第 4 章 网络层

第 4 章 网络层

- 4.1 网络层提供的两种服务
- 4.2 网际协议 IP
- 4.3 划分子网和构造超网
- 4.4 网际控制报文协议 ICMP
- 4.5 互联网的路由选择协议
- 4.6 IPv6
- 4.7 IP 多播
- 4.8 虚拟专用网 VPN 和网络地址转换 NAT
- 4.9 多协议标记交换 MPLS

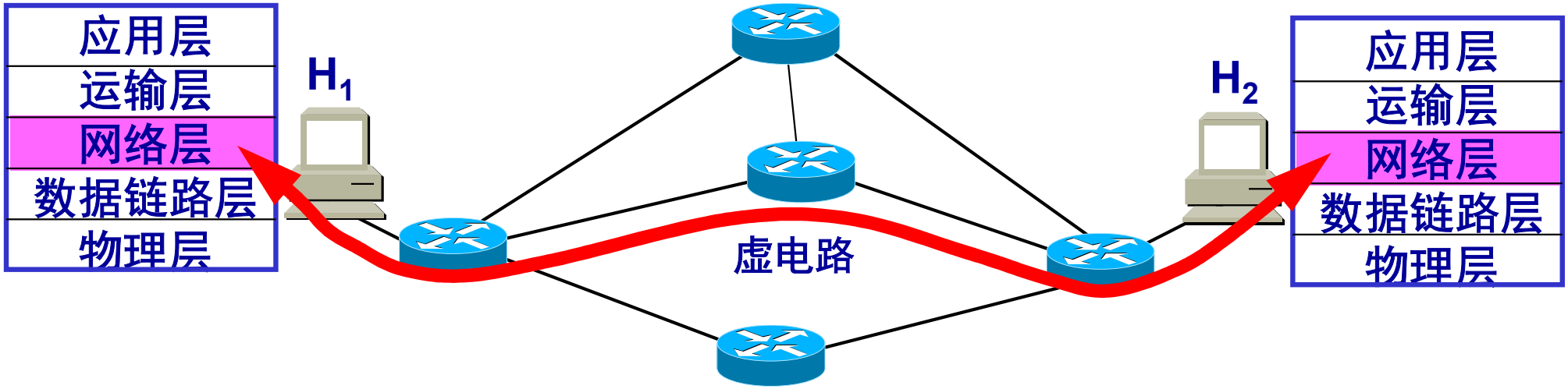
4.1 网络层提供的两种服务

- 在计算机网络领域，网络层应该向运输层提供怎样的服务（“面向连接”还是“无连接”）曾引起了长期的争论。
- 争论焦点的实质就是：在计算机通信中，可靠交付应当由谁来负责？是网络还是端系统？

一种观点：让网络负责可靠交付

- 这种观点认为，应借助于电信网的成功经验，让网络负责可靠交付，计算机网络应模仿电信网络，使用面向连接的通信方式。
- 通信之前先建立虚电路 (Virtual Circuit)，以保证双方通信所需的一切网络资源。
- 如果再使用可靠传输的网络协议，就可使所发送的分组无差错按序到达终点，不丢失、不重复。

虚电路服务



H_1 发送给 H_2 的所有分组都沿着同一条虚电路传送

虚电路是逻辑连接

- 虚电路表示这只是一条逻辑上的连接，分组都沿着这条逻辑连接按照存储转发方式传送，而并不是真正建立了一条物理连接。
- 请注意，电路交换的电话通信是先建立了一条真正的连接。
- 因此分组交换的虚连接和电路交换的连接只是类似，但并不完全一样。

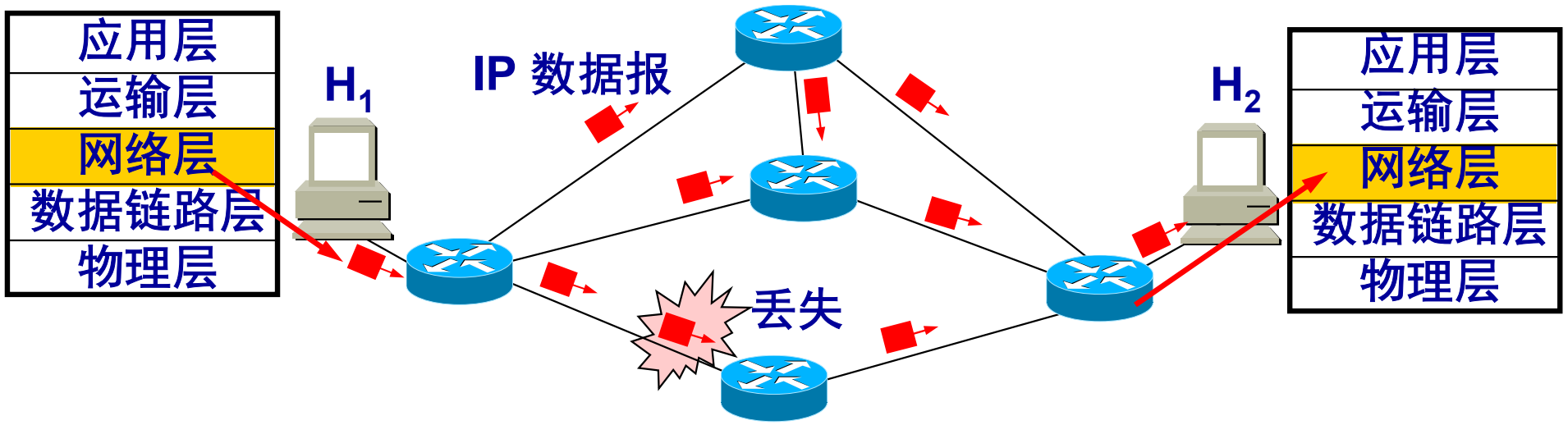
另一种观点：网络提供数据报服务

- 互联网的先驱者提出了一种崭新的网络设计思路。
- 网络层向上只提供简单灵活的、无连接的、尽最大努力交付的数据报服务。
- 网络在发送分组时不需要先建立连接。每一个分组（即 IP 数据报）独立发送，与其前后的分组无关（不进行编号）。
- 网络层不提供服务质量的承诺。即所传送的分组可能出错、丢失、重复和失序（不按序到达终点），当然也不保证分组传送的时限。

尽最大努力交付

- 由于传输网络不提供端到端的可靠传输服务，这就使网络中的路由器可以做得比较简单，而且价格低廉（与电信网的交换机相比较）。
- 如果主机（即端系统）中的进程之间的通信需要是可靠的，那么就由网络的主机中的运输层负责可靠交付（包括差错处理、流量控制等）。
- 采用这种设计思路的好处是：网络的造价大大降低，运行方式灵活，能够适应多种应用。
- 互连网能够发展到今日的规模，充分证明了当初采用这种设计思路的正确性。

数据报服务



H_1 发送给 H_2 的分组可能沿着不同路径传送

虚电路服务与数据报服务的对比

对比的方面	虚电路服务	数据报服务
思路	可靠通信应当由网络来保证	可靠通信应当由用户主机来保证
连接的建立	必须有	不需要
终点地址	仅在连接建立阶段使用，每个分组使用短的虚电路号	每个分组都有终点的完整地址
分组的转发	属于同一条虚电路的分组均按照同一路由进行转发	每个分组独立选择路由进行转发
当结点出故障时	所有通过出故障的结点的虚电路均不能工作	出故障的结点可能会丢失分组，一些路由可能会发生变化
分组的顺序	总是按发送顺序到达终点	到达终点时不一定按发送顺序
端到端的差错处理和流量控制	可以由网络负责，也可以由用户主机负责	由用户主机负责

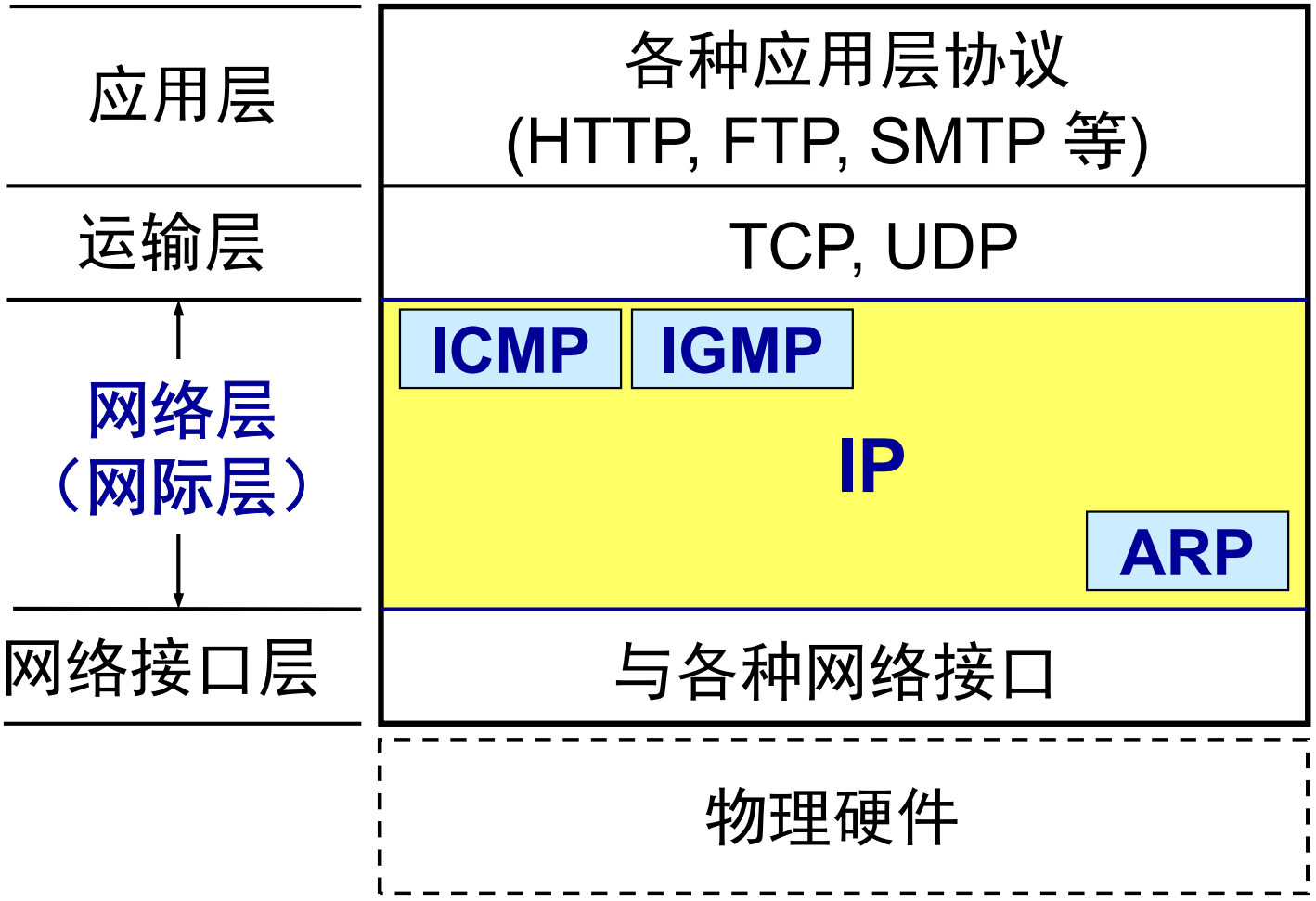
4.2 网际协议 IP

- 4.2.1 虚拟互连网络
- 4.2.2 分类的 IP 地址
- 4.2.3 IP 地址与硬件地址
- 4.2.4 地址解析协议 ARP
- 4.2.5 IP 数据报的格式
- 4.2.6 IP 层转发分组的流程

4.2 网际协议 IP

- 网际协议 IP 是 TCP/IP 体系中两个最主要的协议之一。
- 与 IP 协议配套使用的还有三个协议：
  - 地址解析协议 ARP
    (Address Resolution Protocol)
  - 网际控制报文协议 ICMP
    (Internet Control Message Protocol)
  - 网际组管理协议 IGMP
    (Internet Group Management Protocol)

网际层的 IP 协议及配套协议



4.2.1 虚拟互连网络

- 将网络互连并能够互相通信，会遇到许多问题需要解决，如：
  - 不同的寻址方案
  - 不同的最大分组长度
  - 不同的网络接入机制
  - 不同的超时控制
  - 不同的差错恢复方法
  - 不同的状态报告方法
  - 不同的路由选择技术
  - 不同的用户接入控制
  - 不同的服务（面向连接服务和无连接服务）
  - 不同的管理与控制方式 等

如何将异构的网络互相连接起来？

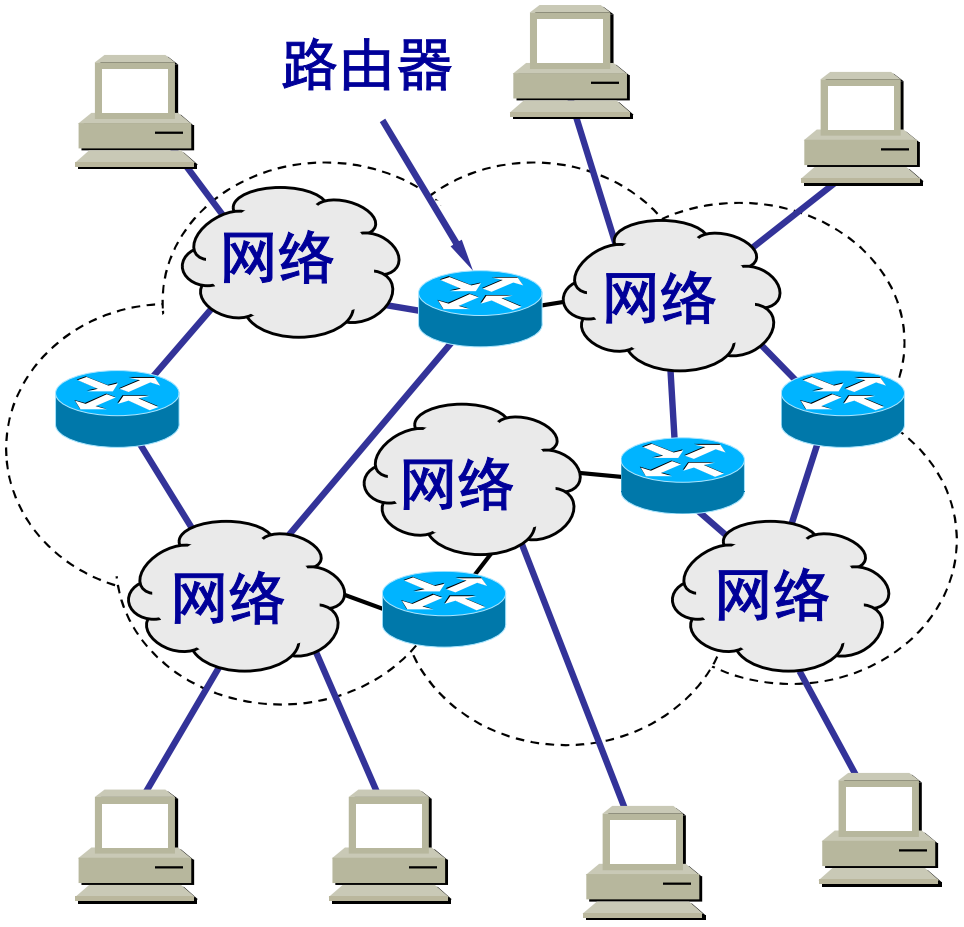
使用一些中间设备进行互连

- 将网络互相连接起来要使用一些中间设备。
- 中间设备又称为中间系统或中继 (relay)系统。
- 有以下五种不同的中间设备：
  - 物理层中继系统：转发器 (repeater)。
  - 数据链路层中继系统：网桥 或 桥接器 (bridge)或交换机（switch）。
  - 网络层中继系统：路由器 (router)。
  - 网桥和路由器的混合物：桥路器 (brouter)、三层交换机（L3-Switch）。
  - 网络层以上的中继系统：网关 (gateway)。

网络互连使用路由器

- 当中继系统是转发器或网桥时，一般并不称之为网络互连，因为这仅仅是把一个网络扩大了，而这仍然是一个网络。
- 网关由于比较复杂，目前使用得较少。
- 网络互连都是指用路由器进行网络互连和路由选择。
- 由于历史的原因，许多有关 TCP/IP 的文献将网络层使用的路由器称为网关。

互连网络与虚拟互连网络



(a) 互连网络

(b) 虚拟互连网络

IP 网的概念

虚拟互连网络的意义

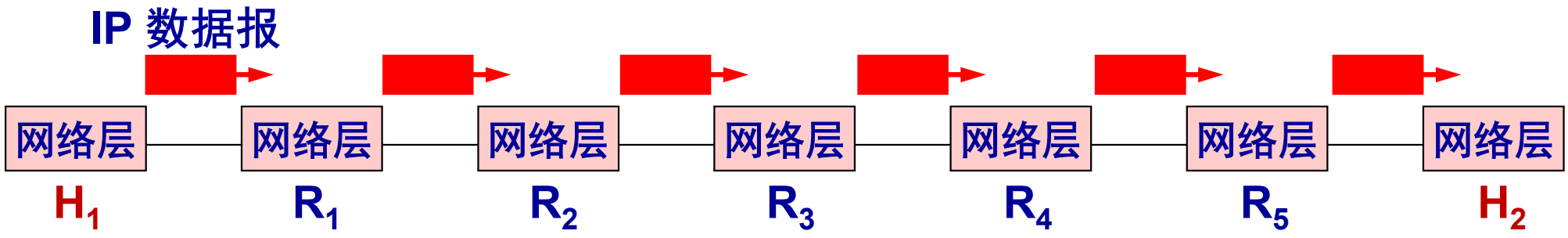
- 所谓虚拟互连网络也就是逻辑互连网络，它的意思就是互连起来的各种物理网络的异构性本来是客观存在的，但是我们利用 IP 协议就可以使这些性能各异的网络从用户看起来好像是一个统一的网络。
- 使用 IP 协议的虚拟互连网络可简称为 IP 网。
- 使用虚拟互连网络的好处是：当互联网上的主机进行通信时，就好像在一个网络上通信一样，而看不见互连的各具体的网络异构细节。
- 如果在这种覆盖全球的 IP 网的上层使用 TCP 协议，那么就是现在的互联网 (Internet)。

分组在互联网中的传送

互联网可以由多种异构网络互连组成。

从网络层看 IP 数据报的传送

- 如果我们只从网络层考虑问题，那么 IP 数据报就可以想象是在网络层中传送。



4.2.2 分类的 IP 地址

- 在 TCP/IP 体系中，IP 地址是一个最基本的概念。
- 本部分重点学习：
  - 1. IP 地址及其表示方法
  - 2. 常用的三种类别的 IP 地址

1. IP 地址及其表示方法

- 我们把整个因特网看成为一个单一的、抽象的网络。
- IP 地址就是给每个连接在互联网上的主机（或路由器）分配一个在全世界范围是唯一的 32 位的标识符。
- IP 地址现在由互联网名字和数字分配机构 ICANN (Internet Corporation for Assigned Names and Numbers)进行分配。

IP 地址的编址方法

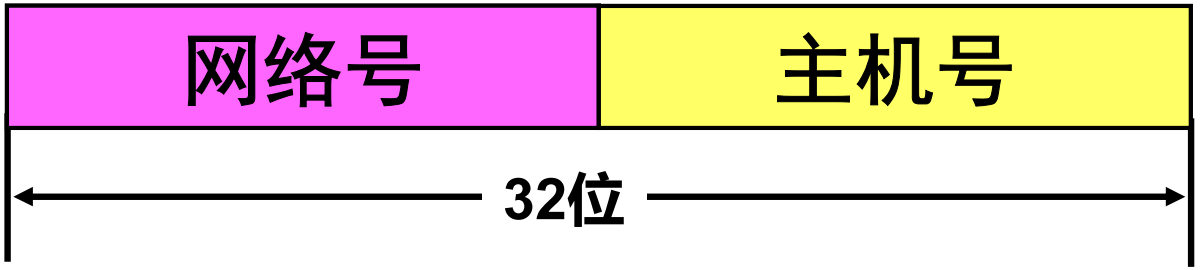
- 分类的 IP 地址。这是最基本的编址方法，在 1981 年就通过了相应的标准协议。
- 子网的划分。这是对最基本的编址方法的改进，其标准[RFC 950] 在 1985 年通过。
- 构成超网。这是比较新的无分类编址方法。1993 年提出后很快就得到推广应用。

分类 IP 地址

- 将IP地址划分为若干个固定类。
- 每一类地址都由两个固定长度的字段组成，其中一个字段是网络号 net-id，它标志主机（或路由器）所连接到的网络，而另一个字段则是主机号 host-id，它标志该主机（或路由器）。
- 主机号在它前面的网络号所指明的网络范围内必须是唯一的。
- 由此可见，一个 IP 地址在整个互联网范围内是唯一的。

分类 IP 地址

- 这种两级的 IP 地址结构如下：

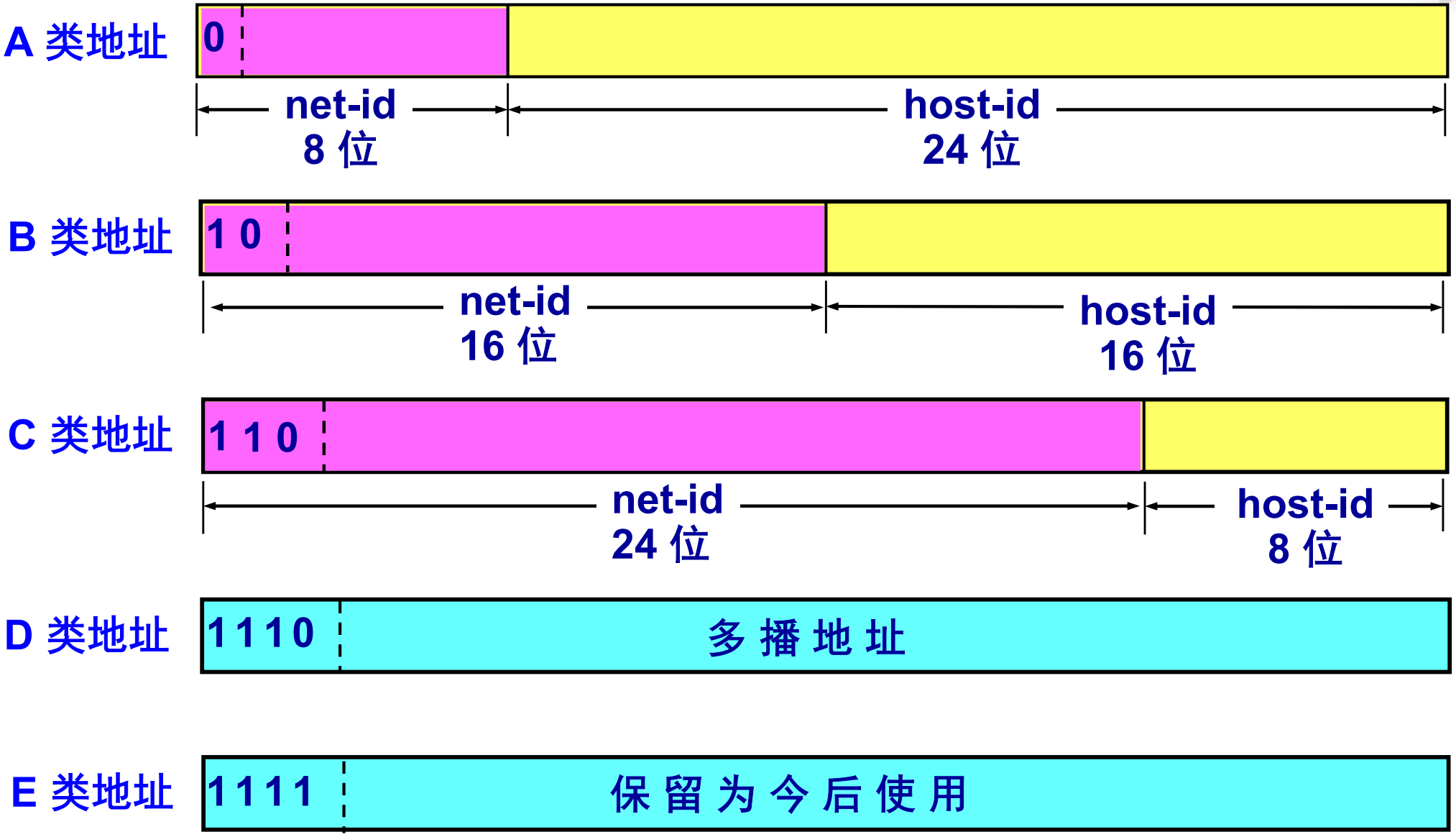


- 这种两级的 IP 地址可以记为：

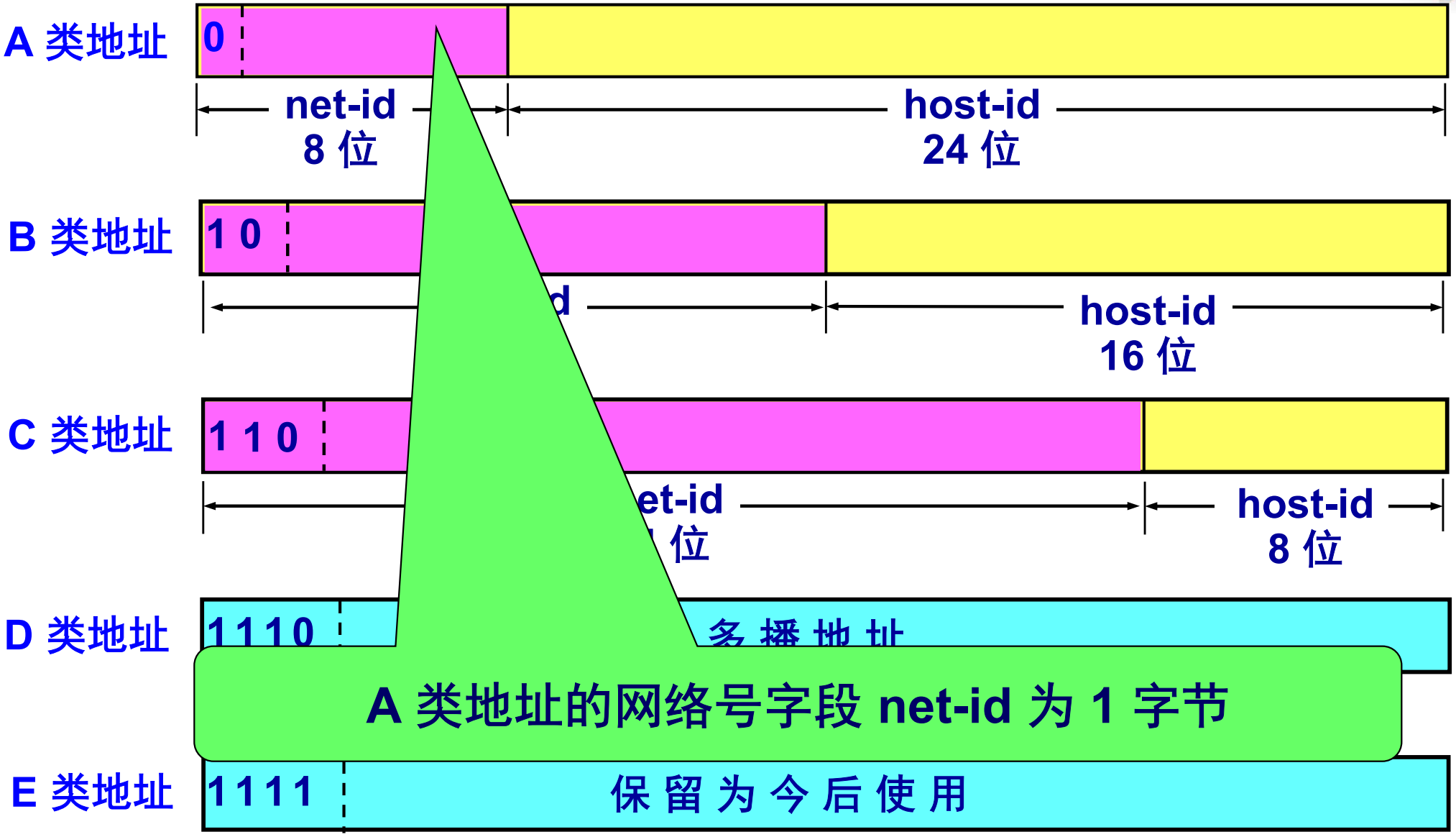
IP 地址 ::= { <网络号>, <主机号>}　　(4-1)

::= 代表“定义为”

各类 IP 地址的网络号字段和主机号字段

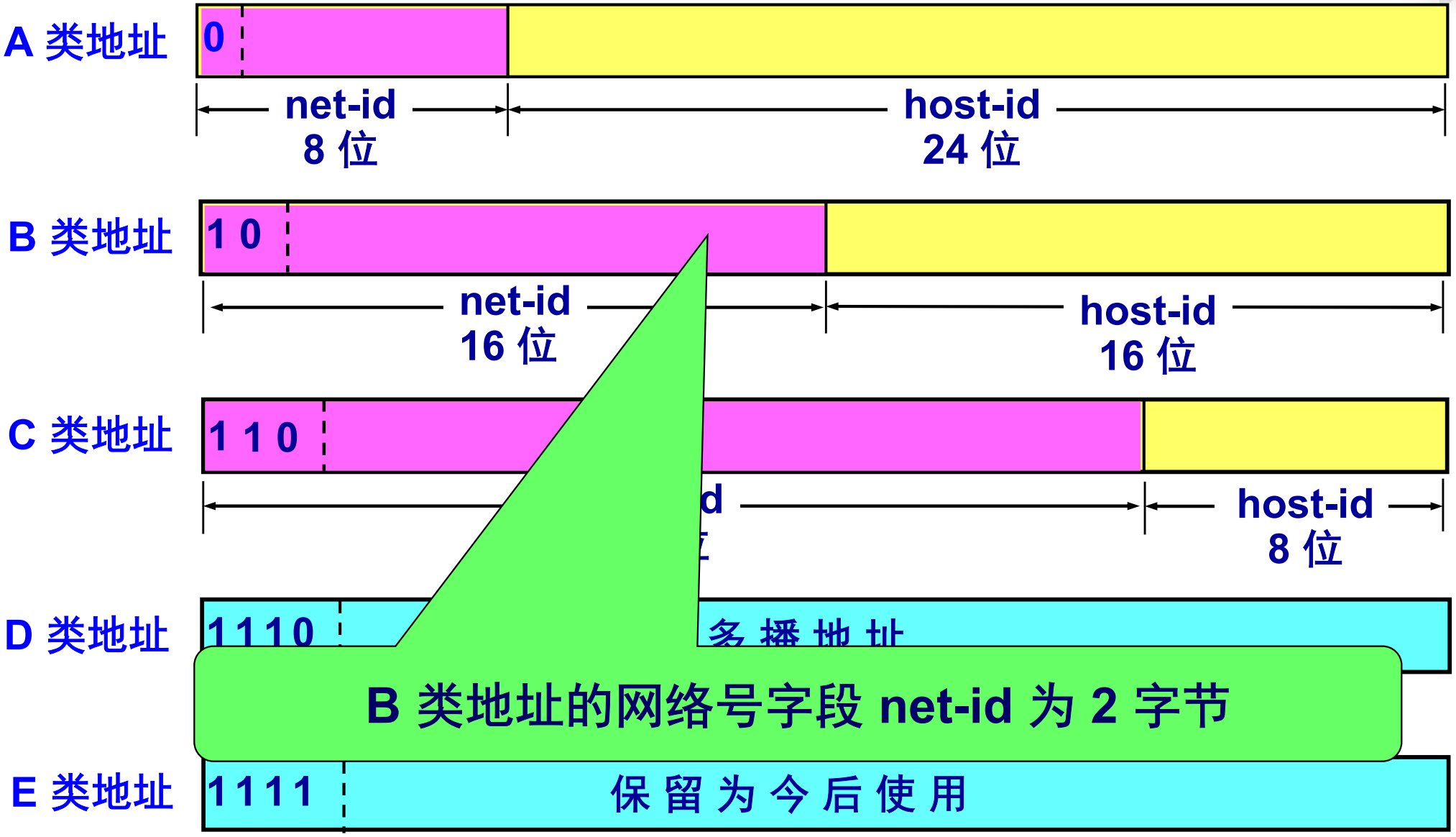


各类 IP 地址的网络号字段和主机号字段



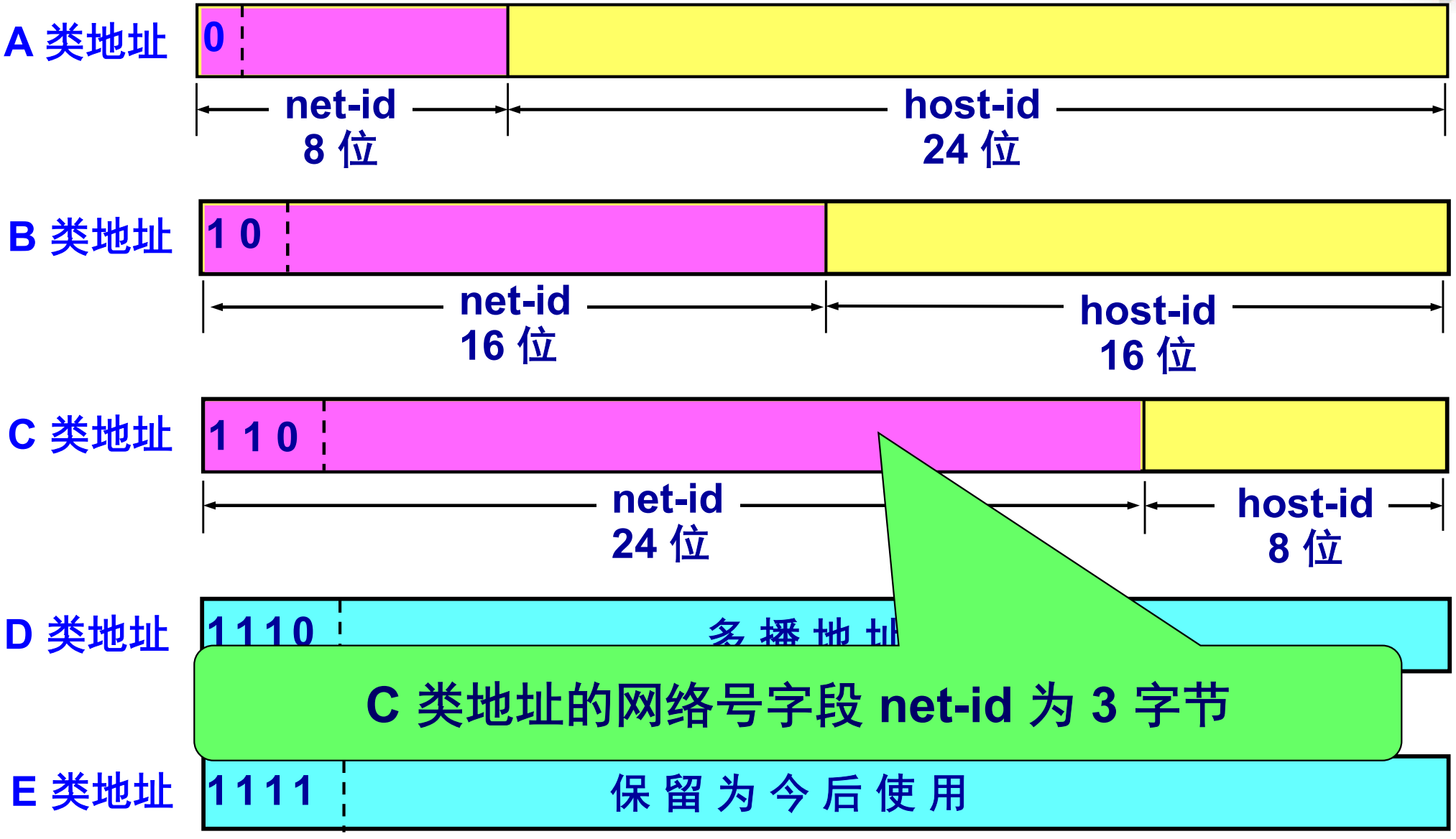
A 类地址的网络号字段 net-id 为 1 字节

各类 IP 地址的网络号字段和主机号字段

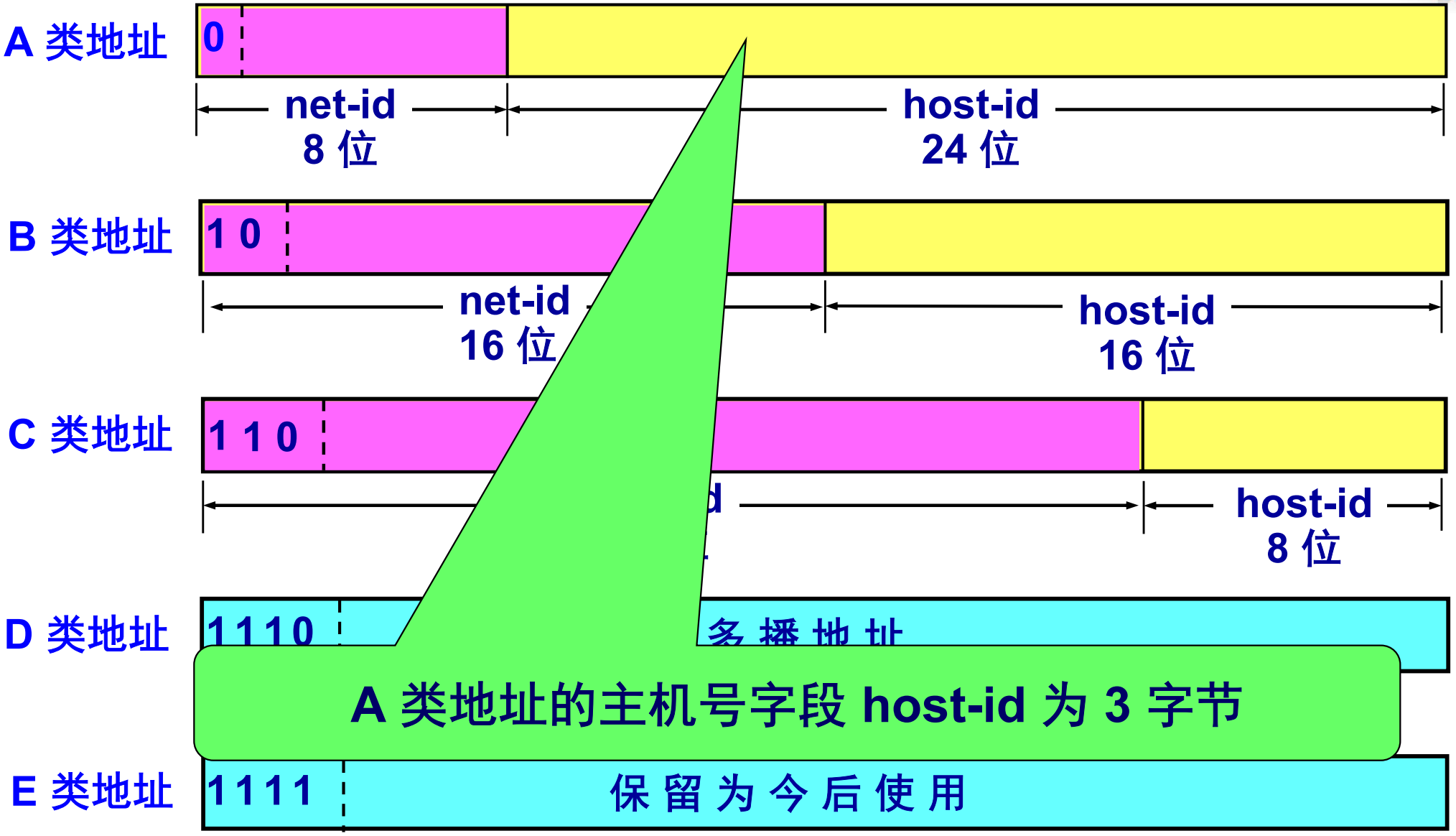


B 类地址的网络号字段 net-id 为 2 字节

各类 IP 地址的网络号字段和主机号字段

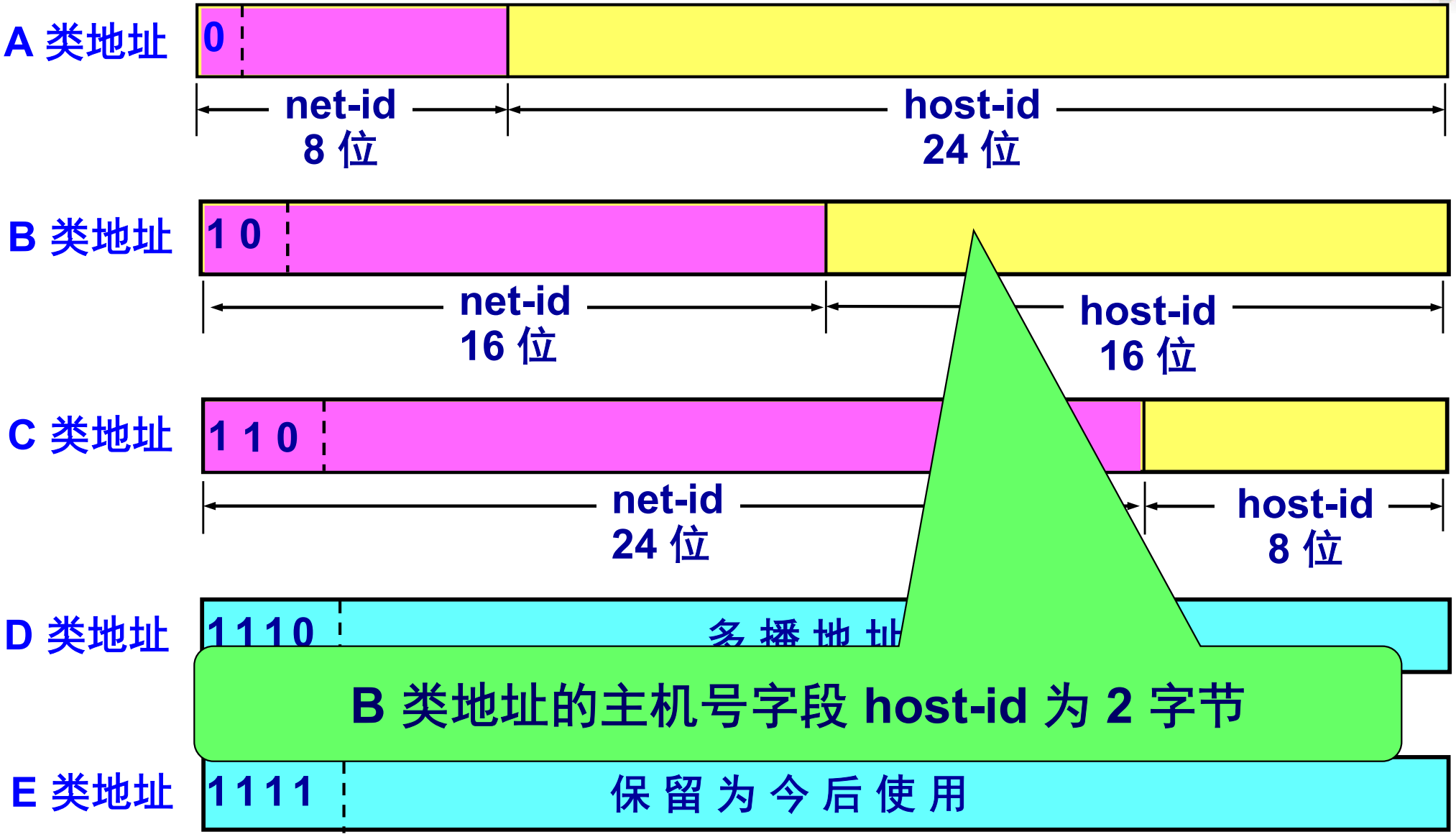


各类 IP 地址的网络号字段和主机号字段



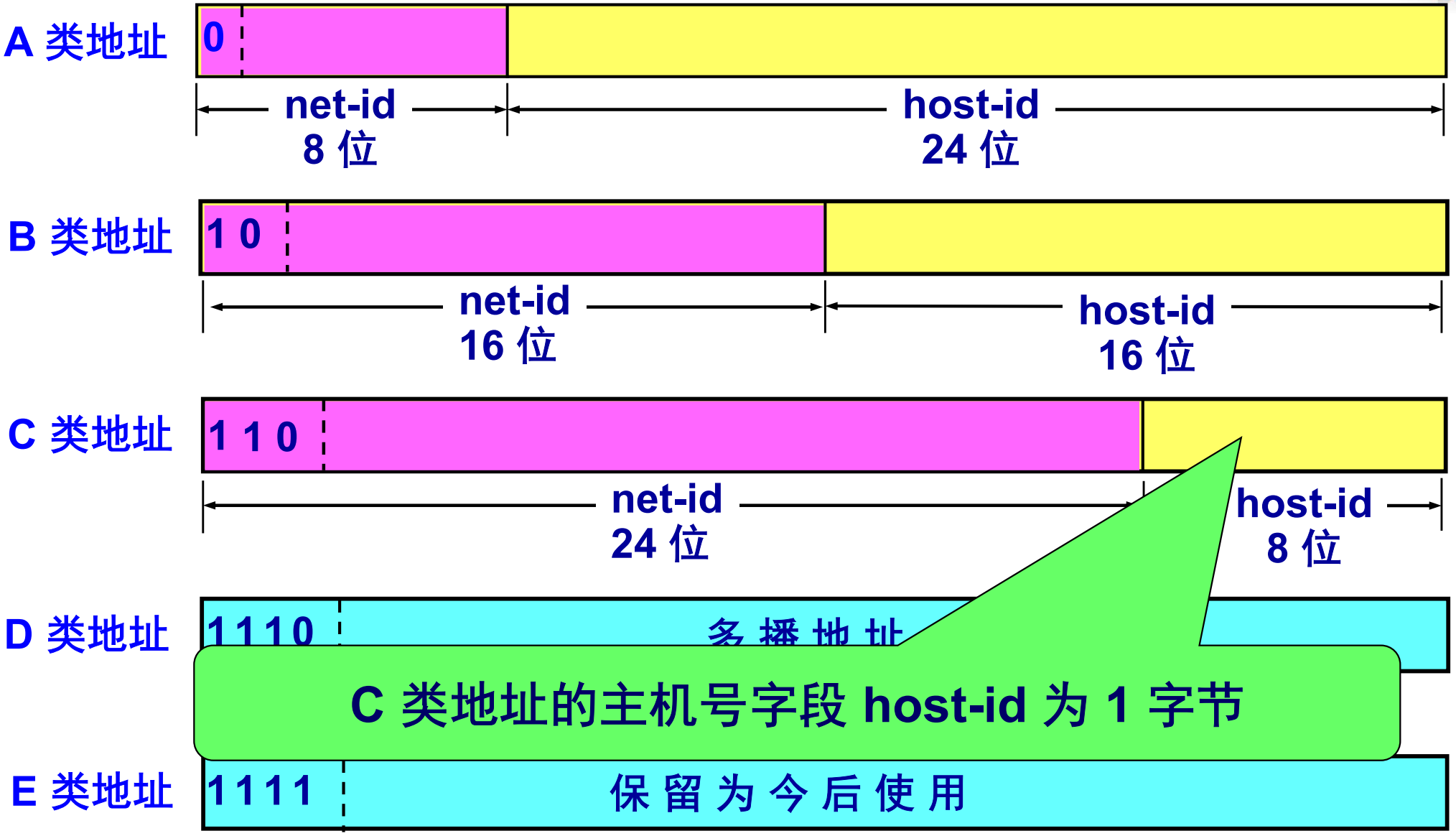
A 类地址的主机号字段 host-id 为 3 字节

各类 IP 地址的网络号字段和主机号字段

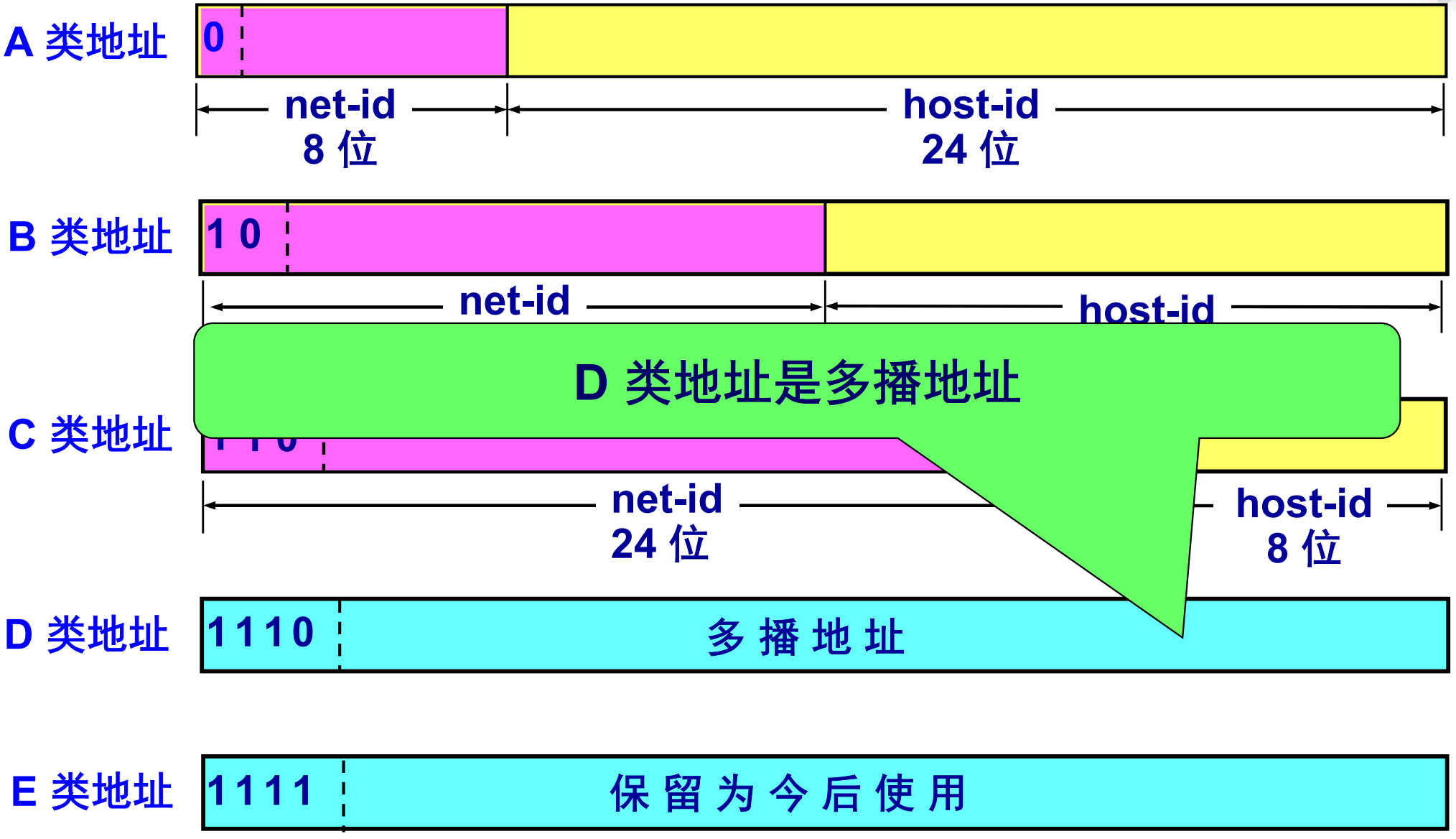


B 类地址的主机号字段 host-id 为 2 字节

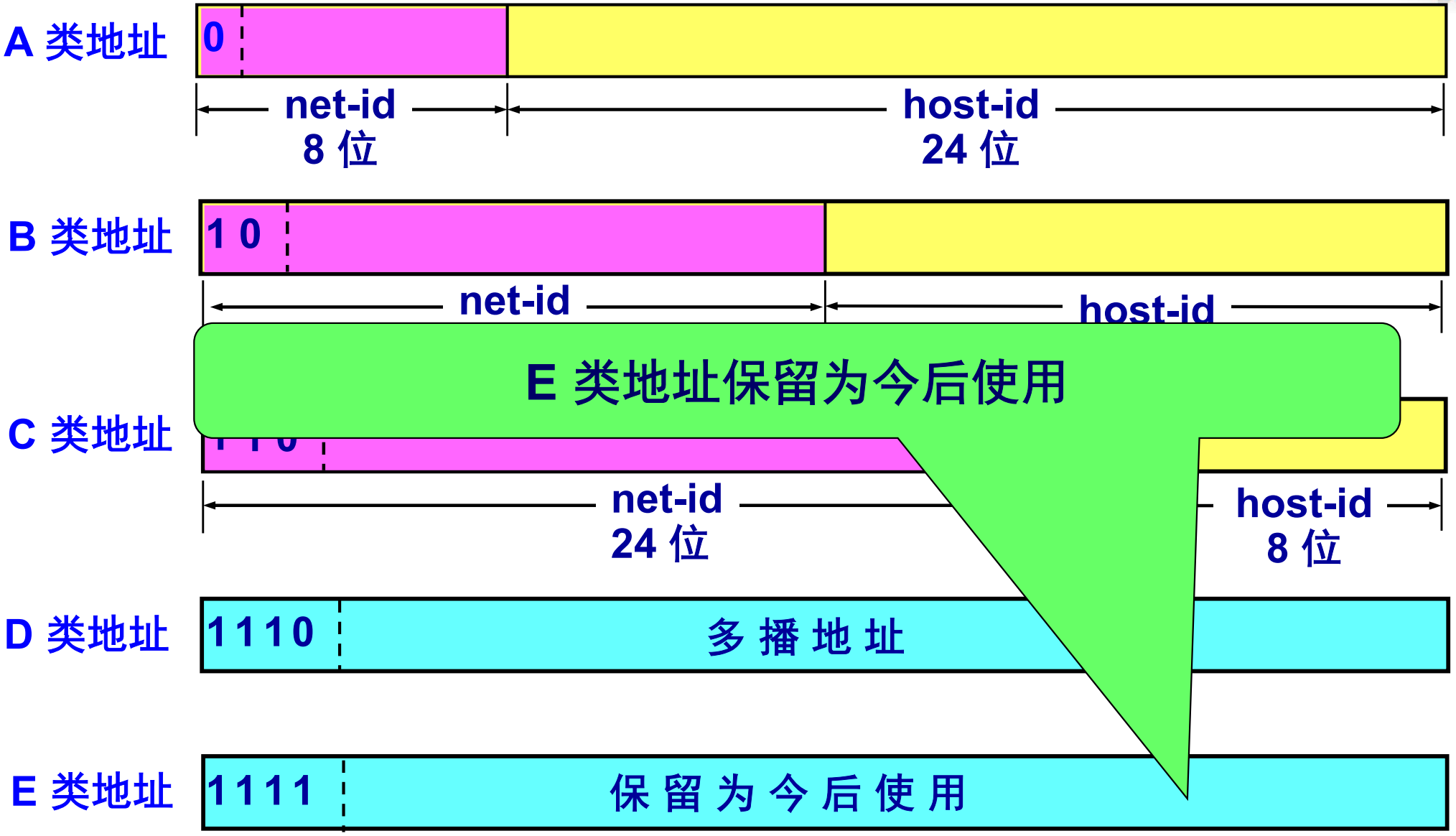
各类 IP 地址的网络号字段和主机号字段



各类 IP 地址的网络号字段和主机号字段



各类 IP 地址的网络号字段和主机号字段



点分十进制记法

机器中存放的 IP 地址
是 32 位二进制代码

10000000000010110000001100011111

每 8 位为一组

10000000 00001011 00000011 00011111

将每 8 位的二进制数
转换为十进制数

128 11 3 31

采用点分十进制记法
则进一步提高可读性

128.11.3.31

点分十进制记法举例

32 位二进制数	等价的 点分十进制数
10000001 00110100 00000110 00000000	129.52.6.0
11000000 00000101 00110000 00000011	192.5.48.3
00001010 00000010 00000000 00100101	10.2.0.37
10000000 00001010 00000010 00000011	128.10.2.3
10000000 10000000 11111111 00000000	128.128.255.0

2. 常用的三种类别的 IP 地址

IP 地址的指派范围

网络类别	最大可指派的网络数	第一个可指派的网络号	最后一个可指派的网络号	每个网络中最大主机数
A	126 ($2^7 - 2$)	1	126	16777214
B	16383 ($2^{14} - 1$)	128.1	191.255	65534
C	2097151 ($2^{21} - 1$)	192.0.1	223.255.255	254

一般不使用的特殊的 IP 地址

网络号	主机号	源地址使用	目的地址使用	代表的意思
0	0	可以	不可	在本网络上的本主机（见6.6节DHCP协议）
0	host-id	可以	不可	在本网络上的某台主机 host-id
全1	全1	不可	可以	只在本网络上进行广播（各路由器均不转发）
net-id	全1	不可	可以	对net-id上的所有主机进行广播
127	非全0或全1的任何数	可以	可以	用作本地软件环回测试之用

IP 地址的一些重要特点

- **(1) IP 地址是一种分等级的地址结构。**分两个等级的好处是：
  - 第一，IP 地址管理机构在分配 IP 地址时只分配网络号，而剩下的主机号则由得到该网络号的单位自行分配。这样就方便了 IP 地址的管理
  - 第二，路由器仅根据目的主机所连接的网络号来转发分组（而不考虑目的主机号），这样就可以使路由表中的项目数大幅度减少，从而减小了路由表所占的存储空间。

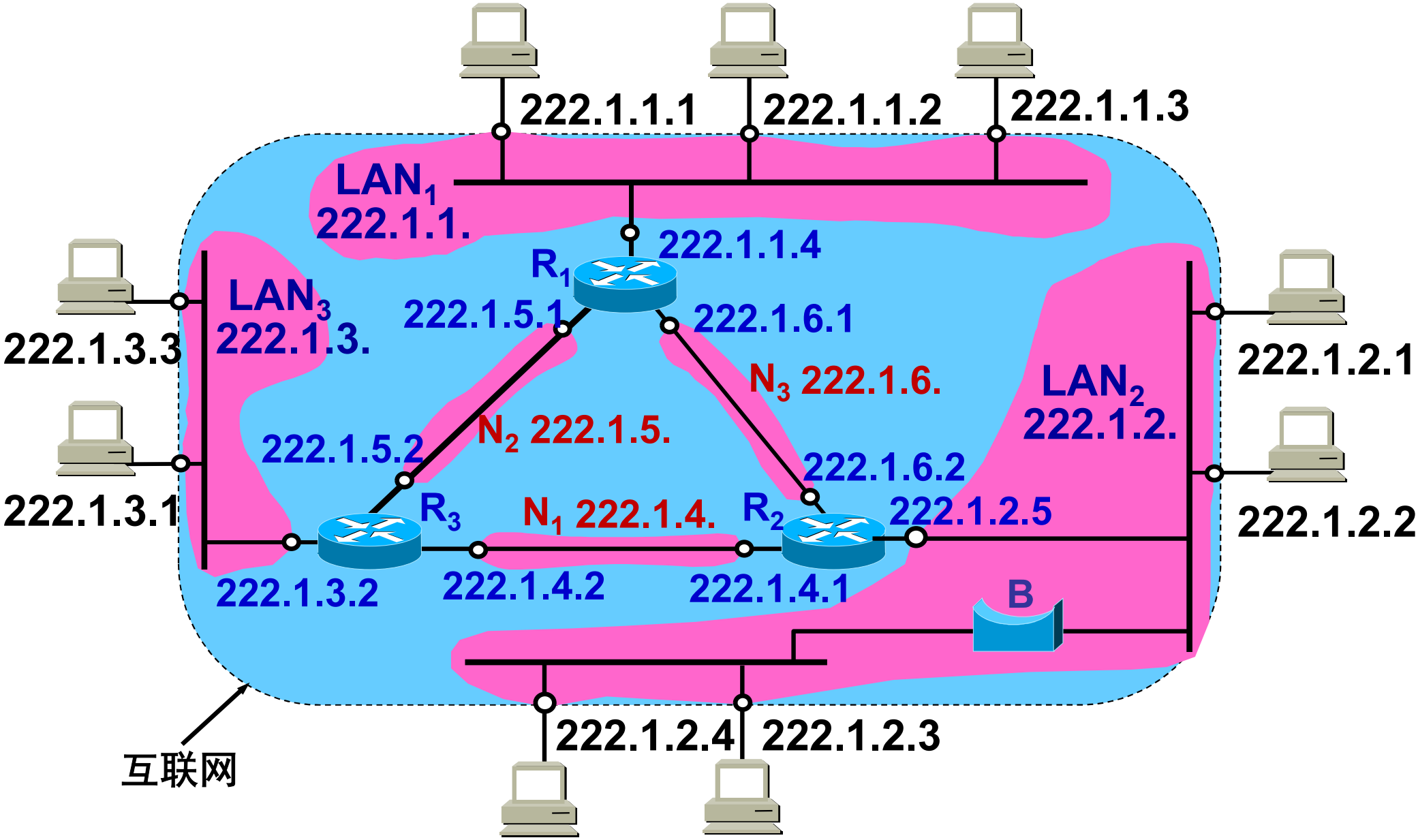
IP 地址的一些重要特点

- (2) 实际上 IP 地址是标志一个主机（或路由器）和一条链路的接口。
  - 当一个主机同时连接到两个网络上时，该主机就必须同时具有两个相应的 IP 地址，其网络号 net-id 必须是不同的。这种主机称为多归属主机 (multihomed host)。
  - 由于一个路由器至少应当连接到两个网络（这样它才能将 IP 数据报从一个网络转发到另一个网络），因此一个路由器至少应当有两个不同的 IP 地址。

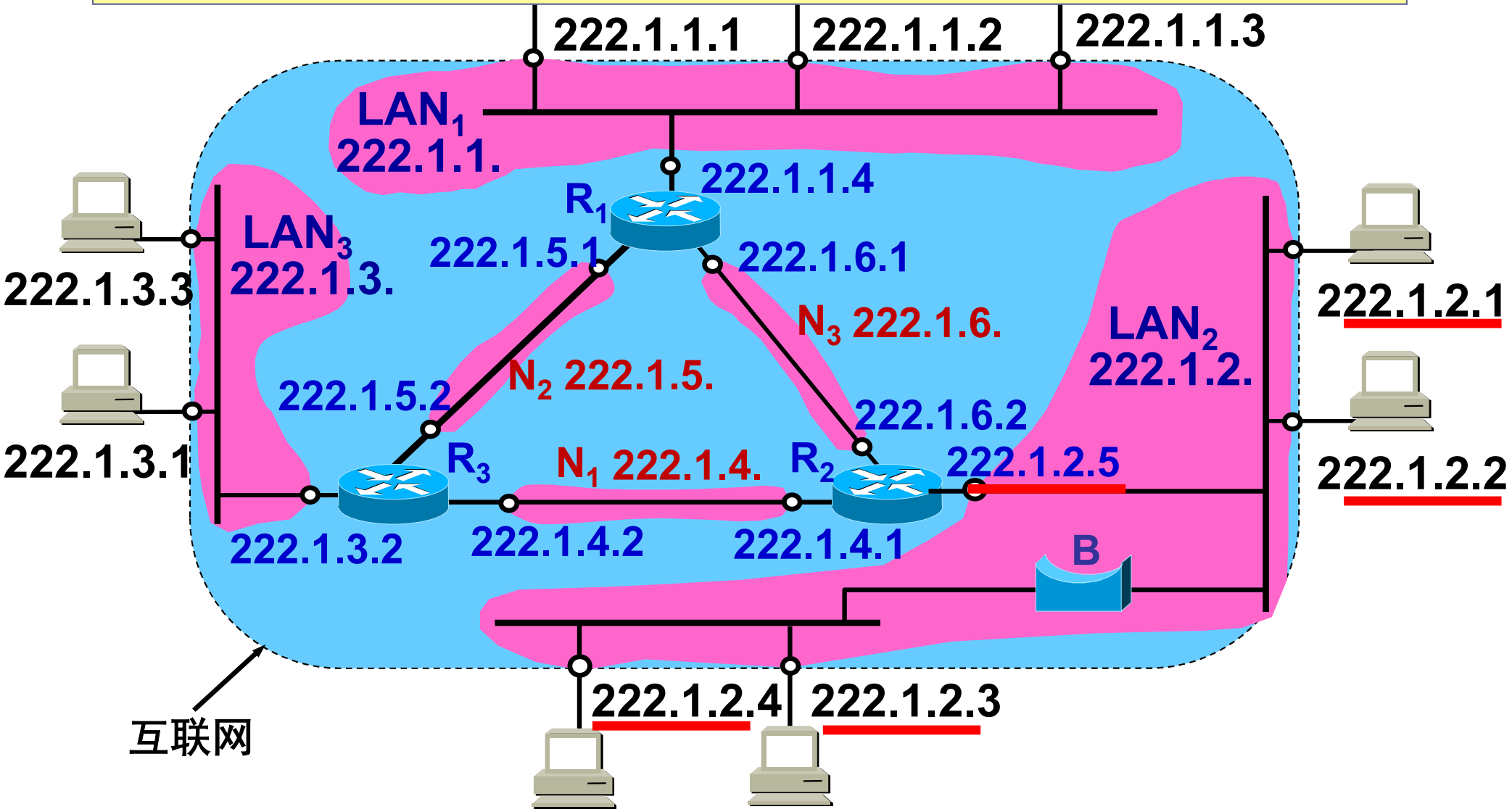
IP 地址的一些重要特点

- (3) 用转发器或网桥连接起来的若干个局域网仍为一个网络，因此这些局域网都具有同样的网络号 **net-id**。
- (4) 所有分配到网络号 **net-id** 的网络，无论是范围很小的局域网，还是可能覆盖很大地理范围的广域网，都是平等的。

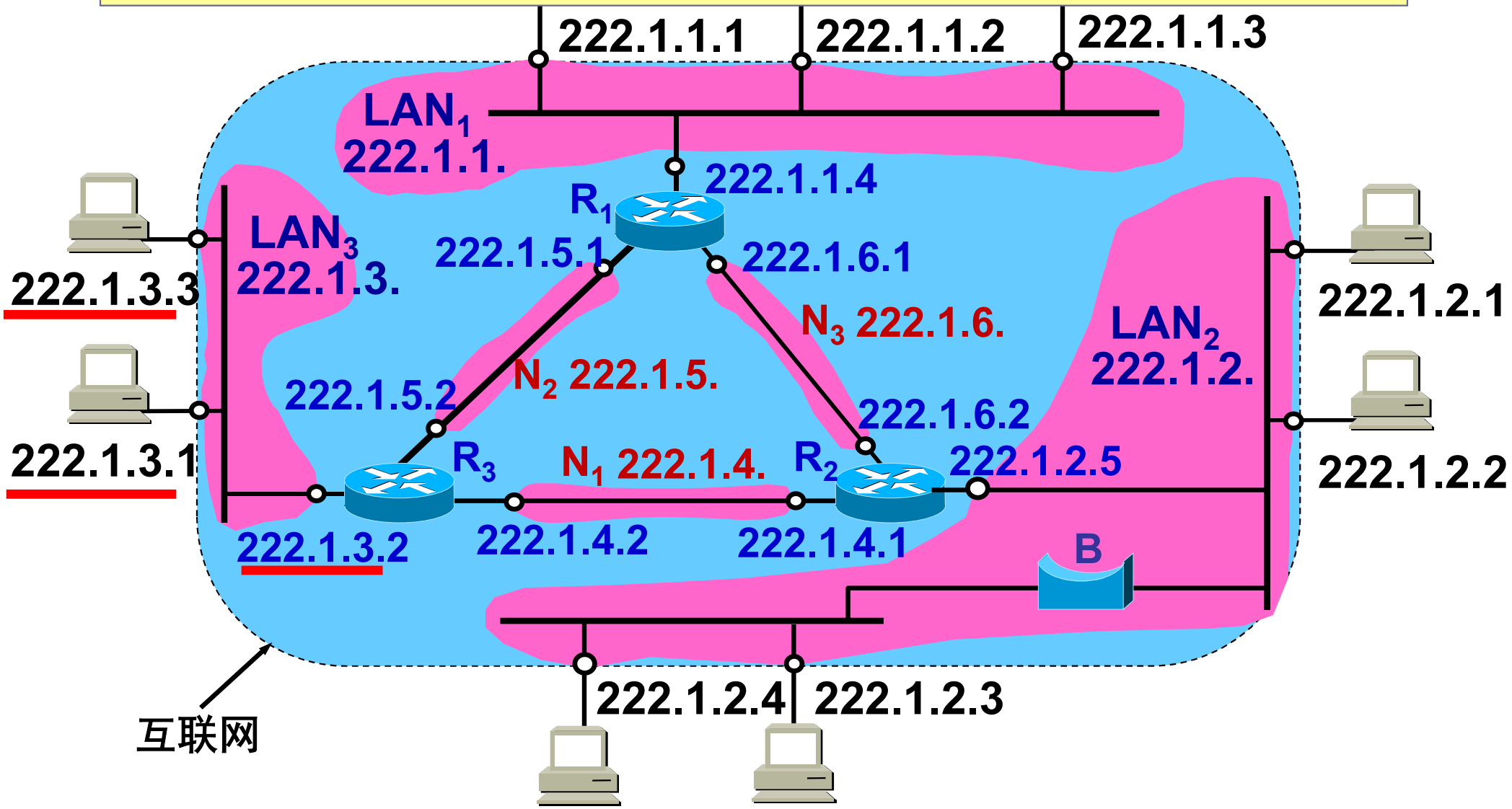
互联网中的 IP 地址


222.1.1.1
222.1.1.2
222.1.1.3
LAN1
222.1.1.
222.1.1.4
R1
LAN3
222.1.3.
222.1.5.1
222.1.6.1
222.1.3.3
222.1.2.1
N3 222.1.6.
LAN2
222.1.2.
N2 222.1.5.
222.1.5.2
222.1.6.2
222.1.3.1
222.1.2.2
R3
N1 222.1.4.
R2
222.1.2.5
222.1.3.2
222.1.4.2
222.1.4.1
B
222.1.2.4
222.1.2.3
互联网

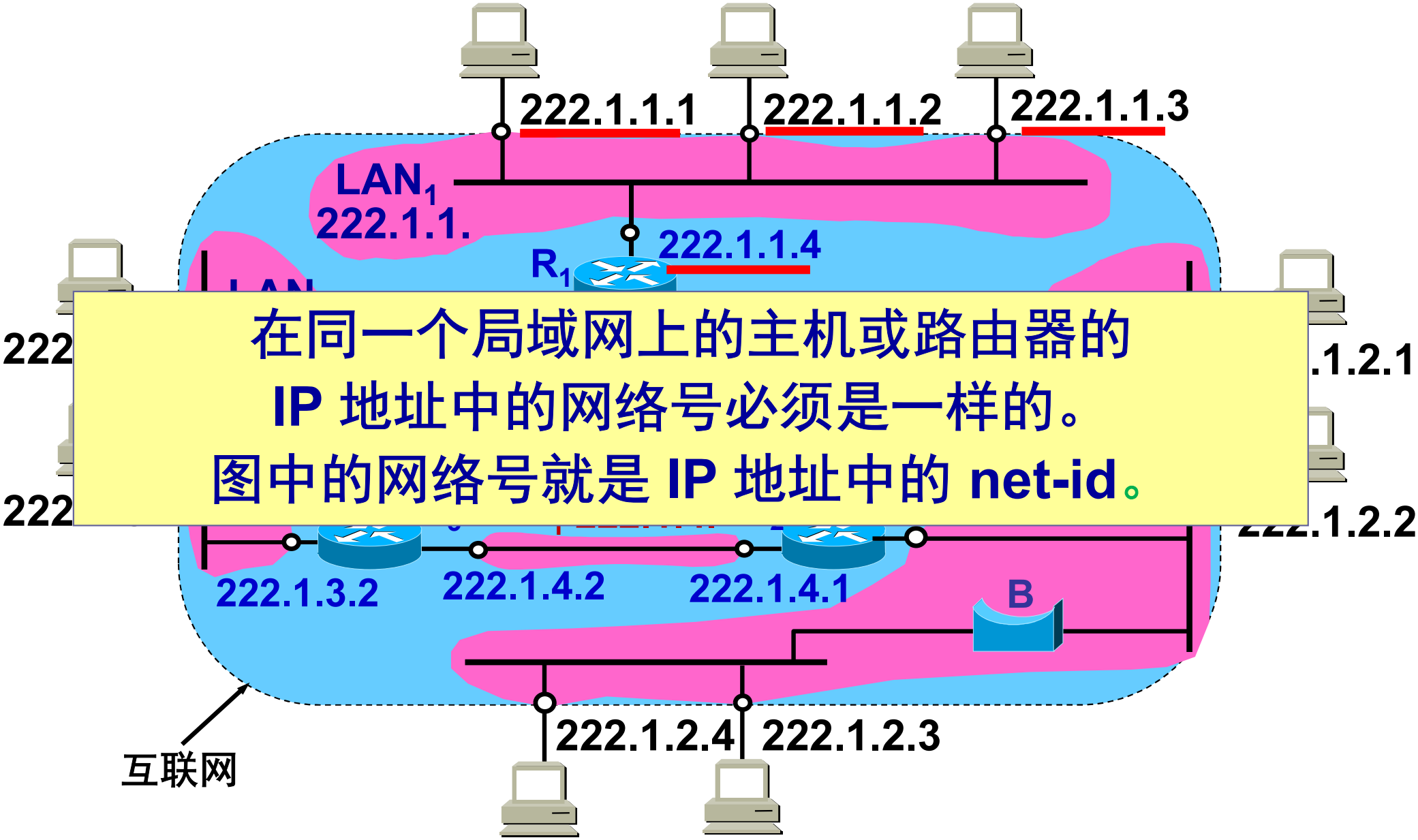
在同一个局域网上的主机或路由器的
IP 地址中的网络号必须是一样的。
图中的网络号就是 IP 地址中的 net-id



在同一个局域网上的主机或路由器的
IP 地址中的网络号必须是一样的。
图中的网络号就是 IP 地址中的 net-id。



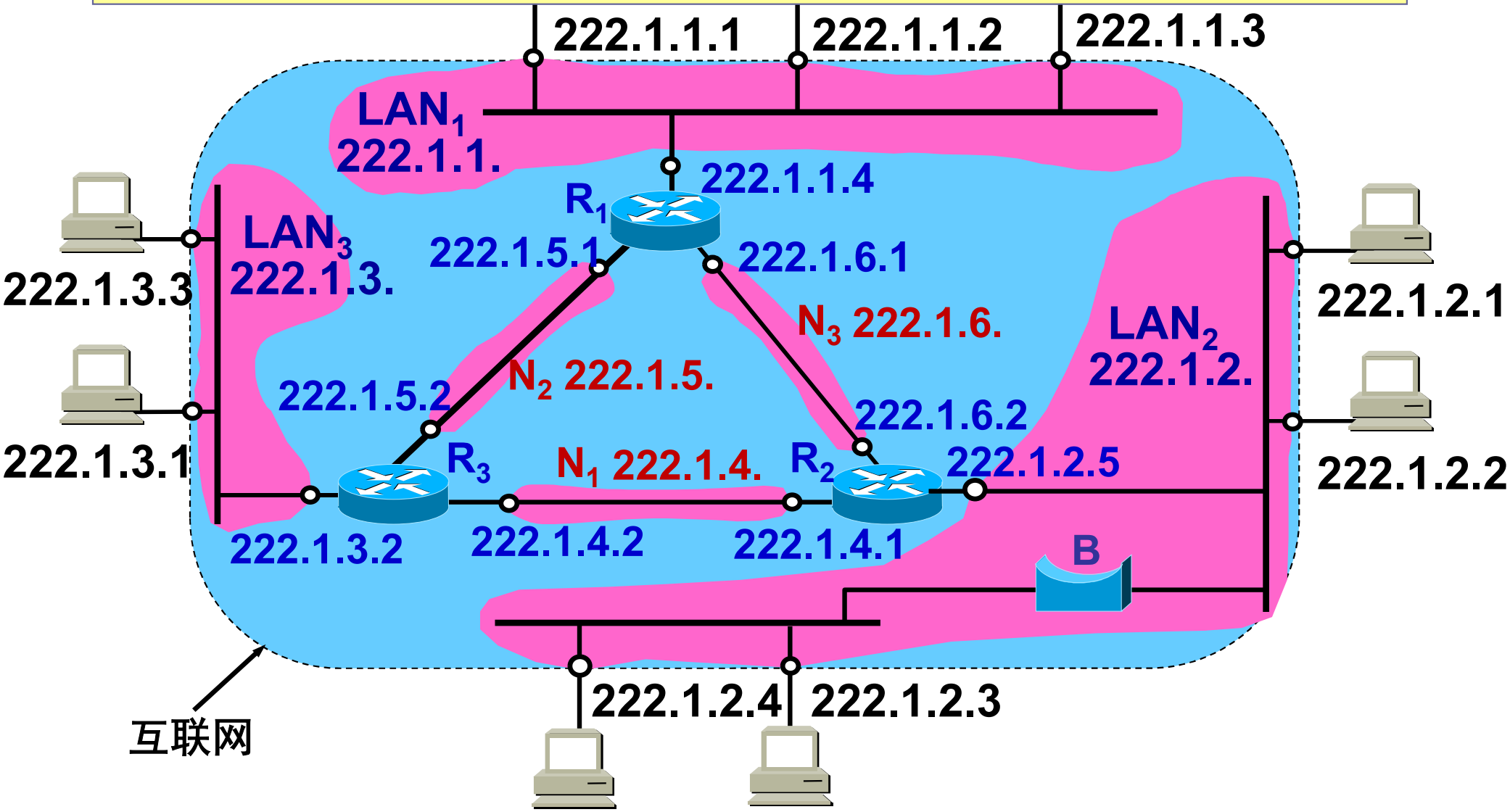
互联网中的 IP 地址



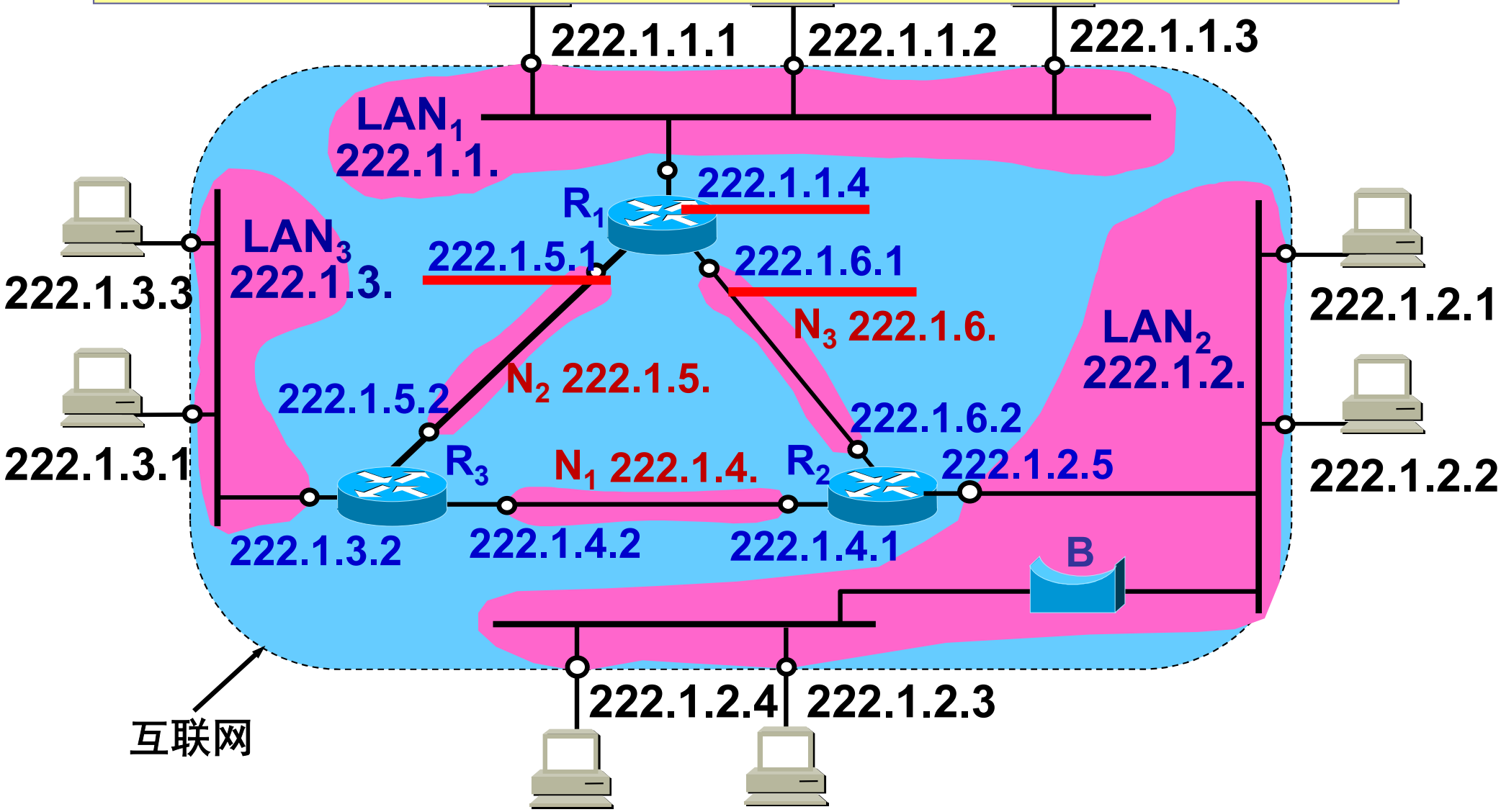
在同一个局域网上的主机或路由器的 IP 地址中的网络号必须是一样的。

图中的网络号就是 IP 地址中的 net-id。

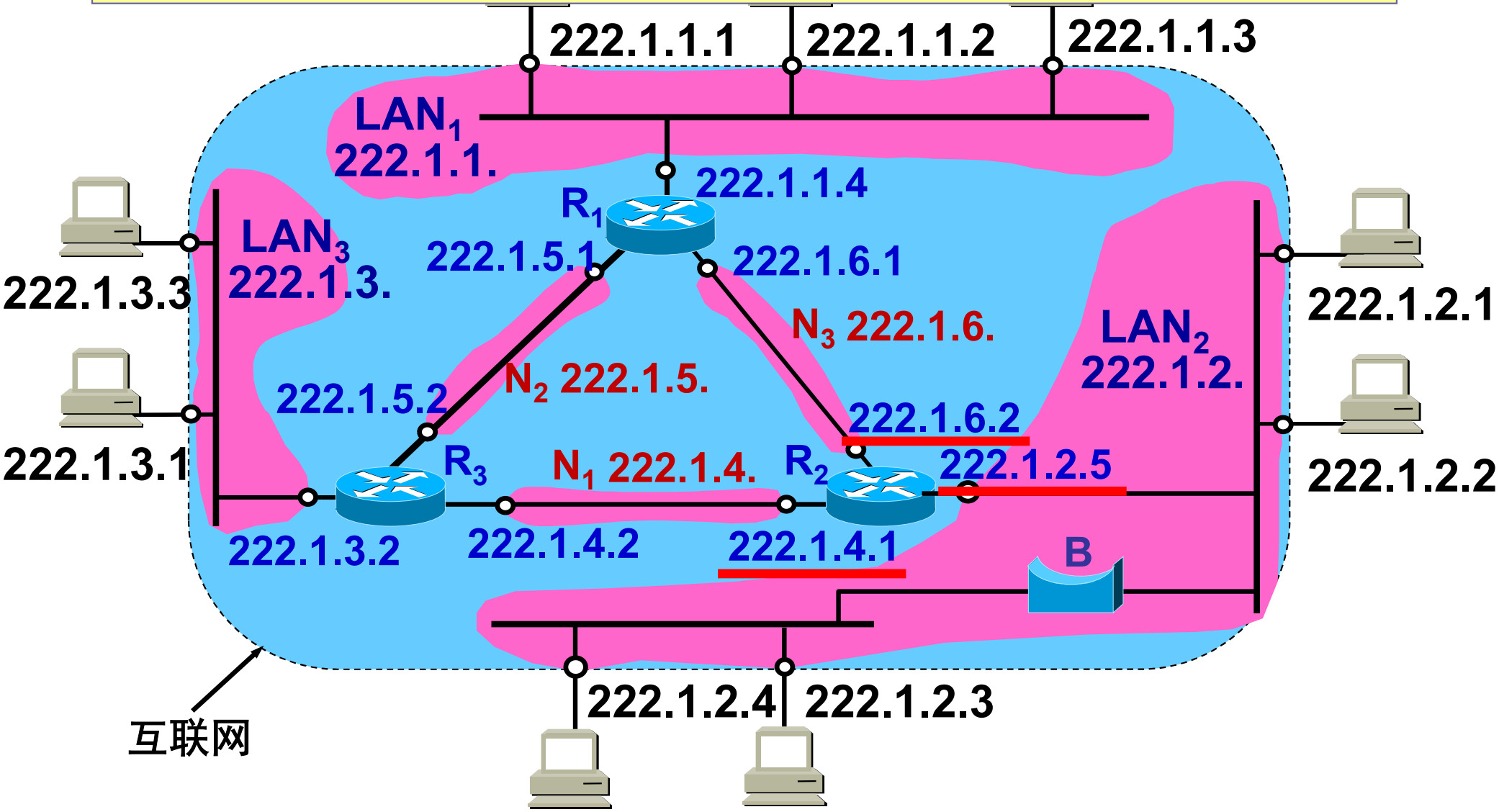
在同一个局域网上的主机或路由器的
IP 地址中的网络号必须是一样的。
图中的网络号就是 IP 地址中的 net-id。



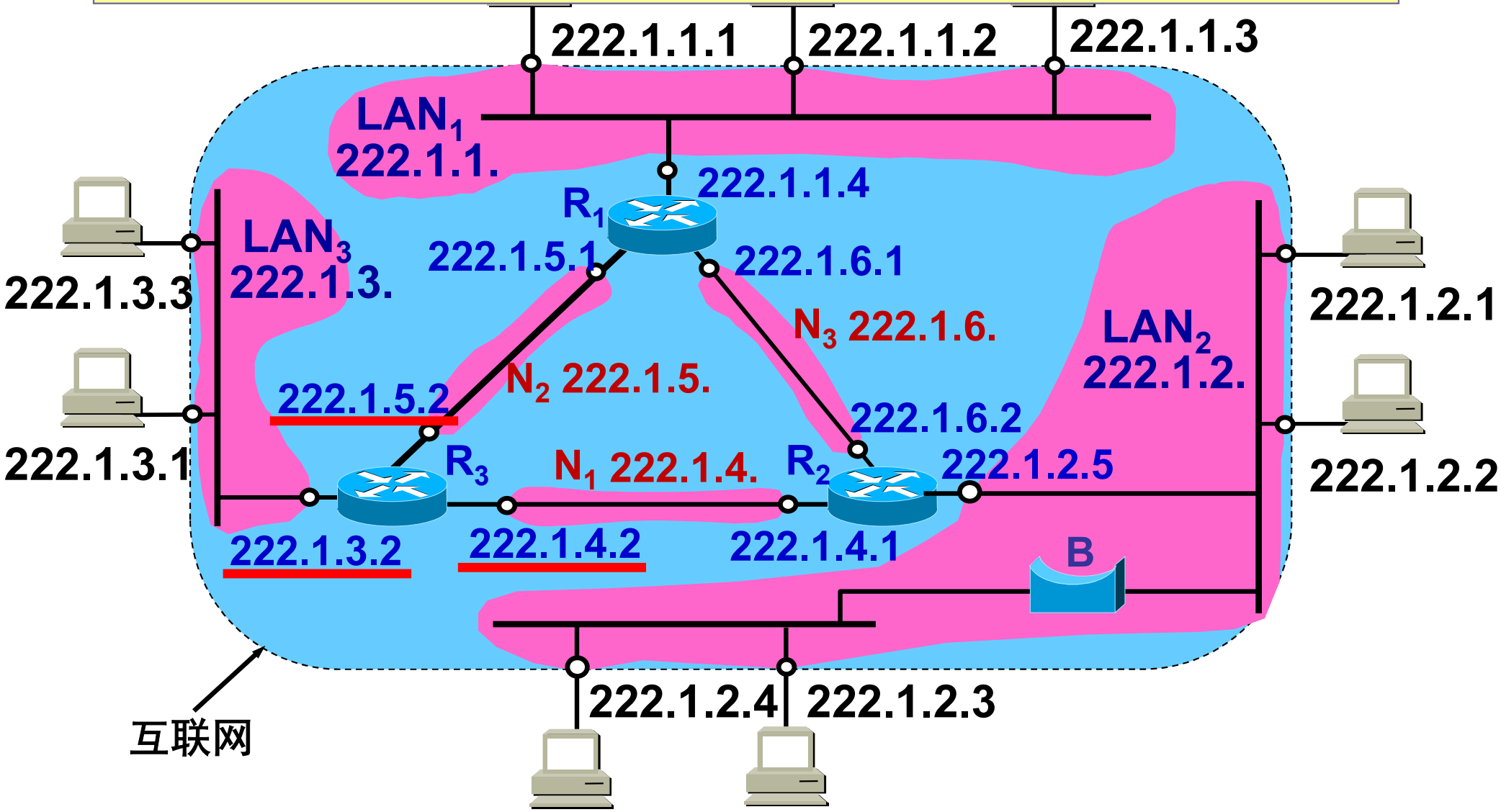
路由器总是具有两个或两个以上的 IP 地址。
路由器的每一个接口都有一个
不同网络号的 IP 地址。



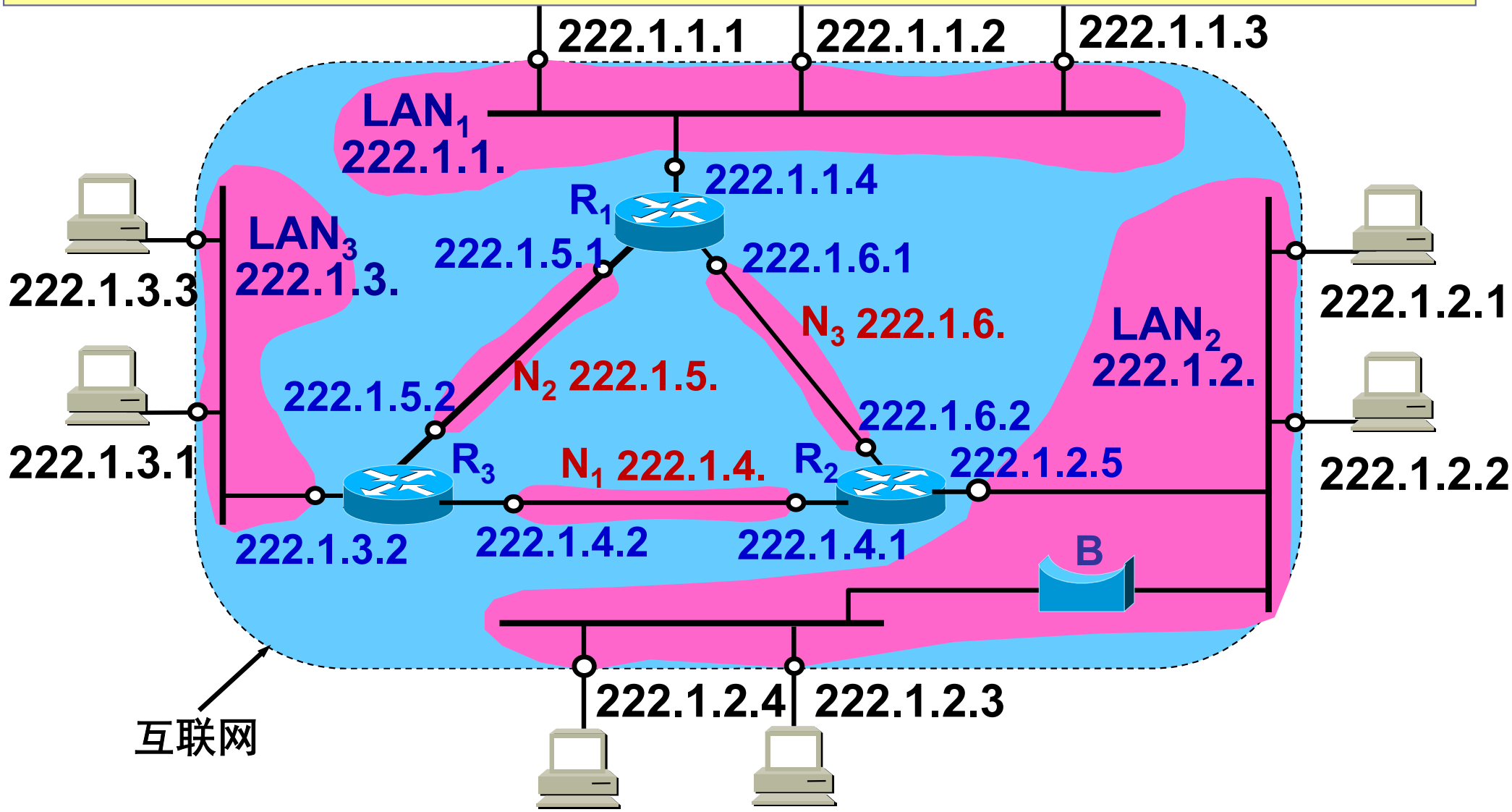
路由器总是具有两个或两个以上的 IP 地址。
路由器的每一个接口都有一个
不同网络号的 IP 地址。



路由器总是具有两个或两个以上的 IP 地址。
路由器的每一个接口都有一个
不同网络号的 IP 地址。



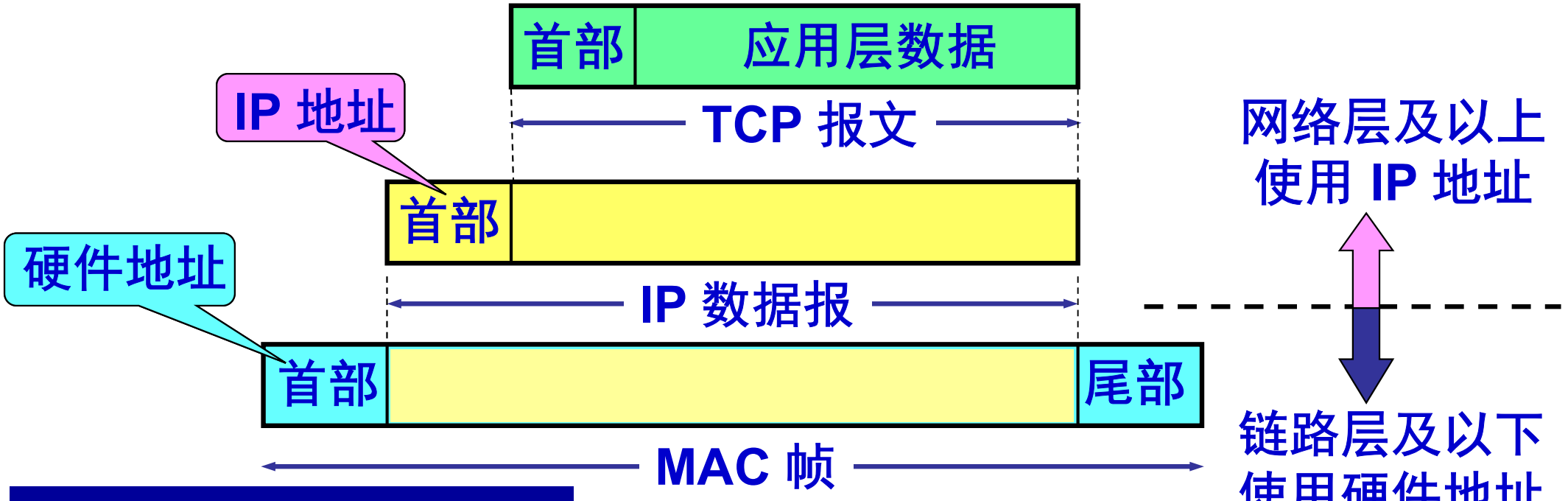
两个路由器直接相连的接口处，可指明也可不指明 IP 地址。如指明 IP 地址，则这一段连线就构成了一种只包含一段线路的特殊“网络”。现在常不指明 IP 地址。



4.2.3 IP 地址与硬件地址

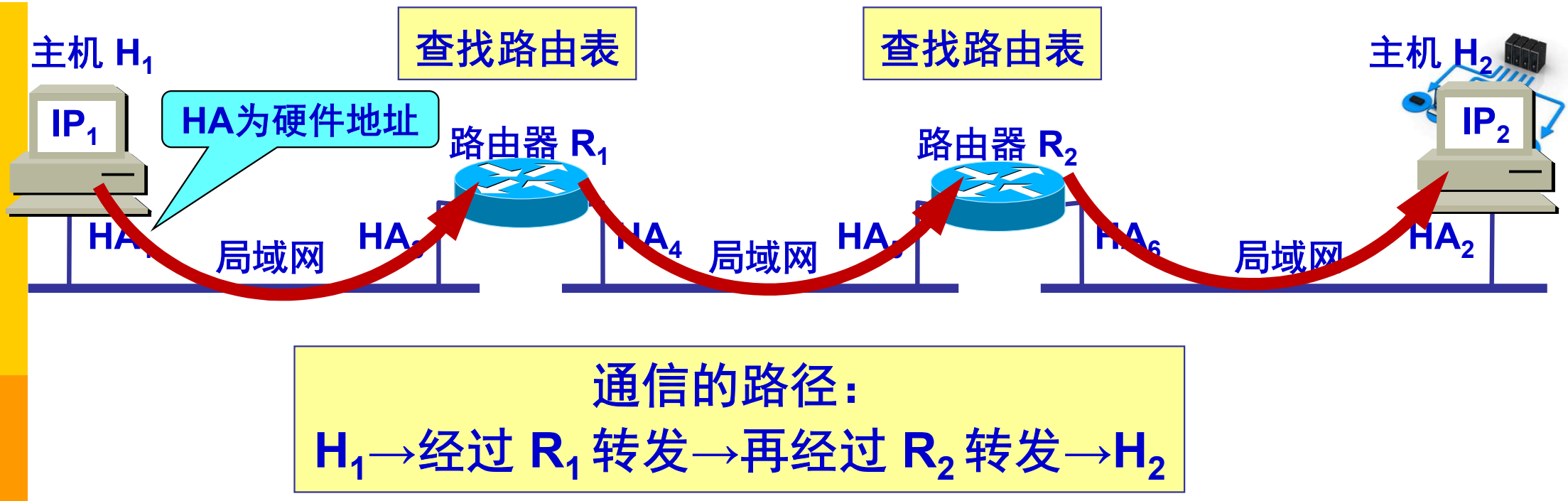
- IP 地址与硬件地址是不同的地址。
- 从层次的角度看，
  - 硬件地址（或物理地址）是数据链路层和物理层使用的地址。
  - IP 地址是网络层和以上各层使用的地址，是一种逻辑地址（称 IP 地址是逻辑地址是因为 IP 地址是用软件实现的）。

4.2.3 IP 地址与硬件地址



IP 地址放在 IP 数据报的首部，而硬件地址则放在 MAC 帧的首部。

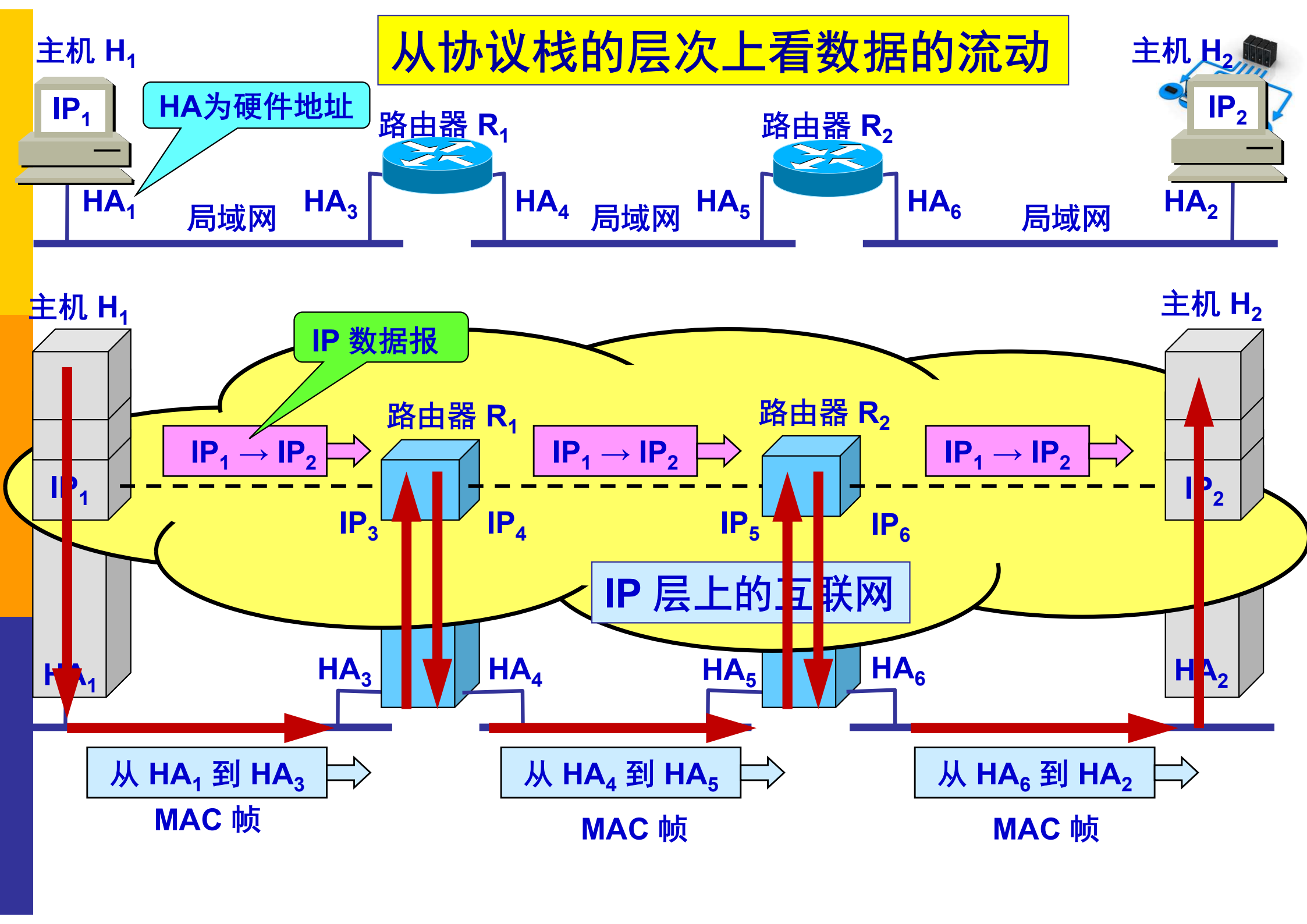
IP 地址与硬件地址的区别



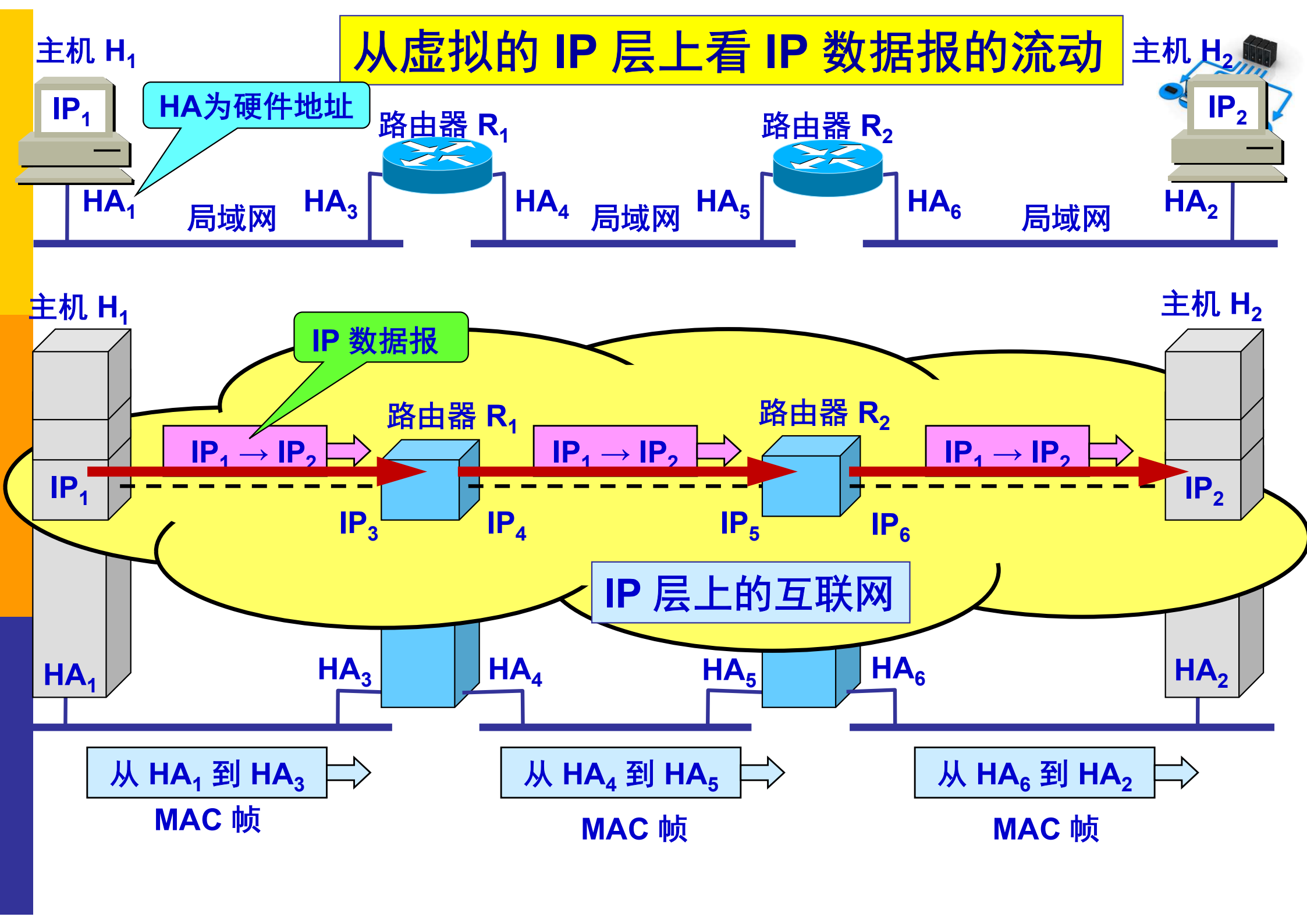
通信的路径：

H_1→经过 R_1 转发→再经过 R_2 转发→H_2

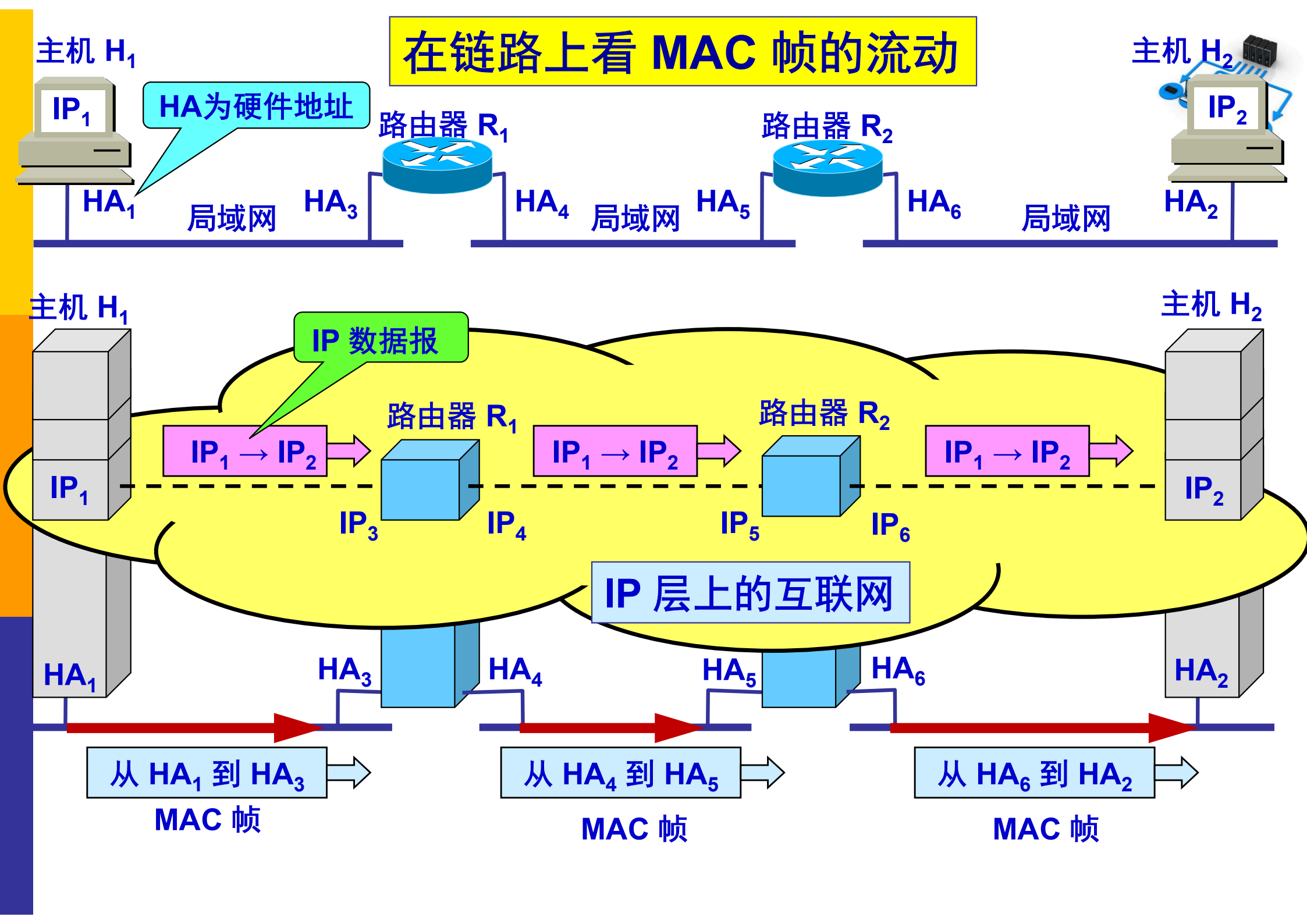
从协议栈的层次上看数据的流动



从虚拟的 IP 层上看 IP 数据报的流动



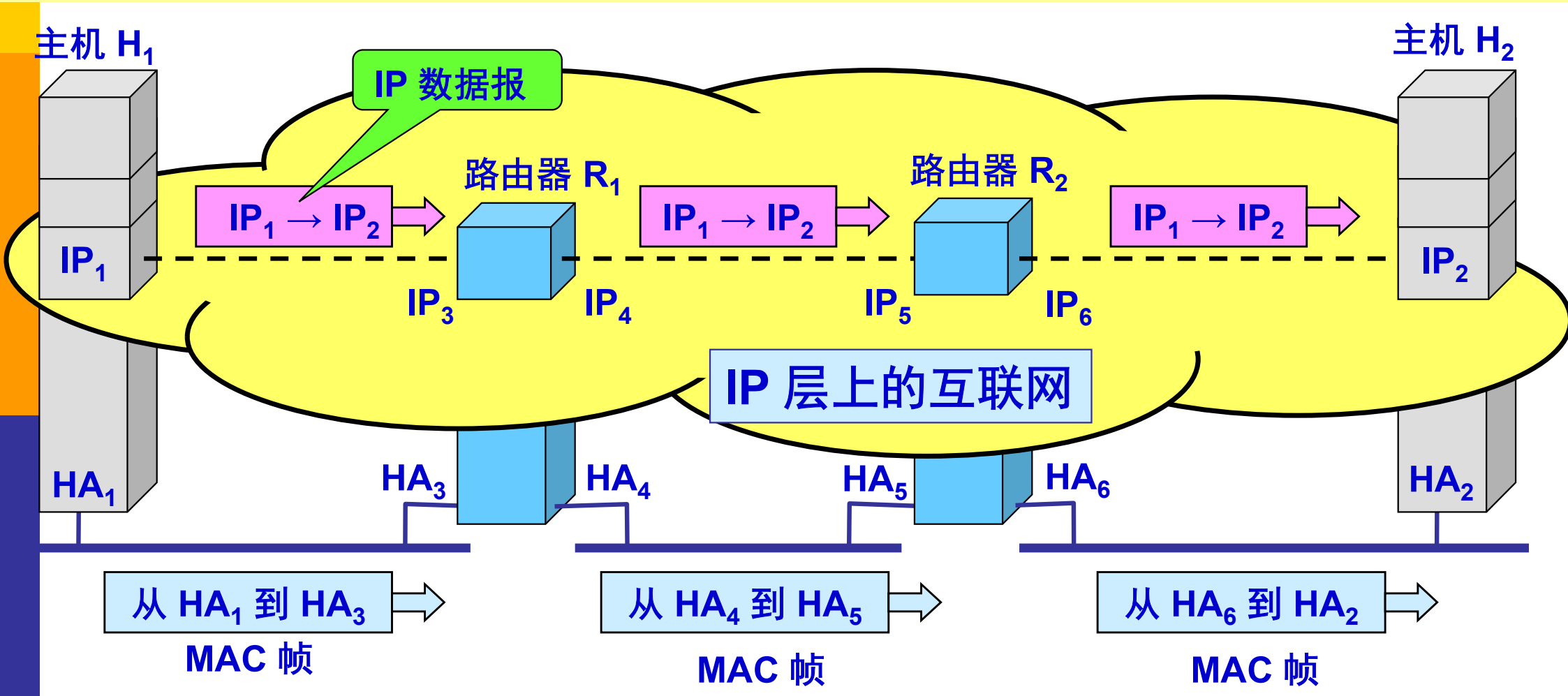
在链路上看 MAC 帧的流动



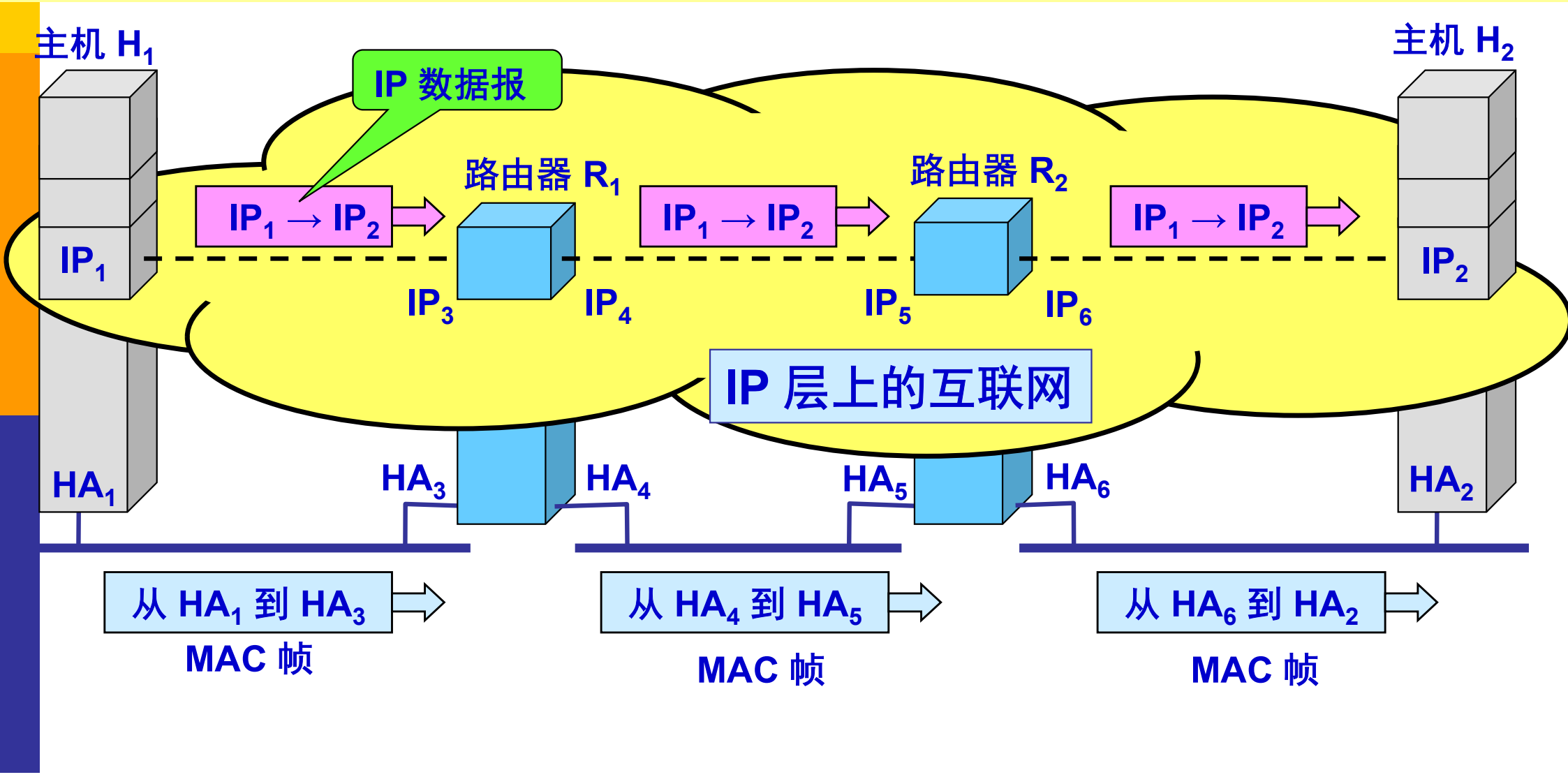
在 IP 层抽象的互联网上只能看到 IP 数据报。

图中的 $IP_1 \rightarrow IP_2$ 表示从源地址 IP_1 到目的地址 IP_2 。

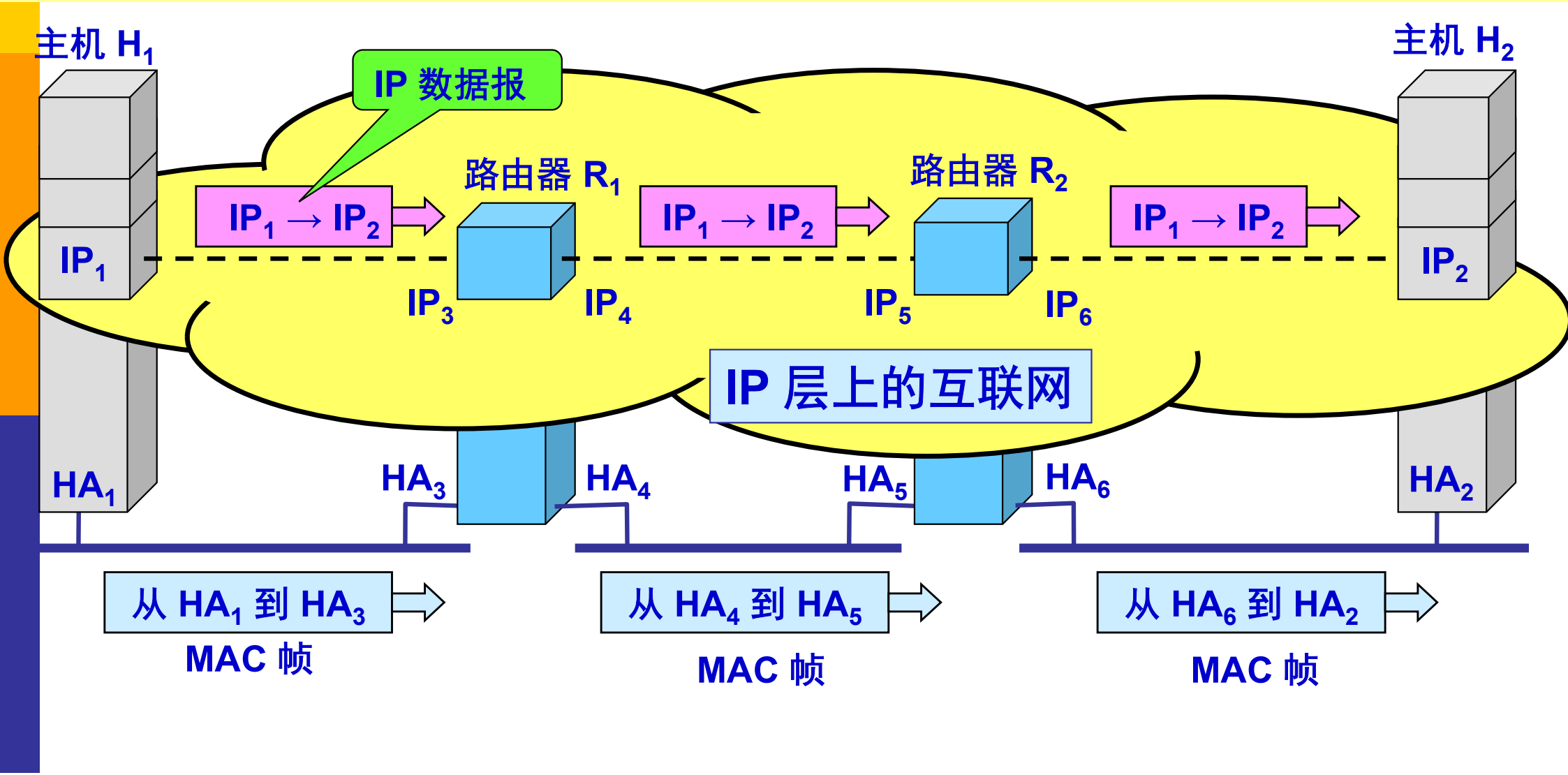
两个路由器的 IP 地址并不出现在 IP 数据报的首部中。



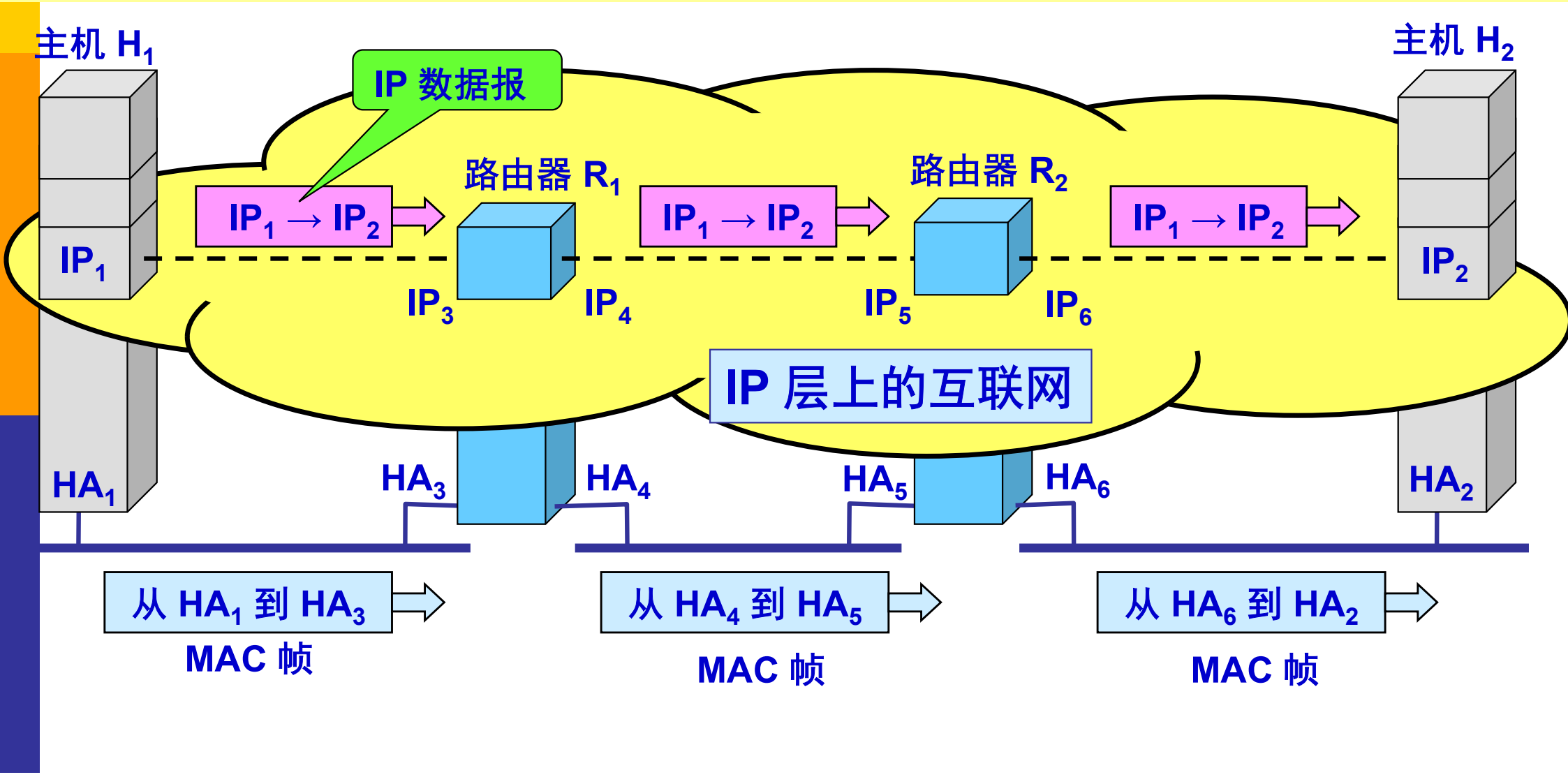
路由器只根据目的站的 IP 地址的网络号进行路由选择。



在具体的物理网络的链路层
只能看见 MAC 帧而看不见 IP 数据报



IP 层抽象的互联网屏蔽了下层很复杂的细节。
在抽象的网络层上讨论问题，就能够使用
统一的、抽象的 IP 地址
研究主机和主机或主机和路由器之间的通信 。

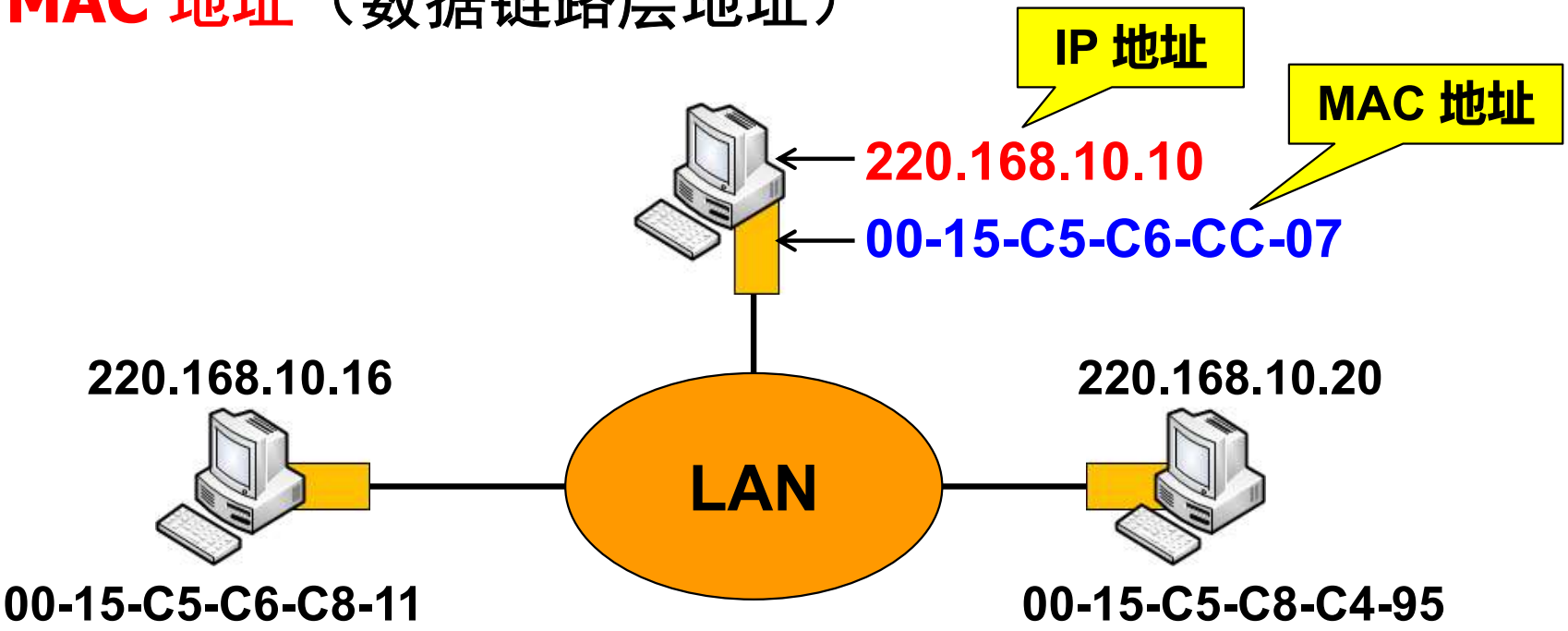


主机 H_1 与 H_2 通信中使用的 IP地址 与 硬件地址HA

	在网络层 写入IP数据报首部的地址		在数据链路层 写入MAC帧首部的地址	
	源地址	目的地址	源地址	目的地址
从H_1到R_1	IP_1	IP_2	HA_1	HA_3
从R_1到R_2	IP_1	IP_2	HA_4	HA_5
从R_2到H_2	IP_1	IP_2	HA_6	HA_2

4.2.4 地址解析协议 ARP

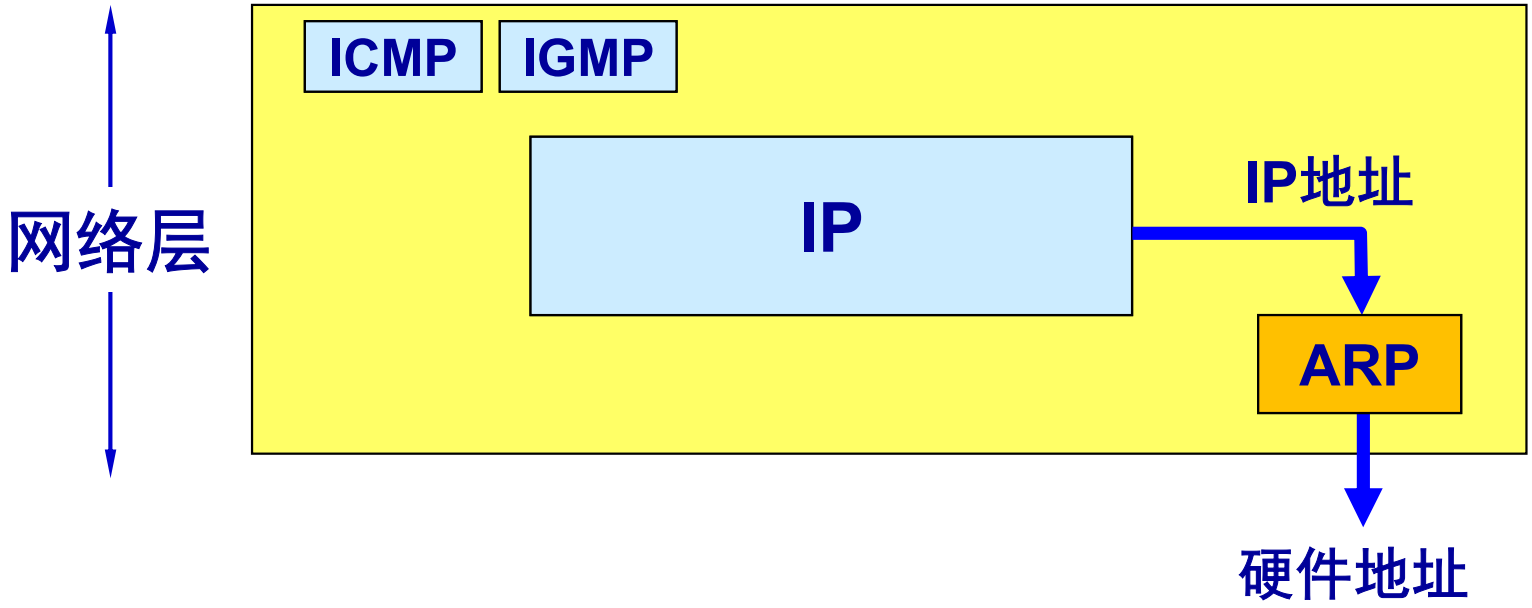
- 通信时使用了两个地址：
  - **IP 地址**（网络层地址）
  - **MAC 地址**（数据链路层地址）



每个接口都有两个地址

地址解析协议 ARP 的作用

- 已经知道了一个机器（主机或路由器）的IP地址，如何找出其相应的硬件地址？
- 地址解析协议 ARP 就是用来解决这样的问题的。



ARP 协议的作用

ARP 作用：
从网络层使用的 IP 地址，解析出在数据链路层使用的硬件地址。

地址解析协议 ARP 要点

- 不管网络层使用的是什么协议，在实际网络的链路上传送数据帧时，最终还是必须使用硬件地址。
- 每一个主机都设有一个 **ARP 高速缓存** (ARP cache)，里面有所在的局域网上的各主机和路由器的 IP 地址到硬件地址的映射表。

< IP address；MAC address；TTL >

TTL (Time To Live)：地址映射有效时间 。

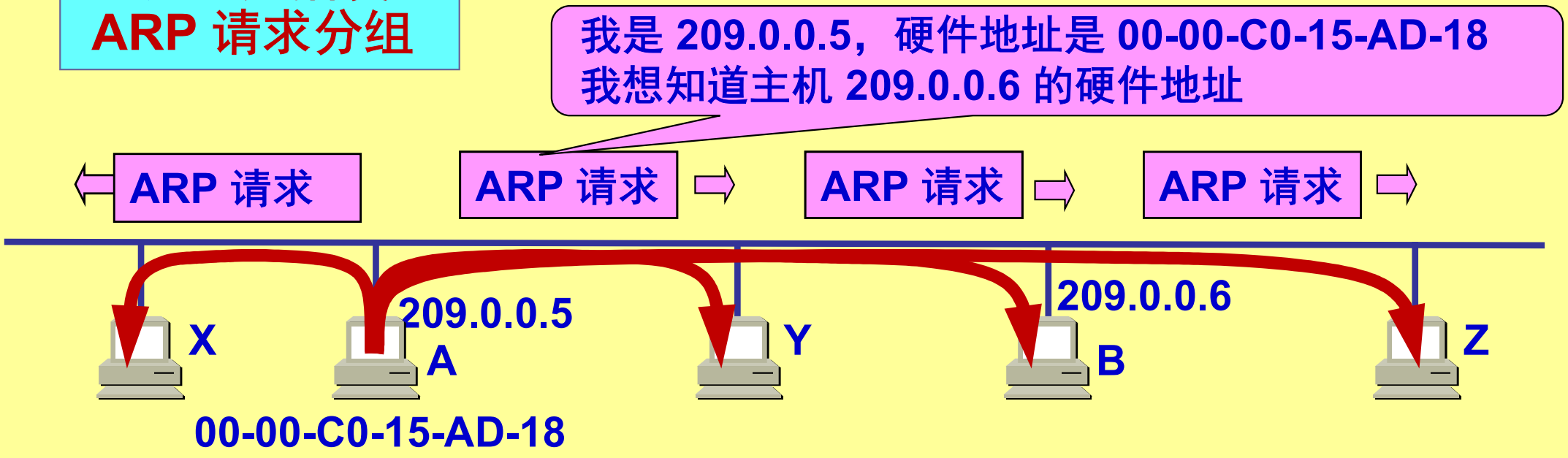
地址解析协议 ARP 要点

- 当主机 A 欲向本局域网上的某个主机 B 发送 IP 数据报时，就先在其 ARP 高速缓存中查看有无主机 B 的 IP 地址。
  - 如有，就可查出其对应的硬件地址，再将此硬件地址写入 MAC 帧，然后通过局域网将该 MAC 帧发往此硬件地址。
  - 如没有， ARP 进程在本局域网上广播发送一个 ARP 请求分组。收到 ARP 响应分组后，将得到的 IP 地址到硬件地址的映射写入 ARP 高速缓存。

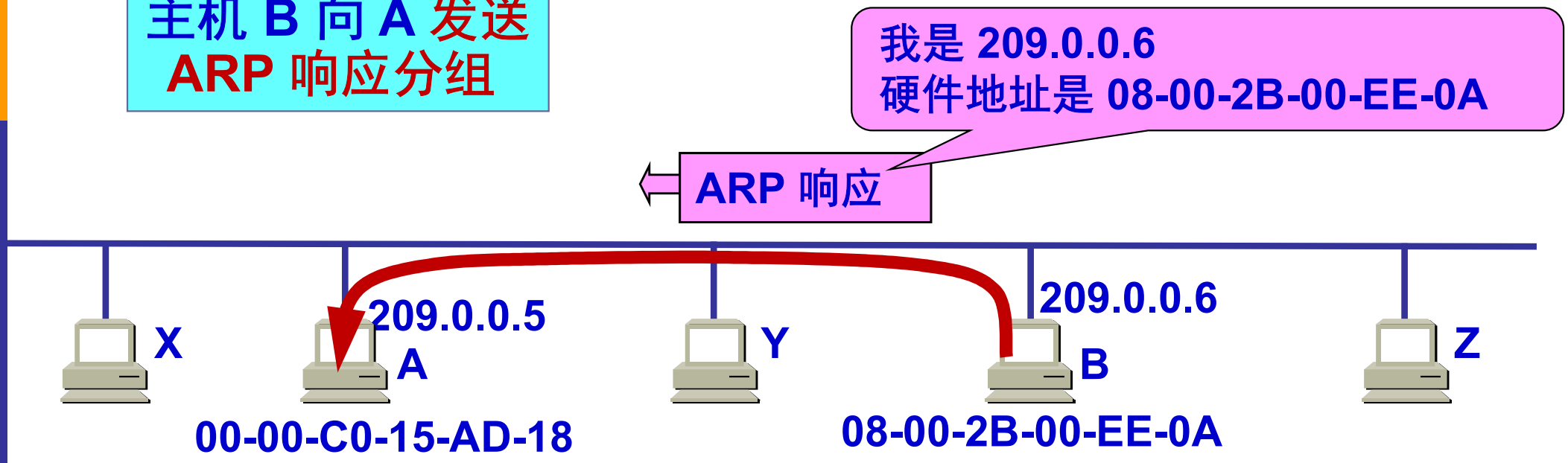
地址解析协议 ARP 要点

- **ARP请求分组：**包含发送方硬件地址 / 发送方 IP 地址 / 目标方硬件地址(未知时填 0) / 目标方 IP 地址。
- 本地广播 ARP 请求（路由器不转发ARP请求）。
- **ARP 响应分组：**包含发送方硬件地址 / 发送方IP地址 / 目标方硬件地址 / 目标方 IP 地址。
- ARP 分组封装在物理网络的帧中传输。

主机 A 广播发送
ARP 请求分组



主机 B 向 A 发送
ARP 响应分组



ARP 高速缓存的作用

- 存放最近获得的 IP 地址到 MAC 地址的绑定，以减少 ARP 广播的数量。
- 为了减少网络上的通信量，主机 A 在发送其 ARP 请求分组时，就将自己的 IP 地址到硬件地址的映射写入 ARP 请求分组。
- 当主机 B 收到 A 的 ARP 请求分组时，就将主机 A 的这一地址映射写入主机 B 自己的 ARP 高速缓存中。这对主机 B 以后向 A 发送数据报时就更方便了。

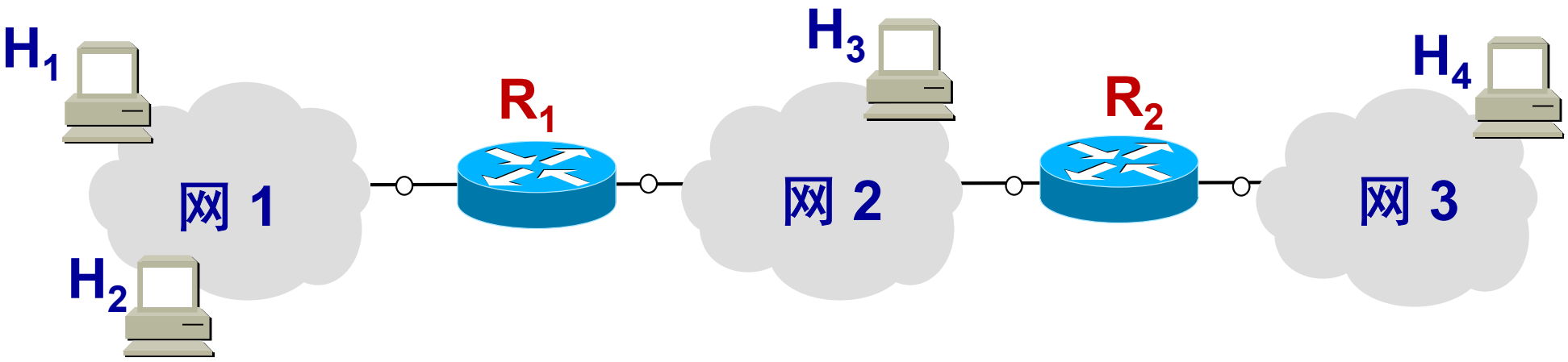
应当注意的问题

- **ARP 是解决同一个局域网上的主机或路由器的 IP 地址和硬件地址的映射问题。**
- 如果所要找的主机和源主机不在同一个局域网上，那么就要通过 ARP 找到一个位于本局域网上的某个路由器的硬件地址，然后把分组发送给这个路由器，让这个路由器把分组转发给下一个网络。剩下的工作就由下一个网络来做。

应当注意的问题（续）

- 从 IP 地址到硬件地址的解析是自动进行的，主机的用户对这种地址解析过程是不知道的。
- 只要主机或路由器要和本网络上的另一个已知 IP 地址的主机或路由器进行通信，ARP 协议就会自动地将该 IP 地址解析为链路层所需要的硬件地址。

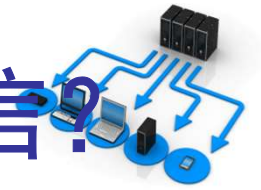
使用 ARP 的四种典型情况



使用 ARP 的四种典型情况

- 发送方是主机，要把 IP 数据报发送到本网络上的另一个主机。这时用 ARP 找到目的主机的硬件地址。
- 发送方是主机，要把 IP 数据报发送到另一个网络上的一个主机。这时用 ARP 找到本网络上的一个路由器的硬件地址。剩下的工作由这个路由器来完成。
- 发送方是路由器，要把 IP 数据报转发到本网络上的一个主机。这时用 ARP 找到目的主机的硬件地址。
- 发送方是路由器，要把 IP 数据报转发到另一个网络上的一个主机。这时用 ARP 找到本网络上另一个路由器的硬件地址。剩下的工作由这个路由器来完成。

什么？我们不直接使用硬件地址进行通信？

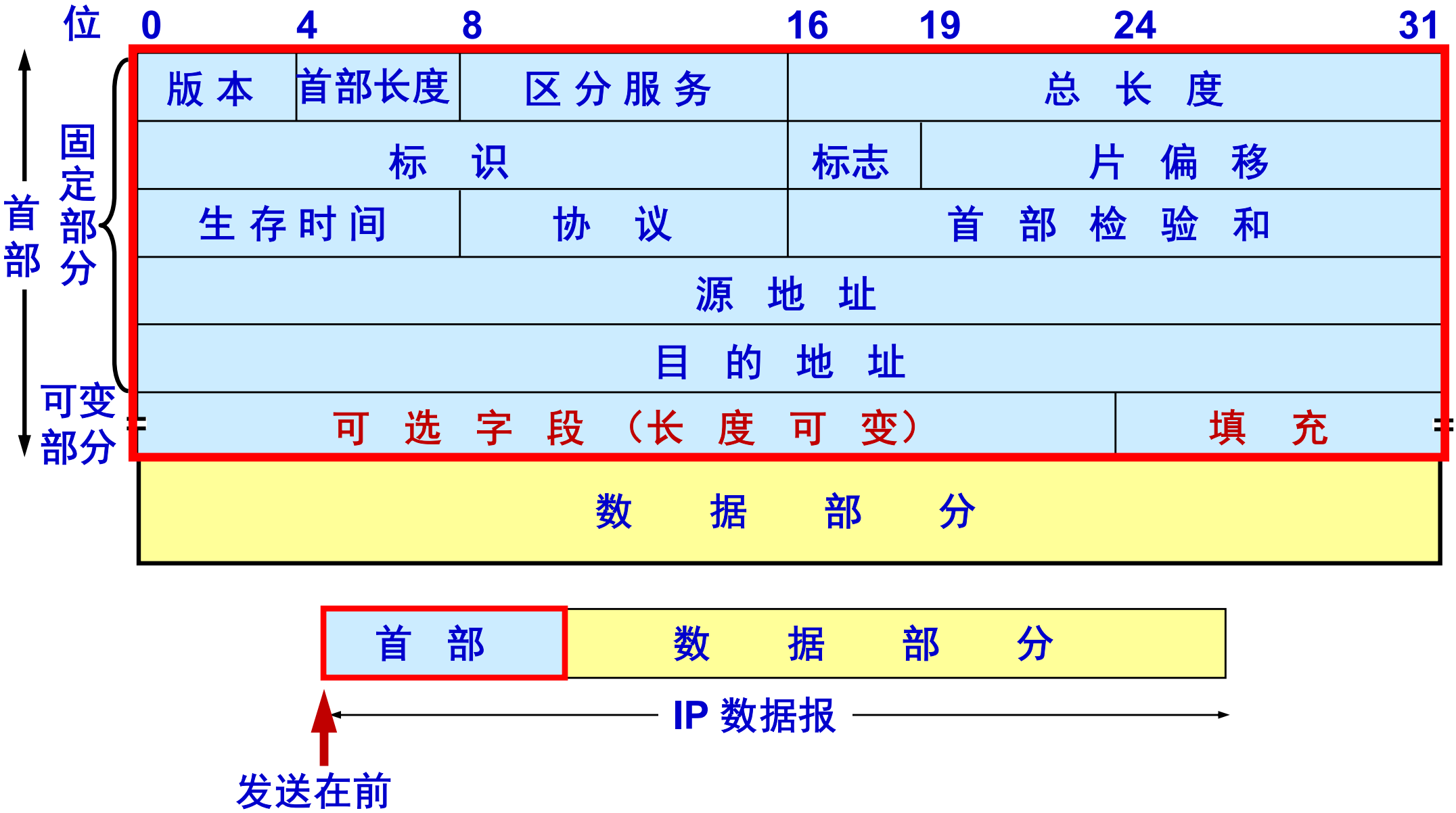


- 由于全世界存在着各式各样的网络，它们使用不同的硬件地址。要使这些异构网络能够互相通信就必须进行非常复杂的硬件地址转换工作，因此几乎是不可能的事。
- **IP 编址把这个复杂问题解决了。**连接到互联网的主机只需各自拥有一个唯一的 IP 地址，它们之间的通信就像连接在同一个网络上那样简单方便，因为上述的调用 ARP 的复杂过程都是由计算机软件自动进行的，对用户来说是看不见这种调用过程的。
- **因此，在虚拟的 IP 网络上用 IP 地址进行通信给广大的计算机用户带来了很大的方便。**

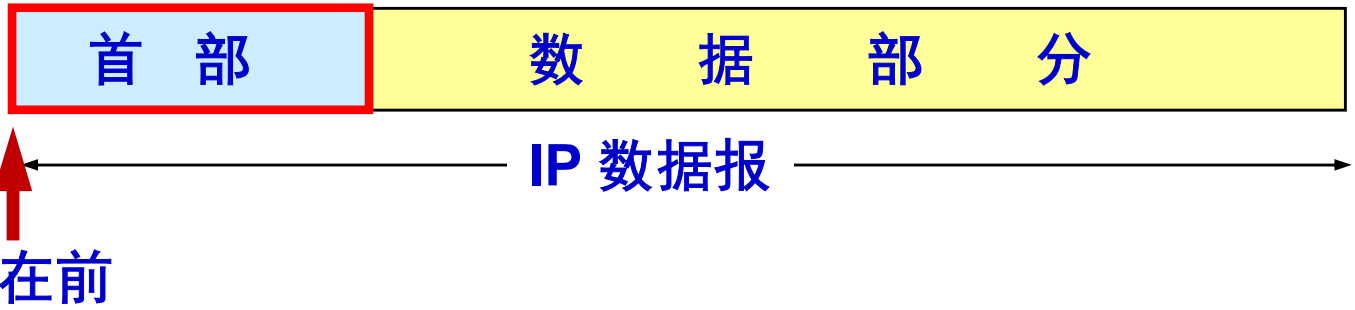
4.2.5 IP 数据报的格式

- 一个 IP 数据报由首部和数据两部分组成。
- 首部的前一部分是固定长度，共 20 字节，是所有 IP 数据报必须具有的。
- 在首部的固定部分的后面是一些可选字段，其长度是可变的。

IP 数据报由首部和数据两部分组成



首部的前一部分是固定长度，共 20 字节，是所有 IP 数据报必须具有的。



可选字段，其长度是可变的

1. IP 数据报首部的固定部分中的各字段

版本——占 4 位，指 IP 协议的版本。
目前的 IP 协议版本号为 4 (即 IPv4)。

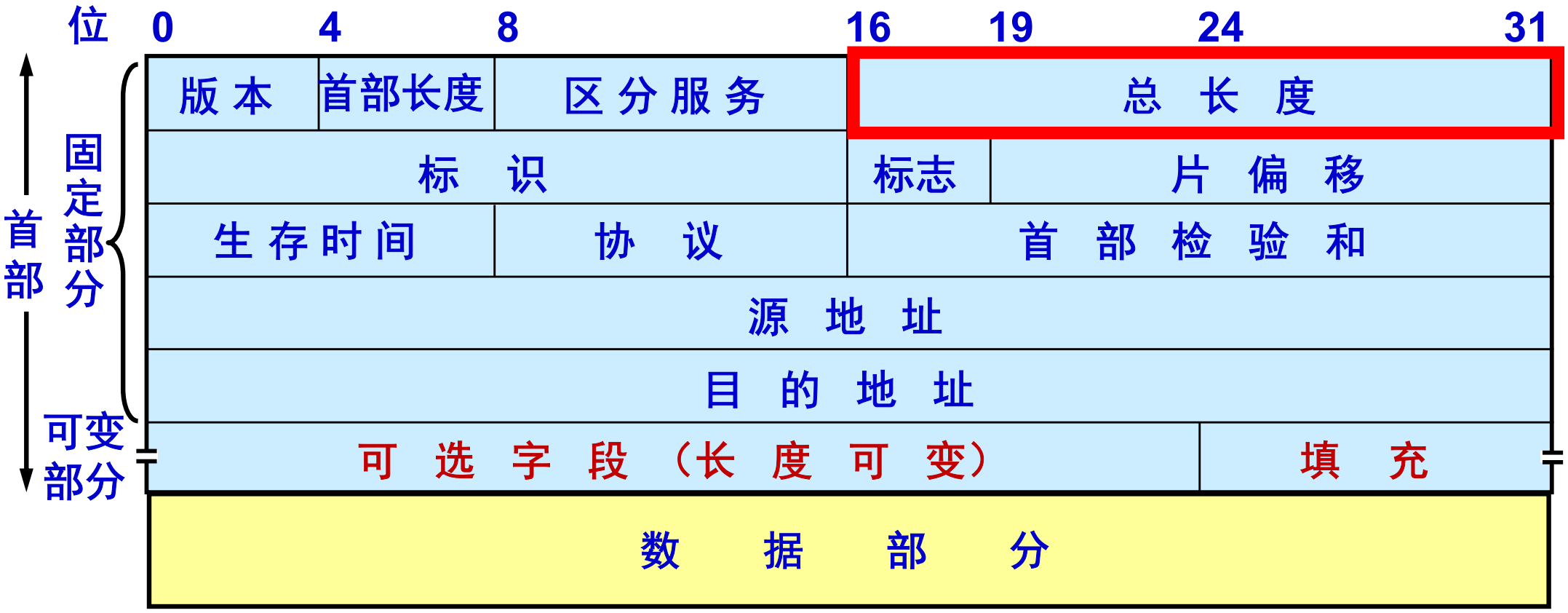
1. IP 数据报首部的固定部分中的各字段

首部长度——占 4 位，可表示的最大数值是 15 个单位(一个单位为 4 字节)，
因此 IP 的首部长度的最大值是 60 字节。

1. IP 数据报首部的固定部分中的各字段

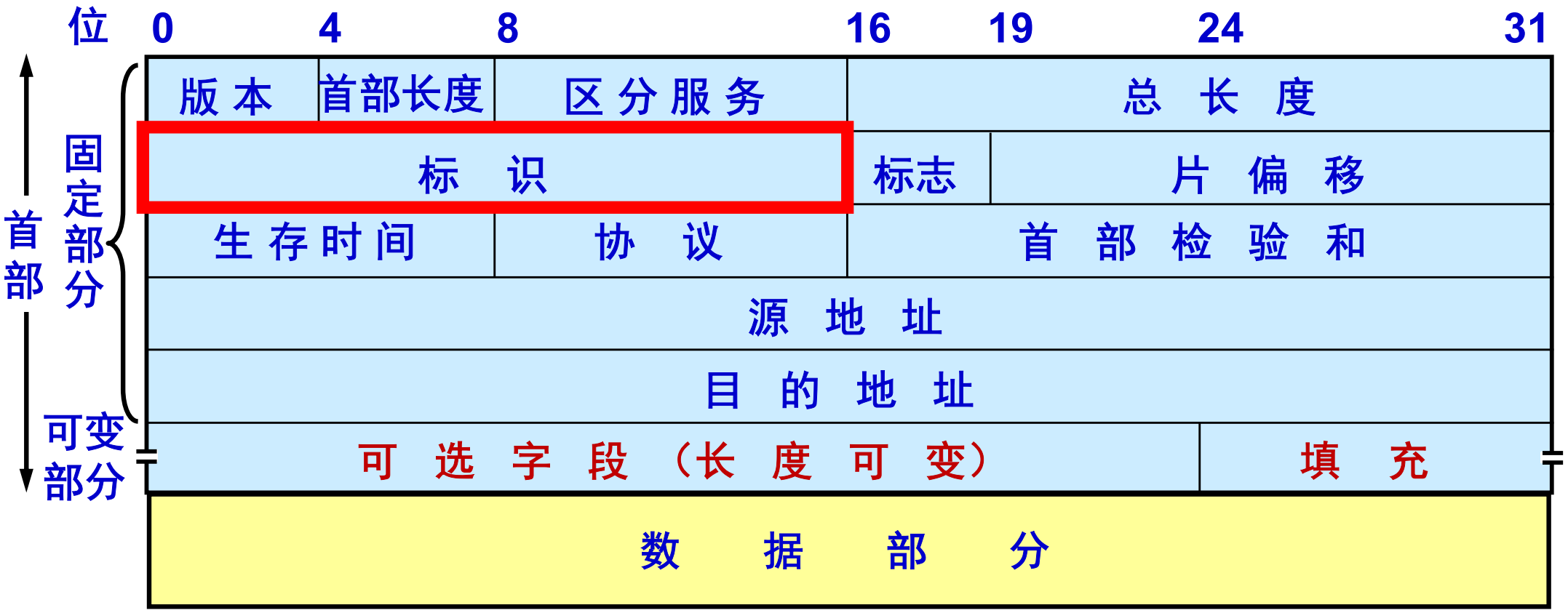
区分服务——占 8 位，用来获得更好的服务。
在旧标准中叫做服务类型，但实际上一直未被使用过。
1998 年这个字段改名为区分服务。
只有在使用区分服务（DiffServ）时，这个字段才起作用。
在一般的情况下都不使用这个字段

1. IP 数据报首部的固定部分中的各字段



总长度——占 16 位，指首部和数据之和的长度，
单位为字节，因此数据报的最大长度为 65535 字节。
总长度必须不超过最大传送单元 MTU。

1. IP 数据报首部的固定部分中的各字段



标识(identification) ——占 16 位，
它是一个计数器，用来产生 IP 数据报的标识。

1. IP 数据报首部的固定部分中的各字段

标志(flag) ——占 3 位，目前只有前两位有意义。

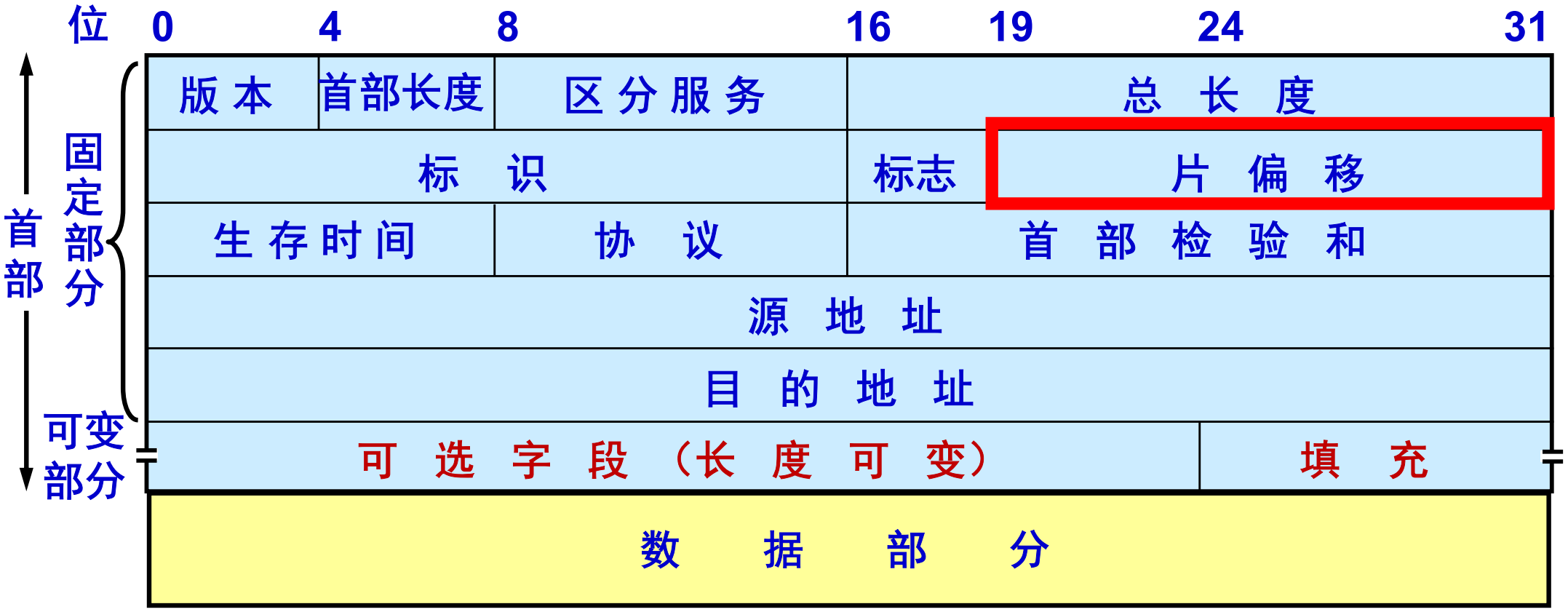
标志字段的最低位是 MF (More Fragment)。

MF = 1 表示后面“还有分片”。MF = 0 表示最后一个分片。

标志字段中间的一位是 DF (Don't Fragment) 。

只有当 DF = 0 时才允许分片。

1. IP 数据报首部的固定部分中的各字段



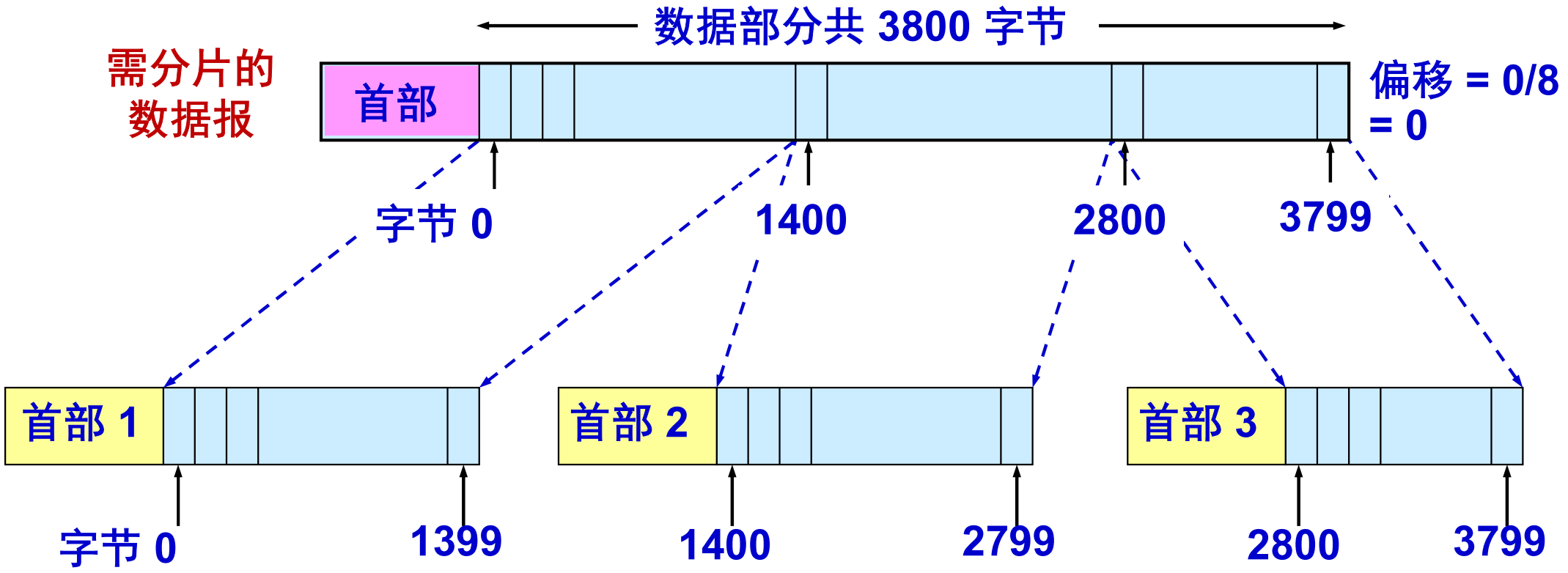
片偏移—— 占13 位，指出：较长的分组在分片后某片在原分组中的相对位置。

片偏移以 8 个字节为偏移单位。

【例4-1】 IP 数据报分片

- 一数据报的总长度为 3820 字节，其数据部分的长度为 3800 字节（使用固定首部），需要分片为长度不超过 1420 字节的数据报片。
- 因固定首部长度为 20 字节，因此每个数据报片的数据部分长度不能超过 1400 字节。
- 于是分为 3 个数据报片，其数据部分的长度分别为 1400、1400 和 1000 字节。
- 原始数据报首部被复制为各数据报片的首部，但必须修改有关字段的值。

【例4-1】 IP 数据报分片



数据报片 1
偏移 = 0/8 = 0

数据报片 2
偏移 = 1400/8 = 175

数据报片 3
偏移 = 2800/8 = 350

【例4-1】 IP 数据报分片

IP 数据报首部中与分片有关的字段中的数值

	总长度	标识	MF	DF	片偏移
原始数据报	3820	12345	0	0	0
数据报片1	1420	12345	1	0	0
数据报片2	1420	12345	1	0	175
数据报片3	1020	12345	0	0	350

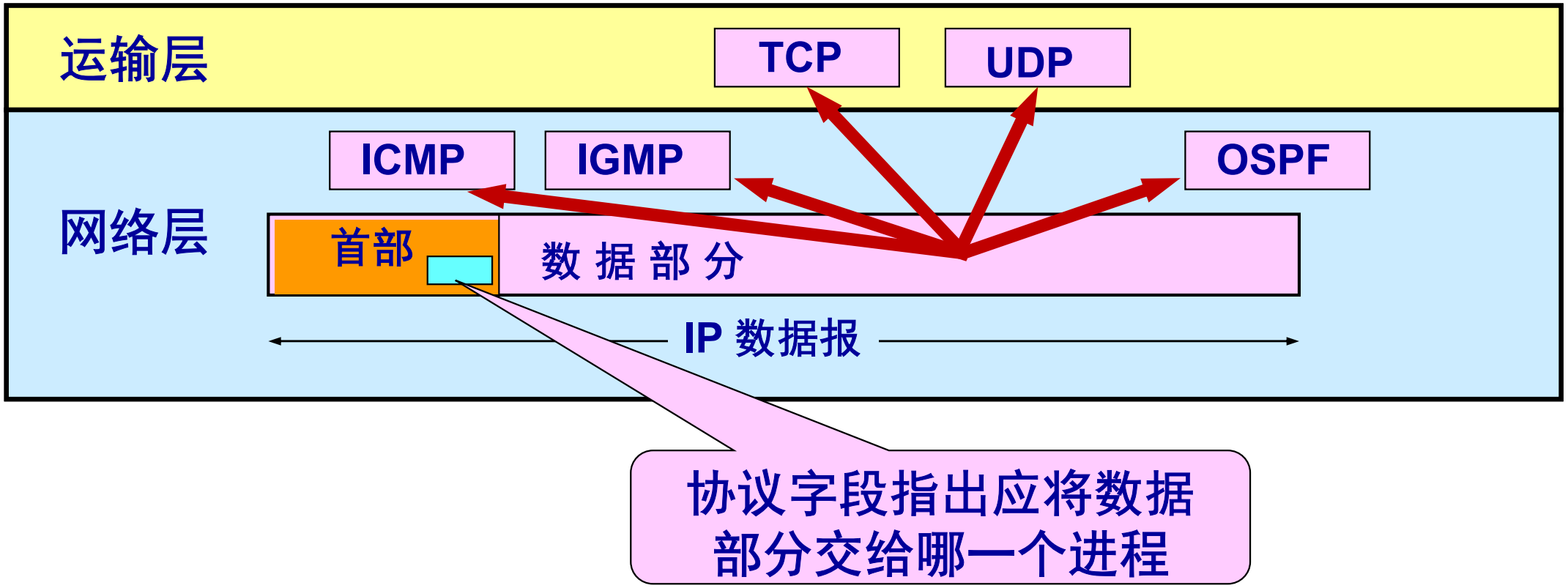
1. IP 数据报首部的固定部分中的各字段

生存时间——占8 位，记为 TTL (Time To Live)，
指示数据报在网络中可通过的路由器数的最大值。

1. IP 数据报首部的固定部分中的各字段

协议——占8 位，指出此数据报携带的数据使用何种协议，
以便目的主机的 IP 层将数据部分
上交给那个处理过程

IP 协议支持多种协议，
IP 数据报可以封装多种协议 PDU。



1. IP 数据报首部的固定部分中的各字段

首部检验和——占16 位，只检验数据报的首部，不检验数据部分。这里不采用 CRC 检验码而采用简单的计算方法。

IP 数据报首部检验和的计算采用 16 位二进制反码求和算法

发送端

数据报首部

字 1 — 16 位

字 2 — 16 位

…

检验和 — 置为全 0

…

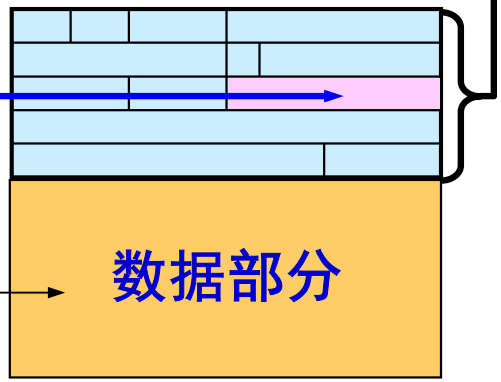
字 n — 16 位

反码算术运算求和 — 16 位

取反码

检验和 — 16 位

IP 数据报



数据部分

数据部分不参与检验和的计算

接收端

字 1 — 16 位

字 2 — 16 位

…

检验和 — 16 位

…

字 n — 16 位

反码算术运算求和 — 16 位

取反码

结果 — 16 位

若结果为 0, 则保留；
否则，丢弃该数据报

1. IP 数据报首部的固定部分中的各字段

源地址和目的地址都各占 4 字节

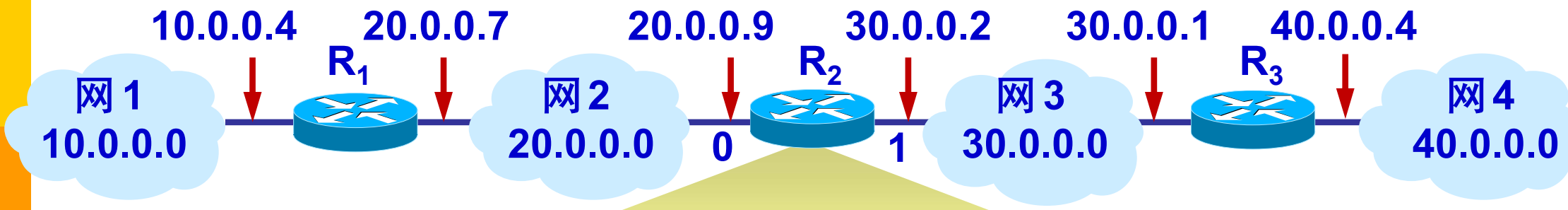
2. IP 数据报首部的可变部分

- IP 首部的可变部分就是一个选项字段，用来支持排错、测量以及安全等措施，内容很丰富。
- 选项字段的长度可变，从 1 个字节到 40 个字节不等，取决于所选择的项目。
- 增加首部的可变部分是为了增加 IP 数据报的功能，但这同时也使得 IP 数据报的首部长度成为可变的。这就增加了每一个路由器处理数据报的开销。
- 实际上这些选项很少被使用。

4.2.6 IP 层转发分组的流程

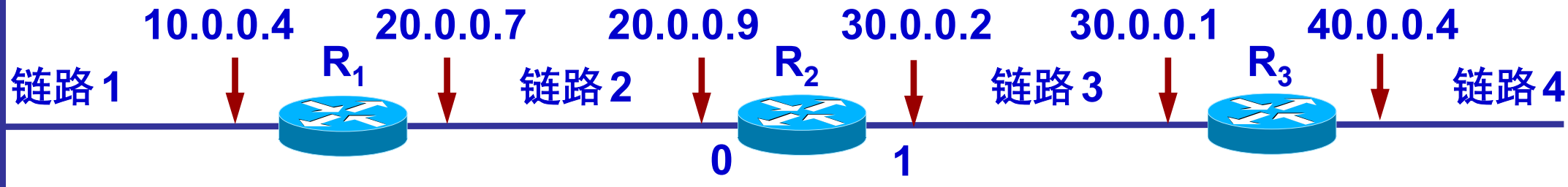
- 假设：有四个 A 类网络通过三个路由器连接在一起。每一个网络上都可能有成千上万个主机。
- 可以想像，若按目的主机号来制作路由表，每一个路由表就有 4 万个项目，即 4 万行（每一行对应于一台主机），则所得出的路由表就会过于庞大。
- 但若按主机所在的网络地址来制作路由表，那么每一个路由器中的路由表就只包含 4 个项目（每一行对应于一个网络），这样就可使路由表大大简化。

在路由表中，对每一条路由，最主要的是（目的网络地址，下一跳地址）



路由器 R_2 的路由表

目的主机所在的网络	下一跳地址
20.0.0.0	直接交付，接口 0
30.0.0.0	直接交付，接口 1
10.0.0.0	20.0.0.7
40.0.0.0	30.0.0.1



查找路由表

根据目的网络地址就能确定下一跳路由器，这样做的结果是：

- IP 数据报最终一定可以找到目的主机所在目的网络上的路由器（可能要通过多次的间接交付）。
- 只有到达最后一个路由器时，才试图向目的主机进行直接交付。

特定主机路由

- 虽然互联网所有的分组转发都是基于目的主机所在的网络，但在大多数情况下都允许有这样的特例，即为特定的目的主机指明一个路由。
- 采用特定主机路由可使网络管理人员能更方便地控制网络和测试网络，同时也可在需要考虑某种安全问题时采用这种特定主机路由。

默认路由 (default route)

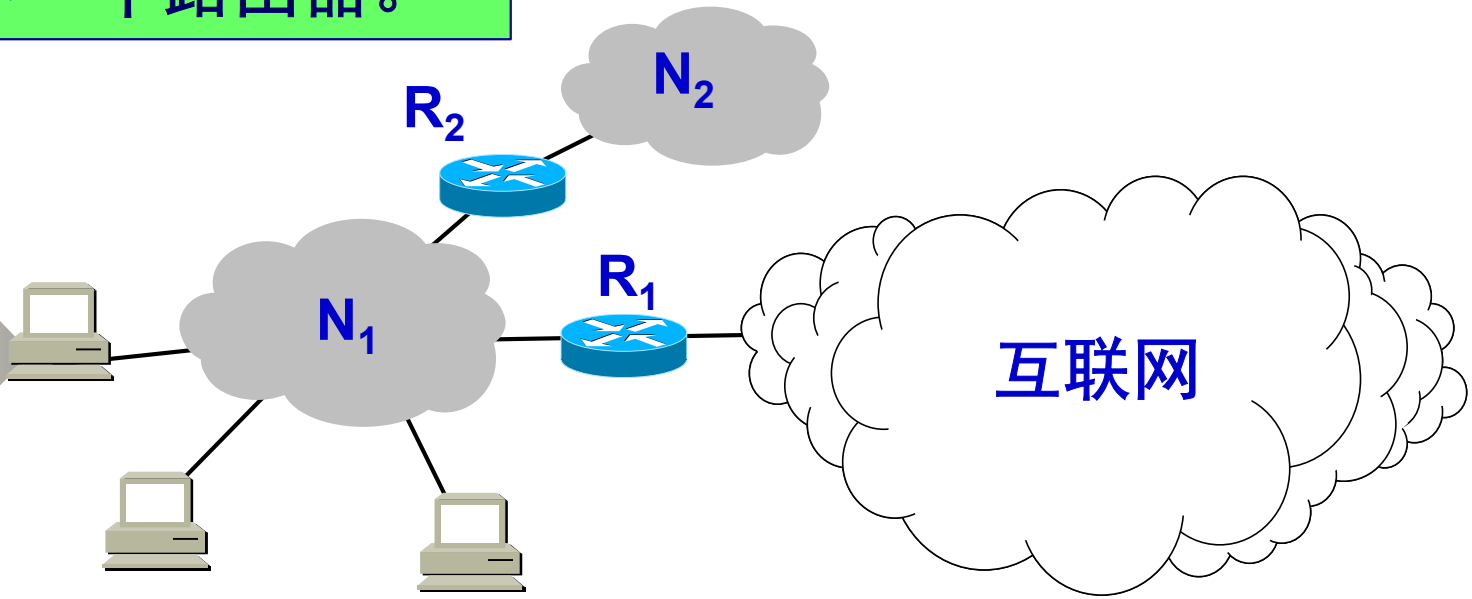
- 路由器还可采用默认路由以减少路由表所占用的空间和搜索路由表所用的时间。
- 这种转发方式在一个网络只有很少的对外连接时是很有用的。
- 默认路由在主机发送 IP 数据报时往往更能显示出它的好处。
- 如果一个主机连接在一个小网络上，而这个网络只用一个路由器和互联网连接，那么在这种情况下使用默认路由是非常合适的。

默认路由举例

只要目的网络不是 N_1 和 N_2，
就一律选择默认路由，
把数据报先间接交付路由器 R_1，
让 R_1 再转发给下一个路由器。

路由表

目的网络	下一跳
N_1	直接
N_2	R_2
默认	R_1



路由器 R_1 充当网络 N_1 的默认路由器

必须强调指出

- IP 数据报的首部中没有地方可以用来指明“下一跳路由器的 IP 地址”。
- 当路由器收到待转发的数据报，不是将下一跳路由器的IP 地址填入 IP 数据报，而是送交下层的网络接口软件。
- 网络接口软件使用 ARP 负责将下一跳路由器的 IP 地址转换成硬件地址，并将此硬件地址放在链路层的 MAC 帧的首部，然后根据这个硬件地址找到下一跳路由器。

路由器分组转发算法

(1) 从数据报的首部提取目的主机的 IP 地址 D，得出目的网络地址为 N。

(2) 若网络 *N* 与此路由器直接相连，则把数据报直接交付目的主机 *D*；否则是间接交付，执行(3)。

(3) 若路由表中有目的地址为 *D* 的特定主机路由，则把数据报传送给路由表中所指明的下一跳路由器；否则，执行(4)。

(4) 若路由表中有到达网络 *N* 的路由，则把数据报传送给路由表指明的下一跳路由器；否则，执行(5)。

(5) 若路由表中有一个默认路由，则把数据报传送给路由表中所指明的默认路由器；否则，执行(6)。

(6) 报告转发分组出错。

关于路由表

- 路由表没有给分组指明到某个网络的完整路径。
- 路由表指出，到某个网络应当先到某个路由器（即下一跳路由器）。
- 在到达下一跳路由器后，再继续查找其路由表，知道再下一步应当到哪一个路由器。
- 这样一步一步地查找下去，直到最后到达目的网络。

4.3 划分子网和构造超网

- 4.3.1 划分子网
- 4.3.2 使用子网时分组的转发
- 4.3.3 无分类编址 CIDR（构造超网）

4.3.1 划分子网

1. 从两级 IP 地址到三级 IP 地址

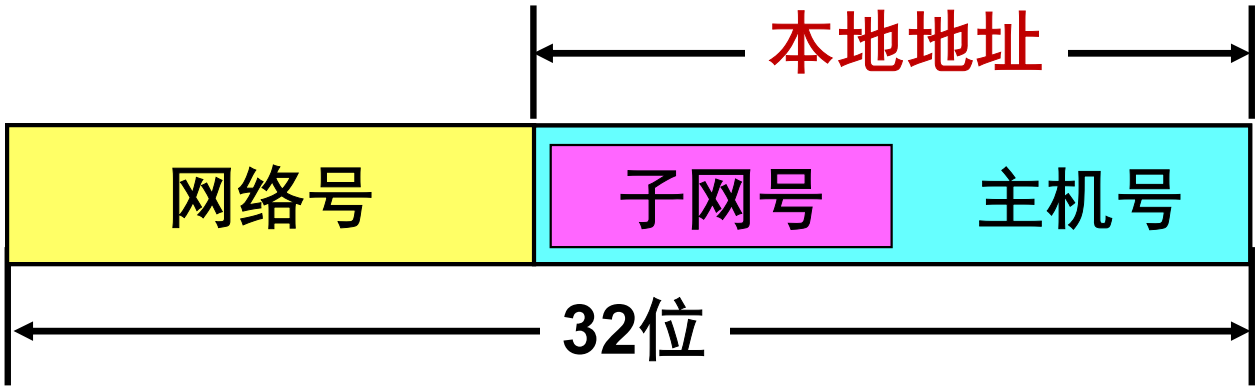
- 在 ARPANET 的早期，IP 地址的设计确实不够合理：
  - (1) IP 地址空间的利用率有时很低。
  - (2) 给每一个物理网络分配一个网络号会使路由表变得太大因而使网络性能变坏。
  - (3) 两级的 IP 地址不够灵活。

三级 IP 地址

- 从 1985 年起在 IP 地址中又增加了一个“子网号字段”，使两级的 IP 地址变成为三级的 IP 地址。
- 这种做法叫作划分子网 (subnetting) 。
- 划分子网已成为互联网的正式标准协议。

划分子网的基本思路

- 划分子网纯属一个单位内部的事情。单位对外仍然表现为没有划分子网的网络。
- 从主机号借用若干个位作为子网号 **subnet-id**，而主机号 **host-id** 也就相应减少了若干个位。

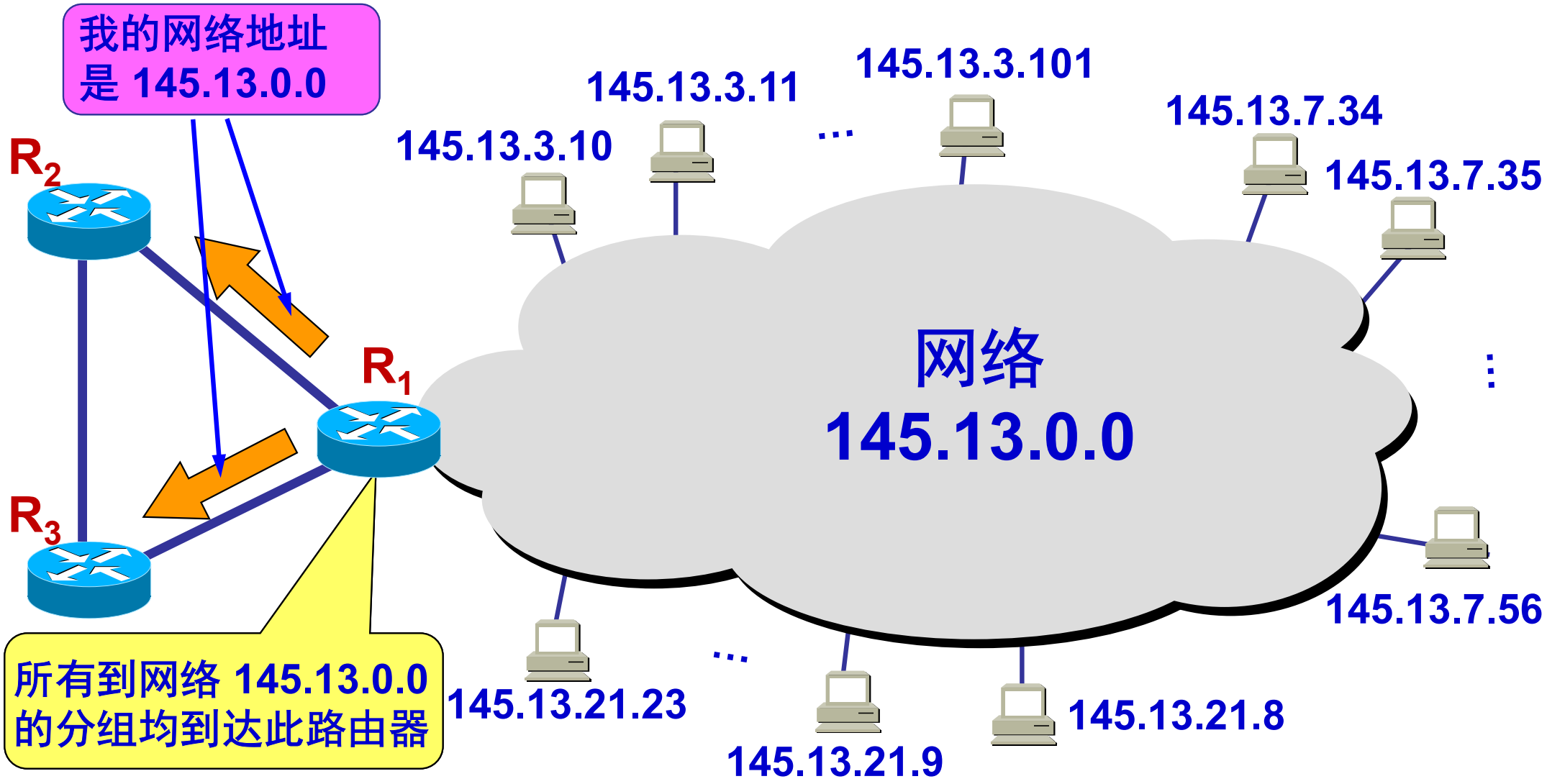


IP地址 ::= {<网络号>, <子网号>, <主机号>}　　(4-2)

划分子网的基本思路（续）

- 凡是从其他网络发送给本单位某个主机的 IP 数据报，仍然是根据 IP 数据报的目的网络号 net-id，先找到连接在本单位网络上的路由器。
- 然后此路由器在收到 IP 数据报后，再按目的网络号 net-id 和子网号 subnet-id 找到目的子网。
- 最后就将 IP 数据报直接交付目的主机。

一个未划分子网的 B 类网络145.13.0.0



划分为三个子网后对外仍是一个网络

所有到达网络145.13.0.0的分组均到达此路由器

R_2

R_3

R_1

145.13.3.10

145.13.3.11

…

145.13.3.101

子网 145.13.3.0

145.13.7.34

145.13.7.35

子网 145.13.7.0

⋮

145.13.7.56

子网 145.13.21.0

…

145.13.21.23

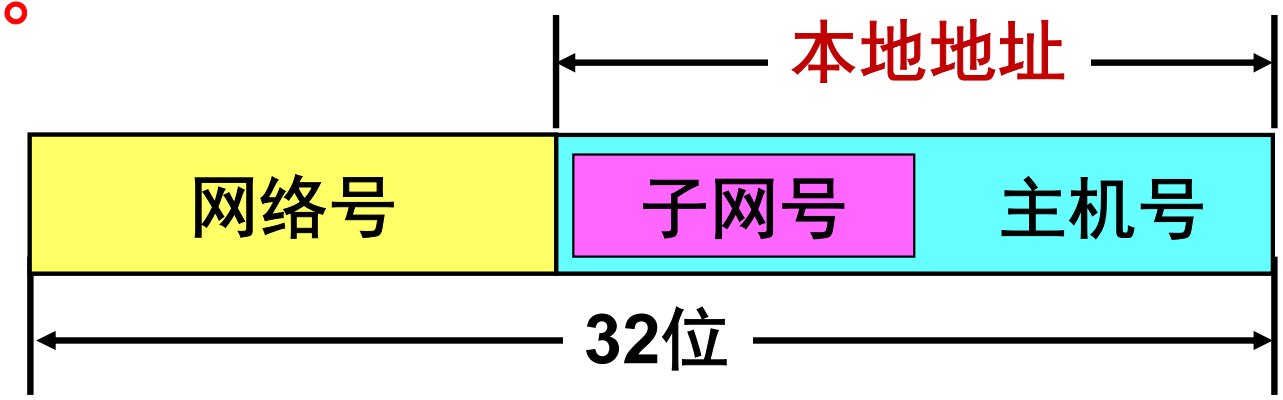
145.13.21.9

145.13.21.8

网络 145.13.0.0

划分子网后变成了三级结构

- 当没有划分子网时，IP 地址是两级结构。
- 划分子网后 IP 地址就变成了三级结构。
- 划分子网只是把 IP 地址的主机号 host-id 这部分进行再划分，而不改变 IP 地址原来的网络号 net-id。



划分子网后变成了三级结构

- 优点
  - 减少了 IP 地址的浪费
  - 使网络的组织更加灵活
  - 更便于维护和管理
- 划分子网纯属一个单位内部的事情，对外部网络透明，对外仍然表现为没有划分子网的一个网络。

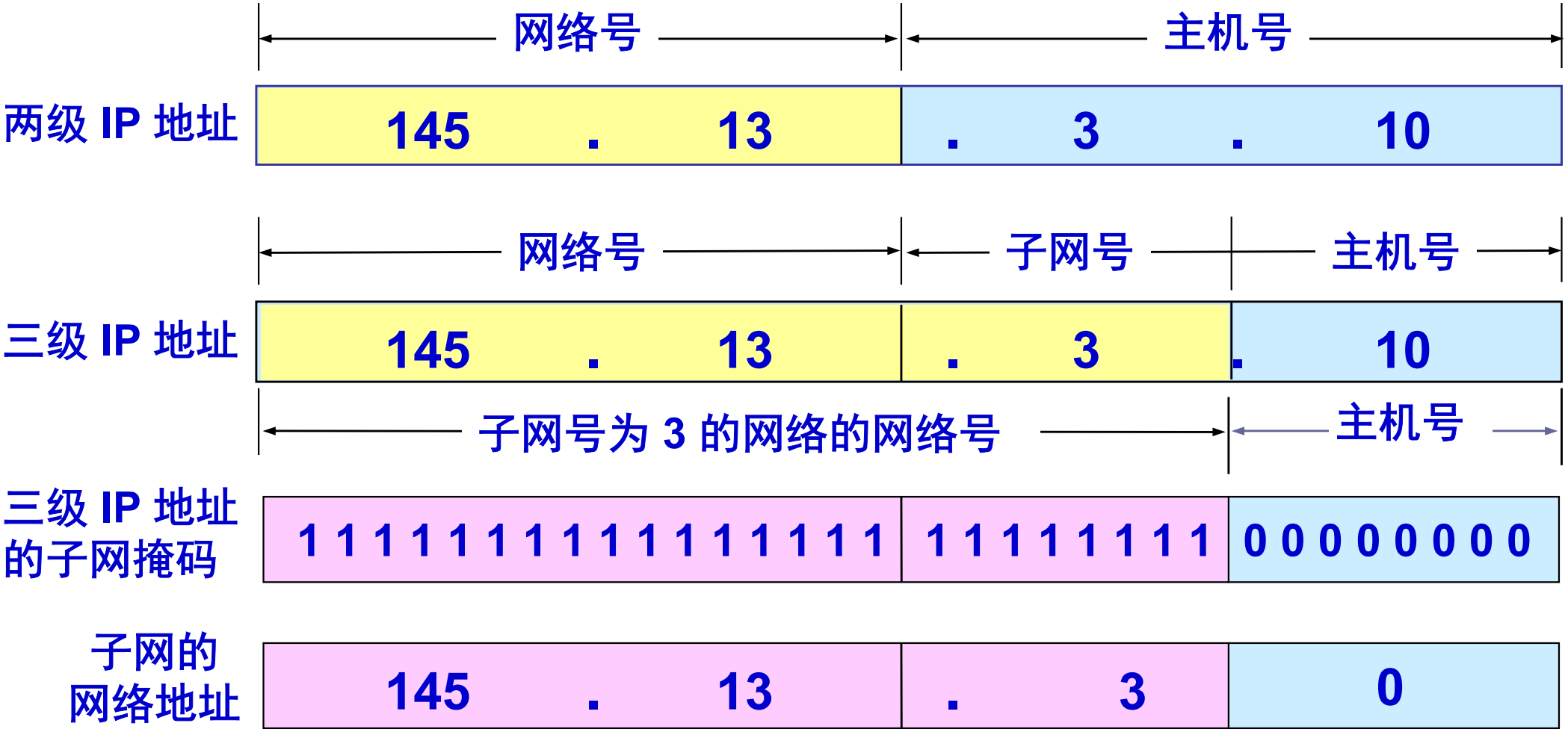
2. 子网掩码

- 从一个 IP 数据报的首部并无法判断源主机或目的主机所连接的网络是否进行了子网划分。
- 使用子网掩码(subnet mask)可以找出 IP 地址中的子网部分。

规则：

- 子网掩码长度＝32位
- 某位＝1：IP地址中的对应位为网络号和子网号
- 某位＝0：IP地址中的对应位为主机号

IP 地址的各字段和子网掩码



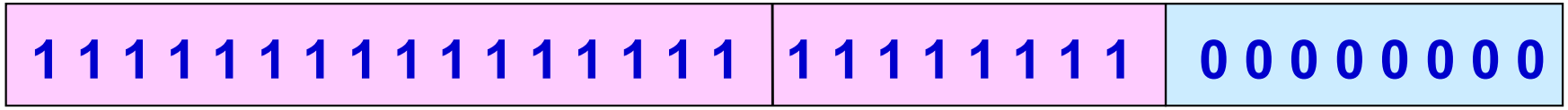
(IP 地址) AND (子网掩码) =网络地址

两级 IP 地址

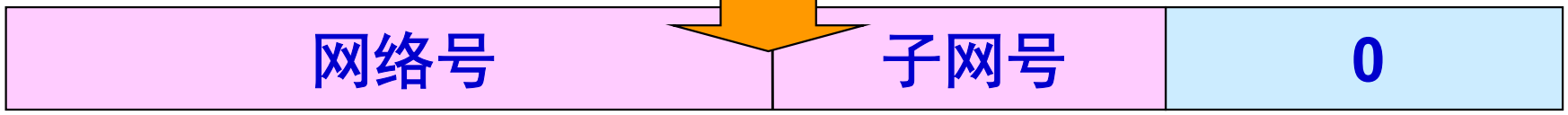
三级 IP 地址

逐位进行 AND 运算

三级 IP 地址
的子网掩码



子网的
网络地址



默认子网掩码

A类地址	网络地址	网络号	主机号为全 0
	默认子网掩码 255.0.0.0	1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
B类地址	网络地址	网络号	主机号为全 0
	默认子网掩码 255.255.0.0	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
C类地址	网络地址	网络号	主机号为全 0
	默认子网掩码 255.255.255.0	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0

子网掩码是一个重要属性

- 子网掩码是一个网络或一个子网的重要属性。
- 路由器在和相邻路由器交换路由信息时，必须把自己所在网络（或子网）的子网掩码告诉相邻路由器。
- 路由器的路由表中的每一个项目，除了要给出目的网络地址外，还必须同时给出该网络的子网掩码。
- 若一个路由器连接在两个子网上就拥有两个网络地址和两个子网掩码。

子网划分方法

- 有固定长度子网和变长子网两种子网划分方法。
- 在采用固定长度子网时，所划分的所有子网的子网掩码都是相同的。
- 虽然根据已成为互联网标准协议的RFC 950文档，子网号不能为全1或全0，但随着无分类域间路由选择CIDR的广泛使用，现在全1和全0的子网号也可以使用了，但一定要谨慎使用，确认你的路由器所用的路由选择软件是否支持全0或全1的子网号这种较新的用法。
- 划分子网增加了灵活性，但却减少了能够连接在网络上的主机总数。

B 类地址的子网划分选择（使用固定长度子网）

子网号的位数	子网掩码	子网数	每个子网的主机数
2	255.255.192.0	2	16382
3	255.255.224.0	6	8190
4	255.255.240.0	14	4094
5	255.255.248.0	30	2046
6	255.255.252.0	62	1022
7	255.255.254.0	126	510
8	255.255.255.0	254	254
9	255.255.255.128	510	126
10	255.255.255.192	1022	62
11	255.255.255.224	2046	30
12	255.255.255.240	4094	14
13	255.255.255.248	8190	6
14	255.255.255.252	16382	2

表中的“子网号的位数”中没有0, 1, 15和16这四种情况，因为这没有意义。

【例4-2】已知 IP 地址是 141.14.72.24，子网掩码是 255.255.192.0。试求网络地址。

(a) 点分十进制表示的 IP 地址

141 . 14 . 72 . 24

(b) IP 地址的第 3 字节是二进制

141 . 14 .01001000. 24

(c) 子网掩码是 255.255.192.0

11111111 11111111 11000000 00000000

(d) IP 地址与子网掩码逐位相与

141 . 14 .01000000. 0

(e) 网络地址（点分十进制表示）

141 . 14 . 64 . 0

【例4-3】上例中，若子网掩码改为255.255.224.0，试求网络地址，讨论所得结果。

(a) 点分十进制表示的 IP 地址　141 . 14 . 72 . 24

(b) IP 地址的第 3 字节是二进制　141 . 14 .01001000. 24

(c) 子网掩码是 255.255.224.0　11111111 11111111 11100000 00000000

(d) IP 地址与子网掩码逐位相与　141 . 14 .01000000. 0

(e) 网络地址（点分十进制表示）　141 . 14 . 64 . 0

不同的子网掩码得出相同的网络地址。
但不同的掩码的效果是不同的。

4.3.2 使用子网时分组的转发

- 在不划分子网的两级 IP 地址下，从 IP 地址得出网络地址是个很简单的事。
- 但在划分子网的情况下，从 IP 地址却不能唯一地得出网络地址来，这是因为网络地址取决于那个网络所采用的子网掩码，但数据报的首部并没有提供子网掩码的信息。
- 因此分组转发的算法也必须做相应的改动。

在划分子网情况下路由器转发分组的算法

(1) 从收到的分组的首部提取目的 IP 地址 D。

(2) 先用各网络的子网掩码和 *D* 逐位相“与”，看是否和相应的网络地址匹配。若匹配，则将分组直接交付。否则就是间接交付，执行 (3)。

(3) 若路由表中有目的地址为 *D* 的特定主机路由，则将分组传送给指明的下一跳路由器；否则，执行 (4)。

(4) 对路由表中的每一行，将子网掩码和 *D* 逐位相“与”。若结果与该行的目的网络地址匹配，则将分组传送给该行指明的下一跳路由器；否则，执行 (5)。

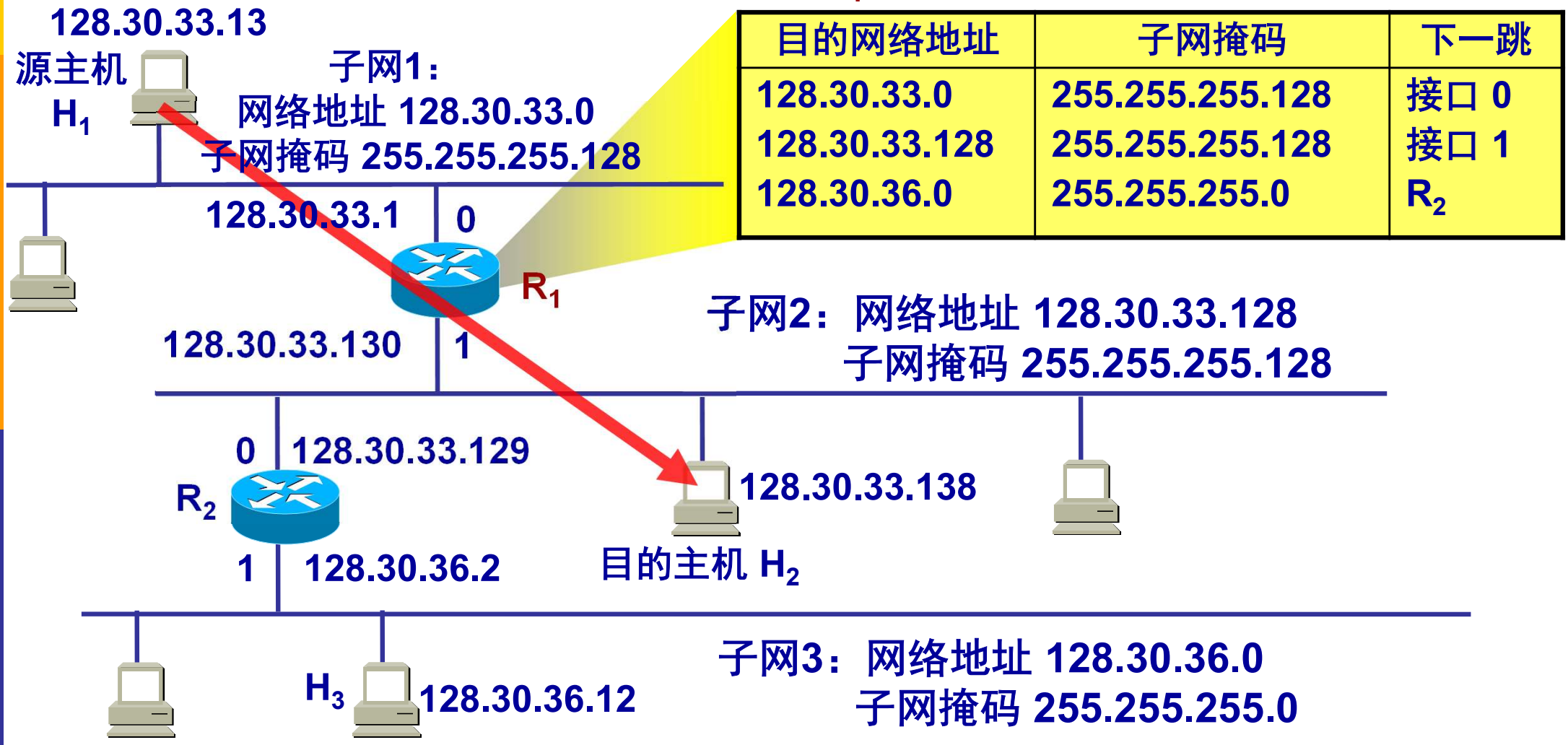
(5) 若路由表中有一个默认路由，则将分组传送给路由表中所指明的默认路由器；否则，执行 (6)。

(6) 报告转发分组出错。

【例4-4】已知互联网和路由器 R_1 中的路由表。主机 H_1 向 H_2 发送分组。试讨论 R_1 收到 H_1 向 H_2 发送的分组后查找路由表的过程。

R_1 的路由表（未给出默认路由器）

目的网络地址	子网掩码	下一跳
128.30.33.0	255.255.255.128	接口 0
128.30.33.128	255.255.255.128	接口 1
128.30.36.0	255.255.255.0	R_2

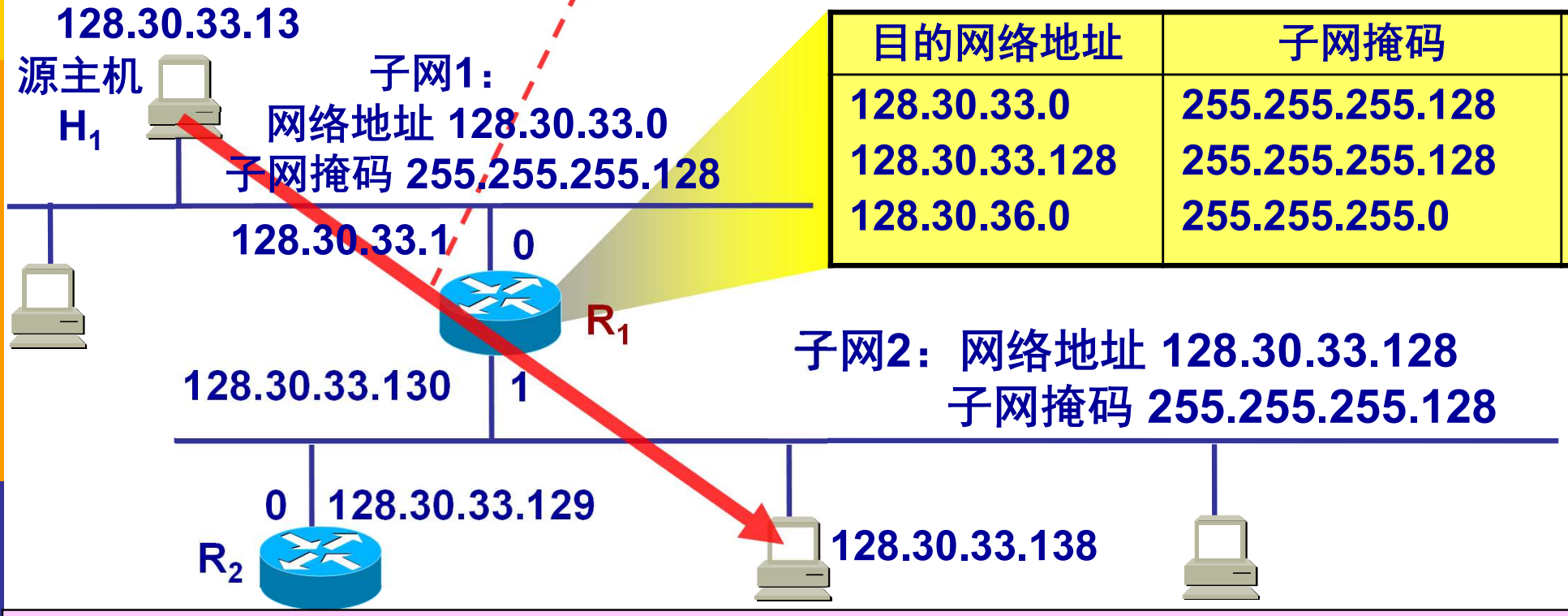


主机 H_1 要发送分组给 H_2

要发送的分组的目的 IP 地址：128.30.33.138

R_1 的路由表（未给出默认路由器）

目的网络地址	子网掩码	下一跳
128.30.33.0	255.255.255.128	接口 0
128.30.33.128	255.255.255.128	接口 1
128.30.36.0	255.255.255.0	R_2



因此 H_1 首先检查主机 128.30.33.138 是否连接在本网络上

如果是，则直接交付；

否则，就送交路由器 R_1，并逐项查找路由表。

主机 H_1 首先将
本子网的子网掩码 255.255.255.128
与分组的目的 IP 地址 128.30.33.138 相“与”(AND 操作)

255.255.255.128 AND 128.30.33.138 的计算

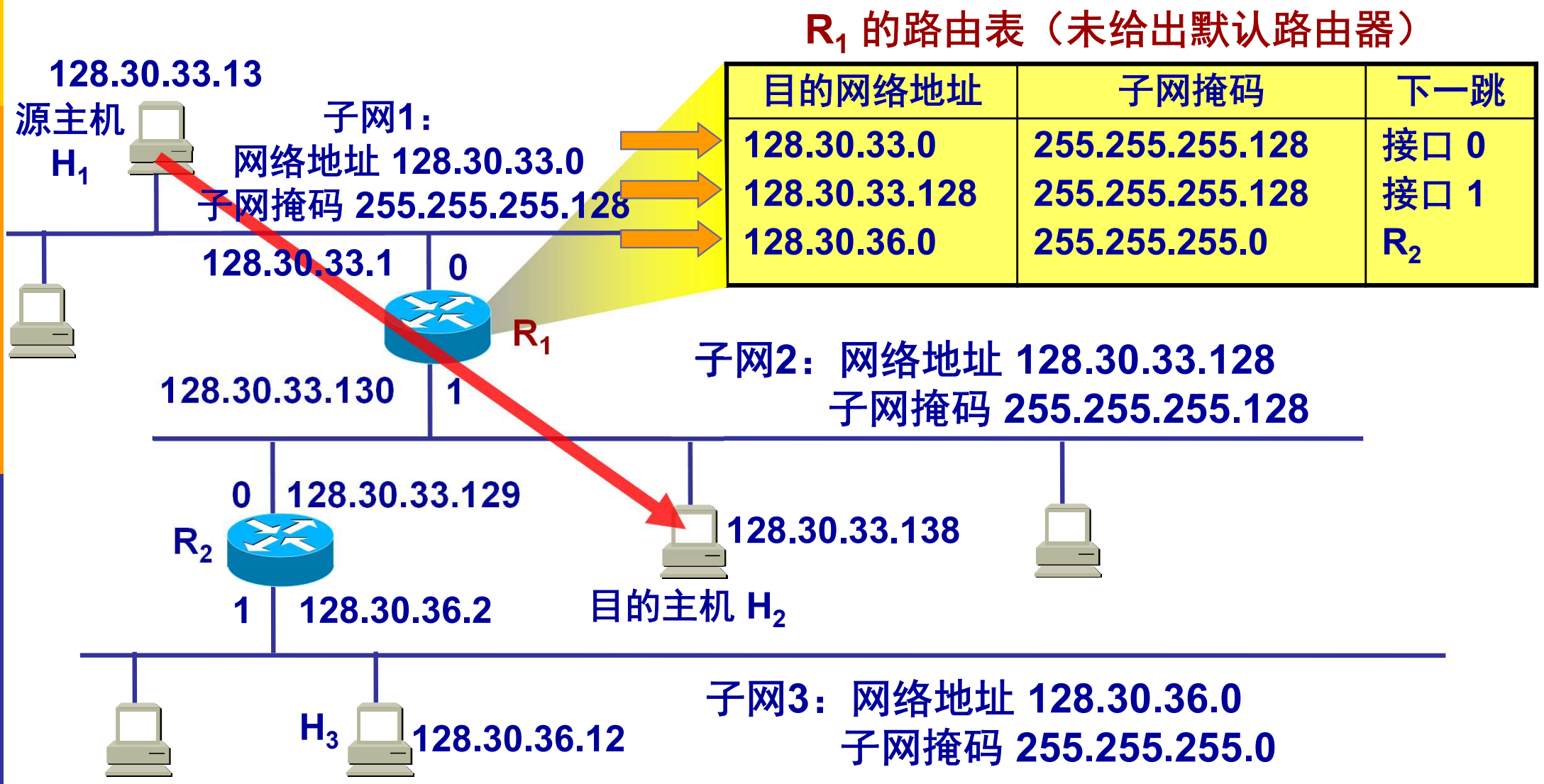
255 就是二进制的全 1，因此 255 AND xyz = xyz，
这里只需计算最后的 128 AND 138 即可。

128 → 10000000
138 → 10001010
逐比特 AND 操作后 10000000 → 128

	255.255.255.128
逐比特 AND 操作	128. 30. 33.138
	128. 30. 33.128

≠ H_1 的网络地址

因此 H_1 必须把分组传送到路由器 R_1 然后逐项查找路由表

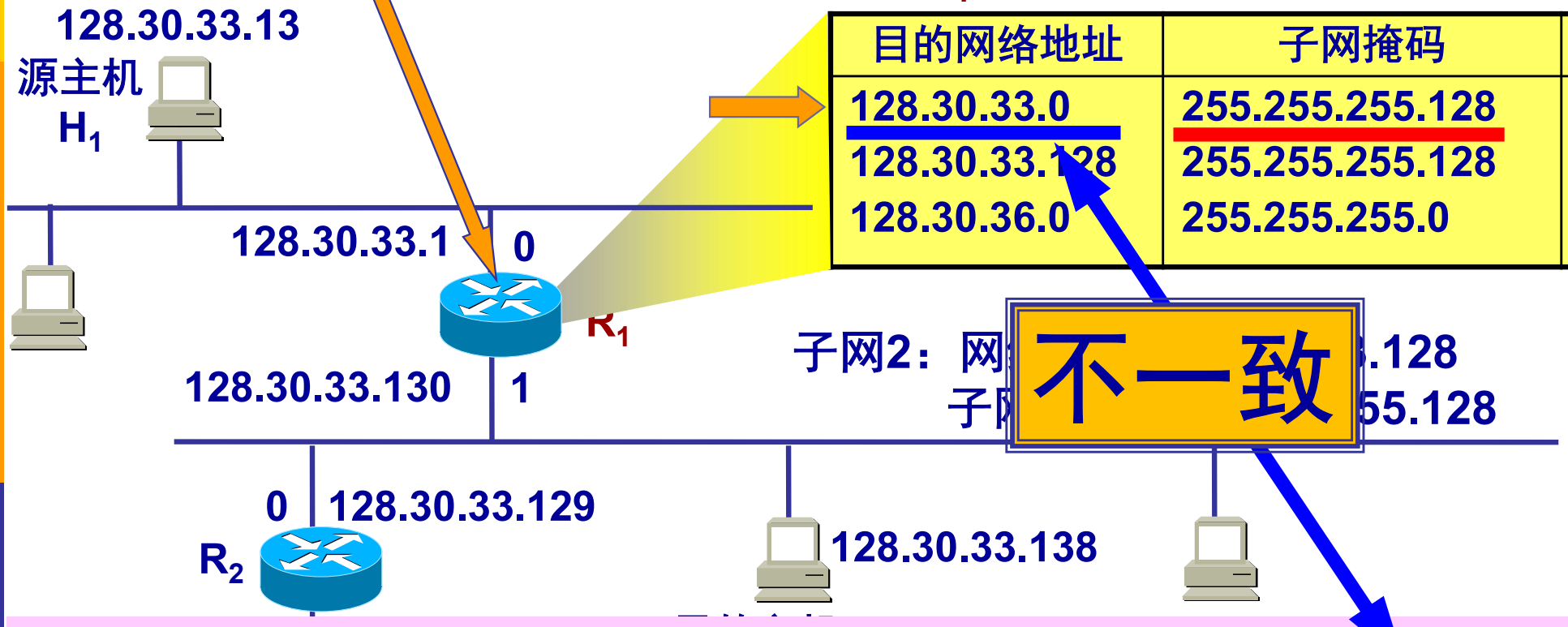


路由器 R_1 收到分组后就用路由表中第 1 个项目的子网掩码和 128.30.33.138 逐比特 AND 操作

R_1 收到的分组的目的 IP 地址：128.30.33.138

R_1 的路由表（未给出默认路由器）

目的网络地址	子网掩码	下一跳
128.30.33.0	255.255.255.128	接口 0
128.30.33.128	255.255.255.128	接口 1
128.30.36.0	255.255.255.0	R_2



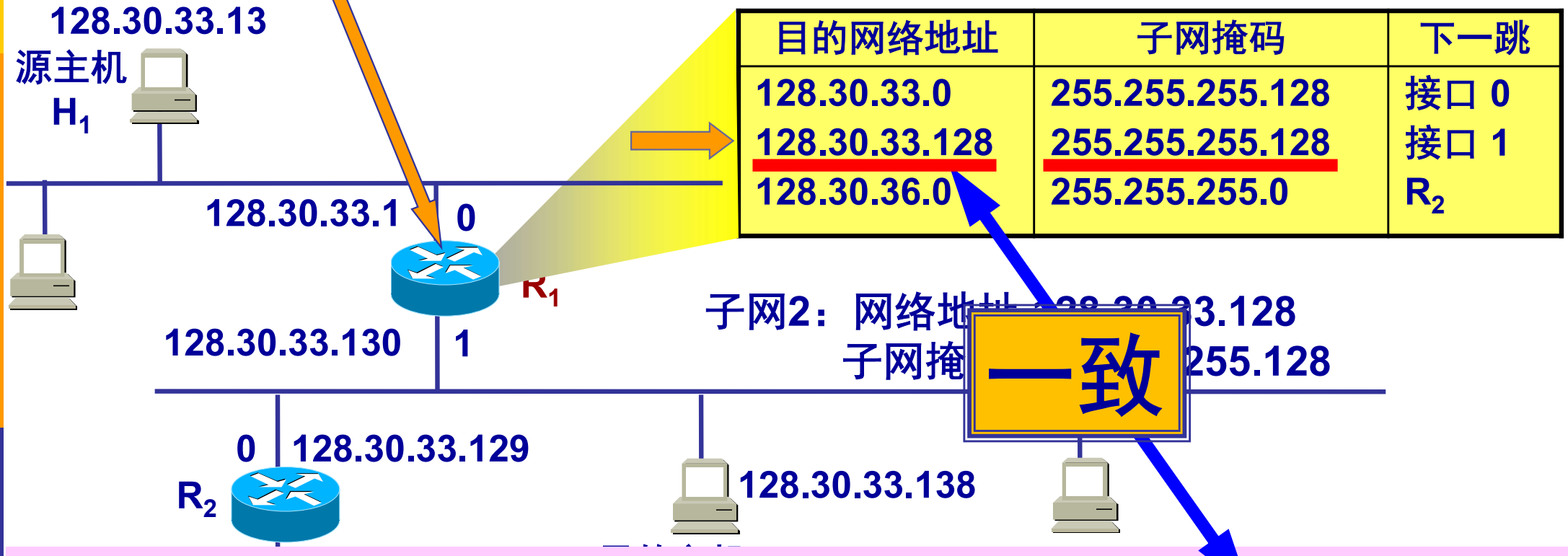
255.255.255.128 AND 128.30.33.138 = 128.30.33.128

不匹配!

（因为128.30.33.128 与路由表中的 128.30.33.0 不一致）

路由器 R_1 收到分组后就用路由表中第 1 个项目的子网掩码和 128.30.33.138 逐比特 AND 操作

R_1 收到的分组的目的 IP 地址：128.30.33.138



R_1 的路由表（未给出默认路由器）

目的网络地址	子网掩码	下一跳
128.30.33.0	255.255.255.128	接口 0
128.30.33.128	255.255.255.128	接口 1
128.30.36.0	255.255.255.0	R_2

255.255.255.128 AND 128.30.33.138 = 128.30.33.128

匹配!

这表明子网 2 就是收到的分组所要寻找的目的网络。

4.3.3 无分类编址 CIDR

1. 网络前缀

划分子网在一定程度上缓解了互联网在发展中遇到的困难。然而在 1992 年互联网仍然面临三个必须尽早解决的问题：

- (1) B 类地址在 1992 年已分配了近一半，眼看就要在 1994 年 3 月全部分配完毕！
- (2) 互联网主干网上的路由表中的项目数急剧增长（从几千个增长到几万个）。
- (3) 整个 IPv4 的地址空间最终将全部耗尽。

IP 编址问题的演进

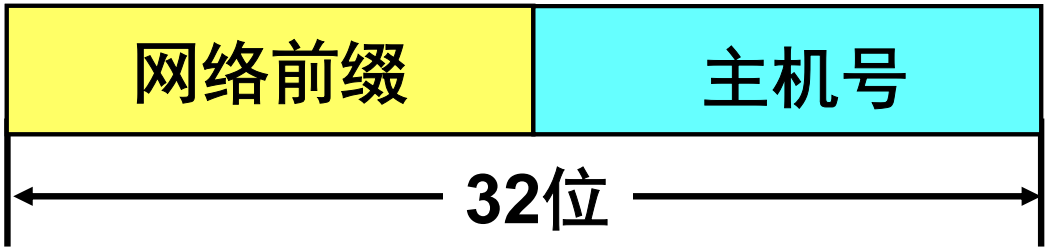
- 1987 年，RFC 1009 就指明了在一个划分子网的网络中可同时使用几个不同的子网掩码。
- 使用变长子网掩码 VLSM (Variable Length Subnet Mask)可进一步提高 IP 地址资源的利用率。
- 在 VLSM 的基础上又进一步研究出无分类编址方法，它的正式名字是无分类域间路由选择 CIDR (Classless Inter-Domain Routing)。

CIDR 最主要的特点

- CIDR 消除了传统的 A 类、B 类和 C 类地址以及划分子网的概念，因而可以更加有效地分配 IPv4 的地址空间。
- CIDR使用各种长度的“网络前缀”(network-prefix)来代替分类地址中的网络号和子网号。
- IP 地址从三级编址（使用子网掩码）又回到了两级编址。

无分类的两级编址

- 无分类的两级编址的记法是：



IP地址 ::= {<网络前缀>, <主机号>}　　(4-3)

- CIDR 使用“斜线记法”(slash notation)，它又称为 CIDR 记法，即在 IP 地址面加上一个斜线“/”，然后写上网络前缀所占的位数（这个数值对应于三级编址中子网掩码中 1 的个数）。例如： 220.78.168.0/24

CIDR 地址块

- CIDR 把网络前缀都相同的连续的 IP 地址组成“CIDR 地址块”。
- 128.14.32.0/20 表示的地址块共有 2^{12} 个地址（因为斜线后面的 20 是网络前缀的位数，所以这个地址的主机号是 12 位）。
  - 这个地址块的起始地址是 128.14.32.0。
  - 在不需要指出地址块的起始地址时，也可将这样的地址块简称为“/20 地址块”。
  - 128.14.32.0/20 地址块的最小地址：128.14.32.0
  - 128.14.32.0/20 地址块的最大地址：128.14.47.255
  - 全 0 和全 1 的主机号地址一般不使用。

128.14.32.0/20 表示的地址（2^{12} 个地址）

最小地址 →

10000000 00001110 00100000 00000000
10000000 00001110 00100000 00000001
10000000 00001110 00100000 00000010
10000000 00001110 00100000 00000011
10000000 00001110 00100000 00000100
10000000 00001110 00100000 00000101

. . .　　. . .

10000000 00001110 00101111 11111011
10000000 00001110 00101111 11111100
10000000 00001110 00101111 11111101
10000000 00001110 00101111 11111110

最大地址 → 10000000 00001110 00101111 11111111

所有地址的 20 位前缀都是一样的

路由聚合 (route aggregation)

- 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，它使得路由表中的一个项目可以表示很多个（例如上千个）原来传统分类地址的路由。
- 路由聚合有利于减少路由器之间的路由选择信息的交换，从而提高了整个互联网的性能。
- 路由聚合也称为构成超网 (supernetting)。
- CIDR 虽然不使用子网了，但仍然使用“掩码”这一名词（但不叫子网掩码）。
- 对于 /20 地址块，它的掩码是 20 个连续的 1。 斜线记法中的数字就是掩码中1的个数。

CIDR 记法的其他形式

- 10.0.0.0/10 可简写为 10/10，也就是把点分十进制中低位连续的 0 省略。
- 10.0.0.0/10 隐含地指出 IP 地址 10.0.0.0 的掩码是 255.192.0.0。此掩码可表示为：

11111111 11000000 00000000 00000000

255 192 0 0

掩码中有 10 个连续的 1

- 网络前缀的后面加一个星号 * 的表示方法，如 00001010 00*，在星号 * 之前是网络前缀，而星号 * 表示 IP 地址中的主机号，可以是任意值。

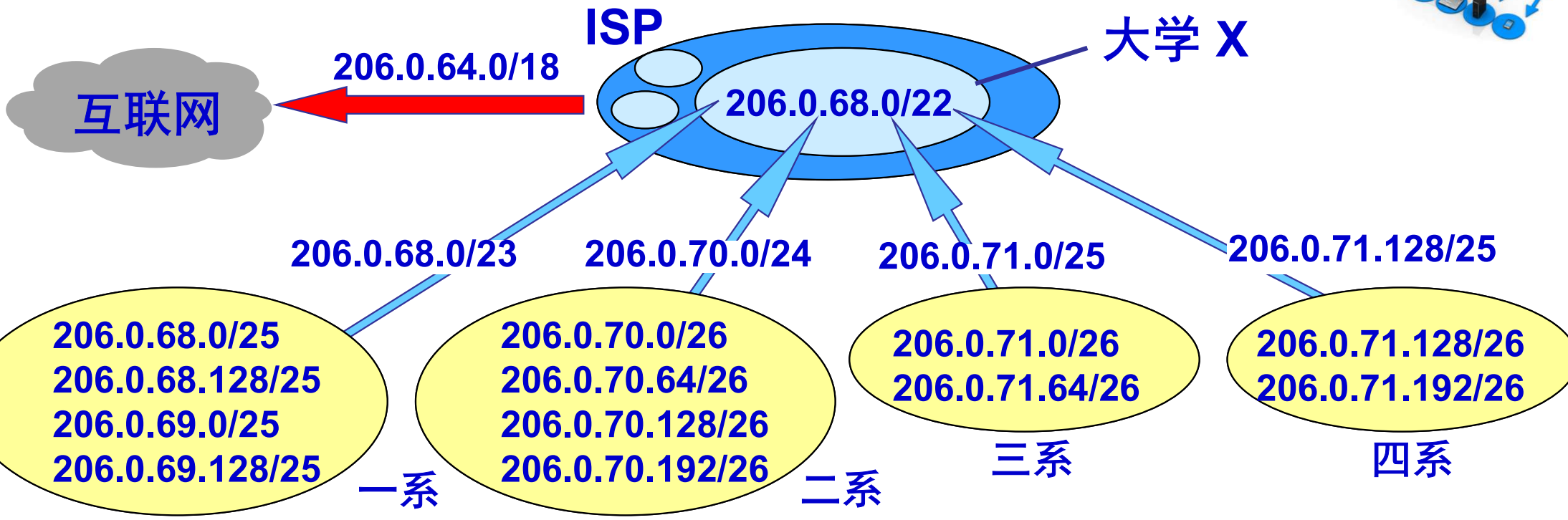
常用的 CIDR 地址块

CIDR 前缀长度	点分十进制	包含的地址数	相当于包含分类的网络数
/13	255.248.0.0	512 K	8个B类或2048个C类
/14	255.252.0.0	256 K	4个B类或1024个C类
/15	255.254.0.0	128 K	2个B类或512个C类
/16	255.255.0.0	64 K	1个B类或256个C类
/17	255.255.128.0	32 K	128个C类
/18	255.255.192.0	16 K	64个C类
/19	255.255.224.0	8 K	32个C类
/20	255.255.240.0	4 K	16个C类
/21	255.255.248.0	2 K	8个C类
/22	255.255.252.0	1 K	4个C类
/23	255.255.254.0	512	2个C类
/24	255.255.255.0	256	1个C类
/25	255.255.255.128	128	1/2个C类
/26	255.255.255.192	64	1/4个C类
/27	255.255.255.224	32	1/8个C类

构成超网

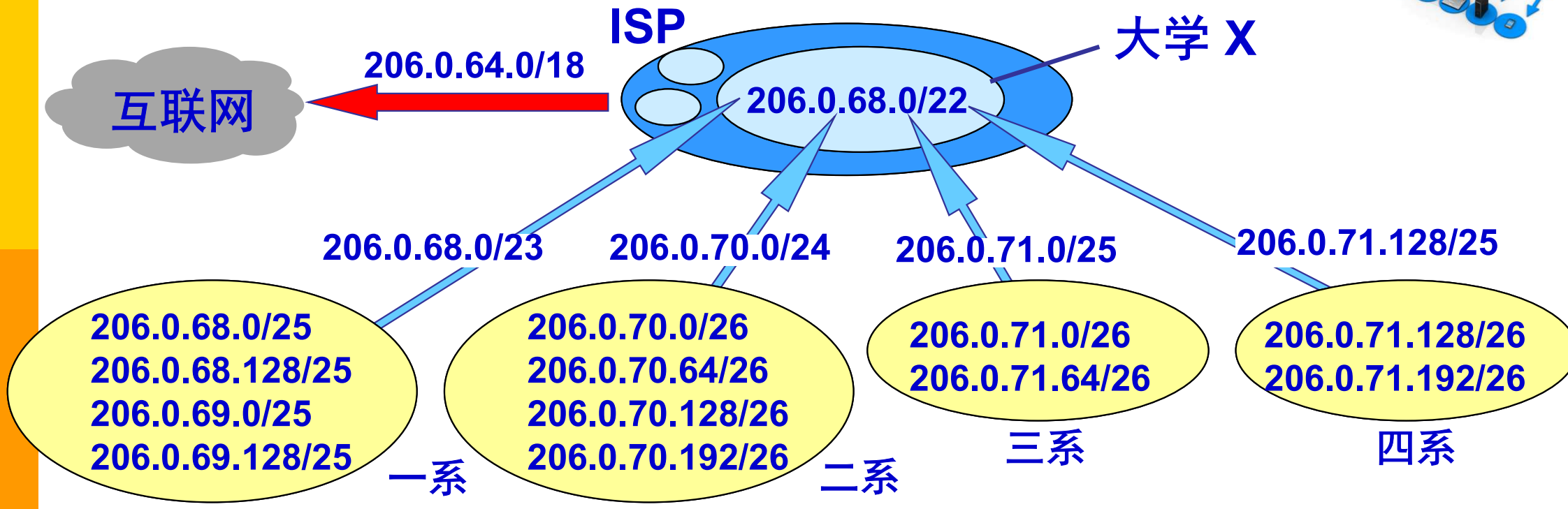
- 前缀长度不超过 23 位的 CIDR 地址块都包含了多个 C 类地址。
- 这些 C 类地址合起来就构成了超网。
- CIDR 地址块中的地址数一定是 2 的整数次幂。
- 网络前缀越短，其地址块所包含的地址数就越多。而在三级结构的IP地址中，划分子网是使网络前缀变长。
- CIDR 的一个好处是：可以更加有效地分配 IPv4 的地址空间，可根据客户的需要分配适当大小的 CIDR 地址块。

CIDR 地址块划分举例



单位	地址块	二进制表示	地址数
ISP	206.0.64.0/18	11001110.00000000.01*	16384
大学	206.0.68.0/22	11001110.00000000.010001*	1024
一系	206.0.68.0/23	11001110.00000000.0100010*	512
二系	206.0.70.0/24	11001110.00000000.01000110.*	256
三系	206.0.71.0/25	11001110.00000000.01000111.0*	128
四系	206.0.71.128/25	11001110.00000000.01000111.1*	128

CIDR 地址块划分举例



这个 ISP 共有 64 个 C 类网络。如果不采用 CIDR 技术，则在与该 ISP 的路由器交换路由信息的每一个路由器的路由表中，就需要有 64 个项目。但采用地址聚合后，只需用路由聚合后的 1 个项目 206.0.64.0/18 就能找到该 ISP。

2. 最长前缀匹配

- 使用 CIDR 时，路由表中的每个项目由“网络前缀”和“下一跳地址”组成。在查找路由表时可能会得到不止一个匹配结果。
- 应当从匹配结果中选择具有最长网络前缀的路由：最长前缀匹配 (longest-prefix matching)。
- 网络前缀越长，其地址块就越小，因而路由就越具体 (more specific) 。
- 最长前缀匹配又称为最长匹配或最佳匹配。

最长前缀匹配举例

收到的分组的目的地址 *D* = 206.0.71.130

路由表中的项目：206.0.68.0/22　　1

206.0.71.128/25　　2

查找路由表中的第 1 个项目：

第 1 个项目 206.0.68.0/22 的掩码 *M* 有 22 个连续的 1。

M = 11111111 11111111 11111100 00000000

因此只需把 *D* 的第 3 个字节转换成二进制。

	M =	11111111	11111111	11111100 00000000
AND	*D* =	206.	0.	01000111. 130
		206.	0.	01000100. 0

与 206.0.68.0/22 匹配!

最长前缀匹配举例

收到的分组的目的地址 *D* = 206.0.71.130

路由表中的项目：206.0.68.0/22　　1

206.0.71.128/25　　2

查找路由表中的第 2 个项目：

第 2 个项目 206.0.71.128/25 的掩码 *M* 有 25 个连续的 1。

M = 11111111 11111111 11111100 00000000

因此只需把 *D* 的第 4 个字节转换成二进制。

	M =	11111111.	11111111.	11111111.	10000000
AND	*D* =	206.	0.	71.	10000010
		206.	0.	71.	10000000

与 206.0.71.128/25 匹配!

最长前缀匹配举例

D AND (11111111 11111111 11111100 00000000)
= 206.0.68.0/22　　匹配

D AND (11111111 11111111 11111111 10000000)
= 206.0.71.128/25　　匹配

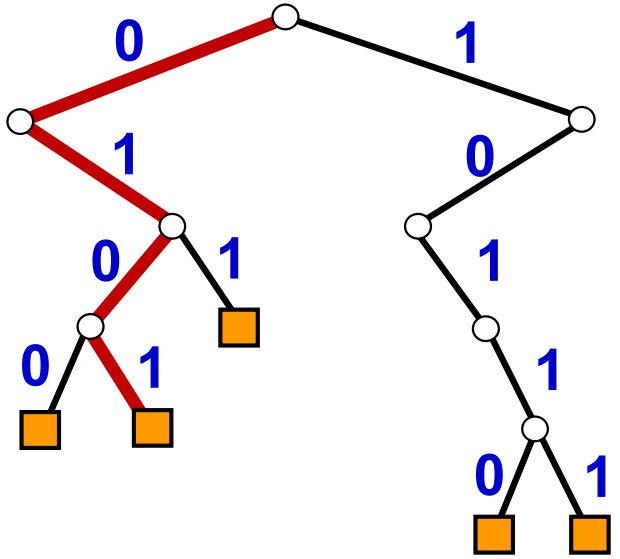
选择两个匹配的地址中更具体的一个，即选择最长前缀的地址。

3. 使用二叉线索查找路由表

- 当路由表的项目数很大时，怎样设法减小路由表的查找时间就成为一个非常重要的问题。
- 为了进行更加有效的查找，通常是将无分类编址的路由表存放在一种层次的数据结构中，然后自上而下地按层次进行查找。这里最常用的就是二叉线索 (binary trie)。
- IP 地址中从左到右的比特值决定了从根结点逐层向下层延伸的路径，而二叉线索中的各个路径就代表路由表中存放的各个地址。
- 为了提高二叉线索的查找速度，广泛使用了各种压缩技术。

用 5 个前缀构成的二叉线索

32 位的 IP 地址	唯一前缀
01000110 00000000 00000000 00000000	0100
01010110 00000000 00000000 00000000	0101
01100001 00000000 00000000 00000000	011
10110000 00000010 00000000 00000000	10110
10111011 00001010 00000000 00000000	10111



从二叉线索的根节点自顶向下的深度最多有32层，每一层对应于IP地址中的一位。一个IP地址存入二叉线索的规则很简单。先检查IP地址左边的第一位，如为 0，则第一层的节点就在根节点的左下方；如为 1，则在右下方。然后再检查地址的第二位，构造出第二层的节点。依此类推，直到唯一前缀的最后一位。

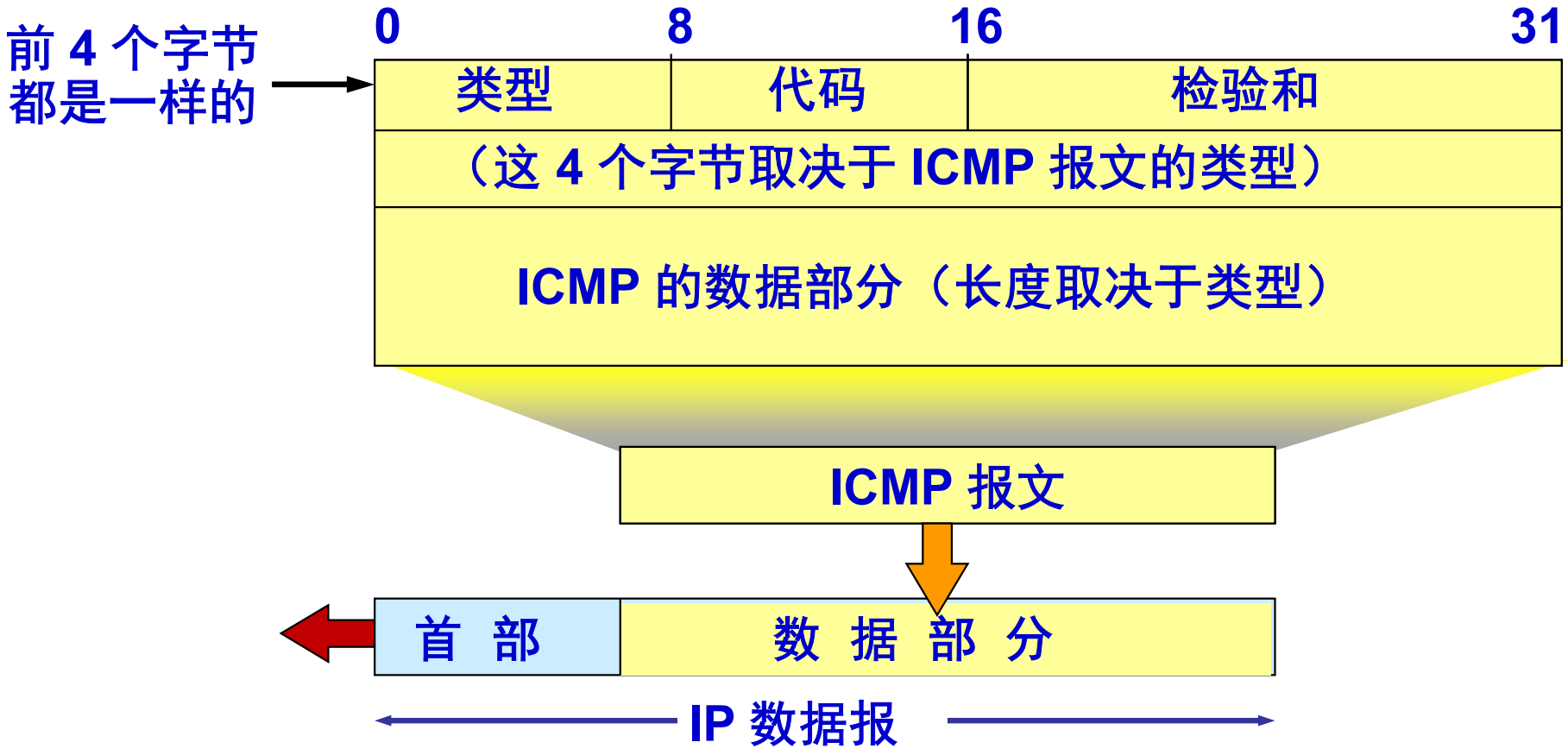
4.4 网际控制报文协议 ICMP

- 4.4.1 ICMP 报文的种类
- 4.4.2 ICMP 的应用举例

4.4 网际控制报文协议ICMP

- 为了更有效地转发 IP 数据报和提高交付成功的机会，在网际层使用了网际控制报文协议 ICMP (Internet Control Message Protocol)。
- ICMP 是互联网的标准协议。
- 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。
- 但 ICMP 不是高层协议（看起来好像是高层协议，因为 ICMP 报文是装在 IP 数据报中，作为其中的数据部分），而是 IP 层的协议。

ICMP 报文的格式


前 4 个字节
都是一样的
0
8
16
31
类型
代码
检验和
（这 4 个字节取决于 ICMP 报文的类型）
ICMP 的数据部分（长度取决于类型）
ICMP 报文
首 部
数 据 部 分
IP 数据报

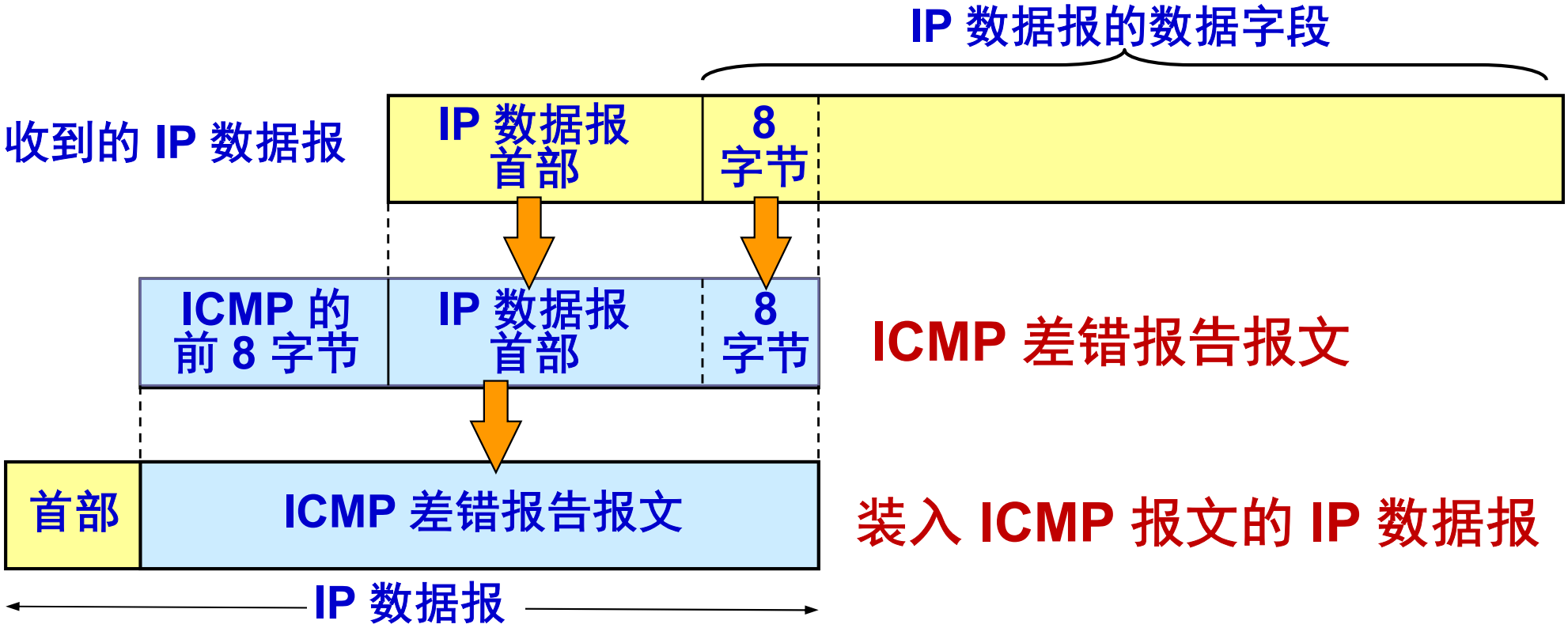
4.4.1 ICMP 报文的种类

- ICMP 报文的种类有两种，即 ICMP 差错报告报文和 ICMP 询问报文。
- ICMP 报文的前 4 个字节是统一的格式，共有三个字段：即类型、代码和检验和。接着的 4 个字节的内容与 ICMP 的类型有关。

ICMP 差错报告报文共有 4 种

- 终点不可达
- 时间超过
- 参数问题
- 改变路由（重定向）**(Redirect)**

ICMP 差错报告报文的数据字段的内容



不应发送 ICMP 差错报告报文的几种情况

- 对 ICMP 差错报告报文不再发送 ICMP 差错报告报文。
- 对第一个分片的数据报片的所有后续数据报片都不发送 ICMP 差错报告报文。
- 对具有多播地址的数据报都不发送 ICMP 差错报告报文。
- 对具有特殊地址（如127.0.0.0 或 0.0.0.0）的数据报不发送 ICMP 差错报告报文。

ICMP 询问报文有两种

- 回送请求和回答报文
- 时间戳请求和回答报文

下面的几种 ICMP 报文不再使用：

- 信息请求与回答报文
- 掩码地址请求和回答报文
- 路由器询问和通告报文
- 源点抑制报文

4.4.2 ICMP的应用举例

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。**
- **PING 是应用层直接使用网络层 ICMP 的例子，它没有通过运输层的 TCP 或UDP。**

PING 的应用举例

```
C:\Documents and Settings\XXR>ping mail.sina.com.cn

Pinging mail.sina.com.cn [202.108.43.230] with 32 bytes of data:

Reply from 202.108.43.230: bytes=32 time=368ms TTL=242
Reply from 202.108.43.230: bytes=32 time=374ms TTL=242
Request timed out.
Reply from 202.108.43.230: bytes=32 time=374ms TTL=242

Ping statistics for 202.108.43.230:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 368ms, Maximum = 374ms, Average = 372ms
```

用 PING 测试主机的连通性

4.4.2 ICMP的应用举例

Traceroute 的应用举例

- 在 Windows 操作系统中这个命令是 tracert。
- 用来跟踪一个分组从源点到终点的路径。
- 它利用 IP 数据报中的 TTL 字段和 ICMP 时间超过差错报告报文实现对从源点到终点的路径的跟踪。

4.4.2 ICMP的应用举例

```
C:\Documents and Settings\XXR>tracert mail.sina.com.cn

Tracing route to mail.sina.com.cn [202.108.43.230]
over a maximum of 30 hops:

  1     24 ms     24 ms     23 ms  222.95.172.1
  2     23 ms     24 ms     22 ms  221.231.204.129
  3     23 ms     22 ms     23 ms  221.231.206.9
  4     24 ms     23 ms     24 ms  202.97.27.37
  5     22 ms     23 ms     24 ms  202.97.41.226
  6     28 ms     28 ms     28 ms  202.97.35.25
  7     50 ms     50 ms     51 ms  202.97.36.86
  8    308 ms    311 ms    310 ms  219.158.32.1
  9    307 ms    305 ms    305 ms  219.158.13.17
 10    164 ms    164 ms    165 ms  202.96.12.154
 11    322 ms    320 ms   2988 ms  61.135.148.50
 12    321 ms    322 ms    320 ms  freemail43-230.sina.com [202.108.43.230]

Trace complete.
```

用 **tracert** 命令获得到目的主机的路由信息

4.5 互联网的路由选择协议

- 4.5.1 有关路由选择协议的几个基本概念
- 4.5.2 内部网关协议 RIP
- 4.5.3 内部网关协议 OSPF
- 4.5.4 外部网关协议 BGP
- 4.5.5 路由器的构成

4.5.1 有关路由选择协议的几个基本概念

1. 理想的路由算法

- 算法必须是正确的和完整的。
- 算法在计算上应简单。
- 算法应能适应通信量和网络拓扑的变化，这就是说，要有自适应性。
- 算法应具有稳定性。
- 算法应是公平的。
- 算法应是最佳的。

关于“最佳路由”

- 不存在一种绝对的最佳路由算法。
- 所谓“最佳”只能是相对于某一种特定要求下得出的较为合理的选择而已。
- 实际的路由选择算法，应尽可能接近于理想的算法。
- 路由选择是个非常复杂的问题
  - 它是网络中的所有结点共同协调工作的结果。
  - 路由选择的环境往往是不断变化的，而这种变化有时无法事先知道。

从路由算法的自适应性考虑

- 静态路由选择策略——即非自适应路由选择，其特点是简单和开销较小，但不能及时适应网络状态的变化。
- 动态路由选择策略——即自适应路由选择，其特点是能较好地适应网络状态的变化，但实现起来较为复杂，开销也比较大。

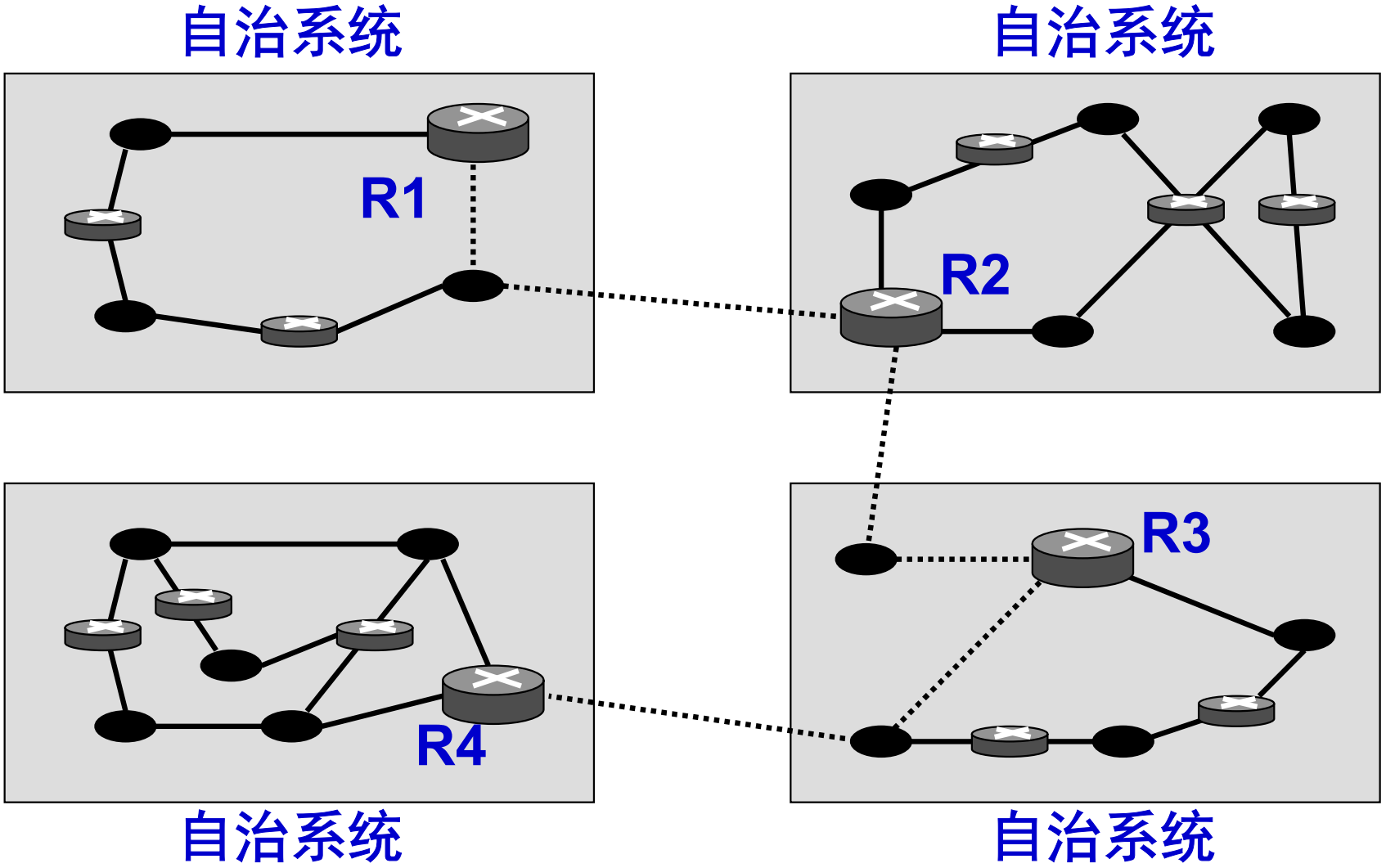
2. 分层次的路由选择协议

- 互联网采用分层次的路由选择协议。这是因为：
  - (1) 互联网的规模非常大。如果让所有的路由器知道所有的网络应怎样到达，则这种路由表将非常大，处理起来也太花时间。而所有这些路由器之间交换路由信息所需的带宽就会使互联网的通信链路饱和。
  - (2) 许多单位不愿意外界了解自己单位网络的布局细节和本部门所采用的路由选择协议（这属于本部门内部的事情），但同时还希望连接到互联网上。

自治系统 AS (Autonomous System)

- 自治系统 AS 的定义：在单一的技术管理下的一组路由器，而这些路由器使用一种 AS 内部的路由选择协议和共同的度量以确定分组在该 AS 内的路由，同时还使用一种 AS 之间的路由选择协议用以确定分组在AS之间的路由。
- 现在对自治系统 AS 的定义是强调下面的事实：尽管一个 AS 使用了多种内部路由选择协议和度量，但重要的是一个 AS 对其他 AS 表现出的是一个单一的和一致的路由选择策略。

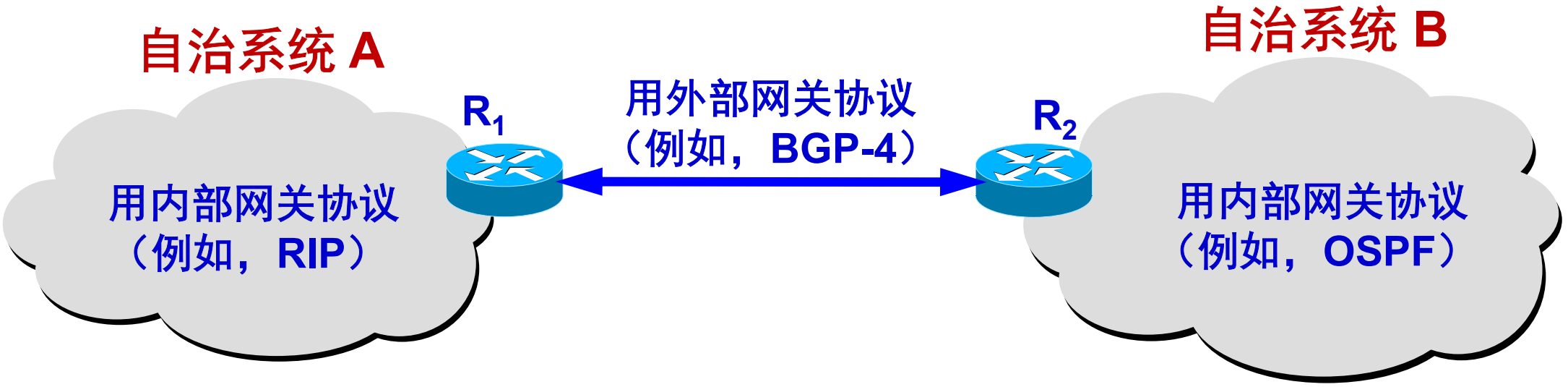
自治系统 AS



互联网有两大类路由选择协议

- 内部网关协议 IGP (Interior Gateway Protocol)
  - 在一个自治系统内部使用的路由选择协议。
  - 目前这类路由选择协议使用得最多，如 RIP 和 OSPF 协议。
- 外部网关协议 EGP (External Gateway Protocol)
  - 若源站和目的站处在不同的自治系统中，当数据报传到一个自治系统的边界时，就需要使用一种协议将路由选择信息传递到另一个自治系统中。这样的协议就是外部网关协议 EGP。
  - 在外部网关协议中目前使用最多的是 BGP-4。

自治系统和
内部网关协议、外部网关协议



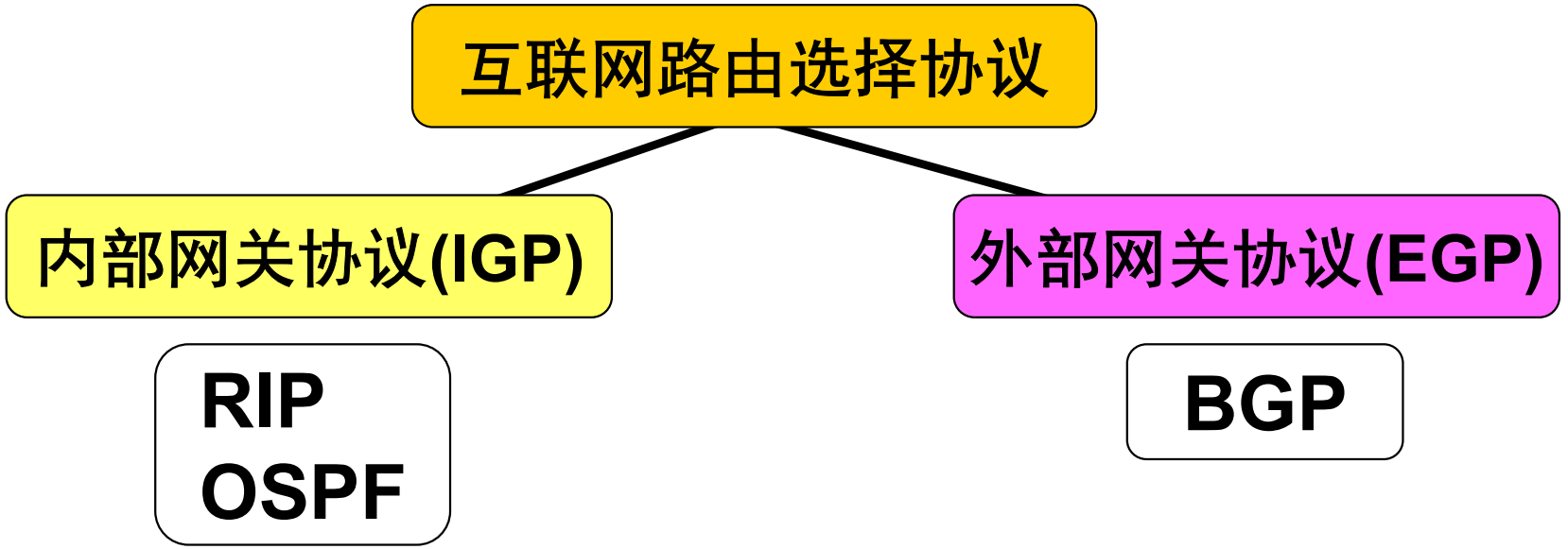
自治系统之间的路由选择也叫作域间路由选择 (interdomain routing)，在自治系统内部的路由选择叫作域内路由选择 (intradomain routing)。

这里要指出两点

- 互联网的早期 RFC 文档中未使用“路由器”而是使用“网关”这一名词。但是在新的 RFC 文档中又使用了“路由器”这一名词。应当把这两个术语当作同义词。
- IGP 和 EGP 是协议类别的名称。但 RFC 在使用 EGP 这个名词时出现了一点混乱，因为最早的一个外部网关协议的协议名字正好也是 EGP。因此在遇到名词 EGP 时，应弄清它是指旧的协议 EGP 还是指外部网关协议 EGP 这个类别。

互联网的路由选择协议

- 内部网关协议 IGP：具体的协议有多种，如 RIP 和 OSPF 等。
- 外部网关协议 EGP：目前使用的协议就是 BGP。



4.5.2 内部网关协议 RIP

1. 工作原理

- 路由信息协议 **RIP (Routing Information Protocol)** 是内部网关协议 **IGP** 中最先得到广泛使用的协议。
- **RIP** 是一种分布式的、基于距离向量的路由选择协议。
- **RIP** 协议要求网络中的每一个路由器都要维护从它自己到其他每一个目的网络的距离记录。

“距离”的定义

- 从一个路由器到直接连接的网络的距离定义为 1。
- 从一个路由器到非直接连接的网络的距离定义为所经过的路由器数加 1。
- RIP 协议中的“距离”也称为“跳数”(hop count)，因为每经过一个路由器，跳数就加 1。
- 这里的“距离”实际上指的是“最短距离”。

“距离”的定义

- RIP 认为一个好的路由就是它通过的路由器的数目少，即“距离短”。
- RIP 允许一条路径最多只能包含 15 个路由器。
- “距离”的最大值为 16 时即相当于不可达。可见 RIP 只适用于小型互联网。
- RIP 不能在两个网络之间同时使用多条路由。RIP 选择一个具有最少路由器的路由（即最短路由），哪怕还存在另一条高速(低时延)但路由器较多的路由。

RIP 协议的三个特点

- (1) 仅和相邻路由器交换信息。
- (2) 交换的信息是当前本路由器所知道的全部信息，即自己的路由表。
- (3) 按固定的时间间隔交换路由信息，例如，每隔 30 秒。当网络拓扑发生变化时，路由器也及时向相邻路由器通告拓扑变化后的路由信息。

路由表的建立

- 路由器在刚刚开始工作时，只知道到直接连接的网络的距离（此距离定义为1）。它的路由表是空的。
- 以后，每一个路由器也只和数目非常有限的相邻路由器交换并更新路由信息。
- 经过若干次更新后，所有的路由器最终都会知道到达本自治系统中任何一个网络的最短距离和下一跳路由器的地址。
- RIP 协议的收敛 (convergence) 过程较快。“收敛”就是在自治系统中所有的结点都得到正确的路由选择信息的过程。

2. 距离向量算法

路由器收到相邻路由器（其地址为 X）的一个 RIP 报文：

(1) 先修改此 RIP 报文中的所有项目：把“下一跳”字段中的地址都改为 X，并把所有的“距离”字段的值加 1。

(2) 对修改后的 RIP 报文中的每一个项目，重复以下步骤：

若项目中的目的网络不在路由表中，则把该项目加到路由表中。

否则

若下一跳字段给出的路由器地址是同样的，则把收到的项目替换原路由表中的项目。

否则

若收到项目中的距离小于路由表中的距离，则进行更新，

否则，什么也不做。

(3) 若 3 分钟还没有收到相邻路由器的更新路由表，则把此相邻路由器记为不可达路由器，即将距离置为 16（表示不可达）。

(4) 返回。

2. 距离向量算法

- 距离向量算法的基础就是 **Bellman-Ford** 算法（或 **Ford-Fulkerson** 算法）。
- 这种算法的要点是这样的：

  设**X**是结点 **A** 到 **B** 的最短路径上的一个结点。

  若把路径 **A→B** 拆成两段路径 **A→X** 和 **X→B**，则每一段路径 **A→X** 和 **X→B** 也都分别是结点 **A** 到 **X** 和结点 **X** 到 **B** 的最短路径。

路由器之间交换信息与路由表更新

- RIP 协议让互联网中的所有路由器都和自己的相邻路由器不断交换路由信息，并不断更新其路由表，使得从每一个路由器到每一个目的网络的路由都是最短的（即跳数最少）。
- 虽然所有的路由器最终都拥有了整个自治系统的全局路由信息，但由于每一个路由器的位置不同，它们的路由表当然也应当是不同的。

【例4-5】已知路由器 R_6 有表4-9(a)所示的路由表。现在收到相邻路由器 R_4 发来的路由更新信息，如表4-9(b)所示。试更新路由器 R_6 的路由表。

表4-9(a) 路由器R_6的路由表

目的网络	距离	下一跳路由器
Net2	3	R_4
Net3	4	R_5
…	…	…

表4-9(b) R_4发来的路由更新信息

目的网络	距离	下一跳路由器
Net1	3	R_1
Net2	4	R_2
Net3	1	直接交付

距离加1

表4-9(c) 修改后的表4-9(b)

目的网络	距离	下一跳路由器
Net1	4	R_4
Net2	5	R_4
Net3	2	R_4

计算更新

表4-9(d) 路由器R_6更新后的路由表

目的网络	距离	下一跳路由器
Net1	4	R_4
Net2	5	R_4
Net3	2	R_4
…	…	…

【例】路由表更新

从C来的RIP报文

Net2	4
Net3	8
Net6	4
Net8	3
Net9	5

增加跳数以后
从C来的RIP报文

Net2	5
Net3	9
Net6	5
Net8	4
Net9	6

Net1:没有新信息，不变
Net2:相同的下一跳，替换
Net3:一条新路由，增加
Net6:不同的下一跳，新跳数小，替换
Net8:不同的下一跳，跳数相同，不变
Net9:不同的下一跳，新跳数大，不变

旧路由表

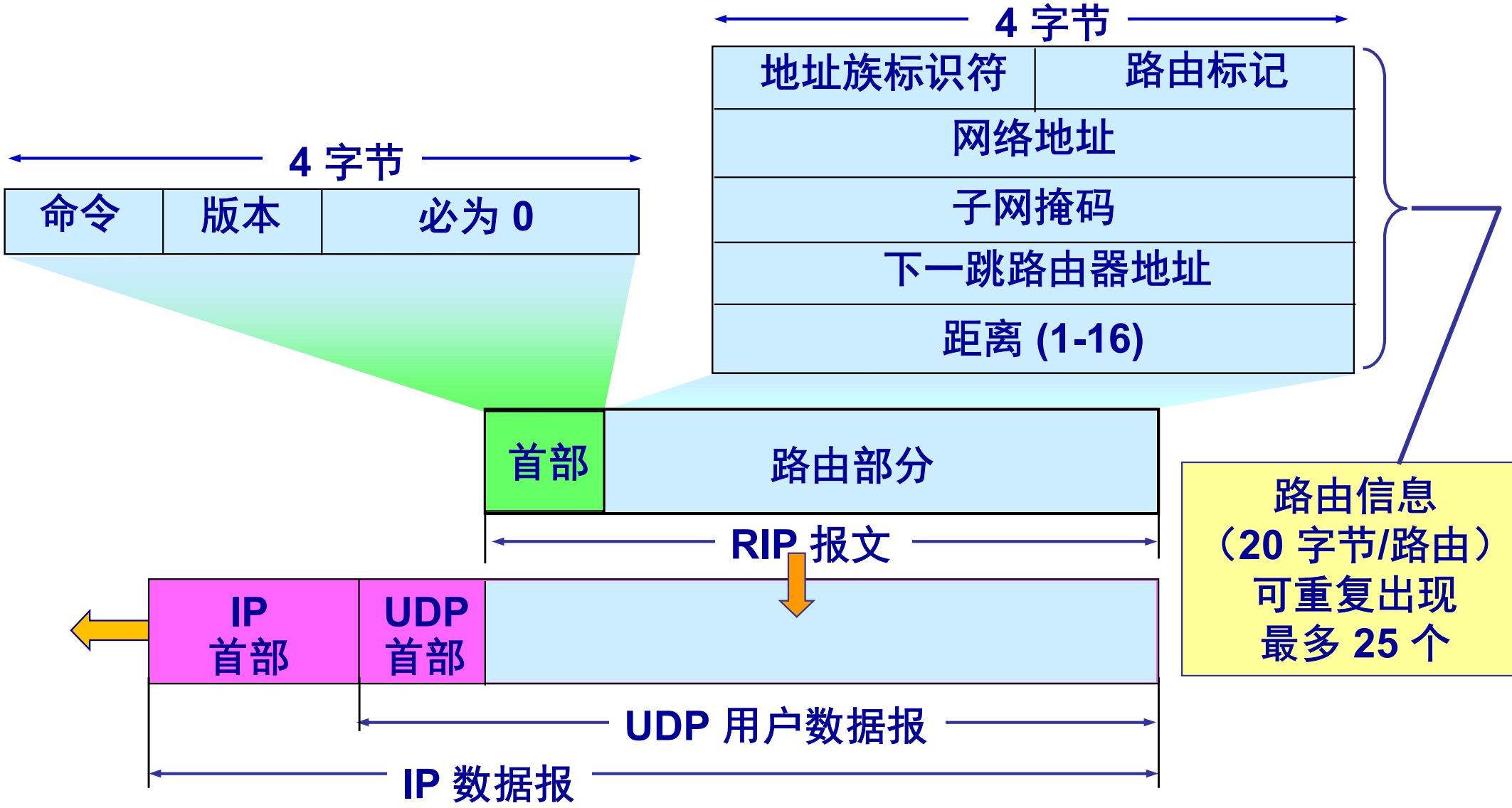
Net1	7	A
Net2	2	C
Net6	8	F
Net8	4	E
Net9	4	F

更新算法

新路由表

Net1	7	A
Net2	5	C
Net3	9	C
Net6	5	C
Net8	4	E
Net9	4	F

3. RIP2 协议的报文格式



RIP2 报文

- RIP2 报文由首部和路由部分组成。
- RIP2 报文中的路由部分由若干个路由信息组成。每个路由信息需要用 20 个字节。地址族标识符（又称为地址类别）字段用来标志所使用的地址协议。
- 路由标记填入自治系统的号码，这是考虑使RIP有可能收到本自治系统以外的路由选择信息。
- 再后面指出某个网络地址、该网络的子网掩码、下一跳路由器地址以及到此网络的距离。

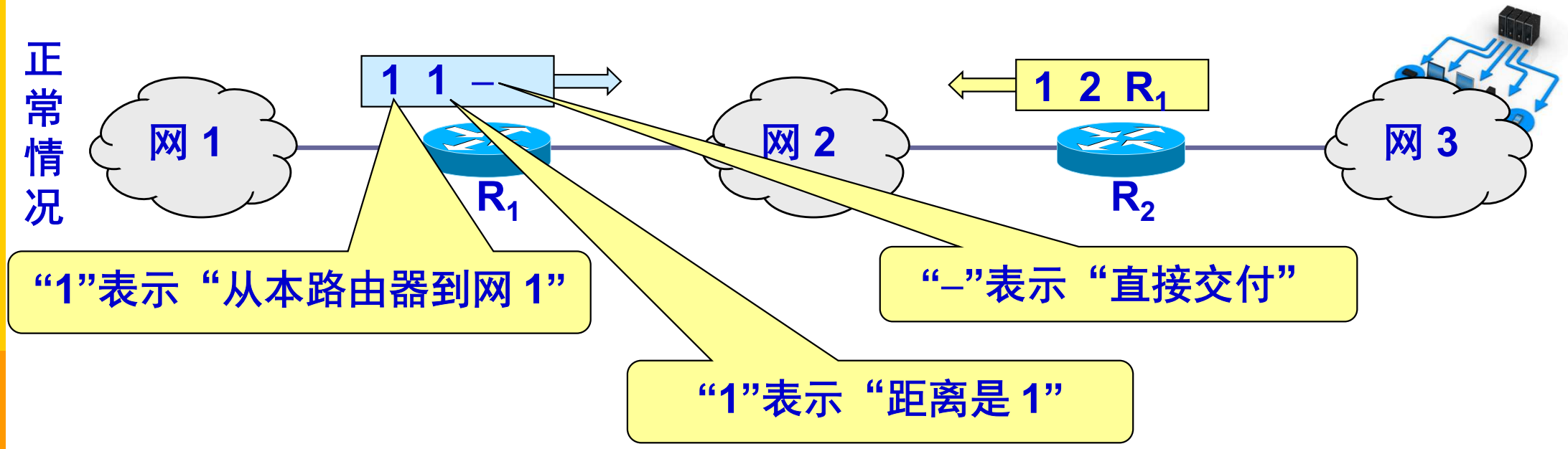
RIP2 报文

- 一个 RIP 报文最多可包括 25 个路由，因而 RIP 报文的最大长度是4 ＋20 × 25 ＝ 504 字节。如超过，必须再用一个 RIP 报文来传送。
- RIP2 具有简单的鉴别功能。
  - 若使用鉴别功能，则将原来写入第一个路由信息（20字节）的位置用作鉴别。
  - 在鉴别数据之后才写入路由信息，但这时最多只能再放入 24 个路由信息。

好消息传播得快，坏消息传播得慢

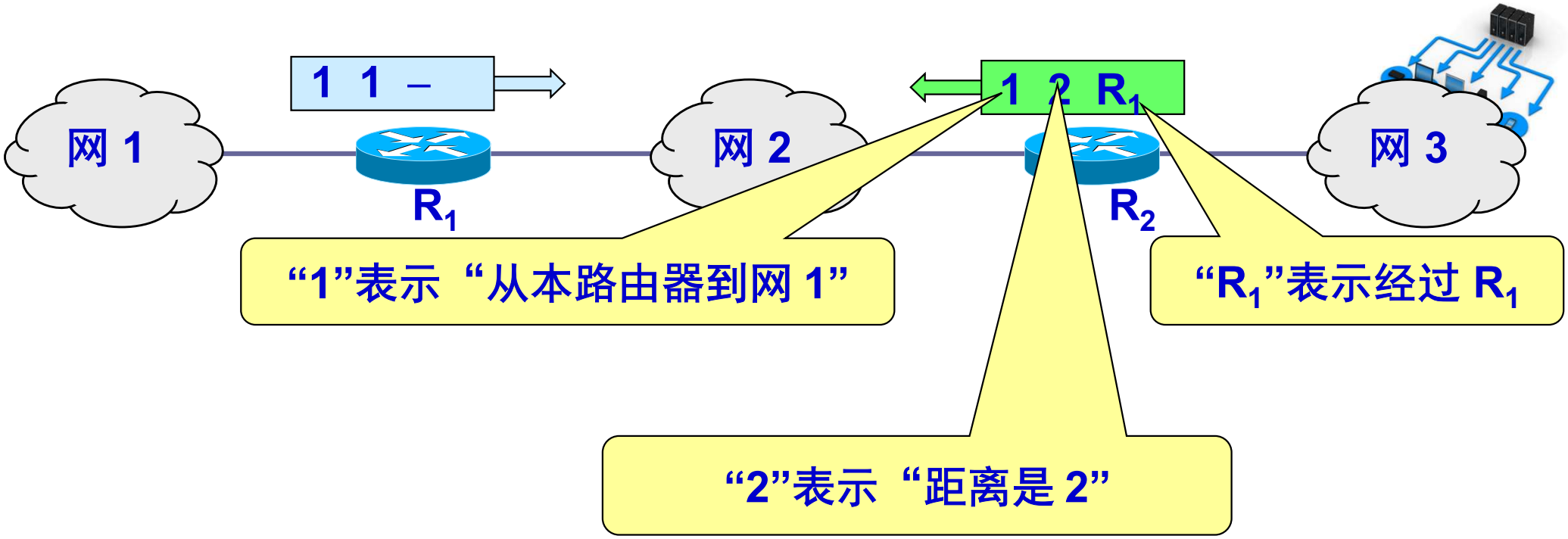
- **RIP协议特点：**好消息传播得快，坏消息传播得慢。
- **RIP存在的一个问题：**当网络出现故障时，要经过比较长的时间 (例如数分钟) 才能将此信息传送到所有的路由器。

正常情况



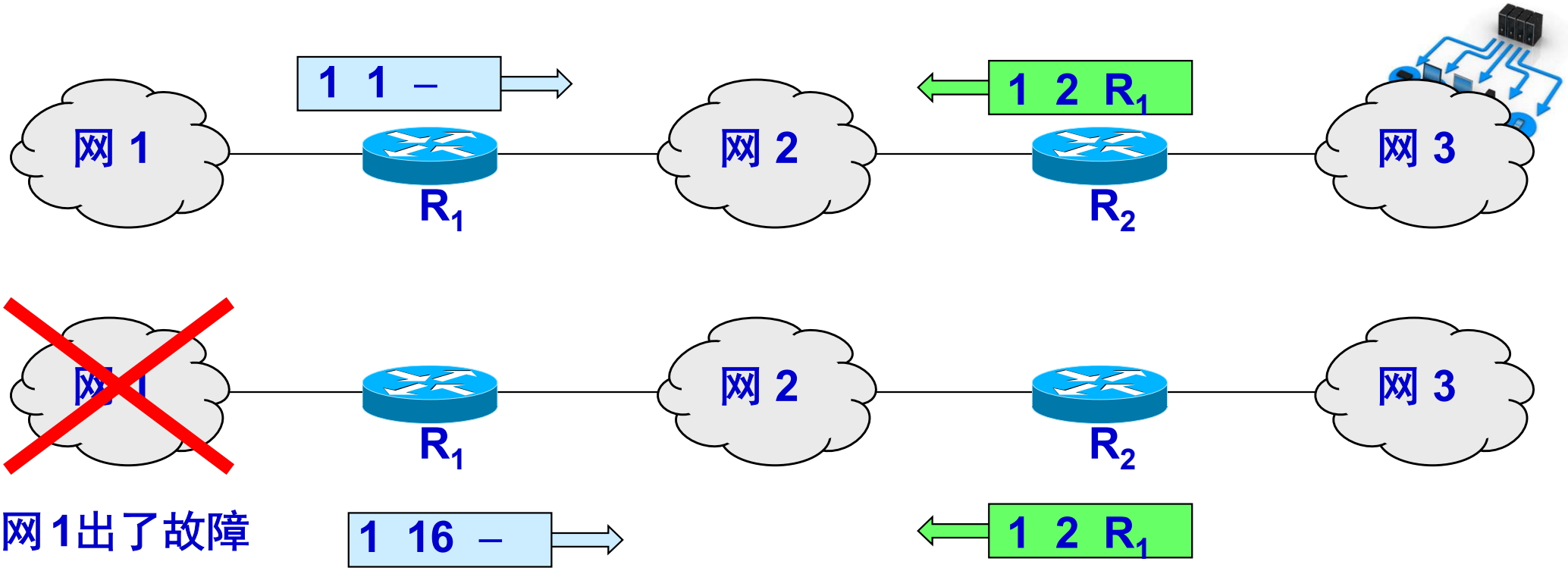
R_1 说：“我到网 1 的距离是 1，是直接交付。”

正常情况



R2 说：“我到网 1 的距离是 2，是经过 R1。”

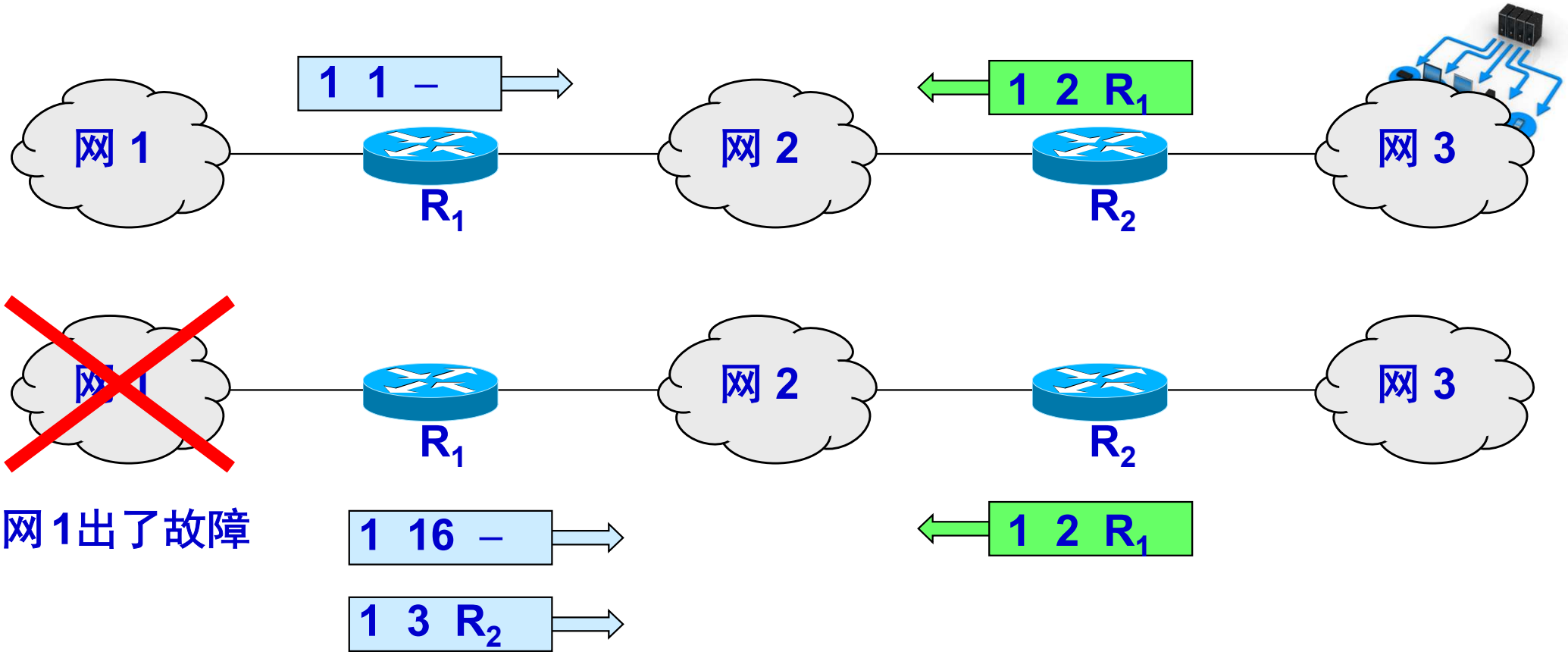
正常情况



R_1 说：“我到网 1 的距离是 16 （表示无法到达），是直接交付。”

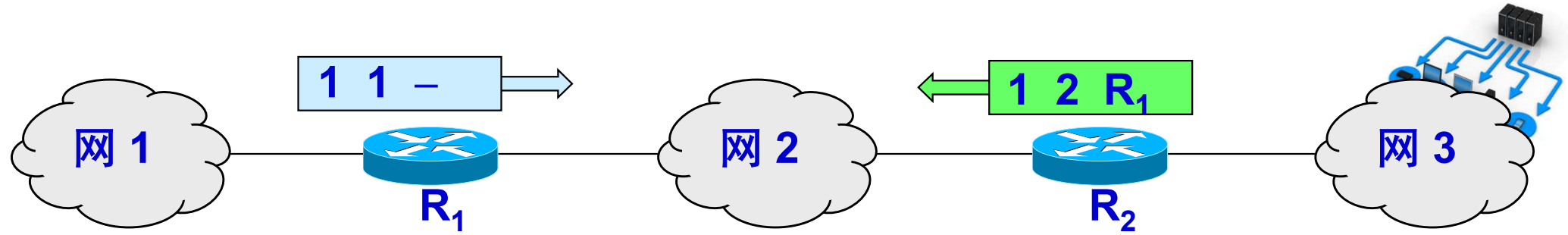
但 R_2 在收到 R_1 的更新报文之前，还发送原来的报文，因为这时 R_2 并不知道 R_1 出了故障。

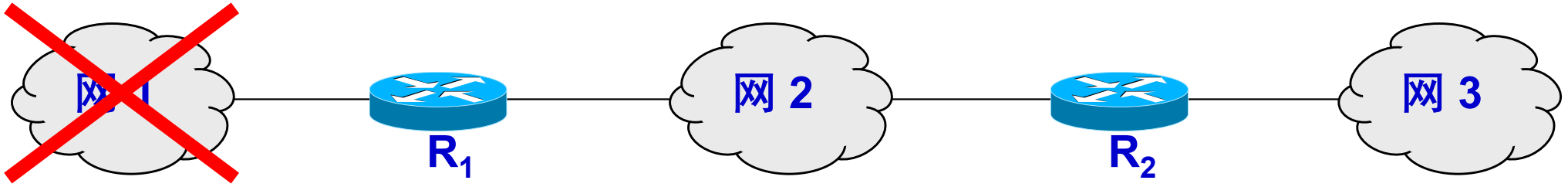
正常情况



R_1 收到 R_2 的更新报文后，误认为可经过 R_2 到达网1，于是更新自己的路由表，说：“我到网 1 的距离是 3，下一跳经过 R_2”。然后将此更新信息发送给 R_2。

正常情况





网1出了故障

1 16 –

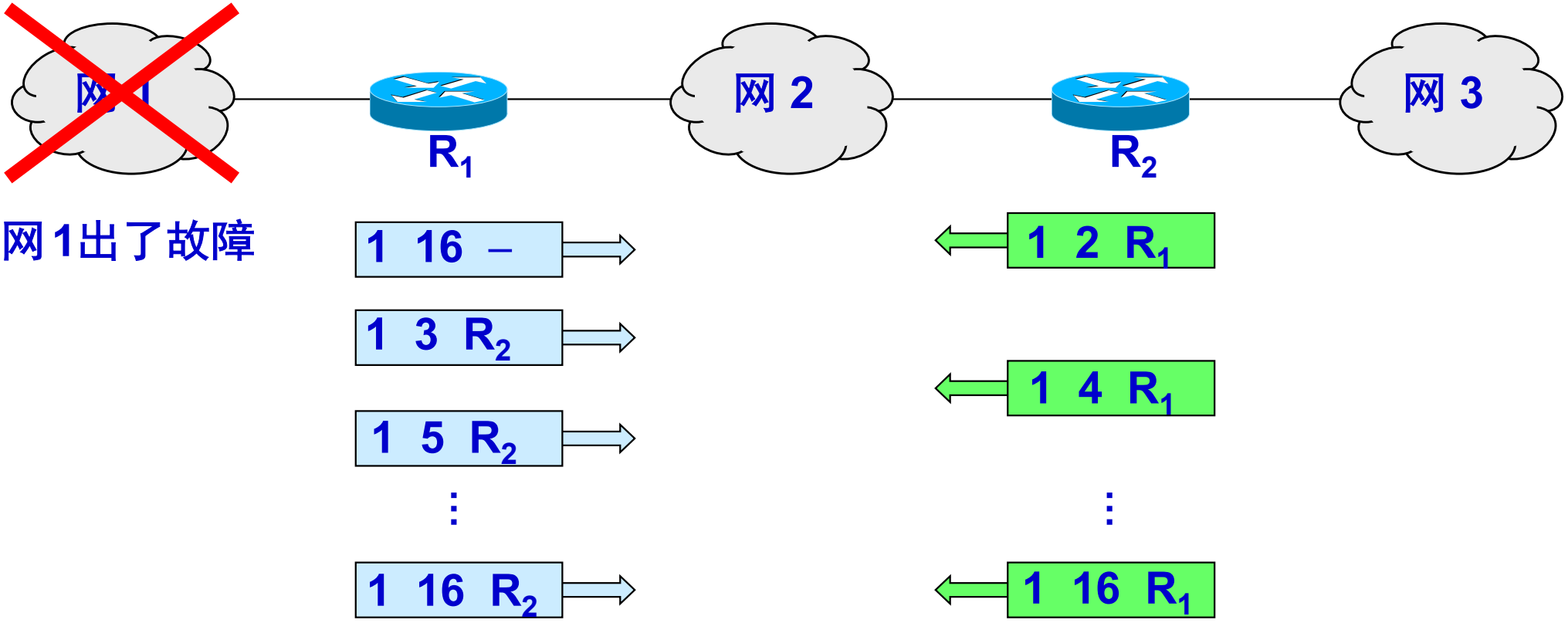
1 2 R_1

1 3 R_2

1 4 R_1

R_2 以后又更新自己的路由表为“1, 4, R_1”，表明 “我到网 1 距离是 4，下一跳经过 R_1”。

这就是好消息传播得快，而坏消息传播得慢。网络出故障的传播时间往往需要较长的时间(例如数分钟)。这是 RIP 的一个主要缺点。



这样不断更新下去，直到 R_1 和 R_2 到网 1 的距离都增大到 16 时，R_1 和 R_2 才知道网 1 是不可达的。

RIP 协议的优缺点

- 优点：
  - 实现简单，开销较小。
- 缺点：
  - RIP 限制了网络的规模，它能使用的最大距离为 15（16 表示不可达）。
  - 路由器之间交换的路由信息是路由器中的完整路由表，因而随着网络规模的扩大，开销也就增加。
  - “坏消息传播得慢”，使更新过程的收敛时间过长。

4.5.3 内部网关协议 OSPF

- 开放最短路径优先 OSPF (Open Shortest Path First)是为克服 RIP 的缺点在1989年开发出来的。
- OSPF 的原理很简单，但实现起来却较复杂。

1. OSPF 协议的基本特点

- “开放”表明 OSPF 协议不是受某一家厂商控制，而是公开发表的。
- “最短路径优先”是因为使用了 Dijkstra 提出的最短路径算法 SPF
- 采用分布式的链路状态协议 (link state protocol)。
- 注意：OSPF 只是一个协议的名字，它并不表示其他的路由选择协议不是“最短路径优先”。

三个要点

- 向本自治系统中所有路由器发送信息，这里使用的方法是洪泛法。
- 发送的信息就是与本路由器相邻的所有路由器的链路状态，但这只是路由器所知道的部分信息。
  - “链路状态”就是说明本路由器都和哪些路由器相邻，以及该链路的“度量”(metric)。
- 只有当链路状态发生变化时，路由器才用洪泛法向所有路由器发送此信息。

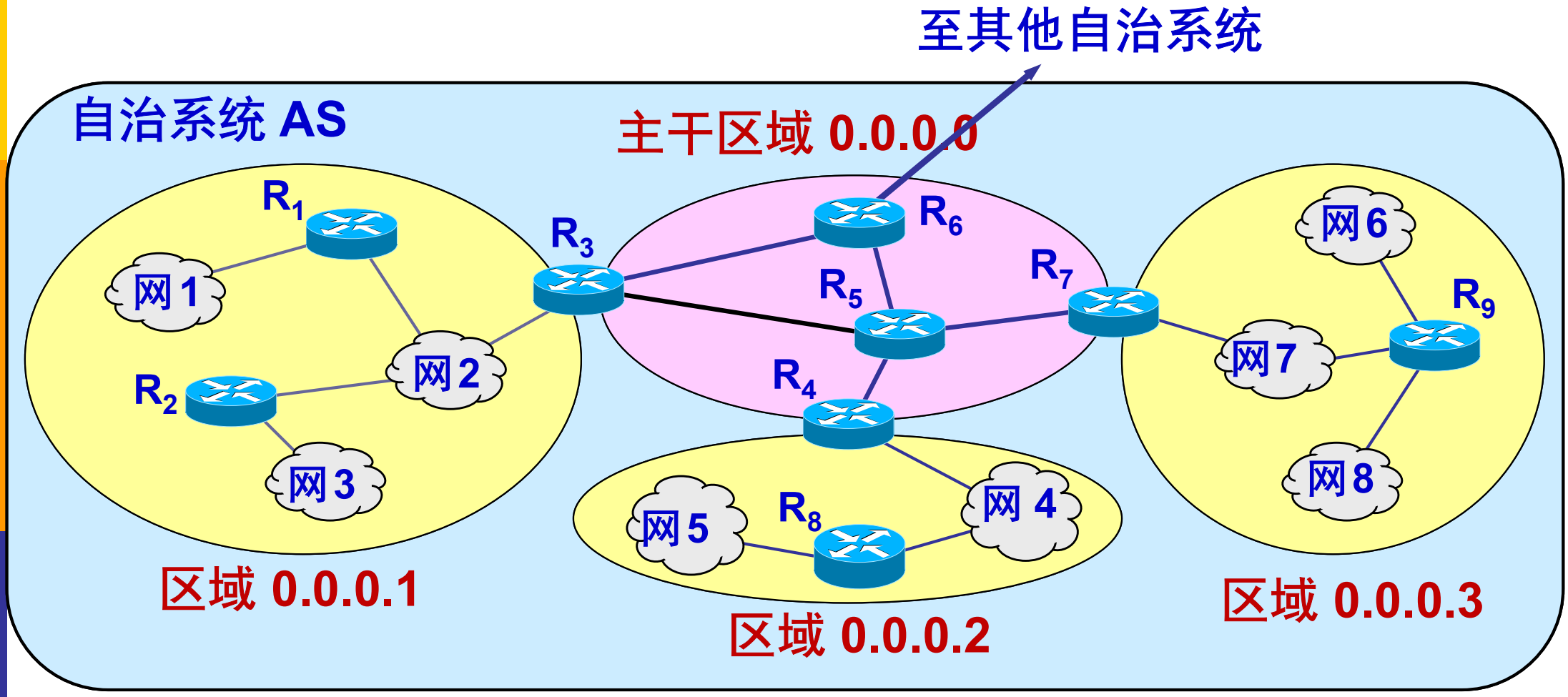
链路状态数据库 (link-state database)

- 由于各路由器之间频繁地交换链路状态信息，因此所有的路由器最终都能建立一个链路状态数据库。
- 这个数据库实际上就是全网的拓扑结构图，它在全网范围内是一致的（这称为链路状态数据库的同步）。
- **OSPF** 的链路状态数据库能较快地进行更新，使各个路由器能及时更新其路由表。
- **OSPF** 的更新过程收敛得快是其重要优点。

OSPF 的区域 (area)

- 为了使 OSPF 能够用于规模很大的网络，OSPF 将一个自治系统再划分为若干个更小的范围，叫作区域。
- 每一个区域都有一个 32 位的区域标识符（用点分十进制表示）。
- 区域也不能太大，在一个区域内的路由器最好不超过 200 个。

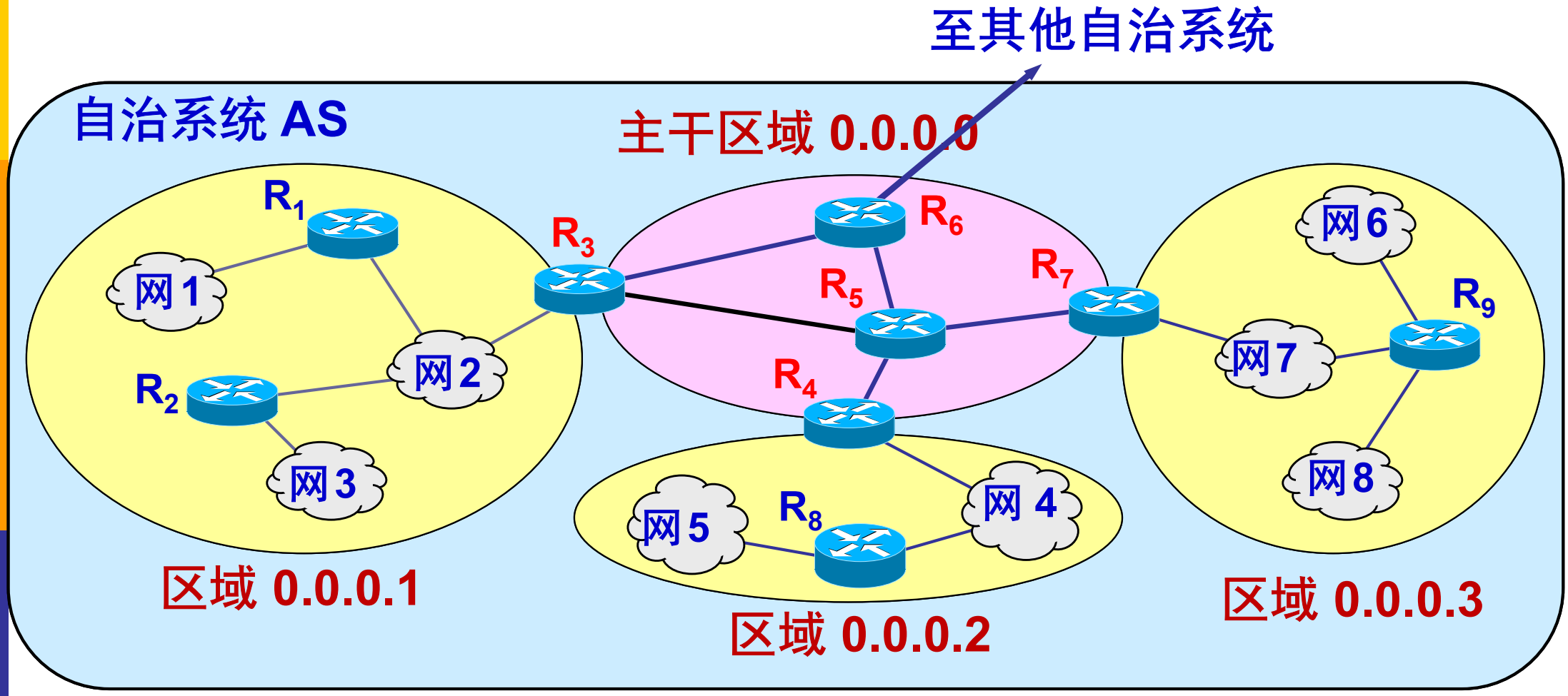
OSPF 划分为两种不同的区域



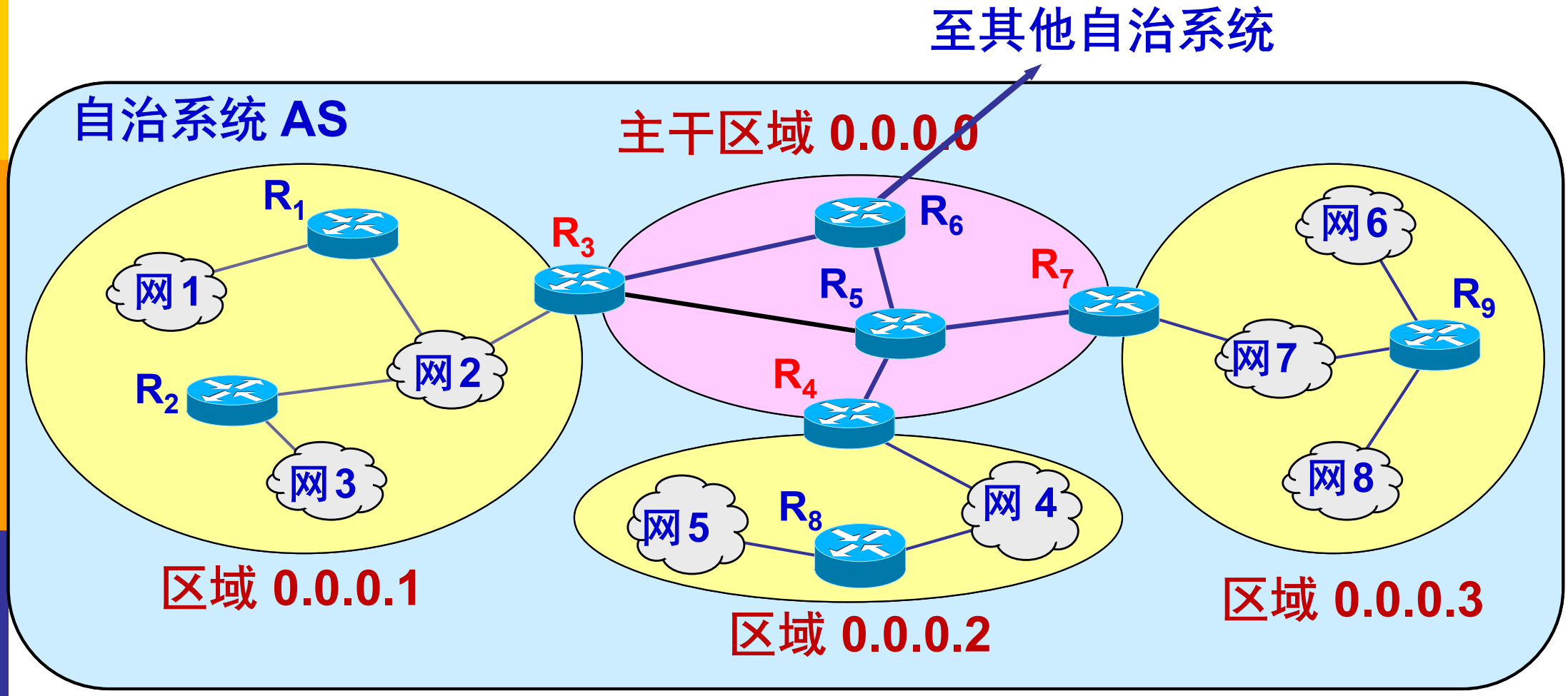
划分区域

- 划分区域的好处就是将利用洪泛法交换链路状态信息的范围局限于每一个区域而不是整个的自治系统，这就减少了整个网络上的通信量。
- 在一个区域内部的路由器只知道本区域的完整网络拓扑，而不知道其他区域的网络拓扑的情况。
- **OSPF** 使用层次结构的区域划分。在上层的区域叫作主干区域 **(backbone area)**。
- 主干区域的标识符规定为**0.0.0.0**。主干区域的作用是用来连通其他在下层的区域。

主干路由器


至其他自治系统
自治系统 AS
主干区域 0.0.0.0
R1
R2
R3
R4
R5
R6
R7
R8
R9
网 1
网 2
网 3
网 4
网 5
网 6
网 7
网 8
区域 0.0.0.1
区域 0.0.0.2
区域 0.0.0.3

区域边界路由器


至其他自治系统
自治系统 AS
主干区域 0.0.0.0
R1
R2
R3
R4
R5
R6
R7
R8
R9
网 1
网 2
网 3
网 4
网 5
网 6
网 7
网 8
区域 0.0.0.1
区域 0.0.0.2
区域 0.0.0.3

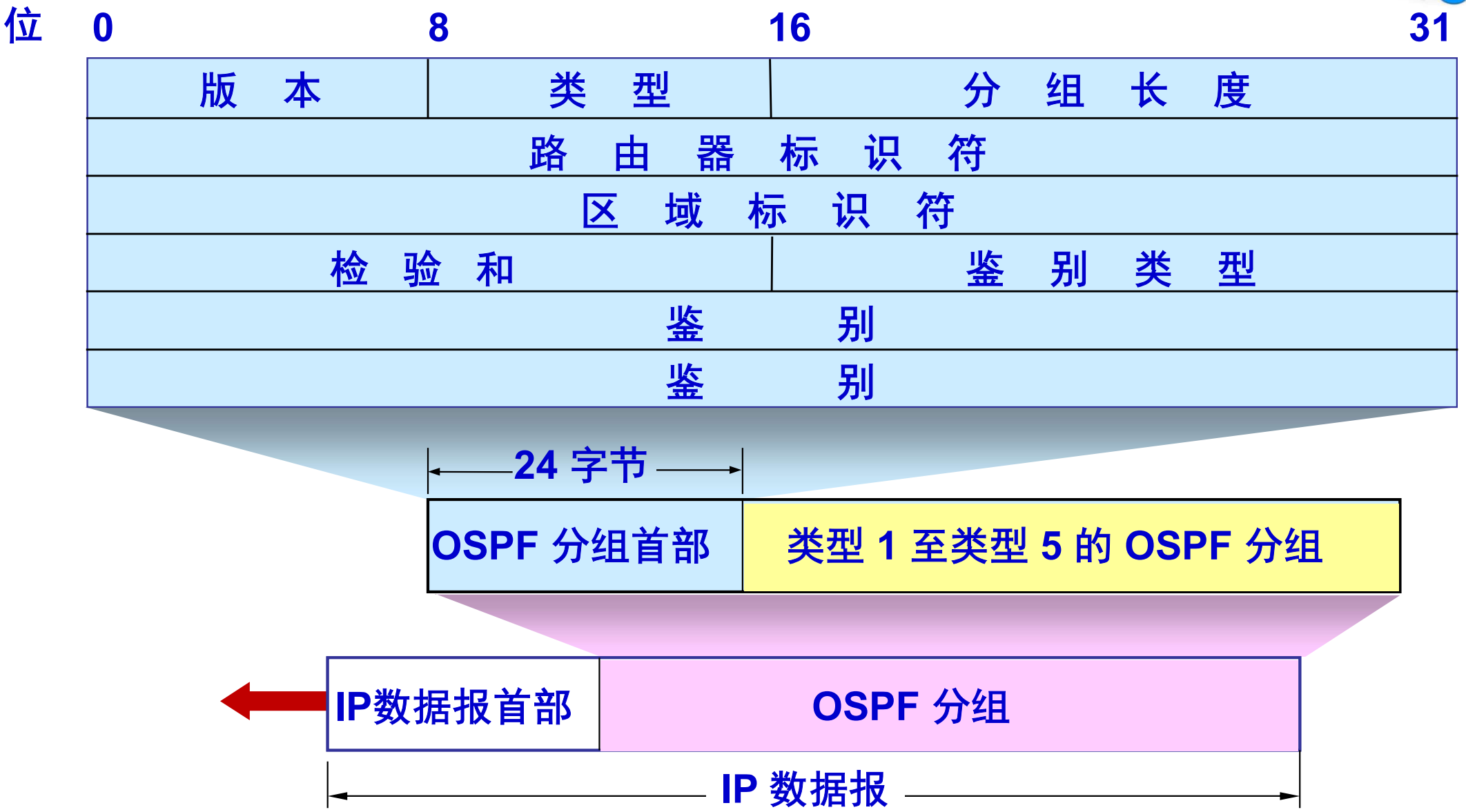
OSPF 直接用 IP 数据报传送

- **OSPF 不用 UDP 而是直接用 IP 数据报传送。**
- **OSPF** 构成的数据报很短。这样做可减少路由信息的通信量。
- 数据报很短的另一好处是可以不必将长的数据报分片传送。
- 但分片传送的数据报只要丢失一个，就无法组装成原来的数据报，而整个数据报就必须重传。

OSPF 的其他特点

- OSPF 对不同的链路可根据 IP 分组的不同服务类型 TOS 而设置成不同的代价。因此，OSPF 对于不同类型的业务可计算出不同的路由。
- 如果到同一个目的网络有多条相同代价的路径，那么可以将通信量分配给这几条路径。这叫作多路径间的负载平衡。
- 所有在 OSPF 路由器之间交换的分组都具有鉴别的功能。
- 支持可变长度的子网划分和无分类编址 CIDR。
- 每一个链路状态都带上一个 32 位的序号，序号越大状态就越新。

OSPF 分组



OSPF 分组用 IP 数据报传送

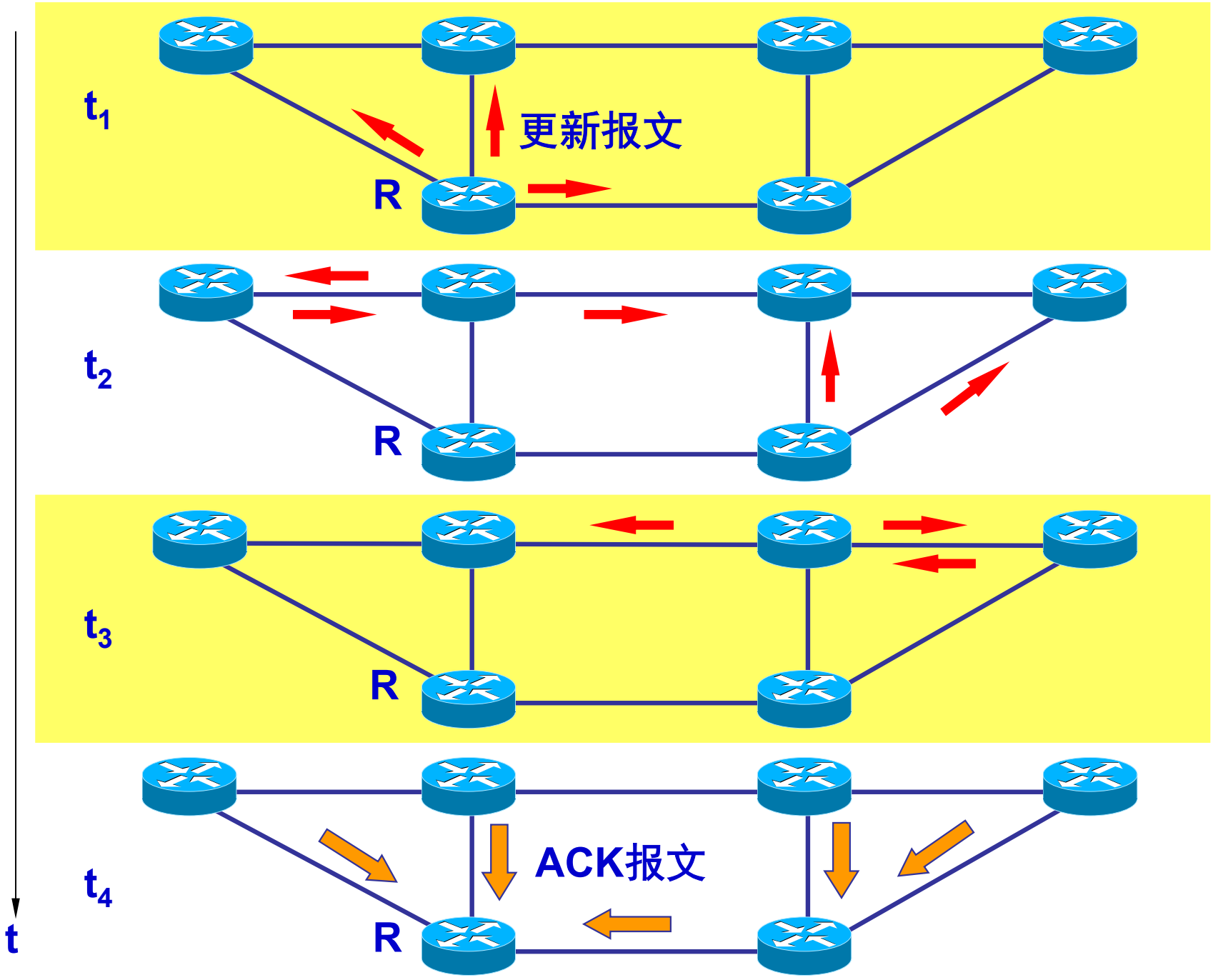
2. OSPF 的五种分组类型

- 类型1，问候 (Hello) 分组。
- 类型2，数据库描述 (Database Description) 分组。
- 类型3，链路状态请求 (Link State Request) 分组。
- 类型4，链路状态更新 (Link State Update) 分组，用洪泛法对全网更新链路状态。
- 类型5，链路状态确认 (Link State Acknowledgment) 分组。

OSPF 的基本操作


确定可达性
问候
问候
达到数据库的同步
数据库描述
数据库描述
数据库描述
数据库描述
新情况下的同步
链路状态请求
链路状态更新
链路状态确认

OSPF 使用可靠的洪泛法发送更新分组



OSPF 的其他特点

- OSPF 还规定每隔一段时间，如 30 分钟，要刷新一次数据库中的链路状态。
- 由于一个路由器的链路状态只涉及到与相邻路由器的连通状态，因而与整个互联网的规模并无直接关系。因此当互联网规模很大时，OSPF 协议要比距离向量协议 RIP 好得多。
- OSPF 没有“坏消息传播得慢”的问题，据统计，其响应网络变化的时间小于 100 ms。

指定的路由器

- 多点接入的局域网采用了指定的路由器 (designated router) 的方法，使广播的信息量大大减少。
- 指定的路由器代表该局域网上所有的链路向连接到该网络上的各路由器发送状态信息。

4.5.4 外部网关协议 BGP

- BGP 是不同自治系统的路由器之间交换路由信息的协议。
- BGP 较新版本是 2006 年 1 月发表的 BGP-4（BGP 第 4 个版本），即 RFC 4271 ~ 4278。
- 可以将 BGP-4 简写为 BGP。

BGP 使用环境不同

- 互联网的规模太大，使得自治系统之间路由选择非常困难。对于自治系统之间的路由选择，要寻找最佳路由是很不现实的。
  - 当一条路径通过几个不同 AS 时，要想对这样的路径计算出有意义的代价是不太可能的。
  - 比较合理的做法是在 AS 之间交换“可达性”信息。
- 自治系统之间的路由选择必须考虑有关策略。
- 因此，边界网关协议 BGP 只能是力求寻找一条能够到达目的网络且比较好的路由（不能兜圈子），而并非要寻找一条最佳路由。

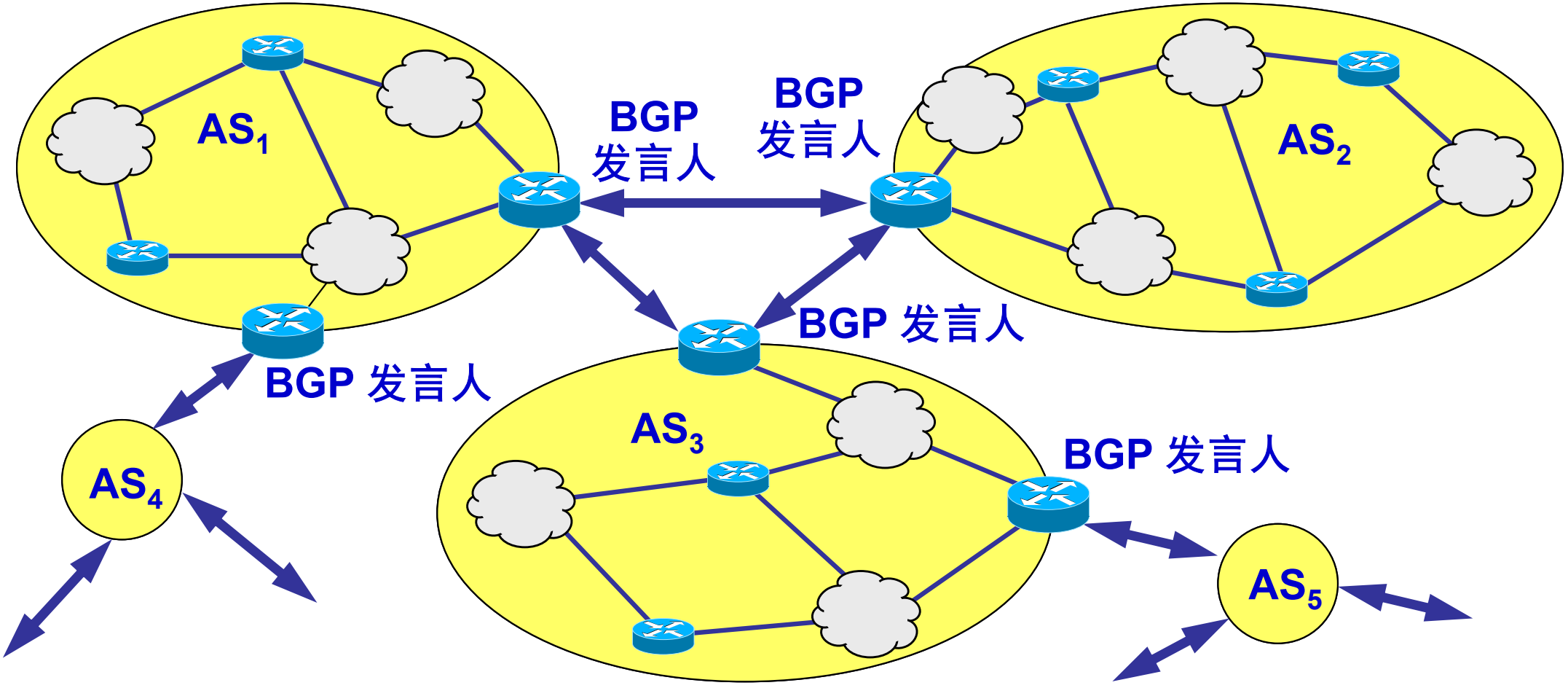
BGP 发言人

- 每一个自治系统的管理员要选择至少一个路由器作为该自治系统的“ BGP 发言人” (BGP speaker) 。
- 一般说来，两个 BGP 发言人都是通过一个共享网络连接在一起的，而 BGP 发言人往往就是 BGP 边界路由器，但也可以不是 BGP 边界路由器。

BGP 交换路由信息

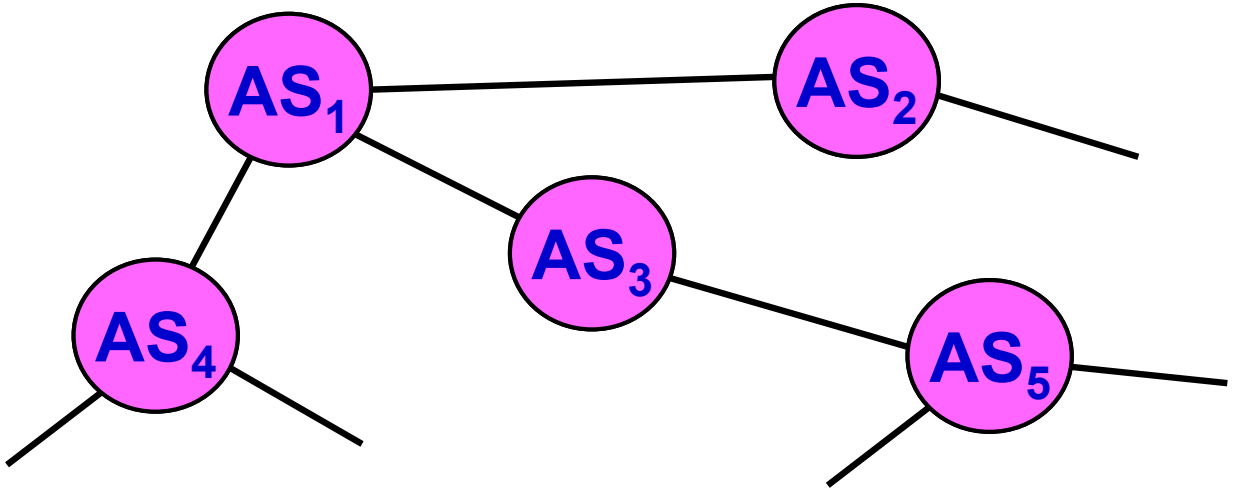
- 一个 BGP 发言人与其他自治系统中的 BGP 发言人要交换路由信息，就要先建立 TCP 连接，然后在此连接上交换 BGP 报文以建立 BGP 会话(session)，利用 BGP 会话交换路由信息。
- 使用 TCP 连接能提供可靠的服务，也简化了路由选择协议。
- 使用 TCP 连接交换路由信息的两个 BGP 发言人，彼此成为对方的邻站(neighbor)或对等站(peer) 。

BGP 发言人和自治系统 AS 的关系



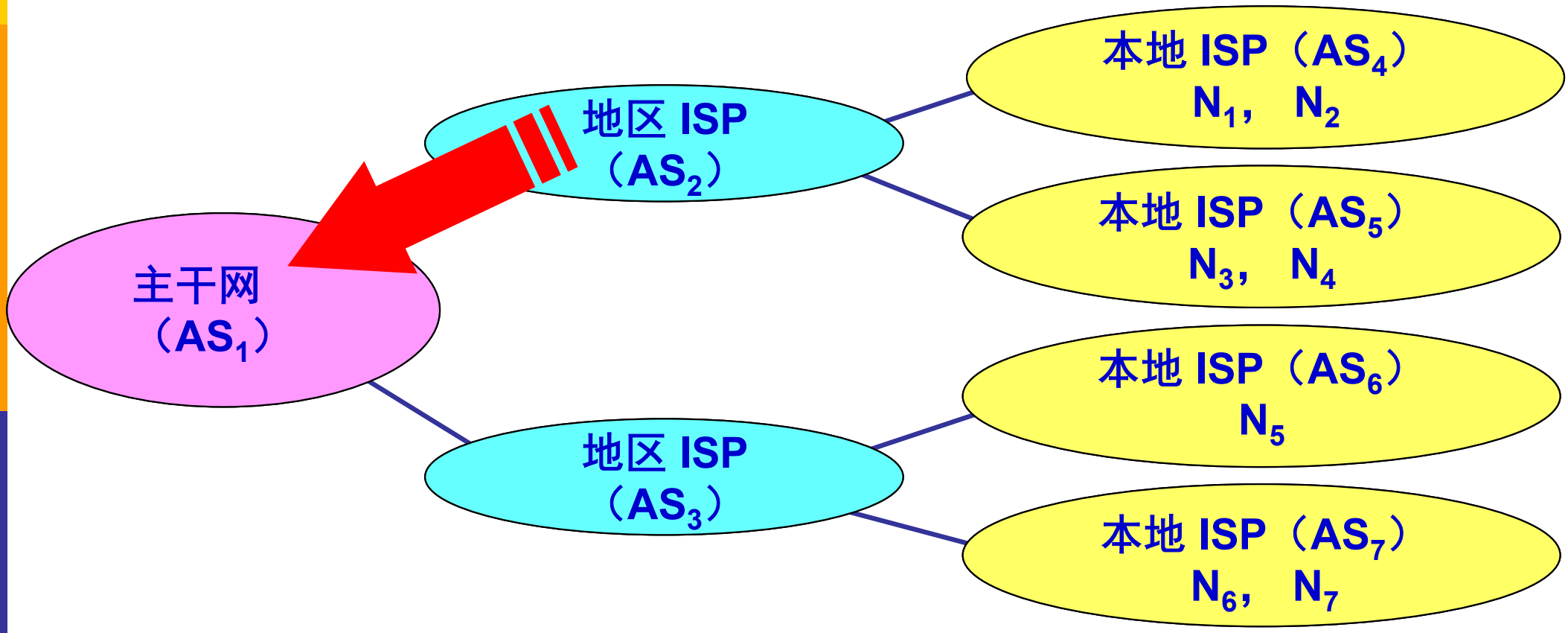
AS 的连通图举例

- BGP 所交换的网络可达性的信息就是要到达某个网络所要经过的一系列 AS。
- 当 BGP 发言人互相交换了网络可达性的信息后，各 BGP 发言人就根据所采用的策略从收到的路由信息中找出到达各 AS 的较好路由。



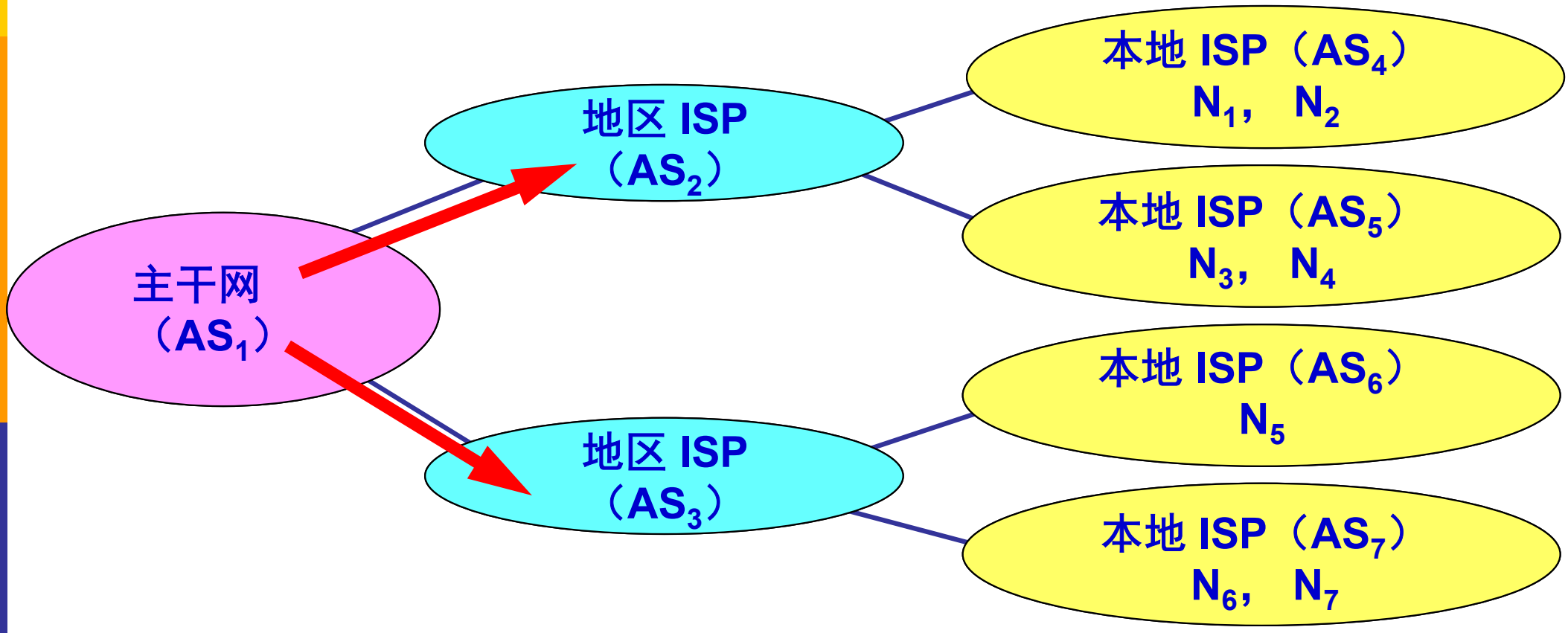
BGP 发言人交换路径向量

自治系统 AS_2 的 BGP 发言人通知主干网 AS_1 的 BGP 发言人：
“要到达网络 N_1、N_2、N_3 和 N_4 可经过 AS_2。”



BGP 发言人交换路径向量

主干网还可发出通知：“要到达网络 N5、N6 和 N7 可沿路径（AS1, AS3）。”



BGP 协议的特点

- BGP 协议交换路由信息的结点数量级是自治系统数的量级，这要比这些自治系统中的网络数少很多。
- 每一个自治系统中 BGP 发言人（或边界路由器）的数目是很少的。这样就使得自治系统之间的路由选择不致过分复杂。

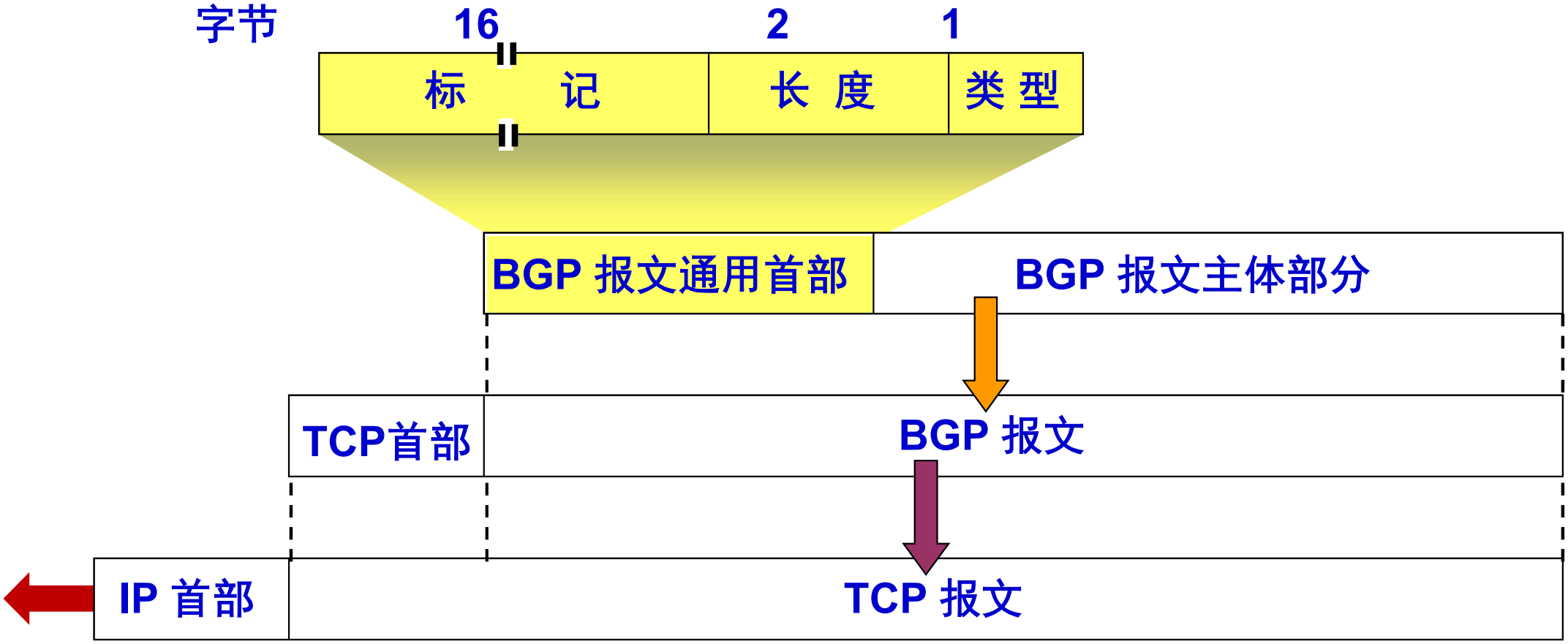
BGP 协议的特点

- **BGP 支持 CIDR，** 因此 BGP 的路由表也就应当包括目的网络前缀、下一跳路由器，以及到达该目的网络所要经过的各个自治系统序列。
- 在BGP 刚刚运行时，BGP 的邻站是交换整个的 BGP 路由表。但以后只需要在发生变化时更新有变化的部分。这样做对节省网络带宽和减少路由器的处理开销都有好处。

BGP-4 共使用四种报文

- (1) 打开 (OPEN) 报文，用来与相邻的另一个BGP发言人建立关系。
- (2) 更新 (UPDATE) 报文，用来发送某一路由的信息，以及列出要撤消的多条路由。
- (3) 保活 (KEEPALIVE) 报文，用来确认打开报文和周期性地证实邻站关系。
- (4) 通知 (NOTIFICATION) 报文，用来发送检测到的差错。

BGP 报文具有通用首部



4.5.5 路由器的构成

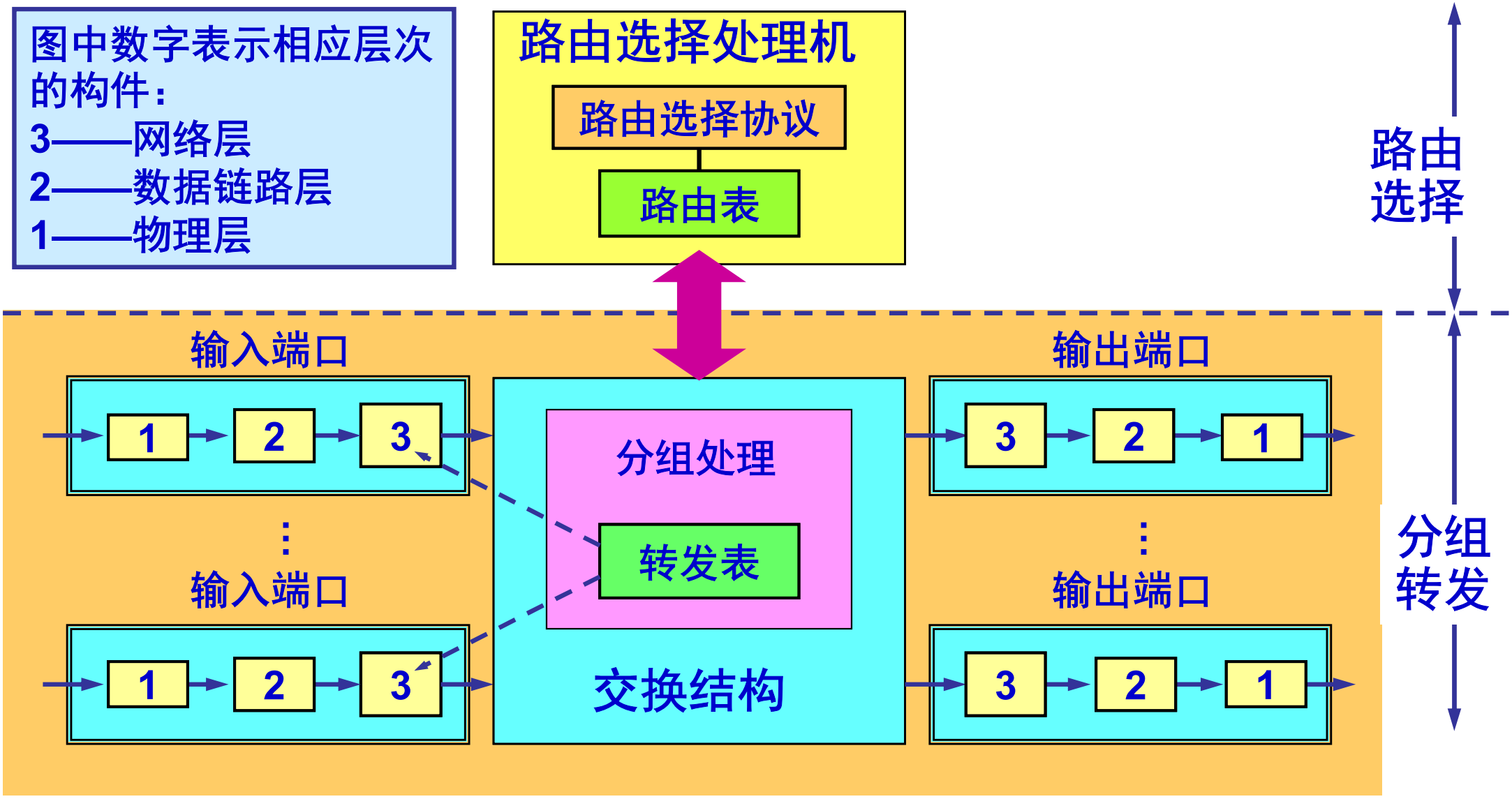
- 路由器是一种典型的网络层设备。
- 路由器是互联网中的关键设备。
- 路由器的主要作用是：
  - 连通不同的网络。
  - 选择信息传送的线路。选择通畅快捷的近路，能大大提高通信速度，减轻网络系统通信负荷，节约网络系统资源，提高网络系统畅通率，从而让网络系统发挥出更大的效益来。

1. 路由器的结构

- 路由器是一种具有多个输入端口和多个输出端口的专用计算机，其任务是转发分组。也就是说，将路由器某个输入端口收到的分组，按照分组要去的目的地（即目的网络），把该分组从路由器的某个合适的输出端口转发给下一跳路由器。
- 下一跳路由器也按照这种方法处理分组，直到该分组到达终点为止。
- 路由器的转发分组正是网络层的主要工作。

典型的路由器的结构

图中数字表示相应层次的构件：
3——网络层
2——数据链路层
1——物理层



典型的路由器的结构

- 整个的路由器结构可划分为两大部分：
  - 路由选择部分
  - 分组转发部分
- 路由选择部分
  - 也叫作控制部分，其核心构件是路由选择处理机。
  - 路由选择处理机的任务是根据所选定的路由选择协议构造出路由表，同时经常或定期地和相邻路由器交换路由信息而不断地更新和维护路由表。

典型的路由器的结构

- 分组转发部分由三部分组成：
  - 交换结构 (switching fabric)：又称为交换组织，其作用是根据转发表 (forwarding table) 对分组进行处理。
  - 一组输入端口
  - 一组输出端口

（请注意：这里的端口就是硬件接口）

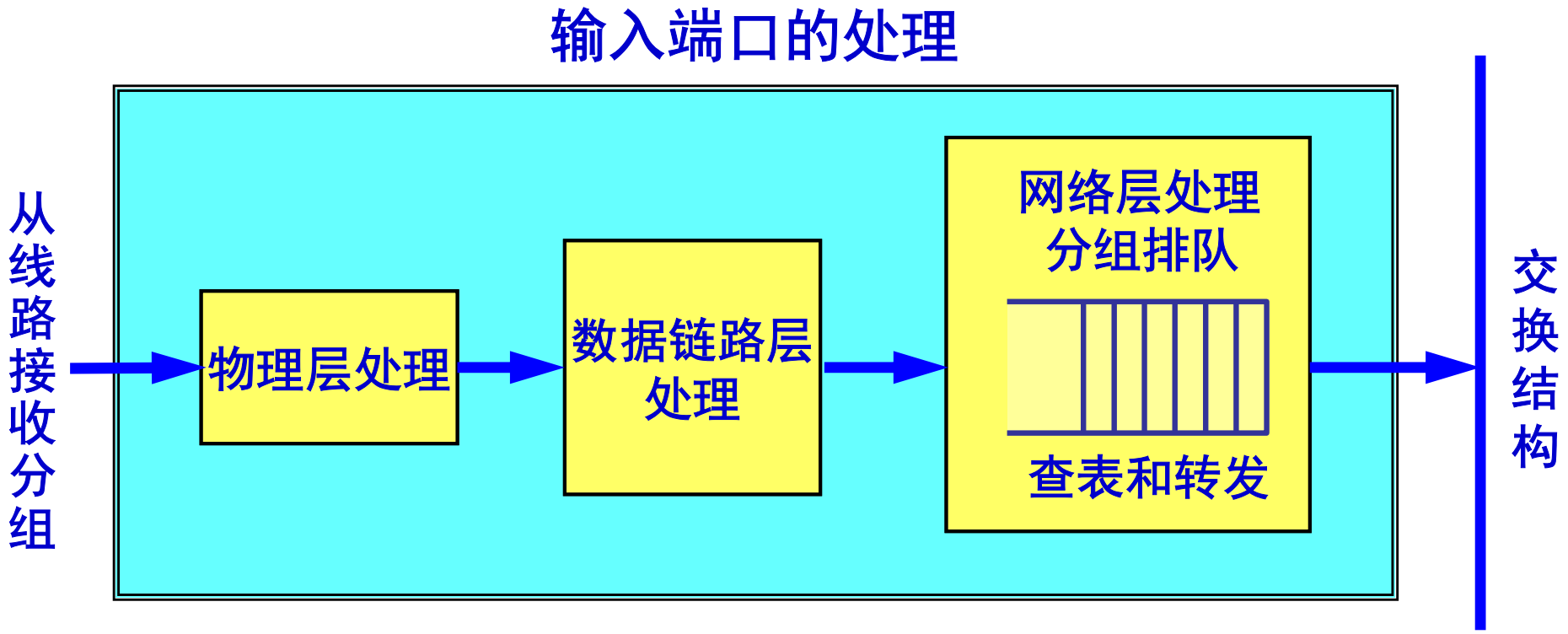
“转发”和“路由选择”的区别

- “转发”(forwarding) 就是路由器根据转发表将用户的 IP 数据报从合适的端口转发出去。
- “路由选择”(routing) 则是按照分布式算法，根据从各相邻路由器得到的关于网络拓扑的变化情况，动态地改变所选择的路由。
- 路由表是根据路由选择算法得出的。而转发表是从路由表得出的。
- 在讨论路由选择的原理时，往往不去区分转发表和路由表的区别。

输入端口对线路上收到的分组的处理

- 路由器的输入端口里面装有物理层、数据链路层和网络层的处理模块。
- 数据链路层剥去帧首部和尾部后，将分组送到网络层的队列中排队等待处理。这会产生一定的时延。
- 输入端口中的查找和转发功能在路由器的交换功能中是最重要的。

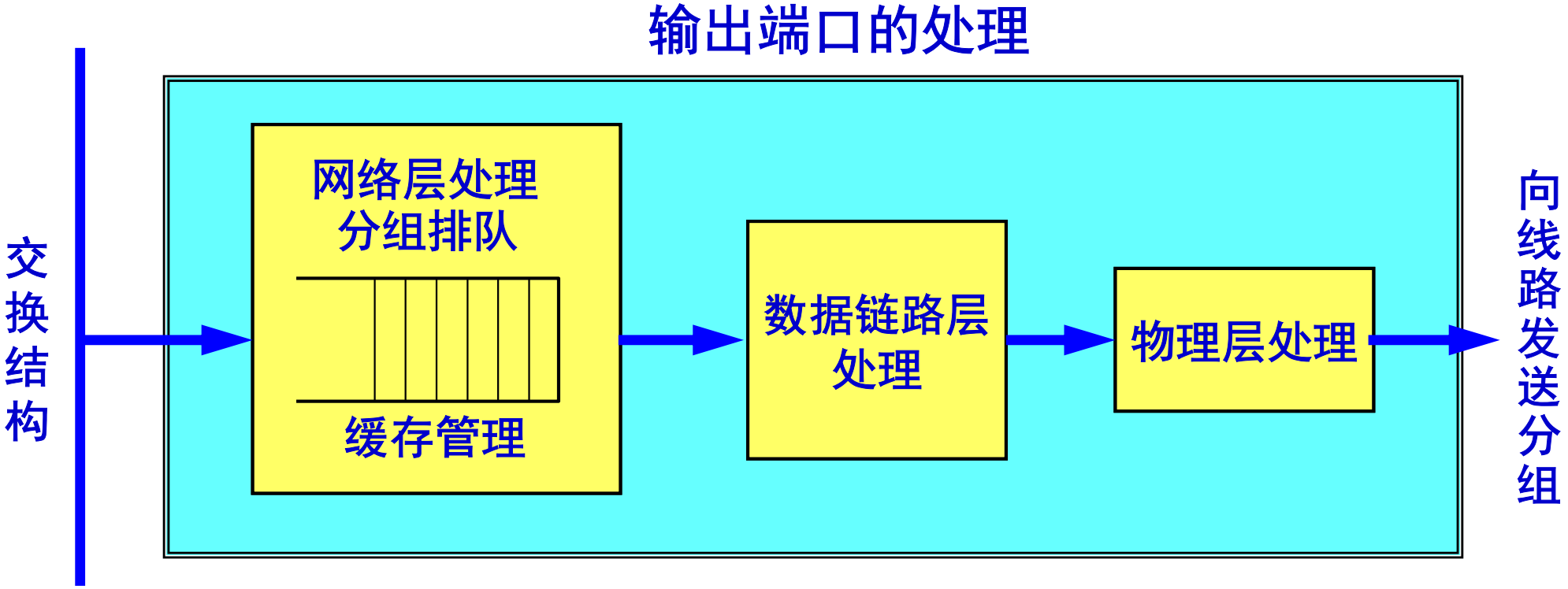
输入端口对线路上收到的分组的处理



输出端口将交换结构传送来的分组发送到线路

- 输出端口里面装有物理层、数据链路层和网络层的处理模块。
- 输出端口从交换结构接收分组，然后把它们发送到路由器外面的线路上。
- 在网络层的处理模块中设有一个缓冲区（队列）。当交换结构传送过来的分组的速率超过输出链路的发送速率时，来不及发送的分组就必须暂时存放在这个队列中。
- 数据链路层处理模块将分组加上链路层的首部和尾部，交给物理层后发送到外部线路。

输出端口将交换结构传送来的分组发送到线路



分组丢弃

- 若路由器处理分组的速率赶不上分组进入队列的速率，则队列的存储空间最终必定减少到零，这就使后面再进入队列的分组由于没有存储空间而只能被丢弃。
- 路由器中的输入或输出队列产生溢出是造成分组丢失的重要原因。

2. 交换结构

- 交换结构是路由器的关键构件。
- 正是这个交换结构把分组从一个输入端口转移到某个合适的输出端口。
- 实现交换有多种方法。常用交换方法有三种：
  - 通过存储器
  - 通过总线
  - 通过纵横交换结构

2. 交换结构

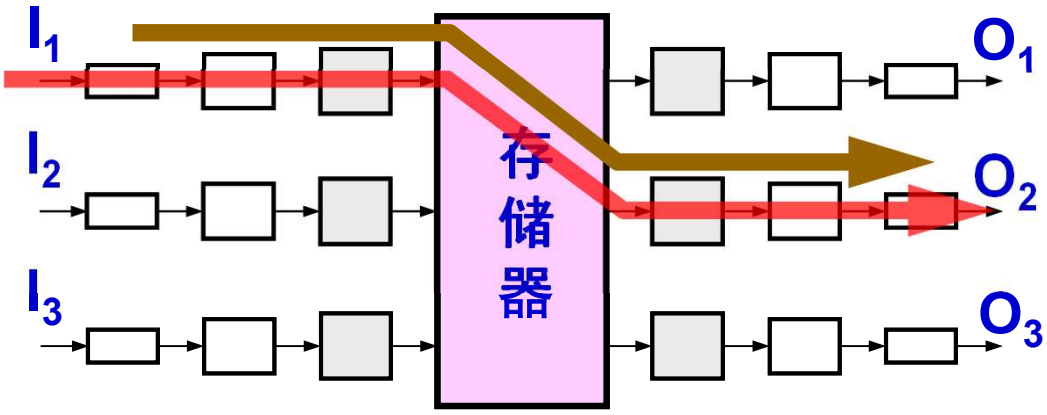
- 通过存储器
  - 当路由器的某个输入端口收到一个分组时，就用中断方式通知路由选择处理机。然后分组就从输入端口复制到存储器中。
  - 路由器处理机从分组首部提取目的地址，查找路由表，再将分组复制到合适的输出端口的缓存中。
  - 若存储器的带宽（读或写）为每秒 ***M*** 个分组，那么路由器的交换速率（即分组从输入端口传送到输出端口的速率）一定小于 ***M*/2**。

2. 交换结构

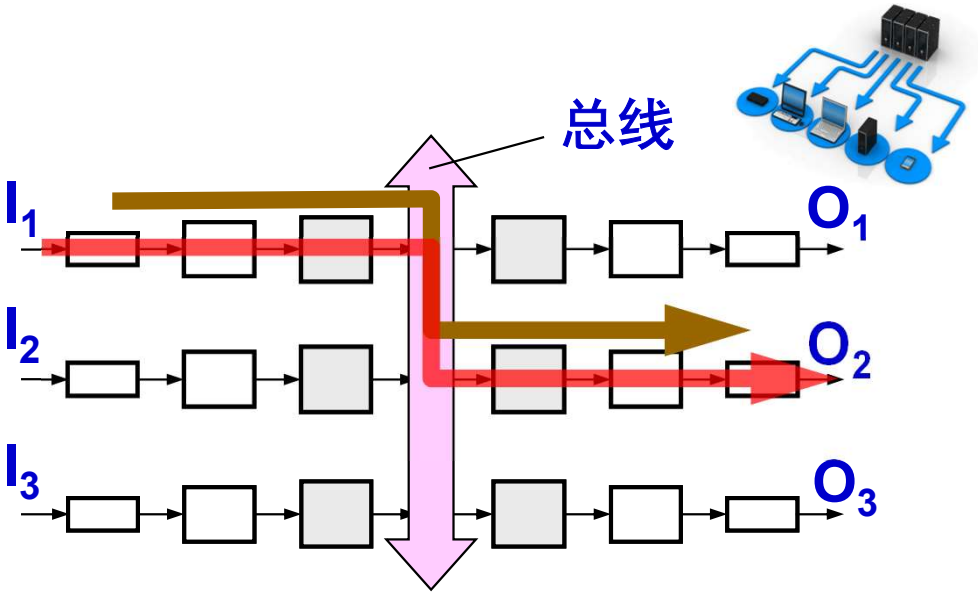
- 通过总线
  - 数据报从输入端口通过共享的总线直接传送到合适的输出端口，而不需要路由选择处理机的干预。
  - 因为每一个要转发的分组都要通过这一条总线，因此路由器的转发带宽就受总线速率的限制。
  - 现代的技术已经可以将总线的带宽提高到每秒吉比特的速率，因此许多的路由器产品都采用这种通过总线的交换方式。

2. 交换结构

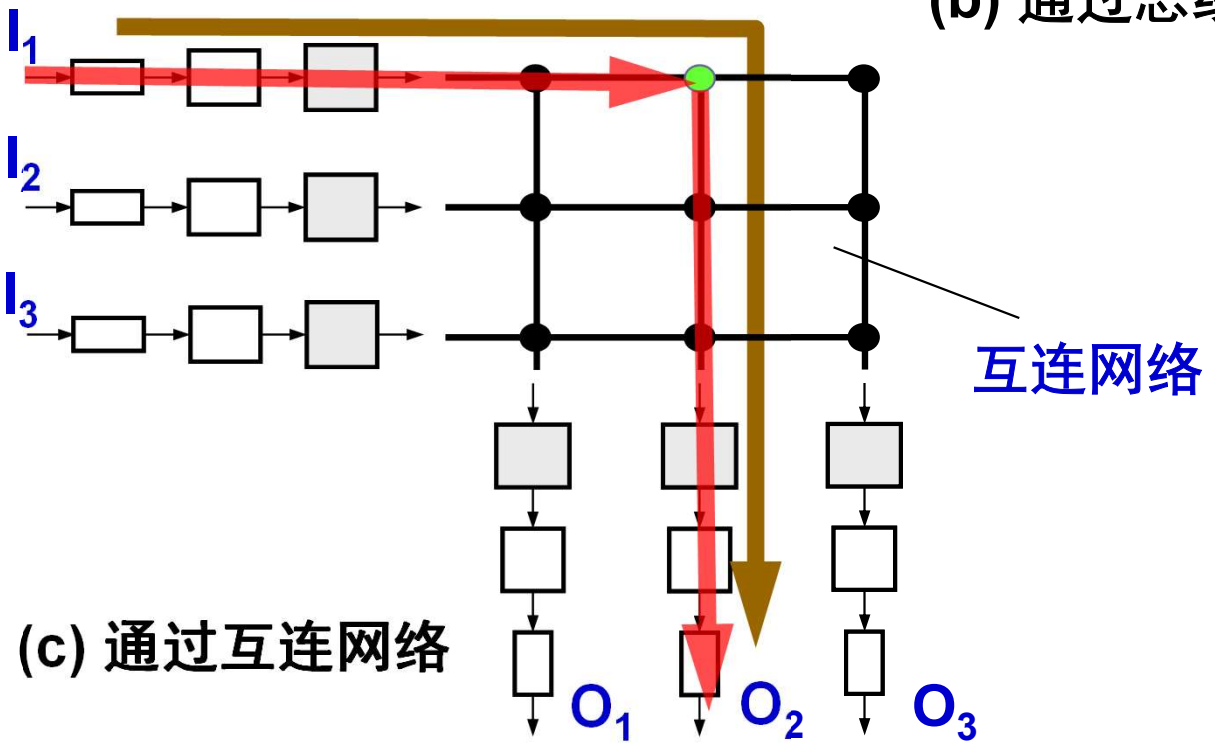
- 通过纵横交换结构 (crossbar switch fabric)
  - 这种交换结构常称为互连网络 (interconnection network)。
  - 它有2*N*条总线，可以使*N*个输入端口和*N*个输出端口相连接。
  - 当输入端口收到一个分组时，就将它发送到与该输入端口相连的水平总线上。
  - 若通向所要转发的输出端口的垂直总线是空闲的，则在这个结点将垂直总线与水平总线接通，然后将该分组转发到这个输出端口。
  - 但若该垂直总线已被占用（有另一个分组正在转发到同一个输出端口），则后到达的分组就被阻塞，必须在输入端口排队。



(a) 通过存储器



(b) 通过总线



(c) 通过互连网络

三种常用的交换方法

4.6 IPv6

- 4.6.1 IPv6的基本首部
- 4.6.2 IPv6的地址
- 4.6.3 从IPv4向IPv6过渡
- 4.6.4 ICMPv6

4.6 IPv6

- IP 是互联网的核心协议。
- 互联网经过几十年的飞速发展，到2011年2月，IPv4 的 32 位地址已经耗尽。
- ISP 已经不能再申请到新的 IP 地址块了。
- 我国在2014-2015年也逐步停止了向新用户和应用分配 IPv4 地址。
- 解决 IP 地址耗尽的根本措施就是采用具有更大地址空间的新版本的 IP，即 IPv6。

4.6.1 IPv6 的基本首部

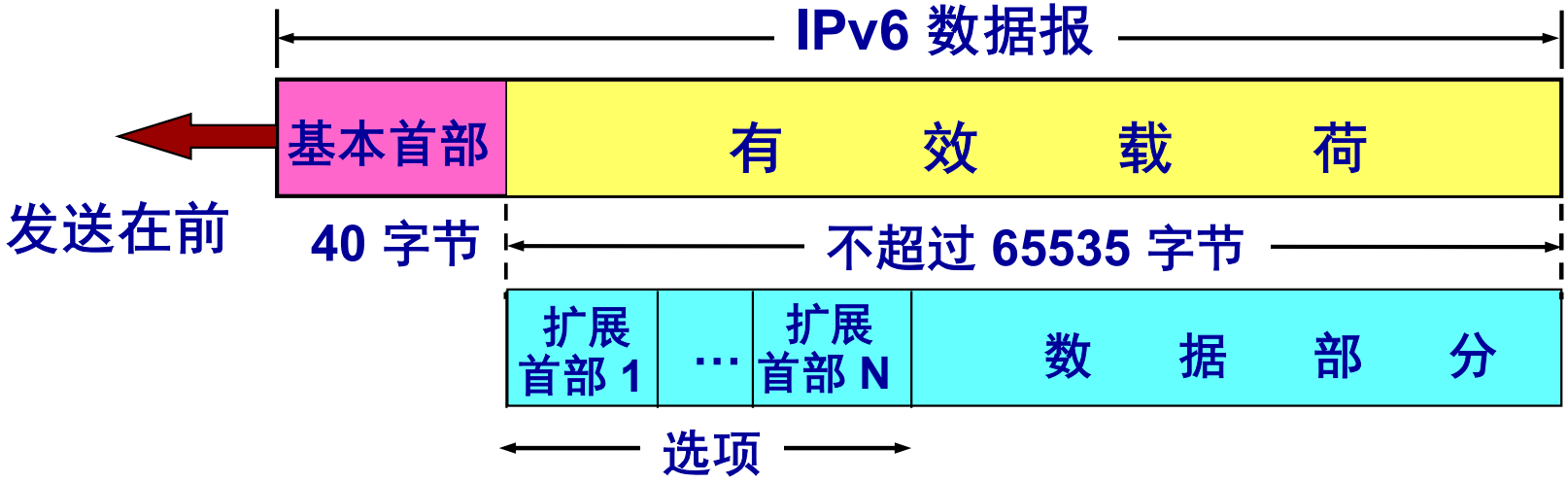
- IPv6 仍支持无连接的传送，但将协议数据单元 PDU 称为分组。为方便起见，本书仍采用数据报这一名词。
- 所引进的主要变化如下：
  - 更大的地址空间。IPv6 将地址从 IPv4 的 32 位 增大到了 128 位。
  - 扩展的地址层次结构。
  - 灵活的首部格式。 IPv6 定义了许多可选的扩展首部。
  - 改进的选项。 IPv6 允许数据报包含有选项的控制信息，其选项放在有效载荷中。

4.6.1 IPv6 的基本首部

- 所引进的主要变化如下（续）：
  - 允许协议继续扩充。
  - 支持即插即用（即自动配置）。因此 IPv6 不需要使用 DHCP。
  - 支持资源的预分配。 IPv6 支持实时视像等要求，保证一定的带宽和时延的应用。
  - IPv6 首部改为 8 字节对齐。首部长度必须是 8 字节的整数倍。原来的 IPv4 首部是 4 字节对齐。

IPv6 数据报的一般形式

- IPv6数据报由两大部分组成：
  - 基本首部 (base header)
  - 有效载荷 (payload)。有效载荷也称为净负荷。有效载荷允许有零个或多个扩展首部(extension header)，再后面是数据部分。

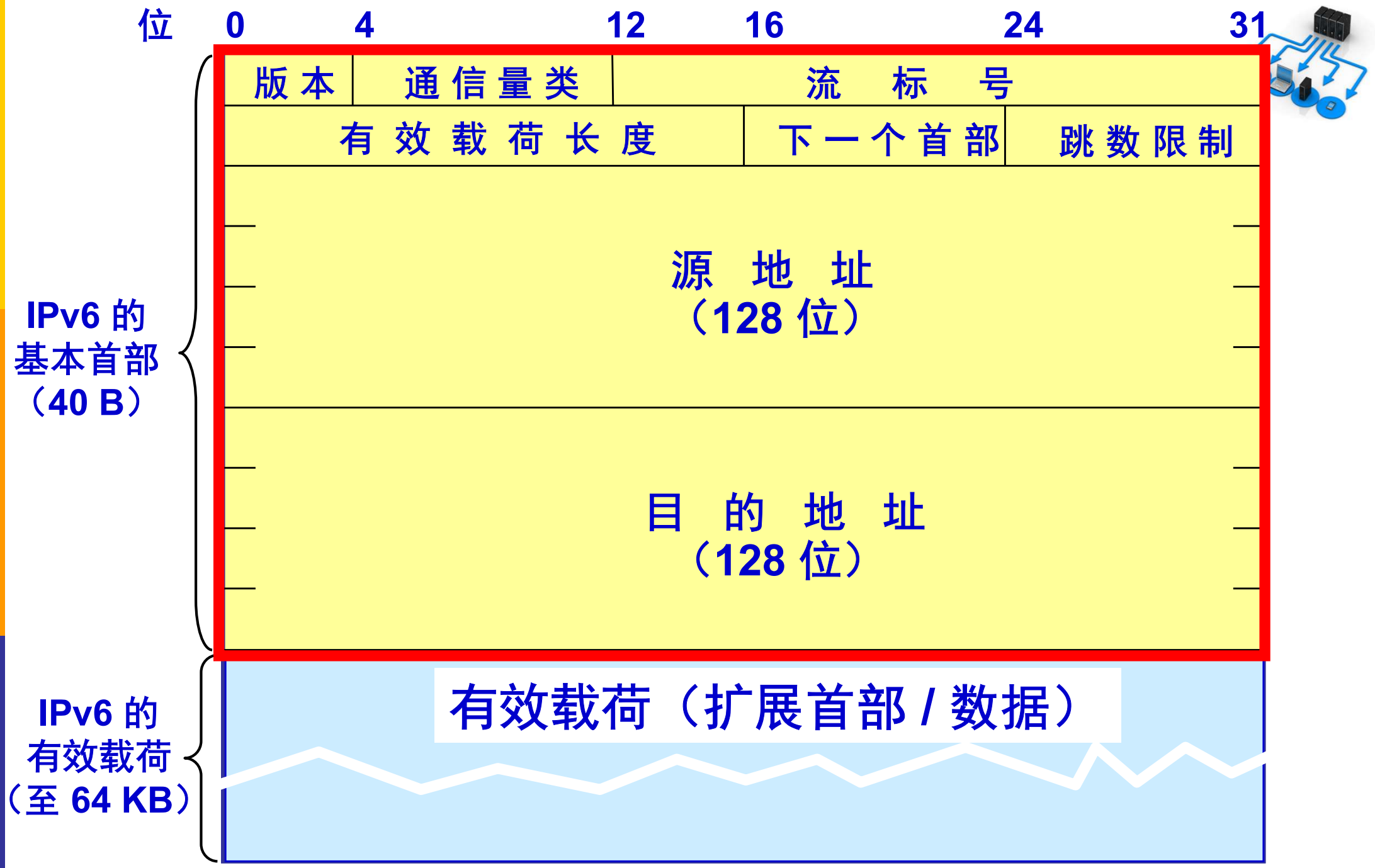


具有多个可选扩展首部的 IPv6 数据报的一般形式

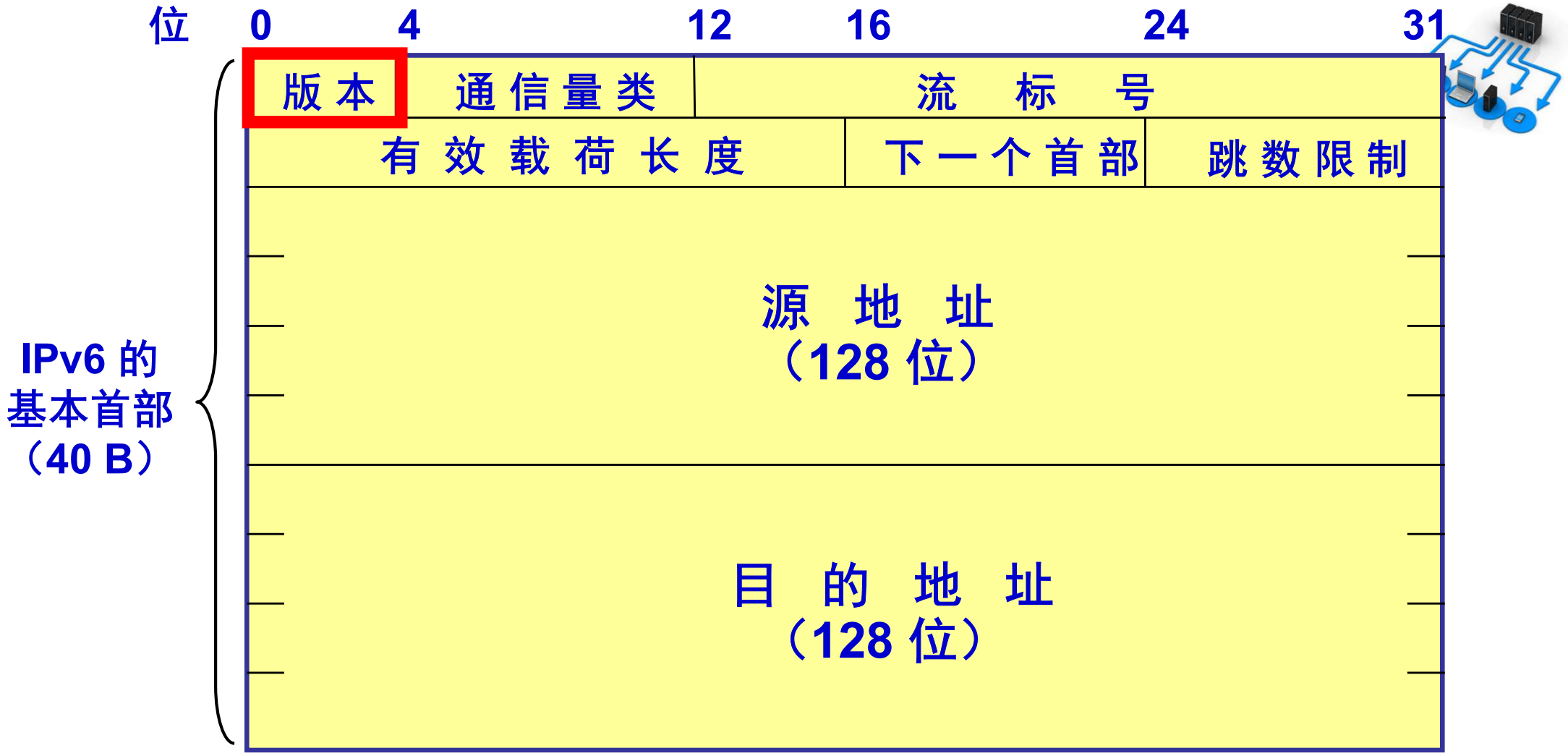
IPv6 数据报的基本首部

- IPv6 将首部长度变为固定的 40 字节，称为基本首部。
- 把首部中不必要的功能取消了，使得 IPv6 首部的字段数减少到只有 8 个。
- IPv6 对首部中的某些字段进行了如下的更改：

• 取消了首部长度字段，因为首部长度是固定的 40 字节； • 取消了服务类型字段； • 取消了总长度字段，改用有效载荷长度字段；	• 把 TTL 字段改称为跳数限制字段； • 取消了协议字段，改用下一个首部字段； • 取消了检验和字段； • 取消了选项字段，而用扩展首部来实现选项功能。



40 字节长的 IPv6 基本首部



版本(version)—— 4 位。它指明了协议的版本，对 IPv6 该字段总是 6。

通信量类(traffic class)—— 8 位。这是为了区分不同的 IPv6 数据报的类别或优先级。目前正在进行不同的通信量类性能的实验。

流标号(flow label)—— 20 位。“流”是互联网络上从特定源点到特定终点的一系列数据报，“流”所经过的路径上的路由器都保证指明的服务质量。所有属于同一个流的数据报都具有同样的流标号。

有效载荷长度(payload length)—— 16 位。它指明 IPv6 数据报除基本首部以外的字节数（所有扩展首部都算在有效载荷之内），其最大值是 64 KB。

下一个首部(next header)—— 8 位。它相当于 IPv4 的协议字段或可选字段。

跳数限制(hop limit)—— 8 位。源站在数据报发出时即设定跳数限制。路由器在转发数据报时将跳数限制字段中的值减 1。当跳数限制的值为零时，就要将此数据报丢弃。

源地址—— 128 位。是数据报的发送站的 IP 地址。

目的地址—— 128 位。是数据报的接收站的 IP 地址。

IPv6 的扩展首部

- IPv6 把原来 IPv4 首部中选项的功能都放在扩展首部中，并将扩展首部留给路径两端的源站和目的站的主机来处理。
- 数据报途中经过的路由器都不处理这些扩展首部（只有一个首部例外，即逐跳选项扩展首部）。
- 这样就大大提高了路由器的处理效率。

六种扩展首部

在 RFC 2460 中定义了六种扩展首部：

- (1) 逐跳选项
- (2) 路由选择
- (3) 分片
- (4) 鉴别
- (5) 封装安全有效载荷
- (6) 目的站选项

每一个扩展首部都由若干个字段组成，它们的长度也各不相同。但所有扩展首部的第一个字段都是8位的“下一个首部”字段。此字段的值指出了在该扩展首部后面的字段是什么。

4.6.2 IPv6 的地址

- IPv6 数据报的目的地址可以是以下三种基本类型地址之一：
  - (1) 单播 (unicast)：传统的点对点通信。
  - (2) 多播 (multicast)：一点对多点的通信。
  - (3) 任播 (anycast)：这是 IPv6 增加的一种类型。任播的目的站是一组计算机，但数据报在交付时只交付其中的一个，通常是距离最近的一个。

结点与接口

- IPv6 将实现 IPv6 的主机和路由器均称为结点。
- 一个结点就可能有多个与链路相连的接口。
- IPv6 地址是分配给结点上面的接口的。
  - 一个接口可以有多个单播地址。
  - 其中的任何一个地址都可以当作到达该结点的目的地址。即一个结点接口的单播地址可用来唯一地标志该结点。

冒号十六进制记法

- 在IPv6中，每个地址占 128 位，地址空间大于 3.4×10^{38} 。
- 为了使地址再稍简洁些，IPv6 使用冒号十六进制记法 (colon hexadecimal notation, 简写为colon hex)。
- 每个 16 位的值用十六进制值表示，各值之间用冒号分隔。例如：

  68E6:8C64:FFFF:FFFF:0:1180:960A:FFFF

- 在十六进制记法中，允许把数字前面的0省略。例如把0000中的前三个0省略，写成1个0。

零压缩

- 冒号十六进制记法可以允许零压缩 (zero compression)，即一连串连续的零可以为一对冒号所取代。

  FF05:0:0:0:0:0:0:B3　可压缩为：

  FF05::B3

- 注意：在任一地址中只能使用一次零压缩。

点分十进制记法的后缀

- 冒号十六进制记法可结合使用点分十进制记法的后缀，这种结合在 IPv4 向 IPv6 的转换阶段特别有用。
- 例如：0:0:0:0:0:0:128.10.2.1

  再使用零压缩即可得出： ::128.10.2.1
- CIDR 的斜线表示法仍然可用。
- 例如：60 位的前缀 12AB00000000CD3 可记为：

  12AB:0000:0000:CD30:0000:0000:0000:0000/60

或 12AB::CD30:0:0:0:0/60 （零压缩）

或 12AB:0:0:CD30::/60 （零压缩）

IPv6 地址分类

地址类型	二进制前缀
未指明地址	00…0（128位），可记为 ::/128。
环回地址	00…1（128位），可记为 ::1/128。
多播地址	11111111（8位），可记为 FF00::/8。
本地链路单播地址	1111111010（10位），可记为 FE80::/10。
全球单播地址	（除上述四种外，所有其他的二进制前缀）

IPv6 地址分类

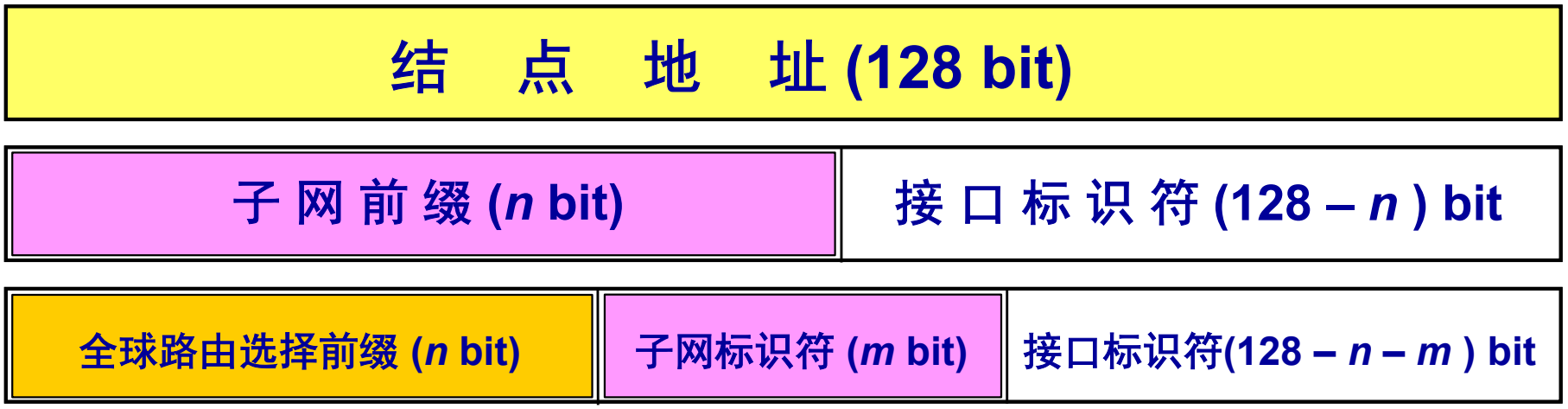
- 未指明地址
  - 这是 16 字节的全 0 地址，可缩写为两个冒号“::”。
  - 这个地址只能为还没有配置到一个标准的 IP 地址的主机当作源地址使用。
  - 这类地址仅此一个。
- 环回地址
  - 即 0:0:0:0:0:0:0:1（记为 ::1）。
  - 作用和IPv4的环回地址一样。
  - 这类地址也是仅此一个。

IPv6 地址分类

- 多播地址
  - 功能和 IPv4 的一样。
  - 这类地址占 IPv6 地址总数的 1/256。
- 本地链路单播地址 (Link-Local Unicast Address)
  - 有些单位的网络使用 TCP/IP 协议，但并没有连接到互联网上。连接在这样的网络上的主机都可以使用这种本地地址进行通信，但不能和互联网上的其他主机通信。
  - 这类地址占 IPv6 地址总数的 1/1024。

IPv6 地址分类

- 全球单播地址
  - IPv6 的这一类单播地址是使用得最多的一类。
  - 曾提出过多种方案来进一步划分这128位的单播地址。
  - 根据2006年发布的草案标准RFC 4291的建议，IPv6 单播地址的划分方法非常灵活。



IPv6 单播地址的几种划分方法

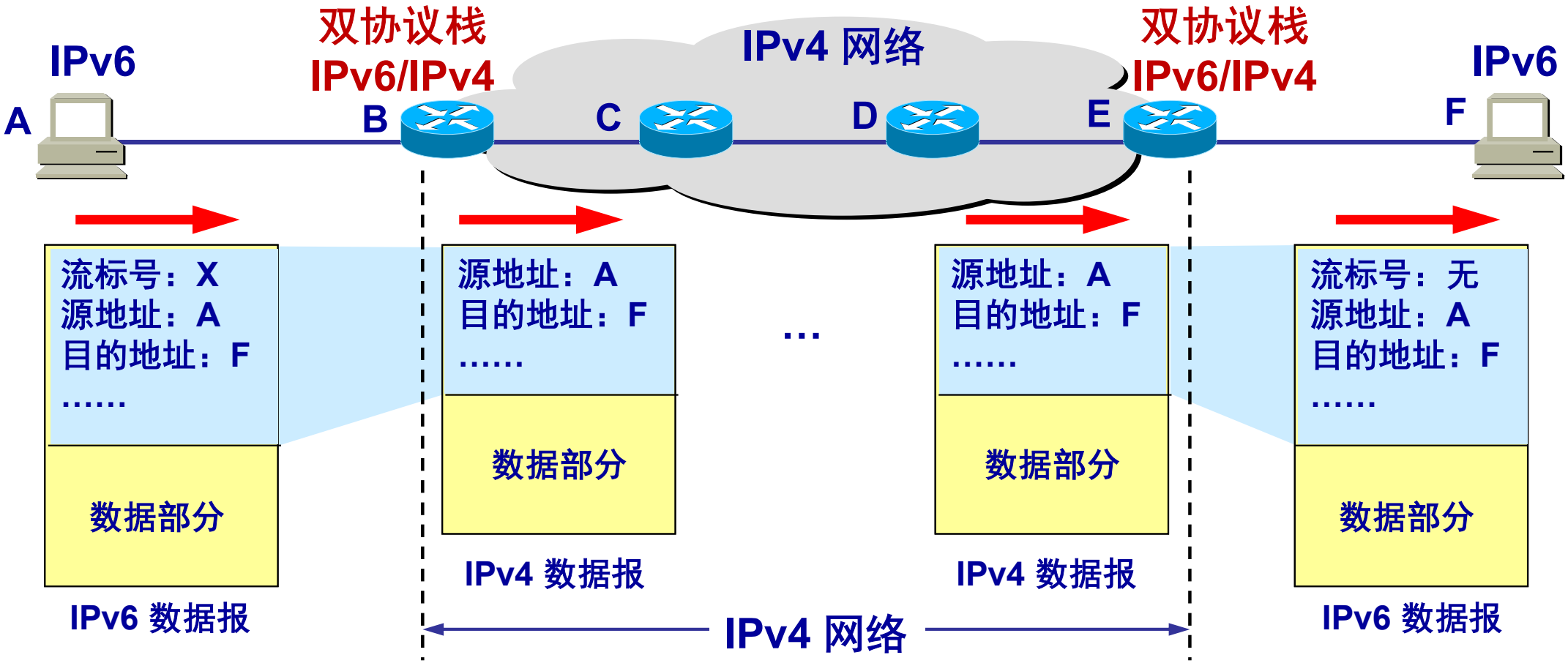
4.6.3 从 IPv4 向 IPv6 过渡

- 向 IPv6 过渡只能采用逐步演进的办法，同时，还必须使新安装的 IPv6 系统能够向后兼容：IPv6 系统必须能够接收和转发 IPv4 分组，并且能够为 IPv4 分组选择路由。
- 两种向 IPv6 过渡的策略：
  - 使用双协议栈
  - 使用隧道技术

双协议栈

- 双协议栈(dual stack)是指在完全过渡到 IPv6 之前，使一部分主机（或路由器）装有两个协议栈，一个 IPv4 和一个 IPv6。
- 双协议栈的主机（或路由器）记为 IPv6/IPv4，表明它同时具有两种 IP 地址：一个 IPv6 地址和一个 IPv4 地址。
- 双协议栈主机在和 IPv6 主机通信时是采用 IPv6 地址，而和 IPv4 主机通信时就采用 IPv4 地址。
- 根据 DNS 返回的地址类型可以确定使用 IPv4 地址还是 IPv6 地址。

双协议栈

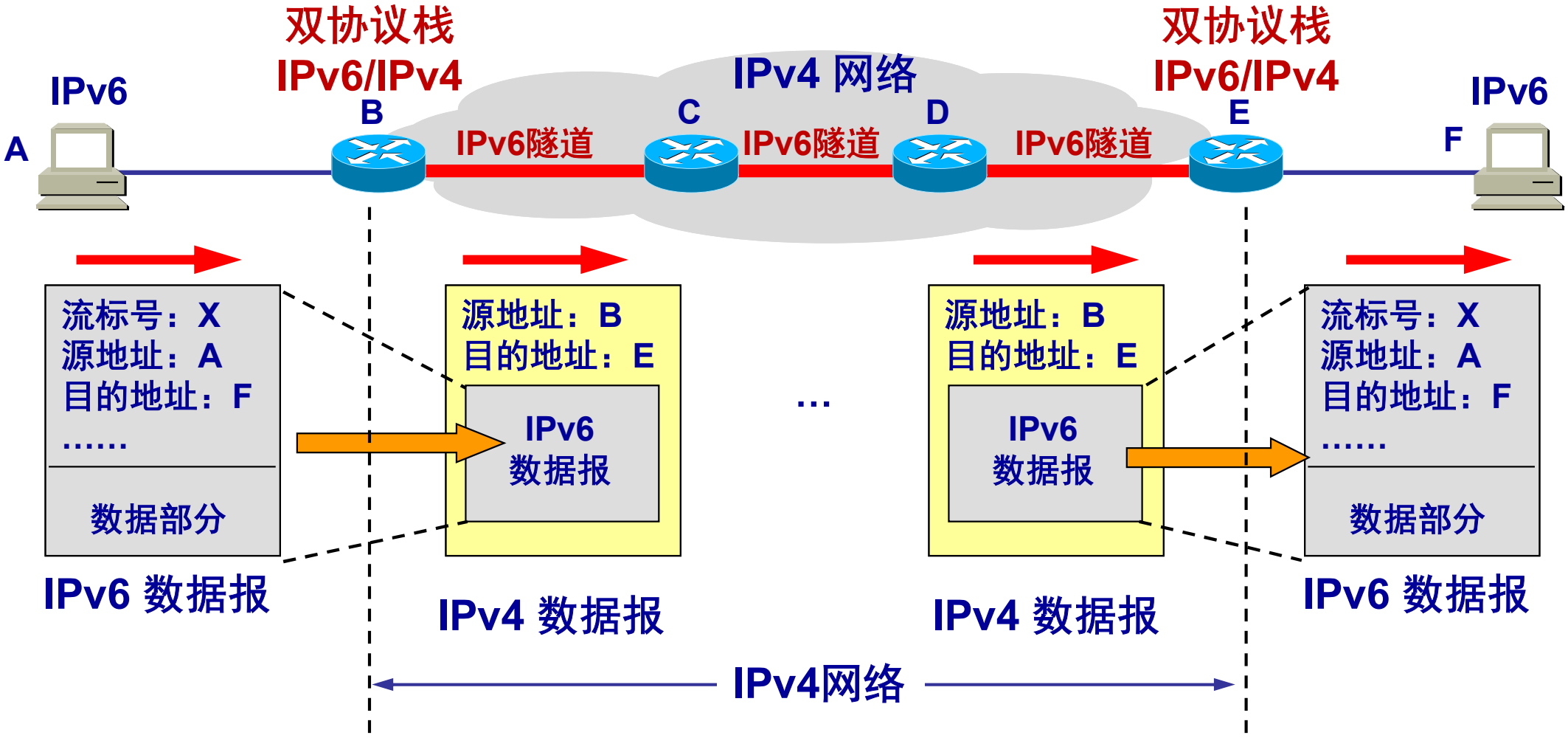


使用双协议栈进行从 IPv4 到 IPv6 的过渡

隧道技术

- 在 IPv6 数据报要进入IPv4网络时，把 IPv6 数据报封装成为 IPv4 数据报，整个的 IPv6 数据报变成了 IPv4 数据报的数据部分。
- 当 IPv4 数据报离开 IPv4 网络中的隧道时，再把数据部分（即原来的 IPv6 数据报）交给主机的 IPv6 协议栈。

隧道技术



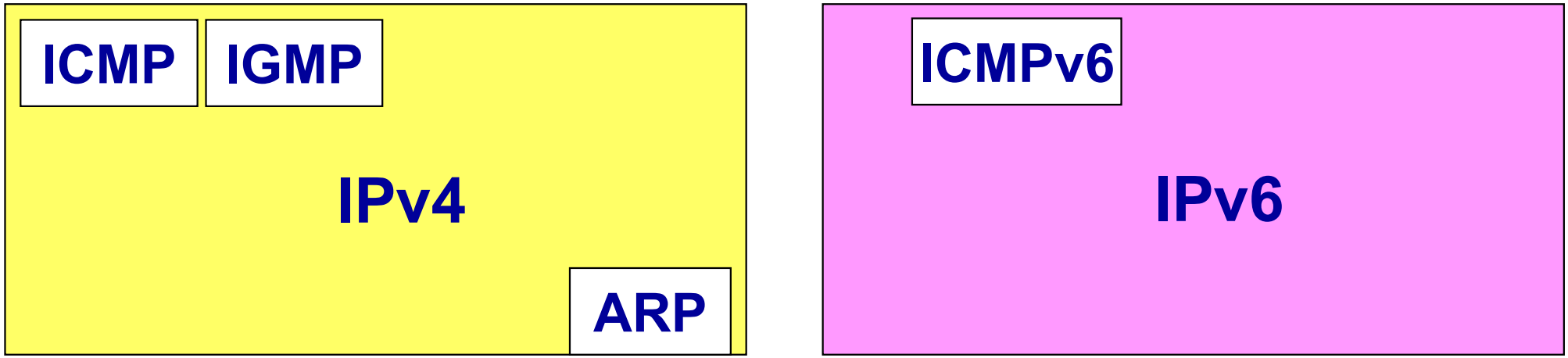
使用隧道技术进行从 IPv4 到 IPv6 的过渡

4.6.4 ICMPv6

- IPv6 也不保证数据报的可靠交付，因为互联网中的路由器可能会丢弃数据报。
- 因此 IPv6 也需要使用 ICMP 来反馈一些差错信息。新的版本称为 ICMPv6。

4.6.4 ICMPv6

- 地址解析协议 ARP 和网际组管理协议 IGMP 协议的功能都已被合并到 ICMPv6 中。

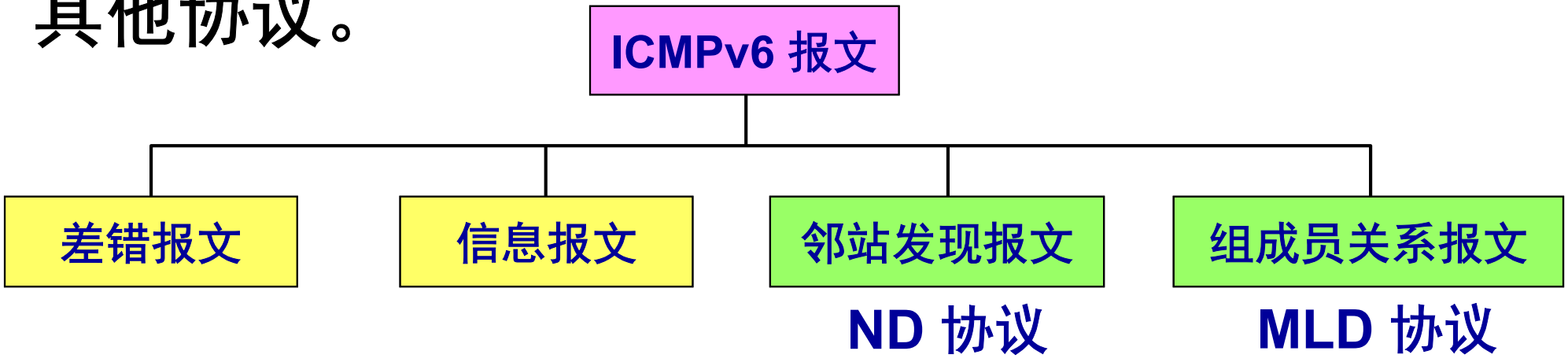


版本 4 中的网络层　　版本 6 中的网络层

新旧版本中的网络层的比较

ICMPv6 报文的分类

- CMPv6 是面向报文的协议，它利用报文来报告差错，获取信息，探测邻站或管理多播通信。
- ICMPv6 还增加了几个定义报文的功能及含义的其他协议。



ND (Neighbor-Discovery)：邻站发现
MLD (Multicast Listener Delivery)：多播听众交付

ICMPv6 报文的分类

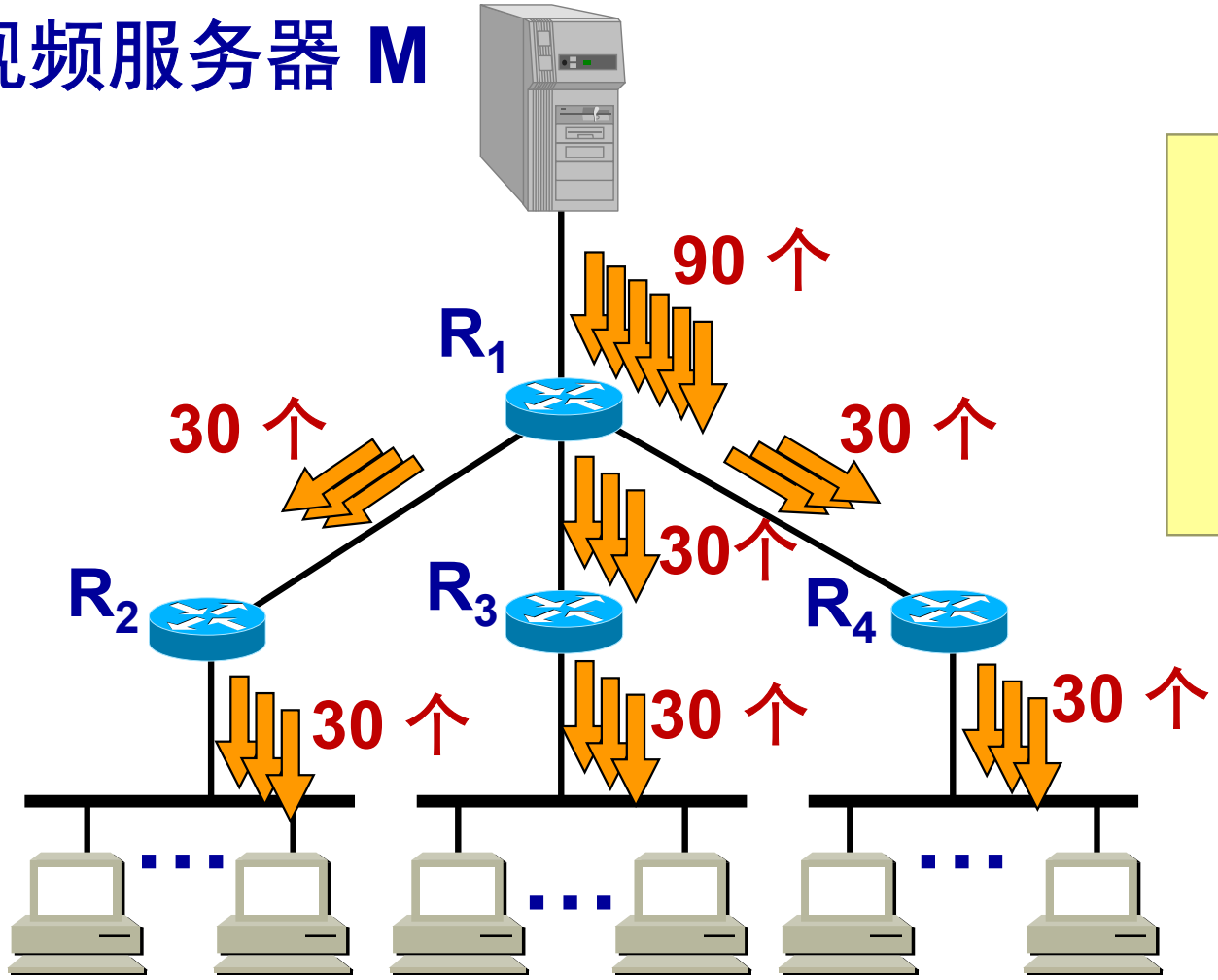
4.7 IP 多播

- 4.7.1 IP 多播的基本概念
- 4.7.2 在局域网上进行硬件多播
- 4.7.3 网际组管理协议 IGMP 和多播路由选择协议

4.7.1 IP 多播的基本概念

- IP 多播 (multicast，以前曾译为组播) 已成为互联网的一个热门课题。
- 目的：更好第支持一对多通信。
- 一对多通信：一个源点发送到许多个终点。
  - 例如，实时信息的交付（如新闻、股市行情等），软件更新，交互式会议及其他多媒体通信。

多播可大大节约网络资源

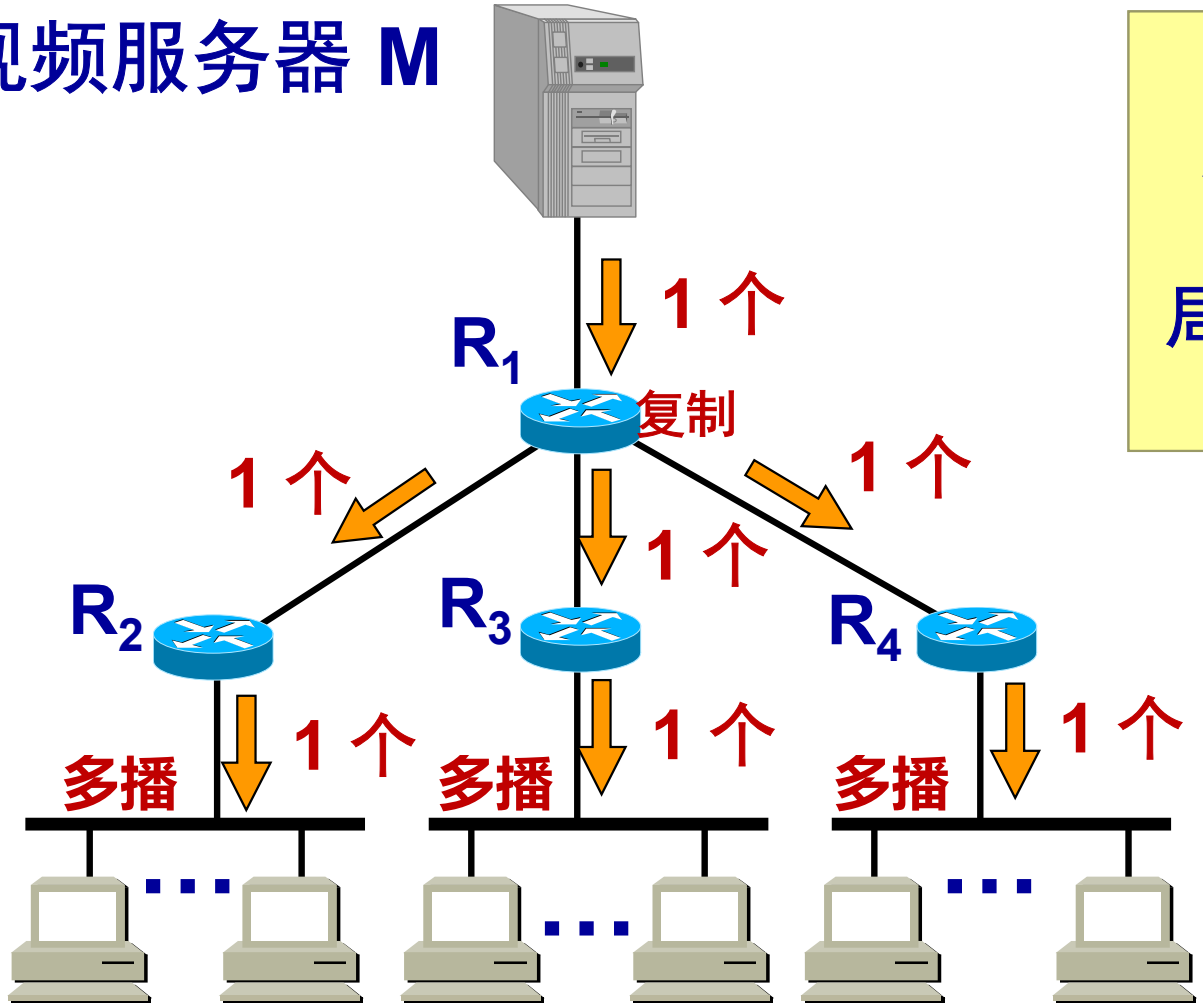


采用单播方式，
向 90 台主机传送
同样的视频节目
需要发送 90 个单播

共有 90 个主机接收视频节目

单播

多播可大大节约网络资源



采用多播方式，
只需发送一次到多播组。
路由器复制分组。
局域网具有硬件多播功能，
不需要复制分组。

当多播组的主机数很大时（如成千上万个），采用多播方式就可明显地减轻网络中各种资源的消耗。

IP 多播

- 在互联网上进行多播就叫作 **IP 多播**。
- 互联网范围的多播要靠路由器来实现。
- 能够运行多播协议的路由器称为**多播路由器 (multicast router)**。当然它也可以转发普通的单播IP数据报。
- 从1992年起，在互联网上开始试验虚拟的**多播主干网MBONE (Multicast Backbone On the InterNEt)**。 现在多播主干网已经有了相当大的规模。

多播 IP 地址

- IP 多播所传送的分组需要使用多播 IP 地址。
- 在多播数据报的目的地址写入的是多播组的标识符。
- 多播组的标识符就是 IP 地址中的 D 类地址（多播地址）。
- 每一个D类地址标志一个多播组。
- 多播地址只能用于目的地址，不能用于源地址。

多播数据报

- 多播数据报和一般的 IP 数据报的区别就是它使用 D 类 IP 地址作为目的地址，并且首部中的协议字段值是2，表明使用网际组管理协议 IGMP。
- 多播数据报也是“尽最大努力交付”，不保证一定能够交付多播组内的所有成员。
- 对多播数据报不产生 ICMP 差错报文。因此，若在 PING 命令后面键入多播地址，将永远不会收到响应。

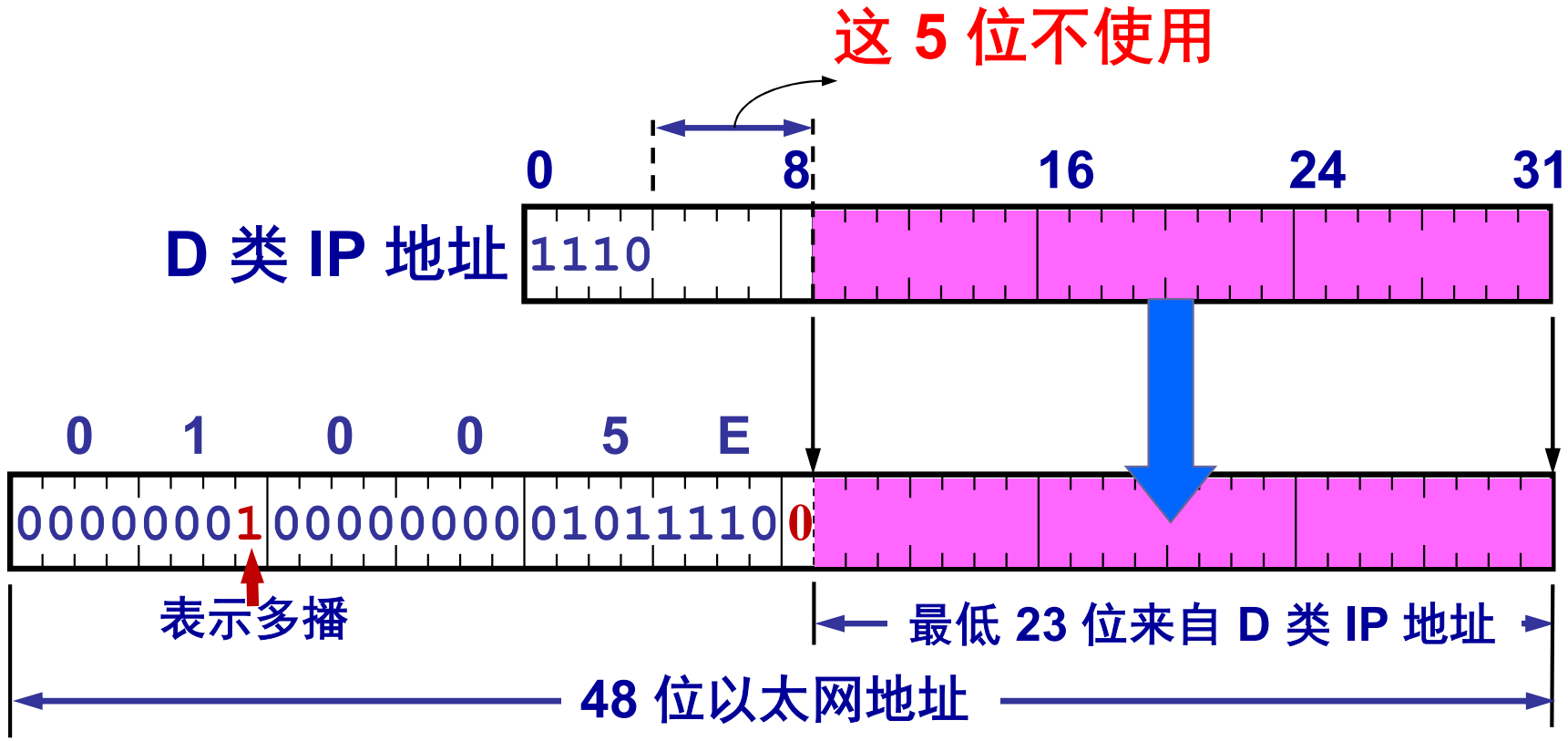
4.7.2 在局域网上进行硬件多播

- 互联网号码指派管理局 IANA 拥有的以太网地址块的高 24 位为 00-00-5E。
- 因此 TCP/IP 协议使用的以太网多播地址块的范围是

  从 00-00-5E-00-00-00

  到 00-00-5E-FF-FF-FF
- 不难看出，在每一个地址中，只有23位可用作多播。
- D 类 IP 地址可供分配的有 28 位，在这 28 位中的前 5 位不能用来构成以太网硬件地址。

D 类 IP 地址与以太网多播地址的映射关系



D 类 IP 地址与以太网多播地址的映射关系

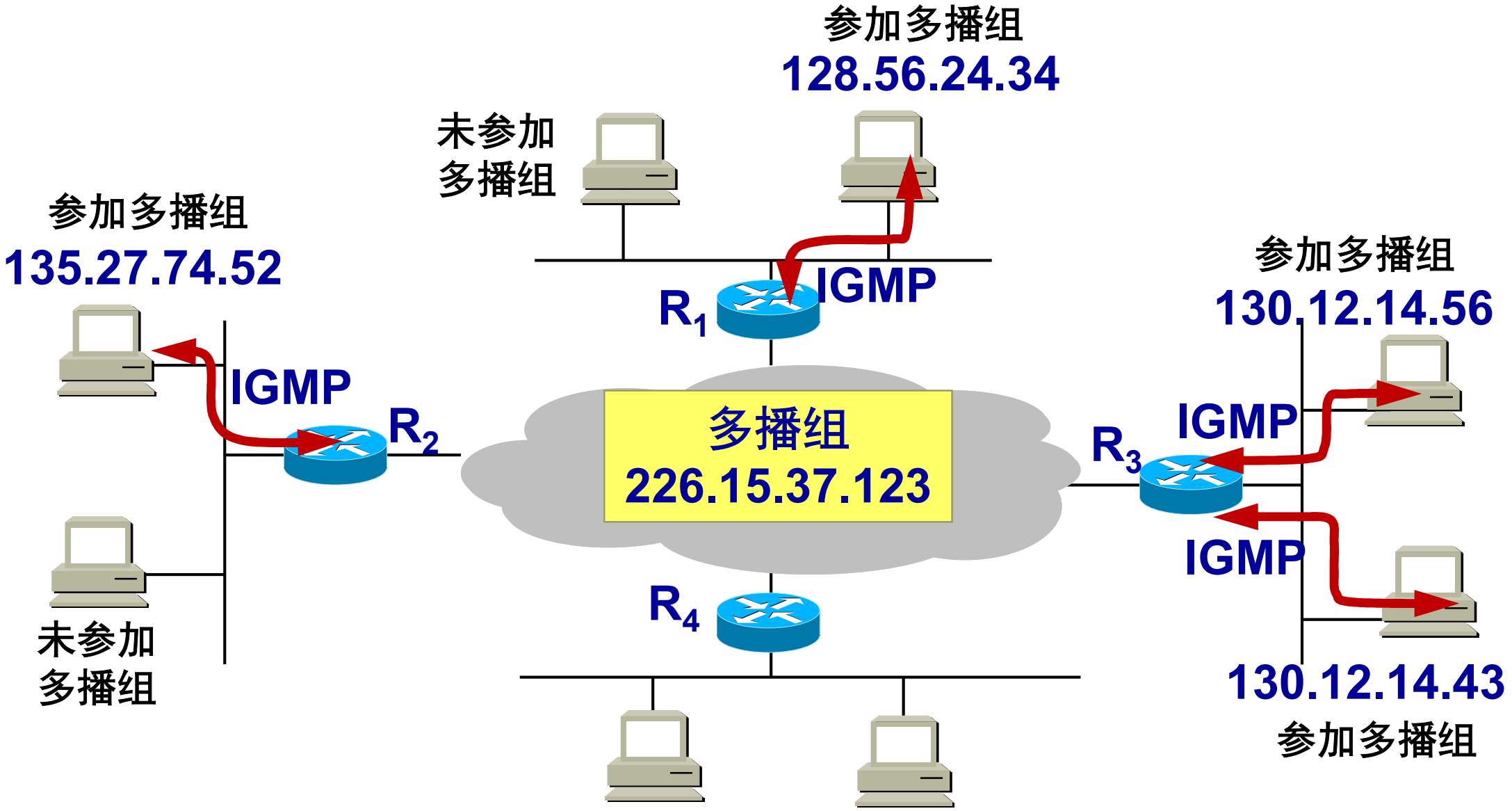
- 由于多播IP地址与以太网硬件地址的映射关系不是唯一的，因此收到多播数据报的主机，还要在 IP 层利用软件进行过滤，把不是本主机要接收的数据报丢弃。

4.7.3 网际组管理协议 IGMP 和多播路由选择协议

1. IP 多播需要两种协议

- 为了使路由器知道多播组成员的信息，需要利用网际组管理协议 IGMP (Internet Group Management Protocol)。
- 连接在局域网上的多播路由器还必须和互联网上的其他多播路由器协同工作，以便把多播数据报用最小代价传送给所有的组成员。这就需要使用多播路由选择协议。

IGMP 使多播路由器知道多播组成员信息

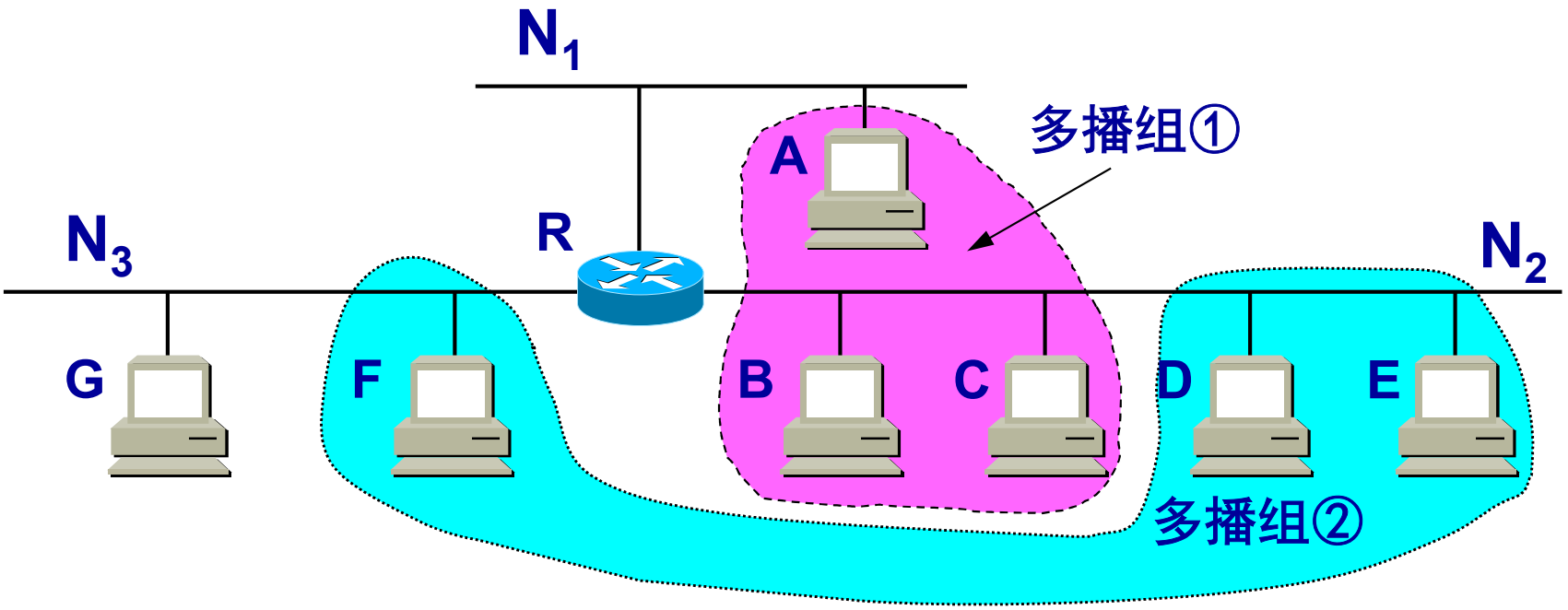


IGMP 的使用范围

- IGMP 并非在互联网范围内对所有多播组成员进行管理的协议。
- IGMP 不知道 IP 多播组包含的成员数，也不知道这些成员都分布在哪些网络上。
- IGMP 协议是让连接在本地局域网上的多播路由器知道本局域网上是否有主机（严格讲，是主机上的某个进程）参加或退出了某个多播组。

多播路由选择协议更为复杂

多播路由选择协议比单播路由选择协议复杂得多。



多播路由选择协议更为复杂

- 多播转发必须动态地适应多播组成员的变化（这时网络拓扑并未发生变化）。请注意，单播路由选择通常是在网络拓扑发生变化时才需要更新路由。
- 多播路由器在转发多播数据报时，不能仅仅根据多播数据报中的目的地址，而是还要考虑这个多播数据报从什么地方来和要到什么地方去。
- 多播数据报可以由没有加入多播组的主机发出，也可以通过没有组成员接入的网络。

2. 网际组管理协议 IGMP

- 1989 年公布的 RFC 1112（IGMPv1）早已成为了互联网的标准协议。
- 1997 年公布的 RFC 2236（IGMPv2，建议标准）对 IGMPv1 进行了更新。
- 2002 年 10 月公布了 RFC 3376（IGMPv3，建议标准），宣布 RFC 2236（IGMPv2）是陈旧的。

IGMP 是整个网际协议 IP 的一个组成部分

- 和 ICMP 相似，IGMP 使用 IP 数据报传递其报文（即 IGMP 报文加上 IP 首部构成 IP 数据报），但它也向 IP 提供服务。
- 因此，我们不把 IGMP 看成是一个单独的协议，而是属于整个网际协议 IP 的一个组成部分。

IGMP 工作可分为两个阶段

- 第一阶段：加入多播组。
  - 当某个主机加入新的多播组时，该主机应向多播组的多播地址发送IGMP 报文，声明自己要成为该组的成员。
  - 本地的多播路由器收到 IGMP 报文后，将组成员关系转发给互联网上的其他多播路由器。

IGMP 可分为两个阶段

- 第二阶段：探询组成员变化情况。
  - 因为组成员关系是动态的，因此本地多播路由器要周期性地探询本地局域网上的主机，以便知道这些主机是否还继续是组的成员。
  - 只要对某个组有一个主机响应，那么多播路由器就认为这个组是活跃的。
  - 但一个组在经过几次的探询后仍然没有一个主机响应，则不再将该组的成员关系转发给其他的多播路由器。

IGMP 采用的一些具体措施

- 在主机和多播路由器之间的所有通信都是使用 IP 多播。
- 多播路由器在探询组成员关系时，只需要对所有的组发送一个请求信息的询问报文，而不需要对每一个组发送一个询问报文。默认的询问速率是每 125 秒发送一次。
- 当同一个网络上连接有几个多播路由器时，它们能够迅速和有效地选择其中的一个来探询主机的成员关系。

IGMP 采用的一些具体措施（续）

- 在 IGMP 的询问报文中有一个数值 ***N***，它指明一个最长响应时间（默认值为 10秒）。当收到询问时，主机在 0 到 ***N*** 之间随机选择发送响应所需经过的时延。对应于最小时延的响应最先发送。
- 同一个组内的每一个主机都要监听响应，只要有本组的其他主机先发送了响应，自己就可以不再发送响应了。

3. 多播路由选择

- 多播路由选择协议尚未标准化。
- 一个多播组中的成员是动态变化的，随时会有主机加入或离开这个多播组。
- 多播路由选择实际上就是要找出以源主机为根结点的多播转发树。
- 在多播转发树上的路由器不会收到重复的多播数据报。
- 对不同的多播组对应于不同的多播转发树。
- 同一个多播组，对不同的源点也会有不同的多播转发树。

3. 多播路由选择

- 多播路由选择协议在转发多播数据报时使用三种方法：
  - (1) 洪泛与剪除
  - (2) 隧道技术 (tunneling)
  - (3) 基于核心的发现技术

(1) 洪泛与剪除

- 这种方法适合于较小的多播组，而所有的组成员接入的局域网也是相邻接的。
- 一开始，路由器转发多播数据报使用洪泛的方法（这就是广播）。
- 为了避免兜圈子，采用了叫作反向路径广播 **RPB (Reverse Path Broadcasting)**的策略。

RPB 的要点

- 路由器收到多播数据报时，先检查它是否是从源点经最短路径传送来的。
- 若是，就向所有其他方向转发刚才收到的多播数据报（但进入的方向除外），否则就丢弃而不转发。
- 如果存在几条同样长度的最短路径，那么只能选择一条最短路径，选择的准则就是看这几条最短路径中的相邻路由器谁的 IP 地址最小。
- 最后就得出了用来转发多播数据报的多播转发树，以后就按这个多播转发树转发多播数据报。避免了多播数据报的兜圈子，同时每一个路由器也不会接收重复的多播数据报。

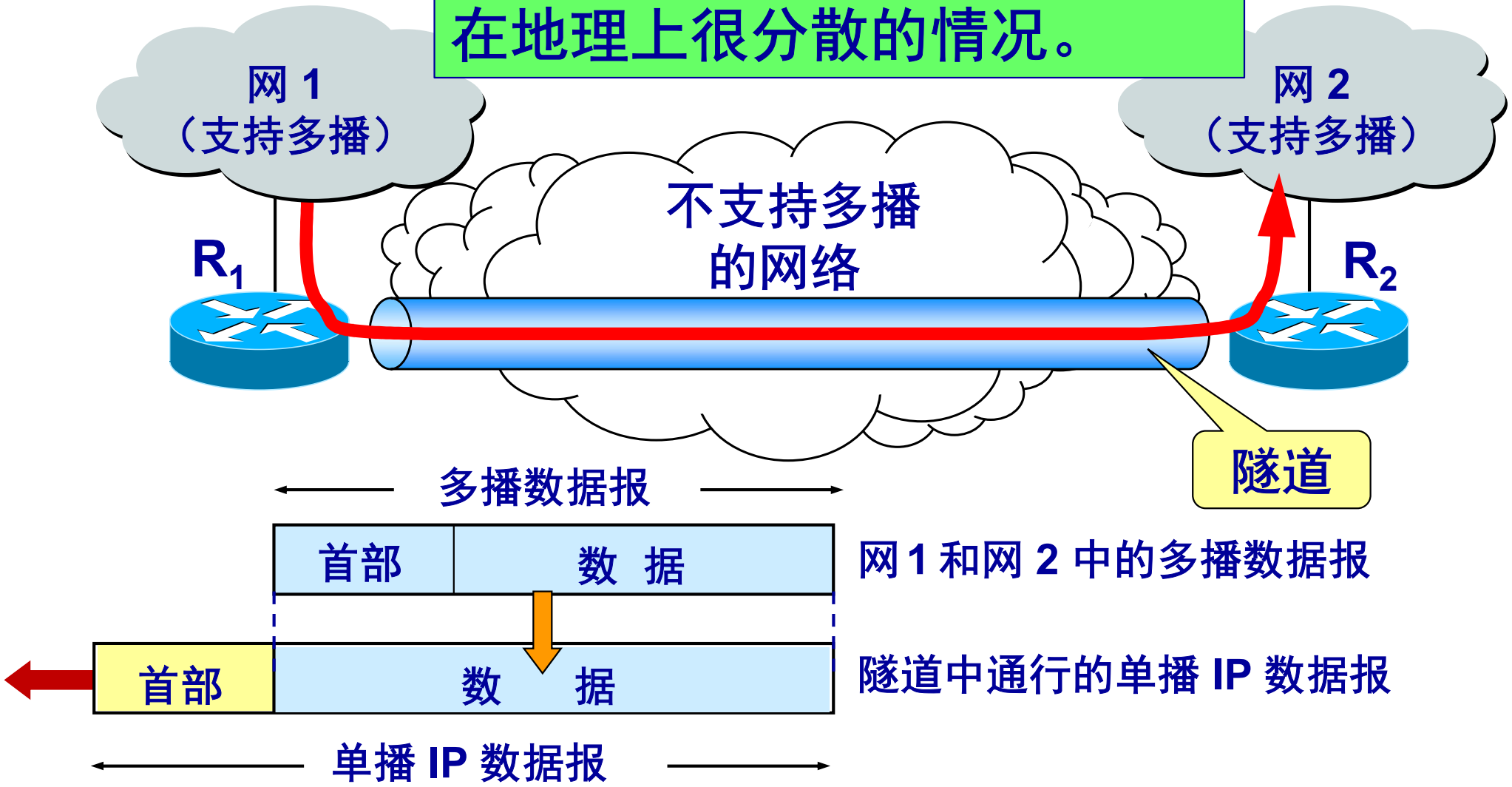
RPB 的要点

- 如果在多播转发树上的某个路由器发现它的下游树枝（即叶节点方向）已没有该多播组的成员，就应把它和下游的树枝一起剪除。
- 当某个树枝有新增加的组成员时，可以再接入到多播转发树上。

反向路径广播 RPB 和剪除

(2) 隧道技术 (tunneling)

隧道技术适用于多播组的位置在地理上很分散的情况。



隧道技术在多播中的应用

(3) 基于核心的发现技术

- 这种方法对于多播组的大小在较大范围内变化时都适合。
- 这种方法是对每一个多播组 G 指定一个核心 (core) 路由器，给出它的 IP 单播地址。
- 核心路由器按照前面讲过的方法创建出对应于多播组 G 的转发树。

几种多播路由选择协议

- 距离向量多播路由选择协议 DVMRP (Distance Vector Multicast Routing Protocol)
- 基于核心的转发树 CBT (Core Based Tree)
- 开放最短通路优先的多播扩展 MOSPF (Multicast Extensions to OSPF)
- 协议无关多播-稀疏方式 PIM-SM (Protocol Independent Multicast-Sparse Mode)
- 协议无关多播-密集方式 PIM-DM (Protocol Independent Multicast-Dense Mode)

4.8 虚拟专用网 VPN和网络地址转换 NAT

- 4.8.1 虚拟专用网 VPN
- 4.8.2 网络地址转换 NAT

4.8.1 虚拟专用网 VPN

- 由于 IP 地址的紧缺，一个机构能够申请到的IP地址数往往远小于本机构所拥有的主机数。
- 考虑到互联网并不很安全，一个机构内也并不需要把所有的主机接入到外部的互联网。
- 假定在一个机构内部的计算机通信也是采用 TCP/IP 协议，那么从原则上讲，对于这些仅在机构内部使用的计算机就可以由本机构自行分配其 IP 地址。

本地地址与全球地址

- 本地地址——仅在机构内部使用的 IP 地址，可以由本机构自行分配，而不需要向互联网的管理机构申请。
- 全球地址——全球唯一的 IP 地址，必须向互联网的管理机构申请。
- 问题：在内部使用的本地地址就有可能和互联网中某个 IP 地址重合，这样就会出现地址的二义性问题。

本地地址与全球地址

- 问题：在内部使用的本地地址就有可能和互联网中某个 IP 地址重合，这样就会出现地址的二义性问题。
- 解决：RFC 1918指明了一些专用地址 (private address)。专用地址只能用作本地地址而不能用作全球地址。在互联网中的所有路由器，对目的地址是专用地址的数据报一律不进行转发。

RFC 1918 指明的专用 IP 地址

三个专用 IP 地址块：

(1) 10.0.0.0 到 10.255.255.255

A类，或记为10.0.0.0/8，它又称为24位块

(2) 172.16.0.0 到 172.31.255.255

B类，或记为172.16.0.0/12，它又称为20位块

(3) 192.168.0.0 到 192.168.255.255

C类，或记为192.168.0.0/16，它又称为16位块

专用网

- 采用这样的专用 IP 地址的互连网络称为专用互联网或本地互联网，或更简单些，就叫作专用网。
- 因为这些专用地址仅在本机构内部使用。专用IP地址也叫作可重用地址(reusable address)。

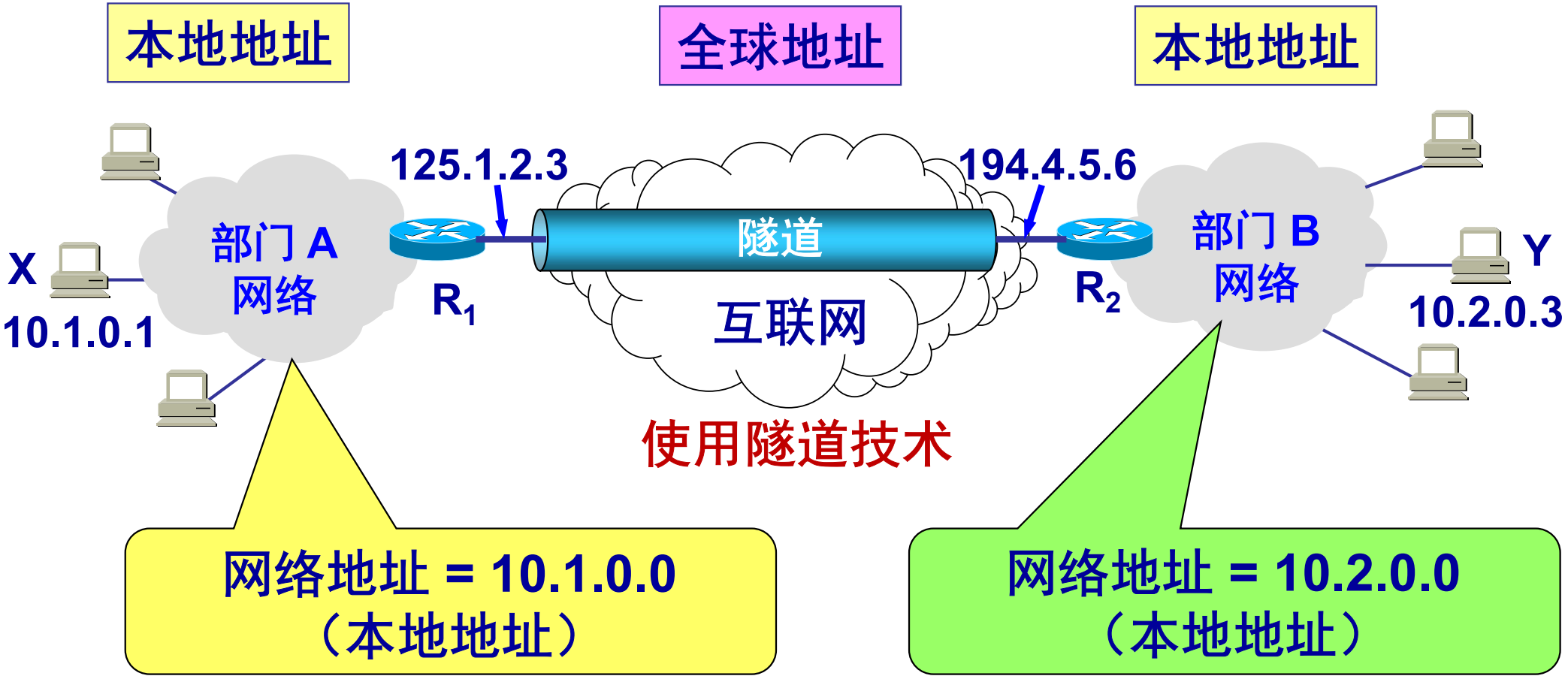
虚拟专用网 VPN

- 利用公用的互联网作为本机构各专用网之间的通信载体，这样的专用网又称为虚拟专用网 VPN (Virtual Private Network)。
- “专用网”是因为这种网络是为本机构的主机用于机构内部的通信，而不是用于和网络外非本机构的主机通信。
- “虚拟”表示“好像是”，但实际上并不是，因为现在并没有真正使用通信专线，而VPN只是在效果上和真正的专用网一样。

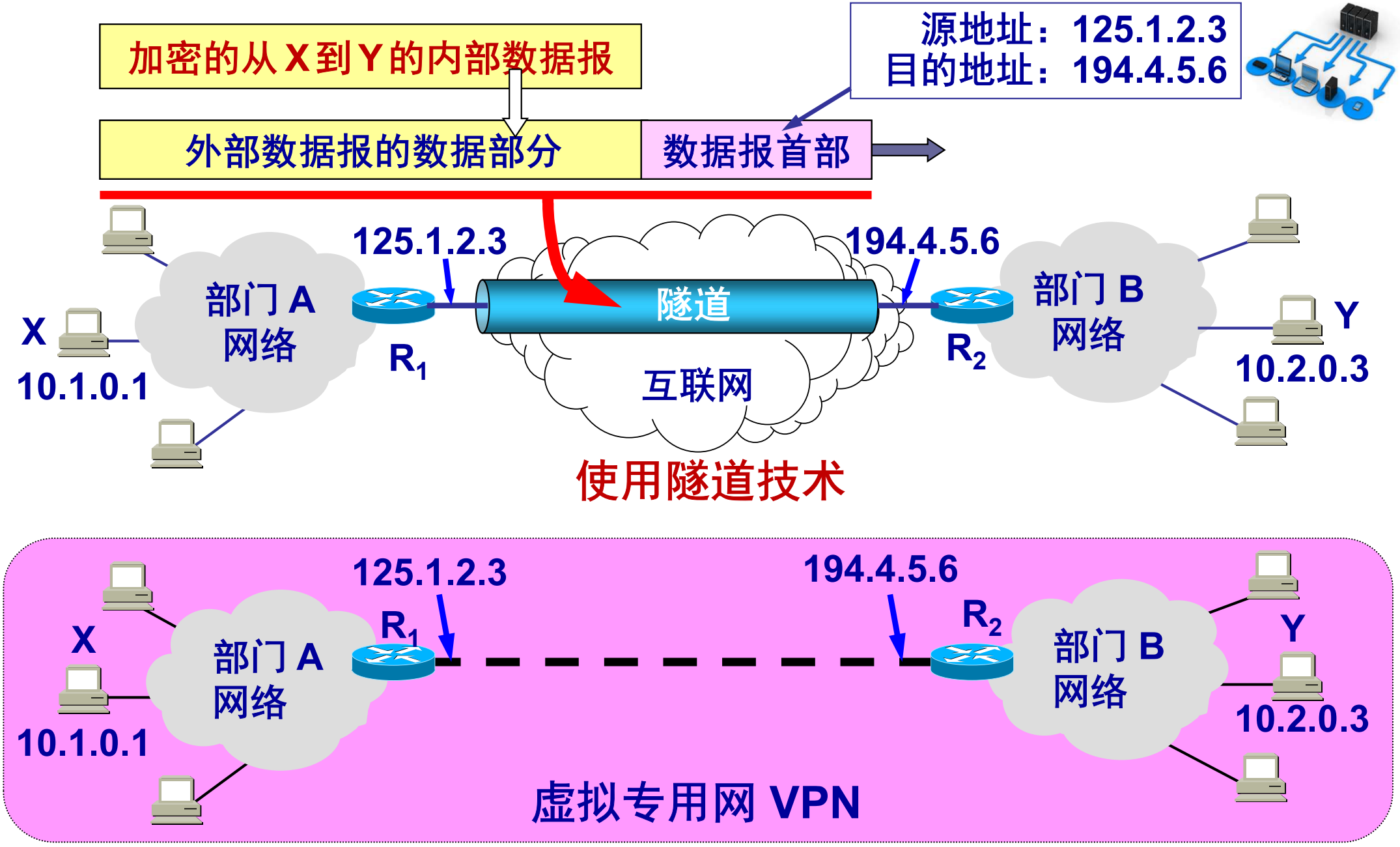
虚拟专用网 VPN 构建

- 如果专用网不同网点之间的通信必须经过公用的互联网，但又有保密的要求，那么所有通过互联网传送的数据都必须加密。
- 一个机构要构建自己的 VPN 就必须为它的每一个场所购买专门的硬件和软件，并进行配置，使每一个场所的 VPN 系统都知道其他场所的地址。

用隧道技术实现虚拟专用网



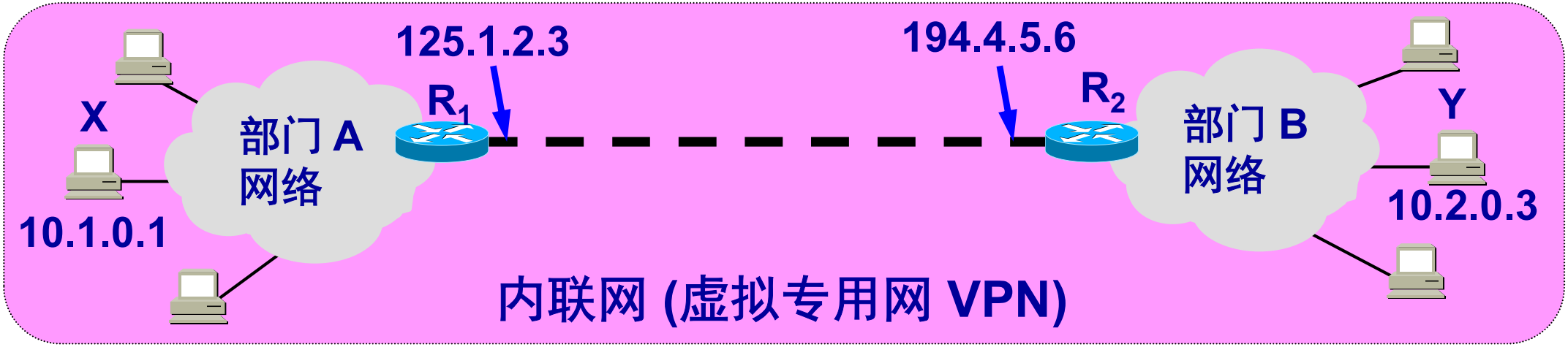
隧道技术



用隧道技术实现虚拟专用网

内联网 intranet 和外联网 extranet

- 它们都是基于 TCP/IP 协议的。
- 由部门 A 和 B 的内部网络所构成的虚拟专用网 VPN 又称为内联网 (intranet)，表示部门 A 和 B 都是在同一个机构的内部。
- 一个机构和某些外部机构共同建立的虚拟专用网 VPN 又称为外联网 (extranet)。



远程接入 VPN

- 远程接入 VPN (remote access VPN)可以满足外部流动员工访问公司网络的需求。
- 在外地工作的员工拨号接入互联网，而驻留在员工 PC 机中的 VPN 软件可在员工的 PC 机和公司的主机之间建立 VPN 隧道，因而外地员工与公司通信的内容是保密的，员工们感到好像就是使用公司内部的本地网络。

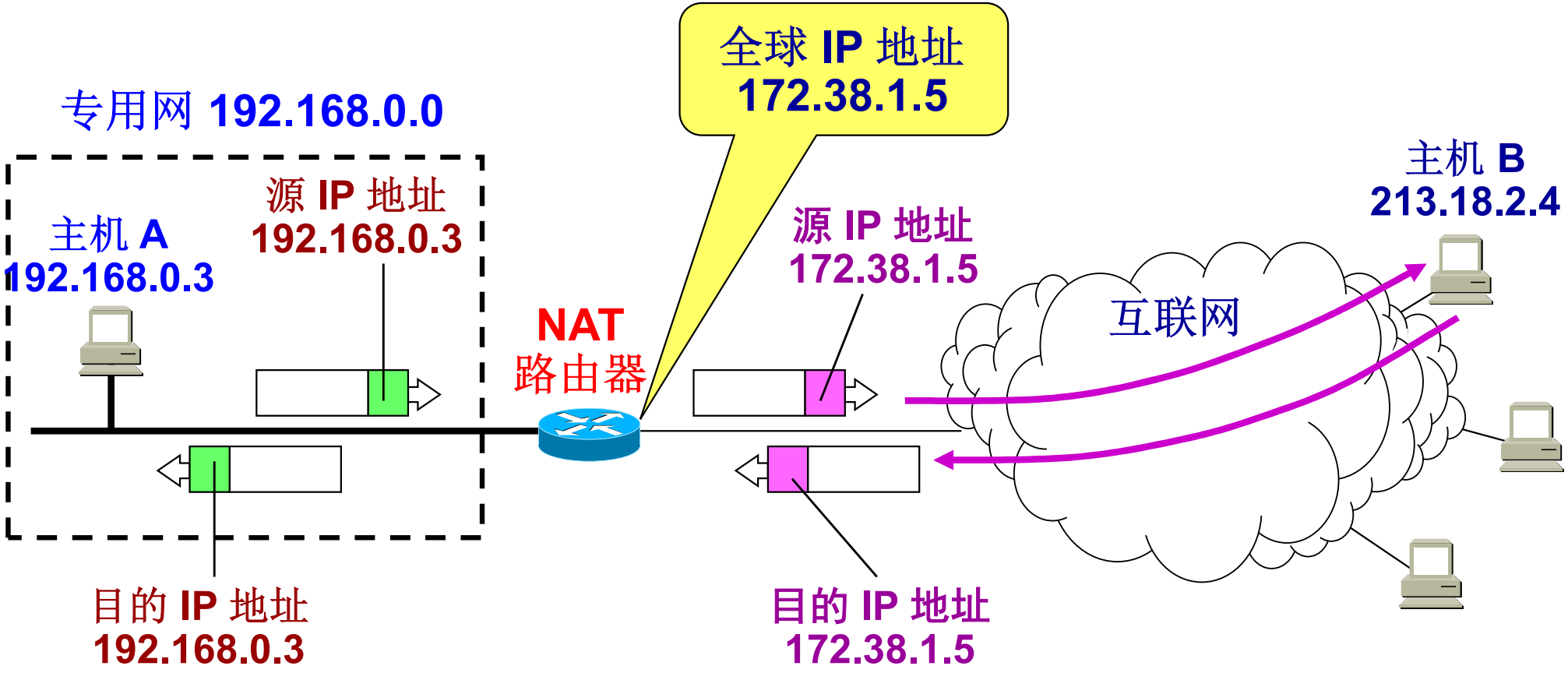
4.8.2 网络地址转换 NAT

- 问题：在专用网上使用专用地址的主机如何与互联网上的主机通信（并不需要加密）？
- 解决：
  - (1) 再申请一些全球 IP 地址。但这在很多情况下是不容易做到的。
  - (2)采用网络地址转换 NAT。这是目前使用得最多的方法。

网络地址转换 NAT

- 网络地址转换 NAT (Network Address Translation) 方法于1994年提出。
- 需要在专用网连接到互联网的路由器上安装 NAT 软件。装有 NAT 软件的路由器叫作 NAT 路由器，它至少有一个有效的外部全球IP地址。
- 所有使用本地地址的主机在和外界通信时，都要在 NAT 路由器上将其本地地址转换成全球 IP 地址，才能和互联网连接。

网络地址转换的过程



NAT 路由器的工作原理

网络地址转换的过程

- 内部主机 A 用本地地址 IP_A 和互联网上主机 B 通信所发送的数据报必须经过 NAT 路由器。
- NAT 路由器将数据报的源地址 IP_A 转换成全球地址 IP_G，并把转换结果记录到NAT地址转换表中，目的地址 IP_B 保持不变，然后发送到互联网。
- NAT 路由器收到主机 B 发回的数据报时，知道数据报中的源地址是 IP_B 而目的地址是 IP_G。
- 根据 NAT 转换表，NAT 路由器将目的地址 IP_G 转换为 IP_A，转发给最终的内部主机 A。

网络地址转换的过程

- 可以看出，在内部主机与外部主机通信时，在NAT路由器上发生了两次地址转换：
  - 离开专用网时：替换源地址，将内部地址替换为全球地址；
  - 进入专用网时：替换目的地址，将全球地址替换为内部地址；

NAT地址转换表举例

方向	字段	旧的IP地址	新的IP地址
出	源IP地址	192.168.0.3	172.38.1.5
入	目的IP地址	172.38.1.5	192.168.0.3
出	源IP地址	192.168.0.7	172.38.1.6
入	目的IP地址	172.38.1.6	192.168.0.7

网络地址转换 NAT

- 当 NAT 路由器具有 *n* 个全球 IP 地址时，专用网内最多可以同时有 *n* 台主机接入到互联网。这样就可以使专用网内较多数量的主机，轮流使用 NAT 路由器有限数量的全球 IP 地址。
- 通过 NAT 路由器的通信必须由专用网内的主机发起。专用网内部的主机不能充当服务器用，因为互联网上的客户无法请求专用网内的服务器提供服务。

网络地址与端口号转换 NAPT

- 为了更加有效地利用 NAT 路由器上的全球IP地址，现在常用的 NAT 转换表把运输层的端口号也利用上。这样，就可以使多个拥有本地地址的主机，共用一个 NAT 路由器上的全球 IP 地址，因而可以同时和互联网上的不同主机进行通信。
- 使用端口号的 NAT 叫作网络地址与端口号转换 NAPT (Network Address and Port Translation)，而不使用端口号的 NAT 就叫作传统的 NAT (traditional NAT)。

NAPT 地址转换表

NAPT 地址转换表举例

方向	字段	旧的IP地址和端口号	新的IP地址和端口号
出	源IP地址:TCP源端口	192.168.0.3:30000	172.38.1.5:40001
出	源IP地址:TCP源端口	192.168.0.4:30000	172.38.1.5:40002
入	目的IP地址:TCP目的端口	172.38.1.5:40001	192.168.0.3:30000
入	目的IP地址:TCP目的端口	172.38.1.5:40002	192.168.0.4:30000

NAPT把专用网内不同的源 IP 地址，都转换为同样的全球 IP 地址。但对源主机所采用的 TCP 端口号（不管相同或不同），则转换为不同的新的端口号。因此，当 NAPT 路由器收到从互联网发来的应答时，就可以从 IP 数据报的数据部分找出运输层的端口号，然后根据不同的目的端口号，从 NAPT 转换表中找到正确的目的主机。

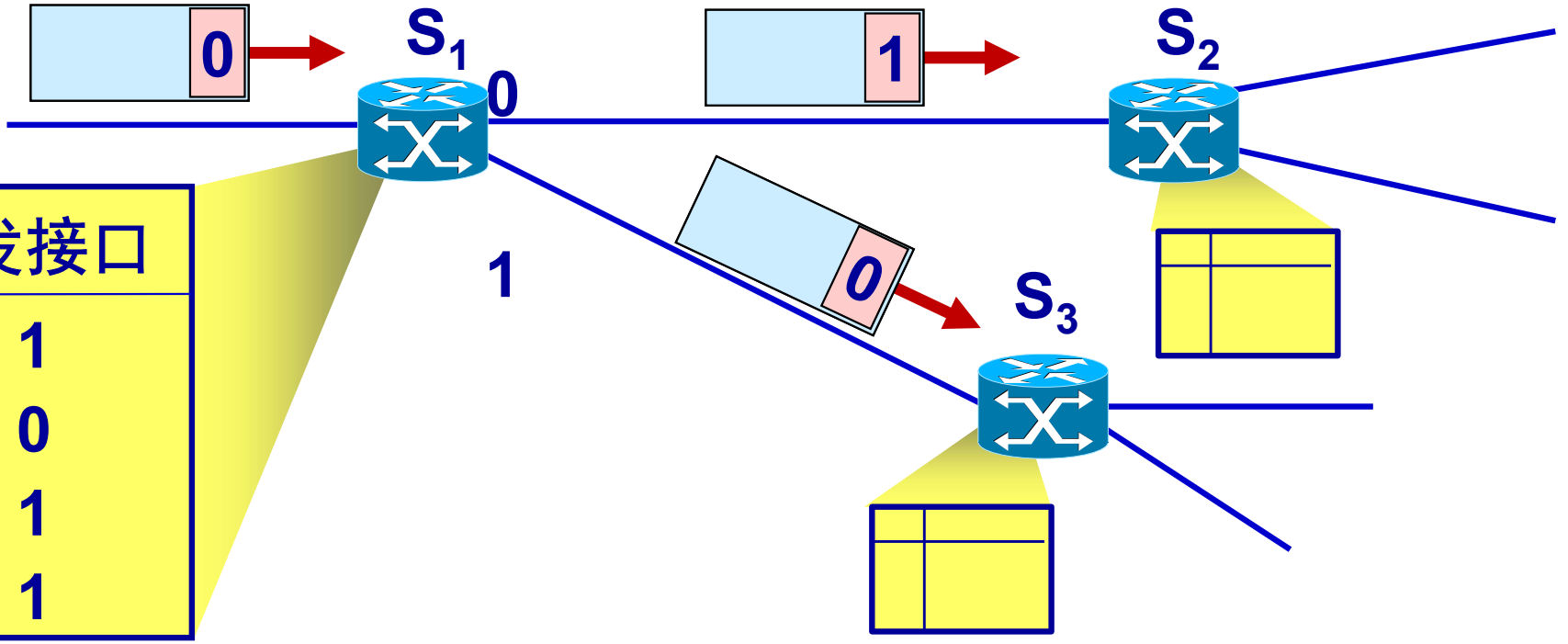
4.9 多协议标记交换 MPLS

- 4.9.1 MPLS 的工作原理
- 4.9.2 MPLS 首部的位置与格式

4.9 多协议标记交换 MPLS

- IETF于1997年成立了 MPLS 工作组，开发出一种新的协议——多协议标记交换 MPLS (MultiProtocol Label Switching)。
- “多协议”表示在 MPLS 的上层可以采用多种协议，例如：IP，IPX；可以使用多种数据链路层协议，例如：PPP，以太网，ATM 等。
- “标记”是指每个分组被打上一个标记，根据该标记对分组进行转发。

为了实现交换，可以利用面向连接的概念，
使每个分组携带一个叫作标记 (label) 的小整数。
当分组到达交换机（即标记交换路由器）时，
交换机读取分组的标记，
并用标记值来检索分组转发表。
这样就比查找路由表来转发分组要快得多。



标号	转发接口
0	1
1	0
2	1
3	1

S_1 的转发表

MPLS 特点

- MPLS并没有取代 IP，而是作为一种 IP 增强技术，被广泛地应用在互联网中。
- MPLS 具有以下三个方面的特点：
  - (1) 支持面向连接的服务质量；
  - (2) 支持流量工程，平衡网络负载；
  - (3) 有效地支持虚拟专用网 VPN。

4.9.1 MPLS 的工作原理

1. 基本工作过程

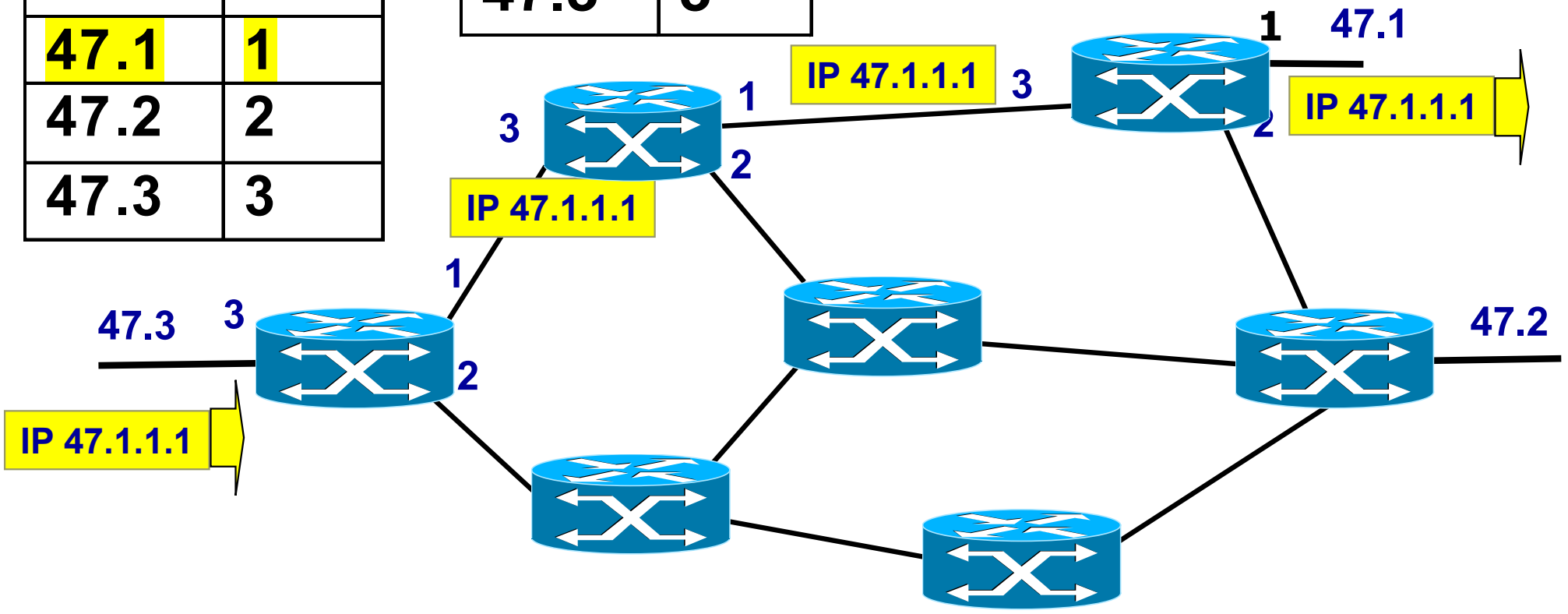
- **IP 分组的转发**
  - 在传统的 IP 网络中，分组每到达一个路由器后，都必须提取出其目的地址，按目的地址查找路由表，并按照“最长前缀匹配”的原则找到下一跳的 IP 地址（请注意，前缀的长度是不确定的）。
  - 当网络很大时，查找含有大量项目的路由表要花费很多的时间。
  - 在出现突发性的通信量时，往往还会使缓存溢出，这就会引起分组丢失、传输时延增大和服务质量下降。

IP 分组的转发

Dest	Out
47.1	1
47.2	2
47.3	3

Dest	Out
47.1	1
47.2	2
47.3	3

Dest	Out
47.1	1
47.2	2
47.3	3



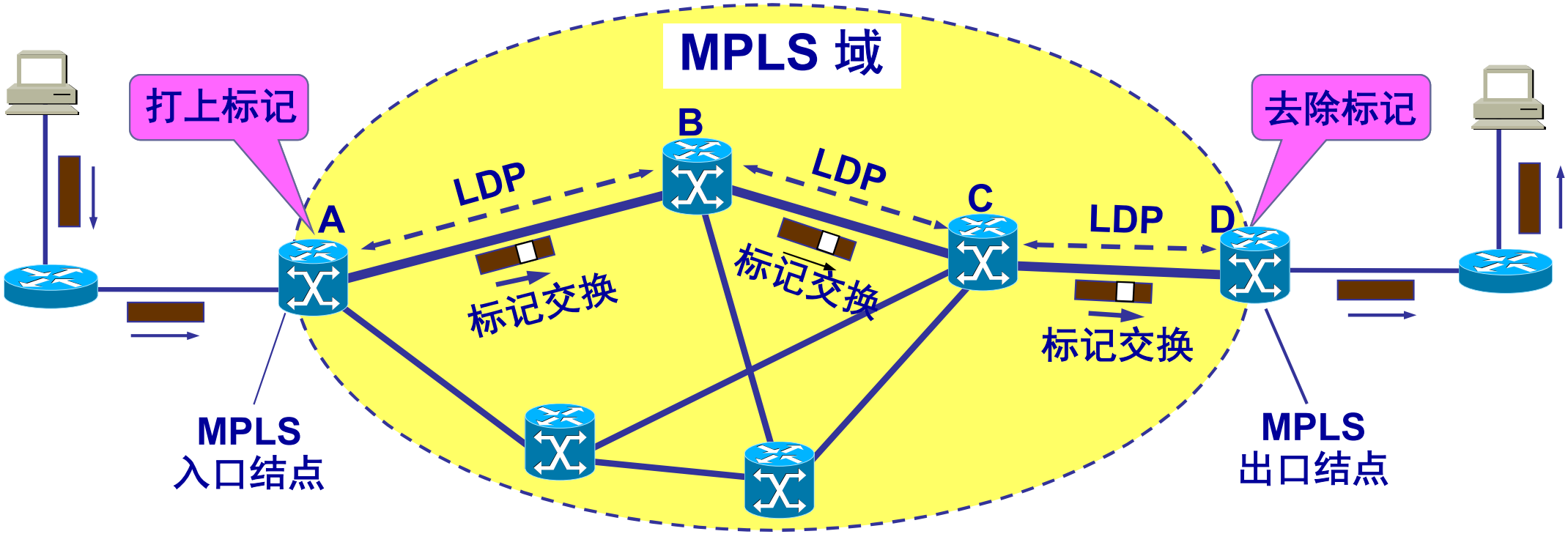
MPLS 协议的基本原理

- 在 MPLS 域的入口处，给每一个 IP 数据报打上固定长度“标记”，然后对打上标记的 IP 数据报用硬件进行转发。
- 采用硬件技术对打上标记的 IP 数据报进行转发就称为标记交换。
- “交换”也表示在转发时不再上升到第三层查找转发表，而是根据标记在第二层（链路层）用硬件进行转发。

MPLS 协议的基本原理

- **MPLS 域 (MPLS domain) 是指该域中有许多彼此相邻的路由器，并且所有的路由器都是支持 MPLS 技术的标记交换路由器 LSR (Label Switching Router)。**
- **LSR 同时具有标记交换和路由选择这两种功能，标记交换功能是为了快速转发，但在这之前 LSR 需要使用路由选择功能构造转发表。**

MPLS 协议的基本原理



普通 IP 分组

打上标记的分组

普通路由器

标记交换路由器 LSR

MPLS 协议的基本原理

MPLS 的基本工作过程

(1) MPLS 域中的各 LSR 使用专门的标记分配协议 LDP 交换报文，并找出标记交换路径 LSP。各 LSR 根据这些路径构造出分组转发表。

(2) 分组进入到 MPLS 域时， MPLS 入口结点把分组打上标记，并按照转发表将分组转发给下一个 LSR。给IP数据报打标记的过程叫作分类(classification)。

MPLS 的基本工作过程

(3) 一个标记仅仅在两个标记交换路由器 LSR 之间才有意义。分组每经过一个 LSR，LSR 就要做两件事：一是转发，二是更换新的标记，即把入标记更换成为出标记。这就叫作标记对换 (label swapping)。

转发表

入接口	入标记	出接口	出标记
0	3	1	1

项目含义：从入接口0 收到一个入标记为 3 的IP 数据报，转发时，应当把该 IP 数据报从出接口1转发出去，同时把标记对换为1。

MPLS 的基本工作过程

(4) 当分组离开 MPLS 域时，MPLS 出口结点把分组的标记去除。再以后就按照一般分组的转发方法进行转发。

上述的这种“由入口 LSR 确定进入 MPLS 域以后的转发路径”称为显式路由选择 (explicit routing)，它和互联网中通常使用的“每一个路由器逐跳进行路由选择”有着很大的区别。

2. 转发等价类 FEC

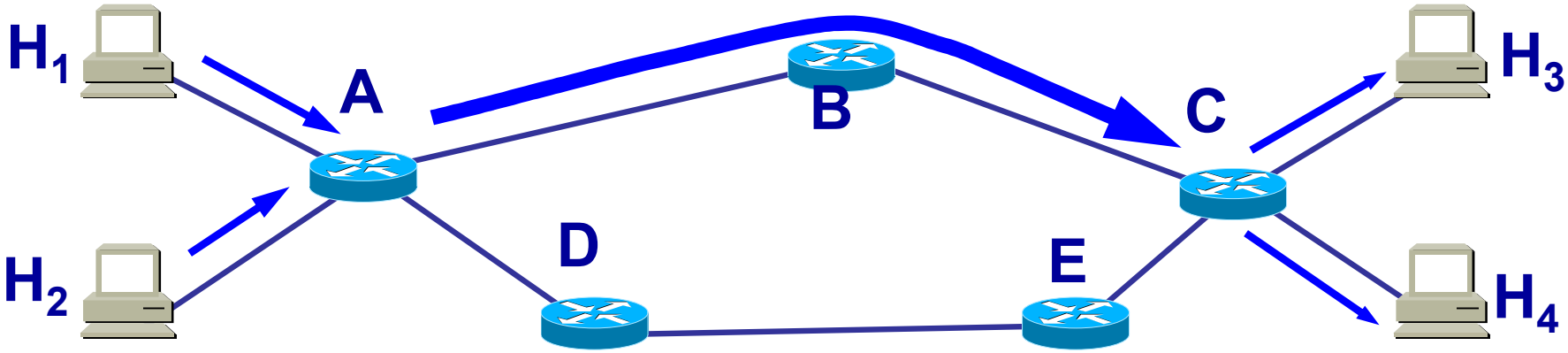
- MPLS 有个很重要的概念就是转发等价类 FEC (Forwarding Equivalence Class)。
- “转发等价类”就是路由器按照同样方式对待的分组的集合。

“按照同样方式对待”表示：从同样接口转发到同样的下一跳地址，并且具有同样服务类别和同样丢弃优先级等。

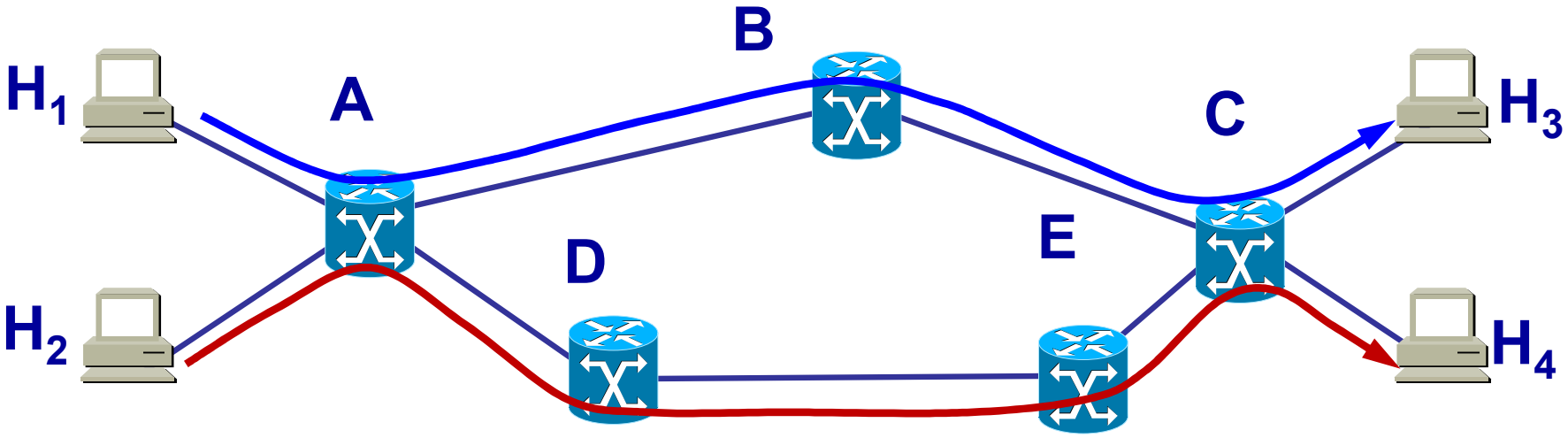
2. 转发等价类 FEC

- 划分 FEC 的方法不受什么限制，这都由网络管理员来控制，因此非常灵活。
- 入口结点并不是给每一个分组指派一个不同的标记，而是将属于同样 FEC 的分组都指派同样的标记。
- FEC 和标记是一一对应的关系。

FEC 用于负载平衡



(a) 传统路由选择协议使最短路径 A→B→C 过载



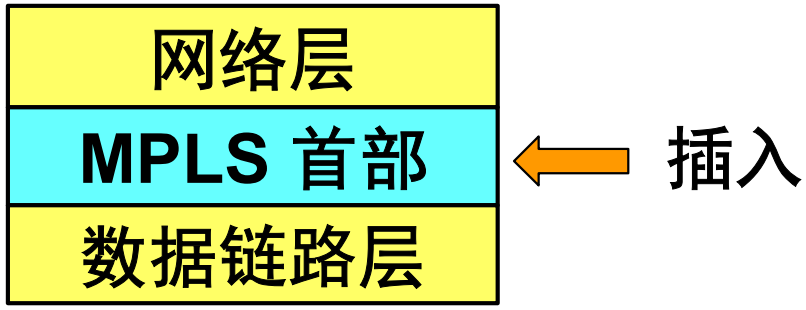
(b) 设置两种 FEC，利用 FEC 使通信量分散

流量工程

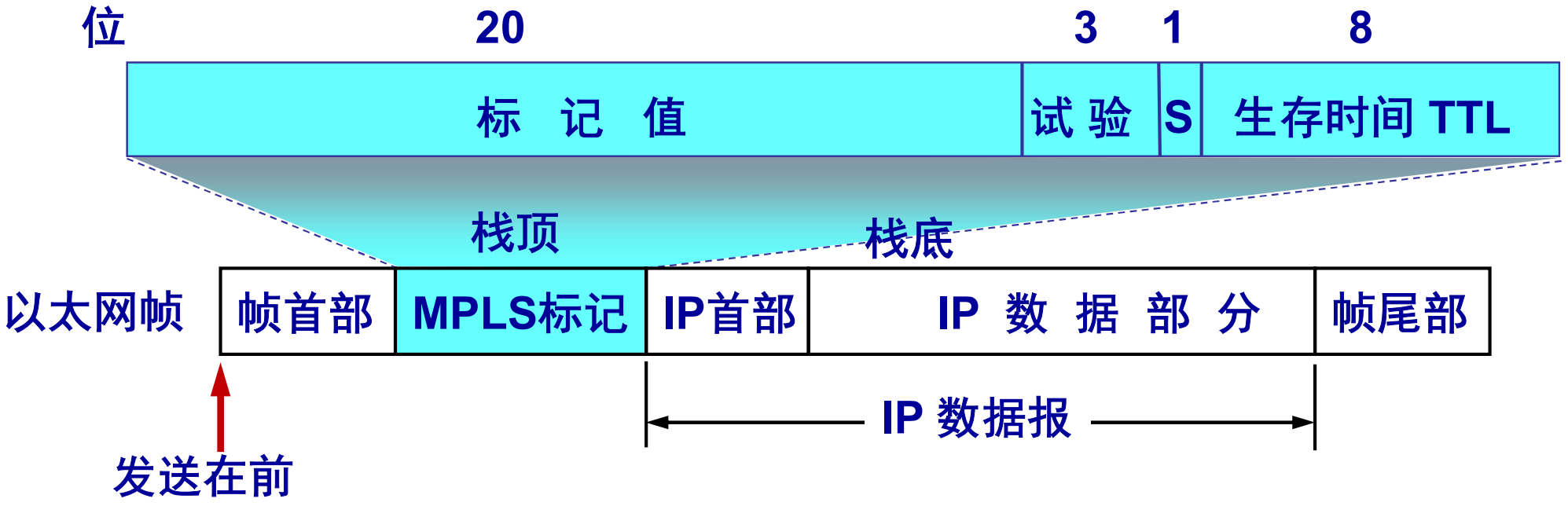
- 网络管理员采用自定义的 **FEC** 就可以更好地管理网络的资源。
- 这种均衡网络负载的做法也称为**流量工程 TE (Traffic Engineering)** 或通信量工程。

4.9.2 MPLS 首部的位置与格式

- MPLS 并不要求下层的网络都使用面向连接的技术。
- 下层的网络并不提供打标记的手段，而 IPv4 数据报首部也没有多余的位置存放 MPLS 标记。
- 这就需要使用一种封装技术：在把 IP 数据报封装成以太网帧之前，先要插入一个 MPLS 首部。
- 从层次的角度看，MPLS 首部就处在第二层和第三层之间。



MPLS 首部的格式



“给IP数据报打上标记”其实就是在以太网的帧首部和IP数据报的首部之间插入一个 4 字节的 MPLS 首部。

MPLS 首部的格式

- MPLS 首部共包括以下四个字段：
  - (1) 标记值（占 20 位）。可以同时容纳高达 2^{20} 个流（即 1048576 个流）。实际上几乎没有哪个 MPLS 实例会使用很大数目的流，因为通常需要管理员人工管理和设置每条交换路径。
  - (2) 试验（占 3 位）。目前保留用作试验。
  - (3) 栈S（占 1 位）。在有“标记栈”时使用。
  - (4) 生存时间TTL（占 8 位）。用来防止 MPLS 分组在 MPLS 域中兜圈子。