

# 一种 Web 地图服务引擎的设计与实践

李晓斌

(华南理工大学 自动化科学与工程学院, 广东 广州 510641)

**摘 要:** 为了解决目前 WebGIS 应用中存在的响应速度慢等不足, 设计和开发了一种基于瓦片的 Web 地图服务引擎, 并探讨了系统架构设计、定位原理、栅格地图制作、业务数据制作、客户端 Maps API 实现等关键技术. 该地图服务引擎已应用于门户网站中, 取得了良好的应用效果.

**关键词:** 瓦片; 地图服务引擎; 栅格地图; WebGIS

**中图分类号:** TP317.4

**文献标识码:** A

**文章编号:** 1000-7180(2011)04-0042-03

## Design and Implementation of a Web Map Service Engine

LI Xiao-bin

(Institute of Automation Science and Engineering, South China University of Technology, Guangzhou 510641, China)

**Abstract:** In order to overcome the shortage of slow response speed of the WebGIS currently applied, this paper designs and develops a map service engine based on tile map, and discuss the system architecture, positioning theory, grid map and business data creation, implementation of client ends Maps API etc. key technique. That engine has already been applied to the portal website and obtains good application effect.

**Key words:** tile; map service engine; Grid map; WebGIS

### 1 引言

早期的 Web-GIS 引擎有两种不同的地图发布技术: 矢量地图发布模式和实时栅格地图发布模式<sup>[1]</sup>. 这两种提供服务的模式都存在缺陷, 矢量地图发布时, 需要在客户端实时渲染, 耗费客户端的资源和时间, 并且在客户端显示时需要安装插件; 实时栅格地图发布时, 服务端的地图引擎需要实时切图, 这是一个耗时操作, 使系统响应时间加长. 所以以上两种 Web-GIS 技术, 均不适合互联网应用环境.

Google 在 2005 年推出瓦片(Tile)式互联网地图 Google Maps, 其基本理念是将地图预先按照缩放级别切割成固定大小的栅格图片, 并按照一定的命名规则存放地图文件. 与业务相关的地物的空间和属性信息则经转换后, 保存在数据库中. 客户端使用 Javascript、AJAX、VML 或 SVG 等 Web 技术, 与服务端进行数据交互. 当客户端请求某一视野范围内的地图时, 在本地计算出所需要的图片, 只需向

服务端请求相应的栅格图片, 并在客户端进行拼接, 这样不需要实时渲染和切图, 速度大大加快. 再加上浏览器本身的本地缓存技术, 使浏览速度大大提高<sup>[2-4]</sup>. 本文将依托全国连片式导航地图, 研究并应用瓦片式地图引擎技术, 建立一个面向公众互联网服务的 Web-GIS 系统, 并探讨系统实现的关键技术.

### 2 系统架构

基于瓦片式地图的 Web 地图服务引擎的结构如图 1 所示, 主要分为三层: 数据层、业务层和表现层. 在整个 Web 地图服务引擎开始工作前, 需要进行数据准备, 包括栅格图片和空间地物数据库.

#### (1) 数据准备

在数据准备阶段, 通过数据转换工具从全国矢量导航电子地图中提取出与地物相关的空间和属性信息后, 保存进空间地物数据库. 同样通过栅格地图制作工具, 读取全国矢量导航电子地图中的空间和

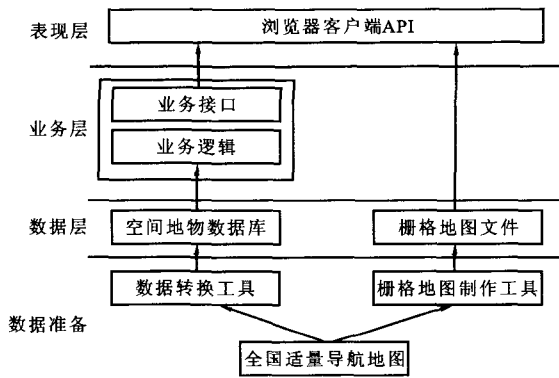


图1 地图服务引擎系统架构

属性信息,并进行地图渲染,然后切成固定大小的栅格地图,按照一定的规则命名和保存.这样就完成了数据准备工作。

### (2)数据层

在该层中具有空间地物数据库和按照一定规则命名的栅格地图文件。

### (3)业务层

在该层封装了操作空间地物数据库的业务逻辑,如:POI搜索、道路搜索、行政区搜索、地理编码及逆地理编码服务等.这些业务逻辑被封装成接口API供表示层调用。

### (4)表示层

表示层主要提供对各种空间数据的呈现.通过多种浏览器技术,包括:HTML、CSS、Javascript、VML、SVG、DOM等技术,使用客户端API开发包对数据层中的地物数据和栅格图片进行调用,然后在客户端的地图上显示出各类数据的空间效果。

## 3 切图及定位原理

待切地图根据所在不同地理范围采用不同的地图投影,对于世界地图主要采用正圆柱、伪圆柱和多圆锥投影.因为需要进行全国范围的切图,所以选用墨卡托 Mercator 正圆柱投影法。

### (1)切图原理

栅格地图引擎的所发布的地图需要预先处理成影像金字塔结构.如图2所示,在0层,世界地图是一张以经纬度为中心的一张  $256 \times 256$  的瓦片地图,在1层,则裂变成4张瓦片,在2层,则裂变成16张瓦片.设当前缩放级别为  $Level \geq 0$ ,则第  $Level$  级的地图切片行数或列数  $row = col = 2^{level}$ ,瓦片数为  $row \times col = 2^{level} \times 2^{level} = 2^{2 \times level}$ .设地球为规则球体,球半径为  $R$ ,经纬度坐标  $(\theta, \varphi)$  处对应的墨卡托投影坐标为  $(X, Y)$ ,则:

$$X = R \times (\theta \times \pi / 180) \quad (1)$$

$$Y = R \times \ln \left( \frac{1 + \sin(\varphi \times \pi / 180)}{\cos(\varphi \times \pi / 180)} \right) \quad (2)$$

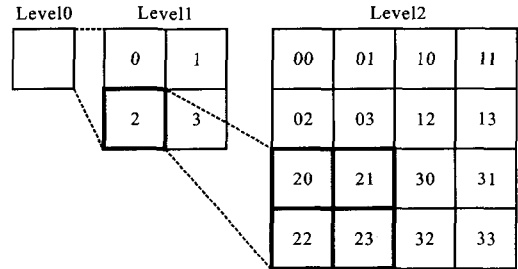


图2 切图原理

(2) 投影坐标计算出瓦片所在行列号和瓦片像素坐标

如图3所示,待切地图的左上角投影坐标为  $(X_{min}, Y_{max})$  和右下角投影坐标为  $(X_{max}, Y_{min})$ ,地图瓦片分辨率为  $PX \times PY$ . 设最左上角瓦片行列坐标为  $(0, 0)$ ,行号向下递增,列号向右递增。

设某点的投影坐标为  $(x, y)$ ,则该点所在瓦片的行号  $i$  和列号  $j$  分别为

$$i = \left\lceil \frac{Y_{max} - y}{Y_{max} - Y_{min}} \times row \right\rceil \quad (3)$$

$$j = \left\lceil \frac{x - X_{min}}{X_{max} - X_{min}} \times col \right\rceil \quad (4)$$

$\lceil \rceil$  为向下取整. 设该点在瓦片上的像素坐标为  $(m, n)$ , 则

$$m = \left( \frac{(x - X_{min}) \times col}{X_{max} - X_{min}} - \left\lceil \frac{x - X_{min}}{X_{max} - X_{min}} \times col \right\rceil \right) \times PX \quad (5)$$

$$n = \left( \frac{(Y_{max} - y) \times row}{Y_{max} - Y_{min}} - \left\lceil \frac{Y_{max} - y}{Y_{max} - Y_{min}} \times row \right\rceil \right) \times PY \quad (6)$$

根据上述公式,在图3中A点的像素坐标为  $(128, 128)$ 。

(3) 瓦片像素坐标和瓦片所在行列号计算出投影坐标

设某点在瓦片上的像素坐标为  $(m, n)$ ,瓦片所在行号为  $i$ ,列号为  $j$ ,则瓦片左上角的投影坐标为  $(tile X_{min}, tile Y_{max})$ :

$$tile X_{min} = X_{min} + j \times \frac{X_{max} - X_{min}}{col} \quad (7)$$

$$tile Y_{max} = Y_{max} - i \times \frac{Y_{max} - Y_{min}}{row} \quad (8)$$

该点的投影坐标  $(x, y)$  为

$$x = \text{tile } X_{\min} + \frac{m \times (X_{\max} - X_{\min})}{PX \times \text{col}} \quad (9)$$

$$y = \text{tile } Y_{\max} - \frac{n \times (Y_{\max} - Y_{\min})}{PY \times \text{row}} \quad (10)$$

根据上述公式,在图 3 中 B 点的投影坐标为(1.5,3.5)。

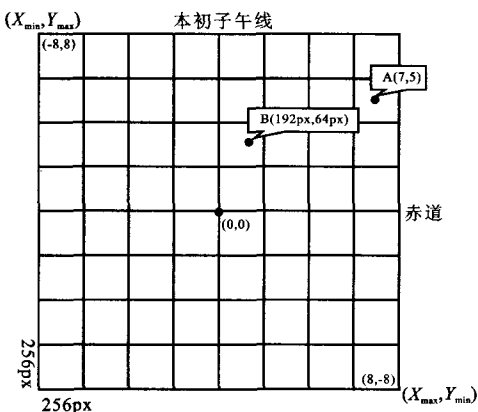


图 3 网格划分

#### (4) 两点之间的地表距离

通过经纬度计算两个经纬度坐标点之间的地表距离,设两个经纬度坐标点 $(x_1, y_1)$ 和 $(x_2, y_2)$ ,则距离 $L$ 为

$$L = R \times \arccos(\cos(x_1 \times \pi/180 - x_2 \times \pi/180) \times \cos(y_1 \times \pi/180) \times \cos(y_2 \times \pi/180) + \sin(y_1 \times \pi/180) \times \sin(y_2 \times \pi/180)) \quad (11)$$

式中, $R$ 为地球半径参数:6 378 137 m.

## 4 瓦片地图及业务数据制作

### (1) 瓦片地图制作

将地图空间数据保存在 Oracle Spatial 中,然后通过栅格地图制作工具读取 Oracle Spatial 中实体的空间信息,按照预配置的样式表和地图等级比例尺,实时渲染地图,并按网格划分将地图图面输出到图片文件中,形成瓦片地图。地图图片为  $256 \times 256$  像素的 png 格式的图片。

### (2) 业务数据制作

业务数据(POI、道路、行政区数据等的属性和空间信息)一般是保存在矢量地图数据库中的,需要将这些业务数据转成关系型数据库表格式,以使用普通的数据库检索技术或搜索引擎检索技术方便地检索和查询。

在业务数据制作时,首先根据矢量地图数据库的格式调用相应厂家的 SDK 读取矢量数据到内存中,经过循环读取矢量图形数据的各个图层中的点、线、面空间和属性信息,并按规范格式进行标准化,

最后保存到关系型数据库中存储。根据搜索引擎的要求建立各种数据索引,就可以对地物的空间和属性信息进行高速检索<sup>[5]</sup>。

## 5 客户端 Maps API 实现技术

客户端 Maps API 目前有两种实现技术,JavaScript 和 Flash ActionScript。下面以 Javascript 技术为例,阐述实现 Maps API 的方法和技术。Maps API 使用 Ajax 提供与服务器异步通信的能力,从而使用户从请求/响应的循环中解脱出来。借助于 Ajax,可以在用户触发请求时,使用 JavaScript 和 DHTML 立即更新 UI,并向服务器发出异步请求。当请求返回时,就可以使用 JavaScript 和 CSS 来相应地更新 UI,而不是刷新整个页面,这样用户体验大大提高。Maps API 使用两个主流的矢量图形标记语言 VML 和 SVG,以在不同的浏览器上绘制点、线、面等图形。VML 最初由 Microsoft 开发,IE5.0 以上版本对 VML 提供支持,而 SVG 由 W3C 制定,是一个开放标准。为了提高浏览器兼容性,本文所研究和开发的客户端 API 对 VML(IE 浏览器支持)和 SVG(FireFox 浏览器支持)都提供了支持。

客户端 Maps API 主要包含以下类包:基类包、RMAP 主类包、工具类包、搜索类包、叠加类包和事件类包。基类包中包含一些公共基础类,如经纬度点类、像素点类等。RMAP 主类主要实现地图的显示和浏览。主要方法包括获取或设置当前的缩放级别、移动地图、设置中心点等数十个方法。工具类包中包含数十个界面工具类。搜索类包中的类主要用来与各种后台的搜索服务接口,传递客户搜索请求,并将搜索结果返回到客户端。后台的搜索服务包括公交搜索、信息点搜索和驾车路线搜索等。叠加类包中包含地图标注类和浮动窗口类等叠加在地图上的信息显示及处理类。事件类包中包含处理自定义事件的类,系统使用观察者(Observer)模式来设计事件的触发和处理。

## 6 结束语

为了解决目前 WebGIS 应用中存在的响应速度慢等不足,本文设计了一种基于瓦片的 Web 地图服务引擎,并探讨了实现该地图引擎的系列关键技术。目前本文提出的地图服务引擎已应用于地图门户网站([www.365ditu.com](http://www.365ditu.com))和中国公路信息网,均取得良好的应用效果。为了实现地图服务引擎

(下转第 48 页)

每次读取 str\_log 结构后,先判断 str\_log.date 是否大于指定的时间,若小于该时间点,就按照上面的流程恢复数据,若大于该时间点,就说明恢复已经完成,成功退出。

#### 4 结束语

通过以上的分析可以看出基于日志文件的热备份方法具有如下优点:

(1) 它是以二进制字符流的形式按数据块来进行备份的,因此每一条日志与文件真实的数据结构之间没有关系,难以破译。以数据库文件为例,一个数据块可能包括一个表中的多条记录,也有可能仅包括半条记录,甚至也可能包括一个完整的小表。因此备份内容可读性很差,安全性很高。

(2) 该算法是以日志为基础实现的,日志文件的大小可以任意控制,可以方便的实现异地实时备份。要备份的文件每有一处改动,就生成一个日志文件,然后立即通过网络传输到异地的其他机器上,这样,无论何时这台机器出现故障,都可以通过备份机器上的数据进行恢复。

(3) 恢复点是任意的。该算法备份了任意时刻的中间数据,并且它的恢复完全模拟了系统修改数据的全过程,因此,可以恢复数据到任意时刻。这对于安全性要求很高的数据是很有必要的,因为它可以追查对数据的所有改动。

(4) 该算法经过的改进后,可以跨平台,跨文件实现方式应用。不同系统的文件块的大小可能不同,但是经过计算可以找到其中的对应关系,并将 str\_

log.content 的内容分几次进行复制就可以实现。一种病毒一般是针对某种操作系统下的文件才会发作,因此实现跨平台备份后,可以更有效地保证数据的安全。

#### 参考文献:

- [1] Farley M. SAN 存储区域网络[M]. 孙功星,蒋文保,范勇,等译. 北京:机械工业出版社,2002.
- [2] 窦万峰,朱鸣,陈卉. 实时协同编辑中群组可靠性和容错研究[J]. 微电子学与计算机,2007,24(9):95-97.
- [3] 刘茂诚,刘展,袁海兰,等. 数据备份系统的关键参数估计[J]. 计算机工程,2008,34(13):70-71,74.
- [4] 高航,罗宇. RTEMS 上基于软件重算的容错支撑设计研究[J]. 微电子学与计算机,2008,25(10):151-154.
- [5] 颜洪梅,战守义,杨方廷. 网络系统中数据备份技术的研究[J]. 计算机应用研究,2006,23(12):154-156.
- [6] 范丽雅,王万森,关永. 数据库中数据恢复的研究[J]. 计算机工程与设计,2008,29(3):563-565.
- [7] 贺亚茹. Oracle 数据库日志文件损坏时修复方法的实验研究[J]. 计算机应用,2009,29(12):393-395.
- [8] 廖国琼,刘云生. 基于实时日志的嵌入式实时数据库恢复策略[J]. 计算机学报,2007,30(4):672-679.

#### 作者简介:

申新鹏 男,(1975-),讲师,博士.研究方向为对等计算、网络计算。

李战怀 男,(1961-),教授,博士生导师.研究方向为数据库、数据挖掘。

赵晓南 女,(1979-),博士研究生.研究方向为海量数据存储、数据管理。

(上接第 44 页)

的快速部署,未来可以对目前的散体文件存储方式进行改进,使用海量图片打包方式,解决海量栅格图片的快速部署问题。

#### 参考文献:

- [1] 彭纪奎,胡平. 基于 Web 的空间数据可视化研究[J]. 微电子学与计算机,2010,27(1):147-150.
- [2] Joe Schwartz. Bing Maps Tile System[EB/OL]. (2010-06-22)[2010-08-12]. <http://msdn.microsoft.com/en-us/library/bb259689.aspx>.
- [3] OSGEO. Tile Map Service Specification[EB/OL]. (2010

-03-06)[2010-08-12]. [http://wiki.osgeo.org/wiki/Tile\\_Map\\_Service\\_Specification](http://wiki.osgeo.org/wiki/Tile_Map_Service_Specification).

- [4] 朱玉周,杨芳,魏璐,等. 瓦片网络地图技术在河南交通气象服务系统中的应用[C]// 2007 年中国气象学会年会. 广州:中国气象学会,2007.

- [5] 胡双双,秦杰,谢惠. 一种实用元搜索引擎的设计与实现[J]. 微电子学与计算机,2009,26(4):67-69.

#### 作者简介:

李晓斌 男,(1971-),博士研究生.研究方向为智能交通、交通诱导。