

# 基于 WEB 和 SVG 的 EAST 真空 监控和数据处理系统

陈 跃, 王 玲, 杨道文, 胡建生, 王小明

(中科院等离子体物理研究所, 合肥 0551)

**摘要:** 目前自动化领域的 WEB 组态通常在客户端嵌入 J2EE / Applet 实现 C/S 的回归, 这种方法在很多方面都有极大优势, 如安全性、移植性、动态性等, 然而编程相对复杂。随着 SVG (Scalable Vector Graphics) 和 AJAX (Asynchronous JavaScript And XML) 被广泛地支持, 允许我们在绝大多数的浏览器上实现矢量绘图和简单的 C/S 回归, 实现类似于 Applet 的功能, 如信息的异步传输、数据的图表显示和查询等。介绍一种通过 SVG、AJAX、OPCXML-DA、COM 等多种技术实现 Web 组态的功能。

**关键词:** ASP.NET; AJAX; SVG; OPC; COM

**中图分类号:** TP 277 **文献标识码:** A **文章编号:** 0258-0934(2010)03-0600-06

EAST (Experimental Advanced Superconducting Tokamak) 装置是我国自主建造的第一个非圆截面的超导托卡马克, 在世界聚变研究中有着相当重要的意义。EAST 真空系统是其基础系统之一, 是装置运行的前提, 保障真空系统的稳定、可靠运行有着非常重要的意义。原有真空控制系统分为二层结构, 第一层为现场的手动控制, 第二层为基于 Kingview 软件的控制网内 C/S 控制模式。这种结构不利于实现 Internet 上的远程操控和数据的共享及发布, 有必要在原有控制系统上进行网络扩展以实现远程操作及数据查询。实现此功能难点在于多种数据源的数据交互、监控信息的异步传输、大量数据的快速处理及发布。本系统采用 OPCXML-DA 网关及 WebService 实现数据的网络传输, 利用 Oracle 数据库完成数据的压缩存储和中转, 使用 SVG、AJAX 发布监控图

形和监控信息的异步传输, 通过 MATLAB、COM 手段进行数据的图形化处理。通过以上多种技术的综合应用可以方便地实现控制系统的网络扩展。

## 1 EAST 真空控制系统构成及网络拓扑

EAST 真空系统是一个含 PROFIBUS、RS232、RS485、MODBUS 等多总线协议的混合通讯结构, 其控制对象有 15 台旋片泵、6 台罗茨泵、18 台分子泵、11 台低温冷凝泵、若干台活动机组、100 多个阀门、100 多道模拟信号等千余点数的变量。控制单元主要由 2 台 S7-400PLC 构成, PUNPING 机主要负责控制真空设备的运行和真空度的采集, GIS 机主要负责等离子体加料, 装置温度采集和部分真空度的采集。2 台 PLC 通过 PROFIBUS-DP 总线与控制室的主控计算机连接, 主控计算机采用 Kingview 软件进行组态, 完成对所有真空设备的实时监控。Vacuum DataBase 设计为一台真空数据服务器, 存储数据, 同时提供 IIS 服务 (WebServer), 实现真空控制的网络扩展即远程控制和数据查询。EAST 真空监控系统网络

收稿日期: 2009 08 25

基金项目: 国家自然科学基金(10705030)部分支持。

作者简介: 陈跃(1983-), 研究实习员, 从事自动化控制。

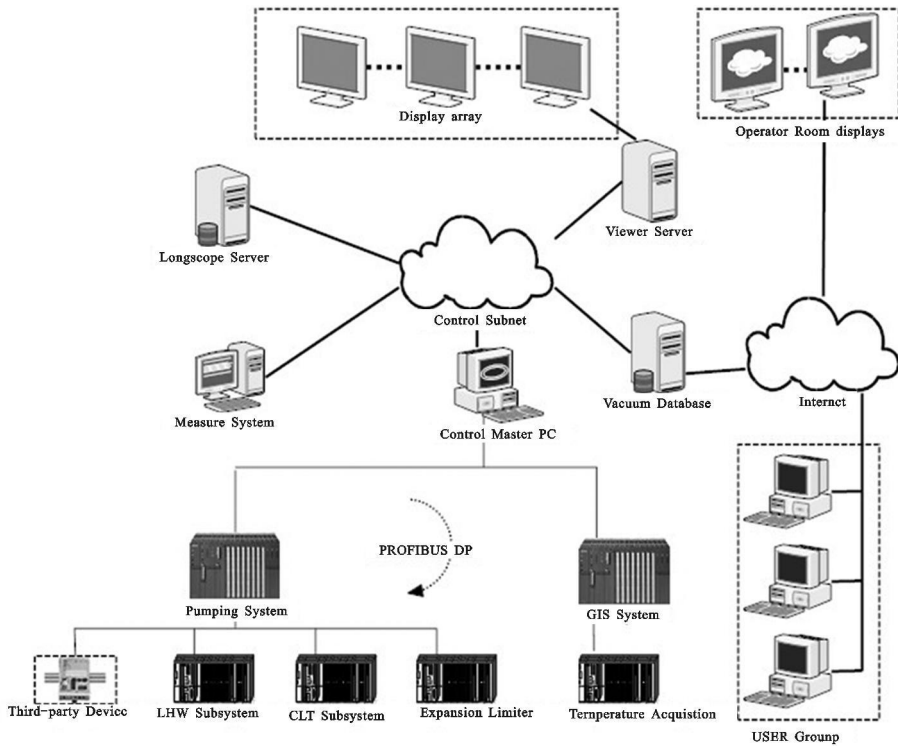


图 1 EAST 真空监控系统网络拓扑

## 2 数据库构建

数据库为该系统的核心部分, 不仅承担数据的存储, 还作为实时数据显示和在线监控的中转站, 因此数据库设计的成功与否将直接影响数据的访问速度。该系统采用 Oracle10g 数据库(见图 4)。主控计算机上的 OPC-DA 访问 Vacuum DataBase 中的数据库, 每秒更新一次实时数据表(RT SignData Table)和实时设备状态表(RT DeviceState Table), 每次更新后触发器将信号转存到历史子表中(History SignData Subtable 和 History DeviceData Table)。对于模拟量的信号数据采用通用的工业数据存储方式, 即按数据变化率存储(History SignData Subtable)。由于真空度和装置温度

均是缓慢变化的量, 其数据压缩率可以达到 90% 以上, 大大减轻了数据吞吐量, 提高了读取速度。而对于布尔量的设备状态采用任意设备状态变更则存储全局状态的方案, 并对设备的多种状态采用编码的方式进行存储, 即 0-255 的状态码来标志设备某时刻的运行状态, 如表 1。这样做的优点在于一方面减少了数据的存储量, 另一方面可以减少网络传输的带宽, 状态码可以由客户端的 JavaScript 脚本进行解析, 减轻 WebServer 的负担。另外还在数据库中建立了一个过程函数用来每 1 h 提取所有信号数据, 形成报表并进行单独存储(Report Table), 以避免浏览报表时引起多表查询, 降低 WebServer 性能。

表 1 低温泵状态编码表

启/停	再生	水温高	缺相	线圈故障	气温高	水流故障	压机故障
-----	----	-----	----	------	-----	------	------

## 3 WEB 在线监控实现

SVG(Scalable Vector Graphics)可伸缩矢量图形是一种基于 XML 文档的文本图形标记语言, 可以在 WEB 上实现矢量绘图、动画, 并支持 JavaScript 脚本, 符合 DOM 2 规范, 可以

在客户端动态地改变图形的样式, 被广泛应用在 WebGIS 上<sup>[1]</sup>。近几年在自动化控制领域也得到了初步的应用。

### 3.1 创建图元文件

对于一个工业控制的图标, 往往包含很多信息, 如设备的启停操作(事件), 设备状态(颜

色)等等,所以必须为每种设备定义可复用 SVG 图元,然后通过< use> 标签引用绘制出控制界面即可<sup>[2]</sup>。但是由于 SVG 对 DOM 影像树操作较弱,所以必须对设备图标每个变量对应的图形区域进行图元化,如图 2 分子泵图标,便是由表征操作的图元- pumper 和表征设备状态的图元- tmp 组合而成。



2 分子泵 SVG 图标

```
< symbol id= "pumper">
  < rect x= "0" y= "0" width= "54" height= "54"
 />
< /symbol>
< symbol id= "tmp">
  < circle cx= "27" cy= "27" r= "25" />
  < line y1= "7" y2= "50" x1= "27" x2= "27" />
  < line y1= "27" y2= "27" x1= "15" x2= "39" />
  < line y1= "34" y2= "34" x1= "15" x2= "39" />
  < line y1= "41" y2= "41" x1= "15" x2= "39" />
  < line y1= "20" y2= "20" x1= "15" x2= "39" />
  < line y1= "13" y2= "13" x1= "15" x2= "39" />
< /symbol>
```

要对具体的设备进行操作还必须为每个设备添加事件响应函数,表征设备状态也需要定义脚本函数来连接状态码和图形样式,详见客户端代码。

### 3.2 AJAX: 监控指令的异步传输

AJAX ( Asynchronous JavaScript And XML) 异步, JavaScript 及 XML, 是由微软创立, Google 推广的一种 B/S 异步通讯手段, 可以实现网页的局部刷新<sup>[3]</sup>。其核心组件 XMLHttpRequest 对象, 目前已经被绝大多数的浏

览器所支持,也是 Web2.0 的关键技术。

对于每个自动化控制的设备图标需要实现两种功能,即对设备的操作和对设备状态的监视。它们均通过 AJAX 和 WebServer 进行异步通讯,不同的是访问的内容和途径。当对设备进行操作的时候,客户端传递设备 ID 和指令码给 WebServer,这时 WebServer 访问位于主控计算机的 OPCServer。对设备的监视只需访问本地 Oracle 数据库中的实时设备状态表即可。如何访问局域网中 OPCServer,通常的做法在 .NET 环境中调用 OPC 服务商提供的客户端 dll 文件,编写网络 OPCClient,但是这种方法受限于 WINDOWS 操作系统的安全机制。这里使用 OPCXML-DA 规约在 主控计算机上架设 WebService 以供 WebServer 调用<sup>[4]</sup>, 流程如图 3 所示。

客户端点击一个设备图形发送启动或停止命令, WebServer 端得到命令后通过调用架设在主控计算机上的 WebService 提供的接口函数将命令提交给 OPCXML-DA 网关,通过 OPC-DA 的 DCOM/COM 接口改变 OPC-Server 对应地址的值<sup>[5]</sup>,进而启动或停止设备。因为现场设备响应时间较长,所以 WebServer 不会向客户端返回操作结果,其结果由 Oracle 数据库中的实时设备状态表来间接反映。可以设置在客户端周期执行一个请求函数向 WebServer 请求设备状态, WebServer 通过 ADO.NET 访问 Oracle 数据库将结果处理后发送给客户端,如果没有设备变化则返回空,如果有设备变化则返回设备的 ID 和状态,客户端解析后改变相应设备图形的颜色,从而实现对设备的监视。对于查询历史设备状态的方法与查询实时状态相同,只是访问的表不同而已。

#### // 客户端主要代码

```
var xmlhttp = new ActiveXObject ( " Microsoft.
XMLHTTP" ); // 以 IE 为例

function controlpageinit( evt )
{
  addevent( evt ); // 为设备添加操作事件;
  loadstate( evt ); // 初始化设备状态
}

function sendcommand( evt )
{
  var deviceID = evt.target.getAttribute( "id" );
  ajaxSend( deviceID );
}

function ajaxSend( id )
{
  xmlhttp.open( "get", "operator.aspx? id= " +
  id, true );
```

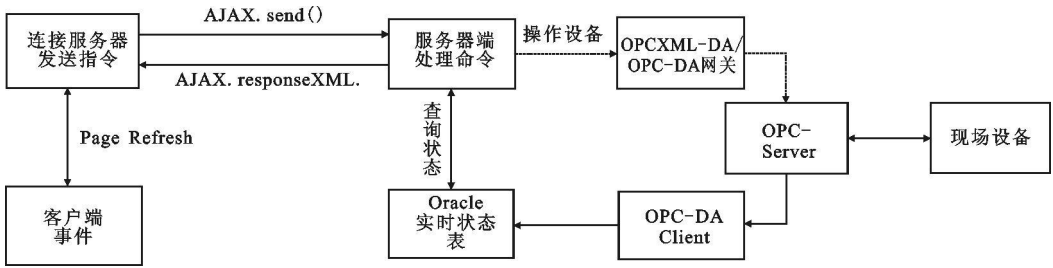


图 3 AJAX 访问流程

```
xmlhttp.onreadystatechange = operateresult;
xmlhttp.send(null)
function operateresult(evt)
{
    var reback; // 存储 WebServer 返回的信息
    var deviceID; // 存储状态更改的设备 ID;
    var deviceState; // 存储相应的状态码;
    svgdoc = evt
    if (xmlhttp.readyState == 4 && xmlhttp.status == 200)
        reback = xmlhttp.responseXML;
    if (reback == "Operate Success!") // 如果返回此字符串, 表示操作指令完成;
        alert("指令已经成功传送!");
    else // 否则为查询设备状态。
    {
        var xmldoc = reback.documentElement;
        deviceID = xmldoc.getElementsByTagName("device");
        deviceState = xmldoc.getElementsByTagName("state");
        decode(deviceID, deviceState); // 解码
    }
}
function addevent(evt)
{
    var svgdoc = evt.target.ownerDocument;
    for (var i = 0; i < operatelist.length; i++)
    {
        svgdoc.getElementById(operatelist[i]).addEventListener("click", sendcommand, false);
    }
}
function loadstate(evt)
{
    {
        ajaxSend(-1); // 请求初始化界面
        setInterval(ajaxSend(0), 10000); // 每 10s 更新一次界面
    }
}
```

服务器端主要代码:

首先在 ASP.NET 中添加对 HTTP://192.168.0.10/simaticOPCservice.asmx 主控机上的西门子 OPCService 的引用。

```
using System;
using System.Collections;
using System.Web;
using System.Web.Services;
using System.Web.HttpRequest;
```

```
using System.Data;
using System.Data.SqlClient;
using System.Data.OracleClient;
private void Page_Load(object sender, System.EventArgs e)
{
    string id = Request.Form["id"].ToString();
    DataSet ds;
    XmlDocument xmldoc;
    if (id == -1 || id == 0) // 请求初始化或更新
    {
        ds = myfunction.query();
        xmldoc = myfunction.toXML(ds);
        // 将实时状态表的数据转化为 xml 文档
        xml = myfunction.compare(xmldoc);
        // 与 SESSION 中存储的状态表比较, 返回状态改变设备的 xml
        // 文档需要在会话开始的时候向 session 定义一个 xml 空文档
        Response.Write(xmldoc); // 返回客户端
    }
    else
    {
        simaticOPCService operater;
        operate.write(myfunction.toaddress(id));
        // toaddress() 返回设备对应的底层地址。write 将状态置反
        Response.Write("操作成功");
    }
}
```

## 4 数据的在线查询

数据的查询可分为历史数据查询和实时数据显示, 对于前者由于数据量巨大往往会占用用户极大的带宽并会导致 WebServer 响应缓慢, 因此此类数据应该由 WebServer 端的第三方软件处理, 并需要与 IIS 相对隔离, 防止 WEB 服务响应缓慢, 而实时数据的查询由于数据量极小而且实效性要求较高, 适宜在客户端执行。

### 4.1 MATLAB: 历史数据的查询

MATLAB 作为一种强大的数据图形处理软件, 提供很多方法供第三方语言调用, 如 ActiveX、DDE 等等, 这里我们使用 COM (Component Object Model) 组件的方式来调用。这有

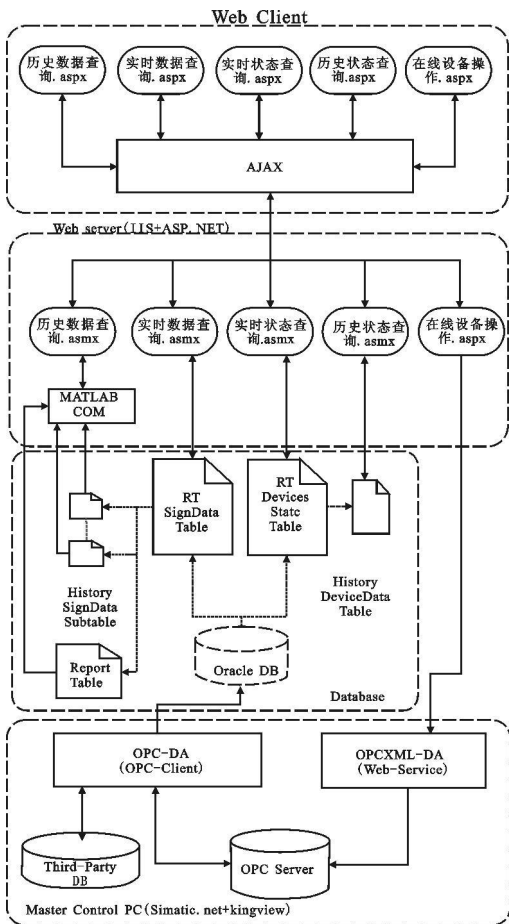


图 4 WEB 服务器构架

几个方面的优势: (1) COM 编写相对独立, 只要设计好数据处理函数的入口和参数就可以直接在 .NET 上调用, 易于维护。(2) 在调用 COM 组件的计算机上不需要安装 MATLAB 软件, 移植性较好。(3) 调用方便, .NET 对 COM 组件有极佳的支持, 不需要注册环境变量。

首先在 MATLAB 中编写  $CurvePath = HistoryDataCurve (begindate, enddate, signame)$  函数, 其中  $begindate, enddate$  为所要查询数据的起始时间,  $signame$  为所要查询的信号名数组,  $HistoryDataCurve$  会根据信号名和起始时间连接在 Oracle 数据库中相应的数据表, 查询数据并将数据图形化存储在指定的目录中, 并返回图形文件的路径。然后使用  $mbuild, -setup$  将 VS.NET 2005 设置为当前的编译器, 注意如果服务器为 64 位, 必须安装 VS.NET 2005 的 64 位编译器, 之后使用 MATLAB Deployment Tool 的将  $HistoryDataCurve$  生成的  $m$  文件编译成  $dll$  文件即可。

在 ASP.NET 引用中添加刚才生成  $dll$  文件, 就可以在程序集中找到  $HistoryDataCurve$  的函数集了, 客户端查询历史数据时  $WebServer$  调用这个组件将生成的图片返回给客户端, 完成一次查询, 同时  $WebServer$  端要将这个图形路径存储在  $Session$  中, 以便会话结束后删除所有图片。

## 4.2 SVG: 实时数据显示

实时数据显示是为了提供给 EAST 放电控制中心, 以决定当前的真空状态是否满足放电条件, 因此使用 SVG 实时绘制曲线可以更直观地展示给操作人员, 当然也可以表格的形式在 Web 上发布。使用 SVG 绘制实时数据曲线是非常方便的, 首先在客户端编写一个周期执行的函数和一个  $< path >$  标签 (用于显示曲线), 每隔一段时间使用 AJAX 向  $WebServer$  请求数据,  $WebServer$  访问数据库中的实时信号表返回当前值, 客户端将这个数据 Push 到对应的数组, 之后将这个数组转化为符合 SVG  $< path >$  标签  $d$  属性的格式, 并对其赋值, 这样就可以得到信号的实时曲线。另外还需要限制曲线的点数以减少内存的消耗和选取合适  $path transform$  属性以进行坐标转换。

## 5 结束语

AJAX 技术作为一种 B/S 系统下的异步通信技术已经被绝大部分的浏览器支持, 并已被应用在很多网站上, 用在工业组态上并不多见。SVG 是项方兴未艾的技术, 更多的浏览器将对其提供原生的支持。该技术较多应用在 WebGIS 上, 在工业组态软件中尚没有成熟的产品。OPCXML-DA 网关则是为将 OPC-DA 数据格式转化为适合 Internet 访问的 XML 格式, 同时也可以解决 Window 安全机制的限制。本系统尝试将多种技术应用在 Web 组态上, 系统的安全性将由 asp.net 安全机制来负责, 同时在 OPCServer 中为控制功能添加了软件锁, 只有得到本地的允许才能操作设备。目前该系统已经在 EAST 实验中得到预期的控制效果, 并将不断完善以提供更多的功能扩展。

参考文献:

[1] 韩双旺. 基于 ASP.NET 与 SVG 的 WebGIS 实现

- 技术研究[J]. 测绘科学, 2009, 34(2): 157-160.
- [2] 侯晓静, 苑律莎, 李中, 等. 基于 SVG 的网络图形制作方法研究[J]. 电力系统通信, 2007, 28(6): 43-46.
- [3] 吕小平, 周继, 董丽丽. 基于 Ajax 技术的 Web2.0 开发应用[J]. 现代电子技术, 2009, 32(6): 91-93.
- [4] 辜裕新. 基于 OPCXML 的建筑智能化系统集成及  
技术实现[J]. 现代商贸工业, 2007, 19(9): 259-260.
- [5] 汪洪涛, 高璟, 夏士雄. 基于 Web 服务的 OPCXML-  
DA 系统的研究[J]. 计算机应用研究, 2004, 21  
(11): 101-103.

## The Design of EAST Vacuum Control System Based on Web and SVG

CHEN Yue, WANG Ling, YANG Dao-wen, HU Jian-sheng, WANG Xiao-ming

(Institute of plasma physics, Chinese Academy of Sciences, Hefei, 0551)

**Abstract:** the J2EE / Applet is usually embedded in the client to achieve the C/S regression of the web configuration in automatic field. This method has great advantages on many respects such as security, portability, dynamic and so on, but programming is not easy. With SVG (Scalable Vector Graphics) and AJAX (Asynchronous JavaScript and XML) being widely supported, the vector plotting and simple C/S regression can be allowed in the vast majority of browsers. So the functions similar with Applet can be accomplished, e. g. information asynchronous transmission, data graph and search. It introduces a method to achieve the web configuration by applying the techniques of SVG, AJAX, OPCXML-DA, and COM.

**Key words:** ASP.NET, AJAX, SVG, OPC, COM

(上接第 596 页, Continued from page 596)

## The Design of High Precise Event Time Stamping module in BESIII TRIGGER System

ZHOU Zhong liang<sup>1,2,3</sup>, LIU Zhen an<sup>1,2</sup>, LI Lu<sup>1,2,3</sup>, JIN Da peng<sup>1,2</sup>,  
ZHAO Di xin<sup>1,2</sup>, WANG Qiang<sup>1,2,3</sup>

(1. Institute of High Energy Physics, Beijing, 100049,

2. Key laboratory of Nuclear Electronics and Detection of Chinese Academy of Sciences, Beijing, 100049,

3. Graduate School of Chinese Academy of Science, Beijing, 100049)

**Abstract:** It introduces about the design and implementation of High Precise Event Time Stamping module, which is responsible for stamping the good event pass signal L1\*, generated by trigger system. The module is based on GPS timing solution, supported the VME typical read/write cycles and protocol like CBLT for data transfer; all functions are implemented in FPGA. Time Stamping data output is given at the end of the paper. The design achieves the preliminary design object.

**Key words:** trigger, GPS timing, time stamping, FPGA, CBLT