



第七章 应用层

授课教师：张圣林

南开大学



本章目标



1. 掌握应用进程通信方式以及服务进程工作模式
2. 掌握域名系统基本原理和工作机制
3. 掌握电子邮件系统体系结构及基本工作原理
4. 了解WWW系统结构框架、静态和动态Web及其应用技术，掌握HTTP协议及其工作原理
5. 了解流媒体基本概念、数字音视频与编码、流式存储媒体、直播与实时音视频、流媒体动态自适应传输
6. 了解内容分发背景及内容分发网络、以及P2P网络的工作机制
7. 掌握Telnet、FTP、SNMP等应用层协议的工作原理



本章内容



➤ 7.1 应用层概述 (P7)

1. 应用进程通信方式
2. 服务器进程工作方式

➤ 7.2 域名系统 (P25)

➤ 7.3 电子邮件 (P74)

➤ 7.4 WWW (P106)

➤ 7.5 流式音频和视频 (P147)

➤ 7.6 内容分发 (P165)

➤ 7.7 其它应用层协议 (P182)



应用进程通信方式



- 每个应用层协议都是为了解决某一应用问题，通过位于不同主机中的多个应用进程之间的通信和协同工作来完成
 - 两台主机通信实际是其上对应的两个应用进程(process)在通信
 - 应用进程: 为解决具体应用问题而彼此通信的进程
- 应用层的具体内容就是规定应用进程在通信时所遵循的协议

客户/服务器（C/S, Client/Server）方式

浏览器/服务器（B/S, Browser/Server）方式

对等（P2P, Peer to Peer）方式



应用进程通信方式



(1) 客户/服务器 (C/S, Client/Server) 方式

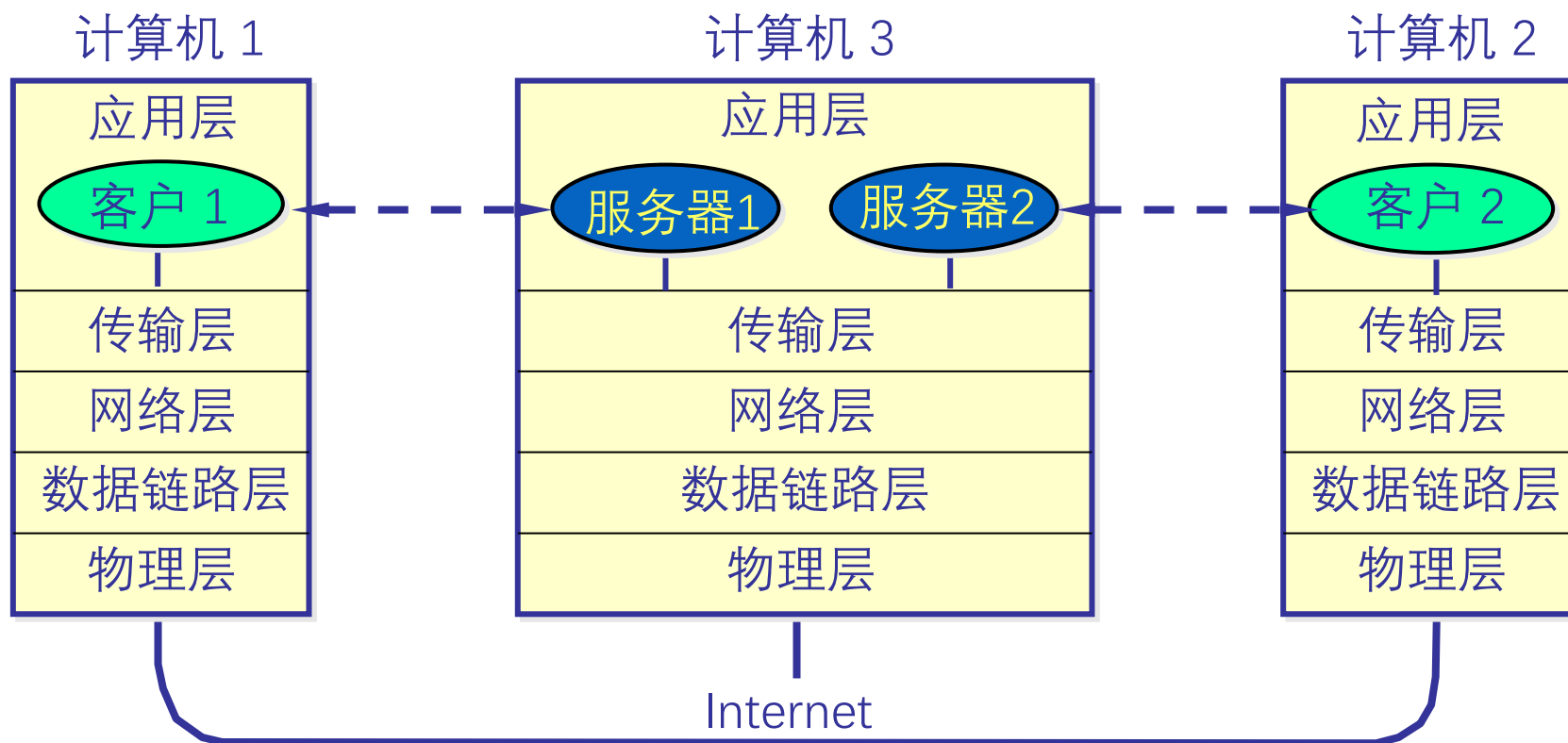
- 应用层的许多协议是基于C/S方式，例如，在移动互联网环境下，每个应用APP都是一个客户端
 - 客户(client)和服务端(server)是指通信中所涉及的2个应用进程
 - 客户/服务器方式描述的是应用进程之间服务和被服务的关系
 - 客户是服务请求方（主动请求服务，被服务）
 - 服务器是服务提供方（被动接受服务请求，提供服务）
- C/S方式可以是面向连接的，也可以是无连接的
- 面向连接时，C/S通信关系一旦建立，通信就是双向的，双方地位平等，都可发送和接收数据



应用进程通信方式



- 功能较强的计算机可同时运行多个服务器进程





应用进程通信方式



➤ 客户进程的特点

- 在进行通信时临时成为客户，它也可在本地进行其它的计算
- 在用户计算机上运行，在打算通信时**主动**向远地服务器**发起通信**
- **客户方必须知道服务器进程所在主机的IP地址才能发出服务请求**
- 需要时可以与多个服务器进行通信



应用进程通信方式



➤ 服务器进程的特点

- 专门用来提供某种服务的程序，可“同时”处理多个远地或本地客户的请求
- 必须始终处于运行状态才有可能提供服务
- 通信开始之前服务器进程不需要知道客户进程所在主机的IP地址，无论客户请求来自哪里，服务器进程被动等待服务请求的到来即可
- 通常是当系统启动时即自动调用并一直运行着。某些服务器程序也可以由用户或其它进程在通信前启动
- 被动等待并接受来自多个客户的通信请求



应用进程通信方式



(2)浏览器/服务器(B/S, Browser/Server) 方式

- B/S方式可以看做C/S方式的特例，即客户软件改为浏览器了
- B/S方式采取浏览器请求、服务器响应的工作模式
- 在B/S方式下，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





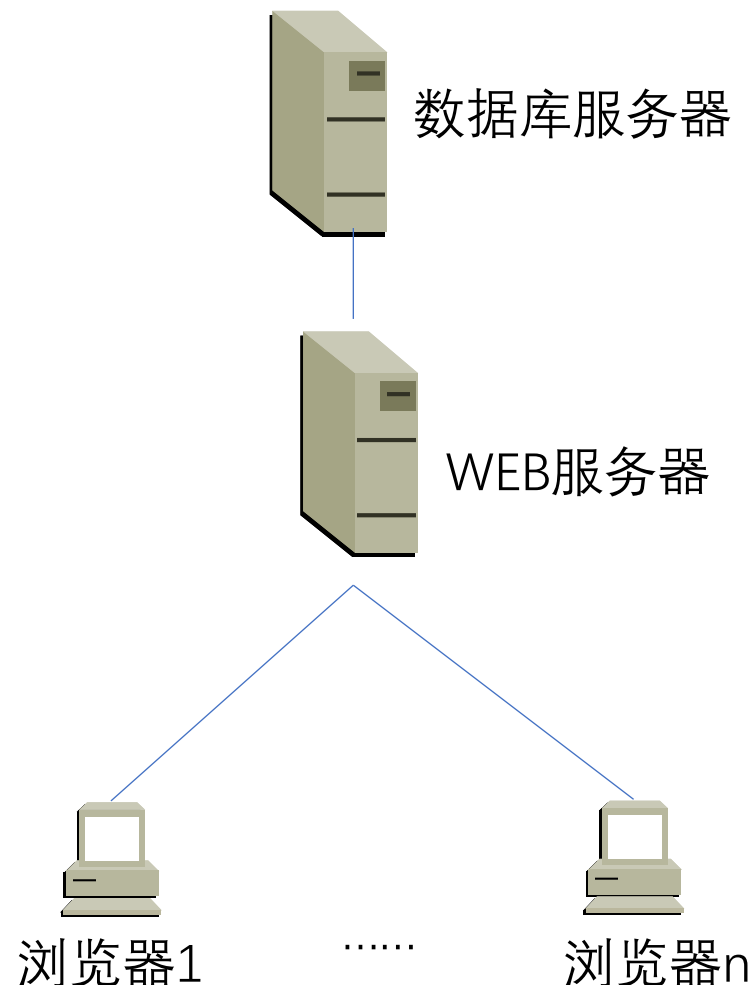
应用进程通信方式



➤ B/S方式通常采取3层架构实现

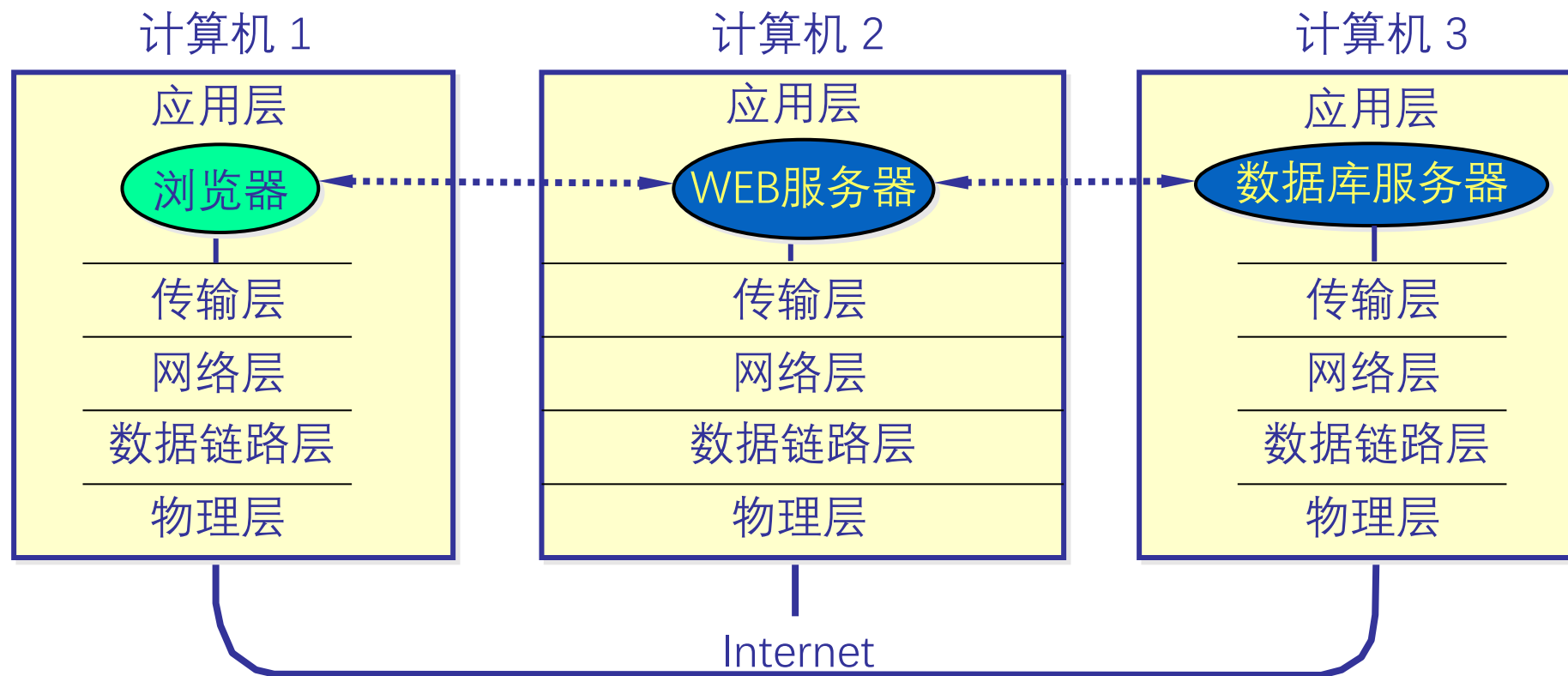
- 数据层：由**数据库服务器**承担**数据处理逻辑**，其任务是接受Web服务器对数据库服务器提出的数据操作请求，然后由数据库服务器进行数据处理并把处理结果返回给Web服务器
- 处理层：由**Web服务器**承担**业务处理逻辑**和**页面存储管理**，接受客户浏览器的任务请求，执行相应的事务处理
- 表现层：**浏览器**仅承担网页信息的浏览功能，以**超文本格式**实现信息的输入和浏览

➤ 实际部署时也可以把数据库服务器和Web服务器部署在同一台设备上





应用进程通信方式



B/S方式的3层架构示意



应用进程通信方式



➤ B/S方式的特点

- 界面统一，使用简单。客户端只需要安装浏览器软件
- 易于维护。对应用系统升级时，只需更新服务器端的软件，减轻了系统维护和升级的成本
- 可扩展性好。采用标准的TCP/IP和HTTP协议，具有良好的扩展性
- 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
- 需要注意的是，在一种浏览器环境下开发的界面在另一种浏览器环境下可能有不完全适配的情况，这时需要安装对应的浏览器



应用进程通信方式



(3) 对等 (P2P, Peer to Peer) 方式

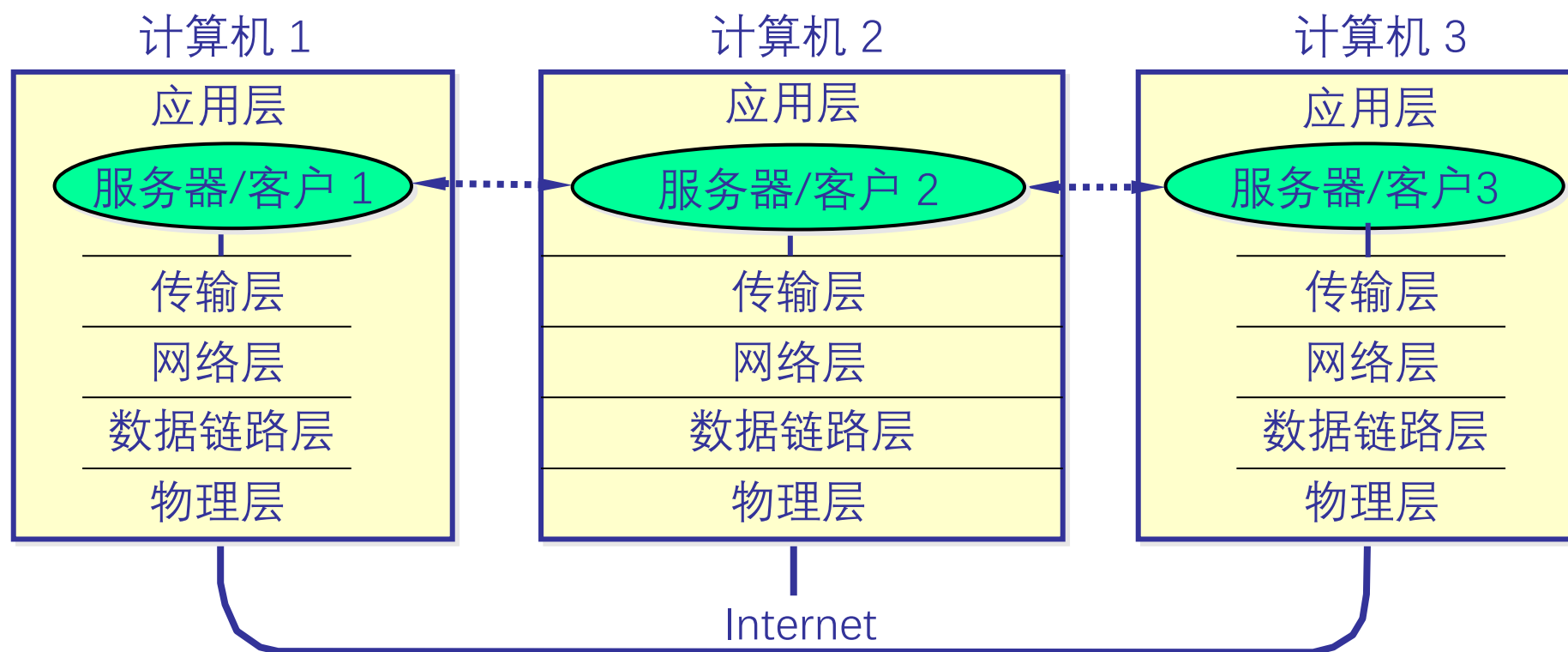
- **对等方式**是指两个**进程**在通信时并不区分服务的请求方和服务的提供方
 - 只要两个主机都运行P2P软件，它们就可以进行**平等、对等的通信**
 - 双方都可以下载对方存储在硬盘中的共享文档
- 音频/视频应用推动了P2P对等通信方式的发展
- 音频/视频流量已占主要比例



应用进程通信方式



- P2P方式从本质上看仍然是使用了C/S方式，但强调的是通信过程中的对等，这时每一个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
 - 当有多个客户进程请求服务时，服务进程就按请求的先后顺序依次做出响应 (阻塞方式)
- 并发方式(concurrent mode)
 - 可以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)



服务器进程工作方式



➤ 无连接循环方式服务

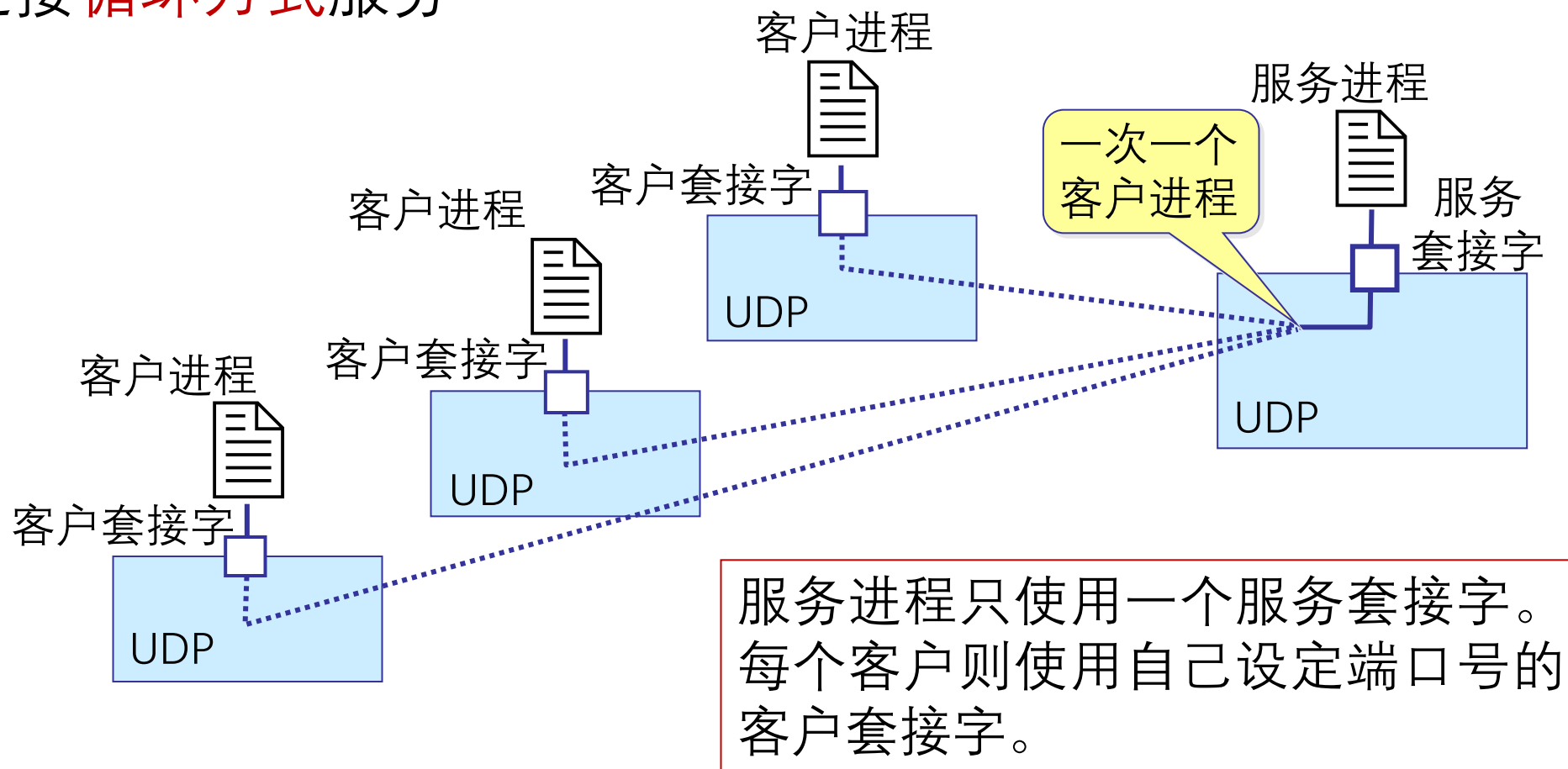
- 使用无连接的UDP服务进程通常都工作在循环方式，即一个服务进程在同一时间只能向一个客户进程提供服务。(顺序服务)
 - 服务进程收到客户进程的请求后，就发送UDP用户数据报响应该客户
 - 对其他客户进程发来的请求则暂时不予理睬，这些请求都在服务端的队列中排队等候服务进程的处理
 - 当服务进程处理完毕一个请求时，就从队列中读取来自下一个客户进程的请求，然后继续处理



服务器进程工作方式



无连接循环方式服务





服务器进程工作方式



➤ 面向连接的并发方式服务

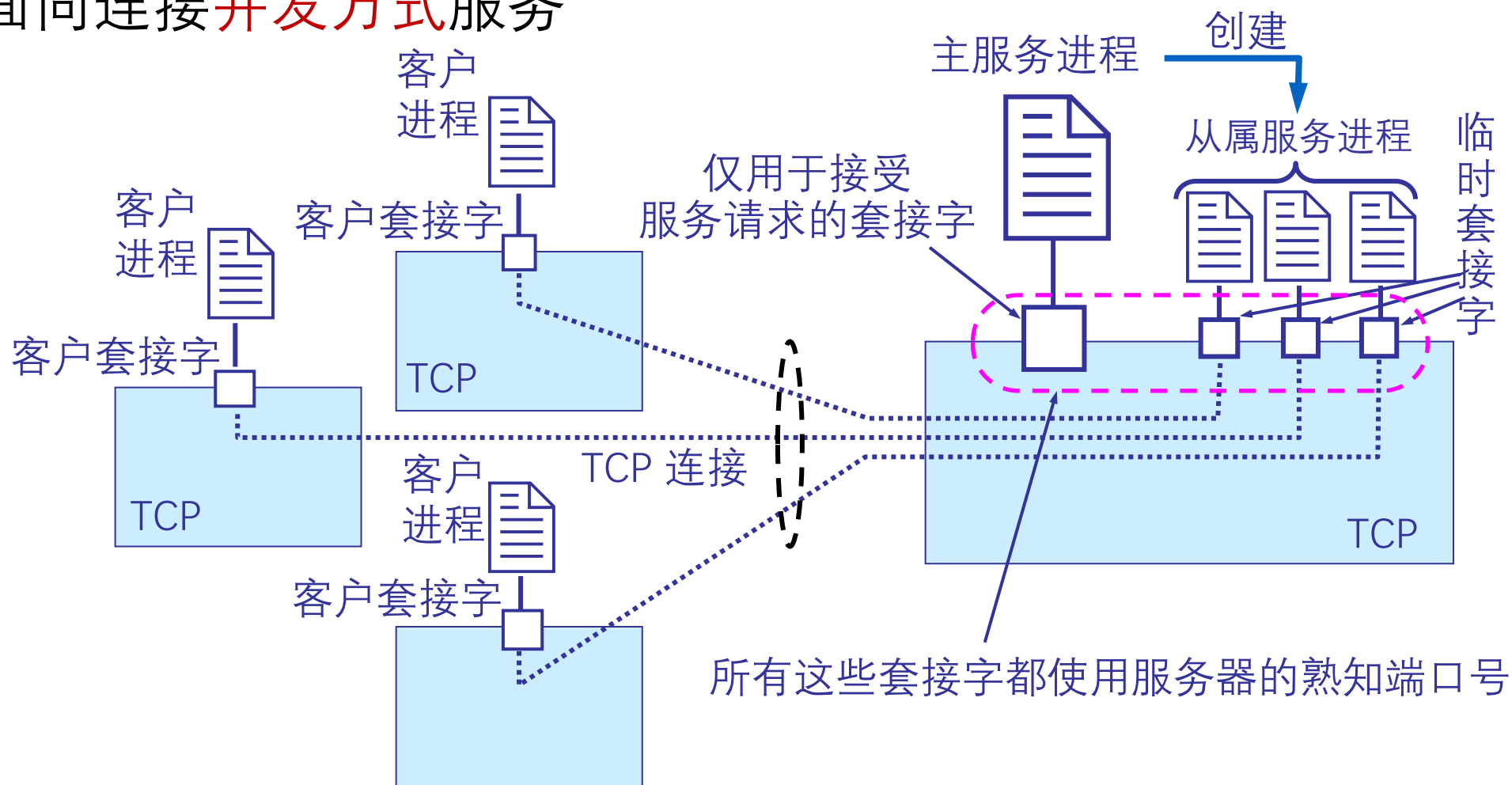
- 面向连接的TCP服务进程通常都工作在并发服务方式，服务进程在同一时间可同时向多个客户进程提供服务。(并发服务)
- 在TCP服务进程与多个客户进程之间必须建立多条TCP连接，每条TCP连接在其数据传送完毕后释放
- 一个TCP连接对应一个（熟知）服务端口
- 主服务进程在熟知端口等待客户进程发出的请求。一旦收到客户的请求，就创建一个从属服务进程，并指明从属服务进程使用临时套接字与该客户建立TCP连接，然后主服务进程继续在原来的熟知端口等待向其他客户提供服务



服务器进程工作方式



面向连接并发方式服务



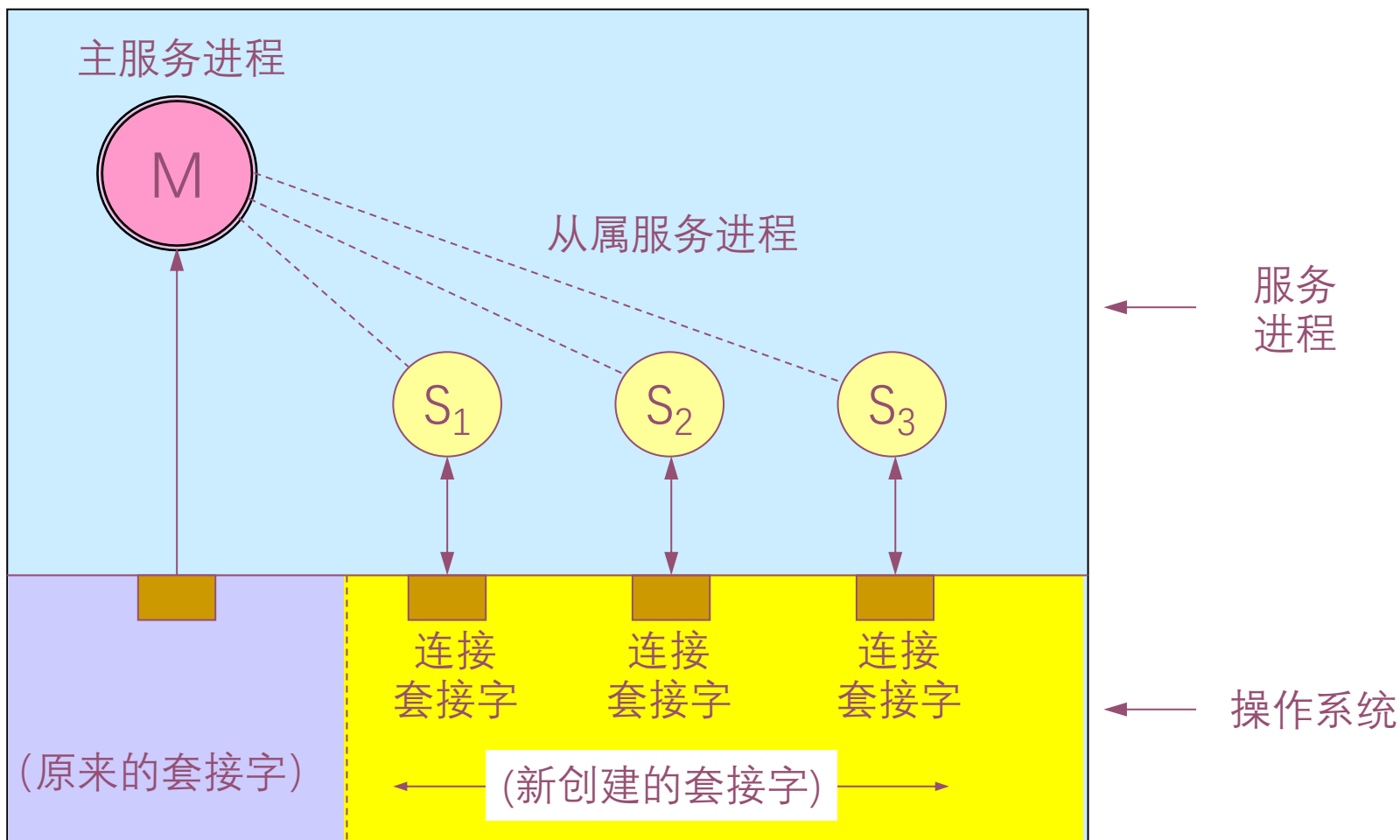
主服务进程也称为父服务进程，而从属服务进程又称为子服务进程。



服务器进程工作方式



并发方式工作的服务进程





本章内容



7.1 应用层概述 (P7)

7.2 域名系统 (P25)

7.3 电子邮件 (P74)

7.4 WWW (P106)

7.5 流式音频和视频 (P147)

7.6 内容分发 (P165)

7.7 其它应用层协议 (P182)

1. 历史和概述
2. 域名系统名字空间和层次结构
3. 域名服务器
4. 域名解析过程
5. 域名系统查询和响应(选讲)
6. 域名系统高速缓存
7. 域名系统隐私(选讲)



历史与概述



域名系统历史

➤ ARPANET时期

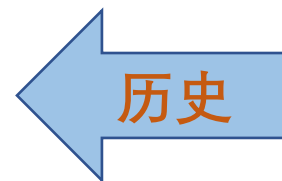
- **Hosts.txt文件**列出了所有计算机名称和它们的IP地址
- 所有主机在晚上从指定站点上获取Hosts.txt文件
- Hosts.txt文件的使用在当时表现不错

➤ 互联网发展

- Hosts.txt文件变得**越来越大**
- 需要**集中管理**来防止主机名冲突

➤ 1987年发布了域名系统的RFC文档（RFC1034、RFC1035）

- 采用了一种**层次的、基于域的**命名模式，并使用**分布式数据库系统**实现
- 主要用途是将**主机名映射成IP地址**





➤ 域名系统概述

- **域名系统** (DNS, Domain Name System) 是互联网重要的基础设施之一, 向所有需要域名解析的应用提供服务, 主要负责将可读性好的**域名映射成IP地址**
- Internet采用**层次结构的命名树**作为主机的名字, 并使用**分布式的**域名系统 DNS。Internet的DNS是一个**联机分布式数据库系统**
- 名字到域名的解析是由**若干个**域名服务器程序完成的。域名服务器程序在专设的节点上运行, 相应的节点也称为**名字服务器**(Name Server)或**域名服务器**(Domain Name Server)





域名系统名字空间和层次结构

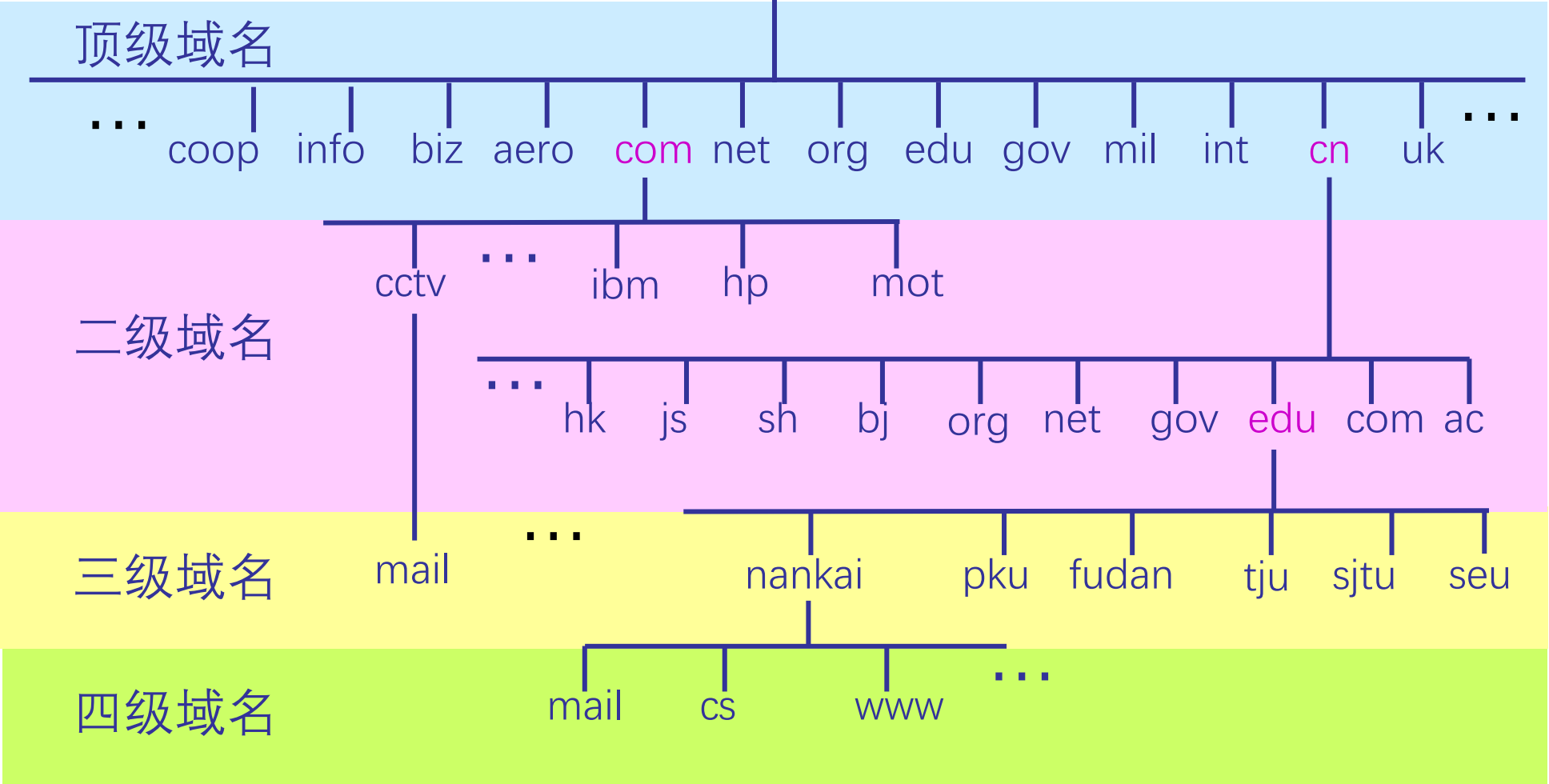


按级划分

树根

层次树状结构

域
树





域名系统名字空间和层次结构



Internet的域名结构

- Internet的域名结构采用了层次树状结构的命名方法
- 域名的结构由若干个分量组成，各分量之间用小数点(.)隔开，总长不超过255个字符
- 各分量分别代表不同级别的域名。(≤63字符)
- 合法域名中，点“.”的个数至少为一个
- 通常，点“.”对应的英文单词为dot，也可以读为point

... .三级域名.二级域名.顶级域名



域名系统名字空间和层次结构



顶级域名TLD (Top Level Domain) 一般有三类

- 国家或地区顶级域TLD，也记为ccTLD (cc: country code)
 - 例如.cn 表示中国，.us 表示美国，.uk 表示英国。目前有300多个
- 基础设施域.arpa (Address and Routing Parameter Area)
 - 专用于Internet基础设施目的
 - 目前有二级域ip6.arpa ; iris.arpa ; in-addr.arpa ; uri.arpa ; urn.arpa ; home.arpa ; as112.arpa ; in-addr-servers.arpa ; ipv4only.arpa等
- 通用顶级域gTLD
 - 早期规定了20个通用顶级域名，2011年批准新通用顶级域名 (New Generic Top-level Domain, New gTLD)
 - 截至2020年，已注册有1200多个通用顶级域名



域名系统名字空间和层次结构



早期的通用顶级域名

- .com 表示公司企业 (commerce)
- .net 表示网络服务机构 (network)
- .org 表示非赢利性组织 (organization)
- .edu 表示教育机构(美国专用) (education)
- .gov 表示政府部门(美国专用) (government)
- .mil 表示军事部门(美国专用) (military)
- .int表示政府间国际合约建立的国际性组织 (international)
- .mobi 用于提供移动产品和服务的用户和供应商 (mobile)



域名系统名字空间和层次结构



国家顶级域名 **.cn** 下的二级域名分为三类

➤ 类别域名 7 个

➤ .edu.cn 教育

➤ .gov.cn 政府

➤ .org.cn 非营利组织

➤ .net.cn 网络服务

➤ .com.cn 工商金融等企业

➤ .ac.cn 科研

➤ .mil.cn 国防机构

➤ 行政区域名 34 个：省、直辖市、自治区、特区等行政区域名，每个行政区域名为两个字母，例如北京bj、河北he等

➤ 无类别域名：例如 www.google.cn、www.tianya.cn等



域名系统名字空间和层次结构



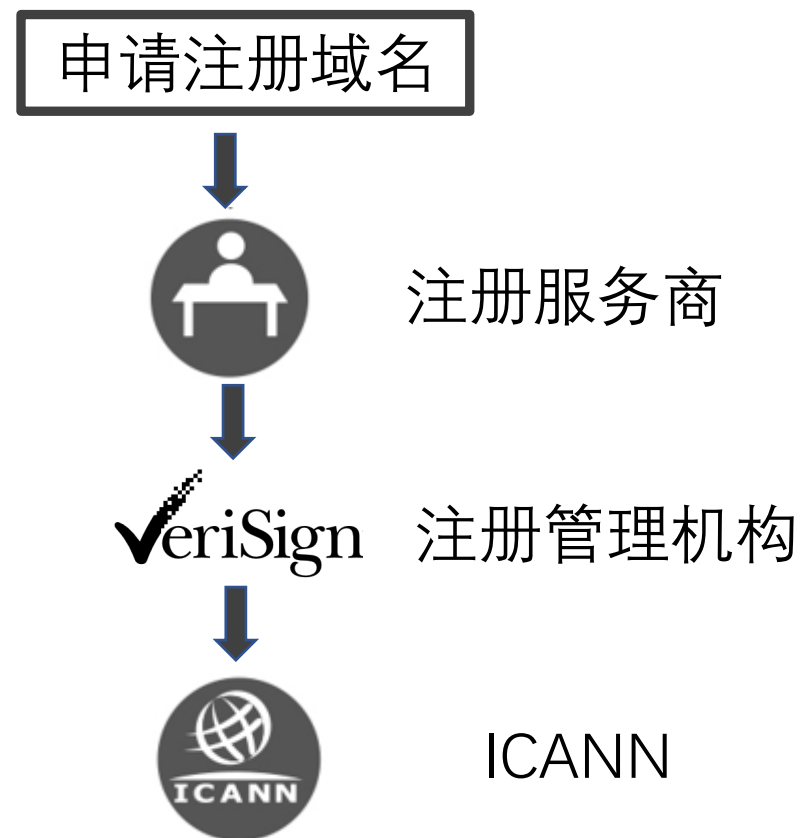
域名的管理

➤ 域名管理机构分级负责域名注册

- Internet的域名管理机构： ICANN (Internet Corporation for Assigned Names and Numbers) www.icann.org
- ccTLD (国家或地区顶级域) 下的二级域名各国自行确定
- 三级域名注册由其所属二级域名机构负责, 以此类推

➤ .edu.cn下三级域名注册由CERNET负责

➤ 我国的其它二级域名注册由中国互联网 络信息中心(CNNIC)负责



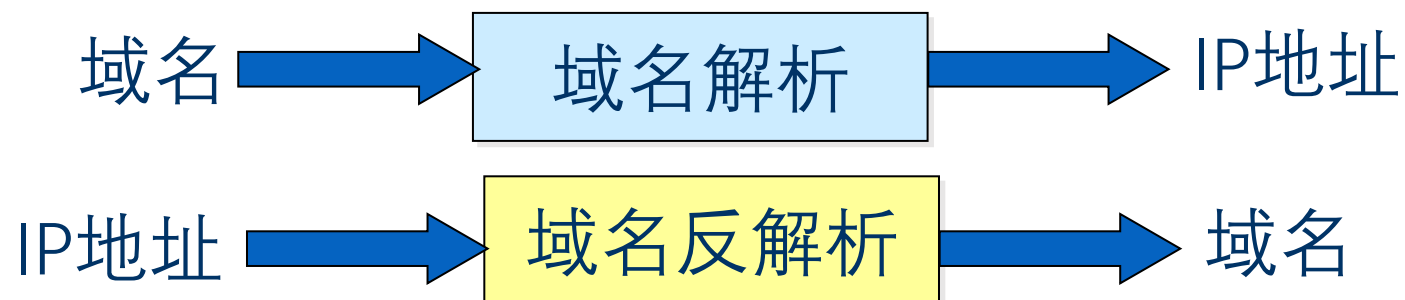
注册管理机构与注册服务商之间的关系



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，负责域名解析工作
- 每个域名服务器具有连向其它有关域名服务器的信息，当需要查询其它域的信息时，能够知道或使其它域名服务器知道到哪里去找别的域名服务器，使域名解析过程能够完成
- 域名与IP地址可以是一对一、一对多或者多对一的关系
- 域名解析过程对用户透明





域名服务器



域名系统的区域

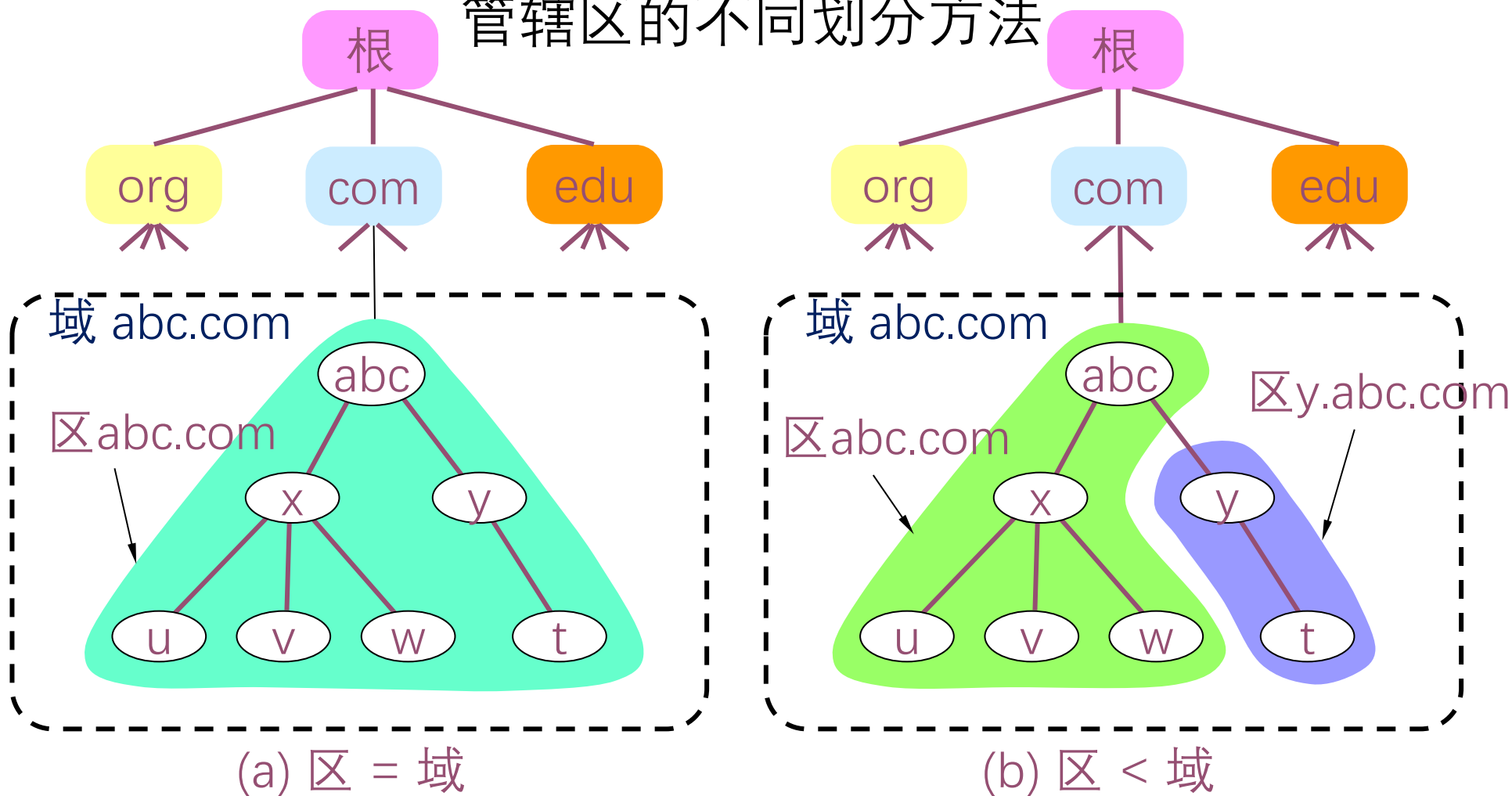
- Internet上域名服务器系统按域名层次树状安排，每个域名服务器管辖一部分域
- 一个域名服务器所负责或管辖（有权限的）范围称为管辖区(zone)，简称为区
- 各单位根据具体情况来划分自己管辖区，在一个区中的所有节点必须是能够连通的
- 域名服务器的管辖范围以“区”为单位，而不是以“域”为单位
- 管辖区是域名“域”的子集，管辖区可以小于或等于域，但不可能大于域



域名服务器



管辖区的不同划分方法





域名服务器



域名服务器类别

➤ 总体上，域名系统的域名服务器分为两大类

- **权威域名服务器**(authoritative name server)
 - 一种根据本地知识知道一个DNS区(zone)内容的服务器，它可以回答有关该DNS区的查询而无需查询其他服务器
 - 每个DNS区至少应有一个IPv4可访问的权威域名服务器提供服务
- **递归解析器**(recursive resolver)/递归服务器
 - 以递归方式运行的、使用户程序联系域(domain)名字服务器的程序。



域名服务器

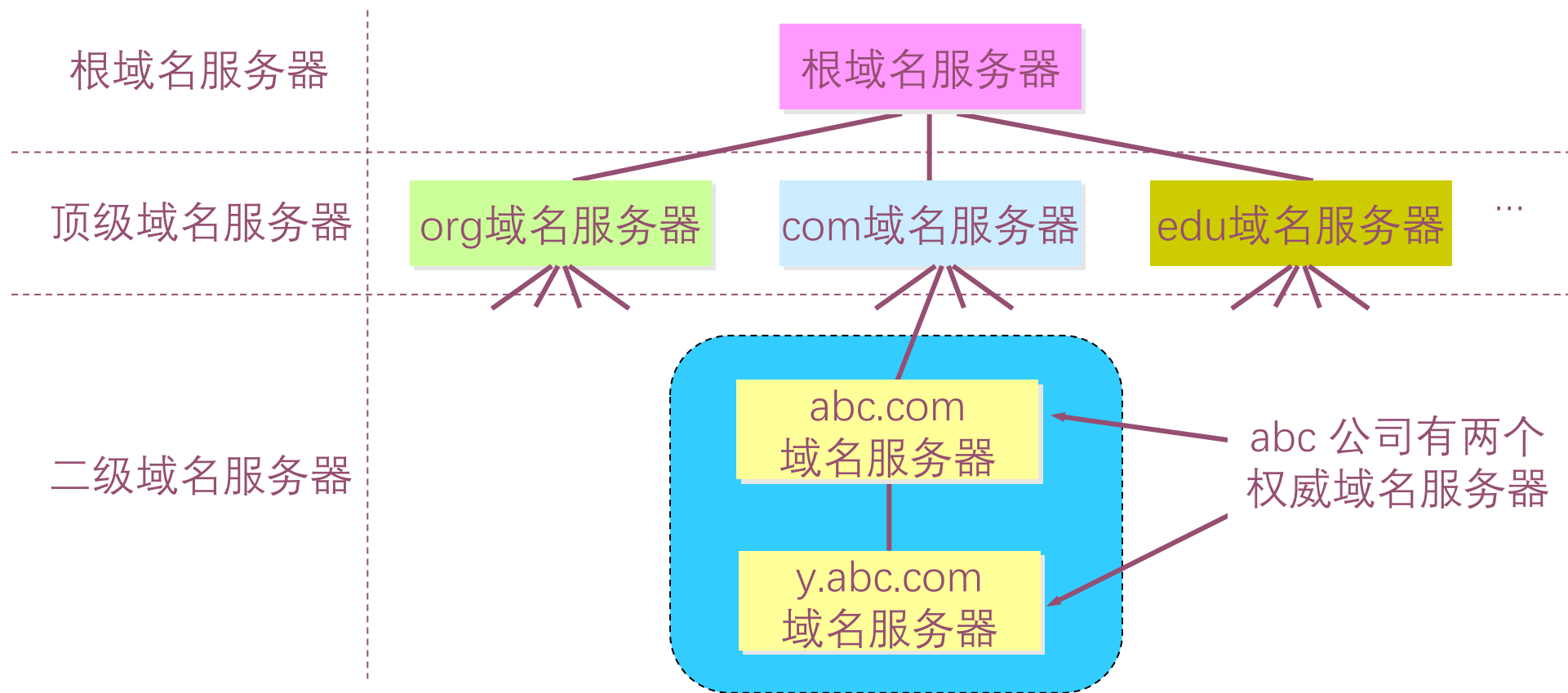


权威域名服务器类别

- 根据对应域的层次，权威域名服务器又进一步分为以下类别
 - 根域名服务器(root name server) /根服务器(root server)
 - 顶级域域名服务器(top-level domain name server)
 - 二级域域名服务器(second level domain name server)
 - 三级域域名服务器(third level domain name server)
 -
- 三级域及以下的域名服务器(例如nankai.edu.cn)通常在用户本地区域，因此三级域及以下的域名服务器也统称为**本地域名服务器**



层次树状结构的权威域名服务器





域名服务器



(1)根服务器

- 根服务器是最高层次的域名服务器，是根服务器系统云的入口点(实例)的名称。根服务器是特定的任意播(anycast)实例，即回答关于根区域内容查询的权威域名服务器
- 每个根服务器都知道所有的顶级域名服务器的域名及其IP地址
- 路由器把查询转发到距离递归服务器最近的一个根服务器(采用了任意播技术)，提高了查询效率
- 根服务器并不直接把主机用户所查的域名转换成IP地址



域名服务器



(1)根服务器

- 根服务器共有**13套**(不是13台机器)，这些根服务器相应的域名分别是：a.rootservers.net – m.rootservers.net
- 更改根服务器数据只在a.rootservers.net上进行，然后同步到另外12套中，这样既能保证数据一致性，也提高了域名服务可靠性
- 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。注意，**同步需要一定的时间才能完成**
- 根服务器之间采用任意播(anycast)技术互联。目前全球已设置了**1000多台镜像根服务器**，用户域名解析请求时延进一步降低。这样可以方便用户，使世界上大部分DNS域名服务器都能**就近**找到一个根服务器



域名服务器



(2) 顶级域名服务器

- 顶级域(TLD)域名服务器负责管理在该顶级域名服务器注册的所有二级域名
- 当收到DNS查询请求时就给出相应的回答（可能是最后的结果，也可能是下一步应当找的二级域域名服务器的 IP 地址）



域名服务器



(3)二级域域名服务器

- 每一个主机都必须在某个二级域域名服务器处注册登记。因此二级域域名服务器知道其管辖的主机名应当转换成什么IP地址
 - 各个单位根据自己的具体情况把本单位的域名划为若干个域名**管辖区(zone)**，也可简称为**区**
- **每个区**设置相应的**权威域名服务器**，用来保存该区中的所有主机的域名到IP地址的映射
- 一般来说，**每个域名服务器只负责解析本辖域的域名**



域名服务器



(4)递归解析器/递归服务器

- 每一个Internet服务提供者ISP(Internet Service Provider), 都至少有一个**递归服务器**, 距离用户主机较近
- 当用户主机发出DNS查询报文时, 这个查询报文就首先被送往该主机所在区域的递归服务器
- 如果所要查询的主机也处在本地ISP的管辖范围, 则本地域名服务器就能进行域名解析, 否则就需要再询问其他的域名服务器
- 不同网络中的递归服务器、本地域名服务器的部署层次可能不同
- 最简单的部署层次是只有一层本地域域名服务器, 即三级域域名服务器, 且同时也作为递归服务器



域名服务器



(4)递归解析器/递归服务器

- 实际部署中，解析请求路径上的递归服务器/本地域名服务器可能有多层
- 在递归服务器/本地域名服务器的多层结构中，当本层服务器没有解析结果时，通常是逐级向上递归进行查询
- 当查询请求到达本地域最上一层域名服务器时，该层服务器对自己无法解析的任何一个域名，首先求助于根服务器（而不是二级域名服务器），开始逐步迭代查询



域名解析过程概述

- 当某一应用进程需要进行域名解析时，该应用进程将域名放在DNS请求报文（UDP数据报，端口号为53）发给递归服务器（使用UDP是为了减少开销）。递归服务器得到查询结果后，将对应IP地址放在应答报文中返回给应用进程
- 域名查询有递归查询(recursive query)和迭代查询(或循环查询，iterative query)两种方式
 - 主机向递归解析器/本地域域名服务器的查询一般采用递归查询
 - 递归解析器/本地域域名服务器向根服务器可以采用递归查询，但一般优先采用迭代查询

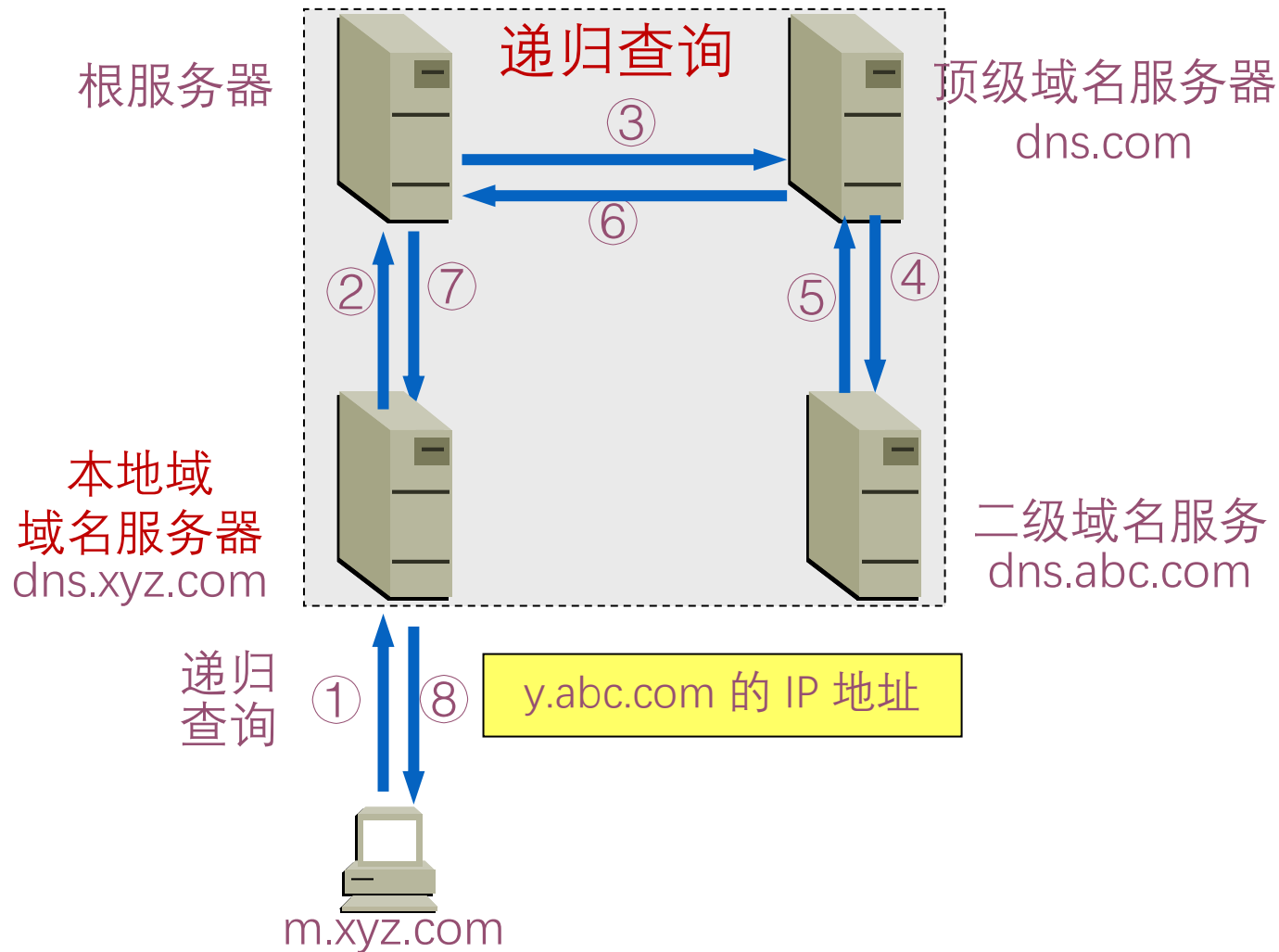


域名解析过程



➤ 递归查询

- 当收到查询请求报文的域名服务器不知被查询域名的IP地址时，该域名服务器就以DNS客户的身分向下一步应查询的域名服务器发出查询请求，即替递归服务器继续查询
- 较少使用



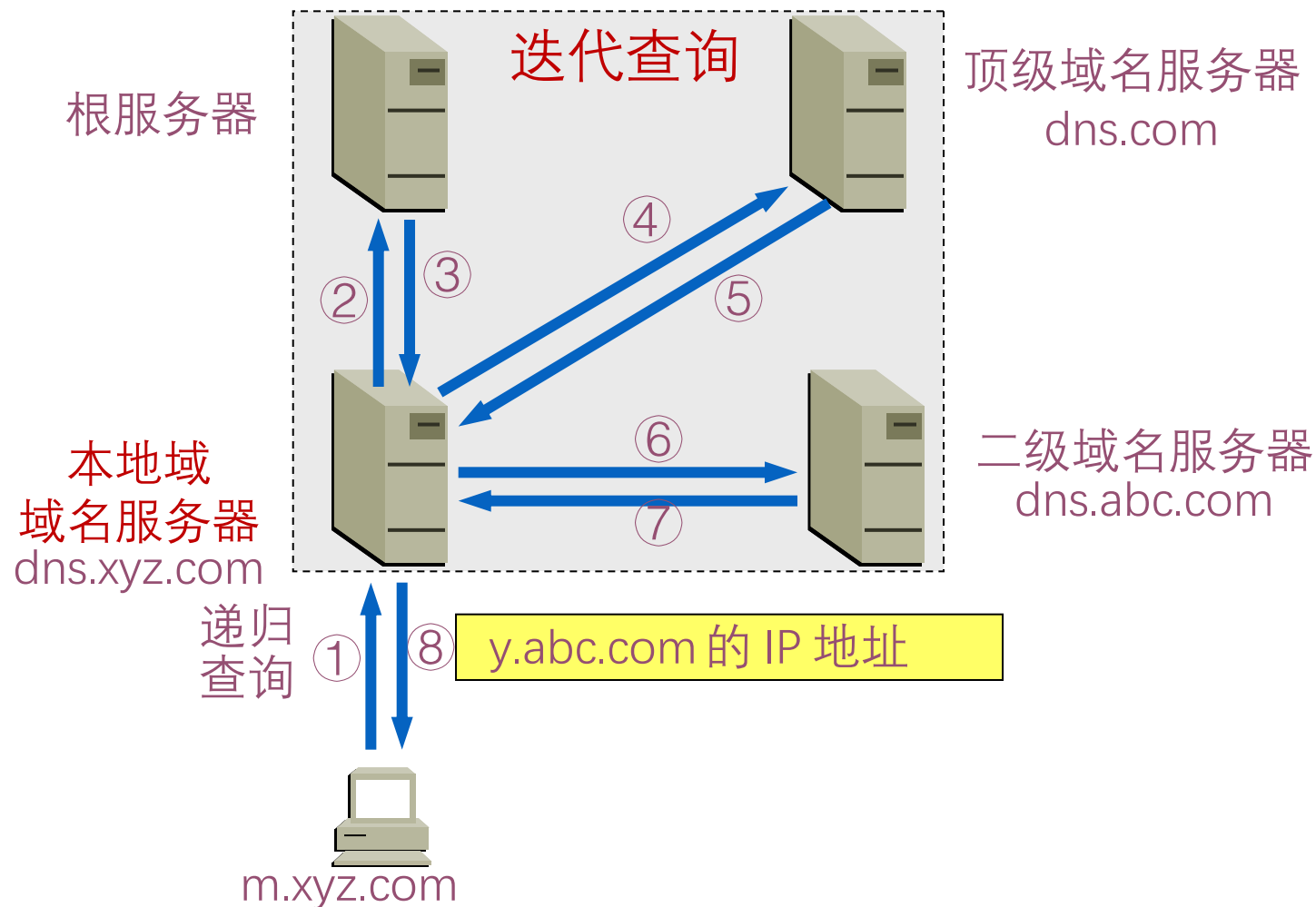


域名解析过程



➤ 迭代查询

- 当收到查询请求报文的域名服务器不知道被查询域名的IP地址时，就把自己知道的下一步应查询的域名服务器IP地址告诉**本地域名服务器**，由本地域名服务器继续向该域名服务器查询，直到得到所要解析的域名的IP地址，或者查询不到所要解析的域名的IP地址
- 通常使用





本章内容



- 7.1 应用层概述 (P7)
- 7.2 域名系统 (P25)
- 7.3 电子邮件 (P74)
- 7.4 WWW (P106)
- 7.5 流式音频和视频 (P147)
- 7.6 内容分发 (P165)
- 7.7 其它应用层协议 (P182)

1. 体系结构和服务
2. 用户代理
3. 邮件格式
4. 邮件传送
5. 最后传递



电子邮件概述



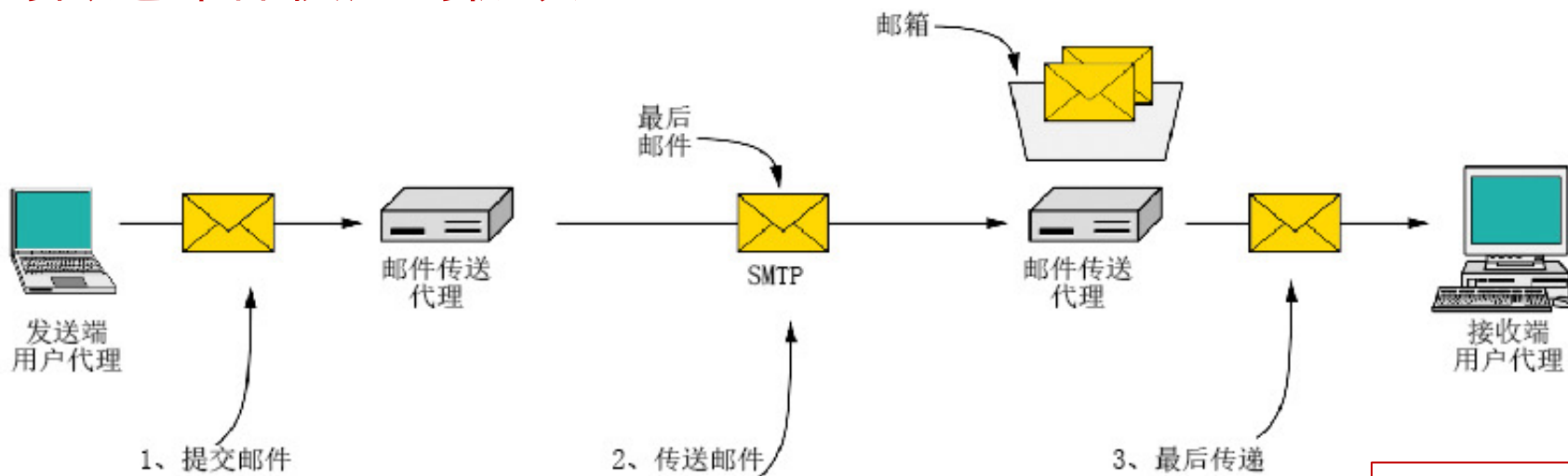
- 电子邮件（E-mail）是自早期Internet出现以来最广泛的应用，是一种异步通讯媒介
- 优越性
 - 比纸质信件更快更便宜
 - 使用门槛低
 - 收件人可随时上网到自己邮箱读取邮件
 - 电子邮件不仅可传送文字信息，而且还可附上声音和图像等



电子邮件系统体系结构



- 用户代理 (user agent) ——邮件客户端
- 传输代理 (message transfer agent) ——邮件服务器
- 简单邮件传输协议SMTP (Simple Mail Transfer Protocol) ——邮件服务器之间传递邮件使用的协议



电子邮件系统的构成

电子邮件系统采用C/S
工作模式



电子邮件系统体系结构



- 用户代理（邮件客户端）
 - 编辑和发送邮件
 - 接收、读取和管理邮件
 - 管理地址簿
 - 无统一标准
- 传输代理（邮件服务器）
 - 邮箱：保存用户收到的邮件
 - 邮件输出队列：存储等待发送的邮件
 - 运行电子邮件协议
- SMTP协议
 - smtp客户：发送邮件端
 - smtp服务器：接收邮件端

➤ 邮箱是邮件服务器中的一块内存区域，其标识即为电子邮件地址（邮箱名）

➤ 电子邮件地址是一个字符串，用于指定邮件接收者

local-part@domain-name

- domain-name: 域名
- local-part: 域名中的邮箱名

例：aaa@hotmail.com、
david@nankai.edu.cn



用户代理



- 用户代理是一个程序，用户通过它和电子邮件系统交互
- 让用户能够阅读和发送电子邮件（email message）
- 与邮件服务器交互，接收、发送电子邮件
- 常用的用户代理：
Outlook, Thunderbird, 邮箱大师, Web客户端等

邮件夹

Mail Folders
All items
Inbox
Networks
Travel
Junk Mail

Search

邮箱搜索

邮件概述

From		Subject	Received
trudy	✉	Not all Trudys are nasty	Today
Andy	✎	Material on RFID privacy	Today
djw	I	Have you seen this?	Mar 4
Amy N. Wong		Request for information	Mar 3
guido		Re: Paper acceptance	Mar 3
lazowska		More on that	Mar 2
Olivia	✎	I have an idea	Mar 2
...	

A. Student	Graduate studies?	Mar 1
Dear Professor, I recently completed my undergraduate studies with distinction at an excellent university. I will be visiting your		
...

邮件

用户代理接口的典型元素



邮件传输



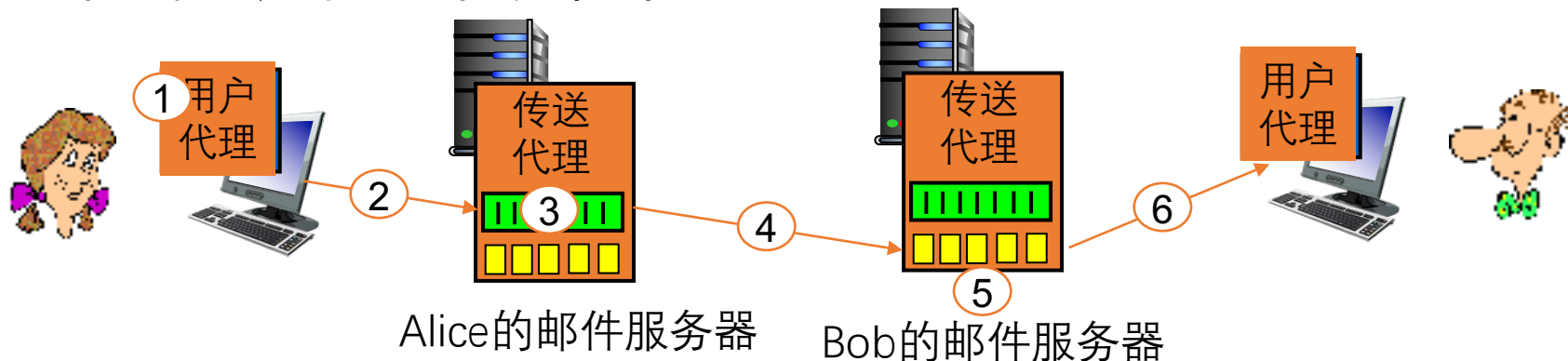
- 邮件传输代理将邮件从发件人中继给收件人
- 邮件传输采用的协议是SMTP
 - SMTP利用TCP可靠地从客户向服务器传递邮件，使用端口25
 - 直接投递: 发送端直接到接收端
 - SMTP的3个阶段：连接建立、邮件传送、连接关闭
 - 命令/响应（以HTTP为例）
 - 命令: ASCII字符串
 - 响应: 状态码+短语



场景: Alice发送邮件给Bob



- ① Alice使用用户代理撰写发送给 Bob@someschool.edu的邮件
- ② Alice的用户代理 发送邮件到她的邮件服务器；邮件存放在邮件队列
- ③ SMTP客户端打开与Bob的邮件服务器的TCP连接
- ④ SMTP客户端通过TCP连接发送Alice的邮件
- ⑤ Bob的邮件服务器把邮件存放在Bob的邮箱
- ⑥ Bob调用他的用户代理来读取邮件





POP3协议



- POP3由RFC1939定义，是一个非常简单的最终交付协议
- 当用户代理打开一个到端口110上的TCP连接后，客户/服务器开始工作
- POP3的三个阶段：
 - 认证(Authorization)：处理用户登录的过程
 - 事务处理(Transactions)：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使用客户/服务器工作方式，在接收邮件的用户PC机中必须运行POP客户程序，而在用户所连接的邮件服务器中则运行POP服务器程序



IMAP



- IMAP—Internet邮件访问协议[RFC 2060]
 - 用于最终交付的主要协议
 - IMAP是POP3的改进版
 - 邮件服务器运行侦听端口为143的IMAP服务
 - 用户代理运行一个IMAP客户端
 - 客户端连接到服务器并开始发出命令



IMAP特性



- IMAP允许用户在不同的地方使用不同的计算机随时上网阅读和处理自己的邮件
- IMAP服务器把每个邮件与一个文件夹联系起来
 - 当邮件第一次到达服务器时，它与收件人的INBOX文件夹相关联
 - 收件人能够把邮件移到一个新的文件夹中，阅读邮件，删除邮件等
- IMAP还为用户提供了在远程文件夹中查询邮件的命令，按指定条件去查询匹配的邮件
- 与POP3不同，IMAP服务器维护了IMAP会话的用户状态信息
 - 例如，文件夹的名字、哪些邮件与哪些文件夹相关联



目 录



- 7.1 应用层概述 (P7)
- 7.2 域名系统 (P25)
- 7.3 电子邮件 (P74)
- 7.4 WWW (P106)
- 7.5 流式音频和视频 (P147)
- 7.6 内容分发 (P165)
- 7.7 其它应用层协议 (P182)

1. Telnet
2. FTP
3. SNMP



Telnet



- 远程登录是网络最早提供的基本服务之一
 - 通过终端仿真协议实现对远程计算机系统的访问，就像访问本地资源一样
 - 这个过程对用户是透明的
- 远程登录需要解决异构计算机系统的差异性问题上
 - 主要体现在对终端键盘输入命令的解释上
- Telnet协议最早出现在20世纪60年代后期
- 1983年由RFC 854确定为Internet标准
- Telnet协议使用C/S方式实现
 - 在本地系统运行Telnet客户进程，在远程主机运行Telnet服务器进程
- Telnet协议使用TCP连接通信
 - 服务器进程默认监听TCP 23端口，使用主进程等待新的请求，并产生从属进程来处理每一个连接



Telnet



- Telnet协议引入网络虚拟终端NVT (Network Virtual Terminal)
 - 使用一种专门的键盘定义来屏蔽不同计算机系统对键盘输入的差异性
 - 定义客户进程与远程服务器进程之间的交互过程
- NVT是Telnet协议定义的一组通用字符集
 - 统一的数据表示方式来保证不同硬件、软件与数据格式的终端与主机之间通信的兼容性
- Telnet交互过程及特点
 - 本地终端输入的字符首先由本地Telnet客户进程转换为NVT格式
 - 通过网络将NVT格式的字符传输到远程主机
 - 远程Telnet服务器进程将NVT格式的字符转换为远程主机能够识别和处理的字符格式
 - 特点：使用Telnet协议在网络中传输的数据都是NVT格式，不同的用户终端与服务器进程均与本地终端格式无关

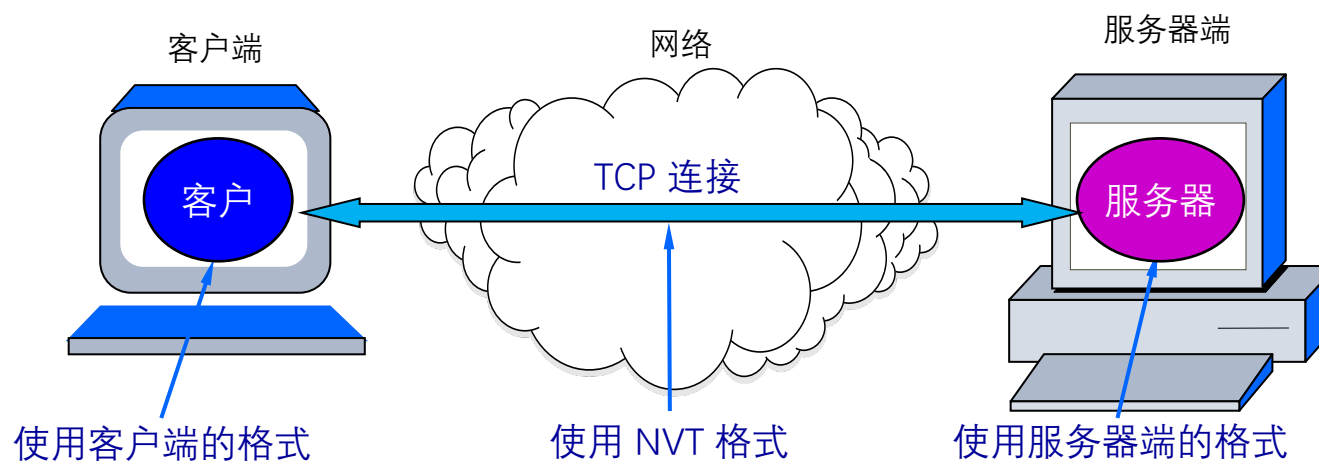


Telnet



➤ Telnet的工作过程

- 本地Telnet客户进程与远程主机上的Telnet服务器进程建立TCP连接
- 将本地终端上输入的用户名和口令及以后输入的任何命令或字符以网络虚拟终端NVT格式传输给远程主机
- 将远程主机输出的NVT格式数据转化为本地所接受的格式送回本地终端，包括输入命令回显和命令执行结果
- 本地终端对远程主机撤销连接，从而结束 Telnet远程登录过程





FTP



- 网络环境下复制文件的复杂性
 - 计算机存储数据的格式不同
 - 文件目录结构和文件命名规则不同
 - 对于相同的文件存取功能，操作系统使用的命令不同
 - 访问控制方法不同
- 文件传输协议FTP(File Transfer Protocol)是Internet上使用最广泛的应用层协议之一
 - FTP提供交互式的访问，允许用户指明文件的类型与格式，并允许文件具有存取权限
 - FTP屏蔽了各计算机系统的细节，适用于在异构网络中任意计算机之间传送文件
 - RFC 959早在1985年就已经成为Internet的正式标准



FTP



- FTP使用C/S方式实现
- FTP工作过程
 - 服务器主进程打开TCP 21端口，等待客户进程发出的连接请求
 - 客户可以用分配的任意一个本地端口号与服务器进程的TCP 21端口进行连接
 - 客户请求到来时，服务器主进程启动从属进程来处理客户进程发来的请求
 - 服务器从属进程对客户进程的请求处理完毕后即终止，但从属进程在运行期间根据需要还可能创建其他子进程
 - 服务器主进程返回，继续等待接收其他客户进程发来的连接请求，服务器主进程与从属进程并行工作

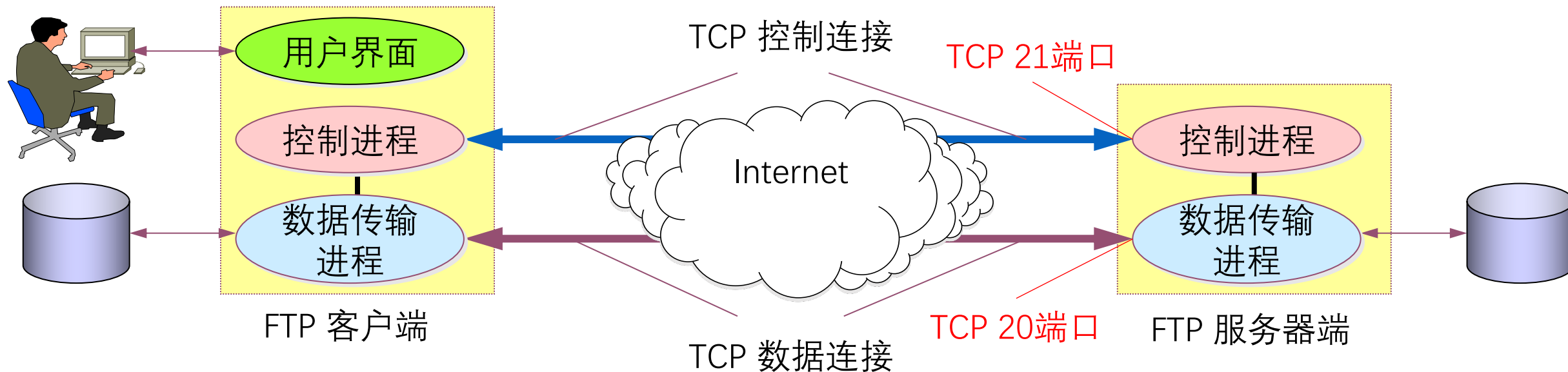


FTP



➤ FTP的两个端口与两个连接

- **控制连接**在整个会话期间一直保持，客户进程发出的文件传输请求通过控制连接发送给**服务器控制进程（工作在TCP 21端口）**，但控制连接不用来传输文件
- 服务器控制进程在接收到客户进程发送来的文件传输请求后就创建**数据传输进程（工作在TCP 20端口）**和**数据连接**
- 数据连接用来连接客户进程和服务端数据传输进程，实际完成文件的传输。服务器数据传输进程在文件传输完毕后关闭数据连接并结束运行





FTP



➤ 简单文件传输协议 TFTP

- TFTP(Trivial File Transfer Protocol) 是一个很小且易于实现的文件传输协议
- 使用C/S方式和UDP协议实现
- 只支持文件传输而不支持交互
- 没有庞大的命令集，没有列目录的功能，也不能对用户进行身份鉴别
- 支持ASCII 码或二进制传送
- 支持对文件进行读或写
- 使用很简单的首部



SNMP (Simple Network Management Protocol)

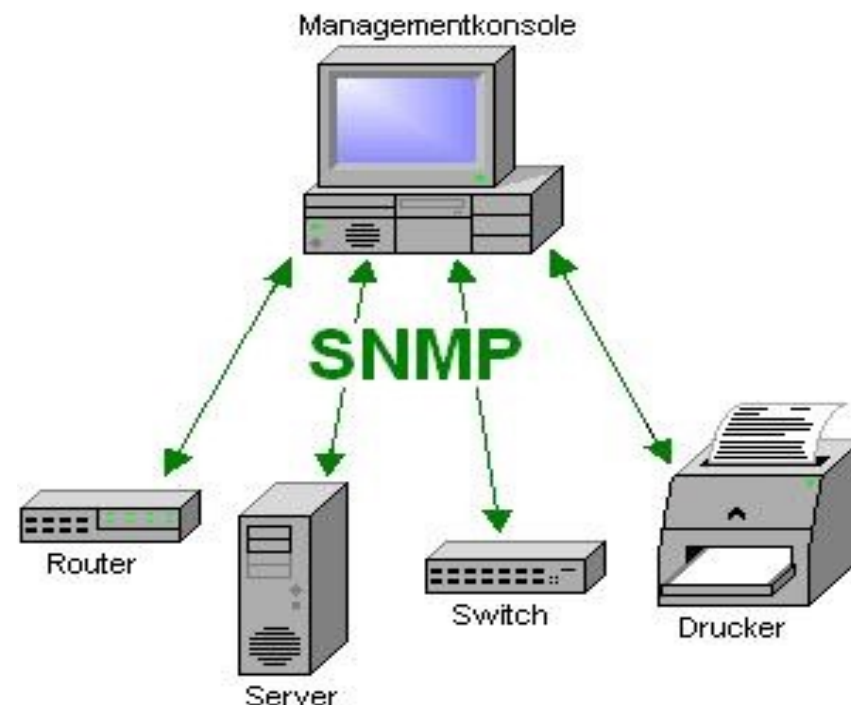


➤ 网络管理的目的

- 有效利用网络资源
- 及时报告和处理网络故障
- 保障网络正常、高效运行

➤ 网络管理的功能

- 配置管理 (Configuration Management)
- 故障管理 (Fault Management)
- 性能管理 (Performance Management)
- 计费管理 (Accounting Management)
- 安全管理 (Security Management)





➤ 网络管理系统

- 管理器 (Manager) : 网络管理进程, 提供网络管理用户界面, 完成管理任务
- 管理对象 (Managed Object) : 网络中的软硬件系统
- 代理 (Agent)
 - 管理代理 (Management Agent) : 管理对象中的管理进程
 - 外部代理 (Proxy Agent) : 管理对象外部增加的管理进程
- 管理信息库 (MIB, Management Information Base)
 - 本地MIB : 每个代理管理自己的本地MIB, 仅包含本地设备相关信息
 - 网络MIB : 代理与管理器交换网络状态信息, 共同构成整个网络的MIB
- 网络管理协议 (Network Management Protocol)
 - TCP/IP的简单网络管理协议SNMP (Simple Network Management Protocol)
 - OSI的公共管理信息协议CMIP (Common Management Information Protocol)



SNMP



➤ SNMP概述

- 指导思想
 - **尽可能简单**：用简单的系统结构和协议来解决复杂的网络管理问题
 - 协议交互过程简单，仅有4类操作对网络进行管理
 - 使用**UDP协议**，以简化设计和降低通信开销
 - **影响尽量小**：网络中任何软硬件的增减对原有管理对象的影响尽量小
- 版本变迁
 - SNMPv1：RFC 1067(1988)、RFC 1098(1989)、RFC 1155~1157(1990)
 - SNMPsec：RFC 1351~1353(1992)
 - SNMPv2：RFC 1441~1445(1993)
 - **SNMPv3**：RFC 3410~3418(2002)，安全性和可管理性大为提高



SNMP



- SNMP协议实现
 - SNMP使用无连接的UDP协议实现
 - SNMP使用C/S方式实现
- SNMP是有效的网络管理协议
 - 使用周期性轮询方式实现对网络资源的实时监控与管理
 - 使用陷阱方式报告重要事件的发生



本章目标



1. 掌握应用进程通信方式以及服务进程工作模式
2. 掌握域名系统基本原理和工作机制
3. 掌握电子邮件系统体系结构及基本工作原理
4. 了解WWW系统结构框架、静态和动态Web及其应用技术，掌握HTTP协议及其工作原理
5. 了解流媒体基本概念、数字音视频与编码、流式存储媒体、直播与实时音视频、流媒体动态自适应传输
6. 了解内容分发背景及内容分发网络、以及P2P网络的工作机制
7. 掌握Telnet、FTP、SNMP等应用层协议的工作原理



欢迎大家选修 “软件测试与维护”

不考试！



谢谢！