

用网络矢量图像 SVG 技术实现网络监控图形的显示^①

Graphics Display of WEB Monitoring and Control Software Based on SVG

王志梅 (上海交通大学 计算机科学与工程系 200030)
(温州职业技术学院计算机系 325035)

田启明 曹月琴 (温州职业技术学院计算机系 325035)

摘要:本文介绍了一种新的网络矢量图形格式 SVG 及其优越性,通过分析监控软件的基本结构及其人机界面需求,结合网络矢量图像 SVG 规范提出了适合监控系统软件的 SVG 图形显示方法。

关键词:SVG 监控软件 图形显示 XML

1 引言

SVG 是可任意矢量图像模式 (Scalable Vector Graphics) 的简称。它基于 XML, 是一个全新的标准开放的矢量图像和动画格式。使用 SVG 可以在网页上显示出各种各样的高质量矢量图形, 支持众多的功能。但不同于传统的二元 (平面) 图像, SVG 使用简单的一元 (文本) 语句完成矢量图像, 以及诸如色彩变化、对象运动、滤镜和遮照、音效添加等各种动画效果。它完全是用普通文本来描述的, 也就是说 SVG 是一种专门为网络而设计的基于文本的图像格式。SVG 这种基于纯文本的特性, 使它具备了像文字信息一样的可检索性, 从而 SVG 中的文字可以被网络搜索引擎做为关键字搜寻, 可以真正实现基于内容的图像搜索引擎, 从而使人们对 WEB 图形检索的梦想成为现实。

对于目前正面临着种种难题的 Internet 及其应用来说, SVG 的诞生有着十分积极的意义, 其应用前景十分广阔。短小灵活的矢量图形是缓解 Internet 网络带宽日益紧张的有效手段。SVG 在本质上是 XML (可扩展标识语言) 的一个应用, 它与当前已有的和未来将有的其它 XML 应用之间存在着天然的交互和融合, 从而使这些前程远大的技术和应用之间的协作变得相对简单。

2 基于 XML 的 SVG 的优势

SVG 的语法和结构是基于 XML 的。以下为一个简

单的 SVG 文档代码示例, 该文档指示浏览器创建一个矩形。

```
< xml version = "1.0" standalone = "no" >
< ! DOCTYPE svg PUBLIC " -//W3C//DTD SVG1.0//EN"
http://www. w3. org/TR/2001/REC - SVG -
20010904/DTD/svg10. dtd >
< svg width = "400" height = "150" xmlns = "ht-
tp://www. w3. org/2000/svg" >
< rect x = "30" y = "15" width = "300" height =
80" fill = "red" stroke = "blue" stroke - width = "5"/
>
< /svg >
```

由此示例可知, SVG 文档本质上就是规范的 XML 文档, 其语法和结构完全基于 XML。

XML 提供的功能远远超过了 HTML (超文本链接语言)。与 HTML 不同, XML 完全分离了网站和网站构架。通过定义结构数据类型, 用户端应用程序能够显示和处理, 而不仅仅像 HTML 那样仅仅能显示网页或数据。例如, 一个利用 XML 标签定义的电话号码, 可以根据需要由浏览器进行拨号。由于 SVG 的开发和创作是基于 XML 的, 除了能绘出矢量图像外, 使得也能基于 SVG 开发出更多新功能以提供更多的网络服务, 如制作智能化数据图像, 图像中的数据可以根

^① 此项目为浙江省教育厅资助项目 (20050335)

据需要,由应用程序读取、修改和统计并最终在图像中显示,这些应用是目前 HTML 及相关图像技术远不可及的。

SVG 图像还可以方便的由程序语言来动态的生成。例如使用 JavaScript, Perl, JSP, ASP 等,下面就是一个利用 JSP 动态生成简单的矩形 SVG 图像的例子:

```
< % @ page contentType = " image/svg - xml "
% >
< %
int width = 300 , height = 200 ;
String strcolor = " #eeeeff "
% >
< svg width = " < % = width % > " height = " <
% = height % > " >
< rect x = " 10 " y = " 20 " width = " 150 " height = "
70 " fill = " < % = strcolor % > " stroke = " red " stroke
-width = " 1 " / >
< /svg >
```

3 监控软件图形显示的基本需求分析

监控软件人机界面的基本需求包括:将监控对象的各种信息展示给用户;对各种监控对象的配置信息进行管理;通过各种监控对象的操作接口实现对监控对象的操作控制。

人机界面的发展趋势是网络化,即在局域网甚至 Internet 上通过一定的人机界面实现对监控对象的监控。基于网络的人机界面的基本结构模式一般都采用 B/S 模式。B/S 模式易学易用易管理,并且具有信息共享度高、扩展性好、安全性好、广域网支持等优点,必将成为监控软件的主流模式。基于 WEB 的监控软件人机界面的需求信息交互形式与基于 B/S 模式的监控软件一样,主要包括了集中管理、表单、列表和图形等形式,除了图形形式在 WEB 上实现有一定难度外,其余形式的实现相对比较简单,技术也比较成熟。WEB 图形的实现方式从客户端图形显示方式上分为客户端图形显示控件和客户端 Script 程序显示两类,客户端图形显示控件依赖于客户端操作系统对窗体图形的支持,客户端 Script 程序显示方式依赖于 WEB 浏览器在图形方面的支持;从图形生成方式上分为服务器端生成和客户端生成两类,服务器端生成是不通过 WEB 浏

览器实现的传统方式,客户端生成方式与客户端图形显示方式一样分为两类。本文主要探讨的是基于 Script 程序的客户端图形显示方式。

监控软件的图形显示应包括:

基本图形显示:线段、折线、矩形、圆及文字等。

公共图形属性:填充颜色、方式、透明度、线的颜色、线型、宽度等。

图形功能:图库、成组、坐标变换、分层显示等。

动画:颜色变换、大小变换、位置变换、旋转变换等。

交互:能在客户端正确区分鼠标动作对象。

4 SVG 的图形规范

SVG 文档需要遵循一些简单的规则才能成为有效的文档。SVG 基于 XML,但自己也定义了一套元素,这些元素表示类似形状和线条的这样的对象。最基本的要求是,SVG 文档必须以 < svg > 元素开始。SVG 文档也可以指定为文档片段,也就是说,一个 SVG 文档可以嵌入在另一个 XML 文档内。但是通常将 SVG 文档指定为单独的文档,其扩展名为 .SVG。

4.1 画规则图形

表 1 列出了 SVG 定义的常见规则形状的元素和属性,其中 style 属性指定元素样式,如填充色和边框色,该属性是可选的。

4.2 画不规则图形

SVG 规范定义了 < path > 元素一点一点地画不规则形状的轮廓,使用 < path > 不仅可以画出以上规则图形,还能画出任意的二维曲线。< path > 元素在它的 d 属性上指定一套指令,由这些指令来表示要画的路径。这些指令的形式都是一个字符后跟一个数值。

下面给出一个使用曲线指令画线的例子:

```
< svg style = " height : 200px ; width : 200px " >
< path d = " M0 , 200 L100 , 0 L200 , 2000 Z " >
< path d = " M100 , 200 C100 , 100 250 , 100 , 250 ,
200 " style = " fill : none ; stroke : black ; " / >
< /svg >
```

另外,SVG 还具有内置的五种动画类型,在 SVG 文档中,可以使用组合功能将多个对象组合在一起,并为动画和效果提供引用对象。整个形状的集合可以作为一个实体进行移动、定位、样式化和复制。

表 1 SVG 规则形状的元素和属性

形状	元素	常用属性	说明
矩形	rect	width height x y style	x y 为矩形左上角的位置
直线	line	x1 y1 x2 y2 style	(x1, y1) 为起点坐标, (x2, y2) 为终点坐标
圆形	circle	cx, cy, r, style	(cx, cy) 为圆心坐标, r 为圆半径
椭圆	ellipse	cx, cy, rx, ry, style	(cx, cy) 为圆心坐标, rx 为 X 轴半径, ry 为 Y 轴半径
多边形	polygon	points, style	Points 按划线次序列出多边形各点坐标
折形	polygon	points, style	如要填充, 折线应闭合
文字	text	x, y, font - family, font - size letter - spacing	x, y 为文字左下角的 x, y 坐标, font - family 为字体, font - size 为字体大小 letter - spacing 为字符间距

5 基于 SVG 技术的监控图形显示

SVG 具有完整而丰富的静态图形标记集和强大的修饰参数功能, 完全能满足监控软件的可编辑性, 精选适合于监控系统的 SVG 静态图形标记集, 修饰参数则采用直接属性标记, 摒弃 SVG 的 < style > 标记及 css 定义。

5.1 基本图形显示

如表 1 所示, 去掉所有的 < style > 标记, 便可方便地显示这些基本图形。

对表 1 中没有包含的基本形状如椭圆弧、扇形等, 由于 SVG 中没有专门的元素, 但可以使用功能丰富的 < path > 元素来实现它们的显示。

5.2 公共图形属性

对于任意的 SVG 图形, 只要具备相应的修饰显示属性, 都可以利用 SVG 的修饰参数加以定义并显示。

对于封闭图形: fill 修饰填充颜色及填充方式; fill - opacity 修饰填充的透明度。

对于线(边线): stroke 修饰线的颜色; stroke - dash - array 修饰边线的线型; stroke - width 修饰边线线条的宽度等等。

5.3 图形功能

(1) 图库。图库功能对于监控软件是必需的。表示同一类型的设备等目标采用相同的图形元素组合, 不但能方便地实现图形绘制与显示, 还可以简化图形文件。SVG 的 < defs > 框架中提供了预定义的模板元素 < symbol >, 其所有子元素构成一个图符, 多个 <

symbol > 元素构成图库。使用 < use > 引用元素实现对图符的应用。一般情况下是将图库与监控图形一起保存在服务器上, 以服务器存取技术作为辅助手段, 保证监控图形的一致性。

(2) 坐标变换。默认的坐标系统以用户代理(通常为 WEB 浏览器)左上角的点(0,0)为原点, 其正的 X 轴向右, 正的 Y 轴向下, SVG 可以方便地创建新的坐标系统。在监控软件中, 对一个已经绘制好的图形元素进行坐标变换是一种常见的需求。SVG 提供了专门的坐标变换属性 transform, 用于修饰任意的图形实体元素, 实现所属元素的移动、旋转、伸缩、歪斜等变换功能。

(3) 动画。动画一般通过更改对象属性值来实现, SVG 也不例外, 但实现 SVG 实时动态图形还有一种方式, 那就是通过 SVG 动画元素由 SVG 插件自动实现动画。更改元素属性值, 其主要的思路是利用程序增减元素或改变任意的元素的任意属性, 这也是实现监控系统软件动画的主要方式; 通过 SVG 动画元素由 SVG 插件自动实现动画, 对于非再编辑的动画有比较好的显示能力, 很多动画元素的显示效果是比较复杂, 精选一部分动画元素作为监控系统软件动画的必要补充, 可充分体现 SVG 的优越性能。结合这两种方式, 可以充分满足监控系统的应用需求, 提供一个比较完美的动画界面。根据监控显示对象的状态实时启停 SVG 专门的动画元素可实现理想的动画效果。

SVG 定义了专用于动画的 5 种元素, 即 animate,

animateMotion, animateColor, animateTransform 和 set。下面介绍其中的两种。

① < animate > 元素: 在声明的一段时间内将指定的特性从一个值变化到另一个值, 下面的代码使圆的半径在 10 秒内从 1 变为 50。

```
< svg width = "500px" height = "500px;" >
  < circle cx = "200" cy = "200" r = "1" fill = "red;" >
    < animate attributeName = "r" from "1" to "50"
dur = "10s" repeatCount = "indefinite" >
  < /circle >
< /svg >
```

② < animateMotion > 元素: 沿一条路径移动一个对象, 这条路径可以是直线, 也可以是曲线。下面的例子是一个矩形沿一条曲线边移动边旋转的动画。

```
< svg width = "500px" height = "500px;" >
  < rect x = "10" y = "10" width = "20" height = "20" >
    < animateMotion dur = "4s" repeatCount = "indefinite" path = "M 100,250 C100,50 400,250" rotate = "auto" / >
  < /rect >
< /svg >
```

5.4 交互性

SVG 可以通过捕获某些事件来启动脚本以实现交互性。由于 SVG 基于 XML, 因而能够制作出空前强大的动态交互。SVG 图像能对用户动作做不同的响应, 例如高亮、声效、特效、动画等。在客户端支持鼠标动作支持是系统交互的必需功能。SVG 为所有的实体元素提供了客户端的鼠标动作事件支持, 能在客户端区分鼠标动作对象。这些动作包括: 获得焦点 (onfocusin)、失去焦点 (onfocusout)、点击 (onclick)、鼠标压下 (onmousedown)、鼠标释放 (onmouseup)、鼠标移进 (onmouseover)、鼠标移动 (onmousemove)、鼠标移出 (onmouseout) 等。

下面部分代码段 (交通信号监控) 以鼠标单击图形监控界面的交互示例来说明 SVG 交互功能的实现:

```
< xml version = "1.0" encoding = "Utf - 8" >
```

```
< svg xmlns = http://www. w3. org/2000/svg
width = "600" height = "600" >
  < script language = "javascript" > < ! [CDATA
[
  function openpage( roadno ) {
    open ( " Object/oneRoadInfo. asp RegNo =
1&RdNo = " + roadno );
  ] ] > < /script >
  .....
  < g id = " road - cross" >
    < ! -- 处理鼠标点击事件 -- >
    < g onclick = " road15" cx = "155" cy = "15" r = "
10" style = " fill: url( #road - color) " / >
  < /g > < /g >
< /svg >
```

6 结语

由于 XML 与许多程序语言完美的结合, 使得 SVG 图像可以方便地由程序语言 JavaScript, Perl, JSP, ASP 等动态生成。通过以上分析说明, SVG 图形完全能满足监控系统软件对图形显示的要求, 随着 SVG 的普及, 基于网络矢量图形 SVG 的监控软件具有很好的发展前景。

参考文献

- 1 Scalable Vector Graphics(SVG)1.0 Specification [EB/OL], <http://www.w3.org/TR/SVG/>, 2001.
- 2 侯宇、李素有, 基于 XML 的 SVG 技术及其应用, 计算机应用研究, 2002(5)。
- 3 Simple Realization and Introduction of Threadpool [EB/OL], <http://www-900.ibm.com>.
- 4 魏应彬, 用 XSP 将 XML 文档数据转换成 SVG 图像, 福建电脑 2003(11)。
- 5 黄晓文, 基于 XML 的网络矢量图像 SVG 的实现与应用研究, 上海电力学院学报, 2003。
- 6 Mike Jasnowski. Java, XML 和 WEB 服务宝典, 北京电子工业出版社, 2002。