

电子科技大学

UNIVERSITY OF ELECTRONIC SCIENCE AND TECHNOLOGY OF CHINA

工程硕士学位论文

ENGINEERING MASTER DISSERTATION



论文题目: 基于 ASP.NET AJAX 技术的国家精品课程网站设计与实现

工程领域: 软件工程

指导教师: 桑楠 教授

作者姓名: 邓超

班学号: 200792301004

分类号 _____ 密级 _____
UDC 注1 _____

学 位 论 文

基于 ASP.NET AJAX 技术的国家精品课程网站设计与实现

(题名和副题名)

邓 超

(作者姓名)

指导教师姓名 桑 楠 教 授

电子科技大学 成 都

贺建飏 高 工

湖南金码科技发展有限公司 长 沙

(职务、职称、学位、单位名称及地址)

申请专业学位级别 硕士 专业学位类别 工 程 硕 士

工程领域名称 软 件 工 程

提交论文日期 2010.11 论文答辩日期 2010.12

学位授予单位和日期 电 子 科 技 大 学

答辩委员会主席 _____

评阅人 _____

年 月 日

独 创 性 声 明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。据我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得电子科技大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

签名：

日期： 年 月 日

关于论文使用授权的说明

本学位论文作者完全了解电子科技大学有关保留、使用学位论文的规定，有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许论文被查阅和借阅。本人授权电子科技大学可以将学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

（保密的学位论文在解密后应遵守此规定）

签名：

导师签名：

日期： 年 月 日

摘要

摘 要

网页切换的速率和网页的互动性是 WEB 应用程序开发中重要的性能指标。Ajax (Asynchronous JavaScript And XML) 技术是一种使用 JavaScript 来进行异步通信(Asynchronous) 以交换 XML 数据的 WEB 应用程序开发技术,它改变了传统的网页需通过刷新整个页面来获得新页面的工作方式,代之以异步通信技术完成页面的局部刷新,从而可以获得诸如页面快速切换、减少网络流量、降低服务器负荷等一系列的 WEB 应用程序性能提升,并能够为用户提供更高体验度的互动和个性化服务。

国家精品课程网站是精品课程建设的重要一环节,网站建设的优劣直接影响到访问量和影响度。目前的精品课程网站普通存连通性差、交互功能弱、用户不够友好等诸多技术性的问题。如果应用 Web 2.0 时代的新技术 Ajax 提升精品课程网站的交互度和用户友好度,势必将有效地解决上述问题。

Ajax 技术有诸多框架,其中不乏许多应用广泛的框架,基于精品课程网站是在 ASP.net 平台上完成,故本论文选用了 Microsoft 公司的 AJAX 框架实现课程网站中诸多功能改造,主要研究如何利用 Ajax 技术提高 WEB 应用的交互度,从而提高用户体验度。理论方面,阐述了 Ajax 技术原理和功能架构,对比了 Ajax 开发模式与传统网页开发模式,为使用 Ajax 技术进行 WEB 应用开发的最佳实践提供了理论依据。实践方面,对《湖南科技职院软件学院精品课程网站》进行需求分析和设计,同时也对原有基于传统 WEB 应用开发模式的精品课程网站中的不足进行了分析和描述,通过应用 Ajax 技术改进了原有系统的性能和用户交互度,使得 Ajax 技术在教育教学系统中得以发挥它特有的优势,这也是本论文的重点。

关键词: Ajax, Web 应用开发, ASP.NET 3.5, 局部刷新, 精品课程网站

ABSTRACT

ABSTRACT

Web switching speed and interactivity of web pages is a WEB application development key performance indicators. Ajax (Asynchronous JavaScript And XML) technology is the use of JavaScript to conduct a kind of asynchronous communication (Asynchronous) in exchange for XML data in WEB application development techniques. It has changed the traditional web page need to refresh the entire page to get the new page ways of working and replace them with the partial refresh the page through the asynchronous communication technologies, to get such as fast switching, reducing network traffic, reduce server load and a series of The WEB application performance, and can provide users with a higher degree of interaction and experience, personalized service.

Advanced Courses Course Construction site is an important aspect of website construction will directly affect the degree of traffic and influence. Ordinary Course Website currently exist connectivity is poor, the weak interactive features, user-friendly enough, and many other technical issues. If the application of new technology era of Web 2.0 Course Website Ajax interactions to enhance reliability and user friendliness, is bound to effectively address the issue.

This thesis is Microsoft's AJAX framework for ASP.NET 3.5 to achieve the Ajax technology, primarily looking at how to use Ajax technology to enhance the interactive WEB applications degrees, thereby enhancing the user experience degrees. Theory expounded the principles and functions architecture of Ajax technology, Ajax development model compared with traditional web development model for the use of Ajax technology for WEB application development best practices to provide theoretical basis. Practice of "Hunan Science and Technology Vocational-Technical School of Software Quality Course Web site (C #)" requirements analysis and design, Also analyze and descript on the WEB application development based on the traditional mode of Quality Course site in less than Through the application of Ajax technology to improve the performance of the original system and the degree of user interaction, making Ajax technology in education and teaching system to play its unique advantages, which is the focus of this thesis.

Keywords: Ajax , Web Application Development, ASP.NET 3.5, Partial refresh, Quality Course Website

目 录

第一章	绪 论.....	1
1.1	课题的研究目的及意义.....	2
1.2	国内外研究现状.....	3
1.3	本文研究目标及研究内容.....	5
1.4	本文章节组织.....	5
第二章	Ajax 技术概述与分析	7
2.1	Ajax 简介	7
2.2	Ajax 核心及开发步骤.....	7
2.3	Ajax 技术与传统 WEB 开发技术的对比	8
2.4	Ajax 框架介绍	12
2.5	Ajax 经典案例分析	12
2.6	Ajax 的安全与应用	15
2.7	ASP.NET Ajax 3.5 介绍.....	18
2.8	使用 ASP.NET Ajax 3.5.....	20
2.9	本章小结.....	21
第三章	基于 Ajax 的精品课程网站的概要设计	22
3.1	项目简介	22
3.2	项目功能分析.....	22
3.3	Ajax 技术应用于精品课程网站的设计	25
3.4	本章小结.....	30
第四章	精品课程网站的设计与实现.....	31
4.1	课程网站核心功能模块设计与实现.....	31
4.2	C#国家精品课程网站 Ajax 应用功能示例	47
4.3	系统运行及测试结果.....	53
4.4	本章小结.....	61
第五章	结论与展望.....	63
5.1	论文工作总结.....	63

目录

5.2 论文主要特色.....	64
5.3 研究工作展望.....	64
致 谢	66
参考文献	67

第一章 绪 论

教育部于 2003 年启动的国家精品课程建设是国家教育“质量工程”的重要组成部分，其目的在于推进教育创新，深化教学改革，促进优质教育教学资源共享，全面提高教学质量。目前，全国各级各类院校都在积极建设自己的精品课程以及精品课程网站，实现了国家、省、校三级精品课程体系。截至 2010 年，全国共有 4000 门课程被选为国家精品课程，各级各类达到 17283 门^[1]，包括本科、高职高专、网络教育等各级各类院校的课程。国家教育部在评审精品课程时，采用网上申报、网上观摩教学录像、网上教学资源评审、网上学生评价和网上公示。所以，构建精品课程网站已成为精品课程建设整体工作中至关重要的环节。

精品课程网站是国家构建开放式教育服务平台的载体，要充分有效地实现信息共享、资源共享、师生互动，就必需把握 Web 开发技术动态，使网站形式上的变革紧紧跟上 Web 技术的发展。

回顾万维网（World Wide Web，WWW）的历史，自 1989 年 CERN(欧洲粒子物理实验室)的研究人员开发的远程访问系统开始也不过短短的二十几年时间，万维网已从最初单纯地共享静态文本资源，发展到能共享包括文字、图像、音频、视频等信息的更为开放、更具有交互性的资源共享、信息交流的平台。而 2005 年 Ajax 概念的提出，无疑是万维网在其不长的发展过程中的又一里程碑事件。

当 Ajax（Asynchronous JavaScript And XML）被工作在 Web 设计咨询公司的用户体验总监 Jesse James Garrett 在他的一篇名为《Ajax：一种 Web 应用程序开发的新方法》的文章中提出之后，引发了人们争相追捧的热潮，Garrett 在该文章中讨论了如何消除胖客户（或桌面）应用与瘦客户（或 Web）应用之间的界限。

从 Garrett 提出 Ajax 这个名词之后的几年，Ajax 掀起了 Web 应用的又一轮革命，特别是在近两三年，“Ajax 化”即对传统网页的 Ajax 改造已经成为各大网站的首要技术目标。实际上，Ajax 所涉技术包括 HTML、XHTML、CSS、DOM、XML、XSLT 和 JavaScript 等早已广为人知且应用广泛的 Web 开发技术，Ajax 的创造性在于它将这些技术有机地结合，基于 Ajax 的 Web 应用打破了传统 Web 应用的 click—and—wait 交互模式，使用起来将更接近桌面应用^[2]，从而使 Web 用户获得最佳体验。

Google 是业界领先的 Ajax 早期使用者,其 Ajax 应用包括 Google Maps、Google Suggest、Gmail 和 iGoogle 等,正是这些应用将鲜为人知的 Ajax 推到了人潮攒动的台前。Google Maps 的快速、Google Suggest 的友好、iGoogle 的“随心所欲”都让用户对采用 Ajax 技术开发的网站充满了喜爱,短短几年时间,国内外众多网站也因此都开始了“Ajax 化”的步伐。无疑,如果能够在精品课程类网站中引入 Ajax 特性,将让精品课程类网站体现开放性的同时,展现出更好的服务性。

本章将简要介绍 Ajax 技术的基本特点,随后讨论 Ajax 的应用现状及发展趋势,并从 Web 开发技术角度讨论国家精品课程网站的发展状况以及存在的问题。最后介绍本论文的总体结构和拟开展的工作。

1.1 课题的研究目的及意义

众所周知,桌面应用程序具有丰富的界面和快速响应的优势,但缺乏网络应用程序的数据共享的优点,而 Web 应用程序的优点恰恰在于轻松实现数据共享,而且部署和维护过程非常简单,但缺乏桌面应用所具有的丰富的界面和实时响应能力。于是,人们不得不在两种应用之间做出选择,即只能获得一种应用的优势;但是,随着 Ajax 的出现,以上情况正在发生着变化^[3]。

以 Google Map、Google Suggest、iGoogle 为代表的 Ajax 应用,正逐步淡化桌面应用与 Web 应用程序之间的界限,展现了一个令人充满惊喜和期待的 Web 应用开发的新时代。Ajax 应用使得浏览器页面可以只做必要的局部更新,服务器端可以更快速地处理请求,发送响应信息时将占用更少的带宽。并且客户端的脚本代码可以向服务器端应用程序发起异步 (Synchronized) 的或者非阻塞 (Non-blocking) 的请求,即在上一个请求还没有收到服务端响应之前,用户仍然可以在页面上发起新一轮的其他请求,给用户带来良好的体验。总之, Ajax 所具有的数据传输量小、响应速度快、高度交互性、无需下载安装、个性化定制等优点让他能够提供与桌面应用相媲美的界面和响应速度。国内外的各类型网站继 Google 之后均在研究“Ajax 化”的问题,精品课程网站的“Ajax 化”研究将拓宽 Ajax 的服务领域,同时也能够推动精品课程网站的设计革命。

到目前为止, Ajax 技术的发展步伐从未停止过,关于 Ajax 的技术框架、理论也是层出不穷。但 Ajax 应用到底该如何在不同的专业领域发挥它的优势,这还需要不断实践、总结。如何让他的优势恰到好处地在不同的网站中有选择性地体现,如在精品课程类网站中该如何发挥它的优势,使优质教育资源借 Ajax 的技术优势

充分发挥出能效，是一个非常具有实用意义甚至经济价值的课题，值得认真研究。本文通过对 Ajax 及 Asp.net Ajax 特性进行研究，围绕“Ajax 化”国家精品课程网站所需解决的问题，面临的的安全性和性能挑战，为 Ajax 技术在不同专业领域的实用价值研究提供了理论和实践依据。

1.2 国内外研究现状

1.2.1 AJAX 的应用现状及发展趋势

尽管 Ajax 技术在 1998 年前后在 Internet Explorer 4.0 中就得到了应用。但直到 2005 年初，许多事件的发生才使得 Ajax 被大众所接受。Google 在它著名的交互应用程序中使用了异步通讯，如 Google 讨论组、Google 地图、Google 搜索建议、Gmail 等。Google 公司是 Ajax 应用的早期领跑者，直到现在 Google 也是这项技术的领航者，它相继推出的 Ajax 应用项目，一次次地颠覆了人们对 Web 应用的固有认识。访问 iGoogle 之后我们便知道门户网站的内容也可以不用一层不变，而是可以为我们个人而定制，从小小的便条纸、翻译工具、日历，到新闻，股票行情，甚至是游戏都可以出现在一张网页上，而且他的出现与否，完全由用户自己来设置。除 Google 外，IBM、Oracle、Yahoo!、BEA、RedHat、Novell 这些高科技公司无不为了 Ajax 而动容，自 2006 年他们联合宣布“Open Ajax”计划以来，软件领域关于 Ajax 的框架、工具便日益丰富。有了这些软件巨头的引领，各路商业网站更是对 Ajax 表现出了极大的热情。Amazon、苹果、Flickr、电子商务网站 Netflix、美国在线和道琼斯网站等等都在他们的网站中大量地运用了 Ajax 技术。国内许多成功的商业网站也使用 Ajax 技术对网站部分功能进行了重构，如淘宝网的登录功能、商品细节图显示功能，网易邮箱的“极速风格”等。这些网站提供的功能无不简单的 Web 网页具有了类似于桌面客户端软件的灵活性，在加载动态化和操作客户端化两方面带来了革命性的突破。

Web 应用的两个终极目标便是可用性和语义网。Ajax 的出现无疑极大地改善了 Web 应用的可用性。可以这样讲，哪里不希望出现“抖动”的网页，哪里就需要 Ajax。未来的 Ajax 应用还将更加成熟，开发也将变得更加简易、标准化。现在 Ajax 面临的问题之一是如何在不同的专业网站中充分发挥出 Ajax 技术的能效，使之成为各行各业的网站服务，而不是仅限于软件巨头或是商业门户网站。

1.2.2 精品课程网站的国内外发展现状

自国家启动精品课程建设项目至今已过去了七年，该项目的成果是丰硕的，各级各类高校建设的 4000 多门精品课程网站提供了教案、视频教学录相等相当丰富的教学资源，可是也存在一些问题。以国务院发展研究中心信息网发布的一篇文章公布的资料显示，在国家精品课程开发过程中，建设者的资源建设标准化意识和版权意识比较淡薄；在应用和更新方面，大部分建设者会使用自己的精品课程，但是他们并不清楚自己的精品课程为他人所使用的情况；还有，大部分建设者没有更新过自己的精品课程^[4]。

笔者从网络技术的角度考察了各级各类国家精品课程网站，现有的精品课程网站主要有以下情况值得引起重视。

一是课程网站连通问题。2008 年 3 月，有学者对国家精品课程支持网站进行连通性检查，发现来自 324 所学校的 1110 门精品课程中居然有 563 家存在这样或那样的连通性问题，表现为无法找到网页、程序出错、用户没有得到授权、找不到服务器、页面无法显示等问题^[5]。有的学院在这方面做得还不错，但是却存在站内无法找到网页、课件无法播放等问题。

二是互动模块问题。目前各精品课程网站提供的互动功能非常有限，还只限于访问统计、网上论坛、在线自测功能。更严重的问题是国家精品课程网站有相当一部分存在连通问题，而在可以连通的课程中，只有 79 门精品课程有访问统计或在线调查等模块，占可以连通的国家精品课程总数的 14.4%^[5]。缺乏这些掌握应用情况的模块，精品课程开发者就无法了解自己的课程的应用情况，闭门造车，盲目开发。

三是网站设计技术过于陈旧或是并未充分发挥技术优势。一般都是直接购买内容管理系统然后进行一下“校内化”工作，或是通过校内教师在短时间内赶出一个几乎都是静态内容的网站。

国外也有类似精品课程网站的课程网，以美国麻省理工学院为例，他们叫做 Open Course Ware，简称 OCW。2003 年 OCW 就已经有 1100 门课程，该课程网对访问者做了详细的统计，每天哪一门课程有多少访问者，访问者分布于世界的哪个地方，均有详细数据。而我国的精品课程网站在统计访问数据和在线调查学习者情况方面还有待进一步加强。

随着精品课建设项目的推进，精品课程网站要在建设者与使用者之间产生共鸣，网站的互动性将是至关重要的一环节。WEB2.0 时代的到来，新技术和新工具不断涌现，也为网站互动性的增强提供了机遇。如使用 RSS 订阅功能的教学资源模块，如嵌入 Wiki 技术的师生交流学习系统。AJAX 技术前景非常乐观，可以提

高系统性能，优化用户界面。该技术在网站中的应用更是能够提高学习者对网站的体验度，从而增强网站的吸引力。精品课程网的 Ajax 应用还在起步阶段，只有少数几位学者提出了一些构想，可供借鉴的案例较少，但精品课程网站的“Ajax 化”将是未来精品课程网站发展的一个重要方向，这也是市场的自然选择。

1.3 本文研究目标及研究内容

1.3.1 研究目标

本文将通过对 Ajax 技术的深入探讨，分析和研究基于传统 Web 应用开发模式的网络应用所面临的问题。另外，通过理论与实践相结合，研究如何在 Web 应用系统中集成 ASP.NET Ajax 框架技术实现在传统开发模式下无法或很难完成的任务，解决目前精品课程网站中普遍存在的互动性差，可维护性差，连通性差等问题，从而弥补传统 Web 应用程序的不足，为其它准备使用 ASP.NET Ajax 框架的 Web 应用提供很好的范例和借鉴。

1.3.2 研究内容

本文的研究内容主要有下面几项：

1. 研究比较传统 Web 应用开发模式与基于 Ajax 的 Web 应用开发模式。
2. 研究 ASP.net Ajax 框架在实际项目中的应用。
3. 研究如何基于 ASP.net Ajax 框架将 Ajax 技术应用到树型目录，以提高树型目录使用时的用户体验度和应用程序运行速度和效率。
4. 研究 Ajax 技术下的 Web 应用设计模式。

1.4 本文章节组织

全文共分五章，各章节具体研究内容如下：

第一章：绪论。主要阐述课题背景、课题的研究目的及意义、国内外研究发展的现状及趋势，以及本论文的研究目标和内容安排。

第二章：Ajax 技术概述与分析。主要阐述 Ajax 的定义及工作原理，并示例典型的开发步骤。还涉及了 Ajax 的开发框架、应用范畴，分析了 Ajax 技术优劣势，并对经典案例进行了列举。

第三章：基于 Ajax 的精品课程网站设计。本章和下一章是本论文的重点，主要阐述了如何使用 Ajax 技术对原有的精品课程网站进行设计改造。以达到增强互动，提高用户体验度的目的。通过对比，充分展现了 Ajax 的应用魅力。

第四章：基于 Asp.Net Ajax 的精品课程网站实现。本章与上一章是本论文的重点，主要阐述了如何基于微软的 Atlas 实现精品课程网站的改造。这一章首先介绍了微软的 Atlas 的特点，以及使用环境的设置情况，然后描述了如何使用该环境完成上一章设计的 Ajax 功能，图文并茂并附有源代码。为精品课程网站的“Ajax 化”提供了可供借鉴的经验。

第五章：总结与展望。总结了全文的研究工作，并对后续工作提出了展望。

第二章 Ajax 技术概述与分析

2.1 Ajax 简介

Web 应用的原始工作形态在今天看来是有些笨拙的，当用户在一个具有交互功能的动态 Web 页面上向服务器发出请求时，浏览器会把整个页面发送给服务器，服务器端应用程序接收到请求后，会根据请求重新创建该页面的 HTML 代码，然后返回给浏览器。细心的人一定会发现，当新旧两个页面内容大不相同甚至完全不同时，这样的工作方式无可厚非，但当新旧页面只有小部分内容差异时，完整回发新页面是相当浪费资源且多此一举的。更重要的是，在当前这一轮请求还未完成时用户就想要开始下一轮的请求是不可能的，在请求没有得到响应前，用户必须等待，这是网络用户最难以忍受的。哪怕是网速快到让用户都感觉不到，新页面重新加载到浏览器总会闪烁一下。Ajax 技术的应用，结束了传统 Web 应用强制用户进入先提交、接着等待、然后重新刷新的交互循环的局面。

Ajax 是什么？可以说它是一整套的技术，也可以说它是一种架构，所以很难用一句话来概括它。但 Ajax 这个名字来源于它所涉及的技术：服务器和浏览器，JavaScript 和 XML 之间的一个异步通信通道(Asynchronous JavaScript and XML)。它使用 XHTML 和 CSS 完成基于标准的界面表示，使用浏览器的文档对象模型（DOM）完成动态显示和交互，使用 XML 和 XSLT 完成数据交换和操作；使用 XMLHttpRequest 或 XMLHttpRequest（微软）完成异步数据检索；通过 JavaScript 将各种技术有机地结合在一起^[5]。

2.2 Ajax 核心及开发步骤

JavaScript 对象 XMLHttpRequest 是 Ajax 技术的核心，通过这个对象向服务器提出请求，不会阻塞用户的操作，即可以实现在客户端和服务器之间实现异步交互。异步交互意味着在 XMLHttpRequest 向服务器发送请求之后，浏览器不用等待服务器响应才可以开始其它操作，而是可以继续处理页面中其它的用户操作。这个过程中，数据在后台传递完毕后会自动装入到页面中，不会产生页面刷新。服务器与客户端的通信过程如图 2-1 所示。

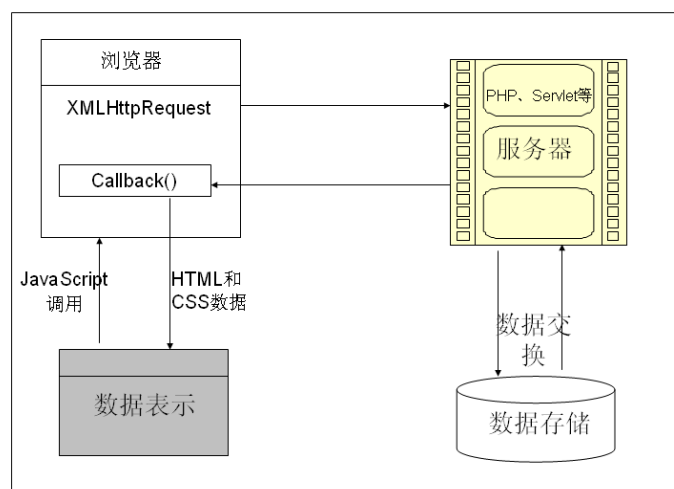


图 2-1:服务器与客户端通信过程图

下面给出 XMLHttpRequest 对象的常用方法和属性：

- open()方法：建立到服务器的请求
- send()方法：向服务器发送请求
- abort()方法：退出当前请求
- readyState 属性：提供当前 HTML 的就绪状态
- responseText 属性：服务器返回的请求响应文本

2.3 Ajax 技术与传统 WEB 开发技术的对比

2.3.2 工作原理

Ajax 技术工作模型与传统 Web 应用的工作模型对比情况如图 2-2 所示。通过左右两张图可以发现应用 Ajax 技术的 Web 应用在浏览器与服务器之间加了一个中间层，这一层就是所谓的 Ajax 引擎。有了这一层之后，浏览器端的请求在到达服务器之前得先通过 Ajax 引擎，诸如数据验证和数据处理之类的工作就可以不用提交到服务器，而是由 Ajax 引擎来完成，只有确实需要提取服务器数据的操作才会由引擎转交到服务器。

第二章 Ajax 技术概述与分析

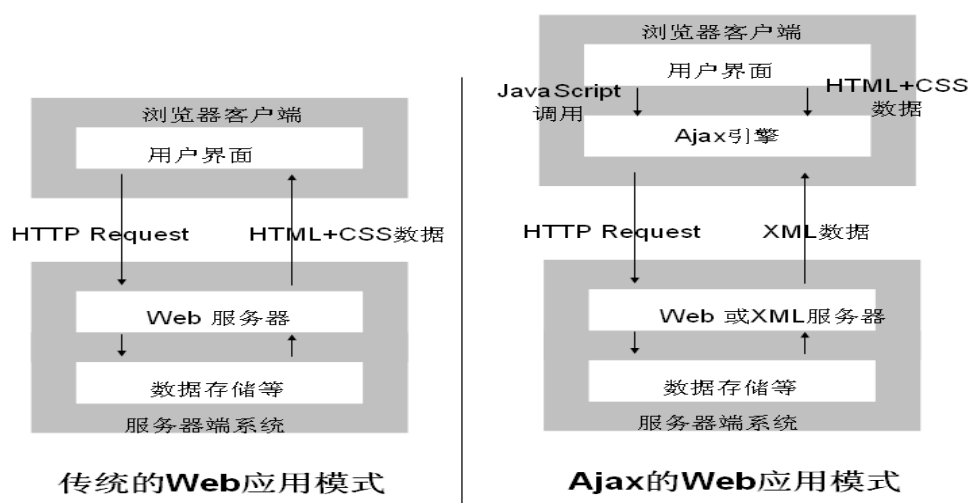


图 2-2 传统 Web 应用模型(左)与 Ajax 模型的比较(右)

有了 Ajax 中间层，用户操作与服务器响应就可以异步化，当服务器还在处理客户端请求时，客户端不需要等待（直到收到服务器回应以后才停止等待），就可以开始其它操作^[10]。用户的感觉自然是服务器的反应更加灵敏迅速。页面上最大的变化就是以往将会产生 HTTP 请求的用户动作将由用 JavaScript 编写的 Ajax 引擎来代替。引擎的职责主要是绘制用户界面和与服务器通讯。关于传统 Web 应用的同步工作方式与 Ajax 应用的异步工作方式比较如图 2-3 所示。

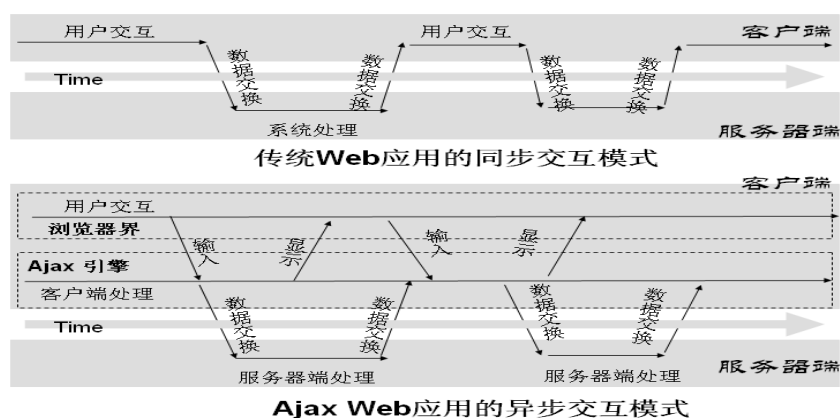


图 2-3 传统 Web 应用的同步交互过程（上）与 Ajax 应用的异步交互过程（下）

2.3.3 Ajax 技术优势

在最传统的 web 应用中，一般是这样的一个浏览器服务器会话过程：首先用户提交填写好的表单(form)，即一个客户端请求被发往 web 服务器。然后传来的表单被服务器接收处理，并向客户端浏览器返回一个新的网页，即服务器做出响应。这种做法显然是一种浪费带宽资源的做法，因为往往请求前与请求后两个页面中的 HTML 代码是大同小异的。这样一种交互的方式，用户得到响应的时间由服务器的处理速度所控制，而且每一轮用户交互都必须向服务器直接发送请求，用户就不得不看着时间漏斗守着一个正在刷新的空白页面等待。哪怕是不等待，用户也会看到一次页面的闪烁。与传统 Web 应用不同的是，AJAX 应用可以仅向服务器发送并取回必需的数据，它使用 SOAP 或其它一些基于 XML 的 Web Service 接口，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，结果我们就能看到响应更快的应用。Ajax 应用的优势主要体现在以下几方面：

1. “按需请求数据，按需获取数据”是 Ajax 技术最显著的特征。这样做就可以最大程度的减少冗余请求从而减轻服务器的响应负担。
2. 无闪烁刷新页面，使得用户的实际等待时间与心理等待时间都有减少。数据量越大的情况下，这个优点就越是突出，传统 Web 应用中经常遇到的白屏现象也不复存在。通过 XMLHttpRequest 对象发送请求；通过 JavaScript 代码操作 DOM 对象得到新页面，这一切仿佛就在转瞬之间，因为有了白屏，用户甚至都感觉不到与服务器有交互，页面就已经得到了更新，与使用桌面应用程序的感受一样。
3. 客户端闲置的运算能力也可以运用起来，利用 Ajax 引擎将原本由服务器承担的工作转移到客户端机器，从而减少了服务器运算负担和网络带宽负担。
4. Ajax 基于标准化的并被广泛支持的技术，故不需要任何浏览器插件或者小程序就可以运行。
5. 通过使用 CSS 等技术，可以有效地将 Web 中的界面与应用（数据）分离，便于 Web 应用的高效设计和维护升级。

总之，使用 Ajax 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 Web 应用程序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。

2.3.4 Ajax 技术不足与解决方案

任何事务都有其两面性, Ajax 技术也不能例外。像 DHTML 应用程序那样 Ajax 程序需要用户浏览器允许 JavaScript 的执行; Ajax 应用程序应用之前必须在众多不同的浏览器和平台上进行严格测试……诸如此类的问题都是 Ajax 技术的软肋。总结一下, 主要有如下几方面的问题。

1. 来自批评者阵营最强烈的声音是, Ajax 它可能破坏浏览器后退按钮的正常动作。使用 Ajax 技术的页面被更新后, 如果用户点击浏览器后退按钮, 想要回到上一个页面的状态, 这一切变得有些复杂了。因为浏览器后退按钮功能仅仅针对历史记录中的非动态页面。可是作为普通的浏览器用户肯定希望得到动态页面的快速灵活的同时还能保有在静态页面上能够获得的诸多浏览器便利功能。类似于后退取消操作, 或是收藏夹功能, 用户是难以割舍的, 但是在 Ajax 应用程序中, 却无法这样做。好在程序员们解决这个问题的办法还挺多, 例如, 通过建立或使用一个隐藏的 IFRAME 在用户点击后退按钮时重现页面上的变更; 通过使用 URL 片断标识符保持跟踪。
2. Ajax 技术的无刷新延迟就像一把双刃剑, 可能让用户无所适从。因为无论动态页面的更新有多快, 在用户请求与服务器响应之间总会产生延迟, 哪怕是片刻的延迟, 如果页面上没有明确的提示, 用户有可能会误认为浏览器没有做出回应; 或者页面更新不明显被用户给忽略掉, 用户都有可能不断地提交请求或是关闭页面。因此, 我们通常会使用一个可视化的组件通知用户系统已经开始在更新页面, 这样用户就知道浏览器已经做出了回应, 同时页面上其它的操作也没有耽误, Ajax 技术在用户那感受到的好处才变得显而易见, 否则, 当后台正在处理或是读取数据时, 用户因为误会而关闭页面就可惜了。
3. 随着 3G 时代的到来, 更多的手持设备(如手机、PDA 等)应用开始水涨船高, 但这些设备上的应用对 Ajax 的支持还非常有限。这是 Ajax 技术未来的发展方向。
4. Ajax 引擎实际上就是用 JavaScript 代码编写而成, 但 JavaScript 的兼容性和漏洞都是无法回避的问题, 这也是 Ajax 技术未来要解决的问题。
5. 对串流媒体的支持没有 FLASH、Java Applet 好。

总之，到目前为止 Ajax 的发展仍处于初级阶段，还存在一些争议和问题。只有正视这些问题，才能在开发过程中去克服。另外，Ajax 技术的使用要适可而止，不可泛滥使用。使用得当则妙笔生辉，反之则过犹不及^[7-9]。

2.4 Ajax 框架介绍

软件系统发展到今天已经足够复杂了，人们为了提高开发效率、增强系统的稳健性，想出了许多重用代码，重用设计的办法。其中框架技术无疑是代码重用与设计重用并举的成功代表作。正如许多的软件技术一样，Ajax 技术开发者、社区和软件公司为了提高 Ajax 应用的开发效率、降低程序开发的风险等诸多好处，也开发了许多针对 Ajax 应用的框架。这些框架有的基于客户端，有些基于服务器端；有些专门为特定语言设计，有些则与语言无关。其中绝大多数都有开源实现，但也有少数是专用的。如何在众多的框架中选择也成为 Ajax 应用开发者们需要慎重考虑的问题。流行的 Ajax 框架主要分为基于浏览器端的 Ajax 框架和基于服务器端的 Ajax 框架。如，Dojo、Open Rico、jQuery 属于浏览器端的 Ajax 框架，而 sajax、JSON/JSON-RPC、DWR、ASP.net Ajax 则属于服务器端的框架。

当然，Ajax 框架众多，没有哪种框架是适合所有的应用场合的，所以，一般开发者在选择框架时，会考虑许多方面的因素，如对浏览器的支持情况（最重要的浏览器/版本，以 Web 统计为准）、框架的文档/教程/帮助的质量、学习曲线、性能、伸缩性、灵活性/可扩展性、与现有站点集成的能力等等。例如本论文设计与实现的国家精品课程网站是基于.NET 开发的，微软免费提供的 Ajax .Net 框架是首选。

2.5 Ajax 经典案例分析

Ajax 应用在 Google 公司的引领下逐渐被人们所熟悉、了解和使用。它的应用范围正持续扩大，特别是在消费型网站的应用已是炉火纯青，甚至是在企业级 Web 应用中也开始崭露头角。下面列举一些最新的经典的国内外 Ajax 应用案例。

2.5.1 iGoogle(个性化 Google)

Google 提供的 iGoogle 服务可以让使用者根据个人的喜好方便地定制和整合不同来源的信息，使 Google 首页成为每个用户拥有的个性化的、特定的门户网站。目前 Google 网站上针对该服务提供的小工具日益丰富，且布局和风格都可以依据

第二章 Ajax 技术概述与分析

客户喜好自由定制。这些能随意增删的定制小工具包括嵌入的 Gmail 邮件系统、各大网站的新闻系统、博客系统、Google 游戏甚至是便条纸在内的涵盖资讯、工具、通讯、娱乐与游戏、财经、运动、生活方式、科技八类内容。更有意思的是这些小工具里的内容可以由用户来提供，即每个用户不但可以成为小工具的使用者还可以成为小工具供稿者。小工具的供稿者只需键入网址，设定一下内容的类型和风格即可。iGoogle 提供的服务使我们有机会设计 Web 页面就好像是在设计自己的 Windows 桌面一样方便，简单地拖曳、鼠标单击便有了自己中意的风格、甚至是内容，这正是人们期盼中的网络生活。

通过网址 <http://www.google.com.hk> 可以访问到该网站



2.5.2 淘宝网：商品大图展示功能

Ajax 技术在消费型网站的应用已有几年的历史，它的应用面正越来越广，最典型的应用场景就是商品展示介绍功能。淘宝网在这两年内也展开了许多的 Ajax 技术升级工作，如在购买页面上的用户登录功能、如商品搜索提示功能，就笔者而言最喜欢的要数这个商品大图提示功能。以往在应用 Ajax 技术之前，每当要查看某个商品的大图以了解商品细节时都需要跳转页面，费时且页面会跳转一下，但现在改造过之后的大图展示，只需将鼠标放到商品的某个部分，关于这个部分的大图就会在小图的右边显示出来，且随着鼠标的移动，细节还会适时地跟着变化。再也看不到跳转、刷新、空白和等待，感觉就犹如使用桌面应用程序。

通过访问 <http://www.taobao.com> 可以访问到该网站，需提醒的是每个淘宝商家购买的服务不一样，有的商家可能没有提供该功能。



2.5.3 企业级应用：美国国防部

与我们日常经常接触到的 Google 地图和 GMail 应用不同，大多数的企业级 Web 应用中的 Ajax 应用主要围绕数据录入、数据编辑或者数据报表构建。例如 CRM（客户关系系统）中的一个客户记录列表，列表中的记录能够添加、删除或者编辑。首先用户需要在数以千计的记录中自如地滚动，其速度应当如同是访问本地计算机中的数据一样快速，而不用通过每翻一页查看多少条记录，再有就是这个列表如果需要更新，应当可以每个字段或是需要的字段能够关联到数据库表中的一个字段，这也需要保证访问的速度足够快到用户感觉不到是在网络上工作，而象是在桌面应用系统上工作一样。Ajax 技术无疑让这一切成为了可能，在美国包括美国国防部在内的企业级 Web 应用网站都使用了这一技术来提高用户体验度。

通过 <http://www.defense.gov/> 可以访问该网站。

第二章 Ajax 技术概述与分析



2.6 Ajax 的安全与应用

2.6.1 Ajax 的安全问题及解决方案

随着 Ajax 技术应用的推广，无不避免地会遇到所有 Web 应用都会要面对的安全和性能问题以及 Ajax 技术特有的安全和性能问题。如何发现并采用有效手段避免或解决这些问题，是 Ajax 技术应用者们一直在研究的课题。下面我们列举一部分及其解决方案。

1. 浏览器兼容问题

Ajax 技术其本质是基于 JavaScript 的，是一项基于浏览器端的技术，作为浏览器端技术无可避免的问题即是兼容性。市面上流行的浏览器功能大同小异，但底层实现各不相同，自然他们对于 JavaScript、DOM、CSS 的支持多少总是会有差异，甚至有的还会有程序漏洞。有时，同一款浏览器的不同版本对于 JavaScript、DOM、CSS 的支持也不尽相同。这些问题就导致程序员要花相当多的时间调试 Ajax 应用的浏览器兼容性，分散了对应用程序本身关注的精力。

开发者针对这个问题的解决方案通常是，使用 js 链接库来开发底层的 Ajax 链接库或 Ajax 框架，在此基础上再定义更高阶的 JavaScript API、JavaScript 对象（模板）、或者 JavaScript Widgets（小工具）。

2. 业务逻辑分散，程序难于维护

Ajax 技术主要是通过客户端和服务端之间局部交换数据来达到节约带宽、加快交互速度的目的，如同普通的 C/S 架构程序一样，无法回避需要在客户端实现部分业务逻辑、或是需要在服务器端保有部分表示层逻辑，这样就会导致程序的层次结构不清晰，代码难以维护升级。

一般开发者的解决方案是在两类框架中进行抉择，一类是表示层业务层放在浏览器端的胖客户端架构，另一类则是部分表示层+业务层+数据层在服务器端的瘦客户端架构。具体在这两者之间选择哪一种，就得依据 Web 应用的需要而定了。

3. 多进程或多线程竞争问题

通过 Ajax 引擎可以实现与服务器的异步通信。但是，服务器在处理来自 Ajax 引擎的请求与处理传统 Web 应用的请求并无不同，反倒是因为异步提交，如果同时有多个 Ajax 请求及窗体提交发生，服务器不能保证先到达的需求先获得服务器响应，这是因为先到达的请求可能会因为等待资源而被 CPU 挂起，此时，如果多个请求都要访问同一个资源时，将造成多进程（process）或多线程（thread）的竞争（racing）问题。更关键的是 Ajax 技术没有提供机制保证处理这种问题，程序员必须自行处理，从而增加了程序员的负担。

开发者一般的解决方案是使用能够提供解决此类问题的 Ajax 框架，如：ZK。

4. 网络攻击

Web 应用最令人头疼的便是安全问题，采用 Ajax 技术的网站安全更是不容忽视。首先不良格式的数据是一大威胁；然后因为异步提交请求，黑客很容易发动拒绝服务攻击；再有就是恶意的用户可能会发送脏数据，或是进行未授权访问；另外，Ajax 技术中服务器通讯对于用户已经完全不可见，因此，黑客可以在用户不查觉的情况下传送数据，这也是安全隐患。诸如此类的问题还有很多，大多数造成的危害就是服务器资源耗尽，或者服务器宕机，再或者是资料被窃取。一般开发者最方便的解决方案就是在服务器端安装应用防火墙。

2.6.2 Ajax 技术的适用范围

Ajax 技术引入 Web 网站往往会产生令人激动的效果，但我们得意识到这个技术并不是万能的，合适的场合使用是锦上添花，不合适地方使用就是画蛇添足。Ajax 的特点在于异步交互，动态更新 web 页面，提高系统的性能，从而使用户获

得更好的体验。因此它的适用范围是交互较多，频繁读取数据的 Web 应用。现在通过几个实例来阐述 Ajax 的适用范围。

1. 异步数据处理

表单填写时，经常需要对数据进行验证（例如注册时的用户名唯一性验证），使用 Ajax 技术，不用提交整个页面，而是通过 XMLHttpRequest 对象发出验证数据的请求，从而可以避免白屏和弹出式窗口，减轻服务器负担的同时又提升了应用的性能，更重要的是在服务器端进行验证时并不影响用户在表单上继续填写数据。

类似于 Google Maps 的地图服务正是利用了这种异步数据处理的优势，当用户移动时，数据早已通过异步读取的方式存储在了内存，自然访问地图就变得流畅了许多。

2. 局部刷新数据

树形结构控件在 Web 应用系统中用来描述组织结构，目录等有层次的内容时非常清晰。但是当在树形结构的任何一个节点上单击都会引起页面重新加载刷新，为了避免反复加载引起的用户不愉快体验，开发者一般会一次性将树形结构所需的后台数据一次性读取出来并写入内存，之后再通过 JavaScript 来控制每个节点展现数据，这样做会产生大量的数据冗余，浪费内存和带宽资源。使用 Ajax 改进后的树形结构控件，每次只会获得用户点击的那一级数据，如，加载页面时，只获取第一级；当用户击第一级下的某个子级时，页面会通过 Ajax 引擎向服务器请求该特定子级下的直接子级，依此类推。页面按需索取数据，减轻了带宽压力也节省了内存空间。同时也大大缩短了用户的等待书形结构展现的时间。

类似于树形结构这样的需要局部刷新数据的应用，如数据列表的操作也可以考虑采用 Ajax 技术。

3. 利用无刷新能力自动更新页面

对于 Web 应用中许多适时数据应用，如股票信息系统，招生系统、天气预报等都可以使用 Ajax 来减少用户手动刷新页面的频率或者免去手动刷新。例如，当加载完页面，就通过 Ajax 引擎在后台定时向服务器发出请求，查看是否有数据变动。如果有则下载变动过的数据。这样做同时还减少了盲目重复刷新造成的资源浪费。

4. 桌面应用

如果需要将 Web 页面变成象 Windows 桌面一样允许用户定制、拖曳，可以使用这种超强能力的 Ajax 技术实现的工具来完成，有许多的框架提供了方便的组件来实现，只需少量的代码配合框架提供的控件即可完成如此酷炫的效果。

总之, Ajax 既不是万能的, 也不是十全十美的。适用时方用, 趋利避害地用, 方能达到优化 Web 应用的目的。否则为了技术而技术, 则有可能矫枉过正。

2.7 ASP.NET Ajax 3.5 介绍

2.7.1 Ajax 开发框架选择

前面我们提到 Ajax 的开发框架有很多种, 在众多框架中选择哪一个, 一般开发者会考虑很多的因素。本精品课程网站是基于 ASP.NET 平台开发的, 我们选择框架时首先考虑到了这个因素, 目前市面上, 支持 ASP.NET 开发的框架有一些, 如: WebORB for .NET 、ComfortASP.NET 、AjaxAspects 等。因为 ASP.NET Ajax 是微软官方推出的框架(严格说它不是框架), 不考虑其它因素的情况下, 我们认为它是最佳选择。

故我们最终选择集成了 ASP.NET 3.5 Ajax 的 Microsoft Visual Studio 2008 做为课程网站的升级开发工具。

2.7.2 ASP.NET Ajax 发展历程

ASP.NET AJAX 是微软于 2005 年夏启动的项目, 其目的是为 ASP.NET 应用程序提供 AJAX 基础支持, 最初的研发代号为 Atlas, 之后 2007 年 1 月份正式推出第一版并更名为 ASP.NET AJAX。起初只是一个可以和 ASP.NET 2.0 一起使用的单独组件, .NET Framework 3.5 之后已被集成到了 ASP .NET 平台中。

ASP.NET AJAX 所提供的基础架构使得原有的 ASP.NET 应用程序可以很轻松获得 Ajax 能力。早期 ASP.NET AJAX 使用 CTP (Community Technology Preview) 来代表它的测试版本, 面世不久便受到 ASP.NET 的开发人员的关注和呵护。2007 年, 基于 ASP.NET 2.0 的 ASP.NET AJAX 1.0 (版本号码为 1.0.61025) 发布了, 这个名称也延用至今。

今年四月, 微软与 .NET Framework 4.0 及 Visual Studio 2010 一起发布了 ASP.NET AJAX 4.0, 在此版本中提供的客户端模板可以轻松地实现将数据绑定到各种元素上, 全新的 DataView 控件使我们可以将模板扩展到整个数据列表, 使用标记扩展功能可以方便快捷地操作模板引擎的行为。

目前市场上应用较成熟的版本是 ASP.NET AJAX 3.5, 基于 NET Framework 3.5 可以运行, 且与开发工具 Visual Studio 2008 已经集成, 即只要用户安装了 Visual Studio 2008 (以下简称 VS 2008), 无需再安装其它附加程序就可以基于 NET

Framework 3.5 进行 ASP.NET Web 应用程序的 Ajax 开发。当在 VS 2008 中创建 ASP.NET 应用时，随应用自动生成的配置文件 web.config 里会包含 AJAX 注册设置，工具箱中也会出现核心 ASP.NET AJAX 服务器控件。ASP.NET AJAX 3.5 最重要的改进就是对 UpdatePanel 的支持，ASP.NET AJAX 3.5 在 ASP.NET AJAX 发展史里具有里程碑意义。

2.7.3 ASP.NET Ajax 组成和架构

Ajax 技术框架五花八门，数量众多，但基本上可以概括为三种类型，第一种是回调框架，一般仅由少数的几个客户端及服务器端库文件组成，只允许从客户端调用服务器端代码片段，并以序列化方式传送输入和输出参数。第二种是 UI 框架，典型的 UI 框架一般都是从现有的专业控件库演变而来，它们往往提供给开发者一些高级网格控件、图表控件和树控件，等等。这些控件都会支持异步回发机制并在客户端生成用于自动刷新页面的 JavaScript 代码。第三种是完整框架，完整框架则提供包含控件和应用程序服务在内的极为丰富的编程模型，适宜于同时应用于客户端和服务器的 Ajax 编程模型。

ASP.NET AJAX 属于第三种类型的框架，它支持两种架构的编程模型，两种不同但互补的 API，即客户端 API 和服务端 API。实际应用时可以选择其中之一，也可同时使用两者。下面分别对其进行介绍。

1. 客户端编程模型

ASP.NET AJAX 以客户端为中心的编程模型的重心是简化客户端编程，支持远程端点调用的能力，这些远程端点通常为 ASP.NET Web 服务和 Windows 通讯基础服务，但也可能是其它内容。

事实上，ASP.NET AJAX 的客户端部分是一组 JavaScript 库。这些库并非特定于 ASP.NET 才能使用，非 ASP.NET 应用也可以使用它们。客户端库并没有提供诸如直接拖放到 Web 页面的预定义功能，它们只是建立了用于开发 ASP.NET AJAX 页面的基础。这个基础包括弥补 JavaScript 的不足（如，增加了对命名空间、继承、接口、枚举和反射的支持），提供一些基本框架（例如，管理组件生命期的方法，对常见的数据类型进行操作，执行反射）。众多的 JavaScript (.js) 文件组成了 MS AJAX 客户端脚本库，极大地简化了程序开发。

2. 服务器端编程模型

其目标是扩展已有的 ASP.NET 服务器端模型，使其自动生成支持胖客户端的 JavaScript 代码。这其中就包括提供类似于 ASP.NET 服务器控件的 MS AJAX 服务器控件，这些 MS AJAX 服务器控件能够为开发人员生成客户端 JavaScript 脚本。这大大简化了 Ajax 应用中客户端 JavaScript 脚本的开发过程。MS AJAX 提供了一整套的服务器控件扩展支持，如针对 Label、Button、Option、TextBox、CheckBox 和 Validator 等均有扩展控件支持，这些控件与现有的 ASP.NET 服务器控件非常类似且都被集成到 Visual Studio 开发环境中，事实上，使用时如果不是特别提示，已经都感受不到它们与普通的服务器控件有可区别。ASP.NET AJAX 的服务端相对于客户端而言，工作层次相对高一些，如它将使用客户端脚本库的控件和组件。

客户端编程模型与服务端编程模型之间总是相辅相成，可以根据需要，调整以服务器为中心和以客户端为中心这两种开发模式的比例。

2.8 使用 ASP.NET Ajax 3.5

2.8.1 开发环境的搭建和配置

如果我们在 VS2008 下工作进行 ASP.NET Ajax 应用程序开发，基本不需要多余的环境搭建和配置工作，VS2008 会在我们使用 Ajax 功能时自动在配置文件中注册信息。但是前面提到的 MS Control Toolkit 控件工具包，因为它是开源项目的关系，控件的数量在不断增加，我们需要将这些控件手动配置到 VS 2008 的开发环境中。

首行，通过访问 <http://ajax.asp.net/ajaxtoolkit> 并按页面上的链接最终可以找到下载页面。在下载页面上，根据你使用的 .NET 版本以及是否需要源代码有几个不同的下载选项。下面以不带源代码的最简单的下载选项压缩文件（2MBZIP 文件）AjaxControlToolkit-Framework3.5-NoSource.zip 为例介绍其配置的方法。

- 1、下载并解压这个 ZIP 文件到硬盘。
- 2、解压后找到包含大量示例网站的 SampleWebSite 的目录。
SampleWebSite\Bin 子目录里是使用 ASP.NET AJAX 所必需的主要支持文件，包括一个名为 AjaxControlToolkit.dll 的核心程序集以及用于支持各种文化的本地化卫星程序集。

复制 SampleWebSite\Bin 目录里的全部内容到当前 Web 应用程序的 Bin 目录。或者通过将新组件（SampleWebSite\Bin 目录下有一个 AjaxControlToolkit.dll）加入到 Visual Studio 的工具箱都可以让我们开始使用 ASP.NET AJAX 控件工具包进行

开发。通过将 `AjaxControlToolkit.dll` 里的所有组件添加到工具箱里，将使我们的开发工作变得更简单。值得注意的是，使用这些控件之前，一定要记得给页面添加一个 `ScriptManager` 控件。

2.9 本章小结

本章全面地阐述了 Ajax 技术及其应用。内容涉及到什么是 Ajax、Ajax 应用的典型开发步骤、Ajax 的工作原理及其优势与不足、介绍了几种流行的客户端和服务端 Ajax 框架及其筛选原则，列举了最新的 Ajax 经典案例、分析了 Ajax 技术的安全性能问题并总结其应用范畴。本章内容为下一章应用 Ajax 技术对国家精品课程网站进行分析与设计奠定了扎实的理论基础。

第三章 基于 Ajax 的精品课程网站的概要设计

3.1 项目简介

目前，国家精品课程建设采用学校先行建设，省、自治区、直辖市择优推荐，教育部组织评审，授予荣誉称号，后补助建设经费的方式进行。精品课展示网站建设是精品课建设的重要环节，是学生、教师学习观摩的重要平台，也是对外交流、资源共享的重要窗口，反映了一个国家、一所高校的精品课建设水平。不仅是国内，在国外许多的高校都在加紧开展院校的精品课建设工作，精品课程展示网站是其中关键的一环节，有力地推动了开放式教育服务平台的建设。

《湖南科职软件学院 C#精品课程网站》围绕 C#这门课程教学资源展示来进行，主要供学生及游客上网查阅课程大纲、教案、按知识点模块浏览课件（知识模块又分为理论录相和实践录相），定制个性化网页内容，网上评论等。另还针对教师提供了网站内容后续维护方面的功能，如公告管理、课件管理、题库管理，作业评改、问题解答功能。该系统还有后台维护功能，主要完成资源审核，权限管理，网站参数设置等工作。

整个系统使用 Microsoft Visual Studio 2005 基于 ASP.NET 平台进行开发，经过一年半的实测，发现该系统正如许多国家精品课程网站一样存在交互功能欠缺或用户体验不佳等问题。现使用 Microsoft Visual Studio 2008 基于 ASP.NET 平台进行系统改造升级，主要是使用 ASP.NET AJAX 框架进行用户体验度改进，以期通过这种新旧 WEB 应用的对比，吸引更多的读者访问该网站。

3.2 项目功能分析

3.2.1 系统总体结构

经过反复的系统需求分析和确认，该网站的主要用户是教师和学生，根据该网站使用一年来的用户反馈，最终就系统改造达成一致意见，并确立系统需求如下：

系统应该支持四类用户使用，分别是游客（未登录用户），注册用户，登录的本校学生，登录的本校老师，系统管理员，各级用户应当具有不同的系统使用权限。

1. 未注册用户可以浏览 C#课程所有资源，如课件，教案，PPT 等。注册用户除可浏览外还可上传下载资源（但下载资源需要有积分方可下载），还能够就资源进行评价评分。
2. 登录的学生及老师用户可以浏览、下载课程资源信息外（下载无需积分），还能够对资源进行评价评分。同时还能够就某一节内容上传视频或 PPT 等其他资源。
3. 课程资源信息采用树型导航显示，内容层次根据 C#这门课程的章节来安排，每一节又分为视频、PPT、教案、习题、扩充资源五个方面。
4. 系统管理员应当可以对所有的课程资源进行管理，并且还能够进行用户管理和评论维护。
5. 所有用户都拥有信息查询功能，包括使用资源名称的简单关键字查询（支持模糊查询功能），还包括根据资源名称、类型等多个关键字的高级查询。
6. 所有用户的评分应该在用户点开该资源查看时用图形方式显示在资源页面下方，该页面同时提供给用户提交评价的界面，但无论评分还是评价都需要用户先登录系统。
7. 本校老师和学生无需注册，登录用户名和密码由系统管理员打印下发到各部门和班级。

根据需求分析，系统包含的模块体系如图 3-1 所示：

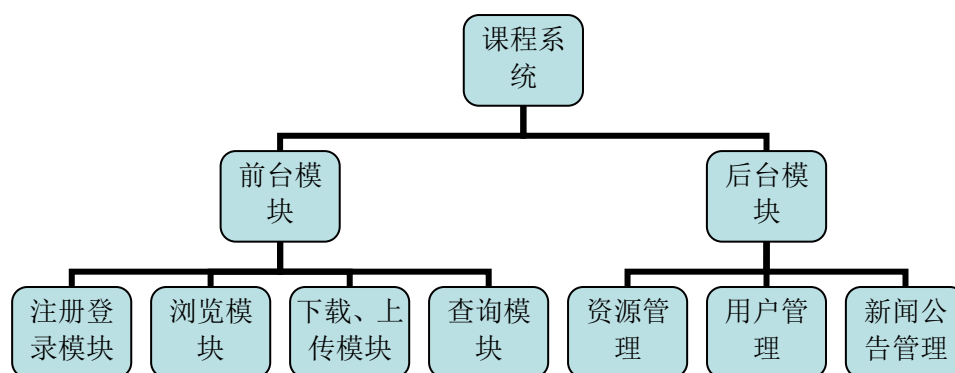


图 3-1:系统总结结构图

3.2.2 系统功能模块

为了简化问题描述，下面以框图的形式给出 C#精品课程网站需要实现的功能模块：

电子科技大学硕士学位论文

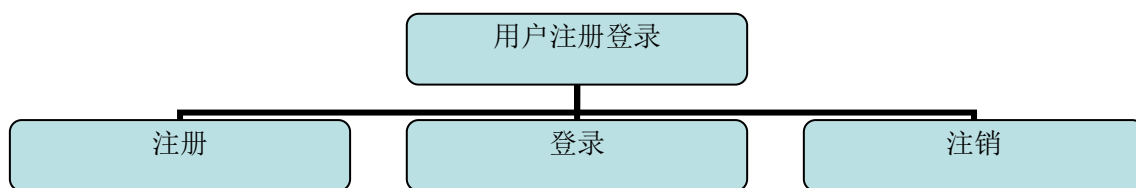


图 3-2：登录注册功能模块

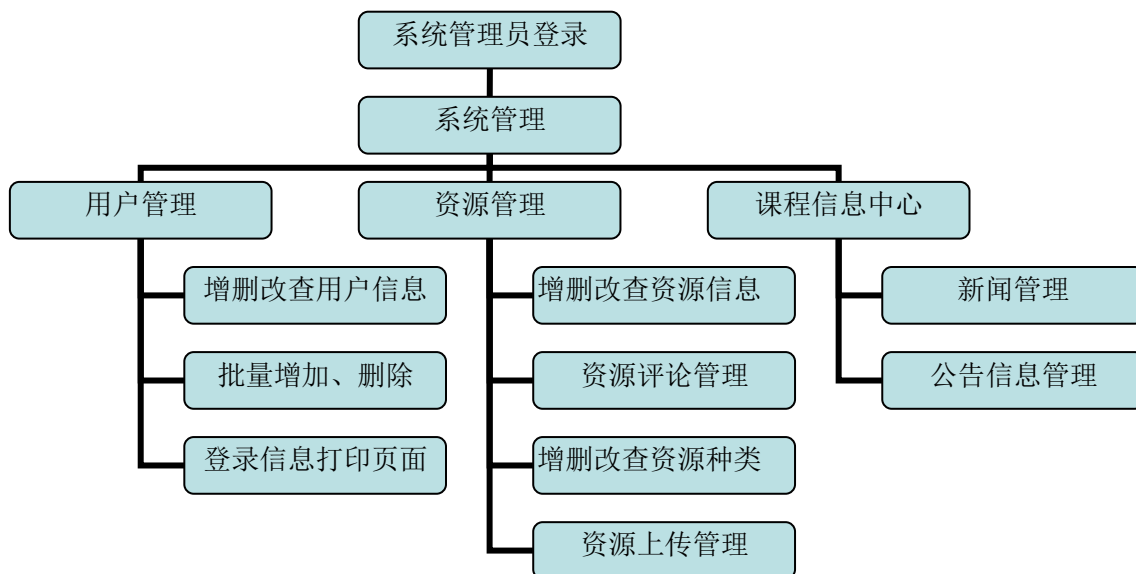


图 3-3：系统管理员主要功能模块

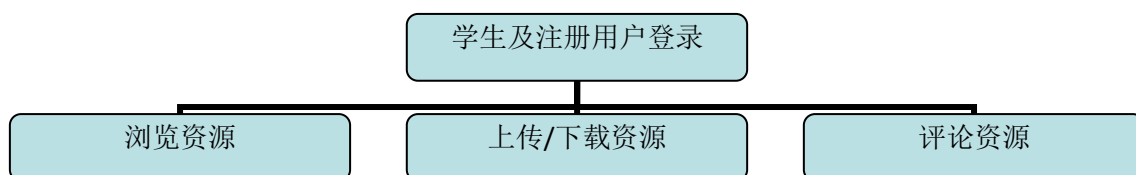


图 3-4：学生及注册用户模块

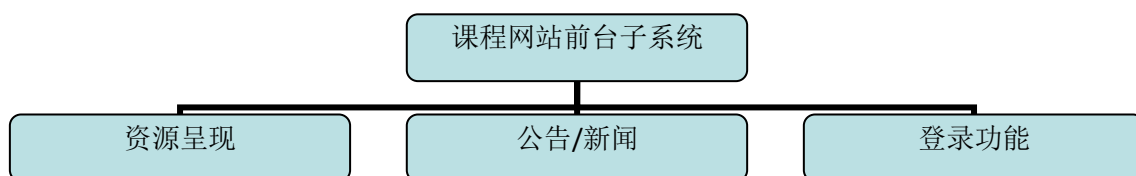


图 3-5：课程网前台子系统

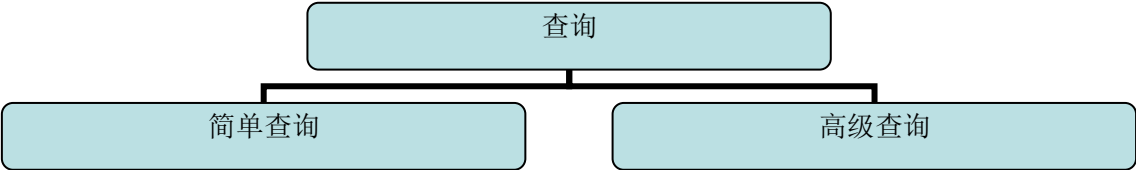


图 3-6: 查询模块

3.2.3 用户权限与功能

作为一个拥有多种用户和读者群的网站，系统需为之提供三种不同的角色支持，并分别在系统中授予不同的权限。这三种不同类型的用户所拥有的功能与权限如表 3-1 所示：

表 3-1:用户权限与功能表

角色	功能与权限
游客	浏览课程资源
注册用户	浏览、上传、有积分下载、评价课程资源
学生	浏览、上传、下载、评价课程资源
老师	浏览、上传、下载、评价课程资源。日后系统升级时还能批改作业，回答问题等
系统管理员	资源管理、用户管理、评价管理

3.3 Ajax 技术应用于精品课程网站的设计

3.3.1 架构设计

本网站将延用原系统的三层架构模式。与两层架构的模式相比，三层及 N 层 C/S 架构能创建更灵活和更具伸缩性的 C/S（客户端/服务器端）环境。

1. N 层架构模式简介

一般 N 层架构模式中，客户端主要负责表示逻辑，而业务逻辑、数据访问逻辑、数据则在服务器端实现，其中业务逻辑和数据访问逻辑一般在应用服务器上，数据则在数据库服务器上。下面分别介绍这几层的作用。

表示层：这一层的主要任务是使用业务逻辑层生成的结果并将结果显示给终端用户。它包括表单、页面、文件等。

业务逻辑层：这一层连接表示层与数据访问层，最终目标通过使用数据访问层将客户端数据访问请求发送到数据层。主要实现对应用程序核心业务的控制。

数据访问层：收到业务逻辑层的调用请求后，发送相关的 SQL 语句到数据层或是调用数据层的存储过程，既不包括业务逻辑也没有数据操作/转换逻辑。

数据层：由一个关系数据库软件支撑，负责数据的存储和检索等工作。

实际应用中，数据访问逻辑是基于微软的 SQLHelper（中小型项目）或 DAAB（Data Access Application Block for .NET，大型项目）。本系统基于微软的 SQLHelper 加 ASP.NET 3.5 下提供的 LINQ to SQL Classes 来完成数据访问工作。

2. 本系统架构

图 3-6 展示了 C#精品课程网站的架构示意图。

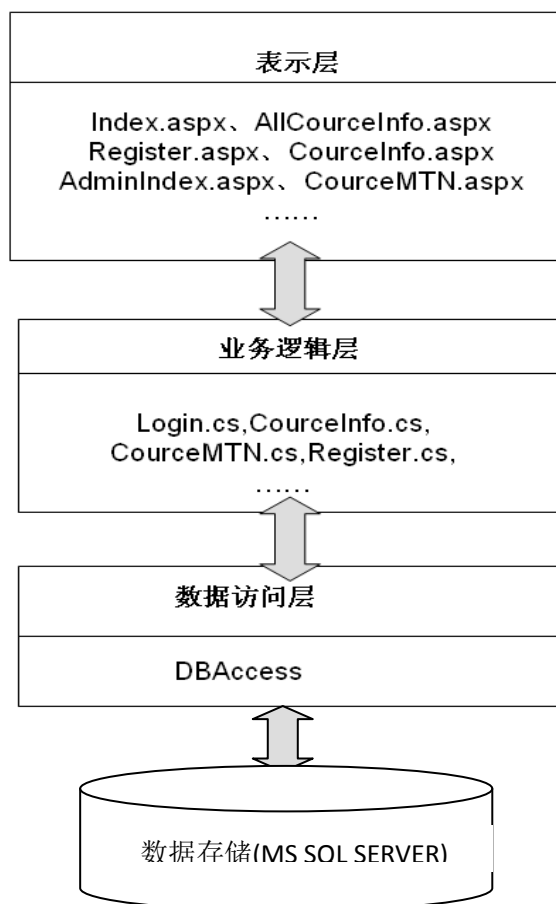


图 3-7：C#精品课程网站架构示意图

图 3-7 中的各层分别说明如下：

数据层：本系统的数据层使用微软的 SQL SERVER 2005 作为后端数据库服务器来存储系统所需的数据。

数据访问层：由一个叫做 DBAccess 的类统一实现对后端数据库的访问。

业务逻辑层：统一位于网站的安全文件夹 App_code 下有若干实现业务逻辑的文件。

表示层：网站中所有的页面、用户控件及主题模板文件。

3.3.2 数据库设计

数据库设计是整个项目开发中的首要环节，设计良好的数据库，将为日后的系统维护、变更及更能扩充提供方便。数据库设计时首先需要完成的工作是数据库需求分析，根据 C#精品课程网站的功能需求，可以简单归纳出数据需求信息：用户数据、资源类别数据、资源数据、评价数据、公告数据、新闻数据等。根据这些数据将可以设计出系统所需的数据库表。

根据数据需求分析的结果，最终确定系统的数据库结构与相关字段，下面用表列出系统中核心的数据表。另外，除了数据表，我们还在数据库中定义了大量的存储过程，以加强系统的模块化与安全性设计。

表 3-2：系统角色信息

字段名	字段类型	字段含义
RoleID	int	主键（自动）
RoleName	varchar(50)	角色名称

表 3-3：用户信息

字段名	字段类型	字段含义
UserID	int	主键（自动加 1）
UserName	varchar(50)	用户昵称
RealName	varchar(50)	真实姓名
Password	varchar(255)	口令
Address	varchar(200)	地址
Phone	varchar(20)	电话号码
Mobile	varchar(20)	手机
Email	varchar(200)	电邮地址
CreateDate	datetime	注册日期
Status	bit	当前状态
RoleID	int	外键，引用 Role 表中字段 RoleID
Remark	text	备注

表 3-4：资源种类

字段名	字段类型	字段含义
CategoryID	int	主键（自动加 1）
CName	varchar(200)	种类名称
Remark	text	备注

表 3-5：资源表

字段名	字段类型	字段含义
DetailID	int	主键（自动加 1）
Name	varchar(50)	资源的详细名称
ChapterNo	int	章号
SectionNo	int	节号
Description	text	详细描述
CreateDate	datetime	创建日期
DetailPath	varchar(500)	资料路径
Mark	int	评分
MarkCount	int	评分人数
Remark	text	备注

表 3-6：资源评论信息表

字段名	字段类型	字段含义
CommnetID	int	主键（自动加 1）
Title	varchar(200)	评论标题
Body	varchar(200)	评论内容
Date	datetime	日期
DetailID	int	评论对应的资源，外键引用
UserID	int	评论对应的用户，外键引用

表 3-7: 公告信息表

字段名	字段类型	字段含义
NotifyID	int	主键（自动加 1）
Title	varchar(200)	公告标题
Body	text	公告详细内容
Date	datetime	公告的创建日期

表 3-8: 新闻信息表

字段名	字段类型	字段含义
NewsID	int	主键（自动加 1）
Title	varchar(200)	新闻标题
Body	text	新闻详细内容
Date	datetime	新闻创建日期时间

3.3.3 系统中需用 Ajax 技术进行改进的功能

原有的 C#精品课程网站中有许多功能，从用户体验度的角度来看的话都需要改进。我们在改造的过程中也尽最大可能地改进各项功能的用户友好度，可以说通过这种改进，整个课程网站实现了 Ajax 化的目标，下面列举其中几个典型的改进功能，对于类似的功能改进在此就省略了。

1. 用户登录注册功能的 Ajax 改进

本课程网站的上传和下载功能需要用户登录后方能进行操作，原有系统中当用户进入网站后，没有登录就直接点击上传下载功能系统页面会自动跳转到另一个登录页面，此时就产生了页面跳转；而用户好不容易找到的资源在登录之后又需要重新寻找，显然这种跳转页面的方式不是最理想的方法，许多用户在页面跳转后甚至就会放弃资源的下载操作，这是精品课程的建设者们不愿看到的。通过 Ajax 技术改进后，则可以当用户没有登录就点击了上传下载资源按钮时，在当前页面上弹出一个模态对话框，此对话框的内容就是登录界面以及注册链接。用户登录成功后，模态对话框会自动消失，此时用户的下载或上传功能就可以使用了。这项功能的改进将大大方便用户，因为许多用户都有一个习惯进入网站首先浏览资源，看中了，才会考虑登录或者是注册。

2. 树形导航的 Ajax 技术优化

本课程网站的所有课程资源的组织是通过树形目录控件来组织的,这种控件显示的内容层次结构清晰,便于用户定位寻找资源。我们考虑需要沿用,但是这种控件有一个弊端,那就是每当用户在其中的一个节点上点击都会导致一次页面回发,即刷新,这也是网络用户不愿看到的,因为即便是页面闪烁用户能够忍受,页面的延迟也是用户不能忍受的。通过 Ajax 技术改进这项功能的目标是,当用户点击该控件的某个节点时不会导致整个页面的回发,而且每展开一个节点,只获取当前节点的直接子节点信息,依此类推。通过这种改进,客户端与服务器端的通信量将减少,页面延迟问题也可以得到改善。

3. 资源描述的 Ajax 技术优化

通过树形目录呈现的资源只是资源的名称,如果用户想要了解关于某个资源的概况,如:资源的作者、创建时间、修改时间、内容描述等,就必须要将该资源页面完全打开。这种方式在用户如果只是大致浏览所有资源概况的情况下,就不得不通过频繁地切换页面来完成。采用 Ajax 技术改进这项功能后,当用户点击该资源的名称时将会随着鼠标弹出一个模态对话框,用户关闭该模态对话框后,又可以继续查看其他资源的概况,整个过程没有页面刷新,即点即出,就像桌面应用的工具提示功能一样方便(tooltips)。

4. 查询功能的 Ajax 技术优化

用户如果使用 Google 搜索引擎会发现一项很智能的功能,当你在查询文本框中键入关键词的一部分时,与此匹配的相关项就会出现在文本框的下拉提示中。这种关键字智能查询提示对用户查询而言是非常方便的。如果在资源查询时能够使用这种智能查询提示功能,无疑给课程资源的使用者们提供了极大的方便。这种功能的实现也需要使用到 Ajax 技术。改进后,用户在智能提示的帮助下将可以更快地找到想要的资源。

3.4 本章小结

本章全面地介绍了一门精品课程网站的功能设计和数据库设计,并提出了 Ajax 技术应用到该网站后的改进点。内容涉及到项目背景介绍、系统功能需求分析及功能结构图、系统数据需求分析及数据库表设计,在介绍完系统的整体情况后,针对系统中较突出的用户体验度问题,提出了使用 Ajax 技术进行优化的目标。本章内容为下一章应用 ASP.NET Ajax 技术实现国家精品课程网站提供了实施蓝图。

第四章 精品课程网站的设计与实现

4.1 课程网站核心功能模块设计与实现

4.1.1 系统登录模块的实现

1. 系统登录注册功能设计

客户注册与登录功能主要在首页上提供。如果用户进入资源页后，仍然没有登录，则登录和注册功能会显示在页面顶部。无论是哪种形态，登录框和注册链接会以嵌套 fieldset 的形式提供在页面上。Fieldset 被包含在一个表单中，表单可将收集到的用户帐号和密码提交到后台服务器进行处理，也可以通过单击注册链接从而进入注册页面。当在客户登录或注册过程中出现一些服务器响应时，这个表格还有显示响应信息的功能。图 4_1 显示了会员登录注册框设计时界面

用户登录

用户名

姓名不能为空

密码

密码不能为空

登录

注册

图 4_1 首页的登录注册框设计时界面

登录功能对应的顺序图如下所示：

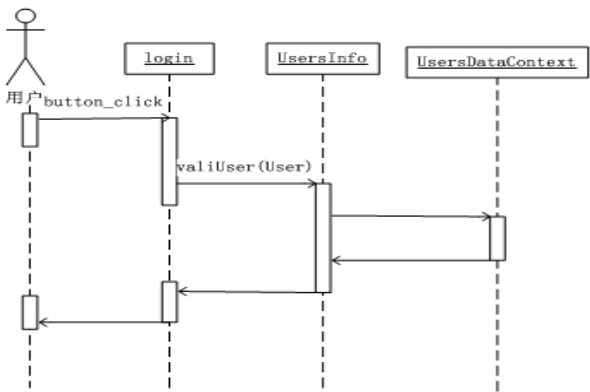


图 4_2:登录功能顺序图

2. 客户登录流程及实现代码

当客户在首页 `index.aspx` 中填写帐号和密码信息后，点击“登录”按钮，将触发登录事件，该事件在后台代码处有绑定的事件处理程序。事件处理程序将首先创建一个实体对象 `user`，进而 `userInfo` 类中的 `ValidUser` 方法将会验证用户名和密码。如果用户名密码正确，该用户的信息将被保存在会话对象中。以备后续受保护页面能够知晓用户的登录信息。使用受保护功能时，将会从会话对象中获取用户信息，如果没有用户信息将会自动阻止用户进入该功能。

该功能对应的代码如清单 4_1 所示，这段代码描述了用户登录页面的 HTML 视图。页面上除拥有基本的文本域控件外，还添加了验证控件，以及注册链接。

代码清单 4_1:

```
<form id="form1" runat="server">
    <fieldset>
        <legend>用户登录</legend>
        <p>
            <label>用户名</label>
            <asp:TextBox ID="tbUser" runat="server"
                Width="155px">
            </asp:TextBox>
            <asp:RequiredFieldValidator ControlToValidate="tbUser"
                ID="RequiredFieldValidator1" runat="server"
                ErrorMessage="姓名不能为空">
            </asp:RequiredFieldValidator>
        </p>
        <p>
            <label>密      码</label>
            <asp:TextBox ID="tbPassword" runat="server" TextMode="Password"
                Width="155px">
            </asp:TextBox>
            <asp:RequiredFieldValidator ControlToValidate="tbPassword"
                ID="RequiredFieldValidator2" runat="server"
                ErrorMessage="密码不能为空">
            </asp:RequiredFieldValidator>
        </p>
    </fieldset>
</form>
```


[illegible]

设计页面对应的部分后台代码如清单 4_2 所示，代码中 `Page_Load` 方法主要完成了用户重复登录的判断，而针对“登录”按钮的事件处理程序中则调用了业务逻辑层的代码实施验证（业务逻辑层的代码是通过数据库访问层的代码实施获得数据库中的用户数据），当验证成功后，需将信息保存到用户会话对象中，这样当用户在其他页面中也能够获知用户是否成功登录的信息。代码清单 4_3 反映了在业务逻辑层中进行数据验证的代码是如何工作的。

代码清单 4_2:

```
public partial class Login : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            var user = Session["user"];
            if (user != null)
            {
                Response.Write("<script type='text/javascript'>
                    alert('你已经登录'),history.go(-1)</script>");
            }
        }
    }
}
```

```
protected void btnLogin_Click(object sender, EventArgs e)
{
    Users user = new Users()
    {
        UserName = tbUser.Text.Trim(),
        UserPassword = tbPassword.Text.Trim()
    };
    bool flag = new UsersInfo().ValiUser(user);
    if (flag)
    {
        Session["user"] = user;
        Response.Redirect("infomanger.aspx");
    }
    else
    {
        Response.Write("<script type='text/javascript'>
            alert('用户名或密码不对')</script>");
    }
}
}
```

代码清单 4_3:

```
public bool ValiUser(Users user)
{
    bool flag = false;
    using (UsersDataContext db = new UsersDataContext())
    {
        int c = db.Users.Where(u => u.UserName == user.UserName)
            .Where(u => u.UserPassword == user.UserPassword)
            .Count();
        if (0 < c)
```

```
{
    flag = true;
}
return flag;
}
```

4.1.2 课程资源展示模块设计与实现

1. 课程资源展示模块设计

当用户进入课程资源页面，即可见到树型目录。树型目录的组织按照课程内容的章节顺序。每小节安排有视频、PPT、教案、其他资源四个方面的内容。当用户选择某个小节名称时，页面将跳转到资源详细显示页面。资源详细显示页面将从数据库取得相应小节的四方面内容。资源详细页面的左部是导航栏，右部将显示视频、教案或 PPT。

课程资源页面的设计时视图如图 4_3 所示。

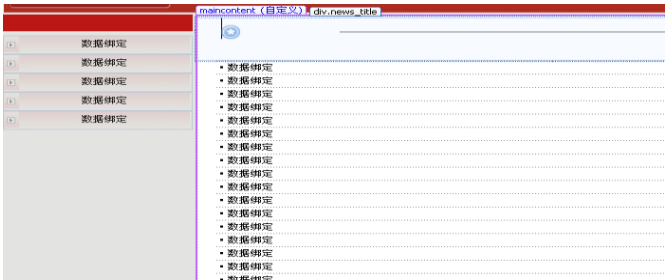


图 4_3:课程资源页面设计时视图

课程资源显示页面对应的顺序图如下所示：

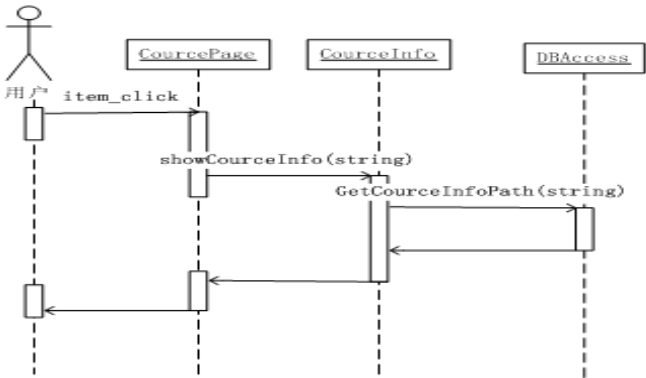


图 4_4:课程资源显示模块顺序图

2. 课程资源展示模块实现

当用户在资源详细显示页面点击导航栏的某个按钮时，对应按钮的所有资源将会显示在屏幕右边。该页面对应的代码如代码清单 4_4 所示。代码反映了这是一个母版页页面，并且使用了 Visual Studio 2008 中新增的 ListView 控件，ListView 控件的数据将从一个 XML 文件中获得，而该文件的内容则是从数据库读出。通过类 ReadContentsFromDB.cs 类完成将数据库数据到 XML 文档的转化过程。

代码清单 4_4:

```
<%@ Page Language="C#" MasterPageFile="~/site.Master" AutoEventWireup="true"
CodeBehind="contentindex.aspx.cs" Inherits="Organization.contentindex" Title="湖南科技
职业学院 C#精品课程网" %>

<%@ Import Namespace="Organization.Models" %>

<asp:Content ID="Content1" ContentPlaceHolderID="maincontent" runat="server">
    <form id="cources" runat="server">
        <div class="cources_title">
            <span ><%=catetitle%></span>
            
        </div>
        <asp:ListView ID="courcesindex" runat="server"
onpagepropertieschanging="courcesindex_PagePropertiesChanging">
            <LayoutTemplate>
                <div id="Div1" class="content_cources" runat="server">

                    <div class="cources_index" id="itemPlaceholder" runat="server"></div>
                </div>
            </LayoutTemplate>
            <ItemTemplate>
                <div class="cources_index">
                    <div class="cources_splitter"></div>

                    <span> </span><a href="CourseInfo.aspx?navid=
                    <%=#(Container.DataItem as courcesIndex).navid%>
```

```

        &coursesid=<%#(Container.DataItem as courcesIndex).courcesid%>"
        target="_blank">
        <%#(Container.DataItem as courcesIndex).courcestitle%></a>

    </div>
</ItemTemplate>
<EmptyDataTemplate>
    谢谢浏览 现在无数据
</EmptyDataTemplate>
</asp:ListView>
<div class="content_pg">
    <!--分页-->
    <asp:DataPager ID="Pager" runat="server"
        PagedControlID="courcesindex" PageSize="20">
        <Fields>
            <asp:NextPreviousPagerField FirstPageText="首页"
                PreviousPageText="上一页"
                NextPageText="下一页"
                LastPageText="末页" ShowFirstPageButton="true"
                ShowLastPageButton="true" />
        </Fields>
    </asp:DataPager>
</div>
</form>
</asp:Content>

```

4.1.3 下载/上传功能模块

1. 下载/上传模块设计

在早期的 HTML 中，表单并不具备上传文件的功能，1995 年后在 RFC1867 规范中对表单做了扩展，表单元素<input type="file">被增加到规范中。如果在表单中使用这个元素，浏览器在解析表单时，会自动生成一个输入框和一个按钮，输

入框可供用户填写本地文件的文件名和路径名，按钮可以让浏览器打开一个文件选择框供用户选择文件。

ASP.NET 提供了一个服务器端控件 <asp:FileUpload/>，功能类似 <input type="file"> 的功能，但他可以直接在 Microsoft Visual Studio 中进行可视化的属性配置。本论文中的精品课程网站使用了 ASP.NET 中的服务器端控件。并通过验证控件实现了对上传文件后缀的控制。

一般来讲，要在应用程序中实现下载功能，只需设置一个超链接，但是这样做暴露了下载文件的真实地址，不利于对资源进行安全保护，而且，利用超链接下载文件，服务器端的文件只能存放在一个 WEB 应用程序所在的目录下。

故在本文中的精品课程网站中是通过编程实现下载，从而增强了程序的安全访问控制，只有得到授权的用户才可以使用下载功能，同时被下载的文件可以放到任意位置，甚至可将文件或文件路径保存到数据库中。文件上传功能如图 4-5 所示，文件下载的功能如图 4_6 所示。

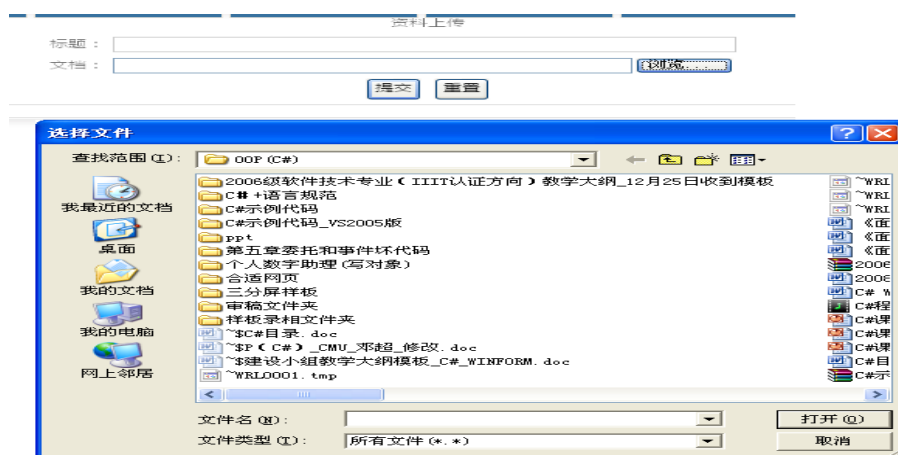


图 4_5:课程资源上传页面



图 4_6:课程资源下载页面

2. 下载/上传模块实现

上传资料的功能需要在用户登录之后才能进行，所以在页面后台代码处进行了用户验证之后才能进入此功能。当用户点击“提交”按钮后，后台代码也做了相应的处理，这部分反映在“提交”按钮的事件处理程序 btnSubmit_Click 方法中。该方法对文件名以及文件后缀进行了判断。具体代码清单如 4_5 所示。

代码清单 4_5:

```
public partial class Upload : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (Session["CurrentUser"] == null)
        {
            Response.Write("<script>alert('请先登陆');");
            document.location='Default.aspx'</script>");
            return;
        }
        if (((usersInfo)Session["CurrentUser"]).checkFlag < 2)
        {
            Response.Write("<script>alert('您没有权限查看此页');");
            document.location='Default.aspx'</script>");
        }
    }

    protected void btnSubmit_Click(object sender, EventArgs e)
    {
        int contentstate = 0;
        string path = Server.MapPath("~/UploadFiles/" +
            DateTime.Today.ToString("yyyyMMdd")
            + DateTime.Now.ToString("hhmmss") + "/"");
        Boolean fileOK = false;
        documents doc = new documents();
        doc.docName = txtTitle.Text;
        doc.date = DateTime.Now;
        doc.docCategoryId = 3;
        usersInfo user = new usersInfo();
        user = Session["CurrentUser"] as usersInfo;
        doc.branchId =
            UsersInfoManagement.GetUsersInfoById(Convert.ToInt32(us
            er.id)).branchId;
        doc.downloadFlag = 1;
        doc.filename = uploadDoc.FileName;
        try
        {
            DocumentsManagement.AddDocuments(doc);
        }
    }
}
```

```

    }
    catch (Exception docerror)
    {
        Console.WriteLine(docerror.Message);
        contentstate++;
    }

    if (uploadDoc.HasFile)
    {
        String fileExtension =
            System.IO.Path.GetExtension(uploadDoc.FileName).ToLower();
        String[] allowedExtensions = { ".txt", ".doc", ".xls",
            ".xlsx", ".docx", ".rar" };
        for (int i = 0; i < allowedExtensions.Length; i++)
        {
            if (fileExtension == allowedExtensions[i])
            {
                fileOK = true;
                break;
            }
            else
            {
                fileOK = false;
            }
        }
    }

    if (fileOK)
    {
        try
        {
            if (!Directory.Exists(path))
            {
                Directory.CreateDirectory(path);
            }
            uploadDoc.PostedFile.SaveAs(path
                + uploadDoc.FileName);
        }
        catch (Exception ex)
        {
            Console.WriteLine(ex.Message);
        }
    }
    else
    {
        Response.Write("<script>alert('Cannot accept files of this type !'
            );</script>");
    }
}

```



```

        if (uploadDoc.HasFile && !fileOK)
        {
            Response.Write("<script>alert('上传格式不正确或上传文件名已存在！！'); history.back()</script>");
        }
        else if (contentstate != 0)
        {
            Response.Write("<script>alert('资料填写错误，请重新填写！！'); history.back()</script>");
        }
        else
        {
            Response.Write("<script>if (confirm('添加成功，是否需要再次添加？')) document.location='Upload.aspx'; else document.location='Default.aspx';</script>");
        }
    }
}

```

程序中下载功能的提供并不是由链接直接指向资源，而是通过后台代码来完成，从而可以只提供此功能给授权用户。并且对资源的状态做出判断。代码清单 4_6 描述了此功能的实现。方法 `Page_Load` 对用户进行了验证，如果没有登录的用户是无法登录此页面的，如果是已登录用户则构造文件，并对文件的存在与否做出判断。如果一切验证通过，则通过设置响应对象 `Response` 相关的属性，以及调用它的 `TransmitFile` 方法完成下载工作。

代码清单 4_6:

```

public partial class Download : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (Session["CurrentUser"] == null)
        {
            Response.Write("<script>alert('请先登录'); document.location='Default.aspx'</script>");
            return;
        }
        int id = Convert.ToInt32(Request.QueryString["id"]);
        if (id == 0)
        {
            Response.Write("<script>alert('非法提交'); document.location='Default.aspx'</script>");
            return;
        }
    }
}

```

```

    }
    else
    {
        documents doc = DocumentsManagement.SelectDocumentById(id);
        string path = Server.MapPath("UploadFiles/" + doc.filename);

        FileInfo downloadFile = new FileInfo(path);

        if (!File.Exists(downloadFile.FullName))
        {
            Response.Write("<script>alert('下载的文件不存在  
'");
            document.location='FileList.aspx'</script>");
            return;
        }

        string DEFAULT_CONTENT_TYPE = "application/unknown";
        RegistryKey regkey, fileextkey;
        string FILE_CONTENT_TYPE;
        try
        {
            regkey = Registry.ClassesRoot;
            fileextkey = regkey.OpenSubKey(downloadFile.Extension);
            FILE_CONTENT_TYPE = fileextkey.GetValue("Content Type",
                DEFAULT_CONTENT_TYPE).ToString();
        }
        catch (Exception ex)
        {
            Console.WriteLine(ex.Message);
            FILE_CONTENT_TYPE = DEFAULT_CONTENT_TYPE;
        }

        Response.ClearHeaders();
        Response.Clear();
        Response.Buffer = false;
        Response.AddHeader("Content-Disposition", "attachment;filename=" +
            HttpUtility.UrlEncode(downloadFile.Name,
                System.Text.Encoding.UTF8));
        Response.AddHeader("Content-Length",
            downloadFile.Length.ToString());
        Response.ContentType = "application/octet-stream";
        Response.TransmitFile(downloadFile.FullName);
        Response.Flush();
        Response.Close();
        Response.End();
    }
}
}

```

4.1.4 后台管理员维护功能模块

1. 后台管理员维护功能模块设计

后台管理员的功能涉及到用户管理、资料管理、公告管理，这些管理在技术实现上有非常相似之处，那就是都将涉及到对数据库的记录进行 **CRUD** 操作。因为这些功能的特点，本文中将以用户管理为例，介绍这些功能的设计与实现。

用户管理涉及到显示全部用户的信息、删除用户，更新用户（包括更改用户的权限）的信息。具体功能展示如图 4-7、图 4-8 所示。

序号	用户名	真名	QQ	电话	所属工会	所属部门	权限	操作
1	birdie	邓超	124968523	13517310450	软件分会	软件学院	1	详细 删除
2	admin	admin	123	123456	软件分会	软件学院	1	详细 删除
3	nihao	你好	1234			软件学院	3	详细 删除
4	jintian	今天	1234			软件学院	1	详细 删除
5	soft	软件	1234			软件学院	2	详细 删除
6	mybirdie	超级用户	123	123	软件分会	软件学院	9	详细 删除
7	bird	大斌	123456	1234567890	教务分会	电子信息工程与技术系	1	详细 删除

图 4_7:用户资料显示及删除功能

用户名	birdie
电话	13517310450
E-mail	mybirdie@126.com
QQ	124968523
所属部门	1
真名	邓超
权限	超级管理员
<input type="button" value="更新"/> <input type="button" value="取消"/>	

图 4_8:用户权限更新功能

2. 后台管理员维护功能模块实现

后台管理员维护功能主要涉及对各个实体对象的增、删、改、查，为了程序的维护升级，我们将此部分功能全部封装在业务逻辑层。因此业务逻辑层将涉及的类有用户维护类 `UsersInfoManagement`、资料维护类 `DetailsManagement` 等。下面将以用户维护类为例，介绍其代码实现。

代码清单 4_7 将主要围绕用户资料的增加、删除、修改、查询展开。清单中 `AddUsersInfo`、`DeleteUsersInfoById`、`UpdateUserInfo` 将用于用户资料的增、删、改；而 `GetUsersInfoById(int id)`、`GetUsersInfoByUsername(string username)`、`GetUsersInfoByUsernameAndPassword(string username, string password)`、`GetUsersInfoByUsernameAndPasswordAndFlag(string userName, string pwd, int flag)`、`GetAllUserInfo()` 则用于查询，这些查询方法分别用于根据 ID 号查询，根据用户名称查询、根据用户名和密码进行查询、根据用户名、密码和标志位查询，以及查询全部用户信息。

代码清单 4_7:

```
namespace HNKYTradeUnion.BLL
{
    public class UsersInfoManagement
    {
        private static HNKYTradeUnionDataContext dc;
        #region 增加
        public static bool AddUsersInfo(usersInfo user)
        {
            dc = new HNKYTradeUnionDataContext();
            try
            {
                dc.usersInfo.InsertOnSubmit(user);
                dc.SubmitChanges();
                return true;
            }
            catch (Exception ex)
            {
                Console.WriteLine(ex.StackTrace);
                return false;
            }
        }
        #endregion

        #region 删除
        public static void DeleteUsersInfoById(int id)
        {
            dc = new HNKYTradeUnionDataContext();
            try
```

```

        {
            var del = from d in dc.usersInfo where d.id == id select d;
            foreach (var e in del)
            {
                dc.usersInfo.DeleteOnSubmit(e);

                dc.SubmitChanges();
            }
        }
        catch (Exception ex)
        {
            Console.WriteLine(ex.StackTrace);
        }
    }
#endregion

#region 查询
public static usersInfo GetUsersInfoById(int id)
{
    dc = new HNKYTradeUnionDataContext();
    try
    {
        var user = from d in dc.usersInfo where d.id == id
                    select d;
        return user.First();
    }
    catch (Exception ex)
    {
        Console.WriteLine(ex.StackTrace);
        return null;
    }
}

public static usersInfo GetUsersInfoByuserName(string username)
{
    dc = new HNKYTradeUnionDataContext();
    try
    {
        var user = from d in dc.usersInfo where d.userName ==
                    username select d;
        return user.First();
    }
    catch (Exception ex)
    {
        Console.WriteLine(ex.StackTrace);
    }
}

```

```

        return null;
    }
}

public static usersInfo GetUsersInfoByuserNameAndPwd(string
    username, string password)
{
    dc = new HNKYTradeUnionDataContext();
    try
    {
        var user = from d in dc.usersInfo where d.userName ==
            username && d.password == password select d;
        return user.First();
    }
    catch (Exception ex)
    {
        Console.WriteLine(ex.StackTrace);
        return null;
    }
}

public static usersInfo
GetUsersInfoByUserNameAndPwdAndFlag(string userName, string pwd, int flag)
{
    dc = new HNKYTradeUnionDataContext();
    try
    {
        var user = from d in dc.usersInfo where d.userName ==
            userName && d.password == pwd && d.checkFlag == flag
            select d;
        return user.First();
    }
    catch (System.Exception ex)
    {
        Console.WriteLine(ex.StackTrace);
        return null;
    }
}

public static IList<usersInfo> GetAllUserInfo()
{
    dc = new HNKYTradeUnionDataContext();
    try
    {

```

```

        var user = from d in dc.usersInfo select d;
        return user.ToList();
    }
    catch (System.Exception ex)
    {

        Console.WriteLine(ex.StackTrace);
        return null;
    }
}

#endregion

#region 修改
public static void UpdateUserInfo(usersInfo user)
{
    dc = new HNKYTradeUnionDataContext();
    try
    {
        var query = (from d in dc.usersInfo where d.id == user.id
                      select d).First();
        query.userName = user.userName;
        query.password = user.password;
        query.mobile = user.mobile;
        query.email = user.email;
        query.qq = user.qq;
        query.branchId = user.branchId;
        query.departmentId = user.departmentId;
        query.realName = user.realName;
        query.checkFlag = user.checkFlag;
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {
        Console.WriteLine(ex.StackTrace);
    }
}
#endregion
}
}

```

4.2 C#国家精品课程网站 Ajax 应用功能示例

4.2.1 设计 Ajax 型的母版页

因为此次精品课程网站改造主要是以用户为中心的体验度交互度改造，各个页面都或多或少会需要添加 Ajax 功能。而基于 ASP.NET AJAX 3.5 实现的 Ajax 页面，每个页面都至少要包括且只能包括一个 ScriptManager 控件，考察原有网站的源代码后，发现原有网站使用了母版页，故在改造实现时，我们决定采用在母版页中添加 ScriptManager，这样其他应用母版页的页面就无需再添加此控件，只需要在实现局部更新的位置添加 UpdatePanel 控件，如果个别页面需要有特殊的数据来源要求，可以通过添加 ScriptManagerProxy 控件实现。要完成此功能，只需按如下步骤设计即可（关于本页面的设计时视图请见图 4-5）。

1. 打开网站原有的母版页 MasterPage.master 的设计视图，从工具箱的 Ajax Extensions 标签页中添加一个 ScriptManager 控件到页面，此控件负责管理所有页面的客户端脚本。
2. 从工具箱的 Ajax Extensions 标签页中添加一个 UpdateProgress 控件到页面，此控件在页面中出现费时操作时，给用户以提示信息。

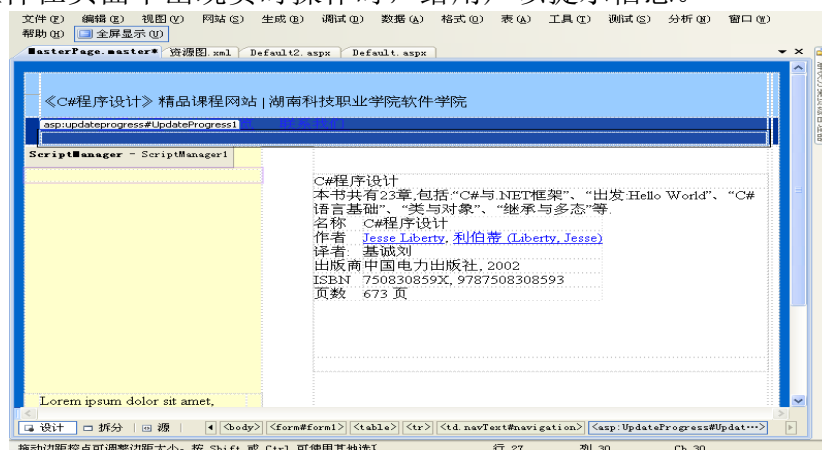


图 4-9: C#精品课程网站母版页设计时视图

4.2.2 用户登录注册功能的 Ajax 改进实现

用户已经找到了合适的资源，准备下载时仍然还没有登录，系统应当给出提示或是页面跳转，但这种情况下使用 Ajax 风格模态对话框直接将登录对话框交给用户应该是用户最急需的，且页面不会跳转（好不容易找到的资源，因为登录跳转而不见了，用户总是会有些不快的）。要实现此功能，定位到系统中的资源（DetailsView.aspx）显示页面，按如下步骤修改页面即可（效果图见图 4-4）：

1. 单击 DetailsView.aspx 页面中的 LinkButton 按钮“下载”右上角的“<”符号，会弹出“添加扩展程序……”菜单，单击此菜单会出现“扩展程序向导”对话框。
2. 在“扩展程序向导”对话框中选择 ModalPopupExtender 扩展器，并给其指定一个 ID 号。
3. 在当前页面上的某个位置添加一个 Panel 控件，一个 Label 控件和一个 LinkButton 控件。它们的作用分别是占位、显示登录框、关闭按钮。与此致关的 HTML 代码如代码清单 4-5 所示：

代码清单 4-8

```
<asp:LinkButton ID="btnDownload" runat="server">下载此资源</asp:LinkButton>
<div align="left">
<asp:Panel ID="Panel1" runat="server" >
<asp:Label ID="Label20"runat="server" Text=""></asp:Label>
<br/>
<div align="right">
<asp:LinkButton ID="LinkButton1" runat="server" Font-Bold="True"> 关 闭
</asp:LinkButton>
</div>
</asp:Panel>
</div>
<br/>
<ajaxToolkit:ModalPopupExtender ID="ModalPopupExtender1" runat="server"
Enabled="True" TargetControlID="btnDownload" PopupControlID="Panel1"
DynamicControlID="Label20"
DynamicServicePath='<%#Page.ResolveClientUrl("~/LoginService.aspx")%>'
DynamicServiceMethod='Login' OkControlID="LinkButton1">
</ajaxToolkit: ModalPopupExtender>
```

上表中的 ModalPopupExtender 扩展器控件通过 PopupControlID 属性与充当占位符的 Panel 关联，通过 TargetControlID 属性关联到使模态对话框弹出的 btnDownload 按钮。

4. 上面的代码有一个 DynamicServicePath 属性关联到的是一个 Web 服务方法，此方法返回的结果会在 Label 控件上显示。此方法会通过 StringBuilder

构建一个包含 HTML 标签组成的字符串，这些复杂的 HTML 标记的作用就是显示出登录对话框来。



图 4-10:模态对话框显示登录对话框

4.2.3 树形导航的 Ajax 技术优化实现

前面提到树形导航因其表达层次结构清晰深受开发者的喜爱，但它也有一个问题，那就是每当用户点击任一个节点都会导致一次屏幕刷新，所以需要使用 Ajax 技术对其进行优化。下面是该树形导航控件的代码清单，从中不难发现，我们通过在页面上增加一个 UpdatePanel 控件，并将 TreeView 控件放置在其子标签 ContentTemplate 中来实现树形导航控件的局部刷新。图 4-7 反映的是其设计时视图。

代码清单 4-9:

```
<asp:UpdatePanel ID="UpdatePanel1" runat="server">
    <ContentTemplate>
        <asp:TreeView ID="TreeView1" runat="server"
DataSourceID="XmlDataSource2">
            <DataBindings>
                <asp:TreeNodeBinding DataMember="c1" Depth="1"
TargetField="#Name" />
                <asp:TreeNodeBinding DataMember="s1" TargetField="name" />
                <asp:TreeNodeBinding DataMember="r1" />
                <asp:TreeNodeBinding DataMember="r2" />
                <asp:TreeNodeBinding DataMember="r3" />
                <asp:TreeNodeBinding DataMember="r4" />
                <asp:TreeNodeBinding DataMember="c2" Depth="1"
TargetField="#Name" />
                <asp:TreeNodeBinding DataMember="s1" />
                <asp:TreeNodeBinding DataMember="r1" />
                .....
                <asp:TreeNodeBinding DataMember="r4" />
                <asp:TreeNodeBinding DataMember="s2" />
                <asp:TreeNodeBinding DataMember="r1" />
            </DataBindings>
        </asp:TreeView>
    </ContentTemplate>
</asp:UpdatePanel>
```

```
<asp:TreeNodeBinding DataMember="r2" />
<asp:TreeNodeBinding DataMember="r3" />
<asp:TreeNodeBinding DataMember="r4" />
<asp:TreeNodeBinding DataMember="c3" Depth="1"
TargetField="#Name" />
<asp:TreeNodeBinding DataMember="s1" />
.....
<asp:TreeNodeBinding DataMember="r2" />
<asp:TreeNodeBinding DataMember="r3" />
<asp:TreeNodeBinding DataMember="r4" />
</DataBindings>
</asp:TreeView>
<asp:XmlDataSource ID="XmlDataSource2" runat="server" DataFile="~/资源
图.xml">
</asp:XmlDataSource>
</ContentTemplate>
</asp:UpdatePanel>
```

实现树形导航控件的局部提交是非常简单，但这里有一个技巧，那就是树形导航控件的数据源问题，我们采用的是 XMLDataSource 控件。

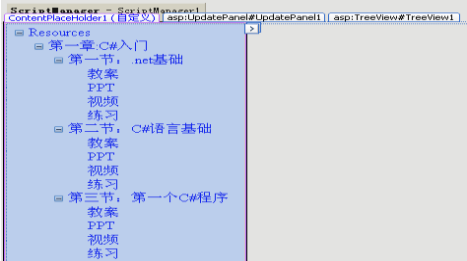


图 4-11：树型导航显示所有课程资源

4.2.4 查询功能的 Ajax 技术优化实现

一般稍大型的门户网站或是邮箱系统，以及消费型网站都提供了智能查询功能。即只要用户输入部分关键字，与此部分匹配的所有关键字便都显示出来（如图 4_8:查询智能提示功能）。智能提示功能给用户带来了方便，应用于课程网站能够吸引更多的用户使用该功能，因此，在实现课程网站时，这一功能得到了应用。



图 4-12:查询智能提示功能

1. AutoCompleteExtender 扩展器+WCF 服务实现查询关键字智能提示

要实现查询关键字智能提示在页面上得添加 AutoCompleteExtender 扩展器控件支持，后台还需要通过 Web 服务来实现连接数据库，并在数据库中搜索满足条件的内容。因为母版页上已经有一个 ScriptManager, 此时因为我们需要连接到不同的服务，所以需要借助 ScriptManagerProxy 来实现给当前页面添加额外引用的目标。

代表清单 4-10:

```
<asp:ScriptManagerProxy ID="ScriptManagerProxy1" runat="server">
    <Services>
        <asp:ServiceReference Path="~/ResourceInfoService.svc" />
    </Services>
</asp:ScriptManagerProxy>
```

接着我们需要针对页面上的查询文本框添加 AutoCompleteExtender 扩展器控件。代码清单 4-4 显示了添加后页面上的代码情况。表 4-1 显示了该控件的常用属性及其含义。

代表清单 4-11:

```
<asp:TextBox ID="KeyWord" runat="server"></asp:TextBox>
<ajaxToolkit:AutoCompleteExtender
ID="AutoCompleteExtender1"
runat="server"
TargetControlID="txtSearch"
ServiceMethod="GetResourceList"
ServicePath="~/ResourceInfoService.svc"
MinimumPrefixLength="1">
</ajaxToolkit:AutoCompleteExtender>
```

TargetControlID	指定要实现提示功能的控件;
ServicePath	WebService 的路径, 提取数据的方法是写在此 WebService 中
ServiceMethod	写在 WebService 中的用于提取数据的方法的名字
MinimumPrefixLength	用来设置用户输入多少字母才出现提示效果
CompletionSetCount	设置提示数据的行数
CompletionInterval	从服务器获取数据的时间间隔, 单位是毫秒。

表 4-1: AutoCompleteExtender 控件常用属性及其含义

2. 使用 TextBoxWatermarkExtender 扩展器控件实现水印输入效果

除智能提示外，我们还在查询文本框中通过使用 UpdatePanel 控件及 TextBoxWatermarkExtender 扩展器实现了文本框水印提示效果，当文本框中没有数据时，可用水印文字给用户一些输入提示，一旦用户开始输入内容，这些水印文字就会消失，实现上述效果如代码清单 4-9 所示。

代码清单 4-12

```
<asp:TextBox ID="KeyWord" runat="server"></asp:TextBox>
<ajaxToolkit:TextBoxWatermarkExtender ID="tbweResources"
runat="server" TargetControlID="KeyWord" WatermarkText="请输入资源名
称……" WatermarkCssClass="watermarked"/>
```

4.3 系统运行及测试结果

4.3.1 系统运行环境的搭建

本系统的运行必须要使用 .NET Framework 3.5 和 IIS 服务器。 .NET Framework 3.5 从微软官方网站免费下载获得，直接运行安装程序即可完成安装过程。 IIS 服务器是微软公司发布的一款主流 Web 服务器，通过它能够方便地在 Intranet（局域网）和 Internet（因特网）上发布 Web 应用程序。 Windows 2000 Advanced Server 和 Windows XP 操作系统中已经包含了 IIS 5.1， Windows Server 2003 操作系统中已经包含了 IIS 6.0，而 Windows Vista 操作系统则默认包含 IIS 7.0。

4.3.2 系统程序的安装和加载

将整个系统文件夹 CSharpCourseNet 拷贝到 IIS 的默认根目录下即可完成该项目的部署，该路径通常默认为： C:\Inetpub\root\。

本系统采用的数据库产品是 Microsoft 的 SqlServer2005，系统所涉及的数据库及其库中的所有表和存储过程都在其中，数据库名称为 csharpcourcenetDB，通过 sqlserver 2005 的主界面可以完成数据库的附加，附加成功后，数据库 csharpcourcenetDB 将被添加到 SqlServer 2005 数据库服务器中。

最后，测试系统的加载是否成功，首先确保 IIS 服务器的服务处于运行状态，然后在 IE 浏览器的地址栏键入 <http://localhost/CSharpCourseNet/>，如果主页能够显示，证明成功。

4.3.3 Web 应用系统测试分析

针对传统的软件进行的测试，与针对 Web 应用系统的测试既有相同的地方，也有不同之处，Web 应用系统的测试是测试领域的新课题。针对 Web 应用系统的测试除按照传统软件的要求，需要检查和验证是否满足最初的设计要求，而且还需要评价系统在各种浏览器和终端的显示兼容性。最重要的是，还要以终端用户的身份对软件的安全性和可用性进行测试，Web 应用程序的可用性在欧美一些国家甚至已经有相关立法，但在我国还只是起步阶段。本文工作的重点是软件的可用性，因此针对本文所设计系统的测试也将以可用性测试作为重点。

1. 功能测试

1) 链接测试

Web 应用系统中页面之间的切换以及跳转主要通过系统中若干的链接完成。因此针对链接的测试，首先需要测试所有链接是否按其设置能够正确链接到目标页面；另外需要测试目标页面在系统中是否存在且能够正确显示。值得关注的是，孤立页面的存在对系统的可用性也是有害的，因为孤立页面必须直接在 URL 地址栏中输入完整正确的地址才能访问，整个系统中根本不存在指向该页面的链接。

针对本系统的链接测试在集成测试阶段进行，因为涉及测试的页面较多，为了提高效率，我们采用了一款链接测试工具 HTML Link Validator，通过该软件的测试表明，本课程网站系统中不存在无法链接的内容。

2) 表单测试

表单是用户提交信息到 Web 服务器的重要手段，用户注册、登录、信息的‘增删改查’操作等都需要通过表单来传递数据。针对表单的测试主要围绕提交操作的完整性展开，以保证提交给服务器的信息是正确有效的。本系统中涉及的表单有登录表单、注册表单、课程资源增加表单、新闻增加表单、公告增加表单等。

表 3-2 列出了针对用户注册表单数据完整性所做的测试用例。表 3-3 列出了测试步骤。

字段名称	输入数据	预期结果	说明
用户名	debbie	效验通过	正确（由不超过 16 位字符组成）且唯一的用户名。
	“ ”	效验不通过	用户名不能为空
	~sony#	效验不通过	用户名不允许出现空格和特殊字符（~、#、&、（）等）
密码	Dcaaa133	效验通过	正确（由大于 6 位小于 16 位的字符或数字组成）

	“”	效验不通过	密码不能为空
	aa	效验不通过	密码位数必须大于 6 位
	Aaa1aa3aaa4a3aa78	效验不通过	密码位数必须小于 16 位
确认密码	Dcaaa133	效验通过	正确(与密码相同)
	Dcaaa132	效验不通过	正确(与密码不相同)
出生日期	2011-1-1	效验不通过	出生日期超过了服务器当前日期
	1998-1-1	效验通过	正确（小于服务器当前日期）
移动电话	13517310450	效验通过	正确（小于 11 位的数字）
	A1343214a78	效验不通过	不能包含除数字外的字符
E-mail 地址	abc@126.com	效验通过	有效邮件地址
	abc	效验不通过	无效邮件地址
验证码	与界面提示一样	验证通过	用户输入与系统随机生成相同
	与界面提示不一样	验证不通过	用户输入与系统随机生成不相同

表 3-2 “用户注册”功能表单测试用例

<p>步骤:</p> <p>输入“用户名”</p> <p>输入“密码”</p> <p>输入“确认密码”</p> <p>输入“出生日期”</p> <p>输入“电话号码”</p> <p>输入“E-mail 地址”</p> <p>输入“验证码”</p> <p>单击“注册”按钮</p> <p>结果:</p> <p>“预期结果”：错误则提示用户，正确则回到拥有登录功能的系统主页面。</p>

表 3-3 “用户注册”功能测试步骤

3) Cookies 测试

Cookies 是服务器保存用户信息到客户端机器的常用手段，这种方式可以让客户机分担一部分服务器负担从而给用户提供一些个性化的服务功能。当用户第一

次访问涉及 Cookies 的资源时，服务器端将发送一个相关的 Cookie 对象到客户端机器上。本课程网站系统中“用户登录”模块有一个附加功能，可以让用户“两周内不用登录”，这个功能是通过 Cookies 实现的。为了测试 Cookies 是否能正常工作，刻意将 Cookies 的有效期从两周缩短为一分钟。测试时从两方面观察运行结果，一是当用户勾选了“两周内不用登录”选项时，Cookies 是否成功写入客户端，二是当一分钟时间过去时，该 Cookies 是否失效。经过测试，该功能运行正常。表 3-4 反映了登录 Cookies 的测试用例

测试目的	测试 Cookies 是否能够正确创建和失效。
并发用户数	15 个
模拟用户行为	1) 进入系统主界面。 2) 在登录窗口部分输入正确的用户名和密码 3) 将“两周内不用登录”复选框选中。 4) 点击登录按钮。 5) 关闭主界面 6) 再次进入系统主界面
预期结果	当用户再次进入主界面，不再出现登录框，而是显示欢迎“XXX”用户的信息，说明 Cookies 已创建。 一分钟后，当用户再次登录主界面，仍然出现登录界面，说明 Cookies 已失效。

表 3-4 登录 Cookies 测试用例

2. 性能测试

1) 连接速度测试

用户访问 Web 应用系统中的页面速率受各种因素影响。如用户的上网方式、上网带宽、服务器的速率等。有数据显示，如果用户等待一个页面的时间超过 5 秒就会有放弃访问的想法。因此对 Web 应用系统响应速度的测试成为 Web 应用非常重要的一环。本系统在局域网环境下测试了各项目功能的速率，基本正常。受条件限制，未能在公网进行测试。表 3-5 反映增加课程资源页面的响应速度测试用例。

测试目的	测试新增课程资源记录，系统新增资源的响应时间。
并发用户数	15 个

模拟用户行为	1) 进入后台系统管理员登录主界面。 2) 在登录窗口部分输入正确的用户名和密码 3) 登录系统后选择新增课程资源导航条。 4) 进入新增课程资源页面后, 填写待新增的课程资源信息。 5) 点击提交按钮
预期结果	新增信息成功提交 新增信息提交响应时间低于 5 秒

表 3-5 新增课程资源响应速度测试用例

2) 负载和压力测试

为了测量 Web 应用系统在某一时刻应对一定级别负载的能力需要进行负载测试, 从而保证 Web 系统在其需求范围内的负载到来时能够应对。关于本系统所做的测试主要有两方面, 一方面是某个特定时刻同时访问课程网站系统的用户数量, 另一方面是在线数据处理的数量。经过测试, 本课程网站应用系统至少能同时允许 200 个用户同时在线, 超过这个数量的负载, 受条件限制未能实现测试。另外, 本课程网站应用系统可以处理大量用户 (至少 200 个) 对同一个页面的访问请求。

本系统的压力测试受条件限制必须留待系统发布到公网后方能进行。

3. 可用性测试

1) 导航测试

Web 站点众多的网页之间, 或是页面的控制总是通过导航控件来组织, 组织站点时在一个页面上放的信息过多往往会让用户反感。一般的用户都趋向于能够通过导航快速定位自己的目标内容。如果通过导航不能获知自己感兴趣内容的存在很快便会关闭对此站点的浏览。因此, 一个优秀准确的导航系统可以帮助提升网站的可用性, 当然也是网站人气的重要保证之一。

本课程网站涉及到的课程资源多且繁杂, 其中还涉及到知识点的层次结构组织, 为了让用户能够快速掌握整个课程网站的核心内容, 在用户进入课程资源浏览页面后, 使用树型目录是最可靠最容易的导航方式, 且非常直观。同时为了保证树型目录的可用性, 在此功能部分引入了 Ajax 技术。通过 Ajax 技术的引用大大提高了树型目录的加载速度。

表 3-6 反映了此树型目录响应时间的测试用例。

测试目的	测试树型目录的导航准确性以及响应时间。
并发用户数	25 个
模拟用户行为	1) 进入系统主界面。 2) 选择课程资源导航条进入课程资源浏览页面。 3) 在课程资源页面的树型导航控件中选择任意一个节点展开。 4) 双击其中的某个叶子节点, 以在页面上显示相应内容。
预期结果	系统能够成功打开树型目录任一子节点的内容并显示在资源展示页面。 系统打开资源管理页面的时间不能超过 5 秒。(比未应用 Ajax 技术的树型目录加载要快) 系统处理节点展开的时间不能超过 8 秒。(比未应用 Ajax 技术的树型目录节点展开速度要快)

表 3-6 树型目录响应时间测试用例

2) 整体界面测试

整个 Web 应用系统的设计是否合理, 整体风格是否一致, 是否能给用户友好的体验和舒适感。这些都依赖于整体界面设计。我们在课程网站中大量引入了 Ajax 技术来提高用户友好度。这些都对系统的可用性产生了积极的影响。

表 3-7、表 3-8、表 3-9 分别反映了系统中采用了 Ajax 技术的智能查询、水印效果、及时登录功能的测试用例。

测试目的	测试智能查询功能。
并发用户数	4 个
模拟用户行为	1) 进入系统主界面。 2) 在主界面的查询窗口输入关键字 3) 点击“搜索”按钮
预期结果	当用户在查询文本框中输入部分关键字时, 与此部分关键字匹配的所有信息将显示在文本框的下拉列表中。

表 3-7 智能提示功能测试用例

测试目的	测试文本框的水印提示功能。
------	---------------

并发用户数	2 个
模拟用户行为	1) 进入系统主界面。 2) 当用户将焦点置于搜索文本框。
预期结果	当用户未置焦点于搜索文本框时，文本框中将显示水印提示文字，而当该文本框获得焦时时，水印提示文字将自动消失。

表 3-8 水印功能测试用例

测试目的	及时登录功能的测试
并发用户数	2 个
模拟用户行为	1) 进入系统主界面。 2) 选择课程资源导航条进入课程资源浏览页面。 3) 在课程资源页面的树型导航控件中选择任意一个节点展开。 4) 双击其中的某个 ppt 子节点，以在页面上显示相应内容。 5) 点击内容下方的“下载”按钮 6) 在系统弹出的登录提示框中输入用户名和密码并登录。 7) 再次点击页面上的“下载”按钮。
预期结果	用户未登录点击“下载”按钮的情况下，能够及时弹出模态“登录”对话框。 用户通过模态“登录”对话框登录成功后，可以直接回到下载页面进行下载。

表 3-9 及时登录功能测试用例

4. 客户端兼容性测试

1) 平台测试

Windows、Unix、Macintosh、Linux 等是市场上比较常见的操作系统。而 Web 应用系统的终端用户使用的操作系统有可能是众多的操作系统中的任一款产品，这样，就很有可能产生系统兼容性问题。我们在发布该课程网站应用前针对平台兼容性也作了测试。本系统在分别在 Windows 平台和 Linux 平台上进行了兼容性

测试，测试结果证明该课程系统网页在这两个系统基于 1024*768 及以上分辨率的情况下能够正常显示页面，各功能运行正常。

2) 浏览器测试

在 Web 工作模型中浏览器是核心的构件，但目前市场上不同厂商的浏览器对各种技术 (Java、javascript、ActiveX、plug-ins) 的支持均有不同。例如，ActiveX 在 Microsoft 的 Internet Explore 运行一切正常，但在 Netscape 中则未必能够保证。许多页面因为使用了框架和层 (DIV)，他们在不同的浏览器中将会有不同的显示结果，甚至无法显示。另外，不同的浏览器的安全特性也不一样。本课程网站在目前比较主流的两大浏览器 IE6.0 和 FireFox 中均做了测试。测试结果显示网站功能在这两个浏览中均运行正常，页面显示差异不影响终端用户的使用。

3) 安全性测试

因特网与生俱来的安全问题使得 Web 应用系统的安全性测试显得至关重要。为了保证课程网站的安全性能，特针对安全问题设计了若干测试用例，如表 3-10、3-11、3-12 所示。

测试目的	测试需认证页面是否在登录保护之中。
并发用户数	2 个
模拟用户行为	1) 进入系统主界面。 2) 在登录框输入正确的用户名和密码并登录系统。 3) 进入系统的资源页面，并点击某个资源下的“下载”按钮，从而在地址栏获得受保护资源的访问地址。 4) 复制地址栏中的地址，并关闭系统所有页面 5) 再次登录浏览器，粘贴上一步中复制的地址到浏览器的地址栏中。
预期结果	如果用户可以不用登录，直接在地址栏输入地址就访问到必须要受到保护的资源，那么系统的安全性就有问题

表 3-10 权限认证测试用例

测试目的	测试登录超时。
并发用户数	2 个
模拟用户	1) 进入系统主界面。

行为	2) 通过输入正确的用户名和密码登录系统。 3) 然后进入某个资源下载页面。 4) 将页面缩小, 15 分钟内不对系统做任何操作
预期结果	用户登录后在 15 分钟内没有点击任何页面, 需要提示用户登录超时, 必须重新登录。

表 3-11 登录超时测试用例

4.3.4 测试分析总结及说明

通过单元测试和集成测试, 发现了系统中的若干较隐蔽的问题, 但系统中仍然有可能存在一些未被发现的漏洞, 这些都必须通过部署到公网后进行实测才能完全暴露出来。考虑该门课程网站目前只是供校内软件技术专业的学生使用, 工作负荷暂时不会太高, 但考虑到日后随着课程影响力的增加, 有可能会对服务端工作负载能力提出更高要求, 服务端的硬件配置必须达到中上水平, 尤其要考虑内存容量的问题。

本课程网站的资源及其路径信息存储在数据库中, 其中还将包含用户通过终端上传的资料, 一旦丢失, 弥补工作量相当大, 甚至无法完成。所以管理员必须注意定期对数据库做备份和维护工作, 确保系统的高效稳定运行。

测试完成后, 关于系统的整体规划仍然存在一定的问题, 例如, 系统中中文信息提交时偶尔会出现乱码问题, 通过分析发现是因为客户端和数据库服务器使用的字符集不同。为处理此问题, 系统专门添加了字符编码处理模块。另外, 首页的导航菜单为静态内容, 一旦关于该网站的内容需要做出调整, 则需要重新修改、编译代码, 无法通过系统后台完成页面的维护。考虑后期需要对这个部分进行修改。另外, 课程网站的资料直接面向学生和老师, 他们的评价能够直接影响该门课程的内容和教学资料修订工作, 但系统没有为资料设定评价模块。后期升级维护系统时, 也需要考虑这个功能模块的添加。

4.4 本章小结

本章主要围绕 ASP.NET AJAX 3.5, 以及通过它实现 C#精品课程网站改造来进行阐述。其中包括使用 ASP.NET AJAX 框架对精品课程网站的实现, 并给出了具有 Web 应用系统风格的部分测试用例, 最后还总结了测试结果, 并对系统中存

在的问题提出了规划。本章的核心是使用 ASP.NET AJAX 3.5 在 VS 2008 下如何实现对 C#精品课程网站的改造。本章也是整篇论文的工作重点。

第五章 结论与展望

5.1 论文工作总结

基于 Ajax 的 Web 应用程序在如今已经蓬勃发展起来了,无论是国内还是国外使用该技术的门户网、消费网、博客网甚至是企业级的 Web 应用都表现出了强有力的竞争力。因为“用户就是上帝”,而 Ajax 应用所倡导的正是“以用户为中心”。经过“Ajax 化”的网站具有了某些桌面系统的特点,系统的兼容性、应用范围和用户体验都得到极大的改善。不过,尽管该技术在一些大型商业或企业网站上得到了广泛的应用,但在中小企业网站,或是教育部门的网站上却还没有得到普及。类似于国家精品课程的“Ajax 化”问题还很少有人分析和研究,但市场证明各类网站的“Ajax 化”已是 Web 世界的大势所趋,没有应用该技术的网站将逐渐淡出人们的视线。因此将 Ajax 技术应用到国家精品课程网站的研究有很大的理论和实践价值。本文通过对基于 ASP.NET Ajax 的精品课程网站的设计和开发,主要得出以下几个结论:

(1)页面刷新速度加快,系统性能更好。在改造精品课程网站的过程中,我们在凡属有 Ajax 应用价值的功能处都作出了改进,发现通过这种局部刷新的改进后,客户端与服务器端传输的数据冗余确定减少了,因而页面刷新的速度也有所改善,系统整体性能得到了提升。

(2)用户交互度和体验度得到改善,网站更人性化。在无需改变用户操作习惯的前提下完成了用户体验度的改善,以往可能出现白屏、闪烁的功能经过改进后已无此现象,而以往没有提示让用户不知所措的功能现在也得到了改进,使用者明显感受到了网站的交互度提高和人性化设计的优势。

(3)开发和发布都“智能化”,费用合理。与 .NET Framework 和 VS 2008 几乎是无缝配合的 ASP. NET AJAX 免去了开发人员钻研 JavaScript 的大把时间,让开发者专注于业务的设计,更难能可贵的是,这个框架是可以给开发者免费使用,且可以做定制开发的,这都为开发者提供了方便。是支持完全面向对象方式的编程方式,使得系统的开发为方便。对用户而言,应用该技术发布的系统客户端无需任何插件或附加程序便可以获得类似于 WINDOWS 桌面应用程序一样的便捷功能。

(4)精品课程网站设计得到改进, 对使用者的吸引力增强。由于以往受技术所限类似于树形导航控件这种会导致频繁刷新或性能危害的设计很少出现在高校的课程网站中, 但有了 Ajax 技术的局部更新支持, 系统应用此技术性能也得到提升, 许多对服务器要求较高但对用户而言更方便的设计我们也能够出现在课程网站中。无疑给教育平台的发展带来了福音。

5.2 论文主要特色

本文在研究 ASP.NET Ajax 技术应用于国家精品课程网站的设计与开发过程中, 借鉴大量商业和企业网站的研究成果, 并在此基础上结合课程网站的实际情况有所创新, 主要的创新点如下:

(1)将 Ajax 技术应用到了一个目前还没有得到普及的领域, 初步实现了一个基于 Ajax 的精品课程网站, 使得网站性能得到大幅提升, 对用户的吸引力增加。

(2)充分挖掘了 ASP.NET Ajax 3.5 的功能, 研究和分析了它的客户端和服务端开发模型, 在两者之间应用抉择得当, 使得系统开发方便快捷的同时, 又不失其灵活性。

(3)开发过程中, 考虑到表示层与业务层的分离, 引入了 WCF 和微软的 Web 服务, 这一设计, 使得 Ajax 技术常常导致的系统架构不清晰的问题得到了很好的解决。

(4)将微软开发社区许多新开发的控件应用到了精品课程网站, 使得网站在最短的时间内获得了技术改造。

5.3 研究工作展望

应用了 Ajax 技术的 Web 应用系统在最近两年发展势头很猛, 以 Google 为代各路网络巨头均在此技术上投入了较大研发力度, 各类门户及消费型网站也将他的应用推向了寻常百姓的寻常生活。本课题主要研究如何让 Ajax 技术在一个较新的领域发光发热, 研究证明这种尝试是可行的, 但受各种客观条件限制, 仍有一些工作需要进一步完善, 具体如下所示:

(1)受实验条件所限, 改进后的 C#精品课程网站只是在 IE 和 FireFox 下测试过他的所有功能, 该系统与其他浏览器的兼容性问题还只有理论数据。

(2)尽管 Ajax 能够优化系统的性能, 但受他本身技术所限, 也带来了新的性能和安全问题, 这些问题都有待日后应用中去发现和克服。

(3)在 Web 开发标准化问题中, Ajax 技术与网站的可访问性密切相关,但网站可访问性问题在国内的重视不够,因此在这一块的理论研究还有进一步的需要。

(4)除了技术层面的问题,精品课程网站的建设模式何去何从,如何使其发挥最大效能,如何通过开放平台的改进让其“实用化”这些都值得我们开发者们进一步探讨。

放眼未来,在各大软件巨头和社区开源力量的推动下, Ajax 日趋成熟的敏捷开发模式将可以帮助我们实现更智能化更快速开发 Ajax 应用的梦想; Ajax 技术也将朝软件设计工程化迈出坚实的步伐,面向对象的世界必将深入到脚本编程的各个方面,将 Ajax 技术应用于企业级的完整体系架构和实现支撑也将离我们不远。

致 谢

论文即将完成，我谨向所有帮助我的老师、朋友和亲人表示最诚挚的感谢！首先，最要感谢的是我的导师桑楠教授和开题时的指导老师罗克露教授，他们给予了我无私的帮助和指导。其次，感谢在项目开发小组的全体同事和朋友给予我的帮助。

在电子科技大学软件学院老师的帮助下，我即将毕业，其间老师或授业，或指导，或建议我，在此我向本专业的所有老师表示我最崇高的敬意和最深的感谢！感谢我的家人，在我一边学习一边工作其间，给予了生活上、精神上的极大支持、关心和鼓励，使我能够全身心的投入到学习和研究工作中。

参考文献

- [1] <http://www.jingpinke.com/xpe/portal/20a4bb00-1188-1000-8b6c-ac9dee625938> 国家精品课程资源网
- [2] (美) 霍斯拉维 著, 刘志忠 译 ASP.NET AJAX 编程参考手册(涵盖 ASP.NET 3.5 及 2.0). 清华大学出版社, 2009
- [3] 朱先忠, 邵建玉, 温莹洁 编著. 开发者突击: 精通 ASP.NET AJAX 网络程序开发. 电子工业出版社 2008
- [4] <http://edu.drcnet.com.cn/DRCnet.common.web/docview.aspx?version=edu&docid=2264506&leafid=98&chnid=33> 国务院发展研究中心信息网
- [5] 黄硕, 项国雄. 国家精品课程网站建设现状研究. 现代教育技术. 2008 年
- [6] Justin Gehtland, Ben Galbraith, Dion Almaer. Pragmatic Ajax: A Web 2.0 Primer. Pragmatic Bookshelf; 1 edition (April 10, 2006), 978-0976694083
- [7] (美) 赖尔 等著, 苏金国 等译, Head First Ajax(中文版). 中国电力出版社, 2010
- [8] Jesse James Garrett. The Elements of User Experience: User-Centered Design for the Web. Peachpit Press (October 21, 2002), 978-0735712027
- [10] Jesse James Garrett. Ajax: A New Approach to Web Applications. <http://www.adaptivepath.com/ideas/essays/archives/000385.php> (2005)
- [11] (美) Matthew MacDonald, Mario Szpuszta 著. 博思工作室译. ASP.NET 3.5 高级程序设计: 第 2 版. 人民邮电出版社, 2008
- [12] 朱先忠编著. ASP.NET 3.5 前沿技术与实战案例精化. 中国铁道出版社, 2010
- [13] ms-help://MS.VSCC.v90/MS.MSDNQTR.v90.chs/dv_vwdcon/html/407f453c-f490-4713-8748-6b6cff909aa8.htm 微软 MSDN 帮助文档
- [14] 万峰科技编著. JSP 网站开发四酷全书. 电子工业出版社, 2005 年
- [15] David Flanagan 编著, 张铭泽等译. JAVASCRIPT 权威指南. 机械工业出版社, 2003 年
- [16] [美] Ryan Asleson, Nathaniel T. Schutta 著, 金灵等译. AJAX 基础教程. 人民邮电出版社, 2006 年
- [17] Steven Holzner. XML 使用详解. 机械工业出版社. 1999
- [18] Charles F. Goldfarb, Paul Prescod. XML 实用技术第 1 版. 北京: 清华大学出版社, 1999

- [19] 柯自聪, Ajax 开发精要[M].北京:电子工业出版社, 2006
- [20] Charles Babcock.Ajax Advantage InformationWeek. Manhasset Mar 27, 2006:20
- [21] 姚敦红,陈蜀宇,蒋启明. 基于 ASP.NET 的动态导航树的实现[J]. 电脑知识与技术,2008(4):146-148
- [22] 赵吉,AJAX 深入浅出.MSDN 网络广播.2007 年 3 月
- [23] 苏鹏, AJAX 和 ATLAS 开发系列课程. MSDN 网络广播.2006 年 3 月
- [24] 林海浏览器/服务器应用开发.北京:科学出版社, 2000 年
- [25] 陈楠.Web 服务器与数据库服务器接口技术探析.计算机系统应用, 1998
- [26] 魏高山.三层 client 和 server 结构分析与应用.计算机工程与应用, 2000
- [27] 游丽贞, 郭宇春, 李纯喜.Ajax 引擎的原理和应用明.微计算机信息, 2006
- [28] Karl Seguin.ASP.NET Spiced:AJAX.MSND,2005
- [29] 马瑞民,王文红,杨王黎,史国蕊.一个超媒体著作系统中导航树的设计与实现[J]. 大庆石油学院学报,2001(2):40-43.
- [30] <http://www.okajax.com/a/200805/052S0R008.html>,Ajax 中国网站
- [31]http://www.cww.net.cn/tech/html/2007/12/28/200712271354407530_3.htm, 通信世界网
- [32] 赵永屹,宿红毅等.结合AJAX 与J2EE 技术的WEB 信息检索系统的设计与实现[J].微计算机信息.2006,9-3:P149-151.
- [33]王丹,杜晓明,常雷. 基于XML 导航树的交互式电子技术手册[J]. 软件技术,2005(5):113-114.
- [34] (美)Stephen Walther著, 汤涛 译, ASP.NET揭密 (第二版), 中国电力出版社, 2004
- [35]陈志平,许锡山,陈玉教. 一种基于Ajax 的动态树形结构的设计与实现[J].计算机与信息技术,2005(12): 63-66.
- [36] 傅元略, 庄明来.Ajax风格的Web开发.MSDN开发精选.2002.5(2):2
- [37] 戴上平,丁士锋,等.ASP.NET 3.5完全自学手册 [M].机械工业出版社,2009.
- [38]jQueryAPI参考文档中文版翻译 [EB/OL].
<http://jqueryapizh-cn.googlecode.com/svn/trunk/index.html>.
- [39] [美]KLAUS MICHELSEN,c# PRIMER PLUS(中文版)[M].北京:人民邮电出版社,2002.
- [40] 肖汉.基于可重用构件的软件开发模式研究[J].微电子学与计算机,2007(1):178-181.
- [41] 王魁生,王晓波.基于 Ajax 技术的动态导航树.微计算机信息,2010.

基于ASP.NET AJAX技术的国家精品课程网站设计与实现

作者: [邓超](#)
学位授予单位: [电子科技大学](#)

引用本文格式: [邓超](#) [基于ASP.NET AJAX技术的国家精品课程网站设计与实现](#)[学位论文]硕士 2010