第六章 网络层基础

崔勇

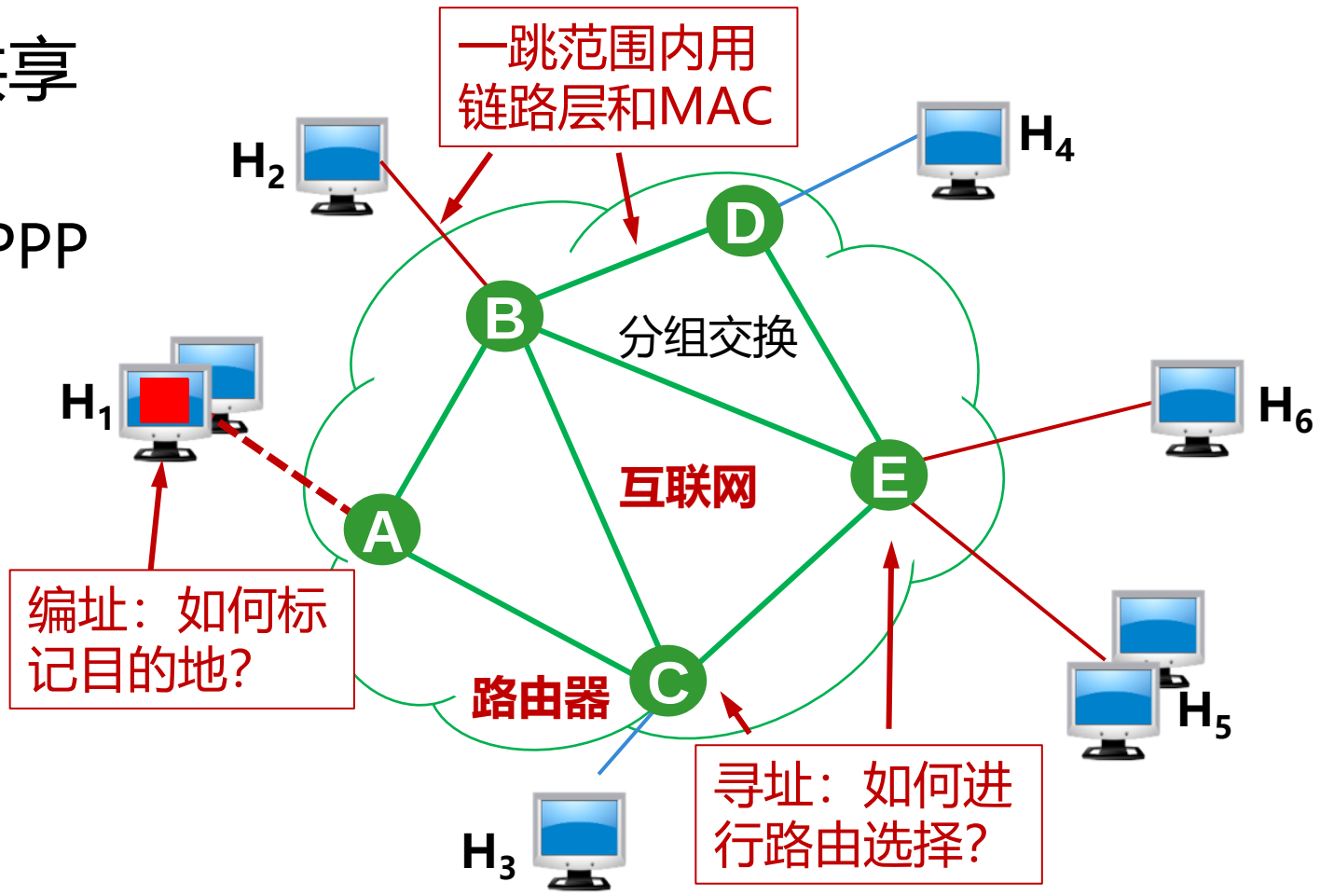
清华大学

计算机网络
教案社区

致谢社区成员	
首都师范大学 陈文龙	武汉大学 吴黎兵
西安邮电大学 谢晓燕	仲恺农业工程学院 邹莹
枣庄学院 李旭宏	辽宁大学 曲大鹏
西南民族大学 方诗虹	浙江理工大学 舒挺
内蒙古农业大学 白云莉	荆楚理工学院 余琨
华为技术有限公司 李振斌	

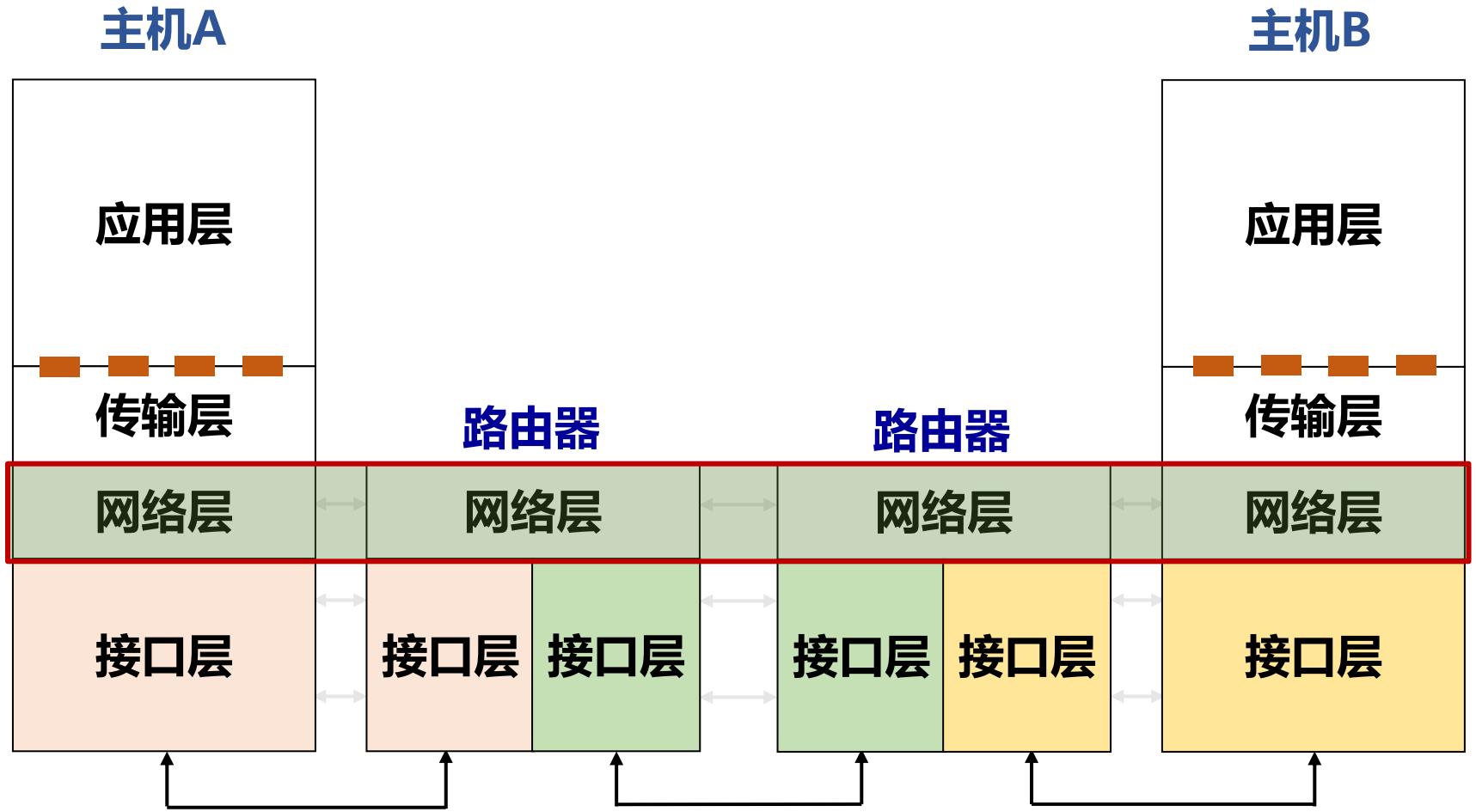
思考与展望

- 数据链路层：点到点 & 共享
  - 成帧、检错纠错编码
  - P1-P6：流量控制，重传，PPP
  - ALOHA、CSMA
  - 以太网/交换，WLAN
- 网络层服务
  - 到目的地？多跳找路？
  - 两大功能：编址 & 寻址
  - 发明：IP协议族（编址）
  - 发明：路由 & 转发（寻址）



从全世界：ping 166.111.4.100

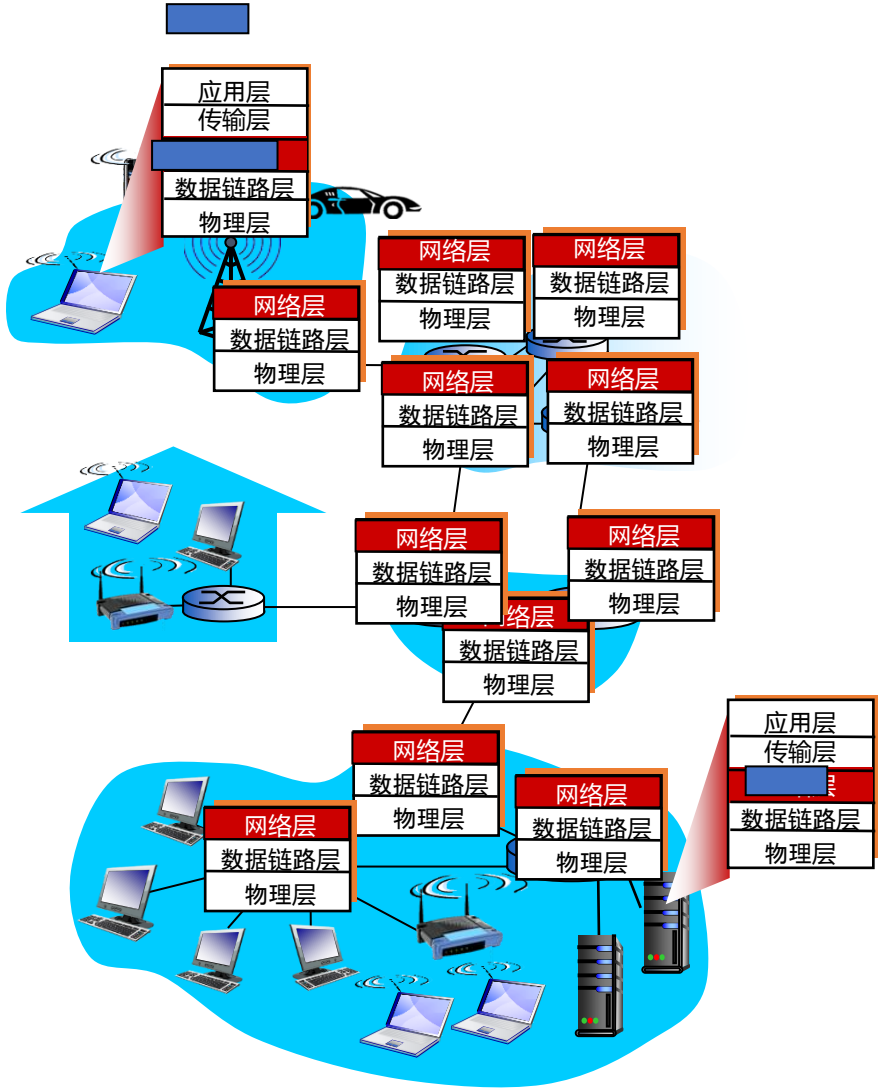
网络层在哪里？



接口层通常包括数据链路层和物理层

网络层服务的实现

- 网络层功能存在每台主机和路由器中
  - 发送端：将传输层数据单元封装在数据包中
  - 接收端：解析接收的数据包中，取出传输层数据单元，交付给传输层
  - 路由器：检查数据包首部，转发数据包
- 问题分解：网络层核心功能
  - 目标：实现设备间的多跳可达
  - 控制面：编址(Addressing)和路由(Routing)
  - 数据面：分组转发

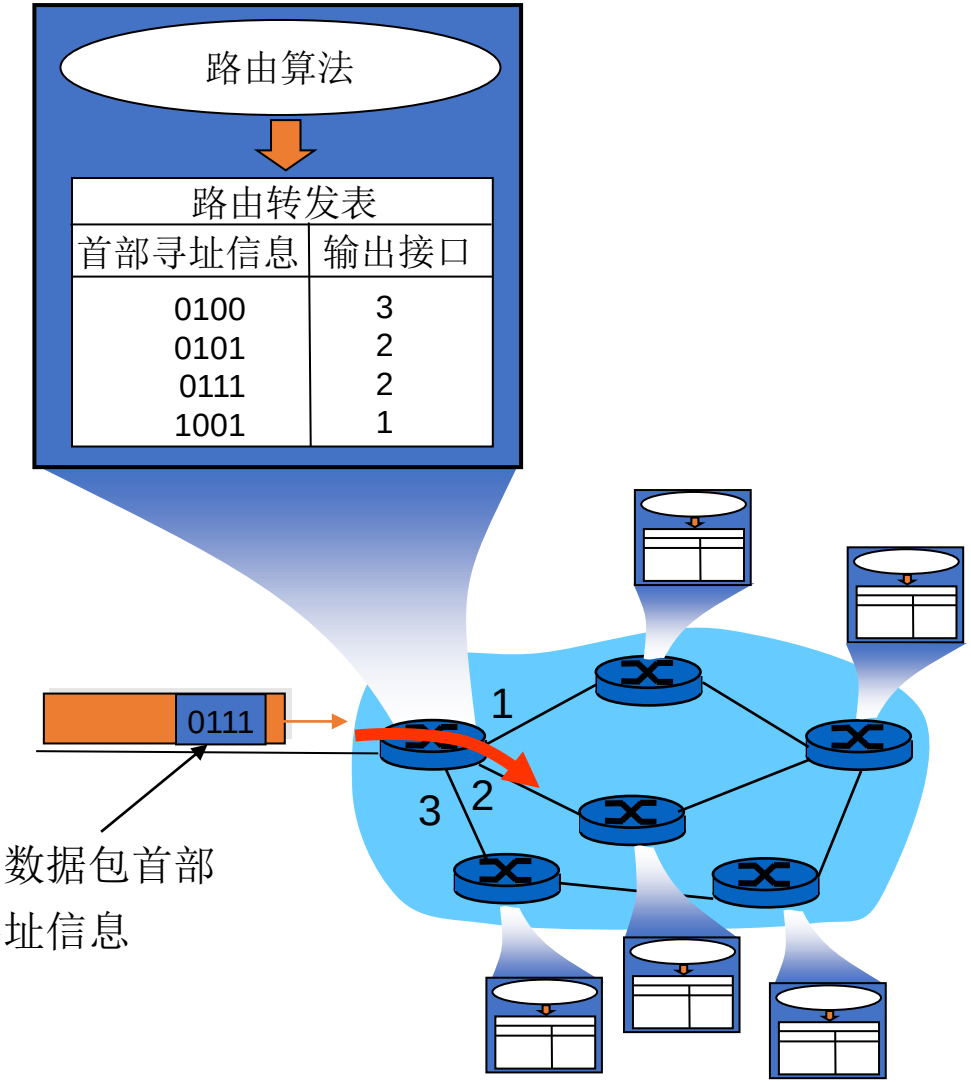


网络层关键功能

- 路由（控制面）
  - 选择数据报从源端到目的端的路径
  - 目标：大规模、动态性
  - 路由协议与算法

静态路由 v.s. 动态路由

- 转发（数据面）
  - 将数据报从路由器的输入接口传送到正确的输出接口
  - 目标：高效、快速，大容量



本节目标

1. 学习网络层服务概念
2. 理解IP协议的设计需求和设计思路
3. 掌握IPv4编址方法，子网划分和最长前缀匹配
4. 了解常见特殊地址，掌握单播、组播、广播、任播的概念
5. 了解IPv4分组格式，掌握分片重组的方法
6. 学习Internet网络层协议：DHCP, ARP, ICMP, NAT

本节内容

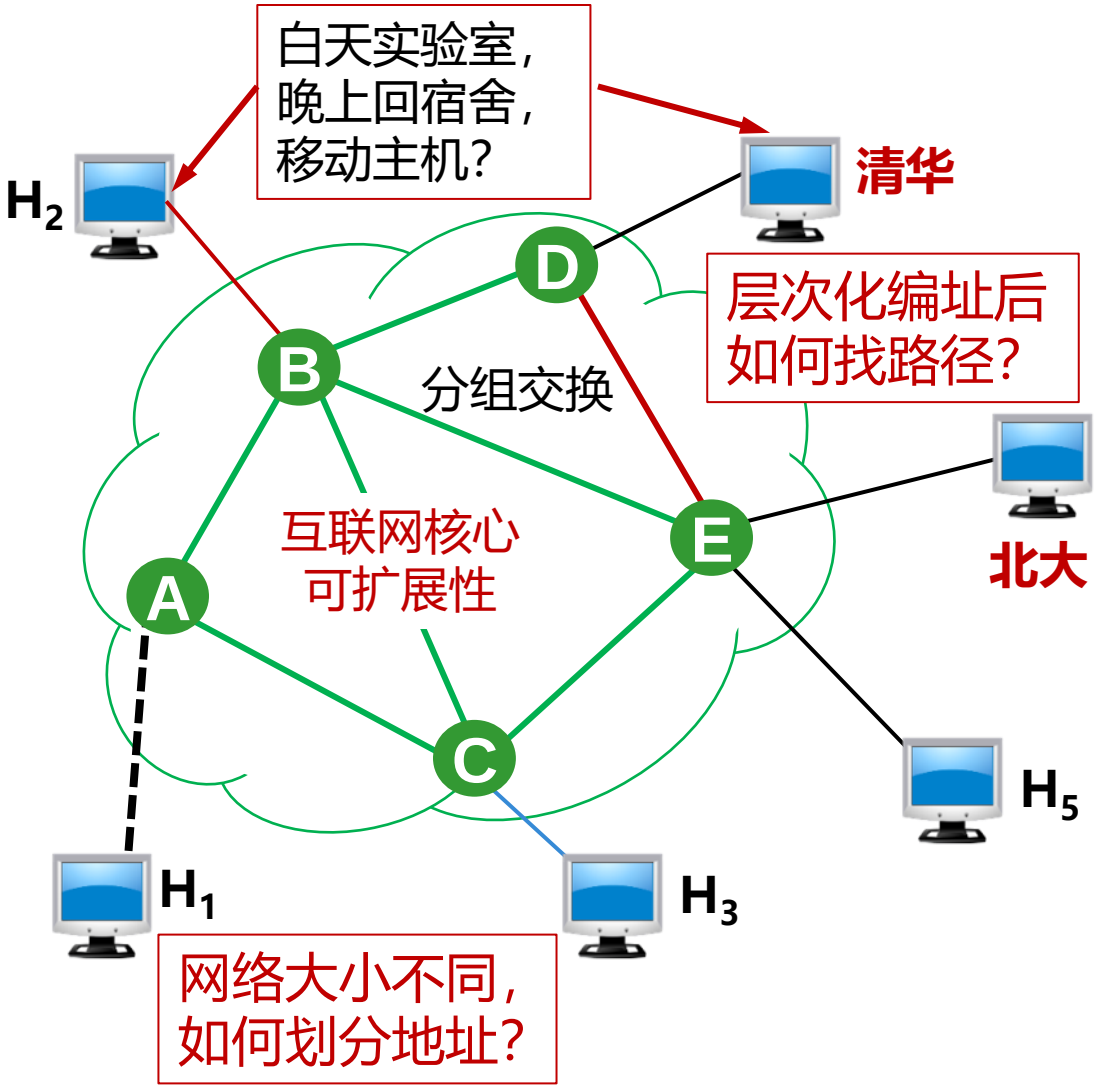
6.1 IPv4 与编址

6.2 网络层典型协议和技术

1. 分类编址
2. 无类域间路由CIDR
3. 子网划分
4. 最长前缀匹配
5. IPv4 数据报格式
6. 数据报分片
7. 特殊 IP 地址(单播、组播、广播、任播概念及地址)

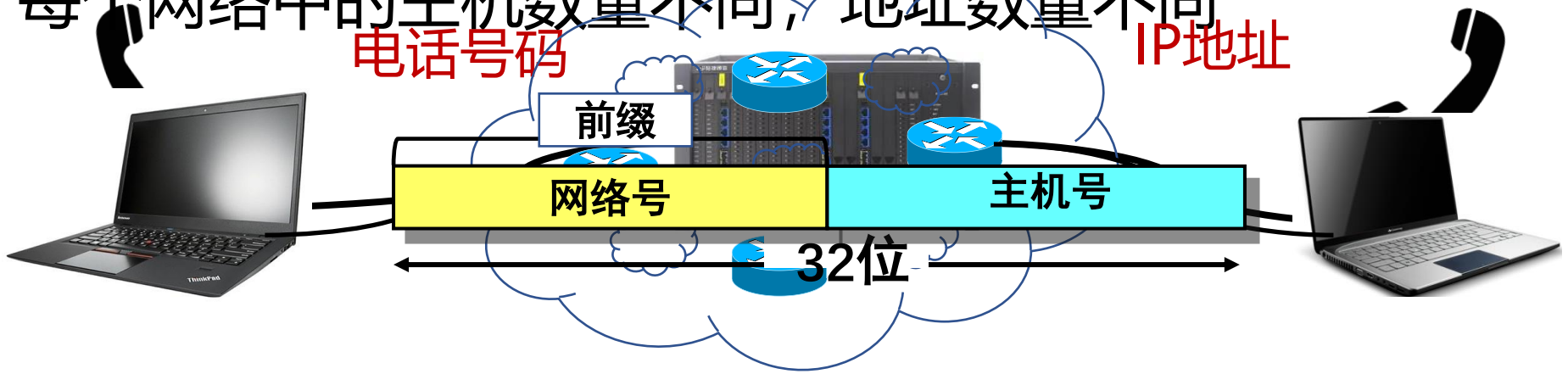
思考与发明

- 如何实现编址(Addressing)?
  - 网卡上的MAC地址管用吗?
  - 设备移动要告诉所有人吗?
  - 电话网：国家号->区号……
  - 层次化编址：8610-62785822
  - 层次化编址后如何找路径?
  - IP地址：定长还是变长?
- 传输中的其他问题?
  - 路由回路？分组损坏?
  - 不同链路帧长限制不同?



IP地址

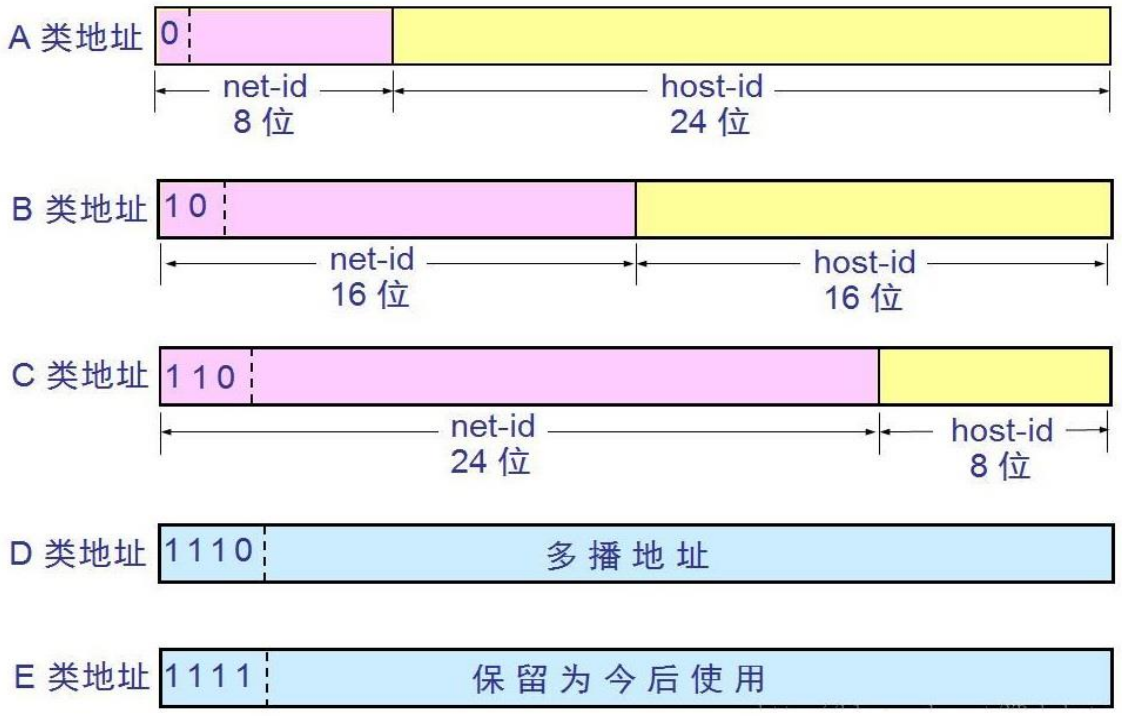
- IP地址
  - 又称逻辑地址，网络上的每一台主机（或路由器）的每一个接口都会分配一个全球唯一的32位的标识符
- 将IP地址划分为：网络地址和网络中的主机地址
- 每个网络中的主机数量不同，地址数量不同



网络号又称为网络前缀，如何给全球设备分配地址?

分类的IP地址

- IP地址的书写采用点分十进制记法，其中每一段取值范围为0到255
- IP地址分为五类：A、B、C、D、E，其中A类、B类、C类为单播地址



请判断下列地址的类型

10.2.1.1	**A类**
128.63.2.100	**B类**
201.222.5.64	**C类**
256.241.201.10	**不存在，超出范围**

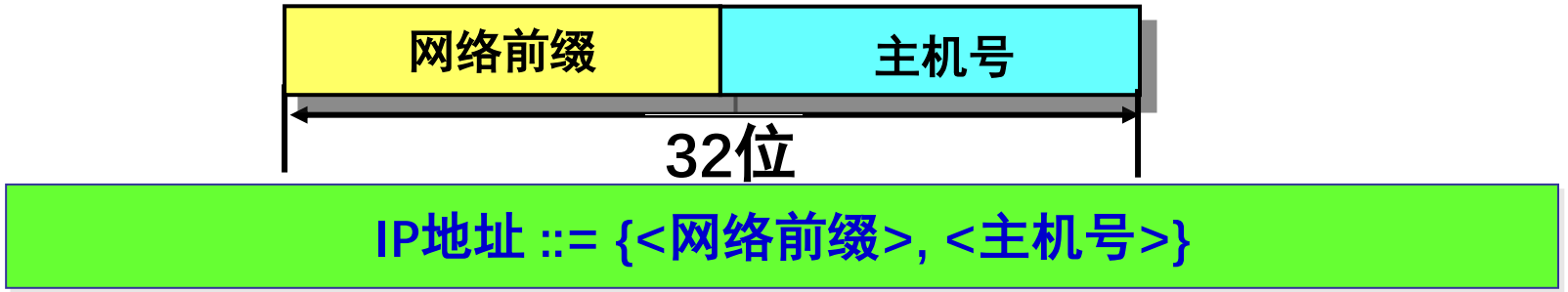
每类网络容纳主机数不同

300台主机需要B类前缀，
但浪费的部分怎么办？

无类域间路由

➢ CIDR（Classless Inter-Domain Routing）

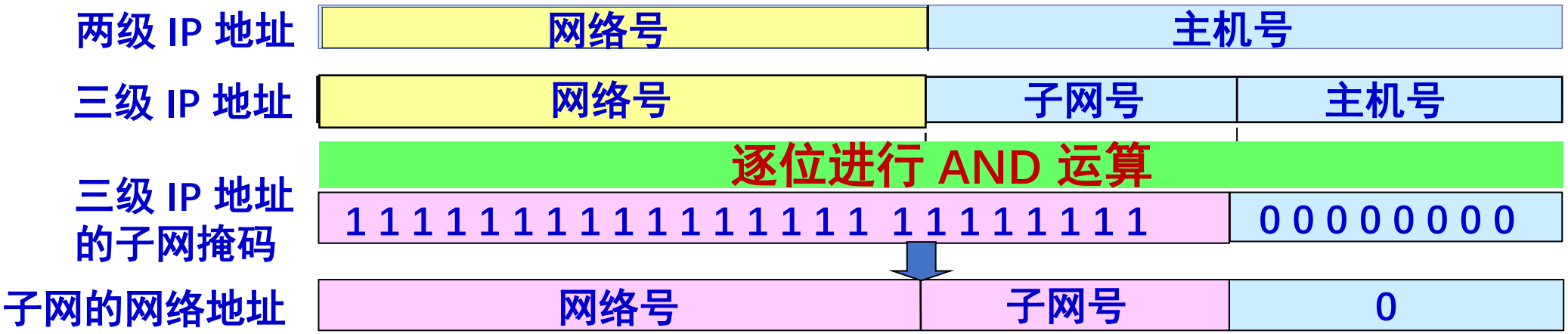
- 将32位的IP地址划分为前后两个部分，并采用斜线记法，即在IP地址后加上“/”，然后再写上网络前缀所占位数，如172.16.2.128/20



- 一个 CIDR 地址块可以表示很多地址，这种地址的聚合常称为路由聚合（route aggregation），也称为构成超网 (supernet)
- 聚合技术在Internet中大量使用，导致大量前缀重叠

子网划分

- 如何减少 IP 地址的浪费，使得网络的组织更加灵活、便于维护和管理?
- 子网划分(subnetting)，在网络内部将一个网络块进行划分以，供多个内部网络使用，对外仍是一个网络
- 子网(subnet)，一个网络进行子网划分后得到的一系列结果网络称为子网
- 子网掩码(subnet mask)，与 IP 地址一一对应，是32 bit 的二进制数，置1表示网络位，置0表示主机位



子网划分

地址	172	16	2	160
掩码	255	255	255	192
	10101100	00010000	00000010	10100000
	11111111	11111111	11111111	11000000
前缀	10101100	00010000	00000010	10000000
	10101100	00010000	00000010	10111111

逐位进行 AND 运算

前缀 ← → 主机位

主机位全0，子网地址
172.16.2.128

主机位全1，广播地址
172.16.2.191

可分配IP地址范围
172.16.2.128+1~
172.16.2.191-1

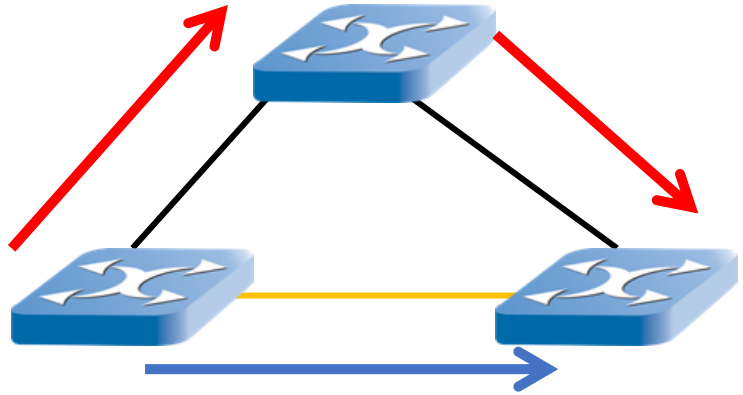
子网拥有主机数量
$2^n-2=62$（n=6）

路由转发分组时基于路由匹配选择出接口

最长前缀匹配

- CIDR + 子网划分
  - 细粒度分配地址前缀
  - 相同前缀的网段地址可以统一路由
  - 地址结构可以无冲突地按照树形逻辑管理
- 树形？任意拓扑？
  - 子网间互联，有必要绕道上层吗？
  - 如果都经过上层，方案扩展性？
  - ”抄近道“显然更优

路由转发表	
首部寻址信息	输出接口
0100	3
0101	2
0111	2
1001	1



如何用某种机制描述这种“抄近道”的路由策略？

路由转发分组时基于路由匹配选择出接口

最长前缀匹配

最长前缀匹配（Longest prefix match）

- CIDR可变长子网掩码以及路由聚合，需要最长前缀匹配来实现最精确匹配
- IP地址与IP前缀匹配时，总是选取子网掩码最长的匹配项
- 主要用于路由器转发表项的匹配，也应用于ACL规则匹配等

IP前缀（2种描述方式）		出接口号
200.23.16.0/21	11001000 00010111 00010	0
200.23.24.0/23	11001000 00010111 0001100	1
200.23.24.0/21	11001000 00010111 00011	2
Otherwise 0.0.0.0/0	--	3

IP地址：200.23.22.161 （11001000 00010111 00010110 10100001），接口0

IP地址：200.23.24.170 （11001000 00010111 00011000 10101010），接口1

最长前缀匹配

2.128.0.0/9	00000010 1 0000000	interface 1
2.192.0.0/10	00000010 11 000000	interface 2
2.0.0.0/8	00000010 00000000	interface 3
2.2.3.0/24	00000010 00000010 00000011	interface 4
0.0.0.0/0		interface 5

根据最长前缀匹配，下述目的IP将匹配哪个表项（出接口）？

2.5.1.2　00000010 00000101 00000001 00000010　Interface 3

2.150.1.2　00000010 10010110 00000001 00000010　Interface 1

最长前缀匹配

2.128.0.0/9	00000010 10000000	interface 1
2.192.0.0/10	00000010 11000000	interface 2
2.0.0.0/8	00000010 00000000	interface 3
2.2.3.0/24	00000010 00000010 00000011	interface 4
0.0.0.0/0		interface 5

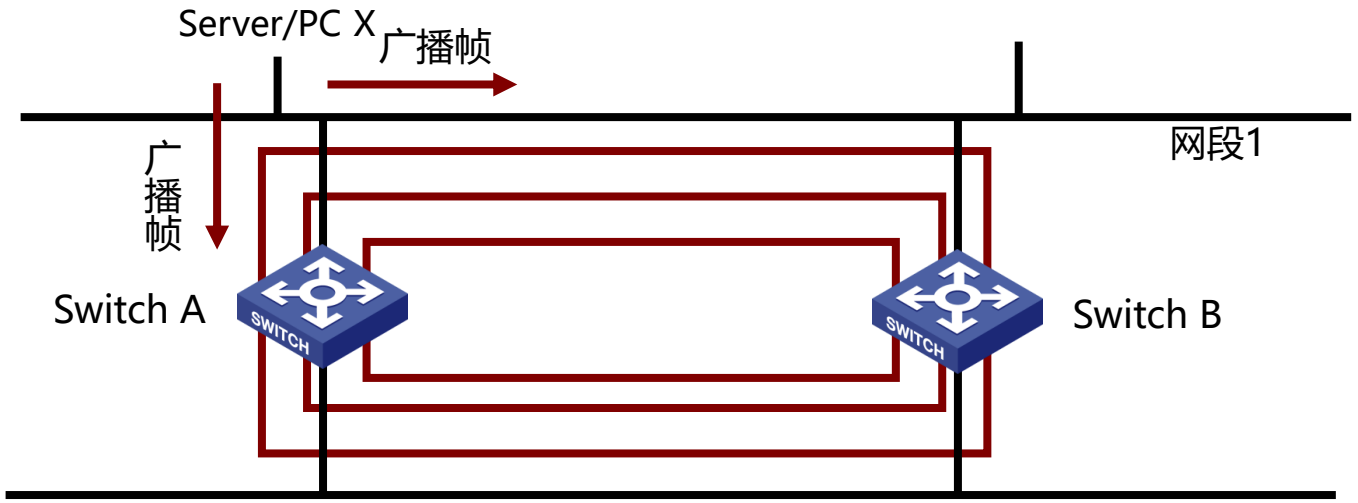
根据最长前缀匹配，下述目的IP将匹配哪个表项（出接口）？

2.200.1.2 00000010 11001000 00000001 00000010 Interface 2

3.150.1.2 00000011 10010110 00000001 00000010 Interface 5

IP协议其他功能

- 路由回路问题
  - 路由环路使数据包不断循环，耗尽网络资源
  - 链路层的生成树协议很好，但是……
  - 网络层的路由器负责消除环路？路由协议？
- 数据转发过程的终极解决方案？ 计数器TTL



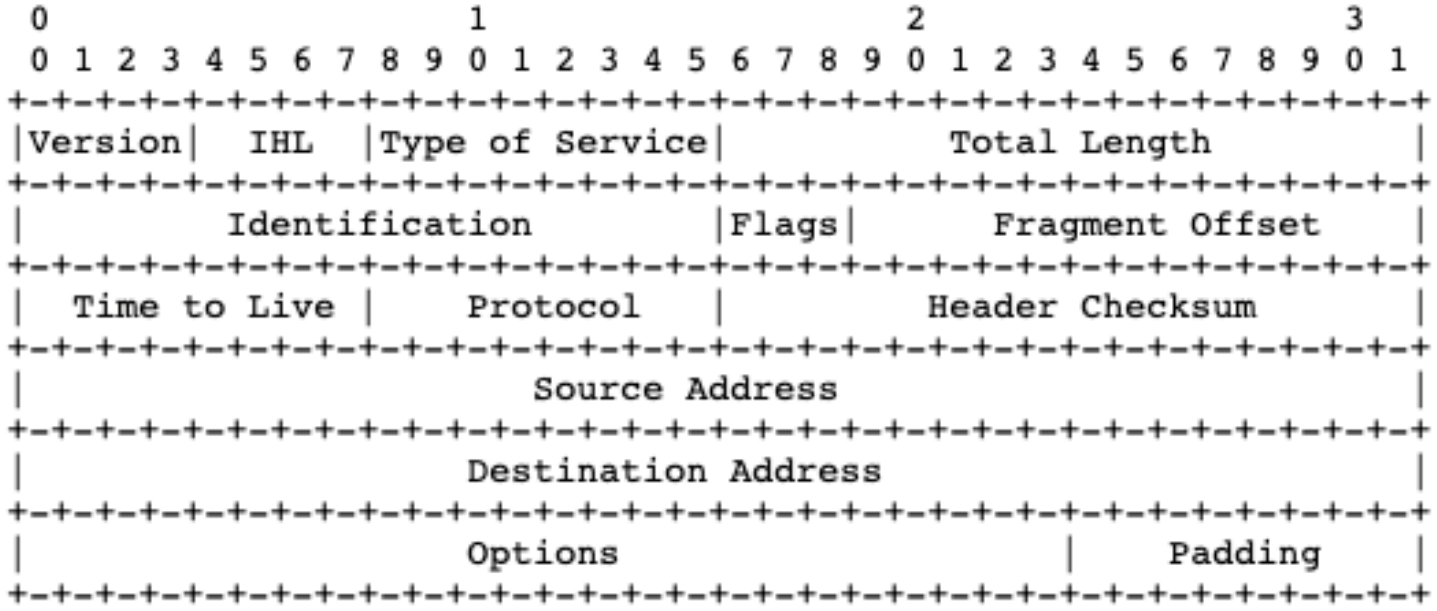
IP协议功能需求

- 网络层基本功能
  - 支持多跳寻路将IP数据报送达目的端：*目的IP地址*
  - 表明发送端身份：*源IP地址*
  - 根据IP头部协议类型，提交给不同上层协议处理：*协议*
- 其它相关问题
  - 数据报长度大于传输链路的MTU的问题，通过分片机制解决：*标识、标志、片偏移*
  - IP报头错误导致无效传输，通过头部校验解决：*首部校验和*
  - 防止循环转发浪费网络资源（路由错误、设备故障...），通过跳数限制解决：*生存时间TTL*

IPv4协议

- IPv4协议
  - Internet Protocol，网际协议版本4，一种无连接的协议，是互联网的核心，也是使用最广泛的网际协议版本，其后继版本为IPv6

- IP协议两个基本功能
  - 编址(addressing)
  - 分片(fragmentation)



RFC 791

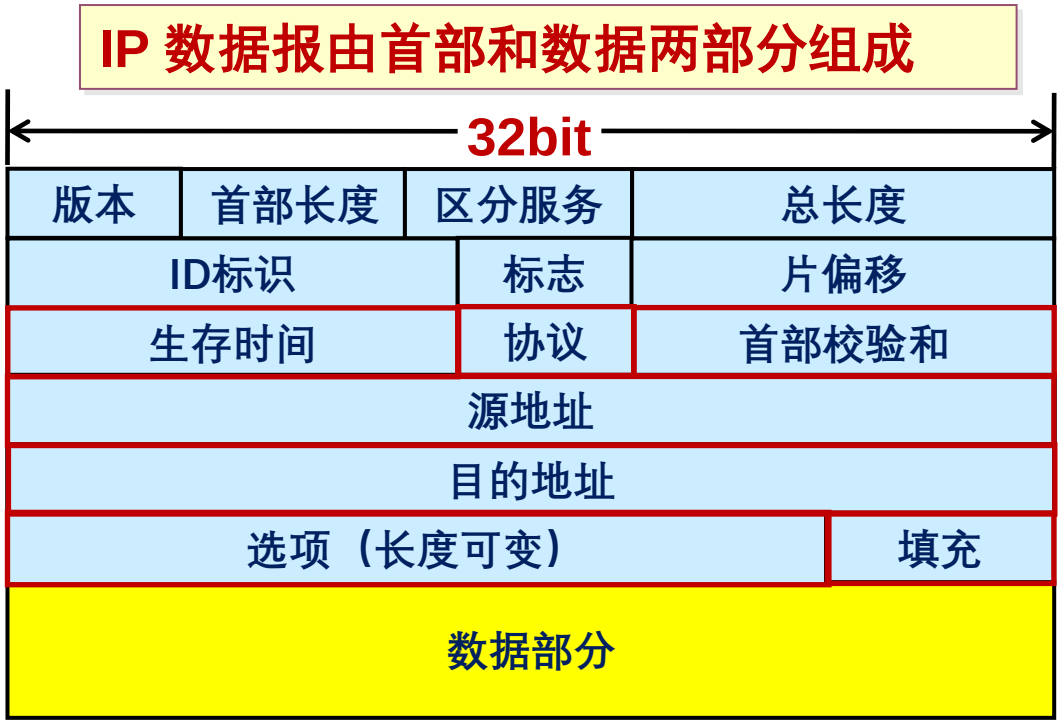
IPv4数据报格式

- 版本： 4bit ，表示采用的IP协议版本
- 首部长度： 4bit，表示整个IP数据报首部的长度
- 服务类型ToS： 8bit，该字段一般情况下不使用
- 总长度： 16bit，表示整个IP报文的长度,能表示的最大字节为2^16-1=65535字节
- 标识ID： 16bit，IP软件通过计数器自动产生，每产生1个数据报计数器加1，在ip分片后用来标识同一片的分片
- 标志： 3bit，目前只有两位有意义； MF，置1表示后面还有分片，置0表示这是数据报片的最后1个；DF，不能分片标志，置0时表示允许分片
- 片偏移：13bit，表示IP分片后，相应的IP片在总的IP片的相对位置(8 octets)

IP 数据报由首部和数据两部分组成

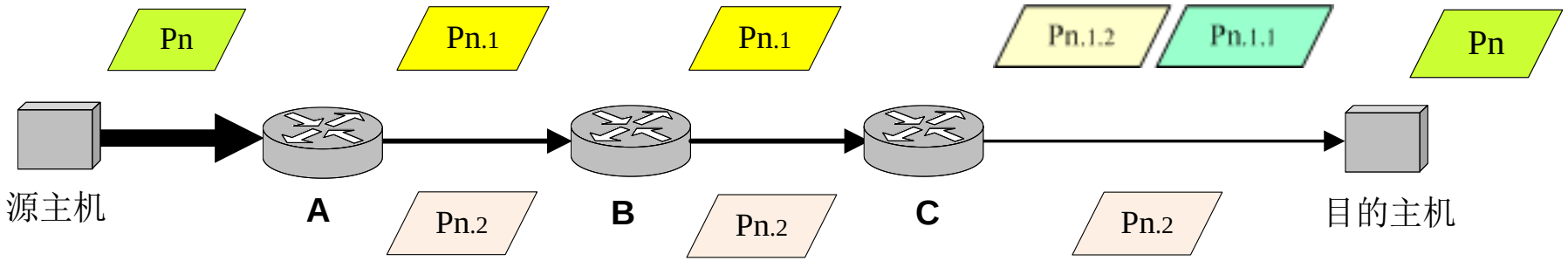
IPv4数据报格式

- 生存时间TTL(Time To Live)：8bit,表示数据报在网络中的生命周期，用通过路由器的数量来计量，即跳数（每经过一个路由器会减1）
- 协议：8bit，标识上层协议（TCP/UDP/ICMP...）
- 首部校验和：16bit，对数据报首部进行校验，不包括数据部分
- 源地址：32bit，标识IP片的发送源IP地址
- 目的地址：32bit，标识IP片的目的地IP地址
- 选项：可扩充部分，具有可变长度，定义了安全性、严格源路由、松散源路由、记录路由、时间戳等选项
- 填充Padding：用全0的填充字段补齐为4字节的整数倍

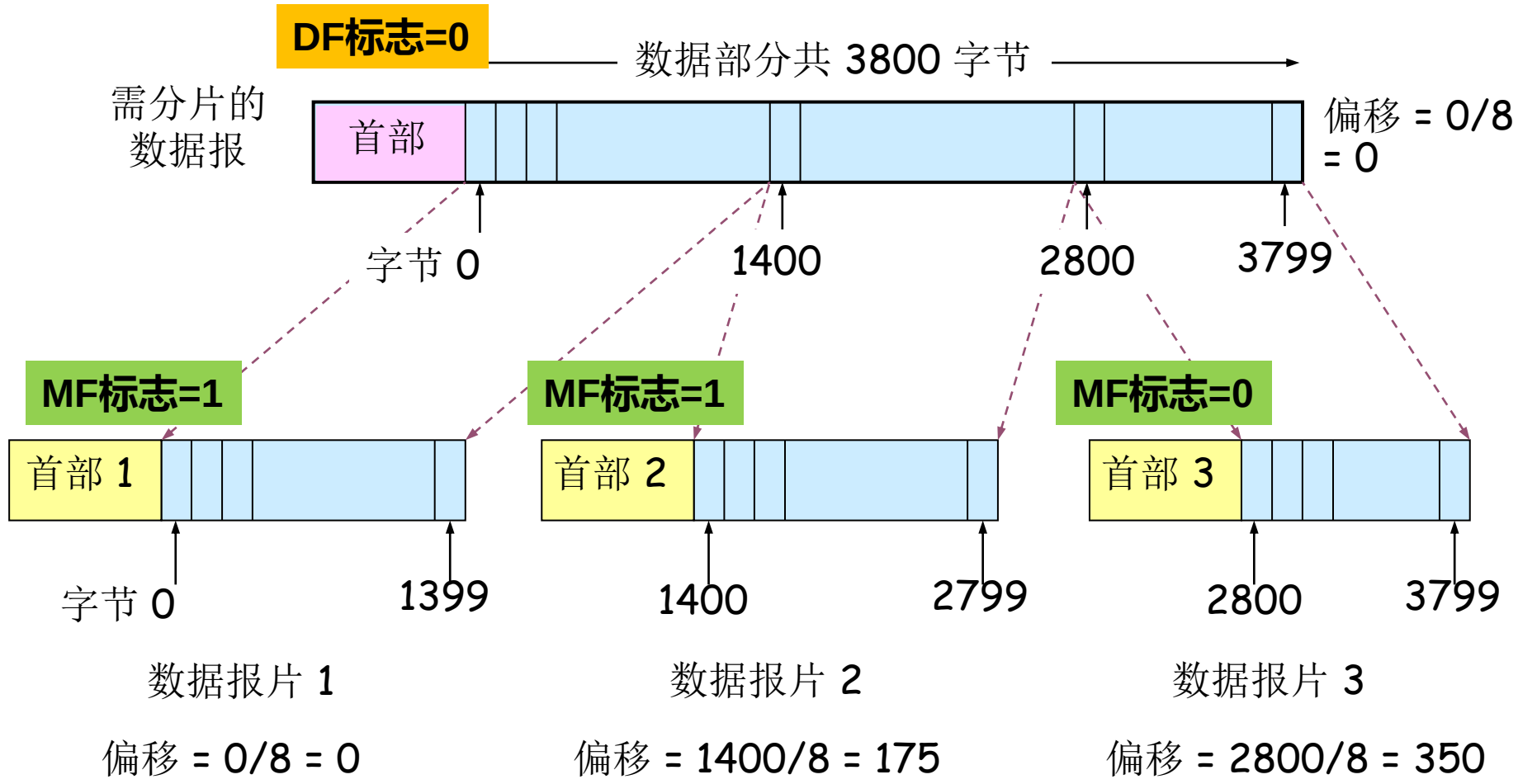


数据报分片

- 回顾：链路层最大传输单元MTU
  - 最小帧长 = 46+18 = 64B
  - 最大帧长 = 1500+18 = 1518B （最大传输单元MTU：1500B）
- 每一跳承载IP数据包的能力都可能不同
- IP数据包需要能支持被切分（IP数据包分片机制）



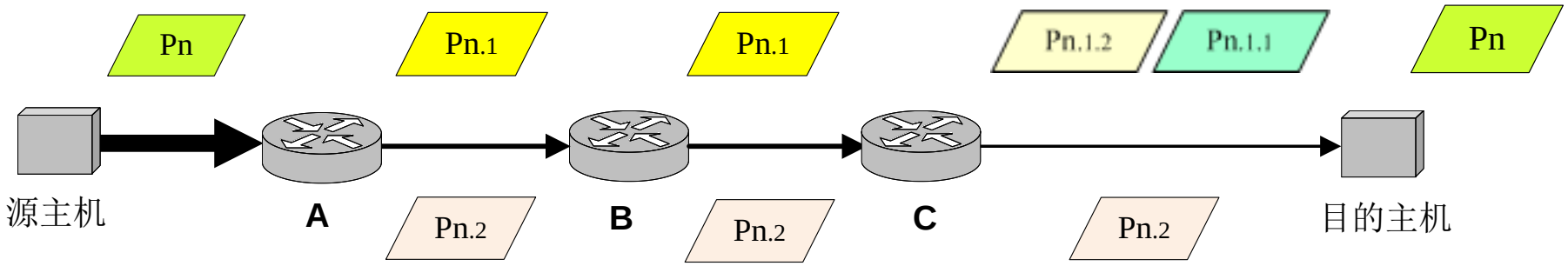
数据报分片



原始报文和分片报文具有相同的IP标识（IP头部字段）

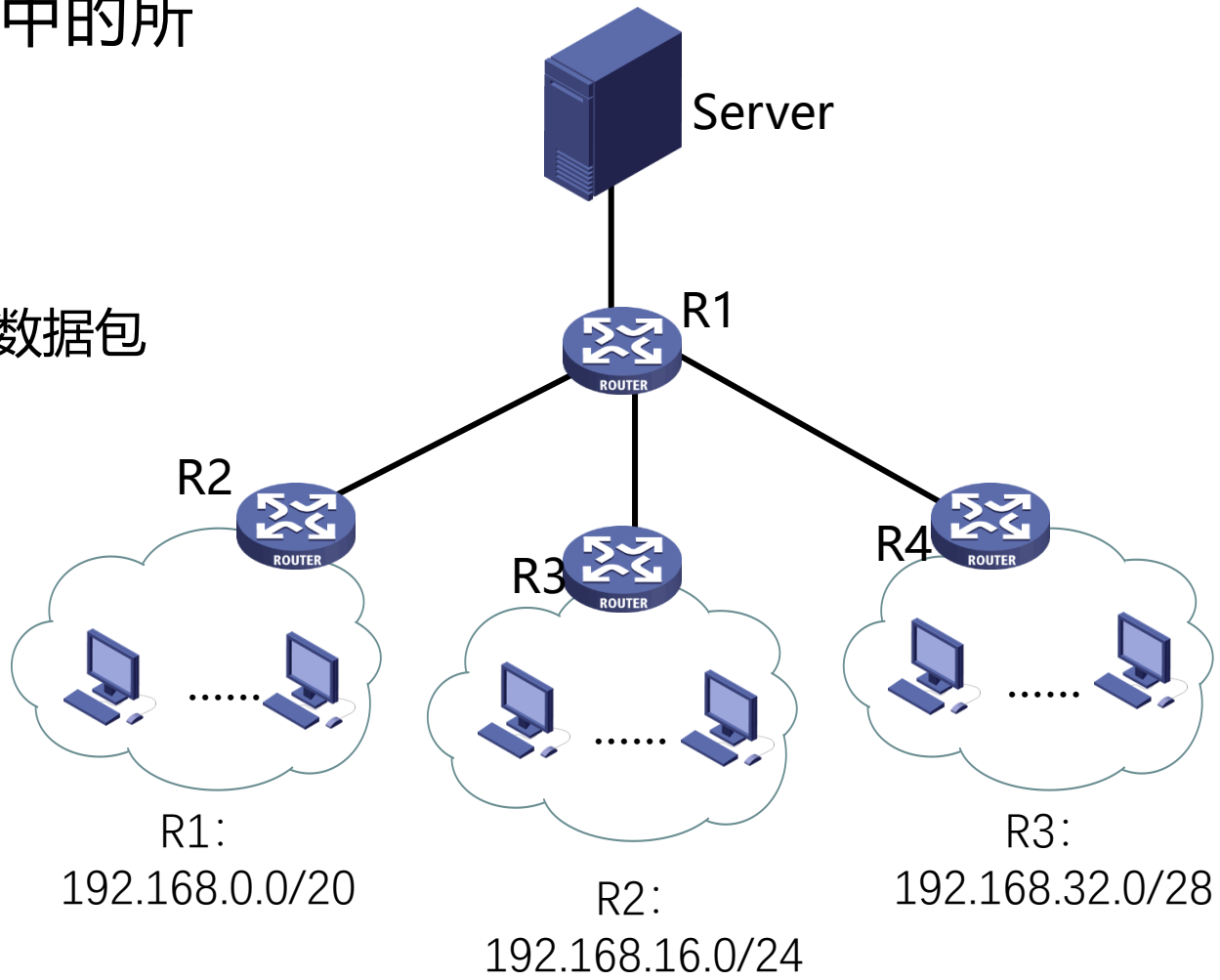
数据报分片

- IPv4分组在传输途中可以多次分片
  - 源端系统、中间路由器等，可通过标志位设定是否允许分片
- 在哪里重组?
  - 重组所需信息：原始数据报编号、分片偏移量、是否收集所有分片
  - 互联网设计原则：途中重组，实施难度大，还有安全问题?
  - IPv4分片只在目的IP对应的目的端系统进行重组
- 分片机制的优缺点探讨
  - 避免分片的IPv6：发出的数据报长度小于路径MTU（路径MTU发现机制）



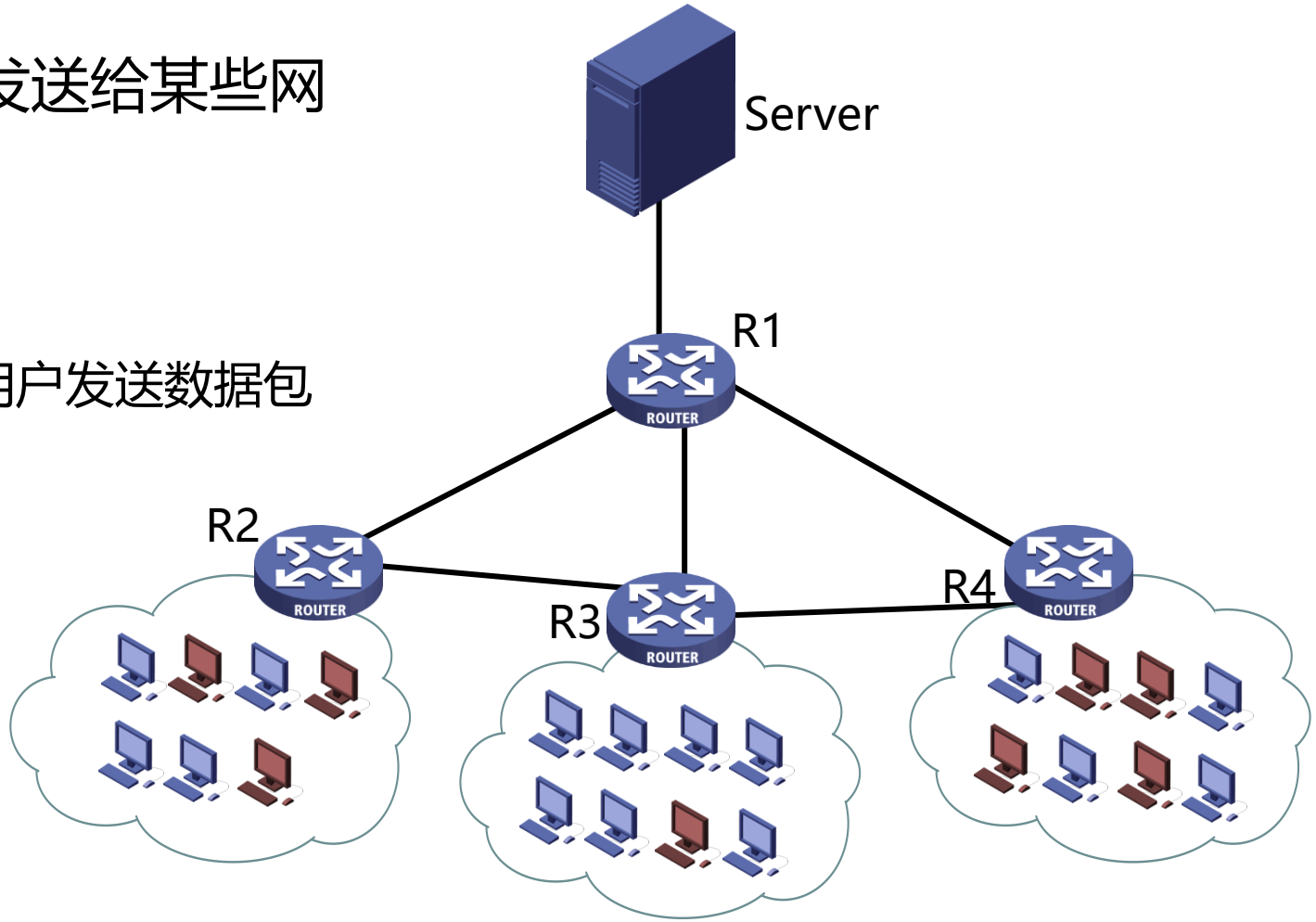
广播路由

- 服务器希望将视频广播给3个网络中的所有30个用户
- 广播（Broadcast）
  - 源主机同时给全部目标地址发送同一个数据包
- 广播地址
  - 网段内全1主机地址为广播地址
  - R1广播地址：192.168.15.255
  - R2广播地址：192.168.16.255
  - R3广播地址：192.168.32.15



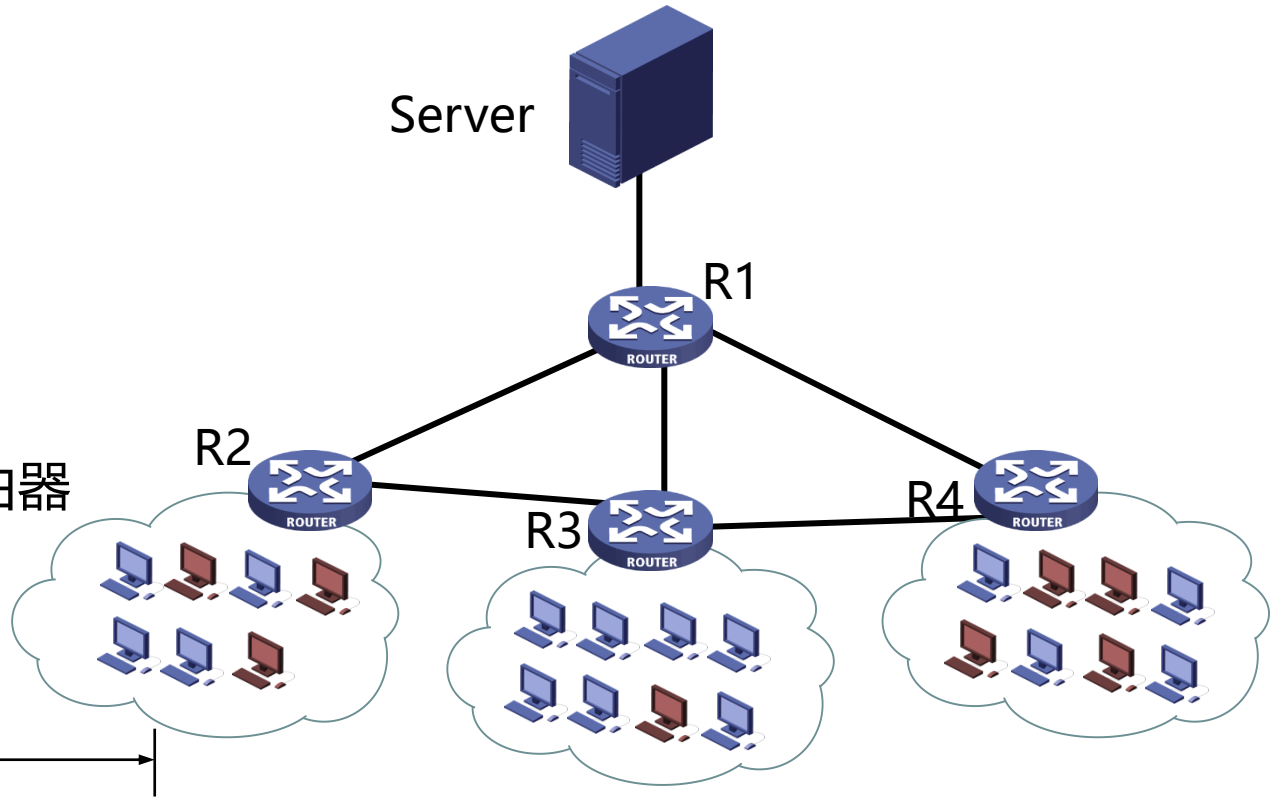
组播路由

- 服务器希望将体育直播视频发送给某些网络中的个别用户
- 组播（Multicast）
  - 源主机给网络中的一部分目标用户发送数据包



组播路由

- 常用组播地址段：224.0.0.0/24
- 局域网组播地址(一跳子网内使用)
  - 224.0.0.1 LAN上所有设备
  - 224.0.0.2 LAN上所有路由器
  - 224.0.0.5 LAN上所有OSPF路由器
  - 224.0.0.6 LAN上所有OSPF DR路由器
  - 224.0.0.9 LAN上所有RIPv2路由器
  - 224.0.0.251 LAN上所有DNS服务器



D 类地址（组播地址）

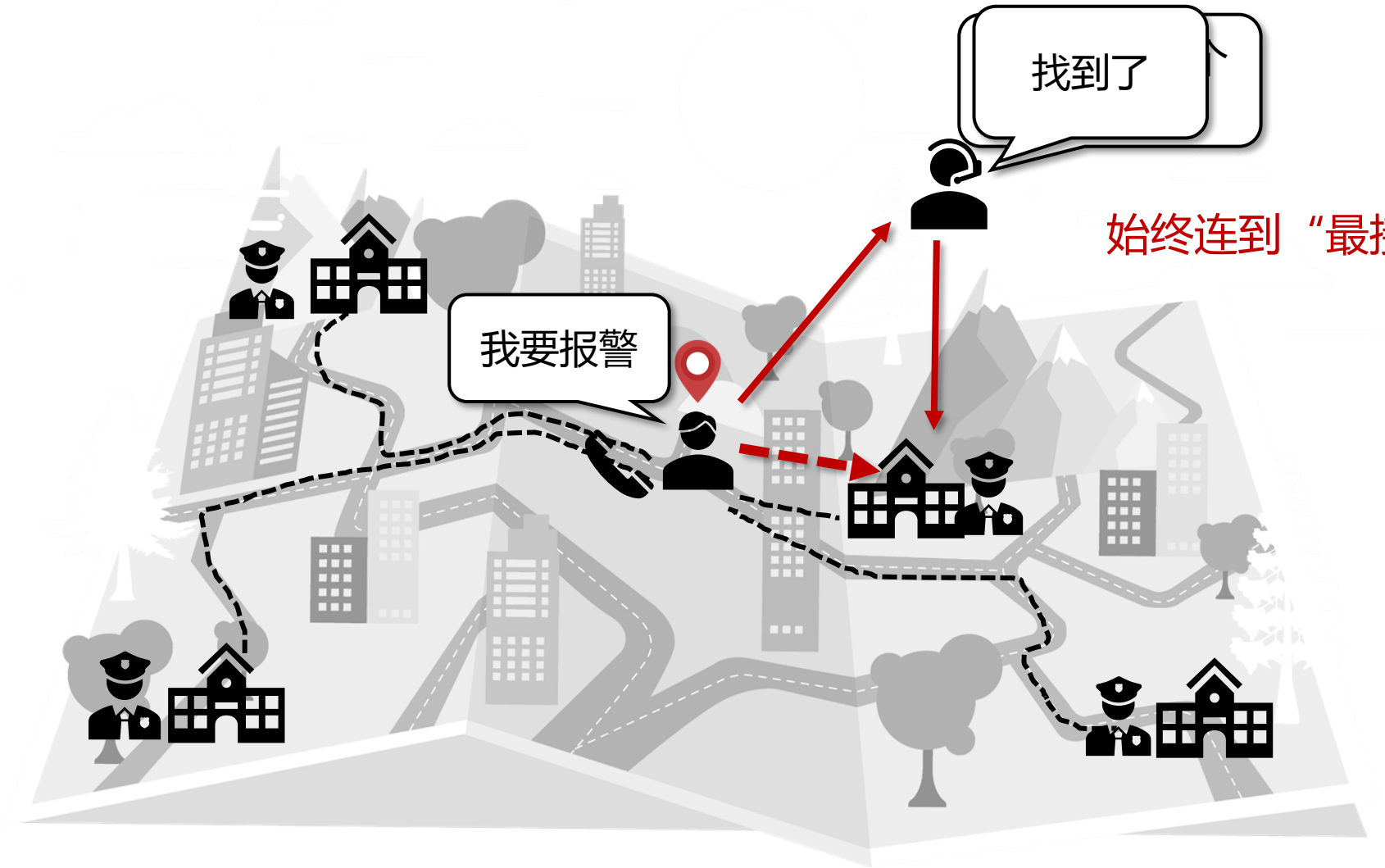
组播的应用

- 音频/视频会议
- 共享电子白板
- 数据分发
- 实时数据组播：音频视频点播、网络收看体育比赛直播、炒股……
- 游戏与仿真：同时有大量参与者的网络游戏

互联网设计原则
如何降低网络开销?
聪明终端笨网络?




选播路由（任意播）



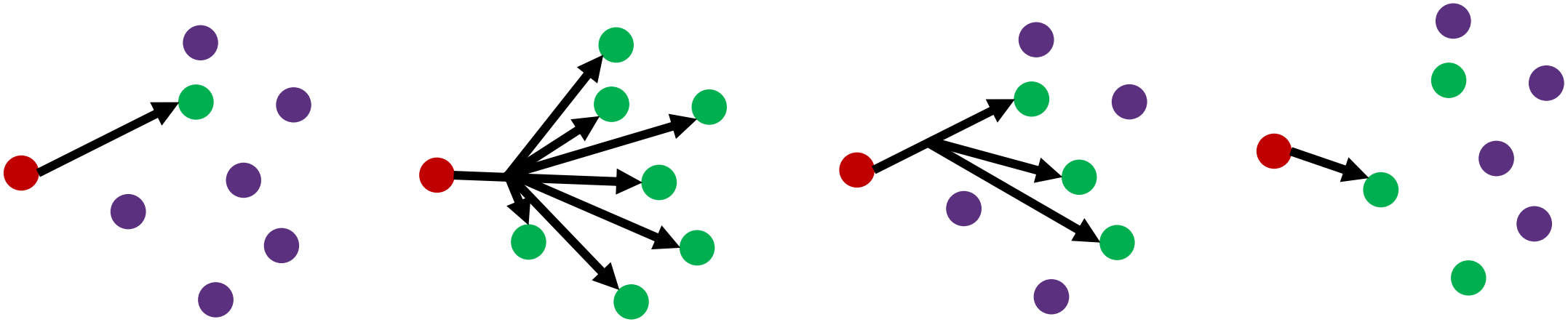
始终连到“最接近”的服务器

选播路由（任意播）

- 选播（Anycast）
  - 将数据包传送给最近的一个组成员
  - 在有多个服务器的情况下，用户希望快速获得正确信息，而不在乎从哪个服务器获得
- 选播的典型应用：DNS
  - 在没有指定DNS服务器的情况下，用户将始终连接到“最近”(从路由协议角度来看) 服务器，可以减少延迟，并提供一定程度的负载平衡
  - 易于配置管理，不必根据服务器/工作站的部署位置(亚洲、美国、欧洲)配置不同的DNS服务器，而是在每个位置配置一个IP地址
  - 提升高可用性，一旦服务器发生故障，用户请求将无缝转发到下一个最接近的DNS实例，而无需任何手动干预或重新配置

单播-广播-组播-任播

- 不同的需求，不同的设计
  - 单播Unicast：一对一通信，传给特定成员
  - 广播Broadcast ：一对多通信，传给子网全部成员
  - 组播Multicast：一对多通信，传给组内的部分成员
  - 任播Anycast：一对一通信，传给最近的组员



单播　广播　组播　选播

IP特殊地址

地址	用途
全0网络地址	只在系统启动时有效，用于启动时临时通信，又叫主机地址
网络127.0.0.0	指本地，用于测试，（浪费了1700万个地址☹）
全0主机地址	用于指定网络本身，称之为网络地址或者网络号
全1主机地址	用于广播，也称定向广播，需要指定目标网络
0.0.0.0/0	匹配任意地址
255.255.255.255	用于本地广播，也称有限/受限广播，无须知道本地网络地址

主机上有路由表吗?

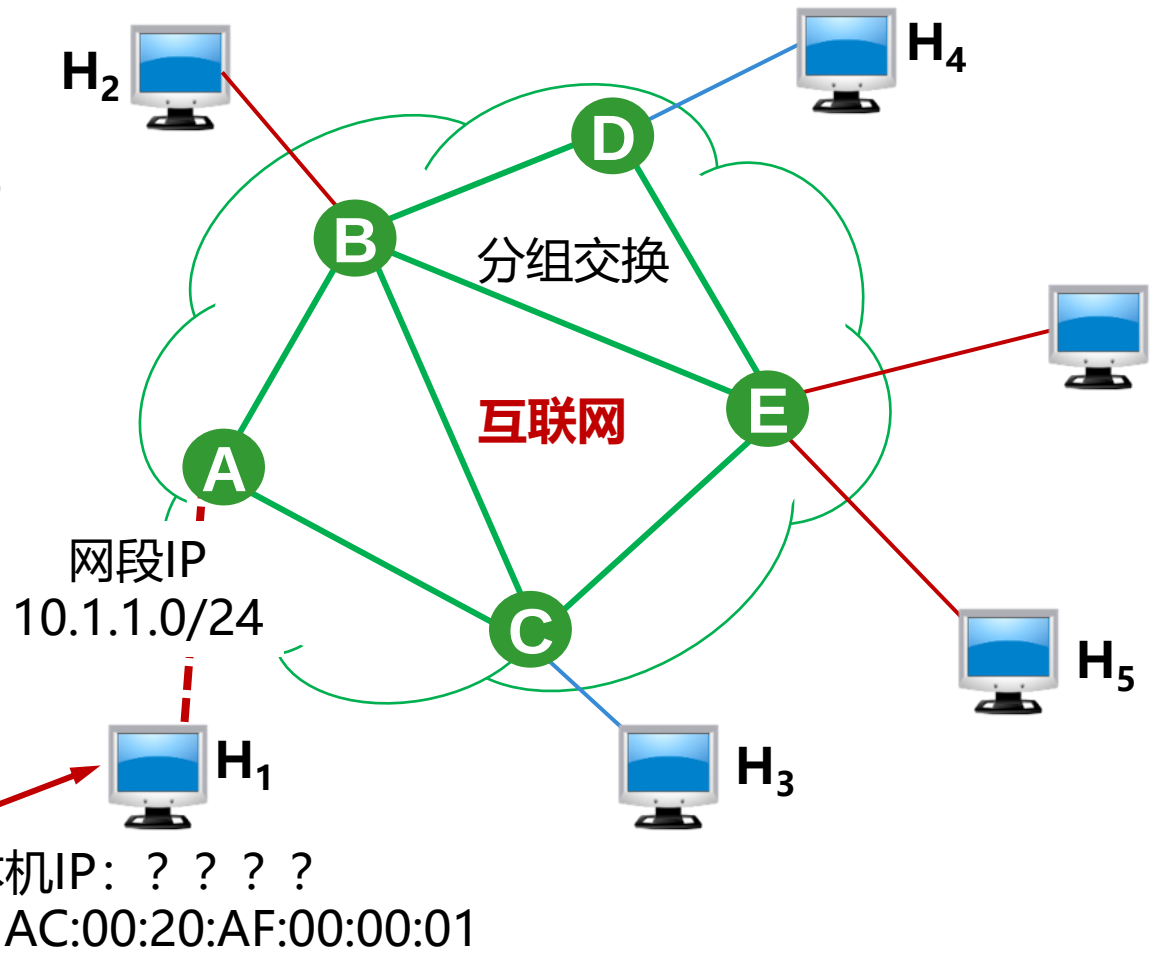
```
dgdeMacBook-Pro:~ yongcui$ netstat -nr
Routing tables

Internet:
Destination        Gateway            Flags        Netif
default            192.168.3.1        UGSc           en0
127                127.0.0.1          UCS            lo0
127.0.0.1          127.0.0.1          UH             lo0
```

IPv4与编址-小结

- 编址问题
  - 如何给全球的主机分配地址?
  - 分类地址：ABCDE类，前缀固定长度，不够灵活
  - CIDR：任意长度前缀（前缀长度 || 子网掩码）
  - CIDR地址聚合，怎么转发？怎么“抄近道“？最长前缀匹配
- IPv4协议 & 数据报分片
  - 其他功能(最大帧长 & 分片（Fragmentation）、包头检错 & 校验和、环路避免 & TTL) + 核心功能（编址）= IPv4报文格式
- 特殊的IP地址
  - 子网大小计算时的保留地址
  - 不同的需求构成了不同的设计(单播、组播、广播、任播)

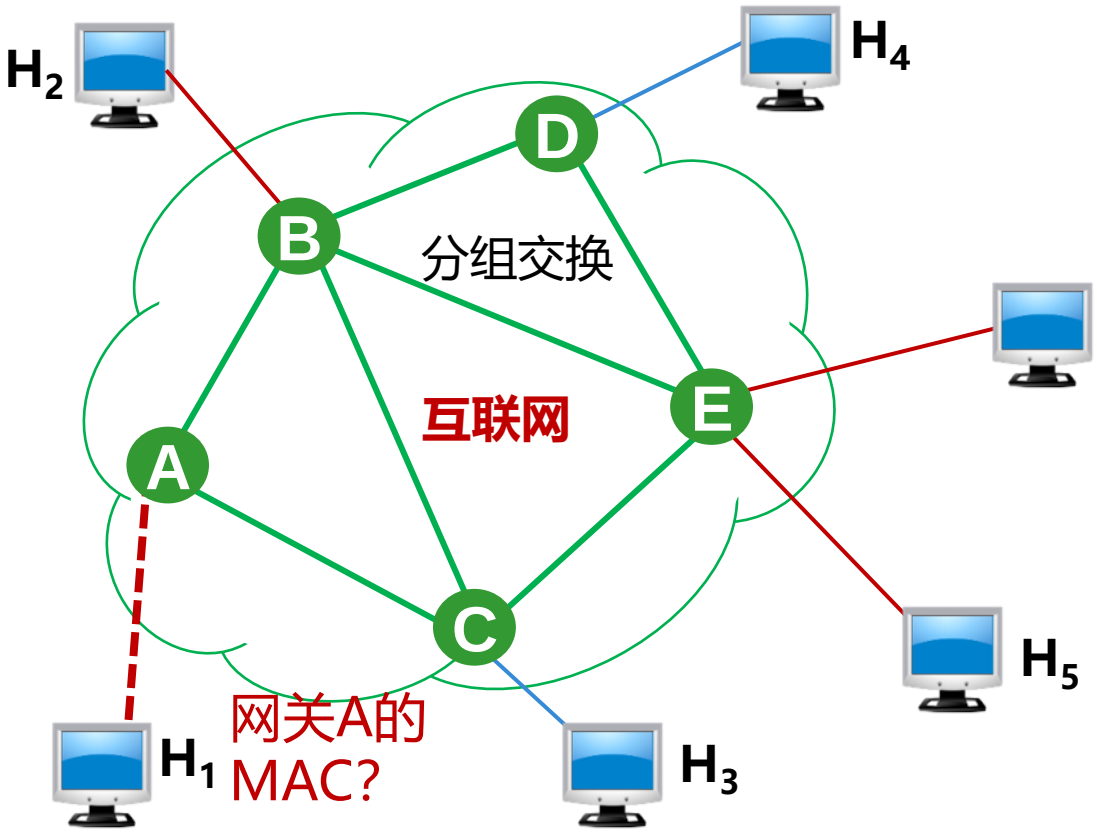
思考与发明

- IPv4协议构成了网络层服务的基石，但需要更多协议完成实际功能
- IPv4地址成段分配给网络，网络如何指定给主机?
- 原则：在网段范围内且保证唯一分配
- 手动指定每台设备的地址？太麻烦
- 动态主机配置协议DHCP



思考与发明

- 如何获取对方MAC地址?
  - 手动绑定映射表？不现实
  - ARP协议自动IP->MAC转换
- 网络出现故障怎么办?
  - 设计协议辅助排查：ICMP
- IP地址不够用怎么办?
  - 只有一个公网地址，但有多个设备☹
  - 手机笔记本时分复用?
  - 网络地址转换NAT自动配置



网段IP: 10.1.1.0/24
本机IP：10.1.1.200
MAC:00:20:AF:00:00:01

本节内容

6.1 IPv4 与编址

6.2 网络层典型协议和技术

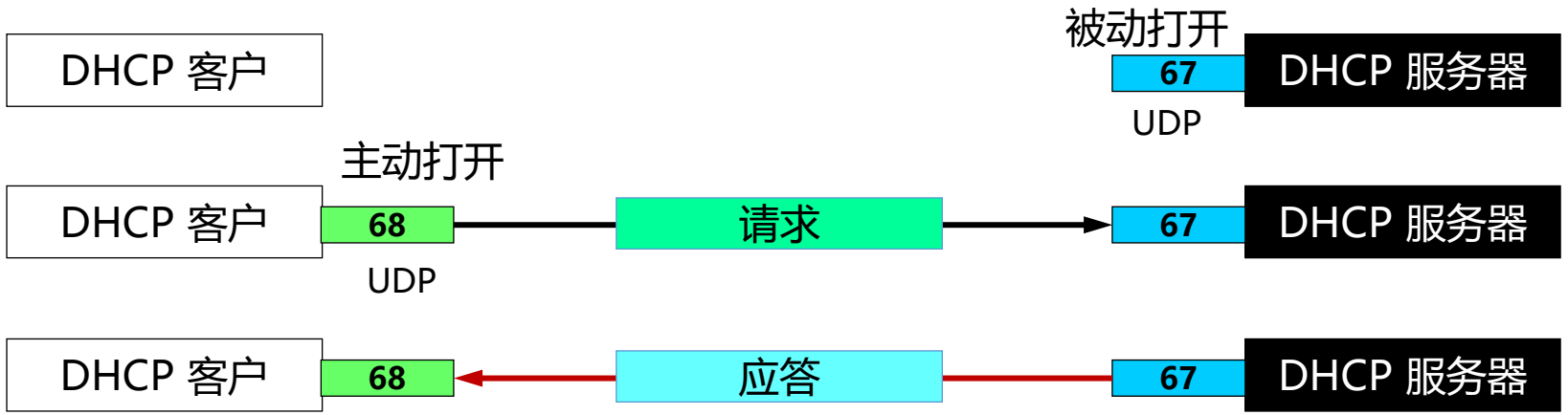
1. DHCP
2. ARP
3. ICMP
4. NAT

IPv4地址如何获取

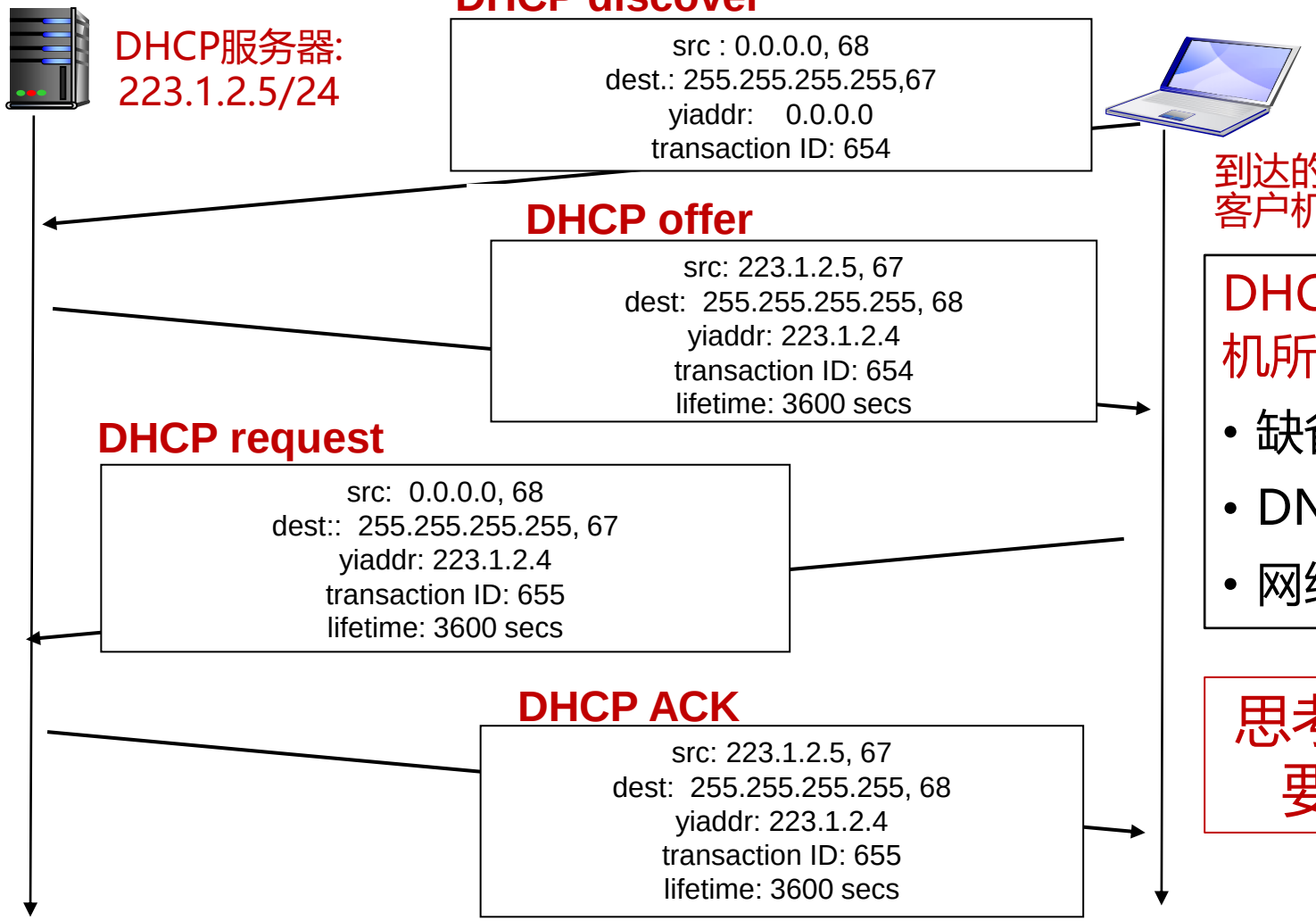
- 公有IP地址要求全球唯一
  - ICANN（Internet Corporation for Assigned Names and Numbers）即互联网名字与编号分配机构向ISP分配，ISP再向所属机构或组织逐级分配
- 静态设定
  - 申请固定IP地址，手工设定，如路由器、服务器
- 动态获取
  - 使用DHCP协议或其他动态配置协议
  - 当主机加入IP网络，允许主机从DHCP服务器动态获取IP地址
  - 可以有效利用IP地址，方便移动主机的地址获取

DHCP动态主机配置协议

- DHCP：动态主机配置协议
  - 当主机加入IP网络，允许主机从DHCP服务器动态获取IP地址
  - 可以有效利用IP地址，方便移动主机的地址获取
- 工作模式：客服/服务器模式（C/S）
  - 基于 UDP 工作，服务器运行在 67 号端口， 客户端运行在 68 号端口



DHCP 工作过程



DHCP服务不只返回客户机所需的IP地址，还包括:

- 缺省路由器IP地址
- DNS服务器IP地址
- 网络掩码

思考：为什么DHCP服务需要四次交互才能完成?

DHCP 工作过程

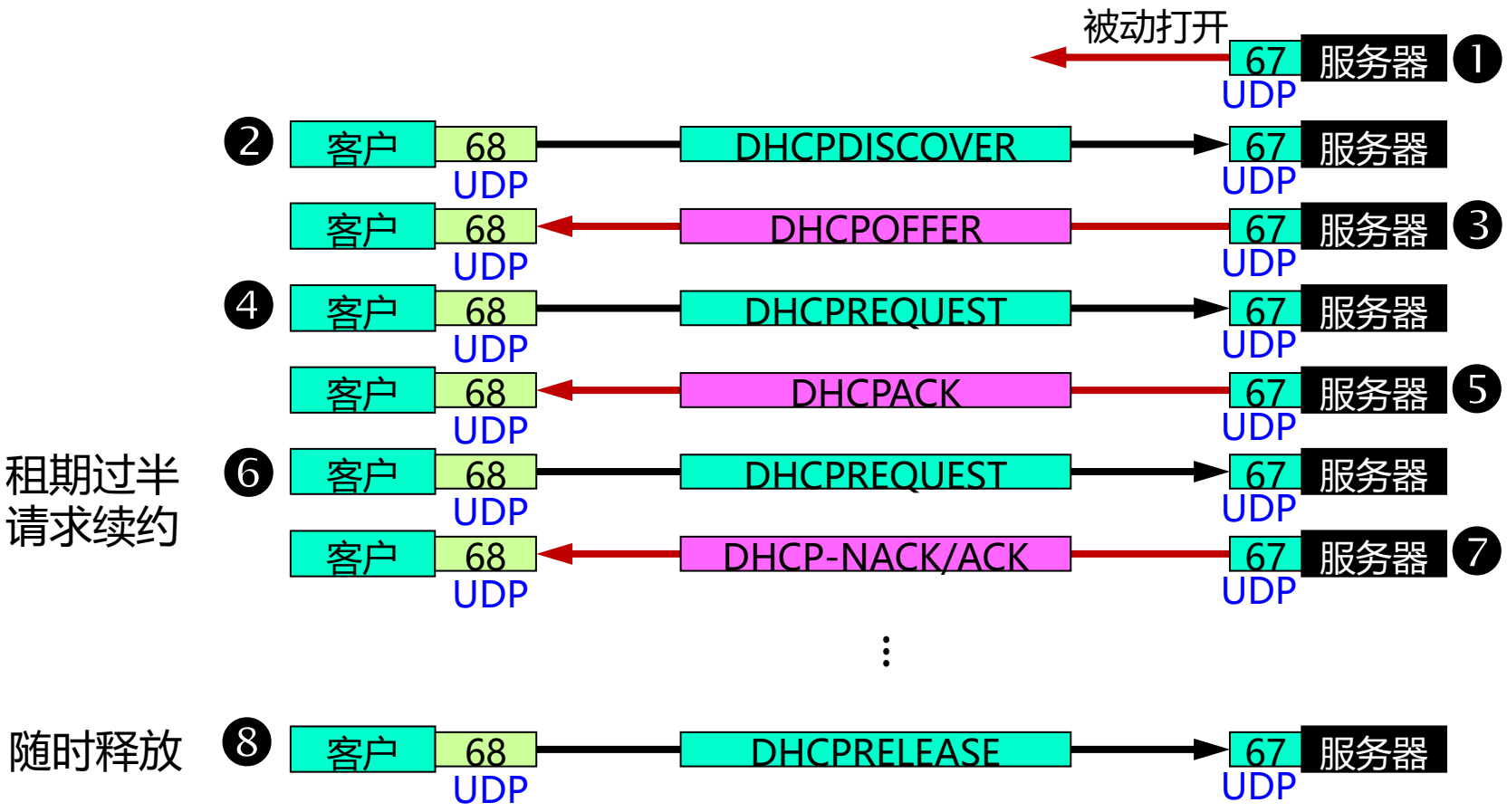
- DHCP 客户从UDP端口68以广播形式向服务器发送发现报文（DHCP DISCOVER）
- DHCP 服务器单播发出提供报文（DHCP OFFER）
- DHCP 客户从多个DHCP服务器中选择一个，并向其以广播形式发送DHCP请求报文（DHCP REQUEST）
- 被选择的DHCP服务器单播发送确认报文（DHCP ACK）

DHCP 工作过程

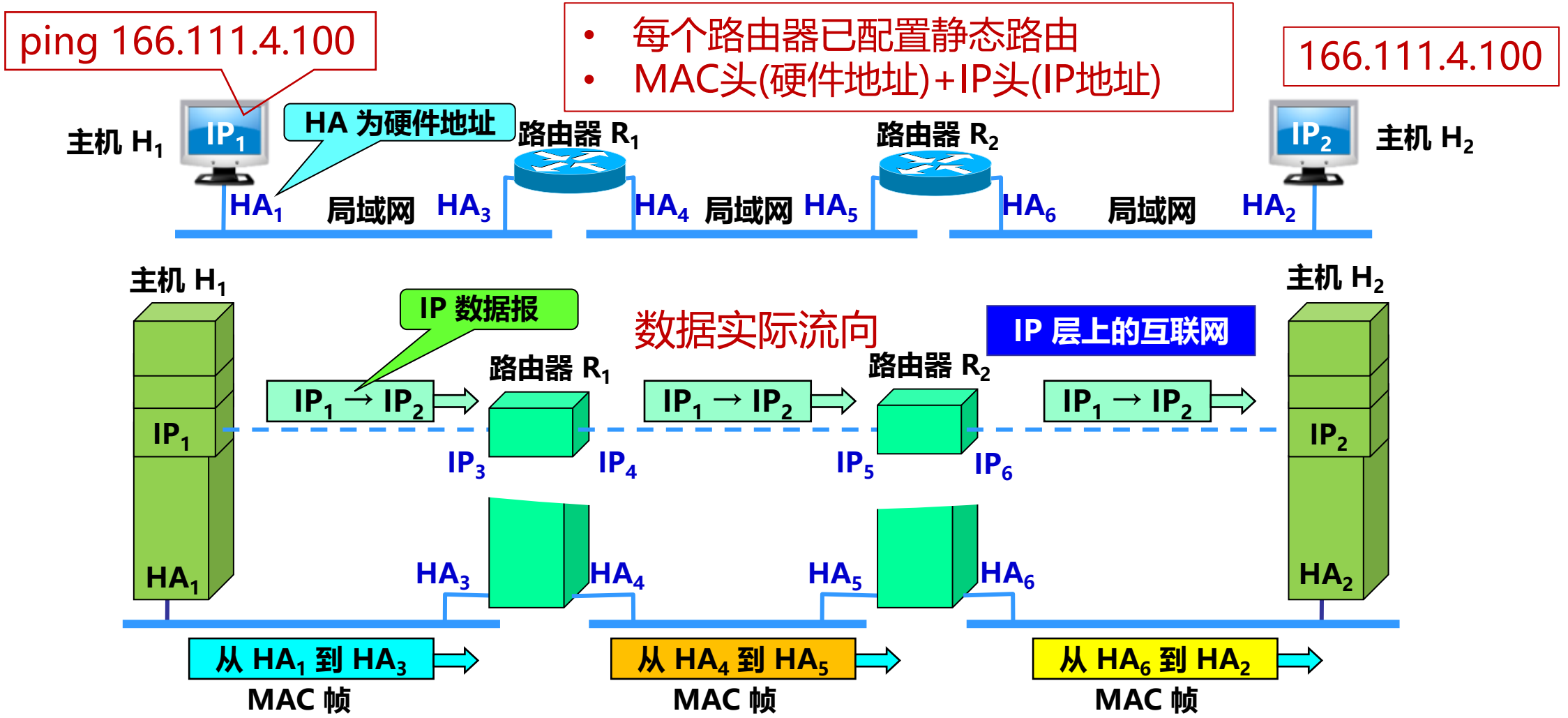
- 小结

阶段	源MAC	目标MAC	源IP	目标IP	链路层
Discover	PC机的MAC	全FF	0.0.0.0	255.255.255.255	广播
Offer	DHCP服务器（如路由器）的MAC	DHCP客户机的MAC	DHCP服务器（如路由器）的IP地址	255.255.255.255	单播
Request	PC机的MAC	全FF	0.0.0.0	255.255.255.255	广播
Ack	DHCP服务器（如路由器）的MAC	DHCP客户机的MAC	DHCP服务器（如路由器）的IP地址	255.255.255.255	单播

DHCP 完整工作过程


被动打开
服务器
67
UDP
客户
68
DHCPDISCOVER
DHCPOFFER
DHCPREQUEST
DHCPACK
DHCP-NACK/ACK
DHCPRELEASE
租期过半
请求续约
随时释放
1
2
3
4
5
6
7
8

IP 与 MAC地址

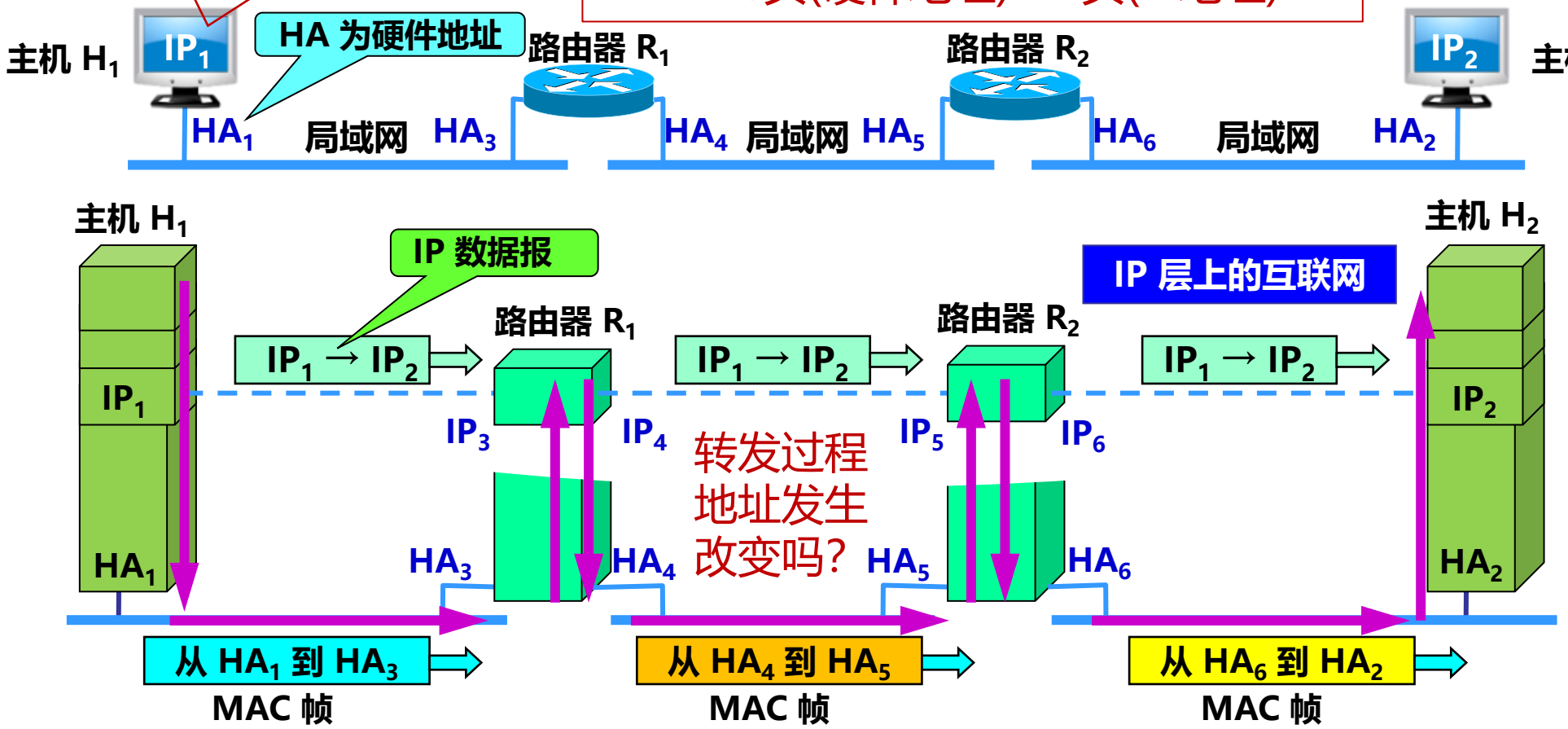

ping 166.111.4.100
• 每个路由器已配置静态路由
• MAC头(硬件地址)+IP头(IP地址)
166.111.4.100
主机 H1
IP1
HA 为硬件地址
路由器 R1
路由器 R2
IP2
主机 H2
HA1
局域网
HA3
HA4
局域网
HA5
HA6
局域网
HA2
主机 H1
IP 数据报
数据实际流向
IP 层上的互联网
路由器 R1
路由器 R2
主机 H2
IP1 → IP2
IP1 → IP2
IP1 → IP2
IP1
IP2
IP3
IP4
IP5
IP6
HA1
HA3
HA4
HA5
HA6
HA2
从 HA1 到 HA3
从 HA4 到 HA5
从 HA6 到 HA2
MAC 帧
MAC 帧
MAC 帧

IP 与 MAC地址

ping 166.111.4.100

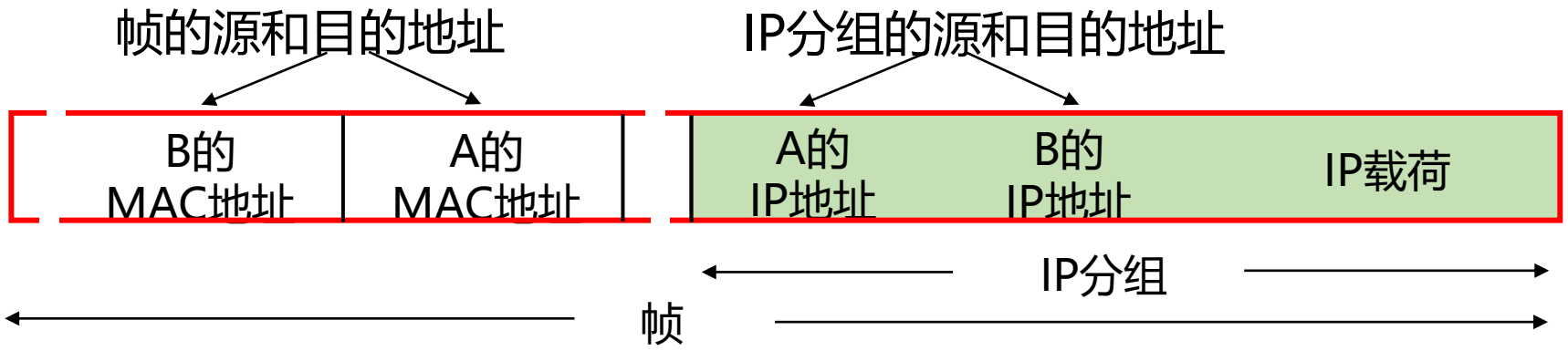
- 每个路由器已配置静态路由
- MAC头(硬件地址)+IP头(IP地址)

166.111.4.100

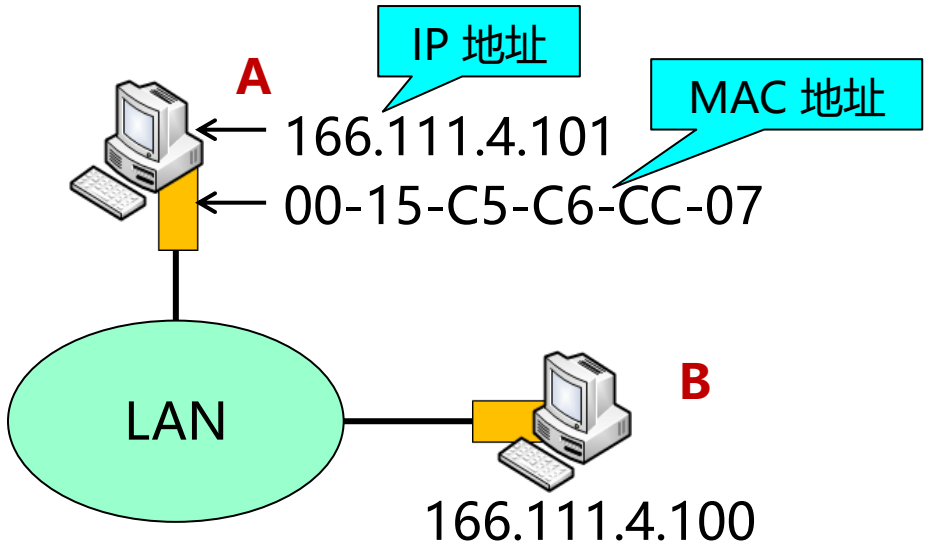


转发过程如何获取**下一跳**的MAC地址?

ARP地址解析协议



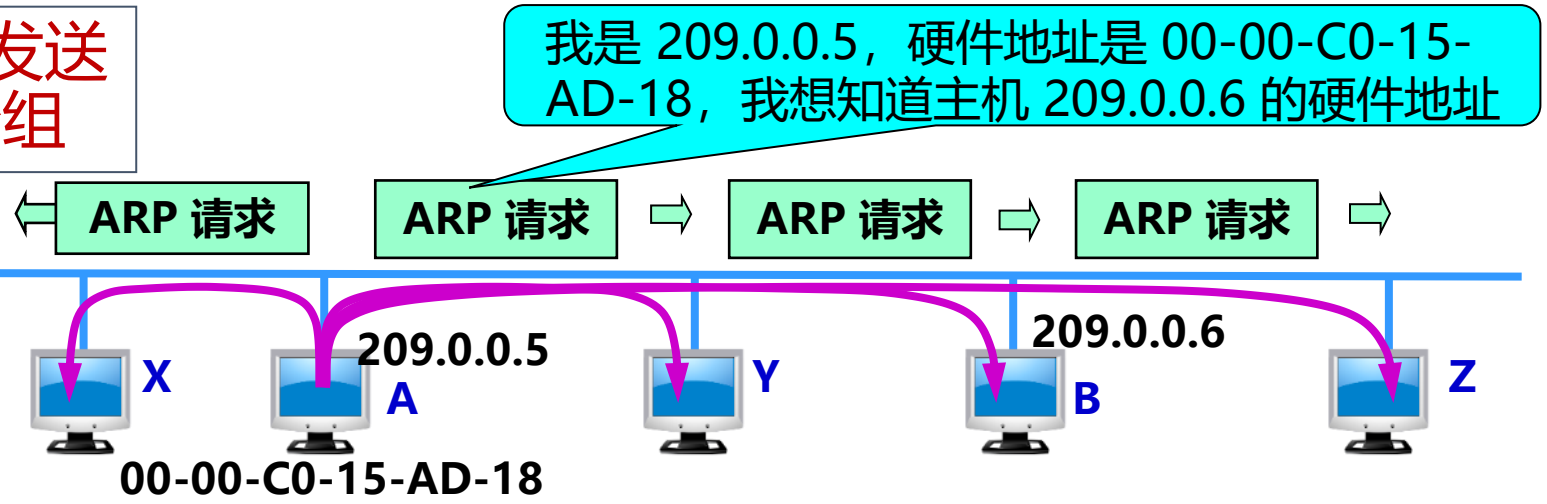
- IP数据包转发：从主机A到主机B
  - 检查目的IP地址的网络号部分
  - 确定主机B与主机A属相同IP网络
  - 将IP数据包封装到链路层帧中，直接发送给主机B



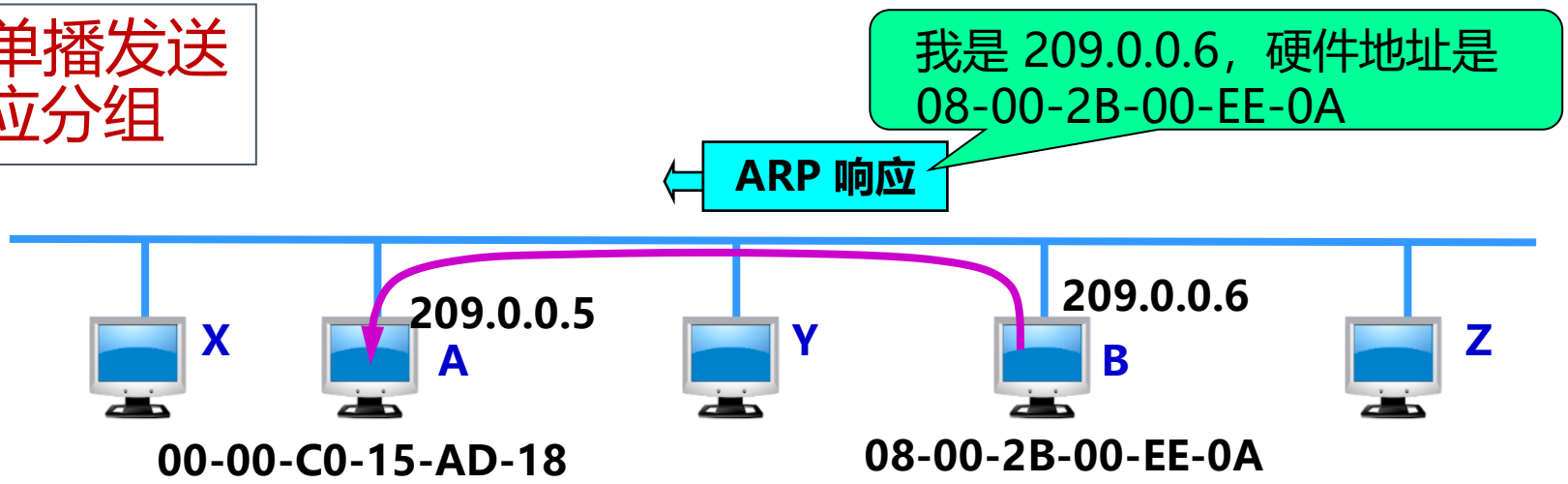
A如何获取B的MAC地址(已知B IP地址)?
设计ARP协议?

ARP协议工作过程

主机 A 广播发送 ARP 请求分组



主机B向A单播发送 ARP 响应分组



ARP地址解析协议

- 在ARP表中缓存IP地址和MAC地址的映射关系（即ARP高速缓存）

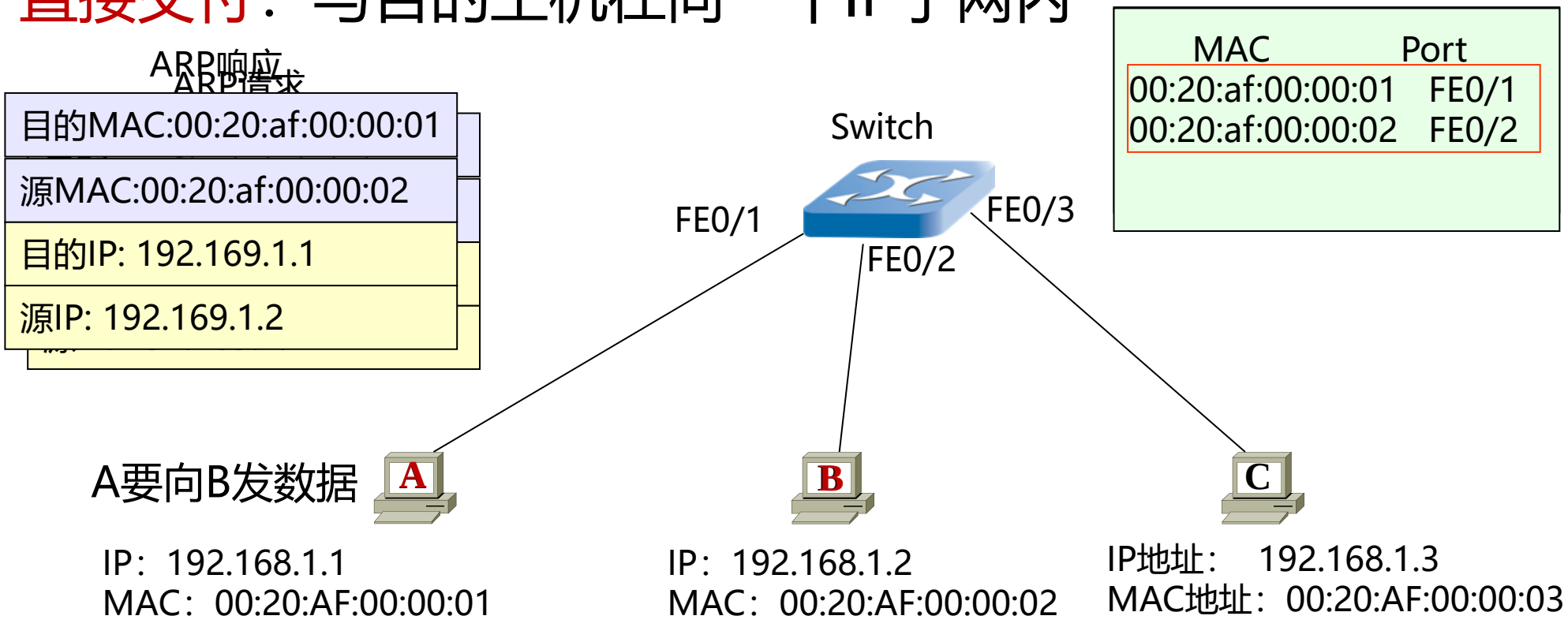
```
dgdeMacBook-Pro:~ yongcui$ arp -all
Neighbor                Linklayer Address  Expire(0)
192.168.3.1             90:17:c8:0:99:99   2m18s
192.168.3.255           ff:ff:ff:ff:ff:ff  (none)
224.0.0.251             1:0:5e:0:0:fb      (none)
239.255.255.250         1:0:5e:7f:ff:fa    (none)
```

何时查询ARP表？不命中怎么办？

思考：添加、删除的优化策略？

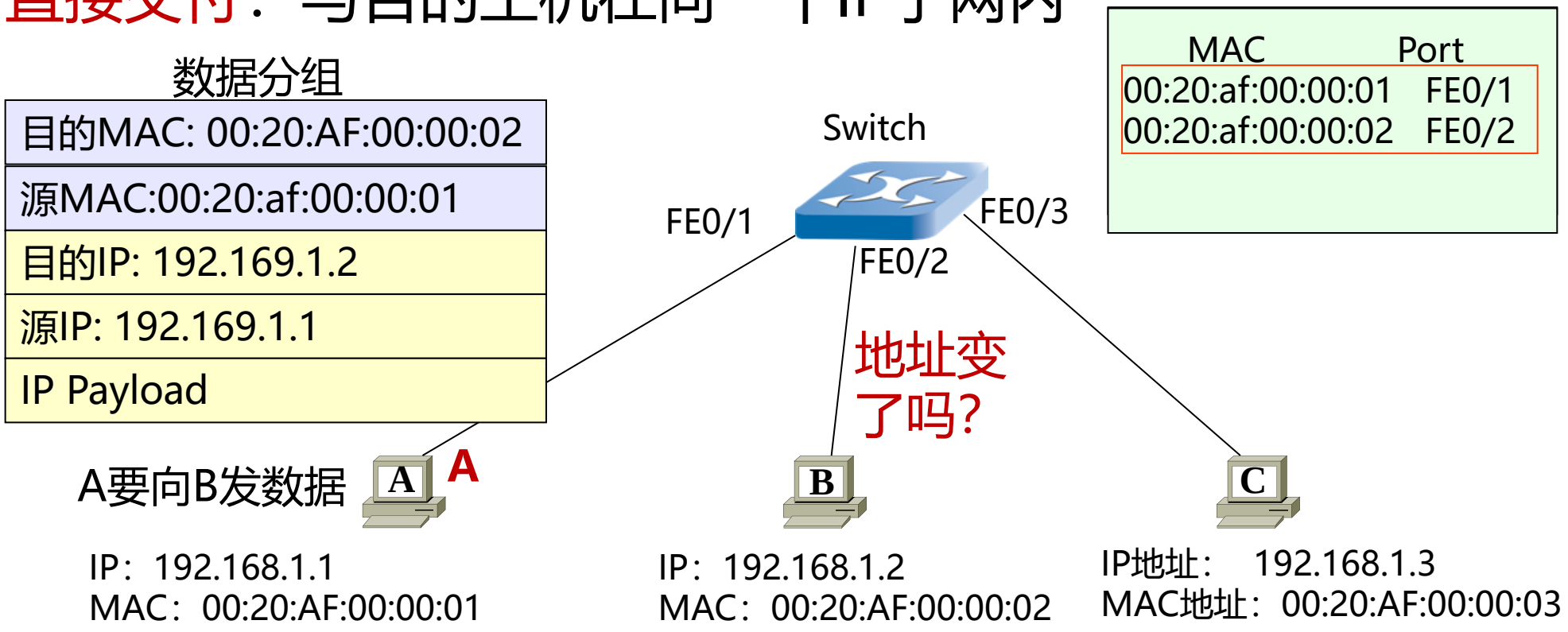
IP包转发

- 如何获取填写目的MAC地址?
- 直接交付：与目的主机在同一个IP子网内



IP包转发

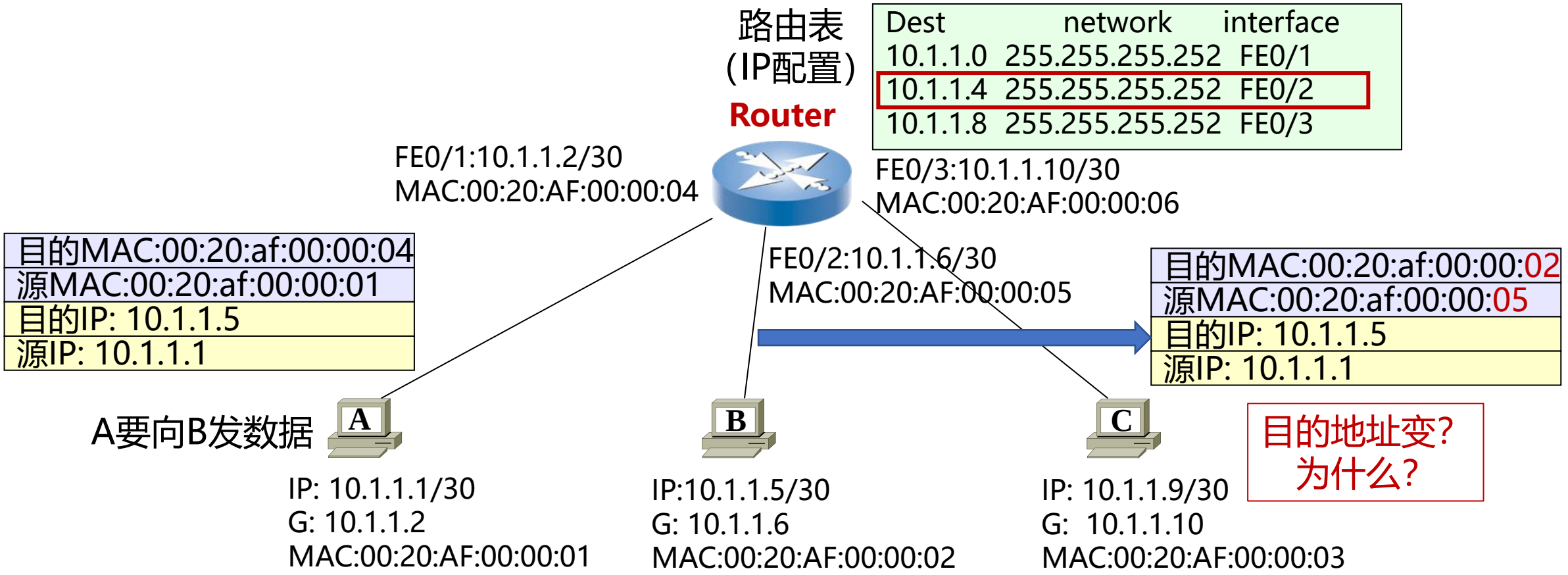
- 如何获取填写目的MAC地址?
- 直接交付：与目的主机在同一个IP子网内



直接交付如何填写目的地址?

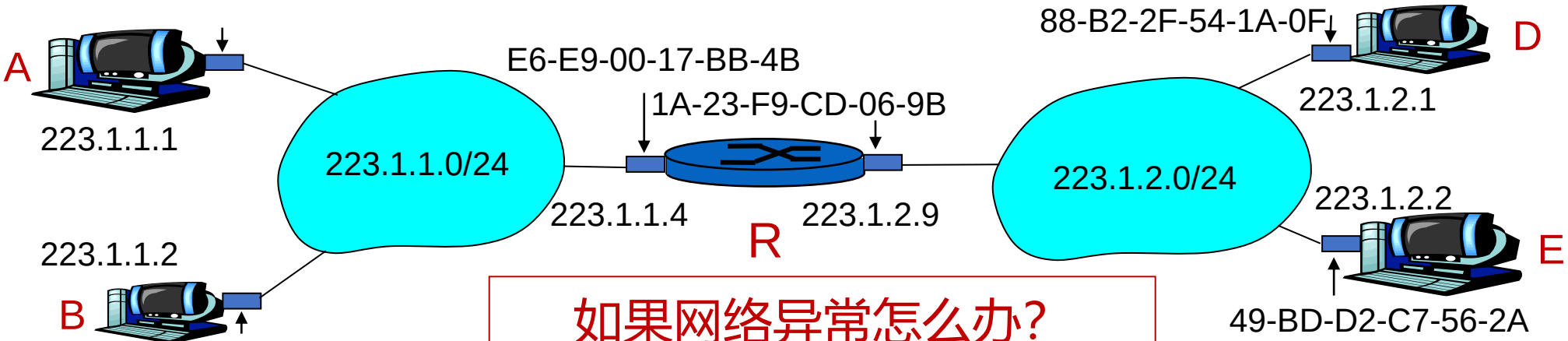
IP包转发

- 间接交付：与目的主机不在同一个IP子网内



路由到另一个局域网

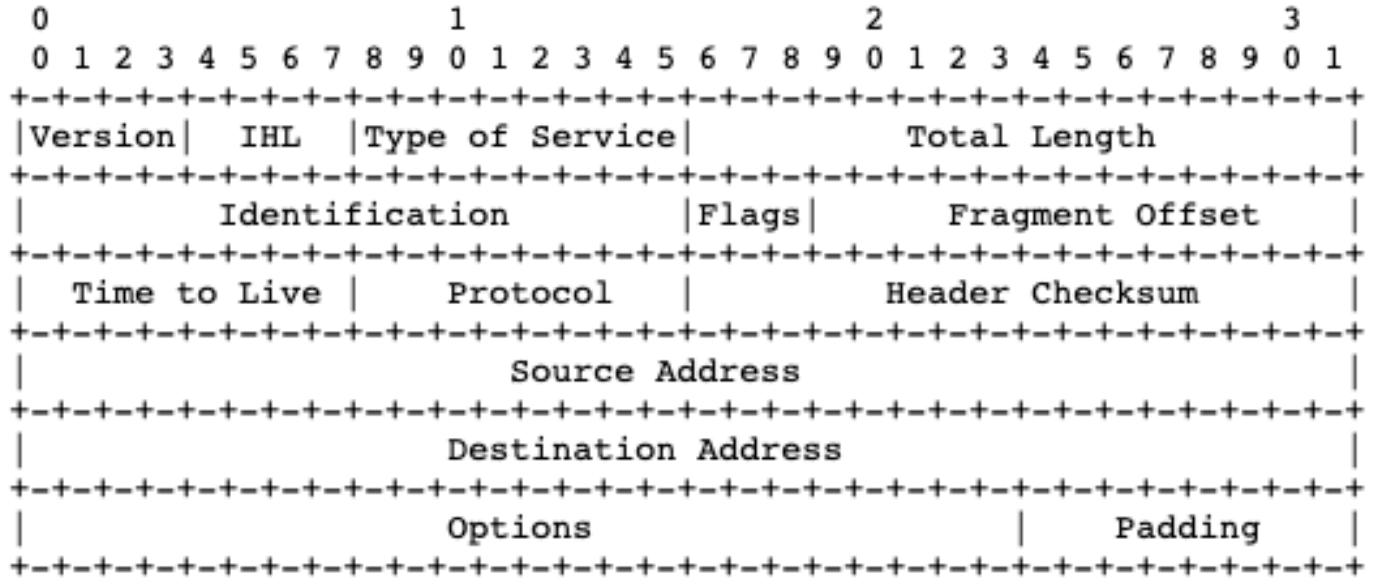
1. 主机A接入网络（希望发数据给E）
2. 申请IP地址：使用DHCP获取动态IP地址
3. A创建IP数据包：源为A、目的为E
4. 主机A查找路由表：找到路由器R的IP地址223.1.1.4
5. A获得R的MAC地址：根据R的IP地址223.1.1.4，使用ARP协议获得MAC地址
6. A创建数据帧：目的地址为R的MAC地址
7. A封装数据帧：封装A到E的IP数据包
8. A发送数据帧：A发送后，R接收数据帧



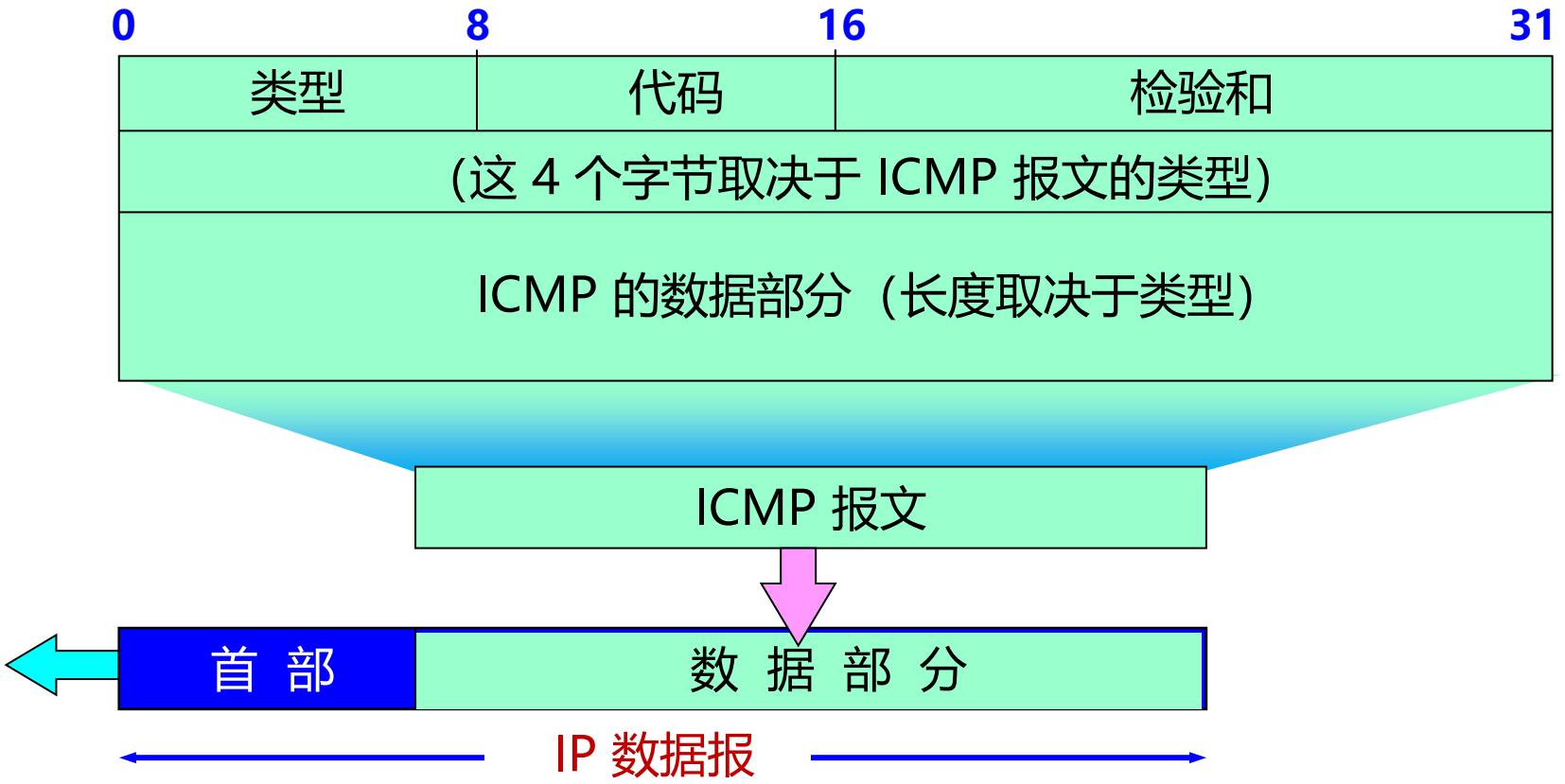
如果网络异常怎么办？
手动检查每个中间设备？

ICMP协议

- ICMP: 互联网控制报文协议
  - ICMP 允许主机或路由器报告差错情况和提供有关异常情况的报告
  - 由主机和路由器用于网络层信息的通信
  - ICMP 报文携带在IP 数据报中： IP上层协议号为1
- ICMP报文类型
  - ICMP 差错报告报文
    - 终点不可达：不可达主机、不可达网络，无效端口、协议
  - ICMP 询问报文
    - 回送请求/回答 (ping使用)



ICMP 报文格式



- ICMP报文的前 4 个字节包含格式统一的三个字段：类型、代码、检验和
- 相邻的后四个字节内容与ICMP的报文类型有关

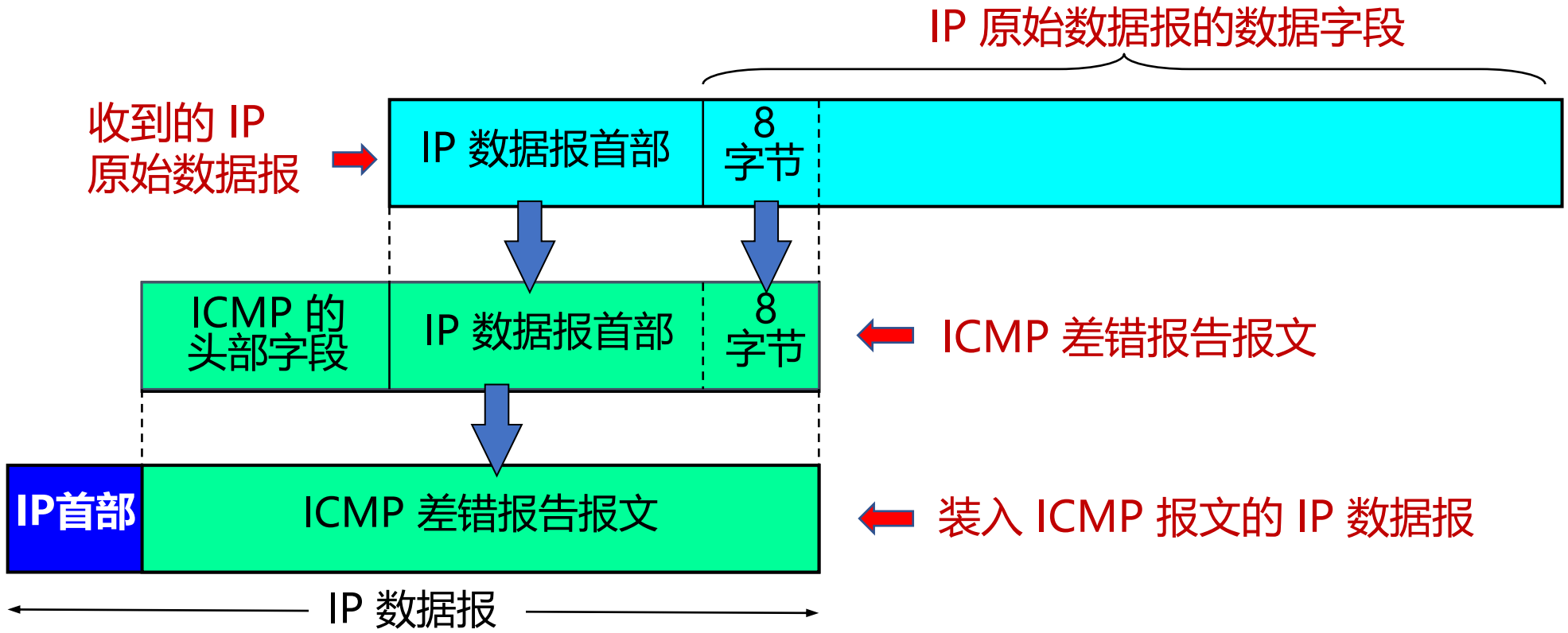
ICMP报文类型及功能

ICMP报文类型	类型值	功能描述
差错报告报文	3	终点不可达
	5	改变路由(Redirect)
	11	时间超时
	12	参数问题
询问报文	8或0	回送(Echo)请求或应答
	13或14	时间戳(Timestamp)请求或回答

类型	代码	功能描述
0	0	回送应答 (ping)
3	0	目的网络不可达
3	1	目的主机不可达
3	2	目的协议不可达
3	3	目的端口不可达
3	6	目的网络未知
3	7	目的主机未知
8	0	回送请求 (ping)
9	0	路由通告
10	0	路由发现
11	0	TTL过期
12	0	坏的IP首部

差错报告报文

是什么错误，哪个报文的错误？



Ping和ICMP

- PING (Packet InterNet Groper)
  - PING 用来测试两个主机之间的连通性
  - PING 使用了 ICMP 回送请求与回送回答报文

```
C:\>ping www.baidu.com

正在 Ping www.a.shifen.com [110.242.68.4] 具有 32 字节的数据:
来自 110.242.68.4 的回复: 字节=32 时间=32ms TTL=53
来自 110.242.68.4 的回复: 字节=32 时间=29ms TTL=53
来自 110.242.68.4 的回复: 字节=32 时间=29ms TTL=53
来自 110.242.68.4 的回复: 字节=32 时间=31ms TTL=53

110.242.68.4 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
往返行程的估计时间(以毫秒为单位):
    最短 = 29ms, 最长 = 32ms, 平均 = 30ms
```

连通性　　往返时延　　单向转发跳数

思考：

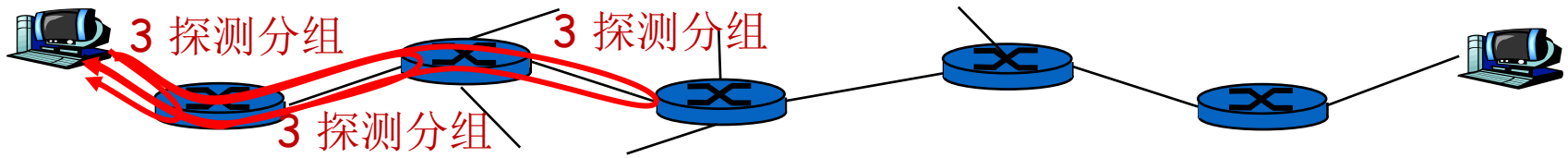
如何利用Ping命令返回的TTL值(报文剩余跳数)，来判断对方主机操作系统的类型?

默认操作系统的TTL值:

1、WINDOWS NT/2000　TTL：128
2、WINDOWS 95/98　TTL：32
3、UNIX　TTL：255
4、LINUX　TTL：64
5、WIN7　TTL：64

Traceroute和ICMP

➢ 如何知道整个路径上路由器的地址？使用TraceRT/Traceroute命令



➢源向目的地发送一系列UDP段

- 第一个 TTL=1
- 第二个 TTL=2
- ……

➢当第n个数据报到达第n个路由器

- 路由器丢弃数据报
- 向源发送ICMP报文 (类型 11, TTL过期)
- 报文的源IP地址是该路由器的IP地址

路由器会偷懒吗?

Traceroute和ICMP

C:\>tracert www.taobao.com

通过最多 30 个跃点跟踪

到 [27.211.197.171] 的路由:

1	3 ms	2 ms	4 ms	192.168.3.1
2	3 ms	6 ms	3 ms	SMBSHARE [192.168.1.1]
3	6 ms	9 ms	5 ms	27.215.136.1
4	6 ms	11 ms	7 ms	61.162.199.89
5	7 ms	18 ms	21 ms	61.162.199.9
6	23 ms	29 ms	22 ms	61.156.223.69
7	28 ms	31 ms	20 ms	112.230.160.54
8	19 ms	27 ms	36 ms	60.208.64.230
9	15 ms	15 ms	16 ms	119.164.254.86
10	19 ms	13 ms	13 ms	27.211.197.171

- 当源收到ICMP报文，计算RTT
- Tracert针对同一RTT值执行上述过程3次

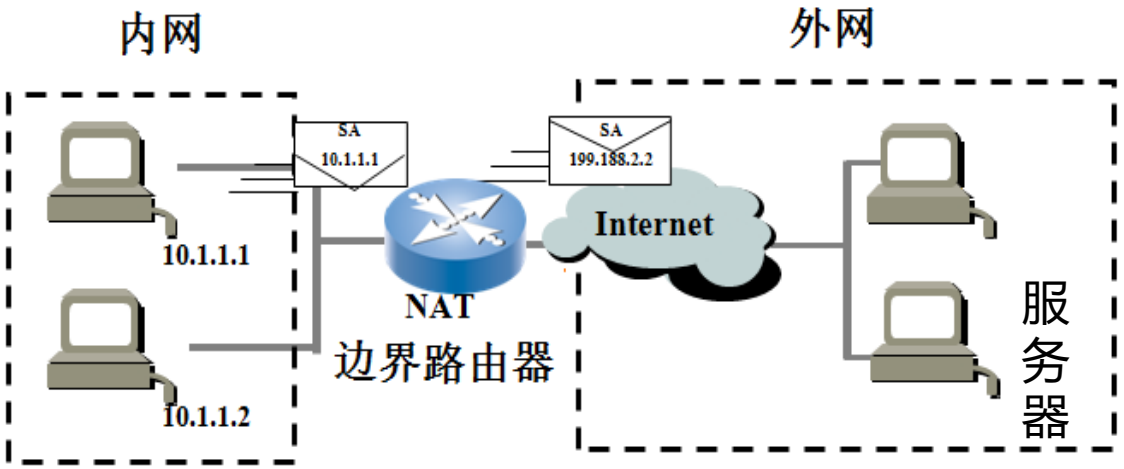
停止条件

- UDP段最终到达目的地主机
- 目的地返回ICMP 分组
- 当源得到该ICMP，停止

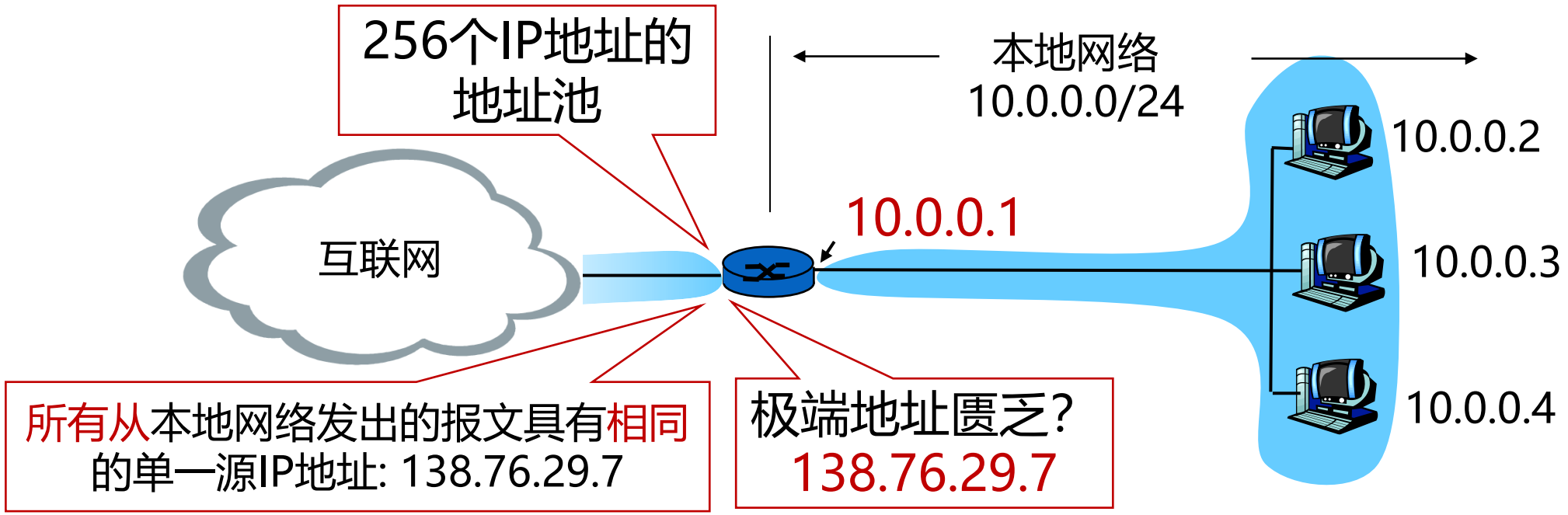
自己建个网络感受一下？
你有IP吗☹

网络地址转换

- 自己建网络自己用：私有IP地址
  - A类地址：10.0.0.0--10.255.255.255
  - B类地址：172.16.0.0--172.31.255.555
  - C类地址：192.168.0.0--192.168.255.255
- IP地址不够用怎么办？
  - 能否设计地址共享/复用机制？
- 网络地址转换(NAT)
  - 用于解决IPv4地址不足的问题，是一种将私有（保留）地址转化为公有IP地址的转换技术
  - 在不同时间复用，还是同时使用？
  - 谁出去，回来给谁？



NAT工作机制

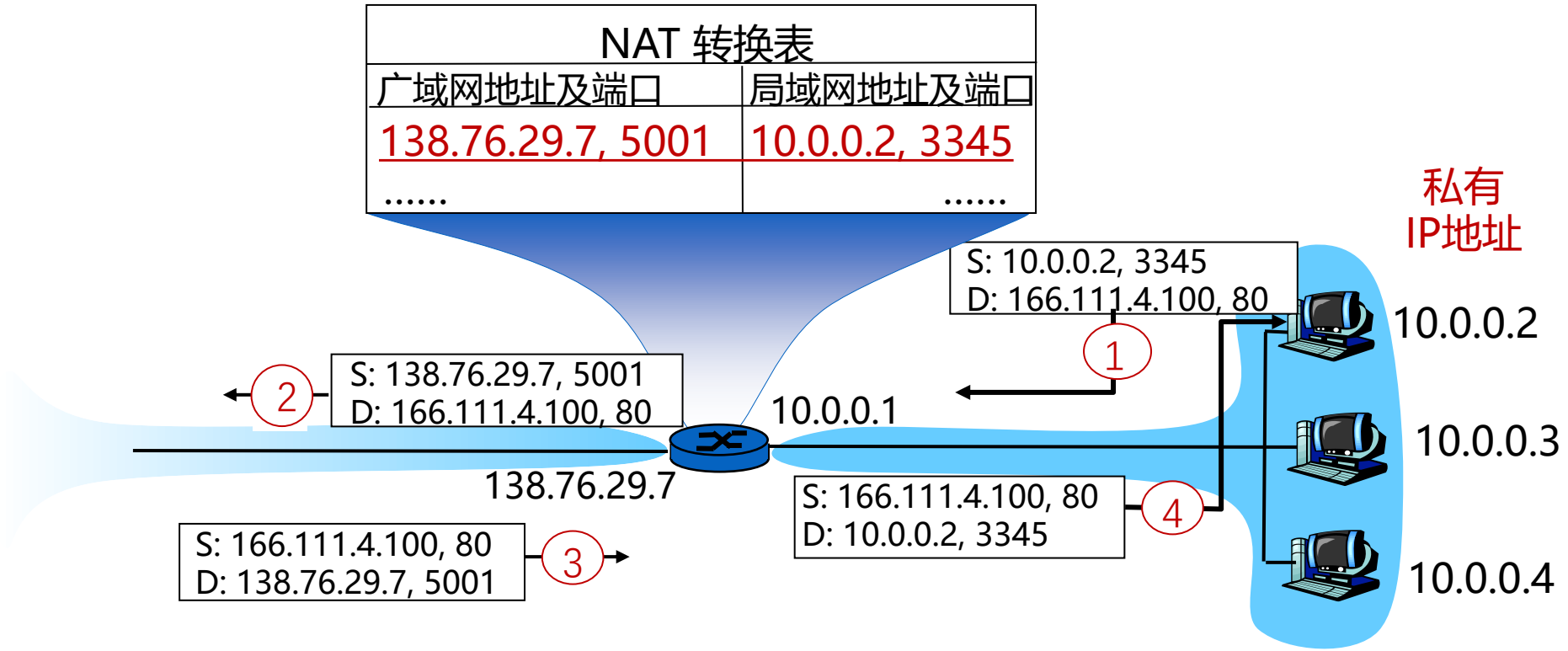


不同的源端口号

传输层TCP/UDP拥有16-bit 端口号字段
所以WAN侧一个地址可支持>60,000个并发连接

思考：同一主机不同应用，或者不同主机的同一端口，NAT转换如何处理?

NAT工作机制



什么时候建立NAT表?
自动建立还是手动建立?

网络地址转换

- 有效提升地址复用率，解决IPv4地址不足的问题
  - 一个WAN侧地址可支持>60,000个并行连接
  - NAT根据不同的IP上层协议进行NAT表项管理
  - TCP，UDP等

好吗?

- NAT的优势
  - 节省合法地址，减少地址冲突
  - 灵活连接Internet
  - 保护局域网的私密性

NAT是协议吗?
大家喜欢吗?

- 问题或缺点
  - 违反了IP的结构模型，路由器处理传输层协议
  - 违反了端到端的原则
  - 违反了最基本的协议分层规则
  - 不能处理IP报头加密
  - 新型网络应用的设计者必须要考虑 NAT场景，如 P2P应用程序

网络层典型协议和技术-小结

- DHCP协议
  - 动态主机配置协议，C/S模式动态分配IP地址（广播 | 单播）
- ARP协议
  - ARP：给定IP地址，如何获得对应的MAC?
  - 多跳网络中，IP 数据报中IP地址不改变，Mac帧中的硬件地址逐跳改变
  - 直接交付和间接交付的区别?
- ICMP协议
  - IP层报告差错情况
  - 两种检错工具：Ping、Traceroute
- NAT网络地址转换
  - 想用私人网络怎么办？IP地址不够用怎么办?
  - NAT转换表：内网(IP，端口)<->外网(IP，端口)

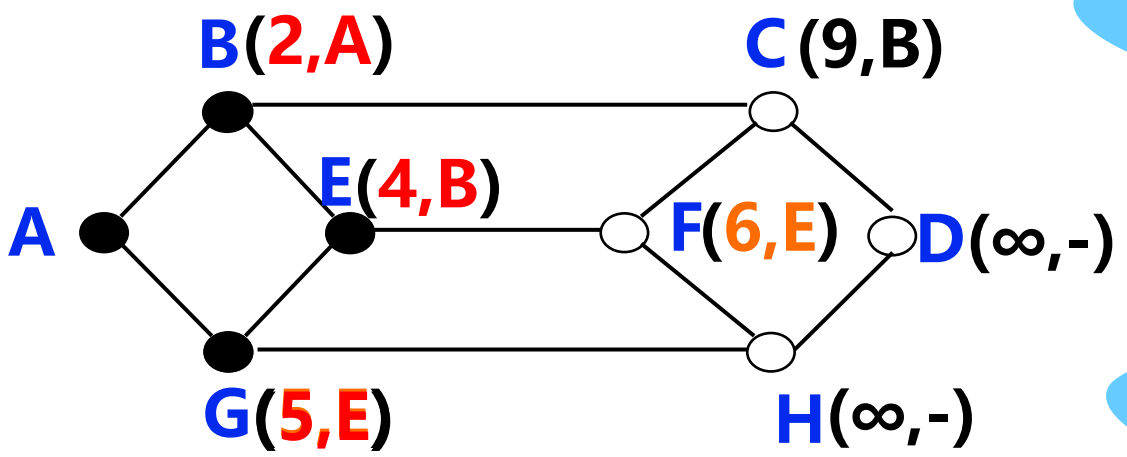
家用路由器有这些功能吗?

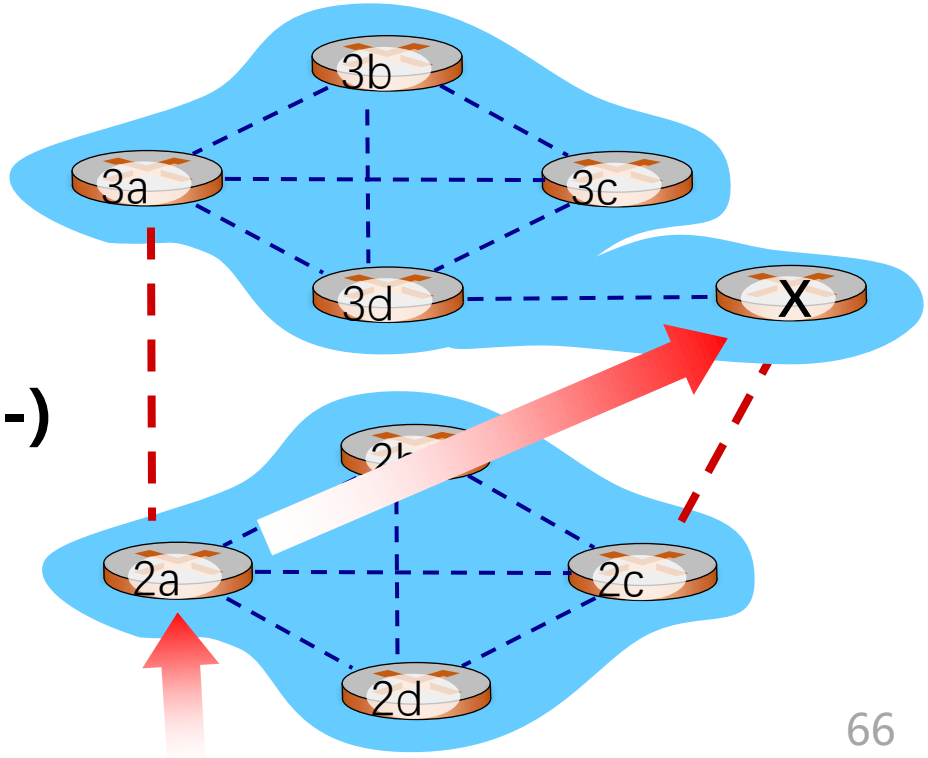
总结

- 网络层服务
  - 实现多跳网络传输：控制面编址&路由、数据面转发
- IPv4 与编址
  - 地址分配问题：分类地址->CIDR，路由聚合+最长前缀匹配
  - 其他功能：数据包分片、校验和、TTL
  - 特殊的IP地址：单播、广播、组播、任播
- 网络层典型协议和技术
  - IP协议是基础，DHCP、ARP、NAT、ICMP是辅助
  - DHCP：动态IP分配；ARP：IP-MAC转换
  - ICMP：控制报文协议；NAT：网络地址转换

下周预告

- 漂洋过海来看你：大规模?
  - 手动配置全球网络（静态路由表）？
  - 可扩展性：如何支持百万节点网络动态路由?
  - 动态性：链路失效、节点新增、设备故障、韧性……
  - OSPF->RIP->BGP，一起发明路由协议





作业

- 《Computer Networks-5th Edition》章节末习题
  - CHAPTER 5：25（IP分片与选项），26（分类IP地址），28（子网划分），33（最长前缀匹配），34（NAT），36（IP分片）
- 截止时间：下下周三晚11:59，提交网络学堂

致谢社区本章贡献者

贡献者姓名	单 位	贡献内容
陈文龙	首都师范大学	本章统稿 5.5 5.9(IPv6协议)
吴黎兵	武汉大学	5.6 5.7
谢晓燕	西安邮电大学	5.8
邹莹	仲恺农业工程学院	5.4.1 5.4.2 5.4.4
李旭宏	枣庄学院	5.1.2 5.1.3 5.2.3 5.2.4 5.2.6 5.4.3
曲大鹏	辽宁大学	5.3.1 5.3.2
方诗虹	西南民族大学	5.1.4 5.3.5 5.3.6 5.3.7 5.3.8
舒挺	浙江理工大学	5.1.1
白云莉	内蒙古农业大学	5.3.3 5.3.4
余琨	荆楚理工学院	5.2.1 5.2.2 5.2.5
李振斌	华为技术有限公司	5.9(SRv6)

致谢社区本章贡献者

陈文龙

首都师范大学

5.路由器工作原理
9.IPv6技术

吴黎兵

武汉大学

6.拥塞控制算法
7.服务质量

谢晓燕

西安邮电大学

8.三层交换和VPN

邹莹

仲恺农业工程学院

4.Internet路由协议

李旭宏

枣庄学院

1.网络层服务
2.Internet网际协议
4.Internet路由协议

《计算机网络：自顶向下方法》(原书第7版)，库罗斯 罗斯，机械工业出版社，2018年06月
《计算机网络（第5版）》，Tanenbaum & Wetherall，清华大学出版社，2012年3月
《计算机网络（第7版）》，谢希仁，电子工业出版社，2017年01月
《计算机网络教程（第6版）》，吴功宜，电子工业出版社，2018年03月
《计算机网络（第3版）》，徐敬东、张建忠，清华大学出版社，2013年6月1日

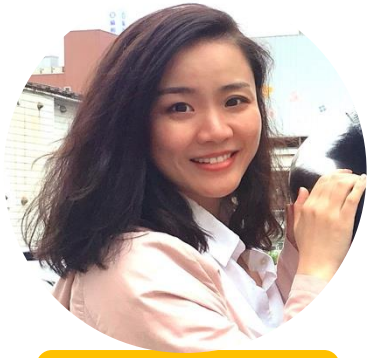
特别致谢：
部分内容取材于此

致谢社区本章贡献者

曲大鹏

辽宁大学

3.路由算法



方诗虹

西南民族大学

1.网络层服务
3.路由算法

舒挺

浙江理工大学

1.网络层服务

白云莉

内蒙古农业大学

3.路由算法

余琨

荆楚理工学院

2.Internet网际协议

李振斌

华为技术公司

9.IPv6技术

《计算机网络：自顶向下方法》(原书第7版)，库罗斯 罗斯，机械工业出版社，2018年06月
《计算机网络（第5版）》，Tanenbaum & Wetherall，清华大学出版社，2012年3月
《计算机网络（第7版）》，谢希仁，电子工业出版社，2017年01月
《计算机网络教程（第6版）》，吴功宜，电子工业出版社，2018年03月
《计算机网络（第3版）》，徐敬东、张建忠，清华大学出版社，2013年6月1日

特别致谢：
部分内容取材于此