



数据交付的详细环节分析报告

15卓越 火忻



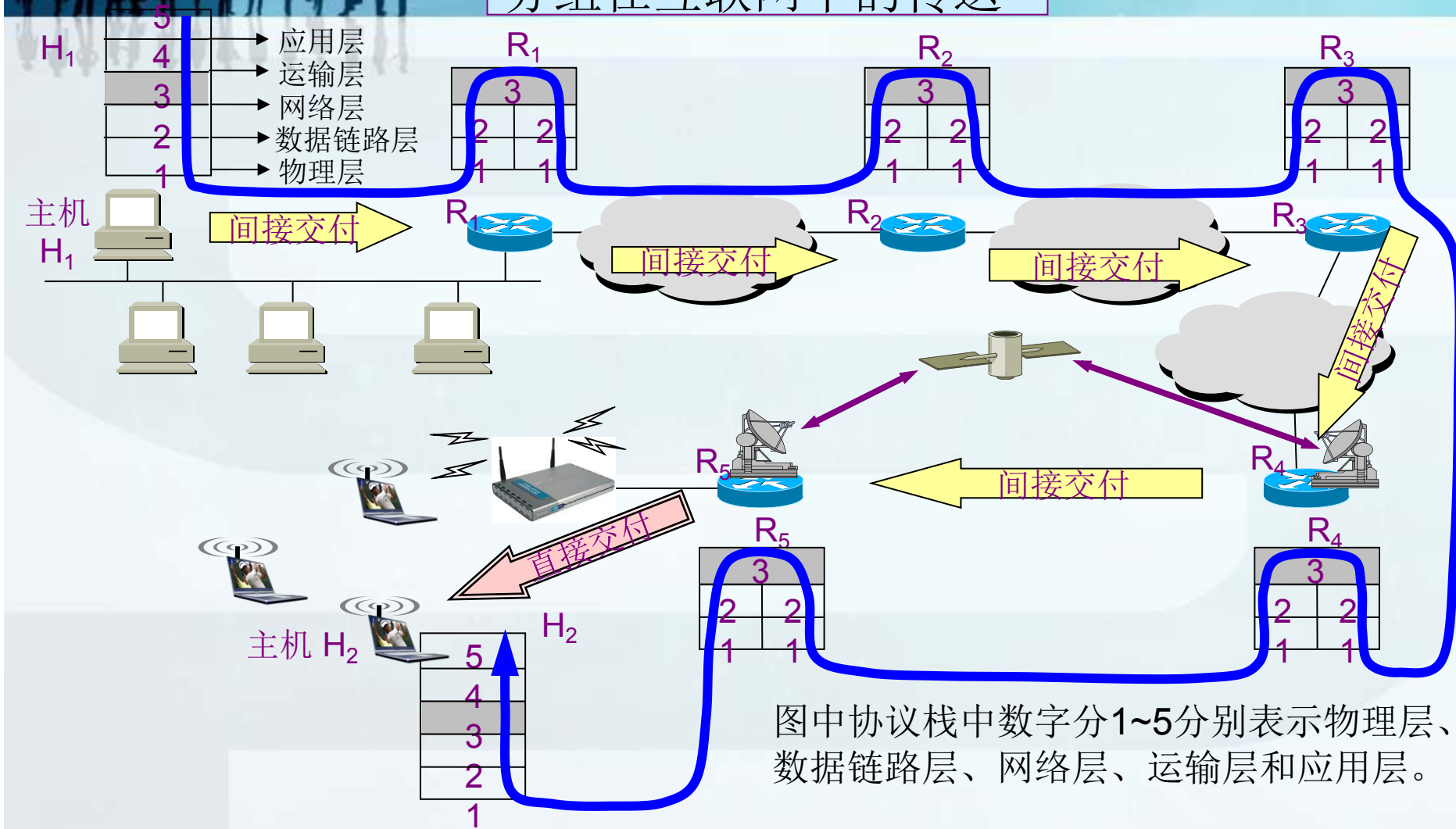
一、总体图

二、细节描述

三、各层协议

四、路由器转发

分组在互联网中的传送

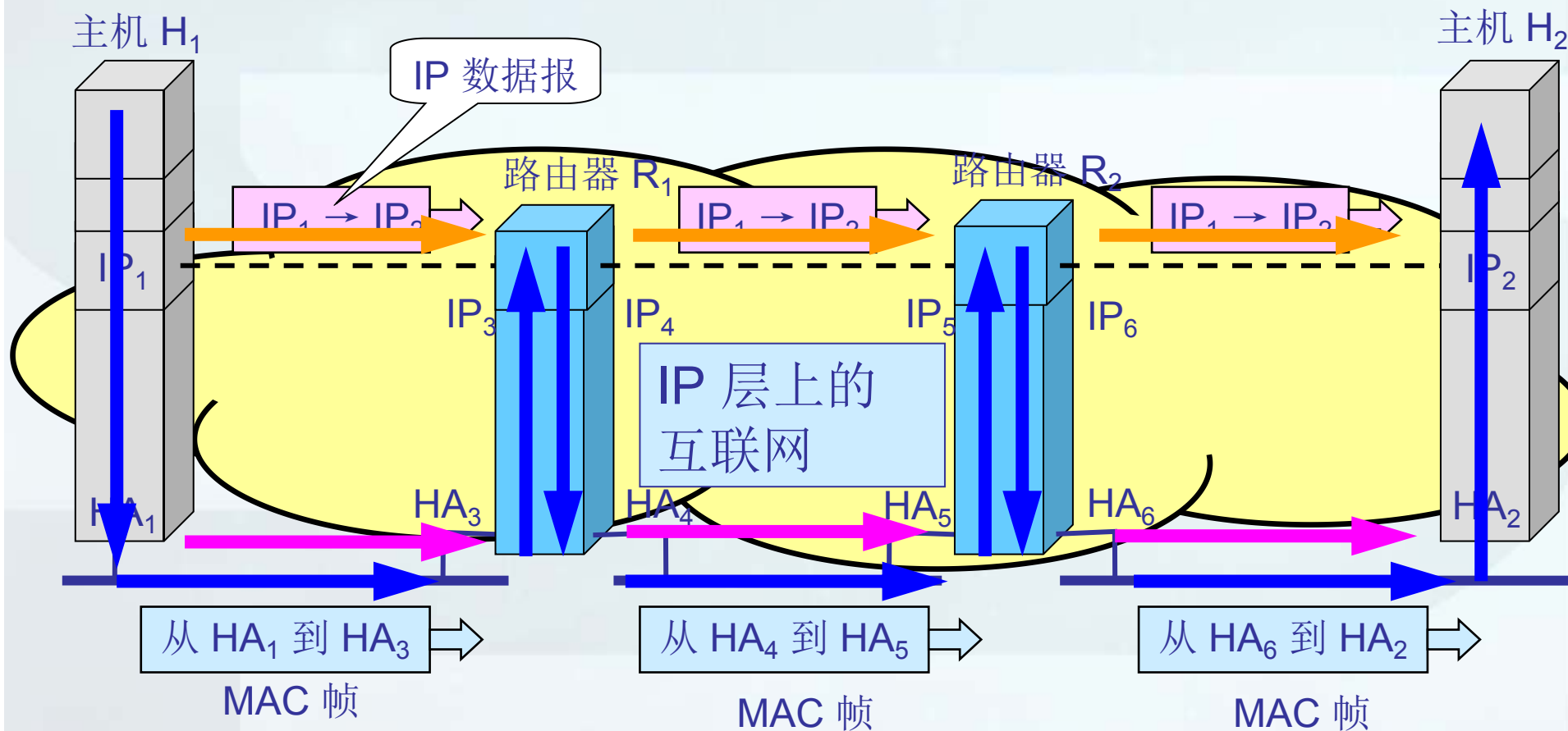


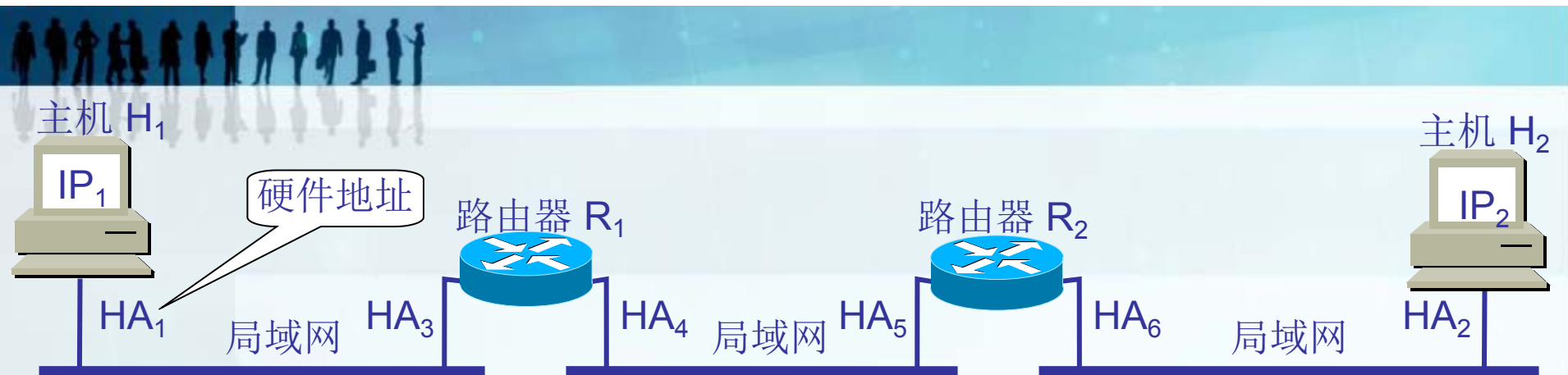
数据流动

从协议栈

从IP层

在链路上看MAC帧



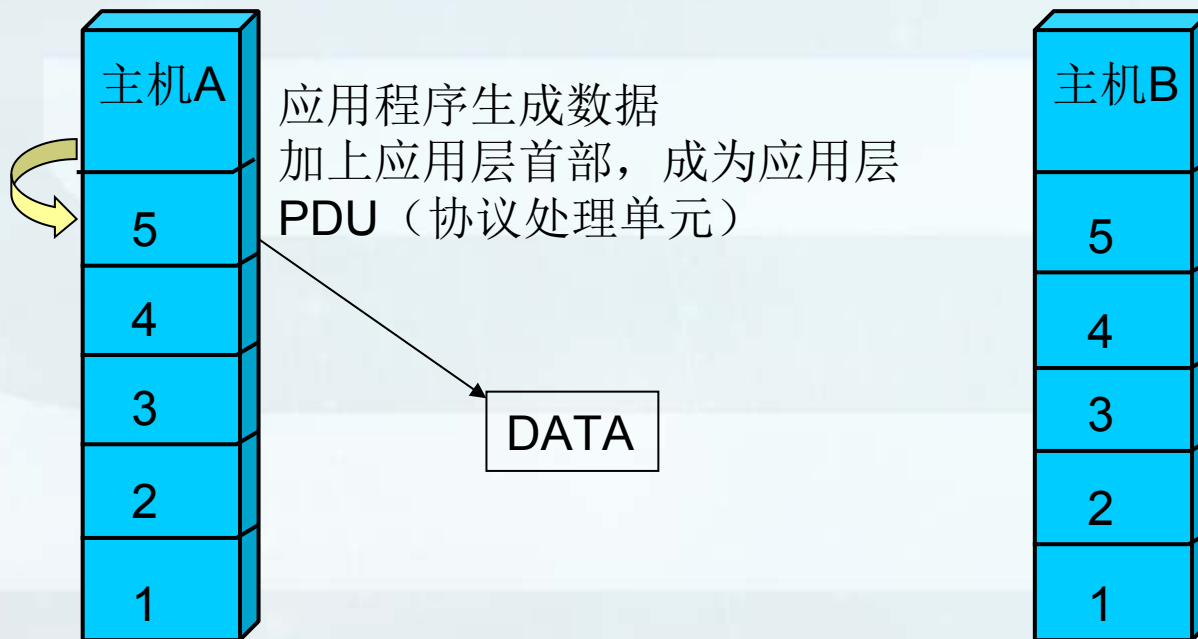


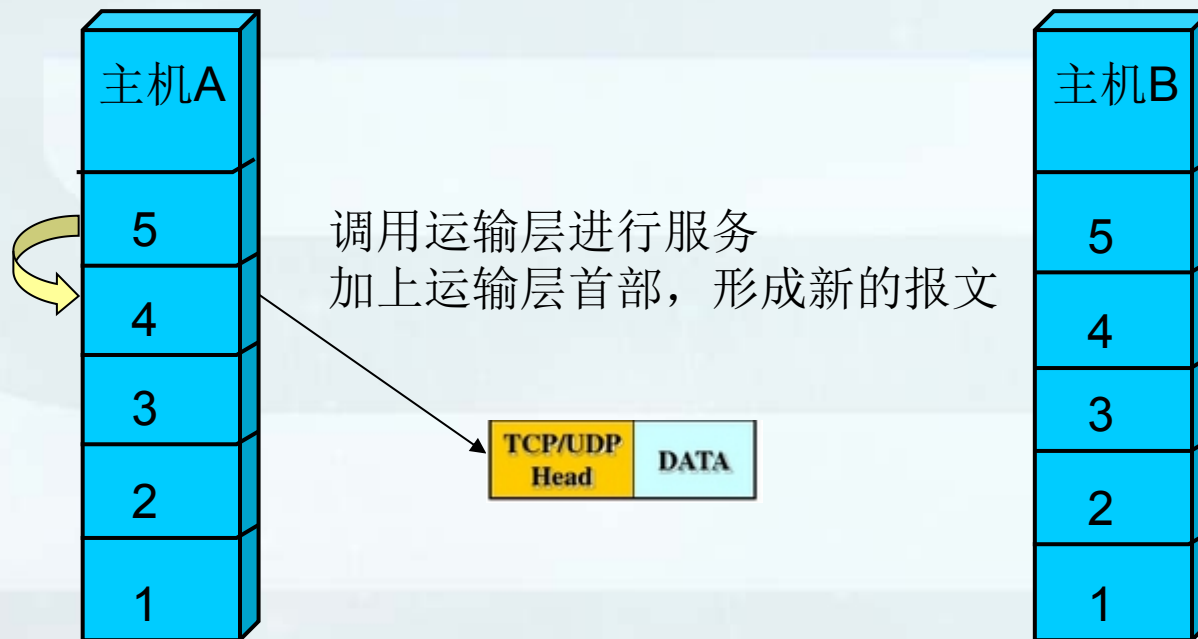
主机H1向H2发送数据，通信的路径：

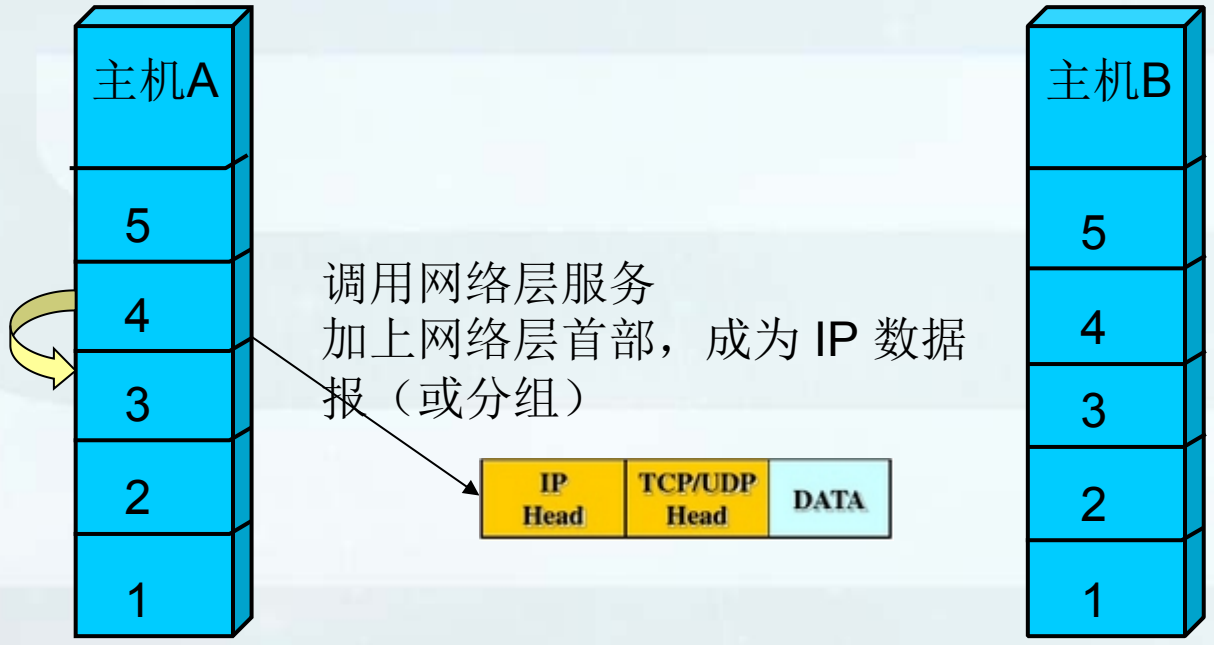
H₁ → 经过 R₁ 转发 → 再经过 R₂ 转发 → H₂

从层次角度看：

物理地址是**数据链路层**和**物理层**使用的地址，而IP地址是**网络层**和**以上各层**使用的地址，是一种**逻辑地址**。



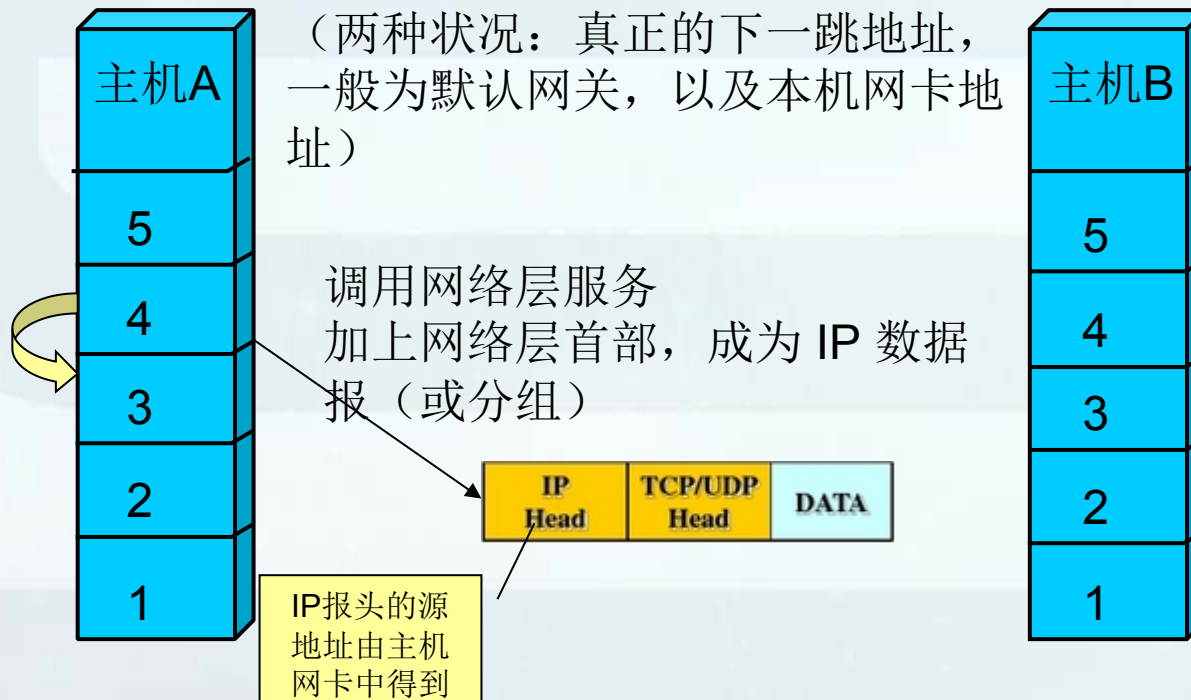




查主机路由表

使用目的IP与PC机路由表比较，
得下一跳ip地址和出接口。

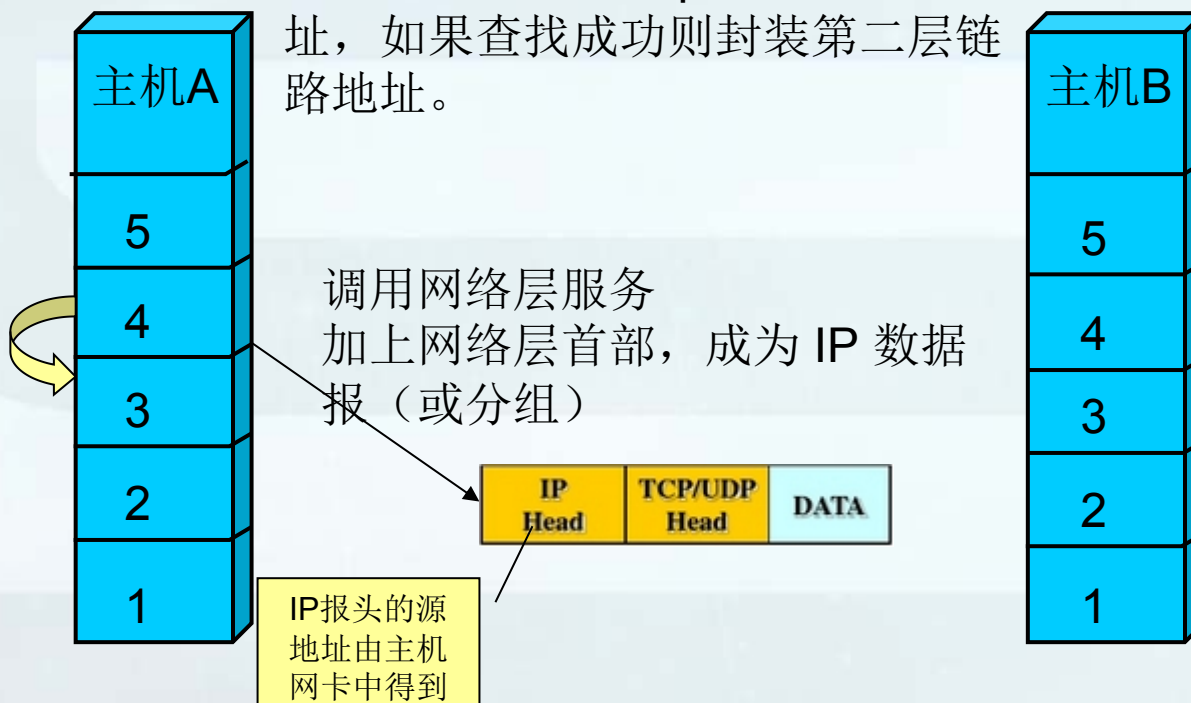
（两种状况：真正的下一跳地址，
一般为默认网关，以及本机网卡地
址）



ARP表查找下一跳mac地址

情况一：查找下一跳mac地址；

情况二：查找目的ip对应的mac地址，如果查找成功则封装第二层链路地址。

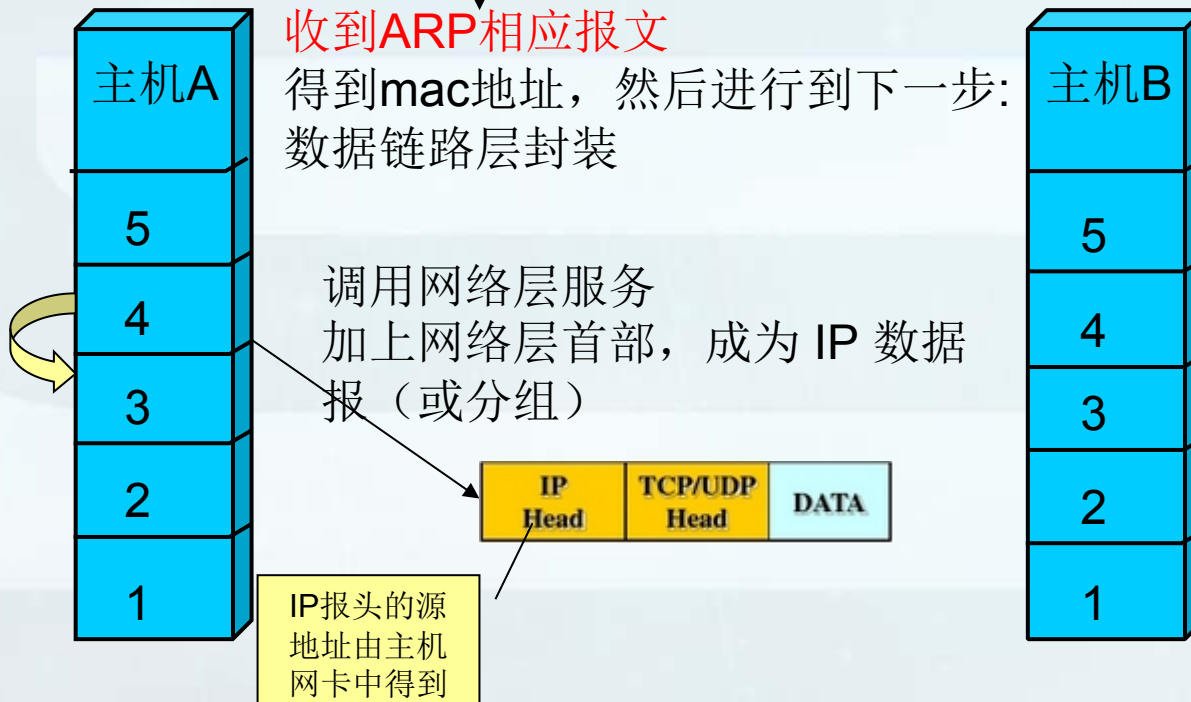


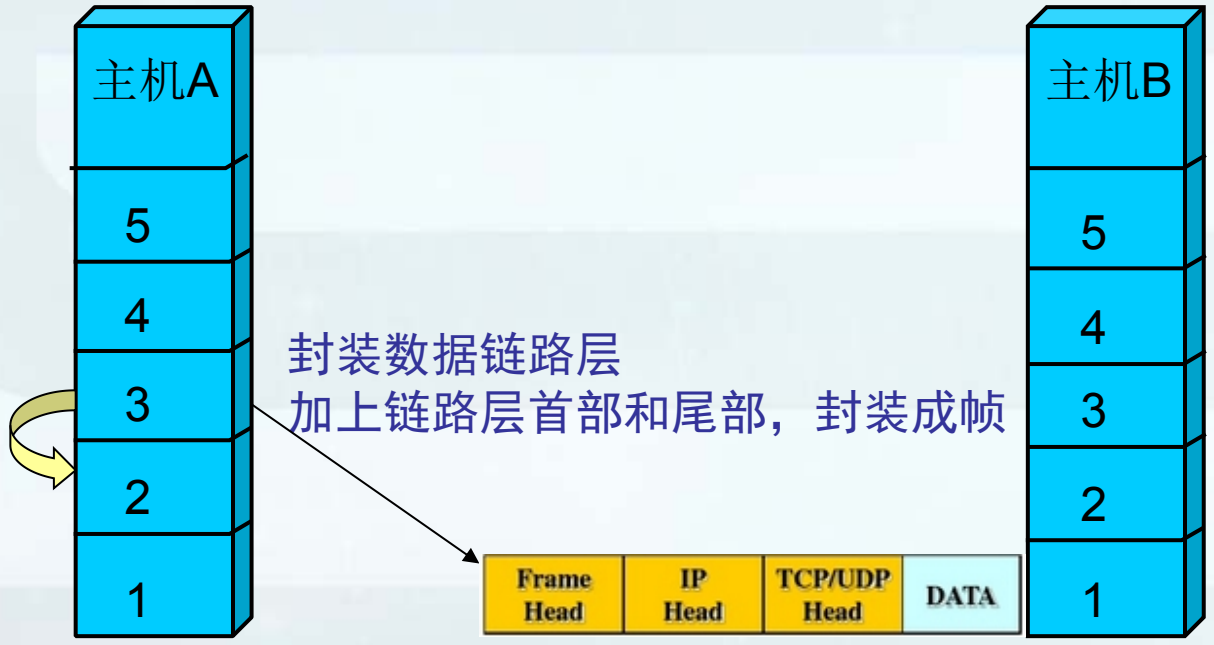
发送ARP请求

获得下一跳或目标IP的mac地址

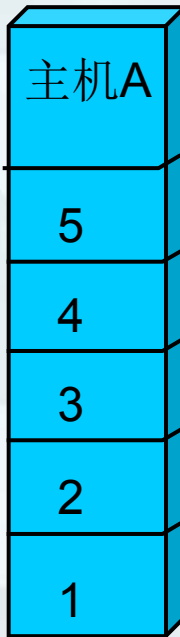
收到ARP相应报文

得到mac地址，然后进行到下一步：
数据链路层封装

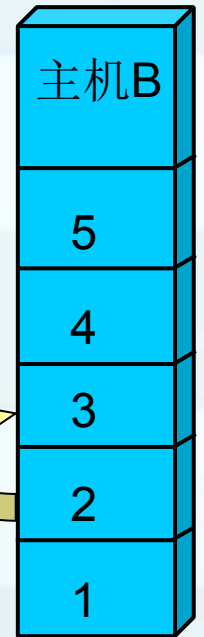
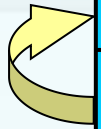


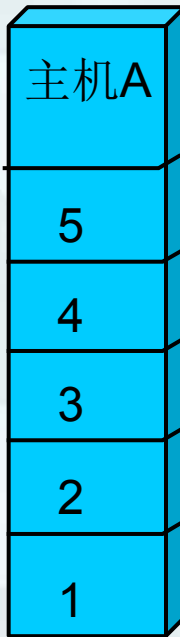




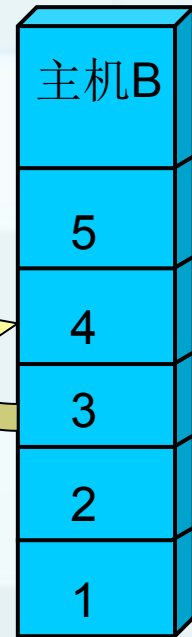


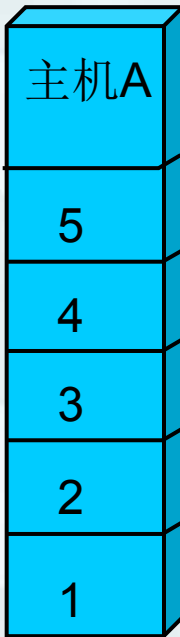
去掉链路层地址
取出数据部分，上交给
网络层





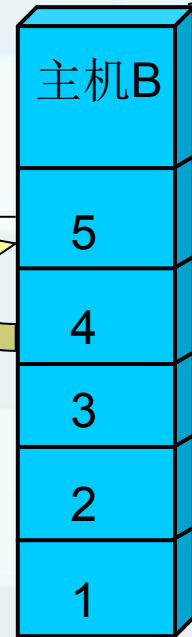
若目的ip是接口ip,则去掉ip包头,根据协议号送上层处理





交由应用程序进行处理

Text indicating that the data is handed over to the application for processing.



接受成功


Text indicating successful reception.



数据包变换

在发送数据时，数据从高层下到低层，然后才能到通信链路上传输。使用IP地址的IP数据报一旦交给数据链路层，就被封装成MAC帧，MAC帧在传送时使用的源地址和目的地址都是硬件地址，这两个硬件地址都写在MAC帧的首部了。





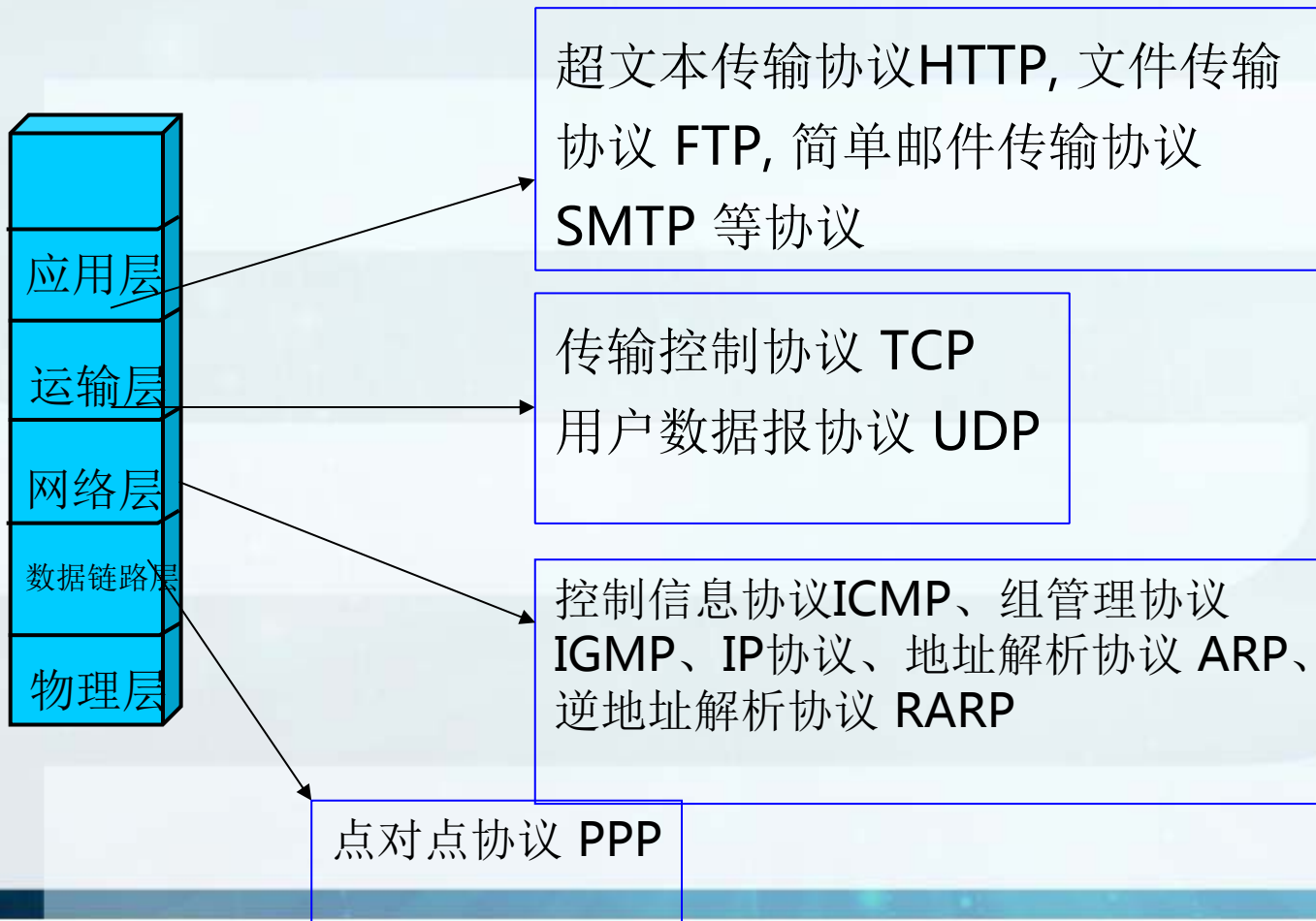
IP和硬件是如何相互转换的，换句话说，知道一个机器的IP如何知道它的硬件地址？

ARP协议

ARP 高速缓存(ARP cache)

每一个主机都设有一个 ARP 高速缓存(ARP cache) 存储所在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。

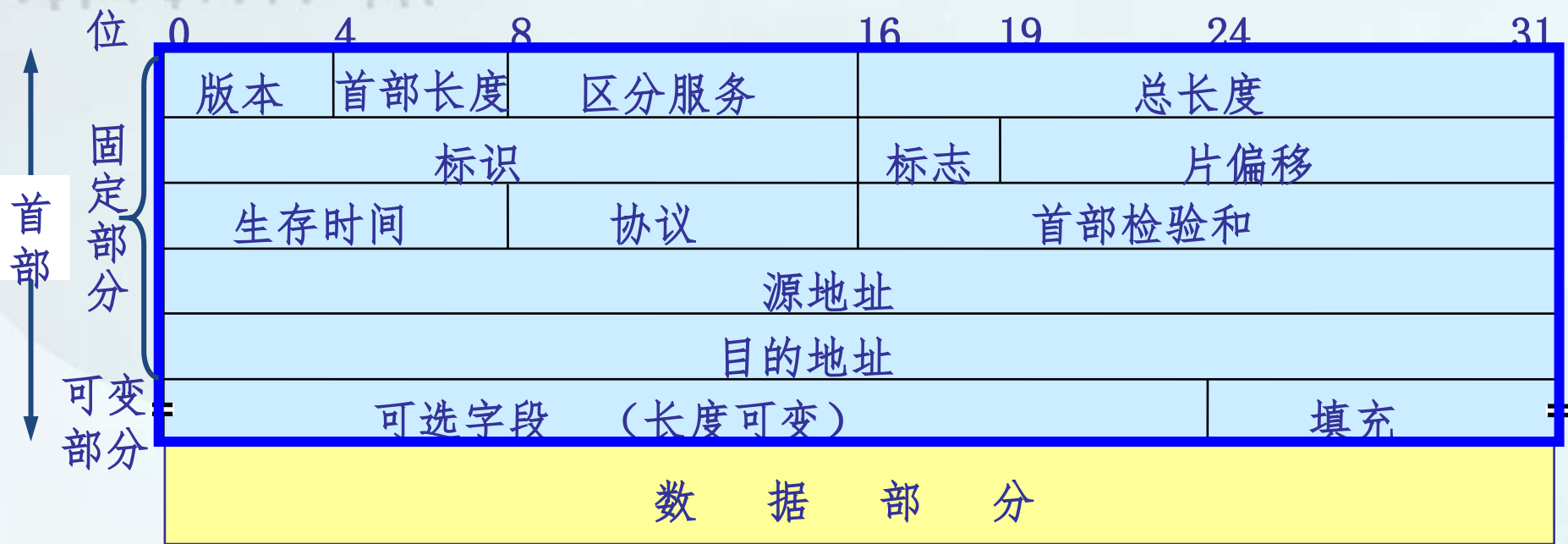
各层涉及协议






IP数据报

- 是IP协议的基本处理单元，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，长度可变
- IP 数据报首部的固定部分中的各字段包括
 - (1) 版本 (2) 首部长度 (3) 区分服务 (4) 总长度
 - (5) 标识 (6) 标志 (7) 片偏移 (8) 生存时间
 - (9) 协议 (10) 首部检验和 (11) 源地址 (12) 目的地
址



IP 数据报

发送在前

- 
- (1) 版本：占 4 位，指 IP 协议的版本；目前为 4 (即 IPv4)
 - (2) 首部长度：占 4 位，最大可表示 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。

- (3) 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。
- (4) 总长度：占 16 位，首部和数据长度之和，单位为字节；因此，数据报的最大长度为 65535 字节。
- (5) 标识(Identification)：占 16 位，它是一个计数器，用来产生数据报的标识，每产生一个数据报计数器加 1
- (6) 标志(flag)：占 3 位。
 - MF=1：后面“还有分片”；MF=0：最后一个分片
 - DF=0：允许分片；DF=1，该报文不能被分片，此时，若 IP 数据报的长度大于网络的 MTU 值，则该报文被丢弃，同时向源端报错



- (7) 片偏移：12 位，分片后，某个分片在原报文中**相对于用户数据字段的相对位置**；
- (8) 生存时间：8 位，记为 TTL (Time To Live)
 - 现在，TTL表示数据报在网络中**可通过的路由器数的最大**跳数。凡超过生存时间的项目就从高速缓冲中删除掉
- (9) 协议：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适当的处理进程
- (10) 首部检验和：16 位，只检验数据报的首部，不检验数据部分，以减少路由转发过程中，每个路由器的检验工作量
 - 不采用 CRC 检验码，而采用更简单的计算方法





路由转发

- (1) 从数据报的首部提取目的站的ip地址D，得出目的网络地址为N。
- (2) 若N就是与此路由器直接相连的某个网络地址，则这种交付为直接交付，即不需要再经过其它的路由器。这时就直接通过该网络将数据报交付给目的站D（这里包括将目的主机地址D转换为具体的硬件地址，将数据报封装为MAC帧，再发送此帧）；否则就是间接交付，执行（3）。



- (3) 若路由器表中有目的地址为**D**的特定主机路由，则将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，则将数据报传送给路由表中所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。



路由选择

- 距离的含义
 - 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”
- 路由表建立思想
 - 路由器开始工作时，只知道到直接连接的网络的距离（距离=1）
 - 以后，每一个路由器也和相邻路由器交换并更新路由信息
 - 最终，所有路由器都会知道到达本AS中任何一个网络的最短距离和下一跳路由器的地址
- 优点
 - RIP 协议的**收敛**(convergence)过程较快
 - 即，在AS中所有的结点都得到正确的路由选择信息的过程

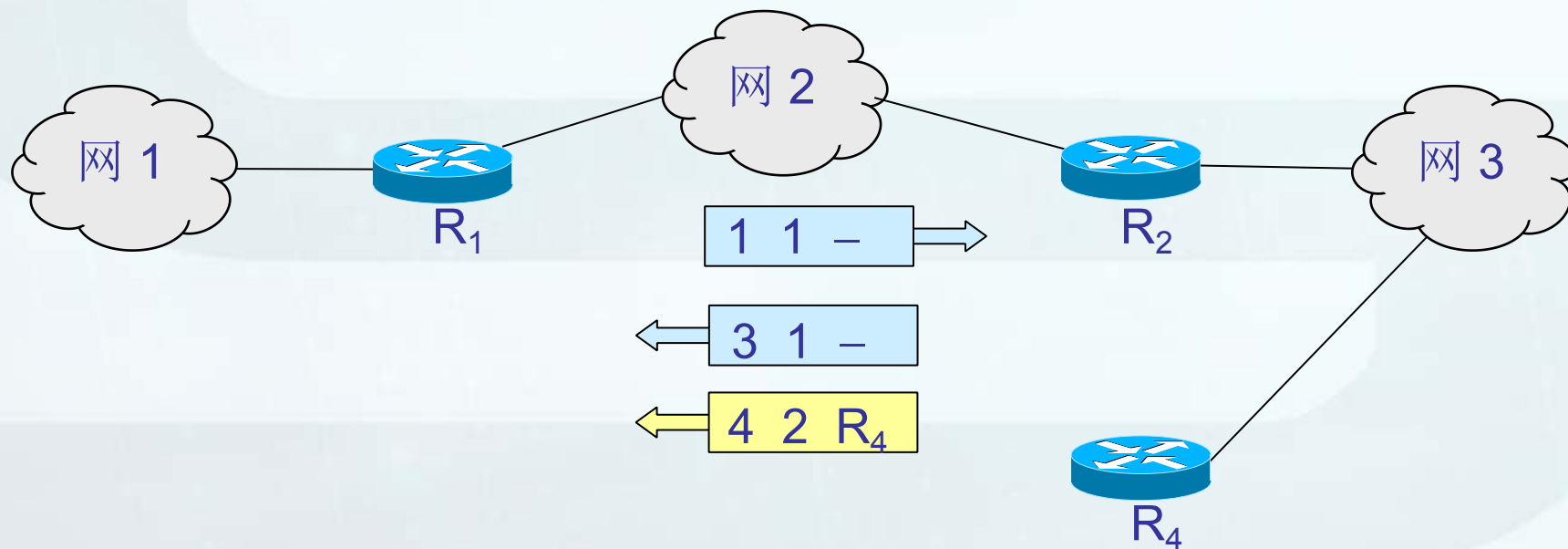
距离向量算法

- 1. 收到相邻路由器X的一个RIP报文
- 2. 预处理
 - 修改此RIP报文中的所有项目，“下一跳” \Rightarrow X，“距离”+ 1
- 3. 对修改后的RIP报文中的每一个项目：
 - 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 用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；
- 4. 若3分钟未收到相邻路由器的更新报文，此路由器不可达，距离=16
- 5. 返回

路由项的构成

例如: Net₁ 3 X

表示**当前路由器**, 通过下一跳 (路由器X), 到达Net₁的距离是3





THANKS