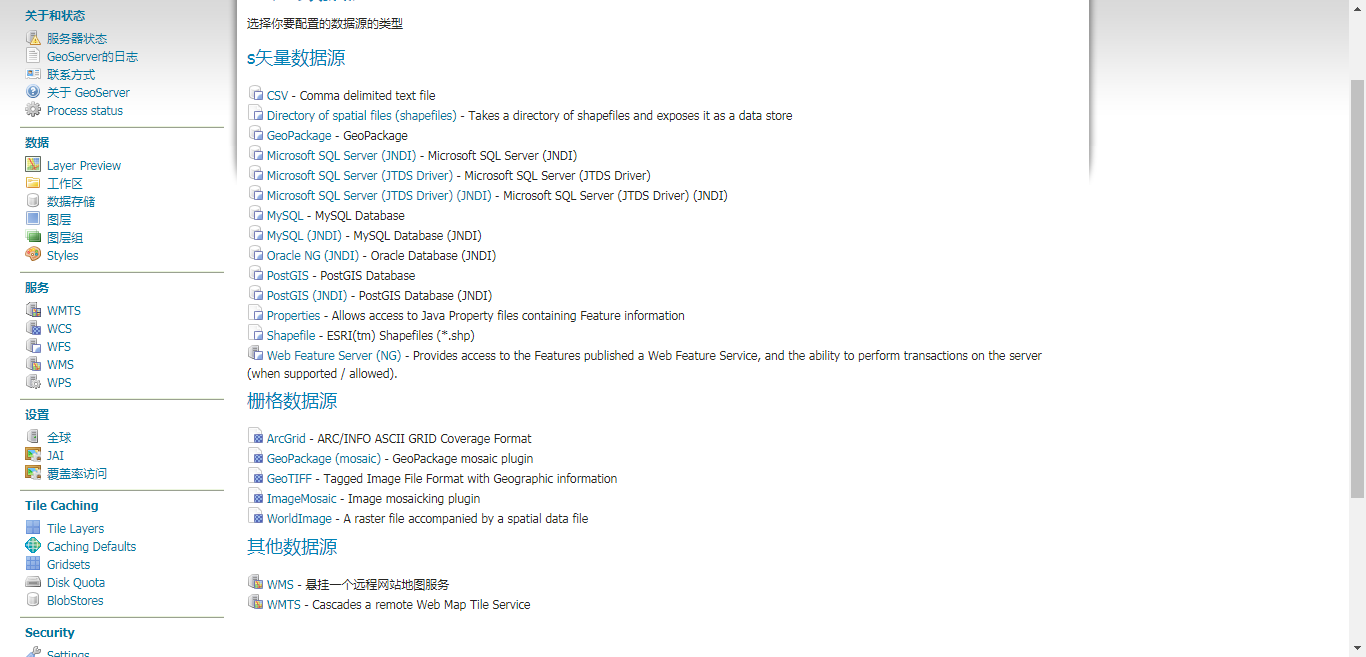
**矢量切片数据存储及管理方案预研**

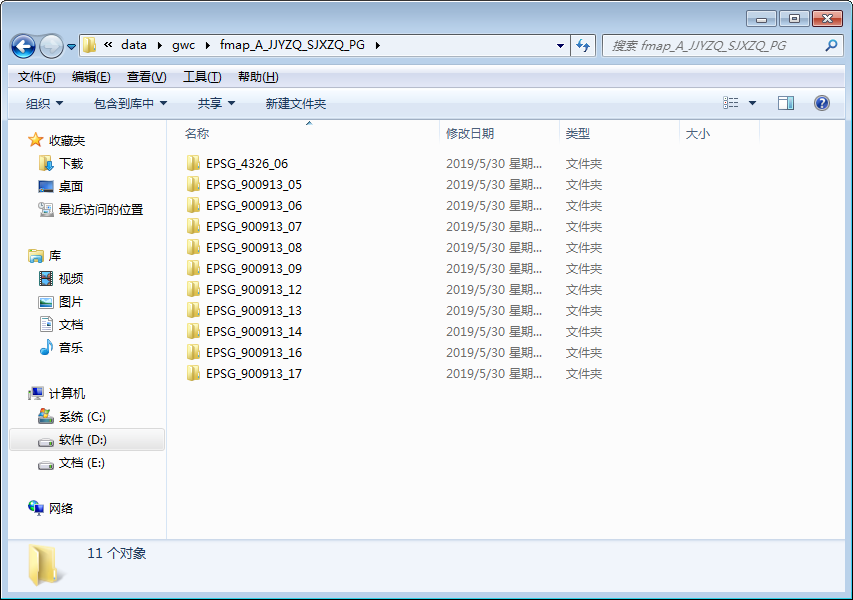
1. Hbase + Tomcat 后台服务 方案
2. 矢量数据生成pbf文件
3. 根据矢量数据类型，导入geoserver

目前geoserver支持多种数据库及不同数据源类型的矢量数据接入，并可以通过图层发布的形式，将数据发布为mapbox标准的矢量切片。



（2）获取pbf文件

图层发布之后，在geoserver数据目录下gwc(GeoWebCache)里会生成一个与图层名字相对应的文件目录，里面就是矢量数据切成的pbf文件。根据地图类型，选择坐标系对应的文件目录。4326选择4326数据目录，3857选择900913数据目录。



1. Pbf文件导入Hbase
2. 配置Hbase地址

原有的HbaseImport类用于存储png图片，初始化方法中从config.properties配置文件读取zookeeper地址，需要提前配置。

1. 定义存储数据命名规范

Geoserver切出来的Pbf数据路径类似于EPSG\_900913\_14\052\_036\013346\_009274.pbf

EPSG\_900913是坐标系

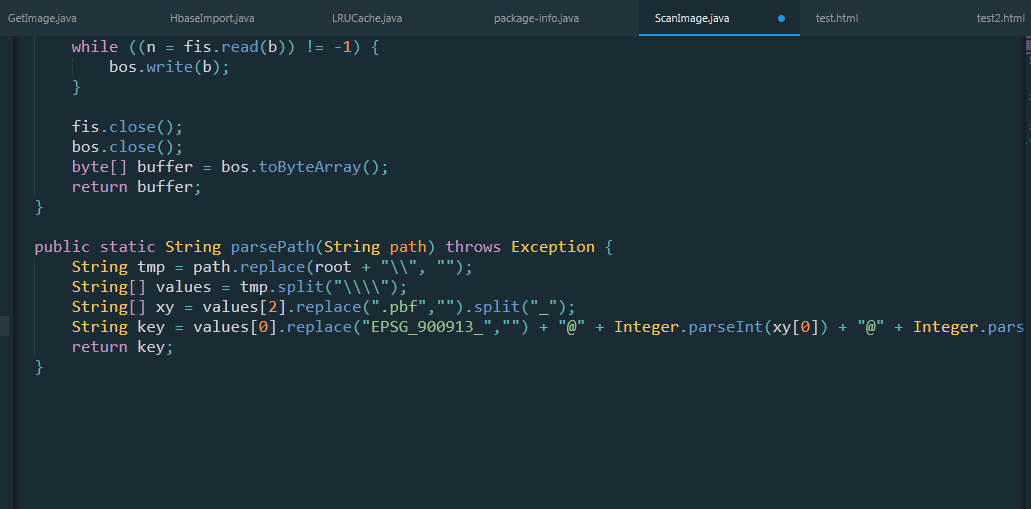
14是层级z

013346是x

009274是y

Mapbox在请求矢量切片的时候，是根据实际数字请求的，不会有补位，所以在存储的时候需要将坐标系和补位0去掉。

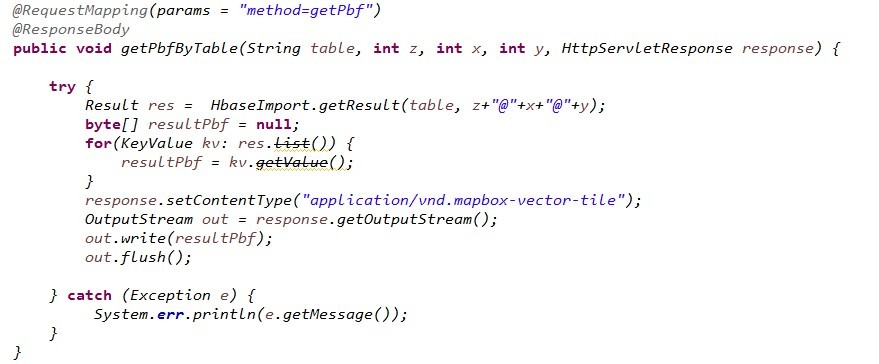
原程序通过ScanImage类读取指定目录下的所有图片文件，存入Hbase，现在我们将其改为读取pbf文件，并按照规则命名。下图为具体修改方法parsePath



1. 修改执行程序存入数据

指定文件根目录，指定存入的表名，执行程序

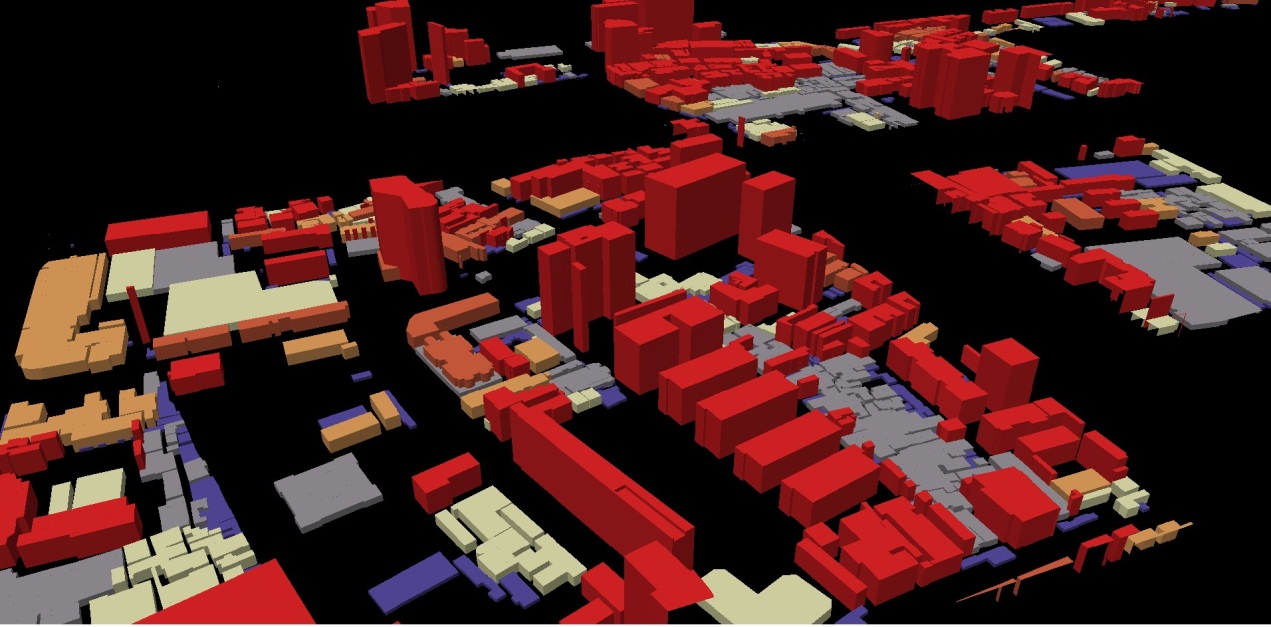
1. Mapbox调用服务
2. 编写调用接口



1. Mapbox调用



1. Mapbox效果

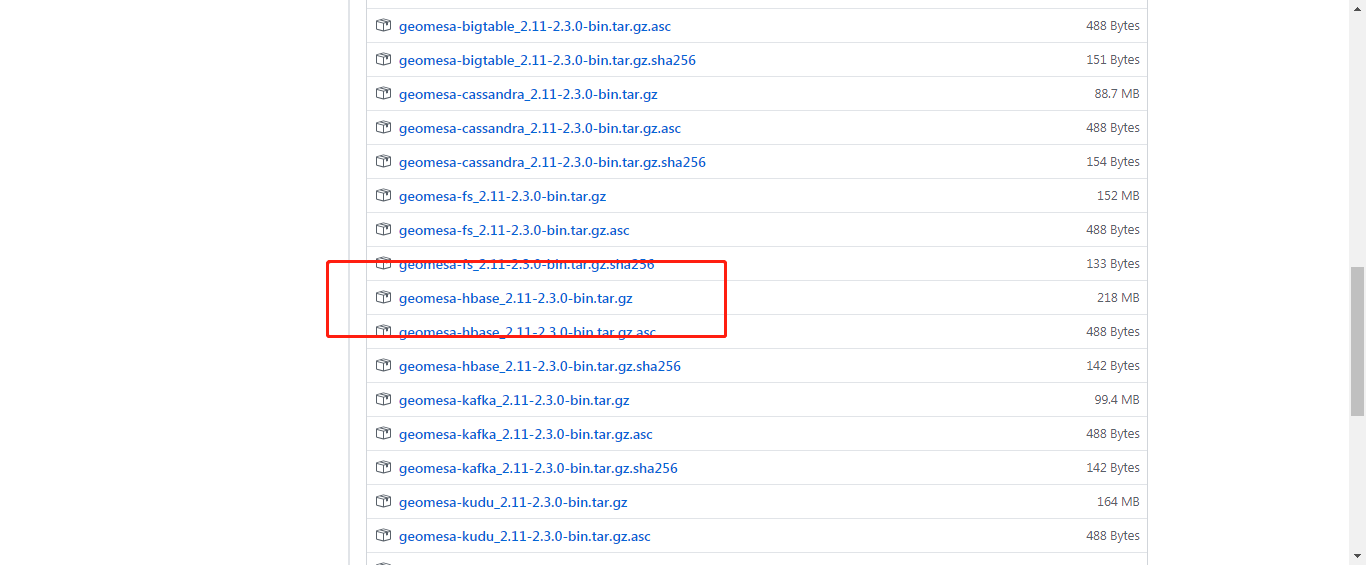


1. Hbase + geomesa + geoserver 方案

因为hbase有单机版和集群版，所以两个版本安装geomesa的方式不一样，这里介绍的是集群版的部署方法

1. 在Cloudera CDH环境下部署geomesa
2. 下载安装geomesa

到GitHub上面下载geomesa-hbase连接工具的压缩包，解压到服务器上。在下面的步骤里，GEOMESA\_HBASE\_HOME代表着geomesa-hbase\_2.11-$VERSION/ 路径



1. 配置geomesa运行需要用到的变量

修改 $GEOMESA\_HBASE\_HOME/conf/geomesa-env.sh 添加下列变量

setvar HADOOP\_HOME /opt/cloudera/parcels/**CDH**/lib/hadoop

setvar HADOOP\_CONF\_DIR /etc/hadoop/conf

hadoop**CDH**="1"

setvar HADOOP\_COMMON\_HOME /opt/cloudera/parcels/CDH/lib/hadoop

setvar HADOOP\_HDFS\_HOME /opt/cloudera/parcels/CDH/lib/hadoop-hdfs

setvar YARN\_HOME /opt/cloudera/parcels/CDH/lib/hadoop-yarn

setvar HADOOP\_MAPRED\_HOME /opt/cloudera/parcels/CDH/lib/hadoop-mapreduce

setvar ZOOKEEPER\_HOME /opt/cloudera/parcels/CDH/lib/zookeeper

这些路径基本都是默认的位置，如果有修改，需要对应上

1. 配置geomesa-hbase分布式运行环境

复制$GEOMESA\_HBASE\_HOME/dist/hbase/geomesa-hbase-distributed-runtime\_2.11-$VERSION.jar 到hbase的运行依赖库里。

配置geomesa-site.xml。在$GEOMESA\_HBASE\_HOME/conf目录下有一个geomesa-site.xml.template模板，可以直接修改里面的内容。如下图所示，把geomesa.hbase.coprocessor.path属性值填上刚才的jar包路径，其中[name\_node]

要改为hbase的节点名称。

<?xml version="1.0" encoding="UTF-8"?>

<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

**<configuration>**

**<property>**

**<name>**geomesa.hbase.coprocessor.path**</name>**

**<value>**hdfs://[name\_node]:8020/hbase/lib/geomesa-hbase-distributed-runtime\_2.11-2.0.0.jar**</value>**

**<description>**HDFS or local path to GeoMesa-HBase Coprocessor JAR. If a local path is provided it must be

the same for all region servers. A path provided through the DataStore parameters will always

override this property.

**</description>**

**<final>**false**</final>**

**</property>**

**</configuration>**

修改完成后将文件另存为geomesa-site.xml，放在 $GEOMESA\_HBASE\_HOME/conf

目录下。

执行下面的命令，在hbase配置目录下创建hbase-site.xml文件的软链接。

ln -s /etc/hbase/conf.cloudera.hbase/hbase-site.xml $GEOMESA\_HBASE\_HOME/conf/hbase-site.xml

1. 添加依赖包

由于许可限制，需要执行$GEOMESA\_HBASE\_HOME/bin/install-jai.sh和$GEOMESA\_HBASE\_HOME/bin/install-jline.sh，将支持shapefile文件的依赖包下载到$GEOMESA\_HBASE\_HOME/lib目录下。

修改$GEOMESA\_HBASE\_HOME/bin/install-hbase.sh脚本，将hbase\_version

改为1.2.0，然后执行这个脚本，将所用到的jar包下载到$GEOMESA\_HBASE\_HOME/lib目录下。如果目录里原来有低版本的包，要将其删掉。

执行下面两个命令，在hbase库里创建对应依赖jar包的软链接

ln -s /opt/cloudera/parcels/**CDH**/lib/hbase/lib/metrics-core-2.2.0.jar **\**

$GEOMESA\_HBASE\_HOME/lib/metrics-core-2.2.0.jar;

ln -s /opt/cloudera/parcels/**CDH**/lib/hbase/lib/htrace-core-3.2.0-incubating.jar **\**

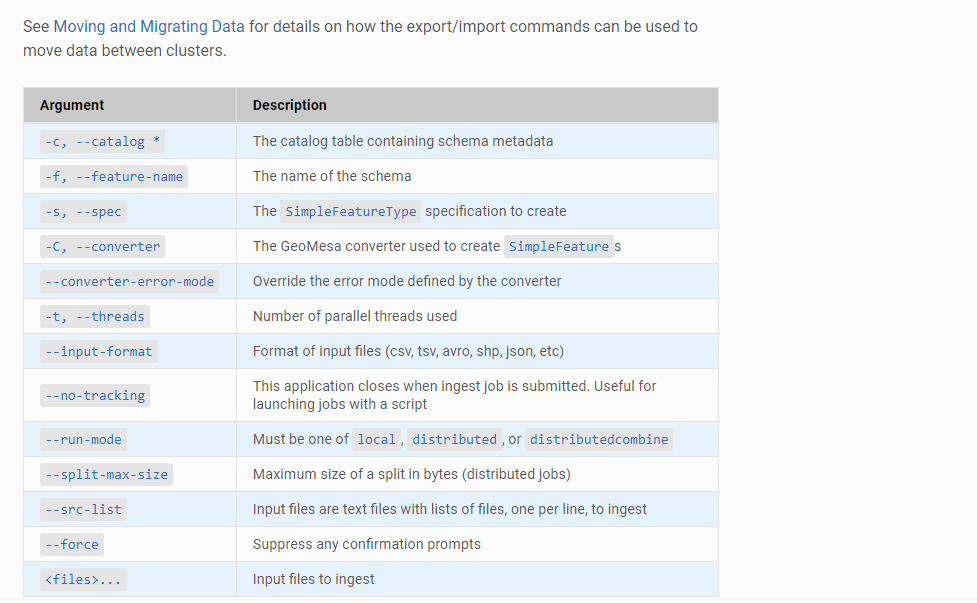
$GEOMESA\_HBASE\_HOME/lib/htrace-core-3.2.0-incubating.jar;

1. 导入数据

进入$GEOMESA\_HBASE\_HOME，执行

bin/geomesa-hbase ingest –catalog 表名 –feature-name 要素名 –-input-format shp “/opt/tools/A\_JMDJSS\_JMD\_PG.shp”

在这过程中geomesa会自动读取shapefile里的所有字段，并生成对应的格式转换文件，通常只需要默认就可以了，然后它会转换成所认识的数据格式，存入hbase中。下面是导入命令ingest的一些参数及说明



1. 在geoserver包里添加geomesa插件及相关依赖

这里的geoserver是指打包好的geoserver.war

1. 安装geomesa插件

$GEOMESA\_HBASE\_HOME/dist/gs-plugins目录下有一个压缩包和一个jar包，将压缩包解压出来可以得到插件的jar包，把这两个jar包放到geoserver\WEB-INF\lib目录下。

修改$GEOMESA\_HBASE\_HOME/bin/install-hadoop.sh文件，改为对应的hadoop和zookeeper版本，执行该文件，把依赖包下载到本地并放到geoserver\WEB-INF\lib目录下。这里建议到hadoop依赖目录下去找对应的hadoop和zookeeper相关依赖包，更为稳妥。

如果geoserver里已经有同样的jar包，选择版本更高的。

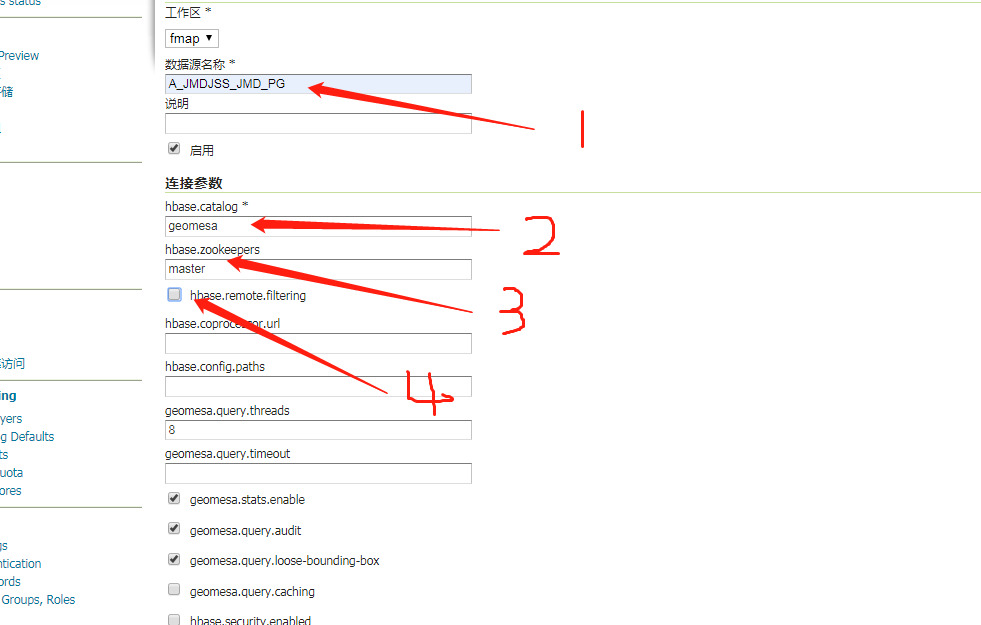
Geomesa依赖库里的netty用的可能还是3.x版本，可以把这里下载的新版本netty-all-${netty4\_version}.jar放到Geomesa依赖库里。

1. 通过geoserver连接hbase

A、新建数据源的时候选择hbase



B、填入hbase对应的参数，读取数据表



1是自定义的数据源名称

2是存入hbase的表名

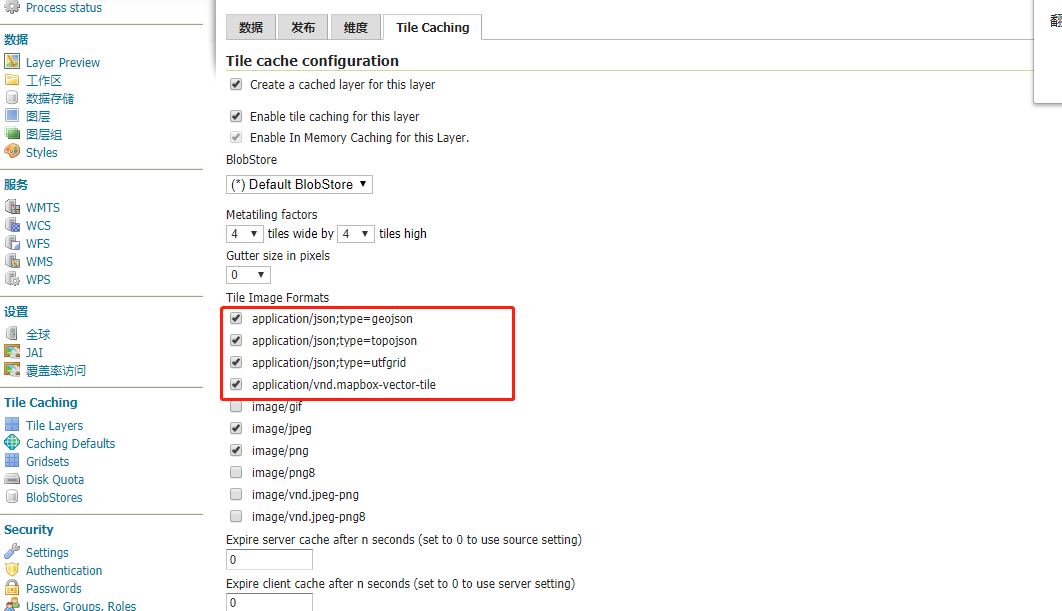
3是zookeeper节点

4是hbase请求过滤器，不勾选

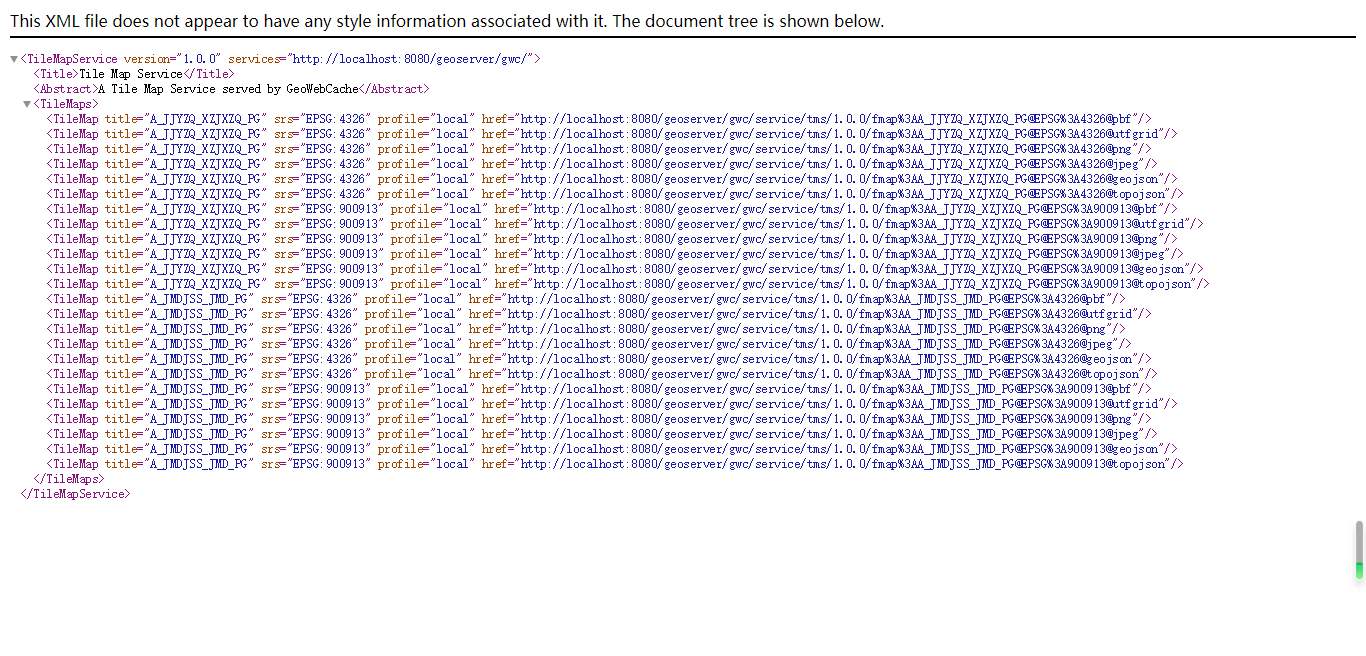
保存之后，就可以使用这个数据源了。

（3） 在建好的数据源下选择图层发布。

（4） 发布时跟普通数据源一样，选择对应坐标系，计算数据范围，发布类型记得选上vector tiles



（5） 可以在tms查看可用的服务



（6）mapbox调用效果

