

# Design and Implementation of WebGIS Point Symbol Database Based on SVG

Lele Han, Jianhua Xu, Zhihong Li, Cui Li

School of Geographic Sciences, East China Normal University, Shanghai  
Email: hanzhiyufeng@163.com, jhxu@geo.ecnu.edu.cn

Received: Oct. 18<sup>th</sup>, 2015; accepted: Nov. 13<sup>th</sup>, 2015; published: Nov. 17<sup>th</sup>, 2015

Copyright © 2015 by authors and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

---

## Abstract

Through analyzing the advantages and disadvantages of the current vector symbol database, the symbol production, upload and storage, retrieval, extraction and display of SVG vector point symbol database are discussed in detail. The new idea and method are given in the design and implementation of vector symbol database for Web, which has been used in mapping application of WebGIS platform—ECNUGIS with good practical effects.

## Keywords

SVG, Vector Symbol, WebGIS, Symbol Retrieval, Thinning Algorithm

---

# 基于SVG的WebGIS点符号库的设计与实现

韩乐乐, 徐建华, 李治洪, 李 翠

华东师范大学地理科学学院, 上海  
Email: hanzhiyufeng@163.com, jhxu@geo.ecnu.edu.cn

收稿日期: 2015年10月18日; 录用日期: 2015年11月13日; 发布日期: 2015年11月17日

---

## 摘 要

通过分析目前矢量符号库的优缺点, 对SVG矢量点符号库的符号制作, 上传与存储, 检索、抽稀与显示

到地图进行了详细讨论,并结合WebGIS平台软件ECNUGIS的制图应用,为Web端矢量符号库的设计与实现提供了新的思路和方法。

## 关键词

SVG, 矢量符号, WebGIS, 符号检索, 抽稀算法

## 1. 引言

地图符号是直观表达地理要素特征及其分布规律的地图图解语言。WebGIS 的产生为地理信息的存储、管理、分析、发布和共享提供了方便而有效的途径[1]。由于目前很难用同一种模式描述所有地图符号[2],各系统软件的差异也使得地图符号在规范化、可引用性与共享性等方面难以实现地图符号库的共享。因此,如何利用一种开放式的描述方法来绘制地图符号,并以此作为地理空间数据存储、传输和显示的基础,是目前 WebGIS 领域研究的基本问题[3]。

点符号通过不同形状的几何图形单元表示小面积地物或点状地物,一般位置固定,数量较多。SVG 是一种基于 XML 的二维矢量图形描述语言,可以较好的满足地理信息系统对于空间数据中图形显示的要求,并且为用户提供多样化且直观易懂的图形界面,从而成为空间数据可视化的有效格式。用 SVG 表现地图,其可缩放性、可定位特性、渲染速度优良等特性十分适合 Web 地图的浏览和检索。鉴于以上分析,本文以公共地理信息通用地图符号的国家标准对点状符号进行分类,并详细讨论 Web 端 SVG 矢量点符号库的符号设计、存储、检索与抽稀显示,以便为网络地图矢量符号的存储、传输、表现及共享提供研究基础。

## 2. SVG 矢量 GIS 符号库的优缺点

### 2.1. SVG 特点

SVG 是一种可缩放矢量图形,允许三种形式的图形对象存在矢量图形、栅格图像和文本。各种图形对象能够组合、变换,并且修改其样式,也能够定义成预处理对象。SVG 提供了丰富的图形对象,可以有有效的表现空间信息,如线<line>、矩形<rect>、圆形<circle>、椭圆<ellipse>、折线<polyline>、多边形<polygon>、路径<path>等预定义的形状元素,可被开发者使用和操作。所有定义的简单几何体都可以通过这些基本形状与路径得以显示表现,我们也可以通过定义形状的填充颜色<fill>、边框颜色<stroke>、边框宽度<stroke-width>、尺寸<width、height>、透明度<opacity>等属性构建更为丰富的图像。与其他图形格式相比,SVG 有着无可比拟的优势,尤其在网络应用方面来说,它具有支持用户与图形的交互、文档自我描述性、可检索性、图形对象可在分布式环境下创建而且和其它标准兼容性好的特点[4],目前标准浏览器在其内部直接支持矢量图形。

由于 SVG 是通过基于 XML 的文本语句来生成矢量格式图形,图形的大小只与文件中的坐标点数有关,即图形越复杂,用于描述数据的坐标点数越多,与图形的具体尺寸并没有关系。通常其数据量很小,并且支持 GZIP 等算法对其进行压缩,特别是当它经过压缩后,文件大小可减少 75%~80% [5],可以有效减小数据量,减轻网络传输负载。

### 2.2. SVG 矢量 GIS 符号库的优缺点分析

传统的 GIS 符号库的设计与实现是利用 True Type 字体库,该类型符号库是以 True Type 字体(True Type Font, 简称 TTF)的形式存储在 Windows 字库中。由于其字形描述准确、可无极缩放等特性,很多

制图软件(如 ArcGIS、MapInfo)在地图符号的表达上都采用 True Type 字体生成符号。在 ArcGIS 符号编辑的界面选择 Character Marker Symbol 类型,软件便会读取系统字库,其中就包含了所存储的字体符号;而在 MapInfo 中是通过文本输入工具来选择存有符号的字体样式。在 web 开发中,使用字体来制作 icon 应用的也越来越多。但是在使用字体图标时,我们需要借助 css 中内容生成的 content 参数来定义,而这需要复杂的代码才能实现;同时很难用一种统一调整的方法将其准确地显示在地图,而 SVG 能让图标引用及定位变得简单高效。

以前 SVG 矢量格式无法在 IE6、IE7 下有优秀的表现,可能不利于 SVG 矢量 GIS 符号库的发展,但随着高级浏览器对矢量格式的支持不断加强以及市场份额也不断增大(相关统计显示 IE 低版本浏览器的市场份额只占 6.4%左右),浏览器性能对 SVG 矢量技术应用推广更加有利。SVG 具有强大的空间数据表达能力,基于 SVG 组建的 GIS 矢量符号库,可以供 WebGIS 实现不同地图时进行重复引用[6],并且方便地通过代码定义图标尺寸、颜色以及其它的一些图形化功能,从而有利于实现地图符号库的共享,为不同 Web GIS 应用提供丰富的地图符号服务。

ECNUGIS 是华东师范大学具有自主知识产权的 WebGIS 平台软件,为用户提供便捷的在线地图编辑与专题地图制作。SVG 在地图表现中具有可缩放性、可定位特性、渲染速度优良等特性,同时 SVG 是客户端可操作的图形数据,利用客户端可控性来操作矢量数据能够实现部分 GIS 功能,给服务器减轻压力[7],因而我们在 ECNUGIS 平台上将其作为 WebGIS 空间数据可视化表达方案,设计与开发 WebGIS 点符号库,探讨 SVG 符号在 Web 端进行丰富、高效地可视化表现有一定的实际意义。

### 2.3. GIS 点符号库内容

国家标准(GB/T 24354—2009)将公共地理信息通用地图符号分为基础地理信息符号和公共场所与服务机构信息符号,前者作为地图表达的定位基础,其点符号主要包含居民点、交通、特殊建筑物等;后者作为表达公共地理信息的专题信息,其点符号包括政府机构、党政机关、科研教育、餐饮住宿等。同时结合互联网时代用户的贴近现实的信息可视化需求,我们为用户初步提供了基本符号、交通符号、建筑物符号、公共场所符号、天气符号、表情符号、我的符号七类。

## 3. SVG 矢量点符号库关键问题研究

### 3.1. SVG 点符号制作与存储

在 ECNUGIS 点符号库的设计中,我们利用 Illustrator 矢量制图软件,绘制了各类 SVG 符号,如图 1 所示。矢量图标是可以无限伸缩的,在图标引用时也可以通过 css 统一定义尺寸等样式。为便于制作规

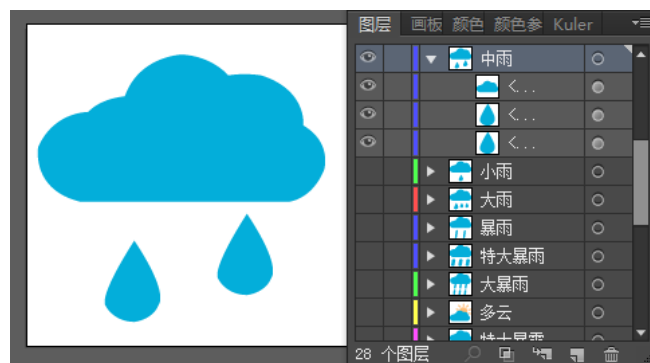


Figure 1. The organization of SVG point symbol in illustrator  
图 1. Illustrator 中 SVG 点符号的组织

范统一的符号库，我们定义的图标大小为 32 px\*32 px，选用 AI 提供的钢笔工具和基础图形来绘制。同时为了方便调用，我们对符号的不同组成部分进行编组，并通过命名对其进行属性标识。绘制完毕即可通过“文件”——“另存为”选项，将文件保存为 SVG 格式。

符号制作完毕后，就要考虑相当数量的不同种类的符号管理问题。这不仅对符号存储、符号读取与显示效率有很大影响，对于后期符号库的扩展与更新也极为重要。由于 SVG 也是一种 XML 节点的文件，首先为节省存储空间，我们借助 `svg-gui` 客户端将文件压缩。该客户端没有任何交互按钮，我们只需要将文件拖动到客户端界面，软件会自动压缩文件，并显示压缩前大小、压缩后大小、压缩比率，同时压缩前的文件会被自动替换。SVG 图标可以单独保存为一个 SVG 文件，为了节省 HTTP 请求次数，我们根据点符号库的分类将每一类的图标合并到同一个 SVG 文件。我们利用 `<symbol>` 元素来存放 SVG 图标，如果把一个 SVG 文件比喻成一个书柜的话，那么 `<symbol>` 就表示书柜中一本本不同类别的书籍，这些一本本不同类别的书本就是我们要使用的 `<symbol>` 图标。将多个 SVG 图标整合成一个 SVG 文件之后如图所示，可以看到代码中有不同类别的 `<symbol>` 元素，它们就是我们要引用的图标，并且每一个图标通过 ID 被引用到页面。由于符号库图标数量较多，手动合并相当数量的图标并不现实，因此我们使用用于处理 SVG Symbols 用的 `glup` 插件 `gulp-svg-symbols` 来实现自动合并。实现过程如图 2 所示。

### 3.2. SVG 符号上传

为更好地满足用户的需要，我们利用 HTML5 提供的 File API 实现上传图标的功能。Javascript 通过 File API 提供的 File、FileList、Blob 接口便可以读取本地文件。在 html5 中，File 对象支持选择多个文件，用户选择了某些文件之后，便会触发 file 类型的 input 元素的 onchange 事件句柄。File 对象中包含了文件的所有可访问信息，而不仅仅是文件名。示例的 HTML 代码如下所示：

```
<div id="browserFilesTxt"></div> //用来输入文件名
<div id='inputFileDiv'>
<span id="uploadSvgTxt">浏览</span> //用来打开文件选择对话框选择文件
<input type="file" id='uploadSvgFiles' multiple >
// multiple 属性指明了该 file 控件可进行多选操作
</div>
```

File API 还提供了一个异步读取文件的接口——`FileReader`，利用该接口我们可以异步地将文件内容加载到内存中，赋予某个 js 变量。`FileReader` 包含了一套完整的事件模型，用于捕获读取文件时的状态如 `onabort` (中断)、`onerror` (出错)、`onloadstart` (开始)、`onprogress` (正在读取)、`onload` (成功读取)等。

基于 `FileReader`，我们编写了用于多文件上传的 js 类库 `gUpload.js`，同时封装 `uploadSvg` 接口来实例化 `Upload` 类，实现本地文件的读取、变量的处理、文件上传至服务器。在上传图标文件时，我们建议用用户定义图标的关键词。同时以用户名 + 手机号码为每一个用户创建服务器中属于自己的图标目录，并将 `symID` (符号文件名)、`typeID` (用户目录名)、`symDescriptor` (关键词)三项录入符号库表 `g_symbol`，便于符号的检索和读取。

### 3.3. SVG 符号检索

为便于用户快速找到所需符号，我们为每一类图标、单独的图标都定义了描述符，并设计了符号库表 `g_symbol`，如表 1 所示。

用户在搜索框内输入所需符号的描述词语，如“天气”，便可得到所有与天气相关的符号，如果输入“云”或“多云”，则得到表示多云的特定符号。用户点击图标便可对其进行样式设置，我们通过

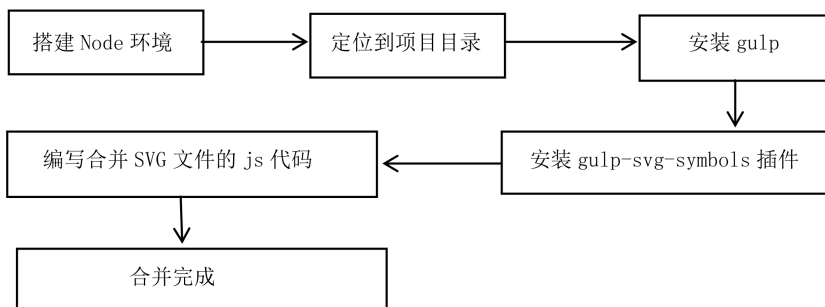


Figure 2. The flow-process diagram of merging SVG icons automatically  
图 2. 自动合并 SVG 图标流程图

Table 1. The index table for storing SVG symbol information

表 1. 存储 SVG 符号信息的索引表

符号 ID symID	所属类 ID typeID	所属类描述符 typeDescriptor	符号描述符 symDescriptor
0001-circle	basic	基本符号	圆形
0002-province_cap	basic	基本符号	省级行政中心
0001-cloud	weather	天气符号	多云、阴
0010-rain	weather	天气符号	中雨、降雨
0014-sunny	weather	天气符号	晴、太阳

符号 ID 由四位数字 + 表示符号属性的英文单词构成，作为数据库中每一个符号的唯一标识。所属类 ID 是符号所在子类的英文单词，同一子类包含若干符号。所属类描述符是对子类的描述所定义的用于检索的关键词。符号描述符是对每一个符号所定义的用于检索的关键词。

getSymValue 接口获取样式设置参数，并转换成如下形式：basic,#circle,#ffffff,0,#333333,30,90,1。参数定义依次为符号所属类 ID、符号 ID、边框颜色值、边框宽度、填充颜色值、尺寸、旋转角度、透明度，之后我们封装 update\_svgStyle 接口将参数存储到数据库的 svg\_style 字段中。

### 3.4. SVG 嵌入网页文档

通常，将 svg 文件嵌入 HTML 页面有以下主要方法：

1) 使用标签引用 svg。embed、object、img、iframe 标签等都被大部分或较新的浏览器支持，使用这些标签可以很方便地嵌入 svg。这种方法的缺陷是要求每个图标都必须是一个 svg 文件，这样不利于符号的分类与管理，而且在使用时也是单独请求，会增加 HTTP 请求次数而影响效率。

2) 使用内联 svg。即直接引用 svg 的命名空间，通过声明 SVG 元素、创建 SVG 图形、添加元素样式，我们就可以很方便地使用 SVG 图标。代码如下所示：

```

<html xmlns:svg="http://www.w3.org/2000/svg">
  <body>
    <title>circle icon</title>
    <svg xmlns="http://www.w3.org/2000/svg" width="400" height="300" viewBox="0 0 400 300">
      <circle cx="50" cy="50" r="50" fill="#448ed8" stroke="#999999" stroke-width="5" ></circle>
    </svg>
  </body>
</html>
  
```

由于内联 svg 内联 SVG 直接在文档中包含了所有的 XML 数据，因此不需要单独请求，对图标的修



改和更新也比较方便。缺点是需要在文档中嵌入大块代码，比较繁琐，不适合手写，图标也不易复用。

3) SVG sprite。类似于 CSS sprite，将图标整合在一起。我们可以使用 svg 中的<symbol>元素来制作 icon，SVG 本身的定义允许使用<use>的方式直接引用 SVG 中的某一部分，代码如下所示：

```
<svg class="icon">
<use xlink:href = "/svg-icon/assets/svg/transport.svg#0001-bike"></use>
</svg>
<svg class="icon">
<use xlink:href = "/svg-icon/assets/svg/transport.svg#0002-car"></use>
</svg>
<svg class="icon">
<use xlink:href = "/svg-icon/assets/svg/transport.svg#0003-taxi"></use>
</svg>
<svg class="icon">
<use xlink:href = "/svg-icon/assets/svg/transport.svg#0004-bus"></use>
</svg>
```

### 3.5. SVG 符号抽稀

根据点符号的地理坐标和样式参数将其显示到地图上时，通常会出现大量点符号密集显示的情况。在电子地图的符号化表达技巧中，利用点抽稀来表达兴趣点是我们经常用到的配图技术[8]。为了更清晰、均匀地呈现点符号，我们采用垂距限值法对点数据进行抽稀。垂距限值法的主要原理是从第一点开始依次筛选，去除冗余点。即以第一点为起点，计算第二点至第一点和第三点连线的垂直距离，若此距离大于某阈值，则保留第二点，并将其作为新起点，计算第三点至第二点和第四点连线的距离；否则，去除第二点，计算第三点至第一点和第四点连线距离，依次类推，直至最后一点。其阈值一般取相应地物最大允许误差或更小，该方法循环简单，易于编程处理，是一种较理想的抽稀算法[9]。本文算法流程如图 3 所示。

经测试，当阈值等于 SVG 符号宽度/10 时，抽稀效果较好。

基于 SVG 的 WebGIS 点符号库的目的是建立一个符合国家标准的矢量符号库，同时能够保证符号的矢量呈现，并且具有良好的成图效果。我们从 SVG 矢量符号的制作与存储、上传、检索、引用及抽稀显示五个方面详细讨论了有效的解决方案，这些方案的选取对符号库能否更便捷、更高效地应用以及对符号库后期的扩展与更新都起着决定性作用。

## 4. 应用分析

作为华东师范大学具有自主知识产权的 WebGIS 平台软件，ECNUGIS 为用户提供了在线创建编辑地图与制作专题地图功能，在地图符号表达方面，我们基于 SVG 设计和实现了矢量点符号库。用户使用自己的平台账号登录，进入空间数据管理界面，点击右侧图层面板中的点符号，就可以对符号尺寸、颜色、角度、透明度等进行配置；单击点符号，便会进入符号库界面，用户在此可以通过搜索、直接点击选择所需符号进而进行样式修改；如果平台符号库提供的种类不能满足需求，用户可以通过简洁的上传界面上传本地的矢量符号。

在 ECNUGIS 平台中，为分析符号抽稀显示的精度，我们还通过判断点符号最小外接矩形是否相交的方法来剔除重叠的符号，并将该方法的抽稀结果与垂距限值法进行了比较分析。在运算次数方面，假设地图视野内共有  $n$  点，最小外接矩形相交法的最大运算次数为  $n*n$ ，垂距限值法的最大运算次数为  $n$ ；在抽稀率方面，同一缩放级别下，两者的抽稀率分别约为 36%，74%。利用垂距限值法得到的结果如图 4 所示，可见后者的抽稀显示效果更为清晰、均匀。

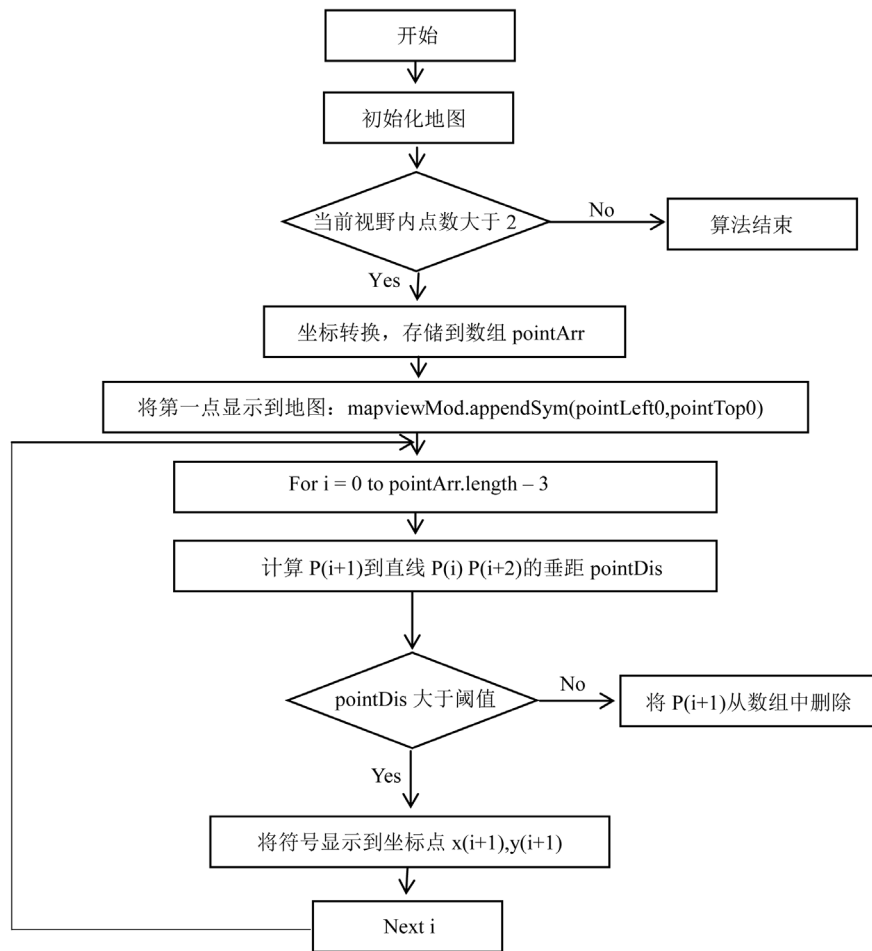


Figure 3. The flow-process diagram of thinning algorithm for vertical distance  
图 3. 垂距限值抽稀法流程图



Figure 4. The SVG symbol database on ECNU platform  
图 4. ECNUGIS 平台符号库应用

## 5. 结论

SVG 在地图表现中具有可缩放性、可定位特性、渲染速度优良等特性, 并且可以通过代码定义其图形化功能。在基于 SVG 的 WebGIS 地图数据可视化中, 地图符号可实现一次定义, 多次引用, 减少了 HTTP 请求次数, 并且可以通过有效方法实现 SVG 矢量符号合理地存储、检索与显示。SVG 应用于 ECNUGIS 平台软件中, 通过简洁、可控的交互操作, 在不同地图中表现丰富、高效的符号可视化, 为 Web 地图矢量符号库的设计与实现提供了具有实际意义的思路和方法。

## 参考文献 (References)

- [1] 李治洪. WebGIS 原理与实践[M]. 北京: 高等教育出版社, 2010: 10-12.
- [2] Emmanuel, S. (2002) Representation of Map Objects with Semi-Structured Data Models. In: *Symposium on Geospatial Theory, Processing and Applications*, Ottawa.
- [3] 王琤, 李霖, 尹章才. 基于 SVG 的面状地图符号设计与实现[J]. 测绘科学, 2007(3): 168-170.
- [4] 韩双旺, 周珑, 李德录, 崔兆顺, 马海云. 基于 SVG 的矢量 WebGIS 地图符号库的构建与引用[J]. 自动化与仪器仪表, 2010(2): 50-53.
- [5] 冯艳杰. 基于 SVG 的 WebGIS 实现技术[D]: [硕士学位论文]. 武汉: 武汉大学, 2005.
- [6] 彭海龙, 邹彬, 劲岩, 于福江. SVG 在海区信息 WebGIS 中的应用研究[J]. 海洋预报, 2005, 22(1): 5-10.
- [7] 解永青. 基于 SVG 的矢量 WebGIS 性能优化方法研究[D]: [硕士学位论文]. 合肥: 安徽农业大学, 2012.
- [8] 王屏, 宋昊, 刘淑红. 基于点距离的点抽稀算法在电子地图符号化中的应用[J]. 科技与企业, 2014(17): 311.
- [9] 刘彦花, 叶国华. 矢量曲线抽稀算法分析[J]. 城市勘测, 2001(4): 49-52.