


随它去吧

博客园 :: 首页 :: 新随笔 :: 联系 :: 订阅  :: 管理 posts - 21, comments - 393, trackbacks - 0, articles - 0

<	2016年9月						>
日	一	二	三	四	五	六	
28	29	30	31	1	2	3	
4	5	6	7	8	9	10	
11	12	13	14	15	16	17	
18	19	20	21	22	23	24	
25	26	27	28	29	30	1	
2	3	4	5	6	7	8	



公告

昵称： 随它去吧
园龄： 5年2个月
荣誉： 推荐博客
粉丝： 308
关注： 4
[+加关注](#)



搜索

找找看
谷歌搜索



常用链接

[我的随笔](#)
[我的评论](#)
[我的参与](#)
[最新评论](#)
[我的标签](#)
[更多链接](#)



我的标签

[Sencha Touch \(6\)](#)
[MVC \(4\)](#)
[android \(4\)](#)
[官方文档 \(4\)](#)
[翻译 \(4\)](#)

【原创】说说JSON和JSONP，也许你会豁然开朗，含jQuery用例

Posted on 2012-04-19 23:37 随它去吧 阅读(152924) 评论(236) 编辑 收藏

前言：

说到AJAX就会不可避免的面临两个问题，第一个是AJAX以何种格式来交换数据？第二个是跨域的需求如何解决？这两个问题目前都有不同的解决方案，比如数据可以用自定义字符串或者用XML来描述，跨域可以通过服务器端代理来解决。

但到目前为止最被推崇或者说首选的方案还是用JSON来传数据，靠JSONP来跨域。而这就是本文将要讲述的内容。

JSON和JSONP虽然只有一个字母的差别，但其实他们根本不是一回事儿：JSON是一种数据交换格式，而JSONP是一种依靠开发人员的聪明才智创造出的一种非官方跨域数据交互协议。我们拿最近比较火的谍战片来打个比方，JSON是地下党们用来书写和交换情报的“暗号”，而JSONP则是把用暗号书写的情报传递给自己同志时使用的接头方式。看到没？一个是描述信息的格式，一个是信息传递双方约定的方法。

既然随便聊聊，那我们就不再采用教条的方式来讲述，而是把关注重心放在帮助开发人员理解是否应当选择使用以及如何使用上。

什么是JSON？

iphone(3)
jQuery(3)
ASP.NET(2)
userAgent(2)
浏览器兼容性(2)
更多

随笔分类

ASP.NET(2)
CSS+JS(4)
jQuery(3)
Sencha Touch学习(11)
Web开发(13)
业界(1)

随笔档案

2012年4月 (1)
2012年3月 (5)
2012年2月 (6)
2011年10月 (5)
2011年7月 (4)

最新评论

1. Re: 【原创】说说JSON和JSONP，也许你会豁然开朗，含jQuery用例
不愧是大牛，讲解非常透彻清楚
--TTeeaa
2. Re: 【原创】说说JSON和JSONP，也许你会豁然开朗，含jQuery用例
非常感谢！
--陈磊_9527
3. Re: 【原创】说说JSON和JSONP，也许你会豁然开朗，含jQuery用例
初次接触，很好懂，感谢
--lqv587
4. Re: 【原创】说说JSON和JSONP，也许你会豁然开朗，含jQuery用例
原理讲得很清晰，好文章，赞一个
--Amarantin
5. Re: 【原创】说说JSON和JSONP，也许你会豁然开朗，含jQuery用例

前面简单说了一下，JSON是一种基于文本的数据交换方式，或者叫做数据描述格式，你是否该选用他首先肯定要关注它所拥有的优点。

JSON的优点：

- 1、基于纯文本，跨平台传递极其简单；
- 2、Javascript原生支持，后台语言几乎全部支持；
- 3、轻量级数据格式，占用字符数量极少，特别适合互联网传递；
- 4、可读性较强，虽然比不上XML那么一目了然，但在合理的依次缩进之后还是很容易识别的；
- 5、容易编写和解析，当然前提是你要知道数据结构；

JSON的缺点当然也有，但在作者看来实在是无关紧要的东西，所以不再单独说明。

JSON的格式或者叫规则：

JSON能够以非常简单的方式来描述数据结构，XML能做的它都能做，因此在跨平台方面两者完全不分伯仲。

- 1、JSON只有两种数据类型描述符，大括号{}和方括号[]，其余英文冒号:是映射符，英文逗号,是分隔符，英文双引号"是定义符。
- 2、大括号{}用来描述一组“不同类型的无序键值对集合”（每个键值对可以理解为OOP的属性描述），方括号[]用来描述一组“相同类型的有序数据集合”（可对应OOP的数组）。
- 3、上述两种集合中若有多个子项，则通过英文逗号,进行分隔。
- 4、键值对以英文冒号:进行分隔，并且建议键名都加上英文双引号"，以便于不同语言的解析。
- 5、JSON内部常用数据类型无非就是字符串、数字、布尔、日期、null 这么几个，字符串必须用双引号引起来，其余的都不用，日期类型比较特殊，这里就不展开讲述了，只是建议如果客户端没有按日期排序功能需求的话，那么把日期时间直接作为字符串传递就好，可以省去很多麻烦。

JSON实例：

非常好，通俗易懂

--YunGaZeon



阅读排行榜

1. 【原创】说说JSON和JSONP，也许你会豁然开朗，含jQuery用例(152915)
2. 巨文！浏览器自身的BUG —— 不同浏览器在页面缩放状态下的渲染差异分析！让你的网站在iPhone和Android上同样完美！(21023)
3. Sencha Touch 2 入门 Demo 之一：搭建 Sencha Touch 2 的运行环境(14106)
4. Sencha Touch 1.1.1 之初接触（一）怎样入手并写一个漂亮的demo(13025)
5. Web开发兼容性系列文章（二）：根据userAgent值的特性判断客户端设备及浏览器类型(11186)



评论排行榜

1. 【原创】说说JSON和JSONP，也许你会豁然开朗，含jQuery用例(236)
2. 巨文！浏览器自身的BUG —— 不同浏览器在页面缩放状态下的渲染差异分析！让你的网站在iPhone和Android上同样完美！(64)
3. 【原创】唯冠科技 与 iPad 商标的前世今生(22)
4. Sencha Touch 1.1.1 之初接触（一）怎样入手并写一个漂亮的demo(17)
5. Web开发兼容性系列文章（一）：不同设备浏览器的userAgent值大全(11)



推荐排行榜

1. 【原创】说说JSON和JSONP，也许你会豁然开朗，含jQuery用例(471)
2. 巨文！浏览器自身的BUG —— 不同浏览器在页面缩放状态下的渲染差异分析！让你的网站在iPhone和Android上同样完美！(39)
3. 【原创】唯冠科技 与 iPad 商标的前世今生(10)
4. Sencha Touch 2 入门 Demo 之一：搭建 Sencha Touch 2 的运行环境(5)
5. Web开发兼容性系列文章（二）：根据userAgent值的特性判断客户端设备及浏览器类型(5)



// 描述一个人

```
var person = {
    "Name": "Bob",
    "Age": 32,
    "Company": "IBM",
    "Engineer": true
}
```

// 获取这个人的信息

```
var personAge = person.Age;
```

// 描述几个人

```
var members = [
    {
        "Name": "Bob",
        "Age": 32,
        "Company": "IBM",
        "Engineer": true
    },
    {
        "Name": "John",
        "Age": 20,
        "Company": "Oracle",
        "Engineer": false
    },
    {
        "Name": "Henry",
        "Age": 45,
        "Company": "Microsoft",
        "Engineer": false
    }
]
```

// 读取其中John的公司名称

```
var johnsCompany = members[1].Company;
```

// 描述一次会议

```
var conference = {
    "Conference": "Future Marketing",
    "Date": "2012-6-1",
    "Address": "Beijing",
    "Members":
    [
        {
            "Name": "Bob",
            "Age": 32,
            "Company": "IBM",
```

```
        "Engineer": true
    },
    {
        "Name": "John",
        "Age": 20,
        "Company": "Oracle",
        "Engineer": false
    },
    {
        "Name": "Henry",
        "Age": 45,
        "Company": "Microsoft",
        "Engineer": false
    }
]

// 读取参会者Henry是否工程师

var henryIsAnEngineer =
    conference.Members[2].Engineer;
```



关于JSON，就说这么多，更多细节请在开发过程中查阅资料深入学习。

什么是JSONP？

先说说JSONP是怎么产生的：

其实网上关于JSONP的讲解有很多，但却千篇一律，而且云里雾里，对于很多刚接触的人来讲理解起来有些困难，小可不才，试着用自己的方式来阐释一下这个问题，看看是否有帮助。

- 1、一个众所周知的问题，Ajax直接请求普通文件存在跨域无权限访问的问题，甭管你是静态页面、动态网页、web服务、WCF，只要是跨域请求，一律不准；
- 2、不过我们又发现，Web页面上调用js文件时则不受是否跨域的影响（不仅如此，我们还发现凡是拥有"src"这个属性的标签都拥有跨域的能力，比如<script>、、<iframe>）；
- 3、于是可以判断，当前阶段如果想通过纯web端（ActiveX控件、服务端代理、属于未来的HTML5之Websocket等方式不算）跨域访问数据就只有一种可

能，那就是在远程服务器上设法把数据装进js格式的文件里，供客户端调用和进一步处理；

4、恰巧我们已经知道有一种叫做JSON的纯字符数据格式可以简洁的描述复杂数据，更妙的是JSON还被js原生支持，所以在客户端几乎可以随心所欲的处理这种格式的数据；

5、这样子解决方案就呼之欲出了，web客户端通过与调用脚本一模一样的方式，来调用跨域服务器上动态生成的js格式文件（一般以JSON为后缀），显而易见，服务器之所以要动态生成JSON文件，目的就在于把客户端需要的数据装入进去。

6、客户端在对JSON文件调用成功之后，也就获得了自己所需的数据，剩下的就是按照自己需求进行处理和展现了，这种获取远程数据的方式看起来非常像AJAX，但其实并不一样。

7、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。

如果对于callback参数如何使用还有些模糊的话，我们后面会有具体的实例来讲解。

JSONP的客户端具体实现：

不管jQuery也好，extjs也罢，又或者是其他支持jsonp的框架，他们幕后所做的工作都是一样的，下面我来循序渐进的说明一下jsonp在客户端的实现：

1、我们知道，哪怕跨域js文件中的代码（当然指符合web脚本安全策略的），web页面也是可以无条件执行的。

远程服务器remoteserver.com根目录下有个remote.js文件代码如下：

```
alert('我是远程文件');
```

本地服务器localhost.com下有个jsonp.html页面代码如下：



```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <title></title>
    <script type="text/javascript"
src="http://remoteserver.com/remote.js">
</script>
</head>
<body>

</body>
</html>
```



毫无疑问，页面将会弹出一个提示窗体，显示跨域调用成功。

2、现在我们在jsonp.html页面定义一个函数，然后在远程remote.js中传入数据进行调用。

jsonp.html页面代码如下：



```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <title></title>
    <script type="text/javascript">
        var localHandler = function(data){
            alert('我是本地函数，可以被跨域的
remote.js文件调用，远程js带来的数据是：' +
data.result);
        };
    </script>
    <script type="text/javascript"
src="http://remoteserver.com/remote.js">
</script>
</head>
<body>

</body>
</html>
```



remote.js文件代码如下：

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

运行之后查看结果，页面成功弹出提示窗口，显示本地函数被跨域的远程js调用成功，并且还接收到了远程js带来的数据。很欣喜，跨域远程获取数据的目的基本实现了，但是又一个问题出现了，我怎么让远程js知道它应该调用的本地函数叫什么名字呢？毕竟是jsonp的服务者都要面对很多服务对象，而这些服务对象各自的本地函数都不相同啊？我们接着往下看。

3、聪明的开发者很容易想到，只要服务端提供的js脚本是动态生成的就行了呗，这样调用者可以传一个参数过去告诉服务端“我想要一段调用XXX函数的js代码，请你返回给我”，于是服务器就可以按照客户端的需求来生成js脚本并响应了。

看jsonp.html页面的代码：



```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <title></title>
    <script type="text/javascript">
        // 得到航班信息查询结果后的回调函数
        var flightHandler = function(data) {
            alert('你查询的航班结果是：票价 ' +
data.price + ' 元，' + '余票 ' + data.tickets
+ ' 张。');
        };
        // 提供jsonp服务的url地址（不管是什么类型的地址，最终生成的返回值都是一段javascript代码）
        var url =
"http://flightQuery.com/jsonp/flightResult.aspx?code=CA1998&callback=flightHandler";
        // 创建script标签，设置其属性
        var script =
document.createElement('script');
        script.setAttribute('src', url);
        // 把script标签加入head，此时调用开始
        document.getElementsByTagName('head')
[0].appendChild(script);
    </script>
</head>
<body>

</body>
```


</html>



这次的代码变化比较大，不再直接把远程js文件写死，而是编码实现动态查询，而这也正是jsonp客户端实现的核心部分，本例中的重点也就在于如何完成jsonp调用的全过程。

我们看到调用的url中传递了一个code参数，告诉服务器我要查的是CA1998次航班的信息，而callback参数则告诉服务器，我的本地回调函数叫做flightHandler，所以请把查询结果传入这个函数中进行调用。

OK，服务器很聪明，这个叫做flightResult.aspx的页面生成了一段这样的代码提供给jsonp.html（服务端的实现这里就不演示了，与你选用的语言无关，说到底就是拼接字符串）：

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

我们看到，传递给flightHandler函数的是一个json，它描述了航班的基本信息。运行一下页面，成功弹出提示窗口，jsonp的执行全过程顺利完成！

4、到这里为止的话，相信你已经能够理解jsonp的客户端实现原理了吧？剩下的就是如何把代码封装一下，以便于与用户界面交互，从而实现多次和重复调用。

什么？你用的是jQuery，想知道jQuery如何实现jsonp调用？好吧，那我就好人做到底，再给你一段jQuery使用jsonp的代码（我们依然沿用上面那个航班信息查询的例子，假定返回jsonp结果不变）：



```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0  
Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-  
transitional.dtd">  
<html xmlns="http://www.w3.org/1999/xhtml" >  
<head>  
    <title>Untitled Page</title>  
    <script type="text/javascript"  
src=jquery.min.js"></script>  
    <script type="text/javascript">  
        jQuery(document).ready(function() {
```



```
$.ajax({
    type: "get",
    async: false,
    url:
"http://flightQuery.com/jsonp/flightResult.aspx?code=CA1998",
    dataType: "jsonp",
    jsonp: "callback", //传递给请求处理
    程序或页面的，用以获得jsonp回调函数名的参数名(一般默
    认为:callback)
    jsonpCallback: "flightHandler", //
    自定义的jsonp回调函数名称，默认为jQuery自动生成的随
    机函数名，也可以写"?"，jQuery会自动为你处理数据
    success: function (json) {
        alert('您查询到航班信息：票价：
' + json.price + ' 元，余票： ' + json.tickets
+ ' 张。');
    },
    error: function () {
        alert('fail');
    }
});
});
</script>
</head>
<body>
</body>
</html>
```



是不是有点奇怪？为什么我这次没有写flightHandler这个函数呢？而且竟然也运行成功了！哈哈，这就是jQuery的功劳了，jquery在处理jsonp类型的ajax时（还是忍不住吐槽，虽然jquery也把jsonp归入了ajax，但其实它们真的不是一回事儿），自动帮你生成回调函数并把数据取出来供success属性方法来调用，是不是很爽呀？

好啦，写到这里，我已经无力再写下去，又困又累，得赶紧睡觉。朋友们要是看这不错，觉得有启发，给点个“推荐”呗！由于实在比较简单，所以就不再提供demo源码下载了。

4月20日下午的补充：

没想到上了博客园的头条推荐。看到大家对这篇文章的认

可和评论，还是很开心的，这里针对ajax与jsonp的异同再做一些补充说明：

1、ajax和jsonp这两种技术在调用方式上“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和ext等框架都把jsonp作为ajax的一种形式进行了封装；

2、但ajax和jsonp其实本质上是不同的东西。ajax的核心是通过XMLHttpRequest获取非本页内容，而jsonp的核心则是动态添加<script>标签来调用服务器提供的js脚本。

3、所以说，其实ajax与jsonp的区别不在于是否跨域，ajax通过服务端代理一样可以实现跨域，jsonp本身也不排斥同域的数据的获取。

4、还有就是，jsonp是一种方式或者说非强制性协议，如同ajax一样，它也不一定非要用json格式来传递数据，如果你愿意，字符串都行，只不过这样不利于用jsonp提供公开服务。

总而言之，jsonp不是ajax的一个特例，哪怕jquery等巨头把jsonp封装进了ajax，也不能改变着一点！

注意：此文章属博主原创，转载请注明作者信息和原始链接，谢谢合作。

分类: [jQuery](#), [Sencha Touch学习](#), [Web开发](#)

标签: [jQuery](#), [Sencha Touch](#), [ajax](#), [json](#), [jsonp](#)

好文要顶

关注我

收藏该文



随它去吧

关注 - 4

粉丝 - 308

荣誉: [推荐博客](#)

[+加关注](#)

471

0

« 上一篇: [Sencha Touch 2 入门 Demo 之一: 搭建 Sencha](#)

Touch 2 的运行环境

[< Prev](#) [1](#) [2](#) [3](#) [4](#) [5](#)

Feedback

#201楼
2015-10-10 10:24 by 钻葛格

这是我看过的写jsonp最好的一篇文章，感谢分享，感谢楼主。
[支持\(6\)](#) [反对\(0\)](#)

#202楼
2015-10-10 13:53 by Eyes-Chan

1024 讲的明白 有意思
[支持\(0\)](#) [反对\(0\)](#)

#203楼
2015-10-10 13:55 by Eyes-Chan

@ 乔布儿
他说得可能是，在不需要跨域的时候，使用 ajax post 提交 和
jsonp 效率差距会有多少
[支持\(0\)](#) [反对\(0\)](#)

#204楼
2015-10-10 13:56 by Eyes-Chan

@享受奔跑
哈哈，@享受奔跑 你是个讲师吧？
[支持\(0\)](#) [反对\(0\)](#)

#205楼
2015-10-27 14:21 by 一起走过的日子.....

一看就懂，以前只是有模糊的影响，但是不懂原理，现在看了楼主的
文章，完全能理解了。
[支持\(1\)](#) [反对\(0\)](#)

#206楼
2015-11-17 22:04 by wliyongfeng

通俗易懂，值得推荐。

支持(1) 反对(0)

#207楼

2015-11-25 10:28 by 受戒人

昨天看的，今天才评论，太好了

支持(0) 反对(0)

#208楼

2015-12-03 10:23 by 沃勒

后台的一起发出来呗嘻嘻

支持(0) 反对(0)

#209楼

2015-12-07 16:26 by bfz0d003

写得非常好，不评价对不起楼主的共享精神，赞一个

支持(1) 反对(0)

#210楼

2015-12-19 20:44 by arkflag777

我是臺灣人，這篇真的寫的很棒，淺顯易懂到讓我忍不住創了個帳號來感謝！

支持(2) 反对(0)

#211楼

2015-12-21 10:38 by Heaanc

好文章！
真心好感谢！

支持(1) 反对(0)

#212楼

2015-12-24 10:34 by wzy77cn

写的非常好。看过的jsonp有关的最好的一篇，特意注册来感谢。

支持(0) 反对(0)

#213楼

2015-12-25 15:58 by cssddnnc

写的相当好，代码清晰完整，层次循序渐进，文章通俗易懂。

支持(0) 反对(0)

#214楼

2015-12-25 17:37 by 我是一只小小码农

非常清晰，感谢楼主

支持(0) 反对(0)

#215楼

2016-02-04 15:27 by KissAngels

楼主写的很详细了，真的很棒！

支持(0) 反对(0)

#216楼

2016-03-08 14:55 by 王朋

楼主，jQuery调用那个的服务端代码，能发下吗？

支持(0) 反对(0)

#217楼

2016-03-09 11:32 by cardaminexhz

@ 王朋

只需要建一个remote.js文件

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

然后启动一个服务就ok。启动server，我用的是python。安装下python，进入存放remote.js的目录，cmd执行命令：python -m SimpleHTTPServer 端口号

支持(0) 反对(0)

#218楼

2016-03-29 10:43 by 马铃薯

感谢楼主，好文，易懂。

支持(0) 反对(0)

#219楼

2016-04-15 14:54 by 小凡同学

简直良心，很容易懂！太棒了。

支持(0) 反对(0)

#220楼

2016-04-19 17:25 by lovdin

好文□□□

支持(0) 反对(0)

#221楼

2016-05-03 17:03 by 園田海未

作为一个初学菜鸟，有幸看到这一片文章，谢谢博主

支持(0) 反对(0)

#222楼

2016-05-05 16:48 by 王端端

写的很好，有空再读第二遍。

支持(0) 反对(0)

#223楼

2016-05-12 18:36 by 张凡文

博主最后还是留了一手哈。

我把最终的AJAX JSONP的实现代码放这了。

<http://www.cnblogs.com/zhangfanwen/p/5486860.html>

支持(0) 反对(0)

#224楼

2016-06-16 14:31 by 猫爷喵

json中没有变量的概念吧？有变量的只能是对象好像

支持(0) 反对(0)

#225楼

2016-06-24 15:51 by 间隔年

感谢分享，jsonp理解的透彻多了

支持(0) 反对(0)

#226楼

2016-06-25 13:26 by 洪研

写的非常好，顶一个

支持(0) 反对(0)

#227楼

2016-07-13 19:53 by Midori

一看就懂了，谢谢楼主

支持(0) 反对(0)

#228楼

2016-07-14 11:21 by WideWeide

<http://flightQuery.com/jsonp/flightResult.aspx?code=CA1998>

这个jsonp页面也是由服务器提供才行，如果该服务器只提供webservice接口怎么办呢？

支持(0) 反对(0)

#229楼

2016-07-14 16:48 by sundalei

支持一下，弄清楚了jsonp和jsonpCallback了，谢谢

支持(0) 反对(0)

#230楼

2016-07-28 21:54 by 学习会让你青春永驻

楼主写的真不多,很是感谢能写出这样的分享,希望楼主可以写出更好的文章,让我们更多的能够去学习!

支持(0) 反对(0)

#231楼

2016-08-16 19:11 by svce

看了这个就知道为嘛jsonp都是异步的了。。。

支持(0) 反对(0)

#232楼
2016-08-23 13:53 by YunGaZeon

非常好，通俗易懂

支持(0) 反对(0)

#233楼
2016-09-04 17:33 by Amarantin

原理讲得很清晰，好文章，赞一个

支持(0) 反对(0)

#234楼
2016-09-08 15:36 by lqv587

初次接触，很好懂，感谢

支持(0) 反对(0)

#235楼
2016-09-08 19:35 by 陈磊_9527

非常感谢！

支持(0) 反对(0)

#236楼
2016-09-16 22:48 by TTeeaa

不愧是大牛，讲解非常透彻清楚

支持(0) 反对(0)

- 爱立信计划关闭瑞典制造业务 裁员3000名
 - 海洋清理项目将动用飞机考察大太平洋垃圾带
 - 微软为研发盲文阅读器的麻省理工学院学生申请专利
 - Google为10月4日新品发布会启动线下宣传
 - AI的艺术梦还是很萌的，虽然我们不总能get到
- » 更多新闻...

最新知识库文章：

- 陈皓：什么是工程师文化？
 - 没那么难，谈CSS的设计模式
 - 程序猿媳妇儿注意事项
 - 可是姑娘，你为什么要编程呢？
 - 知其所以然（以算法学习为例）
- » 更多知识库文章...

Powered by:

博客园

Copyright © 随它去吧