簡素化されます。MY_WALLET_DIRECTORYのかわりにWALLET_LOCATIONを使用するようにクライアント接続文字列を変更することをお薦めします。

親トピック: Oracle Database 23cで非推奨になったパラメータ

FIPSパラメータの非推奨

Oracle Database 23c以降では、FIPS_140に関連付けられた複数のパラメータが非推奨になりました。

日付: 2023年4月

FIPS.ORAのFIPS_140を使用すると、Oracle Database 23c以降のすべての機能に対してFIPSを有効にできます。次のFIPSパラメータが非推奨になりました:

- SQLNET.ORA: ネイティブ・ネットワーク暗号化のためのFIPSを有効にするFIPS_140
- FIPS.ORA: TLS用のFIPSを有効にするSSLFIPS_140
- 初期化パラメータ: TDEおよびDBMS_CRYPTO用のFIPSを有効にするDBFIPS_140

親トピック: Oracle Database 23cで非推奨になったパラメータ

ONE_STEP_PLUGIN_FOR_PDB_WITH_TDE初期化パラメータの非推奨

パラメータONE_STEP_PLUGIN_FOR_PDB_WITH_TDEは、Oracle Database 23cでは非推奨です。

日付: 2023年7月

ONE_STEP_PLUGIN_FOR_PDB_WITH_TDEを使用すると、Transparent Data Encryption (TDE)キーストアのパスワードを指定せずに、リモートでクローニングまたは暗号化されたPDBを再配置できます。ただし、EXTERNAL STOREは同じ機能を提供し、TDE構成を変更しないすべてのADMINISTER KEY MANAGEMENT文に広く適用できます。Oracleでは、ONE_STEP_PLUGIN_FOR_PDB_WITH_TDEを使用するかわりに、ADMINISTER KEY MANAGEMENT文にIDENTIFIED BY EXTERNAL STORE句を使用することをお薦めします。

親トピック: Oracle Database 23cで非推奨になったパラメータ

従来の監査初期化パラメータの非推奨

Oracle Database 23での従来の監査ポリシーの作成および変更のサポートが終了したことに伴い、従来の監査パラメータは非推奨です。

日付: 2023年7月

従来の監査ポリシーはアップグレード後に使用可能ですが、それらのポリシーに関連付けられた初期化パラメータは非推奨です。

これらの非推奨のパラメータには、次のものが含まれます。

- AUDIT_TRAIL
- AUDIT_SYS_OPERATIONS
- AUDIT_FILE_DEST
- AUDIT_SYSLOG_LEVEL

Oracleでは、これらの従来の監査パラメータは将来のデータベース・リリースで削除できるため、早急に統合監査に移行することをお薦めします。

親トピック: Oracle Database 23cで非推奨になったパラメータ

PRE_PAGE_SGA初期化パラメータの非推奨

Oracleでは、PRE_PAGE_SGAは廃止されているため非推奨です。通常、パラメータを設定すると、パフォーマンスの利点はほとんどまたはまったく得られず、問題が発生する可能性があります。

日付: 2023年8月

SGAパッケージ化のためのOracle Database設計は時間の経過とともに進化しているため、SGAに対するプロセスの起動効果はインスタンスの起動後に開始され、PRE_PAGE_SGAの値を変更することにはほとんどメリットがありません。このパラメータの唯一のユースケースは、PRE_PAGE_SGAをTRUEに設定するOracle Exadataシステムにあります。

親トピック: Oracle Database 23cで非推奨になったパラメータ

Oracle Database 21cの動作の変更、サポート終了および非推奨

Oracle Database 21cリリースの変更の説明を確認してください。

- Oracle Database 21cアップグレード計画の動作の変更
  Oracle Database 21cへのアップグレードの計画に役立つ動作の変更を確認してください
- Oracle Database 21cでサポートが終了した機能
  アップグレード計画の一環として、このOracle Databaseリリースでサポートが終了した機能を確認してください。
- Oracle Database 21cでサポートが終了したビュー
  このリリースでのビュー設定の変更点および代替機能については、サポートが終了したビューのリストを確認してください。
- Oracle Database 21cでサポートが終了した初期化パラメータ
  このリリースのパラメータ設定の変更および代替機能については、サポートが終了した初期化パラメータのリストを確認してください。
- Oracle Database 21cで非推奨となった機能
  アップグレード計画の一環として、このOracle Databaseリリースで非推奨となった機能を確認し、アプリケーション戦略に対する代替方法を確認してください。
- Oracle Database 21cで非推奨になったビュー
  アップグレード計画の一環として、このOracle Databaseリリースから非推奨になったビューを確認してください。
- Oracle Database 21cで非推奨になったパラメータ
  アップグレード計画の一環として、このOracle Databaseリリースから非推奨になった初期化パラメータを確認してください。

親トピック: Oracle Databaseの変更、サポート終了および非推奨

Oracle Database 21cアップグレード計画の動作の変更

Oracle Database 21cへのアップグレードの計画に役立つ動作の変更を確認してください

動作の変更には、パラメータまたは機能のデフォルトの変更、製品名の変更、および注意が必要なデータベース構成に対するその他の変更が含まれる場合があります。

- Oracle Database 21cの読取り専用のOracleホームについて
  Oracle Database 21c以降、Oracle Databaseのインストールでは、デフォルトですべてのOracle Databaseホームが読取り専用モードで構成されます。
- Oracle Database 21cでのみのマルチテナントのアップグレード
  Oracle Database 21c以降、Oracle Databaseはマルチテナント・アーキテクチャを使用した場合のみサポートされます。
- ロジカル・スタンバイと新しいデータ型
  ロジカル・スタンバイを使用する場合(DBMS_ROLLINGの一部として使用されていない場合)、使用できるのは、Oracle Database 12cリリース2 (12.2)より前に追加されたデータ型のみです
- HRサンプル・スキーマの再配置
  Oracle Database 21c以降、HRサンプル・スキーマはOracle Databaseの一部として出荷されなくなりました。
- PDBでのDRCPの管理
  Oracle Database 21c以降、データベース常駐接続プーリング(DRCP)は個々のPDBに対して構成および管理できます。
- Oracle Advanced Cluster File System (Oracle ACFS)の名前の変更
  Oracle Database 21c以降、Oracle Automatic Storage Management Cluster File System (Oracle ACFS)の名前は、Oracle Advanced Cluster File System (Oracle ACFS)に変更されています。
- Windows認証でNTLMがデフォルトで使用されなくなりました
  AUTHENTICATION_SERVICES=NTSを使用したMicrosoft Windowsインストールでは、このOracle Databaseリリースでsqlnet.oraファイルのSQLNET.NO_NTLMパラメータの設定はデフォルトでTRUEになり、ORA-12638エラーが発生する可能性があります。

親トピック: Oracle Database21cの動作の変更、サポート終了および非推奨

Oracle Database 21cの読取り専用Oracle Homesについて

Oracle Database 21c以降、Oracle Databaseのインストールでは、デフォルトですべてのOracle Databaseホームが読取り専用モードで構成されます。

> ノート:
>
> 読取り専用の Oracle ホーム・インストールのこの説明では、Oracle Database 21c のデフォルトについて説明しており、使用可能な構成は読取り専用 Oracle ホームのみであり、ソフトウェアのみの Oracle Database インストールはオプションではありませんでした。

読取り専用のOracleホームになることで、インストールと構成が別々に実装され、それによりプロビジョニングが簡素化されます。

Oracle Database 21cより前のデフォルトのORACLE_HOMEレイアウトでは、ORACLE_HOME、ORACLE_BASE_HOMEおよびORACLE_BASE_CONFIGが1つの場所にまとめられていました。Oracle Database 21c以降、使用可能な構成は読取り専用のORACLE_HOMEであり、ORACLE_BASE_HOMEとORACLE_BASE_CONFIGはORACLE_HOMEとは別に配置されています。

Oracleホームが読取り専用の場合、すべての構成データおよびログ・ファイルがその読取り専用Oracleホーム以外の場所に格納されます。

従来のORACLE_BASEおよびORACLE_HOMEディレクトリとは別に、ORACLE_HOMEに格納されていたファイルは次のディレクトリに格納されます。

- ORACLE_BASE_HOME
- ORACLE_BASE_CONFIG

ノート:

この機能は、データベース管理者によるシステム・パフォーマンスの監視、診断および調整の方法に影響を及ぼすものではありません。

親トピック: Oracle Database 21cアップグレード計画の動作の変更

Oracle Database 21cでのみのマルチテナントのアップグレード

Oracle Database 21 c以降、Oracle Databaseはマルチテナント・アーキテクチャを使用した場合のみサポートされます。

ノート:

マルチテナント・コンテナ・データベースが、Oracle Database 21c 以降のリリースで唯一サポートされているアーキテクチャです。ドキュメントが改訂されている間は、従来の用語が残っている可能性があります。ほとんどの場合、「データベース」と「非 CDB」は、コンテキストに応じて CDB または PDB を指しています。アップグレードなどのコンテキストでは、「非 CDB」が以前のリリースの非 CDB を指す場合もあります。

親トピック: Oracle Database 21cアップグレード計画の動作の変更

ロジカル・スタンバイと新しいデータ型

ロジカル・スタンバイを使用する場合(DBMS_ROLLINGの一部として使用されていない場合)、使用できるのは、Oracle Database 12cリリース2 (12.2)より前に追加されたデータ型のみです

日付: 2021年5月

Oracle Database 12cリリース1 (12.1)以降に追加された新しいデータ型は、Oracle Data Guardロジカル・スタンバイではサポートされません。たとえば、Oracle Data Guardロジカル・スタンバイは、長い識別子、複雑な抽象データ型(ADT)および空間データ型をサポートしていませんこの制限は、Oracle Data Guardフィジカル・スタンバイ・データベース、DBMS_ROLLINGまたはOracle GoldenGateには存在しないことに注意してください。より新しいデータ型のスタンバイ・データベースの利点を得るために、フィジカル・スタンバイ・データベース、スナップショット・スタンバイ・データベースのいずれかの使用を検討するか、またはOracle GoldenGateの論理レプリケーション機能の使用を検討することをお薦めします。

親トピック: Oracle Database 21cアップグレード計画の動作の変更

HRサンプル・スキーマの再配置

Oracle Database 21c以降、HRサンプル・スキーマはOracle Databaseの一部として出荷されなくなりました。

HRスキーマをインストールするには、*Oracle Database*データベース・サンプル・スキーマのサンプル・スキーマのインストールに関する項を参照してください

関連項目

- サンプル・スキーマのインストール

親トピック: Oracle Database 21cアップグレード計画の動作の変更

PDBでのDRCPの管理

Oracle Database 21c以降、データベース常駐接続プーリング(DRCP)は個々のPDBに対して構成および管理できます。

DRCPが個々のPDBから管理されるように構成する場合、個々のプラガブル・データベース(PDB)でプールを構成、管理および監視できます。

以前のリリースでは、DRCPプールはコンテナ・データベース(CDB)全体で使用されていました。この機能は、DRCPプールの粒度をCDB全体から個々のPDBのDRCPプールに変更することでDRCPプールの管理を容易にします。この変更により、テナント管理者は独立したテナント固有のDRCPプールを構成および管理できます。

この機能は、新しいデータベース・パラメータENABLE_PER_PDB_DRCPによって有効になります。デフォルトでは、パラメータはENABLE_PER_PDB_DRCP=FALSEに設定されています。ENABLE_PER_PDB_DRCP=TRUEを設定すると、この機能が有効になります。

PDBをOracle Database 21cにアップグレードすると、PDBのデフォルトの接続プールがPDBに作成されます。その後、初期化パラメータを使用してこの機能を有効にすると、CDBで接続プーリングを管理するかわりに、接続プールはPDBによってローカルで管理されます。

親トピック: Oracle Database 21cアップグレード計画の動作の変更

Oracle Advanced Cluster File System (Oracle ACFS)の名前の変更

Oracle Database 21c以降、Oracle Automatic Storage Management Cluster File System (Oracle ACFS)の名前は、Oracle Advanced Cluster File System (Oracle ACFS)に変更されています。

この変更は名前の変更のみです。Oracleのクラスタ・ファイル・システムの基本機能は引き続き同じです。オラクル社は、Oracle ACFSの開発および強化を続けていきます。

親トピック: Oracle Database 21cアップグレード計画の動作の変更

Windows認証でNTLMがデフォルトで使用されなくなった

AUTHENTICATION_SERVICES=NTSを使用したMicrosoft Windowsインストールでは、このOracle Databaseリリースでsqlnet.oraファイルのSQLNET.NO_NTLMパラメータの設定がデフォルトでTRUEになり、ORA-12638エラーが発生する可能性があります。

以前のリリースでは、AUTHENTICATION_SERVICES=NTSのデフォルトはFALSEでした。SQLNET.NO_NTLMは、NTS認証でNTLMを使用できるかどうかを制御します。TRUE設定が、NTS認証でNTLMを使用できないことを意味します。NTLMは、通常、相互認証を提供せず安全性が低くなるため、SQLNET.NO_NTLMにTRUEを設定すると、データベースとクライアントがよりセキュアになります。

SQLNET.NO_NTLMパラメータは、サーバーとクライアントの両方で使用されます。SQLNET.NO_NTLMが設定されていないOracleデータベースまたはクライアントのMicrosoft Windowsインストールをアップグレードした場合、デフォルトはTRUEになります。この場合、sqlnet.oraにSQLNET.AUTHENTICATION_SERVICES=NTSがあると、クライアントでエラーORA-12638: Credential retrieval failedが発生する可能性があります。

特定のクライアントに対してNTLM認証を使用する場合は、クライアント側のsqlnet.oraファイルで必要に応じてこのパラメータを設定します。

```
SQLNET.NO_NTLM=FALSE
```

この設定はサーバーとクライアントの両方に含める必要があり、両方で同じ設定になっている必要があります。SQLNET.NO_NTLMがTRUEに設定されていることを確認することをお薦めします。ただし、extproc、仮想アカウントまたはWindowsのローカル・アカウントで認証に失敗した場合は、クライアントSQLNET.NO_NTLMをFALSEに設定してからログインを再試行してください。サーバーのSQLNET.NO_NTLMを変更した場合は、データベースを再起動する必要があります。

関連項目

- Oracle Database Microsoft Windowsインストールの認証を保護するためのガイドライン
- 今すぐデータベースをアップグレード - ORA-12638 (Windowsのみ)は、Oracle 19.10.0以降
- Windows：2021年1月のWINDBBP 19.10.0.0.210119をWindows 64/32ビット・データベース・クライアントに適用した後、データベースに接続してORA-12638を取得している間(ドキュメントID 2757734.1)

親トピック: Oracle Database 21cアップグレード計画の動作の変更

Oracle Database 21cでサポートが終了した機能

アップグレード計画の一環として、このOracle Databaseリリースでサポートが終了した機能を確認してください。

- DBMS_OBFUSCATION_TOOLKITパッケージのサポート終了
  Oracle Database 21c以降、パッケージDBMS_OBFUSCATION_TOOLKITはサポート対象外になり、DBMS_CRYPTOに置き換えられました。
- 複数のXML Database (XDB)機能のサポート終了
  Oracle Database 21c以降、複数のXML Database機能はサポートされなくなりました。
- DBMS_LOB.LOADFROMFILEおよびLOBバッファリングのサポート終了
  Oracle Database 21c以降、ラージ・オブジェクト(LOB)機能のDBMS_LOB.LOADFROMFILEおよびLOBバッファリングはサポートされなくなりました。
- Oracle Data Guard Brokerプロパティおよびロジカル・スタンバイのサポート終了
  Oracle Data Guard Brokerプロパティおよびロジカル・スタンバイ・プロパティはサポートされなくなりました。これらの機能は、DGMGRLの新しいEDIT … SET PARAMETERコマンドに置き換えられます。
- DBMS_CRYPTO_TOOLKIT_TYPESおよびDBMS_CRYPTO_TOOLKITのサポート終了
  Oracle Database 21c以降、データ型DBMS_CRYPTO_TOOLKIT_TYPESおよびパッケージDBMS_CRYPTO_TOOLKITはサポートされなくなりました。
- 非CDB Oracle Databasesのサポート終了
  Oracle Database 21c以降、非CDB Oracle Databaseアーキテクチャのインストールはサポートされなくなりました。
- クラスタ・ドメインのメンバー・クラスタのサポート終了
  Oracle Grid Infrastructure 21c以降、Oracleクラスタ・ドメイン・アーキテクチャに含まれるメンバー・クラスタはサポートされなくなりました。
- Unicode照合アルゴリズム(UCA) 6.1照合のサポート終了
  Oracle Database 21c以降、Unicode照合アルゴリズム(UCA) 6.1照合(UCA0610_*)はサポートされなくなりました。かわりにUCA 12.1を使用します。
- Microsoft WindowsでのACFSのサポート終了
  Oracle Database 21c以降、Microsoft WindowsでのOracle Grid Infrastructure機能のAdvanced Storage Management Cluster File System (Oracle ACFS)はサポートされなくなりました
- Oracle ACFSセキュリティ(Vault)およびACFS監査のサポート終了
  Oracle Grid Infrastructure 21c以降、Oracle Advanced Cluster File System (ACFS)セキュリティ(Vault)およびACFS監査はサポートされなくなりました。

- メンバー・クラスタでのOracle ACFSの使用(ACFSリモート)のサポート終了
  Oracle Grid Infrastructure 21c以降、メンバー・クラスタでのOracle Advanced Cluster File System (ACFS)の使用(ACFSリモート)はサポートされなくなりました。
- SolarisおよびWindowsでのACFS暗号化のサポート終了
  Oracle Database 21c以降、Oracle SolarisおよびMicrosoft Windowsで代替機能がないため、Oracle ACFS暗号化はサポートされなくなりました。
- ACFSレプリケーションREPV1のサポート終了
  Oracle Database 21c以降、Oracle ACFSレプリケーション・プロトコルrepv1はサポートされなくなりました。
- Oracle Clusterwareとベンダー・クラスタウェアの統合のサポート終了
  Oracle Clusterware 21c以降では、Oracle Clusterwareとベンダーまたはサード・パーティのクラスタウェアとの統合はサポートされなくなりました。
- VERIFY_FUNCTIONおよびVERIFY_FUNCTION_11Gのサポート終了
  `VERIFY_FUNCTION`および`VERIFY_FUNCTION_11G`パスワード検証関数は、Oracle Database 21cでサポートが終了しました。
- 非推奨となったOracle Database Vaultロールのサポート終了
  Oracle Database Vaultロール`DV_PUBLIC`、`DV_REALM_OWNER`および`DV_REALM_RESOURCE`は、Oracle Database 21cでサポートが終了しました。
- 匿名RC4暗号スイートのサポート終了
  非認証TLS接続での匿名RC4暗号スイートの使用は、Oracle Database 21c (`SSL_DH_anon_WITH_RC4_128_MD5`)でサポートが終了しました
- Adobe FlashベースのOracle Enterprise Manager Expressのサポート終了
  FlashベースのOracle EM Expressは、Oracle 21cでサポートが終了しました。JETベースのOracle EM Express (デフォルト)を使用します。
- インテリジェント・データ配置(IDP)のサポート終了
  インテリジェント・データ配置(IDP)は、Oracle Database 21cでサポートが終了しました。
- XML DB Content Connectorのサポート終了
  Oracle XML DB Content Connectorは、Oracle Database 12cリリース2で非推奨になりました。Oracle Database 21cでサポートが終了し、削除されました。
- DBMS_XMLSAVEのサポート終了
  PL/SQLパッケージ`DBMS_XMLSAVE`は、Oracle Database 21cでサポートが終了しました。かわりに`DBMS_XMLSTORE`を使用してください。
- DBMS_XMLQUERYのサポート終了
  PL/SQLパッケージ`DBMS_XMLQUERY`は、Oracle Database 21cでサポートが終了しました。かわりに`DBMS_XMLGEN`を使用してください。
- FIPS保護およびプロセス強度0のサポート終了
  FIPSでは、保護およびプロセス強度0 (RSAキー長512)相当はサポートが終了しました。この強度はFIPS以外のモードでも引き続き使用できます。
- PDBフラット・ファイル・ディクショナリ・ダンプのサポート終了
  プラガブル・データベース(PDB)のフラット・ファイル・ディクショナリ・ダンプを作成する機能は、Oracle Database 21cでサポートが終了しました。
- Oracle Fail Safeのサポート終了
  Oracle Database 21c以降、Oracle Fail SafeはOracle Databaseリリースでサポートされなくなりました。

親トピック: Oracle Database21cの動作の変更、サポート終了および非推奨

DBMS_OBFUSCATION_TOOLKITパッケージのサポート終了

Oracle Database 21c以降、パッケージDBMS_OBFUSCATION_TOOLKITはサポート対象外になり、DBMS_CRYPTOに置き換えられました。

日付: 2021年5月

DBMS_OBFUSCATION_TOOLKITはOracle Database 10gリリース2で非推奨になりました。また、Oracle Database 21cで削除されました。DBMS_OBFUSCATION_TOOLKITによって提供されていた機能は、DBMS_CRYPTOに置き換えられます。DBMS_CRYPTOには、セキュリティ要件に対応する最新かつセキュアな暗号化技術が組み込まれています。

親トピック: Oracle Database 21cでサポートが終了した機能

複数のXML Database (XDB)機能のサポート終了

Oracle Database 21c以降、XML Databaseのいくつかの機能がサポートされなくなりました。

日付: 2021年5月

次の機能のサポートが終了します。

- パッケージDBMS_XDBT。これに替わる機能はありません。
- Oracle XQuery関数ora:contains。かわりにXQuery Full Textを使用してください。
- Oracle SQL関数XMLRoot。かわりに、バージョン番号付きのSQL/XML関数XMLSerialize()を使用してください。
- 索引構成表(IOT)として格納されたネストした表。これには、ネストした表の列Nを持つ表を作成するときに、オプションDBMS_XMLSCHEMA.REGISTER_NT_AS_IOTと句NESTED TABLE N STORE AS ... (ORGANIZATION INDEX)の両方の使用が含まれます。かわりに、ネストした表の列はヒープ・ストレージを使用して格納してください(PL/SQLプロシージャDBMS_XMLSCHEMA.registerSchemaのデフォルトの動作)。
- PL/SQLプロシージャDBMS_XSLPROCESSOR.CLOB2FILE。かわりにDBMS_LOB.CLOB2FILEを使用してください。
- PL/SQL関数DBMS_XSLPROCESSOR.READ2CLOB。かわりにDBMS_LOB.LOADCLOBFROMFILEを使用してください。
- Oracle XML DB Content Connector。

親トピック: Oracle Database 21cでサポートが終了した機能

DBMS_LOB.LOADFROMFILEおよびLOBバッファリングのサポート終了

Oracle Database 21c以降、ラージ・オブジェクト(LOB)機能のDBMS_LOB.LOADFROMFILEおよびLOBバッファリングはサポートされなくなりました。

日付: 2021年5月

次の機能のサポートが終了します。

- DBMS_LOB.LOADFROMFILEプロシージャ。かわりに、DBMS_LOB.LoadClobFromFileまたはDBMS_LOB.LoadBlobFromFileを使用してください。
- LOBバッファリング・サブシステムAPI:
  - OCILobEnableBuffering()
  - OCILobDisableBuffering()
  - OCILobFlushBuffer()

これらのLOBバッファリング関数を使用するかわりに、LOBプリフェッチ機能を使用してください。

親トピック: Oracle Database 21cでサポートが終了した機能

Oracle Data Guard Brokerプロパティおよびロジカル・スタンバイのサポート終了

Oracle Data Guard Brokerプロパティおよびロジカル・スタンバイ・プロパティはサポートされなくなりました。これらの機能は、DGMGRLの新しいEDIT … SET PARAMETERコマンドに置き換えられます。

日付: 2021年5月

次のOracle Data Guard BrokerプロパティはOracle Database 21cでサポートされなくなりました。

- `ArchiveLagTarget`
- `DataGuardSyncLatency`
- `LogArchiveMaxProcesses`
- `LogArchiveMinSucceedDest`
- `LogArchiveTrace`
- `StandbyFileManagement`
- `DbFileNameConvert`
- `LogArchiveFormat`
- `LogFileNameConvert`

ロジカル・スタンバイに影響を与える次のOracle Data Guard Brokerプロパティは、Oracle Database 21cでサポートされなくなりました。

- `LsbyMaxEventsRecorded`
- `LsbyMaxServers`
- `LsbyMaxSga`
- `LsbyPreserveCommitOrder`
- `LsbyRecordAppliedDdl`
- `LsbyRecordSkippedDdl`
- `LsbyRecordSkipErrors`
- `LsbyParameter`

親トピック: Oracle Database 21cでサポートが終了した機能

DBMS_CRYPTO_TOOLKIT_TYPESおよびDBMS_CRYPTO_TOOLKITのサポート終了

Oracle Database 21c以降、データ型`DBMS_CRYPTO_TOOLKIT_TYPES`およびパッケージ`DBMS_CRYPTO_TOOLKIT`はサポートされなくなりました。

日付: 2021年5月

データ型`DBMS_CRYPTO_TOOLKIT_TYPES`および`DBMS_CRYPTO_TOOLKIT`パッケージはOracle9i Databaseで非推奨になりました。これらのデータ型およびパッケージはOracle Database 21cから削除されました。

親トピック: Oracle Database 21cでサポートが終了した機能

非CDB Oracle Databaseのサポート終了

Oracle Database 21c以降、非CDB Oracle Databaseアーキテクチャのインストールはサポートされなくなりました。

日付: 2021年5月

非CDBアーキテクチャはOracle Database 12cで非推奨になりました。Oracle Database 21cではサポートが終了しました。Oracle Universal Installerを非CDB Oracle Databaseインスタンスの作成に使用することはできなくなりました。

親トピック: Oracle Database 21cでサポートが終了した機能

クラスタ・ドメインのメンバー・クラスタのサポート終了

Oracle Grid Infrastructure 21c以降、Oracleクラスタ・ドメイン・アーキテクチャに含まれるメンバー・クラスタはサポートされなくなりました。

日付: 2021年5月

特定のクラスタ機能のサポートを終了し、制限付きの採用とすることにより、オラクル社では、すべての機能の中核となるスケーリング、可用性および管理性の向上に注力できます。Oracleクラスタ・ドメインは、ドメイン・サービス・クラスタ(DSC)とメンバー・クラスタで構成されます。これには、リモートOracle Advanced Storage Management Cluster File System (リモートACFS)も含まれます。まずメンバー・クラスタが導入され、大規模なクラスタの管理が簡素化され、特定の障害および構成の停止時間が最小限に抑えられました。ただし、スタンドアロン・クラスタに追加された拡張機能により同様のメリットが得られます。これらの拡張機能では、メンバー・クラスタを使用する必要がありません。そのため、現在メンバー・クラスタを使用している場合は、スタンドアロン・クラスタを使用することをお薦めします。

親トピック: Oracle Database 21cでサポートが終了した機能

Unicode照合アルゴリズム(UCA) 6.1照合のサポート終了

Oracle Database 21c以降、Unicode照合アルゴリズム(UCA) 6.1照合(`UCA0610_*`)はサポートされなくなりました。かわりにUCA 12.1を使用します。

日付: 2021年5月

サポートされている最新バージョンのUnicode照合アルゴリズム(UCA)照合(Oracle Database 21cではUCA 12.1)を使用することをお薦めします。UCA 6.1照合は、Oracle Database 12cリリース2で非推奨になりました。UCA 12.1にはバージョン6.1以降のすべてのUCA拡張機能、およびUnicode 6.1以降に導入されたすべての新しい文字に対する適切な照合加重割当が組み込まれています。

親トピック: Oracle Database 21cでサポートが終了した機能

Microsoft WindowsでのACFSの使用のサポート終了

Oracle Database 21c以降、Microsoft WindowsでのOracle Grid Infrastructure機能のAdvanced Storage Management Cluster File System (Oracle ACFS)はサポートされなくなりました

日付: 2021年5月

Oracle ACFSは主に、次の2つのユースケースで使用されます。

- Oracle Real Application Clusters (Oracle RAC)のOracle Databaseファイル
- 複数のホスト間で共有する必要のある汎用ファイル(非構造化データ)。

Oracle Real Application Clustersファイルでは、Oracle ASMを使用することをお薦めします。汎用ファイルでは、ユースケースに応じて、Oracle Databaseファイル・システム(DBFS)にファイルを移動するか、Microsoft Windowsの共有ファイルにファイルを移動することをお薦めします。

親トピック: Oracle Database 21cでサポートが終了した機能

Oracle ACFSセキュリティ(Vault)およびACFS監査のサポート終了

Oracle Grid Infrastructure 21c以降、Oracle Advanced Cluster File System (ACFS)セキュリティ(Vault)およびACFS監査はサポートされなくなりました。

日付: 2021年5月

セキュリティおよび監査を管理する場合は、オペレーティング・システムのアクセス制御および監査システムを使用することをお薦めします。たとえば、LinuxではLinux監査システムを使用できます。このサポート終了の一環として、次のビューが削除されます。

```
V$ASM_ACFS_SEC_ADMIN
V$ASM_ACFS_SEC_CMDRULE
V$ASM_ACFS_SEC_REALM
V$ASM_ACFS_SEC_REALM_FILTER
V$ASM_ACFS_SEC_REALM_GROUP
V$ASM_ACFS_SEC_REALM_USER
V$ASM_ACFS_SEC_RULE
V$ASM_ACFS_SEC_RULESET
V$ASM_ACFS_SEC_RULESET_RULE
V$ASM_ACFS_SECURITY_INFO
```

親トピック: Oracle Database 21cでサポートが終了した機能

メンバー・クラスタでのOracle ACFSの使用(ACFSリモート)のサポート終了

Oracle Grid Infrastructure 21c以降、メンバー・クラスタでのOracle Advanced Cluster File System (ACFS)の使用(ACFSリモート)はサポートされなくなりました。

日付: 2021年5月

メンバー・クラスタでのOracle Advanced Cluster File System (ACFS)の使用(ACFSリモート)はサポートされなくなりました。特定のクラスタ機能のサポートを終了し、制限付きの採用とすることにより、オラクル社では、すべての機能の中核となるスケーリング、可用性および管理性の向上に注力できます。特定のクラスタ機能を非推奨にし、制限付きの採用とすることにより、オラクル社では、すべての機能の中核となるスケーリング、可用性および管理性の向上に注力できます。

親トピック: Oracle Database 21cでサポートが終了した機能

SolarisおよびWindowsでのACFS暗号化のサポート終了

Oracle Database 21c以降、Oracle SolarisおよびMicrosoft Windowsで代替機能がないため、Oracle ACFS暗号化はサポートされなくなりました。

日付: 2021年5月

Oracle SolarisおよびMicrosoft WindowsでのOracle ACFS暗号化は、RSAテクノロジに基づいています。RSAテクノロジのリタイアが通知されました。Linuxでは代替テクノロジを使用するため、Oracle ACFS暗号化はLinuxで引き続きサポートされ、この非推奨に影響を受けません。

親トピック: Oracle Database 21cでサポートが終了した機能

ACFSレプリケーションREPV1のサポート終了

Oracle Database 21c以降、Oracle ACFSレプリケーション・プロトコル`repv1`はサポートされなくなりました。

日付: 2021年5月

初期ACFSレプリケーション・プロトコル`repv1`は、Oracle Database 11gリリース2 (11.2)でリリースされました。Oracle Database 12cリリース2 (12.2)以降、`replv2`プロトコルが必要でした。`replv1`プロトコルは、12.2以降のリリースで必要な`replv2`へのアップグレード時にのみ使用可能でした。`replv1`プロトコルは、Oracle Database 19cで非推奨になりました。Oracle Database 21cではサポートが終了しました。

親トピック: Oracle Database 21cでサポートが終了した機能

Oracle Clusterwareとベンダー・クラスタウェアの統合のサポート終了

Oracle Clusterware 21c以降では、Oracle Clusterwareとベンダーまたはサード・パーティのクラスタウェアとの統合はサポートされなくなりました。

日付: 2021年5月

Oracle Clusterwareとベンダー・クラスタウェアの統合は、Oracle Database 21cでサポートされなくなりました。特定のクラスタ機能のサポートを終了し、制限付きの採用とすることにより、オラクル社では、すべての機能の中核となるスケーリング、可用性および管理性の向上に注力できます。異なるクラスタ・ソリューション間の統合がない場合、システムでクラスタ・ソリューションの競合の問題が生じることがあります。独立した各クラスタ・ソリューションでは、特定の障害が発生した場合に実行する必要がある修正処理が個別に決定される可能性があります。競合を回避するには、いつの時点においても1つのクラスタ・ソリューションのみがアクティブになるようにしてください。このため、次回のソフトウェアまたはハードウェアのアップグレードで、ベンダー・クラスタ・ソリューションを使用しない構成に移行することをお薦めします。

親トピック: Oracle Database 21cでサポートが終了した機能

VERIFY_FUNCTIONおよびVERIFY_FUNCTION_11Gのサポート終了

`VERIFY_FUNCTION`および`VERIFY_FUNCTION_11G`パスワード検証関数は、Oracle Database 21cでサポートが終了しました。

日付: 2021年5月

これらの古い関数は、以前のリリースからの脆弱なパスワード制限を強制するため、サポートされなくなりました。かわりに、`ORA12C_VERIFY_FUNCTION`、`ORA12C_STRONG_VERIFY_FUNCTION`または`ORA12C_STIG_VERIFY_FUNCTIONS`関数を使用します。これらの関数は、より強力でより最新のパスワード検証制限を強化しています。

親トピック: Oracle Database 21cでサポートが終了した機能

非推奨となったOracle Database Vaultロールのサポート終了

Oracle Database Vaultロール`DV_PUBLIC`、`DV_REALM_OWNER`および`DV_REALM_RESOURCE`は、Oracle Database 21cでサポートが終了しました。

日付: 2021年5月

Oracleでは、これらのOracle Database VaultロールはOracle Database 19cで非推奨になりました。ロールには強力な権限が付与されていますが、ほとんど使用されませんでした。Oracle Database 21cへのアップグレード中に、これらのロールは削除され、Oracle Databaseの新規インストールでは使用できません。これらのロールを使用していて、データベースをアップグレードする場合は、データベースに付与されているロールおよび権限を書き留め、アップグレードの完了後にこれらの権限のロールを作成します。

親トピック: Oracle Database 21cでサポートが終了した機能

匿名RC4暗号スイートのサポート終了

認証されていないTLS接続に対する匿名RC4暗号スイートの使用は、Oracle Database 21c (`SSL_DH_anon_WITH_RC4_128_MD5`)でサポートが終了しました

日付: 2021年5月

Oracle Databaseで使用可能なよりセキュアな認証済接続を使用することをお薦めします。エンタープライズ・ユーザー・セキュ

リティのためにOracle Internet Directoryに接続するためにRC4で匿名Diffie-Hellmanを使用する場合は、別のアルゴリズム接続を使用するように移行する必要があります。TLS一方向または証明書を使用した相互認証のいずれかを使用することをお薦めします。

親トピック: Oracle Database 21cでサポートが終了した機能

Adobe FlashベースのOracle Enterprise Manager Expressのサポート終了

フラッシュベースのOracle EM Expressは、Oracle 21cでサポートが終了しました。JETベースのOracle EM Express (デフォルト)を使用します。

日付: 2021年5月

Adobe Flashは2021年1月以降のすべての主要ブラウザでサポートが終了しました。Oracle Database 19c以降のリリースのOracle Enterprise Manager Express (Oracle EM Express)では、FlashのかわりにOracle JavaScript Extension Toolkit (JET)テクノロジを使用します。

親トピック: Oracle Database 21cでサポートが終了した機能

インテリジェント・データ配置(IDP)のサポート終了

インテリジェント・データ配置(IDP)は、Oracle Database 21cでサポートが終了しました。

日付: 2021年5月

インテリジェント・データ配置は、待機時間を短縮するために物理ストレージ・ディスクにデータを配置するのに役立ちました。この機能は、Oracle Database 12cリリース2 (12.2)で非推奨となりました。これに替わる機能はありません。この機能に関連付けられているビューおよびASMディスク・グループ属性もサポートされなくなりました。

親トピック: Oracle Database 21cでサポートが終了した機能

XML DB Content Connectorのサポート終了

Oracle XML DB Content Connectorは、Oracle Database 12cリリース2で非推奨になりました。Oracle Database 21cでサポートが終了し、削除されました。

日付: 2021年5月

XML DB Content Connectorは、Java標準JSR-170 APIを実装していました。この標準は、Java標準JSR-283に置き換えられています。

親トピック: Oracle Database 21cでサポートが終了した機能

DBMS_XMLSAVEのサポート終了

PL/SQLパッケージ`DBMS_XMLSAVE`は、Oracle Database 21cでサポートが終了しました。かわりに`DBMS_XMLSTORE`を使用してください。

日付: 2021年5月

`DBMS_XMLSAVE`コールは`DBMS_XMLSTORE`に置き換えます。

親トピック: Oracle Database 21cでサポートが終了した機能

DBMS_XMLQUERYのサポート終了

PL/SQLパッケージ`DBMS_XMLQUERY`は、Oracle Database 21cでサポートが終了しました。かわりに`DBMS_XMLGEN`を

使用してください。

日付: 2021年5月

DBMS_XMLQUERYへのコールはDBMS_XMLGENに置き換えます。

親トピック: Oracle Database 21cでサポートが終了した機能

FIPS保護およびプロセス強度0のサポート終了

FIPSでは、保護およびプロセス強度0 (RSAキー長512)相当はサポートされなくなりました。この強度はFIPS以外のモードでも引き続き使用できます。

日付: 2021年5月

Federal Information Processing Standard (FIPS)モードのデフォルトの保護およびプロセス強度は現在80です。同等のRSAキー強度は1024です。同等のECCキー強度は、ECC曲線キーの最小長160の曲線、ECC名前付き曲線P192、K163およびB163以上の曲線です。同等のDH/DSA (Diffie Hellman、デジタル署名アルゴリズム)キーの長さは1024です。デフォルトのFIPS保護強度80は非推奨であるため、FIPS保護強度112を使用することをお薦めします。

親トピック: Oracle Database 21cでサポートが終了した機能

PDBフラット・ファイル・ディクショナリ・ダンプのサポート終了

プラガブル・データベース(PDB)のフラット・ファイル・ディクショナリ・ダンプを作成する機能は、Oracle Database 21cでサポートが終了しました。

日付: 2021年5月

以前のリリースでは、フラット・ファイル・ディクショナリの使用は、データ・ディクショナリがフラット・ファイル内に含まれていた特定のPDBに関連付けられた変更のREDOログをマイニングする手段でした。この機能はサポートされなくなりました。Oracle Database 21c以降、DBMS_LOGMNR.START_LOGMNRをコールし、マイニングするシステム変更番号(SCN)または時間範囲を指定することをお薦めします。START_LOGMNRのSCNまたは時間範囲オプションは、個々のPDBのマイニングをサポートするように拡張されています。

親トピック: Oracle Database 21cでサポートが終了した機能

Oracle Fail Safeのサポート終了

Oracle Database 21c以降、Oracle Fail SafeはOracle Databaseリリースでサポートされなくなりました。

日付: 2021年6月

このサポート終了の通知は、旧リリースのOracle Databaseには適用されません。Oracle Fail Safeは、Oracle Database 19cの存続期間中引き続きサポートされます。Oracle Database 21c以降のOracle Databaseリリースでは、Microsoft WindowsでOracle Databaseを実行している場合、Oracle RAC One NodeやOracle Database Standard Edition高可用性などの他のフェイルオーバー・ソリューションを調査することをお薦めします。

親トピック: Oracle Database 21cでサポートが終了した機能

Oracle Database 21cでサポートが終了したビュー

このリリースでのビュー設定の変更点および代替機能については、サポートが終了したビューのリストを確認してください。

- V$OBJECT_USAGEビューのサポート終了
  ビューV$OBJECT_USAGEは、Oracle Database 21cではサポートが終了しています。かわりに

USER_OBJECT_USAGEを使用してください。

親トピック: Oracle Database21cの動作の変更、サポート終了および非推奨

V$OBJECT_USAGEビューのサポート終了

ビューV$OBJECT_USAGEは、Oracle Database 21cではサポートが終了しています。かわりにUSER_OBJECT_USAGEを使用してください。

日付: 2021年10月

V$OBJECT_USAGEビューでは、現行ユーザーが所有する索引について、データベースから収集した索引使用状況に関する統計が示されていました。このビューは、Oracle Database 12cリリース1非推奨となりました。現在は、サポートが終了しています。ユーザーが所有する索引の使用状況について統計を取得するには、V$OBJECT_USAGEビューのかわりにUSER_OBJECT_USAGEビューを使用します。

親トピック: Oracle Database 21cでサポートが終了したビュー

Oracle Database 21cのサポートが終了した初期化パラメータ

パラメータ設定の変更および代替機能については、サポートが終了した初期化パラメータのこのリストを参照してください。

- UNIFIED_AUDIT_SGA_QUEUE_SIZEのサポート終了
  Oracle Database 21c以降、初期化パラメータ`UNIFIED_AUDIT_SGA_QUEUE_SIZE`はサポートされなくなりました。
- パスワードのIGNORECASEパラメータのサポート終了
  Oracle Database 21c以降、`orapwd`ファイルの`IGNORECASE`パラメータはサポートされなくなりました。新しく作成されるすべてのパスワード・ファイルでは、大/小文字が区別されます。
- DISABLE_DIRECTORY_LINK_CHECKのサポート終了
  Oracle Database 21c以降、`DISABLE_DIRECTORY_LINK_CHECK`パラメータはサポートされなくなり、代替はありません。
- REMOTE_OS_AUTHENTパラメータのサポート終了
  Oracle Database初期化パラメータ`REMOTE_OS_AUTHENT`は、Oracle Database 21cから削除されました。
- SEC_CASE_SENSITIVE_LOGONのサポート終了
  `SEC_CASE_SENSITIVE_LOGON`パラメータは、Oracle Database 21cでサポートが終了しました。
- CLUSTER_DATABASE_INSTANCESパラメータのサポート終了
  `CLUSTER_DATABASE_INSTANCES`パラメータは、21cではサポートが終了しています。これに替わる機能はありません。

親トピック: Oracle Database21cの動作の変更、サポート終了および非推奨

UNIFIED_AUDIT_SGA_QUEUE_SIZEのサポート終了

Oracle Database 21c以降、初期化パラメータ`UNIFIED_AUDIT_SGA_QUEUE_SIZE`はサポートされなくなりました。

2021年5月

`UNIFIED_AUDIT_SGA_QUEUE_SIZE`パラメータはOracle Database 12cリリース2 (12.2)で非推奨になり、パラメータの値は使用できなくなりました。これは削除されました。

親トピック: Oracle Database 21cでサポートが終了した初期化パラメータ

パスワードのIGNORECASEパラメータのサポート終了

Oracle Database 21c以降、orapwdファイルのIGNORECASEパラメータはサポートされなくなりました。新しく作成されるすべてのパスワード・ファイルでは、大/小文字が区別されます。

2021年5月

大/小文字が区別されるパスワード・ファイルは、大/小文字が区別されない古いパスワード・ファイルよりもセキュリティが向上します。セキュリティを強化するために、大/小文字が区別されるパスワードを使用することをお薦めします。ただし、以前のOracle Databaseリリースからアップグレードされたパスワード・ファイルは、大/小文字が区別されない元のパスワードを保持できます。パスワード・ファイルで大/小文字が区別されるようにするため、次の構文を使用して、パスワード・ファイルをある形式から別の形式に移行することにより、大/小文字を区別することをお薦めします。orapwd input_file=input_password_file file=output_password_file

親トピック: Oracle Database 21cでサポートが終了した初期化パラメータ

DISABLE_DIRECTORY_LINK_CHECKのサポート終了

Oracle Database 21c以降、DISABLE_DIRECTORY_LINK_CHECKパラメータはサポートされなくなり、代替はありません。

2021年5月

DISABLE_DIRECTORY_LINK_CHECKパララメータは無効になっています。以前このパラメータで管理されていたシンボリック・リンクは、新しいOracle Databaseリリースで失敗します。アップグレード後に影響を受ける機能を使用しようとして、その機能でシンボリック・リンクが使用された場合、ORA-29283: 無効なファイル操作です。 : パスがsymlinkをトラバースしていますが発生します。

親トピック: Oracle Database 21cでサポートが終了した初期化パラメータ

REMOTE_OS_AUTHENTパラメータのサポート終了

Oracle Database初期化パラメータREMOTE_OS_AUTHENTは、Oracle Database 21cから削除されました。

2021年5月

REMOTE_OS_AUTHENTは、Oracle Database 11gで非推奨になりました。データベースへのセキュアでない可能性がある接続を防ぐために、Oracle Database 21cでこの認証オプションは削除されています。

親トピック: Oracle Database 21cでサポートが終了した初期化パラメータ

SEC_CASE_SENSITIVE_LOGONのサポート終了

SEC_CASE_SENSITIVE_LOGONパラメータは、Oracle Database 21cでサポートが終了しました。

2021年5月

SEC_CASE_SENSITIVE_LOGONパラメータは、Oracle Database 12cリリース1 (12.1)では非推奨でした。新しいパスワードの大文字と小文字を区別するために、OracleはこのパラメータをOracle Database 21cから削除しています。

関連項目

- 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

親トピック: Oracle Database 21cでサポートが終了した初期化パラメータ

CLUSTER_DATABASE_INSTANCESパラメータのサポート終了

CLUSTER_DATABASE_INSTANCESパラメータは、21cではサポートが終了しています。これに替わる機能はありません。

2021年10月

init.oraのパラメータCLUSTER_DATABASE_INSTANCESにより、構成済のOracle Real Application Clusters (Oracle RAC)インスタンスの数を指定していました。このパラメータはOracle Database 19cで非推奨となりました。これは、構成可能なOracle RACインスタンスの数がOracle Clusterwareリソース定義から自動的に導出されるためです。このパラメータを使用する理由はないため、このパラメータは置き換わりません。

親トピック: Oracle Database 21cでサポートが終了した初期化パラメータ

Oracle Database 21cで非推奨となった機能

アップグレード計画の一環として、このOracle Databaseリリースで非推奨となった機能を確認し、アプリケーション戦略に対する代替方法を確認してください。

- AUTO OPTIMIZEフレームワークの非推奨
  Oracle Databaseリリース21cでは、プロシージャADD_AUTO_OPTIMIZEおよびREMOVE_AUTO_OPTIMIZE、ビューCTX_AUTO_OPTIMIZE_INDEXES、CTX_USER_AUTO_OPTIMIZE_INDEXESおよびCTX_AUTO_OPTIMIZE_STATUSが非推奨になりました。
- CTXFILTERCACHE問合せ演算子の非推奨
  Oracle Database Release 21c以降、CTXFILTERCACHEが非推奨になり、CTX_FILTER_CACHE_STATISTICSおよびQUERY_FILTER_CACHE_SIZEも非推奨になりました。
- ポリシー管理型データベースの非推奨
  Oracle Grid Infrastructure 21c以降、ポリシー管理型データベースは非推奨になりました。
- 従来の監査の非推奨
  従来の監査は、Oracle Database 21cで非推奨になりました。Oracle Database内で選択的でより効果的な監査を行うことができる、統合監査を使用することをお薦めします。
- 古い暗号化アルゴリズムの非推奨
  Oracle Database 21c以降、古い暗号化およびハッシュ・アルゴリズムは非推奨になりました。
- クラスタ・ドメインの非推奨 - ドメイン・サービス・クラスタ
  Oracle Grid Infrastructure 21c以降、Oracleクラスタ・ドメイン・アーキテクチャに含まれるドメイン・サービス・クラスタ(DSC)は非推奨になりました。
- エンタープライズ・ユーザー・セキュリティ(EUS)ユーザー移行ユーティリティの非推奨
  エンタープライズ・ユーザー・セキュリティ(EUS)ユーザー移行ユーティリティ(UMU)は、Oracle Database 21cで非推奨になりました。かわりにEUSマネージャ(EUSM)機能を使用してください。
- ロジカル・スタンバイと新しいデータ型
  ロジカル・スタンバイを使用する場合(DBMS_ROLLINGの一部として使用されていない場合)、使用できるのは、Oracle Database 12cリリース2 (12.2)より前に追加されたデータ型のみです
- シャード・キューの非推奨
  AQシャード・キューは、Oracle Database 21cでは非推奨です。かわりにトランザクション・イベント・キュー(TEQ)を使用してください。
- Oracle用MySQLクライアント・ライブラリ・ドライバの非推奨
  Oracle用MySQLクライアント・ライブラリ・ドライバは、Oracle Database 21cでは非推奨です。
- TLS 1.0および1.1トランスポート・レイヤー・セキュリティの非推奨

Oracle Database 21c以降、トランスポート・レイヤー・セキュリティ・プロトコル・バージョン1.0および1.1 (TLS 1.0およびTLS 1.1)は非推奨になりました。

- Unix Crypt (またはMD5crypt)パスワード検証機能の非推奨
  Unix Crypt (MD5crypt)パスワード検証機能アルゴリズムはOracle Database 21cサーバーおよびクライアントで非推奨となり、このアルゴリズムを使用するパスワードは将来のリリースでは機能しなくなります。
- ODP.NET OracleConfiguration.DirectoryTypeプロパティの非推奨
  Oracle Data Provider for .NETのOracleConfiguration DirectoryTypeプロパティおよび.NET構成ファイルのDIRECTORY_TYPE設定は、Oracle Database 21cで非推奨となり、将来のリリースでサポートされなくなる可能性があります。
- 脆弱な暗号化キーの強度の非推奨
  脆弱な暗号化キーの使用は、Oracle Database 21cでは非推奨です。
- 適応しきい値機能DBSNMPのDBSNMPパッケージの非
  推奨適応しきい値機能に関連付けられたPL/SQLパッケージは、Oracle Database 21cでは非推奨です。
- Oracle Sharding高可用性のためのOracle GoldenGateレプリケーションの非推奨
  Oracle Sharding高可用性に対するOracle GoldenGateレプリケーション・サポートは、Oracle Database 21cでは非推奨です。
- アウトバウンドTLS接続での匿名暗号スイートの非推奨
  アウトバウンドTLS接続(データベースから他のサービス)の匿名暗号スイートは、Oracle Database 21cで非推奨になりました。
- KERBEROS5PREアダプタの非推奨
  KERBEROS5PREアダプタの使用は、Oracle Database 21cでは非推奨です。かわりに、KERBEROS5アダプタを使用することを推奨します。
- Oracle Wallet Managerの非推奨
  Oracle Wallet Manager (OWM)は、Oracle Database 21cでは非推奨です。
- Oracle Enterprise Manager Database Expressの非推奨
  Oracle Enterprise Manager Database Express (EM Express)は非推奨となり、将来のOracle Databaseリリースで削除されます。
- サービス属性値SESSION_STATE_CONSISTENCY = STATICの非推奨
  セッション属性値FAILOVER_TYPE = TRANSACTIONとSESSION_STATE_CONSISTENCY = STATICの組合せは、Oracle Database 21cでは非推奨になりました。
- SQLNETおよびDBMS_CRYPTOでのSHA-1の使用の非推奨
  DBMS_CRYPTO、SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENTおよびSQLNET.CRYPTO_CHECKSUM_TYPES_SERVERでのSHA-1の使用は非推奨になりました。
- ACFSUTIL REPL REVERSEの非推奨
  acfsutil repl reverseコマンドは、Oracle Database 21cでは非推奨になりました。かわりにrepl failoverまたはrepl switchoverを使用します。
- Oracle OLAPの非推奨
  分析ワークスペース、OLAP DMLプログラミング言語およびOLAP Java APIは、Oracle Database 21cでは非推奨になりました。
- Oracle Database Extensions for .NETの非推奨
  Oracle Database Extensions for .NETは、Oracle Database 21cでは非推奨になりました。.NETコードを中間層に配置するか、外部プロシージャ機能を使用するか、PL/SQLまたはJavaを使用してコードをリライトすることをお薦めします。

- リポジトリ・イベントの非推奨
  アプリケーション・アクションをトリガーするためのリポジトリ・イベントの使用は、Oracle Database 21c (21.3)では非推奨になりました。これに替わる機能はありません。
- Quality of Service Managementの非推奨
  Oracle Database Release 21c以降では、Oracle Quality of Service Management (QoSMまたはQoS Management)は非推奨となっています。
- Grid Infrastructure管理リポジトリの非推奨
  Grid Infrastructure管理リポジトリ(GIMR)または管理データベース(MGMTDB)は、Oracle Database 21cでは非推奨です。

親トピック: Oracle Database21cの動作の変更、サポート終了および非推奨

AUTO OPTIMIZEフレームワークの非推奨

Oracle Databaseリリース21cでは、プロシージャADD_AUTO_OPTIMIZEおよびREMOVE_AUTO_OPTIMIZE、ビューCTX_AUTO_OPTIMIZE_INDEXES、CTX_USER_AUTO_OPTIMIZE_INDEXESおよびCTX_AUTO_OPTIMIZE_STATUSが非推奨になりました。

日付: 2021年5月

基本的な最適化は、すべての索引に対して自動化されるようになりました。追加の最適化は、CREATE INDEX文で宣言してスケジュールすることもできます。この拡張により、AUTO_OPTIMIZEフレームワーク(プロシージャおよびビュー)は必要なくなりました。CTX_DDLパッケージ内の2つのプロシージャ(ADD_AUTO_OPTIMIZEおよびREMOVE_AUTO_OPTIMIZE)が非推奨になりました。また、ビューCTX_AUTO_OPTIMIZE_INDEXES、CTX_USER_AUTO_OPTIMIZE_INDEXESおよびCTX_AUTO_OPTIMIZE_STATUSは非推奨になりました。

親トピック: Oracle Database 21cで非推奨となった機能

CTXFILTERCACHE問合せ演算子の非推奨

Oracle Database Release 21c以降、CTXFILTERCACHEが非推奨になり、CTX_FILTER_CACHE_STATISTICSおよびQUERY_FILTER_CACHE_SIZEも非推奨になりました。

日付: 2021年5月

CTXFILTERCACHE問合せ演算子は、問合せでよく使用される式を高速化するために設計されました。Oracle Databaseリリース21cでは、この機能は内部の改善された他の機能に置き換えられています。CTXFILTERCACHE演算子は非推奨になりました(そのオペランドを通過して通常の問合せとして実行されます)。これらは機能しなくなったため、ビューCTX_FILTER_CACHE_STATISTICSおよび記憶域属性QUERY_FILTER_CACHE_SIZEも非推奨になりました。

親トピック: Oracle Database 21cで非推奨となった機能

ポリシー管理型データベースの非推奨

Oracle Grid Infrastructure 21c以降、ポリシー管理型データベースは非推奨になりました。

日付: 2021年5月

既存のサーバー・プールは引き続き使用でき、新しいプールとポリシーを作成できます。既存のサーバー・プールを使用するリソースは、引き続きこれらを透過的に使用できます。

CRS構成ポリシーおよびCRSポリシー・セットの使用は、将来のリリースでサポートされなくなる可能性があります。サーバー・プールおよびポリシー管理型データベースのかわりに、新しいマージ型管理スタイルの使用をお薦めします。

親トピック: Oracle Database 21cで非推奨となった機能

従来の監査の非推奨

従来の監査は、Oracle Database 21cで非推奨になりました。Oracle Database内で選択的でより効果的な監査を行うことができる、統合監査を使用することをお薦めします。

日付: 2021年5月

Oracle Databaseの標準の従来の監査は20年以上提供されてきました。従来の監査は、文、権限およびオブジェクトを監査するための組込みサポートを提供していました。長年、データ監査がデータ戦略を成功に導く重要な要素であったため、OracleはOracle Database内で選択的で効果的な監査を提供する必要があると認識していました。この必要性に対処するために、Oracle Database 12cでは統合監査が導入されました。統合監査は、組込みの監査操作をサポートするだけでなく、データベース内の監査の管理を簡素化し、条件に基づいて監査を高速化し、データベースによって生成される監査データのセキュリティを向上させます。統合監査と従来の監査(混合モード)は、Oracle Database 12c以降デフォルトの監査モードでした。混合モードの監査は、統合監査について理解し、従来の監査から移行できるようにするために提供されました。このリリースで従来の監査が非推奨になったため、統合監査に移行することをお薦めします。『Oracle Databaseセキュリティ・ガイド』の移行手順を参照してください。

親トピック: Oracle Database 21cで非推奨となった機能

古い暗号化アルゴリズムの非推奨

Oracle Database 21c以降、古い暗号化およびハッシュ・アルゴリズムは非推奨になりました。

日付: 2021年5月

非推奨のアルゴリズムには、`DBMS_CRYPTO`およびネイティブ・ネットワーク暗号化の場合はMD4、MD5、DES、3DESおよびRC4関連アルゴリズム、透過的データ暗号化(TDE)の場合は3DESがあります。安全性の低い古い暗号化アルゴリズムを削除すると、これらのアルゴリズムを誤って使用することがなくなります。セキュリティ要件を満たすために、Advanced Encryption Standard (AES)などの最新の暗号化アルゴリズムを使用することをお薦めします。

この非推奨の結果、非推奨のアルゴリズムの使用が指定されているかどうかを確認するために、ネットワーク暗号化構成を確認することをお薦めします。見つかった場合は、AESなどの最新の暗号の使用に切り替えます。また、現在TDEデプロイメントに3DES暗号化を使用している場合は、AESなどの最新のアルゴリズムへの移行を計画する必要があります。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 21cで非推奨となった機能

クラスタ・ドメインの非推奨 - ドメイン・サービス・クラスタ

Oracle Grid Infrastructure 21c以降、Oracleクラスタ・ドメイン・アーキテクチャに含まれるドメイン・サービス・クラスタ(DSC)は非推奨になりました。

日付: 2021年5月

特定のクラスタ機能を非推奨にし、制限付きの採用とすることにより、オラクル社では、すべての機能の中核となるスケーリング、可用性および管理性の向上に注力できます。＊Oracleクラスタ・ドメインは、ドメイン・サービス・クラスタ(DSC)とメンバー・クラスタで構成されます。メンバー・クラスタは、Oracle Grid Infrastructure 19cで非推奨になりました。DSCは、引き続き本番クラスタにサービスを提供できます。ただし、それらのサービスの大部分ではホスティングにDSCを必要としなくなったため、DSCのインストールは非推奨になり、将来のリリースでサポートされなくなる可能性があります。該当する場合、以前にDSCでホストされていたサービスに、任意のクラスタまたはシステムを使用することをお薦めします。Oracleでは、代替システムで各サービスを

使用できるようになるまで、共有サービスをホストするためのDSCのサポートを継続します。

親トピック: Oracle Database 21cで非推奨となった機能

エンタープライズ・ユーザー・セキュリティ(EUS)ユーザー移行ユーティリティの非推奨

エンタープライズ・ユーザー・セキュリティ(EUS)ユーザー移行ユーティリティ(UMU)は、Oracle Database 21cで非推奨になりました。かわりにEUSマネージャ(EUSM)機能を使用してください。

日付: 2021年5月

組織のディレクトリ・サービスにはすでにすべての従業員のレコードがあるため、データベース・ユーザーをディレクトリ・サービスに一括移行する必要はありません。EUSマネージャ(EUSM)には、EUS UMUと同じ機能が多数あります。EUS UMUのかわりに使用することをお薦めします。

親トピック: Oracle Database 21cで非推奨となった機能

ロジカル・スタンバイと新しいデータ型

ロジカル・スタンバイを使用する場合(`DBMS_ROLLING`の一部として使用されていない場合)、使用できるのは、Oracle Database 12cリリース2 (12.2)より前に追加されたデータ型のみです

日付: 2021年5月

Oracle Database 12cリリース1 (12.1)以降に追加された新しいデータ型は、Oracle Data Guardロジカル・スタンバイではサポートされません。たとえば、Oracle Data Guardロジカル・スタンバイは、長い識別子、複雑な抽象データ型(ADT)および空間データ型をサポートしていませんこの制限は、Oracle Data Guardフィジカル・スタンバイ・データベース、`DBMS_ROLLING`またはOracle GoldenGateには存在しないことに注意してください。より新しいデータ型のスタンバイ・データベースの利点を得るために、フィジカル・スタンバイ・データベース、スナップショット・スタンバイ・データベースのいずれかの使用を検討するか、またはOracle GoldenGateの論理レプリケーション機能の使用を検討することをお薦めします。

親トピック: Oracle Database 21cで非推奨となった機能

シャード・キューの非推奨

AQシャード・キューは、Oracle Database 21cでは非推奨です。かわりにトランザクション・イベント・キュー(TEQ)を使用してください。

日付: 2021年5月

Oracle Database 21c以降、AQシャード・キューはトランザクション・イベント・キュー(TEQ)として再パッケージされています。Oracle Database 21cリリースでは、TEQはAQシャード・キューと共存します。ただし、AQシャード・キューは、将来のリリースではサポートされなくなり、TEQに置き換えられます。Oracle Real Application Clusters (Oracle RAC)でのスループットを向上させ、パフォーマンスを改善するためにTEQに移行することをお薦めします。

親トピック: Oracle Database 21cで非推奨となった機能

Oracle用MySQLクライアント・ライブラリ・ドライバの非推奨

Oracle用MySQLクライアント・ライブラリ・ドライバは、Oracle Database 21cでは非推奨です。

日付: 2021年5月

MySQLクライアント・ライブラリ・ドライバ`liboramysql`は非推奨です。将来のリリースで`liboramysql`のサポートが終了する可能性があります。これに替わる機能はありません。この非推奨は、`liboramysql`を使用してデータベースに接続する以

前のOracle Databaseクライアント・リリースの機能に影響を与えることはありません。ただし、これらのクライアントで使用できる機能が最終的に制限される可能性があります。

親トピック: Oracle Database 21cで非推奨となった機能

TLS 1.0および1.1トランスポート層セキュリティの非推奨

Oracle Database 21c以降、トランスポート・レイヤー・セキュリティ・プロトコル・バージョン1.0および1.1 (TLS 1.0およびTLS 1.1)は非推奨になりました。

日付: 2021年5月

セキュリティのベスト・プラクティスに従って、OracleはTLS 1.0およびTLS 1.1の使用を非推奨にしました。セキュリティ要件を満たすために、Oracleは、かわりにTLS 1.2を使用することをお薦めします。

親トピック: Oracle Database 21cで非推奨となった機能

Unix Crypt (またはMD5crypt)パスワード検証機能の非推奨

Unix Crypt (`MD5crypt`)パスワード検証機能アルゴリズムは、Oracle Database 21cサーバーおよびクライアントで非推奨となり、このアルゴリズムを使用するパスワードは将来のリリースで機能しなくなります。

日付: 2021年5月

Oracle Internet Directory (OID)のエンタープライズ・ユーザーを持つエンタープライズ・ユーザー・セキュリティ(EUS)の顧客は、OIDに移行する前は、OIDまたはオペレーティング・システム、またはUnix Cryptによって生成された古い安全性の低いパスワード検証機能を使用できました。Unix Cryptは、パスワードをハッシュするための安全性の低いアルゴリズムです。`MD5crypt`が削除されると、Oracle DatabaseはEUSまたはOIDユーザーをUnix Cryptパスワード検証機能タイプで認証できなくなります。新しいセキュアなハッシュ・アルゴリズムを使用して、OIDのパスワードを今すぐリセットすることをお薦めします。

親トピック: Oracle Database 21cで非推奨となった機能

ODP.NET OracleConfiguration.DirectoryTypeプロパティの非推奨

Oracle Data Provider for .NET `OracleConfiguration DirectoryType`プロパティおよび.NET構成ファイルの`DIRECTORY_TYPE`設定は、Oracle Database 21cでは非推奨となり、将来のリリースでサポートされなくなる可能性があります。

日付: 2021年5月

`OracleConfiguration DirectoryServerType`プロパティは、`DirectoryType`プロパティを置き換えます。.NET構成ファイルの`DIRECTORY_SERVER_TYPE`設定によって、`DIRECTORY_TYPE`設定が置換されます。これらのプロパティはすべて同一の機能を持ちます。新しいプロパティを使用して移行することを開発者にお薦めします。`DirectoryServerType`および`DIRECTORY_SERVER_TYPE`の名前は、同等の機能を提供する`DIRECTORY_SERVER_TYPE`よりも、`ldap.ora`パラメータに合わせて調整されています。

親トピック: Oracle Database 21cで非推奨となった機能

脆弱な暗号化キー強度の非推奨

脆弱な暗号化キーの使用は、Oracle Database 21cでは非推奨です。

日付: 2021年5月

暗号アルゴリズムのセキュリティ強度は、最新のRSA BSAFE Micro Edition Suite (MES) v4.5の導入により、Oracle

Database 21cで変更されました。次の暗号化アルゴリズムは非推奨です。

- FIPSモードの場合
  - FIPSのデフォルトの保護強度80は非推奨になりました。この強度は引き続き使用できますが、将来のデフォルトの保護強度にはなりません。FIPSモードの新しいデフォルトの保護強度は112です。
    - 新しいリリースでデフォルトのFIPS保護強度が80から112に変更された場合でも、パラメータを設定することで、古いセキュアでないFIPS保護強度80を使用するように戻すことができます。
  - 1024キー・サイズのDiffie HellmanおよびDigital Signature Algorithm (DH/DSA)は非推奨になりました。サポートされる新しい最小キー・サイズは2048です。1024キー・サイズのサポートは、デフォルトの保護強度が112ビットのセキュリティ強度(2048キー・サイズと同等)に変更され、プロセスの強度が80ビットのセキュリティ強度(1024キー・サイズと同等)のままである場合に使用できます。
- FIPS以外のモードの場合
  - 保護およびプロセス強度0 (RSAキー長512)は非推奨です。デフォルトでは、保護とプロセス強度の両方が80になりました。保護および処理強度0 (RSAキー512および同等のもの)は引き続き使用できますが、使用することはお薦めしません。

関連項目

- 非推奨の脆弱なアルゴリズム・キーの管理

親トピック: Oracle Database 21cで非推奨となった機能

適応しきい値機能のDBSNMPパッケージの非推奨

適応しきい値機能に関連付けられたDBSNMP PL/SQLパッケージは、Oracle Database 21cで非推奨になりました。

日付: 2021年5月

Oracle Enterprise Manager Cloud Control 13.5以降、すべてのOracle Databaseターゲットのデータベース・サーバー・適応しきい値およびベースライン・メトリックしきい値のすべての機能が削除されます。この機能をサポートするデータベース・サーバー側パッケージ`DBSNMP.BSLN`、`DBSNMP.BSLN_INTERNAL`および`DBSNMP.MGMT_RESPONSE`は使用できなくなります。

この非推奨の詳細は、My Oracle Support 2697846.1を参照してください

関連項目

- My Oracle Supportノート2697846.1

親トピック: Oracle Database 21cで非推奨となった機能

Oracle Sharding高可用性のためのOracle GoldenGateレプリケーションの非推奨

Oracle Sharding高可用性のためのOracle GoldenGateレプリケーション・サポートは、Oracle Database 21cでは非推奨になりました。

日付: 2021年5月

親トピック: Oracle Database 21cで非推奨となった機能

アウトバウンドTLS接続での匿名暗号スイートの非推奨

アウトバウンドTLS接続(データベースから他のサービス)の匿名暗号スイートは、Oracle Database 21cでは非推奨です。

日付: 2021年5月

匿名暗号スイートは、クライアントとサーバー間の接続のみを暗号化し、一方のパーティは認証しないため、中間者攻撃に対して脆弱なままになります。このリリースでは、次の3つの匿名暗号スイートが非推奨になりました。

- `NZTLS_DH_ANON_WITH_AES_256_GCM_SHA384`
- `NZTLS_DH_ANON_WITH_AES_128_GCM_SHA256`
- `NZSSL_DH_ANON_WITH_3DES_EDE_CBC_SHA`

接続の保護を強化するために、可能なかぎり強力な非匿名TLS暗号スイートを使用することをお薦めします。

親トピック: Oracle Database 21cで非推奨となった機能

KERBEROS5PREアダプタの非推奨

KERBEROS5PREアダプタの使用は、Oracle Database 21cでは非推奨です。かわりに、KERBEROS5アダプタを使用することを推奨します。

日付: 2021年5月

KERBEROS5アダプタは、Kerberos認証用にOracleでサポートされているプライマリ・アダプタです。KERBEROS5PREは不要になり、将来のリリースでサポートされなくなります。この変更により、Kerberosの構成が簡略化されます。

親トピック: Oracle Database 21cで非推奨となった機能

Oracle Wallet Managerの非推奨

Oracle Wallet Manager (OWM)は、Oracle Database 21cでは非推奨です。

日付: 2021年5月

Oracle Wallet Managerを使用するかわりに、コマンドライン・ツール`orapki`を使用することをお薦めします。

親トピック: Oracle Database 21cで非推奨となった機能

Oracle Enterprise Manager Database Expressの非推奨

Oracle Enterprise Manager Database Express (EM Express)は非推奨になりました。今後のOracle Databaseリリースで削除される予定です。

日付: 2021年5月

EM Expressは、Oracle Database内に構築されるWebベースのデータベース管理ツールです。これは、キー・パフォーマンス管理および基本的なデータベース管理機能をサポートしています。EM Expressの機能の多くは、すべてのOracle Databaseエディションに含まれるOracle SQL Developerでも使用できます。EM ExpressをOracle SQL Developerに置き換えることをお薦めします。

親トピック: Oracle Database 21cで非推奨となった機能

サービス属性値SESSION_STATE_CONSISTENCY = STATICの非推奨

セッション属性値`FAILOVER_TYPE = TRANSACTION`と`SESSION_STATE_CONSISTENCY = STATIC`の組合せは、Oracle Database 21cでは非推奨になりました。

日付: 2023年5月

セッション属性値`FAILOVER_TYPE = TRANSACTION`と`SESSION_STATE_CONSISTENCY = STATIC`の使用は、サポートされているサービス属性の組合せではなくなりました。

かわりに、サービス構成で次のいずれかの組合せを使用します。

- FAILOVER_TYPE = AUTOおよびSESSION_STATE_CONSISTENCY = AUTO
- FAILOVER_TYPE = TRANSACTIONおよびSESSION_STATE_CONSISTENCY = DYNAMIC

これらの構成により、Oracle Databaseでのセッション状態トラッキングが強制され、セッション移行およびセッションのフェイルオーバー時にセッション状態が維持されます。Oracle Database 21c以降は、RESET_STATE属性を設定して、リクエストの最後にアプリケーションにより設定されたセッション状態をクリアすることもできます。詳細は、RESET_STATEを参照してください。

親トピック: Oracle Database 21cで非推奨となった機能

SQLNETおよびDBMS_CRYPTOに対するSHA-1の使用の非推奨

DBMS_CRYPTO、SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENTおよびSQLNET.CRYPTO_CHECKSUM_TYPES_SERVERでのSHA-1の使用は非推奨になりました。

2021年5月

パラメータSQLNET.CRYPTO_CHECKSUM_TYPES_CLIENTおよびSQLNET.CRYPTO_CHECKSUM_TYPES_SERVERを指定したSHA-1 (Secure Hash Algorithm 1)の使用は、このリリースでは非推奨となり、将来のリリースではサポートされなくなる可能性があります。DBMS_CRYPTOでのSHA-1暗号の使用も非推奨になりました(HASH_SH1、HMAC_SH1)。SHA1を使用するかわりに、SHA-1暗号のかわりにより強力なSHA-2暗号の使用を開始することをお薦めします。

親トピック: Oracle Database 21cで非推奨となった機能

ACFSUTIL REPL REVERSEの非推奨

acfsutil repl reverseコマンドは、Oracle Database 21cでは非推奨になりました。かわりにrepl failoverまたはrepl switchoverを使用します。

日付: 2021年8月

Oracle Advanced Cluster File System (ACFS)コマンド・ユーティリティacfsutilには、コマンドrepl failoverおよびrepl switchoverが含まれます。これらのコマンドは、acfsutil repl reverseのすべての機能を含め、さらに多くの機能を提供します。このため、acfsutil repl reverseコマンドは非推奨になっています。

親トピック: Oracle Database 21cで非推奨となった機能

Oracle OLAPの非推奨

分析ワークスペース、OLAP DMLプログラミング言語およびOLAP Java APIは、Oracle Database 21cでは非推奨になりました。

日付: 2021年8月

高度な分析機能が必要な新しいアプリケーションの場合、分析ビュー(Oracle Databaseの機能)またはOracle Essbaseを検討することをお薦めします。Oracle分析ビューは、あらゆるOracle Databaseエディションの機能です。アプリケーションでディメンション問合せおよびレポート・アプリケーションにOLAPを使用する場合は、Oracle分析ビューをOLAPの代替として検討することをお薦めします。分析ビューは、既存のデータベース表およびビューに格納されているデータの分析問合せを迅速に効率的に作成する方法を提供します。Oracle分析ビューを使用すると、"キューブ・ビルド/更新"プロセスを必要とせずに、ディメンション問合せモデルおよびサポート・メタデータを取得できます。キューブ・ビルド/更新プロセスを省くことで、スケーラビリティ制約(モデルの複雑さとデータ量)が緩和され、データ準備パイプラインが簡素化され、データ待機時間が短縮または省かれます。また、これらの非リレーショナル・データ型がリレーショナル・ビューによってラップされている場合、分析ビューをOLTPアプリケーション、外部表、非リレーショナル・データ型(JSONなど)とともに使用できます。

親トピック: Oracle Database 21cで非推奨となった機能

Oracle Database Extensions for .NETの非推奨

Oracle Database Extensions for .NETは、Oracle Database 21cでは非推奨になりました。.NETコードを中間層に配置するか、外部プロシージャ機能を使用するか、PL/SQLまたはJavaを使用してコードをリライトすることをお薦めします。

日付: 2021年9月

Oracle Database Extensions for .NETは、Microsoft Windows上のOracle Databaseの機能であり、.NETによって管理される言語(C#など)で記述されたストアド・プロシージャおよびファンクションが使用できるようになります。

Oracle Databaseは、Oracleデータベース・プロセスの外にある外部プロセス内でMicrosoft Common Language Runtime(CLR)をホストします。アプリケーション開発者は、C#やVB.NETなど、任意の.NET準拠言語を使用してストアド・プロシージャおよびファンクションを記述し、これらの.NETストアド・プロシージャを他のPL/SQLまたはJavaストアド・プロシージャと同様にデータベースで使用できます。.NETストアド・プロシージャは、PL/SQLパッケージ、プロシージャ、ファンクションおよびSQL文からのトリガー、またはPL/SQLプロシージャやファンクションをコール可能な場所ならどこからでもコールできます。

次の移行オプションがあります。

- 中間層への.NETコード(アセンブリ)の移動
- 外部プロシージャ機能を使用した、外部プロセスのロードおよび.NETアセンブリの実行
- PL/SQLまたはJavaを使用したストアド・プロシージャのリライト

親トピック: Oracle Database 21cで非推奨となった機能

リポジトリ・イベントの非推奨

アプリケーション・アクションをトリガーするためのリポジトリ・イベントの使用は、Oracle Database 21c (21.3)では非推奨になりました。これに替わる機能はありません。

2021年10月

リポジトリ・イベントは、アプリケーション・アクションのトリガーに使用できるイベントです。リポジトリ・イベントには、リソースの作成、削除、ロック、ロック解除、レンダリング、リンク、リンク解除、バージョン管理、チェックイン、チェックアウト、チェックアウト解除(チェックアウトされたバージョンの回復)、オープン、更新などのリポジトリの変更が含まれます。リポジトリ・イベントの使用の非推奨には、`DBMS_XEVENT`パッケージおよび次のサブプログラム・グループの非推奨が含まれます。

- `XDBevent`
- `XDBRepositoryEvent`
- `XDBHandler`
- `XDBHandlerList`
- `XDBPath`
- `XDBLink`

親トピック: Oracle Database 21cで非推奨となった機能

Quality of Service Managementの非推奨

Oracle Database Release 21c以降では、Oracle Quality of Service Management (QoSMまたはQoS Management)は非推奨となっています。

日付: 2022年11月

Oracle QoSMでは、事前定義されたポリシーに基づいてシステム構成を調整し、必要なパフォーマンス・レベルでアプリケーションを継続的に実行できるようにすることで、システム全体のワークロード管理が自動化されます。アプリケーションおよびデータベースはますます、QoSMのリソース管理機能の一部を提供するシステムにデプロイされるようになってきています。同時に、OracleのAutonomous Health Frameworkは、システムとその関連コンポーネントの健全性および運用能力に影響を与える事

象や状態を緩和するための推奨事項を調整したり提供するように強化されています。このため、QoSMはOracle Database 21cでは非推奨となっています。

親トピック: Oracle Database 21cで非推奨となった機能

グリッド・インフラストラクチャ管理リポジトリの非推奨

Oracle Database 21cでは、グリッド・インフラストラクチャ管理リポジトリ(GIMR)または管理データベース(`MGMTDB`)は非推奨です。

日付: 2023年5月

Oracle Autonomous Health Framework (AHF)、Cluster Health Monitor (CHM)、Oracle Cluster Health Advisor (CHA)、Oracle Fleet Patching and Provisioning (FPP)などのGIMR (GIMRクライアント)を使用するソリューションは、必要に応じて代替リポジトリに自動的に移行します。GIMRクライアントのみが使用する専用データベースとして、将来のリリースでのGIMRの非推奨と潜在的な削除は、Oracle Grid Infrastructureと関連サービスでシームレスになります。

親トピック: Oracle Database 21cで非推奨となった機能

Oracle Database 21cで非推奨になったビュー

アップグレード計画の一環として、このOracle Databaseリリースから非推奨になったビューを確認してください。

- 従来の監査ビューの非推奨
  従来の監査が非推奨になった結果、従来の監査に関連付けられたビューも非推奨になりました。

親トピック: Oracle Database21cの動作の変更、サポート終了および非推奨

従来の監査ビューの非推奨

従来の監査が非推奨になった結果、従来の監査に関連付けられたビューも非推奨になりました。

日付: 2021年5月

- 静的データ・ディクショナリ・ビュー:
  - `ALL_DEF_AUDIT_OPTS`
  - `AUDIT_ACTIONS`
  - `DBA_AUDIT_EXISTS`
  - `DBA_AUDIT_OBJECT`
  - `DBA_AUDIT_SESSION`
  - `DBA_AUDIT_STATEMENT`
  - `DBA_AUDIT_TRAIL`
  - `DBA_COMMON_AUDIT_TRAIL`
  - `DBA_FGA_AUDIT_TRAIL`
  - `DBA_OBJ_AUDIT_OPTS`
  - `DBA_PRIV_AUDIT_OPTS`

- DBA_STMT_AUDIT_OPTS
- USER_AUDIT_OBJECT
- USER_AUDIT_SESSION
- USER_AUDIT_STATEMENT
- USER_AUDIT_TRAIL
- USER_OBJ_AUDIT_OPTS

- 動的パフォーマンス・ビュー:
  - V$XML_AUDIT_TRAIL

親トピック: Oracle Database 21cで非推奨になったビュー

Oracle Database 21cで非推奨になったパラメータ

アップグレード計画の一環として、このOracle Databaseリリースから非推奨になった初期化パラメータを確認してください。

- 従来の監査初期化パラメータの非推奨
  従来の監査が非推奨になった結果、従来の監査に関連付けられた初期化パラメータも非推奨になりました。

親トピック: Oracle Database21cの動作の変更、サポート終了および非推奨

従来の監査初期化パラメータの非推奨

従来の監査が非推奨になった結果、従来の監査に関連付けられた初期化パラメータも非推奨になりました。

日付: 2021年5月

- 初期化パラメータ:
  - AUDIT_FILE_DEST
  - AUDIT_SYS_OPERATIONS
  - AUDIT_SYSLOG_LEVEL
  - AUDIT_TRAIL

親トピック: Oracle Database 21cで非推奨になったパラメータ

Oracle Database 19cの動作の変更、サポート終了および非推奨

Oracle Database 19cリリースの変更の説明を確認してください。

- Oracle Database 19cアップグレード計画の動作の変更
  Oracle Database 19cへのアップグレードの計画に役立つ動作の変更を確認してください
- Oracle Database 19cでサポートが終了した機能
  アップグレード計画の一環として、Oracle Database 19cでサポートが終了した機能を確認してください。
- Oracle Database 19cでサポートが終了したパラメータ
  アップグレード計画の一環として、Oracle Database 19c以降でサポートされなくなった初期化パラメータを確認して

ください。

- Oracle Database 19cで非推奨となった機能
  アップグレード計画の一環として、Oracle Database 19で非推奨となった機能を確認し、アプリケーション戦略に対する代替方法を確認してください。
- Oracle Database 19cで非推奨となった初期化パラメータ
  アップグレード計画の一環として、Oracle Database 19cで非推奨となった初期化パラメータを確認し、代替方法を確認してください。

親トピック: Oracle Databaseの変更、サポート終了および非推奨

Oracle Database 19cアップグレード計画の動作の変更

Oracle Database 19cへのアップグレードの計画に役立つ動作の変更を確認してください

動作の変更には、パラメータまたは機能のデフォルトの変更、製品名の変更、および注意が必要なデータベース構成に対するその他の変更が含まれる場合があります。

- リリース更新(RU)に含まれるすべてのタイム・ゾーン・ファイル(DST)
  Oracle Database 19c RU 19.18.0以降では、使用可能なすべてのDSTパッチがRUとともにインストールされ、`Oracle_home/oracore/zoneinfo`ディレクトリにデプロイされます。
- Oracle Data Guardプロパティ管理の変更点
  Oracle Database 19c以上で、Oracle Data Guard構成のプロパティは、外部構成ファイルではなくOracle Databaseに格納されます。
- 高速ホーム・プロビジョニング(RHP)の名前の変更
  Oracle Database 19cおよびOracle Grid Infrastructure 19c以上では、高速ホーム・プロビジョニングの名前がフリート・パッチ適用およびプロビジョニング(FPP)に変更されました。
- OCRおよび投票ディスクの直接ファイル配置の再サポート
  Oracle Grid Infrastructure 19c以上では、共有ファイル・システム上のOCRおよび投票ファイルの直接配置のサポート終了は、Oracleスタンドアロン・クラスタに対して廃棄されます。
- グリッド・インフラストラクチャ管理リポジトリのオプションのインストール
  Oracle Grid Infrastructure 19c以上では、グリッド・インフラストラクチャ管理リポジトリ(GIMR)は、Oracleスタンドアロン・クラスタの新規インストールでオプションです。Oracleドメイン・サービス・クラスタでは、GIMRをサービス・コンポーネントとしてインストールする必要があります。
- DBMS_JOBのサポート
  `DBMS_JOB`パッケージは引き続きサポートされます。ただし、`DBMS_JOB`ジョブを発行するデータベース・スキーマに`CREATE JOB`権限を付与する必要があります。
- Standard Edition高可用性について
  Oracle Database 19 cリリース更新(19.7)以降、Oracle Database Standard Edition 2は高可用性モードでインストールできます。
- DBMS_OPTIM_BUNDLEによる「インストールされているが無効になっている」モジュールのバグ修正の管理
  SQL実行計画の変更の原因となるOracle Databaseのバグ修正の実装を管理するには、`DBMS_OPTIM_BUNDLE`を使用します。
- Windows認証でNTLMがデフォルトで使用されなくなりました
  `AUTHENTICATION_SERVICES=NTS`を使用したMicrosoft Windowsインストールでは、このOracle Databaseリリースで`sqlnet.ora`ファイルの`SQLNET.NO_NTLM`パラメータの設定はデフォルトで`TRUE`になり、ORA-12638エラーが発生する可能性があります。

親トピック: Oracle Database 19cの動作の変更、サポート終了および非推奨

リリース更新(RU)に含まれるすべてのタイム・ゾーン・ファイル(DST)

Oracle Database 19c RU 19.18.0以降では、使用可能なすべてのDSTパッチがRUとともにインストールされ、`Oracle_home/oracore/zoneinfo`ディレクトリにデプロイされます。

DSTパッチをインストールしても、現在のデータベース操作には影響しません。ただし、RUとともにパッチをインストールすると、データベースのタイムゾーン・バージョンを簡単に調整できます(そのようにする必要がある場合)。たとえば、トランスポータブル表領域またはフル・トランスポータブル・エクスポート/インポートを使用している場合は、ソース・データベースとターゲット・データベースで同じ文字セットとタイム・ゾーン設定が使用されていることを確認する必要があります。この変更により、宛先データベースをデフォルトとは異なるタイムゾーン・ファイル・バージョンを使用するように簡単に変更できます。

> ノート:
>
> デフォルトでは、AutoUpgrade はデータベースのタイム・ゾーンを最新の使用可能なレベルに変更します。タイム・ゾーンをアップグレードしない場合は、AutoUpgrade 構成ファイル内のローカル・パラメータ `timezone_upg` を明示的に no に設定する必要があります。たとえば:
>
> ```
> upg1.timezone_upg=no
> ```
>
> RU 19.18 以降のパッチをデータベースに適用すると、そのパッチが適用された Oracle ホームの Database Configuration Assistant (DBCA)で作成される新しいデータベースは最新のタイム・ゾーン・ファイルを使用して作成されます。

関連項目

- タイム・ゾーン・ファイルの選択
- RUには使用可能なすべてのDSTパッチが含まれています
- Primary Note DST FAQ : Updated DST Transitions and New Time Zones in Oracle RDBMS and OJVM Time Zone File Patches (ドキュメントID 412160.1)

親トピック: Oracle Database 19cアップグレード計画の動作の変更

Oracle Data Guardのプロパティ管理の変更点

Oracle Database 19c位以上では、Oracle Data Guard構成のプロパティは外部構成ファイルではなく、Oracle Databaseに格納されます。

Oracle Data Guardのプロパティ名、記憶域の場所および動作は、Oracle Database 19cで変更されています。

プロパティ名の変更

表10-1 Oracle Data Guardのプロパティ名の変更

プロパティ	Oracle Database 18c以前のリリース	Oracle Database 19c以上のリリース
アーカイブの場所	`OnlineArchiveLocation`	`ArchiveLocation`

プロパティ	Oracle Database 18c以前のリリース	Oracle Database 19c以上のリリース
代替の場所	`OnlineAlternateLocation`	`AlternateLocation`
スタンバイ・アーカイブの場所	`StandbyArchiveLocation`	`StandbyArchiveLocation`
スタンバイの代替の場所	`StandbyAlternateLocation`	`StandbyAlternateLocation`

プロパティ動作の変更

- `StandbyArchiveLocation`および`StandbyAlternateLocation`に空の文字列がある場合、`ArchiveLocation`および`AlternateLocation`がオンライン・ログ・ファイルとスタンバイ・ログ・ファイルの両方の場所になります
- `StandbyArchiveLocation`および`StandbyAlternateLocation`に空でない文字列がある場合、`ArchiveLocation`および`AlternateLocation`がオンライン・ログ・ファイルのみの場所になります
- `StandbyArchiveLocation`および`StandbyAlternateLocation`の動作は変更されません。これらのプロパティは、スタンバイ・ログ・ファイルの場所でのみ使用されます。

範囲の変更

Oracle Database 19c以上では、4つのOracle Data Guardプロパティすべてに`Database`という範囲があります。以前のリリースでは、*インスタンス*という範囲がありました。

インポートおよびアップグレード

Oracle Database 19c以上では、Oracle Data Guardによるプロパティのインポートおよびアップグレードの管理方法に対する次の変更に注意してください。

- Oracle Data Guard Brokerは、ローカル・アーカイブの場所のプロパティを自動的にインポートしなくなりました。
- Oracle Data Guard Brokerは、Oracle Database 18c以前のリリースのData Guard Brokerエクスポートから作成されたメタデータ・ファイルから、以前のリリースのプロパティ設定を自動的にアップグレードしなくなりました。

親トピック: Oracle Database 19cアップグレード計画の動作の変更

高速ホーム・プロビジョニング(RHP)の名前の変更

Oracle Database 19cおよびOracle Grid Infrastructure 19c以上では、高速ホーム・プロビジョニングの名前がフリート・パッチ適用およびプロビジョニング(FPP)に変更されました。

親トピック: Oracle Database 19cアップグレード計画の動作の変更

OCRおよび投票ディスクの直接ファイル配置の再サポート

Oracle Grid Infrastructure 19c以上では、共有ファイル・システム上のOCRおよび投票ファイルの直接配置のサポート終了は、Oracleスタンドアロン・クラスタに対して廃棄されます。

Oracle Grid Infrastructure 12cリリース2 (12.2)では、直接、共有ファイル・システム上でOracle Grid InfrastructureのOracle Cluster Registry (OCR)および投票ファイルの配置がサポートされなくなることが発表されました。このサポート終了は現在破棄されました。Oracle Grid Infrastructure 19c (19.3)以上では、Oracleスタンドアロン・クラスタとともに、OCRおよび投票ディスク・ファイルを共有ファイル・システムに直接配置できます。ただし、Oracleドメイン・サービス・

クラスタの場合、Oracle Automatic Storage Management (Oracle ASM)で管理される定数障害グループにOCRおよび投票ファイルを引き続き配置する必要があります。

親トピック: Oracle Database 19cアップグレード計画の動作の変更

グリッド・インフラストラクチャ管理リポジトリのオプションのインストール

Oracle Grid Infrastructure 19c以上では、グリッド・インフラストラクチャ管理リポジトリ(GIMR)は、Oracleスタンドアロン・クラスタの新規インストールでオプションです。Oracleドメイン・サービス・クラスタでは、GIMRをサービス・コンポーネントとしてインストールする必要があります。

Oracleスタンドアロン・クラスタは、GIMRをOracle ASMディスク・グループまたは共有ファイル・システムにローカルにホストし、このGIMRは単一プラガブル・データベース(PDB)を含むマルチテナント・データベースです。グローバルGIMRは、Oracleドメイン・サービス・クラスタで実行されます。Oracleドメイン・サービス・クラスタは、GIMRを別のOracle ASMディスク・グループにローカルにホストします。データベース用のOracleメンバー・クラスタなどのクライアント・クラスタは、Oracleドメイン・サービス・クラスタにあるリモートGIMRを使用します。2ノード・クラスタまたは4ノード・クラスタの場合、リモート・クラスタにクラスタのGIMRをホスティングすると、クラスタ上の追加のインフラストラクチャ・リポジトリを実行する際のオーバーヘッドが削減されます。Oracleドメイン・サービス・クラスタ用のGIMRは、1つのPDBと、追加される各メンバー・クラスタ用の追加のPDBを持つマルチテナント・データベースです。

親トピック: Oracle Database 19cアップグレード計画の動作の変更

DBMS_JOBのサポート

DBMS_JOBパッケージは引き続きサポートされます。ただし、DBMS_JOBジョブを発行するデータベース・スキーマにCREATE JOB権限を付与する必要があります。

Oracle Schedulerは、DBMS_JOBパッケージを置換します。DBMS_JOBは下位互換性のために引き続きサポートされますが、DBMS_JOBからOracle Schedulerに切り替えることをお薦めします。

Oracle Database 19c以降のリリースでは、アップグレードでDBMS_SCHEDULERを使用して既存のDBMS_JOBジョブを再作成できる場合、下位互換性のために、アップグレード後も引き続きDBMS_JOBはDBMS_SCHEDULERジョブに対するレガシー・インタフェースとして機能します。メタデータの問題のため、DBMS_SCHEDULERを使用して既存のジョブを再作成できない場合は、アップグレードの事前チェックを実行した際にJOB_TABLE_INTEGRITY警告が表示されます。その場合、次の3つのオプションがあります。

- メタデータを修正します。アップグレード後も引き続きDBMS_JOBSをインタフェースとして使用して実行し、DBMS_SCHEDULERジョブとして実行します。
- 不要な場合は、ジョブを削除します。
- DBMS_JOBSジョブを削除し、DBMS_SCHEDULERを使用して手動でジョブを再作成します。

アップグレード時に再作成されるDBMS_JOBで作成された既存のジョブの場合、DBMS_JOBレガシー・ジョブは引き続きインタフェースとして存在しますが、これを使用すると常にDBMS_SCHEDULERエントリが作成されます。インタフェースは別として、ジョブはDBMS_SCHEDULERジョブとして実行されます。その後、アップグレード前に作成されたDBMS_JOBジョブを無効にすると、DBMS_SCHEDULERジョブも無効になります。この動作を回避するには、レガシー・ジョブを削除し、DBMS_SCHEDULERジョブで置換します。

すべての新しいジョブには、DBMS_SCHEDULERを使用します。

親トピック: Oracle Database 19cアップグレード計画の動作の変更

Standard Edition High Availabilityについて

Oracle Database 19 cリリース・アップデート(19.7)以降、高可用性モードでOracle Database Standard Edition 2をインストールできます。

Standard Edition高可用性では、Oracle Clusterwareを使用し、単一インスタンスのStandard Edition Oracle Databases用にクラスタベースのフェイルオーバーを提供します。

Oracle Standard Edition High Availabilityでは、Oracle Clusterware、Oracle Automatic Storage Management (Oracle ASM)、Oracle ASM Cluster File System (Oracle ACFS)など、すでにOracle Grid Infrastructureの一部であるクラスタ機能およびストレージ・ソリューションを利用しています。

Oracle ASMやOracle ACFSなど、統合され同時にマウントされた共有記憶域をデータベース・ファイルおよび非構造化データに使用すると、Oracle Grid Infrastructureでは、フェイルオーバーのノード上でOracle Databaseを、ボリュームおよびファイル・システムのフェイルオーバーやリマウントに依存するクラスタ・ソリューションよりもはるかに速く再起動できます。

Standard Edition高可用性は、Linux x86-64、Oracle Solaris on SPARC (64ビット)およびMicrosoft Windowsでサポートされます。

Oracle Database 19cリリース更新(19.13)以降、Standard Edition高可用性がIBM AIX on POWER Systems (64ビット)でサポートされます。

ノート:

この項は、Standard Edition Oracle Databases 19c にクラスタ・ベースのデータベース・フェイルオーバーを提供する、Standard Edition 高可用性に固有のものです。Oracle Database の高可用性オプションの詳細は、『Oracle Clusterware 管理およびデプロイメント・ガイド』を参照してください。

親トピック: Oracle Database 19cアップグレード計画の動作の変更

DBMS_OPTIM_BUNDLEによる「インストールされているが無効になっている」モジュールのバグ修正の管理

SQL実行計画の変更の原因となるOracle Databaseのバグ修正の実装を管理するには、`DBMS_OPTIM_BUNDLE`を使用します。

データベースをアップグレードした後、リリース更新に含まれる実行計画の変更の原因となるバグ修正パッチがデフォルトで無効になっている状態でインストールされます。これらのバグ修正は、修正を有効にするまで有効になりません。これらの修正を手動で有効にするか、`DBMS_OPTIM_BUNDLE`パッケージを使用できます。

本番システムで使用するこれらの無効になっているパッチを有効にし、アップグレード・テスト計画の一部としてこれらのパッチの使用を徹底的にテストすることをお薦めします。

`DBMS_OPTIM_BUNDLE`を使用して、実行計画を変更する可能性があるために無効にされたパッチを有効にする方法の詳細は、*Oracle Database PL/SQL*パッケージおよびタイプ・リファレンス、およびMy Oracle Supportノート2147007.1を参照してください。

関連項目

- DBMS_OPTIM_BUNDLE
- My Oracle SupportドキュメントID 2147007.1 DBMS_OPTIM_BUNDLEによるデータベース・リリース更新での「インストールされているが無効になっている」バグ修正の管理

親トピック: Oracle Database 19cアップグレード計画の動作の変更

Windows認証でNTLMがデフォルトで使用されなくなった

AUTHENTICATION_SERVICES=NTSを使用したMicrosoft Windowsインストールでは、このOracle Databaseリリースでsqlnet.oraファイルのSQLNET.NO_NTLMパラメータの設定がデフォルトでTRUEになり、ORA-12638エラーが発生する可能性があります。

日付: 2023年8月

以前のリリースでは、AUTHENTICATION_SERVICES=NTSのデフォルトはFALSEでした。SQLNET.NO_NTLMは、NTS認証でNTLMを使用できるかどうかを制御します。TRUE設定が、NTS認証でNTLMを使用できないことを意味します。NTLMは、通常、相互認証を提供せず安全性が低くなるため、SQLNET.NO_NTLMにTRUEを設定すると、データベースとクライアントがよりセキュアになります。

SQLNET.NO_NTLMパラメータは、サーバーとクライアントの両方で使用されます。SQLNET.NO_NTLMが設定されていないOracleデータベースまたはクライアントのMicrosoft Windowsインストールをアップグレードした場合、デフォルトはTRUEになります。この場合、sqlnet.oraにSQLNET.AUTHENTICATION_SERVICES=NTSがあると、クライアントでエラーORA-12638: Credential retrieval failedが発生する可能性があります。

特定のクライアントに対してNTLM認証を使用する場合は、クライアント側のsqlnet.oraファイルで必要に応じてこのパラメータを設定します。

```
SQLNET.NO_NTLM=FALSE
```

この設定はサーバーとクライアントの両方に含める必要があり、両方で同じ設定になっている必要があります。SQLNET.NO_NTLMがTRUEに設定されていることを確認することをお薦めします。ただし、extproc、仮想アカウントまたはWindowsのローカル・アカウントで認証に失敗した場合は、クライアントSQLNET.NO_NTLMをFALSEに設定してからログインを再試行してください。サーバーのSQLNET.NO_NTLMを変更した場合は、データベースを再起動する必要があります。

関連項目

- 今すぐデータベースをアップグレード - ORA-12638 (Windowsのみ)は、Oracle 19.10.0以降
- Windows : 2021年1月のWINDBBP 19.10.0.0.210119をWindows 64/32ビット・データベース・クライアントに適用した後、データベースに接続してORA-12638を取得している間(ドキュメントID 2757734.1)

親トピック: Oracle Database 19cアップグレード計画の動作の変更

Oracle Database 19cでサポートが終了した機能

アップグレード計画の一環として、Oracle Database 19cでサポートが終了した機能を確認してください。

- Oracle Data Provider for .NETの昇格可能なトランザクション設定のサポート終了
  Oracle Data Provider for .NETのPromotableTransaction設定は、不要になったためサポートが終了しました。すべての互換性のあるデータベース・サーバー・バージョンで、トランザクションの昇格がサポートされています。
- Oracle Multimediaのサポート終了
  Oracle MultimediaはOracle Database 19cでサポートが終了し、実装が削除されています。
- LogMinerのCONTINUOUS_MINE機能のサポート終了
  dbms_logmnr.start_logmnrパッケージのcontinuous_mineオプションはOracle Database 19cでサポートが終了し、使用できなくなりました。
- 拡張データ型サポート(EDS)のサポート終了

拡張データ型サポート(EDS)機能はOracle Database 19cでサポートが終了しました。EDS機能でサポートされるすべてのデータ型は、ロジカル・スタンバイとOracle GoldenGateの両方でネイティブにサポートされます。

- Data Guard BrokerのMaxConnectionsプロパティのサポート終了
  Oracle Database 19c以上では、Oracle Data Guard Brokerの`MAX_CONNECTIONS`属性のサポートが終了しました。
- フレックス・クラスタ・アーキテクチャでのリーフ・ノードのサポート終了
  リリーフ・ノードはOracle Grid Infrastructure 19cのOracle Flex Clusterアーキテクチャでサポートされなくなりました。
- Oracle Streamsのサポート終了
  Oracle Database 19c (19.1)以上では、Oracle Streamsはサポートが終了しました。Oracle GoldenGateがOracle Database用の代替ソリューションです。
- PRODUCT_USER_PROFILE表のサポート終了
  Oracle Database 19c以上では、SQL*Plus表`PRODUCT_USER_PROFILE` (PUP表)はサポートが終了しました。
- Standard Edition 2 (SE2)データベース・エディション用のOracle Real Application Clustersのサポート終了
  Oracle Database 19c以上では、Oracle Real Application Clusters (Oracle RAC)はOracle Database Standard Edition 2 (SE2)でサポートされなくなりました。

親トピック: Oracle Database 19cの動作の変更、サポート終了および非推奨

Oracle Data Provider for .NETの昇格可能なトランザクション設定のサポート終了

Oracle Data Provider for .NETの`PromotableTransaction`設定は、不要になったためサポートが終了しました。すべての互換性のあるデータベース・サーバー・バージョンで、トランザクションの昇格がサポートされています。

日付: 2019年4月

Oracle Data Provider for .NETのレジストリ、構成およびプロパティ設定`PromotableTransaction`は、アプリケーションでトランザクションをローカルとして維持するか、またはすべての単一接続トランザクションをローカルとして開始してから、2番目の接続の登録時にトランザクションを分散型に昇格するかどうかを指定します。これは、昇格可能なトランザクションの概念です。

`PromotableTransaction`設定は、Oracle Data Provider for .NET 18cでサポートが終了しました。これは、このプロバイダ・バージョンと互換性のあるすべてのデータベース・バージョンが昇格可能なトランザクションをサポートしているためです。開発者が昇格可能なトランザクションを使用している場合、この設定を使用する必要がなくなりました。既存のアプリケーションは、昇格可能なトランザクションが使用されているかどうかに関係なく影響を受けません。

親トピック: Oracle Database 19cでサポートが終了した機能

Oracle Multimediaのサポート終了

Oracle MultimediaはOracle Database 19cでサポートが終了し、実装が削除されています。

日付: 2019年4月

イメージ処理および変換のかわりに、SecureFiles LOBにマルチメディア・コンテンツを格納し、APEX Media Extension (AME)などのサード・パーティ製品を使用することをお薦めします。`ORDIM`コンポーネントはレジストリに残り、`VALID`ステータスのままです。Oracle Multimediaオブジェクトおよびパッケージはデータベースに残ります。ただし、これらのオブジェクトおよびパッケージは機能しなくなり、使用しようとすると例外が発生します。Oracle Locatorは、Oracle Multimediaのサポート終了による影響を受けません。

関連項目

- https://www.apexmediaextension.com

親トピック: Oracle Database 19cでサポートが終了した機能

LogMinerのCONTINUOUS_MINE機能のサポート終了

`dbms_logmnr.start_logmnr`パッケージの`continuous_mine`オプションはOracle Database 19cでサポートが終了し、使用できなくなりました。

日付: 2019年4月

LogMinerパッケージの`continuous_mine`機能は廃止されています。これは、Oracle Database 12cリリース2 (12.2)で非推奨になっていました。かわりとなる機能はありません。

親トピック: Oracle Database 19cでサポートが終了した機能

拡張データ型サポート(EDS)のサポート終了

拡張データ型サポート(EDS)は、Oracle Database 19cでサポートが終了しました。EDS機能でサポートされるすべてのデータ型は、ロジカル・スタンバイとOracle GoldenGateの両方でネイティブにサポートされます。

日付: 2019年4月

拡張データ型サポート(EDS)機能では、ネイティブなREDOベースのサポートが欠如した特定のOracleデータ型をサポートするためのロジカル・スタンバイ用のメカニズムが提供されます。たとえば、EDSは、`SDO_GEOMETRY`列を持つ表のレプリケートに使用されていました。ただし、Oracle Database 12cリリース2 (12.2)以上では、EDSでサポートされているOracleデータ型で、ロジカル・スタンバイまたはGoldenGateでネイティブにサポートされていないものはありません。この機能は、Oracle Database 19c (19.1)でサポートが終了しました。

親トピック: Oracle Database 19cでサポートが終了した機能

Data Guard BrokerのMaxConnectionsプロパティのサポート終了

Oracle Database 19c以上では、Oracle Data Guard Brokerの`MAX_CONNECTIONS`属性のサポートが終了しました。

日付: 2019年4月

Oracle Data Guard Brokerの`MaxConnections`プロパティ(`LOG_ARCHIVE_DEST_n`パラメータの`MAX_CONNECTIONS`属性に関連)は、Oracle Database 19cではサポートが終了しました。これは削除されました。コマンドを使用してDGMGRLからこのプロパティを設定すると、エラーが返されます。

親トピック: Oracle Database 19cでサポートが終了した機能

フレックス・クラスタ・アーキテクチャでのリーフ・ノードのサポート終了

リリーフ・ノードはOracle Grid Infrastructure 19cのOracle Flex Clusterアーキテクチャでサポートされなくなりました。

日付: 2019年4月

Oracle Grid Infrastructure 19c (19.1)以上のリリースでは、Oracle Flex Clusterのすべてのノードはハブ・ノードとして機能します。Oracle Flex Clusterアーキテクチャの元の実装でリーフ・ノードによって提供された機能は、ハブ・ノードで容易に提供できます。したがって、リーフ・ノードはサポートされなくなりました。

親トピック: Oracle Database 19cでサポートが終了した機能

Oracle Streamsのサポート終了

Oracle Database 19c (19.1)以上では、Oracle Streamsはサポートが終了しました。Oracle GoldenGateがOracle Database用の代替ソリューションです。

日付: 2019年4月

Oracle Databaseアドバンスト・キューイングは非推奨ではなく、Oracle Database 19cで完全にサポートされています。Oracle Streamsは、マルチテナント・アーキテクチャ、`LONG VARCHAR`およびその他の新機能など、Oracle Database 12c (12.1)以降のリリースに追加された機能をサポートしていませんでした。Oracle Streamsレプリケーション機能はGoldenGateによって置き換えられます。

親トピック: Oracle Database 19cでサポートが終了した機能

PRODUCT_USER_PROFILE表のサポート終了

Oracle Database 19c以上では、SQL*Plus表`PRODUCT_USER_PROFILE` (PUP表)はサポートが終了しました。

日付: 2019年4月

このSQL*Plusの製品レベルのセキュリティ機能は、Oracle Database 19cでは使用できません。Oracle Databaseの設定を使用してデータを保護し、すべてのクライアント・アプリケーション間で一貫したセキュリティを確保することをお薦めします。

親トピック: Oracle Database 19cでサポートが終了した機能

Standard Edition 2 (SE2)データベース・エディション用のOracle Real Application Clusterのサポート終了

Oracle Database 19c以上では、Oracle Real Application Clusters (Oracle RAC)はOracle Database Standard Edition 2 (SE2)でサポートされなくなりました。

日付: 2019年4月

Oracle Real Application Clusters (Oracle RAC)機能を使用するOracle Database Standard Editionデータベースを以前のリリースからOracle Database 19cにアップグレードすることはできません。これらのデータベースをOracle Database 19cにアップグレードするには、アップグレードを開始する前にOracle RAC機能を削除するか、Oracle Database Standard EditionからOracle Database Enterprise Editionにアップグレードします。アップグレード後のシステムの再構成方法を含む各ステップの詳細は、My Oracle Supportノート2504078.1「Oracle Database Standard Edition 19cでのOracle Real Application Clusters(RAC)のサポート終了」を参照してください。

関連項目

- My Oracle Supportのドキュメント2504078.1

親トピック: Oracle Database 19cでサポートが終了した機能

Oracle Database 19cでサポートが終了したパラメータ

アップグレード計画の一環として、Oracle Database 19c以降、サポートされなくなった初期化パラメータを確認してください。

- EXAFUSION_ENABLED初期化パラメータのサポート終了
  Oracle Exadata Database Machine初期化パラメータ`EXAFUSION_ENABLED`は、Oracle Database 19cでサポートが終了しました。
- LOG_ARCHIVE_DEST_nパラメータのMAX_CONNECTIONS属性のサポート終了

Oracle Data GuardのREDOトランスポートのLOG_ARCHIVE_DEST_nパラメータのMAX_CONNECTIONS属性は廃止されています。Oracle Database 19cではサポートが終了しました。

- O7_DICTIONARY_ACCESSのサポート終了
  初期化パラメータO7_DICTIONARY_ACCESSIBILITYは、Oracle Database 19cでサポートが終了しました。
- OPTIMIZE_PROGRESS_TABLEパラメータのサポート終了
  Oracle GoldenGate Integrated Replicat、XStream InおよびロジカルスタンバイでのOPTIMIZE_PROGRESS_TABLEは、Oracle Database 19cでサポートが終了しました。

親トピック: Oracle Database 19cの動作の変更、サポート終了および非推奨

EXAFUSION_ENABLED初期化パラメータのサポート終了

Oracle Exadata Database Machine初期化パラメータEXAFUSION_ENABLEDは、Oracle Database 19cでサポートが終了しました。

日付: 2019年4月

Exafusion機能はOracle Database 12cリリース1 (12.1.0.2)で導入されましたが、デフォルトでは無効になっています。これはLinuxオペレーティング・システム用にのみ使用でき、Oracle Exadata Database Machineでのみ使用できます。この機能を有効にするには、EXAFUSION_ENABLED初期化パラメータを1に設定します。Oracle Database 12cリリース2 (12.2)で、この機能はOracle Linux上で動作するOracle Exadata Database Machineでデフォルトで有効になりました。この機能を無効にするには、EXAFUSION_ENABLEDパラメータ設定を0に変更します。ただし、Oracle Database 18c以上のリリースでは、Exafusion機能を無効にすることはできません。この理由のため、EXAFUSION_ENABLEDパラメータは機能を提供していないため、Oracle Database 19cでサポートが終了しました。

親トピック: Oracle Database 19cでサポートが終了したパラメータ

LOG_ARCHIVE_DEST_nのMAX_CONNECTIONS属性のサポート終了

Oracle Data Guard REDOトランスポートのLOG_ARCHIVE_DEST_nパラメータのMAX_CONNECTIONS属性は廃止されています。Oracle Database 19cではサポートが終了しました。

日付: 2019年4月

MAX_CONNECTIONS属性は、Oracle Database 11gで導入された新しいREDOトランスポート・ストリーミング・メカニズムと干渉し、ギャップの解決に必要な時間が増える可能性があります。これらのタイプのエラーを防止するために、Oracleではこの属性のサポートを終了し、削除しました。

親トピック: Oracle Database 19cでサポートが終了したパラメータ

O7_DICTIONARY_ACCESSのサポート終了

初期化パラメータO7_DICTIONARY_ACCESSIBILITYは、Oracle Database 19cでサポートが終了しました。

日付: 2019年11月

SYSが所有するオブジェクトにシステム権限でアクセスする場合の制限は、O7_DICTIONARY_ACCESSIBILITYパラメータによって制御されていました。これは、以前のリリースのアプリケーションの特定の下位互換性を保つために維持されています。廃止された機能のサポートを終了することにより、オラクル社では、すべての機能におけるセキュリティに注力できます。標準のセキュリティのベスト・プラクティスに従って、システム権限を管理することをお薦めします。

親トピック: Oracle Database 19cでサポートが終了したパラメータ

OPTIMIZE_PROGRESS_TABLEパラメータのサポート終了

Oracle GoldenGate Integrated Replicat、XStream Inおよびロジカル・スタンバイでの`OPTIMIZE_PROGRESS_TABLE`は、Oracle Database 19cでサポートが終了しました。

日付: 2020年9月

Oracle GoldenGate Integrated Replicat、XStream Inおよびロジカル・スタンバイでの適用パラメータ`OPTIMIZE_PROGRESS_TABLE`は、Oracle Database 19cでサポートが終了しました。Oracle Database 19にアップグレードする前に、このパラメータをオフにする必要があります。`OPTIMIZE_PROGRESS_TABLE`がONに設定されている場合は、適用を適切に停止し、パラメータをオフにして適用を再開します。GoldenGateの適用およびXStreamの場合、このパラメータはデフォルトで`OFF`に設定されています。

親トピック: Oracle Database 19cでサポートが終了したパラメータ

Oracle Database 19cで非推奨となった機能

アップグレード計画の一環として、Oracle Database 19で非推奨となった機能を確認し、アプリケーション戦略に対する代替方法を確認してください。

- URL_DATASTOREテキスト型の非推奨
  Oracle Database 19c以降、Oracle Textの`URL_DATASTORE`型は非推奨になりました。かわりに`NETWORK_DATASTORE`を使用してください。
- FILE_DATASTORE型の非推奨
  Oracle Database 19c以降、Oracle Textの`FILE_DATASTORE`型は非推奨になりました。かわりに`DIRECTORY_DATASTORE`を使用してください。
- Oracle Data Guard Brokerの非推奨となったプロパティ
  Oracle Database 19c以上では、初期化パラメータに関連付けられた複数のOracle Data Guard Brokerプロパティが非推奨になりました。これらの機能は、`DGMGRL`の新しい`EDIT … SET PARAMETER`コマンドに置き換えられています。
- Oracle Data Guardのロジカル・スタンバイ・プロパティの非推奨となった機能
  Oracle Database 19c以上では、Oracle Data Guard Brokerのロジカル・スタンバイ・プロパティが非推奨になりました。
- コマンドラインのASMCMD PWCREATEの非推奨
  Oracle ASMコマンドライン・ユーティリティの`ASMCMD`コマンド・オプション`pwcreate password`を使用したASMパスワードの作成は、Oracle Grid Infrastructure 19c (19.1)では非推奨になりました。
- Addnodeスクリプトの非推奨
  `addnode`スクリプトはOracle Grid Infrastructure 19cで非推奨になりました。クラスタにノードを追加する機能は、インストーラ・ウィザードで使用できます。
- clone.plスクリプトの非推奨
  `clone.pl`スクリプトはOracle Database 19cで非推奨になりました。ゴールド・イメージを使用してソフトウェアのみのインストールを実行する機能は、インストーラ・ウィザードで使用できます。
- Oracle Fail Safeの非推奨
  Oracle Fail SafeはOracle Database 19cで非推奨になりました。今後のリリースではサポートが終了し、使用できなくなる可能性があります。
- GDSCTLオペレーティング・システム・コマンドライン・パスワード・リセットの非推奨
  セキュリティを強化するために、Oracle Database 19c以上では、オペレーティング・システム・プロンプトからコールさ

れたときにグローバル・データ・サービス制御ユーティリティ(GDSCTL)のコマンドラインからパスワードを指定する機能は非推奨になりました。

- Oracle Enterprise Manager Expressの非推奨
  フラッシュベースのEnterprise Manager ExpressはOracle Database 19cで非推奨になりました。Oracle Database 19c以上では、Enterprise Manager ExpressはユーザーインタフェースにJava JETテクノロジを使用します。
- DV_REALM_OWNERロールの非推奨
  Oracle Database VaultロールのDV_REALM_OWNERロールは非推奨になり、代替機能もありません。
- DV_REALM_RESOURCEロールの非推奨
  Oracle Database VaultロールのDV_REALM_RESOURCEは非推奨になり、代替機能もありません。
- DV_PUBLICロールの非推奨
  Oracle Database VaultロールのDV_PUBLICロールは非推奨になり、代替機能もありません。
- Oracle ACFSレプリケーション・プロトコルREPV1の非推奨
  Oracle Database 19c (19.3)以上では、Oracle ACFSレプリケーション・プロトコルrepv1は非推奨になりました。
- Oracle OLAPの非推奨
  分析ワークスペース、OLAP DMLプログラミング言語およびOLAP Java APIは、Oracle Database 19cでは非推奨になりました。
- SolarisおよびWindowsでのOracle ACFS暗号化の非推奨
  Oracle Database 19c (19.3)以上では、Oracle SolarisおよびMicrosoft Windowsで代替機能がないため、Oracle ACFS暗号化は非推奨になりました。
- WindowsでのOracle ACFSの使用の非推奨
  Oracle Grid Infrastructure 19c (19.5)以降では、Microsoft WindowsでのOracle ASM Cluster File System (ACFS)の使用は非推奨になりました。
- Oracle ACFSセキュリティ(Vault)とACFS監査の非推奨
  Oracle Grid Infrastructure 19c (19.5)以降では、Oracle ASM Cluster File System (ACFS)セキュリティ(Vault)およびACFS監査は非推奨になりました
- メンバー・クラスタでのOracle ACFSの使用(ACFSリモート)の非推奨
  Oracle Grid Infrastructure 19c (19.5)以降では、メンバー・クラスタ(ACFSリモート)上のOracle ASM Cluster File System (ACFS)は非推奨になりました。
- クラスタ・ドメインの非推奨 - メンバー・クラスタ
  Oracle Grid Infrastructure 19c (19.5)以降、Oracleクラスタ・ドメイン・アーキテクチャに含まれるメンバー・クラスタは非推奨になりました。
- Oracle Clusterwareとベンダー・クラスタウェアの統合の非推奨
  Oracle Clusterware 19c (19.5)以降では、Oracle Clusterwareとベンダーまたはサード・パーティのクラスタウェアとの統合は非推奨になりました。
- Oracle Clusterwareを使用したブラック・ボックス仮想マシン管理の非推奨
  Oracle Clusterwareを使用した仮想マシン(VM)リソースの直接管理は、Oracle Grid Infrastructure 19cで非推奨となり、将来のリリースで削除される可能性があります。
- アウトオブプレース・パッチ適用のOPatchおよびOPatchAutoの非推奨
  OpatchおよびOPatchAutoを使用したOracle推奨のアウトオブプレース・パッチ適用ですが、Oracleは、Oracle Database 23cオンプレミスから、アウトオブプレース・パッチ適用のためのよりシンプルで簡単なGoldイメージ・プロセスに移行しています。
- データ・リカバリ・アドバイザ(DRA)の非推奨

Oracle Database 19c以降、データ・リカバリ・アドバイザ(DRA)機能は非推奨になりました。

- DBUAおよび手動アップグレードの非推奨
  Oracleでは、データベースのアップグレードにAutoUpgradeを使用することをお薦めします。

親トピック: Oracle Database 19cの動作の変更、サポート終了および非推奨

URL_DATASTOREテキスト型の非推奨

Oracle Database 19c以降、Oracle TextのURL_DATASTORE型は非推奨になりました。かわりにNETWORK_DATASTOREを使用してください。

日付: 2019年4月

URL_DATASTORE型は、インターネット上のファイル(HTTPまたはFTPを介してアクセス)に格納されているテキスト、およびローカル・ファイル・システムのファイル(ファイル・プロトコルを介してアクセス)に格納されているテキストに使用されます。これはNETWORK_DATASTOREに置き換えられ、ACLを使用して特定のサーバーへのアクセスを許可します。この変更により、データベースからURLにアクセスするための標準的な運用およびセキュリティ・モデルに対して、Oracle Textの整合性が高まります。

親トピック: Oracle Database 19cで非推奨となった機能

FILE_DATASTORE型の非推奨

Oracle Database 19c以降、Oracle TextのFILE_DATASTORE型は非推奨になりました。かわりにDIRECTORY_DATASTOREを使用してください。

日付: 2019年4月

FILE_DATASTOREテキスト索引を、Oracle Database 19c以降で使用可能なDIRECTORY_DATASTORE索引タイプに置き換えることをお薦めします。DIRECTORY_DATASTOREによってディレクトリ・オブジェクトに基づいたファイル・アクセスが可能になるため、セキュリティが向上します。

親トピック: Oracle Database 19cで非推奨となった機能

Oracle Data Guard Brokerの非推奨となったプロパティ

Oracle Database 19c以上では、初期化パラメータに関連付けられた複数のOracle Data Guard Brokerプロパティが非推奨になりました。これらの機能は、DGMGRLの新しいEDIT … SET PARAMETERコマンドに置き換えられています。

日付: 2019年4月

次のOracle Data Guard BrokerプロパティがOracle Database 19cで非推奨になりました。

- ArchiveLagTarget
- DataGuardSyncLatency
- LogArchiveMaxProcesses
- LogArchiveMinSucceedDest
- LogArchiveTrace
- StandbyFileManagement
- DbFileNameConvert
- LogArchiveFormat
- LogFileNameConvert

これらのプロパティを指定した現在のEDIT ... SET PROPERTYコマンドは、引き続き使用できます。ただし、新しいコマンド

では更新が自動的に行われ、パラメータ・データはブローカのメタデータ・ファイルに格納されなくなりました。

`InconsistentProperties`プロパティも非推奨になりました。矛盾する値がなくなる可能性があるため、このパラメータには常に値がありません。

新しい`EDIT ... SET PARAMETER`コマンドでは、ブローカとデータベース間で構成データが矛盾する可能性がなくなります。新しい`EDIT...SET PARAMETER`コマンドを使用する場合、新しいブローカ・コマンドを使用するか、標準のSQL*Plus `ALTER SYSTEM`コマンドを使用して、これらのパラメータを変更できます。ただし、ブローカ・コマンドを使用する場合は、構成内の任意のデータベースに接続できます。また、構成内の他のデータベースに対してパラメータ変更を実行できます。

親トピック: Oracle Database 19cで非推奨となった機能

Oracle Data Guardのロジカル・スタンバイ・プロパティの非推奨となった機能

Oracle Database 19c以上では、Oracle Data Guard Brokerのロジカル・スタンバイ・プロパティは非推奨になりました。

日付: 2019年4月

ロジカル・スタンバイに影響する次のOracle Data Guard Brokerプロパティは非推奨になりました。

- `LsbyMaxEventsRecorded`
- `LsbyMaxServers`
- `LsbyMaxSga`
- `LsbyPreserveCommitOrder`
- `LsbyRecordAppliedDdl`
- `LsbyRecordSkippedDdl`
- `LsbyRecordSkipErrors`
- `LsbyParameter`

`EDIT ... SET PROPERTY`コマンドは、引き続き使用できます。ただし、設定に関するデータはブローカのメタデータ・ファイルに格納されなくなりました。かわりに、SQL*Plusパッケージ`DBMS_LOGSTDBY`を使用してロジカル・スタンバイ・プロパティを変更することをお薦めします。Oracle Data Guard Brokerのロジカル・スタンバイ・プロパティは、将来のリリースでサポートされなくなる可能性があります。

SQL*Plusパッケージ`DBMS_LOGSTDBY`を直接使用すると、ブローカとロジカル・スタンバイ・データベース間の構成データが矛盾する可能性がなくなり、ロジカル・スタンバイを管理するためのインタフェースが1つ提供されます。

親トピック: Oracle Database 19cで非推奨となった機能

コマンドラインのASMCMD PWCREATEの非推奨

Oracle ASMコマンドライン・ユーティリティのASMCMDコマンド・オプション`pwcreate password`を使用したASMパスワードの作成は、Oracle Grid Infrastructure 19c (19.1)では非推奨になりました。

日付: 2019年4月

コマンドラインでパスワードを指定するオプションは、引き続きOracle Database 19cで有効です。ただし、セキュリティを強化するために、Oracleではこの方法で新しいOracle ASMパスワードを作成することをお薦めしていません。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。ASMCMDの`pwcreate`オプションを指定すると、コマンドラインでパスワードを指定できます。ただし、コマンド`asmcmd pwcreate`を実行してコマンドラインでパスワードを指定しない場合、パスワードの入力を求められます。

親トピック: Oracle Database 19cで非推奨となった機能

Addnodeスクリプトの非推奨

addnodeスクリプトはOracle Grid Infrastructure 19cで非推奨になりました。クラスタにノードを追加する機能は、インストーラ・ウィザードで使用できます。

日付: 2019年4月

addnodeスクリプトは今後のリリースで削除される可能性があります。addnodeスクリプト(addnode.shまたはaddnode.bat)を使用するのではなく、インストーラ・ウィザードを使用してノードを追加します。インストーラ・ウィザードには、addnodeスクリプトに対する多数の拡張機能が用意されています。インストーラ・ウィザードを使用すると、すべてのソフトウェア・ライフサイクル操作を1つのツールに統合することで管理が簡略化されます。

親トピック: Oracle Database 19cで非推奨となった機能

clone.plスクリプトの非推奨

clone.plスクリプトはOracle Database 19cで非推奨になりました。ゴールド・イメージを使用してソフトウェアのみのインストールを実行する機能は、インストーラ・ウィザードで使用できます。

日付: 2019年4月

clone.plスクリプトは今後のリリースで削除される可能性があります。clone.plスクリプトを使用するかわりに、インストーラ・ウィザードを使用して、抽出されたゴールド・イメージをホームとしてインストールすることをお薦めします。

親トピック: Oracle Database 19cで非推奨となった機能

Oracle Fail Safeの非推奨

Oracle Fail SafeはOracle Database 19cで非推奨になりました。今後のリリースではサポートが終了し、使用できなくなる可能性があります。

日付: 2019年4月

Oracle RAC One Nodeなどの単一ノード・フェイルオーバー・オプションを評価することをお薦めします。

親トピック: Oracle Database 19cで非推奨となった機能

GDSCTLオペレーティング・システム・コマンドライン・パスワード・リセットの非推奨

セキュリティを強化するために、Oracle Database 19c以上では、オペレーティング・システム・プロンプトからコールされたときにグローバル・データ・サービス制御ユーティリティ(GDSCTL)のコマンドラインからパスワードを指定する機能は非推奨になりました。

日付: 2019年4月

この非推奨機能は、ユーザー・コマンドライン・プロンプトからGDSCTLがコールされる場合にパスワードを変更する際にのみ適用されます。たとえば、次のコマンドは非推奨になりました。

```
$ gdsctl add database -connect inst1 -pwd gsm_password
```

GDSCTLユーティリティからのパスワードの指定は、引き続き有効です。たとえば、次のコマンドは有効です。

```
GDSCTL> add database -connect inst1 -pwd gsm_password
```

この非推奨機能では、オペレーティング・システム・プロンプトからコールされたGDSCTLコマンドでパスワードを指定する際のセキュリティの脆弱性に対処しています。GDSCTLで要求された場合のみ、グローバル・データ・サービスのパスワードを入力します。

親トピック: Oracle Database 19cで非推奨となった機能

Oracle Enterprise Manager Expressの非推奨

フラッシュベースのEnterprise Manager ExpressはOracle Database 19cで非推奨になりました。Oracle Database 19c以上では、Enterprise Manager ExpressはユーザーインタフェースにJava JETテクノロジを使用します。

日付: 2019年4月

業界標準に従って、OracleではフラッシュベースのOracle Enterprise Manager Express (Oracle EM Express)を非推奨にしています。Oracle Database 19c以上では、Oracle Databaseのデフォルト管理オプションであるの Oracle EM Expressは、Java JETテクノロジに基づいています。この初期リリースでは、フラッシュベースのOracle EM Expressで使用可能ないくつかのオプションがありますが、JETバージョンでは使用できなくなりました。必要に応じて、次のコマンドを使用すると、フラッシュのOracle EM Expressに戻ります。

```
SQL> @?/rdbms/admin/execemx emx
```

JET Oracle EM Expressに戻るには、次のコマンドを使用します。

```
SQL> @?/rdbms/admin/execemx omx
```

親トピック: Oracle Database 19cで非推奨となった機能

DV_REALM_OWNERロールの非推奨

Oracle Database VaultロールのDV_REALM_OWNERロールは非推奨になり、代替機能もありません。

日付: 2019年4月

DV_REALM_OWNERロールは、レルムを定義する複数のスキーマでデータベース・オブジェクトを管理するレルム管理に使用されます。Oracleでは、このロールの使用は非推奨になりました。今後のリリースで削除される可能性があります。

さらに、次のDV_REALM_OWNER権限はDV_REALM_OWNERロールから取り消されます: CREATE ROLE、ALTER ANY ROLE、DROP ANY ROLE、GRANT ANY ROLE、GRANT ANY PRIVILEGEおよびGRANT ANY OBJECT PRIVILEGE。必要に応じて、DV_REALM_OWNERロールにこれらの権限を付与できます。たとえば:

```
SQL> GRANT CREATE ROLE ON tablename TO DV_REALM_OWNER;
```

親トピック: Oracle Database 19cで非推奨となった機能

DV_REALM_RESOURCEロールの非推奨

Oracle Database VaultロールのDV_REALM_RESOURCEは非推奨になり、代替機能もありません。

日付: 2019年4月

DV_REALM_RESOURCEロールはレルム・リソースの管理のために使用します。Oracleでは、このロールの使用は非推奨になりました。今後のリリースで削除される可能性があります。

親トピック: Oracle Database 19cで非推奨となった機能

DV_PUBLICロールの非推奨

Oracle Database VaultロールのDV_PUBLICロールは非推奨になり、代替機能もありません。

日付: 2019年4月

インストール時にDV_PUBLICロールは引き続き作成されますが、ロールや権限は付与されません。以前のリリースでDV_PUBLICに付与された権限はすべてPUBLICロールに直接付与されるようになりました。このロールは廃止されています。

今後のリリースで削除される可能性があります。

親トピック: Oracle Database 19cで非推奨となった機能

Oracle ACFSレプリケーション・プロトコルREPV1の非推奨

Oracle Database 19c (19.3)以上では、Oracle ACFSレプリケーション・プロトコルrepv1は非推奨になりました。

日付: 2019年4月

初期ACFSレプリケーション・プロトコルrepv1は、Oracle Database 11gリリース2 (11.2)でリリースされました。Oracle Database 12cリリース2では、新しいACFSレプリケーション・プロトコルであるOracle ACFSスナップショットベースのレプリケーション(repv2)が導入されました。同じ管理インタフェースが引き続き使用されます。Oracle Database 19c以上では、以前のACFSレプリケーション・プロトコル(repv1)は非推奨になりました。スナップショットベースのレプリケーションに対して更新します。

親トピック: Oracle Database 19cで非推奨となった機能

Oracle OLAPの非推奨

分析ワークスペース、OLAP DMLプログラミング言語およびOLAP Java APIは、Oracle Database 19cでは非推奨になりました。

日付: 2019年4月

この非推奨は、Oracle Database 19cおよびOracle Database 21cで発表されています。高度な分析機能が必要な新しいアプリケーションの場合、分析ビュー(Oracle Databaseの機能)またはOracle Essbaseを検討することをお薦めします。Oracle分析ビューは、あらゆるOracle Databaseエディションの機能です。アプリケーションでディメンション問合せおよびレポート・アプリケーションにOLAPを使用する場合は、Oracle分析ビューをOLAPの代替として検討することをお薦めします。分析ビューは、既存のデータベース表およびビューに格納されているデータの分析問合せを迅速に効率的に作成する方法を提供します。Oracle分析ビューを使用すると、"キューブ・ビルド/更新"プロセスを必要とせずに、ディメンション問合せモデルおよびサポート・メタデータを取得できます。キューブ・ビルド/更新プロセスを省くことで、スケーラビリティ制約(モデルの複雑さとデータ量)が緩和され、データ準備パイプラインが簡素化され、データ待機時間が短縮または省かれます。また、これらの非リレーショナル・データ型がリレーショナル・ビューによってラップされている場合、分析ビューをOLTPアプリケーション、外部表、非リレーショナル・データ型(JSONなど)とともに使用できます。

親トピック: Oracle Database 19cで非推奨となった機能

SolarisおよびWindowsでのOracle ACFS暗号化の非推奨

Oracle Database 19c (19.3)以上では、Oracle SolarisおよびMicrosoft Windowsで代替機能がないため、Oracle ACFS暗号化は非推奨になりました。

日付: 2019年4月

Oracle SolarisおよびMicrosoft WindowsでのOracle ACFS暗号化は、RSAテクノロジに基づいています。RSAテクノロジのリタイアが通知されました。Linuxでは代替テクノロジを使用するため、Oracle ACFS暗号化はLinuxで引き続きサポートされ、この非推奨に影響を受けません。

親トピック: Oracle Database 19cで非推奨となった機能

WindowsでのOracle ACFSの非推奨

Oracle Grid Infrastructure 19c (19.5)以降では、Microsoft WindowsでのOracle ASM Cluster File System (ACFS)の使用は非推奨になりました。

日付: 2019年10月

特定のクラスタ機能を非推奨にし、制限付きの採用とすることにより、オラクル社では、すべての機能の中核となるスケーリング、可用性および管理性の向上に注力できます。Microsoft WindowsでのACFSファイル・システムの使用は非推奨であり、将来のリリースでサポートされなくなる可能性があります。ユース・ケースに応じて、現在のACFSファイル・システムを置き換えるために、Oracle Automatic Storage Management (Oracle ASM)、Oracle Databaseファイル・システム(DBFS)またはMicrosoft Windowsの共有に移行することをお薦めします。

親トピック: Oracle Database 19cで非推奨となった機能

Oracle ACFSセキュリティ(Vault)およびACFS監査の非推奨

Oracle Grid Infrastructure 19c (19.5)以降では、Oracle ASM Cluster File System (ACFS)セキュリティ(Vault)およびACFS監査は非推奨になりました

日付: 2019年10月

特定のクラスタ機能を非推奨にし、制限付きの採用とすることにより、オラクル社では、すべての機能の中核となるスケーリング、可用性および管理性の向上に注力できます。Oracle ACFSセキュリティ(Vault)およびACFS監査は非推奨であり、将来のリリースでサポートされなくなる可能性があります。

親トピック: Oracle Database 19cで非推奨となった機能

メンバー・クラスタでのOracle ACFSの使用(ACFSリモート)の非推奨

Oracle Grid Infrastructure 19c (19.5)以降では、メンバー・クラスタでのOracle ASM Cluster File System (ACFS)の使用(ACFSリモート)は非推奨になりました。

日付: 2019年10月

メンバー・クラスタでのOracle ASM Cluster File System (ACFS)の使用(ACFSリモート)は非推奨であり、将来のリリースで削除される可能性があります。特定のクラスタ機能を非推奨にし、制限付きの採用とすることにより、オラクル社では、すべての機能の中核となるスケーリング、可用性および管理性の向上に注力できます。

親トピック: Oracle Database 19cで非推奨となった機能

クラスタ・ドメインの非推奨 - メンバー・クラスタ

Oracle Grid Infrastructure 19c (19.5)以降、Oracleクラスタ・ドメイン・アーキテクチャに含まれるメンバー・クラスタは非推奨になりました。

日付: 2019年10月

特定のクラスタ機能を非推奨にし、制限付きの採用とすることにより、オラクル社では、すべての機能の中核となるスケーリング、可用性および管理性の向上に注力できます。Oracleクラスタ・ドメインは、ドメイン・サービス・クラスタ(DSC)とメンバー・クラスタで構成されます。メンバー・クラスタが非推奨になったことは、DSCと使用しているクラスタリングに影響を及ぼしますが、他の本番クラスタにサービスをホストする機能には影響しません。ソフトウェアまたはハードウェアを次にアップグレードする機会にあわせて、クラスタ・ドメイン-メンバー・クラスタをオフに移行することをお薦めします。

親トピック: Oracle Database 19cで非推奨となった機能

Oracle Clusterwareとベンダー・クラスタウェアの統合の非推奨

Oracle Clusterware 19c (19.5)以降、ベンダーまたはサード・パーティのクラスタウェアとOracle Clusterwareの統合は非推奨になっています。

日付: 2019年10月

Oracle Clusterwareとベンダー・クラスタウェアの統合は非推奨であり、将来のリリースでサポートされなくなる可能性があります。特定のクラスタ機能を非推奨にし、制限付きの採用とすることにより、オラクル社では、すべての機能の中核となるスケーリング、可用性および管理性の向上に注力できます。異なるクラスタ・ソリューション間の統合がない場合、システムでクラスタ・ソリューションの競合の問題が生じることがあります。独立した各クラスタ・ソリューションでは、特定の障害が発生した場合に実行する必要がある修正処理が個別に決定される可能性があります。競合を回避するには、いつの時点においても1つのクラスタ・ソリューションのみがアクティブになるようにしてください。このため、次回のソフトウェアまたはハードウェアのアップグレードで、ベンダー・クラスタ・ソリューションを使用しない構成に移行することをお薦めします。

親トピック: Oracle Database 19cで非推奨となった機能

Oracle Clusterwareを使用したブラック・ボックス仮想マシン管理の非推奨

Oracle Clusterwareを使用した仮想マシン(VM)リソースの直接管理は、Oracle Grid Infrastructure 19cで非推奨となり、将来のリリースで削除される可能性があります。

日付: 2020年5月

Oracleでは引き続き、リモートにデプロイされたOracle VMの可用性を監視および管理するためにOracle Clusterwareの使用をサポートします。ただし、このサポートは今後のリリースで削除される予定です。

親トピック: Oracle Database 19cで非推奨となった機能

OPatchおよびOPatchAutoアプトオブプレース・パッチ適用の非推奨

OracleではOpatchおよびOPatchAutoを使用したアウトオブプレース・パッチ適用を推奨しましたが、Oracleでは、Oracle Database 23cオンプレミスからアウトオブプレース・パッチ適用のための、よりシンプルで簡単なGoldイメージ・プロセスに移行しています。

日付: 2023年5月

Oracleでは、Oracle Database 19cおよび21cのOracle DatabaseおよびOracle Grid Infrastructureホームにアウトオブプレース・パッチ適用を引き続き推奨しています。ただし、Oracleは、Oracle Databaseのアウトオブプレース・パッチ適用にGoldイメージを使用する効率的なプロセスに移行しています。また、Oracleは、ソフトウェアにパッチを適用するための完全に自動化されたフリート・パッチ適用およびプロビジョニング・ソリューションも提供します。したがって、Oracleは、アウトオブプレース・パッチ適用にOpatchおよびOpatchAutoを使用しないという非推奨を発表しています。この非推奨は、インプレース・パッチ適用には影響しません。

Oracle Grid Infrastructureの場合、Oracleでは、`switchGridHomes`オプションを指定して`gridSetup.sh`を使用することをお薦めします。スタンドアロン・データベース・ホームの場合、Oracleでは、新規インストールを使用し、`applyRU`および`applyOneOffs`オプションを指定してOracle Universal Installerを使用してアウトオブプレース・パッチを実行することをお薦めします。その後、AutoUpgradeまたはフリート・パッチ適用およびプロビジョニングのいずれかを使用してデータベースにパッチを適用できます。これにより、パッチ・プロセス全体が自動化されます。

親トピック: Oracle Database 19cで非推奨となった機能

データ・リカバリ・アドバイザ(DRA)の非推奨

Oracle Database 19c以降、データ・リカバリ・アドバイザ(DRA)機能は非推奨になりました。

日付: 2023年7月

DRAの非推奨には、`LIST FAILURE`、`ADVISE FAILURE`、`REPAIR FAILURE`および`CHANGE FAILURE`のOracle

Recovery Manager (RMAN)コマンドの非推奨が含まれています。データベース管理者はこれらのコマンドにアクセスできなくなります。DRAに代わる機能はありません。

親トピック: Oracle Database 19cで非推奨となった機能

DBUAおよび手動アップグレードの非推奨

Oracleでは、データベースのアップグレードにAutoUpgradeを使用することをお薦めします。

日付: 2023年7月

Oracle Databaseのアップグレードの信頼性とサポートを改善するために、Oracleでは、Database Upgrade Assistant (DBUA)、およびパラレル・アップグレード・ユーティリティ(`catctl.pl`)、およびデータベース・アップグレード・スクリプト`dbupgrade`および`dbupgrade.cmd`を使用する手動アップグレードをOracle Database 19cで非推奨にしています。Oracle DatabaseアップグレードでAutoUpgradeを使用します。これらの廃止されたアップグレード方法は、将来のリリース更新で削除される可能性があります。廃止されたアップグレード方法を非推奨にすることで、オラクル社はAutoUpgradeユーティリティの機能と管理性の改善と拡張に集中できます。

親トピック: Oracle Database 19cで非推奨となった機能

Oracle Database 19cで非推奨となった初期化パラメータ

アップグレード計画の一環として、Oracle Database 19cで非推奨となった初期化パラメータを確認してください。

- 非推奨となったCLUSTER_DATABASE_INSTANCES初期化パラメータ
  Oracle Database初期化パラメータ`CLUSTER_DATABASE_INSTANCES`は、Oracle Database 19c (19.1)で非推奨になりました。
- SQLNET.ENCRYPTION_WALLET_LOCATIONパラメータの非推奨
  SQLNET.ENCRYPTION_WALLET_LOCATION `sqlnet.ora`パラメータは、Oracle Database 19cで非推奨になりました。
- SERVICE_NAMES初期化パラメータの非推奨
  Oracle Database 19c以降では、ユーザーによる`SERVICE_NAMES`パラメータの使用は非推奨になりました。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。

親トピック: Oracle Database 19cの動作の変更、サポート終了および非推奨

非推奨となったCLUSTER_DATABASE_INSTANCES初期化パラメータ

Oracle Database初期化パラメータ`CLUSTER_DATABASE_INSTANCES`は、Oracle Database 19c (19.1)で非推奨になりました。

日付: 2019年4月

`init.ora`パラメータ`CLUSTER_DATABASE_INSTANCES`では、構成済のOracle Real Application Clusters (Oracle RAC)インスタンスの数を指定します。Oracle Database 19c以上のリリースでは、構成可能なOracle RACインスタンスの数がOracle Clusterwareリソース定義から自動的に導出されます。このパラメータを使用する理由はないため、このパラメータは置き換わりません。

親トピック: Oracle Database 19cで非推奨となった初期化パラメータ

SQLNET.ENCRYPTION_WALLET_LOCATIONパラメータの非推奨

SQLNET.ENCRYPTION_WALLET_LOCATION `sqlnet.ora`パラメータは、Oracle Database 19cで非推奨にな

りました。

日付: 2019年4月

SQLNET.ENCRYPTION_WALLET_LOCATIONパラメータは、透過的データ暗号化(TDE)のソフトウェア・キーストアの場所を定義します。SQLNET.ENCRYPTION_WALLET_LOCATIONを設定するかわりにソフトウェア・キーストアの場所を構成する場合、WALLET_ROOT初期化パラメータおよびTDE_CONFIGURATION動的初期化パラメータを設定することをお薦めします。

できるだけ早くWALLET_ROOTインスタンス初期化パラメータを使用することをお薦めします。これは、インスタンスが起動されると値が読み取られるので、起動後にすべてのセッションおよびサーバーのバックグラウンド・プロセスが同じパスを共有するためです。SQLNET.ENCRYPTION_WALLET_LOCATIONパラメータが使用されている場合、異なるセッションが異なるSQLNETパラメータ値を持つ可能性があるため、誤った構成につながり、混乱が生じる可能性があります。WALLET_ROOTを使用するもう1つの理由は、これがOracle Enterprise User Securityやランスポート・レイヤー・セキュリティなどの他の機能のウォレットを配置できるディレクトリであるためです。この場所は、すべてのサーバー側ウォレットの主要な場所になる可能性があります。

関連項目

- Oracle Database Advanced Securityガイド

親トピック: Oracle Database 19cで非推奨となった初期化パラメータ

SERVICE_NAMES初期化パラメータの非推奨

Oracle Database 19c以降では、ユーザーによるSERVICE_NAMESパラメータの使用は非推奨になりました。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。

日付: 2019年11月

SERVICE_NAMESパラメータの使用は、アクティブにはサポートされなくなりました。高可用性(HA)デプロイメントでは使用しないでください。HA操作ではサービス名パラメータの使用はサポートされません。この制限には、FAN、ロード・バランス、FAILOVER_TYPE、FAILOVER_RESTORE、SESSION_STATE_CONSISTENCYなどの用途が含まれます。

サービスの管理には、SRVCTLまたはGDSCTLコマンドライン・ユーティリティ、あるいはDBMS_SERVICEパッケージを使用することをお薦めします。

親トピック: Oracle Database 19cで非推奨となった初期化パラメータ

Oracle Database 18cの動作の変更、サポート終了および非推奨

Oracle Database 18cリリースの変更の説明を確認してください。

- Oracle Database 18cアップグレード計画の動作の変更
  Oracle Database 18cへのアップグレードの計画に役立つ動作の変更を確認してください
- Oracle Database 18cでサポートが終了した機能
  アップグレード計画の一環として、このサポートが終了した機能のリストを参照してください。
- Oracle Database 18cでサポートが終了した初期化パラメータ
  このリリースのパラメータ設定の変更および代替機能については、サポートが終了した初期化パラメータのリストを確認してください。
- Oracle Streamsの最終リリース
  Oracle Database 18cは、Oracle Streamsのサポートの最終リリースです。Oracle Streamsは、Oracle Database 19c以降サポートされなくなります。

- Oracle Database 18cで非推奨となった機能
  この項にリストされている非推奨となった機能を参照して、アップグレード後に使用する代替方法を準備してください。

親トピック: Oracle Databaseの変更、サポート終了および非推奨

Oracle Database 18cアップグレード計画の動作の変更

Oracle Database 18cへのアップグレードの計画に役立つ動作の変更を確認してください

- 簡素化されたイメージベースのOracle Databaseのインストール
  Oracle Database 18c以降では、Oracle Databaseソフトウェアのインストールおよび構成が、イメージベースのインストールにより簡素化されています。
- 64文字を超えるJSONキー名の索引付けのサポート
  JSONキーを使用すると、長いキー名を使用してHASH MAP形式の構造から生成されたJSONドキュメントの検索効率が向上します。
- 既存のデータベースのアップグレードにかわるイメージ・インストール
  Oracle Database 18c以上では、既存のサービスはインストールで移行されなくなりました。Database Upgrade Assistant (DBUA)を使用してサービスを移行します。
- RPMベースのOracle Databaseのインストールについて
  Oracle Database18c以降では、RPMパッケージを使用して単一インスタンスのOracle DatabaseまたはOracle Database Instant Clientソフトウェアをインストールできます。
- Oracle Text索引のトークン制限
  Oracle Databaseリリース18c以上では、索引付けされたトークンの最大サイズがシングルバイト文字セットで255文字まで拡張されています。
- /ALL/USER/DBAユーザー・ビューおよびPL/SQL外部ライブラリの変更
  Oracle Database 18c以上では、`/USER/ALL/DBA_ARGUMENTS`ビューと`/USER/ALL/DBA_IDENTIFIERS`ビュー、およびPDBでの`LIBRARY`オブジェクトの作成方法が変更されています。
- シンボリック・リンクおよびUTL_FILE
  シンボリック・リンクで`UTL_FILE`を使用することはできません。かわりに、ディレクトリ・オブジェクトを使用してください。
- DBCAを使用したリスナーの直接登録の非推奨
  Database Configuration Assistant (DBCA)を使用したOracle DatabaseのOracle Internet Directory (OID)への登録は、Oracle Database 18cで非推奨になりました。
- INIT.ORA内のUNIFORM_LOG_TIMESTAMP_FORMATの変更
  デフォルトでは、Oracle Database 12cリリース2 (12.2)以上のタイムスタンプの書式は異なります。アラート・ログを表示するには、Oracle Databaseユーティリティである自動診断リポジトリ・コマンド・インタプリタ(ADRCI)ユーティリティを使用します。

親トピック: Oracle Database 18cの動作の変更、サポート終了および非推奨

簡素化されたイメージベースのOracle Databaseのインストール

Oracle Database 18c以降では、Oracle Databaseソフトウェアのインストールおよび構成が、イメージベースのインストールにより簡素化されています。

Oracle Database 18c以降、Oracle Databaseソフトウェアは、イメージ・ファイルとしてダウンロードおよびインストールできます。Oracleホームを配置するディレクトリにイメージ・ソフトウェアを抽出し、`runInstaller`スクリプトを実行して、Oracle Databaseインストールを開始します。詳細は、ご使用のオペレーティング・システム・プラットフォームの『*Oracle Databaseインストレーション・ガイド*』を参照してください。

ノート:

Oracle Database ホームを配置するディレクトリにイメージ・ソフトウェア(`db_home.zip`)を抽出し、`runInstaller` スクリプトを実行して、Oracle Database のインストールおよび構成を開始します。作成した Oracle ホームのディレクトリ・パスが Oracle Optimal Flexible Architecture の推奨事項に準拠することをお薦めします。

関連項目

- Oracle Databaseのインストレーション・ガイド

親トピック: Oracle Database 18cアップグレード計画の動作の変更

64文字を超えるJSONキー名の索引付けのサポート

JSONキーを使用すると、長いキー名を使用してHASH MAP形式の構造から生成されたJSONドキュメントの検索効率が向上します。

JSON検索索引で索引付けできるJSONキー名の上限が引き上げられています。Oracle Database 12cリリース2 (12.2.0.2)以降のリリースでのJSONキー名の上限は、255バイトです。以前のリリースで作成されたJSON検索索引では、64バイトを超えるキー名は索引付けされませんでした。

親トピック: Oracle Database 18cアップグレード計画の動作の変更

既存のデータベースのアップグレードにかわるイメージ・インストール

Oracle Database 18c以上では、既存のサービスはインストールで移行されなくなりました。Database Upgrade Assistant (DBUA)を使用してサービスを移行します。

既存のOracle Databaseで使用されているサービスを移行する必要がある場合は、Oracleホーム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から、DBUAを起動する必要があります。

Windowsの場合、Microsoft Transaction Serviceを新しいOracleホームに移行するには、コマンド`%ORACLE_HOME%¥bin¥oramtsctl.exe -new`も実行する必要があります

親トピック: Oracle Database 18cアップグレード計画の動作の変更

RPMベースのOracle Databaseのインストールについて

Oracle Database18c以降では、RPMパッケージを使用して単一インスタンスのOracle DatabaseまたはOracle Database Instant Clientソフトウェアをインストールできます。

RPMベースのインストールでは、インストール前チェックが実行され、データベース・ソフトウェアが展開され、展開されたソフトウェアの所有権が事前構成済ユーザーおよびグループに再割当てされ、Oracleインベントリがメンテナンスされ、単一インスタンスのOracleデータベースの作成と構成のためのOracle Databaseソフトウェアの構成に必要なすべてのroot操作が実行されます。

RPMベースのインストール・プロセスでは、インストールの最小要件が満たされていない場合にこれを検出し、これらのインストールの最小前提条件を完了するように求めます。

RPMベースのインストールでは、ソフトウェアのみのOracle Databaseインストールが実行され、Oracleホームが作成されます。Oracleホームの作成後、Oracle Database Configuration Assistant (Oracle DBCA)を使用してOracle Databaseを作成できます。

RPMベースのインストール・プロセスでは、/etc/init.d/oracledb_ORCLCDB-19cサービスの構成スクリプトを使用して、デフォルト設定でデータベースを作成できます。

親トピック: Oracle Database 18cアップグレード計画の動作の変更

Oracle Text索引のトークン制限

Oracle Databaseリリース18c以上では、索引付けされたトークンの最大サイズがシングルバイト文字セットで255文字まで拡張されています。

Oracle Databaseリリース18cより前では、SDATAセクションを除くすべてのOracle Text索引タイプで、VARCHAR2 (64 BYTE)型の表列にトークンが格納されていました。Oracle Databaseリリース18c以上では、CTXCATおよびCTXRULE索引を除くすべてのOracle Text索引タイプで、VARCHAR2 (255 BYTE)型の表列にトークンが格納されます。この変更では、索引付けされたトークンの最大サイズがシングルバイト文字セットで255文字まで拡張されています。サイズの拡張は、マルチバイト文字セットや可変長文字セットではこれより短くなります。255バイトより長いトークンは切り捨てられます。トークンが切り捨てられたとしても、トークン文字列全体の検索が妨げられることはありません。ただし、システムでは最初の255バイトが同じ2つのトークンを区別できません。

> ノート:
>
> Oracle Database リリース 18c より前では、64 バイトを超えるトークンは 64 バイトに切り捨てられていました。Oracle Database リリース 18c にアップグレードすると、トークン表が 64 バイトから 255 バイトに拡張されます。検索トークン(つまり、検索文字列の 1 語)が 64 バイトを超える検索では、64 バイトに切り捨てられたトークンを見つけることができません。この問題を回避するには、索引を再構築します。64 バイトを超える検索トークンを使用しない場合は、索引を再構築する必要はありません。

SDATAセクションでは、VARCHAR2 (249 BYTE)型の表列にトークンが格納されます。CTXCATおよびCTXRULE索引では、VARCHAR2 (64 BYTE)型の表列にトークンが格納されます。

親トピック: Oracle Database 18cアップグレード計画の動作の変更

/ALL/USER/DBA_ユーザー・ビューおよびPL/SQL外部ライブラリの変更

Oracle Database 18c以上では、/USER/ALL/DBA_ARGUMENTSビューと/USER/ALL/DBA_IDENTIFIERSビュー、およびPDBでのLIBRARYオブジェクトの作成方法が変更されています。

作業に影響する可能性のある変更を確認してください。

ALL/USER/DBA_ARGUMENTSユーザー・ビューの変更点

ARGUMENTSビュー内の行数が減りました。具体的には、最上位(DATA_LEVEL=0)の項目のみがARGUMENTSビューに格納されます。

以前のOracle Databaseリリースでは、PL/SQLコンパイラにより、PL/SQLデータ型のネストされたすべてのタイプに関するメタデータが収集されていました。DATA_LEVELは、タイプのネスト・レベルを表していました。Oracle Database 18c以上では、最上位のタイプのメタデータ(DATA_LEVEL=0)のみがARGUMENTSビューに格納されます。

例として、create-or-replaceパッケージNestedTypesExampleの変更点に注目します。

```
Type Level2Record is RECORD (Field1 NUMBER);
Type Level1Collection is TABLE of Level2Record index by binary_integer;
Type Level0Record is RECORD (Field1 Level1Collection);
Procedure NestedTypesProc (Param1 Level0Record);
```

以前のOracle Databaseリリースでは、NestedTypeProcプロシージャの最上位のタイプ(パラメータParam1、Level0Record)とともに、Level0Record内のネストされたすべてのタイプの詳細な説明が返されます。たとえば:

```
SQL> select argument_name,type_subname,position,sequence,data_level from
user_arguments where object_name='NESTEDTYPESPROC';
ARGUMENT_NAME   TYPE_SUBNAME      POSITION   SEQUENCE  DATA_LEVEL
--------------- ----------------- ---------- ---------- ---------
PARAM1          LEVEL0RECORD             1          1           0
FIELD1          LEVEL1COLLECTION         1          2           1
                LEVEL2RECORD             1          3           2
FIELD1                                   1          4           3
```

それに対し、18.1データベースで同じ問合せを使用した場合は、次の結果が返されます。

```
ARGUMENT_NAME   TYPE_SUBNAME      POSITION   SEQUENCE  DATA_LEVEL
--------------- ----------------- ---------- ---------- ---------
PARAM1          LEVEL0RECORD             1          1           0
```

Oracle Database 12c (12.1)より前のリリースでは、PL/SQLパッケージ・タイプの説明的なメタデータには、最上位のオブジェクト・タイプのメタデータにアクセスする場合と同じ方法ではアクセスできませんでした。最上位のオブジェクト・タイプとコレクションについては、ALL_TYPESに加え、関連するユーザー・ビューALL_TYPE_ATTRSおよびALL_COLL_TYPESを問い合せて、タイプのメタデータを取得できます。一方、Oracle Database 12.1より前では、レコードやパッケージ化されたコレクションなど、PL/SQLパッケージ・タイプのメタデータを取得できませんでした。関数またはプロシージャのパラメータでこれらのPL/SQLパッケージ・タイプを参照していた場合は、ARGUMENTSビューでこれらのタイプ(ネストされたタイプを含む)に関するすべてのメタデータが公開されていました。

このアプローチの問題は、深くネストされたタイプがSYS表領域で大量のメモリーを消費する可能性があることです。また、ARGUMENTSビューでタイプのメタデータを共有する方法がないため、深くネストされたタイプを持つ各パラメータには、タイプのメタデータの冗長コピーがそれぞれ個別に必要でした。ARGUMENTSビューおよびSYS表領域内のメタデータの量によっては、PL/SQLコンパイラのパフォーマンス低下など、様々な問題を生じる可能性があります。パフォーマンス低下の原因は、PL/SQLが基礎となるディクショナリ表の行を更新するのに多くの時間を要することです。

Oracle Database 12.1リリースでは、PL/SQLでのパッケージ・タイプに対するサポートが拡張され、新しいユーザー・ビューALL_PLSQL_TYPES、ALL_PLSQL_TYPE_ATTRSおよびALL_PLSQL_COLL_TYPESが追加されています。これらのビューは、名前が示すとおり、ALL_TYPESビュー・ファミリに類似しています。ただし、拡張されたPL/SQLタイプ・ビューを使用して、最上位のオブジェクト・タイプおよびコレクション・タイプのかわりに、PL/SQLパッケージ・タイプに関するメタデータを問い合せることができます。

Oracle Database 12.1ではパッケージ・タイプが追加されているため、説明的なメタデータを大量にARGUMENTSビューに挿入する必要がなくなりました。各パラメータ・タイプについてARGUMENTSビューで必要になるのは、タイプ名を含む1行のメタデータのみです。PL/SQLタイプ・ビューおよびネストされたタイプに対する問合せでは、タイプ名の完全な説明を取得できます。

OCIDescribeAny()は、ARGUMENTSビューで使用されるものと同じメタデータに基づきます。また、OCIDescribeAny()では、各パラメータ・タイプについてそれぞれ1行のみが返されます(Oracle Database 12.1で変更されるまでは一般に複数の行が返されていました)。

ALL/DBA/USER_ARGUMENTSには、新しい列タイプTYPE_OBJECT_TYPEが含まれています。TYPE_OWNER、TYPE_NAMEおよびTYPE_SUBNAMEで表されるタイプのタイプを調べるには、TYPE_OBJECT_TYPE列を使用します。候補となる値は、TABLE、VIEW、PACKAGEおよびTYPEです。

ARGUMENTSビューで引き続きALL_TYPESとそれに関連するユーザー・ビュー(ALL_TYPE_ATTRSおよびALL_COLL_TYPES)を収集する場合は、イベントをevents='10946, level 65536'に設定できます。このイベントを設定すると、ARGUMENTSビューが12.1より前のOracle Databaseリリースの動作に戻ります。つまり、DATA_LEVELを0

より大きくすることができ、タイプおよびネストされたタイプに関する説明的なメタデータがビューに含まれます。このように変更する場合は、イベントを設定した後で、影響を受けるパッケージを再コンパイルする必要があります。影響を受けるパッケージの再コンパイル時には、コンパイラによって追加のメタデータが再収集されます。また、このイベントを設定すると、OCIDescribeAny()が12.1より前のOracle Databaseリリースの動作に戻ります。

Oracle Database 12cリリース1 (12.1.0.2)以上では、引数なしでプロシージャを入力すると、ARGUMENTSビューに行は含まれません。この変更は、ARGUMENTSビューの行削減とは別に加えられたものです。Oracle Database 12.1.0.2より前では、引数なしのプロシージャもARGUMENTSビューに1行として表示されていました。

USER/ALL/DBA_IDENTIFIERSユーザー・ビューの変更点

Oracle Database 18c以上では、PL/Scopeが拡張され、PL/SQLコード内のユーザー識別子に関する追加情報が取得されるようになりました。追加情報には、識別子に適用されている制約、関数がPL/SQLのSQLビルトインであることを示すインジケータなどが含まれます。

次の列は、Oracle Database 18cのUSER/ALL/DBA_IDENTIFIERSビューに新規に追加されました。

- CHARACTER_SET: この列には、列が変数識別子宣言で使用されている場合に文字セット句の値が格納されます。候補となる値は、CHAR_CS、NCHAR_CSおよびIDENTIFIER (文字セットが別の変数識別子から派生している場合)です。
- ATTRIBUTE: この列には、%attributeが変数宣言で使用されている場合に属性値が格納されます。候補となる値は、ROWTYPE、TYPEおよびCHARSETです。
- CHAR_USED: この列には、制約が文字列長制約宣言で使用されている場合に長さ制約のタイプが格納されます。候補となる値は、CHARおよびBYTEです。
- LENGTH: この列には、文字列長制約宣言の長さ制約の値が格納されます。
- PRECISION: この列には、数値精度が変数宣言で使用されている場合にその精度が格納されます。
- PRECISION2: この列には、変数宣言で使用されている第2精度値(間隔タイプなど)が格納されます。
- SCALE: この列には、変数宣言で使用されているスケール値が格納されます。
- LOWER_RANGE: この列には、範囲制約付きの変数宣言で使用されている範囲の下限値が格納されます。
- UPPER_RANGE: この列には、範囲制約付きの変数宣言で使用されている範囲の上限値が格納されます。
- NULL_CONSTRAINT: この列は、NULL制約が変数宣言で使用されている場合に設定されます。候補となる値は、NULLまたはNOT NULLです。
- SQL_BUILTIN: PL/SQLから発行されたSQL文で使用されるSQLビルトインを識別子としている場合、この列はYESに設定されます。識別子がSQLビルトインでない場合、列はNOに設定されます。

PL/SQL外部ライブラリの変更点

Oracle Database 18c以上では、PATH_PREFIXが事前定義されたOracle Database 18c PDBにLIBRARYオブジェクトを作成する方法が変更されています。

- PATH_PREFIXが事前定義されたPDBに新しいLIBRARYオブジェクトを作成する場合、LIBRARYでDIRECTORYオブジェクトを使用する必要があります。DIRECTORYオブジェクトにより、LIBRARYオブジェクトに対してPATH_PREFIXのルールが適用されます。LIBRARYオブジェクトでDIRECTORYオブジェクトを使用しないと、PLS-1919コンパイル時エラーが発生します。
- PATH_PREFIXが事前定義されたPDBとしてデータベースをCDBに接続する場合に、DIRECTORYオブジェクトを

使用していないLIBRARYオブジェクトを使用しようとすると、ORA-65394ランタイム・エラーが発生します。LIBRARYオブジェクトは無効化されません。ただし、(ランタイム・エラーが常に発行されるのとは対照的に) LIBRARYを有効に活用するには、DIRECTORYオブジェクトを使用するようにLIBRARYオブジェクトを再作成する必要があります。

このように変更すると、ファイル・システムにおけるLIBRARYダイナミック・リンク・ライブラリ(DLL)の場所を示すPATH_PREFIXの値が考慮されるようになり、PDBでのLIBRARYオブジェクトのセキュリティと管理性が向上します。また、DIRECTORYオブジェクトを使用すると、管理者はDLLディレクトリへのアクセスを許可するユーザーを指定できます。

親トピック: Oracle Database 18cアップグレード計画の動作の変更

シンボリック・リンクおよびUTL_FILE

シンボリック・リンクでUTL_FILEを使用することはできません。かわりに、ディレクトリ・オブジェクトを使用してください。

Oracle Database 18c以降のリリースでは、UTL_FILE_DIRのシンボリック・リンク・パスがサポートされなくなりました。アップグレード後、アプリケーションでUTL_FILEを介したシンボリック・リンクを使用してデータベースを操作しようとすると、これらのリンクは失敗します。ディレクトリ・オブジェクトを使用することをお薦めします。必要に応じて、UTL_FILE内のファイル名のターゲットとなる実際のファイルを作成できます。

このサポート終了は、Oracle Data Pump、BFILEおよび外部表を含む(ただし、これに制限されない)、シンボリック・リンクを使用する以前のリリースの機能に影響する可能性があります。アップグレード後に影響を受ける機能を使用しようとして、その機能でシンボリック・リンクが使用された場合、ORA-29283: 無効なファイル操作です。 : パスがsymlinkをトラバースしていますが発生します。アップグレードの前に、修正する必要があるシンボリック・リンクを特定するのに役立つように、AutoUpgradeをanalyzeモードで実行します。シンボリック・リンクのかわりにディレクトリ・オブジェクトを使用することをお薦めします。

例10-1 UTL_FILEおよびシンボリック・リンクによるエラー・メッセージの例

UTL_FILEのアドレスを指定するシンボリック・リンクを使用するアプリケーションで、エラーが発生しました。たとえば、次のようなシンボリック・リンクを作成しようとするとします。Ia.cはシンボリック・リンク・ファイルです。

```
create or replace directory TEMP as '/home/PLSQL/TEMP';
declare
f utl_file.file_type;
begin
f := utl_file.fopen('TEMP','la.c','r');
end;
/
```

このコマンドは、次のエラーが発生して失敗します。

```
ERROR at line 1:
ORA-29283: invalid file operation
ORA-06512: at "SYS.UTL_FILE", line 536
ORA-29283: invalid file operation
ORA-06512: at line 4
```

親トピック: Oracle Database 18cアップグレード計画の動作の変更

DBCAを使用したリスナーの直接登録の非推奨

Database Configuration Assistant (DBCA)を使用したOracle DatabaseのOracle Internet Directory (OID)への登録は、Oracle Database 18cで非推奨になりました。

アップグレード時に、DBCAを使用してデータベース・ホームにリスナーを移行または登録するかわりに、Net Configuration AssistantまたはNet Managerを使用して、新しいリリースのOracleホームにLISTENER.ORAファイルを作成してからこのリ

スナーを起動します。DBCAを使用してリスナーを登録解除し、OIDに再び登録することもできます。

親トピック: Oracle Database 18cアップグレード計画の動作の変更

INIT.ORA内のUNIFORM_LOG_TIMESTAMP_FORMATの変更

デフォルトでは、Oracle Database 12cリリース2 (12.2)以上のタイムスタンプの書式は異なります。アラート・ログを表示するには、Oracle Databaseユーティリティである自動診断リポジトリ・コマンド・インタプリタ(ADRCI)ユーティリティを使用します。

スクリプトを使用してタイムスタンプ日付のアラート・ログを解析する場合は、タイムスタンプ書式のデフォルト値は`init.ora`パラメータUNIFORM_LOG_TIMESTAMP_FORMATで設定されていることに注意してください。このパラメータのデフォルト値はTRUEです。TRUEの場合、タイムスタンプ書式は「日-月-年-時刻」書式から「年-月-日-時刻」書式に変更されます。たとえば、`2017-05-17T10:00:54.799968+00:00`のようになります。

UNIFORM_LOG_TIMESTAMP_FORMATの値をFALSEに変更することで、以前のリリースで使用されていたタイムスタンプ書式に変更できます。アラート・ログではなく`log.xml`を解析するスクリプトを使用することもできます。

Oracleでは、Oracleエラーやトレース・ファイルを検索して分析できるように、自動診断リポジトリ・コマンド・インタプリタ(ADRCI)と呼ばれる専用のコマンドライン・ユーティリティを提供しており、このADRCIユーティリティを使用してエラーを管理することをお薦めしています。

たとえば、ADRCIコマンド`show alert`を使用してアラート・ログを表示できます。

```
$ oracle@user> adrci
adrci> show alert -tail -f
```

また、ADRCIでは、`show log`コマンドを使用して、問合せの述語を渡すこともできます。たとえば:

```
adrci> show log -p "message_text like '%tablespace%'"
```

ADRCIユーティリティを使用する方法の詳細は、『*Oracle Database*ユーティリティ』を参照してください。

関連項目

- Oracle Databaseユーティリティ

親トピック: Oracle Database 18cアップグレード計画の動作の変更

Oracle Database 18cでサポートが終了した機能

アップグレード計画の一環として、サポートが終了した機能をこのリストで確認してください。

- Oracle Administration Assistant for Windowsのサポート終了
  Windows用Oracle Administration Assistantツールは、Oracle Database 18cでサポートが終了しました。
- Oracle Multimedia DICOMのサポート対象外機能
  いくつかのOracle Multimedia DICOMの機能は、Oracle Database 18cでサポートが終了しました。DICOMは、Oracle SecureFilesおよびサードパーティのDICOM製品に置き換わります。
- Oracle Multimedia Java Clientクラスのサポート終了
  Oracle Multimediaプロキシ・クラスとOracle Multimediaサーブレットおよび JSPクラスのサポートが終了しました。
- Oracle XML DBのサポート対象外機能
  Oracle Database 18c以降、`DBMS_XMLSCHEMA`のスキーマ・サブプログラム、多数の`DBMS_XDB`サブプログラムおよびその他の多くのOracle XML DBスキーマ機能がサポートされなくなりました。
- ODP.NET管理対象ドライバの分散トランザクションのDLLのサポート終了
  Oracle Database 18cで`Oracle.ManagedDataAccessDTC.dll`ファイルのサポートは終了しました。

- Data Guard Broker DGMGRL ALTER構文のサポート終了
  Oracle Database 18c以上では、DGMGRLのOracle Data Guard BrokerのALTERコマンドのサポートが終了しました。
- Microsoft WindowsシステムでのCRSUSERのサポート終了
  Windowsサービス・ユーザーを変更するためのcrsuserユーティリティおよびCRSTokenメソッドは、Oracle Database 18cでサポートが終了しました。

親トピック: Oracle Database 18cの動作の変更、サポート終了および非推奨

Oracle Administration Assistant for Windowsのサポート終了

Windows用Oracle Administration Assistantツールは、Oracle Database 18cでサポートが終了しました。

Oracle Administration Assistant for Windowsは、現行のデータベースのリリースではサポートされていません。Oracle Administration Assistant for Windowsは、Windows上でデータベース管理者、オペレータ、ユーザーおよびロールを作成するツールでした。Oracle Administration Assistantでは、データベース・サービス、起動と停止設定、そしてWindowsレジストリ・パラメータの管理もできました。これに替わる機能はありません。

親トピック: Oracle Database 18cでサポートが終了した機能

Oracle Multimedia DICOMのサポート対象外機能

いくつかのOracle Multimedia DICOMの機能は、Oracle Database 18cでサポートが終了しました。DICOMは、Oracle SecureFilesおよびサードパーティのDICOM製品に置き換わります。

Digital Imaging and Communications in Medicine (DICOM)とは、放射線デバイスの接続性をサポートする医療用画像処理テクノロジです。OracleのネイティブのDICOM機能が非推奨になり、その一部がこのリリースでサポートされなくなりました。Oracle Multimedia DICOMのサポートが終了する機能は、次のとおりです。

- Oracle Multimedia DICOMプロトコル
- Oracle Multimedia DICOM中間層サポート
- WebCenter統合用のOracle Multimedia Oracle DICOMコンポーネント(DICOM/UCM)

次のOracle Multimedia DICOM機能は引き続き推奨されません。

- Oracle Multimedia ORDImageオブジェクトのDICOMサポート
- Oracle Multimedia DICOMオブジェクトおよびパッケージ

Oracle Multimedia DICOMの代替機能はありません。Oracle Multimedia DICOMのかわりに、DICOM機能を持つサードパーティ製品とともにOracle SecureFilesを使用することをお薦めします。たとえば: サードパーティのDICOM機能を使用して、メタデータ管理、DICOM画像の変換などを実行します。

親トピック: Oracle Database 18cでサポートが終了した機能

Oracle Multimedia Javaクライアント・クラスのサポート終了

Oracle Multimediaプロキシ・クラスとOracle MultimediaサーブレットおよびJSPクラスのサポートが終了しました。

Oracle Multimedia Javaクライアントの次のクラスについては、Oracle Database 18cでサポートが終了しました。

- DICOMプロキシ・クラスを含むOracle Multimediaプロキシ・クラス
- Oracle Multimediaサーブレット/JSPクラス

Oracle Database内でマルチメディア・コンテンツを管理するJavaアプリケーションを開発する場合は、JavaにPL/SQLブロックを埋め込むことをお薦めします。

親トピック: Oracle Database 18cでサポートが終了した機能

Oracle XML DBのサポート対象外機能

Oracle Database 18c以降、DBMS_XMLSCHEMAのスキーマ・サブプログラム、多数のDBMS_XDBサブプログラムおよびその他の多くのOracle XML DBスキーマ機能がサポートされなくなりました。

Oracle Database 12cリリース1 (12.1)で、PL/SQLパッケージDBMS_XDB_CONFIGが導入されました。同時に、パッケージDBMS_XDBからDBMS_XDB_CONFIGに移行したOracle XML DBのすべての構成関数、プロシージャおよび定数が非推奨になり、さらに他の一連のDBMS_XMLSCHEMA、DBMS_XDBサブプログラムおよびその他のスキーマ機能が非推奨になりました。これらのコンポーネントは、現在サポートされていません。

サポートが終了したパッケージDBMS_XMLSCHEMA内のPL/SQLサブプログラム

パッケージDBMS_XMLSCHEMA内の次のPL/SQLサブプログラムは、サポートが終了しました。

- generateSchema
- generateSchemas

これらの構造体の代用はありません。この変更に対する回避策はありません。

サポートが終了したOracle XML DBの構成関数、プロシージャおよび定数

パッケージDBMS_XDBからDBMS_XDB_CONFIGに移行したOracle XML DBの構成に関する関数、プロシージャおよび定数はすべてサポートが終了しました。DBMS_XDB_CONFIGを使用します。

次のリストに示すサブプログラムは、パッケージDBMS_XDBでのサポートが終了しました。

- ADDHTTPEXPIREMAPPING
- ADDMIMEMAPPING
- ADDSCHEMALOCMAPPING
- ADDSERVLET
- ADDSERVLETMAPPING
- ADDSERVLETSECROLE
- ADDXMLEXTENSION
- CFG_GET
- CFG_REFRESH
- CFG_UPDATE
- DELETEHTTPEXPIREMAPPING
- DELETEMIMEMAPPING
- DELETESCHEMALOCMAPPING
- DELETESERVLET
- DELETESERVLETMAPPING
- DELETESERVLETSECROLE
- DELETEXMLEXTENSION
- GETFTPPORT
- GETHTTPPORT

- GETLISTENERENDPOINT
- SETFTPPORT
- SETHTTPPORT
- SETLISTENERENDPOINT
- SETLISTENERLOCALACCESS

次の定数は、パッケージDBMS_XDBでのサポートが終了しました。

- XDB_ENDPOINT_HTTP
- XDB_ENDPOINT_HTTP2
- XDB_PROTOCOL_TCP
- XDB_PROTOCOL_TCPS

サポートが終了したOracle XQuery関数

次のOracle XQuery関数はサポートが終了しました。かわりに、対応する標準のXQuery関数を使用します。対応する関数は同じ名前を持つ関数ですが、それは名前空間接頭辞fnを使用します。

- ora:matches。かわりにfn:matchesを使用します。
- ora:replace。かわりにfn:replaceを使用します。

親トピック: Oracle Database 18cでサポートが終了した機能

ODP.NET管理対象ドライバの分散トランザクションのDLLのサポート終了

Oracle Database 18cでOracle.ManagedDataAccessDTC.dllファイルのサポートは終了しました。

Oracleは、Oracle.ManagedDataAccessDTC.dllを使用するOracle Data Provider for .NET (ODP.NET)管理対象ドライバについて、ネイティブの管理対象分散トランザクション・サポートを提供していました。.NET Framework 4.5.2で、Microsoft社は、管理対象ODP.NETで使用できる、独自のネイティブ管理対象分散トランザクション実装を導入しました。新しい.NET Frameworkでは、Oracle.ManagedDataAccessDTC.dllは不要になりました。さらに、Microsoft社では、4.5.2より前のすべての.NET Framework 4バージョンがサポートされなくなりました。Microsoft社の方針に従い、Oracle.ManagedDataAccessDTC.dllファイルのサポートは終了しました。

このサポート終了には、UseManagedDTC .NET構成ファイル・パラメータとOracle.ManagedDataAccessDTC.dllの削除が含まれます。

親トピック: Oracle Database 18cでサポートが終了した機能

Data Guard Broker DGMGRL ALTER構文のサポート終了

Oracle Database 18c以上では、DGMGRLのOracle Data Guard BrokerのALTERコマンドのサポートが終了しました。

Data Guard BrokerのDGMGRLコマンドライン・インタフェースのALTERコマンド構文は、Oracle Database 10gリリース1では非推奨で、EDIT CONFIGURATION、EDIT DATABASEおよびEDIT INSTANCEの構文に置き換えられました。

親トピック: Oracle Database 18cでサポートが終了した機能

Microsoft WindowsシステムでのCRSUSERのサポート終了

Windowsサービス・ユーザーを変更するためのcrsuserユーティリティおよびCRSTokenメソッドは、Oracle Database 18cでサポートが終了しました。

リリース12c (12.1)より前のOracle Grid Infrastructureリリースでは、Oracle Real Application Clusters (Oracle RAC)でのcrsuserユーティリティを使用した、LocalSystemからユーザーIDへのOracle Databaseサービスのデータベース・ログオン・プロパティの変更がサポートされていました。

Oracle Grid Infrastructure 12cリリース1 (12.1)では、DBホームのOracleホーム・ユーザーのシステム権限ロールが導入されました。このロールにより、crsuser機能が不要になります。また、crsuser機能は、以前はLocalSystem以外のWindowsユーザーとして実行されたユーザー定義のCRSリソースの作成にも使用されていました。ただし、Oracle Grid Infrastructure 12cリリース1 (12.1)以降のリリースでは、crsctl add wallet -type OSUSERと同じ機能が提供されます。crsuser機能は動作しなくなりました。これは開発またはサポートされなくなりました。

crsctl add wallet -type OSUSERコマンドの詳細は、*Oracle Clusterware*管理およびデプロイメントを参照してください。

関連項目

- Oracle Clusterware管理およびデプロイメント・ガイド

親トピック: Oracle Database 18cでサポートが終了した機能

Oracle Database 18cのサポートが終了した初期化パラメータ

パラメータ設定の変更および代替機能については、サポートが終了した初期化パラメータのこのリストを参照してください。

- STANDBY_ARCHIVE_DEST初期化パラメータのサポート終了
  Oracle Database 18cでは、初期化パラメータSTANDBY_ARCHIVE_DESTのサポートが削除されました。
- UTL_FILE_DIR初期化パラメータのサポート終了
  Oracle Database 18c以降では、UTL_FILE_DIRパラメータがサポート対象外になりました。かわりに、ディレクトリ・オブジェクトの名前を指定します。

親トピック: Oracle Database 18cの動作の変更、サポート終了および非推奨

STANDBY_ARCHIVE_DEST初期化パラメータのサポート終了

Oracle Database 18cでは、初期化パラメータSTANDBY_ARCHIVE_DESTがサポート対象外になりました。

Oracle Database 11gリリース2 (11.2)では、パラメータLOCALおよびREMOTEを使用したアーカイブのLOG_ARCHIVE_DEST_nの数が31個まで拡張されています。この拡張に加え、ALTERNATE属性も拡張されたことで、ローカルおよびリモート・アーカイブの可用性が高まり、アーカイブ先に障害が発生しても結果をより詳細に制御できるようになりました。これらの拡張により、STANDBY_ARCHIVE_DESTは必須ではなくなり、実質的に使用しなくなりました。

親トピック: Oracle Database 18cでサポートが終了した初期化パラメータ

UTL_FILE_DIR初期化パラメータのサポート終了

Oracle Database 18c以降では、UTL_FILE_DIRパラメータがサポート対象外になりました。かわりに、ディレクトリ・オブジェクトの名前を指定します。

UTL_FILE_DIR初期化パラメータは、V$SYSTEM_PARAMETERおよび関連するビューにリストされなくなりました。このパラメータを設定しようとすると、試行は失敗します。UTL_FILE.FOPENのLOCATIONパラメータを使用するか、またはFOPEN_NCHARのLOCATIONパラメータを使用して、オペレーティング・システム・ファイルを直接指定しようとすると、その試行も失敗します。かわりに、ディレクトリ・オブジェクトの名前を指定します。

UTL_FILEおよびその他のOracle Databaseサブシステムにディレクトリ・オブジェクトを使用するためのセキュリティ・モデルは、

明確な権限モデルがあることから安全です。ただし、明示的なオペレーティング・システム・ディレクトリの使用は、関連する権限モデルがないことから安全ではありません。UTL_FILE_DIR初期化パラメータの非推奨は、Oracle Database 12cリリース2 (12.2)で通知されました。Oracle Database 18cでは、このパラメータは現在サポートが終了しています。

サポートされていないディレクトリ・パスでのUTL_FILEパッケージのシンボリック・リンク

UTL_FILEパッケージを使用したシンボリック・リンクへのアクセスは、新しいOracle Databaseリリースでは失敗します。この問題を回避するには、シンボリック・リンクが含まれないようにディレクトリ・オブジェクトとファイル名を変更する必要があります。このサポート終了は、Oracle Data Pump、BFILEおよび外部表を含む(ただし、これに制限されない)、シンボリック・リンクを使用する以前のリリースの機能に影響する可能性があります。アップグレード後に影響を受ける機能を使用しようとして、その機能でシンボリック・リンクが使用された場合、ORA-29283: 無効なファイル操作です。 : パスがsymlinkをトラバースしていますが発生します。

親トピック: Oracle Database 18cでサポートが終了した初期化パラメータ

Oracle Streamsの最終リリース

Oracle Database 18cは、Oracle Streamsのサポートの最終リリースです。Oracle Streamsは、Oracle Database 19c以降サポートされなくなります。

Oracle Streamsは、Oracle Database 12cリリース1 (12.1)で非推奨になりました。これは、Oracle Database 12c以降のリリースで導入された機能(マルチテナント・アーキテクチャ、LONG VARCHARデータ型、長い識別子およびその他の機能を含む)をサポートしていません。Oracle GoldenGateがOracle Database用の代替ソリューションです。

親トピック: Oracle Database 18cの動作の変更、サポート終了および非推奨

Oracle Database 18cで非推奨となった機能

この項にリストされている非推奨となった機能を参照して、アップグレード後に使用する代替方法を準備してください。

> ノート:
>
> 非 CDB アーキテクチャは Oracle Database 12c で非推奨になりました。Oracle Database 19c の後のリリースではサポート対象から外れ、使用できなくなる可能性があります。

- Data GuardのMAX_CONNECTIONS属性の非推奨化
  Data GuardのREDOトランスポートのLOG_ARCHIVE_DEST_nパラメータのMAX_CONNECTIONS属性は、Oracle Database 18cで非推奨になりました。
- 拡張データ型サポート(EDS)の非推奨化
  拡張データ型サポート(EDS)は、Oracle Database 18cで非推奨になりました。
- DBMS_DATA_MININGパッケージのGET_*ファンクションの非推奨化
  Oracle Database 18c以降、DBMS_DATA_MININGのGET_*ファンクションは非推奨になりました。かわりに、Oracle Data Mining (ODM)のモデル・ディテール・ビューを使用します。
- パッケージDBMS_XMLQUERYの非推奨化
  PL/SQLパッケージDBMS_XMLQUERYは、Oracle Database 18cで非推奨になりました。かわりにDBMS_XMLGENを使用してください。
- パッケージDBMS_XMLSAVEの非推奨化
  PL/SQLパッケージDBMS_XMLSAVEは、Oracle Database 18cで非推奨になりました。かわりに

DBMS_XMLSTOREを使用してください。

- Oracle Label Securityビューで非推奨となった列
  Oracle Database 18c以降、ALL_SA_USER_LABELSビューおよびDBA_SA_USER_LABELSビューでLABELS列が非推奨になりました。
- NUMBERを使用したTrueまたはFalseのJSON戻り値の非推奨化
  Oracle Database 18c以降、JSON値のtrueまたはfalseの戻り値として、SQL NUMBERの値(1または0)を指定するオプションは非推奨になりました。
- Oracle TextのMAIL_FILTERの非推奨
  Oracle Database 18c以降、Oracle TextのMAIL_FILTERの使用は非推奨になりました。データベースに電子メールを追加する前に、電子メールを索引付け可能プレーン・テキストまたはHTMLにフィルタ処理します。
- asmcmd showversionオプションの非推奨
  Oracle Database 18c以降、asmcmd showversionのコマンド・オプションは、新しいasmcmdオプションに置き換えられました。
- Oracle TextのNEWS_SECTION_GROUPの非推奨
  Oracle Database 18c以降、Oracle TextのNEWS_SECTION_GROUPの使用は非推奨になりました。かわりに外部処理を使用します。
- Oracle Net ServicesのSDPのサポートの非推奨化
  Oracle Database 18c以降、Oracle Net Servicesのソケット・ダイレクト・プロトコル(SDP)のサポートは非推奨になりました。
- フレックス・クラスタ(ハブ/リーフ)アーキテクチャの非推奨
  Oracle Database 18c以降、リーフ・ノードはOracle Flex Clusterアーキテクチャの一部として非推奨になりました。
- PRODUCT_USER_PROFILE表の非推奨
  Oracle Database 18c以降、SQL*Plus表PRODUCT_USER_PROFILE (PUP)表は非推奨になりました。
- Oracle Multimediaの非推奨
  Oracle Database 18c以降、Oracle Multimediaは非推奨になりました。Oracle Multimediaは、Oracle Database 19cではサポートされなくなります。

親トピック: Oracle Database 18cの動作の変更、サポート終了および非推奨

Data GuardのMAX_CONNECTIONS属性の非推奨化

Data GuardのREDOトランスポートのLOG_ARCHIVE_DEST_nパラメータのMAX_CONNECTIONS属性は、Oracle Database 18cで非推奨になりました。

Oracle Database 11gリリース1 (11.1)で、REDOトランスポートに新しいストリーミング非同期モデルが導入されました。Oracle Data Guardでアーカイブ・ログ・ファイルのギャップが解消されると、MAX_CONNECTIONS属性設定を使用するメリットがなくなります。

親トピック: Oracle Database 18cで非推奨となった機能

拡張データ型サポート(EDS)の非推奨化

拡張データ型サポート(EDS)は、Oracle Database 18cで非推奨になりました。

拡張データ型サポート(EDS)機能では、ネイティブなREDOベースのサポートが欠如した特定のOracleデータ型をサポートするためのロジカル・スタンバイ用のメカニズムが提供されます。たとえば、EDSは、SDO_GEOMETRY列を持つ表のレプリケートに使用されていました。ただし、Oracle Database 12cリリース2 (12.2)以降では、EDSでサポートされているOracleデータ

型で、論理データまたはGoldenGateでネイティブにサポートされていないものはありません。この機能は現在では廃止されています。

親トピック: Oracle Database 18cで非推奨となった機能

DBMS_DATA_MININGパッケージのGET_*ファンクションの非推奨化

Oracle Database 18c以降、DBMS_DATA_MININGのGET_*ファンクションは非推奨になりました。かわりに、Oracle Data Mining (ODM)のモデル・ディテール・ビューを使用します。

Oracle Database 12cリリース1以前のリリースでは、DBMS_DATA_MININGパッケージは、データ・マイニング・アルゴリズムごとに個別のGET_MODEL_DETAILSファンクションをサポートしています。一般化線形モデル、期待値最大化、単一値分解および相関ルールについては、グローバル詳細も使用できます。多くのDBMS_DATA_MINING Get_*ファンクションがあります。たとえば:

- GET_MODEL_DETAILS
- DBMS_DATA_MINING.GET_MODEL_TRANSFORMATIONS

たとえば、デシジョン・ツリーのモデル・ディテール・ビューは、分岐情報ビュー、ノード統計ビュー、ノード説明ビューおよびコスト・マトリックス・ビューを説明しています。

Oracle Database 18c以降では、GET_MODEL_DETAILS_XMLファンクションのかわりにOracle Data Miningのモデル・ディテール・ビューを使用することをお薦めします。分岐情報ビューDM$VPmodel_nameは、ビューの接頭辞にOracle Data Miningモデルの名前を付加する意思決定ツリー階層を表します。

親トピック: Oracle Database 18cで非推奨となった機能

パッケージDBMS_XMLQUERYの非推奨化

PL/SQLパッケージDBMS_XMLQUERYは、Oracle Database 18cで非推奨になりました。かわりにDBMS_XMLGENを使用してください。

DBMS_XMLQUERYは、データベースからXMLTypeへの変換機能を提供します。DBMS_XMLQUERYではなくDBMS_XMLGENを呼び出すことをお薦めします。DBMS_XMLGENはCで作成されてカーネルにコンパイルされるため、優れたパフォーマンスが得られます。

親トピック: Oracle Database 18cで非推奨となった機能

パッケージDBMS_XMLSAVEの非推奨化

PL/SQLパッケージDBMS_XMLSAVEは、Oracle Database 18cで非推奨になりました。かわりにDBMS_XMLSTOREを使用してください。

DBMS_XMLSAVEパッケージは、Oracle XML SQL Utilityに含まれています。これは、オブジェクト・リレーショナル表のXML文書に対するデータの挿入、更新および削除に使用されます。DBMS_XMLSAVEではなくDBMS_XMLSTOREを呼び出すことをお薦めします。DBMS_XMLSTOREはCで作成されてカーネルにコンパイルされるため、優れたパフォーマンスが得られます。

たとえば: DBMS_XMLSAVEを使用しないようにするために、以前のリリースのOracle DatabaseでDBMS_XMLSAVEを呼び出すために使用していたラッパー・ファンクションまたはプロシージャを作成して、DBMS_XMLSTOREの呼出しに変更できます。または、シノニムを作成できます。

たとえば: DBMS_XMLSAVEを使用しないようにするために、以前のリリースのOracle DatabaseでDBMS_XMLSAVEを呼び出すために使用していたラッパー・ファンクションまたはプロシージャを作成して、DBMS_XMLSTOREの呼出しに変更できます。または、シノニムを作成できます。

```
CREATE OR REPLACE PUBLIC SYNONYM DBMS_XMLSAVE FOR DBMS_XMLSTORE;
GRANT EXECUTE ON DBMS_XMLSAVE TO PUBLIC;
```

親トピック: Oracle Database 18cで非推奨となった機能

Oracle Label Securityビューで非推奨となった列

Oracle Database 18c以降、ALL_SA_USER_LABELSビューおよびDBA_SA_USER_LABELSビューでLABELS列が非推奨になりました。

表10-2 Oracle Label Securityビューで非推奨となった列

データ・ディクショナリ・ビュー	非推奨となった列
ALL_SA_USER_LABELS	LABELS
ALL_SA_USERS	USER_LABELS
DBA_SA_USER_LABELS	LABELS
DBA_SA_USERS	USER_LABELS

LABELSとUSER_LABELSの列の情報は重複します。この情報は、これらのデータ・ディクショナリ・ビューの他の列に表示されます。

親トピック: Oracle Database 18cで非推奨となった機能

NUMBERを使用したTrueまたはFalseのJSON戻り値の非推奨化

Oracle Database 18c以降、JSON値のtrueまたはfalseの戻り値として、SQL NUMBERの値(1または0)を指定するオプションは非推奨になりました。

Oracle Database 12cリリース1 (12.1)では、返される列の型としてNUMBERを指定する関数を含むJSONデータのサポートが提供されました。NUMBERを指定するオプションは非推奨です。true/false問合せのJSONデータの出力としてNUMBERを指定するのではなく、JSONのブール値に対して返されたデフォルトのSQL値を使用して、trueまたはfalseとして文字列を指定できます。数値を必要とするアプリケーションを使用している場合は、ブールのJSON値をSQL VARCHAR2値として返し、その値をテストし、そのテストの結果としてSQL NUMBERの値を返すことができます。

親トピック: Oracle Database 18cで非推奨となった機能

Oracle TextのMAIL_FILTERの非推奨

Oracle Database 18c以降、Oracle TextのMAIL_FILTERの使用は非推奨になりました。データベースに電子メールを追加する前に、電子メールを索引付け可能プレーン・テキストまたはHTMLにフィルタ処理します。

MAIL_FILTERは廃止された電子メール・プロトコルRFC-822に基づいています。最新の電子メール・システムはRFC-822をサポートしていません。これに替わる機能はありません。

親トピック: Oracle Database 18cで非推奨となった機能

asmcmd showversionオプションの非推奨

Oracle Database 18c以降、asmcmd showversionのコマンド・オプションは、新しいasmcmdオプションに置き換えられました。

コマンドasmcmd showversion --softwarepatchのかわりに、新しいオプションasmcmd showpatches -lを使用します。コマンドasmcmd showversion --releasepatchのかわりに、新しいオプションasmcmd showversion --activeを使用します。

親トピック: Oracle Database 18cで非推奨となった機能

Oracle TextのNEWS_SECTION_GROUPの非推奨

Oracle Database 18c以降、Oracle TextのNEWS_SECTION_GROUPの使用は非推奨になりました。かわりに外部処理を使用します。

USENET投稿に索引を付ける場合、Oracle Text内でBASIC_SECTION_GROUPまたはHTML_SECTION_GROUPを使用するように投稿を前処理します。USENETは、商用ではあまり使用されません。

USENETは現在、重大な目的はあまり使用されません。このセクション・グループ・タイプを使用して実行する索引処理は廃止されています。

親トピック: Oracle Database 18cで非推奨となった機能

Oracle Net ServicesのSDPのサポートの非推奨化

Oracle Database 18c以降、Oracle Net Servicesのソケット・ダイレクト・プロトコル(SDP)のサポートは非推奨になりました。

かわりにTCPを使用することをお薦めします。

親トピック: Oracle Database 18cで非推奨となった機能

フレックス・クラスタ(ハブ/リーフ)アーキテクチャの非推奨

Oracle Database 18c以降、リーフ・ノードはOracle Flex Clusterアーキテクチャの一部として非推奨になりました。

障害発生時の再構成時間を短縮化するためにOracle Clusterwareスタックを継続的に改善してきたことにより、オンプレミスまたはクラウドで、お客様のニーズを満たすクラスタの実装にリーフ・ノードが不要になりました。

親トピック: Oracle Database 18cで非推奨となった機能

PRODUCT_USER_PROFILE表の非推奨

Oracle Database 18c以降、SQL*Plus表PRODUCT_USER_PROFILE (PUP)表は非推奨になりました。

PRODUCT_USER_PROFILE (PUP)表でのみ使用するのは、SQL*Plusの製品レベル・セキュリティを制御するメカニズムを提供するためです。Oracle Database 18c以降、このメカニズムは関連しません。このSQL*Plusの製品レベル・セキュリティ機能は、Oracle Database 19cでは使用できません。Oracle Databaseの設定を使用してデータを保護し、すべてのクライアント・アプリケーション間で一貫したセキュリティを確保することをお薦めします。

親トピック: Oracle Database 18cで非推奨となった機能

Oracle Multimediaの非推奨

Oracleデータベース18c以降では、Oracle Multimediaは非推奨になりました。Oracle Multimediaは、Oracle Database 19cではサポートされなくなります。

SecureFiles LOBにマルチメディア・コンテンツを格納し、イメージ処理および変換にオープン・ソースまたはPictionなどのサードパーティ製品を使用することをお薦めします。Oracle Locatorは、Oracle Multimediaの非推奨による影響を受けません。

親トピック: Oracle Database 18cで非推奨となった機能

Oracle Database 12cリリース2 (12.2)の動作の変更、サポート終了および非推奨

Oracle Database 12cリリース2 (12.2)の変更の説明を確認してください。

- Oracle Database 12cリリース2 (12.2)での動作の変更
  Oracle Database 12cリリース2 (12.2)へのアップグレードの計画に役立つ動作の変更を確認してください
- Oracle Database 12cリリース2 (12.2)でサポートが終了した機能
  アップグレード計画の一環として、このサポートが終了した機能のリストを参照してください。
- Oracle Database 12cリリース2 (12.2)でサポートが終了した初期化パラメータ
  このリリースのパラメータ設定の変更および代替機能については、サポートが終了した初期化パラメータのこのリストを参照してください。
- Oracle Database 12cリリース2 (12.2)で非推奨となった機能
  この項にリストされている非推奨となった機能を参照して、アップグレード後に使用する代替方法を準備してください。
- Oracle Database 12cリリース2 (12.2)で非推奨となった初期化パラメータ
  パラメータ設定の変更および代替機能を理解するには、12.2リリースで非推奨となったパラメータを参照してください。これらのパラメータは、以降のリリースで削除される可能性があります。

親トピック: Oracle Databaseの変更、サポート終了および非推奨

Oracle Database 12cリリース2 (12.2)での動作の変更

Oracle Database 12cリリース2 (12.2)へのアップグレードの計画に役立つ動作の変更を確認してください

- Oracle Database 12cリリース2 (12.2)の初期化パラメータのデフォルトの変更
  Oracle Database 12cリリース2 (12.2)の、初期化パラメータのデフォルト設定のこの変更リストを参照してください。
- Database Upgrade Assistant (DBUA)の拡張機能と変更点
  Oracle Database 12cリリース2 (12.2)にはDBUAに対するいくつかの拡張機能が含まれており、また、一部の機能は削除または変更されています。
- Oracle Data Guard Brokerおよびローリング・アップグレードの拡張機能
  Oracle Database 12cリリース2 (12.2)以上では、Oracle Data Guard Brokerにローリング・アップグレードを支援する機能がさらに含まれます。
- Oracle DatabaseのデフォルトのSGA権限の変更について
  Oracle Database 12cリリース2 (12.2.0.1)以降、デフォルトでは、システム・グローバル領域(SGA)への読取りおよび書込み権限はOracleソフトウェア・インストール所有者のみに限定されます。
- ネットワーク・アクセス制御リストおよびOracle Database 12cへのアップグレード
  ネットワーク・アクセス制御リスト(ACL)は、12cでReal Application Security ACLとして実装され、既存のACLは、アップグレード中にXML DB ACLから移行されて名前が変更されます。
- パラレル・アップグレード・ユーティリティのバッチ・スクリプト
  Oracle Database 12cリリース2以降のリリースでは、コマンドラインのバッチ・スクリプトを使用してパラレル・アップグレード・ユーティリティを実行できます。`catupgrd.sql`は配布されなくなりました。
- 統合監査のAUDIT_ADMINおよびAUDIT_VIEWERロールの変更
  場合によっては、アップグレードの前に`AUDIT_ADMIN`および`AUDIT_VIEWER`ロールを名前変更するか、削除する必要があります。

- Oracleバッチ更新のバッチ・サイズ設定の無効化
  Oracleバッチ更新設定はOracle Database 12cリリース2 (12.2)では無効になりました。かわりにJDBCバッチ処理を使用します。
- Oracle管理タイプに依存する表のアップグレードについて
  Oracle Database 12cリリース2 (12.2)から、表に設定する-TオプションをREAD ONLYに指定して、パラレル・アップグレード・ユーティリティを実行できます。
- パスワードの大/小文字の区別およびORA-1017無効なユーザー名またはパスワード
  Oracle Database 12cリリース2 (12.2)のデフォルトの認証プロトコルは12 (排他モード)です。このプロトコルでは認証に大/小文字のパスワードが必要です。以前のリリースのパスワード・バージョンがある場合は、オプションを確認してください。
- Oracle Fleet Patching and Provisioningを使用したOracle Grid Infrastructureのデプロイについて
  Oracle Fleet Patching and Provisioning(Oracle FPP)は、Oracleホームのプロビジョニングおよびメンテナンスのためのソフトウェア・ライフサイクル管理方法です。Oracleフリート・パッチ適用およびプロビジョニングにより、データベース、クラスタ、およびユーザー定義ソフトウェア・タイプの標準的な操作環境の一括デプロイメントおよびメンテナンスが可能になります。
- Zero Data Loss Recovery Applianceリリース12.1のバックアップに関する使用上の制限
  Zero Data Loss Recovery Applianceリリース12.1は、Oracle Database 12c リリース2 (12.2)を使用している保護されたデータベース・クライアントからのバックアップをサポートしていません。
- クライアントおよびフォアグラウンド・サーバーのプロセス・メモリーの変更
  最適化およびパフォーマンスを向上させるために、Oracle Databaseクライアントとサーバーのプロセス・メモリーの要件は以前のリリースよりも大きい値になっています。

親トピック: Oracle Database 12cリリース2 (12.2)の動作の変更、サポート終了および非推奨

Oracle Database 12cリリース2 (12.2)での初期化パラメータのデフォルトの変更

Oracle Database 12cリリース2 (12.2)の、初期化パラメータのデフォルト設定のこの変更リストを参照してください。

OPTIMIZER_ADAPTIVE_PLANSおよびOPTIMIZER_ADAPTIVE_STATISTICS

OPTIMIZER_ADAPTIVE_FEATURE関数は、次の2つの新しいパラメータで置き換えられました: OPTIMIZER_ADAPTIVE_PLANS、OPTIMIZER_ADAPTIVE_STATISTICS。

OPTIMIZER_ADAPTIVE_PLANSは適応計画を制御します。デフォルトではTRUEに設定されます。TRUEに設定された場合、このパラメータは問合せ実行時に収集された統計に基づいて、代替選択肢で構築される実行計画を指定します。

OPTIMIZER_ADAPTIVE_STATISTICSは適応統計を制御します。デフォルトではFALSEに設定されます。TRUEに設定された場合、オプティマイザはSQL実行計画を改善するためにSQL文の解析時に収集された適応統計を補充して、データベースで収集された統計を増やします。一部の問合せは、ベース表統計だけを使用するには複雑すぎます。オプティマイザは最適なSQL実行計画をより正確に決定するために、適応統計を補充してそれらを増やします。

SQL92_SECURITY初期化パラメータのデフォルトはTRUE

SQL標準では、セキュリティ管理者には次の権限が必要であると定めています: WHERE句またはSET句により表列値を参照するUPDATE文またはDELETE文の実行において、表に対するSELECT権限の所持を実行ユーザーに要求する権限。SQL92_SECURITYでは、ユーザーに、UPDATE文やDELETE文などを実行するために、SELECTオブジェクト権限を付与する必要があるかどうかを指定します。

Oracle Database 12cリリース2 (12.2)以降、このパラメータのデフォルト設定はFALSEからTRUEに変更されます。

このパラメータをTRUEに設定した場合、ユーザーは削除または更新するオブジェクトに対するSELECT権限が必要です。

関連項目

- 『Oracle Databaseリファレンス』

親トピック: Oracle Database 12cリリース2 (12.2)での動作の変更

Database Upgrade Assistant (DBUA)の機能拡張と変更点

Oracle Database 12cリリース2 (12.2)にはDBUAに対するいくつかの拡張機能が含まれており、また、一部の機能は削除または変更されました。

お客様の要望に応え、また、機能向上のために、Database Upgrade Assistant (DBUA)には新しい機能とコード拡張が含まれています。また、以前のリリースの一部の機能は削除されました。

DBUAの新機能

DBUAにはOracle Database 12cリリース2 (12.2)の次の新機能が含まれています。

- 選択的PDBのプラグイン・アップグレード: 以前のリリースで作成されたPDBをリリース12.2マルチテナント・アーキテクチャのCDB環境に接続し、リリース12.2のCDBホームから起動したDBUAを使用してPDBをアップグレードできます。

  CDBからPDBを切断し、CDBおよびCDBに接続されたPDBをアップグレードしてから、以前のリリースのPDBに接続してDBCAを使用してそれらをアップグレードできます。

- 優先度に基づくPDBアップグレード: 優先度の高いPDBを先にアップグレードするように、PDBのアップグレードに優先度を設定できます。
- 再試行と無視の機能: エラーを修正してからアップグレードを再試行するか、または、特定のエラーを無視してアップグレードを継続することを選択できます。
- 中断と継続機能: アップグレードを中止し、後でアップグレードを続行できます。
- スタンドアロンの前提条件チェック: 新しい`-executePrereqs`オプションを指定してDBUAを実行し、いつでもアップグレードの前提条件を確認できます。
- データベース移動中のリスナーの構成: データベースの移動操作中に新しいリスナーを持つデータベースを構成できます。
- ロギング機構の向上: DBUAにタイム・スタンプ付きのログが導入されました。
- パフォーマンスの向上: DBUAには、アップグレード・プロセス中のインスタンス再起動の回数を低減するコード拡張が含まれています。
- 拡張されたエラー報告: すべてのDBUAエラーはエラー・コード接頭辞DBTを使用して報告され、また、すべてのエラーはメッセージ・ウィンドウに表示されるかわりにプログレス・ページにリストで報告されます。

DBUAの削除された機能

以前のリリースで使用可能だったDBUAの次の機能は、Oracle Database 12cリリース2 (12.2)で削除されました。

- データ・ファイルの移動: アップグレード中のデータ・ファイルの移動はできなくなりました。
- アップグレード中のデータベースの名前変更: アップグレード中のOracle Databaseの名前変更はサポートされなくなりました。
- 並列度選択はDBUAから削除: ユース・ケースに応じてデフォルトの並列度が計算されます。

    - アップグレード: DBUAを使用するデフォルトの並列度は、パラレル・アップグレード・ユーティリティで手動アップグレードに使用される値と同じです。ただし、アップグレード操作時に、使用するコアの数を指定することにより、このデフォルト値をオーバーライドできます。
    - 再コンパイル: オブジェクトの再コンパイルについてのデフォルトの並列度は、手動アップグレードで使用される`utlrp`スクリプトにより決まります。
- 再コンパイルの並列度はデフォルトで、アップグレード並列度と同じ値です。
- 診断および監査保存先の変更不可: 診断および監査の保存先の変更は、DBUAコマンドラインのオプション`-initParam`を使用してのみ、可能です。
- リモートDBUAがサポート対象外に: 以前のリリースでは、DBUAはWindowsプラットフォームでOracle Database リモート・アップグレードをサポートするオプションがありました。この機能はサポート対象外になりました。

親トピック: Oracle Database 12cリリース2 (12.2)での動作の変更

Oracle Data Guard Brokerおよびローリング・アップグレードの拡張機能

Oracle Database 12cリリース2 (12.2)以上では、Oracle Data Guard Brokerにローリング・アップグレードを支援する追加機能が含まれます。

Oracle Data Guard Brokerでは、現在、Oracle Active Data Guardのローリング・アップグレードがサポートされます。Oracle Active Data Guardのローリング・アップグレードは、Oracle Database 12cリリース1 (12.1)で導入されました。これにより、多くの手動ステップを単純なPL/SQLパッケージ(DBMS_ROLLING)で自動化することで、一時ロジカル・データベースのローリング・アップグレード処理の実行が容易になります。データベースのローリング・アップグレードが容易になるだけでなく、自動化処理により信頼性も大幅に向上します。Oracle Data Guard Brokerでは、現在、DGMGRLコマンドライン・インタフェースからOracle Active Data Guardのローリング・アップグレードを制御できます。また、ブローカのサポートによって、REDOトランスポート宛先の設定などの作業が透過的に処理され、ローリング・アップグレード処理が大幅に簡略化されます。

Oracle Database 12cリリース2 (12.2)以上のリリースでは、DBMS_ROLLING PL/SQLパッケージを使用してローリング・アップグレードを実行する場合、ブローカを無効にする必要がなくなりました。また、ブローカによって、ローリング・アップグレードの適用時期がレポートされ、そのステータスが追跡されるようになりました。ステータス情報は、DGMGRLコマンドのSHOW CONFIGURATIONおよびSHOW DATABASEの出力に表示されます。

Oracle Data Guard Brokerを使用してデータベースのローリング・アップグレードを管理することで、本番環境に変更を導入する際の停止時間とリスクが最小化され、アップグレード処理が簡略化されます。

関連項目

- Oracle Data Guard Broker

親トピック: Oracle Database 12cリリース2 (12.2)での動作の変更

Oracle DatabaseのデフォルトのSGA権限の変更について

Oracle Database 12cリリース2 (12.2.0.1)以降、デフォルトでは、システム・グローバル領域(SGA)への読取りおよび書込み権限はOracleソフトウェア・インストール所有者のみに制限されます。

以前のリリースでは、Oracleインストール所有者アカウントとOSDBAグループのメンバーの両方が、共有メモリーに対するアクセス権を持っていました。Oracle Database 12cリリース2 (12.2)以降のリリースでは、デフォルトでOracleインストール所有者アカウントにアクセス権が制限されるように変更され、以前の構成よりセキュリティが強化されています。ただし、この変更によって、Oracleインストール所有者アカウントへのアクセス権を持たないDBAは、データベースを管理できなくなる可能性があります。

Oracle Databaseの初期化パラメータALLOW_GROUP_ACCESS_TO_SGAは、Oracle Databaseインストール所有者アカウント(Oracleドキュメントの例ではoracle)のみがデータベースのシステム・グローバル領域(SGA)への読取りおよび書込みを行えるか、またはOSDBAグループのメンバーがSGAの読取りを行えるかどうかを決定します。Oracle Database 12cリリース2 (12.2)以降のリリースでは、このパラメータのデフォルト値はFALSEで、Oracle Databaseインストール所有者のみがSGAへの読取りおよび書込み権限を持っています。SGAに対するグループ・アクセス権は、デフォルトで削除されています。この変更は、すべてのLinuxおよびUNIXプラットフォームに影響します。

OSDBAグループのメンバーがSGAに対する読取り権限を必要とする場合は、初期化パラメータALLOW_GROUP_ACCESS_TO_SGAの設定をFALSEからTRUEに変更します。SGAへのアクセスをoracleユーザー・アカウントに制限する、デフォルトの権限を受け入れることをお薦めします。

関連項目

- 『Oracle Databaseリファレンス』

親トピック: Oracle Database 12cリリース2 (12.2)での動作の変更

ネットワーク・アクセス制御リストおよびOracle Database 12cへのアップグレード

ネットワーク・アクセス制御リスト(ACL)は、12cでReal Application Security ACLとして実装され、既存のACLは、アップグレード中にXML DB ACLから移行されて名前が変更されます。

Oracle Databaseの12cリリース1 (12.1)以上のリリースへのアップグレード中に、Oracle Databaseのネットワーク・アクセス制御は、Real Application Securityのアクセス制御リスト(ACL)を使用して実装されます。XDBの既存のACLは、アップグレード中に移行されます。DBMS_NETWORK_ACL_ADMIN PL/SQLパッケージの既存のAPIおよびカタログ・ビューは、非推奨です。これらの非推奨のビューは、Oracle Database 12cで同等の新しい項目に置き換えられました。

Oracle Database 12cリリース1 (12.1)以上では、DBMS_NETWORK_ACL_ADMIN.APPEND_HOST_ACEを使用してホストACLにアクセス制御エントリ(ACE)を追加することで、ネットワーク権限を付与できます。既存のホストACLが存在しないホストにACEを追加すると、新しいホストACLが暗黙的に作成されます。ホストACLが存在する場合、ACEはその既存のACLに追加されます。

Real Application Security ACLの変更による影響

アップグレード中に次の変更が行われます。

- 既存のネットワークACLは、Oracle Database 11g XML DBからOracle Database 12c Real Application Securityに移行されます。既存のACLの権限は、すべてこの移行で保持されます。
- 既存のACLの名前は変更されます。

アップグレード前に実行する必要があること

- アップグレード前に既存のネットワークACLを確認してください。
- 既存のネットワークACLおよび権限(DBA_NETWORK_ACLSおよびDBA_NETWORK_ACL_PRIVILEGES)を中間ステージング表に保持してください。既存の権限を表に保持することで、自動移行が失敗した場合や、アップグレードをロールバックする場合に、それらをリストアできます。

関連項目

- 『Oracle Databaseセキュリティ・ガイド』
- 『Oracle Databaseリファレンス』

親トピック: Oracle Database 12cリリース2 (12.2)での動作の変更

パラレル・アップグレード・ユーティリティのバッチ・スクリプト

Oracle Database 12cリリース2以降のリリースでは、コマンドラインのバッチ・スクリプトを使用してパラレル・アップグレード・ユーティリティを実行できます。`catupgrd.sql`は配布されなくなりました。

Oracle Database 12cリリース2 (12.2)以降では、LinuxおよびUnixではシェル・コマンド`dbupgrade`を使用して、また、Microsoft Windowsでは`dbupgrade.com`を使用して、コマンドラインから直接パラレル・アップグレード・ユーティリティ(`catctl.pl`)を実行できます。これらのシェル・スクリプトによって、アップグレード・バイナリ・ホームから`catctl.pl`スクリプトがコールされます。これらのスクリプトは、デフォルト値を使用して実行するか、Perlプロンプトから`catctl.pl`を実行する際に使用するものと同じ入力パラメータを使用して実行できます。

ノート:

Oracle Databaseリリース2 (12.2)用に公開されているこの通知の情報は廃止されています。Oracle Databaseをアップグレードするためのすべての手動スクリプト・メソッドは、Oracle Database 19c以降では非推奨であり、Oracle Database 23cではサポートされなくなりました。すべてのリリースで、Oracleは、AutoUpgradeユーティリティを使用してアップグレードを実行することをお薦めします。この変更についてRoy Swongerの話を聞き、AutoUpgradeの開発の背景にある歴史についてMike Dietrichのページを読んで、AutoUpgradeを使用して独自のアップグレードをテストする方法を示すDaniel Overby Hansonのビデオを見て、Oracle Database 19cにアップグレードするためのヒッチハイク・ガイドをご覧ください。

関連項目

- 今すぐデータベースをアップグレード: Oracle Database 23cでのDBUAの非推奨と今後のサポート終了
- Oracle Database 19cにアップグレードするためのヒッチハイク・ガイド

親トピック: Oracle Database 12cリリース2 (12.2)での動作の変更

統合監査のAUDIT_ADMINおよびAUDIT_VIEWERロールの変更

場合によっては、アップグレードの前に`AUDIT_ADMIN`および`AUDIT_VIEWER`ロールを名前変更するか、削除する必要があります。

Oracle Database 12cでは、統合監査を使用する場合、アップグレードに影響する2つのAUDSYSロール(`AUDIT_ADMIN`および`AUDIT_VIEWER`)がOracle Database 11gリリース2 (11.2.0.4)以下のリリースに存在する可能性があります。これらのロールは変更されたため、Oracle Database 12cリリース1 (12.1)以上にアップグレードする前に、これらの以前のリリースのユーザーまたはユーザー・ロールを削除する必要があります。

`AUDIT_ADMIN`および`AUDIT_VIEWER`ユーザーまたはロールをOracle Database 12cリリース1 (12.1)で作成した場合、それらのユーザーまたはロールを削除する必要はありません。

次の両方の条件を満たす場合にのみ、`AUDSYS`スキーマと`AUDIT_ADMIN`ロールおよび`AUDIT_VIEWER`ロールを削除します。

- アップグレードの対象となるバージョンがOracle Database 12cリリース1 (12.1)より前である。
- `AUDSYS`という名前のカスタム・スキーマを作成した

この要件による影響を受けるが、これらのAUDSYSロールを削除できない場合、`UNIFIED_AUDIT_TRAIL`ビューを選択し、同様の定義を使用して独自の表を作成し、その表を使用して統合監査データをバックアップしてください。以前のリリースのデータベースにダウングレードする機能を保持する場合は、この手順も実行することをお薦めします。

データベースでそれらの名前を使用しないことをお薦めします。これらのユーザーまたはロールが存在する場合は、Oracle Database 12にアップグレードする前に、必要に応じて、それらの名前を変更するか削除してください。

参照:

データベースのセキュリティに関する権限およびロール認証の構成の詳細は、『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 12cリリース2 (12.2)での動作の変更

Oracleバッチ更新のバッチ・サイズ設定の無効化

Oracleバッチ更新設定はOracle Database 12cリリース2 (12.2)では無効になりました。かわりにJDBCバッチ処理を使用します。

Oracleバッチ更新は、Oracle Database 12c リリース1 (12.1)で非推奨となりました。Oracle Database 12cリリース2 (12.2)以降、Oracleバッチ更新はオペレーション・コードなし(no-op)になりました。つまり、Oracle Database 12cリリース2 (12.2)のJDBCドライバを使用してアプリケーションでOracleバッチ更新を実装すると、指定したバッチ・サイズが設定されず、バッチ・サイズが1になるということです。バッチがこの設定の場合、アプリケーションは一度に1行ずつを処理します。Oracle Database 12cリリース2 (12.2)のJDBCドライバを使用する場合は、標準のJDBCバッチを使用することを強くお薦めします。

親トピック: Oracle Database 12cリリース2 (12.2)での動作の変更

Oracle管理タイプに依存する表のアップグレードについて

Oracle Database 12cリリース2 (12.2)から、表に設定する-TオプションをREAD ONLYに指定して、パラレル・アップグレード・ユーティリティを実行できます。

-Tオプションを指定してパラレル・アップグレード・ユーティリティを実行した場合、Oracle管理オブジェクトを含まないすべての表領域はREAD ONLYに設定されます。これらの表をREAD ONLYに設定すると、データベースをアップグレードする前にバックアップが必要なデータ量を減らすことができます。

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手動でそれらの表をアップグレードする必要があります。

アップグレードが完了した後でOracle管理タイプに依存する表をアップグレードするには、スクリプト`utluptabdata.sql`を実行して、アップグレード中にREAD ONLYに設定された表領域の表に対してALTER TABLE UPGRADEコマンドを実行します。

Oracle Database 12cリリース2以降では、ALTER TYPE文の動作も変更されています。アクセス可能な表領域に従属表がある場合は、それが自動的に新しいバージョン・タイプにアップグレードされます。従属表がREAD ONLY表領域にある場合は、自動的にアップグレードされません。`utluptabdata.sql`スクリプトを実行して、アップグレード中にREAD ONLY表領域状態に設定された表をアップグレードします。`utluptabdata.sql`スクリプトを実行する必要があるのは、-Tオプションを指定してパラレル・アップグレード・ユーティリティを実行し、アップグレードする場合だけです。

ノート:

これらの表を READ ONLY に設定すると、表領域内のすべての表は、ユーザーの更新権限レベルに関係なく更新できません。たとえば、SYSDBA として接続しているユーザーは、アプリケーション・データを変更できません。

関連項目

- Oracle管理タイプに依存する表のアップグレード

- 読取り専用およびオフラインの表領域を使用したアップグレードの実行

親トピック: Oracle Database 12cリリース2 (12.2)での動作の変更

パスワードの大/小文字の区別およびORA-1017無効なユーザー名またはパスワード

Oracle Database 12cリリース2 (12.2)のデフォルトの認証プロトコルは12 (排他モード)です。このプロトコルでは認証に大/小文字のパスワードが必要です。以前のリリースのパスワード・バージョンがある場合は、オプションを確認してください。

Oracle Database 12cリリース2 (12.2)以降、SQLNET.ORAパラメータALLOWED_LOGON_VERSION_SERVERのデフォルト値は12に変更されました。このパラメータはOracle Databaseリリースにではなく、サーバーに使用されるログオン認証プロトコルのことです。

デフォルトで、Oracleでは大/小文字を区別しないパスワードに基づく認証はもうサポートせず、新しいパスワード・バージョン(11Gおよび12C)のみ使用できます。大/小文字を区別しない10Gのパスワード・バージョンはもう生成されません。

次の条件が当てはまる場合は、12.2へのアップグレード後にデータベースにログインできないアカウントがある可能性があります。

- 以前のOracle Databaseリリースで作成されたユーザー・アカウントがあるサーバーをアップグレードしている。
- 以前のリリースで作成されたユーザー・アカウントでは、以前のリリースの認証プロトコルの大/小文字を区別しないパスワード・バージョン(10Gパスワード・バージョンなど)を使用している。
- 以前のリリースのユーザー・アカウントでパスワードを再設定していない。
- サーバーは、SEC_CASE_SENSITIVE_LOGONをFALSEに設定して構成されているため、10Gの大/小文字の区別のないパスワード・バージョンを持つユーザーだけを認証できる。

10Gパスワード・バージョンが必要なアカウントがある場合は、そのパスワードを使用しているアカウントがデータベースでロックされないように、排他モードではなく、より許容度の高い認証プロトコルに変更できます。

> ノート:
>
> 排他モードで実行している間は、大/小文字を区別しないパスワードに基づく認証はサポートされません。Oracle Database 12c リリース 2 (12.2)のデフォルトの認証プロトコルは排他モードです。Oracle では次の条件付きでのみ、大/小文字の区別のない認証をサポートしています。
>
> - サーバーが排他モード以外のモードで実行されている
> - 10G パスワード・バージョンがある

10Gパスワード・バージョンのみを使用したアカウントがあるサーバーのオプション

10Gパスワード・バージョンを使用したアカウントを有効にするには、Oracle Database 12cリリース2 (12.2)にアップグレードした後で次の手順を実行します。

1. 管理者としてログインします。
2. SQLNET.ORAファイルを編集して、SQLNET.ALLOWED_LOGON_VERSION_SERVER設定をデフォルトの12から、11以下に変更します。たとえば:
   SQLNET.ALLOWED_LOGON_VERSION_SERVER=11

より許容度の高いSQLNET.ALLOWED_LOGON_VERSION_SERVER設定に変更したら、ユーザーがパスワードを変更

する必要が発生するように、そうしたユーザーのパスワードを期限切れにします。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 12cリリース2 (12.2)での動作の変更

Oracle Fleet Patching and Provisioningを使用したOracle Grid Infrastructureのデプロイについて

Oracle Fleet Patching and Provisioning(Oracle FPP)は、Oracleホームのプロビジョニングとメンテナンスのためのソフトウェア・ライフサイクル管理方法です。Oracleフリート・パッチ適用およびプロビジョニングにより、データベース、クラスタ、およびユーザー定義ソフトウェア・タイプの標準的な操作環境の一括デプロイメントおよびメンテナンスが可能になります。

ノート:

Oracle Grid Infrastructure 19c 以上では、以前は高速ホーム・プロビジョニング(RHP)と呼ばれていた機能が Oracle Fleet Patching and Provisioning(Oracle FPP)になりました。

Oracleフリート・パッチ適用およびプロビジョニングにより、クラスタのインストールと、Oracle Grid Infrastructure、Oracle RestartおよびOracle Databaseホームのプロビジョニング、パッチ適用、スケール変更およびアップグレードを行うことができます。サポートされているバージョンは、11.2、12.1、12.2、18cおよび19cです。アプリケーションおよびミドルウェアもOracleフリート・パッチ適用およびプロビジョニングを使用してプロビジョニングできます。

Oracle Fleet Patching and Provisioningは、次のいずれかのモードで使用できるOracle Grid Infrastructureのサービスです。

- セントラルOracle Fleet Patching and Provisioningサーバー

  Oracle Fleet Patching and Provisioningサーバーは、ゴールド・イメージという標準化されたイメージを格納および管理します。ゴールド・イメージは、データ・センターの任意の数のノードにデプロイできます。デプロイされたホームに新しいクラスタおよびデータベースを作成し、それらを使用して、既存のインストールのパッチ適用、アップグレードおよびスケール変更を行うことができます。

  Oracle Fleet Patching and Provisioningサーバーでは、次のタイプのインストールを管理できます。

  - Oracle Fleet Patching and Provisioningサーバー自体をホストするクラスタ上のソフトウェア・ホーム。
  - Oracle Grid Infrastructure 12cリリース2 (12.2)、18cおよび19cを実行しているOracle Fleet Patching and Provisioningクライアント。
  - Oracle Grid Infrastructure 11gリリース2 (11.2)および12cリリース1 (12.1)を実行しているインストール。
  - Oracle Grid Infrastructureなしで実行しているインストール。

  Oracle Fleet Patching and Provisioningサーバーは、新規インストールをプロビジョニングし、既存のインストールに変更を加える必要なく、既存のインストールを管理できます。Oracle Fleet Patching and Provisioningサーバーにより、自動的にピア・サーバー間でゴールド・イメージを共有して、地理的に分散したデータ・センターを持つ企業をサポートできます。

- Oracle Fleet Patching and Provisioningクライアント

  Oracle Fleet Patching and Provisioningクライアントは、Oracle Fleet Patching and Provisioningサーバーから管理することも、クライアント自体でコマンドを実行することで直接管理することもできます。Oracle Fleet

Patching and Provisioningクライアントは、Oracle Grid Infrastructureに組み込まれているサービスであり、Oracle Grid Infrastructure 12cリリース2 (12.2)以上のリリースで使用できます。Oracle Fleet Patching and Provisioningクライアントは、Oracle Fleet Patching and Provisioningサーバーからゴールド・イメージを取得し、ポリシーに基づいて新しいイメージをアップロードし、メンテナンス操作をそれ自体に適用できます。

Oracleフリート・パッチ適用およびプロビジョニング

Oracle Fleet Patching and Provisioningを使用したOracleソフトウェアのデプロイには、次の利点があります。

- デプロイされるソフトウェアのゴールド・イメージおよび系統の管理により、標準化および高レベルの自動化が可能になります。
- アクティブ・データベースまたはクラスタを中断させることなく、アウトオブプレースでイメージ(ゴールド・イメージと呼ばれます)として新しいホームをデプロイすることで、停止時間を最小限に抑えられます。
- データベース・バージョンおよびデプロイメント・モデル間で一貫性のある単純なAPIを使用して起動される自動化が提供されるため、メンテナンスが簡単になります。
- 組込みの検証と、操作をテストするテスト実行モードにより、メンテナンス・リスクが低くなります。
- 予期せぬ問題が発生した場合にコマンドを再開または再起動できるため、メンテナンス操作のリスクが低くなります。
- 次の機能により、パッチ適用およびアップグレードの影響が最小化、また通常は解消されます。
    - 追加のノードまたは外部ストレージを必要とせずにデプロイメント内で完全に実行される完全自動化アップグレードによる、停止時間ゼロのデータベース・アップグレード。
    - ローリング・パッチ適用中のデータベース・セッションおよびOJVMのアダプティブ管理。
    - 一括デプロイメントの管理オプション。
- デプロイメントおよびメンテナンス操作により、自動化されたワークフローに環境固有のアクションを含めるようにカスタマイズできます。

関連項目

- Oracle Clusterware管理およびデプロイメント・ガイド

参照:

Oracle Fleet Patching and Provisioningサーバーおよびクライアントの設定、およびOracle Grid InfrastructureおよびOracle Databaseホームのプロビジョニングおよびパッチ適用のためのゴールド・イメージの作成および使用の詳細は、『*Oracle Clusterware*管理およびデプロイメント・ガイド』を参照してください。

親トピック: Oracle Database 12cリリース2 (12.2)での動作の変更

Zero Data Loss Recovery Applianceリリース12.1のバックアップに関する使用上の制限

Zero Data Loss Recovery Applianceリリース12.1は、Oracle Database 12c リリース2 (12.2)を使用している保護されたデータベース・クライアントからのバックアップをサポートしていません。

Zero Data Loss Recovery Applianceリリース12.2 (リカバリ・アプライアンス)は、保護されたリリース12.2データベース・クライアントからのバックアップをサポートしていません。

データベースをリカバリ・アプライアンスにバックアップする場合は、リカバリ・アプライアンスをリリース12.2にアップグレードするまで、

データベースをリリース12.2にアップグレードしないことをお薦めします。

親トピック: Oracle Database 12cリリース2 (12.2)での動作の変更

クライアントおよびフォアグラウンド・サーバーのプロセス・メモリーの変更

最適化およびパフォーマンスを向上させるために、Oracle Databaseクライアントとサーバーのプロセス・メモリーの要件は以前のリリースよりも大きい値になっています。

Oracle Databaseのすべてのリリースには、新機能が含まれています。拡張されたデータベースの機能に対して最適なパフォーマンスを提供するために、Oracle Databaseクライアントで拡張できます。また、Oracle Databaseサーバーはいずれかのリリースから次のリリースまで拡張できます。メモリー要件の増加は、プラットフォームによって異なる場合があります。

アップグレード計画の一部として、新しいOracle Databaseリリースに存在する可能性のあるメモリー要件の拡張を確認して決定します。たとえば、Oracle Database 11gリリース2 (11.2)と比較すると、一部のプラットフォーム上のOracle Database 12cはクライアントごとに5MBのメモリー増加、サーバーごとに10MBの増加を行うことができます。

親トピック: Oracle Database 12cリリース2 (12.2)での動作の変更

Oracle Database 12cリリース2 (12.2)でサポートが終了した機能

アップグレード計画の一環として、サポートが終了した機能をこのリストで確認してください。

- アドバンスト・レプリケーションのサポート終了
  Oracle Database 12cリリース2 (12.2)から、Oracle Databaseのアドバンスト・レプリケーション機能のサポートが終了しました。
- OCRおよび投票ファイルのファイル・システムへの直接配置のサポート終了
  Oracle ASMへのファイルの配置が優先され、OCRおよび投票ディスク・ファイルの共有ファイル・システムへの配置のサポートは終了しました。
- JPublisherのサポート終了
  Oracle JPublisherのすべての機能は、Oracle Database 12cリリース2 (12.2.0.1)でサポート対象外となりました。
- preupgrd.sqlおよびutluppkg.sqlのサポート終了
  `preupgrd.sql`および`utluppkg.sql`スクリプトは、アップグレード前情報ツール(`preupgrade.jar`)によって置き換えられました。
- トランザクション・ガード用Oracle Data Provider for .NET APIのサポート終了
  ここにリストされたトランザクション・ガード用アプリケーション・プログラミング・インタフェース(API)は、Oracle Database 12cリリース2 (12.2)でサポート対象外になりました。
- Oracle Database 12cリリース2 (12.2)でのビューのサポート終了
  このトピックにリストされているビューは、Oracle Database 12cリリース2 (12.2)でサポートが終了しました。
- Oracle Database内のSQLJのサポート
  Oracle Database 12cリリース2 (12.2)以降、サーバー側のSQLJコードの実行はサポート対象外になります。
- 一部のXML DB機能のサポート終了
  Oracle Database 12cリリース2 (12.2)以降、ここにリストしたXML DB機能はサポート対象外になります。

親トピック: Oracle Database 12cリリース2 (12.2)の動作の変更、サポート終了および非推奨

アドバンスト・レプリケーションのサポート終了

Oracle Database 12c リリース2 (12.2)から、Oracle Databaseのアドバンスト・レプリケーション機能のサポートが終了

しました。

Oracle Databaseのアドバンスト・レプリケーション機能は、全面的にサポートが終了しました。この機能のサポート終了には、この機能に関連するすべての機能(マルチマスター・レプリケーション、更新可能なマテリアライズド・ビュー、デプロイメント・テンプレートなど)が含まれます。読取り専用のマテリアライズド・ビューは、引き続き、ベーシック・レプリケーションでサポートされます。

アドバンスト・レプリケーションの使用は、Oracle GoldenGateに置き換えることをお薦めします。

親トピック: Oracle Database 12cリリース2 (12.2)でサポートが終了した機能

OCRおよび投票ファイルのファイル・システムへの直接配置のサポート終了

Oracle ASMへのファイルの配置が優先され、OCRおよび投票ディスク・ファイルの共有ファイル・システムへの配置のサポートは終了しました。

Oracle Grid Infrastructure 12cリリース2 (12.2)以上では、Oracle ClusterwareファイルがOracle Automatic Storage Management (Oracle ASM)によって管理されるようになり、Oracle Clusterwareファイルの配置(共有ファイル・システムでのOracle Cluster Registry (OCR)や投票ファイルの直接配置)がサポート対象外になりました。Oracle Clusterwareファイルは共有ファイル・システムに直接配置できません。サポート対象の共有ファイル・システム(ネットワーク・ファイル・システムまたは共有クラスタ・ファイル・システム)をネイティブ・ディスク・デバイスのかわりに使用する必要がある場合は、Oracle Grid Infrastructureをインストールする前に、Oracle Clusterwareファイルをホストするために使用する予定のサポート対象のネットワーク・ファイル・システム上に、Oracle ASMディスクを作成する必要があります。これにより、Oracle ASMディスク・グループでOracle ASMディスクを使用して、Oracle Clusterwareファイルを管理できます。

使用しているOracle Databaseファイルが共有ファイル・システムに保存されている場合は、Oracle ASMストレージに移動せずに、引き続き同じファイル・システムを使用できます。

親トピック: Oracle Database 12cリリース2 (12.2)でサポートが終了した機能

JPublisherのサポート終了

Oracle JPublisherのすべての機能は、Oracle Database 12cリリース2 (12.2.0.1)でサポート対象外となりました。

次の代替機能の使用をお薦めします。

- 引き続きWebサービス・コールアウトを使用するには、Webサービス・コールアウト・ユーティリティのかわりとなるOracle JVM Webサービス・コールアウト・ユーティリティを使用することをお薦めします。
- 他のJPublisher自動化機能(ユーザー定義SQLタイプまたはSQLタイプのマッピング、PL/SQLパッケージのラッピング、その他類似機能など)を置き換えるには、開発者がSQLJプリコンパイラを使用して事前コンパイルする、Java STRUCTクラスを構築する、または他の、事前構成されたオプションを使用するなど、明示的なステップを採用することをお薦めします。

関連項目

- https://support.oracle.com/rs?type=doc&id=1937939.1

参照:

JDeveloperの非推奨とサポート終了の詳細は、My Oracle Supportノート1937939.1を参照してください。

親トピック: Oracle Database 12cリリース2 (12.2)でサポートが終了した機能

preupgrd.sqlおよびutluppkg.sqlのサポート終了

`preupgrd.sql`および`utluppkg.sql`スクリプトは、アップグレード前情報ツール(`preupgrade.jar`)によって置き換えられました。

Oracle Database 12cリリース2 (12.2)以上では、アップグレード前情報ツールのスクリプト`preupgrd.sql`および`utluppkg.sql`は、Oracle Databaseリリースの一部として提供されなくなりました。これらのファイルは、両方ともアップグレード前情報ツールの`preupgrade.jar`によって置き換えられました。

`preupgrade.jar`アップグレード前情報ツールは、Oracle Database 12cリリース2 (12.2)に付属しています。このスクリプトは、置き換えられたスクリプトと同じ機能を持っています。これは、Oracle Database 12cリリース2 (12.2)への直接のアップグレードがサポートされるOracle Databaseリリースとともにインストールされた Java Development Kit (JDK)を使用して実行できます。

ノート:

アップグレード前情報ツールは AutoUpgrade ユーティリティに置き換えられ、これが Oracle Database 11.2.0.4 以降のリリースからのアップグレードに役立ちます。使用可能な最新バージョンをダウンロードします。

関連項目

- My Oracle Supportノート2485457.1

親トピック: Oracle Database 12cリリース2 (12.2)でサポートが終了した機能

トランザクション・ガード用Oracle Data Provider for .NET APIのサポート終了

ここにリストされたトランザクション・ガード用アプリケーション・プログラミング・インタフェース(API)は、Oracle Database 12cリリース2 (12.2)でサポート対象外になりました。

トランザクション・ガード用の次のOracle Data Provider for .NETアプリケーション・プログラミング・インタフェースは、Oracle Database 12cリリース2 (12.2)で、サポート対象が終了します。

- `OracleLogicalTransactionStatus`クラス
- `OracleConnection.GetLogicalTransactionStatus`メソッド
- `OracleConnection.LogicalTransactionId`プロパティ
- `OracleConnection.OracleLogicalTransaction`プロパティ
- `OracleLogicalTransaction.DataSource`プロパティ
- `OracleLogicalTransaction.GetOutcome()`メソッド
- `OracleLogicalTransaction.GetOutcome(`文字列、文字列、文字列`)`メソッド
- `OracleLogicalTransaction.UserId`プロパティ

親トピック: Oracle Database 12cリリース2 (12.2)でサポートが終了した機能

Oracle Database 12cリリース2 (12.2)でのビューのサポート終了

このトピックにリストされているビューは、Oracle Database 12cリリース2 (12.2)でサポートが終了しました。

次のビューを使用しているSQL文は、すべて修正してください。

DBA_REGISTERED_MVIEW_GROUPSビュー

V$REPLPROPビュー

V$REPLQUEUEビュー

親トピック: Oracle Database 12cリリース2 (12.2)でサポートが終了した機能

Oracle Database内のSQLJのサポート

Oracle Database 12cリリース2 (12.2)以降、サーバー側のSQLJコードの実行はサポート対象外になります。

クライアント側SQLJを使用してサポートされます。ただし、データベース環境でのストアド・プロシージャ、関数、トリガーの実行など、サーバー側のSQLJの使用はサポートされていません。

親トピック: Oracle Database 12cリリース2 (12.2)でサポートが終了した機能

一部のXML DB機能のサポート終了

Oracle Database 12cリリース2 (12.2)以降、ここにリストしたXML DB機能はサポート対象外になります。

次の機能のサポートが終了します。

- パッケージ`oracle.xdb.dom`のJavaクラス
- Oracle XPath関数`ora:instanceof`。かわりにXQuery演算子`instance of`を使用してください。
- Oracle XPath関数`ora:instanceof-only`。かわりにXMLスキーマ属性`xsi:type`を使用してください。
- XMLTypeに対する関数ベースの索引。かわりに構造化コンポーネントではXMLIndexを使用してください。
- Oracle XQuery関数`ora:view`。かわりにXQuery関数`fn:collection`を使用してください。
- PL/SQLプロシージャ`DBMS_XDB_ADMIN.CreateRepositoryXMLIndex`
- PL/SQLプロシージャ`DBMS_XDB_ADMIN.XMLIndexAddPath`
- PL/SQLプロシージャ`DBMS_XDB_ADMIN.XMLIndexRemovePath`
- PL/SQLプロシージャ`DBMS_XDB_ADMIN.DropRepositoryXMLIndex`
- XMLスキーマの注釈(属性)`csx:encodingType`
- ハイブリッドXMLType記憶域のCLOB部分(オブジェクト・リレーショナル記憶域内に埋込みのCLOBデータの索引)

親トピック: Oracle Database 12cリリース2 (12.2)でサポートが終了した機能

Oracle Database 12cリリース2 (12.2)でサポートが終了した初期化パラメータ

パラメータ設定の変更および代替機能については、サポートが終了した初期化パラメータのこのリストを参照してください。

GLOBAL_CONTEXT_POOL_SIZE初期化パラメータ

`GLOBAL_CONTEXT_POOL_SIZE`初期化パラメータは削除され、このリリースでサポートが終了しました。

`GLOBAL_CONTEXT_POOL_SIZE`は、グローバル・アプリケーション・コンテキストの格納および管理用に、SGAに割り当てるメモリー量を指定していました。このパラメータのデフォルト値はNULLでした。このパラメータはOracle Database 10gリリース2 (10.2)で非推奨となっていました。

MAX_ENABLED_ROLES初期化パラメータ

MAX_ENABLED_ROLES初期化パラメータは削除され、このリリースでサポートが終了しました。

このパラメータの代替機能はありません。Oracle Databaseは、Oracle Database 10gリリース2 (10.2)以降、このパラメータを使用していません。

OPTIMIZER_ADAPTIVE_FEATURES初期化パラメータ

OPTIMIZER_ADAPTIVE_FEATURES初期化パラメータは削除され、このリリースでサポートが終了しました。

このパラメータの機能は2つの新しいパラメータに置き換わっています。OPTIMIZER_ADAPTIVE_PLANSのデフォルト値はTRUEです。TRUEに設定された場合、このパラメータは問合せ実行時に収集された統計に基づいて代替の実行計画を指定します。OPTIMIZER_ADAPTIVE_STATISTICSはデフォルトでFALSEに設定されています。TRUEに設定された場合、オプティマイザはSQL実行計画を改善するためにSQL文の解析時に収集された適応統計を補充して、データベースで収集された統計を増やします。

PARALLEL_AUTOMATIC_TUNING初期化パラメータ

PARALLEL_AUTOMATIC_TUNING初期化パラメータは削除され、このリリースでサポートが終了しました。

PARALLEL_AUTOMATIC_TUNING初期化パラメータは、パラレル処理を制御するパラメータのデフォルト値を決定していました。これは、Oracle Database 10gリリース2 (10.2)で非推奨となっていました。

PARALLEL_IO_CAP_ENABLED初期化パラメータ

PARALLEL_IO_CAP_ENABLED初期化パラメータは、Oracle Databaseによるデフォルトの並列度の制限を、サポートされるI/Oシステム以下のレベルに設定するかどうかを決定していました。このパラメータは、Oracle Databaseリリース11.2で非推奨となっていました。このパラメータの機能は、PARALLEL_DEGREE_LIMITパラメータによって置き換えられました(このパラメータがIOに設定されている場合)。

PARALLEL_SERVER初期化パラメータ

PARALLEL_SERVER初期化パラメータは削除され、このリリースでサポートが終了しました。

PARALLEL_SERVER初期化パラメータは、Oracle Parallel Serverモードでデータベースを起動するために使用されていました。このパラメータは、Oracle9i Databaseリリース1 (9.0.1)で非推奨になりました。Oracle Parallel Serverは、CLUSTER_DATABASE初期化パラメータを使用するOracle Real Application Clustersで置き換えられました。

PARALLEL_SERVER_INSTANCES初期化パラメータ

PARALLEL_SERVER_INSTANCES初期化パラメータは削除され、このリリースでサポートが終了しました

PARALLEL_SERVER_INSTANCES初期化パラメータは、Oracle Parallel Serverモードの構成済インスタンスの数を指定していました。このパラメータは、Oracle9i Databaseリリース1 (9.0.1)で非推奨になりました。Oracle Parallel Serverは、CLUSTER_DATABASE_INSTANCES初期化パラメータを使用するOracle Real Application Clustersで置き換えられました。

USE_INDIRECT_DATA_BUFFERS初期化パラメータ

初期化パラメータUSE_INDIRECT_DATA_BUFFERSは、このリリースで削除され、サポートが終了しました。

このパラメータは32ビット・プラットフォームのVery Large Memory機能を有効にするために使用されていました。これらのプラットフォームはサポートされなくなりました。

関連項目

- 『Oracle Databaseリファレンス』

Oracle Database 12cリリース2 (12.2)で非推奨となった機能

この項にリストされている非推奨となった機能を参照して、アップグレード後に使用する代替方法を準備してください。

- ALTER TYPE REPLACEの非推奨
  Oracle Database 12cリリース2 (12.2.0.1)以降、ALTER TYPEのREPLACE句は非推奨になりました。
- configToolAllCommandsスクリプトの非推奨
  インストール後チェック・スクリプトconfigToolAllCommandsはOracle Database 12cリリース1 (12.1)で非推奨になりました。
- DBMS_DEBUGパッケージの非推奨
  Oracle Database 12cリリース2 (12.2)では、DBMS_DEBUGパッケージの使用は非推奨です。DBMS_DEBUG_JDWPの使用をお薦めします。
- インテリジェント・データ配置(IDC)の非推奨
  インテリジェント・データ配置は、Oracle Database 12cリリース2 (12.2)で非推奨となりました。
- CONTINUOUS_MINEオプションの非推奨
  Oracle Database 12cリリース2 (12.2.0.1)からは、LogMiner CONTINUOUS_MINEオプションが非推奨になりました。
- 非CDBアーキテクチャの非推奨
  非CDBアーキテクチャはOracle Database 12cで非推奨になりました。Oracle Database 19cの後のリリースではサポート対象から外れ、使用できなくなる可能性があります。
- Oracle Administration Assistant for Windowsの非推奨
  Oracle Administration Assistant for WindowsはOracle Database 12cリリース2 (12.2)で非推奨になりました。
- Oracle Data Provider for .NETのPromotableTransaction設定の非推奨
  Oracle Data Provider for .NETの`PromotableTransaction`設定は、不要になったため非推奨となりました。
- oracle.jdbc.OracleConnection.unwrap()の非推奨
  Oracle Database 12cリリース2 (12.2)以上では、Javaパッケージ`oracle.jdbc.OracleConnection.unwrap()`は非推奨となりました。
- oracle.jdbc.rowsetパッケージの非推奨
  Oracle Database 12cリリース2 (12.2)以上では、Java `oracle.jdbc.rowset`パッケージは非推奨となりました。
- oracle.sql.DatumWithConnectionクラスの非推奨
  `oracle.sql.DatumWithConnection`を拡張する`oracle.sql`クラスはOracle Database 12cリリース2 (12.2)で非推奨になり、`oracle.jdbc`拡張機能にかわりました。
- Oracle Multimedia Java APIの非推奨
  Oracle Multimedia Java APIは、Oracle Database 12cリリース2では非推奨になっています。
- Oracle MultimediaのDICOMサポートの非推奨
  Oracle Database 12cリリース2 (12.2)以降では、Oracle Multimedia DICOM機能が非推奨になっています。
- Multimedia SQL/MM Still Image標準サポートの非推奨
  Oracle Database 12cリリース2 (12.2)以上では、Oracle Multimedia SQL/MM Still Image標準サポートは非推奨です。
- Unicode照合アルゴリズム(UCA) 6.1照合の廃止

Oracle Database 12cリリース2以降、Unicode照合アルゴリズム(UCA) 6.1照合は廃止されています。

- UNIFIED_AUDIT_SGA_QUEUE_SIZEの非推奨
  Oracle Database 12cリリース2以降、初期化パラメータUNIFIED_AUDIT_SGA_QUEUE_SIZEは非推奨になりました。
- VERIFY_FUNCTIONおよびVERIFY_FUNCTION_11Gの非推奨
  `VERIFY_FUNCTION`および`VERIFY_FUNCTION_11G`パスワード検証機能は、以前のリリースのより脆弱なパスワード制限を強化しているため、非推奨になりました。
- V$MANAGED_STANDBYの非推奨
  V$MANAGED_STANDBYビューはOracle Database 12cリリース2 (12.2.0.1)で非推奨になりました。新しいビューV$DATAGUARD_PROCESSの使用をお薦めします。
- 一部のXML DB関数の非推奨
  Oracle Database 12cリリース2 (12.2)から、このトピックでリストされているオプションは非推奨になりました。
- Oracle XML Databaseの非推奨となった機能
  これらの機能は、Oracle Database 12cリリース1では非推奨になり、将来のリリースでサポートが終了する可能性があります。

親トピック: Oracle Database 12cリリース2 (12.2)の動作の変更、サポート終了および非推奨

ALTER TYPE REPLACEの非推奨

Oracle Database 12cリリース2 (12.2.0.1)以降、ALTER TYPEのREPLACE句は非推奨になりました。

かわりに、ALTER TYPEメソッドのADDおよびDROP、またはALTER TYPEメソッドADDの使用をお薦めします。

関連項目

- Oracle Database PL/SQL言語リファレンス

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

configToolAllCommandsスクリプトの非推奨

インストール後チェック・スクリプトconfigToolAllCommandsはOracle Database 12cリリース1 (12.1)で非推奨になりました。

`configToolAllCommands`スクリプトは、インストール後のOracle製品を構成するためにレスポンス・ファイル・モードで実行します。別のパスワード・レスポンス・ファイルを使用します。Oracle Database 12cリリース2 (12.2)以上では、`configToolAllCommands`は非推奨です。今後のリリースでサポートが終了する予定です。

インストール・プロセスの一環として、インストール後のチェックを入手できるようになりました。Oracleではオプション-`executeConfigTools`を指定して、Oracle DatabaseまたはOracle Grid Infrastructureインストーラを実行することをお薦めしています。インストール時に作成されるものと同じレスポンス・ファイルを使用して、インストール後の構成を実行できます。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

DBMS_DEBUGパッケージの非推奨

Oracle Database 12cリリース2 (12.2)では、DBMS_DEBUGパッケージの使用は非推奨です。DBMS_DEBUG_JDWPの使用をお薦めします。

以前のリリースではPL/SQLにDBMS_DEBUGパッケージが含まれ、社内ツールやサードパーティのツールでPL/SQLプログラム

のデバッグが可能でした。DBMS_DEBUGパッケージではブレークポイントの設定、変数値の取得などのAPIを提供しています。この機能は、複数のリリースでDBMS_DEBUG_JDWPパッケージにより提供されてきました。DBMS_DEBUG_JDWPは同等のPL/SQLデバッグ機能を提供し、Javaストアド・プロシージャによりサーバー側のJava (OJVM)への呼出しまたはそこからの呼出しの際に、PL/SQLルーチンのシームレスなデバッグを可能にします。

関連項目

- 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 12cリリース2 (12.2)で非推奨となった機能

インテリジェント・データ配置(IDC)の非推奨

インテリジェント・データ配置は、Oracle Database 12cリリース2 (12.2)で非推奨となりました。

インテリジェント・データ配置により、最高のパフォーマンスを実現するためにOracle ASMディスク上のディスク・リージョンを指定できます。ディスク・リージョン設定を使用すると、アクセス頻度の高いデータを、より高速でバンド幅も大きい一番外側の(ホット)トラックに確実に配置できます。また、アクセス・パターンが似ているファイルは物理的に近くに配置され、待機時間が短縮されます。異なるホット・リージョンまたはコールド・リージョンに、プライマリ・エクステントおよびミラー・エクステントを配置することもできます。

この機能は、Oracle Database 12cリリース2 (12.2)で非推奨となりました。

関連項目

- Oracle Automatic Storage Management管理者ガイド

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

CONTINUOUS_MINEオプションの非推奨

Oracle Database 12c リリース2 (12.2.0.1)からは、LogMiner CONTINUOUS_MINEオプションが非推奨になりました。

LogMiner `CONTINUOUS_MINE`オプションは、下位互換性のために引き続きサポートされています。ただし、使用を停止することをお薦めします。かわりとなる機能はありません。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

非CDBアーキテクチャの非推奨

非CDBアーキテクチャはOracle Database 12cで非推奨になりました。Oracle Database 19cの後のリリースではサポート対象から外れ、使用できなくなる可能性があります。

CDBアーキテクチャの使用をお薦めします。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Oracle Administration Assistant for Windowsの非推奨

Oracle Database 12cリリース2 (12.2)では、Oracle Administration Assistant for Windowsの使用は非推奨です。

Oracle Administration Assistant for Windowsは、Windows上でデータベース管理者、オペレータ、ユーザーおよびロールを作成するツールです。また、データベース・サービス、起動と停止設定、そしてWindowsレジストリ・パラメータの管理もできます。

Oracle Administration Assistant for Windowsを使用するかわりに、ネイティブのWindows管理ツールを使用してください

さい。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Oracle Data Provider for .NETのPromotableTransaction設定の非推奨

Oracle Data Provider for .NETの`PromotableTransaction`設定は、不要になったため非推奨となりました。

昇格可能なトランザクションそのものは非推奨ではありません。この特定の設定だけが非推奨です。

Oracle Data Provider for .NETのレジストリ設定`PromotableTransaction`は、アプリケーションでトランザクションをローカルとして維持するか、またはすべての単一接続トランザクションをローカルとして開始してから、2番目の接続の登録時にトランザクションを分散型に昇格するかどうかを指定します。これは、昇格可能なトランザクションの概念です。

PromotableTransaction設定は、Oracle Database 12cリリース2 (12.2)で非推奨となりました。昇格可能なトランザクションを使用しない理由はありません。デフォルト値の`promotable`を受け入れることをお薦めします。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

oracle.jdbc.OracleConnection.unwrap()の非推奨

Oracle Database 12cリリース2 (12.2)以上では、Javaパッケージ`oracle.jdbc.OracleConnection.unwrap()`は非推奨となりました。

Javaパッケージ`oracle.jdbc.OracleConnection.unwrap()`は、Oracle Database 12cリリース2以上のリリースで非推奨となりました。このパッケージの代替機能はありません。

アプリケーションに含まれるこのJDBCメソッドは、標準のJavaメソッドに置き換えることをお薦めします。

関連項目

- https://support.oracle.com/rs?type=doc&id=2024500.1

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

oracle.jdbc.rowsetパッケージの非推奨

Oracle Database 12cリリース2 (12.2)以上では、Java `oracle.jdbc.rowset`パッケージは非推奨となりました

この機能にかわる標準のJDBC RowSetパッケージを使用することをお薦めします。

関連項目

- 『Oracle Database JDBC開発者ガイド』
- https://support.oracle.com/rs?type=doc&id=2024500.1

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

oracle.sql.DatumWithConnectionクラスの非推奨

`oracle.sql.DatumWithConnection`を拡張する`oracle.sql`クラスはOracle Database 12cリリース2 (12.2)で非推奨になり、`oracle.jdbc`拡張機能にかわりました。

以前のリリースのOracle Databaseには、`oracle.sql`と`oracle.jdbc`の両方のJavaパッケージで特定のタイプの拡張機能およびパフォーマンス強化を提供するOracle JDBCドライバが含まれていました。Oracle Database 12cリリース2 (12.2)以降、`oracle.sql.DatumWithConnection`を拡張する`oracle.sql`クラスは非推奨になりました。`oracle.jdbc`拡張機能は、引き続きサポートされます。

たとえば、次に、非推奨となった`oracle.sql`クラスの一部のリストを示します。

- ARRAY
- BFILE
- BLOB
- CLOB
- OPAQUE
- REF
- STRUCT

アプリケーションに含まれる、`oracle.sql.DatumWithConnection`を拡張する`oracle.sql`クラスは、標準のJavaタイプ、または`oracle.jdbc`拡張機能で置き換えることをお薦めします。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Oracle Multimedia Java APIの非推奨

Oracle Multimedia Java APIは、Oracle Database 12c リリース2では非推奨になっています。

次のJava APIは、Oracle Database 12c リリース2 (12.2)では非推奨であり、将来のリリースではサポートされなくなる可能性があります。

- Oracle Multimedia Java API
- Oracle Multimedia Servlets and JSP Java API
- Oracle Multimedia DICOM Java API
- Oracle Multimedia中間層Java API

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Oracle MultimediaのDICOMのサポートの非推奨

Oracle Database 12cリリース2 (12.2)以降では、Oracle Multimedia DICOM機能が非推奨になっています。

Oracle Databaseには、DICOMサポートを代替する機能はありません。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Multimedia SQL/MM Still Image標準サポートの非推奨

Oracle Database 12cリリース2 (12.2)以上では、Oracle Multimedia SQL/MM Still Image標準サポートは非推奨です。

画像処理操作の場合、Oracle Multimedia開発者は新しいORD_IMAGE PL/SQLパッケージ、またはORDImageメソッドを呼び出すことができます。

画像照合の場合、Oracle Database開発者はOpenCVなどのオープン・ソース・パッケージを使用できます。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Unicode照合アルゴリズム(UCA) 6.1照合の廃止

Oracle Database 12cリリース2以降、Unicode照合アルゴリズム(UCA) 6.1照合は廃止されています。

Unicode照合アルゴリズム(UCA) 6.1照合(UCA0610_*)は廃止されています。今後のリリースではサポート対象外になり、使用できなくなる可能性があります。多言語データのソートには、サポート対象の最新バージョンのUCA照合を使用することをお薦めします。

関連項目

- Oracle Databaseグローバリゼーション・サポート・ガイド

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

UNIFIED_AUDIT_SGA_QUEUE_SIZEの非推奨

Oracle Database 12cリリース2以降、初期化パラメータUNIFIED_AUDIT_SGA_QUEUE_SIZEは非推奨になりました。

UNIFIED_AUDIT_SGA_QUEUE_SIZEパラメータは非推奨となり、このパラメータの値はもう使用できなくなります。ただし、このパラメータは後方互換性のため、現在は保留されています。

統合監査の詳細は、『*Oracle Database*セキュリティ・ガイド』を参照してください。

関連項目

- 『Oracle Databaseセキュリティ・ガイド』

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

VERIFY_FUNCTIONおよびVERIFY_FUNCTION_11Gの非推奨

`VERIFY_FUNCTION`および`VERIFY_FUNCTION_11G`パスワード検証機能は、以前のリリースのより脆弱なパスワード制限を強化しているため、非推奨になりました。

関数`ORA12C_VERIFY_FUNCTION`および`ORA12C_STRONG_VERIFY_FUNCTION`の使用をお薦めします。これらの関数は、より強力でより最新のパスワード検証制限を強化しています。

参照:

『Oracle Databaseセキュリティ・ガイド』

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

V$MANAGED_STANDBYの非推奨

V$MANAGED_STANDBYビューはOracle Database 12cリリース2 (12.2.0.1)で非推奨となりました。新しいビューV$DATAGUARD_PROCESSSの使用をお薦めします。

V$DATAGUARD_PROCESSSビューにはOracle Data Guardで使用されるプロセスについて、はるかに多くの情報が含まれています。

関連項目

- 『Oracle Databaseリファレンス』

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

一部のXML DB関数の非推奨

Oracle Database 12cリリース2 (12.2)から、このトピックでリストされているオプションは非推奨になりました。

次のオプションが非推奨になりました。

- Oracle XQuery関数`ora:contains`。かわりにXQuery Full Textを使用してください。
- Oracle SQL関数XMLRoot。かわりに、バージョン番号付きのSQL/XML関数`XMLSerialize()`を使用してください。
- 索引構成表(IOT)として格納されたネストした表。これには、ネストした表の列Nを持つ表を作成するときに、オプション`DBMS_XMLSCHEMA.REGISTER_NT_AS_IOT`と句`NESTED TABLE N STORE AS ... (ORGANIZATION INDEX)`の両方の使用が含まれます。かわりに、ネストした表の列はヒープ・ストレージを使用して格納してください(PL/SQLプロシージャ`DBMS_XMLSCHEMA.registerSchema`のデフォルトの動作)。
- PL/SQLプロシージャ`DBMS_XSLPROCESSOR.CLOB2FILE`。かわりに`DBMS_LOB.CLOB2FILE`を使用してください。
- PL/SQL関数`DBMS_XSLPROCESSOR.READ2CLOB`。かわりに`DBMS_LOB.LOADCLOBFROMFILE`を使用してください。
- Oracle XML DBでのXLinkの使用。
- Oracle XML DB Content Connector。

詳細は、『*Oracle XML DB*開発者ガイド』を参照してください。

関連項目

- 『Oracle XML DB開発者ガイド』

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Oracle XML Databaseの非推奨となった機能

これらの機能は、Oracle Database 12cリリース1では非推奨になり、将来のリリースでサポートが終了する可能性があります。

- `XMLType`の`CLOB`記憶域(非構造化記憶域とも呼ばれる)は非推奨になりました。かわりに`XMLType`のバイナリXML記憶域を使用してください。

  XMLファイルで空白を保持するには、元のXML文書の2つのコピーを格納します。一方のファイルは、データベースでの使用やXML処理のための`XMLType`インスタンスとして使用し、他方のファイルは、文書の忠実性を実現するための`CLOB`インスタンスとして使用します。

- オブジェクト・リレーショナル`XMLType`データに埋め込まれている`CLOB`インスタンスとして格納されたXMLフラグメントに対する`XMLIndex`索引の作成は、非推奨になりました。このようなフラグメントのデータを索引付けする必要がある場合は、オブジェクト・リレーショナル記憶域のかわりに、バイナリXML記憶域を使用して文書を格納します。
- 次に示す、パッケージ`DBMS_XMLSCHEMA`内のPL/SQLサブプログラムは非推奨になりました。
  - `generateSchema`
  - `generateSchemas`

  これらの構造体の代用はなく、この変更に対する回避策はありません。

- `DBMS_XDB_CONFIG`は新しいPL/SQLパッケージです。Oracle XML DBのすべての構成関数、プロシージャおよび定数は、パッケージ`DBMS_XDB`から`DBMS_XDB_CONFIG`に移動されています。これらの関数、プロシージャおよび定

数は、パッケージDBMS_XDBでは非推奨になりました。かわりにパッケージDBMS_XDB_CONFIGでこれらを使用してください。

次に、パッケージDBMS_XDBで非推奨となったサブプログラムのリストを示します。

- ADDHTTPEXPIREMAPPING
- ADDMIMEMAPPING
- ADDSCHEMALOCMAPPING
- ADDSERVLET
- ADDSERVLETMAPPING
- ADDSERVLETSECROLE
- ADDXMLEXTENSION
- CFG_GET
- CFG_REFRESH
- CFG_UPDATE
- DELETEHTTPEXPIREMAPPING
- DELETEMIMEMAPPING
- DELETESCHEMALOCMAPPING
- DELETESERVLET
- DELETESERVLETMAPPING
- DELETESERVLETSECROLE
- DELETEXMLEXTENSION
- GETFTPPORT
- GETHTTPPORT
- GETLISTENERENDPOINT
- SETFTPPORT
- SETHTTPPORT
- SETLISTENERENDPOINT
- SETLISTENERLOCALACCESS

次に、パッケージDBMS_XDBで非推奨となった定数のリストを示します。

- XDB_ENDPOINT_HTTP
- XDB_ENDPOINT_HTTP2
- XDB_PROTOCOL_TCP
- XDB_PROTOCOL_TCPS

- XMLデータを更新するためのOracle SQL関数はすべて非推奨になっています。これらの関数のかわりにXQuery Updateを使用してください。次に、非推奨となったXML更新関数のリストを示します。
  - updateXML
  - insertChildXML
  - insertChildXMLbefore
  - insertChildXMLafter
  - insertXMLbefore
  - insertXMLafter

    - appendChildXML
    - deleteXML
- Oracle SQL関数sys_xmlgenは非推奨になりました。かわりにSQL/XML生成関数を使用してください。
- 次のOracle XQuery関数は非推奨になりました。かわりに、対応する標準のXQuery関数(同じ名前に名前空間接頭辞fnの付いた関数)を使用してください。
    - ora:matches: かわりにfn:matchesを使用
    - ora:replace: かわりにfn:replaceを使用
- XML変換をサポートする次のOracle構造体は非推奨になりました。
    - PL/SQLパッケージDBMS_XMLTRANSLATIONS
    - Oracle XPath関数ora:translate
    - XML Schema注釈xdb:maxOccurs、xdb:srclangおよびxdb:translate

  これらの構造体の代用はなく、この変更に対する回避策はありません。
- 次のXML Schema注釈は非推奨になりました。
    - xdb:defaultTableSchema
    - xdb:maintainOrder
    - xdb:mapUnboundedStringToLob
    - xdb:maxOccurs
    - xdb:SQLCollSchema
    - xdb:SQLSchema
    - xdb:srclang
    - xdb:storeVarrayAsTable
    - xdb:translate

  これらの構造体の代用はなく、この変更に対する回避策はありません。
- エクスポート・パラメータdata_optionsの値xml_clobsは、Oracle Database 12cから非推奨になりました。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Oracle Database 12cリリース2 (12.2)で非推奨となった初期化パラメータ

パラメータ設定の変更および代替機能を理解するには、12.2リリースで非推奨となったパラメータを参照してください。これらのパラメータは、以降のリリースで削除される可能性があります。

O7_DICTIONARY_ACCESSIBILITY初期化パラメータ

初期化パラメータO7_DICTIONARY_ACCESSIBILITYは、SYSTEM権限の制限を制御します。このパラメータがTRUEに設定されている場合、SYSスキーマ内のオブジェクトに対するアクセスが許可されます。デフォルト設定はFALSEです。このデフォルト設定の場合、すべてのスキーマ内のオブジェクトにアクセスできるシステム権限は、SYSスキーマ内のオブジェクトにアクセスできません。O7_DICTIONARY_ACCESSIBILITYパラメータは非推奨になりました。

ASM_PREFERRED_READ_FAILURE_GROUPS初期化パラメータ

ASM_PREFERRED_READ_FAILURE_GROUPS初期化パラメータはOracle Automatic Storage Management 12cリリース2 (12.2.0.1)では非推奨になりました。Oracle Automatic Storage Management (Oracle ASM) 12c

リリース2 (12.2.0.1)以上では、優先読取り失敗グループの指定が自動的に行われるようになったため、ASM_PREFERRED_READ_FAILURE_GROUPS初期化パラメータの使用は不要になりました。優先読取り機能を制御するには、PREFERRED_READ.ENABLEDディスク・グループ属性を使用します。

PARALLEL_ADAPTIVE_MULTI_USER初期化パラメータ

初期化パラメータPARALLEL_ADAPTIVE_MULTI_USERは、パラレル実行を使用するマルチユーザー環境で、パフォーマンスを改善するための適応アルゴリズムを使用するかどうかを指定します。このパラメータは非推奨となり、デフォルト値はFALSEになりました。このパラメータの代替機能はありません。Oracle Database機能のパラレル文キューイングを使用して、パラレル実行のパフォーマンスを向上することをお薦めします。

UTL_FILE_DIR初期化パラメータ

初期化パラメータUTL_FILE_DIRは、PL/SQLファイルのI/O用にアクセス可能なディレクトリを指定します。このパラメータは非推奨となったため、UTL_FILE_DIRのアクセスは指定しないことをお薦めします。かわりにディレクトリ・オブジェクト機能を使用することをお薦めします。この機能によって、UTL_FILE_DIRが置き換えられます。ディレクトリ・オブジェクトには次のメリットがあります。

- UTL_FILEアプリケーション管理者に対してより高い柔軟性と詳細な制御を提供します
- データベースを停止せずに動的に管理できます
- 他のOracleツールと一貫性があります。

関連項目

- 『Oracle Databaseリファレンス』

親トピック: Oracle Database 12cリリース2 (12.2)の動作の変更、サポート終了および非推奨

Oracle Database 12cリリース1 (12.1)の動作の変更、サポート終了および非推奨

Oracle Database 12cリリース1 (12.1)の変更の説明を確認してください。

- Oracle Database 12cリリース1 (12.1)の動作の変更
  Oracle Database 12cリリース1 (12.1)へのアップグレードの計画に役立つ動作の変更を確認してください。
- Oracle Database 12cリリース1 (12.1)でサポートが終了した機能
  アップグレード計画の一環として、サポートが終了した機能のリストを確認してください。
- Oracle Database 12cリリース2 (12.2)で非推奨となった機能
  この項にリストされている非推奨となった機能を参照して、アップグレード後に使用する代替方法を準備してください。
- Oracle Database 12cリリース1 (12.1)で非推奨となった初期化パラメータ
  アップグレード計画の一環として、Oracle Database 12cリリース1 (12.1)で非推奨となった初期化パラメータを確認し、代替方法を確認してください。
- Oracle Database 12cリリース1 (12.1)で非推奨となったビュー
  この項にリストされている非推奨となった機能を確認して、アップグレード後に使用する代替方法を準備します。以前のリリースで非推奨になった機能は、今後のリリースでサポートされなくなる可能性があります。

親トピック: Oracle Databaseの変更、サポート終了および非推奨

Oracle Database 12cリリース1 (12.1)の動作の変更

Oracle Database 12cリリース1 (12.1)へのアップグレードの計画に役立つ動作の変更を確認してください。

- catupgrd.sql Run Without PARALLEL=NOに関連したエラー
  `catctl.pl`を実行するかわりに`catupgrd.sql`スクリプトの実行を選択した場合は、追加の入力パラメータであるPARALLELの情報を提供する必要があります。
- スタンドアロン削除ツールの変更点
  Oracle Database 12c以上では、削除スタンドアロン・ユーティリティは、Oracle Universal Installer (OUI)を使用した削除オプションに置き換えられました。
- セキュリティ監査機能の変更点
  Oracle Database 12cリリース1 (12.1)以上のリリースでは、監査機能のフル・セットが自動的に使用可能になります。
- SQLNET.ALLOWED_LOGON_VERSIONパラメータが設定されていないシステムのアップグレード
  SQLNET.ALLOWED_LOGON_VERSION_SERVERのパラメータ設定を参照して、アップグレードしたデータベースのセキュリティおよびクライアント接続に対する影響を確認してください。
- Oracle Warehouse Builder (OWB)はOracle Databaseでインストールされない
  Oracle Warehouse Builderは、個別にインストールする必要があります。
- Oracle Databaseリリース10.2または11.1およびOCFSとRAWデバイスのアップグレードについて
  WindowsまたはRAWデバイスのOCFSにOracle Clusterwareファイルを格納するOracle Databaseリリース10.2.0.5またはリリース11.1.0.7環境をアップグレードする場合、Oracle Database 12cリリース1 (12.1)へ直接アップグレードすることはできません。
- VARCHAR2、NVARCHAR2およびRAWデータ型の変更点
  Oracle Database 12cリリース1 (12.1)以上のリリースでは、これらのデータ型のMAX_STRING_SIZE値を`32767`バイトに増加できます。
- RESOURCE_LIMITパラメータのデフォルトの変更
  RESOURCE_LIMITは、デフォルトでTRUEに設定されます。
- Oracle XML DBは必須でアンインストールできない
  Oracle Database 12c以上では、Oracle XML DBは、Oracle Databaseの必須コンポーネントです。
- Oracle RACのデフォルトで有効になっているDirect NFS
  Oracle Database 12cリリース1 (12.1)以上のリリースでは、Direct NFS (DNFS)はデフォルトでOracle Real Application Clusters (Oracle RAC)インストールで有効になっています。

親トピック: Oracle Database 12cリリース1 (12.1)の動作の変更、サポート終了および非推奨

PARALLEL=NOなしで実行されたcatupgrd.sqlに関連するエラー

`catctl.pl`のかわりに`catupgrd.sql`スクリプトを実行する場合、追加の入力パラメータであるPARALLELの情報を指定する必要があります。

たとえば:

```
SQL> catupgrd.sql PARALLEL=NO
```

パラメータを使用せずに`catupgrd.sql`を実行した場合、次のエラー・メッセージが表示されます。

```
NOTE

The catupgrd.sql script is being deprecated in the 12.1 release of
Oracle Database. Customers are encouraged to use catctl.pl as
```

```
the replacement for catupgrd.sql when upgrading the database dictionary.

                 cd $ORACLE_HOME/rdbms/admin
                 $ORACLE_HOME/perl/bin/perl catctl.pl -n 4 catupgrd.sql

Refer to the Oracle Database Upgrade Guide for more information.

This database upgrade procedure must be called with the following
argument when invoking from the SQL prompt:

                 @catupgrd.sql PARALLEL=NO
```

親トピック: Oracle Database 12cリリース1 (12.1)での動作の変更

スタンドアロン削除ツールの変更点

Oracle Database 12c以上では、削除スタンドアロン・ユーティリティは、Oracle Universal Installer (OUI)を使用した削除オプションに置き換えられました。

削除ツールは、Oracle Database、Oracle Database ClientまたはOracle Grid Infrastructureのインストール・メディアのベース・ディレクトリからも実行できます。

削除ツールを実行する場合、`-deinstall`および`-home`オプションを指定し、LinuxおよびUNIXでは`runInstaller`コマンドを、Windowsでは`setup.exe`を使用します。

参照:

- OUIおよび`runInstaller`の使用の詳細は、ご使用のオペレーティング・システムの『Oracle Databaseインストレーション・ガイド』を参照してください。
- ご使用のオペレーティング・システムの『Oracle Grid Infrastructureインストレーション・ガイド』

親トピック: Oracle Database 12cリリース1 (12.1)での動作の変更

セキュリティ監査機能の変更点

Oracle Database 12cリリース1 (12.1)以上のリリースでは、監査機能のフル・セットが自動的に使用可能になります。

監査機能は、Oracle Database 12cで再設計されました。Oracle Database 12cで新しいデータベースを作成すると、監査の拡張機能のフル・セットが自動的に使用可能になります。以前のリリースからアップグレードした場合は、新しい監査機能の一部およびアップグレード元のリリースからの監査機能を使用するためのオプションが提供されます。最新の監査機能のフル・セットに移行することを強くお薦めします。

参照:

新しい監査機能およびセキュリティの変更の詳細は、『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 12cリリース1 (12.1)での動作の変更

SQLNET.ALLOWED_LOGON_VERSIONパラメータが設定されていないシステムのアップグレード

SQLNET.ALLOWED_LOGON_VERSION_SERVERのパラメータ設定を参照して、アップグレードしたデータベースのセキュリティおよびクライアント接続に対する影響を確認してください。

SQLNET.ALLOWED_LOGON_VERSIONパラメータが設定されていないシステム(デフォルトのSQLNet設定の8を使用し

ているシステム)をアップグレードする場合、ログイン・ベリファイアの新しいデフォルト値は11に設定され、それによりOracle Database 18cでは11gパスワード・ベリファイア(11G)が設定されます。この値によって、Oracle Database 10g、11g、12cおよび18cクライアントは、データベースとOracle Database 19cクライアントに接続できます。

アップグレードしたOracle DatabaseサーバーでSQLNET.ALLOWED_LOGON_VERSION_SERVER (または非推奨となったSQLNET.ALLOWED_LOGON_VERSION)に対してパラメータ設定を指定しない場合、Oracle Database 19cのデフォルトは11になります。この値によって、重要なパッチ更新*CPU Oct 2012*以上のパッチが適用されていない、および接続のためにOracle Database 10gベリファイア(10G)を使用する必要のあるOracle Databaseリリース11.2.0.3より前のリリースを使用するクライアントからの接続が可能になります。

設定を高くすると、ベリファイアの使用がさらに制限されます。8に設定すると、ほとんどのベリファイアが許可されます。たとえば、この設定では10G、11Gおよび12Cベリファイアがすべて許可されます。12aに設定すると、12Cベリファイアのみが許可されます。セキュリティを強化するために、12aへのSQLNET.ALLOWED_LOGON_VERSION_SERVERの設定を検討してください。12に設定すると、11Gおよび12Cベリファイアが認証に使用されます。

参照:

ベリファイア設定およびクライアント・アクセスの詳細は、『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 12cリリース1 (12.1)での動作の変更

Oracle Warehouse Builder (OWB)はOracle Databaseでインストールされない

Oracle Warehouse Builderは、個別にインストールする必要があります。

Oracle Database 12c以上では、Oracle Warehouse Builder (OWB)は、Oracle Databaseのソフトウェアの一部としてインストールされません。Oracle Warehouse Builderのインストーラは、Oracle Technology Networkで入手できます。以前のリリースから存在しているOWBコンポーネントは、Oracle Databaseのアップグレード処理の一部としてアップグレードされません。

親トピック: Oracle Database 12cリリース1 (12.1)での動作の変更

Oracle Databaseリリース10.2または11.1およびOCFSとRAWデバイスのアップグレードについて

WindowsまたはRAWデバイスのOCFSにOracle Clusterwareファイルを格納するOracle Databaseリリース10.2.0.5またはリリース11.1.0.7環境をアップグレードする場合、Oracle Database 12cリリース1 (12.1)へ直接アップグレードすることはできません。

まず、Oracle Databaseリリース11.2への中間アップグレードを実行し、その後にOracle ClusterwareファイルをOracle Automatic Storage Management (Oracle ASM)に移行する必要があります。その後、リリース11.2からOracle Database 12cリリース1 (12.1)にアップグレードできます。

参照:

- Oracle Database 11gリリース2 (11.2)用の『*Oracle Database*アップグレード・ガイド』を取得するには、http://www.oracle.com/technetwork/documentation/index.htmlを参照してください
- Oracle Databaseソフトウェアのインストールの手順は、ご使用のオペレーティング・システムの『Oracle Databaseインストレーション・ガイド』を参照してください
- Oracle Automatic Storage Management (Oracle ASM)を使用するためのデータベースの移行の詳細は、

『Oracle Automatic Storage Management管理者ガイド』を参照してください。

親トピック: Oracle Database 12cリリース1 (12.1)での動作の変更

VARCHAR2、NVARCHAR2およびRAWデータ型の変更点

Oracle Database 12cリリース1 (12.1)以上のリリースでは、これらのデータ型のMAX_STRING_SIZE値を32767バイトに増加できます。

Oracle Database 12cリリース1 (12.1)以上では、VARCHAR2、NVARCHAR2およびRAWデータ型の最大サイズを32767バイトに増加できます。このサイズの増加が可能になるのは、COMPATIBLE初期化パラメータが12.0以上に設定され、MAX_STRING_SIZE初期化パラメータがEXTENDEDに設定されている場合です。このサイズの増加は、非CDB Oracle Databaseインスタンスと、マルチテナント・アーキテクチャのPDBで使用できます。

この変更を行う場合、UPGRADEモードでutl32k.sqlスクリプトを実行する必要があります。スクリプトの実行後、MAX_STRING_SIZEをEXTENDEDに設定できます。

注意:

COMPATIBLE 初期化を 12.0 以上に増加すると、その変更は元に戻すことができません。

親トピック: Oracle Database 12cリリース1 (12.1)での動作の変更

RESOURCE_LIMITパラメータのデフォルトの変更

RESOURCE_LIMITは、デフォルトでTRUEに設定されます。

Oracle DatabaseのRESOURCE_LIMITパラメータは、リソース制限がデータベース・プロファイルで施行されるかどうかを判断します。このリリースでは、RESOURCE_LIMITパラメータはデフォルトでTRUEに設定されます。Oracleのリソース制限が無効化された場合、定義されたプロファイル制限は無視されます。この動作は、パスワード・リソースには適用されません。

次のSQL*Plusコマンドを入力して、resource_limitパラメータの設定を確認します。

```
SQL> select value from v$parameter where name = 'resource_limit';
```

親トピック: Oracle Database 12cリリース1 (12.1)での動作の変更

Oracle XML DBは必須でアンインストールできない

Oracle Database 12c以上では、Oracle XML DBは、Oracle Databaseの必須コンポーネントです。

Oracle XML DBをアンインストールすることはできず、Oracle Databaseの作成時にそれを除外するためのオプションはありません。Oracle XML DBは、既存のOracle DatabaseをOracle Database 12cにアップグレードするときに自動的にインストールまたはアップグレードされます。

親トピック: Oracle Database 12cリリース1 (12.1)での動作の変更

Oracle RACのデフォルトで有効になっているDirect NFS

Oracle Database 12cリリース1 (12.1)以上のリリースでは、Direct NFS (DNFS)はデフォルトでOracle Real Application Clusters (Oracle RAC)インストールで有効になっています。

デフォルトでは、Oracle RACのインストール時にDirect NFSクライアントが有効な状態でインストールされます。これは、Direct NFSを有効にする必要があるOracle Database単一インスタンスのインストールとは異なります。Direct NFSクライ

アントはデフォルトでインストールの一部として無効になっています。Oracle RAC Oracle Databaseリリースをリリース12.1以上にアップグレードし、ソース・データベースでDirect NFSを有効にしていない場合、エラー・メッセージORA-17500: 「ODMエラー:操作は許可されていません。」が発生します。

この変更の詳細およびアップグレードの準備のオプションを確認するには、My Oracle Supportノート954425.1および1966267.1を参照してください

関連項目

- My Oracle SupportドキュメントID 954425.1
- My Oracle SupportドキュメントID 1966267.1

親トピック: Oracle Database 12cリリース1 (12.1)での動作の変更

Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

アップグレード計画の一環として、サポートが終了した機能をこのリストで確認してください。

- RAW記憶域デバイスのサポート終了
  Oracle Database 12cリリース1 (12.1)以上では、RAWデバイスのブロック・ファイル記憶域はサポートされていません。
- テキスト索引のALTER INDEX OPTIMIZEのサポート終了
  ALTER INDEX OPTIMIZE `[token index_token | fast | full [maxtime (time | unlimited)]`操作は、Oracle Database 12cではサポートされません。
- CLEANUP_ORACLE_BASEプロパティのサポート終了
  Oracle Database 12cでは、レスポンス・ファイル(サイレント)での削除用のCLEANUP_ORACLE_BASEプロパティが削除されました。
- Oracle Enterprise Manager Database Controlのサポート終了
  Oracle Database 12c以上では、Oracle Enterprise Manager Database Controlはサポートが終了したため、使用できません。
- Secure Sockets Layer (SSL)の暗号スイートのサポート終了
  Oracle Advanced Securityを使用する場合、サポートが終了した暗号スイートのこのリストを参照してください。
- データベースのRules Manager (RUL)およびExpression Filter (EXF)のサポート終了
  Oracle Database 12cリリース1以上では、Expression Filter (EXF)およびデータベースのRules Manager (RUL)機能のサポートが終了しました。
- OCFS用のcluvfy comp cfsのサポート終了
  `cluvfy comp cfs`コンポーネント検証コマンド・オプションは、Oracle Database 12cリリース1 (12.1)から削除されました。
- チェンジ・データ・キャプチャのサポート終了
  Oracleチェンジ・データ・キャプチャは、Oracle Database 12cに含まれておらず、Oracle GoldenGateに置き換えられました。
- Oracle Data Miningのサポートが終了した機能
  これらのOracle Data Mining機能は、Oracle Database 12cでサポートが終了しました。
- 暗黙的接続キャッシュのサポート終了
  暗黙的接続キャッシュは、Oracle Database 12cではサポートが終了しました。
- Oracle Data MiningのアップグレードでのABNモデルのサポート終了
  Oracle Database 12c以上では、Data Mining Java APIおよびAdaptive Bayes Network (ABN)アルゴ

リズムは、サポートが終了しました。

- OLAPカタログ(AMD)のサポート終了
  Oracle Database 12c以上では、共通ウェアハウス・メタモデル(CWM)標準のOLAPカタログ(AMD)に対するサポートが終了しました。
- OracleグローバリゼーションのCSSCANおよびCSALTERのサポート終了
  Oracle Database 12cには、Oracle Database Migration Assistant for Unicode (DMU)が含まれており、レガシー・データベース・ツールのCSSCANおよびCSALTERのサポートは終了しました。
- Oracle Netの接続プーリングのサポート終了
  Oracle Database 12cでは、Oracle Netの接続プーリングはサポートされません。
- Oracle Net Listenerパスワードのサポート終了
  Oracle Database 12cでは、Oracle Net Listenerのパスワード機能はサポートされません。
- Oracle Namesのサポート終了
  Oracle Namesは、Oracle Database 11gでネーミング・メソッドとしてのサポートが終了しました。ディレクトリ・ネーミングに移行する必要があります。
- Oracle Net ServicesのOracle Names制御ユーティリティのサポート終了
  Oracle Names制御ユーティリティはサポートが終了しました。Oracle Database 10g以上では、これは使用できません。
- Oracle TextおよびOracle XML DBでのCTXXPATHのサポート終了
  Oracle Database 12cリリース1 (12.1)以上では、`CTXSYS.CTXXPATH`索引はサポートが終了しました。
- Oracle Net ServicesのSQLNET.KERBEROS5_CONF_MITパラメータのサポート終了
  SQLNET.KERBEROS5_CONF_MITネットワーク・パラメータは、不要になったため`sqlnet.ora`ではサポートされません。
- テキスト索引のALTER INDEX OPTIMIZEのサポート終了
  ALTER INDEX OPTIMIZE `[token index_token | fast | full [maxtime (time | unlimited)]`操作は、Oracle Database 12cではサポートされません。
- テキスト索引のSYNC [MEMORY memsize]のサポート終了
  `SYNC [MEMORY memsize]`操作は、Oracle Database 12cではサポートされません
- Microsoft Windowsプラットフォームでのサポートが終了した機能
  Microsoft Windowsで非推奨となったかサポートが終了したOracle Database 12cの機能を理解するには、これらのトピックを参照してください。

親トピック: Oracle Database 12cリリース1 (12.1)の動作の変更、サポート終了および非推奨

RAW記憶域デバイスのサポート終了

Oracle Database 12cリリース1 (12.1)以上では、RAWデバイスのブロック・ファイル記憶域はサポートされていません。

RAWデバイス上に格納されているすべてのデータファイルを、Oracle ASM、クラスタ・ファイル・システムまたはネットワーク・ファイル・システム(NFS)に移行する必要があります。

このサポート終了のガイドラインは、Oracle ClusterwareのOCRおよび投票ディスク・ファイルにも適用されます。OCRまたは投票ディスク・ファイルをRAWデバイスに格納することはできません。アップグレードを開始する前に、Oracle ClusterwareファイルをRAWデバイスからサポートされている記憶域オプションに移動する必要があります。

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

テキスト索引のALTER INDEX OPTIMIZEのサポート終了

ALTER INDEX OPTIMIZE `[token index_token | fast | full [maxtime (time | unlimited)]`操作は、Oracle Database 12cではサポートされません。

索引を最適化するには、CTX_DDL.OPTIMIZE_INDEXを使用してください。

参照:

OPTIMIZE_INDEXの詳細は、『Oracle Textリファレンス』を参照してください。

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

CLEANUP_ORACLE_BASEプロパティのサポート終了

Oracle Database 12cでは、レスポンス・ファイル(サイレント)での削除用の`CLEANUP_ORACLE_BASE`プロパティが削除されました。

CLEANUP_ORACLE_BASEを使用して、サイレントまたはレスポンス・ファイル・モードの削除中にOracleベースを削除することは、サポートされなくなりました。

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

Oracle Enterprise Manager Database Controlのサポート終了

Oracle Database 12c以上では、Oracle Enterprise Manager Database Controlはサポートが終了したため、使用できません。

代替として、Oracle Enterprise Manager Database Express (Oracle EM Express)が導入されました。Oracle EM Expressは、Oracle Database 12cにアップグレードするときにインストールされます。

Oracle EM ExpressのHTTPポートの手動構成を実行できます。

1. `init.ora/spfile` (デフォルト設定)で次の文字列を検索します。

   ```
   dispatchers=(PROTOCOL=TCP)(SERVICE=sample XDB)
   ```

   Oracle EM Expressのポート構成を確認します。

   ```
   SQL> select DBMS_XDB_CONFIG.getHTTPport() from dual;
   SQL> select DBMS_XDB_CONFIG.getHTTPSport() from dual;
   ```

2. 新しいポートを設定します。たとえば:

   ```
   SQL> exec DBMS_XDB_CONFIG.setHTTPport(5500);
   SQL> exec DBMS_XDB_CONFIG.setHTTPSport(8080);
   ```

3. ブラウザでOracle EM Expressのホーム・ページにアクセスします。たとえば、指定した設定を使用して、次のパスを確認します(`database`はSID、`hostname`は完全修飾ドメイン名、`port`は割当て済のポートです)。

   ```
   http://database-hostname:port/em
   ```

   たとえば:

   ```
   http://localhost:5500/em
   ```

4. CDBとすべてのPDBで、異なるポート番号を使用してこの構成ステップを繰り返します。

参照:

- Oracle EM Expressの概要は『Oracle Database概要』を参照してください。
- Oracle EM Expressの使用の詳細は、『Oracle Database 2日でデータベース管理者』を参照してください。
- Oracle Enterprise Manager Cloud Controlの概要は、『Oracle Database概要』を参照してください。

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

Secure Sockets Layer (SSL)の暗号スイートのサポート終了

Oracle Advanced Securityを使用する場合、サポートが終了した暗号スイートのこのリストを参照してください。

Oracle Advanced Securityでは、Oracle Database 12cの次の暗号スイートのサポートが終了しました。

- `SSL_DH_anon_WITH_DES_CBC_SHA`
- `SSL_RSA_EXPORT_WITH_DES40_CBC_SHA`
- `SSL_RSA_EXPORT_WITH_RC4_40_MD5`
- `SSL_RSA_WITH_DES_CBC_SHA`

参照:

サポートされるSSL暗号スイートの詳細は、『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 12cリリース1 (12.1)でサポートが終了した機能

データベースのRules Manager (RUL)およびExpression Filter (EXF)のサポート終了

Oracle Database 12cリリース1以上では、Expression Filter (EXF)およびデータベースのRules Manager (RUL)機能のサポートが終了しました。

Rules Managerを使用している場合、Oracle Fusion Middlewareのコンポーネントであるオラクル Business Rulesに移行することをお薦めします。Oracle DatabaseのExpression Filter機能は、Continuous Query Notificationに置き換えられました。

このスクリプトはアップグレード・プロセスによって実行されます。アップグレード前にこれらのコンポーネントを削除するには、アップグレード前に`catnoexf.sql`スクリプトを実行します。`catnoexf.sql`スクリプトは、`ORACLE_HOME/rdbms/admin/`の下にあります。

参照:

https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1244535.1

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

OCFS用のcluvfy comp cfsのサポート終了

`cluvfy comp cfs`コンポーネント検証コマンド・オプションは、Oracle Database 12cリリース1 (12.1)から削除されました。

クラスタ検証ユーティリティ(`cluvfy`)のオプション`comp cfs`は、Oracle Cluster File System (OCFS)のサポートが終了

したため、Oracle Database 12.1で削除されました。

参照:

クラスタ管理の詳細は、『Oracle Clusterware管理およびデプロイメント・ガイド』を参照してください。

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

チェンジ・データ・キャプチャのサポート終了

Oracleチェンジ・データ・キャプチャは、Oracle Database 12cに含まれておらず、Oracle GoldenGateに置き換えられました。

参照:

詳細は、Oracle GoldenGateのドキュメントを参照してください。

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

Oracle Data Miningのサポートが終了した機能

これらのOracle Data Mining機能は、Oracle Database 12cでサポートが終了しました。

- Oracle Data Mining Java APIは使用できません。Oracle Data Mining 12cに対するプログラム・インタフェースは、`DBMS_DATA_MINING`および`DBMS_DATA_MINING_TRANSFORM`の2つのPL/SQLパッケージと、データのスコアリング用のSQL言語の関数ファミリで構成されます。

  参照:

  Data Mining PL/SQLパッケージの詳細は、Oracle Data Miningユーザー・ガイドを参照してください。

- Adaptive Bayes Network (ABN)アルゴリズムは使用できません。Oracle Data Mining 12cでは、デシジョン・ツリーアルゴリズムはABNを置き換えます。

  参照:

  デシジョン・ツリー・アルゴリズムの詳細は、Oracle Data Mining概要を参照してください。

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

暗黙的接続キャッシュのサポート終了

暗黙的接続キャッシュは、Oracle Database 12cではサポートが終了しました。

かわりに、ユニバーサル接続プールを使用してください。

参照:

詳細は、『Oracle Universal Connection Pool for JDBC開発者ガイド』を参照してください。

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

Oracle Data Miningのアップグレードでの ABNモデルのサポート終了

Oracle Database 12c以上では、Data Mining Java APIおよびAdaptive Bayes Network (ABN)アルゴリズムは、サポートが終了しました。

Oracleではこれらの機能のサポートが終了しているため、Oracle Data Mining Java APIによって作成されたモデルをOracle Databaseリリース11gからOracle Database 12cにアップグレードすることはできません。その他のすべてのモデルおよびメタデータは、Oracle Database 11gからOracle Database 12cへのアップグレード時に自動的にアップグレードされます。ABNモデルのアップグレードは可能ですが、Oracle Database 12cデータベースでそれらを使用することはできません。アップグレードの前または後にABNモデルを削除する必要があります。ABNモデルは、Oracle Database 12cデータベースで新しい分類モデルを作成することによって置き換えが可能です。

参照:

Data Mining APIおよびデータ・マイニング・モデルの詳細は、Oracle Data Miningユーザー・ガイドを参照してください。

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

OLAPカタログ(AMD)のサポート終了

Oracle Database 12c以上では、共通ウェアハウス・メタモデル(CWM)標準のOLAPカタログ(AMD)に対するサポートが終了しました。

CWM標準のサポートは、Oracle Database 11gリリース2 (11.2)で非推奨となりました。既存のデータベースでOLAPカタログにCWMメタデータがあり、Oracle Database 12cにアップグレードした場合は、アップグレードしたデータベースにはAMDコンポーネントがあります。アップグレードしたデータベースにAMDコンポーネントがない場合は、新しいOracle Database 12cインストールにAMDが含まれていないため、アップグレードしたOracle Database 12cデータベースにもAMDコンポーネントがありません。データベースにAMDコンポーネントがあり、それを削除する場合は、パス`ORACLE_HOME/olap/admin/catnoamd.sql`にある`catnoamd.sql`スクリプトを実行します。このスクリプトは、アップグレードが完了する前または後のどちらでも実行できます。

ノート:

アップグレードするデータベースに OLAP カタログが存在する場合、`AMD OLAP Catalog OPTION OFF` および無効な CWM OLAP オブジェクトが表示されることがあります。無効な OLAP オブジェクトは不要なため、無視しても問題ありません。

参照:

- OLAP Java APIメタデータの詳細は、『*Oracle OLAP Java API*開発者ガイド』を参照してください。
- Oracle DatabaseのOLAPオプションおよびオンライン分析処理の詳細は、『*Oracle OLAP*ユーザーズ・ガイド』を参照してください

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

OracleグローバリゼーションのCSSCANおよびCSALTERのサポート終了

Oracle Database 12cには、Oracle Database Migration Assistant for Unicode (DMU)が含まれており、レガシー・データベース・ツールのCSSCANおよびCSALTERのサポートは終了しました。

DMUでは、データベース管理者に対する包括的なエンドツーエンドのユニコード移行ソリューションが提供されます。Oracle Database 12c以上では、Oracle DatabaseにDMUが含まれます。CSSCANおよびCSALTERツールは、含まれないか、サポートされなくなりました。

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

Oracle Netの接続プーリングのサポート終了

Oracle Database 12cでは、Oracle Netの接続プーリングはサポートされません。

Oracle Netの接続プーリングはOracle Database 11gで非推奨となりました。この非推奨には、`DISPATCHERS`属性の`TICKS`、`SESSIONS`および`CONNECTIONS`が含まれていました。

参照:

ディスパッチャの構成の詳細は、『Oracle Database Net Services管理者ガイド』を参照してください

Oracle Connection Managerパラメータ(cman.ora)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: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

Oracle Net Listenerパスワードのサポート終了

Oracle Database 12cでは、Oracle Net Listenerのパスワード機能はサポートされません。

Oracle Databaseではローカル・オペレーティング・システム認証によって認証が実施されるため、この変更によってセキュリティが低下することはありません。

関連項目

- 『Oracle Database Net Servicesリファレンス・ガイド』

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

Oracle Namesのサポート終了

Oracle Namesは、Oracle Database 11gでネーミング・メソッドとしてのサポートが終了しました。ディレクトリ・ネーミングに移行する必要があります。

参照:

ディレクトリ・ネーミングへの移行方法の詳細は、『Oracle Database Net Services管理者ガイド』を参照してください。

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

Oracle Net ServicesのOracle Names制御ユーティリティのサポート終了

Oracle Names制御ユーティリティはサポートが終了しました。Oracle Database 10g以上では、これは使用できません。

Oracle Names制御ユーティリティのサポート終了により、関連するすべての制御ユーティリティ・コマンドのサポートも終了しました。Oracle Databaseクライアントは、接続文字列の解決にNames Serverを使用できません。アプリケーションをOracle Internet DirectoryにLDAPディレクトリ・ネーミングを使用して移行します。

参照:

ディレクトリ・ネーミング・メソッドの構成の詳細は、『Oracle Database Net Servicesリファレンス』を参照してください。

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

Oracle TextおよびOracle XML DBでのCTXXPATHのサポート終了

Oracle Database 12cリリース1 (12.1)以上では、`CTXSYS.CTXXPATH`索引はサポートが終了しました。

`CTXSYS.CTXXPATH`のサポート終了は、`CTXCAT`に影響しません。かわりに`XMLIndex`索引を使用します。

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

Oracle Net ServicesのSQLNET.KERBEROS5_CONF_MITパラメータのサポート終了

SQLNET.KERBEROS5_CONF_MITネットワーク・パラメータは、不要になったため`sqlnet.ora`ではサポートされません。

デフォルトでは、SQLNET.KERBEROS5_CONF_MITの値はFALSEに設定されます。このパラメータ設定は、Kerberos構成に影響を与えません。以前のリリースでは、SQLNET.KERBEROS5_CONF_MITがTRUEに設定されると、このパラメータによってMIT Kerberos 5で指定された形式で解析が設定されました。ただし、このパラメータ設定は、現在は不要になりました。Oracle Database 12c以上では、MIT Kerberos 5構成のみがサポートされます。

参照:

`sqlnet.ora`ファイルのKerberosパラメータの詳細は、『Oracle Database Net Servicesリファレンス』を参照してください

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

テキスト索引のALTER INDEX OPTIMIZEのサポート終了

ALTER INDEX OPTIMIZE `[token index_token | fast | full [maxtime (time | unlimited)]`操作は、Oracle Database 12cではサポートされません。

索引を最適化するには、CTX_DDL.OPTIMIZE_INDEXを使用してください。

参照:

OPTIMIZE_INDEXの詳細は、『Oracle Textリファレンス』を参照してください。

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

テキスト索引のSYNC [MEMORY memsize]のサポート終了

`SYNC [MEMORY memsize]`操作は、Oracle Database 12cではサポートされません。

索引を同期化するには、`CTX_DDL.SYNC_INDEX`を使用してください。

関連項目

- Oracle Textリファレンス

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

Microsoft Windowsプラットフォームでのサポートが終了した機能

Microsoft Windowsで非推奨となったかサポートが終了したOracle Database 12cの機能を理解するには、これらのトピックを参照してください。

- WindowsでのOracle COM Automationのサポート終了
  Oracle Database 12cにはOracle COM Automationは含まれません。
- Oracle Objects for OLEのサポート終了
  Oracle Database 12cにはMicrosoft WindowsシステムのOracle Objects for OLEは含まれません。
- Oracle Counters for Windows Performance Monitorのサポート終了
  Oracle Database 12cにはOracle Counters for Windows Performance Monitorは含まれません。
- WindowsでのOracle Cluster File System (OCFS)のサポート終了
  Oracle Database 12c以上では、Oracle Cluster File System (OCFS)のWindowsでのサポートが終了しました。このサポート終了通知によるLinuxでのOCFS (OCFSおよびOCFS2)のサポートおよび配布に対する影響はありません。

親トピック: Oracle Database 12cリリース1 (12.1)でサポートが終了した機能

WindowsでのOracle COM Automationのサポート終了

Oracle Database 12cにはOracle COM Automationは含まれません。

この機能は、Oracle Database 11g (データベース・コンポーネントOracle COM Automationを含む最後のデータベース・リリース)で非推奨になりました。ご使用のOracle COMアプリケーションを.NET Frameworkなどの最近のテクノロジに移行することをお薦めします。

参照:

https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1175293.1

親トピック: Microsoft Windowsプラットフォームでのサポートが終了した機能

Oracle Objects for OLEのサポート終了

Oracle Database 12cにはMicrosoft WindowsシステムのOracle Objects for OLEは含まれません。

Oracle Objects for OLEは、Oracle Database 11gで非推奨になりました。コードをOLE DBデータ・アクセス標準およびActiveX Data Objects (ADO)に移行するか、アプリケーションを.NET (またはJavaあるいは別のアプリケーション・アーキテクチャ)に移行して別のドライバを使用できます。

参照:

https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1175303.1

親トピック: Microsoft Windowsプラットフォームでのサポートが終了した機能

Oracle Counters for Windows Performance Monitorのサポート終了

Oracle Database 12cにはOracle Counters for Windows Performance Monitorは含まれません。

Oracle Counters for Windows Performance Monitorは、Oracle Database 11gで非推奨となっていました。以前のリリースでは、カウンタはデフォルトではインストールされず、カウンタはWindows上でのみ動作します。監視には、Oracle Enterprise Manager Cloud Controlを使用することをお薦めします。

参照:

https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1175297.1

親トピック: Microsoft Windowsプラットフォームでのサポートが終了した機能

WindowsでのOracle Cluster File System (OCFS)のサポート終了

Oracle Database 12c以上では、Oracle Cluster File System (OCFS)のWindowsでのサポートが終了しました。このサポート終了通知によるLinuxでのOCFS (OCFSおよびOCFS2)のサポートおよび配布に対する影響はありません。

現在Windows上で、Oracleクラスタ・ファイル(Oracle Cluster Registryおよび投票ファイル)またはデータベース・ファイル、またはその両方をホストするためにOCFSを使用しているデータベースでは、Oracle Database 12cにアップグレードする前に、これらのファイルをOCFSから移行する必要があります。

参照:

- Oracle Automatic Storage Management (Oracle ASM)を使用するためのデータベースの移行の詳細は、『Oracle Automatic Storage Management管理者ガイド』を参照してください。
- https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1392280.1

親トピック: Microsoft Windowsプラットフォームでのサポートが終了した機能

Oracle Database 12cリリース2 (12.2)で非推奨となった機能

この項にリストされている非推奨となった機能を参照して、アップグレード後に使用する代替方法を準備してください。

- 非CDBアーキテクチャの非推奨
  非CDBアーキテクチャはOracle Database 12cで非推奨になりました。Oracle Database 19cの後のリリースではサポート対象から外れ、使用できなくなる可能性があります。
- catupgrd.sqlスクリプトの非推奨およびパラレル・アップグレード・ユーティリティの導入
  Oracle Database 12cリリース1 (12.1)では、`catupgrd.sql`のかわりにパラレル・アップグレード・ユーティリティ(`catctl.pl`)を使用してパラレル・アップグレードを有効化することをお薦めします。
- 非推奨となったDELETE_CATALOG_ROLE
  `DELETE_CATALOG_ROLE`データベース・ロールは、Oracle Database 12cでは非推奨です。
- Oracle Label Securityで非推奨となった機能とパラメータ
  Oracle Database 12cリリース1では、9個のOracle Label Securityの機能が非推奨になりました。
- 非推奨となったOracle Database VaultのAPI
  DVSYS.DBMS_MACADM.SYNC_RULESプロシージャは、その機能がルール作成機能に組み込まれたため、非推奨になりました。
- 非推奨となったOracle Database Vaultのデフォルトのレルム
  Oracleデータ・ディクショナリのレルムおよび Oracle Enterprise Managerのレルムは、このリリースでは非推奨になりました。

- 非推奨となったOracle Database Vaultのデフォルトのルール・セット
  ここにリストされているルール・セットは、Oracle Database 12cリリース1では非推奨です。
- NTLMプロトコルを使用したWindows NTS認証の非推奨
  セキュリティ脆弱性の理由から、NTLMは、Oracle Database 12cでは非推奨です。
- 透過的データ暗号化に対する公開キー・インフラストラクチャの非推奨
  透過的データ暗号化(TDE)に対する公開キー・インフラストラクチャ(PKI)は、Oracle Database 12cでは非推奨です。
- Oracle Data Guard Brokerの非推奨となった機能
  Oracle Data Guard Brokerを使用する場合、これらの非推奨を確認してください。
- 非推奨となったEndToEndMetrics関連のAPI
  EndToEndMetrics関連のAPIは、Oracle Database 12cでは非推奨です。かわりに、ユニバーサル接続プールを使用してください。
- Oracle Restartの非推奨
  Oracle Restart (スタンドアロン・サーバー用のOracle Grid Infrastructureとも呼ばれる)は、Oracle Database 12cでは非推奨です。
- QOSCTLサービスのクオリティ(QoS)コマンドの-checkpasswdの非推奨
  構文`qosctl -checkpasswd username password`は非推奨となりました。
- 非推奨となったOracle Net Services用のNT LAN Manager (NTLM)プロトコル
  ドメイン認証用のNT LAN Manager (NTLM)プロトコルは、Oracle Windowsアダプタでは非推奨となっています。
- Oracle Call Interfaceの非推奨となった機能
  ここにリストされているOracle Call Interface (OCI)機能はOracle Database 12cでは非推奨です。これらは、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
- Oracle Streamsの非推奨
  Oracle Streamsは、Oracle Database 12cでは非推奨です。Oracle Databaseの新しいリリースではサポート対象から外れ、使用できなくなる可能性があります。
- Oracle Data Provider for .NETの非推奨のプログラミング・インタフェース
  トランザクション・ガード用のOracle Data Provider for .NETアプリケーション・プログラミング・インタフェースは、Oracle Database 12c (12.1.0.2)で非推奨となり、将来のリリースでサポート終了になる可能性があります。
- 非推奨となったOracle DatabaseのセマンティクスのVPDサポート
  仮想プライベート・データベース(VPD)は、Oracle Database 12c (12.1)のOracle Databaseのセマンティク・テクノロジでは非推奨です。
- Oracle Databaseのセマンティク・テクノロジのバージョン対応モデルのサポート
  バージョン対応モデル機能は、Oracle Database 12c (12.1)のOracle Databaseのセマンティク・テクノロジでは非推奨です。
- 非推奨となったアドバンスト・レプリケーション
  Oracle Databaseアドバンスト・レプリケーションは、Oracle Database 12cでは非推奨です。
- 1文字SRVCTL CLIオプションの非推奨
  Oracle Database 12c以上では、1文字オプションは非推奨です。これらは、将来のリリースでサポートが終了する可能性があります。
- クラスタ管理での管理ユーザーの格納されたリストの非推奨
  クラスタ管理は、Oracle Database 12c以上では異なる方法で管理されます。
- サーバー内部SQLJの非推奨
  データベース・サーバー内部でのSQLJの使用は、このリリースでは非推奨です。

- 非推奨となったOracleバッチ更新
  Oracleバッチ更新APIは、Oracle Database 12cでは非推奨です。
- 非推奨となったEndToEndMetrics関連のAPI
  EndToEndMetrics関連のAPIは、Oracle Database 12cでは非推奨です。かわりに、ユニバーサル接続プールを使用してください。
- 非推奨となったストアド・アウトライン
  ストアド・アウトラインは、Oracle Database 12cでは非推奨です。
- 非推奨となったoracle.sqlパッケージの具象クラス
  `oracle.sql`パッケージの具象クラスは、Oracle Database 12cでは非推奨です。
- 非推奨となったdefineColumnTypeメソッド
  JDBCメソッド`defineColumnType`は、Oracle Database 12cリリース1では非推奨です。
- 非推奨となったCONNECTION_PROPERTY_STREAM_CHUNK_SIZEプロパティ
  JDBCプロパティCONNECTION_PROPERTY_STREAM_CHUNK_SIZEは、このリリースでは非推奨です。
- Oracle Data Pumpエクスポート・ユーティリティの機能
  Oracle Data PumpエクスポートのDATA_OPTIONSパラメータの`XML_CLOBS`オプションは非推奨です。
- Oracle Databaseのセマンティク・テクノロジのバージョン対応モデルのサポート
  バージョン対応モデル機能は、Oracle Database 12c (12.1)のOracle Databaseのセマンティク・テクノロジでは非推奨です。
- 非推奨となったdefineColumnTypeメソッド
  JDBCメソッド`defineColumnType`は、Oracle Database 12cリリース1では非推奨です。
- 非推奨となったOracle Net Services用のNT LAN Manager (NTLM)プロトコル
  ドメイン認証用のNT LAN Manager (NTLM)プロトコルは、Oracle Windowsアダプタでは非推奨となっています。
- Oracle Call Interfaceの非推奨となった機能
  ここにリストされているOracle Call Interface (OCI)機能はOracle Database 12cでは非推奨です。これらは、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
- クラスタ管理での管理ユーザーの格納されたリストの非推奨
  クラスタ管理は、Oracle Database 12c以上では異なる方法で管理されます。
- -cleanupOBaseの非推奨
  削除ツールの`-cleanupOBase`フラグは、Oracle Database 12cリリース1 (12.1)では非推奨です。
- Oracle XML Databaseの非推奨となった機能
  これらの機能は、Oracle Database 12cリリース1では非推奨になり、将来のリリースでサポートが終了する可能性があります。
- 非推奨となったアドバンスト・レプリケーション
  Oracle Databaseアドバンスト・レプリケーションは、Oracle Database 12cでは非推奨です。
- Oracle Database 12cリリース1 (12.1)で非推奨となったOracle MultimediaのDICOM
  Oracle Database 12cでは、Oracle Multimedia DICOMのORDImageのサポートは非推奨です。
- JPublisherの非推奨

親トピック: Oracle Database 12cリリース1 (12.1)の動作の変更、サポート終了および非推奨

非CDBアーキテクチャの非推奨

非CDBアーキテクチャはOracle Database 12cで非推奨になりました。Oracle Database 19cの後のリリースではサポート対象から外れ、使用できなくなる可能性があります。

CDBアーキテクチャの使用をお薦めします。

ノート:

CDB アーキテクチャで正常に動作しない機能がいくつか残っています(README の第 2.2.1 項「マルチテナント・コンテナ・データベースで制限される機能または使用できない機能」を参照)。これらの機能を使用する必要がある場合、それらが CDB アーキテクチャで正常に動作するようになるまで、非 CDB アーキテクチャを使用してください。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

catupgrd.sqlスクリプトの非推奨およびパラレル・アップグレード・ユーティリティの導入

Oracle Database 12cリリース1 (12.1)では、`catupgrd.sql`のかわりにパラレル・アップグレード・ユーティリティ(`catctl.pl`)を使用してパラレル・アップグレードを有効化することをお薦めします。

Oracle Database 12cリリース1 (12.1)では、新しいパラレル・アップグレード・ユーティリティ(`catctl.pl`)が導入されました。このユーティリティは、以前のリリースで使用されていた`catupgrd.sql`スクリプトを置き換えます。`catupgrd.sql`スクリプトの使用は可能ですが、Oracle Database 12cからは非推奨となります。これは、将来のリリースでは削除される予定です。新しいパラレル・アップグレード・ユーティリティを使用してデータベースのアップグレードを実行することをお薦めします。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

非推奨となったDELETE_CATALOG_ROLE

`DELETE_CATALOG_ROLE` データベース・ロールは、Oracle Database 12cでは非推奨です。

参照:

このロールおよびディクショナリ・オブジェクトの詳細は、Oracle Databaseリファレンスを参照してください

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Oracle Label Securityで非推奨となった機能とパラメータ

Oracle Database 12cリリース1では、9個のOracle Label Securityの機能が非推奨になりました。

ここにリストされたOracle Label Securityの機能は、Oracle Database 12cリリース1では非推奨になりました。これらは、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。次に示す代替機能を使用することをお薦めします。

- LEAST_UBOUND.かわりにOLS_GREATEST_LBOUNDを使用します。
- LUBD。かわりにOLS_GLBDを使用します。
- DOMINATES。かわりにOLS_DOMINATESを使用します。
- DOM。かわりにOLS_STRICTLY_DOMINATESを使用します。
- STRICTLY_DOMINATES。かわりにOLS_STRICTLY_DOMINATESを使用します。
- S_DOM。かわりにOLS_STRICTLY_DOMINATESを使用します。
- DOMINATED_BY。かわりにOLS_DOMINATED_BYを使用します。
- DOM_BY。かわりにOLS_DOMINATED_BYを使用します。
- STRICTLY_DOMINATED_BY。かわりにOLS_STRICTLY_DOMINATED_BYを使用します。

- S_DOM_BY。かわりにOLS_STRICTLY_DOMINATED_BYを使用します。

参照:

- GREATEST_LBOUNDでの下限の最大値の検索の詳細は、Oracle Label Security管理者ガイドを参照してください
- OLS_GLBDの詳細は、Oracle Label Security管理者ガイドを参照してください
- OLS_DOMINATESのスタンドアロン機能の詳細は、Oracle Label Security管理者ガイドを参照してください
- OLS_STRICTLY_DOMINATESのスタンドアロン機能の詳細は、Oracle Label Security管理者ガイドを参照してください
- OLS_DOMINATED_BYのスタンドアロン機能の詳細は、Oracle Label Security管理者ガイドを参照してください
- OLS_STRICTLY_DOMINATED_BYのスタンドアロン機能の詳細は、Oracle Label Security管理者ガイドを参照してください

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

非推奨となったOracle Database VaultのAPI

DVSYS.DBMS_MACADM.SYNC_RULESプロシージャは、その機能がルール作成機能に組み込まれたため、非推奨になりました。

参照:

ルール作成の詳細は、『Oracle Database Vault管理者ガイド』を参照してください。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

非推奨となったOracle Database Vaultのデフォルトのレルム

Oracleデータ・ディクショナリのレルムおよびオラクル Enterprise Managerのレルムは、このリリースでは非推奨になりました。

以前Oracleデータ・ディクショナリのレルムで保護されていたオブジェクトは、新しいレルムに移行されています。

参照:

新しいレルムおよびデフォルトのレルムの詳細は、『Oracle Database Vault管理者ガイド』を参照してください。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

非推奨となったOracle Database Vaultのデフォルトのルール・セット

ここにリストされているルール・セットは、Oracle Database 12cリリース1では非推奨です。

次のルール・セットは、このリリースでは非推奨です。

- 「Oracle Data Pump操作を許可」ルール・セット
- 「スケジューラ・ジョブの許可」ルール・セット

関連項目

- Oracle Database Vaultの管理者ガイド

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

NTLMプロトコルを使用したWindows NTS認証の非推奨

セキュリティ脆弱性の理由から、NTLMは、Oracle Database 12cでは非推奨です。

Windowsユーザーは、NT Lan Manager (NTLM)プロトコルが必要なWindowsクライアントおよびサーバー上でNTSアダプタを使用して認証することはできません。WindowsユーザーはKerberosは使用できます。NTLMは、ローカル・ユーザーの認証、およびデータベース・サービスがローカル・ユーザーで実行されている場合にはまだ使用できます。

新しいクライアント側`sqlnet.ora`のブール・パラメータNO_NTLM (デフォルト値はFALSE)を使用すると、NTS認証でNTLMを使用できるかどうかを制御できます。NO_NTLMをTRUEに設定すると、このパラメータ値により、NTLMがWindows NTS認証で使用されなくなります。

参照:

このリリースでのWindowsプラットフォームに影響する変更の詳細は、Oracle Databaseプラットフォーム・ガイドfor Microsoft Windowsを参照してください。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

透過的データ暗号化に対する公開キー・インフラストラクチャの非推奨

透過的データ暗号化(TDE)に対する公開キー・インフラストラクチャ(PKI)は、Oracle Database 12cでは非推奨です。

TDEを構成するには、ADMINISTER KEY MANAGEMENT SQL文を使用します。その他のPKIの実装に影響はありません。

参照:

TDEの構成の詳細は、『Oracle Database Advanced Securityガイド』を参照してください。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Oracle Data Guard Brokerの非推奨となった機能

Oracle Data Guard Brokerを使用する場合、これらの非推奨を参照してください。

義務の分離機能に対するOracle Data Guard Brokerのサポートの一部として、Oracle Database 12cでは、次のブローカ・プロパティは非推奨です。

- `LsbyASkipCfgPr`
- `LsbyASkipErrorCfgPr`
- `LsbyASkipTxnCfgPr`
- `LsbyDSkipCfgPr`
- `LsbyDSkipErrorCfgPr`
- `LsbyDSkipTxnCfgPr`
- `LsbySkipTable`

- `LsbySkipTxnTable`

代替となるものはありません。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

非推奨となったEndToEndMetrics関連のAPI

EndToEndMetrics関連のAPIは、Oracle Database 12cでは非推奨です。かわりに、ユニバーサル接続プールを使用してください。

次のAPIは非推奨となり、JDBCのJavadocでは非推奨とマークされます。

- getEndToEndMetrics
- getEndToEndECIDSequenceNumber
- setEndToEndMetrics
- setApplicationContext
- clearAllApplicationContext

参照:

JDBCユニバーサル接続プールの詳細は、『Oracle Database JDBC開発者ガイド』およびJDBCのJavadocを参照してください。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Oracle Restartの非推奨

Oracle Restart (スタンドアロン・サーバー用のOracle Grid Infrastructureとも呼ばれる)は、Oracle Database 12cでは非推奨です。

Oracle Restartは、単一インスタンスOracle DatabaseおよびOracle ASMインスタンスのみの管理に制限されており、今後のリリースではサポート終了となる可能性があります。Oracle ASMは、スタンドアロンおよびクラスタ・デプロイメント用のOracle Grid Infrastructureインストールの一部として引き続き提供され、Oracle RAC One Nodeも引き続き提供されます。

参照:

- Oracle Restartが非推奨となった発表とその代替製品については、次のMy Oracle Supportのノート1584742.1を参照してください。

  https://support.oracle.com/epmos/faces/DocumentDisplay?id=1584742.1&displayIndex=1
- Oracle Database管理者ガイド

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

QOSCTLサービスのクオリティ(QoS)コマンドの-checkpasswdの非推奨

構文`qosctl -checkpasswd username password`は非推奨となりました。

参照:

QOSCTL構文とコマンドの詳細は、『Oracle Database Quality of Service Managementユーザーズ・ガイド』を参照してください

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

非推奨となったOracle Net Services用のNT LAN Manager (NTLM)プロトコル

ドメイン認証用のNT LAN Manager (NTLM)プロトコルは、Oracle Windowsアダプタでは非推奨となっています。

NTSアダプタには、Kerberos認証のみが使用されます。

参照:

`sqlnet.ora`ファイルのKerberosパラメータの詳細は、『Oracle Database Net Servicesリファレンス』を参照してください。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Oracle Call Interfaceの非推奨となった機能

ここにリストされているOracle Call Interface (OCI)機能はOracle Database 12cでは非推奨です。これらは、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

- sqlnet.oraのOCIデプロイ・パラメータは非推奨です。これらには次のパラメータが含まれます。
  - OCIのクライアント結果キャッシュ・パラメータ: `OCI_RESULT_CACHE_MAX_SIZE`、`OCI_RESULT_CACHE_MAX_RSET_SIZE`および`OCI_RESULT_CACHE_MAX_RSET_ROWS`

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Oracle Streamsの非推奨

Oracle Database 12cではOracle Streamsは非推奨です。Oracle Databaseの新しいリリースではサポート対象から外れ、使用できなくなる可能性があります。

Oracle GoldenGateを使用して、Oracle Streamsのすべてのレプリケーション機能を置き換えます。

ノート:

Oracle Database アドバンスト・キューイングは Oracle Streams とは無関係で、今後も機能強化されます。

参照:

詳細は、Oracle GoldenGateのドキュメントを参照してください。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Oracle Data Provider for .NETの非推奨のプログラミング・インタフェース

トランザクション・ガード用のOracle Data Provider for .NETアプリケーション・プログラミング・インタフェースは、Oracle

Database 12c (12.1.0.2)で非推奨となり、将来のリリースでサポートされなくなる可能性があります。

次のアプリケーション・プログラミング・インタフェースは非推奨です。

- `OracleLogicalTransactionStatus`クラス
- `OracleConnection.LogicalTransactionId`プロパティ
- `OracleConnection.GetLogicalTransactionStatus`メソッド

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

非推奨となったOracle Databaseのセマンティク・テクノロジのVPDサポート

仮想プライベート・データベース(VPD)は、Oracle Database 12c (12.1)のOracle Databaseのセマンティク・テクノロジでは非推奨です。

仮想プライベート・データベース(VPD)に依存する既存のセマンティク・テクノロジ・アプリケーションを、Oracle Label Security (OLS)をかわりに使用するように移行します。

参照:

RDFデータに対するファイングレインなアクセス制御の詳細は、Oracle Databaseのセマンティク・テクノロジ開発者ガイドを参照してください。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Oracle Databaseのセマンティク・テクノロジのバージョン対応モデルのサポート

バージョン対応モデル機能は、Oracle Database 12c (12.1)のOracle Databaseのセマンティク・テクノロジでは非推奨です。

セマンティク・データでのWorkspace Managerの使用に対する具体的な代替手段は、アプリケーションの用途に応じて異なります。

参照:

RDFデータに対するワークスペース・マネージャのサポートの詳細は、Oracle Databaseのセマンティク・テクノロジ開発者ガイドを参照してください。

使用例および推奨される代替手段の詳細は、https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1468273.1を参照してください

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

非推奨となったアドバンスト・レプリケーション

Oracle Databaseアドバンスト・レプリケーションは、Oracle Database 12cでは非推奨です。

読取り専用のマテリアライズド・ビューは、引き続き、ベーシック・レプリケーションでサポートされます。

Oracle GoldenGateを使用して、アドバンスト・レプリケーションのすべての機能を置き換えます。このガイダンスには、マルチマスター・レプリケーション、更新可能なマテリアライズド・ビューおよびデプロイメント・テンプレートが含まれます。

参照:

Oracle GoldenGateのドキュメント

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

1文字SRVCTL CLIオプションの非推奨

Oracle Database 12c以上では、1文字オプションは非推奨です。これらは、将来のリリースでサポートが終了する可能性があります。

サーバー制御ユーティリティ(SRVCTL)のコマンドライン・インタフェース(CLI)では、以前のリリースで使用されていた短いCLIオプションに加えて、GNUスタイルの長いオプションもサポートされます。

参照:

SRVCTLの詳細は、『Oracle Real Application Clusters管理およびデプロイメント・ガイド』を参照してください。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

クラスタ管理での管理ユーザーの格納されたリストの非推奨

クラスタ管理は、Oracle Database 12c以上では異なる方法で管理されます。

Oracle Database 12c以上では、管理ユーザーの格納されたリストを使用したクラスタの管理方法は、ポリシー・セットのアクセス制御リストを構成することによる管理ユーザー・ロールのより包括的な管理によって置き換えられています。

参照:

クラスタ管理の詳細は、『Oracle Clusterware管理およびデプロイメント・ガイド』を参照してください。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

サーバー内部SQLJの非推奨

データベース・サーバー内部でSQLJを使用することは、このリリースで非推奨です。

データベース内部でSQLJアプリケーションを翻訳して実行する機能は、これらかのリリースでは使用できません。SQLJは、Oracle Databaseに接続し、クライアントとして実行するために使用するアプリケーションの翻訳用クライアント・ツールとしてのみ使用できます。SQLJはストアド・プロシージャ、ストアド・ファンクションまたはストアド・トリガー内部では使用できません。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

非推奨となったOracleバッチ更新

Oracleバッチ更新APIは、Oracle Database 12cでは非推奨です。

次のAPIは非推奨となり、JDBCのJavadocでは非推奨とマークされます。

- `OraclePreparedStatement.setExecuteBatch()`
- `OraclePreparedStatement.getExecuteBatch()`
- `OracleCallableStatement.setExecuteBatch()`

かわりに標準バッチ更新を使用してください。

関連項目

- 『Oracle Database JDBC開発者ガイド』

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

非推奨となったEndToEndMetrics関連のAPI

EndToEndMetrics関連のAPIは、Oracle Database 12cでは非推奨です。かわりに、ユニバーサル接続プールを使用してください。

次のAPIは非推奨となり、JDBCのJavadocでは非推奨とマークされます。

- getEndToEndMetrics
- getEndToEndECIDSequenceNumber
- setEndToEndMetrics
- setApplicationContext
- clearAllApplicationContext

参照:

JDBCユニバーサル接続プールの詳細は、『Oracle Database JDBC開発者ガイド』およびJDBCのJavadocを参照してください。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

非推奨となったストアド・アウトライン

ストアド・アウトラインは、Oracle Database 12cでは非推奨です。

かわりに、計画ベースラインを使用してください。

参照:

ストアド・アウトラインの計画ベースラインへの移行の詳細は、『Oracle Database SQLチューニング・ガイド』を参照してください

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

非推奨となったoracle.sqlパッケージの具象クラス

`oracle.sql`パッケージの具象クラスは、Oracle Database 12cでは非推奨です。

これらのクラスは、`oracle.jdbc`パッケージの新しいインタフェースで置き換えられます。

参照:

`oracle.jdbc`の新しいインタフェースの詳細は、『Oracle Database JDBC開発者ガイド』を参照してください

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

非推奨となったdefineColumnTypeメソッド

JDBCメソッドdefineColumnTypeは、Oracle Database 12cリリース1では非推奨です。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

非推奨となったCONNECTION_PROPERTY_STREAM_CHUNK_SIZEプロパティ

JDBCプロパティCONNECTION_PROPERTY_STREAM_CHUNK_SIZEは、このリリースでは非推奨になりました。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Oracle Data Pumpエクスポート・ユーティリティの機能

Oracle Data PumpエクスポートのDATA_OPTIONSパラメータのXML_CLOBSオプションは非推奨です。

参照:

エクスポートのDATA_OPTIONSパラメータの詳細は、『Oracle Databaseユーティリティ』を参照してください。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Oracle Databaseのセマンティク・テクノロジのバージョン対応モデルのサポート

バージョン対応モデル機能は、Oracle Database 12c (12.1)のOracle Databaseのセマンティク・テクノロジでは非推奨です。

セマンティク・データでのWorkspace Managerの使用に対する具体的な代替手段は、アプリケーションの用途に応じて異なります。

参照:

RDFデータに対するワークスペース・マネージャのサポートの詳細は、Oracle Databaseのセマンティク・テクノロジ開発者ガイドを参照してください。

使用例および推奨される代替手段の詳細は、https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1468273.1を参照してください

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

非推奨となったdefineColumnTypeメソッド

JDBCメソッドdefineColumnTypeは、Oracle Database 12cリリース1では非推奨です。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

非推奨となったOracle Net Services用のNT LAN Manager (NTLM)プロトコル

ドメイン認証用のNT LAN Manager (NTLM)プロトコルは、Oracle Windowsアダプタでは非推奨となっています。

NTSアダプタには、Kerberos認証のみが使用されます。

参照:

sqlnet.oraファイルのKerberosパラメータの詳細は、『Oracle Database Net Servicesリファレンス』を参照してください。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Oracle Call Interfaceの非推奨となった機能

ここにリストされているOracle Call Interface (OCI)機能はOracle Database 12cでは非推奨です。これらは、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

- sqlnet.oraのOCIデプロイ・パラメータは非推奨です。これらには次のパラメータが含まれます。
  - OCIのクライアント結果キャッシュ・パラメータ: OCI_RESULT_CACHE_MAX_SIZE、OCI_RESULT_CACHE_MAX_RSET_SIZEおよびOCI_RESULT_CACHE_MAX_RSET_ROWS

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

クラスタ管理での管理ユーザーの格納されたリストの非推奨

クラスタ管理は、Oracle Database 12c以上では異なる方法で管理されます。

Oracle Database 12c以上では、管理ユーザーの格納されたリストを使用したクラスタの管理方法は、ポリシー・セットのアクセス制御リストを構成することによる管理ユーザー・ロールのより包括的な管理によって置き換えられています。

参照:

クラスタ管理の詳細は、『Oracle Clusterware管理およびデプロイメント・ガイド』を参照してください。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

-cleanupOBaseの非推奨

削除ツールの-cleanupOBaseフラグは、Oracle Database 12cリリース1 (12.1)では非推奨です。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Oracle XML Databaseの非推奨となった機能

これらの機能は、Oracle Database 12cリリース1では非推奨になり、将来のリリースでサポートが終了する可能性があります。

- XMLTypeのCLOB記憶域(非構造化記憶域とも呼ばれる)は非推奨になりました。かわりにXMLTypeのバイナリXML記憶域を使用してください。

  XMLファイルで空白を保持するには、元のXML文書の2つのコピーを格納します。一方のファイルは、データベースでの使用やXML処理のためのXMLTypeインスタンスとして使用し、他方のファイルは、文書の忠実性を実現するためのCLOBインスタンスとして使用します。

- オブジェクト・リレーショナルXMLTypeデータに埋め込まれているCLOBインスタンスとして格納されたXMLフラグメントに対するXMLIndex索引の作成は、非推奨になりました。このようなフラグメントのデータを索引付けする必要がある場合は、オブジェクト・リレーショナル記憶域のかわりに、バイナリXML記憶域を使用して文書を格納します。

- 次に示す、パッケージDBMS_XMLSCHEMA内のPL/SQLサブプログラムは非推奨になりました。
  - generateSchema
  - generateSchemas

  これらの構造体の代用はなく、この変更に対する回避策はありません。

- DBMS_XDB_CONFIGは新しいPL/SQLパッケージです。Oracle XML DBのすべての構成関数、プロシージャおよび定数は、パッケージDBMS_XDBからDBMS_XDB_CONFIGに移動されています。これらの関数、プロシージャおよび定数は、パッケージDBMS_XDBでは非推奨になりました。かわりにパッケージDBMS_XDB_CONFIGでこれらを使用してください。

  次に、パッケージDBMS_XDBで非推奨となったサブプログラムのリストを示します。

  - ADDHTTPEXPIREMAPPING
  - ADDMIMEMAPPING
  - ADDSCHEMALOCMAPPING
  - ADDSERVLET
  - ADDSERVLETMAPPING
  - ADDSERVLETSECROLE
  - ADDXMLEXTENSION
  - CFG_GET
  - CFG_REFRESH
  - CFG_UPDATE
  - DELETEHTTPEXPIREMAPPING
  - DELETEMIMEMAPPING
  - DELETESCHEMALOCMAPPING
  - DELETESERVLET
  - DELETESERVLETMAPPING
  - DELETESERVLETSECROLE
  - DELETEXMLEXTENSION
  - GETFTPPORT
  - GETHTTPPORT
  - GETLISTENERENDPOINT
  - SETFTPPORT
  - SETHTTPPORT
  - SETLISTENERENDPOINT
  - SETLISTENERLOCALACCESS

  次に、パッケージDBMS_XDBで非推奨となった定数のリストを示します。

  - XDB_ENDPOINT_HTTP
  - XDB_ENDPOINT_HTTP2
  - XDB_PROTOCOL_TCP
  - XDB_PROTOCOL_TCPS

- XMLデータを更新するためのOracle SQL関数はすべて非推奨になっています。これらの関数のかわりにXQuery Updateを使用してください。次に、非推奨となったXML更新関数のリストを示します。

  - updateXML
  - insertChildXML
  - insertChildXMLbefore
  - insertChildXMLafter

    - insertXMLbefore
    - insertXMLafter
    - appendChildXML
    - deleteXML
- Oracle SQL関数sys_xmlgenは非推奨になりました。かわりにSQL/XML生成関数を使用してください。
- 次のOracle XQuery関数は非推奨になりました。かわりに、対応する標準のXQuery関数(同じ名前に名前空間接頭辞fnの付いた関数)を使用してください。
    - ora:matches: かわりにfn:matchesを使用
    - ora:replace: かわりにfn:replaceを使用
- XML変換をサポートする次のOracle構造体は非推奨になりました。
    - PL/SQLパッケージDBMS_XMLTRANSLATIONS
    - Oracle XPath関数ora:translate
    - XML Schema注釈xdb:maxOccurs、xdb:srclangおよびxdb:translate

  これらの構造体の代用はなく、この変更に対する回避策はありません。
- 次のXML Schema注釈は非推奨になりました。
    - xdb:defaultTableSchema
    - xdb:maintainOrder
    - xdb:mapUnboundedStringToLob
    - xdb:maxOccurs
    - xdb:SQLCollSchema
    - xdb:SQLSchema
    - xdb:srclang
    - xdb:storeVarrayAsTable
    - xdb:translate

  これらの構造体の代用はなく、この変更に対する回避策はありません。
- エクスポート・パラメータdata_optionsの値xml_clobsは、Oracle Database 12cから非推奨になりました。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

非推奨となったアドバンスト・レプリケーション

Oracle Databaseアドバンスト・レプリケーションは、Oracle Database 12cでは非推奨です。

読取り専用のマテリアライズド・ビューは、引き続き、ベーシック・レプリケーションでサポートされます。

Oracle GoldenGateを使用して、アドバンスト・レプリケーションのすべての機能を置き換えます。このガイダンスには、マルチマスター・レプリケーション、更新可能なマテリアライズド・ビューおよびデプロイメント・テンプレートが含まれます。

参照:

Oracle GoldenGateのドキュメント

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Oracle Database 12cリリース1 (12.1)で非推奨となったOracle MultimediaのDICOM

Oracle Database 12cでは、Oracle Multimedia DICOMのORDImageサポートは非推奨です。

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

JPublisherの非推奨

2014年10月以降、Oracle JPublisherはOracle Database 12cリリース1で非推奨になり、またJPublisherのすべての機能はOracle Database 12cリリース2でサポート対象外となり、使用できなくなります。次に示す代替方法を使用することをお薦めします。

- 引き続きWebサービス・コールアウトを使用するには、Webサービス・コールアウト・ユーティリティの代わりとなるOJVM Webサービス・コールアウト・ユーティリティを使用することをお薦めします。
- PL/SQLプログラムおよびSQLオブジェクトにJavaクライアント・アプリケーションを作成するには、開発者は、Java STRUCTクラスの作成を支援する他のJDK開発ツールおよび他の事前構成済オプションを使用することをお薦めします。

参照:

JPublisherの非推奨およびサポート終了の詳細は、My Oracle Supportのノート1937939.1を参照してください。

https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1937939.1

Oracle Technology NetworkのJDKツールおよびユーティリティも参照してください。

http://docs.oracle.com/javase/8/docs/technotes/tools/

親トピック: Oracle Database 12cリリース2 (12.2)で非推奨となった機能

Oracle Database 12cリリース1 (12.1)で非推奨となった初期化パラメータ

アップグレード計画の一環として、Oracle Database 12cリリース1 (12.1)で非推奨となった初期化パラメータを確認してください。

- 非推奨となったFILE_MAPPING初期化パラメータ
  FILE_MAPPING初期化パラメータは非推奨です。これは、下位互換性を保つためにのみサポートされています。
- 非推奨となったRDBMS_SERVER_DN初期化パラメータ
  RDBMS_SERVER_DN初期化パラメータは、Oracle Databaseリリース12.1.0.2では非推奨です。
- IGNORECASEおよびSEC_CASE_SENSITIVE_LOGONの非推奨
  ORAPWDのIGNORECASE引数およびSEC_CASE_SENSITIVE_LOGONシステム・パラメータは、Oracle Database 12cでは非推奨です。
- SQLNET.ALLOWED_LOGON_VERSIONパラメータの非推奨
  SQLNET.ALLOWED_LOGON_VERSIONパラメータはOracle Database 12cでは非推奨です。
- LOG_ARCHIVE_LOCAL_FIRST初期化パラメータのサポート終了
  LOG_ARCHIVE_LOCAL_FIRST初期化パラメータは、Oracle Database 12cで削除され、サポートが終了しました。

親トピック: Oracle Database 12cリリース1 (12.1)の動作の変更、サポート終了および非推奨

非推奨となったFILE_MAPPING初期化パラメータ

FILE_MAPPING初期化パラメータは非推奨です。これは、下位互換性を保つためにのみサポートされています。

参照:

FILE_MAPPING初期化パラメータの詳細は、『Oracle Databaseリファレンス』を参照してください

親トピック: Oracle Database 12cリリース1 (12.1)で非推奨となった初期化パラメータ

非推奨となったRDBMS_SERVER_DN初期化パラメータ

RDBMS_SERVER_DN初期化パラメータは、Oracle Databaseリリース12.1.0.2では非推奨です。

RDBMS_SERVER_DNのかわりにLDAP_DIRECTORY_ACCESSを使用します。

参照:

このパラメータの詳細は、『Oracle Databaseリファレンス』を参照してください

親トピック: Oracle Database 12cリリース1 (12.1)で非推奨となった初期化パラメータ

IGNORECASEおよびSEC_CASE_SENSITIVE_LOGONの非推奨

ORAPWDのIGNORECASE引数およびSEC_CASE_SENSITIVE_LOGONシステム・パラメータは、Oracle Database 12cでは非推奨です。

デフォルトでは、Oracle Database 12cのパスワードは大文字と小文字が区別されます。

参照:

認証およびパスワード保護の構成の詳細は、『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 12cリリース1 (12.1)で非推奨となった初期化パラメータ

SQLNET.ALLOWED_LOGON_VERSIONパラメータの非推奨

SQLNET.ALLOWED_LOGON_VERSIONパラメータはOracle Database 12cでは非推奨です。

SQLNET.ALLOWED_LOGON_VERSIONは、2つの新しいOracle Net Servicesパラメータに置き換えられました。

- SQLNET.ALLOWED_LOGON_VERSION_SERVER
- SQLNET.ALLOWED_LOGON_VERSION_CLIENT

関連項目

- SQLNET.ALLOWED_LOGON_VERSIONパラメータが設定されていないシステムのアップグレード
- SQLNET.ALLOWED_LOGON_VERSIONパラメータの動作の確認

参照:

SQLNET.ALLOWED_LOGON_VERSION_SERVERの詳細は、『Oracle Database Net Servicesリファレンス』を

参照してください

SQLNET.ALLOWED_LOGON_VERSION_CLIENTの詳細は、『Oracle Database Net Servicesリファレンス』を参照してください

この非推奨の詳細は、『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 12cリリース1 (12.1)で非推奨となった初期化パラメータ

LOG_ARCHIVE_LOCAL_FIRST初期化パラメータのサポート終了

LOG_ARCHIVE_LOCAL_FIRST初期化パラメータは、Oracle Database 12cで削除され、サポートが終了しました。

LOG_ARCHIVE_LOCAL_FIRSTは、アーカイバ・プロセス(ARCn)がREDOデータをリモートのスタンバイ・データベース宛先に送信するときに指定されます。これはOracle Database 11gで非推奨になっています。

Oracleでは、LOG_ARCHIVE_DEST_nを使用することをお薦めします。LOG_ARCHIVE_DEST_n初期化パラメータの属性を設定して、REDO転送サービスが本番用またはプライマリ・データベースの宛先から別の(スタンバイ)データベースの宛先にREDOデータを送信する方法について、様々な側面を制御します。

親トピック: Oracle Database 12cリリース1 (12.1)で非推奨となった初期化パラメータ

Oracle Database 12cリリース1 (12.1)で非推奨となったビュー

この項にリストされている非推奨となった機能を参照して、アップグレード後に使用する代替方法を準備してください。以前のリリースで非推奨になった機能は、今後のリリースでサポートされなくなる可能性があります。

次のビューは非推奨です。

- ALL_SCHEDULER_CREDENTIALSビュー。詳細は、『Oracle Databaseリファレンス』を参照してください。
- DBA_NETWORK_ACL_PRIVILEGESビュー。詳細は、『Oracle Databaseリファレンス』を参照してください。
- DBA_NETWORK_ACLSビュー。詳細は、『Oracle Databaseリファレンス』を参照してください。
- DBA_SCHEDULER_CREDENTIALSビュー。詳細は、『Oracle Databaseリファレンス』を参照してください。
- USER_NETWORK_ACL_PRIVILEGESビュー。詳細は、『Oracle Databaseリファレンス』を参照してください。
- USER_SCHEDULER_CREDENTIALSビュー。詳細は、『Oracle Databaseリファレンス』を参照してください。
- V$OBJECT_USAGEビュー。かわりにUSER_OBJECT_USAGEビューを使用してください。詳細は、『Oracle Databaseリファレンス』を参照してください。

参照:

スケジューラのジョブの資格証明の指定の詳細は、『Oracle Database管理者ガイド』を参照してください。

親トピック: Oracle Database 12cリリース1 (12.1)の動作の変更、サポート終了および非推奨

A AutoUpgrade REST API

Oracle Database用のAutoUpgrade REST APIを使用すると、Web環境でRESTfulサービスを使用してデータベースのアップグレードを自動化できます。

- AutoUpgrade REST APIの概要
  Oracle DatabaseのアップグレードのAutoUpgradeユーティリティREST APIメソッドについて学習します。
- スタート・ガイド
  AutoUpgrade REST APIを使用してデータベースをアップグレードするようにシステムを設定します。
- AutoUpgradeのREST API
  これらのREST APIは、Oracle REST Data Services (ORDS)のAutoUpgrade RESTサービス機能で使用できます。

AutoUpgrade REST APIの概要

Oracle DatabaseのアップグレードのAutoUpgradeユーティリティREST APIメソッドについて学習します。

- AutoUpgrade REST APIについて
  AutoUpgrade REST API実装では、Oracle REST Data Servicesプラグイン・フレームワークを使用します。

親トピック: AutoUpgrade REST API

AutoUpgrade REST APIについて

AutoUpgrade REST API実装では、Oracle REST Data Servicesプラグイン・フレームワークを使用します。

AutoUpgrade REST APIは、クライアント/サーバー・アーキテクチャ上でAutoUpgradeに統一されたインタフェースを提供し、ユーザーはHTTPおよびHTTPSプロトコルを介してAutoUpgrade機能を利用して統合し、Oracle Databasesを自動的にアップグレードできます。Oracle REST Data Servicesの実装は、強化されたセキュリティ、ファイル・キャッシュ機能およびRESTful Webサービスを提供するJava Enterprise Edition (Java EE)データ・サービスに基づいています。このサービスにより、スタンドアロン・デプロイメントだけでなく、Oracle WebLogic ServerやApache Tomcatなどのサーバーにおいても柔軟性が向上します。Oracle REST Data Servicesは、ほとんどのOracle Databaseコンポーネントおよびツールで使用される一般的なプラグイン・メカニズムです。

関連項目

- Oracle Database REST API

親トピック: AutoUpgrade REST APIの概要

スタート・ガイド

AutoUpgrade REST APIを使用してデータベースをアップグレードするようにシステムを設定します。

- cURLのインストール
  このドキュメント内の例では、cURLコマンドライン・ツールを使用して、Autoupgrade REST APIにアクセスする方法を示しています。
- AutoUpgrade REST APIを設定して使用する方法
  AutoUpgrade REST APIを使用するには、Oracle REST Data Servicesをインストールし、AutoUpgrade

RESTサービスを有効にする必要があります。

- AutoUpgrade REST API認証権限を使用する方法
  AutoUpgrade REST APIにアクセスするためのAutoUpgrade管理者ユーザー権限を持つユーザーを構成するには、この設定手順を実行します。

親トピック: AutoUpgrade REST API

cURLのインストール

このドキュメント内の例では、cURLコマンドライン・ツールを使用して、Autoupgrade REST APIにアクセスする方法を示しています。

サーバーに安全に接続するには、SSLをサポートしているバージョンのcURLをインストールし、Verisign社のCA証明書に対して認証される、SSL認証局(CA)証明書ファイルまたはバンドルを指定する必要があります。

1. ブラウザでcURLホーム・ページに移動し、左のナビゲーション・メニューで「Download」をクリックします。
2. cURLの「Releases and Downloads」ページで、使用するオペレーティング・システム用のSSL対応バージョンcURLソフトウェアを探し、ZIPファイルをダウンロードするリンクをクリックして、ソフトウェアをインストールします。
3. cURLの「CA certificates extracted from Mozilla」ページにナビゲートし、Mozilla CA証明書ストアをPEM形式(`cacert.pem`)でcURLをインストールしたフォルダにダウンロードします。
4. コマンド・ウィンドウを開いて、cURLをインストールしたディレクトリにナビゲートし、cURL環境変数`CURL_CA_BUNDLE`をSSL認証局(CA)証明書バンドルの場所に設定します。たとえば:

```
C:¥curl> set CURL_CA_BUNDLE=cacert.pem
```

これで、cURLを使用してAutoUpgrade REST APIインスタンスにRESTリクエストを送信する準備が整いました。

関連項目

- cURLコマンドライン・ツールおよびライブラリ
- Mozillaから抽出されたCA証明書

親トピック: スタート・ガイド

AutoUpgrade REST APIを設定して使用する方法

AutoUpgrade REST APIを使用するには、Oracle REST Data Servicesをインストールし、AutoUpgrade REST サービスを有効にする必要があります。

AutoUpgrade REST APIを有効化またはインストールする前に、スタンドアロン・モードでのOracle REST Data Servicesのインストールが完了していることを確認します。

Oracle REST Data Servicesの構成およびインストール

1. `ords.war`をインストールしてからORDSサービスを起動する前に、`autoupgrade.api.enabled`プロパティを`true`(デフォルトは`false`)に設定し、APIの環境変数を設定する必要があります。

   次の例では、

   - `autoupgrade.api.loglocation`がパス`/databases/working`に指定されています
   - `autoupgrade.api.aulocation`は、パス`/autoupgrade/ords/ords/autoupgrade.jar`にautoupgrade.jarファイルが設定されていることを指定します
   - `autoupgrade.api.jvmlocation`は、Javaの場所が`/usr/bin/java`にあることを指定します。

```
java -jar ords.war set-property autoupgrade.api.enabled true
java -jar ords.war set-property autoupgrade.api.loglocation /databases/working
java -jar ords.war set-property autoupgrade.api.aulocation
/autoupgrade/ords/ords/autoupgrade.jar
java -jar ords.war set-property autoupgrade.api.jvmlocation /usr/bin/java
```

2. JSON形式を使用するようにAutoUpgrade構成ファイルを修正します。

   AutoUpgrade REST APIを使用して新しいタスクを発行するには、標準のAutoUpgrade構成ファイル形式をJSON形式に変換する必要があります。

   たとえば、AutoUpgradeの標準構成ファイル(`config.cfg`)を次に示します。

```
[oracle@localhost curl]$ more config.cfg
#Global configurations
#Autoupgrade's global directory, non-job logs generated,
#temp files created and other autoupgrade files will be
#send here
global.autoupg_log_dir=/autoupgrade/test/glog
#
# Database number 1
#
upg1.start_time=NOW
upg1.source_home=/databases/product/12.2.0/dbhome_1
upg1.target_home=/databases/product/19c/dbhome_1
upg1.sid=db122
upg1.log_dir=/autoupgrade/test/log
upg1.upgrade_node=localhost
upg1.target_version=19.1
```

   次に、同じパラメータ情報を指定する同じ構成ファイルを示しますが、AutoUpgrade REST APIメソッドで使用するためにJSON構成入力ファイル(`config.json`)として書き換えられます。

```
[oracle@localhost curl]$ more config.json
{
"global": {
                "autoupg_log_dir": "/autoupgrade/test/glog"
        },
        "jobs": [{
                "start_time": "NOW",
                "source_home": "/databases/product/12.2.0/dbhome_1",
                "target_home": "/databases/product/19c/dbhome_1",
                "sid": "db122",
                "log_dir": "/autoupgrade/test/log",
                "upgrade_node": "localhost",
                "target_version": "19.1"
        }]
}
```

親トピック: スタート・ガイド

AutoUpgrade REST API認証権限を使用する方法

AutoUpgrade REST APIにアクセスするためのAutoUpgrade管理者ユーザー権限を持つユーザーを構成するには、この設定手順を実行します。

AutoUpgrade REST APIには、`AutoUpgrade`管理者と呼ばれるOracle REST Data Services (ORDS)に対する独自の認証ロールがあります。ORDS構成パラメータ`autoupgrade.api.ordsauth`が`true`に設定されている場合、ORDSユーザーには、AutoUpgrade REST APIを使用するために`AutoUpgrade`管理者ロールが必要です。

この機能を無効にするには、次のコマンドを入力します。

```
java -jar ords.war set-property autoupgrade.api.ordsauth false
```

autoupgrade.api.ordsauthがtrueに設定されている場合、ユーザーがAutoUpgrade REST APIにアクセスできるようにするには、そのユーザーにAutoUpgrade管理者ロールを明示的に付与する必要があります。

AutoUpgrade REST API管理者ロールを有効または無効にする前に、スタンドアロン・モードでのOracle REST Data Servicesのインストールが完了していることを確認します。

Oracle REST Data Servicesの構成およびインストール

1. ords.warをインストールした後でORDSサービスを開始する前に、autoupgrade.api.ordsauthがtrueに設定されていることを確認します。

```
 java -jar ords.war set-property autoupgrade.api.ordsauth true
```

2. AutoUpgrade管理者ロールを持つユーザーを作成します。たとえば、次のコマンドでは、AutoUpgrade管理者ロールを持つユーザーroderigoを作成します。

```
[oracle@localhost ords]$ java -jar ords.war user roderigo "AutoUpgrade
Administrator"
Enter a password for user roderigo:
Confirm password for user roderigo:
```

3. ords.warサーバーを起動します。
4. 作成したユーザーがAutoUpgradeエンドポイントに正常にアクセスできることを確認します。

   ユーザーroderigoを使用して成功した認証の例を次に示します。

```
[oracle@localhost curl]$ curl -u roderigo
"http://localhost:8080/autoupgrade/tasks"
Enter host password for user 'roderigo':
{
    "total_tasks": 0,
    "tasks": [
    ]
}
```

   失敗した認証の例を次に示します。

```
[oracle@localhost curl]$ curl "http://localhost:8080/autoupgrade/tasks"
{
    "code": "Unauthorized",
    "message": "Unauthorized",
    "type": "tag:oracle.com,2020:error/Unauthorized",
    "instance": "tag:oracle.com,2020:ecid/LI6fVLjXrI2rvmEFoBzAAw"
}
```

5. 必要に応じて、AutoUpgrade REST APIに対して認証ユーザーを追加で作成します。
6. 次のコマンドを使用して、AutoUpgrade REST API認可を無効にします。

```
java -jar ords.war set-property autoupgrade.api.ordsauth false
```

7. ORDSサーバーを再起動します。
8. これで、パスワード認証を必要とせずに、AutoUpgrade REST APIエンドポイントに直接アクセスできるようになります。

```
[oracle@localhost curl]$ curl "http://localhost:8080/autoupgrade/tasks"
{
    "total_tasks": 0,
```

```
    "tasks": [
    ]
}
```

親トピック: スタート・ガイド

AutoUpgradeのREST API

これらのREST APIは、Oracle REST Data Services (ORDS)のAutoUpgrade RESTサービス機能で使用できます。

- 新しいAutoUpgradeタスクの作成
  JSON構成ファイルを使用するかどうかに関係なく、REST APIを介してAutoUpgradeタスクを発行します。
- AutoUpgradeタスクの進行状況レポート・リクエスト
  `progress` APIを使用して、`taskid`で指定されたAutoUpgradeタスクの進行状況レポートを取得します。
- AutoUpgradeタスクのコンソール・リクエスト
  `console` APIを使用して、`taskid`で指定されたAutoUpgradeタスクのコンソールを取得します。
- AutoUpgradeのタスク詳細リクエスト
  `task` APIを使用して、`taskid`で指定されたタスクのAutoUpgradeタスクの詳細を取得します。
- AutoUpgradeタスクのタスク・リスト・リクエスト
  `tasks` APIを使用して、AutoUpgradeタスク・リストを取得します。
- AutoUpgradeタスクのログまたはログ・リスト
  `log` APIを使用してログを開くか、`taskid`で指定されたタスクのログ・リストを取得します。
- AutoUpgradeのタスク・リクエストのステータス・レポート
  `status` APIを使用して、`taskid`で指定されたAutoUpgradeタスクのステータス・レポートを取得します。

親トピック: AutoUpgrade REST API

新しいAutoUpgradeタスクの作成

JSON構成ファイルを使用するかどうかに関係なく、REST APIを介してAutoUpgradeタスクを発行します。

方法

POST

パス

```
/autoupgrade/task
```

使用上のノート

次の2つの使用方法があります。

1. AutoUpgradeタスクを発行します。タスクを発行するには、AutoUpgrade構成ファイルで指定されているデータベースがOracle REST Data Servicesサーバー・インスタンスと同じホスト上にある必要があり、ターゲット・データベース・リリースは、ソース・データベース・リリースからの直接アップグレードがサポートされているリリースである必要があります。AutoUpgrade構成ファイルは、HTTPまたはHTTPSリクエスト本文を介してJSON形式でアップロードする必要があります。

2. AutoUpgrade REST APIを使用して既存のタスクを再発行します。既存の`taskid`パラメータを指定してタスクを再発行すると、そのタスクに以前関連付けられていた構成ファイルが使用されるため、構成ファイルを指定する必要はありません。ただし、パラメータ`mode`および`restore_on_fail`には異なる値を指定できます。たとえば、`mode`パラ

メータがanalyzeに設定されているタスクを発行できます。チェックの失敗なしで分析タスクが完了した後、同じtaskidでタスクを再発行できますが、modeパラメータをdeployに変更してデータベースのアップグレードを続行します。

リクエスト

パス・パラメータ:

パラメータ	説明
mode	(必須) AutoUpgrade 処理モード。オプション: analyze、fixups、deploy、upgrade、postfixups。
restore_on_fail	(オプション)アップグレード中にリストア・ポイントを生成します。オプション: yes、no
taskid	(オプション) taskid で指定した既存の AutoUpgrade タスクを再実行します。

レスポンス

- 201レスポンス: 新しいAutoUpgradeタスクの説明。
- 400レスポンス: パラメータがない場合に返されます。

例

例A-1 AutoUpgradeタスク・リクエストの発行

次の例は、cURLを使用してRESTリソースに対するPOSTリクエストを発行することで、AutoUpgrade構成ファイルを発行し、分析モードで新しいAutoUpgradeタスクを作成する方法を示しています

```
curl --data-binary "@config.json" -X POST --header "Content-Type:application/json"
'http://localhost:8080/autoupgrade/task?mode=analyze'
```

この分析リクエストのJSON構成ファイル(config.json)には、Oracle Database 12c (12.2)からOracle Database 19cにアップグレードするように指定するソース・データベース・インスタンスdb122が1つあります。

```
{
"global": {
                "autoupg_log_dir": "/autoupgrade/test/glog"
        },
        "jobs": [{
                "start_time": "NOW",
                "source_home": "/databases/product/12.2.0/dbhome_1",
                "target_home": "/databases/product/19.0.0/dbhome_1",
                "sid": "db122",
                "log_dir": "/autoupgrade/test/log",
                "upgrade_node": "localhost",
                "target_version": "19.1"
        }]
}
```

このタスク・リクエストの戻りヘッダーの場所を使用して、AutoUpgradeタスクを表示できます。

```
Type:application/json" 'http://localhost:8080/autoupgrade/task?mode=analyze' -i
HTTP/1.1 201 Created
Date: Thu, 14 Oct 2021 19:52:59 GMT
Set-Cookie: JSESSIONID=node01efwo2229s7kj11rxw2l2mltc10.node0; Path=/
Expires: Thu, 01 Jan 1970 00:00:00 GMT
```

```
Content-Type: application/json
X-Frame-Options: SAMEORIGIN
Location: http://localhost:8080/autoupgrade/task?taskid=job_2021-10-14_15_53_00.085_0
Transfer-Encoding: chunked
```

この分析リクエストに対する201レスポンス本文には、AutoUpgradeジョブ・ステータスの説明が表示されます。JSON形式で返されます。

```
{
    "taskid": "job_2021-10-14_15_53_00.085_0",
    "status": "submitted",
    "message": "",
    "link": "http://localhost:8080/autoupgrade/task?taskid=job_2021-10-
14_15_53_00.085_0",
    "config": {
        "global": {
            "autoupg_log_dir": "/autoupgrade/test/glog"
        },
        "jobs": [
            {
                "start_time": "NOW",
                "source_home": "/databases/product/12.2.0/dbhome_1",
                "target_home": "/databases/product/19.1.0/dbhome_1",
                "sid": "db122",
                "log_dir": "/autoupgrade/test/log",
                "upgrade_node": "localhost",
                "target_version": "19.1"
            }
        ]
    }
}
```

例A-2 問題のあるリクエスト後のAutoUpgradeジョブのレスポンス

次の例では、AutoUpgradeジョブが無効なmodeパラメータを使用して発行されます。

```
curl --data-binary "@config.json" -X POST --header "Content-Type:application/json"
'http://localhost:8080/autoupgrade/task?mode=NULL'
```

問題のあるリクエストに対する400レスポンス・ヘッダーは次のとおりです。

```
Type:application/json" 'http://localhost:8080/autoupgrade/task?mode=null' -i
HTTP/1.1 400 Bad Request
Date: Wed, 20 Oct 2021 18:02:46 GMT
Content-Type: application/json
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
```

次の例は、この400レスポンス本文を示しています。JSON形式で返されます。

```
{
    "taskid": "job_2021-10-20_14_02_46.855_4",
    "status": "Bad Request",
    "message": "AutoUpgrade parameter mode is invalid.",
    "link": "",
    "config": null
}
```

例A-3 taskidを使用したAutoUpgradeジョブの再実行

この例では、タスクID job_2021-10-14_15_53_00.085_0が分析モードで再実行されます。

```
curl -X POST 'http://localhost:8080/autoupgrade/task?mode=analyze&taskid=job_2021-10-
14_15_53_00.085_0'
```

この再実行ジョブに対する201レスポンス・ヘッダーは次のとおりです。

```
HTTP/1.1 201 Created
Date: Mon, 10 Jan 2022 19:44:40 GMT
Set-Cookie: JSESSIONID=node0sjsoy3w4cntl13qnat979zc6n4.node0; Path=/
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Content-Type: application/json
X-Frame-Options: SAMEORIGIN
Location: http://localhost:8080/autoupgrade/task?taskid=job_2022-01-10_14_19_09.156_1
Transfer-Encoding: chunked
```

タスクの本文では、JSON形式のタスクについて説明します。

```
{    "taskid": "job_2021-10-14_15_53_00.085_0",    "status": "submitted",
"message": "",    "link": "http://localhost:8080/autoupgrade/task?taskid=job_2021-10-
14_15_53_00.085_0",
          "config": {
              "global": {
                  "autoupg_log_dir": "/autoupgrade/test/glog"
              },
              "jobs": [
                  {
                      "start_time": "NOW",
                      "source_home": "/databases/product/12.2.0/dbhome_1",
                      "target_home": "/databases/product/19.1.0/dbhome_1",
                      "sid": "db122",
                      "log_dir": "/autoupgrade/test/log",
                      "upgrade_node": "localhost",
                      "target_version": "19.1"
                  }
              ]
          } }
```

親トピック: AutoUpgradeのREST API

AutoUpgradeタスクの進行状況レポート・リクエスト

`progress` APIを使用して、`taskid`で指定されたAutoUpgradeタスクの進行状況レポートを取得します。

方法

GET

パス

```
/autoupgrade/progress
```

使用上のノート

`taskid`で指定したタスクの進行状況リクエストを発行すると、その`taskid`で識別されるAutoUpgradeタスクの進捗状況のレポートを受信します。

リクエスト

リクエスト・パラメータ:

パラメータ	説明
`taskid`	(必須)進行状況レポートを取得するタスクを指定します。

レスポンス

- 200レスポンス: taskidで指定したタスクの進行状況をレポートします。
- 400レスポンス: taskidがないことをレポートします
- 404レスポンス: taskidが無効であることをレポートします

例

例A-4 タスクの進行状況レポートのGETリクエストの発行

次の例は、cURLでAutoUpgrade taskidを指定して、AutoUpgradeタスクの進行状況を取得する方法を示しています。

```
curl http://localhost:8080/autoupgrade/progress?taskid=job_2021-10-20_14_44_57.11_1
```

これは、ジョブの前述のGETタスク進行状況レポートに対する200レスポンスです。

```
HTTP/1.1 200 OK
Date: Wed, 20 Oct 2021 19:56:32 GMT
Set-Cookie: JSESSIONID=node01rkapzylgd2tljf28bghba9bx6.node0; Path=/
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Content-Type: application/json
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
```

ジョブ・リクエストのこのGET進行状況レポートに対する200レスポンス本文は、次のとおりです(JSON形式)。


```
{
    "taskid": "job_2021-10-20_14_44_57.11_1",
    "status": "successful",
    "message": "",
    "autoupgrade_progress": {
        "totalJobs": 1,
        "totalFinishedJobs": 1,
        "startTime": "2021-10-20 14:44:59",
        "lastUpdateTime": "2021-10-20 14:45:10",
        "jobs": [
            {
                "sid": "db122",
                "jobNo": 100,
                "isCDB": false,
                "totalPercentCompleted": 100,
                "totalContainers": 1,
                "totalStages": 1,
                "additionalInfo": "",
                "lastUpdateTime": "2021-10-20 14:45:10",
                "stages": [
                    {
                        "stage": "PRECHECKS",
                        "percentCompleted": "100",
                        "totalChecks": 104,
                        "totalCompletedChecks": 104,
                        "additionalInfo": "",
                        "lastUpdateTime": "2021-10-20 14:44:59",
                        "containers": [
                            {
                                "container": "db122",
                                "totalChecks": 104,
                                "completedChecks": 104,
                                "succeededChecks": 91,
                                "failedChecks": 13,
                                "runningChecks": [
                                ],
                                "finishedChecks": [
                                    "PLUGIN_COMP_COMPATIBILITY",
                                    "AMD_EXISTS",
                                    "COMPATIBLE_NOT_SET",
                                    "COMPATIBLE_PARAMETER",
```

```
                                        "CASE_INSENSITIVE_AUTH",
                                        "CYCLE_NUMBER",
                                        "CREATE_WINDOWS_SERVICE",
                                        "AUDTAB_ENC_TS",
                                        "CONC_RES_MGR",
                                        "DATA_MINING_OBJECT",
                                        "DV_ENABLED",
                                        "AWR_EXPIRED_SNAPSHOTS",
                                        "DV_SIMULATION",
                                        "DEPEND_USR_TABLES",
                                        "FLASH_RECOVERY_AREA_SETUP",
                                        "FILES_NEED_RECOVERY",
                                        "INVALID_SYS_TABLEDATA",
                                        "INVALID_OBJECTS_EXIST",
                                        "JAVAVM_STATUS",
                                        "ORACLE_RESERVED_USERS",
                                        "HIDDEN_PARAMS",
                                        "JVM_MITIGATION_PATCH",
                                        "TRGOWNER_NO_ADMNDBTRG",
                                        "MV_REFRESH",
                                        "PENDING_DST_SESSION",
                                        "RAISE_COMPATIBLE",
                                        "COMPATIBLE_PFILES",
                                        "INVALID_USR_TABLEDATA",
                                        "POST_JVM_MITIGAT_PATCH",
                                        "PURGE_RECYCLEBIN",
                                        "PRE_FIXED_OBJECTS",
                                        "SYNC_STANDBY_DB",
                                        "SYS_DEFAULT_TABLESPACE",
                                        "TWO_PC_TXN_EXIST",
                                        "UNDERSCORE_EVENTS",
                                        "UNIAUD_TAB",
                                        "XBRL_VERSION",
                                        "PARAMETER_DEPRECATED",
                                        "XDB_RESOURCE_TYPE",
                                        "PARAMETER_NEW_NAME_VAL",
                                        "PARAMETER_OBSOLETE",
                                        "PARAMETER_RENAME",
                                        "MIN_ARCHIVE_DEST_SIZE",
                                        "NEW_TIME_ZONES_EXIST",
                                        "PARAMETER_MIN_VAL",
                                        "DISK_SPACE_FOR_RECOVERY_AREA",
                                        "PA_PROFILE",
                                        "UPG_BY_STD_UPGRD",
                                        "MIN_RECOVERY_AREA_SIZE",
                                        "OLS_VERSION",
                                        "ROLLBACK_SEGMENTS",
                                        "ARCHIVE_MODE_ON",
                                        "PGA_AGGREGATE_LIMIT",
                                        "APEX_MANUAL_UPGRADE",
                                        "TABLESPACES",
                                        "OLAP_PAGE_POOL_SIZE",
                                        "STANDARD_EDITION",
                                        "TEMPTS_NOTEMPFILE",
                                        "TS_FORUPG_STATUS",
                                        "APPL_PRINCIPAL",
                                        "TDE_IN_USE",
                                        "STREAMS_SETUP",
                                        "DIR_SYMLINKS",
                                        "ORDIM_DESUPPORT",
                                        "UTLRP_RUN_SERIAL",
                                        "DESUPPORT_RAC_ON_SE",
                                        "COMPATIBLE_GRP",
                                        "EDS_EXISTS",
                                        "UNIAUD_RECORDS_IN_FILE",
                                        "JOB_TABLE_INTEGRITY",
                                        "TARGET_CDB_COMPATIBILITY",
```

```
                                    "DUPLICATE_DB_HASHES",
                                    "UNPLUG_PLUG_UPGRADE",
                                    "ENABLE_LOCAL_UNDO",
                                    "AUTO_LOGIN_KEYSTORE_REQUIRED",
                                    "NO_KEYSTORE_FILES",
                                    "INFORM_DROP_GRP",
                                    "PRE_DISABLE_BCT_UPG",
                                    "MAX_STRING_SIZE_ON_DB",
                                    "POST_RECOMPILE_IN_CDB",
                                    "TEMPTS_ALLOFFLINE",
                                    "PRE_RECOMPILE_IN_CDB",
                                    "MAX_DB_FILES_EXCEEDED",
                                    "POST_DISABLE_BCT_UPG",
                                    "TARGET_HOME_REGISTERED_INVENTORY",
                                    "ISRAC_SWITCHEDON_TARGETHOME",
                                    "KEYSTORE_CONFLICT",
                                    "WALLET_ROOT_KEYSTORE",
                                    "ORACLE_HOME_KEYSTORE",
                                    "DEPLOY_JOB_VALIDATIONS",
                                    "DUPLIC_SYS_SYSTEM_OBJS"
                                ],
                                "checksWithExecutionError": [
                                ],
                                "checksFailed": [
                                    "DICTIONARY_STATS",
                                    "POST_DICTIONARY",
                                    "POST_FIXED_OBJECTS",
                                    "OLD_TIME_ZONES_EXIST",
                                    "MANDATORY_UPGRADE_CHANGES",
                                    "RMAN_RECOVERY_VERSION",
                                    "TABLESPACES_INFO",
                                    "ORDIM_INFO_DESUPPORT",
                                    "POST_UTLRP",
                                    "COMPONENT_INFO",
                                    "INVALID_ORA_OBJ_INFO",
                                    "INVALID_APP_OBJ_INFO",
                                    "TIMESTAMP_MISMATCH"
                                ],
                                "additionalInfo": "",
                                "lastUpdateTime": "2021-10-20 14:45:10"
                            }
                        ]
                    }
                ],
                "summary": [
                    {
                        "stage": "PRECHECKS",
                        "complete": "Yes",
                        "duration": "11.08S"
                    }
                ]
            }
        ]
    }
}
```


例A-5 タスクIDが無効なタスク進行状況レポート・リクエスト

この例では、タスク・レポートの進行状況が無効なタスクIDに対して発行されます。

```
curl http://localhost:8080/autoupgrade/progress?taskid=null
```

前述の無効なリクエストに対する404レスポンス・ヘッダーは次のとおりです。

```
HTTP/1.1 404 Not Found
Date: Wed, 20 Oct 2021 18:49:12 GMT
```

```
Set-Cookie: JSESSIONID=node06xkycdfv7k1f1bus3q43zefum2.node0; Path=/
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Content-Type: application/json
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
```

このリクエストに対する404レスポンス本文は次のとおりです。JSON形式で返されます。

```
{
    "taskid": "null",
    "status": "Bad Request",
    "message": "taskid is invalid.",
    "link": "",
    "config": ""
}
}
```

親トピック: AutoUpgradeのREST API

AutoUpgradeタスクのコンソール・リクエスト

console APIを使用して、taskidで指定されたAutoUpgradeタスクのコンソールを取得します。

方法

GET

パス

```
/autoupgrade/console
```

使用上のノート

taskidで指定されたAutoUpgradeタスクのコンソール出力を取得します。

リクエスト

パス・パラメータ:

パラメータ	説明
taskid	(必須)コンソール出力を取得するタスクを指定します。

レスポンス

- 200レスポンス: AutoUpgradeコンソールが開きます。
- 400レスポンス: taskidがないことをレポートします
- 404レスポンス: taskidが無効であることをレポートします

例

例A-6 タスク・コンソールGETリクエストの発行

次の例は、cURLでAutoUpgrade taskidを指定してコンソール・リクエストを発行する方法を示しています。

```
curl http://localhost:8080/autoupgrade/console?taskid=job_2021-10-20_14_44_57.11_1
```

前述のGETタスク・コンソール・リクエストに対する200レスポンス・ヘッダーは、次のとおりです。

```
HTTP/1.1 200 OK
```

```
Date: Wed, 20 Oct 2021 20:09:04 GMT
Set-Cookie: JSESSIONID=node01hwo60q3c5irz1pnn4fa8hhm2k9.node0; Path=/
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Content-Type: plain/text
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
```

このリクエストに対する200レスポンス本文は、プレーン(テキスト)形式で返されるAutoUpgradeジョブの進行状況をレポートします。

```
AutoUpgrade tool launched with default options
Processing config file ...
+--------------------------------+
| Starting AutoUpgrade execution |
+--------------------------------+
1 databases will be analyzed
Job 100 database db122
Job 100 completed
------------------- Final Summary --------------------
Number of databases            [ 1 ]

Jobs finished                  [1]
Jobs failed                    [0]
Jobs restored                  [0]
Jobs pending                   [0]

Please check the summary report at:
/databases/working/job_2021-10-
20_14_44_57.11_1/cfgtoollogs/upgrade/auto/status/status.html
/databases/working/job_2021-10-
20_14_44_57.11_1/cfgtoollogs/upgrade/auto/status/status.log
```

例A-7 タスクIDが無効なコンソール・リクエストのレスポンス・ヘッダー

この例では、無効なタスクIDに対してコンソール・リクエストが発行されます。

```
curl http://localhost:8080/autoupgrade/console?taskid=null
```

前述のリクエストでは、コンソール・リクエストに次の404ヘッダー・レスポンスが返されます。

```
HTTP/1.1 404 Not Found
Date: Wed, 20 Oct 2021 18:49:12 GMT
Set-Cookie: JSESSIONID=node06xkycdfv7k1f1bus3q43zefum2.node0; Path=/
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Content-Type: application/json
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
```

このコンソール・リクエストに対する404レスポンス本文は次のとおりです。JSON形式で返されます。

```
{
    "taskid": "null",
    "status": "Bad Request",
    "message": "taskid is invalid.",
    "link": "",
    "config": ""
}
```

親トピック: AutoUpgradeのREST API

AutoUpgradeのタスク詳細リクエスト

`task` APIを使用して、`taskid`で指定されたタスクのAutoUpgradeタスクの詳細を取得します。

方法

GET

パス

```
/autoupgrade/task
```

使用上のノート

taskidを指定すると、そのtaskidに関連付けられているタスクの詳細を取得します。

たとえば:

```
curl http://localhost:8080/autoupgrade/task?taskid=job_2021-10-20_14_44_57.11_1
```

リクエスト

パス・パラメータ:

パラメータ	説明
taskid	(必須)詳細を取得するタスクのシステム生成識別子

レスポンス

- 200レスポンス: taskidで指定したタスクの詳細をレポートします。
- 400レスポンス: taskidがないことをレポートします
- 404レスポンス: taskidが無効であることをレポートします

例A-8 タスク詳細GETリクエストの発行

次の例は、cURLでAutoUpgrade taskidを指定してAutoUpgrade getタスク詳細リクエストを発行する方法を示しています。

```
curl http://localhost:8080/autoupgrade/task?taskid=job_2021-10-20_14_44_57.11_1
```

前述のgetタスク詳細リクエストに対する200レスポンスは、次のとおりです。Locationヘッダーは、AutoUpgradeタスクの表示に使用できるURIを返します。

```
HTTP/1.1 200 OK
Date: Wed, 20 Oct 2021 18:45:20 GMT
Set-Cookie: JSESSIONID=node0tl6lz63xj3xd11i05p8bfmw8u1.node0; Path=/
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Content-Type: application/json
X-Frame-Options: SAMEORIGIN
Location: http://localhost:8080/autoupgrade/task?taskid=job_2021-10-20_14_44_57.11_1
Transfer-Encoding: chunked
```

このGETタスク詳細リクエストに対する200レスポンス本文は次のとおりです。JSON形式で返されます:

```
{
    "taskid": " job_2021-10-20_14_44_57.11_1",
    "status": "submitted",
    "message": "",
    "link": "http://localhost:8080/autoupgrade/task?taskid=job_2021-10-
14_15_53_00.085_0",
    "config": {
        "global": {
            "autoupg_log_dir": "/autoupgrade/test/glog"
```

```
        },
        "jobs": [
            {
                "dbname": "db122",
                "start_time": "NOW",
                "source_home": "/databases/product/12.2.0/dbhome_1",
                "target_home": "/databases/product/19c/dbhome_1",
                "sid": "db122",
                "log_dir": "/autoupgrade/test/log",
                "upgrade_node": "localhost",
                "target_version": "19.1"
            }
        ]
    }
}
```

例A-9 taskidが無効なタスク詳細リクエスト

無効なtaskidを持つGETリクエストが発行されます。

```
curl http://localhost:8080/autoupgrade/task?taskid=null
```

無効なリクエストに対する404レスポンス・ヘッダーは次のとおりです。

```
{HTTP/1.1 404 Not Found
Date: Wed, 20 Oct 2021 18:49:12 GMT
Set-Cookie: JSESSIONID=node06xkycdfv7k1f1bus3q43zefum2.node0; Path=/
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Content-Type: application/json
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
```

このリクエストに対する404レスポンス本文は次のとおりです。JSON形式で返されます。

```
{
    "taskid": "null",
    "status": "Bad Request",
    "message": "taskid is invalid.",
    "link": "",
    "config": ""
}
```

親トピック: AutoUpgradeのREST API

AutoUpgradeタスクのタスク・リスト・リクエスト

tasks APIを使用して、AutoUpgradeタスク・リストを取得します。

方法

GET

パス

```
/autoupgrade/tasks
```

リクエスト

パラメータはありません。

レスポンス

200レスポンス: すべてのタスクの詳細をレポートします。

使用上のノート

このメソッドは、タスクのリストを取得するために使用します。

例A-10 タスク・リストのGETリクエストの発行

次の例は、cURLを使用してAutoUpgradeタスク・リストを取得する方法を示しています。

```
curl http://localhost:8080/autoupgrade/tasks
```

次の例は、GETタスク・リスト・リクエストに対するレスポンス・ヘッダーを示しています。

```
HTTP/1.1 200 OK
Date: Wed, 20 Oct 2021 19:51:46 GMT
Set-Cookie: JSESSIONID=node0yawyihk3msz91eeauxijfaw6z3.node0; Path=/
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Content-Type: application/json
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
```

このリスト・リクエストに対する200レスポンス本文は次のとおりです。JSON形式で返されます。

```
{
    "total_tasks": 2,
    "tasks": [
        {
            "mode": "analyze",
            "taskid": "job_2021-10-20_14_44_57.11_1",
            "config": {
                "jobs": [
                    {
                        "source_home": "/databases/product/12.2.0/dbhome_1",
                        "sid": "db122"
                    }
                ]
            },
            "link": "http://localhost:8080/autoupgrade/task?taskid=job_2021-10-
20_14_44_57.11_1"
        },
        {
            "mode": "analyze",
            "taskid": "job_2021-10-20_15_54_01.568_2",
            "config": {
                "jobs": [
                    {
                        "source_home": "/databases/product/12.2.0/dbhome_1",
                        "sid": "db122"
                    }
                ]
            },
            "link": "http://localhost:8080/autoupgrade/task?taskid=job_2021-10-
20_15_54_01.568_2"
        }
    ]
}
```

親トピック: AutoUpgradeのREST API

AutoUpgradeタスクのログまたはログ・リスト

`log` APIを使用してログを開くか、`taskid`で指定されたタスクのログ・リストを取得します。

方法

GET

パス

```
/autoupgrade/log
```

使用上のノート

タイプまたは名前を指定したログ・リクエストを発行すると、taskidで指定されたタスクのAutoUpgradeログを受け取ります。タイプまたは名前を指定しない場合、taskidで指定されたタスクのすべてのHTML、JSONおよびXMLログ・ファイルを含むAutoUpgradeログ・リストを受け取ります。

リクエスト

リクエスト・パラメータ:

パラメータ	説明
taskid	(必須)ログまたはリスト・ログを開く AutoUpgrade タスク ID を指定します。
type	(オプション) AutoUpgrade ログのタイプを指定します。オプション: autolog: AutoUpgrade ログ・ファイル errlog: エラー・ログ・ファイル userlog: ユーザー・ログ・ファイル
name	(オプション)タスク ID のログ・リスト内の特定のログのファイル名を指定します。

レスポンス

- 200レスポンス: AutoUpgradeは、指定したtaskidのAutoUpgradeタスクのerr/user/autoログを開きます。
- 400レスポンス: taskidがないことをレポートします
- 404レスポンス: taskidが無効であることをレポートします

例

例A-11 タスク・ログのGETリクエストの発行

次の例は、cURLでAutoUpgrade taskidおよびtypeを指定してAutoUpgradeユーザー・ログを取得する方法を示しています。

```
curl "http://localhost:8080/autoupgrade/log?type=userlog&taskid=job_2021-10-
20_14_44_57.11_1"
```

userlogとしてtypeが指定されている前述のタスク・ログ・リクエストに対する200レスポンス・ヘッダーは次のとおりです。

```
HTTP/1.1 200 OK
Date: Wed, 20 Oct 2021 20:30:02 GMT
Set-Cookie: JSESSIONID=node0jrq0bg40ya86mr8icb87d8z16.node0; Path=/
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Content-Type: plain/text
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
```

このリクエストに対する200レスポンス本文は次のとおりです。user logとしてtypeが指定されているタスク・ログに対してプレーン(テキスト)形式で返されます。

```
2021-10-20 14:44:57.737 INFO
build.hash:71778e3
build.version:23.0.0
build.date:2021/05/05 10:42:40 -0700
build.max_target_version:23
build.supported_target_versions:12.2,18,19,21,23
build.type:beta
build.hash_date:2021/04/15 15:06:03 -0700
build.label:rest_api

2021-10-20 14:44:57.791 INFO Loading user config file metadata
2021-10-20 14:44:58.598 INFO The target_version parameter was updated from 19.1 to
19.3.0.0.0 due to finding a more accurate value
2021-10-20 14:44:58.844 INFO srvctl command is not available or user does not have
permission to execute it
2021-10-20 14:44:59.163 INFO Finished processing dbEntry job_0
2021-10-20 14:44:59.475 INFO Current settings Initialized
2021-10-20 14:44:59.576 INFO Job 100 database db122
2021-10-20 14:45:11.685 INFO Job 100 completed
```

例A-12 タスクIDが無効なログ・リクエストのレスポンス・ヘッダー

この例では、GETログ・リクエストが、無効なタスクIDに対して発行されます。

```
curl http://localhost:8080/autoupgrade/log?taskid=null
```

このリクエストは、次の404ヘッダー・レスポンスを返します。

```
HTTP/1.1 404 Not Found
Date: Wed, 20 Oct 2021 18:49:12 GMT
Set-Cookie: JSESSIONID=node06xkycdfv7k1f1bus3q43zefum2.node0; Path=/
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Content-Type: application/json
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
```

無効なtaskidに対する404レスポンス本文は次のとおりです。JSON形式で返されます。

```
{
    "taskid": "null",
    "status": "Bad Request",
    "message": "taskid is invalid.",
    "link": "",
    "config": ""
}
```

¥

例A-13 タスクIDのログ・リストのレスポンス本文

次の例は、cURLでAutoUpgrade taskidを指定してAutoUpgradeログ・リストを取得する方法を示しています。

```
 curl 'http://localhost:8080/autoupgrade/log?taskid='job_2022-01-10_14:19:09.156_1
```

次の例は、taskidのログ・リストを示す200レスポンス本文を示しています。JSON形式で返されます。

```
{
    "logs": [
        {
            "filename": "cfgtoollogs/upgrade/auto/autoupgrade.log",
```

```
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=cfgtoollogs/upgrade/auto/autoupgrade.log"
        },
        {
            "filename": "cfgtoollogs/upgrade/auto/autoupgrade_user.log",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=cfgtoollogs/upgrade/auto/autoupgrade_user.log"
        },
        {
            "filename": "cfgtoollogs/upgrade/auto/autoupgrade_err.log",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=cfgtoollogs/upgrade/auto/autoupgrade_err.log"
        },
        {
            "filename": "cfgtoollogs/upgrade/auto/status/status.json",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=cfgtoollogs/upgrade/auto/status/status.json"
        },
        {
            "filename": "cfgtoollogs/upgrade/auto/status/progress.json",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=cfgtoollogs/upgrade/auto/status/progress.json"
        },
        {
            "filename": "cfgtoollogs/upgrade/auto/status/status.html",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=cfgtoollogs/upgrade/auto/status/status.html"
        },
        {
            "filename": "cfgtoollogs/upgrade/auto/status/status.log",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=cfgtoollogs/upgrade/auto/status/status.log"
        },
        {
            "filename": "db122/100/autoupgrade_20220110.log",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/100/autoupgrade_20220110.log"
        },
        {
            "filename": "db122/100/autoupgrade_20220110_user.log",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/100/autoupgrade_20220110_user.log"
        },
        {
            "filename": "db122/100/autoupgrade_err.log",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/100/autoupgrade_err.log"
        },
        {
            "filename": "db122/100/prechecks/prechecks_db122.log",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/100/prechecks/prechecks_db122.log"
        },
        {
            "filename": "db122/100/prechecks/db122_checklist.xml",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/100/prechecks/db122_checklist.xml"
        },
        {
            "filename": "db122/100/prechecks/db122_checklist.json",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/100/prechecks/db122_checklist.json"
        },
        {
            "filename": "db122/100/prechecks/db122_preupgrade.html",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/100/prechecks/db122_preupgrade.html"
```

```
        },
        {
            "filename": "db122/100/prechecks/upgrade.xml",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/100/prechecks/upgrade.xml"
        },
        {
            "filename": "db122/100/prechecks/db122_preupgrade.log",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/100/prechecks/db122_preupgrade.log"
        },
        {
            "filename": "db122/101/autoupgrade_20220110.log",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/101/autoupgrade_20220110.log"
        },
        {
            "filename": "db122/101/autoupgrade_20220110_user.log",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/101/autoupgrade_20220110_user.log"
        },
        {
            "filename": "db122/101/autoupgrade_err.log",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/101/autoupgrade_err.log"
        },
        {
            "filename": "db122/101/prechecks/prechecks_db122.log",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/101/prechecks/prechecks_db122.log"
        },
        {
            "filename": "db122/101/prechecks/db122_checklist.xml",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/101/prechecks/db122_checklist.xml"
        },
        {
            "filename": "db122/101/prechecks/db122_checklist.json",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/101/prechecks/db122_checklist.json"
        },
        {
            "filename": "db122/101/prechecks/db122_preupgrade.html",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/101/prechecks/db122_preupgrade.html"
        },
        {
            "filename": "db122/101/prechecks/upgrade.xml",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/101/prechecks/upgrade.xml"
        },
        {
            "filename": "db122/101/prechecks/db122_preupgrade.log",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/101/prechecks/db122_preupgrade.log"
        },
        {
            "filename": "db122/102/autoupgrade_20220110.log",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/102/autoupgrade_20220110.log"
        },
        {
            "filename": "db122/102/autoupgrade_20220110_user.log",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/102/autoupgrade_20220110_user.log"
        },
        {
```

```
            "filename": "db122/102/autoupgrade_err.log",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/102/autoupgrade_err.log"
        },
        {
            "filename": "db122/102/prechecks/prechecks_db122.log",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/102/prechecks/prechecks_db122.log"
        },
        {
            "filename": "db122/102/prechecks/db122_checklist.xml",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/102/prechecks/db122_checklist.xml"
        },
        {
            "filename": "db122/102/prechecks/db122_checklist.json",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/102/prechecks/db122_checklist.json"
        },
        {
            "filename": "db122/102/prechecks/db122_preupgrade.html",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/102/prechecks/db122_preupgrade.html"
        },
        {
            "filename": "db122/102/prechecks/upgrade.xml",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/102/prechecks/upgrade.xml"
        },
        {
            "filename": "db122/102/prechecks/db122_preupgrade.log",
            "link": "http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/102/prechecks/db122_preupgrade.log"
        }
    ]
}
```

例A-14 nameパラメータを使用した特定のログのリクエスト

次の例は、nameパラメータにファイル名を指定し、ログを生成したタスクのtaskidを指定して、タスクIDのログ・リスト内の特定のログ・ファイルをリクエストする方法を示しています。

```
curl 'http://localhost:8080/autoupgrade/log?taskid=job_2022-01-
10_14_19_09.156_1&name=db122/102/autoupgrade_20220110_user.log'
```

このリクエストでは、指定されたファイル名のログ・ファイル情報を表示する200レスポンス本文が返されます。

```
2022-01-10 14:44:43.447 INFO
build.hash:71778e3
build.version:23.0.0
build.date:2021/05/05 10:42:40 -0700
build.max_target_version:23
build.supported_target_versions:12.2,18,19,21,23
build.type:beta
build.hash_date:2021/04/15 15:06:03 -0700
build.label:rest_api

2022-01-10 14:44:49.529 INFO Analyzing db122, 104 checks will run using 4 threads
```

親トピック: AutoUpgradeのREST API

AutoUpgradeのタスク・リクエストのステータス・レポート

`status` APIを使用して、`taskid`で指定されたAutoUpgradeタスクのステータス・レポートを取得します。

方法

GET

パス

```
/autoupgrade/status
```

使用上のノート

JSON形式でAutoUpgradeタスク・ステータスを受信するには、AutoUpgrade `taskid`を発行します。

リクエスト

パス・パラメータ:

パラメータ	説明
`taskid`	(必須)ステータスを取得するタスクのシステム生成識別子。

レスポンス

- 200レスポンス: `taskid`で指定したタスクのステータスをレポートします。
- 400レスポンス: `taskid`がないことをレポートします
- 404レスポンス: `taskid`が無効であることをレポートします

例

例A-15 タスク・ステータスGETリクエストの発行

次の例は、cURLでAutoUpgradeタスク・ステータス・リクエストを取得する方法を示しています。

```
curl http://localhost:8080/autoupgrade/status?taskid=job_2021-10-20_14_44_57.11_1
```

これは、前述のGETタスク・ステータス・レポート・リクエストに対する200レスポンス・ヘッダーです:

```
HTTP/1.1 200 OK
Date: Wed, 20 Oct 2021 19:51:46 GMT
Set-Cookie: JSESSIONID=node0yawyihk3msz91eeauxijfaw6z3.node0; Path=/
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Content-Type: application/json
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
```

これは、JSON形式のGETタスク・ステータス・レポート・リクエストに対するレスポンス本文です:

```
{
    "taskid": "job_2021-10-20_14_44_57.11_1",
    "status": "successful",
    "message": "",
    "autoupgrade_status": {
        "totalJobs": 1,
        "lastUpdateTime": "2021-10-20 14:45:11",
        "jobs": [
            {
                "sid": "db122",
                "jobNo": 100,
```

```
                "logDirectory": "/databases/working/job_2021-10-
20_14_44_57.11_1/db122/100",
                "conNumber": 1,
                "lastUpdateTime": "2021-10-20 14:45:11",
                "modules": [
                    {
                        "moduleName": "PRECHECKS",
                        "status": 0,
                        "errors": [
                        ],
                        "lastUpdateTime": "2021-10-20 14:45:11"
                    }
                ]
            }
        ]
    }
}
```

例A-16 taskidが無効なタスク・ステータスGETリクエストの発行

次の例では、AutoUpgrade GETタスク・ステータス・リクエストが、無効なtaskidパラメータで発行されます。

```
curl http://localhost:8080/autoupgrade/status?taskid=null
```

前述の無効なGETタスク・ステータス・リクエストに対する404レスポンス・ヘッダーは、次のとおりです。

```
HTTP/1.1 404 Not Found
Date: Wed, 20 Oct 2021 18:49:12 GMT
Set-Cookie: JSESSIONID=node06xkycdfv7k1f1bus3q43zefum2.node0; Path=/
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Content-Type: application/json
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
```

この無効なGETタスク・ステータス・リクエストに対するレスポンス本文は、次のとおりです(JSON形式)。

```
{
    "taskid": "null",
    "status": "Bad Request",
    "message": "taskid is invalid.",
    "link": "",
    "config": ""
}
```

親トピック: AutoUpgradeのREST API

B Oracle Databaseアップグレード・ユーティリティ

Oracleアップグレード・ユーティリティ・スクリプトは、Oracle Databaseのアップグレードの実行に役立ちます。

- Oracle Databaseのアップグレード用スクリプト
  Oracleには、アップグレードの前、途中および後に実行するアップグレード用のツールおよびスクリプトのセットがあります。
- アップグレード前情報ツールおよびAutoUpgradeのアップグレード前処理
  AutoUpgradeを`preupgrade`句とともに使用して、アップグレード前情報ツール(`preupgrade.jar`)を介して以前に提供されたものと同じ機能を取得する方法について説明します。

Oracle Databaseのアップグレード用スクリプト

Oracleには、アップグレードの前、途中および後に実行するアップグレード用のツールおよびスクリプトのセットがあります。

ノート:

Oracle に付属する一部のスクリプトは、UPGRADE モードでは実行できません。

次の表に、様々なスクリプトおよびツールとそれぞれの説明を示します。

表B-1 アップグレード・スクリプト、アップグレード後スクリプトおよびダウングレード・スクリプト

スクリプト	説明
`catcon.pl`	UPGRADE モードで実行する必要があります。このスクリプトは、CDB をアップグレードする際に使用されます。
`catctl.pl`	パラレル・アップグレード・ユーティリティ(`catctl.pl` とシェル・スクリプト `dbupgrade`)。これらのスクリプトは、UPGRADE モードで実行する必要があります。Oracle Database 12c 以上のリリースでは、パラレル・アップグレード・ユーティリティによって、SQL アップグレード・ユーティリティ `catupgrd.sql` が置き換えられています。パラレル・アップグレード・ユーティリティを使用すると、アップグレード・スクリプトとアップグレード処理をパラレルに実行できます。この機能では、サーバーの CPU 容量が最大限に利用されるため、アップグレード時間を短縮できます。DBUA ではこのツールが使用されます。
`dbdowngrade`	`catdwgrd.sql` スクリプトをコールするシェル・スクリプト・ユーティリティ。`catcon.pl` スクリプトのコールでダウングレードの推奨値が使用されることを確認します。この機能は、アップグレード時に過剰なスレッドが生成されたために発生するエラーの削減に役立ちます。マルチテナント・アーキテクチャ(CDB)のダウングレードに特に役立ちます。このスクリプトの使用は、Oracle の推奨ダウングレード手順です。
`dbupgrade`、`dbupgrade.cmd`	`catctl.pl` スクリプトを呼び出すシェル・スクリプト。これらのシェル・スクリプトは、Oracle Database 12c リリース 2 (12.2)以上で使用できます。これらのスクリプトによって、シェル・コマンド・プロンプトで `dbupgrade` コマンドを入力できます。これらのスクリプトは、デフォルト値を使用して実行するか、Perl プロンプトから `catctl.pl` を実行する際に使用するものと同じ入力パラメータを使用して実行できます。Linux およ

スクリプト	説明
	び UNIX システムでは dbupgrade を使用し、Windows システムでは dbupgrade.cmd を使用します。
catdwgrd.sql	これは、ダウングレード・スクリプトで、アップグレードする前の以前のリリースへダウングレードする手順で使用します。
catnoamd.sql	UPGRADE モードでは実行しないでください。アップグレード後にのみ、このスクリプトを実行します。catnoamd.sql スクリプトは、OLAP カタログのコンポーネント(AMD とも呼ばれます)を削除します。
catnoexf.sql	アップグレード・プロセスによって、このスクリプトが実行されますが、アップグレード前に catnoexf.sql スクリプトを実行できます。このスクリプトを UPGRADE モードで実行しないでください。catnoexf.sql スクリプトによって、このリリースでサポートが終了した Expression Filter (EXF)およびデータベース Rules Manager (RUL)コンポーネントが削除されます。
catuppst.sql	このスクリプトは、DBUA を通じて実行するか、手動で実行する必要があります(手動アップグレードを実行する場合)。 DBUA では、catuppst.sql は自動的に実行されます。手動アップグレードの場合のみ、このスクリプトを個別に実行する必要があります。 これは、UPGRADE モードで実行しないでください。ORACLE_HOME/rdbms/admin ディレクトリにある catuppst.sql を実行し、データベースを UPGRADE モードにする必要がない残りのアップグレード操作を実行します。Oracle バンドルパッチまたはパッチ・セット更新(PSU または BP)が Oracle ホームにインストールされている場合は、このスクリプトによってそのパッチ・セット更新がデータベースに自動的に適用されます。 注意: 手動アップグレードを実行する場合に、catuppst.sql を実行しないと、時間とともにデータベースのパフォーマンスが低下します。
catuptabdata.sql	catuptabdata.sql スクリプトは、catuppst.sql によって自動的に実行され、アップグレード時の Oracle 管理タイプに対する変更によって影響を受けるすべての Oracle 管

スクリプト	説明
	理表に ALTER TABLE UPGRADE が実行されます。 catuptabdata.sql スクリプトは、手動で実行して、Oracle 管理タイプに対する変更のためにアップグレードが必要なすべての Oracle 管理表をアップグレードできます。コマンドは、SYSDBA システム権限が付与された、SYSDBA として接続しているユーザー・アカウントで実行する必要があります。
emdwgrd	このスクリプトを UPGRADE モードで実行しないでください。emdwgrd ユーティリティは、リリース 11g データベースの Oracle Enterprise Manager Database Control (DB Control)構成ファイルおよびデータを保存します。 Oracle Database 18c を Oracle Database 11g (11.2)にダウングレードする必要があり、Oracle DB Control をリストアする必要があると考えられる場合、アップグレードの前に emdwgrd を実行する必要があります。 関連項目: Oracle サポートのドキュメント ID 1484775.1。
emremove.sql	emremove.sql スクリプトは、Oracle Enterprise Manager 関連のスキーマおよびオブジェクトを削除します。このスクリプトを使用して、DB Control を手動で削除します。emremove.sql をアップグレード処理の前に実行することで、停止時間が最小化されます。パラレル・アップグレード・ユーティリティおよび DBUA が自動的にこのスクリプトを実行するため、これはオプションのアップグレード前ステップです。 注意: ダウングレードして DB Control をリストアするオプションを使用するために DB Control の構成およびデータを保持する場合は、emdwgrd を使用した手順を最初に行う必要があります。
olspreupgrade.sql	このスクリプトを UPGRADE モードで実行しないでください。olspreupgrade.sql スクリプトは事前処理スクリプトで、Oracle Label Security (OLS)および Oracle Database Vault を使用する Oracle Database リリース 12.1 より前のデータベースをアップグレードする場合に必要です。

スクリプト	説明
postupgrade_fixups.sql	postupgrade_fixups.sql スクリプトは、Oracle Database に付属しています。アップグレード後に、postupgrade_fixups.sql を実行します。このスクリプトは、DBUA によって自動的に実行されますが、アップグレード後にいつでもこのスクリプトを実行することができます。
preupgrd.sql	(12.2 以上のリリースで廃止) preupgrd.sql アップグレード前情報ツールは、Oracle Database 12c リリース 1 (12.1)に付属しています。データベースを分析するために、preupgrd.sql をアップグレード前に実行します。アップグレード前情報ツールでは、DBUA が確認する項目のプレビューが提供され、ソース・データベースでフラグ付けされている問題を解決するために実行できる修正スクリプトと呼ばれるスクリプトが生成されます。リリース 12.1 で、utlu121i.sql スクリプトが preupgrd.sql スクリプトに置き換えられました。リリース 12.2 で、preupgrd.sql は preupgrade.jar に置き換えられています。
preupgrade.jar	preupgrade.jar アップグレード前情報ツールは、Oracle Database 12c リリース 2 (12.2)に付属しています。preupgrade.jar の機能は、preupgrade.jar によって置き換えられた preupgrd.sql と同じです。
utlrp.sql	utlrp.sql を使用して、ストアド PL/SQL および Java コードを再コンパイルします。DBUA ではこのスクリプトが自動的に実行されます。手動でアップグレードする場合、このスクリプトを実行する必要があります。
utlusts.sql	utlusts.sql アップグレード後の状態ツールは、Oracle Database に付属していて、DBA_REGISTRY 内の各コンポーネントのバージョンおよびアップグレード経過時間が表示されます。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いつでも実行できます。
utluptabdata.sql	アップグレード後に utluptabdata.sql スクリプトを実行して、アップグレード時に変更された Oracle 管理タイプに依存するすべてのユーザー表に ALTER TABLE UPGRADE コマンドを実行します。 utluptabdata.sql スクリプトは、アップグレードするすべての表に対する ALTER TABLE システム権限が割り当てら

スクリプト	説明
	れたユーザー・アカウントを使用するか、SYSDBA として接続している SYSDBA システム権限を持つユーザー・アカウントを使用して実行する必要があります。 ユーザー表は、データベースのアップグレード時に新しいタイプのバージョンに自動的にアップグレードされないため、ユーザー表領域が READ ONLY に設定された状態でアップグレードを実行できます。

親トピック: Oracle Databaseアップグレード・ユーティリティ

アップグレード前情報ツールおよびAutoUpgradeのアップグレード前処理

AutoUpgradeを`preupgrade`句とともに使用して、アップグレード前情報ツール(`preupgrade.jar`)を介して以前に提供された機能と同じ機能を取得する方法を学習します。

Oracle Database 19cでは、アップグレード前のチェックにAutoUpgradeツールを使用することをお薦めします。アップグレード前情報ツールを使用して以前に取得したすべての機能が、AutoUpgradeツールを使用して提供されるようになりました AutoUpgradeツールの使用方法の詳細は、『*Oracle Database*アップグレード・ガイド』を参照してください。

- アップグレード前情報ツールについて
  システムがアップグレード可能かどうかを判断するには、従来のアップグレード前情報ツール(`preupgrade.jar`)を使用できます。
- アップグレード前情報ツールによって生成されるアップグレード前スクリプト
  アップグレード前情報ツールによって生成されるアップグレード前スクリプトを実行することによって、新しいOracle Databaseリリースへのアップグレード前に多くの問題を修正できます。
- アップグレード前情報ツールによって生成されるアップグレード後スクリプト
  アップグレード後に、アップグレード前情報ツールによって生成されるアップグレード後スクリプトを実行することによって、アップグレードしたターゲット・データベースの修正の一部を完了できます。
- 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、ツールを実行するOracleユーザーの環境変数を設定します。
- アップグレード前情報ツール(preupgrade.jar)のコマンド
  アップグレード前情報ツール(`preupgrade.jar`)のコマンドを使用して、アップグレード前にシステムをチェックします。
- アップグレード前情報ツールの出力
  アップグレード前情報ツール(`preupgrade.jar`)はDIRコマンドライン・オプションで指定する出力ディレクトリに修正スクリプトとログ・ファイルを作成します。
- 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)。
- 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。

関連項目

- Oracle DatabaseアップグレードでのAutoUpgradeの使用
- AutoUpgradeを使用したアップグレード前情報ツールで以前に提供されたチェックの取得に関する項

親トピック: Oracle Databaseアップグレード・ユーティリティ

アップグレード前情報ツールについて

システムがアップグレード可能かどうかを判断するには、従来のアップグレード前情報ツール(`preupgrade.jar`)を使用できます。

正常なアップグレードを確実に行うために、分析処理モードでAutoUpgradeユーティリティを実行し、AutoUpgradeユーティ

リティの修正を使用した後、アップグレードを完了する方法をお薦めします。ただし、アップグレードを開始する前に、アップグレード前情報ツールを使用してこれらのタスクを引き続き実行できます。

従来のアップグレード前情報ツールを使用する場合は、アップグレードが完了した後に、ツールで検出される問題の修正を助けるために生成されるアップグレード後スクリプトを使用できます。最新の更新を取得するには、My Oracle Supportノート884522.1.からツールの最新バージョンをダウンロードすることをお薦めします。

ツールはオペレーティング・システムのコマンドラインから実行できます。アップグレード前情報ツールにより、アップグレード開始前に一部の問題を修正するためのアップグレード前スクリプト、およびアップグレード完了後に一部の問題を修正するためのアップグレード後スクリプトが作成されます。

アップグレード前情報ツール(`preupgrade.jar`)は次のファイルを作成します。

- ログ・ファイル`preupgrade.log`。

  ログ・ファイルには、アップグレード前情報ツールの出力が含まれます。

- PDBの場合は`preupgrade_fixups_pdbname.sql` (`pdbname`はPDBの名前)

  アップグレードを実行する前に、SQL*Plusでアップグレード前修正スクリプトを手動で実行して、アップグレード前のツールで識別された多くの問題を解決できます。

- `postupgrade_fixups_pdbname.sql` (PDBの場合、`pdbname`はPDBの名前)または`postupgrade_fixups.sql`スクリプト(非CDBデータベース)。

  データベースのアップグレードが完了した後に、このスクリプトを実行して問題を修正できます。

関連項目

- My Oracle Supportノート884522.1

親トピック: アップグレード前情報ツールおよびAutoUpgradeのアップグレード前処理

アップグレード前情報ツールによって生成されるアップグレード前スクリプト

アップグレード前情報ツールによって生成されるアップグレード前スクリプトを実行することによって、新しいOracle Databaseリリースへのアップグレード前に多くの問題を修正できます。

`preupgrade_fixups.sql`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いなくても、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。

- ログ・ファイル(preupgrade.log)。

  ログ・ファイルには、アップグレード前情報ツールのログ出力が含まれます。

- アップグレード前修正SQLスクリプト。ソース・データベースのタイプによって異なります。

  - 非CDB: preupgrade_fixups.sql
  - CDB: 2つの異なるスクリプト・セット:

    1. preupgrade_fixups.sql: すべてのPDB用の統合スクリプト。
    2. 複数のpreupgrade_fixups_pdbname.sqlスクリプト。pdbnameは、スクリプト生成の対象となるPDBの名前です。個々のスクリプトを特定のPDBで実行します。

  catcon.plを使用するか、SQL*Plusコマンドを使用することによって、スクリプトを実行します。データベースのアップグレードを開始する前に、これらのスクリプトを実行して問題を修正する必要があります。これらのスクリプトにより、アップグレード前ツールによって検出された多くの問題が解決されます。

  スクリプトによって検出された問題には、それぞれ問題の説明と、その問題を解決するために実行できる作業が含まれています。アップグレード前ツールそのものは、エラーを修正するための変更をデータベースに加えるわけではありません。しかし、生成されたスクリプトを実行して、検出されたエラーを修正できます。スクリプトは自動スクリプトが安全に修正できる問題だけを修正します。自動スクリプトが安全に修正できないアップグレード前問題には一般に、ユーザー・アプリケーションのDBAの知識が必要です。それらの問題には手動で対処できます。

親トピック: アップグレード前情報ツールおよびAutoUpgradeのアップグレード前処理

アップグレード前情報ツールによって生成されるアップグレード後スクリプト

アップグレード後に、アップグレード前情報ツールによって生成されるアップグレード後スクリプトを実行することによって、アップグレードしたターゲット・データベースの修正の一部を完了できます。

アップグレード前情報ツールによって、アップグレード後修正スクリプトが生成されます。アップグレード後にそのスクリプトを実行することによって、アップグレード後に修正できる問題を修正できます。

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いなくても、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。

スクリプトには、次の2つの異なるセットがあります。

1. `postupgrade_fixups.sql`: すべてのPDB用の統合スクリプト
2. 複数の`postupgrade_fixups_pdbname.sql`スクリプト。`pdbname`は、スクリプト生成の対象となるPDBの名前です。個々のスクリプトを特定のPDBで実行します。

自動スクリプトが安全に修正できないアップグレード後問題には一般に、ユーザー・アプリケーションのDBAの知識が必要です。それらの問題には手動で対処できます。

非CDBデータベースに対するアップグレード後修正スクリプトの実行ガイドライン

アップグレード後スクリプトを実行する際には、結果がログ・ファイルにスプールされるように設定して、出力を読めるようにすることをお薦めします。ただし、`admin`ディレクトリには結果をスプールしないでください。

親トピック: アップグレード前情報ツールおよびAutoUpgradeのアップグレード前処理

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、ツールを実行するOracleユーザーの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。マルチテナント・アーキテクチャのアップグレードでは、ツールで分析するすべてのPDBのオープンも必要です。

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。

   たとえば:

   ```
   $ 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
   ```

3. (マルチテナント・アーキテクチャのアップグレード): CDBでアップグレード前情報ツールを実行している場合は、コマンド`alter pluggable database all open`を使用してPDBをオープンし、ツールで分析ができるようにします。たとえば: データベースにログインし、次のコマンドを入力してすべてのPDBをオープンします。

   ```
   $ sqlplus ¥ as SYSDBA
   .
   .
   .
   SQL> alter pluggable database all open;
   ```

親トピック: アップグレード前情報ツールおよびAutoUpgradeのアップグレード前処理

アップグレード前情報ツール(preupgrade.jar)のコマンド

アップグレード前情報ツール(`preupgrade.jar`)のコマンドを使用して、アップグレード前にシステムをチェックします。

前提条件

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。

ファイル・パス

`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。たとえば:

`/tmp/preupgrade.jar`

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]
```

オプション

コマンド・オプション	説明
`FILE|TERMINAL`	スクリプトの出力場所です。スクリプトの出力ファイルを指定するには、`FILE` を使用します。出力先端末を指定するには、`TERMINAL` を使用します。値を指定しない場合、デフォルトは `FILE` になります。`TERMINAL` を指定した場合、画面出力はディスプレイに表示され、スクリプトとログは出力ディレクトリのパスに置かれます。
`TEXT|XML`	出力タイプです。XML 出力を指定するには `XML` を使用します。出力タイプを指定しない場合、デフォルトは `TEXT` になります。
`DIR output_dir`	出力は指定したディレクトリに送られます。`DIR` オプションを使用しないで出力ディレクトリを指定した場合は、出力は次のデフォルトの場所のいずれかに置かれます。 ● `DIR` を使用して出力ディレクトリを指定していなくても、`ORACLE_BASE` 環境変数を定義した場合は、生成されるスクリプトとログ・ファイルが作成されます。 ● 出力ディレクトリを指定せず、`ORACLE_BASE` 環境変数も定義しなかった場合は、生成されるスクリプトとログ・ファイ

コマンド・オプション	説明
	ルは次のファイル・パスに作成されます。 `ORACLE_HOME/cfgtoollogs/dbunique_name/preupgrade`
`-c "pdb1 pdb2 pdb3"` (Windows) `-c 'pdb1 pdb2 pdb3'` (Linux および UNIX)	処理に含める CDB 内のコンテナのリスト(許可リスト)を指定します。処理する PDB のスペース区切りリストを指定します。リストを指定するには、Linux および UNIX オペレーティング・システムでは単一引用符を、Windows システムでは二重引用符を使用します。 `-c` も`-C` も指定しない場合、CDB 内のすべての PDB が処理されます。
`-C "pdb1 pdb2 pdb3"` (Windows) `-C 'pdb1 pdb2 pdb3'` (Linux および UNIX)	処理から除外する CDB 内のコンテナのリスト(ブロック・リスト)を指定します。処理から除外する PDB のスペース区切りリストを指定します。リストを指定するには、Linux および UNIX オペレーティング・システムでは単一引用符を、Windows システムでは二重引用符を使用します。 `-c` も`-C` も指定しない場合、CDB 内のすべての PDB が処理されます。
`-loadonly`	他のどの操作も実行せずに、`READ WRITE` モードのデータベースに `DBMS_PREUP` パッケージをロードします。 このパラメータを使用すると、アップグレード前情報ツールを実行したときに `DBMS_PREUP` パッケージがロードされて DB (DB またはコンテナ)が `READ-ONLY` モードになるように、特定の非 CDB または CDB データベースを準備できます。ツールを使用してデータベースを読取り専用モードで分析する場合は、データベースを `READ-ONLY` モードに設定する前に、このコマンドを使用して、`READ WRITE` モードのデータベースに `DBMS_PREUP` パッケージをロードする必要があります。 このパラメータは、読取り専用のスタンバイ・データベースでも使用できます。この場合、パッケージをプライマリ・データベースにロードし、スタンバイでパッケージを実行します。
`-p password`	ユーザーのパスワードを指定します。 データベースに接続するのにオペレーティング・システムの認証を使用しない場合は、`-p` オプションを使用してコマンドラインでパスワードを

コマンド・オプション	説明
	指定します。-u オプションを使用して、コマンドラインでユーザー名が指定されているが、-p でパスワードが指定されていない場合は、ツールのプロンプトでパスワード入力が求められます。
-u username	チェックするデータベースに SYSDBA として接続するために使用するユーザーのユーザー名を指定します。このオプションは、オペレーティング・システムの認証を使用してデータベースに接続しない場合にのみ、使用します。 たとえば、チェックするデータベースの OSDBA グループのメンバーではないユーザーとしてログインします。その場合、そのユーザー・アカウントには SYSDBA システム権限に代わるオペレーティング・システムの認証権限がありません。SYSDBA システム権限を持つユーザーとしてログインするには、-u および -p オプションを使用してデータ・ディクショナリ認証を指定します。
-oh oracle_home	チェックする Oracle ホームを指定します。チェックする Oracle ホームのパスを指定します。 チェックする Oracle ホームのパスを指定しない場合は、アップグレード前情報ツールにより Oracle ホームのユーザー環境変数で指定されたパスがデフォルトとして使用されます。この変数は、Linux および Unix システムでは$ORACLE_HOME、Windows システムでは%ORACLE_HOME%です。
-sid system_identifier	チェックする Oracle システムの ID を指定します。チェックするデータベースの ORACLE_SID を指定します。
-help	コマンドライン構文のヘルプ・テキストを表示します。

例B-1 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 Server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。たとえば:

```
$ORACLE_HOME/jdk/bin/java -jar
/u01/app/oracle/product/19.0.0/rdbms/admin/preupgrade.jar TERMINAL TEXT
```

例B-2 ソースOracleホームのCDB

1. すべてのプラガブル・データベースをオープンします。

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

2. 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
```

3. -cオプションを使用し、包含リストを指定してアップグレード前情報ツールを実行します。この例で包含リストはPDB1とPDB2で、コマンドはLinuxまたはUNIXシステムで実行されます。コマンドの出力は端末に表示され、出力はテキストで表示されます。

```
$ORACLE_HOME/jdk/bin/java -jar
/u01/app/oracle/product/19.0.0/dbhome_1/rdbms/admin/preupgrade.jar TERMINAL
TEXT -c 'pdb1 pdb2'
```

親トピック: アップグレード前情報ツールおよびAutoUpgradeのアップグレード前処理

アップグレード前情報ツールの出力

アップグレード前情報ツール(preupgrade.jar)はDIRコマンドライン・オプションで指定する出力ディレクトリに修正スクリプトとログ・ファイルを作成します。

アップグレード前情報ツールを実行するときに、出力ディレクトリとして指定したディレクトリ内に次のファイルが生成されます。

ログ・ファイル(preupgrade.log)

ファイルpreupgrade.logは、FILEオプションを指定してコマンドを実行するたびにアップグレード前情報ツールにより生成されるレポートです。ログ・ファイルにはアップグレードについてのツールからのすべての推奨事項要件が含まれています。ログ・ファイルは次のパスにあります: $ORACLE_BASE/cfgtoollogs/dbua/upgrade/SID/。TERMINALオプションを指定してコマンドを実行すると、このファイルの内容がディスプレイに出力されます。ログ・ファイルの例は、「アップグレード前情報ツールの出力例」の項を参照してください。

XML DBUAファイル(upgrade.xml)

アップグレード前情報ツールのコマンドラインでXMLファイル出力を指定した場合は、preupgrade.logのかわりにupgrade.xmlファイルが生成されます。

アップグレード修正ファイル(preupgrade_fixups.sql)およびアップグレード後修正ファイル(postupgrade_fixups.sql)

アップグレード前情報ツールがアップグレードをブロックまたは妨げる問題を特定します。

使用しているアプリケーションの具体的な目標を自動スクリプトが理解することはできないため、一部の問題はDBAが解決する必要があります。ただし、他の問題は解決が難しいということはありません。そのような場合、アップグレード前情報ツールは問題を解決するために必要なSQL文が含まれたスクリプトを自動的に生成します。これらのスクリプトを使用することで、考えられるアップグレードの問題を解決するためにDBAが実行する必要がある作業を実行、追加、簡素化できます。アップグレード前に問題を解決するSQL文はpreupgrade_fixups.sqlスクリプトに配置されます。アップグレード後に問題を解決するSQL文はpostupgrade_fixups.sqlスクリプトに配置されます。マルチテナント・アーキテクチャのOracle Databaseでアップグレード前情報ツールを実行するときには、すべてのコンテナにわたって統合スクリプトのpreupgrade_fixups.sqlスクリプトとpostupgrade_fixups.sqlを実行できます。catcon.plを使用して統合スクリプトを実行します。

これらの修正ファイルはどちらもアップグレード前情報ツールのDIRコマンドライン・オプションで指定した出力ディレクトリに生成さ

れます。

スクリプトは次のステップを実行してアップグレード前問題やアップグレード後問題を解決します。

1. アップグレード前情報ツールが特定したそれぞれの問題についてアップグレード前情報ツール・チェックを再実行して、問題がまだ残っているかどうかを判定します。
2. 問題がまだ残っている場合、問題の修正ルーチンがあるかどうかに応じて、ツールは次のいずれかを実行します。
   - その問題についてOracleが提供している修正ルーチンがない場合、スクリプトがその修正ルーチンを実行します。その後にアップグレード前情報ツール・チェックが再実行され、その問題が解決されたことを確認します。問題が解決している場合、問題が解決されたというメッセージが表示されます。
   - Oracleにより提供された修正ルーチンがない場合、問題がまだ存在しているというメッセージが表示されます。

チェックが完了したら、スクリプトが指摘した問題がまだ存在しているか、確認します。アップグレードする前に、「**必須**」マークが付いたすべての問題を解決します。推奨される修正や修正情報を解決のために選択できます。`preupgrade_fixups.sql`スクリプトと`postupgrade_fixups.sql`スクリプトは何度でも必要な回数だけ実行できます。問題解決計画の一部として、スクリプトを進捗レポートして使用して、残っている問題を追跡できます。

アップグレード前修正およびアップグレード後修正スクリプトにより特定された問題を解決した後で、アップグレードを行う前にもう一度、アップグレード前情報ツール(`preupgrade.jar`)を再実行することをお薦めします。もう一度ツールを実行すると、解決しようとしたすべての問題が解決されたことを確認できるようになります。

マルチテナント環境のシステム(CDBとPDB)をチェックしている場合、`preupgrade_fixups.sql`と`postupgrade_fixups.sql`スクリプトには個別のすべてのPDBの修正コードのロールアップが含まれます。それらは、現在のPDBのコードだけが実行されるようにコード化されます。各PDBで同じスクリプトを実行できます。スクリプトはそのPDBの問題だけを修正します。その結果、`catcon.pl`で`preupgrade_fixups.sql`と`postupgrade_fixups.sql`を使用してCDB全体で修正を実行することが容易です。

関連項目

- アップグレード前情報ツールの出力例
- Oracle Databaseのアップグレード用スクリプト

親トピック: アップグレード前情報ツールおよびAutoUpgradeのアップグレード前処理

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.

                                                        Min Size
```

```
      Tablespace                        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$ROOT
  which are identified above as AFTER UPGRADE "(AUTOFIXUP)" can be resolved by
  executing the following from within the container
    SQL>@/u01/app/oracle/cfgtoollogs/CDB2/preupgrade/postupgrade_fixups.sql
Report generated by Oracle Database Pre-Upgrade Information Tool Version
18.0.0.0.0 on 2018-06-13T15:08:58
Upgrade-To version: 18.0.0.0.0
======================================
Status of the database prior to upgrade
======================================
      Database Name:  CDB2
     Container Name:  PDB$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.

                                                   Min Size
      Tablespace                        Size     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: アップグレード前情報ツールおよびAutoUpgradeのアップグレード前処理

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。

詳細は、インストールされている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。

- 以前のOracle Databaseリリースからのパスワードが指定されたデータベース・リンクについて
  この情報は、アップグレードの実行後に元のデータベース・リリースにダウングレードする場合にのみ重要となります。
- TIMESTAMP WITH TIME ZONEデータ型に対するOracle Databaseの警告について
  Oracle Databaseのアップグレードには、既存のTIMESTAMP WITH TIME ZONEデータ型に影響する可能性のある、更新されたタイムゾーン・データ型が含まれます。

関連項目

- 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: アップグレード前情報ツールおよびAutoUpgradeのアップグレード前処理

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。

参照:

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。たとえば:

```
SELECT * FROM DBA_DEPENDENCIES
WHERE referenced_name IN
('UTL_TCP','UTL_SMTP','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。

関連項目

- 『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

以前のOracle Databaseリリースからのパスワードが指定されたデータベース・リンクについて

この情報は、アップグレードの実行後に元のデータベース・リリースにダウングレードする場合にのみ重要となります。

新しいOracle Databaseリリースへのアップグレード中に、データベース・リンク内のすべてのパスワードは暗号化されます。

- アップグレード元のリリースにダウングレードするには、ダウングレードする前に、暗号化されたパスワードが指定されたすべてのデータベース・リンクを削除する必要があります。したがって、ダウングレードしたデータベースにはデータベース・リンクは存在しません。
- 元のリリースにダウングレードできるようにしておく必要がある場合は、影響を受けるデータベース・リンクの情報を`SYS.LINK$`表から保存しておくことで、ダウングレード後にデータベース・リンクを再作成できます。

- 以前のリリースの詳細は、アップグレードする前のOracle Databaseリリース用の元のドキュメントを参照してください。使用しているプラットフォーム別の以前のリリースの『*Oracle Database*インストレーション・ガイド』も参照してください。

参照:

認証およびデータベース・リンクの詳細は、『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

TIMESTAMP WITH TIME ZONEデータ型に対するOracle Databaseの警告について

Oracle Databaseのアップグレードには、既存の`TIMESTAMP WITH TIME ZONE`データ型に影響する可能性のある、更新されたタイムゾーン・データ型が含まれます。

Oracle Databaseアップグレードに付属のタイムゾーン・ファイルは、一部のタイムゾーン・リージョンの法律および移行ルールに対して行われた変更を反映するために更新されています。これらの変更は、既存の`TIMESTAMP WITH TIME ZONE`データ型に影響する場合があります。

データベースをアップグレードする前に、最新のタイムゾーン・ファイルがあることを確認することをお薦めします。アップグレード対象のデータベースのタイムゾーン・ファイルのバージョンが、Oracle Databaseの新しいリリースで使用可能なタイムゾーン・ファイルの最新バージョンでない場合は、アップグレード前情報ツールによって、警告が表示されて続行方法が説明されます。

注意:

タイムゾーン・ファイルのバージョンが一致していない場合、データベースに格納されている `TIMESTAMP WITH TIME ZONE` 型のデータは、アップグレード中に破損する可能性があります。

リリース更新(RU)に含まれるすべてのタイム・ゾーン・ファイル(DST)

Oracle Database 19c RU 19.18.0以降では、使用可能なすべてのDSTパッチがRUに付属しており、`Oracle_home/oracore/zoneinfo`ディレクトリにデプロイされます。DSTパッチをインストールしても、データベース操作には影響しません。ただし、RUとともにパッチをインストールすると、データベースのタイムゾーン・バージョンを簡単に調整できます(そのようにする必要がある場合)。たとえば、トランスポータブル表領域またはフル・トランスポータブル・エクスポート/インポートを使用している場合は、ソース・データベースとターゲット・データベースで同じ文字セットとタイム・ゾーン設定が使用されていることを確認する必要があります。この変更により、宛先データベースをデフォルトとは異なるタイムゾーン・ファイル・バージョンを使用するように簡単に変更できます。

デフォルトでは、AutoUpgradeはデータベースのタイム・ゾーンを最新の使用可能なレベルに変更します。タイム・ゾーンをアップグレードしない場合は、AutoUpgrade構成ファイル内のローカル・パラメータ`timezone_upg`を明示的にnoに設定する必要があります。たとえば:

```
upg1.timezone_upg=no
```

古いデータベース・タイムゾーン・ファイルを使用する場合は、データベース環境変数`ORA_TZFILE`を古いタイムゾーン・ファイルに設定します。タイムゾーン・ファイルは`Oracle_home/oracore/zoneinfo`にあります。環境変数`ORA_TZFILE`を使用すると、最も番号の大きいタイムゾーン・ファイルを使用するデフォルトをオーバーライドできます。

タイムゾーン・ファイル・バージョンの手動更新

次の表は、アップグレード中のデータベースのタイムゾーン・ファイルのバージョンが、インストール済のOracle Databaseリリースで指定されているバージョンと異なる場合の、アップグレード前情報ツールの操作方法を示しています。

表B-2 タイムゾーン・ファイルのバージョンの修正を行う場合の選択肢

アップグレード中のデータベースのタイムゾーン・バージョン	タイムゾーン・ファイルを修正するとき
新しいデータベース・リリースに含まれている最新バージョンより前のバージョンの場合、アップグレード前情報ツールには「Database is using a time zone file older than version n」と表示されます。	データベースのアップグレードの完了後。 DBMS_DST PL/SQL パッケージを使用して、タイムゾーン・ファイルおよび TIMESTAMP WITH TIME ZONE データのアップグレード・ステップに関する説明に従います。次のドキュメントを参照してください。 Oracle Database グローバリゼーション・サポート・ガイド
新しいデータベース・リリースに含まれているバージョンより新しいバージョンの場合、アップグレード前情報ツールには、「Database is using a time zone file greater than version n.」と表示されます。	データベースのアップグレードの開始前。 使用しているタイムゾーン・ファイルのバージョンに適した RDBMS DST パッチを使用して、Oracle ホームにパッチを適用する必要があります。アップグレードする各データベースにパッチを適用します。適用しない場合、データベースをアップグレードすることなくアップグレード・スクリプトが終了します。RDBMS DST パッチは My Oracle Support ノート ID 412160.1 から入手できます。

関連項目

- 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
- DBMS_DST (ドキュメントID 1509653.1)を使用した12cリリース1 (12.1.0.1)以上のRDBMS DSTバージョンの更新
- Primary Note DST FAQ : Updated DST Transitions and New Time Zones in Oracle RDBMS and OJVM Time Zone File Patches (ドキュメントID 412160.1)
- Oracle Databaseグローバリゼーション・サポート・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

C Oracle Database Upgrade Assistant (DBUA)を使用したアップグレード

Database Upgrade Assistant (DBUA)では、Oracle Databaseのアップグレードを順を追って説明するためのグラフィカル・ユーザー・インタフェースが提供されています。DBUAは、CDBおよび非CDBのデータベース・システム用に動作します。

> ノート:
>
> ユーザー・インタフェースが表示されないサイレント・モードで DBUA を起動できます。サイレント・モードは、大規模なロールアウトやスクリプトで役立つことがあります。

- DBUAの使用要件
  マルチテナント・アーキテクチャおよび非CDB Oracle DatabaseのアップグレードでDatabase Upgrade Assistant (DBUA)を使用するには、このトピックを参照してください。
- アップグレード時のDBUAの停止について
  DBUAを停止する場合、アップグレードを手動で完了する必要があります。
- DBUAによるOracle Databaseのアップグレードの処理方法
  DBUAは、データベース・ソフトウェアのインストールの一環として起動することも、ソフトウェアのインストール後に手動で起動することもできます。
- DBUAによって起動されるアップグレード・スクリプト
  アップグレード中に、DBUAによって、アップグレードを自動化して停止時間を最小化する適切なアップグレード・スクリプトが自動的に実行されます。
- 既存のOracleホームからのデータベースの移動
  この手順を使用して、Oracle Databaseを別のOracle Database 18c以上のホームに移行します。
- 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。
- サイレント・モードのDBUAを使用したOracle Databaseのアップグレード
  DBUAを`-silent`コマンドライン・オプションとともに使用すると、非対話型(「サイレント」)アップグレードを実行できます。

関連項目

- サイレント・モードのDBUAを使用したOracle Databaseのアップグレード

DBUAを使用するための要件

マルチテナント・アーキテクチャおよび非CDB Oracle DatabaseのアップグレードでDatabase Upgrade Assistant (DBUA)を使用するには、このトピックを参照してください。

Oracle Database Vaultが有効になっている場合は、「Oracle Database Vaultを使用しているOracle Databaseのアップグレードの要件」のトピックを参照してください。

DBUAでアップグレード前チェックを実行できるように、最新のAutoUpgradeユーティリティをダウンロードし、インストールする必要があります。

ノート:

Oracle Database 19c は、非 CDB アーキテクチャの最終リリースです。

ノート:

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。

DBUAを使用して、マルチテナント・アーキテクチャ・コンテナ・データベース(CDB)、プラガブル・データベース(PDB)および非CDBデータベースをアップグレードできます。手順は同じですが、アップグレードのタイプに応じて、選択する必要のある内容とDBUAの動作は異なります。

- My Oracle Supportドキュメント884522.1から最新の`preupgrade.jar`バージョンをダウンロードします。`Oracle_home/rdbms/admin`の下のOracle Database 19c Oracleホームにファイルをダウンロードし、その場所にファイルを解凍します。次に、このガイドの「アップグレード前情報ツール(`preupgrade.jar`)コマンド」の手順に従います
- 保証付きリストア・ポイントを使用するには、アップグレード中にデータベースのARCHIVE LOGモードおよびFLASHBACKモードがオンであることを確認します。オンであることを確認するには、次のSQLコマンドを入力します。

```
SQL> select log_mode,flashback_on from v$database;
```

- データベース・インスタンスが実行されていない場合、DBUAはインスタンスを起動しようとします。インスタンスが起動され、実行されている場合、DBUAはそのインスタンスに接続します。
- DBUAを使用せずに手動でデータベースをリストアする場合は、DBUAを起動する前に`ORACLE_HOME/cfgtoollogs/dbua/logs/`ディレクトリにある`Welcome_SID.txt`ファイルを削除します。このファイルが検出されると、DBUAは再実行操作で起動します。
- リストア・スクリプトでは、一般にデータベース(非CDBのシングル・インスタンス、高可用性、またはOracle RAC)を以前のリリースおよび以前のOracleホームの場所にリストアできます。ただし、データベースをOracle Internet Directory (OID)に登録した場合は、リストア・スクリプトでOracle Internet Directoryを登録解除できません。権限を持つユーザーとしてログインし、新しいリリースのデータベースを手動で登録解除する必要があります。

関連項目

- My Oracle Supportノート884522.1
- アップグレード前情報ツール(preupgrade.jar)のコマンド
- My Oracle Supportノート2485457.1
- Oracle Label SecurityおよびOracle Database Vaultを使用しているデータベースのアップグレードの要件

親トピック: Oracle Database Upgrade Assistant (DBUA)を使用したアップグレード

アップグレード時のDBUAの停止について

DBUAを停止する場合、アップグレードを手動で完了する必要があります。

アップグレードを停止するが、データベースをリストアしない場合、DBUAを使用してアップグレードを継続することはできません。かわりに、手動(コマンドライン)のアップグレード手順を使用してアップグレードを継続する必要があります。データベースをリストアしないかぎり、元のOracle Databaseのサーバーに戻ることはできません。

関連項目

- マルチテナント・コンテナOracle Database (CDB)の手動でのアップグレード

親トピック: Oracle Database Upgrade Assistant (DBUA)を使用したアップグレード

DBUAによるOracle Databaseのアップグレードの処理方法

DBUAは、データベース・ソフトウェアのインストールの一環として起動することも、ソフトウェアのインストール後に手動で起動することもできます。

新しいOracle Databaseソフトウェアをインストールし、既存のOracle Databaseのアップグレードを指定した場合は、DBUAが自動的に起動されます。インストールの完了後、単独でDBUAを起動することもできます。

アップグレードの実行中、DBUAによって、各コンポーネントのアップグレード処理の進行状況が表示されます。DBUAは、詳細なトレース・ログおよびログ・ファイルを書き込み、後で参照するための完全なHTMLレポートを作成します。セキュリティを強化するために、DBUAは、アップグレード後のデータベースで、新しいユーザー・アカウントを自動的にロックします。その後、DBUAは、新しいOracleホームへの新しい構成ファイル(パラメータ・ファイルおよびリスナー・ファイル)の作成へと進みます。

DBUAは、アップグレード前のすべてのステップが完了するまで、アップグレード処理を開始しません。

関連項目

- Oracle Databaseのアップグレードの再実行
- Oracle Databaseのアップグレードを準備する作業

親トピック: Oracle Database Upgrade Assistant (DBUA)を使用したアップグレード

DBUAによって起動されるアップグレード・スクリプト

アップグレード中に、DBUAによって、アップグレードを自動化して停止時間を最小化する適切なアップグレード・スクリプトが自動的に実行されます。

前提条件フェーズでは、DBUAは、アップグレード前パラメータ(`autoupgrade.jar -preupgrade -mode`)を使用してAutoUpgradeを実行します。これにより、データベース・チェックとアップグレード前修正が実行され、アップグレードを開始する前にほとんどの問題が修正されます。

DBUAは、次のロジックを使用して、新しい必須表領域を変更または作成します。

- データファイルが自動拡張可能で、十分なディスク領域がある場合、DBUAはアップグレードを続行します。
- データファイルを自動拡張できない場合、ファイルの自動拡張を可能にするようにDBUAから求められます。
- 表領域が自動拡張可能で、`MAXSIZE`初期化パラメータに調整が必要な場合、DBUAはこの調整をユーザーに求め、`MAXSIZE`パラメータを調整します。
- 拡張できるディスク容量がない場合、DBUAは、別のデータファイルを追加して領域を作成するようユーザーに要求します。DBUAはファイルの作成場所を決定できないため、DBUAによる新しいデータファイルの追加は自動的に行われま

せん。

DBUAは前提条件フェーズで検出された多くの問題に対処します。たとえば、DBUAは正しいタイムゾーン・ファイルが使用されていて、ネットワーク・アクセス制御リストのACLの調整を行うことを確認できます。

アップグレード・フェーズ中に、DBUAは、アップグレード処理をシリアルではなくパラレルに実行する`catctl.pl`を実行します。パラレル実行によって、CPUリソースの使用率が最適化され、アップグレードの高速化と停止時間の最小化が実現します。

関連項目

- preupgrade

親トピック: Oracle Database Upgrade Assistant (DBUA)を使用したアップグレード

既存のOracleホームからのデータベースの移動

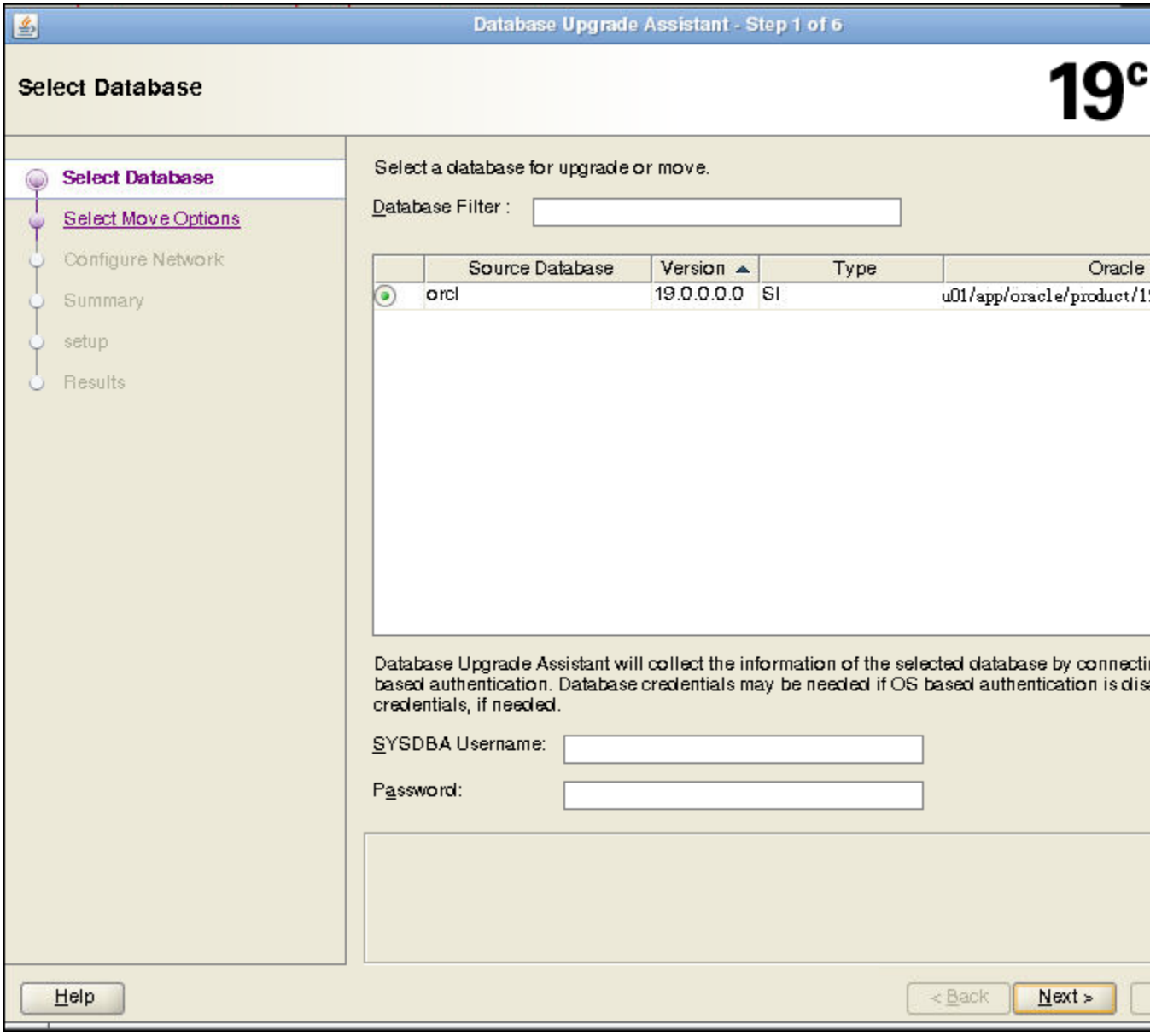
この手順を使用して、Oracle Databaseを別のOracle Database 18c以上のホームに移行します。

Oracle Database 18c以上のリリースでは、Database Upgrade Assistant (DBUA)を使用して、既存のOracleホームから別のOracleホームにOracle Databaseデータベースを移行できます。

1. DBUAを起動します。

   DBUAで「データベースの選択」ウィンドウが開きます。

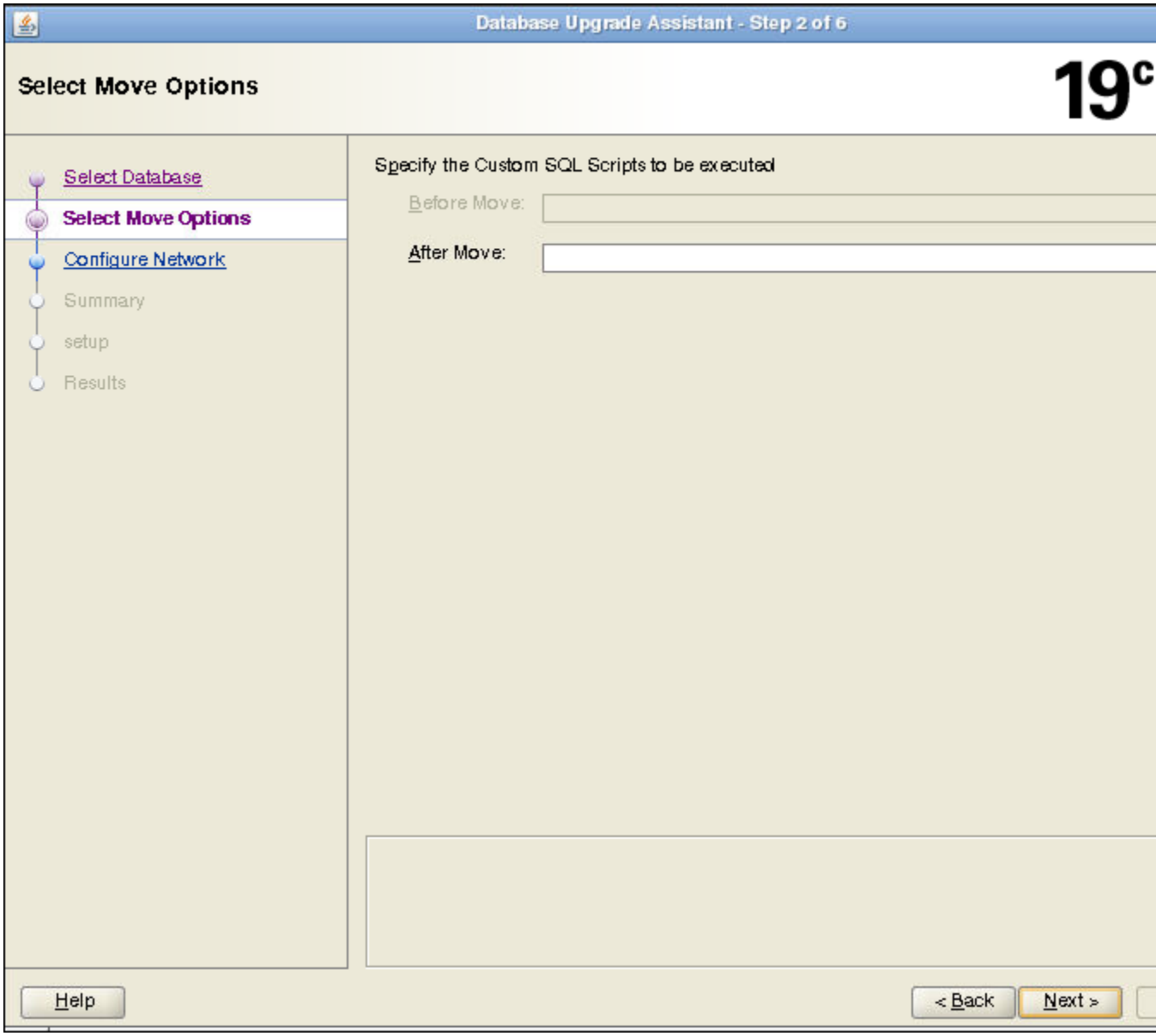
   サーバー上のすべてのデータベースがリストされます。DBUAは、データベースのリリースと場所に応じて、データベースごとに実行できる操作のタイプ(アップグレード、移動、インプレース)を示します。



新しいOracleホームに移動するデータベースを選択します。データベースに対してオペレーティング・システムの認証を有効にしなかった場合は、選択したデータベースのSYSDBAユーザー名とパスワードを指定します。

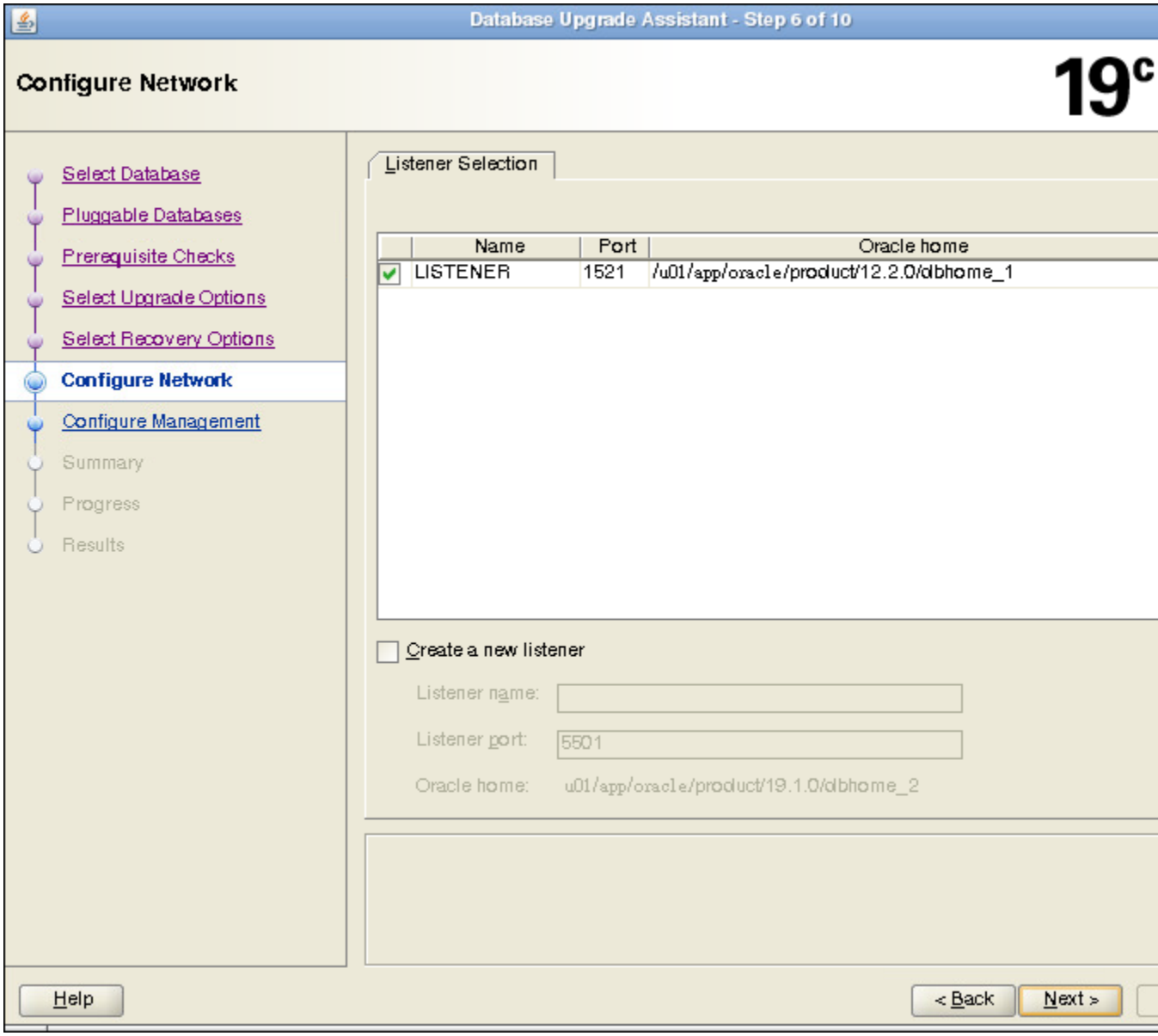
「次へ」をクリックします。「データベース・オプションの移動」ウィンドウが表示されます。

2. 「移動オプションの選択」ウィンドウで、データベースの移動後に実行するカスタムSQLスクリプトと、ファイルを置く場所を指定します。



「次へ」をクリックします。「ネットワークの構成」ウィンドウが表示されます。

3. 単一インスタンス・システムでは、既存のリスナーを選択するか、または新しいリスナーを作成できます。新しいリスナーを作成する場合、リスナー名と、そのリスナーのポート番号を指定する必要があります。



「次へ」をクリックします。「データベースの移動サマリー」ウィンドウが表示されます。

4. 移動操作のサマリーを確認して、「次へ」をクリックします。
5. 「設定」ウィンドウが表示され、データベースの移動に伴ってDBUAプロセスが示されます。
6. 移動操作が完了したら、「終了」をクリックします。
7. 「結果」ウィンドウが表示されます。ソースおよびターゲットのデータベース情報を表示したり、移動ステップを確認したり、ログ・ファイルを確認できます。確認が終了したら、「閉じる」をクリックします。

親トピック: Oracle Database Upgrade Assistant (DBUA)を使用したアップグレード

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

DBUAのグラフィカル・ユーザー・インタ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。

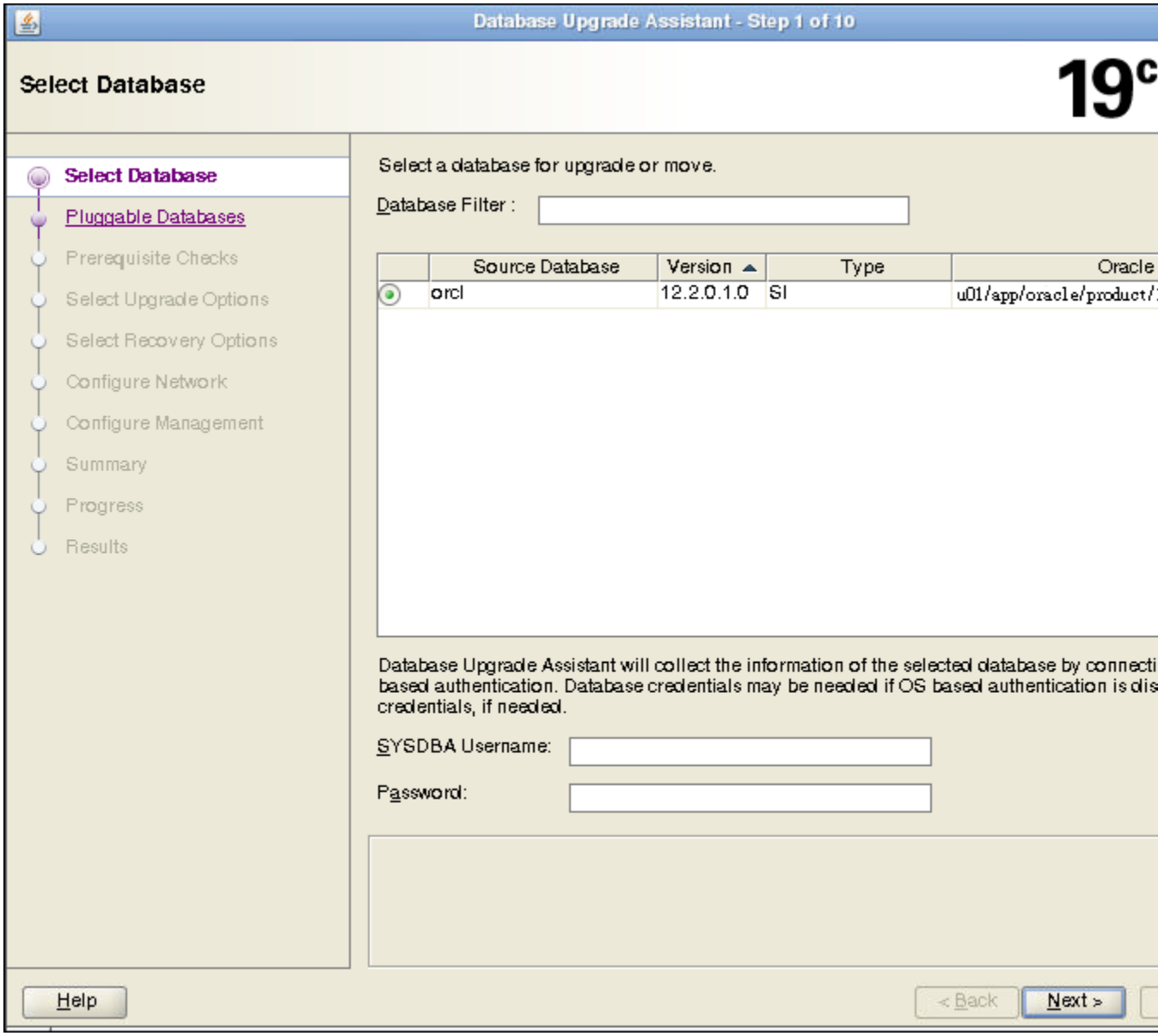
1. 新しいデータベース・ソフトウェアがインストールされたOracleホームからOracle Database Upgrade Assistant (DBUA)を起動します。dbua実行可能ファイルはディレクトリ・パスORACLE_HOME/binにあります。

   - 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。

ノート:

- 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
  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
    sid Oracle_SID、-oracleHome Oracle_home および sysDBAPassword pass
    スをアップグレードするように明示的に指示できます。たとえば:

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

- 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および password はユーザー名のパ

  ```
  dbua -sid mydb –oracleHome /u01/app/oracle/18.1.0/dbhome1 -sysDBAUserNam
  ```

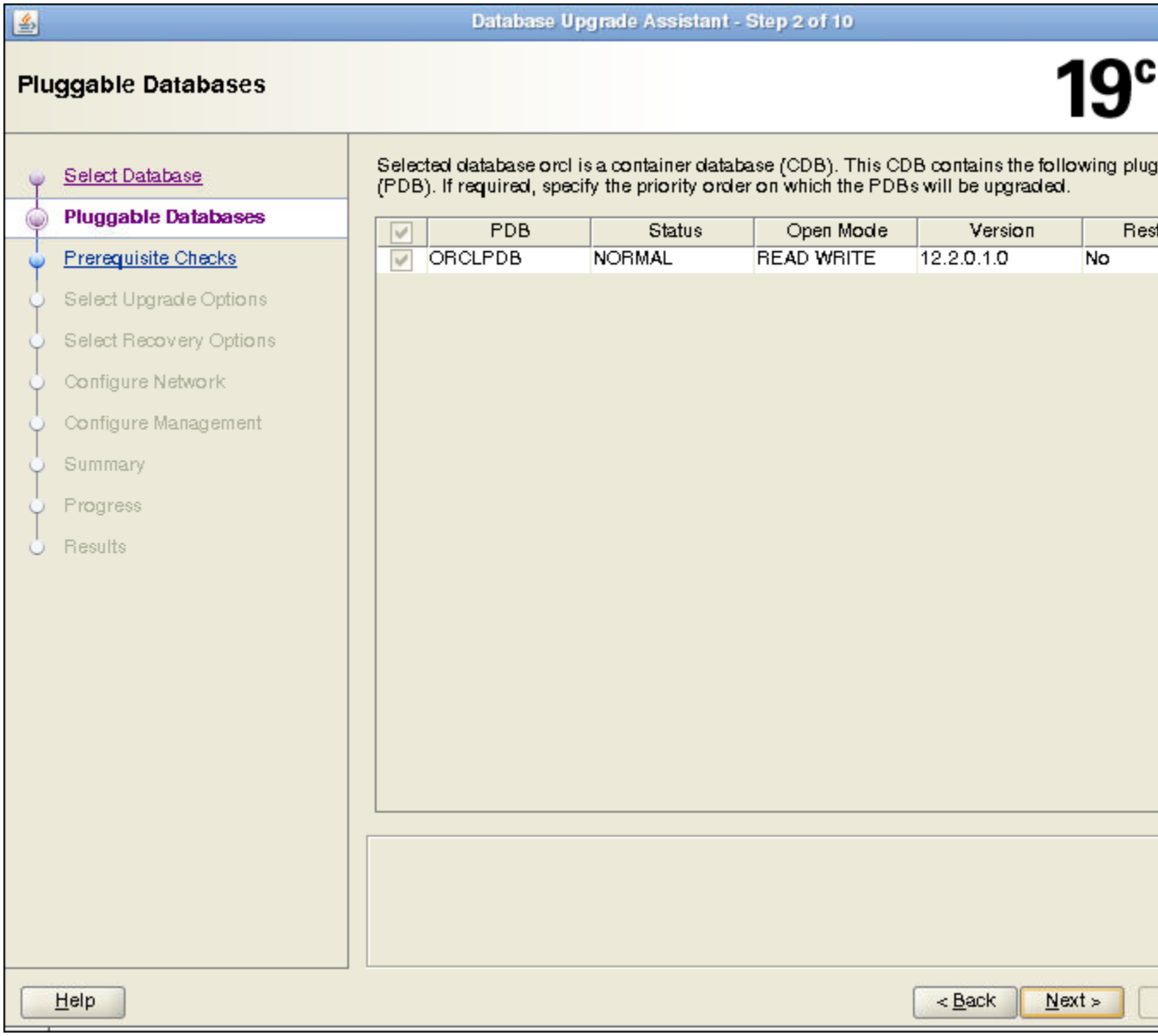
3. Oracle Real Application Clusters (Oracle RAC)のアップグレードで、データベースがリストに表示されない場合、次のcrsctlコマンドを入力してOracle RACインスタンスを確認してください。

   ```
   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 LocalSystem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・アクセス権限を制限することで、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「プラガブル・データベース」ウィンドウが表示されます。「プラガブル・データベース」ウィンドウにはCDBに含まれているプラガブル・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。

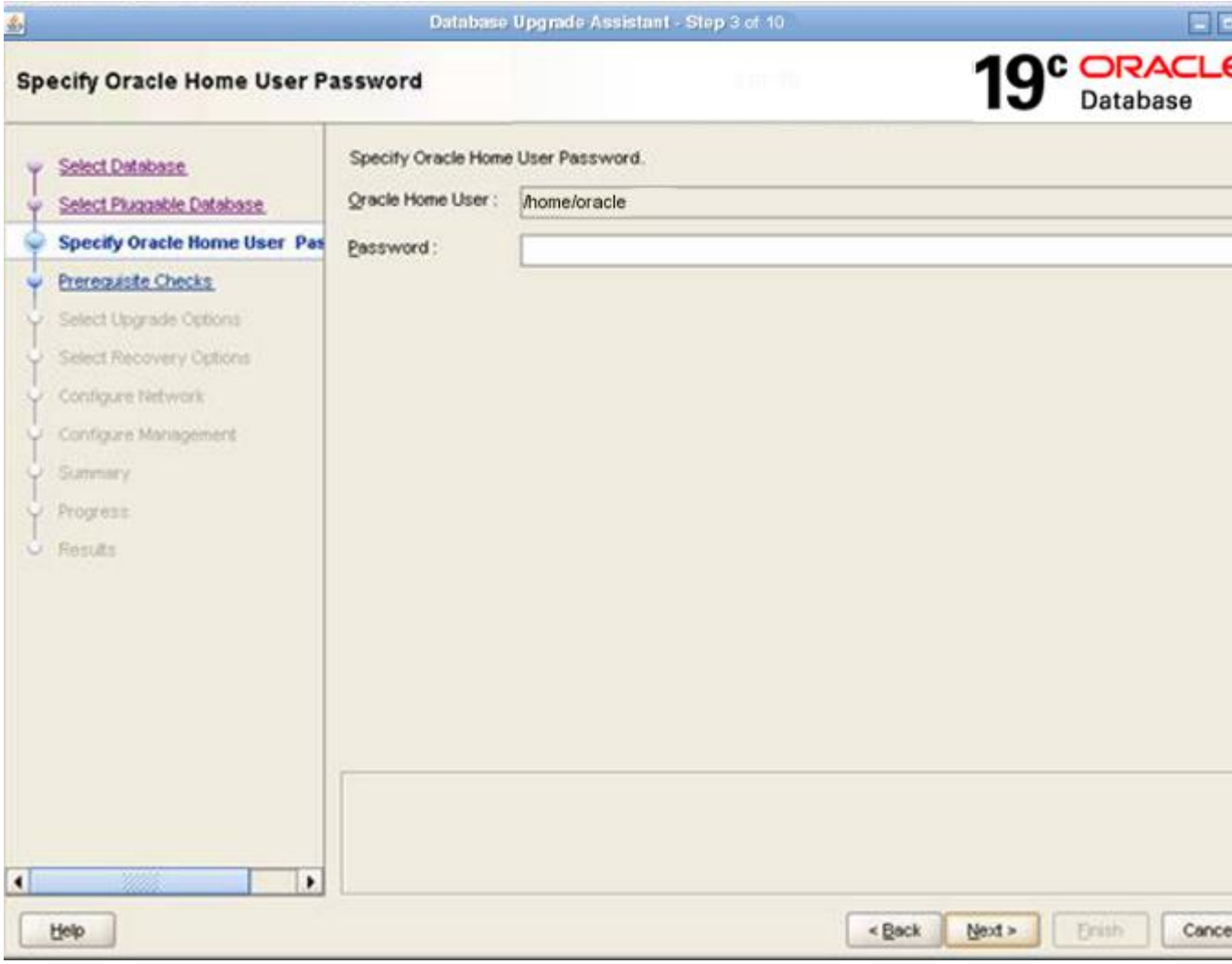


PDBのアップグレードの優先度を選択できます。各PDBの優先度列をクリックし、アップグレード優先度の数値を入力します(1が最初のアップグレード、2が2番目のアップグレードというように続きます)。

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。

6. 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。



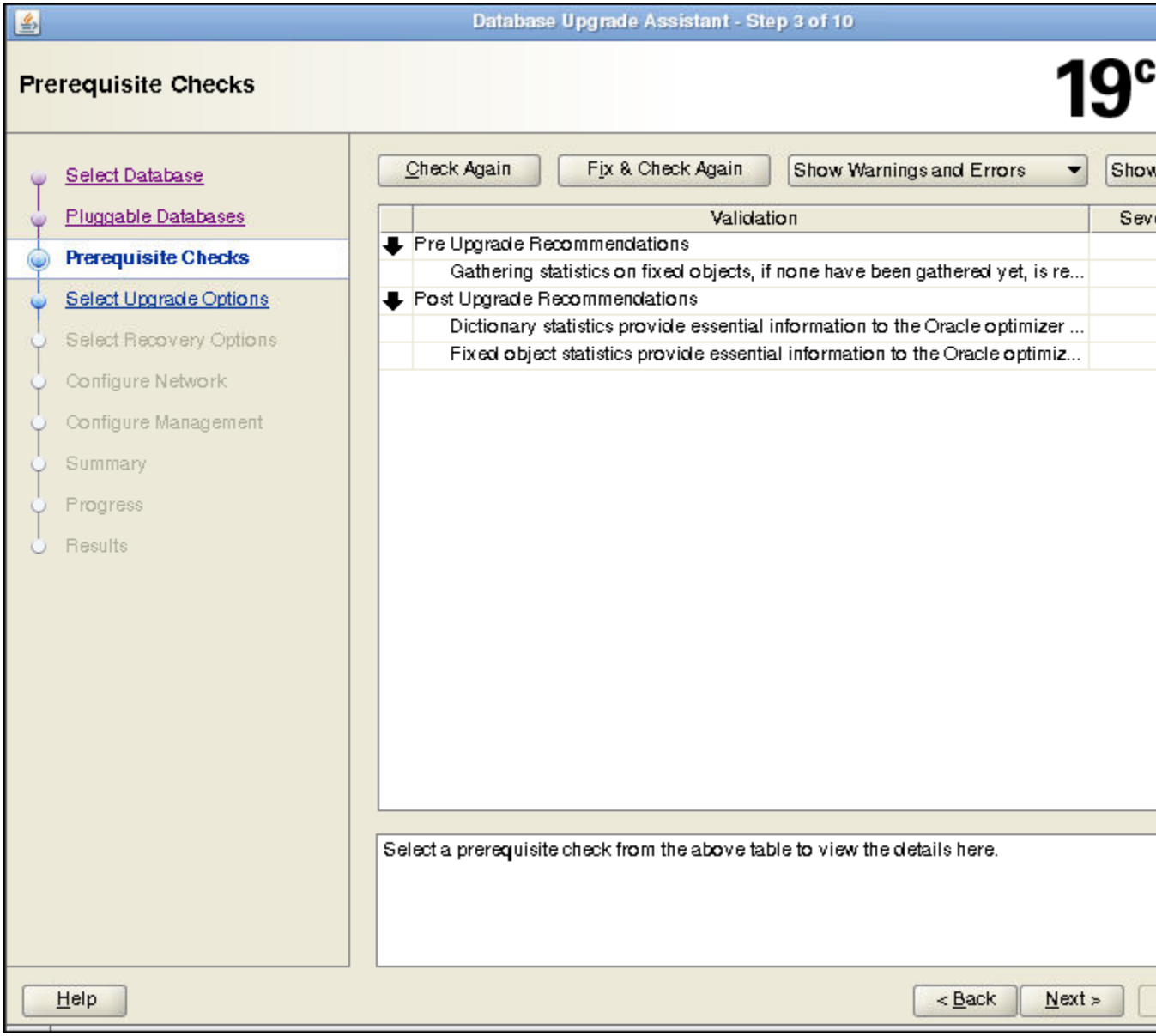
Oracleホームのユーザー名と、そのユーザー・アカウントのパスワードを指定して、「次へ」をクリックします。

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。

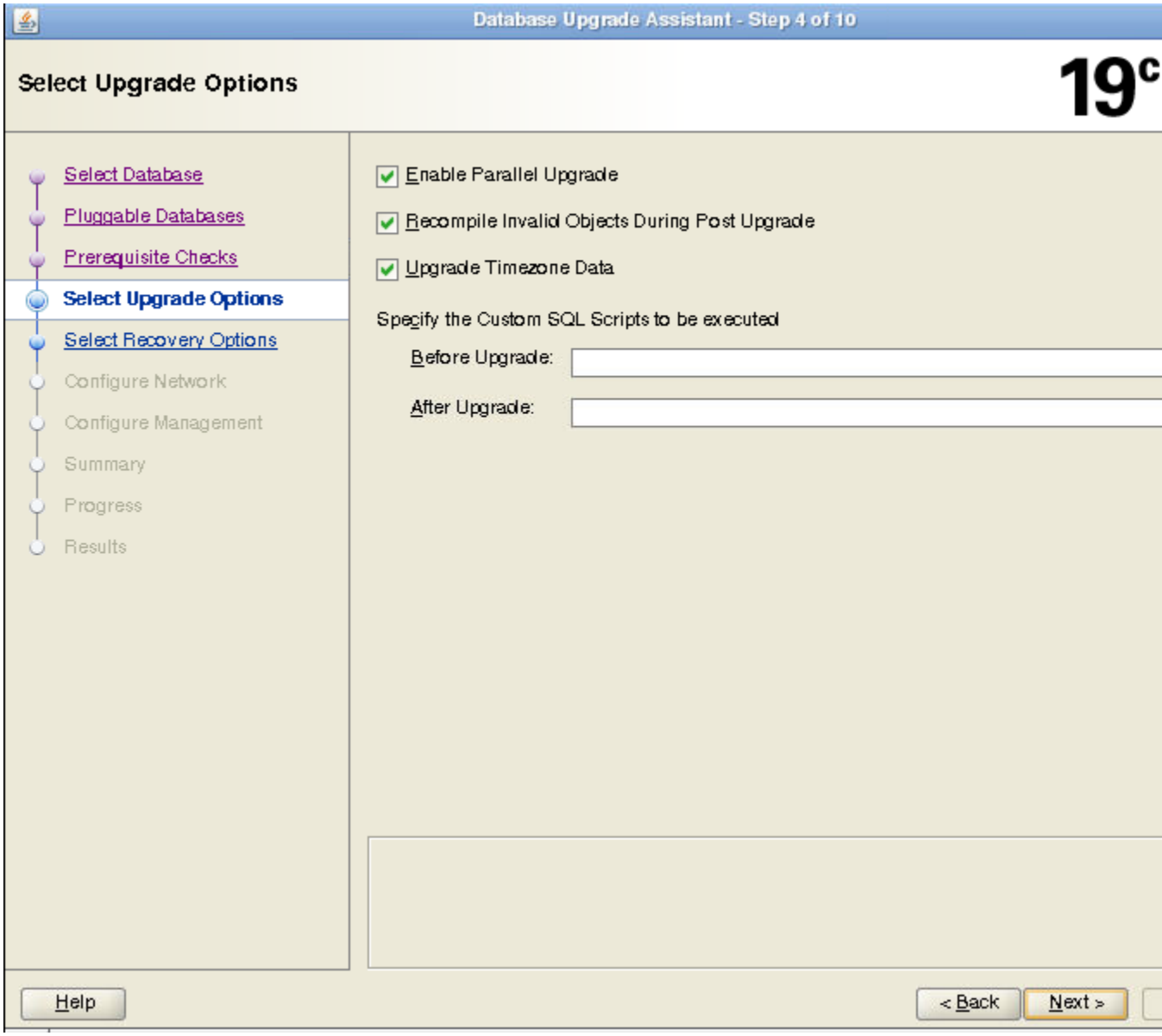
DBUAで修正できるエラーが表示されている場合、「修正および再チェック」を選択します。

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パラレル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パラレル実行は、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。

     ノート:

     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に、すべてのスタンバイ・データベースをもう一度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

     次のRMANバックアップ・オプションから選択します。

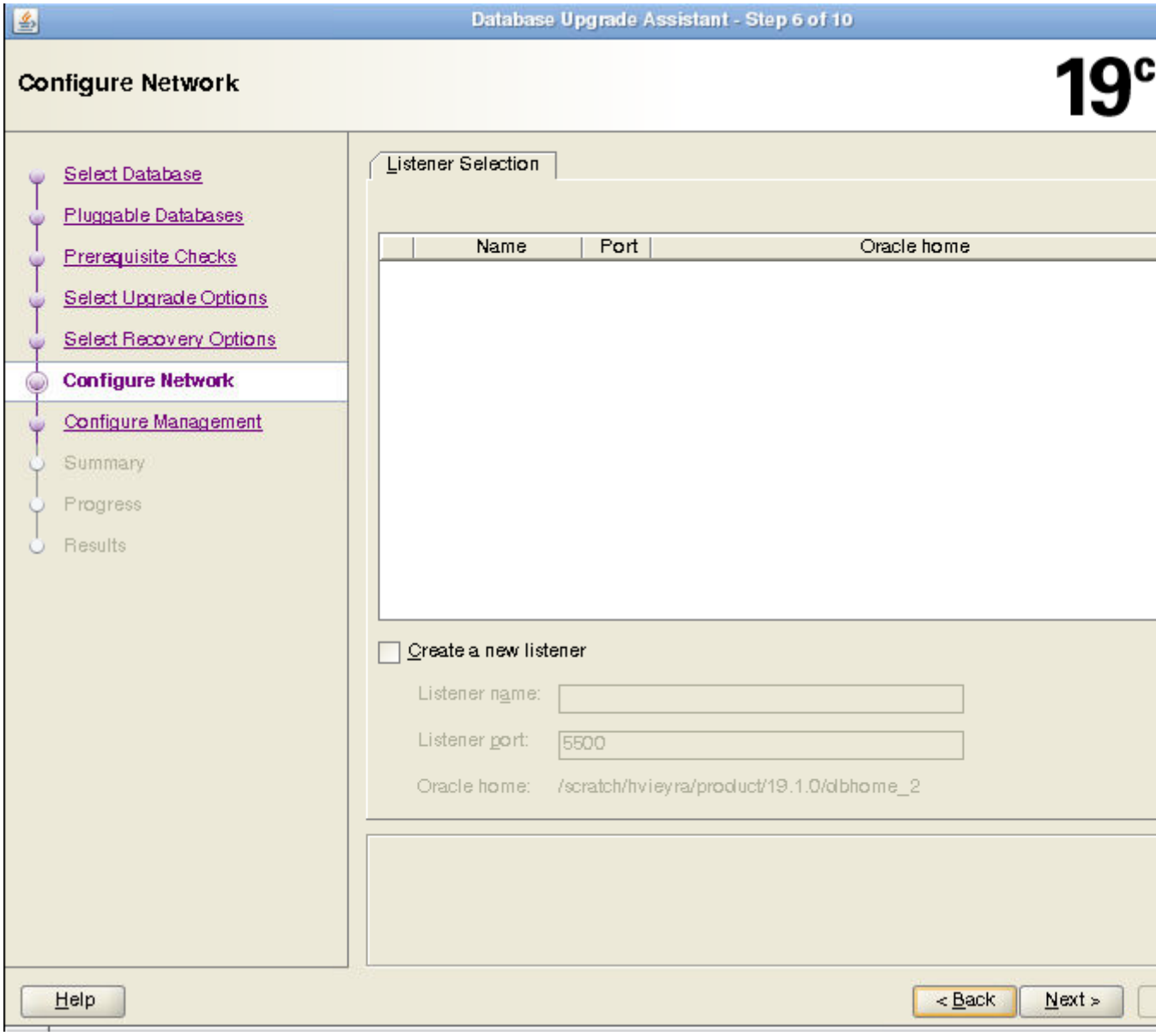
     - 新規オフラインRMANバックアップの作成:このオプションを選択した場合は、バックアップを配置する場所のパスを指定します。
     - ユーザー表領域をR/Oに設定した新規部分オフラインRMANバックアップの作成:このオプションを選択した場合、アップグレード中にユーザー表領域が読取り専用モードになります。
     - 最新の使用可能な完全RMANバックアップの使用:このオプションを選択した場合は、「リストア・スクリプトの表示/編集」をクリックして、使用するバックアップを選択します。

   - 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

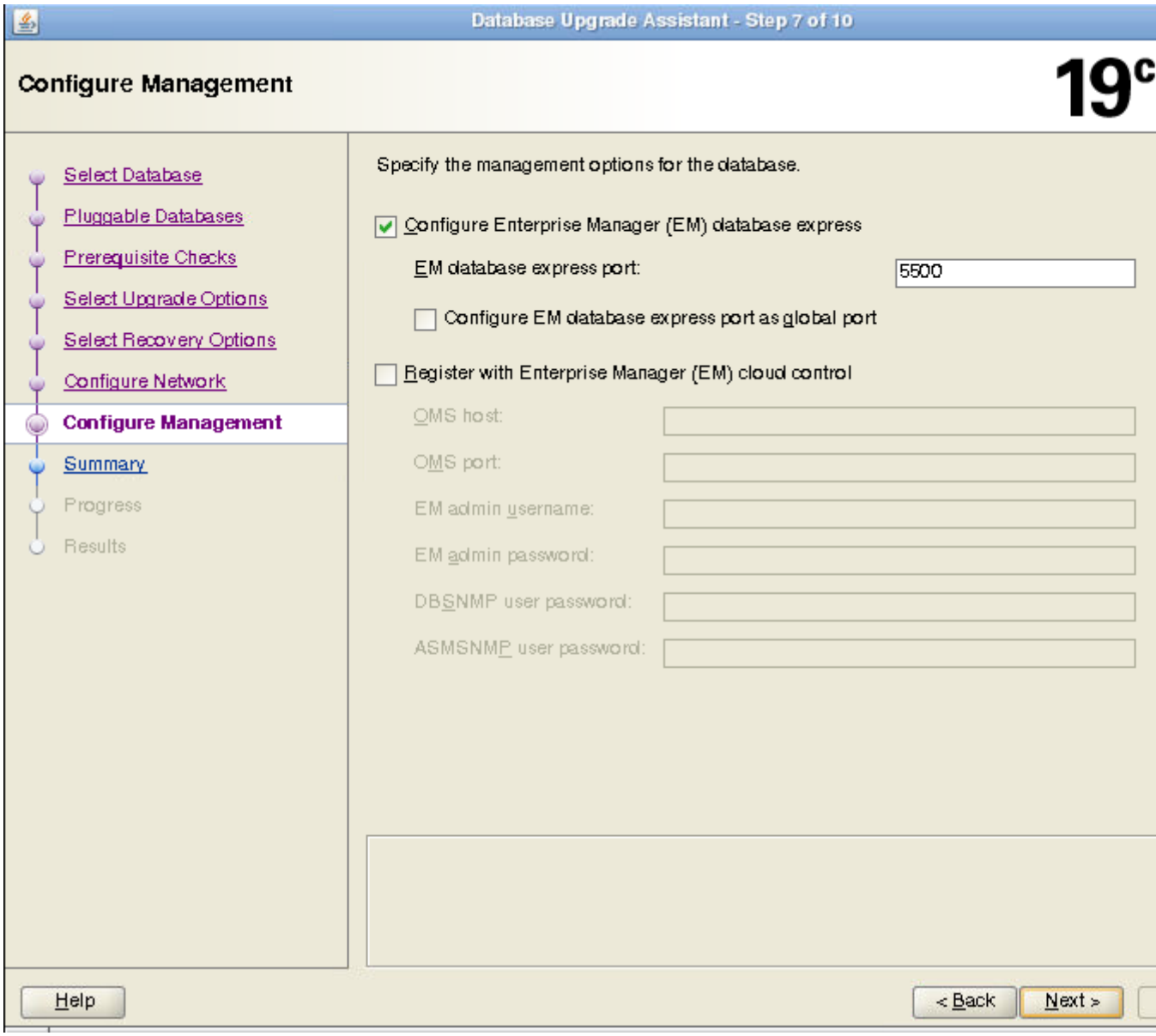
されます。

- 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。

> ノート:
>
> 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可能だった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。

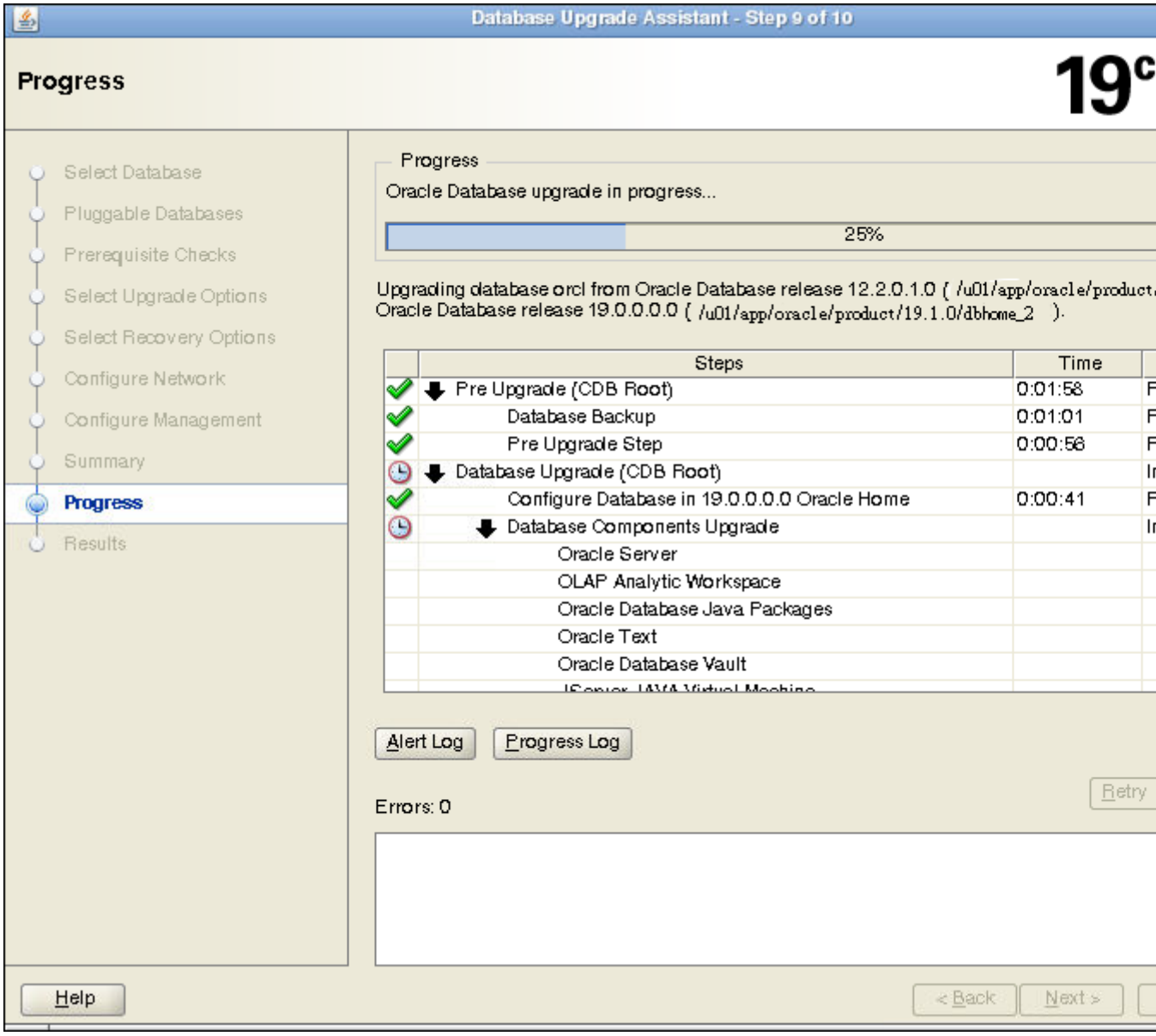
12. 「サマリー」ウィンドウが開きます。「サマリー」ウィンドウに、アップグレードのために指定した情報が表示されます。情報を確認するには、リストを下にスクロールします。サマリーには次のような情報が含まれます。

- 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。

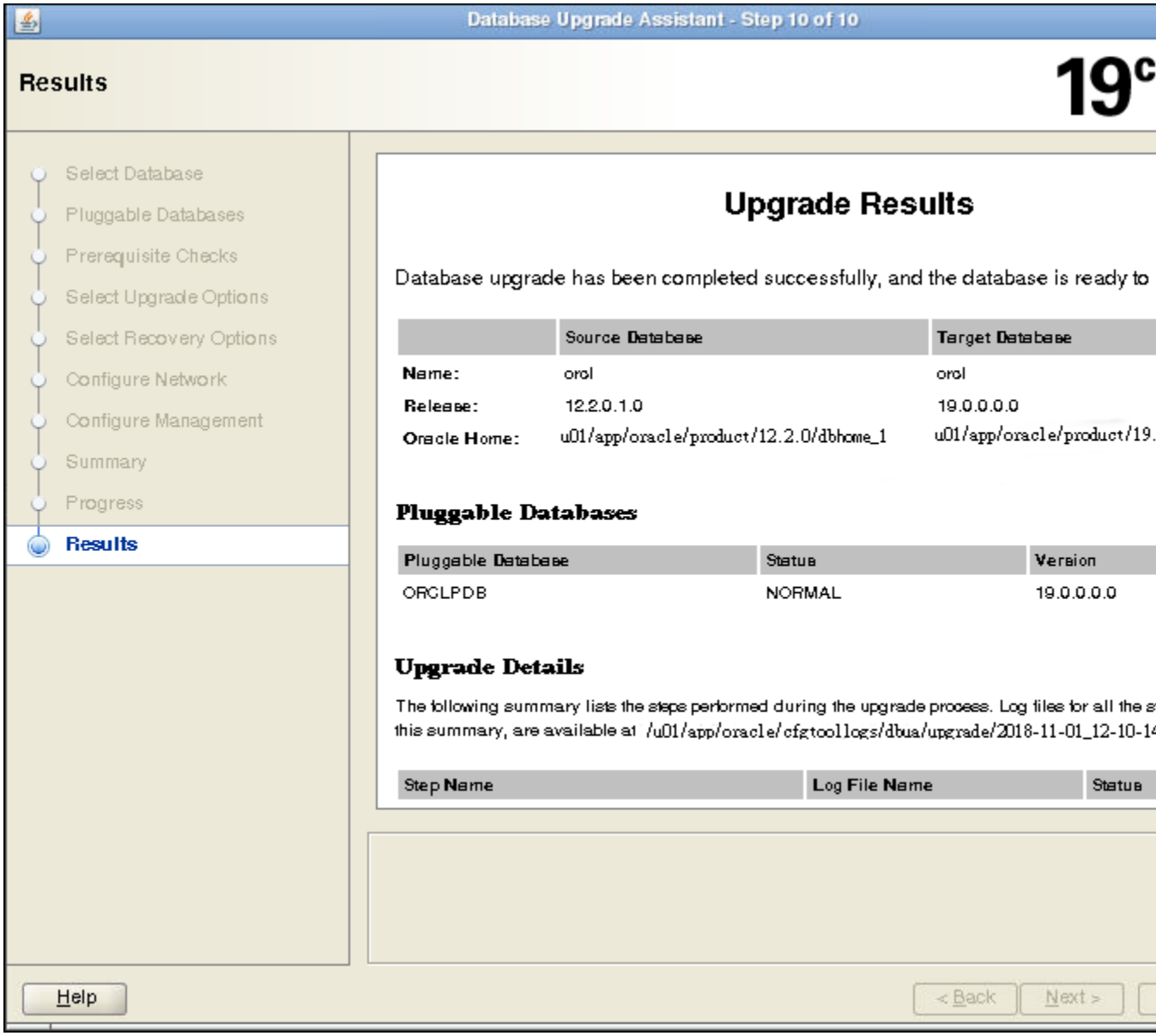
- 「サマリー」ウィンドウに修正が必要な情報が表示された場合は、更新する項目のリンクをクリックするか、または、「戻る」をクリックしてDBUA構成ウィザードを逆に移動します。
- 「サマリー」ウィンドウに表示されている情報が正しい場合、「終了」をクリックします。「終了」をクリックするとアップグレードが開始されます。

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。

13. 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、プラガブル・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。



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

ノート:

「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薦めします。

Oracle Database 12c 以上のリリースに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 Upgrade Assistant (DBUA)を使用したアップグレード

サイレント・モードのDBUAを使用したOracle Databaseのアップグレード

DBUAを-silentコマンドライン・オプションとともに使用すると、非対話型(「サイレント」)アップグレードを実行できます。

サイレント・モードでは、DBUAのユーザー・インタフェースは表示されません。DBUAは、メッセージ(情報、エラーおよび警告を含む)をORACLE_HOME/cfgtools/dbua/upgradesession_timestampのログ・ファイルに書き込みます(session_timestampはDBUAが実行したアップグレードのタイムスタンプを表します)。確実にアップグレードを行うために、DBUAによって生成されたログ・ファイルを確認することを強くお薦めします。

- サイレント・モードでのDBUAの実行
  この手順を使用して、DBUAを非対話(サイレント)モードで起動します。
- アクティブ・モードおよびサイレント・モードでのDBUAのコマンドライン構文
  この構文を使用して、対話形式または-silentオプション付きでDatabase Upgrade Assistant (DBUA)を実行します。

親トピック: Oracle Database Upgrade Assistant (DBUA)を使用したアップグレード

サイレント・モードでのDBUAの実行

この手順を使用して、DBUAを非対話(サイレント)モードで起動します。

1. サイレント・モードでDBUAを起動するには、dbua -silent -sidコマンドを入力します。このコマンドによってDBUAがサイレント・モードで起動され、アップグレードするデータベースが識別されます。

   たとえば、データベース名がORCLの場合、次のコマンドを入力します。

   ```
   dbua -silent -sid ORCL &
   ```

親トピック: サイレント・モードのDBUAを使用したOracle Databaseのアップグレード

アクティブ・モードおよびサイレント・モードでのDBUAのコマンドライン構文

この構文を使用して、対話形式または-silentオプション付きでDatabase Upgrade Assistant (DBUA)を実行します。

目的

コマンドライン・オプションを使用してDBUAを実行する場合、スクリプト形式ですべての有効なDBUAオプションを指定できます。スクリプト形式を使用すれば、グラフィック・ユーザー・インタフェース・ダイアログで構成情報の入力をしないで指定できます。

ファイル・パス

$ORACLE_HOME/directory_name

構文

使用方法: dbua [flag] [option]

フラグ

-createPartialBackup ユーザー表領域を読取り専用モードで設定して、新しいオフライン部分RMANバックアップを作成するフラグ。

-backupLocation RMANバックアップのパスを識別するフラグ。

-disableParallelUpgrade データベース・アップグレードのパラレル実行を無効にするフラグ。

-executePreReqs 指定されたデータベースに対してアップグレード前チェックのみを実行するフラグ。

-sid | -dbName

-sid

-dbName

-help - この使用方法のヘルプを表示します。

-ignorePreReqs アップグレード前チェックでエラー条件を無視します。

-silent 構成をサイレント・モードで実行します。

-sid | -dbName

-sid

-dbName

-skipListenersMigration データベースのアップグレードの一部としてリスナー移行プロセスをバイパスするフラグ。

オプション

DBUAのコマンドライン・オプションは次のとおりです。

[-asmsnmpPassword - ASMSNMP user password]

[-backupLocation - Directory-where-you-want-to-back-up-your-database-before-starting-upgrade

[-createGRP - [True | false] データベースがアーカイブ・ログおよびフラッシュバック・モードになっている場合に、保証付きリストア・ポイントを作成します。

[-createListener - [true [listenrName:lsnrPort] | false]] 以降のリリースのOracleホームにリスナーを作成します。trueの場合は、listenerName:lsnrPortを追加します

[-dbName - database-name]

[-oracleHome - Oracle-home-path-of-database]

[-sysDBAUserName - User-name-with-SYSDBA-privileges]

[-sysDBAPassword - Password-for-sysDBAUserName-user-name]

[-dbsnmpPassword - DBSNMP-user-password]

[-disableUpgradeScriptLogging - [true | false]] このコマンドはアップグレード処理中のSQLスクリプトの実行での詳細なログの生成を無効にします。デフォルトでは、これは有効になります。ログ生成を有効にするには、このコマンドを指定しないでください。

[-emConfiguration - [DBEXPRESS | CENTRAL | BOTH | NONE] Enterprise Managerの情報を指定します。

[-dbsnmpPassword - DBSNMP-user password]

[-emPassword - Enterprise-Manager-administration-user-password]

[-emUser - Enterprise-Manager-Administration-username-to-add-or-modify-targets]

[-emExpressPort - port-for-EM-Express]

[-omsHost - Enterprise-Manager-management-server-host-name]

[-omsPort - Enterprise-Manager-management-server-port-number]

[-asmsnmpPassword - ASMSNMP-user-password]

[-ignoreScriptErrors - [true | false]] カスタム・スクリプトでORAエラーを無視する場合にこのフラグを指定します。

[-initParam - name=value,name=value,...] name=value,name=valueという形式の初期化パラメータ値のカンマ区切りリストを指定します。

[-initParamsEscapeChar - [escape character] 特定のinitParamに複数の値がある場合のカンマのエスケープ文字。エスケープ文字が指定されていない場合は、バックスラッシュ(¥)がデフォルトのエスケープです

[-excludeInitParams - comma-delimited-list-of-initialization-parameters-to-exclude]

[-keepDeprecatedParams - [true | false]] データベースのアップグレード中に非推奨のパラメータが維持されます(または維持されません)。

[-localListenerWithoutAlias] TNS別名を使用せずにLOCAL_LISTENERを設定します。

[-listeners listenerName:Oracle Home, listenerName:Oracle Home, …] 既存のリスナーにデータベースを登録します。listenerName:Oracle Homeという形式のカンマ区切りリストを使用してリスナーを指定します。以前のリリースのOracleホームからのリスナーは、newer release,lsnrName2または-listeners lsnrName1:Oracle home path、-listeners lsnrName2:Oracle home pathに移行されます。DBUAは、指定されたリスナーを、Grid Infrastructureホーム(構成されている場合)、ターゲット・ホームおよびソース・ホームから検索します。

[-localRacSid - local-System-Identifier-of-cluster-database] クラスタ・データベースがOracle Cluster Registry (OCR)に登録されていない場合に使用します。

[-logDir - Path-to-a-custom-log-directory]

[-newGlobalDbName - New Global Database Name] このオプションは、Oracle Express Editionのアップグレードにのみ使用できます

[-newSid - New System Identifier] このオプションは、Oracle Express Editionのアップグレードにのみ使用できます

[-newInitParam - name=value,name=value,...] name=value,name=valueという形式の初期化パラメータ値のカンマ区切りリストを指定します。このオプションは、ターゲットのOracleホームでのみ許可されるパラメータを指定するために使用します。

[-initParamsEscapeChar - escape character] 特定のinitParamに複数の値がある場合のカンマのエスケープ文字を指定します。エスケープ文字を指定しない場合は、バックスラッシュ(¥)がデフォルトのエスケープです。

[-oracleHomeUserPassword - Oracle-Home-user-password]

[-pdbs - [All | NONE | pdb, pdb,…]] すべてのPDBを選択する場合はALL、PDBを選択しない場合はNONE、またはアップグレードする

プラガブル・データベース(PDB)の名前のカンマ区切りリストを指定します。

-sid | -dbName

-sid - System Identifier

[-oracleHome - データベースのOracleホームのパス]

[-sysDBAUserName - User-name-with-SYSDBA-privileges]

[-sysDBAPassword - Password-for-sysDBAUserName-user-name]

-dbName - Database-Name

[-oracleHome - Oracle-home-path-of-database]

[-sysDBAUserName - User-name-with-SYSDBA-privileges]

[-sysDBAPassword - Password-for-sysDBAUserName-user-name]

[-pdbsWithPriority - pdb:priority, pdb:priority, pdb:priority,...] アップグレードするプラガブル・データベース(PDB)のカンマ区切りリスト(対応する優先度(1が最優先)を含む)を指定します。

-sid | -dbName

-sid - system-identifier

[-oracleHome - Oracle-home-path-of-database

[-sysDBAUserName - User-name-with-SYSDBA-privileges]

[-sysDBAPassword - Password-for-sysDBAUserName-user-name]

-dbName - database-name

[-oracleHome - Oracle-home-path-of-the-database]

[-sysDBAUserName - User-name-with-SYSDBA-privileges]

[-sysDBAPassword - Password-for-sysDBAUserName-user-name]

[-performFixUp - [true | false] サイレント・アップグレード・モードで修正を有効または無効にします。

[-postUpgradeScripts - SQLscript, SQLscript,…] SQLスクリプトのカンマ区切りリストを完全なパス名で指定します。アップグレードが完了すると、これらのSQLスクリプトが実行されます。

[-preUpgradeScripts - SQLscript, SQLscript,…]SQLスクリプトのカンマ区切りリストを完全なパス名で指定します。アップグレードが開始される前に、これらのSQLスクリプトが実行されます。

[-recompile_invalid_objects - [true | false]] trueの場合は、アップグレードの一部として、無効なオブジェクトが再コンパイルされます。

[-upgrade_parallelism - number] パラレル・アップグレードに使用するCPU数の数値。

[-upgradeTimezone - [true | false]] trueの場合は、アップグレード中にデータベースのタイムゾーン・ファイルがアップグレードされます。

[-upgradeXML - Path-to-existing-preupgrade-XML-file] このオプションは、インプレース・データベース・アップグレードにのみ適用されます。

[-useExistingBackup - [true | false]] 既存のRMANバックアップを使用したデータベースのリストアを有効にする場合に使用します。

[-useGRP - Name-of-existing-guaranteed-restore-point] 指定された保証付きリストア・ポイントを使用したデータベースのリストアを有効にする場合に使用します。

例C-1 DBUAを使用するアップグレードのデータベースの選択

次のコマンドでは、アップグレードのためにデータベースorclを選択します。

```
dbua -sid orcl
```

ノート:

DBUA コマンドを使用してパスワードを設定できます。Oracle Database 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。

例C-2 非対話型(サイレント)オプションを使用したDBUAによるアップグレードのデータベースの選択

次のコマンドでは、非対話(サイレント)オプションを使用してアップグレードのためにデータベースorclを選択します。

```
dbua -silent -sid orcl
```

例C-3 非対話(「サイレント」)モードでのDBUAの実行のユース・ケース

後続の例では、DBUAを非対話型(「サイレント」)オプションとも使用して、様々なアップグレード・シナリオを実行する方法を説明しています。

```
dbua -silent -sid sidb112 -backupLocation /u01/sidb1123/backup -sysDBAUserName sys -
sysDBAPassword r3aDy2upg -oracleHome /u01/app/product/11.2.0/dbhome_1 -
upgradeTimezone true dbua -silent -sid sidb1123 -backupLocation /u01/sidb1123/backup
-sysDBAUserName sys -sysDBAPassword r3aDy2upg -oracleHome
/u01/app/product/11.2.0/dbhome_1 -upgrade_parallelism 1 -upgradeTimezone true

dbua -silent -sid db1124 -backupLocation /u01/sidb1123/backup -sysDBAUserName sys -
sysDBAPassword r3aDy2upg -performFixUp true -upgradeTimezone true

dbua -silent -dbName rdbcdb -oracleHome /u01/app/product/11.2.0/dbhome_1 -
sysDBAUserName sys -sysDBAPassword r3aDy2upg -backupLocation /u01/sidb1123/backup -
recompile_invalid_objects true -upgradeTimezone true

dbua -silent -dbName amdb -oracleHome /u01/app/product/11.2.0/dbhome_1 -
sysDBAUserName sys -sysDBAPassword r3aDy2upg -recompile_invalid_objects true -useGRP
GRP_20170620bfupgrade -upgradeTimezone true

dbua -silent -dbName rdb12 -oracleHome /u01/app/product/12.2.0/dbhome_2 -
sysDBAUserName sys -sysDBAPassword r3aDy2upg -backupLocation /u01/sidb12/backup -
recompile_invalid_objects true -upgradeTimezone true

dbua -silent -dbName ronedb -oracleHome /u01/app/product/12.2.0/dbhome_2 -
sysDBAUserName sys -sysDBAPassword r3aDy2upg -recompile_invalid_objects true -
upgradeTimezone true -createGRP true
```

ノート:

セキュリティのベスト・プラクティスの詳細は、『*Oracle Database* セキュリティ・ガイド』を参照してください。

関連項目

- 『Oracle Databaseセキュリティ・ガイド』

親トピック: サイレント・モードのDBUAを使用したOracle Databaseのアップグレード

D 異なるORACLE_HOME所有者でのDBUAの実行

Oracle Databaseホームが異なるオペレーティング・システム・ユーザー・アカウントによって所有されているか、「upgrade.xmlが見つかりません」エラーが発生した場合、このトピックを参照してください。

デフォルトのDBUAアップグレードでは、ソースOracleホームとターゲットOracleホームの両方が同じユーザーによって所有されていると想定しています。各Oracleホームが同じユーザーに所有されていない場合、データベース・ファイルの権限を変更し、DBUAに追加のパラメータを渡す必要があります。これを行わない場合、アップグレード中にDBUAの「前提条件チェック」ページに「upgrade.xmlが見つかりません」エラーがレポートされます。このエラーが修正されるまで、アップグレードを継続することはできません。

- すべてのOracle Databaseインストール所有者は、プライマリ・グループとしてOINSTALLグループ(またはOracleインベントリ・グループ)に指定したグループを保持している必要があります。すべてのデータベース・ファイル(データファイル、REDOファイル、制御ファイル、アーカイブ・ログの保存先、リカバリ領域、SPFILEおよびパスワード・ファイル)が新しいターゲット・リリースとソース(以前の)リリースの両方のバイナリ所有者によって読取りおよび書込み可能であることを確認します。そうでない場合は、各インストール所有者がそのプライマリ・グループと同じグループを持っていることを確認し、OINSTALLグループのメンバーが以前のリリースとそれ以降のリリースのすべてのOracle Databaseファイルおよびディレクトリに対する読取り/書込みアクセス権を持っていることを確認します。
- -logdirコマンドライン・オプションを指定してDBUAを実行し、新しいリリースと以前のリリースの両方のバイナリ所有者が書き込むことのできるディレクトリを指定します。たとえば、/tmpのように入力します。DBUAは、logdirパラメータで指定されたディレクトリを使用して、アップグレード前情報ツールからの出力を格納し、アップグレード中に生成されたDBUAログ・ファイルを格納します。以前のリリースのOracle Databaseインスタンスのアップグレード前情報ツールは、以前のリリースのOracle Databaseインストール所有者のユーザー・アカウントとして実行します。

  たとえば:

  ```
  dbua -logdir /tmp
  ```

索引

記号 A B C D E F G H I K L M N O P R S T U V W X Z

記号

- .NET
  - PromotableTransactionの非推奨 10.8.4.8

A

- アクセス制御リスト(ACL)
  - ネットワーク・ユーティリティ・パッケージへのアクセス権の付与 B.2.8.1
  - XDB ACLの移行 10.8.1.5
- ACFS-9427:「ADVM/ACFSドライバのアンロードに失敗しました。」5.11
- ACFS-9428「ADVM/ACFSドライバのロードに失敗しました。」5.11
- ACL
  - 「アクセス制御リスト(ACL)」を参照
- ネットワーク・ユーティリティ・パッケージのACLの追加 B.2.8.2
- addnode.bat
  - 非推奨 10.6.4.6
- addnode.sh
  - 非推奨 10.6.4.6
- 手動アップグレード後の調整 6.8.5.2
- amd_exists 4.3.1
- apex_upgrade_msg 4.3.1
- アプリケーション・コード
  - アップグレード後に変更されない 7.7.1
- アプリケーション
  - パッケージの依存性の確認 6.3.7
  - 互換性 7.2
  - リンクとアップグレード 7.4.2
  - より新しいライブラリへのリンク 7.4.2.2
  - 古いサーバーでの実行 7.4.1.2
  - アップグレード 7
    - クライアント/サーバー構成 7.3
    - 互換性規則 7.4
    - オプション 7.7
    - 再リンク規則 7.4
- apxrelod.sqlファイル
  - ダウングレード後の再ロード 8.5
- ArchiveLogTargetの非推奨 10.6.4.3

- ARGUMENTSユーザービュー
  - 変更 10.7.1.6
- ASM_PREFERRED_READ_FAILURE_GROUPS 6.6.4
- asmcmd pwcreate (非推奨) 10.6.4.5
- 属性
  - xdb defaultTableSchema (非推奨) 10.8.4.20, 10.9.3.37
  - xdb maintainOrder (非推奨) 10.8.4.20, 10.9.3.37
  - xdb mapUnboundedStringToLob (非推奨) 10.8.4.20, 10.9.3.37
  - xdb maxOccurs (非推奨) 10.8.4.20, 10.9.3.37
  - xdb SQLCollSchema (非推奨) 10.8.4.20, 10.9.3.37
  - xdb SQLSchema (非推奨) 10.8.4.20, 10.9.3.37
  - xdb srclang (非推奨) 10.8.4.20, 10.9.3.37
  - xdb storeVarrayAsTable (非推奨) 10.8.4.20, 10.9.3.37
  - xdb translate (非推奨) 10.8.4.20, 10.9.3.37
- 監査
  - アップグレード後の監査レコードの転送の概要 6.4.17.1
  - データベース、使用不可の場合 2.5.12
  - 監査レコードの統合監査証跡へのロード 2.5.12
  - アップグレード後の統合監査レコードの転送 6.4.17.2
  - 統合監査移行
    - 概要 6.4.14
    - 監査オプション 6.4.14.1
    - 非統合監査に関するドキュメント参照 6.4.14.5
    - 移行後の古い監査レコードの管理 6.4.14.3
    - プロシージャ 6.4.14.2
    - removing unified auditing 6.4.14.4
- 自動診断リポジトリ(ADR) 1.4.3
- 自動UNDO管理
  - 移行 6.4.12
- Autoupgrade
  - jobid 3.3.1
- AutoUpgrade 3.1
  - 高速リカバリ領域(FRA) 3.14
  - 保証付きリストア・ポイント 3.14
  - アプトオブプレース・パッチ適用 2.1.5
  - パスワード・ファイル 3.14
  - 透過的データ暗号化(TDE) 3.14
- autoupgrade -config-
  - clear_recovery_data 3.8.4, 3.8.9, 3.8.10
  - -console 3.8.4, 3.8.9, 3.8.10
  - -debug 3.8.4, 3.8.9, 3.8.10
  - -mode 3.8.4, 3.8.9, 3.8.10
  - -noconsole 3.8.4, 3.8.9, 3.8.10

    - -restore_on_fail 3.8.4, 3.8.9, 3.8.10
    - -zip 3.8.4, 3.8.9, 3.8.10
- autoupgrade -config_values-
    - clear_recovery_data 3.8.5
    - -console 3.8.5
    - -debug 3.8.5
    - -mode 3.8.5
    - -noconsole 3.8.5
    - -restore_on_fail 3.8.5
    - -zip 3.8.5
- autoupgrade -preupgrade 3.8.13

B

- データベースのバックアップ 2.12
- バックアップ
    - アップグレード後 6.4.1
    - ダウングレード前 8.4
    - Zero Data Loss Recovery Applianceの制限 10.8.1.12
- ベンチマーク 2.1.6.7
- プリコンパイラおよびOracle Call Interface (OCI)アプリケーションのアップグレード用のオプションの利点 7.7
- BFILE
    - 移行 6.4.13
- BLOB
    - 移行 6.4.13
- BP (バンドル・パッチ) 2.5.1, 6.9
- ブローカ構成
    - エクスポート 2.1.8.3
- バンドル・パッチ・セット 1.3.1

C

- データベースのワークロードの取得およびリプレイ 2.1.6.6.1
- 大/小文字の区別
    - パスワード用 6.4.5
- catcon
    - ダウングレード 8.9.7
    - SQLコマンドの実行 8.9.9
- catcon.pl 4.2.3, 4.6.5, 8.6.2, B.1
    - PDBのチェックに使用 B.2.6
- catctl.pl 4.1.2, 4.1.3, B.1
    - シェル・コマンドの実行による起動 4.1.1

- catdwgrd.sql B.1
- catdwgrd.sqlスクリプト 8.6.2
- catnoamd.sql B.1
- catnoexf.sql B.1
- CATRELOD.SQLスクリプト 8.6.2, 8.7
- catupgrd.logファイル
    - CDBおよびPDB 4.7.4.1
- catupgrd.sql
    - catctl.plによる置換 4.4
- CATUPGRD.SQLスクリプト
    - 「CATCTL.PLスクリプト」を参照
    - サポート終了
        - 「CATCTL.PLスクリプト」を参照
- catuppst.sql B.1
- CDB_JAVA_POLICY 5.10
- CDB_REGISTRY 8.9.10
- CDB C.1
    - catctl.plログ・ファイル 4.7.4.1
    - ダウンロード 8.5
    - Oracle Label Security 4.6.4
    - CDBおよびPDBのアップグレードの再実行 4.7.3.1
    - アップグレードの再実行 4.7.3
    - 失敗したフェーズからの再開 4.7.4
    - アップグレードの再開 4.7.4.2
    - アップグレード・シナリオ 4.2
    - アップグレード 4.2.1
    - catconの使用 8.9.9
- パスワードの変更
    - オラクル社が提供するアカウント 6.8.1
- PDBのアップグレード優先度の変更 4.2.4
- 新機能を使用するためのスクリプトの変更 7.8
- クライアントおよびサーバー
    - 構成とアップグレード 7.3.1
- クライアント/サーバー構成 1.4.4
- クライアントサイド動的ライブラリ 7.4.2.4
- クライアント・ソフトウェア
    - アップグレード 7.4.2
- CLOB
    - 移行 6.4.13
- clone.pl
    - 非推奨 10.6.4.7
- CLUSTER_DATABASE初期化パラメータ 2.7, 5.1
- コマンドライン・アップグレード

    - 「手動アップグレード」を参照
- コマンド
    - orabase 8.6.1
    - orabasehome 8.6.1
    - oracle_home 8.6.1
- 互換性
    - アプリケーション 7.2
    - Oracleリリース間 1.3
    - 非互換性の調査 8.3
    - COMPATIBLE初期化パラメータ 1.3.3.3
    - ダウングレード 1.3.3.5
    - Oracle Databaseの概要 1.3.3.1
- COMPATIBLE初期化パラメータ 1.3.3.3, 6.6.3
    - PDBの互換性 1.3.3.3
    - レベルの確認 1.3.3.7
    - ダウングレードの考慮点 1.3.3.5
    - デフォルト 1.3.3.3
    - Javaの初期遅延 1.3.3.3
    - 推奨事項 1.3.3.2
    - 設定 6.8.5.1
    - 値 1.3.3.4
- コンポーネントのステータス 5.12.1
- 圧縮
    - sqlnet.oraファイル・パラメータ 2.5.3
- 圧縮方式 2.5.3
    - SQLNET.COMPRESSION 2.5.3
    - SQLNET.COMPRESSION_LEVELS 2.5.3
    - SQLNET.COMPRESSION_THRESHOLD 2.5.3
- 構成ファイル
    - コピー 2.7
- continuous_mineオプションのサポート終了 10.6.2.3
- 構成ファイルのコピー 2.7
- CREATE pfile FROM spfile 2.7
- CREATE TABLE AS 1.1.2
- crsuser (非推奨) 10.7.2.7
- カーソル・キャッシュ, SMB 2.1.6.6.3

D

- データベース・リンク B.2.8.3
- データベース・リプレイ
    - アップグレード前のデータベースのワークロード 2.1.6.6.1
- データベース

    - ダウンロード 8.6.1
    - ダウングレードとOracle Internet Directoryの登録 C.1
    - 手動ダウングレード 8.6.2
    - クライアント・ソフトウェアのアップグレード 7.4.2
- Database Upgrade Assistant(DBUA)
    - 利点 2.1.4.2, 2.1.4.5
    - マルチテナント・アーキテクチャ・アップグレード 10.8.1.2
    - CDB C.1
    - コマンドライン・オプション C.7.2
    - -executePrereqsオプション 10.8.1.2
    - 保証付きリストア・ポイント C.1
    - 非CDB C.1
    - 非対話型(サイレント)コマンドライン構文 C.7.2
    - 一時停止と続行 10.8.1.2
    - PDB C.1
    - listener.oraファイルへのデータベースの登録 2.5.3
    - 削除された機能 10.8.1.2
    - 実行 C
    - サイレント・モード C.7
    - スタンドアロン前提条件チェック 10.8.1.2
    - 起動 C.3, C.6
- Database Upgrade Assistant (DBUA)の新機能 10.8.1.2
- Database XE 1.5.6
- データのコピー
    - エクスポート/インポートの使用 9
- データ定義言語(DDL) 1.3.5
- データ・ディクショナリ
    - アプリケーションに影響する変更の概要 7.2
    - 状態の確認 6.2
- DataGuardSyncLatency (非推奨) 10.6.4.3
- データ・ポンプ・エクスポート/インポート
    - 使用することの利点 2.1.4.4
    - 推奨事項 9.1
    - オリジナルのエクスポート/インポートとの対比 9.1
    - 使用のタイミング 9.1
    - 本番データベースのサブセット 2.10
- データ型 10.9.1.7
- DBA_ACL_NAME_MAP 10.8.1.5
- DBA_REGISTERED_MVIEW_GROUPS
    - サポート終了 10.8.2.6
- DBA_REGISTRY 8.9.10
- DBA_REGISTRYビュー 5.12.1
- dbdowngradeユーティリティ 8.6.1

- DbFileNameConvertの非推奨 10.6.4.3
- DBMS_DEBUG
    - 非推奨 10.8.4.3
- DBMS_DSTパッケージ B.2.8.4
- DBMS_LDAP.initパラメータ B.2.8.1
- DBMS_LDAPパッケージ B.2.8.1
- DBMS_NETWORK_ACL_ADMIN
    - 非推奨 10.8.1.5
- DBMS_NETWORK_ACL_ADMIN_APPEND_HOST_ACE 10.8.1.5
- DBMS_NETWORK_ACL_ADMINパッケージ B.2.8.2
- DBMS_PDBパッケージ 4.6.2
- DBMS_ROLLING 10.8.1.3
- DBMS_STATS.GATHER_DICTIONARY_STATS 8.9.7
- DBMS_STATS.GATHER_DICTIONARY_STATSプロシージャ 8.9.9
- DBMS_STATS.GATHER_FIXED_OBJECTS_STATS 8.9.8
- DBMS_STATSパッケージ
    - 統計表のアップグレード 6.3.13
- DBMS_XMLQUERY, 非推奨 10.7.5.4
- DBTエラー・メッセージ 10.8.1.2
- DBUA
    - 「Database Upgrade Assistant」を参照。
- dbupgdiag.sql 6.2
- dbupgrade 4.1.1, 4.1.2, B.1
    - 手動アップグレード 4.6.5
- dbupgrade.cmd 4.1.2, B.1
- dbupgradeシェル・コマンド
    - 引数 4.1.3
- defaultTableSchema属性(非推奨) 10.8.4.20, 10.9.3.37
- demo
    - 読取り専用Oracleホームでの置換え 6.3.17
- 18cの非推奨になった機能 10.7.5
- 非推奨となった初期化パラメータ 10.8.5
- 非推奨となったパラメータとサポートが終了したパラメータ 2.7
- DESCRIBEプロシージャ 4.6.2
- サポートが終了した初期化パラメータ 10.8.3
- 開発者アプリケーション
    - Formsのアップグレード 7.9
- DGMGRL 10.8.1.3
- 診断データ 1.4.3
- 直接のアップグレード 1.1.3
- ディスク・グループの互換性 6.6.3
- ディスク
    - 優先読取りの障害グループの指定 6.6.4

- 分散アップグレード 3.9.2.41
- ダウングレード
  - Oracle Data Guard Broker構成ファイルのバックアップ 2.1.8.2
  - パッチの再適用 8.9.1
- ダウングレード
  - ディクショナリ統計の収集 8.9.7
  - データベースのバックアップ 8.4
  - CATRELOD.SQL 8.6.2, 8.7
  - 非互換性の調査 8.3
  - レジストリの確認 8.9.10
  - データベース・リンク B.2.8.3
  - ダウングレード・ログ・ファイルの場所 8.6.1
  - Oracle Data Guardスタンバイ・データベース 8.2
  - Oracle Enterprise Manager 8.9.5
  - ORADIM 8.6.2
  - パッチ・セット・リリース 8.1
  - 修正したオブジェクト統計の再収集 8.9.8
  - 失効している統計の再収集 8.9.9
  - スクリプト 8.6.1, 8.6.2
    - 再実行 8.6.2
- DV_PUBLIC
  - 非推奨 10.6.4.13
- DV_REALM_OWNER
  - 非推奨 10.6.4.11
- DV_REALM_RESOURCE
  - 非推奨 10.6.4.12
- dvsys.dbms_macadm.enable_dv() 6.3.19.1

E

- em_present 4.3.1
- emca -restoreコマンド 8.9.5
- emdwgrd B.1
- emdwgrdユーティリティ 8.9.5
- emremove.sql B.1
- エミュレーション 2.10.1
- パスワードの大/小文字区別の強制 6.4.5
- 環境変数
  - アップグレードに必要 4.2.3, 4.6.5
- exafusion_enabledのサポート終了 10.6.3.1
- 除外リスト
  - 概要 4.2.5
  - PDBのアップグレード 4.2.7.5

    - 再開 4.7.3.2
- エクスポート/インポート
    - メリットとデメリット 2.1.4.4
    - 利点 2.1.4.4.2
    - アップグレードされたデータベースへの影響 2.1.4.4.1
    - 推奨事項 9.1
    - 時間要件 2.1.4.4.3
- エクスポート
    - ブローカ構成 2.1.8.3
- エクスポート・ユーティリティ 9.1
    - データのコピー 9
- 拡張データ型サポート
    - 非推奨 10.7.5.2
- 拡張遠距離クラスタの構成
    - 優先読取りディスク 6.6.4
- エクステント
    - セカンダリからの読取り 6.6.4

F

- 失敗したフェーズ 4.7.1
- 高速リカバリ領域 5.6
- ファイングレイン・アクセス・コントロール
    - ネットワーク・ユーティリティ・パッケージ B.2.8.1
- フレックス・クラスタ・アーキテクチャ(非推奨) 10.7.5.12
- Forms
    - Oracle Formsアプリケーションのアップグレード 7.9
- フル・トランスポータブル・エクスポート/インポート 4.5

G

- GET_MODELビュー
    - 非推奨 10.7.5.3

H

- Hardware Assisted Resilient Data(HARD)
    - システムのアップグレード 6.8.5.1
- HttpUriTypeタイプ B.2.8.1

I

- IFILE (インクルード・ファイル)
  - エントリの編集 2.7
- イメージ
  - インストール 1.7, 7.4.2.1, 10.7.1.4
- インポート・ユーティリティ 9.1
  - データのコピー 9
- 包含リスト 4.7.3.4
- 包含リスト
  - 概要 4.2.5
- 非互換性
  - 確認 8.3
- init.ora
  - SGA権限 10.8.1.4
- 初期化パラメータ
  - 調整 2.7, 6.8.5
  - ASM_PREFERRED_READ_FAILURE_GROUPS 6.6.4
  - COMPATIBLE 1.3.3.3, 6.6.3
- インストール
  - Oracle Databaseソフトウェア 2.3, 2.4
- インストール・ログ
  - コンポーネント・アップグレード・スクリプト 5.12.1
- インスタンス
  - ダウングレード後に起動 8.6.2
- インテリジェント・データ配置(IDC)
  - 非推奨 10.8.4.4
- 中間アップグレード 1.1.3
- 中間リリース
  - 中間アップグレード 1.1.3
- 相互運用性 1.3.4
- 無効なオブジェクト
  - utlrp.sql 1.3.5
  - 再コンパイル 6.3.2
  - utlrp.sqlスクリプト 4.2.3, 4.6.5, 5.12.1, 8.6.2
- INVALIDオブジェクト 5.3, 5.4
- INVALIDステータス
  - コンポーネントのステータス 5.12.1

K

- ナレッジ・ベース 1.1.4

L

- listener.oraファイル 2.5.3
    - 変更 2.5.3
- リスナー 6.8.5.2
    - Oracle RACのアップグレード 6.5
    - Oracle Net Configuration Assistantを使用した変更 2.5.3
- ロード
    - アップグレード時の同時要求のレベル 2.1.6.7
- ロード・テスト 2.1.6.7
- log_archive_dest max_connections属性の非推奨 10.6.3.2
- LOG_ARCHIVE_LOCAL_FIRSTのサポート終了 10.8.3
- LogArchiveFormatの非推奨 10.6.4.3
- LogArchiveMaxProcessesの非推奨 10.6.4.3
- LogArchiveMinSucceedDestの非推奨 10.6.4.3
- LogArchiveTraceの非推奨 10.6.4.3
- LogFileNameConvertの非推奨 10.6.4.3
- ロジカル・スタンバイ・データベース
    - ローリング・アップグレード 1.5.1
- LogMiner
    - CONTINUOUS_MINEの非推奨 10.8.4.5
- lsinventoryコマンド 6.3.5
- lsnrctlコマンド
    - Oracle Grid Infrastructureホーム 2.5.3

M

- maintainOrder属性(非推奨) 10.8.4.20, 10.9.3.37
- 手動修正
    - 例 4.3.1
- 手動アップグレード 1.1.2, 6.8.5.2
    - 利点 2.1.4.3
    - データベースのバックアップ 2.12
    - OCR構成 6.8.4
- 手動アップグレード 4.2.3
- 手動アップグレードの例 4.3
- 手動アップグレード
    - 再実行または再起動 4.7.1
- mapUnboundedStringToLob属性(非推奨) 10.8.4.20, 10.9.3.37
- matches Oracle XQuery関数(非推奨) 10.8.4.20, 10.9.3.37
- matches XQuery関数 10.8.4.20, 10.9.3.37
- max_connectionsのサポート終了 10.6.2.5
- xdb名前空間のmaxOccurs属性(非推奨) 10.8.4.20, 10.9.3.37
- 移行
    - 定義 1.1.1

- データの移行 1.1.1
  - 異なるオペレーティング・システム 1.6.2
- lsnrctlコマンドを使用したOracleホームからのリスナーの移行 2.5.3
- データ・ポンプ・エクスポート/インポートによるデータの移動 9.1
- マルチテナント
  - アップグレードの再開 4.7.4.2
- 複数のOracleホームのサポート
  - 利点 1.4.4
- マルチテナント
  - アップグレード後の統合監査レコードの転送 6.4.17.2
- マルチテナント・アーキテクチャ
  - パラレル・アップグレード 4.2.6.2
- マルチテナント・アーキテクチャ・データベース
  - アップグレード・シナリオ 4.2
- マルチテナント・アーキテクチャ・データベース
  - PDB COMPATIBILITYパラメータの設定 1.3.3.3
- マルチテナント・コンテナ・データベース 4.2.3
- マルチテナント・コンテナ・データベース
  - 「CDB」を参照
- マルチテナント・データベース
  - リストを使用したアップグレード優先度の設定 4.2.5
- 複数バージョン 1.4.4
- mv_refresh 4.3.1
- My Oracle Support
  - ナレッジ・ベース 1.1.4

N

- NCLOB
  - 移行 6.4.13
- ネットワーク管理ファイルの場所 6.3.4
- ネットワーク名とリスナー 6.8.5.2
- ネットワーク
  - ネットワーク・ユーティリティ・パッケージへのACLアクセス権の付与 B.2.8.1
- 新機能
  - アップグレード後の追加 6.4.9
  - 使用するためのスクリプトの変更 7.8
- 新機能，概要 2.1.1
- NO AUTHENTICATIONステータスのアカウント 2.1.7, 7.6
- noncdb_to_pdb.sqlスクリプト 4.6.3
- 非CDBアーキテクチャ
  - 非推奨 10.8.4.6
- 非CDB C.1

- 非CDBからCDBおよびPDBへのアップグレード 1.5.1
- NO SCRIPTステータス 5.12.1
- アップグレードしたアプリケーションを再リンクしない 7.7.1
- NVARCHAR2データ型
    - EXTENDED 10.9.1.7

O

- OCIアプリケーション
    - 変更 7.7.4.2
    - 新機能を使用するための変更 7.7.4.2
    - 動的リンク 7.4.2.4
    - 静的リンク 7.4.2.3
    - アップグレードおよびリンク 7.4.2.2
    - アップグレード 7.5
    - アップグレードのオプション 7.7
- OCR 10.8.2.2
- OFA 1.4, 1.4.3
    - 「Optimal Flexible Architecture」も参照
- olspreupgrade.sql B.1
- OPatch lsinventoryコマンド 6.3.5
- オペレーティング・システム
    - データの移行 1.6.2
- オペレーティング・システム要件 1.6.1
- Optimal Flexible Architecture
    - 「OFA」を参照
    - 概要 1.4.3
- OPTIMIZER_ADAPTIVE_PLANS 10.8.1.1
- OPTIMIZER_ADAPTIVE_STATISTICS 10.8.1.1
- オプティマイザ統計
    - ダウングレード後の再収集 8.9.9
- OPTION OFFステータス 5.12.1
- プリコンパイラおよびOracle Call Interface (OCI)アプリケーションのアップグレード用のオプション 7.7
- ORA_TZFILE
    - ダウングレード後の設定解除 8.5
- ORA-00336 ログ・ファイルのサイズがxxxxブロックのエラー 5.1
- ORA-00401 パラメータcompatibleの値のエラー 5.1
- ORA-00600内部エラー 8.5
- ORA-00704: ブートストラップ・プロセス障害です。 5.1
- ORA-00904 「TZ_VERSION」無効な識別子のエラー 5.5
- ORA-00942 表またはビューが存在しないエラー 5.5
- ORA-01092: Oracleインスタンスが終了しました。強制的に切断されます 5.1
- ORA-01562 ロールバック・セグメント番号の拡張に失敗したエラー 5.6

- ORA-01650: ロールバック・セグメントを拡張できない 5.6
- ORA-01651: UNDOセグメント保存を拡張できない 5.6
- ORA-01652: 一時セグメントを拡張できない 5.6
- ORA-01653: 表を拡張できない 5.6
- ORA-01654: 索引を拡張できない 5.6
- ORA-01655: クラスタを拡張できない 5.6
- ORA-01722 無効な数値のエラー 5.5
- ORA-03134: このバージョンのサーバーへの接続は、サポートされていません。 10.8.1.10
- ORA-04031 共有メモリーのnnnバイトを割当てできないエラー 5.6
- ORA-1017 無効なユーザー名/パスワード 10.8.1.10
- ORA-17500: ODMエラー:操作は許可されていません 10.9.1.10
- ORA-19815 警告db_recovery_file_dest_sizeのエラー 5.6
- ORA-20001: ダウングレードを続行できません 8.10.1
- 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たエラー B.2.8.1
- 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 6.3.18, B.2.8.1
- ORA-27248: sys.dra_reevaluate_open_failuresが実行されています 5.16
- ORA-28040: 一致する認証プロトコルがありません 6.3.20
- ORA-28040 "一致する認証プロトコルがありません 2.5.5 2.5.6
- ORA-28040 一致する認証プロトコルがありません 10.8.1.10
- ORA-28365 5.9
- ORA-29283: ファイル・オプションが無効です 10.7.1.7
- ORA-39700: データベースは、UPGRADEオプションを使用してオープンしてください 5.1
- ORA-39701 データベースをEXCLUSIVEでマウントするように要求するエラー 5.1
- ORA-39709: コンポーネントのダウングレードが不完全です 8.6.2, 8.7
- ORA-39709: コンポーネントのダウングレードが不完全です; stringのダウングレードは中断されました 8.10.1
- oracle.dbc.OracleConnection
    - 非推奨 10.8.4.9
- oracle.jdbc.rowset
    - 非推奨 10.8.4.10
- Oracle ACFS暗号化
    - Oracle SolarisおよびMicrosoft Windowsで非推奨 10.6.4.16
- Oracle ACFSレプリケーション・プロトコルREPV1
    - 非推奨 10.6.4.14
- Oracle Administration Assistantツール 10.7.2.1
- Oracle Application Express
    - apexrelod.sqlファイル 8.5
- Oracle ASM
    - Oracleホームの場所の変更 6.4.8
- Oracle Automatic Storage Management。
    - ディスク・グループの互換性 6.6.3
    - パスワード・ファイル(PWFILE) 5.14
    - 優先読取りの障害グループ 6.6.4

    - ローリング・アップグレード 1.5.1
- Oracleベース 1.4.3
- Oracle Call Interface(OCI
    - アプリケーションのアップグレード 2.10.2
- Oracleチェンジ・データ・キャプチャ
    - 「Oracle GoldenGate」を参照
- Oracle Cluster Registry(OCR)
    - 手動アップグレード 6.8.4
- Oracle Databaseクライアント
    - バックアップの制限事項 10.8.1.12
- Oracle Database Enterprise Edition
    - Enterprise EditionからStandard Editionへの変換 1.5.5
- Oracle Database Express Edition 1.5.6
    - アップグレード後の推奨作業 6.7
    - Oracle Databaseへのアップグレード 1.5.6
- Oracle Database Standard Edition
    - Enterprise Editionへの変換 1.5.4
- Oracle Database Vault 2.3, 2.4
    - アップグレード後の有効化 6.3.19.1
    - アップグレード 6.3.19
- Oracle Databaseのビューによる確認 8.9.10
- Oracle Database XE 1.5.6
    - Oracle Databaseへのアップグレード 1.5.6
- Oracle Data Guard
    - AutoUpgrade 3.10.5.1, 3.10.6.1
    - プロパティ格納の変更 10.6.1.2
    - ローリング・アップグレード 1.5.1
    - DBUAを使用したローリング・アップグレード 2.6
- Oracle Data Mining
    - モデル・ディテール・ビュー 10.7.5.3
- Oracle Data Provider for .NET。
    - Oracle.ManagedDataAccessDTC.dllのサポート終了 10.7.2.5
- Oracle Data Pump 1.1.2
- Oracle Enterprise Manager Cloud Control
    - アップグレード 1.5.2
- Oracleフリート・パッチ適用およびプロビジョニング 10.8.1.11
- Oracle FPP 10.8.1.11
- Oracle GoldenGate 10.9.2.8
    - アップグレード 1.5.3
- Oracle Grid Infrastructure
    - OCRと投票ディスクのファイルの場所 10.8.2.2
- Oracleホーム
    - 構成ファイルのコピー 2.7

    - 複数 1.4.2
    - Microsoft WindowsでのORACLE_HOMEデータベース・ディレクトリ 2.7
    - LinuxまたはUNIXでのORACLE_HOME dbsデータベース・ディレクトリ 2.7
    - アウトオブプレース要件 2.1.5
- Oracleホーム・ユーザー 4.3.2
- Oracle Label Security 2.3, 2.4
- Oracleレイヤー・ファイル・システム 10.8.1.11
- Oracle Multimedia Java API
    - 非推奨 10.8.4.12
- Oracle Multitenant
    - アップグレード・エラー 5.15
- Oracle Multitenantのアップグレード 4.2.1, 4.2.7.1
- Oracle Namesのサポート 10.9.2.16
- Oracle Netコンフィギュレーション・アシスタント 2.5.3
- Oracle Optimal Flexible Architecture
    - 「Optimal Flexible Architecture」を参照
- Oracleオプティマイザ
    - DBMS_STATS 6.4.4
- Oracle RAC
    - Oracle Database Standard Edition 2 (SE2)でのサポート終了 10.6.2.9
- Real Application Clusters 1.3.5
- Oracleのリリース番号 1.3.1
- Oracle RMAN
    - データベースのバックアップ 2.12
- Oracle Streams
    - 最終リリース 10.7.4
- オラクル社が提供するアカウント
    - パスワードの変更 6.8.1
- Oracle Text
    - MAIL_FILTER 10.7.5.8
    - アップグレード 6.8.3
    - 拡大されたトークン列 6.4.15
- Oracle Textが提供するナレッジ・ベース
    - アップグレード 6.3.15
- Oracle Universal Installer 1.1.2, 2.3, 2.4
- Oracleバッチ更新サイズ無効化 10.8.1.8
- Oracleウォレット
    - アップグレード 5.9
- ORADIM 4.3.2
    - ダウングレード 8.6.2
    - アップグレード 4.2.3, 4.6.5
- orapki
    - RSA 512キーおよび1024キー (非推奨) 10.5.5.14

- orapwSIDパスワード・ファイル 2.7
- オリジナルのエクスポート/インポート
  - データ・ポンプ・エクスポート/インポートとの対比 9.1
- OUI
  - 「Oracle Universal Installer」を参照
- アプトオブプレース・パッチ適用 2.1.5
- アウトオブプレース・アップグレード 2.1.5

P

- PARALLEL_ADAPTIVE_MULTI_USERの非推奨 10.8.5
- PARALLEL_AUTOMATIC_TUNINGのサポート終了 10.8.3
- PARALLEL_IO_CAP_ENABLEDのサポート終了 10.8.3
- PARALLEL_SERVER_INSTANCEのサポート終了 10.8.3
- PARALLEL_SERVERのサポート終了 10.8.3
- パラレル・アップグレード・ユーティリティ 4.1.2, 5.1
  - 概要 4.4
  - スキーマベースの表領域のアップグレード機能 2.5.9
  - 手動アップグレード 4.6.5
  - アップグレードの再実行 4.7.2
  - 再起動 4.7.1
  - 再開オプション 4.7.1
  - 指定したCDBでの実行 4.7.3.2
  - 「catctl.pl」、「dbupgrade」、「dbupgrade.cmd」も参照 4.4
  - 表領域のREAD ONLYへの設定 10.8.1.9
- パラメータ・ファイル
  - SGAに対する読取り/書込み権限 10.8.1.4
  - バックアップ 2.7
- パスワード
  - 10Gパスワード・バージョン、確認と再設定 6.4.6
  - 大/小文字区別 6.4.5
- パスワード検証機能 2.5.5
- パスワード・バージョン 2.5.6
- パッチ・セット・リリース
  - ダウングレード 8.1
- パッチ・セット 1.3.1
- パッチ・セット更新 2.5.1, 6.9
- 一時停止と続行 10.8.1.2
- PDB
  - COMPATIBILITYパラメータおよびCDB 1.3.3.3
- PDB$SEED
  - アップグレード中に1つのPDBとしてカウント 4.1.4
- PDB C.1

    - catupgrd.logファイル 4.7.4.1
    - ダウンロード 8.5
    - 非CDBの移動 4.6.2
    - Oracle Label Security 4.6.4
    - プラガブル・アップグレード 10.8.1.2
    - 優先度に基づくPDBのアップグレード 10.8.1.2
    - アップグレードの再実行 4.7.3
    - 失敗したフェーズからの再開 4.7.4
    - アップグレードの再開 4.7.4.2
    - リストを使用したアップグレード優先度の設定 4.2.5
    - アップグレード・エラー 5.15
    - アップグレード・シナリオ 4.2
    - アップグレード 10.8.1.2
    - アップグレード 4.2.1
    - 個別アップグレード 4.2.7.1
    - 優先度リストを使用したアップグレード 4.2.4
- CDBのアップグレード後のPDBのアップグレード 4.2.7.5
- パフォーマンス
    - 統合監査証跡 6.4.17.1
- フィジカル・スタンバイ・データベース
    - ローリング・アップグレード 1.5.1
- PL/SQLパッケージ
    - 確認 6.3.7
- 共有ストレージへの配置の非推奨 10.8.2.2
- プラガブル・データベース
    - 「PDB」を参照
- プラガブル・データベース
    - 統合監査の移行 6.4.14.2
- postupgrade_fixups.sql B.1, B.2.3, B.2.6
- アップグレード後の状態ツール
    - 警告 5.4
- アップグレード後の状態ツール 6.1
    - アップグレードの再実行 4.7.2
- プリコンパイラ・アプリケーション
    - 新機能を使用するための変更 7.7.4.1
- プリコンパイラ・アプリケーション
    - アップグレード 2.10.2
- プリコンパイラ
    - アプリケーション
        - 変更 7.7.4.1
        - アップグレードのオプション 7.7
    - アプリケーションのアップグレード 7.5
- 優先読取りの障害グループ

    - 設定 6.6.4
- 事前処理スクリプト
    - OLSおよびDV B.1
- preupgrade_fixups.sql 4.3.1, B.2.2, B.2.6
    - 概要 B.2.1
    - postupgrade_fixups.sql
        - 概要 B.2.1
- preupgrade.jar 4.2.3, B.1, B.2.7
    - 「アップグレード前情報ツール」を参照
    - PDB B.2.4
    - コマンド構文 B.2.5
    - ユーザー環境変数の設定 B.2.4
- preupgrade.log 4.3.1, B.2.6
- アップグレード前ディレクトリ B.2.1
- アップグレード前情報ツール B.2.1
    - 「preupgrade.jar」も参照
    - postupgrade_fixups.sql 6.4.2
    - コマンド構文 B.2.5
    - Windowsでの例 4.3.1
    - 警告および対処法 B.2.8
    - 操作 B.2.8
- アップグレード前のステップ 2.1.4.3
- preupgrd.jar 5.3
- preupgrd.sql 5.3
    - 12.2で廃止 B.1
- 優先度リスト 4.2.4
    - 概要 4.2.5
    - アップグレードのテスト 2.10.1
- PRKH-1014エラー 2.3, 2.4
- Pro*C/C++アプリケーション 2.1.6.5
- 事前修正 3.9.2.41
- PRODUCT_USER_PROFILE
    - 非推奨 10.7.5.13
- プロキシPDB
    - アップグレード 4.2.2
- PSU 2.5.1, 6.9

R

- 高速ホーム・プロビジョニング 4.2
    - アップグレード 1.1.2
- RAWデータ型
    - EXTENDED 10.9.1.7

- RAWデバイス
  - サポート終了 10.9.1.6, 10.9.2.1
  - OCFS
    - Windowsでのサポート終了 10.9.1.6
- RDBMS DSTパッチ B.2.8.4
- 読取り専用Oracleホーム 10.4.1.6, 10.5.1.1
- 読取り専用Oracleホーム
  - demoディレクトリの置換え 6.3.17
- 読取り専用表領域 2.5.9
- 無効なオブジェクトの再コンパイル 4.2.3, 4.6.5, 6.3.2, 8.6.2
- リカバリ・カタログ
  - アップグレード 6.3.11
- リリース番号 1.3.1
- リリース
  - 定義 1.3.1
  - 複数 1.4.4
  - アップグレード・パス 1.1.3
- リリース更新(Update) 2.5.1, 6.9
- リリース更新(Update, RU) 1.3.1
- リリース更新リビジョン (Revision, RUR) 1.3.1
- リリース更新リビジョン(Revision) 2.5.1, 6.9
- REMOVEDステータス 5.12.1
- replace Oracle XQuery関数(非推奨) 10.8.4.20, 10.9.3.37
- replace XQuery関数 10.8.4.20, 10.9.3.37
- アップグレードの再実行
  - マルチテナント・データベース・アーキテクチャ 4.7.3
- 予約語
  - 追加および適用 7.2
- アップグレードの再開 4.7.1
- ロールバック・セグメント
  - 自動UNDO管理への移行 6.4.12
- ローリング・アップグレード
  - Oracle Clusterware 1.5.1
  - フィジカル・スタンバイ・データベース 1.5.1
  - ローリング・アップグレード
    - SQL Applyおよびロジカル・スタンバイ・データベース 1.5.1
  - SQL Apply
    - ローリング・アップグレード 1.5.1
  - 方法の概要 1.5.1
- Active Data Guardを使用するローリング・アップグレード 1.5.2
- rootupgrade.shスクリプト 2.3, 2.4
- リンクのrpathオプション 7.4.2.4
- RPMベースのデータベース・インストール 10.7.1.4

- rpm –ivh 10.7.1.4
- 複数リリースのOracleの実行 1.4.4
- ランタイム・ライブラリの検索パス 7.4.2.4

S

- スキーマ専用アカウント 2.1.7, 7.6
- スクリプト
  - Oracleデータ・ディクショナリの状態の確認 6.2
  - ダウンロード 8.6.1, 8.6.2
  - 手動アップグレード 4.2.3, 4.6.5
- シームレスなパッチ適用 10.4.1.6, 10.5.1.1
- SEC_CASE_INSENSITIVE_LOGON 10.8.1.10
- セキュリティ
  - 大/小文字を区別するパスワード 6.4.5
  - ORA_CIS_PROFILE 2.5.7
  - ORA_STIG_PROFILE 2.5.7
- サーバー
  - 互換性規則 7.4.1
- サーバー・パラメータ・ファイル(SPFILE) 2.7
  - 移行 6.8.2
  - HARD互換のストレージを使用するシステムのアップグレード 6.8.5.1
- サービス
  - 移行 10.7.1.3
- 単一クライアント・アクセス名(SCAN) 6.5
- SP2-0152:「Oracleが正常に機能していない可能性があります。」 8.5
- SP2-1503: Oracle Call Interfaceを初期化できません 8.5
- SP2-1540「Oracle Databaseはエディション・セッションでは起動できません。」エラー 5.7
- アップグレード結果をログ・ファイルにスプール B.2.1
- SQL_92_SECURITYデフォルトの変更 10.8.1.1
- SQL/MM still image標準のサポート 10.8.4.14
- SQL*Plus
  - 製品レベルのセキュリティ(非推奨) 10.7.5.13
  - スクリプト
    - アップグレード 7.8
- SQLCollSchema属性(非推奨) 10.8.4.20, 10.9.3.37
- SQL実行計画 2.1.6.6.3, 2.1.6.6.3.4
- SQLJ
  - クライアント側のサポートのみ 10.8.2.7
  - サーバーでの非推奨 10.9.3.23
- SQL管理ベース(SMB) 2.1.6.6.3
  - カーソル・キャッシュ 2.1.6.6.3
- SQLNET.ALLOWED_LOGON_VERSION_SERVER 2.5.5, 2.5.6

- sqlnet.oraファイル
  - 圧縮 2.5.3
- sqlnet.oraの場所 6.3.4
- SQLパフォーマンス・アナライザ 2.1.6.6.2
- SQL計画ベースライン
  - 展開 2.1.6.6.3, 2.1.6.6.3.4
- SQL計画管理 2.1.6.6.3
  - SQL管理ベース 2.1.6.6.3
- SQL問合せ
  - テスト 2.1.6.6.3, 2.1.6.6.3.4
- SQLSchema属性(非推奨) 10.8.4.20, 10.9.3.37
- SQLワークロード 2.1.6.6.2
- srclang属性(非推奨) 10.8.4.20, 10.9.3.37
- ステージング表
  - 作成 2.1.6.6.3, 2.1.6.6.3.4
- Standard Edition
  - エクスポート・ユーティリティ 1.5.5
  - 初期データベース 5.13
- Standard Edition High Availability
  - ガイドライン 2.5.11.2
- 標準運用環境 10.8.1.11
- StandbyFileManagementの非推奨 10.6.4.3
- STARTUP UPGRADEコマンド 8.6.2
- 静的にリンクされたOracleクライアントサイド・ライブラリ・コード 7.4.2.3
- 統計表
  - アップグレード 6.3.13
- ステータス
  - NO SCRIPT 5.12.1
  - OPTION OFF 5.12.1
  - REMOVED 5.12.1
  - UPGRADED 5.12.1
- ステータスINVALID
  - コンポーネントのステータス 5.12.1
- storeVarrayAsTable属性(非推奨) 10.8.4.20, 10.9.3.37
- サポート
  - 「My Oracle Support」を参照
- サポート・ノート472937.1、インストールされたデータベース・コンポーネントに関する情報 B.2.8
- サポート・ノート753041.1、NON VALIDステータスのコンポーネントの診断方法 B.2.8
- サポート・ノートID 412160.1、RDBMS DSTパッチ B.2.8.4
- シンボリック・リンク
  - Oracle Universal Installer 7.4.2.4
- 構文チェック
  - アプリケーション・コード 7.7.2

- SYS.LINK$表 B.2.8.3
- システム・グローバル領域
    - 読取り/書込み権限 10.8.1.4

T

- 表領域
    - 読取り専用 2.5.9
- テスト
    - アップグレード用のアプリケーション 2.1.6.8, 6.4.18
    - 計画の開発 2.1.6
    - アップグレードの機能 2.1.6.3
    - アップグレードの高可用性 2.1.6.4
    - アップグレードの統合 2.1.6.5
    - アップグレードの最小機能 2.1.6.2
    - マルチテナント・アーキテクチャ・アップグレード 2.10.1
    - アップグレードのパフォーマンス 2.1.6.6
    - アップグレードされたテスト・データベース 2.1.6.8
    - データベース・リプレイの使用 2.1.6.6.1
    - 優先度リスト・エミュレーションの使用 2.10.1
    - アップグレードのボリューム/ロード・ストレス 2.1.6.7
- アップグレード処理のテスト 2.10
- テキスト初期化パラメータ・ファイル(PFILE) 2.7
- TIMESTAMP WITH TIMEZONEデータ型 B.2.8.4
- TIMESTAMP WITH TIME ZONEデータ型 B.2.8.4
- タイムゾーン・ファイル
    - 不一致の解決方法 B.2.8.4
    - ダウングレード後の設定解除 8.5
- TNS-03505: 名前の決定に失敗しました。 6.3.4
- tnsnames.ora
    - 手動アップグレード後の調整 6.8.5.2
- tnsnames.oraの場所 6.3.4
- トークンの制限事項 10.7.1.5
- トレーニング, 入手先 2.1.1
- translate属性(非推奨) 10.8.4.20, 10.9.3.37
- 透過的データ暗号化(TDE)
    - Oracleウォレットのアップグレード 5.9
- トランスポータブル・エクスポート/インポート 4.5
- trgowner_no_admndbtrg 4.3.1
- トラブルシューティング
    - 再開の使用 4.7.1
    - 認証プロトコルのエラー 2.5.5, 2.5.6
    - AutoUpgrade 3.4.1

    - Autoupgradeユーティリティ 3.13
    - 致命的なアップグレード失敗後の表領域の表示 2.5.9
    - catupgrd.sqlの非推奨エラー 10.9.1.1
    - CDB_JAVA_POLICYエラー 5.10
    - datapatchの失敗 5.17
    - EDITIONセッションのエラー 5.7
    - フラッシュ・リカバリ領域 5.6
    - INVALIDオブジェクト 5.3, 5.4
    - ORA-00942「表またはビューが存在しません。」 5.5
    - ORA-03134: このバージョンのサーバーへの接続は、サポートされていません。 10.8.1.10
    - ORA-1017 無効なユーザー名/パスワード 10.8.1.10
    - ORA-39709 8.7
    - ORA-45415 2.6
    - ORA-65394ランタイム・エラー 10.7.1.6
    - Oracle Internet Directory
        - 以前のリリースへのダウングレード C.1
    - PDBアップグレード 4.2.7.5
    - PLS-1919コンパイル時エラー 10.7.1.6
    - REMOTE_LOGIN_PASSWORDFILE警告 2.5.4
    - リストア・スクリプトとOracle Internet Directoryの登録 C.1
    - ロールバック・セグメント/UNDO表領域 5.6
    - リソースの不足 5.6
    - アップグレード後の古いOracleホームで実行されているサービス 10.7.1.3
    - 共有メモリー 5.6
    - アップグレード・モードでのデータベースの起動 5.1
    - SYSTEM表領域およびSYSAUX表領域 5.6
    - アップグレード 5
    - アップグレード中断
        - ORA-00904が原因 5.5
        - ORA-01722が原因 5.5
- トラブルシューティング
    - ORA-39709 8.6.2
- アップグレードのトラブルシューティング
    - ORA_00942による終了 5.5
- ソフトウェアのアップグレード・タイプ 7.3

U

- UNDO_MANAGEMENT初期化パラメータ 6.4.12
- unicode照合アルゴリズム6.1
    - 非推奨 10.8.4.15
- 統合監査
    - 「監査」を参照

    - アップグレード後の監査レコードの転送の概要 6.4.17.1
    - アップグレード後の統合監査レコードの転送 6.4.17.2
- 統合監査証跡
    - 監査レコードのロード先 2.5.12
    - パフォーマンスの向上 6.4.17.1
- upg_summary.rpt 5.12.3
- UPG-3009: プラグイン違反があります 10.4.1.1
- アップグレード
    - 手動アップグレードの例 4.3
- upgrade.xmlが見つからないエラー 5.2, D
- UPGRADEDステータス 5.12.1
- アップグレードされたテスト・データベース 2.1.6.8
- アップグレード方法
    - 選択 2.1.4
    - Database Upgrade Assistant 1.1.2
    - Database Upgrade Assistant (DBUA) 2.1.4.2, 2.1.4.5
    - データ・ポンプ・エクスポート/インポート 2.1.4.4
    - エミュレーション 2.10.1
    - 手動 2.1.4.3
    - サイレント・モード C.7
- アップグレード・パス
    - 判断 1.1.3
    - 表 1.1.3
- アップグレード手順
    - エラー・メッセージ 10.8.1.2
    - 主なステップの概要 1.2
- アップグレード処理のテスト 2.10
    - utlrp.sql 1.3.5
- アップグレード・サマリー・レポート
    - 場所 5.12.3
- アップグレード
    - アプリケーション 7
        - 互換性規則 7.4
        - オプション 7.7
        - 再リンク 7.4
    - 定義 1.1.1
    - 初期化パラメータ 2.7
    - new administrative procedures 6.4.10
    - Oracle Formsアプリケーション 7.9
    - ORADIM 4.2.3, 4.6.5
    - アップグレード後の処理 6
    - 準備 2.1
    - リカバリ・カタログ 6.3.11

    - スクリプトおよび手動アップグレード 4.2.3, 4.6.5
    - SQL*Plusスクリプト 7.8
    - 統計表 6.3.13
    - テスト 2.1.6
    - トラブルシューティング 5
    - Database Upgrade Assistantの使用 C
    - 情報の入手先 1.1.4
- クラスタ・データベースのアップグレード
    - CLUSTER_DATABASE初期化パラメータの設定 2.7
- UTL_FILE 10.7.1.7
- UTL_FILE_DIRの非推奨 10.8.5
- UTL_INADDRパッケージ B.2.8.1
- UTL_MAILパッケージ B.2.8.1
- UTL_SMTPパッケージ B.2.8.1
- UTL_TCPパッケージ B.2.8.1
- utlrp.sql 1.3.5, 6.3.2, B.1
- utlrp.sqlスクリプト
    - 無効なオブジェクトの再コンパイル 4.2.3, 4.6.5, 8.6.2
- utluptabdata.sql 10.8.1.9
- utlusts.sql 5.4, 5.12.1, 6.1, B.1

V

- V$OPTIONビュー 5.12.1
- V$REPLPROP
    - サポート終了 10.8.2.6
- V$REPLQUEUE
    - サポート終了 10.8.2.6
- VARCHAR2データ型
    - EXTENDED 10.9.1.7
- VERIFY_FUNCTION
    - 非推奨 10.8.4.17
- VERIFY_FUNCTION_11G
    - 非推奨 10.8.4.17
- ビュー
    - サポート終了 10.8.2.6
- ボリューム
    - アップグレードされるデータの量 2.1.6.7
- 投票ディスク・ファイル
    - 共有ストレージへの配置の非推奨 10.8.2.2

W

- 「警告: XDBは現在無効です」エラー 5.15
- Windows
  - 仮想アカウント 4.3.2
  - 手動アップグレード 4.3.1, 4.3.2
  - リモート・アップグレードの非推奨 10.8.1.2
  - ORADIMの使用 4.3.2
- ワークロード
  - 取得およびリプレイ 2.1.6.6.1

X

- xdb defaultTableSchema属性(非推奨) 10.8.4.20, 10.9.3.37
- xdb maintainOrder属性(非推奨) 10.8.4.20, 10.9.3.37
- xdb mapUnboundedStringToLob属性(非推奨) 10.8.4.20, 10.9.3.37
- xdb maxOccurs属性(非推奨) 10.8.4.20, 10.9.3.37
- xdb SQLCollSchema属性(非推奨) 10.8.4.20, 10.9.3.37
- xdb SQLSchema属性(非推奨) 10.8.4.20, 10.9.3.37
- xdb srclang属性(非推奨) 10.8.4.20, 10.9.3.37
- xdb storeVarrayAsTable属性(非推奨) 10.8.4.20, 10.9.3.37
- xdb translate属性(非推奨) 10.8.4.20, 10.9.3.37
- XE 1.5.6
- XML DB
  - サポートが終了した関数とプロシージャ 10.8.2.8
- XQuery言語
  - 関数 10.8.4.20, 10.9.3.37
    - matches 10.8.4.20, 10.9.3.37
    - matches (非推奨, Oracle) 10.8.4.20, 10.9.3.37
    - replace (非推奨, Oracle) 10.8.4.20, 10.9.3.37

Z

- Zero Data Loss Recovery Appliance
  - バックアップの制限事項 10.8.1.12
- 停止時間ゼロのアップグレード 4.2