



EDB Postgres Language Pack

リリース *15.2.0*

EnterpriseDB Corporation

2023 年 03 月 03 日

目次

第 1 章	使用される規則	3
1.1	Supported database server versions	3
1.2	Installing and configuring Language Pack	3
1.2.1	言語パックのインストール	3
1.2.2	言語パックの構成	7
1.3	Using the procedural languages	11
1.3.1	PL / Perl	12
1.3.2	PL / Python	12
1.3.3	PL / Tcl	13
1.4	Uninstalling Language Pack	14

このガイドでは、言語パックをインストールして構成する方法と、手続き型言語（PL / Perl、PL / Python、および PL / Tcl）の使用方法に関する情報を提供します。

言語パックインストーラーには、EDB Postgres Advanced Server および EnterpriseDB PostgreSQL データベースインストーラーで利用できるサポートされている言語が含まれています。言語パックインストーラーを使用すると、サードパーティベンダーのサポートソフトウェアをインストールせずに、Perl、Tcl/Tk、および Python をインストールできます。

Language Pack 2.0 インストーラーには以下が含まれます。

- TK バージョン 8.6 の Tcl
- Perl バージョン 5.26
- Python バージョン 3.7

Perl パッケージには `cpan` パッケージマネージャーが含まれており、Python には `pip` および `easy_install` パッケージマネージャーが含まれています。Tcl/Tk のパッケージマネージャーはありません。

以前の Postgres リリースでは、`plpython` は ActiveState の `python` ライブラリに静的にリンクされていました。Language Pack インストーラーは、Python 用の共有オブジェクトと動的にリンクします。Python 用の ActiveState Linux インストーラーには、ダイナミックライブラリはありません。これらの変更の結果、`plpython` は ActiveState インストーラーで動作しなくなりました。

第1章 使用される規則

Postgres という用語は、PostgreSQL または EDB Postgres Advanced Server を指します。

1.1 Supported database server versions

言語パックのインストーラーはバージョン固有です。EDB Postgres Advanced Server 14 の場合、言語パックバージョン 2 を使用します。

1.2 Installing and configuring Language Pack

このセクションでは、言語パックのインストールと構成について説明します。

1.2.1 言語パックのインストール

グラフィカルインストーラーは、[EDB website](#)、および Stack Builder および StackBuilder Plus から入手できます。StackBuilder Plus は EDB Postgres Advanced Server とともに配布され、Stack Builder は PostgreSQL とともに配布されます。

グラフィカルインストーラーの呼び出し

管理者権限を想定し、インストーラーアイコンをダブルクリックします。プロンプトが表示されたら、Administrator アカウントに関連付けられたパスワードを入力します。プロンプトが表示されたら、インストール言語を選択し、[OK] をクリックします。

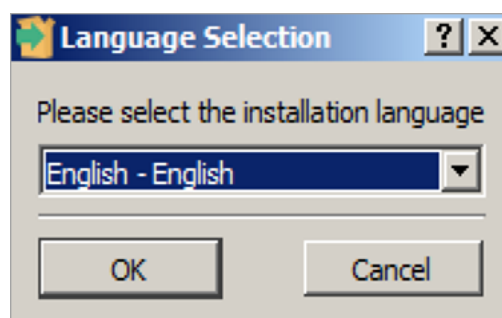


図 1: The Language Selection Window

図 1：言語選択ウィンドウ

言語パックセットアップウィザードのウェルカムウィンドウが開きます。

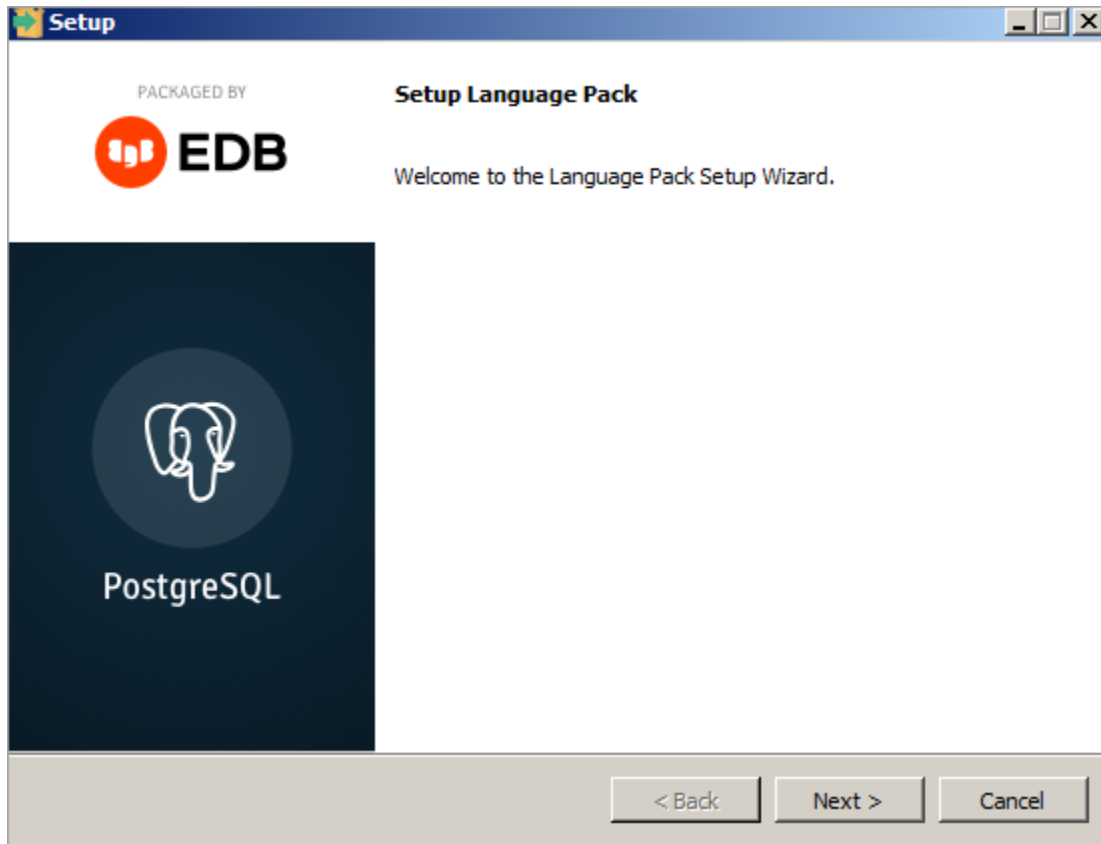


図 2: The Language Pack Welcome Window

図 2：言語パックのウェルカムウィンドウ

Next をクリックして続行します。

[インストールの準備ができました] ウィンドウに、言語パックのインストールディレクトリが表示されます。

Windows 64 の場合： C:/edb/languagepack/v2

OSX の場合： /Library/edb/languagepack/v2

図 3：インストールの準備完了ダイアログ

インストールディレクトリは変更できません。 **Next** をクリックして続行します。

進行状況バーは、インストールの進行状況を示します。

図 4：インストールダイアログ

Next をクリックして続行します。

インストーラーは、言語パックのインストールが完了したことを通知します。 **Finish** をクリックしてインストーラーを終了します。

図 5：言語パックのセットアップの完了ダイアログ

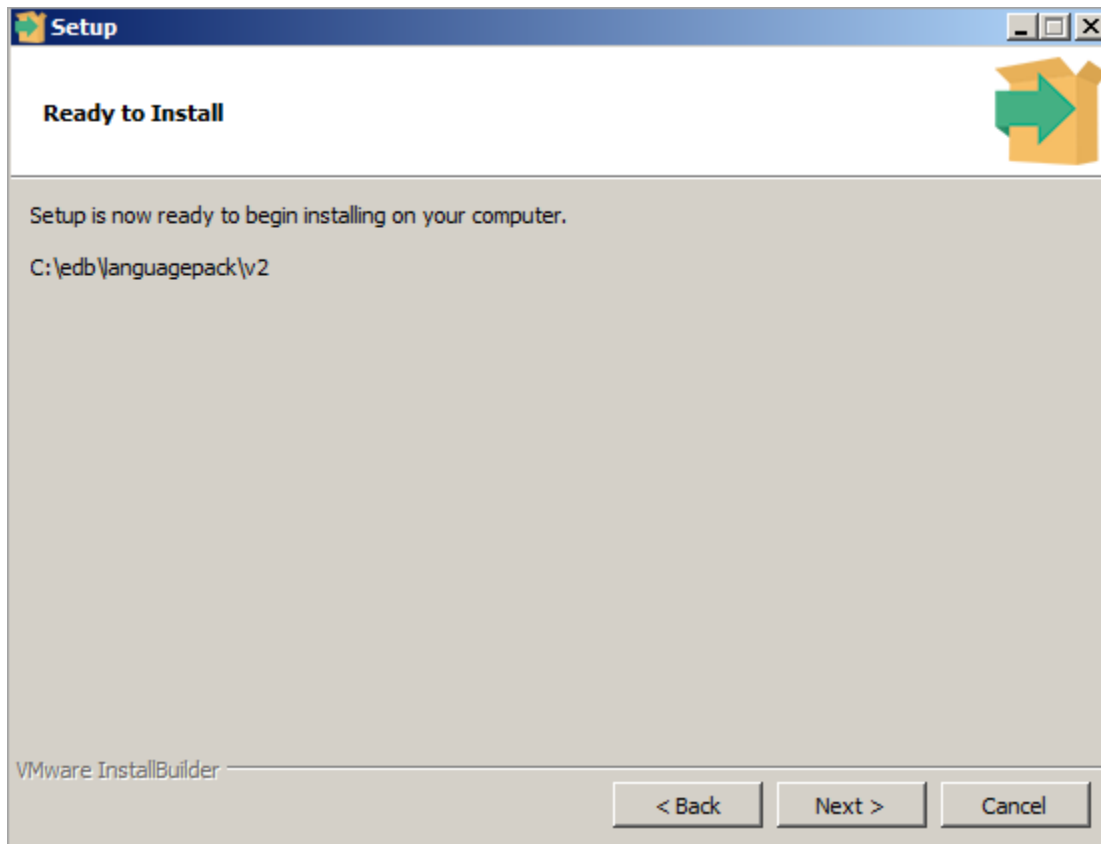


図 3: The Ready to Install dialog

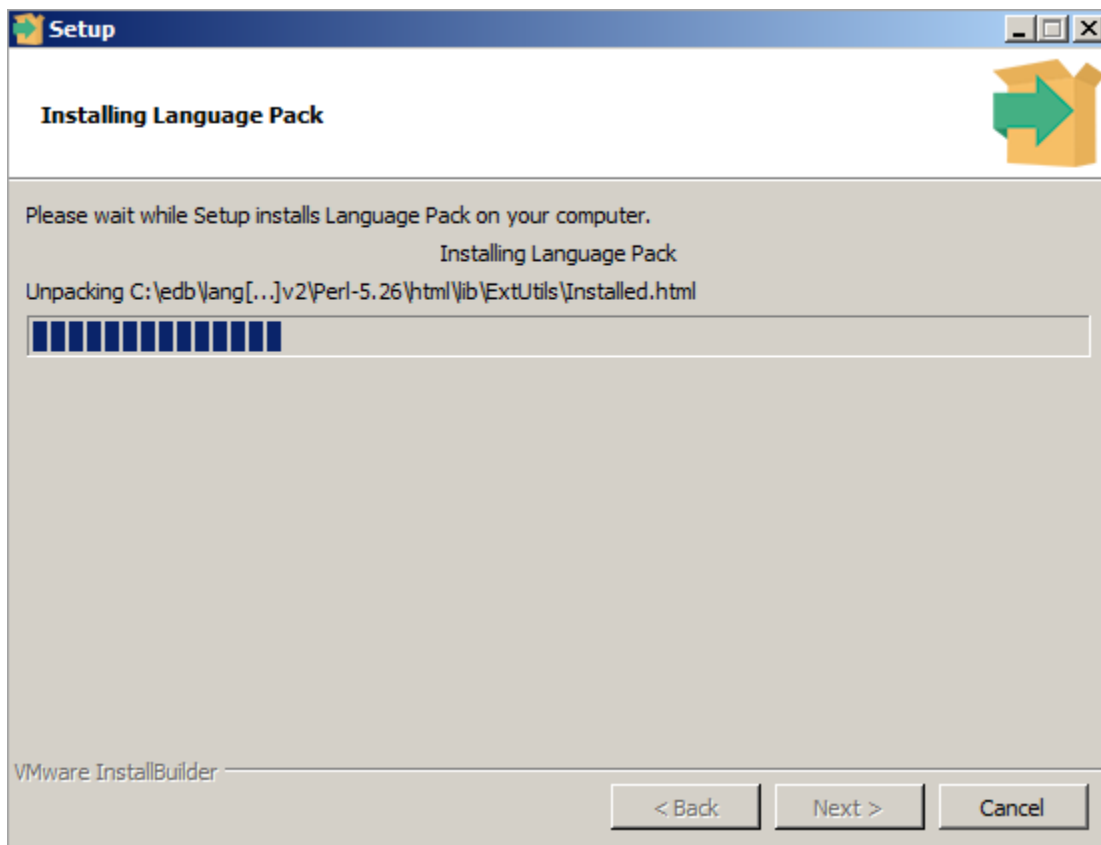


図 4: The Installing dialog

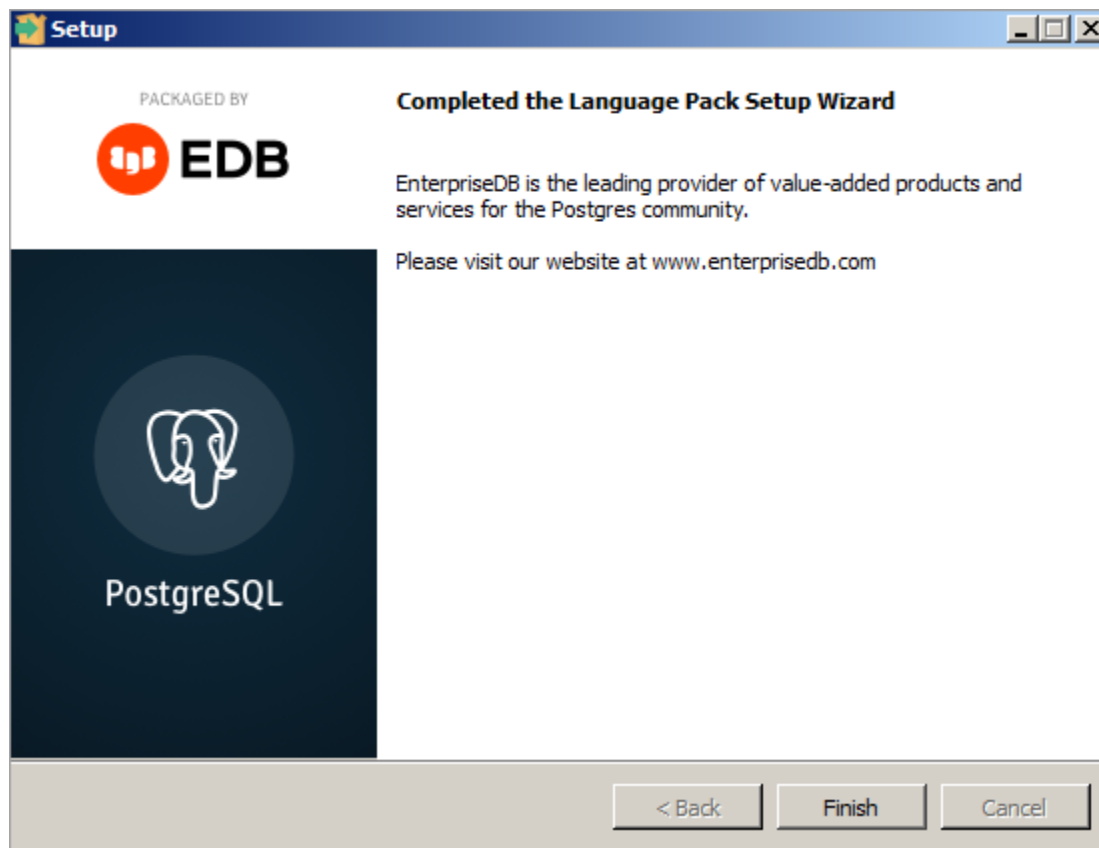


図 5: The Language Pack Setup Complete dialog

スタックビルダーで言語パックをインストールする

Stack Builder または StackBuilder Plus を使用して、言語パックのグラフィカルインストーラーをダウンロードして呼び出すことができます。StackBuilder Plus は EDB Postgres Advanced Server とともに配布され、Stack Builder は PostgreSQL とともに配布されます。

次のセクションでは、Stack Builder で Language Pack をインストールする手順について説明します。

Stack Builder ユーティリティは、PostgreSQL インストールを補完するモジュールのダウンロードと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ーフェイスを提供します。

スタックビルダーにはインターネットアクセスが必要です。PostgreSQL のインストールがファイアウォール（インターネットアクセスが制限されている）の内側にある場合、Stack Builder はプロキシサーバーを介して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 プロキシを介して転送され、FTP プロキシ情報は使用されません。

PostgreSQL 14 メニューから Application Stack Builder メニューオプションを選択することにより、インストールの完了後にいつでもスタックビルダーを呼び出すことができます。

Stack Builder の Welcome ウィンドウのドロップダウンメニューからサーバーを選択し、Next をクリックして続行します。

図 6：スタックビルダーのウェルカムウィンドウ

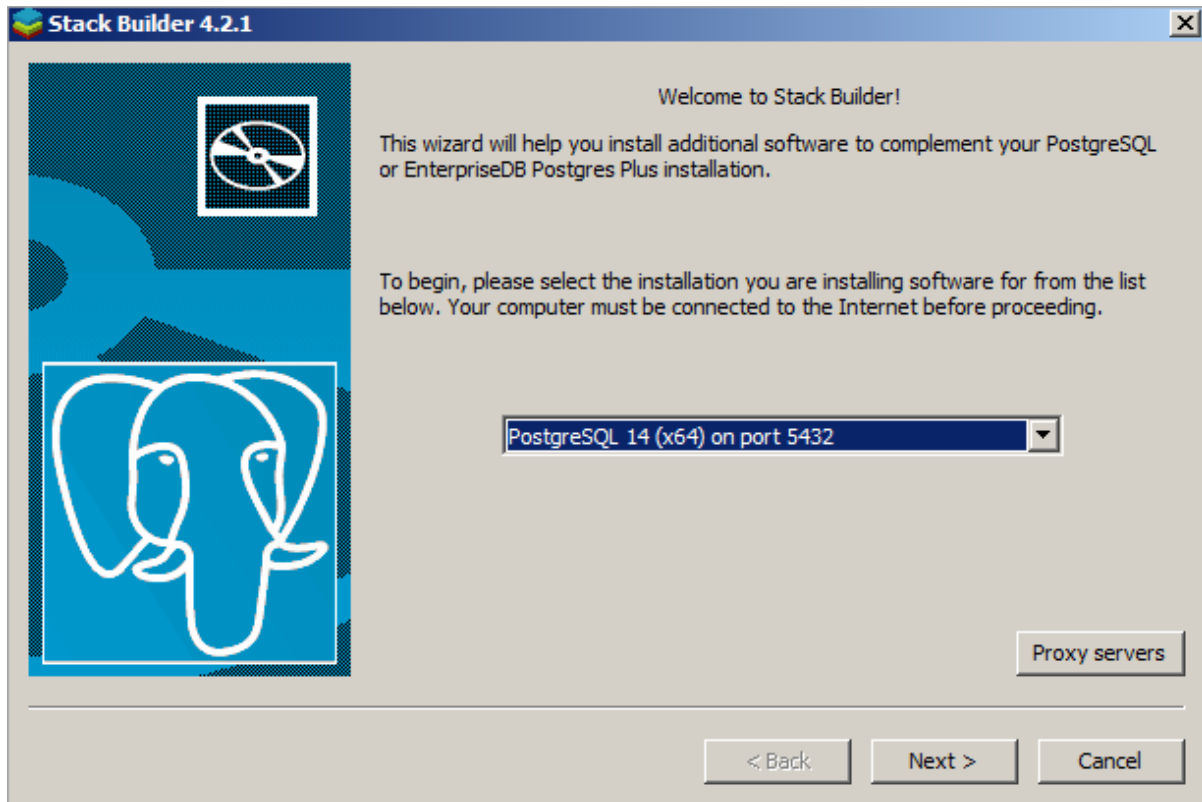


図 6: The Stack Builder Welcome Window

Categories ツリーコントロールの Add-ons, tools and utilities ノードを展開し、 EDB Language Pack v2.0-1 の横にあるボックスをオンにします。Next をクリックして続行します。

図 7: 言語パック選択ウィンドウ

Stack Builder は、インストーラーをダウンロードする前にパッケージの選択を確認します。Next をクリックして続行します。

図 8: 選択されたパッケージウィンドウ

ダウンロードが完了すると、Stack Builder はインストーラーを呼び出すか、より都合の良い時までインストールを遅らせるかを提案します。インストーラーを起動するには、Next をクリックし、 Invoking the Graphical Installer section に記載されている手順に従います。

図 9: StackBuilder のインストール完了

1.2.2 言語パックの構成

このセクションでは、EDB Postgres Advanced Server および PostgreSQL ホストで言語パックを構成する手順について説明します。

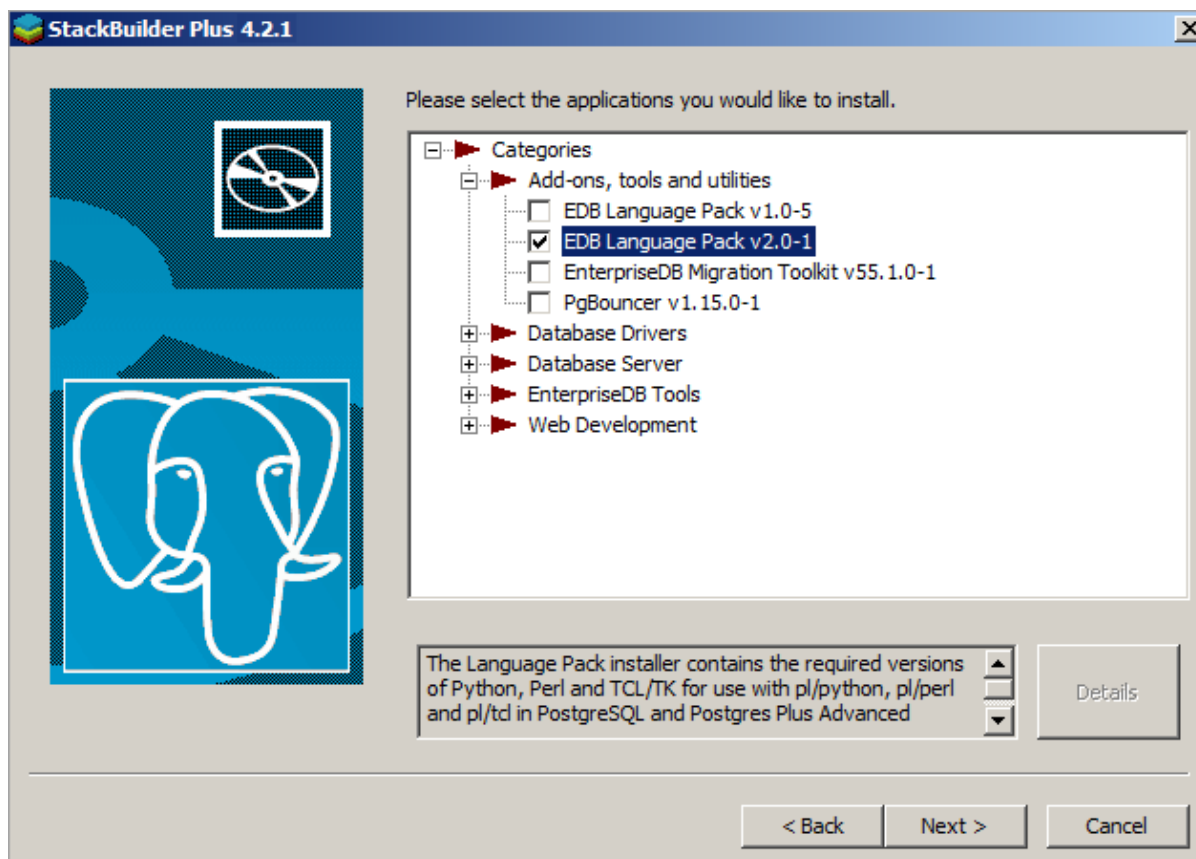


図 7: The Language Pack Selection Window

EDB Postgres Advanced Server ホストでの言語パックの構成

EDB Postgres Advanced Server ホストに言語パックをインストールした後、インストールを構成する必要があります。

Windows で言語パックを構成する

Windows では、言語パックインストーラーは言語を次の場所に配置します。

C:\edb\as\languagepack\v2

注釈: 実行している言語パックのバージョンが、ファイルパスおよび次のコマンドのバージョンと一致することを確認します。 !!!

言語パックをインストールした後、次の変数を設定する必要があります。

```
set PYTHONHOME=C:\edb\as\languagepack\v2\Python-3.7
```

次の手順に従って、Python、Perl、および Tcl を言語パックに追加します。

1. この PC>プロパティ>システムの詳細設定>環境変数 に移動します。
2. システム全体の PATH 変数を編集します。

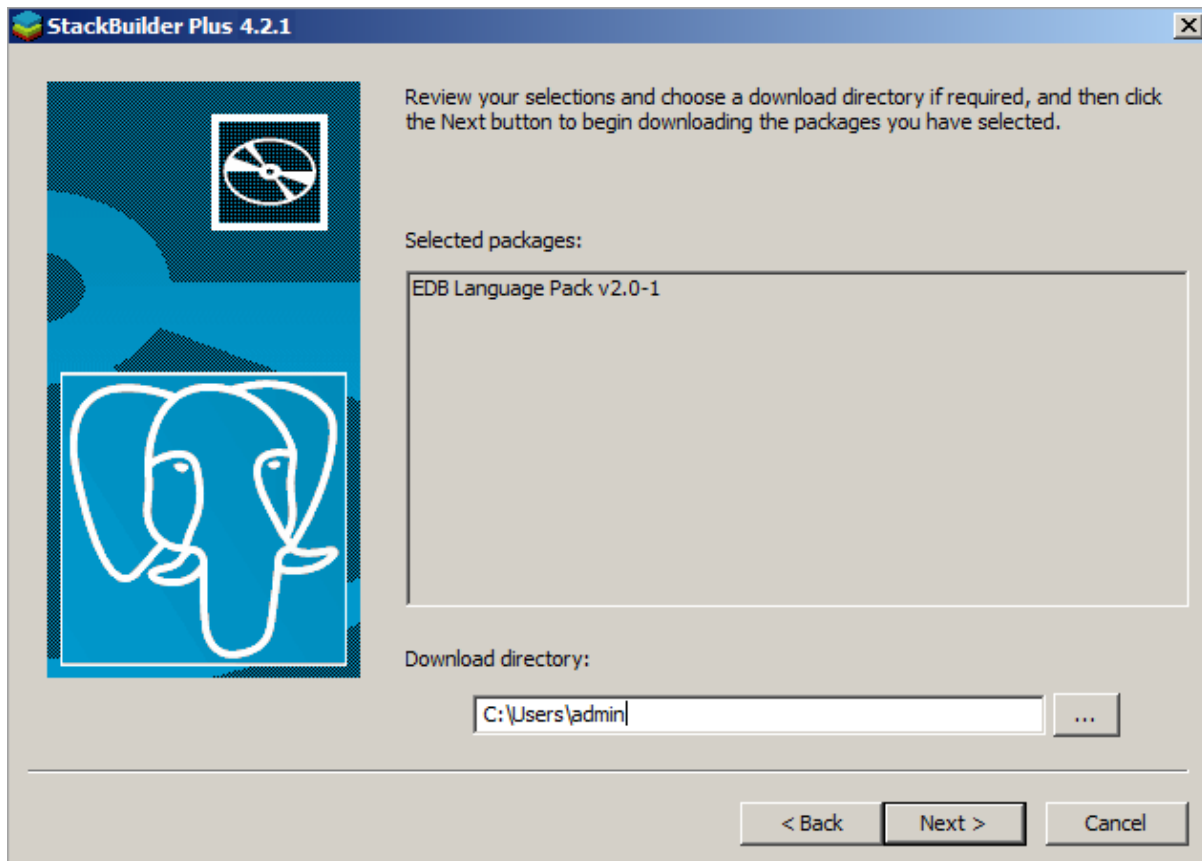


図 8: The Selected Packages Window

Python を追加するには、次のように入力します。

```
C:\edb\as\languagepack\v2\Python-3.7\  
C:\edb\as\languagepack\v2\Python-3.7\Scripts
```

Perl と Tcl を追加するには、次のように入力します。

```
C:\edb\as\languagepack\v2\Perl-5.26\bin  
C:\edb\as\languagepack\v2\Tcl-8.6\bin
```

1. ダイアログボックスを閉じます。

Windows Services アプレットを使用して EDB Postgres Advanced Server を再起動します。

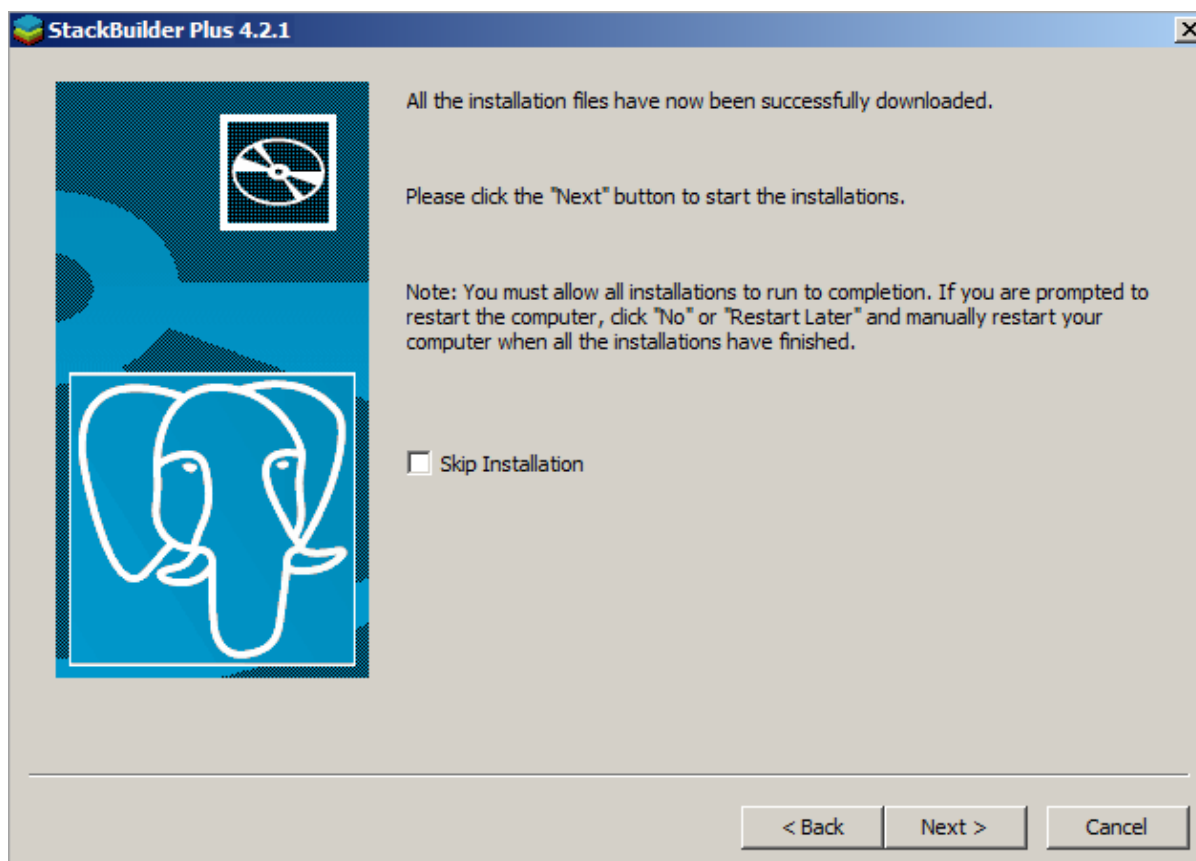


図 9: The Selected Packages Window

PostgreSQL ホストでの言語パックの構成

PostgreSQL ホストに言語パックをインストールした後、インストールを構成する必要があります。

Windows で言語パックを構成する

言語パックをインストールした後、次の変数を設定する必要があります。

```
set PYTHONHOME=C:\edb\languagepack\v2\Python-3.7
```

次の手順に従って、Python、Perl、および Tcl を言語パックに追加します。

1. この PC>プロパティ>システムの詳細設定>環境変数 に移動します。
2. システム全体の PATH 変数を編集します。

Python を追加するには、次のように入力します。

```
C:\edb\languagepack\v2\Python-3.7\
C:\edb\languagepack\v2\Python-3.7\Scripts
```

Perl と Tcl を追加するには、次のように入力します。

```
C:\edb\languagepack\v2\Perl-5.26\bin
C:\edb\languagepack\v2\Tcl-8.6\bin
```

1. ダイアログボックスを閉じます。
2. データベースサーバを再起動します。

OSX で言語パックを構成する

PATH または LD_LIBRARY_PATH の値の設定を簡素化するために、インストール場所を識別する環境変数を作成できます。

```
PERLHOME=/Library/edb/languagepack/v2/Perl-5.26
PYTHONHOME=/Library/edb/languagepack/v2/Python-3.7
TCLHOME=/Library/edb/languagepack/v2/Tcl-8.6
```

次に、次のコマンドを実行して、Python の場所を Python インタープリターに指示します。

```
export PYTHONHOME
```

PATH の値を設定するときに、同じ環境変数を使用できます。

```
export PATH=$PYTHONHOME/bin:
$PERLHOME/bin:
$TCLHOME/bin:$PATH
```

最後に、次の変数を設定して、共有ライブラリの場所を OSX に指示します。

```
export DYLD_LIBRARY_PATH=$PYTHONHOME/lib:
$PERLHOME/lib/CORE:$TCLHOME/lib:
$DYLD_LIBRARY_PATH
```

1.3 Using the procedural languages

Postgres 手続き型言語（PL / Perl、PL / Python、および PL / Tcl）は、言語パックインストーラーによってインストールされます。ネイティブパッケージを使用して、EDB Postgres Advanced Server インストールに手続き型言語機能を追加することもできます。

1.3.1 PL / Perl

PL / Perl 手続き型言語を使用すると、Postgres アプリケーションで Perl 関数を使用できます。

PL/Perl ファクションを作成する前に、各データベース（またはテンプレートデータベース）に PL/Perl をインストールする必要があります。EDB-PSQL コマンドラインで `CREATE LANGUAGE` コマンドを使用して、PL / Perl をインストールします。EDB-PSQL クライアントを開き、PL / Perl をインストールするデータベースへの接続を確立して、次のコマンドを入力します。

```
CREATE EXTENSION plperl;
```

Postgres クライアントアプリケーションを使用して、PL / Perl 言語の機能にアクセスできるようになりました。次の PL/Perl の例では、2つの整数値のうち大きい方を返す `perl_max` という名前のファクションを作成します。

```
CREATE OR REPLACE FUNCTION perl_max (integer, integer) RETURNS integer AS
$$
if ($_[0] > $_[1])
{ return $_[0]; }
return $_[1];
$$ LANGUAGE plperl;
```

関数を呼び出すときに2つの値を渡します。

```
SELECT perl_max(1, 2);
```

サーバーは次を返します。

```
__OUTPUT__
perl_max
- - - - -
      2
(1 row)
```

Perl 手続き言語の使用の詳細については、公式の [PostgreSQL documentation](#) を参照してください。

1.3.2 PL / Python

PL / Python 手続き型言語を使用すると、Postgres アプリケーション内で Python で記述された関数を作成して実行できます。EDB Postgres Advanced Server および PostgreSQL で使用される PL / Python のバージョンは信頼できません (`plpython3u`)。潜在的なセキュリティリスクを防ぐためのユーザーに対する制限はありません。

PL/Python ファクションを作成する前に、各データベース（またはテンプレートデータベース）に PL/Python をインストールします。EDB-PSQL コマンドラインで `CREATE LANGUAGE` コマンドを使用して、PL/Python をインストールできます。EDB-PSQL を使用して PL/Python をインストールするデータベースに接続し、次のコマンドを入力します。

```
CREATE EXTENSION plpython3u;
```

データベースに PL/Python をインストールすると、PL/Python 言語の機能を使用できます。

注釈: EDB-PSQL にサンプル関数を入力するときに、次の例に示すインデントを含める必要があります。

次の PL / Python の例では、2 つの整数値のうち大きい方を返す `pymax` という名前のファンクションを作成します。

```
CREATE OR REPLACE FUNCTION pymax (a integer, b integer) RETURNS integer AS
$$
if a > b:
return a
return b
$$ LANGUAGE plpython3u;
```

`pymax` 関数を呼び出すときは、以下に示す 2 つの値を渡します。

```
SELECT pymax(12, 3);
```

サーバーは次を返します。

```
__OUTPUT__
pymax
- - - - -
      12
(1 row)
```

Python 手続き言語の使用の詳細については、公式の [PostgreSQL documentation](#) を参照してください。

1.3.3 PL / Tcl

PL / Tcl 手続き型言語を使用すると、アプリケーションで Tcl / Tk 関数を使用できます。

PL / Tcl ファンクションを作成する前に、各データベース（またはテンプレートデータベース）に PL / Tcl をインストールする必要があります。EDB-PSQL コマンドラインで `CREATE LANGUAGE` コマンドを使用して、PL / Tcl をインストールします。psql クライアントを使用して、PL / Tcl をインストールするデータベースに接続し、次のコマンドを入力します。

```
CREATE EXTENSION pltcl;
```

`pltcl` 言語を作成したら、Postgres サーバー内から PL / Tcl 言語の機能を使用できます。

次の PL/Tcl の例では、2 つの整数値のうち大きい方を返す `tcl_max` という名前のファンクションを作成します。

```
CREATE OR REPLACE FUNCTION tcl_max(integer, integer) RETURNS integer AS
$$
if {[argisnull 1]} {
```

(次のページに続く)

(前のページからの続き)

```

if {[argisnull 2]} { return_null }
return $2
}
if {[argisnull 2]} { return $1 }
if {$1 > $2} {return $1}
return $2
$$ LANGUAGE plpgsql;

```

関数を呼び出すときに2つの値を渡します。

```
SELECT tcl_max(1, 2);
```

サーバーは次を返します。

```

__OUTPUT__
tcl_max
- - - - -
      2
(1 row)

```

Tcl 手続き言語の使用の詳細については、公式の [PostgreSQL documentation](#) を参照してください。

1.4 Uninstalling Language Pack

次のセクションでは、言語パックをアンインストールするプロセスの概要を説明します。

言語パックのグラフィカルインストーラーは、言語パックの削除に使用できるアンインストーラーを作成します。アンインストーラーはインストールディレクトリに作成されます。

言語パックをアンインストールするには、次の手順を実行します。

1. アンインストーラーを含むディレクトリに移動し、スーパーユーザー権限を引き受けます。アンインストーラーを開き、**Yes** をクリックして言語パックのアンインストールを開始します。

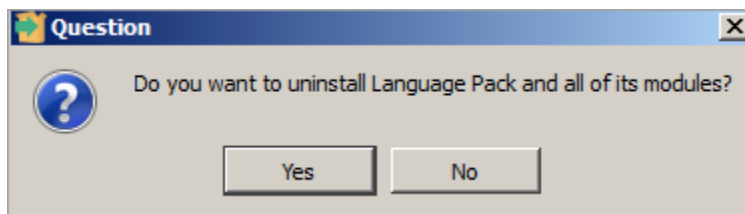


図 10: The Language Pack Uninstaller

図 1：言語パックのアンインストーラー

2. アンインストールが開始されます。アンインストールが完了したら **OK** をクリックします。

図 2：言語パックのアンインストール

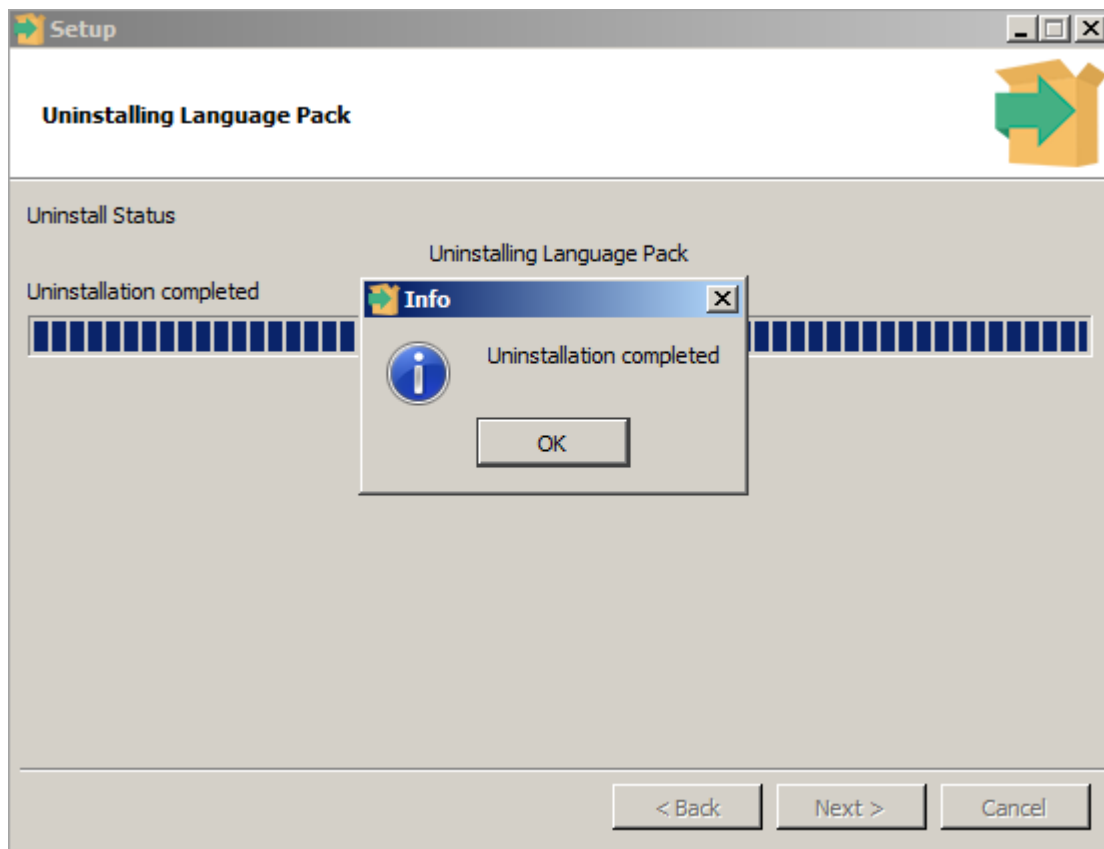


図 11: Uninstalling Language Pack