

計算機網路概論



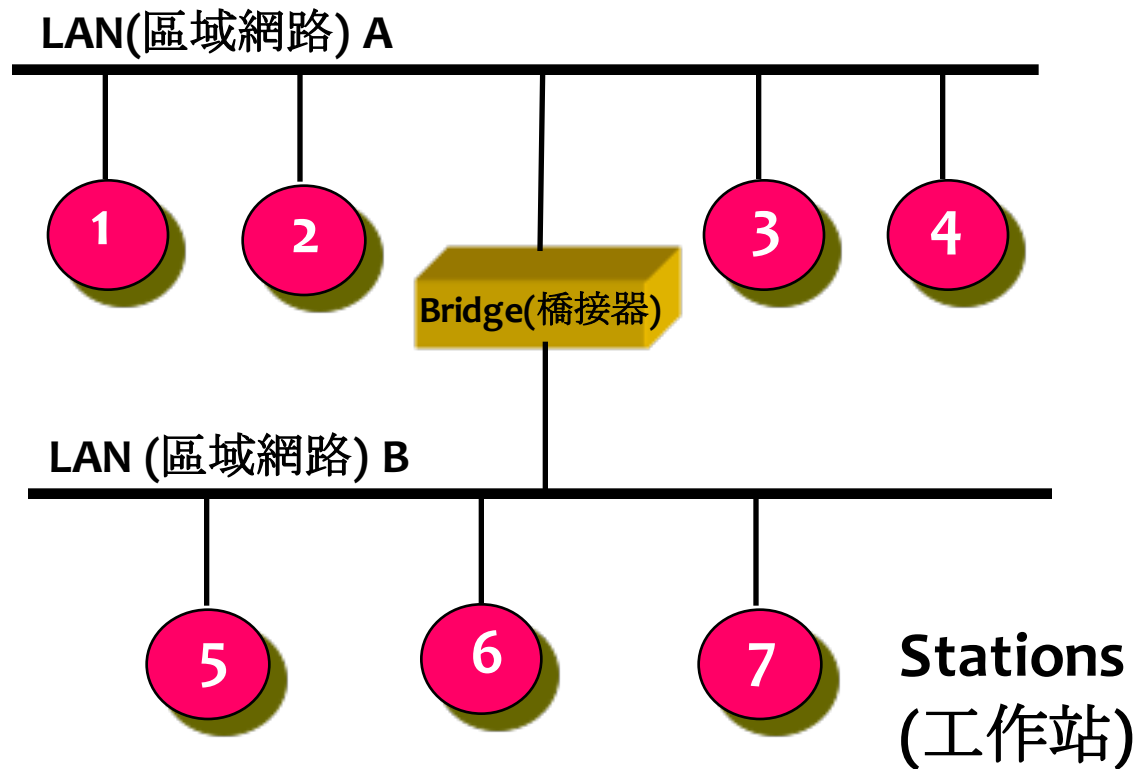
IEEE 802.1D 擴張樹演算法

© 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

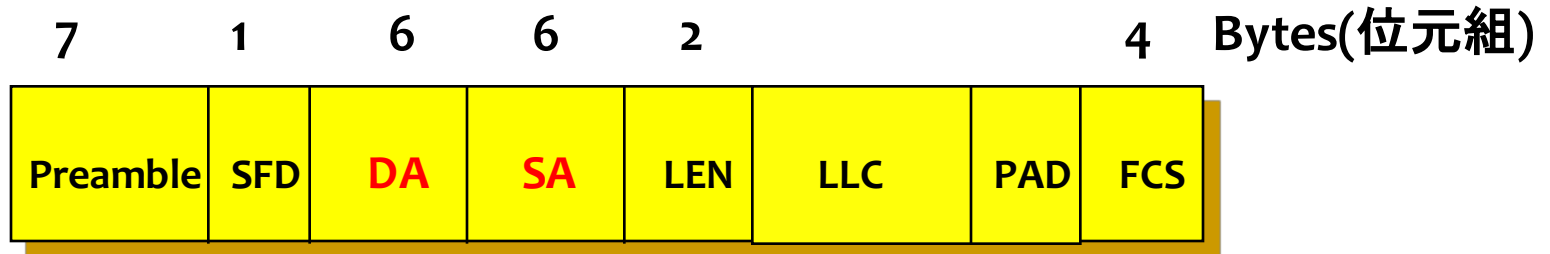
- 背景介紹
- 訊框轉送和位址學習
- 迴圈問題及解析
- 擴張樹演算法
- 擴張樹的維護

一個簡單的橋接網路範例

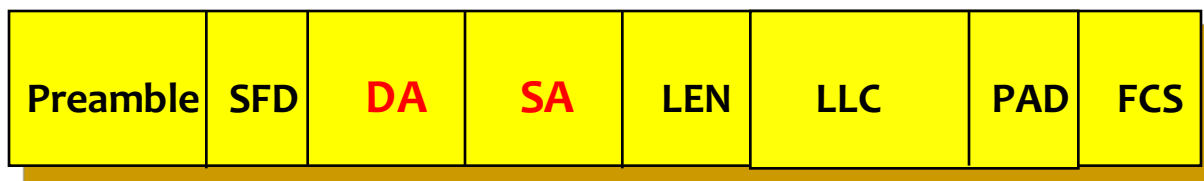
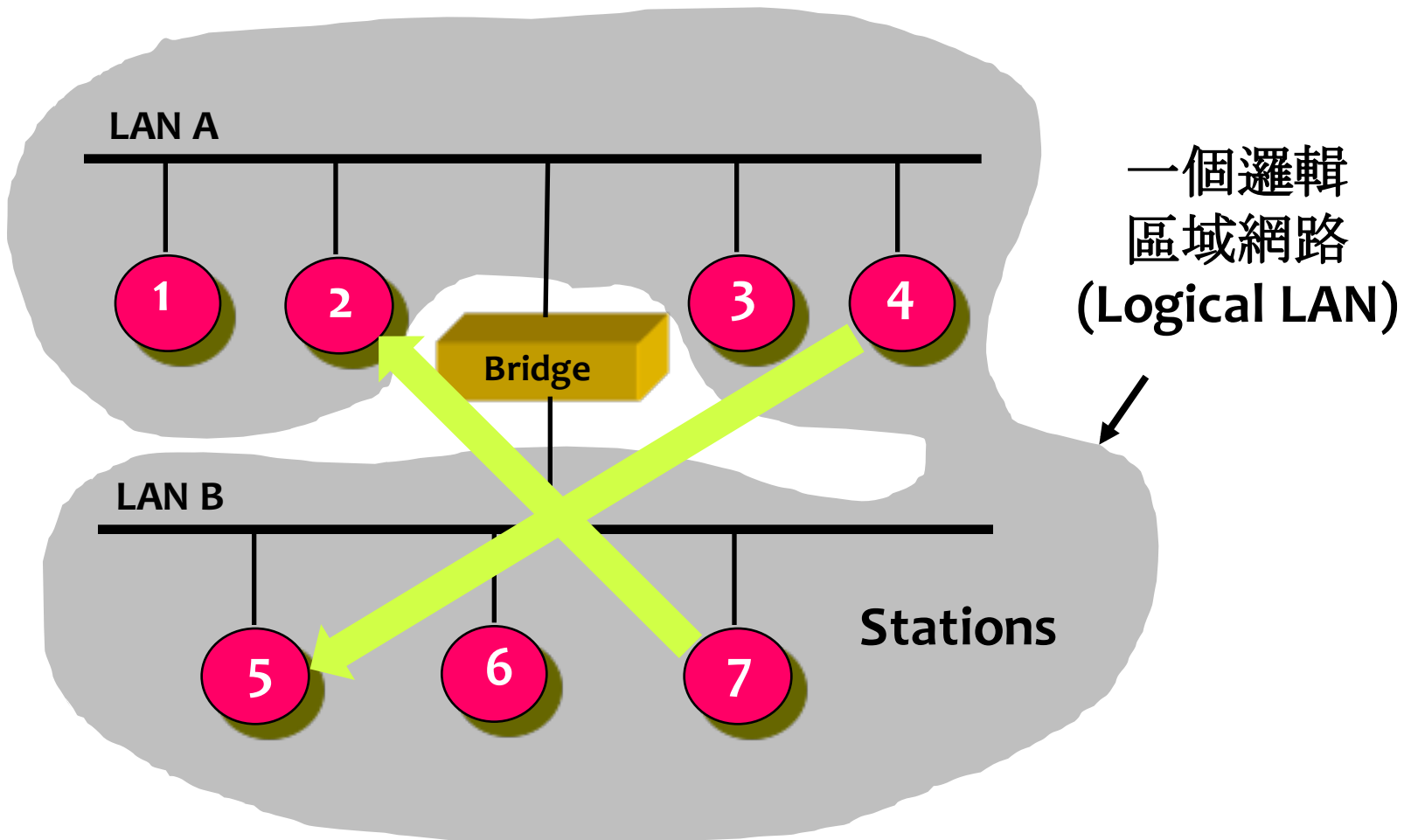


什麼是橋接器？

- 橋接器(bridge)是一個MAC層(第二層)的網路設備，用來連接並協助實體分離區域網路上的訊框轉送，以讓終端主機將這些實際分離的區域網路視為邏輯上同一個區域網路



邏輯區域網路的概念



MAC_3 MAC_7

橋接器的功能

■ 基礎功能:

- 訊框的轉送與過濾
- MAC 位址學習
- 解決在拓樸中可能存在的迴圈

■ 附加的功能:

- 壅塞控制 (足夠的緩衝器)
- 靜態過濾 (安全性)
- 訊框轉譯 (多個橋接器)
- 路徑 (多個橋接器)
- 切割

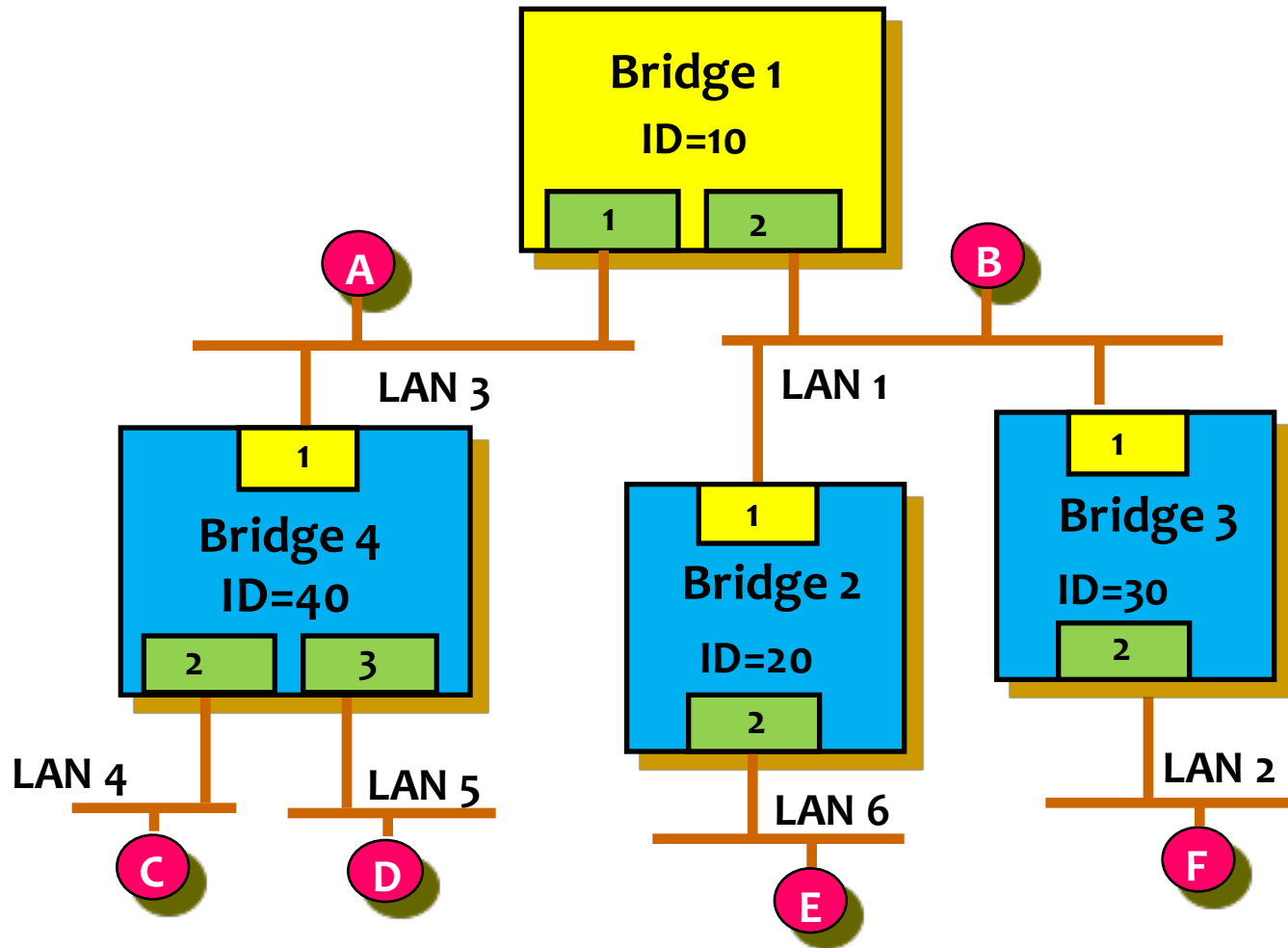
橋接器設計上的考量

- 不對訊框的格式以及內容做修改
- 包含足夠的緩衝區已應付突發的大量需求
- 有位址已及路徑上的判斷能力
- 一個橋接器可能連結多個網路
- 為什麼需要橋接網路(Bridged LAN, BLAN)?
 - 提高可靠性
 - 增加效能
 - 提升安全性
 - 跨地區網路連接(地理因素)

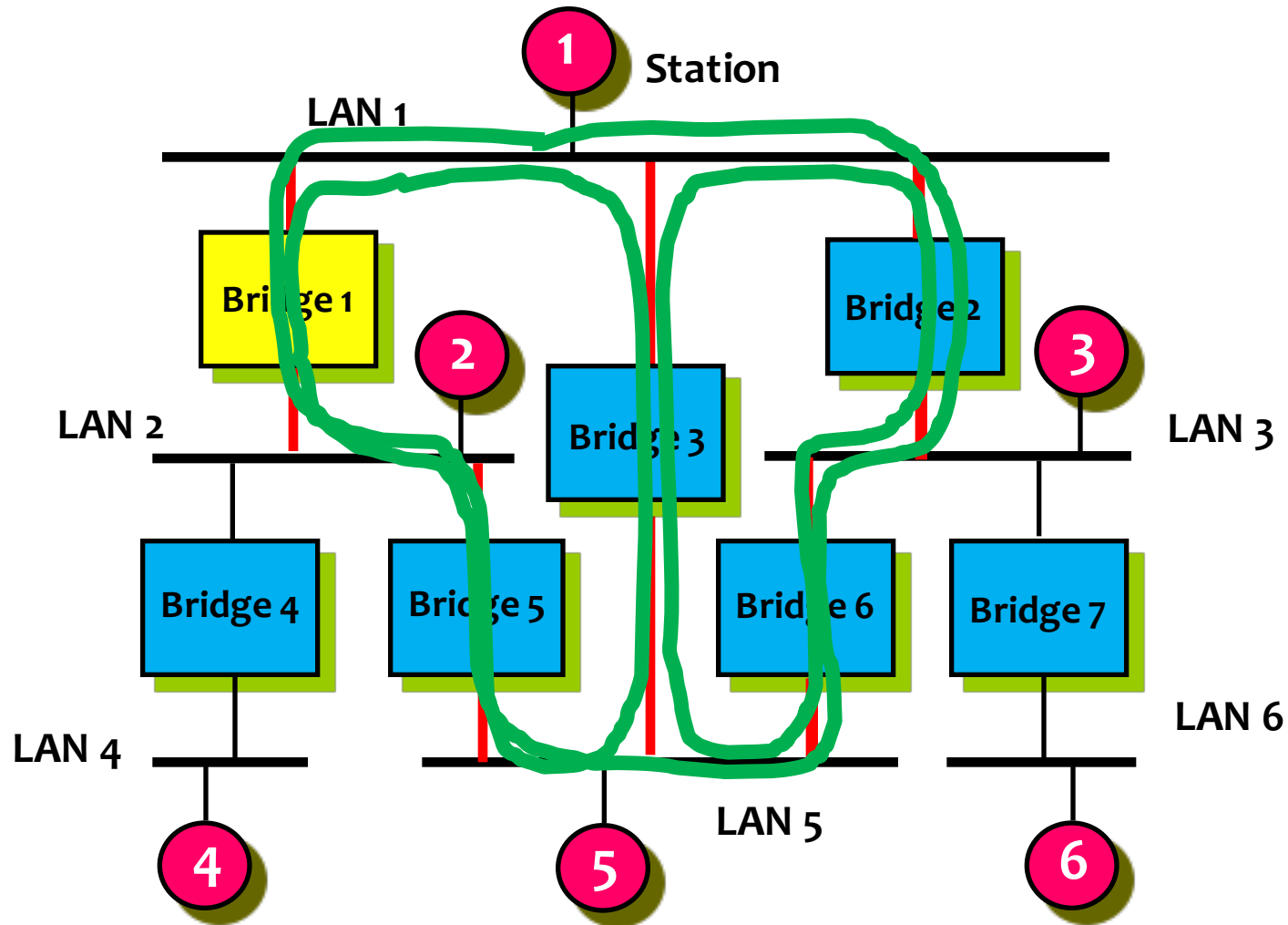
橋接網路的路徑

- 橋接器必須有轉送封包的能力
- 傳送的路徑可能不永遠唯一(迴圈的存在)
- 拓樸的改變必須要被考慮到
- 橋接器知道所有工作站的位址 (**Filtering Database, 過濾資料庫**)

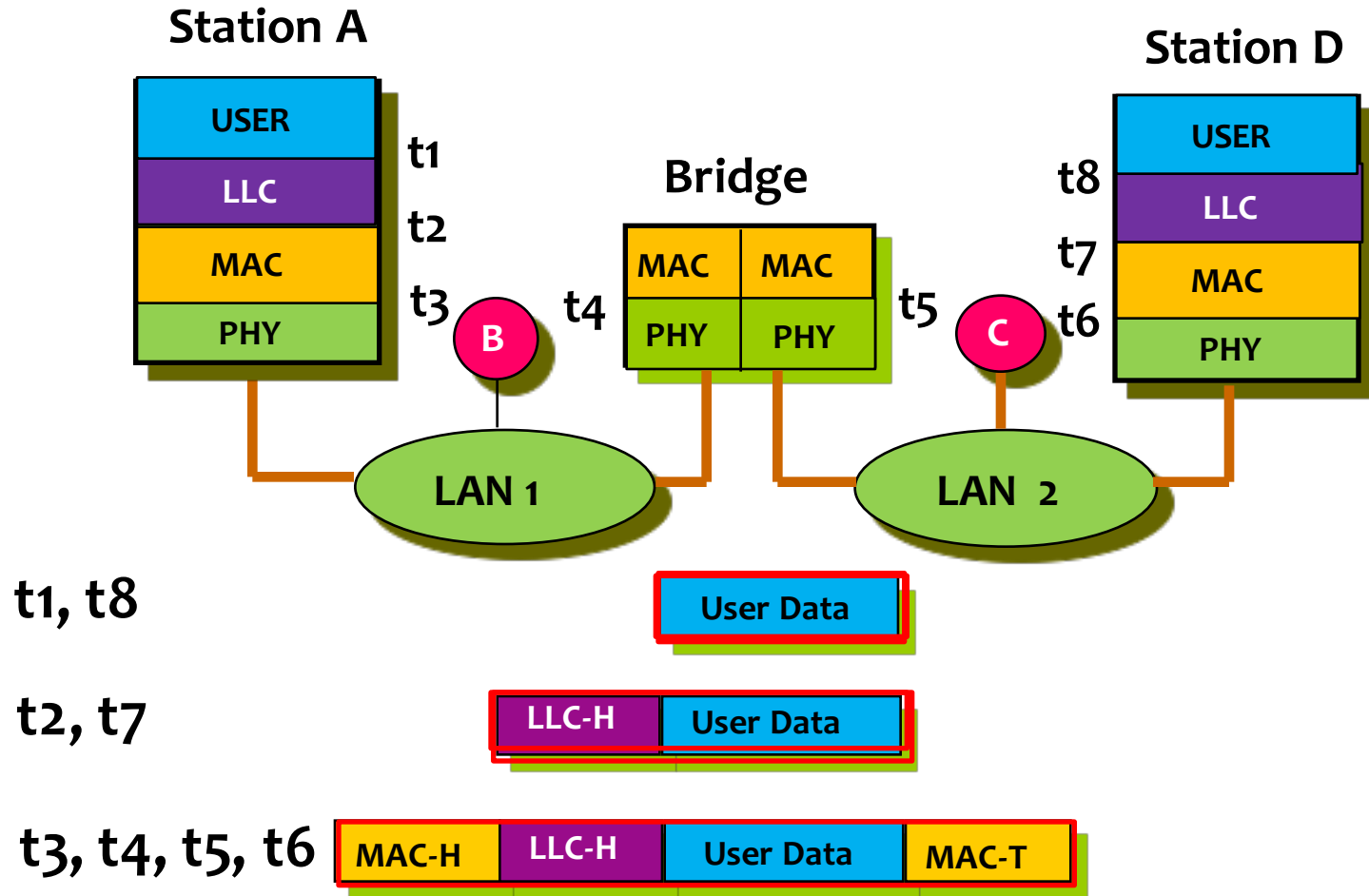
橋接網路範例 (無迴圈)



橋接網路範例 (有迴圈)



橋接器協議的架構



擴張樹上的傳送

■ 訊框的轉送以及過濾

■ 利用訊框中的目的地MAC位址(DMAC)欄位

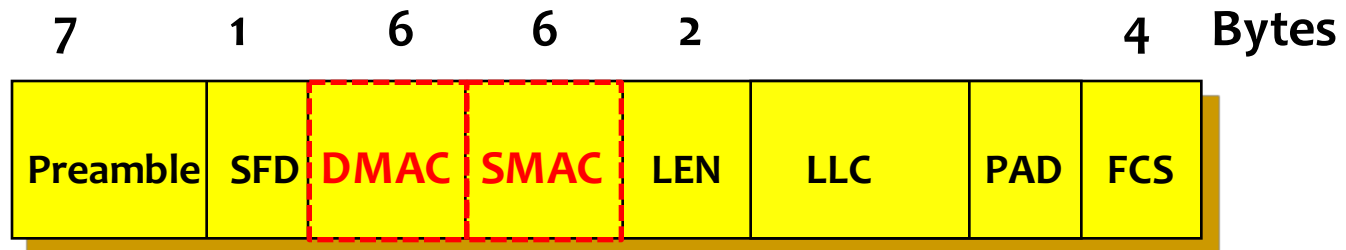
- 橋接器會維護一個過濾資料庫，其中的欄位如下：

[MAC 位址, 隸屬埠(Port), 時間 (Time)]

■ 位址學習

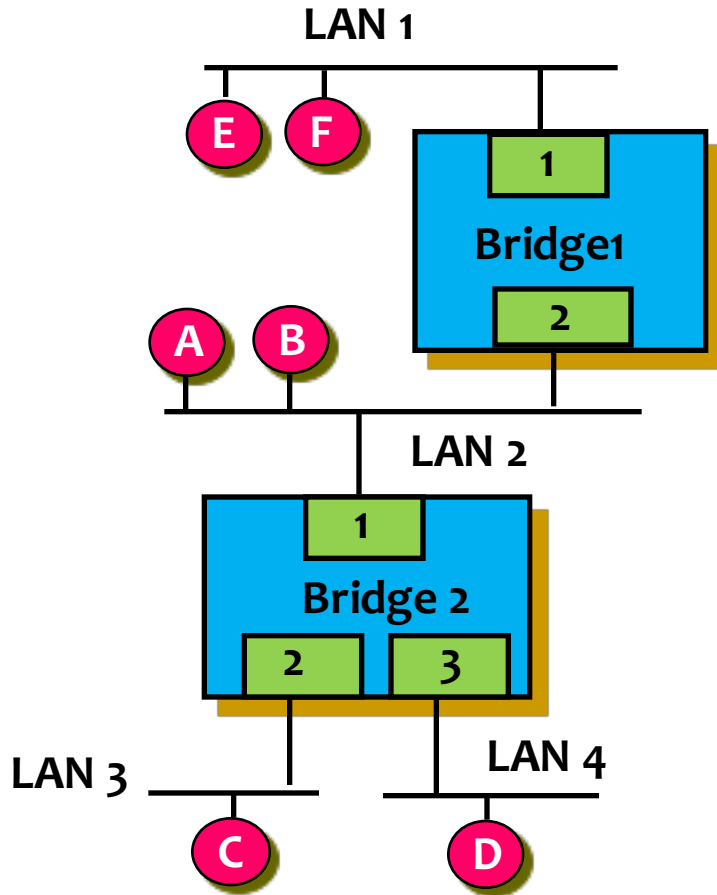
■ 利用訊框中的來源MAC位址(SMAC)欄位

- 如果該記錄已存在過濾資料庫中，則更新對應的欄位值，並重設時間值
- 如果該記錄不在過濾資料庫中，則建立一個新的記錄，且設定其時間值為初始值



過濾資料庫範例

過濾資料庫 (Bridge 1)



MAC Addr	Port	Time (S)
A	2	20
B	2	18
C	2	25
D	2	4
E	1	5
F	1	12

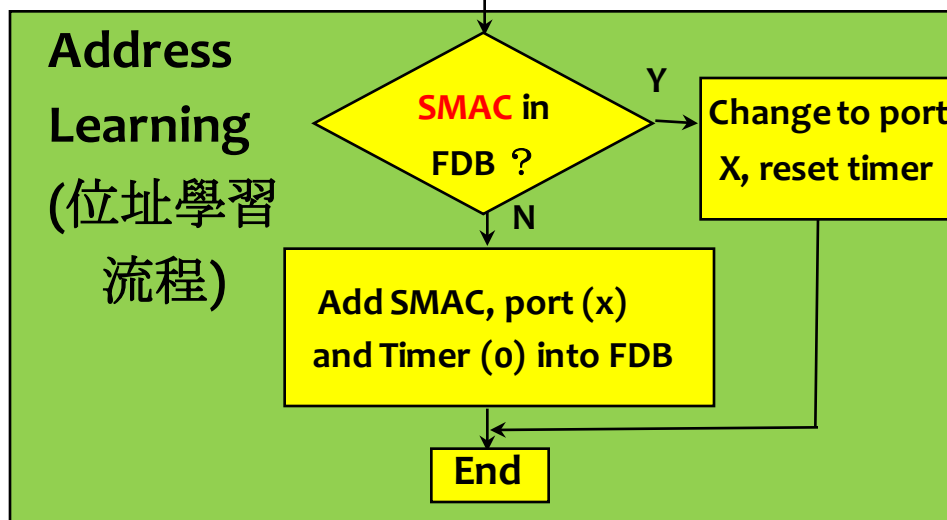
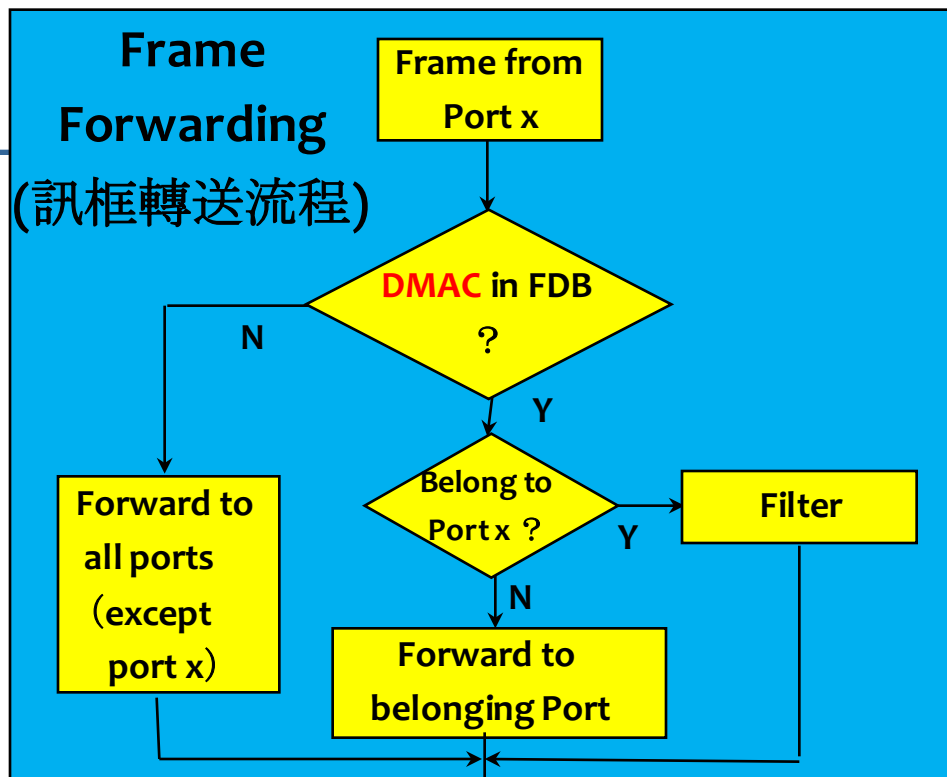
過濾資料庫 (Bridge 2)

MAC Addr	Port	Time(S)
A	1	19
B	1	17
C	2	24
D	3	3
E	1	6
F	1	13

大綱

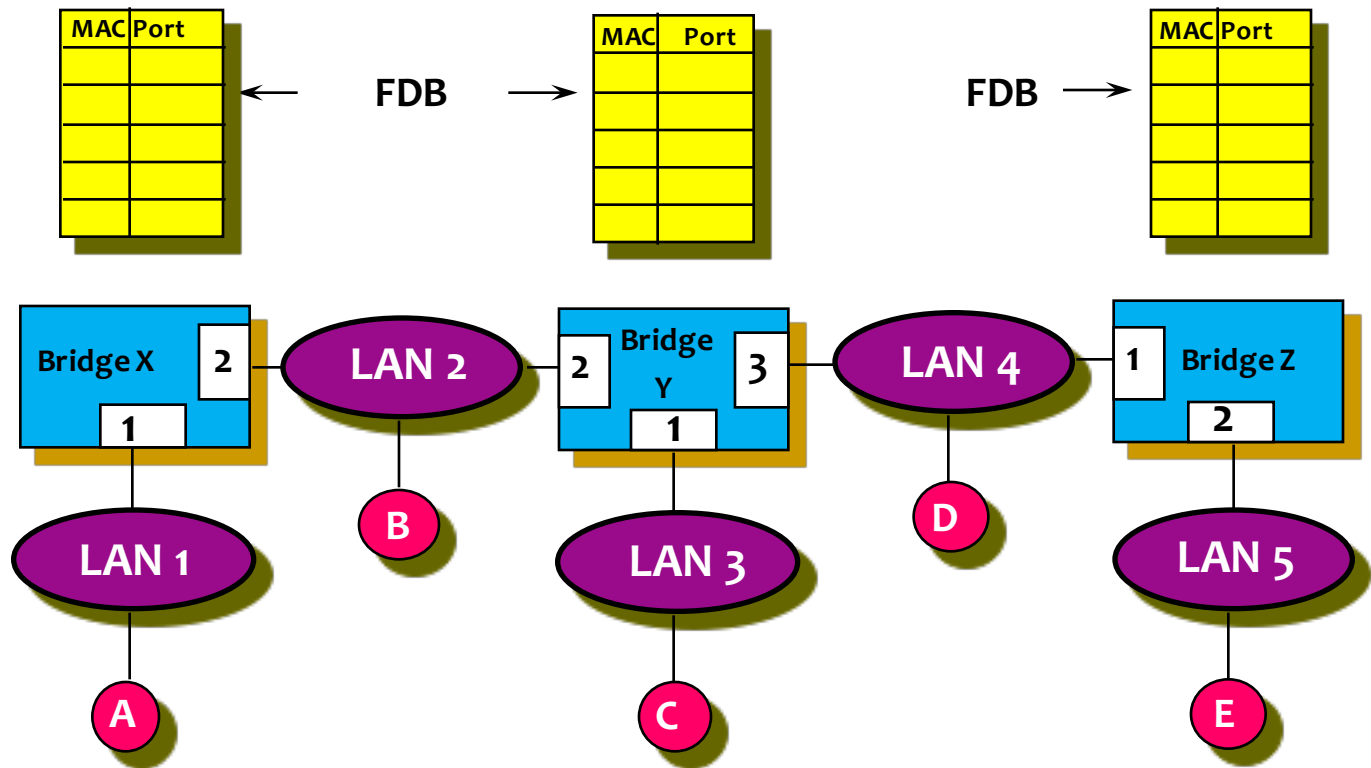
- 背景介紹
- 訊框轉送和位址學習
- 迴圈問題及解析
- 擴張樹演算法
- 擴張樹的維護

訊框轉送 與 位址學習 演算法

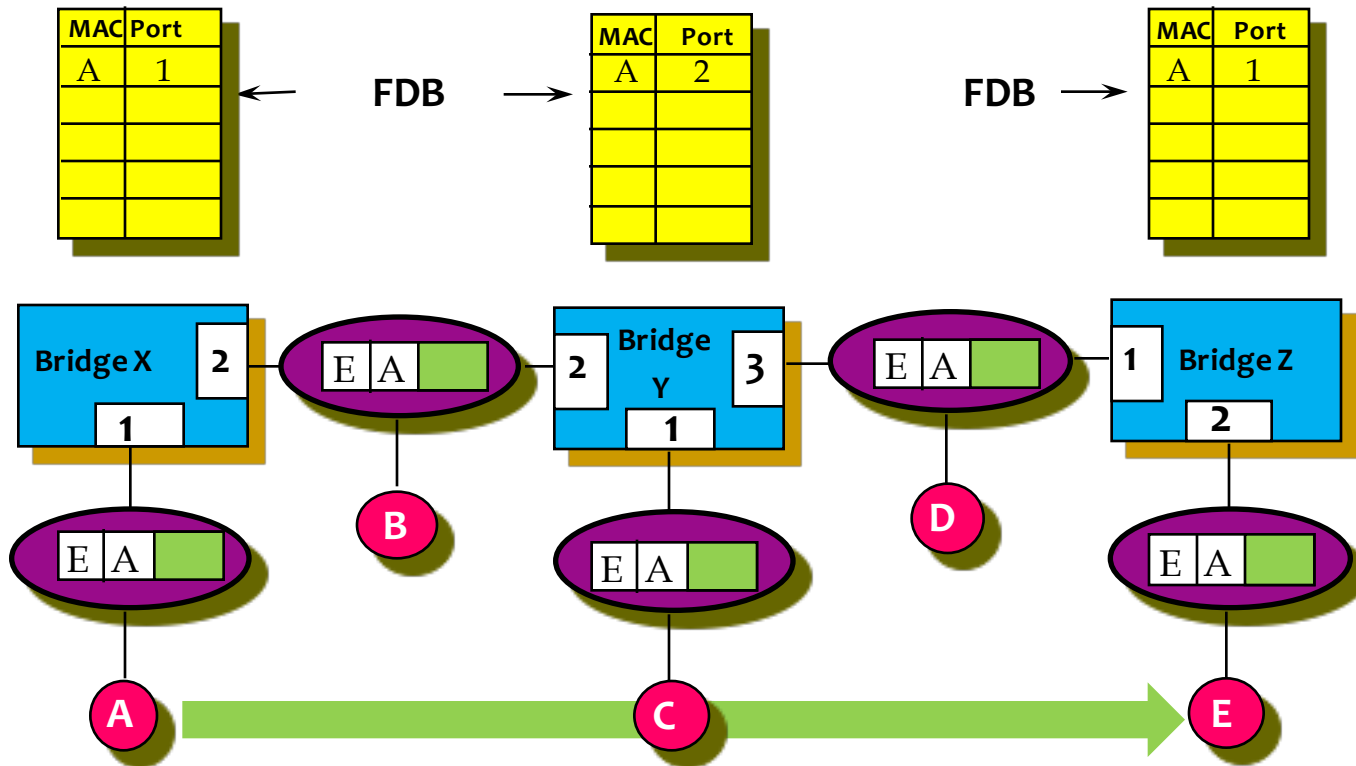


位址學習範例

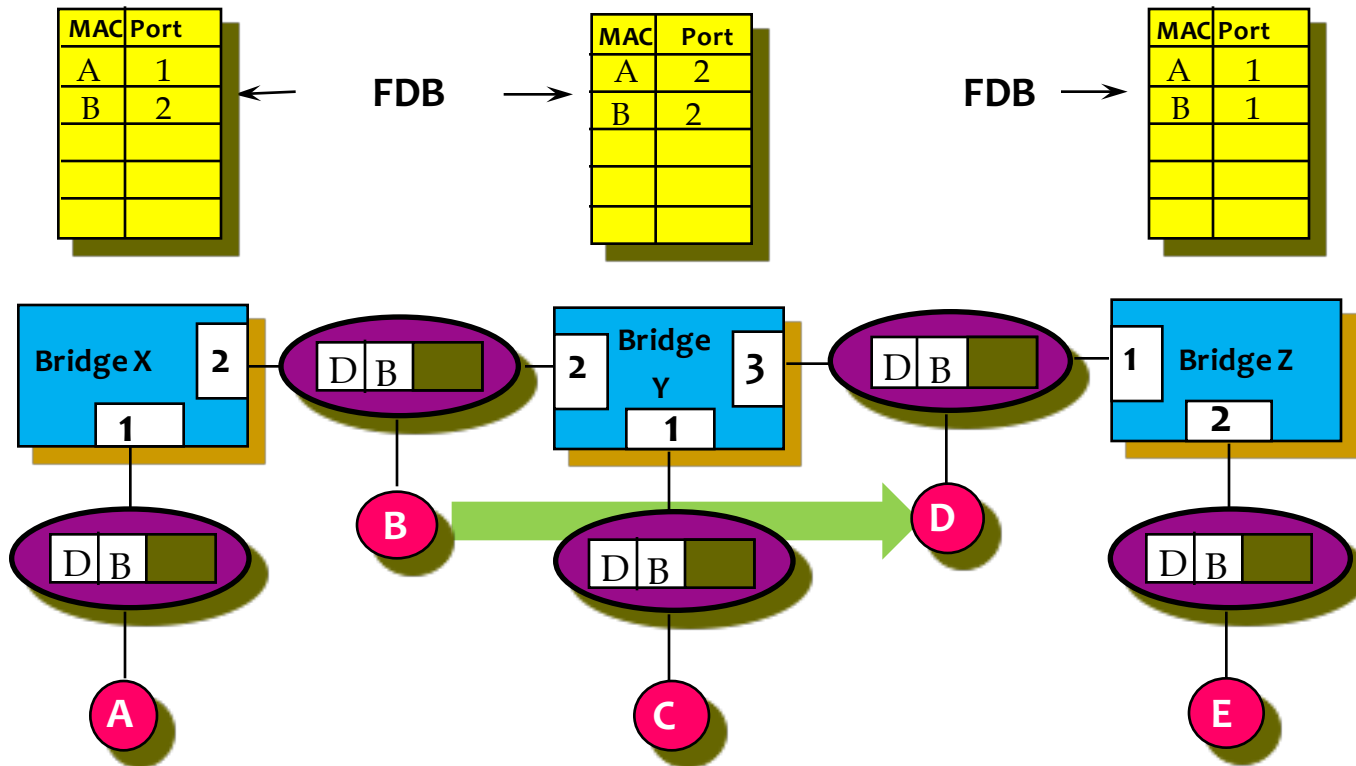
1. A -> E
2. B -> D
3. C -> B
4. D -> A
5. E -> C



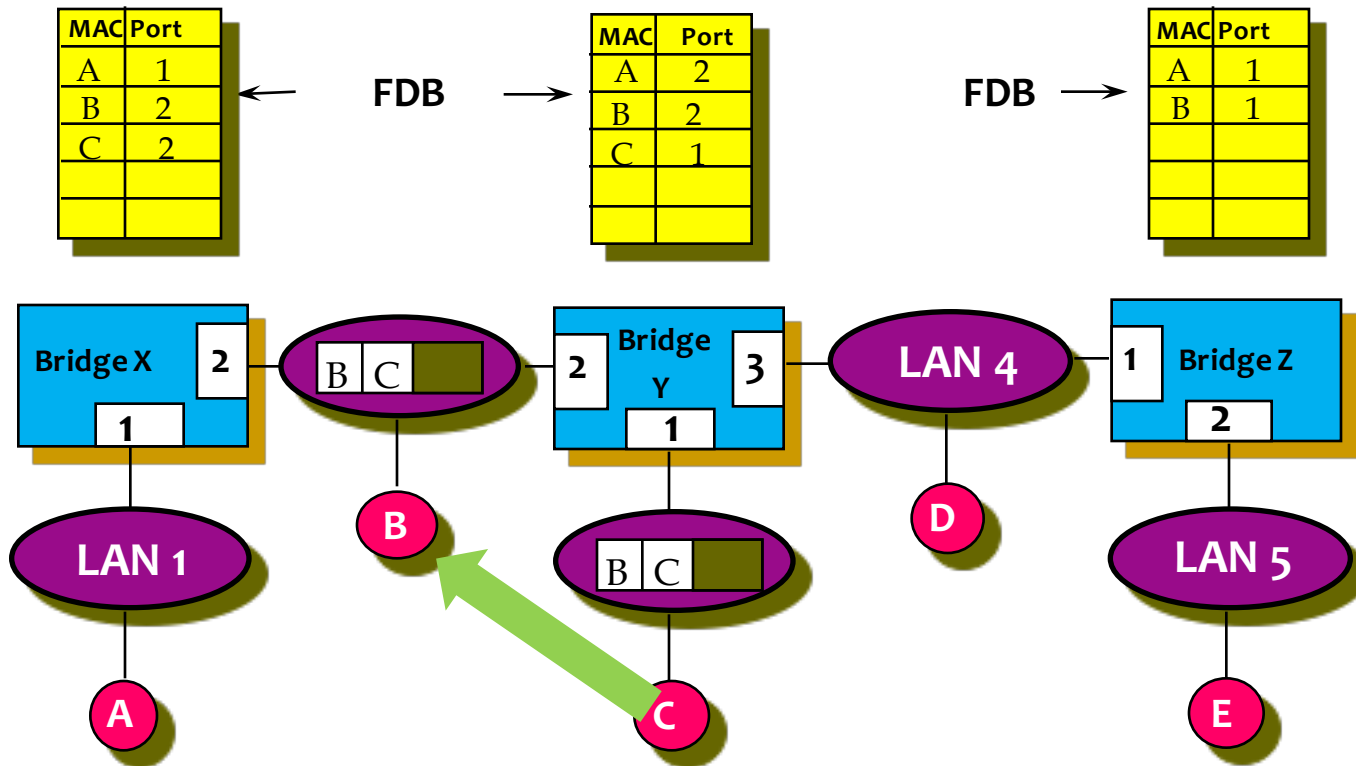
位址學習範例 (A→E)



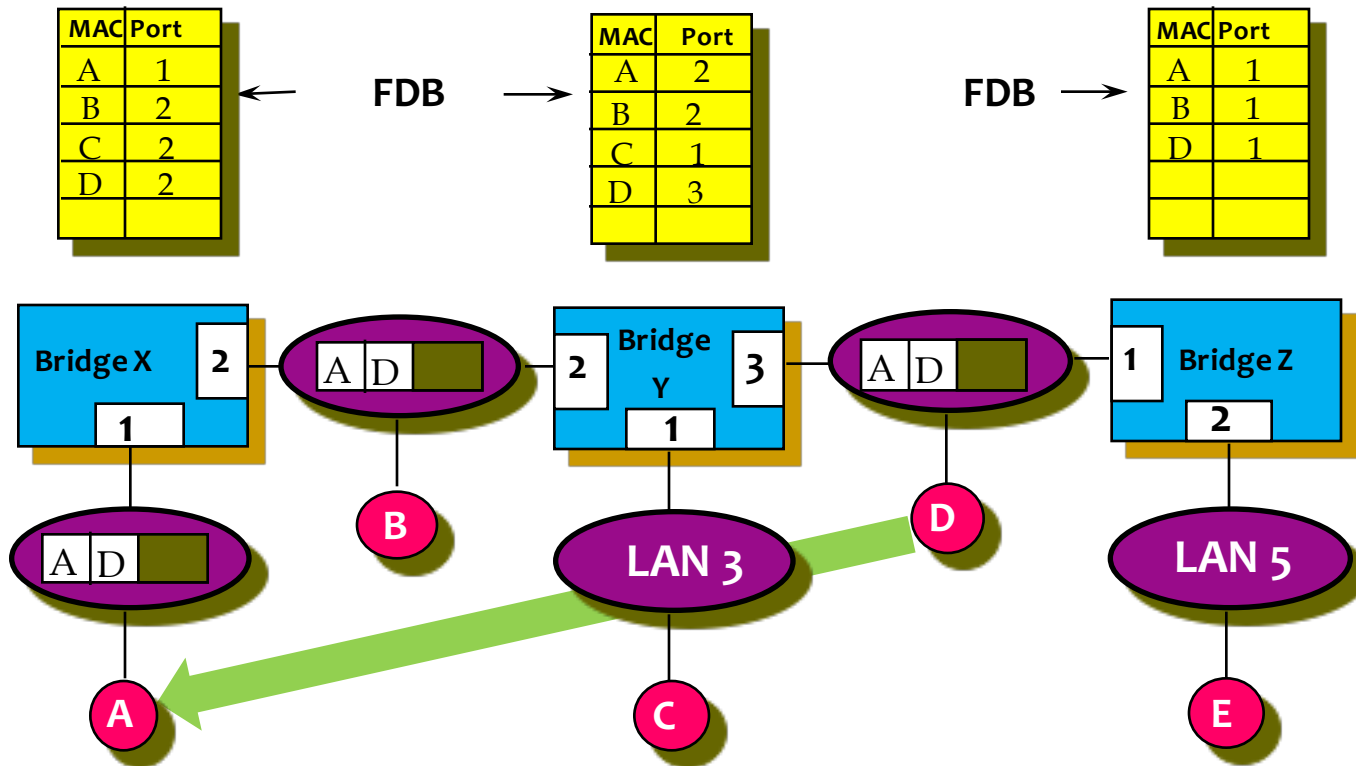
位址學習範例 (B→D)



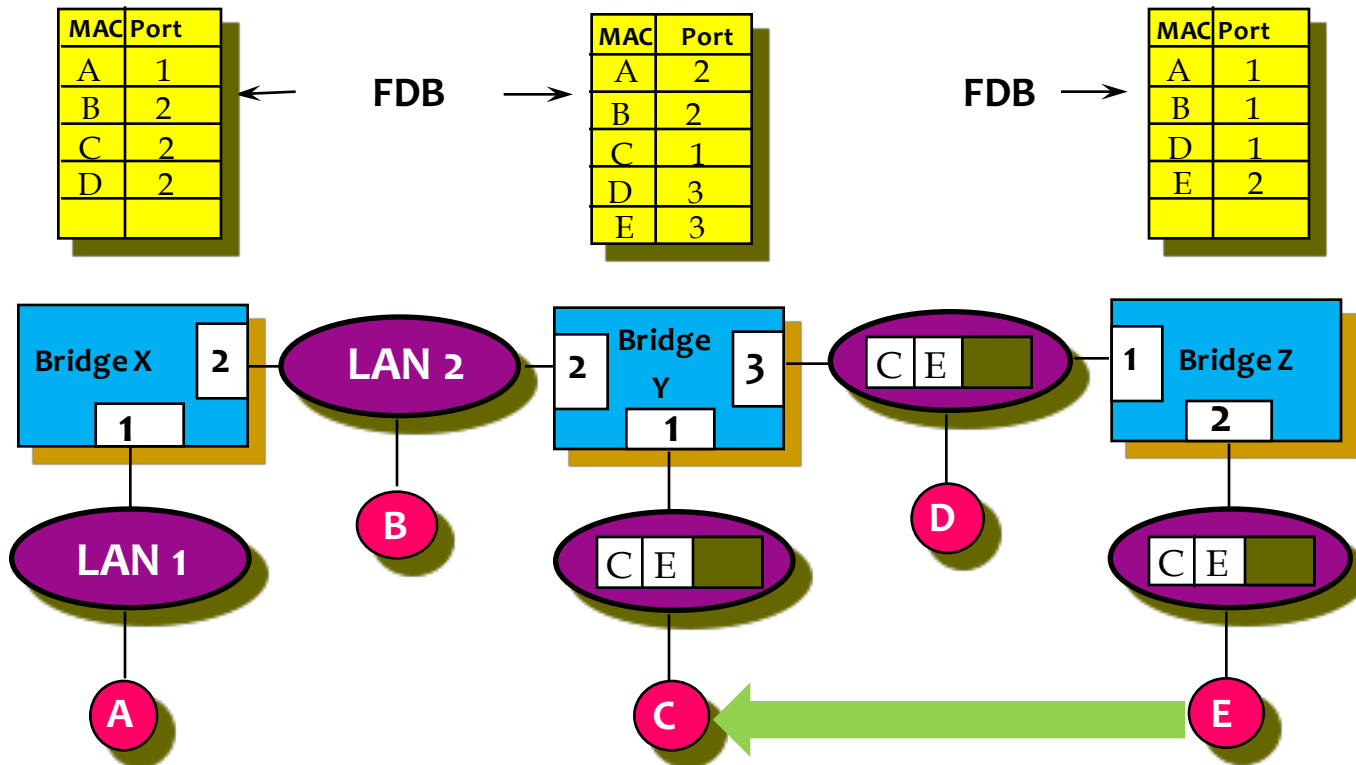
地址學習範例 (C→B)



地址學習範例 (D→A)



位址學習範例 (E→C)

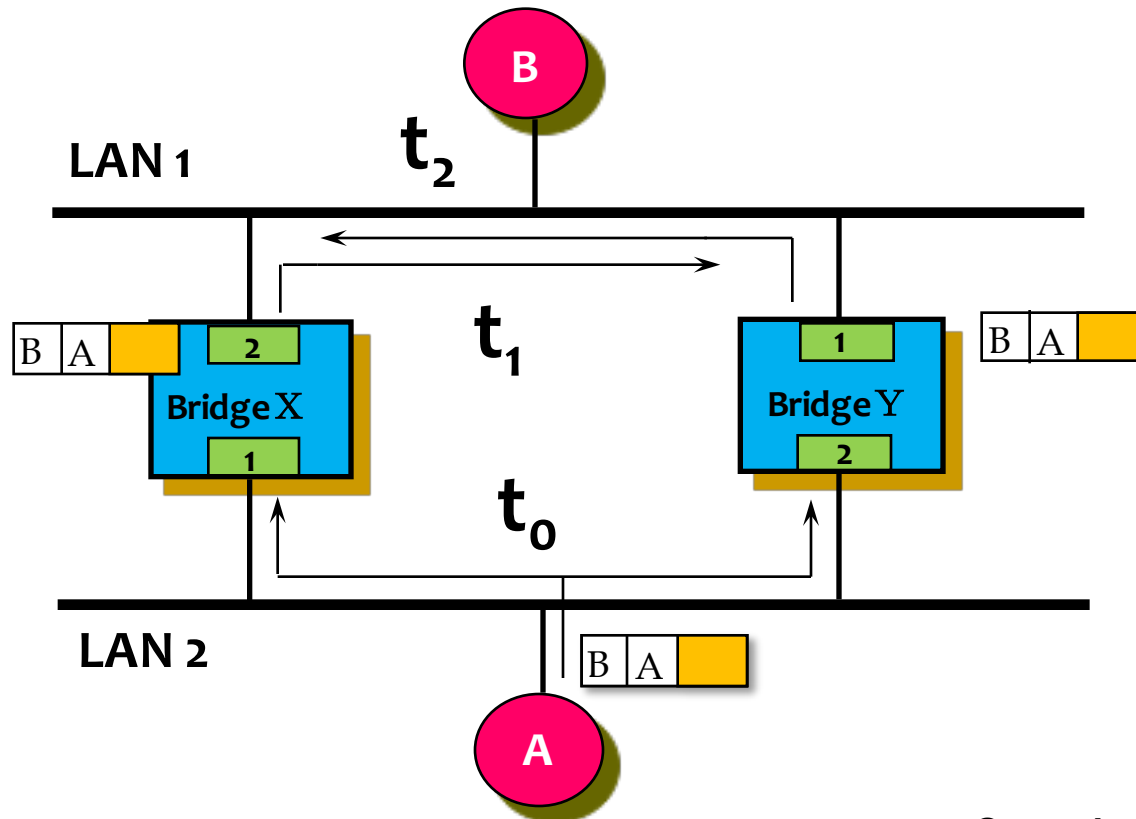


大綱

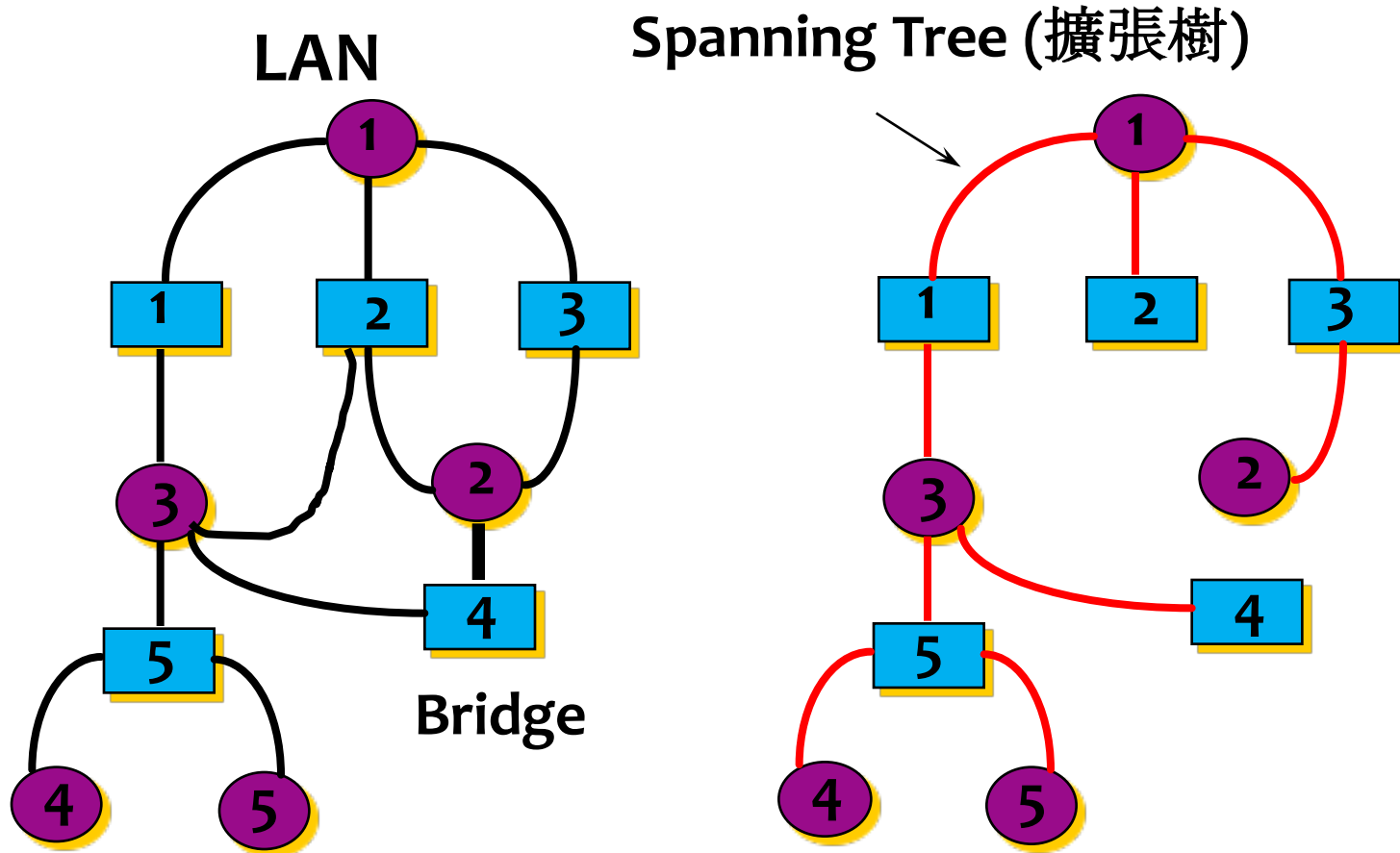
- 背景介紹
- 訊框轉送和位址學習
- 迴圈問題及解析
- 擴張樹演算法
- 擴張樹的維護

迴圈問題及解析

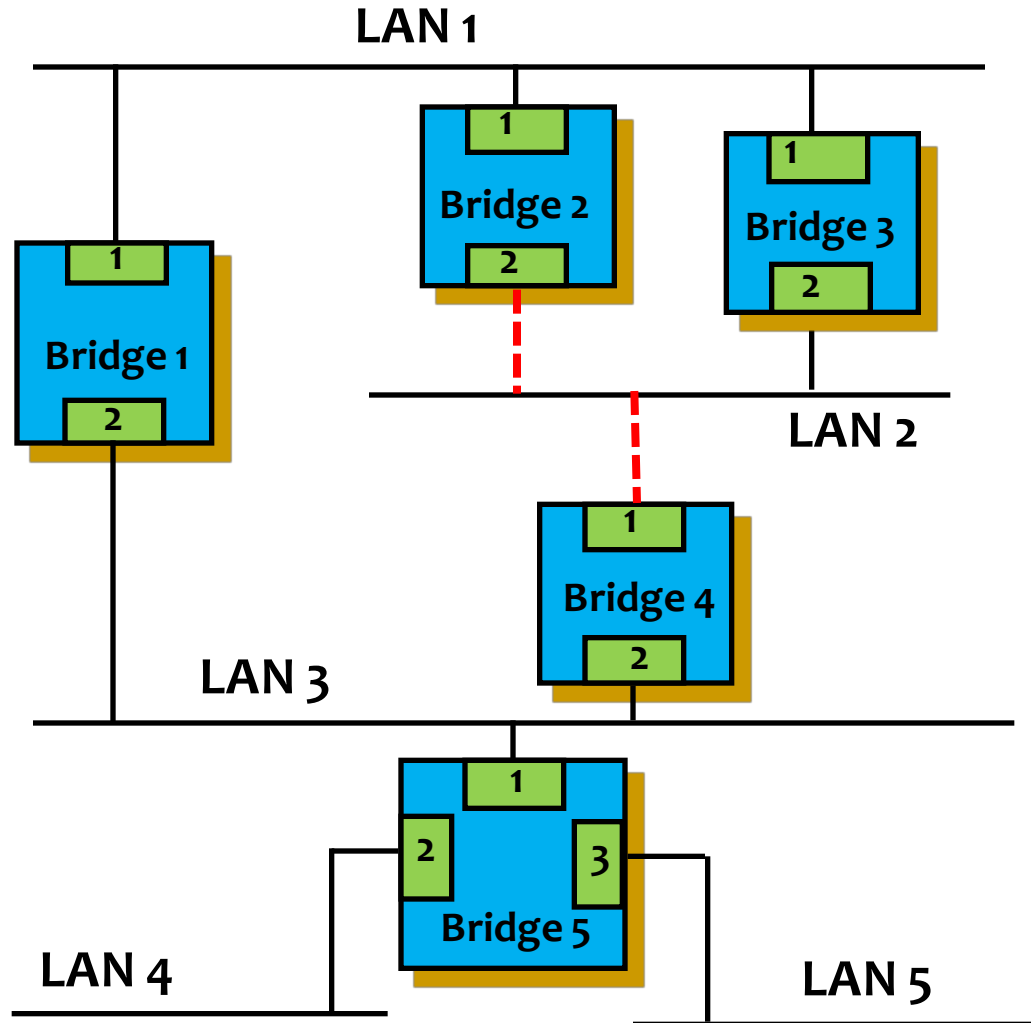
- 迴圈為網路帶來較高的可靠性
- 但是迴圈會造成訊框的重複接收
- 迴圈也會使位址學習發生錯誤



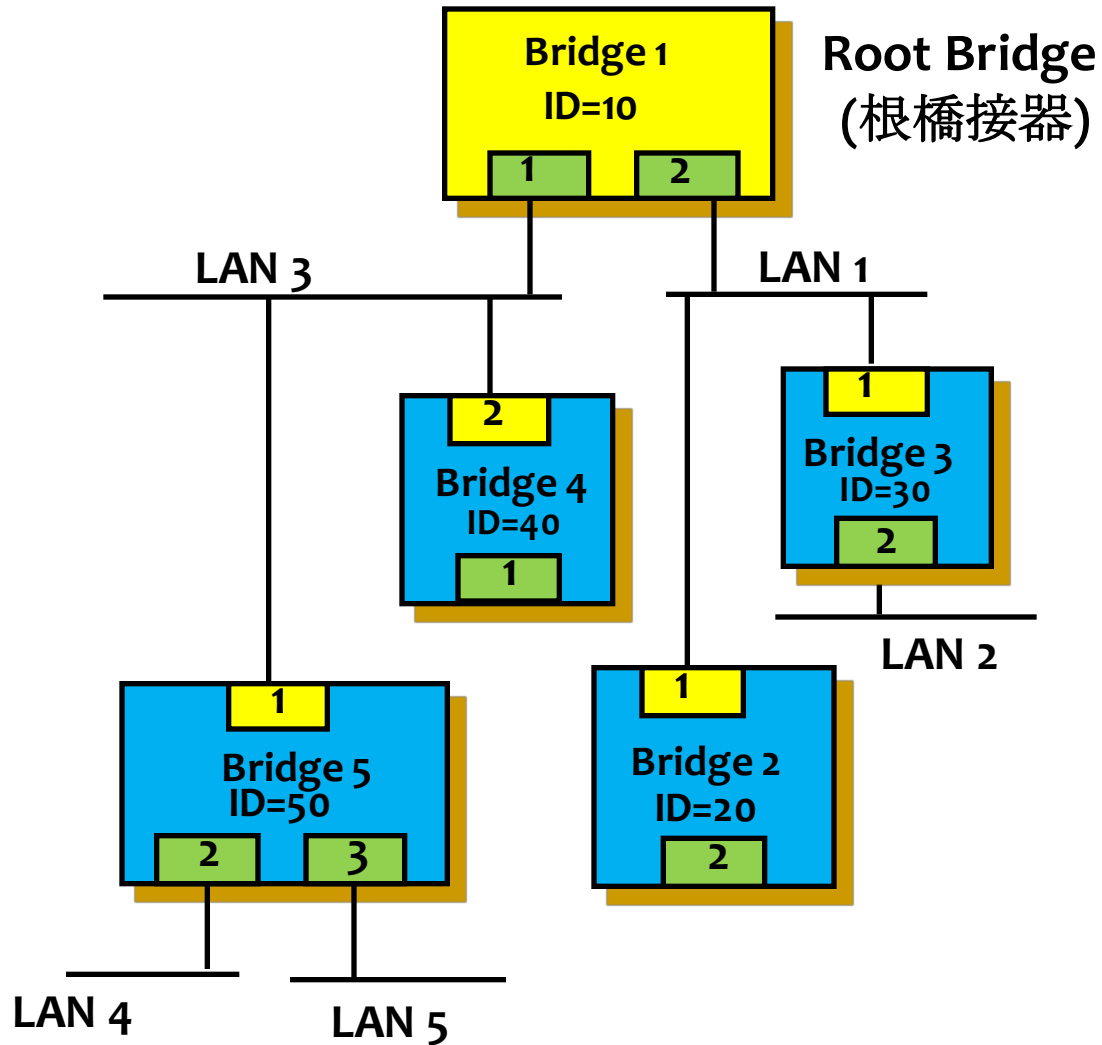
橋接網路的圖形範例



擴張樹範例 1



擴張樹範例 1 (續前頁)



大綱

- 背景介紹
- 訊框轉送和位址學習
- 迴圈問題及解析
- 擴張樹演算法
- 擴張樹的維護

擴張樹演算法(必要條件)

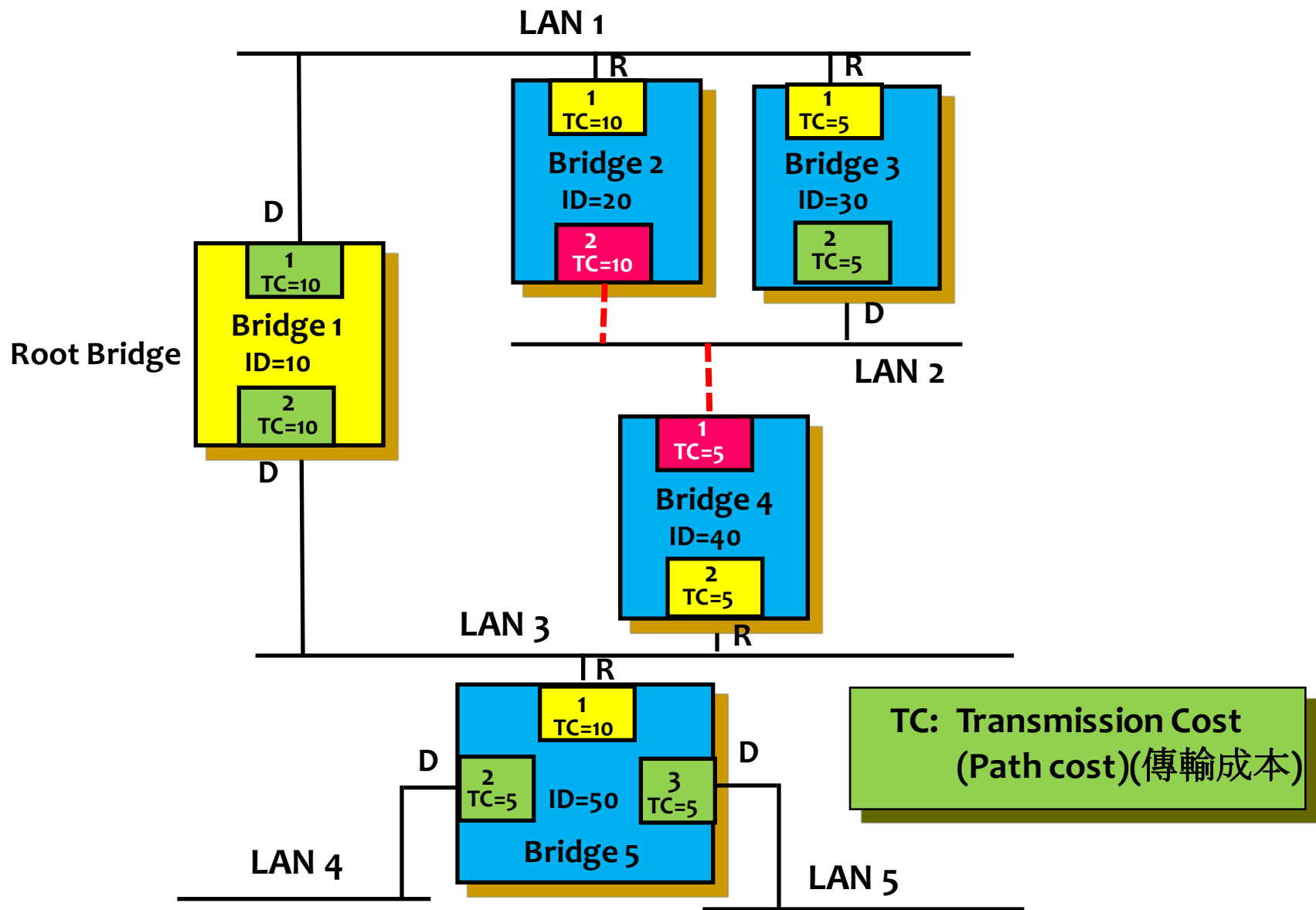
■ 橋接器

- 每個橋接器都會有一個唯一的辨識碼 (Bridge ID, 8 個位元組):
 - ▶ 優先權的部分 (兩個位元組): 可程式化(管理者可設定其值)
 - ▶ 地址的部分 (六個位元組): MAC 位址
- 所有橋接器都屬於一個特別的群組MAC位址:
01-80-C2-00-00-00 (群播位址)
10000000-00000001-01000011-
- 橋接器上每個埠都具唯一的埠辨識碼 (port ID).

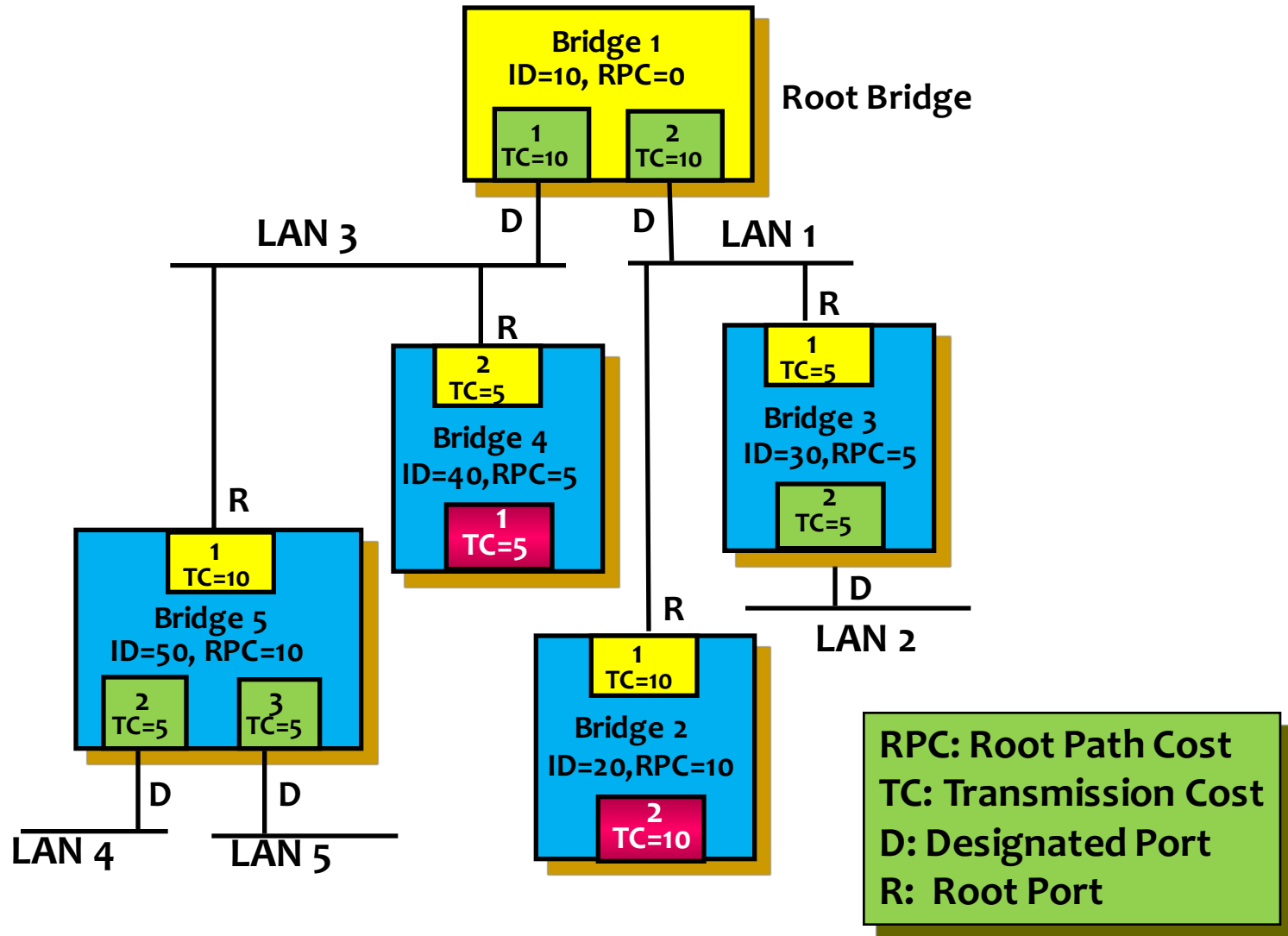
擴張樹演算法 (定義)

- **根橋接器 (Root Bridge):** 橋接器識別碼最小的橋接器。
- **路徑成本 (Path Cost):** 對於每一個埠，將一個訊框經由此埠送進區域網路的成本。
- **根埠 (Root Port):** 對於每個橋接器，經由此埠到根橋接器有最低路徑成本
- **根路徑成本 (Root Path Cost, RPC):** 對於每個橋接器，經由根埠到根橋接器的路徑成本
- **代理橋接器 (Designated Bridge):** 區域網路經由此橋接器到根橋接器的路徑成本最小，所有進入到此區域網路的訊框以及由此區域網路送出來的訊框都需要經過其代理橋接器。
- **代理埠 (Designated Port):** 區域網路與其代理橋接器相連結的埠，所有要進出該區域網路的流量都須經由此代理埠。

擴張樹範例 2



擴張樹範例 2 (接續前頁)

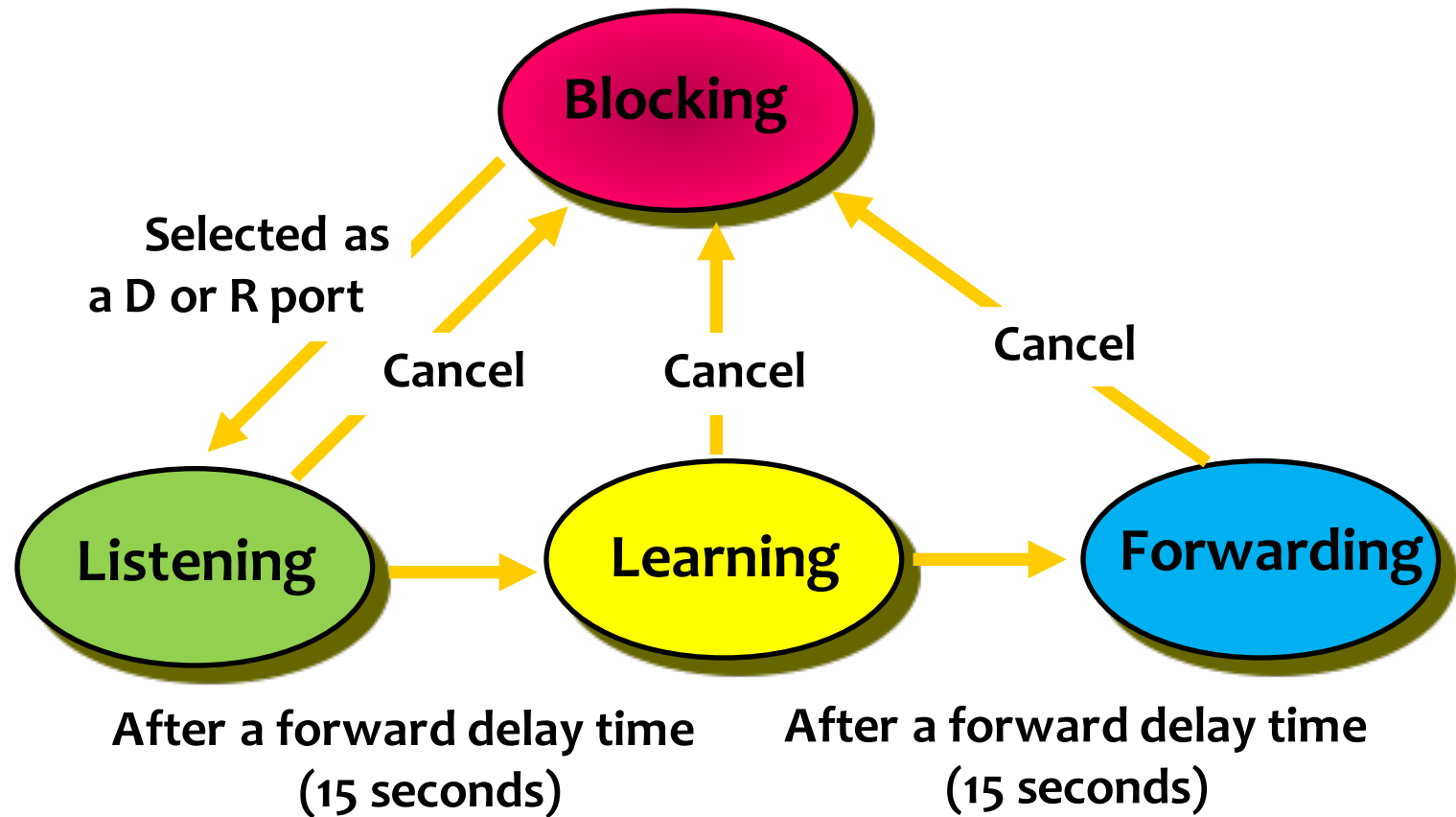


擴張樹演算法

■ 三個步驟:

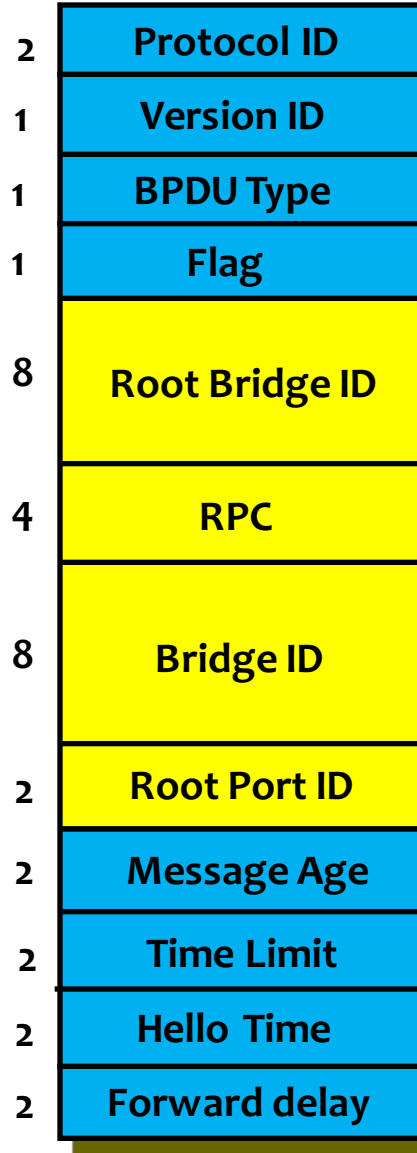
1. 決定根橋接器.
2. 決定所有其他橋接器的根埠.
3. 決定所有區域網路的代理埠.
 - 經由該埠到根橋接器需有**最小的根路徑成本**.
 - 當一個區域網路有多個連接橋接器有相同且最小的根路徑成本時, 選擇具最高優先權(辨識碼最小的)橋接器作為其**代理橋接器**.
 - 如果區域網路的代理橋接器上同時有多個埠連結到此區域網路, 則選擇具最小辨識碼的埠作為**代理埠**.

橋接器埠的狀態圖



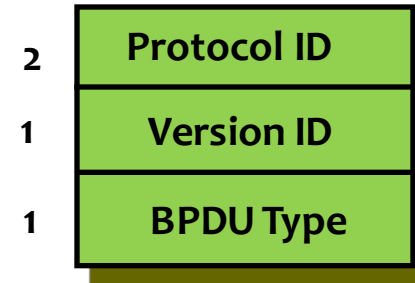
橋接器協議資料單位 (BPDU)

Bytes



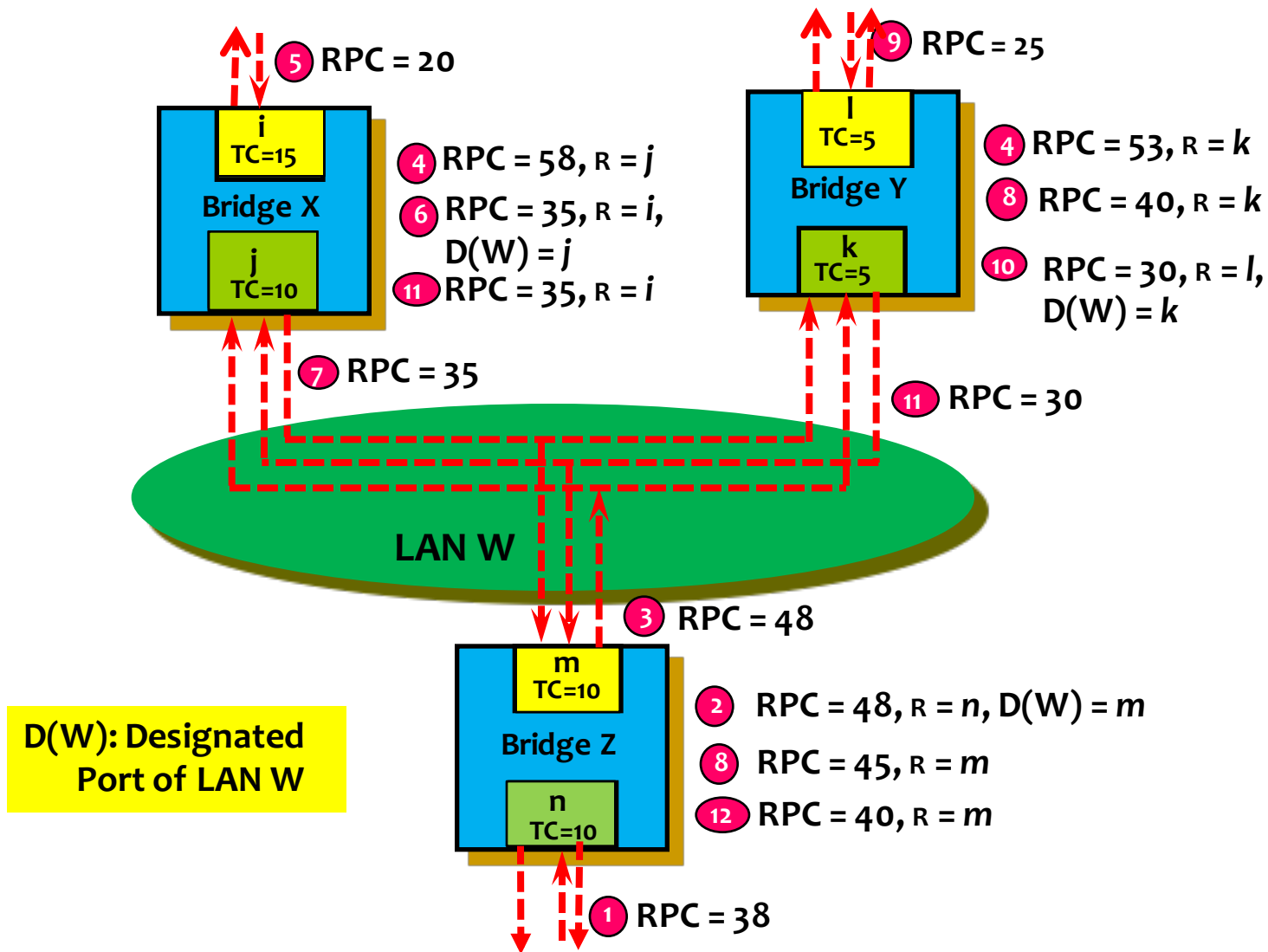
(a) 網路架構 BPDU

Bytes

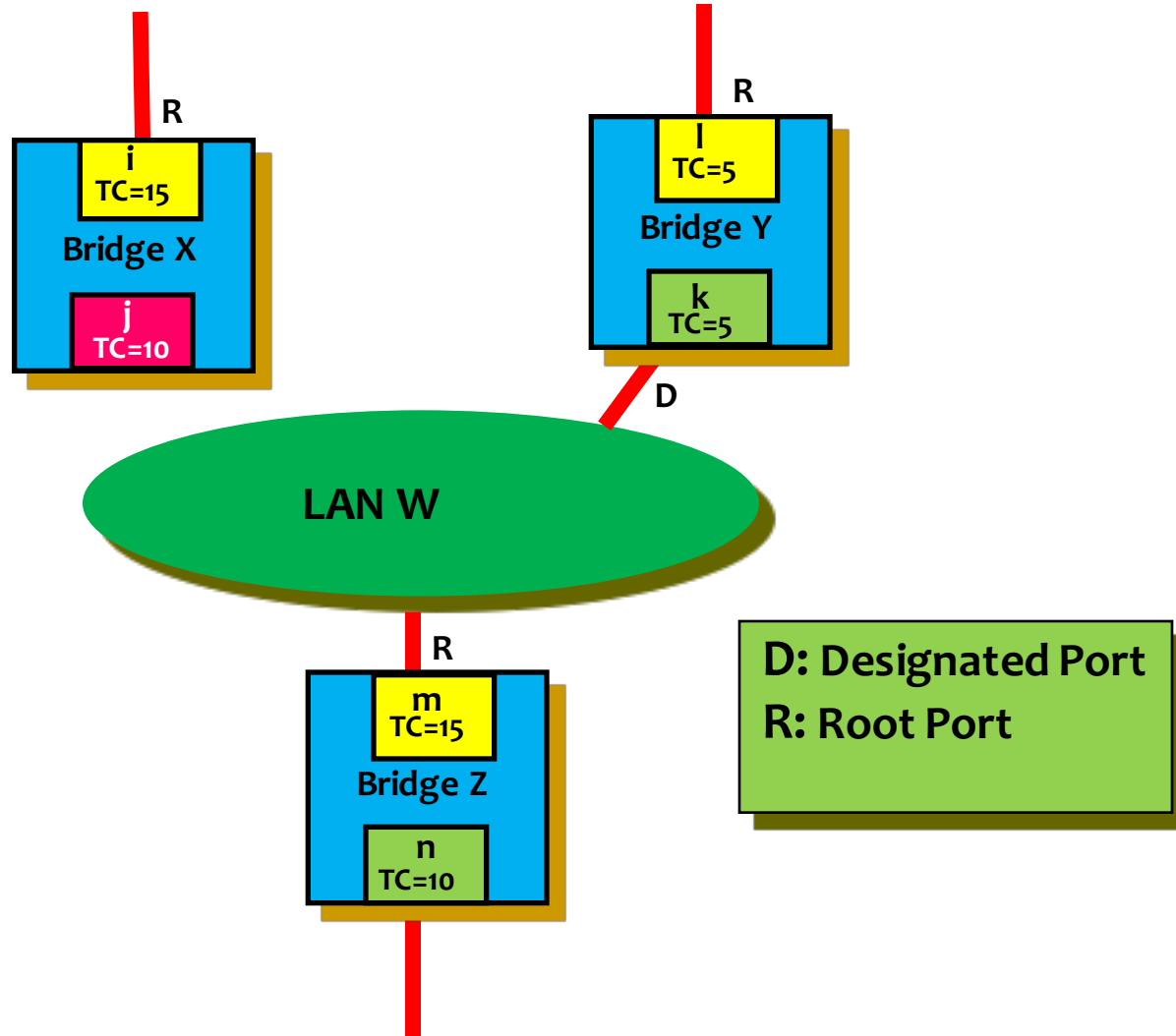


(b) 拓樸變更 BPDU

擴張樹演算法範例 (過程)



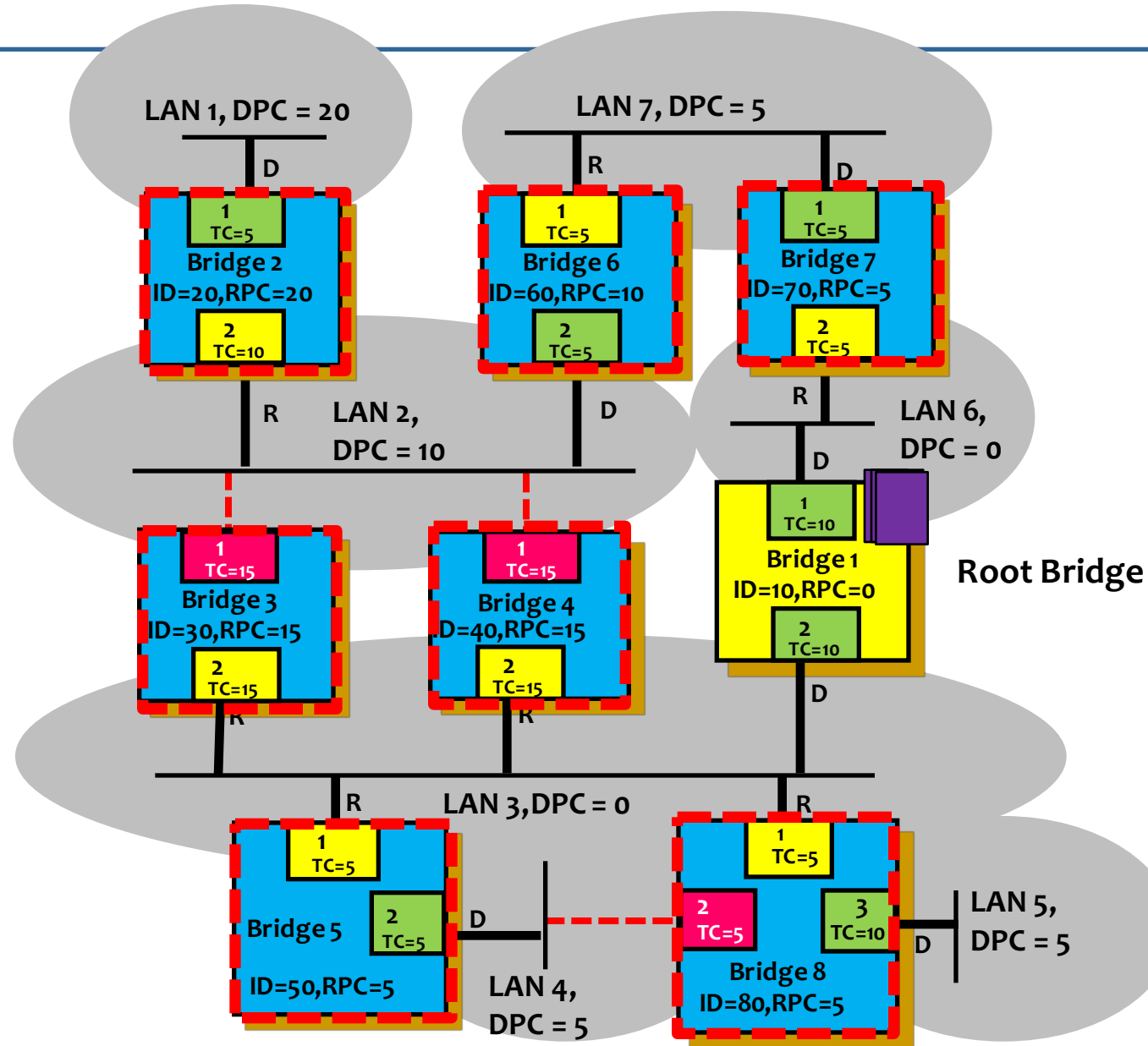
擴張樹演算法範例 (結果)



擴張樹的特徵

- IEEE 802.1D 演算法所建造的擴張樹中，**每個橋接器到達根橋接器的最短路徑都會包括在內。**
- **每個區域網路經由其代理橋接器到根橋接器的路徑為最短(最少根路徑成本, minimum root path cost, RPC).**
- 所以擴張樹通常不是一般所謂的“最低成本擴張樹”(minimum cost spanning tree).
- 一個橋接網路(由橋接器串連的網路)的擴張樹是可預測的或是可以計算出來的。
- 如果有橋接網路的拓樸(可包含迴圈)與一些可設定的參數，此橋接網路的擴張樹是可以手動計算出來的。

擴張樹是可預測的



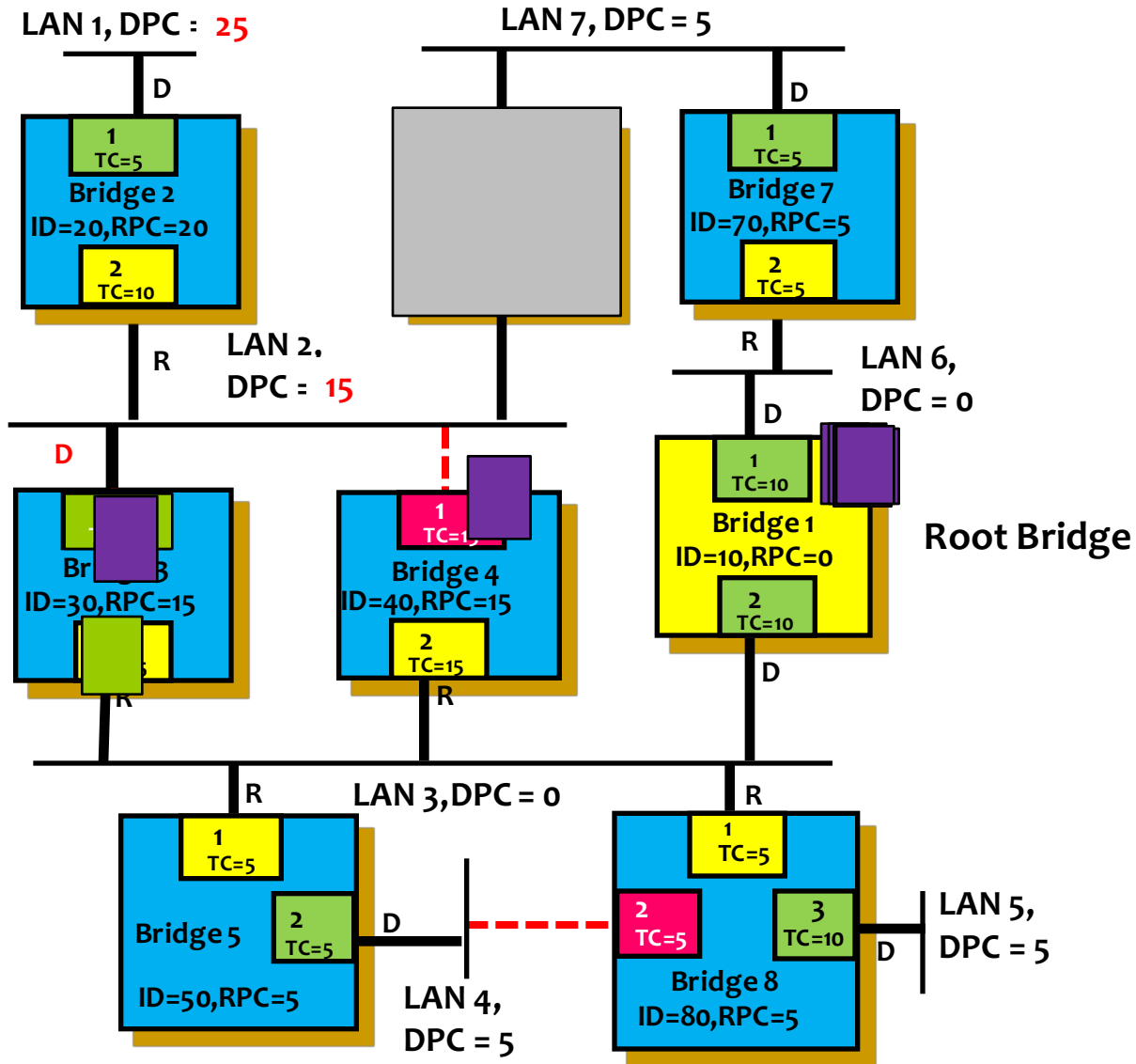
大綱

- 背景介紹
- 訊框轉送和位址學習
- 迴圈問題及解析
- 擴張樹演算法
- 擴張樹的維護

擴張樹的維護

- Configuration (網路架構) BPDU 的傳輸是由根橋接器發動。
- 根橋接器會每隔一段時間(一個Hello time(問候時間))發送一個 **configuration BPDU** 給所有與他有連結的區域網路
- 當一個橋接器由他的根埠收到configuration BPDU的時候, 他會將這個訊息傳遞給所有他認為自己在所連結的區域網路中扮演著代理橋接器的區域網路
- Configuration BPDU將會散布在整個擴張樹中。
- 橋接器可能會改變擴張樹的拓撲
- **TCN BPDU (架構變更改用BPDU)**會以較可靠的方式轉傳關於新的擴張樹的資訊給根橋接器 (由橋接器往根橋接器方向轉傳)。
- 根橋接器會設定 configuration BPDU 中的 **Topology Change flag(拓撲更改旗標)** 告知所有橋接器, 網路拓撲有異動, 過濾資料庫中的內容可能需要重新學習。

擴張樹的維護範例₁ (橋接器故障)



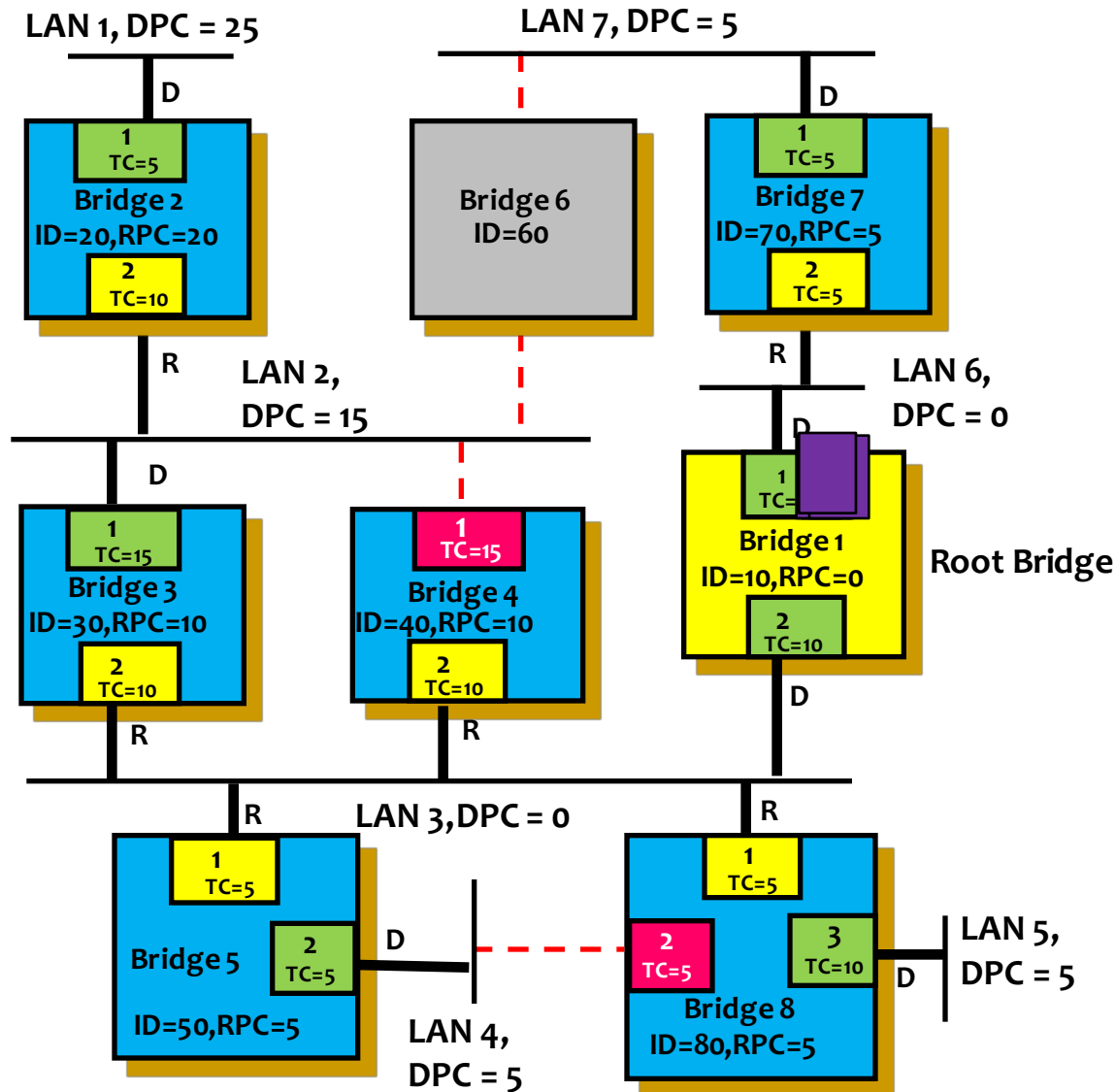
擴張樹的維護範例₁ (橋接器故障)

- 假設橋接器 6 故障了 (識別碼 = 60).
- 所有由根橋接器送往橋接器 6 的Hello BPDU將不會送到區域網路₂.
- 橋接器 3 和橋接器 4 都會發生逾時 (timeout) 的事件, 這也意味區域網路₂ 的代理橋接器(橋接器6)已故障了.
- 橋接器 3 和橋接器 4 都會想成為區域網路₂的代理橋接器, 因此會轉送 configuration BPDU.
- 假設橋接器4首先發送了一個 RPC = 15 的 BPDU.

擴張樹的維護範例₁ (橋接器故障)

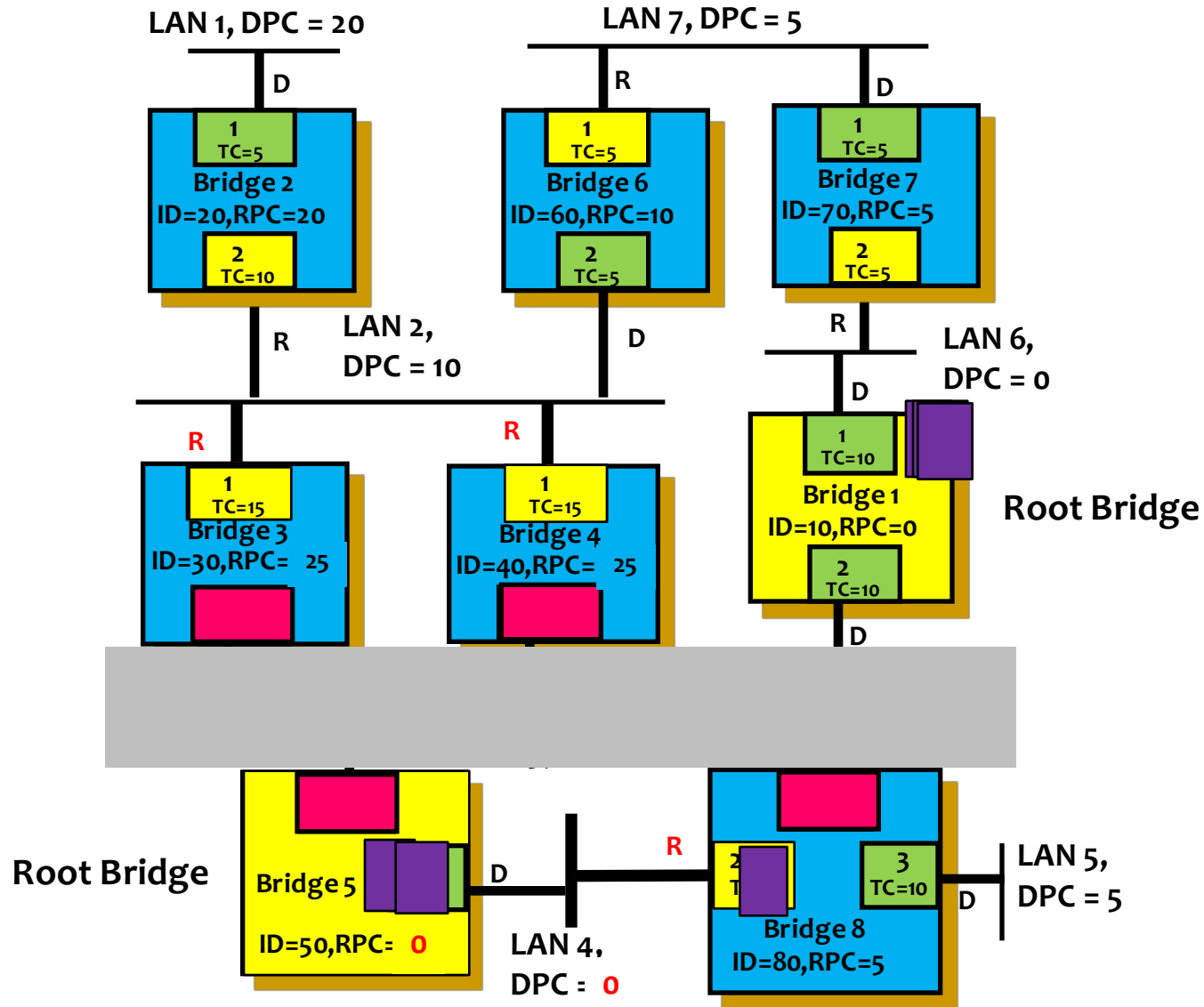
- 因為橋接器₃的優先順位高於橋接器₄ (同樣的RPC, 同樣的識別碼), 所以橋接器₃認為自己更合適擔任代理橋接器的角色, 於是回傳另一個 RPC=15 的 BPDUs.
- 經過了兩個前送延遲時間 (forward delay) 後, 橋接器₃將會成為區域網路₂的代理橋接器, 而且其 DPC 將會為 15.
- 同時之間區域網路₁的DPC將由15變成25.
- 橋接器₃將回送出一個**拓樸變更通知 (TCN)** BPDUs給根橋接器.
- 根橋接器在一段時間後會將送出的 configuration 訊息上的**拓樸變更旗標(flag)**設為 1.

範例1的最終擴張樹架構



擴張樹的維護

範例二 (區域網路故障)



擴張樹的維護

範例二 (區域網路故障)

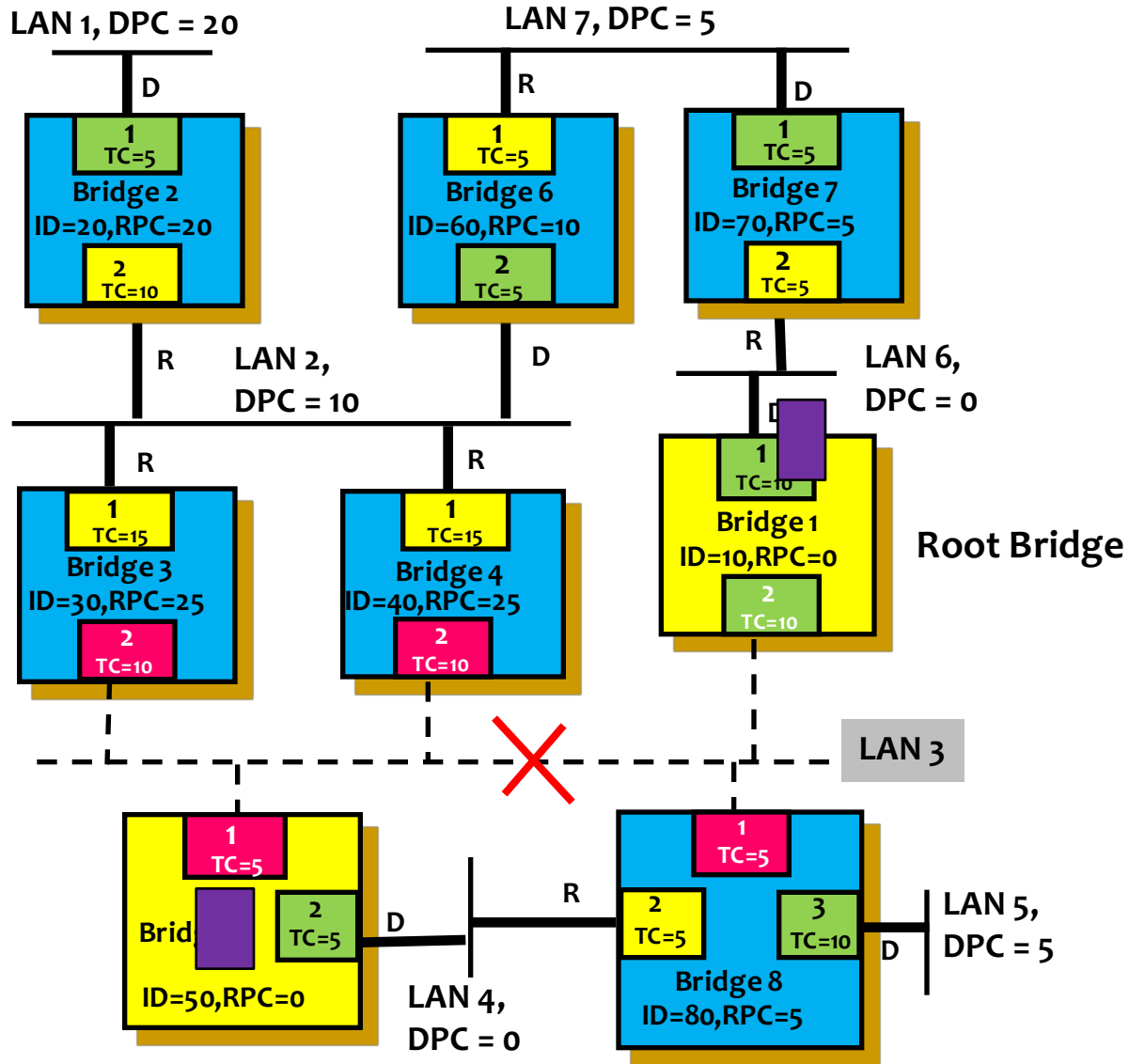
- 假設區域網路 3 故障。
- 則所有由根橋接器送出的 Hello BPDU 將無法送達區域網路 3。
- 所有連結在區域網路 3 上的埠，包括橋接器 3 的埠 2，橋接器 4 的埠 2，橋接器 5 的埠 1，以及橋接器 8 的埠 1，將會由原先的“forwarding”轉送狀態轉為“block”阻斷狀態。
- 所有的橋接器將不會再有“R”埠 (根埠) 而且他們將會開始競爭成為根橋接器的機會。
- 橋接器 3 和橋接器 4 將可以繼續由埠 1 收到 Hello BPDU，所以他們會把自己的根埠該為埠 1。

擴張樹的維護

範例二 (區域網路毀損)

- 橋接器 5 和橋接器 8 將會遵循擴張樹的協議，以交換BPDU的方式來競爭成為根橋接器的機會。
- 假設橋接器8先送出一個RPC=0的 BPDU。
- 橋接器 5 將會回傳另一個RPC=0的BPDU因為他認為自己的優先權比橋接器8 高 (擁有較小的識別碼)。
- 經過了兩個前送延遲後，橋接器 5 將會成為跟橋接器，而橋接器 8 的埠 1 將會成為根埠。
- 最後，我們將會有兩個分離的(互相沒有連結)擴張樹。

範例二最終架構



總結

- 橋接器是第二層設備，用來連接並協助實體分離區域網路上的訊框轉送，以讓終端主機將這些實際分離的區域網路視為邏輯上同一個區域網路。
- 橋接器的基本功能：
 - 訊框的轉送以及過濾
 - 位址學習
 - 移除在拓樸中可能出現的迴圈
- 由IEEE 802.1D 的演算法造出的擴張樹會包含每個橋接器到根橋接器的最短路徑(最小的根路徑費用, RPC).

總結

- 每個區域網路經由代理橋接器連向根橋接器的最短路徑都(**minimum root path cost, RPC**)會被包含在擴張樹內。
- 橋接網路的擴張樹是可預測的或是可計算出來的。
- 依據橋接網路的拓撲(可包含一些迴圈), 其擴張樹可以被推算出來。
- 擴張樹演算法有能力維護一個擴張樹, 並且處理可能發生的**橋接器故障**以及**區域網路故障**。