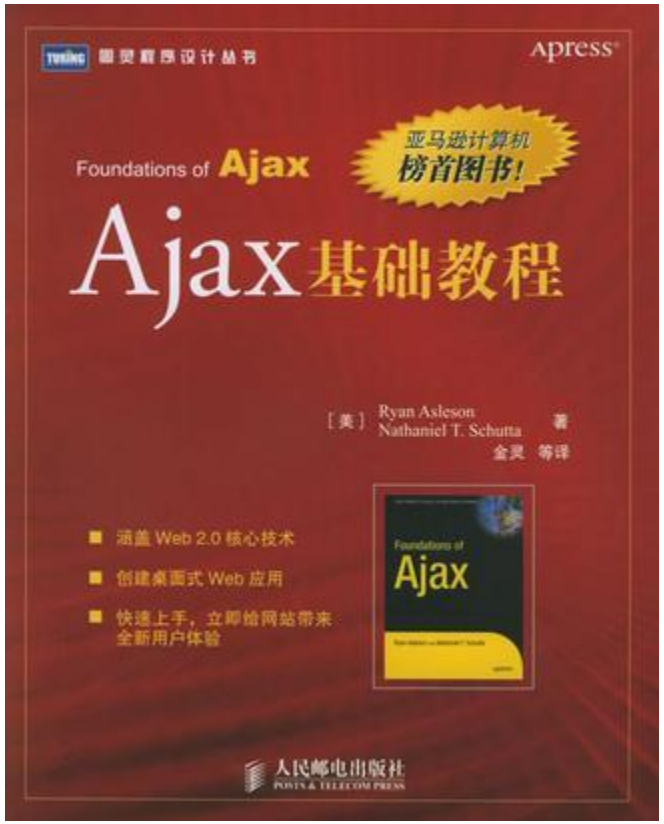
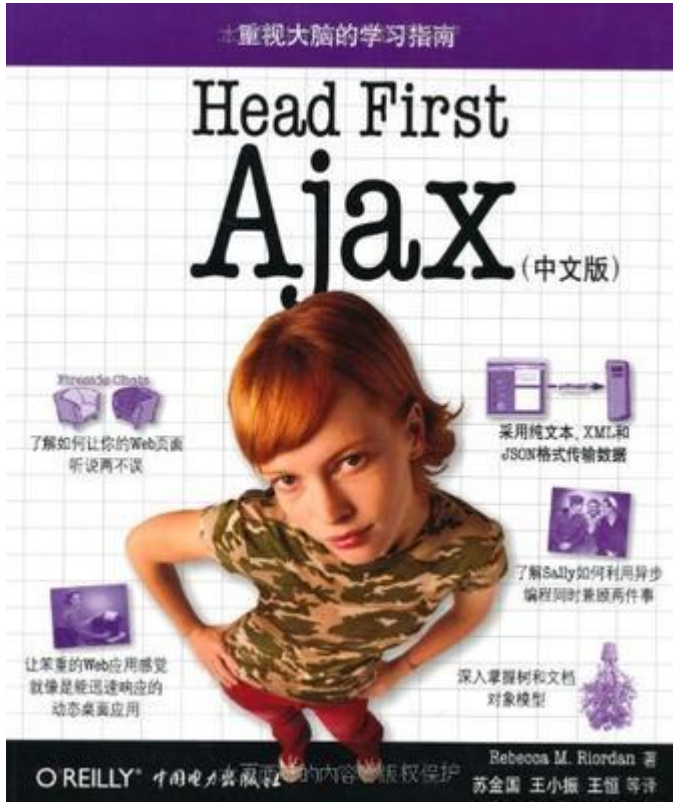
Ajax

凯盛软件

推荐书籍

凯盛软件


图灵程序设计丛书
Apress
亚马逊计算机
榜首图书!
Foundations of Ajax
Ajax基础教程
[美] Ryan Asleson
Nathaniel T. Schutta 著
金灵 等译
涵盖 Web 2.0 核心技术
创建桌面式 Web 应用
快速上手，立即给网站带来全新用户体验
Foundations of
Ajax
人民邮电出版社
POSTS & TELECOM PRESS


重视大脑的学习指南
Head First
Ajax
(中文版)
了解如何让你的Web页面听说再不误
采用纯文本，XML和JSON格式传输数据
了解Sally如何利用异步编程同时兼顾两件事
让笨重的Web应用感觉就像是敏捷迅速响应的动态桌面应用
深入掌握树和文档对象模型
O'REILLY
Rebecca M. Riordan 著
苏金国 王小振 王恒 等译


AJAX
AJAX
AJAX
AJAX
AJAX

Ajax

凯盛软件

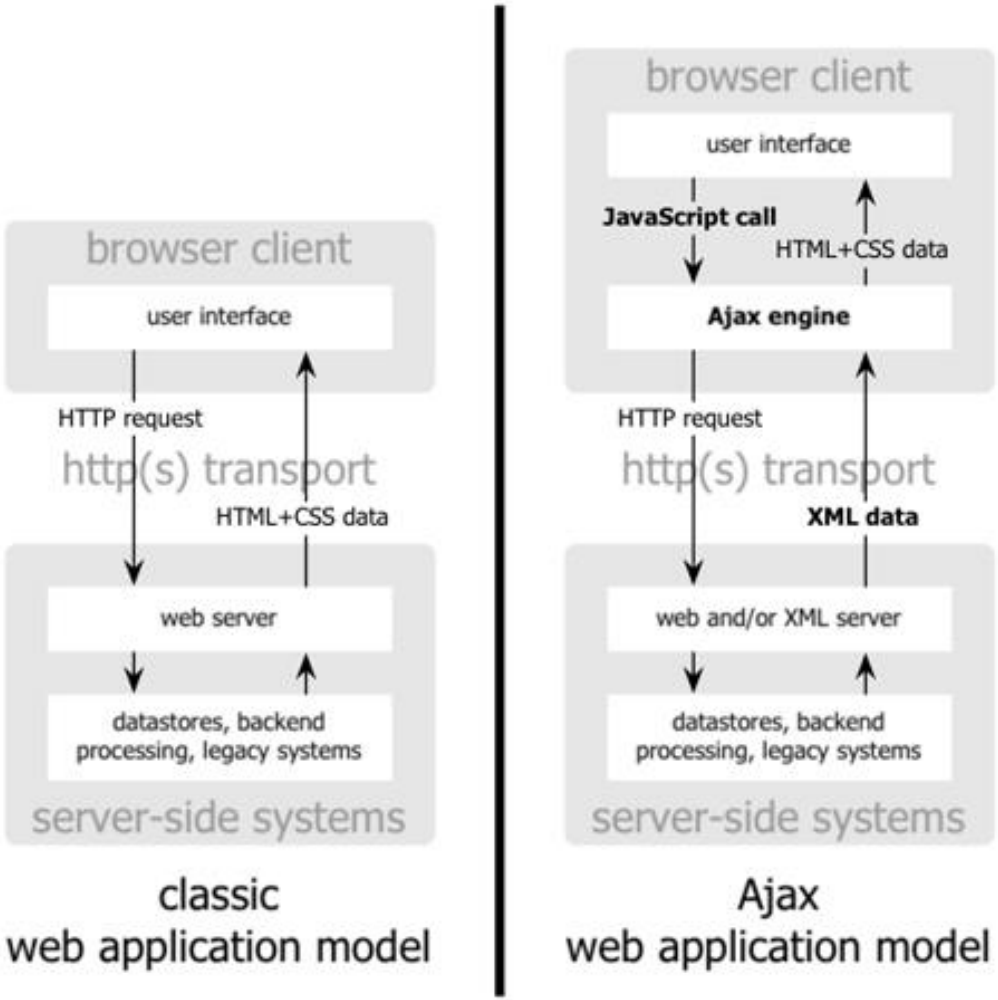
- AJAX的全称是“Asynchronous JavaScript and XML(**异步**JavaScript与XML)”
- 学习Ajax要求掌握以下知识：
  - JavaScript技术
  - 使用JavaScript与DOM进行交互操作的技术
  - 基于Web标准的页面布局
  - XML技术
  - JavaScript解析XML的技术
  - 利用XMLHttpRequest对象发出异步请求的技术
  - 服务器端编程技术（如JSP或Servlet）

http://zh.wikipedia.org/wiki/AJAX

哪里用Ajax

凯盛软件

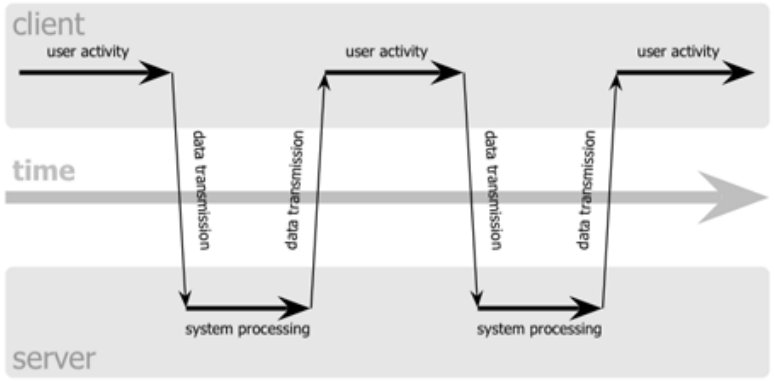
Ajax原理


browser client
user interface
HTTP request
http(s) transport
HTML+CSS data
web server
datastores, backend processing, legacy systems
server-side systems
classic web application model
browser client
user interface
JavaScript call
HTML+CSS data
Ajax engine
HTTP request
http(s) transport
XML data
web and/or XML server
datastores, backend processing, legacy systems
server-side systems
Ajax web application model

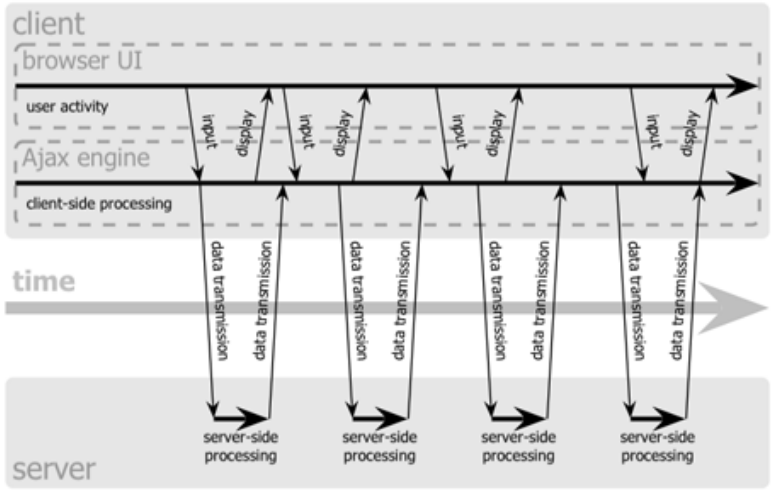
异步请求

凯盛软件

classic web application model (synchronous)



Ajax web application model (asynchronous)



Ajax与传统请求的差异

凯盛软件

不同点	常规操作	AJAX操作
是否刷新	刷新页面	不刷新页面
用户操作	中断，等待新页面下载后继续	不中断
性能	服务器返回整个新页面（包括HTML代码、CSS代码、相关图片等）	服务器仅返回需要的数据
编码复杂度	较简单	较复杂

Ajax引擎

应该做什么？

- 能够被JavaScript调用。JavaScript需要指定请求地址、请求方式与参数，同时JavaScript还需要能够从“引擎”中得到服务器返回的数据。
- 能够异步请求服务器并接受返回的数据。JavaScript本身并不能访问网络，这部分功能必须委托“引擎”实现。
- “引擎”也要能够调用JavaScript。这样才是实现当服务器数据返回时通知JavaScript进行处理。

不该做什么？

- “引擎”不会处理用户的请求，也不会处理业务逻辑，只是将请求转发给服务器，由服务器端的程序进行处理。
- 当服务器端数据返回后，“引擎”不会代替JavaScript完成页面显示工作，只是将通知JavaScript，由JavaScript进行后续的处理。

XMLHttpRequest

IE实现


```
if(window.ActiveXObject){
        var xmlHttp = new ActiveXObject("Microsoft.XMLHTTP");
}
```

```
var xmlHttp = new XMLHttpRequest();
```

兼容模式

凯盛软件

```
var xmlHttp;
function createXMLHttpRequest(){
    if(window.ActiveXObject){
        xmlHttp = new ActiveXObject("Microsoft.XMLHTTP");
    } else {
        xmlHttp = new XMLHttpRequest();
    }
}
```

发出请求

xmlHttp.open(请求方式,请求地址,[是否为异步请求])；
xmlHttp.send()；

Get请求

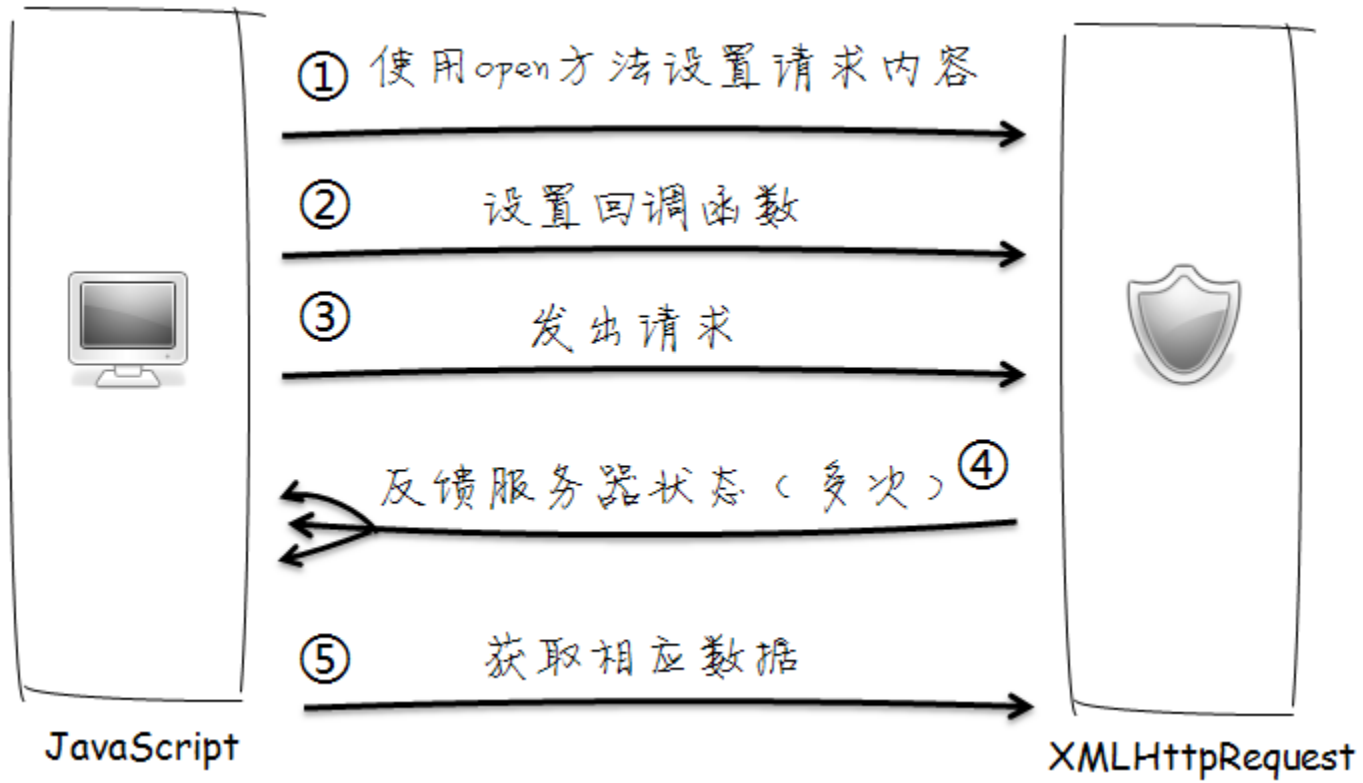

```
xmlHttp.open("GET","ajax.do?name=tom",true);
xmlHttp.send();
```

Post请求

凯盛软件

```
xmlHttp.open("POST","ajax.do",true);
xmlHttp.setRequestHeader("Content-Type","application/x-www-form-urlencoded");
xmlHttp.send("name=tom");
```

接收服务器响应


①使用open方法设置请求内容
②设置回调函数
③发出请求
④反馈服务器状态（多次）
⑤获取相应数据
JavaScript
XMLHttpRequest

```
function sendAjax(){
    createXMLHttpRequest();
    xmlHttp.open("GET","ajax.do",true);
    xmlHttp.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");
    xmlHttp.onreadystatechange = callback;
    xmlHttp.send();
}

function callback(){
        alert("callback");
}
```

获取服务器状态码

凯盛软件

var code = xmlHttp.readyState；

1. 已初始化
2. 发送请求数据
3. 响应数据传输中
4. 响应完毕

不同浏览器状态码出现顺序不同

获取HTTP状态码


```
var serverCode = xmlHttp.status;
```

状态码	含义
200	服务器正常处理了请求并响应
404	请求的页面未找到
403	没有权限访问请求的页面
405	页面不接收该请求方式（比如用GET请求一个只支持doPost方法的Servlet）
408	请求超时
500	服务器处理请求时遇到错误（可能因为应用程序抛出异常导致）
503	服务暂时不可用（可能出现在服务器尚未初始化完成时）
304	网页未修改

获取服务器端的文本值

凯盛软件

```
function callback(){
        if(xmlHttp.readyState == 4){
                if(xmlHttp.status == 200){
                        var result = xmlHttp.responseText;
                        alert(result);
                }
        }
}
```

```
protected void doGet(HttpServletRequest request,
 HttpServletResponse response) throws ServletException, IOException {
    PrintWriter out = response.getWriter();
    out.print("OK");
    out.flush();
    out.close();
}
```

避免GET缓存

使用无意义参数避免缓存

JavaScript代码

```
var http = new XMLHttpRequest();
var timer = new Date().getTime();
http.open('GET', '<%=path%>/servlet/CacheDemo?dummy=' + timer);
http.send();
```

获取当前时间（毫秒为单位）

附加在URL后面

在服务端添加响应头

Servlet代码

```
response.setContentType("text/html; charset=UTF-8");
response.addHeader("pragma", "no-cache");
response.addHeader("cache-control", "no-cache");
response.addHeader("expires", "0");
```

这三个回应头
表示不使用缓存

对中文的支持 GET

凯盛软件

JavaScript代码

先进行客户端转码

```
var http = new XMLHttpRequest();
var v = encodeURIComponent('这是汉字');
http.open('GET', 'url?param=' + v);
http.send();
```

Servlet代码

每个参数分别进行转码

```
String v = request.getParameter("param");
v = new String(v.getBytes("ISO-8859-1"), "UTF-8");
```

对中文的支持 POST

JavaScript代码

```
var http = new XMLHttpRequest();
http.open('POST', 'url');
http.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");
http.send('v=这是汉字');
```

不用转码

Servlet代码

统一进行转码

```
request.setCharacterEncoding("UTF-8");
String v = request.getParameter("v");
```

XML

凯盛软件

- XML 指可扩展标记语言（**EX**tensible **M**arkup **L**anguage）
- XML 是一种**标记语言**，很类似 HTML
- XML 的设计宗旨是**传输数据**，而非显示数据
- XML 标签没有被预定义。您需要**自行定义标签。**
- XML 被设计为具有**自我描述性。**
- XML 是 **W3C 的推荐标准**

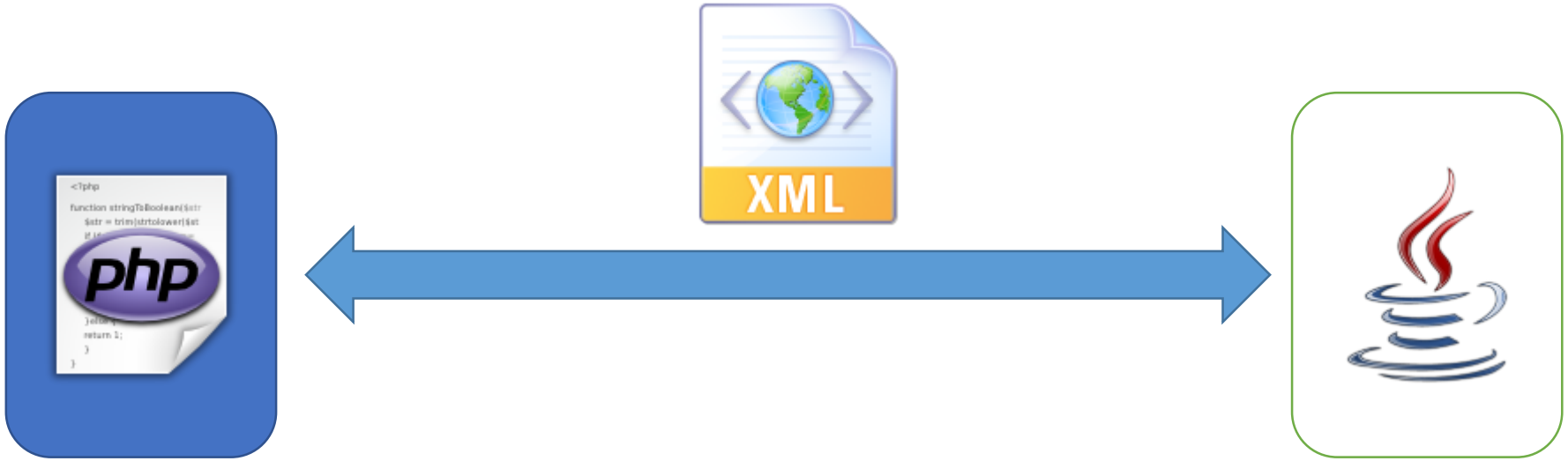
http://www.w3school.com.cn/x.asp

XML 与 HTML 的主要差异

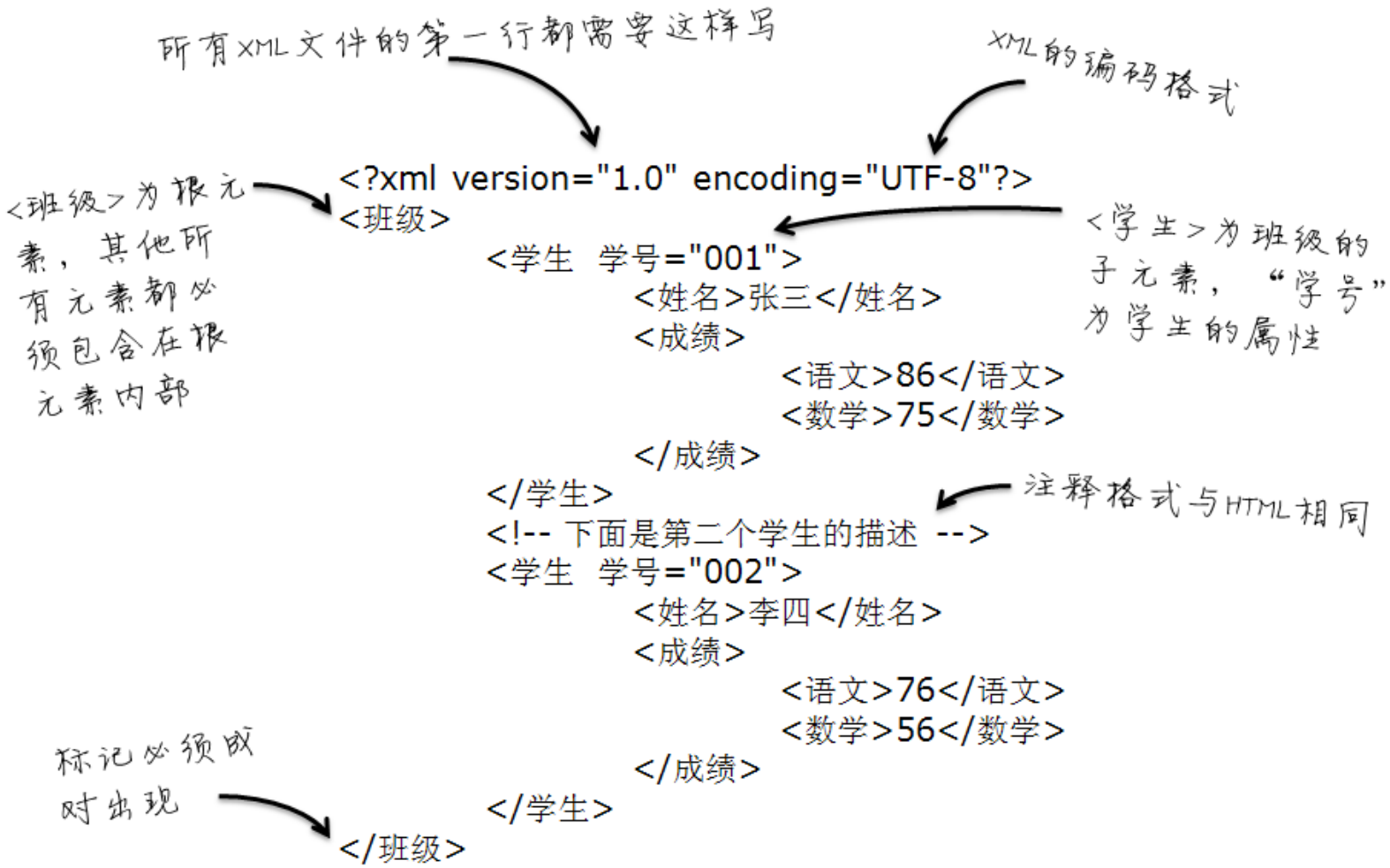
凯盛软件

- XML 不是 HTML 的替代。
- XML 和 HTML 为不同的目的而设计：
- XML 被设计为传输和存储数据，其焦点是数据的内容。
- HTML 被设计用来显示数据，其焦点是数据的外观。
- HTML 旨在显示信息，而 XML 旨在传输信息。

跨语言间传输数据



一个XML文档


所有XML文件的第一行都需要这样写
XML的编码格式
<班级>为根元素，其他所有元素都必须包含在根元素内部
<学生>为班级的子元素，“学号”为学生的属性
注释格式与HTML相同
标记必须成对出现
<?xml version="1.0" encoding="UTF-8"?>
<班级>
<学生 学号="001">
<姓名>张三</姓名>
<成绩>
<语文>86</语文>
<数学>75</数学>
</成绩>
</学生>
<!-- 下面是第二个学生的描述 -->
<学生 学号="002">
<姓名>李四</姓名>
<成绩>
<语文>76</语文>
<数学>56</数学>
</成绩>
</学生>
</班级>

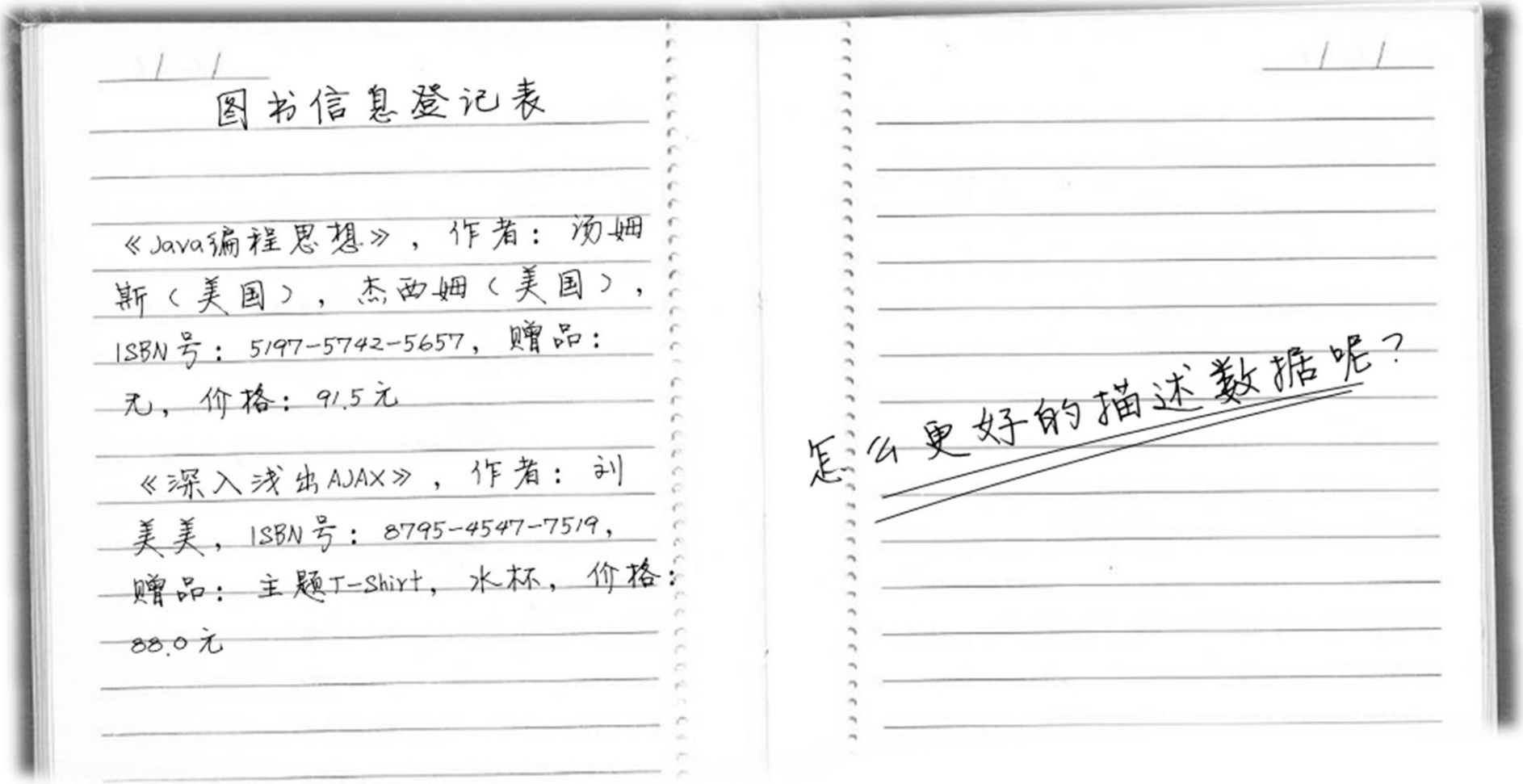
语法规则

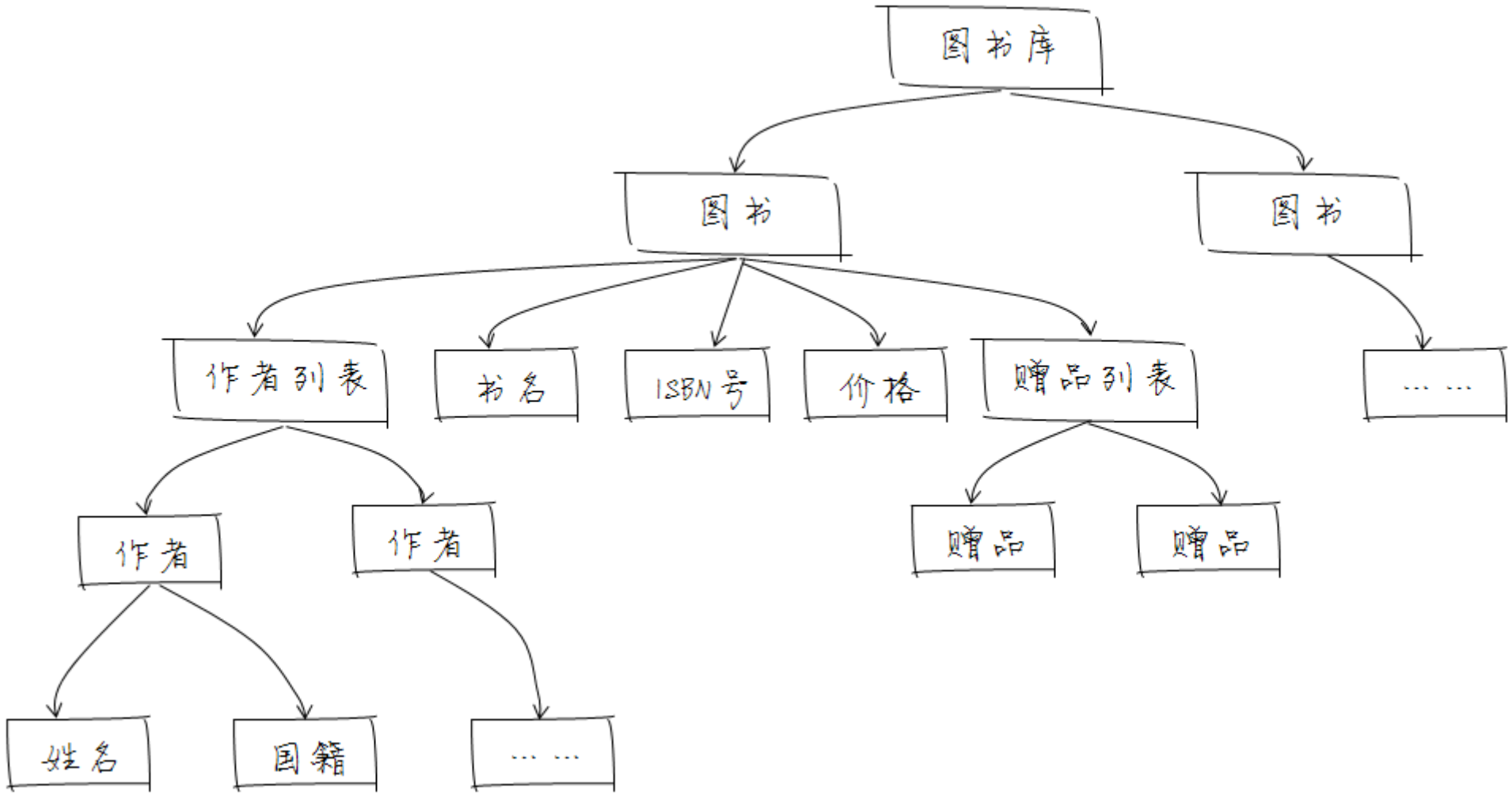
凯盛软件

- 所有 XML 元素都须有关闭标签
- XML 标签对大小写敏感
- XML 必须正确地嵌套
- XML 文档必须有根元素
- XML 的属性值须加引号
- 元素名称不能以数字和"_"（下划线）开头、不能以XML开头、不能包含空格与冒号
- HTML中的转义符在XML中也可以使用，如果文本中需要转义的字符太多，还可以使用“<![CDATA[需要转义的文本]]>”进行转义

使用XML来描述信息

凯盛软件




图书库
图书
图书
作者列表
书名
ISBN号
价格
赠品列表
……
作者
作者
赠品
赠品
姓名
国籍
……

books.xml

凯盛软件

```
<?xml version="1.0" encoding="UTF-8"?>
<books>
        <book ISBN="5197-5742-5657">
                <name>Java编程思想</name>
                <price>91.5</price>
                <authors>
                        <author>
                                <name>汤姆斯</name>
                                <nation>美国</nation>
                        </author>
                        <author>
                                <name>杰西姆</name>
                                <nation>美国</nation>
                        </author>
                </authors>
                <gift/>
        </book>
        … …
</books>
```

https://gist.github.com/fankay/5044626

服务器返回XML

凯盛软件

```
response.setContentType("text/xml");
PrintWriter out = response.getWriter();
out.println("<?xml version=\"1.0\" encoding=\"UTF-8\"?>");
out.println("<books>");
out.println("<book ISBN=\"5197-5742-5657\">");
out.println("<name>Java编程思想</name>");
out.println("<price>91.5</price>");
out.println("<authors>");
out.println("<author>");
out.println("<name>汤姆斯</name>");
out.println("<nation>美国</nation>");
out.println("</author>");
out.println("<author>");
out.println("<name>杰西姆</name>");
out.println("<nation>美国</nation>");
… … …
out.flush();
out.close();
```

Ajax接收返回的XML文档

凯盛软件

```
function callback(){
if(xmlHttp.readyState == 4){
    if(xmlHttp.status == 200)
    {
        var xmlDom = xmlHttp.responseXML;
        … …
    }
}
```

JavaScript解析XML API

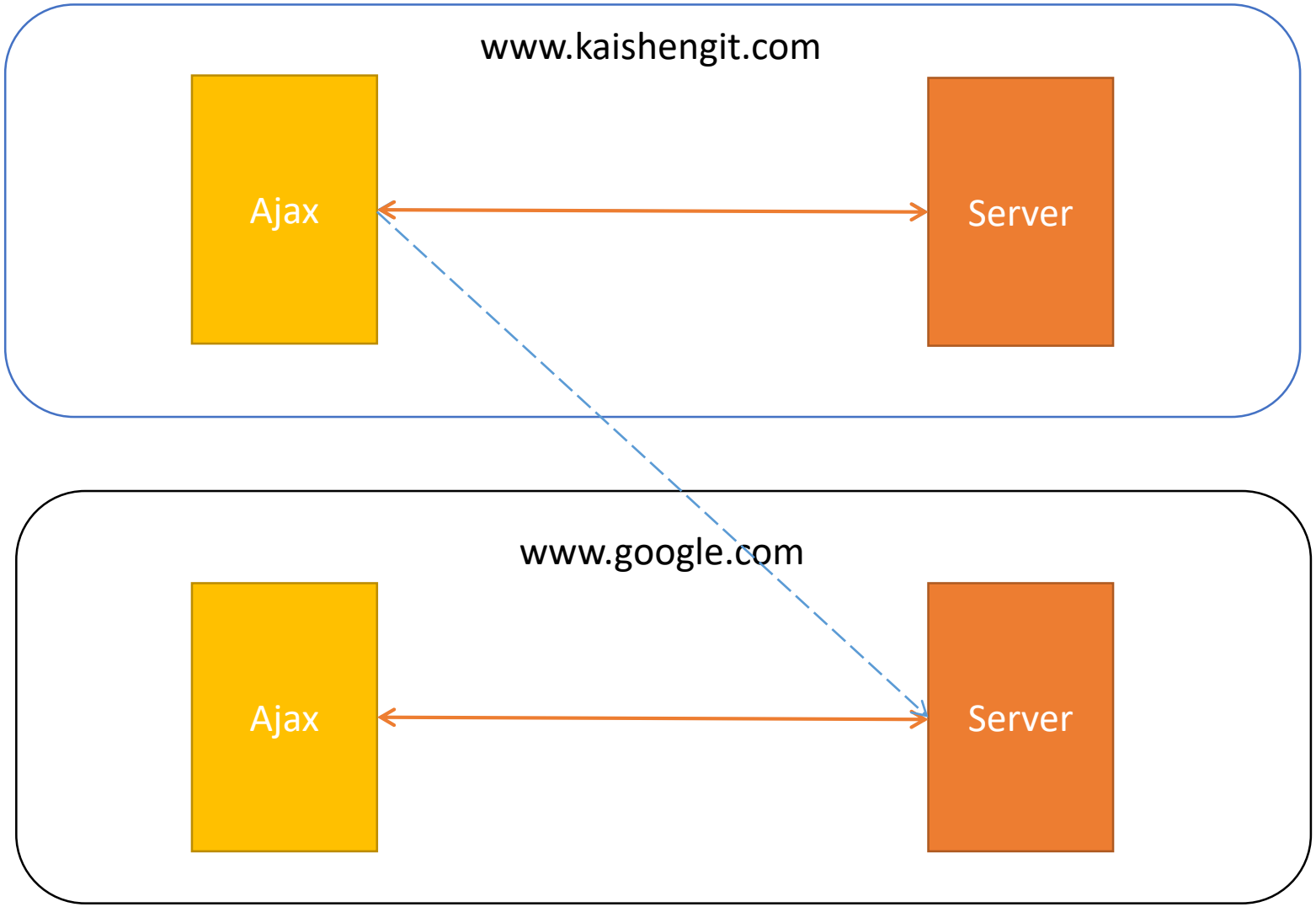
凯盛软件

名称	类型	作用
getElementsByTagName(name)	方法	返回当前元素中有指定标记名的子元素数组
childNodes	属性	返回当前元素所有子元素的数组
nodeValue	字符串	获取节点值：如果节点为元素，返回null或undefined；如果节点为文本，返回文本值
getAttribute(name)	方法	返回元素的属性值，属性有name指定

解析books.xml

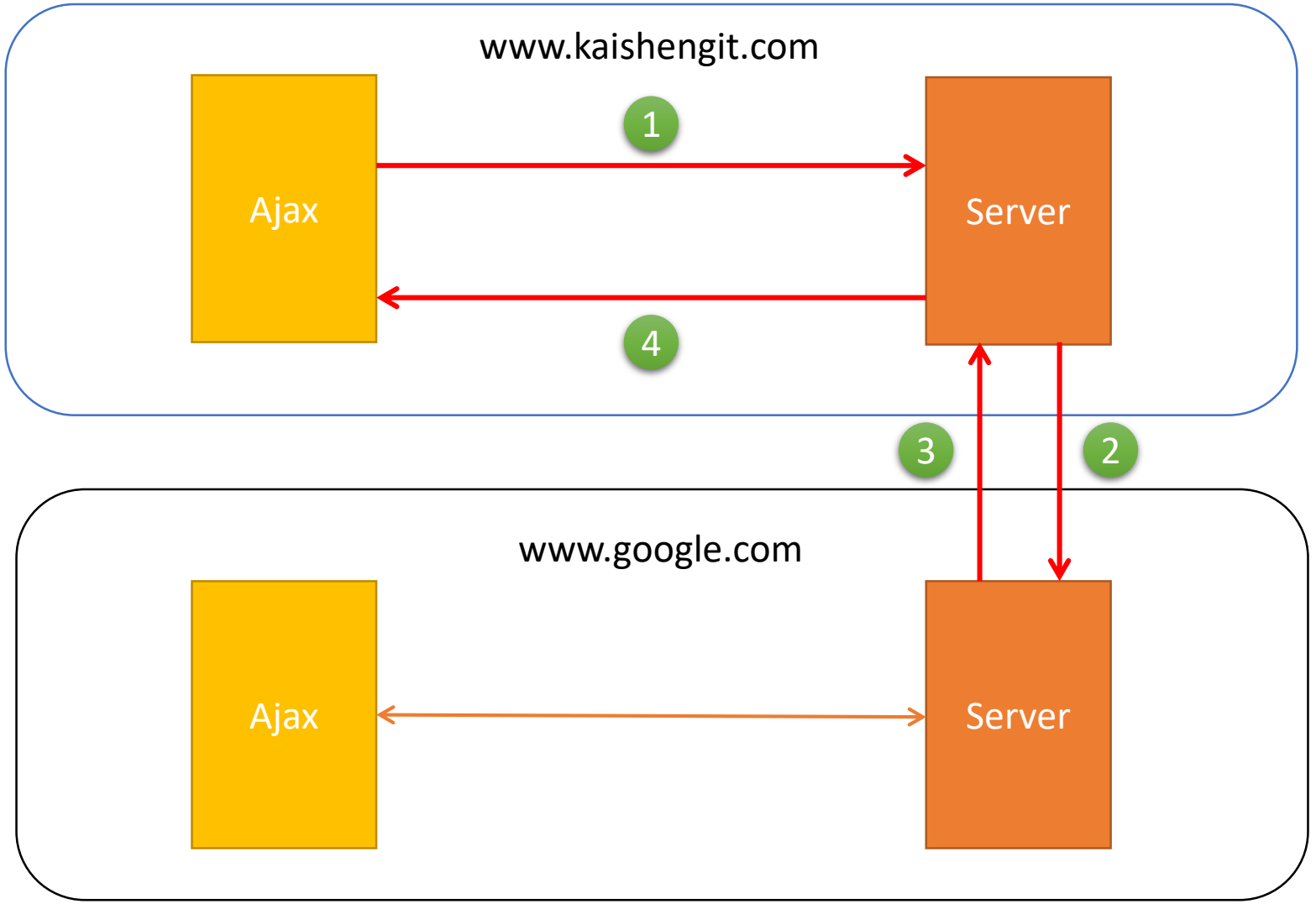

```
var books = xmlDom.getElementsByTagName("book");

alert(books.length);

for(var i = 0;i<books.length;i++){
    var isbn = books[i].getAttribute("ISBN");
    var name = books[i].getElementsByTagName("name")[0].childNodes[0].nodeValue;
    var price = books[i].getElementsByTagName("price")[0].childNodes[0].nodeValue;
    alert("name:" + name + "\tisbn:" + isbn + "\tprice:" + price);
    var authors = books[i].getElementsByTagName("authors")[0].getElementsByTagName("author");
    for(var j = 0;j<authors.length;j++)
    {
        var author = authors[j];
        var aName = author.getElementsByTagName("name")[0].childNodes[0].nodeValue;
        var native = author.getElementsByTagName("nation")[0].childNodes[0].nodeValue;
        alert("Authro Name:" + aName + "\tNation:" + native);
    }

    var gift = books[i].getElementsByTagName("gift")[0];
    var items = gift.getElementsByTagName("item");
    for(var k = 0;k < items.length;k++)
    {
        alert(items[k].childNodes[0].nodeValue);
    }
}
```

跨域



使用代理模式进行跨域操作

凯盛软件



API

凯盛软件

有道翻译API
http://fanyi.youdao.com/openapi?path=data-mode

API key : 1587754017
keyfrom : kaishengit

Apache HttpComponents

凯盛软件

http://hc.apache.org/

https://gist.github.com/fankay/5045142
http://www.mkyong.com/java/apache-httpclient-examples/

GET


```
CloseableHttpClient httpClient = HttpClients.createDefault();

HttpGet httpGet = new HttpGet("http://www.kaishengit.com");
HttpResponse resp = httpClient.execute(httpGet);

int httpCode = resp.getStatusLine().getStatusCode();
if(httpCode == 200) {
    InputStream in = resp.getEntity().getContent();
    String result = IOUtils.toString(in);

    System.out.println(result);
} else {
    System.out.println("请求异常：" + httpCode);
}

httpClient.close();
```

POST


```
CloseableHttpClient httpClient = HttpClients.createDefault();

HttpPost post = new HttpPost("http://192.168.0.111:8080/login.do");
List<NameValuePair> list = new ArrayList<NameValuePair>();
list.add(new BasicNameValuePair("name","tom"));
list.add(new BasicNameValuePair("password","123123"));

post.setEntity(new UrlEncodedFormEntity(list));

HttpResponse response = httpClient.execute(post);

httpClient.close();
```

JSON

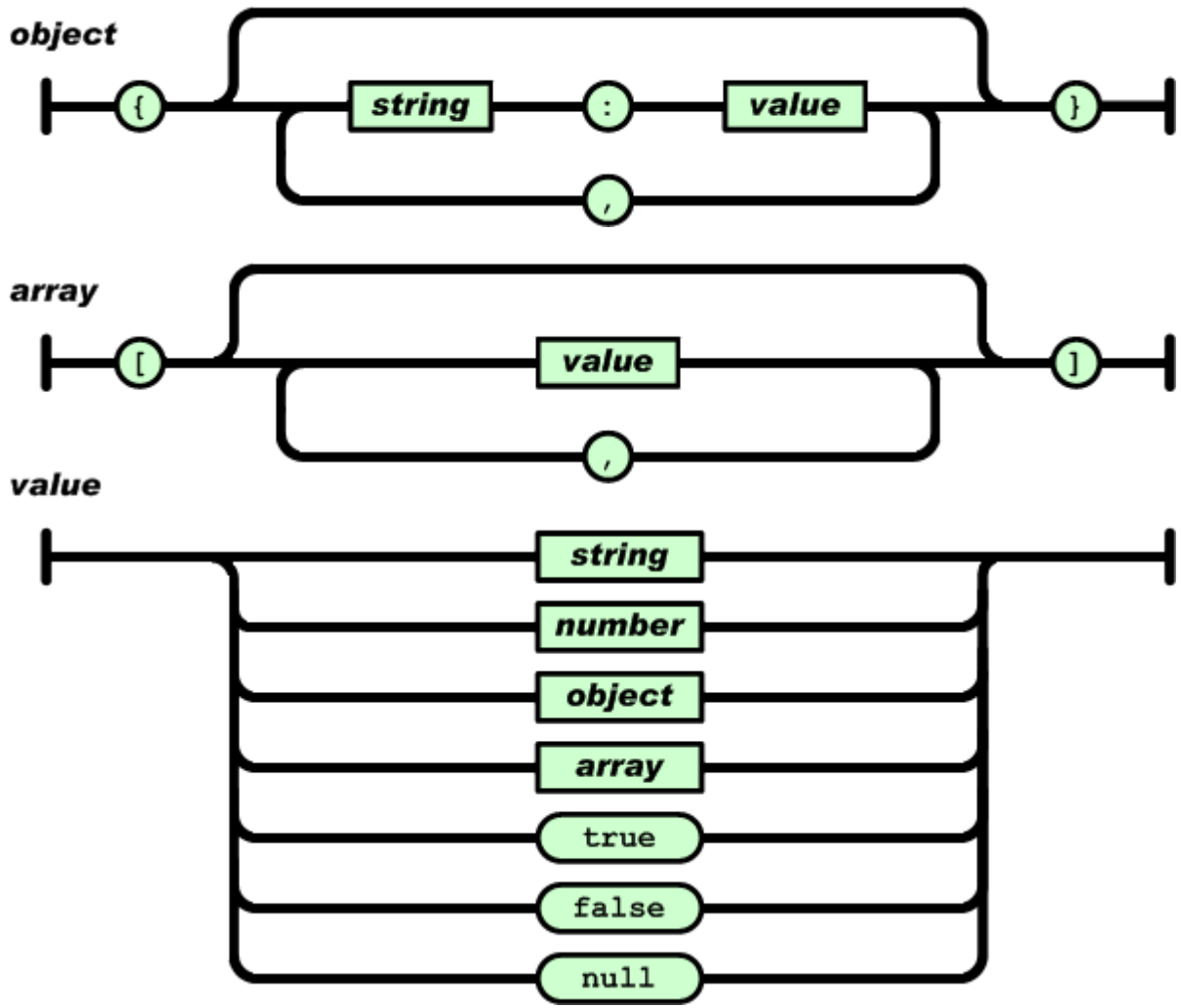
凯盛软件

JSON的全称是JavaScript Object Notation（即JavaScript对象标识），实际上是通过组合使用JavaScript中的数组与键值对（hash）对象来描述数据的结构。

JSON中两种结构：

1. 数组用来表示有序结构
2. 键值对用来表示对应关系

http://json.org/json-zh.html


object
{
string
:
value
}
,
array
[
value
]
,
value
string
number
object
array
true
false
null

JSON示例

凯盛软件

```
var json = {'陕西' : ['西安', '宝鸡', '汉中'], '山东' : ['青岛', '济南']};
```

键　　值

城市使用有序列表　　详细信息使用键值对

```
var json = [{'name' : '西安市', 'people' : '387万', 'area' : '9871', 'places' : ['兵马俑
', '华清池', '骊山', '钟楼']},
{'name' : '宝鸡市', 'people' : '127万', 'area' : '2723', 'places' : ['法门寺', '太白山']}];
```

名胜使用有序列表

JSON类库

- google-gson

Google-gson

凯盛软件

```
Gson gson = new Gson();
String json = gson.toJson(object);
```

Servlet输出JSON

凯盛软件

```
response.setContentType("application/json");

Gson gson = new Gson();
String json = gson.toJson(obj);

PrintWriter out = response.getWriter();
out.print(json);
out.flush();
out.close();
```

JavaScript解析JSON

凯盛软件

```
var json = {"fid":3,"id":21,"typeName":"计算机类"};
alert(json.id);
alert(json.typeName);
```

转换


```
var json = JSON.parse(xmlHttp.responseText);

for(var i = 0;i<json.length;i++){
    var js = json[i];
    var id = js.id;
    var title = js.title;
    var typeid = js.typeid;

    … …
```

JQuery和Ajax

凯盛软件

```
$.ajax({
   type: "POST",
   url: "some.do",
   data: "name=John&location=Boston",
   success: function(msg){
     alert( "Data Saved: " + msg );
   }
});
```

Post

凯盛软件

```
$.post("ajax.do",{name:n},function(date){
        alert(date);
});
```

Get

凯盛软件

```
$.get("ajax.do",{name:n},function(date){
        alert(date);
});
```

使用JQuery操作JSON

凯盛软件

```
$.getJSON("json.do",{id:v},function(json){
    $(json).each(function(){
        var id = this.id;
        var typeId = this.typeid;
        var title = this.title;

        … …
```

使用JQuery操作XML

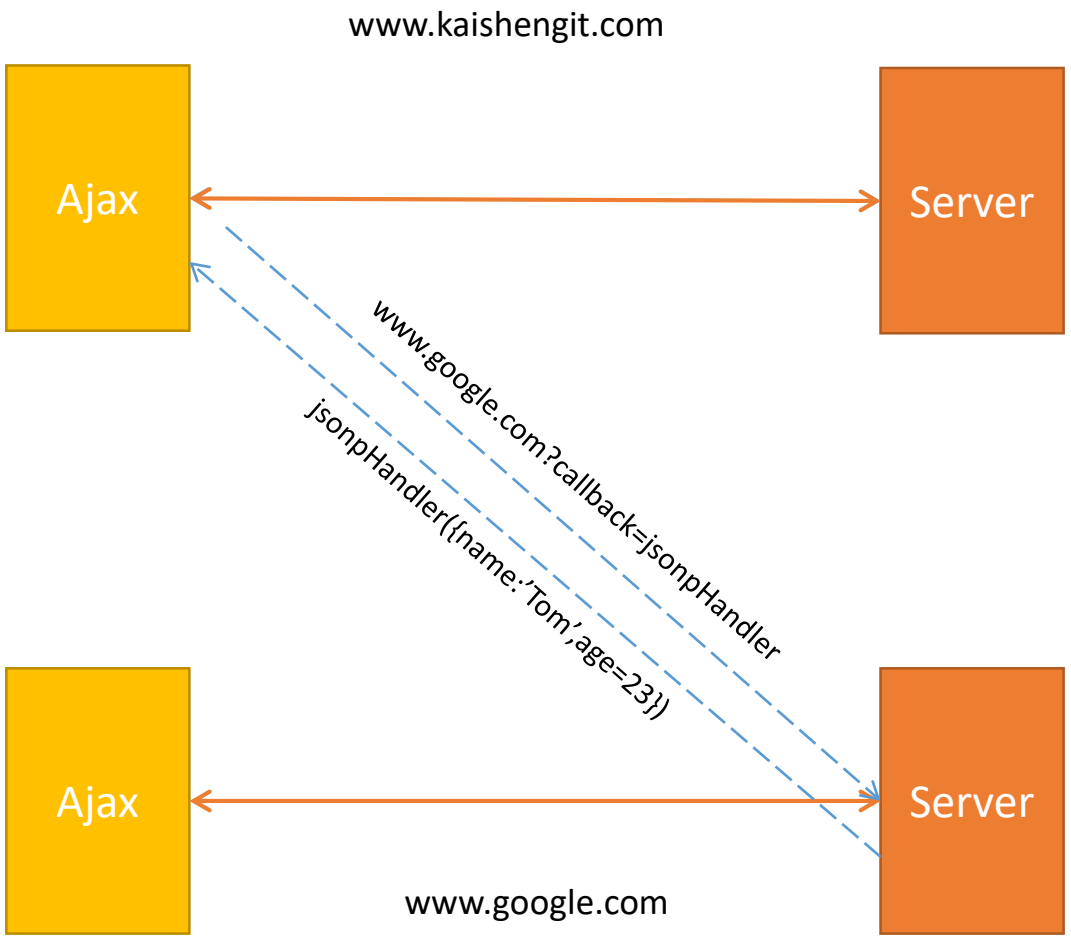
凯盛软件

```
$.get("type.do",{id:selectId},function(xml){

    $(xml).find("type").each(function(){

        var id = $(this).attr("id");
        var fid = $(this).attr("fid");
        var title = $(this).text();

    });

});
```

JSONP

凯盛软件

JSONP（JSON with Padding），用来解决跨域问题的另一种解决方案，需要服务器端的支持。



http://zh.wikipedia.org/wiki/JSONP

客户端


```
<script type="text/javascript">
function jsonpHandler(data) {
        alert(data.name);
}
</script>
<script type="text/javascript" src="jsonp.jspx?callback=jsonpHandler"></script>
```

服务端

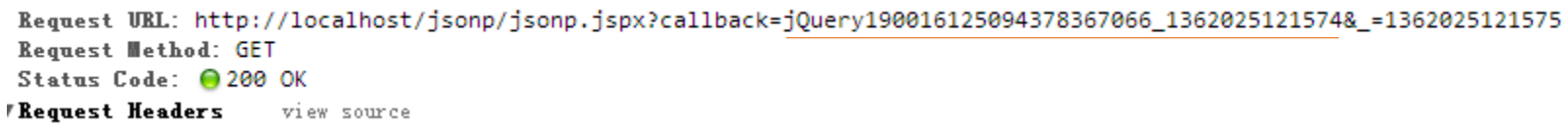

```
String callback = request.getParameter("callback");


response.setContentType("application/json;charset=UTF-8");
PrintWriter out = response.getWriter();
out.print(callback+"({name:'fankai',age:12})");
out.flush();
out.close();
```

JQuery对JSONP的支持

凯盛软件

```
$("#btn").click(function(){
    $.getJSON("jsonp.jspx?callback=?",function(data){
        alert(data.name);
    });
});
```

JQuery会自动将callback后面的问号替换成生成的函数名称



调用有道词典JSONP API

凯盛软件

```
$("#btn").click(function(){
    var t = $("#txt").val();

    $.getJSON("http://fanyi.youdao.com/openapi.do?keyfrom=kaishengit&key=158775401
    7&type=data&doctype=jsonp&callback=?&version=1.1&q="+t,function(data){
        alert(data.basic.explains);
    });
});
```

案例

凯盛软件

动态获取服务器中的数据

案例

Ajax文件上传

http://fex.baidu.com/webuploader/