



EDB

EDB Postgres PostGIS

Release 3.2.0

EnterpriseDB Corporation

Sep 09, 2022

CONTENTS

1	Release notes	3
1.1	Version 3.2.1	3
1.2	Version 3.2.0	3
1.3	Version 3.1.5	4
1.4	Version 3.1.4	4
1.5	Version 3.1.2	4
2	Supported platforms	5
3	Installing PostGIS	7
3.1	Installing PostGIS on Linux	7
3.1.1	Installing PostGIS on Linux x86 (amd64)	8
3.1.2	Installing PostGIS on IBM Power (ppc64le)	15
3.2	Installing PostGIS on Windows	17
4	Creating extensions	25
5	Upgrading PostGIS	27
5.1	Example 1: Upgrading from PostGIS version 3.1 to 3.2	27
5.2	Example 2: Upgrading from PostGIS versions 2.5 or later to 3.1.4 on the same EDB Postgres Advanced Server version	28
5.3	Example 3: Upgrading from PostGIS versions earlier than 2.5 to 3.1.4 on a different EDB Postgres Advanced Server version	29
6	Using PostGIS	35
7	Uninstalling PostGIS	37
7.1	Uninstalling PostGIS on a CentOS/RHEL/Rocky Linux/AlmaLinux host	37
7.2	Uninstalling PostGIS on a Debian/Ubuntu host	37
7.3	Uninstalling PostGIS on a SLES host	38
7.4	Uninstalling PostGIS on a Windows host	38

EDB PostGIS is a PostgreSQL extension that allows you to store geographic information systems (GIS) objects in an EDB Postgres Advanced Server database. It includes functions for analyzing and processing GIS objects and support for GiST-based R-Tree spatial indexes.

The complete documentation for PostGIS is added to your system during the installation and is available online at the [PostGIS project site](#) .

RELEASE NOTES

The EDB PostGIS documentation describes the latest version of EDB PostGIS 3.2, including minor releases and patches. These release notes cover what was new in each release.

Version	Release date
<i>Version 3.2.1</i>	2022 Aug 04
<i>Version 3.2.0</i>	2022 Dec 01
<i>Version 3.1.5</i>	2022 Aug 03
<i>Version 3.1.4</i>	2021 Dec 01
<i>Version 3.1.2</i>	2021 Jun 24

1.1 Version 3.2.1

EDB PostGIS is a PostgreSQL extension that allows you to store geographic information systems (GIS) objects in an EDB Postgres Advanced Server database.

New features, enhancements, bug fixes, and other changes in PostGIS 3.2.1 include:

Type	Description
Upstream merges	Merged with community PostGIS 3.2.1. See the community Release notes for details.

1.2 Version 3.2.0

EDB PostGIS is a PostgreSQL extension that allows you to store geographic information systems (GIS) objects in an EDB Postgres Advanced Server database.

New features, enhancements, bug fixes, and other changes in PostGIS 3.2.0 include:

Type	Description
Upstream merges	Merged with community PostGIS 3.2.0. See the community Release notes for details.

1.3 Version 3.1.5

EDB PostGIS is a PostgreSQL extension that allows you to store geographic information systems (GIS) objects in an EDB Postgres Advanced Server database.

New features, enhancements, bug fixes, and other changes in PostGIS 3.1.5 include:

Type	Description
Upstream merges	Merged with community PostGIS 3.1.5. See the community Release notes for details.

https://postgis.net/docs/manual-3.1/release_notes.html#idm41935

1.4 Version 3.1.4

EDB PostGIS is a PostgreSQL extension that allows you to store geographic information systems (GIS) objects in an EDB Postgres Advanced Server database.

New features, enhancements, bug fixes, and other changes in PostGIS 3.1.4 include:

Type	Description
Upstream merges	Merged with community PostGIS 3.1.4. See the community Release notes for details.

1.5 Version 3.1.2

EDB PostGIS is a PostgreSQL extension that allows you to store geographic information systems (GIS) objects in an EDB Postgres Advanced Server database.

New features, enhancements, bug fixes, and other changes in PostGIS 3.1.4 include:

Type	Description
Upstream merges	Merged with community PostGIS 3.1.2. See the community Release notes for details.

Note: To upgrade from PostGIS version 2.5.4 (or lower) to 3.1.2, you must perform the following actions: - Since the return type of the raster functions have changed, you must drop and re-create the raster extension as part of the upgrade process. - Before upgrading to version 3.1.2, you must upgrade to 3.1.1 first as an intermediate step. - When the PostGIS data has a dependency on the raster functions, upgrading to PostGIS 3.1.2 requires dumping and reloading the data.

See the upgrade section for details.

SUPPORTED PLATFORMS

This table lists the latest PostGIS versions and their supported corresponding EDB Postgres Advanced Server (EPAS) versions. PostGIS is supported on the same platforms as EDB Postgres Advanced Server. See

[Product Compatibility](#) for details.

PostGIS	EPAS 14	EPAS 13	EPAS 12	EPAS 11	EPAS 10
3.2	Y	Y	Y	Y	N
3.1	Y	Y	Y	Y	Y
3.0	Y	Y	Y	Y	N
2.5	N	N	Y	Y	N
2.4	N	N	N	Y	Y
2.3	N	N	N	N	Y

INSTALLING POSTGIS

You can install PostGIS on:

- *Installing PostGIS on Linux*
- *Installing PostGIS on Windows*

3.1 Installing PostGIS on Linux

To install PostGIS, you must have credentials that allow access to the EnterpriseDB repository. To request credentials that allow you to access an EnterpriseDB repository, see the [EDB Repository Access instructions](#) .

For platform-specific install instructions, see:

Linux x86-64 (amd64):

- RHEL 8/OL 8
- Rocky Linux 8/AlmaLinux 8
- RHEL 7/OL 7
- *Installing PostGIS on SLES 15 ppc64le*
- *Installing PostGIS on SLES 12 ppc64le*
- *Installing PostGIS on Ubuntu 20.04 x86_64*
- *Installing PostGIS on Ubuntu 18.04 x86_64*
- *Installing PostGIS on Debian 10 x86_64*

Linux on IBM Power (ppc64le):

- *Installing PostGIS on RHEL 8 ppc64le*
- *Installing PostGIS on RHEL 7 or OL 7 x86_64 s*
- *Installing PostGIS on SLES 15 ppc64le*
- *Installing PostGIS on SLES 12 ppc64le*

3.1.1 Installing PostGIS on Linux x86 (amd64)

For operating system-specific install instructions, see:

- RHEL 8/OL 8
- Rocky Linux 8/AlmaLinux 8
- *Installing PostGIS on CentOS 7 x86_64*
- RHEL 7/OL 7
- *Installing PostGIS on SLES 15 ppc64le*
- *Installing PostGIS on SLES 12 ppc64le*
- *Installing PostGIS on Ubuntu 20.04 x86_64*
- *Installing PostGIS on Ubuntu 18.04 x86_64*
- *Installing PostGIS on Debian 10 x86_64*

Installing PostGIS on RHEL 8 or OL 8 x86_64

Prerequisites

Before you begin the installation process:

- **Install EDB Postgres Advanced Server on the same host. See**
Installing EDB Postgres Advanced Server .
- Set up the repository

Setting up the repository is a one-time task. If you have already set up your repository, you do not need to perform this step.

To set up the repository, go to [EDB repositories](#) and follow the instructions provided there.

- Address other prerequisites

```
# Install the EPEL repository:
sudo dnf -y install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.
↪rpm

# Enable additional repositories to resolve dependencies:
ARCH=$( /bin/arch ) subscription-manager repos --enable "codeready-builder-for-rhel-8-$
↪{ARCH}-rpms"

# Disable the built-in PostgreSQL module:
sudo dnf -qy module disable postgresql
```

Install the package

```
# To install PostGIS 3.2:

sudo dnf -y install edb-as<xx>-postgis32

# To install PostGIS 3.1 using EDB Postgres Advanced Server 13-14:

sudo dnf -y install edb-as<xx>-postgis3

# To install PostGIS 3.1 using EDB Postgres Advanced Server 11-12:

sudo dnf -y install edb-as<xx>-postgis
```

Where <xx> is the version of EDB Postgres Advanced Server. Replace <xx> with the version of EDB Postgres Advanced Server you are using. For example, edb-as14-postgis32 .

Installing PostGIS on AlmaLinux 8 or Rocky Linux 8 x86_64

Prerequisites

Before you begin the installation process:

- **Install EDB Postgres Advanced Server on the same host. See**
Installing EDB Postgres Advanced Server .
- Set up the repository

Setting up the repository is a one-time task. If you have already set up your repository, you do not need to perform this step.

To set up the repository, go to [EDB repositories](#) and follow the instructions provided there.

- Address other prerequisites shell # Install the EPEL repository: `sudo dnf -y install epel-release` # Enable additional repositories to resolve dependencies: `sudo dnf config-manager --set-enabled PowerTools` # Disable the built-in PostgreSQL module: `sudo dnf -qy module disable postgresql`

Install the package

```
# To install PostGIS 3.2:

sudo dnf -y install edb-as<xx>-postgis32

# To install PostGIS 3.1 using EDB Postgres Advanced Server 13-14:

sudo dnf -y install edb-as<xx>-postgis3

# To install PostGIS 3.1 using EDB Postgres Advanced Server 11:

sudo dnf -y install edb-as11-postgis
```

Where <xx> is the version of EDB Postgres Advanced Server. Replace <xx> with the version of EDB Postgres Advanced Server you are using. For example, edb-as14-postgis32 .

Installing PostGIS on CentOS 7 x86_64

Prerequisites

Before you begin the installation process:

- **Install EDB Postgres Advanced Server on the same host. See** Installing EDB Postgres Advanced Server .
- Set up the repository

Setting up the repository is a one-time task. If you have already set up your repository, you do not need to perform this step.

To set up the repository, go to [EDB repositories](#) and follow the instructions provided there.

- Address other prerequisites

```
# Install the EPEL repository:  
sudo yum -y install https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.  
↪rpm
```

Install the package

```
# To install PostGIS 3.2:  
  
sudo yum -y install edb-as<xx>-postgis32  
  
# To install PostGIS 3.1 using EDB Postgres Advanced Server 13-14:  
  
sudo yum -y install edb-as<xx>-postgis3  
  
# To install PostGIS 3.1 using EDB Postgres Advanced Server 10-12:  
  
sudo yum -y install edb-as<xx>-postgis
```

Where <xx> is the version of EDB Postgres Advanced Server. Replace <xx> with the version of EDB Postgres Advanced Server you are using. For example, `edb-as14-postgis32` .

Installing PostGIS on RHEL 7 or OL 7 x86_64

Prerequisites

Before you begin the installation process:

- **Install EDB Postgres Advanced Server on the same host. See** Installing EDB Postgres Advanced Server .
- Set up the repository

Setting up the repository is a one-time task. If you have already set up your repository, you do not need to perform this step.

To set up the repository, go to [EDB repositories](#) and follow the instructions provided there.

- Address other prerequisites shell # Install the EPEL repository: `sudo yum -y install https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm` # Enable additional repositories to resolve dependencies: `subscription-manager repos --enable "rhel-*-optional-rpms" --enable "rhel-*-extras-rpms" --enable "rhel-ha-for-rhel-*-server-rpms"`

Install the package

```
# To install PostGIS 3.2:

sudo yum -y install edb-as<xx>-postgis32

# To install PostGIS 3.1 using EDB Postgres Advanced Server 13-14:

sudo yum -y install edb-as<xx>-postgis3

# To install PostGIS 3.1 using EDB Postgres Advanced Server 10-11:

sudo yum -y install edb-as<xx>-postgis
```

Where <xx> is the version of EDB Postgres Advanced Server. Replace <xx> with the version of EDB Postgres Advanced Server you are using. For example, `edb-as14-postgis32`.

Installing PostGIS on SLES 15 x86_64

Prerequisites

Before you begin the installation process:

- **Install EDB Postgres Advanced Server on the same host. See** [Installing EDB Postgres Advanced Server](#).
- Set up the repository

Setting up the repository is a one-time task. If you have already set up your repository, you do not need to perform this step.

To set up the repository, go to [EDB repositories](#) and follow the instructions provided there.

- Address other prerequisites

```
# Activate the required SUSE module
sudo SUSEConnect -p PackageHub/15.3/x86_64

# Refresh the metadata
sudo zypper refresh
```

Install the package

```
# To install PostGIS 3.2:

sudo zypper -n install edb-as<xx>-postgis32

# To install PostGIS 3.1:

sudo zypper -n install edb-as<xx>-postgis3
```

Where <xx> is the version of EDB Postgres Advanced Server. Replace <xx> with the version of EDB Postgres Advanced Server you are using. For example, `edb-as14-postgis32` .

Installing PostGIS on SLES 12 x86_64

Prerequisites

Before you begin the installation process:

- **Install EDB Postgres Advanced Server on the same host.** See [Installing EDB Postgres Advanced Server](#) .
- Set up the repository

Setting up the repository is a one-time task. If you have already set up your repository, you do not need to perform this step.

To set up the repository, go to [EDB repositories](#) and follow the instructions provided there.

- Address other prerequisites

```
# Activate the required SUSE module

sudo SUSEConnect -p PackageHub/12.5/x86_64
sudo SUSEConnect -p sle-sdk/12.5/x86_64

# Refresh the metadata

sudo zypper refresh
```

Install the package

```
# To install PostGIS 3.2:

sudo zypper -n install edb-as<xx>-postgis32

# To install PostGIS 3.1 using EDB Postgres Advanced Server 13-14:

sudo zypper -n install edb-as<xx>-postgis3

# To install PostGIS 3.1 using EDB Postgres Advanced Server 12:

sudo zypper -n install edb-as12-postgis
```

Where <xx> is the version of EDB Postgres Advanced Server. Replace <xx> with the version of EDB Postgres Advanced Server you are using. For example, `edb-as14-postgis32` .

Installing PostGIS on Ubuntu 20.04 x86_64

Prerequisites

Before you begin the installation process:

- **Install EDB Postgres Advanced Server on the same host. See**
Installing EDB Postgres Advanced Server .
- Set up the repository

Setting up the repository is a one-time task. If you have already set up your repository, you do not need to perform this step.

To set up the repository, go to [EDB repositories](#) and follow the instructions provided there.

Install the package

```
# To install PostGIS 3.2:

sudo apt-get -y install edb-as14-postgis32

# To install PostGIS 3.1:

sudo apt-get -y install edb-as<xx>-postgis3
```

Where <xx> is the version of EDB Postgres Advanced Server. Replace <xx> with the version of EDB Postgres Advanced Server you are using. For example, `edb-as14-postgis32` .

Installing PostGIS on Ubuntu 18.04 x86_64

Prerequisites

Before you begin the installation process:

- **Install EDB Postgres Advanced Server on the same host. See**
Installing EDB Postgres Advanced Server .
- Set up the repository

Setting up the repository is a one-time task. If you have already set up your repository, you do not need to perform this step.

To set up the repository, go to [EDB repositories](#) and follow the instructions provided there.

Install the package

```
# To install PostGIS 3.2 using EDB Postgres Advanced Server 13-14:
sudo apt-get -y install edb-as<xx>-postgis32

# To install PostGIS 3.2 using EDB Postgres Advanced Server 11-12:
sudo apt-get -y install edb-as<xx>-postgis-3.2

# To install PostGIS 3.1 using EDB Postgres Advanced Server 13-14:
sudo apt-get -y install edb-as<xx>-postgis3

# To install PostGIS 3.1 using EDB Postgres Advanced Server 11-12:
sudo apt-get -y install edb-as<xx>-postgis-3.1
```

Where <xx> is the version of EDB Postgres Advanced Server. Replace <xx> with the version of EDB Postgres Advanced Server you are using. For example, `edb-as14-postgis32` .

Installing PostGIS on Debian 10 x86_64

Prerequisites

Before you begin the installation process:

- **Install EDB Postgres Advanced Server on the same host. See**
Installing EDB Postgres Advanced Server .
- Set up the repository

Setting up the repository is a one-time task. If you have already set up your repository, you do not need to perform this step.

To set up the repository, go to [EDB repositories](#) and follow the instructions provided there.

Install the package

```
# To install PostGIS 3.2 using EDB Postgres Advanced Server 13-14:
sudo apt-get -y install edb-as<xx>-postgis32

# To install PostGIS 3.2 using EDB Postgres Advanced Server 11-12:
sudo sudo apt-get -y install edb-as<xx>-postgis-3.2

# To install PostGIS 3.1 using EDB Postgres Advanced Server 13-14:
sudo apt-get -y install edb-as<xx>-postgis3

# To install PostGIS 3.1 using EDB Postgres Advanced Server 12:
```

(continues on next page)

(continued from previous page)

```
sudo apt-get -y install edb-as12-postgis-3.1
```

Where <xx> is the version of EDB Postgres Advanced Server. Replace <xx> with the version of EDB Postgres Advanced Server you are using. For example, `edb-as14-postgis32`.

3.1.2 Installing PostGIS on IBM Power (ppc64le)

For operating system-specific install instructions, see:

- *Installing PostGIS on RHEL 8 ppc64le*
- *Installing PostGIS on SLES 15 ppc64le*
- *Installing PostGIS on SLES 12 ppc64le*

Installing PostGIS on RHEL 8 ppc64le

Prerequisites

Before you begin the installation process:

- **Install EDB Postgres Advanced Server on the same host.** See [Installing EDB Postgres Advanced Server](#).
- Set up the repository

Setting up the repository is a one-time task. If you have already set up your repository, you do not need to perform this step.

To set up the repository, go to [EDB repositories](#) and follow the instructions provided there.

- Address other prerequisites

```
# Install the EPEL repository:
sudo dnf -y install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.
↪rpm

# Refresh the cache:
sudo dnf makecache

# Enable additional repositories to resolve dependencies:
ARCH=$( /bin/arch ) subscription-manager repos --enable "codeready-builder-for-rhel-8-$
↪{ARCH}-rpms"

# Disable the built-in PostgreSQL module:
sudo dnf -qy module disable postgresql
```

Install the package

```
# To install PostGIS 3.2:

sudo dnf -y install edb-as<xx>-postgis32

# To install PostGIS 3.1 using EDB Postgres Advanced Server 13-14:

sudo dnf -y install edb-as<xx>-postgis3

# To install PostGIS 3.1 using EDB Postgres Advanced Server 11-12:

sudo dnf -y install edb-as<xx>-postgis
```

Where <xx> is the version of EDB Postgres Advanced Server. Replace <xx> with the version of EDB Postgres Advanced Server you are using. For example, `edb-as14-postgis32` .

Installing PostGIS on SLES 15 ppc64le

Prerequisites

Before you begin the installation process:

- **Install EDB Postgres Advanced Server on the same host. See** [Installing EDB Postgres Advanced Server](#) .
- Set up the repository

Setting up the repository is a one-time task. If you have already set up your repository, you do not need to perform this step.

To set up the repository, go to [EDB repositories](#) and follow the instructions provided there.

- Address other prerequisites

```
# Activate the required SUSE module
sudo SUSEConnect -p PackageHub/15.3/ppc64le

# Refresh the metadata
sudo zypper refresh
```

Install the package

```
# To install PostGIS 3.2:

sudo zypper -n install edb-as<xx>-postgis32

# To install PostGIS 3.1:

sudo zypper -n install edb-as<xx>-postgis3
```

Where <xx> is the version of EDB Postgres Advanced Server. Replace <xx> with the version of EDB Postgres Advanced Server you are using. For example, `edb-as14-postgis32` .

Installing PostGIS on SLES 12 ppc64le

Prerequisites

Before you begin the installation process:

- **Install EDB Postgres Advanced Server on the same host. See** [Installing EDB Postgres Advanced Server](#) .
- Set up the repository

Setting up the repository is a one-time task. If you have already set up your repository, you do not need to perform this step.

To set up the repository, go to [EDB repositories](#) and follow the instructions provided there.

- Address other prerequisites

```
# Activate the required SUSE module
sudo SUSEConnect -p PackageHub/12.5/ppc64le
sudo SUSEConnect -p sle-sdk/12.5/ppc64le

# Refresh the metadata
sudo zypper refresh
```

Install the package

```
# To install PostGIS 3.2:

zypper -n install edb-as<xx>-postgis32

# To install PostGIS 3.1 using EDB Postgres Advanced Server 13-14:

zypper -n install edb-as<xx>-postgis3

# To install PostGIS 3.1 using EDB Postgres Advanced Server 12:

zypper -n install edb-as12-postgis
```

Where <xx> is the version of EDB Postgres Advanced Server. Replace <xx> with the version of EDB Postgres Advanced Server you are using. For example, `edb-as14-postgis32` .

3.2 Installing PostGIS on Windows

You must install Advanced Server before installing PostGIS. If you have used the graphical Setup wizard to install EDB Postgres Advanced Server, you can use StackBuilder Plus to add PostGIS to your installation. For details about using the graphical installer to install and configure Advanced Server, see the [EDB Postgres Advanced Server Installation Guide for Windows](#).

Note: To install PostGIS version 3.1 on EDB Postgres Advanced Server version 10 on Windows, you need to upgrade it to the latest EDB Postgres Advanced Server minor version of 10.16.25 (or later), and then proceed with PostGIS 3.1

installation.

1. Open StackBuilder Plus and select your Advanced Server installation from the drop-down list on the Welcome window. Click Next to continue to the application selection page.

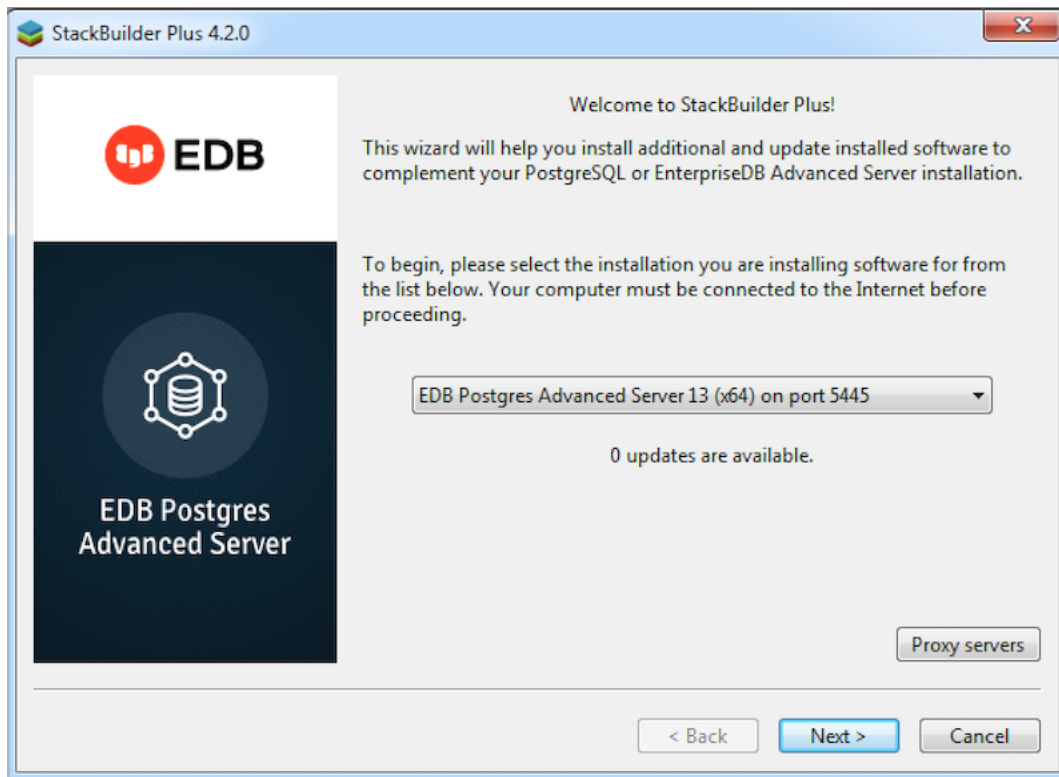


Fig. 1: The

StackBuilder Plus Welcome window

2. Expand the Spatial Extensions node, and check the box next to the PostGIS version. Click Next to continue.
3. The selected packages and the default download directory are displayed; change the locations if required. Click Next .

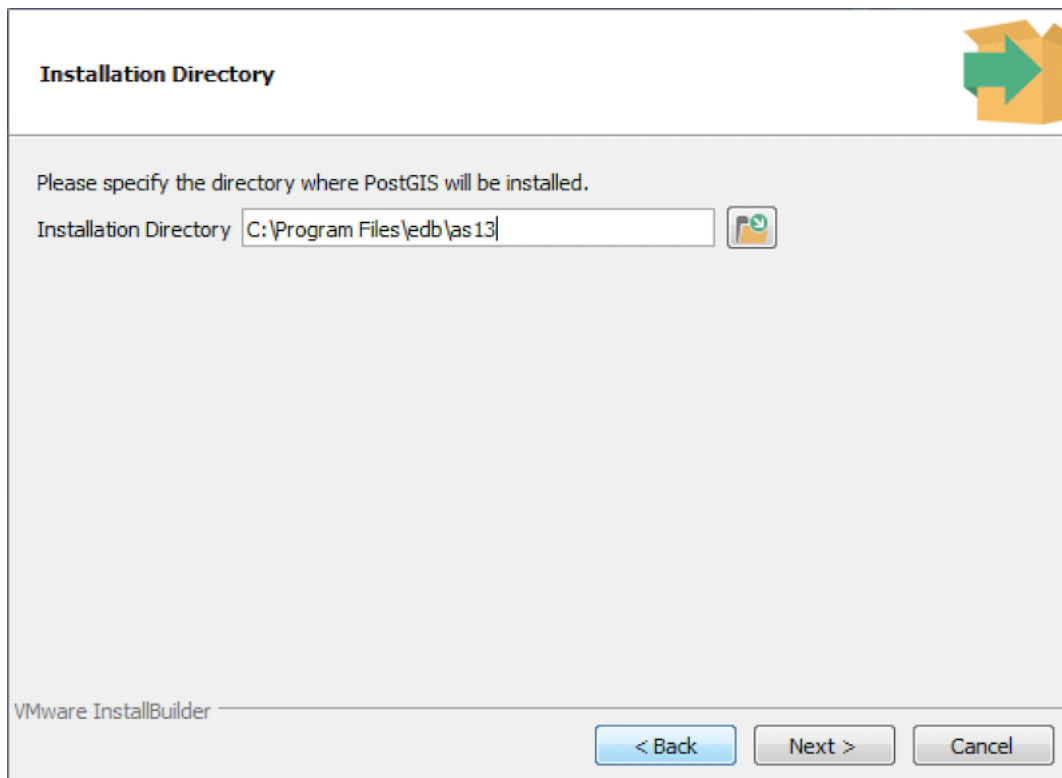


Fig. 2: The In-

stallation Directory window

4. Once you have downloaded the installation files, a confirmation message is displayed. Click **Next** to start the PostGIS installation.

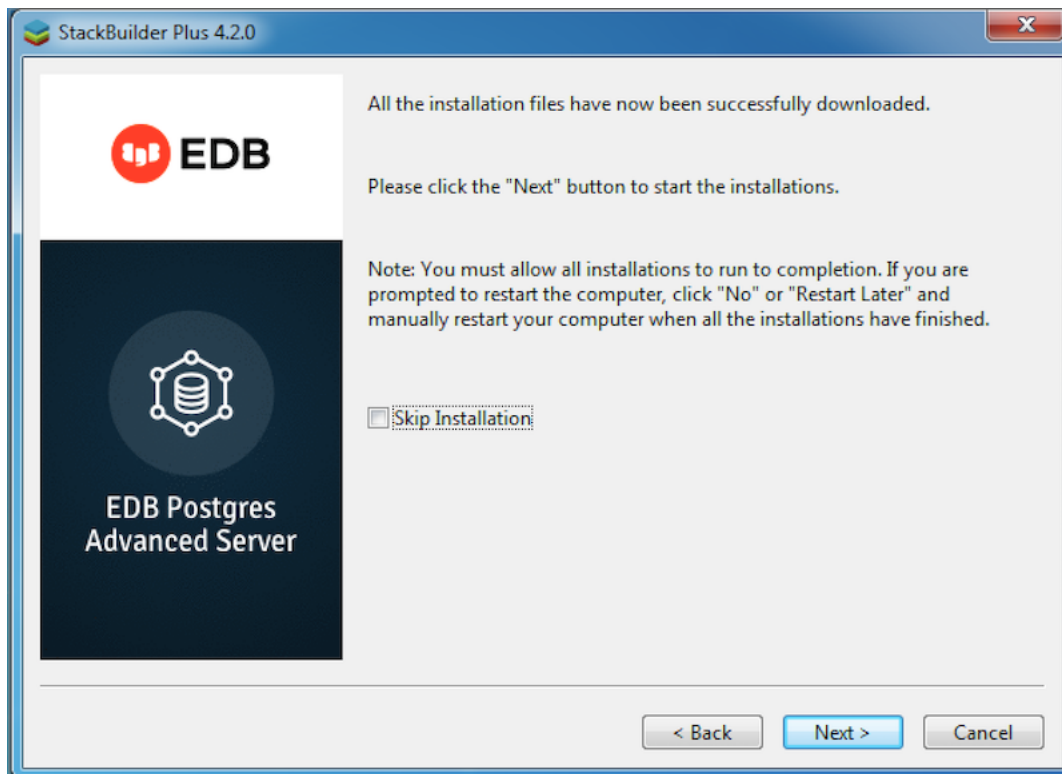


Fig. 3: In-

stalling Postgis

5. Select an installation language and click OK .
6. The PostGIS welcome screen is displayed. Click Next .

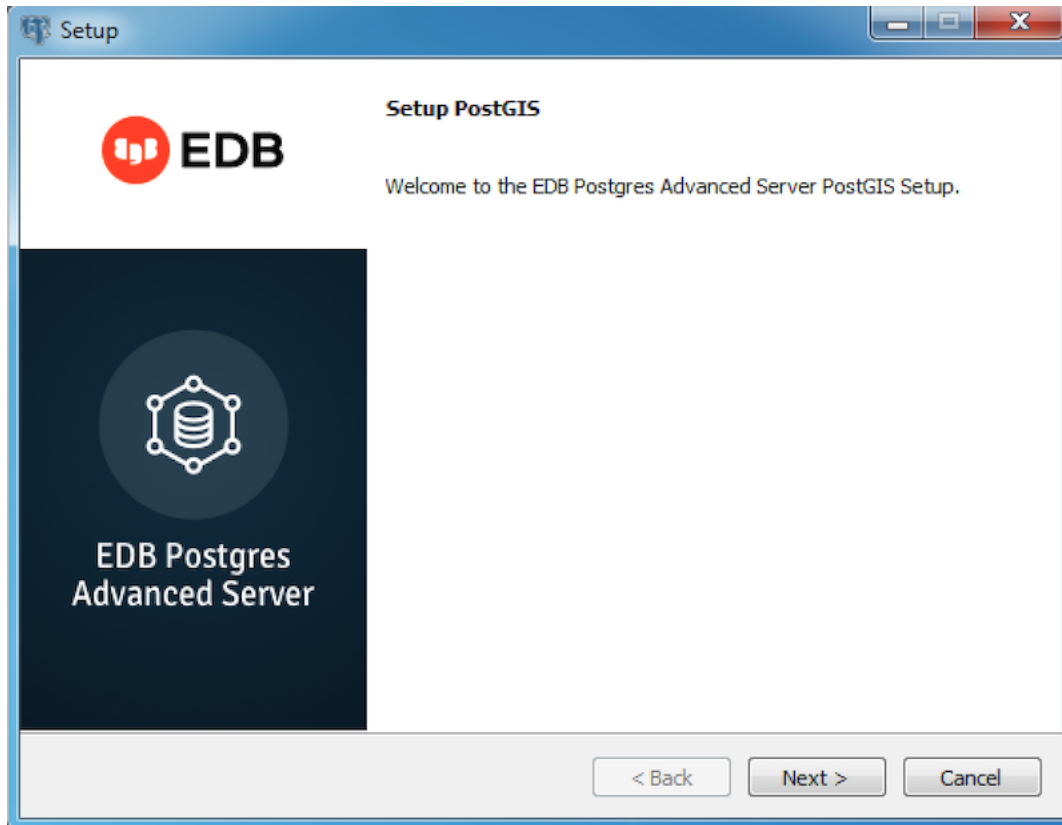
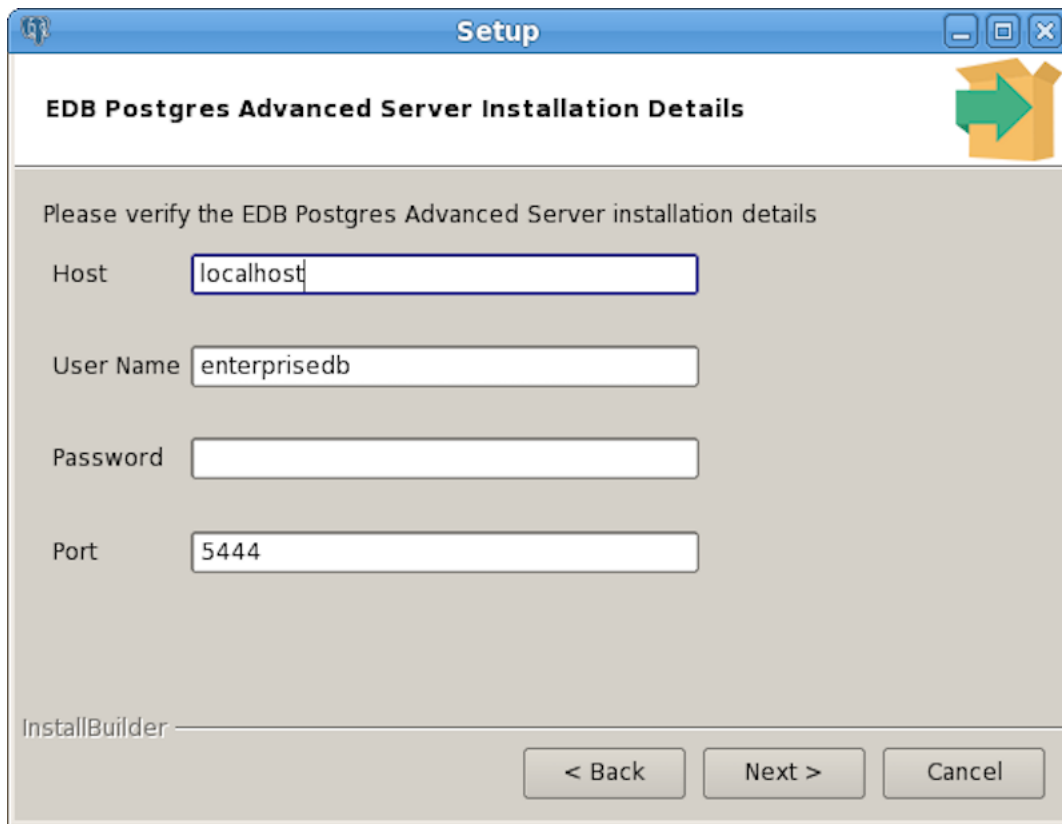


Fig. 4: The

Postgis welcome window

7. Use the **Installation Directory** field to specify the directory you wish to install the PostGIS software. Click **Next** to continue.
8. Use fields on the **EDB Postgres Advanced Server Installation Details** window to provide connection information for the Advanced Server host:
 - Use the **Host** field to identify the system on which Advanced Server resides.
 - Provide the name of the role that PostGIS will use for connections to the server in the **User Name** field.
 - Provide the password associated with the role in the **Password** field.
 - Use the **Port** field to identify the listener port that Advanced Server monitors for client connections.

Click **Next** to continue.



The screenshot shows a Windows-style window titled "Setup". The main heading is "EDB Postgres Advanced Server Installation Details". Below this, a message says "Please verify the EDB Postgres Advanced Server installation details". There are four input fields: "Host" with "localhost", "User Name" with "enterprisedb", "Password" (empty), and "Port" with "5444". At the bottom left is the text "InstallBuilder". At the bottom right are three buttons: "< Back", "Next >", and "Cancel". A green arrow icon is in the top right corner of the window.

Fig. 5: The Ad-

vanced Server installation details window

9. The Ready to Install window notifies you when the installer has all of the information needed to install PostGIS on your system. Click Next .

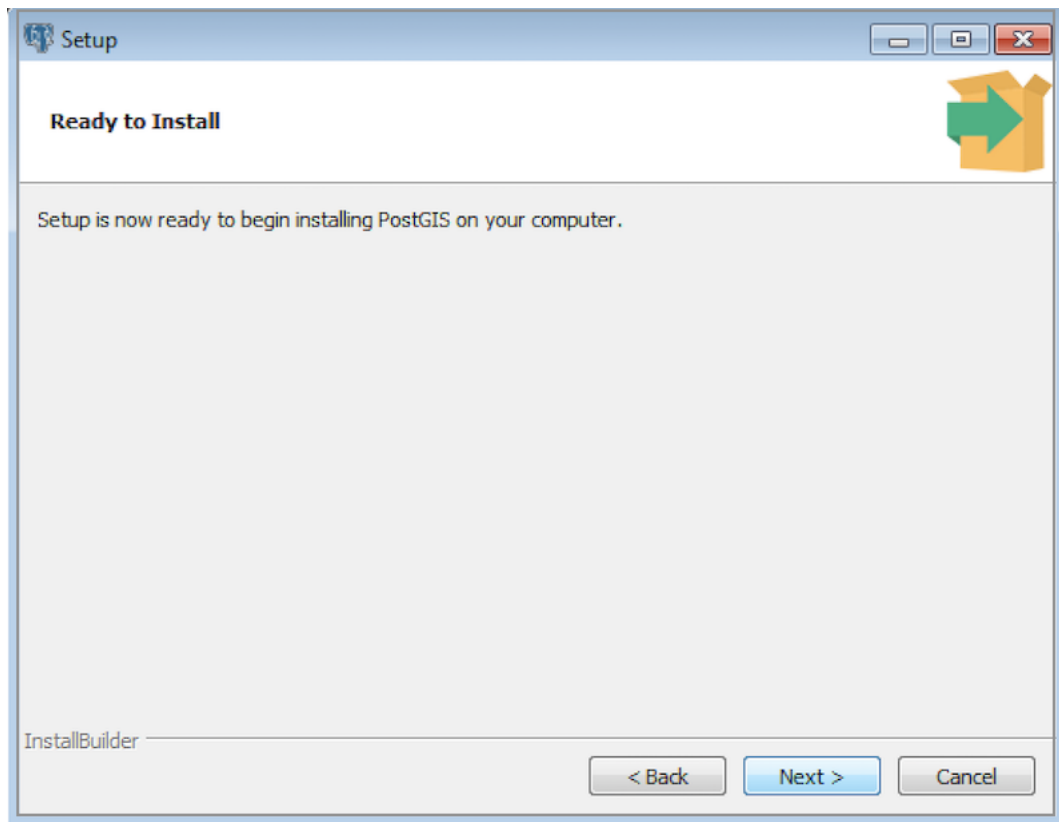


Fig. 6: The

ready to install window

10. Progress bars inform you as the installation progresses. Click **Finish** to exit the installer when the PostGIS installation completes.

Fig. 7: The installation is complete

StackBuilder Plus will install PostGIS and create the `template_postgis` database and PostGIS functions.

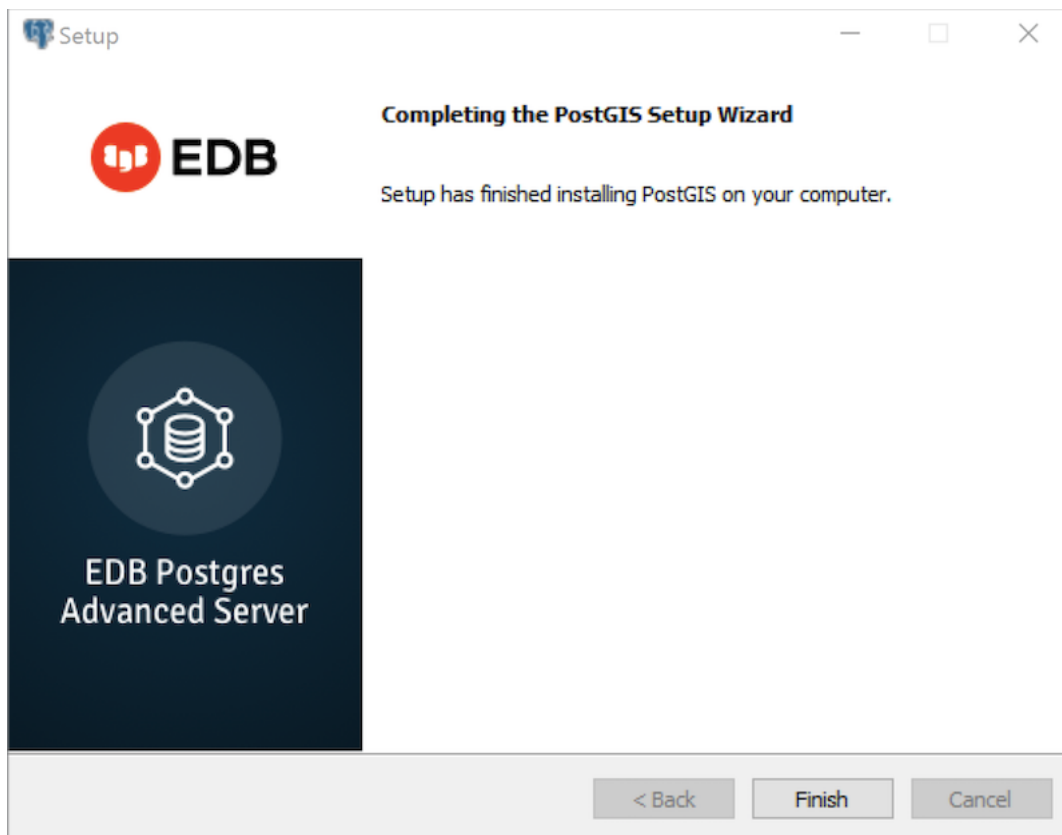


Fig. 1: The installation is complete

CREATING EXTENSIONS

After installing PostGIS, create a PostGIS database and the extensions in each database where you want to use PostGIS extensions. Don't create the extensions in the postgres or edb database.

1. Before creating the postgis database, we recommend creating a superuser to administer the database. To create the user, navigate to the bin directory under your EDB Postgres Advanced Server installation and connect to the server with the psql client:

```
./psql -d edb -U enterprisedb -h 127.0.0.1
```

2. Invoke the following command to create a privileged role:

```
CREATE ROLE gisadmin LOGIN PASSWORD password SUPERUSER;
```

3. Log out of psql and connect as gisadmin:

```
edb=# \q  
./psql -d edb -U gisadmin -h 127.0.0.1
```

4. Invoke the following command to create the postgis database owned by gisadmin:

```
CREATE DATABASE postgis;
```

5. Use the \c command to switch to the postgis database, and use the CREATE EXTENSION command to create the PostGIS extensions:

```
\c postgis  
CREATE EXTENSION postgis;  
CREATE EXTENSION postgis_topology;  
CREATE EXTENSION fuzzystrmatch;  
CREATE EXTENSION address_standardizer;  
CREATE EXTENSION address_standardizer_data_us;  
CREATE EXTENSION postgis_tiger_geocoder;  
CREATE EXTENSION postgis_sfcgal;  
CREATE EXTENSION postgis_raster;
```

Note:

- The postgis-sfcgal extension isn't available on Ubuntu 18, Ubuntu 20, SLES 12, RHEL/CentOS 7 - ppc64le, and Windows platforms.
-

- The `postgis_raster` extension isn't supported on SLES 15 x86 and SLES 15 PPCLE. On SLES 15 x86 and SLES 15 PPCLE, `libOpenCL1` is required by GDAL and comes from the SUSE Linux Enterprise Workstation Extension 15 SP4 x86_64 repository.

When connected with pgAdmin, you can see PostGIS extensions, functions, tables, and trigger functions beneath the `postgis` database public schema. The `postgis` database is now geospatially enabled. You can use it as a template to create geospatial databases.

UPGRADING POSTGIS

This section walks you through the following upgrade examples for RHEL/CentOS 7 platforms:

- *Example 1: Upgrading from PostGIS version 3.1 to 3.2*
- *Example 2: Upgrading from PostGIS versions 2.5 or later to 3.1.4 on the same EDB Postgres Advanced Server version*
- *Example 3: Upgrading from PostGIS versions earlier than 2.5 to 3.1.4 on a different EDB Postgres Advanced Server version*

5.1 Example 1: Upgrading from PostGIS version 3.1 to 3.2

The following example walks you through upgrading from PostGIS version 3.1 to version 3.2 for EDB Postgres Advanced Server 14.

To avoid accidental upgrades, PostGIS version 3.1 and 3.2 use different package names. So, upgrading to version 3.2 requires that you first uninstall version 3.1. Once you install version 3.2, you can upgrade the PostGIS extensions.

1. Create the PostGIS extensions in PostGIS version 3.1, if you haven't already. For instructions, see [Creating extensions](#).
2. Uninstall PostGIS version 3.1. For EPAS version 14:

```
yum -y erase edb-as14-postgis3
```

For older versions of EDB Postgres Advanced Server:

```
yum erase edb-as<xx>-postgis3
``` Where `<xx>` is the version of EDB Postgres Advanced Server.
```

For more information, see `:ref:`Uninstalling PostGIS<Uninstalling PostGIS>``.

1. Install PostGIS version 3.2. For instructions, see `:ref:`Installing PostGIS on RHEL_7/OL 7 x86 </postgis/latest/01a_installing_postgis/installing_on_linux/x86_amd64/postgis_rhel7_x86/>``.

1. Upgrade the PostGIS extensions using the following command:

.. Note::

If you first upgraded from a PostGIS version earlier than 2.5 to PostGIS version 3.1, `then` you must `DROP` the `postgis_raster` extension before executing the command. !!

(continues on next page)

(continued from previous page)

```

`sql
SELECT postgis_extensions_upgrade();
__OUTPUT__
NOTICE: Updating extension postgis from 3.1.4 to 3.2.1
NOTICE: Updating extension postgis_sfcgal from 3.1.4 to 3.2.1
NOTICE: Updating extension postgis_raster from 3.1.4 to 3.2.1
NOTICE: Updating extension postgis_topology from 3.1.4 to 3.2.1
NOTICE: Updating extension postgis_tiger_geocoder from 3.1.4 to 3.2.1
 postgis_extensions_upgrade

Upgrade completed, run SELECT postgis_full_version(); for details
(1 row)

```

## 5.2 Example 2: Upgrading from PostGIS versions 2.5 or later to 3.1.4 on the same EDB Postgres Advanced Server version

The following example walks you through the process of upgrading from PostGIS version 2.5.5 to PostGIS version 3.1.4 for EDB Postgres Advanced Server version 12.

**Note:** It is assumed that you have already created extensions for PostGIS version 2.5.5 as described in [Creating extensions](#) and EDB Advanced Server 12 service is running.

1. To upgrade PostGIS version 2.5.5 to PostGIS version 3.1.4 for EDB Postgres Advanced Server 12, assume the superuser privileges and invoke the following command:

```
yum upgrade edb-as12-postgis-3.1.4 -y
```

2. To update extensions, switch to enterprisedb user, connect to the database (where you have already created extensions) with the psql client application, and execute the following commands:

```

edb=# alter extension postgis update to 3.1.4;
WARNING: unpackaging raster
WARNING: PostGIS Raster functionality has been unpackaged
HINT: type `SELECT postgis_extensions_upgrade();` to finish the upgrade. After upgrading,
 ↪ if you want to drop raster, run: DROP EXTENSION postgis_raster;
ALTER EXTENSION

```

```

edb=# alter extension address_standardizer update to 3.1.4;
ALTER EXTENSION

```

```

edb=# SELECT postgis_extensions_upgrade();
NOTICE: Updating extension postgis_sfcgal from 2.5.5 to 3.1.4
NOTICE: Packaging extension postgis_raster
NOTICE: Updating extension postgis_topology from 2.5.5 to 3.1.4
NOTICE: Updating extension postgis_tiger_geocoder from 2.5.5 to 3.1.4
 postgis_extensions_upgrade

```

(continues on next page)

(continued from previous page)

```
Upgrade completed, run SELECT postgis_full_version(); for details
(1 row)
```

### 5.3 Example 3: Upgrading from PostGIS versions earlier than 2.5 to 3.1.4 on a different EDB Postgres Advanced Server version

The following example walks you through the process of upgrading PostGIS version 2.4.6 for EDB Postgres Advanced Server 10 to PostGIS version 3.1.4 for EDB Postgres Advanced Server version 14:

- Step 1 — Upgrade to PostGIS 3.1.1. This is an intermediate step required to resolve dependency issues.
- Step 2 — Upgrade to PostGIS 3.1.4.
- Step 3 — Upgrade EDB Postgres Advanced Server version 10 to 14.

When the PostGIS data has a dependency on the raster functions, upgrading to PostGIS version 3.1.4 requires **dumping and reloading the data**.

Step 1 — To upgrade PostGIS version 2.4.6 to 3.1.4, you need to upgrade it to 3.1.1 first:

1. Navigate to the bin directory of EDB Advanced Server 10:

```
cd /usr/edb/as10/bin/
```

2. Assume the superuser privileges and invoke the following command to upgrade to PostGIS version 3.1.1:

```
yum upgrade edb-as10-postgis-3.1.1 -y
```

**Note:** See the [Installing PostGIS on a Debian/Ubuntu Host section](#) for information about Debian platform commands.

3. To update extensions, switch to `enterprisedb` user, connect to the database (where you have already created extensions) with the `psql` client application, and execute the following commands:

```
edb=# alter extension postgis update to 3.1.1;
WARNING: unpackaging raster
WARNING: PostGIS Raster functionality has been unpackaged
HINT: type `SELECT postgis_extensions_upgrade();` to finish the upgrade. After upgrading,
→ if you want to drop raster, run: DROP EXTENSION postgis_raster;
ALTER EXTENSION
```

```
edb=# alter extension address_standardizer update to 3.1.1;
ALTER EXTENSION
```

```
edb=# SELECT postgis_extensions_upgrade();
__OUTPUT__
NOTICE: Updating extension postgis_sfcgal from 2.4.6 to 3.1.1
NOTICE: Packaging extension postgis_raster
NOTICE: Updating extension postgis_topology from 2.4.6 to 3.1.1
NOTICE: Updating extension postgis_tiger_geocoder from 2.4.6 to 3.1.1
 postgis_extensions_upgrade

```

(continues on next page)

(continued from previous page)

```
Upgrade completed, run SELECT postgis_full_version(); for details
(1 row)
```

Step 2 — Upgrade to PostGIS version 3.1.4:

1. To upgrade PostGIS version 3.1.1 to 3.1.4, invoke the following command for EDB Postgres Advanced Server 10:

```
yum upgrade edb-as10-postgis-3.1.4 -y
```

2. To update extensions, switch to `enterprisedb` user and execute the following commands:

```
edb=# alter extension postgis update to 3.1.4;
WARNING: unpackaging raster
WARNING: PostGIS Raster functionality has been unpackaged
HINT: type \SELECT postgis_extensions_upgrade();` to finish the upgrade. After upgrading,
→ if you want to drop raster, run: DROP EXTENSION postgis_raster;
ALTER EXTENSION
```

With PostGIS version 3.1.4, the return type of the raster functions has changed, which requires dropping and creating the raster extension as part of the upgrade process.

```
edb=# drop extension postgis_raster;
DROP EXTENSION
```

```
edb=# SELECT postgis_extensions_upgrade();
NOTICE: Updating extension postgis_sfcgal from 3.1.1 to 3.1.4
NOTICE: Updating extension postgis_topology from 3.1.1 to 3.1.4
NOTICE: Updating extension postgis_tiger_geocoder from 3.1.1 to 3.1.4
 postgis_extensions_upgrade

Upgrade completed, run SELECT postgis_full_version(); for details
(1 row)
```

```
edb=# alter extension address_standardizer update to 3.1.4;
ALTER EXTENSION
```

Quit and re-open the `psql` client session:

```
edb=# \q
./psql -d edb -p 5444
```

```
edb=# create extension postgis_raster;
CREATE EXTENSION
```

Step 3 — Upgrade EDB Postgres Advanced Server version 10 to 14:

1. Assume the superuser privileges to stop the EDB Postgres Advanced Server 10 service:

```
systemctl stop edb-as-10
```

2. Install the EDB Postgres Advanced Server version 14:

```
yum install edb-as14-server -y
```

3. Navigate to the bin directory of EDB Advanced Server 14 and initialize the cluster:

```
cd /usr/edb/as14/bin/
./edb-as-14-setup initdb
```

4. Install the PostGIS version 3.1.4 for EDB Postgres Advanced Server version 14.0:

```
yum install edb-as14-postgis3-3.1.4 -y
```

5. Assume the superuser privileges to stop the EDB Postgres Advanced Server 14.0 service:

```
systemctl stop edb-as-14
```

6. Switch to enterprisedb user and create a temp folder:

```
su enterprisedb
cd $(mktemp -d)
```

7. Execute the following commands to check cluster compatibility and consistency, as well as to perform the upgrade.

```
bash-4.2$ /usr/edb/as14/bin/pg_upgrade -d /var/lib/edb/as10/data/ -D /var/lib/edb/as14/
↳data/ -U enterprisedb -b /usr/edb/as10/bin/ -B /usr/edb/as14/bin/ -p 5444 -P 5445 -c
```

Performing Consistency Checks

```

Checking cluster versions ok
Checking database user is the install user ok
Checking database connection settings ok
Checking for prepared transactions ok
Checking for reg* data types in user tables ok
Checking for contrib/isn with bigint-passing mismatch ok
Checking for tables WITH OIDS ok
Checking for invalid "sql_identifier" user columns ok
Checking for invalid "unknown" user columns ok
Checking for hash indexes ok
Checking for presence of required libraries ok
Checking database user is the install user ok
Checking for prepared transactions ok
Checking for new cluster tablespace directories ok
```

Clusters are compatible

```
bash-4.2$ /usr/edb/as14/bin/pg_upgrade -d /var/lib/edb/as10/data/ -D /var/lib/edb/as14/
↳data/ -U enterprisedb -b /usr/edb/as10/bin/ -B /usr/edb/as14/bin/ -p 5444 -P 5445 --
↳link
```

Performing Consistency Checks

```

Checking cluster versions ok
```

(continues on next page)

(continued from previous page)

```
Checking database user is the install user ok
Checking database connection settings ok
Checking for prepared transactions ok
Checking for reg* data types in user tables ok
Checking for contrib/isn with bigint-passing mismatch ok
Checking for tables WITH OIDS ok
Checking for invalid "sql_identifier" user columns ok
Checking for invalid "unknown" user columns ok
Creating dump of global objects ok
Creating dump of database schemas
ok
Checking for presence of required libraries ok
Checking database user is the install user ok
Checking for prepared transactions ok
Checking for new cluster tablespace directories 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 in the new cluster ok
Deleting files from new pg_xact ok
Copying old pg_clog to new server ok
Setting next transaction ID and epoch for new cluster ok
Deleting files from new pg_multixact/offsets ok
Copying old pg_multixact/offsets to new server ok
Deleting files from new pg_multixact/members ok
Copying old pg_multixact/members to new server ok
Setting next multixact ID and offset for new cluster ok
Resetting WAL archives ok
Setting frozenxid and minmxid counters in new cluster ok
Restoring global objects in the new cluster ok
Restoring database schemas in the new cluster
ok
Adding ".old" suffix to old global/pg_control ok
```

If you want to start the old cluster, you will need to remove the ".old" suffix from /var/lib/edb/as9.6/data/global/pg\_control.old. Because "link" mode was used, the old cluster cannot be safely started once the new cluster has been started.

#### Linking user relation files

```
ok
Setting next OID for new cluster ok
Sync data directory to disk ok
Creating script to analyze new cluster ok
Creating script to delete old cluster ok
Checking for hash indexes ok
```

#### Upgrade Complete

Optimizer statistics are not transferred by pg\_upgrade so,

(continues on next page)

(continued from previous page)

```
once you start the new server, consider running:
./analyze_new_cluster.sh
```

```
Running this script will delete the old clusters data files:
./delete_old_cluster.sh
```

8. Assume the superuser privileges, navigate to the bin directory of EDB Postgres Advanced Server 14, and start the service:

```
cd /usr/edb/as14/bin/

systemctl start edb-as-14
```

9. To update extensions, switch to enterprisedb user, connect to the database (where you have already created extensions) with the psql client application, and execute the following commands:

```
su enterprisedb
./psql -d edb -p 5444
edb=# SELECT PostGIS_Extensions_Upgrade();
__OUTPUT__
NOTICE: Updating extension postgis 3.1.4
 postgis_extensions_upgrade

Upgrade completed, run SELECT postgis_full_version(); for details
(1 row)
```



## USING POSTGIS

The following examples use PostGIS functions to create and query spatial objects. For more information about the PostGIS functions, see the

[official PostGIS documentation](#) .

The following command creates a table named `roads` that holds GIS data and a geometry column.

```
CREATE TABLE roads (ID int4, NAME varchar(128));
```

Use the PostGIS `AddGeometryColumn` function to add a column to the table:

```
SELECT AddGeometryColumn(roads, geom, -1, GEOMETRY, 2);
```

Use the following SQL commands to insert data into the table `roads` . This data consists of the geometry of the type of `Linestring` (a line between two points):

```
INSERT INTO ROADS (ID,GEOM,NAME) VALUES (1,ST_GeomFromText(LINESTRING(0 10,0 0),-1),
↪Beacon Road);
INSERT INTO ROADS (ID,GEOM,NAME) VALUES (2,ST_GeomFromText(LINESTRING(0 0,0 10),-1),
↪Violet Road);
INSERT INTO ROADS (ID,GEOM,NAME) VALUES (3,ST_GeomFromText(LINESTRING(0 0,10 0),-1),
↪Skelton Street);
INSERT INTO ROADS (ID,GEOM,NAME) VALUES (4,ST_GeomFromText(LINESTRING(0 0,10 10),-1),
↪Fifth Avenue);
INSERT INTO ROADS (ID,GEOM,NAME) VALUES (5,ST_GeomFromText(LINESTRING(0 10,0 0),-1),
↪Main Street);
INSERT INTO ROADS (ID,GEOM,NAME) VALUES (6,ST_GeomFromText(LINESTRING(10 0,0 0),-1),
↪Lipton Street);
```

You can use the `GIST` function to create an index on the geometry column:

```
CREATE INDEX roads_index ON roads using GIST (geom);
```

`AsText(geometry)` is a PostGIS function that returns a text representation of the geometry:

```
SELECT id, ST_AsText(geom) AS geom, name FROM ROADS order by id;
__OUTPUT__
id | geom | name
---+-----+-----
 1 | LINESTRING(0 10,0 0) | Bacon Road
 2 | LINESTRING(0 0,0 10) | Violet Road
 3 | LINESTRING(0 0,10 0) | Skelton Street
```

(continues on next page)

(continued from previous page)

```

4 | LINESTRING(0 0,10 10) | Fifth Avenue
5 | LINESTRING(0 10,0 0) | Main Street
6 | LINESTRING(10 0,0 0) | Lipton Street
(6 rows)

```

After an index is created, you can use the && operator in a query:

```

SELECT NAME, ST_AsText(GEOM) FROM ROADS WHERE GEOM && SetSRID(BOX3D(10 10,10 10)::box3d,-
↪1);
__OUTPUT__
 name | astext
-----+-----
Fifth Avenue | LINESTRING(0 0,10 10)
(1 row)

```

Use the BOX3D function to specify a bounding box. The && operator uses the index to quickly reduce the result set down to only those geometries with bounding boxes that overlap the specified area.

You can use the ~= operator to check if two geometries are geometrically identical:

```

SELECT ID, NAME FROM roads WHERE GEOM ~= ST_GeomFromText(LINESTRING(0 10,0 0),-1) order
↪by id;
__OUTPUT__
 id | name
---+-----
 1 | Bacon Road
 5 | Main Street
(2 rows)

```

## UNINSTALLING POSTGIS

This section walks you through the process of uninstalling PostGIS.

### 7.1 Uninstalling PostGIS on a CentOS/RHEL/Rocky Linux/AlmaLinux host

To uninstall PostGIS on a CentOS/RHEL host, assume the identity of the root user and invoke the following command:  
On CentOS/RHEL 7 for EPAS version 13:

```
yum -y erase edb-as13-postgis3*
```

On CentOS/RHEL 7 for older versions of EPAS:

```
yum erase edb-as<xx>-postgis-*<y.y.y>
```

Where <xx> is the Advanced Server version and <y.y.y> is the PostGIS version you want to uninstall.

On Rocky Linux/AlmaLinux/RHEL 8 for EPAS version 13:

```
dnf -y erase edb-as13-postgis3*
```

On Rocky Linux/AlmaLinux/RHEL 8 for older versions of EPAS:

```
dnf erase edb-as<xx>-postgis-*<y.y.y>
```

### 7.2 Uninstalling PostGIS on a Debian/Ubuntu host

To uninstall PostGIS on a Debian or Ubuntu host, invoke the following command:

```
apt-get remove edb-as<xx>-postgis-<y.y>*
```

Where <xx> is the Advanced Server version and <y.y> is the PostGIS version you want to uninstall

## 7.3 Uninstalling PostGIS on a SLES host

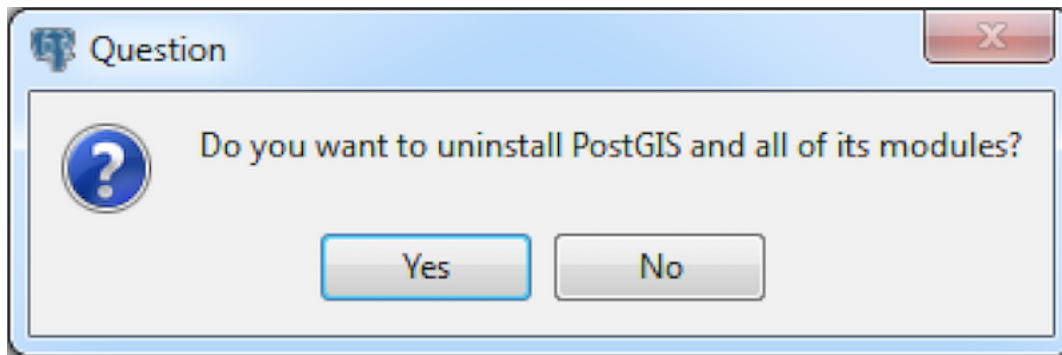
To uninstall PostGIS on a SLES host, assume the identity of the root user and invoke the following command:

```
zypper remove edb-as12-postgis*
```

## 7.4 Uninstalling PostGIS on a Windows host

The PostGIS graphical installer creates an uninstaller that you can use to remove PostGIS. The uninstaller is created in the installation directory that you have specified while installing PostGIS (default is C:\Program Files\edb\as13).

1. Navigate into the directory that contains the uninstaller and assume superuser privileges. Open the uninstaller and click Yes to begin uninstalling PostGIS:



Uninstaller Confirmation Dialog

Fig. 9: The

2. The uninstallation process begins. Click OK when the uninstallation completes:

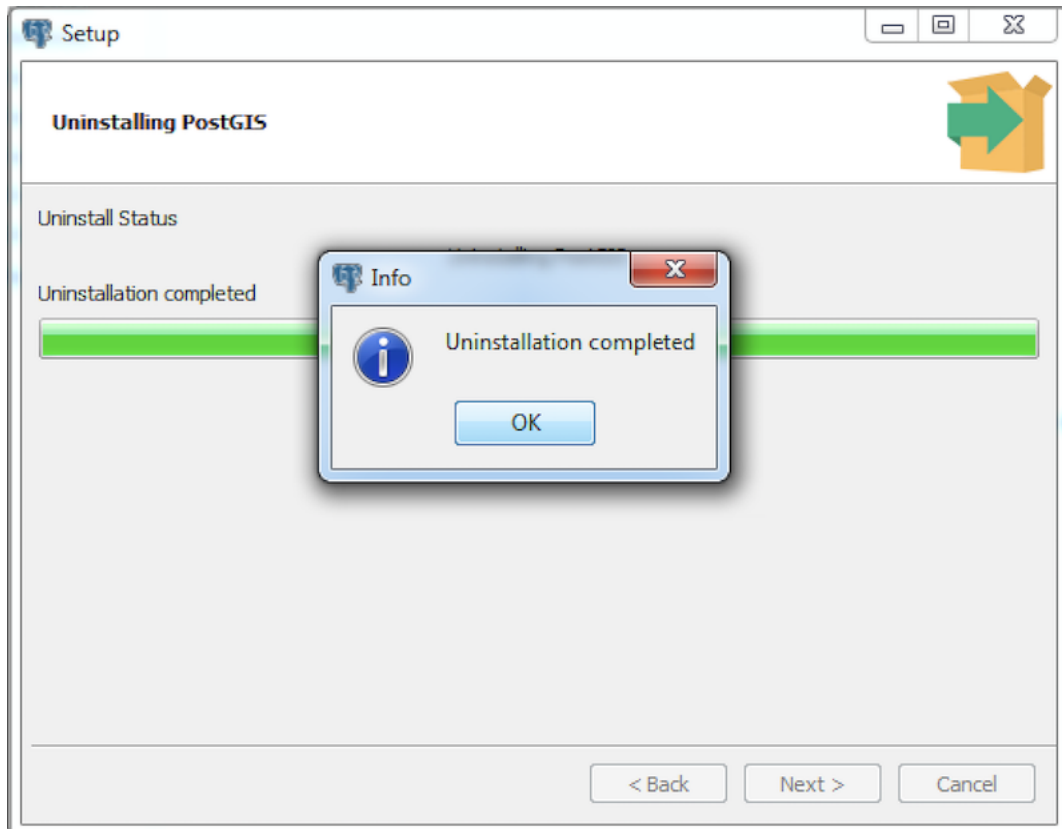


Fig. 10: Unin-

stallation is Complete