Wang,Jie

第三方嵌入

嵌入解决方案

|  |  |
| --- | --- |
| 文档名称 | 第三方嵌入解决方案 |
| 使用场景 | 嵌入第三方网站 |
| 负责人 | 王杰 |
| 文档编写日期 | 2015-04-11 |
| 版本 | 1.0.0.1 |

1. 概述

第三方嵌入通常只在宿主网站页面中引用第三方嵌入脚本，在宿主网站主要用于信息推广，提供沟通与服务的链接平台。

1. 解决方案

嵌入适用场景

嵌入的使用场景主要在需要在第三方网站中推广使用自身的信息时，可使用嵌入第三方网站页面的方式，将信息挂载在第三方网站中，从而使信息的传播性更广泛。

嵌入关键方面

由于需要嵌入到第三方的网站中，然而第三方网站的不可控性，寄宿的页面环境错综复杂。所有在开发嵌入的脚本时需要特殊关注一些方面来保证嵌入信息能够稳定健壮的挂载在第三方网站中。

I、代码运行环境的独立性

代码的执行环境应该与第三方网站的环境关联分离开来，在变量、执行作用域、样式设定中都要注意执行环境的独立性。例如嵌入代码的总入口采用自执行函数进行自触发执行、变量与class命名采用特有语义化名称、采用私有变量对外尽量不暴漏和访问全局变量和函数。

II、嵌入的跨域通信

在第三方网站的页面中，通常需要我们收集用户或者与后台进行通信，以便能让用户通过嵌入的信息提供给用户实质的服务。

通信基本包含两类：

只关注发送，不关注接收

不仅关注发送还关注服务器的反馈

以上的两类请求在于后台通信时都面临是同域还是跨域的问题。由于嵌入在了第三方网站中，所以所有的请求通信基本都是采用跨域通信来与后台进行信息传递与接收。

跨域主要分为跨协议、跨子域、跨域名三种情况。此处主要介绍跨域名通信方式。嵌入中主要属于跨域名通信情况。

跨域名通常采用的方式有window.name、jsonp、iframe+form等。

window.name方式主要用于通过iframe发送跨域请求，服务器返回js形式内容更新window.name以此来进行通信。

Jsonp是比较流行的跨域通信方案。原理就是在发起请求时，动态的在页面加载一个script标签，因为src可以接收跨域资源，然后这个script标签的资源是执行一个js方法，并且将服务端返回的数据作为参数传递过来。这种情况唯一的缺点就是只能发送get请求，不适用用post请求。

Iframe+form可以满足进行跨域发送post请求的场景。在需要保存信息到后台的时候可以使用此种方法实现跨域保存信息。

III、嵌入的脚本的兼容性

嵌入的第三方网站由于环境的不确定性，需要满足在各类浏览器和各类版本的浏览器中顺利执行。所有需要考虑方法执行的兼容性和样式的兼容性，以此来满足各类浏览器和各类版本浏览器的呈现一致性和正确性。

在脚本兼容性需要注意的例如事件绑定的兼容性、事件对象的兼容性、ajax兼容性、json、dom的设置文本内容等等。

嵌入的结构设计时尽量采用非常用的容器元素，如用ins替代使用div进行布局。尽量不采用span、p等元素，以降低被第三方网站样式设置影响的可能性。

嵌入的原型设计

对嵌入式的模块代码设计实现如下：

dep

|\_\_依赖包

doc

|\_\_文档

entry

|\_\_index.html（测试页）

resource

|\_\_img

| |\_\_图片资源

src

|\_\_common(公共脚本)

|\_\_css(样式)

|\_\_embed(嵌入业务脚本)

|\_\_main.js(入口控制脚本)

原型图

嵌入作用域

Server

Ajax

Config

Event

Dom

Template

View

Render

1. 使用案例

现在使用嵌入方案的有新商桥嵌入。同类的也有商桥也有嵌入式。

可参考网址：http://w3help.org/zh-cn/causes/index.html