

文章编号: 1005-0523(2007)05-0074-03

AJAX 技术在网络考试分析中的应用

丁振凡

(华东交通大学 信息工程学院, 江西 南昌 330013)

摘要: 结合网络教学测试系统中的学生解答分析程序的设计, 讨论了 AJAX 技术在实现图文分析处理中的具体应用, 系统利用 XML 进行数据的封装处理, 利用 DHTML 技术实现页面的动态显示。

关键词: AJAX; XML; 考试分析; JavaScript; DOM

中图分类号: TP311

文献标识码: A

0 引言

传统 Web 应用中表单的提交是采用页面重载的办法, 也就是请求提交后等待服务器的响应, 根据响应页面刷新浏览器的显示。而很多情况下, 用户的页面内容带有一定的稳定性, 也就是经过某个互动操作, 仅变化页面上的某个局部区域。在此过程中可能还需要与服务器交换信息, 但是交换的信息不是产生新的页面, 而是传递数据信息。Ajax 技术的引入为创建此类更加动态的 Web 应用程序提供了方便, 提高了整个应用的性能。

教学测试目的是测试学生的水平, 了解其对知识的掌握情况, 与此同时要发现学生的不足, 进行针对性指导。因此, 在学生测试完成后, 对学生的解答进行统计分析对于发现教学中的问题, 改进教学具有很大的实际意义。为此笔者在网络教学平台软件设计中将教学测试分析作为测试子系统的一个辅助功能。它能对考试试卷中学生对每道题的解答情况进行统计分析, 计算各种解答选项学生的选择比例, 用条形图的形式直观地显示在 Web 页面中。目前, 国内外网络考试的开展已日益增多, 但考试分析软件的设计相对很少, Web 图文显示处理大多采用 Applet、Flash、VML、SVG 等技术实现, 笔者曾采用 Java Applet 实现该功能。由于 Java Applet 在运行时下载较费时, 为此笔者采用 AJAX 技术对该应用进行了重新设计。同时也对 AJAX 技术进行 Web 图文信息处理进行了有趣的探索。

1 AJAX 简述

AJAX 是 web2.0 的核心技术, AJAX 编程的特点是将应用的大量任务放在客户浏览端完成, 减轻服务器的处理负担, 提高页面更新效果。其基本思路是利用客户浏览器页面上的事件触发执行 JavaScript 脚本代码, 在保持页面不刷新的情况下, 利用 XMLHttpRequest 对象异步读取服务器的数据, 在对数据进行分析的基础上, 利用 DHTML 技术实现页面内容的动态改变。

2 考试数据分析的设计与实现

2.1 设计思路与效果

图 1 为单选题的统计分析界面。教师可以通过上部的试题类型选择按钮切换分析题型, 通过底部的翻动试题按钮浏览各道试题的解答统计, 试题内容和标准答案在右边的文本域中显示, 中央采用条形图的方式显示各选项的统计分析结果所占比例。这里图形的显示直接利用图片的高度调整来实现, 图形中的坐标轴则是采用表格标记样式设计实现。为了实现数据分析的实时动态变化, 利用 AJAX 技术访问服务器, 获取分析试题的内容和学生解答信息, 用 DHTML 技术实现页面动态显示处理。基本设计思路如图 2 所示:

其核心是在 Javascript 事件代码中通过 XMLHttpRequest 对象发送请求访问服务器, 并将返回的结果借

收稿日期: 2007-08-20

基金项目: 2006 年江西省教改项目, 项目名称为“研究性学习支撑环境的建设与应用”(编号: JXJG-06-5-33)

作者简介: 丁振凡(1965-), 男(汉族), 江西丰城人, 教授, 硕士生导师。研究方向: 电子商务、网络教学。

助 DHTML 技术在页面上动态显示. 笔者采用 XML 封装数据信息, 利用 XML 文档对象模型实现数据分析处理. 该机制具有数据传递少, 页面显示效果好的特点.

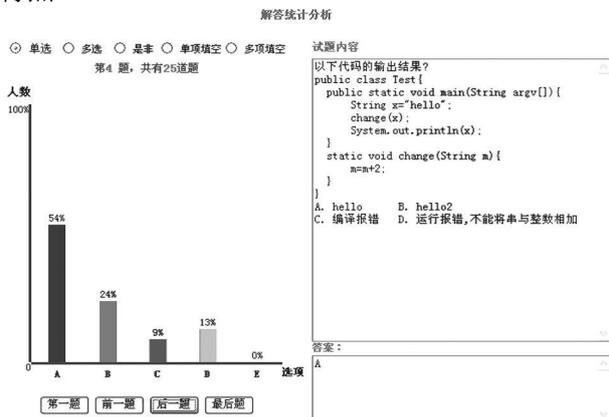


图 1 单选题的解答统计分析界面

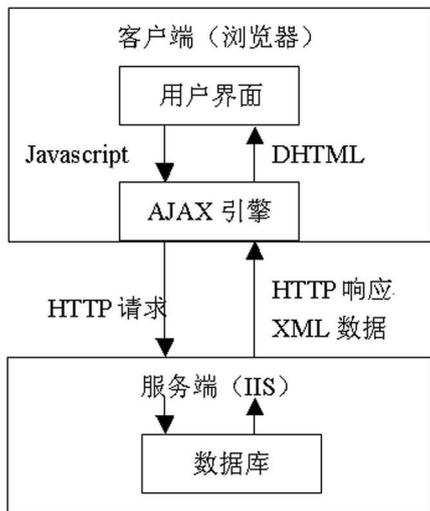


图 2 应用工作原理

2.2 分析数据的获取与显示处理

2.2.1 服务端数据的产生与发送

每次客户端的分析请求将提交给 Web 服务端, 由 ASP 程序 (get.asp) 根据试题序号和题型从数据库中获取数据并进行统计计算, 将结果发送给客户端. 为节省篇幅, 将从数据库获取数据以及统计计算的代码略去, 重点介绍响应消息的发送. 为了便于客户端对数据的提取, 响应消息封装为 XML 格式, 这里有一个关键问题是中文的处理, 注意将响应字符集设置为 gb2312. 程序中利用 XMLDOM 实现消息文档的创建, 消息的根结点为 message, 试题内容 (content)、标准答案、各选项的人数比例、试题数等消息均作为根结点下子结点进行封装. 关键代码如下:

```
Response.ContentQuestion-type="text/xml"
Response.CharSet="gb2312"
```

```
set xmlDoc = Server.CreateObject ("Microsoft.XMLDOM")
```

```
xmlDoc.loadxml("<? xml version = '1.0' & encoding = 'gb2312' ? ><message ></message >")
```

```
set tmpNode = xmlDoc.createElement ("content", "")
```

```
xmlDoc.documentElement.appendChild (tmpNode) '将新建结点作为根结点下的子结点
```

```
tmpNode.text = content '将试题内容作为结点文本写入结点
```

```
...
response.write xmlDoc.xml '将生成的 XML 文档发送给客户端
```

2.2.2 利用 XMLHTTP 实现数据的传输

在客户端利用 Javascript 代码实现动态交互处理, 程序中将需要共享的变量定义为全局变量, xmlhttp 和 xmlDoc 两个对象引用变量也定义为全局变量, 避免函数调用中反复创建. 这里, 将当前试题的数据获取和显示处理封装在自定义函数 fetch() 中, 其关键是利用 xmlhttp 对象发送 URL 访问请求获取试题数据, 并利用该对象的 responsetext 属性获取响应数据, 将响应消息加载为 XML 文档对象进行消息的提取分析, 然后, 利用 DHTML 技术实现消息的显示处理.

```
<script language="javascript">
var question-no=1; //当前试题序号
var question-type=1; //题型
var amount=0; //当前类题目总数量
var xmlhttp = new ActiveXObject ("Microsoft.XMLHTTP");
var xmlDoc = new ActiveXObject ("Microsoft.XMLDOM");
```

```
function fetch() {
myurl="get.asp? number=" + question-no + "& type=" + question-type; //根据题号和题型访问服务器
```

```
xmlhttp.Open("POST", myurl, false);
xmlhttp.send(null); //发送请求
xmlDoc.loadXML(xmlhttp.responsetext); //获取响应
```

```
content = xmlDoc.selectSingleNode ("//content").text;
```

```
//读取试题内容结点的文本
.....//根据获取的数据进行页面的显示处理, 见 2.2.3
```

```
}
.....//其他函数
```

```
</script >
```

2.2.3 数据信息的显示处理

利用 DHTML 技术实现的数据信息的显示处理, HTML4.0 的新标准中将页面的元素均可以看作对

象,通过改变对象的属性和内容可实现页面的动态变化.本例涉及显示处理对象有3类:

(1) 显示试题的文本域,其 value 属性存放试题内容,通过改变该属性值实现当前试题内容的显示,例如:

```
document.contentarea.value = content;
```

(2) 试题序号、百分比等内容的显示,可通过 span 标记定义显示内容,例如,标识名为 p1 的标记代表一个选项的百分比部分:

```
<span id="p1"></span>
```

通过改变标记对应的 HTML 内容实现显示动态变化.

```
document.getElementById("p1").innerHTML = rate + "%";
```

(3) 图形的显示调整

条形图用图片显示,其高度变化可通过改变图形显示的高度实现.

```
document.getElementById("aimg").height = h;
```

其中,aimg 为某选项对应的图形对象的标识名.

2.3 事件驱动处理

2.3.1 文档加载时的事件处理

开始显示文档时,默认看到的是第一道单选题的学生解答分析数据,在 JavaScript 的全局变量中给代表题型(question-type)的变量初值定为单选,给代表题目序号(question-no)的变量初值定为1,为了在文档加载时显示默认试题,可以通过文档对象的 onload 事件执行 fetch() 方法实现,如下所示:

```
<body onload="fetch()">
```

2.3.2 翻动试题的处理

在分析过程中,可以通过“下一题”、“上一题”等按钮查看各道试题的解答统计.与试题翻动按钮点击相关的事件代码要解决的问题就是改变试题的序号,然后执行前面定义的 fetch() 方法实现试题的显示处理.以下为对应“下一题”按钮的事件处理函数:

```
function next() { // 点击下一道题
    if (question-no < amount) // 是否到达最后位置
        question-no = question-no + 1; // 题号加 1
}
```

```
fetch();
}
```

2.3.3 试题类型的切换处理

可以通过“单选题”、“多选题”等按钮查看各类试题的解答统计.相应的按钮事件处理代码的设计包括如下任务:首先要改变代表题型、题号的全局变量,另外根据题型的不同要改变页面上的某些区域的显示,单选类试题包括5个选项的统计,其他类试题则只负责对错统计,因此只有对错两个选项,处理办法与百分比的显示处理相同,相关处理代码限于篇幅略.最后,调用 fetch() 方法显示该题型的第1道题的分析结果.

```
function change(type) {
    question-no = 1; // 第 1 道题
    question-type = type;
    if (type == 1) { // 1 代表单选题
        ..... // 单选的页面变化处理
    }
    else {
        ..... // 其它类试题的页面变化处理
    }
    fetch(); // 显示试题和分析结果
}
```

3 结语

利用 AJAX 技术实现网络教学平台中考试分析处理的方案具有响应快、应用效率高、显示效果好的特点,笔者利用 XML 技术实现数据信息的包装和分析处理,利用 DHTML 技术实现用户与页面的交互和内容的动态显示处理.其实现思路对 AJAX 应用开发具有广泛的参考价值.

参考文献:

- [1] 王星. 基于 AJAX 技术的 Web 模型在网站开发中的应用研究[J]. 微计算机信息, 2006, 27: 205~207.
- [2] 吕林涛. 基于 AJAX 的 Web 无刷新页面快速更新数据方法[J]. 计算机应用研究, 2006, 11: 199~201.

The Application of AJAX in the Analysis of E-learning Examination

DING Zhen-fan

(School of Information, East China Jiaotong University, Nanchang 330013, China)

Abstract: Combined with the design of e-learning examination analysis, the paper discusses the application of AJAX in the process of text and bar graph. It also introduces encapsulation of the data with the help of system XML, and dynamic display process of web page through implement of DHTML techniques.

Keywords: AJAX; XML; analysis of exam; JavaScript; document object model