

一种基于 Ajax 和 UserData 浏览器行为的 Web 系统数据存取交换模型

A Model of Web System Data Access and Exchanging
Based on Ajax and UserData Behavior

李 强 陈遵德 (顺德职业技术学院计算机技术系 广东顺德 528300)

摘 要: 传统 Web 系统数据存取、处理集中于 Web 服务器,无法满足用户比较多、人机交互次数较频繁、存取数据量较大的 Web 系统的实时性要求。基于 Ajax 和 UserData 浏览器行为的 Web 系统数据存取交换模型使 Web 服务器、浏览器和浏览器端计算机三者之间都能进行数据交换,可以充分利用性能较高的浏览器端计算机的数据存取、处理能力,减小了 Web 服务器负载,提高了 Web 系统性能。

关键词: Ajax UserData 浏览器行为 Web 数据模型

1 引言

由于传统 Web 系统的数据存取、处理集中于 Web 服务器,所以当 Web 系统用户比较多、人机交互次数较频繁、存取数据量较大时,就会对 Web 系统的实时性形成极大的挑战。例如 Web 考试系统由于用户比较多、人机交互次数频繁、存取数据量大就面临实时性问题。

通过 Ajax 和 UserData 浏览器行为可以使 Web 服务器、浏览器和浏览器端计算机三者互相交换数据,将浏览器需要重复读取或更新的数据暂存到浏览器端计算机,充分发挥浏览器端计算机的数据存取、处理能力,减小 Web 服务器负载。本文提出了基于 Ajax 和 UserData 浏览器行为的 Web 系统数据存取交换模型,并进行了具体应用,实现了实时通用计算机课程 Web 考试系统。

2 系统数据存取交换模型

基于 Ajax 可以构建以浏览器为主体的 Web 系统数据交换模型,基于 UserData 浏览器行为可以构建以浏览器端计算机为数据暂存器的 Web 数据存取模型,基于 Ajax 和 UserData 浏览器行为可以构建 Web 服务器、浏览器和浏览器端计算机三者可以互相交换数据,并充分发挥浏览器端计算机的数据存取、处理能力的

Web 数据存取交换模型。

2.1 基于 Ajax 的 Web 系统数据交换模型

传统的浏览器与 Web 服务器数据交换模型如图 1 所示,浏览器读取数据的过程是:浏览器首先请求服务器页面,Web 服务器响应浏览器的请求,并执行服务器页面,最后将页面执行结果传送给浏览器。浏览器存放数据的过程是:浏览器首先通过表单页面收集数据,然后请求服务器页面,Web 服务器响应浏览器的请求,并执行服务器页面,从而完成数据的存放。可见,数据交换中的主体是 Web 服务器,而浏览器只是数据显示和收集的工具。

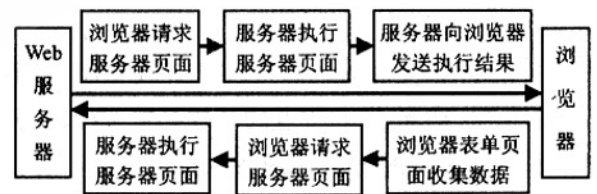


图 1 传统 Web 系统数据交换模型

Ajax 即异步 JavaScript 和 XML (Asynchronous JavaScript and XML),它是现有的 DHTML、CSS、COM、XML 等技术的综合应用。其核心思路是浏览器使用浏览器端脚本语言 (JavaScript 或 VBScript) 以异步方式向 Web 服务器发出服务器页面请求,Web 服务器执行服

务器页面后生成 XML 文档形式的执行结果并通知浏览器,最后浏览器端使用脚本语言分析处理服务器传送过来的 XML 文档形式数据^[1]。它提供了一种全新的浏览器与 Web 服务器交换数据模式,也使 Web 服务器与浏览器端计算机进行数据交换成为可能。

基于 Ajax 的浏览器与 Web 服务器数据交换模型如图 2 所示,浏览器读取数据的过程是:浏览器使用脚本语言请求服务器页面,服务器响应浏览器请求,执行服务器页面之后通知浏览器,浏览器使用脚本语言读取并处理服务器传送的 XML 文档形式的数据。浏览器存放数据的过程是:浏览器首先使用脚本语言收集数据,然后使用脚本语言请求服务器页面,Web 服务器响应浏览器的请求,并执行服务器页面,从而完成数据的存放^[2]。可见,数据交换中的主体是浏览器,并且这种交换模式充分发挥了浏览器的数据处理能力。

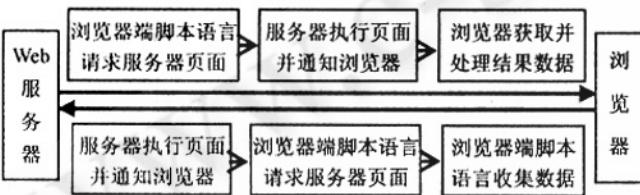


图 2 基于 Ajax 的 Web 系统数据交换模型

2.2 基于 UserData 浏览器行为的 Web 系统数据存取模型

传统的 Web 系统数据存取模型如图 3 所示,浏览器从 Web 服务器读取数据,也将数据存放到 Web 服务器,浏览器本身不保存任何数据,即使浏览器需要之前已经读取的数据,也要再次从 Web 服务器获取。对于访问用户数少,人机交互次数少、存取数据量小的系统,传统的 Web 系统数据存取模型可以胜任,但是对于访问用户比较多、人机交互次数较频繁、存取数据量较大时系统,传统的 Web 系统存取模型将会产生较长延时,无法满足用户的实时性要求。

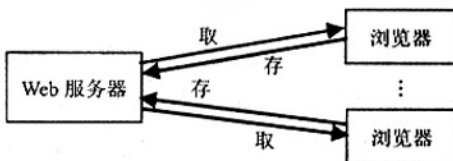


图 3 传统 Web 系统数据存取模型

UserData 浏览器行为是 Internet Explorer 的一个内置组件,用户不需对浏览器进行任何设置就可以使用,它可以让浏览器使用浏览器端脚本语言 (JavaScript 或 VBScript) 将数据以 XML 文件形式存放到浏览器端计算机中,并可以让浏览器跨会话访问^[3]。

基于 UserData 浏览器行为的 Web 系统数据存取模型如图 4 所示。浏览器将所需数据从 Web 服务器一次性获取,并通过 UserData 浏览器行为以 XML 文件形式存放到浏览器端计算机,然后浏览器从其所在的计算机上获取数据,并将更新数据也存放在其所在的计算机上,最后一次性存放到 Web 服务器上。

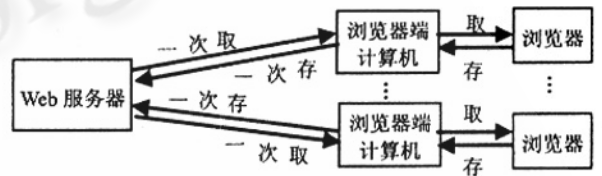


图 4 基于 UserData 浏览器行为的 Web 系统数据存取模型

2.3 基于 UserData 浏览器行为的 Web 系统数据存取模型

基于 Ajax 和 UserData 浏览器行为的 Web 系统数据存取交互模型如图 5 所示,浏览器、浏览器所在计算机、Web 服务器三者之间都可以交换数据,Web 服务器与浏览器端计算机基于 Ajax 和 UserData 进行数据交换,其数据操作主要是浏览器一次性获取需要重复读取的数据,一次性存放需要重复更新的数据;Web 服务器与浏览器基于 Ajax 进行数据交换,其数据操作主要是浏览器读取不需要重复读取的数据,存放不需要重复更新的数据;浏览器与浏览器端计算机基于 UserData 进行数据交换。

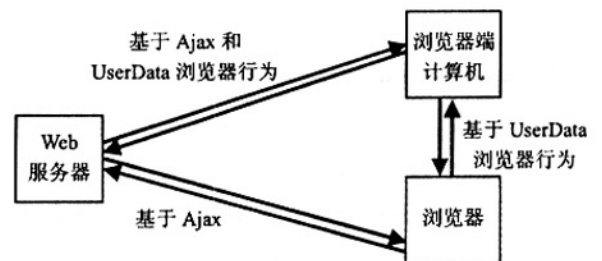


图 5 基于 Ajax 和 UserData 的 Web 系统数据存取交换模型

3 具体实现

基于 Ajax 和 UserData 浏览器行为的 Web 系统数据存取交互模型使浏览器、浏览器端计算机、Web 服务器三者之间都可以交换数据,其具体实现主要体现在三者交换数据的实现上。

3.1 浏览器与 Web 服务器交换数据的实现

Web 服务器与浏览器基于 Ajax 进行数据交换,其实现过程如下:

(1) 对于读取数据的服务器页面,要将其结果数据改造成 XML 形式。

(2) 使用浏览器端脚本语言 (JavaScript 或 VB-Script) 实现 Web 服务的请求程序。具体过程是首先创建 XMLHttpRequest 对象,然后使用 XMLHttpRequest 对象发出异步方式的 Web 服务请求^[2]。

(3) 如果需要读取服务器页面结果数据,使用浏览器端脚本语言实现 XML 文档形式结果数据的解析程序。具体过程是首先使用 XMLHttpRequest 对象取得服务器页面的 XML 文档形式的执行结果,然后创建 XMLHttpRequest 对象,并使用该对象解析 XML 文档^[3]。

3.2 浏览器与浏览器端计算机交换数据的实现

浏览器与浏览器端计算机基于 UserData 浏览器行为进行数据交换,其具体实现过程如下:

(1) 在页面中声明 UserData 浏览器行为样式,形如:

```
<STYLE >
.StoreUserData { Behavior: URL( #Default#UserDa-
ta) ;}
</STYLE >。
```

(2) 在页面表单中创建隐藏控件并应用上面定义的 UserData 浏览器行为样式,形如:

```
<INPUT CLASS = " StoreUserData" TYPE = " Hid-
den" ID = " TextInfo" >。
```

(3) 通过浏览器端脚本语言 (JavaScript) 使用 UserData 浏览器行为对象 TextInfo 存取数据。存放数据的过程是:首先调用 textInfo 的 setAttribute 方法赋值,然后调用 textInfo 的 save 方法将数据存以 XML 文档的形式放到浏览器所在计算机上。读取数据的过程是:首先调用 textInfo 的 load 方法从浏览器所在的计算机上读取 XML 文件,然后调用 textInfo 的 getAttribute

方法读取具体值^[3]。

3.3 Web 服务器与浏览器端计算机交换数据的实现

Web 服务器与浏览器端计算机基于 Ajax 和 UserData 浏览器行为进行数据交换,读取数据的具体实现过程如下:

(1) 使用 Ajax 请求服务器页面,并取得服务器页面 XML 形式的数据结果。

(2) 使用 UserData 浏览器行为将数据存放到浏览器所在计算机上。

存放数据的具体实现过程如下:

(1) 使用 UserData 浏览器行为从浏览器所在计算机上读取数据。

(2) 使用 Ajax 请求服务器页面,并将数据传送给服务器页面。

(3) 服务器页面取得参数值执行之后完成数据存储。

4 结束语

通用计算机课程 Web 考试系统承担我院所有计算机课程的客观题考试,由于考生人数多、人机交互频繁、存取数据量大,所以系统的实时性成了系统设计与实现的最大难题,本文通过应用基于 Ajax 和 UserData 浏览器行为的 Web 系统数据存取交互模型,成功地解决了这一难题。由于考生的试卷信息是固定的,所以考试开始时,系统将考生的所有试卷信息一次性读取并存放于考生所在计算机上,考生考试时,直接从其所在计算机上读取试卷信息。另外,由于考生会对答题信息不断进行修改,所以系统在考试过程中将考生答题信息暂存在考生计算机上,考生交卷时,系统再将考生答题信息一次性存放到 Web 服务器上。

通用计算机课程 Web 考试系统目前已投入运行,可以让 2000 学生同时考试,并且考试过程中没有延时。

参考文献

- 1 Alexei White. Ajax accelerates Web applications. Network World, 2006, 23(2): 33 - 34.
- 2 Andrew Turner. Adding Ajax to a Website. MacTech Magazine, 2006, 22(1): 48 - 50, 52, 54, 56, 58.
- 3 <http://msdn.microsoft.com/library>