

# 基于 SVG/AJAX/Internet 的电厂运行监视系统

胡 冰<sup>1</sup>, 章坚民<sup>1\*</sup>, 马国梁<sup>2</sup>, 方文道<sup>1</sup>, 郭明泽<sup>1</sup>

(1. 杭州电子科技大学 电子信息学院, 浙江 杭州 310018; 2. 浙江电力调度通信中心, 浙江 杭州 310008)

**摘要:**为了解决电厂运行监视系统的运行状态和监视指标的实时性和便利性问题,提出了一种利用 AJAX 技术实现系统的 Web 发布方案,以 SVG 作为图形系统的 Web 发布和显示格式,采用以 AJAX 技术为核心的异步通信机制,有效解决了信息实时动态刷新问题。实际应用表明,采用该方法实现的热电厂监控系统具有节省网络带宽、减少传输时延、页面更新无闪烁的优点,操作人员可以很方便的通过 Web 浏览器了解热电厂机组的运行状态,有利于协调政府、电网公司、电厂三者不同关切和利益。

**关键词:** AJAX; 监控系统; SVG; 异步通信; Web 发布

中图分类号: TP291; TH62

文献标志码: A

文章编号: 1001-4551(2011)08-0905-05

## Web publishing of plant operation monitoring system based on SVG/AJAX/Internet

HU Bing<sup>1</sup>, ZHANG Jian-min<sup>1</sup>, MA Guo-liang<sup>2</sup>, FANG Wen-dao<sup>1</sup>, GUO Ming-ze<sup>1</sup>

(1. College of Electronic Information, Hangzhou Dianzi University, Hangzhou 310018, China;

2. Zhejiang Electric Power Dispatching and Communication Center, Hangzhou 310008, China)

**Abstract:** In order to solve the problems of the real-time and convenience of operation status and monitoring index of plant operation monitoring system, a method was presented for the Web publishing of monitoring system of power plant based on AJAX. SVG was used as the Web publishing and display format of graphics system. In order to solve the problem of information's real-time dynamic refreshment, the AJAX was elected as the asynchronous communication mechanism. The application results show that this method has the advantages of saving network bandwidth, reducing transmission delay and flashed-free of page updates. Operation officers can easily learn the operation status of the power plant unit through Web browser. It's benefit for the coordination of government, grid state, power plant.

**Key words:** asynchronous JavaScript and XML(AJAX); monitoring system; scalable vector graphic (SVG); asynchronous communication; Web publishing

## 0 引 言

随着计算机技术在广域管理信息系统中的应用与发展,基于 B/S 的系统结构得到了广泛的应用。AJAX (即异步 JavaScript 与 XML) 技术的引入,大大改变了基于 B/S 结构的应用服务表示层灵敏不足的现状,通过向服务器发出异步请求,执行更新或查询数据库,在请求返回时,用 JavaScript 和 CSS 来相应地更新 UI,使得 Web 站点看起来是即时响应的,也给基于 Web 的

监控技术提供了一套解决方案。对在广域网或基于公共 Internet 下的监控系统,若采用基于“提交-等待-重新显示”的模式,尤其是监控页面仍采用非公开的位图图像技术生成,其数据更新必须通过手动刷新,不仅给用户操作带来极大的不便,也无法满足系统的实时性和互操作性。

国内的一些学者已经在图形的 Web 发布方面做了积极的探索和研究,并取得了不错的成果。文献[1-3]采用 applet 技术或 Activex 控件实现图形的 Web 发布,显示效果较好且可以实现丰富的交互功能,但由于

收稿日期: 2011-02-23

作者简介: 胡 冰(1986-),男,浙江衢州人,主要从事电子信息集成及在电力系统应用等方面的研究。E-mail: hub128@126.com

通信联系人: 章坚民,男,教授,硕士生导师。E-mail: zhangjm@tom.com

applet 是在客户端绘制图形,因此当下载到客户端的图形较复杂时,效率会较低,而且通常需要对客户端进行设置,降低了通用性;另外,Activex 控件在客户端的平台可移植性差,在升级和维护方面存在问题较多。文献[4]采用了 AJAX 技术实现电力系统图形的 Web 发布,有效解决了数据信息实时动态刷新问题,并为多用户协同设计中存在的关键问题(设计者角色问题和图形数据的管理问题)提供了解决思路。文献[5]提出了一种基于 SVG 和 Web Service 技术的图形 Web 发布方案,该方案具有图形显示质量较好、交互功能较强的优点,但其对服务器的配置要求较高,必须要有 UDDI 服务器的支持。

本研究将从地方热电厂管理信息系统实时监控机制的特点出发,提出以可缩放矢量图形(SVG)作为图形系统的基础,以 STRUTS 为系统的基本框架,引入以 AJAX 引擎为核心的异步通信机制,解决监控系统中的信息实时动态刷新问题,满足 Web 服务的实时性和互操作性要求<sup>[6-7]</sup>。

## 1 SVG 监控图

目前,地方热电厂的 SVG 监控图一般都是基于图数模一体化的思想,对不同设备建立图元,并对应于数据库,利用 Oracle 存储过程自动生成。SVG 图中每个图元对应于电厂的某个设备,将按照编码规则组成的设备编码设置为其 ID,用 `<tspan>` 元素存放设备指标实时数据,并将其 ID 设置为设备 ID 与指标名的组合,用于之后的 SVG 监控图异步刷新数据显示,包括设备的指标、运行曲线、汽轮机组累计量等数据。

```
<symbol viewBox = "0 0 40 40" id = "SteamTurbine_c" >
<g >
<rect width = "40" x = "0.0" height = "40" y = "0.0"
style = "fill-opacity:0;stroke-fill-opacity:0;" />
<polygon style = "fill: RGB (255,255,0); stroke: RGB (0,
0,0);"
points = "0.0,12.0 0.0,28.0 60.0,40.0 60.0,0.0 " />
</g >
</symbol >
<use id = "5022H00020H01S" x = "-60" xlink: href = "#
SteamTurbine_c" y = "0" onclick = "show_data(evt)" transform
= "translate(470 60)" />
```

实例代码中, symbol 中:定义了汽轮机运行状态的图元,其 id 为 "SteamTurbine\_c"; use 中: id 标识该设备编号, xlink: href 指出引用的图元, translate(470 60) 给出其坐标,并用 onclick 说明设备点击事件。

## 2 AJAX 技术介绍

AJAX 不是一种新的技术,它混合了 XHTML、CSS、DOM、XML 及 XSLT 等几项技术,并且利用 JavaScript 将上述技术整合到一起。使用它可以构建更为动态和响应更灵敏的 Web 应用程序,它使浏览器可以为用户提供更为自然的浏览体验,为数据交互频繁、数据分类良好的 Web 应用提供了一个很好的解决方案<sup>[8-9]</sup>。AJAX 的工作模式如图 1 所示。

XMLHttpRequest 是 AJAX 技术的核心,它允许一个客户端脚本来执行 HTTP 请求,并且将会解析一个 XML 格式的服务器响应。在 AJAX 处理过程中,首先会创建一个 XMLHttpRequest 实例,并设置 HTTP 方法 (GET 或 POST) 来处理请求。数据请求的 URL 被设置到 XMLHttpRequest 对象上,同时向 XMLHttpRequest 对象注册一个回调函数,XMLHttpRequest 请求被异步发出后控制权马上就被返回到浏览器。当请求的数据到达时,注册的回调函数将会被调用,以完成数据的解析和显示<sup>[10-11]</sup>。

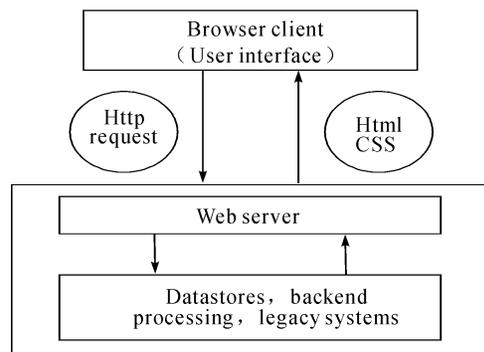


图 1 AJAX 的模式

AJAX 引擎的核心代码如下所示:

```
//xmlHttp 为 XMLHttpRequest 对象
xmlHttp.open("POST", url, true);
xmlHttp.onreadystatechange = handleStateChange;
xmlHttp.setRequestHeader("Content-Type", "text/xml");
xmlHttp.send(xml);
```

## 3 基于 AJAX 技术的监控系统 Web 发布的实现

热电厂 Web 发布系统的一个基本功能是向运行人员呈现大量实时遥测、遥信信息,以便于运行人员了解监测机组的运行状态。应用 AJAX 技术可以使监控系统摆脱手动刷新页面的困扰,将实时数据无闪烁地更新到与其关联的 SVG 图元中,从而达到对电厂实时

运行状态监控的目的。

### 3.1 监控系统的总体框架

基于 B/S 模式的热电厂监控系统采用了 3 层结构框架,包括数据库服务器、应用服务器和客户端。该系统提供了实时数据库查看、历史数据库查询、运行安全报警管理等功能。底层监测设备将各种传感器采集上来的数据经过预处理后存储到数据库中,应用服务器利用 AJAX 技术实现实时数据的动态显示。

### 3.2 浏览器客户端

本研究在客户端首先创建一个 XMLHttpRequest 对象,通过 open 函数来设置服务器端进行事物出来的 URL 及数据传递方式,接着通过 onreadystatechange 属性设置异步响应接收数据的函数,最后通过 send 函数来发送数据。

### 3.3 服务器端的响应

在该系统中服务器端的事务处理程序是用 JSP 应用程序,它的主要任务是负责与实时数据库通讯,获取实时数据,并将所得数据按照客户端的要求转换成 XSLT 格式的数据,然后返回到客户端。

### 3.4 实时页面的刷新

本研究在图形显示页面中利用 JavaScript 来实例化一个 XMLHttpRequest 对象,并每隔 2 s 向 XMLHttpRequest.jsp 页面发出请求,然后把返回的数据赋值给页面的 literal 元素进行显示。其基本步骤如下:

(1) 利用 DOM 收集 SVG 中的遥测量。

SVGViewer 插件提供了 getSVGDocument() 函数,调用该函数可以获得包含 SVG 文档所有元素的 DOM 对象,然后再调用 DOM 对象的 getElementById() 函数即可遍历所有图元,并从中找出所有的遥测量。

获取 SVG 中所有遥测量的 JavaScript 代码:

```
//取得 SVGViewer 对象
Var svgObj = document.getElementById("anbaibian");
//取得包含 SVG 文档所有元素的 DOM 对象
var svgDoc = svgObj.getSVGDocument();
//取得遥测量所在的层
Var measurementLayer = svgDoc.getElementById("measurementlayer");
//取得所有遥测量图元
var measurementNodes
= measurementLayer.getChildNodes();
//将所有遥测量 ID 添加到 xml 文档
var root = xmlTree.documentElement;
For(i < ( measurementNodes.length-1); i + = 2)
```

```
Var elem = xmlTree.createElement("cim:
MeasurementValue");
var attr = xmlTree.createAttribute("rdf:ID");
attr.value = measurementNodes.item(i).getAttribute("id");
elem.setAttributeNode(attr);
root.appendChild(elem);
}
```

(2) 利用 XMLHttpRequest 异步发送数据请求,并预设回调函数。

遥测量数据收集完成后,将交由 XMLHttpRequest 对象来完成数据的异步请求,及 Web 服务器返回数据的处理。

下面的代码以支持多种浏览器的方式创建了 XMLHttpRequest 对象:

```
var xmlhttp = false;
try {
xmlhttp = new ActiveXObject("Msxml2.XMLHTTP");
} catch (e) {
try {
xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
} catch (e2) {
xmlhttp = false;
}
}
if (! xmlhttp && typeof (XMLHttpRequest) != 'undefined')
xmlhttp = new XMLHttpRequest();
}
下面的代码实现了数据的异步请求和返回处理:
//定义 url 地址
var url = "....."
xmlhttp.open("POST",url,true);
xmlhttp.onreadystatechange = function(e) {
if (xmlhttp.readyState == 4 && xmlhttp.status == 200) {
processXML(xmlhttp);
}
}
```

遥测量数据被 XMLHttpRequest 对象以 POST 方式异步提交给服务器端进行处理,并预设 processXML() 函数处理返回的实时数据。浏览器获得返回数据后,XMLHttpRequest 对象调用预设的 processXML() 函数对数据进行分析处理,并通过 DOM 提供的方法与实时数据关联的 SVG 元素,根据数据的类型,更新 SVG 元素的状态。

从上面的处理过程可以看出,从实时数据信息的收集、数据请求提交、返回数据处理直到 SVG 的状态更新的过程中,XMLHttpRequest 是整个处理过程的核心。

心,而 JavaScript 则绑定了一切交互操作。

### 3.5 历史曲线的显示

在历史曲线中无需实时动态显示,只需要用一个记录集 recordset 将数据库中的历史数据表读取过来,然后绑定到 VML 中的 line 函数即可。

## 4 实例应用

该监控系统是为某热电厂的运行管理而进行设计和开发的,以 STRUTS 为基本框架,应用 AJAX 技术实现异步刷新 SVG 图形,从而达到对电厂实时运行状态监控的目的。

在该系统中,客户端的电厂监控 SVG 图运用 js 的定时技术将每隔一段时间向服务器发送一次请求,其 url 为: var url = "/realTime.do? PCommand = RT\_Update&ts = " + new Date().getTime()。其中 realTime.do 为 STRUTS 的 Acion,PCommand 指定了该请求将调用类 RealTimeAction 的处理函数 RT\_Update。在 url 中添加递交时间是为了避免客户端认为命令重复而没有向服务器请求。SVG 监控图的热电厂 ID 用 xml 文件通过 send(xml) 函数发送给服务器。

STRUTS 的控制器 Servlet 接收到客户端发送上来的请求后,通过 Dispatch 转到 RealTimeAction 的 RT\_Update 处理函数。RT\_Update 函数通过 request 实例读取参数 XML 文档并解析获得热电厂 ID;然后访问数据库取得全厂、锅炉、汽轮机组和减温减压器的更新数据,并将其打包成 XML;接着通过 response.getWriter() 获得 PrintWriter 对象,用其 print 方法将之前生成 XML 输出;最后 return mapping.findForward(null)。

其中返回的 XML 文档结构形式为多个 < map > 节

点,其中每个 < map > 对象包含一个 < name > 和一个 < value > 子节点。< name > 对象里存储的是设备指标的 ID,对应 SVG 监控图中文本 < tspan > 元素的 ID 属性值,两者是一一对应的;< value > 对象里存储的是采集数据实际值加一位数据越界标识位,两者之间用“-”隔开。其中越位标识为 1 时,说明该设备指标的采集值已经超过了其允许的最大、最小值范围,将在 SVG 监控图中对其字体颜色设置为红色,以达到告警目的。

在客户端这一侧,当 xmlhttp.readyState 等于 4 时,即请求完成,便调用更新 SVG 图数据的函数 updateList(),解析返回的 XML 数据,通过 JavaScript 操作 SVG Document 接口,先根据指标 ID 获得替换元素 var teTarget = svgdoc.getElementById(id),再创建新的文本元素 var newText = svgdoc.createTextNode(data[0]),然后替换 SVG 监控图中的原文本内容 teTarget.replaceChild(newText, teTarget.getFirstChild()),最后判断该指标的越位标识并作相应处理,实现数据的实时刷新。

按照上述架构设计的系统具有通用性,目前开发的监控系统已经投入使用,这种监控系统的基本功能如下:

(1)热电厂 SVG 运行图的实时更新。使用 JavaScript 的 setTimeout 函数每 2 s 向服务器请求电厂各设备的最新采集数据和全厂、机组累加量数据,并通过 SVG Document 接口更新 SVG 图形的文本内容,实现数据的实时性;同时能用红色标注越界的采集值、用流向标识蒸汽管道和电流的流通状态,监控系统的画面如图 2 所示。

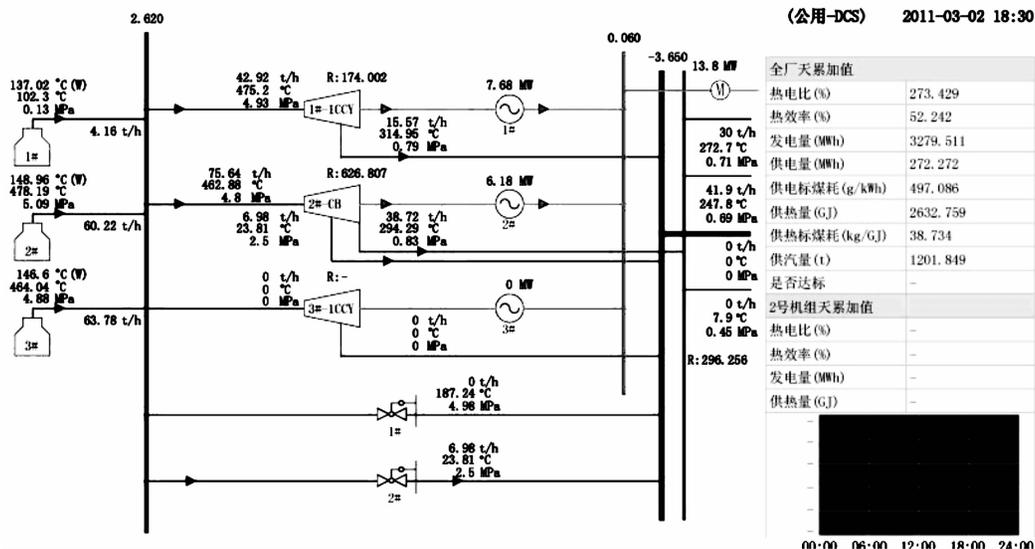


图 2 电厂监控系统画面

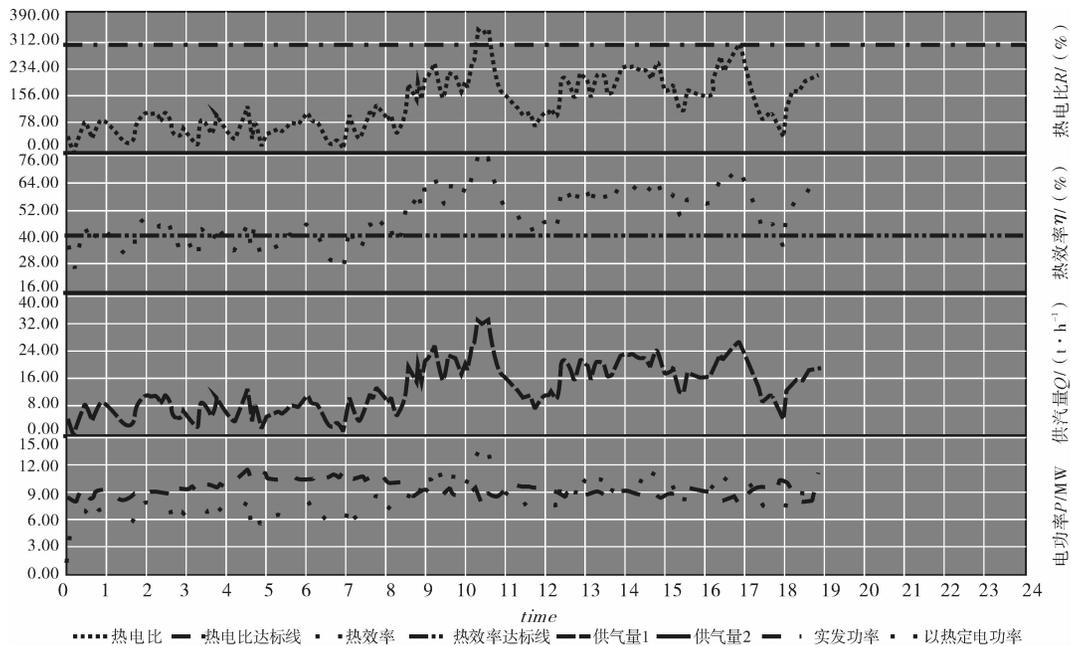


图3 热电厂热电比、供气量、以热定电功率等指标的当日曲线

(2) 点击显示电厂设备指标的运行曲线。当鼠标点击 SVG 图上的设备指标时, 显示该指标的当天运行曲线图, 如图 3 所示。其中横坐标为时间, 纵坐标为指标量。图中显示了电厂热电比(%)、热电比达标线(%)、热效率(%)、供气量( $t \cdot h^{-1}$ )、实发功率(MW)等指标一天 24 小时的运行曲线图。运行人员可以很方便地掌握各个设备指标的数据, 从而快速判断出设备有无出现异常。

(3) 点击显示汽轮机组累加量数据。用鼠标点击汽轮机或发电机图形时, 可以将机组的当天累加值在固定的文本区显示。运行人员可以清晰、快捷地获得各个汽轮机组的数据信息从而做出正确的决策。

## 5 结束语

基于 Internet 和 Web 形式的电厂运行监视系统是电厂安全运行和发展的必然趋势。监控系统由于其自身的特点, 往往需要让现场运行人员、专业工程师、管理人员在不同地点同时查看, 以提高运行与管理的协同能力。该系统正是基于这种考虑建立起来的。该套监控系统已经在某地方热电厂管理信息系统中得到了运用, 取得了很好的效果, 达到了预期的目标, 该系统具有以下优点: ①基于 B/S 模式的设计, 可以方便地用 IE 进行浏览而不需要安装额外的软件, 操作简单; ②该系统可以很容易地与其他系统进行交互; ③开发费用低, 由于该系统采用了统一的设计模式, 对各个监测量其设计思想是一样的, 因此, 在开发完一个子系统后, 代码移植起来就很方便, 不需要做太多的修改工作。

## 参考文献 (References):

- [1] 张洁. 基于 Web 技术的 EMS/DMS 图形和数据访问[J]. 电力系统自动化, 1999, 23(15): 45-46.
- [2] 江宏, 孙忠利, 张国浩, 等. 变电站监控系统的 Web 服务器实现方案[J]. 电力系统自动化, 2003, 27(3): 80-81.
- [3] 马杰, 赵传刚, 李军, 等. 电网故障信息的动态网页发布系统[J]. 电力系统自动化, 2002, 26(4): 76-77.
- [4] 敖丽敏, 李林辉. 基于 AJAX 的电力图形系统的实现. 电力系统及其自动化, 2007, 31(19): 47-50.
- [5] 冒宇清, 唐晓莉, 姜彬. 基于 SVG/Web Service 的 Web 监控技术在 EMS 中的应用[J]. 电力系统自动化, 2007, 31(4): 94-95.
- [6] 余名高, 吴海林, 熊童满, 等. AJAX 在 Struts 中的应用[J]. 计算机技术与发展, 2007, 17(10): 69-72.
- [7] SERRANO N, AROZTEGI J P. Ajax frameworks in interactive Web Apps[J]. *Software, IEEE*, 2007, 24(5): 12-14.
- [8] 刘海林. 基于 SVG 的实时数据 WEB 发布系统的设计与实现[D]. 天津: 天津大学计算机科学与技术学院, 2007: 35-42.
- [9] 周旋, 王丽芳, 蒋泽军. 基于 Ajax 的即时消息系统的设计与实现[J]. 科学技术与工程, 2009, 19(2): 447-448.
- [10] 赵鑫, 向军, 天寿, 等. 基于 AJAX 技术的小水电站运行效率监测系统设计与实现[J]. 电力系统保护与控制, 2009, 37(2): 83-85.
- [11] MOHAMMED S, FIAIDHI J, GHENNIWA H, et al. Developing a secure Web service architecture for SVG image delivery[J]. *Journal of Computer Science*, 2006, 2(2): 171-179.

[编辑: 罗向阳]