



EDB Postgres Language Pack Guide

リリース *14.1.0*

EnterpriseDB

2021 年 12 月 18 日

目次

第 1 章	Supported Database Server Versions	3
第 2 章	Installing and Configuring Language Pack	5
2.1	言語パックのインストール	5
2.1.1	グラフィカルインストーラーの呼び出し	5
2.1.2	Stack Builder で言語パックをインストールする	8
2.2	言語パックの構成	9
2.2.1	Advanced Server ホストでの言語パックの構成	10
2.2.2	PostgreSQL ホストでの言語パックの構成	11
第 3 章	Using the Procedural Languages	13
3.1	PL / Perl	13
3.2	PL / Python	14
3.3	PL / Tcl	15
第 4 章	Uninstalling Language Pack	17

このガイドでは、言語パックをインストールおよび構成する方法と、手続き言語（PL / Perl、PL / Python、および PL / TCL）を使用する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が含まれ、Python には `pip` および `easy_install` パッケージマネージャが含まれています。TCL / TK 用のパッケージマネージャはありません。

以前の Postgres リリースでは、`plpython` は ActiveState の `python` ライブラリと静的にリンクされていました。Language Pack Installer は、Python の共有オブジェクトと動的にリンクします。Python 用 ActiveState Linux インストーラーには、動的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す。

このガイドで使用されている規約

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

第1章 Supported Database Server Versions

</ div>

言語パックインストーラーは、バージョンおよびプラットフォーム固有です。EDB Postgres Advanced Server または PostgreSQL サーババージョンに対応する言語パックインストーラを選択します。

**** Linux : ****

EDB Postgres Advanced Server/PostgreSQL Version	Language Pack Version	Procedural Language Version
9.6, 10	1.0	Perl 5.26, Python 3.7, Tcl 8.6

RPM パッケージを使用して言語パックを追加する方法の詳細については、[EDB website](#) にある * EDB Postgres Advanced Server インストールガイド*を参照してください。

**** Mac OS : ****

PostgreSQL Version	Language Pack Version	Procedural Language Version
9.6, 10, 11, 12, 13	1.0	Perl 5.26, Python 3.7, Tcl 8.6

**** Windows 32 : ****

EDB Postgres Advanced Server/PostgreSQL Version	Language Pack Version	Procedural Language Version
9.6, 10	1.0	Perl 5.26, Python 3.7, Tcl 8.6

**** Windows 64 : ****

EDB Postgres Advanced Server/PostgreSQL Version	Language Pack Version	Procedural Language Version
<i>PostgreSQL 9.6, 10, 11, 12, 13</i>	1.0	Perl 5.26, Python 3.7, Tcl 8.8
<i>EDB Postgres Advanced Server 12</i>	1.0	Perl 5.26, Python 3.7, Tcl 8.6

第2章 Installing and Configuring Language Pack

</ div>

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。

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は、[EDB website](#) から利用できるだけでなく、Stack Builder および StackBuilder Plus から利用できます。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。

</ div>

2.1.1 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、管理者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。

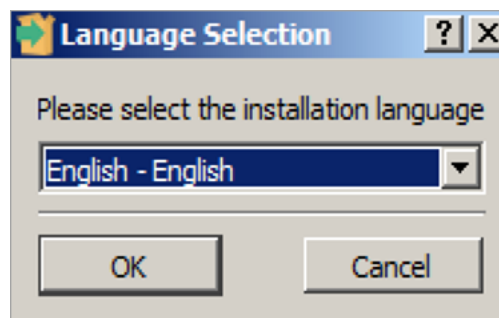


図 1: The Language Selection Window

図 1：言語選択ウィンドウ</ div>

言語パックセットアップウィザードのようこそウィンドウが開きます。

図 2：言語パックのウェルカムウィンドウ</ div>

Next をクリックして続行します。

[インストールの準備完了] 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

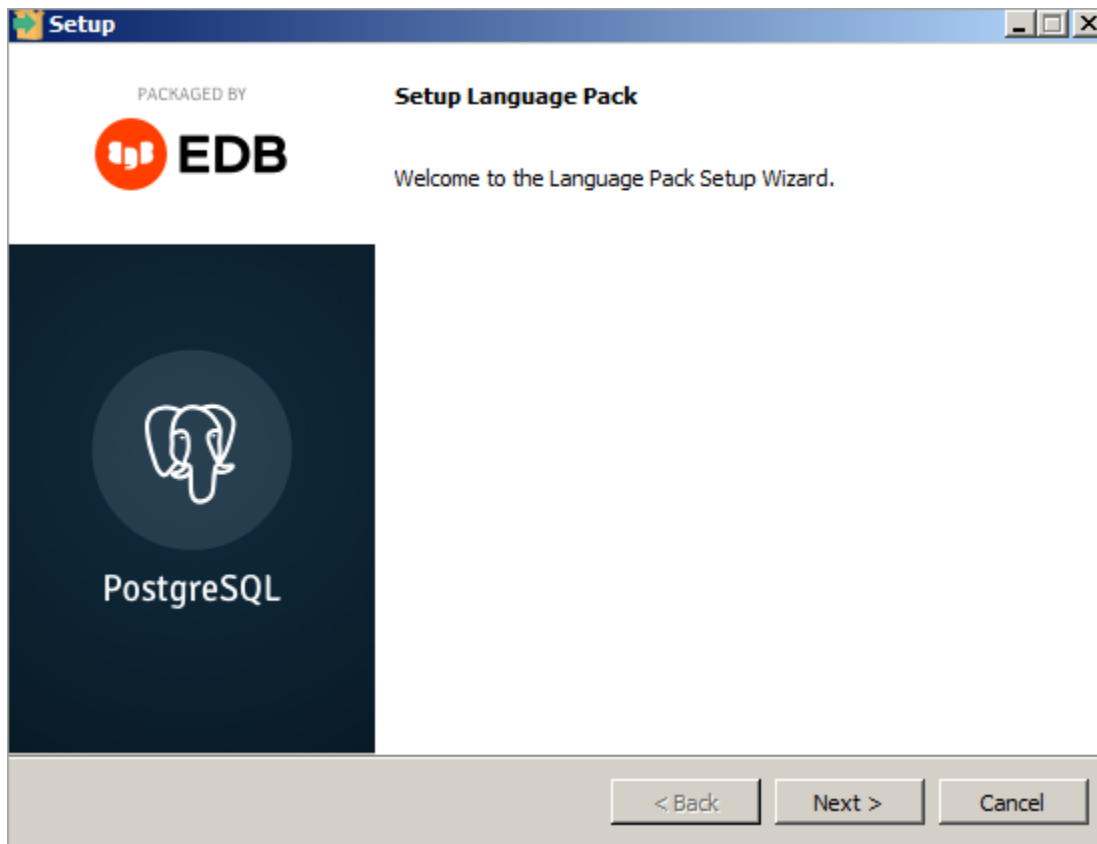


図 2: The Language Pack Welcome Window

OSX の場合：/Library/edb/languagepack/v2

図 3：インストールの準備ダイアログ</div>

インストールディレクトリを変更することはできません。**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</div>

Next をクリックして続行します。

インストーラは、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、インストーラを終了します。

図 5：言語パックのセットアップ完了ダイアログ</div>

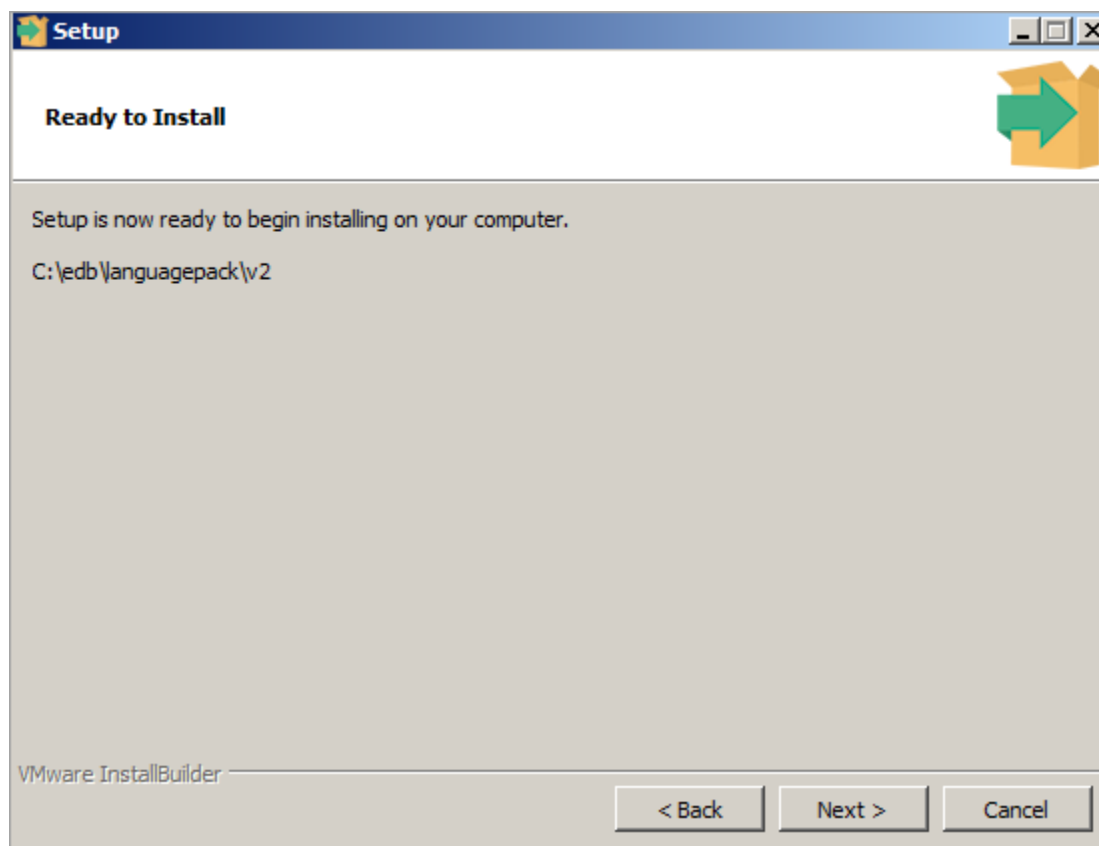


図 3: The Ready to Install dialog

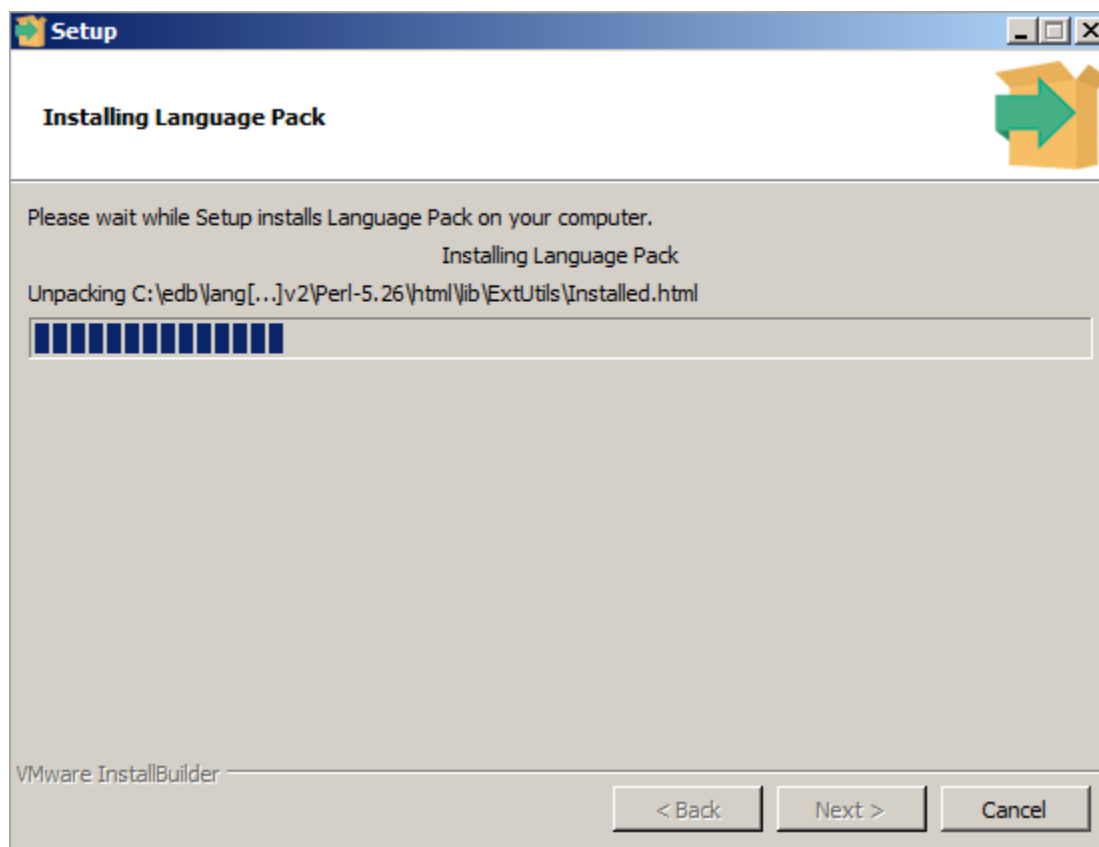


図 4: The Installing dialog

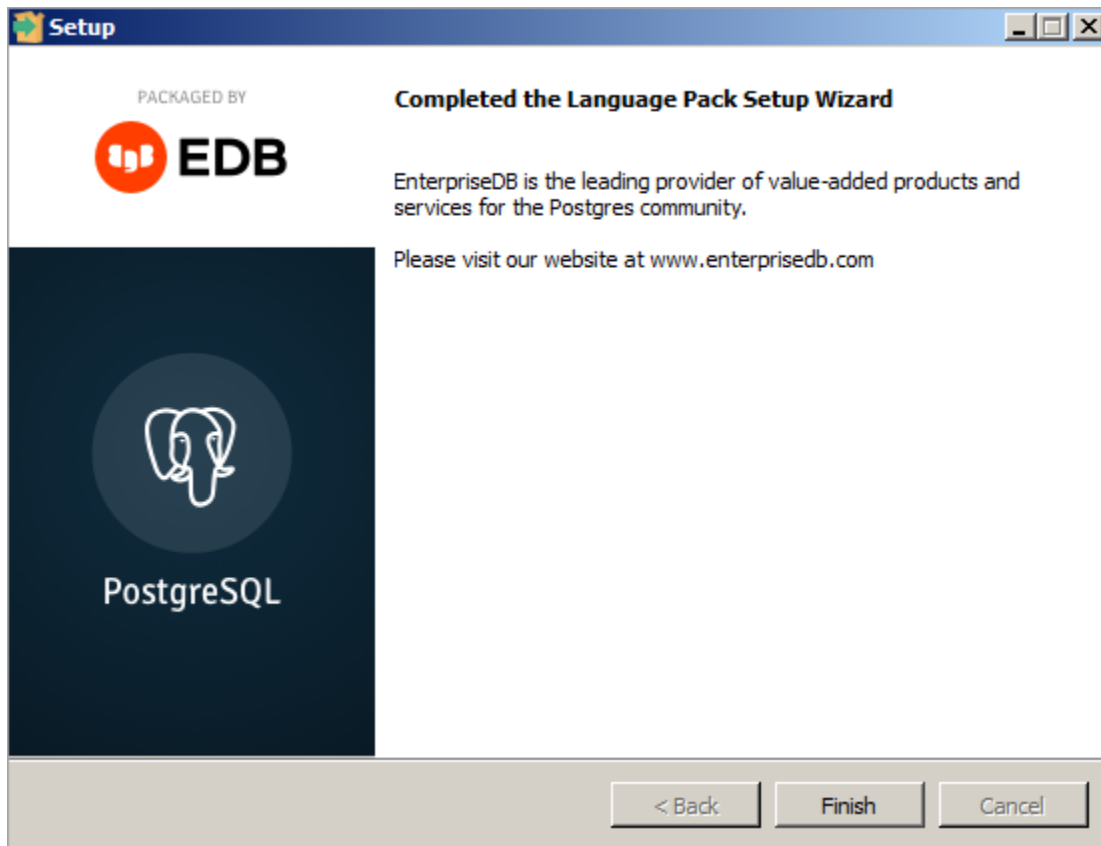


図 5: The Language Pack Setup Complete dialog

2.1.2 Stack Builder 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、Language Pack グラフィカルインストーラをダウンロードして起動できます。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 を使用した言語パックのインストール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フェースを提供します。

Stack Builder 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の内側にある場合（インターネットアクセスが制限されている場合）、Stack Builder はプロキシサーバー経由でプログラムインストーラ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決定します。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。

インストールが完了した後、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で、いつでも Stack Builder を呼び出すことができます。

Stack Builder 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：Stack Builder Welcome ウィンドウ

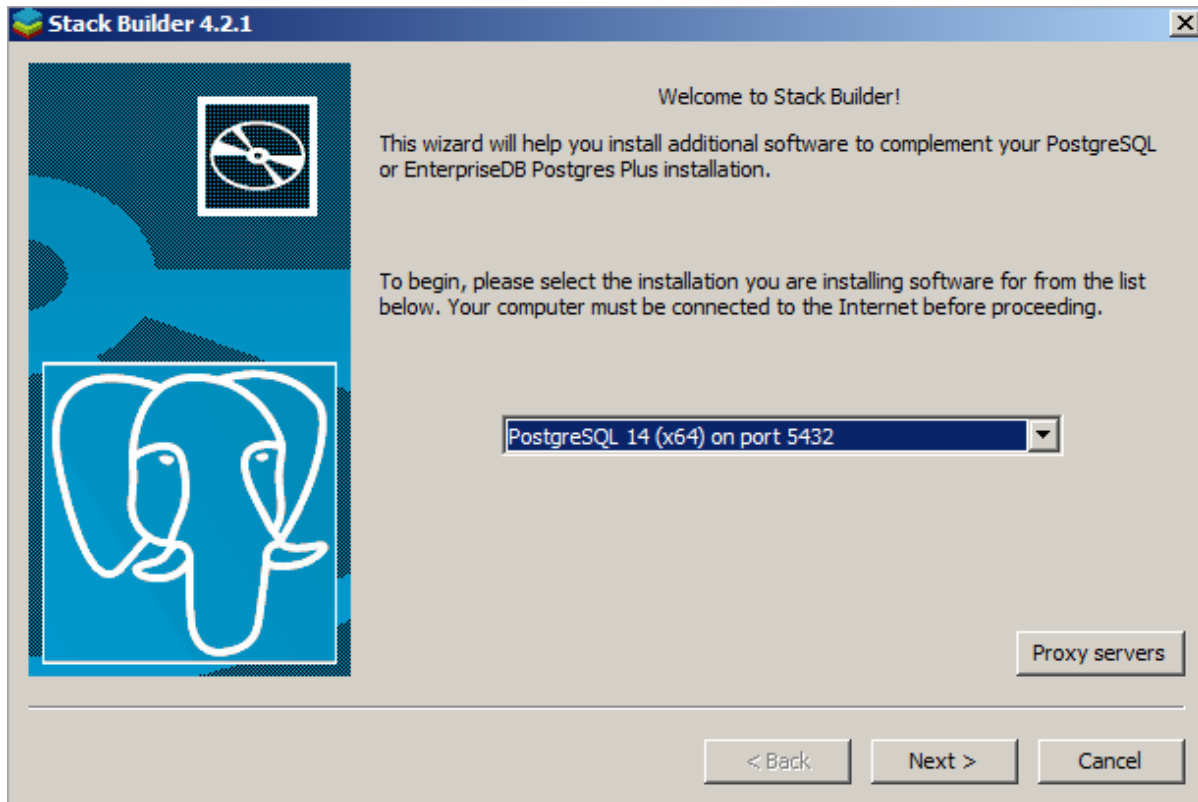


図 6: The Stack Builder Welcome Window

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 の隣の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: 選択し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 はインストーラの呼び出しを提案するか、より都合の良い時間までインストールを遅らせます。インストーラを起動するには、Next をクリックして、*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

2.2 言語パックの構成

このセクションでは、Advanced Server および PostgreSQL ホストでの言語パックの構成について説明します。

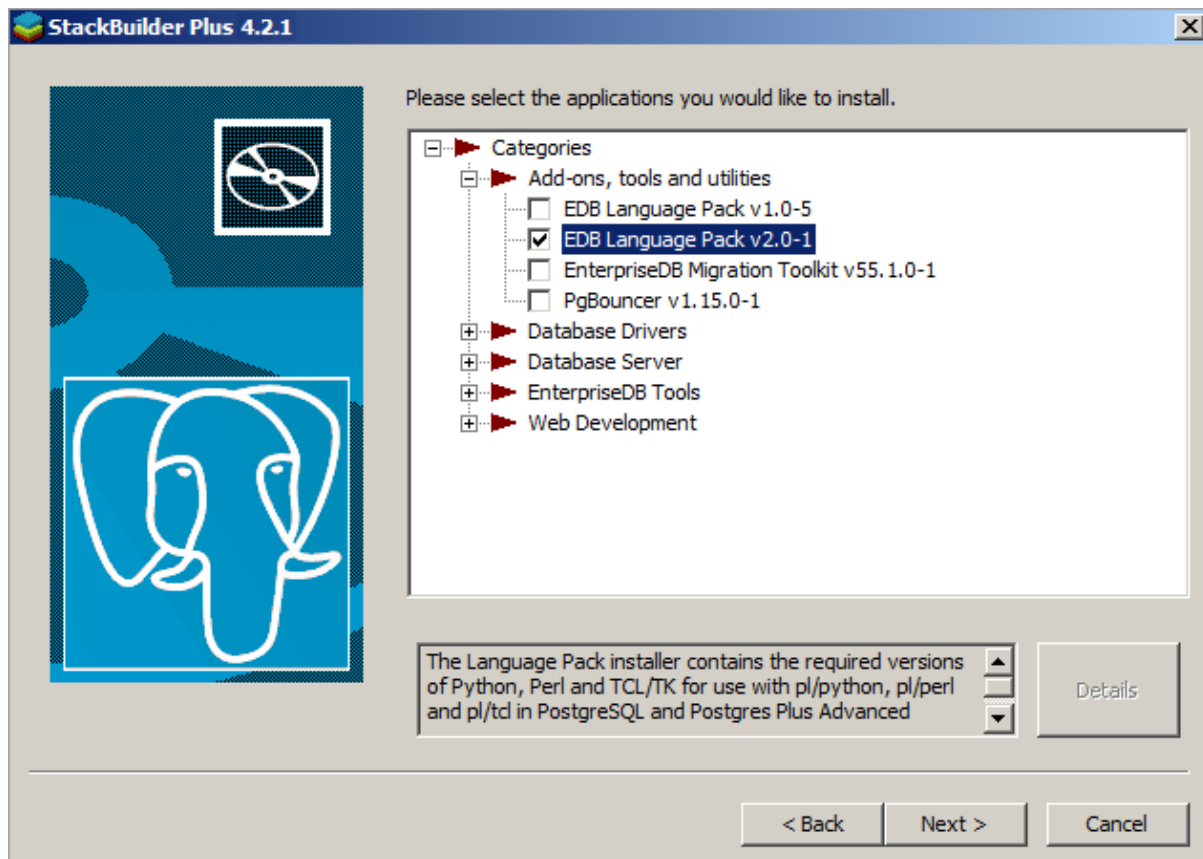


図 7: The Language Pack Selection Window

2.2.1 Advanced Server ホストでの言語パックの構成

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 での言語パックの構成****

Windows では、言語パックインストーラは言語を次の場所に配置します。

C:\edb\languagepack\v2

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
```

次のコマンドを使用して、Python、Perl、および Tcl を検索パスに追加します。

```
set PATH=C:\edb\languagepack\v2\Python-3.7;
C:\edb\languagepack\v2\Perl-5.26\bin;
C:\edb\languagepack\v2\Tcl-8.6\bin;%PATH%
```

Windows で言語パックを構成するために必要な手順を実行した後、Windows Services アプレットを使用して Advanced Server をリスタートします。

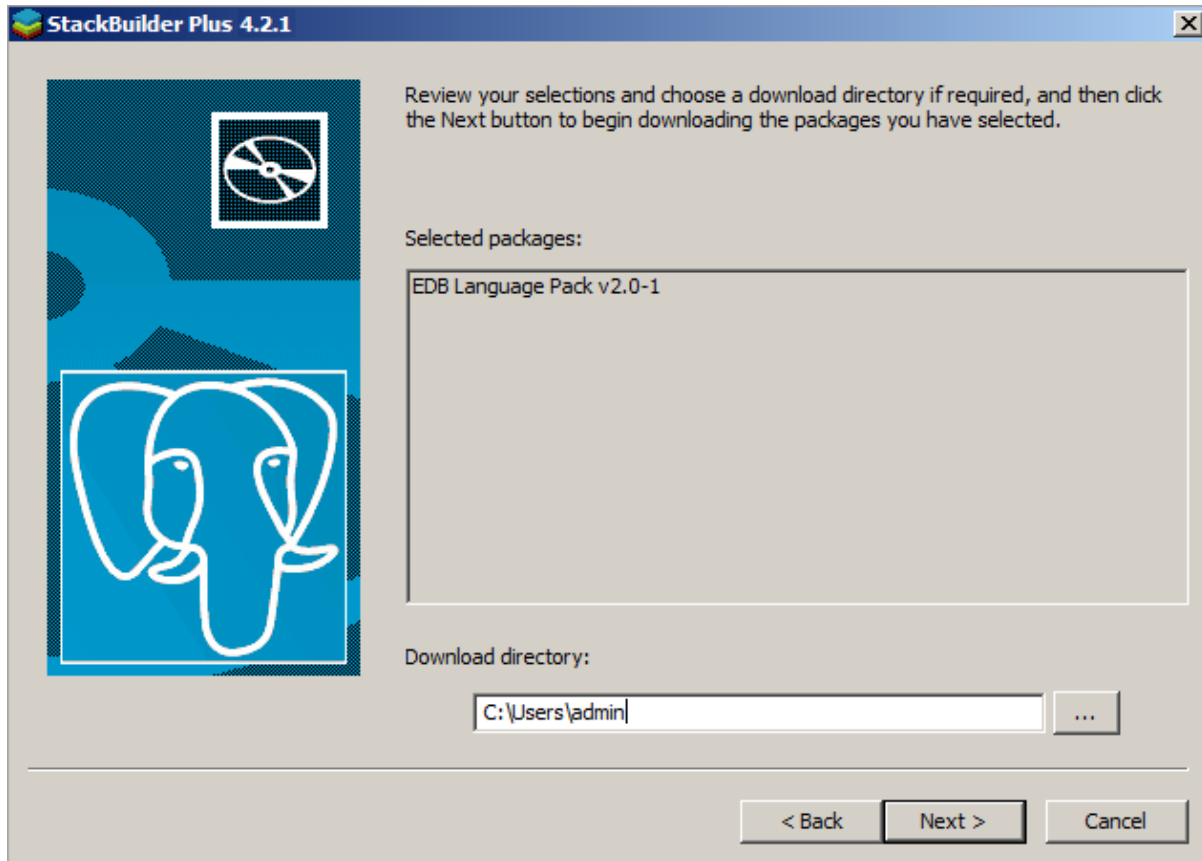


図 8: The Selected Packages Window

2.2.2 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 での言語パックの構成**

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
```

次に、次のコマンドを使用して、検索パスに言語パックを追加します。

```
set PATH=C:\edb\languagepack\v2\Python-3.7;  
C:\edb\languagepack\v2\Perl-5.26\bin;  
C:\edb\languagepack\v2\Tcl-8.6\bin;%PATH%
```

Windows で言語パックを構成するために必要なシステム固有の手順を設定した後、データベースサーバをリスタートします。

** OSX での言語パックの構成**

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

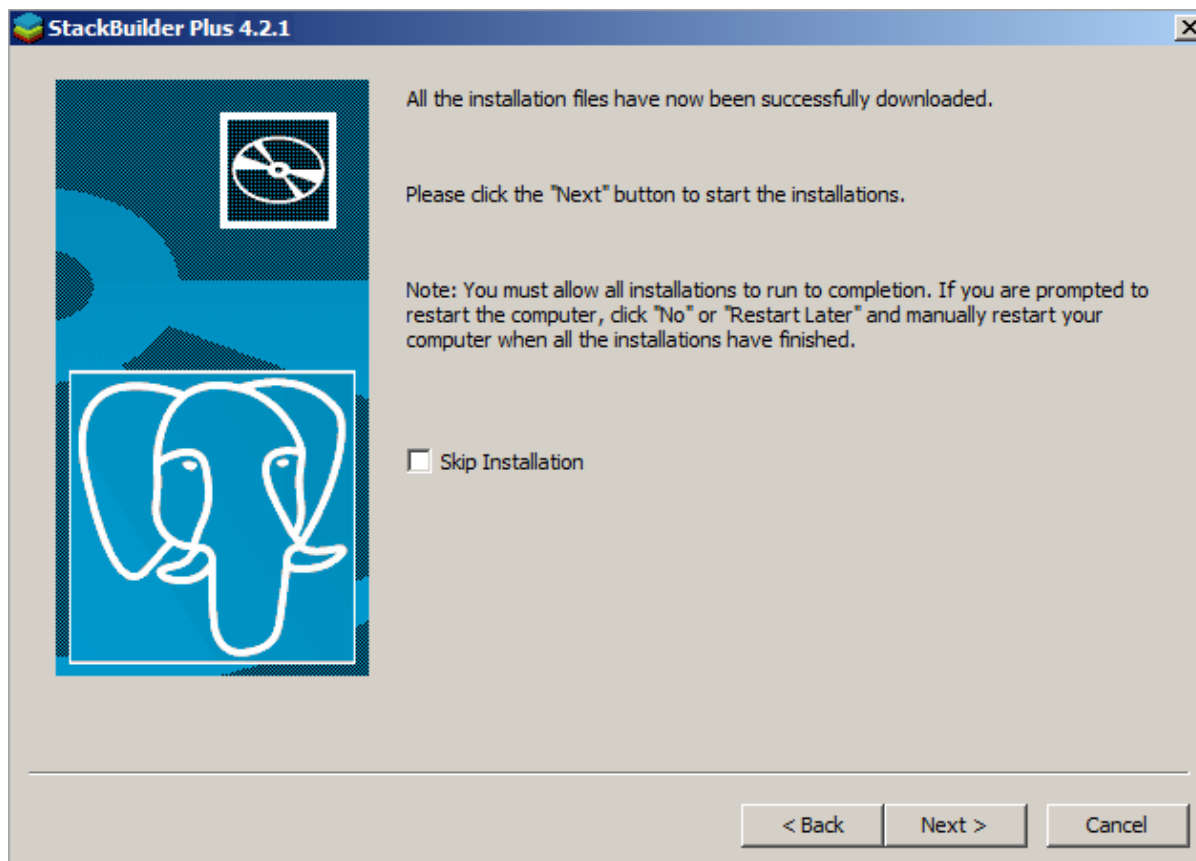


図 9: The Selected Packages Window

```
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 インタプリタに Python の検索場所を指示します。

```
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、OSX に共有ライブラリの場所を指示します。

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

第3章 Using the Procedural Languages

</div>

Postgres 手続き言語（PL / Perl、PL / Python、および PL / Java）は、言語パックインストーラによってインストールされます。RPM パッケージを使用して、プロシージャ言語機能を EDB Postgres Advanced Server インストールに追加することもできます。RPM パッケージの使用の詳細については、[EDB website](#) にある* EDB Postgres Advanced Server インストールガイド*を参照してください。

3.1 PL / Perl

PL / Perl プロシージャ言語を使用すると、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 / 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、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 の例は、2つの整数値のうち大きい方を返す `perl_max` という名前付けのファン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;
```

ファンクションを呼び出すときに2つの値を渡します。

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

サーバーは以下を返します。

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

Perl プロシージャ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。

3.2 PL / Python

PL / Python プロシージャ言語を使用すると、Postgres アプリケーション内で Python で記述された関数を作成および実行できます。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されていません (plpython3u)。潜在的なセキュリティリスクを防止するためのユーザーに対する制限はありません。

PL / Python 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 / 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 をインストールした後、PL / Python 言語の機能を使用できます。

!!! Note * 次の例に示すインデントは、EDB -PSQL でサンプルファンクションを入力するときに含める必要があります。

次の PL / Python の例は、2つの整数値のうち大きい方を返す `pymax` という名前付け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` ファンクションを呼び出すとき、次のように2つの値を渡します。

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

サーバーは以下を返します。

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

Python プロシージャ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。

3.3 PL / Tcl

PL / Tcl プロシージャ言語を使用すると、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 言語を作成した後、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 の例は、2 つの整数値のうち大きい方を返す `tcl_max` という名前付け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
$$
if {[argisnull 1]} {
if {[argisnull 2]} { return_null }
return $2
}
if {[argisnull 2]} { return $1 }
if {$1 > $2} {return $1}
return $2
$$ LANGUAGE pltcl;
```

ファンクションを呼び出すときに 2 つの値を渡します。

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

サーバーは以下を返します。

```
tcl_max
-----
*      2
(1 row)
```

Tcl プロシージャ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。

第4章 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は、言語パックを削除するため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、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。

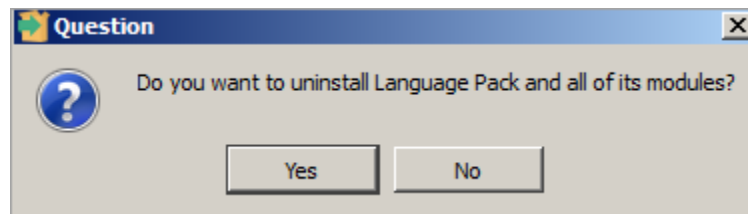


図 1: The Language Pack Uninstaller

図 1：言語パックアンインストーラー

2. アンインストールプロセスが開始されます。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

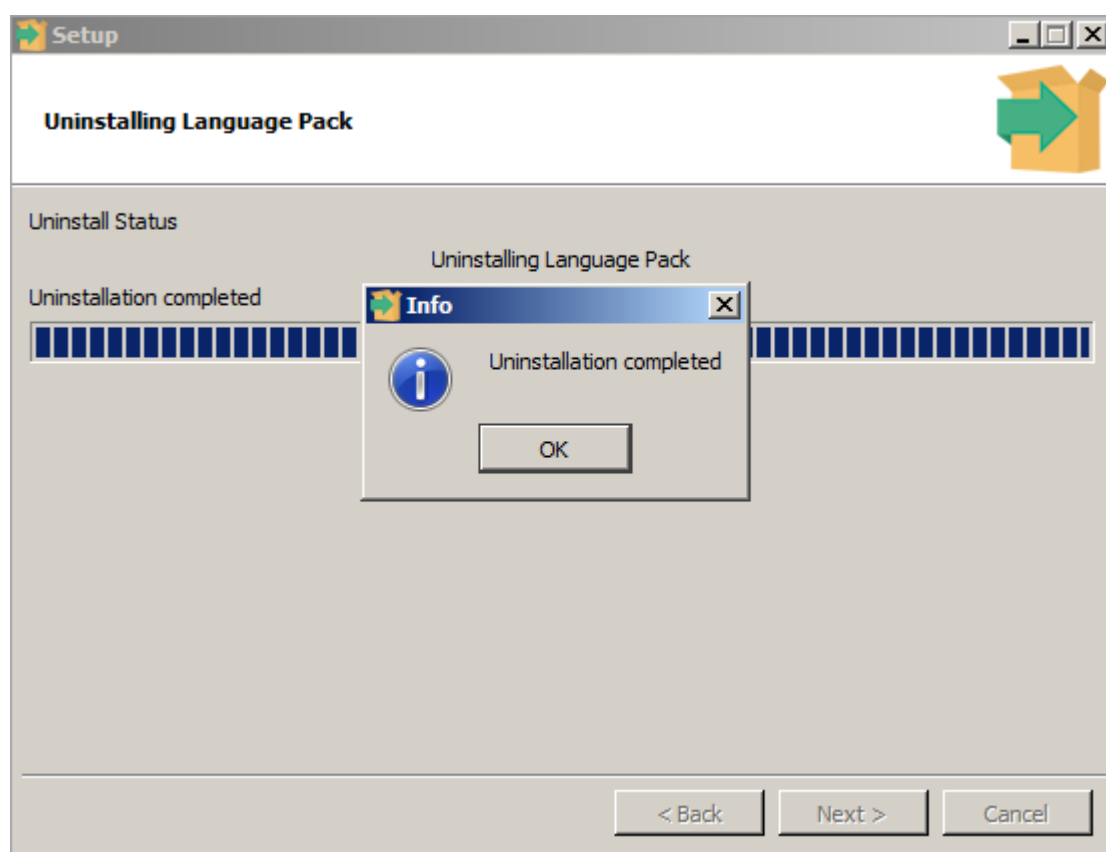


図 2: Uninstalling Language Pack