

以数据为中心的异步 Web 应用研究

肖飒

武汉理工大学信息学院, 武汉 (430070)

E-mail: sashaosashao@163.com

摘 要: 本文介绍了一种 Ajax (Asynchronous JavaScript and XML, 即异步的 JavaScript 和 XML) 与服务器的交互模式, 从应用的角度思考, 提出了一种基于数据为中心的交互模式, 提供了如何响应 XML 和 JSON 的数据格式的范例, 然后基于这个模式, 设计了 Ajax 与一个 MVC 框架的整合, 实现了一个异步树结构。最后, 评价了这个模式的优缺点。

关键词: Ajax, JSON, 异步

中图分类号: TP393.09

1. 引言

当今的因特网界已经认识到, 用于显示文档内容的基本 Web 浏览器技术, 已经无法胜任更高层次面的任务了。Ajax (Asynchronous JavaScript+XML, 即异步 JavaScript+XML) 是一个相当新的名字, 是由 Adaptive Path 的咨询顾问首先提出来的^[1]。Ajax 的一些部分, 以前称为动态的 HTML (Dynamic HTML) 和远程教本 (remote scripting), 而 Ajax 的名字更加简介。Ajax 让人耳目一新。技术上, Ajax 极大的发掘了 Web 浏览器的潜力, 开启了大量新的可能性。商业上, Google 和其他主要的参与者通过眼花缭乱的 Ajax 应用, 让普通大众对于新一代 Web 应用充满了期待^[2]。目前, 在大多数 B/S 模式的 Web 应用中, 为了避免页面的臃肿和维护的困难, 服务器端的 Web 层应用采用 MVC 设计模式。所以, 若能充分利用 Ajax 固有的优越性, 又能很好的与现有的 MVC 设计模式想结合, 就能进一步完善 Web 应用, 使得 Web 不仅层次分明, 可重用性强, 具有高的健壮性和可伸缩性, 而且具有良好, 丰富的用户体验。

目前, 由于 Ajax 出现的比较晚。在 Web 数据的交换领域, 还没用专门的模式来知道 Ajax 与服务器端程序的交互。所以, 在模式语言在这个领域并没得到很好的发展。本文将从应用的角度出发, 提出一种以数据为中心的交互模式。并设计了 XML 和 JSON 两种格式作为示范。最后让 Ajax 与 WebWork 整合实现, 以 JSON 格式实现了一个异步树。

2. Ajax 技术

Ajax 本身并不是全新的技术, 而是一种集成方法: 包括异步, XMLHttpRequest, XML (Extensible Markup Language 可扩展标记语言)。DOM (Document Object Model 文档对象交换模型), JavaScript, CSS (Cascading Style Sheets 层叠样式表) 等。表 1 简要的介绍了这些技术, 以及他们的作用。

表 1 Ajax 的关键元素

JavaScript	JavaScript 是著名的教本语言, 用来嵌入在某种应用程序之中, Web 浏览器中嵌入的 JavaScript 解释器容许通过 JavaScript 程序与浏览器的许多功能进行交互, Ajax 应用就是使用 JavaScript 编写的。
CSS	CSS 为 Web 应用提供了一种可重用的可视化样式的定义方法, 它提供了简单而又强大的方法, 以一致的方式定义和使用可视化样式。在 Ajax 应用中, 用户界面的样式可以通过 CSS 独立修改。

DOM	DOM 以一种可编程对象的形式展现浏览器显示内容的树性结构, JavaScript 修改 DOM, Ajax 应用程序可以在运行时改变用户界面。
XMLHttpRequest 对象	XMLHttpRequest 对象允许 Web 应用开发者从 Web 服务器以后台活动的方式获取数据。数据格式通常是 XML。但也可以支持任何基于文本的数据格式。Ajax 的异步性就是通过 XMLHttpRequest 对象与服务器交互实现的。

可以看见 JavaScript 是 Ajax 的中心内容, 它就像胶水一样, 将其它技术粘连起来。通过 JavaScript 操作 DOM 来改变和刷新用户界面, 不断重新组织数据和数据的显示方式提供给用户, 来实现富客户端的效果^[3]。CSS 为应用提供了一致的外观。XMLHttpRequest 对象则用来异步通信, 在用户操作的时候提交用户的请求并且获取最新的数据信息。平常在 Google 上搜索的时候, 输入某个汉字, 就会得到一些提示来告诉用户那些字是经常被搜索的。这样就极大的方便了用户的搜索。

令人高兴的是, 所有的这些技术已经预先安装在绝大多数的现代 Web 浏览器上。因为所有现代的操作系统都会预先安装一个现代的浏览器。也就是说, 大多数的电脑和笔记本电脑都已经为运行 Ajax 应用程序做好了准备。比如微软的 IE, Mozilla 的 Firefox。

3. 以数据为中心的交互模式

3.1 概述

大多数情况下, Ajax 客户端希望和其他类型的前端共享数据, 例如 Java、.NET 智能客户端、蜂窝电话/PDA。这种情况下, 我们应该使用一种中立的数据格式而不是一套 JavaScript 指令。由自己的客户端代码而不是 JavaScript 来解析数据。

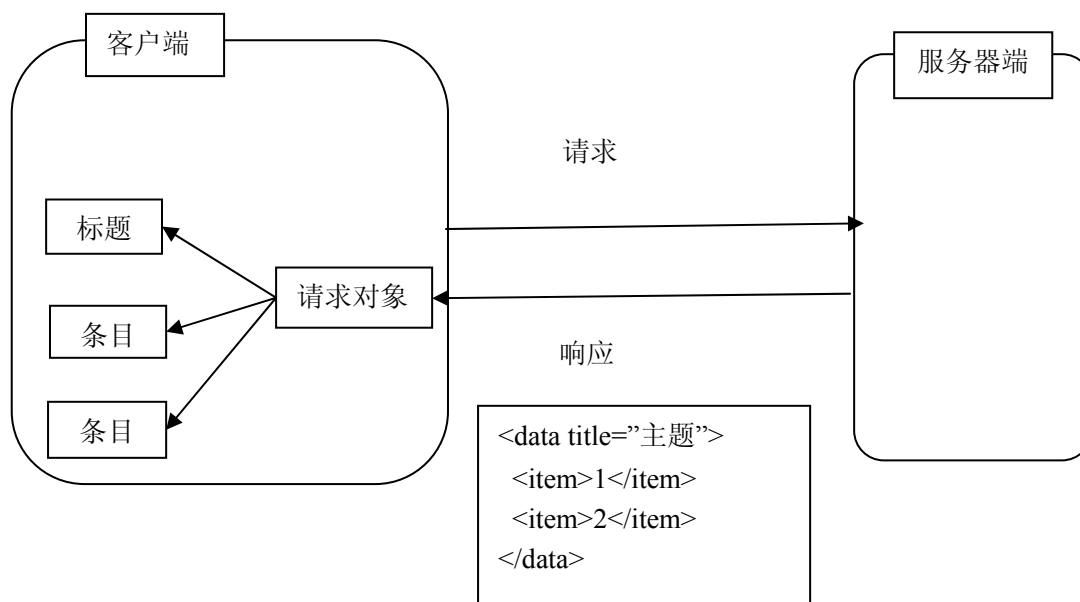


图 1 服务器返回的数据流 (这里是 XML), 它在客户端解析, 然后更新到用户界面

现在对原始的 Ajax 进行简单的包装成一个 DataLoader 对象, 以便下面使用。

```
var DataLoader = function(url, onload) {
    this.url = url;
    this.req = null;
    this.onload = onload;
    this.loadXML(url);
};
DataLoader.prototype={
    loadXML:function(url){
        if (window.XMLHttpRequest) {
            this.req = new XMLHttpRequest();
        } else {
            this.req = new ActiveXObject("Microsoft.XMLHTTP");
        }
        var loader = this;
        this.req.onreadystatechange=function(){
            loader.onReadyState.call(loader);
        }
        this.req.open("GET",url,true);
        this.req.send(null);
    },
    onReadyState:function() {
        var req = this.req;
        if (req.readyState==4) {
            if (req.status==200 || req.status==0) {
                this.onload.call(this);
            }
        }
    }
}
```

参数里的 `onload` 函数就是用来响应服务器发送回来的数据。下面有两个范例格式，我们将看到如何基于 `DataLoader` 对象来响应服务器数据，从而实现以数据为中心的交互模式。

3.2 使用 XML 数据格式

XML 是当今互联网时代非常重要的一种格式。Ajax 利用 `XMLHttpRequest` 对象，天生就对 XML 有着很好的支持。如果 `XMLHttpRequest` 对象接受到一个 XML 内容的类型，它会自动将响应数据解析为一个 DOM 结构。

假设现在服务器 `book.jsp` 要返回来一个表示图书的 XML 数据。

```
<books>
<book name="C 专家编程" author="Van Der Linden"/>
<book name="Java 编程思想" author="Bruce Eckel"/>
<book name="精通 JavaScript" author="John Resig"/>
</books>
```

借助于上节封装好的 `DataLoader` 对象，可以由如下的 XML 响应方式

```
function xmlSomethingHappens() { //这函数个是用户触发的
    new DataLoader("book.jspl", parseXML);
```

```
}  
function parseXML() {  
    var xmlDoc = this.req.responseXML;  
    var books = xmlDoc.getElementsByTagName("book")[0];  
    //下面就可以依照需求使用 books 对象做具体的显示逻辑了  
}
```

通过使用 XML，我们带到了对服务器层和客户层的更好分离。如果两端都能理解文档的格式，那么客户端和服务端端的代码就能独立于对象而修改。

3.3 使用 JSON 数据格式

XMLHttpRequest 能够接受任何形式的文本信息，所以我觉得这个对象的名字也许不是很恰当。一种将数据传输到 Ajax 客户端的有用数据格式是 JSON，它是表示 JavaScript 对象的良好方式。

和上一节一样的书库内容，但这次的格式是 JSON 格式，如下

```
("book":  
    {"name": "C 专家编程", "author": "Van Der Linden"},  
    {"name": "Java 编程思想", "author": "Van Der Linden"},  
    {"name": "C 专家编程", "author": "Van Der Linden"},  
)  
function jsonSomethingHappens() { //这函数个是用户触发的  
    new DataLoader("book.jsp", parseXML);  
}  
function parseXML() {  
    var jsonText = this.req.responseText;  
    var json = eval("(" + jsonText + ")");  
    var books = json.books;  
    //下面就可以依照需求使用 books 对象做具体的显示逻辑了  
}
```

如果在服务器端和客户端都有一个可用的 JSON 解释器，那么 JSON 会是比 XML 更好的选择，JSON 是更容易理解和操作。

4. 应用

可以说，基于数据的交互模式能够广泛应用于 Web 开发领域。我在上一个项目中，思科中国渠道大学中就使用 WebWork 一些特点，支持了 JSON 格式的传输。可以说是一个非常好的应用范例，下面看是怎么实现的。

WebWork 是 J2EE 领域的著名 MVC 框架，它有着能让开发者自行创造返回类型的功能。要让服务器响应支持 JSON，写一个 WebWork 的 ResultType 就满足要求。首先写一个 WebWork 的 Action 抽象类：

```
public abstract class AbstractJSONDataAction extends ActionSupport {  
    protected JSON jsonObject;  
    public String execute() {  
        this.jsonObject = generateJSONObject();  
    }  
    public abstract JSON generateJSONObject();  
}
```

```
}
```

下面就是 JSON 返回类型的 Result, 它的工作直接向客户端发送 JSON 格式。最后在 WebWork 里面配置这个返回类型。

使用的时候只配置返回类型是 JSON 就可以提示 WebWork 发送 JSON 数据格式。

```
<result name="success" type="json"/>。
```

作为客户端使用的 Ajax 的 Ext 框架, 它对 JSON 格式的数据已经做了良好的封装。最后的效果如图:



图 2 思科中国渠道大学后台管理异步树

5. 结论

以数据为中心的交互模式将会是大部分基于 Ajax 的 Web 应用的交互方式, 它使得客户端于服务器端的代码都能独立于对象修改。使 Web 应用层次分明, 一定程度上减轻了服务器的运算压力, 增加了 Web 应用服务器端的可维护性。

以数据为中心的交互模式主要局限性是它将解析的负担完全交给了客户端。因此, 客户端的代码会变得更加复杂。不利于客户端的维护。

参考文献

- [1] Dave Crane, Eric Pascarello, Darren James. 《Ajax in Action》[M]. 北京: 人民邮电出版社, 2007.3.
- [2] Ryan Asleson, Nathaniel T.Schutta. 《Ajax 基础教程》[M]. 北京: 人民邮电出版社, 2006.2.
- [3] Brett McLaughlin. 掌握 Ajax [EB/OL]. <http://www.ibm.com/developerworks/cn/xml/wa-ajaxintro1.html>, 2006.1.

The Research of Asynchronous Web Application Base on the Center of Data

Xiao Sa

Wuhan University of Technology, Wuhan (430070)

Abstract

This paper produce a pattern between Ajax(Asynchronous JavaScript and XML) and Web Server. From the perspective of Application., A pattern that is data-centred interaction has been proposed. Propose a example on how to response data type of XML and JSON. and base on this pattern, design a Integral of Ajax and MVC framework, implement an asynchronous tree. At last, the paper conclude the advantage and disadvantage of the pattern.

Keywords: Ajax JSON Asynchronous