



EDB

Migration Portal

リリース 4.5.1

EnterpriseDB Corporation

2023 年 07 月 27 日

目次

第 1 章	Release notes	3
1.1	Version 4.5.1	3
1.1.1	サポート終了のお知らせ	3
1.2	Version 4.5	4
1.2.1	サポート終了のお知らせ	5
1.3	Version 4.4	5
1.3.1	サポート終了のお知らせ	7
1.4	Version 4.3	7
1.5	Version 4.2	8
1.6	Version 4.1	9
1.7	Version 4.0	11
第 2 章	Supported platforms	13
2.1	サポートされているブラウザ	13
第 3 章	Using Migration Portal	15
3.1	Overview of the Migration Portal home page	16
3.2	Overview of the Migration Portal Projects page	17
3.3	Migration Portal quick start	18
第 4 章	Migrating a database	21
4.1	Performing a schema extraction	21
4.1.1	EDB DDL エクストラクタを使用したスキーマの抽出	21
4.1.2	スキーマとオブジェクトのサポート	24
4.2	Performing a schema assessment	29
4.2.1	スキーマ評価を実行する	30
4.2.2	評価レポートを生成する	33
4.2.3	評価レポートを評価する	34
4.3	Performing a schema migration	35
4.3.1	オフライン移行	36
4.3.2	オンライン移行	38
4.4	Performing a data migration	40
4.4.1	高度なデータ移行	40
第 5 章	Known issues, limitations, and notes	41
5.1	Migration Portal	41
5.1.1	ラップされたオブジェクト	41
5.1.2	ファイルのエンコード	41
5.1.3	ALTER ステートメント	42
5.1.4	ユーザー、ロール、プロファイル、および付与	42
5.1.5	付与	42

5.1.6	特権	42
5.1.7	パスワード	43
5.1.8	プロフィール	43
5.1.9	Oracle のデフォルトのケースを使用する	43
5.2	EDB DDL エクストラクタ	47
5.3	Oracle データ ポンプ ユーティリティ	47

Migration Portal は、Oracle データベース スキーマを EDB Postgres Advanced Server プラットフォームに移行するための Web ベースのサービスです。Migration Portal は、Oracle データベーススキーマを評価および分析し、以下を変換します。

- テーブル
- シーケンス
- 制約
- インデックス (LOB インデックスとマテリアライズドビューのインデックスを除く)
- 同義語
- ビュー
- マテリアライズドビュー
- DB リンク
- 型と型本体
- トリガー
- 機能
- 手順
- パッケージ
- ユーザー
- 役割
- プロフィール
- 役割とオブジェクトの付与

Migration Portal は、EDB Postgres Advanced Server と互換性のあるデータ定義言語 (DDL) ステートメントを生成します。

ユーザーフレンドリーなポータルインターフェイスにより、評価と移行のプロセスが簡素化されます。ポータルにログインし、

migration process を開始します。

EnterpriseDB は、企業が既存のデータベース システムを Postgres に移行するのを長年にわたって支援してきました。詳細については、

[EnterpriseDB website](#) を参照してください。

第1章 Release notes

Migration Portal のドキュメントでは、Migration Portal 4.0 の最新バージョンについて説明しています。

Version	Release date
<i>Version 4.5.1</i>	12 Jul 2023
<i>Version 4.5.1</i>	18 May 2023
<i>Version 4.4</i>	14 Feb 2023
<i>Version 4.3</i>	08 Dec 2022
<i>Version 4.2</i>	04 Oct 2022
<i>Version 4.1</i>	04 Aug 2022
<i>Version 4.0</i>	22 Apr 2022

1.1 Version 4.5.1

Migration Portal 4.5.1 の新機能、機能拡張、バグ修正、その他の変更点は次のとおりです。

Type	Description
Bug fix	何千ものテーブルを含む大規模なスクリプトを処理するときにメモリの過剰使用が原因で評価フェーズが停止する問題を修正しました。
Bug fix	[オブジェクト名を一意にするためにサフィックスを追加] オプションが選択されており、データベース オブジェクト名に \$ 文字が含まれている場合に評価が失敗する問題を修正しました。

1.1.1 サポート終了のお知らせ

EDB Postgres Advanced Server バージョン 11 の標準サポートの終了は 2023 年 11 月 20 日に予定されているため、新しい Migration Portal プロジェクトでは 2023 年以降もサポートされる EDB Postgres Advanced Server のより新しいバージョンを選択してください。EDB Postgres Advanced Server 11 のサポートは、2023 年 5 月 20 日に Migration Portal から削除され始める予定で、この日以降に新しい EDB Postgres Advanced Server 11 プロジェクトが作成されないようにする予定です。

利用可能なサポート期間を最大限に延長し、新機能、特に Oracle との互換性を追加する機能を活用するには、最新の EDB Postgres Advanced Server バージョンを Migration Portal プロジェクトのターゲットとして選択することをお勧めします。EDB Postgres Advanced Server の各メジャーバージョンのサポート期間の詳細については、EDB [Platform Compatibility](#) ページを参照してください。

1.2 Version 4.5

Migration Portal 4.5 の新機能、機能拡張、バグ修正、およびその他の変更点は次のとおりです。

Type	Description
Enhancement	移行ポータルに移行の複雑さと機能の分析機能は、以前はキャプチャされていなかった次の Oracle 機能の存在をキャプチャしてレポートできるようになりました: <i>SELECT FOR UPDATE OF</i> 、 <i>ANY with = <> != operators</i> 、 <i>ANY with > < >= <= operators</i> 、 <i>IS JSON CHECK CONSTRAINT</i> 、 <i>LISTAGG ON OVERFLOW</i> 、 <i>UNDER IN TYPE DEFINITION</i> 、 <i>REF IN TYPE SPECIFICATIONS</i> 、 <i>OVERRIDING SUBPROGRAM DECLARATION IN TYPE</i> 、および <i>OF IN OBJECT VIEW CLAUSE</i> 。
Enhancement	EDB DDL エクストラクタユーティリティでは、スキーマの抽出時に必要な場合にのみ <i>GRANT</i> ステートメントを抽出するオプションが提供されるようになりました。以前は、 <i>GRANT</i> ステートメントは常に抽出されていました。これで、それらを抽出することを選択できるようになりました。デフォルトでは、 <i>GRANT</i> ステートメントは抽出されません。
Enhancement	Migration Portal は、プロジェクトが通常サポートの終了に近づいている EDB Postgres Advanced Server データベース バージョンをターゲットにしていることを示し、ターゲット データベースのサポートが Migration Portal から削除されると、プロジェクトへのさらなる更新を制限するようになりました。
Enhancement	Migration Portal では、プロジェクトの作成日と、プロジェクトの作成時に使用された Migration Portal のバージョンが、Migration Portal ユーザー インターフェイスの 概要 セクションに表示されるようになりました。
Enhancement	Migration Portal は、ターゲット データベース バージョン 11 ~ 14 と互換性のない関数として <i>SQLERRM()</i> 関数を検出するようになりました。
Enhancement	Migration Portal には、 <i>= ANY</i> オペレーターの使用を <i>IN</i> に、 <i>!= ANY</i> オペレーターの使用を <i>NOT IN</i> に置き換える新しい修復ハンドラー (ERH-2103) が追加されました。
Enhancement	修復ハンドラー ERH-2100 は、EDB Postgres Advanced Server でサポートされていない <i>WRITE ON DIRECTORY</i> 付与権限を検出して削除するようになりました。
Bug fix	<i>IMPORT FULL</i> や <i>EXPORT FULL</i> などの一部のサポートされていないシステム許可が Migration Portal によって削除されない問題を修正しました。
Bug fix	<i>DBMS_UTILITY.FORMAT_CALL_STACK</i> 関数と <i>DBMS_UTILITY.LNAME_ARRAY</i> 関数が EDB Postgres Advanced Server でサポートされている場合に、互換性がないとして誤ってフラグが立てられる問題を修正しました。
Bug fix	別のロールに割り当てられているロールに対して <i>CREATE ROLE</i> ステートメントが抽出されない問題を修正しました。
Bug fix	プロジェクトのエクスポート SQL ファイルから不要な <i>SET SEARCH_PATH=GLOBAL_OBJECTS</i> ステートメントを削除しました。
Bug fix	特定の使用例でファイル解析フェーズ中に Migration Portal の評価プロセスでエラーが発生する原因となっていた問題を修正しました。

次のページに続く

表 3 – 前のページからの続き

Type	Description
Bug fix	Oracle Data Pump ユーティリティによって抽出され、アップロードされたソース DDL に含まれているネストしたテーブルが解析エラーを引き起こす問題を修正しました。EDB Postgres Advanced Server は、ネストされたテーブルをサポートしません。その結果、それらは Migration Portal による処理と評価の前にソース DDL から削除されます。
Security fix	Migration Portal で使用される <code>ejs</code> ライブラリ (Node.js 用) は、 CVE-2022-29078 および CVE-2021-23566 で特定されたセキュリティの脆弱性に対処するために更新されました。
Security fix	Migration Portal で使用される <code>terser</code> (NPM) ライブラリは、 CVE-2022-25858 で特定されたセキュリティの脆弱性に対処するために更新されました。

1.2.1 サポート終了のお知らせ

EDB Postgres Advanced Server (EPAS) バージョン 11 の標準サポートの終了は 2023 年 11 月 20 日に予定されているため、新しい Migration Portal プロジェクトでは 2023 年以降もサポートされる EPAS のより新しいバージョンを選択してください。EPAS 11 のサポートは、この日以降に新しい EPAS 11 プロジェクトが作成されないようにするため、2023 年 5 月 20 日に Migration Portal から削除され始める予定です。

利用可能なサポート期間を最大限に延長し、新機能、特に Oracle との互換性を高める機能を活用するには、最新の EPAS バージョンを Migration Portal プロジェクトのターゲットとして選択することをお勧めします。EDB Postgres Advanced Server の各メジャーバージョンのサポート期間の詳細については、EDB [Platform Compatibility](#) ページを参照してください。

1.3 Version 4.4

Migration Portal 4.4 の新機能、機能拡張、バグ修正、その他の変更点は次のとおりです。

Type	Description
Enhancement	Migration Portal は、EDB Postgres Advanced Server EDB Postgres Advanced Server バージョン 15 をサポートするようになりました。

次のページに続く

表 4 – 前のページからの続き

Type	Description
Enhancement	Migration Portal が強化され、移行プロジェクト内のオブジェクト名に Oracle のデフォルトの大文字と小文字を保存するオプションが提供されるようになりました。Postgres では、オブジェクト名のデフォルトの大文字は小文字です。Oracle では大文字です。ほとんどの場合、データベースを Postgres に移行する場合は、Postgres のデフォルトのケースを使用する必要があります。ただし、この新しいオプションは、データベース クエリで引用符で囲まれた大文字の識別子を特に使用するアプリケーションをより適切にサポートするために提供されています。新しい機能により、すべてのオブジェクトを引用符付きの大文字名で作成できるようになりますが、最初の実装ではプロシージャコードで指定されたオブジェクト名の参照が更新されないことに注意してください。
Enhancement	Migration Portal の移行の複雑さと機能の分析機能が強化され、外部結合の条件としての <i>BETWEEN</i> 句の使用に対応する <i>OUTER JOIN WITH BETWEEN</i> 機能など、これまでキャプチャされていなかった特定の Oracle 機能の存在をキャプチャしてレポートできるようになりました。クエリで。
Enhancement	Migration Portal は、EDB Postgres Advanced Server ではこの機能がサポートされていないため、 <i>CREATE VIEW</i> ステートメントから <i>CONSTRAINT</i> 句を削除する新しい ERH-3008 修復ハンドラーで強化されました。この機能強化の一環として、スキーマ評価レポートの機能分析セクションに関連する機能が更新され、 <i>VIEW COLUMN CONSTRAINT</i> の発生を「既存の利用可能な回避策を持つ機能」の分類で識別してレポートするようになりました。
Enhancement	ERH-2057 修復ハンドラーが拡張され、 <i>LONG</i> データ型から精度部分が削除されました。
Bug fix	EDB DDL エクストラクタ が更新され、以前はフェッチされていなかった <i>ROLE</i> から <i>ROLE</i> 許可を抽出するようになりました。
Bug fix	日付リテラルの指定に使用される場合、 <i>VIEW</i> の DDL が <i>DATE</i> キーワードの後ろで切り捨てられる問題を修正しました。
Bug fix	列名として <i>EXTENDED</i> を指定したテーブル列の <i>DEFAULT</i> 句が削除される問題を修正しました。
Bug fix	ユーザーが手動でオブジェクトを修復した後、評価レポートに誤ったオブジェクト数が表示される問題を修正しました。
Bug fix	Oracle <i>PIPE ROW</i> 構文から Postgres <i>RETURN NEXT</i> 構文に変換した後、スキーマ名プレフィックスで指定された行タイプ オブジェクトからスキーマ名が誤って削除されるという、ERH-2059 修復ハンドラーの問題を修正しました。
Other	EDB Postgres Advanced Server ではサポートされていない、単一の引数を使用した <i>TO_TIMESTAMP_TZ()</i> 関数呼び出しを移行するための回避策案を記載した Migration Portal のナレッジ ベース エントリを追加しました。

次のページに続く

表 4 – 前のページからの続き

Type	Description
Other	EDB Postgres Advanced Server バージョン 10 のサポートは削除されました。ユーザーは、新しい移行プロジェクトのターゲット データベースとして EDB Postgres Advanced Server 10 を選択できなくなりました。また、ユーザーは新しいスキーマをアップロードしたり、既存の EDB Postgres Advanced Server 10 プロジェクト内のオブジェクトを編集および再評価したりすることもできなくなります。

1.3.1 サポート終了のお知らせ

EDB Postgres Advanced Server (EPAS) バージョン 11 の標準サポートの終了は 2023 年 11 月 20 日に予定されているため、新しい Migration Portal プロジェクトには、2023 年以降もサポートされる EPAS のより新しいバージョンを選択する必要があります。EPAS 11 のサポートは、2023 年 5 月 20 日に Migration Portal から削除され始める予定で、この日以降に新しい EPAS 11 プロジェクトが作成されないようにする予定です。

利用可能なサポート期間を最大限に延長し、新機能、特に Oracle との互換性を追加する機能を活用するために、EDB は、Migration Portal プロジェクトのターゲットとして最新の EPAS バージョンを選択することをお勧めします。EDB Postgres Advanced Server の各メジャー バージョンのサポート期間の詳細については、EDB

[Platform Compatibility](#) ページを参照してください。

1.4 Version 4.3

Migration Portal 4.3 の新機能、機能拡張、バグ修正、その他の変更点は次のとおりです。

Type	Description
Enhancement	Migration Portal は、ソース Oracle データベースからのユーザー、ロール、プロファイル、および権限の評価と移行をサポートするように拡張されました。
Enhancement	Migration Portal が更新され、ユーザー、ロール、プロファイル、権限の抽出をサポートする EDB DDL エクストラクタの新しいバージョンへのリンクが提供されました。Migration Portal は、古いバージョンの EDB DDL エクストラクタによって生成された DDL を引き続き処理および評価できます。ただし、新しくサポートされたユーザー関連情報は、それらの古いバージョンでは抽出できません。
Enhancement	Migration Portal のスキーマ移行複雑さのスコアリングおよび機能分析機能が強化され、以前のリリースの場合のようにバージョン 14 だけでなく、選択したターゲット EDB Postgres Advanced Server バージョンに基づくスコアおよび機能分析が提供されるようになりました。

次のページに続く

表 5 – 前のページからの続き

Type	Description
Enhancement	スキーマをターゲット データベースに移行するための Migration Portal ユーザー インターフェイスが簡素化され、ステップ数と冗長オプションが減り、オフラインおよびオンライン移行モードが有効になりました。
Enhancement	Migration Portal の移行の複雑さと機能の分析機能が強化され、 <i>CONNECT BY CONDITION</i> 、 <i>CONNECT BY NOCYCLE</i> 、および組み込み/SYS 所有関数 <i>GROUP_ID</i> など、これまでキャプチャされていなかった Oracle 機能の存在をキャプチャしてレポートできるようになりました。
Security fix	Migration Portal で使用される jackson-databind ライブラリは、 CVE-2022-42003 および CVE-2022-42004 で特定されたセキュリティの脆弱性に対処するために更新されました。
Bug fix	警告を回避し、ターゲット EDB Postgres Advanced Server データベースでサポートされる <i>MAXVALUE</i> に許可される最大値に関する問題を解決するために、シーケンスの <i>MAXVALUE</i> の処理を修正しました。
Bug fix	Oracle Data Pump ユーティリティを使用して生成された抽出された DDL にある仮想表列に関連する `ALTER TABLE MODIFY` 文の処理の問題を修正しました。
Bug fix	<i>PIPELINED</i> 句の機能がパッケージ関数定義内で句が使用されている場合に報告されない問題を修正しました。
Bug fix	特定の場合に互換性のある機能でカタログ ビューが検出されず、レポートされないことに関連する問題を修正しました。

1.5 Version 4.2

Migration Portal 4.2 の新機能、機能拡張、バグ修正、その他の変更点は次のとおりです。

Type	Description
Enhancement	スキーマ移行の複雑さスコアを計算し、4.1.0 リリースより前に作成された既存のプロジェクトの機能分析を生成するためのサポートが追加されました。
Enhancement	オブジェクトを手動で修復した後のスコアの更新に対応できるように、スキーマ移行の複雑さのスコアリング機能が強化されました。
Enhancement	スキーマ評価レポートの機能分析セクションに、未解決のエラーのあるオブジェクトと、それらのオブジェクトで使用されている互換性のない Oracle 機能のリストをリストする新しい「未解決のオブジェクト」テーブルが追加されました。
Enhancement	多数の Oracle 機能を含むプロジェクトのスキーマ評価レポート生成のパフォーマンスが最大 3 倍向上しました。
Bug Fix	互換性のない組み込みパッケージと SYS スキーマ TYPE オブジェクトの使用をキャプチャし、より正確にレポートするように Migration Portal を修正しました。

次のページに続く

表 6 – 前のページからの続き

Type	Description
Bug Fix	次の Oracle 機能の存在をより正確に把握し、それらがスキーマ移行の複雑さに及ぼす影響を評価するように Migration Portal を修正しました: USER DEFINED EXCEPTION RISING、USER DEFINED EXCEPTION HANDLING、XMLAGG、XMLQUERY、XMLTABLE、XMLPI、XMLPARSE、および XML-ROOT。
Bug Fix	CACHE キーワードで定義された Oracle テーブルの重複エントリが DDL ファイルに出力される問題を修正するために、新しいバージョンの DDL エクストラクタを作成しました。Migration Portal が更新され、新しいバージョンの DDL エクストラクタへのリンクが提供されるようになりました。Migration Portal は、古いバージョンの DDL エクストラクタによって生成された DDL を引き続き処理および評価できます。
Bug Fix	スキーマ評価レポートの「機能分析」セクションで、PRAGMA EXCEPTION_INIT 宣言が互換性のない機能として誤って報告される問題を修正しました。
Bug Fix	LONG RAW が「機能分析」セクションで互換性のない機能として誤って報告される問題を修正しました。
Bug Fix	ユーザー定義の例外宣言が、スキーマ評価レポートの「機能分析」セクションで互換性のない機能として誤って報告される問題を修正しました。
Bug Fix	スキーマ評価レポートの「機能分析」セクションで BULK COLLECT INTO 句が互換性のない機能として誤って報告される問題を修正しました。
Bug Fix	スキーマ評価レポートの「機能分析」セクションで NULLIF 関数が互換性のない機能として誤って報告される問題を修正しました。
Bug Fix	スキーマ評価レポートの「機能分析」セクションで、REGEXP_LIKE 句の回避策ステータスを「なし / 調査が必要」から「存在する」に更新しました。現在、修復ハンドラーが利用可能であり、REGEXP_LIKE 句に自動的に適用されます。
Bug Fix	EDB Postgres Advanced Server でサポートされる Oracle 事前定義 PL/SQL 例外とサポートされない Oracle 事前定義 PL/SQL 例外のリストを更新しました。
Bug Fix	ユーザーがオブジェクトのサブセット (スキーマ、テーブル、ビューなど) を選択し、ウィザード ページ間を移動すると、ターゲット DDL オブジェクト選択ページのユーザー選択状態が失われる問題を修正しました。

1.6 Version 4.1

Migration Portal 4.1 の新機能、機能拡張、バグ修正、その他の変更点は次のとおりです。

Type	Description
Enhancement	Migration Portal のスキーマ評価レポートには、アップロードされた DDL で見つかった Oracle 構造 (つまり、機能) に関連する作業タイプと複雑さのレベルに基づいて、移行の複雑さのスコアを提供する新しい機能分析セクションが含まれるようになりました。この新しいセクションでは、DDL で見つかった互換性のない機能の概要も提供します。現在、「機能分析」セクションは新しい移行プロジェクトに対してのみ生成され、このリリースより前に作成された既存のプロジェクトに対しては生成されないことに注意してください。
Enhancement	ユーザーは、プロジェクト内の使用可能なスキーマと基礎となるオブジェクトにフィルタリングを適用して、選択したオブジェクト タイプのサブセットのターゲット データベース DDL を生成できるようになりました。さらに、ダウンロード用に単一の DDL SQL ファイルが作成されるのではなく、選択したオブジェクト タイプごとに個別のファイルと、プロジェクトおよび選択したスキーマの統合スクリプトが含まれる zip アーカイブが作成されます。
Enhancement	Q で囲まれた文字列をドルで囲まれた文字列または E で囲まれた文字列に変換するための新しい修復ハンドラー ERH-3005 が追加されました。
Enhancement	ERH-2078 修復ハンドラーが強化され、パイプライン化された特定の変換をパッケージ ヘッダー レベルで適用できるようになりました。
Enhancement	ERH-2034 修復ハンドラーを更新し、PACKAGE 内で提供される TYPE 定義から NULL を削除するようにしました。
Security fix	セキュリティ問題 (CVE-2021-43138 、 CVE-2022-1650 、 CVE-2022-23221 、 CVE-2021-42392 、 CVE-2021-20066 、 および CVE-2022-31129) に対処するために、async、eventsources、h2、jsdom、および moment ライブラリを最新バージョンにアップグレードしました。
Bug fix	Oracle Data Pump ユーティリティを使用して生成されたソース Oracle DDL ファイルを処理するときに、一部のインデックスがスキップされる特殊なケースを修正しました。
Bug fix	ALTER TABLE ステートメントの ENABLE NOVALIDATE 句から ENABLE キーワードのみが削除される動作が修正され、完全な句が削除されるようになりました。また、関連するリペア ハンドラー ERH-2003 および ERH-2005 を更新しました。
Bug fix	長時間実行されるスキーマ評価と並行して実行される他の評価のパフォーマンスに影響を与えていた同期スレッドの問題を修正しました。
Bug fix	Oracle Data Pump ユーティリティによって生成される CREATE TABLE 文の DEDUPLICATE LOB 句によって DDL 解析関連のエラーが発生する問題が修正されました。

1.7 Version 4.0

Migration Portal 4.0 の新機能、機能拡張、バグ修正、その他の変更点は次のとおりです。

Type	Description
Enhancement	Migration Portal では、EDB DDL エクストラクタの使用が再びサポートされるようになりました。DDL エクストラクタに関連する問題は解決され、DDL エクストラクタファイル形式の使用に関して導入された一時的な制限が削除されました。Migration Portal は、 documentation の指示に従って Oracle Data Pump ユーティリティの使用を引き続きサポートします。ただし、EDB DDL エクストラクタの使用をお勧めします。
Enhancement	スキーマ評価アルゴリズムは、いくつかのパフォーマンス関連の問題に対処し、全体的なスキーマ評価時間を短縮するために最適化されています。
Enhancement	EDB Postgres Advanced Server が CACHE/NOCACHE キーワードをサポートしていないため、TABLE 定義から CACHE/NOCACHE キーワードを削除する修復ハンドラーを追加しました。
Security fix	jackson-databind ライブラリと h2 ライブラリは、セキュリティ問題 (CVE-2020-36518、CVE-2022-23221、CVE-2021-42392、および CVE-2021-23463) に対処するために最新バージョンにアップグレードされます。
Bug fix	EDITIONABLE キーワードを削除する修復ハンドラーの実行をスキップする解析エラーを解決する文法ルールを修正しました。
Bug fix	Oracle PIPELINED および PIPE ROW 関数呼び出しの適切な変換を実行するように修復ハンドラーを修正しました。
Bug fix	オブジェクト名が Unicode 文字に基づいている場合に大文字と小文字が区別されなくなる問題を修正しました。
Bug fix	新しいユーザーが初めて Migration Portal のランディング ページを開いたときに、開始ウィザードが表示される問題を修正しました。
Bug fix	ターゲット EPAS DDL スクリプトでの変換中に VIEW 定義が切り捨てられる問題を修正しました。
Bug fix	以前は、UTF-8 と互換性のない文字を含む DDL ファイルをアップロードすると、評価フェーズの開始時に失敗していました。Migration Portal は SQL ファイルを UTF-8 として読み取るようになりました。UTF-8 以外のファイルがアップロードされた場合、UTF-8 と互換性のないすべての文字は置換文字「☒」に変換され、評価は失敗せずに続行されます。
Bug fix	ソース Oracle DDL の名前のない制約に対する制約名の割り当てが修正されました。
Bug fix	Oracle Data Pump で生成されたソース DDL ファイルの OIDINDEX ストレージ句を削除するために修復ハンドラーを更新しました。
Bug fix	COLLATE キーワードと COMPUTATION キーワードを識別子として使用できるようになりました。
Bug fix	単一行コメントの一部として表示される複数行コメントの処理を修正しました。
Bug fix	ラップされたオブジェクトを適切に識別してスキップするように文法を修正しました。

次のページに続く

表 8 – 前のページからの続き

Type	Description
Bug fix	複数のプロジェクトを同時に実行中に発生する <code>ConcurrentModificationException</code> を修正しました。
Bug fix	複数のスキーマにわたる一意のオブジェクト名の処理を修正しました。オブジェクト名は、異なるスキーマ間で一意である必要はありません。
Bug fix	特定の順序は必要ないため、一部の句の順序を無視するようにテーブル作成文法をいくつか修正しました。

第2章 Supported platforms

Migration Portal は、Oracle 11g、12c、18c、および 19c から EDB Postgres Advanced Server 11、12、13、または 14 への評価と移行をサポートします。Migration Portal は、次のブラウザでサポートされます。

2.1 サポートされているブラウザ

最高のユーザーエクスペリエンスを実現するには、Google Chrome ブラウザの使用をお勧めします。Migration Portal は次のブラウザでもサポートされています。

Browser	Supported version
Apple Safari on Macintosh OS	14 and above
Google Chrome	68 and above
Microsoft Edge	42 and above
Mozilla Firefox	60 and above

第3章 Using Migration Portal

Migration Portal と、データベース スキーマを Oracle から EDB Postgres Advanced Server に簡単に移行できます。評価用にスキーマをアップロードすると、フィードバックや提案がすぐに得られます。ポータルを使用すると、すべてのオブジェクトの評価済み DDL をダウンロードし、オンプレミスまたはクラウドに EDB Postgres データベースを作成できます。

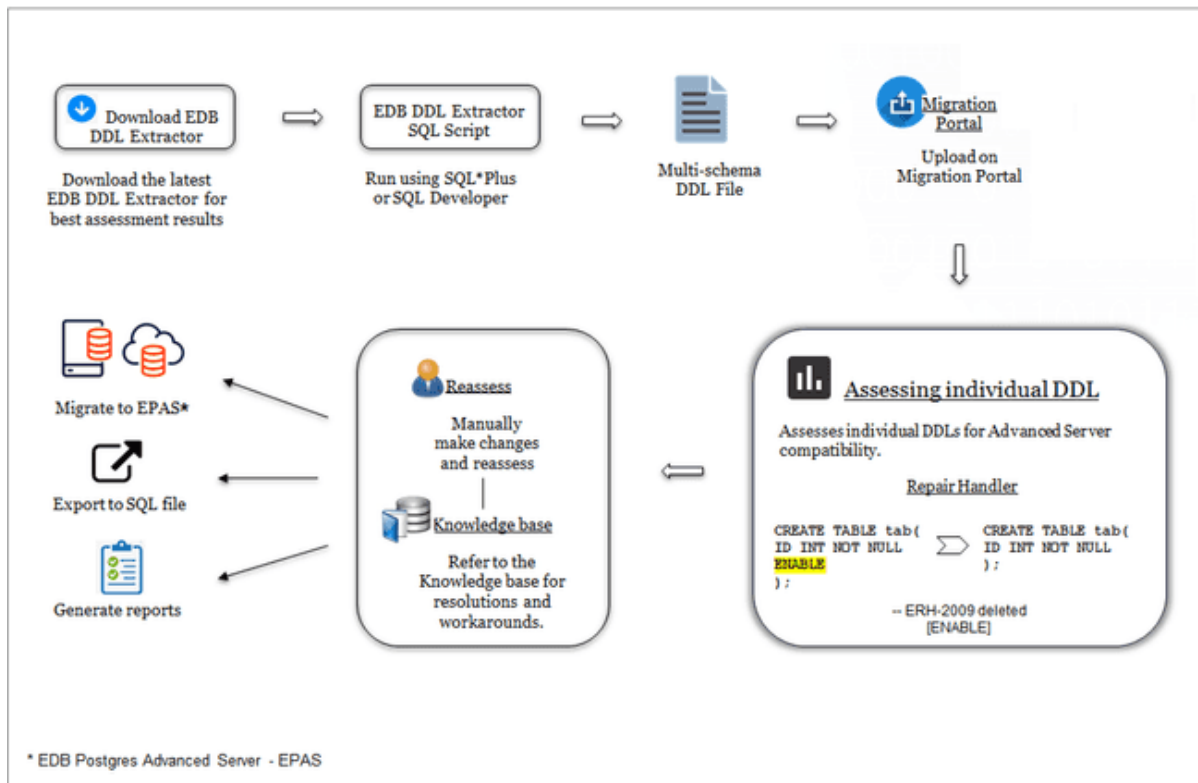


図 1: The assessment and migration process

注釈: 現在、Migration Portal にはチーム コラボレーション機能がありません。共同会議、画面共有、およびダウンロード可能な評価レポートを使用して結果を共有することをお勧めします。

Migration Portal にアクセスするには:

1. ブラウザを開いて、<https://www.enterprisedb.com/> に移動します。
2. EnterpriseDB ホームページで、**PostgreSQL ソフトウェア>移行**を選択します。
3. [Migration Portal] ページで、[**詳細を確認>[MigrationPortal を開く**]]を選択します。
4. 資格情報を使用してログインします。Migration Portal を使用して Oracle スキーマを移行する概要に

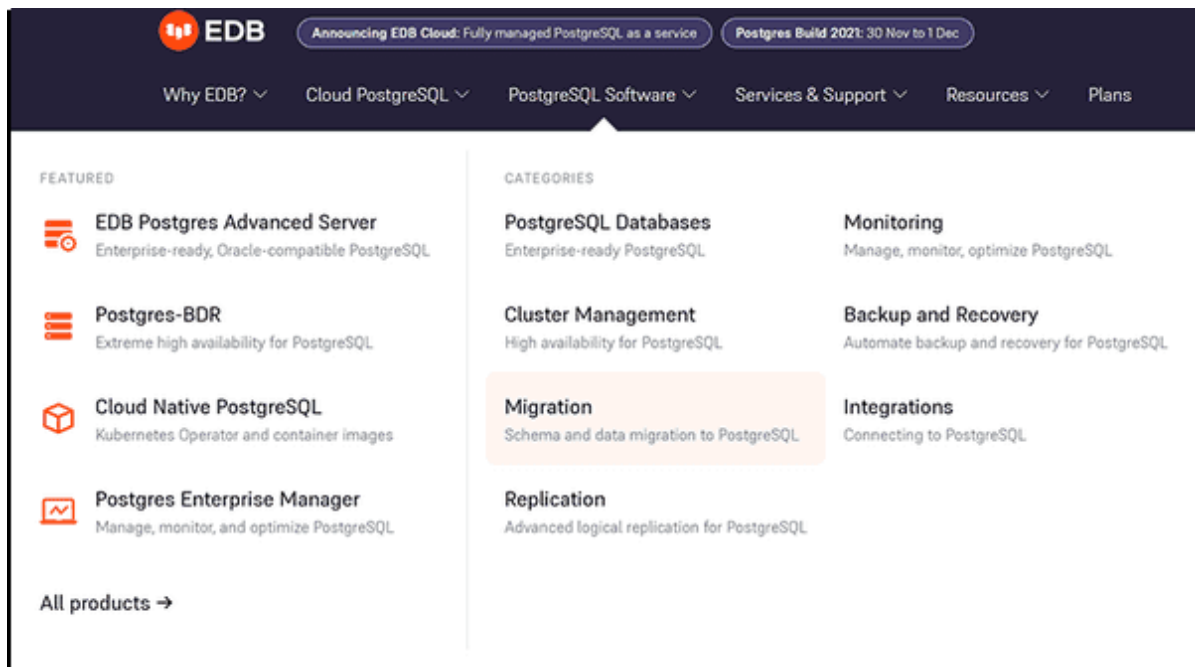


図 2: Migration Portal on the EnterpriseDB home page

については、[Migration Portal quick start](#) を参照してください。

3.1 Overview of the Migration Portal home page

Migration Portal のホームページでは、移行ツールに簡単にアクセスできます。

Migration Portal のホームページでは、次の Migration Portal の機能にアクセスできます。

- **プロジェクト**：[プロジェクト] パネルには、評価されたプロジェクトのリストが表示されます。
- **プロジェクトの作成**：+新規を選択してプロジェクトを作成します。
- **プロジェクトの検索**：プロジェクトを検索するには、**検索ボックス**を使用します。
- **概要**：[概要] パネルには、選択したプロジェクトに関する詳細が表示され、スキーマ評価後の互換性の割合が表示されます。
- **移行先**：[移行先] を選択して、EDB Postgres Advanced Server と互換性のある .sql ファイルをダウンロードするか、オンプレミスまたはクラウド内の EDB Postgres Advanced Server にスキーマを移行します。詳細については、「

[Performing a schema migration](#)」を参照してください。

- **レポート**：[レポート] を選択して、スキーマ評価レポートを表示およびダウンロードします。
- **削除**：選択したプロジェクトを削除するには、**削除**を選択します。
- **警告サイン**：プロジェクトまたはスキーマの互換性が 70% 未満である場合、または複数回試行しても DDL が成功しない場合は、警告メッセージが表示されます。

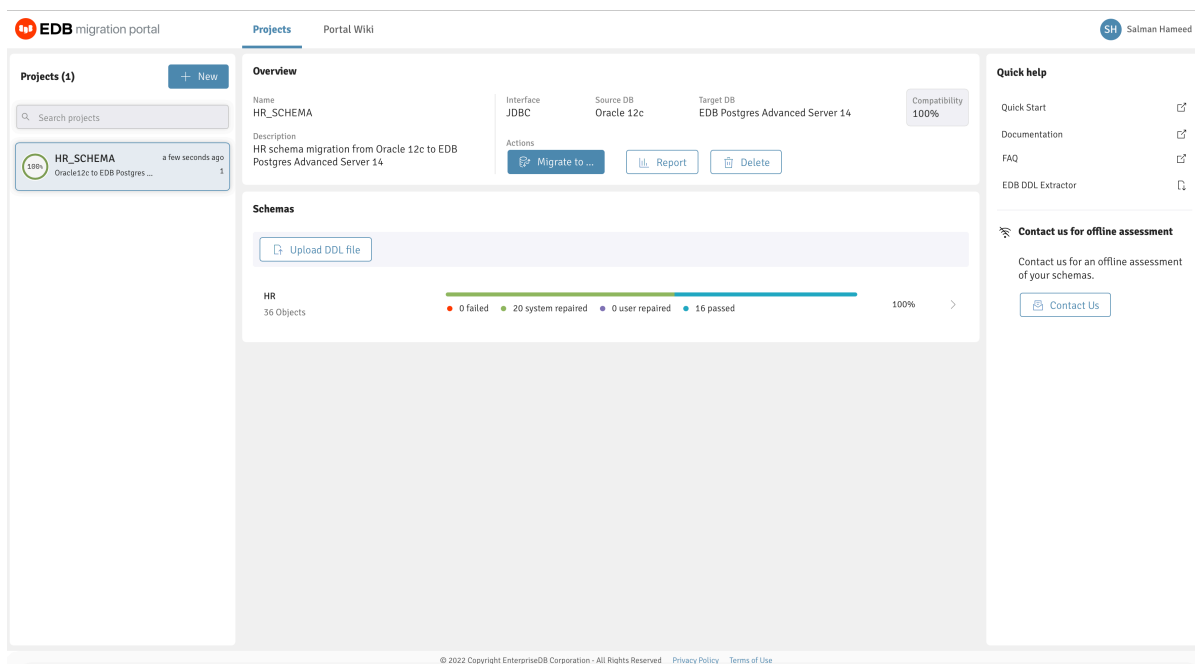


図 3: The Migration Portal home page.

- **DDL ファイルのアップロード**：新しい DDL ファイルをアップロードするには、**新しいスキーマのアップロード** を選択します。
- **スキーマ**：[スキーマ] パネルには、アップロードされた DDL ファイルからの評価結果が表示されます。
- **クイックヘルプ**：クイック ヘルプパネルには、いくつかの必要なリソースへのクイックリンクが含まれています。
- **ポータル Wiki**：製品情報やその他のヘルプガイドへのリンクにアクセスするには、**ポータル Wiki** を選択します。

3.2 Overview of the Migration Portal Projects page

Migration Portal のプロジェクトページには、移行に関する詳細情報が表示されます。

次のリソースを使用して、移行プロジェクトに関する情報を収集します。

- **互換性**：互換性ゲージは、評価されたスキーマの互換性パーセンテージに基づいて色を表示します。
- **スキーマ数**：プロジェクト内のスキーマの数を表示します。
- **移行先**：[移行先] を選択して、スキーマをオンプレミスまたはクラウドの EDB Postgres Advanced Server に移行します。
- **オブジェクトの検索**：オブジェクトを検索するには、**検索ボックス**を使用します。
- **フィルター**：[プロジェクト] ページの左側のパネルから、システム修復されたオブジェクトと手動修復されたオブジェクトをフィルターできます。1 つ以上のフィルターの組み合わせを選択して、情報を絞り込むことができます。

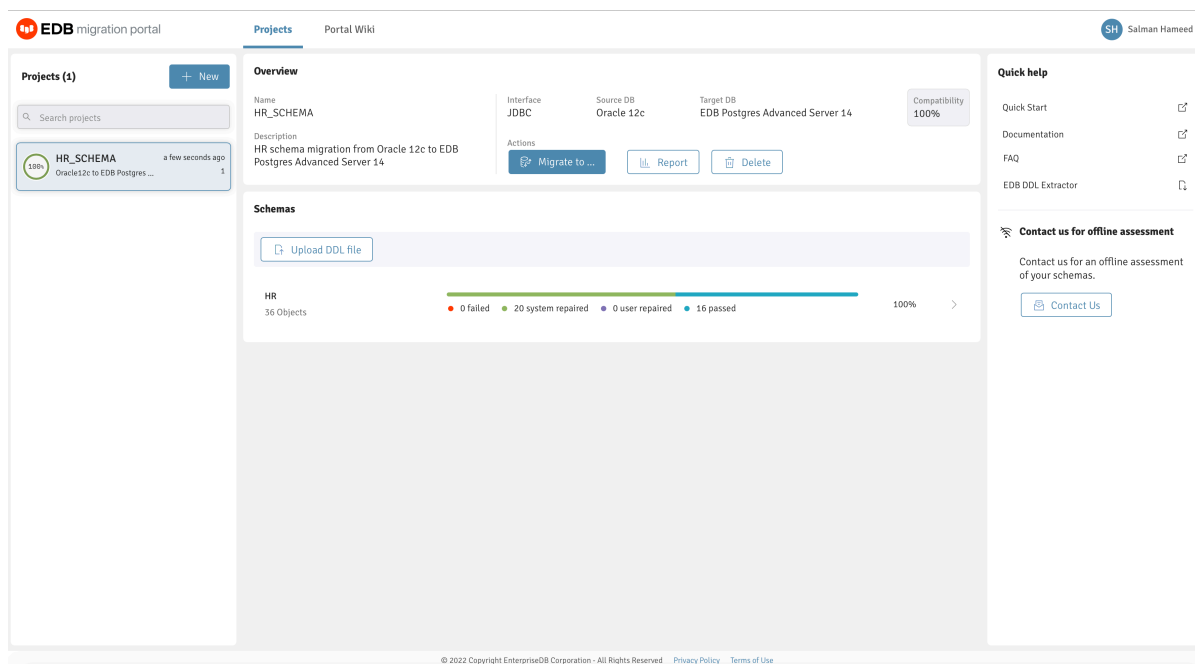


図 4: The Migration Portal Projects Page overview.

- **オブジェクト**：選択したスキーマのオブジェクトを表示します。
- **一般的なエラー**：選択したスキーマのオブジェクトが失敗した理由を表示します。

注釈：プロジェクトの一般的な失敗に関する .csv ファイルをダウンロードできます。

- **スキーマ**：評価の結果を表示します。
- **ツールチップ**：結果セットの上にマウスを置くと、合格、失敗、修復されたオブジェクトの数を示すツールチップが表示されます。
- **クイックヘルプ**：クイック ヘルプ パネルには、ナレッジベースの記事と修復ハンドラーのドキュメントへのリンクが表示されます。
- **検索**：検索 ボックスを使用して、ナレッジベースのエントリまたは修復ハンドラーのドキュメントを検索して特定の情報を探します。

3.3 Migration Portal quick start

Migration Portal を使用して Oracle スキーマを移行するには:

1. スキーマのメタデータ情報を含む .sql ファイルを生成します。詳細については、「[Performing a schema extraction](#)」を参照してください。
2. EDB 資格情報を使用して [Using Migration Portal](#) にログインします。
3. 新しいプロジェクトを作成します。
4. スキーマ メタデータを含む .sql ファイルをアップロードします。

注釈: 大きな DDL ファイルをアップロードすると、ブラウザがフリーズまたはクラッシュする可能性があります (使用しているブラウザとシステム リソースによって異なります)。このシナリオでは、別のブラウザで Migration Portal を実行するか、抽出するスキーマの数を減らしてそのファイルを Migration Portal にアップロードしてみてください。

1. [作成と評価] を選択します。Migration Portal はすべての構成を確認し、 EDB Postgres Advanced Server との互換性を向上させる修復アクションを実行し、手動介入が必要な残りのエラーにフラグを立てます。
2. DDL オブジェクト (たとえば、100% の成功率を示さない TABLES) を確認します。
3. EDB Postgres Advanced Server と互換性のないオブジェクトを選択します。
4. EDB Postgres Advanced Server と互換性のないオブジェクトに対して考えられる回避策を確認するには、ナレッジベースを参照してください。
5. 必要な変更を加え、[再評価] を選択して分析を再実行します。
6. スキーマ内のエラーを解決した後、変更された EDB 互換 DDL ファイルをダウンロードし、pgAdmin や PSQL クライアントなどのクライアントアプリケーションを使用して、スキーマを EDB Postgres Advanced Server に移行できます。クラウド上の EDB Postgres Advanced Server に移行することもできます。詳細については、「[Performing a schema migration](#)」を参照してください。

第4章 Migrating a database

データベースを移行するには、次の手順を完了する必要があります。

1. *Performing a schema extraction* を実行します。
2. *Performing a schema assessment* を実行します。
3. *Performing a schema migration* を実行します。
4. *Performing a data migration* を実行します。

4.1 Performing a schema extraction

次のいずれかの方法を使用してスキーマ抽出を実行できます。EDB では、EDB DDL エクストラクターを使用してスキーマを抽出することをお勧めします。 -

EDB DDL エクストラクタを使用したスキーマの抽出 (推奨方法) - Extracting schemas using Oracle Data Pump utilities

詳細については、「*Known issues, limitations, and notes*」を参照してください。

4.1.1 EDB DDL エクストラクタを使用したスキーマの抽出

最新の EDB DDL エクストラクタスクリプトを Migration Portal のプロジェクトページからダウンロードします。

注釈: Migration Portal は、システムとブラウザのリソースの可用性によっては、非常に大きな SQL ファイルに対して応答しなくなる可能性があります。これを解決するには、一度に抽出するスキーマの数を減らして SQL ファイルのサイズを減らしてみてください。

前提条件

EDB DDL エクストラクタスクリプトは SQL Developer または SQL*Plus で実行できます。Oracle の DBMS_METADATA 組み込みパッケージを使用して、スキーマ (スクリプトの実行中に指定) の下のさまざまなオブジェクトの DDL を抽出します。EDB DDL エクストラクタは、ポータルにアップロードされる DDL ファイルを作成し、EDB Postgres Advanced Server との互換性を分析します。

注釈: *CONNECT* ロールと *SELECT_CATALOG_ROLE* ロール、および *CREATE TABLE* 権限が必要です。

SQL*Plus の場合

1. SQL*Plus に接続し、次のコマンドを実行します。

```
@edb_ddl_extractor.sql
```

2. プロンプトが表示されたら、スクリプトを実行している Oracle ユーザーに必要な権限がある場合は、**Enter** を押してスクリプトの実行を続行します。
3. プロンプトが表示されたら、スキーマ名を入力します。複数のスキーマを抽出する場合は、カンマ (',') を区切り文字として使用します。

注釈: 現在のデータベースからすべてのユーザー スキーマを抽出する場合は、抽出中にスキーマ名を指定しないでください。ただし、抽出するスキーマ名を指定することをお勧めします。

4. プロンプトが表示されたら、抽出された DDL を保存するエクストラクターのパスまたはディレクトリを入力します。パスを指定しない場合、抽出された DDL は、SQL*Plus コマンドを実行したディレクトリに出力されます。
5. プロンプトが表示されたら、他のスキーマから依存オブジェクトを抽出するかどうかに応じて、**yes** または **no** を入力します。
6. プロンプトが表示されたら、他のスキーマから Grant ステートメントを抽出するかどうかに応じて、**yes** または **no** を入力します。

例えば：

1. 抽出スクリプトを実行します。

```
@edb_ddl_extractor.sql
```

2. **Enter** を押します
3. 3 つのスキーマをカンマで区切って指定します。

```
Enter a comma separated list of schemas to be extracted (Default all schemas): HR,
→ SCOTT, FINANCE
```

4. 出力ファイルのパスを入力します。

Linux では、次のものを使用できます。

```
Location for output file (Default current location) : /home/oracle/extracted_ddls/
```

Windows でも同様に:

```
Location for output file (Default current location) : c:\Users\Example\Desktop\
```

5. yes と入力して、他のスキーマ内の依存オブジェクトを抽出します。

WARNING:

Given schema(s) list may contain objects which are dependent on objects from other schema(s), not mentioned in the list. `Assessment may fail for such objects. It is suggested to extract all dependent objects together.

Extract dependent object from other schemas?(yes/no) (Default no / Ignored for all schemas option): yes

6. yes と入力して許可ステートメントを抽出します。

```
Extract GRANT statements?(yes/no) (Default no): yes
```

SQL 開発者の場合

edb_ddl_extractor.sql スクリプトを SQL Developer にロードし、ソース Oracle データベースに接続した後、スクリプトを実行します。スクリプトが実行されると、プロンプトに応答します。

1. データベース接続のユーザーが [スクリプト出力] タブにリストされている必要な権限を持っている場合は、**はい**を選択してスクリプトの実行を続行します。
2. スキーマのカンマ区切りリストを入力し、**OK** を選択します。
3. 出力ファイルのパスを入力し、**OK** を選択します。デフォルトは、DDL エクストラクタスクリプトの場所です。
4. yes または no を入力して依存オブジェクトを抽出し、**OK** を選択します。デフォルトでは、依存オブジェクトは抽出されません。
5. yes または no を入力して許可ステートメントを抽出し、**OK** を選択します。デフォルト設定では、grant ステートメントを抽出しません。

DDL エクストラクタ実行の出力が [スクリプト出力] タブに表示されます。出力ファイルの名前は、スクリプト出力の Extraction Completed メッセージの後に表示されます。

注釈: SQL*Plus と SQL Developer の両方に単一のスキーマ名を入力することもできます。

次に、スクリプトはソース データベース内のオブジェクトタイプを反復処理します。タスクが完了すると、.SQL 出力は入力した場所 (c:\Users\Example\Desktop\ など) に保存されます。

Migration Portal で想定されるファイルエンコード形式については、[ファイルのエンコード](#) を参照してください。

4.1.2 スキーマとオブジェクトのサポート

次のリストと表は、Migration Portal でサポートされているスキーマとオブジェクトとサポートされていないスキーマとオブジェクトを示しています。

サポートされていないスキーマ

SQL ダンプ ファイルの生成中に、これらの Oracle システムスキーマを除外します。

1	2	3	4	5
ANONYMOUS	APEX_PUBLIC_USER	APEX_030200	APEX_040000	APEX_040200
APPQOSSYS	AUDSYS	CTXSYS	DMSYS	DBSNMP
DBSFWRUSER	DEMO	DIP	DMSYS	DVF
DVSYs	EXFSYS	FLows_FILES	FLows_020100	FRANCK
GGSYs	GSMADMIN_INTERNAL	GSMCATUSER	GSMROOTUSER	GSMUSER
LBACSYs	MDDATA	MDSYS	MGMT_VIEW	OJVMSYS
OLAPSYs	ORDPLUGINS	ORDSYS	ORDDATA	OUTLN
ORACLE_OCM	OWBSYS	OWBYSS_AUDIT	PDBADMIN	RMAN
REMOTE_SCHEDULER_AGENT	SI_INFORMTN_SCHEMA	SPATIAL_CSW_ADMIN_USR	SPATIAL_WFS_ADMIN_USR	SQLTXADMIN
SQLTXPLAIN	SYSSUMF	SYS	SYSBACKUP	SYSDG
SYSKM	SYSRAC	SYSTEM	SYSMAN	TSM SYS
WKPROXY	WKSYs	WK_TEST	WMSYS	XDB
XS\$NULL				

注釈: EDB Postgres Advanced Server は、`pg_` で始まるスキーマ名をサポートしません。

サポートされているオブジェクトの種類

- テーブル
- シーケンス
- 制約
- インデックス (LOB インデックスとマテリアライズドビューのインデックスを除く)
- 同義語
- ビュー
- マテリアライズドビュー
- DB リンク
- 型と型本体
- トリガー
- 機能
- 手順
- パッケージ
- ユーザー
- 役割
- プロフィール
- 役割とオブジェクトの付与

注釈: 列、テーブル、マテリアライズドビューのコメントもサポートされています。

サポートされていないオブジェクト タイプ

- エディション
- オペレーター
- スケジューラ
- LOB インデックスとマテリアライズドビューのインデックス
- XML スキーマ

- テーブルスペース
- ディレクトリ
- RLS ポリシー
- キュー *
- ライブラリー
- インデックスタイプ

* EDB Postgres Advanced Server はキューテーブルのサポートを提供しますが、Migration Portal は現在それをサポートしていません。ソース DDL のキューテーブルは、ソースおよびターゲット DDL オブジェクトとしてアップロードされません。

Extracting schemas using Oracle Data Pump utilities

Migration Portal では、抽出されたスキーマを評価するために SQL ダンプファイル内のメタデータのみが必要です。

前提条件

- 自分のものではないスキーマをエクスポートする予定がある場合は、DATAPUMP_IMP_FULL_DATABASE ロールが割り当てられていることを確認してください。それ以外の場合は、独自のスキーマのみをエクスポートできます。
- 表領域にオブジェクトを作成するのに十分な表領域割り当てがあることを確認してください。
- SQL *Plus で、サーバーファイルシステム上のディレクトリを指すディレクトリオブジェクトを作成します。例: **Linux の場合**:

```
create directory DMPDIR as /sample/file/path;
```


Windows の場合 :


```
```sql
create directory DMPDIR as c:\sample\file\path\;
```


```

- SQL*Plus では、Data Pump ユーティリティを実行しているユーザーに読み取り/書き込み権限を付与します。

```
grant read,write on directory DMPDIR to <Username>;
```

次のいずれかの手順を実行します。

- データベース内の 1 つ以上のスキーマを抽出します
- データベース内のすべてのスキーマを抽出します

データベース内の 1 つ以上のスキーマを抽出します

1. expdp コマンドを実行する前に、サーバー上に .par 拡張子を持つファイル (export.par など) を作成します。属性と値をファイルに追加します。

edb_notranlate_3 このコマンドの属性と値は、次のオプションを指定します。- CONTENT=metadata_only は、スキーマのメタデータのみを抽出します。- INCLUDE= は、抽出に含めるデータベース オブジェクト タイプを指定します。Migration Portal でサポートされているデータベースオブジェクト タイプのみを抽出します。 [サポートされているオブジェクトの種類](#) を参照してください。

2. コマンドラインから、export コマンドを実行して schemas_metadata.dump ファイルを生成します。たとえば、<Schema_1>、<Schema_2>、<Schema_3> などのメタデータ情報を抽出するには、次を実行します。

```
expdp <Username>@<ConnectIdentifier> DIRECTORY=DMPDIR SCHEMAS=<Schema_1>,<Schema_2>,  
→<Schema_3> DUMPFILE=schemas_metadata.dump parfile=export.par
```

3. ダンプ ファイルから SQL ファイルを生成するには、import コマンドを実行します。

たとえば、schemas_metadata.dump ファイルから YourSchemas.SQL ファイルを生成するには、次のように入力します。

```
impdp <Username>@<ConnectIdentifier> DIRECTORY=DMPDIR TRANSFORM=OID:n,SEGMENT_  
→ATTRIBUTES:n EXCLUDE=STATISTICS SQLFILE=YourSchemas.sql DUMPFILE=schemas_metadata.  
→dump
```

Migration Portal で期待されるファイルエンコード形式については、 [ファイルのエンコード](#) を参照してください。

データベース内のすべてのスキーマを抽出します

注釈: 除外スキーマ リストに属するユーザー アカウントからこの手順を実行しないでください ([サポートされていないスキーマ](#) を参照)。コマンドを実行しているユーザー アカウントがスキーマの除外リストに含まれている場合、impdp コマンドは失敗します。

1. expdp コマンドを実行する前に、サーバー上に .par 拡張子を持つファイル (export.par など) を作成します。属性と値をファイルに追加します。

edb_notranlate_6 このコマンドの属性と値は、次のオプションを指定します。- CONTENT=metadata_only は、スキーマのメタデータのみを抽出します。- INCLUDE= は、抽出に含めるデータベース オブジェクト タイプを指定します。Migration Portal でサポートされているデータベースオブジェクト タイプのみを抽出します。 [サポートされているオブジェクトの種類](#) を参照してください。- FULL=Y はデータベース内のすべてのスキーマを抽出します。

2. コマンド ラインから、export コマンドを実行して schemas_metadata.dump ファイルを生成します。

```
expdp <Username>@<ConnectIdentifier> DIRECTORY=DMPDIR DUMPFILE=schemas_metadata.dump_
↳parfile=export.par
```

3. `impdp` コマンドを実行する前に、サーバー上に `.par` 拡張子を持つパラメータ ファイル (`import.par` など) を作成します。属性と値をファイルに追加します。

```
EXCLUDE=SCHEMA:"IN(select username as schema_name from sys.dba_users where_
↳LOWER(username) LIKE pg_% OR username IN
(ANONYMOUS,APEX_PUBLIC_USER,APEX_030200,APEX_040000,APEX_040200,APPQOSSYS,AUDSYS,
↳CTXSYS,DMSYS,DBSNMP,
DBSFUSER,DEMO,DIP,DMSYS,DVF,DVSY,EXFSYS,FLWS_FILES,FLWS_020100,FRANCK,GGSYS,
↳GSMADMIN_INTERNAL,
GSMCATUSER,GSMROOTUSER,GSMUSER,LBACSYS,MDDATA,MDSYS,MGMT_VIEW,OJVMSYS,OLAPSYS,
↳ORDPLUGINS,ORDSYS,
ORDDATA,OUTLN,ORACLE_OCM,OWBSYS,OWBYSS_AUDIT,PDBADMIN,RMAN,REMOTE_SCHEDULER_AGENT,SI_
↳INFORMTN_SCHEMA,
SPATIAL_CSW_ADMIN_USR,SPATIAL_WFS_ADMIN_USR,SQLTXADMIN,SQLTXPLAIN,SYSSUMF,SYS,
↳SYSBACKUP,SYSDG,SYSKM,
SYSRAC,SYSTEM,SYSMAN,TSMSYS,WKPROXY,WKSYS,WK_TEST,WMSYS,XDB,XS$NULL))"
EXCLUDE=STATISTICS
```

このコマンドの属性と値は、次のオプションを指定します。 - `EXCLUDE` は、除外するスキーマを指定します。 **サポートされていないスキーマ**のリストを追加して、データ・ポンプ・ユーティリティがシステム・スキーマ (`pg_` で始まるスキーマを含む) を抽出しないようにします。 - `EXCLUDE=STATISTICS` は、SQL 出力ファイルに統計関連の情報を含まないように指定します。これらを SQL ファイルに含めないことにより、移行ポータルの DDL を評価する機能に影響を与えることなく、ファイルのサイズが大幅に削減されます。

4. ダンプ ファイルから SQL ファイルを生成するには、`import` コマンドを実行します。

たとえば、`schemas_metadata.dump` ファイルから `YourSchemas.SQL` ファイルを生成するには、次のように入力します。

`edb_notranlate_9` Migration Portal で予期されるファイルエンコード形式については、[ファイルのエンコード](#) を参照してください。

4.2 Performing a schema assessment

Oracle データベース スキーマと EDB Postgres Advanced Server との互換性を評価できます。評価に関するレポートを生成することもできます。

4.2.1 スキーマ評価を実行する

1. *Using Migration Portal* に移動します。
2. EDB 資格情報を入力します。
3. [**新規**] を選択してプロジェクトを作成します。
4. ダイアログ ボックスで、 **プロジェクト名** フィールドにプロジェクト名を入力します。
5. プロジェクトの詳細を指定します。
 - アプリケーションインターフェイスを選択します。 - ソース DB とソースのバージョンを選択します。 - ターゲット DB とターゲットのバージョンを選択します。
6. スキーマ メタデータで生成された .SQL ファイルをアップロードするには、[**ファイルを選択**] を選択します。

注釈: .SQL ファイルを変更しないでください。

7. プロジェクトの作成時にインデックス (_IDX) および制約 (_CON) にサフィックスを追加するように指定するには、[**オブジェクト名を一意にするためにサフィックスを追加**] チェックボックスをオンにします。 EDB Postgres Advanced Server テーブルとインデックスの同じ名前がサポートされていませんが、Oracle ではサポートされているため、この選択によりより良い評価結果が得られます。
8. プロジェクトの作成時に Oracle のデフォルトのケースを使用することを指定するには、**スキーマ移行に Oracle のデフォルトのケースを使用する** チェックボックスをオンにします。 Oracle データベースから EDB Postgres Advanced Server データベースにスキーマを移行するときに、Oracle のデフォルトの大文字と小文字の機能を使用して、すべてのデータベースオブジェクトに対して Oracle のデフォルトの大文字命名規則を保持します。この機能の詳細については、 *Known issues, limitations, and notes* を参照してください。
9. [**作成と評価**] を選択して、新しいプロジェクトを作成し、スキーマを評価します。

分析ツールはすべての構成をレビューし、 EDB Postgres Advanced Server との互換性を向上させる修復アクションを実行します。手動介入が必要な残りのエラーにフラグを立てます。

10. 100% の成功率を示さない DDL オブジェクト (TABLES など) を確認します。
11. EDB Postgres Advanced Server と互換性のないオブジェクトを選択し、詳細を表示します。中央パネルの下部に、修復アクションの詳細が表示されます。
12. 右パネルのナレッジベース情報を参照して、 EDB Postgres Advanced Server とすぐには互換性がないオブジェクトの考えられる回避策を見つけます。ポータル wiki ページでナレッジ ベース情報を表示することもできます。
13. [**ナレッジベース**] タブで、 EDB Postgres Advanced Server と互換性のないオブジェクトに関するエラー メッセージを入力し、 **検索** を選択します。

オブジェクト詳細パネルには、失敗したオブジェクトの回避策または解決策が表示されます。そのオブジェクトの [**評価**] タブで手動で変更を加え、[**再評価**] を選択できます。

New project ×

Project name

e.g. First project

Application interface

☒ JDBC ☐ ODBC ☐ .NET ☐ OCI ☐ ProC ☐ Other

Source DB

Oracle

Version

11g

Target DB

EDB Postgres Advanced Server

Version

15

DDL file [\(How to get a DDL file?\)](#)

No file chosen

 Choose file

☐ Add suffix to make object names unique ?

☐ Use Oracle's default case for schema migration. ?

Description(Optional)

Say something about the project...

200 characters left

× Cancel

✓ Create & assess

図 1: The Migration Portal New project dialog

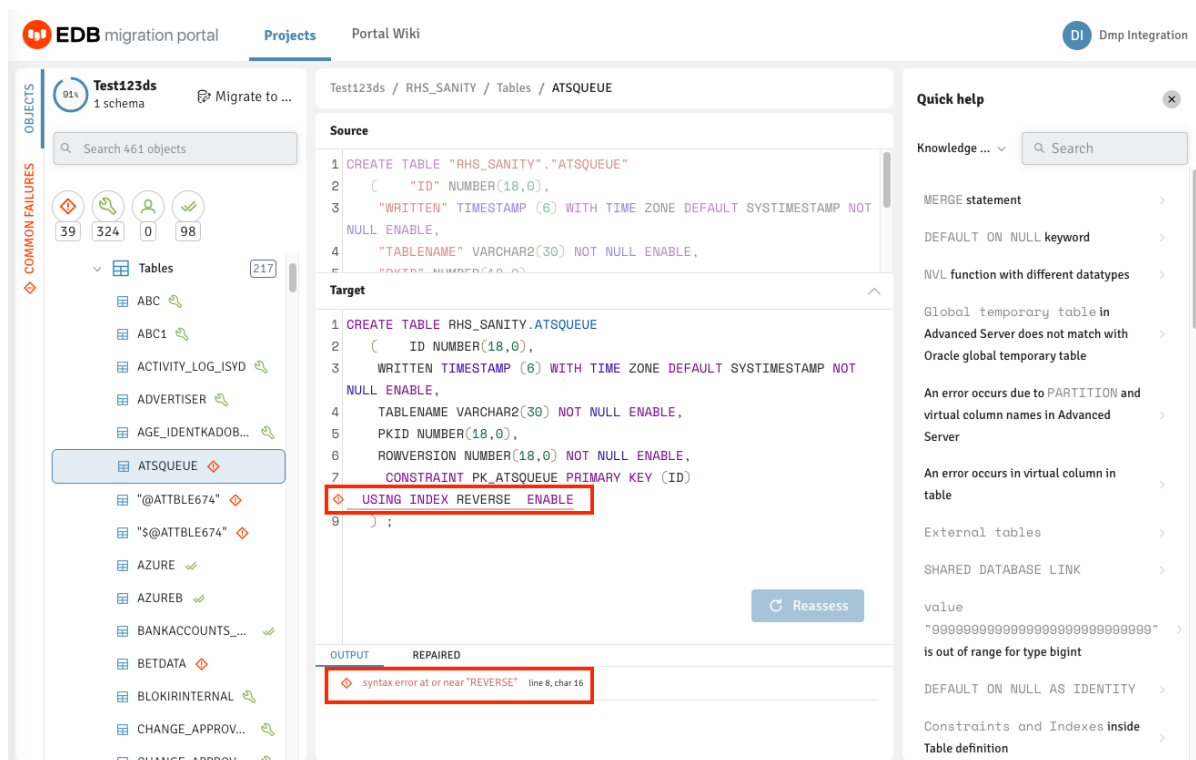


Figure 2: Incompatible objects are identified

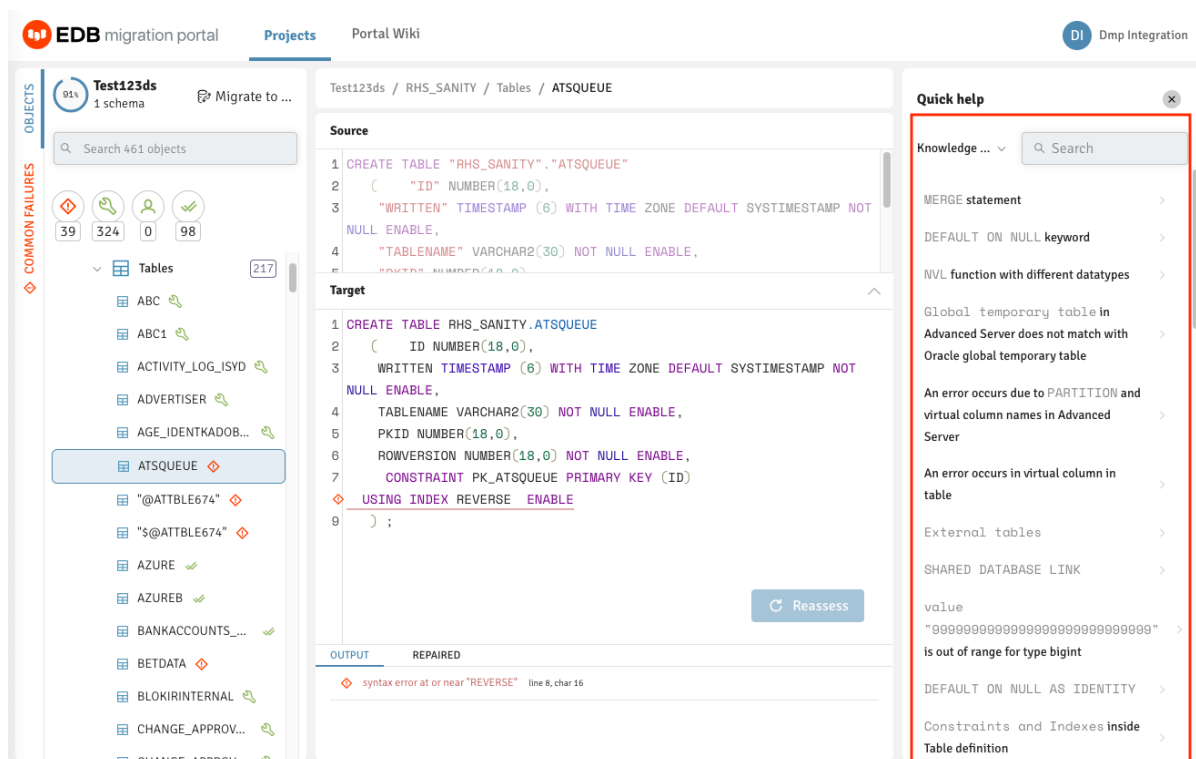


Figure 3: Incompatible objects are identified

注釈: 再評価中に失敗したオブジェクトが合格した場合、そのオブジェクトの依存オブジェクトも再評価されます。

同様に、互換性のないオブジェクトをすべて互換性のあるものにすることができます。

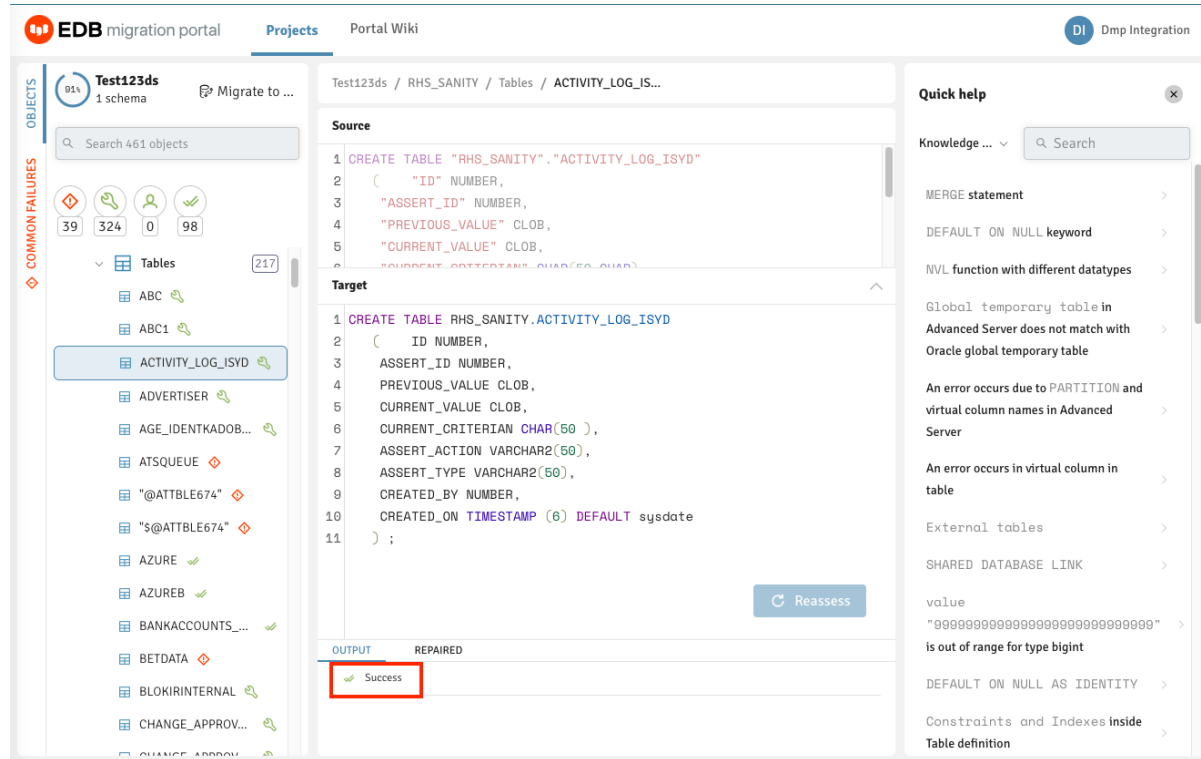


図 4: Workaround or resolution for incompatible objects

注釈: 互換性のないオブジェクトに関する情報や回避策がナレッジベースで入手できない場合は、サポートチームに問い合わせてください。

DDL の操作が終了したら、変更された EDB 互換 DDL を .sql ファイルとしてダウンロードするか、スキーマを移行できます。 [Performing a schema migration](#) を参照してください。

4.2.2 評価レポートを生成する

Migration Portal レポートは、プロジェクトに対して評価されたスキーマの高レベルの評価概要を提供します。さらに、レポートでは、失敗したオブジェクトと失敗の原因に関する詳細が提供されます。

1. [レポート] を選択します。
2. レポートに含めるスキーマを選択します。
3. 画面上のレポートを生成するには、[生成] を選択します。

必要に応じて、[**PDF の生成**] を選択して、レポートを PDF 形式で保存できます。選択したスキーマの DDL に適用された個別の修復ハンドラーの数を表示することもできます。

注釈: Migration Portal は、Oracle でラップされたオブジェクトに隠された機密性の高い PL/SQL ソースコードを評価しません。これらのラップされたオブジェクトは評価されたオブジェクトの数には含まれないため、互換性パーセンテージの実際の値は、評価レポートで計算された値と異なる可能性があります。ラップされたオブジェクトの詳細については、[Known issues and notes](#) の注を参照してください。

4.2.3 評価レポートを評価する

評価レポートには、スキーマおよびプロジェクト全体におけるオブジェクトと機能の互換性に関するデータと情報が表示されます。

- **オブジェクト:** Migration Portal は、処理するすべての CREATE ステートメントのオブジェクトを生成します。さらに、アップロードされた DDL に制約を追加する ALTER TABLE ステートメントごとにオブジェクトを生成します。
- **機能:** オブジェクトの作成に使用される Oracle 句または構成要素。フィーチャは 1 つ以上のオブジェクトで複数回使用できます。

評価レポートには、**互換性のない機能** と **未解決のオブジェクト** の表がリストされます。

オブジェクトの互換性

オブジェクトがターゲットデータベース上で正常に実行される場合、そのオブジェクトには互換性があります。評価レポートには、次のカテゴリに基づいたオブジェクトの互換性の概要が表示されます。

| Object level | compatibility | Description |
|-----------------|---------------|--|
| Passed | | 修復ハンドラーを適用せずにターゲット データベース上で実行します。 |
| System repaired | | 1 つ以上の修復ハンドラーを適用した後、ターゲット データベースで正常に実行されます。修復ハンドラーは変換ロジックを使用して、DDL を EDB Postgres Advanced Server と互換性のあるものに自動的に作成します。オブジェクトの動作に予期しない影響があった場合に備えて、レポートの 適用された修復ハンドラー セクションの説明を必ず注意深く確認してください。 |
| User repaired | | ユーザーが手動で修復して再評価した後、ターゲット データベースで正常に実行されます。 |
| Failed objects | | ターゲット データベースで正常に実行されません。 |

注釈: 表示されるオブジェクト分析は、ターゲット DDL 内のオブジェクトの既存の状態に基づいており、オブジェクトが更新および再評価されると変更される可能性があります。たとえば、修正されたオブジェクトは、再評価後に **未解決のオブジェクト** テーブルから削除されます。

機能の互換性

一部の機能は、EDB Postgres Advanced Server ではネイティブに使用できない場合があります。評価レポートには、機能の発生と互換性レベルの概要が表示されます。

| Feature level | compatibility | Workaround status |
|--|---------------|----------------------------|
| Compatible features | | |
| • | | |
| Features with existing available workaround | | Exists |
| Features with reasonable effort new workaround | | Possible |
| Features with no workaround or requiring further investigation | | None / Needs Investigation |

注釈: 回避策がない機能の場合は、アプリケーション レベルの変更によって機能の非互換性に対処できる可能性があります。#### 制限事項

ユーザーがターゲット DDL を手動で変更し、**再評価**を選択すると、評価レポートが更新されます。ただし、レポート内の**互換性のない機能**テーブルは更新されず、ソース DDL 内の互換性のない機能が引き続きリストされます。スキーマ移行の複雑さの評価スコアは、互換性のない機能を持つ互換性のないオブジェクトが解決されると更新されます。また、オブジェクトが修正され再評価された後、それらのオブジェクトは、それらのオブジェクトで元々見つかった互換性のない機能のリストとともに、**未解決のオブジェクト** テーブルから削除されます。

4.3 Performing a schema migration

スキーマ内のエラーを解決した後、スキーマを EDB Postgres Advanced Server データベースに移行できます。

Migration Portal では、次のターゲット データベースオプションのいずれかを使用してスキーマを移行するためのガイド付き手順が提供されます。

- オフライン移行- データ定義言語 (DDL) SQL ファイルをダウンロードし、psql クライアントや pgAdmin などのクライアント アプリケーションを使用してターゲット EDB Postgres Advanced Server データベースにスキーマを手動で作成します。このオプションは、オンプレミスとクラウドベースの両方の移行に適しています。
- オンライン移行- ターゲットの EDB Postgres Advanced Server データベースに直接接続し、その中にスキーマを自動的に作成します。一般に、このオプションは、クラウド (BigAnimal クラスターまたはセルフマネージド インスタンス) で実行されているデータベースへの移行に最も適しています。

EDB では、スキーマを移行する前に、[Known issues, limitations, and notes](#) でターゲット・データベースの構成要件を特定することをお勧めします。

4.3.1 オフライン移行

クライアントを使用して DDL SQL ファイルを実行することにより、Windows または Linux プラットフォーム上の EDB Postgres Advanced Server にスキーマを手動で移行します。

1. [オフライン移行] を選択します。
2. EDB Postgres Advanced Server に移行する 1 つ以上のスキーマを選択します。スキーマを展開して、各スキーマで特定のオブジェクトタイプを選択することもできます。

Generate and download DDL SQL scripts for offline migration of schemas ?

1 Select Schemas
2 Download Schema
3 Finish

| Project Name | Interface | Source DB | Target DB | Compatibility |
|---------------|-----------|------------|---------------------------------|---------------|
| TestMigration | JDBC | Oracle 18c | EDB Postgres Advanced Server 14 | 100% |

Select one or more schemas

Note: Click the arrows next to the schema name to see and select specific object types to be migrated. Also note that the numbers in parenthesis next to the object type name indicate the number of compatible objects, total number of objects, and % of compatible objects currently available in the schema for the object type.

☒ All

☒ GLOBAL_OBJECTS

☒ HRPLUS

☒ CONSTRAINT (10/10 - 100%)
 ☒ FUNCTION (2/2 - 100%)
 ☒ INDEX (11/11 - 100%)
 ☒ PACKAGE (1/1 - 100%)
 ☒ PACKAGE BODY (1/1 - 100%)
 ☒ PROCEDURE (9/9 - 100%)
 ☒ SEQUENCE (3/3 - 100%)
 ☒ TABLE (8/8 - 100%)
 ☒ TRIGGER (3/3 - 100%)
 ☒ VIEW (8/8 - 100%)

☒ SUPER

Cancel

← Previous

→ Next

注釈: スキーマの互換性が 100% でない場合は、バナーが表示されます。必要に応じて、お問い合わせフォームに記入してください。

1. [SQL ファイルのダウンロード] を選択して、評価済みのスキーマをダウンロードします。zip パッケージは、選択したスキーマとオブジェクトの DDL を含む SQL ファイルで構成されます。DDL は、プロジェクト、スキーマ、またはオブジェクトごとに SQL ファイルにグループ化されます。

36

第 4 章 Migrating a database

| File name | Description | Example |
|-------------------------------|---|---|
| <ProjectName>_ALL_OBJECTS.sql | 選択したすべてのスキーマとオブジェクトの DDL で構成される単一の統合ファイル。 | AllTestMigration_ALL_OBJECTS.sql consists of DDLs for all objects of schemas OE, SH, IX, and PM, and only selected objects of schema HR. |
| <SchemaName>.sql | 1 つ以上のファイル。各ファイルは特定のスキーマ内の選択されたオブジェクトの DDL で構成されます。 | HR.sql consists of DDLs for all the selected objects of the HR schema. |
| <SchemaName>#<ObjectType>.sql | 1 つ以上のファイル。各ファイルは、選択した 1 つのオブジェクトの DDL で構成されます。 | HR_TABLE.sql consists of DDLs only for TABLE object of the HR schema. |

1. zip ファイルのダウンロードが完了したら、[完了] を選択して Migration Portal のメイン ページに戻ります。
2. スキーマをターゲット データベースにインポートします。
3. edb psql コマンドラインユーティリティを使用して、変換されたスキーマの移行元となる EDB Postgres Advanced Server データベースに接続します。
4. 新しいデータベースを作成します。

```
CREATE DATABASE <database_name>;
```

1. 次の psql コマンドを使用して、新しいデータベースに接続します。

```
\connect <database_name>
```

1. 次の psql コマンドを使用して、必要な SQL ファイルを新しいデータベースにインポートします。

```
\i <path_to_exported_schema_file>.sql
```

注釈: インポートには *pgAdmin* クライアントを使用することもできます。スキーマをターゲット データベースにインポートします。

4.3.2 オンライン移行

ターゲットの EDB Postgres Advanced Server データベースに直接接続してスキーマを作成します。

1. [オンライン移行] を選択します。
2. EDB Postgres Advanced Server に移行する 1 つ以上のスキーマを選択します。スキーマを展開して、各スキーマで特定のオブジェクトタイプを選択することもできます。

Connect to database and perform online migration of schemas ?

1 Select Schemas
2 Connect
3 Deploy
4 Finish

| Project Name | Interface | Source DB | Target DB | Compatibility |
|---------------|-----------|------------|---------------------------------|---------------|
| TestMigration | JDBC | Oracle 18c | EDB Postgres Advanced Server 14 | 100% |

Select one or more schemas

Note: Click the arrows next to the schema name to see and select specific object types to be migrated. Also note that the numbers in parenthesis next to the object type name indicate the number of compatible objects, total number of objects, and % of compatible objects currently available in the schema for the object type.

☒ All

> ☒ GLOBAL_OBJECTS

< ☒ HRPLUS

☒ CONSTRAINT (10/10 - 100%)
 ☒ FUNCTION (2/2 - 100%)
 ☒ INDEX (11/11 - 100%)
 ☒ PACKAGE (1/1 - 100%)
 ☒ PACKAGE BODY (1/1 - 100%)
 ☒ PROCEDURE (9/9 - 100%)
 ☒ SEQUENCE (3/3 - 100%)
 ☒ TABLE (8/8 - 100%)
 ☒ TRIGGER (3/3 - 100%)
 ☒ VIEW (8/8 - 100%)

> ☒ SUPER

Cancel

← Previous

→ Next

3. Connection ページに必要な接続の詳細を入力します。

1. 接続の詳細を確認するには、**接続のテスト** を選択します。

注釈: [接続の編集] を選択して、接続の詳細を変更し、接続の詳細を再テストできます。

1. テスト接続が成功した場合は、[次へ] を選択してターゲットデータベースに接続し、選択したスキーマをターゲットサーバーに作成します。

1. 移行が完了すると、移行が正常に完了したかエラーが発生したかを示すメッセージがページに表示されます。[概要をダウンロード] ボタンを選択して、移行試行に対して生成されたログをダウンロードして確認します。

Connect to database and perform online migration of schemas



✓ Select Schemas

2 Connect

3 Deploy

4 Finish

| | | | | |
|---------------|-----------|------------|---------------------------------|---------------|
| Project Name | Interface | Source DB | Target DB | Compatibility |
| TestMigration | JDBC | Oracle 18c | EDB Postgres Advanced Server 14 | 100% |

Connect to your cluster

Target Database

TestMigration

Host Name/Address

Host Name/Address

Port

5432

Note: Ensure that your database server is accessible from Migration Portal.

Maintenance Database

edb_admin

Username

edb_admin

Password

Password

Test Connection

Cancel

← Previous

→ Next

図 5: Connecting to the cloud cluster

1. [完了] を選択して、Migration Portal のメイン ページに戻ります。

4.4 Performing a data migration

スキーマの移行を実行した後、データを移行します。

1. Migration Toolkit を使用してデータを移行します。 Migration Toolkit の使用方法の詳細については、「 Migration Toolkit 」を参照してください。

2. Migration Toolkit の `toolkit.properties` ファイルを構成し、ソースデータベースとターゲット データベースの接続情報がプロパティファイルで利用できることを確認します。

```
SRC_DB_URL = jdbc: oracle:thin:@localhost:1521:ORCL
SRC_DB_USER = user_name
SRC_DB_PASSWORD = password
TARGET_DB_URL= jdbc:edb://localhost:5444/migration
TARGET_DB_USER = enterprisedb
TARGET_DB-PASSWORD = password
```

詳細については、「 Building the toolkit.properties File 」を参照してください。

3. `-dataOnly` モードで Migration Toolkit を起動します。 `-truncLoad` キーワードを含めて、テーブル間の外部キーの依存関係を解決します。

たとえば、次のコマンドはデータを `hr` という名前のスキーマに移行します。

```
runMTK.sh -dataOnly -targetSchema hr -truncLoad HR
```

このコマンドは、指定された `source_schema` を `target_schema` に移行します。データは、`enterprisedb` という名前のデータベーススーパーユーザーと `password` のパスワードを使用して、ローカルにインストールされた EDB Postgres Advanced Server インスタンスにロードされます。

注釈: テーブルはデータのロードを試行する前に切り詰められます。

4.4.1 高度なデータ移行

並列データロードが必要な大規模なデータベースの場合は、次のいずれかの方法を使用できます。

- EDB Postgres Advanced Server データベースリンク機能 (Oracle データベースとの互換性のため)
- Dblink またはデータベース リンク スタイルの移行 (データに CLOB データが含まれている場合)

詳細については、 Migration Toolkit を参照してください。

第5章 Known issues, limitations, and notes

このページには、次の既知の問題、制限事項、および注意事項がリストされています。

- *Using Migration Portal - EDB DDL エクストラクタを使用したスキーマの抽出 - Extracting schemas using Oracle Data Pump utilities*

5.1 Migration Portal

5.1.1 ラップされたオブジェクト

Migration Portal はラップされたオブジェクトを評価できません。抽出された DDL にそれらを含めた場合、それらは Migration Portal にロードされず、評価されるオブジェクトの数にも含まれません。ラップされたオブジェクトを評価し、EDB Postgres Advanced Server に移行する場合は、Migration Portal にアップロードする DDL ファイルに、ラップされていないバージョンのオブジェクトを含めます。これを行うための推奨される方法は、スキーマ抽出を実行する前に、Oracle データベース内のラップされたバージョンをクリアテキストバージョンに置き換えることです。スキーマ抽出を実行した後、オブジェクトをラップされたバージョンに置き換えることができます。

5.1.2 ファイルのエンコード

Migration Portal では、.SQL 出力ファイルを UTF-8 エンコード形式にすることをお勧めします。非 UTF-8 エンコードを使用して .SQL ファイルをアップロードすると、UTF-8 と互換性のないすべての文字が出力 DDL 内の置換文字「☒」に変換されます。

Tip: Linux の iconv ユーティリティまたは Windows の Liblconv ユーティリティを使用して、抽出されたファイルを UTF-8 形式に手動で変換できます。たとえば、データベースの文字セットが Latin-1 (ISO-8859-1) である場合、次のように、抽出されたファイルを UTF-8 形式に変換できます。

5.1.3 ALTER ステートメント

ALTER TABLE と ALTER TRIGGER を除き、Migration Portal は DDL 内の他の ALTER ステートメントを処理しません。

5.1.4 ユーザー、ロール、プロファイル、および付与

5.1.5 付与

Migration Portal は、抽出およびアップロードされたスキーマの Oracle ユーザーの評価と移行をサポートするようになりました。プロジェクトに関連付けられたスキーマのすべてのユーザー、ロール、プロファイルは、Migration Portal ビューの GLOBAL_OBJECTS 擬似スキーマの下にグループ化されます。現在、システム権限、オブジェクト権限、およびロールの付与は、ユーザーオブジェクトの DDL に含まれています。同様に、ロールに対するシステム権限とオブジェクト権限の付与は、ロールオブジェクトの DDL に含まれています。

現在、権限はユーザー オブジェクトとロールオブジェクトに付加されているため、ユーザーに権限を付与されている 1 つ以上のオブジェクトが失敗すると、ユーザーオブジェクト自体の評価が失敗します。この問題を解決するには、失敗したオブジェクトの許可ステートメントを、個々のステートメントに対して SQL 単一行コメント (--) を使用するか、連続する許可ステートメントに対して SQL 複数行コメント (/ ** /) を使用してコメントアウトしてから、オブジェクトを再評価します。実行される。失敗したグラントステートメントに対応する失敗したオブジェクトが修復された後、以前に失敗したグラントステートメントのコメントを解除して、ユーザーオブジェクトを再評価できます。

RESOURCE ロールの付与は、被付与者と同じ名前を共有するスキーマに対する CREATE 権限および USAGE 権限を付与することと同じです。RESOURCE ロールを付与する場合は、最初にユーザーまたはロールと同じ名前のスキーマを作成する必要があります。そうしないと、「スキーマが存在しません」エラーが発生します。

5.1.6 特権

スキーマをターゲット EDB Postgres Advanced Server データベースに移行するには、移行の実行に使用されるターゲットデータベース ユーザーに次の権限が必要です。

- プロファイルを移行するための SUPERUSER
- ユーザーとロールを移行するための SUPERUSER または CREATE ROLE

5.1.7 パスワード

ユーザーを移行する場合、Migration Portal はユーザーのパスワードを移行しません。ユーザーは、ターゲット EDB Postgres Advanced Server データベースにパスワードなしで作成されます。ユーザーにパスワードを設定することが必要な場合は、ユーザーの移行後にターゲットデータベースで手動で行うことができます。

5.1.8 プロファイル

Migration Portal は、スキーマ ユーザーに割り当てられた非 DEFAULT プロファイルを移行します。Oracle プロファイルには、パスワードとリソース制限の両方が含まれます。EDB Postgres Advanced Server は、Oracle パスワード関連の制限のみをサポートします。リソース制限はプロファイルで抽出されますが、パスワード制限のみが Migration Portal によってアップロード、評価、移行されます。また、Oracle データベースでは、DEFAULT プロファイルが新しい制限値で上書きされている可能性があります。DEFAULT プロファイルは、Migration Portal によって移行されません。Oracle DEFAULT プロファイルに適用されている制限オーバーライドを使用して EDB Postgres Advanced Server の DEFAULT プロファイルを構成する場合は、ALTER PROFILE ステートメントを使用してターゲットの EDB Postgres Advanced Server データベースに制限オーバーライドを手動で適用する必要があります。

注釈: バージョン 4.3.0 以降、Migration Portal はユーザー、ロール、プロファイル、および付与の評価をサポートします。ただし、Migration Portal の以前のバージョン (4.3.0 より前) を使用して作成されたプロジェクトの場合、この機能はサポートされません。ユーザー、ロール、プロファイル、権限を評価するには、最新バージョンの EDB DDL エクストラクタを使用して、新しいプロジェクトを作成します。

5.1.9 Oracle のデフォルトのケースを使用する

Oracle のデフォルトの大文字と小文字の区別機能を使用するオプションでは、Oracle データベースから EDB Postgres Advanced Server データベースにスキーマを移行するときに、すべてのデータベースオブジェクトに対して Oracle のデフォルトの大文字命名規則が保持されます。

EDB DDL エクストラクタおよび Oracle Data Pump ユーティリティは、PL/SQL 本体内で参照されるオブジェクト名を除き、Oracle のデフォルトの大文字 (つまり、オブジェクト名の大文字、または大文字と小文字が区別されるオブジェクト名のそれぞれの大文字) ですべての DDL を抽出します。DDL の作成時にユーザーが明示的に二重引用符で囲んでいない限り、オブジェクトは削除されます。

したがって、Migration Portal は、この機能のソース スキーマファイルのすべての二重引用符を保持し、さらに、Migration Portal は以下に二重引用符を適用します。

- ビューおよびマテリアライズド・ビューの SELECT ステートメント内の列名。
- チェック制約内の列名。
- トリガーヘッダー内のテーブル名と列名。

Oracle の大文字と小文字を使用するオプションが選択されていない場合、Migration Portal のデフォルトの動作では、すべてのオブジェクトが小文字 (EDB Postgres Advanced Server のデフォルトのケース)、または

引用符で囲まれた大文字と小文字が混合されたオブジェクト名を使用して指定されている場合は大文字と小文字が混合された状態で作成されます。ソース DDL 内。

useOraCase 機能を使用して、以下のソース DDL とターゲット DDL を検索します。

```
-- Source DDL
CREATE TABLE "RHS_SANITY"."DEPT"
(
  "DEPTNO" NUMBER(2,0),
  "DNAME" VARCHAR2(14),
  "LOC" VARCHAR2(14),
  CONSTRAINT "DEPTDEPTNO_PK" PRIMARY KEY ("DEPTNO")
USING INDEX ENABLE,
  CONSTRAINT "DEPTDNAME_UK" UNIQUE ("DNAME")
USING INDEX ENABLE
);

-- Target DDL
CREATE TABLE "RHS_SANITY"."DEPT"
(
  "DEPTNO" NUMBER(2,0),
  "DNAME" VARCHAR2(14),
  "LOC" VARCHAR2(14),
  CONSTRAINT "DEPTDEPTNO_PK" PRIMARY KEY ("DEPTNO")
USING INDEX ENABLE,
  CONSTRAINT "DEPTDNAME_UK" UNIQUE ("DNAME")
USING INDEX ENABLE
);
```

Oracle のデフォルトのケースを使用している場合、他のプロジェクトと比較して互換性率が低下する可能性があります。この互換性率の低下の一因となる可能性のある理由は次のとおりです。

- 最も一般的な理由は、オブジェクト/リレーションが存在しないことです。このシナリオは、Oracle のケースを維持するためにテーブル/オブジェクトが二重引用符で囲まれた大文字になっているのに対し、障害が発生したオブジェクトでは二重引用符なしで参照されていることが原因で発生します。例えば：

```
CREATE TABLE "HR"."EMP"
(
  "EMPNO" NUMBER(4,0),
  "ENAME" VARCHAR2(10),
  "JOB" CLOB,
  "MGR" NUMBER(4,0),
  "HIREDATE" DATE,
  "SAL" NUMBER(7,2),
  "COMM" NUMBER(7,2),
  "DEPTNO" NUMBER(2,0),
  CONSTRAINT "PK_EMP" PRIMARY KEY ("EMPNO")
USING INDEX ENABLE
);
```

(次のページに続く)

(前のページからの続き)

```

CREATE OR REPLACE PROCEDURE "HR"."PROC"
IS
  CURSOR emp_cur IS SELECT hiredate, CAST(hiredate AS timestamp(6)) AS timestamped_
  FROM emp;
  REC emp_cur%rowtype;
BEGIN
  OPEN emp_cur;
  LOOP
    FETCH emp_cur INTO rec;
    EXIT WHEN emp_cur%NOTFOUND ;
    DBMS_OUTPUT.PUT_LINE(rec.hiredate || ||rec.timestamped);
  END LOOP;
  CLOSE EMP_CUR;
END ;

```

エラーは次のように表示されます。

edb_notranlate_2 上の例では、テーブル名とその列名は、CREATE TABLE ステートメントで引用符で囲まれた大文字の名前で指定されています。ただし、CREATE OR REPLACE PROCEDURE ステートメントでは、テーブルとその列は引用符で囲まれていない小文字の識別子を使用して参照されます。

- %type を使用して定義された変数は、前の table.column-name が employees.employee_id%type のように小文字であるのに対し、実際のテーブルのテーブル名と列名は二重引用符で囲まれた大文字であるため、コードオブジェクトで失敗します。例えば：

```

CREATE TABLE "HR"."EMPLOYEES"
(
  "EMPLOYEE_ID" NUMBER(6,0),
  "FIRST_NAME" VARCHAR2(20),
  "LAST_NAME" VARCHAR2(25) CONSTRAINT "EMP_LAST_NAME_NN" NOT NULL ENABLE,
  "EMAIL" VARCHAR2(25) CONSTRAINT "EMP_EMAIL_NN" NOT NULL ENABLE,
  "PHONE_NUMBER" VARCHAR2(20),
  "HIRE_DATE" DATE CONSTRAINT "EMP_HIRE_DATE_NN" NOT NULL ENABLE,
  "JOB_ID" VARCHAR2(10) CONSTRAINT "EMP_JOB_NN" NOT NULL ENABLE,
  "SALARY" NUMBER(8,2),
  "COMMISSION_PCT" NUMBER(2,2),
  "MANAGER_ID" NUMBER(6,0),
  "DEPARTMENT_ID" NUMBER(4,0),
  CONSTRAINT "EMP_SALARY_MIN" CHECK ("SALARY" > 0) ENABLE,
  CONSTRAINT "EMP_EMP_ID_PK" PRIMARY KEY ("EMPLOYEE_ID")
  USING INDEX ENABLE,
  CONSTRAINT "EMP_EMAIL_UK" UNIQUE ("EMAIL")
  USING INDEX ENABLE
);
CREATE OR REPLACE PROCEDURE "HR"."PKG" AS
-- declare variables for data fetched from cursor

```

(次のページに続く)

(前のページからの続き)

```

empid      employees.employee_id%TYPE; -- variable for employee_id
hiredate   employees.hire_date%TYPE;   -- variable for hire_date
firstname  employees.first_name%TYPE;  -- variable for first_name
lastname   employees.last_name%TYPE;   -- variable for last_name
rowcount   NUMBER;
bonusamount NUMBER;
yearsworked NUMBER;
-- declare the cursor with a parameter
CURSOR cursor1 (thismonth NUMBER) IS
    SELECT employee_id, first_name, last_name, hire_date FROM employees
        WHERE EXTRACT(MONTH FROM hire_date) = thismonth;
BEGIN
    null;
END;
```

エラーは次のように表示されます。

```
syntax error at or near "%" (line 3, char 37)
```

- テーブルの列データ型に TYPE 参照がある場合、大文字と小文字の違いにより失敗します。例えば：

```

CREATE OR REPLACE TYPE "HR"."USER_TYPE" IS OBJECT(
    "ID" NUMBER,
    "NAME" VARCHAR2(256)
);

CREATE TABLE "HR"."MY_TABLE"
(
    "ID" NUMBER,
    "MY_NAME" user_type
);
```

エラーは次のように表示されます。

```

type "user_type" does not exist
LINE 4:      "MY_NAME" user_type
              ^
SQL state: 42704
Character: 58
```

5.2 EDB DDL エクストラクタ

- EDB DDL エクストラクタスクリプトは、Flashback を使用して復元された、BIN\$b54+4X1EYwPgUAB/AQBWwA==\$0 のような名前がまだ残っているオブジェクトを抽出しません。これらのオブジェクトを抽出する場合は、オブジェクトの名前を変更し、抽出プロセスを再実行する必要があります。
- EDB DDL エクストラクタ は、`nologging` テーブルを通常のテーブルとして抽出します。これらのテーブルが EDB Postgres Advanced Server に移行されると、WAL ログファイルが作成されます。
- EDB DDL エクストラクタ は、`VALID` ステータスのオブジェクトのみを抽出します。Migration Portal で評価する必要がある `INVALID` ステータスを持つオブジェクトについては、まずそれらを `VALID` に更新します。
- EDB DDL エクストラクタ は、Oracle ラップ機能を使用して難読化されたオブジェクトを抽出しません。そのため、これらのオブジェクトは、Migration Portal によって評価される DDL のセットには含まれません。これらのオブジェクトを評価して EDB Postgres Advanced Server に移行する場合は、オブジェクトのラップされたバージョンをラップされていないバージョンに置き換えます。詳細については、[ラップされたオブジェクト](#)を参照してください。
- EDB DDL エクストラクタは、スキーマ名とその依存関係情報を保存するグローバル一時テーブルを作成します。これらのテーブルは、抽出が成功すると削除されます。
- EDB DDL エクストラクタスクリプトは、名前が `PG_` で始まるスキーマを抽出しません。これは、PostgreSQL がサポートしていないためです。これらのスキーマを抽出する場合は、抽出前にスキーマの名前を変更する必要があります。
- EDB DDL エクストラクタは、プロファイル、ロール、および付与の情報を自動的に抽出します。
- EDB DDL エクストラクタ は現在、Oracle 11g からの `ROLES`、`SYSTEM GRANTS ON ROLES`、`OBJECT GRANTS ON ROLES`、および `ROLE GRANTS` の抽出をサポートしていません。この動作により、抽出されたファイルのこれらのオブジェクトタイプに対応するセクションにエラーメッセージが書き込まれます。これらのエラーは、Migration Portal によるこれらのファイルの評価に問題を引き起こすことはありません。

5.3 Oracle データ ポンプ ユーティリティ

- Oracle の引用演算子 (`IDENTIFIED BY VALUES q'[:1]'` など) を含む `IDENTIFIED BY` 句を使用してデータベースリンクを作成する場合、Migration Portal は SQL ファイルの解析に失敗する可能性があります。ファイルを正常に解析するには、`IDENTIFIED BY my_password` などの実際のパスワードを使用してみてください。
- Oracle Data Pump ユーティリティによって生成された DDL には、Migration Portal によって処理されない `ALTER FUNCTION`、`ALTER PACKAGE` および `ALTER TYPE` などの `ALTER STATEMENTS` が含まれる場合があります。
- Oracle Data Pump がスキーマ・モードで実行される場合、つまり、`expdp` コマンドの実行時に `SCHEMAS` パラメータが使用される場合、プロファイルとロールは抽出されません。エクスポートされるスキーマユーザーにプロファイルが割り当てられている場合、そのプロファイルはエクスポートされた `impdp` ファイルの一部ではないため、Migration Portal でのスキーマユーザーオブジェクトの評価は失敗し

ます。この問題を解決するには、ユーザーオブジェクトのターゲット DDL でプロファイルの割り当てを削除し、ユーザー作成オブジェクトを再評価します。

- スキーマモードで `impdp` または `expdp` を使用して Oracle 11g から `ROLES`、`SYSTEM GRANTS ON ROLES`、`OBJECT GRANTS ON ROLES`、および `ROLE GRANTS` を抽出しようとする、エラーが発生する場合があります。回避策は、`full=y` オプションを使用することです。これにより、すべてのスキーマとこれらのオブジェクトタイプがエラーなく抽出されます。