

# 第五章:連結層和區域網路

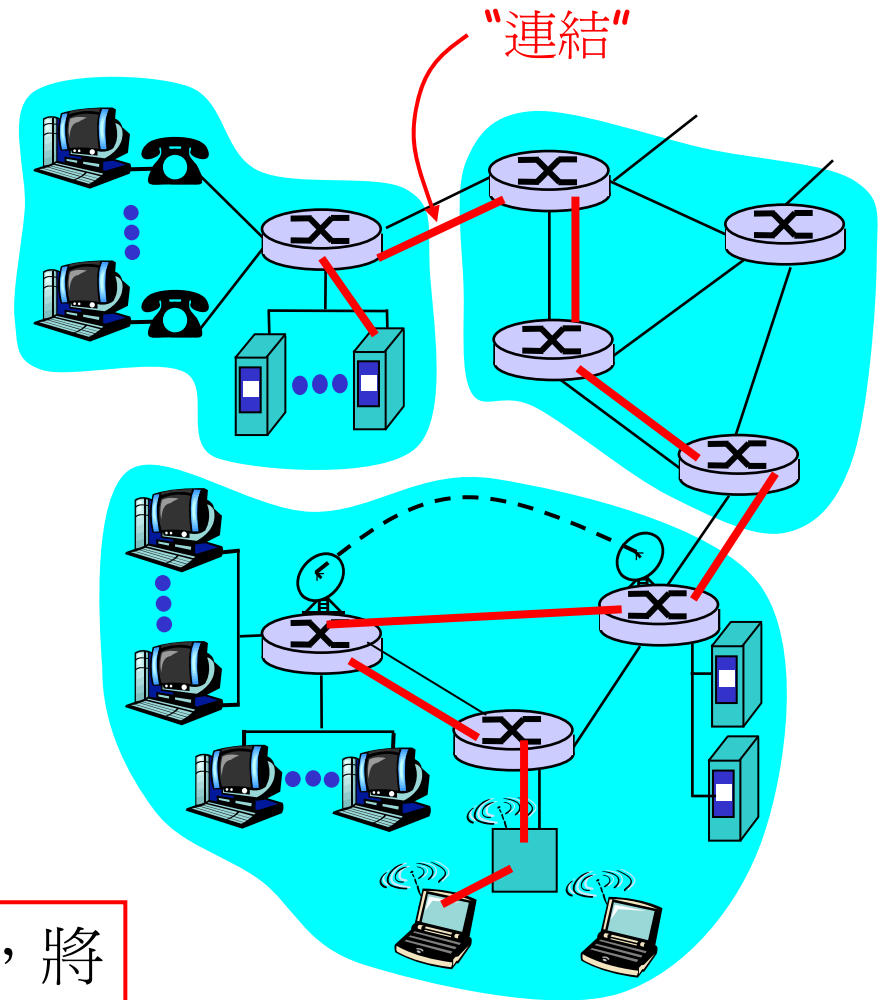
- 5.1 簡介與服務
- 5.2 錯誤偵測和更正技術
- 5.3 多重存取協定
- 5.4 連結層定址
- 5.5 乙太網路
- 5.6 集線器和交換器
- 5.7 PPP
- 5.8 虛擬連結: ATM 與 MPLS

# 連結層：簡介

## 一些術語：

- ❑ 主機和路由器皆稱為**節點**
- ❑ 沿著通訊路徑，連結相鄰節點的通訊通道，稱為**連結**
  - 有線連結
  - 無線連結
  - 區域網路
- ❑ 第二層的封包稱為**訊框**，將資料段封裝

**資料連結層**的責任為，經由連結，將資料段從一個節點傳輸到相鄰的節點



# 連結層: 內容

- ❑ 資料段在不同的連結上經由不同的連結協定傳輸:
  - 例如, 第一個連結為乙太網路, 中間連結為訊框傳送, 最後一個連結為 **802.11**
- ❑ 每一種連結協定提供不同的服務
  - 例如, 在連結上可能提供或不提供 **rdt**

## 運輸的比方

- ❑ 從普林斯頓到洛桑市的行程
  - 小型巴士: **Princeton** 到 **JFK**
  - 飛機: **JFK** 到日內瓦
  - 火車: 日內瓦到洛桑
- ❑ 旅客 = **資料段**
- ❑ 每段旅程 = **通訊連結**
- ❑ 運輸模式 = **連結層協定**
- ❑ 旅行社 = **路由演算法**

# 連結層服務

## ❑ 訊框化, 連結存取:

- 將資料段封裝成訊框，加入標頭及標尾
- 通道存取假如媒介共享
- “**MAC**” 位址在訊框標頭中，用來識別來源端、目的端
  - 與 **IP** 位址不同!

## ❑ 相鄰節點間的可靠傳輸

- 我們已經學習過了 (第三章)!
- 在位元錯誤率低的連結上很少使用 (光纖，某些雙絞線)
- 無線連結: 錯誤率高
  - 問: 為什麼同時需要連結層和端點對端點的可靠性?

# 連結層服務 (更多)

## □ 流量控制:

- 調整相鄰傳送端和接收端節點之間的步調

## □ 錯誤偵測:

- 因訊號衰減或雜訊所產生的錯誤
- 接收端偵測到錯誤的存在:
  - 通知傳送端重新傳送訊息或是將訊框丟掉

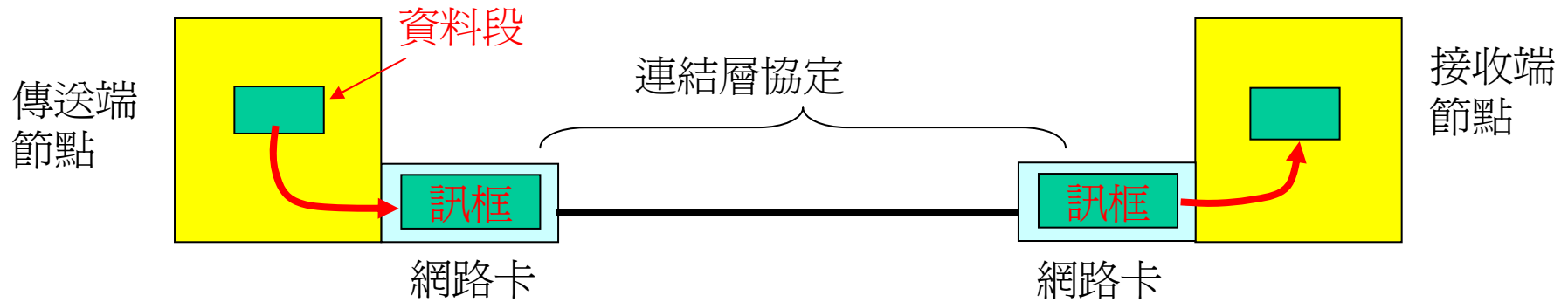
## □ 錯誤更正:

- 不憑藉重新傳送，接收端辨識並更正位元錯誤

## □ 半雙工和全雙工

- 在半雙工中，連結兩端的節點可以傳送，但無法同時

# 網路卡通訊



## ❑ 連結層協定以「網路卡」實作 (也稱為 NIC)

- 乙太網路卡, PCMCIA 卡, 802.11 卡

## ❑ 傳送端:

- 將資料段封裝在訊框中
- 加入錯誤確認位元, rdt, 流量控制等等

## ❑ 接收端

- 尋找錯誤, rdt, 流量控制等等
- 取出資料段, 將它傳送到接收端節點

## ❑ 網路卡是半自主裝置

## ❑ 連結和實體層

# 第五章:連結層和區域網路

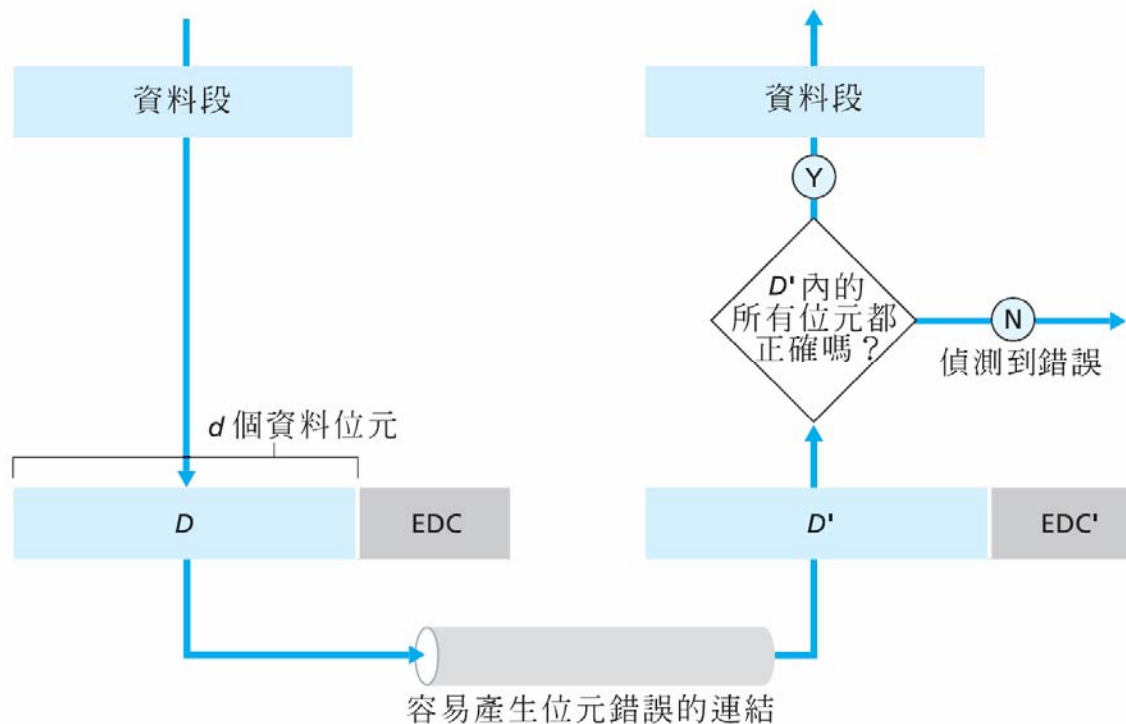
- 5.1 簡介與服務
- 5.2 錯誤偵測和更正技術
- 5.3 多重存取協定
- 5.4 連結層定址
- 5.5 乙太網路
- 5.6 集線器和交換器
- 5.7 PPP
- 5.8 虛擬連結: ATM 與 MPLS

# 錯誤偵測

EDC= 錯誤偵測及更正位元 (冗餘)

D = 被錯誤檢查所保護的資料，可能包含標頭欄位

- 錯誤偵測並不是 100% 可靠的!
  - 協定可能會遺漏某些錯誤，不過這很少發生
  - 更大的 **EDC** 欄位能夠得到最佳的偵測與更正

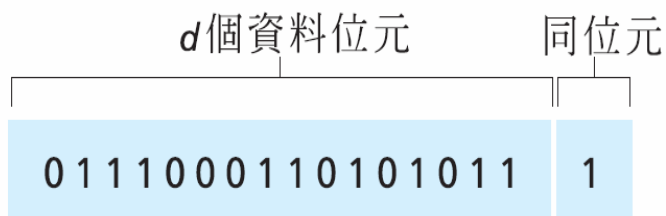




# 同位檢查

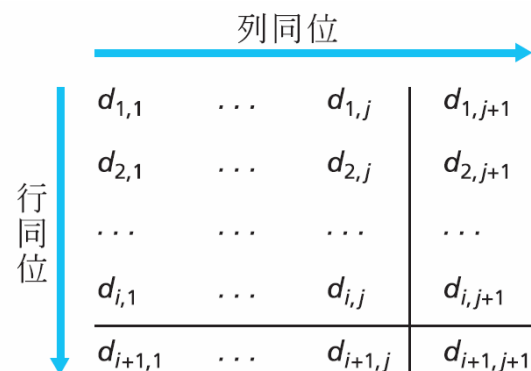
## 單一的同位元:

偵測單一位元錯誤

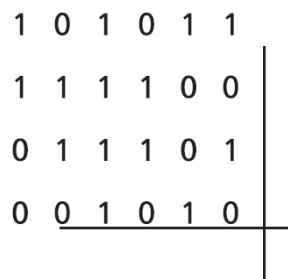


## 二維同位元:

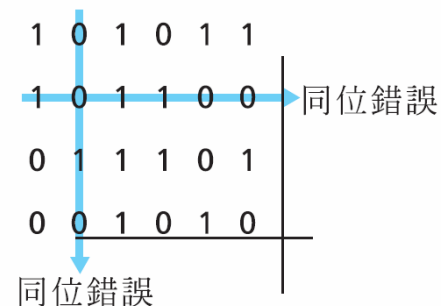
偵測並更正單一位元錯誤



沒有錯誤



可更正的單一位元錯誤



# 網際網路檢查和

目標: 偵測在傳送的資料分段中的“錯誤”(例如，翻轉的位元)(注意: 只用在傳輸層)

## 傳送端:

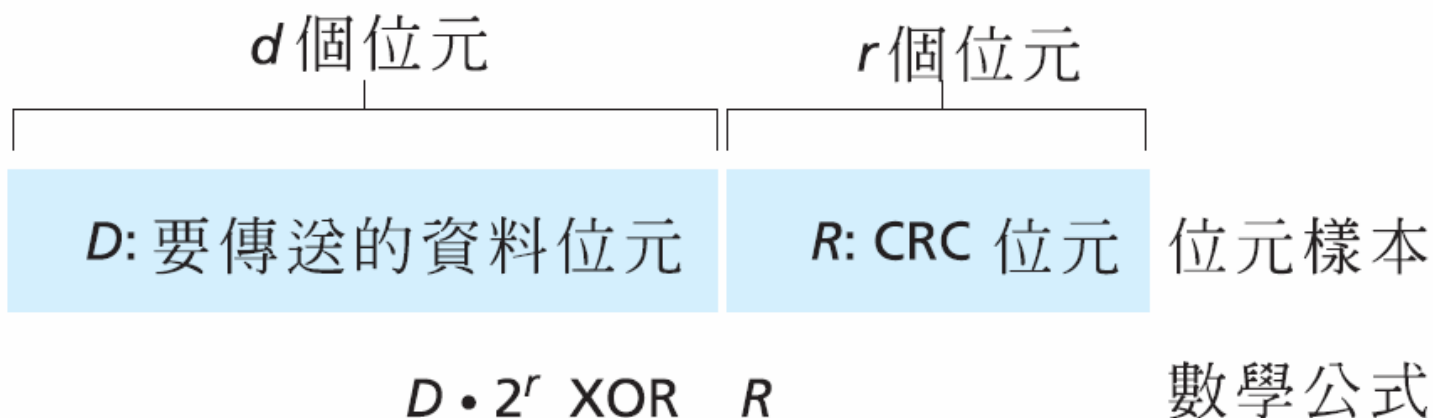
- 將資料分段的內容視為一連串**16**位元的整數
- 檢查和: 資料分段內容相加 (**1**的補數和)
- 傳送端會將檢查和的值放入**UDP**檢查和欄位中

## 接收端:

- 計算收到的資料分段的檢查和
- 確認計算出來的檢查和是否和檢查和欄位中的相等:
  - **NO** - 偵測到錯誤
  - **YES** - 沒有偵測到錯誤。但是仍然可能有錯誤? 後面有更多介紹....

# 檢查和: 循環冗餘檢查

- ❑ 將資料位元 **D** 視為二進位數字
- ❑ 選擇 **r+1** 個位元的樣本 (產生器), **G**
- ❑ 目標: 選擇 **r** 個 **CRC** 位元, **R**, 使得
  - $\langle D, R \rangle$  正好能被 **G** 整除 (模數 2)
  - 接收端知道 **G**, 將  $\langle D, R \rangle$  除以 **G**, 假如餘數不為 0: 偵測到錯誤!
  - 可以偵測出任何少於 **r+1** 個位元的叢發錯誤
- ❑ 實務上經常使用 (ATM, HDCL)



# CRC 範例

想要:

$$D \cdot 2^r \text{ XOR } R = nG$$

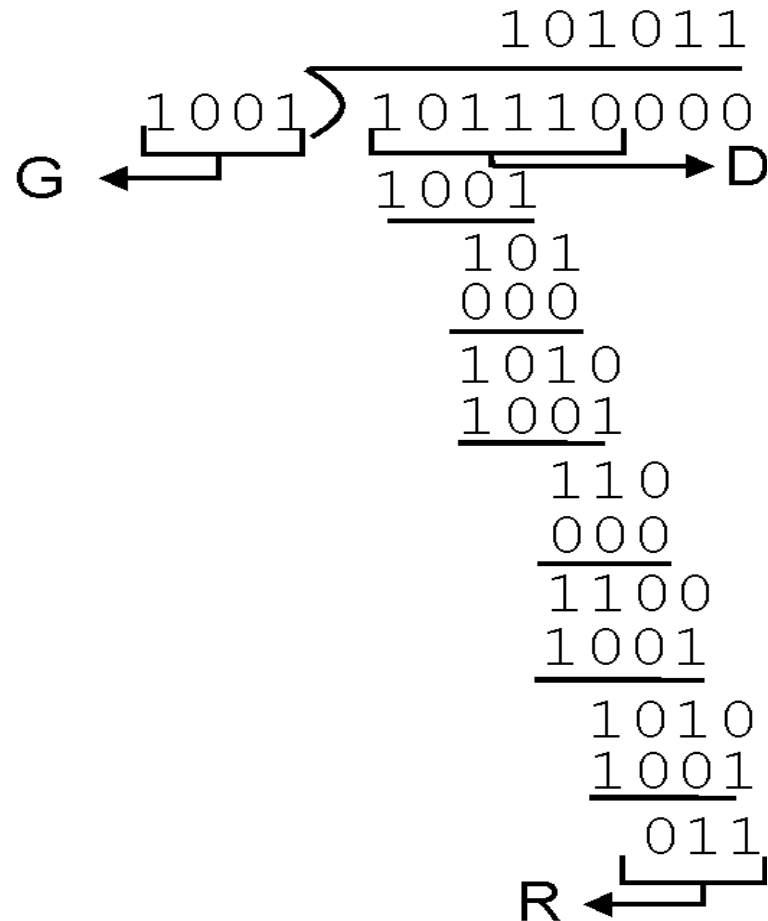
同樣地:

$$D \cdot 2^r = nG \text{ XOR } R$$

同樣地:

假如我們將  $D \cdot 2^r$  除以  $G$ ，應得到餘數  $R$

$$R = \text{餘數} \left[ \frac{D \cdot 2^r}{G} \right]$$



# 第五章:連結層和區域網路

- 5.1 簡介與服務
- 5.2 錯誤偵測和更正技術
- 5.3 多重存取協定
- 5.4 連結層定址
- 5.5 乙太網路
- 5.6 集線器和交換器
- 5.7 PPP
- 5.8 虛擬連結: ATM 與 MPLS

# 多重存取連結與協定

## 兩種型態的“連結”：

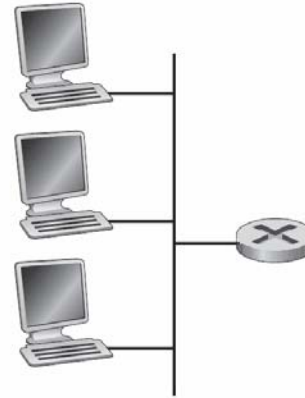
### □ 點對點

- 撥號存取的PPP
- 乙太網路交換器和集線器之間的點對點連結

### □ 廣播 (分享的線路或媒介)

- 傳統的乙太網路
- 上傳 HFC
- 802.11 無線區網

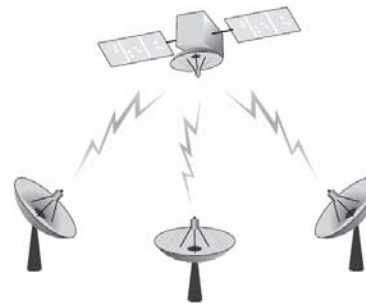
共享網路  
(例如，乙太網路)



共享無線  
(例如，Wifi)



衛星網路



雞尾酒宴會



# 多重存取協定

- ❑ 單一的分享廣播通道
- ❑ 同時兩個或更多節點的傳送: 干擾
  - 碰撞 假如節點在同一個時間收到兩個或更多個信號

## 多重存取協定

- ❑ 決定節點如何分享通道的分散式演算法。例如，決定節點何時能夠傳送
- ❑ 必須使用通道本身來傳送有關通道分享的通訊!
  - 沒有不同頻帶的通道做為協調的功能

# 理想的多重存取協定

## 速率為 $R$ bps 的廣播通道

1. 當一個節點要傳送資料時，它可以擁有  $R$  bps 流通率
2. 當  $M$  個節點要傳送資料時，每一個節點以平均速率  $R/M$  來傳送資料
3. 完全的非集中式：
  - 沒有用來協調傳輸的特殊節點
  - 沒有時脈和時槽的同步
4. 簡單



# MAC 協定: 分類

三種常用的類別:

## □ 通道分割

- 將通道分割成「小塊」(時槽、頻率、碼)
- 將這些「小塊」分配為每個節點專用的

## □ 隨機存取

- 不分割通道，允許碰撞
- 從碰撞「復原」

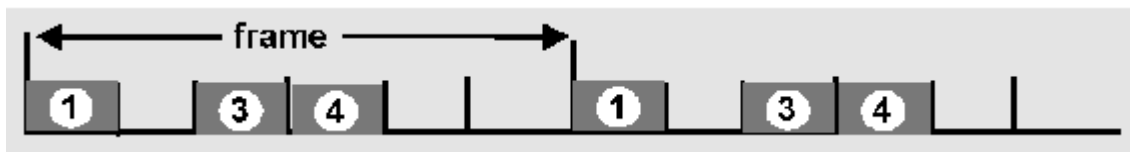
## □ “輪流存取”

- 節點可以輪流，但是要傳送較多資料的節點可以傳送較長的時間

# 通道分割 MAC 協定: TDMA

## TDMA: 時間分割多重存取

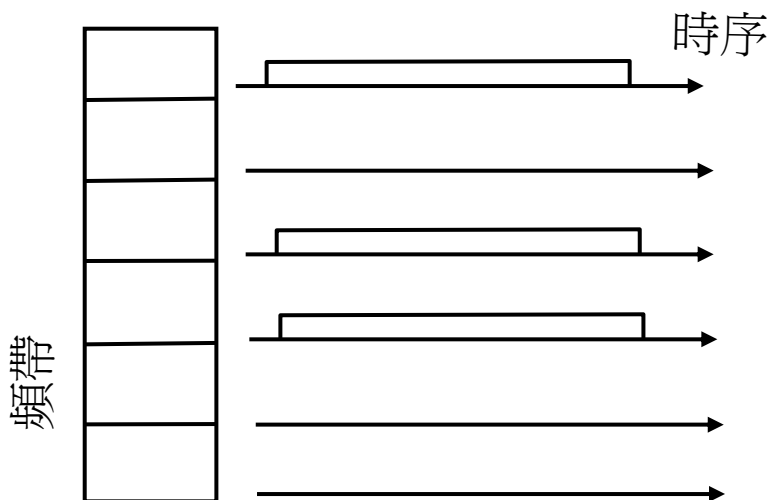
- ❑ 每一個「回合」中，存取通道
- ❑ 每一站在每一回合中，得到固定長度的時槽 (長度 = 封包傳送時間)
- ❑ 沒有使用的時槽會變成閒置的
- ❑ 範例: 6個站的 LAN, 1,3,4 有封包, 時槽 2,5,6 為閒置的



# 通道分割 MAC 協定: FDMA

## FDMA: 頻率分割多重存取

- 通道頻譜被分割成頻帶
- 每一站分配到固定的頻帶
- 沒有用到的傳輸頻帶會閒置
- 範例: 6個站的 LAN, 1,3,4 有封包, 頻帶 2,5,6 為閒置的



# 隨機存取協定

- 當節點有封包要傳送時
  - 以全部的通道資料速率  $R$  傳送
  - 節點間沒有做前置的協調
- 兩個以上的節點傳送資料 → “碰撞”
- 隨機存取 **MAC** 協定說明:
  - 要如何偵測碰撞
  - 要如何從碰撞中復原 (例如，經由延遲重傳)
- 隨機存取 **MAC** 協定的範例:
  - 時槽式 ALOHA
  - ALOHA
  - CSMA, CSMA/CD, CSMA/CA

# 時槽式 ALOHA

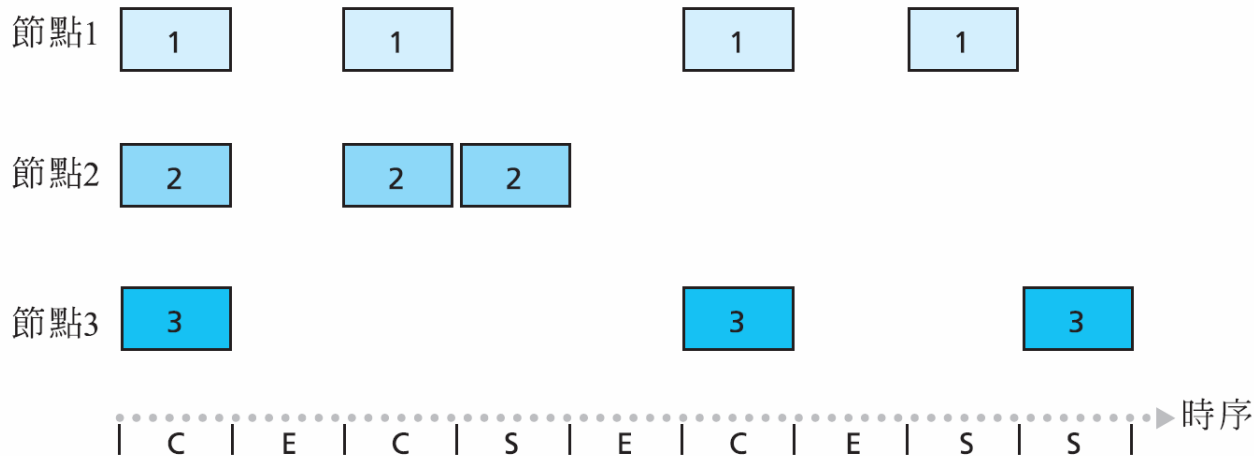
## 假設

- 所有的訊框都是同樣大小
- 時間被切分為同樣大小的時槽，長度可傳送1個訊框
- 節點只有在時槽開始時，才會傳送訊框
- 節點為同步的
- 如果兩個以上的節點在時槽內傳送，所有的節點都會偵測到碰撞

## 運作

- 當節點要傳送新的訊框時，它會在下一個時槽傳送
- 如果沒有碰撞，節點會在下一個時槽傳送新的訊框
- 假如產生碰撞，節點會依照機率  $p$  在接下來的每個時槽中重新傳送訊框，直到成功為止

# 時槽式 ALOHA



## 優點

- ❑ 單一的活動節點能以通道的全部速率連續傳送
- ❑ 高度的非集中式: 只有節點中的時槽需要同步
- ❑ 簡單

## 缺點

- ❑ 碰撞, 浪費時槽
- ❑ 閒置的時槽
- ❑ 節點需要在比傳輸封包少的時間內偵測碰撞
- ❑ 時脈同步

# 時槽式 Aloha 的效率

**效率** 為當有大量節點，每個節點有大量的訊框要傳送時，長時間下成功時槽所佔的比例

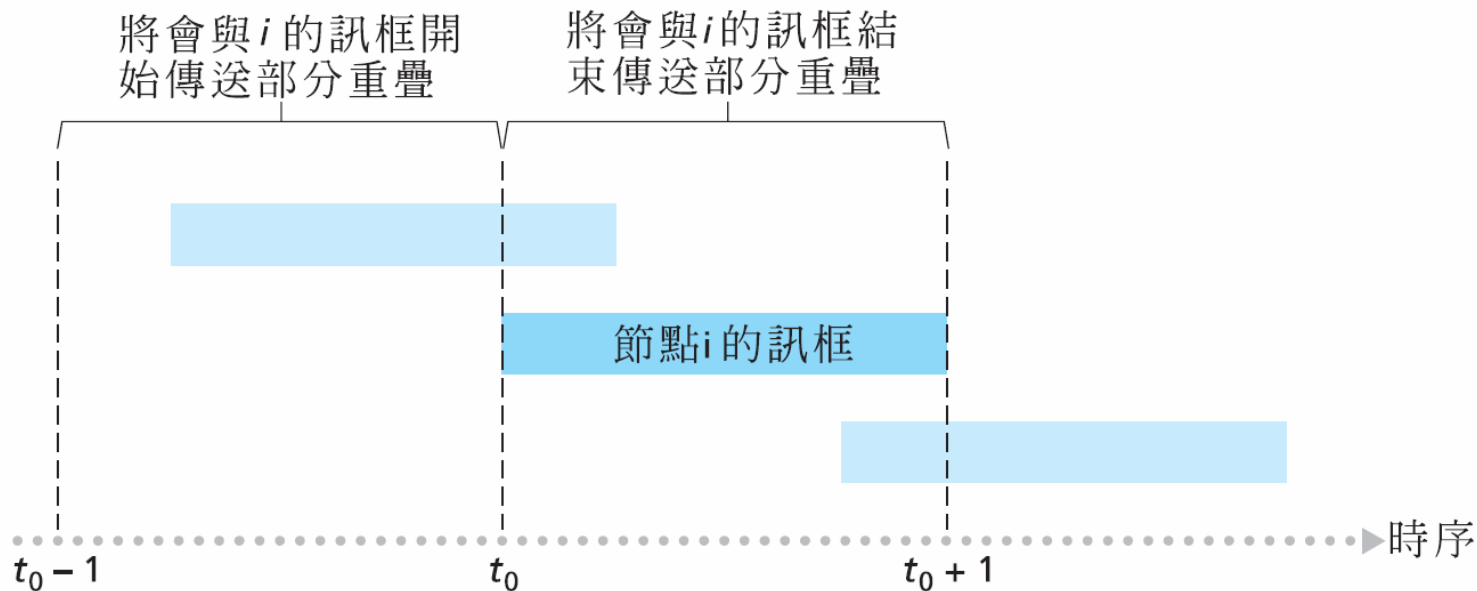
- 假設有  $N$  個節點有很多訊框要傳送，每個時槽傳送的機率為  $p$
- 節點 1 在一個時槽中成功的機率為  $= p(1-p)^{N-1}$
- 任意節點傳送成功的機率為  $= Np(1-p)^{N-1}$

- 為了得到  $N$  個節點的最大效率，我們必須找出  $p^*$  使得  $Np(1-p)^{N-1}$  具有最大值
- 當有許多節點時， $N$  趨近於無限大，我們得到  $Np^*(1-p^*)^{N-1}$  的極限值， $1/e = .37$

**最佳狀況：**通道成功用來傳輸的時間只有**37%**!

# 純粹 (非時槽式的) ALOHA

- ❑ 非時槽式的 **Aloha**: 較簡單, 非同步的
- ❑ 當訊框一抵達時
  - 會立即傳送
- ❑ 會增加碰撞的機率:
  - 在  $t_0$  傳送的訊框會與在  $[t_0-1, t_0+1]$  傳送的其他訊框碰撞





# 純粹 Aloha 的效率

$P(\text{節點成功傳送的機率}) = P(\text{節點傳送}) \cdot$

$P(\text{沒有其他節點在 } [p_o-1, p_o] \text{ 傳送}) \cdot$

$$\begin{aligned} &P(\text{沒有其他節點在 } [p_o-1, p_o] \text{ 傳送}) \\ &= p \cdot (1-p)^{N-1} \cdot (1-p)^{N-1} \\ &= p \cdot (1-p)^{2(N-1)} \end{aligned}$$

... 選擇最佳的  $p$  並讓  $n \rightarrow$  無限大 ...

更糟！

$$= 1/(2e) = .18$$

# CSMA (載波感測多重存取)

CSMA: 傳送之前先聆聽:

假如通道感測到閒置: 傳送整個訊框

□ 假如通道感測到忙碌，延後傳送

□ 人類的比方: 不要打斷別人的談話!

# CSMA 碰撞

碰撞還是會發生:

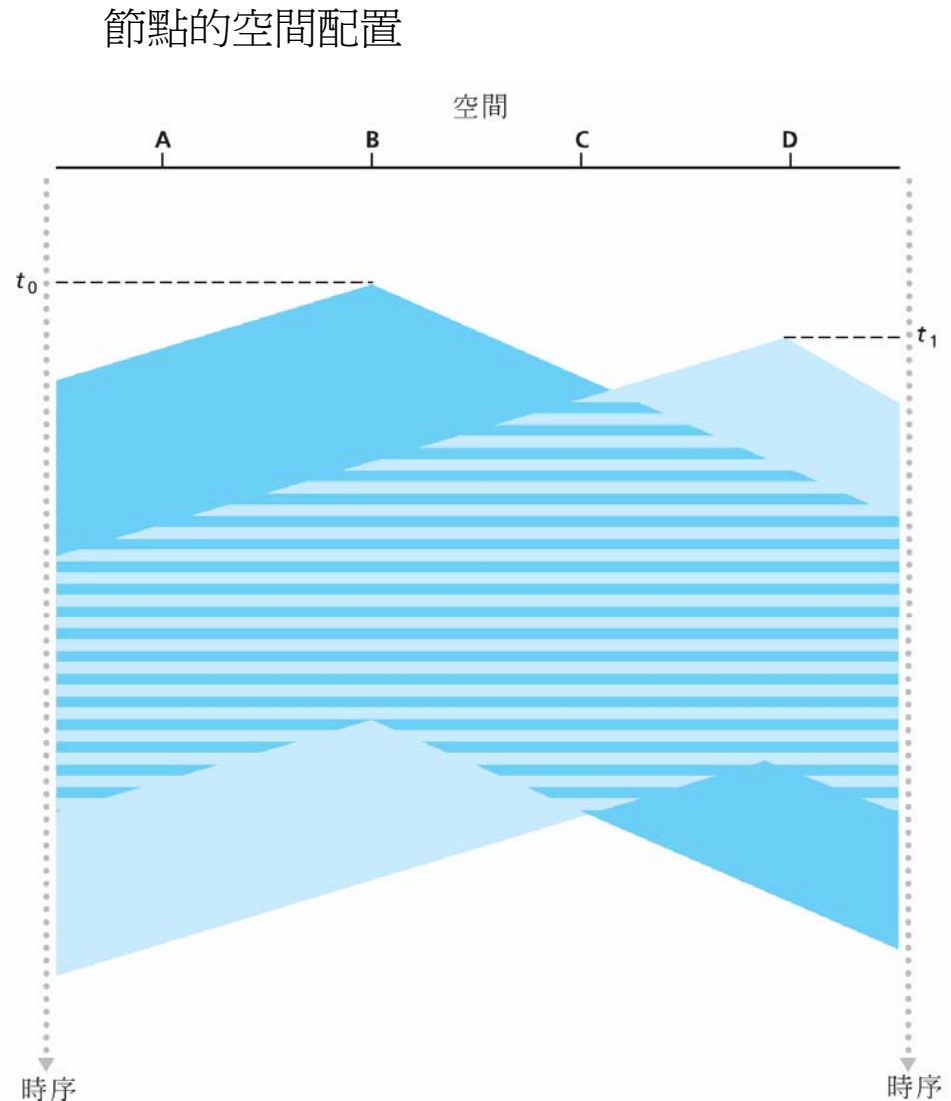
傳遞延遲使得兩個節點可能無法聽到其他人的傳送

碰撞:

傳送整個封包  
時間的浪費

注意:

距離和傳遞延遲決定了碰撞的機率



# CSMA/CD (碰撞偵測)

**CSMA/CD**:載波感測(carrier sensing), 如CSMA一般會延遲傳送

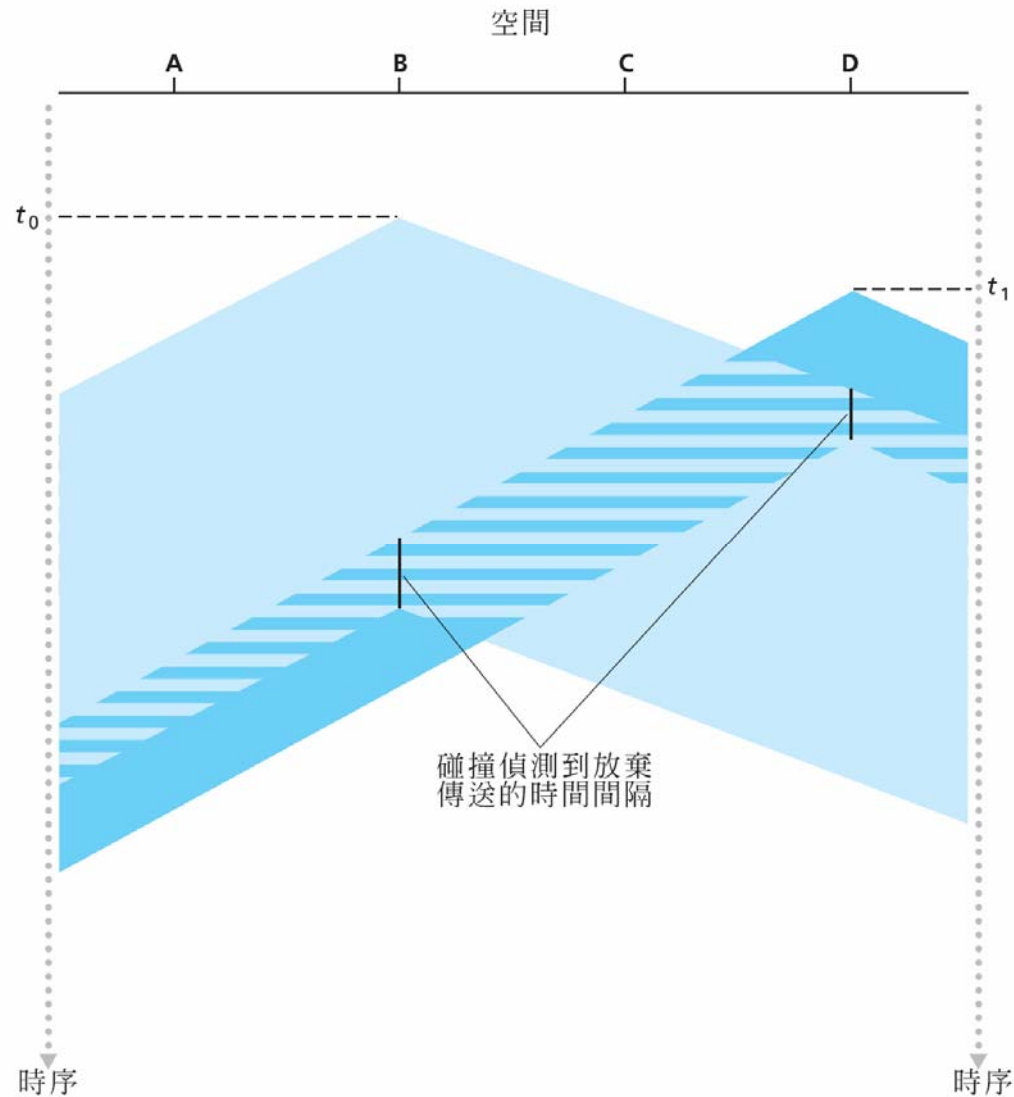
- 取消碰撞的傳送，減少通道的浪費

## □ 碰撞偵測:

- 在區域網路中是簡單的: 測量訊號強度，比較傳送和接收訊號
- 在無線區域網路中是困難的: 當傳送時，接收端關閉

## □ 人類的比方: 有禮貌的交談者

# CSMA/CD 碰撞偵測



# “輪流” MAC 協定

## 通道分割 MAC 協定:

- 在高承載狀況下，有效且公平地分享通道
- 在低承載狀況下，效率差: 通道存取的延遲，在只有一個活動節點的狀況下，也只能分配到 $1/N$ 的頻寬！

## 隨機存取 MAC 協定

- 在低承載的狀況下有效率: 單一節點可以完全利用通道
- 高承載: 碰撞的負擔

## “輪流” 協定

尋找兩邊的優點！

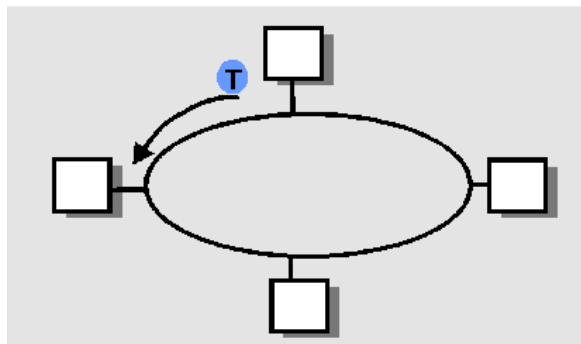
# “輪流” MAC 協定

## 輪詢(Polling):

- ❑ 主節點輪流邀請從屬節點傳送資料
- ❑ 問題點:
  - 輪詢的額外負擔
  - 延遲
  - 單點故障 (主節點)

## 記號傳遞(Token passing):

- ❑ 控制記號依序從一個節點傳到下一個節點
- ❑ 記號訊息
- ❑ 問題點:
  - 記號的額外負擔
  - 延遲
  - 單點故障 (記號)



# MAC 協定總結

## □ 使用分享媒體做些什麼？

- 通道分割, 以時間, 頻率或碼
  - 時間分割, 頻率分割
- 存機切分 (動態的),
  - **ALOHA, S-ALOHA, CSMA, CSMA/CD**
  - 載波感測: 在某些技術中是簡單的 (有線), 其他的有困難 (無線)
  - **CSMA/CD** 使用在乙太網路
  - **CSMA/CA** 使用在 802.11
- 輪流
  - 從中心點輪詢, 傳遞記號



# 區域網路的技術

到目前介紹的連結層：

- 服務, 錯誤偵測/更正, 多重存取

接下來：區域網路的技術

- 定址
- 乙太網路
- 集線器, 交換器
- PPP

# 第五章:連結層和區域網路

- 5.1 簡介與服務
- 5.2 錯誤偵測和更正技術
- 5.3 多重存取協定
- 5.4 連結層定址
- 5.5 乙太網路
- 5.6 集線器和交換器
- 5.7 PPP
- 5.8 虛擬連結: ATM 與 MPLS

# MAC 位址與 ARP

## □ 32位元的 IP 位址:

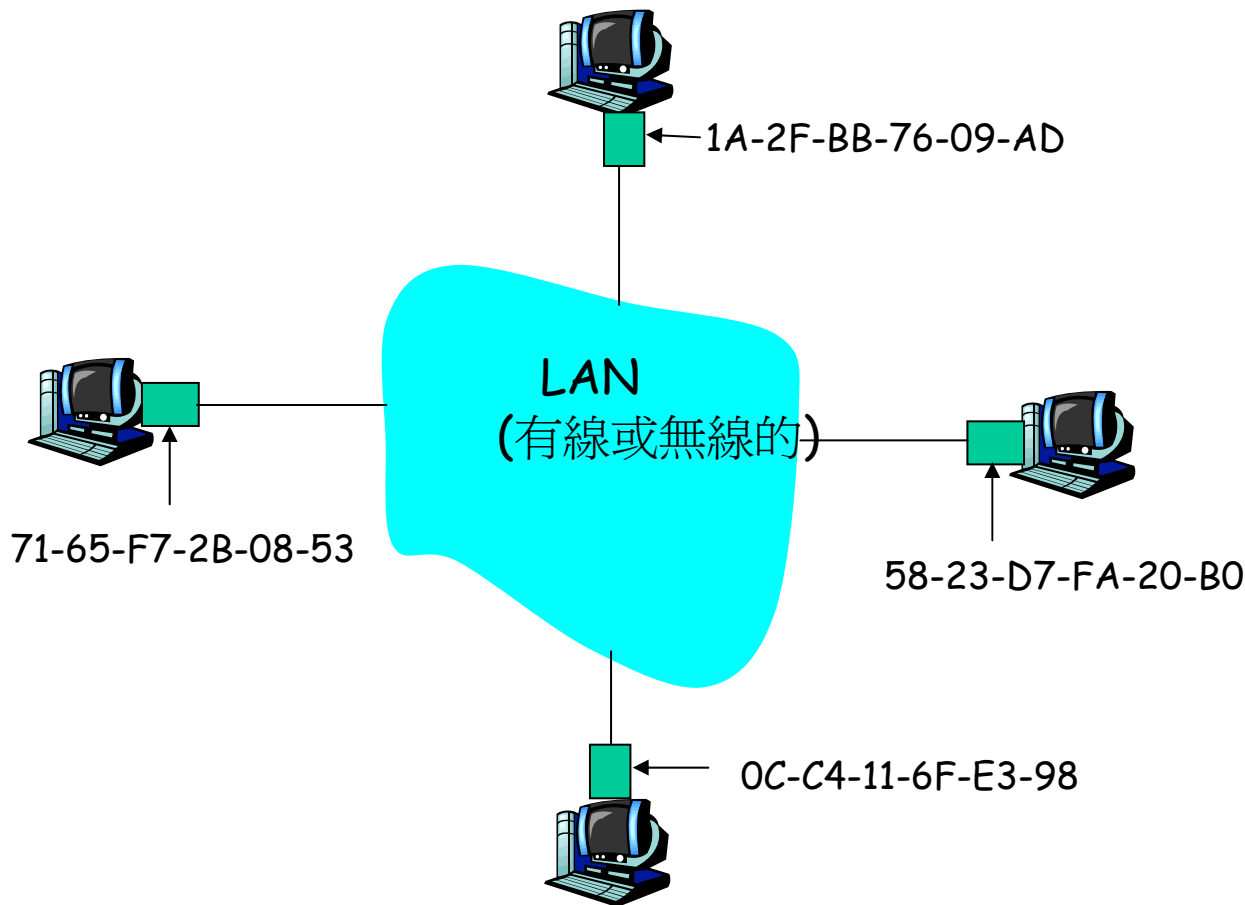
- 網路層位址
- 用來接收資料段，到目的IP子網路

## □ MAC (或是區域網路、實體、乙太網路) 位址:

- 用來接收訊框，從一個介面到另一個實體連結介面 (同一個網路)
- 48 位元 MAC 位址 (大多數的 LAN)，燒在網路卡 ROM中

# LAN 位址與 ARP

每一張網路卡都具有唯一的LAN位址



廣播位址 =  
FF-FF-FF-FF-FF-FF

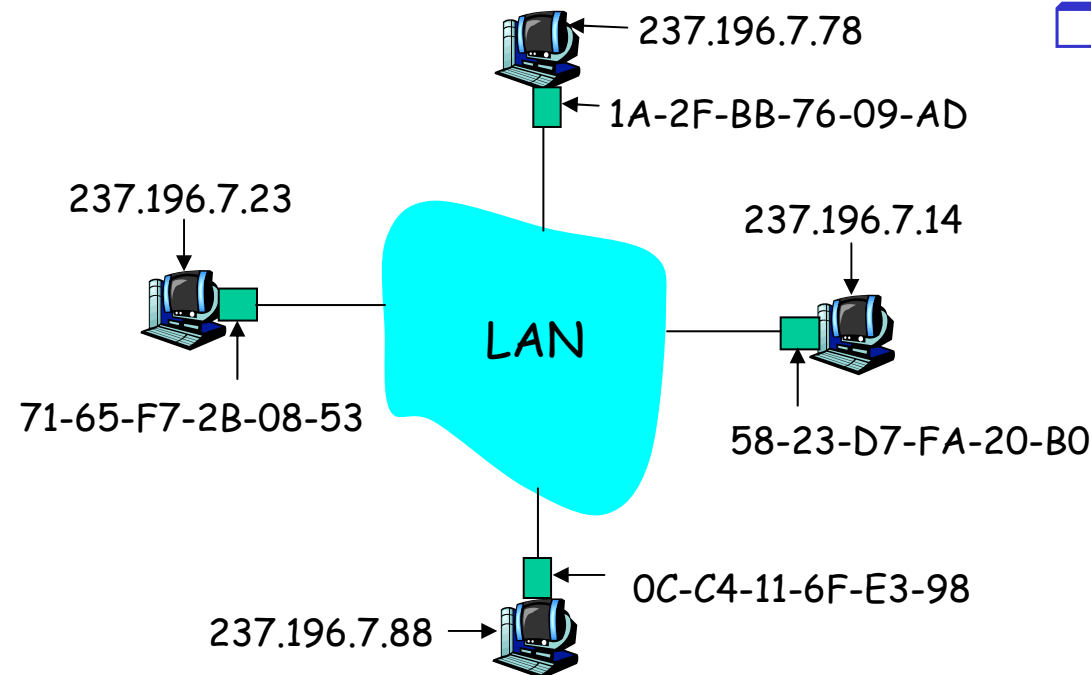
■ = 網路卡

# LAN 位址 (更多)

- **MAC**位址的分配由**IEEE**管理
- 製造商購買部分的**MAC**位址(保證唯一性)
- 比方:
  - (a) **MAC** 位址: 如同社會安全號碼
  - (b) **IP** 位址: 如同郵寄地址
- **MAC**位址是平面的→ 可攜性
  - 可以將區域網路卡從一個**LAN**帶到另一個**LAN**
- **IP** 階層是位址 不具可攜性
  - 與節點連接到得**IP**子網路相關

# ARP: 位址解析協定

問題: 當我知道 **B** 的 **IP** 位址時, 要怎麼決定 **B** 的 **MAC** 位址?



□ 區域網路上的每一個 IP 節點 (主機, 路由器) 都有 **ARP** 表

□ **ARP** 表: 某些區域網路節點的 **IP/MAC** 位址對應

< **IP** 位址; **MAC** 位址; **TTL** >

- **TTL** (存活期): 位址對應會被遺忘的時間 (通常 20 分鐘)

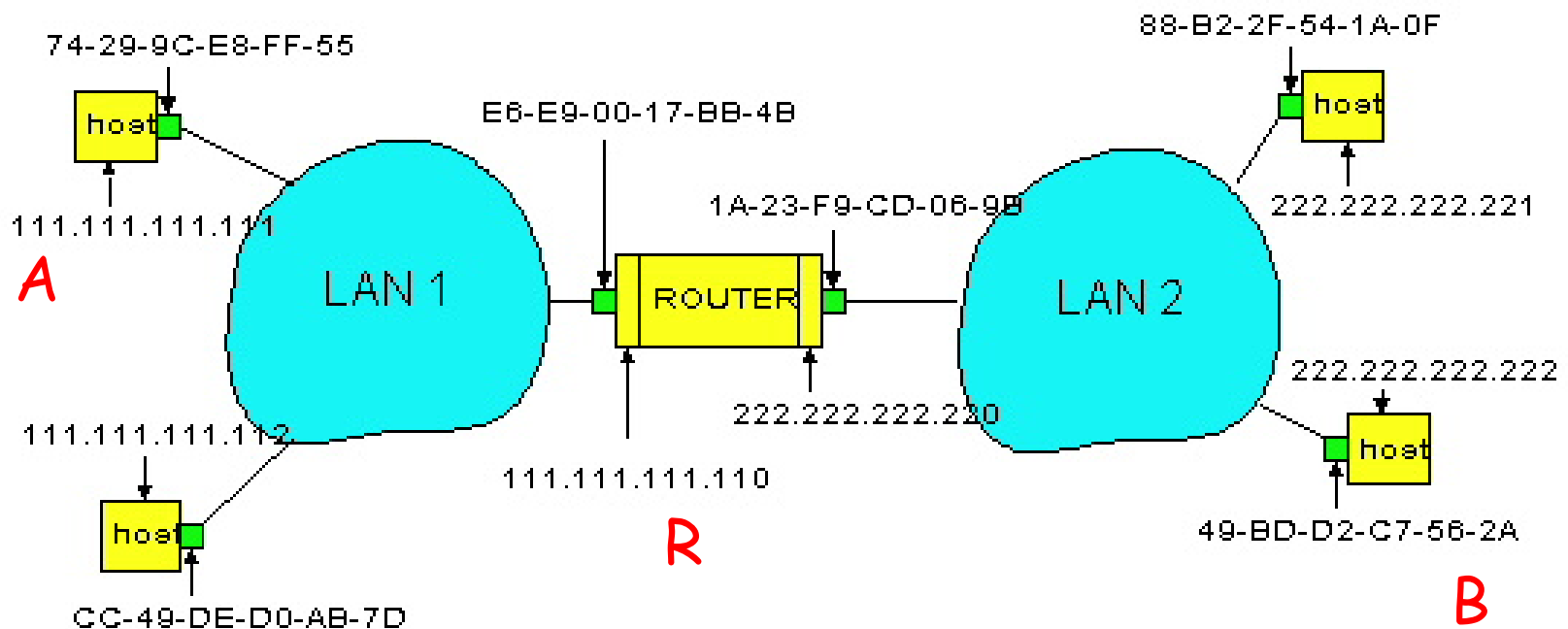
# ARP 協定: 同一個 LAN (網路)

- A 想要傳送封包給 B，B 的 **MAC** 位址不在 A 的 **ARP** 表中。
- A **廣播** **ARP** 查詢封包，內含 B 的 **IP** 位址
  - 目的端 **MAC** 位址 = FF-FF-FF-FF-FF-FF
  - **LAN** 上的所有機器都會收到 **ARP** 查詢
- B 收到 **ARP** 封包, 將它(B)的 **MAC** 位址回應給 A
  - 訊框傳送給 A 的 **MAC** 位址(單點傳播)
- A 在它的 **ARP** 表中儲存 **IP**-到-**MAC** 的位址對，直到資訊變舊 (逾時)。
  - 軟式狀態: 資訊會逾時 (離開) 除非被更新
- **ARP** 為 “隨插即用”的:
  - 節點在沒有網路管理者介入的狀況下建立它們的 **ARP**表

# 傳送到另外一個 LAN

走過: 經由 **R** 從 **A** 到 **B** 傳送資料段

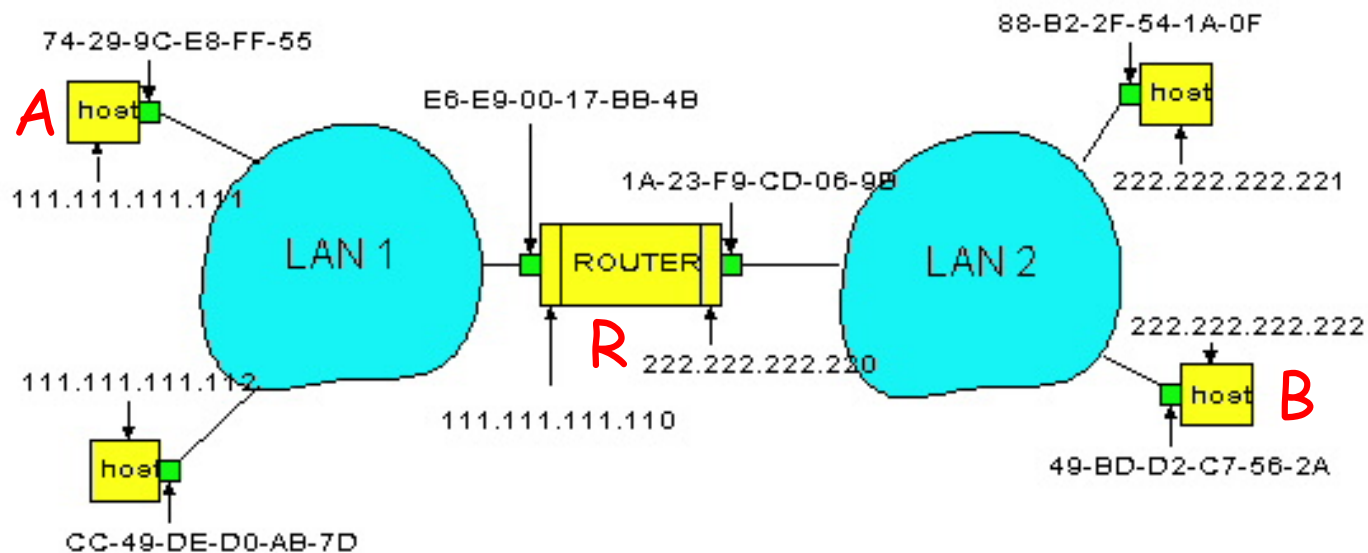
假設 **A** 知道 **B** 的 IP 位址



□ 路由器 **R** 中有兩個 **ARP** 表，一個 **IP** 網路 (**LAN**) 一個



- ❑ A 建立來源端為 A，目的端為 B 的資料段
- ❑ A 使用 ARP，為 111.111.111.110 取得 R 的 MAC 位址
- ❑ A 使用 R 的 MAC 位址做為目的端，建立連結層訊框，訊框內包含了 A 到 B 的 IP 資料段
- ❑ A 的網路卡傳送訊框
- ❑ R 的網路卡接收訊框
- ❑ R 從乙太網路訊框移除 IP 資料段，看到它的目的端是到 B
- ❑ R 使用 ARP 取得 B 的 MAC 位址
- ❑ R 建立包含 A 到 B 的 IP 資料段的訊框，傳送給 B



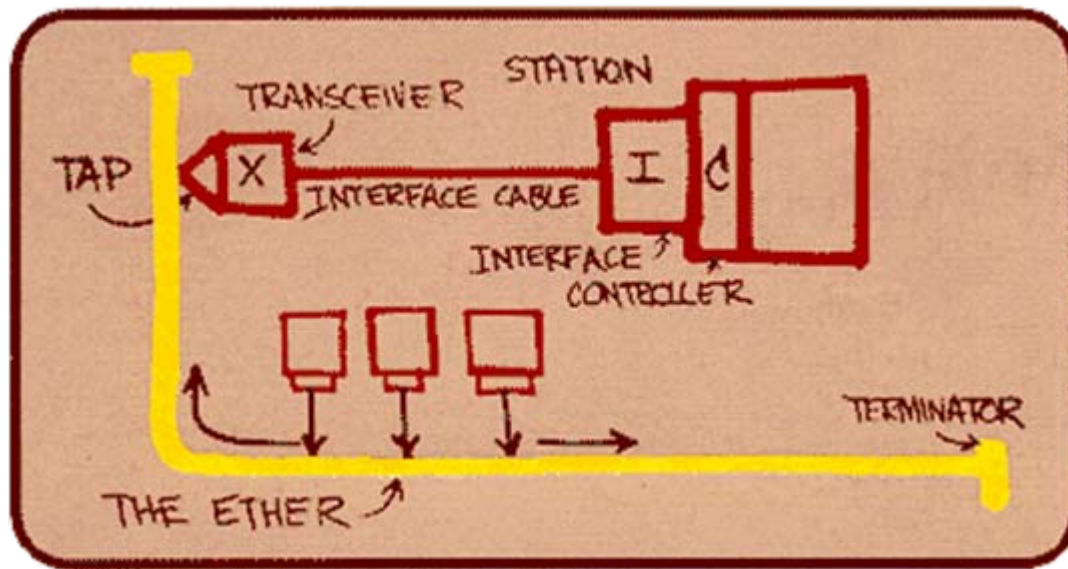
# 第五章:連結層和區域網路

- 5.1 簡介與服務
- 5.2 錯誤偵測和更正技術
- 5.3 多重存取協定
- 5.4 連結層定址
- 5.5 乙太網路
- 5.6 集線器和交換器
- 5.7 PPP
- 5.8 虛擬連結: **ATM** 與 **MPLS**

# 乙太網路

“主要的” 有線區域網路技術:

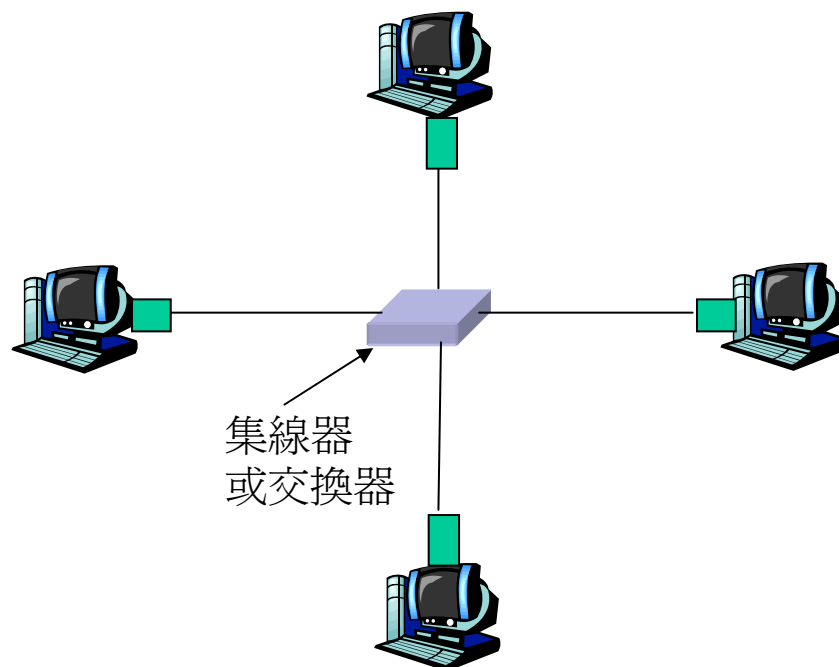
- ❑ 便宜， **100Mbps** 只要 **\$20!**
- ❑ 第一個廣泛使用的區域網路技術
- ❑ 比記號環區域網路以及**ATM** 簡單，便宜
- ❑ 速度不斷地提升: **10 Mbps - 10 Gbps**



Metcalfe's 的乙太網路  
設計圖

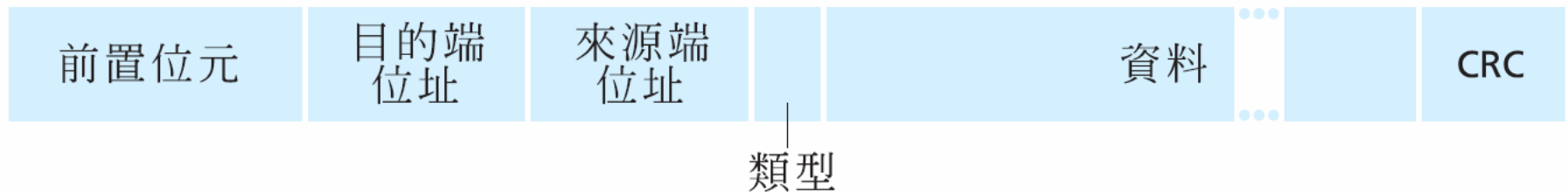
# 星狀拓樸

- ❑ 在1990年代中期，匯流排拓樸是非常普及的
- ❑ 目前流行的是星狀拓樸
- ❑ 選擇連接方式: 集線器或交換器 (後面再介紹)



# 乙太網路訊框結構

傳送端網路卡將 **IP** 資料段 (或是其他的網路層協定封包) 封裝在**乙太網路訊框**中

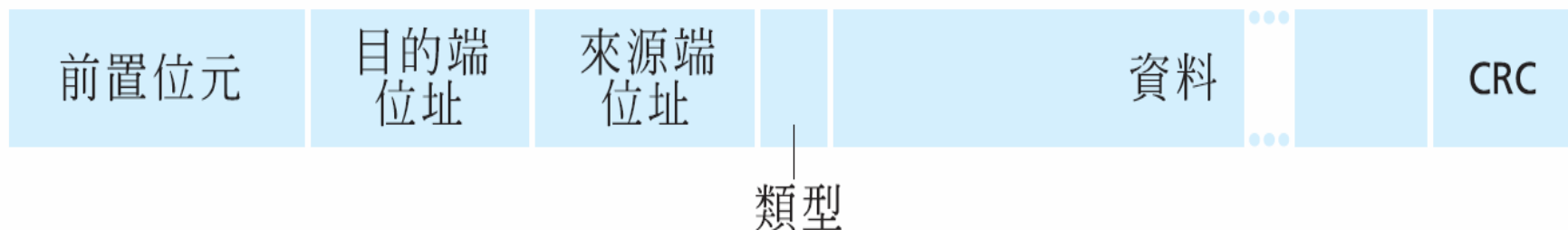


**前置:**

- ❑ 一個位元的**10101011**之後有**7**個位元組的**10101010**
- ❑ 用來同步傳送端和接收端的時脈速率

# 乙太網路訊框結構 (更多)

- **位址:** 6 個位元組
  - 假如網路卡收到與目的端位址相符的訊框或是廣播位址訊框(例如 **ARP**封包)，它會將訊框中的資料傳送到網路層協定
  - 否則，網路卡會刪除訊框
- **類型:**表示上一層的協定 (通常是 **IP** 或是其他可能支援的協定，如 **Novell IPX** 和 **AppleTalk**)
- **CRC:**接收端會檢查此一欄位, 如果錯誤就會丟棄此一訊框



# 不可靠的非預接式服務

- **非預接式:** 傳送端和接收端網路卡之間沒有交握程序
- **不可靠的:** 接收端網路卡不會傳送 **ack** 或是 **nack** 到傳送端網路卡
  - 傳送到網路層的資料段串流可能會出現間斷
  - 假如應用程式使用 **TCP**，間斷會被填滿
  - 否則，應用程式會看到間斷

# 乙太網路使用 CSMA/CD

- 沒有時槽
- 假如網路卡感測到其他網路卡正在傳送，則它不會傳送，也就是載波感測
- 假如傳送端網路卡感測到其他網路卡也在傳送，它會停止停止，也就是碰撞偵測
- 在嘗試重新傳送之前，網路卡會等待一段隨機時間，也就是隨機存取



# 乙太網路 CSMA/CD 演算法

1. 網路卡從網路層接收資料段並建立訊框
2. 假如網路卡感測通道是閒置的，它會開始傳送訊框。假如它感測到通道是忙碌的，它會等待通道閒置，接著開始傳送
3. 假如網路卡將整個訊框傳送出去而沒有偵測到其它傳送，則網路卡完成這個訊框的傳送！
4. 假如網路卡在傳送的同時，偵測到其他傳送，它會停止並且傳送一個擁擠訊號
5. 在停止之後，網路卡會進入**指數退回**：在第  $m$  次碰撞之後，網路卡會從  $\{0, 1, 2, \dots, 2^m - 1\}$  中隨機選擇  $K$ 。網路卡會等待  $K \cdot 512$  個位元時間後，再回到步驟 2

# 乙太網路的 CSMA/CD (更多)

**擁擠訊號:**用來保證所有的節點  
都知道發生碰撞, 48 bits

**位元時間:**對10 Mbps的乙太網路而言, 是0.1 毫秒;  
當  $K=1023$ , 等待時間約為 50 毫秒

請參考AWL網站上的Java  
applet:大力推薦!

**指數退回:**

- **目標:** 網路卡重送機制嘗試著估計目前的負載
  - 負載較重: 隨機等候會較久
- **第一次發生碰撞:** 從{0,1}中選出K; 延遲時間等於  $K \times 512$  位元的傳輸時間
- **第二次發生碰撞:** 從{0,1,2,3}選出K
- **在第十次碰撞以後,** 從{0,1,2,3,4,...,1023}選出K

# CSMA/CD 效能

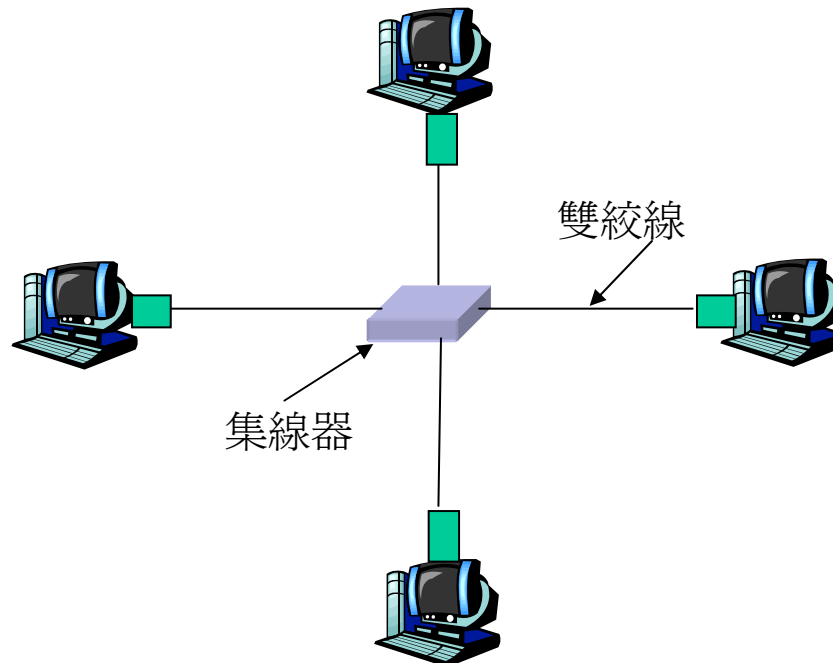
- $T_{prop}$  = LAN上兩個節點之間的最大傳播時間
- $t_{trans}$  = 傳送最大訊框所需要的時間

$$\text{效能} = \frac{1}{1 + 5t_{prop} / t_{trans}}$$

- 當  $t_{prop}$  接近 0 時，效能接近 1
- 當  $t_{trans}$  接近無限大時，效能接近 1
- 比 **ALOHA** 好很多，但仍為非集中式的，簡單且便宜

# 10BaseT 以及 100BaseT

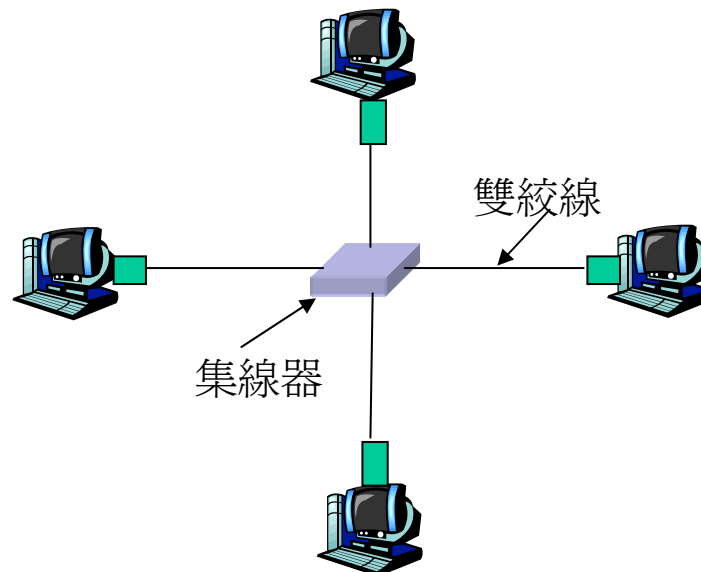
- ❑ 10/100 Mbps 的速率; 接下來稱為“高速乙太網路”
- ❑ T 代表雙絞線
- ❑ 連接到集線器的節點: “星狀拓樸”; 節點和集線器之間的距離最長**100** 公尺



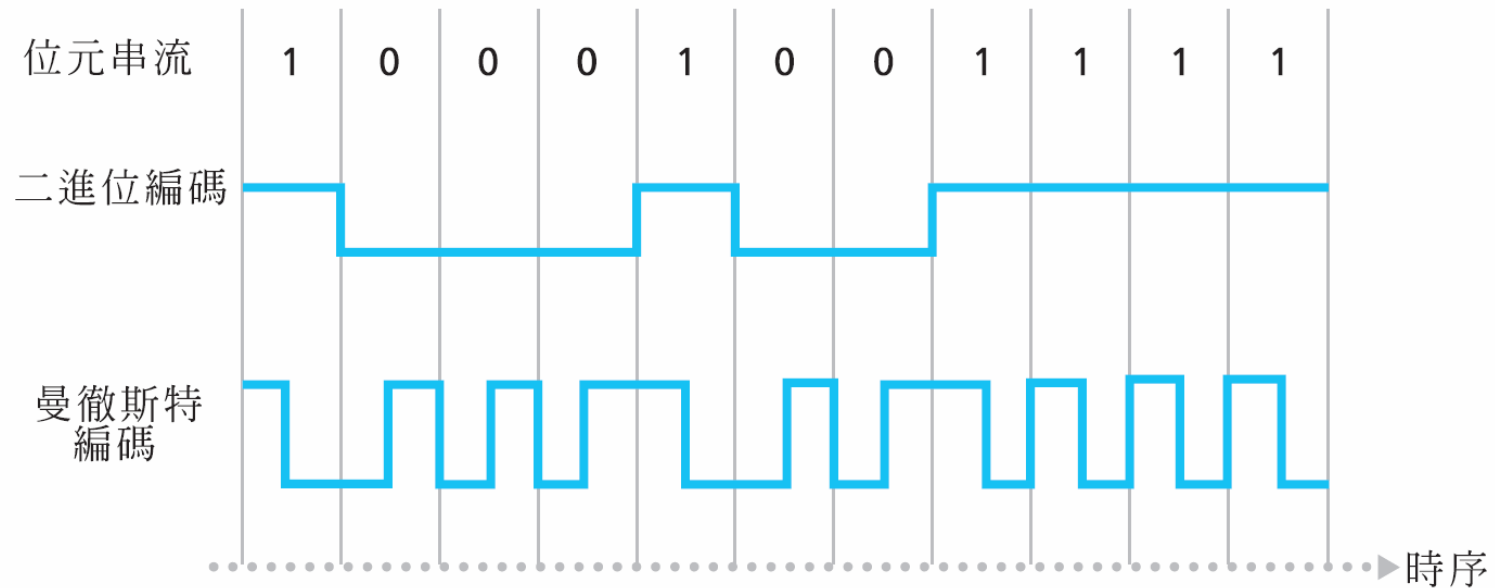
# 集線器

集線器主要是實體層的中繼器：

- 從一個連結進入的位元會被送出到所有其他的連結
- 以同樣的速率
- 沒有訊框緩衝區
- 集線器沒有 **CSMA/CD**：網路卡偵測碰撞
- 提供網路管理的功能



# 曼徹斯特編碼



- ❑ 使用在 10BaseT 中
- ❑ 每個位元都包含一個電位轉換
- ❑ 允許傳送端和接收端節點的時脈同步
  - 節點間不需要中央的，整體的時脈!
- ❑ 嘿，這是實體層的東西！

# Gbit 乙太網路

- ❑ 使用標準的乙太網路訊框格式
- ❑ 允許點對點連線和共享廣播頻道
- ❑ 在共享模式中，使用**CSMA/CD**，需要較短的節點距離以達到效能
- ❑ 使用集線器，這裡稱為「緩衝分配器」
- ❑ 允許在點對點連結中，以 **1 Gbps** 速率進行全雙工運作
- ❑ 目前**10 Gbps** !

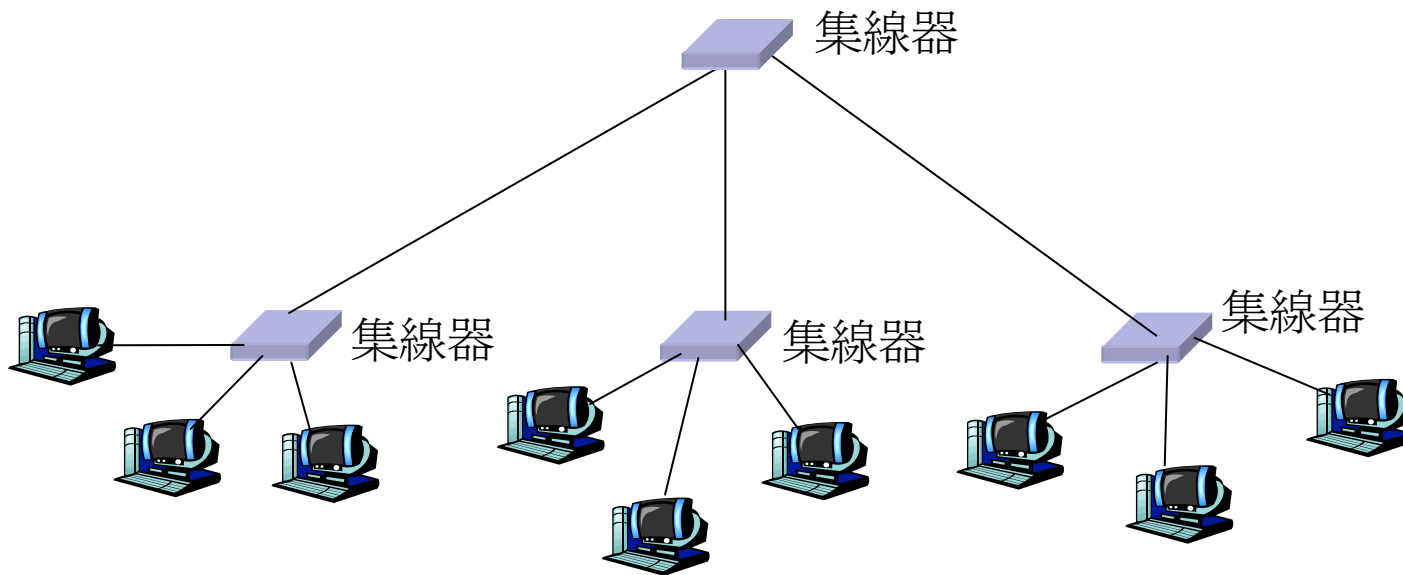
# 第五章:連結層和區域網路

- 5.1 簡介與服務
- 5.2 錯誤偵測和更正技術
- 5.3 多重存取協定
- 5.4 連結層定址
- 5.5 乙太網路
- 5.6 集線器和交換器
- 5.7 PPP
- 5.8 虛擬連結: ATM 與 MPLS



# 以集線器互相連結

- ❑ 主幹集線器連接區域網路的區段
- ❑ 延伸節點間的最大距離
- ❑ 但是個別的碰撞區域會變成一個大的碰撞區域
- ❑ 10BaseT 和 100BaseT 無法互相連結



# 交換器

## ❑ 連結層裝置

- 儲存及轉送乙太網路訊框
- 檢驗訊框以及選擇性地以**MAC**目的端位址轉送訊框
- 當訊框被轉送到區段時，使用 **CSMA/CD** 存取區段

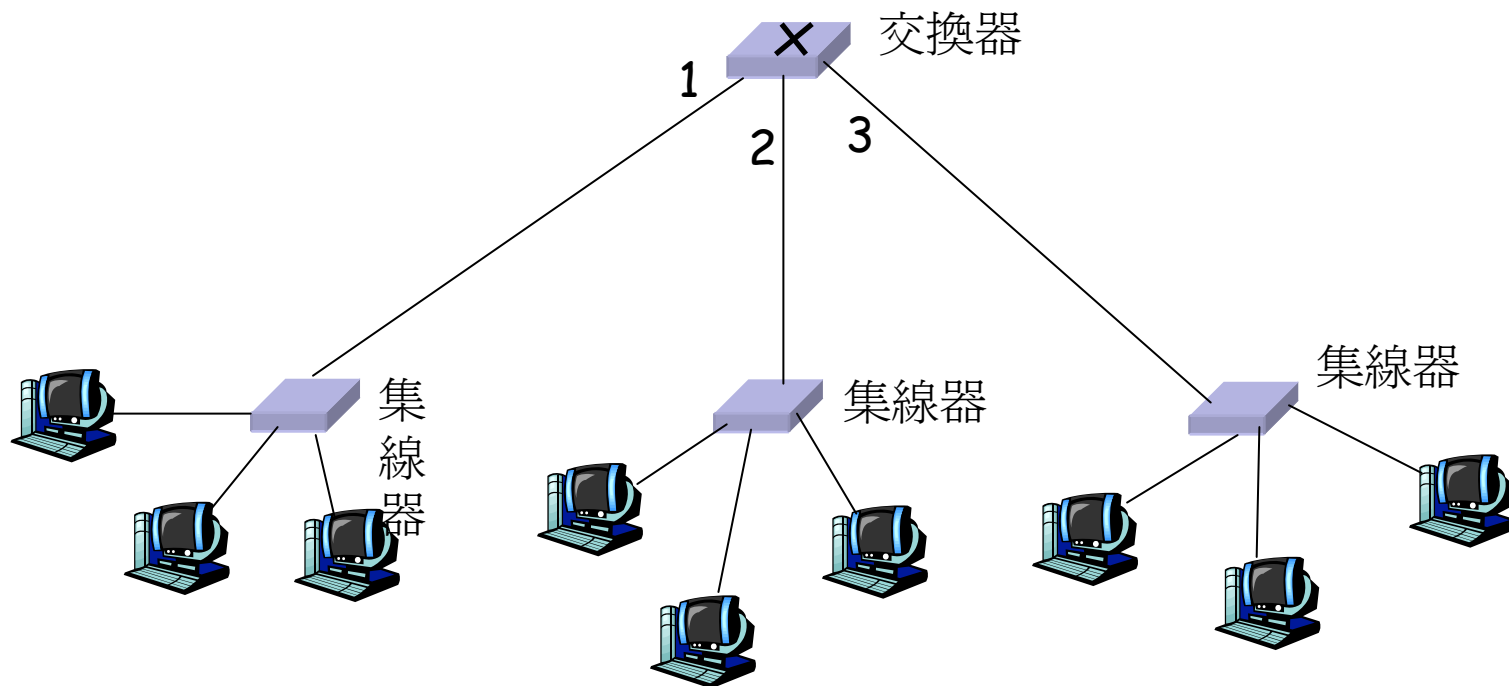
## ❑ 透明的

- 主機感覺不到交換器的存在

## ❑ 隨插即用，自我學習

- 交換器不需要設定

# 轉送



- 如何決定將訊框轉送到哪一個LAN區段?
- 看起來像是路由問題...

# 自我學習

- ❑ 交換器有一個**交換表**
- ❑ 交換表中的紀錄：
  - (**MAC** 位址, 介面, 時間標記)
  - 失去時效的紀錄會被丟棄 (**TTL** 為 60 分鐘)
- ❑ 交換器會**學習**哪一個主機經由哪一個介面可以到達
  - 當訊框抵達時，交換器會“學習”傳送端的位置: 進入的 **LAN** 區段
  - 在交換表中紀錄傳送端/位置對

# 過濾/轉送

當交換器收到一個訊框時:

索引的交換表使用 **MAC** 目的端位址

**if** 在紀錄裡找到目的端

**then{**

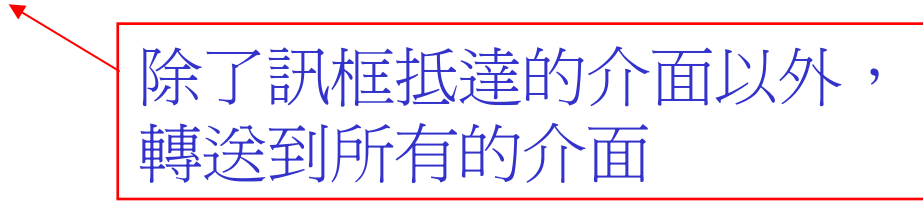
**if** 目的端位在訊框來自的區段

**then** 丟掉訊框

**else** 將訊框轉送到指定的介面

**}**

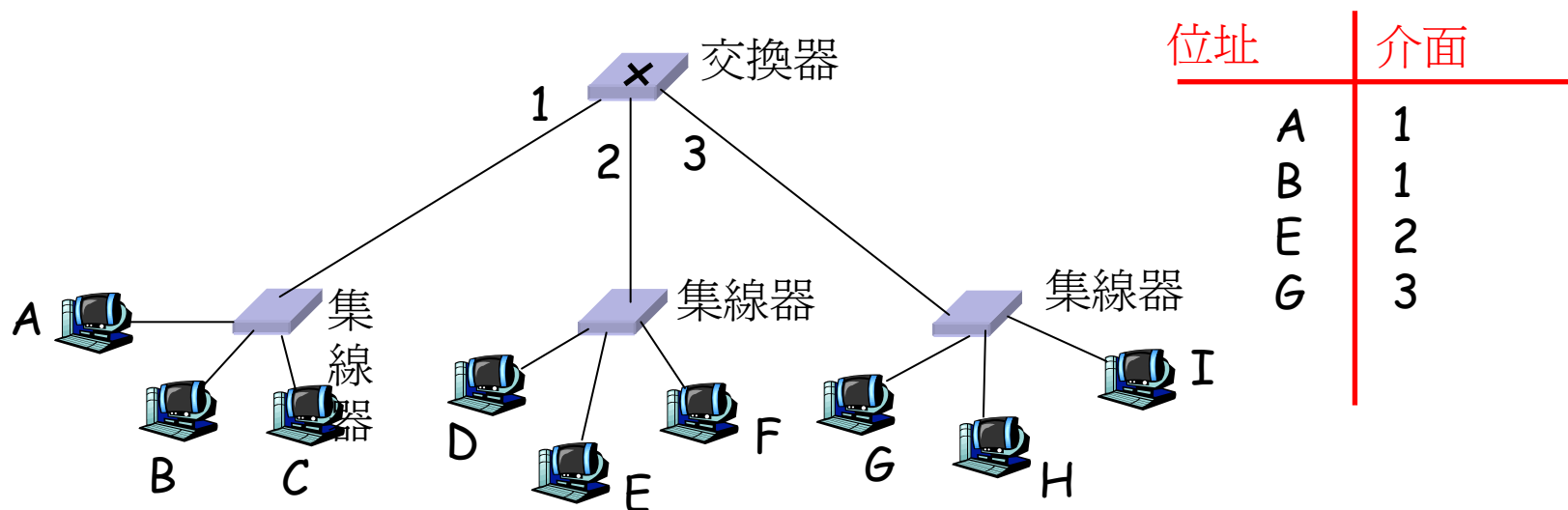
**else** 使用洪流法



除了訊框抵達的介面以外，  
轉送到所有的介面

# 交換器範例

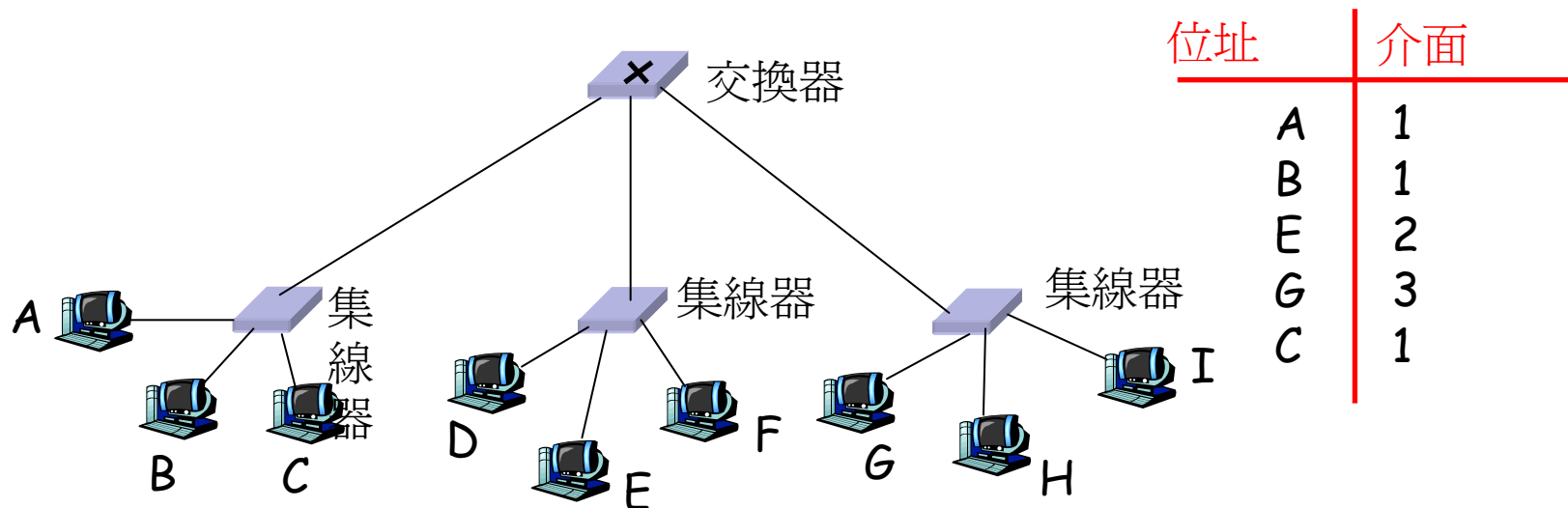
假設 **C** 傳送訊框到 **D**



- 交換器從 **C** 接收到訊框
  - 注意在轉接表中 **C** 位在介面 1
  - 因為 **D** 不在表中，交換器會將訊框轉送到介面 2 和 3
- **D** 收到訊框

# 交換器範例

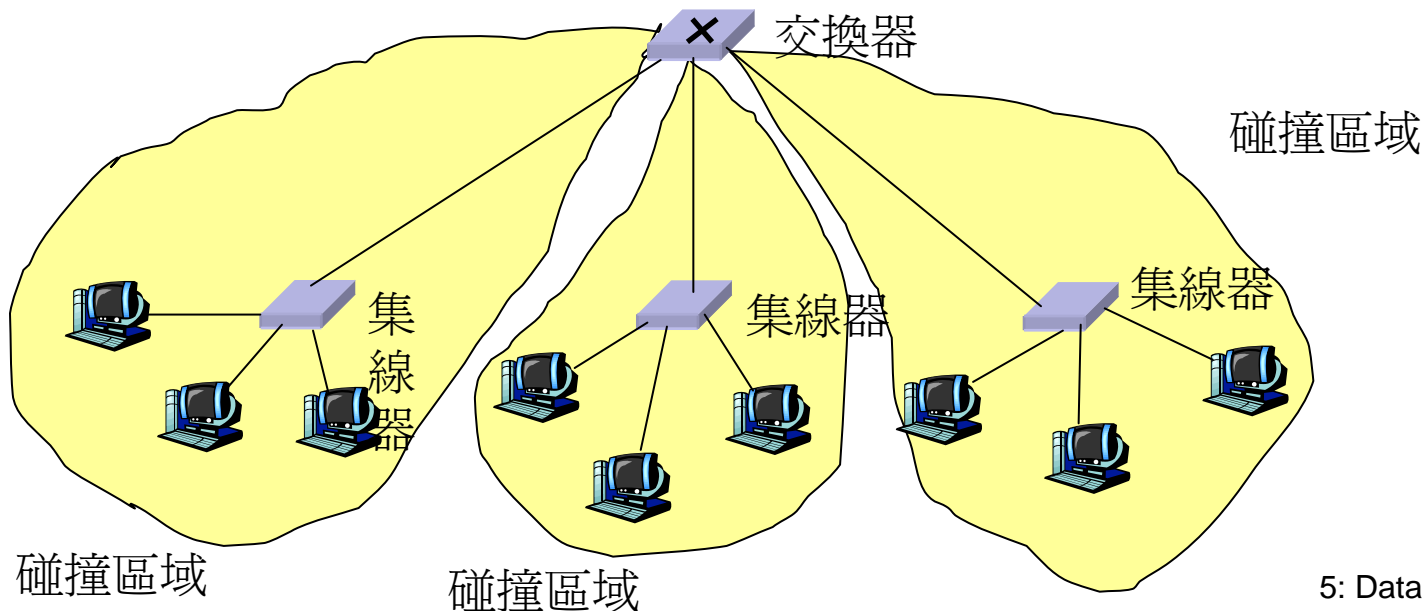
假設 **D** 將訊框回應給 **C**



- 交換器從 **D** 收到訊框
  - 注意在轉接表中 **D** 位在介面 2
  - 因為 **C** 在表中，交換器只會將訊框轉送到介面 1
- **C** 收到訊框

# 交換器：流量隔離

- ❑ 安裝交換器會將子網路分成區域網路區段
- ❑ 交換器過濾封包：
  - 同一個 LAN 區段中的訊框不會時常被轉送到其他的 LAN 區段中
  - 區段變成分離的碰撞區域

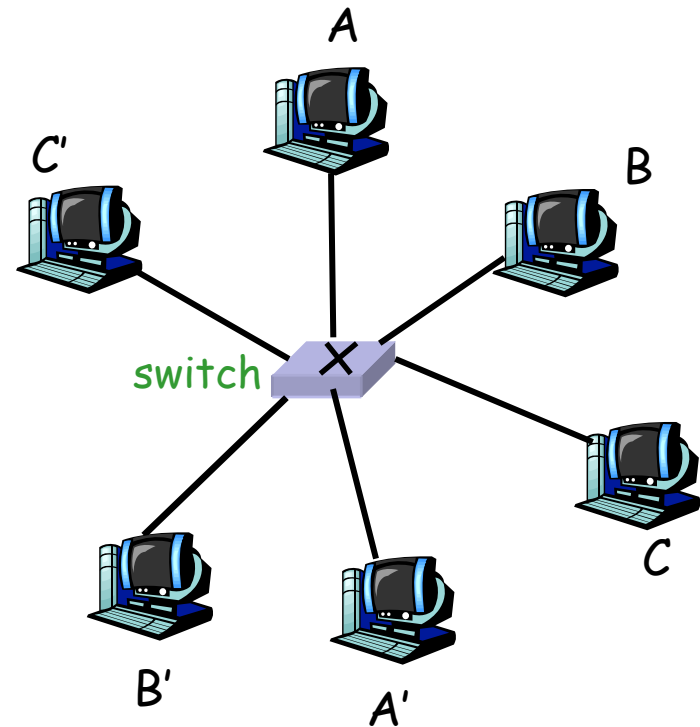




# 交換器：專線

- ❑ 有許多介面的交換器
- ❑ 主機直接連接到交換器
- ❑ 沒有碰撞：全雙工

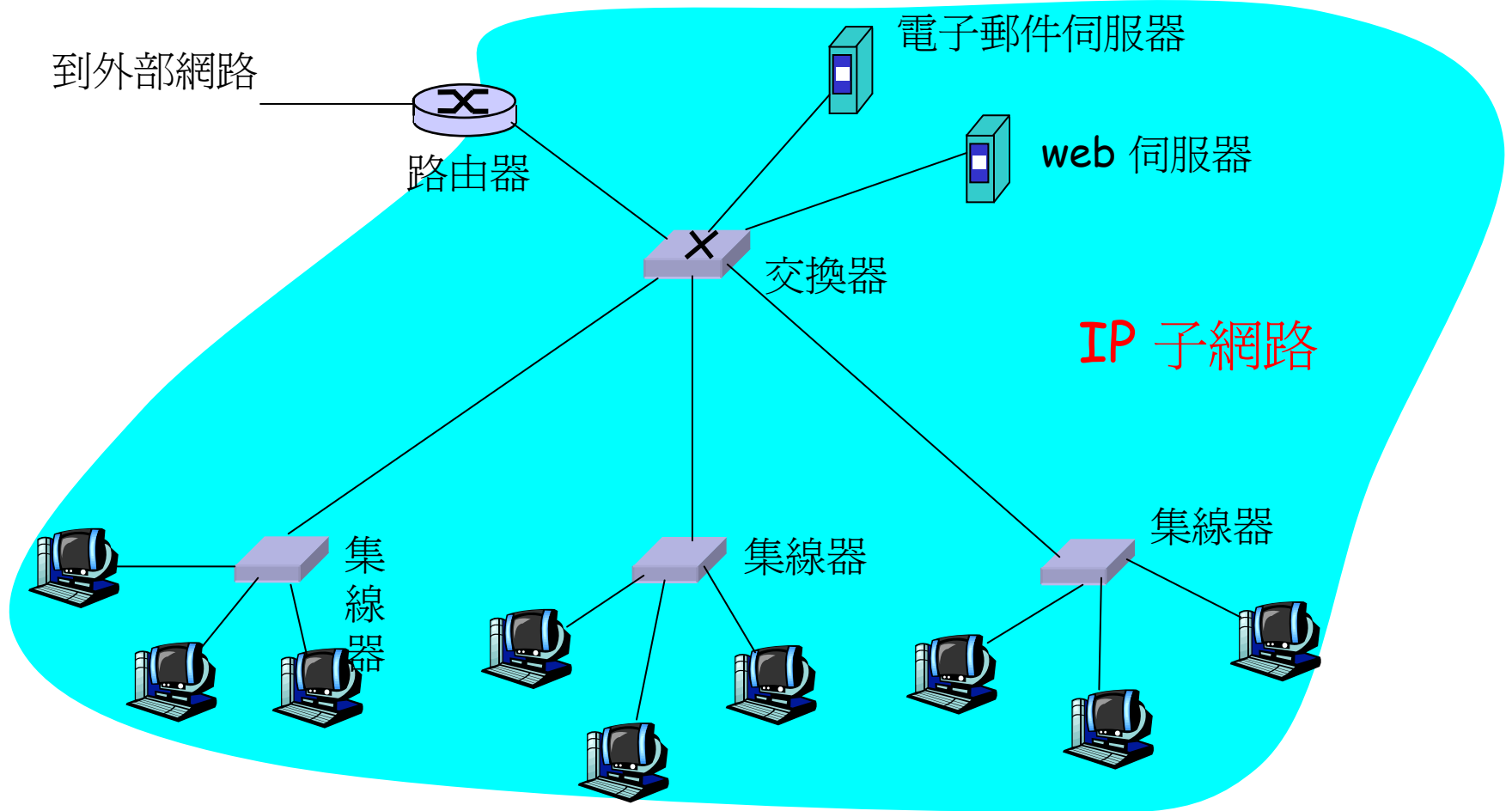
交換：A-到-A' 以及 B-到-B' 同時，沒有碰撞



# 更多關於交換器

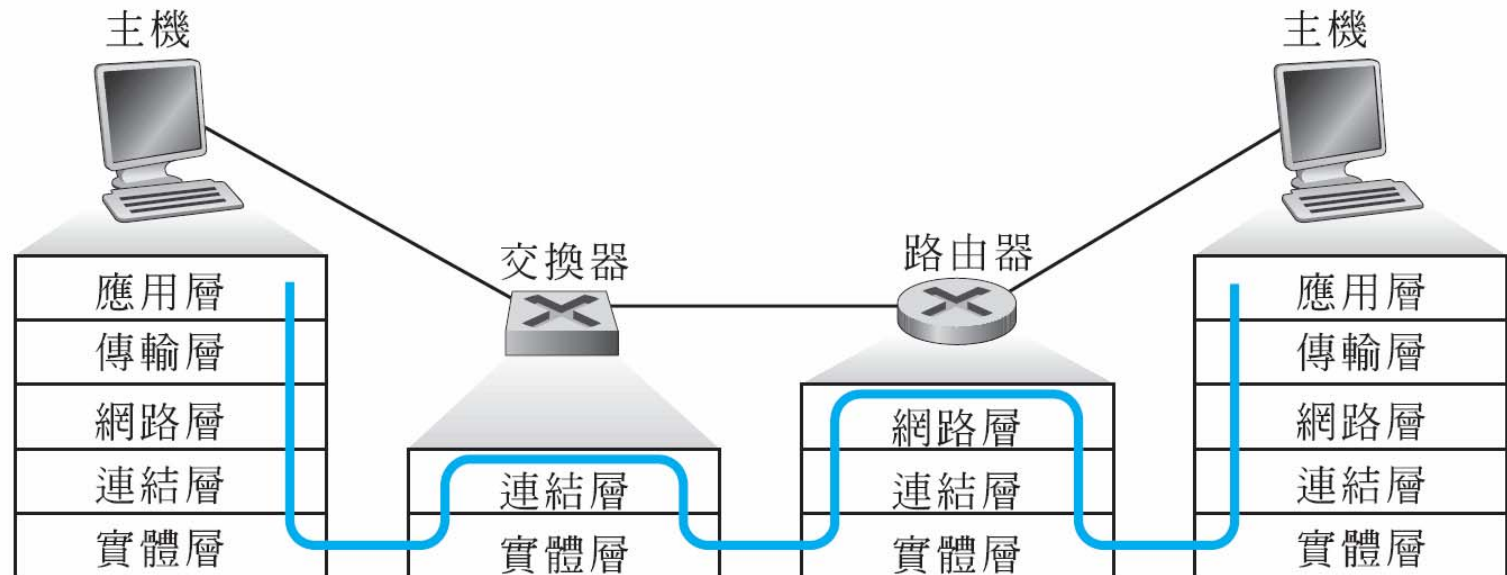
- ❑ **直接傳送交換法:** 訊框從輸入端到輸出端轉送出去，不會先收集整個訊框
  - 稍微減少延遲
- ❑ 分享/專線混合, 10/100/1000 Mbps 介面

# 機構網路



# 交換器 vs. 路由器

- 都是儲存並轉送的裝置
  - 路由器: 網路層裝置 (檢驗網路層標頭)
  - 交換器是連結層裝置
- 路由器維護路由表，實作路由演算法
- 交換器維護交換表，實作過濾，學習演算法



# 總結比較

## 集線器

## 路由器

## 交換器

流量隔離

無

有

有

---

隨插即用

有

無

有

---

最佳化路由

無

有

無

---

直接傳送

有

無

有

# 第五章:連結層和區域網路

- 5.1 簡介與服務
- 5.2 錯誤偵測和更正技術
- 5.3 多重存取協定
- 5.4 連結層定址
- 5.5 乙太網路
- 5.6 集線器和交換器
- 5.7 PPP
- 5.8 虛擬連結: ATM 與 MPLS

# 點對點資料連結層協定

- ❑ 一個傳送端，一個接收端，一個連結：比廣播連結簡單：
  - 不需要媒體存取控制
  - 不需要明確的**MAC**定址
  - 例如，撥接連結, **ISDN** 線路
- ❑ 流行的點對點 **DLC** 協定：
  - **PPP** (點對點協定)
  - **HDLC**: 高階資料連結控制 (使用的資料連結被認為是協定層中的「高階」！)

# PPP 設計需求 [RFC 1557]

- ❑ **封包訊框化**: 將網路層的資料段封裝在資料連結訊框中
  - 同時載送任何網路層協定(不僅是 **IP**) 的網路層資料
  - 朝上解多工的能力
- ❑ **位元通透性**: 必須載送資料欄位中的任何位元樣本
- ❑ **錯誤偵測** (沒有更正)
- ❑ **連線狀況**: 連結層故障的偵測，並通知網路層
- ❑ **網路層位址協商**: 端點可以學習/設定彼此的網路位址



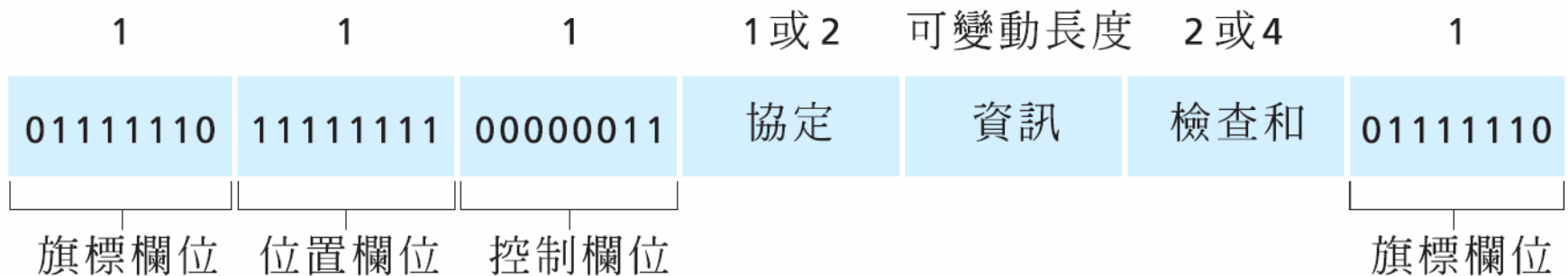
# PPP 不需要的

- ❑ 不需要錯誤更正/復原
- ❑ 不需要流量控制
- ❑ 傳送順序不正確 — OK
- ❑ 不需要支援多點連結 (例如,輪詢)

錯誤復原，流量控制，資料重新排序都歸類到更高層!

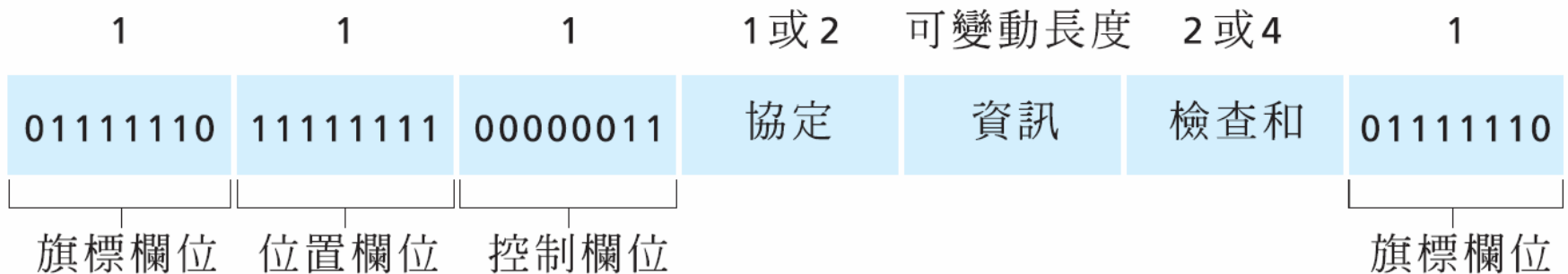
# PPP 資料訊框

- ❑ **旗標:** 定介符 (訊框)
- ❑ **位址:** 沒做什麼 (只有一個選擇)
- ❑ **控制:** 沒做什麼; 未來可能是多控制欄位
- ❑ **協定:** 訊框要傳送的上層協定 (例如, PPP-LCP, IP, IPCP, etc)



# PPP 資料訊框

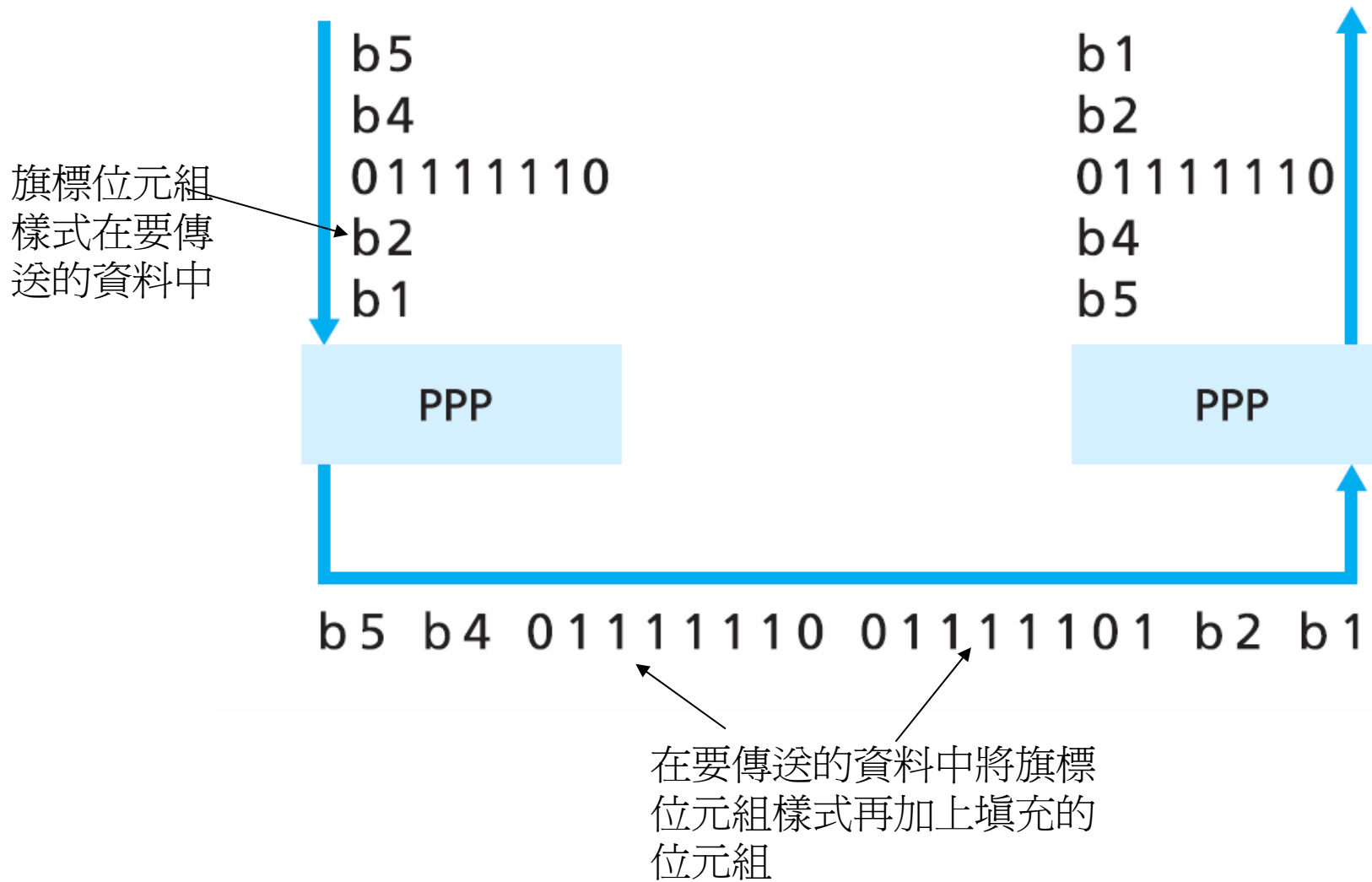
- **資料:** 載送的上層資料
- **檢查和:** 錯誤偵測的循環冗餘檢查



# 位元組填充

- “資料通透性” 要求: 資料欄位必須也允許旗標樣式 **<01111110>** 的存在
  - 問: 收到的 **<01111110>** 是資料還是旗標?
  
- 傳送端: 在每一個 **< 01111110>** **資料**位元組後面加入 (“填充”) 額外的 **< 01111110>** 位元組
- 接收端:
  - 連續收到兩個**<01111110>**, 就丟棄一個, 另一個當作是資料
  - 收到單一**<01111110>**, 視為旗標位元組

# 位元組填充

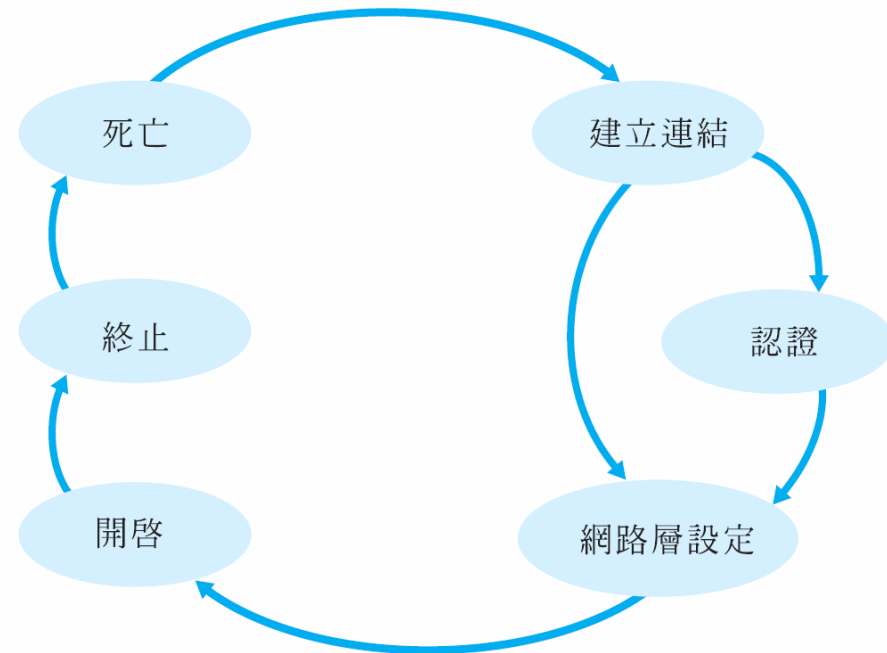


# PPP 資料控制協定

在傳輸網路層資料之前, 鏈結層  
必須先做以下的工作

- ❑ 設定PPP連線 (最大訊框長度, 認證資料)
- ❑ 學習/設定 網路層資訊

以IP來講, 載送IP控制協定訊息(IPCP)設定/學習IP位址



# 連結層

- 5.1 簡介與服務
- 5.2 錯誤偵測和更正技術
- 5.3 多重存取協定
- 5.4 連結層定址
- 5.5 乙太網路
- 5.6 集線器和交換器
- 5.7 PPP
- 5.8 虛擬連結: ATM 與 MPLS

# 虛擬網路

虛擬資源: 系統工程中功能強大的抽象化概念:

- ❑ 計算機的範例: 虛擬記憶體, 虛擬裝置
  - 虛擬機器: 例如, java
  - 1960/70 年代的 **IBM VM os**
- ❑ 抽象的階層: 不需要煩惱較低層的細節, 只需要抽象地處理低層



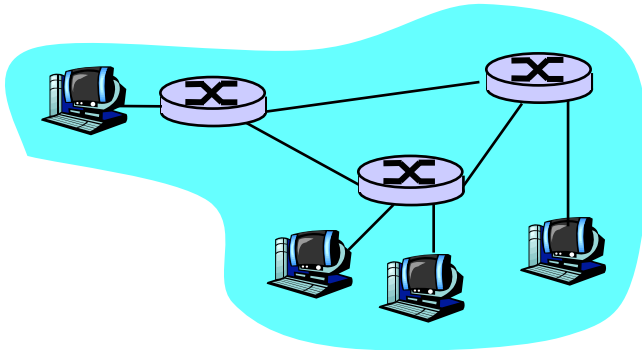
# 網際網路：虛擬網路

1974: 多個未連結的網路

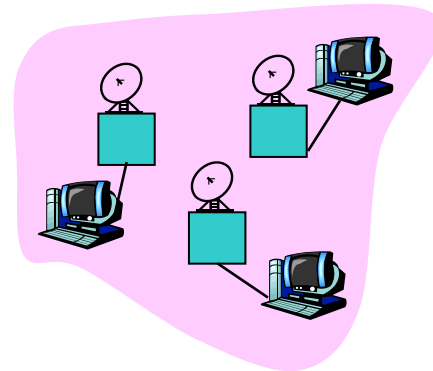
- ARPAnet
- 經由電纜傳輸的資料網路
- 封包衛星網路 (Aloha)
- 封包無線網路

... 相異處:

- 位址原則
- 封包格式
- 錯誤復原
- 路由



ARPAnet



衛星網路

"A Protocol for Packet Network Intercommunication",  
V. Cerf, R. Kahn, IEEE Transactions on Communications,  
May, 1974, pp. 637-648.

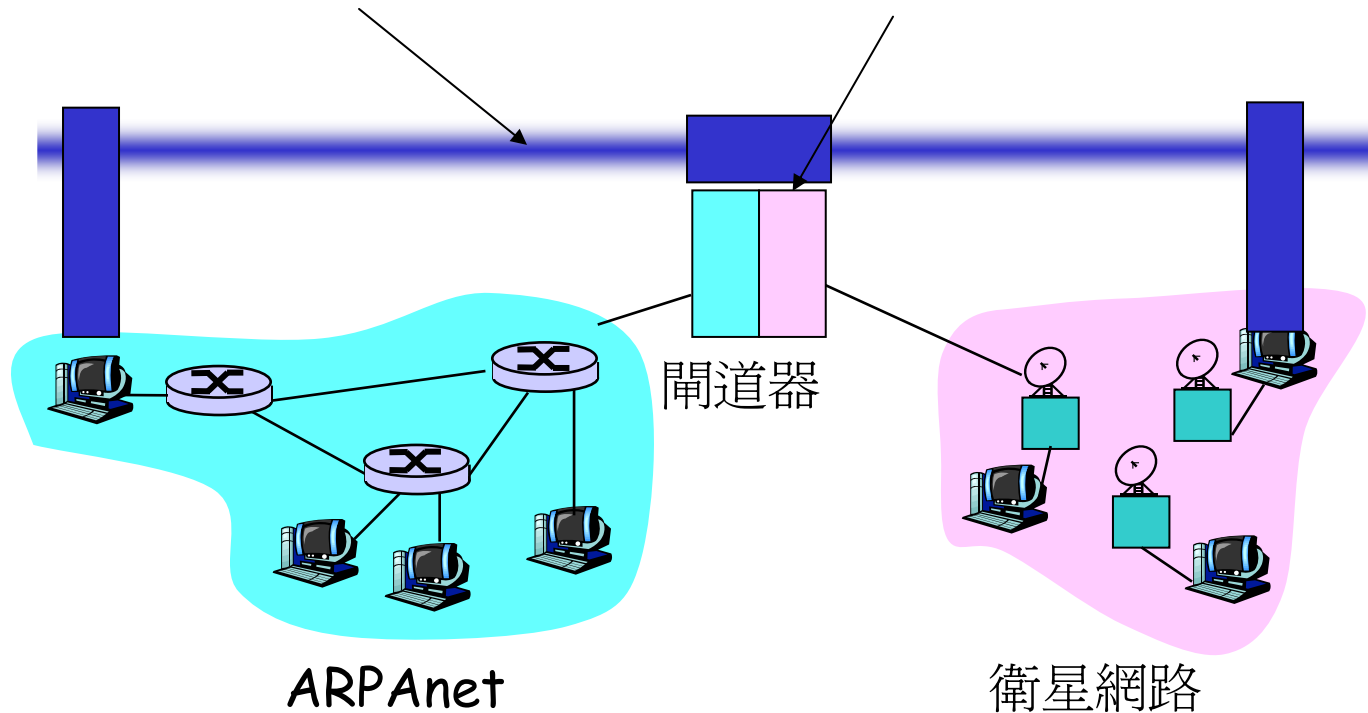
# 網際網路：虛擬網路

## 互連網路層 (IP):

- 位址：互連網路看起來好像一個單一的、統一的實體，無論下方的區域網路異質性
- 網路組成的網路

## 閘道器：

- 「將互連網路的封包嵌入區域封包格式中或是將它們解開」
- 路由（在互連網路層）到下一個閘道器



# Cerf & Kahn 的互連網路架構

什麼是虛擬的？

- 兩層位址：互連網路與區域網路
  - 新的階層 (**IP**) 使的互連網路層上的一切具同質性
  - 下層的區域網路技術
    - 電纜
    - 衛星
    - 56K 電話數據機
    - 今日: **ATM, MPLS**
- ... 互連網路層的“隱形” 。如同連結層技術到**IP**!

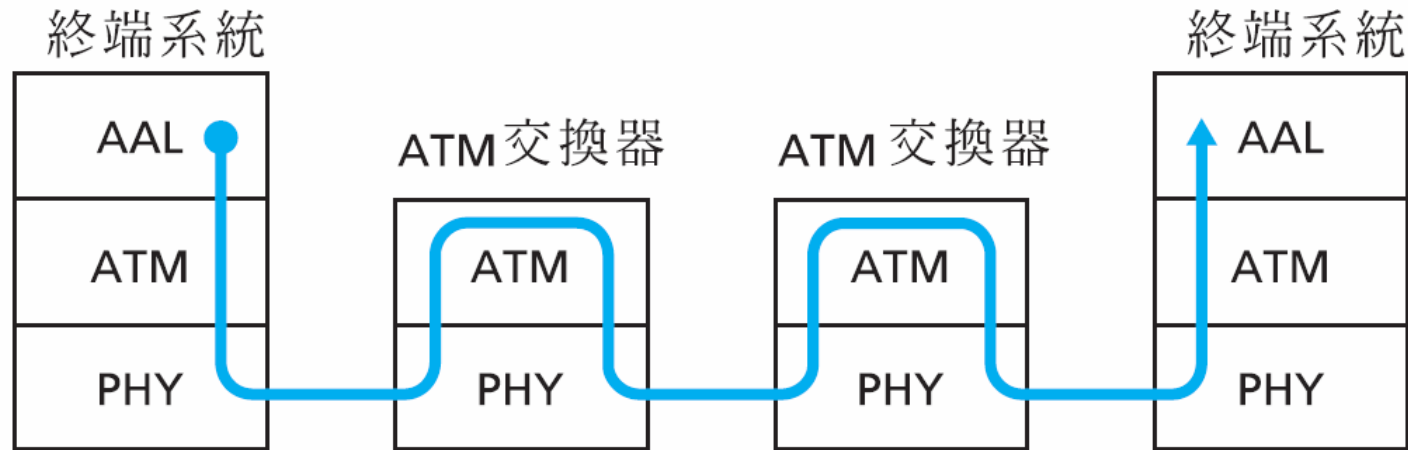
# ATM 以及 MPLS

- ❑ **ATM, MPLS** 以它們自己的專利來分離網路
  - 不同的服務模型, 定址, 來自網際網路的路由
- ❑ 網際網路將它們視為以 **IP** 路由器連接的邏輯連結
  - 如同撥接連結實際上是分離網路的一部分 (電話網路)
- ❑ **ATM, MPLS**: 它們自己的專利技術

# 非同步傳輸模式網路: ATM

- **1990/2000年代的高速標準 (155Mbps 到 622 Mbps 或更高)** 寬頻整合服務數位網路 架構
- **目標:** 整合的，端點對端點的傳輸，可載送音訊、視訊、資料
  - 符合音訊及視訊的時序/QoS需求 (v.s. 網際網路的盡全力傳送模型)
  - “下一代” 電話: 根源於電話世界的技術
  - 封包交換 (固定長度的封包，稱為「細胞」)，使用虛擬線路

# ATM 架構



- **轉接層:** 在**ATM**網路的邊緣
  - 資料分段/重組
  - 大約類比於網際網路傳輸層
- **ATM 層:** “網路”層
  - 細胞交換, 路由
- **實體層**

# ATM: 網路或連結層？

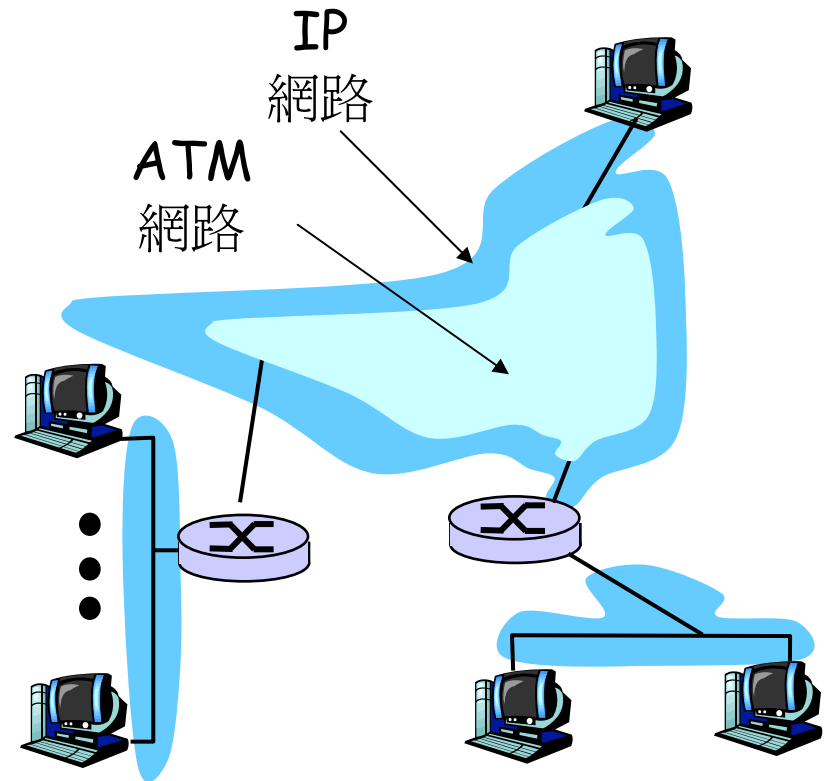
可見的: 端點對端點的傳輸:

“ATM 從桌上型電腦到桌上型電腦”

- ATM 為網路層技術

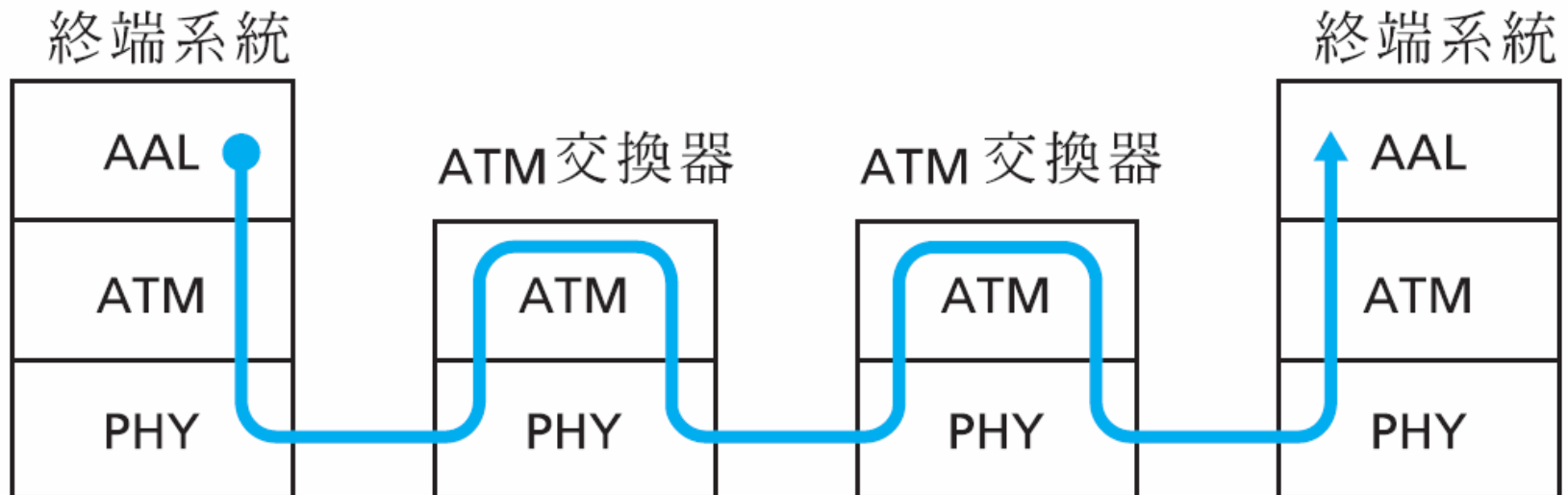
實際的: 用來連接 IP 主幹路由器

- “ATM 上的 IP”
- ATM 如同交換的連結層，連接 IP 路由器



# ATM 轉接層 (AAL)

- **ATM 轉接層 (AAL):** “轉接” 上層 (IP 或本地 **ATM** 應用程式) 到下層的 **ATM** 層
- **AAL** 只在終端系統中出現，不會出現在交換器中
- **AAL** 層的資料分段 (標頭/標尾欄位, 資料) 切割橫越多個 **ATM** 細胞
  - 比方: **TCP** 資料分段在許多 **IP** 封包中

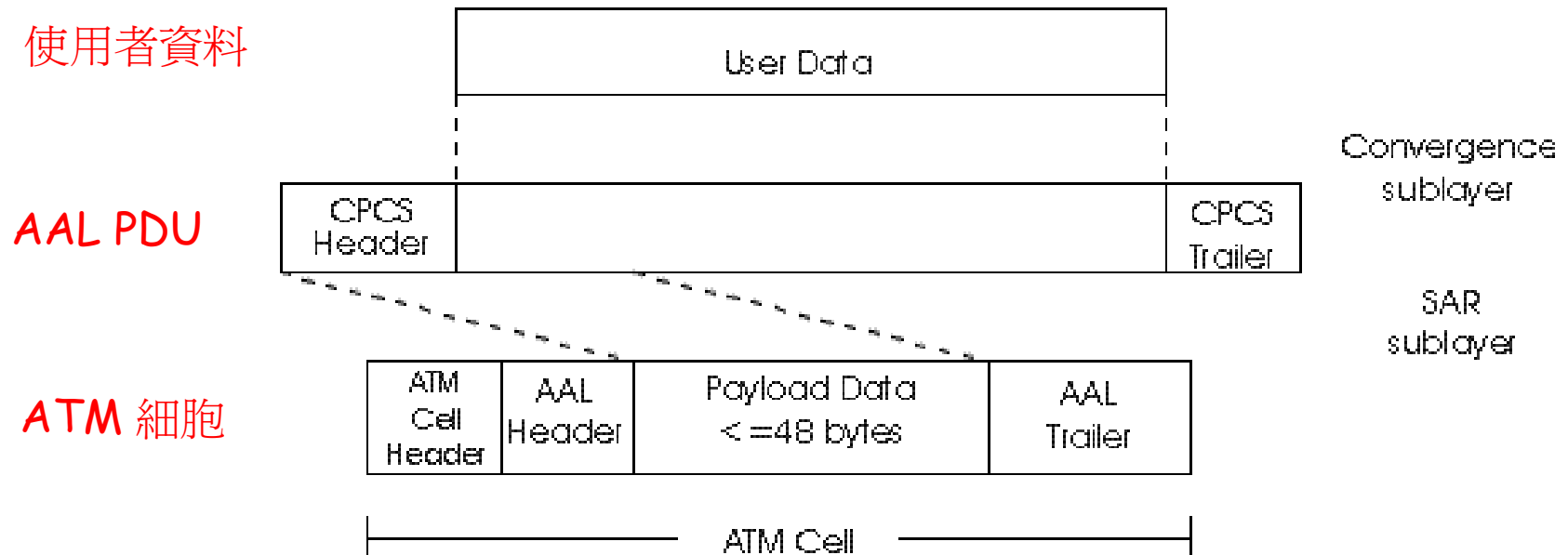




# ATM 轉接層 (AAL) [更多]

不同版本的 **AAL** 層，視 **ATM** 服務等級而定：

- ❑ **AAL1**: 用在 **CBR** (固定位元速率) 服務，例如，線路模擬
- ❑ **AAL2**: 用在 **VBR** (變動位元速率) 服務，例如，**MPEG** 視訊
- ❑ **AAL5**: 用在資料 (例如 **IP** 資料段)



# ATM 層

服務: 傳輸細胞橫越 ATM 網路

- 類比於 IP 網路層
- 與 IP 網路層非常不同的服務

網路 架構	服務 模型	保證 ?				壅塞 指示
		頻寬	遺失	排序	時序	
網際網路	盡全力 服務	無	無	無	無	無 (藉由遺失 來推論)
ATM	CBR	常數速率	有	有	有	不發生壅塞
ATM	VBR	保證速率	有	有	有	不發生壅塞
ATM	ABR	保證最小 速率	無	有	無	有
ATM	UBR	無	無	有	無	無

# ATM 層: 虛擬線路

- **VC 傳輸:** 從來源端到目的端的細胞在 **VC** 中被載送
  - 在傳送資料流之前，會先建立連線，拆除連線
  - 每個封包都載送 **VC** 識別碼 (非目的**ID**)
  - 在來源對目的的路徑上，每個路由器都必須維護每個經過的連線的「狀態」
  - 連結和路由器的資源 (頻寬、緩衝區) 可能會配置給**VC**: 得到類似線路的效果
- **永久性 VC (PVC)**
  - 長期持續的連結
  - 通常是: **IP**路由器間的「永久」路由
- **交換式 VC (SVC):**
  - 每次連線都需要動態建立

# ATM VC

## □ ATM VC 方法的優點:

- 對應到VC的連線具有 QoS 的效能保證 (頻寬，延遲，延遲時基誤差)

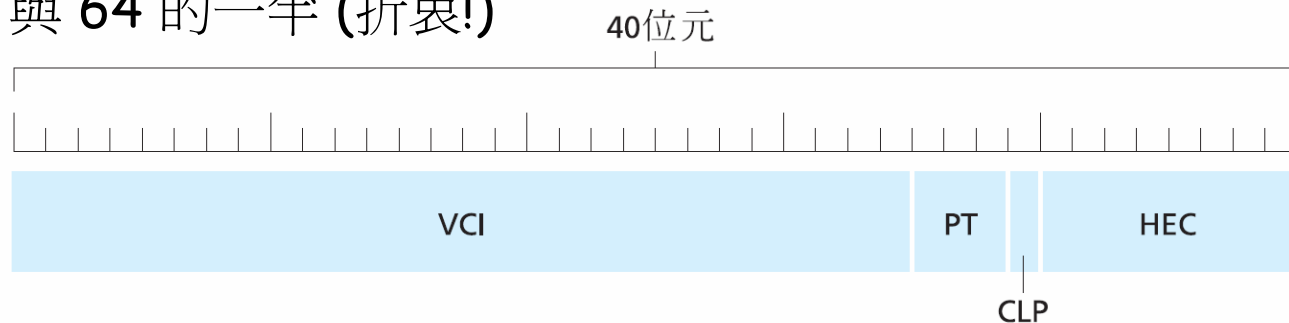
## □ ATM VC 方法的缺點:

- 資料段的資料流支援不足
- 每個來源端/目的端之間，一個 PVC無法擴充 (需要  $N*2$  連結)
- SVC 產生連線建立的延遲，處理短期連線的額外負擔

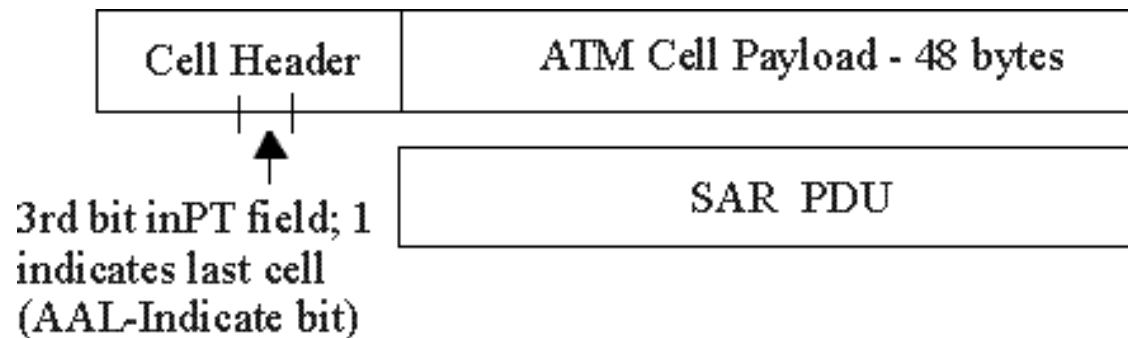
# ATM 層: ATM 細胞

- ❑ 5 位元組的 **ATM** 細胞標頭
- ❑ 48 位元組的資料
  - 為什麼?: 較小的資料 -> 數位音訊的較短細胞建立延遲
  - 32 與 64 的一半 (折衷!)

細胞標頭

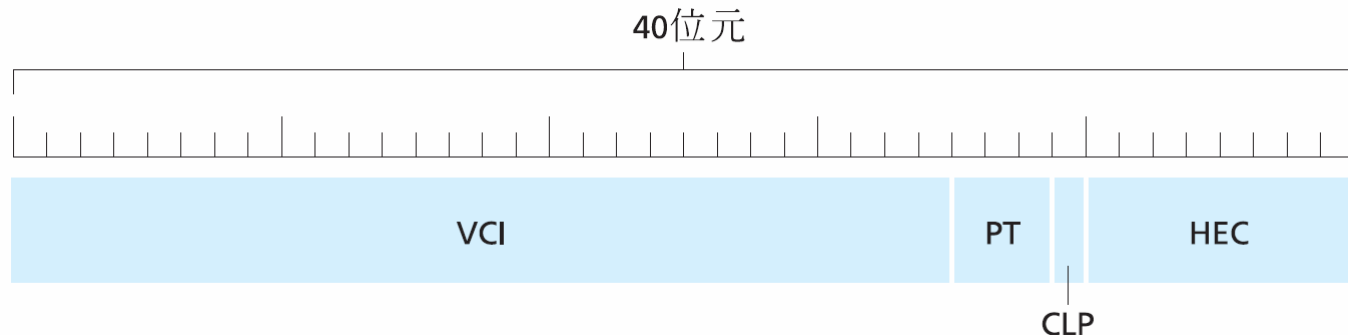


細胞格式



# ATM 細胞標頭

- **VCI:** 虛擬通道 ID
  - 經過網路時，會隨著連結到連結改變
- **PT:** 承載資料型態 (例如 **RM** 細胞 v.s. 資料細胞)
- **CLP:** 細胞遺失優先權位元
  - **CLP = 1** 為低優先權細胞，可以在壅塞時被刪除
- **HEC:** 標頭錯誤檢查和Header Error Checksum
  - 循環冗餘檢查



# ATM 實體層 (更多)

實體層的兩塊 (子層)：

- ❑ 傳輸收斂子層 (Transmission Convergence Sublayer, TCS)：調節 ATM 層和下方的 PMD 子層
- ❑ 實體媒體相關子層 (Physical Medium Dependent)：和所使用的實際媒介有關

TCS 功能：

- 標頭檢查和產生 8 位元的 CRC
- 細胞區隔
- 在“無結構的” PMD 子層中，當沒有資料細胞要傳送時，會傳送閒置細胞

# ATM 實體層

## 實體媒體相關子層 (PMD)

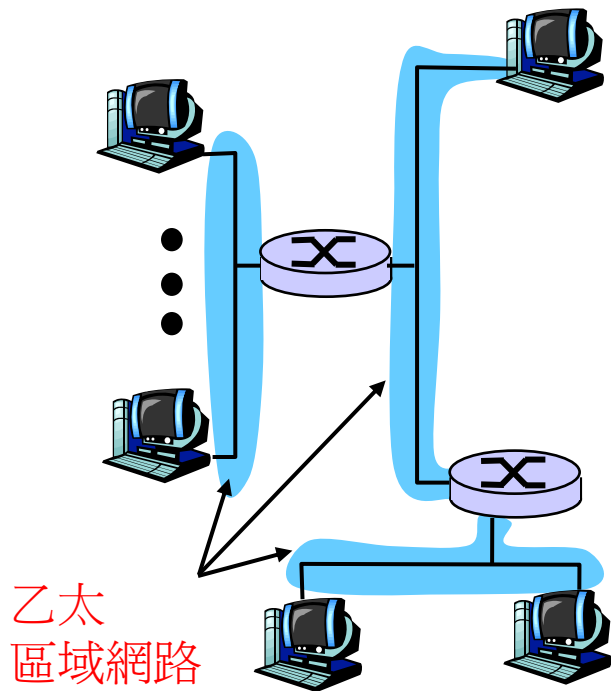
- **SONET/SDH:** 傳輸訊框的結構 (如同載送位元的容器):
  - 位元同步;
  - 頻寬的分割 (TDM);
  - 有不同速度: OC3 = 155.52 Mbps; OC12 = 622.08 Mbps; OC48 = 2.45 Gbps, OC192 = 9.6 Gbps
- **TI/T3:** 訊框結構傳輸 (舊的電化階層): 1.5 Mbps/ 45 Mbps
- **無結構的:** 只有細胞 (忙碌/閒置)



# ATM 上的 IP

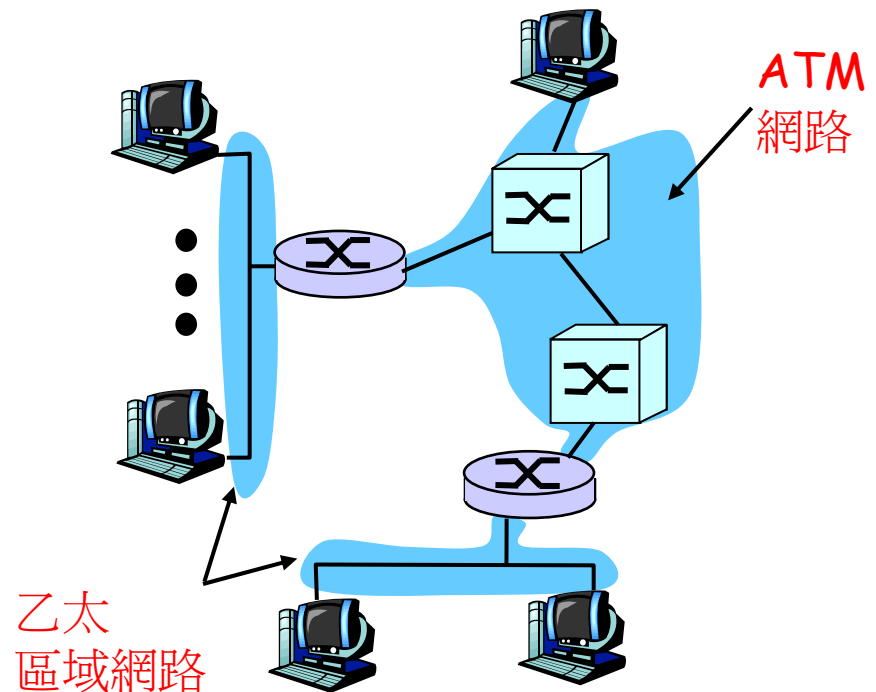
只有傳統 IP

- 3 個“網路” (例如, LAN 區段)
- MAC (802.3) 和 IP 位址

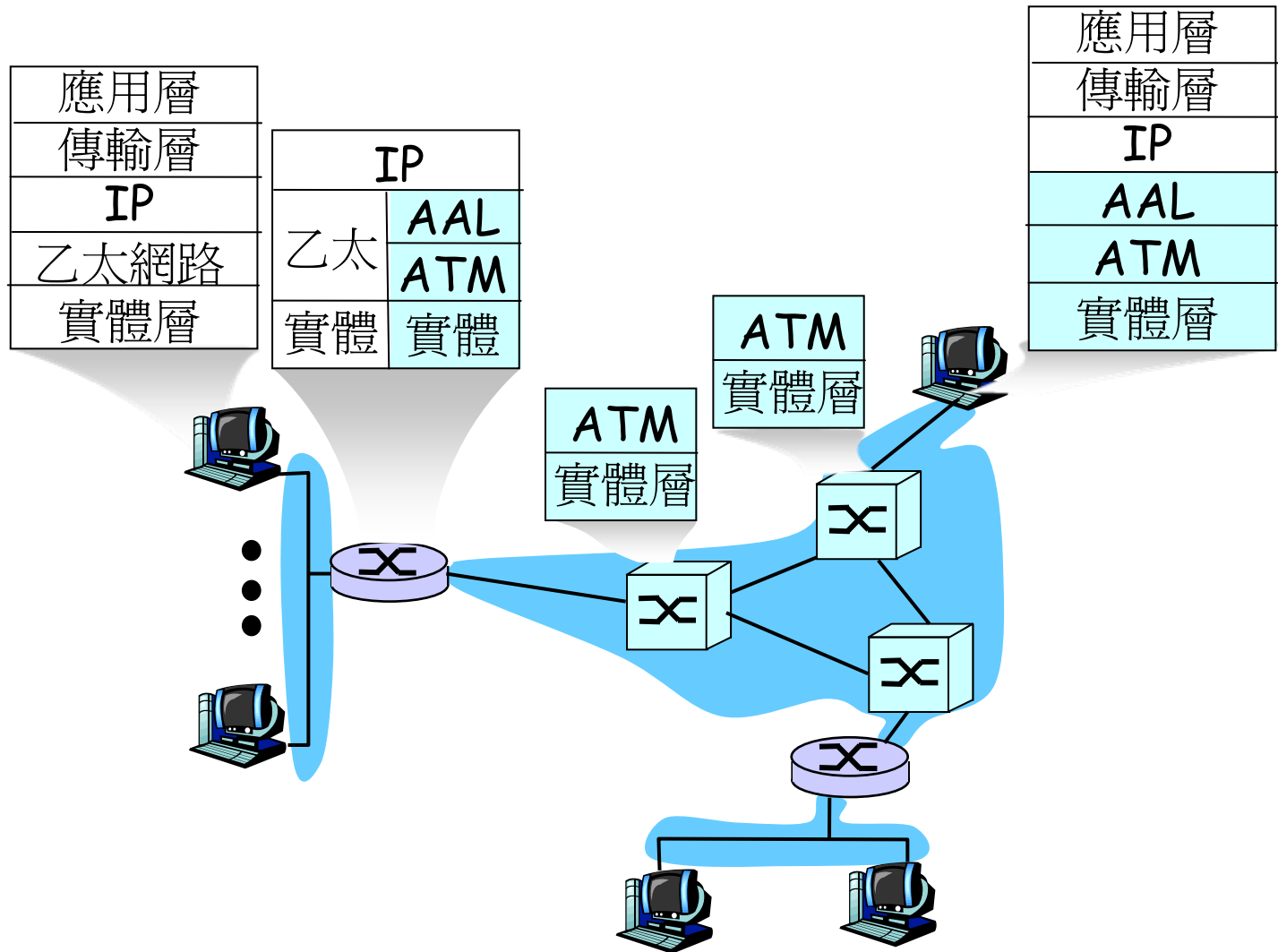


ATM 上的 IP

- 以 ATM 網路更換 “網路” (例如, LAN 區段)
- ATM 位址, IP 位址



# ATM 上的 IP



# ATM 上的 IP 網路中資料段的旅行

## □ 在來源端主機:

- IP 層在 IP，ATM 目的位址之間對應 (使用 ARP)
- 將資料段傳道 AAL5
- AAL5 封裝資料，資料段細胞，傳送給 ATM 層

## □ ATM 網路: 沿著 VC 移動細胞到目的端

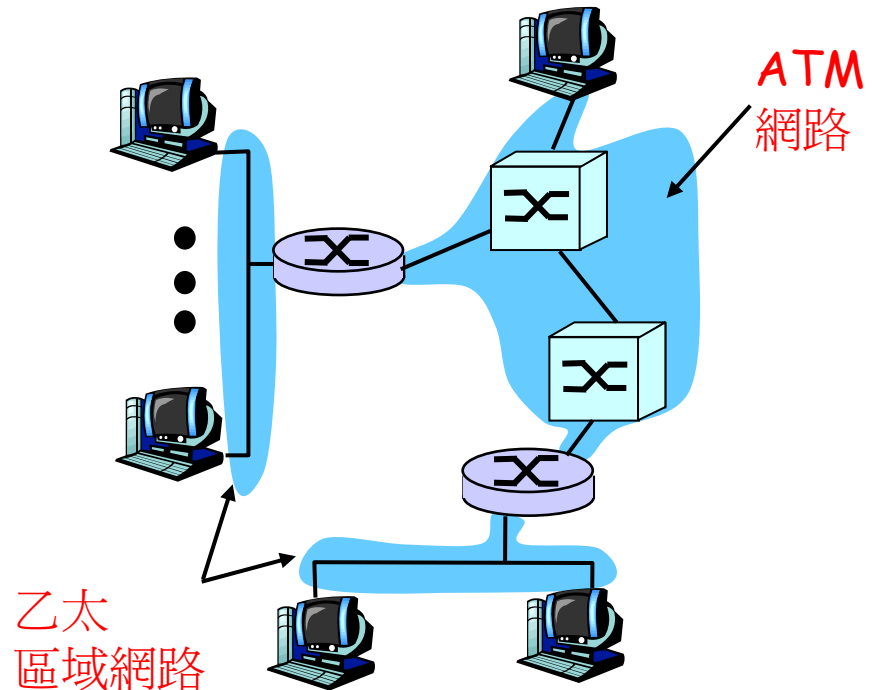
## □ 在接收端主機:

- AAL5 將細胞重組成資料段
- 假如 CRC OK，資料段會傳送給 IP

# ATM 上的 IP

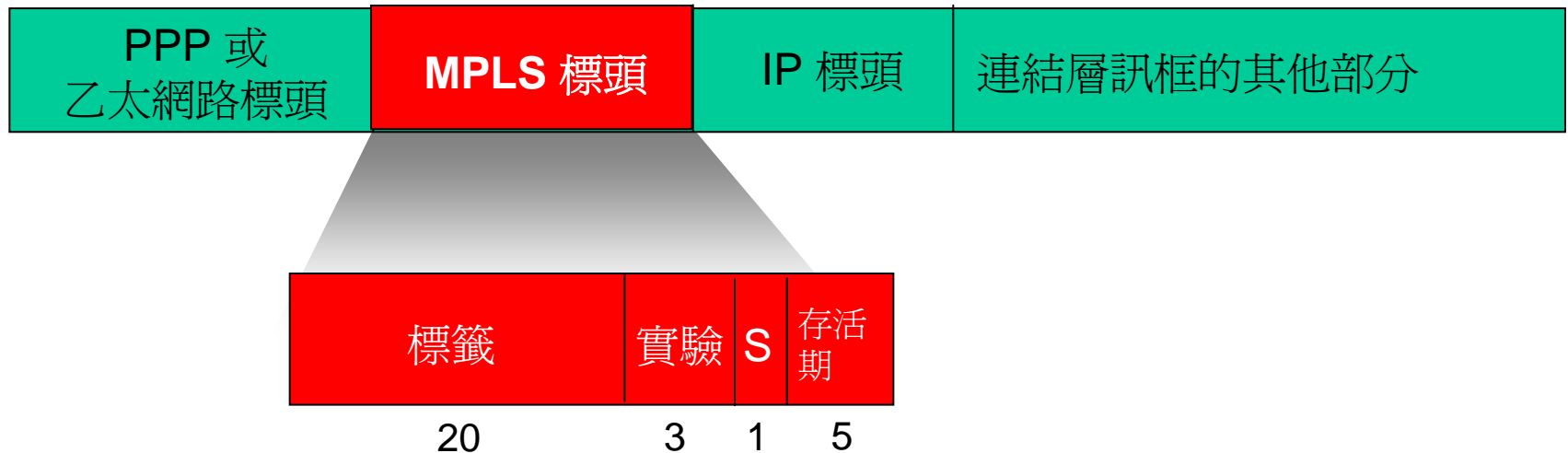
## 問題:

- ❑ IP 資料段進入 ATM AAL5 PDU
- ❑ 從 IP 位址到 ATM 位址
  - 如同 IP 位址到 802.3 MAC 位址!



# 多重協定標籤交換 (MPLS)

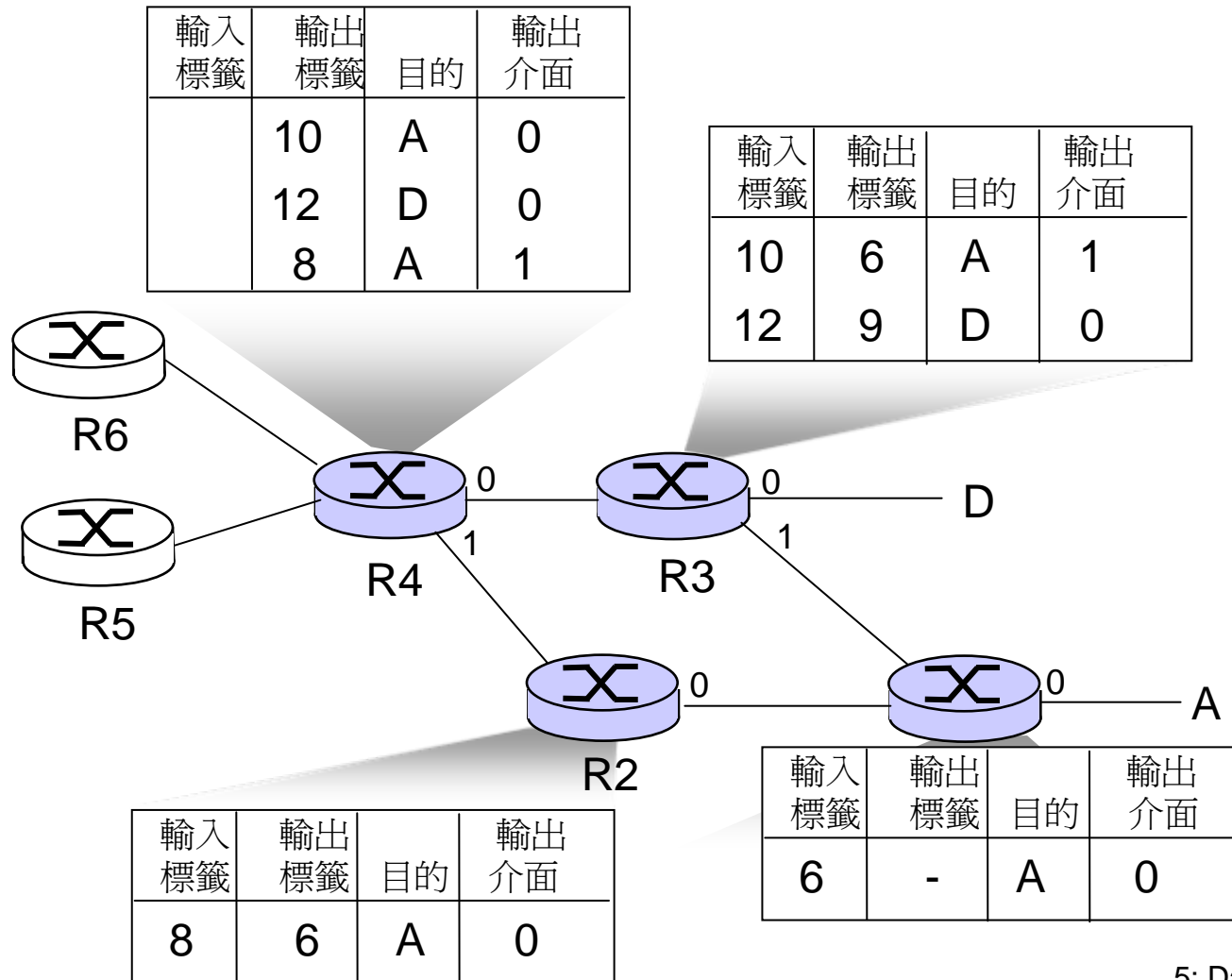
- ❑ 初始目標: 使用固定長度的標籤加快 **IP** 的轉送 (替代 **IP** 位址)
  - 從虛擬線路 (**VC**) 得來的方法
  - 但是 **IP** 資料段仍然保持 **IP** 位址!



# 可執行MPLS的路由器

- ❑ 也稱為標籤交換路由器
- ❑ 只使用標籤值(不檢查**IP**位址)，將封包轉送到輸出介面
  - **MPLS** 轉送表與 **IP** 轉送表不同
- ❑ 訊號協定需要設定轉送
  - **RSVP-TE**
  - 轉送可能會沿著只有 **IP** 時不允許的路徑 (例如, 來源指定的路由) !!
  - 使用 **MPLS** 做流量工程
- ❑ 需要與只執行 **IP** 的路由器共存

# MPLS 轉送表



# 第五章：總結

- ❑ 資料連結層服務的原則：
  - 錯誤偵測, 更正
  - 分享寬頻通道: 多種存取
  - 連結層位址
- ❑ 各種連結層技術的例證和實作
  - 乙太網路
  - 交換區域網路
  - PPP
  - 連結層的虛擬網路: **ATM, MPLS**