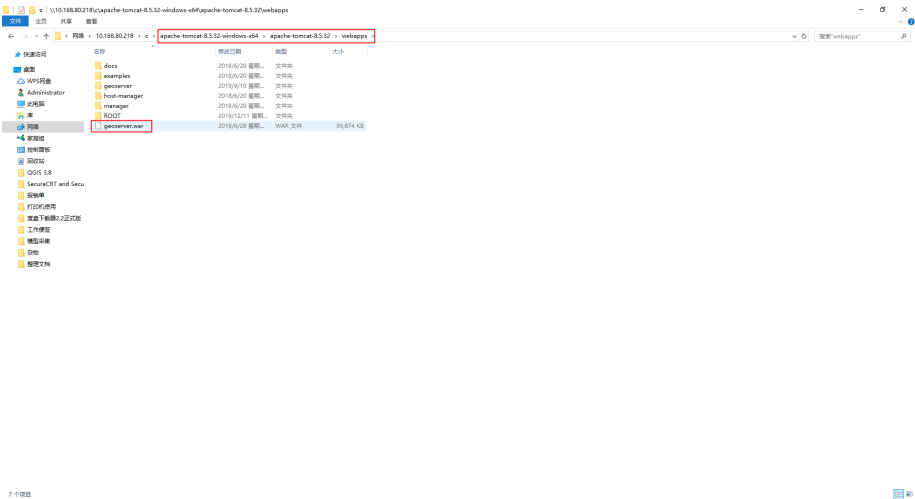


geoserver部署流程

一、部署geoserver

1.准备geoserver.war文件,放入tomcat目录下



2.如果访问地图跨域,需要先解决跨域问题

(1) 下载cors-filter-1.7.jar, Java-property-utils-1.9.jar两个库文件

下载后, 将其放到<Geoserver>\webapps\geoserver\WEB-INF\lib目录下。

下载地址: <http://files.cnblogs.com/files/ytwy/jar.rar>

(2) filter设置

打开<Geoserver>\webapps\geoserver\web.xml文件, 找到文件中<filter>平级的位置, 添加如下内容:

```
1 <filter-name>CORS</filter-name>
2 <filter-class>com.thetransactioncompany.cors.CORSFilter</filter-class>
3 <init-param>
4   <param-name>cors.allowOrigin</param-name>
5   <param-value>*</param-value>
6 </init-param>
7 <init-param>
8   <param-name>cors.supportedMethods</param-name>
9   <param-value>GET, POST, HEAD, PUT, DELETE</param-value>
10 </init-param>
11 <init-param>
12   <param-name>cors.supportedHeaders</param-name>
13   <param-value>Accept, Origin, X-Requested-With, Content-Type, Last-Modified</param-value>
14 </init-param>
15 <init-param>
16   <param-name>cors.exposedHeaders</param-name>
17   <param-value>Set-Cookie</param-value>
18 </init-param>
19 <init-param>
20   <param-name>cors.supportsCredentials</param-name>
21   <param-value>true</param-value>
22 </init-param>
```

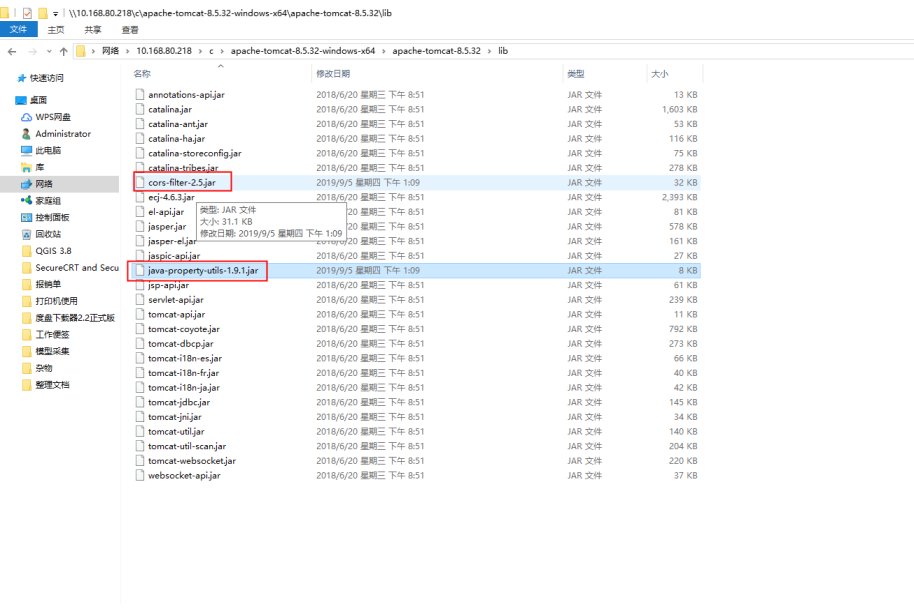
(3) filter-mapping设置

找到文件中<filter-mapping>平级的位置, 添加如下内容:

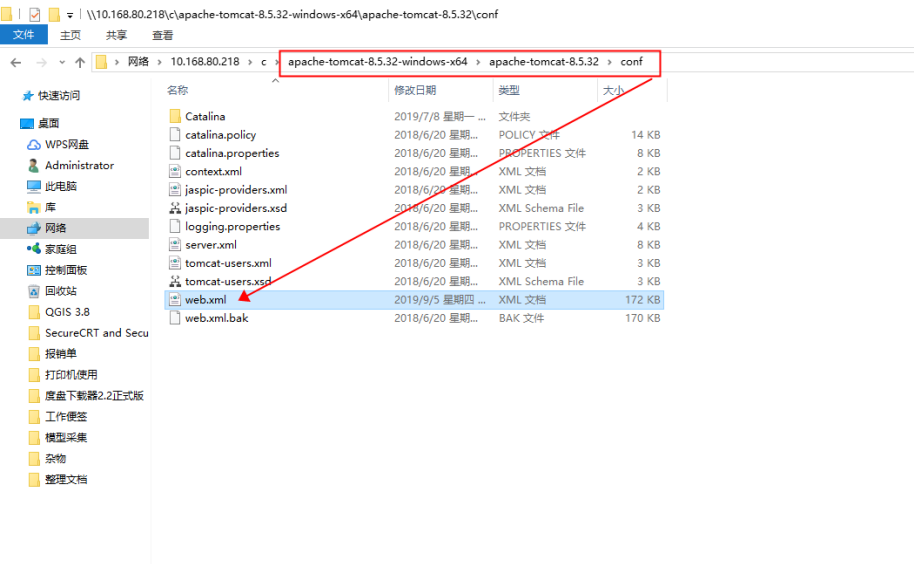
```
1 <filter-mapping>
2   <filter-name>CORS</filter-name>
3   <url-pattern>*</url-pattern>
4 </filter-mapping>
```

或者直接解决tomcat跨域问题

首先下载cors-filter-1.7.jar， java-property-utils-1.9.jar包放入目录下



修改web.xml文件



添加如下内容:

```
<filter>
    <filter-name>CORS</filter-name>
    <filter-
class>com.thetransactioncompany.cors.CORSFilter</filter-
class>
    <init-param>
        <param-name>cors.allowOrigin</param-name>
        <param-value>*</param-value>
    </init-param>
    <init-param>
        <param-name>cors.supportedMethods</param-name>
        <param-value>GET, POST, HEAD, PUT,
DELETE</param-value>
    </init-param>
    <init-param>
```

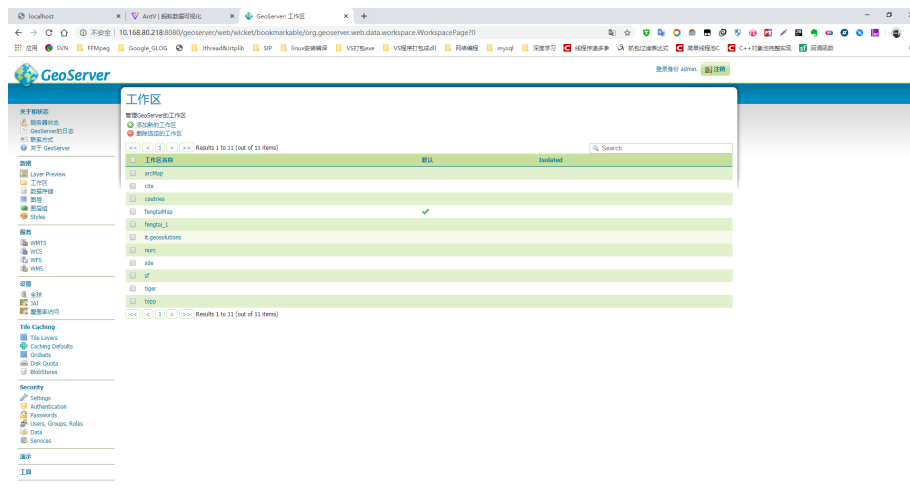
```

<param-name>cors.supportedHeaders</param-name>
  <param-value>Accept, Origin, X-Requested-With,
Content-Type, Last-Modified</param-value>
</init-param>
<init-param>
  <param-name>cors.exposedHeaders</param-name>
  <param-value>Set-Cookie</param-value>
</init-param>
<init-param>
  <param-name>cors.supportsCredentials</param-
name>
  <param-value>true</param-value>
</init-param>
</filter>

<filter-mapping>
  <filter-name>CORS</filter-name>
  <url-pattern>/*</url-pattern>
</filter-mapping>

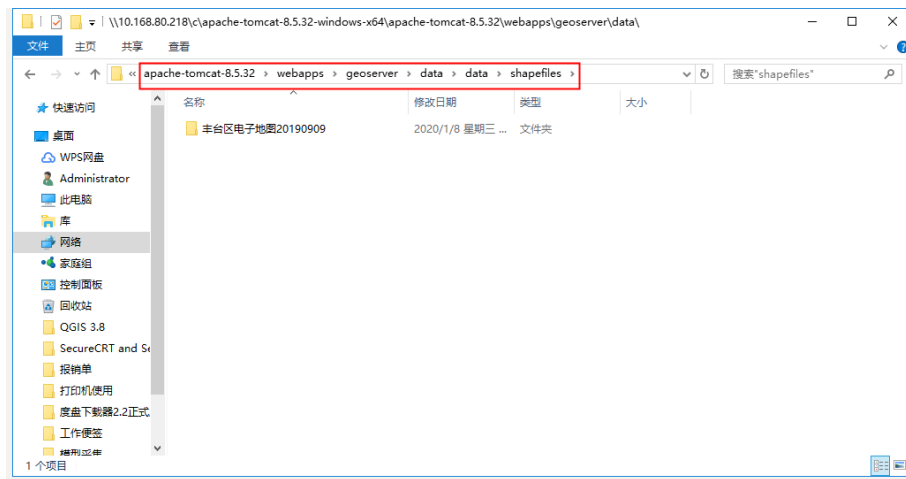
```

3.重启tomcat,访问<http://10.168.80.218:8080/geoserver> (IP:8080/geoserver)



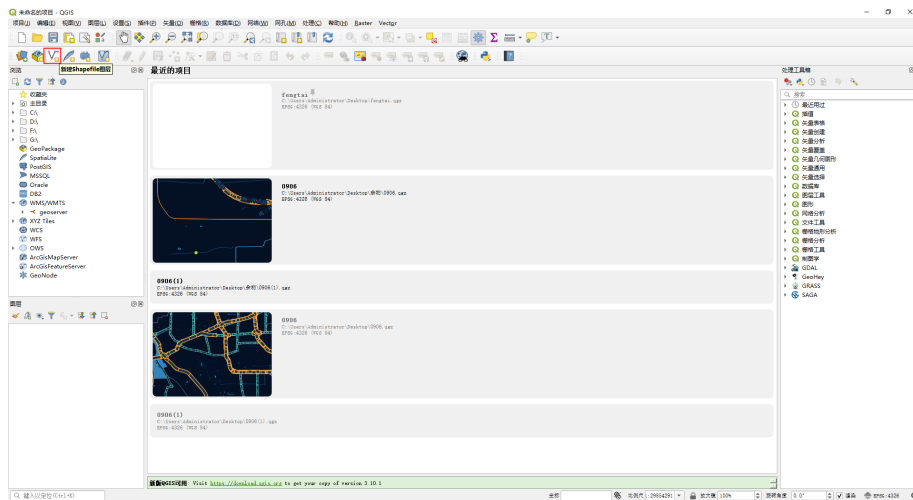
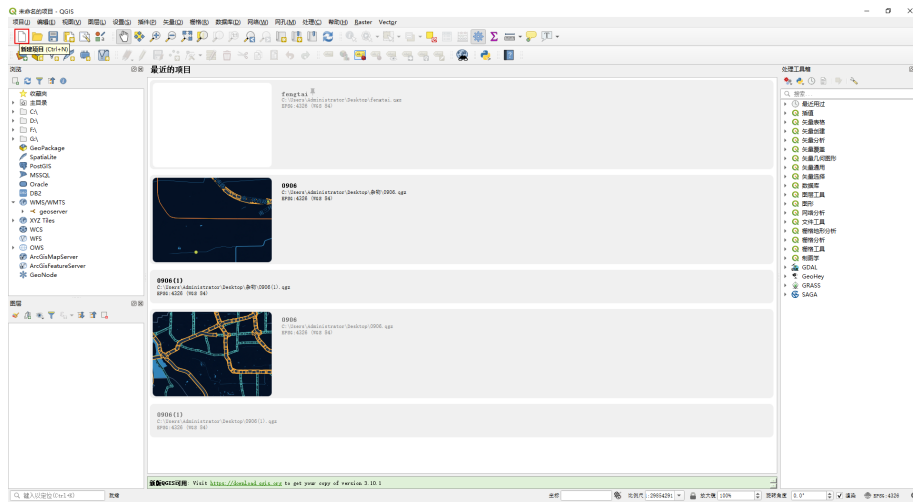
二、发布服务

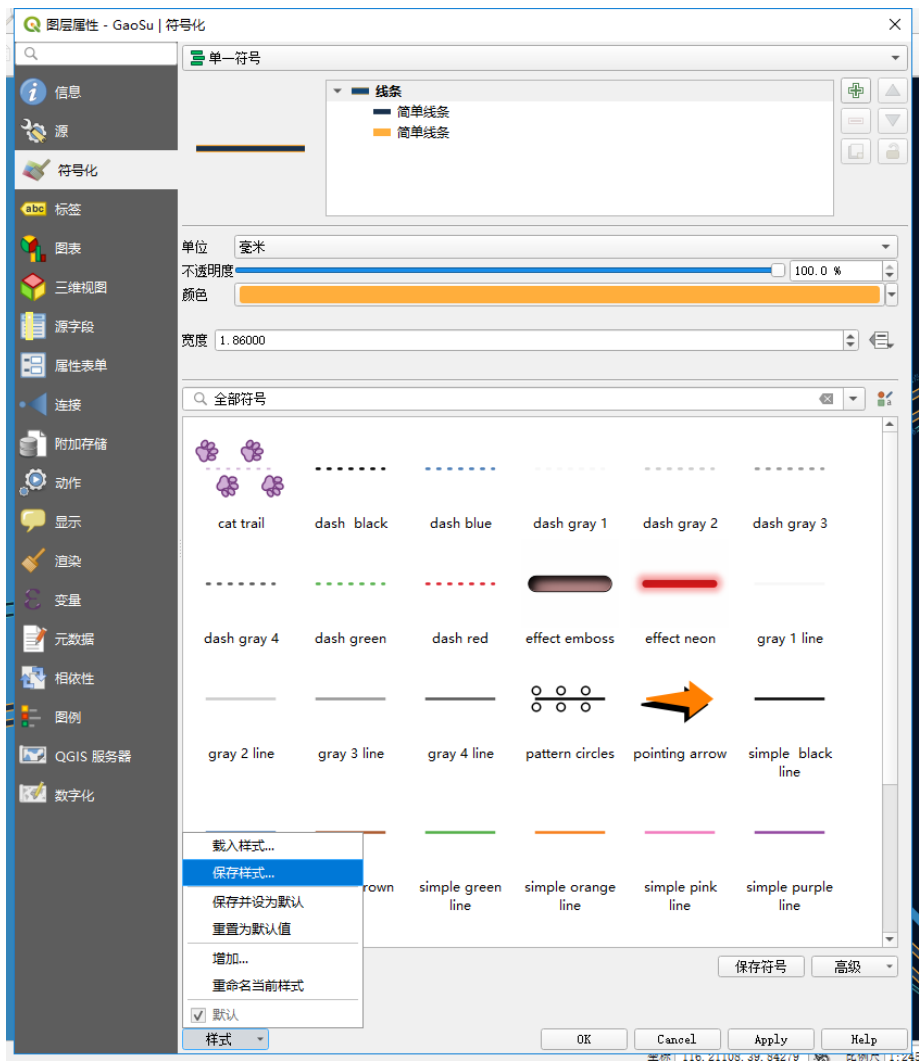
1.将shp源文件放入指定文件夹内

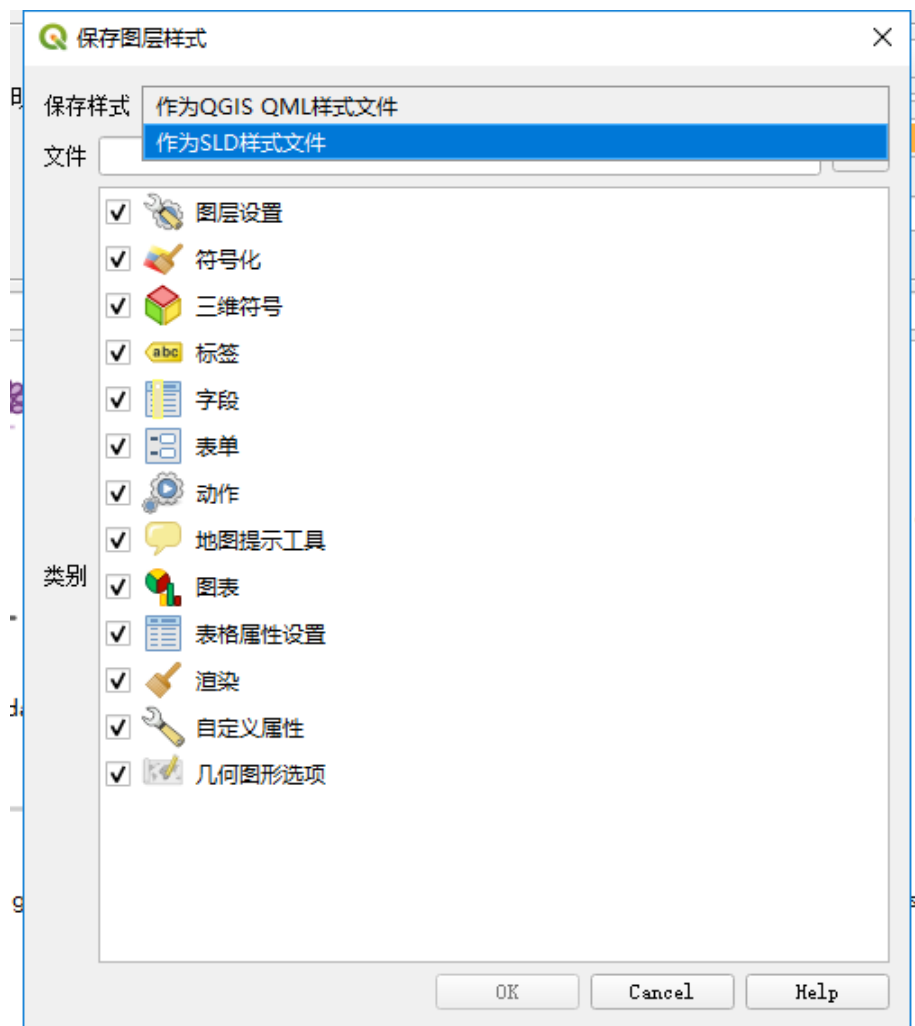


2.设计地图样式并导出sld样式文件(使用QGIS或udig)

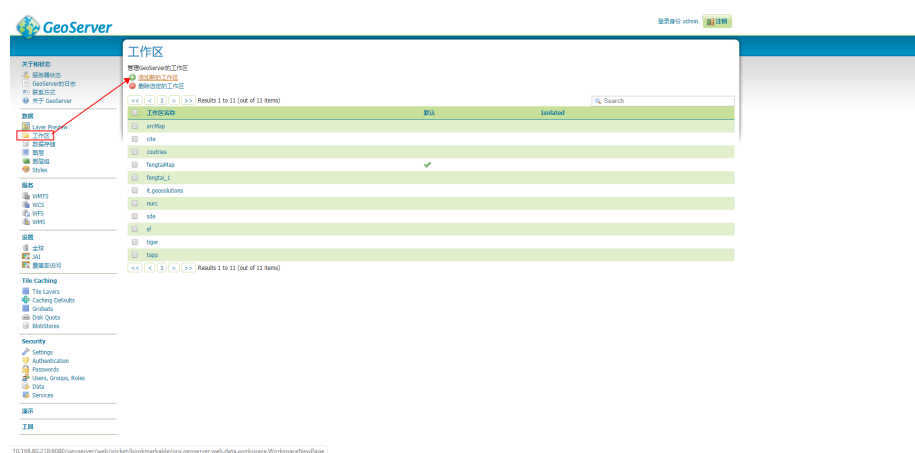
可以新建项目或者导入已经设计好的qgz项目文件



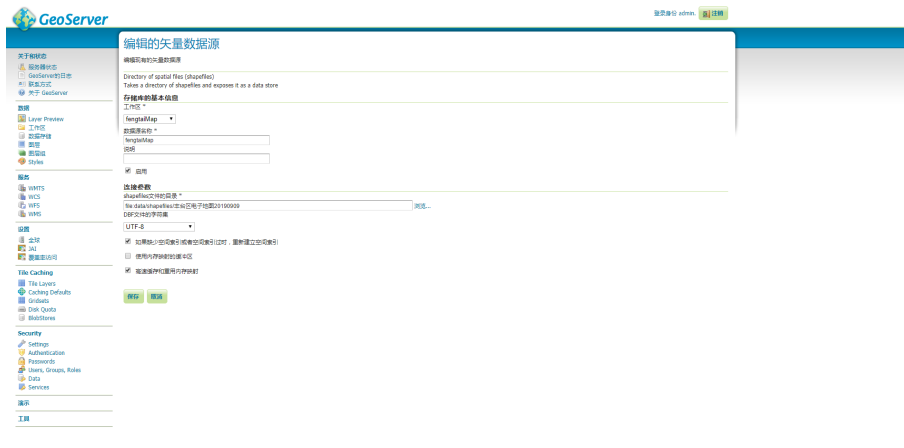




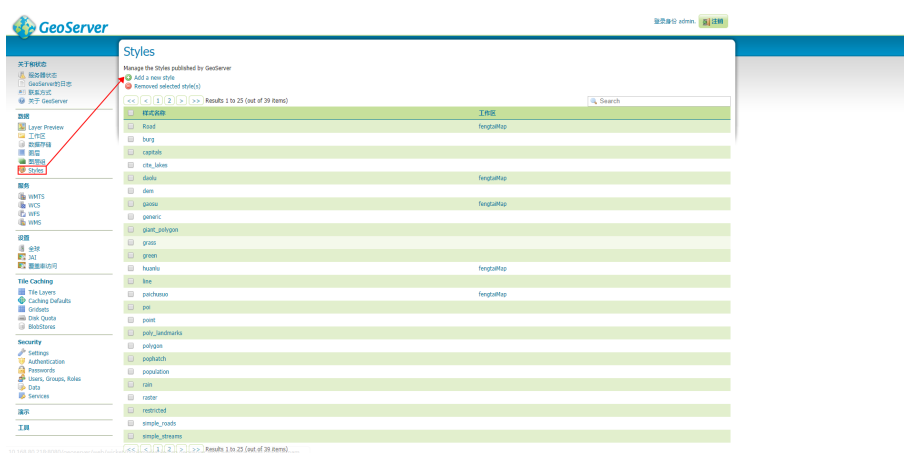
3.新建工作区

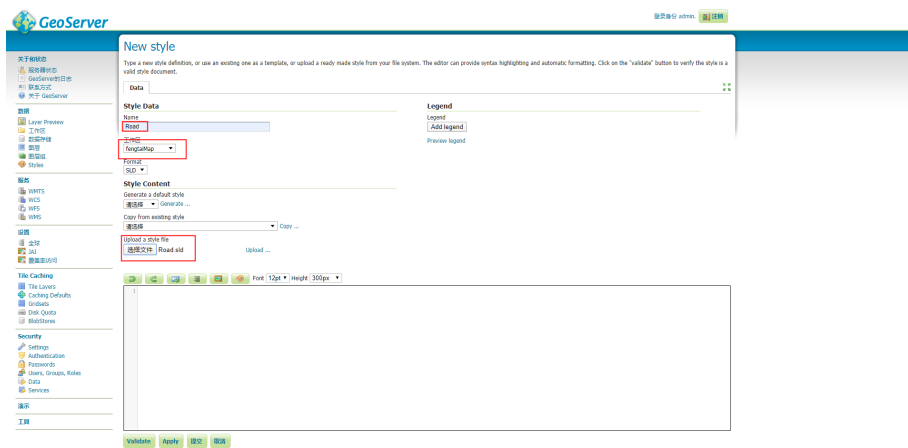


4.新建数据存储

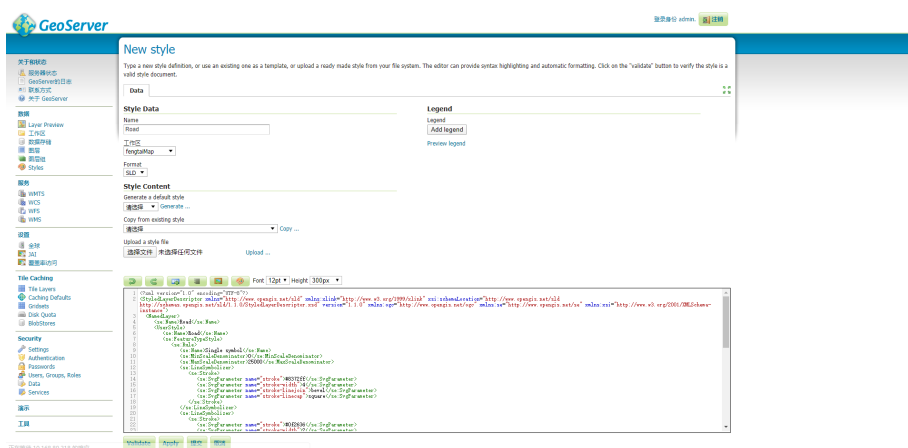


5.新建地图样式





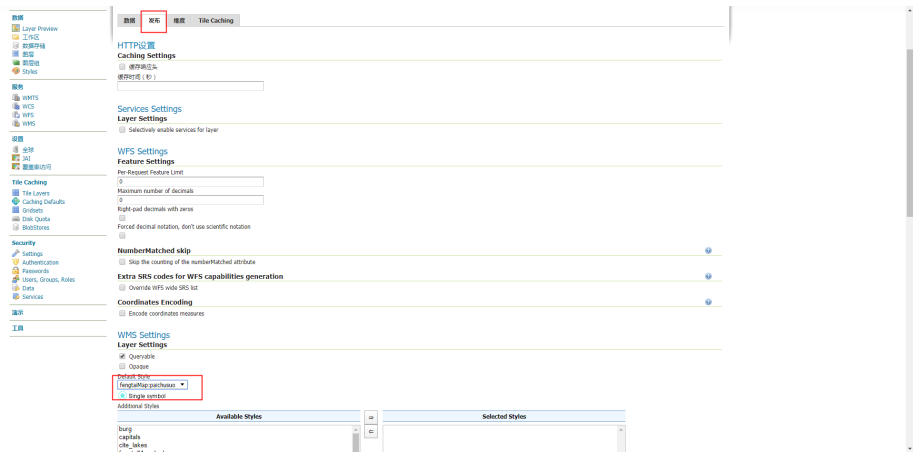
输入样式名称,选择工作区,选择对应的sld文件,点击upload



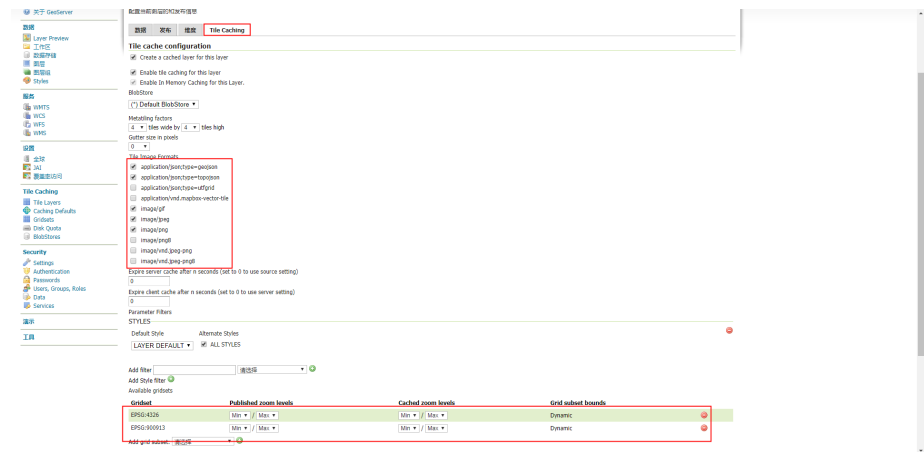
样式上传后会显示在下方,无误后点击提交

6.新建图层





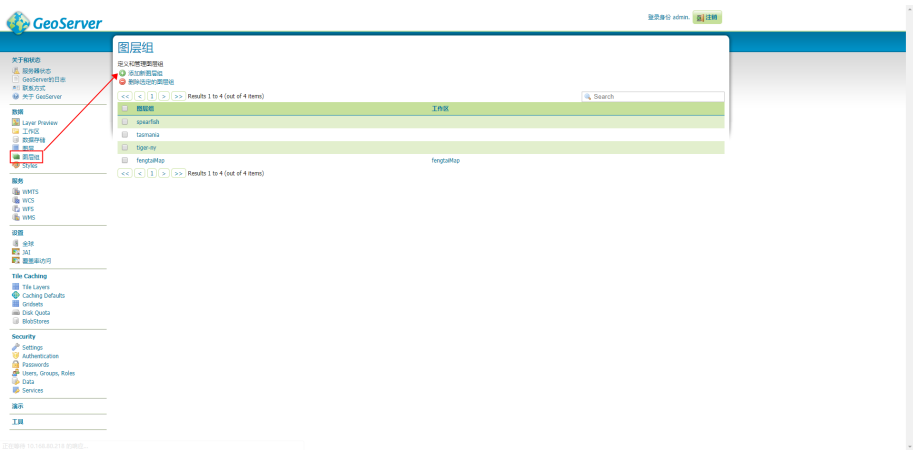
点击发布选项,指定对应的style样式

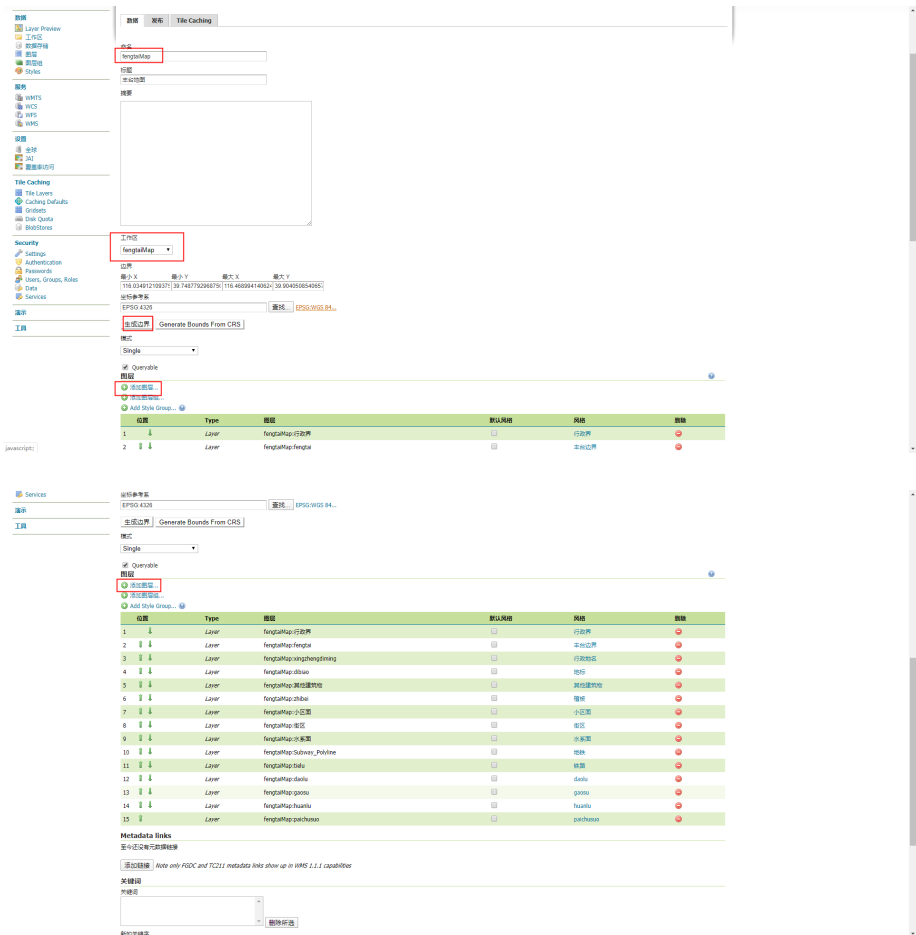


在缓存选项,可以修改发布的瓦片类型(geojson等矢量瓦片需要下载对应的jar包才有缓存选项)可以指定发布的坐标系类型(EPSC:4326即经纬度坐标,EPSC:900913即谷歌墨卡托投影坐标),确认无误后点击保存

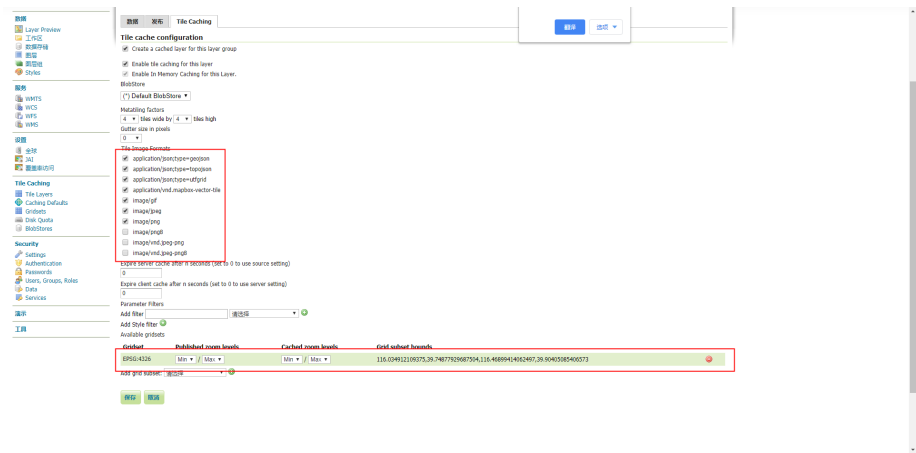
样式和图层是配合使用的,有多少shp源文件就需要发布多少图层,图层对应style样式文件即可。

7.新建图层组



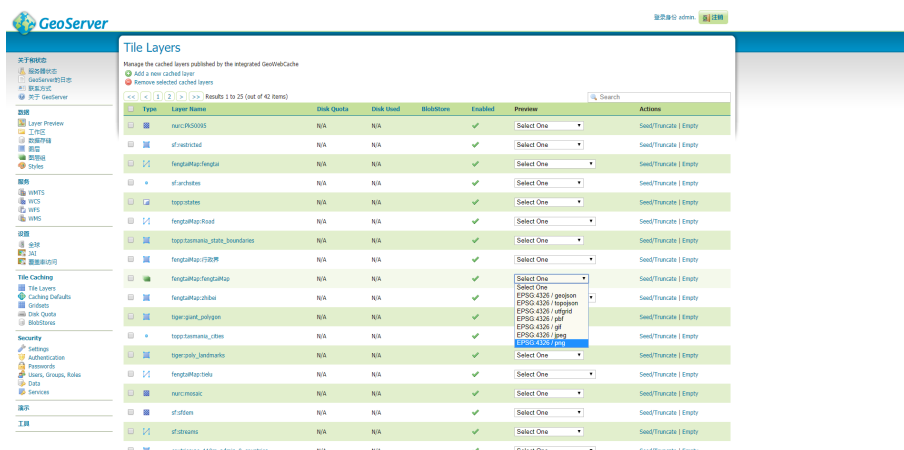
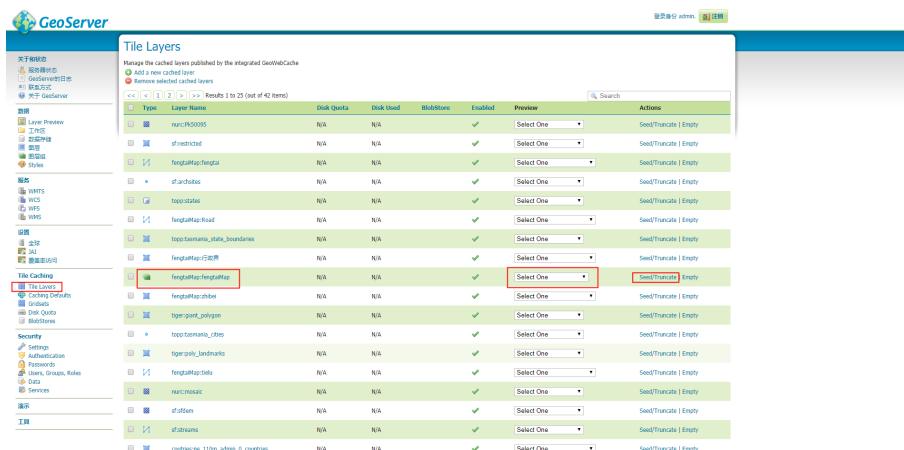


选中工作区后,添加相应的图层,点击生成边界按钮自动生成.

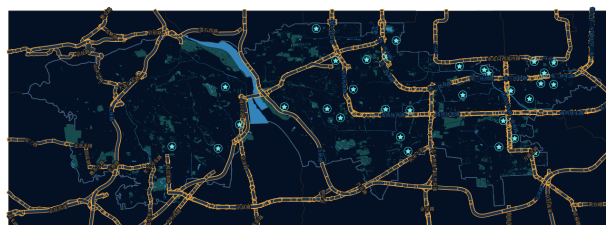


选择可以发布的格式和坐标,点击保存

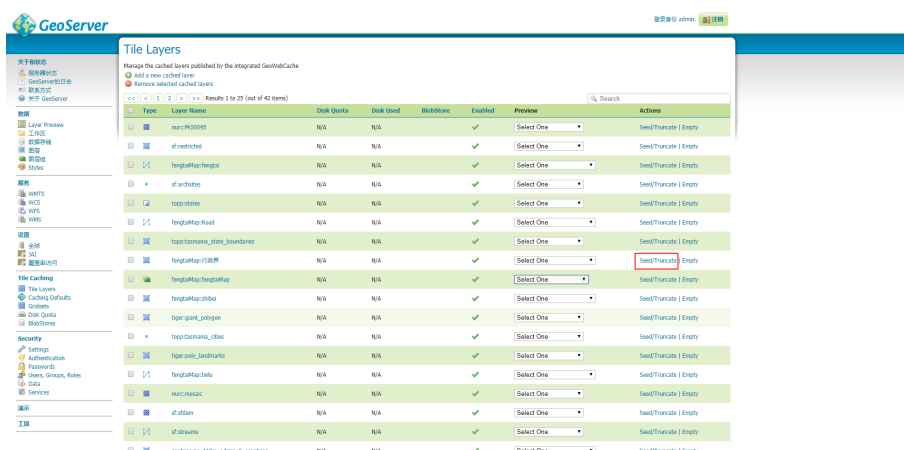
8.查看地图并缓存切片

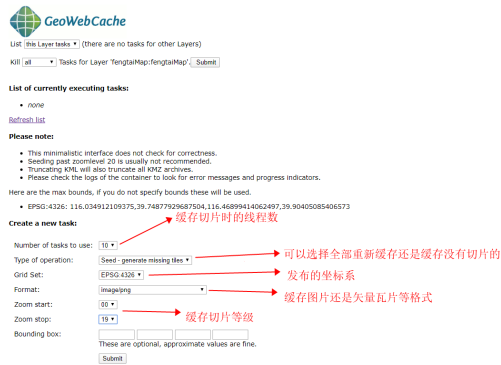


116.18355314941406,39.81651300162354
Scale = 1:136K

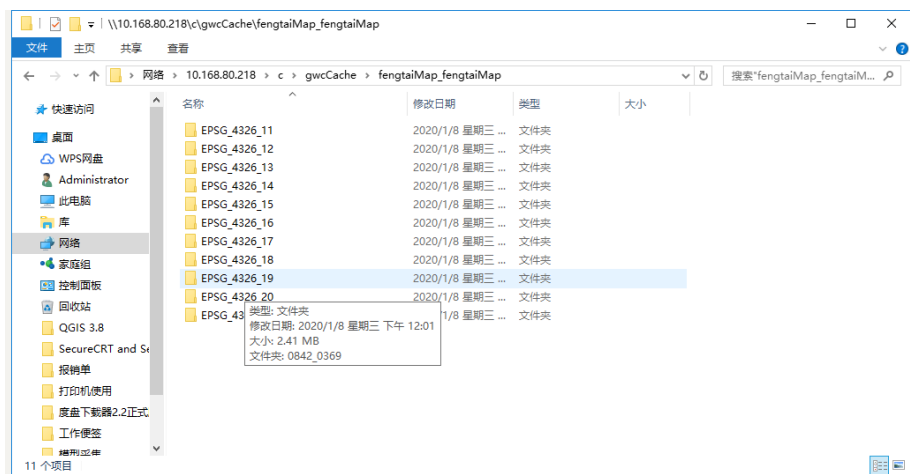
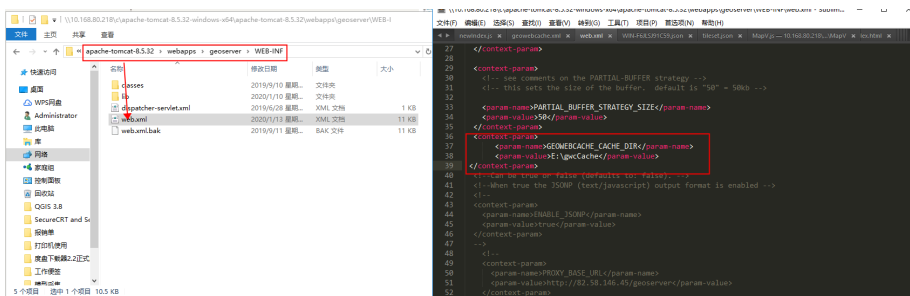


如果出现,说明发布成功,如果一片空白,说明发布的图层中有数据损坏或错误的,需要找到错误的图层并改正(一般空白都是shp源文件引起的,可以在QGIS中转换成shp文件再进行导入)

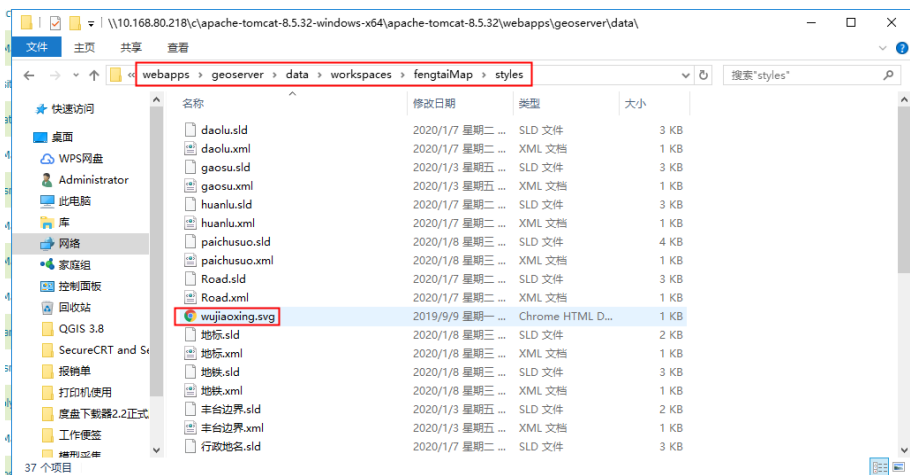




可以修改缓存目录方便找到



geoserver导入SVG图像



The screenshot displays the GeoServer Style Editor for a style named 'paichusuo'. The interface includes a sidebar with navigation options and a main workspace. The 'Style Content' section shows options to generate a default style, copy from an existing style, or upload a style file. The 'Code' tab is active, showing the XML content of the style. The XML includes a title 'paichusuo', a description, and a legend with a red box highlighting the 'fengtailmap:paichusuo' entry.