

基于 Web 的多媒体 课件页面优化

徐景琦 林初建

(天津南开大学网管中心 300071)

摘要: 基于 Web 的多媒体 CAI 课件是多媒体 CAI 课件与 Web 技术的结合, 它将成为网络教学多媒体课件 CAI 的主要形式。本文介绍了基于 Web 的多媒体 CAI 课件开发过程中对课件页面进行优化的原则和方法。

关键词: 多媒体 CAI 课件 Web

1 引言

传统类型的课件由于投资成本太大(如基于编程为主要的整体式课件), 或者由于升级更新困难(如基于 CD-ROM 的多媒体课件), 已不能满足日益发展的需要。为了解决以上矛盾, 伴随着 Web 的出现和 Web 技术的发展, 基于 Web 的多媒体 CAI 课件应运而生。

作为这种全新类型课件的平台, Web 本身具有统一的协议标准, 良好的跨平台特性和前所未有的广泛覆盖性的优点, 在其上开发出来的软件有着统一的、良好的 GUI (Graphic User Interface) 界面, 能够支持多种媒体文件格式, 可提供高度的交互性, 与其他的应用程序有着方便良好的接口, 特别是本身以超级链接作为组织形式, 因此非常适合作为课件开发平台。另一方面, 基于 Web 的课件对运行的软硬件平台也没有什么很高的特殊要求, 这使得基于 Web 的 CAI 课件的开发维护费用相对低廉。目前种种“所见即所得”的全图形化的开发和维护工具也使得对开发维护人员的技术水平要求相对降低; 由于各网页相对独立, 因此可以十分容易的对课件进行升级更新。在网络教育中, 基于 Web 的课件可以以网页形式在网络中传输, 具有较快

的传输速度, 在目前较低的带宽下具有较好的应用前景。

基于 Web 的多媒体 CAI 课件的以上优点, 选择它作为 CAI 课件开发的基本方案是适宜的。由于目前 Web 技术发展的局限性, 使基于 Web 的课件方案也存在一些不足, 主要是面临网络带宽、网络传输速度、平台兼容性等问题的困扰。因此在开发过程中必须对课件页面进行优化, 使之能够在 Web 上很好的运行。

页面优化的问题虽然涉及到多方面, 但其根本的立足点和目的是以下两个: 页面下载速度; 另一个则是使页面能像设计者所预先设想的那样在用户眼前显示, 在 Web 上这主要需要解决一个兼容性问题。下面就分别从两个目的出发, 讨论对页面进行优化时所依据的原则和实现方法。

2 加快页面下载速度的优化

传统上, 在加快页面速度方面所做的努力, 仅仅局限于减少多媒体元素的使用数量和降低使用质量, 以取得较小的文件, 从而加快页面文件在网络上传输速度以达到加快下载速度的目的。我们认为, 仅依靠这方面的努力是远远不够的。页面下载

速度可分成以下三部分: 下载时间、渲染时间和可视性, 好的设计者应通过页面优化在这三者中找到一个平衡点, 进而产生理想的下载。从用户点击请示到页面总体框面显示仅一眨眼时间, 在页面完全渲染完之前用户便可以从中搜索自己需要的信息。

2.1 加快页面下载的普遍原则

(1) 去除不必要的多媒体效果和内容。可以用普通按钮取代图形动态按钮, 用静态图象取代 GIF 动画, 减少图形化的花哨文字标题, 这些措施至少可以让页面下载速度时间缩短一半。

(2) 用一切手段减少各种多媒体元素的大小。

(3) 优先使用相同的图形或多媒体内容。缺省的 IE 或 Netscape 都会在 RAM 或者硬盘上设置缓存来保存最近用到的图形和文件, 如果浏览器认识相同的文件名, 则它会读缓存, 从而大大提高效率。

(4) 提供低带宽备用内容。如果你的文件格式不能被认识或浏览器关闭了图形开关, 这些备用内容就可派上用场了。

(5) 对不是所有人都有必要或想看到的大型多媒体文件另置一页, 在原来的页面上仅提供一个按钮或低分辨率图形, 边上注明所链接文件的大小和预计下载时间。

Web-Based Page Optimizing in Multimedia Courseware

2.2 对图形的优化处理

(1) 减少图形文件大小。减少文件大小对加快下载速度有着突出的作用。可以通过以下一些方法来减少图形文件的大小:

① 减少图形面积—这可能通过剪切图形或者按比例收缩图形来实现。要注意的是,按比例收缩后有可能在图形上出现一些色斑。

② 注意颜色安排—通过降低色彩深度,使每一象素有较少的位数可使文件变小。而在一行内不要放入过多颜色将使文件进一步压缩,这对GIF文件更加有效。因为GIF仅是颜色的列表,它为每一行设置一个颜色列表。因此,当沿水平方向改变颜色时,应尽量使用更多的相同颜色片段;基于相同的原因,当使用连续色调时,应尽量从顶部向底部延展,而不要在左右方向上延展。

③ 调整图像中的对比水平—大多数图像处理软件提供润色选项,如伽马(Gamma)校正和正和加亮(highlight)/中调(midtone)/阴影(shadow)来改变一个图象内的对比。缩减这些值,通常可以减少文件大小。

④ 抑制抖动—抖动是指用现有调色板中颜色值来接近调色板中没有的颜色,它倾向于增大GIF文件的大小。通过选择通用安全216色调色板和用图形程序选择'No Dither'选项可消除抖动。

(2) 在HTML文件中为图形设置WIDTH和HEIGHT属性。用WIDTH和HEIGHT属性来放大和压缩图形面积并不能加快下载,但它可以使浏览器在下载图片前便能确定正确的页面布局,先把文字和其他后继内容显示出来,使用户在下载图片的同时便可开始浏览。

(3) 随通符号使用LOWSRC属性。这个属性允许我们指定另一个图形文件,它将首先下载,并在SRC属性指定的图形显示之前显示出来,一般由它指定一个低分辨率、低色度的与SRC属性指定的图形大小相等的图形,这样不能使SRC指定的图形下载加快,但能使用户在没有看到SRC图形之前对该图形有个初步的了解。

(4) 使用交错式GIF和渐进式JPEG,交错式GIF和渐进式JPEG是对原来图形标准的扩展。其中交错式GIF向后兼容而渐进式JPEG不能,当下载后在页面上显示时效果就如同它是从一套白页窗后显露出来一样。尽管他们不能加快下载速度,但用户的感受却是加快了。

2.3 对表格<TABLE>的优化处理

表格是目前网页中用得最多的布局元素,因此,对表格的优化就显得特别有价值,虽然这一部分优化并不能减少下载时间,但由于浏览器需要在填充表格内容前完全理解表格的结构,所以适当的

表格运用将大大减少网页的渲染时间。在使用表格时应注意做到:

(1) 将一个单一的大表格拆成多个小表格,这将加快页面显示速度,尤其当页面很长而用表格进行整体布局时更是如此。

(2) 对<TABLE>、<TD>适当使用WIDTH属性,在这一点上应注意单元格内的内容的宽度,你定义的单元格宽度至少应与内容宽度一致,以避免浏览器的过滤作用对其进行两次渲染。

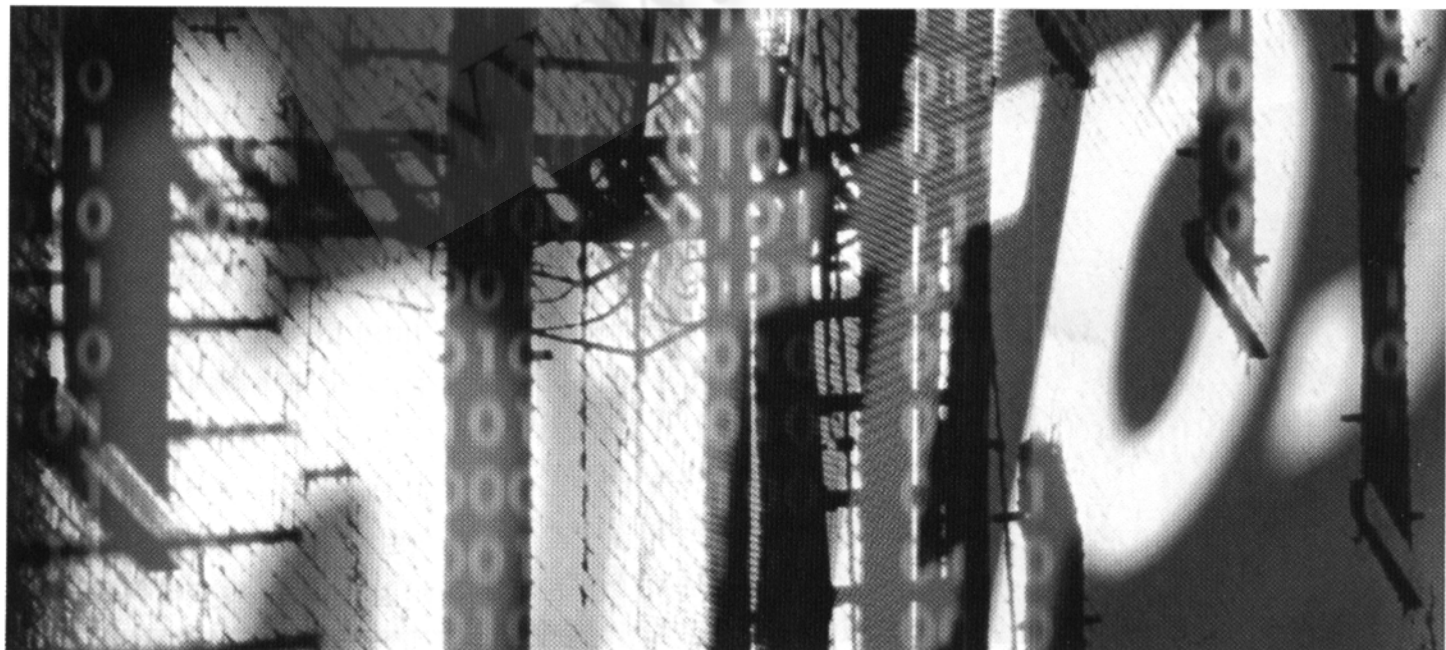
(3) 千万不要嵌套表格,这将使表格处理时间以倍速上升,但如果一定要嵌套,至少应减少嵌套表格保持简单。

(4) 不要将太新的扩展多媒体元素放入表格,这将使某些低版本的浏览器迅速崩溃。

2.4 对多媒体元素的优化处理

(1) 对动画部分的优化。对Direct和Authorware动画应尽量压缩帧数,并使用流式传输处理,对一般网页上常见的GIF动画,FLASH动画或者JavaScript动画应尽量减少,因为他们一般除装饰效果外,很难有什么其他作用,却会使页面尺寸急剧增加。

(2) 对背景音乐的优化。优先选用midi格式,采样仅需8位,11kHz即可。如果需要语音朗读的话,尽可能使用男生,这可以采用更低的设置。尽量将



Web-Based Page Optimizing in Multimedia Courseware

声音做成可选设置, 优先选用流式声音传输格式(如 mp3 格式)。

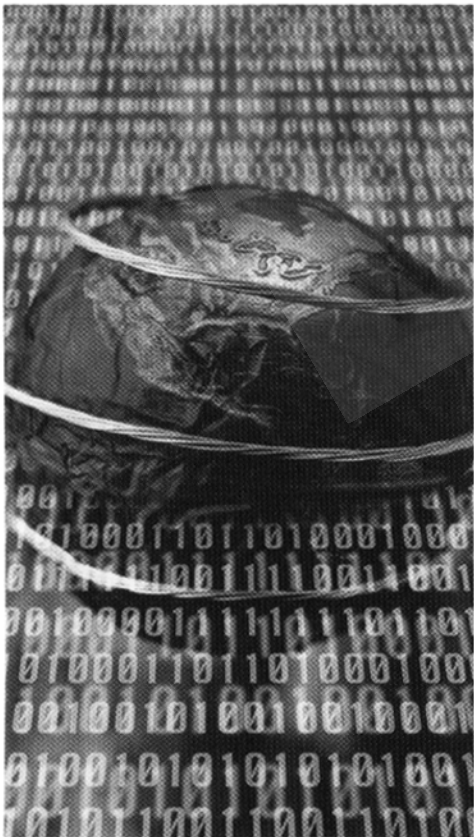
(3) 对脚本程序的优化。许多网页制作者认为页面中的JavaScript等脚本语言不会使页面变慢, 然而大多数的脚本语言都是解释语言, 而非编译语言, 这就意味着他们装载后还需被分析, 这也构成了页面渲染时间的一部分, 在使用脚本语言时, 就注意做到:

① 尽量将脚本嵌入页面中, 而不使用 SRC 属性, 因为这将引起对服务器的另一次请求, 既加重了服务器负荷, 又延长了总的下载时间。

② 如果用脚本控制 Plug-in 和 DHTML, 尽量使用一个内置的转载和运行更快的函数来代替自己编写的函数。

2.5 适当使用<LAYER>/<DIV>和<FRAME>定位

由于<Table>不能在页面上实现精确定位, 所以一些页面的效果将无法实现, 而适当的使用<LAYER>/<DIV>能够弥补这个缺陷, 不足的是<LAYER>/<DIV>有些低版本的浏览器支持得不是很好, <FRAME>可以把页面分成多个子页面, 而在同一窗口中可以显示多个 HTML 文件。



2.6 适当利用现成的扩展插件实现特殊的功能和效果

目前最热门的网页制作软件 Macromedia Dreamweaver 支持外挂插件的功能, 使制作各种各样的多功能和多媒体效果的页面成为可能。

3 对兼容性的优化

在兼容问题上, 既要考虑课件对各种操作系统和硬件平台的兼容性, 更要考虑课件对各种浏览器的兼容性。

3.1 在平台兼容性上的优化

当课件在 INTERNET 上发布时, 无法设想你的用户会在怎样配置的机器上使用怎样的操作系统, 好在 Web 本身的跨平台特性已经很好的解决了这个问题, 使我们免于多次开发, 但为了使课件能够在各种平台上正常显示, 仍需要我们在定义潜在用户群时考虑许多细节, 包括: 显示器的分辨率和色深, 对音频和视频的支持情况, 各种平台上相同字体的差异, 各种平台上显示调色板之间的差别等。

3.2 在浏览器兼容性上的优化

与浏览器不兼容带来的影响相比, 由于平台不兼容对课件开发所带来的影响几乎可以忽略不计了, 各大主流浏览器开发厂商或者由于商业上的考虑而故意制造不兼容情况, 如 IE 和 Netscape; 或者为了追踪技术发展不得不造成新版本某些部分事实上不向后兼容, 如 Netscape Communicator 4.5 与以前版本对 JavaScript 的处理; 再加上其他各种非主流浏览器, 如何消除浏览器兼容性对课件开发的影响, 是课件开发者所真正关心的问题。

考虑到 IE 和 Netscape 加起来占了浏览器使用总数的 90% 以上, 而国内用户几乎 100% 使用 IE, 所以目前的开发以 IE5.0 版为主, 同时兼顾 Netscape5.0 以上版本; 对某些重要的功能, 如果两者需用不兼容技术实现, 则分别开发两个版本, 由服务器对用户浏览器进行判断后分别转向到相应的页面, 在脚本语言方面, 由于 JavaScript

是唯一大多数主流浏览器支持的语言, 所以成为唯一选择。

4 发展方向的一些探讨

从功能上说, 根据学生学习课件后取得的效果, 可把课件分为“知识技能的获取型”, “应用知识分析和解决问题”型, “创造性的解决问题”型三种; 而从目前课件开发情况看, 以第一类居多, 第二类较少, 第三类几乎还是空白, 我们认为, 在这方面应加强研究的力度, 加快发展步伐。

从内容组织上来说, 目前课件基本上都是通过链接的方式组织起来, 学生可以根据自己的学习进度进行学习, 交叉索引也比较方便, 这与以前固定速度的教师授课或者印刷课本相比, 是个较大的进步; 但是课件的各部分内容及组织结构仍是预先固定好的, 弹性较小, 理想的课件应根据学生的知识结构, 学习类型和学习情况动态后生成学习内容及组织机构, 而这一方面, 还有待于在课件中提高智能化程度来解决。

基于 Web 的多媒体 CAI 课件是未来多媒体课件的主要形式, 也是将来发展网上教学的一个资源基础, 今后加紧对这种课件的研究开发, 才能更快的形成此类课件的开发模型, 开发的技术和标准方面的统一, 也才能使网上教学早日进入实用领域。 ■

参考文献

- 1 【HTML 参考大全(第三版)】, 清华大学出版社, [美] Thomas A.Powell 著, 杨正华等译。
- 2 【HTML JavaScript 与 Java 完全实战演练】, 电子工业出版社, 曹建著。
- 3 【HTML 基础与实例】, 电子工业出版社, [美] Elizabeth Castro 著。
- 4 【HTML 4 快速充电】, 北京大学出版社, [美] Elizabeth Castro 著。