

計算機網路概論



網路互連技術

© All rights reserved. No part of this publication and file may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without prior written permission of Professor Nen-Fu Huang (E-mail: nfhuang@cs.nthu.edu.tw).

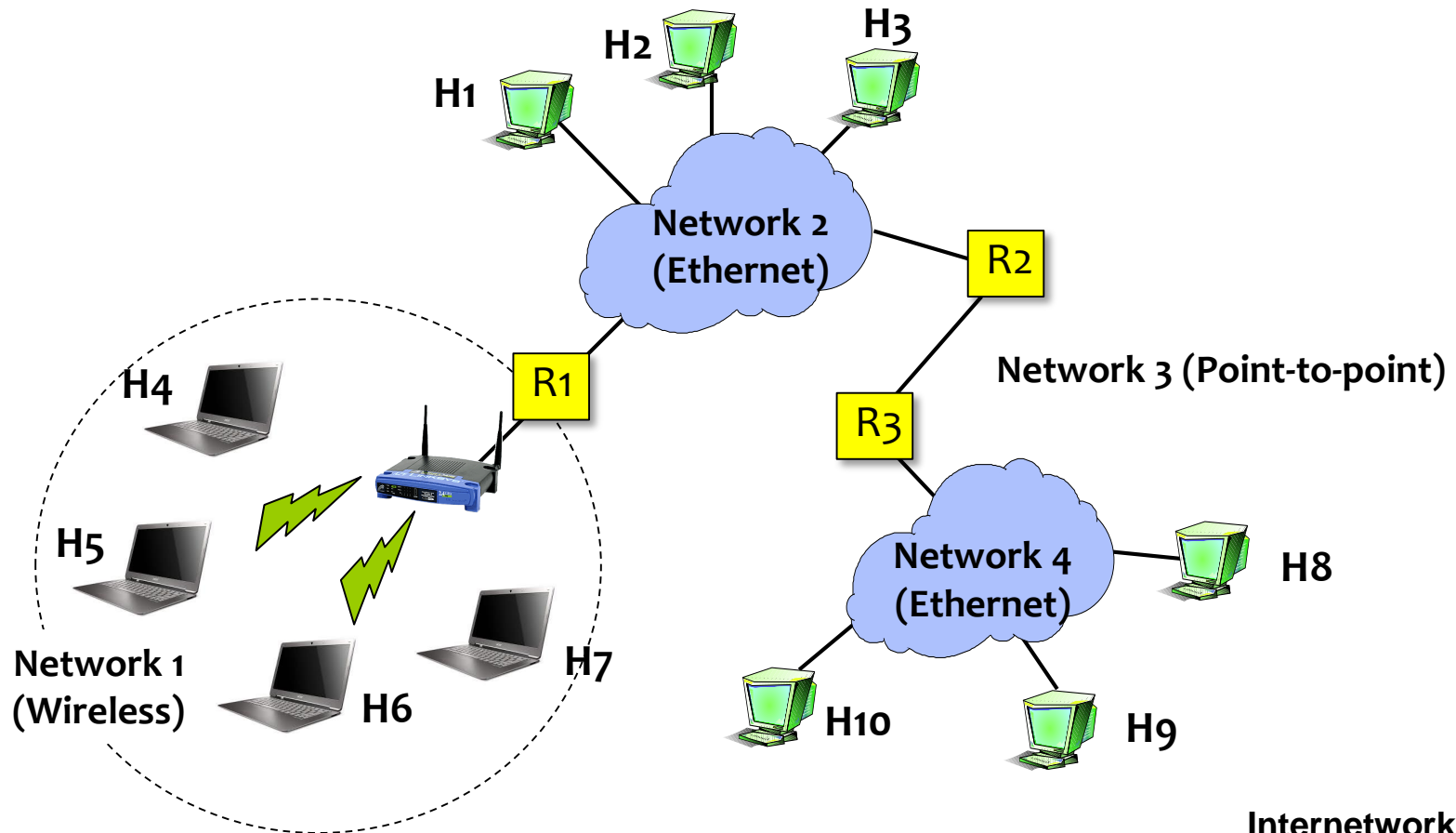
大綱

- 網路互連簡介
- IP 與路由器
- IP 子網路 (Subnetting)
- 無分級網址 (Classless Addressing)
- 路由協議 (Routing protocols)
- 距離向量協議 (Distance Vector protocol)
- 鏈結狀態協議 (Link State protocol)

互連網路

■ 何謂互連網路？

- 一個將許多網路任意連結起來以提供主機對主機的封包傳遞服務

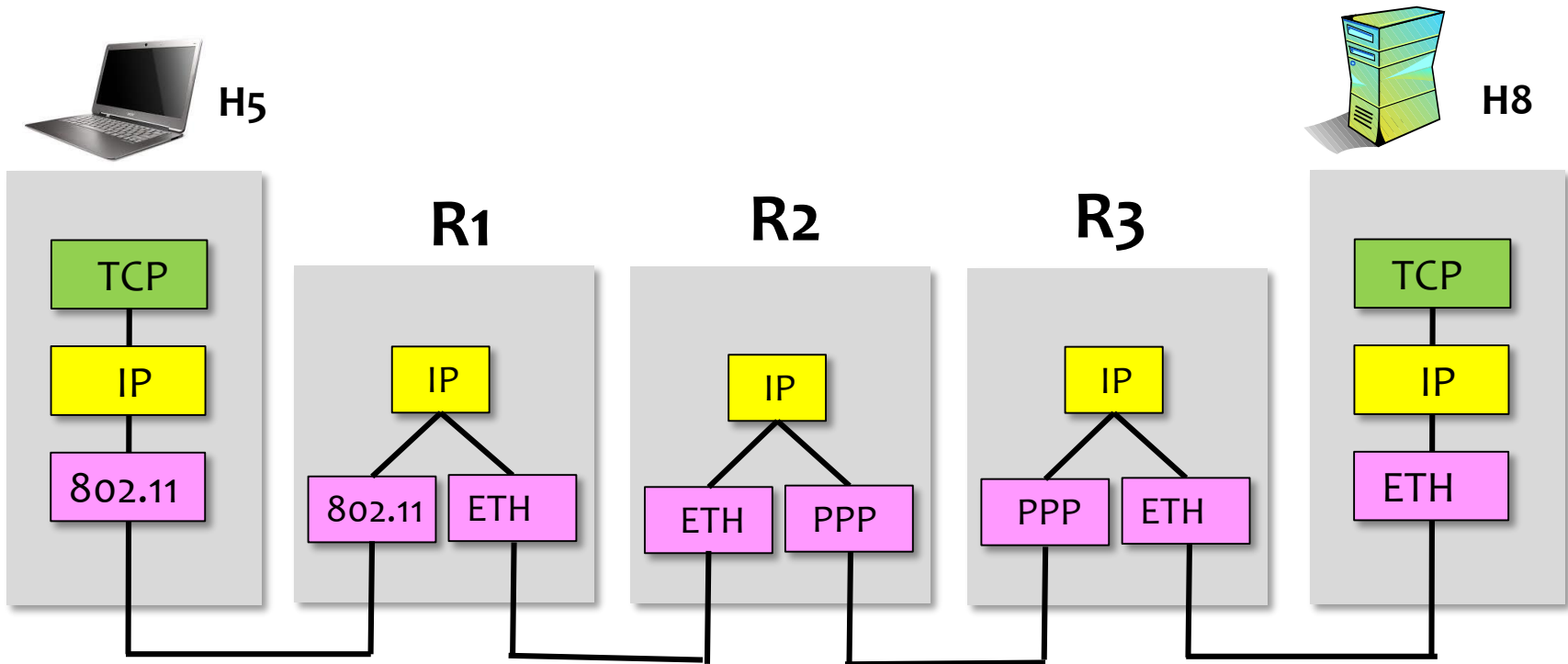


互連網路

■ 何謂 IP 協議？

- IP 代表 Internet Protocol
- 它是今日用來提供大規模異質網路互連的關鍵基礎
- 它運行在互連網路中的所有節點上
- 它定義了基礎架構使得這些節點與網路能運行得像一個互連在一起網路

互連網路



一個簡單的互連網路顯示各層協議的關係

IP 服務模式

■ 封包傳遞模式

- 以無連線模式來傳遞資料
- 盡力式遞送 (不可靠服務)
 - ▶ 封包可能遺失
 - ▶ 封包傳遞順序可能紊亂
 - ▶ 封包可能重複傳遞
 - ▶ 封包遞送可能被延遲很長一段時間

■ 全球定址機制

- 提供一種可以識別網路上所有主機的方法

第三層路由器如何運作？

- 第三層路由器使用存儲轉發 (store and forward) 的方式來轉送收到的 IP 封包 (datagrams).
 - IP 位址查詢 (查詢路由表, 此路由表乃經由路由協議, 如 RIP, OSPF, BGP 等等, 所建構而成)

- IP/MAC 位址對應表

IP	MAC
IP(A)	MAC(A)
IP(B)	MAC(B)
IP(Y)	MAC(Y)
IP(X)	MAC(X)

IP	Next
140.114.77.0	Directly
140.114.78.0	Directly
140.114.79.0	Router Z

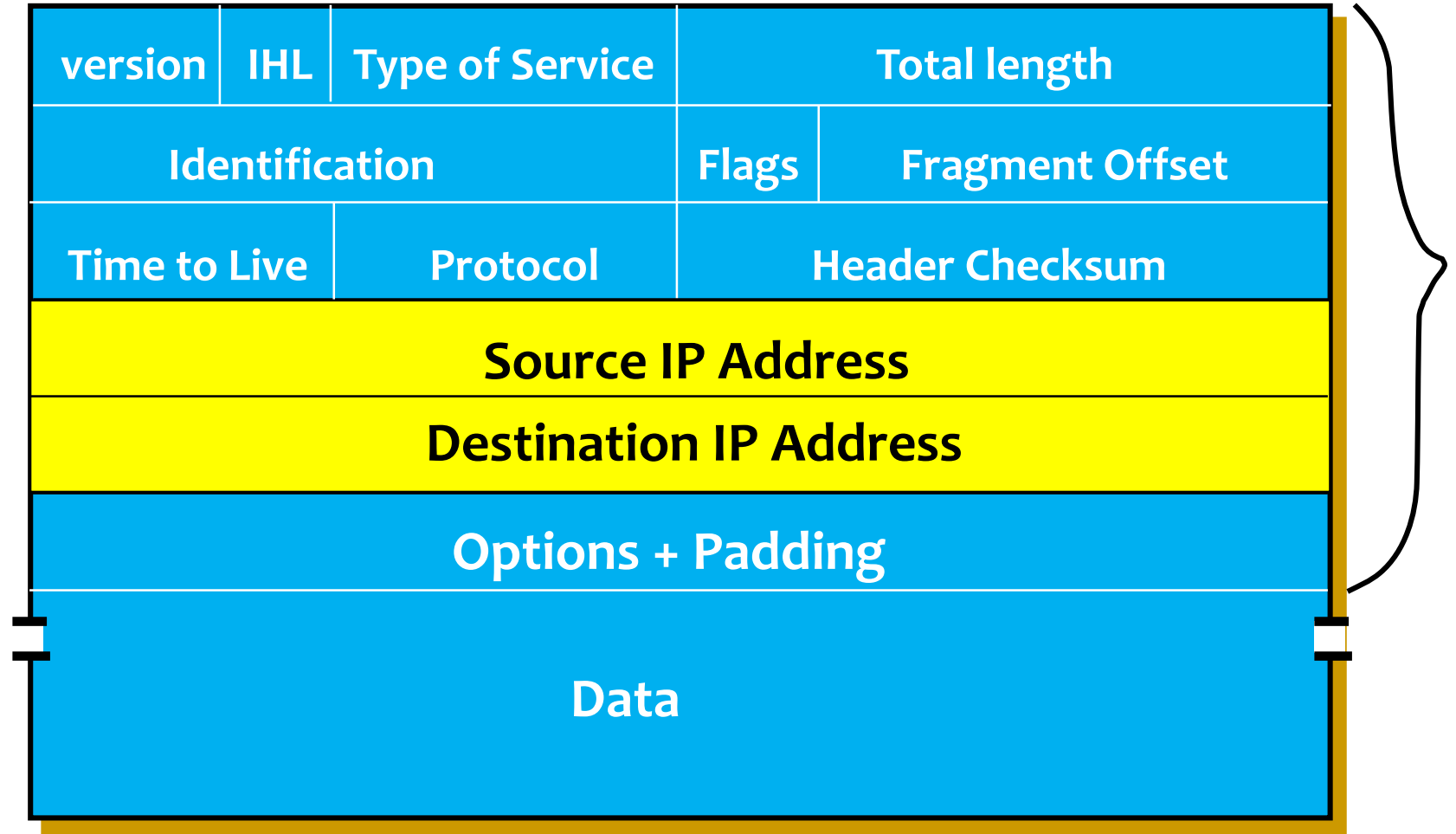
第三層路由器如何運作？

- 如果目的地的IP位址存在於**路由表**，則轉送 IP 封包至路由表紀錄的下一站，否則從預設埠轉送出去。
- 新的路由器架構使用第三層交換器晶片與 IP 位址查詢晶片 (採用快速硬體查詢技術)
- 線速轉送設計可達 Gbps, 10Gbps, 100Gbps, ...
- 但不具隨插即用特色

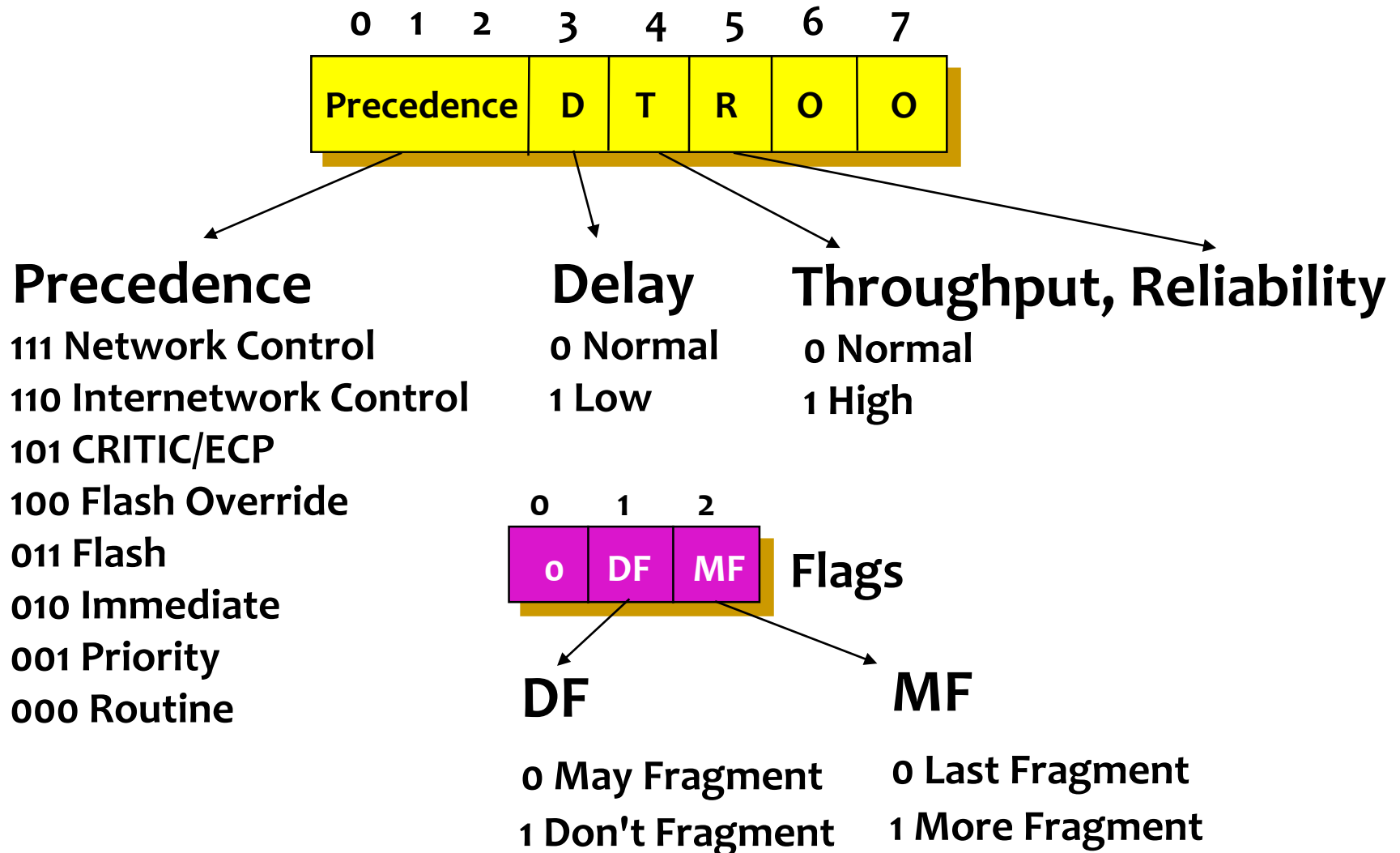
大綱

- 網路互連簡介
- **IP 與路由器**
- **IP 子網路 (Subnetting)**
- **無分級網址 (Classless Addressing)**
- **路由協議 (Routing protocols)**
- **距離向量協議 (Distance Vector protocol)**
- **鏈結狀態協議 (Link State protocol)**

IP 封包標頭格式



IP 標頭中的服務型態 (ToS)



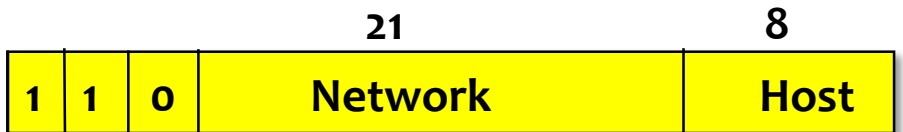
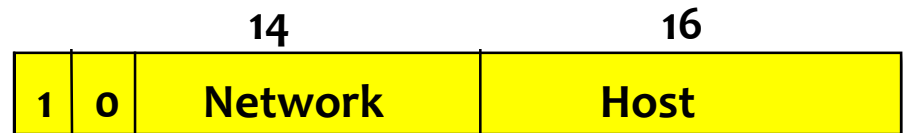
IP 位址

■ 特性

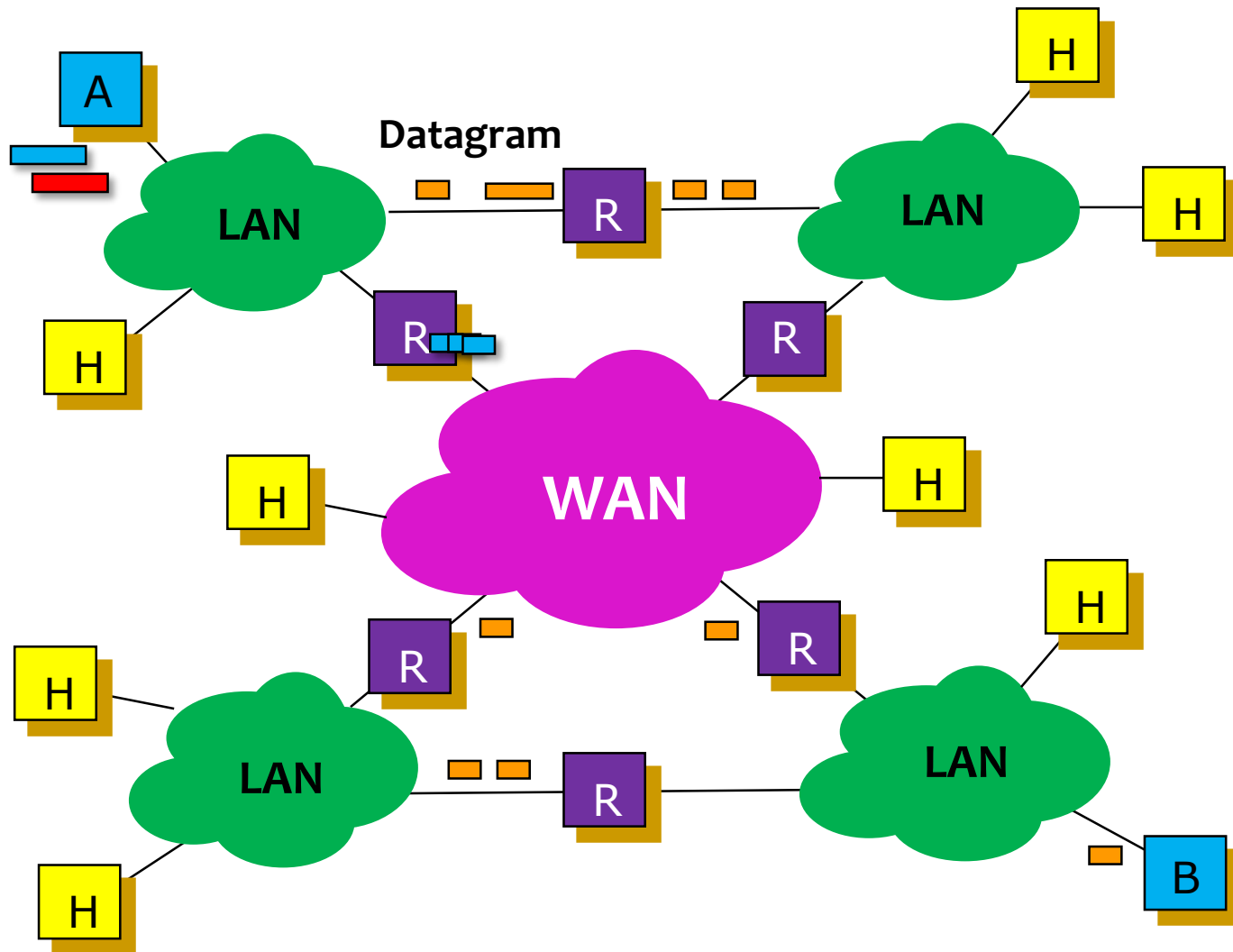
- 具全球唯一性的 32 位元位址
- 階層性結構: 網路位址 + 主機位址
- 約 40 億個 IP 位址
- Class A 型態 (1/2)
- Class B 型態 (1/4)
- Class C 型態 (1/8)

■ “點” (dot) 標記法

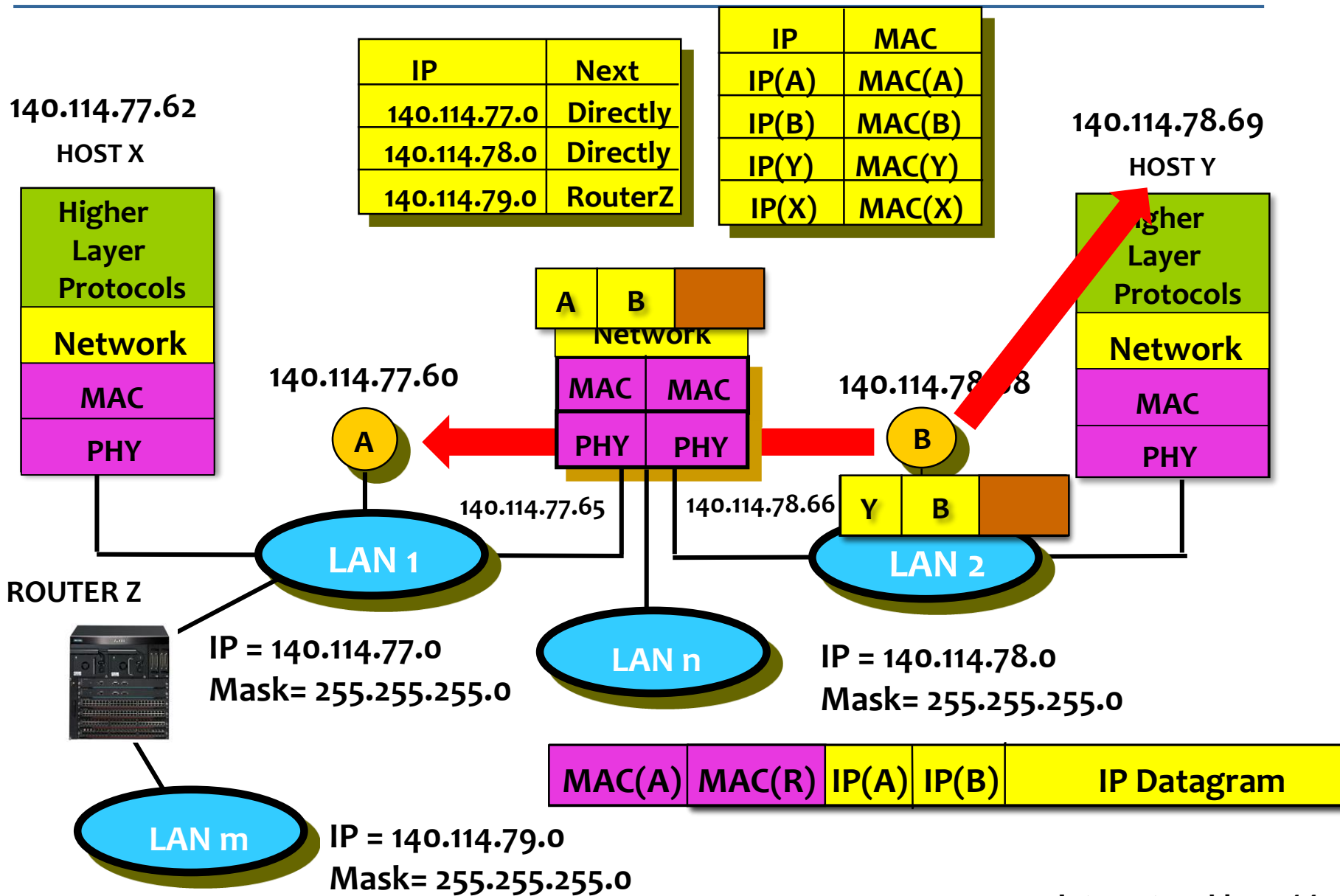
- 10.3.2.4
- 128.96.33.81
- 192.12.69.77



互聯網上如何遞送封包？



路由器



區域網路內 與 區域網路間的通訊

■ B -> Y (區域網路內的通訊):

- 直接傳送訊框至目的地

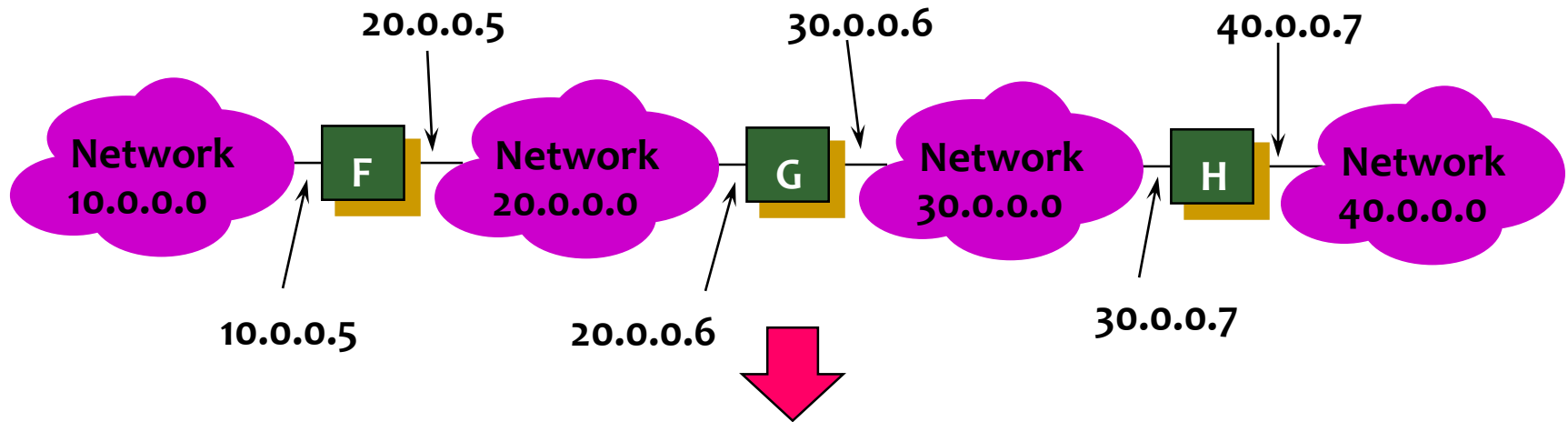


■ B -> A (區域網路間的通訊):

- 先傳送訊框至相連接的路由器。
- 路由器會轉送至目的地。



互聯網路由範例



到此目的地網路
上的主機

傳送至此位址
的路由

20.0.0.0	Deliver Direct
30.0.0.0	Deliver Direct
10.0.0.0	20.0.0.5
40.0.0.0	30.0.0.7

- 路由表 (Routing Table)

路由器 IP 封包轉送

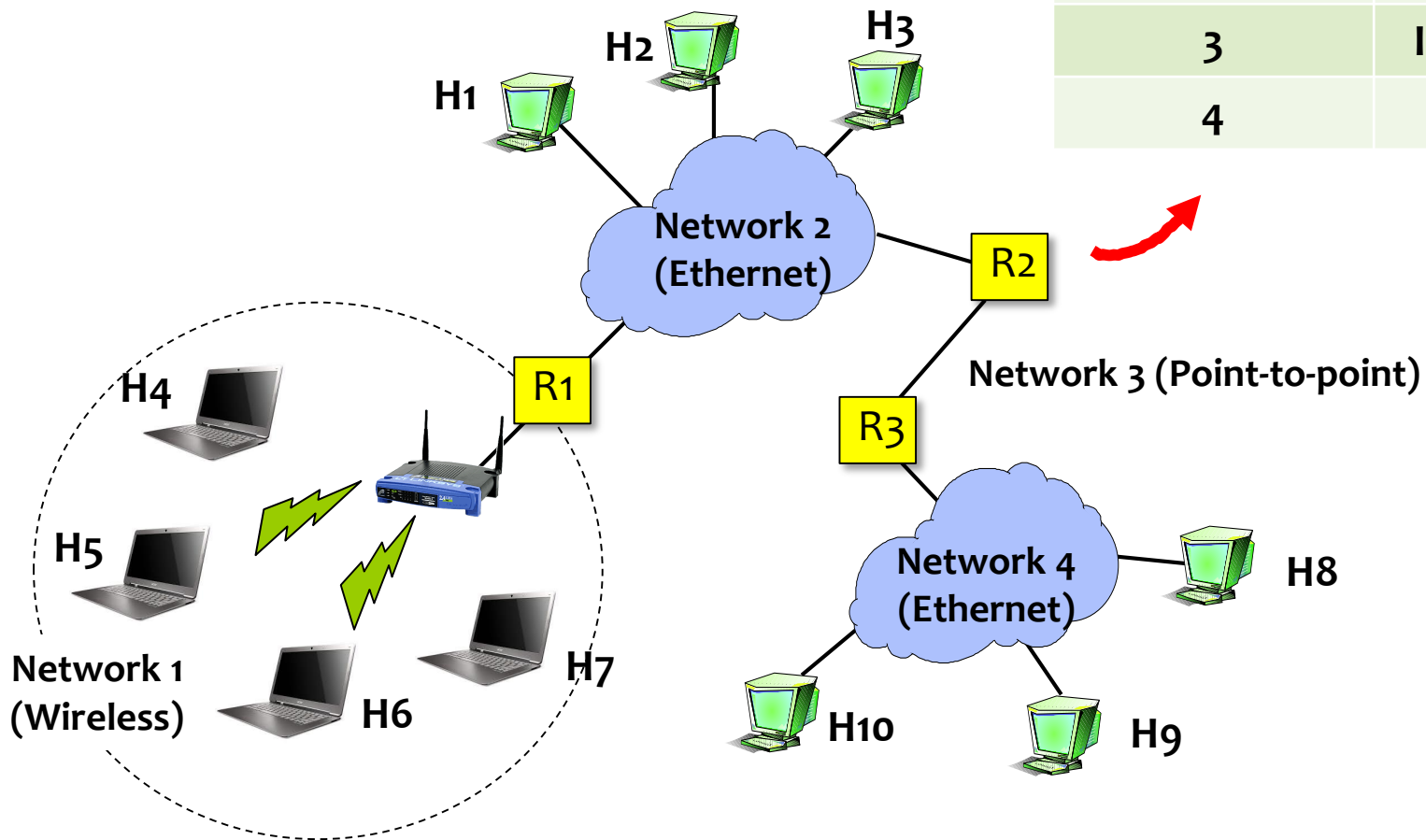
■ 策略

- 每一個封包都含有目的地 IP 位址
- 如果路由器與目的地主機的網路**直接連結**，就直接發送給目的地主機
- 如果路由器與目的地主機的網路**不直接連結**，發送給某一個路由器繼續轉送
- 路由表中有登錄**網路編號與對應的下一站**
- **每個主機有一個預設路由器**
- 每個路由器都一個路由表

路由器 IP 封包轉送範例

■ 範例 (路由器 R2)

NetworkNum	NextHop
1	R1
2	Interface 1
3	Interface 0
4	R3



IP 封包的切割與組合

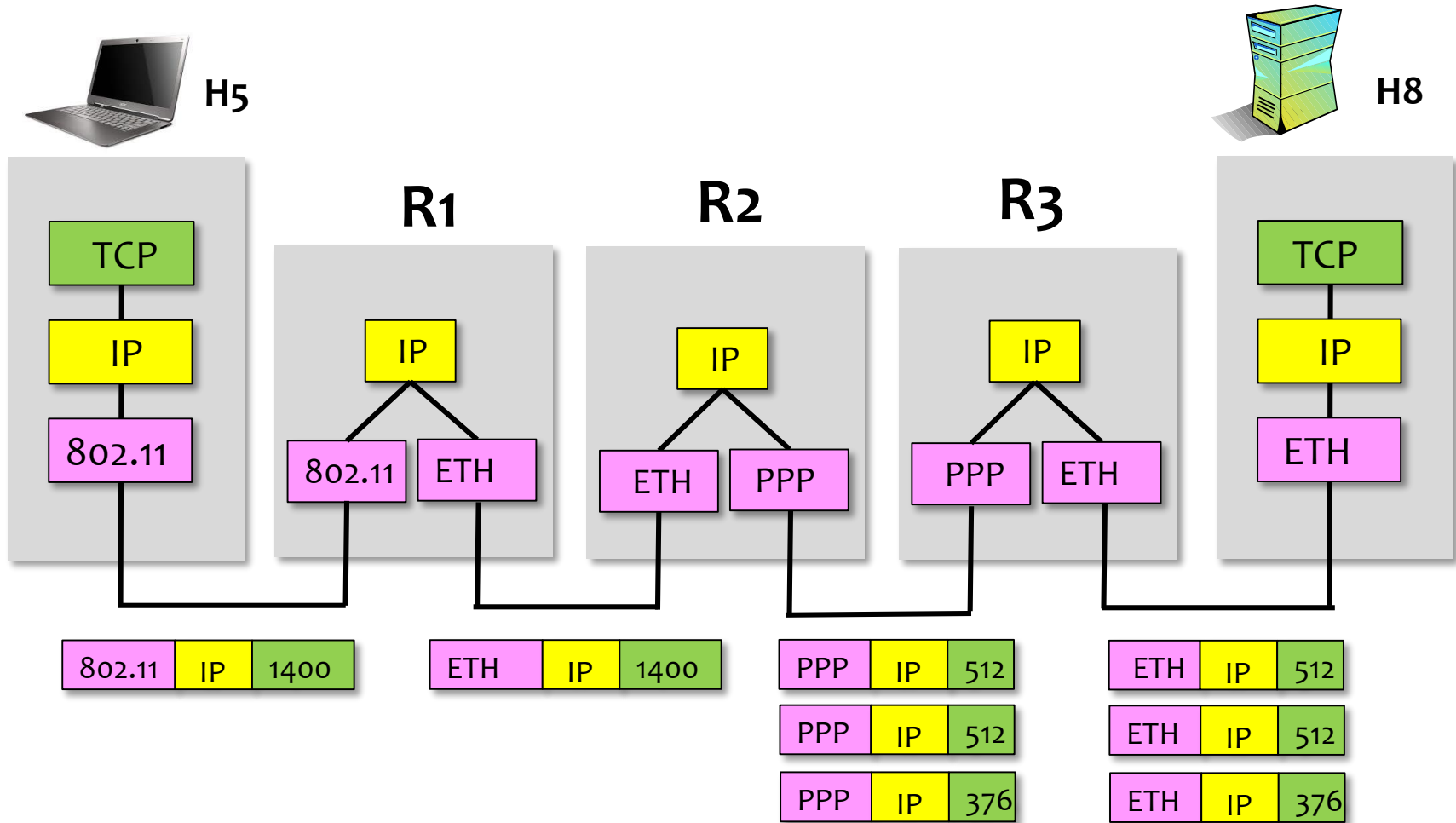
■ 可傳輸單元最大值 (MTU)

- Ethernet 乙太網路 (1518 bytes),
- IEEE 802.11 無線區域網路 (2312 bytes)
- FDDI 光纖網路 (4500 bytes)

■ 策略

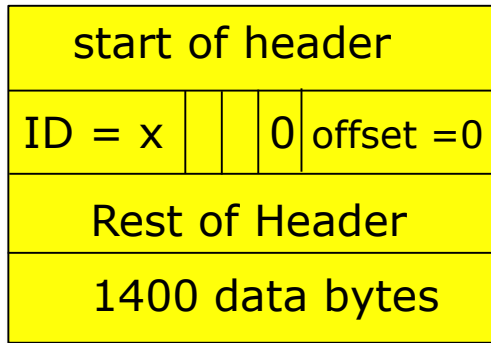
- 當路由器欲將收到的封包轉送到區域網路但是該封包大小超過該區域網路的 **MTU**，路由器會將此封包切割成數個小封包 (稱為片段)
- 接收端主機的 IP 協議負責將片段組合為原封包
- 所有屬於原封包之片段會攜帶相同的**辨識碼**
- 每一個片段都是一個完整的封包(只是較小)
- 任一片段如果遺失, 則 IP 協議直接丟棄該封包

IP 封包的切割與組合範例

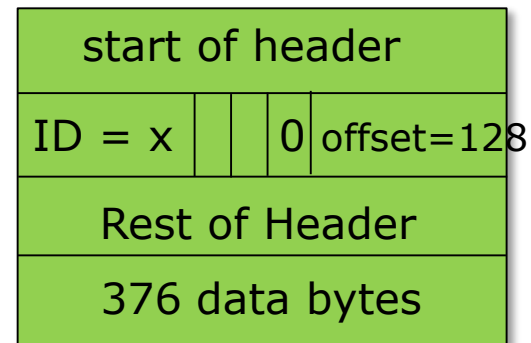
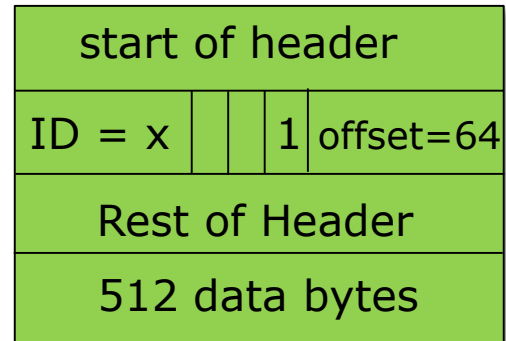
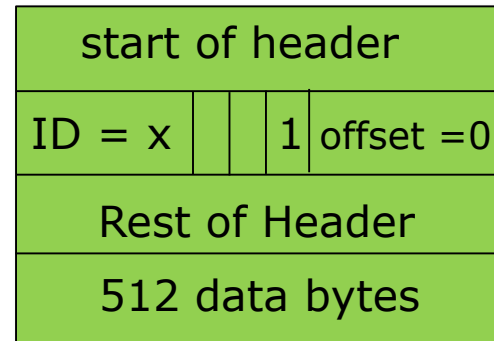
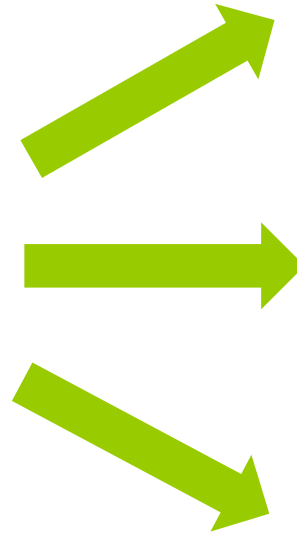


IP 封包經過一連串實體網路範例

IP 封包的切割與組合範例



(a) 原封包



(b) 切割成三個片段

路由器特色

■ 網路層路由

- 與網路層的通訊協議相關
- 過濾 MAC 的廣播訊框與群播訊框
- 能輕易支持不同的傳輸媒介 (光纖, 電纜, 雙絞線....)
- 封包之切割與重組 (重組只在主機進行)
- 使用 IP 位址與資訊來過濾封包
- 可支援流量計費

■ 端點主機與路由器之間的直接溝通

- 路由器功能強大可彈性設定, 但設定技術門檻較高
- 可調節不同傳輸媒介的網速差異
- 可支援擁塞控制與避免機制

路由器特色(接續)

■ 路由協議

- 以任意的拓樸架構將第三層網路互相連結起來
- 決定封包的路由方向
- 靜態路由
- 動態路由協議
 - ▶ RIP 路由信息協議: Routing Information Protocol
 - ▶ OSPF 開放式最短路徑優先: Open Shortest Path First
- 有其他替代的路由以提供可靠性

■ 路由器管理

- 支援問題偵錯功能

橋接器與路由器的差異

橋接器	路由器
第二層網路設備	第三層網路設備
與第三層通訊協議無關	與第三層通訊協議相關
自動化的位址學習/過濾	需要管理來設定位址、介面及路由
無法過濾 MAC 廣播與群播訊框	會過濾 MAC廣播與群播訊框
成本較低	成本較高
無流量/擁塞控制	有流量/擁塞控制
安全性較低	安全性較高
使用者不須知道橋接器的存在	使用者須知道路由器的存在
適用於簡單及小型網路環境	用於廣域網路 及較大的網路環境
無訊框切割與重組	有封包切割與重組
基於擴張數的路由	最佳化路由及負載分擔
隨插即用	需要中央權限管理

大綱

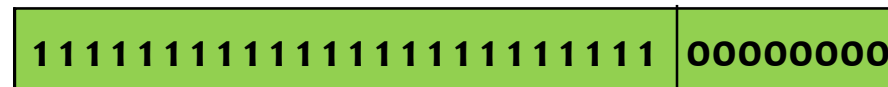
- 網路互連簡介
- IP 與路由器
- **IP 子網路 (Subnetting)**
- 無分級網址 (Classless Addressing)
- 路由協議 (Routing protocols)
- 距離向量協議 (Distance Vector protocol)
- 鏈結狀態協議 (Link State protocol)

子網路

- 對位址/路由階層架構增加另一個層級: **子網路 (subnet)**
- **子網路遮罩 (subnet masks)**: 針對 class A 及 class B 的主機位址部分定義不同的區分方式, 也就是規範子網路的大小



Class B address



Subnet Mask (255.255.255.0)

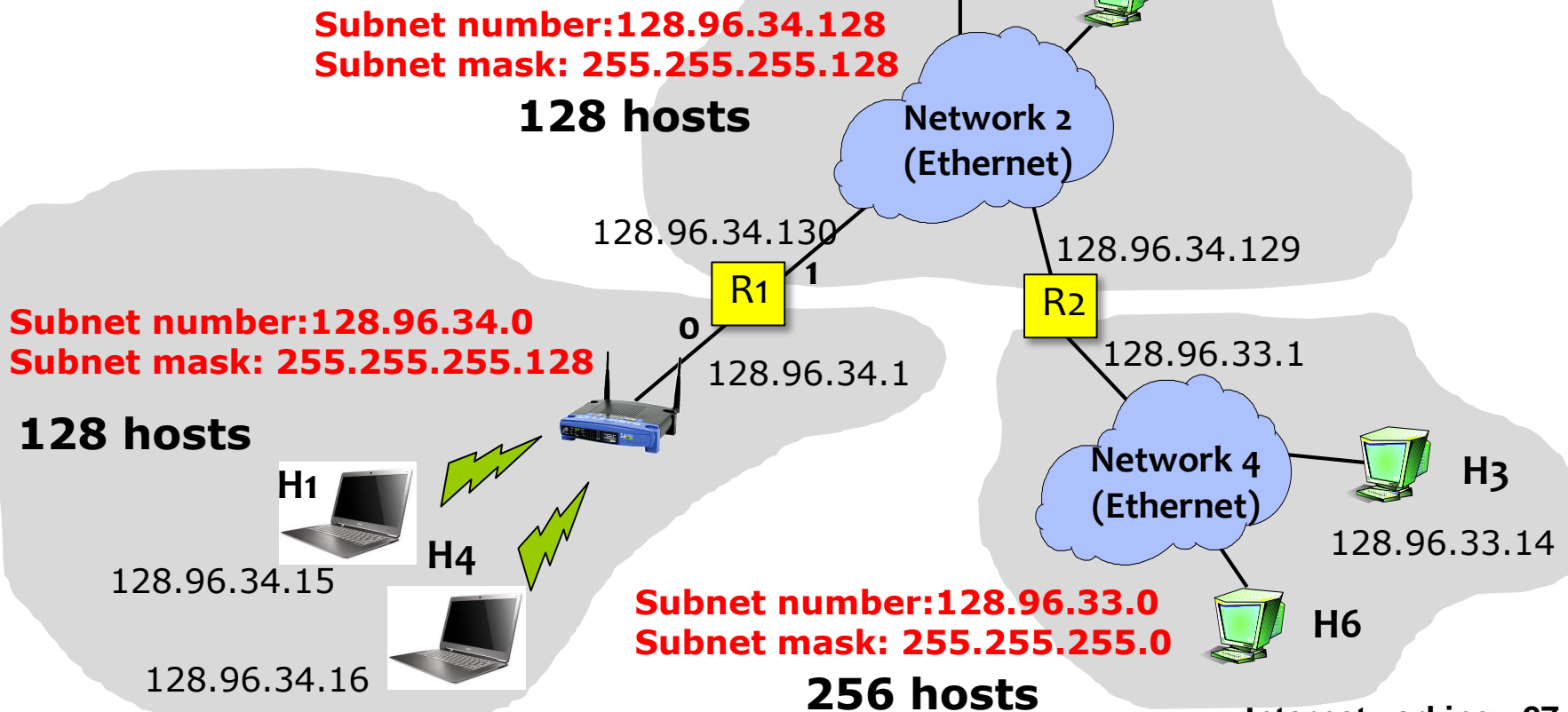


Subnetted address

子網路範例

路由器 R1 的路由表

SubnetNumber	SubnetMask	NextHop
128.96.34.0	255.255.255.128	Interface 0
128.96.34.128	255.255.255.128	Interface 1
128.96.33.0	255.255.255.0	R2



子網路

封包轉送演算法

SubnetNumber	SubnetMask	NextHop
128.96.34.0	255.255.255.128	Interface 0
128.96.34.128	255.255.255.128	Interface 1
128.96.33.0	255.255.255.0	R2

D = destination IP address

for each entry \langle SubnetNum, SubnetMask, NextHop \rangle

$D1 = (\text{SubnetMask}) \text{ AND } (D)$

if **$D1 = \text{SubnetNum}$**

if NextHop is an interface

deliver datagram directly to destination

else

deliver datagram to NextHop (a router)

子網路範例

路由器 R1 的路由表

範例 1

SubnetNumber	SubnetMask	NextHop
128.96.34.0	255.255.255.128	Interface 0
128.96.34.128	255.255.255.128	Interface 1
128.96.33.0	255.255.255.0	R2

D = 128.96.34.15 (H1)

D1 = SubnetMask & D = 255.255.255.128 1000 0000

128. 96. 34. 15 0000 1 1 1 1

128. 96. 34. 0 0000 0000

D1 = SubnetNum 128. 96. 34. 0 → Interface 0

子網路範例

路由器 R1 的路由表

範例 2

SubnetNumber	SubnetMask	NextHop
128.96.34.0	255.255.255.128	Interface 0
128.96.34.128	255.255.255.128	Interface 1
128.96.33.0	255.255.255.0	R2

D = 128.96.34.131 (H5)

D1 = SubnetMask & D = 255.255.255.128 1000 0000

128. 96. 34. 131 1000 0011

128. 96. 34. 128 1000 0000

D1 = SubnetNum 128. 96. 34. 128 → Interface 1

子網路範例

範例 3

路由器 R1 的路由表

SubnetNumber	SubnetMask	NextHop
128.96.34.0	255.255.255.128	Interface 0
128.96.34.128	255.255.255.128	Interface 1
128.96.33.0	255.255.255.0	R2

D = 128.96.33.14 (H3)

D1 = SubnetMask & D = 255.255.255.0 0000 0000

128. 96. 33. 14 0000 1110

128. 96. 33. 0 0000 0000

D1 = SubnetNum 128. 96. 33. 0 → R2

子網路

注意事項

- 如果路由表中找不到對應的內容，送給預設的路由器
- 子網路遮罩不一定全是連續的1
- 一個實體網路能細分成數個子網路
- 在互聯網的其他地方看不到這些子網路

大綱

- 網路互連簡介
- IP 與路由器
- IP 子網路 (Subnetting)
- 無分級網址 (Classless Addressing)
- 路由協議 (Routing protocols)
- 距離向量協議 (Distance Vector protocol)
- 鏈結狀態協議 (Link State protocol)

無分級網址

- 無分級網域間路由 (Classless Inter-Domain Routing, 簡稱 CIDR)
 - 這是處理互聯網上兩個有關**規模問題**的一種技術
 - ▶ 越來越多的連網設備造成路由表**不斷變大**
 - ▶ **32-bit IP** 位址不敷使用

無分級網址

- 位址分配效率問題

- ▶ 此方式被提出是因為 IP 位址被分成 class A, B 和 C 三種型式
- ▶ 迫使我們僅能分發這三種固定大小, 但差異極大的網路位址空間
 - 一個含有2個主機的網路需要一個 class C 的位址
 - » 位址分配效率 = $2/255 = 0.78$
 - 一個含有256個主機的網路需要一個 class B 的位址
 - » 位址分配效率 = $256/65535 = 0.39$

無分級網址

- IP 位址的枯竭主要在於 **Class B** 的網段已經耗盡
- 解決方法
 - 除非一個自治系統 (**Autonomous System, AS**) 提出它們需要的位址數量接近 64K，否則不分發 **class B** 的位址出去
 - 取而代之地，給他們適量的 **class C** 網段
- 此解決方法會有何問題？
 - 造成路由器的路由表需要大量的儲存空間

無分級網址

- 舉例來說，如一個自治系統有16個 class C 的網段
 - 即使到這些網段的路徑都一樣，每個互聯網上骨幹路由器針對此自治區仍需要在其路由表上建置 16個項目
- 如果我們分發一個 class B 的位址空間給這個自治系統
 - 路由器的路由表只需儲存一個項目
 - 但是位址使用效率不佳 = $16 \times 255 / 65,536 = 6.2\%$

無分級網址

- **CIDR 使用累積路由(aggregate routes)**
 - 在路由表中使用**單一項目**來告訴路由器如何**到達大量不同的網段**
 - 打破原本位址分級的界線

無分級網址

- 舉例說明，一個自治系統有16個 class C 的網段
- 取代分發16個隨機網段給該自治系統，我們分發**連續的 class C 網段**給它
- 假設分發的網段是從 192.4.16 到 192.4.31
- 我們可以發現，這些網段的前20個位元都相同 (**11000000 00000100 0001**)
 - 我們已經產生一個 20 位元的網路編號 (介於 class B 的 16 位元與 class C 的 24 位元之間)
- 這些分發的網段需要有**相同的前綴**

無分級網址

■ 192.4.16	1100 0000	0000 0100	0001	0000
■ 192.4.17	1100 0000	0000 0100	0001	0001
■ 192.4.18	1100 0000	0000 0100	0001	0010
■ 192.4.19	1100 0000	0000 0100	0001	0011
■ 192.4.20	1100 0000	0000 0100	0001	0100
■ 192.4.21	1100 0000	0000 0100	0001	0101
■ 192.4.22	1100 0000	0000 0100	0001	0110
■ 192.4.23	1100 0000	0000 0100	0001	0111
■ 192.4.24	1100 0000	0000 0100	0001	1000
■ 192.4.25	1100 0000	0000 0100	0001	1001
■ 192.4.26	1100 0000	0000 0100	0001	1010
■ 192.4.27	1100 0000	0000 0100	0001	1011
■ 192.4.28	1100 0000	0000 0100	0001	1100
■ 192.4.29	1100 0000	0000 0100	0001	1101
■ 192.4.30	1100 0000	0000 0100	0001	1110
■ 192.4.31	1100 0000	0000 0100	0001	1111



192.4.16/20

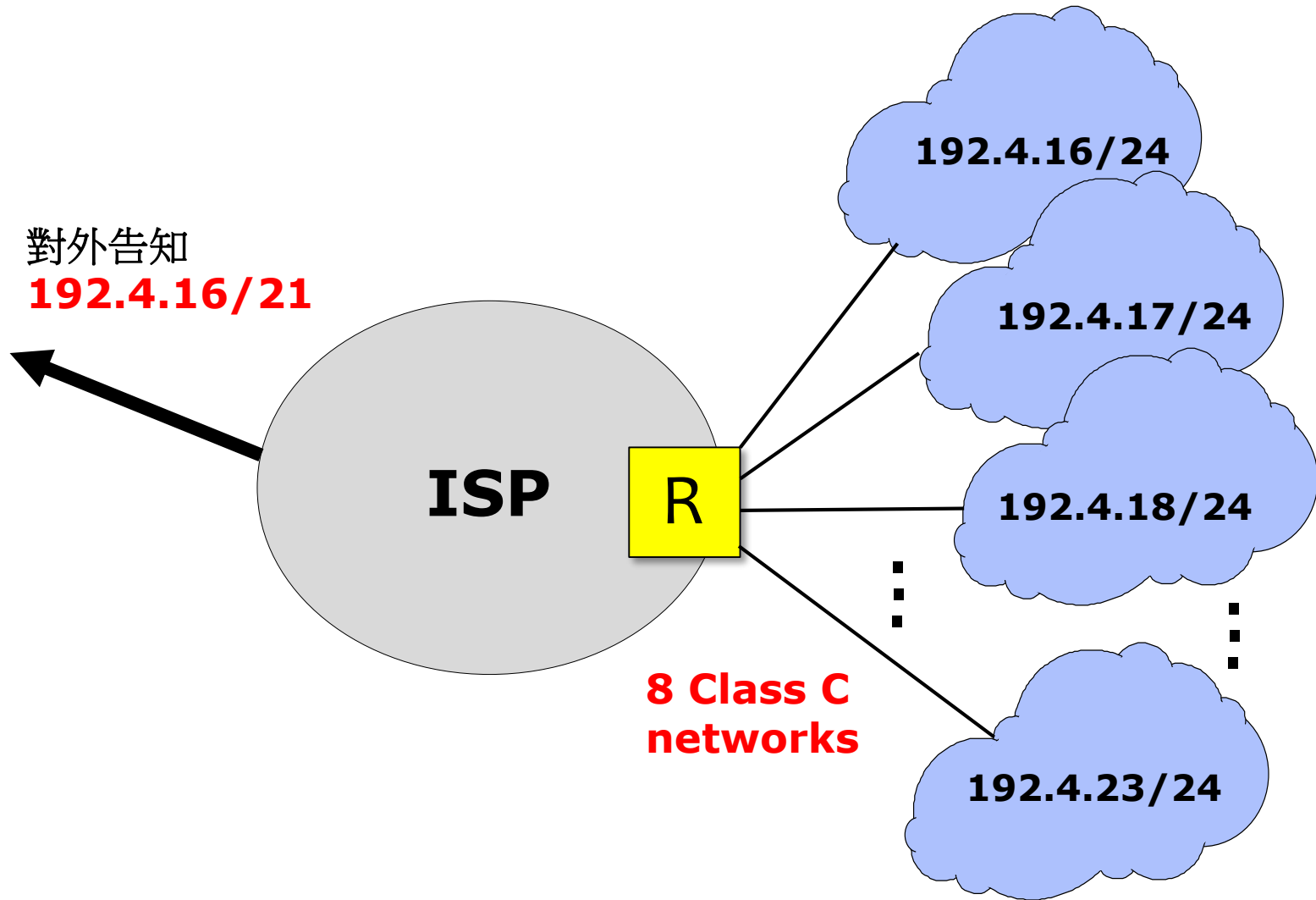
無分級網址

- 習慣上，在前綴後放上“/X”表示這個前綴的長度(位元數)
- 舉例，一個表示從 192.4.16 到 192.4.31 的 20-位元的前綴是 **192.4.16/20**
- 同樣地，想表達一個 class C 的網段 192.4.16，可以寫成 **192.4.16/24**

無分級網址

- 路由協議如何處理無分級的位址？
 - 它必須了解位址中的“網路號碼”可能是任何長度
- 網路號碼使用下列一對數字表示：
<長度, 數值>
- 全部路由器皆必須了解 CIDR
- CIDR 意思是**網段的前綴長度**可以是介於 2到32位元的任意值

無分級網址



經由 **CIDR** 所累積的路由

最長前綴匹配 (Longest prefix matching)

- 前綴在路由表裡也可能發生重疊
 - 有些位址可能匹配到多個前綴
- 舉例，我們可能在一個路由器的路由表 找到以下兩者
 - 171.69 (長度為 16位元的前綴)
 - 171.69.10 (長度為 24位元的前綴)
- 一個目的地是 171.69.10.5 的封包很明顯地與這兩個前綴都匹配
 - 配對規則是“最長前綴優先配對”
 - ▶ 此例中，最長的前綴是 171.69.10
- 一個目的地是 171.69.20.5的封包則會被配對至 171.69 而非 171.69.10

位址解析協議 (ARP)

- 由 IP 位址找出其對應的 MAC 網路位址
 - 目的地主機，下一站的路由器
- 位址解析協議 (ARP, Address Resolution Protocol)
 - IP位址與實體 MAC 位址的對照表
 - 如果IP位址不在對照表中，則廣播請求封包
 - 目的地主機收到請求後回應它的實體 MAC 位址
 - 對照表中的項目具時效性, 逾時未更新會被刪除

ARP 封包格式

0	8	16	31
Hardware type = 1		Protocol type = 0800	
Hlen = 48	Plen = 32	Operation	
SourceHardwareAddr (bytes 0-3)			
SourceHardwareAddr (bytes 4-5)		SourceProtocolAddr (bytes 0-1)	
SourceProtocolAddr (bytes 2-3)		TargetHardwareAddr (bytes 0-1)	
TargetHardwareAddr (bytes 2-5)			
TargetProtocolAddr (bytes 0-3)			

- **HardwareType:** 實體網路的種類 (例如: Ethernet)
- **ProtocolType:** 更上層協議的種類 (例如: IP)
- **HLEN & PLEN:** 實體位址與協議位址的長度
- **Operation:** 請求或回應
- **Source/Target Physical/Protocol addresses** 來源/目的地 實體/協議位址

主機設定

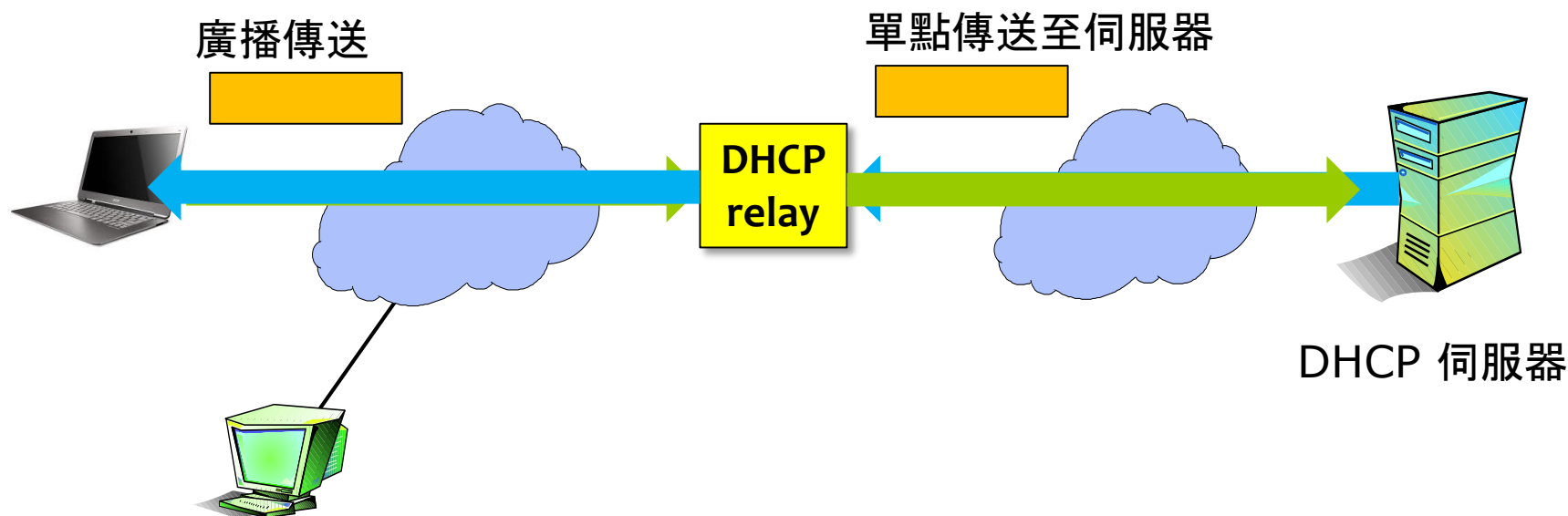
- 乙太網路的位址已被廠家事先設置成唯一性位址
- 互聯網路中的 IP 位址必須具唯一性，同時它必須反應出互連網路的結構
- 大部分作業系統有提供人工設定 IP 資訊的方式給使用者
- 人工設置的缺點
 - ▶ 在一個大型網路環境中，設定所有主機非常耗時
- 需要自動化設置

自動主機設定協議 (DHCP)

- 自動主機設定協議 (Dynamic Host Configuration Protocol)
- DHCP 伺服器負責提供設置資訊給主機
- 在管理網域裡至少有一台**DHCP伺服器**
- DHCP伺服器維護並管理可用的 IP 位址

自動主機設定協議 (DHCP)

- 新啟動或新連接的主機發送 **DHCP DISCOVER** 訊息至一個特殊的 IP 位址 (**255.255.255.255**)
- DHCP 轉送代理 (DHCP relay agent) 單點傳送此訊息至 DHCP 伺服器，並等待回覆



DHCP 範例

The screenshot shows the Windows Network Connections window with the '無線網路連線 狀態' (Wireless Network Connection Status) window open. The '網路連線詳細資料' (Network Connection Details) window is also open, displaying the following information:

內容	值
連線特定 DNS 尾碼	
描述	Intel(R) Centrino(R) Advanced-N 6235
實體位址	C4-85-08-6C-60-A1
DHCP 已啟用	是
IPv4 位址	192.168.0.101
IPv4 子網路遮罩	255.255.255.0
已取得租約	2013年8月20日 下午 06:54:43
租約到期	2013年8月27日 下午 06:54:45
IPv4 預設閘道	192.168.0.1
IPv4 DHCP 伺服器	192.168.0.1
IPv4 DNS 伺服器	192.168.0.1
IPv4 WINS 伺服器	
NetBIOS over Tcpi 已...	是
連結-本機 IPv6 位址	fe80::f9c0:9222:dbcb:4ea8%15
IPv6 預設閘道	
IPv6 DNS 伺服器	

The main window shows the following connection details:

- IPv4 連線能力: 網際網路
- IPv6 連線能力: 無網路存取
- 媒體狀態: 已啟用
- SSID: dlink
- 連線時間: 00:50:50
- 速度: 130.0 Mbps
- 訊號品質:

Activity summary:

- 已傳送: 1,676,517 位元組
- 已收到: 7,024,101 位元組

Buttons: 詳細資料(E)..., 無線內容(W), 內容(P), 停用(D), 診斷(G), 關閉(C)

互聯網控制訊息協議 (ICMP)

- 互聯網控制訊息協議 (Internet Control Message Protocol)
- 定義一組錯誤訊息，當路由器或主機無法成功處理一個 IP 封包時，能將錯誤訊息送回給來源主機
 - 鏈結/節點故障導致無法到達目的地主機
 - 封包重組失敗
 - 封包存活時間 (Time To Live, TTL) 變成 0 (防止封包在網路中永無止盡地繞圈圈)
 - IP 標頭的錯誤檢查碼發現錯誤
- ICMP 重導向 (ICMP-Redirect) 訊息
 - 由路由器傳送給來源主機
 - 告知有一個更好的路由資訊

大綱

- 網路互連簡介
- IP 與路由器
- IP 子網路 (Subnetting)
- 無分級網址 (Classless Addressing)
- 路由協議 (Routing protocols)
- 距離向量協議 (Distance Vector protocol)
- 鏈結狀態協議 (Link State protocol)

路由

■ 轉送與路由

➤ 轉送 (Forwarding):

- 根據目的地位址與路由表選擇一個輸出埠

➤ 路由 (Routing):

- 建立路由表的程序

路由

■ 轉送表 與 路由表

- 轉送表 (Forwarding table)

- 使用於當一個封包正要被轉送
- 轉送表的項目含有網路編號與輸出介面的對應關係以及一些 MAC 資訊，譬如下一站的乙太網路位址

- 路由表 (Routing table)

- 透過路由協議建立
- 基本上包含網路編號與下一站的對應關係

路由

Prefix/Length	Next Hop
140.114/16	171.34.45.12

(a)

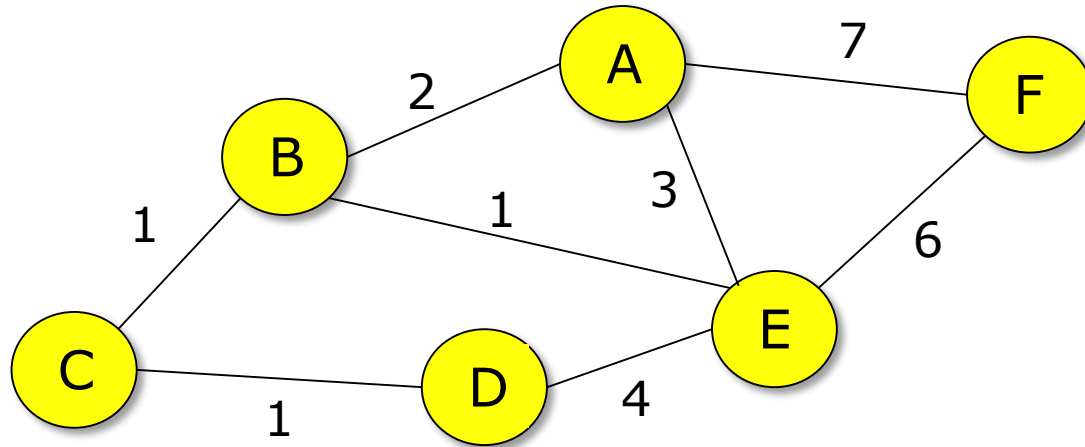
Prefix/Length	Interface	MAC Address
140.114/16	0	8:0:2c:e3:b:20

(b)

範例 (a) 路由表 (b) 轉送表

路由

- 網路可視為一個圖



- 路由的基本問題是找出任意兩節點的 **最小成本路徑 (lowest-cost path)**
 - 一條路徑的成本為路徑上所有邊 (edge) 的成本總和

路由

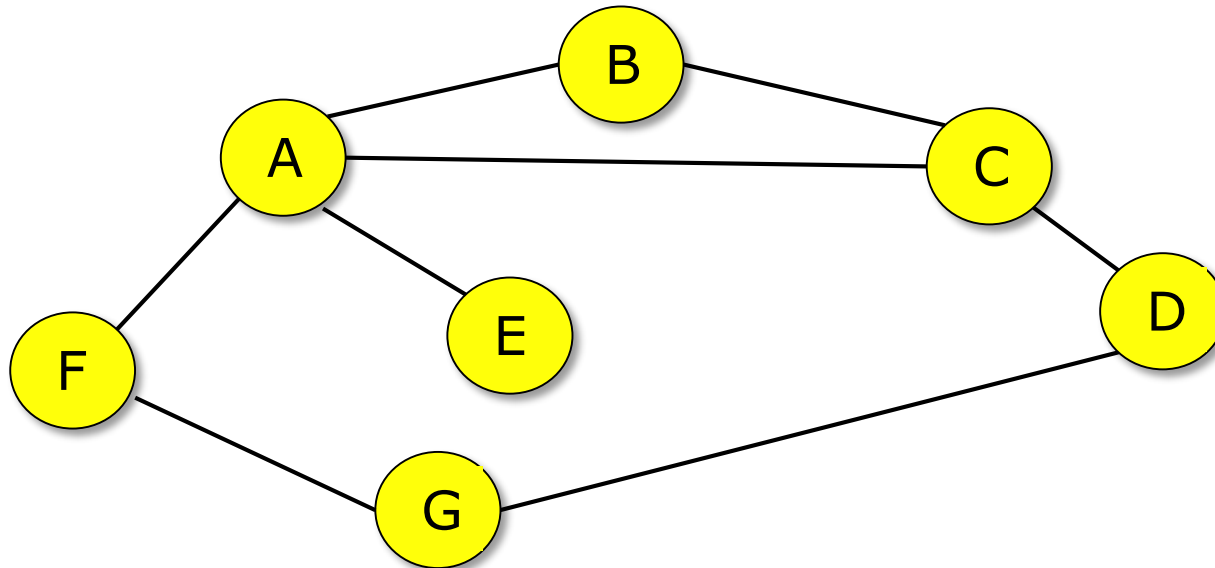
- 對於簡單的網路，我們可以**計算出所有最短路徑**並將之載入至每個節點
- 但這種**靜態方法**有一些缺點
 - 它無法處理節點或鏈結故障的問題
 - 它沒有考慮到新增節點或鏈結的情況
 - 它意味著**邊的成本**不能改變
- **解決方法是什麼？**
 - 需要一個分散式且動態的協議
 - 主要有兩種類別的協議
 - **距離向量 (Distance Vector)**
 - **鏈結狀態 (Link State)**

大綱

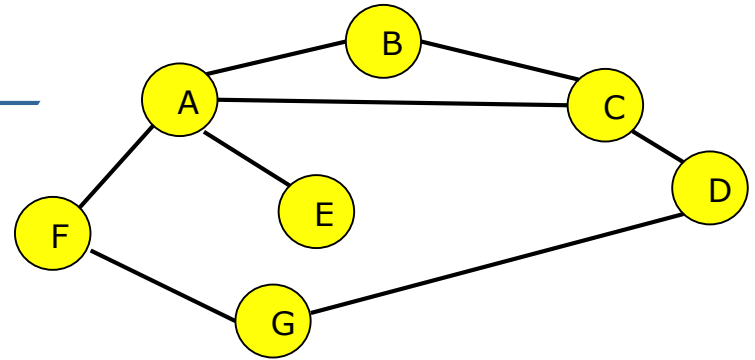
- 網路互連簡介
- IP 與路由器
- IP 子網路 (Subnetting)
- 無分級網址 (Classless Addressing)
- 路由協議 (Routing protocols)
- 距離向量協議 (Distance Vector protocol)
- 鏈結狀態協議 (Link State protocol)

距離向量

- 每個節點建立一個一維陣列(向量)，其包含到所有其他節點”距離”(成本)，並將此距離向量散布給緊鄰的鄰居節點
- 假設每個節點知道它與緊鄰節點之間的成本



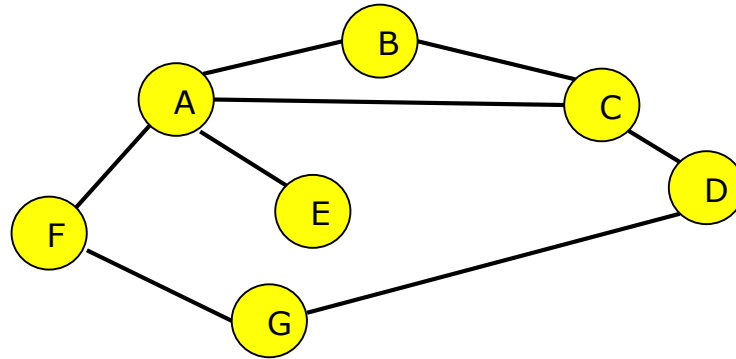
距離向量



每個節點上的距離初始值 (整體彙整)

節點資訊	到其他節點之間的距離						
	A	B	C	D	E	F	G
A	0	1	1	∞	1	1	∞
B	1	0	1	∞	∞	∞	∞
C	1	1	0	∞	∞	∞	1
D	∞	∞	1	0	∞	∞	1
E	1	∞	∞	∞	0	∞	∞
F	1	∞	∞	∞	∞	0	1
G	∞	∞	∞	1	∞	1	0

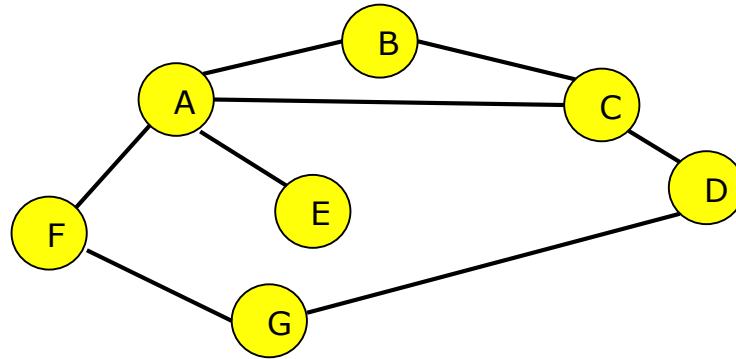
距離向量



Destination	Cost	NextHop
B	1	B
C	1	C
D	∞	--
E	1	E
F	1	F
G	∞	--

節點 A 的初始路由表

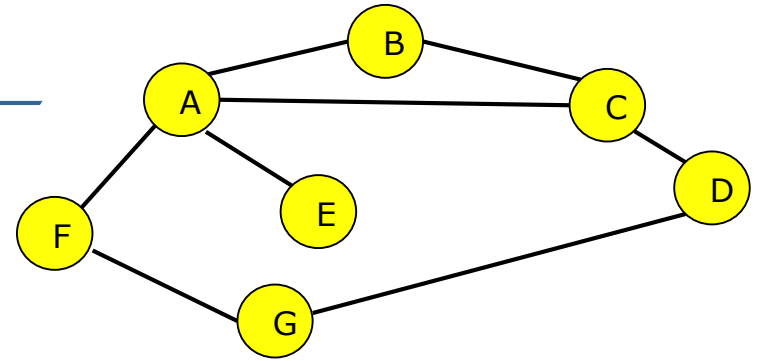
距離向量



Destination	Cost	NextHop
B	1	B
C	1	C
D	2	C
E	1	E
F	1	F
G	2	F

節點 A 的最終路由表

距離向量



每個節點上的最終距離 (整體彙整)

節點資訊	到其他節點之間的距離						
	A	B	C	D	E	F	G
A	0	1	1	2	1	1	2
B	1	0	1	2	2	2	3
C	1	1	0	1	2	2	2
D	2	2	1	0	3	2	1
E	1	2	2	3	0	2	3
F	1	2	2	2	2	0	1
G	2	3	2	1	3	1	0

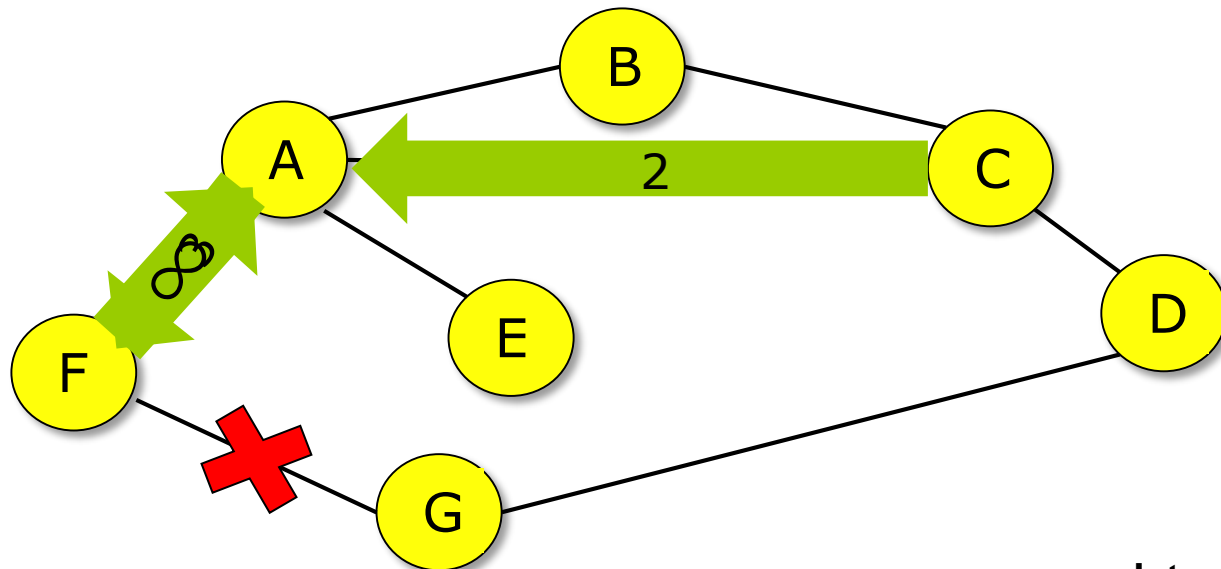
距離向量

- 每經過T秒，每個路由器將它的路由表送給給它的鄰居節點
- 接著，每個路由器根據新的資訊更新它的路由表
- 問題包含
 - 好的消息反應較快
 - 壞的消息反應較慢
 - 更新過程需要過多的訊息

距離向量

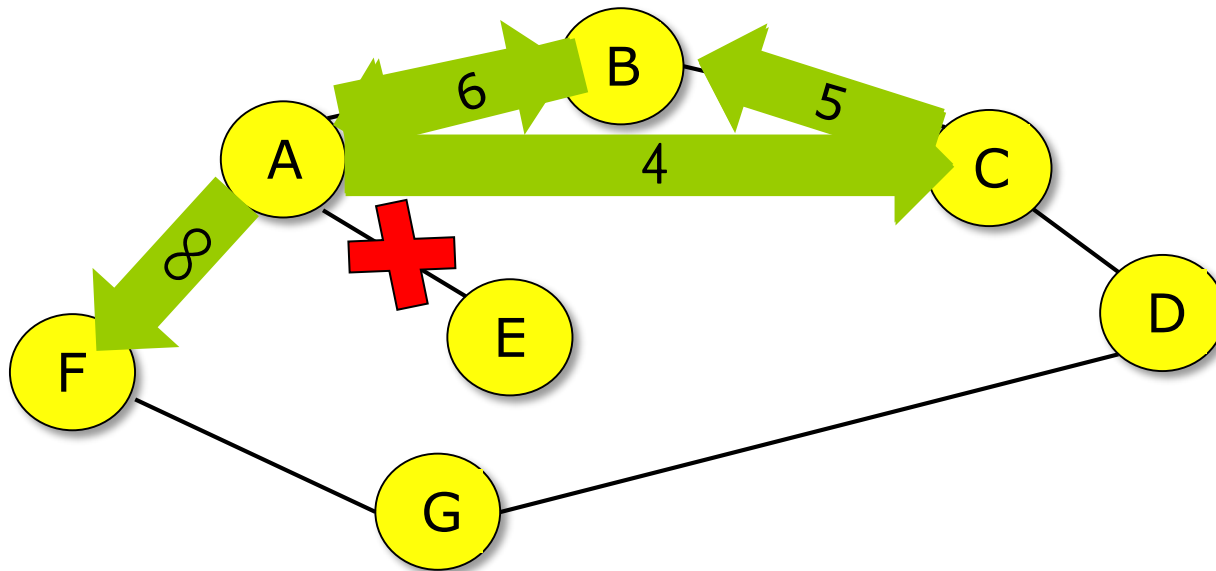
■ 當一個節點偵測出鏈結故障

- ▶ F 偵測出它與 G 之間的鏈結故障
- ▶ F 設定它與 G 的距離無限大，並將此更新告訴 A
- ▶ A 設定它與 G 的距離無限大因為它到 G 的路徑會經過 F
- ▶ A 收到從 C 送來的定期更新說有經過 2 個節點的路徑可到達 G
- ▶ A 設定它與 G 的距離是 3，並將此更新告訴 F
- ▶ F 決定它能藉由 A 經過 可以有 4 個節點的路徑到達 G



距離向量

- 無限計數 (Count-to-infinity) 問題
- 些許不同的情況可能造成網路不穩定
 - 假設 A 與 E 之間的鏈結中斷
 - 在下一次訊息更新中，A 公布它與 E 的距離無限大，但 B 和 C 卻公布它們與 E 的距離是 2



距離向量

- 根據事件發生的確切時間，可能發生底下的事情
 - ▶ B 聽到從 C 來的消息說能經過2個節點到達 E，因此 B 認為能經過3個節點到達 E 並將此通知 A
 - ▶ A 認為它能經過4個節點到達 E，並通知 C
 - ▶ C 認為它能經過5個節點到達 E，並如此反覆下去
 - ▶ 這個循環只會在距離已被設成一個非常大的數字(已可視為無限大)後才停止
 - 無限計數問題

無限計數問題

- 事實上，一個相對小的數字已被來表示趨近無限大
- 例如，通常要經過一個網路，所需跨過的最大節點數量小於 **16 (經過 16 個路由器)**
- **水平分割 (split horizon) 技術**能使路由穩定或是收斂的時間變短
 - 當一個節點發送一個路由更新訊息給它的鄰居，它不將由某個鄰居學到的路徑再回送給該鄰居
 - 舉例來說，如果 B 的路由表裡已有路由 (E, 2, A)，那 B 就知道它一定是向 A 學習來的，所以當 B 要送路由更新給 A 時，該訊息中就不包含路由 (E, 2)

無限計數問題

- 水平分割毒藥反向 (*split horizon with poison reverse*), 一個更強烈版的水平分割
 - B 還是發送該路由更新訊息給 A, 但 B 會在路由內放入負面訊息 (*negative information*) 以保證 A 最終不會使用經由 B 到達 E 的路徑
 - 舉例說明, B 發送路由 (E, ∞) 給 A

大綱

- 網路互連簡介
- IP 與路由器
- IP 子網路 (Subnetting)
- 無分級網址 (Classless Addressing)
- 路由協議 (Routing protocols)
- 距離向量協議 (Distance Vector protocol)
- 鏈結狀態協議 (Link State protocol)

鏈結狀態路由

策略: 將所有直接相連的鏈結資訊 (非整個路由表) 發送給所有節點 (不只有鄰居).

■ 鏈結狀態封包 (LSP), 內容包含

- 產生此 LSP 封包的節點 ID
- 到直接相連鄰居的鏈結成本
- 序號 (SEQNO)
- 封包的 TTL 值

鏈結狀態路由

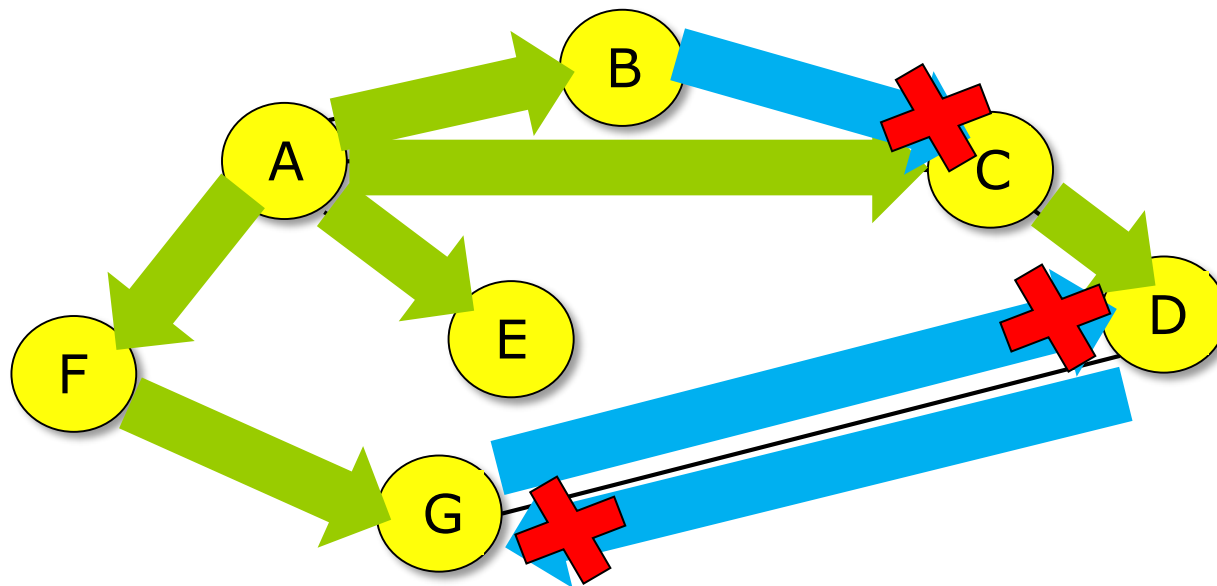
■ 可靠的廣播

- 儲存由各個節點送來的最新 LSP
- 轉送 LSP 給其他所有節點，但不包括發送者
- 定期產生新的 LSP 並增加 SEQNO
- 系統重啟動時，將 SEQNO 歸零
- 將每個儲存 LSP 的 TTL 值減一，當 TTL=0 時則丟棄它

鏈結狀態路由

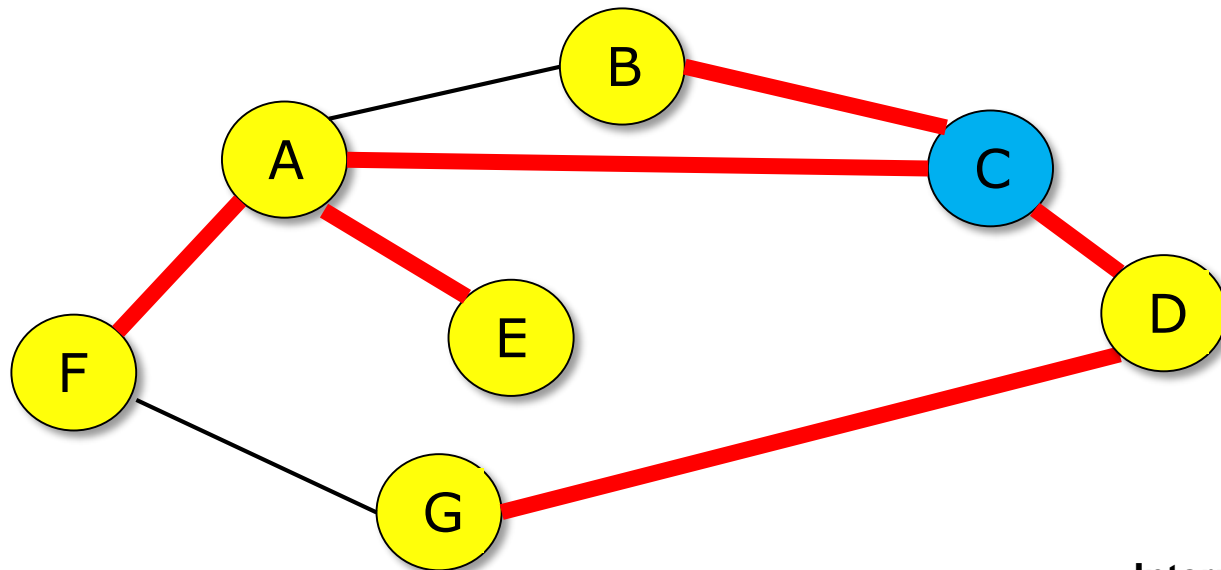
可靠廣播 LSP 封包範例

由節點 A 開始



最短路徑路由

- 開放式最短路徑優先 (Open Shortest Path First, 簡稱 **OSPF**)
- 每個路由器藉由 **Dijkstra 演算法**，根據目前為止收到的 **LSP** 封包去計算它的路由表
- 找出該路由器到所有節點的最短路徑



總結

- 本單元介紹了利用路由器來建構具擴充性且連接異質網路所面臨的問題
- 互聯網協議 (IP)
 - 以無連線模式傳遞資料
 - 盡力式傳送 (不可靠服務)
 - ▶ 封包可能會遺失
 - ▶ 封包傳遞順序可能紊亂
 - ▶ 傳遞重複的封包
 - ▶ 封包在網路中可能被延遲很長一段時間
- 路由器與路由協議
 - 路由表查詢

總結

- IP 子網路
 - 子網路遮罩 → 255.255.255.0
 - 子網路編號
- 無分級網域間路由(CIDR)
 - 累積式路由以減少路由表大小
 - 前綴與前綴長度
 - ▶ 192.4.16/21 表示8個 class C 網段
 - ▶ 192.4.16/22 表示4個 class C 網段
- 動態主機設定協議 (DHCP)
- 互聯網控制訊息協議 (ICMP)

總結

■ 兩種主要類別的協議

● 距離向量

- ▶ 發送整個路由表給直接相連的鄰居
- ▶ 好的消息反應較快
- ▶ 壞的消息反應較慢
- ▶ 無限計數問題
- ▶ 水平分割解決方法
- ▶ 水平分割毒藥反向方法
- ▶ 路由訊息協議 (RIP, Routing Information Protocol)

總結

- 鏈結狀態

- ▶ 將直接相連的鏈結資訊發送給所有節點
- ▶ 可靠且廣播方式傳送 LSP
- ▶ 每個路由器有整個自治系統的網路拓撲
- ▶ 計算路由器到其他所有節點之間的最短路徑
- ▶ 開放式最短路徑優線 (OSPF, Open Shortest Path First)