

基于 Ajax 和 PHP 数据分页的实现^①

刘红坤

(河北省保定市气象局, 保定 071000)

摘要: Web 系统经常会遇到大量的数据分页显示问题, 传统的分页技术在提取数据时用户等待时间较长, 利用 Ajax 技术结合 PHP 语言开发的 Web 系统在实现大量数据分页显示时, 不需重载整个页面, 通过更新局部数据即可, 这样不但减少了用户等待时间, 而且提高了查询效率。下面结合实例谈谈数据的动态分页显示的实现问题。

关键词: 大量数据; 分页; Ajax; PHP; 异步; XMLHttpRequest

Implementation of Data Paging Based on Ajax and PHP

LIU Hong-Kun

(Baoding Meteorological Office, Baoding 071000, China)

Abstract: The question of substantial data paging display is often occurred in Web system. Traditional paging technique causes the user waiting long in extracting data. The Web system is been developed in Ajax technology and PHP when it realizes substantial data paging display. It doesn't need to reload the whole page. It needs to update the part data. It reduces the user's waiting time, and increases query efficiency. The implementing question of will be discussed dynamic data paging display through a example.

Key words: substantial data; paging; Ajax; PHP; Asynchronous; XMLHttpRequest

在 web 应用中, 数据查询非常普遍。如果检索结果数据量过大, 用户浏览会很方便。分批从数据库中提取数据, 分页显示给用户, 将加快锁定目标, 提高反应速度, 因此分页显示很有必要。传统的 web 应用模式采用同步交互方式, 需要重载整个页面才能获得分页显示结果, 导致用户等待时间较长。Ajax 技术采用异步通信方式, 只更新所需数据, 其它页面元素保持不动, 不再刷新, 不用等待即可获得所需数据。这种新型的分页显示方式克服了传统分页显示极为缓慢的弊端, 在实际中得到用户的广泛青睐。

1 Ajax简介

1.1 Ajax 定义

Ajax^[1]是 Asynchronous JavaScript and XML(异步 JavaScript 和 XML)的缩写, 它不是一项新的技术, 而

是多种技术的组合体。每种技术都有其独特之特征, 多种技术组合在一起形成一种功能强大的技术。Ajax 由以下内容组成。

- ① 基于 Web 标准的 XHTML 可扩展标识语言和 CSS 进行表示。
- ② 使用 DOM 进行动态显示及交互。
- ③ 使用 XML 和 XSLT 进行数据交换及相关操作。
- ④ 使用 XMLHttpRequest 对象进行异步数据查询及检索。
- ⑤ 使用 JavaScript 脚本语言将所有的数据绑定在一起。

1.2 Ajax 工作原理

传统的 web 开发与服务器交互采用同步的方式来实现, 其工作过程如图 1 所示。客户端的用户向服务器发送 HTTP 请求, 服务器接收请求并进行分析处理, 再访问数据库系统, 最后将处理结果返回客户端。在

① 收稿时间:2011-06-07;收到修改稿时间:2011-07-14

服务器处理期间, 客户端用户处于等待响应的状态, 且用户每次请求得到的页面都是全新的页面, 这样对于用户来说等待时间较长, 对带宽资源也是一种浪费。Ajax 技术克服了以上不足, 它采用异步交互方式, 其关键技术特征在于不刷新浏览器就能满足用户请求操作。当 JavaScript 获取客户端请求后, 将请求交给 Ajax 引擎^[2], 由 Ajax 核心对象 XMLHttpRequest 把请求异步发送到服务器, 服务器端对请求进行处理后把结果交由 XMLHttpRequest 返回。在此期间客户不必等待, 可继续进行前台的其它操作。其工作过程如图 2 所示。调用 XMLHttpRequest 对象实现与服务器的异步通讯, 相当于在用户和服务器之间加了一个中间层^[1,3], 使用户操作与服务器响应异步化, 但并不是所有的用户请求都提交给服务器, 像一些数据验证和数据处理等由 Ajax 引擎来独立完成, 只有确定需要从服务器读取新数据时再由 Ajax 引擎代为向服务器提交请求。由此可将服务器负担的一些工作转嫁到客户端, 利用客户端闲置的处理能力来处理, 减轻了服务器和带宽的负担。

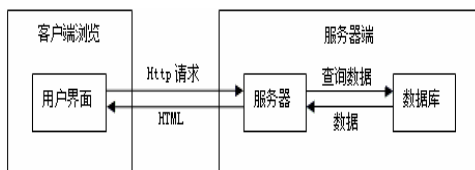


图 1 传统的 Web 应用模型

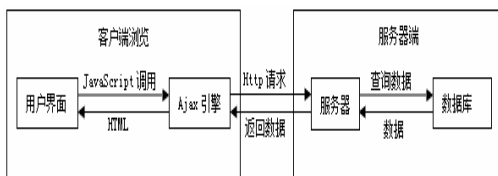


图 2 Ajax 的 Web 应用模型

2 分页技术

分页技术是指 Web 网站针对用户的大批量数据进行查询请求时, 将用户所需数据分批取出, 传送至客户浏览器的技术。分页的目的可减少网络的流量和客户端的负荷, 让用户有更好的使用体会^[4]。传统的 Web 分页技术, 客户端向服务器提出请求页数, 服务器获取此信息并根据查询条件向数据库进行查询, 并将查询结果保存到结果集, 最后再将结果集中的数据返回客户端。返回结果中不仅包含符合条件的数据, 还包含了所有 HTML 元素。客户端更新一次数据都会刷新

整个页面, 造成网页打开速度慢, 等待时间长, 占用资源多。由此可见, 为了显示部分数据而加载整个页面的数据, 没有必要。文中所提出的 Ajax 技术的引入, 能够解决不用重载整个页面, 即可更新局部数据的问题。服务器端只需传送所需的数据, 客户端只更新局部数据内容, 与数据无关的 HTML 元素保持不变, 大大提高了数据显示速度。

3 基于 Ajax 和 PHP 数据分页设计与实现

利用 IP 地址管理系统的子功能即浏览所有 IP 地址的功能为例, 来说明实现数据动态分页的具体思路与方法。

3.1 客户端利用 Ajax 程序实现页面分页显示

1) 创建 XMLHttpRequest 对象

.....

```
if(window.XMLHttpRequest){
var xmlhttp = new XMLHttpRequest();
} else if(window.ActiveXObject) {
```

```
Var xmlhttp=new ActiveXObject("Microsoft. XMLHttpRequest"); //IE 浏览器创建 XMLHttpRequest 对象
}
```

.....

不同的浏览器创建方法不同, 根据类型对其进行相应的初始化, 成功创建异步对象 xmlhttp, 最后返回结果。

2) 客户端通过 XMLHttpRequest 对象向服务器发送分页请求

.....

```
var showpage="page="+p;
xmlhttp.open("post", "showallip.php", true); //建立一个请求, 以 POST 方式发送
xmlhttp.setRequestHeader("Content-Type",
"application/x-www-form-urlencoded");// 修改 MIME 类别
```

```
xmlhttp.send(showpage); //向服务器发送请求信息
.....
```

XMLHttpRequest 对象 open() 方法的第一个参数是 http 请求方法, 可为 GET 或 POST 发送请求; 第二个参数是向目标 URL 地址提交请求; 第三个参数指定 http 请求方式, 值为 true 或者 false。true 则为异步传送方式, false 为同步传送方式。

XMLHttpRequest 对象 send()方法的参数是客户端发向服务器的请求实体,可为 XML 格式文档,也可为字符串。如果以 POST 方式发送,可为任何要传给服务器的内容。但必须调用 xmlrequest. SetRequest Header()修改 MIME 类别。

3) 处理服务器返回的信息

```
.....
xmlrequest.onreadystatechange = function(){
if(xmlrequest.readyState == 4){
if(xmlrequest.status==200){
//处理信息并更新页面
}
}else{
//信息没返回,请求异常
}
}
.....
```

请求状态改变时 XMLHttpRequest 对象调用 onreadystatechange 事件,指定函数来接收服务器返回的数据,根据需求动态更新页面。处理响应前,应先检查 readyState 的值和 HTTP 的状态,请求完成时 readyState 的值为 4,HTTP 状态响应完成时 status 为 200。

3.1 服务器端利用 PHP 程序(showallip.php)根据分页要求将相关数据库记录返回客户端

代码如下:

```
.....
include("conn_two/conntwo.php");//包含连接数据库的操作文件
$pagesize=10;//设置每页显示的记录数
$sql="Select count(DISTINCT twoid) from tb_twolist";//SQL 查询语句
$result = mysql_query($sql,$conntwo);//进行查询,获取数据库总的记录数
$allpage = mysql_fetch_array($result);//总的数据库记录数
$numpage=$allpage[0];
$fenpage=intval($numpage/$pagesize);//计算分页数目
if ($numpage%$pagesize)
$pages++;
if (isset($_POST['page'])){//获取客户端传递页码数
$page=intval($_POST['page']);
}
else{
$page=1; //设置为第一页
}
}
```

```
$first=1;
$prev=$page-1;//上一页
$next=$page+1;//下一页
$last=$pages;//尾页
$offset=$pagesize*($page - 1);//计算记录偏移量
$result=mysql_query("select*from tb_twolist GRO- UP
BY `twoid` order by twoid desc limit $offset, $pa-
gesize",$conntwo);//读取指定记录数
.....
echo "第".$page."页/总".$pages."页";
if ($page>1)echo"<aonclick=\"lookpage(\".$first. \"\")\"
href=#>首页</a>|";
if($page>1)echo"<aonclick=\"lookpage(\".$pre v.\"))\"
href=#>上页</a>|";
if($page<$pages)echo"<aonclick=\"lookpage(\".$next.\"))\"h
ref=#>下页</a>|";
if($page<$pages)echo"<aonclick=\"lookpage(\".$last.\"))\"
href=#>尾页</a>";
.....
```

上述代码,首先获取数据库中总的记录数,根据总记录数计算出可以分几页。客户端获取页码计算的指针偏移量,创建 SQL 语句,得到指定页码要显示的记录,获取记录集,输出结果。

3 结语

本文分析了采用传统的数据分页方式在处理大量数据分页显示时查询效率低的突出缺陷,给出了采用 Ajax 和 PHP 技术开发的快速分页显示大量数据的新思路,经过在 IP 地址管理系统中的实际应用,发现此方法有效减少了用户交互过程中的等待时间,降低了通信流量,节约了网络资源,提高了系统的响应速度,同时因客户端分担了部分工作,从而减轻了服务器的压力。由此可见,基于 Ajax 和 PHP 开发的 Web 分页显示程序高效、实用,适合在 Web 系统开发中推广应用。

参考文献

- 1 潘凯华,邹天思,等.PHP 开发实战宝典.北京:清华大学出版社,2010.452-453.
- 2 游丽贞,郭宇春,李纯喜.AJAX 引擎的原理和应用.微计算机信息,2006,22(2-3):205-207.
- 3 毕振颇,刘志勤.AJAX 在基于 NC 机网络中的应用.计算机工程,2007,33(24):124-125.
- 4 顾志峰,李涓子.Web 应用程序分页策略的研究.计算机工程,2005,31(21):60-62.