

HTTP 协议

1 什么是 HTTP 协议

HTTP 超文本传输协议 (HTTP-Hypertext transfer protocol), 是一个属于应用层的面向对象的协议, 由于其简捷、快速的方式, 适用于分布式超媒体信息系统。它于 1990 年提出, 经过几年的使用与发展, 得到不断地完善和扩展。它是一种详细规定了浏览器和万维网服务器之间互相通信的规则, 通过因特网传送万维网文档的数据传送协议。

2 HTTP 协议的发展历程

超文本传输协议的前身是世外桃源(Xanadu)项目, 超文本的概念是泰德·纳尔森(Ted Nelson)在 1960 年代提出的。进入哈佛大学后, 纳尔森一直致力于超文本协议和该项目的研究, 但他从未公开发表过资料。1989 年, 蒂姆·伯纳斯·李(Tim Berners Lee)在 CERN(欧洲原子核研究委员会 = European Organization for Nuclear Research)担任软件咨询师的时候, 开发了一套程序, 奠定了万维网(WWW = World Wide Web)的基础。1990 年 12 月, 超文本在 CERN 首次上线。1991 年夏天, 继 Telnet 等协议之后, 超文本转移协议成为互联网诸多协议的一分子。

当时, Telnet 协议解决了一台计算机和另外一台计算机之间一对一的控制型通信的要求。邮件协议解决了一个发件人向少量人员发送信息的通信要求。文件传输协议解决一台计算机从另外一台计算机批量获取文件的通信要求, 但是它不具备一边获取文件一边显示文件或对文件进行某种处理的功能。新闻传输协议解决了一对多新闻广播的通信要求。而超文本要解决的通信要求是: 在一台计算机上获取并显示存放在多台计算机里的文本、数据、图片和其他类型的文件; 它包含两大部分: 超文本转移协议和超文本标记语言(HTML)。HTTP、HTML 以及浏览器的诞生给互联网的普及带来了飞跃。

3 HTTP 的主要特点

1. 支持客户端/服务器模式。

2. 简单快速: 客户向服务器请求服务时, 只需传送请求方法和路径。请求方法常用的有 GET、POST、HEAD 等。每种方法规定了客户与服务器联系的类型不同。由于 HTTP 协议简单, 使得 HTTP 服务器的程序规模小, 因而通信速度很快。

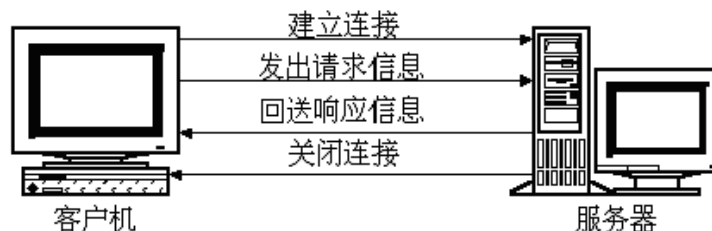
3. 灵活: HTTP 允许传输任意类型的数据对象。正在传输的类型由 Content-Type 加以标记。

4. 无连接: 无连接的含义是连接双方在完成交互后不维持连接状态, 当客户端发送新的请求后则开启新的连接。

5. 无状态: HTTP 协议是无状态协议。无状态是指协议对于事务处理没有记忆能力。缺少状态意味着如果后续处理需要前面的信息, 则它必须重传, 这样可能导致每次连接传送的数据量增大。另一方面, 在服务器不需要先前信息时它的应答就较快。

4 HTTP 协议的会话方式

4.1 浏览器与服务器之间的通信过程要经历四个步骤

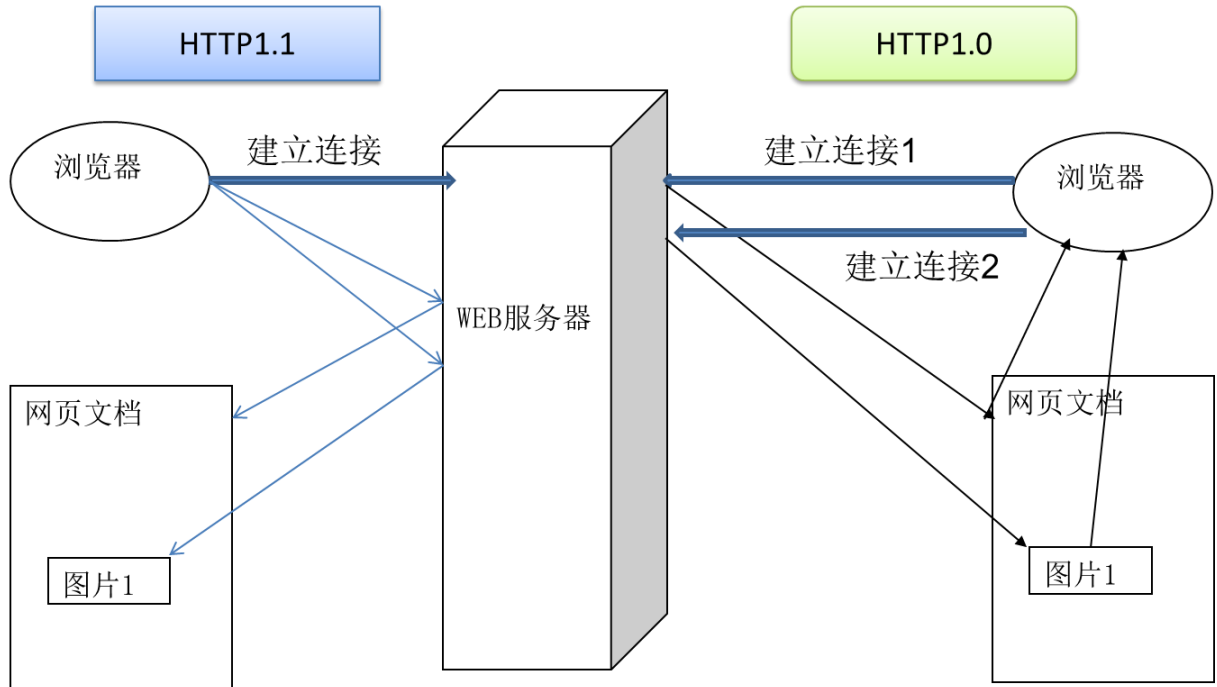


4.2 浏览器与 WEB 服务器的连接过程是短暂的, 每次连接只处理一个请求和响应。对每一个页面的访问, 浏览器与 WEB 服务器都要建立一次单独的连接。

4.3 浏览器到 WEB 服务器之间的所有通讯都是完全独立分开的请求和响应对。

5 HTTP1.0 和 HTTP1.1 的区别

●在 HTTP1.0 版本中，浏览器请求一个带有图片的网页，会由于下载图片而与服务器之间开启一个新的连接；但在 HTTP1.1 版本中，允许浏览器在拿到当前请求对应的全部资源后再断开连接，提高了效率。



6 HTTP 请求

6.1 客户端连上服务器后，向服务器发出获取某个 Web 资源的消息，称之为客户端向服务器发送了一个 HTTP 请求。一个完整的 HTTP 请求包括如下内容：

- ①请求行
- ②若干消息头(请求头)
- ③实体内容(请求体) 有可能没有

```
POST /books/java.html HTTP/1.1
Accept: */*
Accept-Language: en-us
Connection: Keep-Alive
Host: localhost
Referer: http://localhost/links.jsp
User-Agent: Mozilla/4.0
Accept-Encoding: gzip, deflate
```

←请求行

请求行用于描述客户端的请求方式、请求的资源名称，以及使用的HTTP协议版本号

←多个消息头

消息头用于描述客户端请求哪台主机，以及客户端的一些环境信息等

←一个空行

←实体内容

```
name=tom&password=123
```

6.2 请求方式

- ①HTTP 中定义了 7 种请求方式：POST、GET、HEAD、OPTIONS、DELETE、TRACE、PUT。其中最常用的是 GET 和 POST
- ②GET 请求

[1]从字面意思来说，GET 请求是用来向服务器端获取信息而发送的请求。

[2]没有特殊设置，默认情况下浏览器发送的都是 GET 请求，例如点击超链接、在浏览器地址栏直接输入地址访问。

[3]GET 请求也可以向服务器端发送请求参数，形式是在 URL 地址后面加上?，请求参数名和值用=连接，多个请求参数之间使用&分隔。例如：

GET /mail/1.html?name=abc&password=xyz HTTP/1.1

需要注意的是：GET 方式所能够携带的数据是由限制的，其数据大小通常不能超过 4K，不适于提交大量表单数据，故而在表单的提交方式中首选 POST 方式。

③POST 请求

[1]POST 请求的字面含义是向服务器端发送数据，仅在表单中设置 method="post"时，请求方式为 POST 方式[另外在 Ajax 应用中，可以指定请求方式为 POST]。

[2]POST 请求会将请求参数放在请求体中，而不是 URL 地址后面，并且发送数据的大小是没有限制的。

6.3 请求消息头

①请求消息头简称请求头，用来向服务器报告浏览器端的一些基本信息。

②常用请求头

代码	说明
Accept: text/html,image/*	我支持的数据类型
Accept-Charset: utf-8	支持的数据的编码字符集
Accept-Encoding: gzip	支持的压缩方式
Accept-Language: en-us,zh-cn	支持的语言
Host: localhost:8888	请求的主机名
Referer: http://www.it315.org/index.jsp	发送请求的界面对应的 url 防盗链
User-Agent: Mozilla/4.0	浏览器的相关信息
Connection: keep-Alive	请求完成后，到是断开呢还是连接着
Date: Tue, 11 Jul 2000 18:23:51 GMT	发送请求的时间
Cookie: tt=123	对具体客户端所做的标记

6.4 请求体

①GET 请求没有请求体

②POST 请求：如果 form 表单提交的方式为 post,则表单项的数据以请求体的形式发送给服务器，没有大小限制

7 HTTP 响应

7.1 所谓响应其实就是服务器对请求处理的结果，或者如果浏览器请求的直接就是一个静态资源的话，响应的就是这个资源本身。

7.2 HTTP 响应的组成

①响应状态行：包括协议版本、响应状态码、响应状态信息

②响应消息头：响应头

③实体内容：响应体

• 举例：

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/5.0
Date: Thu, 13 Jul 2000 05:46:53 GMT
Content-Length: 2291
Content-Type: text/html
Cache-control: private

<HTML>
<BODY>
.....
```

←状态行

←多个消息头

←一个空行

←实体内容

状态行用于描述服务器对请求的处理结果。

消息头用于描述服务器的基本信息，以及数据的描述，服务器通过这些数据的描述信息，可以通知客户端如何处理等一会儿它回送的数据。

代表服务器向客户端回送的数据

7.3 最常见的响应状态码

代码	表示	说明
200	成功	服务器已成功处理了请求。通常，这表示服务器提供了请求的网页。
302	重定向	代表让浏览器重新请求另一个资源
404	找不到	找不到请求的资源，但有时请求路径正确也返回 404 往往是由于 Web 应用有配置方面的问题，例如按照配置文件中指定的组件的全类名找不到指定的类。
500	错误	服务器内部错误，例如服务器端程序运行时抛出异常。

- 响应状态码以 2 开头的通常表示成功。
- 响应状态码以 3 开头的通常表示转移。
- 响应状态码以 4 开头的通常表示无法访问，其中包括找不到资源或没有权限等。
- 响应状态码以 5 开头的通常表示服务器端程序运行出错。

7.4 响应消息头：简称响应头

- 服务器发送给浏览器的数据，为了告诉浏览器一些情况。

代码	说明
Location: /day05/index.jsp	告诉浏览器重新定向到指定的路径
Server:apache tomcat	使用的什么 web 服务器
Content-Encoding: gzip	告诉浏览器我传给你的数据用的压缩方式
Content-Length: 80	响应体的字节数
Content-Language: zh-cn	响应体数据的语言
content-type: text/html; charset=GB2312	响应体内容的类型
Last-Modified: Tue, 11 Jul 2000 18:23:51 GMT	资源最后被修改的时间
Refresh: 1	定时刷新
Content-Disposition: attachment; filename=aaa.zip	文件下载
Set-Cookie:SS=Q0=5Lb_nQ; path=/search	将 cookie 数据回送给 ie
Expires: -1	告诉浏览器不要缓存起来
Cache-Control: no-cache	当 HTTP1.1 服务器指定 CacheControl = no-cache 时，浏览器就不会缓存该网页
Connection: close/Keep-Alive	是否保持连接
Date: Tue, 11 Jul 2000 18:23:51 GMT	响应的时间

7.5 响应体：服务器给出的响应结果的主体，通常是用来在页面上显示的。

7.6 HTTP 内容类型：当前响应体的数据类型。

①需要指出的是在浏览器和服务端之间传输的数据类型并非都是文本类型，还包括图片、视频、音频等多媒体类型。这些多媒体类型是使用 MIME 类型定义的。

②MIME 的英文全称是 "Multipurpose Internet Mail Extensions" 多功能 Internet 邮件扩充服务。MIME 类型的格式是“大类型/小类型”，并与某一种文件的扩展名相对应。

③常见的 MIME 类型

文件	MIME 类型
超文本标记语言文本	.html,.html text/html
普通文本	.txt text/plain
RTF 文本	.rtf application/rtf
GIF 图形	.gif image/gif
JPEG 图形	.jpeg,.jpg image/jpeg
au 声音文件	.au audio/basic
MIDI 音乐文件	mid,.midi audio/midi,audio/x-midi
RealAudio 音乐文件	.ra, .ram audio/x-pn-realaudio
MPEG 文件	.mpg,.mpeg video/mpeg
AVI 文件	.avi video/x-msvideo
GZIP 文件	.gz application/x-gzip
TAR 文件	.tar application/x-tar