

# 计网HW2

---

## # P1

---

是非判断题。

- a. 假设用户请求由一些文本和 3 幅图像组成的 Web 页面。对于这个页面，客户将发送一个请求报文并接收 4 个响应报文。
- b. 两个不同的 Web 页面（例如，[www.mit.edu/research.html](http://www.mit.edu/research.html) 及 [www.mit.edu/students.html](http://www.mit.edu/students.html)）可以通过同一个持续连接发送。
- c. 在浏览器和初始服务器之间使用非持续连接的话，一个 TCP 报文段是可能携带两个不同的 HTTP 服务请求报文的。
- d. 在 HTTP 响应报文中的 Date：首部指出了该响应中对象最后一次修改的时间。
- e. HTTP 响应报文决不会具有空的报文体。

1. 错

2. 对

3. 错

4. 错

5. 错

## # P3

---

考虑一个要获取给定 URL 的 Web 文档的 HTTP 客户。该 HTTP 服务器的 IP 地址开始时并不知道。在这种情况下，除了 HTTP 外，还需要什么运输层和应用层协议？

- 应用层：DNS

- 运输层：UDP、TCP

## # P7

---

假定你在浏览器中点击一条超链接获得 Web 页面。相关联的 URL 的 IP 地址没有缓存在本地主机上，因此必须使用 DNS lookup 以获得该 IP 地址。如果主机从 DNS 得到 IP 地址之前已经访问了  $n$  个 DNS 服务器；相继产生的 RTT 依次为  $RTT_1$ 、 $\dots$ 、 $RTT_n$ 。进一步假定与链路相关的 Web 页面只包含一个对象，即由少量的 HTML 文本组成。令  $RTT_0$  表示本地主机和包含对象的服务器之间的 RTT 值。假定该对象传输时间为零，则从该客户点击该超链接到它接收到该对象需要多长时间？

$$T = 2RTT_0 + RTT_1 + RTT_2 + \dots + RTT_n$$

## # P8

---

参照习题 P7，假定在同一服务器上某 HTML 文件引用了 8 个非常小的对象。忽略发送时间，在下列情况下需要多长时间：

- a. 没有并行 TCP 连接的非持续 HTTP
- b. 配置有五个并行连接的非持续 HTTP
- c. 持续 HTTP

P7 中时间设为  $T$

1.  $T + 8 \times 2RTT_0 = 18RTT_0 + RTT_1 + RTT_2 + \dots + RTT_n$
2.  $T + 2 \times 2RTT_0 = 6RTT_0 + RTT_1 + RTT_2 + \dots + RTT_n$
3. 流水线：  $T + RTT_0 = 3RTT_0 + RTT_1 + RTT_2 + \dots + RTT_n$

没有流水线：

$$T + 8RTT_0 = 10RTT_0 + RTT_1 + RTT_2 + \dots + RTT_n$$

## # P9

考虑图 2-12，其中有一个机构的网络和因特网相连。假定对象的平均长度为 850 000 比特，从这个机构网的浏览器到初始服务器的平均请求率是每秒 16 个请求。还假定从接入链路的因特网一侧的路由器转发一个 HTTP 请求开始，到接收到其响应的平均时间是 3 秒（参见 2.2.5 节）。将总的平均响应时间建模为平均接入时延（即从因特网路由器到机构路由器的时延）和平均因特网时延之和。对于平均接入时延，使用  $\Delta/(1 - \Delta\beta)$ ，式中  $\Delta$  是跨越接入链路发送一个对象的平均时间， $\beta$  是对象对该接入链路的平均到达率。

- 求出总的平均响应时间。
- 现在假定在这个机构 LAN 中安装了一个缓存器。假定命中率为 0.4，求出总的响应时间。

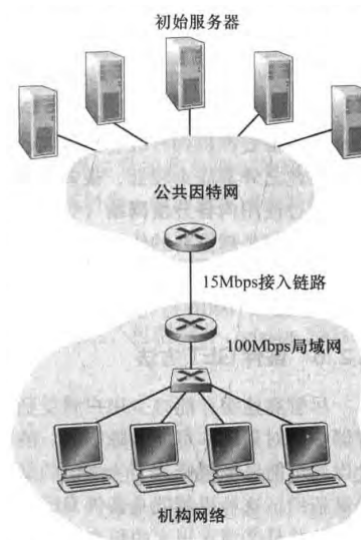


图 2-12 一个机构网络与因特网之间的瓶颈

1.  $\Delta = \frac{850000 \text{ bits}}{15 \text{ Mbps}}$

平均接入时延  $\frac{\Delta}{(1 - \Delta\beta)} \approx 0.6 \text{ s}$

总的平均响应时间  $0.6 + 3 = 3.6 \text{ s}$

2. 未命中平均接入时延  $\frac{\Delta}{(1 - p\Delta\beta)} \approx 0.089 \text{ s}$

未命中总的平均响应时间 $0.089 + 3 = 3.089s$

命中则认为时延几乎为0

平均响应时间为 $0.4 \times 3.089s \approx 1.24s$