**geoinfo微服务架构方案设计**

目录

[geoinfo微服务总述 1](#_Toc485300204)

[1.1 geoinfo概述 2](#_Toc485300205)

1.2 [geoinfo目标定位 2](#_Toc485300206)

[geoinfo微服务 3](#_Toc485300207)

[2.1 geoinfo服务架构 3](#_Toc485300208)

[2.2 geoinfo功能梳理 4](#_Toc485300209)

[2.3 geoinfo业务发布流程 8](#_Toc485300210)

[2.4 geoinfo安装教程 8](#_Toc485300211)

[2.5 geoinfo 部署脚本 8](#_Toc485300211)

[2.6 geoinfo 数据库 8](#_Toc485300211)

# geoinfo微服务总述

## geoinfo概述

Geoinfo中文是地理信息服务：通过坐标搜索某个城市周围的建筑/道路/树木等物件，提供给客户端渲染。

## geoinfo目标定位

为3d世界客户端提供地理查询和寻路功能服务。

# geoinfo微服务

## 2.1 geoinfo服务架构



## 2.2 geoinfo功能梳理

|  |  |  |
| --- | --- | --- |
| 功能名称 | API | 参数 |
| 根据经纬度坐标计算u3d坐标 | [/v1/geoinfoProtobuf/u3dCoord](http://192.168.31.253:9305/swagger-ui.html" \l "!/(30512)%E6%99%AF%E5%92%8C%E5%9C%B0%E7%90%86%E4%BF%A1%E6%81%AF(%E5%9F%8E%E5%B8%82%E6%B8%B2%E6%9F%93)Protobuf/u3dCoordUsingPOST_1) | {  "cityCode": "0280",  "latitude": 1.32,  "longitude": 2.1  } |
| 获取街道,建筑,组件,墙,街道信息 | /v1/geoinfoProtobuf/info | {"center": {"X": 0,"Z": 0},"cityCode": "0280","info": ["buildings"],"scope": 100} |
| 城市边界查询 | /v1/geoinfoProtobuf/bounds | {  "cityCode": "0280"  } |
| 按不同范围分别获取街道,建筑,组件,墙,街道,BLOCK等信息(备注：当前接口和info接口唯一不同在scope 这个允许填写不同范围，但是要保证和info一一对应) | /v1/geoinfoProtobuf/infowithvariousscopes | {"center": {"X": 0,"Z": 0},"cityCode": "0280","info": ["buildings",“walls”],"scope": 100,200 **}** |
| 计算最近街道投影U3D坐标及该街道信息 | /v1/geoinfoProtobuf/neareststreet/u3dcoord | {  "cityCode": "0280",  "latitude": 1.32,  "longitude": 2.1  } |
| 根据街道顶点坐标获取顶点为该坐标的所有街道信息 | /v1/geoinfoProtobuf/streetsbyvertex | {  "cityCode": "0280",  "point": {  "X": 0,  "Z": 0  }  } |
| 计算U3D坐标所在瓦块以及周围临近瓦块ID的列表 | /v1/geoinfoProtobuf/tiles | {  "center": {  "X": 0,  "Z": 0  },  "cityCode": "0280"  } |
| 根据瓦块ID的列表获取对应瓦块中的资源 | /v1/geoinfoProtobuf/infobytiles | {  "cityCode": "0280",  "info": [  "buildings"  ],  "tileIds": [  "00\_11"  ]  } |
| 获取一个或多个瓦块的版本 | /v1/geoinfoProtobuf/tilevsn | {  "cityCode": "0280",  "tileIds": [  "00\_11"  ]  } |
| 寻路接口 | /v1/geoinfoProtobuf/route | {  "cityCode": "0280",  "endPoint": {  "X": 0,  "Z": 0  },  "startPoint": {  "X": 0,  "Z": 0  }  } |

参照：http://192.168.31.253:9305/swagger-ui.html#/(30512)景和地理信息(城市渲染)Protobuf

## 2.3 geoinfo业务发布流程

首先、公司编辑器组同事通过他们辛勤工作产出3d城市所需要的建筑/道路/树木等物件数据然后通过他们质检后台上传这些数据到青云服务器

其次、编辑器组同事告知我们最新数据青云服务器地址，以及上线城市信息，我们得到消息之后通过脚本导入青云服务器最新包数据到数据库（postgres）

最后、客户端请求我们geoinfo服务，得到想要的结果信息

## 2.4 geoinfo通信协议

geoinfo返回协议经过多次讨论决定使用protobuf，请求上还是保持对象模式。

## 2.5 geoinfo安装教程

1、easy\_install pip

2、安装青云（图链接）

<https://docs.qingcloud.com/qingstor/command_line_tools/qsctl.html>

3、如果在青云环境需要减特殊文件权限

chattr -i /etc/passwd

chattr -i /etc/shadow

chattr -i /etc/group

chattr -i /etc/gshadow

4、安装postgres

<http://www.postgresonline.com/journal/archives/362-An-almost-idiots-guide-to-install-PostgreSQL-9.5,-PostGIS-2.2-and-pgRouting-2.1.0-with-Yum.html>

1. 开始教程前，需要先建立一个postgres的用户和组。不然initdb的时候会报错。

useradd postgres

1. 需要安装postgresql和postgis,pgrouting\_95，其他的不用.

sudo rpm -ivh http://yum.postgresql.org/9.5/redhat/rhel-7-x86\_64/pgdg-centos95-9.5-2.noarch.rpm

yum install postgresql95 postgresql95-server postgresql95-libs postgresql95-contrib postgresql95-devel

/usr/pgsql-9.5/bin/postgresql95-setup initdb

yum -y install epel-release

sudo yum install postgis2\_95

yum install pgrouting\_95

systemctl enable postgresql-9.5.service

systemctl restart postgresql-9.5.service

1. 修改配置

/var/lib/pgsql/9.5/data/pg\_hba.conf

192.168.31.252 修改为本机内网IP

# TYPE DATABASE USER ADDRESS METHOD

# "local" is for Unix domain socket connections only

local all all trust

# IPv4 local connections:

host all all 127.0.0.1/32 trust

host all all 172.17.0.1/24 trust

host all all 192.168.31.252/32 trust

# IPv6 local connections:

host all all ::1/128 trust

# Allow replication connections from localhost, by a user with the

# replication privilege.

#local replication postgres peer

#host replication postgres 127.0.0.1/32 ident

#host replication postgres ::1/128 ident

/var/lib/pgsql/9.5/data/postgresql.conf

listen\_addresses = '\*' # what IP address(es) to listen on;

重启

systemctl enable postgresql-9.5.service

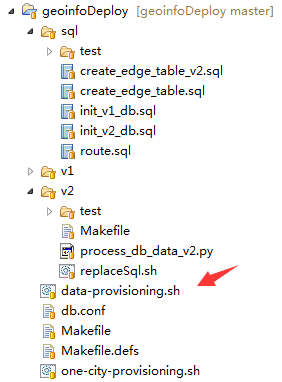
systemctl restart postgresql-9.5.service

5、用命令su - postgres -c "psql" 看是否正常进入

## 2.6 geoinfo 部署脚本

geoinfo脚本主要做的工作是负责将 编辑器产出数据导入到本地252 postgres数据库中，DBA负责将数部署到青云其他环境。

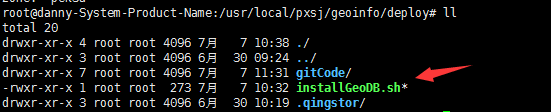
git地址：[http://192.168.31.222:8080/java/geoinfoDeploy.git](http://192.168.31.222:8080/platform/workspace.git)



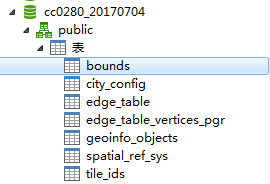
启动脚本 ./data-provisioning.sh

开发环境部署脚本:

192.168.31.252: /usr/local/pxsj/geoinfo/deploy/installGeoDB.sh



**2.7 geoinfo数据库**



1. bounds 城市边界配置表
2. city\_config 城市瓦块配置表
3. edge\_table 路网表
4. edge\_table\_vertices\_pgr 在edge\_table表的source和target列中表现，是一组点的组合
5. geoinfo\_objects 资源原始数据表
6. Spatial\_ref\_sys PostGis 默认表，记录合肥的空间坐标系统
7. tile\_ids 瓦块表, 记录瓦块对应的点击