浅谈 JSONP 的原理与实现

一、什么是JSONP

1.1 同源策略

如果两个页面拥有相同的**协议**，**端口**（如果指定），和**主机**，那么这两个页面就属于同一个**源**。
同源策略分为：

- DOM同源策略：禁止对不同源页面DOM进行操作
- XMLHttpRequest同源策略：禁止向不同源的地址发起HTTP请求

由此可见，Ajax禁止跨域。

1.2 JSONP的原理

JSONP是JSON with Padding的简称，一般用来解决Ajax跨域的问题。它是这样产生的：

1. 页面上调用js文件时不受跨域的影响，而且，凡是拥有src属性的标签都拥有跨域的能力，比如<script>、<img>、<iframe>、<link>、<form>。
2. 可以在远程服务器上设法把数据装进js格式的文件里，供客户端调用处理，实现跨域。
3. 目前最常用的数据交换方式是JSON，客户端通过调用远程服务器上动态生成的js格式文件（一般以JSON后缀）。
4. 客户端成功调用JSON文件后，对其进行处理。
5. 为了便于客户端使用数据，逐渐形成了一种非正式传输协议，人们把它称作JSONP，该协议的一个要点就是允许用户传递一个callback参数给服务端，然后服务端返回数据时会将这个callback参数作为函数名来包裹住JSON数据，这样客户端就可以随意定制自己的函数来自动处理返回数据了。

二、JSONP的客户端具体实现

2.1 上文已指出，页面可执行跨域的js代码（符合安全策略的），那么：

```
<script type="text/javascript">
  var localHandler = function (data) {
    alert('跨域的remote.js文件可以调用本函数，带来的数据是：' + data.result);
  };
</script>
<script type="text/javascript" src = "http://remoteserver.com/remote.js"></script>
```

remote.js的代码：

```
localHandler({
  "result": "我是远程js带来的数据"
});
```

运行后，成功弹出提示窗口，跨域成功。但问题是，如何让远程js知道它应该调用的本地函数叫什么名字？

2.2 只要服务端提供的js脚本是动态生成的就行了，调用者可以传一个参数过去告诉服务端本地函数的名字，于是服务端就可以按照客户的需求来生成js脚本并相应了。

```
  var flightHandler = function (data) {
    alert('你查询的航班结果是：票价 ' + data.price + '元，余票' + data.tickets + '张。');
  };
  var url = 'http://flightQuery.com/jsonp/flightResult.aspx?code=CA1998&callback=flightHandler';
  var script = document.createElement('script');
  script.setAttribute('src', url);
  document.getElementsByTagName('head')[0].appendChild(script);
```

并没有直接把远程js写死，而是编码事项动态查询。这是JSONP的核心部分。在调用的url中传递了一个code参数，告诉服务器要查的是CA1998次航班的信息，而callback参数告诉服务器，本地调用的函数叫做flightHandler。

服务器的flightResult.aspx生成了以下代码提供给页面：

```
flightHandler({
  "code": "CA1998",
  "price": 1780,
  "tickets": 5
});
```

运行一下页面，成功提示窗口，JSONP的执行全过程顺利完成。

2.3 jQuery代码

```
$(function() {
  $('button').on('click', function(event) {
    event.preventDefault();
    $.ajax({
      type: 'GET',
      async: false,
      url: 'http://flightQuery.com/jsonp/flightResult.aspx?code=CA1998',
      dataType: 'jsonp',
      jsonp: 'callback',
      jsonpCallback: 'flightHandler',  // 默认为jQuery自动生成的随机函数名
      success: function (json) {
        alert('你查询的航班结果是：票价 ' + json.price + '元，余票' + json.tickets + '张。');
      },
      error: function () {
        alert('fail');
      }
    });
  });
});
```

jQuery自动生成回调函数并把数据取出来供success属性方法来调用。

三、JSONP与Ajax的关系

- Ajax与JSONP这两种技术看起来很像，目的也一样，都是请求一个url，然后把服务器返回的数据进行处理，因此jQuery等框架都把JSONP作为Ajax的一种形式。
- 实际上Ajax与JSONP有着本质上的不同。Ajax的核心是通过XMLHttpRequest获取数据，而JSONP的核心则是动态添加<script>标签来调用服务器提供的js文件。
- Ajax与JSONP的区别也不在于是否跨域，Ajax通过服务端代理也可以跨域，JSONP也可获取同源数据。

四、参考

1. 【原创】说说JSON和JSONP，也许你会豁然开朗，含jQuery用例
2. 深入浅出JSONP--解决ajax跨域问题
3. 浏览器的同源策略 - MDN