



Advanced Server Upgrade Guide

リリース *14.1.0*

EnterpriseDB

2021 年 12 月 18 日

目次

第 1 章 Supported Platforms	3
第 2 章 Limitations	5
第 3 章 Upgrading an Installation With pg_upgrade	7
3.1 Performing an Upgrade	8
3.1.1 Linking versus Copying	8
3.2 Invoking pg_upgrade	9
3.2.1 Command Line Options - Reference	10
3.3 Upgrading to Advanced Server 14	12
3.4 Upgrading a pgAgent Installation	19
3.5 pg_upgrade Troubleshooting	20
3.5.1 アップグレードエラー-クラスターにサービスを提供しているポストマスターがあるようです	20
3.5.2 アップグレードエラー-fe_sendauth: パスワードが指定されていません	20
3.5.3 アップグレードエラー-新しいクラスターは空ではありません。出る	20
3.5.4 アップグレードエラー-ライブラリのロードに失敗しました	20
3.6 Reverting to the Old Cluster	21
第 4 章 Performing a Minor Version Update of an RPM Installation	23
第 5 章 Using StackBuilder Plus to Perform a Minor Version Update	25

- EDB Postgres Advanced Server アップグレードガイド*は、EDB Postgres Advanced Server (Advanced Server) のアップグレード処理に関する包括的なガイドです。このガイドでは、次の使用に関する詳細情報を確認できます。
- `pg_upgrade`。以前のバージョンの Advanced Server から Advanced Server 14 にアップグレードします。
- `yum` は、Linux ホストでマイナーバージョンアップグレードを実行します。
- `StackBuilder Plus` は、Windows ホストでマイナーバージョンアップグレードを実行します。

第1章 Supported Platforms

</ div>

Advanced Server がサポートするプラットフォームとバージョンについては、次の EDB ウェブサイトにアクセスしてください。

<https://www.enterprisedb.com/services-support/edb-supported-products-and-platforms#epas>

第2章 Limitations

EDB Postgres Advanced Server には次の制限が適用されます。

- 稼動データベースの data ディレクトリを NFS ファイルシステムに保存しないでください。
- `pg_upgrade` ユーティリティは、外部キーがパーティションテーブルを参照している場合、パーティションテーブルをアップグレードできません。
- バージョン 9.4 サーバーまたは Advanced Server の下位バージョンからアップグレード処理場合、`SUBPARTITION BY` 句を含むテーブルパーティションを使用する場合、`pg_dump` および `pg_restore` を使用して、既存の Advanced Server インストールを Advanced Server の新しいバージョンにアップグレードする必要があり Advanced Server。をアップグレードするには、以下を行う必要があります。
 1. `pg_dump` を使用して、サブパーティションテーブルのコンテンツを保存します。
 2. Advanced Server 9.4 データベースまたは Advanced Server データベースの下位バージョンからテーブルを削除します。
 3. `pg_upgrade` を使用して、Advanced Server データベースの残りをより新しいバージョンにアップグレードします。
 4. `pg_restore` を使用して、サブパーティションテーブルを最新のアップグレードされた Advanced Server データベースに復元します。Advanced Server インストールのアップグレードを実行する場合、アップグレードされたサーバーでハッシュパーティションテーブルを再構築する必要があります。

第3章 Upgrading an Installation With pg_upgrade

</div>

バージョン間のマイナーアップグレードは非常にシンプルで、新しい実行可能ファイルのインストールのみが必要ですが、過去のメジャーバージョンのアップグレードは高価で時間がかかりました。pg_upgrade は、Advanced Server の任意のバージョン（バージョン 9.0 以降）と、同じプラットフォームでサポートされている Advanced Server の以降のリリースとの間の移行を容易にします。

pg_upgrade がない場合、Advanced Server の以前のバージョンから Advanced Server 14 に移行するには、pg_dump を使用してすべてのデータをエクスポートし、新しいリリースをインストールし、initdb を実行して新しいクラスターを作成し、古いデータをインポートする必要があります。

- pg_upgrade は、多くのメジャーバージョンアップに必要な時間とディスクスペースの両方を削減できます。*

pg_upgrade ユーティリティは、Advanced Server と後続のバージョン間で既存のデータのインプレース転送を実行します。

インプレースアップグレードが実用的かどうかは、いくつかの要因によって決まります。

- ユーザー定義テーブルのディスク上の表現形式は、オリジナルバージョンとアップグレードされたバージョンの間で変更してはなりません。
- データ型のディスク上の表現形式は、オリジナルバージョンとアップグレードされたバージョンの間で変更してはなりません。
- pg_upgrade を使用して Advanced Server のメジャーバージョン間でアップグレードするには、両方のバージョンで各データタイプの共通バイナリ表現形式を共有する必要があります。したがって、pg_upgrade を使用して 32 ビットから 64 ビット Linux プラットフォームに移行することはできません。

バージョンアップグレードを実行する前に、pg_upgrade は 2 つのクラスター（古いクラスターと新しいクラスター）に互換性があることを確認します。

アップグレードにデータベースオブジェクトまたはデータのディスク上の表現形式の変更が含まれる場合、またはデータ型のバイナリ表現形式の変更が含まれる場合、pg_upgrade はアップグレードを実行できません。アップグレードするには、古いデータを pg_dump してから、そのデータを新しいクラスターにインポートする必要があります。

pg_upgrade 実行可能ファイルは Advanced Server 14 と共に配布され、Database Server コンポーネントのパートとしてインストールされます。追加のインストールまたは構成手順は必要ありません。

3.1 Performing an Upgrade

Advanced Server の以前のバージョンを現在のバージョンにアップグレードするには、以下を行う必要があります。

Advanced Server の現在のバージョンをインストールします。新しいインストールには、古いインストールと同じサポートサーバーコンポーネントが含まれている必要があります。- ターゲットデータベースを空にするか、`initdb` を使用して新しいターゲットクラスターを作成します。- 両方のデータベースの `pg_hba.conf` ファイルを `trust` 認証モードにします（認証の競合を避けるため）。- 新旧の Advanced Server サービスをシャットダウンします。- `pg_upgrade` ユーティリティを呼び出します。

`pg_upgrade` は、起動時に互換性保証を実行して、必要なすべての実行可能ファイルが存在し、予想されるバージョン番号が含まれていることを確認します。検証プロセスは、古い `$PGDATA` ディレクトリと新しい `$PGDATA` ディレクトリもチェックして、予想されるファイルとサブディレクトリが適切に配置されていることを保証します。検証プロセスが成功すると、`pg_upgrade` は古い `postmaster` を起動し、`pg_dumpall --schema-only` を実行して古いクラスターに含まれるメタデータをキャプチャします。`pg_dumpall` によって生成されたスクリプトは、後の手順で新しいクラスター内のすべてのユーザー定義オブジェクトを再作成するために使用されます。

`pg_dumpall` によって生成されるスクリプトは、ユーザー定義オブジェクトのみを再作成し、システム定義オブジェクトは再作成しないことに注意してください。新しいクラスターには、Advanced Server の最新バージョンによって作成されたシステム定義オブジェクトが**既**に含まれています。

古いクラスターからメタデータを抽出した後、`pg_upgrade` は、新しいクラスターを既存のデータと同期するために必要なブックキーピングタスクを実行します。

`pg_upgrade` は、新しいクラスターに対して `pg_dumpall` スクリプトを実行して、古いクラスターで見つかったものと同じ形状とタイプの（空の）データベースオブジェクトを作成します。次に、`pg_upgrade` は、古いクラスターから新しいクラスターに各テーブルとインデックスをリンクまたはコピーします。

Advanced Server 14 にアップグレード処理し、`edb_dblink_oci` または `edb_dblink_libpq` 拡張機能をインストールした場合、アップグレードを実行する前に拡張機能を削除する必要があります。拡張機能を削除するには、`psql` または PEM クライアントを使用してサーバーに接続し、コマンドを呼び出します。

```
DROP EXTENSION edb_dblink_oci;
DROP EXTENSION edb_dblink_libpq;
```

アップグレード処理が完了したら、`CREATE EXTENSION` コマンドを使用して、現在のバージョンの拡張機能をインストールに追加できます。

3.1.1 Linking versus Copying

</div>

`pg_upgrade` を呼び出すときに、コマンドラインオプションを使用して、`pg_upgrade` が古いクラスターの各テーブルとインデックスを新しいクラスターに**コピー**または**リンク**するかどうかを指定できます。

`pg_upgrade` はクラスター内の各ファイルに対して 2 番目の名前（ハードリンク）を作成するだけなので、リンクははるかに高速です。`pg_upgrade` はオリジナルデータのコピーを作成しない `make`、追加のワーク

スペースも必要ありません。古いクラスターと新しいクラスターをリンクするとき、古いクラスターと新しいクラスターはデータを共有します。新しいクラスターを開始した後、以前のバージョンの Advanced Server でデータを使用できなくなることにノートしてください。

古いクラスターから新しいクラスターにコピーデータことを選択した場合、`pg_upgrade` は、従来の `dump/restore` プロシージャと比較して、アップグレードの実行に必要な時間を短縮します。`pg_upgrade` は、ファイルごとのメカニズムを使用して、古いクラスターから新しいクラスターにコピーデータします (`dump/restore` で使用される行ごとのメカニズムとは異なります)。`pg_upgrade` を使用すると、新しいクラスターでのインデックスの作成を回避できます。各インデックスは、古いクラスターから新しいクラスターに単純にコピーされます。最後に、`dump/restore` プロシージャを使用してアップグレードするには、すべてのデータの中間テキストベースのダンプを保持するために大量のワークスペースが必要ですが、`pg_upgrade` には余分なワークスペースがほとんど必要ありません。

ユーザー定義のテーブルスペースに保存されているデータは、新しいクラスターにコピーされません。ファイルシステム内の同じ場所に残りますが、名前が新しいクラスターのバージョン番号を反映するサブディレクトリにコピーされます。アップグレード処理後にテーブルスペースに保存されているファイルを手動で再配置するには、ファイルを新しい場所に移動し、ファイルを指すようにシンボリックリンク (クラスターの `data` ディレクトリの下に `pg_tblspc` ディレクトリにあります) を更新します。

3.2 Invoking pg_upgrade

`pg_upgrade` を呼び出す場合、新旧のクラスターの PGDATA および実行可能 (`/bin`) ディレクトリーの場所、および Advanced Server スーパーユーザの名前、インストールがリッスンしているポートを指定する必要があります。`pg_upgrade` を呼び出して Advanced Server 13 から Advanced Server 14 に移行する一般的な呼び出しは、次のフォームを取ります。

```
pg_upgrade
--old-datadir <path_to_13_data_directory>
--new-datadir <path_to_14_data_directory>
--user <superuser_name>
--old-bindir <path_to_13_bin_directory>
--new-bindir <path_to_14_bin_directory>
--old-port <13_port> --new-port <14_port>
```

どこで：

```
--old-datadir path_to_13_data_directory
```

`--old-datadir` オプションを使用して、Advanced Server 13 インストール内の `data` ディレクトリへの完全なパスを指定します。

```
--new-datadir path_to_14_data_directory
```

`--new-datadir` オプションを使用して、Advanced Server 14 インストール内の `data` ディレクトリへの完全なパスを指定します。

```
--username superuser_name
```

`--username` オプションを含めて、Advanced Server スーパーユーザの名前を指定します。スーパーユーザ名前は、Advanced Server の両方のバージョンで同じである必要があります。デフォルトでは、Advanced

Server がオラクルモードでインストールされている場合、スーパーユーザの名前付けは **enterprisedb** です。PostgreSQL モードでインストールした場合、スーパーユーザの名前付けは **postgres** です。

Advanced Server のスーパーユーザ名前が両方のクラスターで同じでない場合、クラスターは **pg_upgrade** 整合性チェックに合格しません。

--old-bindir path_to_13_bin_directory

--old-bindir オプションを使用して、Advanced Server 13 インストールの **bin** ディレクトリへの完全なパスを指定します。

--new-bindir path_to_14_bin_directory

--new-bindir オプションを使用して、Advanced Server 14 インストールの **bin** ディレクトリへの完全なパスを指定します。

--old-port 13_port

--old-port オプションを含めて、Advanced Server 13 が接続をリッスンするポートを指定します。

--new-port 14_port

--new-port オプションを含めて、Advanced Server 14 が接続をリッスンするポートを指定します。

3.2.1 Command Line Options - Reference

pg_upgrade は、次のコマンドラインオプションを受け入れます。各オプションは、長いフォームまたは短いフォームで使用できます。

-b path_to_old_bin_directory --old-bindir path_to_old_bin_directory

-b または **--old-bindir** キーワードを使用して、古いクラスターの実行可能ディレクトリの場所を指定します。

-B path_to_new_bin_directory --new-bindir path_to_new_bin_directory

-B または **--new-bindir** キーワードを使用して、新しいクラスターの実行可能ディレクトリの場所を指定します。

-c --check

-c または **--check** キーワードを含めて、バージョンアップグレードを実行せずに **pg_upgrade** が新旧クラスターの整合性チェックを実行するように指定します。

-d path_to_old_data_directory --old-datadir path_to_old_data_directory

-d または **--old-datadir** キーワードを使用して、古いクラスターの **data** ディレクトリの場所を指定します。

-D path_to_new_data_directory --new-datadir path_to_new_data_directory

-D または **--new-datadir** キーワードを使用して、新しいクラスターの **data** ディレクトリの場所を指定します。

ユーザー定義のテーブルスペースに保存されているデータは、新しいクラスターにコピーされません。ファイルシステム内の同じ場所に残りますが、名前が新しいクラスターのバージョン番号を反映するサブディレクトリにコピーされます。アップグレード処理後にテーブルスペースに保存されているファイルを手動で再配置するには、ファイルを新しい場所に移動し、ファイルを指すようにシンボリックリンク（クラスターの `data` ディレクトリの下に `pg_tblspc` ディレクトリにあります）を更新する必要があります。

`-j --jobs`

`-j` または `--jobs` キーワードを含めて、アップグレード中に使用する同時プロセスまたはスレッドの数を指定します。

`-k --link`

`-k` または `--link` キーワードを含めて、新しいクラスターから古いクラスターへのハードリンクを作成します。シンボリックリンクの使用の詳細については、[Linking versus Copying](#) を参照してください。

`-o options --old-options options`

`-o` または `--old-options` キーワードを使用して、古い `postgres` コマンドに渡されるオプションを指定します。オプションが一重引用符またはダブルクォートで囲まれて、グループとして渡されるようにし保証。

`-O options --new-options options`

`-O` または `--new-options` キーワードを使用して、新しい `postgres` コマンドに渡すオプションを指定します。オプションが一重引用符またはダブルクォートで囲まれて、グループとして渡されるようにし保証。

`-p old_port_number --old-port old_port_number`

`-p` または `--old-port` キーワードを含めて、アップグレード処理する Advanced Server インストールのポート番号を指定します。

`-P new_port_number --new-port new_port_number`

`-P` または `--new-port` キーワードを含めて、新しい Advanced Server インストールのポート番号を指定します。

!!! Note * Advanced Server 14 インストーラを呼び出すときにオリジナルの Advanced Server インストールがポート番号 5444 を使用している場合、インストーラは Advanced Server の新規インストールにリスナポート 5445 の使用を推奨します。

`-r --retain`

アップグレードプロセス中に、`pg_upgrade` は 4 つの追加専用ログファイルを作成します。アップグレードが完了すると、`pg_upgrade` はこれらのファイルを削除します。`-r` または `--retain` オプションを含めて、サーバーが `pg_upgrade` ログファイルを保持するように指定します。

`-U user_name --username user_name`

`-U` または `--username` キーワードを含めて、Advanced Server データベーススーパーユーザの名前を指定します。同じスーパーユーザが両方のクラスターに存在する必要があります。

`-v --verbose`

`-v` または `--verbose` キーワードを含めて、アップグレードプロセス中に詳細な出力を有効にします。

`-V --version`

-V または --version キーワードを使用して、pg_upgrade のバージョン情報を表示します。

-? -h --help

-, -h, または --help オプションを使用して、pg_upgrade ヘルプ情報を表示します。

3.3 Upgrading to Advanced Server 14

アップグレードを使用して、Advanced Server の既存のインストールから Advanced Server 14 インストーラによって構築されたクラスター、または `initdb` コマンドを使用して作成された代替クラスターにアップグレードできます。このセクションでは、インストーラによって提供されるクラスターへのアップグレード処理の詳細を提供します。

`initdb` コマンドで作成された空のクラスターにアップグレードする基本的な手順は、Advanced Server 14 インストーラで作成されたクラスターにアップグレードする手順と同じですが、手順 2 (Empty the edb database) を省略して、代替の場所を置き換えることができますアップグレードのターゲットクラスターを指定するときのクラスター。

アップグレードプロセス中に問題が発生した場合は、以前のバージョンに戻すことができます。このプロセスの詳細については、[Reverting to the old cluster](#) セクションを参照してください。

Advanced Server のアップグレードを実行するには、オペレーティングシステムのスーパーユーザまたは Windows 管理者である必要があります。

ステップ 1-新しいサーバーのインストール

Advanced Server 14 をインストールします。前回の Advanced Server のインストール中にインストールされたものと同じ非サーバーコンポーネントを指定します。新しいクラスターと古いクラスターは異なるディレクトリに存在する必要があります。

ステップ 2-ターゲットデータベースを空にします

ターゲットクラスターにデータを含めることはできません。`initdb` コマンドを使用して空のクラスターを作成するか、Advanced Server 14 のインストール中に作成されたデータベースを空にすることができます PostgreSQL モードで Advanced Server をインストールした場合、インストーラは `postgres` 名前付け単一のデータベースを作成します; Advanced Server をオラクルモードでインストールした場合、`postgres` という名前付けのデータベースと `edb` 名前付けのデータベースが作成されます。

ターゲットデータベースを空にする最も簡単な方法は、データベースを削除してから新しいデータベースを作成することです。`DROP DATABASE` コマンドを呼び出す前に、ユーザーを切断し、現在データベースを使用しているサービスを停止する必要があります。

Windows、Control Panel から Services マネージャに移動します。Services リストで各サービスを強調表示し、Stop を選択します。

Linux では、ターミナルウィンドウを開き、スーパーユーザ権限を引き受け、各サービスを手動で停止します。例、次のコマンドを呼び出して pgAgent サービスを停止します。

```
service edb-pgagent-14 stop
```

現在 Advanced Server に接続されているサービスを停止した後、EDB-PSQL コマンドラインクライアントを使用して、データベースを削除および作成できます。クライアントが開いたら、データベーススーパーユーザとして `template1` データベースに接続します。プロンプトが表示されたら、認証情報を提供します。次に、次のコマンドを使用してデータベースを削除します。

```
DROP DATABASE <database_name>;
```

ここで、`database_name` はデータベースの名前です。

次に、`template1` データベースの内容に基づいて空のデータベースを作成します。

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

ステップ 3-両方のサーバーを信頼モードに設定します

アップグレードプロセス中に、`pg_upgrade` は古いサーバーと新しいサーバーに数回接続します。接続プロセスを `make` にするために、`pg_hba.conf` ファイルを編集して、認証モードを `trust` に設定できます。`pg_hba.conf` ファイルを変更するには、Start メニューから EDB Postgres メニューに移動します。Advanced Server メニューに移動し、Expert Configuration メニューを開きます。Edit `pg_hba.conf` メニューオプションを選択して、`pg_hba.conf` ファイルを開きます。

以前の Advanced Server インストールおよび Advanced Server 14 サーバーの信頼認証を許可する必要があります。次の図に示すように、Advanced Server の両方のインストールの `pg_hba.conf` ファイルを編集します。

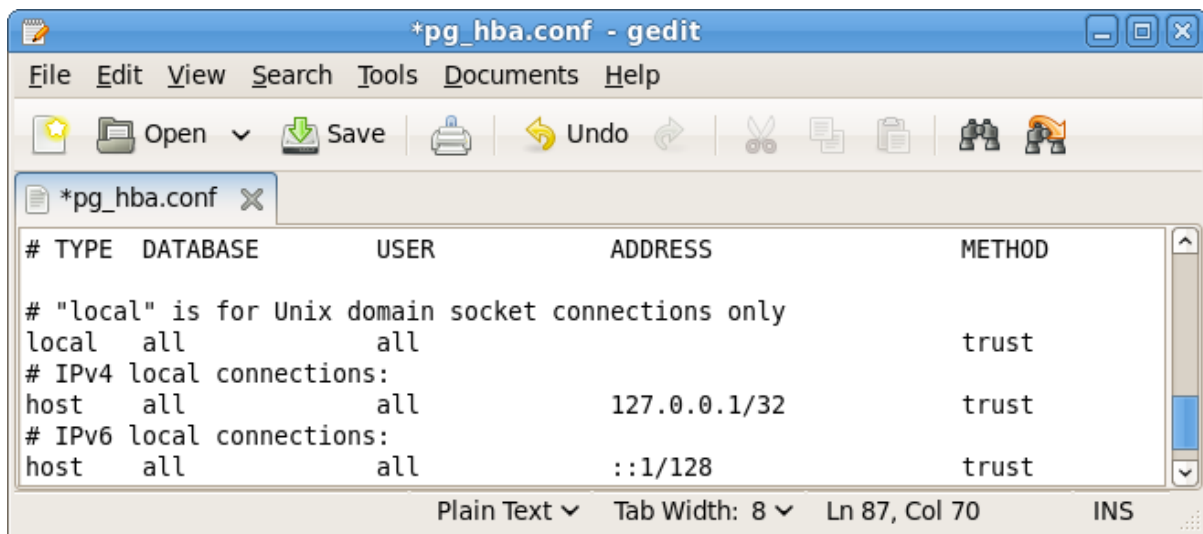


図 1: Configuring Advanced Server to use trust authentication.

図 1：信頼認証を使用するための Advanced Server の構成

各ファイルを編集した後、ファイルを保存してエディタを終了します。

アップグレードプロセス中にシステムが `md5` 認証モードを維持する必要がある場合、パスワードファイル (Windows では `pgpass.conf`、Linux では `pgpass`) でデータベーススーパーユーザのユーザパスワードを指定できます。パスワードファイルの設定の詳細については、次の URL で入手可能な PostgreSQL Core Documentation を参照してください。

< [https://www.postgresql.org/docs/current/ 静的/libpq-pgpass.html](https://www.postgresql.org/docs/current/静的/libpq-pgpass.html) >

ステップ 4-すべてのコンポーネントサービスとサーバーを停止

pg_upgrade を呼び出す前に、オリジナルの Advanced Server インストール、Advanced Server 14、またはサポートコンポーネントに属するサービスを停止する必要があります。これにより、アップグレードプロセス中にサービスがいずれのクラスターにもアクセスしないようにします。

インストールで実行される可能性が最も高いサービスは次のとおりです。

Service:	On Linux:	On Windows:
EnterpriseDB Postgres Advanced Server 9.6	edb-as-9.6	edb-as-9.6
EnterpriseDB Postgres Advanced Server 10	edb-as-10	edb-as-10
EnterpriseDB Postgres Advanced Server 11	edb-as-11	edb-as-11
EnterpriseDB Postgres Advanced Server 12	edb-as-12	edb-as-12
EnterpriseDB Postgres Advanced Server 13	edb-as-13	edb-as-13
EnterpriseDB Postgres Advanced Server 14	edb-as-14	edb-as-14
Advanced Server 9.6 Scheduling Agent (pgAgent)	edb-pgagent-9.6	EnterpriseDB Postgres Advanced Server 9.6 Scheduling Agent
Infinite Cache 9.6	edb-icache	N/A
Infinite Cache 10	edb-icache	N/A
PgBouncer	Pgbouncer	Pgbouncer
PgBouncer 1.6	ppas-pgbouncer-1.6 or ppas-pgbouncer16	ppas-pgbouncer-1.6
PgBouncer 1.7	edb-pgbouncer-1.7	edb-pgbouncer-1.7
PgPool	ppas-pgpool	N/A
PgPool 3.4	ppas-pgpool-3.4 or ppas-pgpool34	N/A
pgPool-II	edb-pgpool-3.5	N/A
Slony 9.6	edb-slony-replication-9.6	edb-slony-replication-9.6
xDB Publication Server 9.0	edb-xdbpubserver-90	Publication Service 90
xDB Publication Server 9.1	edb-xdbpubserver-91	Publication Service 91
xDB Subscription Server	edb-xdbsubserver-90	Subscription Service 90
xDB Subscription Server	edb-xdbsubserver-91	Subscription Service 91
EDB Replication Server v6.x	edb-xdbpubserver	Publication Service for xDB Replication Server
EDB Subscription Server v6.x	edb-xdbsubserver	Subscription Service for xDB Replication Server

次のページに続く

表 1 – 前のページからの続き

Service:	On Linux:	On Windows:

**** Windows でサービスを停止するには：****

Services アプレットを開きます。リストに表示されている各 Advanced Server またはサポートコンポーネントサービスを強調表示し、Stop を選択します。

**** Linux でサービスを停止するには：****

ターミナルウィンドウを開き、コマンドラインで各サービスを手動で停止します。

ステップ 5 Linux のみ-クラスター所有者の ID を想定

Linux を使用している場合は、Advanced Server クラスターの所有者の ID を想定します。（次の例では、Advanced Server がオラクルデータベースモードとのデフォルトの互換性でインストールされ、**enterprisedb** をクラスター所有者として割り当てることを想定しています PostgreSQL データベースモードとの互換性でインストールされた場合、**postgres** がクラスター所有者です。）

```
su - enterprisedb
```

プロンプトが表示されたら、Advanced Server クラスターの所有者パスワードを入力します。次に、実行可能ファイル **pg_upgrade** の場所を含むようにパスを設定します。

```
export PATH=$PATH:/usr/edb/as14/bin
```

アップグレードプロセス中に、**pg_upgrade** は **enterprisedb** ユーザのカレントワーキングディレクトリにファイルを書き込みます。**enterprisedb** ユーザが **write** 特権を持っているディレクトリから **pg_upgrade** を呼び出す必要があります。上記のコマンドを実行した後、**enterprisedb** ユーザがファイルを書き込むための十分な権限を持つディレクトリに移動します。

```
cd /tmp
```

ステップ 6 に進みます。

ステップ 5 Windows のみ-クラスター所有者の ID を想定

Windows を使用している場合は、ターミナルウィンドウを開き、Advanced Server クラスターの所有者の ID を想定して、**pg_upgrade** 実行可能ファイルへのパスを設定します。

Advanced Server の初期インストール中に **--serviceaccount service_account_user** パラメータが指定された場合、**service_account_user** は Advanced Server クラスターの所有者であり、**RUNAS** コマンドで指定されるユーザです。

```
RUNAS /USER:service_account_user "CMD.EXE"
SET PATH=%PATH%;C:\Program Files\edb\as14\bin
```

アップグレードプロセス中に、**pg_upgrade** はファイルをサービスアカウントユーザのカレントワーキングディレクトリに書き込みます。サービスアカウントユーザが **write** 特権を持っているディレクトリから **pg_upgrade** を呼び出す必要があります。上記のコマンドを実行した後、サービスアカウントユーザがファイルを書き込むための十分な権限を持つディレクトリに移動します。

```
cd %TEMP%
```

ステップ 6 に進みます。

Advanced Server の初期インストール中に `--serviceaccount` パラメータが省略された場合、Advanced Server サービスとデータベースクラスタのデフォルトの所有者は `NT AUTHORITY\NetworkService` です。

`NT AUTHORITY\NetworkService` がサービスアカウントユーザの場合、`RUNAS` コマンドはパスワードの入力を求められ、`NT AUTHORITY\NetworkService` アカウントにはパスワードが割り当てられないため、使用できない場合があります。そのため、通常、「ユーザパスワードを取得できません」などのエラーメッセージでエラーが発生します。

この状況では、`PsExec` 名前付けの Windows ユーティリティプログラムを使用して、`CMD.EXE` をサービスアカウント `NT AUTHORITY\NetworkService` として実行する必要があります。

`PsExec` プログラムは、次のサイトで入手可能な `Pstools` をダウンロードして取得する必要があります。

<https://technet.microsoft.com/en-us/sysinternals/bb897553.aspx>。

その後、次のコマンドを使用して `CMD.EXE` を `NT AUTHORITY\NetworkService` として実行し、`pg_upgrade` 実行可能ファイルへのパスを設定できます。

```
psexec.exe -u "NT AUTHORITY\NetworkService" CMD.EXE
SET PATH=%PATH%;C:\Program Files\edb\as14\bin
```

アップグレードプロセス中に、`pg_upgrade` はファイルをサービスアカウントユーザのカレントワーキングディレクトリに書き込みます。サービスアカウントユーザが `write` 特権を持っているディレクトリから `pg_upgrade` を呼び出す必要があります。上記のコマンドを実行した後、サービスアカウントユーザがファイルを書き込むための十分な権限を持つディレクトリに移動します。

```
cd %TEMP%
```

ステップ 6 に進みます。

ステップ 6-整合性チェックを実行します

アップグレードを試みる前に、一貫性チェックを実行して、古いクラスターと新しいクラスターが互換性があり、適切に構成されていることを確認します。`--check` オプションを含めて、`pg_upgrade` に整合性チェックを実行するよう指示します。

次の例は、`pg_upgrade` を呼び出して Linux で整合性チェックを実行する方法を示しています。

```
pg_upgrade -d /var/lib/edb/as13/data
- D /var/lib/edb/as14/data -U enterprisedb
- b /usr/edb/as13/bin -B /usr/edb/as14/bin -p 5444 -P 5445 --check
```

コマンドが成功すると、`*Clusters are compatible*` が結果れます。

Windows を使用している場合は、スペースを含むディレクトリ名をクォートで囲む必要があります。

```
pg_upgrade.exe
- d "C:\Program Files\ PostgresPlus\13AS\data"
- D "C:\Program Files\edb\as14\data" -U enterprisedb
- b "C:\Program Files\PostgresPlus\13AS\bin"
- B "C:\Program Files\edb\as14\bin" -p 5444 -P 5445 --check
```

一貫性チェックプロセス中、**pg_upgrade** は、見つかった不一致をログが呼び出されたディレクトリにあるファイルに記録します。整合性チェックが完了したら、ファイルを確認して、欠落しているコンポーネントまたはアップグレードの競合を特定します。**pg_upgrade** を呼び出してバージョンアップグレードを実行する前に、競合を解決する必要があります。

pg_upgrade がコンポーネントの欠落について警告する場合、**StackBuilder Plus** を使用して、コンポーネントを含むコンポーネントを追加できます。**StackBuilder Plus** を使用する前に、**Advanced Server 14** サービスをリスタートする必要があります。サービスを再起動した後、**Start** メニューから **Advanced Server 14** メニューに移動し、**StackBuilder Plus** を選択して **StackBuilder Plus** を開きます。**StackBuilder Plus** ウィザードの画面上のアドバイスに従って、不足しているコンポーネントをダウンロードしてインストールします。

pg_upgrade がクラスターに互換性があることを確認したら、バージョンアップグレードを実行できます。

ステップ 7- **pg_upgrade** を実行します

クラスターに互換性があることを確認したら、**pg_upgrade** を呼び出して、古いクラスターを **Advanced Server** の新しいバージョンにアップグレードできます。

Linux の場合：

```
pg_upgrade -d /var/lib/edb/as13/data
- D /var/lib/edb/as14/data -U enterprisedb
- b /usr/edb/as13/bin -B /usr/edb/as14/bin -p 5444 -P 5445
```

Windows：

```
pg_upgrade.exe -d "C:\Program Files\PostgresPlus\13AS\data"
- D "C:\Program Files\edb\as14\data" -U enterprisedb
- b "C:\Program Files\PostgresPlus\13AS\bin"
- B "C:\Program Files\edb\as14\bin" -p 5444 -P 5445
```

pg_upgrade は、アップグレードの進行状況を画面に表示します。

```
$ pg_upgrade -d /var/lib/edb/as13/data -D /var/lib/edb/as14/data -U
enterprisedb -b /usr/edb/as13/bin -B /usr/edb/as14/bin -p 5444 -P 5445
Performing Consistency Checks
-----
Checking current, bin, and data directories          ok
Checking cluster versions                            ok
Checking database user is a superuser                ok
Checking for prepared transactions                   ok
Checking for reg* system OID user data types         ok
```

(次のページに続く)

(前のページからの続き)

```

Checking for contrib/isn with bigint-passing mismatch  ok
Creating catalog dump                                ok
Checking for presence of required libraries           ok
Checking database user is a superuser                 ok
Checking for prepared transactions                    ok

```

If pg_upgrade fails after this point, you must re-initdb the new cluster before continuing.

Performing Upgrade

```

-----
Analyzing all rows in the new cluster                  ok
Freezing all rows on the new cluster                  ok
Deleting files from new pg_clog                       ok
Copying old pg_clog to new server                     ok
Setting next transaction ID for new cluster           ok
Resetting WAL archives                               ok
Setting frozenxid counters in new cluster             ok
Creating databases in the new cluster                 ok
Adding support functions to new cluster               ok
Restoring database schema to new cluster              ok
Removing support functions from new cluster            ok
Copying user relation files                           ok

Setting next OID for new cluster                      ok
Creating script to analyze new cluster                ok
Creating script to delete old cluster                 ok

```

Upgrade Complete

Optimizer statistics are not transferred by pg_upgrade so, once you start the new server, consider running:

```
* analyze_new_cluster.sh
```

Running this script will delete the old cluster's data files:

```
* delete_old_cluster.sh
```

pg_upgrade の実行中に、アップグレード中に発生した特別な状況をハンドルする SQL スクリプトが生成される場合があります。例、古いクラスターにラージオブジェクトが含まれている場合、新しいクラスター内のオブジェクトのデフォルトのアクセス許可を定義するスクリプトを呼び出す必要があります。アップグレード前の整合性チェックを実行すると、pg_upgrade は、手動で実行する必要がある可能性のあるスクリプトを警告します。

pg_upgrade の完了後にスクリプトを呼び出す必要があります。スクリプトを呼び出すには、EDB-PSQL コマンドラインクライアントでデータベーススーパーユーザとして新しいクラスターに接続し、\i オプションを使用して各スクリプトを呼び出します。

```
\i complete_path_to_script/script.sql
```

通常、再構築スクリプトが完了するまで、再構築スクリプトで被参照されるテーブルにアクセスすることは安全ではありません。テーブルにアクセスと、誤った結果やパフォーマンスの低下が生じる可能性があります。再構築スクリプトで被参照されていないテーブルには、すぐにアクセスできます。

pg_upgrade がアップグレードプロセスの完了に失敗した場合、\$PGDATA/global/pg_control の名前が pg_control.old に変更され、各テーブルスペースの名前が tablespace.old に変更されることを除いて、古いクラスターは変更されません。呼び出し前の状態に戻すには：

1. 新しいクラスターによって作成されたテーブルスペースディレクトリを削除します。.old サフィックス.3 を削除して、\$PGDATA/global/pg_control の名前を変更します。.old サフィックス.4. を削除して、古いクラスターテーブルスペースディレクトリ名を変更します。アップグレードが失敗する前に移動された可能性のあるデータベースオブジェクトを（新しいクラスターから）削除します。

これらの手順を実行した後、アップグレードを再試行する前に、発生したアップグレードの競合を解決してください。

アップグレードが完了すると、バキューム処理は新しいクラスターのバキュームを推奨し、古いクラスターを削除できるスクリプトを提供します。

!!! Note * 古いクラスターを削除する前に、クラスターが期待どおりにアップグレードされていること、および以前のバージョンに戻す必要がある場合に備えてクラスターのバックアップを保持していることを保証してください。

ステップ 8-pg_hba.conf ファイルの認証設定を復元します

trust 認証を許可するように pg_hba.conf ファイルを変更した場合は、pg_hba.conf ファイルの内容を更新して、好みの認証設定を反映させます。

ステップ 9-ユーザー定義のテーブルスペースを移動して識別する（オプション）

ユーザー定義のテーブルスペースにデータが保存されている場合、アップグレード処理後にテーブルスペースファイルを手動で再配置する必要があります。ファイルを新しい場所に移動し、シンボリックリンク（クラスターの data ディレクトリの下に pg_tblspc ディレクトリにある）を更新して、ファイルを指すようにします。

3.4 Upgrading a pgAgent Installation

既存の Advanced Server インストールで pgAgent を使用する場合、Advanced Server 14 インストーラで提供されるスクリプトを使用して pgAgent を更新できます。スクリプトの名前付けは dbms_job.upgrade.script.sql であり、Advanced Server インストールの /share/contrib/ ディレクトリにあります。

pg_upgrade を使用してインストールをアップグレードする場合、次のことを行う必要があります。

1. アップグレードを実行します。dbms_job.upgrade.script.sql スクリプトを呼び出して、カタログファイルを更新します。既存の pgAgent インストールがスクリプトを使用して実行された場合、更新によりインストールが拡張機能に変換されます。

3.5 pg_upgrade Troubleshooting

このセクションのトラブルシューティングのヒントは、`pg_upgrade` の使用時に発生する可能性のある問題に対処します。

3.5.1 アップグレードエラー-クラスターにサービスを提供しているポストマスターがあるようです

`pg_upgrade` からポストマスターがクラスターにサービスを提供していることが報告された場合は、すべての Advanced Server サービスを停止し、アップグレードを再試行してください。

3.5.2 アップグレードエラー-`fe_sendauth` : パスワードが指定されていません

`pg_upgrade` が不足しているパスワードを参照する認証エラーを報告する場合、新旧クラスターの `pg_hba.conf` ファイルを変更して `trust` 認証を有効にするか、`pgpass.conf` ファイルを使用するようにシステムを構成してください。

3.5.3 アップグレードエラー-新しいクラスターは空ではありません。出る

`pg_upgrade` から、新しいクラスターが空でないことが報告された場合は、新しいクラスターを空にしてください。ターゲットクラスタには、ユーザー定義のデータベースを含めることはできません。

3.5.4 アップグレードエラー-ライブラリのロードに失敗しました

オリジナルの Advanced Server クラスターに Advanced Server 14 クラスターに含まれていないライブラリが含まれていた場合、`pg_upgrade` は、`pg_upgrade` を呼び出したディレクトリの `loadable_libraries.txt` ファイルにエントリーを書き込むことにより、整合性チェック中に不足しているコンポーネントを警告します。通常、メジャーコンポーネントアップグレードのパートではないライブラリが見つからない場合は、次の手順を実行します。

1. Advanced Server サービスを再起動します。
 - StackBuilder Plus を使用して、不足しているモジュールをダウンロードしてインストールします。次に：
2. Advanced Server サービスを停止します。
3. アップグレードプロセスを再開します。`pg_upgrade` を呼び出して、整合性チェックを実行します。
4. 整合性チェックで指摘された残りの問題を解決したら、`pg_upgrade` を呼び出して、古いクラスターから新しいクラスターへのデータ移行を実行します。

3.6 Reverting to the Old Cluster

</div>

前のクラスターに戻すために使用されるメソッドは、`pg_upgrade` を呼び出すときに指定されたオプションによって異なります。

- `pg_upgrade` の呼び出し時に `--check` オプションを指定した場合、アップグレードは実行されておらず、古いクラスターは変更されていません。古いクラスターはいつでも再利用できます。
- `pg_upgrade` を呼び出すときに `--link` オプションを含めた場合、アップグレードが完了した後、古いクラスターと新しいクラスターの間でデータファイルが共有されます。新しいクラスターにサービスを提供しているサーバーを起動した場合、新しいサーバーはそれらの共有ファイルに書き込みを行っているため、古いクラスターを使用するのは安全ではありません。
- `--link` を指定せずに `pg_upgrade` を実行した場合、または新しいサーバーを起動していない場合、`.old` サフィックスが `$PGDATA/global/pg_control` およびテーブルスペースディレクトリに追加されていることを除き、古いクラスターは変更されません。
- 古いクラスターを再利用するには、新しいクラスターによって作成されたテーブルスペースディレクトリを削除し、`$PGDATA/global/pg_control` サフィックスを `$PGDATA/global/pg_control` から削除し、古いクラスターテーブルスペースディレクトリ名を削除し、古いクラスターを処理するサーバーをリスタートします。

第4章 Performing a Minor Version Update of an RPM Installation

RPM パッケージを使用して Advanced Server またはそのサポートコンポーネントをインストールした場合、yum を使用して、マイナーバージョンをより新しいバージョンにアップグレード。システムで利用可能なパッケージ更新のリストを確認するには、コマンドラインを開き、ルート権限を引き受けて、コマンドを入力します。

```
yum check-update <package_name>
```

`package_name` は、更新を検索する検索語です。検索語にワイルドカード値を含めることができることにノートしてください。yum update を使用して更新されたパッケージをインストールするには、次のコマンドを使用します。

```
yum update <package_name>
```

`package_name` は、更新するパッケージの名前です。更新コマンドにワイルドカード値を含めて、単一のコマンドでマルチプルの関連パッケージソフトを更新します。例、次のコマンドを使用して、名前に式 `edb` が含まれるすべてのパッケージソフトを更新します。

```
yum update edb*
```

!!! Note * yum update コマンドは、マイナーリリース間の更新のみを実行します。メジャーリリース間で更新するには、`pg_upgrade` を使用する必要があります。

yum コマンドとオプションの使用の詳細については、コマンドラインに `yum --help` と入力してください。

第5章 Using StackBuilder Plus to Perform a Minor Version Update

StackBuilder Plus は、Windows システムでのみサポートされています。

StackBuilder Plus ユーティリティは、Advanced Server のインストールを補完するモジュールの更新、ダウンロード、インストールのプロセスを簡素化するグラフィカルインタフェースを提供します。StackBuilder Plus でモジュールをインストールすると、StackBuilder Plus はソフトウェアの依存関係を自動的に解決します。

インストールが完了したら、Apps メニューから StackBuilder Plus メニューオプションを選択して、StackBuilder Plus をいつでも呼び出すことができます。システムパスワード（プロンプトが表示されたら）、StackBuilder Plus のようこそウィンドウが開きます。



図 1: The StackBuilder Plus welcome window

図 1：StackBuilder Plus のウェルカムウィンドウ

ようこそウィンドウのドロップダウンリストボックスを使用して、Advanced Server インストールを選択し

ます。

StackBuilder Plus にはインターネットアクセスが必要です。Advanced Server のインストールがファイアウォールの背後にある場合（インターネットアクセスが制限されている場合）、StackBuilder Plus はプロキシサーバー経由でプログラムインストーラーをダウンロードできます。モジュールプロバイダーは、HTTP プロキシまたは FTP プロキシを介してモジュールにアクセスできるかどうかを判断します。現在、すべての更新は HTTP プロキシ経由で転送され、FTP プロキシ情報は使用されません。

選択した Advanced Server インストールでインターネットアクセスが制限されている場合、Welcome ウィンドウの **Proxy Servers** を使用して **Proxy servers** ダイアログを開きます（次の図を参照）。

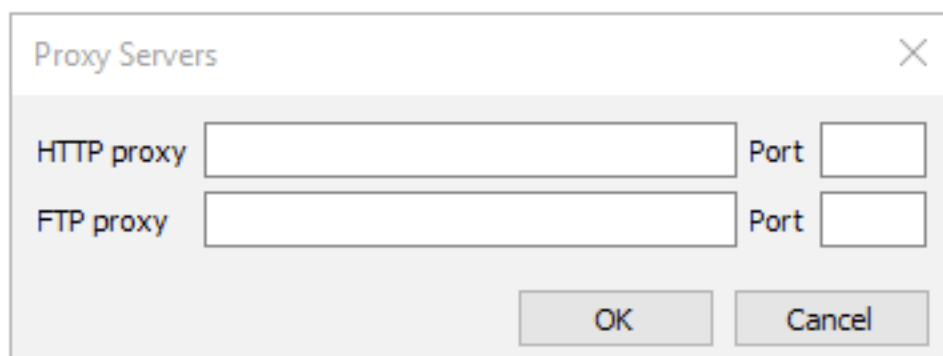


図 2: The Proxy servers dialog

図 2：プロキシサーバーダイアログ

Proxy servers ダイアログの **HTTP proxy** にプロキシサーバーの IP アドレスとポート番号を入力します。現在、すべての StackBuilder Plus モジュールは HTTP プロキシ経由で配布されています（FTP プロキシ情報は無視されます）。OK をクリックして続行します。

図 3：StackBuilder Plus モジュール選択ウィンドウ

StackBuilder Plus モジュール選択ウィンドウ（図を参照）のツリーコントロールには、各モジュールカテゴリのノードが表示されます。

選択した Advanced Server インストールに新しいコンポーネントを追加するか、コンポーネントをアップグレードするには、モジュール名前の左側にあるチェックボックスをオンにして、**Next** をクリックします。プロンプトが表示されたら、StackBuilder Plus 登録ウィンドウでメールアドレスとパスワードを入力します。

図 4：選択したパッケージソフトのリストが表示されたサマリーウィンドウ

StackBuilder Plus は選択されたパッケージソフトを確認します。**Selected packages** ダイアログには、インストーラーの名前とバージョンが表示されます。**Next** をクリックして続行します。

ダウンロードが完了すると、インストールファイルがダウンロードされ、インストールの準備ができていることを確認するウィンドウが開きます。

図 5：ダウンロードプロセスが完了したことの確認

Skip Installation の横にあるボックスをオンにし、**Next** を選択してダウンロードしたファイルをインストールせずに StackBuilder Plus を終了するか、チェックを外して **Next** をクリックしてインストールプロセスをスタートします。

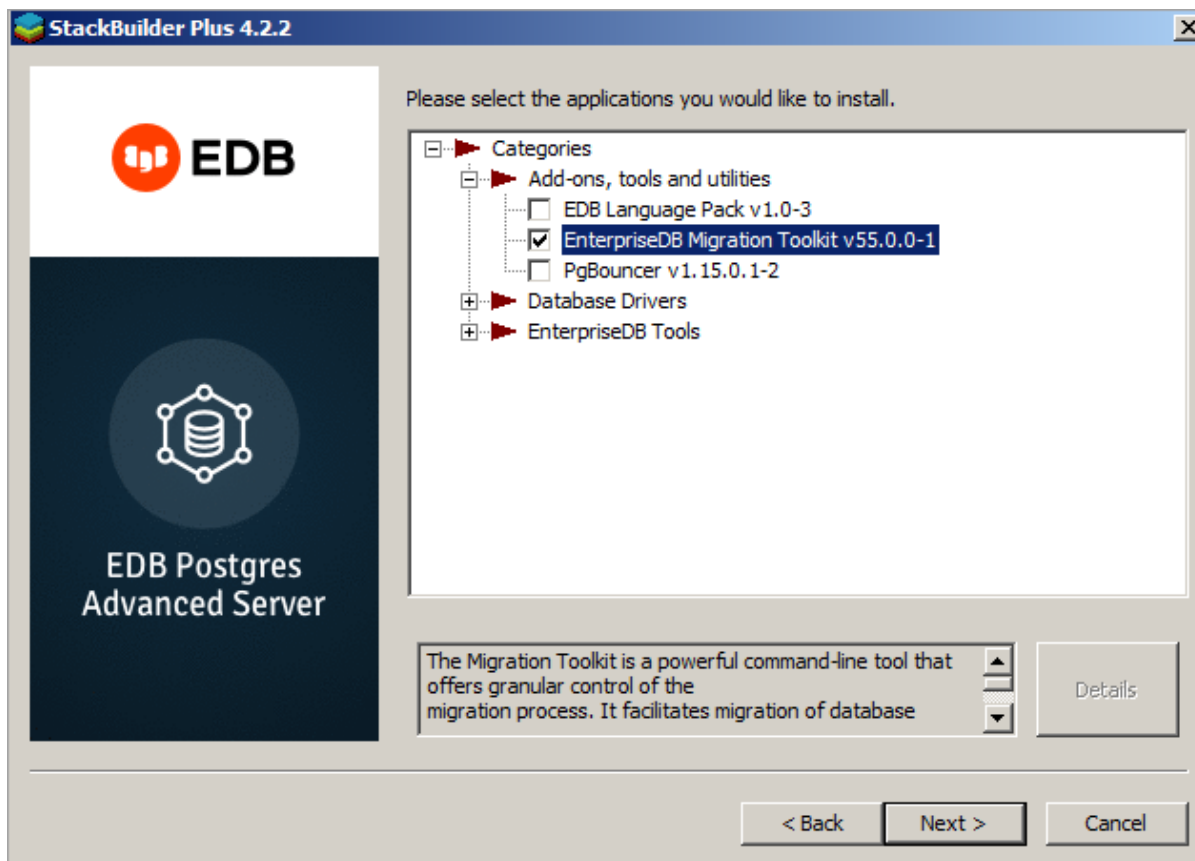


図 3: The StackBuilder Plus module selection window

図 6：StackBuilder Plus は完了したインストールを確認します</ div>

アップグレードが完了すると、StackBuilder Plus は要求されたパッケージのインストールの成功または失敗を警告します。インストーラからコンピュータの再起動を求められた場合は、ここで再起動します。

!!! Note * アップデートのインストールに失敗した場合、StackBuilder Plus はポップアップダイアログでインストールエラーを警告し、%TEMP%のログファイルにメッセージを書き込みます。

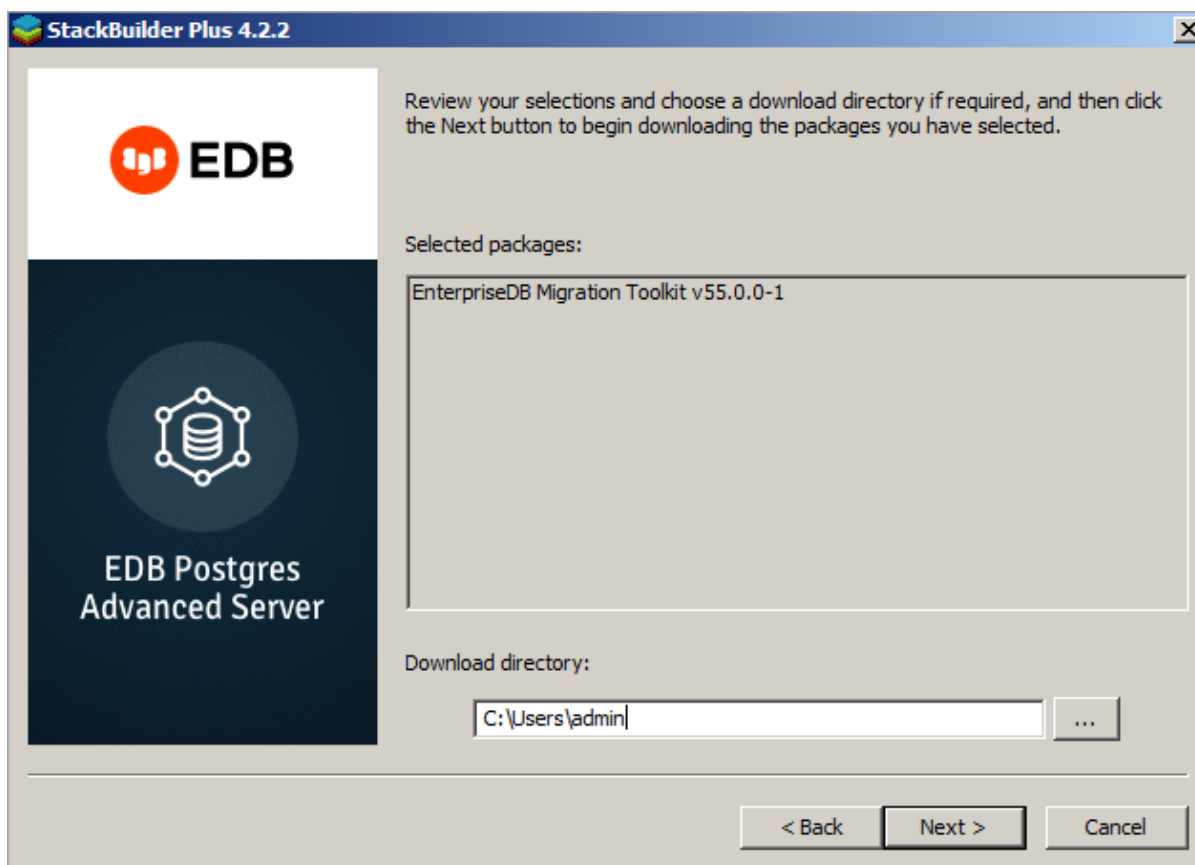


図 4: A summary window displays a list of selected packages

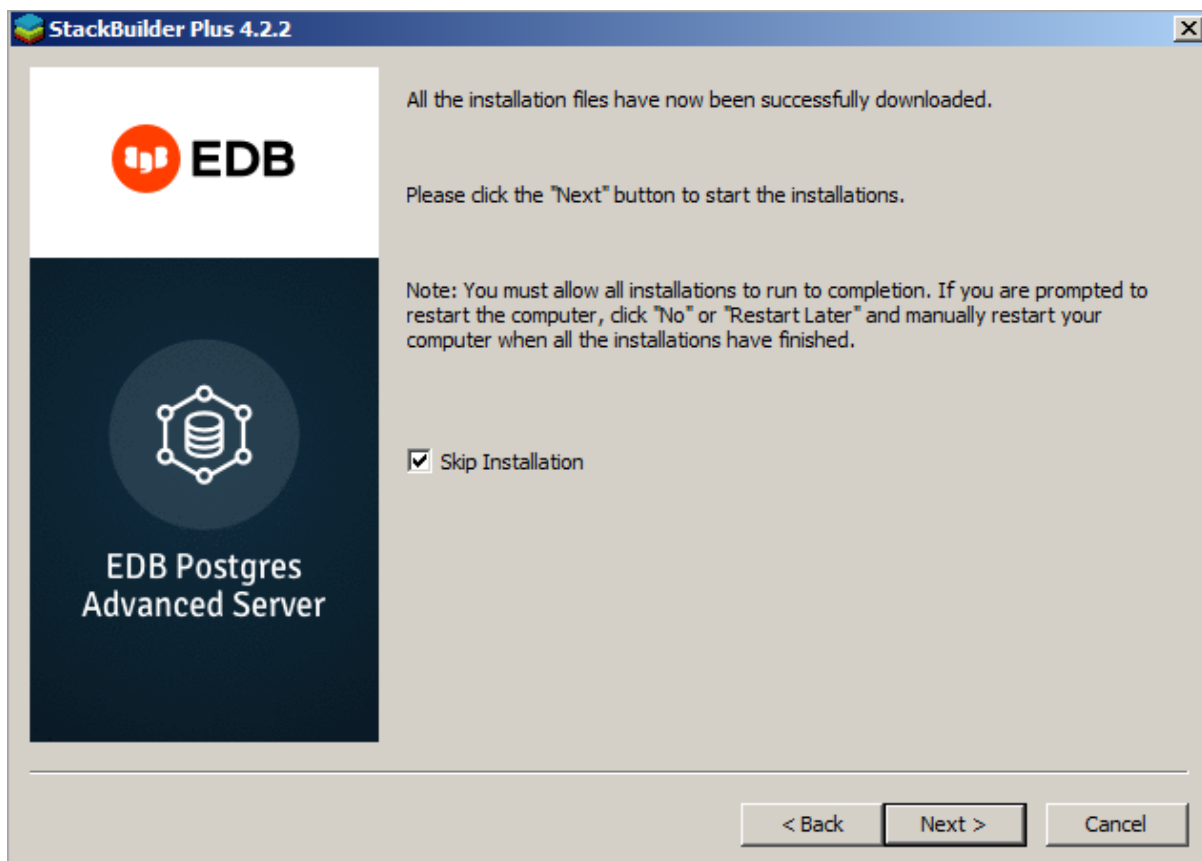


図 5: Confirmation that the download process is complete

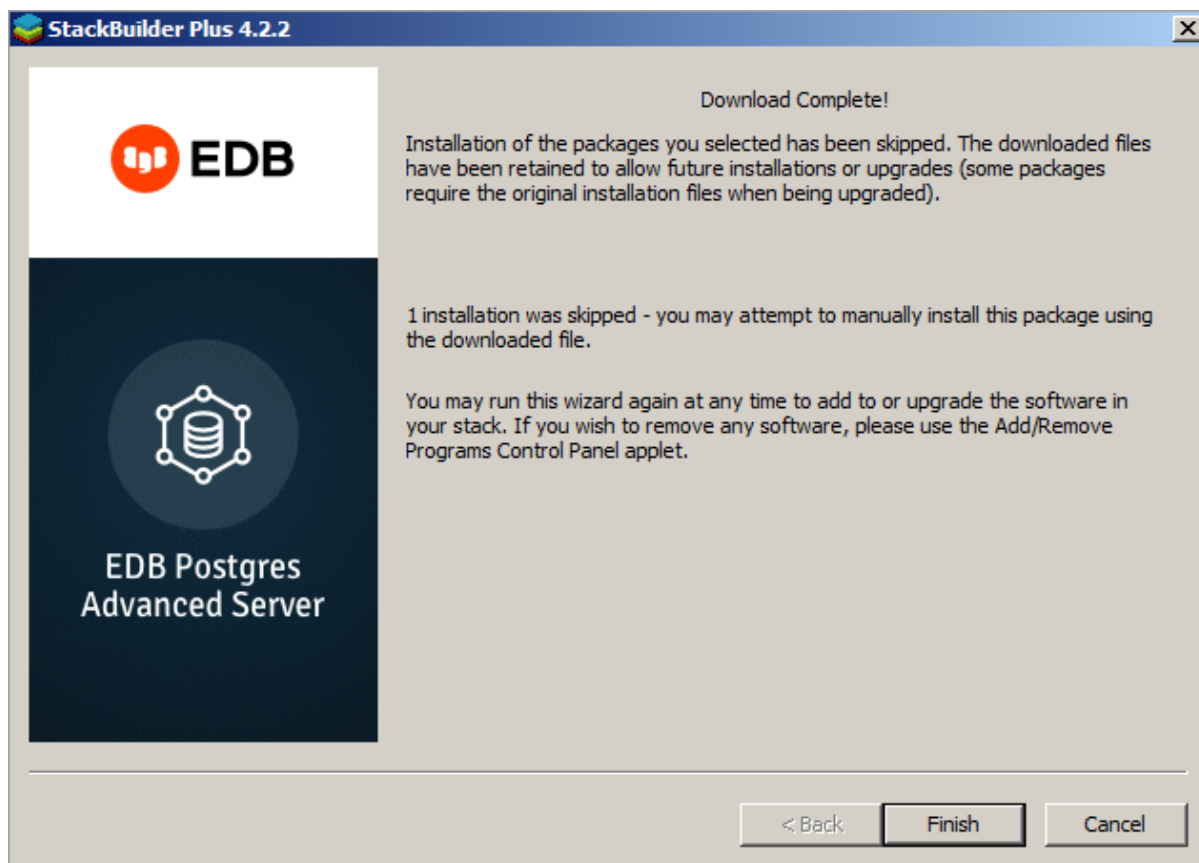


図 6: StackBuilder Plus confirms the completed installation