

The Design of the Lightweight WebGIS Based on GeoJSON by Using HTML5 Canvas*

Chunyu Liang, Xintong Li

College of Geographical Sciences, Fujian Normal University, FuZhou
Email: leybb_123@126.com, lixt@lreis.ac.cn

Received: Sep. 3rd, 2012; revised: Sep. 19th, 2012; accepted: Sep. 27th, 2012

Abstract: Traditionally, WebGIS vector data visualization lack of unified standards in the browser and tile map loading speed is slow. This paper introduces the newly HTML5 and GeoJSON technologies, and the method of using JavaScript to portrayal GeoJSON data which contains geometry information on HTML5 Canvas in the application of lightweight WebGIS, we designed a HTML5 Canvas client-side API for GeoJSON, then the API was used to build Fuzhou city real-time traffic information WebGIS system. The method of HTML5 Canvas combined with GeoJSON technology in the browser side vector data rendering was confirmed in this system to have the advantages of cross-platform and faster response, it can overcome the defects of traditional WebGIS.

Keywords: WebGIS; HTML5 Canvas; GeoJSON; Real-Time Traffic

使用 HTML5 Canvas 构建基于 GeoJSON 的轻量级 WebGIS*

梁春雨, 李新通

福建师范大学地理科学学院, 福州
Email: leybb_123@126.com, lixt@lreis.ac.cn

收稿日期: 2012 年 9 月 3 日; 修回日期: 2012 年 9 月 19 日; 录用日期: 2012 年 9 月 27 日

摘要: 针对传统 WebGIS 矢量数据在浏览器中可视化缺乏统一标准以及采用地图切片方式响应速度慢的现状, 引入了 HTML5 Canvas 和 GeoJSON 技术, 探讨了采用 JavaScript 解析包含位置信息的 GeoJSON 数据, 并将结果作为地理要素绘制在 HTML5 Canvas 上的方法在轻量级 WebGIS 中的应用, 设计实现了针对 GeoJSON 的 HTML5 Canvas 客户端 API, 并依据此 API 构建福州市区实时路况信息发布系统。验证了使用 HTML5 Canvas 结合 GeoJSON 技术在浏览器端矢量数据渲染上具有跨平台、响应速度快的优势, 克服了传统 WebGIS 存在的缺陷。

关键词: WebGIS; HTML5 Canvas; GeoJSON; 实时路况

1. 引言

WebGIS 是 GIS 与 Internet 结合的产物, 计算机技术的持续发展, 新兴技术不断涌现, 为 WebGIS 提供了强大的技术支撑。而现今比较通用的 WebGIS 客户端技术有 Flash、Silverlight 和 SVG(Scalable Vector

Graphics, 可缩放矢量图形)等。Flash 和 Silverlight 凭借其富客户端的体系结构、良好的用户体验受到了 WebGIS 开发者的广泛关注。但二者都属于私有技术, 缺乏与其他开放标准相融合的特性, 使其在 WebGIS 的应用中受到了阻碍。SVG 为可伸缩矢量图形, 采用 XML 格式主要用于在浏览器中显示矢量图形, 因为 SVG DOM(Document Object Model)遵循 DOM1、

*资助信息: 福建省自然科学基金资助项目。

DOM2 规范的大部分内容^[1]，其每个属性和样式都可以通过脚本编程来访问，使 SVG 在 WebGIS 中应用既提高了效率又降低了开发难度，但由于其在 IE 内核的浏览器中支持程度较差，互操作性与跨平台性不佳，并没有得到在 WebGIS 中的广泛应用。HTML5 Canvas 是 HTML5 的一部分，已经取得大多数浏览器的支持，其本质上是一个位图画布，通过 JavaScript 在网页上通过浏览器在不依赖任何插件的情况下高效地绘制绚丽、动态的图形，有效解决了传统技术存在的问题。

目前，各大网站推出的实时路况发布系统大多数使用对发布的路况地图服务进行切片，根据用户请求将具有路况信息的切片返回客户端并叠加在地图上的方式，这种方式不仅浪费流量，而且客户端响应速度和跨平台性不佳，本文试图采用 HTML5 Canvas 结合 GeoJSON 技术实现福州市区实时路况信息的发布，提高了客户端相应速度，增强了系统跨平台性。

2. HTML5 在 WebGIS 中的应用

2.1. HTML5 Canvas

HTML5 将会成为下一代 HTML(Hypertext Markup Language, 超文本标识语言)最主要的版本，用于取代 1999 年所制定的 HTML 4.01 和 XHTML 1.0 标准的 HTML 标准版本^[2]，增强了 Web 网页的表现性能，追加了本地数据库并且在设计之初就充分考虑了其跨平台性，并不需要一定运行 Windows、Mac OS X、Linux、Multics 或者其他任何特定的操作系统，唯一需要的就是一个现代浏览器^[3]。目前 HTML5 现在仍处于发展阶段，各大浏览器厂商已经展开了对 HTML5 标准支持的竞赛，HTML5 的新特性也得到了大部分浏览器的良好支持。广义论及 HTML5，实际指的是包括 HTML、CSS 和 JavaScript 在内的一套技术组合^[4]。它希望能够减少浏览器对于需要插件的网络应用服务(plug-in-based rich internet application, RIA)，如 Adobe Flash、Microsoft Silverlight 的需求，正如作为 W3C 的 HTML5 起草者之一的 IanHickson 所说，我们的目标是让 Web 远离私有技术。

W3C(World Wide Web Consortium, 万维网联盟)在 HTML5 标准中定义<Canvas>标签，是依赖分辨率的、以屏幕坐标为参考的矩形位图画布，可以通过获

取上下文的方式在 Canvas 上绘制任意图形，甚至绘制图像。此外，HTML5 Canvas 提供了一套功能强大的二维绘图 API，包括以丰富的色彩绘制点、线、面、文字，和可操作每个像素的栅格图像等都能通过使用 HTML5 Canvas API 实现，因此，HTML5 Canvas 在 WebGIS 中具有较高的应用价值。

2.2. 基于 HTML5 Canvas 的地理信息显示

HTML5 Canvas 为 WebGIS 基本地理要素的表现提供了良好的支持，通过 Canvas API 提供的函数库可以在 Canvas 中绘制出 WebGIS 表现所需要的多种样式的点、线、面、影像以及文字，使用 HTML5 Canvas 可以绘制福建师范大学校园矢量图，如图 1 所示，校园轮廓及主要建筑物作为面状要素，道路作为现状要素绘制，要素名称以点状要素的形式根据坐标采用绘制文本方式绘制。通过对绘制顺序和绘制时属性设置的不同展现不同的叠加效果，并且根据使用者的需求选择不同样式的图例，提高了灵活性与互操作性。本例使用 IE9、Google Chrome、Opera、Firefox、Maxthon3 等现代浏览器和智能手机浏览器都可以流畅的对地图进行放大、缩小、平移等基本地图操作。可见，HTML5 Canvas 可以作为客户端快速展现复杂的地理数据，并且在展现地理数据的灵活性与互操作性方面优于传统的切片技术。

同时，在绘制大数据量、多图层的情况下使用 HTML5 Canvas 绘制矢量图形相比传统地图切片技术在网页加载速度上也有明显优势，表 1 列出了使用 WebWait 在线测速网站对 Canvas 绘制 25,000 条矢量线段、1050 条路名文本与采用切片技术显示相同数据浏览器的响应时间。

2.3. 使用 Geolocation API 在网页地图中的实现定位

Geolocation API 是 HTML5 中的一个规范，规定了如何使用脚本访问主机设备的地理位置信息。截至到现在，各浏览器对 Geolocation API 的支持情况^[5]如表 2 所示。

根据不同的设备 Geolocation API 可以获取以下数据源：

- IP 地址

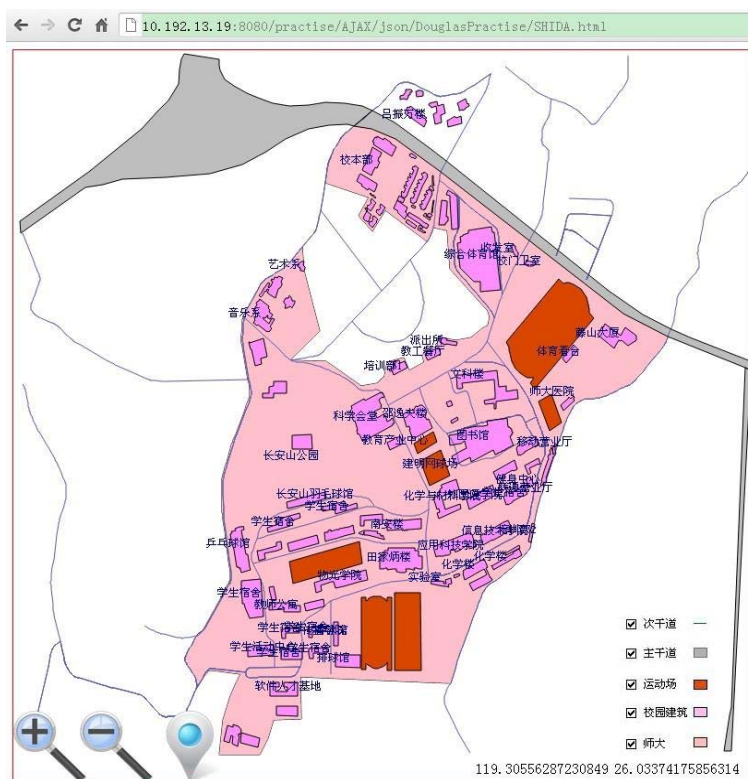


Figure 1. Using HTML5 Canvas draw the map of Fujian Normal University
图 1. HTML5 Canvas 绘制福建师范大学校园矢量图

Table 1. The difference of loading using two kind of method
表 1. 两种方法加载测试结果

| 方法 | 平均值 | 中值 | 标准差 |
|-----------------------------|--------|--------|------|
| HTML5 Canvas | 0.05 s | 0.03 s | 0.03 |
| GeoServer + Open Layers(切片) | 0.22 s | 0.20 s | 0.03 |

Table 2. The condition of browse support Geolocation API
表 2. 各浏览器对 Geolocation API 的支持情况

| 浏览器 | 支持版本 | 目前版本 |
|-------------------|-------|--------|
| Firefox | 3.5+ | 14.0.1 |
| Chrome | 5.0+ | 21 |
| Internet Explorer | 9.0+ | 10.0 |
| Safari | 5.0+ | 6.0 |
| Opera | 10.6+ | 12.01 |
| Android | 2.0+ | 4.0 |
| BlackBerry | 6+ | 7 |
| IOS | 3.1+ | 5.1 |

- 三维坐标
 - GPS(Global Positioning System, 全球定位系统)
 - 从 RFID、Wi-Fi 和蓝牙到 Wi-Fi 的 MAC 地址

- GSM 或 CDMA 手机的 ID^[6]

Geolocation API 可以获取当前设备的纬度、经度、精确度、海拔、速度和方向信息，经度、纬度以十进制地理坐标表示，精确度、海拔以 m 为单位，方向返回相对于正北方向的顺时针夹角，速度以 m/s 为单位，为保证更高的精度，许多设备使用 1 个或者多个数据源的组合。因此，通过 Geolocation API 实现用户位置在网页中的显示具有准确性和跨平台性的优势，也是网页 LBS 和网页导航方面获取用户位置信息的主要实现方式。

3. 以 GeoJSON 为数据格式的 HTML5 Canvas 客户端实现

3.1. 网络地图服务与 GeoJSON

OGC(Open GIS Consortium, 开放地理信息系统协会)制定的开放地理信息系统互操作规范为地理信息规范化提供了基础保障，消除了 GIS 系统间的信息孤岛，使得跨平台分布式的 GIS 系统间可以通过 OGC 标准协议进行交互^[7]。在 OGC 制定的规范中，最重要

的地理信息性服务实现规范是网络地图服务(WMS)、网络要素服务(WFS)和网络覆盖服务(WCS)。其中, WFS 标准定义了一些操作,这些操作允许用户在分布式的环境下通过 HTTP 对空间数据进行查询、编辑等操作^[8]。在 OGC 的规范中规定作为客户端请求 WFS 服务的返回格式有以下几种: GML、CSV、SHAPE-ZIP、JSON 等。目前,虽然 GML 已经得到了广大 GIS 服务器厂商的完美支持,但是 GML 也存在缺陷, GML 存在的主要缺陷具体表现在它是基于 XML 的数据格式,由于 XML 收自身格式的限制,会带来一定程度的信息冗余,降低在网络上的传输效率,并且 JavaScript 在解析 XML 时会存在浏览器的兼容性问题^[9]。

JSON(JavaScript Object Notation)为 Web 应用开发提供一种轻量级的数据交换格式^[10]。GeoJSON 是能够对各种地理数据框架进行编码的格式,是 JSON 在表达地理数据方面的分支,它支持 OGC 定义的以下几种几何类型: Point、LineString、Polygon、MultiPoint、MultiLineString、MultiPolygon 和 GeometryCollection。又因为其具有表达数据的简洁性、JavaScript 原生支持等特点。

3.2. GeoJSON 描述地理数据的方式

任何 OGC 标准下的地理数据类型都可以方便的通过 GeoJSON 进行描述,如下代码展示如何使用 GeoJSON 描述一条包含坐标系、边界范围和要素 id、coordinates、cellroadid 等属性的线状地理要素。

```

{"type": "FeatureCollection",
 "features": [
   {
     "type": "Feature",
     "id": "wfs_roadwgs84.2073",
     "geometry": {
       "type": "MultiLineString",
       "coordinates": [
         [[119.197383,26.070953],[119.197698,26.0710269
         99999997]]
       ]
     },
     "geometry_name": "the_geom",

```

```

"properties": {
  "cellroadid": 2073,
}
},
"crs": {"type": "EPSG", "properties": {"code": "4326"}},
"bbox": [118.621535,25.48216,119.66012,26.543263]
}

```

GeoJSON 是 JavaScript 原生支持的数据格式,浏览器通过 JavaScript 可以快速准确的对 GeoJSON 文件进行解析,得到如图 2 的结构,使用者可以通过脚本访问该树状图每个节点信息,例如获取当前图层最大范围(bbox)、坐标系(crs)、要素坐标位置(coordinates)、要素属性(cellroadid)等。并且 OGC 所支持的所有几何类型在 GeoJSON 中都采用如下结构进行描述,因此,依据其在地理要素的描述上具有简洁性和一致性的特点,选择使用 GeoJSON 作为 HTML5 Canvas 展现地理信息的数据格式。

3.3. HTML5 Canvas 结合 GeoJSON 在网页绘制矢量图形

3.3.1. 绘制方法

HTML5 canvas API 提供强大的函数库以保证使用 JavaScript 在网页中绘制任意类型的地理要素。使用 fillText(“文本内容”, X, Y)方法将文本内容以文本左上角为起始坐标绘制于相应的 X, Y 屏幕坐标上;线状要素使用 moveTo(X,Y)、lineTo(X,Y)方法绘制具有起始坐标的线段,并根据 strokeStyle(“color”)方法设置绘制线段的颜色;使用 moveTo(X,Y)、lineTo(X,Y)方法配合 closePath()方法也可以绘制任意形状的多边形。

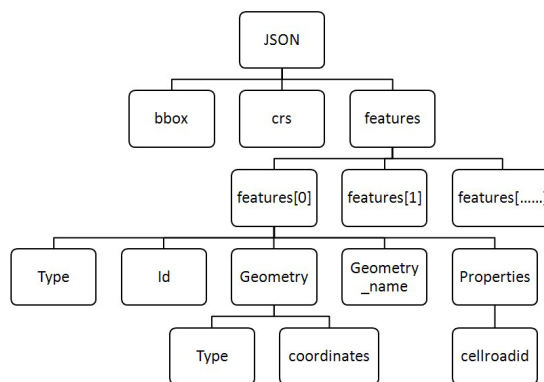


Figure 2. The structure of analytic GeoJSON using JavaScript
图 2. JavaScript 解析 GeoJSON 的结构

使用 JavaScript 解析 GeoJSON 得到的要素坐标保存在变量中, 通过地理坐标转换模块得到屏幕点坐标, 针对不同要素类型使用上述方法进行绘制。

3.3.2. 坐标转换

坐标转换的核心功能是将取得的地理坐标转换成能在浏览器中显示的屏幕坐标, 再将屏幕坐标交给 HTML5 Canvas 进行绘制。

由于地理坐标是延 X 轴向右递增, 延 Y 轴向上递增, 而在 HTML5 Canvas 中, 坐标是从左上角开始, 延 X 轴向右递增, 延 Y 轴向下递增。所以应采用如下公式对地理坐标向屏幕坐标转换:

$$X = (\text{Lon} - \text{minLon}) * 3600 / \text{scaleX} \quad (1)$$

$$Y = (\text{maxLat} - \text{Lat}) * 3600 / \text{scaleY} \quad (2)$$

其中, X 表示转换得到的屏幕横坐标, Y 表示转换得到的屏幕纵坐标, Lon 表示当前要素结点的经度, minLon 表示该图层范围的最大经度, Lat 表示当前要素结点的纬度, minLat 表示该图层范围的最大纬度, minLon 和 minLat 可以通过 GeoJSON 中的 bbox 中获得, 同时, 为保证精度, 将差值乘以 3600 换算成秒。而 scaleX、scaleY 有如下公式得到。

$$\text{scaleX} = ((\text{maxLon} - \text{minLon}) * 3600) / h \quad (3)$$

$$\text{scaleY} = ((\text{maxLat} - \text{minLat}) * 3600) / y \quad (4)$$

h 代表 HTML5 Canvas 画布的高度, y 代表 HTML5 Canvas 画布的宽度。scaleX、scaleY 分别代表 X 轴和 Y 轴上每像素代表的经度/纬度的秒数。

使用以上公式, 可以将任意一点的地理坐标转换为屏幕坐标绘制在 HTML5 Canvas 画布当中, 并且能保证较高的精确度。

3.4. GeoJSON HTML5 Canvas 客户端 API 设计

依据上述方法, 本文使用 JavaScript 语言和 AJAX 技术设计了针对 GeoJSON 文件的 HTML5 Canvas 客户端 API, 帮助开发者以简洁的代码实现对 GeoJSON 文件在 HTML5 Canvas 上的显示与基本操作, 主要类结构如图 3 所示。

MapCanvas API 的核心功能是通过 Map 类构造基础画布, 使用带 URL 参数的 Layer 类构造图层对象, 目前 API 只提供对 GeoJSON 文件 URL 的支持, 可以针对各 Layer 对象的显示颜色、绘制线宽等属性

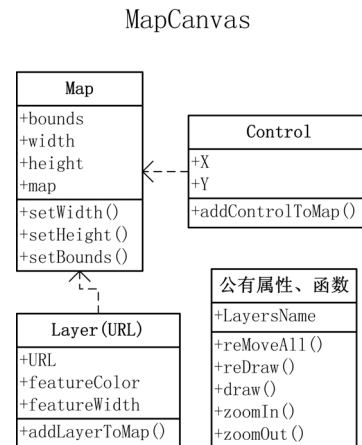


Figure 3. Main class diagram

图 3. 主要类结构图

进行设置, Layer.addLayerToMap()方法将各图层对象添加至基础画布上绘制。Control 类负责构造基本地图操作控件, X、Y 属性控制控件位置, 并使用 Control.addControlToMap()方法将构造好的控件添加至 Map 对象上。

4. 福州市区实时路况系统设计

4.1. 实时路况矢量数据发布现状的分析

随着我国经济迅速发展和城市化水平不断提高, 私家车拥有量激增, 城市道路交通拥堵现象日益凸显, 设计一个方便、快捷的能为市民出行作为参考, 达到方便市民出行的实时路况发布系统, 让用户及时了解所在位置的实时交通状况, 及时选择出行路线, 就显得尤为重要。目前各大地图网站纷纷推出自己的实时路况发布系统, 大多采用传统的地图切片技术、使用 Flash、Silverlight 浏览器插件技术或者客户端软件方式支持 PC 端或手机端对路况进行查询, 表 3 总结了各实现方式的优缺点。

然而依据前文提出的方法, 使用 HTML5 Canvas 技术构建基于 GeoJSON 的福州实时路况发布系统存在以下优势:

- 在客户端无插件或者不使用安装客户端软件的情况下, 给用户丰富的体验效果。
- 前端渲染有利于灵活表现, 达到动态显示效果, 用户体验相对提高。
- 实现不同移动端的跨平台的浏览, 便于开发人员开发、维护、更新。

Table 3. The advantages and disadvantages of each realization way
表 3. 各实现方式的优缺点

| 实现方式 | 代表网站 | 优点 | 缺点 |
|--------------------------|----------------------------|-------------------|---|
| 地图切片技术 | 百度地图、Google Map、Sougou 地图等 | PC 端访问速度快、地图色彩艳丽 | 移动端兼容性差，加载速度慢，耗费流量。 |
| Flash、Silverlight 等的插件技术 | MapABC | 在 PC 端有良好的用户体验 | 客户端必须安装私有插件，不易跨平台，手机端加载速度慢，耗费流量，耗费手机电量。 |
| 安装客户端软件方式(主要在手机端) | 百度地图客户端、搜狗地图客户端等 | 查询方便快捷，功能强大，便于开发。 | 每个平台都要开发相应软件，软件更新不易推广，维护成本较高。 |

4.2. 系统体系结构设计

系统体系结构图如图 4 所示，空间数据库采用 PostGIS 存储数据，将已有的福州道路和路名 Shape File 文件通过 PostGIS 提供的 shp2pgsql 和 psql -f 命令导入 PostGIS 中，其中道路数据为线状地物，主要包含位置信息和道路标识号，路名数据为点状地物，主要包含位置信息和道路名称。

GIS 服务器主要提供 WFS 服务，采用 GeoServer 作为 GIS 服务器将存储在 PostGIS 中的福州道路数据和路名数据发布 WFS 服务，使用 GeoJSON 格式输出并保存成 JSON 文件。因为道路数据短时间不发生变化的性质，将道路 GeoJSON 文件存储在 Web 服务器的工作目录中以便客户端访问。

Web 服务器主要工作是接受客户端使用 AJAX 发送的请求，将本地道路数据的 JSON 文件和本地路名数据 JSON 文件发送至客户端。并且通过 Web 服务器实时获取第三方提供的路况信息，保存为本地的文本文件，随着路况文件的实时变化更新原有路况文件。道路数据文件(JSON 文件)与路况信息文件(txt)通过内部共有的道路标识号相关联。Web 服务器使用 Apache Tomcat，获取第三方路况信息使用 Java 语言实现。

本文表现层主要采用前文设计的 MapCanvas API 实现，具体工作流程如下：通过使用道路数据文件和路名文件的 URL 分别构造 Layer 对象，浏览器通过 MapCanvas API 向 Web 服务器发送 AJAX 请求，读取 URL 指向的本地道路数据和路名数据 JSON 文件的内容，并使用 GeoJSON 解析器解析出每段道路的地理坐标和道路编号，通过正则表达式在路况信息文件中匹配相应道路编号的路况信息，得到该路段的路况数据，使用坐标转换模块将地理坐标向屏幕坐标转换，并将转换后的屏幕坐标以及路况数据以颜色区分绘

制在浏览器中，红色路段代表拥堵，黄色路段代表缓行，绿色路段代表畅通，黑色路段表示暂无数据。

5. 系统功能实现

系统采用 HTML5 Canvas 结合 GeoJSON 的方式构建，能够在浏览器端快速绘制福州市区实时路况信息发布图，具有较好的跨平台性，福州市区实时路况系统主界面如图 5 所示，从路况发布图上可以看出乌山路、道山路等路段以绿色显示，表示当前该路段畅通，八一七北路、津泰路等路段显示为红色表示当前路况拥堵，法海路以及白马北路等路段为黄色表示缓行。可见系统能够清晰展现福州市区实时路况信息，并且实现地理位置定位以及基本地图操作功能。

图 6 为系统在智能手机端运行结果，系统能够在低端智能手机上流畅运行，路况展现清晰，具有强跨平台性，让使用者随时随地使用智能手机访问当前路况信息，安排灵活的出行方案。

6. 结论与展望

本文通过对 HTML5 Canvas 和 GeoJSON 特性的研究，提出了以 GeoJSON 为数据格式、以 HTML5 Canvas 为展现方式的轻量级 WebGIS 构建方法，设计了针对 GeoJSON 的 HTML5 Canvas 客户端 MapCanvas API，并依据此 MapCanvas API 构建福州市区实时路况发布系统。

实验证明，使用 HTML5 结合 GeoJSON 的方式展示多图层、大数据量地理数据相对于传统的使用加载切片方式展现地图数据速度上有了极大的增强，提高了用户体验，节约网络资源，并且系统在手机端使用流畅，在跨平台方面较其它方式具有明显优势。并且，使用此方法构建轻量级 WebGIS 具有简单易行，灵活快速的优点。

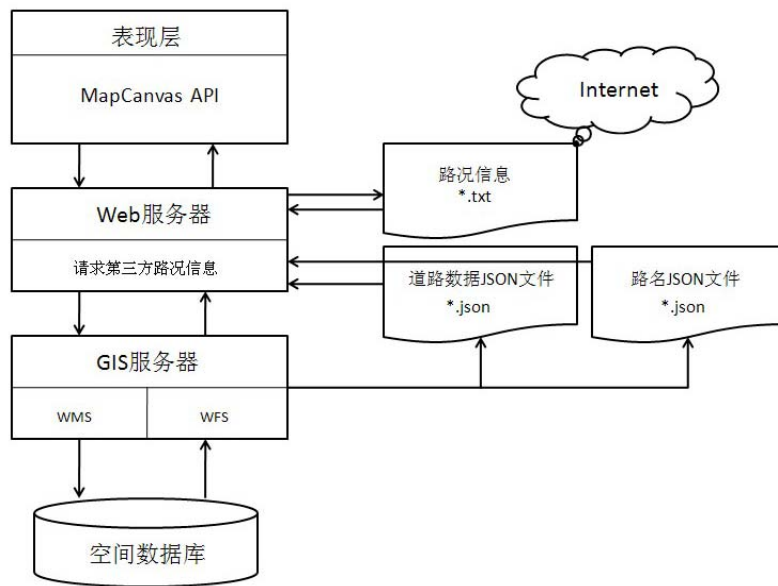


Figure 4. System structure diagram
图 4. 系统体系结构图

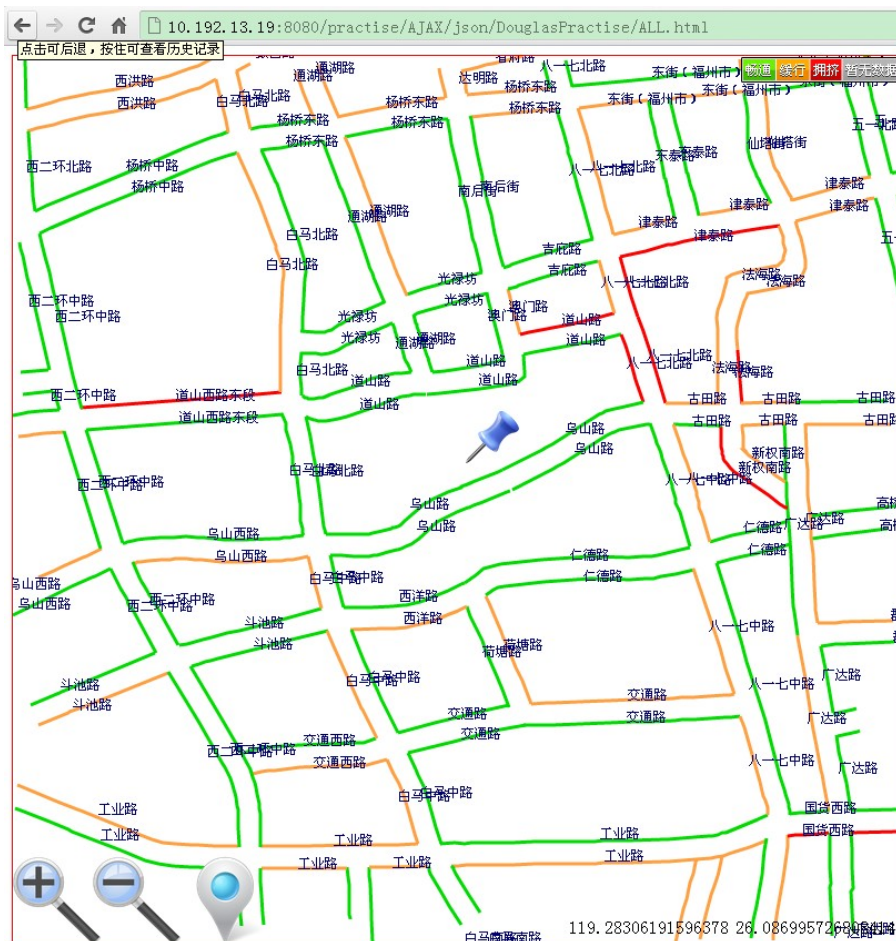


Figure 5. The main interface of Fuzhou real-time traffic system
图 5. 福州市区实时路况系统主界面



Figure 6. The result of using cell-phone
图 6. 手机访问路况效果

HTML5 Canvas 使得客户端可以不借助第三方插件实现快速、灵活的矢量数据在浏览器的绘制，GeoJSON 配合 JavaScript 的使用使得数据在传递过程中更加轻便，更加迅速。所以使用 HTML5 Canvas 结合 GeoJSON 构建 WebGIS 系统将会成为网络地理信息系统应用发展的新方向。

参考文献 (References)

- [1] Scalable vector graphics. Standards, 2012.
<http://www.w3.org/Graphics/SVG>
- [2] P. Lubbers, B. Albers and F. Salim. Pro HTML5 programming. New York: Apress, 2010.
- [3] M. Pilgrim. HTML5: Up and running. Sebastopol: O'Reilly Media, 2010.
- [4] Wikipedia. HTML5, 2012.
<http://zh.wikipedia.org/wiki/HTML5>
- [5] The HTML5 test. Other browsers, 2012.
<http://html5test.com>
- [6] A. Popescu. Geolocation API specification, 2012.
<http://dev.w3.org/geo/api/spec-source.html>
- [7] 龚健雅, 贾文珏, 陈玉敏等. 从平台 GIS 到跨平台互操作 GIS 的发展[J]. 武汉大学学报: 信息科学版, 2004, 29(11): 985-989.
- [8] A. Whiteside, J. Greenwood. OGC web services common standard. Wayland: Open Geospatial Consortium Inc., 2010.
- [9] 陈德权. 基于 GeoJSON 的 WFS 实现方式[J]. 测绘科学技术学报, 2011, 28(1): 66-69.
- [10] Jshomepage. Introducing JSON, 2012.
<http://www.json.org>