

知识点1【AJAX概述】（了解）

- 1、AJAX的特点
- 2、AJAX 核心是 XMLHttpRequest 对象
- 3、工作原理
- 4、AJAX流程图

- 1、创建xmlhttprequest对象
- 2、服务器的状态改变

下面的成员都是服务器的应答：

- 3、open创建请求
- 4、send发送请求

知识点1【AJAX概述】（了解）



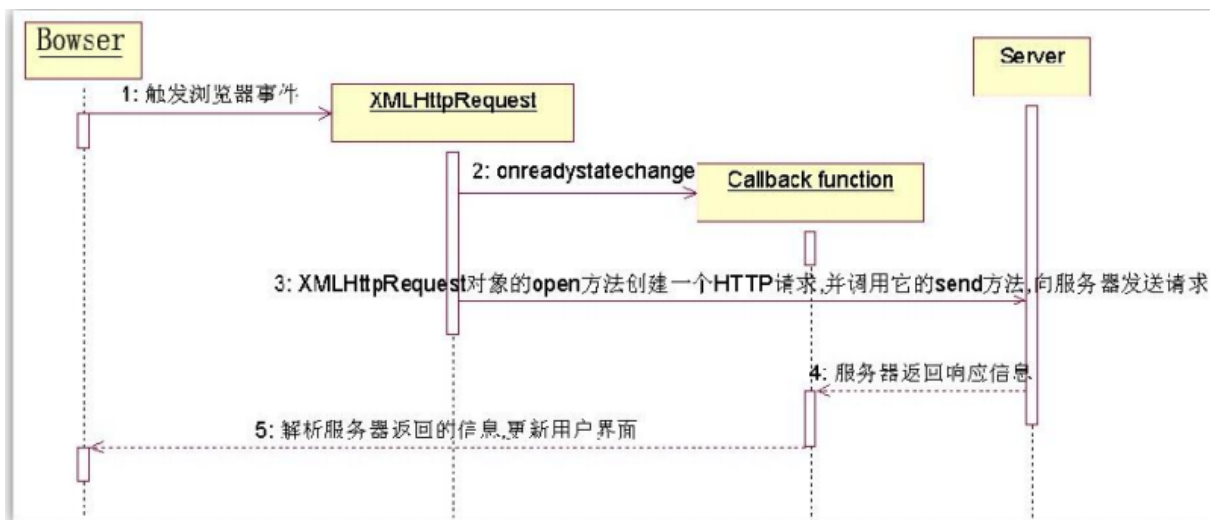
1、AJAX的特点

- 1、AJAX 是 Asynchronous JavaScript And XML 的缩写
- 2、AJAX 是一种用于创建快速动态网页的技术
- 3、AJAX 不是新的编程语言，而是一种使用现有标准的新方法
- 4、AJAX 最大的优点是在不重新加载整个页面的情况下，可以与服务器交换数据并更新部分网页内容。传统的网页（不使用 AJAX）如果需要更新内容，必需重载整个网页面
- 5、AJAX 不需要任何浏览器插件，但需要用户允许 JavaScript 在浏览器上执行。

6、有很多使用 AJAX 的应用程序案例：新浪微博、Google 地图、开心网等等

2、AJAX 核心是 XMLHttpRequest 对象

3、工作原理



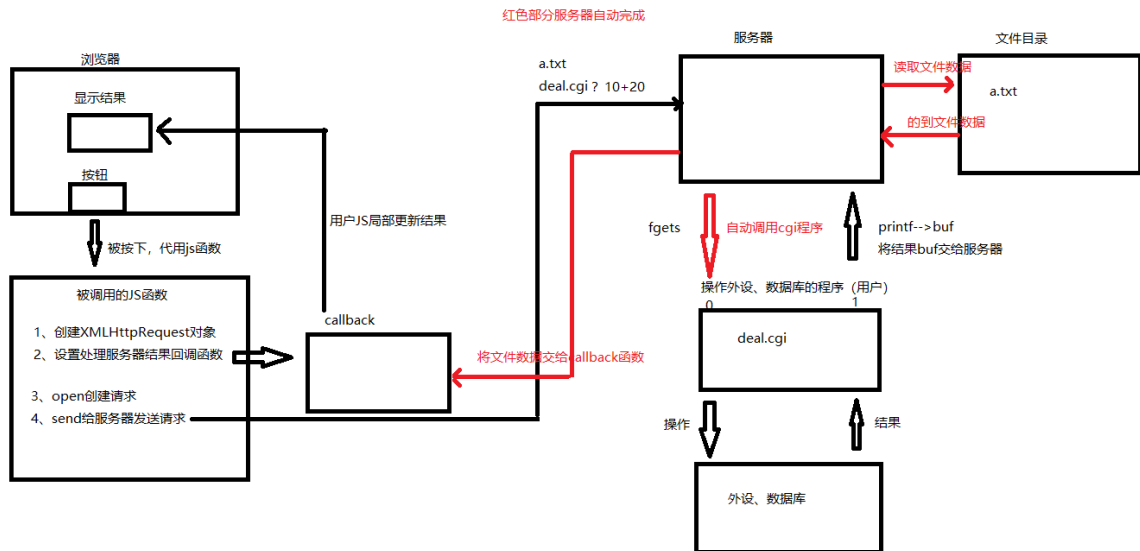
异步流程：

- 1、创建对象
- 2、设置回调函数，fun 函数
- 3、open 创建服务器请求
- 4、send 向服务器发送请求，
- 5、服务器有结果会自动调用 fun 回调函数 在回调函数里面根据服务器返回的响应信息更新用户界

同步流程：

- 1、创建对象
- 3、open 创建服务器请求
- 4、send 向服务器发送请求，
- 5、处理服务的结果

4、AJAX流程图



1、创建xmlhttprequest对象

```

1 function getXMLHttpRequest() {
2     var xmlhttp = null;
3     if (window.XMLHttpRequest) //自动检测当前浏览器的版本，如果是 IE5.0 以上的高版本的浏览器
4     {
5         // code for IE7+, Firefox, Chrome, Opera, Safari
6         xmlhttp = new XMLHttpRequest(); //创建请求对象
7     }
8     else //如果浏览器是低版本的
9     {
10        // code for IE6, IE5
11        xmlhttp = new ActiveXObject("Microsoft.XMLHTTP"); //创建请求对象
12    }
13    return xmlhttp; //返回请求对象
14 }

```

2、服务器的状态改变

Onreadystatechange 每个状态改变时都会触发这个事件处理器，通常会调用一个 JavaScript 函数。状态：

readyState: 请求的状态。0 = 未初始化, 1 = 正在加载, 2 = 已加载, 3 = 交互中, 4 = 完成
 status: 服务器的 HTTP 状态码
 (200 对应 OK, 404 对应 Not Found (未找到))

下面的成员都是服务器的应答：

responseText	服务器的响应，表示为一个字符串
responseXML	服务器的响应，表示为 XML。这个对象可以解析为一个 DOM 对

3、open创建请求

<code>open(“method”, “url”, true)</code>	建立对服务器的请求。method 参数可以是 GET、POST。url 参数可以是相对 URL 或绝对 URL。 true:异步; false:同步
--	---

4、send发送请求

<code>send(content)</code>	向服务器发送请求
----------------------------	----------

index.html

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5     <meta charset="UTF-8">
6     <script type="text/javascript" src="10_js.js"></script>
7     <title>Document</title>
8 </head>
9
10 <body>
11     <div align="center">
12         请选择你的文件: <input type="file" id="file" onchange="get_file_data()">
13         <br>
14         <textarea id="msg" cols="30" rows="10"></textarea>
15     </div>
16 </body>
17
18 </html>
```

10_js.js

```
1 function getXMLHttpRequest() {
2     var xmlhttp = null;
3     if (window.XMLHttpRequest) //自动检测当前浏览器的版本, 如果是 IE5.0 以上的高版本的浏览器
4     { // code for IE7+, Firefox, Chrome, Opera, Safari
5         xmlhttp = new XMLHttpRequest(); //创建请求对象
6     }
7     else //如果浏览器是低版本的
```

```
8     {// code for IE6, IE5
9         xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");//创建请求对象
10    }
11    return xmlhttp;//返回请求对象
12 }
13
14 function get_file_data() {
15     var tmp_name = document.getElementById("file").value;
16     //tmp_name="C:\fakepath\a.txt" 提取a.txt
17     var tmp = tmp_name.slice(tmp_name.lastIndexOf("\\") + 1);
18     var file_name = "./file/";
19     file_name = file_name + tmp;
20
21     // 1、创建xmlHttpRequest对象
22     var xmlhttp;
23     xmlhttp = getXMLHttpRequest();
24
25     // 2、设置服务器回应的回调函数
26     xmlhttp.onreadystatechange = function () {
27         //4交换完成 200是文件找到 才会获取服务器的应答
28         if (xmlhttp.readyState == 4 && xmlhttp.status == 200) {
29             //获取服务器的应答
30             var ret = xmlhttp.responseText;
31             //局部更新网页内容
32             document.getElementById("msg").innerHTML = ret;
33         }
34     }
35
36     // 3、open创建请求
37     var url = file_name;
38     xmlhttp.open("GET", url, true);//true 异步
39
40     // 4、send发送请求
41     xmlhttp.send();
42 }
```

请选择你的文件: [浏览...](#) b.txt

我是b.txt