**Network**

**1. IP地址分类**

IPv4地址分为A、B、C、D、E五类，出去特殊作用的D、E两类，剩下的A、B、C三类地址是我们常见的IP地址段。A类地址的容量最大，可以容纳16777214个主机，B类地址可以容纳65534个主机，C类地址可以容纳254个主机。  
在这三类地址中，绝大多数的IP地址都是公有地址，需要向国际互联网信息中心申请注册。

对应三类IP地址:

A类：1.0.0.0 - 127.255.255.255 最后一个是广播地址。Mask: 255.0.0.0, (2^24) - 2=16777214台计算机 (除去一个网络地址和一个广播位).

B类：128.0.0.0 - 191.255.255.255最后一个是广播地址。Mask: 255.255.0.0, (2^16) - 2=65534台计算机.

C类：192.0.0.0 - 223.255.255.255最后一个是广播地址。Mask: 255.255.255.0, (2^8) - 2=254台计算机.

对应三类内网地址:

A类地址：10.0.0.0--10.255.255.255

B类地址：172.16.0.0--172.31.255.255

C类地址：192.168.0.0--192.168.255.255

**2. Ethernet 和 LAN (Local Area Network)**

以太网是采用Ethernet协议的一种网络，目前使用的绝大部分都是EthernettII协议，常见于局域网。  
局域网是根据网络规模划定的，一个学校内部网络可以是局域网，一个学校在本地有分校，那么主校和分校间构成城域网，也是一种局域网。

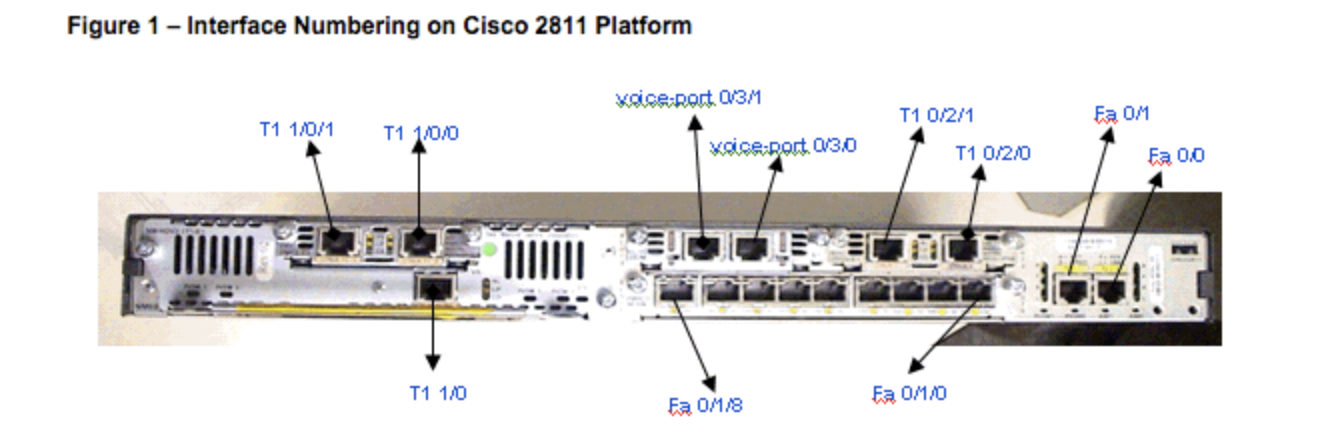
以太网是采用EthernetII协议的局域网，局域网内部还可以使用IEEE 802.3协议标准，所以只是大部分局域网是采用EthernetII协议的局域网，所以大家慢慢也就不做区分了，以太网约等于局域网。  
而EthernetII协议和IEEE 802.3协议其实是一种数据帧封装格式。

S0/0:

s0/0 is the abbreviation of serial port, which is the **external network interface** of serial port. When operating on the console, you can use interfece s0/0 or int s0/0

F0/0:

The first 0 represents the interface card in a router. Many routers support multiple interface cards, and each card can have multiple connections. The second number represents that actual interface on a given interface card. 0/1 refers to the number 1 interface on the number 0 card installed in the router. If the router has other cards, you could see interface numbers like 1/1.

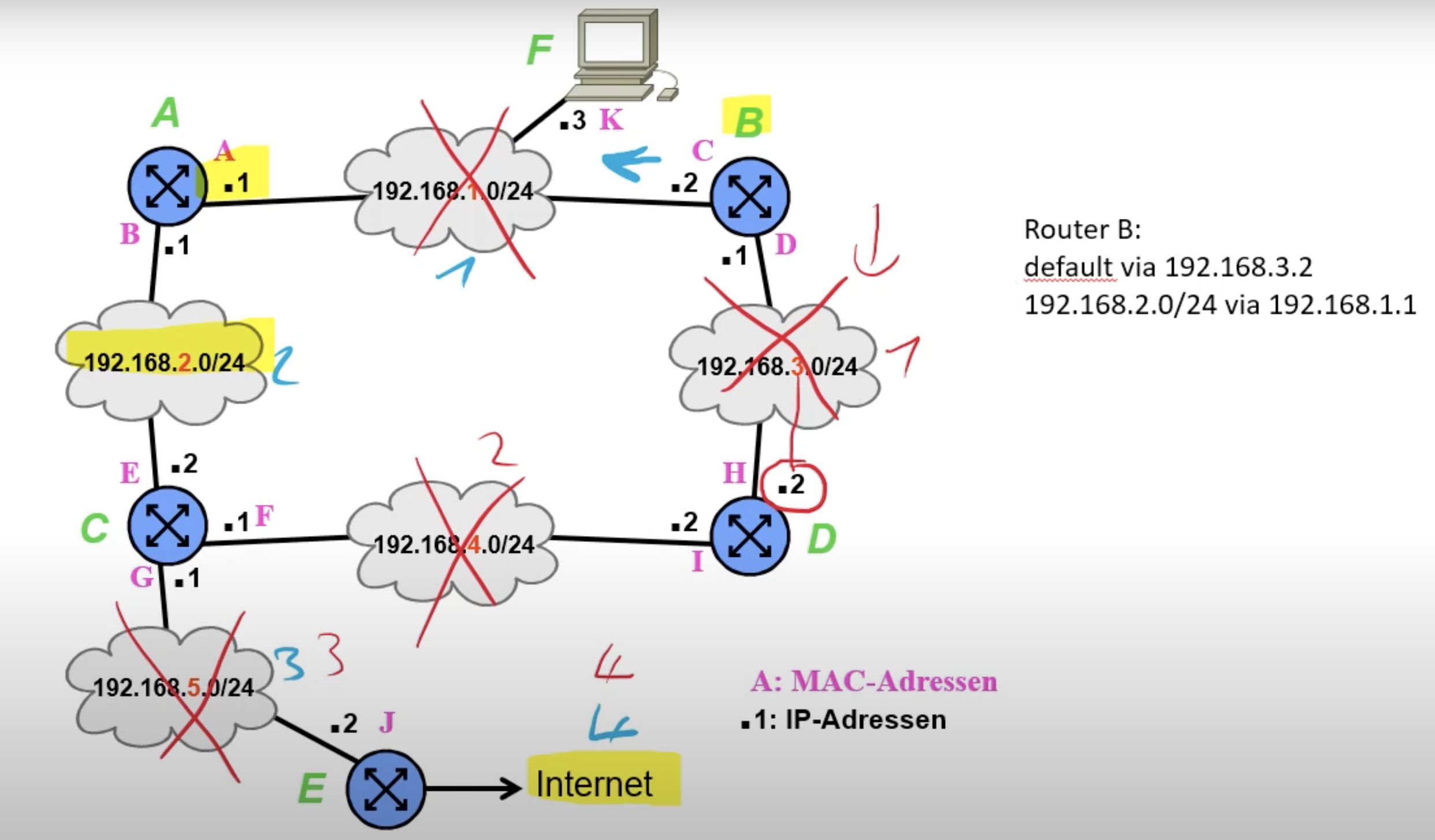


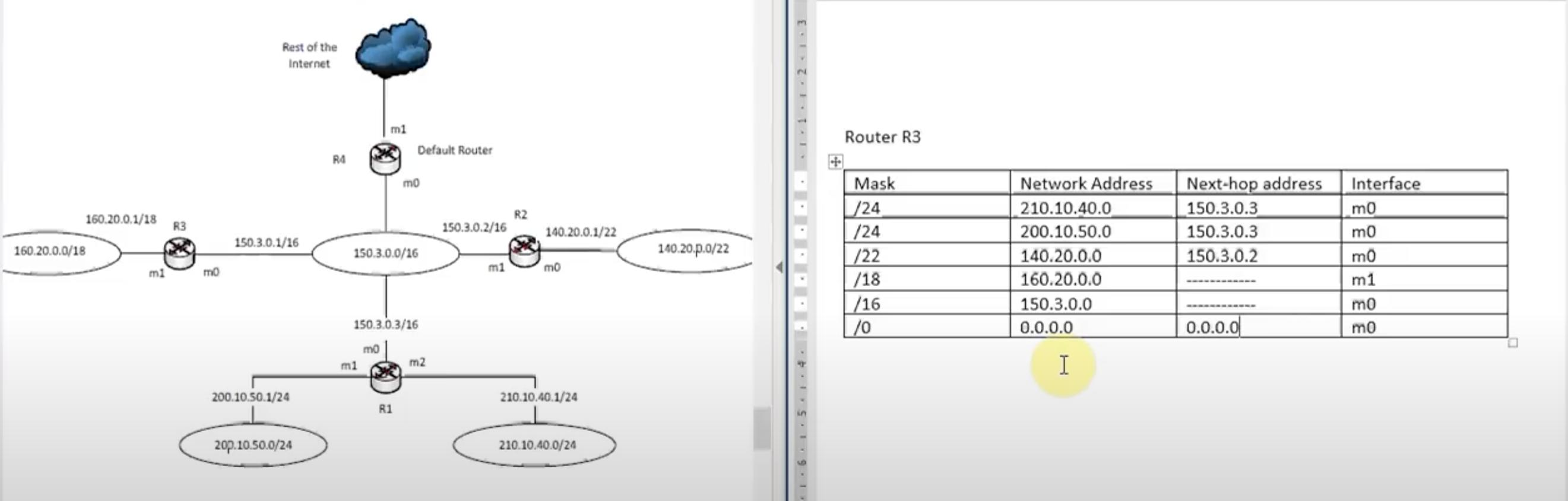
**3. DHCP (Application layer):**

以动态分配方式为客户主机分配ip地址上网**.**

**4. Routing Table**

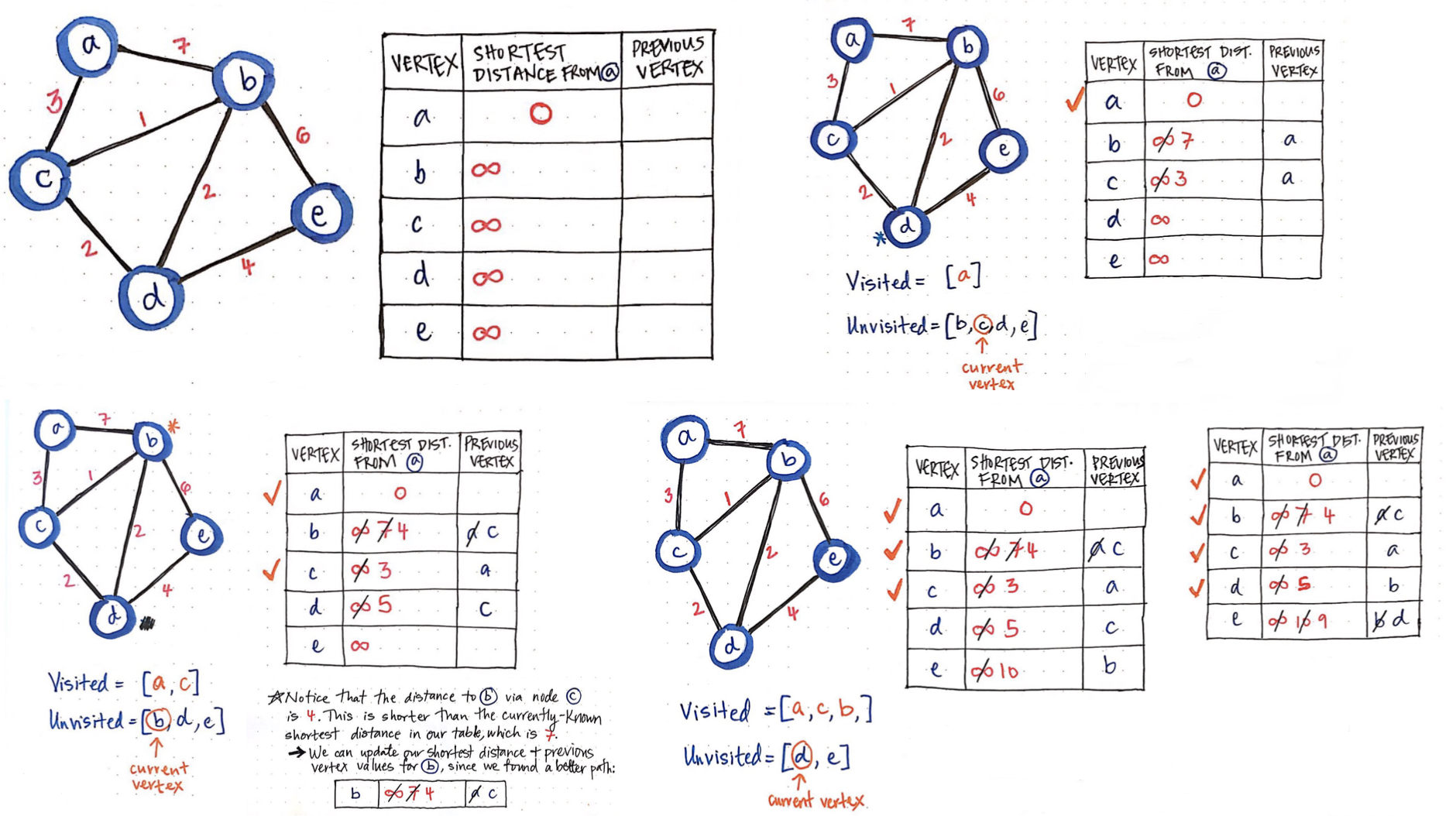
路由类似一个多出口的斗兽场(Router), 网关 (通常网关地址为本子网的第一台机器. B.S. 192.168.2.1) 则是通往另外一个异世界的大门, 与这个路由相连的异世界(Subnet)在table当中都被视作为local; 同样与这个异世界相连的另外的出口也有自己的网关地址, 这个大门与另外一个多出口的斗兽场 (Router)相连; 通常在table中被视为Next-Hop. 这个Next-Hop地址在table当中表示通往其他异世界大门地址. 与异世界大门相连的是另外的多出口斗兽场, 通过斗兽场(Router)后才能到达新的异世界(Subnet).

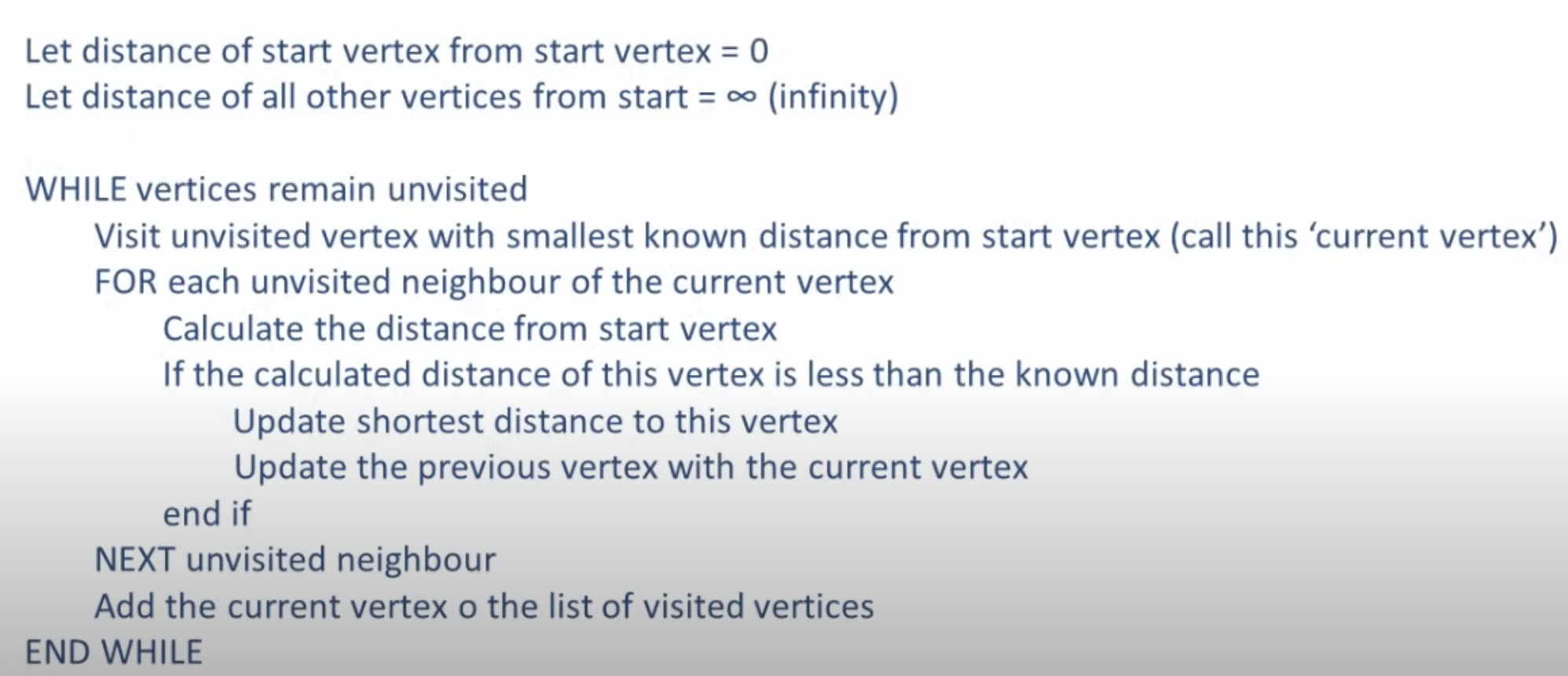


****

**Dijkrstra’s Shortest Path Algorithm**

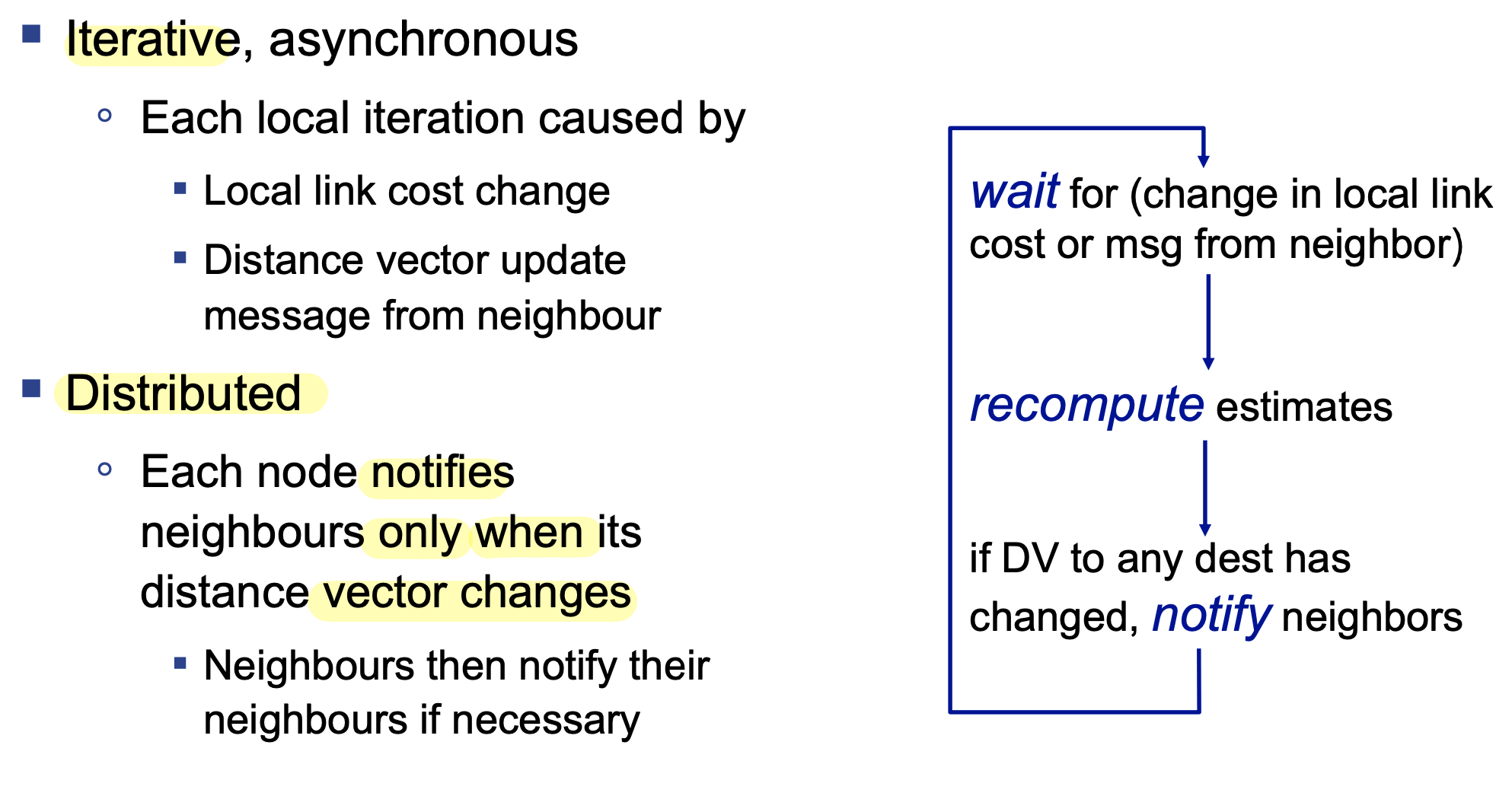
遍历每个unvisited vertex,依次从shortest distance的vertex出发, 不断更新table中的连接路径最小值, 直到所有vertex visited.

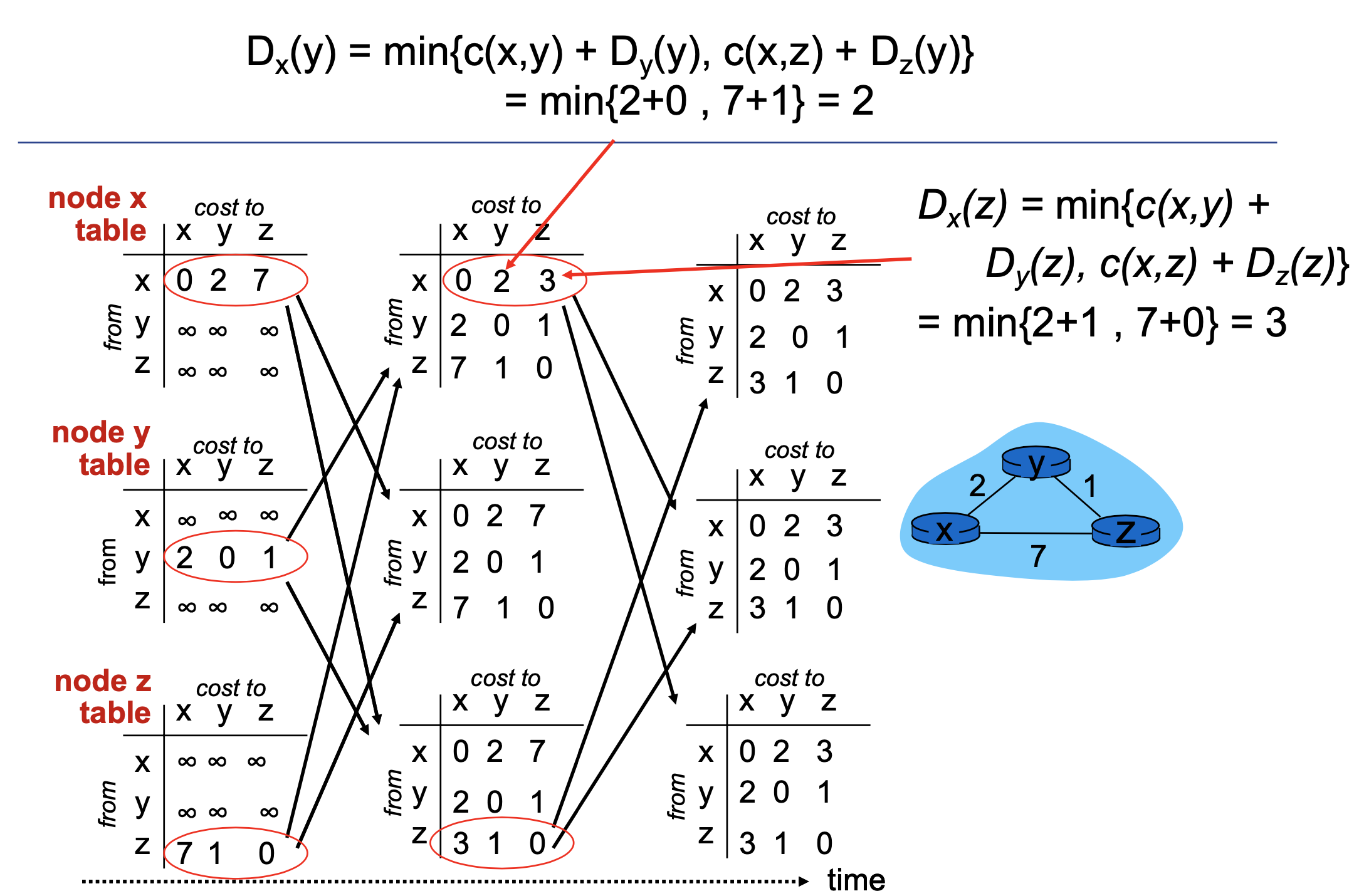




**Distance Vector Algorithm**

Bellman-Ford Equation: Dx(y) = minv {c(x, v) + Dv(y)}

****

****