

第18章 PHP与AJAX

- AJAX是当今Web应用开发中一种相当流行的技术，它最大的优点就是给予用户最佳的浏览体验。从技术角度看，AJAX将几种早已存在的技术有机地结合起来，为Web开发带来历史性地突破，以至于有人以此作为一个新Web时代（即所谓的Web2.0）开始的标志。
- 注意：AJAX并不完全依赖于某种服务器端脚本程序，它是在浏览器端实现的技术。
- 无论是使用Java、ruby还是PHP，都可以实现AJAX应用。本章先介绍什么是AJAX及其工作原理，然后介绍在PHP程序中应用AJAX，最后介绍一个实现AJAX功能的PHP类。



18.1 什么是AJAX

- 凡是关注Web开发的读者，应该对AJAX并不陌生，至少应该知道存在这样一个比较“时尚”的Web开发技术。AJAX全称为“Asynchronous JavaScript and XML”（异步JavaScript和XML），是指一种创建交互式网页应用的网页开发技术。AJAX并不是一门新的语言，甚至也不是一项新的技术。从名称就可以知道，它是几项技术按一定的方式组合在一起，在共同的协作中发挥各自的作用。
- AJAX包括使用XHTML和CSS标准实现Web页面，使用DOM实现动态显示和交互，使用XML进行数据交换与处理，最后使用JavaScript绑定和处理所有数据。



18.2 AJAX的工作原理

- 而且，使用异步方式与服务器通信，不需要打断用户的操作，具有更加迅速的响应能力。重要的是，AJAX技术可以把以前一些服务器负担的工作转到客户端，利用客户端闲置的能力来处理响应，可以最大程度的减少冗余请求，以及由于响应而对服务器造成的负担。
- 在AJAX之前，Web站点强制用户进入提交、等待、刷新页面显示数据的流程，用户的动作总是与服务器的“思考时间”同步。而AJAX提供与服务器异步通信的能力，从而使用户从请求/响应的循环中解脱出来。借助于AJAX，可以在用户单击按钮时，使用JavaScript和DHTML立即更新Web页面，并向服务器发出异步请求，以执行更新或查询数据库。当请求返回时，就可以使用JavaScript和CSS来相应地更新Web页面，而不是刷新整个页面。最重要的是，用户甚至不知道浏览器正在与服务器通信：Web站点看起来是即时响应的。



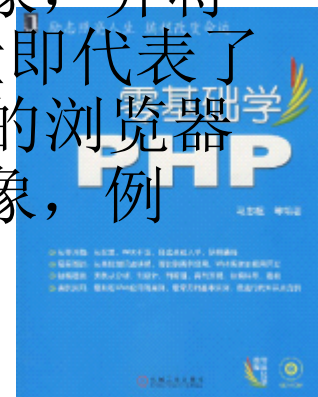
18.3 PHP与AJAX

- AJAX是运行在浏览器端的技术，它在浏览器端和服务器端之间使用异步技术传输数据。但完整的AJAX的应用还需要服务器端的组件，毕竟，浏览器端所有的请求将要由服务器端处理。浏览器端仅仅是通过JavaScript发出请求，并等待服务器的响应，最后处理服务器端传来的数据。服务器端可以使用任何一种语言实现应用，在本书中这种语言只能是PHP。本小节就向读者介绍AJAX和PHP在Web开发中的结合应用。



18.3.1 创建XMLHttpRequest对象

- XMLHttpRequest对象是AJAX的关键所在，也是使用XMLHttpRequest对象是实现AJAX技术的第一步。XMLHttpRequest是一个JavaScript对象，创建该对象很简单，如下代码所示。
- `<script language="javascript">`
- `var xmlHttp = new XMLHttpRequest();`
- `</script>`
- 这段代码创建了一个XMLHttpRequest对象，并将其赋给JavaScript变量xmlHttp，该变量即代表了XMLHttpRequest对象。但是，由于不同的浏览器使用不同的方法创建XMLHttpRequest对象，例如，IE就使用ActiveXObject。



18.3.2 发送异步请求

- 上小节向读者介绍了实现 AJAX应用的关键一步。之所以说它是关键一步，是因为，正是这个“伟大的”XMLHttpRequest 对象将要向Web服务器发出请求。正是这一步实现了通过JavaScript和Web应用程序交互，而不是用户提交给Web服务器的那个HTML表单。
- 那么，到底如何使用XMLHttpRequest 对象向服务器发出请求呢？首先就是要创建一个JavaScript函数，该函数由Web页面调用，比如用户输入数据或者在下拉列表中做了选择时，这个JavaScript函数内将使用XMLHttpRequest 对象向服务器发出请求。在实现使用JavaScript向服务器发出请求之前，读者还需要进一步了解XMLHttpRequest 对象的一些方法，它们在AJAX应用中发挥着至关重要的作用，读者有必要了解每一个方法，这些方法介绍如下。

18.3.3 编写回调函数

- 上一节已经实现了 AJAX 应用中向服务器发送请求的功能，接下来要面对的就是处理服务器的响应。上小节的示例代码的第 22 行所指定的服务器处理完请求之后所要调用的函数，即所谓的回调函数，JavaScript 将在这个函数中完成对服务器返回响应的处理。
- 通过上小节的学习，读者已经知道 XMLHttpRequest 对象的 readyState 属性表示了请求 / 响应过程中的不同状态。当该属性值为 4 时，表示响应加载完毕。XMLHttpRequest 对象的另一个属性 responseText 存放了服务器文本格式的响应，也就是说，服务器将请求的处理结果填充到 XMLHttpRequest 对象的 responseText 属性中。知道了这两点，完成回调函数就顺理成章、水到渠成了。



18.3.4 完整实例

- 有了以上3小节的基础知识，读者已经清楚了 AJAX内部机制，完全可以在具体的应用中实践 AJAX了。这小节将通过一个完整的实例，向读者展示 AJAX与PHP程序的具体应用。
- 这个实例实现的功能是，当用户在 Web页面的下拉列表框中选择某个省的名称后，会在页面上显示该省的省会名称，而此时页面并不刷新。省会名称将有服务器端传送至浏览器端，所以这个完整实例将包含以下 3个部分。
- HTML页面，包含下拉列表框和要显示省会名称的位置。
- JavaScript程序，实现发送请求和处理响应。
- 服务端的PHP程序，用来接收浏览器的请求，向浏览器传送结果数据。



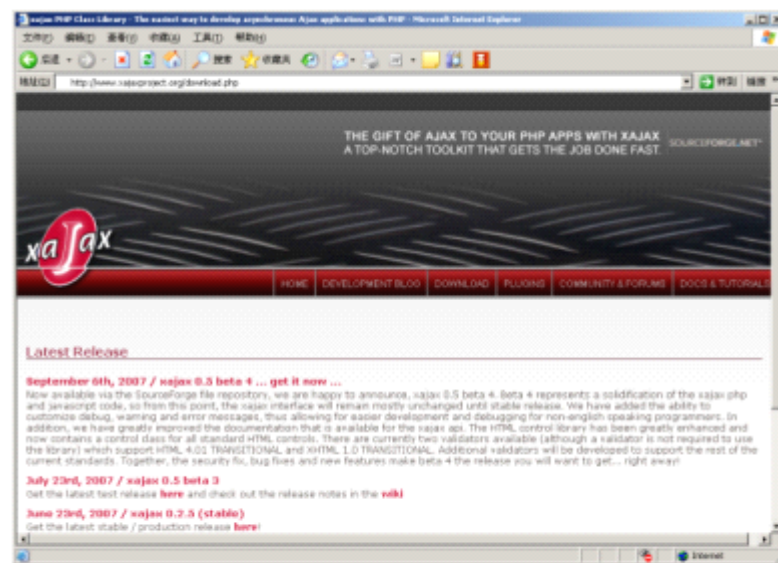
18.4 一个实现AJAX的PHP类——xajax

- 在了解了AJAX的原理和实现，以及PHP和AJAX的配合使用之后，本节向读者介绍一个优秀的实现了AJAX应用的PHP类——xajax。
- xajax是一个开源的PHP类库，它能够综合HTML、CSS、JavaScript和PHP，轻而易举地开发功能强大、基于AJAX的Web应用。使用xajax开发的Web应用，无需重新调入页面，就能够异步调用服务器端的PHP函数和更新Web页面内容。



18.4.1 下载和安装xajax

- xajax的官方网站是<http://www.xajaxproject.org/>，可以在其官网下载最新的或稳定的 xajax程序包。
- 本书将以xajax的一个稳定版本0.2.4为例介绍xajax。xajax不需要复杂的安装，只需将压缩包解压至需要使用xajax的目录下即可。

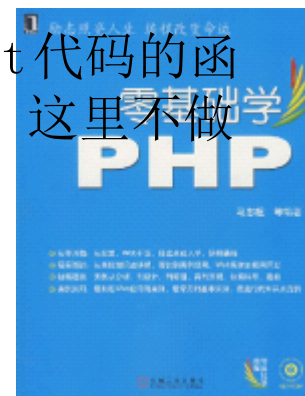


кажак 官方下载页面



18.4.2 xajax简介

- 可以在xajax的程序包（以0.2.4版本为例）中看到以下主要文件。
- xajax.inc.php，这个文件定义了 xajax 类。该类的主要作用是生成页面使用的 JavaScript，该类也可以执行和处理由PHP程序返回到页面的 XML 消息。
- xajaxResponse.inc.php，这个文件定义了 xajaxResponse，该类用来创建返回页面的响应。该响应中包含有一个或多个用来更新 Web 页面的命令消息。稍后将会详细介绍该类。
- xajaxCompress.php，定义了一个压缩 JavaScript 代码的函数，这个函数与本书介绍使用 xajax 的内容无关，这里不做讲述。



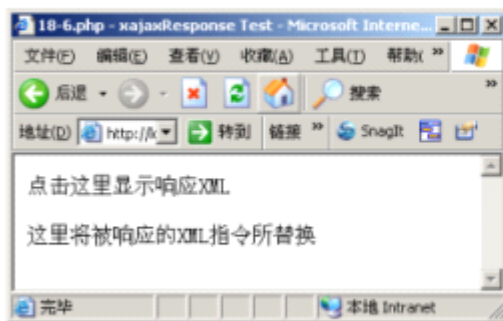
18.4.3 在PHP程序中使用xajax

- 在PHP中使用xajax可以按如下的7个步骤完成。
- (1) 包含xajax类库: `require_once("xajax.inc.php")`。
- (2) 创建xajax对象: `$xajax = new xajax()`。
- (3) 注册需要通过xajax调用的PHP函数的名称: `$xajax->registerFunction("myFunction")`, 注册的函数名为myFunction。
- (4) 编写注册的PHP函数 (如上一步的函数myFunction), 并且在函数之中使用xajaxResponse对象创建向浏览器端返回的返回XML指令。
- (5) 在PHP程序输出任何信息之前, 调用xajax用于接管请求: `$xajax->processRequests()`。
- (6) 在页面的<head></head>标签之间, 使用xajax生成实现AJAX应用所必需的Javascript代码: `$xajax->printJavaScript()`。
- (7) 在程序中, 从Javascript事件或者函数调用前面注册的函数: 例如 `<button onclick="xajax_myFunction(SomeArgument);">`。
- 整个步骤之中, 真正需要开发人员实现的就是编写PHP函数, 然后从函数之中返回xajax的XML响应。而后者通过xajaxResponse类可以非常简单的生成。

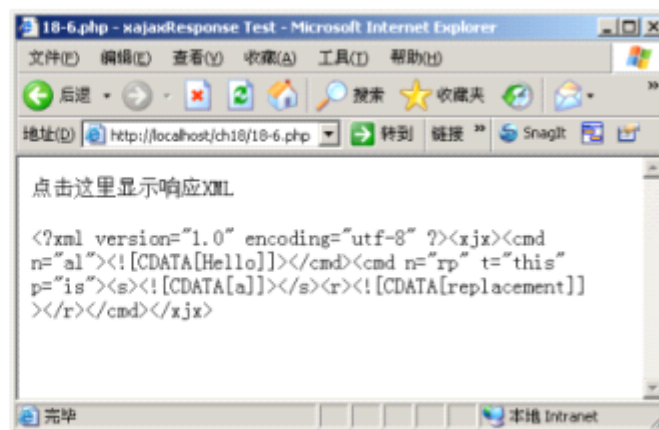


18.4.4 xajax应用实例

- 这节通过一个 xajax 的应用实例，使读者了解 xajax 是如何在 PHP 程序中实现 AJAX 应用的。这个实例将在页面显示有服务器端生成 XML 响应，这个响应以 XML 指令的形式返回至浏览器端，xajax 分消息分析器获得该响应后，通知 xajax 如何更新页面，这个实例中，仅仅是将 XML 指令原封不动地显示到页面上。



xajax 应用效果图 1



xajax 应用效果图 2



18.5 小结

- 本节介绍了AJAX及其和PHP有关的内容，主要讲述的知识点包括以下内容。
- 什么是AJAX。
- AJAX的工作原理。
- 在PHP中如何使用AJAX，包括创建XMLHttpRequest对象、发送异步请求、回调函数的编写，最后通过完整实例介绍PHP程序中的AJAX应用。
- xajax介绍，包括下载和安装xajax、xajax简介、在PHP程序中使用xajax及xajax应用实例。

