

web

对象寻址

url: 统一资源定位器 (uniform resource locator)

协议: //主机地址: 端口/路径。协议忽略时为http

http超文本传输协议, 使用的传输层协议是TCP协议

http是无状态协议: 服务器不维护任何有关客户端过去所发请求的信息

HTTP连接

- 非持久性连接: TCP最多允许传输一个对象, 一次tcp只有一个对象
 - 响应时间: RTT: 客户端发送一个小数据包到服务端, 并返回所经历的时间。
 - 整个响应时间为 $2 * rtt + \text{文件发送时间}$ 。在ppt中
 - 缺点: 时间长, 响应慢
- 持久性连接: tcp传输多个对象, 后续的http消息可通过这个连接发送数据, 保持tcp连接
 - 无流水: 客户端只收到前一个响应之后, 才发送新请求。每个对象耗时一个RTT
 - 带有流水机制: 客户端只要遇到一个引用对象就尽快发送请求。理想情况下, 所以对象耗时一个RTT

HTTP消息

- 请求消息: ASCII:人可直接读
 1. post方法: 把客户端数据放到消息体里
 2. get方法: 通过request行里的url字段上传http1.1中支持上传文件: put方法
- 响应消息: 人可直接读

cookie

为了辨别用户身份、进行session跟踪而存储在用户本地的数据

原理: 服务器在为第一次访问的用户创建一个cookie的id, 然后返回的响应消息里有set-cookie的字段和值。之后客户发送的请求头里都会带上cookie的值。服务器会根据cookie的值发送消息。

web缓存/代理服务器技术

功能: 在不访问服务器的前提下满足用户的http请求

原因:

1. 缩短用户请求的响应时间
2. 减少机构/组织的流量
3. 在大范围实现有效的内容分发

- 步骤:

用户设定浏览器通过缓存进行web访问

浏览器向代理服务器（不是原始服务器）发送所以请求

- 如果所请求对象在缓存中，缓存返回对象
- 否则代理服务器向原始服务器发送http请求，获取对象，然后发送给客户，并且保存该对象。

缓存既充当客户端，也充当服务器

一般由ISP（Internet服务提供商）架设

面对网络的延迟时的解决方法

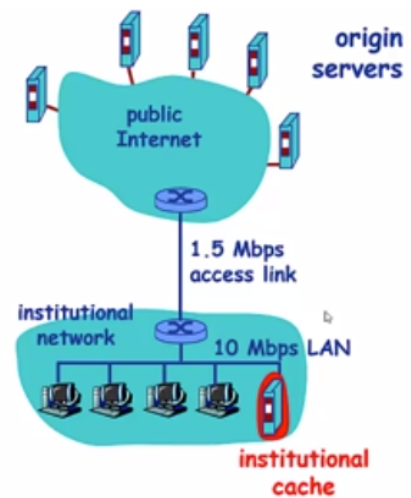
1. 提升互联网带宽：成本太高
2. web代理服务器：不怎么花钱

❖ 解决方案2:

- 安装Web缓存
- 假定缓存命中率是0.4

❖ 网络性能分析:

- 40%的请求立刻得到满足
- 60%的请求通过原始服务器满足
- 接入互联网的链路的利用率下降到60%，从而其延迟可以忽略不计，例如10微秒
- 总的平均延迟=互联网上的延迟+访问延迟+局域网延迟= $0.6 \times 2.01 \text{ 秒} + 0.4 \times n \text{ 微秒} < 1.4 \text{ 秒}$



web缓存怎么解决版本的更新

方法：条件性GET方法

目标：如果缓存有最新的版本，则不需要发送请求对象

- 对缓存来说：在http请求消息上声明所持有版本的日期
- 对服务器来说：如果缓存上的日期版本是最新的，则响应消息中不包含对象。

返回的消息是：HTTP/1.0 304 Not Modified