

基于 Java 和 SVG 的监控画面图形生成软件的设计

杨丰萍, 冯春华, 王健

(华东交通大学电气与电子工程学院, 江西 南昌 330013)

摘要: SVG (Scalable Vector Graphics) 是基于 XML (Extensible Markup Language) 的可扩展二维矢量图形语言, 作为 IEC61970 协议推荐的监控画面的标准图形格式, 拥有其他图形格式不具备的优点。如何开发一套 SVG 图形格式的监控画面图形生成软件, 是人们关注的问题。在阐述电力图形系统的发展现状及 SVG 优点的基础上, 采用 Java 为开发语言, 以 SVG 作为监控画面的图形格式, 介绍了监控画面图形生成软件图元库模块和图元属性模块的设计方法。

关键词: SVG; 图元库; 图元属性; Java; XML; IEC61970

Design of a software of drawing monitoring graphics based on Java and SVG

YANG Feng-ping, FENG Chun-hua, WANG Jian

(College of Electrical and Electronic Engineering, East China Jiaotong University, Nanchang 330013, China)

Abstract: SVG (Scalable Vector Graphics) is a language for describing two dimensional graphics in XML. SVG is superior to other format as a kind of graphics format which is recommended by IEC61970. How to design a software of drawing monitoring graphics using SVG has become what the researchers pay attention to. This paper introduces how to use Java and SVG to design graphic elements database and its properties of a software that creates monitoring graphics through introducing the present condition of power graphics system and the advantage of SVG.

This project is supported by National Natural Science Foundation of China (No.50474020).

Key words: SVG; graphic elements database; graphic elements properties; Java; XML; IEC61970

中图分类号: TM73; TP312 文献标识码: A 文章编号: 1003-4897(2008)03-0032-05

0 引言

SCADA (Supervisory Control And Data Acquisition) 系统主要由调度端、通信通道和 RTU (Remote Terminal Unit) 三部分组成。调度端软件由实时数据库、通信程序和用户图形界面程序组成。其中, 用户图形界面直接联系着系统调度员和监控的设备, 它通过判断实时数据库里对应设备的状态, 以图形的方式实时地呈现给调度员, 以便调度员及时掌握设备参数、运行状态, 其重要性不言而喻。

目前, 大部分 SCADA 系统的监控画面采用的是 DWG 格式或者自己定义图形格式, 然后利用图形 API 进行图形编程^[1]。使用这种方式, 编程相对复杂, 开放性、可扩展性差。电力系统的发展需要推出一种标准来避免重复的劳动和不同厂商图形格

式之间的不兼容^[2]。SVG (Scalable Vector Graphics, 可升级矢量图像) 是 W3C 组织为适应 Internet Web 应用的飞速发展需要而制定的一套基于 XML 语言的可缩放矢量图形语言描述规范^[3], 它具有任意缩放、文件尺寸较小、适合网络传输^[4]、超强的显示效果和色彩控制等特性。由于它具有得天独厚的优势, IEC61970 协议推荐将 SVG 作为监控画面的标准格式^[5]。本文采用 Java 为开发语言, 以 SVG 作为监控画面的图形格式, 介绍了监控画面生成软件图元库模块和图元属性模块的设计方法。

1 图元库模块设计

本文通过设计一个父类 SVGShape 来定义所有针对图元的操作方法, 每一类图元都要通过继承父类 SVGShape 来实现, 这样就继承了父类 SVGShape 对图元的操作方法, 提高了代码的复用率, 使得设计思路更加清晰, 降低编码和维护的工作量。

1.1 父类 SVGShape 的设计

基金项目: 国家自然科学基金项目 (50474020)

从监控画面生成软件中生成设备图元, 调度员都要对设备图元经过一番转换操作, 例如拖拽放大、旋转, 有时还需要同时选中多个图元, 以便统一进行操作。

这些操作是每类图元所共有的操作方法, 而不是针对某一类图元。本文根据这一事实为模型, 抽象出一个各类图元共有的父类 SVGShape, SVGShape 类中定义了针对图元的所有操作方法, 每类图元在实现时都要通过继承父类 SVGShape 来实现。以下就是一些本文对主要操作方法的设计。

translateNodes() 方法中定义了图元平移 (translate) 的方法。通过点击鼠标左键产生一个变换的轮廓框架, 然后根据计算这个轮廓框架移动的坐标锁定图元最后的位置。

resizeNode() 方法中定义了将图元利用鼠标拖拽以变换其尺寸大小的方法。在缩放图元时, 首先应用鼠标选中图元, 然后通过点击鼠标左键, 根据想要缩放的方向 (东、南、西、北、东南、西北、东北、西南) 拖拽鼠标, 从而产生一个变换轮廓框架, 待鼠标停止拖拽时, resizeNode() 方法便通过计算原有图元和轮廓框架之间的位置, 从而得到拖拽后图元的大小。

rotateSkewNode() 方法中定义了图元旋转的方法。同图元变换尺寸相似, 旋转图元也是根据计算图元和变换轮廓框架之间的位置, 从而确定图元最后的旋转位置的。

平移、尺寸变换、旋转这三种对图元的操作, 在 SVG 文件中实际上就是对应着属性 transform 的调整。transform 定义了 SVG 图元转换的属性, 包括平移 (translate)、旋转 (rotate) 等操作。本文定义的平移、尺寸变换、旋转, 都用到了 SVG 规范定义的位置矩阵 (matrix) 作为这三种操作的算法。matrix 中定义了六个数值, 它们代表了一个 3×3 的矩阵:

$$\text{matrix}(a, b, c, d, e, f) = \begin{pmatrix} a & c & e \\ b & d & f \\ 0 & 0 & 1 \end{pmatrix} \quad (1)$$

平移、尺寸变换、旋转这三种操作的设计思想都是通过鼠标点击产生一个变换轮廓框架, 然后根据图元原有的位置和形状比较后, 根据公式 (1) 转换图元的位置或形状, 最后在 SVG 文件中以添加标签:

$$\text{transform} = \text{"matrix}(a, b, c, d, e, f)\text{"} \quad (2)$$

以完成对图元的转换操作。

1.2 基本图元的实现

对于各基本图元的生成, 本文根据 SVG1.1 规

范中关于各类图元地定义, 分别在各图元类里定义了如何生成相应的基本图元。本文设计从图元库工具条中拖拽相应的图元到画布上以完成相应图元的生成, 这就大量地用到了 Java 中关于鼠标事件的操作, 如鼠标拖拽 (mouseDragged)、鼠标移动 (mouseMoved)、鼠标按键 (mousePressed)、鼠标释放 (mouseReleased) 等。基本图元包括线段 (line)、圆 (circle)、矩形 (rect)、文本 (text)、折线 (polyline)、多边形 (polygon)、路径 (path)、椭圆 (ellipse)、图片 (image)。以下就是本文关于圆这类图元的实现。

SVG1.1 规范中对于一个圆定义了三个属性^[6]: cx 、 cy 、 r 。 (cx, cy) 代表了圆心的坐标, r 代表了圆的半径。图 1 所示的就是一个圆图元及其属性。



图 1 圆图元及其三个属性

Fig.1 Circle graphics element and its three attributes

在类 SVGCircle 的 drawCircle() 方法中实现了圆的生成。drawCircle() 方法中有两个参数 frame 和 bounds。frame 定义了 circle 图元所在的容器, 而 bounds 是圆外围的矩形界限, 通过公式 (3) 换算出 cx 、 cy 和 r 。

$$\begin{aligned} cx &= \text{format.format}(\text{bounds.x} + \text{bounds.width} / 2) \\ cy &= \text{format.format}(\text{bounds.y} + \text{bounds.height} / 2) \\ r &= \text{format.format}(\text{bounds.width} / 2) \end{aligned} \quad (3)$$

此方法中, 在相应的 SVG 文件中生成了 <circle> 元素, 以及 cx 、 cy 、 r 三个属性, 定义了默认的 style 属性。在 drawFigure () 方法中, 本文定义了鼠标释放后会生成的 circle 图元的事件。

1.3 电气元件图元的实现

如今的画图工具有很多种, 但是专门的电力图元画图工具却很少, 而且也不是基于 SVG 格式的, 为此开发一个基于 SVG 格式的电力设备元件库有很重要的意义。一个 SVG 电力设备图元可用元素组 <g> 表示, 但是用组表示代码可重复性不高; 也可用元素符号 <symbol> 定义, 然后通过 <use> 引用表示, 这种方法有利于实现图像元素的重用, 提高代码利用率。电气元件图元主要包括双卷变压器、三卷变压器、高压负荷开关、电流互感器、接地、避雷器、电容器, 自动重合闸等。下面的语句就描述了如图 2 所示的一个用符号 <symbol> 定义的变压器图元:

```
<symbol id="sbyq" viewBox="0 0 60 60"
preserveAspectRatio="xMidYMid meet">
<circle cx="9" cy="20" r="8"
```

```

style="fill:none;stroke:green"/>
    <circle cx="9" cy="31" r="8"
style="fill:none;stroke:green"/>
</symbol>

```

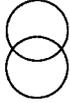


图2 变压器图元

Fig.2 A transformer component

<symbol>元素并不直接出现在画面上,而是通过在图形中对图元的具体引用后才会显示,其代码如下:

```
<use x="60" y="60" width="30" height="30"
xlink:href="#sbyq"/>
```

其中 xlink:href="#sbyq"正是对上述定义图元的引用。

SVG1.1 规范中对于一个<use>引用定义了几个属性: x、y、width、height。这四个属性用来确定<use>元素所引用元素的实例在给定坐标系中的矩形渲染区域的位置。<use>元素相当于将被引用元素的内容复制一份到当前位置,这就要用到 xlink:href 属性,本文通过绝对路径来引用符号定义文件的 id 值。

本文在类 SVGSymbol 中的方法 drawSymbol () 中实现了电力图元的创建。此方法中有三个参数: frame、bounds 和 symbolPath。其中 frame 是电力图元的容器, bounds 代表了外围界限矩形,而 symbolPath 是电力图元符号定义的 id 值。

在drawSymbol ()方法中实现在相应的SVG文件中生成<use>元素,以及x、y、width、height、xlink:href属性的代码如下所示。在drawFigure ()方法中,完成了鼠标在释放后生成的电力图元的事件。

// 生成use元素

```

final Element
use=doc.createElementNS(doc.getDocumentElement()
.getNamespaceURI(),"use");
use.setAttributeNS(null,"x",
format.format(bounds.x));
use.setAttributeNS(null,"y",
format.format(bounds.y));
use.setAttributeNS(null,"width",
format.format(bounds.width==0?1:bounds.width));
use.setAttributeNS(null,"height",
format.format(bounds.height==0?1:bounds.height));
use.setAttributeNS("http://www.w3.org/1999/xlink",
xlink:href,"file:/E:/dianlifuhao.svg#+symbolPath);

```

//将use元素添加到SVG根节点元素下

```
parent.appendChild(use);
```

1.4 功能模块的加载

本文采用了自定义 XML 配置文件的方式加载各功能模块。XML 是一种可以自定义标签,适合于数据交换的一种标准文本格式文件^[7]。利用成熟的 Java 开源产品 DOM4j、JDOM 可以实现对 XML 文件的处理,跨平台性好。修改配置文件中的配置信息时,只需要一个普通的文本编辑器就可以实现对配置信息的修改,而无需安装其它软件。采用 XML 作为配置文件的文件格式,不但使软件结构更加清晰,而且更重要的是使软件的可扩展性增强。增加或减少一个功能模块,只要在自定义的配置文件增加或者屏蔽一条配置信息就可以实现,增减模块对其他功能模块不会造成影响。

如下所示的是一部分功能模块在 modules.xml 配置文件中的配置信息。第一行是声明 XML 的版本号,第二行说明的是配置文件的配置信息 <modules>,从第三行作者开始定义每个功能模块的配置信息,每个功能模块的配置信息均以<module> 为一个节点, class 后面就指明该功能模块的位置,然后通过类 ModulesManager 来加载功能模块。

```

<?xml version="1.0" ?>
<modules>
  <module
class="scada.cn.com.svgeditor.shape.SVGRectangle"
/>
  <module class="
scada.cn.com.svgeditor.shape.SVGCircle" />
  <module class="
scada.cn.com.svgeditor.shape.SVGEllipse" />
  <module class="
scada.cn.com.svgeditor.shape.SVGLine" />
  <module class="
scada.cn.com.svgeditor.shape.SVGText" />
  <module class="
scada.cn.com.svgeditor.shape.SVGPolyline" />
  <module class="
scada.cn.com.svgeditor.shape.SVGPolygon" />
  <module class="
scada.cn.com.svgeditor.shape.SVGPath" />
  <module
class="
scada.cn.com.svgeditor.shape.SVGSymbol" />

```

ModulesManager 充当着管理员的角色。其中, parseXMLModules()是本类中最重要的一个方法,通过此方法可以读取解析 XML 配置文件 modules.xml,从而达到加载各功能模块的目的。此方法的活动图

如图 3 所示。

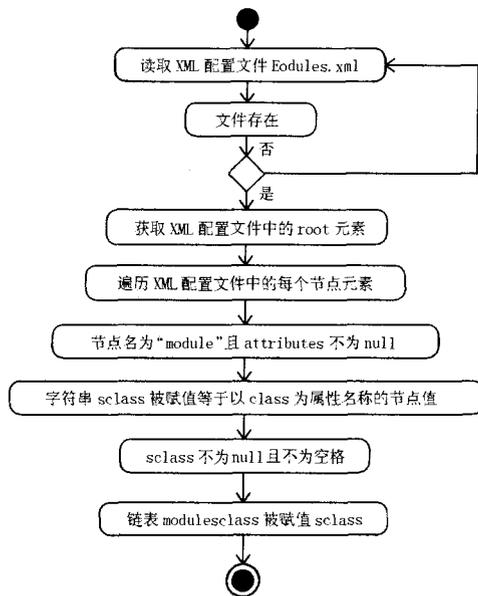


图 3 parseXMLModules() 方法的活动图

Fig.3 Activity diagram of parse XML modules() method

从上面的活动图可以看出, 通过 parseXMLModules() 方法解析配置文件后, 每个功能模块的配置信息被保存在链表 (LinkedList) modulesclass 中, 然后通过方法 createModuleObjects() 创建功能模块对象, 最后这些对象被保存到链表 modules 中, 从而加入到绘图平台模块。

2 图元属性设计

图元属性包括图元几何属性和图元数据库属性。图元几何属性包括图元的 id 号、位置、笔画及填充颜色等。图元数据库属性包括该图元与图库的关联、显示配置及实时对象等信息。

2.1 图元几何属性编辑模块

在监控画面生成的过程中, 需要对监控画面上的一些图元的几何属性 (如: 轮廓颜色、填充颜色、线条宽度, 几何位置等) 进行实时编辑, 使得监控画面美观, 更加方便调度员对设备图元的观测。本文单独设计了一个图元几何属性编辑功能模块来满足对图元几何属性的实时编辑需求。

通过分析, 本文认为: 在监控画面生成软件里对图元几何属性的编辑, 反映到 SVG 文件里就是对对应的标签进行编辑。代码

```
<circle id="c1" cx="50" cy="50" r="30" style="stroke-width:5.0;stroke:red;fill:none;" />
```

是一个圆心坐标为 (50, 50)、半径为 30 个像素、无填充色、边线宽度为 5 个像素且为红色的圆 (circle)

在 SVG 文件中对应的标签。

我们要编辑圆的各项几何属性时, 实际上是对 SVG 文件里 SVG 图元的属性 (attribute) 进行编辑。例如我们要将圆的半径修改为 20, 在监控画面生成软件中修改后反映到 SVG 中就是 $r=20$; 这样我们就建立了图元几何属性编辑器的模型。

本文将图元几何属性编辑模块分成几个小的工具包 (Widget), 每个 Widget 实现一项编辑功能。作者分析了各项图元属性编辑功能, 统一抽象出一个父类 SVGPropertiesWidget, 每个 Widget 都要通过继承父类 SVGPropertiesWidget 来实现。通过继承父类 SVGPropertiesWidget 来实现编辑每个 Widget, 不但可以提高代码的复用率, 而且使得设计思路更加清晰, 降低编码和维护的工作量, 更好的体现了面向对象的设计思想。

最后, 每个编辑 Widget 都放置在相应的标签页上, 从而实现了图元几何属性编辑功能模块。图元几何属性编辑功能模块的截图见图 4。对图元的某种几何属性要修改时, 只要点击相应的标签页进入到相应的编辑选项中, 就可以完成修改。

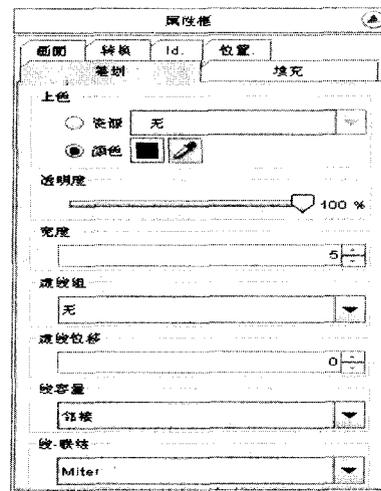


图 4 图元几何属性编辑功能模块的截图

Fig.4 Picture of graphics elements geometrical attributes editing functional module

2.2 图元数据库属性编辑模块

图元数据库属性模块主要完成图元在数据库配置, 同时它也可以帮助调度员读取各图元在数据库中的配置, 对图元在数据库中的配置具有读写功能。监控画面中的状态量图元、状态量派生图元、模拟量图元、脉冲量图元和链接性图元在数据库中有配置, 其中前四者属于实时图元, 在监控画面中需要实时显示。

本文采用了 Java 中最常用的 JDBC 来建立和数

数据库的连接,对于实时图元,它在数据库中涉及到三个表:图元种类表(实时数据库)、图元关联配置表(图元配置数据库)和图元显示配置表(图元配置数据库);对于链接性图元,它只是在图元关联配置表和图元显示配置表中有配置。本文使用了MySQL4.1中的SQL语言来实现对数据库的操作。通过Java Swing中的JTabPane实现了标签页显示功能。分四个标签页来分别显示实时图元和链接图元在数据库里的配置窗口。其中,实时图元在数据库中涉及到的三个表,本文把在每个表中的数据库配置选项做成一个标签页,而单独将链接图元做成一个标签页,功能划分明确。图元数据库属性配置模块如图5所示。

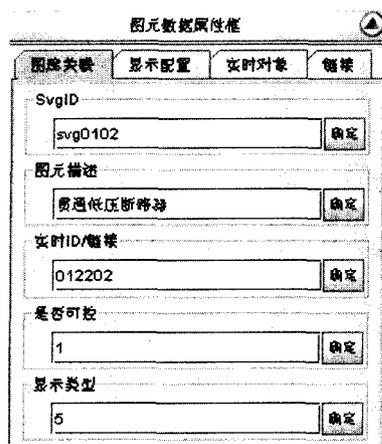


图5 图元数据库属性编辑功能模块的截图

Fig5 the picture of graphics elements database attributes editing functional module

3 结束语

SCADA 监控画面采用 SVG 作为标准图形格式是必然趋势。本文介绍了基于 SVG 的监控画面生成软件中图元库模块和图元属性模块的设计与实现方法,用<symbol>定义电气元件,可以提高代码利用率,并分别定义图元的几何属性与数据库属性,使图元在数据库中的属性配置一目了然。

参考文献

- [1] 高玉柱, 蒋士林, 陈剑云. 使用 Java 和 svg 构建远动系统实时监控画面模块[J]. 工业监控计算机, 2004,17(5): 37-49.
GAO Yu-zhu, JIANG Shi-lin, CHEN Jian-yun. Using

Java and SVG Constructing Real Time Picture Module of SCADA System[J]. Industrial Control Computer, 2004, 17(5): 37-49.

- [2] 刘遵雄, 况志军, 高玉柱. 基于 SVG 的电力图形系统的实现[J]. 继电器, 2005, 33(21): 69-73.
LIU Zun-xiong, KUANG Zhi-jun, GAO Yu-zhu. Realization of Power Graphics System Based on SVG[J]. Relay, 2005, 33(21): 69-73.
- [3] 王江静, 姜久雷. 可伸缩矢量图形(SVG)[J]. 现代电子技术, 2005(24):32-33.
WANG Jiang-jing, JIANG Jiu-lei. Scalable Vector Graphics(SVG)[J]. Modern Electronics Technique, 2005(24):32-33.
- [4] 王仲, 董欣, 陈晓鸥. SVG——一种支持可缩放矢量图形的 Web 语言浏览规范[J]. 中国图形图像学报, 2000, 5A(12): 1039-1043.
WANG Zhong, DONG Xin, CHEN Xiao-ou. SVG——A New Language Specification Supporting Scalable Vector Graphics for Web Browsing[J]. Journal of Image and Graphics, 2000, 5A(12): 1039-1043.
- [5] 贺睿, 苗雪苒, 陈剑云. 基于 SVG 的 SCADA 监控画面生成软件的设计与开发[J]. 工业监控计算机, 2006, 19(6): 32-33.
HE Rui, MIAO Xue-peng, CHEN Jian-yun. Design and Development of A Software of Drawing SCADA's Monitoring Graphics Based on SVG[J]. Industrial Control Computer, 2006, 19(6): 32-33.
- [6] <http://www.w3.org/TR/SVG11/>.
- [7] 陈芳, 徐学军. XML/Java 技术在 WebGIS 中的应用与实现[J]. 电力系统及其自动化学报, 2003, 15(1): 47-52.
CHEN Fang, XU Xue-jun. Application and Realization of WebGIS Based on XML/Java Technology[J]. Proceedings of the EPSA, 2003, 15(1): 47-52.

收稿日期: 2007-07-10; 修回日期: 2007-08-21

作者简介:

杨丰萍(1967-), 女, 副教授, 主要从事交通信息工程及控制, 计算机测控等方面的研究;

冯春华(1982-), 女, 硕士研究生, 主要研究方向为电力系统及其自动化; E-mail: fch820402@sina.com

王健(1978-), 男, 助教, 从事交通信息工程及控制研究。