**上机实验1 地理空间数据库的建立与结构分析**

**【上机目的】**

了解空间数据存储的发展历程，进一步加深对空间数据库的理解，掌握空间数据库的建库技术、熟悉存储空间数据的表结构。

**【上机内容】**

利用空间数据引擎技术（如，ArcSDE、PostGIS等）结合一种常用的关系数据库管理系统（如SQL Server、ProgreSQL等）建立地理空间数据库。对该数据库中的空间数据存储结构进行分析，从而进一步认识空间数据的存储方式。

**【上机要求】**

1. 独立实验，认真、按时完成实验并提交上机实验报告。
2. 做好实验小结工作。

**【上机步骤、主要代码及界面】**

# 1 实验准备

本实验数据来自哈佛大学中国历史地理研究中心，我使用了V6 Time Series系列，该系列是哈佛大学2016年更新，该数据集收录公元前222年至公元1911年间中国各地的人口和行政区划相关数据。

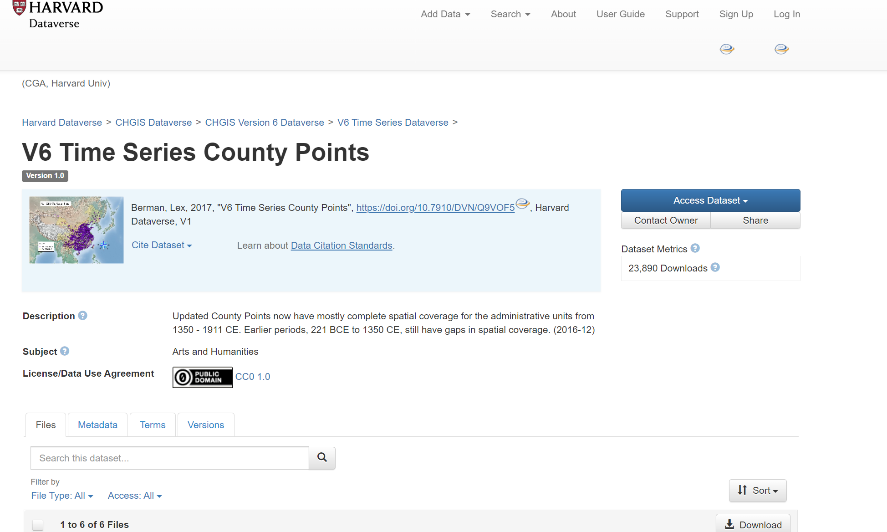


图1 实验数据来源

# 2 使用QGIS+PostGIS展示地理数据

本实验使用的PostGIS版本为3.4.1，首先我们通过XYZ tiles链接加载天地图矢量数据和注记作为底图展示。

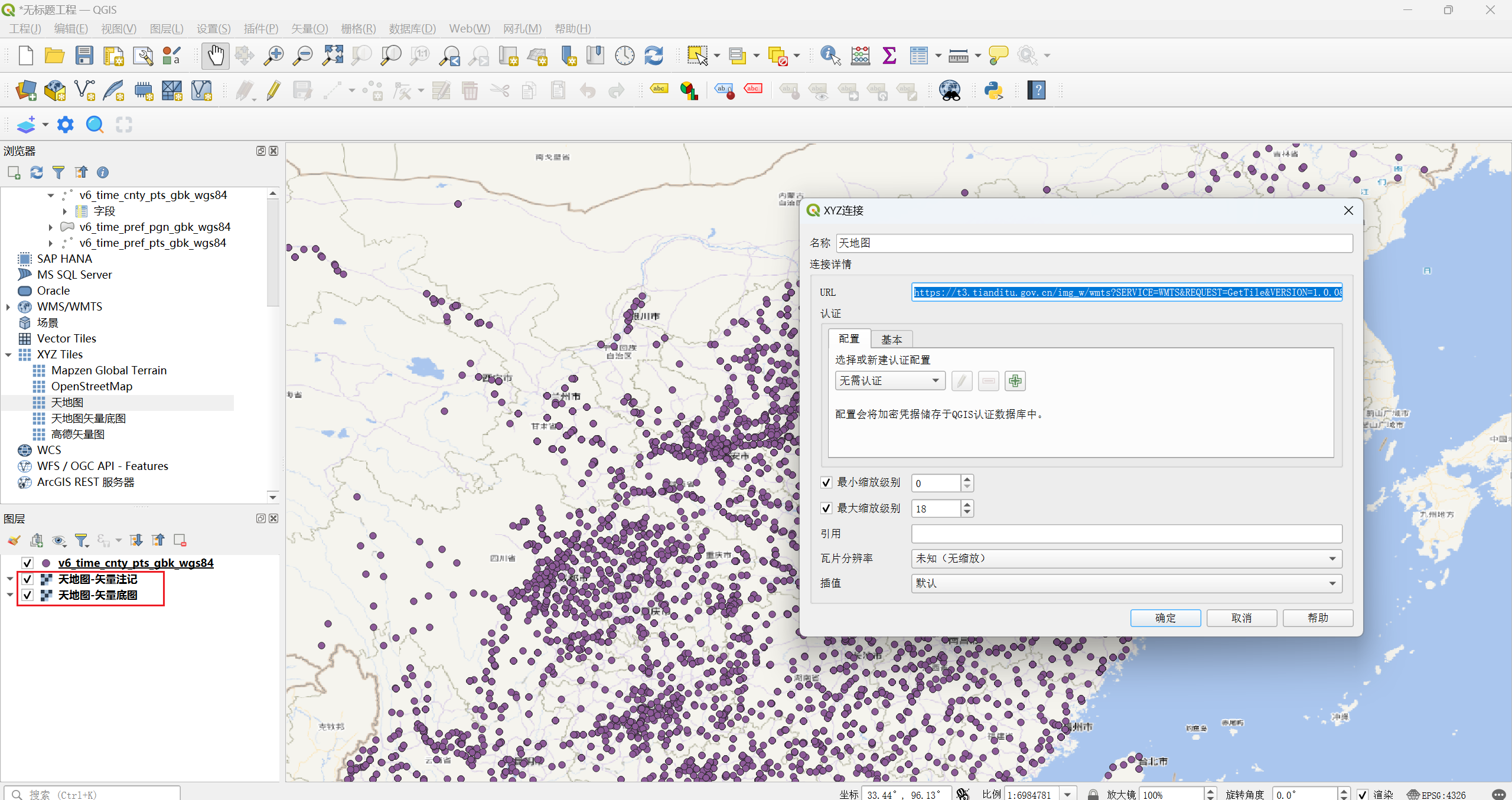


图2 底图引入

接下来我们通过PostgreSQL提供的pgAdmin数据库管理工具创建一个数据库，并添加PostGIS空间扩展，之后使用QGIS链接我们的数据库，如图3。

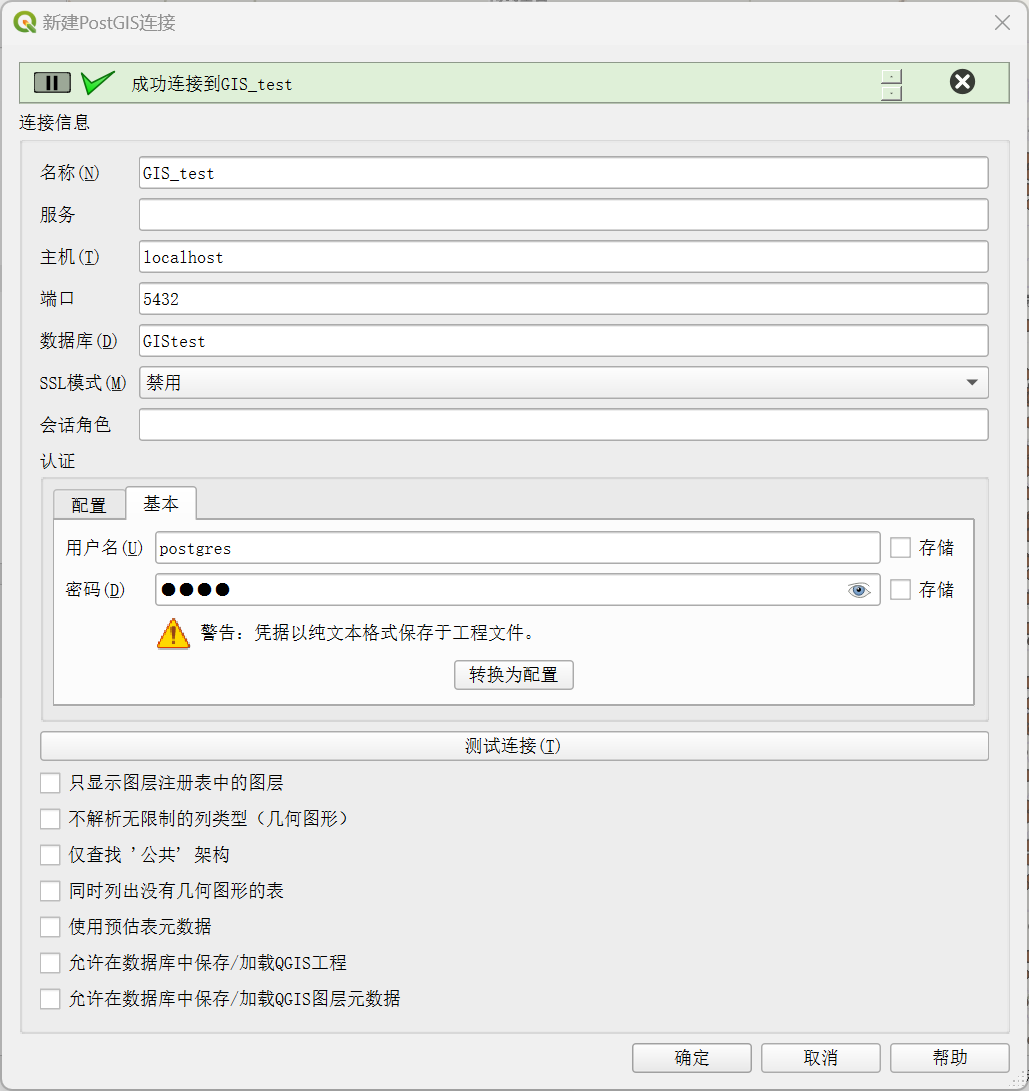


图3 使用QGIS链接空间数据库

链接数据库后，我们可以将我们下载的shp格式的数据导入到数据库中，这样就改变了数据的存储方式，图4。

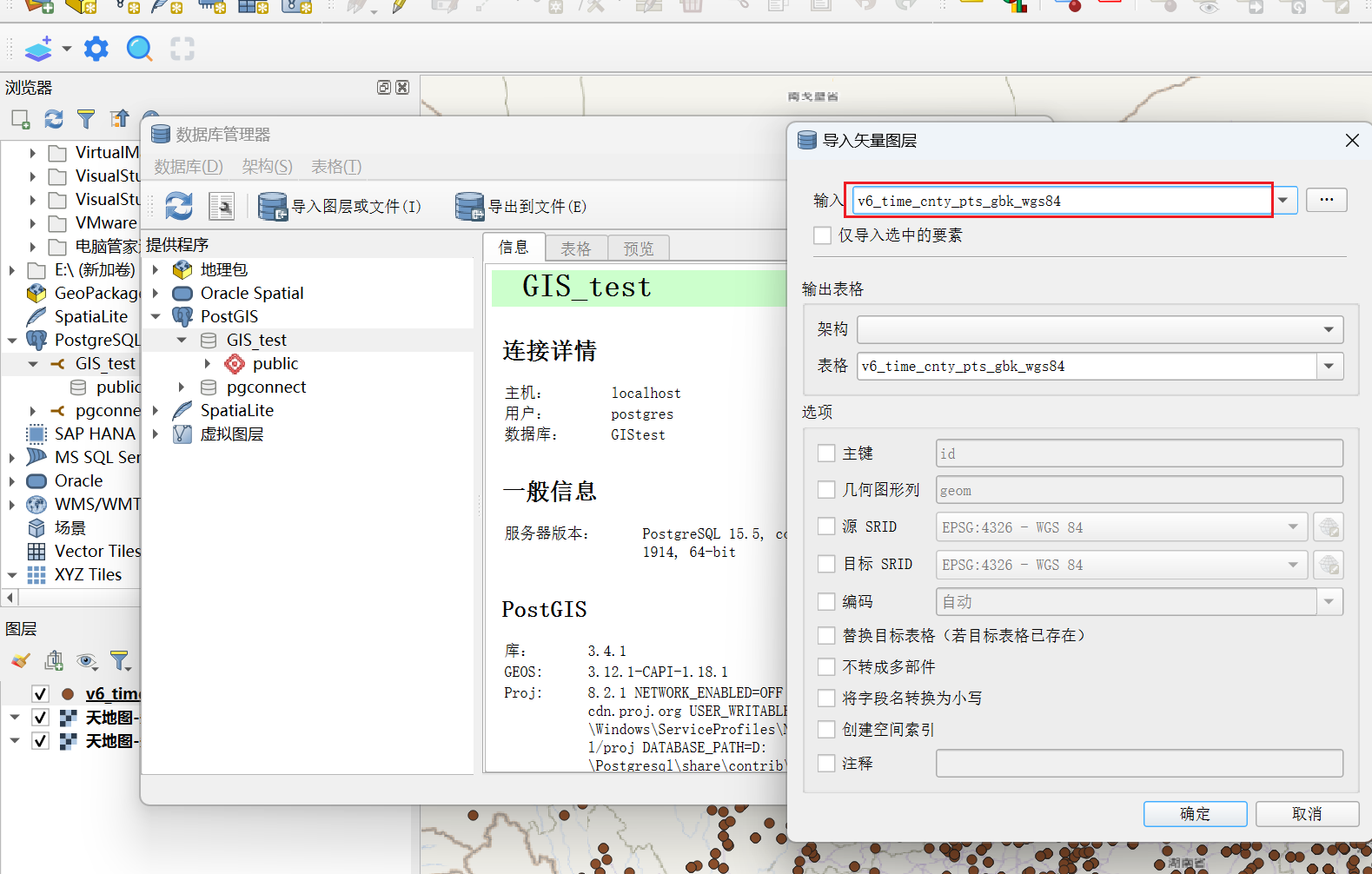


图4 数据入库

接下来我们查看我们导入到空间数据库的图层属性和shp数据对比，虽然展示数据相同，但是它们的存储方式已经发生了质的变化。

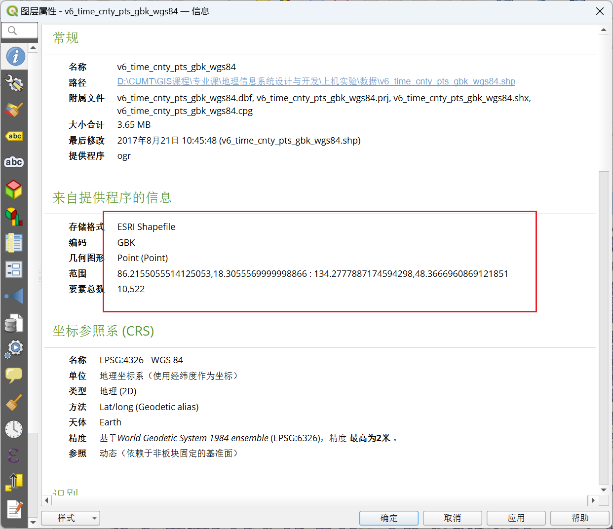
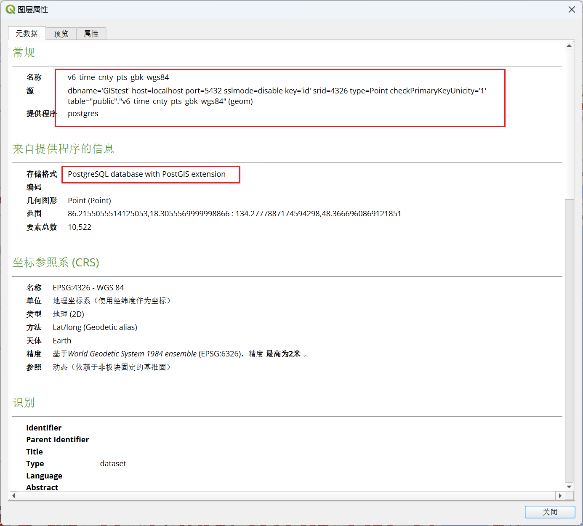


图5 两种存储方式对比，postgis存储（左），shapefile存储（右）

我们可以查看数据的字段信息，该表共有10522行数据，23个字段，属性字段信息如下：

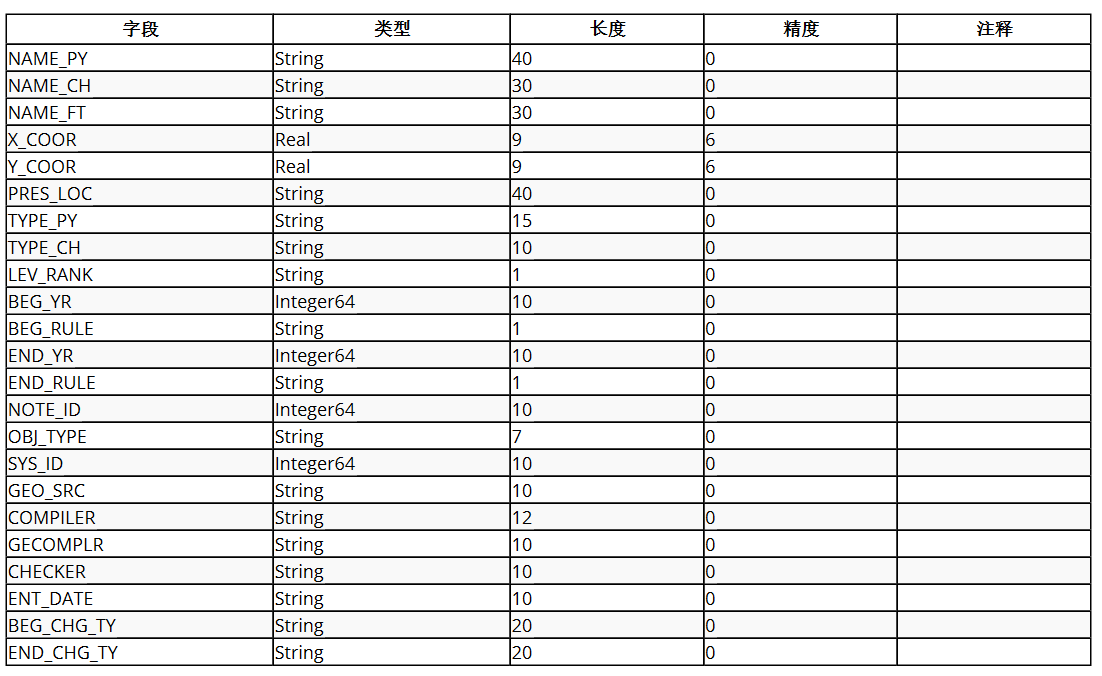


图6 数据表字段展示

下面是数据字段的具体信息：

表1 数据表字段具体信息

|  |  |
| --- | --- |
| NAME\_PY: | 名称的拼音表示（字符串，长度40）。 |
| NAME\_CH: | 中文名称（字符串，长度30）。 |
| NAME\_FT: | 繁体中文名称（字符串，长度30）。 |
| X\_COOR: | X坐标，表示地理位置的经度（实数，精度6）。 |
| Y\_COOR: | Y坐标，表示地理位置的纬度（实数，精度6）。 |
| PRES\_LOC: | 当前位置（字符串，长度40）。 |
| TYPE\_PY: | 类型的拼音表示（字符串，长度15）。 |
| TYPE\_CH: | 类型的中文表示（字符串，长度10）。 |
| LEV\_RANK: | 级别排名（字符串，长度1）。 |
| BEG\_YR: | 开始年份（整数，长度10）。 |
| BEG\_RULE: | 开始规则（字符串，长度1）。 |
| END\_YR: | 结束年份（整数，长度10）。 |
| END\_RULE: | 结束规则（字符串，长度1）。 |
| NOTE\_ID: | 备注ID（整数，长度10）。 |
| OBJ\_TYPE: | 对象类型（字符串，长度7）。 |
| SYS\_ID: | 系统ID（整数，长度10）。 |
| GEO\_SRC: | 地理来源（字符串，长度10）。 |
| COMPILER: | 编译者（字符串，长度12）。 |
| GECOMPLR: | 地理编译者（字符串，长度10）。 |
| CHECKER: | 检查者（字符串，长度10）。 |
| ENT\_DATE: | 进入日期（字符串，长度10）。 |
| BEG\_CHG\_TY: | 开始变更类型（字符串，长度20）。 |
| END\_CHG\_TY: | 结束变更类型（字符串，长度20）。 |

我们可以通过pgAdmin的查看器看到我们存储的数据，图7。这里我们的几何属性是以WKB格式存储的。

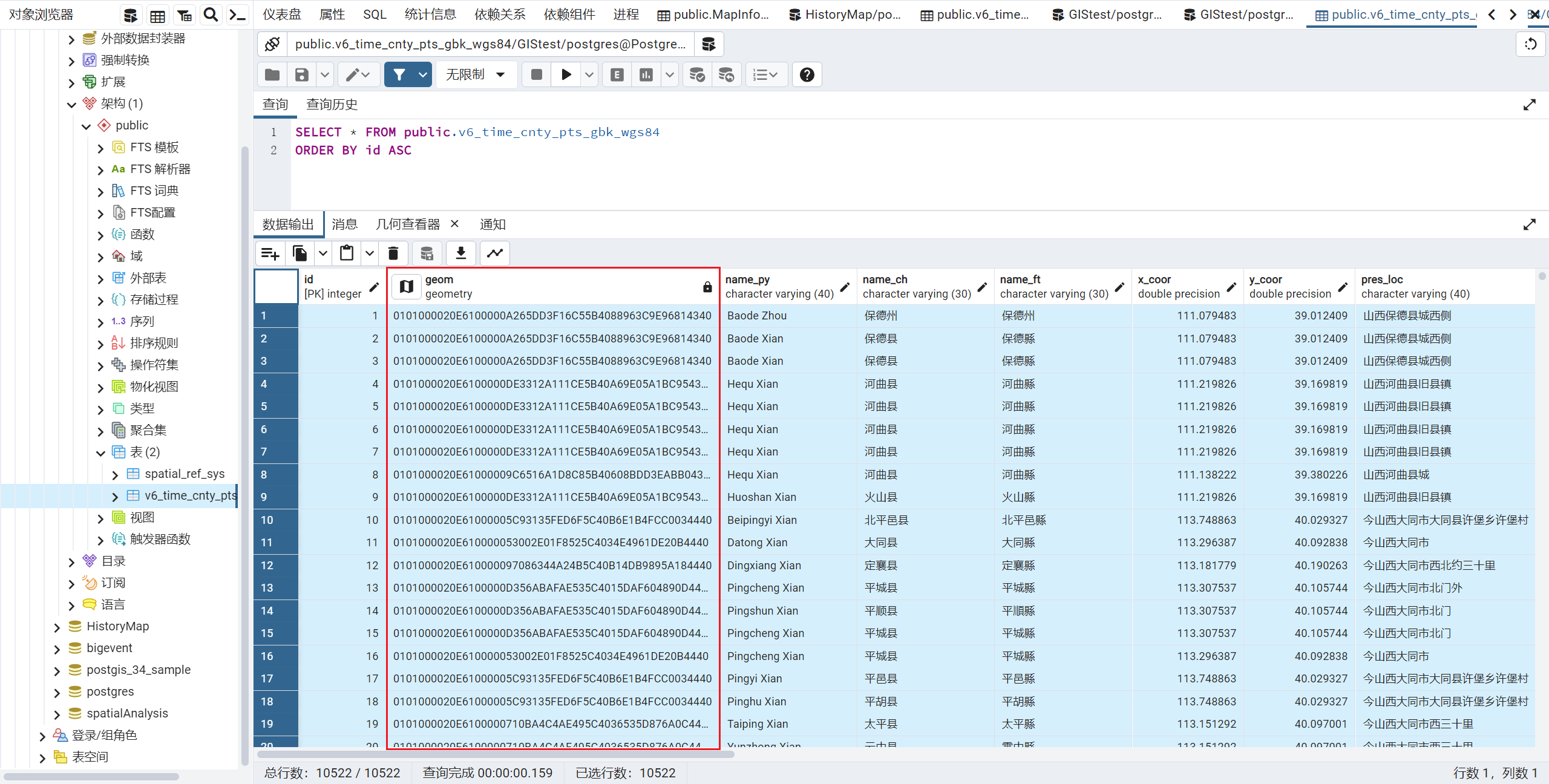


图7 存储数据查看

# 3 实验总结

本实验我使用了QGIS+PostGIS完成，实验过程中让我加深了对地理空间数据库、地理空间数据的引擎的概念，掌握了地理空间数据导入空间数据库的方法，使用了纯开源的方案与传统的ArcSDE相比，更加灵活，可拓展性强，了解到了PostgreSQL数据库的强大性能，查询1万行数据只需不到2ms，此外还对比了PostGIS格式和ESRI Shapefile，让地理数据的存储更加现代化。