

WEB 平台下的基于 HTML5 标准离线应用开发

蹇红梅

(四川理工学院计算机学院,四川 自贡 643000)

摘 要:HTML5 是由 W3C 发布的新一代 HTML 标准,不仅强化了 Web 网页的表现性能,还拥有很多 API 接口,开发者利用这些接口,可以便捷地创建更加丰富、引人入胜的应用程序。在分析研究 HTML5 标准基础上,利用其技术实现了一个离线便签管理程序系统,既直观展示了 HTML5 的内容描述标签、直接支持表单验证、视频音频标签、网页元素的拖拽、工作线程等新特性,又介绍了利用 DOM Storage 和 Web SQL Database 两种存储机制进行离线应用开发的方法。文章对基于 HTML5 的应用开发具有一定的参考价值。

关键词:WEB;HTML5;离线;应用开发;DOM Storage;Web SQL Database

中图分类号:TP393

文献标志码:A

Web 应用程序,即是基于 B/S 模式或网页浏览器的应用程序,它从运行机制上解决了传统 C/S 模式的客户端安装和升级的不便。随着 Web 新技术的迅速发展,B/S 模式程序在交互体验上,也基本能达到传统桌面应用的要求,因此,现在的 Web 程序已经不再只是信息的浏览^[1]。传统的 B/S 应用高度依赖于网络的畅通,如果网速较慢,B/S 系统的应用效果就会大打折扣,这对于 Web 技术应用,是一个巨大的障碍和挑战,随着 HTML5 的推出,使得这个障碍有了解决方案。

HTML5 标准具有“离线 Web 应用程序”与“客户端数据存储”两个特性,利用这两个特性,用户可在客户端设备上使用云功能,使用部署在本地数据库上的应用程序来工作,当再次上线时与云的其他部分共享数据^[2]。

1 HTML5 离线功能简介

在开发支持离线的 Web 应用程序时,应注意以下三个方面的功能:

(1) 离线资源缓存:需要一种方式来指明应用程序离线工作时所需的资源文件。这样,浏览器才能在在线状态时,把这些文件缓存到本地。当用户离线后再访问

应用程序时,这些资源文件会自动加载,从而让用户正常使用。为了能够让用户在离线状态下继续访问 Web 应用,开发者需要提供一个 cache manifest 文件。这个文件列出了所有需要在离线状态下使用的资源,浏览器会把这些资源缓存到本地。

(2) 在线状态检测:开发者需要知道浏览器是否在线,这样才能针对在线或离线的状态,作出相应的处理。在 HTML5 中,通过 cache manifest 缓存资源文件以后,就可以支持离线访问了。但是在开发支持离线的 Web 应用时,不能仅仅满足于静态页面的展现,还必需考虑如何让用户在离线状态下也可以操作数据。离线状态时,把数据存储在本地的;在线以后,再把数据同步到服务器上。HTML5 提供了两种检测是否在线的方式:navigator.online 和 online/offline 事件^[2-3]。

● navigator.online,该属性表示当前是否在线。如为 true,表示在线;如果为 false,表示离线。

● online/offline 事件:当开发离线应用时,通过 navigator.online 获取网络状态通常是不够的。开发者还需要在网络状态发生变化时立刻得到通知,因此 HTML5 还提供了 online/offline 事件。当在线/离线状态切换时,online/

offline 事件将触发在 body 元素上,并且沿着 document、body、document 和 window 的顺序冒泡。因此,开发者可以通过监听它们的 online/offline 事件来获悉网络状态。

(3)本地数据存储:离线时,需要能够把数据存储到本地,以便在线时同步到服务器上。为了满足不同的存储需求,HTML5 提供了 DOM Storage 和 Web SQL Database 两种存储机制。前者提供了易用的 key/value 对存储方式,而后者提供了基本的关系数据库存储功能。在开发支持离线功能的 Web 应用时,开发者需要在本地存储数据。当前浏览器支持的 cookie 虽然也可以用来存储数据,但是 cookie 长度非常小(通常几 KB),而且功能有限。因此,HTML5 中新引入了 DOM Storage 机制,用于存储 key/value 对,它的设计目标是提供大规模、安全且易用的存储功能^[4-6]。

2 离线应用程序示例

2.1 程序描述

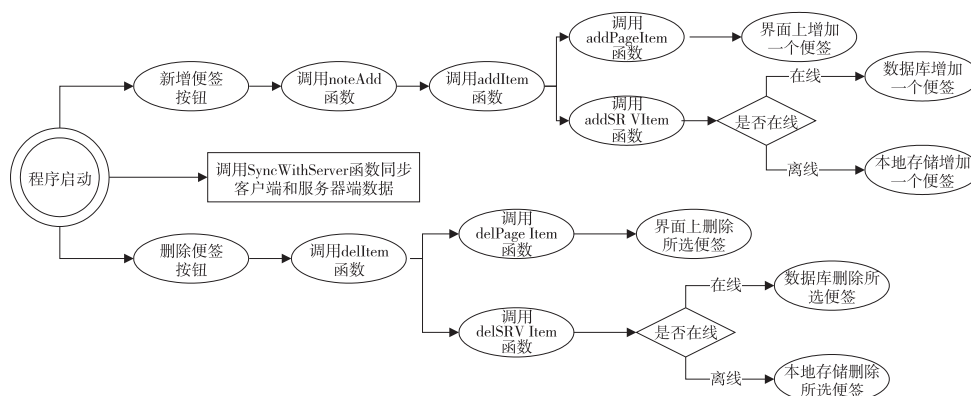


图2 程序处理流程

2.1.3 程序结构

系统设计中含三大主要模块,表1列出了主要的程序文件及功能说明。

表1 系统主要的程序文件及功能说明

程序模块	文件名	功能说明
界面处理模块	index.html	应用程序主界面,用户点击“新增便签”按钮可以在弹出框中创建新的便签,点击每个便签后面的“删除便签”,就表示删除当前所在的便签
	notepad.manifest	定义 cache manifest 文件,声明需要缓存的资源
	padJS.js	包含添加便签和删除便签所对应的 JavaScript 脚本
数据处理模块	padDataProc.js	代码包含添加便签、删除便签和与服务器同步等数据操作
服务器处理模块	padSrvDo.js	客户端和服务器端连接处理的代码

2.1.1 示例应用程序说明

文章通过一个实例来说明使用 HTML5 开发离线应用的基本方法。这个例子会用到前面提到的离线资源缓存、在线状态检测和 DOM Storage 等功能。我们开发的是一个便签管理的 Web 应用程序,允许用户在离线状态下添加、删除便签。它支持离线功能,当用户在线以后能够同步到服务器上。

程序运行的首界面效果如图1所示。用户点击“新增便签”按钮可以在弹出框中创建新的便签,点击每个便签后面的“删除便签”,就表示删除当前所在的便签。

便签管理程序

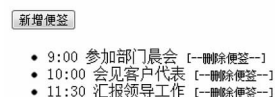


图1 首页效果图

2.1.2 示例应用程序流程图

程序应用处理流程如图2所示。

2.2 界面处理模块

2.2.1 index.html

```

<html manifest = "notepad.manifest" >
<head >
<script type = "text/javascript" src = "padSrvDo.js"
> </script >
<script type = "text/javascript" src = "padDataProc.js" > </script >
<script type = "text/javascript" src = "padJS.js" >
</script >
<title > 便笺离线应用开发 </title >
</head >
<body onload = "SyncWithServer()" >
<header >

```

```

<h1> 便签管理程序 </h1>
</header>
<input type = "button" value = "新增便签" onclick
= "noteAdd()" >
<ul id = "list" > </ul>
</body>
</html>

```

2.2.2 padJS.js

```

function noteAdd()
{
    var title = window.prompt("请输入便签:");
    .....
}

function addItem( title)
{
    addPageItem(title); // 在客户端界面中添加
    addSRVItem(title); // 在后台数据库中添加
}

function delItem( title)
{
    delPageItem(title); // 从客户端界面中删除
    delSRVItem(title); // 从后台数据库中删除
}

.....

```

2.2.3 notepad.manifest

```

CACHE MANIFEST
index.html
padSrvDo.js
padDataProc.js
padJS.js

```

【代码说明】:该文件声明需要缓存“index.html”、“padSrvDo.js”、“padDataProc.js”和“padJS.js”等 4 个文件。除了前面列出的“index.html”外,“padSrvDo.js”、“padDataProc.js”和“padJS.js”分别包含服务器相关、数据存储和用户界面代码^[7-8]。

添加便签流程,如图 3 所示。

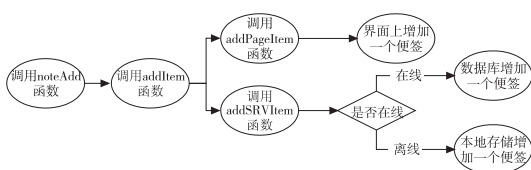


图 3 便签添加流程图

特别指出,addPageItem 函数在页面列表中添加一

项。并且后面增加一个功能,点击调用 onclick 事件的处理函数 delItem,可以删除便签。

便笺添加相关描述,如图 4 所示。

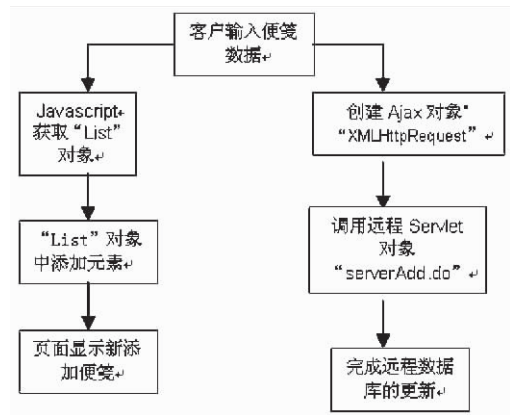


图 4 便笺添加处理细节

通过对程序的运行,系统将得到如图 5 所示效果图。



图 5 便笺添加效果

便笺删除流程,如图 6 所示。

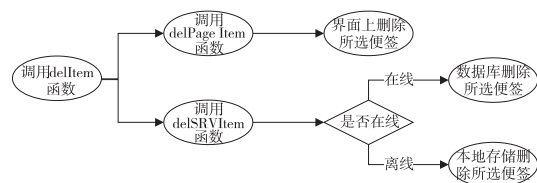


图 6 便笺删除流程

图 7 对删除便笺进行了相关描述。

执行程序,系统运行效果如图 8 所示。

3 结束语

通过系统便笺的添加和删除操作,描述了 HTML5 创建离线应用的步骤,当然该系统仅仅是一个演示程序,应用到实际环境中,还要经过周密细致的需求分析和切合需求的程序结构设计,以期最大程度的利用 HTML5 的特性为需求实现服务。

HTML5 的离线应用的还有很多,比如,阅读和撰写 Email、编辑和显示演示文档 PPT、创建代办事宜列

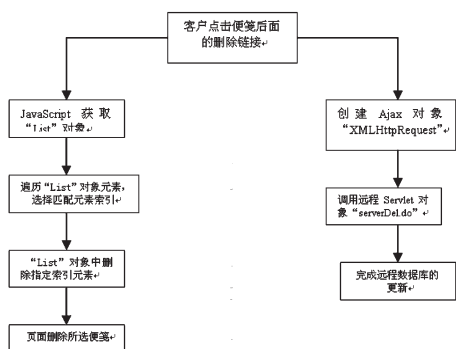


图7 便签删除处理细节

便签管理程序

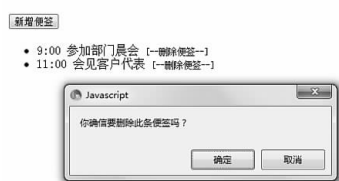


图8 便签删除效果细节

表,等。

使用离线存储,避免了加载应用程序时所需的常规网络请求,如果缓存清单中的文件是最新的,浏览器就知道自己无需再继续检查,这样,应用程序可以非常迅速

的从本地应用缓存中加载完成,节省带宽资源,尤其对于移动 WEB 应用来讲,这点非常重要。

参考文献:

- [1] 梁兴建,周刚,张泽泉.Web 平台下动态统计图表的两种实现方案比较研究[J].四川理工学院学报:自然科学版,2012,25(4):55-58.
- [2] 刘斌.HTML5-未来网络应用的核心技术研究[J].自动化与仪器仪表,2010(4):30-33.
- [3] 马新强,孙兆,袁哲,等.Web 标准与 HTML5 的核心技术研究[J].重庆文理学院学报:自然科学版,2010,29(6):61-64.
- [4] 刘天寅.HTML5 与未来的 WEB 应用平台[J].阴山学刊:自然科学,2010,24(2):86-90.
- [5] 欧少闽,龚明龙,朱凌枫,等.基于 HTML5 的 EX-DRM Web 离线应用系统[J].汕头大学学报:自然科学版,2011,26(4):58-65.
- [6] 何海东,张文秋.基于 Web 的网络硬盘的设计与实现[J].四川理工学院学报:自然科学版,2010,23(2):175-177.
- [7] Mark Pilgrim.HTML5:Up and Running[M].北京:电子工业出版社,2010.
- [8] 柳伯斯,阿伯斯,萨利姆.HTML5 高级程序设计[M].北京:人民邮电出版社,2011.

Offline Application Development Based on the HTML5 Standard of WEB Platform

JIAN Hong-mei

(School of Computer Science, Sichuan University of Science & Engineering, Zigong 643000, China)

Abstract: HTML5, the new HTML standard issued by W3C, not only strengthens the performance of the Web page but also has many API interfaces, which can easily create more application programs. Based on the analysis of the HTML5, technology is used to realize the offline signature management system, directly supporting form verification, audio and video label, webpage element drag, work thread and other new features. The offline application development based on the two storage mechanisms of DOM Storage and Web SQL Database is introduced and explored, and the HTML5 application development discussed in this paper has a certain reference value.

Key words: WEB;HTML5; offline; application development; DOM Storage; Web SQL Database