网络层

谢剑刚
广东开放大学

本章最重要的内容

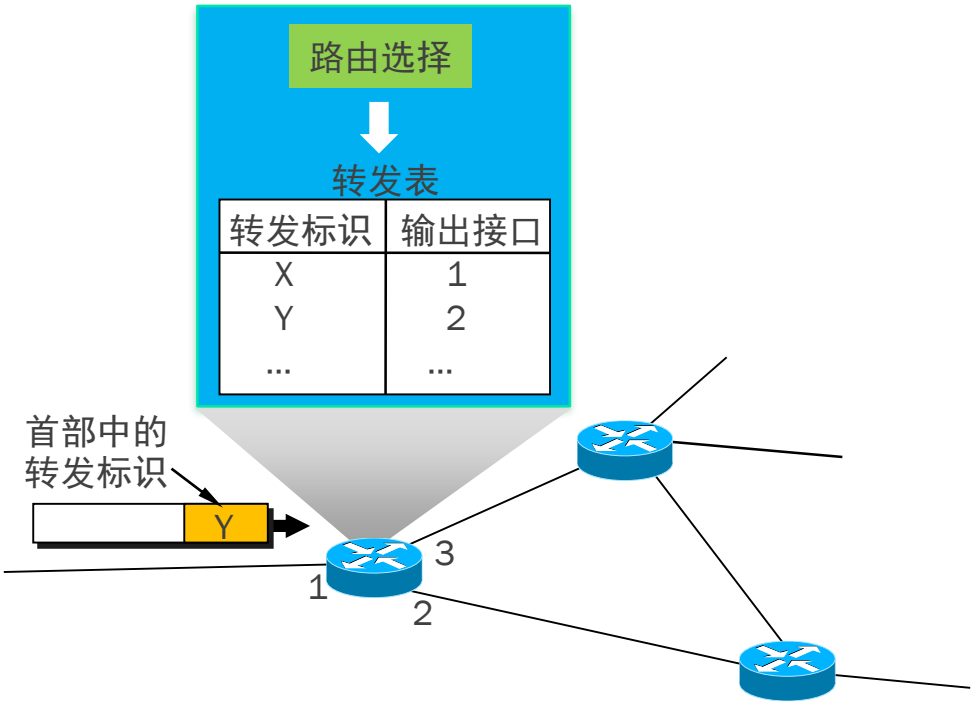
(1) 虚拟互连网络的概念

(2) IP 地址与物理地址的关系

(3) 传统的分类的 IP 地址（包括子网掩码）和无分类域间路由选择 CIDR

(4) 路由选择协议的工作原理

4.1.1 分组转发和路由选择



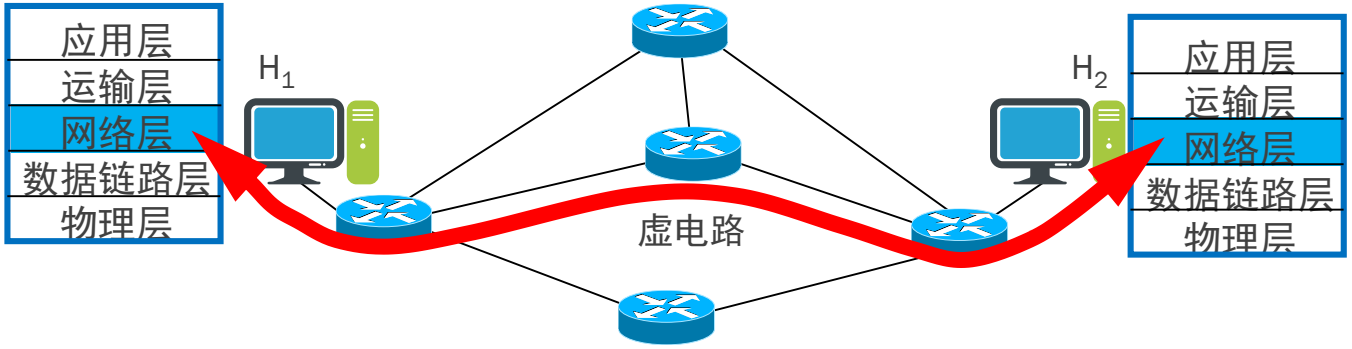
4.1.2 网络层提供的两种服务

- 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**？

电信网的成功经验让网络负责可靠交付

- **面向连接**的通信方式
- 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。

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。

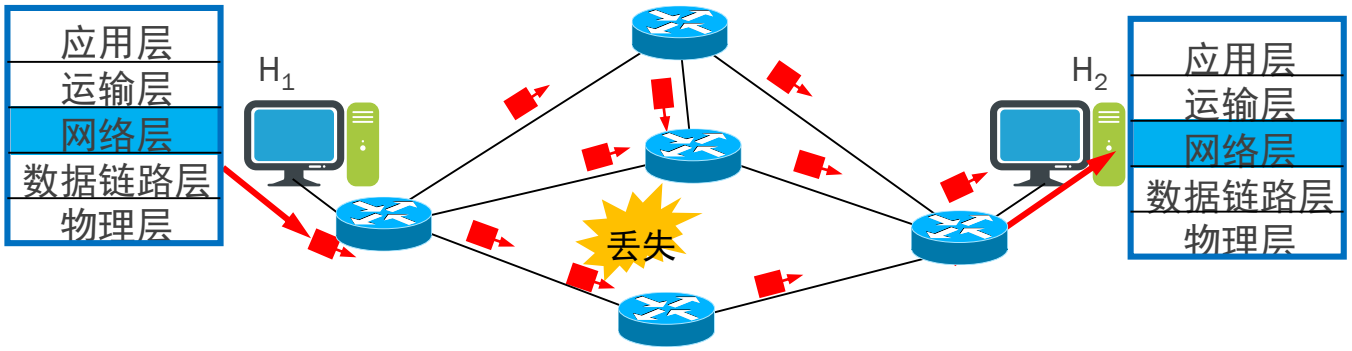
因特网采用的设计思路

- 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。

尽最大努力交付的好处

- 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（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。
- 因特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



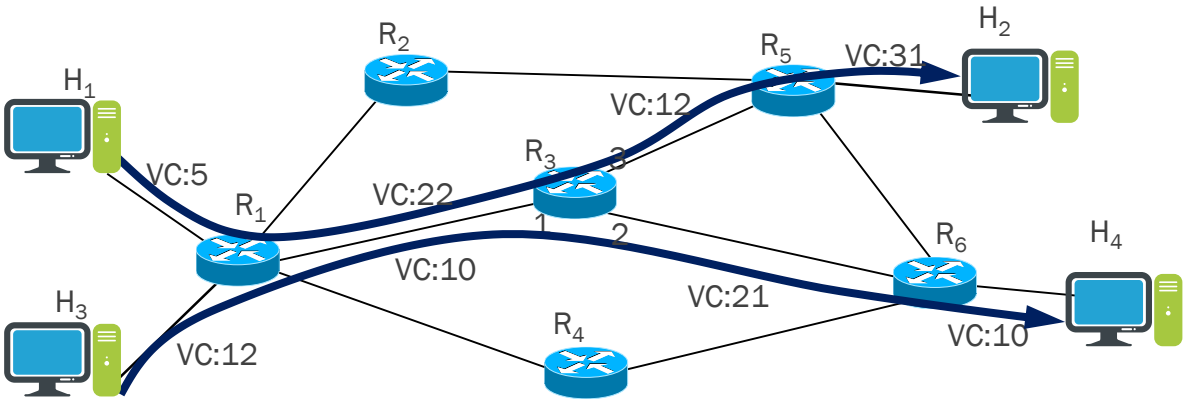
H_1 发送给 H_2 的分组可能沿着不同路径传送

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

4.1.3 虚电路网络

- 一条虚电路的组成如下：
  - 源和目的主机之间的路径（即一系列链路和路由器）；
  - VC号，沿着该路径的每段链路一个号码；
  - 沿着该路径的每台路由器（即虚电路交换机，这里我们统一使用路由器这一名称）中的转发表表项。
- 属于一条虚电路的分组将在它的首部携带一个VC号。
- 一条虚电路在每段链路上可能具有不同的VC号



虚电路转发表

- 每台中间路由器在转发分组时必须用一个新的VC号替代原来的VC号

入接口	入VC	出接口	出VC
1	22	3	12
1	10	2	21
…	…	…	…

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四个协议：**

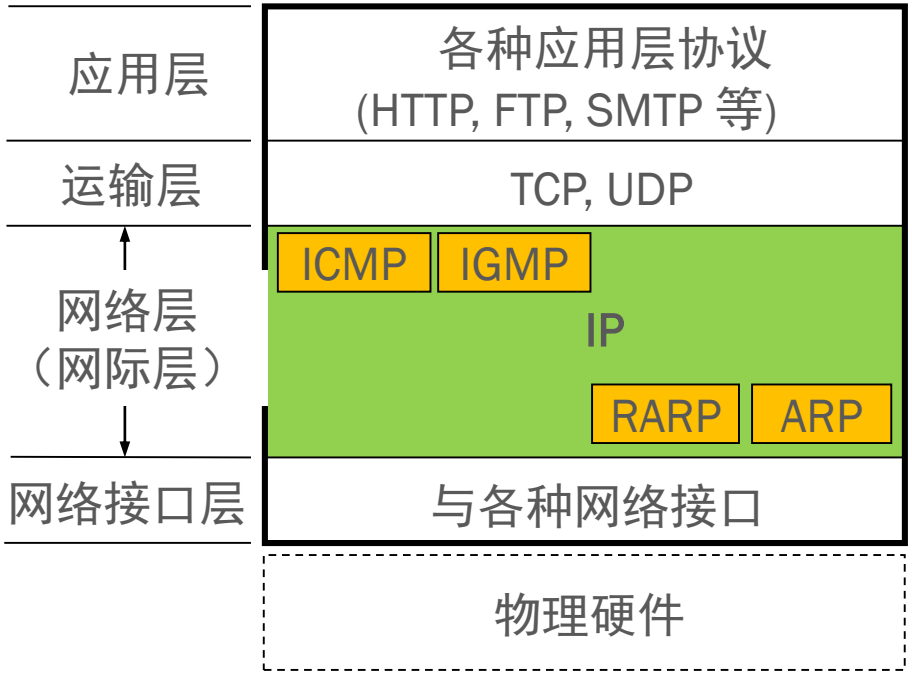
01 地址解析协议 ARP (Address Resolution Protocol)

02 逆地址解析协议 RARP (Reverse Address Resolution Protocol)

03 网际控制报文协议 ICMP (Internet Control Message Protocol)

04 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 异构网络互连

- 互连在一起的网络要进行通信，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

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。**

01 物理层中继系统：转发器(repeater)

02 数据链路层中继系统：网桥或桥接器(bridge)

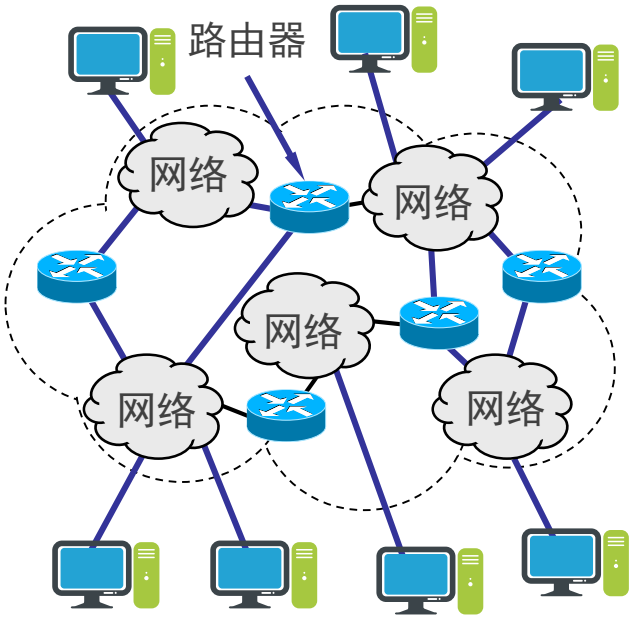
03 网络层中继系统：路由器(router)

04 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。
- 互联网都是指用路由器进行互连的网络。
- 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



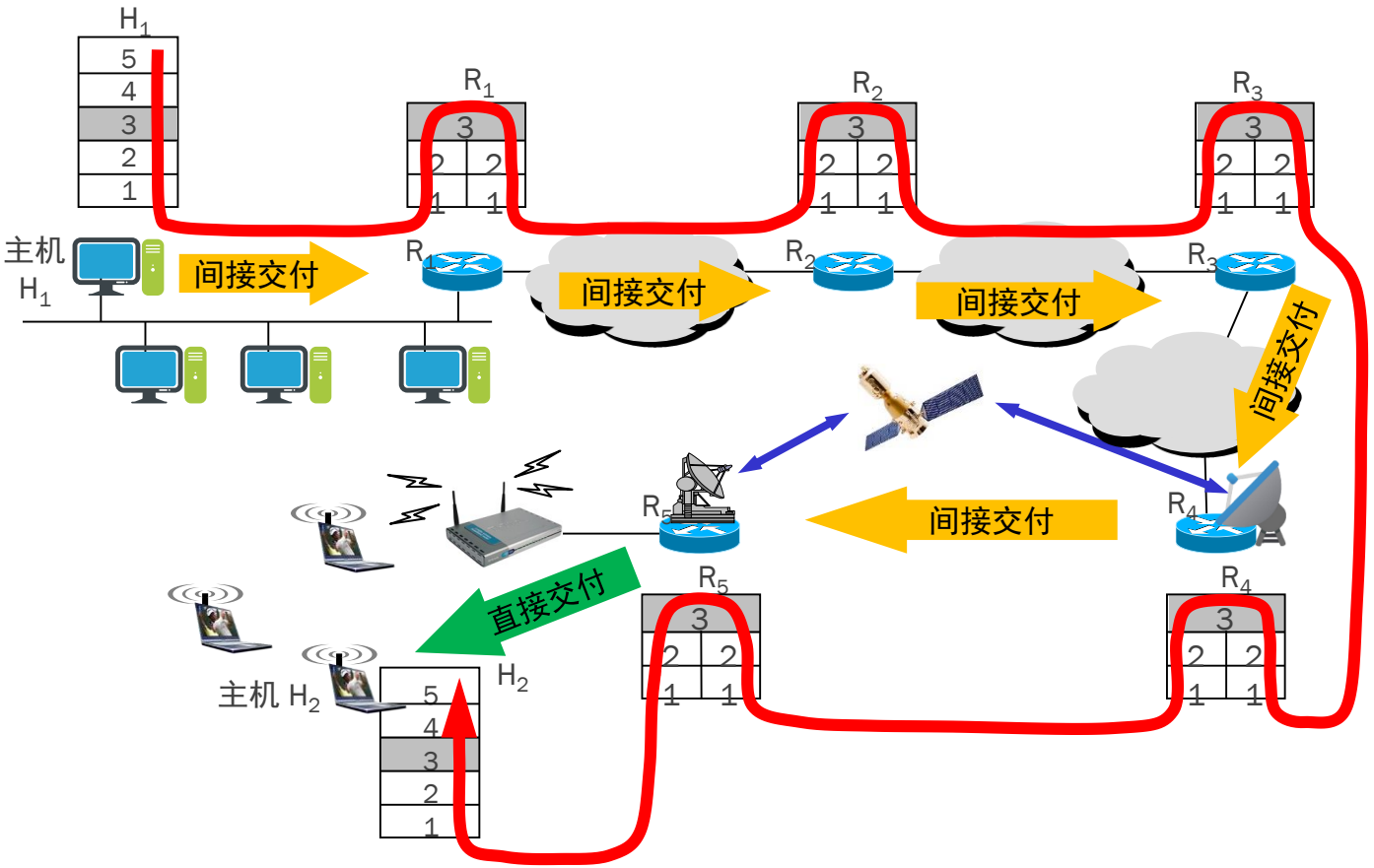
(a) 互连网络

(b) 虚拟互连网络

虚拟互连网络的意义

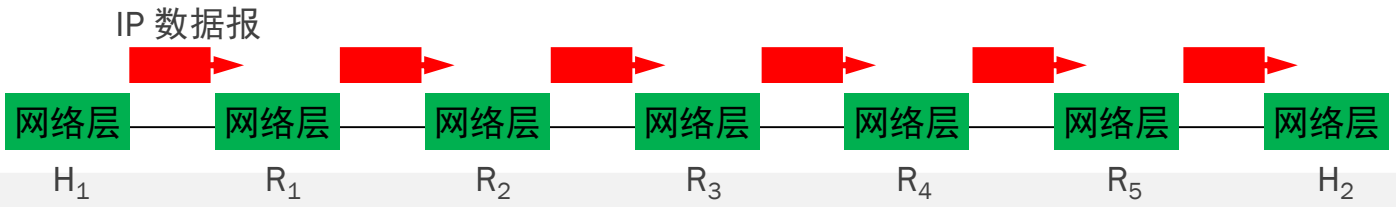
- 所谓虚拟互连网络也就是逻辑互连网络，它的意思就是互连起来的各种物理网络的异构性本来是客观存在的，但是我们利用 IP 协议就可以使这些性能各异的网络从用户看起来好像是一个统一的网络。
- 使用 IP 协议的虚拟互连网络可简称为 IP 网。
- 使用虚拟互连网络的好处是：当互联网上的主机进行通信时，就好像在一个网络上通信一样，而看不见互连的各具体的网络异构细节。

分组在互联网中的传送


H1
5
4
3
2
1
R1
R2
R3
主机
H1
间接交付
间接交付
间接交付
间接交付
R5
R4
间接交付
直接交付
主机 H2
H2

从网络层看 IP 数据报的传送

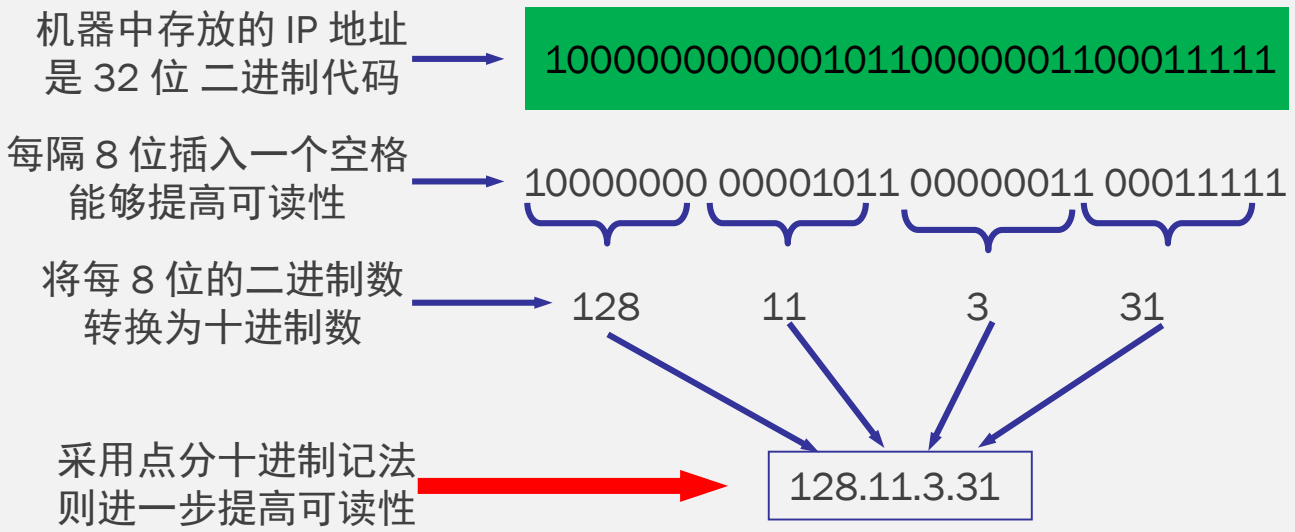
- **如果我们只从网络层考虑问题，那么 IP 数据报就可以想象是在网络层中传送。**



4.2.2 IP地址及编址方式

- **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因特网上的主机（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 地址现在由因特网名字与号码指派公司ICANN (Internet Corporation for Assigned Names and Numbers)进行分配**

点分十进制记法



IP 地址的编址方法

- **分类编址。这是最基本的编址方法，在 1981 年就通过了相应的标准协议。**
- **划分子网。这是对最基本的编址方法的改进，其标准[RFC 950]在 1985 年通过。**
- **无分类编址。这是目前因特网所使用的编址方法。1993 年提出后很快就得到推广应用。**

1. 分类 IP 地址

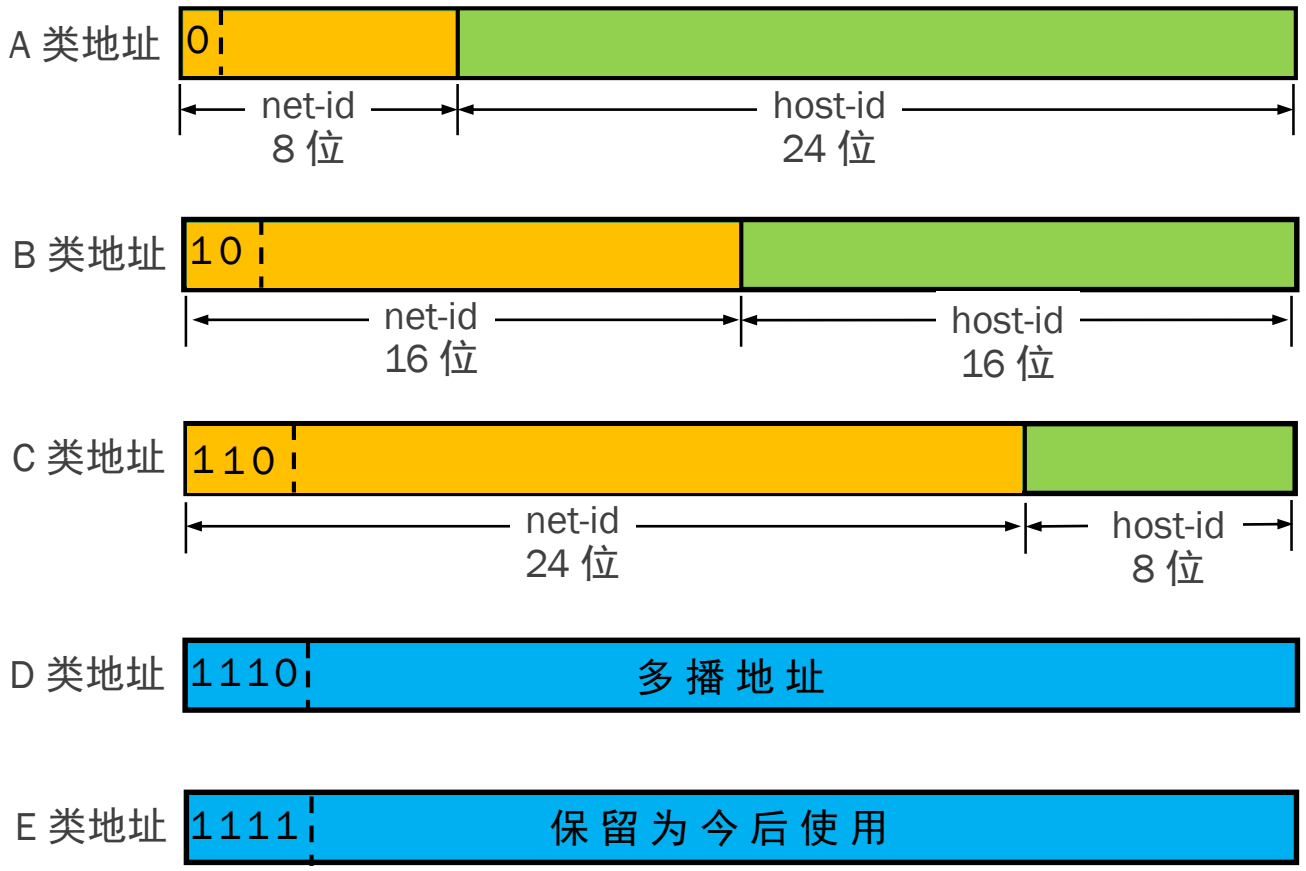
每一类地址都由两个固定长度的字段组成，其中一个字段是网络号 net-id，它标志主机（或路由器）所连接到的网络，而另一个字段则是主机号 host-id，它标志该主机（或路由器）。

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”

IP 地址中的网络号字段和主机号字段



常用的三种类别的 IP 地址

IP 地址的使用范围

网络类别	最大网络数	第一个可用的网络号	最后一个可用的网络号	每个网络中最大的主机数
A	126 (2^7 – 2)	1	126	16,777,214
B	16,383(2^{14} – 1)	128.1	191.255	65,534
C	2,097,151 (2^{21} – 1)	192.0.1	223.255.255	254

2. 划分子网

- 分类编址方式表面上看起来非常合理，但实际上并不够合理。
  - IP 地址空间的利用率有时很低。
  - 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。
  - 两级的 IP 地址不够灵活。

2. 划分子网

- **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)。划分子网已成为因特网的正式标准协议。**

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)

3. 无分类编址 CIDR

划分子网在一定程度上缓解了因特网在发展中遇到的困难。然而在 1992 年因特网仍然面临三个必须尽早解决的问题，这就是：

- 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！
- 因特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）。
- 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)

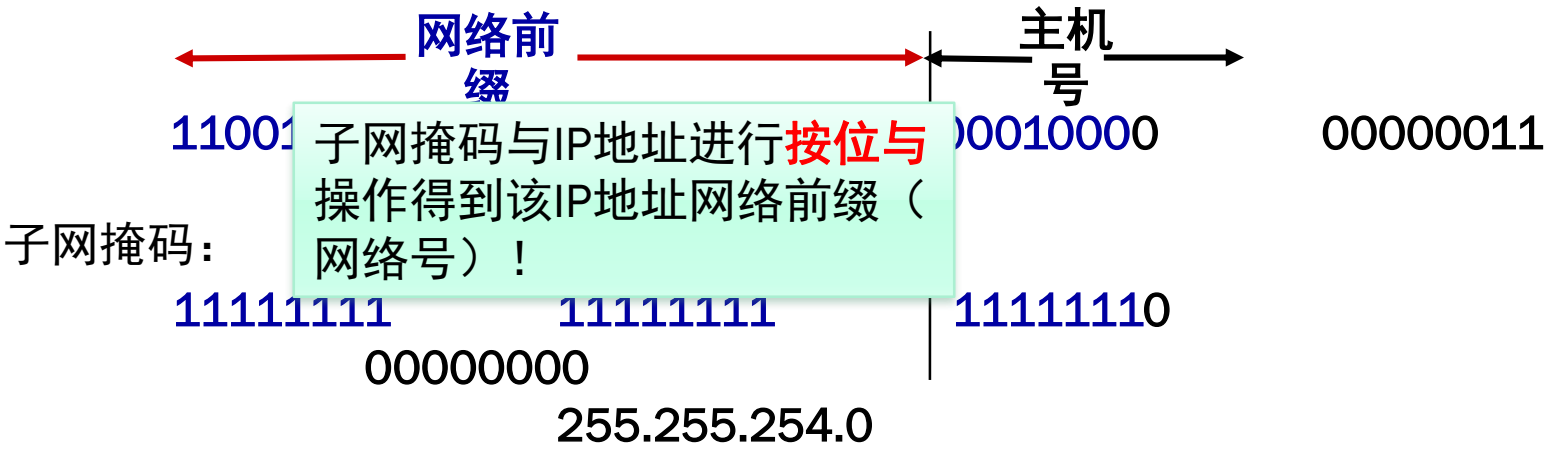
- CIDR 把网络前缀都相同的连续的 IP 地址组成“CIDR 地址块”，每块中的地址个数是2的乘方。
- 将“CIDR 地址块”分配给一个组织，该组织还可以将该地址块划分为多个更小的地址块（前缀更长）分配给组织内的小单位。
- 用不定长的网络前缀来替代原来分类IP地址中的网络号，路由器按目的地址块进行选路和转发。

子网掩码

- 在分类编址中，给定一个IP地址，就确定了它的网络号和主机号。但在无分类编址中，由于网络前缀是不定长，IP地址本身并不能确定其网络前缀和主机号。
- 使用子网掩码(subnet mask)可以找出 IP 地址中的网络部分(网络前缀)。
- CIDR 虽然不使用子网了，但仍然使用“掩码” 或“子网掩码”这一名词。

子网掩码

- 用32位的子网掩码来表示网络前缀的长度



- CIDR 还使用“斜线记法”，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。

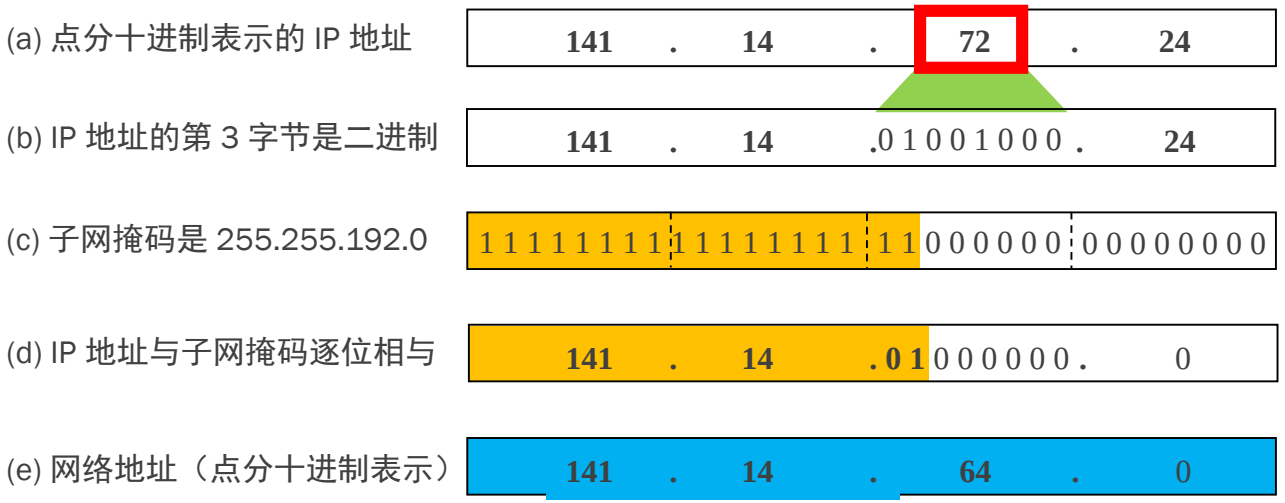
CIDR 记法的其他形式

- 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

掩码中有 10 个连续的 1　　0　　0

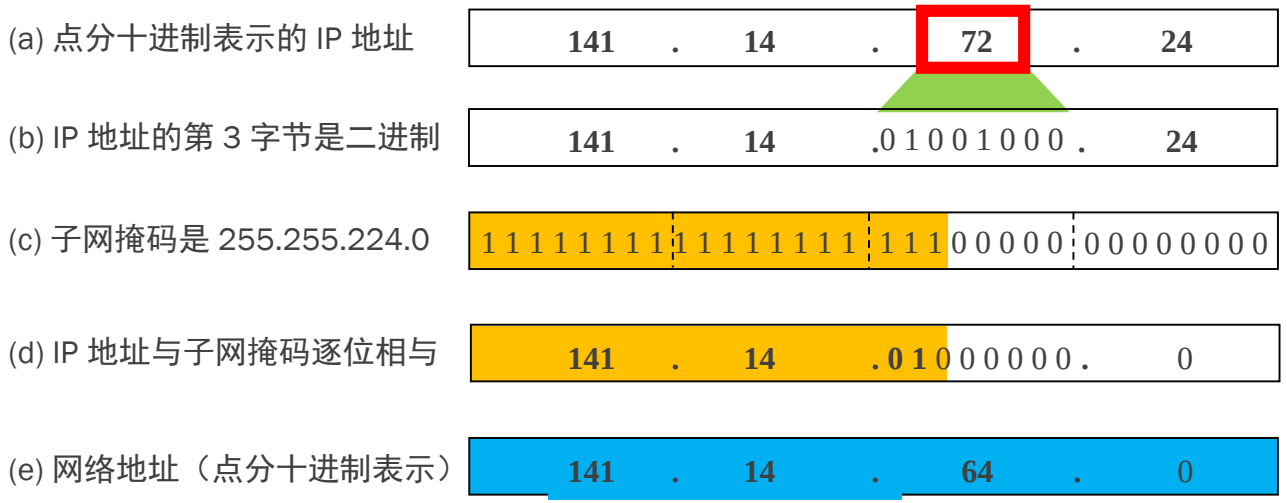
【例4-1】

- **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。**



【例4-2】

- **在上例中，若子网掩码改为255.255.224.0。试求网络地址，讨论所得结果。**

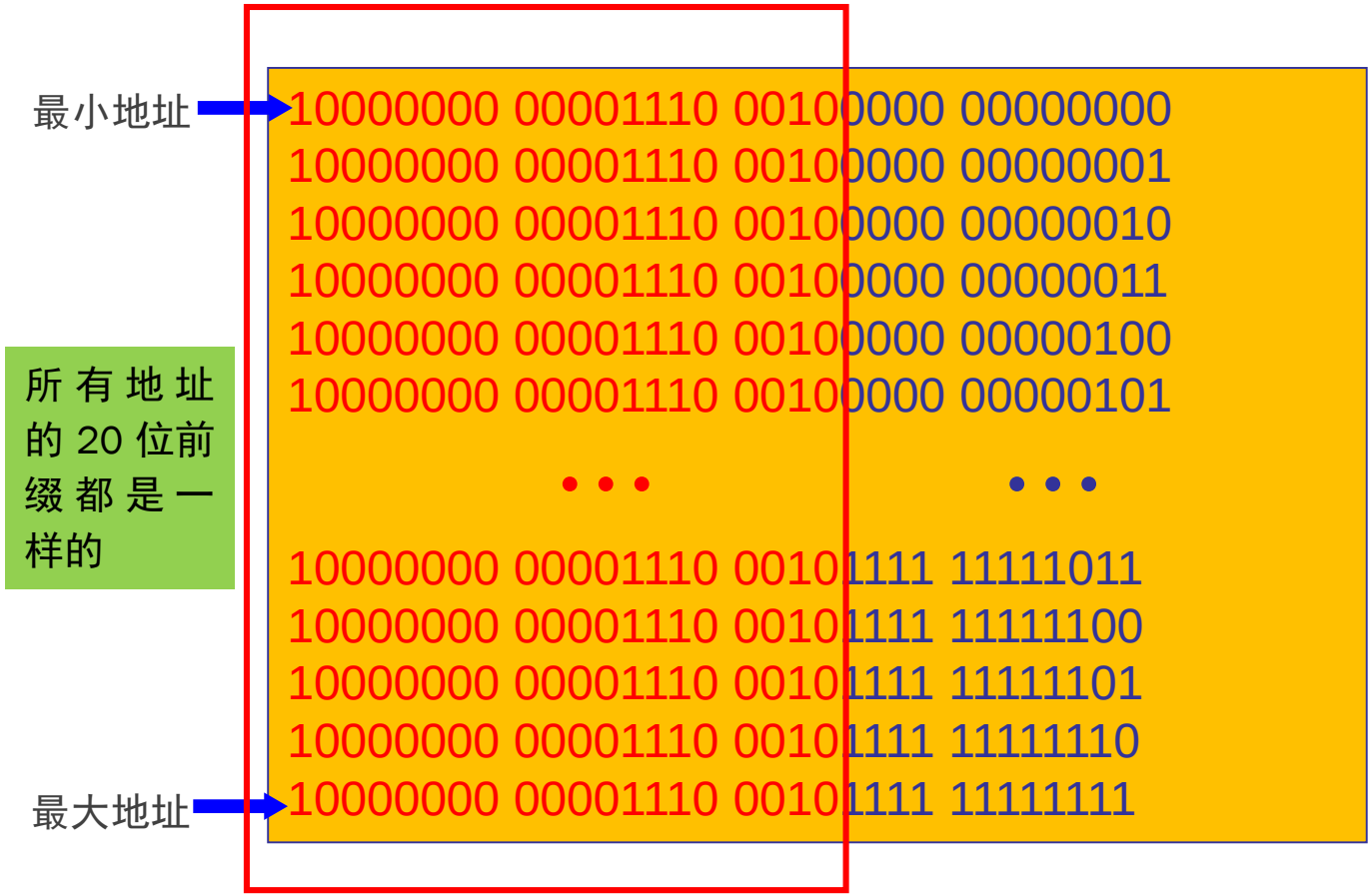


不同的子网掩码得出相同的网络地址。
但不同的掩码的效果是不同的。

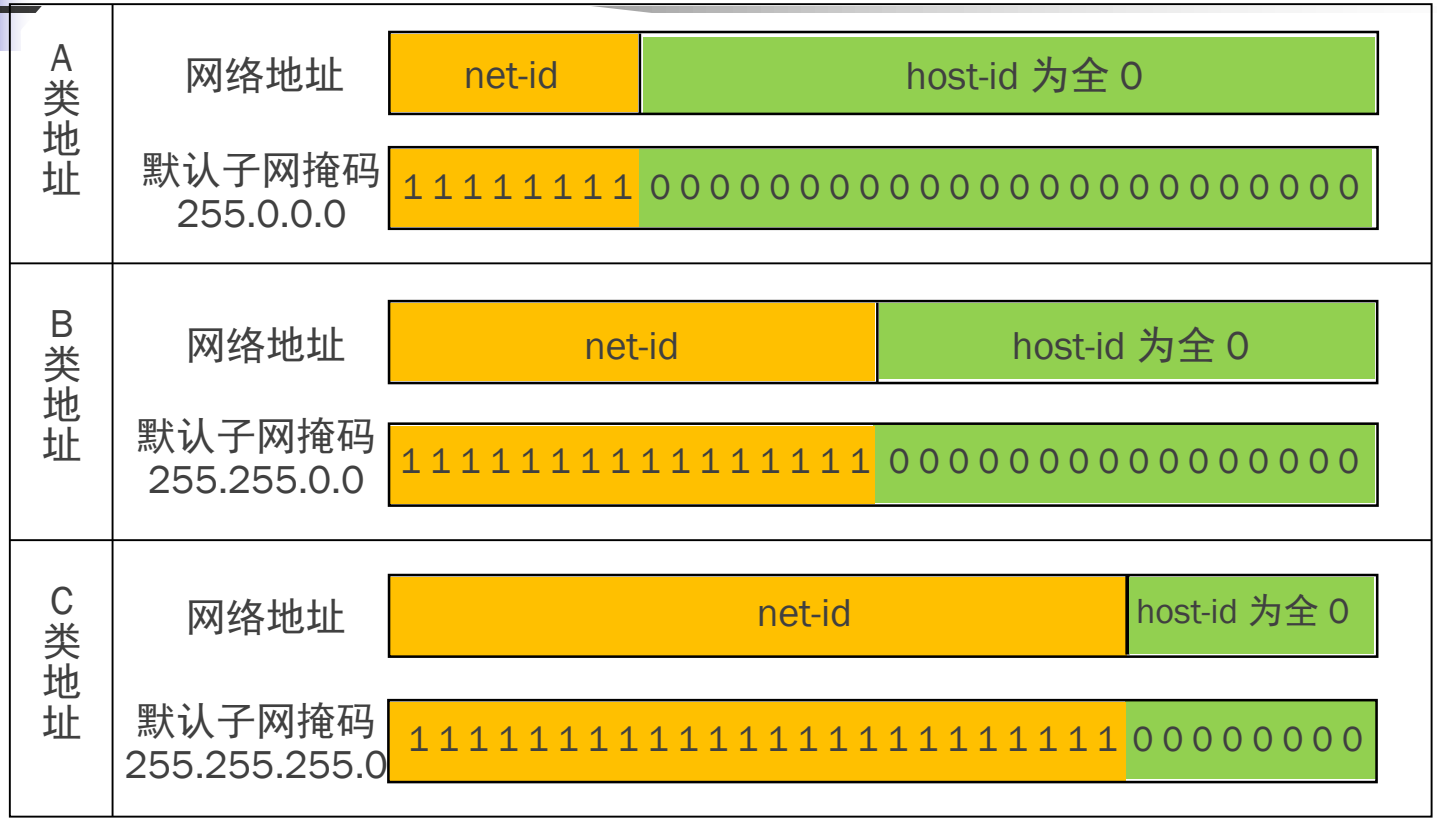
CIDR 地址块

- 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。

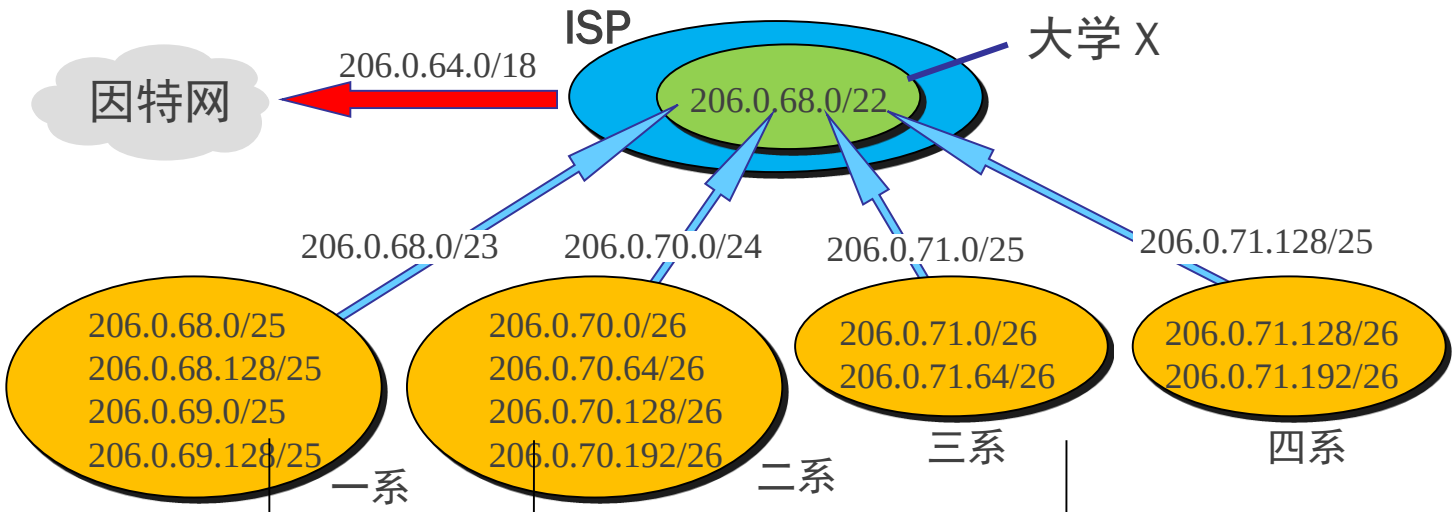
128.14.32.0/20 表示的地址（2^{12} 个地址）



默认子网掩码


A类地址
网络地址
net-id
host-id 为全 0
默认子网掩码
255.0.0.0
11111111 00000000000000000000000
B类地址
网络地址
net-id
host-id 为全 0
默认子网掩码
255.255.0.0
1111111111111111 0000000000000000
C类地址
网络地址
net-id
host-id 为全 0
默认子网掩码
255.255.255.0
11111111111111111111111 00000000

CIDR 地址块划分举例



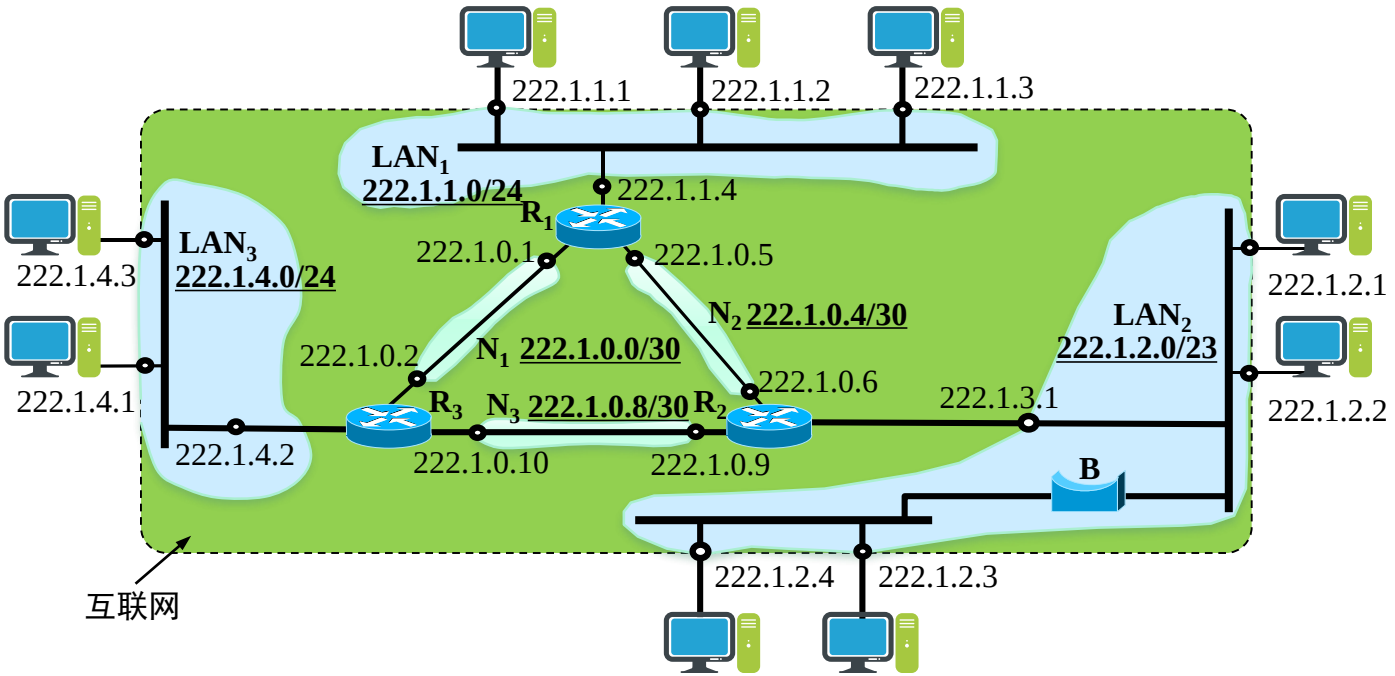
单位	地址块	二进制表示	地址数
ISP	206.0.64.0/18	11001110.00000000.01*	16384
大学	206.0.68.0/22	11001110.00000000.010001*	1024
一系	206.0.68.0/23	11001110.00000000.0100010*	512
二系	206.0.70.0/24	11001110.00000000.01000110.*	256
三系	206.0.71.0/25	11001110.00000000.01000111.0*	128
四系	206.0.71.128/25	11001110.00000000.01000111.1*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

4. 特殊的IP地址

网络前缀	主机号	源地址使用	目的地址使用	代表的意思
全0	全0	可以	不可	在本网络上的本主机（见6.7节DHCP协议）
全0	host-id	可以	不可	在本网络上的某个主机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
net-id	全0	不可	不可	网络地址，用于标识网络前缀为net-id的网络
127	非全0全1	可以	可以	用作本地软件环回测试之用

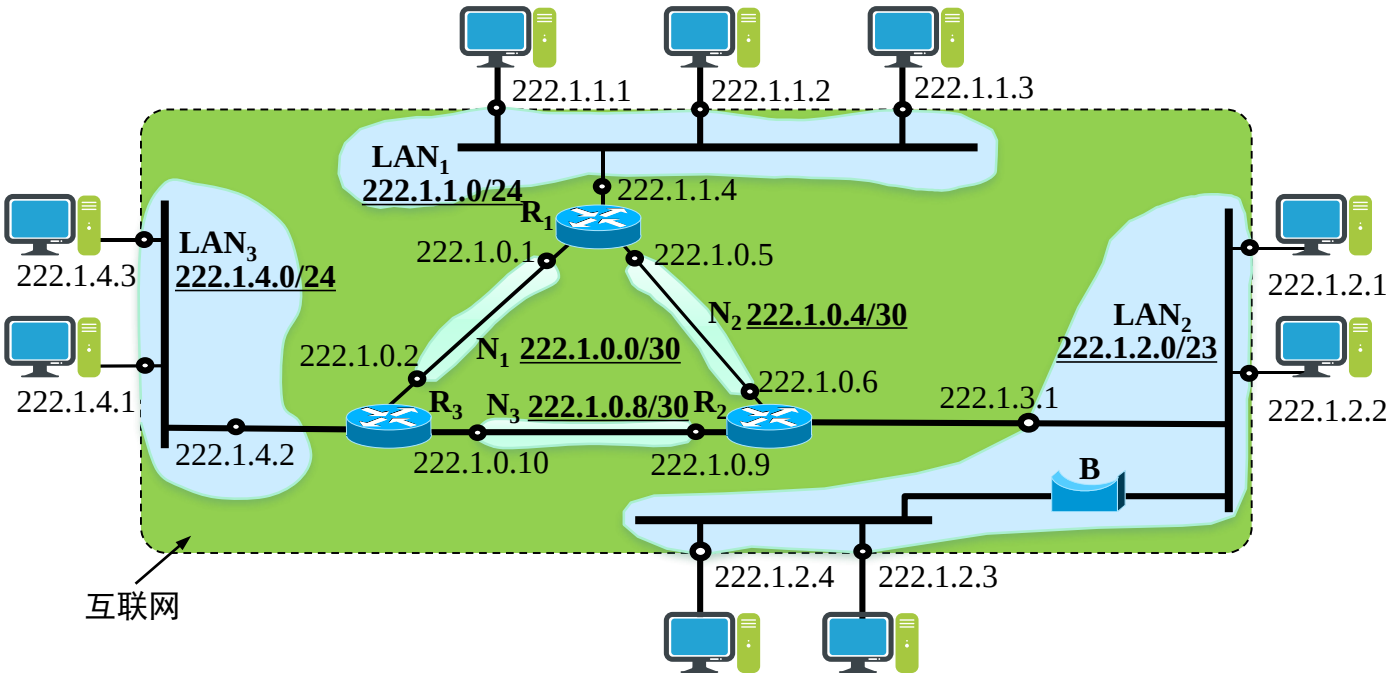
互联网中的 IP 地址

由路由器互连起来的每个网络有一个唯一的网络前缀（由IP地址和子网掩码共同确定）。主机号为全0的IP地址常表示该网络的网络地址。



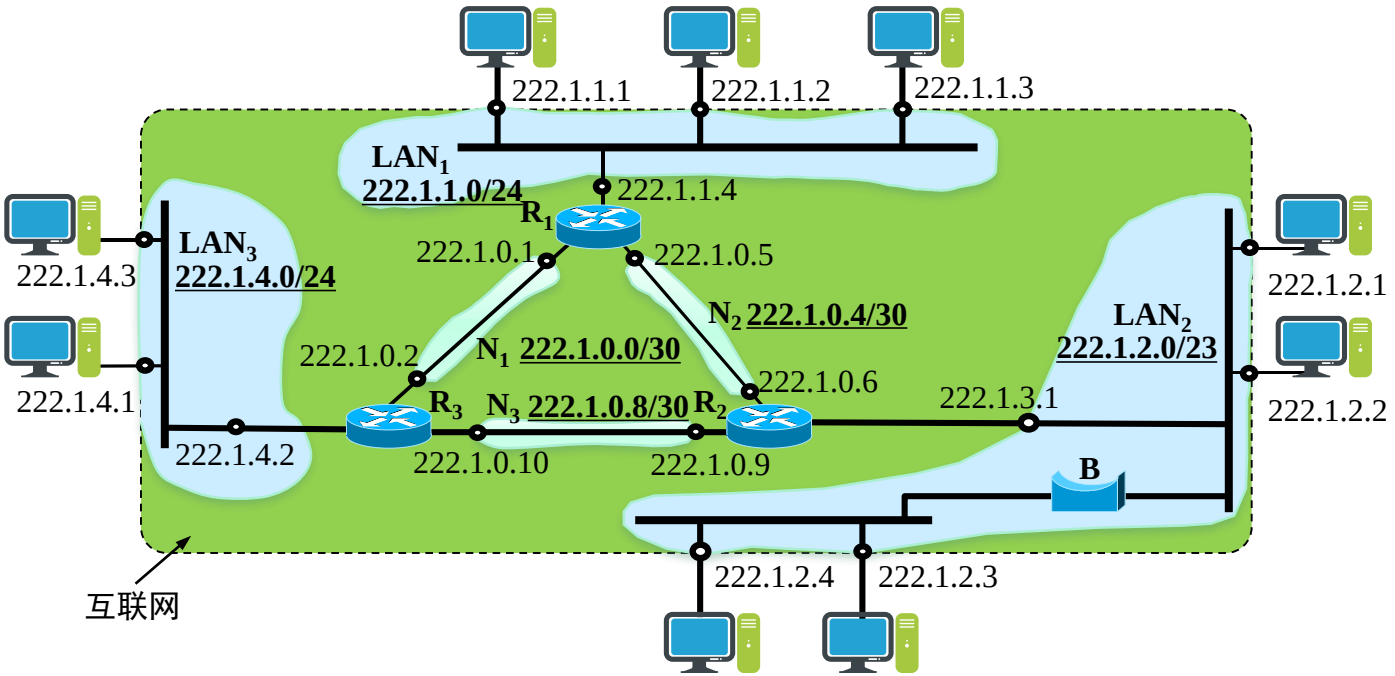
互联网中的 IP 地址

各网络的子网掩码可以不同，即网络前缀的长度可以不同，因此各自的地址空间大小也不相同。



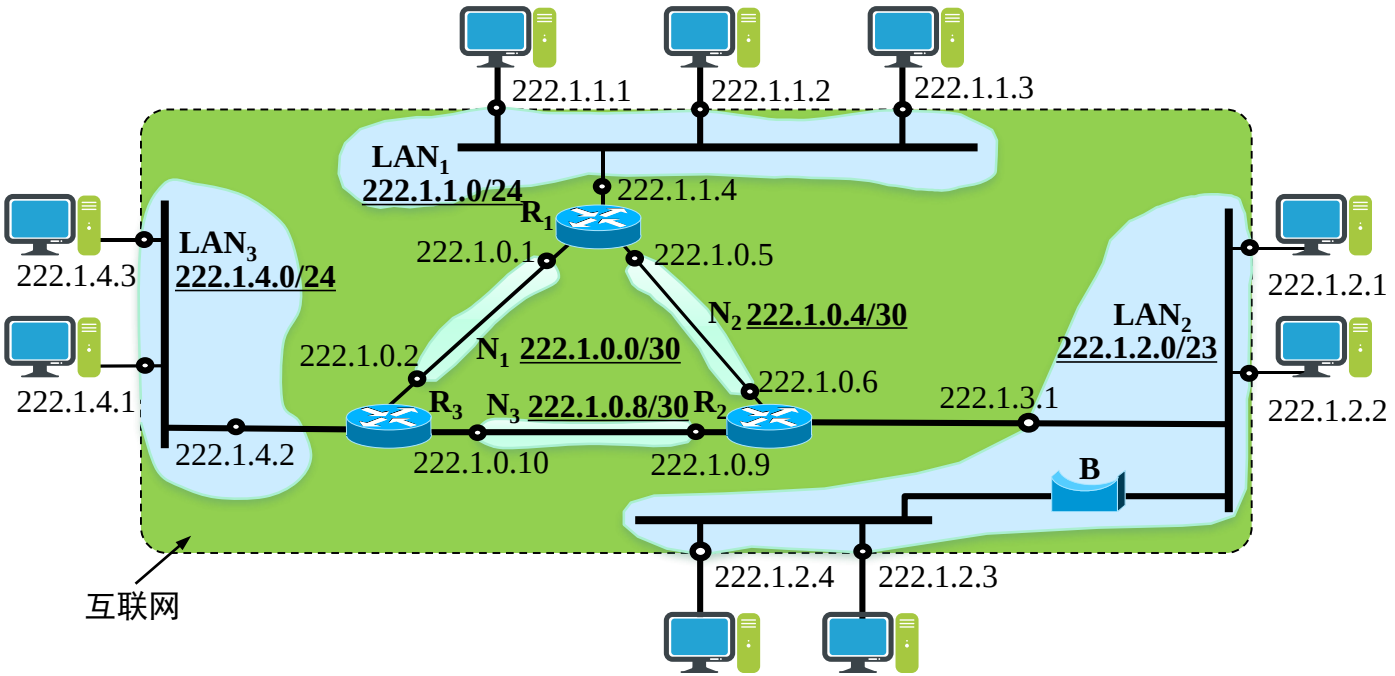
互联网中的 IP 地址

连接在同一个网络上的主机或路由器的IP地址的网络前缀必须与该网络的网络前缀一样。



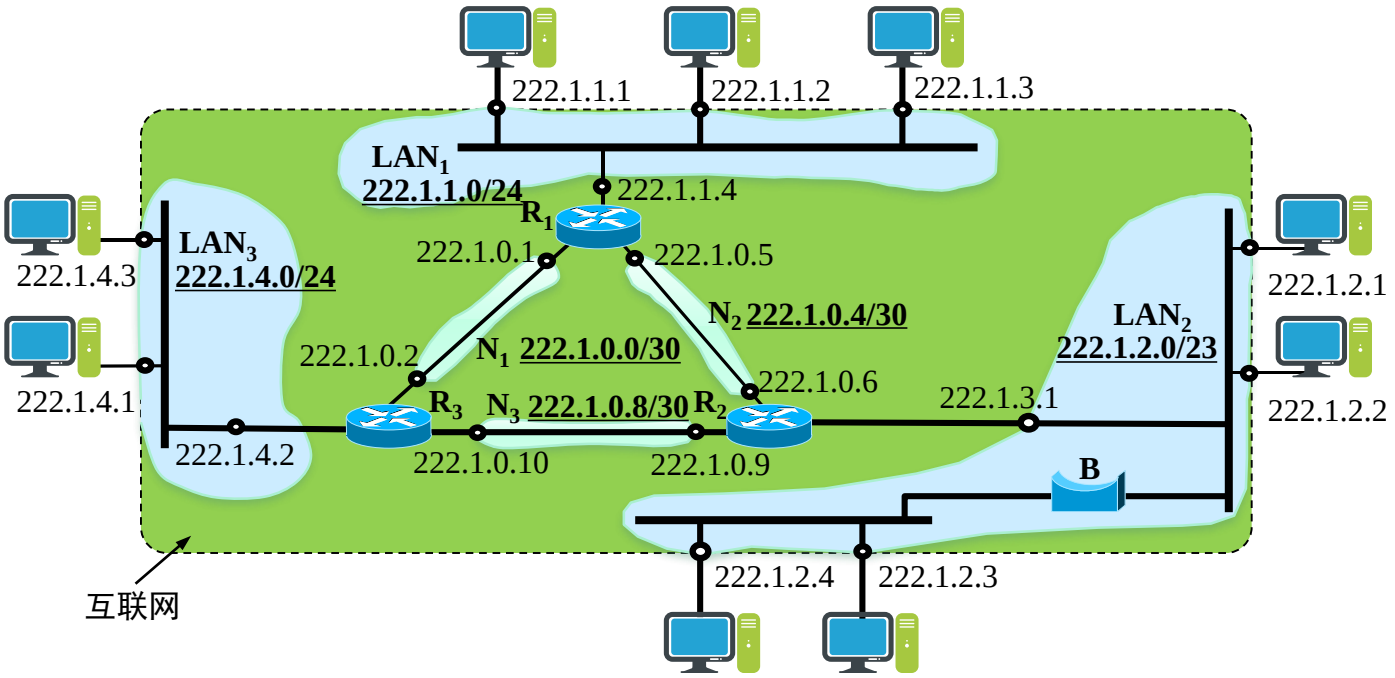
互联网中的 IP 地址

用网桥（它只在链路层工作）互连的网段仍然是一个网络，只能有一个网络地址或网络前缀。



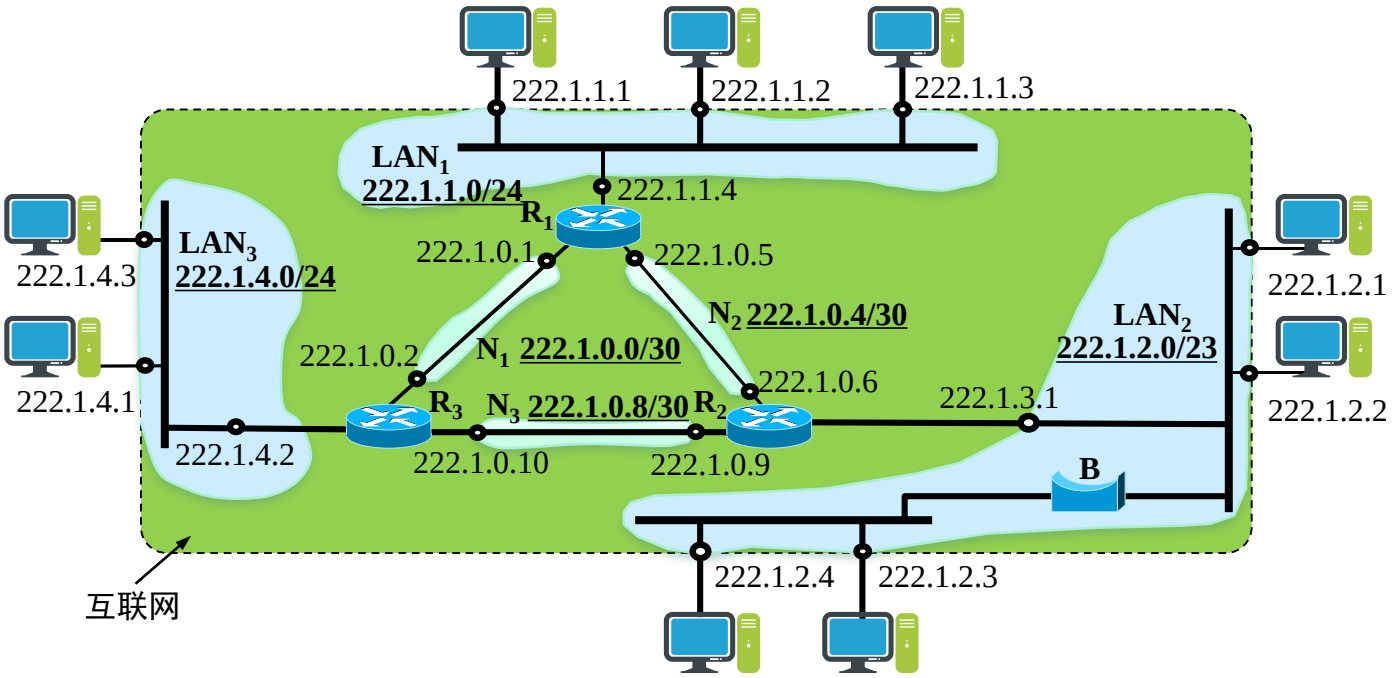
互联网中的 IP 地址

主机号全为0和全为1的IP地址有特殊用途（后面将要介绍），不能分配给主机或路由器使用。



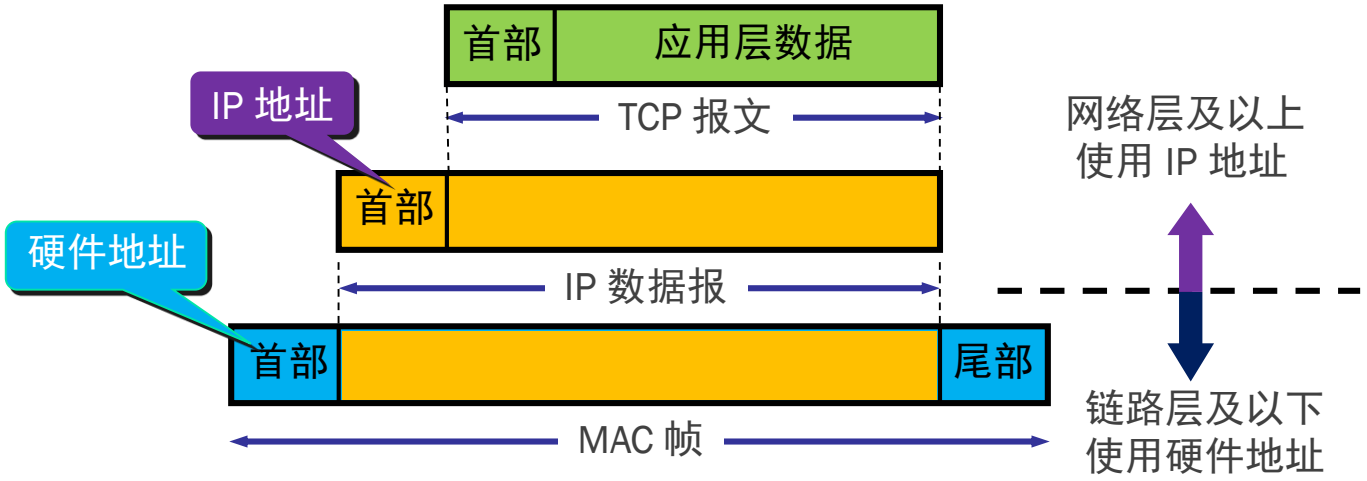
互联网中的 IP 地址

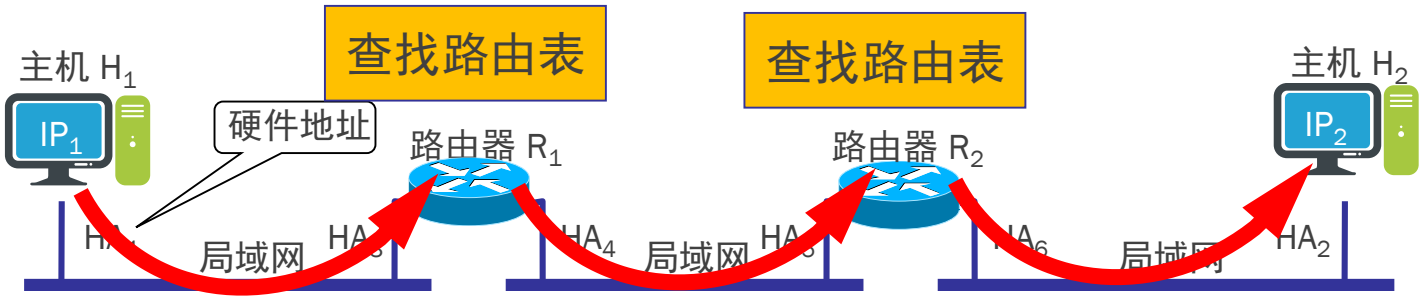
当两个路由器直接相连时（例如通过一条租用线路），在连线两端的接口处，可以分配也可以不分配IP地址。



4.2.3 IP 地址与物理地址

物理地址（也称为硬件地址）

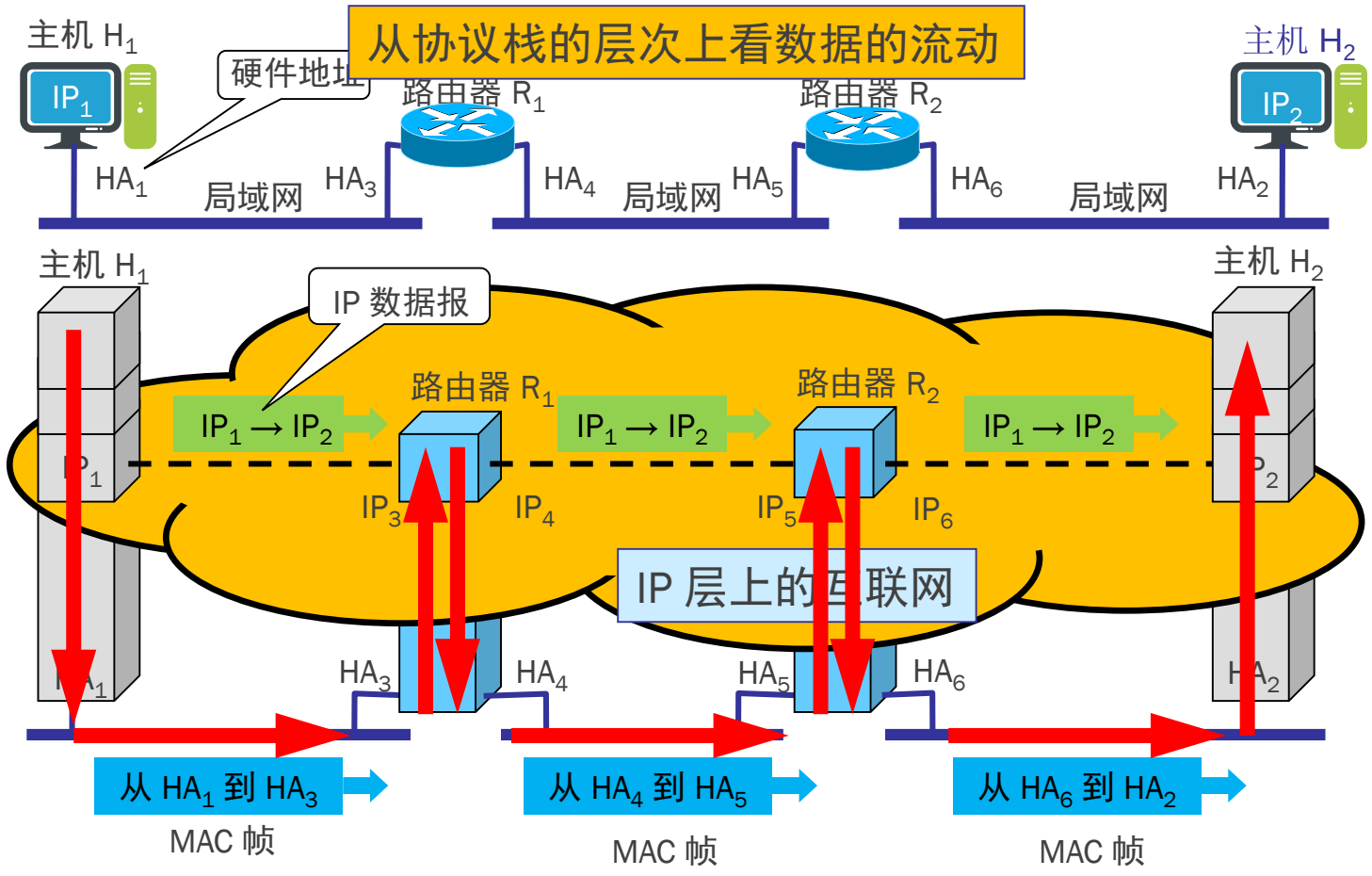




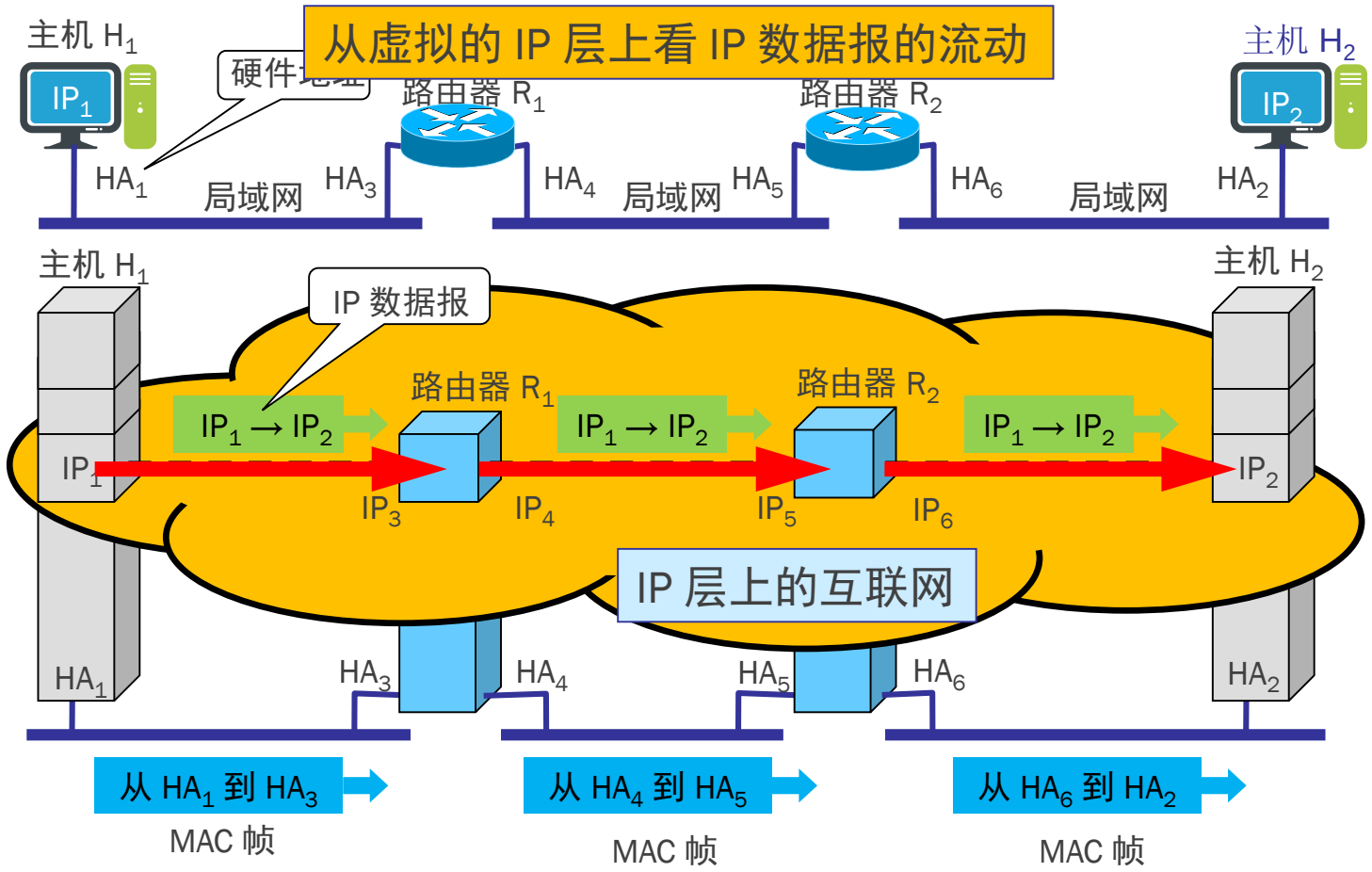
通信的路径

H_1→经过 R_1 转发→再经过 R_2 转发→H_2

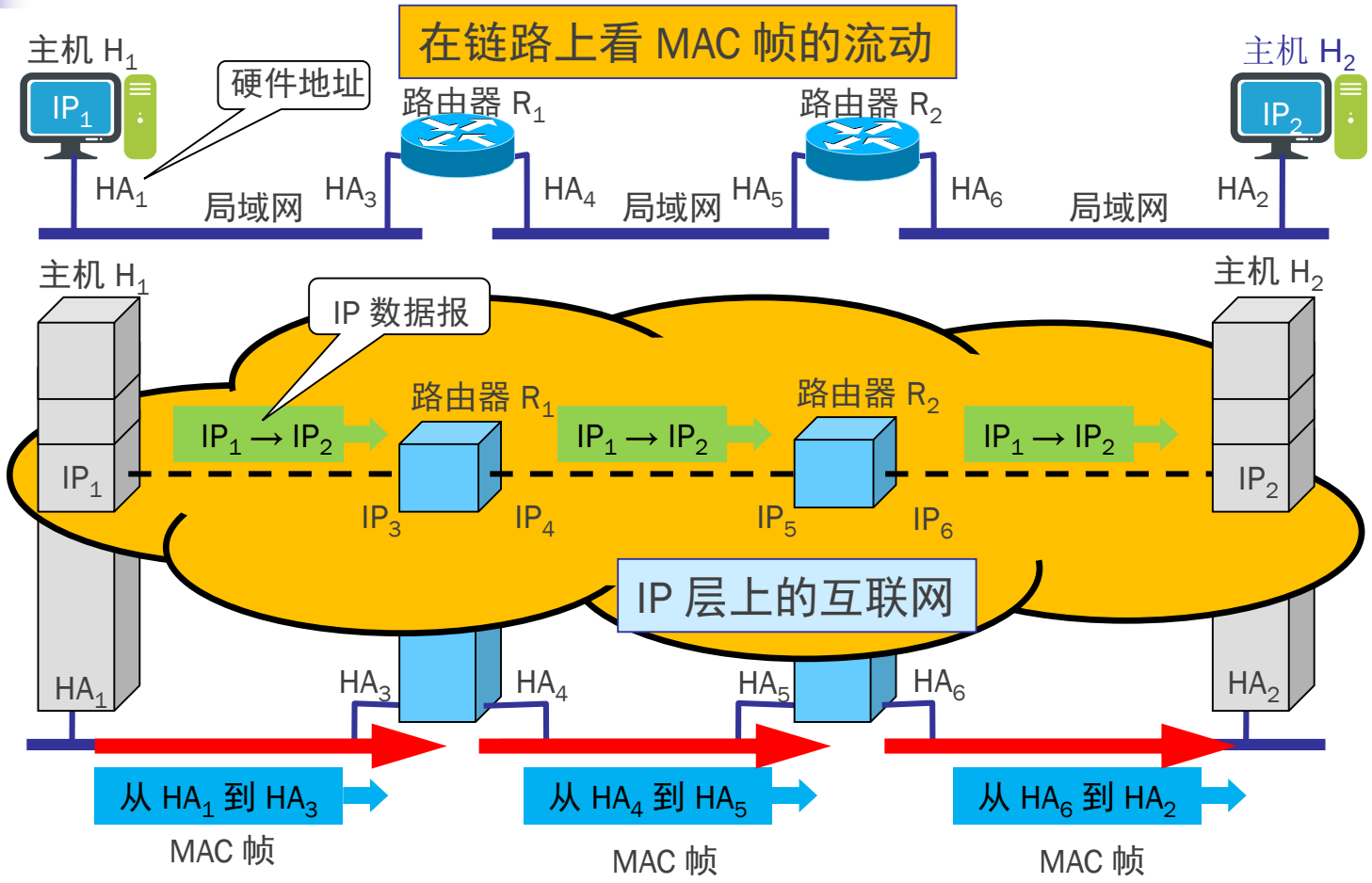
从协议栈的层次上看数据的流动



从虚拟的 IP 层上看 IP 数据报的流动

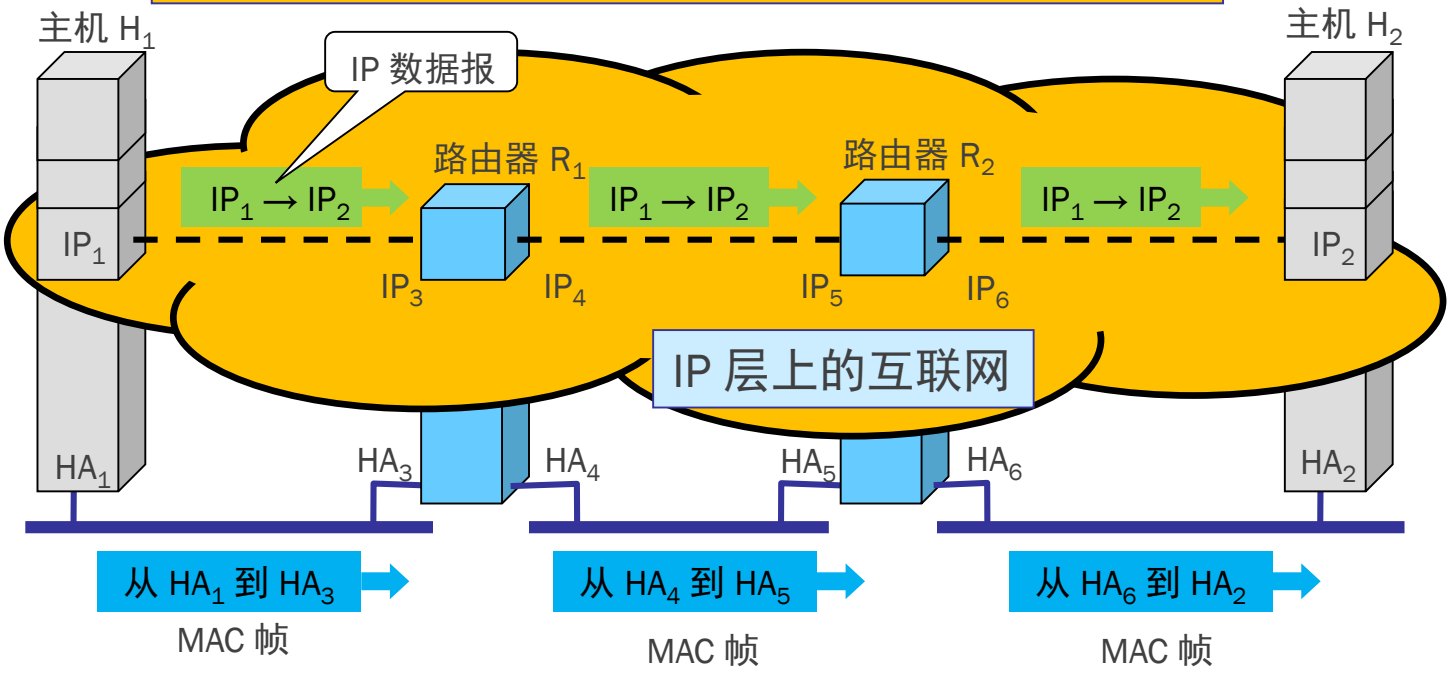


在链路上看 MAC 帧的流动



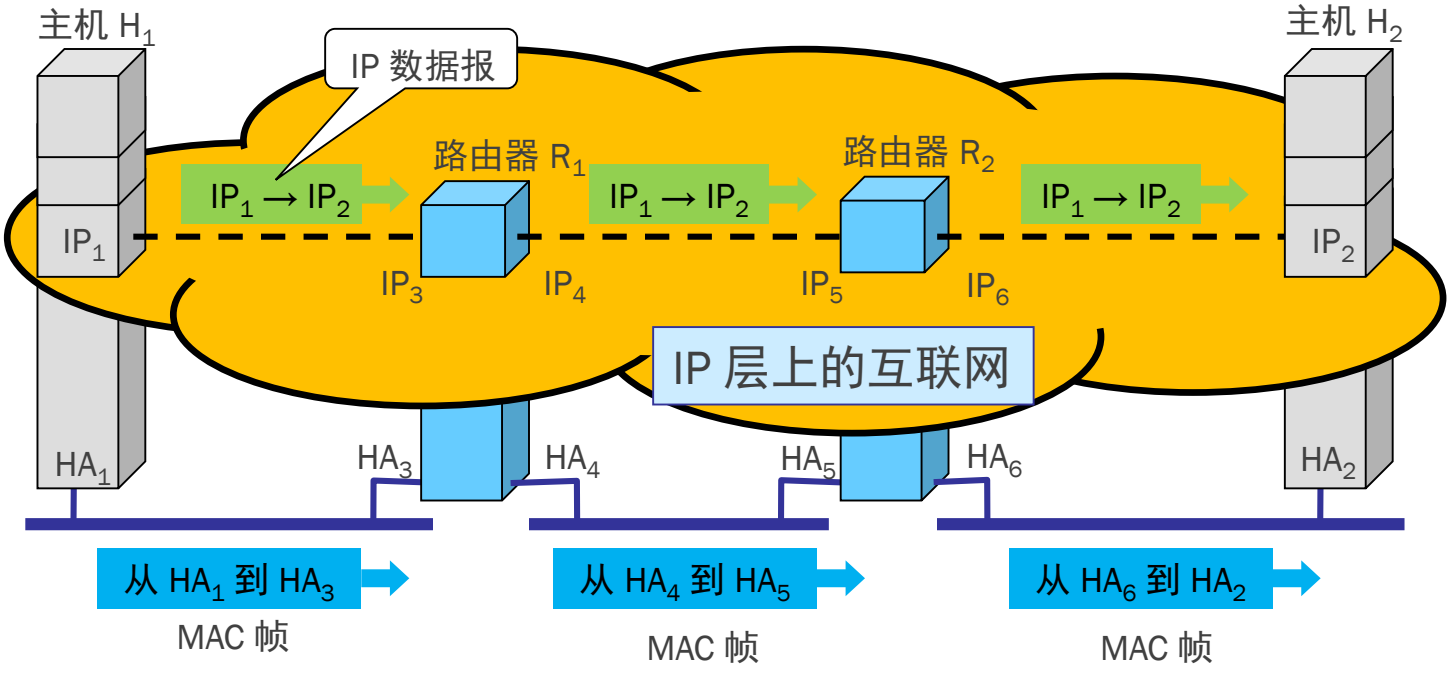
在链路上看 MAC 帧的流动

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



在链路上看 MAC 帧的流动

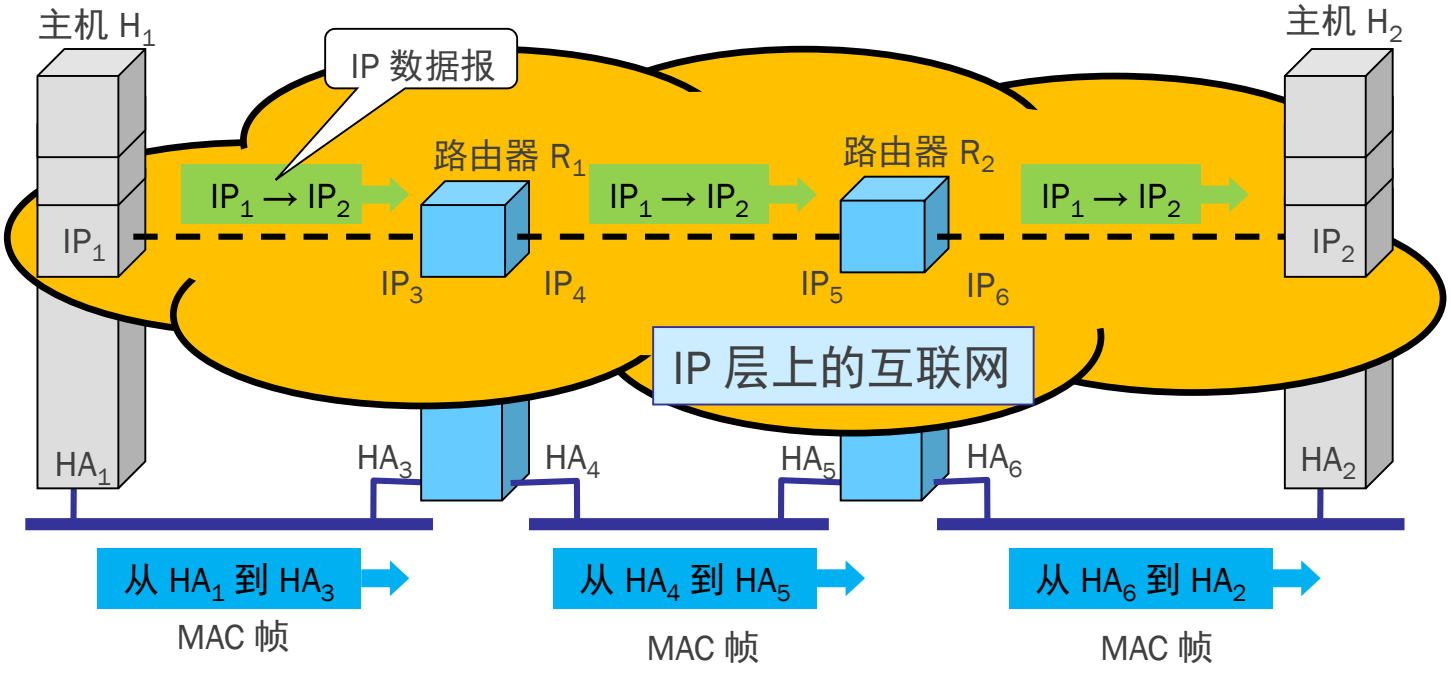
路由器只根据目的站的 IP 地址的网络号进行路由选择



在链路上看 MAC 帧的流动

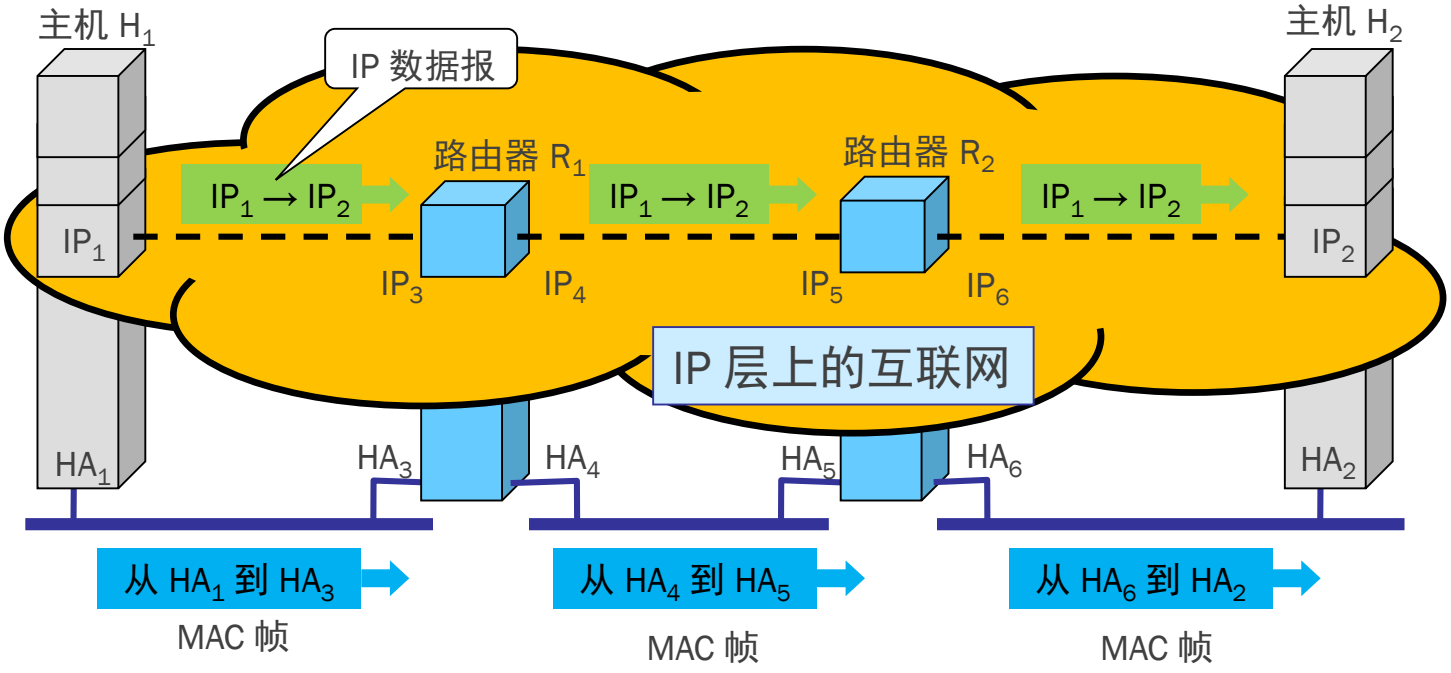
在具体的物理网络的链路层

只能看见 MAC 帧而看不见 IP 数据报

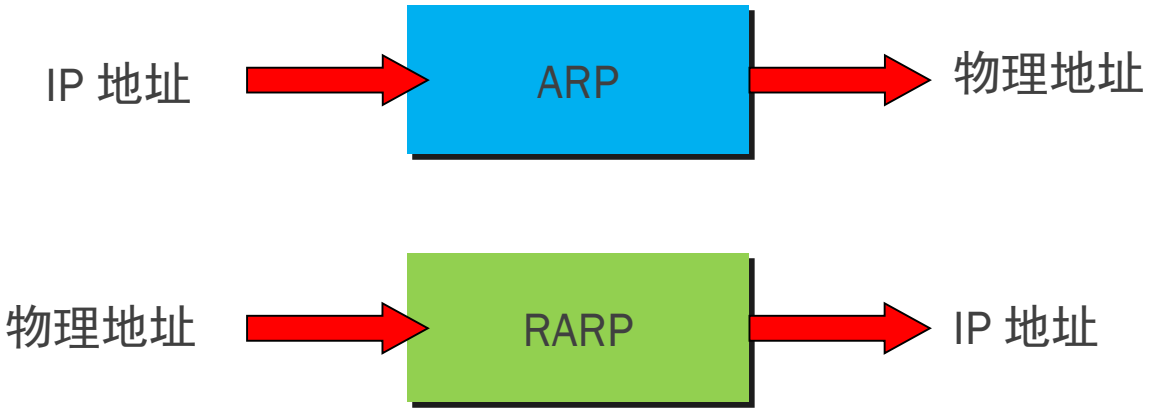


在链路上看 MAC 帧的流动

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



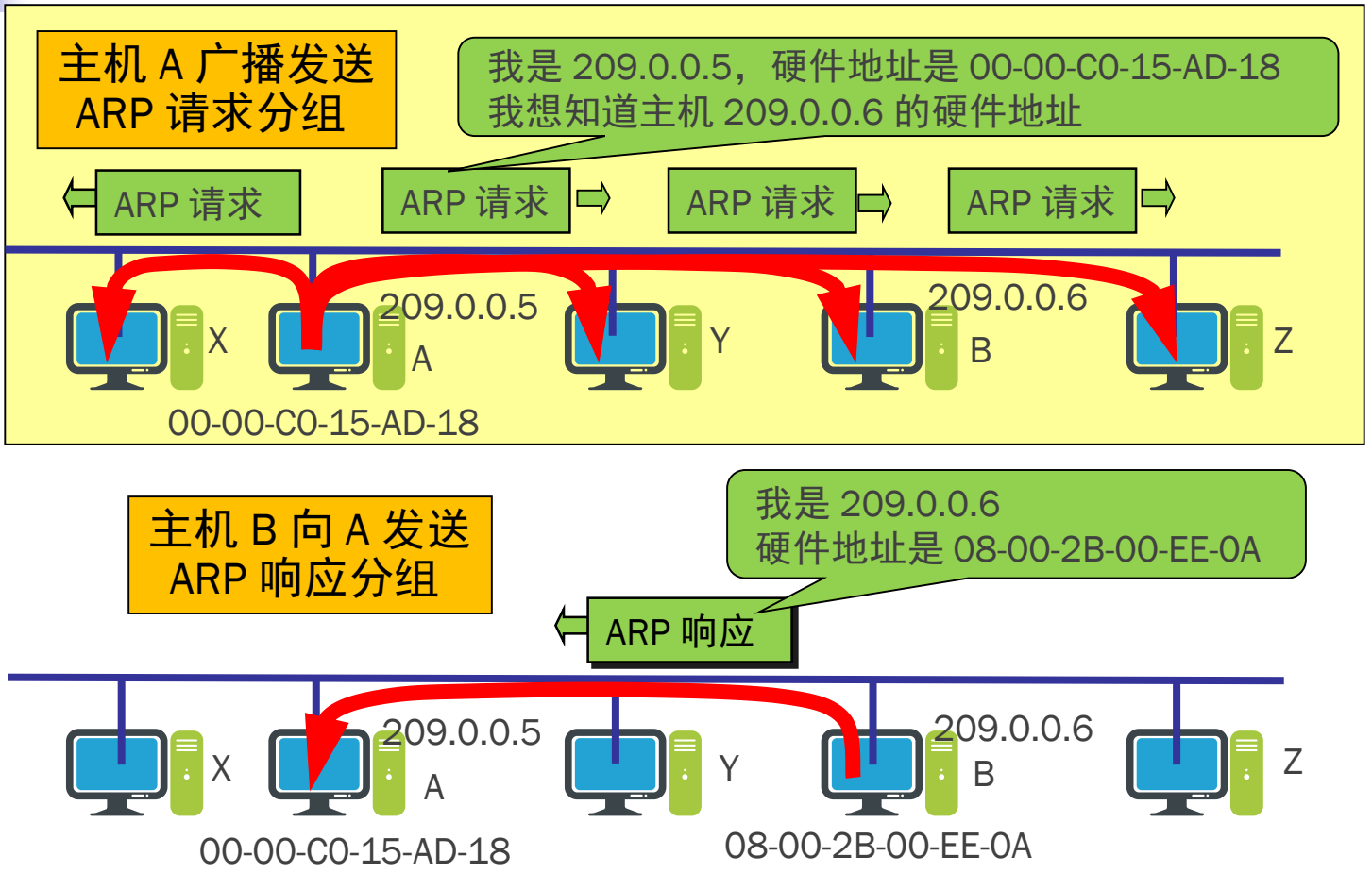
4.2.4 地址解析协议 ARP



地址解析协议 ARP

- 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。
- 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。

地址解析协议 ARP


主机 A 广播发送
ARP 请求分组
我是 209.0.0.5，硬件地址是 00-00-C0-15-AD-18
我想知道主机 209.0.0.6 的硬件地址
ARP 请求
ARP 请求
ARP 请求
ARP 请求
X
209.0.0.5
A
Y
209.0.0.6
B
Z
00-00-C0-15-AD-18
主机 B 向 A 发送
ARP 响应分组
我是 209.0.0.6
硬件地址是 08-00-2B-00-EE-0A
ARP 响应
X
209.0.0.5
A
Y
209.0.0.6
B
Z
00-00-C0-15-AD-18
08-00-2B-00-EE-0A

ARP 高速缓存的作用

- 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。

地址映射项目的生存时间

- ARP把保存在高速缓存中的每一个映射地址项目都设置生存时间（例如，10 ~ 20分钟）。凡超过生存时间的项目就从高速缓存中删除掉。
- 设置这种地址映射项目的生存时间是为了保证高速缓存中信息的新鲜性。

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。
- 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

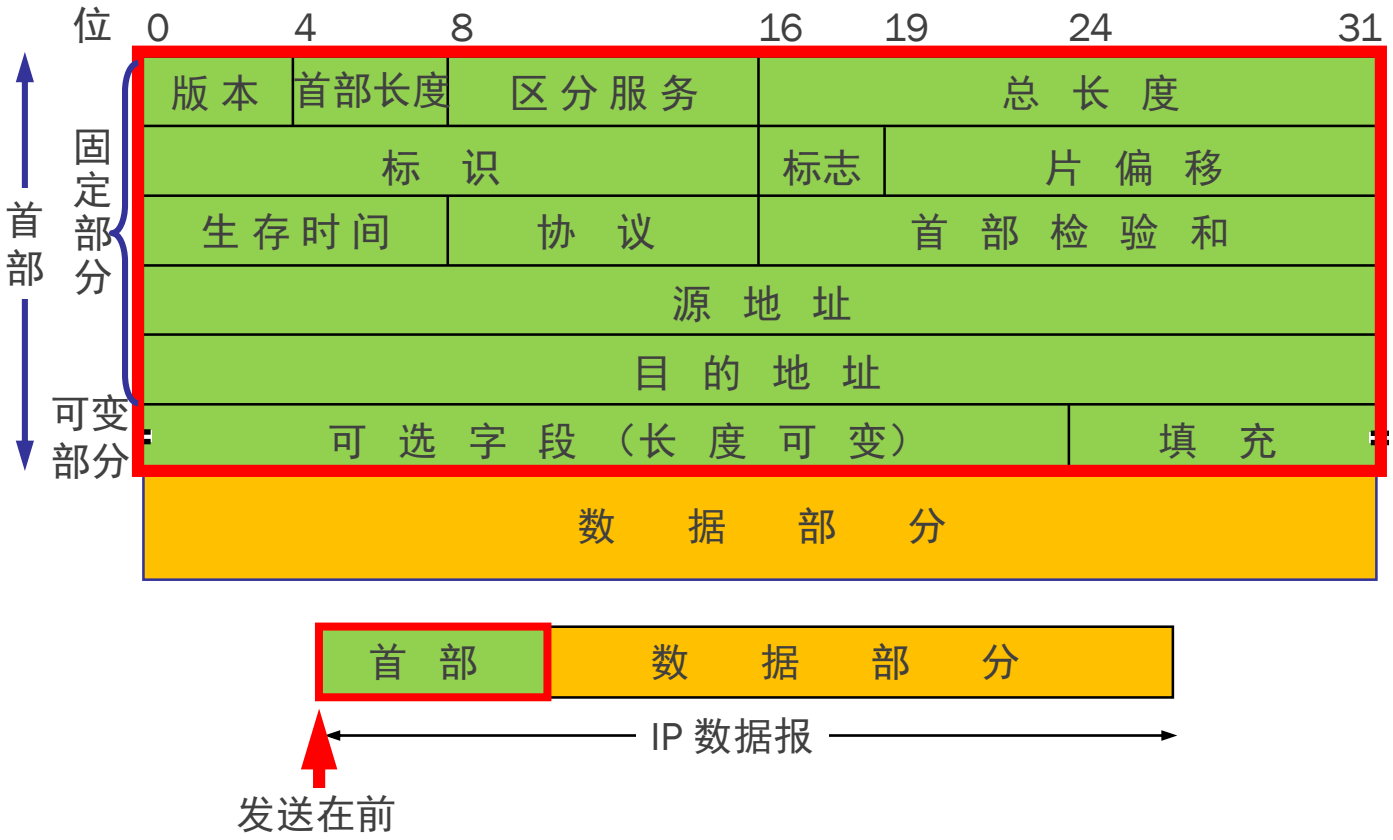
- 发送方是主机，要把IP数据报发送到本网络上的另一个主机。这时用 ARP 找到目的主机的硬件地址。
- 发送方是主机，要把 IP 数据报发送到另一个网络上的一个主机。这时用 ARP 找到本网络上的一个路由器的硬件地址。剩下的工作由这个路由器来完成。
- 发送方是路由器，要把 IP 数据报转发到本网络上的一个主机。这时用 ARP 找到目的主机的硬件地址。
- 发送方是路由器，要把 IP 数据报转发到另一个网络上的一个主机。这时用 ARP 找到本网络上的一个路由器的硬件地址。剩下的工作由这个路由器来完成。

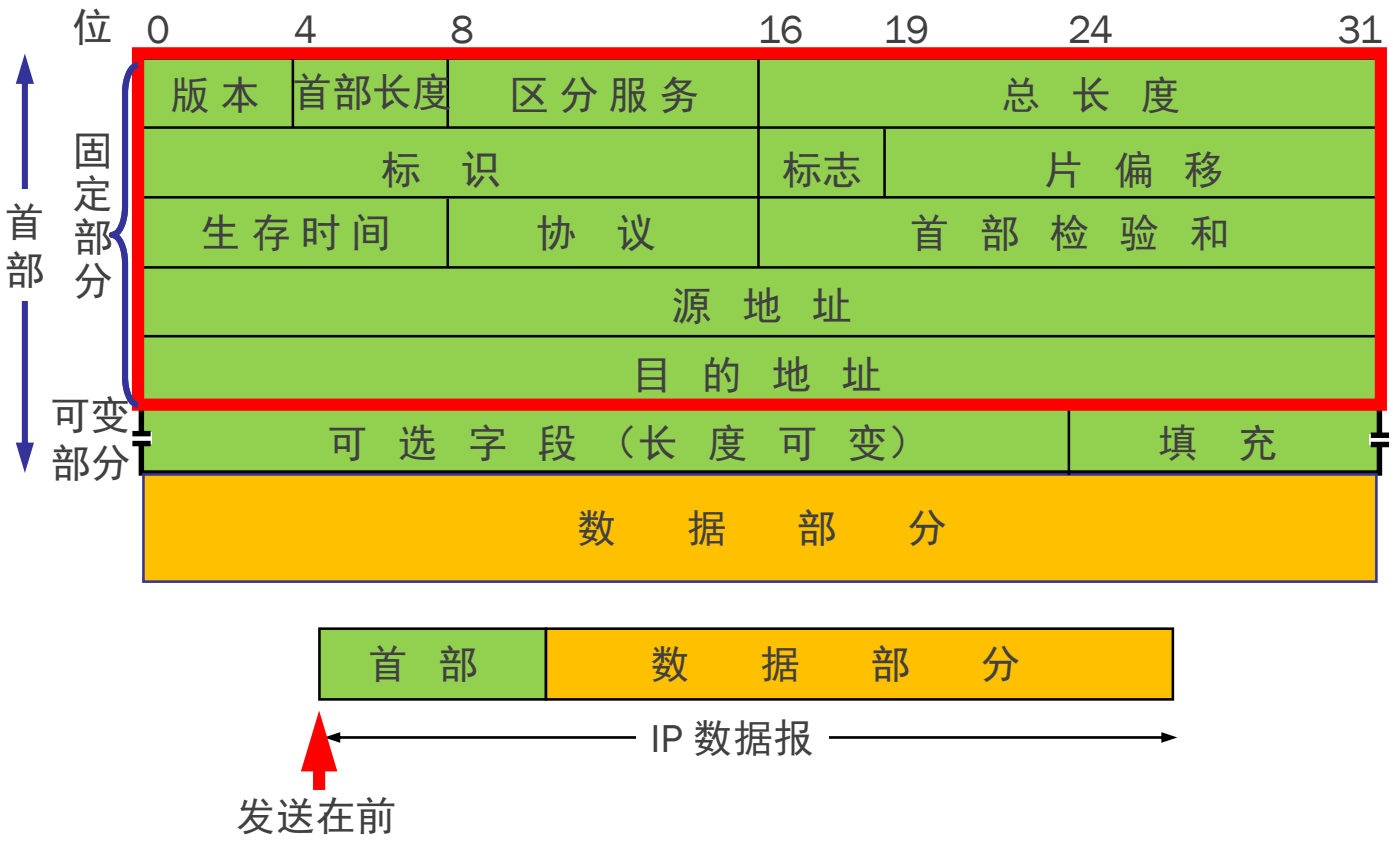
为什么我们不直接使用硬件地址进行通信?

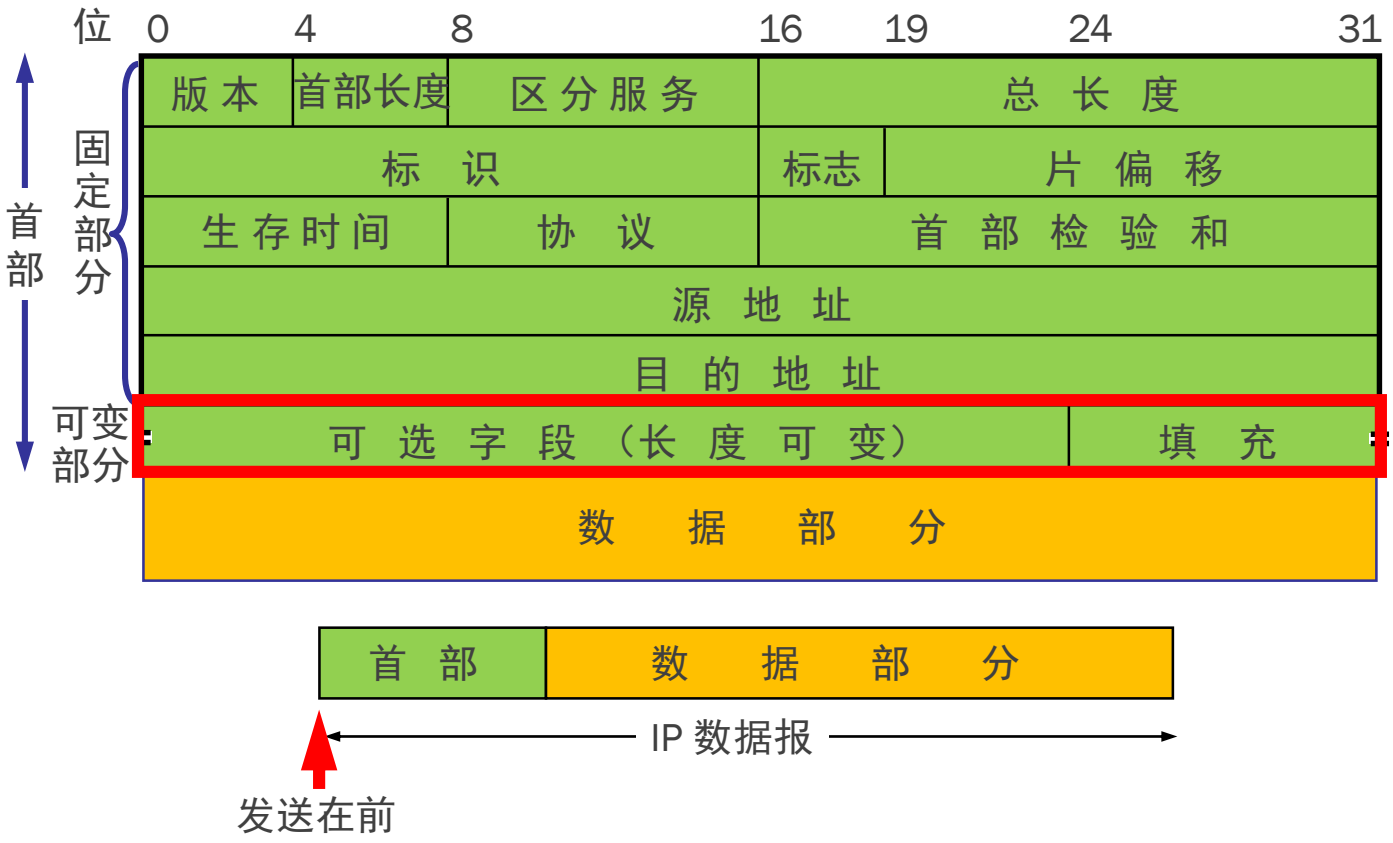
- 由于全世界存在着各式各样的网络，它们使用不同的硬件地址。要使这些异构网络能够互相通信就必须进行非常复杂的硬件地址转换工作，因此几乎是不可能的事。
- 连接到因特网的主机都拥有统一的 IP 地址，它们之间的通信就像连接在同一个网络上那样简单方便，因为调用 ARP 来寻找某个路由器或主机的硬件地址都是由计算机软件自动进行的，对用户来说是看不见这种调用过程的。

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。

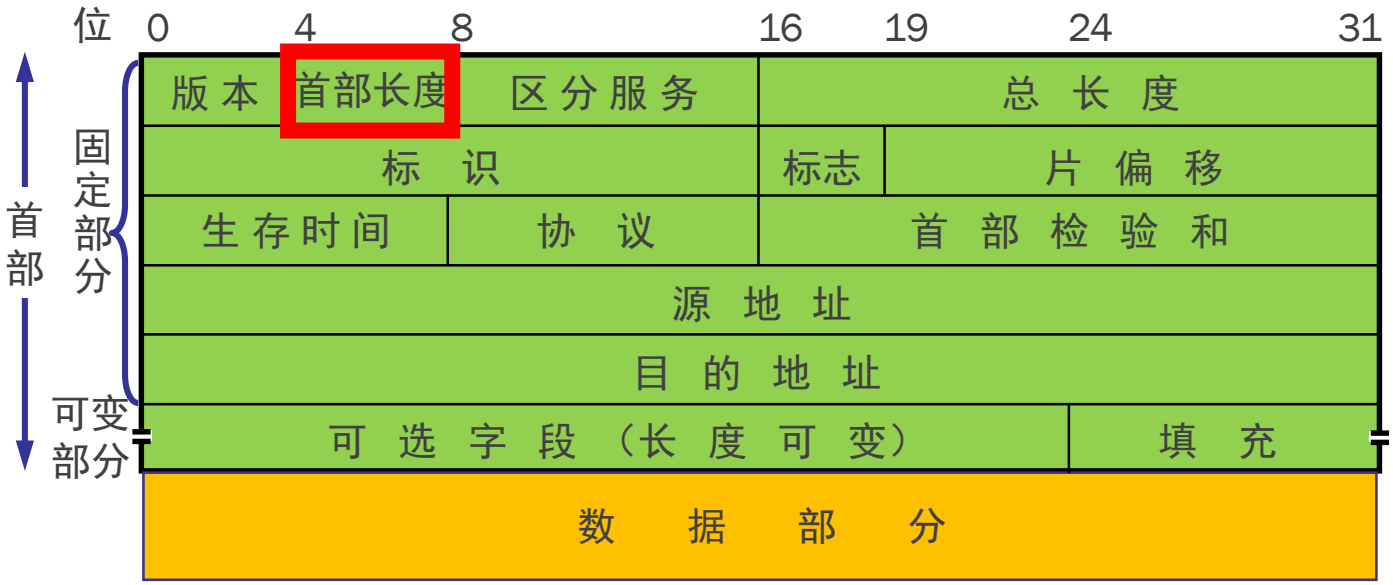

位
0
4
8
16
19
24
31
首部
固定部分
可变部分
版本
首部长度
区分服务
总长度
标识
标志
片偏移
生存时间
协议
首部检验和
源地址
目的地址
可选字段（长度可变）
填充
数据部分
首部
数据部分
IP 数据报
发送在前


位
0
4
8
16
19
24
31
版本
首部长度
区分服务
总长度
标识
标志
片偏移
生存时间
协议
首部检验和
源地址
目的地址
可选字段（长度可变）
填充
数据部分
首部
固定部分
可变部分
首部
数据部分
IP 数据报
发送在前


位
0
4
8
16
19
24
31
版本
首部长度
区分服务
总长度
标识
标志
片偏移
生存时间
协议
首部检验和
源地址
目的地址
可选字段（长度可变）
填充
数据部分
首部
固定部分
可变部分
首部
数据部分
IP 数据报
发送在前

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)



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。

区分服务——占 8 位，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

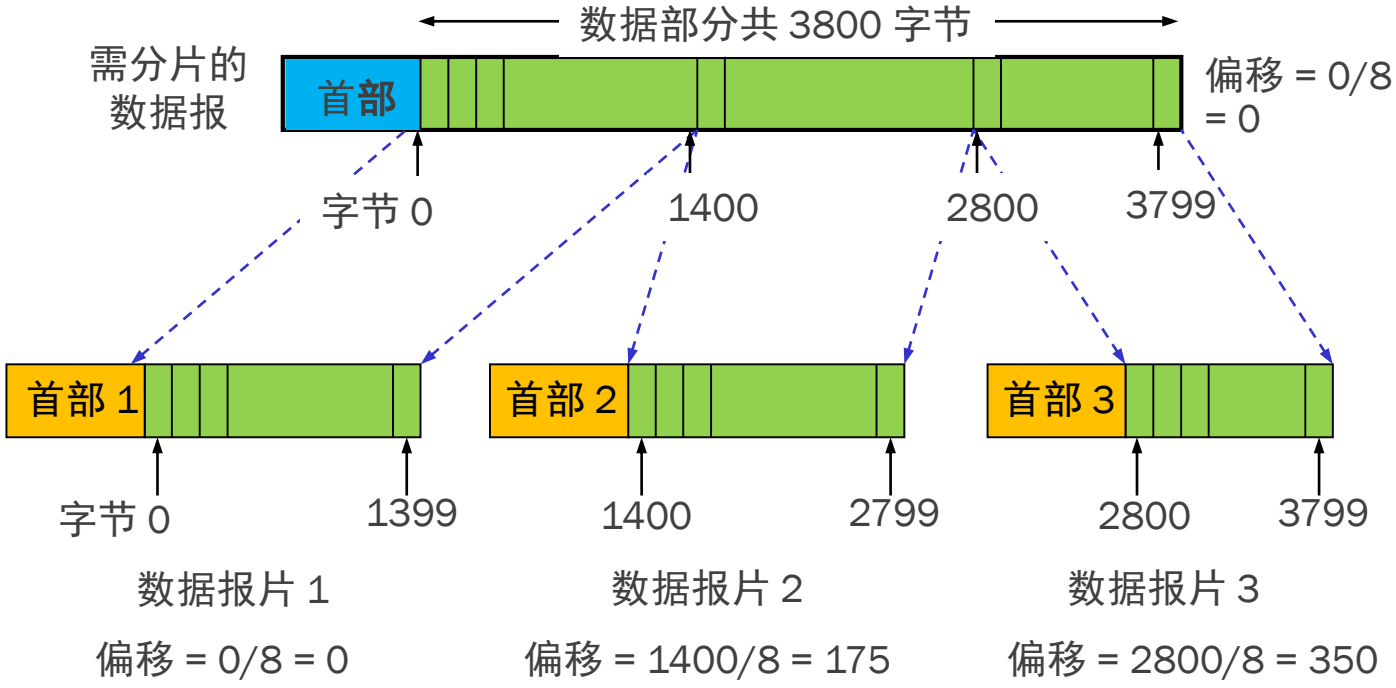
总长度——占 16 位，指首部和数据之和的长度，
单位为字节，因此数据报的最大长度为 65535 字节。
总长度必须不超过最大传送单元 MTU。

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

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。

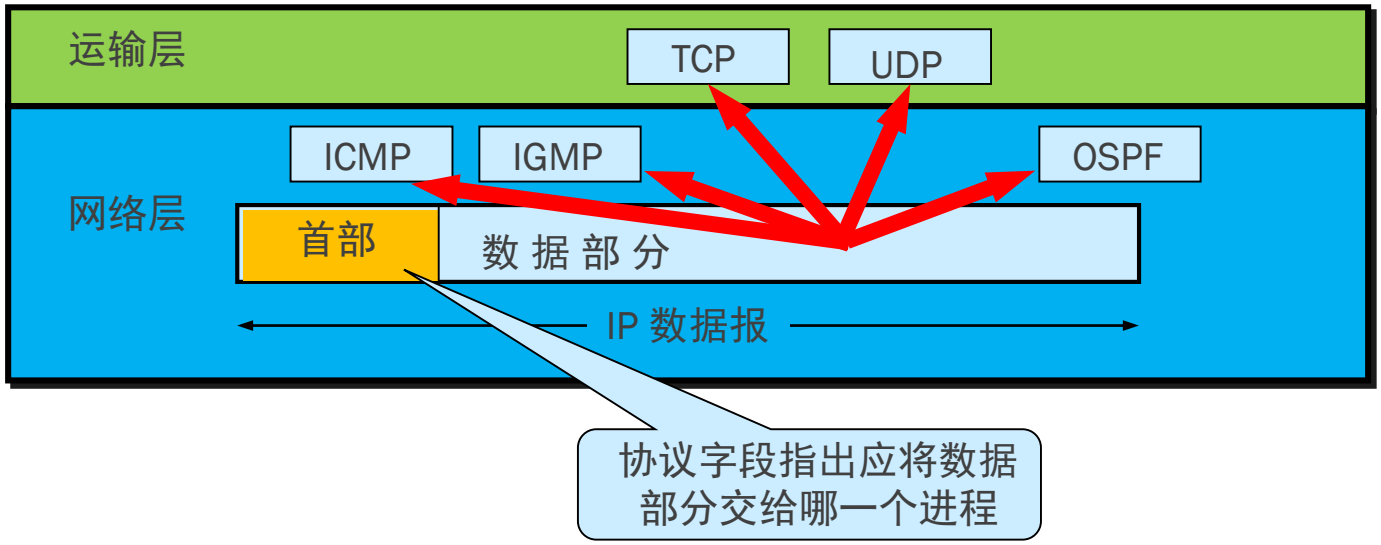
片偏移(12 位)指出：较长的分组在分片后
某片在原分组中的相对位置。
片偏移以 8 个字节为偏移单位。

【例4-3】 IP 数据报分片

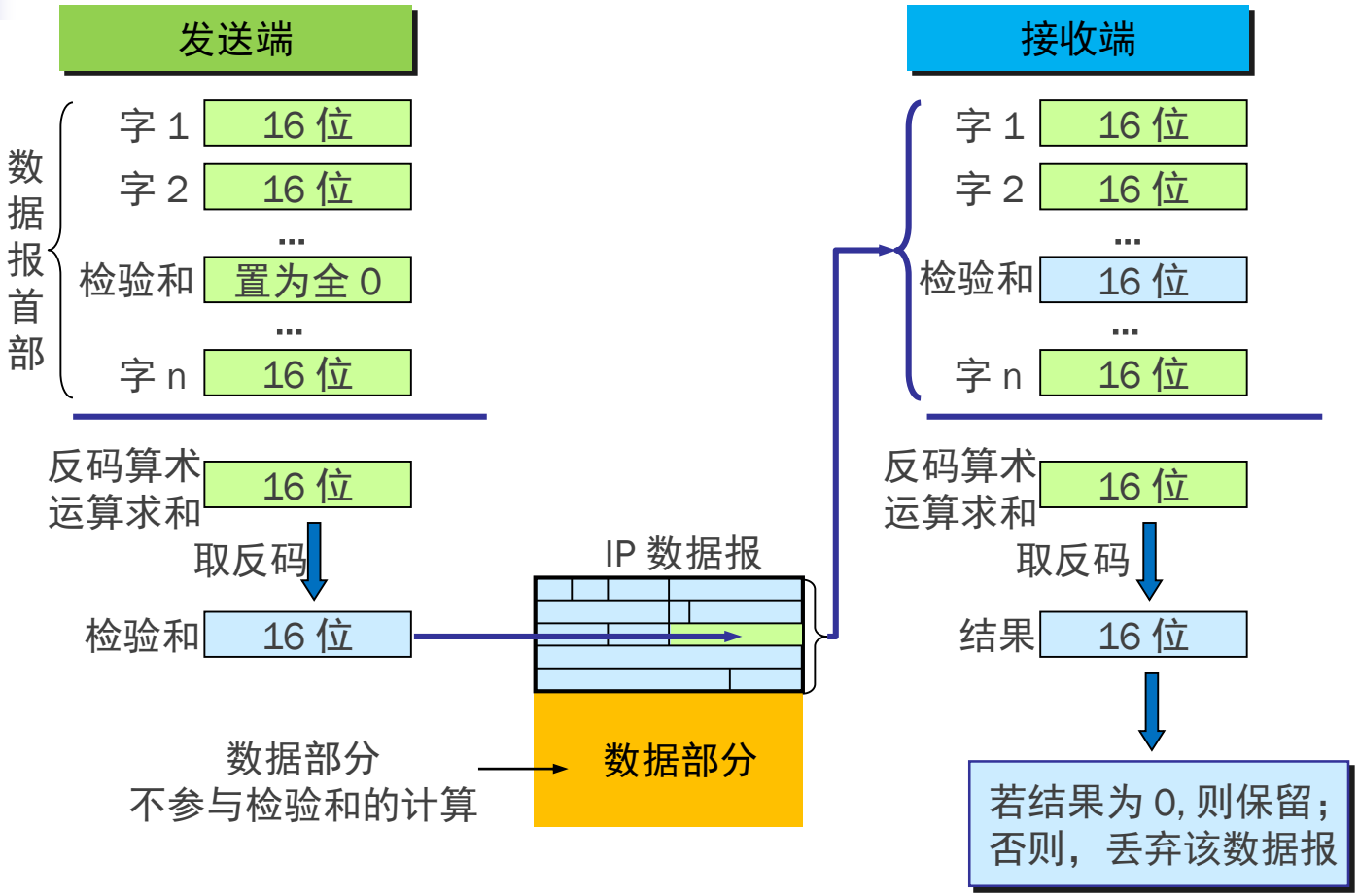

数据部分共 3800 字节
需分片的
数据报
首部
偏移 = 0/8
= 0
字节 0
1400
2800
3799
首部 1
首部 2
首部 3
字节 0
1399
1400
2799
2800
3799
数据报片 1
偏移 = 0/8 = 0
数据报片 2
偏移 = 1400/8 = 175
数据报片 3
偏移 = 2800/8 = 350

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

协议(8 位)字段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哪个处理过程


运输层
TCP
UDP
网络层
ICMP
IGMP
OSPF
首部
数 据 部 分
IP 数据报
协议字段指出应将数据
部分交给哪一个进程

首部检验和(16 位)字段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。


发送端
接收端
数据报首部
字 1
16 位
字 2
16 位
...
检验和
置为全 0
16 位
...
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
结果
16 位
若结果为 0, 则保留；
否则，丢弃该数据报

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

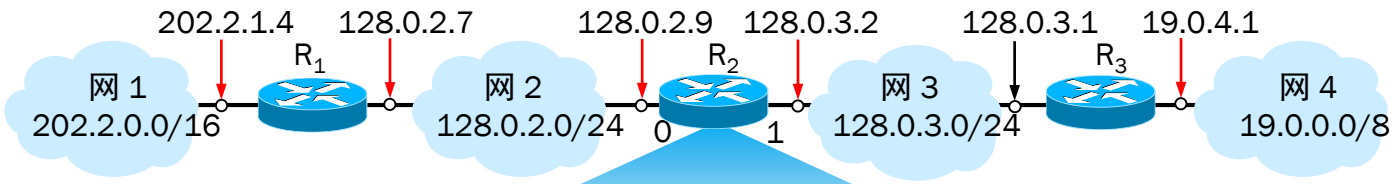
- IP 首部的可变部分就是一个选项字段，用来支持排错、测量以及安全等措施，内容很丰富。
- 选项字段的长度可变，从 1 个字节到 40 个字节不等，取决于所选择的项目。
- 增加首部的可变部分是为了增加 IP 数据报的功能，但这同时也使得 IP 数据报的首部长度成为可变的。这就增加了每一个路由器处理数据报的开销。
- 实际上这些选项很少被使用。

4.2.6 IP 数据报的转发

- 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，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。这样就可使路由表大大简化。

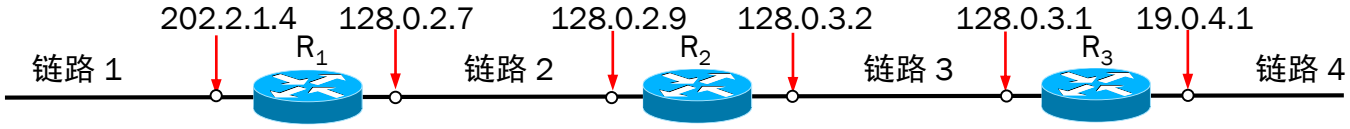
1. 路由表

在路由表中，对每一条路由，最主要的是
（网络地址，掩码，下一跳地址）



路由器 R_2 的路由表

网络地址	子网掩码	下一跳	接口
128.0.2.0	255.255.255.0	--	0
128.0.3.0	255.255.255.0	--	1
202.2.0.0	255.255.0.0	128.0.2.7	0
19.0.0.0	255.0.0.0	128.0.3.1	1



查找路由表

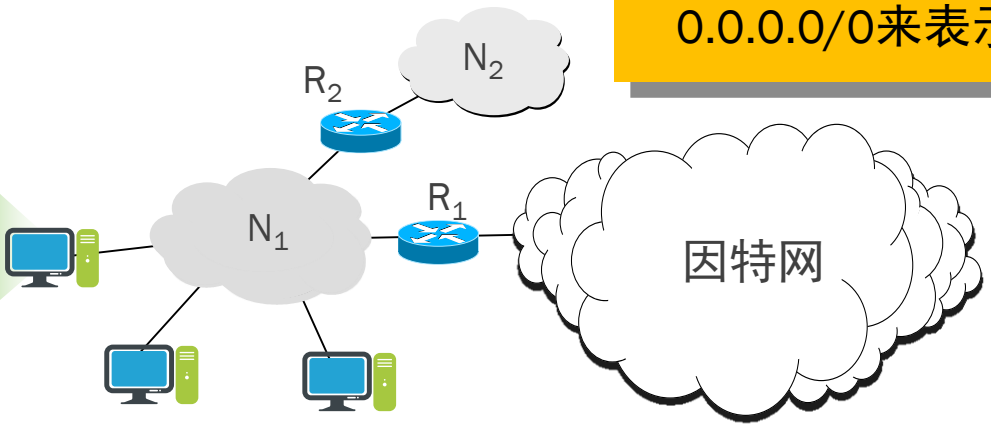
- 根据目的网络地址就能确定下一跳路由器，这样做的结果是：
- IP 数据报最终一定可以找到目的主机所在目的网络上的路由器（可能要通过多次的间接交付）。
- 只有到达最后一个路由器时，才试图向目的主机进行直接交付。

默认路由(default route)

默认路由用网络前缀0.0.0.0/0来表示

路由表

目的网络	下一跳
N_1	直接
N_2	R_2
默认	R_1



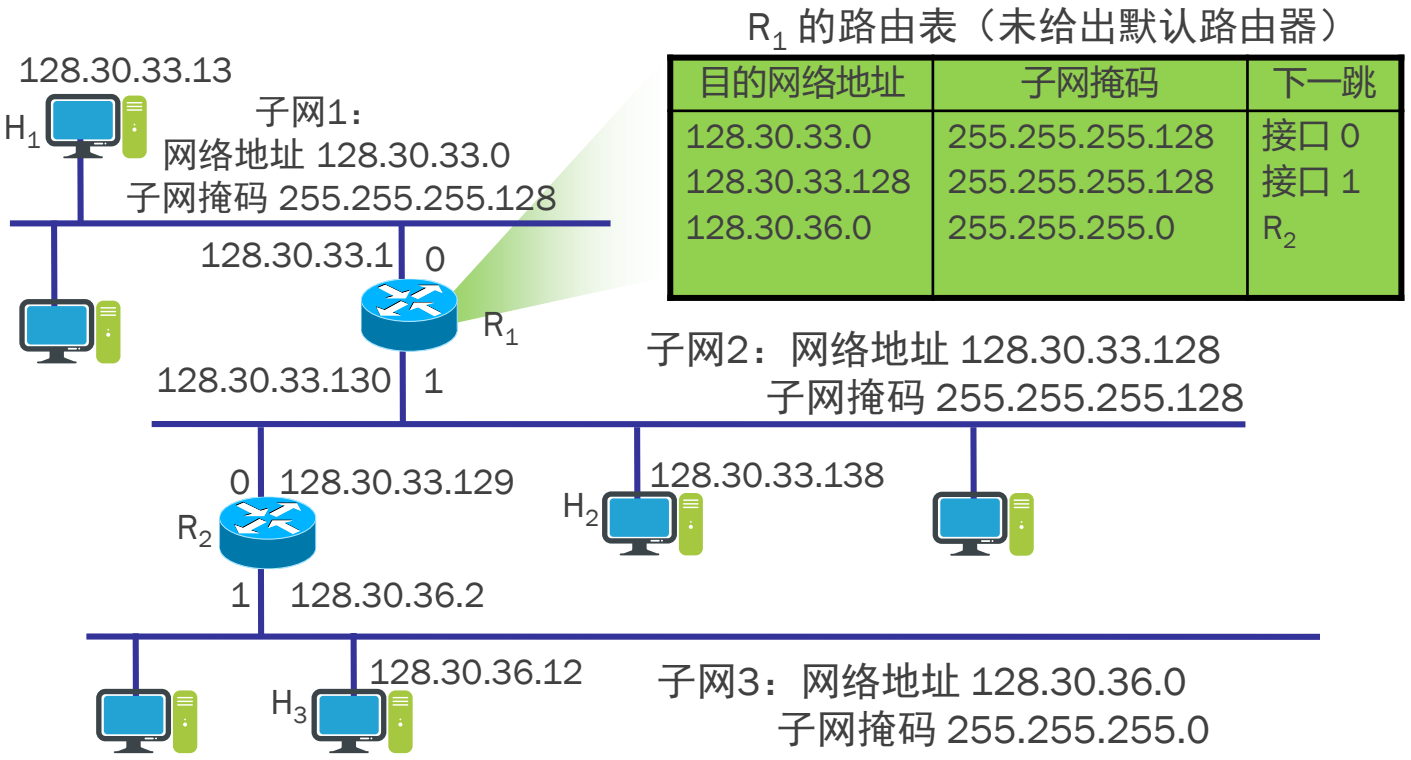
只要目的网络不是 N_1 和 N_2，
就一律选择默认路由，
把数据报先间接交付路由器 R_1，
让 R_1 再转发给下一个路由器。

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因特网连接，那么在这种情况下使用默认路由是非常合适的。

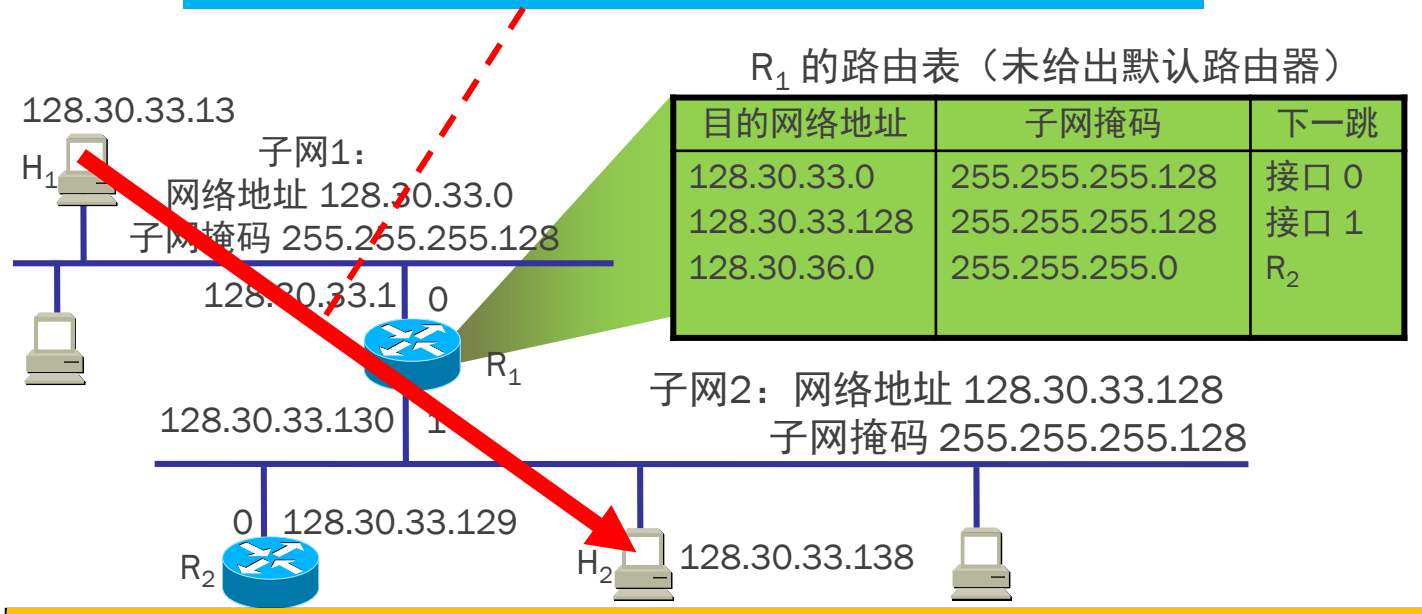
【例4-4】

- **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。**



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 要发送分组给 H_2

- 主机 H_1 首先将本子网的子网掩码 255.255.255.128与分组的 IP 地址 128.30.33.138 逐比特相“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

逐比特 AND 操作

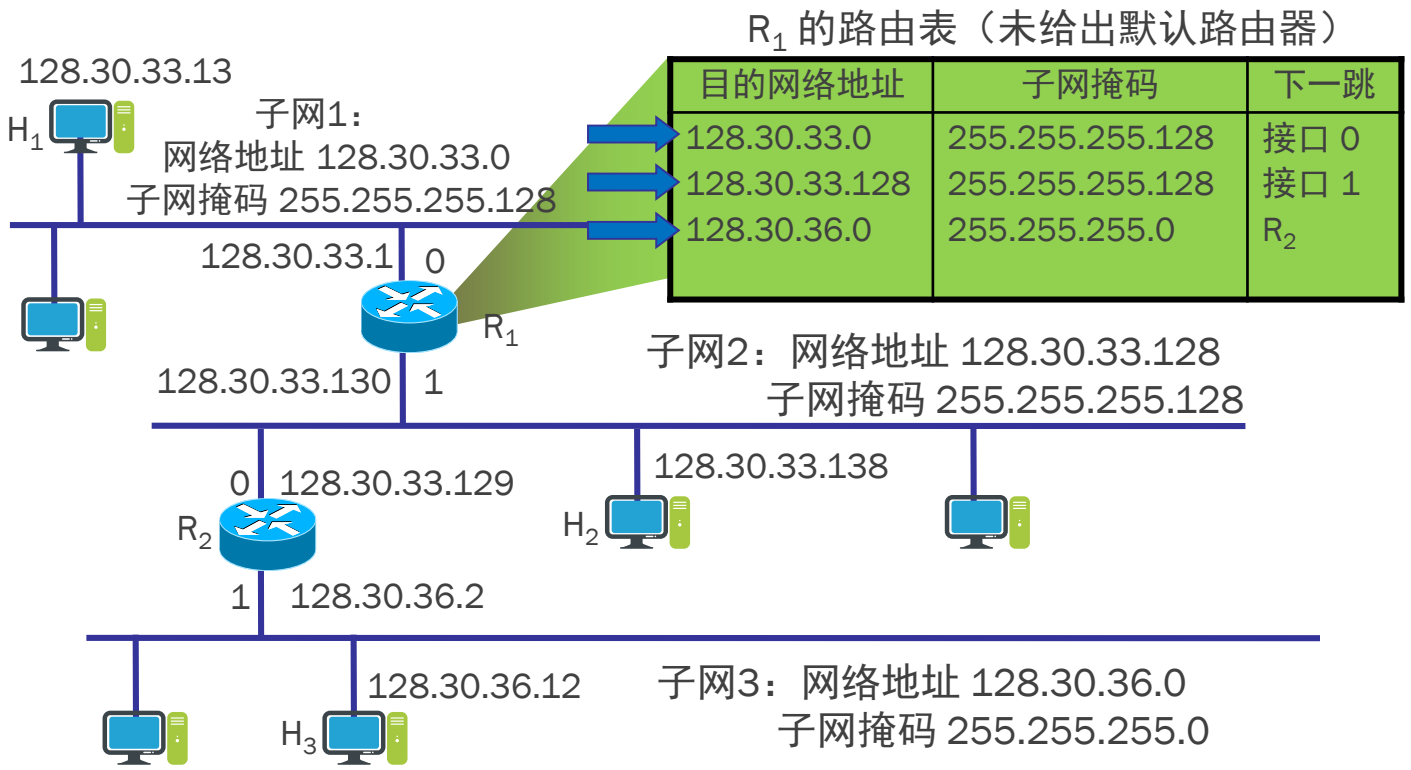
255.255.255.128

128. 30. 33.138

128. 30. 33.128 ≠ H_1 的网络地址

主机 H_1 要发送分组给 H_2

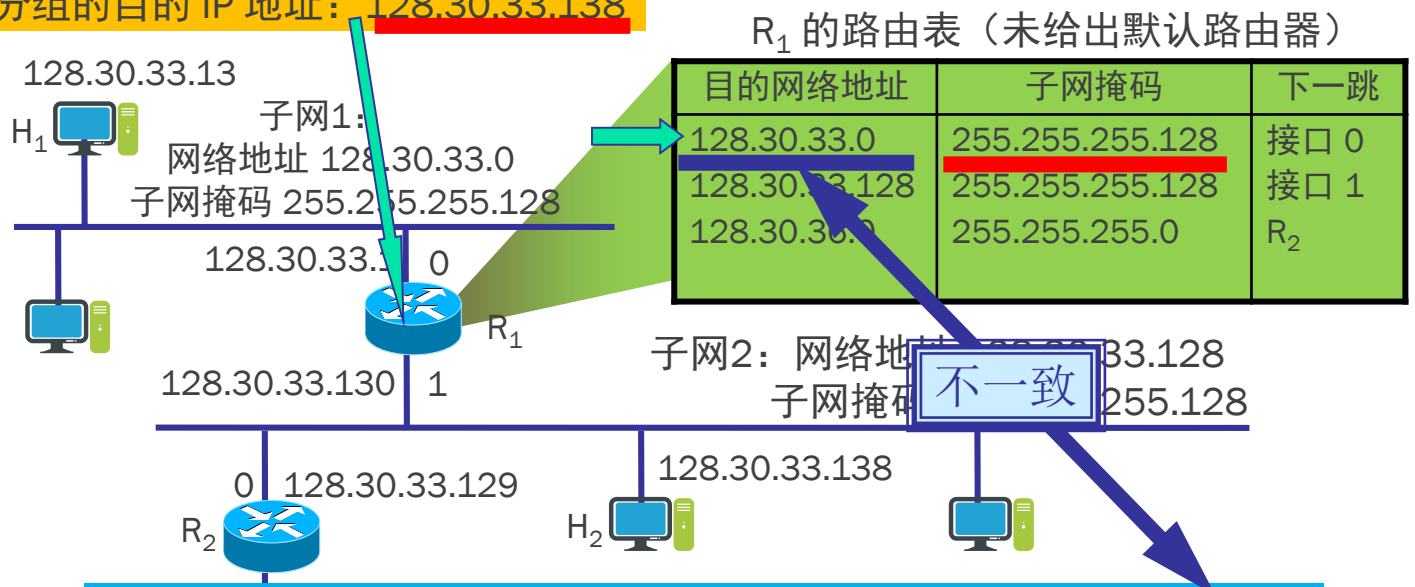
- 因此 H_1 必须把分组传送到路由器 R_1然后逐项查找路由表



主机 H_1 要发送分组给 H_2

- 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
128.30.30.0	255.255.255.0	R_2

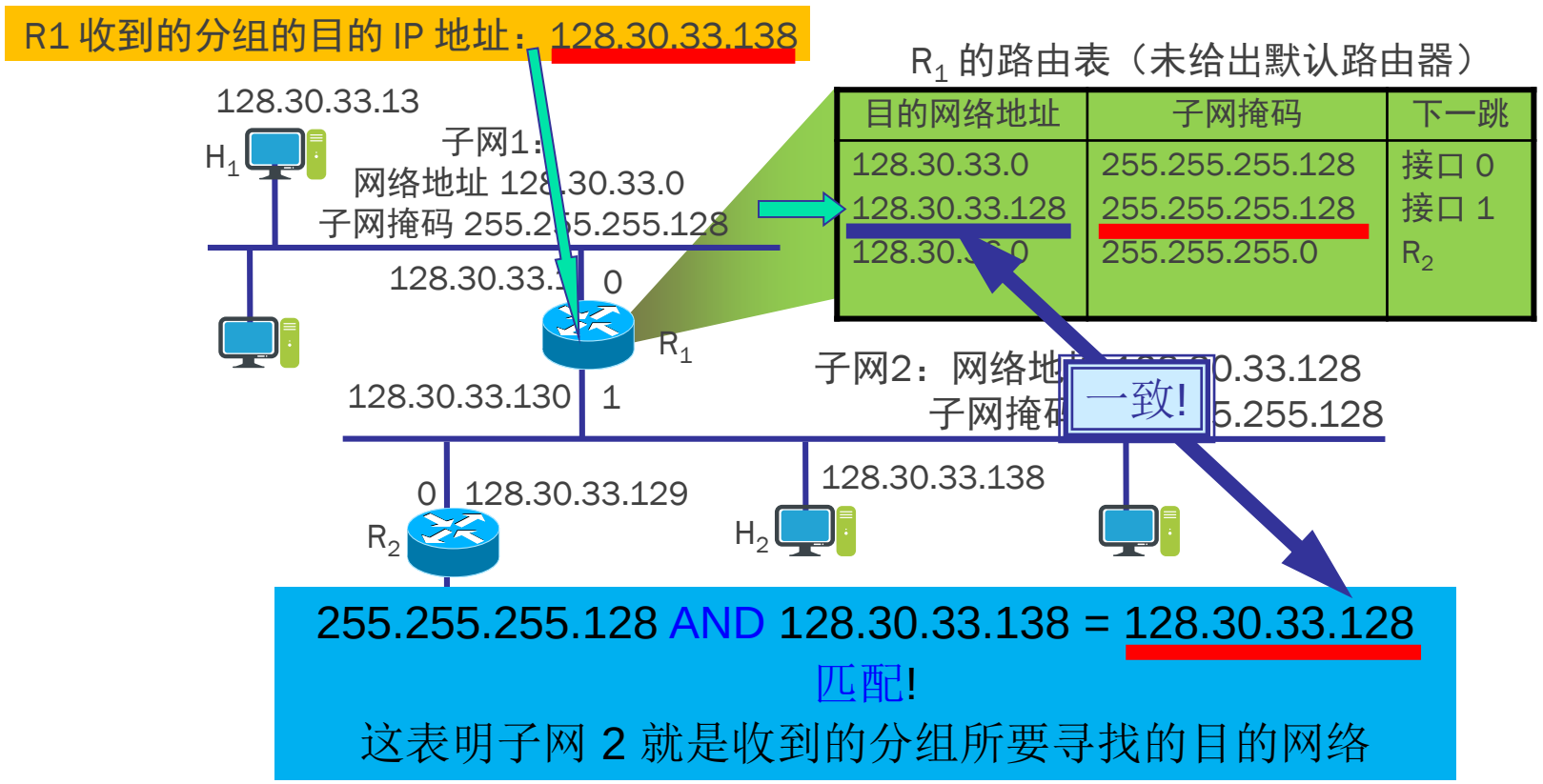
255.255.255.128 AND 128.30.33.138 = 128.30.33.128

不匹配!

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

主机 H_1 要发送分组给 H_2

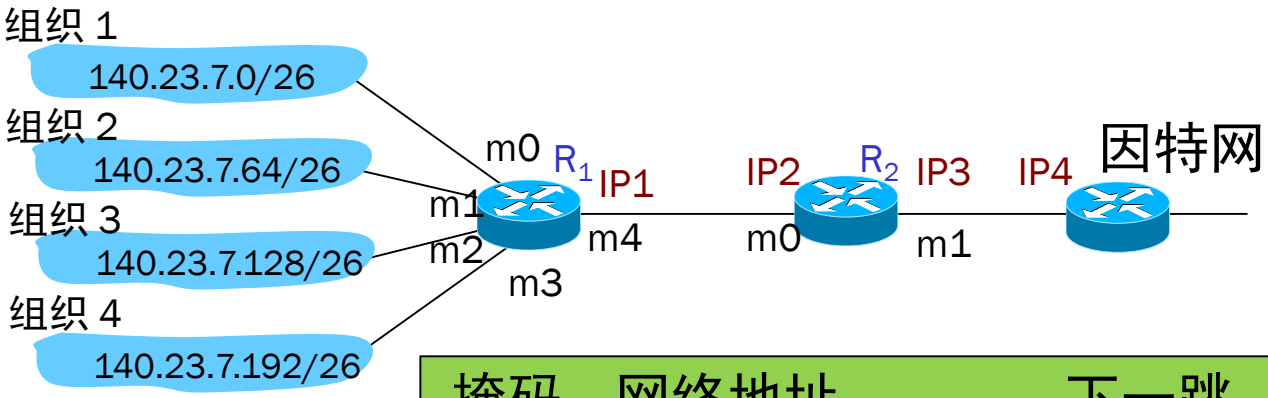
- 路由器 R_1 再用路由表中第 2 个项目的子网掩码和 128.30.33.138 逐比特 AND 操作



2. IP数据报的转发流程

- (1) 从收到的数据报首部提取目的IP地址D。
- (2) 先判断是否为直接交付。对路由器直接相连的网络逐个进行检查，看是否和相应的网络地址匹配。若匹配，则把分组进行直接交付（当然还需要把D转换成物理地址，把数据报封装成帧发送出去），转发任务结束。否则就是间接交付，执行(3)。
- (3) 对路由表中的每一行（目的网络地址，掩码，下一跳，接口），用其中的掩码和D逐位相“与”（AND操作），其结果为N。若N与该行的网络地址匹配，则把数据报传送给该行指明的下一跳路由器；否则，执行(4)。
- (4) 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(5)。
- (5) 报告转发数据报出错。

3. 路由聚合

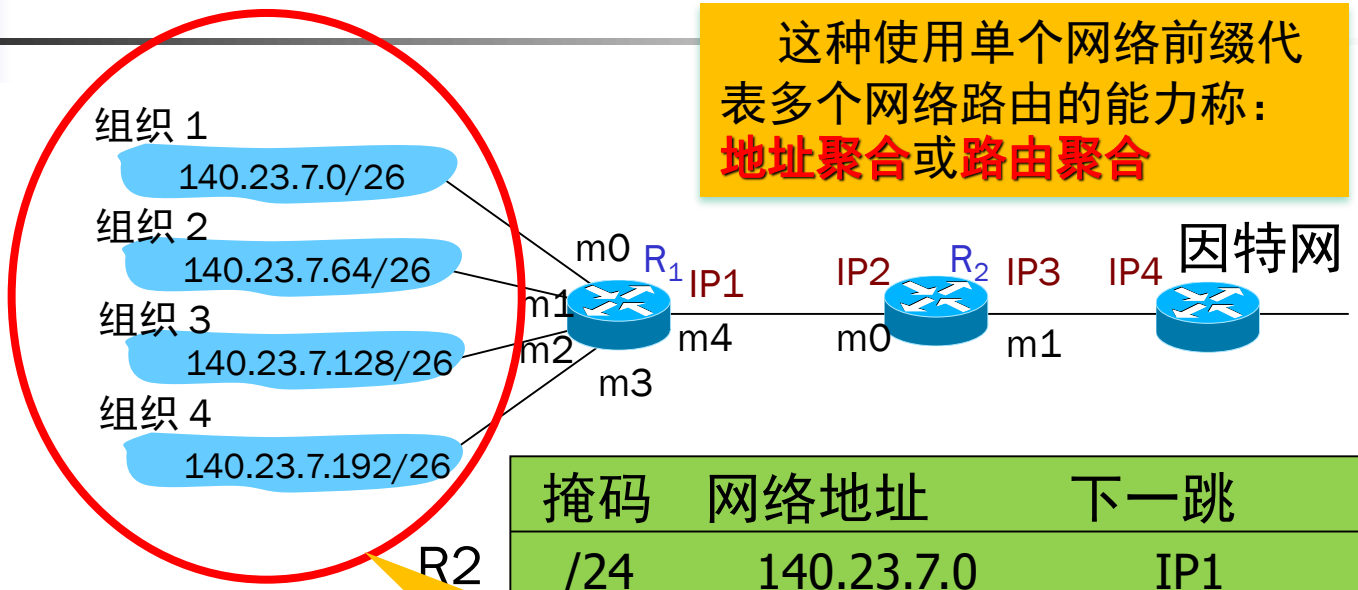


R1 路由表

掩码	网络地址	下一跳	接口
/26	140.23.7.0	--	m0
/26	140.23.7.64	--	m1
/26	140.23.7.128	--	m2
/26	140.23.7.192	--	m3
/0	0.0.0.0	IP2	m4

路由聚合举例

这种使用单个网络前缀代表多个网络路由的能力称：**地址聚合**或**路由聚合**



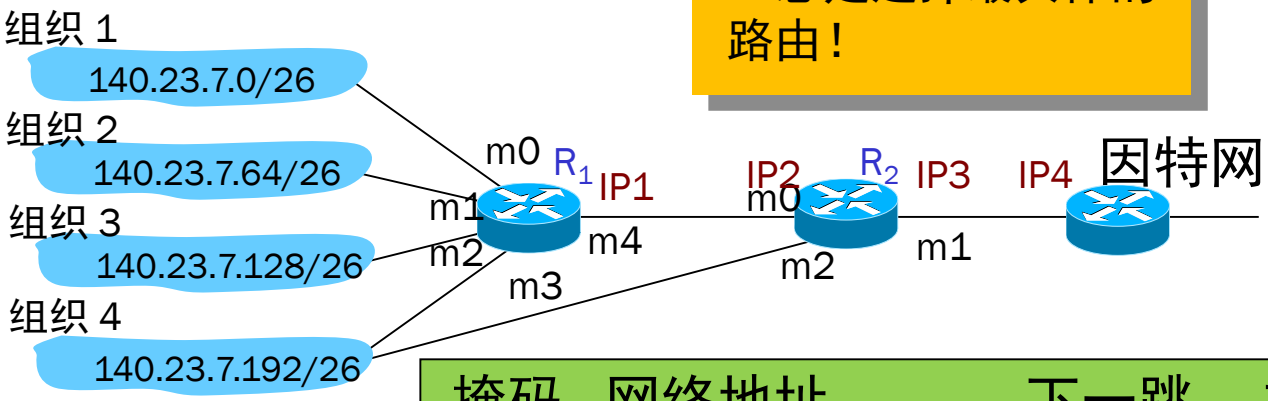
R2 感觉像一个网络

表

掩码	网络地址	下一跳	接口
/24	140.23.7.0	IP1	m0
	0.0.0.0	IP4	m1

路由聚合举例

总是选择最具体的路由！



CIDR使用最长前缀匹配！

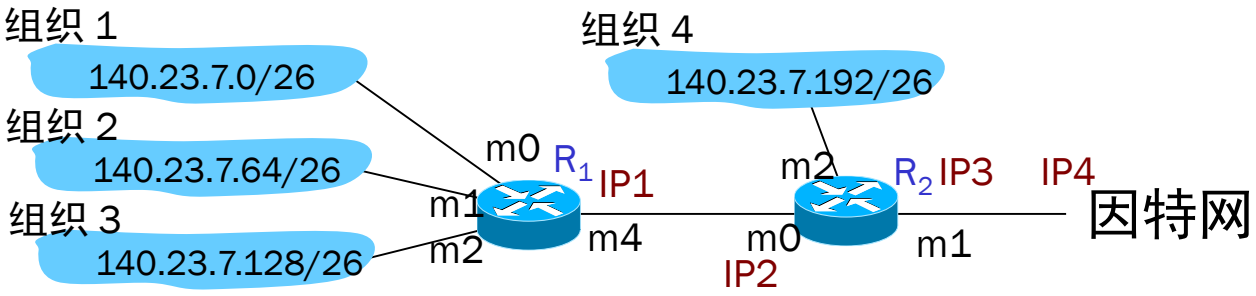
R2路由表

掩码	网络地址	下一跳	接口
/24	140.23.7.0	IP1	m0
/26	140.23.7.192	--	m2
/0	0.0.0.0	IP4	m1

4. 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**。

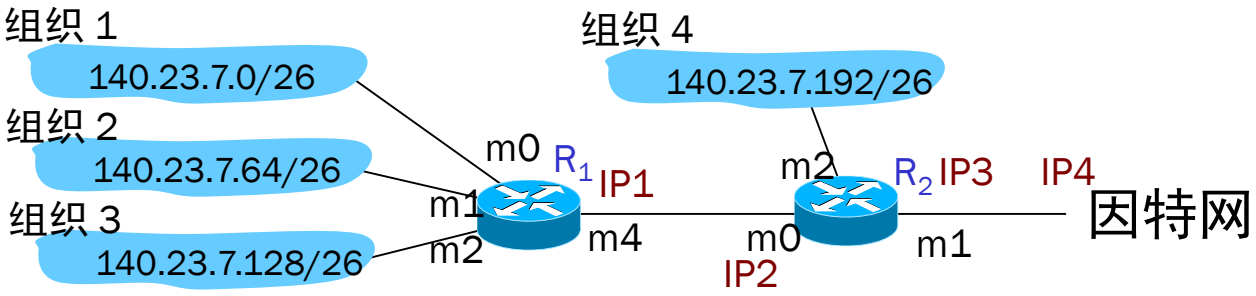
路由聚合举例



R1 路由表

掩码	网络地址	下一跳	接口
/26	140.23.7.0	--	m0
/26	140.23.7.64	--	m1
/26	140.23.7.128	--	m2
/0	0.0.0.0	IP2	m4

路由聚合举例



CIDR使用最长前缀匹配！

R2路由表

掩码	网络地址	下一跳	接口
/24	140.23.7.0	IP1	m0
/26	140.23.7.192	--	m2
/0	0.0.0.0	IP4	m1

特定主机路由

- 这种路由是为特定的目的主机指明一个路由：路由表中前缀为“特定主机IP地址/32”的表项
- 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。

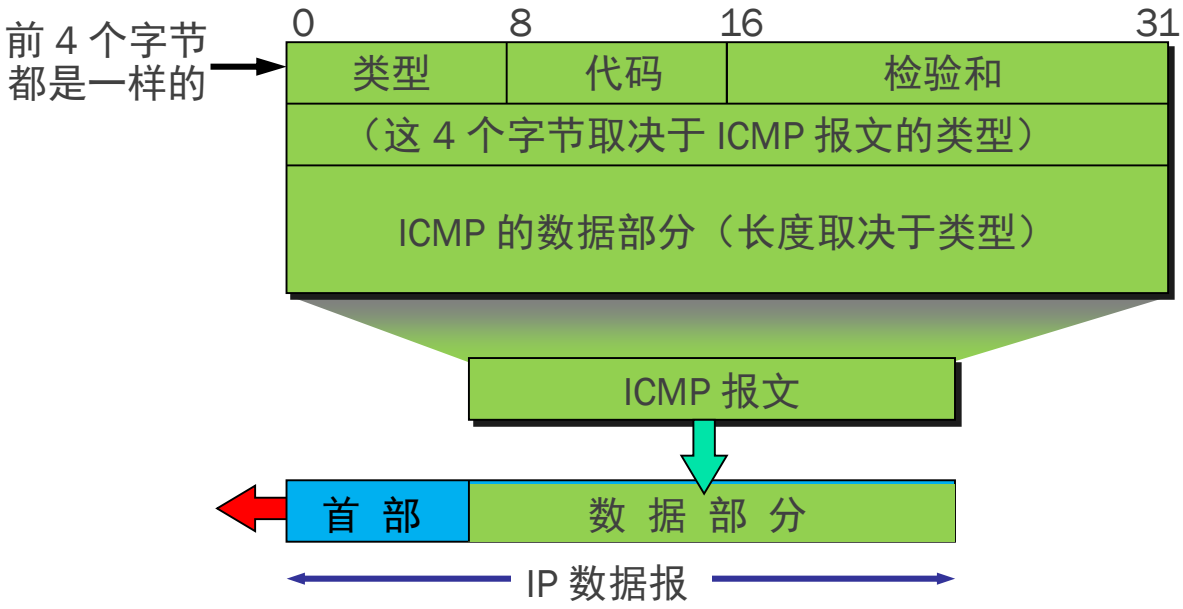
CIDR 最主要的特点

- 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使用不定长的“网络前缀”(network-prefix)来代替分类地址中的网络号和子网号。
- CIDR虽然形式上是二级编址，但实际上可实现多级编址，大的地址块还可以划分为更小的地址块进行分配。
- 等级结构的CIDR地址块分配便于实现路由聚合。

4.3 网际控制报文协议 ICMP

- 为了提高 IP 数据报交付成功的机会，在网际层使用了网际控制报文协议 ICMP (Internet Control Message Protocol)。
- ICMP 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。
- ICMP 不是高层协议，而是 IP 层的协议。
- ICMP 报文作为 IP 层数据报的数据，加上数据报的首部，组成 IP 数据报发送出去。

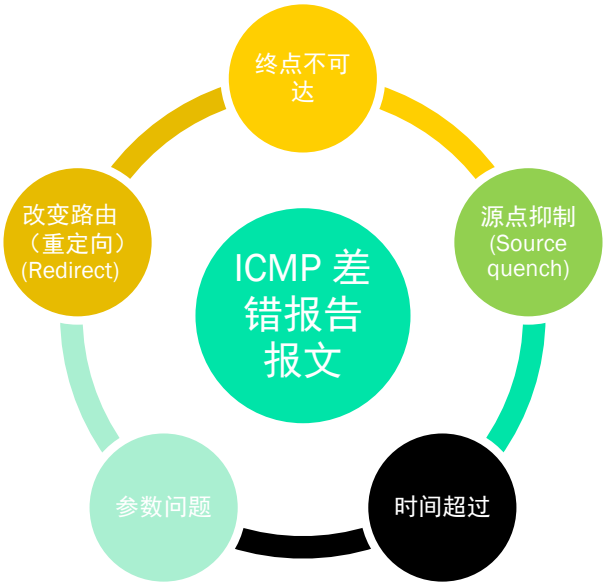
ICMP 报文的格式


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

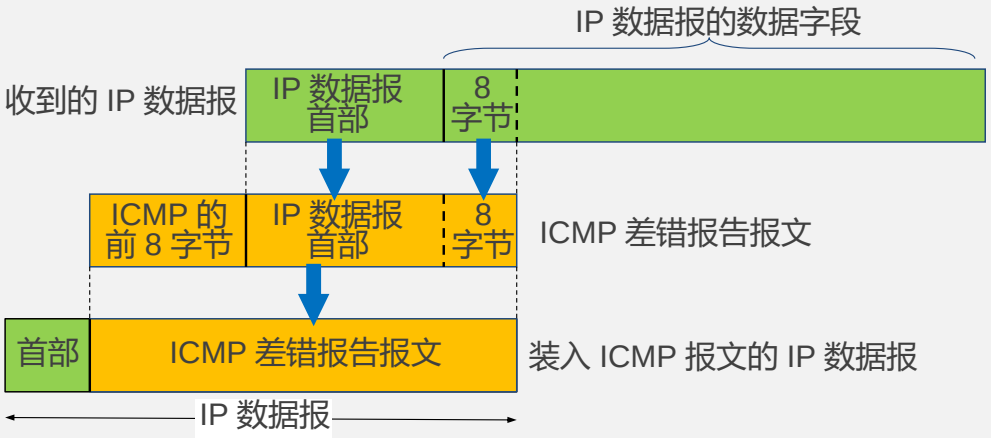
4.4.1 ICMP 报文的种类

- 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 5 种



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

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
```

Traceroute 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.
```

4.4.1 有关路由选择协议的几个基本概念

1. 理想的路由算法

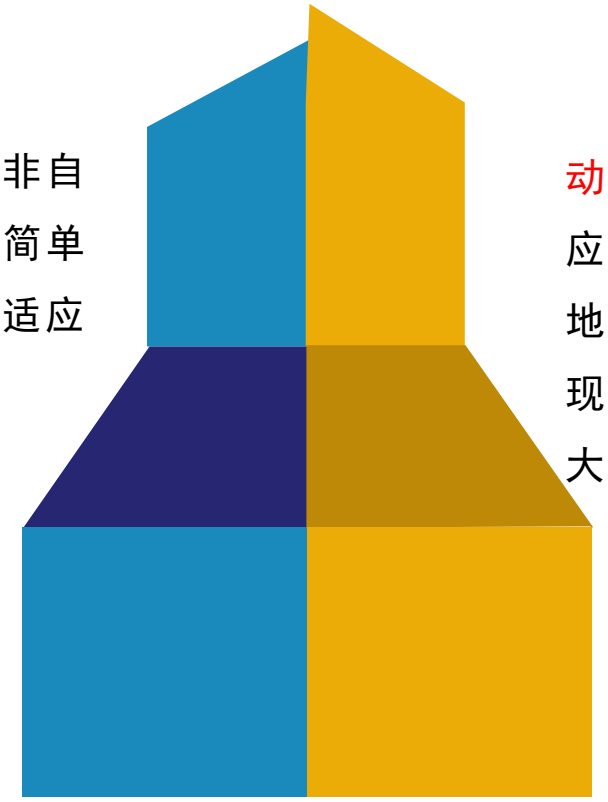
- 算法必须是正确的和完整的。
- 算法在计算上应简单。
- 算法应能适应通信量和网络拓扑的变化，这就是说，要有自适应性。
- 算法应具有稳定性。
- 算法应是公平的。
- 算法应是最佳的。

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

- 因特网采用分层次的路由选择协议。
- 因特网的规模非常大。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因特网的通信链路饱和。
- 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因特网上。

自治系统 AS(Autonomous System)

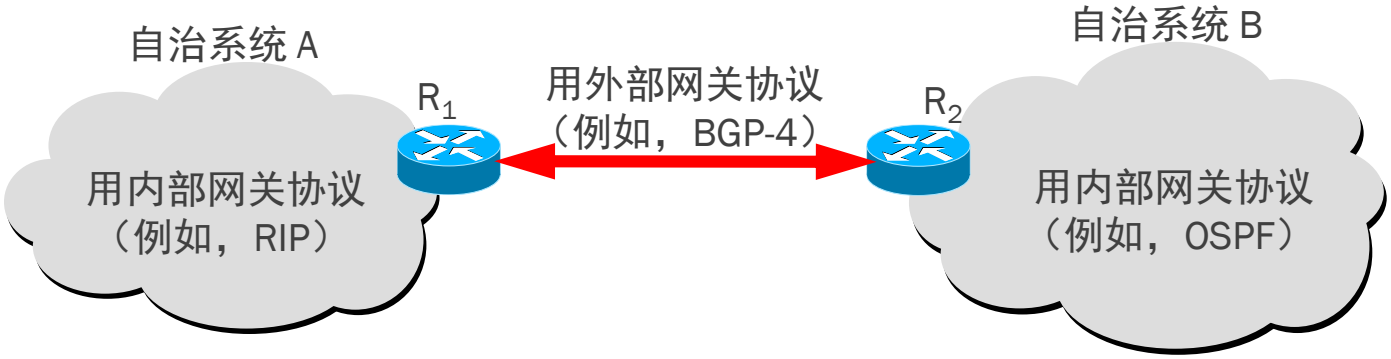
- 自治系统 AS 的定义：在单一的技术管理下的一组路由器，而这些路由器使用一种 AS 内部的路由选择协议和共同的度量以确定分组在该 AS 内的路由，同时还使用一种 AS 之间的路由选择协议用以确定分组在 AS之间的路由。
- 现在对自治系统 AS 的定义是强调下面的事实：尽管一个 AS 使用了多种内部路由选择协议和度量，但重要的是一个 AS 对其他 AS 表现出的是一个**单一的**和**一致的路由选择策略**。

因特网有两大类路由选择协议

内部网关协议 IGP (Interior Gateway Protocol) 即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做
域间路由选择(interdomain routing)，
在自治系统内部的路由选择叫做
域内路由选择(intradomain routing)

4.4.2 内部网关协议 RIP(Routing Information Protocol)

1. 工作原理

- 路由信息协议 RIP 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。

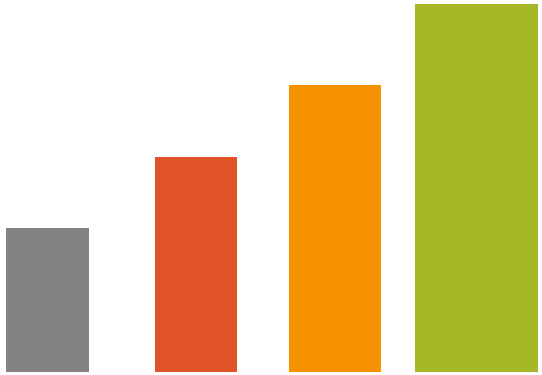
“距离”的定义

- 路由器到直接连接的网络的距离定义为 1。
- 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。
- 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 协议的三个要点

- 仅和**相邻路由器**交换信息。
- 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。
- 按固定的时间间隔**交换路由信息**，例如，每隔 30 秒。
  - 为加快协议的收敛速度，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（即**触发更新**）。

目的	下一跳	距离
网1	R1	5
网2	R2	1
网3	R3	6
…		



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）。
- 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。
- 一般情况下RIP 协议的**收敛**过程较快，即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。

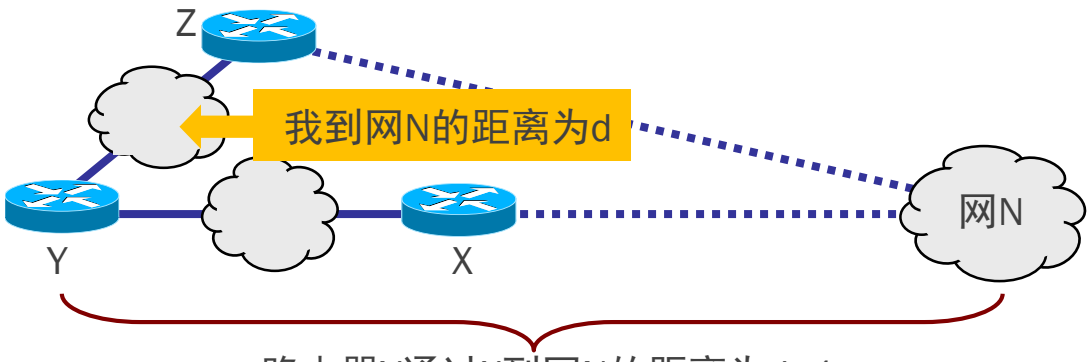
RIP协议路由表的更新

路由器Y通过X到网N的距离为d+1

Y的路由表：

目的	下一站	距离
N	X	d+1

RIP协议路由表的更新

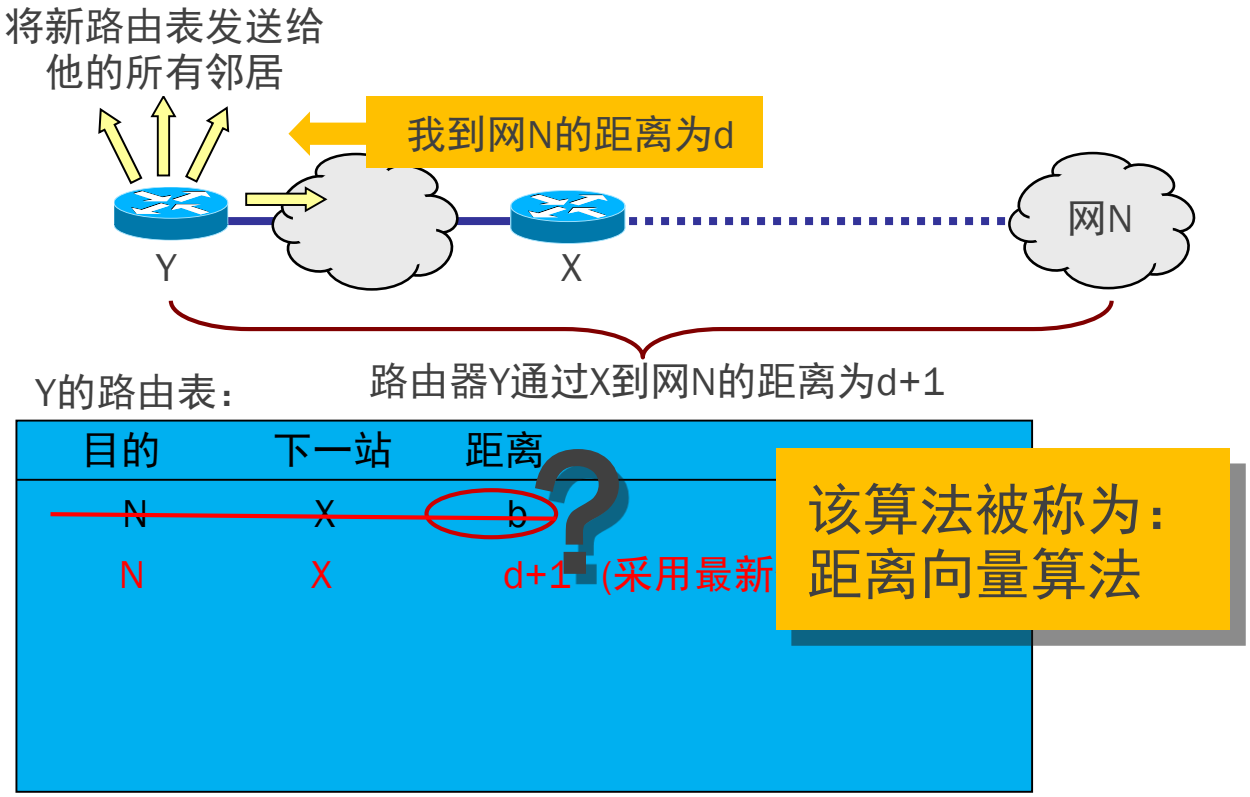


路由器Y通过X到网N的距离为d+1

Y的路由表：

目的	下一站	距离
~~N~~	~~Z~~	b
N	X	d+1 (若d+1<b)

RIP协议路由表的更新



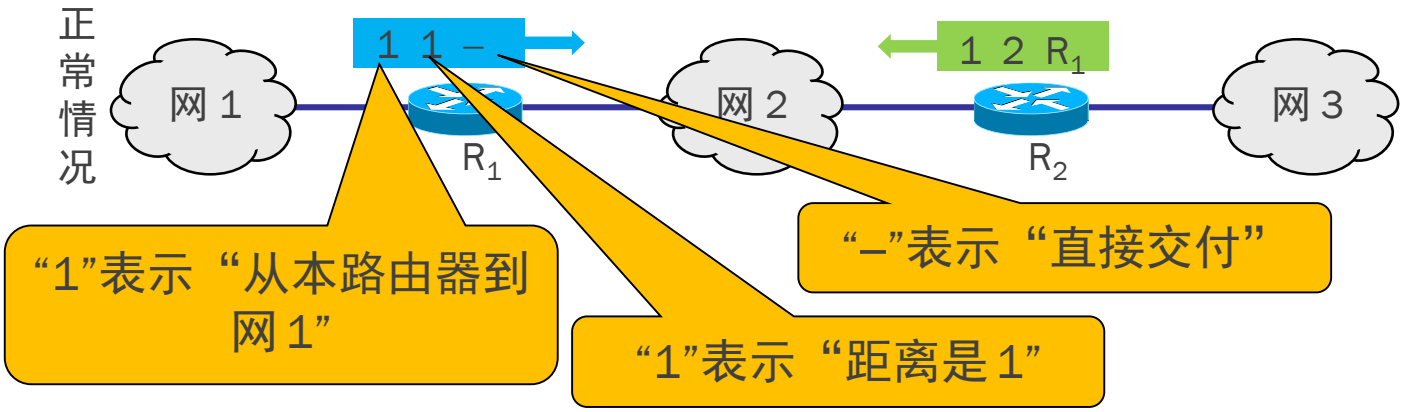
2. RIP的距离向量算法

- 收到相邻路由器（其地址为 X）的一个路由更新报文：
  - (1) 先修改此报文中的所有项目：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。每一个项目都有三个关键数据，即：到目的网络N，距离是d，下一跳路由器是X。
  - (2) 若原路由表中没有目的网络N，则把该项目添加到路由表中。
  - 否则，查看路由表中目的网络为N的表项，若其下一跳是X，则把收到的项目替换原项目。
  - 否则，若收到的项目中的距离d小于路由表中的距离，则进行更新，
  - 否则什么也不做。
  - (3) 若180秒（默认）没有收到某条路由项目的更新报文，则把该路由项目记为无效，即把距离置为16（距离为16表示不可达），若再过一段时间，如120秒，还没有收到该路由项目的更新报文，则将该路由项目从路由表中删除。
  - (4) 若路由表发生变化，向所有相邻路由器发送路由更新报文。
  - (5) 返回。

路由器之间交换信息

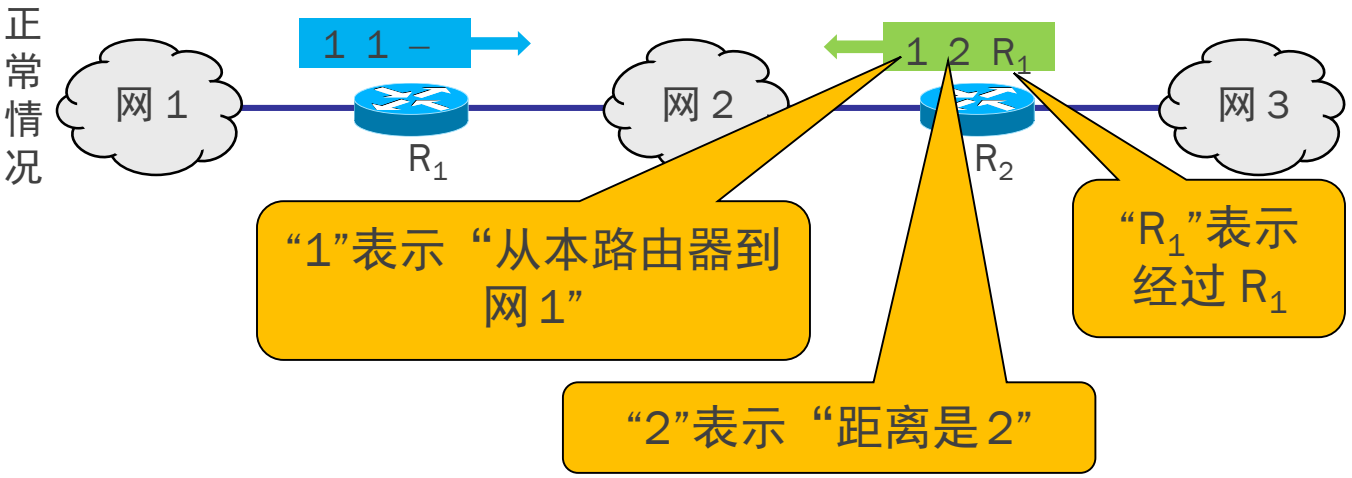
- RIP协议让互联网中的所有路由器都和自己的相邻路由器不断交换路由信息，并不断更新其路由表，使得从每一个路由器到每一个目的网络的路由都是最短的（即跳数最少）。
- 虽然所有的路由器最终都拥有了整个自治系统的全局路由信息，但由于每一个路由器的位置不同，它们的路由表当然也应当是不同的。

路由器之间交换信息



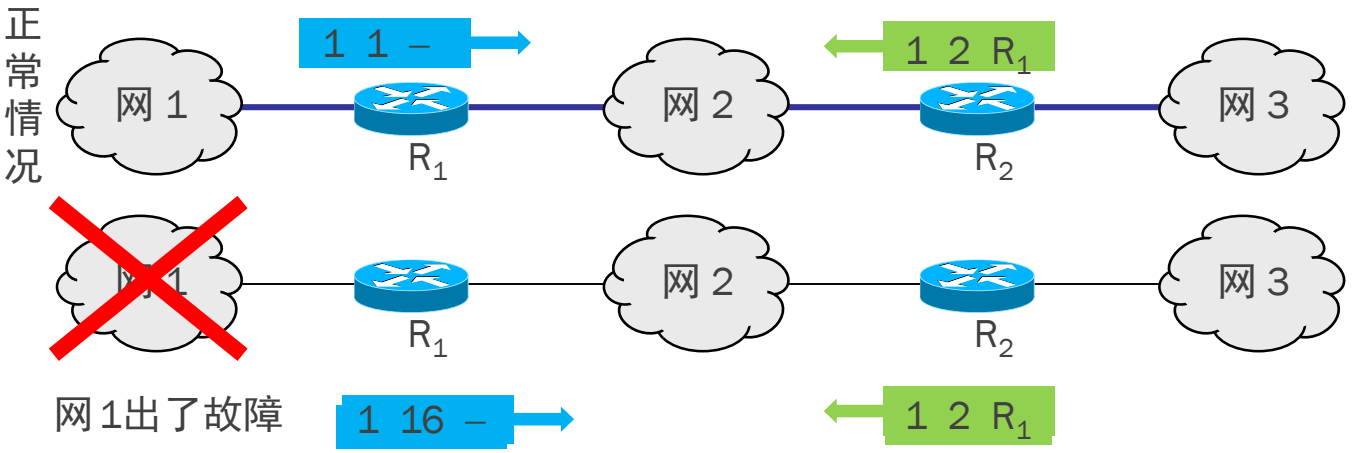
R_1 说：“我到网 1 的距离是 1，是直接交付。”

路由器之间交换信息



R_2 说：“我到网 1 的距离是 2，是经过 R_1。”

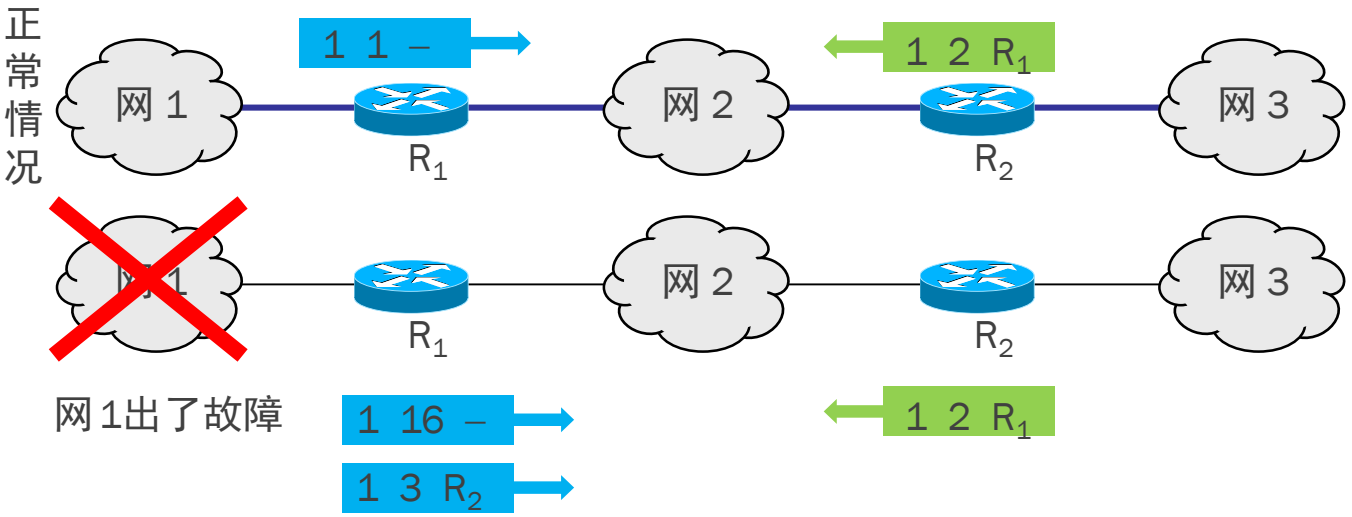
路由器之间交换信息



R_1 说：“我到网 1 的距离是 16（表示无法到达），是直接交付。”

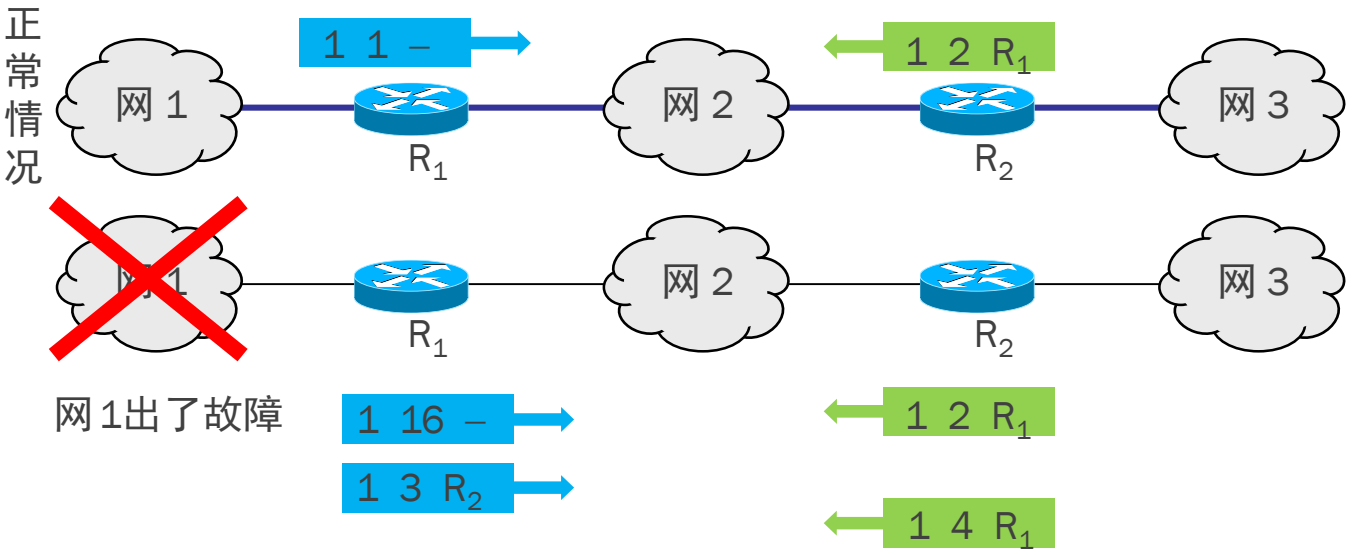
但 R_2 在收到 R_1 的更新报文之前，还发送原来的报文，因为这时 R_2 并不知道 R_1 出了故障。

路由器之间交换信息



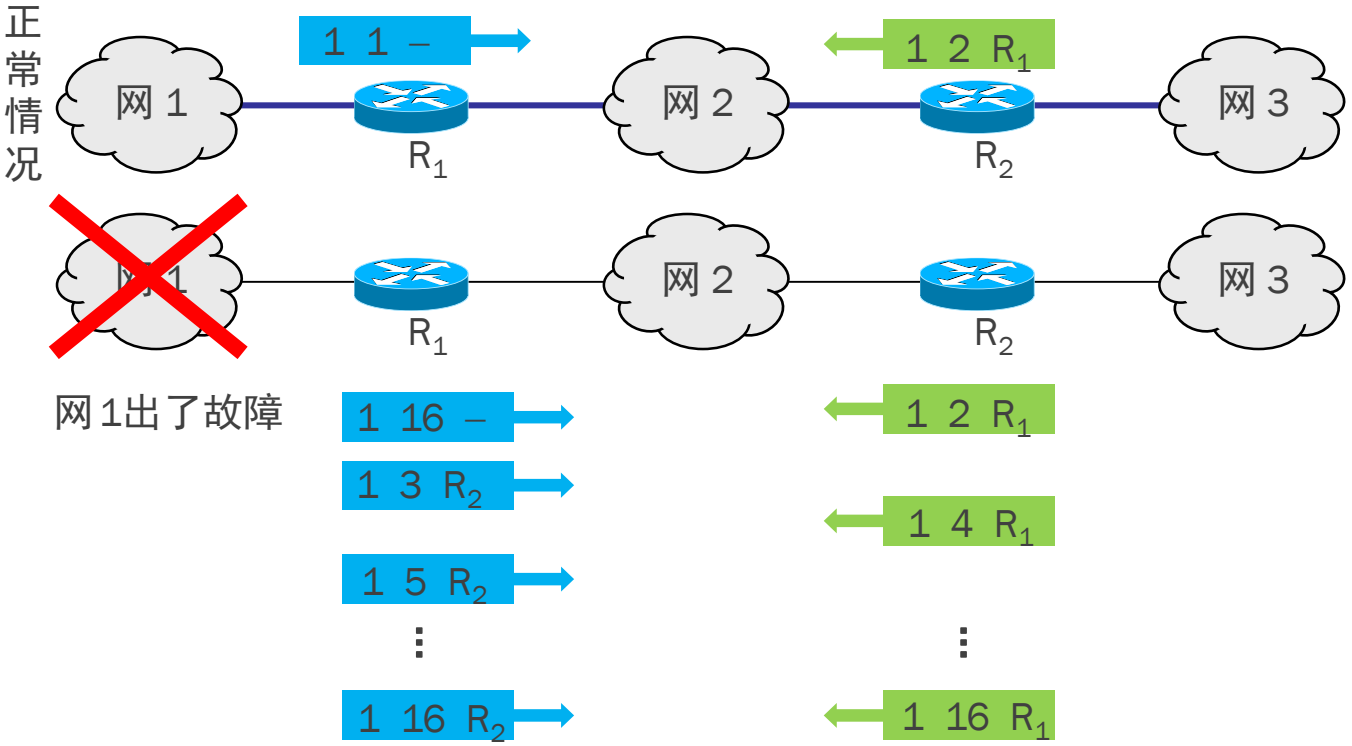
R_1 收到 R_2 的更新报文后，误认为可经过 R_2 到达网1，于是更新自己的路由表，说：“我到网 1 的距离是 3，下一跳经过 R_2”。然后将此更新信息发送给 R_2。

路由器之间交换信息



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”。

路由器之间交换信息



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

- RIP 存在的一个问题是当网络出现故障时，要经过比较长的时间才能将此信息传送到所有的路由器。
- RIP 协议最大的优点就是实现简单，开销较小。
- 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。

4.4.3 内部网关协议 OSPF (Open Shortest Path First)

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
- 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”。
- 是分布式的链路状态协议。

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。

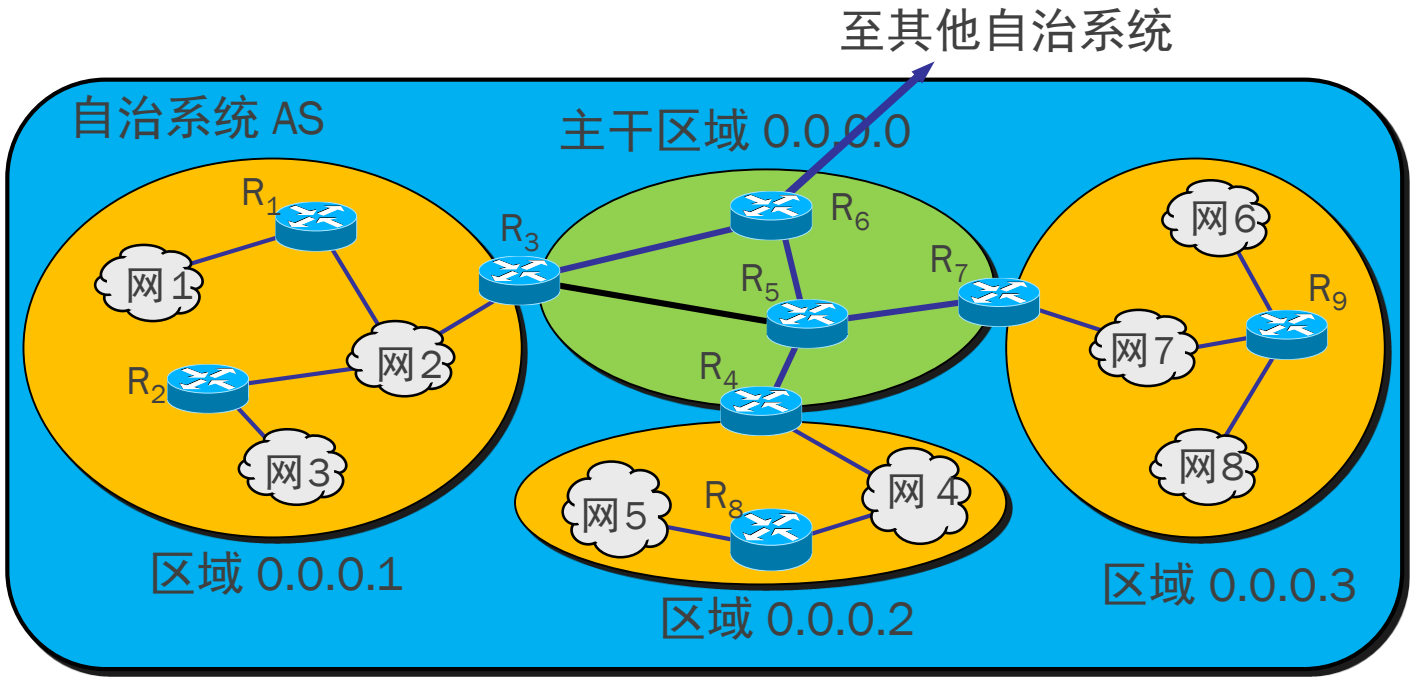
链路状态数据库(link-state database)

- 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。

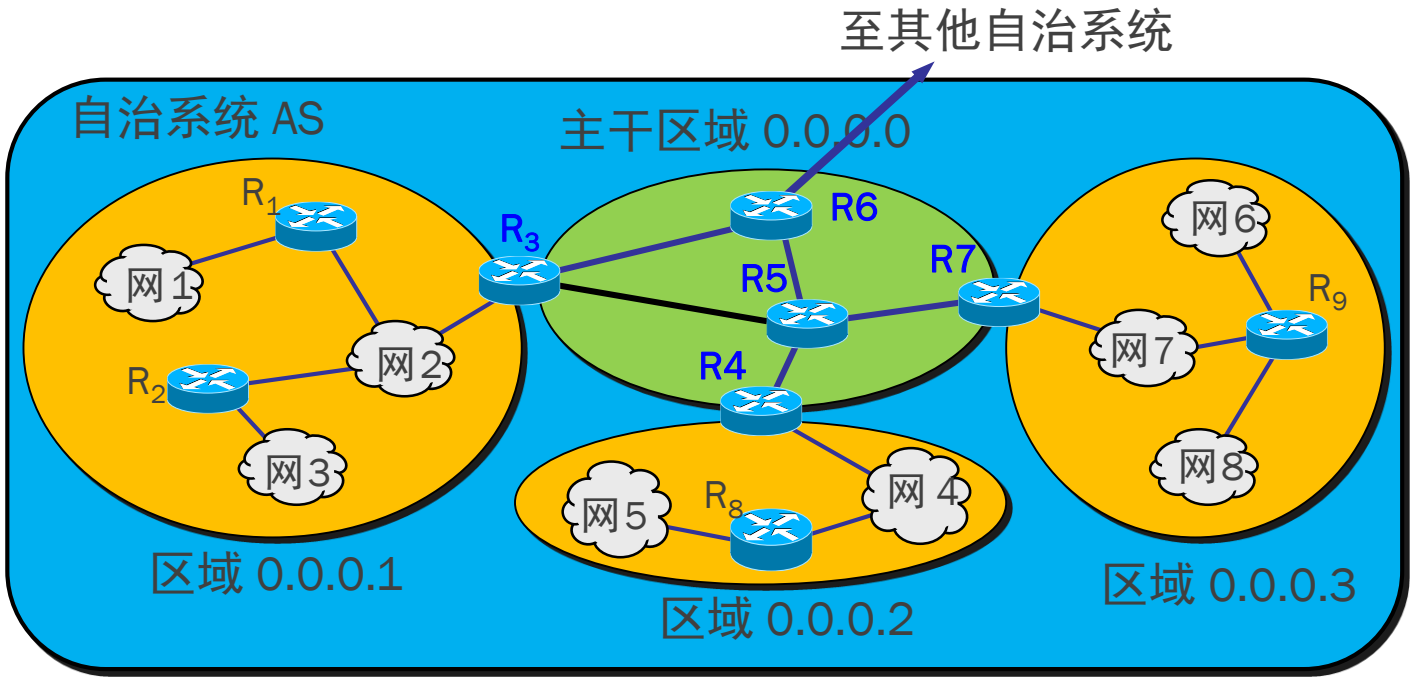
OSPF 划分为两种不同的区域



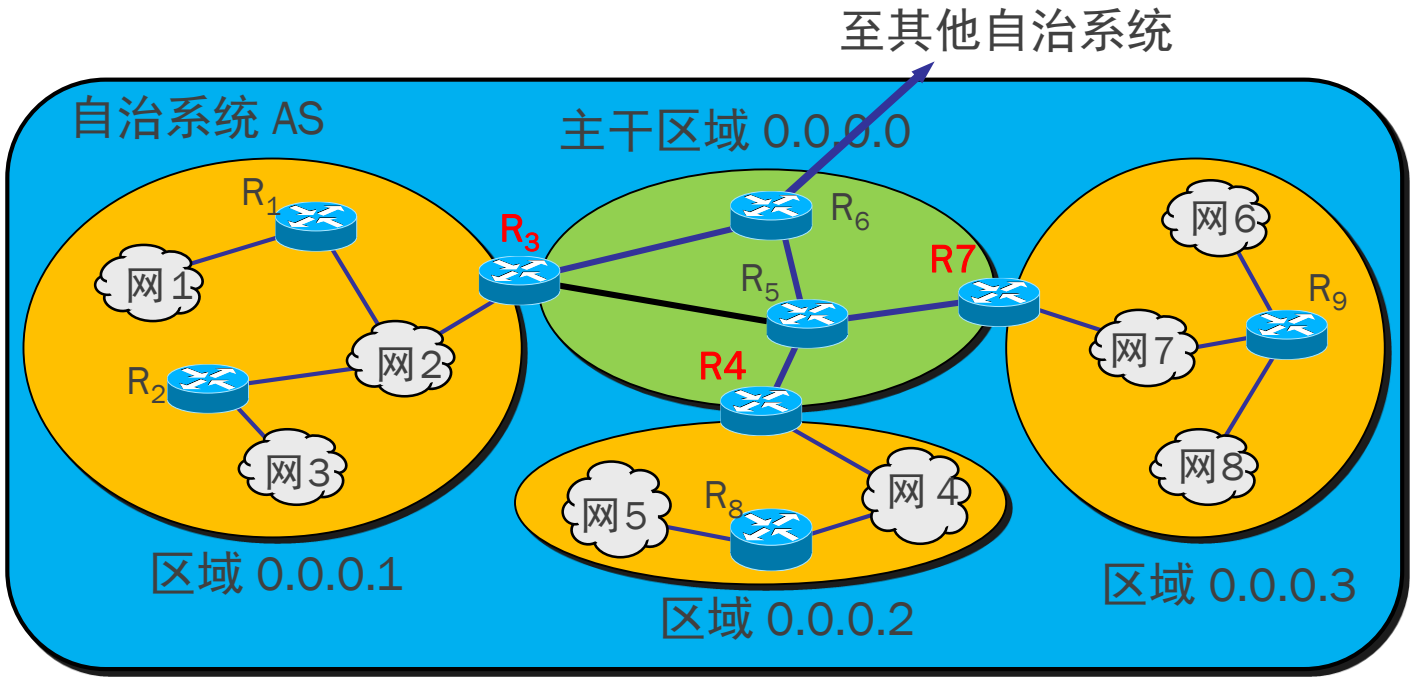
划分区域

- 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
网 1
网 2
R2
网 3
区域 0.0.0.1
R3
R6
R5
R7
R4
网 5
R8
网 4
区域 0.0.0.2
网 6
R9
网 7
网 8
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
网 1
R2
网 2
网 3
区域 0.0.0.1
R3
R6
R5
R7
R4
网 5
R8
网 4
区域 0.0.0.2
网 6
R9
网 7
网 8
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。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。

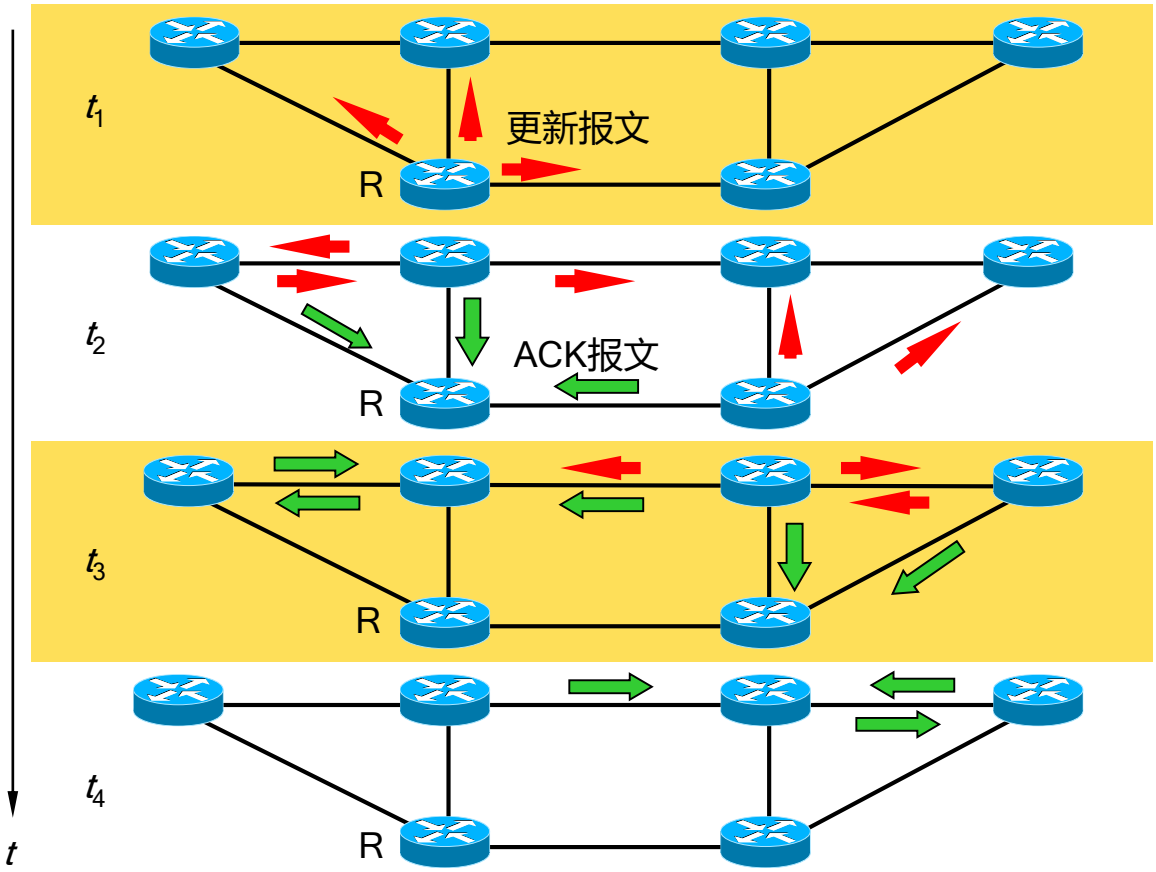
OSPF 的其他特点

- 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。

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 使用的是可靠的洪泛法



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(designated router)

- 多点接入的局域网采用了指定的路由器的方法，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。

4.4.4 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 使用的环境

- 因特网的规模太大，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。

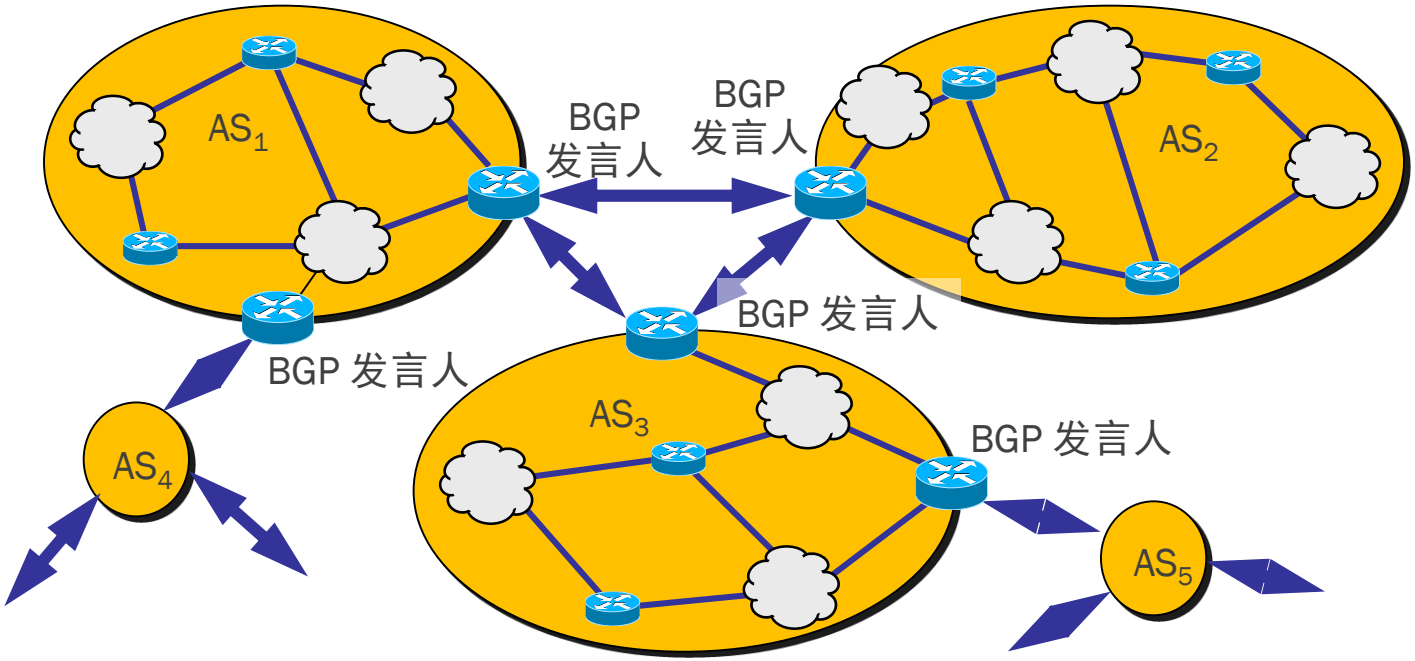
BGP 发言人(BGP speaker)

- 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” 。
- 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

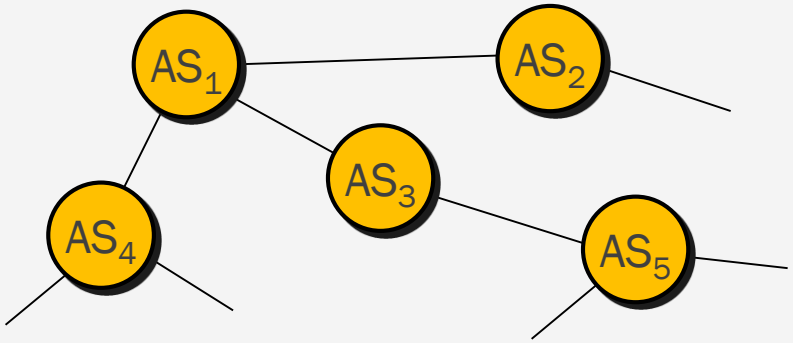
- 一个 BGP 发言人与其他自治系统中的 BGP 发言人要交换路由信息，就要先建立 TCP 连接，然后在此连接上交换 BGP 报文以建立 BGP 会话(session)，利用 BGP 会话交换路由信息。
- 使用 TCP 连接能提供可靠的服务，也简化了路由选择协议。
- 使用 TCP 连接交换路由信息的两个 BGP 发言人，彼此成为对方的邻站或对等站。

BGP 发言人和自治系统 AS 的关系



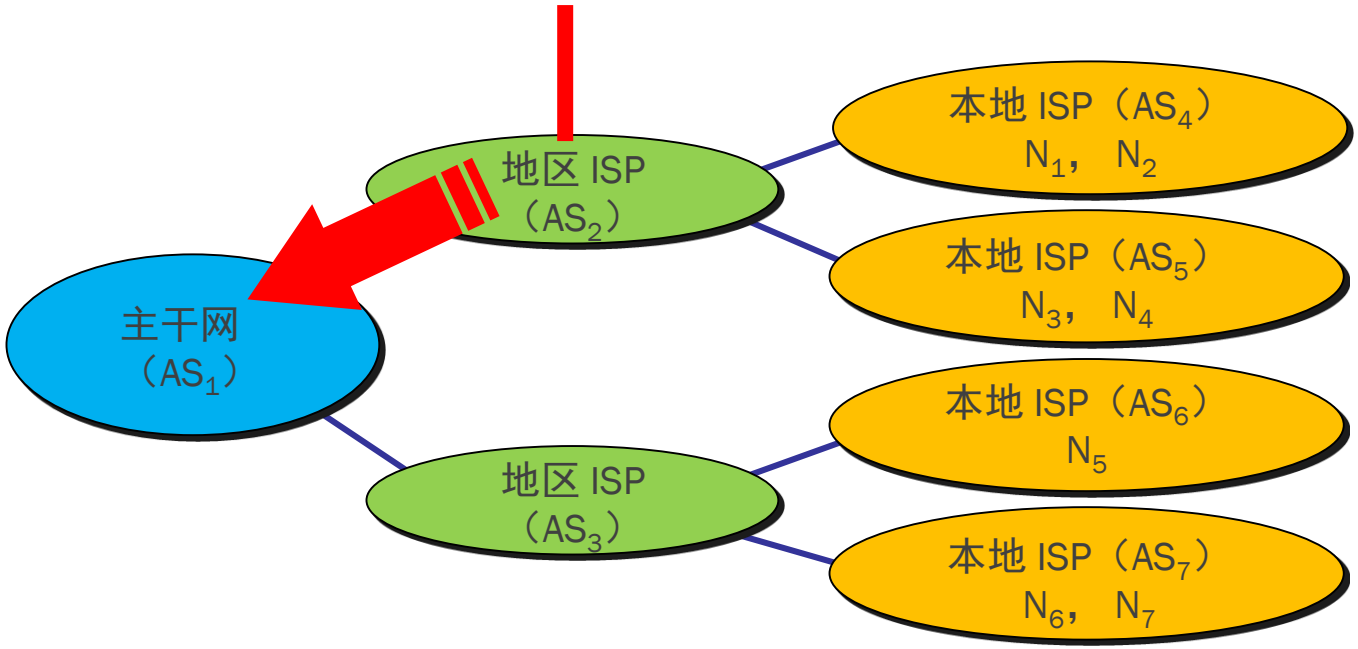
AS 的连通图举例

- 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。



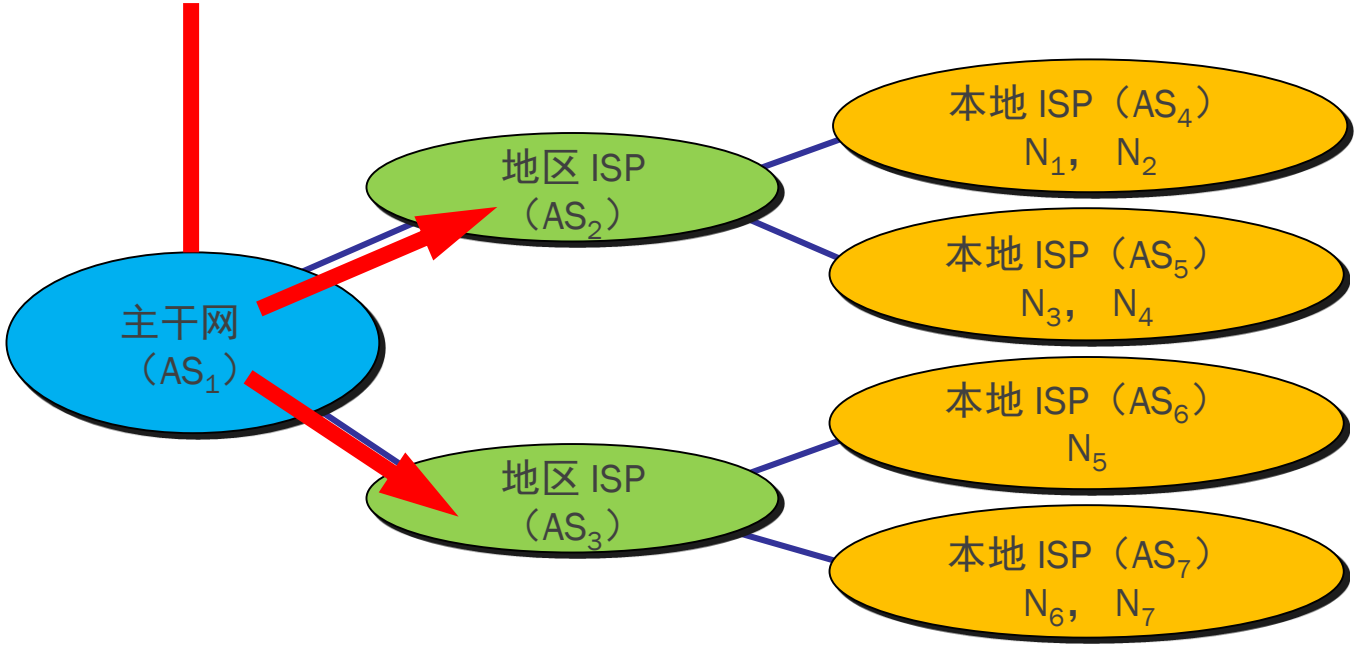
BGP 发言人交换路径向量

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 N_5, N_6 和 N_7 可沿路径（AS_1, AS_3）。”



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方面都有好处。**

BGP-4 共使用四种报文

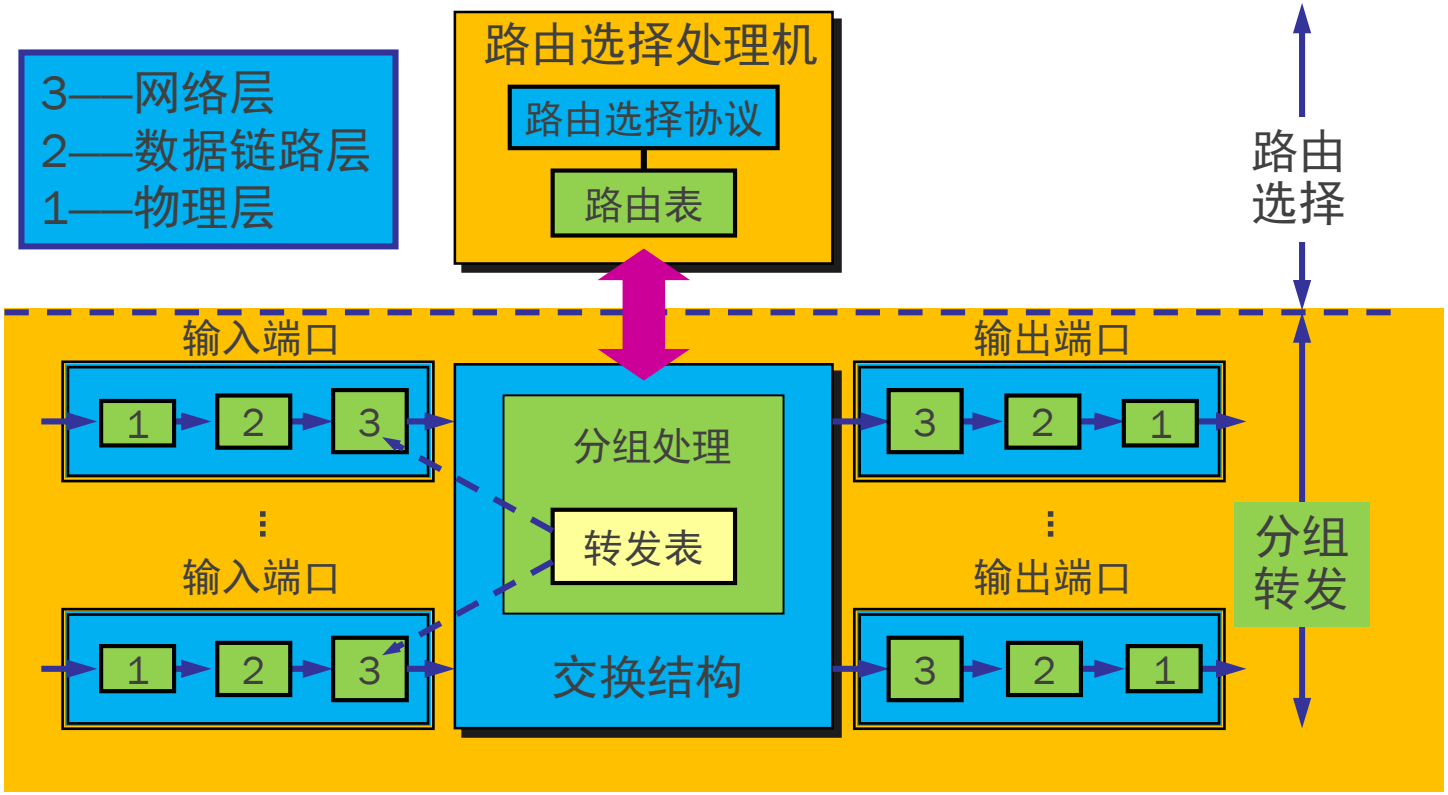
1. 打开(OPEN)报文，用来与相邻的另一个BGP发言人建立关系。
2. 更新(UPDATE)报文，用来发送某一路由的信息，以及列出要撤消的多条路由。
3. 保活(KEEPALIVE)报文，用来确认打开报文和周期性地证实邻站关系。
4. 通知(NOTIFICATION)报文，用来发送检测到的差错。

在 RFC 2918 中增加了 ROUTE-REFRESH 报文，用来请求对等端重新通告。

4.5.1 路由器的构成

- 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。

典型的路由器的结构


3——网络层
2——数据链路层
1——物理层
路由选择处理机
路由选择协议
路由表
路由选择
输入端口
输出端口
1
2
3
分组处理
转发表
3
2
1
输入端口
输出端口
分组转发
1
2
3
交换结构
3
2
1

“转发”和“路由选择”的区别

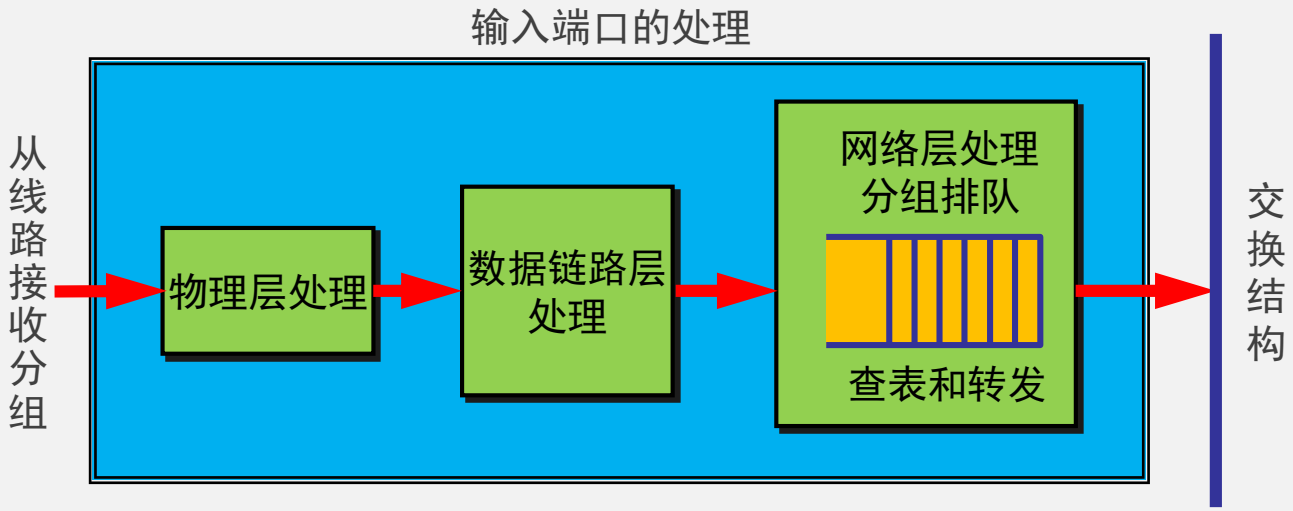
- “**转发**”(forwarding)就是路由器根据转发表将用户的 IP 数据报从合适的端口转发出去。
- “**路由选择**”(routing)则是按照分布式算法，根据从各相邻路由器得到的关于网络拓扑的变化情况，动态地改变所选择的路由。
- 路由表是根据路由选择算法得出的。而转发表是从路由表得出的。
- 在讨论路由选择的原理时，往往不去区分转发表和路由表的区别。

2. 输入端口

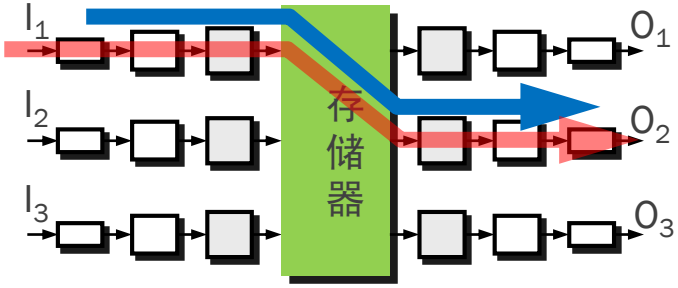
- **数据链路**

若交换结构处理分组的速率赶不上分组进入队列的速率，则会导致输入队列排队！

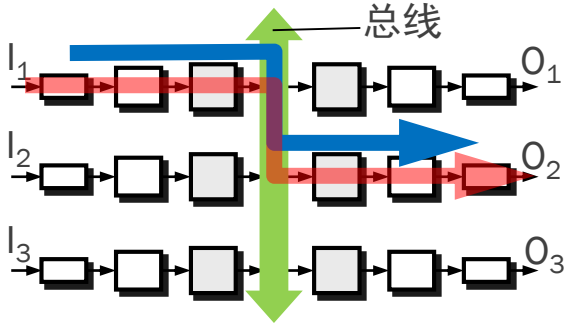
络层的队列中排队等待处理。这会产



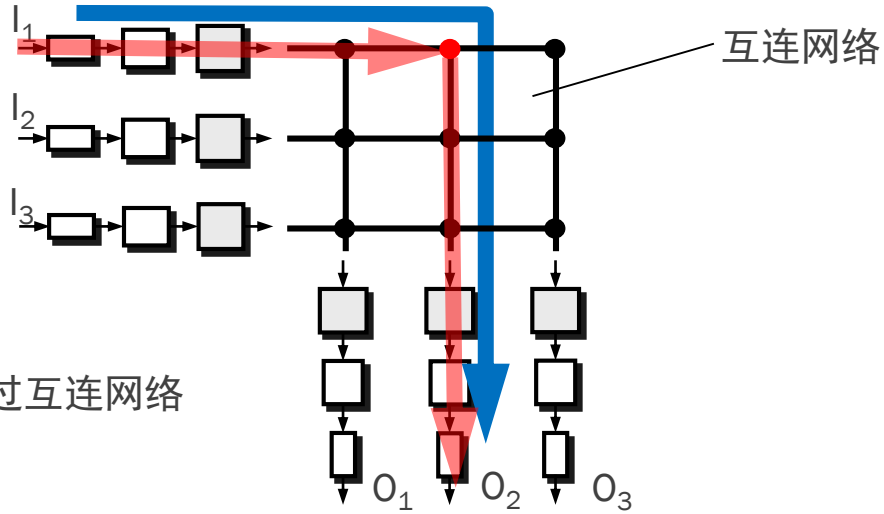
2 交换结构



(a) 通过存储器



(b) 通过总线

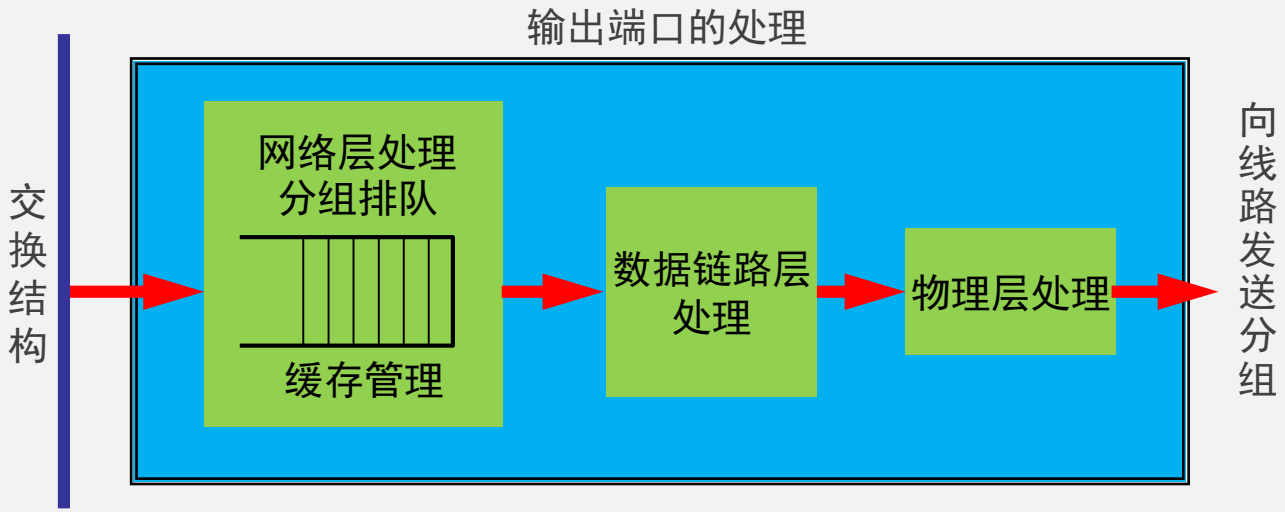


(c) 通过互连网络

3. 输出端口

- **当交换结构传送过来的分组先进行缓存，数据链路层处理模块将分组加上链路层的首部和尾部，交给物理层后发往线路。**

如果路由器足够快是不是就不会出现排队？



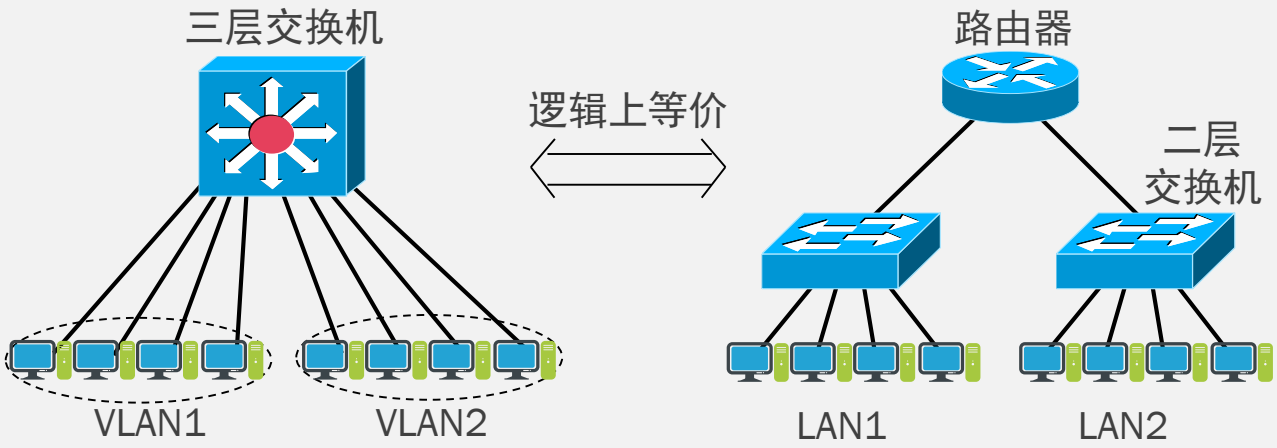
4.5.2 路由器与交换机的比较

- 交换机的优点是即插即用，并具有相对高的分组过滤和转发速度。缺点是：大型交换机网络要求交换机维护大的转发表，主机中维护大的ARP表，并可能产生广播风暴，逻辑拓扑被限制为树。
- 路由器的优点是能提供更加智能的路由选择，并能隔离广播域。缺点是：路由器不是即插即用的，对每个分组处理时间通常比交换机更长。

4.5.3 三层交换机

- “三层交换机”在 VLAN的二层交换机的集成体

为避免混淆，我们使用术语“路由器”而不使用术语“三层交换机”！

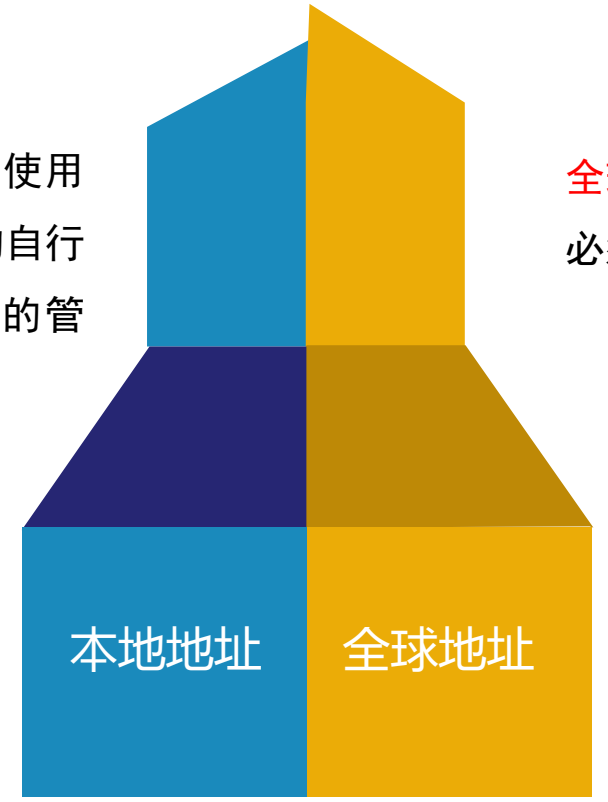


三层交换机的应用

- 处于同一个局域网中的各个子网的互连以及局域网中VLAN间的路由，用三层交换机来代替普通路由器，实现广播域的隔离。
- 只有局域网与广域网互连，或广域网之间互连时才使用普通路由器。

4.6.1 虚拟专用网VPN

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，而不需要向因特网的管理机构申请。

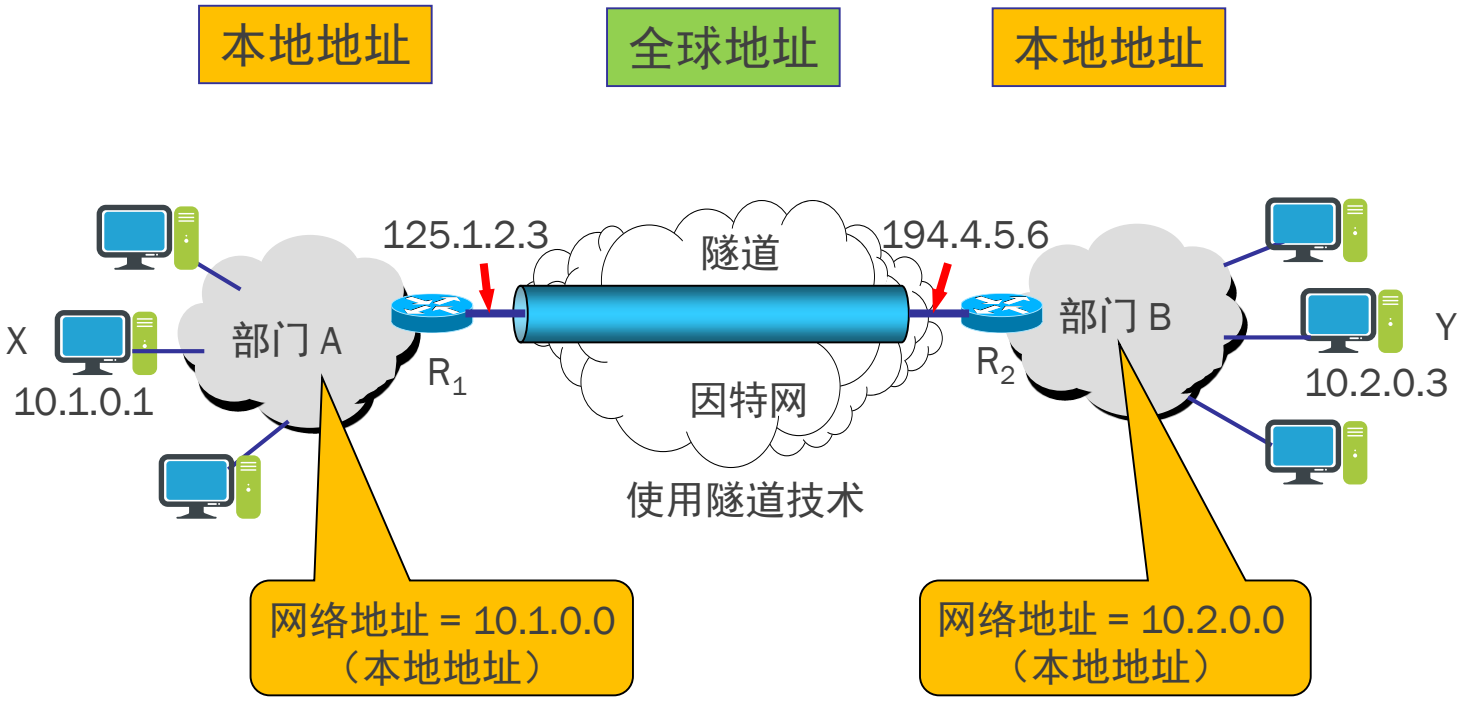


全球地址——全球唯一的IP地址，必须向因特网的管理机构申请。

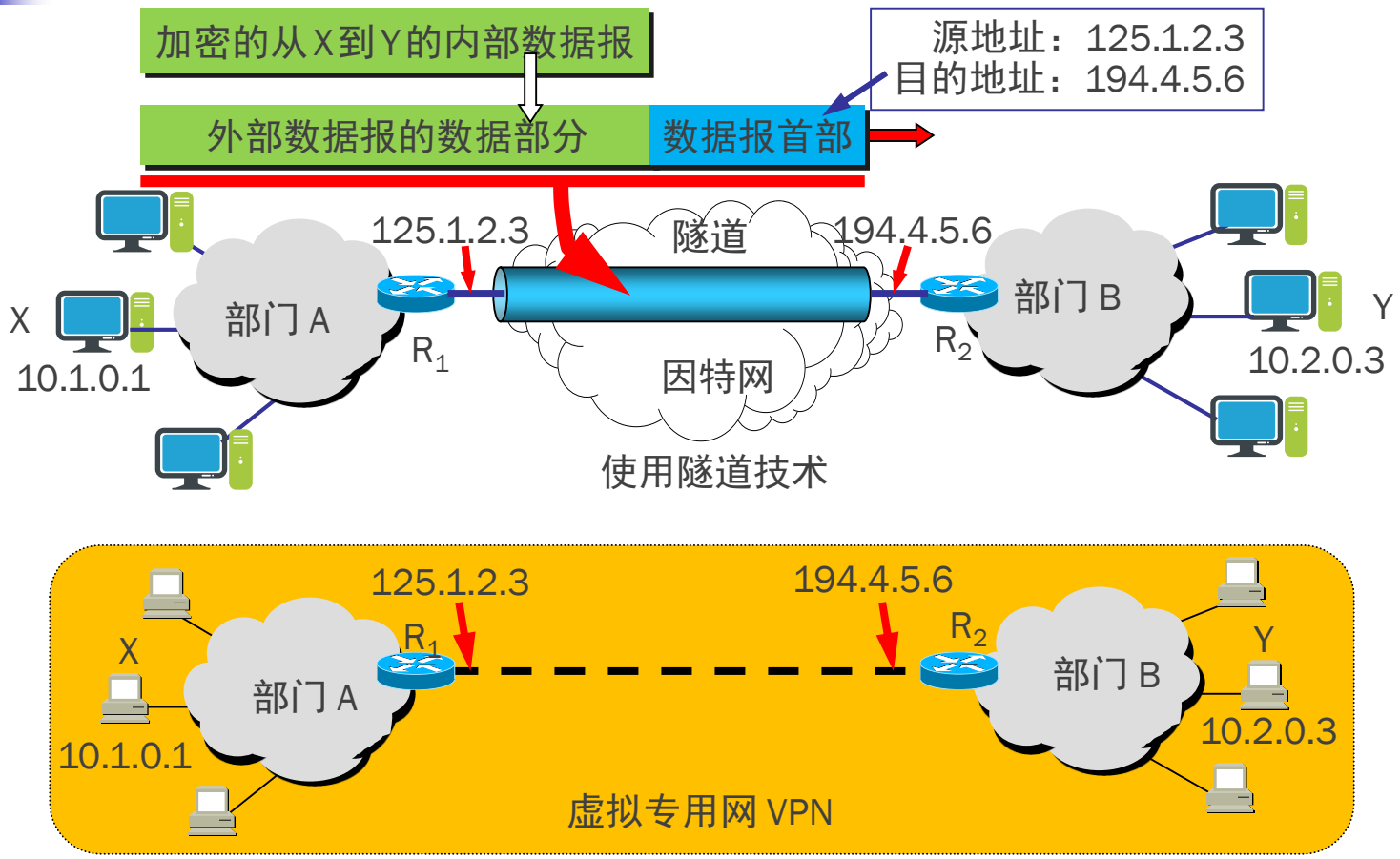
RFC 1918 指明的专用地址(private address)

- 10.0.0.0 到 10.255.255.255
- 172.16.0.0 到 172.31.255.255
- 192.168.0.0 到 192.168.255.255
- 这些地址只能用于一个机构的内部通信，而不能用于和因特网上的主机通信。
- 专用地址只能用作本地地址而不能用作全球地址。在因特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。

用隧道技术实现虚拟专用网



用隧道技术实现虚拟专用网



远程接入VPN(remote access VPN)

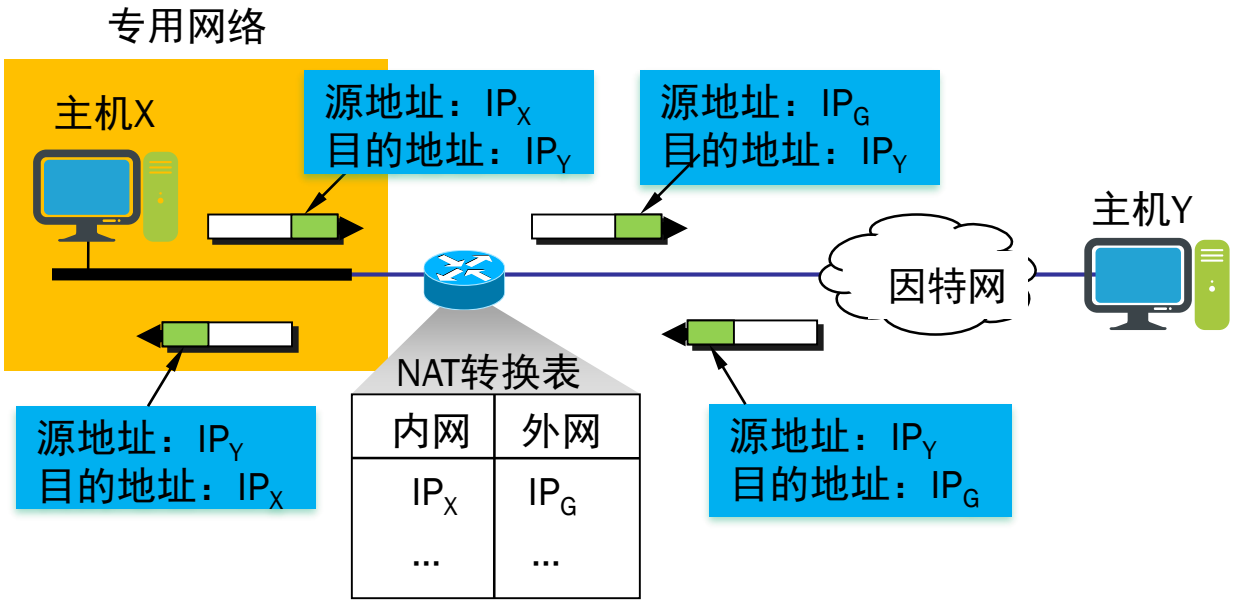
- 有的公司可能没有分布在不同场所的部门，但有很多流动员工在外地工作。公司需要和他们保持联系，远程接入 VPN 可满足这种需求。
- 在外地工作的员工拨号接入因特网，而驻留在员工 PC 机中的 VPN 软件可在员工的 PC 机和公司的主机之间建立 VPN 隧道，因而外地员工与公司通信的内容是保密的，员工们感到好像就是使用公司内部的本地网络。

4.7.2 网络地址转换 NAT (Network Address Translation)

- 网络地址转换 NAT 方法于1994年提出。
- 需要在专用网连接到因特网的路由器上安装 NAT 软件。装有 NAT 软件的路由器叫做 NAT路由器，它至少有一个有效的外部全球地址 IP_G。
- 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 IP_G 才能和因特网连接。

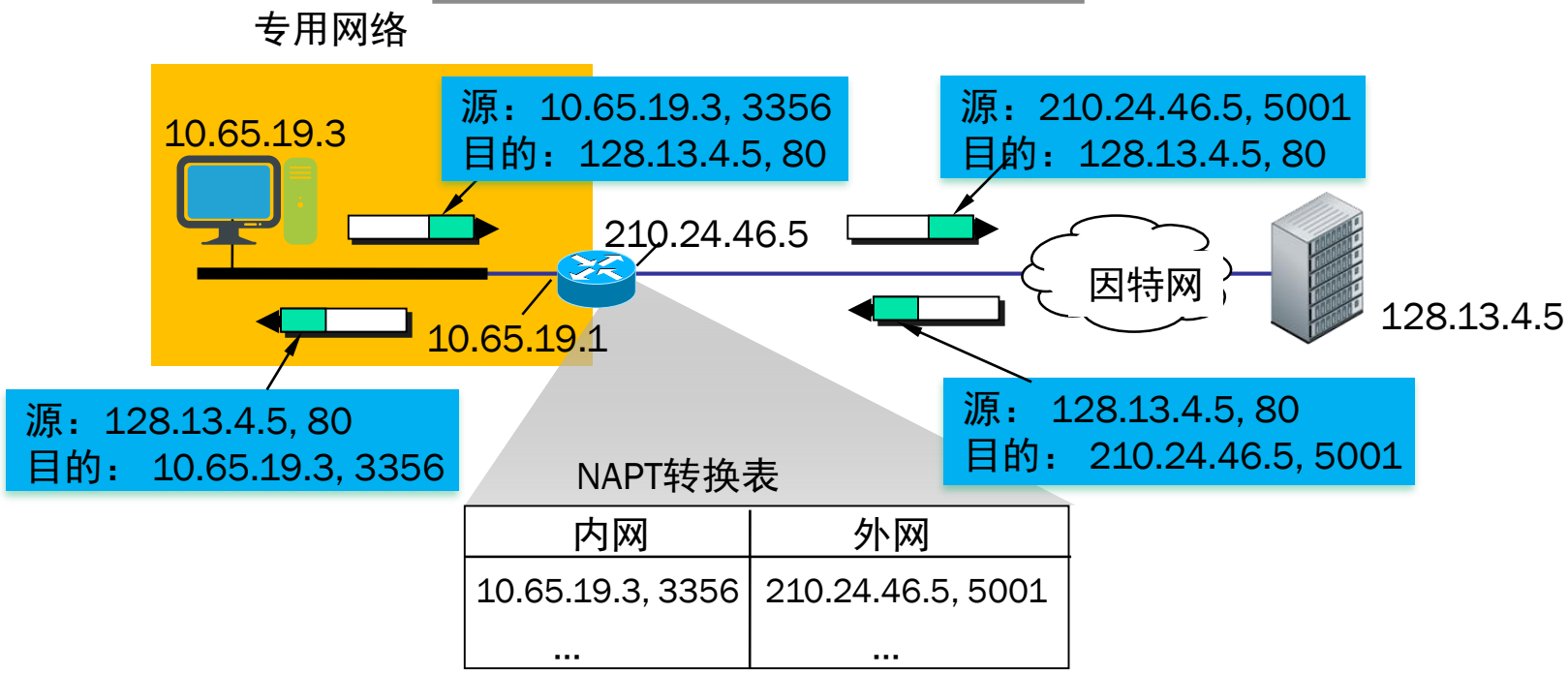
NAT的基本方法

为支持更多主机同时访问外网，可利用报文中的其它字段来区别使用同一外部地址多个内部主机！如：协议字段、目的地址，甚至运输层的端口号！

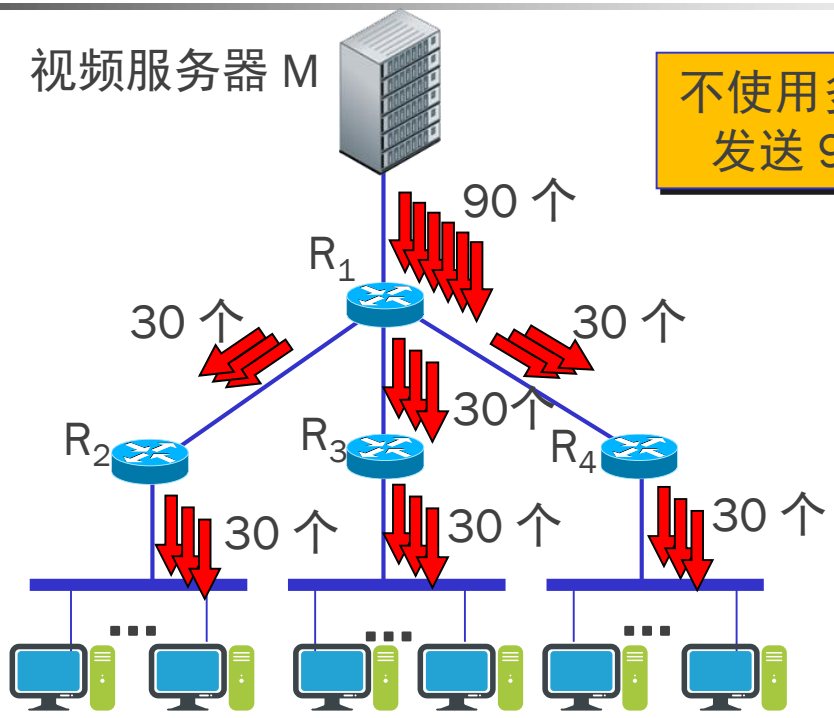


网络地址与端口号转换

由于端口号字段有16比特，因此一个外部IP地址可支持60000多对内部主机与外部主机的通信。



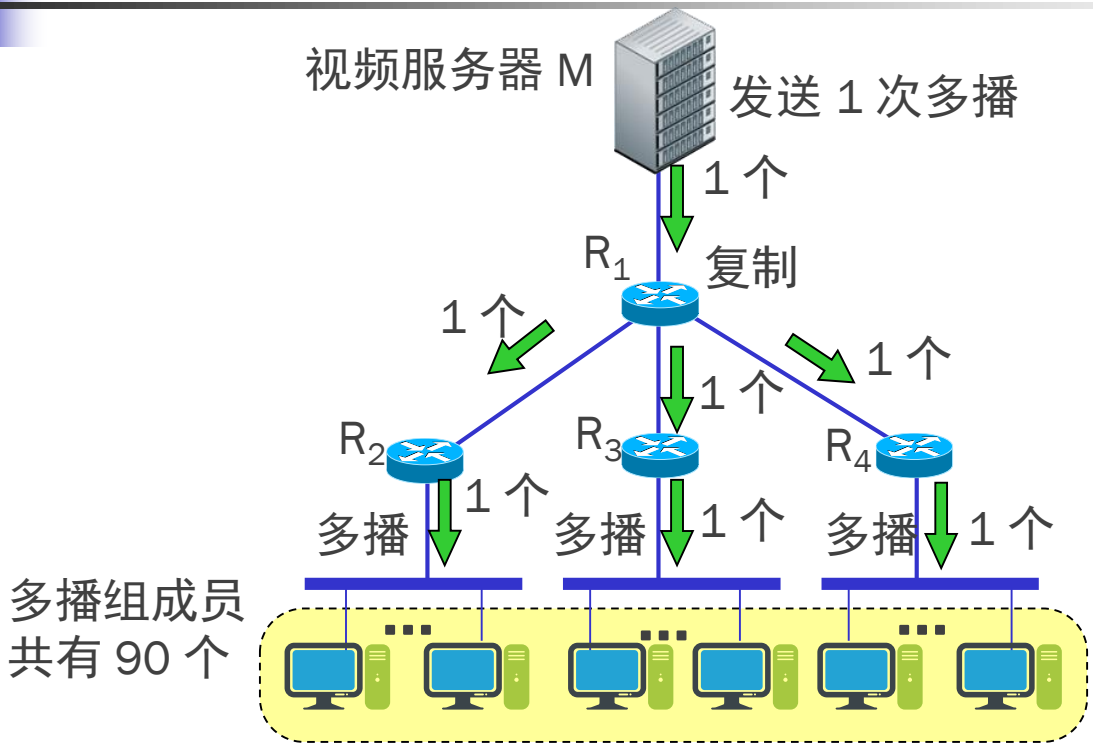
4.7.1 IP 多播的基本概念



不使用多播时需要
发送 90 次单播

共有 90 个主机接收视频节目

4.7.1 IP 多播的基本概念



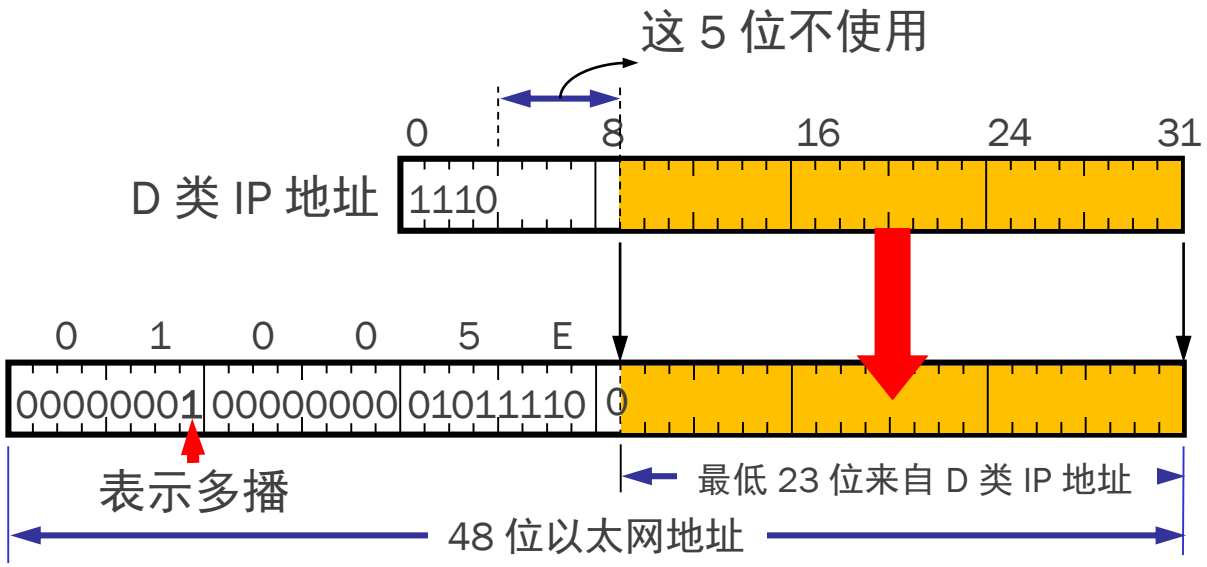
IP 多播的一些特点

- (1) 多播使用组地址—— IP 使用 D 类地址支持多播。多播地址只能用于目的地址，而不能用于源地址。
- (2) 永久组地址——由因特网号码指派管理局 IANA 负责指派。
- (3) 动态的组成员
- (4) 利用局域网的硬件多播

4.7.2 在局域网上进行硬件多播

- 因特网号码指派管理局 IANA 拥有的以太网地址块的高 24 位为 00-00-5E，以太网多播地址块的范围是：从 00-00-5E-00-00-00 到 00-00-5E-7F-FF-FF
- 而 TCP/IP 协议使用的 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。

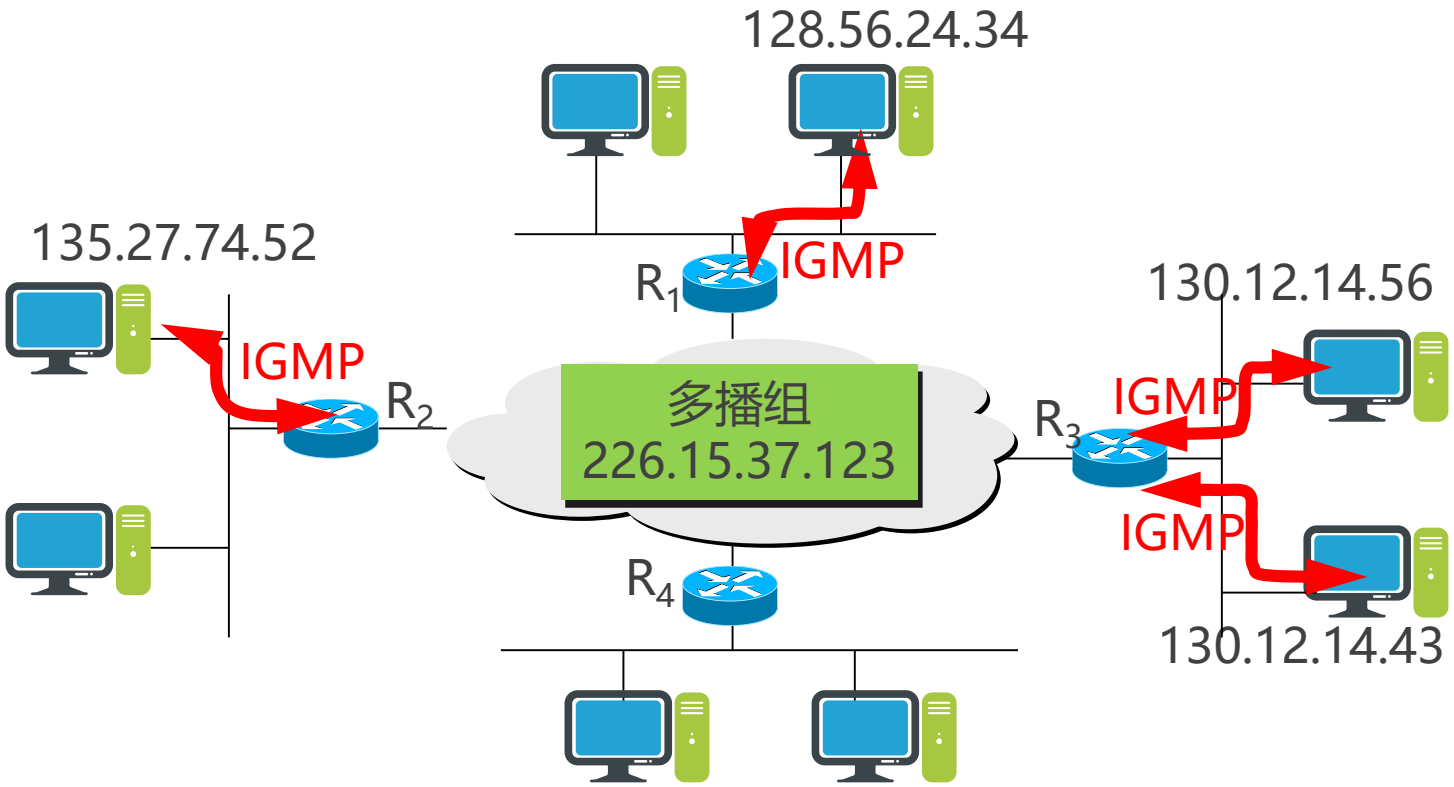
D 类 IP 地址与以太网多播地址的映射关系



4.7.3 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因特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

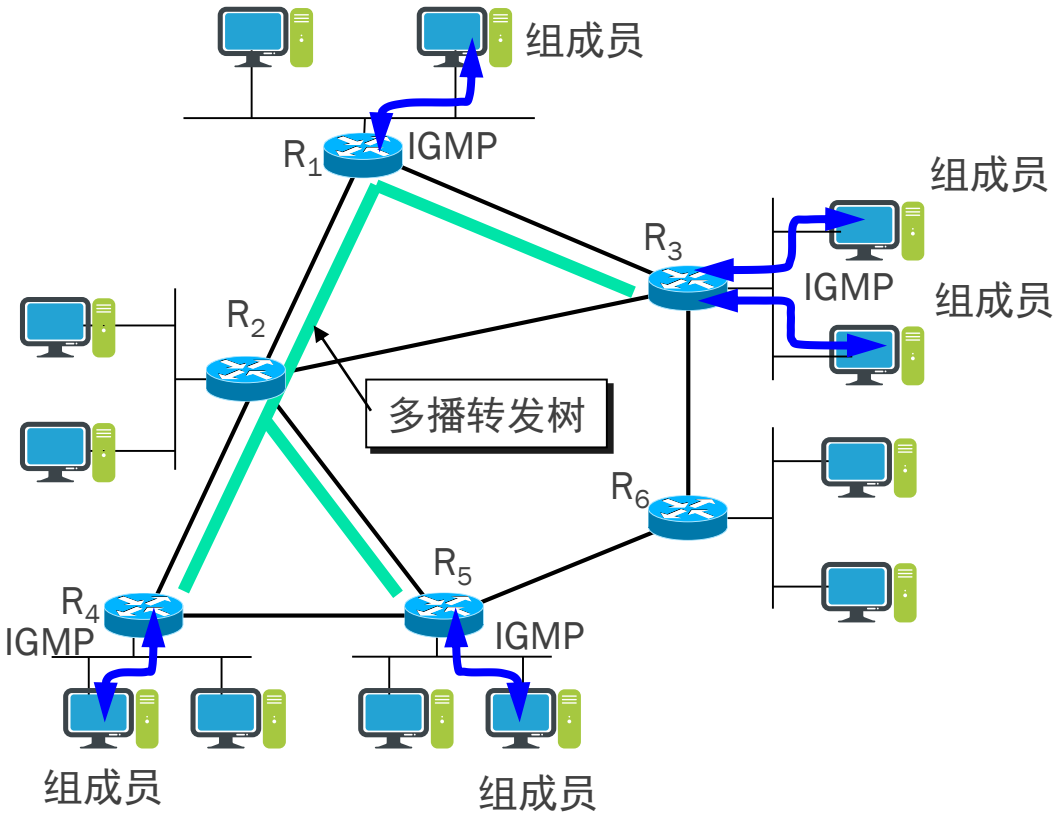


IGMP 的本地使用范围

- IGMP 并非在因特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

- 基本任务就是在多播路由器之间为每个多播组建立一个连接所有拥有该组成员的路由器的多播转发树。



多播路由选择协议比单播路由选择协议要复杂得多

- 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。

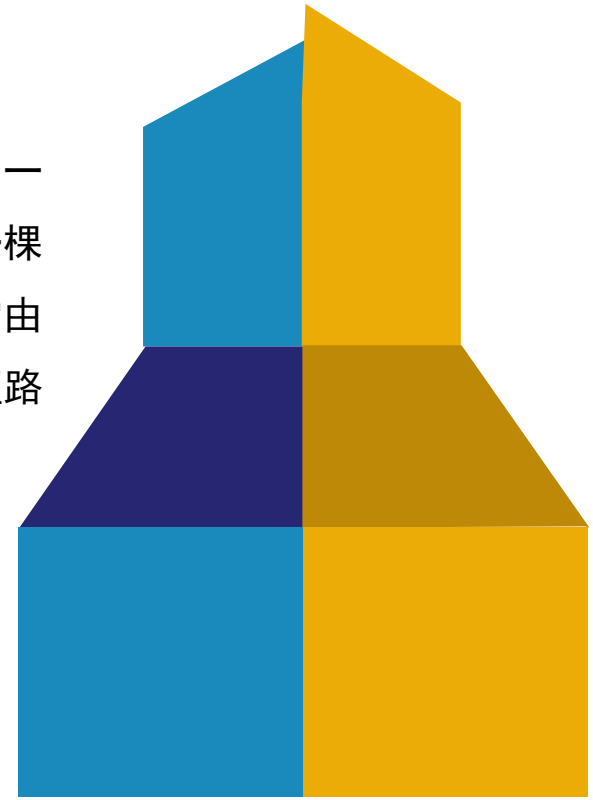
4.7.4 网际组管理协议IGMP

- **加入多播组**。当一台主机要加入某个多播组时，向本网络中的路由器发送一个IGMP**成员报告**报文。报告中包含要加入的多播组的地址。
- **监视成员变化**。多播路由器会周期性地发送一个**成员查询**报文，若长时间没有收到某个多播组的成员报告则认为没有该组的成员。
- **离开多播组**。当主机要退出一个多播组时，可主动发送一个**离开组**报文而不必等待路由器的查询。

为了提高工作效率，IGMP报文本身使用IP多播进行传送！

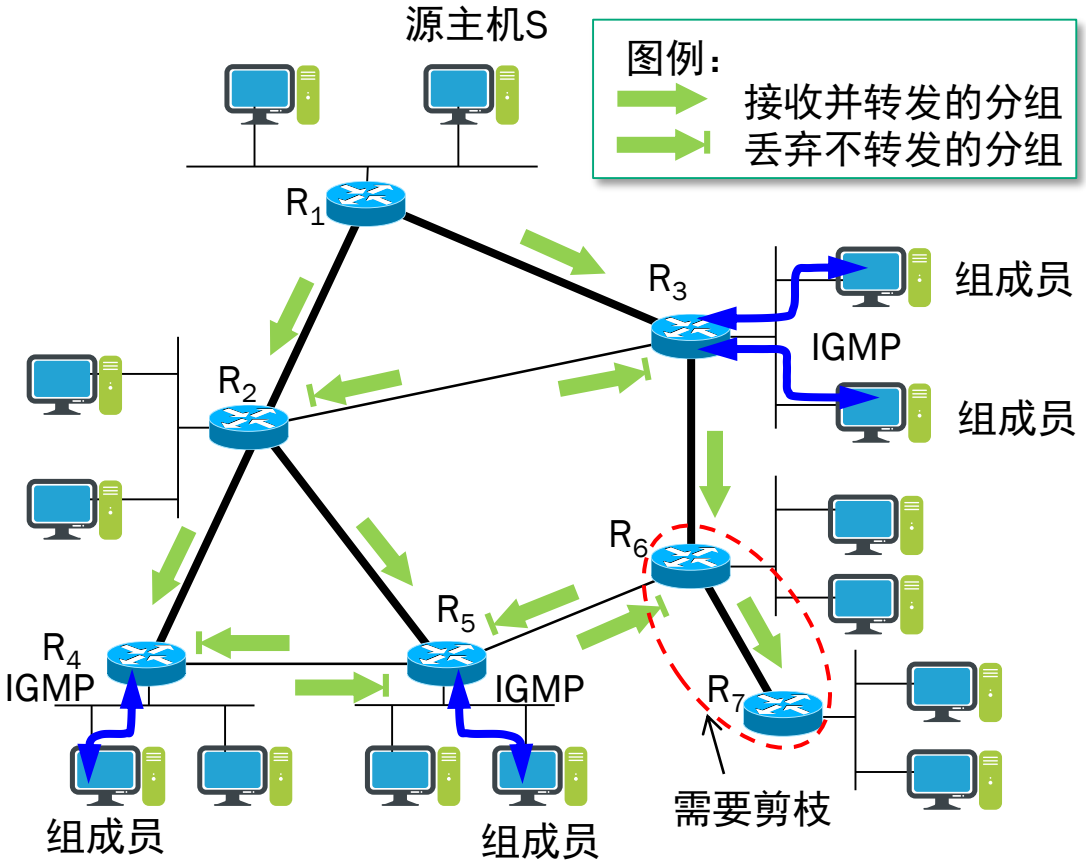
4.7.5 多播路由选择

基于源树多播路由选择：为一个多播组内的每个源构建一棵多播转发树，该转发树通常由每个成员路由器到源的最短路径构成。

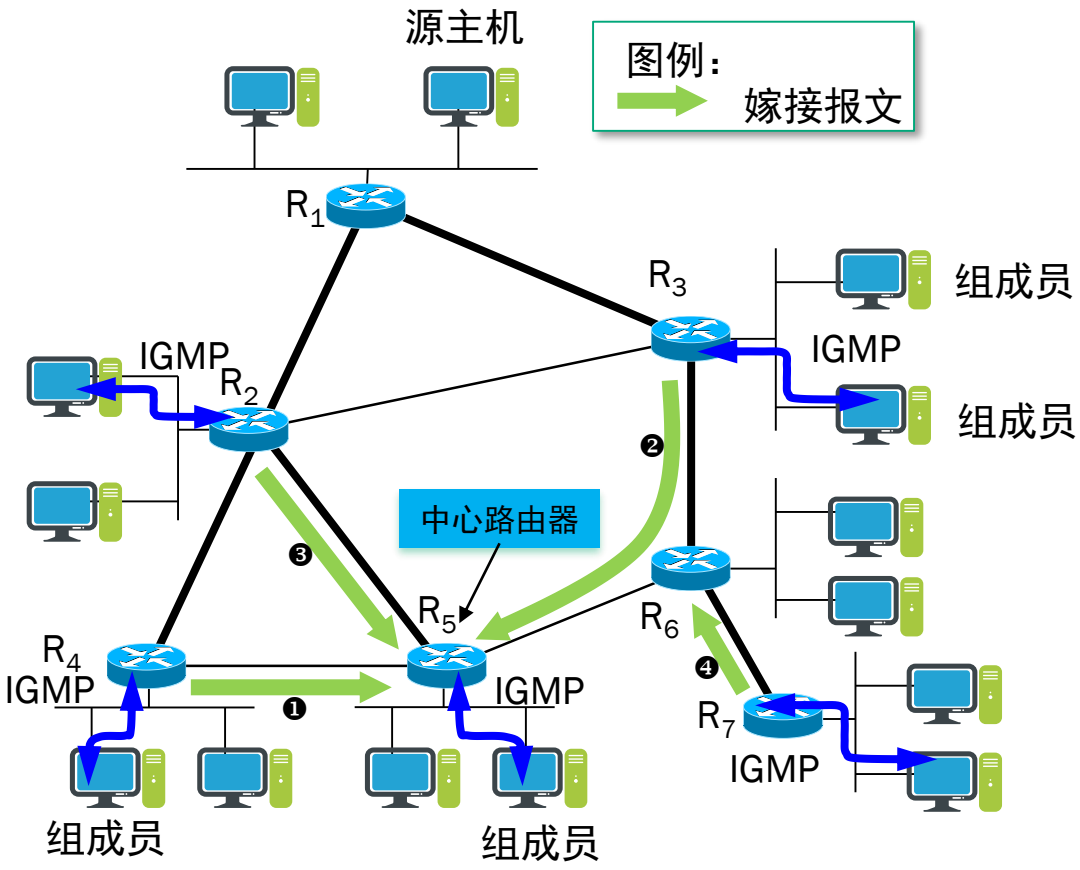


组共享树多播路由选择：在每个多播组中以中心路由器为根建立一棵连接所有成员路由器的多播转发树。组内所有源将多播分组通过单播IP隧道发送到中心路由器，再由中心路由器将多播分组在共享树上进行洪泛。

1. 基于源树多播路由选择



2. 共享树多播路由选择


源主机
图例：
嫁接报文
R1
R3
组成员
IGMP
R2
IGMP
组成员
❷
中心路由器
❸
R5
R6
R4
IGMP
IGMP
❹
❶
R7
IGMP
组成员
组成员

4.8.1 移动性对网络应用的影响

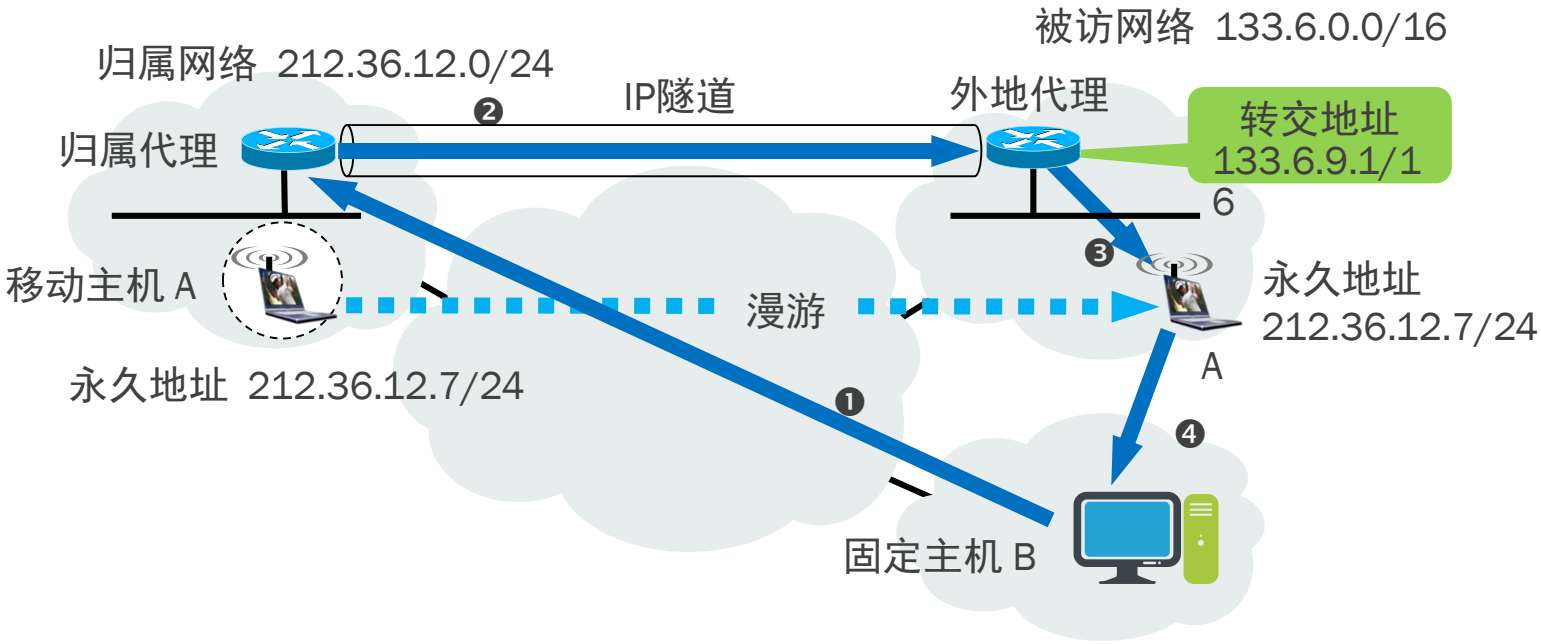
- 在Wi-Fi内部移动对于正在通信的应用程序来说是完全透明的，移动设备并没有改变它的IP地址。
- 但如果移动设备跨越网络进行漫游并不断改变其IP地址，则会给需要持续保持网络连接的应用带来很大的麻烦。
- 移动IP 的任务就是在IP层为上层网络应用提供**移动透明性**。

4.8.2 移动IP的工作原理

- 移动主机初始申请接入的网络被称为**归属网络**(home network)，在归属网络的IP地址被称为**归属地址**(home address)，在归属网络中代表移动主机执行移动管理功能的实体称为**归属代理**(home agent)。
- 移动主机当前漫游所在的网络叫**外地网络**(foreign network)。在外地网络中帮助移动主机执行移动管理功能的实体称为**外地代理**(foreign agent)，外地代理为移动主机提供临时的外地网络的**转交地址**(care-of address)。

移动IP中数据报的转发过程

这些过程对于任何与移动主机进行通信的固定主机来说都是完全透明的！



4.8.3 移动IP的标准

- **代理发现** 定义归属代理或外部代理向移动主机通告其服务时所使用的协议，以及移动主机请求一个外部代理或归属代理的服务时所使用的协议。
- **信息注册** 定义移动主机向外地代理注册或注销永久地址、归宿代理地址等信息，以及移动主机或外地代理向归宿代理注册或注销转交地址时所用的协议。
- **间接路由** 定义了数据报由一个归属代理转发给移动主机的方式，包括转发数据报的规则、差错处理规则和几种不同的封装形式。

4.9.1 解决 IP 地址耗尽的措施

- 从计算机本身发展以及从因特网规模和网络传输速率来看，现在 IPv4 已很不适用。
- 最主要的问题就是 32 位的 IP 地址不够用。
  - 到2011年2月，IPv4的地址已经耗尽
- 要解决 IP 地址耗尽的问题的措施：
  - 采用无类别编址 CIDR，使 IP 地址的分配更加合理。
  - 采用网络地址转换 NAT 方法以节省全球 IP 地址。
  - 采用具有更大地址空间的新版本的 IP 协议 IPv6。

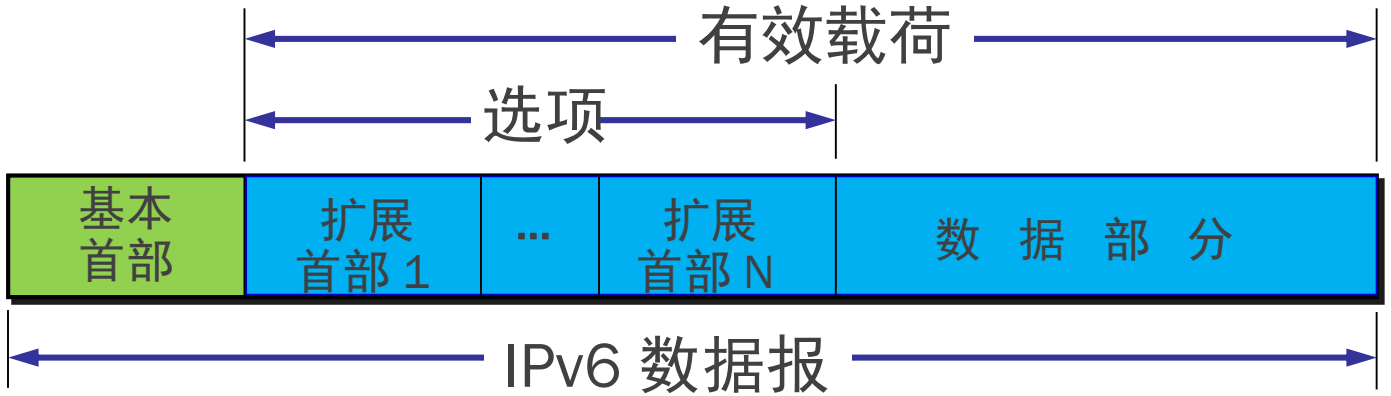
4.9.2 IPv6 的基本首部

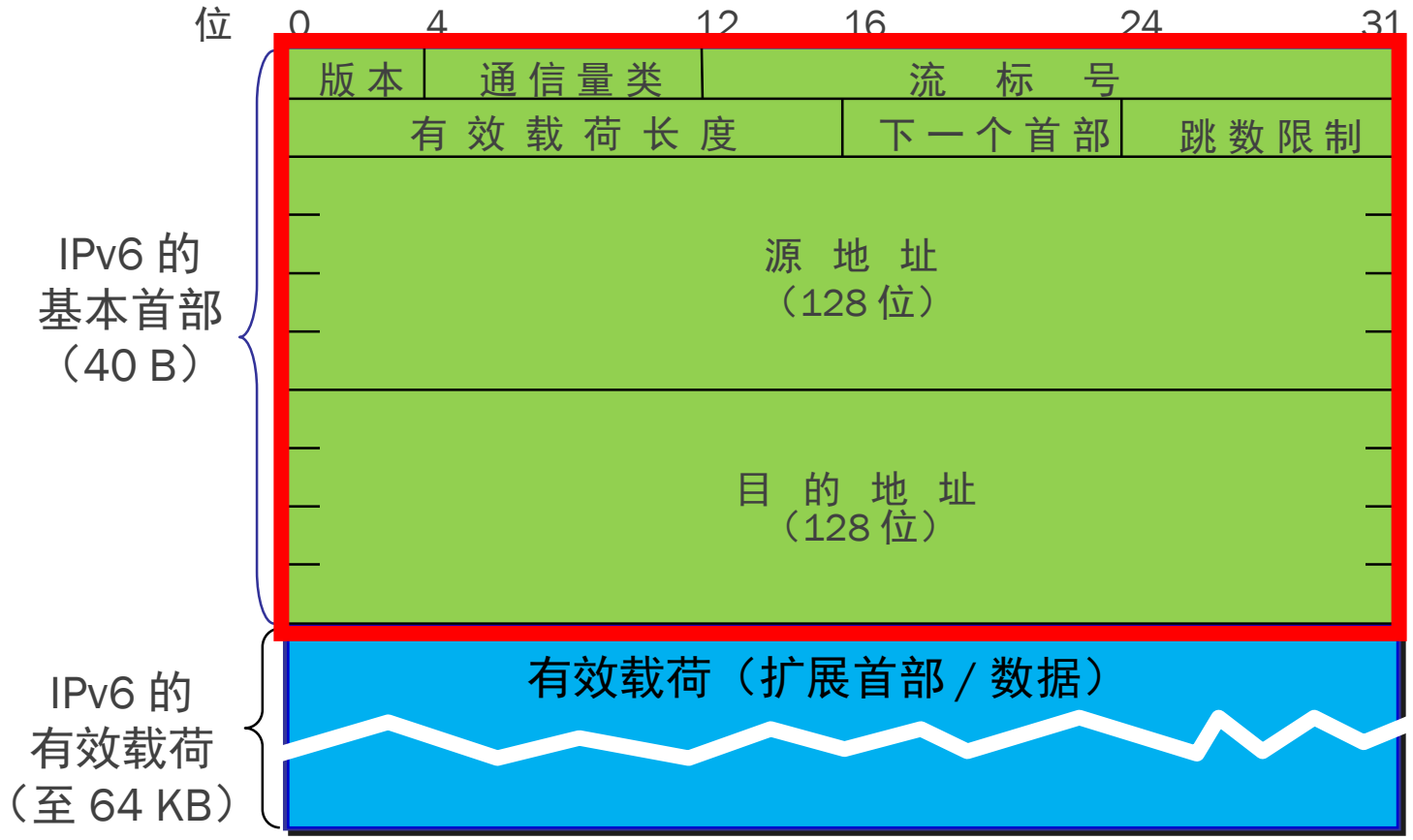
- IPv6 仍支持无连接的传送所引进的主要变化如下
- 更大的地址空间。IPv6 将地址从 IPv4 的 32 位 增大到了 128 位。
- 扩展的地址层次结构。
- 灵活的首部格式。
- 改进的选项。
- 允许协议继续扩充。
- 支持即插即用（即自动配置）
- 支持资源的预分配。

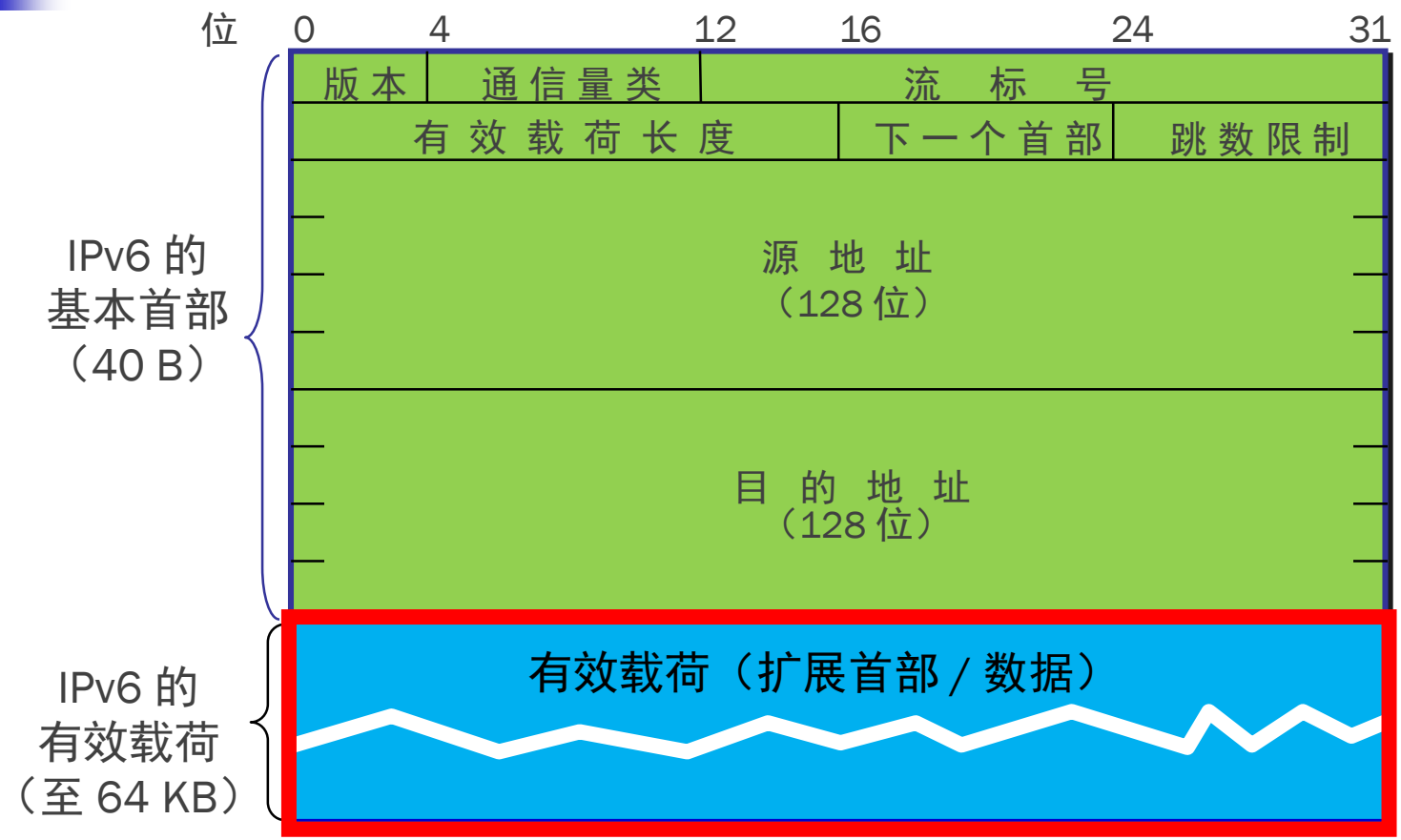
IPv6 数据报的首部

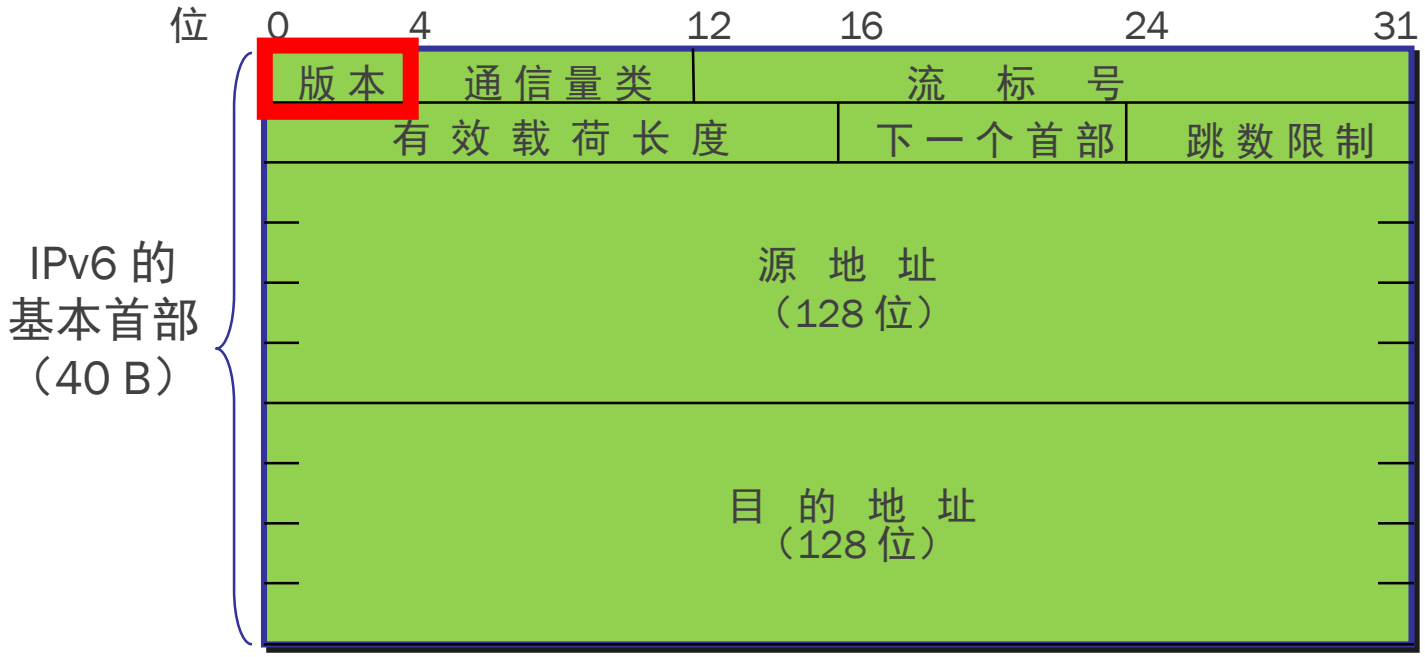
- IPv6 将首部长度变为固定的 40 字节，称为**基本首部**(base header)。
- 将不必要的功能取消了，首部的字段数减少到只有 8 个。
- 取消了首部的检验和字段，加快了路由器处理数据报的速度。
- 在基本首部的后面允许有零个或多个扩展首部。
- 所有的扩展首部和数据合起来叫做数据报的**有效载荷**(payload)或**净负荷**。

IPv6 数据报的一般形式

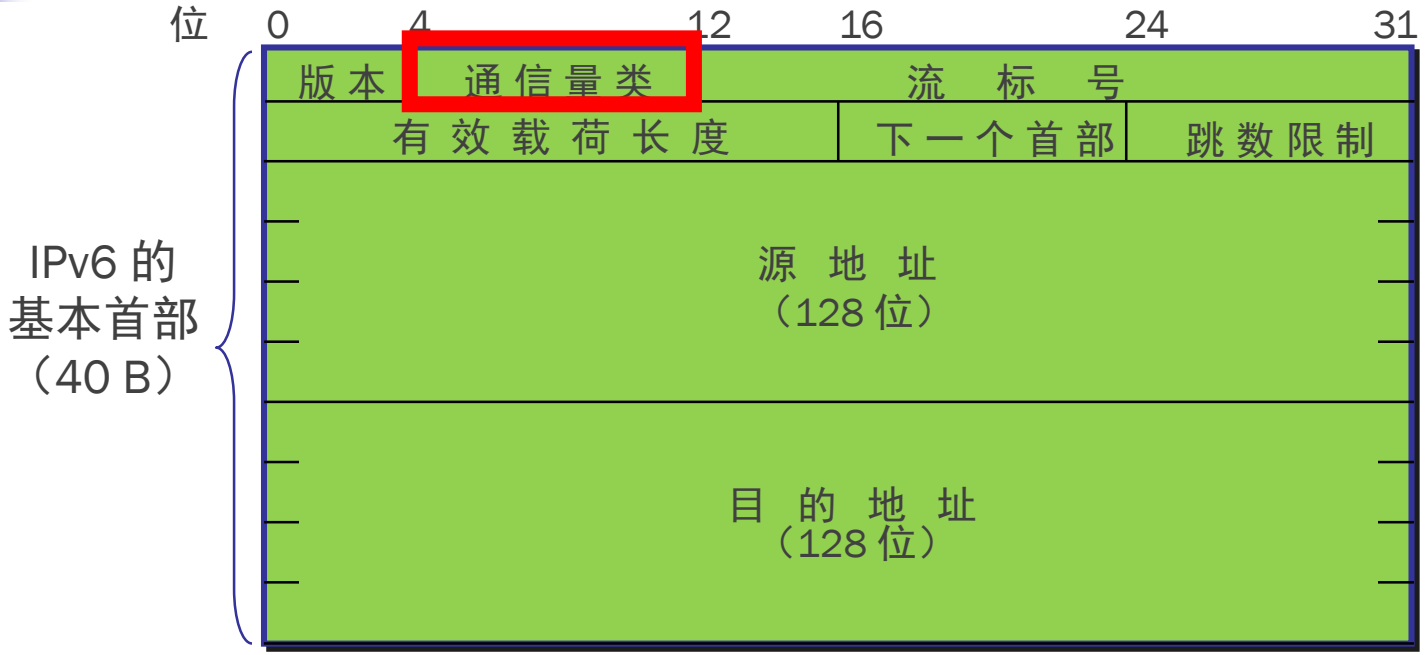



位
0
4
12
16
24
31
版本
通信量类
流 标 号
有效载荷长度
下一个首部
跳数限制
源 地 址
（128 位）
目 的 地 址
（128 位）
IPv6 的
基本首部
（40 B）
有效载荷（扩展首部 / 数据）
IPv6 的
有效载荷
（至 64 KB）

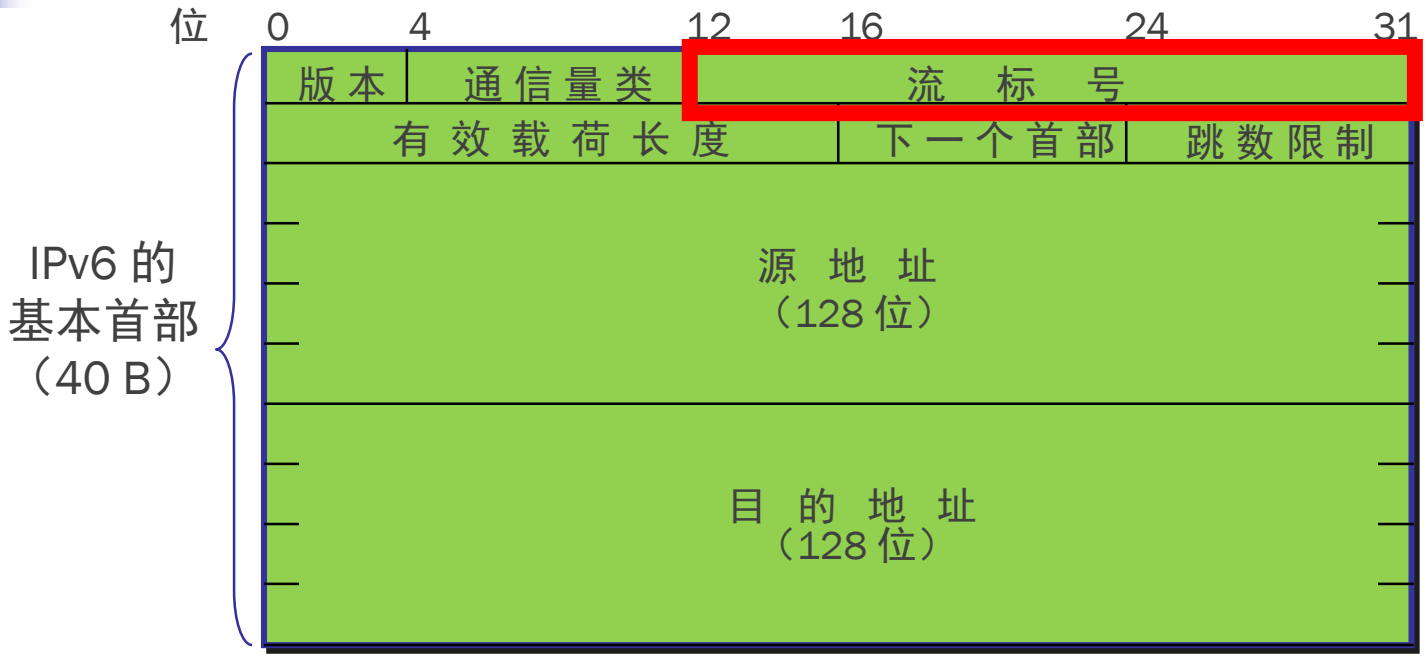

位
0
4
12
16
24
31
版本
通信量类
流 标 号
有效载荷长度
下一个首部
跳数限制
源 地 址
（128 位）
目 的 地 址
（128 位）
IPv6 的
基本首部
（40 B）
有效载荷（扩展首部 / 数据）
IPv6 的
有效载荷
（至 64 KB）



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

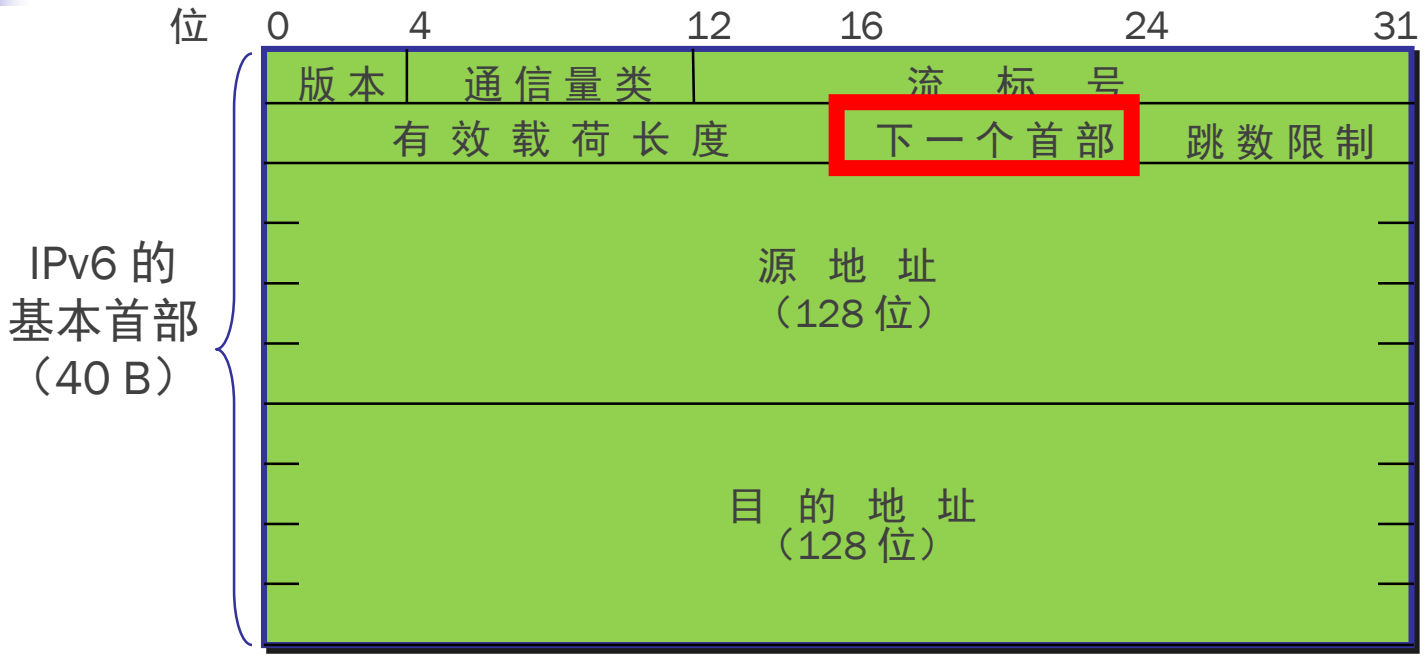


通信量类(traffic class)—— 8 位。这是为了区分不同的 IPv6 数据报的类别或优先级。目前正在进行不同的通信量类性能的实验。

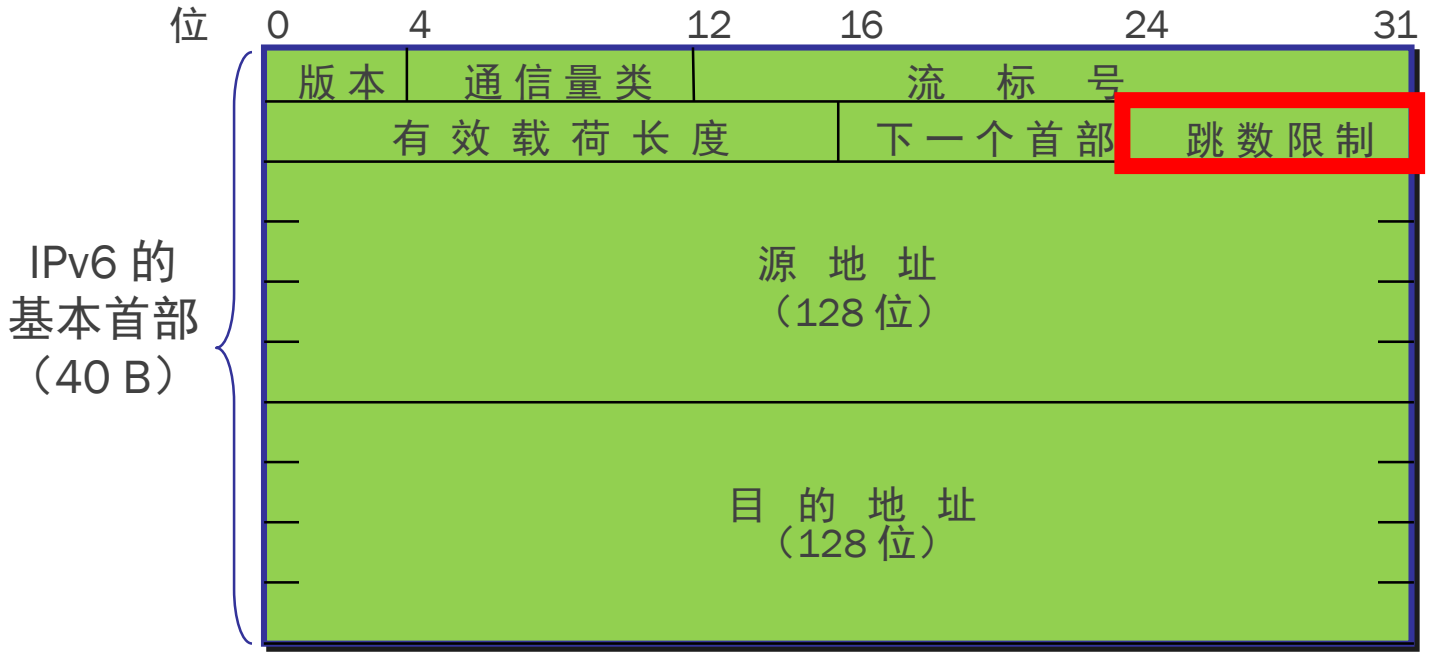


流标号(flow label)—— 20 位。 “流”是互联网络上从特定源点到特定终点的一系列数据报， “流”所经过的路径上的路由器都保证指明的服务质量。
所有属于同一个流的数据报都具有同样的流标号。

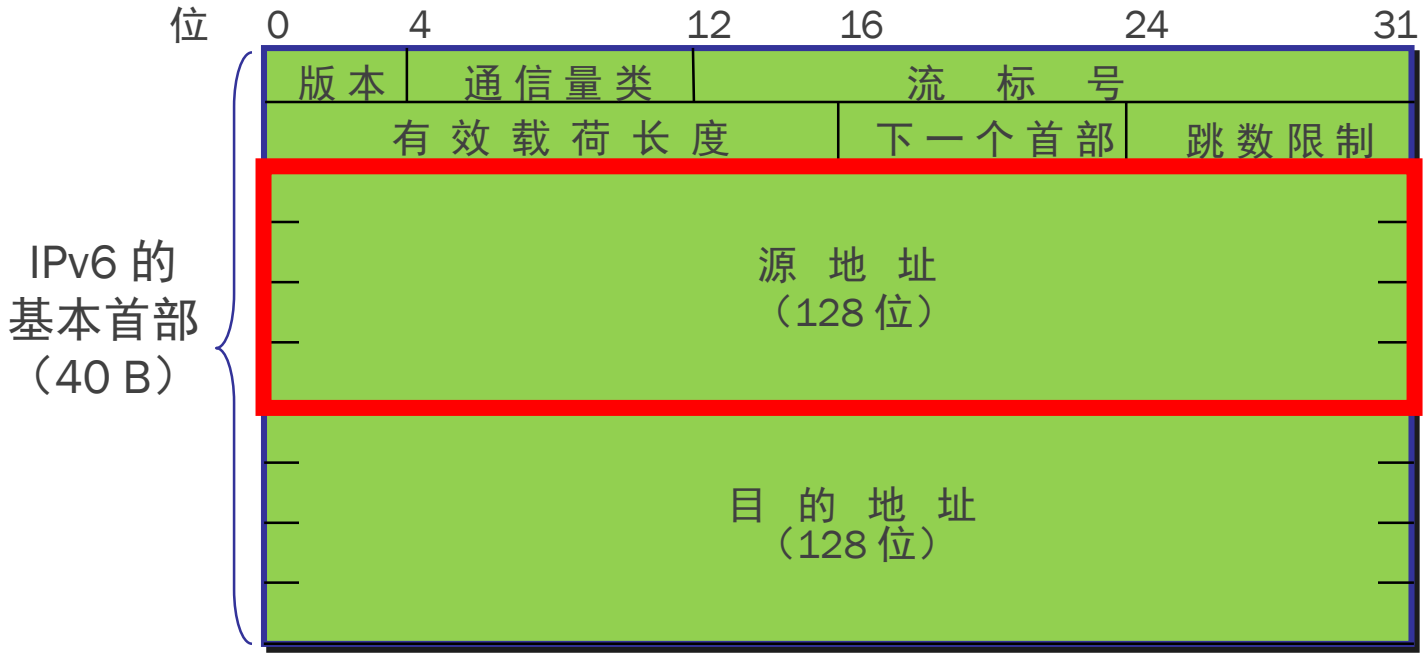
有效载荷长度(payload length)—— 16 位。它指明 IPv6 数据报除基本首部以外的字节数（所有扩展首部都算在有效载荷之内），其最大值是 64 KB。



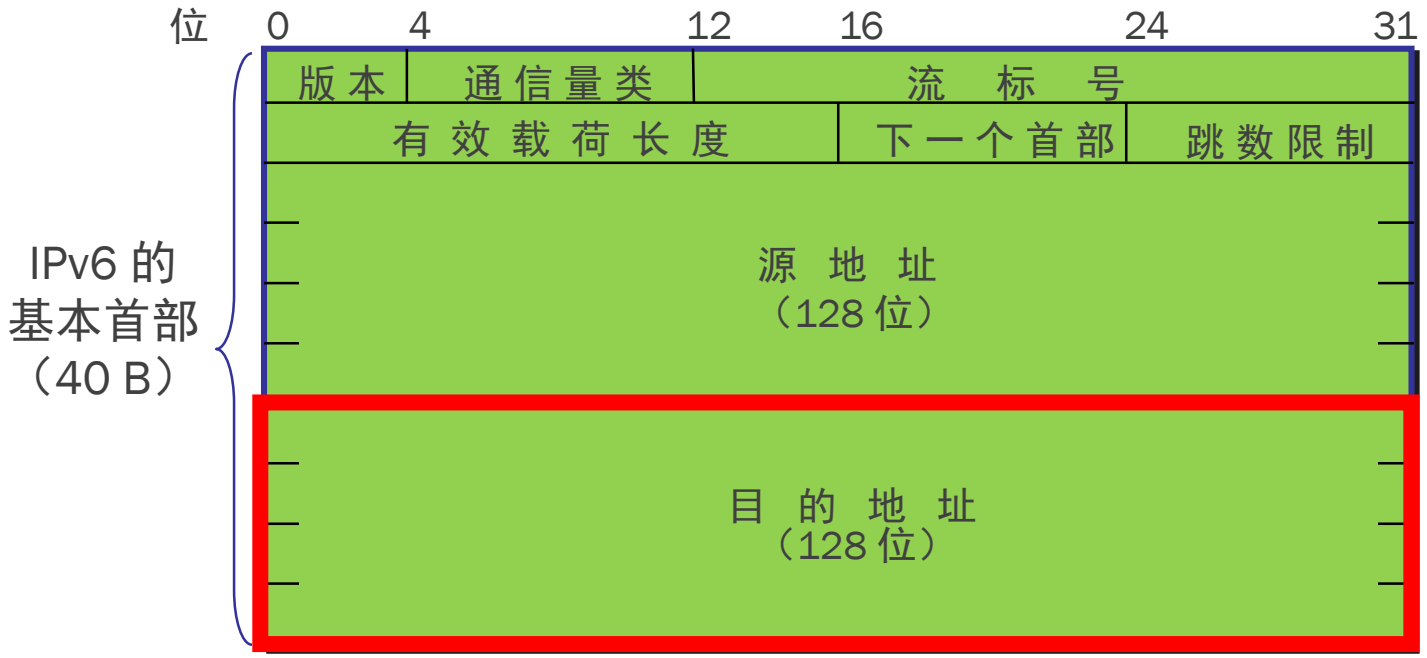
下一个首部(next header)—— 8 位。它相当于 IPv4 的协议字段或可选字段。



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。

4.9.3 IPv6 的编址

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)　单播就是传统的点对点通信。
2. 多播(multicast)　多播是一点对多点的通信。
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。
- IPv6 地址是分配给结点上面的接口。
  - 一个接口可以有多个单播地址。
  - 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(colon hexadecimal notation)

- 每个 16 位的值用十六进制值表示，各值之间用冒号分隔。
  68E6:8C64:FFFF:FFFF:0:1180:960A:FFFF
- 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　可以写成：
  - FF05::B3

点分十进制记法的后缀

- 0:0:0:0:0:0:128.10.2.1（用于IPv4向IPv6过渡）
  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。
- 60 位的前缀 12AB00000000CD3 可记为：
  12AB:0000:0000:CD30:0000:0000:0000:0000/60

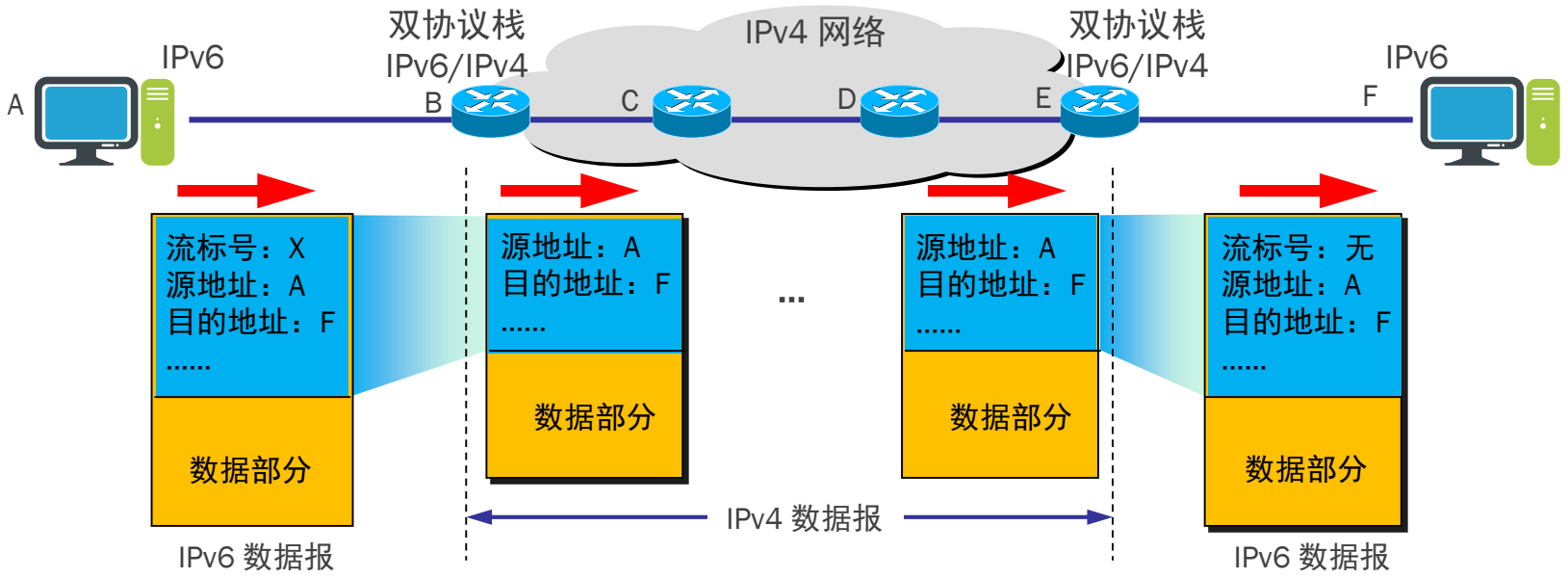
或12AB::CD30:0:0:0:0/60

或12AB:0:0:CD30::/60

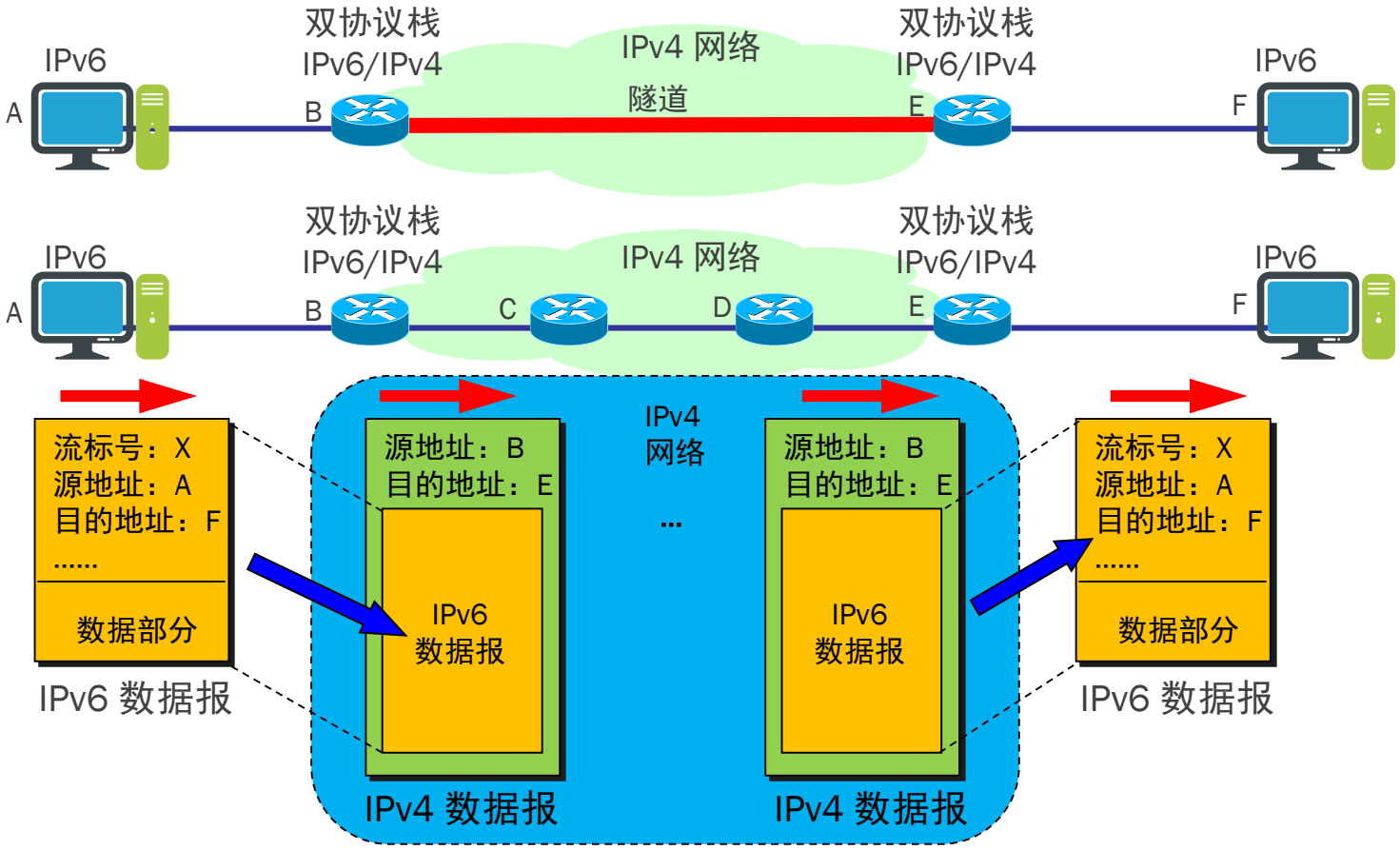
4.9.4 从 IPv4 向 IPv6 过渡

- 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。
- **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。

用双协议栈进行从 IPv4 到 IPv6 的过渡



使用隧道技术从 IPv4 到 IPv6 过渡



4.9.5 ICMPv6

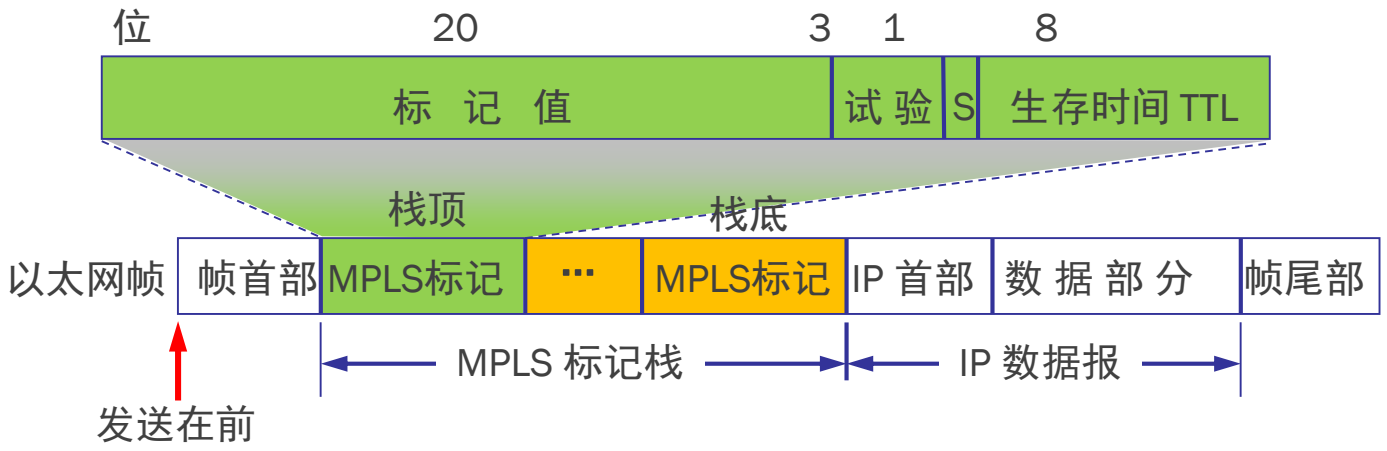
- ICMPv6 的报文格式和 IPv4 使用的 ICMP 的相似，即前 4 个字节的字段名称都是一样的。
- 但 ICMPv6 将第 5 个字节起的后面部分作为报文主体。
- ICMPv6 的报文划分为两大类
  - 差错报告报文
  - 提供信息的报文
- 没有IGMPv6，因为ICMPv6已经包含了所有IGMP的功能。

4.10 多协议标签交换MPLS

- MPLS (Multiprotocol Label Switching) 试图将虚电路的一些特点与数据报的灵活性和健壮性进行结合
- 其最初的目标是通过采用来自虚电路网络界的一个关键概念，即固定长度标签，来改善IP路由器的转发速度
- MPLS使能路由器（标签交换路由器）通过检查相对短的、固定长的标签来转发分组

MPLS 首部的位置与格式

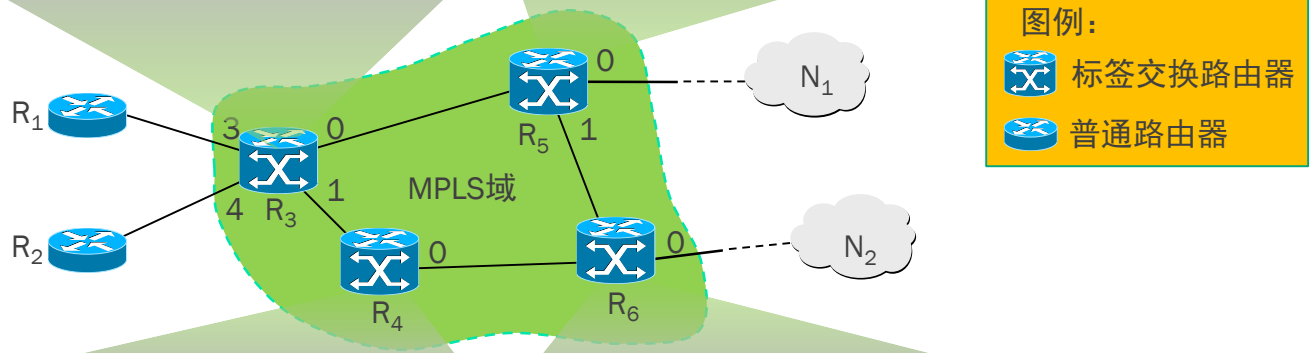
- MPLS 的一个重要功能就可以构成标记栈。
- MPLS 标记的格式以及标记栈：



MPLS帧的转发

入标签	出标签	目的地	出接口
	5	N_2	0
	10	N_1	0
	9	N_2	1

入标签	出标签	目的地	出接口
5	8	N_2	1
10	—	N_1	0



入标签	出标签	目的地	出接口
9	7	N_2	0

入标签	出标签	目的地	出接口
7	—	N_2	0

4.10 多协议标签交换MPLS

- MPLS的真正优点在于它的流量管理能力：提供沿多条路径转发分组的能力，并能灵活地为某些流量指定其中的一条路径
- 这种能力被称为显示路由，其应用之一就是流量工程(traffic engineering)
- MPLS还能用于实现虚拟专用网VPN 和改进网络的服务质量

课题习题

- 1、网络层信息传输是否带稳定的连接？是否安全可靠?
- 2、简述IP不足问题主要通过哪些办法缓解或解决?
- 3、简述物理地址、IP地址和MPLS标签的差异。

谢谢！