# shp导入PostGIS

## 数据处理

### Excel转shp

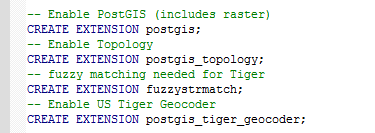
使用ArcGIS添加XY数据

## 数据发布

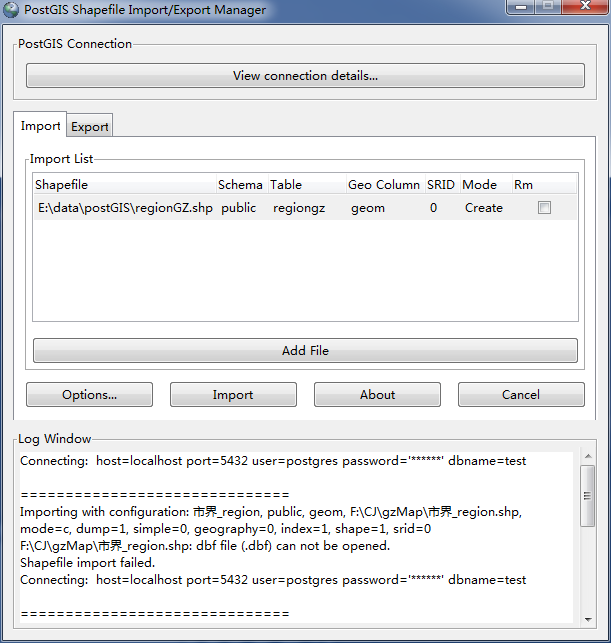
### PostGIS导入导出shp的注意事项

1、postgresql 数据库的表明使用小写，表名不能使用大写，否则会识别不了。

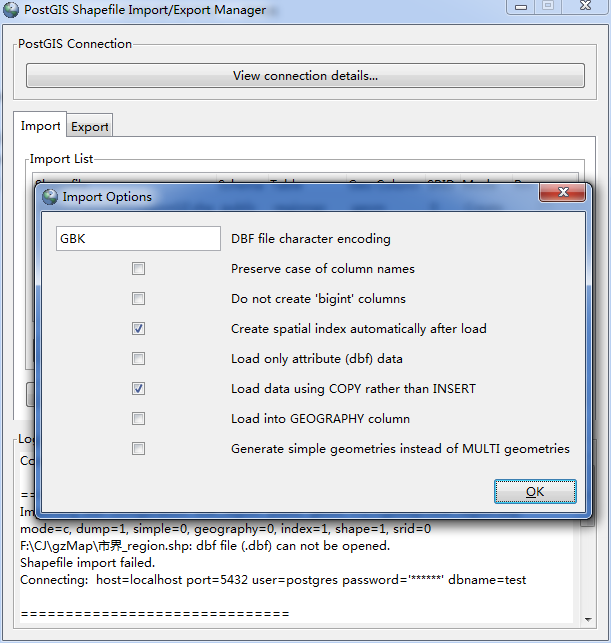
2、数据导入之前需要给需要导入的数据库增加空间扩展功能。



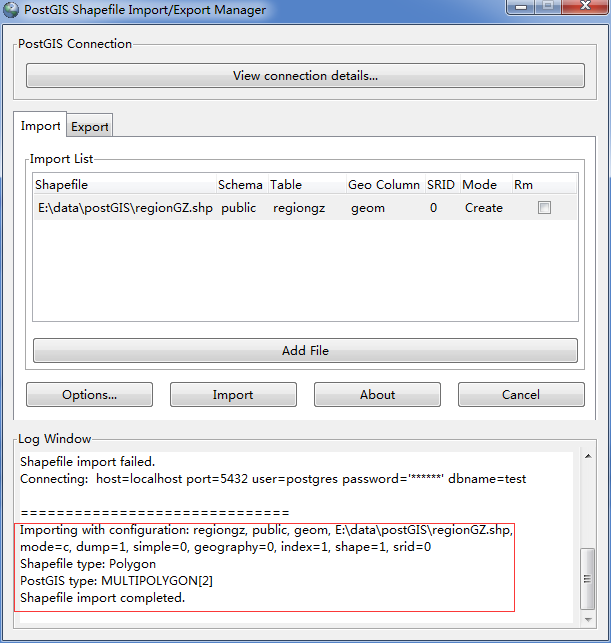
3、使用 postgis shapfile and db loader 插件时，选择的SHP文件路径不能有中文，路径不能太复杂。否则出现 dbf file (.dbf) can not be opened.的错误。



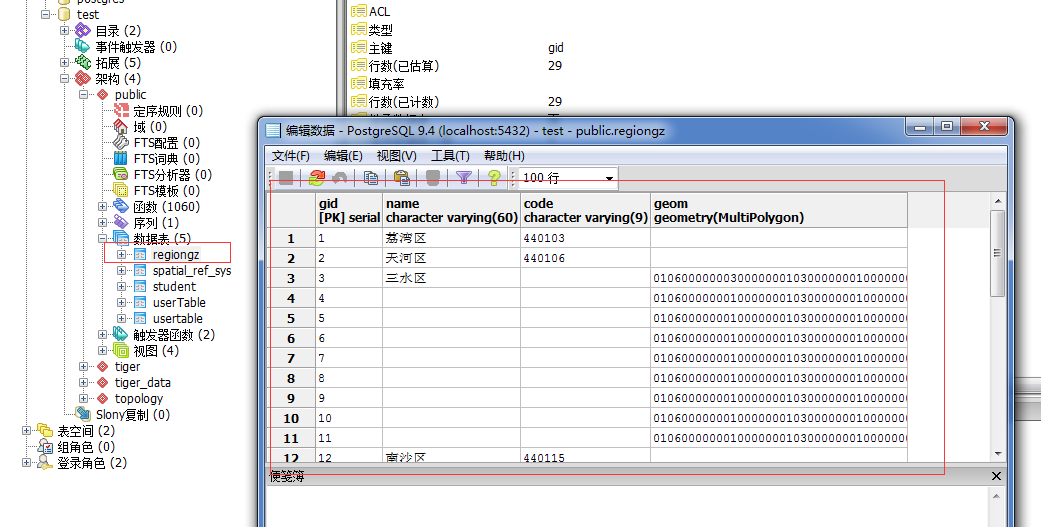
4、导入默认的编码是UTF-8，需要修改为GBK，否则会报错。



5、按照以上步骤，基本能够导入成功。



6、导入成功的SHP文件名会自动变成小写名称，并用做表名。如下图所示为，数据导入后的结果。



**SHP的导出**

shp的导出实际上是用的是bin\pgsql2shp.exe

1、注意编码问题，由于导出的Options没有编码选项可供选择设置。

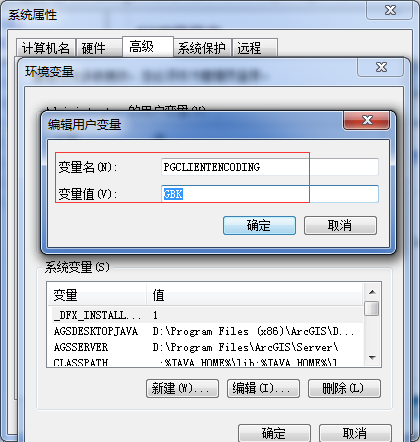
尝试1）：进入PSQL Console修改客戶端字符集，結果无效，設置为GBK后導出的shp属性表中中文依旧乱码

**--修改客戶端字符集**postgres=# show client\_encoding;  
 client\_encoding  UTF8  
(1 row)

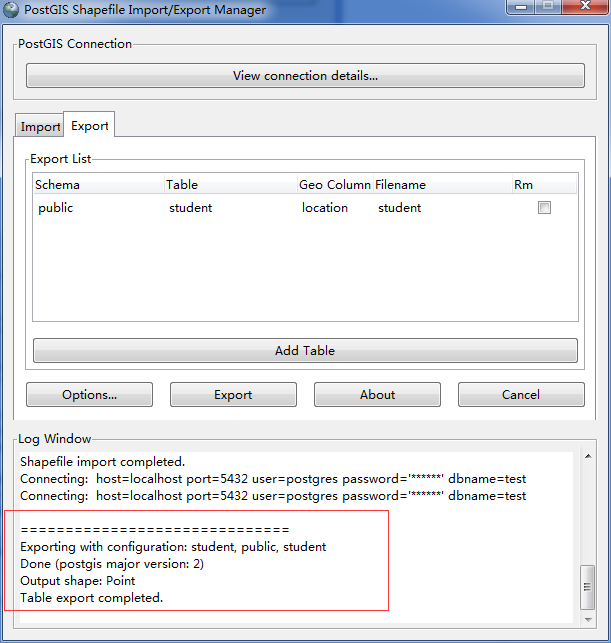
postgres=# set client\_encoding=‘GBK‘;  
SET

尝试2）：设置系统的环境变量 PGCLIENTENCODING=GBK，退出PostgreSQL重新登錄，执行成功！。

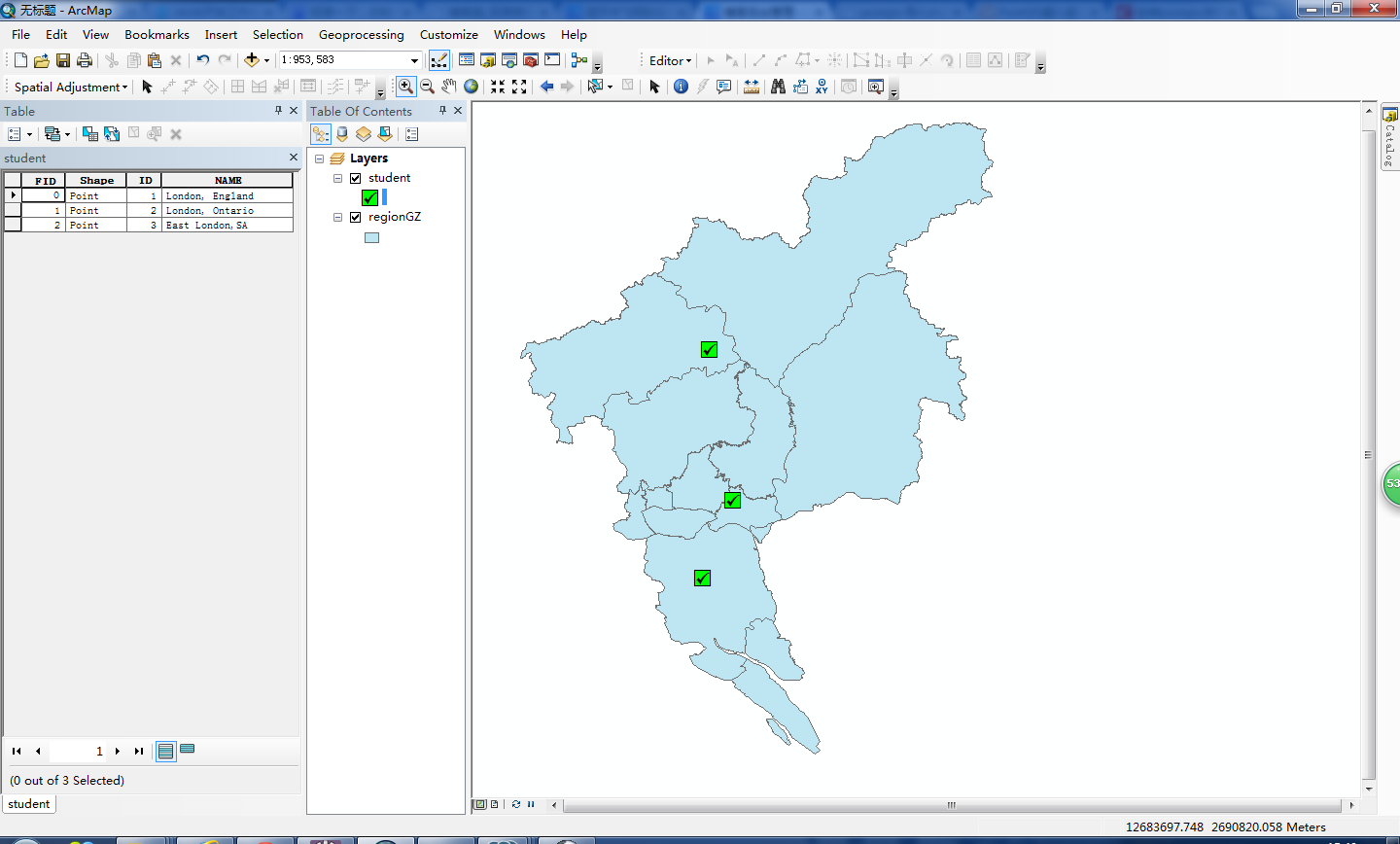
注意：编码问题，由于导出的options没有编码选项可供设置。



然后执行导出操作，导出提示成功。



使用ArcMap打开数据进行查看。



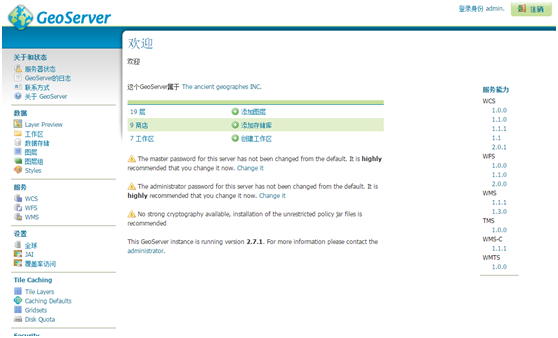
导出的数据没有.prj文件，要加得用ArcCatalog來进行设置。

參考文章里还提到了用命令行进行导入导出的方法，暂且没有尝试，以后有需要再研究。

### GeoServer发布PostGIS数据

GeoServer除了浏览以外，其他譬如发布、删除等等操作均需要用户登陆，默认的账号和密码为:admin , geoserver .

登陆后的首页应该是这样的：



发布一个layer正确的顺序为：工作区->datastore(数据存储)->layer(图层)。

需要先创建工作区，再创建数据源，然后再从数据源中选择数据发布图层。现以PostGIS发布一个图层。

#### 工作区



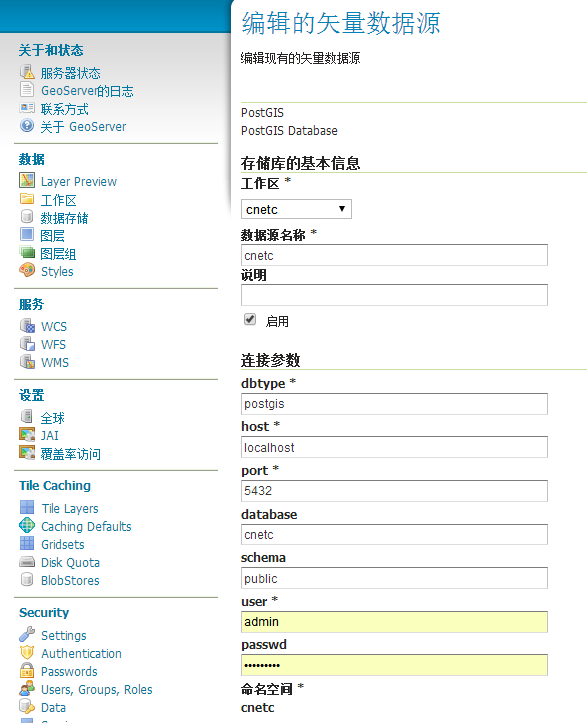
#### 数据存储



添加



填写数据库参数



#### 图层

添加新图层



选择数据存储项

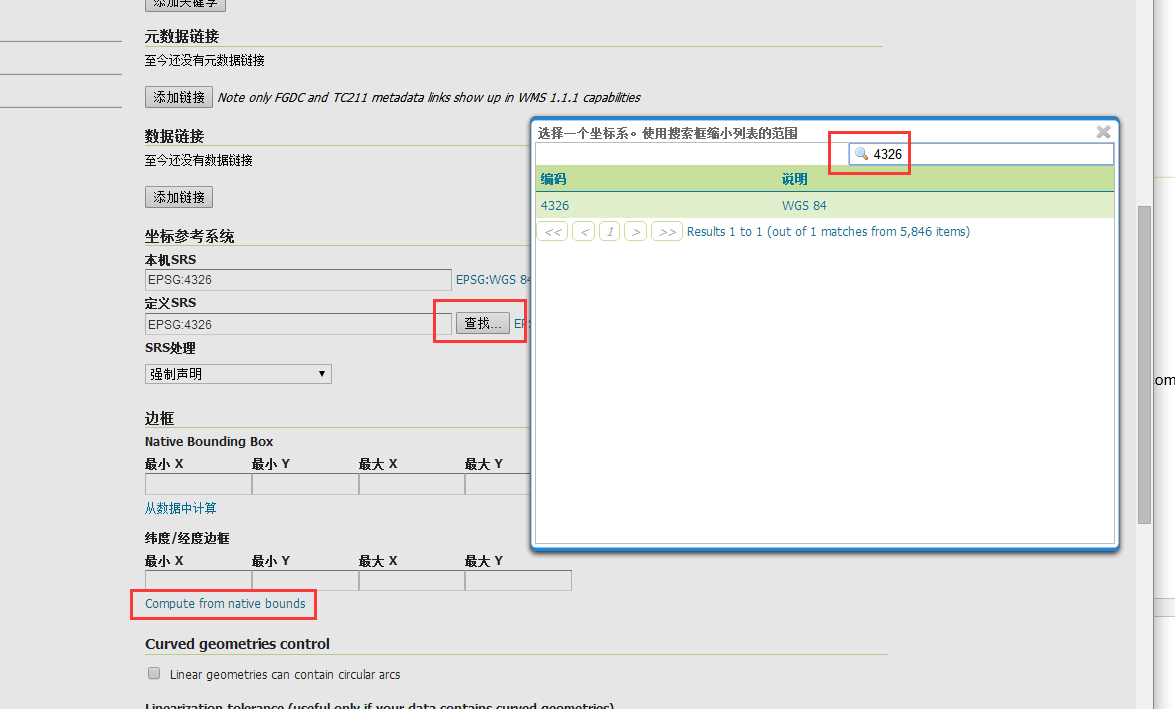


发布图层

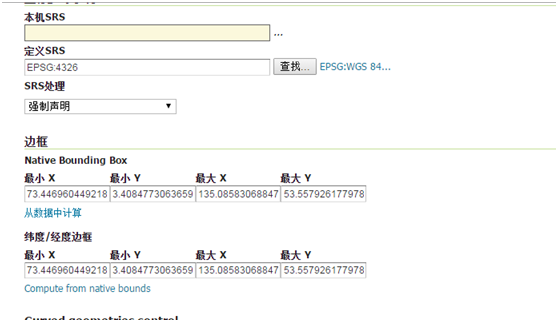




 需要注意的地方有SRS不能为空，需要自己添加，4326为最常用的经纬度坐标系



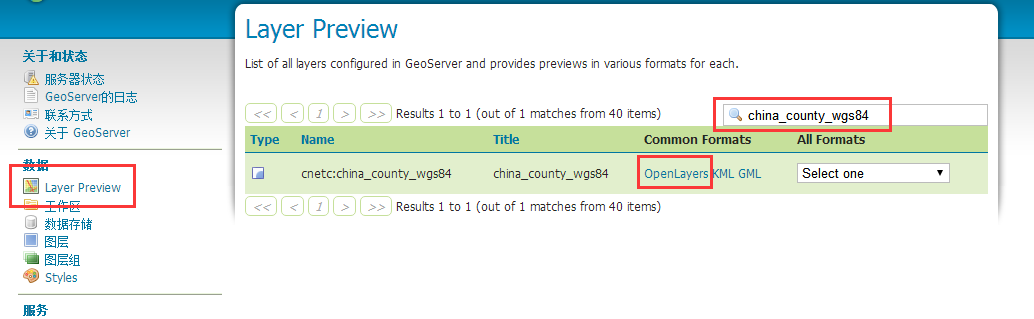
如果读不出需要手动填写一下，填写最小的范围矩阵，可以充点击下方的“compute from native bounds”自动计算。



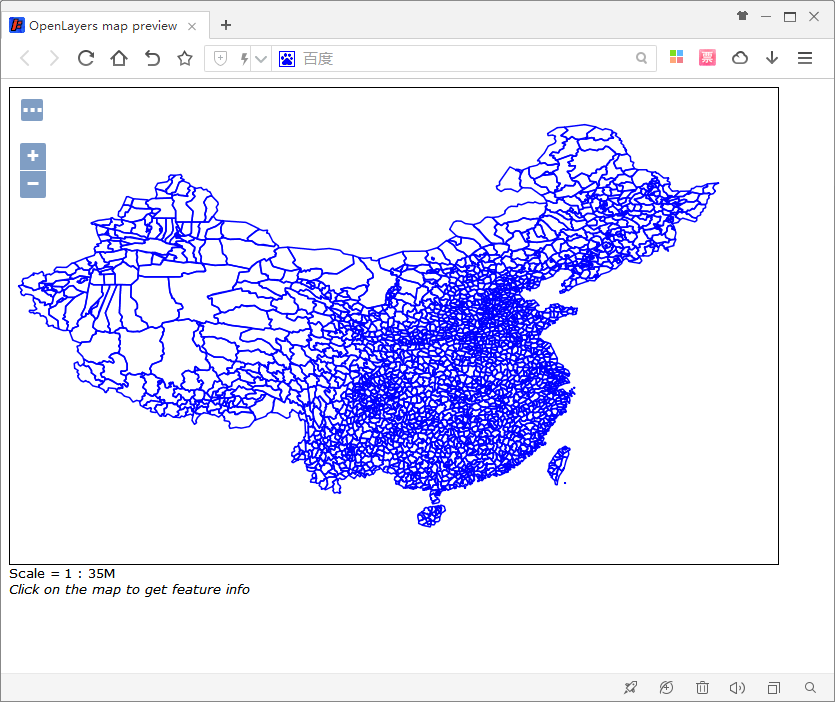
另外提示：如果需要在发布layer的同时指定这个layer的默认样式，那么选择tab页到“发布”中重置默认的 “default style”



发布完成后，预览。点击openlayers可以直接进行ol的预览。



效果如下：



另外说明一下 geoserver2.7的版本采用的ol已经是ol3的版本了，看起来是不是和ol2稍有变化呢。

## 前端应用

### GeoServer跨域

在webapps/geoserver/WEB-INF/web.xml中添加

<context-param>

<param-name>ENABLE\_JSONP</param-name>

<param-value>true</param-value>

</context-param>

### 获取GeoJson

function loadGeoJson() {

var baseGeoServerPath="http://127.0.0.1:8686/geoserver/";

var owsrootUrl = baseGeoServerPath + 'cnetc/ows';

var defaultParameters = {

service: 'WFS',

version: '1.0.0',

request: 'GetFeature',

typeName: 'cnetc:gis\_friendlycity',

outputFormat: 'text/javascript',

format\_options: 'callback:getJson',

SrsName: 'EPSG:4326'

};

var parameters = L.Util.extend(defaultParameters);

var URL = owsrootUrl + L.Util.getParamString(parameters);

var ajax = $.ajax({

//type: "get",

//async: false,

url: URL,

dataType: 'jsonp',

jsonpCallback: 'getJson',

success: loadJson,

jsonp: 'callback'

});

}

# SQL创建

## 手工建立空间数据表格

　　空间数据库已经建立，现在可以建立具有空间信息的表格。

　　首先建立一个常规的表格存储有关城市（cities）的信息。这个表格有两栏，一个是 ID 编号，一个是城市名：

demo=# CREATE TABLE cities ( id int4, name varchar(50) );

　　现在添加一个空间栏用于存储城市的位置。习惯上这个栏目叫做 the\_geom 。它记录了数据为什么类型（点、线、面）、有几维（这里是二维）以及空间坐标系统。此处使用 EPSG:4326 坐标系统：

demo=# SELECT AddGeometryColumn ('cities', 'the\_geom', 4326, 'POINT', 2);

完成后，查询 cities 表单应当显示这个新栏目。同时页面将显示当前表达没有记录（0 rows）。

AddGeometryColumn会自动为你所要的表增加一个代表坐标点的数据列，名为：the\_geom,为geometry类型

AddGeometryColumn(数据库schema, 图形表名, 'the\_geom', 投影坐标(eg: 4326), 图形类型(eg: POLYGON), 图形维数(eg: 2));

demo=# SELECT \* from cities;

id | name | the\_geom

----+------+----------

(0 rows)

　　为添加记录，需要使用 SQL 命令。对于空间栏，使用 PostGIS 的 ST\_GeomFromText 可以将文本转化为坐标与参考系号的记录：

demo=# INSERT INTO cities (id, the\_geom, name) VALUES (1,ST\_GeomFromText('POINT(-0.1257 51.508)',4326),'London, England');

demo=# INSERT INTO cities (id, the\_geom, name) VALUES (2,ST\_GeomFromText('POINT(-81.233 42.983)',4326),'London, Ontario');

demo=# INSERT INTO cities (id, the\_geom, name) VALUES (3,ST\_GeomFromText('POINT(27.91162491 -33.01529)',4326),'East London,SA');

当然，这样的输入方式难以操作。其它方式可以更快的输入数据。就目前来说，表格内已经有了一些城市数据，可以先进行查询等操作。

更新SQL

UPDATE data\_idspark SET geom = st\_geomfromtext('POINT(116.407159 39.90467)',4326) where gid=21;