



EDB

Migration Portal

リリース 4.3.0

EnterpriseDB Corporation

2022 年 12 月 12 日

目次

第 1 章	Release notes	3
1.1	Version 4.3	3
1.2	Version 4.2	4
1.3	Version 4.1	5
1.4	Version 4.0	6
第 2 章	Known issues, limitations, and notes	9
2.1	Migration Portal	9
2.1.1	ラップされたオブジェクト	9
2.1.2	ファイルエンコーディング	9
2.1.3	ALTER ステートメント	10
2.1.4	ユーザー、ロール、プロファイル、および付与	10
2.1.5	Grant	10
2.1.6	特権	10
2.1.7	パスワード	10
2.1.8	プロファイル	11
2.2	EDB DDL エクストラクタ	11
2.3	Oracle Data Pump ユーティリティ	12
第 3 章	Supported platforms	13
3.1	対応ブラウザ	13
第 4 章	Using Migration Portal	15
4.1	Overview of the Migration Portal home page	16
4.2	Overview of the Migration Portal Projects page	17
4.3	Migration Portal quick start	18
第 5 章	Migrating a database	21
5.1	Performing a schema extraction	21
5.1.1	EDB DDL エクストラクタを使用したスキーマの抽出	21
5.1.2	スキーマとオブジェクトのサポート	24
5.2	Performing a schema assessment	28
5.2.1	スキーマ評価を実行する	28
5.2.2	評価レポートを生成する	32
5.2.3	評価レポートを評価する	32
5.3	Performing a schema migration	34
5.3.1	オフライン移行	34
5.3.2	オンライン移行	36
5.4	Performing a data migration	39
5.4.1	高度なデータ移行	39

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
- シークエンス
- 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
- 型と型体
- トリガー
- 機能
- 手続き
- パッケージ
- ユーザー
- ロール
- プロフィール
- ロール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。

使いやすい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 を起動します。

EnterpriseDB は、企業が既存のデータベースシステムを長年 Postgres に移行するのを支援してきました。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 のドキュメントには、 Migration Portal 4.0 の最新バージョンが説明されています。

Version	Release date
<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 4.3

Migration Portal 4.3 の新機能、機能強化、バグ修正、およびその他の変更には次のものが含まれます。

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ています。
Enhancement	Migration Portal が更新され、ユーザー、ロール、プロファイル、および許可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 エクストラクタで生成された 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の情報は、これらの古いバージョンでは抽出できません。
Enhancement	Migration Portal のスキーマ移行の複雑さのスコアリングと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 バージョンに基づいたスコアと機能分析を提供します。
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 のユーザーインターフェイスが簡素化され、手順数と冗長オプションが削減されたオフラインとオンラインの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、以前はキャプチャされなかった Oracle 機能の存在をキャプチャしてレポートします。 <i>CONNECT BY CONDITION</i> 、 <i>CONNECT BY NOCYCLE</i> 、およびビルトイン/SYS 所有のファンクション <i>GROUP_ID</i> 。

次のページに続く

表 2 – 前のページからの続き

Type	Description
Security fix	Migration Portal で使用される jackson-databind ライブラリが更新され、`CVE-2022-42003` < https://nvd.nist.gov/vuln/detail/CVE-2022-42003 >` および `CVE-2022-42004` < <a href="https://nvd.nist.gov/vuln/detail/CVE-2022-42004">https://nvd.nist.gov/vuln/detail/CVE-2022-42004</a> >` で特定されたセキュリティの脆弱性に対処しました。
Bug fix	シーケンスの `MAXVALUE` の処理を修正して、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ている `MAXVALUE` に許可される最大値に関する問題を解決しました。
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 で見つかった仮想テーブルの列に関連する `ALTER TABLE MODIFY` ステートメントの処理の問題を修正しました。
Bug fix	句がパッケージ関数定義内で使用されているときに `PIPELINED` 句機能が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および報告されない問題を修正しました。

1.2 Version 4.2

Migration Portal 4.2 の新機能、機能強化、バグ修正、およびその他の変更には次のものが含まれます。

Type	Description
Enhancement	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スキーマ移行の複雑さスコアを計算し、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するために、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、未解決のエラーが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」テーブルを追加しました。
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	互換性のないビルトインパッケージと SYS スキーマ TYPE オブジェクトの使用をキャプチャしてより正確にレポートするように Migration Portal を修正しました。
Bug Fix	次の Oracle 機能の存在をより正確にキャプチャし、スキーマ移行の複雑さへの影響を評価するように Migration Portal を修正しました。
Bug Fix	DDL エクストラクタの新しいバージョンを作成して、CACHE キーワードで定義された Oracle テーブルのエントリが DDL ファイルに出力される問題を修正しました。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ました。Migration Portal は、古いバージョンの DDL エクストラクタで生成された DDL を引き続き処理および評価できます。

次のページに続く

表 3 – 前のページからの続き

Type	Description
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	「機能分析」セクションで LONG RAW 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、ユーザー定義の例外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	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 句の回避策のステータスを「None / Needs Investigation」から「Exists」に更新しました。現在、修復ハンドラーが利用可能で、REGEXP_LIKE 句に自動的に適用されます。
Bug Fix	EDB Postgres Advanced Server でサポートされているおよびサポートされていない Oracle の事前定義の 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（スキーマ、テーブル、ビューなど）のサブセットを選択し、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でのユーザー選択状態が失われる問題を修正しました。

1.3 Version 4.1

Migration Portal 4.1 の新機能、機能強化、バグ修正、およびその他の変更には次のものが含まれます。

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 コンストラクト（つまり、機能）に関連するエフォートタイプと複雑さレベルに基づいて移行の複雑さスコアを提供する新しい機能分析セクションが含まれています。この新しいセクションでは、DDL にある互換性のない機能の概要も提供します。Feature Analysis セクションは現在、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に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、ダウンロード用に作成される単一の DDL SQL ファイルの代わりに、選択したオブジェクトタイプごとに個別のファイルと、プロジェクトと選択したスキーマの統合スクリプトを含む zip アーカイブが作成されます。
Enhancement	Q の引用符で囲まれた文字列をドルまたは E の引用符で囲まれた文字列に変換する新しい修復ハンドラー ERH-3005 が追加されました。

次のページに続く

表 4 – 前のページからの続き

Type	Description
Enhancement	パッケージヘッダーレベルでパイプライン化された特定の変換を適用するように ERH-2078 修復ハンドラーを強化しました。
Enhancement	パッケージ内で提供される TYPE 定義から NULL を削除するように ERH-2034 修復ハンドラーを更新しました。
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、 <code>async</code> 、 <code>eventsources</code> 、 <code>h2</code> 、 <code>jsdom</code> 、および <code>moment</code> 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一部のインデックスがスキップされるコーナー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た CREATE TABLE ステートメントの DEDUPLICATE LOB 句により、DDL 解析関連のエラーが発生した問題を修正しました。

1.4 Version 4.0

Migration Portal 4.0 の新機能、機能強化、バグ修正、およびその他の変更には次のものが含まれます。

Type	Description
Enhancement	Migration Portal は、EDB DDL エクストラクタの使用を再度サポートするようになりました。DDL エクストラクタに関連する問題が解決され、DDL エクストラクタファイル形式の使用に対して導入された一時的な制限が削除されました。Migration Portal は、 <code>`documentation` <https://www.enterprisedb.com/docs/migration_portal/latest/04_mp_migrating_database/01_mp_schema_extraction/>`</code> 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削減するように最適化されています。
Enhancement	EDB Postgres Advanced Server は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 および 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した。

次のページに続く

表 5 – 前のページからの続き

Type	Description
Bug fix	文法規則を修正して、修復ハンドラーの実行をスキップして EDITIONABLE キーワードを削除する解析エラーを解決しました。
Bug fix	Oracle PIPELINED および PIPE ROW ファンクションコール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い問題を修正しました。
Bug fix	新しいユーザーに対して Migration Portal のランディングページを初めて開いたときにスタート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アップロードされた DDL ファイルは、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 以外のファイルをアップロードした場合、UTF-8 非対応の文字はすべて置換文字「☒」に変換され、問題なく評価が続行されます。
Bug fix	ソース Oracle DDL の名前のない制約の制約名の割り当てを修正しました。
Bug fix	修復ハンドラーを更新して、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しました。
Bug fix	COLLATE および 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	単一行のコメントの一部として表示される複数行の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。
Bug fix	複数のプロジェクトを同時に実行しているときに ConcurrentModificationException が発生する問題を修正しました。
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	特定の順序が必要ないため、一部の句の順序を無視するように create table 文法をいくつか修正しました。

第2章 Known issues, limitations, and notes

このページには、次の既知の問題、制限、および注意事項がリストされています。

- *Migration Portal*
- *EDB DDL* エクストラクタ
- *Oracle Data Pump* ユーティリティ

2.1 Migration Portal

2.1.1 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 に含める場合、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含まれません。ラップされたオブジェクトを評価して EDB Postgres Advanced Server に移行する場合は、Migration Portal にアップロードする DDL ファイルにオブジェクトのアンラップバージョンを含めます。これを行う推奨される方法は、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。

2.1.2 ファイルエンコーディング

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 エンコーディングの .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 で置換文字「`☒`」に変換されます。

Tip: Linux の `iconv` ユーティリティまたは Windows の `LibIconv` ユーティリティを使用して、抽出したファイルを UTF-8 形式に手動で変換できます。たとえば、データベースのキャラクタセットが Latin-1 (ISO-8859-1) の場合、次のように抽出したファイルを UTF-8 形式に変換できます。

```
iconv -f iso-8859-1 -t UTF-8 sample.sql > sample_utf8.sql
```

2.1.3 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。

2.1.4 ユーザー、ロール、プロファイル、および付与

2.1.5 Grant

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、およびプロファイルは、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、ロールへの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。

付与は現在ユーザーオブジェクトとロールオブジェクトにアタッチされているため、ユーザーに権限が付与されている 1 つ以上のオブジェクトに障害が発生すると、ユーザーオブジェクト自身自身が評価に失敗します。この問題を解決するには、個々のステートメントに SQL 単一行コメント (--) を使用するか、連続した grant ステートメントに SQL 複数行コメント (/* */) を使用して、失敗したオブジェクトの grant ステートメントをコメントアウトし、オブジェクトを再評価できます。実行されます。失敗した付与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、以前に失敗した付与ステートメントのコメントを外し、ユーザーオブジェクトを再評価できます。

2.1.6 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てい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 SUPERUSER
- ユーザーとロールを移行する SUPERUSER または CREATE ROLE

2.1.7 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、パスワードなしでターゲット EDB Postgres Advanced Server データベースに作成されます。ユーザーにパスワードを構成することが望ましい場合、または必要な場合は、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。

2.1.8 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の両方の制限が含まれます。EDB Postgres Advanced Server は、Oracle パスワード関連の制限のみをサポートしています。リソース制限はプロファイルとともに抽出されますが、パスワード制限のみが Migration Portal によってアップロード、評価、および移行されます。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オーバーライド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。EDB Postgres Advanced Server の **DEFAULT** プロファイルを、Oracle **DEFAULT** プロファイルに適用された制限のオーバーライドで構成する必要がある場合は、**ALTER PROFILE** ステートメントを使用してターゲットの EDB Postgres Advanced Server データベースに手動で適用する必要があります。

2.2 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オブジェクトで、まだ **BIN\$b54+4X1EYwPgUAB/AQBWwA==\$0** のような名前が付いているものを抽出しません。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て、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、通常のテーブルとして **nologging** テーブルを抽出します。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、ステータスが **VALID** のオブジェクトのみを抽出します。Migration Portal で評価する **INVALID** ステータスのオブジェクトについては、最初に **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情報を保存するための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、PostgreSQL がサポートしていないため、名前が **PG_** で始まるスキーマを抽出しません。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。

2.3 Oracle Data Pump ユーティリティ

- Oracle の引用演算子で `IDENTIFIED BY` 句を使用してデータベースリンクを作成すると、Migration Portal が SQL ファイルの解析に失敗する場合があります。たとえば、`IDENTIFIED BY VALUES q'[:1]'`。ファイルを正常に解析するには、実際のパスワードを使用してみてください。たとえば、`IDENTIFIED BY my_password`。
- Oracle Data Pump ユーティリティによって生成された DDL には、Migration Portal で処理されない `ALTER FUNCTION`、`ALTER PACKAGE`、`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ている場合、つまり `expdp` コマンドの実行時に `SCHEMAS` パラメーターが使用されている場合、プロファイルとロールが抽出されません。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、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。

第3章 Supported platforms

Migration Portal は、Oracle 11g、12c、18c、および 19c から EDB Postgres Advanced Server、または 14 への評価と移行をサポートしています。Migration Portal は次のブラウザーでサポートされています。

3.1 対応ブラウザ

最適なユーザー エクスペリエンスを得るには、Google Chrome ブラウザーの使用をお勧めします。Migration Portal は次のブラウザー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4章 Using Migration Portal

Migration Portal を使用すると、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。評価のためにスキーマをアップロードし、すぐにフィードバックと提案を得ることができます。この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で EDB Postgres データベースを作成できます。

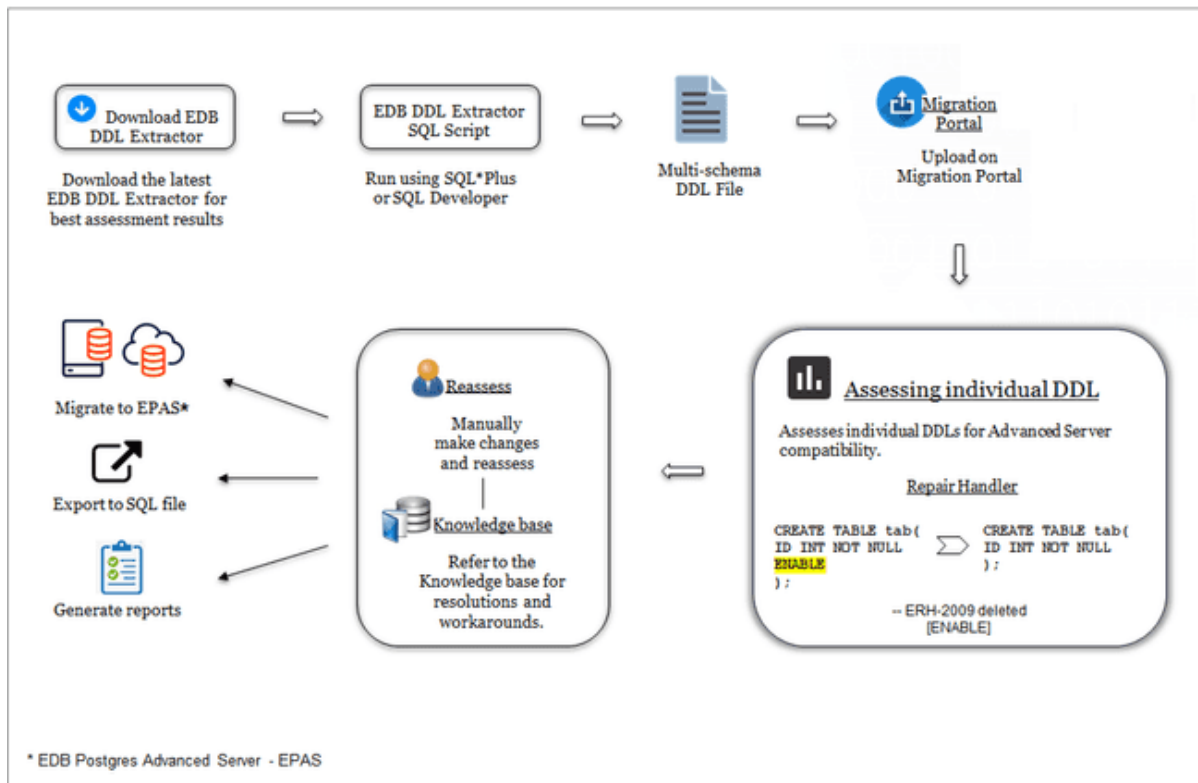


図 1: The assessment and migration process

注釈: Migration Portal には現在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. ブラウザーを開き、<https://www.enterprisedb.com/> に移動します。
2. EnterpriseDB ホームページで、**PostgreSQLSoftware>Migration** を選択します。
3. Migration Portal ページで、**Findoutmore>OpenMigrationPortal** 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

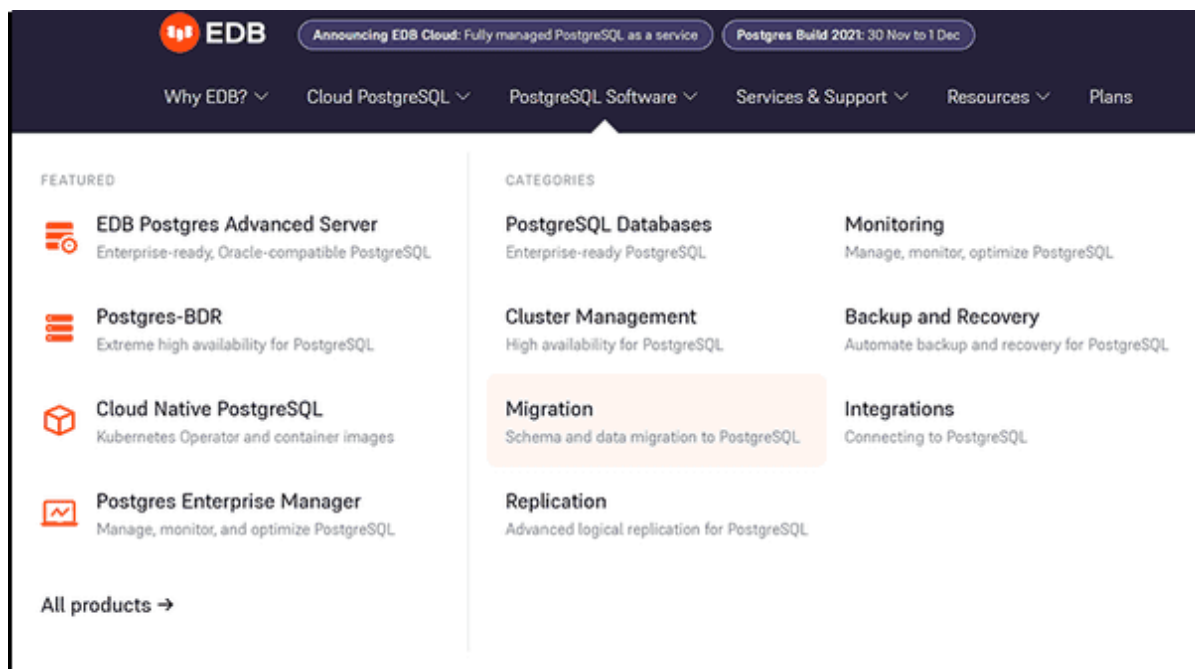


図 2: Migration Portal on the EnterpriseDB home page

ついては、

Migration Portal quick start を参照してください。

4.1 Overview of the Migration Portal home page

Migration Portal ホームページから、移行ツールにすばやくアクセスできます。

Migration Portal ホームページでは、次の Migration Portal 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**：+New を選択してプロジェクトを作成します。
- **プロジェクトを検索する**：検索ボックスを使用してプロジェクトを検索します。
- **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、スキーマ評価後に互換性の割合が表示されます。
- **Migrateto**：Migrateto 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、スキーマをオンプレミスまたはクラウド内の EDB Postgres Advanced Server に移行します。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**：削除を選択して、選択したプロジェクトを削除します。
- **警告記号**：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、警告メッセージが表示されます。

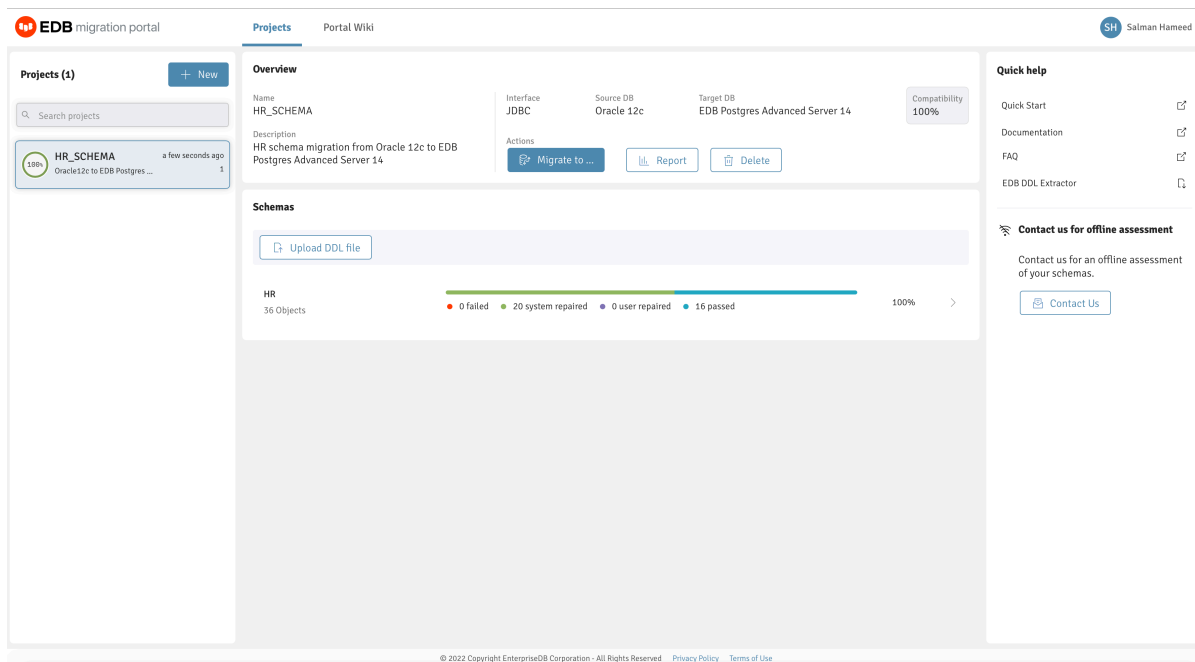


図 3: The Migration Portal home page.

- **DDL ファイルのアップロード**：新しいスキーマのアップロードを選択して、新しい DDL ファイルをアップロードします。
- **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**：ポータル Wiki を選択して、製品情報やその他のヘルプガイドにアクセスします。

4.2 Overview of the Migration Portal Projects page

Migration Portal 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の割合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。
- **Migrateto**：Migrateto を選択して、スキーマをオンプレミスまたはクラウド内の EDB Postgres Advanced Server に移行します。
- **オブジェクトを検索する**：検索ボックスを使用してオブジェクトを検索します。
- **フィルター**：プロジェクトページの左側のパネルから、システムで修復されたオブジェクトと手動で修復されたオブジェクトをフィルタリングできます。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。
- **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。

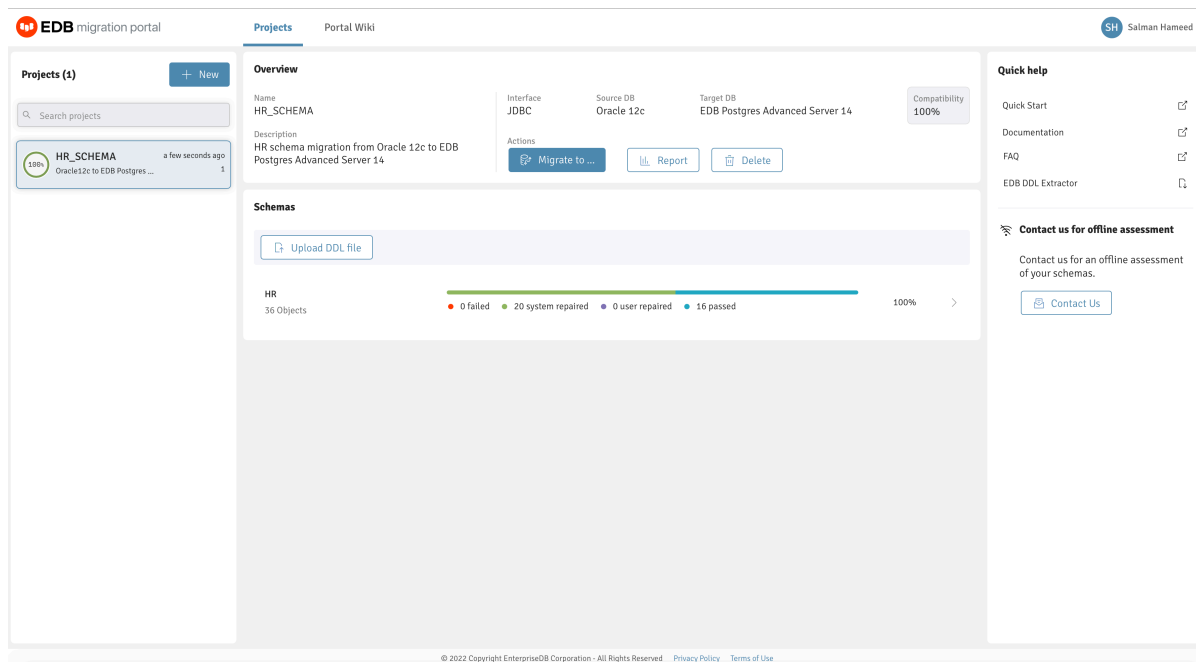


図 4: The Migration Portal Projects Page overview.

- **一般的な障害**：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障害の`.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**：結果セットにカーソルを合わせると、成功、失敗、および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、特定の情報につい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ます。

4.3 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。
1. EDB 資格情報で [Migration Portal](#) にログインします。
 1. 新しいプロジェクトを作成します。
1. 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、ブラウザーがフリーズまたはクラッシュする場合があります（ブラウザーと使用中の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、別のブラウザーで Migration Portal を実行するか、より少ないスキーマを抽出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、修復アクションを実行して EDB Postgres Advanced Server との互換性を向上させ、手動介入が必要な残りのエラーにフラグを付けます。
 2. DDL オブジェクト（100 %の成功率を示していないテーブルなど）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 で互換性のないオブジェクトの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。
1. スキーマのエラーを解決した後、変更した 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5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。

5.1 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 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*（推奨方法）
- *Oracle Data Pump ユーティリティ*

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

5.1.1 EDB DDL エクストラクタを使用したスキーマの抽出

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

注釈: システムとブラウザー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 Migration Portal が応答しなくなる場合があります。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ます。

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 ビルトイン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、EDB Postgres Advanced Server の互換性について分析された DDL ファイルを作成します。

注釈: `CONNECT` および `'SELECT_CATALOG_ROLE'` ロールと `'CREATE TABLE'` 特権が必要です。

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、コンマ（`,`）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言及しないでください。ただし、抽出するスキーマ名に言及することをお勧めします。

4. プロンプトが表示されたら、エクストラクターが抽出した DDL を保存する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** と入力します。

例：

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. コンマで区切って 3 つのスキーマを指定します。

```
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、次を使用できます。

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

同様に Windows の場合：

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

5. yes と入力して、他のスキーマ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

SQL Developer 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たら、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とおりに必要な特権を持っている場合、**はい**を選択してスクリプトの実行を続行します。

2. スキーマのコ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** と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、依存オブジェクトを抽出しません。

DDL エクストラクタの実行の出力が [Script Output] 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 の出力が入力した場所（たとえば、c:\Users\Example\Desktop\）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、[ファイルエンコーディング](#) を参照してください。

5.1.2 スキーマとオブジェクトのサポート

以下のリストと表は、Migration Portal でサポートされているおよび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
DBSFUSER	DEMO	DIP	DMSYS	DVF
DVSY	EXFSYS	FLows_FILES	FLows_020100	FRANCK
GGSYS	GSMAD-MIN_INTERNAL	GSMCATUSER	GSMROOTUSER	GSMUSER
LBACSYS	MDDATA	MDSYS	MGMT_VIEW	OJMSYS
OLAPSYS	ORDPLUGINS	ORDSYS	ORDDATA	OUTLN
ORACLE_OCM	OWBSYS	OWBYSS_AUDIT	PDBADMIN	RMAN
RE-MOTE_SCHEDULER_AGENT	SI_INFORMTN_SCHEMA	TIAL_CSW_ADMIN_USR	SPA-USER_WFS_ADMIN_USR	SQLTXADMIN
SQLTXPLAIN	SY\$UMF	SYS	SYSBACKUP	SYSDG
SYSKM	SYSRAC	SYSTEM	SYSMAN	TSMSYS
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ていません。

サポートされているオブジェクトタイプ

- テーブル
- シークエンス
- 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

- 型と型体
- トリガー
- 機能
- 手続き
- パッケージ
- ユーザー
- ロール
- プロフィール
- ロールとオブジェクトの付与

注釈: 列、テーブル、およびマテリアライズドビューの COMMENTS 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
- スケジューラー
- マテリアライズドビューの LOB インデックスとインデックス
- XML スキーマ
- 表領域
- ディレクトリ
- RLS ポリシー
- キュー*
- 図書館
- 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ていますが、Migration Portal は現在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の 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

- 自分以外の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、自分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ための十分な表領域クォータがあることを確認します。
- 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、Data Pump ユーティリティを実行しているユーザーに読み書き権限を付与します。

```
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する
- データベース内のすべてのスキーマを抽出する

### データベース内の 1 つ以上のスキーマを抽出する

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

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の種類を指定します。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タイプ** 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、[ファイルエンコーディング](#) 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する

**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ある場合、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（たとえば、export.par）を持つファイルを作成します。ファイルに属性と値を追加します。

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の種類を指定します。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タイプ](#) 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（たとえば、import.par）を持つパラメーターファイルを作成します。ファイルに属性と値を追加します。

```
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,
```

(次のページに続く)

(前のページからの続き)

```
SPATIAL_CSW_ADMIN_USR, SPATIAL_WFS_ADMIN_USR, SQLTXADMIN, SQLTXPLAIN, SYSSUMF, SYS,
↳ SYSBACKUP, SYSDG, SYSKM,
SYSRAC, SYSTEM, SYSMAN, TMSYS, 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、Data Pump 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めないことを指定します。それらを SQL ファイルに含めないことで、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、インポート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 で想定されるファイルエンコーディング形式については、

[ファイルエンコーディング](#) を参照してください。

## 5.2 Performing a schema assessment

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

### 5.2.1 スキーマ評価を実行する

1. *Migration Portal* に移動します。
2. EDB 資格情報を入力します。
3. **New** 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。 - ソースのソース DB とバージョンを選択します。 - ターゲットのターゲット DB と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**）と制約（**\_CON**）へのサフィックスの追加を指定するには、 **サフィックスを追加してオブジェクト名を一意にする** チェックボックスをオンにします。 Oracle がサ

## 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 Advance...

Version

13

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

No file chosen

 Choose file

☐ Add suffix to make object names unique 

Description(Optional)

Say something about the project...

200 characters left

 Cancel

 Create & assess

図 1: The Migration Portal New project dialog

ポートしているのに対して、EDB Postgres Advanced Server はテーブルとインデックスに同じ名前をサポートしていないため、この選択により、より良い評価結果が保証されます。

8. 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コンストラクトを確認し、修復アクションを実行して、EDB Postgres Advanced Server との互換性を向上させます。手動介入が必要な残りのエラーにフラグを付けます。

9. 100%の成功率を示していない DDL オブジェクト（TABLES など）を確認します。

10. 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。中央のパネルの下部に、修復アクションの詳細を表示できます。

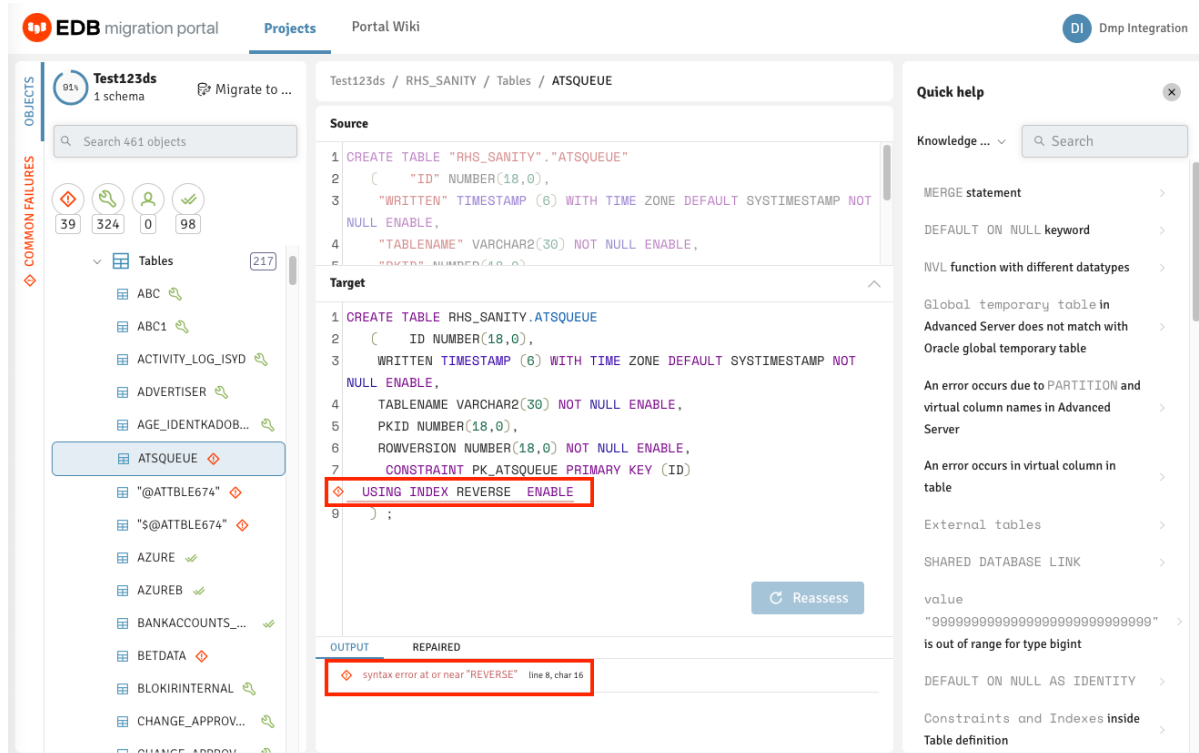


図 2: Incompatible objects are identified

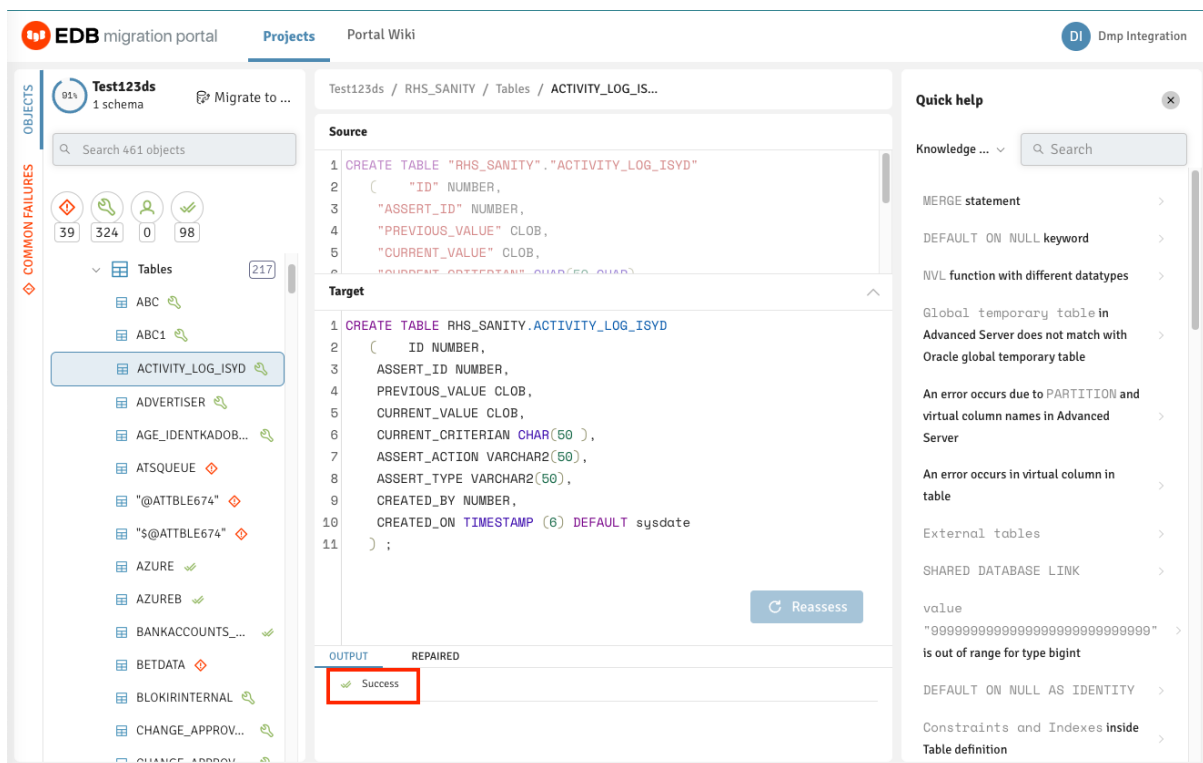
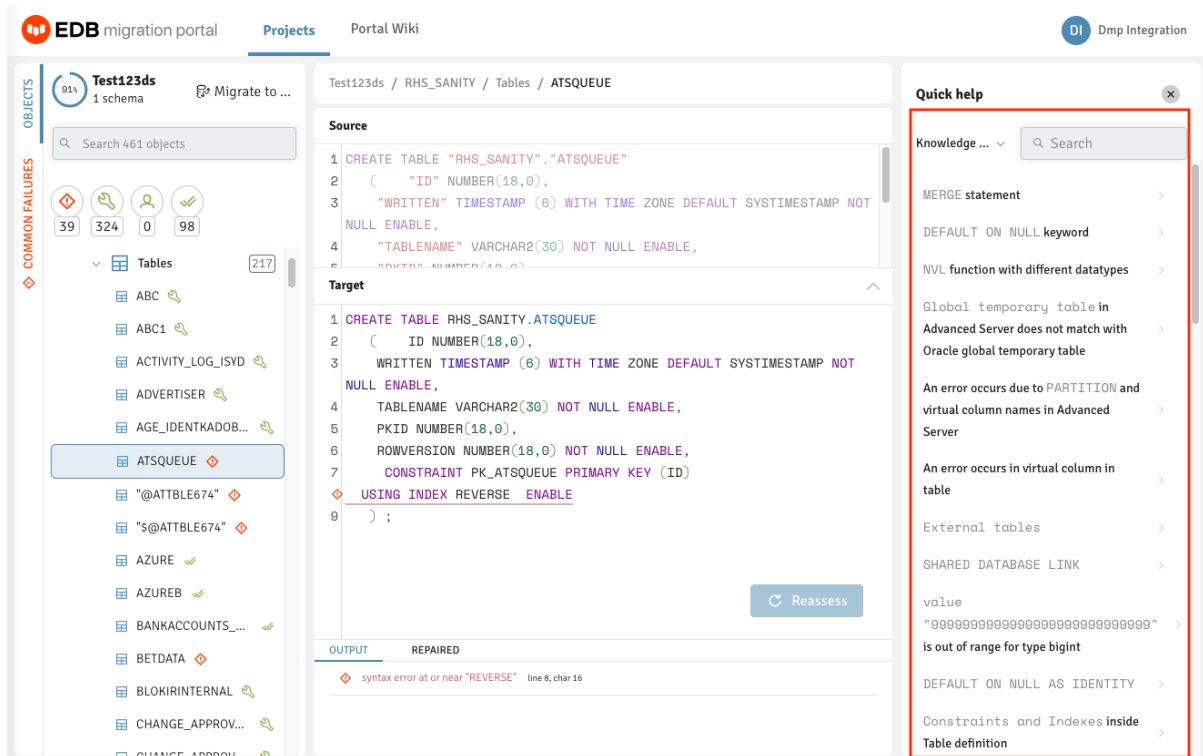
11. 右側のパネルのナレッジベース情報を参照して、EDB Postgres Advanced Server とすぐに互換性がないオブジェクトの回避策を見つけます。Portal wiki ページでナレッジベース情報を表示することもできます。

12. 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のエラーメッセージを入力し、検索を選択できます。

オブジェクトの詳細パネルには、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。

**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、互換性のないすべてのオブジェクトを互換性のあるものにすることができます。



---

**注釈:** 互換性のないオブジェクトの情報または回避策がナレッジベースにない場合は、サポートチームにお問い合わせください。

---

DDL ででの作業が終了したら、変更した 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。

### 5.2.2 評価レポートを生成する

Migration Portal のレポートは、プロジェクト用に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、このレポートは、障害が発生したオブジェクトと障害の原因に関する詳細を提供します。

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、**Generate** を選択します。

オプションで **GeneratePDF** 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 ラップされたオブジェクトに隠されている機密性の高い 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含まれないため、互換性割合の実際の値は評価レポートで計算された値と異なる場合があります。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。

---

### 5.2.3 評価レポートを評価する

評価レポートは、スキーマおよびプロジェクト全体のオブジェクトと機能の互換性に関するデータと情報を示します。

- **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で使用する Oracle 句またはコンストラクト。機能は、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、**非互換機能**と**未解決のオブジェクト**のテーブルがリストされています。

## オブジェクトの互換性

ターゲットデータベースで正常に実行された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てオブジェクトの互換性の概要が表示されます。

| Object compatibility level | 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 と自動的に互換性を持たせます。オブジェクトの動作に予期しない影響が発生した場合に備えて、レポートの**修復ハンドラーが適用されました**セクションの説明を注意深く確認してください。 |
| User repaired              | ユーザーが手動で修復および再評価した後、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:** 提示された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います。これは、オブジェクトが更新および再評価された場合に变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

EDB Postgres Advanced Server では、一部の機能をネイティブに利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示されます。

| Feature compatibility level                                    | 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:** 回避策のない機能の場合、アプリケーションレベルの変更により機能の非互換性に対処できる場合があります。#### 制限事項

評価レポートは、ユーザーがターゲット DDL を手動で変更し、**再評価**を選択すると更新されます。ただし、レポートの**互換性のない機能**の表は更新されず、ソース DDL 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て再評価された後、オブジェクトは、それらのオブジェクトで最初に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。

## 5.3 Performing a schema migration

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

Migration Portal には、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用意されています。

- 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移行の両方に適しています。
- 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、スキーマを自動的に作成します。通常、このオプションはクラウドで実行されているデータベース (BigAnimal クラスターまたはセルフマネージドインスタンス) への移行に最適です。

スキーマを移行する前に、EDB は、*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。

### 5.3.1 オフライン移行

クライアントを使用して DDL SQL ファイルを実行し、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. Select one or more schemas to migrate to EDB Postgres Advanced Server. You can also expand a schema and select specific object types under each schema.

Generate and download DDL SQL scripts for offline migration of schemas



1 Select Schemas

2 Download Schema

3 Finish

|                               |                   |                         |                                              |                       |
|-------------------------------|-------------------|-------------------------|----------------------------------------------|-----------------------|
| Project Name<br>TestMigration | Interface<br>JDBC | Source DB<br>Oracle 18c | Target DB<br>EDB Postgres Advanced Server 14 | Compatibility<br>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:** スキーマに 100%の互換性がない場合は、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入力します。

1. **DownloadSQLfile** 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ています。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。

| 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。 | <b>AllTestMigration_ALL_OBJECTS.sql</b><br>consists of DDLs for all objects of schemas OE, SH, IX, and PM, and only selected objects of schema HR. |

次のページに続く

表 4 – 前のページからの続き

| File name                              | Description                                  | Example                                                                       |
|----------------------------------------|----------------------------------------------|-------------------------------------------------------------------------------|
| <SchemaName>.<br>.sql                  | 特定のスキーマ内の選択されたオブジェクトの DDL で構成される 1 つ以上のファイル。 | <b>HR.sql</b> consists of DDLs for all the selected objects of the HR schema. |
| <SchemaName>#<br><ObjectType>.<br>.sql | それぞれが選択した単一のオブジェクトの DDL で構成される 1 つ以上のファイル。   | <b>HR_TABLE.sql</b> 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] を選択してメインの Migration Portal ページに戻ります。
1. 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。
  1. 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の EDB Postgres Advanced Server データベースに接続します。
1. 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:** または、インポートに `pgAdmin` クライアントを使用できます。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。

### 5.3.2 オンライン移行

ターゲット 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. Select one or more schemas to migrate to EDB Postgres Advanced Server. You can also expand a schema and select specific object types under each schema.

## Connect to database and perform online migration of schemas



1 Select Schemas
2 Connect
3 Deploy
4 Finish

|                               |                   |                         |                                              |                       |
|-------------------------------|-------------------|-------------------------|----------------------------------------------|-----------------------|
| Project Name<br>TestMigration | Interface<br>JDBC | Source DB<br>Oracle 18c | Target DB<br>EDB Postgres Advanced Server 14 | Compatibility<br>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 ページで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ら、**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表示されます。**DownloadSummary** ボタンを選択して、移行の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。

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

### 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%          |

#### 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

## 5.4 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 の使用の詳細については、 Migration Toolkit を参照してください。
2. Migration Toolkit `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使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. Migration Toolkit を `-dataOnly` モードで起動します。 `-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:** データをロードする前にテーブルが切り捨てられます。

---

### 5.4.1 高度なデータ移行

並列データのロードが必要な大規模なデータベースの場合、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 データが含まれる場合）

詳細については、 Migration Toolkit を参照してください。