

探析超文本传输协议

朱永红, 张燕平, 许荣斌, 谢莹

(安徽大学 计算智能与信号处理教育部重点实验室, 安徽 合肥 230039)

摘要: HTTP(超文本传输协议)是浏览器和 Web 服务器共同遵守的协议。介绍了 HTTP/1.1 协议的特点如灵活性、简单性、无连接、无状态等。分析了 HTTP 的工作过程, 分连接、请求、响应、关闭四个步骤, 并且通过具体的实例来对 HTTP 工作过程中的请求和应答两部分作了详细的描述, 最后分析了当前 HTTP 协议的缺点并指出了 HTTP 协议的发展趋势。

关键词: HTTP; 状态代码; HTTP 请求; HTTP 应答

中图分类号: TP312

文献标识码: A

文章编号: 1673-629X(2006)10-0065-03

Exploration on HyperText Transfer Protocol

ZHU Yong-hong, ZHANG Yan-ping, XU Rong-bin, XIE Ying

(Ministry of Education Key Lab. of Intelligence Computing and Signal Processing, Anhui Univ., Hefei 230039, China)

Abstract: HTTP is the protocol used in browser and Web server. This paper first introduces the features of HTTP/1.1, such as flexibility, simplicity, no connection and no state. Then analyzes the process of HTTP, which is composed by connect, request, response and shut, and describes the process of request and response through the specific examples in detail. Last analyzes the shortcomings of the current HTTP and points out how it will develop.

Key words: HTTP; status code; HTTP request; HTTP response

0 引言

HTML 是 Web 出版的基础。超文本传输协议(HTTP, HyperText Transfer Protocol)使用超文本标记语言(HTML), 使 Web 页面可以通过浏览器的客户端软件观看。浏览器利用文本、图形、多媒体传送信息, 而且定义了一种方法, 使文档之间可以通过超级链接进行跳转, 用户可以单击一个单词或图片来得到与某一主题相关的信息, 这一简单的创造是推动 Internet 发展的重要动力。下面先讨论 HTTP 协议的主要特点。

1 HTTP 协议的主要特点

Internet 的基本协议是 TCP-IP 协议, 目前, 广泛使用的 HTTP, FTP 协议是 TCP-IP 应用层协议, 不同的协议对应着不同的应用。HTTP 协议使用可靠的 TCP 连接, 默认端口是 80。HTTP 的第一个版本是 HTTP/0.9, 后来发展到了 HTTP/1.0, 现在最新的版本是 HTTP/1.1^[1]。其主要特点可概括如下:

1) 灵活性。

HTTP 允许传输各种类型的数据对象, Content-Type 就是标示正在传输的数据类型。如果把数据看成是在“罐”里的东西, 那么 Content-Type 是贴在罐上的标签, 它告诉人们里面装的是什么东西。

2) 简单。

HTTP 本身既简单, 又能有效地处理大量请求。当客户机与服务器连接后, 客户机必须传送的信息只是请求方法和路径。所以就使得 HTTP 服务器程序规模小, 而且简单。因此经 HTTP 的通信速度很快, 与其它协议相比, 时间开销要小得多。

3) 无连接。

HTTP 是一个无连接协议, 就是说每次连接只处理一个请求, 客户机与服务器连接后提交一个请求, 在客户机接到应答后马上断开连接。使用这种无连接协议, 在没有请求时, 服务器不会在那里闲等着, 服务器更不会在完成一个请求后还把着原来的请求不放。

4) 无状态。

HTTP 是无状态^[2]的协议, 它既是优点也是缺点。一方面, 由于没有状态, 协议对事物处理没有记忆能力, 如果后续事物处理需要前面处理的有关信息, 那么这些信息必须在协议外面保存。缺少状态意味着所需要的前面信息必须重现, 势必导致每次连接要传送较多的信息。另一方面, 也正是由于缺少状态使得 HTTP 累赘少, 运行速度快, 服务器应答快。

收稿日期: 2006-02-11

基金项目: 国家自然科学基金资助项目(60475017); 教育部博士点基金资助项目(20040357002)

作者简介: 朱永红(1981-), 男, 安徽旌德人, 硕士研究生, 研究方向为智能计算; 张燕平, 教授, 硕导, 研究方向为人工神经网络、机器学习、人工智能在金融工程中的应用。

2 HTTP 工作过程

HTTP 是基于客户机/服务器模式的,而且是面向连接的。典型的 HTTP 事物处理有以下过程:

(1) 连接:客户机与服务器建立连接。

建立一次连接的过程是这样的,客户打开一个套接字(Socket)并把它约束在一个端口上。打开一个套接字也就是建立一个虚拟文件,当向文件上写完数据后,数据在网络上传输。在这之前,HTTP 服务器已在运行,监听某个端口(通常是 80),等待连接的建立。

(2) 请求:客户机向服务器提出请求。

连接建立以后,客户就可以发出请求将请求数据发送到服务器指定的端口,而 HTTP/0.9 只支持很简单的客户请求,例如:GET URI。其中 GET 是方法,用于从服务器请求一个由 URI 标识的资源对象。如果对象是文档或文件,GET 将请求其内容,如果对象是程序或脚本,GET 将请求程序的运行结果或脚本的输出。每当循着超文本链浏览 Web 时,浏览器都要用 GET 方法来获取你请求的资源。

(3) 响应:服务器接受请求,并根据请求返回相应的文件作为应答。

当服务器接收到浏览器发出的请求时,搜索客户所需资源并作出响应。

(4) 关闭:客户机与服务器断开连接。

HTTP 的工作过程如图 1 所示。HTTP 的这种简单特性,对于在小型、低速的桌面计算机上实行客户/服务器系统来说是快捷有效的。

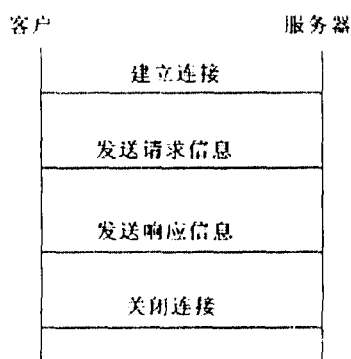


图 1 HTTP 协议工作过程

3 HTTP 请求

HTTP 请求由 3 个部分构成,分别是:方法 - URI - 协议/版本^[3],请求头,请求正文。下面是一个 HTTP 请求的例子:

```
GET /servlet/default.jsp HTTP/1.1
```

```
Accept: text/html
```

```
Accept-Language: en-gb
```

```
Connecion: Keep-Alive
```

```
Host: localhost
```

```
Referer: http://localhost/ch8/SendDetails.htm
```

```
User-Agent: Mozilla/4.0 (compatible; MSIE 4.01; Windows
```

```
98)
```

```
Content-Length: 33
```

```
Content-Type: application/x-www-form-urlencoded
```

```
Accept-Encoding: gzip
```

```
userName=JavaJava&userID=javaID
```

请求的第一行是“方法 - URI - 协议/版本”,其中 GET 就是请求方法,/servlet/default.jsp 表示 URI,HTTP/1.1 是协议和协议的版本。根据 HTTP 标准,HTTP 请求可以使用多种请求方法。例如,HTTP 1.1 支持 7 种请求方法^[4,5]:GET,POST,HEAD,OPTIONS,PUT,DELETE,和 TRACE。在 Internet 应用中,最常用的请求方法是 GET 和 POST。

①GET:该方法就是以实体方式得到请求 URI 指定资源的信息。如果请求 URI 只是一个数据的产生过程,那么最终要在回应实体中返回的是由该处理过程的结果所指向的资源,而不是返回该处理过程的描述文字,除非那段文字恰好是处理的输出。如果请求消息包含 If-Modified-Since 标题域,GET 方法的语法就变成“条件 GET”。条件 GET 方法可以对指定资源进行判断,如果它在 If-Modified-Since 标题域中指定日期后发生了更新,才启动传输,否则不传输。这种条件 GET 允许被缓存的实体在不必经过多次请求或不必要的数据传输就能进行刷新,从而有助于降低网络负载。

②POST:当客户向服务器传送大量数据,要求服务器和 CGI 程序做进一步处理时要使用 POST 方法,如发送 HTML,FORM 内容,让 CGI 程序进行处理。这时 FORM 内容的 URL 编码将随请求一起发出。

URI 完整地指定了要访问的网络资源,通常认为它相对于服务器的根目录而言,因此总是以“/”开头。URL 实际上是 URI 一种类型。最后,协议版本声明了通信过程中使用的 HTTP 协议的版本。

请求头包含许多有关客户端环境和请求正文的有用信息。例如,请求头可以声明浏览器所用的语言,请求正文的长度,等等,它们之间用一个回车换行符号(CRLF)分隔。在请求头和请求正文之间是一个空行,这个行非常重要,它表示请求头已经结束,接下来的就是请求的正文。

4 HTTP 应答

HTTP 应答^[6]也由 3 个部分构成,分别是:协议 - 状态代码 - 描述,应答头,应答正文。在请求和应答过程中都带有版本号,用来帮助客户程序与服务器协商使用协议类型。状态代码是一个三位的十进制数,用来表示服务器对请求的处理结果。描述以文字形式详细解释了响应代码的含义。状态代码的分类方法如下:

(1)消息 1xx。该类状态码不使用,保留以备将来使用。

(2)成功 2xx。表示客户端请求被成功接收、理解、接受。

(3)重定向 3xx。该类状态码表示用户代理要想完成请求,还需要采取进一步操作以完成请求。

(4)客户端错误 4xx。请求包含错误的语法或无服务器故障的情况下不能实现。

(5)服务器错误 5xx。服务器不能实现一个明显有效的请求。

下面是一个 HTTP 应答的例子:

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/4.0
Date: Mon, 3 Jan 1998 17:24:19 GMT
Content-Type: text/html
Last-Modified: Mon, 10 Jan 1998 13:23:42 GMT
Content-Length: 118
<html>
<head>
<title>HTTP 应答示例</title></head><body>
Hello HTTP!
</body>
</html>
```

HTTP 应答的第一行类似于 HTTP 请求的第一行,它表示通信所用的协议是 HTTP 1.1,服务器已经成功地处理了客户端发出的请求(200 表示成功),一切顺利。

应答头也和请求头一样包含许多有用的信息,例如服务器类型、日期时间、内容类型和长度等。应答的正文就是服务器返回的 HTML 页面。应答头和应答正文之间也用 CRLF 分隔。

(上接第 64 页)

```
<class name="net.xiaoyu.model.Book" table="user">
  <id name="id" column="id" unsaved-value="0">
    <generator class="increment"/>
  </id>
  <property name="bookname" column="bookname" not-null="true"/>
  <property name="bookauthor" column="bookauthor" not-null="true"/>
  .....
  <!-- 定义借阅到用户多对一关系 -->
  <many-to-one name="user" column="userid"
    class="net.xiaoyu.model.User"/>
</class>
</hibernate-mapping>
```

6 总结

首先说明了 Spring 和 Hibernate 的基本概念与架构,并通过一个实例详细描述了这两者之间的集成方法。当 Hibernate 遇到了 Spring 时,这两个强大的框架显得更加强大。目前, Spring, Hibernate 日益成为中小企业的首选组合。Hibernate 能极大地减少数据库相关操作代码,

5 结束语

当前 HTTP 协议缺乏模块化的特点,性能也比较差,而且 HTTP 协议在每次 HTTP 会话(即一个请求和回应的过程)都打开一个新的 TCP 连接,这样就对系统和网络资源造成浪费。现在正在研究的 HTTP-NG^[7](HTTP-Next Generation)很可能会取代当前的 HTTP。HTTP-NG 的最大变化是:客户机可以一次连续发送多个请求,服务器依次响应每个请求。这种方法可进一步缩短服务器响应的时间,提供更加高效优质的服务。

参考文献:

- [1] Fielding R, Gettys J, Mogul J C, et al. Hypertext Transfer Protocol-HTTP/1.1[S]. RFC 2068, 1997.
- [2] 冯延峰. ASP 技术手册[M]. 北京:中国电力出版社, 2001.
- [3] Mogul J C, Fielding R, Gettys J, et al. Use and Interpretation of HTTP Version Numbers[S]. RFC 2145, 1997.
- [4] Fielding R. Relative Uniform Resource Locators[S]. RFC 1808, 1995.
- [5] Fielding R, Gettys J, Mogul J C, et al. Hypertext Transfer Protocol, HTTP/1.1[S]. RFC 2616, 1999.
- [6] 林 炜, 蔺永华. HTTP 协议及其发展[J]. 电脑学习, 1999 (1): 2-3.
- [7] 黄宏涛, 王 慧. 下一代 HTTP 协议[J]. 计算机应用研究, 2000(1): 61-62.

能实现一个简单高效的持久化,而 Spring 通过控制反转,层层回调,极大地减少了业务逻辑代码。通过这两者的结合使得整个应用简单,松耦合,且容易测试。这些都是程序员特别关注的部分,这为他们快速稳定地开发信息系统提供了足够好的保证。

参考文献:

- [1] Walls C, Breidenbach R. Spring in Action[Z]. [s.l.]: Manning Publications Co, 2005.
- [2] Johnson R, Hoeller J. Spring 1.2.4: J2EE Application Framework Reference Documentation. Spring 官方网站[EB/OL]. <http://www.springframework.org/>, 2005.
- [3] Bauer C. Hibernate in Action[Z]. [s.l.]: Manning Publications Co, 2005.
- [4] Hibernate 3.0.2: Reference Documentation. Hibernate 官方网站[EB/OL]. <http://www.hibernate.org/>, 2005.
- [5] 田 珂, 谢世波, 方 马. J2EE 数据持久层的解决方案[J]. 计算机工程, 2003, 29(22): 93-95.
- [6] 王卫平, 王松涛, 王名著. 一种基于 J2EE、Spring 和 Hibernate 的轻量级 EAI 构架[J]. 计算机系统应用, 2005(11): 38-41.