

# CH8-应用层

## 1.掌握应用层进程通信方式以及服务进程工作模式

- 应用进程: 为解决具体应用问题而彼此通信的进程
- 每个应用层协议是为了解决某一类应用问题
- 客户/服务器 (C/S, Client/Server) 方式:
  - 客户是服务请求方 (主动请求服务, 被服务)
  - 服务器是服务提供方 (被动接受服务请求, 提供服务)
  - C/S方式可以是面向连接的, 也可以是无连接的:
    - 面向连接时, C/S通信关系一旦建立, 通信就是**双向的**, **双方地位平等**, 都可发送和接收数据
- 浏览器/服务器(B/S, Browser/Server) 方式 : C/S的特例, 客户软件改为浏览器
  - 采取**浏览器请求、服务器响应**的工作模式
  - **用户界面完全通过Web浏览器实现**, 一部分事务逻辑在前端实现, 但主要的事务逻辑在**服务器端实现**
  - 三层架构: 数据层-处理层-表现层



- 优势和不足:
  - 界面统一, 使用简单: 只需要安装浏览器软件
  - 易于维护
  - 可扩展性好: 采用标准的TCP/IP和HTTP协议
  - 信息共享度高: HTML是数据格式的一个开放标准, 目前大多数流行的软件均支持HTML
  - 可能面临浏览器兼容性问题
- P2P对等方式
  - 两个进程在通信时并不区分服务的请求方和服务的提供方, 平等、对等的通信。每一个P2P进程既是客户同时也是服务器
- 服务器进程工作方式
  - 循环方式(iterative mode)
    - 一次只运行一个服务进程, 服务进程按**请求的先后顺序**依次做出响应 (**阻塞方式**)
  - 并发方式(concurrent mode)
    - 同时运行多个服务进程,
    - 每一个服务进程都对某个特定的客户进程做出响应 (**非阻塞方式**)
- 选取合适的传输服务

### TCP服务：

- 面向连接：用户端和服务端需要建立连接
- 接收和发送进程间的可靠传输
- 流量控制：发送方不会淹没接收方
- 拥塞控制：网络负载过高时限制发送方发送
- 不提供：延迟保证，最小带宽保证

### UDP服务：

- 接收和发送进程间的不可靠传输
- 不提供：连接建立，可靠性、流量控制、拥塞控制和带宽保证

需要新的传输服务？  
怎么办？

协议标准 v.s. 私有协议

- TCP不提供延迟保证和最小带宽保证
- UDP不提供可靠性、流量控制、拥塞控制和带宽保证

## 2.掌握域名系统DNS基本原理和工作机制

- 域名映射成IP地址，采用了**层次树状结构**的命名方法
- 域名基本要求：各分量之间用小数点(.)隔开，总长不超过255个字符。**点“.”的个数至少为一个**
- 域名注册：域名管理机构分级负责域名注册。三级域名注册由其所属二级域名机构负责，以此类推。我国的二级域名注册机构为**中国互联网络信息中心(CNNIC)**
- 域名解析
  - 域名服务器：保存关于**域树(domain tree)**的结构和设置信息的服务器程序，称为**名字服务器(name server)**或**域名服务器**
  - 域名和IP地址**不一定是一一映射**。域名**解析过程对用户透明**
  - **权威名字服务器**：根据本地知识知道一个DNS区的内容，无需查询其他服务器
    - 层次树状结构：根名字、顶级域名、二级域名、三级域名（<=3:**本地域名服务器**）
  - **递归解析器**：递归方式运行，用户程序联系域名字服务器
- 域名查询
  - 递归查询(recursive query)：主机进行查询，本地域名服务器向下一步应查询的域名服务器发出查询请求替递归服务器继续查询
  - 迭代查询(或循环查询，iterative query) **优先使用**：域名字服务器向根服务器优先迭代，把下一步应查询的域名服务器IP地址告诉本地域名服务器，让本地域名服务器继续
  - **DNS下使用UDP数据包**，端口号53
  - DNS**高速缓存**作用:域名服务器广泛使用高速缓存，**生存时间TTL**
  - DNS报文的**RR (Resource Record)** 格式

## DNS资源记录的类型 RR格式: (name, ttl, class, type, value)

### ➤ type=A

- name是主机名
- value是主机IPv4地址

### ➤ type=CNAME

- name是某些“规范”名称的别名, 例如 www.cs.vu.nl.是主机名papac022.vu.nl.的别名
- value是IP地址

### ➤ type=AAAA

- name是主机名
- value是主机IPv6地址

### ➤ type=MX

- name是请求域名
- value是与请求域名关联的SMTP邮件服务器的名称

|                    |       |    |       |                          |
|--------------------|-------|----|-------|--------------------------|
| • google.com.      | 299   | IN | AAAA  | 2a00:1450:4017:804::200e |
| • dip0.connect.de. | 86400 | IN | A     | 46.82.174.69             |
| • www.cs.vu.nl.    | 60    | IN | CNAME | papac022.vu.nl.          |

域名系统

引自·徐敬东

37

- 根服务器: 共有13套 (不是13台机器) 域名为a.rootservers.net - m.rootservers.net

- 目前全球已设置了1000多台镜像根服务器, **就近**找到一个根服务器
- 每套都可以有多个镜像(mirrored)根服务器, 其内容定期与上述对应的根服务器同步 (**同步需要一定的时间才能完成**)

- DNS遇到的问题: 安全问题, UDP明文传输, 未加上任何的认证和加密措施。解决方法: DNSSEC 数字签名

•

### 3.掌握电子邮件系统体系结构及基本工作原理

- 体系结构

- 用户代理 (user agent) ——邮件客户端
- 传输代理 (message transfer agent) ——邮件服务器
- 电子邮件系统常采用C/S工作模式
- 简单邮件传输协议SMTP (Simple Mail Transfer Protocol)
  - 利用**TCP**可靠地从客户向服务器传递邮件, 使用**端口25**
  - 3个阶段: 连接建立、邮件传送、连接关闭
  - SMTP是一个简单的**ASCII**协议
  - 不断重复: **命令-响应**
  - 服务发送的行用“S:” 标识; 客户端发送的行用“C:” 标识
  - SMTP 是一个**推协议**, 需要接收方始终在线
- 最终交付 (邮件访问) 协议
  - **POP3** (Post Office Protocol-Version 3), 第三版邮局协议 110端口
    - **认证**(Authorization): 处理用户登录的过程
    - **事务处理**(Trnsactions): 用户收取电子邮件, 并将邮件标记为删除
    - **更新**(Update): 将标为删除的电子邮件删除
    - POP3使用C/S工作方式
  - **IMAP**: Internet Message Access Protocol, Internet邮件访问协议 **143端口**
    - 把每个邮件与一个文件夹联系起来
    - 为用户提供了在远程文件夹中**查询邮件**的命令,
    - 维护了IMAP会话的**用户状态信息**

- 具有**允许用户代理获取邮件某些部分**的命令：可以避免可能包含如音频或视频片段的大邮件
- Webmail (HTTP)：基于Web的电子邮件
  - 使用Web作为界面，用户代理就是普通的浏览器
  - 用户及其远程邮箱之间的通信通过HTTP进行

#### 4.掌握万维网WWW系统结构和HTTP协议工作原理

- 统一资源定位器URL：协议类型+主机名即服务器+端口+路径和文件名

##### ➤ 统一资源定位器URLs (Uniform Resource Locators)

- 例如：<http://www.tsinghua.edu.cn:80/yxsj.htm>

院系设置

协议类型      主机名即服务器      端口      路径和文件名

| 名字     | 用途      | 实例                                   |
|--------|---------|--------------------------------------|
| http   | 超文本HTML | http://www.xaut.edu.cn/xxgk/xxjj.htm |
| https  | 安全超文本   | https://www.overleaf.com/            |
| ftp    | FTP     | ftp://ftp.xaut.edu.cn                |
| file   | 本地文件    | file:///usr/xaut/prog.c              |
| mailto | 发送邮件    | mailto:xautmail@xaut.edu.cn          |
| rtsp   | 流媒体     | rtsp://youtube.com/montypython.mpg   |
| sip    | 多媒体呼叫   | sip:eve@advesary.com                 |

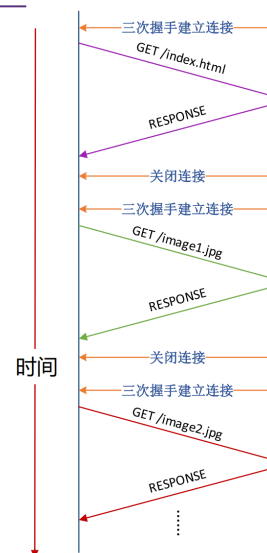
55

- 超文本传输协议HTTP：在传输层通常使用**TCP协议**，缺省使用TCP的**80端口**
  - **无状态协议**，服务器端不保留之前请求的状态信息
- HTTP 1.0

##### ➤ 假设用户输入URL

- <http://cuiyong.net/index.html>
- HTTP基本交互过程
  - TCP三次握手连接建立
  - Request (Get /index.html)
  - Response
  - 关闭TCP连接
- 如某页面包含2幅jpg图像
  - 需要**三次完整的HTTP交互过程**
  - 即TCP连接的建立，数据传输，TCP连接终止
  - 每次**包含TCP建连断连过程**

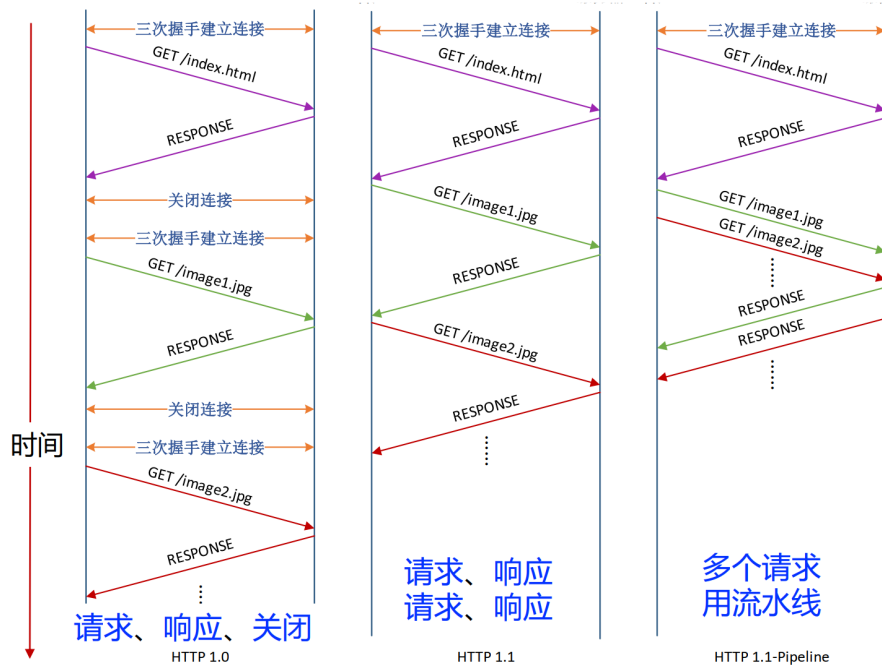
1. 建连
2. 请求
3. 响应
4. 关闭



766

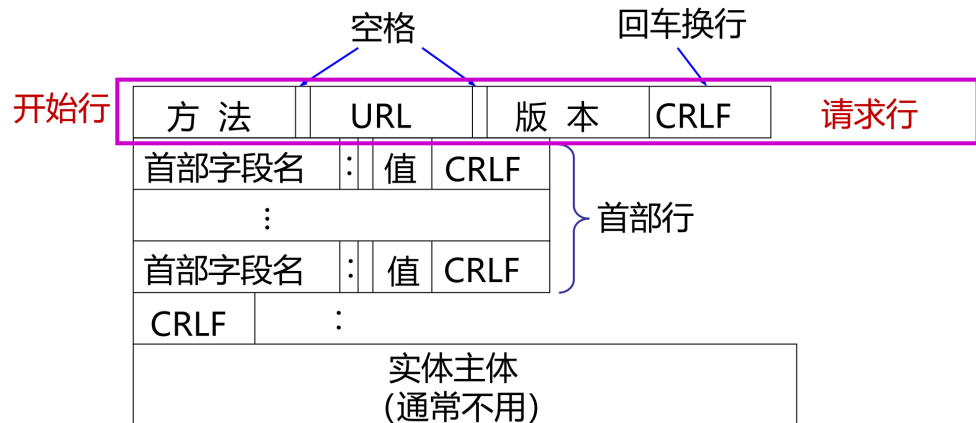
210. 1.1 管理网络 公海万维

- HTTP/1.0缺省为**非持久连接** 每次连接需要经历TCP慢启动阶段
- HTTP/1.1缺省为**持久连接**，支持流水线机制，经历较少的慢启动过程，减少往返时间，降低响应时间

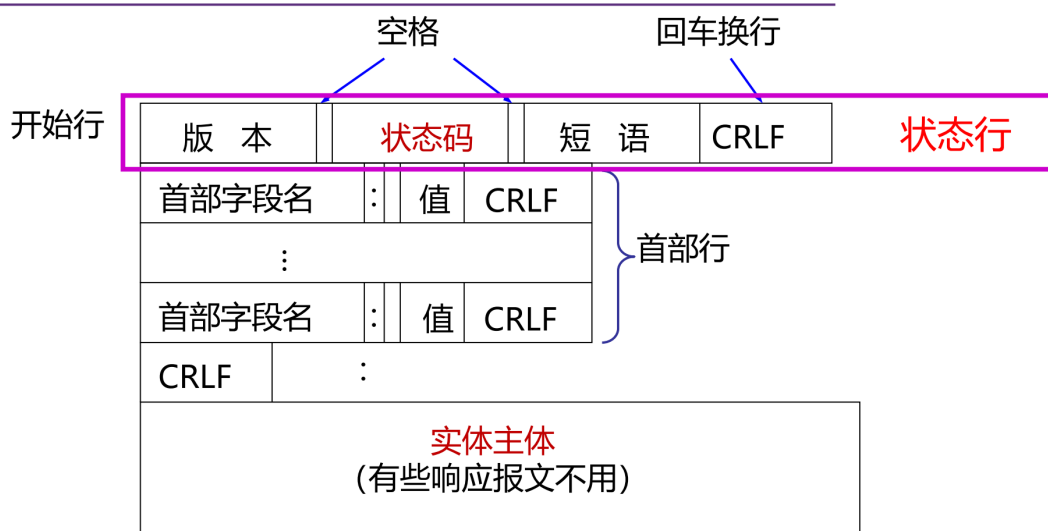


- HTTPS
  - 增加SSL/TLS (TLS 1.2) 层, 在TCP之上提供安全机制
- HTTP/2.0
  - 目标: 提高带宽利用率、降低延迟
  - 增加二进制格式、TCP多路复用、头压缩、服务端推送等功能
- 报文结构: 请求报文
  - 开始行、首部行和实体主体

在请求报文中, 开始行就是请求 `GET /~ross/index.html HTTP/1.0`

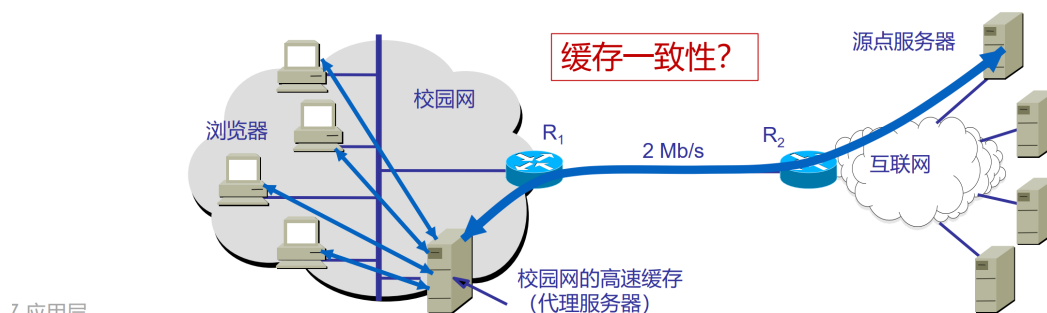


- 报文结构: 相应报文
  - 状态行+首部行+实体主体



示例: HTTP/1.1 200 OK <data>

- 静态/动态
  - 静态对象与静态网页: 文本, 表格, 图片, 图像和视频等多媒体类型的信息 (实现语言: 标记语言, 如: 超文本标记语言HTML, XML, PHP等) 字体、颜色和布局等风格类型的信息 (实现语言: 层叠样式表CSS)
  - 动态对象与动态网页: 交互信息, 比如, 用户注册信息、登录信息等 (实现: PHP/JSP等语言+MySQL等数据库)
- 代理服务器: 万维网高速缓存(Web cache), 它代表浏览器发出 HTTP 请求
  - (1) 浏览器访问服务器时, 先与Proxy/高速缓存建立连接, 并发出 HTTP 请求报文
  - (2) 若高速缓存已经存放了所请求的对象, 则将此对象放入 HTTP 响应报文中返回给浏览器
  - (3) 否则, 高速缓存就代表用户浏览器, 与源点服务器建立 TCP 连接, 并发送 HTTP 请求报文
  - (4) 源点服务器将所请求的对象放在 HTTP 响应报文中返回给校园网的高速缓存
  - (5) 高速缓存收到此对象后, 先复制在其本地存储器中, 再放在 HTTP 响应报文中返回给用户



7 应用层

- Web缓存技术与Web代理
  - 询问式策略
    - 客户端在发送的HTTP请求中指定缓存的时间, 请求头包含 If-modified-since: <date>
    - 服务器: 如果缓存的对象是最新的, 在响应时无需包含该对象, 响应头包含 HTTP/1.1 304 Not Modified, 否则服务器响应HTTP/1.1 200 OK <data>
  - 原始服务器明确指令限制缓存某些Web页
    - no-cache
      - 授权访问的web页也禁止缓存
- 安全与隐私: cookie技术

## 流媒体

类型：点播、直播、实时交互

特性：时序性约束，具有一定容错性

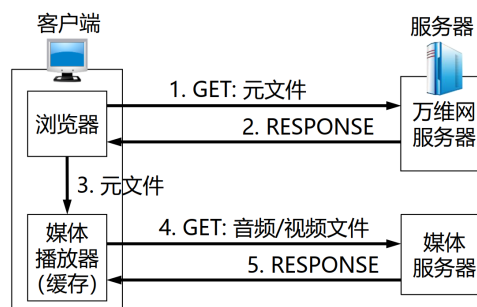
期望：高清、低延迟、不卡顿

MPEG：帧内编码帧 I 帧、预测帧 P 帧、双向帧 B 帧，图像组 GOP：一系列连续 IPB 组成的画面

浏览器从服务器下载流媒体文件的过程：

### ➤ 浏览器从服务器下载流媒体文件

- 浏览器用户使用 HTTP 的 GET 报文接入到万维网服务器；这个超链指向一个**元文件（有音/视频文件的统一资源定位符 URL）**
- 万维网服务器把该元文件装入 HTTP 响应报文的主体，发回给浏览器
- 浏览器调用媒体播放器，把提取出的元文件传输给媒体播放器
- 媒体播放器使用元文件中的 URL，向媒体服务器发送 HTTP 请求报文，要求下载音/视频文件（**如对应的某个 GOP**）
- 媒体服务器发送 HTTP 响应报文，把音/视频文件发送给媒体播放器；媒体播放器边下载边解压缩边播放（**通过时间戳同步音频流和视频流**）



流式存储媒体的典型下载过程