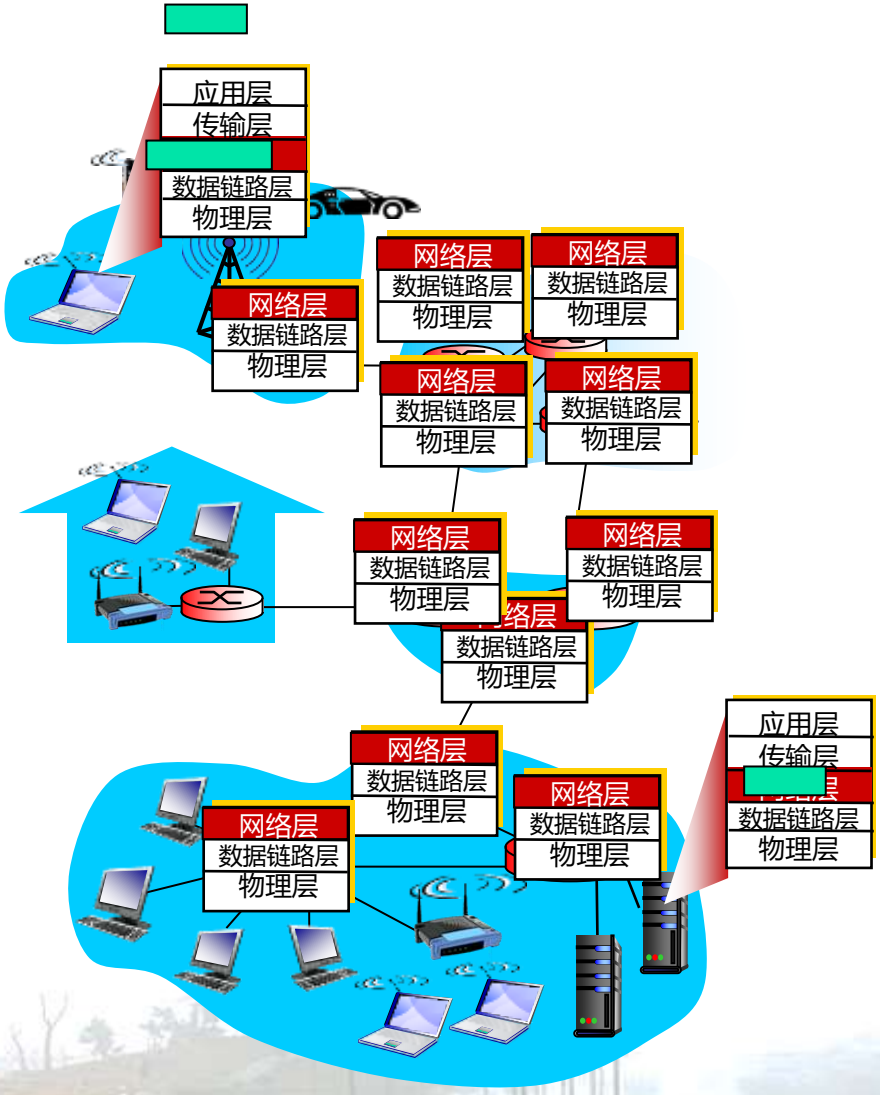
中国科学技术大学
University of Science and Technology of China

第四章 网络层数据平面

目录

网络层

- 将传输层分段从源送到目的地
- 在发送端将分段封装为网络层报文
- 在接收端从报文中提取分段，交给传输层
- 所有的主机、路由器都运行网络层
- 路由器检查所有经过的IP报文头部



两大关键功能

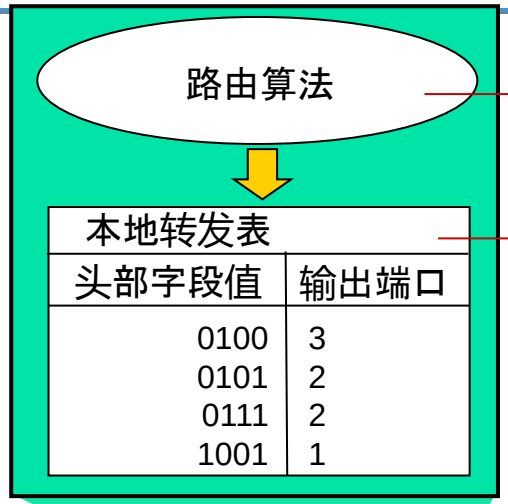
网络层功能：

- 转发： 将数据包从路由器的入端口转移到正确的出端口
- 路由: 决定数据包从源到目的地的转发路径
  - 使用路由算法计算

类比：自驾出游

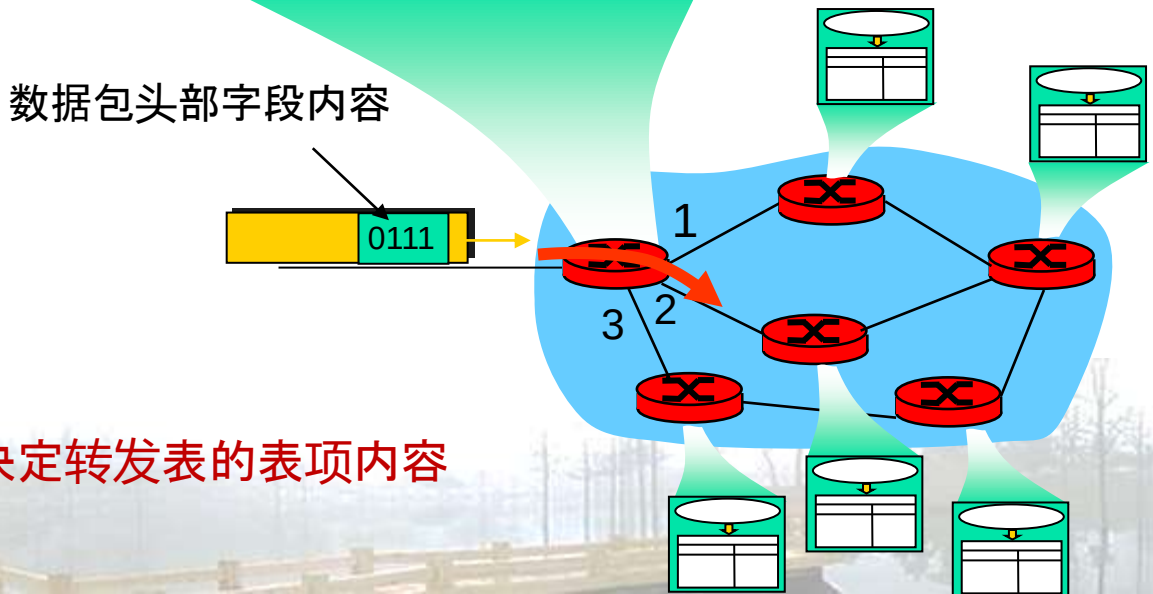
- 转发：在每一个路口的具体行为（等绿灯、左转、右转、直行）
- 路由：规划路线

路由和转发的关系



路由算法计算端到端路径

转发表决定路由器本地的转发行为



路由协议决定转发表的表项内容

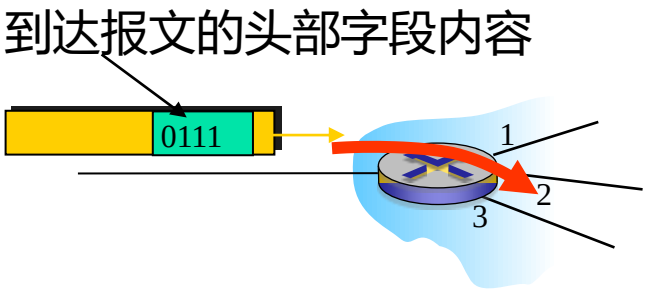
建立连接

- 路由和转发以外的第三种功能，仅限于某些网络体系结构
  - ATM、帧中继、X.25
- 两端中继在传输数据前先建立虚拟电路
  - 需要路由器协助支持
- 网络层连接 vs 传输层连接
  - 网络层：两台主机之间（一些情况需要沿途路由器支持）
  - 传输层：两个进程之间

网络层：数据平面、控制平面

数据平面

- 每个路由器都具备的本地能力
- 决定如何将报文从入端口转发到出端口
- 转发功能

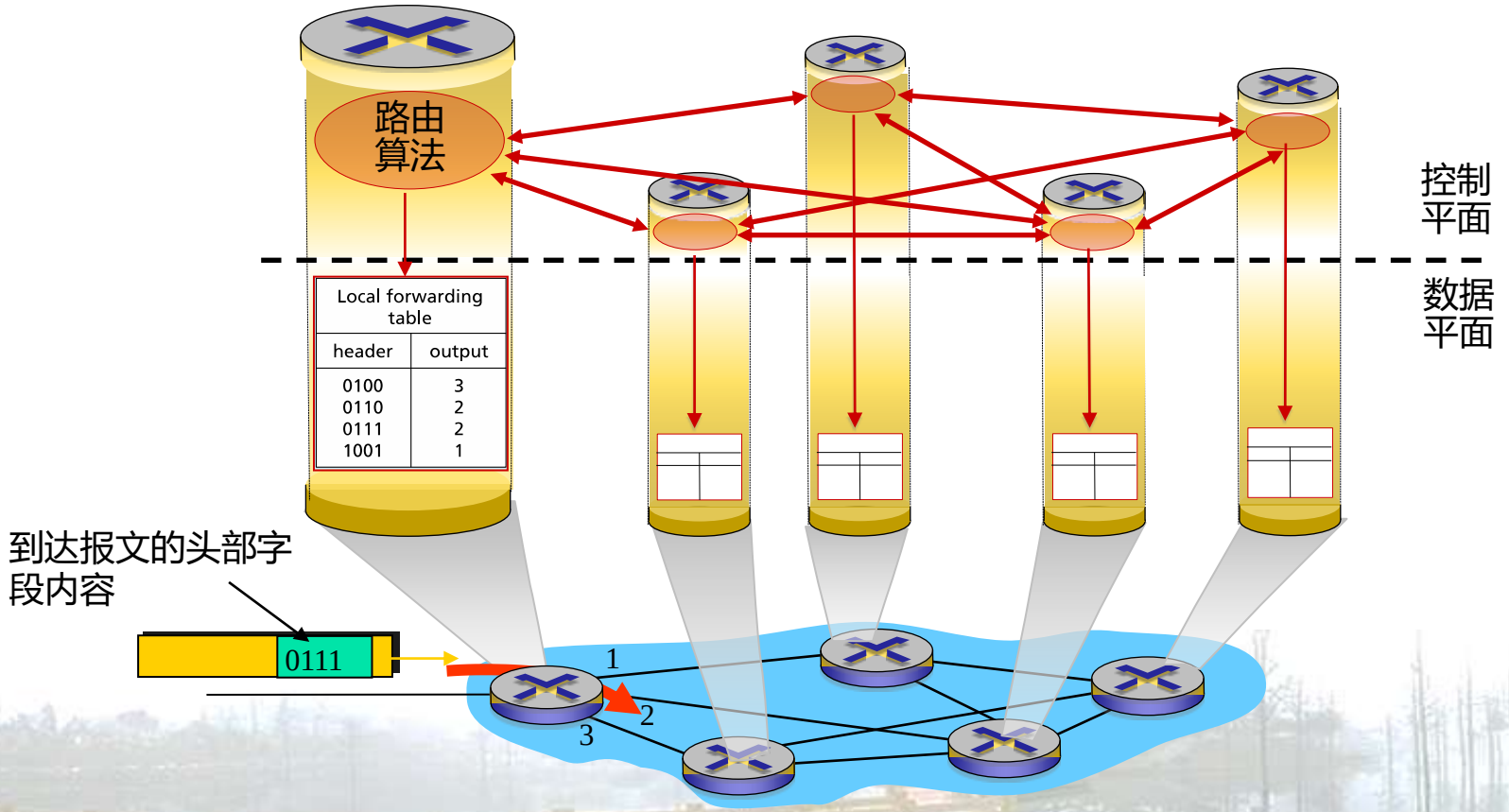


控制平面

- 整个网络具备的能力
- 决定报文从源到目的地的，由沿途路由器路由的，端到端路径
- 两类控制平面方案:
  - 传统的路由算法:在每个路由器上实现
  - 软件定义网络 (SDN): 逻辑上集中实现

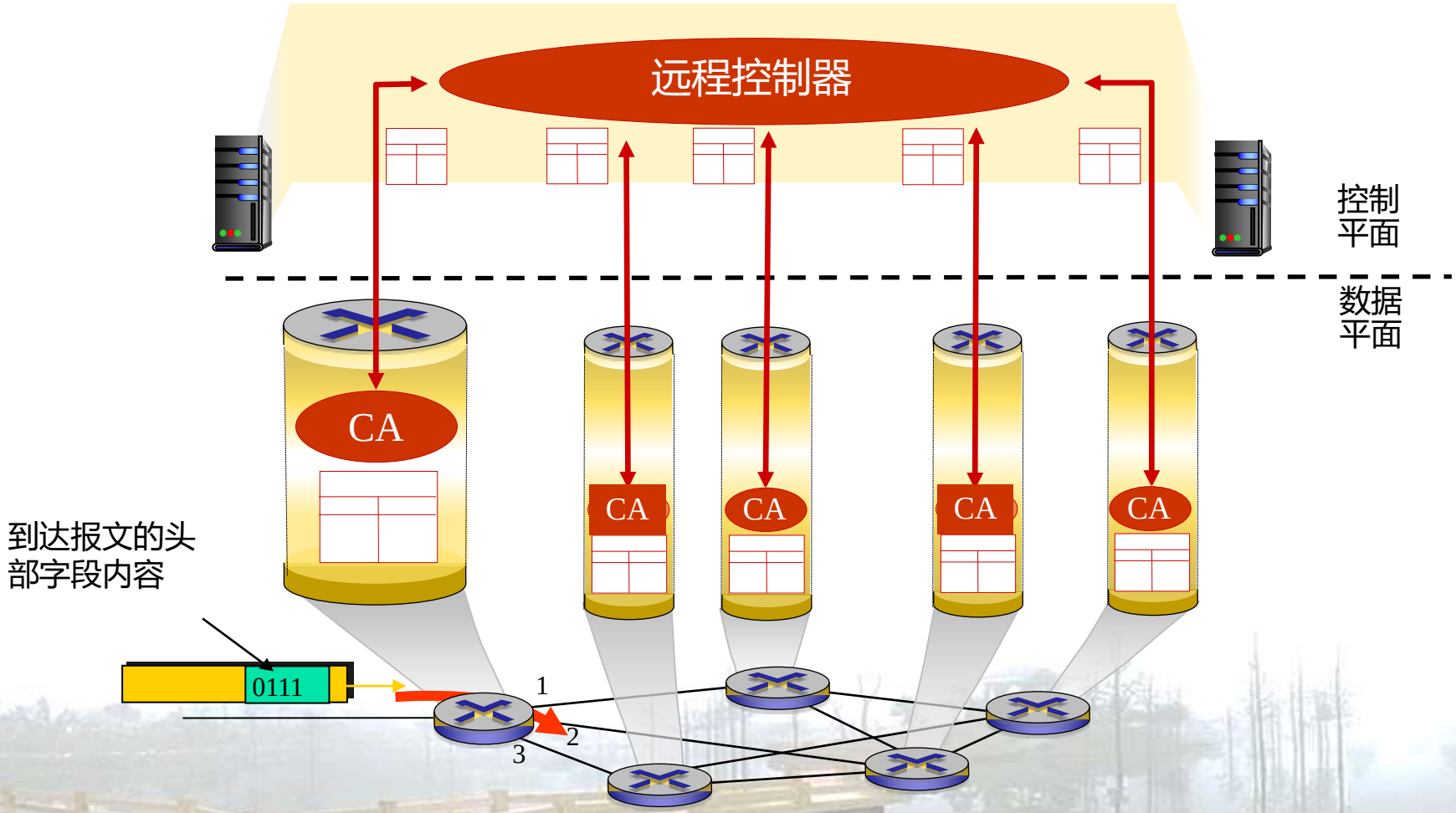
每个路由器参与的控制平面

每个路由器中都有单独的路由算法模块，它们相互交互，一起构成控制平面



逻辑上集中的控制平面

单独的（远程）控制器与路由器本地的控制代理（CA）交互



网络层服务模型

问：作为从发送端到接收端的逻辑信道，提供怎样的服务模型？

为单个报文提供服务的例子:

- 确保送到
- 确保在40毫秒内送到

为一个报文流（flow）提供服务的例子:

- 顺序送达
- 最小带宽保障
- 报文到达时间间隔约束

网络层的服务模型

网络架构	服务模型	保障				拥塞反馈
		带宽	丢包	顺序	限时	
因特网	尽力而为	无	无	无	无	无（由丢包推测）
ATM	CBR	固定比特率	有	有	有	无拥塞
ATM	VBR	保障速率	有	有	有	无拥塞
ATM	ABR	保障最低速率	无	有	无	有
ATM	UBR	无	无	有	无	无

ATM：Asynchronous Transfer Mode，异步传输模式，一种网络交换技术

目录

- 4.1 网络层概览
  - 数据平面
  - 控制平面
- 4.2 虚电路和数据报网络
- 4.3 路由器内部结构和功能
- 4.4 IP协议
  - 报文格式
  - 分片
  - IPv4地址
  - 地址翻译
  - IPv6
- ~~4.5 SDN初步~~

有连接和无连接服务

- 数据报（datagram）网络提供无连接的网络层服务
  - 例如，IP网络
- 虚电路（virtual-circuit）网络提供有连接的网络层服务
  - 例如， ATM网络
- 类似于TCP/UDP在传输层提供面向连接和无连接的服务，但是
  - 网络层提供主机到主机的服务，而非进程到进程
  - 有连接和无连接的服务不能共存，非此即彼
  - 由网络核心部分（路由器）实现

虚电路

"源到目的地的路径类似电话线路"

- 性能保障
- 网络操作的执行对象是整个路径，而非单个路由器

- 在数据传输前后建立和撤销呼叫
- 每个数据包携带虚电路标识，而非目的地主机的地址
- 每个路由器对经过的连接维护状态
- 链路和路由器资源可能被分配到虚电路上（资源独享而非共享）

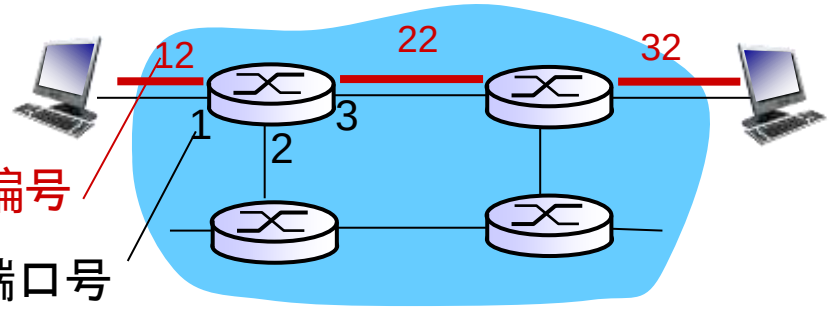
虚电路实现

一条虚电路包含

1. 源到目的地的路径
2. VC编号，路径上每条链路有一个编号
3. 路径上沿途路由器转发表项

- 属于某条虚电路的数据包携带虚电路编号（而非目的地址）
- 同一条虚电路的编号在不同链路上可能不同
  - 对每个数据包，路由器将旧的虚电路编号替换为新的
  - 新的虚电路编号由转发表确定

虚电路转发表



左上位置的路由器转发表

输入端口	输入数据包虚电路编号	输出端口	输出数据包虚电路编号
1	12	3	22
2	63	1	18
3	7	2	17
1	97	3	87
…	…	…	…

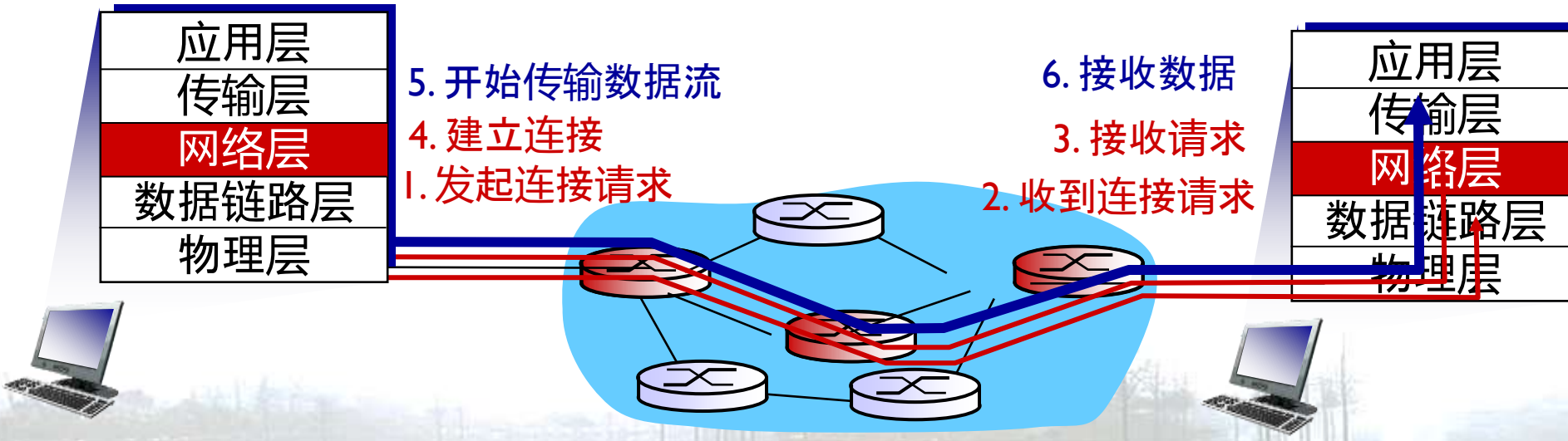
虚电路路由器维护连接状态

虚电路生命期

- 建立虚电路：确定源到目的地路径；沿途路由器确定每条链路的虚电路编号；在转发表中添加表项
- 数据传输：数据包沿路径传输
- 撤销虚电路：更新路由器转发表

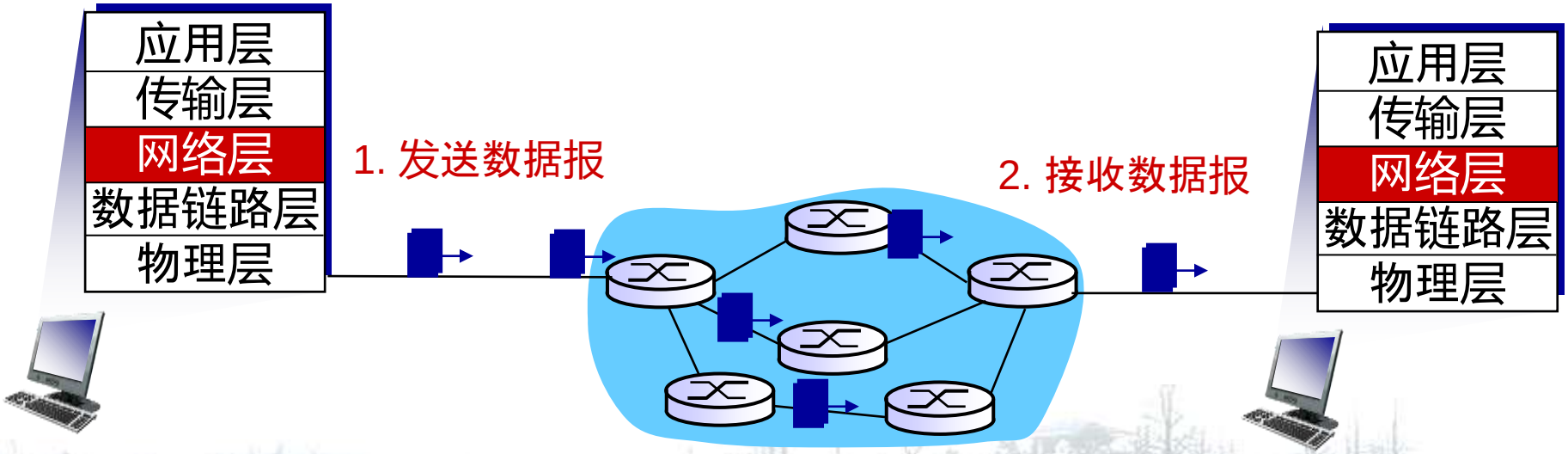
虚电路信号协议

- 用于建立、保持和撤销虚电路
- ATM, 帧中继, X.25
- 因特网不采用虚电路

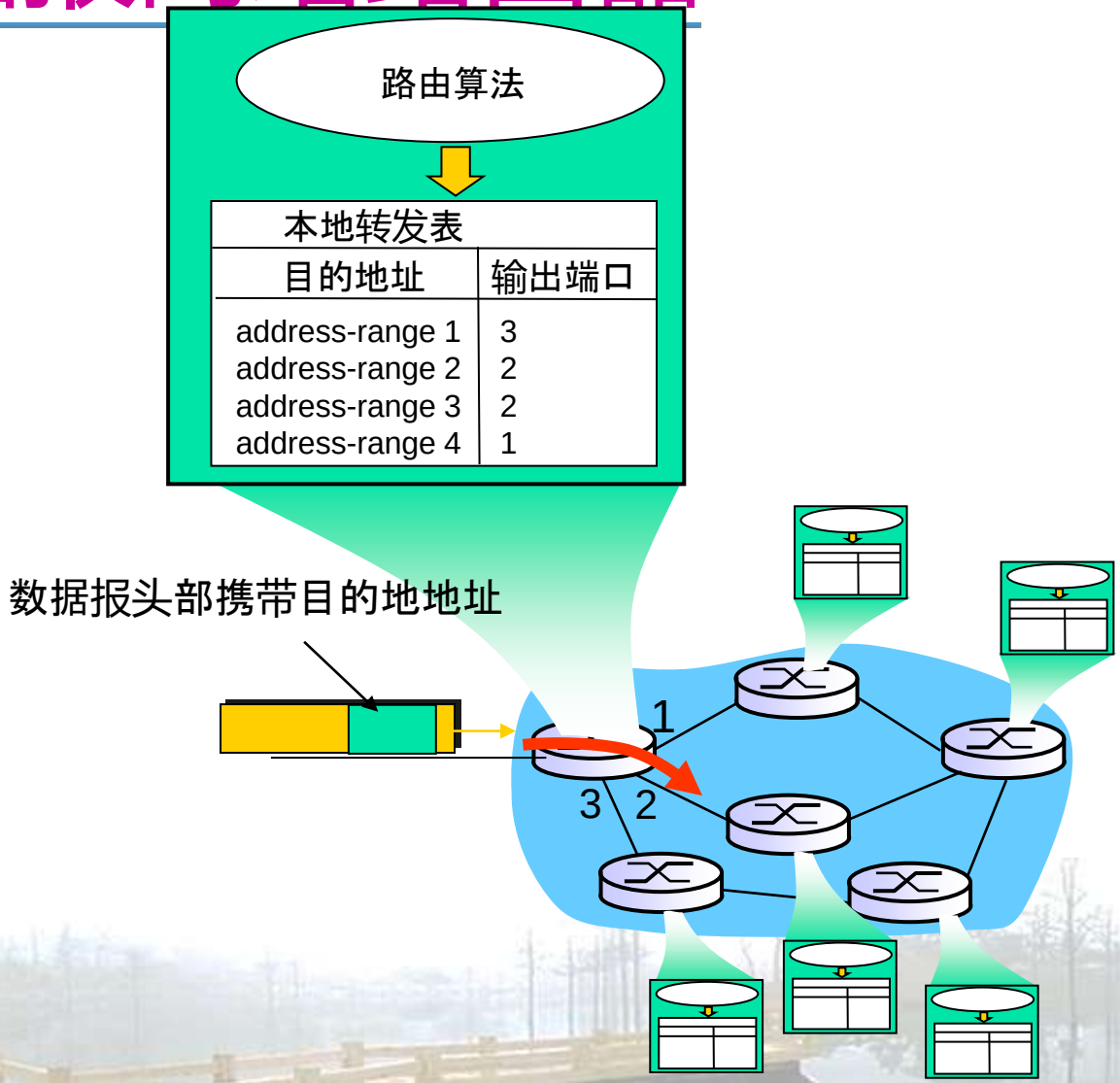


数据报网络

- 不建立网络层连接
- 路由器不保持端到端连接的状态
- 路由器依据数据包的目的地地址转发数据包



数据报网络路由器


路由算法
本地转发表
目的地址
输出端口
address-range 1
address-range 2
address-range 3
address-range 4
3
2
2
1
数据报头部携带目的地地址
1
3
2

数据报网络路由器转发表

目的地址范围	链路端口号	
11001000 00010111 00010000 00000000 到 11001000 00010111 00010111 11111111	0	200.23.16.0 - 200.23.23.255
11001000 00010111 00011000 00000000 到 11001000 00010111 00011000 11111111	1	200.23.24.0 - 200.23.24.255
11001000 00010111 00011001 00000000 到 11001000 00010111 00011111 11111111	2	200.23.25.0 - 200.23.31.255
其它	3	

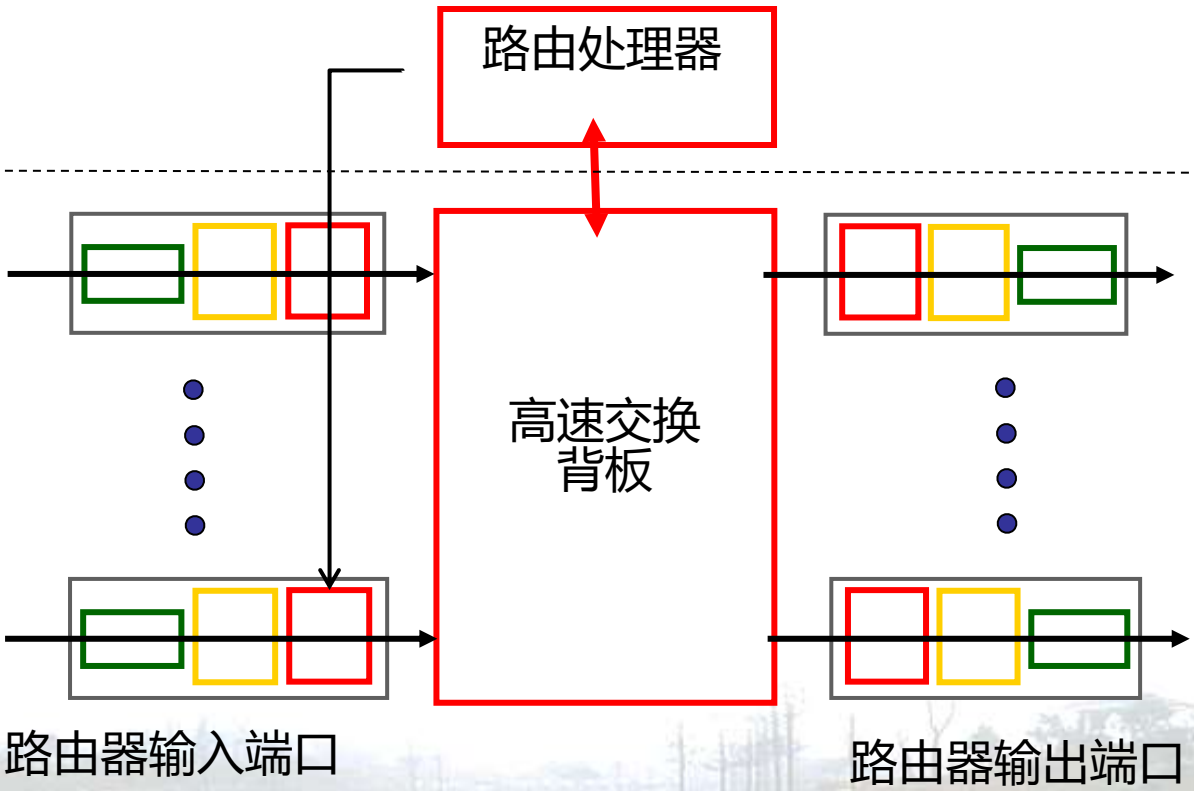
数据报路由

- 路由器在转发表中保持转发状态信息
- 转发表项由路由算法确定
  - 1-5分钟更新一次
  - 转发表项的内容和当前路由器中是否存在相关地址的数据包没有关系
- 从相同的源地址到目的地址的数据报可能经过不同的路径

目录

路由器体系结构概览

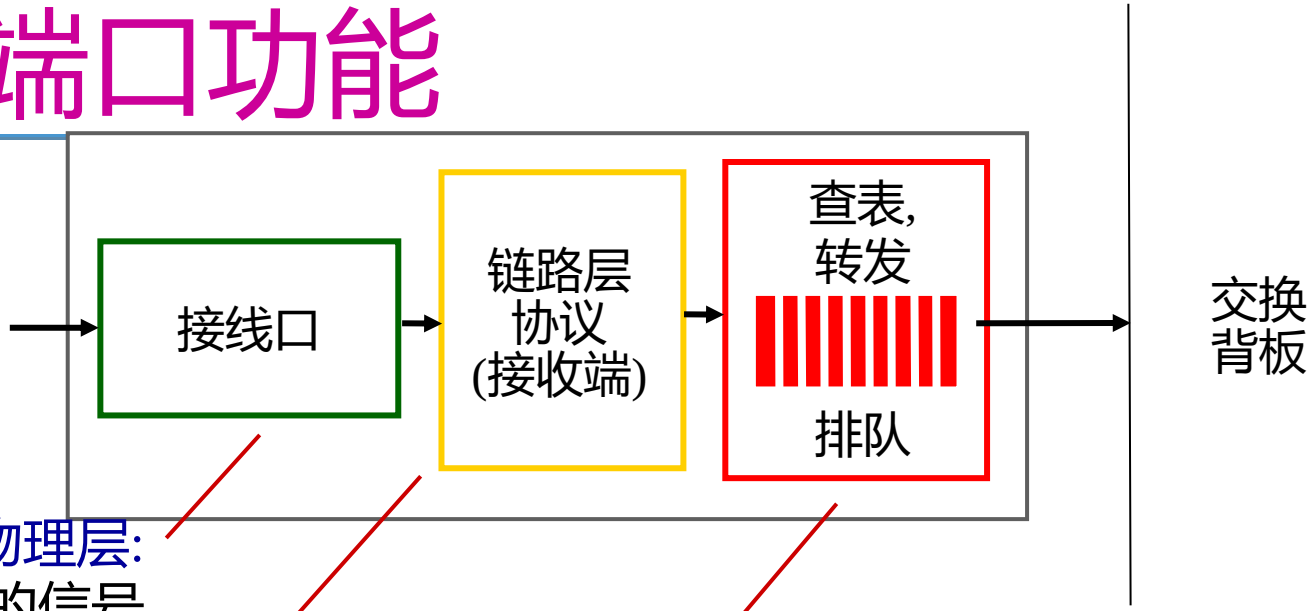
- 路由器体系结构示意图



路由管理控制平面 (软件)
毫秒级别的操作

转发数据平面 (硬件)
纳秒级别的操作

输入端口功能



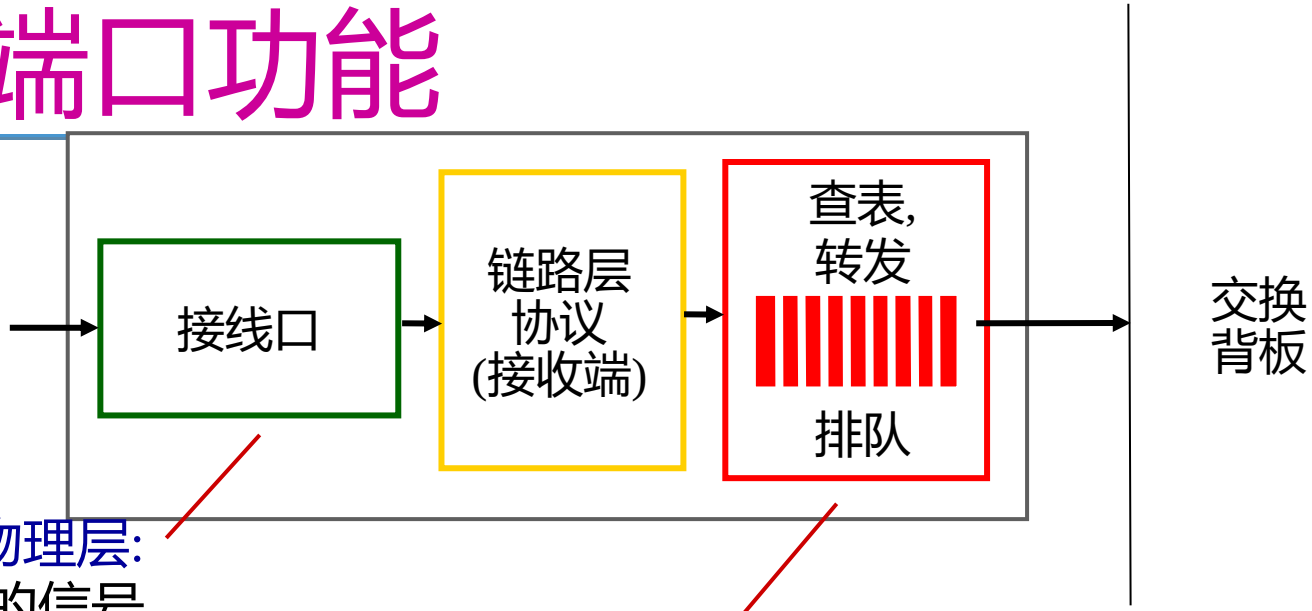
物理层:
接收bit级别的信号

链路层:
例如，以太网、WiFi

分布式交换:

- 使用报文头部字段，查询转发表，确定报文的输出端口。转发表存储在输入端口的内存中（匹配-转发）
- 以线速处理数据包
- 排队: 如果接收数据包的速率高于将数据包发送到交换背板的速率，暂存在队列中

输入端口功能



物理层:
接收bit级别的信号

链路层:
例如，以太网、WiFi

分布式交换:

- 使用报文头部字段，查询转发表，确定报文的输出端口。转发表存储在输入端口的内存中（匹配-转发）
- 基于目的地的转发：依据报文的目的地地址确定输出端口，或者
- 广义转发：依据任意字段内容确定输出端口

基于目的地的转发

转发表

目的地地址范围	链路端口
11001000 00010111 00010000 00000000 到 11001000 00010111 00010111 11111111	0
11001000 00010111 00011000 00000000 到 11001000 00010111 00011000 11111111	1
11001000 00010111 00011001 00000000 到 11001000 00010111 00011111 11111111	2
其余	3

问：如果地址范围有重叠怎么办？

最长前缀匹配（LPM）

最长前缀匹配

当目的地址落在多个表项的范围内，使用产生最长前缀匹配的表项。

目的地地址范围	链路端口
11001000 00010111 00010*** *********	0
11001000 00010111 00011000 *********	1
11001000 00010111 00011*** *********	2
其它	3

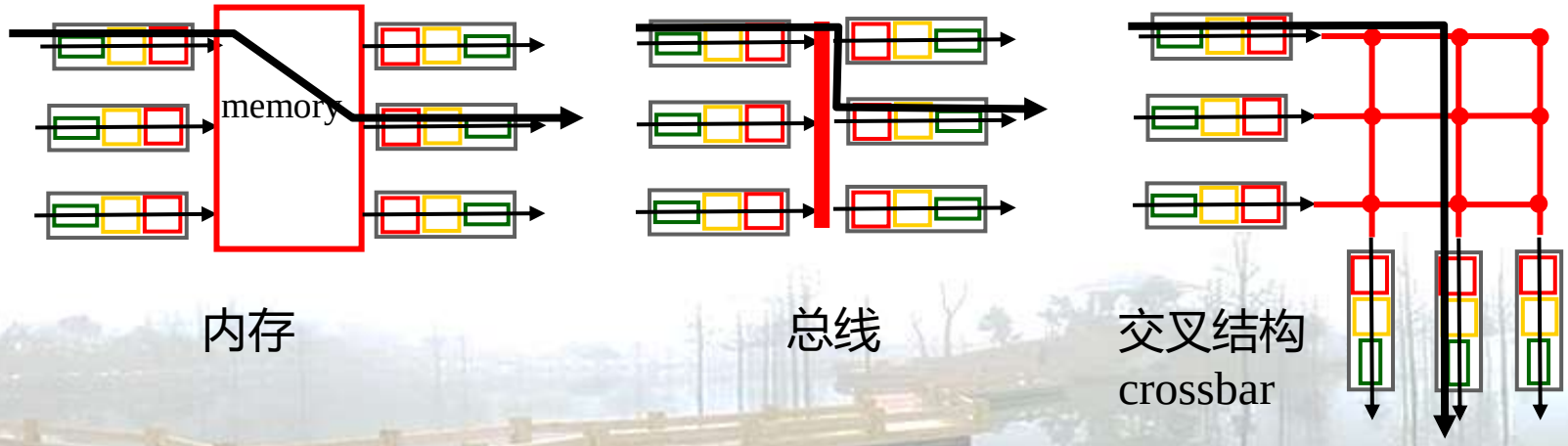
例子：

目的地址: 11001000 00010111 00010110 10100001 哪个端口输出?

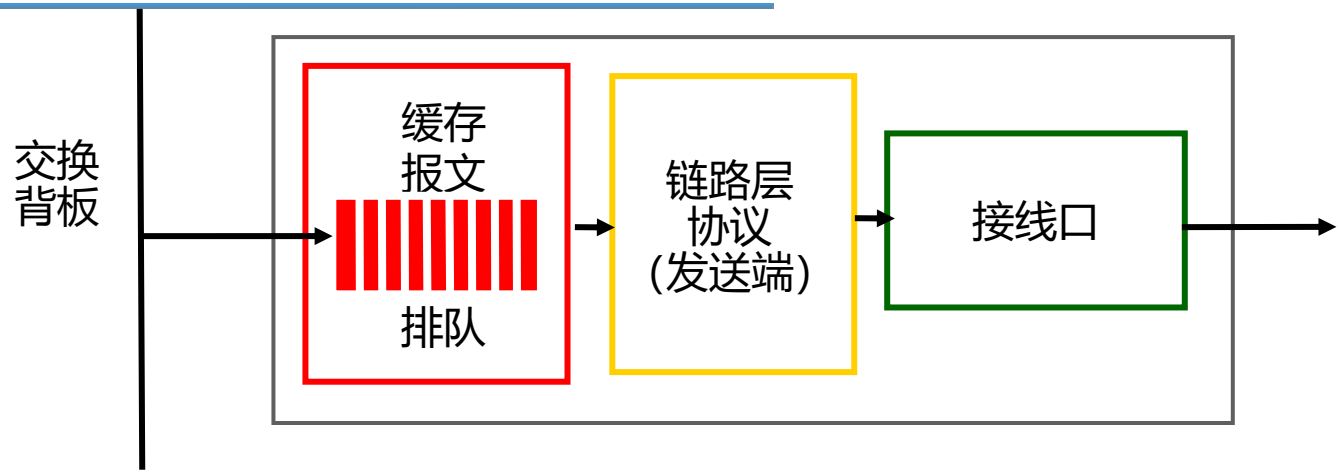
目的地址: 11001000 00010111 00011000 10101010 哪个端口输出?

交换背板

- 将数据包从输入端口队列转移到输出端口队列
- 交换速率：每秒转移的数据包数量
  - 通常用端口线速的倍数标识
  - N个数据端口：理想情况，N倍线速的交换速率
- 三类交换背板



输出端口



- 当从背板到达的数据包
输速率，需要缓存数据
- 调度策略，从队列中选取特定数据包传输

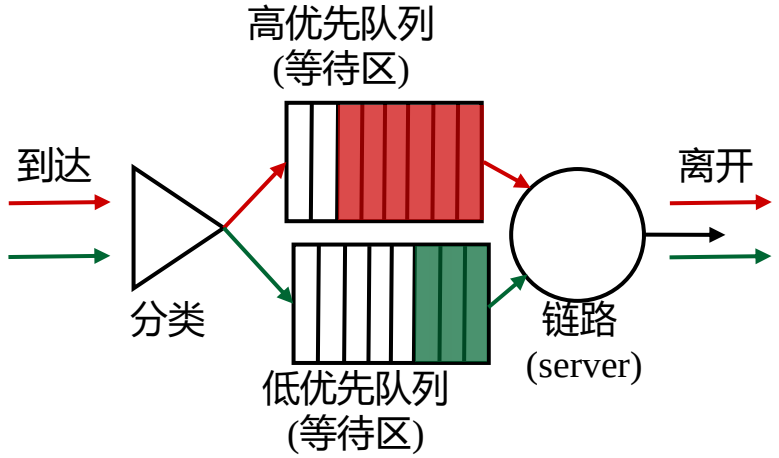
可能因为拥塞
没有缓存空间而丢包

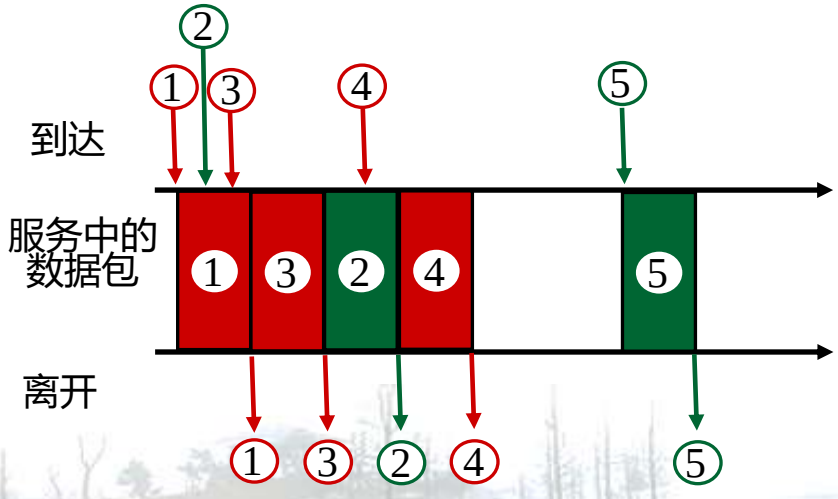
优先调度—决定哪些数据包优先获得服务

调度策略：优先调度

优先调度: 传输在最高优先级队列中排队的数据包

- 数据包分类，对应多个优先级
  - 基于数据包携带的标识或者头部字段，例如，源/目的地址、端口号等等
  - 类似航班登机

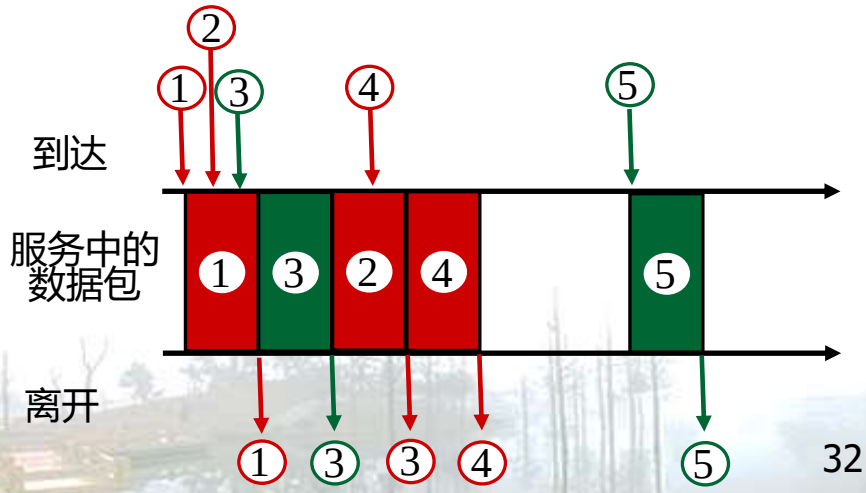




更多调度策略

轮询（round-robin）调度：将数据包分为多个类，对应多个队列

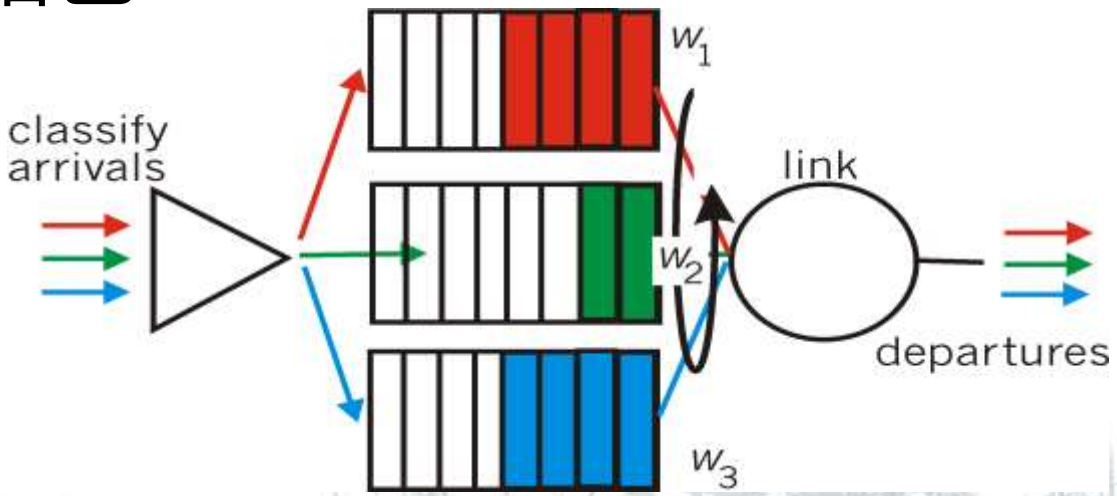
- 循环检查每个队列，如队列非空，传输一个数据包



更多调度策略

加权公平排队（WFQ）

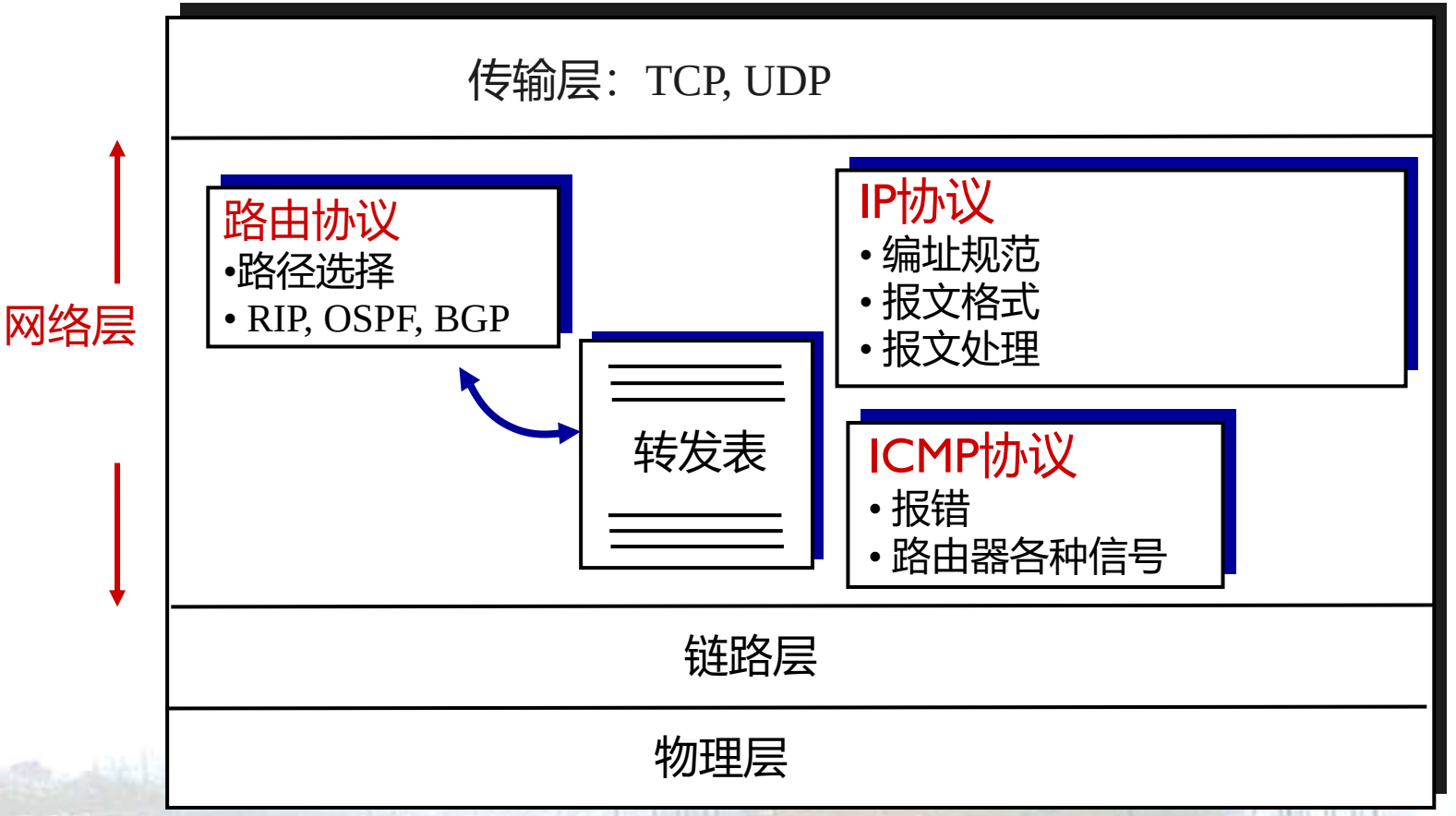
- 广义的轮询
- 每次检查队列时，按权重比例传输队列中的数据包



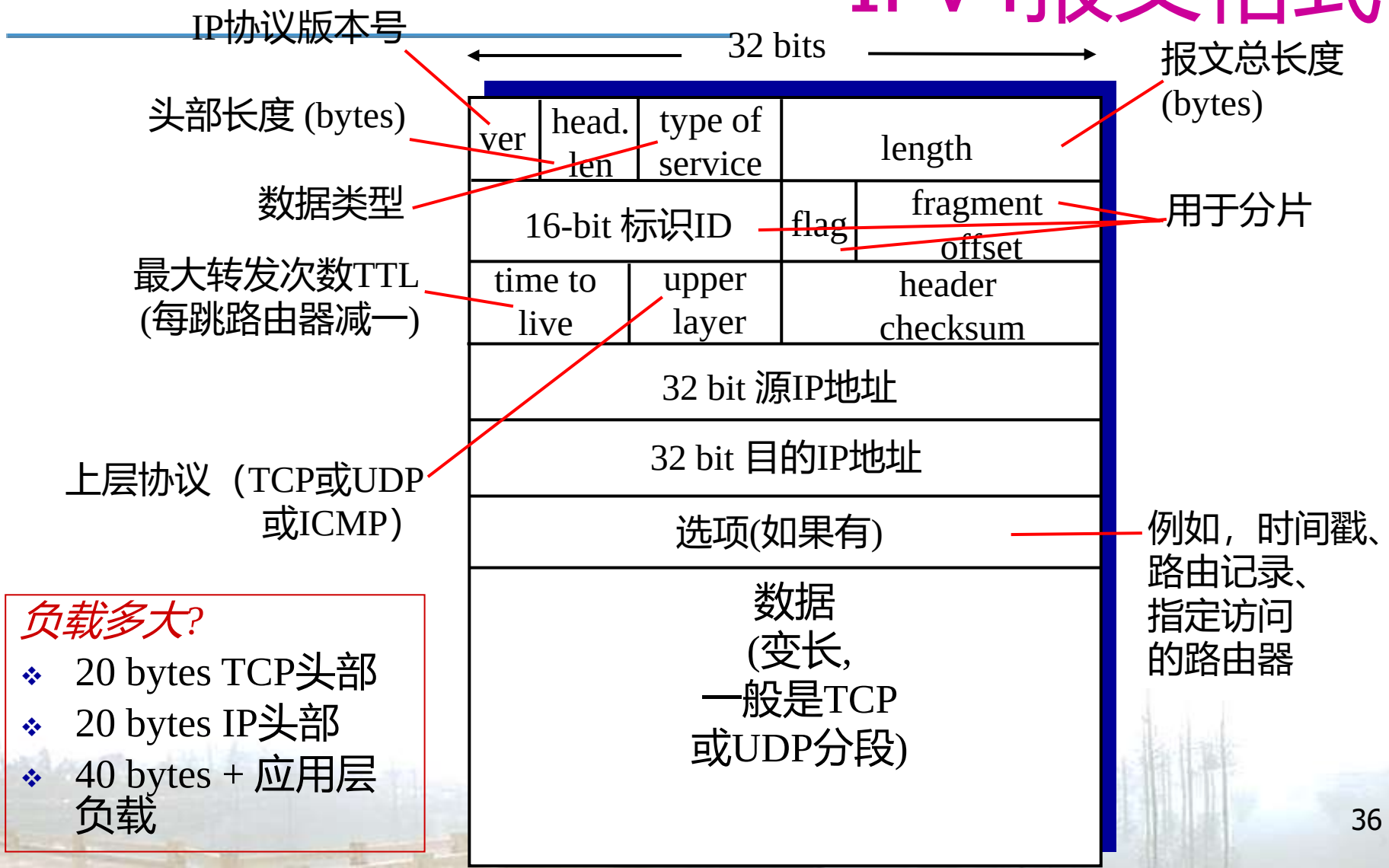
目录

因特网的网络层

主机和路由器承担网络层功能：



IPv4报文格式



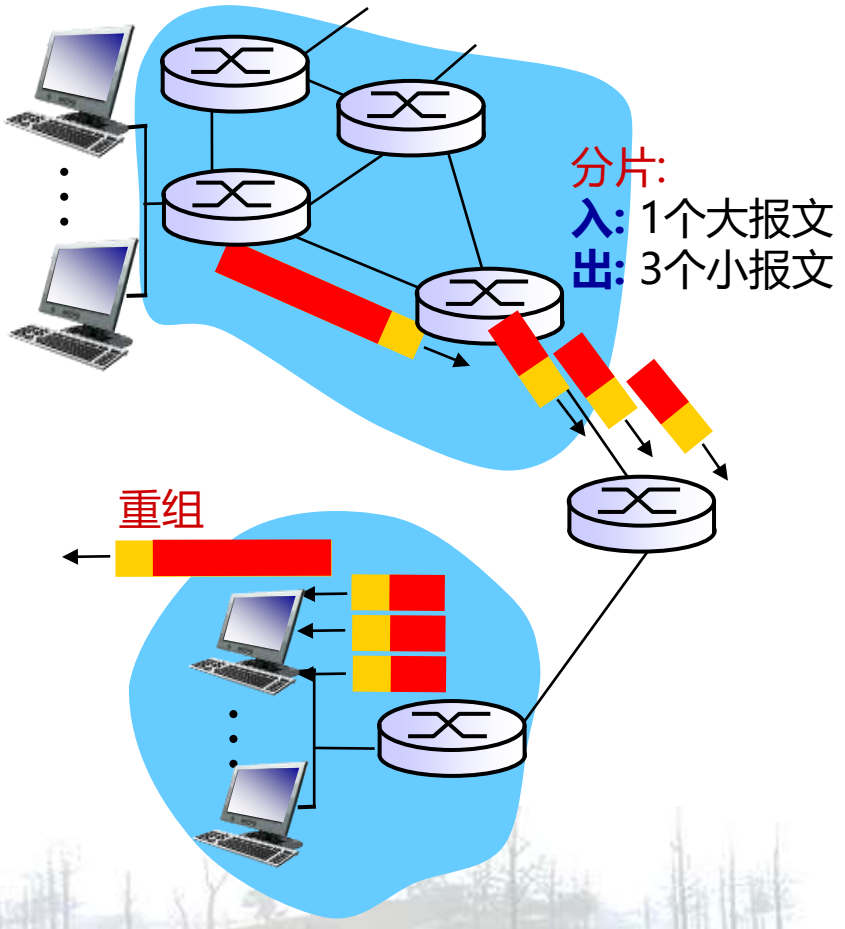
负载多大?

- 20 bytes TCP头部
- 20 bytes IP头部
- 40 bytes + 应用层负载

IP分片和重组

最后一个分片flag设置为0，
其余分片的flag设置为1

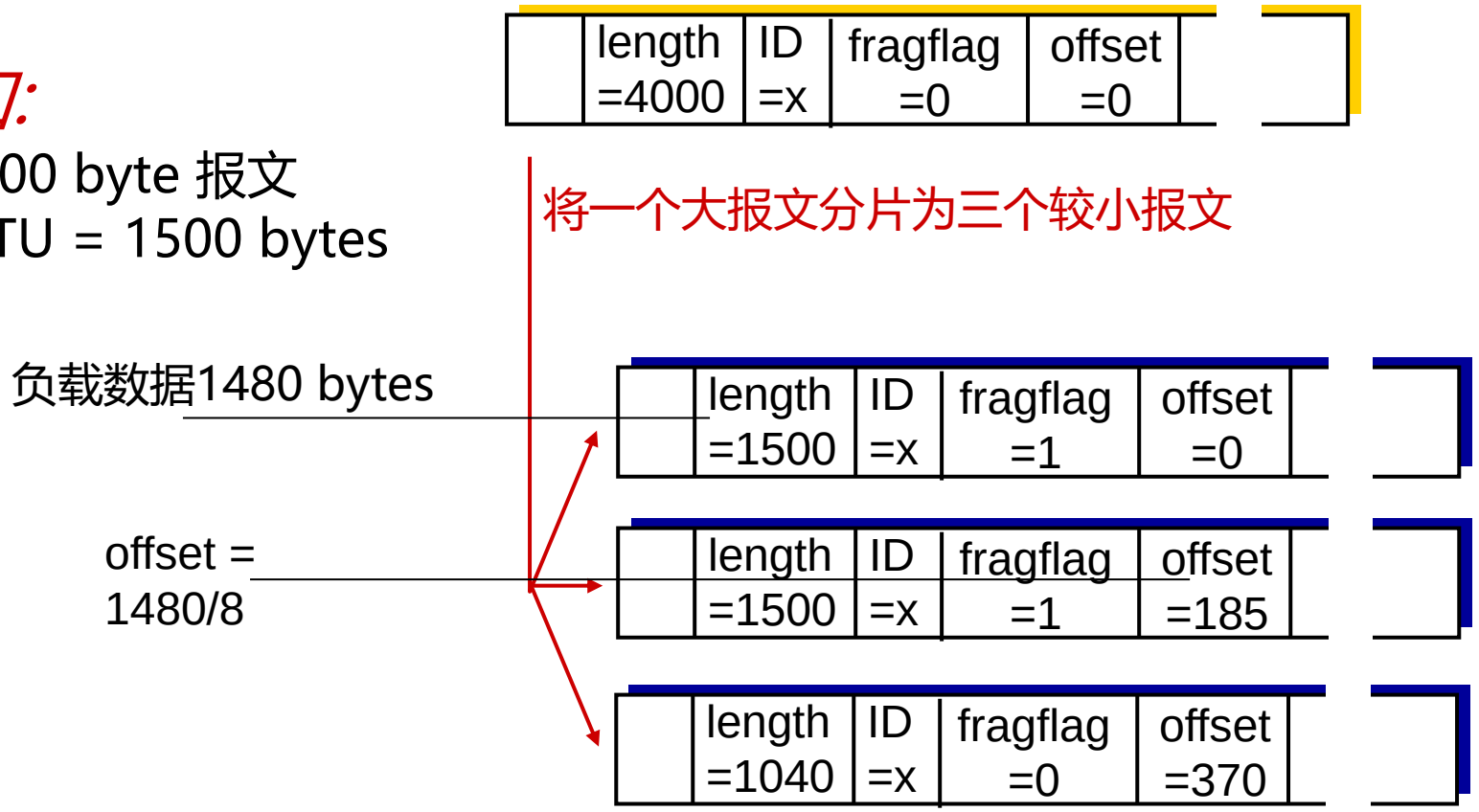
- 网络链路能够传输的最大帧大小被称为MTU
  - 不同的链路类型，不同的MTU
  - 以太网，1500 Bytes
- 当IP报文大小超过链路MTU，由路由器分片
  - 把一个报文分为多个报文
  - 在目的地重组
  - IP头部包含字段，携带分片重组所需信息



IP分片和重组

例如:

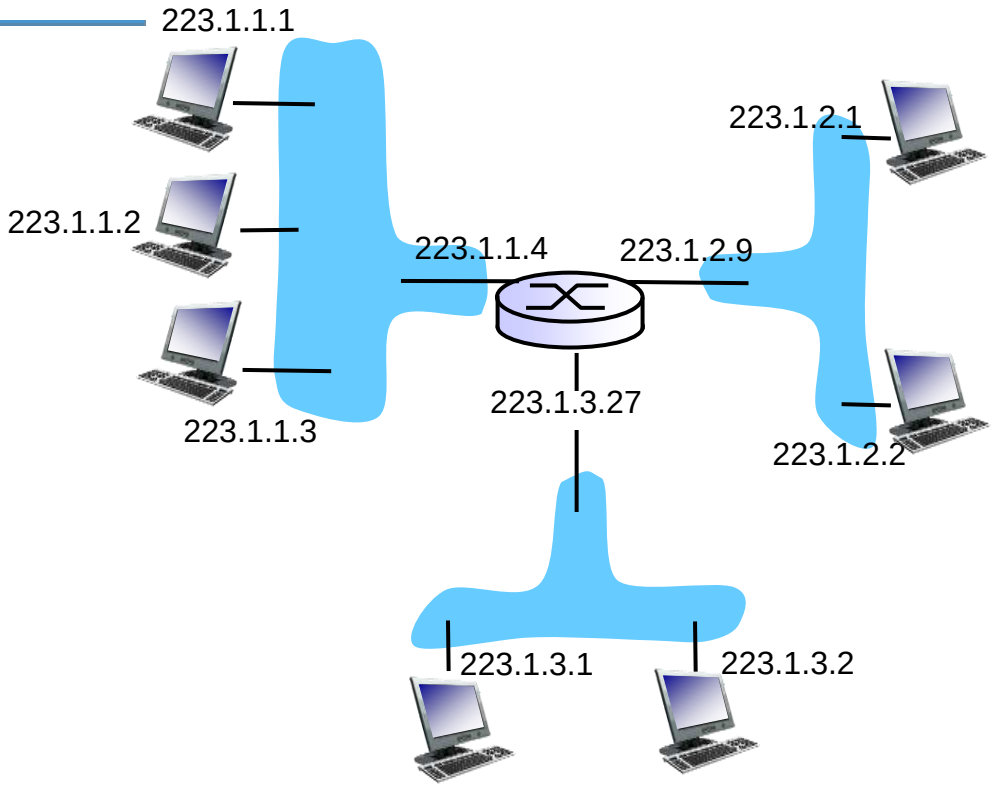
- 4000 byte 报文
- MTU = 1500 bytes



目录

IP地址

- IP地址: 32-bit，用于标识主机和路由器的网络接口
- 网络接口 (interface)：连接主机/路由器和物理传输介质
  - 路由器通常有多个网络接口
  - 主机通常有一个或两个网络接口(例如，有线以太网和无线802.11)
- 每个网络接口都关联一个IP地址



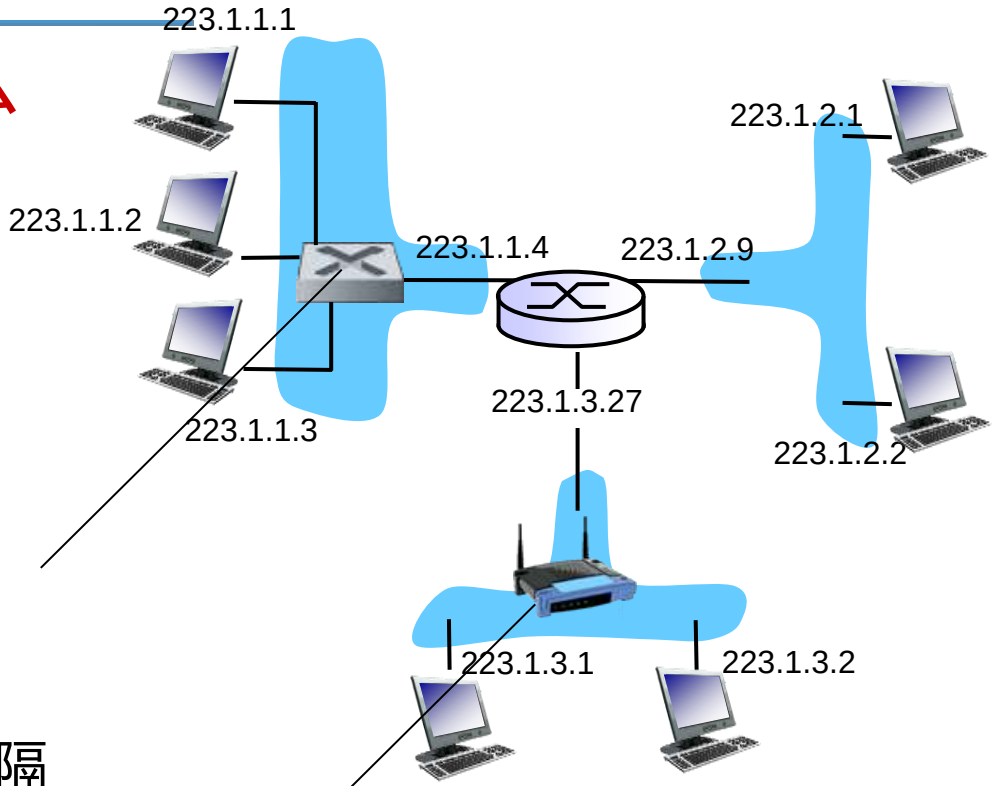
223.1.1.1 = 11011111 00000001 00000001 00000001

223 1 1 1

IP地址

问: 网络接口之间怎么连接?

答：链路层的内容



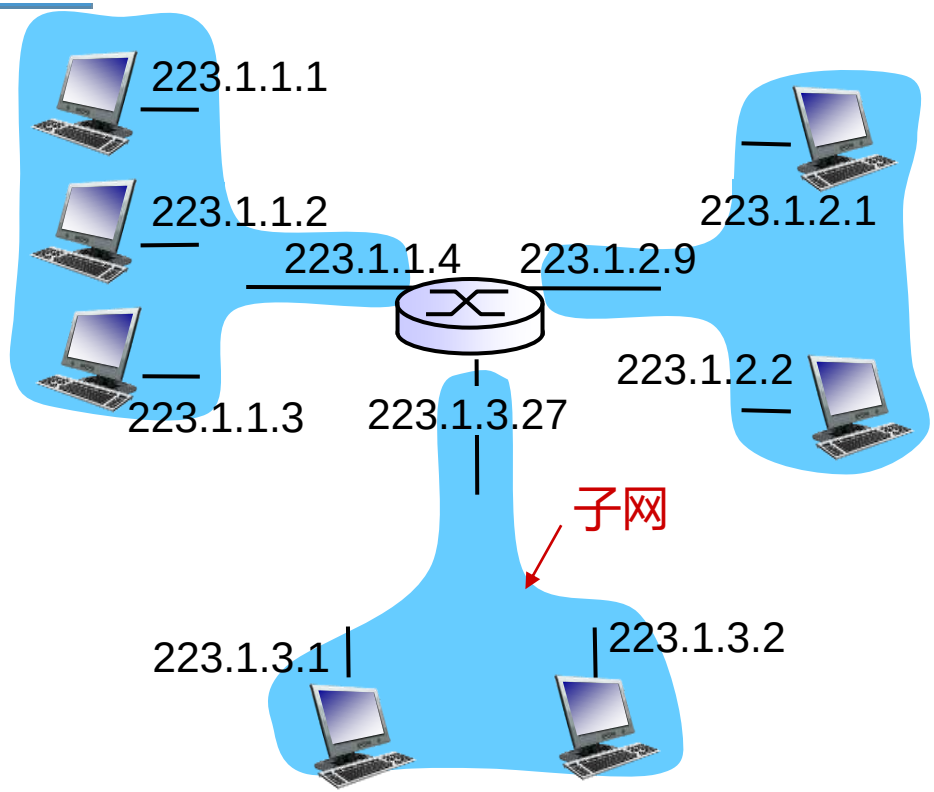
答：有线以太网接口由以太网交换机连接

在网络层：默认没有路由器分隔的网络上，网络接口彼此连接

答：无线WiFi接口由WiFi接入点互连

子网

- IP地址:
  - 子网部分- 高位bit
  - 主机部分- 低位bit
- 什么是子网?
  - 子网里设备接口的IP地址具有相同的子网部分（逻辑意义）
  - 子网中接口之间可以不经过路由器互通连接（物理意义）

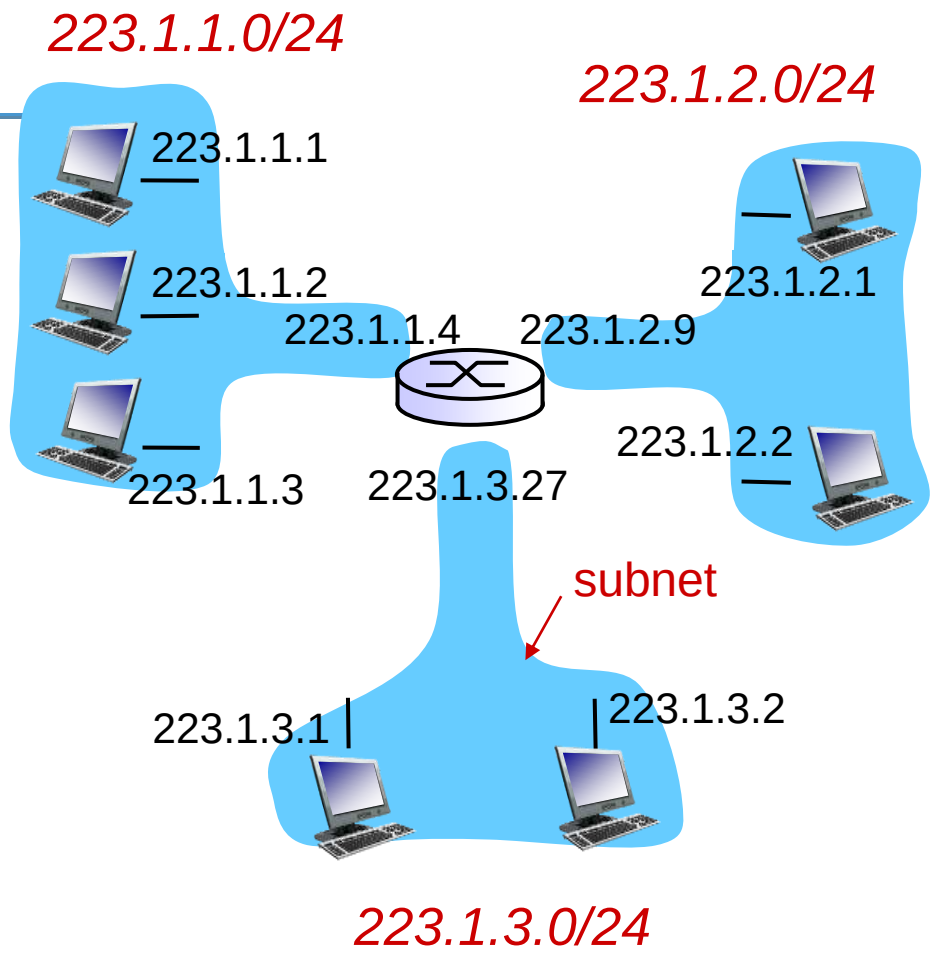


网络包含3个子网

子网

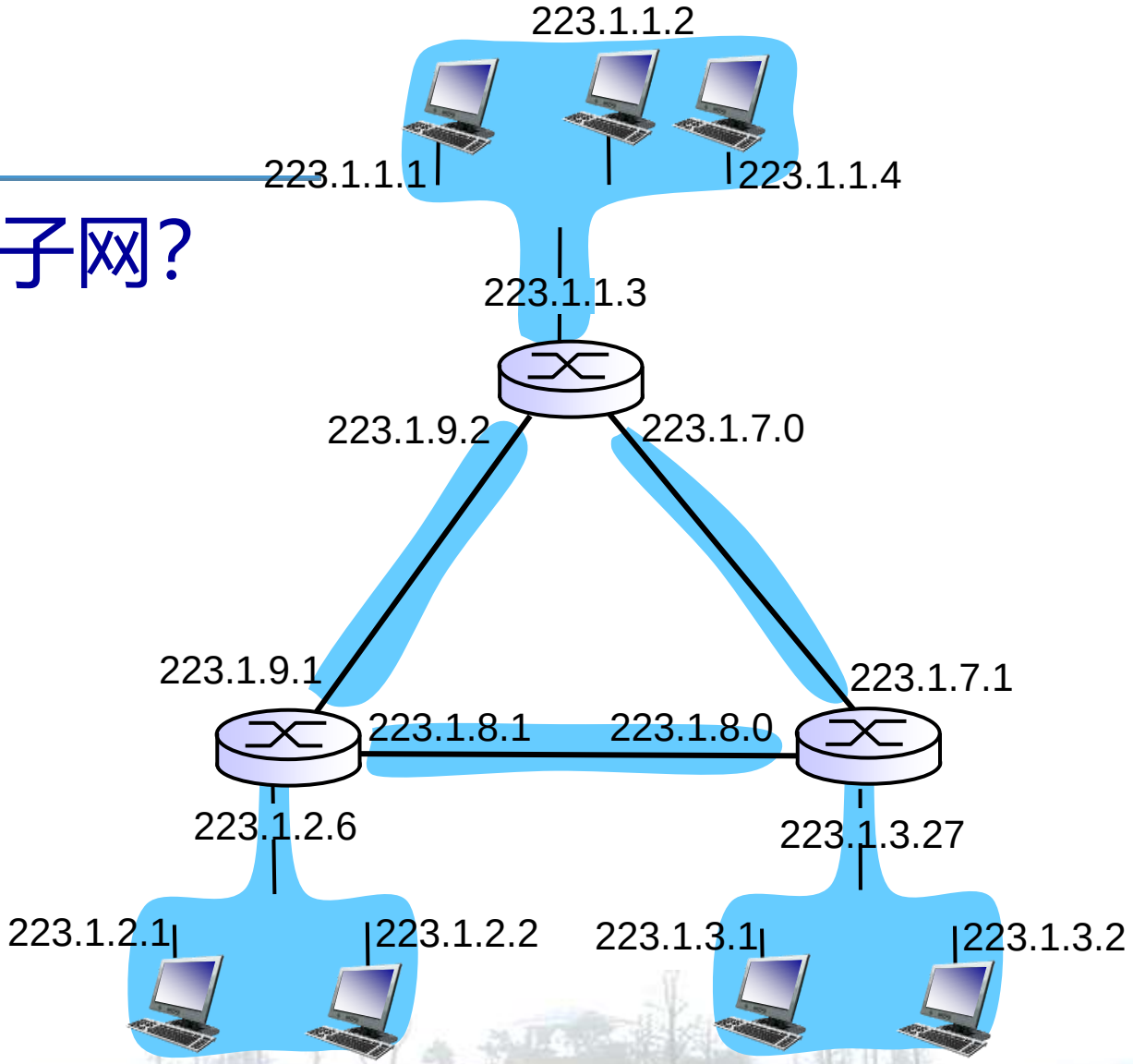
判断子网方法

- 将网络接口从主机或者路由器上剥离,形成一个或多个网络
- 这样形成的网络被称为子网



子网

这里有多少个子网?


223.1.1.2
223.1.1.1
223.1.1.4
223.1.1.3
223.1.9.2
223.1.7.0
223.1.9.1
223.1.7.1
223.1.8.1
223.1.8.0
223.1.2.6
223.1.3.27
223.1.2.1
223.1.2.2
223.1.3.1
223.1.3.2

IP地址：CIDR

CIDR: Classless InterDomain Routing（无类别域间路由）（不是路由协议，是一种IP地址空间划分方式）

- 地址的子网部分可以是任意长度
- CIDR格式: a.b.c.d/x, 其中x是子网部分的比特个数

11001000 00010111 00010000 00000000

200.23.16.0/23

IP地址：CIDR

- 低位bit部分可能包含（也可能不包含）更多的子网结构
  - 200.23.16.0/23 可能包含两个更小的子网: 200.23.16.0/24 和 200.23.17.0/24.
- 在CIDR之前，使用分类地址的方案,
  - 地址中8-、16-、和 24-bit 子网部分的网络被称为A、B、和C类网络。
- 广播地址：255.255.255.255
  - 以该地址作为目的地址，将报文送到子网中每个主机（不能越过路由器）

几个概念

- 子网的网络地址：子网部分不变，主机部分全0
  - 例如：子网使用CIDR地址块10.64.0.0/11，10.64.0.0是该子网的网络地址
- 子网掩码：网络部分全1，主机部分全0
  - 子网中IP地址 AND 子网掩码 =子网的网络地址，
  - 子网部分比特数被称为掩码长度
  - 例如：子网中IP地址是10.80.0.1，子网掩码255.224.0.0，掩码长度为11，子网的网络地址10.64.0.0
    - IP地址：00001010 01010000 00000000 00000001 (10.80.0.1)
    - 子网掩码：11111111 11100000 00000000 00000000 (255.224.0.0)
    - 子网网络地址：00001010 01000000 00000000 00000000 (10.64.0.0)

几个概念

- 广播地址：网络部分不变，主机部分全1
  - 目的地为广播地址的报文能被子网中所有接口接收
  - 例如：10.95.255.255 （00001010 01011111 11111111 11111111）
- 最小用户地址（子网内可以分配给主机的最小地址）：网络部分不变，主机部分最小非全0
  - 例如：10.64.0.1 （01011111 01000000 00000000 00000001）
- 最大用户地址（子网内可以分配给主机的最大地址）：网络部分不变，主机部分最大非全1
  - 例如： 10.95.255.254 （01011111 01011111 11111111 11111110）

在转发表中使用CIDR

- 怎样用CIDR表示?

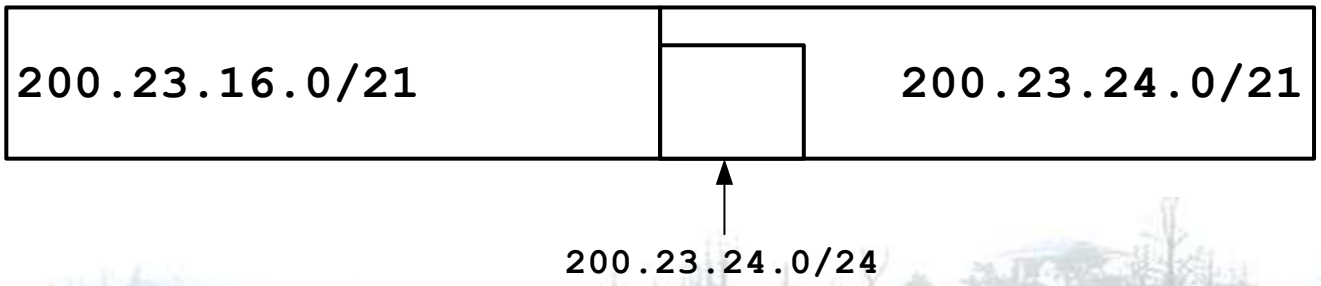
目的地址范围	链路端口号
11001000 00010111 00010000 00000000 到 11001000 00010111 00010111 11111111	0
11001000 00010111 00011000 00000000 到 11001000 00010111 00011000 11111111	1
11001000 00010111 00011001 00000000 到 11001000 00010111 00011111 11111111	2
其它	3

在转发表中使用CIDR

- 11001000 00010111 00010*** ********表示为200.23.16.0/21
- 11001000 00010111 00011000 ********表示为200.23.24.0/24
- 11001000 00010111 00011001 00000000 到11001000 00010111 00011111 1111111怎么表示?
  - 从11001000 00010111 00011*** ********地址块拿掉11001000 00010111 00011000 ********地址块
  - 11001000 00010111 00011*** ********表示为200.23.24.0/21

在转发表中使用CIDR

CIDR	链路端口号
200.23.16.0/21	0
200.23.24.0/24	1
200.23.24.0/21	2
其它	3



主机如何获得IP地址

问：主机如何获得IP地址?

- 由管理员硬编码到操作系统中
  - Windows: 控制面板→网络→配置→TCP/IP →属性
  - UNIX: /etc/rc.config
- DHCP（动态主机配置协议）：从服务器动态地获得地址
  - 按需使用，即插即用

DHCP

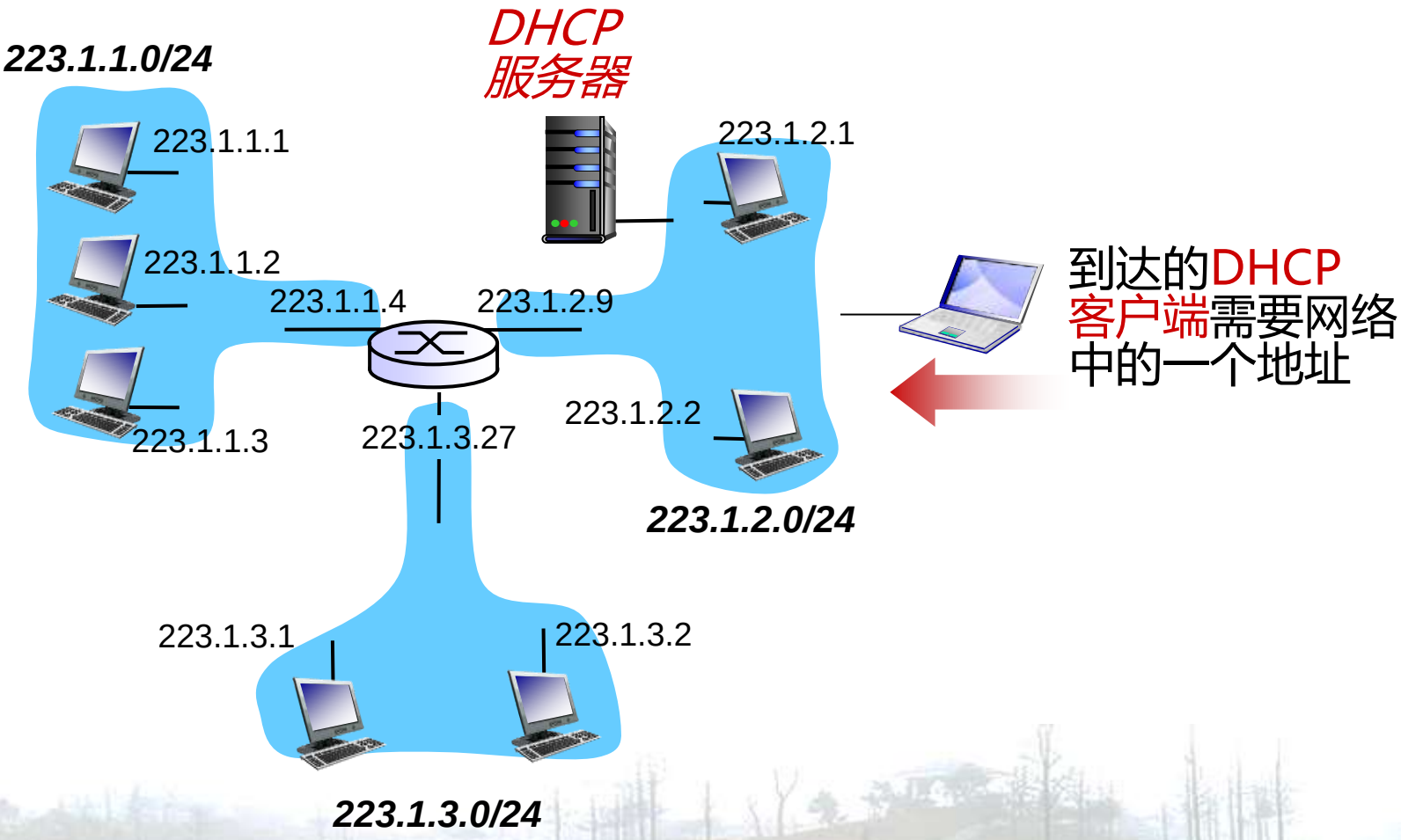
目标: 当主机加入网络时，从网络服务器动态地获得IP地址。

- 从网络中“租”一个地址，可以续租当前使用中的地址。
- 地址可以重用 （主机下线，地址被分配给其它新上线的主机）
- 移动用户加入网络，获得地址

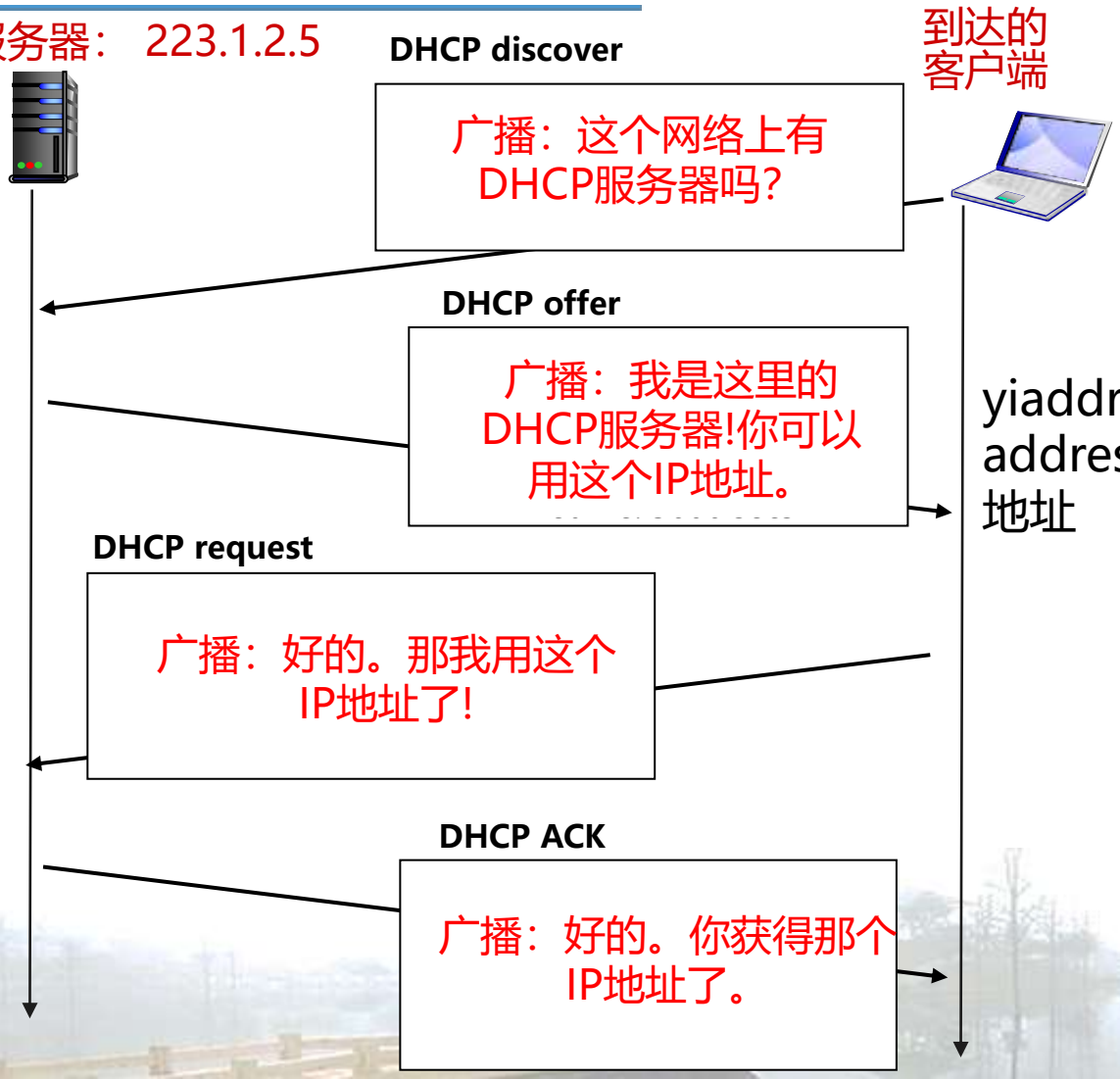
DHCP 概览:

- 主机广播“DHCP discover”消息 [可选]
- DHCP 服务器回应“DHCP offer”消息 [可选]
- 主机请求IP地址：“DHCP request”消息
- DHCP 服务器发送地址：“DHCP ack”消息

DHCP客户端-服务器场景



DHCP客户端-服务器场景



DHCP：不仅仅是分配IP地址

除了IP地址，DHCP可以向主机提供子网中如下信息：

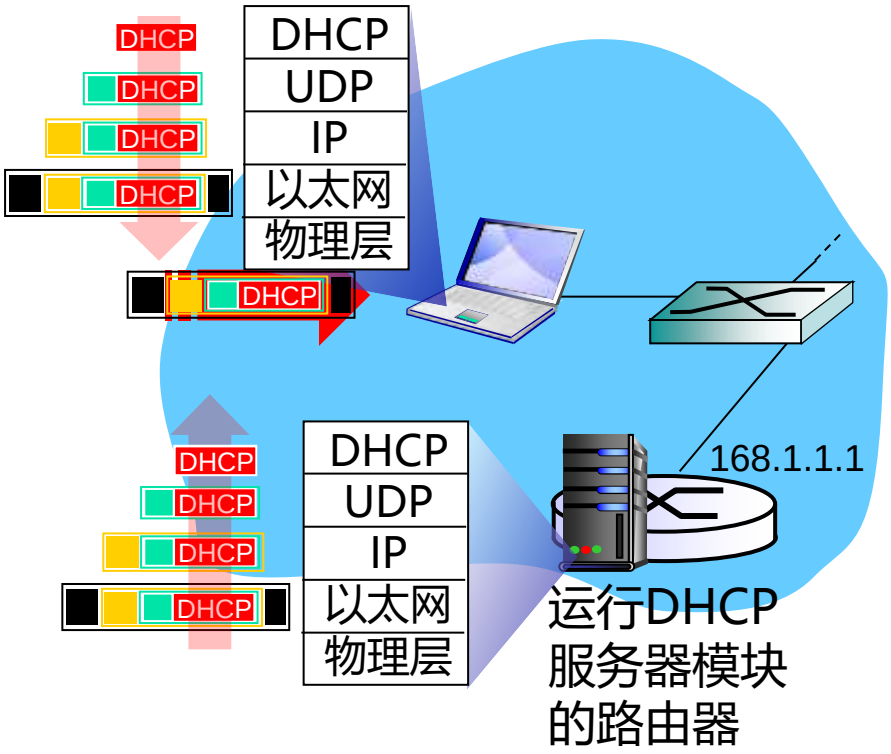
- 子网第一跳路由器的地址（网关地址）
- DNS服务器的名字和地址
- 子网掩码 (地址中子网部分和主机部分的长度)

子网中没有DHCP服务器时，可使用DHCP中继代理转发相关的消息。

移动网络相关

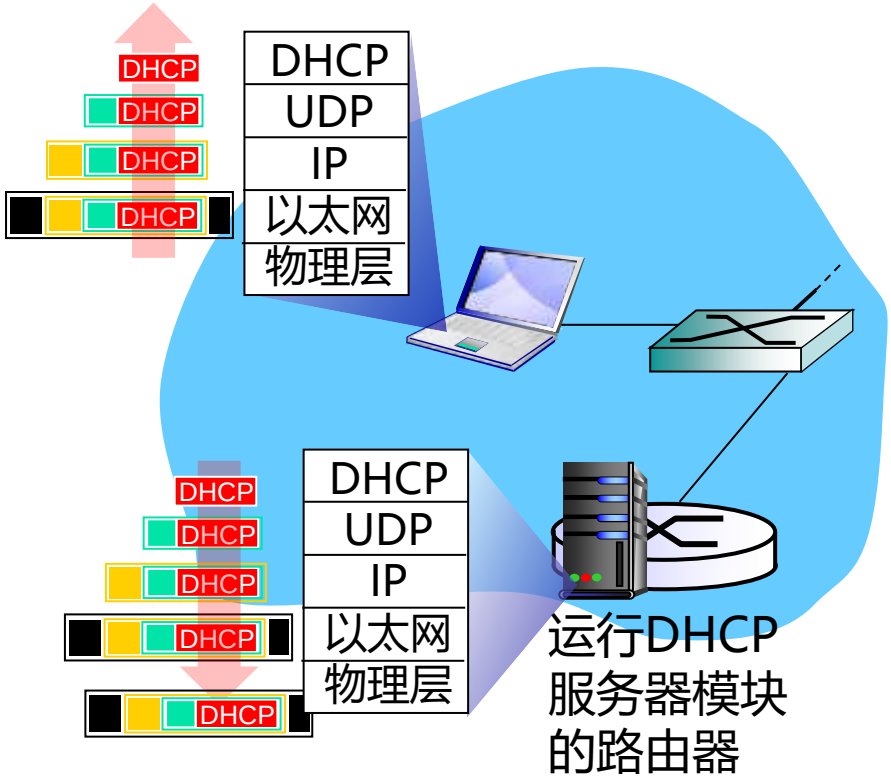
- 移动到新的子网获得新地址
- 无法维持现有的TCP连接（TCP socket四元组）
- 通过移动IP解决

DHCP举例



- 主机插上网线后，需要IP地址、第一跳路由器的地址、以及DNS的地址。使用DHCP
- DHCP 请求在UDP分段中封装，在IP报文中封装，在802.3以太网帧中封装
- 在本地局域网发送以太网广播帧 (目的MAC地址: FFFFFFFFFFFF)，广播帧被运行DHCP服务器的路由器接收
- 以太网-IP-UDP解封装得到DHCP 请求

DHCP举例



- DHCP服务器构造DHCP ACK消息，包括客户端的IP地址、第一跳路由器的IP地址、DNS服务器的名字和地址
- 封装、发送，客户端接收，解封装获得DHCP消息
- 客户端获得了IP地址，并且知道了第一跳路由器的IP地址、DNS服务器的名字和地址

DHCP举例

On Windows
C:> ipconfig /release
C:> ipconfig /renew

No.	Time	Source	Destination	Protocol	Length	Info
59	8.411174	192.168.1.100	192.168.1.1	DHCP	342	DHCP Release
87	11.872892	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover
95	12.369850	192.168.1.1	192.168.1.100	DHCP	590	DHCP Offer
96	12.370265	0.0.0.0	255.255.255.255	DHCP	362	DHCP Request
97	12.373023	192.168.1.1	192.168.1.100	DHCP	590	DHCP ACK

```
Your (client) IP address: 192.168.1.100
Next server IP address: 0.0.0.0
Relay agent IP address: 0.0.0.0
Client MAC address: IntelCor_80:f4:34 (8c:70:5a:80:f4:34)
Client hardware address padding: 00000000000000000000
⊞ Server name option overloaded by DHCP
⊞ Boot file name option overloaded by DHCP
Magic cookie: DHCP
⊞ Option: (53) DHCP Message Type (Offer)
⊞ Option: (54) DHCP Server Identifier
⊟ Option: (1) Subnet Mask
    Length: 4
    Subnet Mask: 255.255.255.0
⊟ Option: (51) IP Address Lease Time
    Length: 4
    IP Address Lease Time: (7200s) 2 hours
```

机构如何获得IP地址

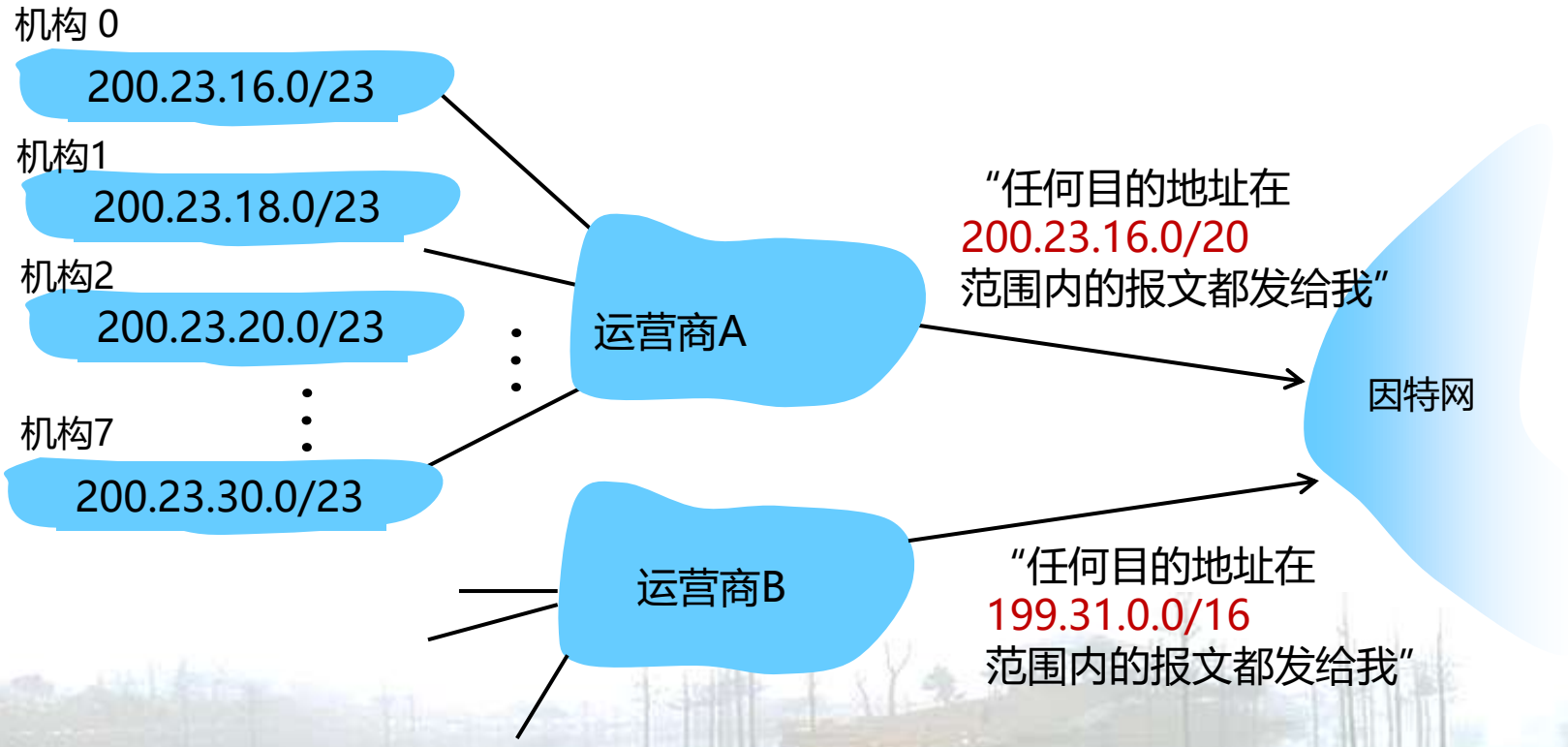
问：如何获得IP地址的子网部分？

答：机构从ISP的地址空间中分配

ISP的地址块	11001000 00010111 00010000	00000000	200.23.16.0/20
机构 0	11001000 00010111 00010000	00000000	200.23.16.0/23
机构 1	11001000 00010111 00010010	00000000	200.23.18.0/23
机构 2	11001000 00010111 00010100	00000000	200.23.20.0/23
…	……	….	….
机构 7	11001000 00010111 00011110	00000000	200.23.30.0/23

地址分层：路由聚合

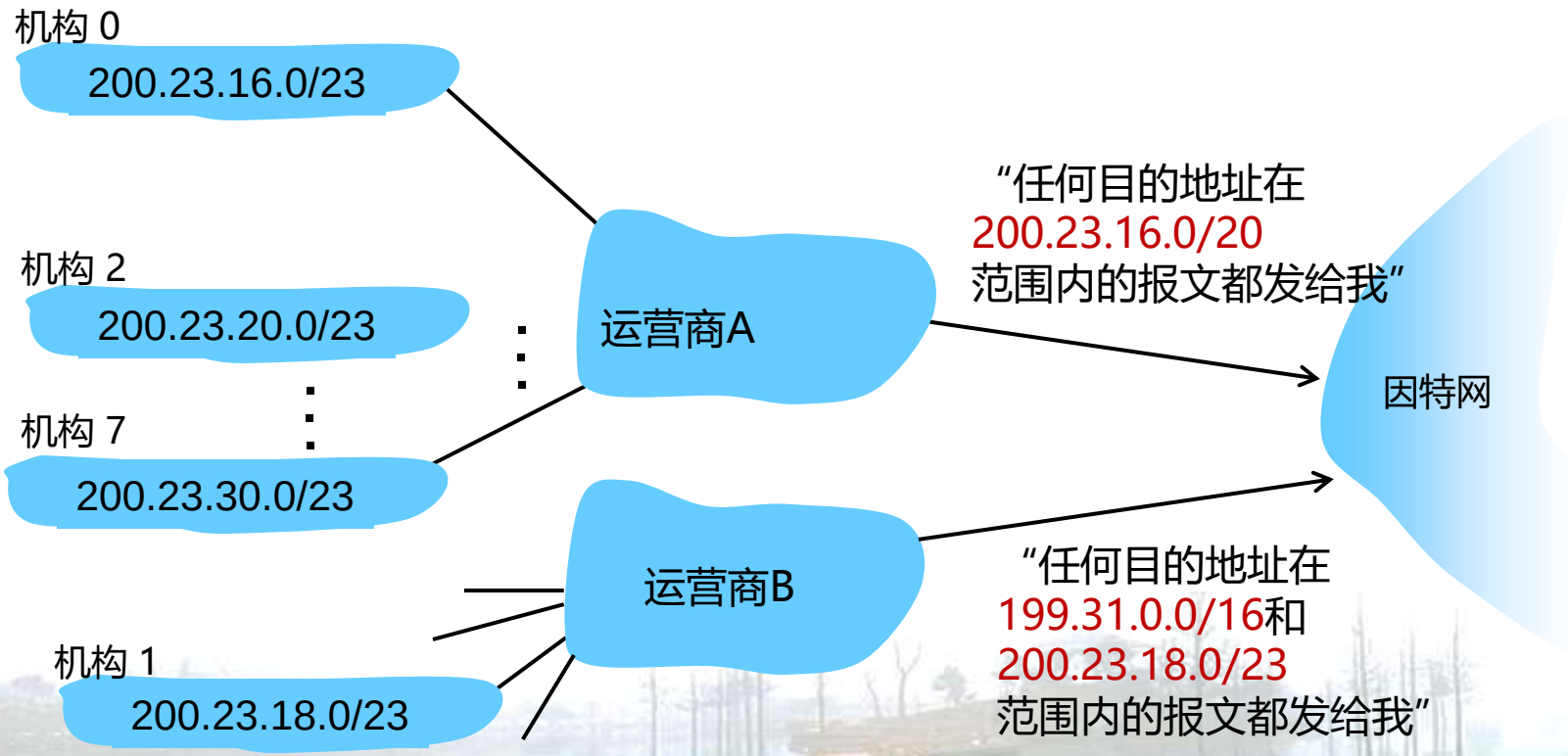
层次化的地址有利于高效地发布路由信息：



地址分层：路由变更

机构1改变其运营商（但不希望改变IP地址），
通过运营商B连到因特网

由于最长前缀匹配，发给机构1的流量通过运营商B送到机构1



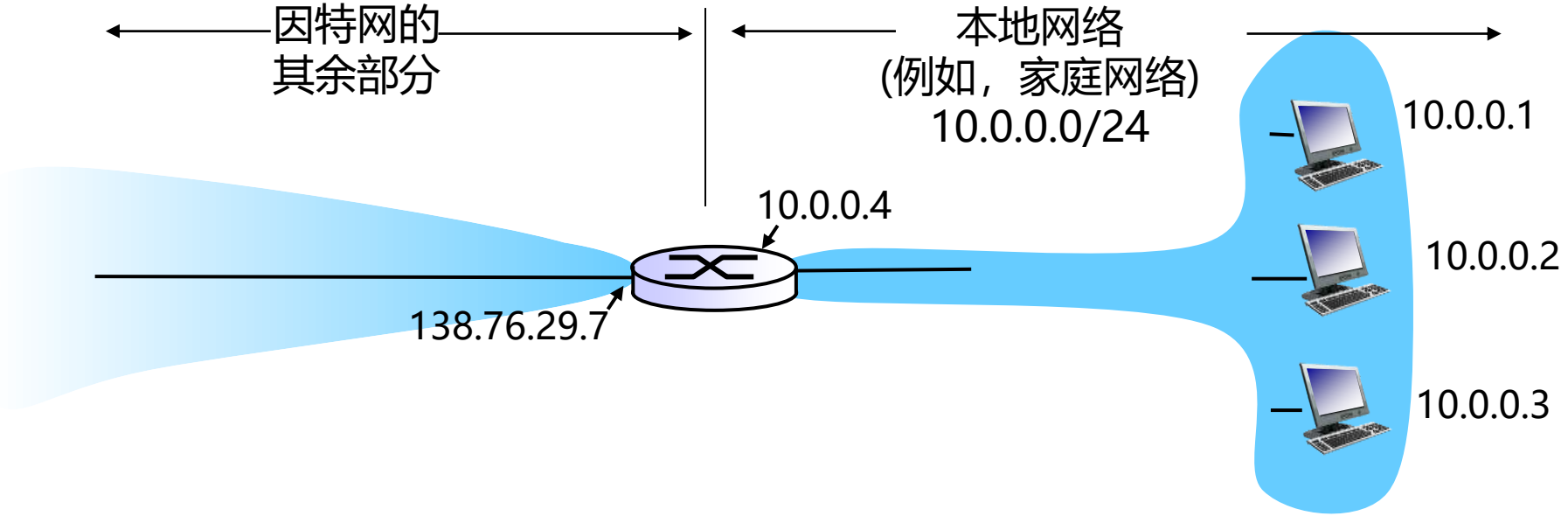
IP地址

问：运营商如何获得地址块?

答：ICANN: Internet Corporation for Assigned Names and Numbers
http://www.icann.org/

- 分配地址
- 管理DNS
- 分配域名、处理纷争

NAT：网络地址翻译（network address translation）



所有离开本地网络的报文源地址都相同，138.76.29.7，但是它们的源端口号不同

所有在本地网络上传输的报文，源地址和目的地址都在10.0.0.0/24范围内

NAT：网络地址翻译

动机：本地网络对外只使用一个IP地址（公网地址）：

- 无需从运营商获得包含多个连续地址的地址段，只需要获得一个地址，本地网络上所有设备都可接入因特网
- 本地网络上改变设备的IP地址，无须对外通知
- 改换ISP时无需改变设备上的IP地址
- 本地网络内部的设备无法从外面寻址和访问(更安全)

NAT：网络地址翻译

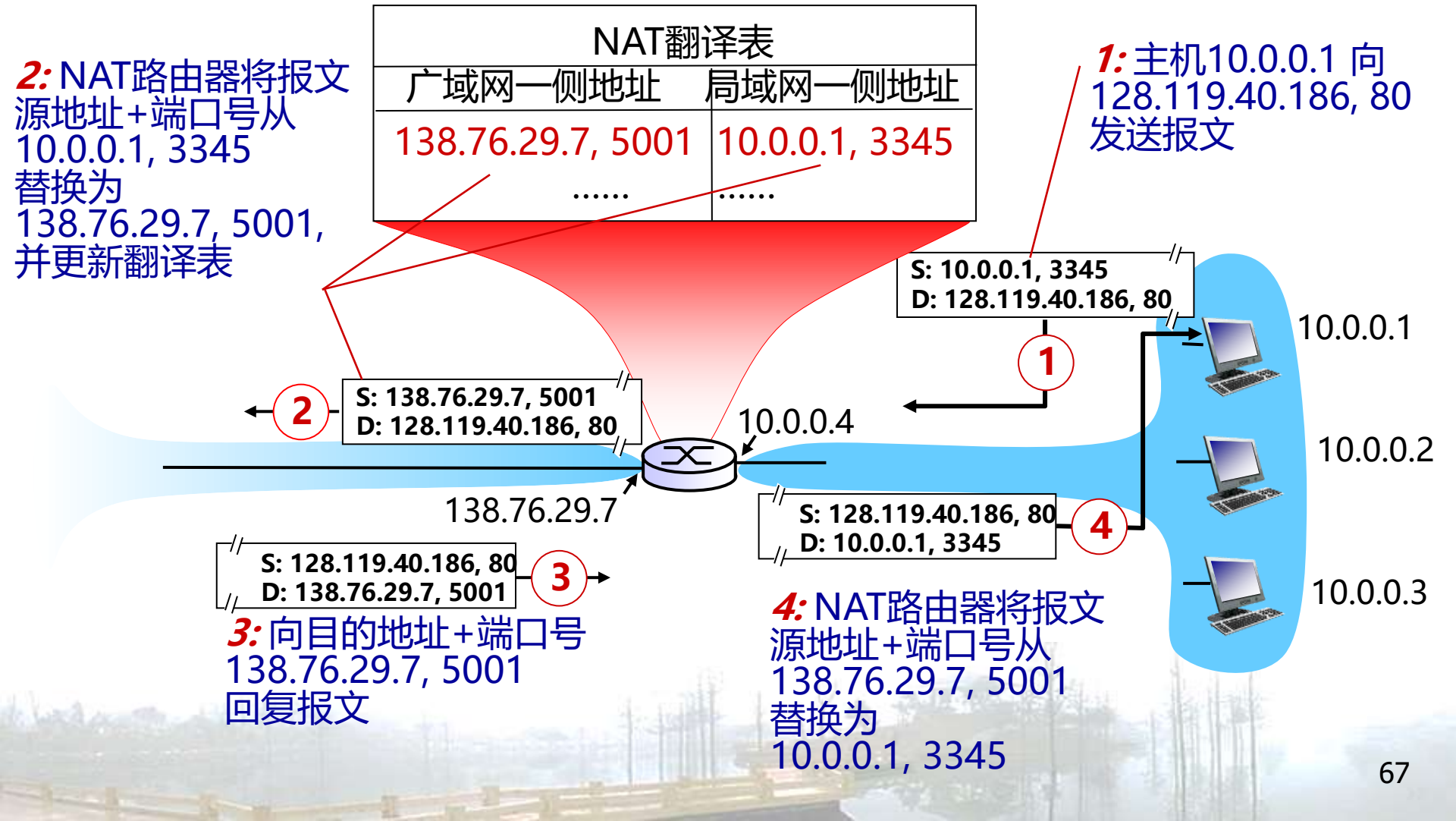
实现: NAT路由器必须:

- 向外的报文：替换每个报文的 (源IP地址, 端口号) 为(NAT路由器的IP地址, 新端口号)

  . . . 因特网上的远程客户端/服务器以(NAT路由器的IP地址，新端口号)作为目的地回复报文

- 在(NAT翻译表)中记住每个(源IP地址, 端口号)和(NAT路由器的IP地址, 新端口号)的映射关系
- 向内的报文: 查询NAT翻译表，替换每个报文的目的地址和端口号字段的 (NAT路由器的IP地址, 新端口号)为(源IP地址, 端口号)

NAT：网络地址翻译

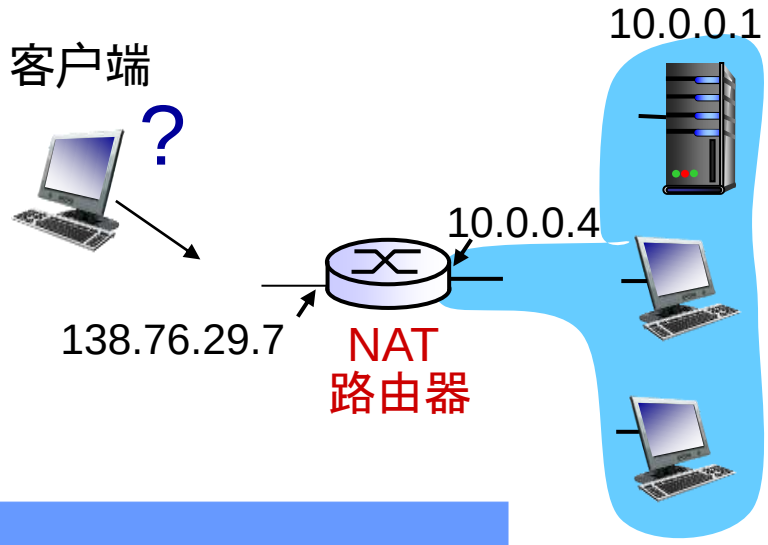

NAT翻译表
广域网一侧地址 局域网一侧地址
138.76.29.7, 5001 10.0.0.1, 3345
....
2: NAT路由器将报文源地址+端口号从10.0.0.1, 3345替换为138.76.29.7, 5001, 并更新翻译表
1: 主机10.0.0.1 向128.119.40.186, 80发送报文
S: 10.0.0.1, 3345
D: 128.119.40.186, 80
1
2
S: 138.76.29.7, 5001
D: 128.119.40.186, 80
10.0.0.4
138.76.29.7
10.0.0.1
10.0.0.2
10.0.0.3
S: 128.119.40.186, 80
D: 10.0.0.1, 3345
4
S: 128.119.40.186, 80
D: 138.76.29.7, 5001
3
3: 向目的地址+端口号138.76.29.7, 5001回复报文
4: NAT路由器将报文源地址+端口号从138.76.29.7, 5001替换为10.0.0.1, 3345

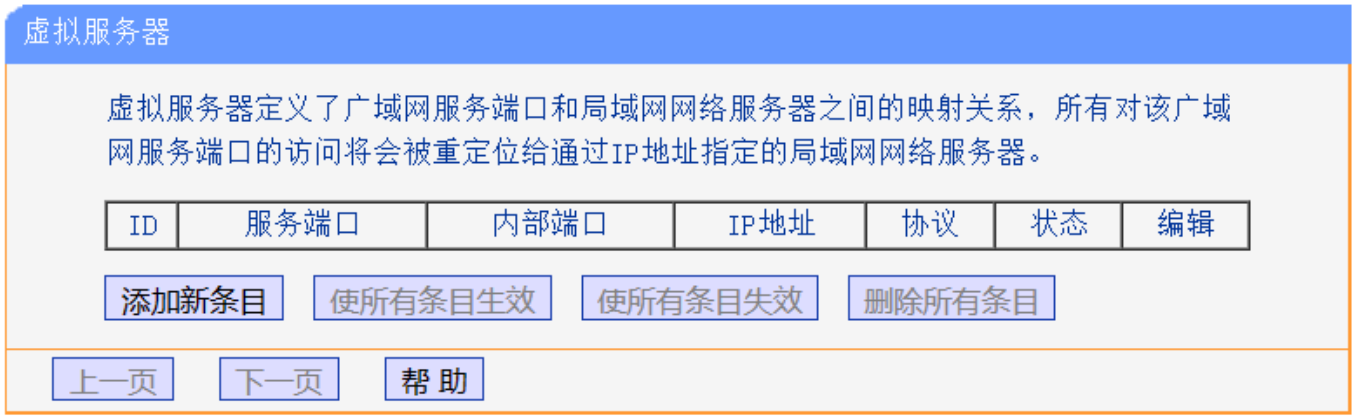
NAT：网络地址翻译

- 16-bit端口号字段:
  - 支持本地局域网一侧的超过60,000并发连接
- NAT有争议:
  - 路由器应该仅具备1-3层的报文处理功能
  - 地址短缺应该由IPv6解决
  - 破坏了传输层端到端的设定
    - 设计实现网络应用时必须考虑NAT的存在，例如，P2P应用
  - 穿越NAT:如果客户端希望连接NAT背后的服务器，该怎么办?

NAT穿越

- 客户端希望连接地址为10.0.0.1的服务器
  - 无法使用10.0.01.作为目的IP地址发出连接请求
  - 只有NAT路由器地址138.76.29.7路由可达
- *方案1:* 静态配置NAT表，将某端口





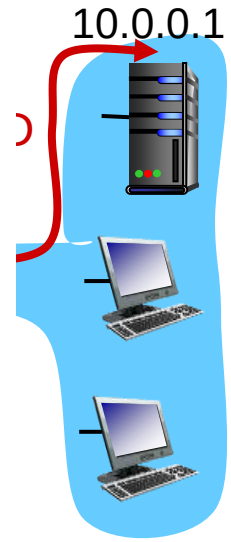
NAT穿越

- *方案2:*
  (UPnP)
  (IGD)
  - 主动
    (138
  - 请求
    端口
    立映

当前UPnP状态：开启 关闭

当前UPnP设置列表

ID	应用描述	外部端口	协议类型	内部端口	IP地址	状态
1	BJSDKat192.168.0.107:6955	6955	TCP	6955	192.168.0.107	已启用
2	BJSDKat192.168.0.107:6955	6955	UDP	6955	192.168.0.107	已启用
3	BJSDKat192.168.0.104:6936	6936	TCP	6936	192.168.0.104	已启用
4	BJSDKat192.168.0.104:6936	6936	UDP	6936	192.168.0.104	已启用
5	BJSDKat192.168.0.107:38875	38875	TCP	38875	192.168.0.107	已启用
6	BJSDKat192.168.0.107:34703	34703	TCP	34703	192.168.0.107	已启用
7	BJSDKat192.168.0.104:6933	6933	TCP	6933	192.168.0.104	已启用
8	BJSDKat192.168.0.104:6933	6933	UDP	6933	192.168.0.104	已启用
9	BJSDKat192.168.0.104:41911	41911	UDP	41911	192.168.0.104	已启用
10	BJSDKat192.168.0.104:41911	41911	TCP	41911	192.168.0.104	已启用
11	BJSDKat192.168.0.105:44727	44727	UDP	44727	192.168.0.105	已启用
12	BJSDKat192.168.0.105:44727	44727	TCP	44727	192.168.0.105	已启用
13	BJSDKat192.168.0.107:37089	37089	UDP	37089	192.168.0.107	已启用
14	BJSDKat192.168.0.107:37089	37089	TCP	37089	192.168.0.107	已启用
15	BJSDKat192.168.0.105:6972	6972	TCP	6972	192.168.0.105	已启用
16	BJSDKat192.168.0.105:6972	6972	UDP	6972	192.168.0.105	已启用
17	BJSDKat192.168.0.105:34095	34095	UDP	34095	192.168.0.105	已启用
18	BJSDKat192.168.0.105:34095	34095	TCP	34095	192.168.0.105	已启用



目录

IPv6：动机

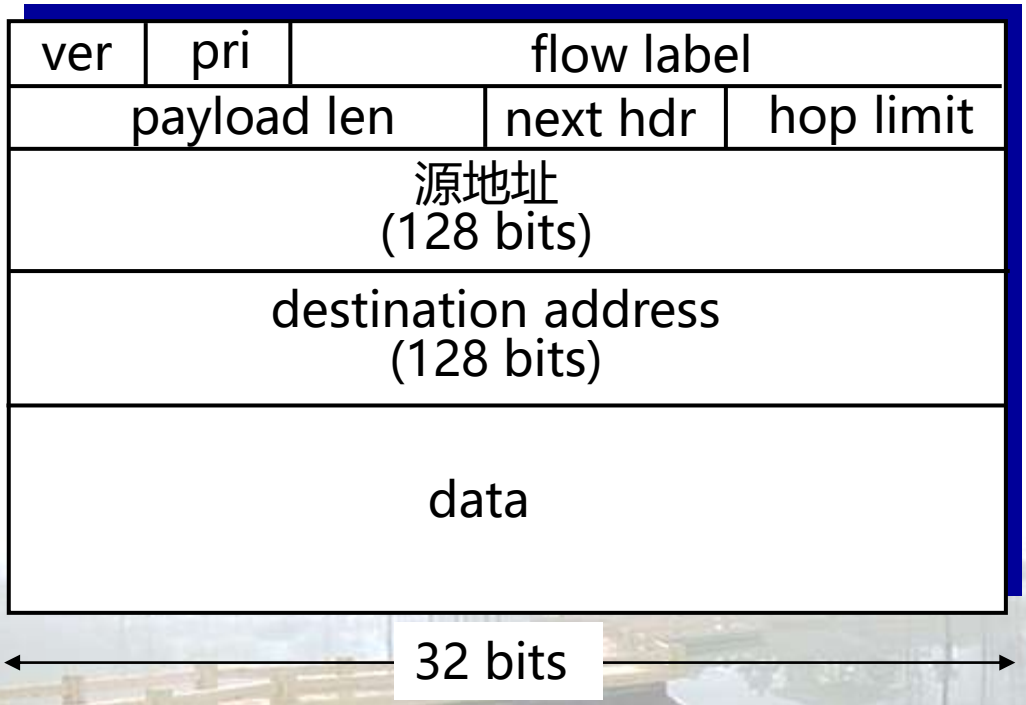
- 初始目的: 32-bit的地址空间即将被耗尽
- 更多目的:
  - 使用更便于路由器处理转发的报文头部格式
  - 在头部中添加支持服务质量差异的字段

IPv6 报文头部:

- 固定40byte头部长度
- 不允许分片

IPv6报文格式

优先级（priority）：数据包优先级
流标签（flow label）：标识属于同一个“流”的报文
下一层头部: 标识上层协议

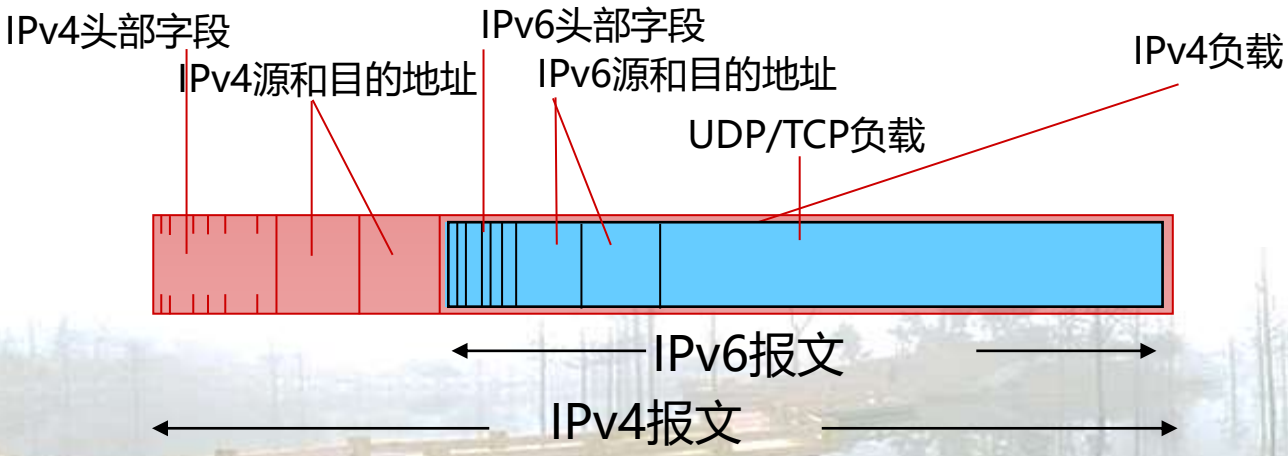


其它不同于IPv4之处

- 校验和: 移除，以节约路由器报文处理的时间
- 选项:仍然支持，但是不属于头部，由“下一层头部”字段指示
- ICMPv6：新版本的ICMP
  - 更多的消息类型, 例如 “数据包太大”
  - 多播的组管理功能

从IPv4到IPv6过渡

- 无法同时升级所有路由器
  - 怎样让IPv4和IPv6在网络中共存?
- 隧道: 把IPv6报文作为负载，由IPv4报文携带，被IPv4路由器处理转发

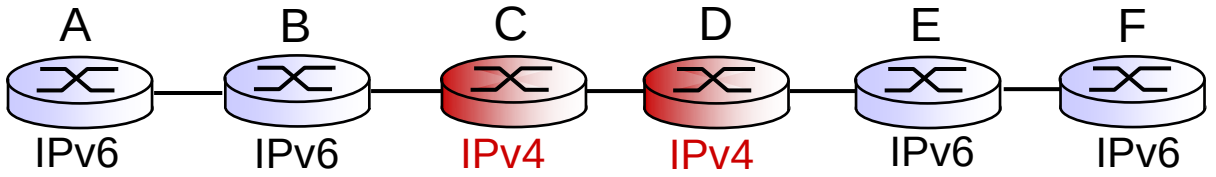


隧道

逻辑视图:


A
B
连接IPv6路由器的IPv4隧道
E
F
IPv6
IPv6
IPv6
IPv6

物理视图:

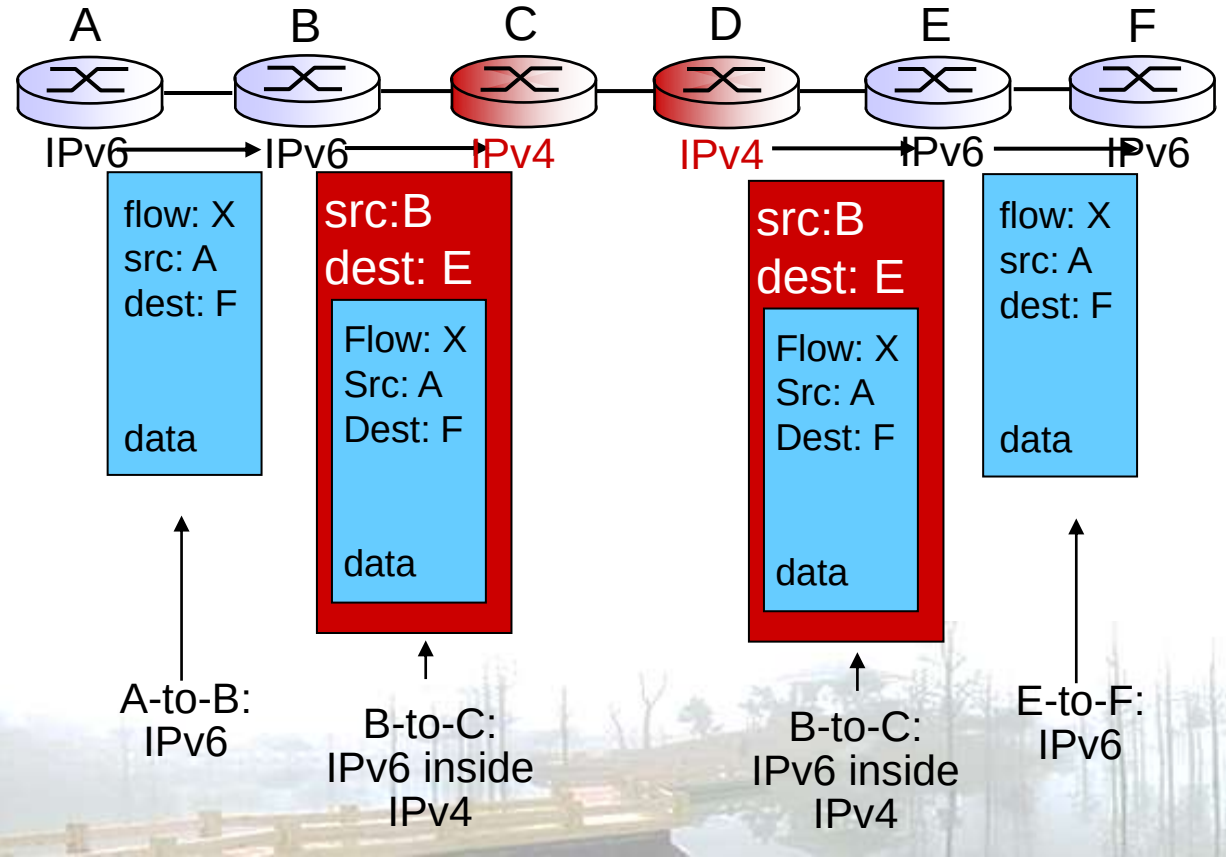

A
B
C
D
E
F
IPv6
IPv6
IPv4
IPv4
IPv6
IPv6

隧道

逻辑视图:

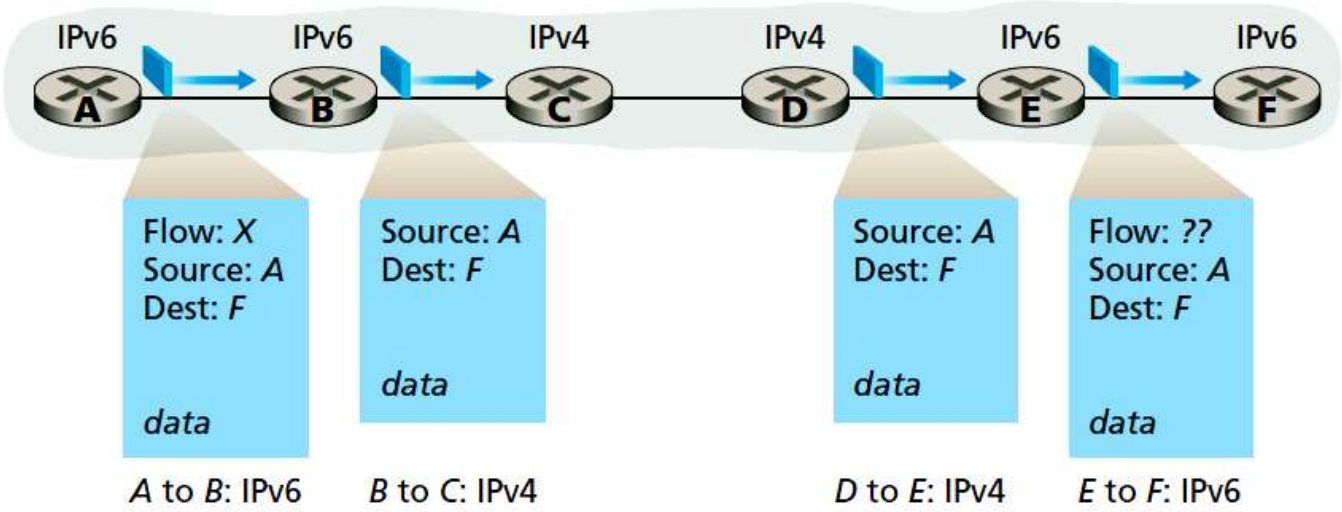

A
B
连接IPv6路由器的IPv4隧道
E
F
IPv6
IPv6
IPv6
IPv6

物理视图:


A
B
C
D
E
F
IPv6
IPv6
IPv4
IPv4
IPv6
IPv6
flow: X
src: A
dest: F
data
src:B
dest: E
Flow: X
Src: A
Dest: F
data
src:B
dest: E
Flow: X
Src: A
Dest: F
data
flow: X
src: A
dest: F
data
A-to-B:
IPv6
B-to-C:
IPv6 inside
IPv4
B-to-C:
IPv6 inside
IPv4
E-to-F:
IPv6

双栈

- 路由器同时具备收发IPv4和IPv6报文的能力
- IPv6-IPv4-IPv6报文转换中，一些v6字段丢失



E到F的IPv6报文不包含从A发出的IPv6报文的所有字段

IPv6：进展

- Google: 8% 的客户端使用IPv6访问因特网
- NIST: 1/3的美国政府网站支持IPv6

- 距离完全部署应用还有很长时间
  - 从诞生到现在已经20年，未来仍有很多年
  - 20年里应用层面发生了天翻地覆的变化
  - 为什么这么慢?