



链滴

开源 GIS 快速入门 + docker 一键启动环境

作者: [Hawkpool](#)

原文链接: <https://ld246.com/article/1578378438066>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



我是革命一块砖，哪里需要往哪搬！身为全栈做的活挺多了，不过公司最近要用 geoserver 发布地图务，又要开始了一个陌生的领域的探索。

因为缺少经费，且需求暂不明确，暂且不考虑需要付费但功能强大的ArcGis,于是参考了[开源WebGIS](#)结构

以上已经给出了用开源软件实现WebGIS的各个环节可采用的软件，但在实际中，还需要根据项目实际做一下筛选。如果是Java版项目，推荐uDig + PostgreSQL/PostGIS + Tomcat + GeoServer + Openlayers组合。

经前端后端 ui 以及 cto 的联合商讨，大致确定

- 由 ui 使用 qgis/arcgis 出图，出 shp 文件
- 后端搭建 PostgreSQL/PostGIS + GeoServer 处理 shp 文件入库及发布
- 前端使用 Openlayers进行地图展示

Geoserver 是一款开源免费的地图服务器，功能十分强大。或许我们会碰到这样一个场景，工作在内网，不能使用外网的天地图资源，这时我们只能把需要的地图下载下来用 geoserver 发布了。然而对多层次的数据地图数据发布方案不是很明确。我们用一些地图下载器下载的资源一般可以分为三种吧：大图拼接(.tif)格式，各种规范的瓦片，各种规范的瓦片包。

环境配置： docker 一键启动

Docker 是个好东西，有了它环境问题再也不用愁了~~

此处奉上使用 docker-compose 编排的一键启动脚本：

包含 PostSql+PostGis+GeoServer+pgadmin(数据库图形化管理界面 Web 版)

```
version: '3.1'
services:
  geoserver:
```

```
restart: always
image: kartoza/geoserver
container_name: geoserver
ports:
- 8888:8080
volumes:
- ./geoserver-data:/opt/geoserver/data_dir
```

```
postgis:
restart: always
image: kartoza/postgis
container_name: postgis
ports:
- 5432:5432
environment:
POSTGRES_USER: hawk
POSTGRES_PASSWORD: 123456
volumes:
- ./postgis-data:/var/lib/postgresql/data
```

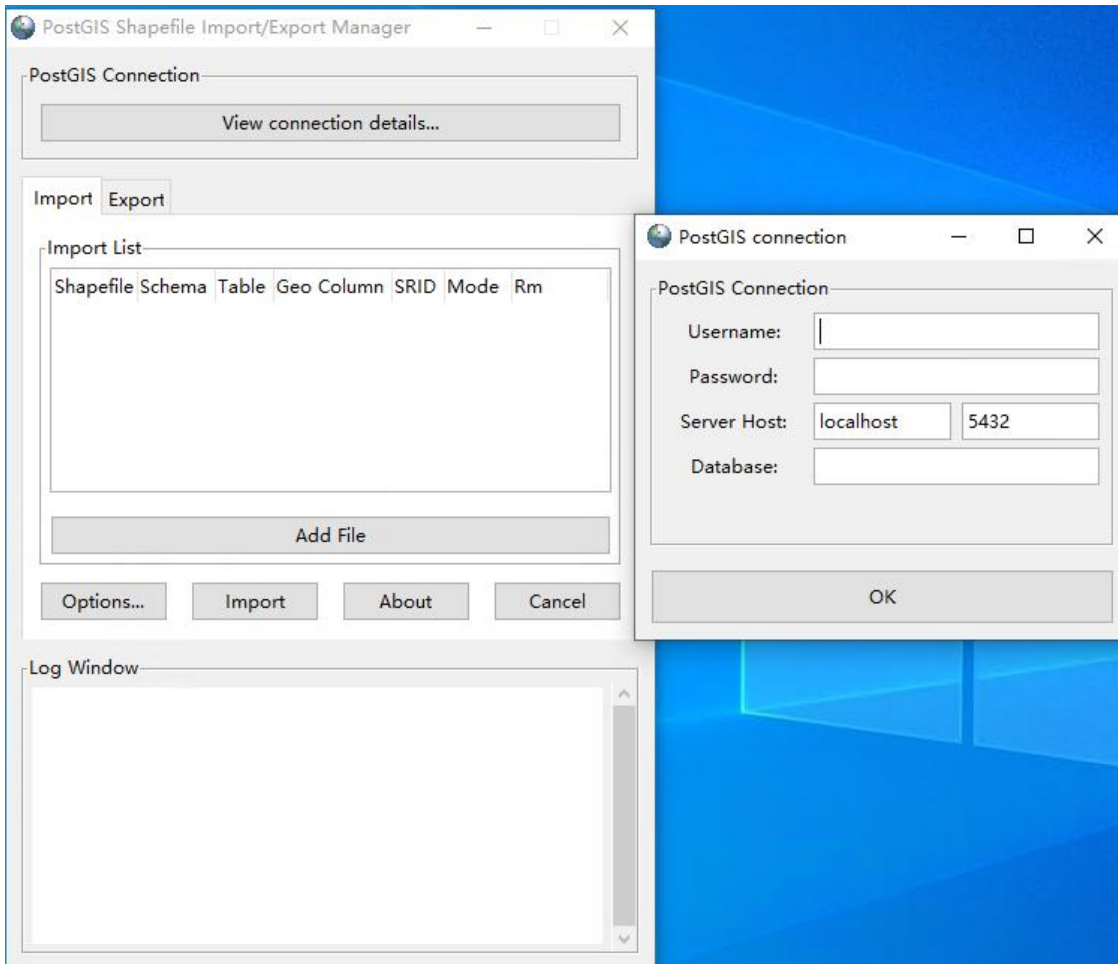
```
pgadmin4:
restart: always
image: dpage/pgadmin4
container_name: pgadmin
ports:
- 8090:80
environment:
PGADMIN_DEFAULT_EMAIL: hawk@XXXX.com
PGADMIN_DEFAULT_PASSWORD: 123456
```

后端需要做的事主要包括

配置整体 GIS 环境(一键启动即可)

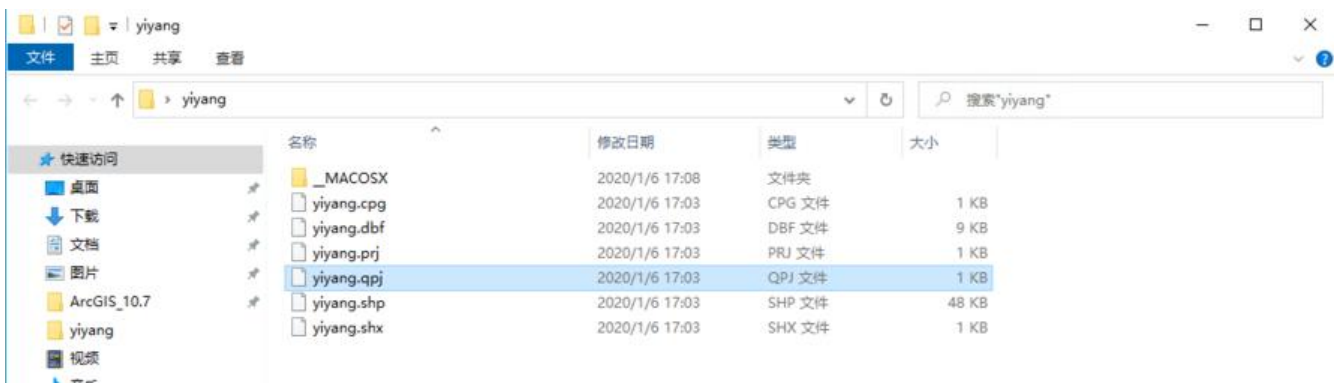
从 ui 获取 shp 文件，操作入库(也可让 ui 自行操作)

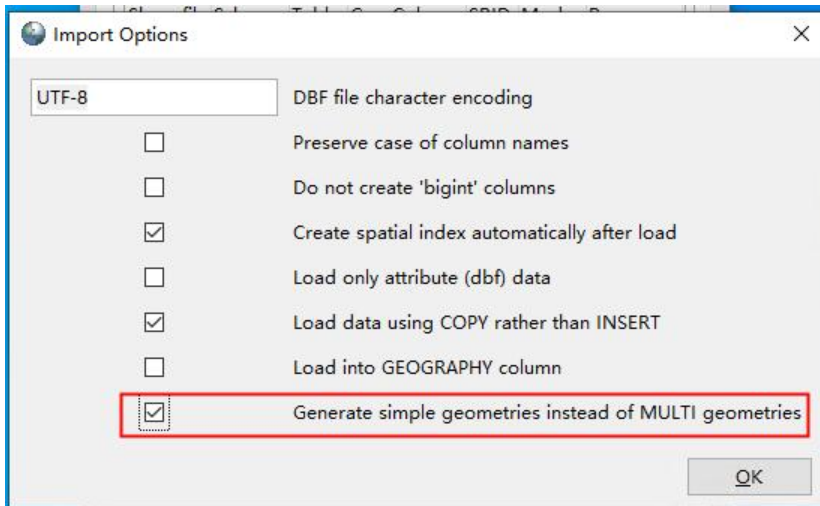
可以通过 **PostGis** 文件导入工具 导入 **PostSql**



切记

- shp 文件存放目录尽量简单，不要包含中文(否则容易报错)
- 配置编码正确格式，包含中文的话尽量使用 UTF-8 或者 GBK
- shp 文件同时还有 dbf 文件、prj 等文件考到同一个目录下，名字要一致





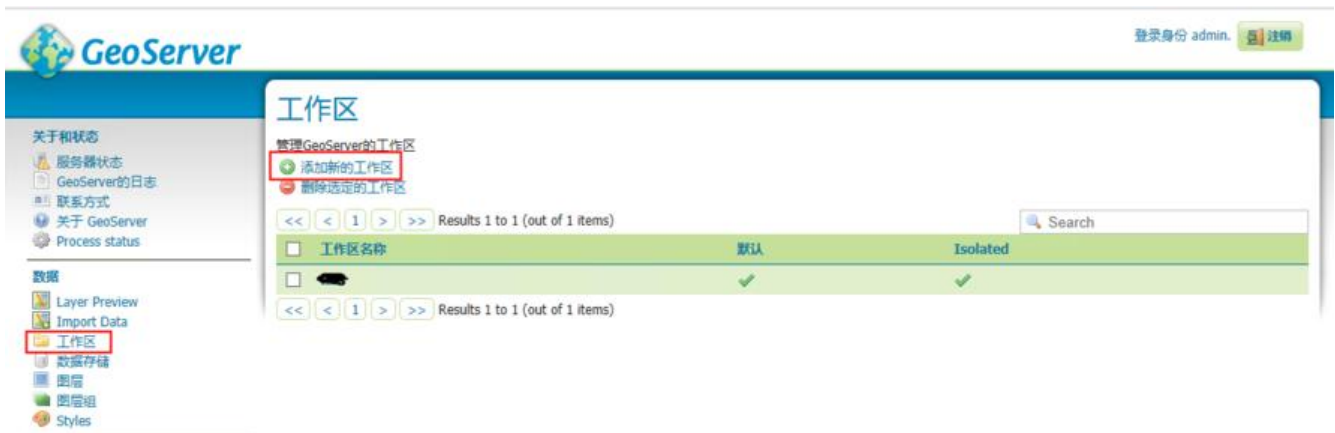
该工具需要 windows 本地安装 PostSql ,并安装 PostGis 方可链接数据库
给大家分享一个 postgis-ui 的下载链接, 使用这个不需要安装 pgAdmin
链接: [下载链接](#) 密码: l1ml

配置 GeoServer

浏览器打开: <http://docker宿主机ip:8888/geoserver>

使用默认 admin 账户登录: admin/geoserver

配置工作区



配置数据存储

geoserver 原生支持识别 shp 文件, 但是仅支持事变本机目录文件, 不支持远程上传
遂通过 postgresql 导入, 实现更佳的可可用性

关于和状态

- 服务器状态
- GeoServer的日志
- 联系方式
- 关于 GeoServer
- Process status

数据

- Layer Preview
- Import Data
- 工作区
- 数据存储**
- 图层
- 图层组
- Styles

数据存储

管理GeoServer的数据存储

[添加新的数据存储](#)

[删除选定的数据存储](#)

Results 1 to 1 (out of 1 items)

数据源名称	工作区	数据存储名称	类型	启用?
			PostGIS	✓

Results 1 to 1 (out of 1 items)

新建数据源

选择你要配置的数据源的类型

矢量数据源

- CSV - Comma delimited text file
- Directory of spatial files (shapefiles) - Takes a directory of shapefiles and exposes it as a data store
- GeoPackage - GeoPackage
- PostGIS - PostGIS Database**
- PostGIS (JNDI) - PostGIS Database (JNDI)
- Properties - Allows access to Java Property files containing Feature information
- Shapefile - ESRI(tm) Shapefiles (*.shp)
- Web Feature Server (NG) - Provides access to the Features published a Web Feature Service, and the ability to perform transactions on the server (when supported / allowed).

栅格数据源

- ArcGrid - ARC/INFO ASCII GRID Coverage Format
- GeoPackage (mosaic) - GeoPackage mosaic plugin
- GeoTIFF - Tagged Image File Format with Geographic information
- ImageMosaic - Image mosaicking plugin
- ImagePyramid - Image pyramidal plugin
- WorldImage - A raster file accompanied by a spatial data file

其他数据源

- WMS - 悬挂一个远程网站地图服务
- WMTS - Cascades a remote Web Map Tile Service

完成 PostGIS 配置后，会自动跳转到图层配置

配置图层

数据库中的数据就是地图的矢量数据

图层有点像 ps 里的一个图层，通过解码对应坐标系解析矢量数据展示相应图像

图层组就像是 ps 中多个图层组合展示的画面

关于和状态

- 服务器状态
- GeoServer的日志
- 联系方式
- 关于 GeoServer
- Process status

数据

- Layer Preview
- Import Data
- 工作区
- 数据存储
- 图层**
- 图层组
- Styles

图层

管理GeoServer发布的图层

[添加新的资源](#)

[删除所选的资源](#)

Results 1 to 2 (out of 2 items)

类型	Title	图层名称	存储	启用?	Native SRS
面	yiyang	tulu:yiyang	tulu	✓	EPSG:900913
点	yyxd	tulu:yyxd	tulu	✓	EPSG:900913

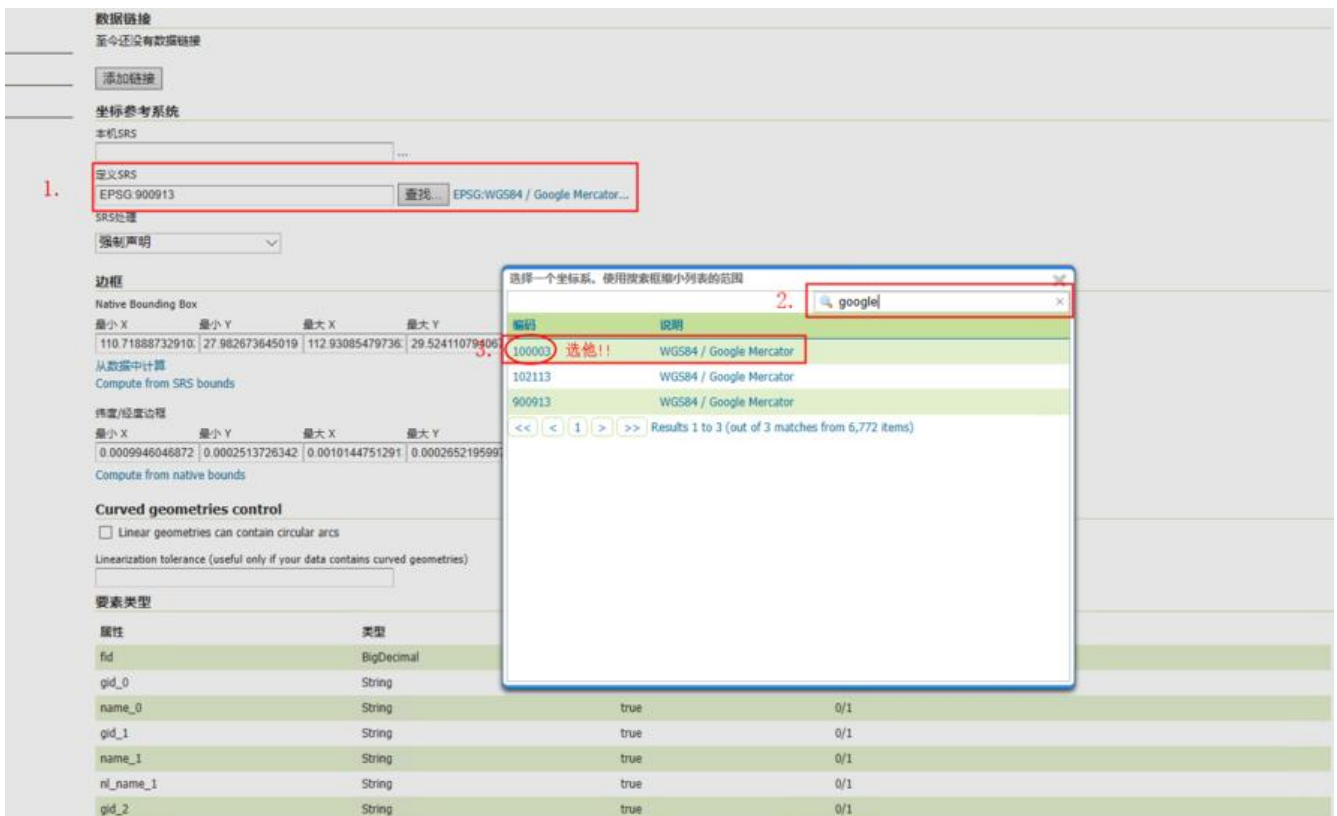
Results 1 to 2 (out of 2 items)

新增图层操作

先配置图层数据库，GeoServer 会自动加载数据库已上传的 shp 文件，即可选中相应图层点击发布



配置坐标系系数，计算边框，渲染地图



数据链接
至今还没有数据链接

[添加链接](#)

坐标参考系统

本机SRS

定义SRS
 [查找...](#) EPSG:WGS84 / Google Mercator...

SRS处理

边框

Native Bounding Box

最小 X	最小 Y	最大 X	最大 Y
110.71888732910	27.982673645019	112.93085479736	29.524110794067

[从数据中计算](#)
[Compute from SRS bounds](#)

纬度/经度边框

最小 X	最小 Y	最大 X	最大 Y
0.0009946046872	0.0002513726342	0.0010144751291	0.0002652195997

[Compute from native bounds](#)

Curved geometries control

Linear geometries can contain circular arcs.

Linearization tolerance (useful only if your data contains curved geometries)

要素类型

属性	类型	Nullable	Min/Max Occurrences
fid	BigDecimal	true	0/1
gid_0	String	true	0/1
name_0	String	true	0/1

保存成功后即可查看预览图

查看图层预览图

GeoServer 登录身份 admin 注册

Layer Preview

List of all layers configured in GeoServer and provides previews in various formats for each.

<< < 1 > >> Results 1 to 3 (out of 3 items)

Type	Title	Name	Common Formats	All Formats
	yiyang	tulu:yiyang	OpenLayers GML KML	Select one
	yyxd	tulu:yyxd	OpenLayers GML KML	Select one
	益阳测试	tulu:益阳测试	OpenLayers KML	Select one

<< < 1 > >> Results 1 to 3 (out of 3 items)

Layer Preview

List of all layers configured in GeoServer and provides previews in various formats for each.

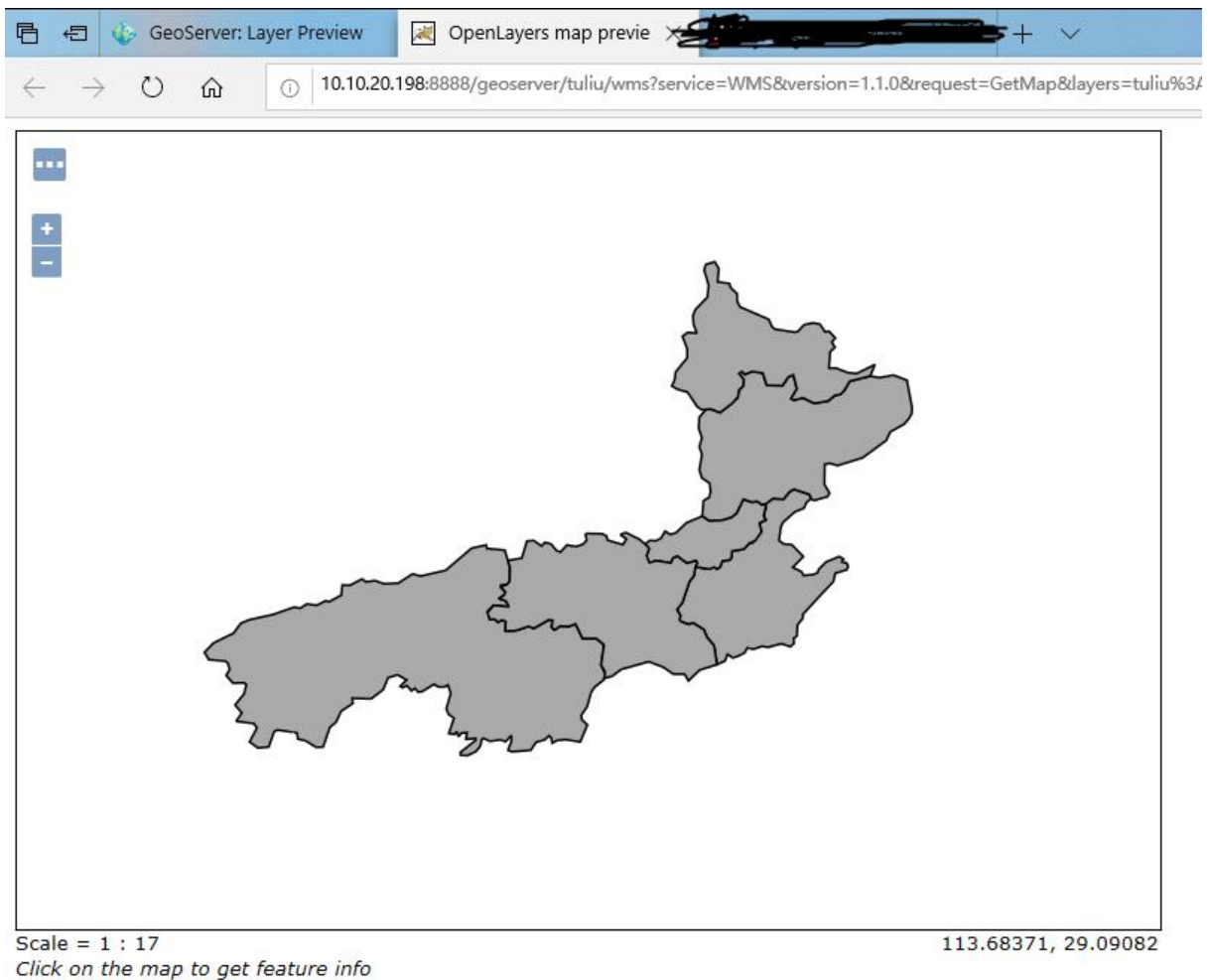
<< < 1 > >> Results 1 to 3 (out of 3 items)

Type	Title	Name	Common Formats	All Formats
	yiyang	tulu:yiyang	OpenLayers GML KML	Select one
	yyxd	tulu:yyxd	OpenLayers GML KML	Select one
	益阳测试	tulu:益阳测试	OpenLayers KML	Select one

<< < 1 > >> Results 1 to 3 (out of 3 items)

点我!!

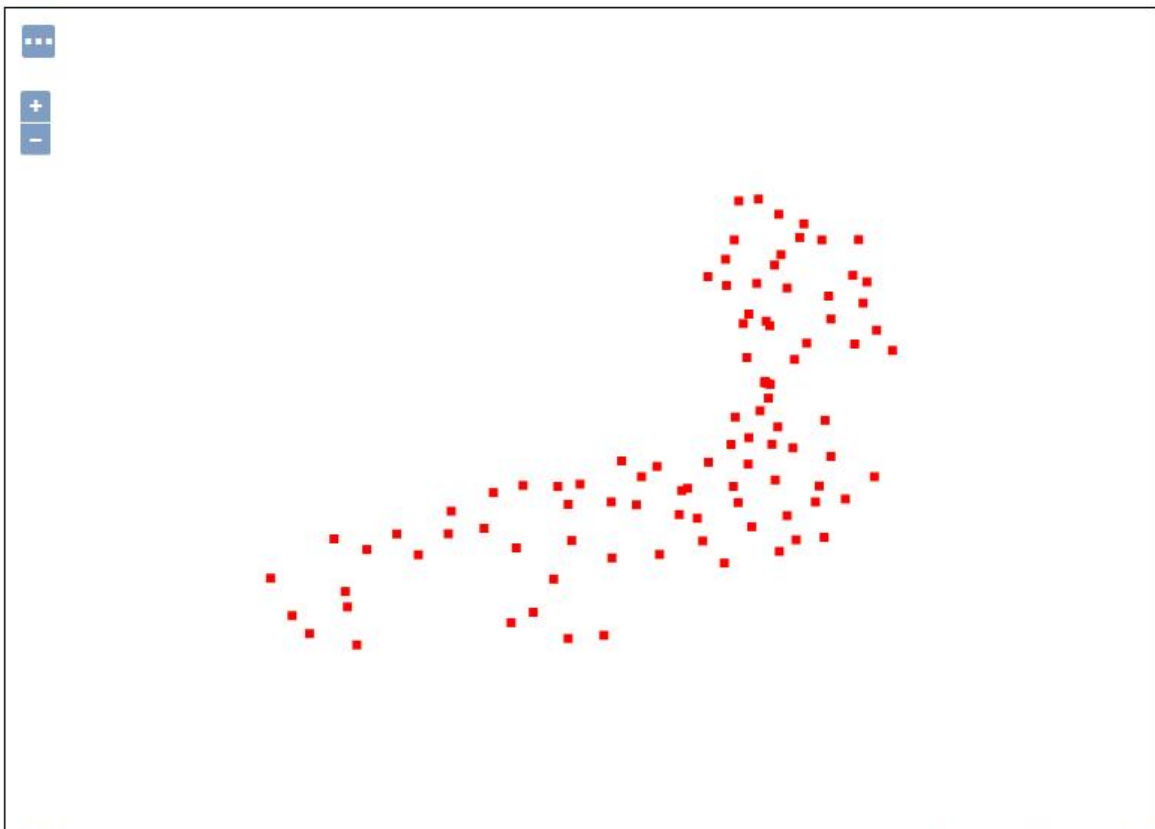
点击 OpenLayers 后会打开一个新的页面



一个地图边框图层预览，GET!

配置图层组

我们再通过上面操作配置一个点状图图层预览

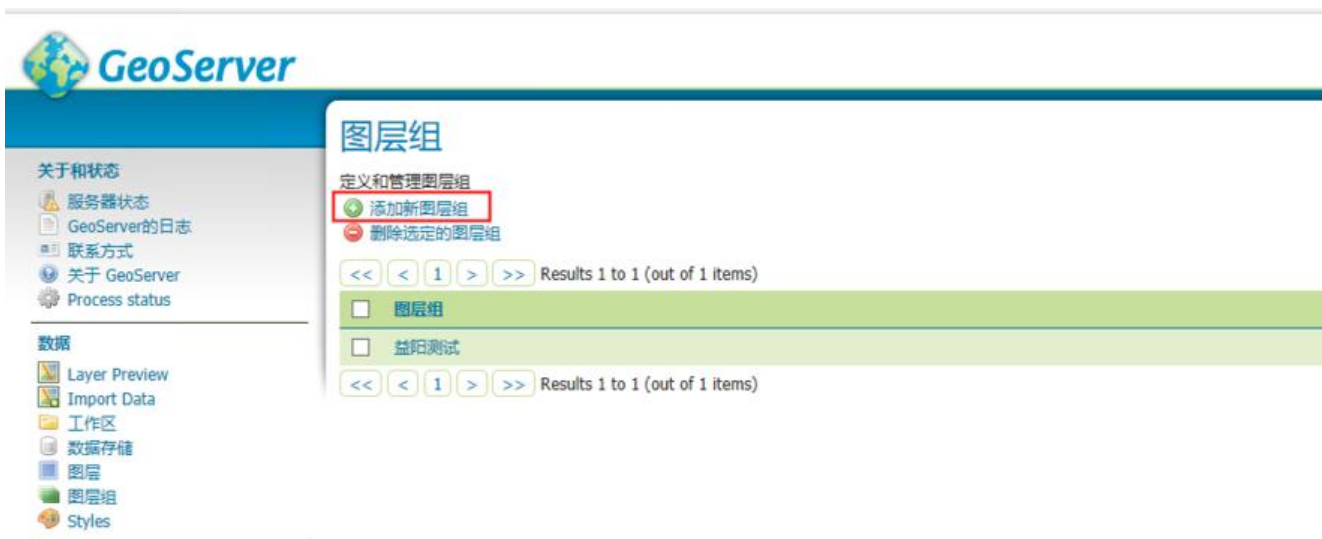


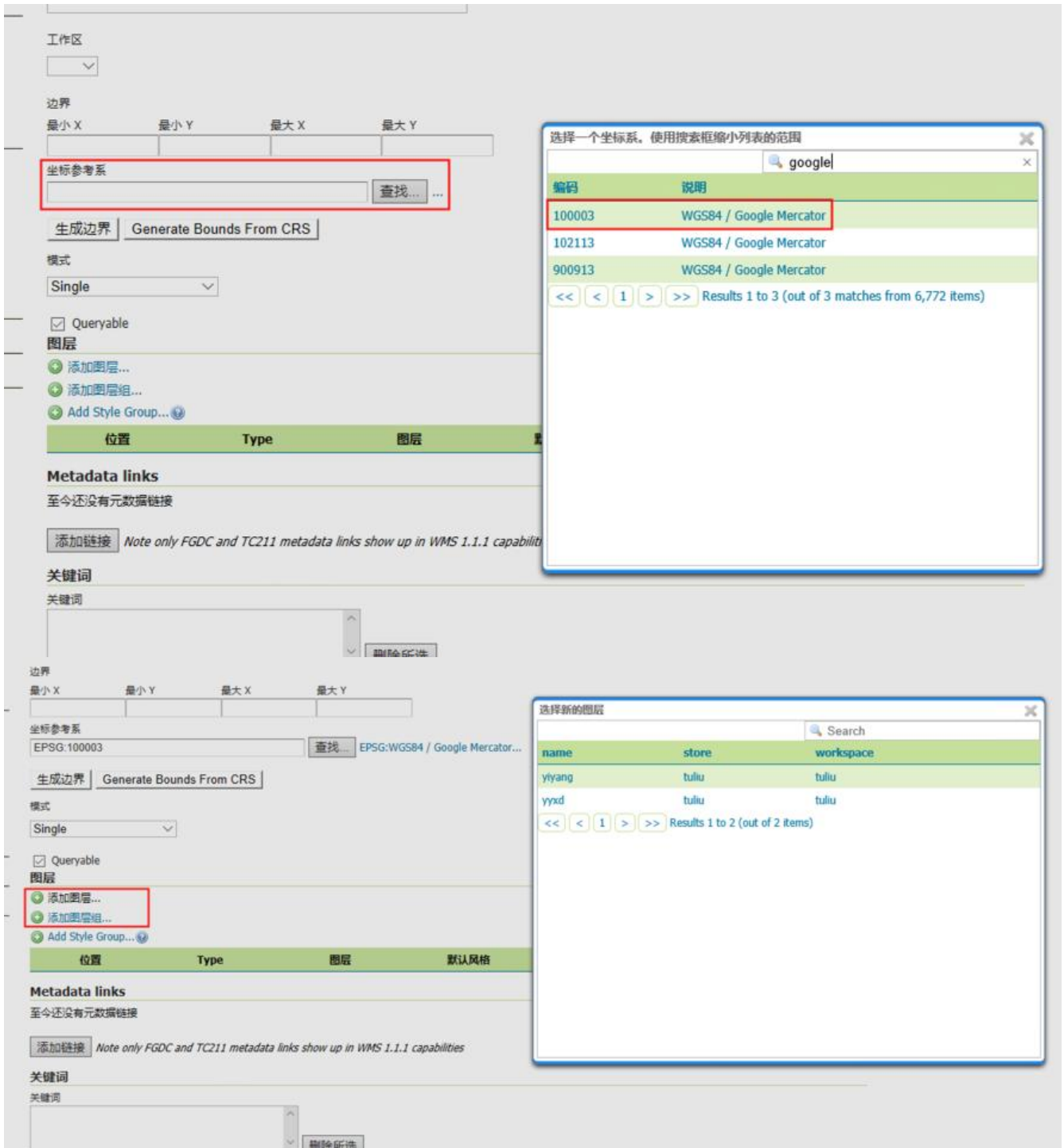
Scale = 1 : 17

Click on the map to get feature info

111.76840, 29.06985

然后我们就可以配置图层组了





工作区
tuliu

边界
最小 X 最小 Y 最大 X 最大 Y
110.71888732910 | 27.982673645019 | 112.93085479736 | 29.524110794067

坐标参考系
EPSG:900913 查找... EPSG:WGS84 / Google Mercator...

生成边界 Generate Bounds From CRS

模式
Single

Queryable

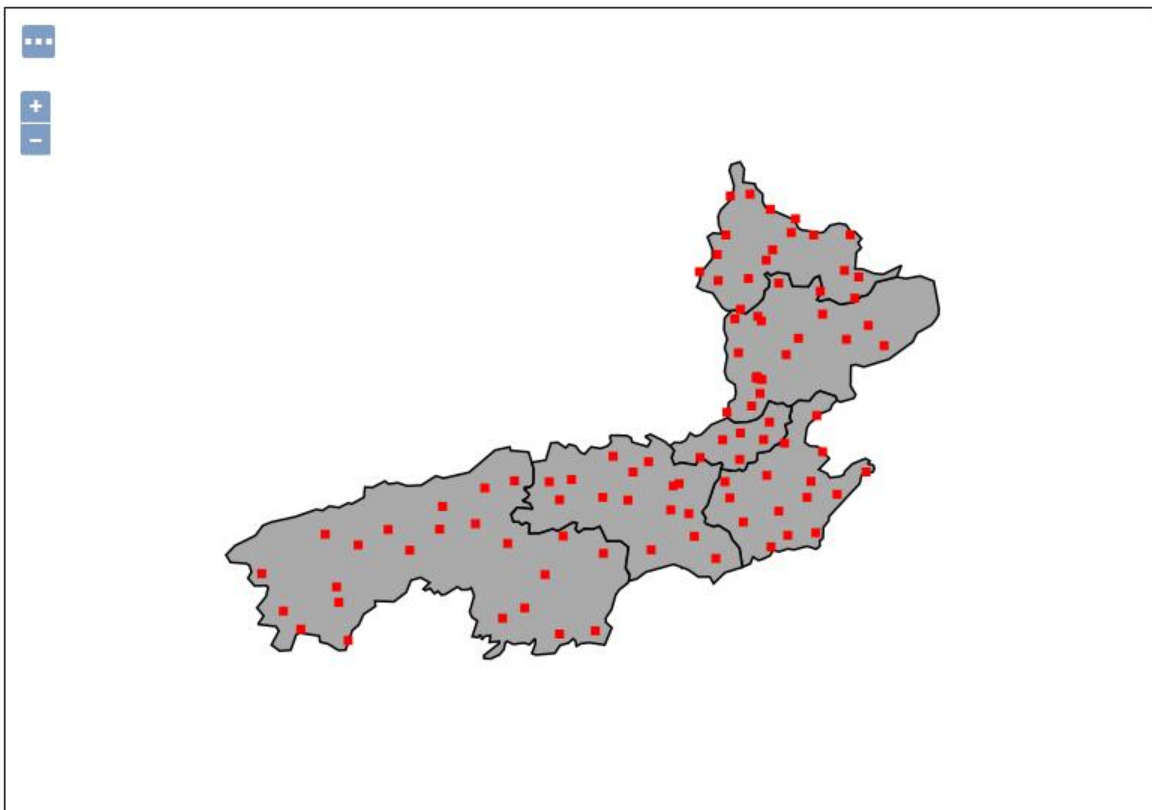
图层

- 添加图层...
- 添加图层组...
- Add Style Group...

位置	Type	图层	默认风格	风格	删除
1 ↓	Layer	tuliu:yyang	<input type="checkbox"/>	polygon	⊖
2 ↑	Layer	tuliu:yyxd	<input type="checkbox"/>	point	⊖

Metadata links
至今还没有元数据链接

完成后即可保存查看预览效果



Scale = 1 : 17

Click on the map to get feature info

[windows相关环境及文件](#) 提取码: arne