

# 基于.NET的 Ajax 控件保值技术及其应用

谭建平, 王成良

(重庆大学软件学院, 重庆 400044)

**摘要:**在页面内使用 Ajax 技术与服务器通信可提高 Web 应用的响应速度, 但 Web 服务器控件引起的页面刷新导致整个页面回传后, 页面的视图状态处理模型并没有记录 Ajax 控件在页面变化之前的值, 造成 Ajax 控件值的丢失。该文提出一种状态保存机制, 将 Ajax 控件状态值存储到服务器的 Session 或者客户端的 userData 中, 使页面刷新后客户端的 Ajax 控件仍可保留原有值, 保证控件状态的持续。

**关键词:** Ajax 控件; userData 行为; 会话; 视图状态; 状态保存机制; .NET 技术

## Ajax Widget's Value Keeping Technology Based on .NET and Its Application

TAN Jian-ping, WANG Cheng-liang

(School of Software, Chongqing University, Chongqing 400044)

**【Abstract】**The process will improve the response speed of Web application by using Ajax technology when Web pages communicate with Web server, however, when the Web pages refreshed by Web server widget, the view state process model of the Web page records Web server widget's state value does not record the Ajax widget's value and leads to the loss of the Ajax widget's value. This paper presents a new state value storage mechanism, which makes Ajax widget's state value be stored in the Web server session object or the user data object of the client side, so that the Ajax widget of the client side can still hold the same value after the Web page refreshed, which ensures the Ajax widgets state continuance.

**【Key words】**Ajax widget; userData behavior; session; view state; state storage mechanism; .NET technology

### 1 概述

Ajax 是多种技术的综合, 在 XHTML 和 CSS 基础上, 使用 DOM 实现动态显示和交互; 使用 XML 和 XSTL 进行数据交换与处理; 使用 XMLHttpRequest 对象进行异步数据读取; 使用 JavaScript 绑定和处理所有数据。

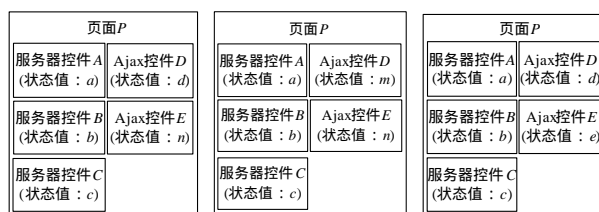
在应用 Ajax 之前, Web 站点强制用户进入提交、等待、重新显示的模式, 用户的动作总是与服务器的思考时间同步。Ajax 提供与服务器异步通信的能力, 使用户可从请求-响应的循环中解脱出来<sup>[1]</sup>。

与传统 Web 开发相比, Ajax 是完全的客户端技术。Ajax 运行机制向服务器发送并获取响应数据, 服务器端返回一个纯文本流, 再由客户端来处理。文本流可为 XML 格式、Html 片段、JavaScript 脚本或字符串。客户端使用 XMLHttpRequest 对象来请求页面, 服务器端处理后将结果文本返回给客户端, 此过程和普通的 Web 开发无任何区别。不同的是, 客户端在异步获取响应数据后, 不是直接将响应数据显示在页面上, 而是由客户端的 JavaScript 脚本处理后再将处理结果显示在页面上<sup>[1-2]</sup>。

在 .NET 中, Web 服务器控件的操作会引起整个页面的刷新, 页面的视图状态(ViewState)机制可保存 Web 服务器控件在刷新之前的值, 因此用户在刷新后仍可看到原状态值, 但视图状态处理模型并不保存 Ajax 控件在页面刷新之前的值。

假设页面 P 存在控件 A, B, C, D, E, 对应控件状态值为 a, b, c, d, e。A, B, C 为服务器标准控件; D, E 为 Ajax 控件。Ajax 控件的数据与状态由 Ajax 技术决定。页面 P 的 ViewState 记录了 A, B, C, D, E 的状态值, 如图 1(a)所示。单击任一 Ajax

控件 D, E 后, 状态值由 d, e 变为 m, n, 如图 1(b)所示。此时 ViewState 中记录的依然只是 D, E 的原状态值 d, e, 而非新状态值 m, n。单击任一服务器标准控件引起整个页面 P 回传后, 服务器会重载 P 并根据 ViewState 恢复状态值, P 上显示控件 A, B, C, D, E 的状态值 a, b, c, d, e, 而通过 Ajax 控件所得数据 m, n 就不被保存, 如图 1(c)所示。



(a) 页面初始状态 (b) Ajax 控件变化后状态 (c) 页面刷新后状态

图 1 控件状态值的变化情况

上述控件的状态值不仅可指控件的具体数据值, 还可表示数据值的一个标识, 如索引值等。为解决此问题, 须建立一套新的状态保存机制, 使页面刷新后客户端得以保存 Ajax 控件的状态值, 实现 Ajax 控件的保值。

### 2 Ajax 控件保值技术

.NET 中提供了特殊的服务器 Session 对象和客户端的 userData 对象, Ajax 控件保值技术的主要过程即是将 Ajax

**作者简介:**谭建平(1982-), 男, 硕士研究生, 主研方向: 软件工程与应用; 王成良, 教授

**收稿日期:**2007-11-08 **E-mail:** tjp112233@163.com

控件状态值存储到服务器 Session 对象或客户端的 userData 对象, 从而建立一套针对 Ajax 控件的状态保存机制。

### 2.1 Session 和 userData 的功能

Session 机制是一种服务器端的机制, 服务器使用一种类似于散列表的结构来保存信息。当需要为某个客户端的请求创建一个 Session 的时候, 服务器首先检查这个客户端的请求里是否已包含一个 Session 标识, 该标识称为 Session id(由服务器随机产生的一个由 24 个字符组成的字符串)。如果包含, 则说明之前已经为此客户端创建过 Session, 服务器就按照 Session id 把这个 Session 检索出来使用; 否则就为此客户端创建一个 Session 并生成一个与此 Session 相关联的 Session id。这个 Session id 将在本次响应中返回给客户端并保存。

一般情况下, 当服务器进程被停止或者重启时, 存储在内存里的 Session 会被清空。如果设置了 Session 的持久化特性, 服务器就会把 Session 保存到硬盘上, 以保证服务器进程重新启动后这些信息还能被再次使用<sup>[2-3]</sup>。

客户端的 userData 机制通过将数据写入一个 userData 存储区(userData store)来保存数据, userData 可将数据以 XML 格式保存在客户端计算机上。该数据将一直存在, 除非人为删除或者用脚本设置了该数据的失效期。userData 提供了一个比 Cookie 更具有动态性和更大容量的数据结构。userData 可通过 Session 机制为每个 Session 标识分配一个 userData 存储区。使用 save 和 load 方法将 userData 存储区数据保存在缓存(cache)中。userData 存储区数据被保存后, 即使浏览器关闭或刷新, 下一次进入该页面, 数据也能重新载入而不会丢失。

### 2.2 Ajax 控件保值过程

本文利用服务器 Session 机制来存储 Ajax 控件 D 的状态值, 用客户端 userData 机制来存储 Ajax 控件 E 的状态值, 具体过程如下:

单击 Ajax 控件 D 会引起页面的异步回传, 将其状态值  $d$  作为参数传到服务器中, 在服务器端根据参数  $d$  进行相关操作得到结果  $m$ , 用 Session 机制储存状态值  $m$ , 即  $\text{Session}["D"] = m$ , 并将结果返回客户端, 使 Ajax 控件 D 的状态得到更新, 如图 2 所示。

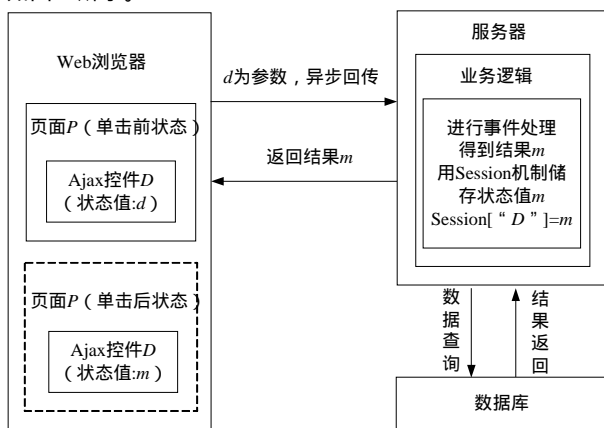


图 2 使用 Session 对象保值过程

单击 Ajax 控件 E 会引起页面的异步回传, 将其状态值  $e$  作为参数传到服务器中; 在服务器端根据参数  $e$  进行相关操作得到结果  $n$ , 并将结果返回客户端; 在客户端更新 Ajax 控件 E 的状态值为  $n$ , 并用 userData 机制存储状态值  $n$ :  $E.\text{setAttribute}("E", n); E.\text{save}("E")$ , 如图 3 所示。

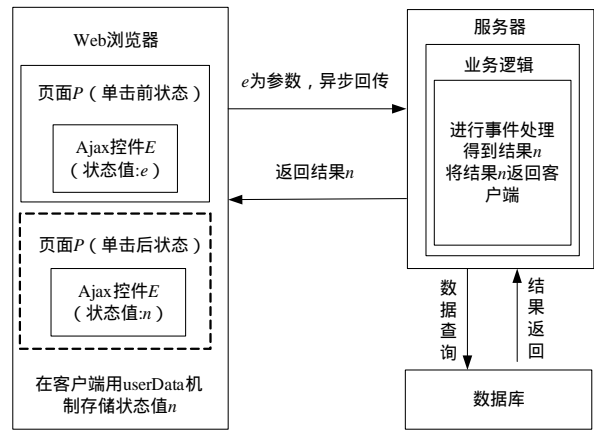


图 3 使用 userData 对象保值过程

当单击任一服务器标准控件引起整个页面 P 回传后, ViewState 机制将恢复页面 P 上服务器控件 A, B, C 的状态值, 用 Session 机制来恢复 Ajax 控件 D 的状态值, 用 userData 机制来恢复 Ajax 控件 E 的状态值, 如图 4 所示。

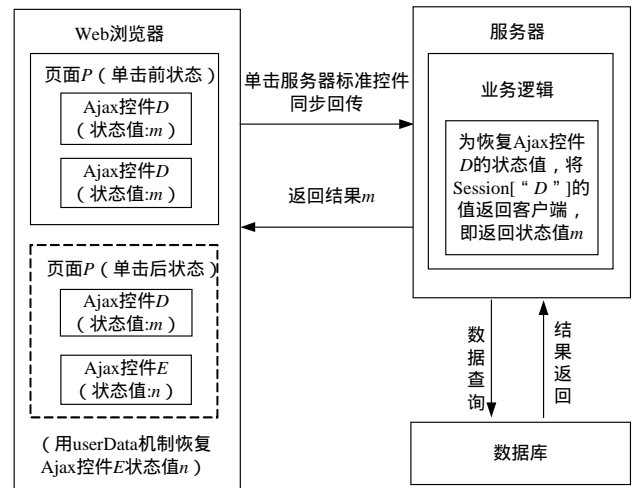


图 4 Ajax 控件值的恢复

通过上述过程, Ajax 控件 D, E 变化后取得的数据  $m, n$  即得以保存。

## 3 Ajax 控件保值应用实例

实例通过在 ASP.NET 中使用 Ajax 控件来实现“省、市、县”3 个下拉框的三级联动。用服务器的 Session 机制来存储市下拉框的状态值, 用客户端的 userData 机制来存储县下拉框的状态值, 从而建立一套新的状态保存机制, 使刷新页面后仍能保持下拉框控件的先前状态。其效果为: 选择省时, 市、县下拉框也跟着对应的变化; 选择市时, 县下拉框跟着变化, 整个过程页面不刷新。

点击刷新按钮进行页面刷新后, 下拉框控件保持最后一次客户端设定时的状态。实例的 3 个下拉框控件数据值状态分别采用不同保存方式: 省下拉框固定在客户端, 无需联动, 使用原 ViewState 机制; 市下拉框是联动主体, 使用服务器端 Session 机制; 县下拉框是最后一级联动列, 使用客户端 userData 机制。

### 3.1 初始设置

对平台环境(Microsoft Visual Studio 2005)进行初始设置<sup>[3-4]</sup>以应用 Ajax.net (选取 SQL Server 2000 作为后台数据管理), 如图 5 所示。

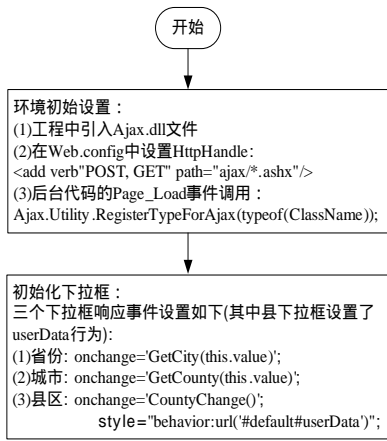


图5 初始设置

### 3.2 下拉框响应事件

当改变下拉框选项时，将会触发下拉框的客户端响应事件，过程描述如图6所示。

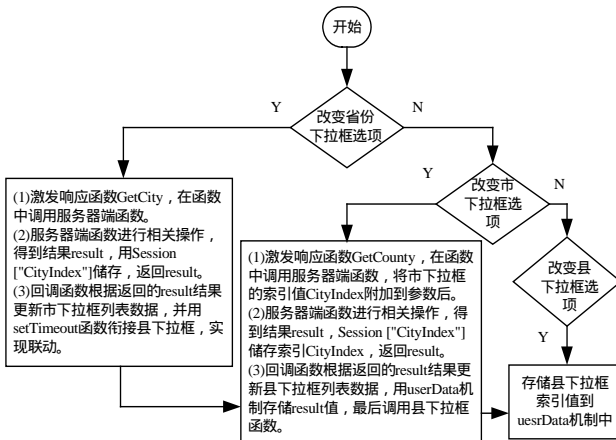


图6 省、市、县三级联动过程

(上接第37页)

的关键技术在某合作科研项目中进行了理论研究，在某飞机机翼项目开发中进行了验证。系统将航空项目管理系统和SmarTeam各自的Web服务进行封装和发布，通过建立BOM和WBS映射关系，实现2个系统数据和功能的集成。集成功能体现在以下4个关键过程：

(1)基于BOM进行WBS分解：SmarTeam系统通过Web服务发布产品BOM信息，PM系统请求此服务导入产品BOM信息，并进行基于BOM的WBS分解，实现BOM节点和WBS节点的映射和关联，以此映射关系和编码作为请求信息的统一入口。

(2)WBS节点任务信息更新：PM系统通过WBS节点编码调用Web服务查找相应的BOM节点信息，依据任务相应的BOM节点信息对项目WBS信息进行更新。

(3)BOM节点信息查看：工程人员在SmarTeam系统中通过BOM节点编码调用相应的Web服务浏览BOM节点对应的计划、进度等管理信息。

(4)项目文件PDM管理：PM系统将其所生成的项目文档在PDM系统中按照任务所映射的BOM编码对文件进行注册、检入，相关人员在PM系统和PDM系统浏览器端通过Web服务访问SmarTeam系统的文档接口，系统通过BOM编

### 3.3 页面刷新的控件保值处理

点击刷新按钮将引起页面刷新，其具体保值处理过程为以下2步：

(1)在服务器端将城市下拉框数据 Session["City"]和索引值 Session["CityIndex"]传送到客户端，以恢复市下拉框的状态值。

(2)在客户端用 userData 机制获取县下拉框数据以及索引值，以恢复县区下拉框的状态值。

### 4 结束语

本文建立一套针对 Ajax 控件的状态保存机制，从而避免 Ajax 控件值的丢失，使 Ajax 技术的应用更灵活，提高技术的扩展性与易控性。

Ajax 控件保值技术中的 ViewState 对象和 userData 对象都将控件状态值保存在客户端，而利用 Session 对象则保存在服务器端。ViewState 机制虽可通过覆写其实现方法保存在 Session 中，减少客户端和服务器端之间的交互数据流量，但在回调过程中却不能更新控件的状态，而且增大了服务器的负担。userData 机制是将需要存储的数据通过 XML 文档记录在系统盘当前用户的 userData 目录中，但目前只有 IE 支持，Fire Fox 并不支持。

### 参考文献

- [1] 游丽贞, 李纯喜. Ajax引擎的原理和应用[M]. 北京: 希望电子出版社, 2006.
- [2] 陈锦, 张建军. Ajax技术中Session服务的改进[J]. 计算机技术与发展, 2006, 6(12): 76-78.
- [3] 杨华. AJAX及在ASP.NET中的实现[J]. 西南科技大学学报, 2006, 29(12): 79-82.
- [4] Lele V. AJAX-style Web Development Using ASP.NET[Z]. (2006-12-20). <http://dotnet.sys-con.com/read/121828.htm>.

码对存储在系统中的项目文档和相关 PDM 文档进行浏览、编辑，实现项目文件在工程人员间的共享。

### 4 结束语

将项目数据与产品数据紧密关联，实现产品信息和项目信息的有效交互，有利于保证新产品开发和管理的有序进行，提高产品开发效率。本文研究基于 Web 服务技术实现项目管理系统与 PDM 系统的集成，主要从产品数据和项目数据自动关联的角度分析两系统集成策略，建立以产品 BOM 为中心的 WBS 分解、项目文件集成管理，实现产品信息和项目信息的紧密关联，有效保证工程信息和管理信息的交互，解决了管理信息与工程信息脱节的问题，提高了产品开发的效率，并以实例验证了集成的有效性。

### 参考文献

- [1] 王玉茂, 薛善良, 陈蔚芳. 项目管理与PDM集成技术研究[J]. 机械与电子, 2003, 5(3): 6-8.
- [2] 张守忠, 李原, 韩毅. 基于面向对象技术的航空项目管理与PDM集成技术研究[J]. 计算机集成制造系统, 2005, 11(7): 932-935.
- [3] 张恒文, 陈先鹏, 蔡颖. 基于PDM的项目管理研究[J]. 机械设计与制造, 2005, 2(9): 122-124.