

- 第6章 应用层
- ▼ 网络应用模型
 - C/S模型(请求-服务)
 - P2P..(分散式)
-
- ▼ 域名系统 (DNS)
 - 层次域名空间(1+4:←根顶二三主)
 - 域名服务器(4)
 - ..解析过程(2询: 1Lat 1Next)
- ▼ 文件传输协议 (FTP)
 - ..工作原理(1模型,1服务)
 - 控制连接 数据 ..(2: 21、20)
- ▼ 电子邮件
 - ..系统的组成结构(3:ASP)
 -
 - ..格式 MIME(标准)
 - SMTP POP3 IMAP(1发2收(25 110)
-
- ▼ 万维网 (WWW)
 - 概念与组成结构(UHH)
 -  超文本传输协议 (HTTP)(3特:无T80 1count:2type: $t_{\text{加载}}$:
n+1连接, 1..,1全发)
- 本章小结及疑难点

好的，同学们，我们来到了最后一章《应用层》。这一章离我们生活最近，也最具体。考纲内的协议，每一个都可能成为选择题或综合题的一部分。本章的复习策略是：**抓核心原理，联系实际应用，理清协议间的协同工作关系。**

跟上我的节奏，我们用最直观的方式，把这些看得见、摸得着的协议彻底搞定！

第6章 应用层

网络应用模型

C/S模型(请求-服务)

- **核心考点:** C/S模型的工作流程与主要特点。
 - **形象记忆:** 去饭店吃饭
 - **服务器 (Server):** 饭店。
 - 一直开门 (持续运行), 提供服务。
 - 被动 等待顾客上门。
 - 能力强大, 可以同时服务多桌客人。
 - **客户端 (Client):** 顾客。
 - 主动 去饭店请求服务 (点菜)。
 - 必须知道饭店的地址。
 - 顾客之间不直接交流。
 - **特点:**
 - 地位不平等: 饭店是服务方, 顾客是请求方。
 - 管理集中: 饭店统一管理。
 - 瓶颈: 客人太多, 饭店 (服务器) 会忙不过来。
 - **典型应用:** Web浏览、FTP、电子邮件。
-

P2P..(分散式)

- **核心考点:** P2P模型的工作原理与优缺点。
- **形象记忆:** 家庭聚餐 (Potluck)
 - **对等方 (Peer):** 每个来参加聚会的人。
 - **工作原理:**
 - 每个人既带自己的**拿手菜** (作为服务器, 提供资源)。
 - 又吃别人带来的菜 (作为客户端, 请求资源)。
 - 大家可以直接交流、分享。
 - **特点:**
 - 地位平等: 没有固定的C/S角色。
 - 分散式: 没有中心服务器, 一个人生病了不影响聚会。
 - 可扩展性好: 来的人越多, 菜色越丰富 (资源越多, 下载越快)。

- **缺点:**
 - 每个人都要出资源（占用自己电脑性能）。
 - 管理混乱，难以追溯。
 - **典型应用:** BT下载、迅雷。
-

域名系统 (DNS)

层次域名空间(1+4:←根顶二三主)

- **核心考点:** 域名的层次结构和命名规则。
- **形象记忆: 行政区划**
 - mail.bupt.edu.cn.
 - ←
 - . (根): 地球
 - cn (顶级域): 中国
 - edu (二级域): 教育系统
 - bupt (三级域): 北京邮电大学
 - mail (主机名): 邮件服务器
 - **规则:** 从小到大，主机.机构.类别.国家，不分大小写。

域名服务器(4)

- **核心考点:** 四种域名服务器的角色和层级关系。
 - **形象记忆: 通讯录查询系统**
 - **根域名服务器: 全球总通讯录。** 它不知道具体电话，但知道去哪个“省”的通讯录查。
-
- **顶级..: “省”级通讯录。** 比如 .com 服务器，它不知道 baidu.com 的电话，但知道要去 baidu 这个“市”的通讯录查。
 - **权限..: “市”...** 比如 baidu.com 的服务器，它记录了自己管辖范围内所有主机的“电话号码”。
 - **本地..: 你小区的门卫大爷。** 他不是官方机构，但他有自己的小本本（高速缓存），帮你跑腿问路，并把查过的结果记下来，方便下次查询。

..解析过程(2询: 1Lat 1Next)

- 核心考点: 递归查询与迭代..的区别。【绝对重点】
- 形象记忆: 帮你问路
 - 场景: 你(主机)想知道 `mail.bupt.edu.cn` 的IP地址, 你去问门卫大爷(本地域名服务器)。
 - i. 你 -> 门卫大爷 (主机 -> 本地域名服务器): 这一步总是 递归查询。
 - 人话: “大爷, 帮我查一下这个地址, 然后把最终结果告诉我就行, 中间过程别烦我。”
 - ii. 门卫大爷 -> 其他服务器 (本地域名服务器 -> 各级服务器): 这一步通常是 迭代查询。
 - 人话: 大爷开始自己跑腿问路。
 - 大爷问 根服务器: “`mail.bupt.edu.cn` 咋走?” 根服务器说: “我不知道, 但你可以去问 `.cn` 服务器。”
 - 大爷问 `.cn` 服务器: “`mail.bupt.edu.cn` 咋走?”
`.cn` 服务器说: “我不知道, 但你可以去问 `.edu.cn` 服务器。”
 - ... 以此类推, 直到问到 `bupt.edu.cn` 的权限服务器, 才拿到最终的IP地址。
 - 迭代的精髓在于: 每次都返回下一步该找谁的线索, 让请求者自己去问。
 - 总结:
 - 递归: 我只问你, 你必须给我最终答案。
 - 迭代: 我问你, 你可以告诉我下一步该问谁。

文件传输协议 (FTP)

..工作原理(1模型,1服务)

- 核心考点: FTP是基于C/S模型, 使用TCP提供可靠服务。
- 形象记忆: 一个搬家公司。通过电话 (控制) 指挥卡车 (数据) 来搬运家具 (文件)。

控制连接 数据 ..(2: 21、20)

- 核心考点: FTP使用两个独立的TCP连接来工作。【绝对重点】

- **形象记忆:** 搬家公司的**电话和卡车**
 - **控制连接** (端口 21): **电话线**。
 - **作用:** 你通过电话和经理沟通, 下达指令 (登录、列出文件 `ls`、下载 `get`)。
 - **生命周期:** 从你开始联系搬家公司, 到所有事情办完, 这条电话线**一直占着**。
 - **数据..** (端口 20): **搬运的卡车**。
 - **作用:** 真正用来运输家具 (文件) 的。
 - **生命周期:** **每搬运一件家具, 就派一辆新卡车**。运完这件, 卡车就走了 (连接关闭)。
-

- **带外控制:** 指令和数据分离传输, 就叫“带外控制”。
-

电子邮件

..系统的组成结构(3:ASP

- **核心考点:** 三大组成部分: 用户代理、邮件服务器、协议。
 - **形象记忆:** 现实中的邮政系统
 - **用户代理 (UA):** 你家里的**书桌** (写信) 和**信箱** (收信), 比如Outlook, Foxmail。
 - **邮件服务器:** 整个城市的**邮局大楼**, 负责存储、转发信件。
 - **协议:** 邮政系统的**工作规则**。
-

..格式 MIME(标准

- **核心考点:** MIME的作用。
- **形象记忆:**
 - **邮件格式:** `信封` + `内容` (信头 + 正文)。
 - **MIME (多用途因特网邮件扩展):** 早期邮局只能寄信 (ASCII文本)。MIME相当于制定了一套**国际包裹打包标准**, 让你可以邮寄**照片、视频、各种文件** (非ASCII数据), 邮递员 (SMTP) 也能认识并投递。

SMTP POP3 IMAP(1发2收(25 110

- **核心考点:** 三大核心邮件协议的作用与分工。【高频考点】
 - **形象记忆:** 邮局的工作人员
 - **SMTP (简单邮件传输协议) - “邮递员”**
 - **作用:** 负责发信和送信。
 - 从你的**用户代理(UA)** -> **发件方邮局**，以及在**邮局之间**传递信件。
 - **特点:** **推 (Push)** 协议，使用**TCP端口25**。
-
- **POP3 (邮局协议) - “憨憨的取信人”**
 - **作用:** 负责收信。
 - 你去邮局，把信箱里的信件**全部取出来，搬回家**（下载到本地），邮局的信箱通常就空了。
 - **特点:** **拉 (Pull)** 协议，使用**TCP端口110**。
 - **IMAP (因特网报文存取协议) - “聪明的取信人”**
 - **作用:** 也是收信。
 - 但你可以在邮局（服务器）就直接**整理信件、分类、只看标题**。你在手机上读了信，电脑上也会显示已读，因为操作都在服务器上同步。
 - **特点:** **拉 (Pull)** 协议，更复杂强大。
-

万维网 (WWW)

概念与组成结构(UHH

- **核心考点:** WWW的三大核心： `URL` ， `HTTP` ， `HTML` 。
 - **形象记忆:** 图书馆系统
 - **URL (统一资源定位符):** 图书馆里每一本书的**唯一编号**（索书号）。 `http://www.example.com/index.html`
-
- **HTTP (超文本传输协议):** 你和管理员之间的**借书规则**。
 - **HTML (超文本标记语言):** 书本的**排版格式**。
-

超文本传输协议 (HTTP)(3特:无T80

1count:2type: $t_{\text{加载}}$: $n+1$ 连接, 1..,1全发

- 核心考点:
 - HTTP的特点: **无状态**、使用**TCP**、默认端口**80**。
 - **持续连接** vs **非..** 的性能分析。【计算分析重点】
 - Cookie的作用。
- 形象记忆:
 - **无状态 (Stateless)**: 图书馆管理员**记性不好**。你每次去借书, 他都不知道你是谁, 你必须重新出示借书证。
 - **Cookie**: 图书馆发的**借阅卡**。服务器发给你一张卡, 你下次来时带上, 管理员一看卡就知道是你了。这是为无状态的HTTP**维持状态**。

• 【计算/分析模板】: 网页加载时间

问题: 访问一个网页, 该网页包含1个HTML文件和N个小图片, 求总加载时间。(假设 RTT 为往返时延)

i. 非持续连接 (HTTP/1.0):

- **过程:** 每请求一个对象 (HTML或图片), 都要经历: 建立TCP($1 RTT$) + 请求和接收对象($1 RTT$)。
- **总对象数:** 1 (HTML) + N (图片)。
- **公式:** $T_{\text{总}} = (N + 1) \times 2RTT + T_{\text{传输总}}$
- **记忆:** “一物一连接, 两次握手跑不掉”。

ii. 持续连接 (HTTP/1.1):

- **过程:** 只建立**一次**TCP连接 ($1 RTT$), 然后在这个连接上传输所有对象。
- **非..:**
 - **过程:** 建连($1 RTT$) -> 请求HTML($1 RTT$) -> 请求图1($1 RTT$) -> 请求图2($1 RTT$) ...
 - **公式:** $T_{\text{总}} = 1 \times RTT(\text{建连}) + (N + 1) \times RTT(\text{请求}) + T_{\text{传输总}}$
 - **记忆:** “一次连接, 串行请求”。
- **流水线方式:**
 - **过程:** 建连($1 RTT$) -> 请求HTML($1 RTT$) -> 一口气发出所有图片请求, 服务器再**流水线式**地发回

所有图片（只需 **1 RTT**）。

- **公式:** $T_{\text{总}} = 1 \times RTT(\text{建连}) + 1 \times RTT(\text{请求HTML}) + 1 \times RTT(\text{请求所有图片}) + T_{\text{传输总}}$
- **记忆:** “一次连接，并行请求”。

本章小结及疑难点

- **客户端端口 vs 服务器端口:**
 - 服务器用的是**熟知端口号**（如HTTP用80，FTP用21），是固定的，像饭店的门牌号。
 - 客户端用的是**临时端口号**，是随机分配的，像顾客拿的排队号。
- **因特网 vs 万维网:**
 - **因特网 (Internet):** 是一个巨大的**物理网络基础设施**，像全球的公路网。
 - **万维网 (WWW):** 是运行在因特网上的一个**应用服务**，像公路上跑的“快递服务”，是因特网最主要的部分，但不是全部。

应用层是协议栈的终点，也是我们网络世界的起点。掌握了这些核心协议的工作原理，你不仅能应对考研，更能深刻理解我们每天都在使用的互联网是如何运作的。祝大家学习顺利，成功上岸！