

## 使用 Yahoo!web 服务的 json(Javascript 对象标识)

Json 是基于 Javascript 语言对象标识的一种轻量级的数据格式.但是它的读写操作却不必依赖于 Javascript;而且他很容易被任意语言、类库和大多数语言中已经存在的处理 json 的工具所解析.

为了能够对json及其用法进行全面了解,我们建议你访问Douglas Crockford的JSON.org(我当时看json就是从这个网站上看到的,保证你很快就能知道json是什么,怎么使用,非常推荐大家去看,Reany注),并且顺便浏览此网站上的JSON: The Fat-Free Alternative to XML.

Yahoo!web服务的很多输出格式都已经由XML转变为Josn格式.本篇文章将会描述Yahoo!web服务的json输出和如何在你的程序中使用它.本文包括下面几个部分:

- ❖ Json 格式总览
- ❖ 如何请求 JSON 输出
- ❖ 在你的脚本和应用中使用 JSON
- ❖ JSON pk XML

### JSON 格式总览

JSON 是基于 Javascript 对象标识的一种非常简单的文本格式.标识包含以下基本元素:

**对象:**对象以大括号({})开始和结束.

**对象成员:**成员由被冒号(:)隔开的字符串和值组成.

**数组:**数组以大括号开始和结束,它包含内部值.这些值被逗号隔开.

**值:**这个值可以是字符串,数字,或者对象,或者数组,或者 true,false 或者 null.

**字符串:**字符串要被双引号括着,它可以容纳 unicode 字符或者通常的转义字符.

简单的 JSON 格式如下:

```
{
  "Image": {
    "Width":800,
    "Height":600,
    "Title":"View from 15th Floor",
```

```

    "Thumbnail":
    {
        "Url": "http://s3.amazonaws.com/yimg.com/image/481989943",
        "Height": 125,
        "Width": "100"
    },
    "IDs": [ 116, 943, 234, 38793 ]
}
}

```

在上例中,Image是根级对象,其他所有数据都是它的成员.Width,Height和Title包含数字和字符数据的简单成员.Thumbnail是一个成员包括URL,Height和Width的哈希对象(原文为a nested object,哈希对象是我自己的理解.js中的哈希表就是这样描述的),IDs是一个包含许多数字的数组.注意:Url字符串值中的斜杠是为了转义("\").

## 如何请求 JSON 输出

许多支持 JSON 的 Yahoo!web 服务使用同一的参数来生成和处理 JSON.

- ❖ 使用"output=json"获取 JSON
- ❖ 使用"callback=function"执行回调
- ❖ Json pk XML

使用"output=json"形式

Yahoo!web 服务默认返回 xml 格式数据.为了获取 JSON 格式,你可以在请求中使用 "output=json"参数:

<http://search.yahooapis.com/ImageSearchService/V1/imageSearch?appid=YahooDemo&query=Madonna&output=json>

使用"callback=function"形式:

回调参数(callback=function)把返回的json文本括在括号和你选择的函数内.例如:

```
class=http://search.yahooapis.com/ImageSearchService/V1/imageSearch?appid=YahooDemo&query=Madonna&output=json&callback=ws_results
```

回调函数的名字只能是用大小写字母(A-Z,a-z),数字(0-9),圆点(.),下划线(\_)和中括号([])方括号必须是使用 UEL-encode.参见下面的注意事项.

回调函数括起来的返回结果如下形式:

```
class=cws_results( ...json output... );
```

由于json本身就是Javascript,所以你并不需要解析回调函数中返回的对象..你可以直接访问里面的元素,就好象传递的对象是直接引用你自己的ws\_result 函数.

另外,对于金蛋的函数名称,回调参数也可以使用Javascript对象或者数组形式,例如:

```
class=callback=ws_results.obj 或者 class=callback=ws_results.obj.array[4]
```

注意:回调参数中请求的方括号必须是URL编码,如:

```
class=http://search.yahooapis.com/ImageSearchService/V1/imageSearch?appid=YahooDemo&query=Madonna&output=json&callback=ws_results.obj.array%5B4%5D
```

当客户端Javascript请求web服务时候,回调形式特别有用.通常,使用XMLHttpRequest的web服务请求与浏览器的安全限制(阻止跨域文件的加载)相冲突.这个限制需要你通过放在服务器上的代理来进行请求,或者使用服务器重写技术,让浏览器认为web数据是来自和你web内容同一站点以此来欺骗浏览器.

使用json和回调形式之后,你可以把web请求放在<script>标签内,并且可以使用页内的Javascript代码对结果进行任何操作.使用这种形式,当封装好的web页面加载后,对Yahoo!web服务的json请求也能够加载.这样,代理技术和服务器欺骗技术都可以抛弃了.

实例如下:

```
<html>
<head>
<title>How Many Pictures Of Madonna Do We Have?</title>
</head>
</body>
<script type="text/javascript">
function ws_results(obj) {
    alert(obj.ResultSet.totalResultsAvailable);
}
</script>
<script type="text/javascript"
src="http://search.yahooapis.com/ImageSearchService/V1/imageSearch?appid=YahooDemo&query=Madonna&output=json&callback=ws_results"></script>
</body></body>
</html>
```

保存代码到桌面的madonna.html文件,拖放到浏览器中,神奇的出现了结果.

在你的脚本或应用中使用JSON

JSON文本可以被Javascript通过"eval()"直接编译.

```
var myObj=eval('(' +jsontext+')');
```

注意:eval会在无安全和健全的检验下对JSON中的文本进行编译..尽管Yahoo!web服务提供的JSON不会包含恶意代码,但是使用一个简单检验JSON文本结构而不执行任何代码的JSON解析器,你会更轻松.crockford.com (json.org)上有一个可用的开源JSON解析器.

## 安全关注

尽管Yahoo!web服务不包含js代码仅仅包含数据,但是动态脚本标记结构确实把你的DOM暴露给了Yahoo!.如果你对来自Yahoo!web服务的JSON输出和这种DOM暴露方式有安全考虑,请不要使用这种动态脚本标记---你可以使用XMLHttpRequest 然后手动对JSON进行解析,或者使用xml来代替.

## JSON pk XML

来自Yahoo!web服务的XML和JSON包含了相同的数据,他们的区别仅仅是格式上的.yahoo开发网上的文档默认是xml的输出格式.

由于依赖于你使用的特殊服务,来自service maps的xml输出格式转化为JSON标记也会不尽相同,但是一般来说,你可以指定以下规则:

- 一个包含回应(典型的结果集)的顶级对象
- 简单XML元素(仅包含内容的元素)转换为 string/value对
- 复杂xml元素(包含嵌套元素)转化为嵌套对象
- 属性转换为 string/value对

如果一个元素既包含内容又包括属性,属性会转换为string/value对,而内容会转换为名称为content的string/value对

重复的元素映射数组.数组命名在元素后.数组的值依赖于重复元素的内容,也就是说,简单的内容转换为简单的值,嵌套的元素转换为对象.但是请注意,如果返回的是一个包含一个元素的重复元素的特殊回应,那么你收到的不会是一个包括这个元素的JSON数组--这个元素已经被简单的看作一个复杂的XML元素.对于重复的元素,你应该确保你对数组或者单一值的测试.

下面的2个例子说明XML和JSON格式的等价输出格式

### Xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<ResultSet
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="urn:yahoo:srchmi"
  xsi:schemaLocation="urn:yahoo:srchmi
  http://search.yahooapis.com/ImageSearchService/V1/ImageSearchResponse.xsd"
  totalResultsAvailable="229307" totalResultsReturned="2" firstResultPosition="1"><Result>
  <Title>madonna 116</Title>
  <Summary>Picture 116 of 184</Summary>
```

<Url><http://www.celebritypicturesarchive.com/pictures/m/madonna/madonna-116.jpg></Url>  
<ClickUrl><http://www.celebritypicturesarchive.com/pictures/m/madonna/madonna-116.jpg></ClickUrl>

<RefererUrl>[http://www.celebritypicturesarchive.com/pgs/m/Madonna/Madonna%20picture\\_116.htm](http://www.celebritypicturesarchive.com/pgs/m/Madonna/Madonna%20picture_116.htm)</RefererUrl>

<FileSize>36990</FileSize>  
<FileFormat>jpeg</FileFormat>  
<Height>530</Height>  
<Width>425</Width>  
<Thumbnail>  
<Url><http://scd.mm-b1.yimg.com/image/481989943></Url>  
<Height>125</Height>  
<Width>100</Width>  
</Thumbnail>

</Result>

<Result>

<Title>madonna 118</Title>  
<Summary>Picture 118 of 184</Summary>  
<Url><http://www.celebritypicturesarchive.com/pictures/m/madonna/madonna-118.jpg></Url>  
<ClickUrl><http://www.celebritypicturesarchive.com/pictures/m/madonna/madonna-118.jpg></ClickUrl>

<RefererUrl>[http://www.celebritypicturesarchive.com/pgs/m/Madonna/Madonna%20picture\\_118.htm](http://www.celebritypicturesarchive.com/pgs/m/Madonna/Madonna%20picture_118.htm)</RefererUrl>

<FileSize>40209</FileSize>  
<FileFormat>jpeg</FileFormat>  
<Height>700</Height>  
<Width>473</Width>  
<Thumbnail>  
<Url><http://scd.mm-b1.yimg.com/image/500892420></Url>  
<Height>130</Height>  
<Width>87</Width>  
</Thumbnail>

</Result>

</ResultSet>

**Json:**

```
{"ResultSet": {
  "totalResultsAvailable": "229307",
  "totalResultsReturned": "2",
  "firstResultPosition": "1",
  "Result": [
    {
      "Title": "madonna 116",
      "Summary": "Picture 116 of 184",
      "Url": "http://www.celebritypicturesarchive.com/pictures/m/madonna/madonna-116.jpg",
      "ClickUrl": "http://www.celebritypicturesarchive.com/pictures/m/madonna/madonna-116.jpg",
      "RefererUrl": "http://www.celebritypicturesarchive.com/pgs/m/Madonna/Madonna%20picture_116.htm",
      "FileSize": "36990",
      "FileFormat": "jpeg",
      "Height": "530",
      "Width": "425",
      "Thumbnail": {
        "Url": "http://scd.mm-b1.yimg.com/image/481989943",
        "Height": "125",
        "Width": "100"
      }
    },
    {
      "Title": "madonna 118",
      "Summary": "Picture 118 of 184",
      "Url": "http://www.celebritypicturesarchive.com/pictures/m/madonna/madonna-118.jpg",
      "ClickUrl": "http://www.celebritypicturesarchive.com/pictures/m/madonna/madonna-118.jpg",
```

```
"RefererUrl":"http://www.celebritypicturesarchive.com/pgs/m/Madonna/Madonna%20picture_118.htm",
  "FileSize":"40209",
  "FileFormat":"jpeg",
  "Height":"700",
  "Width":"473",
  "Thumbnail":{
    "Url":"http://scd.mm-b1.yimg.com/image/500892420",
    "Height":"130",
    "Width":"87"
  }
}
```