

# 万维网使用统一资源定位符URL来指明资源位置

URL的形式：<协议>://<主机>[:<端口>]/<路径>

## HTTP (HyperText Transfet Protocol)

HTTP定义了浏览器怎么向万维网请求万维网文档，以及万维网服务器怎样把万维网文档传送给浏览器。

- HTTP/1.0采用**非持续连接**方式。在该方式下，每次浏览器要请求一个文件都要与服务器建立TCP连接，当收到响应后就立即关闭连接。
  - 每请求一个文档就要有两倍的RTT开销。若一个网页上有很多引用对象，那请求每个对象都会花费2RTT时间。
  - 为了减少时延，浏览器通常会建立多个并行的TCP连接同时请求对个对象。这会大量占用服务器资源。
- HTTP/1.1采用**持续连接**方式。万维网在发送相应后保持该连接，使同一个浏览器和该服务器可以继续在这条连接上传送后续的HTTP请求报文。这并不局限于初始传送同一个页面上引用的对象，而是只要这些文档都在同一个服务器上就行。
  - 为提高效率，HTTP/1.1持续连接还可以使用流水线方式工作，即在浏览器收到HTTP响应报文之前就连续发送多个请求报文。

## HTTP报文格式

HTTP是**面向文本**的，其报文每个**字段**都是一些**ASCII码串**，并且每个字段的**长度都是不确定的**。



Connection为close的话表示非持续连接，即请求完文档后就释放连接。

## 使用Cookie查看服务器上记录用户信息

- Cookie是一种对无状态的HTTP进行状态化的技术。
- Cookie的工作原理：

用户主机向服务器建立一个TCP连接，当主机发送HTTP请求时，服务器进程就会为其产生一个唯一的Cookie识别码，并以此为索引在服务器的后端数据库中创建一个项目，用来记录该用户访问该网站的各种信息；

接着给浏览器发送响应报文，报文中包含一个首部为Set-Cookie的首部行，取值就是Cookie识别码。

浏览器收到相应报文后，就在一个特定的Cookie文件中添加一行，记录该服务器的域名和Cookie的识别码。

当用户再次使用这个浏览器访问这个网站时，每发送一个HTTP请求报文，浏览器都会从Cookie文

件中取出该网站的Cookie识别码，并放到HTTP请求报文的Cookie首部行中，服务器根据Cookie识别码就可以识别出该用户，并返回该用户的个性化网页。

## 万维网缓存和代理服务器

- 可以使用缓存机制提高万维网的效率
- 万维网缓存又称**Web缓存**，可位于客户机，也可位于中间系统上，位于中间系统上的Web缓存又称为**代理服务器**。
- Web把最近的一些请求和响应暂存在本地磁盘中。当新请求到达时，若发现这个请求与暂时存放的请求相同，就返回暂存的响应，而不需要按URL的地址再次去因特网访问该资源。

### 学习笔记内容来源：

<https://www.bilibili.com/video/BV1c4411d7jb?p=73>