

能量管理系统服务子系统 WebS 的设计

薛 蕾

(江苏省电力公司电力科学研究院,江苏 南京 211103)

摘 要:讨论了能量管理系统中服务子系统 WebS 的设计方法。服务子系统 WebS 是能量管理系统中的一个重要子系统,具有对外信息发布、信息查询与服务管理的功能。在 Web 服务系统构架上,对 C/S 模式和 B/S 模式进行了分析研究和方案比较,探讨在 B/S 模式下的异步模型框架下,寻找适当的技术进行能量管理系统的服务子系统 WebS 设计。最终得出了 SVG+Ajax+HTTP 服务器的构架结构,并通过浏览器端使用 HTML 网页内嵌 SVG 图形和 JavaScript 脚本的方法实现页面信息刷新与响应。

关键词:服务子系统 WebS;能量管理系统;B/S 设计模式;异步访问模型

中图分类号:TP393.09;TM73

文献标识码:B

文章编号:1009-0665(2011)04-0048-04

在能量管理系统中,服务子系统 WebS 是 1 个极其重要的子系统。电力系统软件规划者人为地将能量管理系统划分并要求部署在 3 个安全分区中^[1]。其中 I 区作为实时运行信息区,安全要求级别最高。III 区作为安全级别更低的安全区域,部署服务子系统 WebS,进行信息的发布以及与其他非实时应用的信息交互。从 1990 年代开始,各个能量管理系统都已具备各自的 Web 服务功能,并且不断发展。期间对如何设计与开发好服务子系统 WebS 发生过多次的技术争论,实现方法在各具体产品中也有相当大的区别,其展现效果也有巨大的差异。这些方法大都由于当时的计算机技术,包括通信技术、浏览器技术等限制,都或多或少地存在一些缺陷。近年来,先进的计算机技术不断涌现,技术环节的衔接手段与方法也在不断改进,为改进能量管理系统服务子系统 WebS 的设计提供了可能。

1 WebS 的架构模式

1.1 基于 C/S 模式下的 Web 架构

基于 C/S 模式的 Web 客户端采用各种高级语言进行编写,编译后通过文件发布服务将程序发布并更新到客户端。用户直接使用发布到客户端的执行程序进行浏览与操作。Web 的信息发布功能可根据要求灵活配置,比如图形浏览、曲线查询、告警浏览、报表查询等功能均可通过 Web 的更新发布机制提供给客户端。如图 1 所示,Web 服务器上需要部署基本的网页发布服务、实时数据服务、历史数据服务和网络通信服务等基本服务程序。Web 客户端通过网页发布机制将相关程序和资源缓存到本地,再通过主网页提供相关模块的链接进行程序的调用。由于客户端直接采用现有的图形程序进行部

署,所以原有的实时数据服务、历史数据服务、告警查询服务等都可以直接使用。

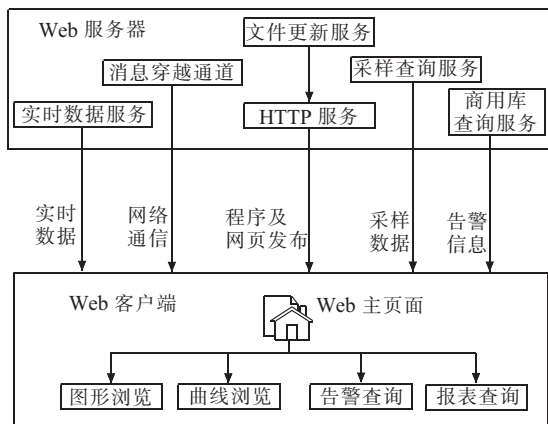


图 1 基于 C/S 模式的 Web 架构

1.2 基于 B/S 模式下的 Web 架构

基于 B/S 模式的 Web 架构与基于 C/S 模式的 Web 架构不同,其主要利用浏览器对脚本语言的支持能力,实现浏览器与服务器之间的 HTTP 请求与响应,将大量的应用逻辑处理放在服务器端进行处理。相对于 C/S 模式而言是一个非常轻量级的客户端展示方式。

B/S 模式下最简单的 Web 应用程序模型为同步模型,即输入数据,发送页面到服务器,等待响应,如图 2 所示。这里存在着 2 个基本的问题:一是必须等待服务器的响应;二是服务器不能发起更新。

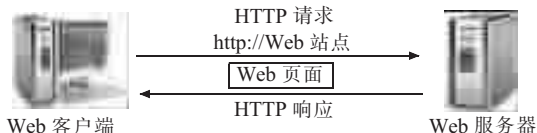


图 2 基于 B/S 模式的 Web 同步模型

为此,异步模型则更加吸引人。在异步模型中,应用程序不会让所有的客户端操作响应都回到服务端去处理,而是部分变化由客户端处理,部分响应由

服务端处理的技术方法。这个模型中所交互的网页即不是所有信息的全集，也不是仅有显示的部分页面，在使用过程中，只有包含新信息的用户界面元素才会被更新，其余部分都保持不变。这就意味着不需要发送全部的信息，不需要长时间等待响应，如图 3 所示。

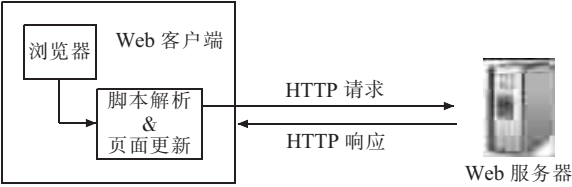


图 3 基于 B/S 模式的 Web 异步模型

1.3 Web 架构模式选择

基于 C/S 模式的服务子系统 WebS 可以实现作为一个服务子系统 WebS 所要实现的各项基本功能要求，包括图形浏览、告警查询、报表浏览等功能。而且由于采用实时系统的应用程序直接作为 Web 客户端程序发布，其功能非常的强大且交互性强。但是基于 C/S 结构的方式有明显的缺点。

(1) 客户端需要下载。对于每一个客户端，都需要下载客户端程序和相关的配置与图形文件到本地。这一过程会导致一些问题：① 下载时间长；② 对磁盘空间有要求，在某些系统上需要 500 MB 甚至 1 GB 以上的空间；③ 磁盘上的文件一旦下载，清除困难，导致存在垃圾文件；④ 可能存在安装失败的情况，异常处理复杂。

(2) 程序更新复杂。任何的修改都需要向客户端进行发布，部分发布就有可能带来匹配问题

(3) 客户端软件冲突。当同一客户端浏览多个 Web 服务器可能会发生客户端软件冲突。

(4) 对浏览器安全策略有要求。由于需要允许下装控件，要求浏览器降低安全权限和可信站点进行配置，不符合安全防护的功能要求，同时也要求使用 Web 客户端的使用者需要手动的进行操作，增加了技术支持人员的工作量。

(5) 网络设备需要维护。网络上要开放数个通信端口以供应用程序通信。而由于系统中网络结构往往比较复杂，跨越多个交换机和路由器的情况时有发生，一旦出现访问异常，很难确定是问题原因。且随着安全防护要求的不断增加，开放的端口越少越好。

(6) 对网络带宽要求比较高。由于使用客户端程序直连，为加快操作与交互的速度，大量初始信息会提取到客户端进行逻辑分析，而非最终的结果发布，导致对网络带宽的要求增加。

(7) 对服务器性能要求比较高。由于每一个客

户端的登陆、浏览都要消耗服务器一些资源，导致服务器的负载较高。因此 C/S 客户端方式支持的在线人数不多。

正是由于基于 C/S 模式的服务子系统 WebS 存在着种种的缺陷，新的服务子系统模式的研究就显得十分重要。需要解决的问题：(1) 下载内容尽量少，甚至零下载，即不落地的 Web；(2) 开放端口尽量少，最好只开放标准的 8000 端口；(3) 服务器资源占用量尽量少，允许更多的访问用户；(4) 客户端对服务器的访问相对独立，即一个客户端可浏览多个 Web 服务器上的内容；(5) 客户端是免维护的。

而基于 B/S 模式的服务子系统 WebS 正好在这方面具有得天独厚的优势。其客户端包含 HTML 页面和嵌入的脚本，这些内容是不落地的；正常浏览仅需开发一个标准的 HTML 端口(8000)或一个指定发布端口；服务器可以采用 Servlet 容器支持并发的页面访问，效率极佳；客户端对服务器的访问完全对立，一个客户端可以任意浏览多个 Web 服务器的内容，而且客户端是免维护的。但是基于 B/S 模式的服务子系统 WebS 依赖 HTML 页面的展现和嵌入的脚本处理能力，其功能和交互特性与 C/S 模式下的客户端相比要逊色不少。

2 异步模型构架技术方案

如前所述，基于 B/S 模式的 Web 构架较基于 C/S 模式的 Web 构架具备很多的优势，而 W3C 发布的 SVG 语言规范，以及异步模式技术 Ajax (Asynchronous JavaScript and XML) 为构建这种架构的服务子系统提供了技术支撑。基于这一思考，研究设计基于 B/S 模式的适合电力能量管理系统使用的服务子系统 WebS 架构方案。在本方案中，SVG/XML 是信息的主要载体，而 HTTP 服务器的搭建与 Ajax 技术的使用则是本设计方案的主要研究内容。

基于 B/S 模式的 Web 异步模型有 3 个重要的环节，如图 4 所示。(1) Web 服务器侧的 HTTP 服务器的搭建与服务提供；(2) 浏览器侧的展示技术。(3) 异步脚本通信与页面更新。

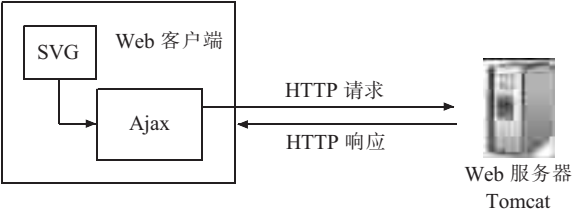


图 4 基于 B/S 模式的服务子系统 WebS 构架

2.1 浏览器展示技术

SVG 的开放性、高效性、灵活性、可扩展性和可

交互性等多方面特性使其具备了成为该系统中承载模型与图形表达以及交互展示的能力^[2]。SVG 将矢量图形的表达技术引入,使 SVG 文件很小,下载浏览更加便捷,而且采用客户机本地解释生成图形的方式,既充分利用了客户机的资源,又减轻了服务器端的负担,可以解决网络瓶颈问题^[3-5]。由于 SVG 是解释性语言,所以在浏览时遇上很大的图形文件,可以以图形元素为单位,接收多少,显示多少,类似与 JPEG 的渐进显示。SVG/XML 既是一种语言,也是一种技术。作为一种语言,它具有易读、可校验、方便修改等特性;作为一种技术,它为计算机领域提供了一种标准化的信息载体,方便地进行信息的共享与交互^[6,7]。加之大量业界厂商对 SVG/XML 作为主流格式的文本载体进行支持,大大扩展并提高了其解析能力、校验能力、传输能力。

2.2 异步脚本通信设计

SVG 的矢量图形格式提供了描述电力系统中设备的形状和外观的能力。但是静态的图形文件对于一个系统的展示功能是远远不够的,还需要图形中的数据信息能够随着系统实际数据的变化而变化。Ajax 技术和 DOM 的结合使用使得这一难题得到了解决。Ajax 技术直接提供了解决了异步通信与部分更新页面的解决方案。在基于数据的应用中,用户需求的数据如联系人列表,可以从独立于实际网页的服务端取得并且可以被动态地写入网页中。由于 Ajax 依赖浏览器的 JavaScript 和 XML,浏览器的兼容性和支持的标准也和 JavaScript 的运行时性能一样重要^[8]。

服务子系统 WebS 利用 Ajax 完成请求发送并取回必需的数据,并在客户端采用 JavaScript 处理来自服务器的响应。因为在服务器和浏览器之间交换的数据大量减少,应用响应得更快。同时很多的处理工作可以在发出请求的客户端机器上完成,所以 Web 服务器的处理时间也减少了。此方案的最大优点,就是能在不更新整个页面的前提下维护数据。这使得 WebS 的客户端更为迅捷地回应用户动作,并避免了在网络上发送那些没有改变过的信息。

服务子系统 WebS 利用 JavaScript 来传递用户界面上的数据到服务端并获取结果。XMLHttpRequest 对象用来响应通过 HTTP 传递的数据,一旦数据返回到客户端就可以立刻使用 DOM 将数据修改到基于 SVG 的网页上。XMLHttpRequest 对象在大部分浏览器上已经实现而且拥有一个简单的接口允许数据从客户端传递到服务端,但并不会打断用户当前的操作。使用 XMLHttpRequest 传送的数据可以是任何格式,而在本文的研究设计中采用了基于

XML 的 SVG 数据格式。

2.3 动态更新网页画面

前端展示的图形页面选择 SVG 作为嵌入对象,在载入静态画面后,需要结合实际应用系统的数据变化进行网页画面的动态更新。如前所述,其数据请求由 JavaScript 脚本函数通过 XMLHttpRequest 对象组织发出,当数据返回后,数据的刷新依赖则是文档对象模型 DOM 来完成。

DOM 可以看作是一组应用程序接口 API,它把 HTML 文档、XML 文档等看成一个文档对象,在接口里面存放的是对这些文档操作的属性和方法的定义,若编程语言实现了这些属性和方法,就可以对文档对象中的数据进行存取,并且利用程序对数据进行进一步的处理。由于 DOM 是基于信息层次的,因而 DOM 被认为是基于树或基于对象的。通过 DOM 接口,应用程序可以是任何时候范围 XML 文档中的任何一部分数据。SVG 也是基于 XML 的,所以与 DOM 也有着密切的关系。

服务子系统 WebS 即通过操纵 SVG DOM 来与 SVG 图形进行交互。当浏览器利用 DOM 载入了 SVG 文件后,就会在内存中按照 SVG DOM 构造一棵节点树。通过遍历 DOM 树,可以获取画面的请求数据集,并通过异步数据访问进行实时数据的获取。在数据返回后,利用 DOM 将解析的数据内容逐个的更新修改节点树上的相应节点,实现画面数据的动态刷新、状态着色、电气岛着色等展示功能,让静态的图形动起来。具体流程如图 5 所示。在这一过程中,所有的软件操作都是基于内存中的 DOM 节点树,原始的 SVG 文件不会发生任何变化。

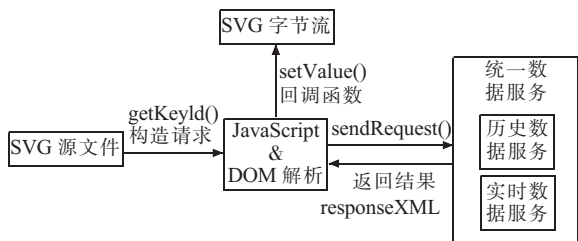


图 5 SVG 数据转换流程

2.4 数据刷新服务

对动态数据(如:量测量等)的处理,可以利用 DOM(文档对象模型)接口操纵 SVG 文档来实现数据的动态刷新。通过脚本语言定时调用文档对象模型的接口,获得画面上所有元素及其属性和属性值,通过 DOM 方式将整个 XML 表述的 SVG 格式的图形文件映射到内存中。内存中每个对象中属性值的变化可以通过订阅机制向服务端定制。

例如开关设备的动态变化可以通过 JavaScript 脚本来实现。(1) 申明 g_svgDoc 为 SVG 的文档对

象, g_svgRoot 为根节点; (2) 定位到指定的节点; (3) 修改 use 节点的属性, 引用该节点的不同状态。

类似的, 动态数据也可以进行修改。 (1) 申明 g_svgDoc 为 SVG 的文档对象, g_svgRoot 为根节点; (2) 定位到指定的节点; (3) 修改文本节点的值。通过这些步骤就可以简单地表达对指定节点进行的修改。

数据的获取则是实现的另一个关键。目前采用了 Ajax 技术来实现。比如上述 SVG 文件可以通过如下步骤来获取数据。 (1) 创建 Ajax 请求对象; (2) 将所有的 key_id 信息拼接起来字符串方式发送给 SvgDataServlet 程序。数据信息的请求发给服务器端的 Java Servlet 程序 (SvgDataServlet), 该程序在 doPost 方法中负责解析请求, 并进一步将 key_id 通过 CORBA 发送给后台服务获取系统数据, 数据获取之后返回给 Web 客户端, 客户端收到结果之后自动调用回调函数进行数据的刷新。以上的处理流程如图 6 所示。

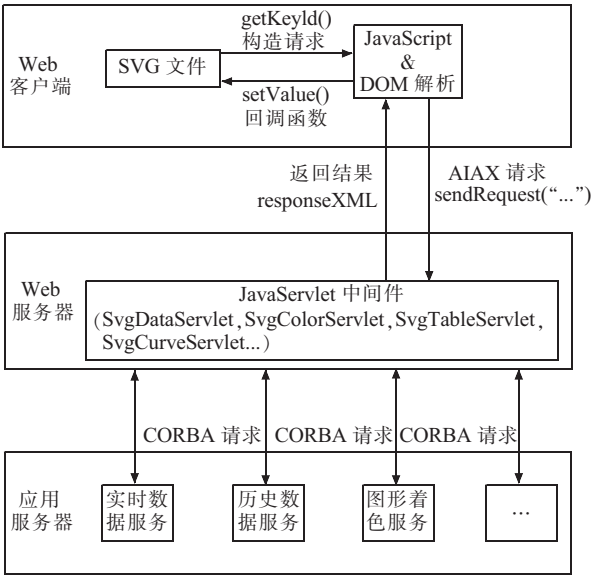


图 6 数据刷新的处理流程

3 结束语

本文针对基于 C/S 模式与基于 B/S 模式构架进行服务子系统 WebS 的搭建进行优缺点比较, 发现实现一套基于 B/S 模式架构的服务子系统 WebS 显得尤为重要。且利用多种计算机技术提出并实现了一套完整的基于 B/S 模式架构的设计方案。首先, 对服务子系统 WebS 的系统架构进行分析, 并对几种常用的系统设计模式进行比较, 提出了基于 B/S 模式架构实现服务子系统 WebS 的设计方案。其次, 通过选取 Servlet 容器技术搭建 HTTP 服务, 以 W3C 标准 SVG 作为前端展示语言与格式, 并利用 Ajax 技术实现客户端与服务器的通信, 从而建立浏览器与服务器之间的异步访问模型。最后, 结合异步访问模型, 建立服务子系统 WebS 的服务实现机制, 对前端展示所必须的数据刷新服务、事件响应等进行了详细的研究设计。

参考文献:

[1] 蒋元晨,潘正钰. 电力管理信息系统中图形的 Web 发布[J]. 电力系统自动化,2003,27(18):61-64.
[2] 秦 华,高毅雄,王康元. 能量管理系统中图形 SVG 导出的实现[J]. 继电器,2006,34(8):70-72.
[3] 黄凯伟. SVG 开发实践[M]. 北京:电子工业出版社,2008.
[4] 李亚平,姚建国,黄海峰,等. SVG 技术在电网调度自动化系统中的应用[J]. 电力系统自动化,2005,29(23):80-82.
[5] 刘 啸. 基于 XML 的 SVG 应用指南[M]. 北京:北京科海电子出版社,2001.
[6] 石东源,卢炎生,王星华,等. SVG 及其在电力系统软件图形化中的应用初探[J]. 继电器,2004,32(16):37-40.
[7] 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.
[8] 阿斯利森,舒 塔. Ajax 基础教程[M]. 金 灵译. 北京:人民邮电出版社,2006.

作者简介:

薛 蕾(1980-),女,江苏南京人,工程师,从事科技信息工作。

Design of EMS Web Service (Webs)

XUE Lei

(Jiangsu Electric Power Company Research Institute, Nanjing 211103, China)

Abstract: The web service (WebS) system is one of the important subsystems of EMS, and it has the functions such as information dissemination, information inquiry and service management. In the paper, the design methods of EMS web service (WebS) system were discussed. Based on the structures of Webs, the C/S mode and the B/S mode were analyzed and compared, the appropriate technology was searched for the design of EMS WebS in the asynchronous invocation B/S mode. Eventually, the server architecture of SVG&Ajax&HTTP was found, and the information refresh and response were realized through the web browser using HTML and JavaScript.

Key words: web service (WebS); energy management system (EMS); B/S mode; asynchronous invocation