

Web前端开发技术与实践

第16章：AJAX

阮晓龙

13938213680 / rxl@hactcm.edu.cn
<http://web.book.51xueweb.cn>

河南中医学院管理信息工程学科
河南中医学院网络信息中心

2015.9

本章主要内容

- 概述
- 基础知识
- 使用JavaScript实现AJAX
- 使用jQuery实现AJAX
- 案例：实时表单验证



1. 概述

1.1 什么是AJAX

- AJAX的全称为（Asynchronous JavaScript and XML，异步JavaScript和XML），从AJAX的组合名称可以看出AJAX其实并不是一种技术，而是多种技术的组合体，每种技术都有其独特之处，合在一起就成了功能强大的技术。AJAX的出现揭开了无刷新更新页面的新时代，并且有代替传统Web请求方式和通过隐藏的框架来进行异步提交的趋势，是Web开发应用的一个里程碑。

1. 概述

1.2为什么要使用AJAX

- Ajax采用了异步交互的方式。它在用户和服务器之间引入了一个中间媒介，从而改变了同步交互过程中“处理－等待－处理－等待”的模式。用户的浏览器在执行任务时也同时装载了AJAX引擎。该引擎是用JavaScript语言编写的，通常位于一个隐藏的框架中，负责转发用户界面和服务器的交互。AJAX引擎允许用户和应用系统之间的交互以异步的方式进行，独立于用户与Web服务器之间的交互。现在，可以使用JavaScript调用AJAX引擎向Web服务器发送请求，产生一个HTTP的用户请求、数据编辑、页面导航和数据验证等操作，而不再需要重新加载整个页面。

1. 概述

1.3AJAX优势

- 基于标准化且被广泛支持的技术
 - AJAX不需要任何浏览器插件，就可以被绝大多数主流浏览器支持，用户只需要允许JavaScript在浏览器上执行即可。
- 减轻服务器和带宽负担
 - 按需取数，可以最大程度的减少冗余请求和响应对服务造成的负担。
 - 可以把一些服务器负担的工作转嫁到客户端，利用客户端闲置的能力来处理，减轻服务器的负担，充分利用带宽资源，节约空间和宽带租用成本。

1. 概述

1.3AJAX优势

- 基于标准化且被广泛支持的技术
 - AJAX不需要任何浏览器插件，就可以被绝大多数主流浏览器支持，用户只需要允许JavaScript在浏览器上执行即可。
- 减轻服务器和带宽负担
 - 按需取数，可以最大程度的减少冗余请求和响应对服务造成的负担。
 - 可以把一些服务器负担的工作转嫁到客户端，利用客户端闲置的能力来处理，减轻服务器的负担，充分利用带宽资源，节约空间和宽带租用成本。

1. 概述

1.3AJAX优势

- 无需刷新页面，减少用户心理和实际的等待时间。
 - 特别是在读取大量数据时，不会像刷新页面那样出现白屏的情况，AJAX使用XMLHttpRequest对象发送请求并且得到服务器响应，在不重新载入整个页面的情况下，用JavaScript操作DOM更新页面。因此在读取数据的过程中，用户所面对的不是白屏，是原来的页面内容，只有在数据接收完毕之后才更新相应部分的内容。
- 带来更好的用户体验。
- 可以调用外部数据。
- 进一步促进页面呈现和数据的分离。

1. 概述

1.4AJAX应用场景

- 表单驱动的交互
 - 传统的表单提交，在文本框输入内容后，点击按钮，后台处理完毕后，页面刷新，再回头检查是否刷新结果正确。使用AJAX，在点击提交按钮后，立刻进行异步处理，并在页面上快速显示了更新后的结果，这样提交不会导致整个页面刷新。
- 深层次的树的导航
 - 深层次的级联菜单（树）的遍历是一项非常复杂的任务，使用JavaScript来控制显示逻辑，使用AJAX延迟加载更深层次的数据可以有效的减轻服务器的负担。

1. 概述

1.4AJAX应用场景

□ 用户间的交流响应

- 在多人参与的交流讨论的场景下，最不愿发生的事情就是让用户一遍又一遍刷新页面以便知道是否有新的讨论出现。新的回复应该以最快的速度显示出来，而把用户从分神不断的刷新中解脱出来，使用Ajax是最好的选择。

□ 类似投票等场景

- 对于投票等类似的场景中，如果提交过程需要达到40秒，很多的用户就会直接忽略过去而不会参与，使用AJAX可以把时间控制在很短的时间内，从而吸引更多的用户加入进来。

1. 概述

1.4AJAX应用场景

□ 过滤场景

- 对数据使用过滤器，按照时间排序、按照时间和名称排序、开关过滤器等。任何要求具备很高交互性数据操作的场合都应该用JavaScript，而不是使用一系列服务器请求来完成。在每次数据更新后，再对其进行查找和处理需要耗费较多的时间，而AJAX可以加速这个过程。

□ 文本输入场景

- 在文本框等输入表单中给予输入提示或者自动完成，可以有效的改善用户体验，尤其是那些自动完成的数据可能来自于服务器端的场合，使用AJAX是很好的选择。

2. 基础知识

2.1XML

- XML (Extensible Markup Language, 可扩展标记语言), 标准通用标记语言的子集, 是一种用于标记电子文件使其具有结构性的标记语言。XML是一种元标记语言, 即定义了用于定义其它特定领域有关语义的、结构化的标记语言, 这些标记语言将文档分成许多部件并对这些部件加以标识。
- XML文档定义方式有: 文档类型定义 (DTD) 和XML Schema。DTD定义了文档的整体结构以及文档的语法, 应用广泛并有丰富工具支持。XML Schema用于定义管理信息等更强大、更丰富的特征。

2. 基础知识

2.1XML

- 在XML中，采用了如下的语法：
 - 任何的起始标签都必须有一个结束标签。
 - 可以采用另一种简化语法，可以在一个标签中同时表示起始和结束标签。这种语法是在大于符号之前紧跟一个斜线 (/)，例如<Web前端开发与amp;实践 />，XML解析器会将其翻译成< Web前端开发与amp;实践></ Web前端开发与amp;实践>。
 - 标签必须按合适的顺序进行嵌套，所有结束标签必须按顺序匹配起始标签。
 - 所有的属性都必须有值。
 - 所有的属性都必须在值的周围加上双引号。

2. 基础知识

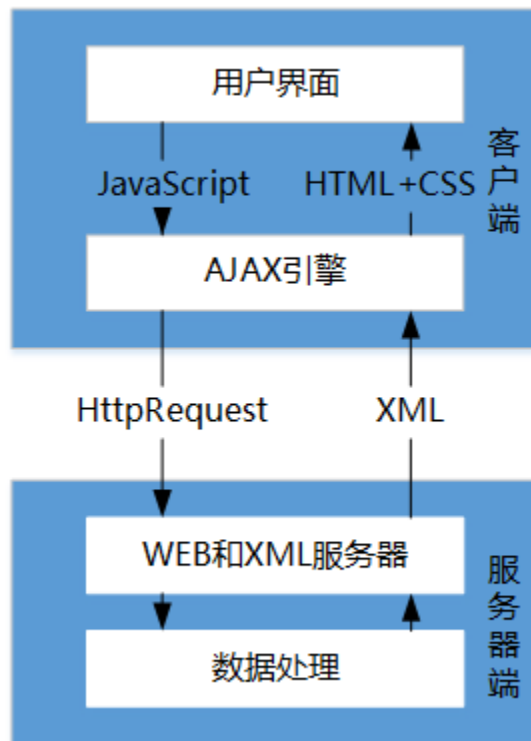
2.2xmlHttpRequest

- xmlHttpRequest, 可扩展超文本传输请求, xmlHttpRequest对象可以在不向服务器提交整个页面的情况下, 实现局部更新网页。xmlHttpRequest对象提供了对HTTP协议的完全的访问, 包括做出GET、POST和HEAD请求的能力。xmlHttpRequest可以同步或异步返回 Web 服务器的响应, 并且能以文本或者一个 DOM 文档形式返回内容。xmlHttpRequest并不限于和 XML 文档一起使用, 可以接收任何形式的文本文档。

2. 基础知识

2.3 工作原理

- Ajax的原理简单来说通过xmlHttpRequest对象来向服务器发异步请求，相当于在用户和服务器之间加了一个中间层，使用户操作和服务器响应异步化。并不是所有的用户请求都提交给服务器，像一些数据验证和数据处理等都交给Ajax引擎自己来做，只有确定需要从服务器读取新数据时再由Ajax引擎代为向服务器提交请求。使用AJAX可以把以前服务器承担的部分工作转移到客户端，利用客户端限制的处理能力来处理，从而减轻服务器的带宽负担，达到节约ISP空间及带宽租用成本的目的。





现场演示:

- 案例16-01: 基于本地XML实现学生成绩册



现场演示:

- 案例16-02: Ajax获取远程数据

3.使用jQuery实现AJAX

4.1基本方法

□ load() 方法

■ 载入HTML文档

- load() 方法是jQuery中最为简单和常用的AJAX方法，能载入远程HTML代码并插入DOM中。它的结构为：
- load(url [, data] [, complete])

表 16-4-1 load()方法参数解释

参数名称	参数选择	参数类型	参数说明
url	必须	String	请求 HTML 的 URL 地址
data	可选	Object	发送至服务器的 key/value 数据
complete	可选	function	请求完成时的回调函数名称



现场演示:

- 案例16-03: 载入HTML文档

3.使用jQuery实现AJAX

4.1基本方法

- 筛选载入HTML文档
 - 在实例16-03中，把16-03-01.html中所有的内容都加载到了16-03.html中id属性值为“ajaxContent”的元素中。通过为load()方法的URL参数来达到从HTML文档里筛选内容的目的。
 - 将16-03.html中，起作用的jQuery代码改为如下所示：

表 16-4-1 load()方法参数解释

参数名称	参数选择	参数类型	参数说明
url	必须	String	请求 HTML 的 URL 地址
data	可选	Object	发送至服务器的 key/value 数据
complete	可选	function	请求完成时的回调函数名称

3.使用jQuery实现AJAX

4.1基本方法

- 传递方式
 - load() 方法的传递方式根据data来自动指定。如果没有参数传递，采用GET方式传递；如果指定了data数据，则传递方式会自动转换为POST方式。

3.使用jQuery实现AJAX

4.1基本方法

■ 回调参数

- 对于必须在加载完成后才能继续的操作，load()方法提供了回调函数。
- 可选的 complete参数规定当load()方法完成后所要允许的回调函数。回调函数可以设置不同的参数：responseTxt，包含调用成功时的结果内容；statusTXT，包含调用的状态，xhr，包含XMLHttpRequest对象。

3.使用jQuery实现AJAX

4.1基本方法

□ \$.get()方法

- \$.get()使用GET方法进行异步请求。它的结构为：
- \$.get(url[, data][, callback][, type])

表 16-4-2 \$.get()方法参数解释

参数名称	参数选择	参数类型	参数说明
url	必须	String	请求 HTML 的 URL 地址
data	可选	Object	发送至服务器的 key/value 数据，作为 QueryString 附加到请求 URL 中
callback	可选	function	载入成功时回调函数(只有当 Response 的返回状态是 success 才调用该方法)自动将请求结果和状态传递给该方法
type	可选	String	服务器端返回的内容格式，包括 xml、html、script、json、text 和 _default

3.使用jQuery实现AJAX

4.1基本方法

- 使用参数
 - 确定请求页面的URL地址。
- 数据格式
 - 服务器返回的数据格式可以有很多中，但都可以完成同样的任务，下面是集中返回数据格式的对比应用。
 - HTML片段
 - 由于服务器端放回的数据格式是HTML片段，因此并需要经过处理就可以将新的HTML数据插入到主页面中。
 - XML文档
 - 由于服务器端返回的数据格式是XML文档，因此需要对返回的数据进行处理，JQuery对DOM有强大的处理能力，处理XML文档可以与处理HTML文档一样，也可以使用常规的attr()，find()，filter()及其它方法。

3.使用jQuery实现AJAX

4.1基本方法

- JSON文件
 - 出现JSON文件很大程度上是因为XML文档体积大和难以解析，JSON文件和XML文档一样，很方便被重用。JSON文件非常简洁，并且很容易阅读。服务器端返回的数据格式是JSON文件，需要对返回的数据处理后才可以将新的HTML数据添加到主页面。
- text
 - 返回纯文本字符串。



现场演示:

- 案例16-04: 使用\$.get()方法进行数据验证, 返回不同的数据格式

3.使用jQuery实现AJAX

4.1基本方法

□ \$. post () 方法

- \$. post () 方法与\$. get () 方法的结构和使用方式都相同，但是在具体使用时有些区别。
- GET请求会将参数跟在URL后进行传递，POST请求则是作为HTTP消息的实体内容发送给Web服务器。在AJAX请求中，这种区别对于用户不可见。
- GET方式对传输的数据大小有限制（通常不能大于2KB），而使用POST方式传递的数据量要比GET大（理论上不受限制）。



现场演示:

- 案例16-05: 使用\$.post()方法验证用户名、密码

3.使用jQuery实现AJAX

4.1基本方法

- GET方式请求的数据会被浏览器缓存起来，其它用户可以从浏览器的历史记录中读取到这些数据，例如帐号和密码等。在个别情况下，GET方式会带来严重的安全问题，使用POST方式传递数据可以避免这一现象。
- GET方式和POST方式传递的数据在服务器端获取的方式有所不同。使用PHP获取GET的数据可以使用`$_GET[]`获取，POST方式则需要使用`$_POST[]`获取。用GET或者POST都可以使用`$_REQUEST[]`来获取。

3.使用jQuery实现AJAX

4.1基本方法

□ \$.getScript()方法

- 在页面加载时就全部加载全部的JavaScript文件是不必要的。在需要那个JavaScript文件时可以动态创建<script>标签，但这种方法并不理想。为此jQuery提供了\$.getScript()方法来直接加载JS文件，与加载HTML片段一样简单方便，并且也不要对JavaScript文件进行处理，JavaScript文件会自动执行。
- 点击按钮加载JavaScript，jQuery代码如下所示：
- `$(“#button”).click(function($.getScript(‘***.js’)););`

3.使用jQuery实现AJAX

4.1基本方法

- \$.getJSON() 方法
 - \$.getJSON() 方法用于加载JSON文件，与\$.getScript() 方法用法相同。

3.使用jQuery实现AJAX

4.1基本方法

□ \$.ajax()方法

- 使用上述的load()、\$.get()、\$.post()方法可以完成一些常规的AJAX程序，如果还需要编写一些复杂的AJAX程序，那么就要用到jQuery中的\$.ajax()方法。\$.ajax()方法不仅能实现与load()、\$.get()、\$.post()方法同样的功能，还可以设定beforeSend（提交前回调函数）、error（请求失败后处理）、success（请求成功后处理）以及complete（请求完成后处理）回调函数，通过这些回调函数，可以给用户更多的AJAX提示信息。还有些参数可以设置AJAX请求的超时时间或者页面的最后更改状态等。
- \$.ajax()方法是jQuery最底层的AJAX实现。它的结构为：
- \$.ajax(options)

3.使用jQuery实现AJAX

4.2jQuery中的全局事件

- JQuery简化AJAX操作不仅体现在调用AJAX方法和处理响应方面，而且还体现在对调用AJAX方法的过程中的HTTP请求的控制。通过JQuery提供的一些自定义全局函数，能够为各种与AJAX相关的事件注册回调函数。
- 如当AJAX请求开始时，会触发ajaxStart()方法的回调函数；当AJAX请求结束时，会触发ajaxStop()方法的回调函数。这些方法都是全局的方法，因此无论创建它们的代码位于何处，只要有AJAX请求发生时，就会触发它们。在请求内容数据很慢的情况下，如果在加载过程中，不给用户提供一些提示和反馈信息，很容易让用户认为按钮点击效果无用，使用户对网站失去访问的信心。

表 16-4-2 JQuery 中 AJAX 全局时间方法

方法名称	参数类型	参数说明
ajaxStart()	function	规定当 AJAX 请求开始时运行的函数。
ajaxStop()	function	规定当 AJAX 请求结束时运行的函数。
ajaxComplete()	function(event,xhr,options)	规定当 AJAX 请求完成时运行的函数。 额外的参数： event——包含 event 对象 xhr——包含 XMLHttpRequest 对象 options——包含 AJAX 请求中使用的选项
ajaxError()	function(event,xhr,options,exc)	规定当请求失败时运行的函数。 额外的参数： event——包含 event 对象 xhr——包含 XMLHttpRequest 对象 options——包含 AJAX 请求中使用的选项 exc——包含 JavaScript exception
ajaxSend()	function(event,xhr,options)	规定请求即将发送时运行的函数。 额外的参数： event——包含 event 对象 xhr——包含 XMLHttpRequest 对象 options——包含 AJAX 请求中使用的选项
ajaxSuccess()	function(event,xhr,options)	规定当请求成功时运行的函数。 额外的参数： event - 包含 event 对象 xhr - 包含 XMLHttpRequest 对象 options - 包含 AJAX 请求中使用的选项

5.实时表单验证

- 网站进行注册时，通常为了提高用户体验，会设计进行实时表单内容验证。例如输入欲注册的用户名后，系统会实时访问数据库进行验证该用户名是否可用。本案例将设计一个实时验证的表单，验证信息通过AJAX和后台系统进行通信。



现场演示：

- 案例16-06：实时表单验证

Thanks.