

# pgRouting安装与开发

---

日期：2019年11月14日

## 1. 了解pgRouting

---

pgRouting扩展了PostGIS/PostgreSQL地理空间数据库，以提供地理空间路由功能。

### 1.1 介绍PostGIS

PostGIS作为PostgreSQL的一个插件，通过向PostgreSQL添加对空间数据类型、空间索引和空间函数的支持，为PostgreSQL提供空间数据库的功能。

空间数据库像存储和操作数据库中其他任何对象一样去存储和操作空间对象。而将空间数据库和数据库关联起来的三个要素是：**空间数据类型**、**空间索引**和**函数**

- 空间数据类型用于指定图形为点（point）、线（line）和面（polygon）
- 多维度空间索引被用于进行空间操作的高效处理
- 空间函数构建于SQL语言中，用于进行空间属性和空间关系的查询

数据类型、空间索引和空间函数组合在一起，提供了灵活的结构用于空间数据库的性能优化和分析。

#### 空间数据类型

空间数据库除了拥有字符串、日期和数值等基础类型，还包括额外的空间类型表示地理特征，例如边界（boundary）和维度（dimension）等。

#### 空间索引

常见的实现是R-tree（在PostGIS中使用），但在其他空间数据库中也有基于四叉树（Quadrees）的实现和基于网格的索引（grid-based indexes）的实现。

#### 空间函数

函数操作一般分为五类：

- 转换 —— 在geometry（PostGIS中存储空间信息的格式）和外部数据格式之间进行转换的函数
- 管理 —— 管理关于空间表和PostGIS组织的信息的函数
- 检索 —— 检索几何图形的属性和空间信息测量的函数
- 比较 —— 比较两种几何图形的空间关系的函数
- 生成 —— 基于其他几何图形生成新图形的函数

### 1.2 pgRouting是什么

pgRouting在PostGIS/PostgreSQL的基础之上扩展了地理空间数据库，提供了地理空间路由的功能。通过数据库进行路由，数据和属性可以由多种客户端修改，比如QGIS和uDig可以通过JDBC、ODBC进行修改，或者直接使用PI/pgSQL进行修改。

pgRouting库包含一些核心的算法：

- Dijkstra算法
- 约翰逊算法
- 弗洛伊德-沃沙尔算法
- A\*算法

- 双向算法（双向Dijkstra算法、双向A\*算法）
- Driving Distance
- 转弯限制最短路径（TRSP）

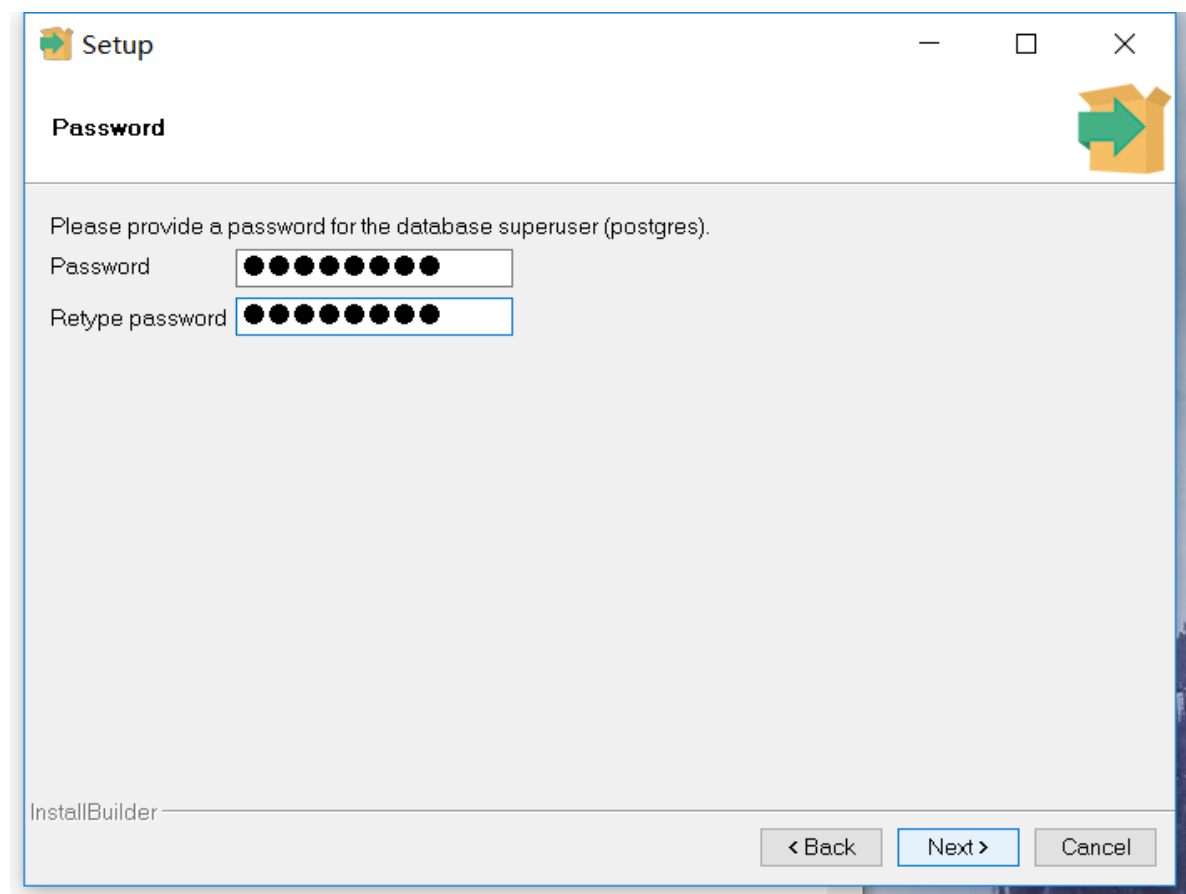
## 2. 如何安装pgRouting

pgRouting可以应用于Linux与Windows两种开发环境，本文重点介绍如何在Windows环境下安装pgRouting。

在Windows环境下，PostGIS的安装包里因为集成了pgRouting，所以安装pgRouting的过程比较简单，只需要在PostgreSQL中安装PostGIS插件。安装过程主要分为两步：PostgreSQL的安装和PostGIS的安装。PostGIS的安装依赖于PostgreSQL。

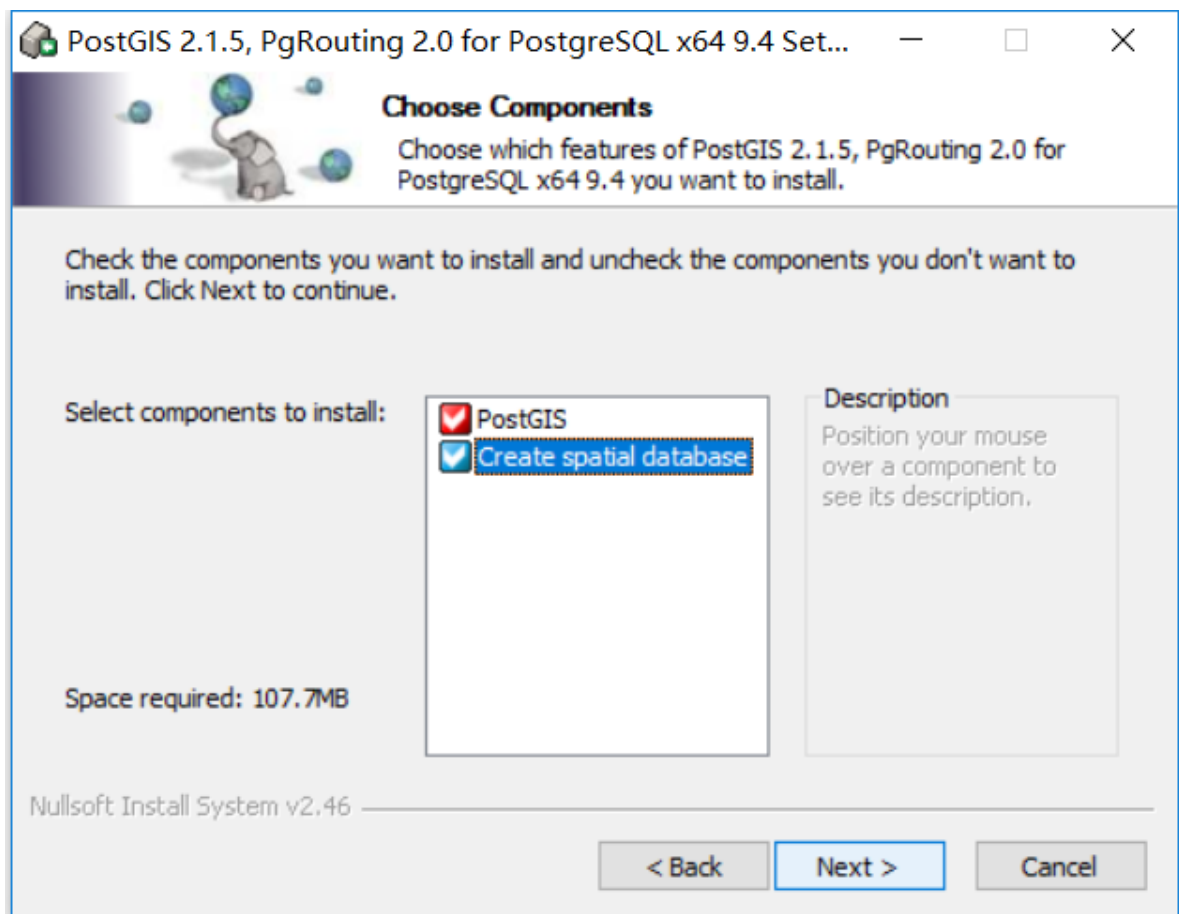
### 2.1 安装PostgreSQL

PostgreSQL安装文件下载地址是[PostgreSQL官网下载链接](#)，这里选取最新的版本12。下载完成后点击运行，中间需要记得设置超级用户postgres的密码。



### 2.2 安装PostGIS

PostgreSQL安装完成后，到[PostGIS官网](#)下载PostGIS安装包进行安装，注意选取对应的pg12目录下的安装包，这里对应PostgreSQL12的版本。下载完成点击运行，在设置安装组件时，需要勾选 **Create spatial database**选项，以便在创建数据库时可以以此作为模板。



在安装过程中注意，PostGIS的安装目录默认和 PostgreSQL同一目录，在设置PostGIS安装目录时需要和PostgreSQL设置为相同。

最后安装完成，弹出的三个对话框全部选择是(Y)。

### 3. 尝试pgRouting

安装已经完成，从开始菜单中选中 PostgreSQL管理工具**pgAdmin4**。pgAdminⅢ是pgAdmin4的上个版本，从pgAdminⅢ到pgAdmin4，这款管理工具完成了从C/S结构到B/S结构的蜕变。

启动pgAdmin之后，如果是第一次运行需要较长时间的一个初始化过程，耐心等待之后输入密码登录进入数据库。

#### 3.1 创建空间数据库

创建有两种方法：

1. 点击新建数据库，输入数据库名称之后，选择Definition，在模板选项内找到 postgis\_30\_template，点击保存，完成创建。
2. 点击新建数据库，输入数据库名称之后之间点击保存，然后运行SQL查询工具，执行下列查询语句完成创建PostGIS插件。

```
CREATE EXTENSION postgis;
```

接着执行下条查询语句完成pgRouting的创建。

```
CREATE EXTENSION pgRouting
```

#### 3.2 获取.shp路网数据

PostGIS安装中包含了Shapefile数据导入工具，但是只能将shapefile文件导入数据库，但是往往OpenStreetMap上下载的路网数据格式是osm，所以需要通过在ArcGIS Desktop软件中使用ArcGIS Editor for OSM插件将osm文件转换为shapefile文件。

### 3.2.1 ArcGIS Desktop安装

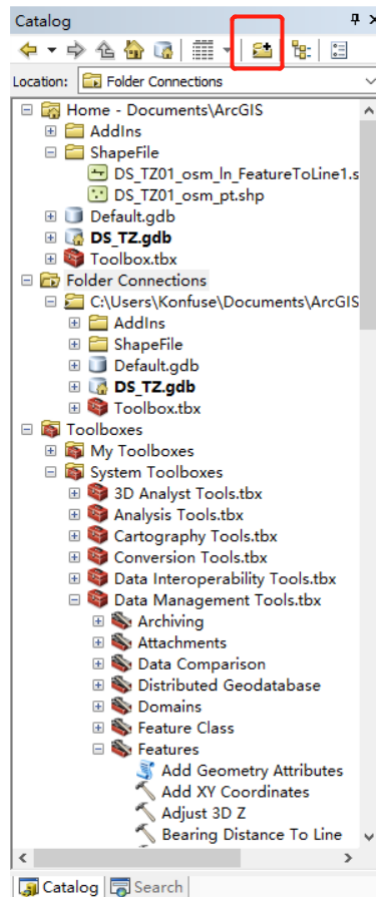
1. 下载安装包与补丁，这里选择的版本是ArcGIS10.6，ArcGIS Desktop软件的安装分为两部分，第一部分是License Manager，负责License管理，第二部分是ArcGIS Desktop。
2. 首先安装 ArcGIS\_License\_Manager\_windows\_106\_161708.exe，安装过程中会首先选择安装文件解压目录，完成解压后，程序自动进入安装过程，安装完成之后，会自动弹出ArcGIS License Manager的界面，此时需要点击 启动/停止License服务 选项卡中的 停止 按钮。
3. 然后进入ArcGIS 10.6 for Desktop安装，点击运行 ArcGIS\_Desktop\_106\_161544.exe，还是会弹出和ArcGIS License Manager一样解压对话框，解压完成后会进入安装过程，安装中会弹出一个Python2.7环境安装的对话框，因为ArcGIS一些功能依赖于这个版本的Python，在配置阶段并不会把Python加入系统环境变量中，安装完成之后会弹出ArcGIS Administrator的运行界面，则安装结束。
4. 选择补丁文件中Desktop10.6与License10.6中文件，放入到安装路径对应的目录中去。
5. 启动License Manager，用 License Server Administrator 打开，点击 启动/停止License服务 选项卡中的 启动 按钮，保证底部显示 RUNNING 状态。
6. 配置 ArcGIS Administrator，依次选择 Desktop，Advanced(ArcInfo)浮动版，更改 的选项卡，并将地址改为 localhost，然后点击 可用性，并刷新，可以看到软件可用的状态。

### 3.2.2 安装ArcGIS Editor for OSM扩展工具

ArcGIS Editor for OSM就是基于ArcGIS Desktop的一个扩展工具，下载版本时需要和ArcGIS Desktop的版本对应，这里选择10.6的版本。下载完成后解压，点击 setup.exe 进行安装。

### 3.2.3 将osm路网数据转换为shapefile文件

打开 ArcMap 软件，在软件内找到 Catalog 选项卡，选中 Folder Connections，在其中新建后缀为.gdb的地理数据库



然后依次点击 Catalog 中的 Toolboxes->System Toolboxes->OpenStreetMap Toolbox.tbx->Load OSM File 导入osm文件到新创建的数据库中。

导入数据库之后，在对应的数据上右键选择 Export，选择到处目录，将数据导出为shapefile文件输入文件中。

### 3.3 导入.shp文件到PostGis空间数据库

1. 打开 PostGIS 3 Shapefile and DBF Loader Exporter 工具
2. 首先点击 View Connection details... 连接到已经创建好的空间数据库
3. 然后点击 Add File，选择需要加入的.shp文件，添加后注意配置 SRID 为对应的坐标系。
4. 导入之前配置点击 options... 设置导入参数，然后勾选 Import Options 中的最后一个选项，使数据以单几何图形的形式导入到数据库。
5. 最好点击 import 导入即可。

### 3.4 为道路数据创建路由拓扑

首先为道路添加source列、target列以及length列：

```
ALTER TABLE ways
ADD soure bigint,
ADD target bigint,
ADD length double precision;
```

然后计算出每条路径边的长度保存在length列中：

```
UPDATE ways
SET length = ST_Length(geom);
```

再调用pgr\_createTopology()方法为道路表创建路由拓扑：

```
SELECT pgr_createTopology('ways', 0.000001, 'geom', 'gid')
```