






应用层

 Courses	 计算机网络
<input checked="" type="checkbox"/> Done	<input checked="" type="checkbox"/>
 Status	Done

网络应用模型

客户/服务器 模型

Client

- 根据需要启动
- Speaks first
- 可能具有动态IP地址

Server

- 总是打开状态
- 向客户端提供请求的服务
- 服务器主机具有永久IP地址

P2P模型

- 没有永远在线的服务器
- 任意端系统间直接通信
- peer request service from and provide service to other peers
- peer之间间歇性连接并更改IP地址

DNS系统

Function

- 将域名转换为IP地址
- e.g. www.baidu.com → 119.75.217.109

层次域名空间

- 顶级域名位于顶部
- 域是子树 □ 例如, .edu、umich.edu、eeecs.umich.edu
- 名称是叶到根路径
- 树的深度是任意的（最大128）
- 轻松避免的名称冲突

域名服务器

- 根域名服务器
 - 如果名称映射未知，则联系根域名服务器
- 顶级域名服务器：负责com、org、net、edu等，以及所有顶级国家域名，如cn、uk、fr
 - TLD：顶级域
- 权威域名服务器
- 本地域名服务器
 - 严格上不属于层次结构；具有最近的查询本地缓存

域名解析过程

迭代查询：已联系的服务器回复，其中包含要联系的**下一个服务器**的名称

DNS资源记录：RR (name, value, type, ttl)

FTP—文件传输协议

FTP协议的工作原理

- 处理异构操作系统和文件系统
- 需要对远程文件系统进行访问控制

控制连接与数据连接

- 在端口21处建立服务器，打开控制连接
- 客户端通过控制连接授权、发送命令来浏览远程目录
- 当收到文件传输命令时，打开第二个与客户端的TCP连接，用于传输文件—每个文件一个链接—一个文件传输后关闭连接
- 控制连接保持“Out of band”：控制信息和数据传输在不同的带宽上进行，避免了相互干扰
- FTP服务器保持“user state”：当前目录，早期身份验证

电子邮件

电子邮件系统的组成结构

- 用户代理：撰写，编辑，阅读邮件消息；例如Outlook、Foxmail
- 邮件服务器：
 - 邮箱Mailbox：包含用户传入的邮件
 - 消息队列：外发邮件
 - SMTP协议：邮件服务器之间发送邮件消息

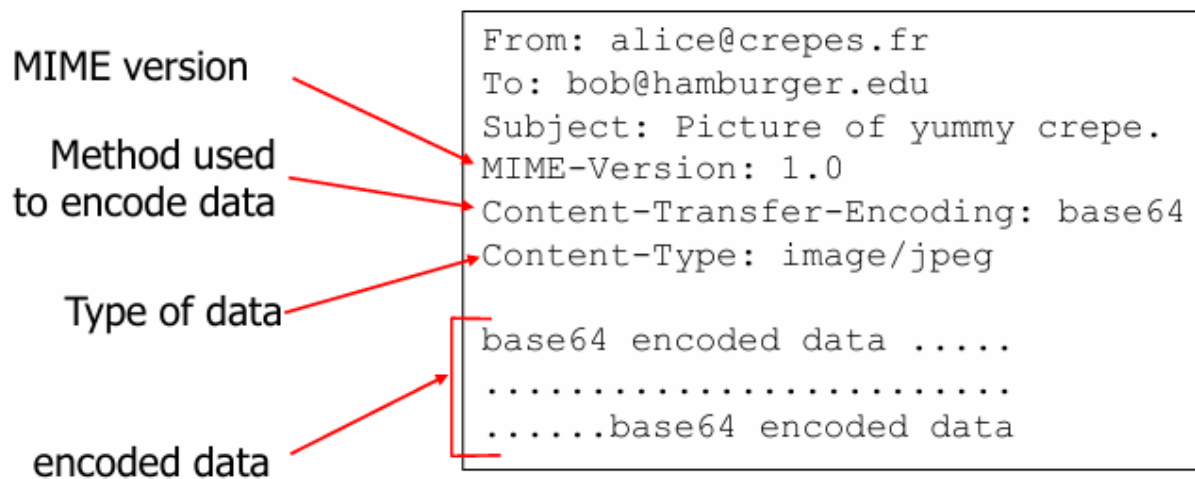
电子邮件格式与MIME

电子邮件格式

Header 和 Body | 空格分开

MIME—多用途Internet邮件扩展

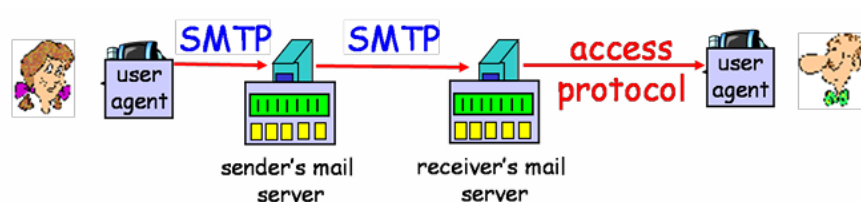
- 允许在单个邮件中包含单独的组件（程序、图片、音频视频等）
- 5个新的邮件表头字段：version、内容类型、编码格式等



SMTP协议与POP3协议

SMTP

- 使用TCP，端口号25
- 直接传输：将电子邮件从客户端传输到服务器
- 格式（MIME规定）：7位ASCII码
- 三个传输阶段
 - 握手（问候）—220 OK—hello or echo—250 OK
 - 传输一个或多个邮件数据
 - 连接终止



POP3

邮件访问协议：从服务器检索邮件

协议工作过程：

- 授权阶段
 - 用户：连接到服务器—端口110
 - 用户：用户名+密码

- 交易阶段
 - 用户：list、retr、dele、quit
- 更新阶段

WWW

WWW的概念与组成结构

组件：

- 基础设施
 - 客户
 - 服务器（DNS、CDN、数据中心）
- 内容
 - URL：命名内容
 - HTTP：格式化内容
- 信息交换协议：HTTP

URL—WWW上对象的唯一标识符

- 格式： `<protocol>://<host>:<port>/<path>?query_string`
 - **protocol**：对象的传输或解释方法
 - **host**：DNS名或IP地址

HTTP协议

- 客户-服务器模型：服务器永远在线、众所周知；客户发起与服务器的联系
- 通过TCP在端口80上运行
- 无状态
 - 每个请求-响应都独立处理，服务器不保留状态
 - 好处：提高了服务器端的可伸缩性（**scalability**）
 - 故障处理更轻松
 - 处理更高的请求率

- 请求顺序无关紧要
- 坏处：某些程序需要持久状态
 - 需要唯一标识用户或存储临时信息（购物车、用户配置文件、使用情况跟踪等）
- 在HTTP/2之前，格式为ASCII码
- Client2Server：请求信息
 - 请求行：方法/资源/协议版本
- Server2Client：响应信息
 - 状态行：http版本/状态码/状态短语
 - 响应头：提供信息
 - 正文：可选数据

Cookies

客户端保持状态

- 客户端代表服务器存储小状态
- 客户端在将来的请求中向服务器发送状态

可以提供身份验证

版本迭代

- HTTP/1.0：非持久性连接 $2RTT + \Delta$ for **each object** in the HTML file
- 并发请求和响应：对network不友好、
- HTTP/1.1：持久连接——一个TCP连接跨多个请求
- 流水线：批量请求和响应以减少数据包数量；一个TCP段中可以包含多个请求

Web缓存

HOW?

- **If-modified-since**：返回“not modified” 如果资源自指定时间以来未修改
- 客户端在请求中制定ims时间
- 服务器将此与上次修改时间进行比较，如果未更改，则返回未更改，否则返回最新版本的“OK”

Where?

反向代理（Reverse proxies）：缓存靠近服务器的文档

- 减少服务器负载
- 由内容提供商提供

正向代理（Forward proxies）：缓存靠近客户端的文档

- 减少网络流量和延迟
- 由ISP或企业提供

CDN—内容分发网络

*CDN的主要目标是通过将内容放置在离用户**更近**的地方，以减少网络延迟，提高访问速度，并增强用户体验。*

- 在整个互联网上数百台服务器上复制内容
- CDN 分发节点协调内容分发
- 将内容放置在靠近用户的位置

支持技术

- DNS
- 路由
- URL重写
- 重定向策略