

# Introduction to Computer Networks



## IEEE 802.1D Spanning Tree Algorithm

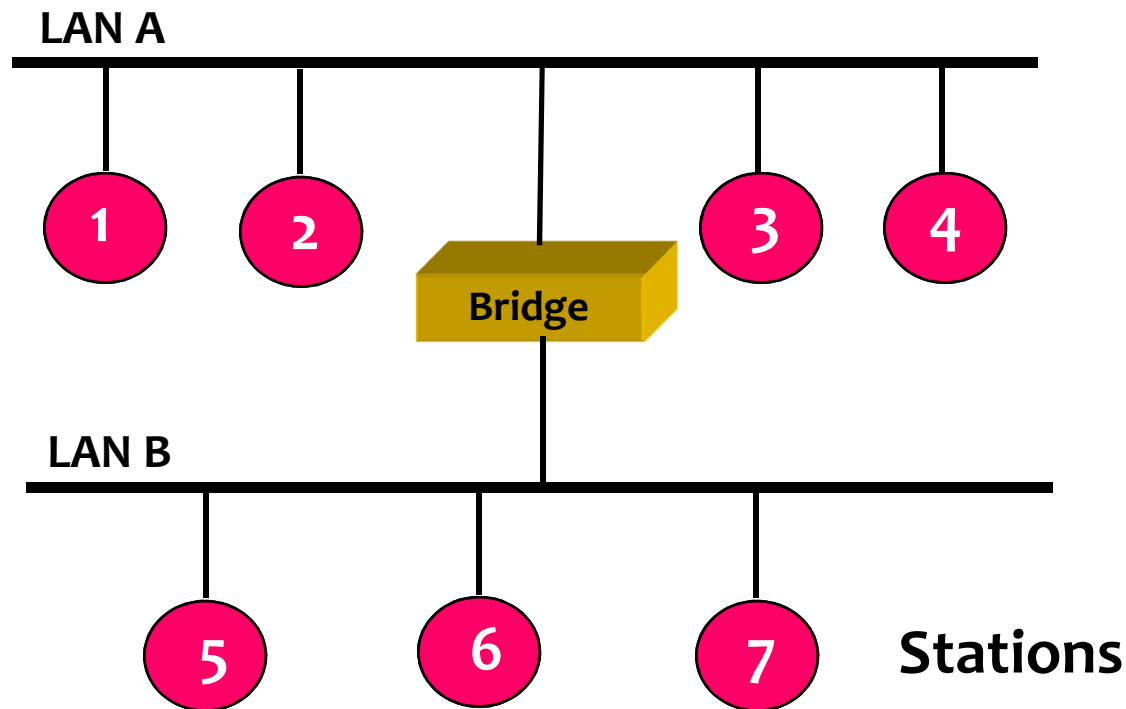
© All rights reserved. No part of this publication and file may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without prior written permission of Professor Nen-Fu Huang (E-mail: [nfhuang@cs.nthu.edu.tw](mailto:nfhuang@cs.nthu.edu.tw)).

# Outline

---

- **Introduction**
- **Frames Forwarding and Addresses Learning**
- **Loop Problem and Resolution**
- **Spanning Tree Algorithm**
- **Spanning Tree Maintenance**

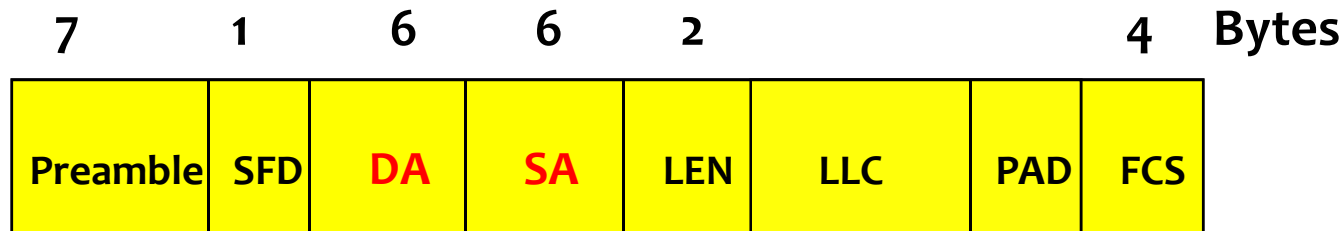
# A Simple Bridge Example



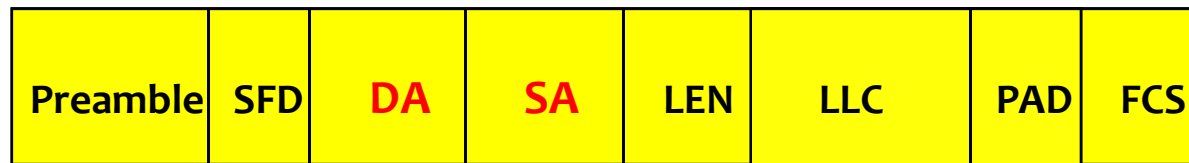
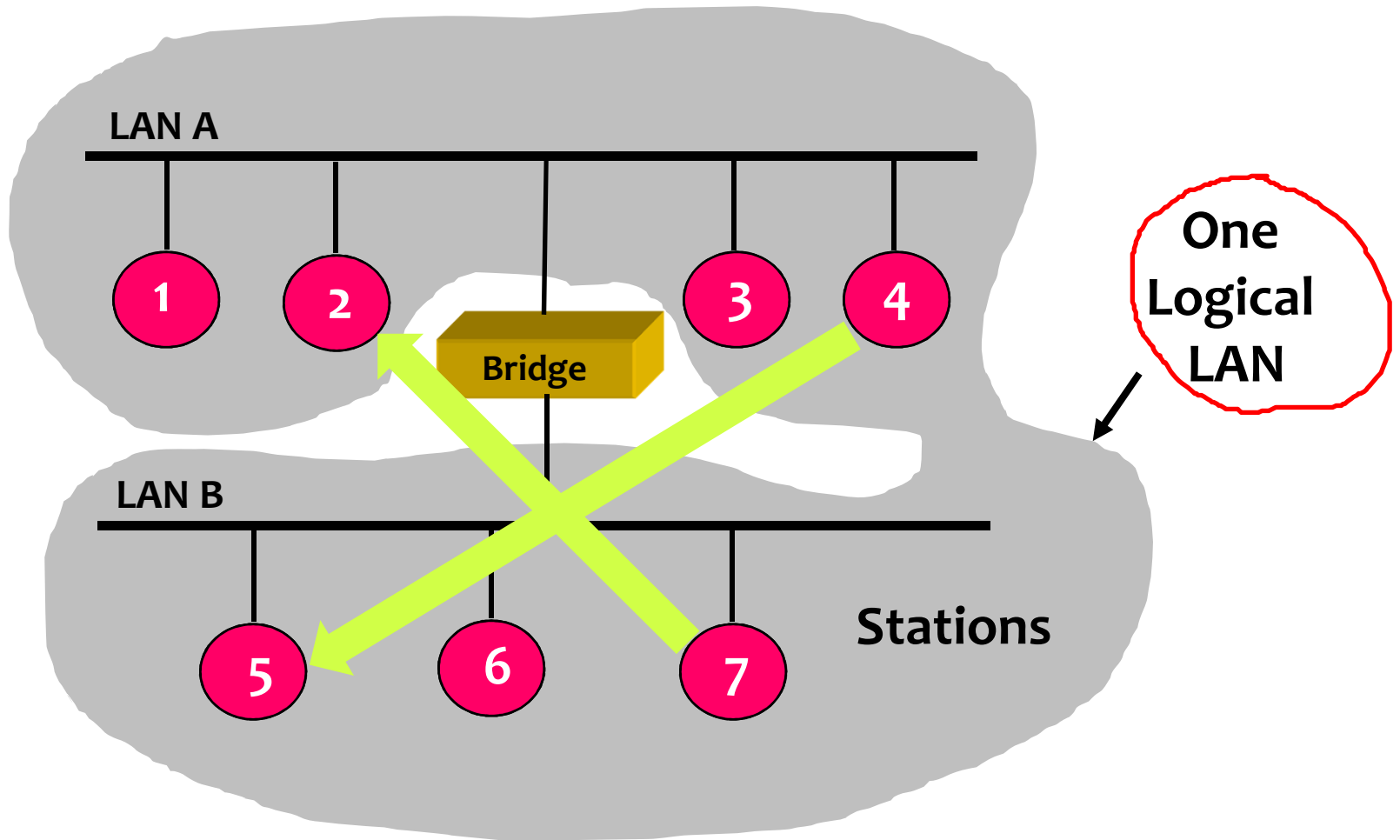
# What is a Bridge ?

---

- A bridge is a **MAC layer (layer 2) device** which relays frames among physically separated LANs and makes the physical LANs appear as **one logical LAN** to the end stations



# The Concept of One Logical LAN



$MAC_g$   $MAC_r$

# Functions of a Bridge

---

## ■ Basic Functions:

- Frame **Forwarding** and **Filtering**
- Address **Learning**
- Resolving Possible **Loops** in the Topology

## ■ Additional Functions:

- Congestion Control (Enough Buffer)
- Static Filtering (Security)
- Translation (Multi-Bridge)
- Routing (Multi-Bridge)
- Segmentation

# Design Considerations

---

- No modifications to the content or format of the frames
- Contain enough buffer space to meet peak demands
- Contain addressing and routing intelligence
- A bridge may connect more than two networks
- Why **Bridged LANs (BLAN)** ?
  - Reliability
  - Performance
  - Security
  - Geography

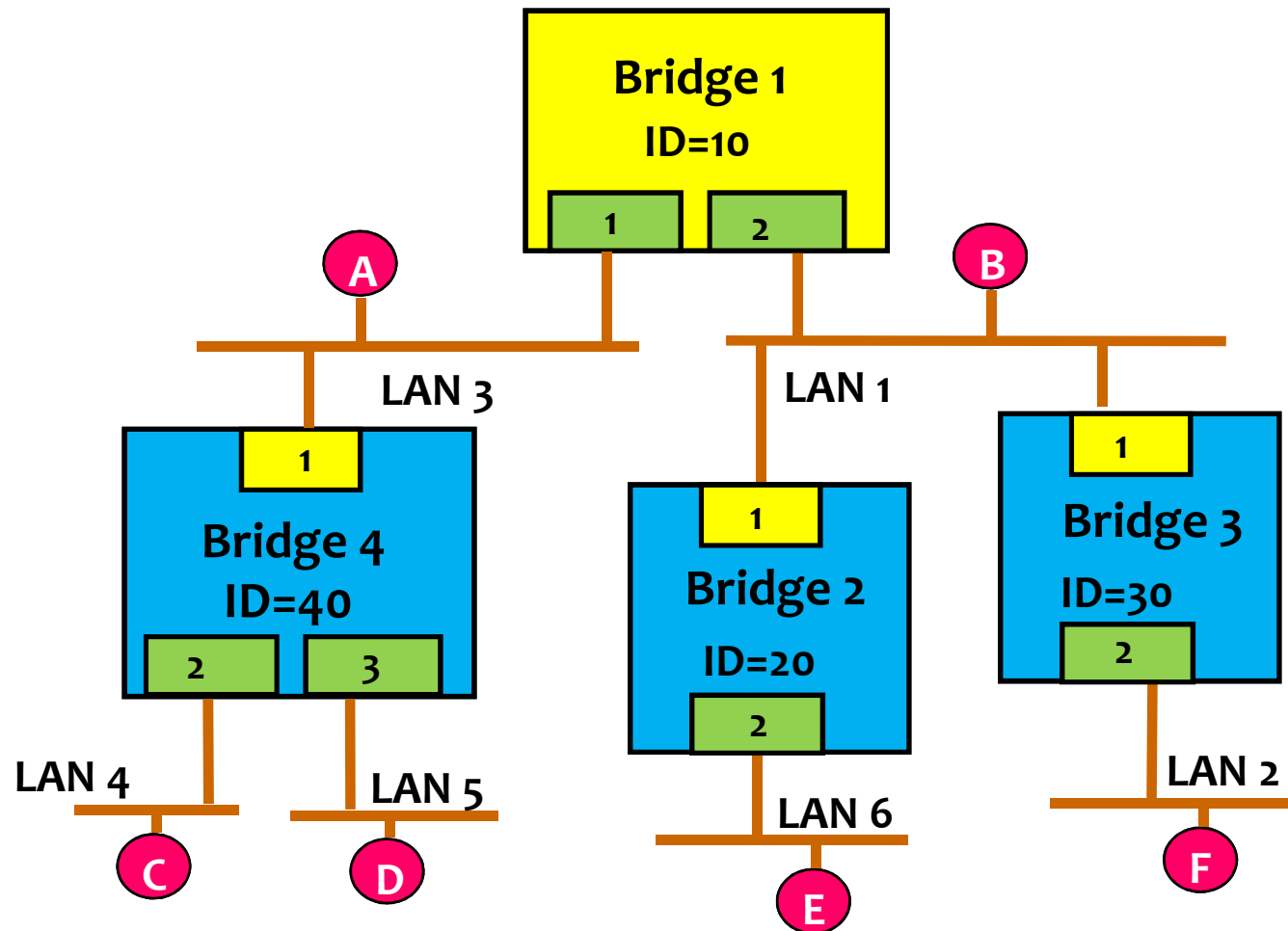
# Bridge Routing

---

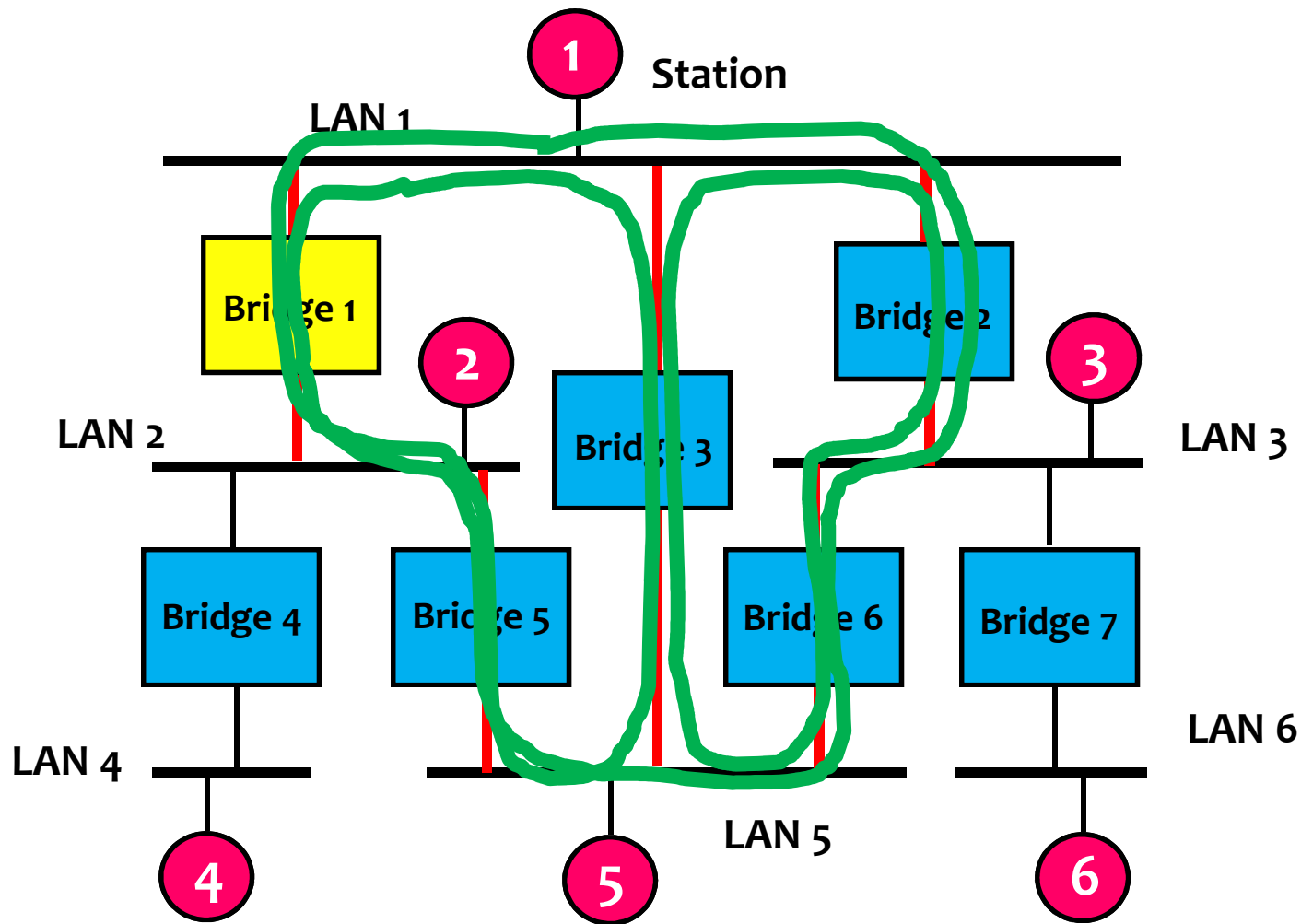
- The Bridges must be equipped with a **routing capability**
- The routing decision may not always be a simple one (loop)
- **Topology changes** have to be considered
- A bridge knows all the station addresses (**Filtering Database**)



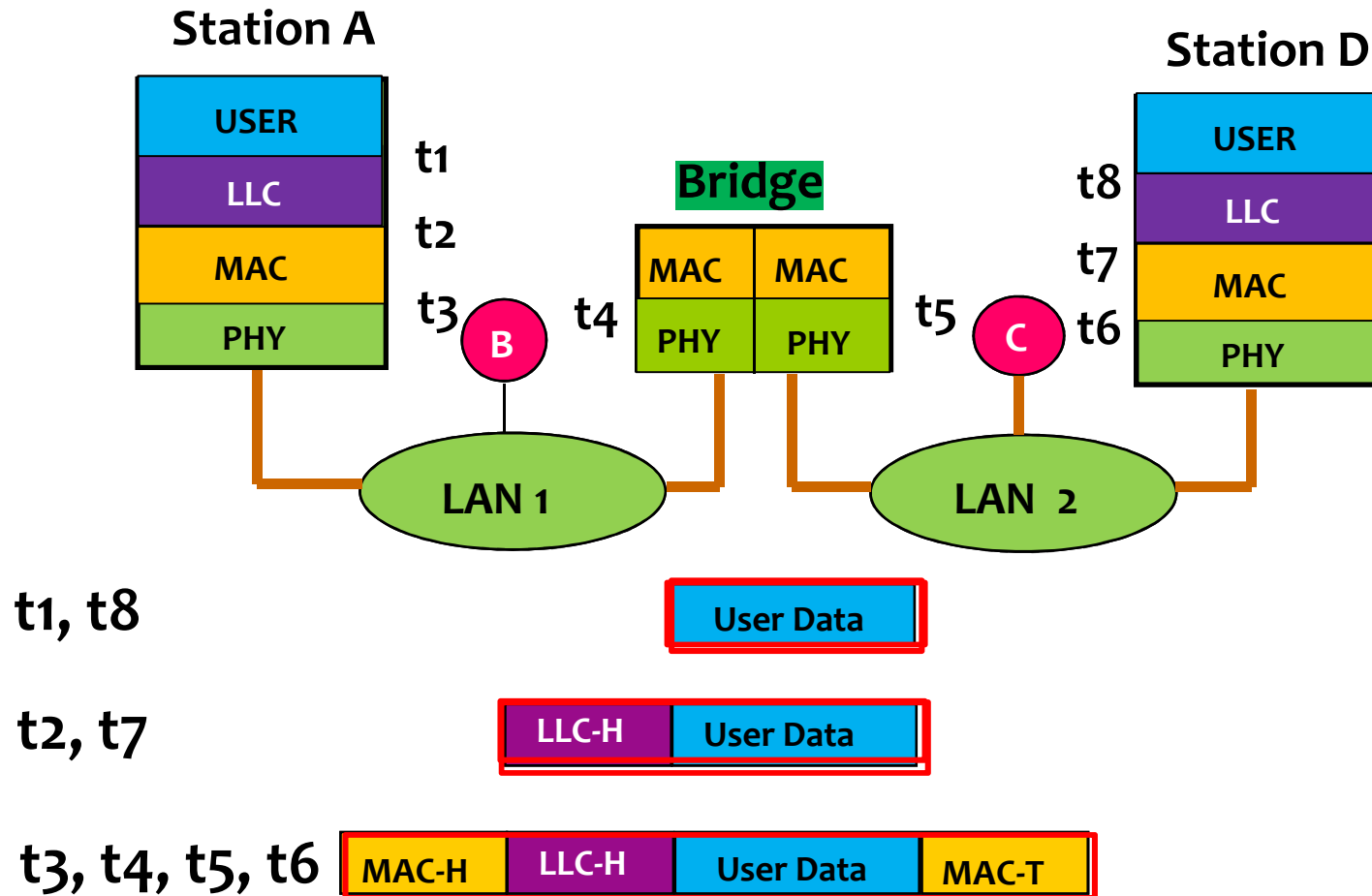
# A BLAN Example Without loop



# A BLAN Example with Loops



# Bridge Protocol Architecture



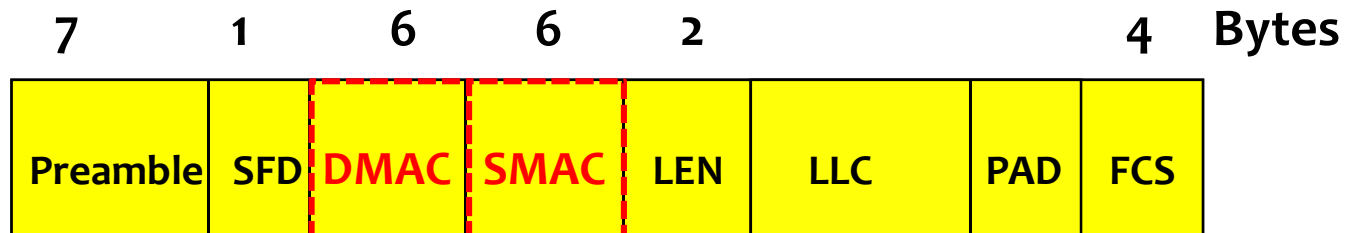
# Spanning Tree Routing

## ■ Frame Forwarding and Filtering

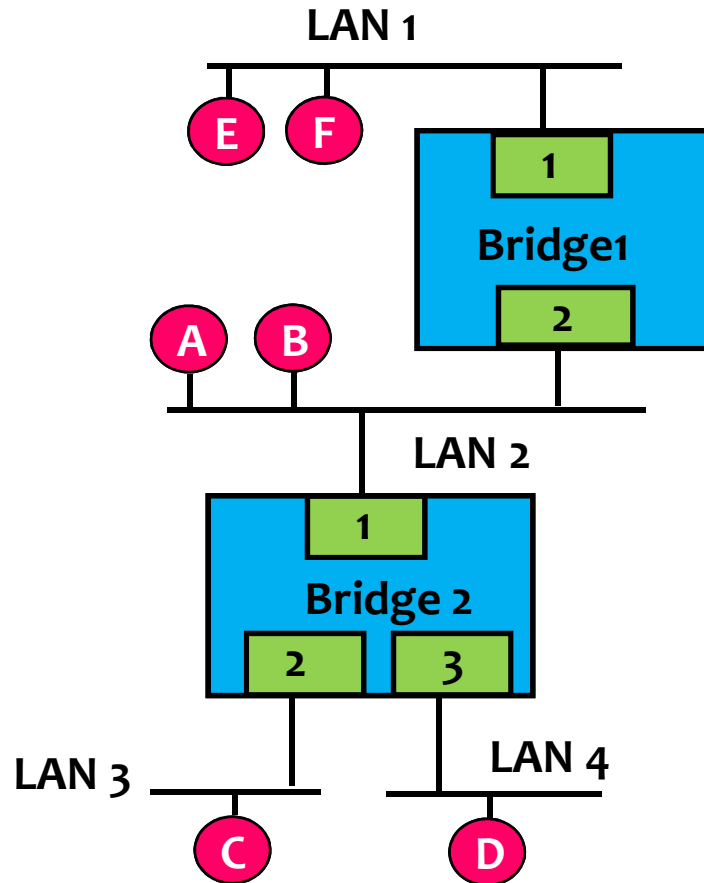
- Use the **destination MAC address (DMAC)** field in each MAC frame
- A bridge maintains a filtering database with entries:  
[Address, Port, Time]

## ■ Address Learning

- Use the **source MAC address (SMAC)** field in each MAC frame
- If the element is **already** in the database, the entry is **updated and the timer is reset**  
第一次是广播
- If the element is not in the database, a new entry is created with its own timer



# Filtering Database Examples



## Filtering Database (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

## Filtering Database (Bridge 2)

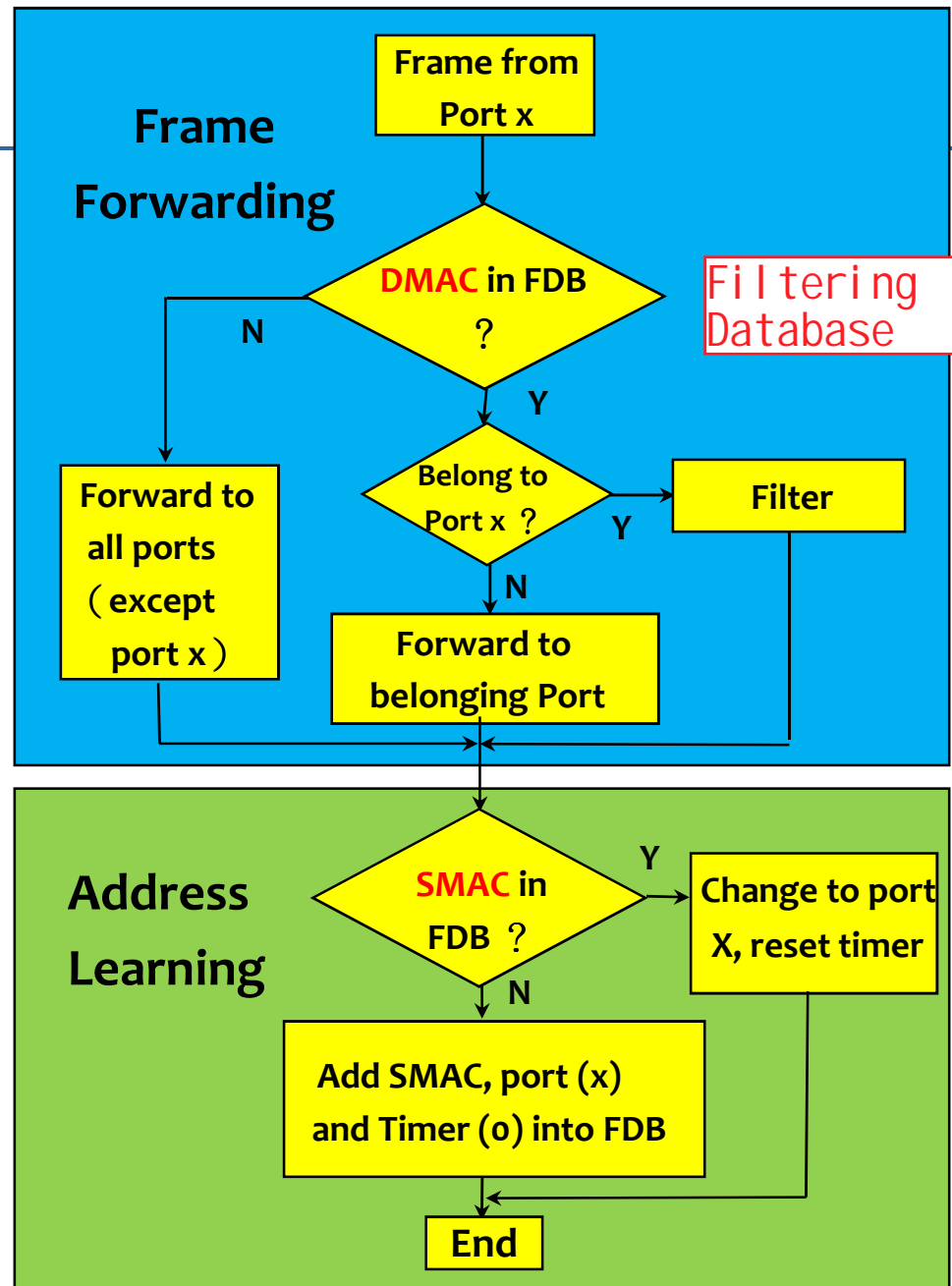
MAC Addr	Port	Time(S)
A	1	19
B	1	17
C	2	24
D	3	3
E	1	6
F	1	13

# Outline

---

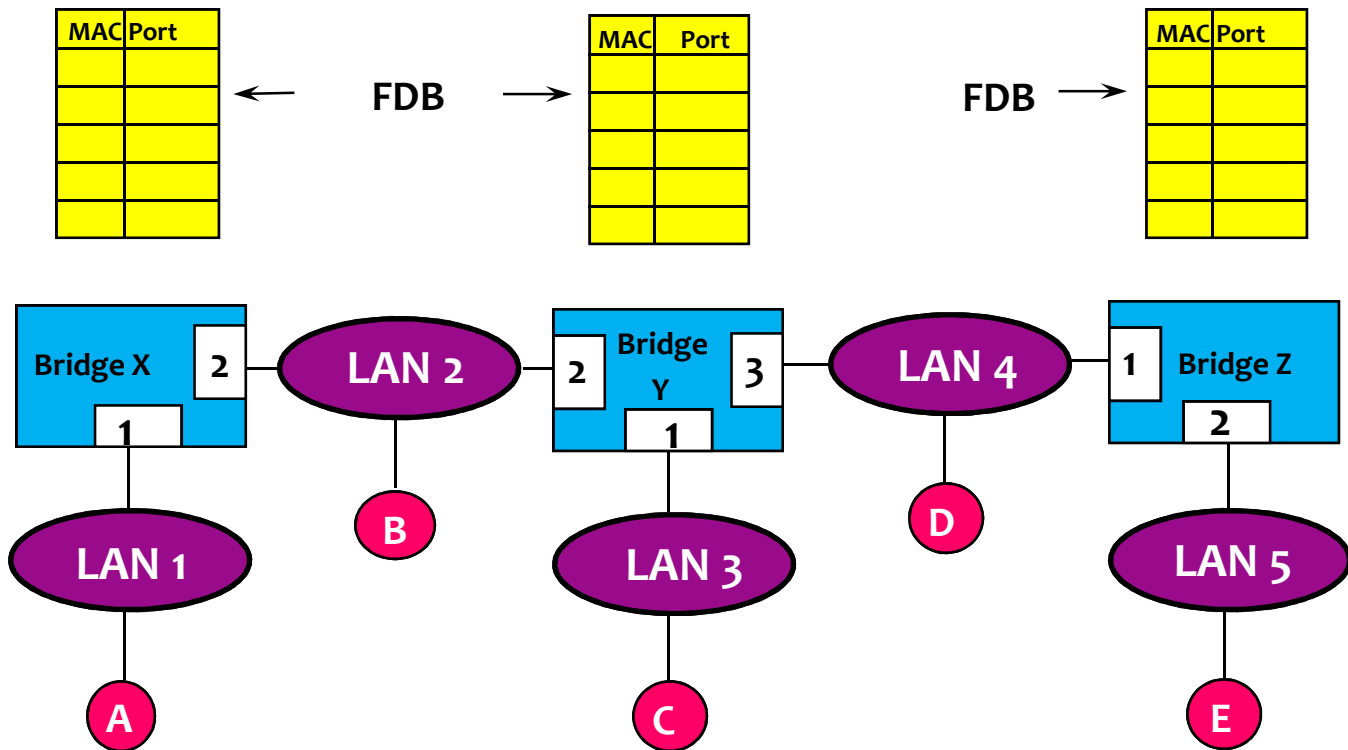
- Introduction
- **Frames Forwarding and Addresses Learning**
- Loop Problem and Resolution
- Spanning Tree Algorithm
- Spanning Tree Maintenance

# Forwarding and Address Learning Algorithm



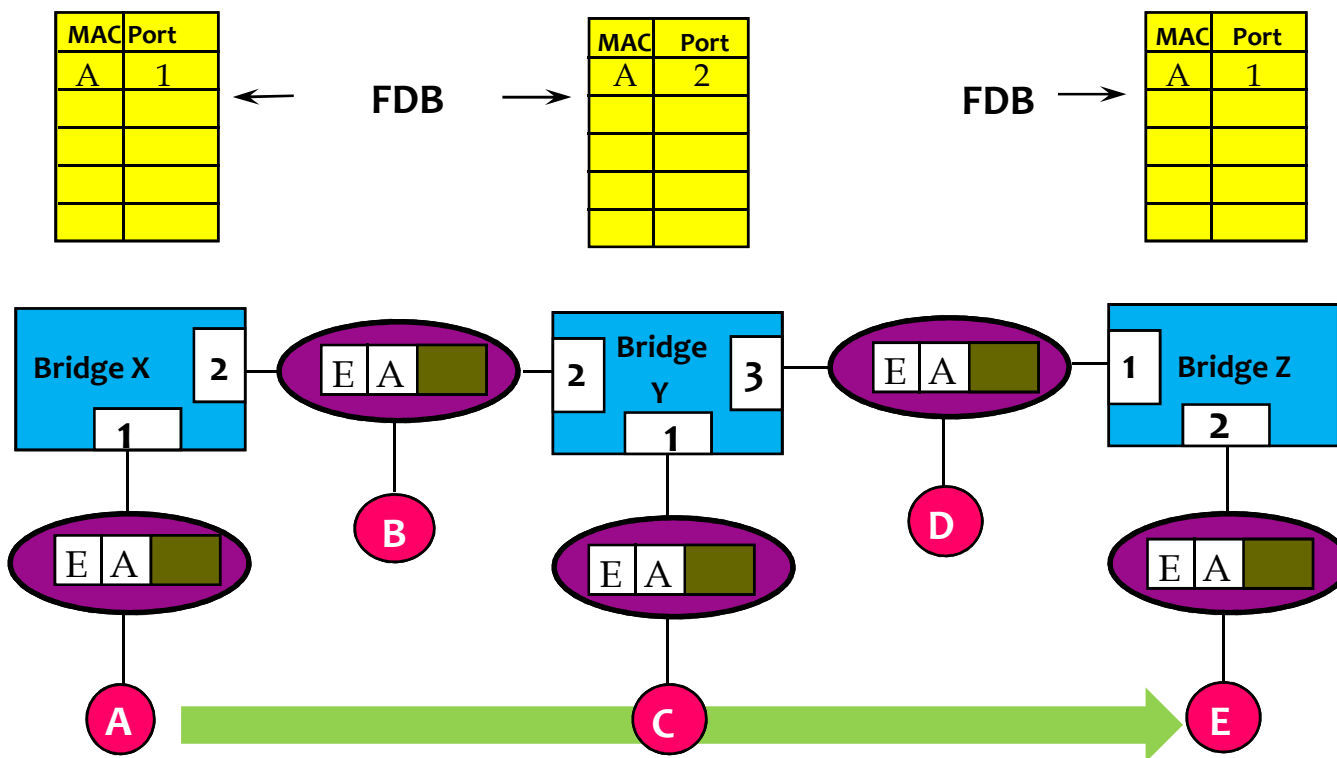
# Addresses Learning Example

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

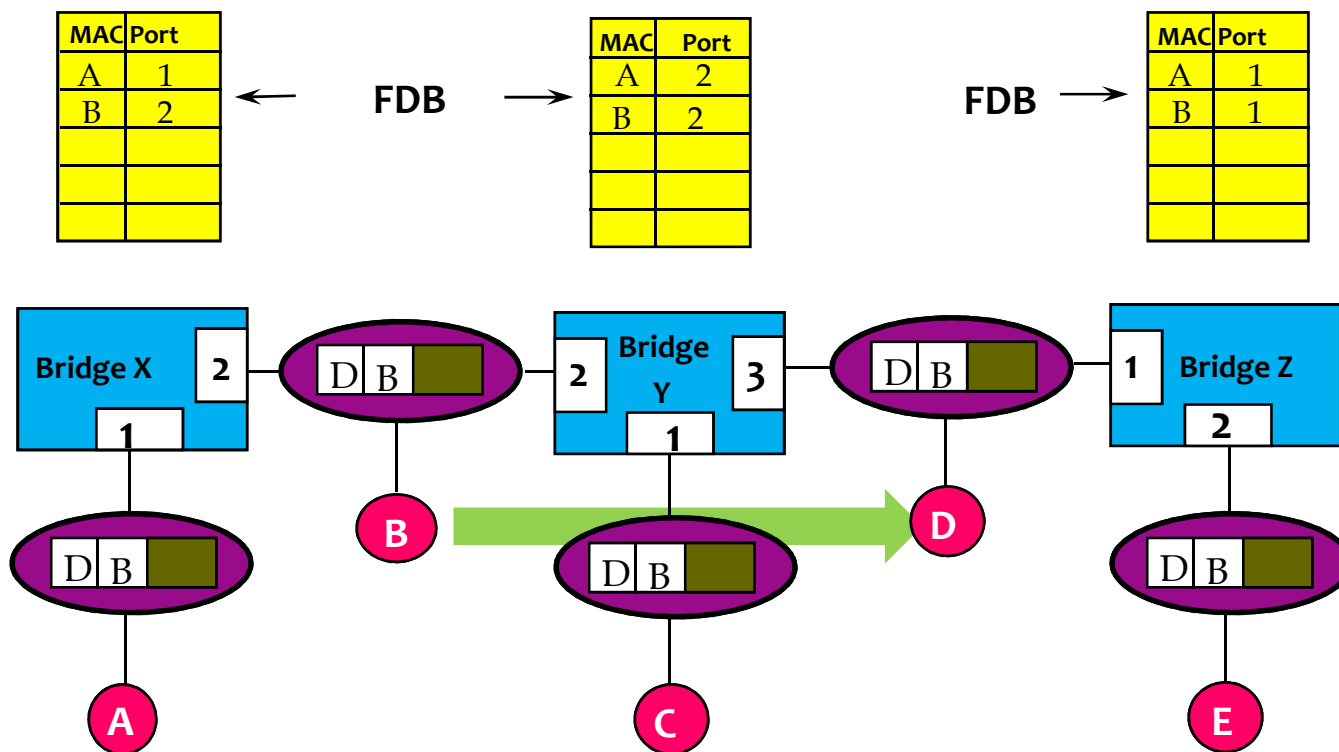




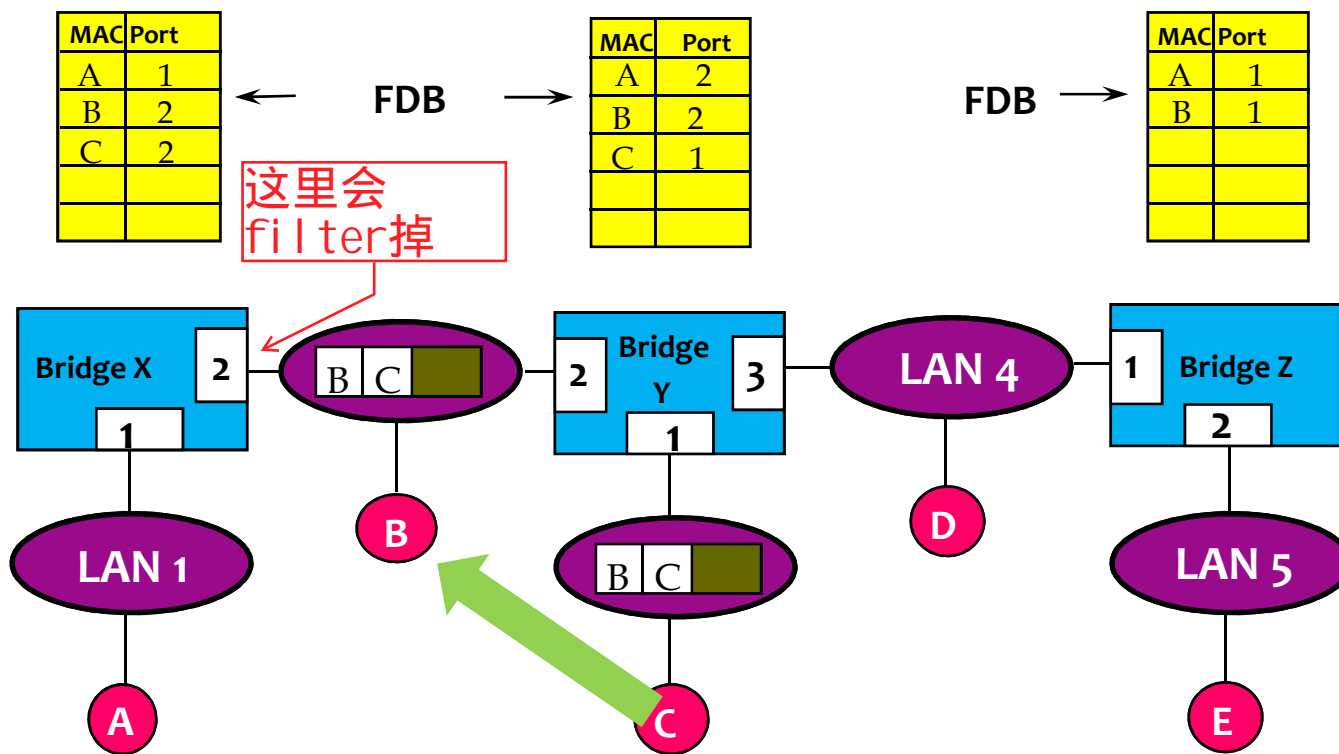
# Addresses Learning Example (A→E)



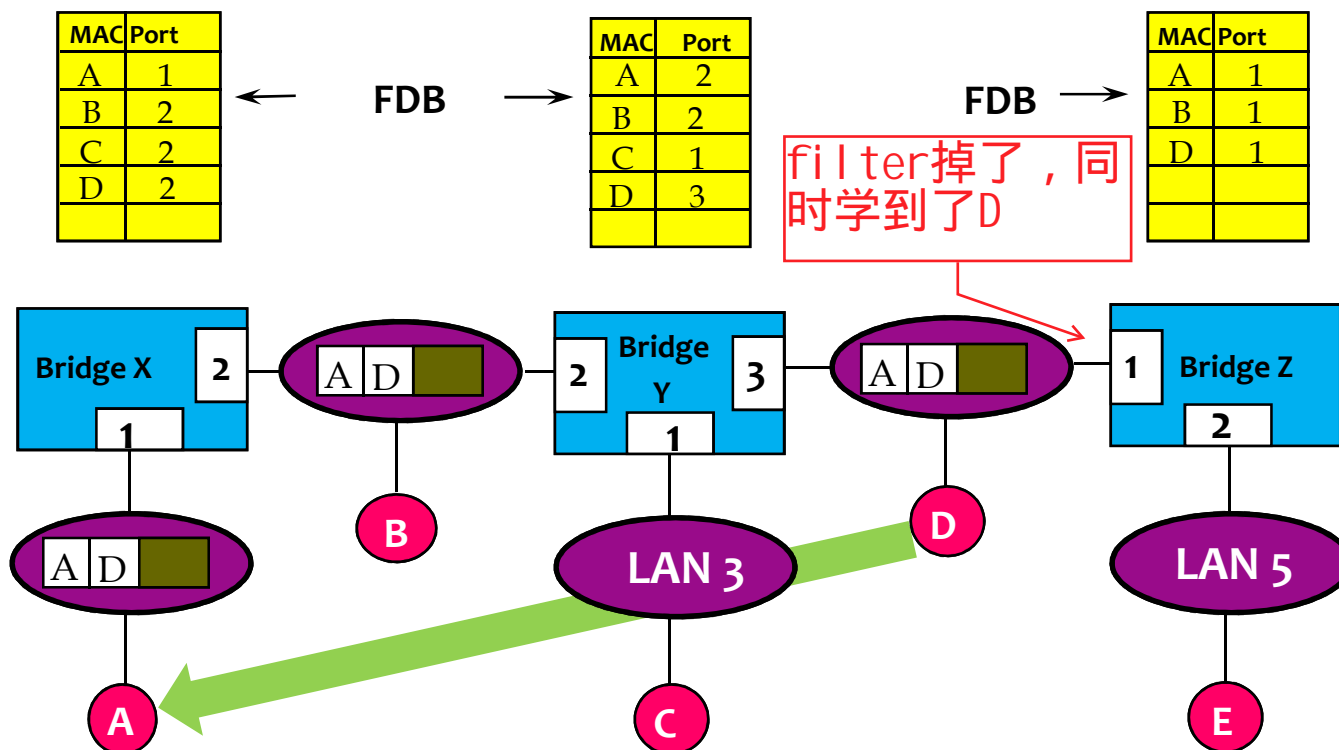
# Addresses Learning Example (B→D)



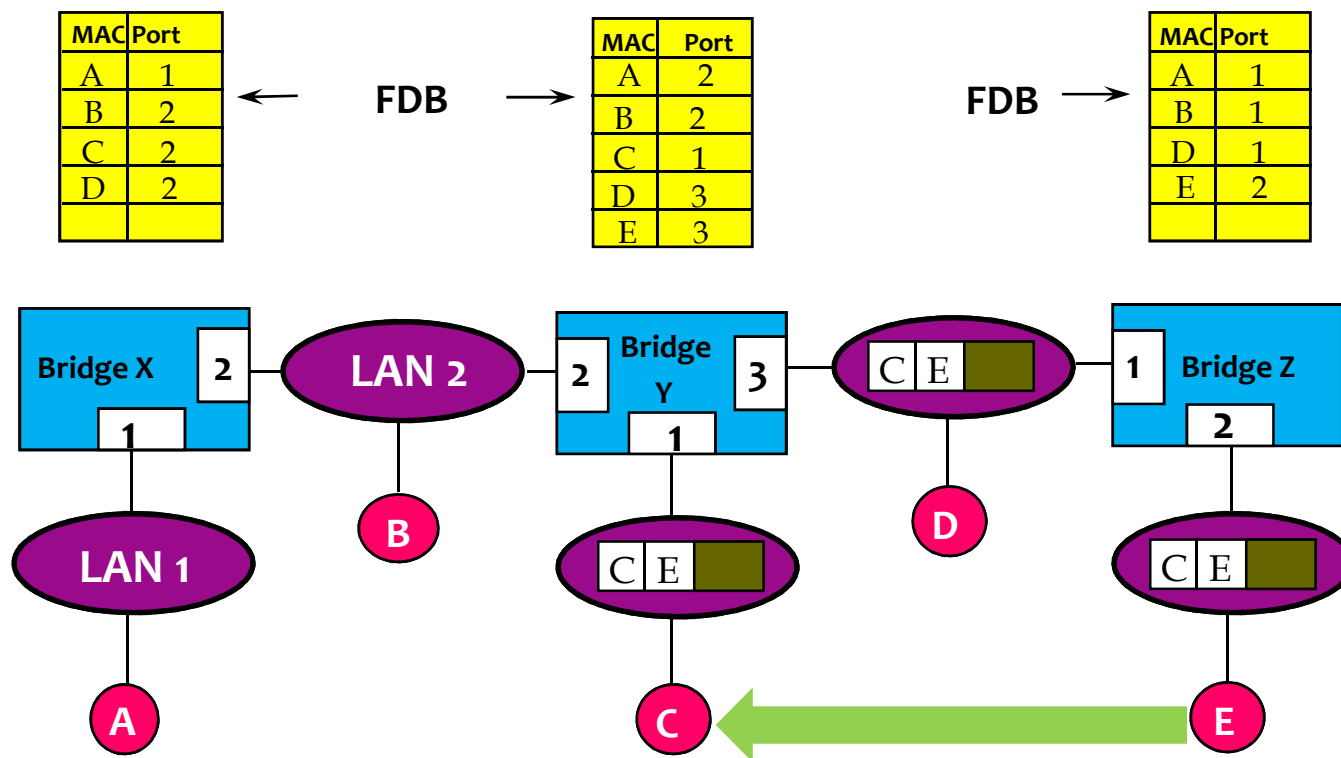
# Addresses Learning Example (C→B)



# Addresses Learning Example (D→A)



# Addresses Learning Example (E→C)



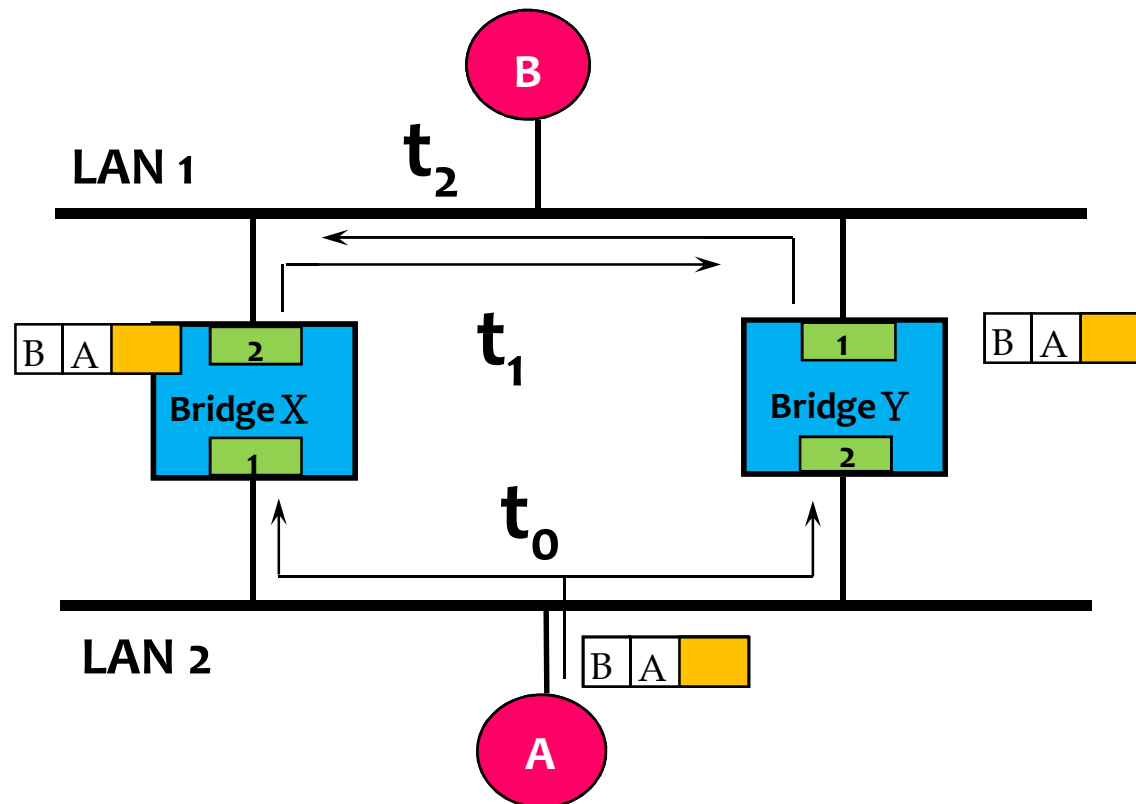
# Outline

---

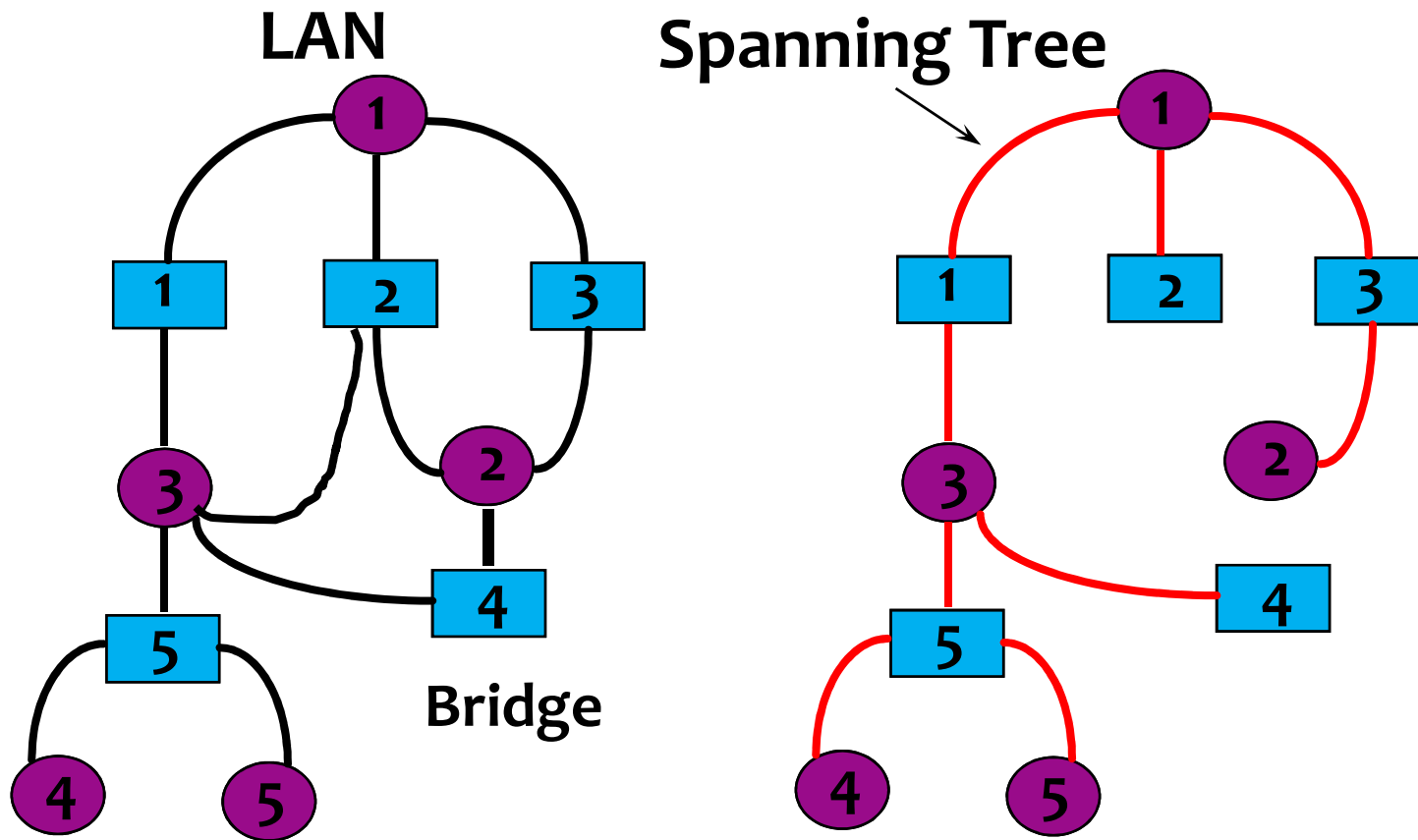
- Introduction
- Frames Forwarding and Addresses Learning
- **Loop Problem and Resolution**
- Spanning Tree Algorithm
- Spanning Tree Maintenance

# Loop Problems and Resolution

- Loops provide network reliability
- But loops make frames duplication
- Loops also make wrong address learning

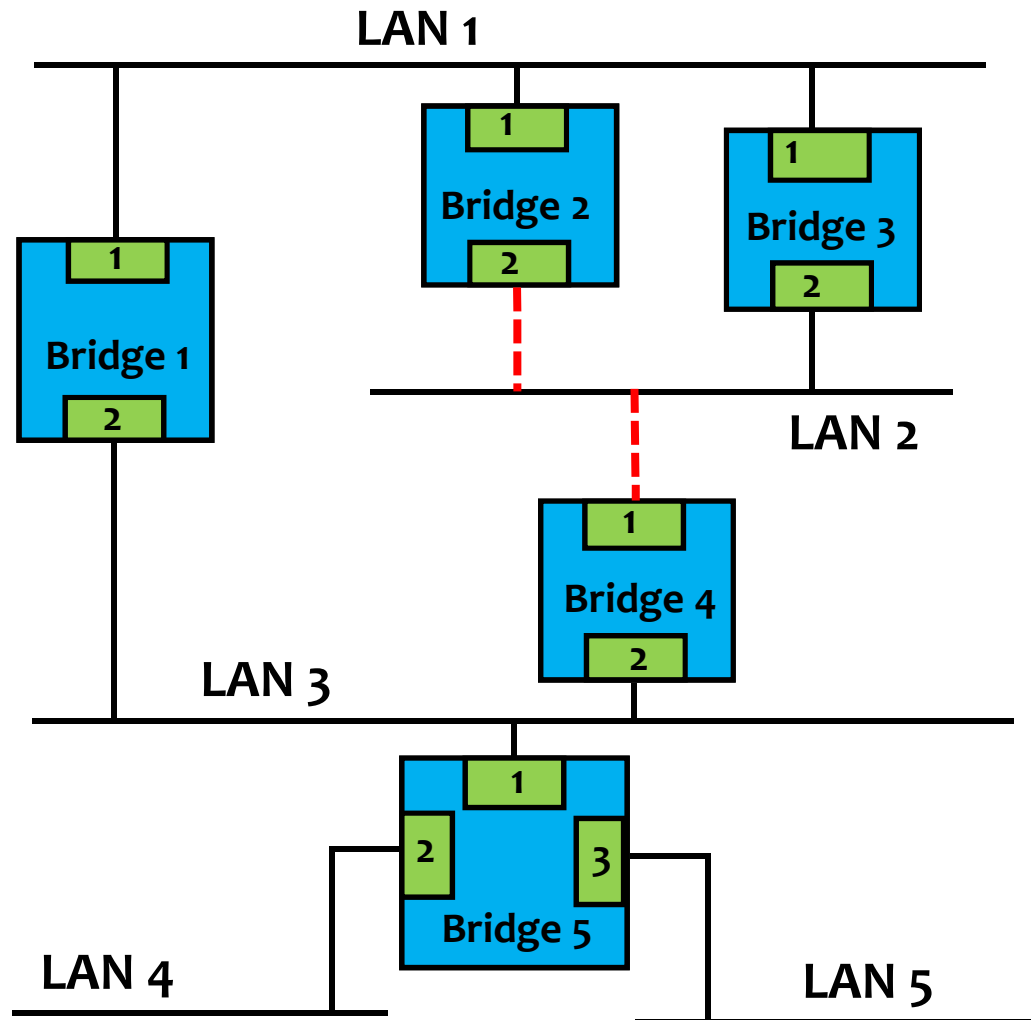


# Graph Representation of a BLAN

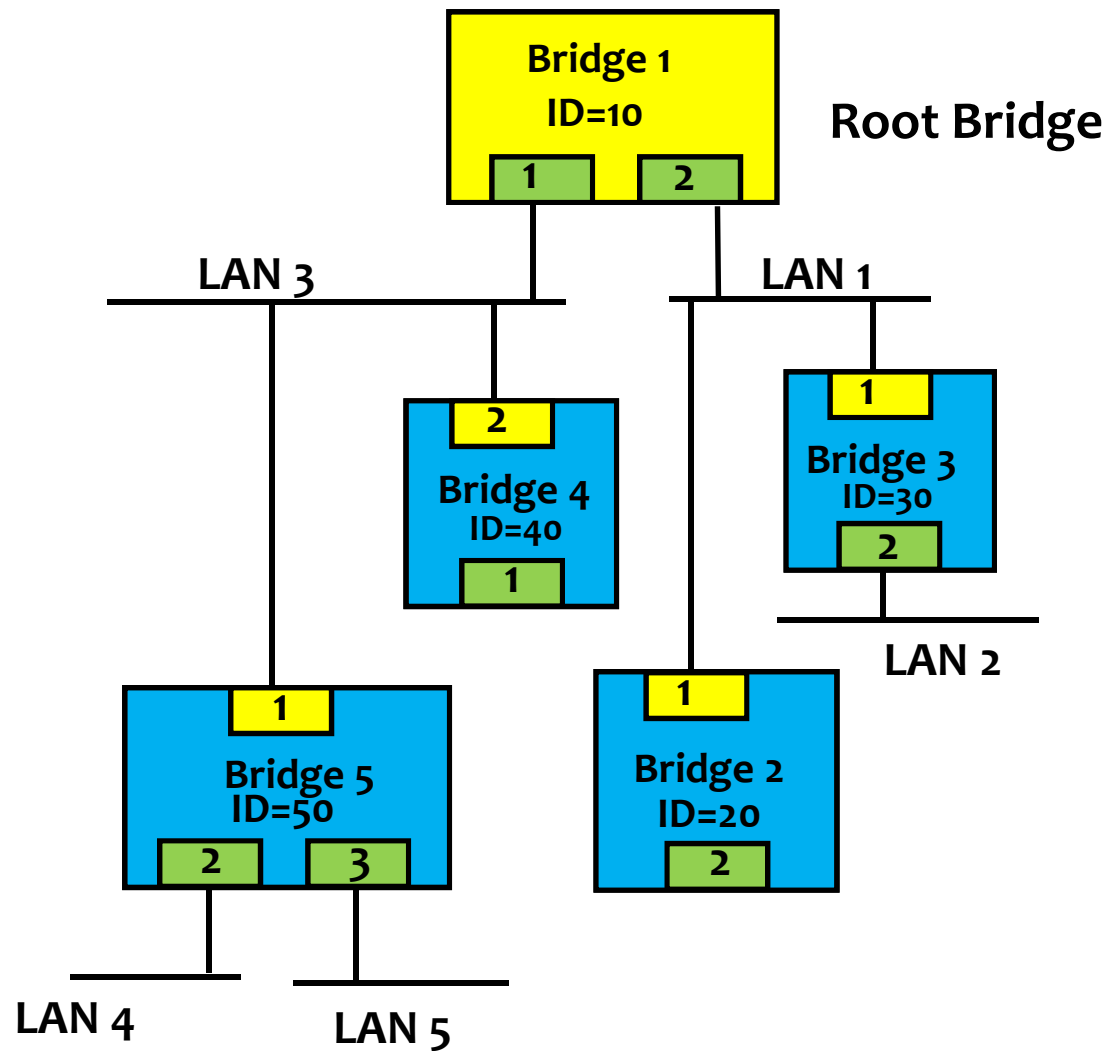




# Spanning Tree Example 1



# Spanning Tree Example 1 (Continued)



# Outline

---

- Introduction
- Frames Forwarding and Addresses Learning
- Loop Problem and Resolution
- **Spanning Tree Algorithm**
- Spanning Tree Maintenance

# Spanning Tree Algorithm (requirements)

---

## ■ Bridges

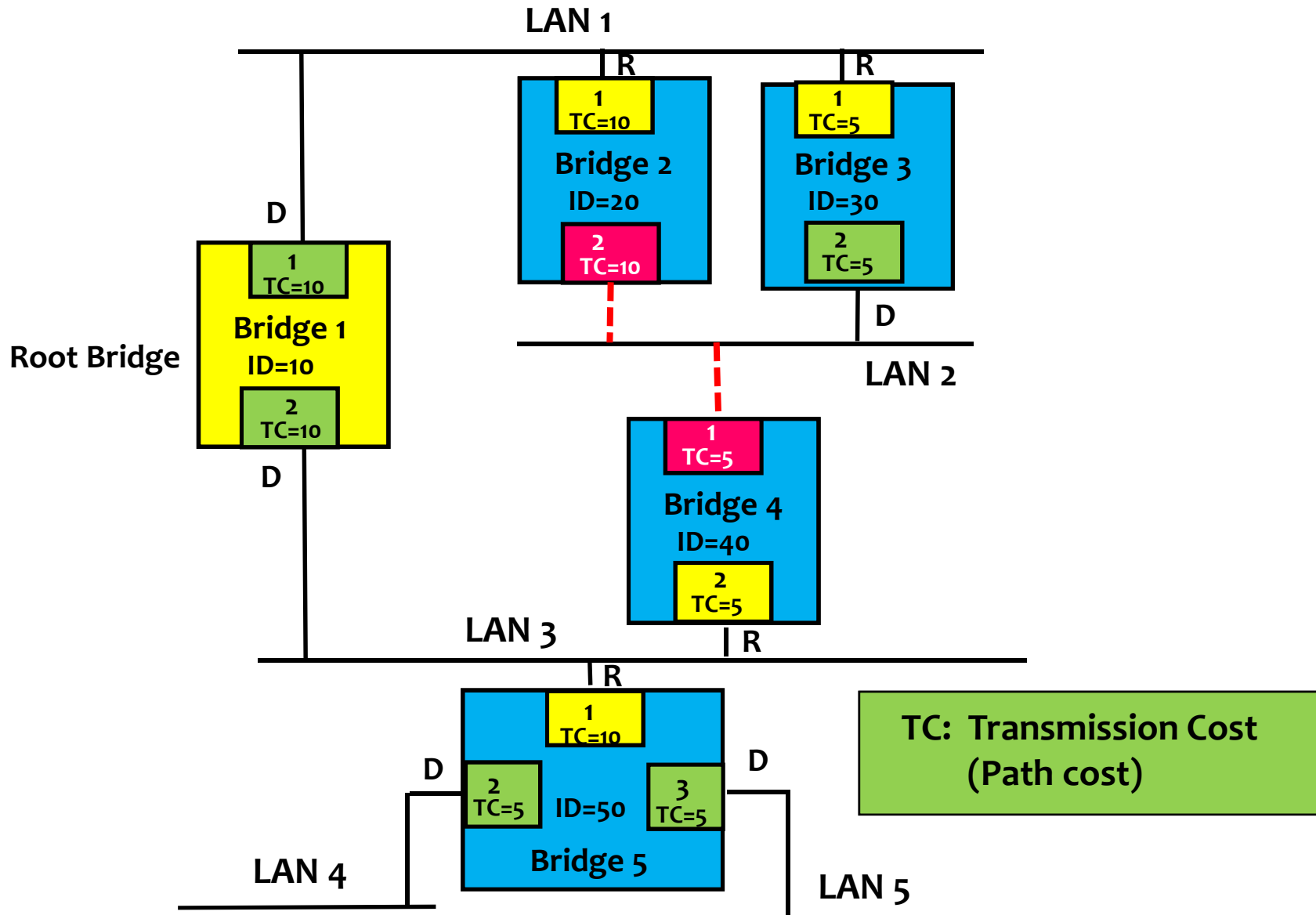
- Each bridge is assigned a unique identifier (8 octets):
  - ▶ Priority part (two octets): programmable
  - ▶ address part (six octets): MAC address
- A special group MAC address for all bridges :  
**01-80-C2-00-00-00 (Multicast address)**  
**10000000-00000001-01000011-**
- Each port of a bridge has a unique port identifier.

# Spanning Tree Algorithm (definitions)

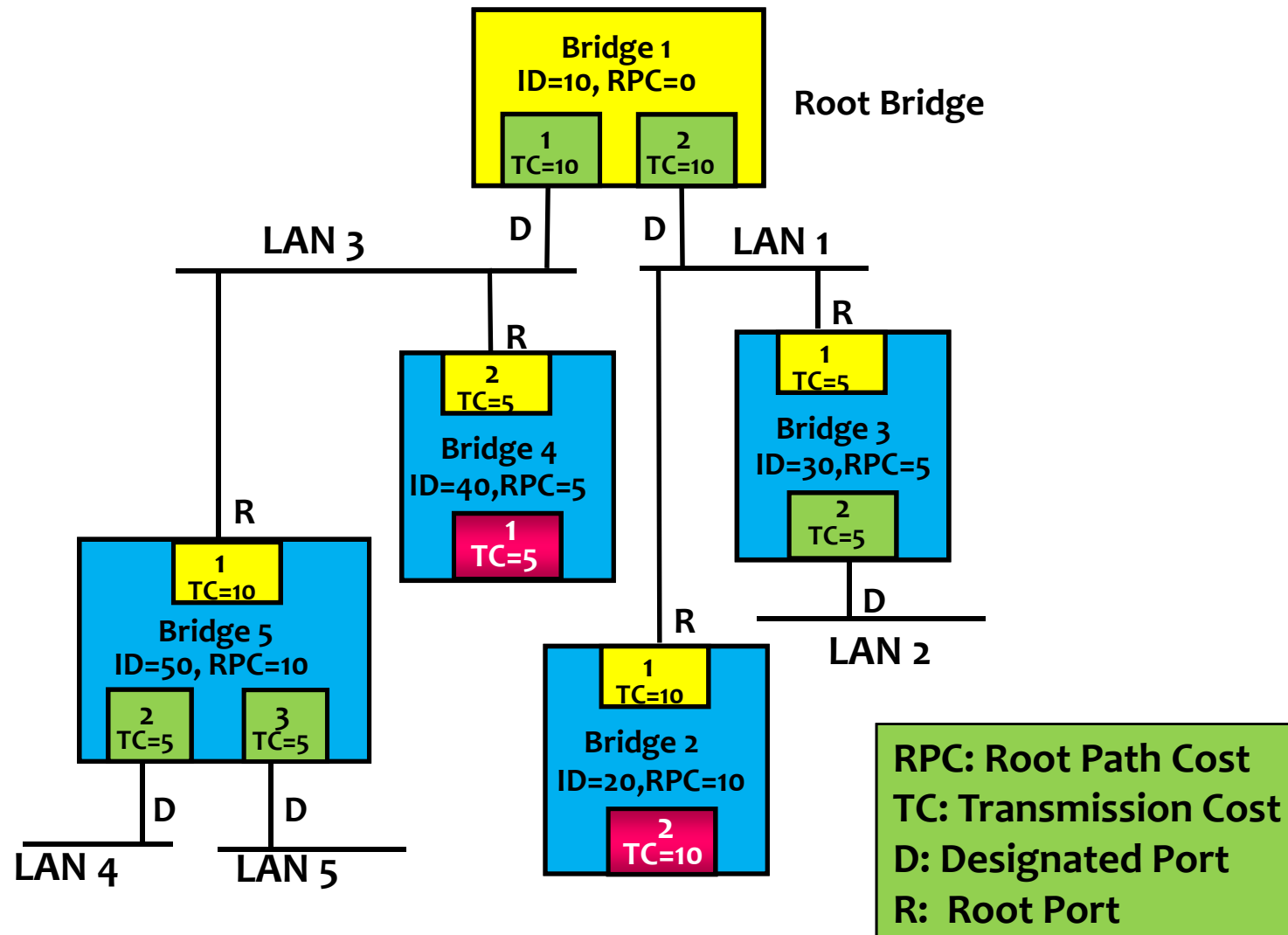
---

- **Root Bridge:** The bridge with the lowest value of bridge identifier.
- **Path Cost:** For each port, the cost of transmitting a frame onto a LAN.
- **Root Port:** For each bridge, the port on the minimum-cost path to the root bridge.
- **Root Path Cost:** For each bridge, the cost of the path to the root bridge with minimum cost.
- **Designated Bridge:** For each LAN, the bridge that provides the minimum cost path to the root bridge. The only bridge allowed to forward frames to and from the LAN.
- **Designated Port:** The port of the designated bridge that attaches the bridge to the LAN. All internet traffic to and from the LAN pass through the designated port.

# Spanning Tree Example 2



# Spanning Tree Example 2 (continued)



# Spanning Tree Algorithm

---

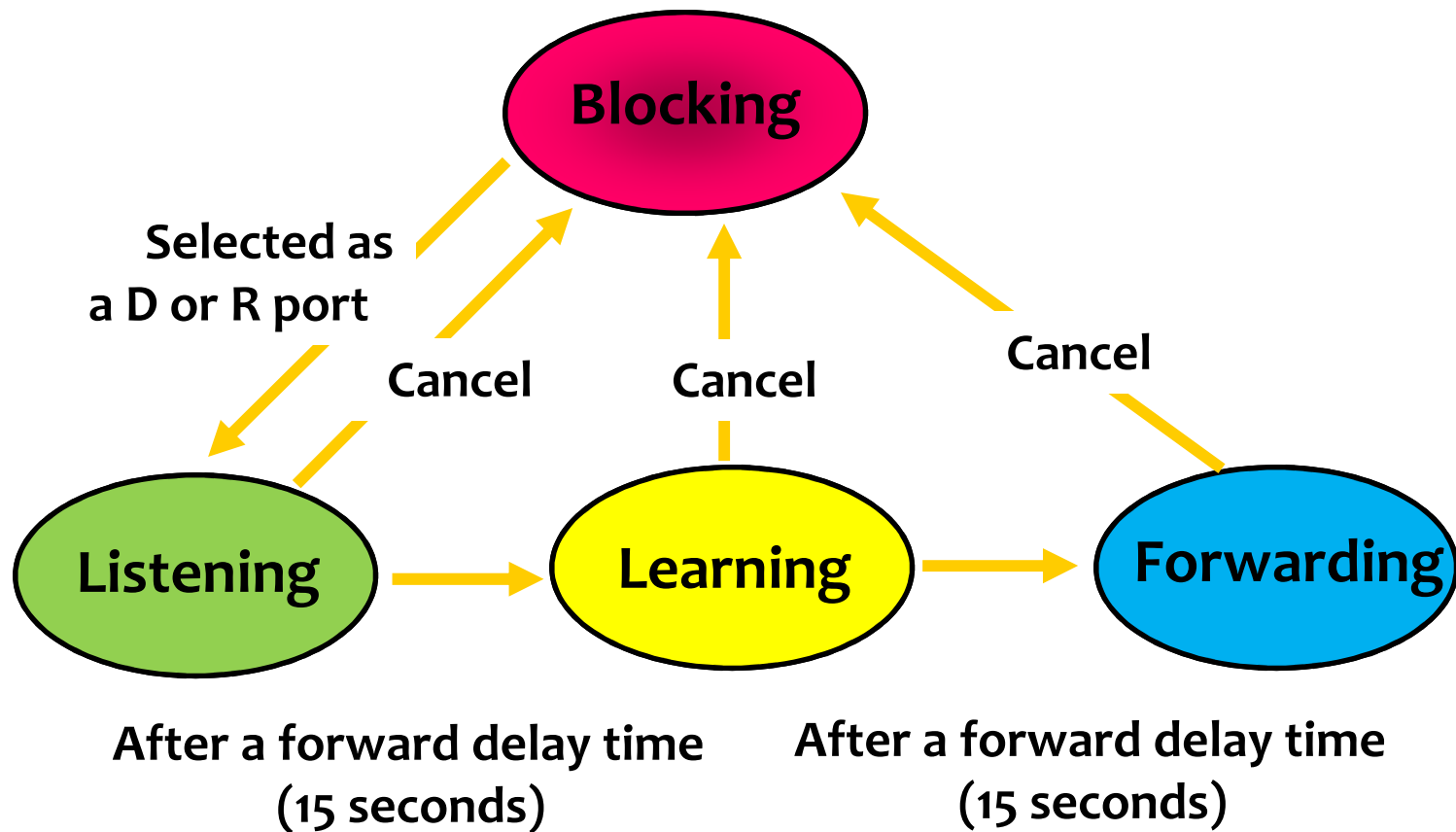
## ■ Three Steps:

1. Determine the root bridge.
2. Determine the root port on all other bridges.
3. Determine the designated port on each LAN.

- The port with the minimum root path cost.
- In the case of two or more bridges with the same root path cost, the highest-priority bridge is selected.
- If the designated bridge has two or more ports attached to this LAN, then the port with the lowest value of identifier is selected.

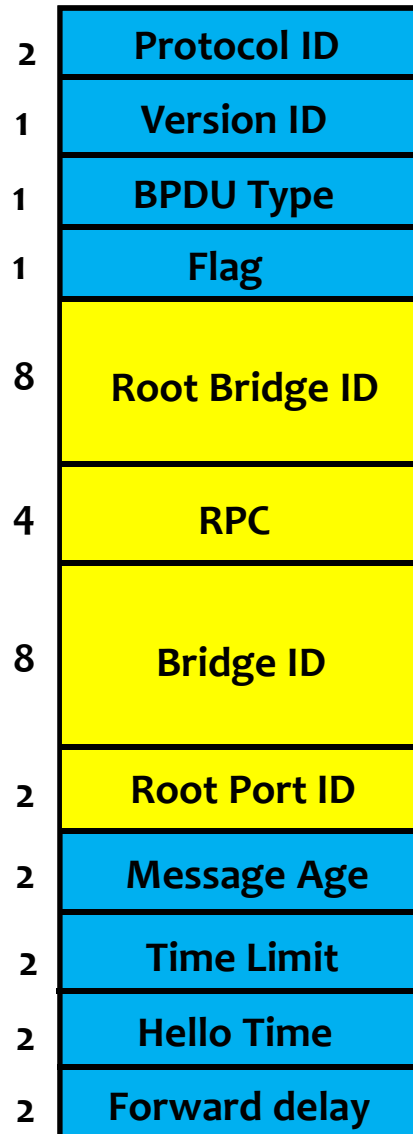


# Bridge Port State Diagram



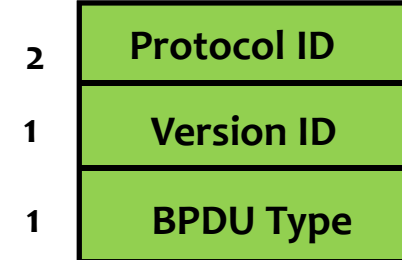
# Bridge Protocol Data Unit (BPDU)

Bytes



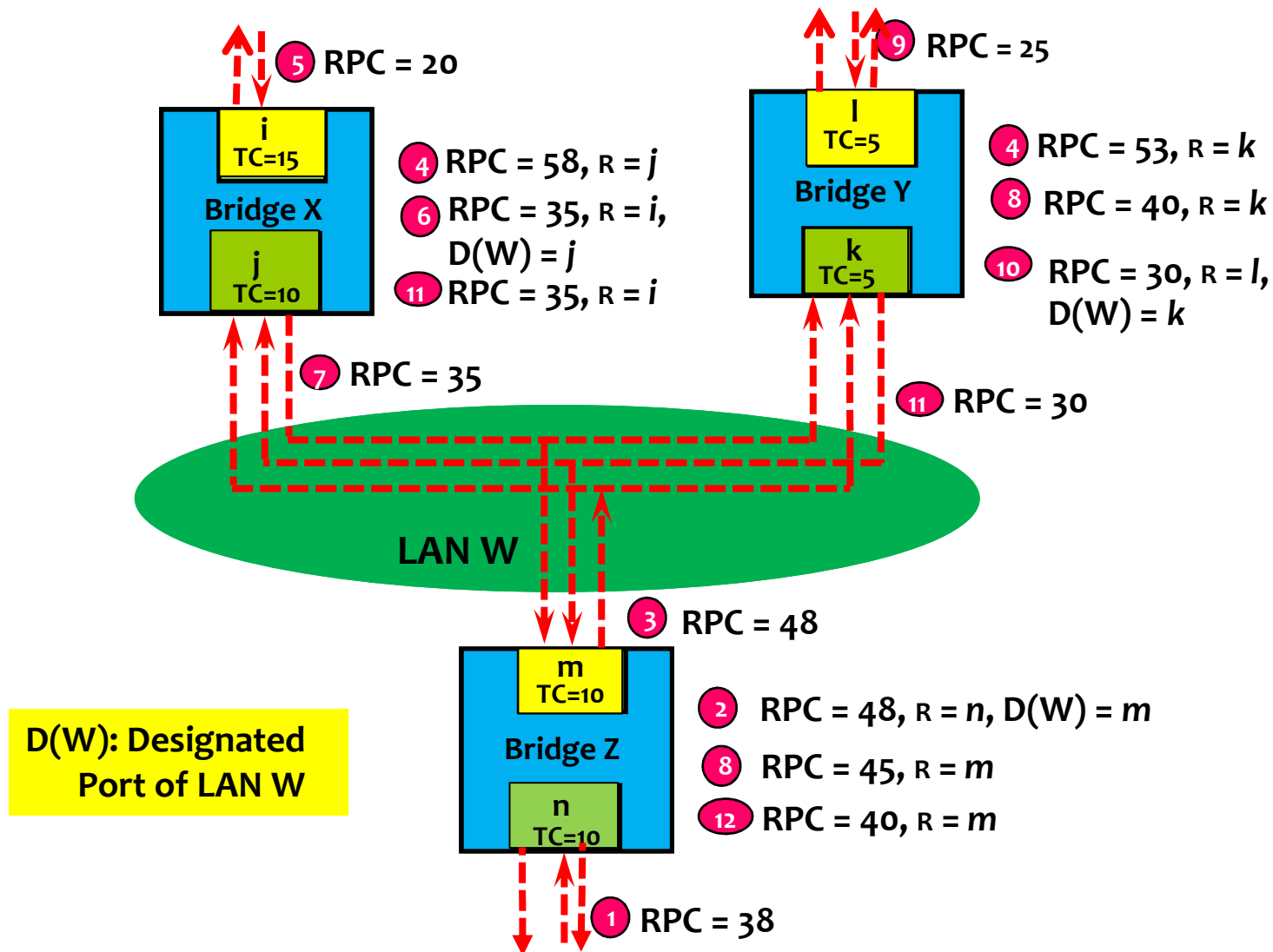
(a) Network Configuration BPDU

Bytes

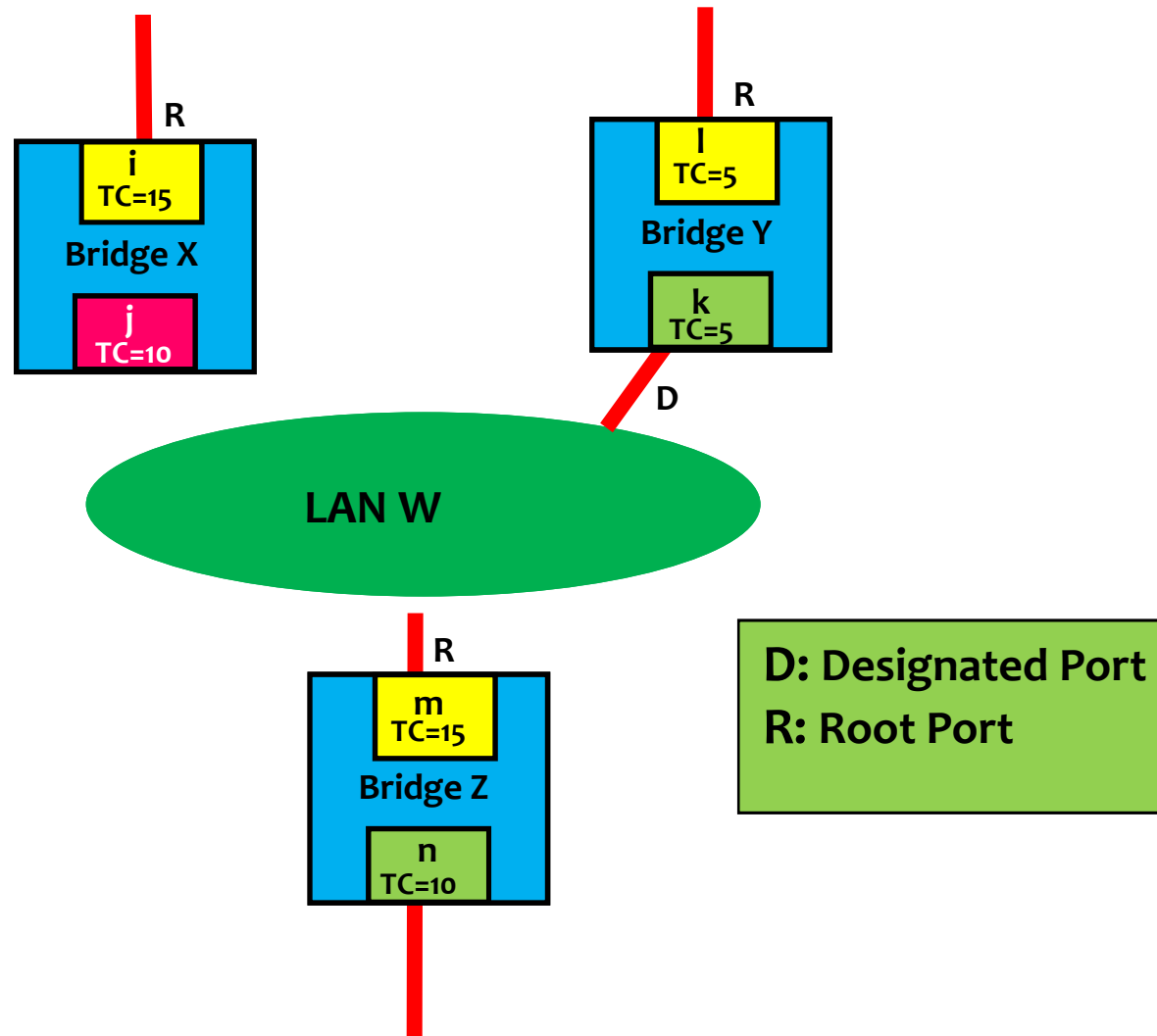


(b) Topology Change BPDU

# Spanning Tree Algorithm Example



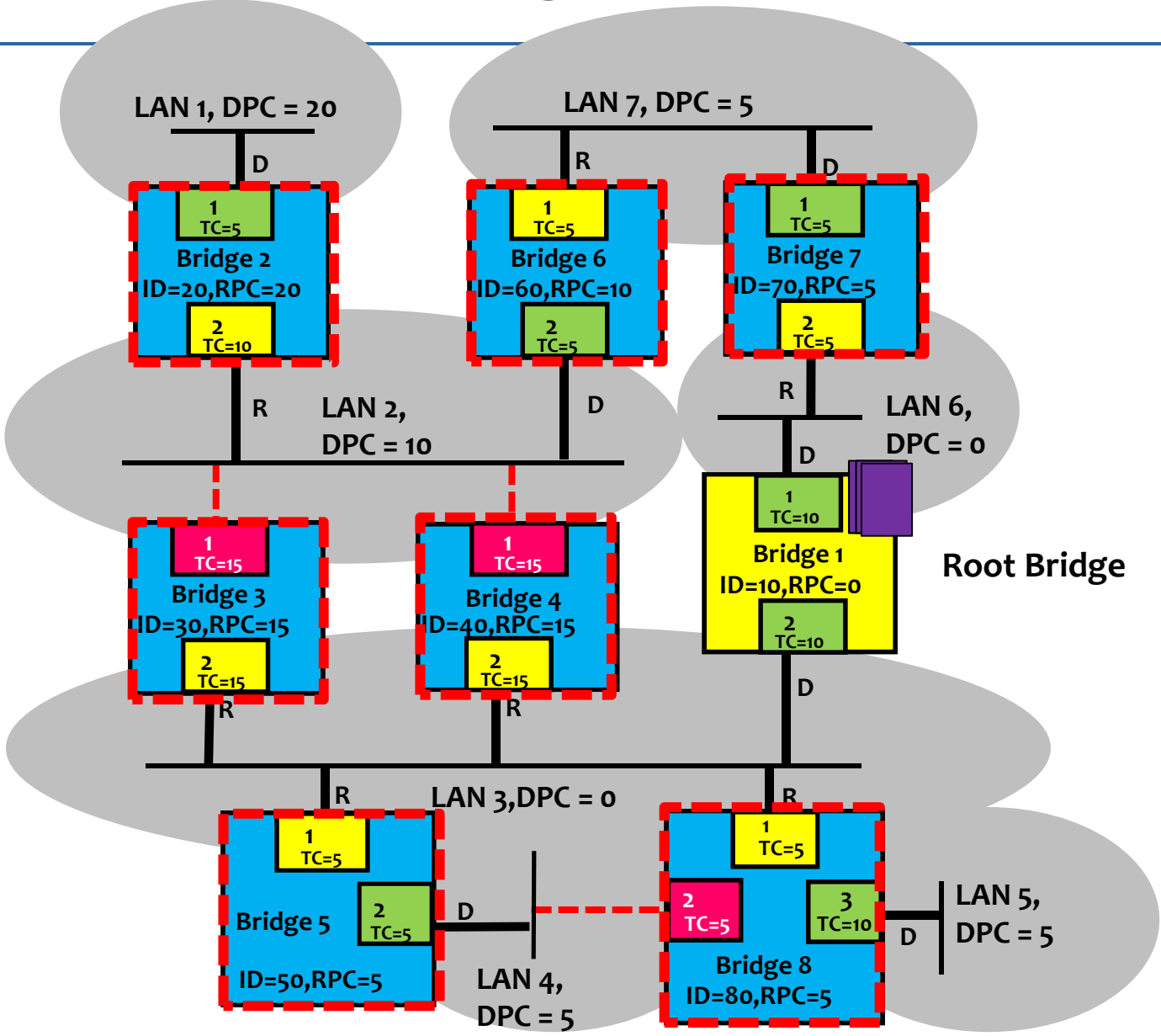
# Spanning Tree Algorithm Example (Continued)



# Spanning Tree Features

---

- The spanning tree constructed by the IEEE 802.1D algorithm has the features that **for each bridge, the shortest path (minimum root path cost, RPC) to the root bridge is included.**
- For each LAN, **the shortest path (minimum root path cost, RPC) to the root bridge via the designated bridge is included.**
- So the spanning tree usually is not a minimum cost spanning tree.
- The spanning tree of a BLAN (or switches connected network) is **predictable or deterministic.**
- Thus, given a BLAN topology (with any loops) and configuration parameters, the spanning tree of the BLAN can be calculated manually.



# Outline

---

- Introduction
- Frames Forwarding and Addresses Learning
- Loop Problem and Resolution
- Spanning Tree Algorithm
- **Spanning Tree Maintenance**

# Spanning Tree Maintenance

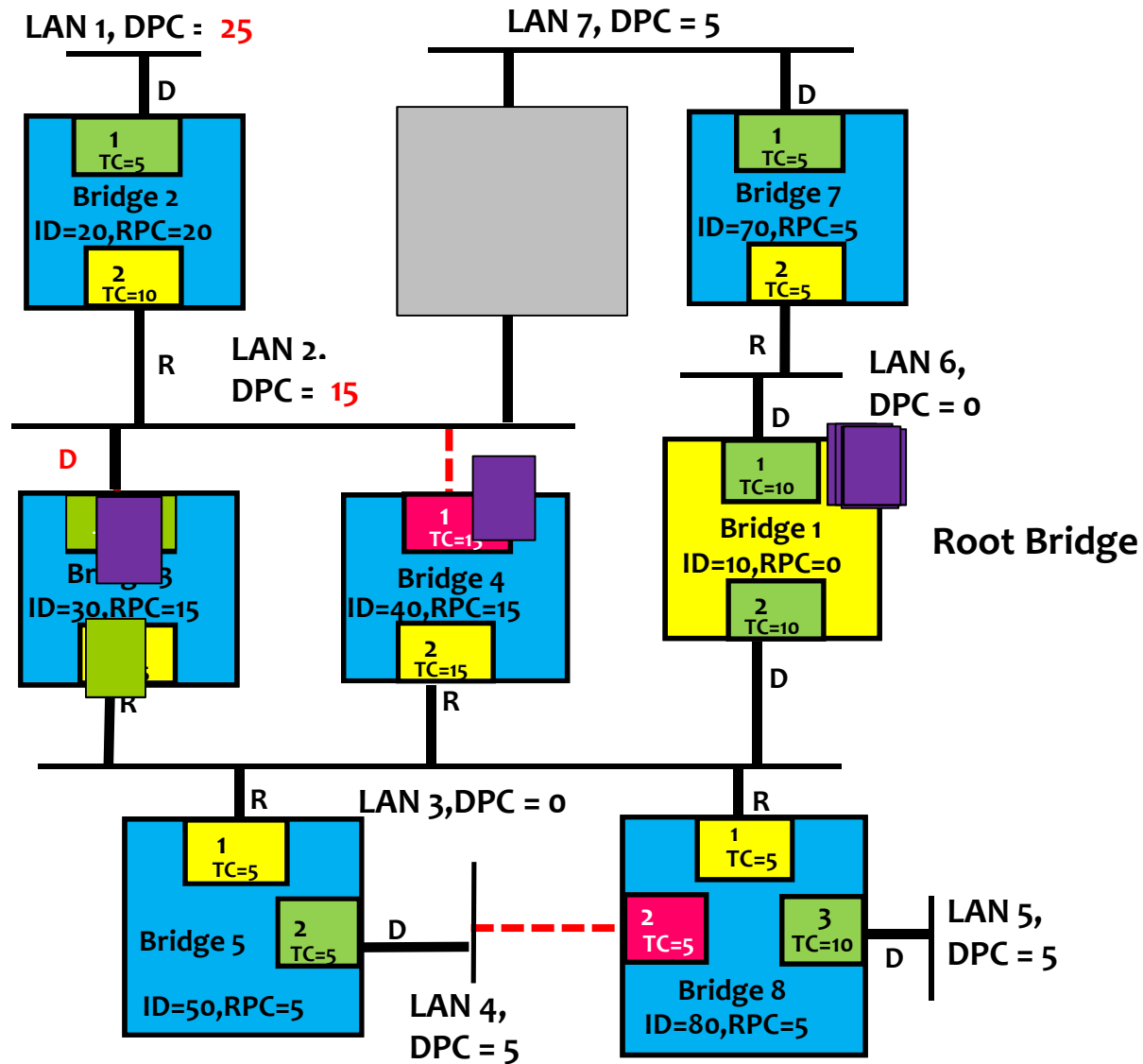
---

- The transmission of the configuration BPDU is triggered by root.
- The root will **periodically** (once every Hello time) issue a **configuration BPDU** on all LANs to which it is attached.
- A bridge that receives a configuration BPDU from its root port passes that information to all LANs for which it believes itself to be the designated bridge.
- A cascade of configuration BPDUs throughout the spanning tree.
- A bridge may change the spanning tree topology
- A **TCN BPDU is reliable relayed** up the new spanning tree to the root bridge (bridge by bridge).
- The root will set the **Topology Change flag** in all configuration messages transmitted for some time.



# Spanning Tree Maintenance

## Example 1 (Bridge Faults)



# Spanning Tree Maintenance

## Example 1 (Bridge Faults)

---

- Assume Bridge 6 (ID = 60) faults.
- Then all the Hello BPDUs sent from root bridge to Bridge 6 will not be forwarded to LAN 2 any more.
- The Bridges 3 and 4 in LAN 2 will trigger the timeout event individually which means the Designated bridge 6 for LAN 2 was gone.
- Then they will try to serve as the Designated bridge of LAN 2 by forwarding a configuration BPDU.
- Assume bridge 4 sends the BPDU first with a RPC = 15.

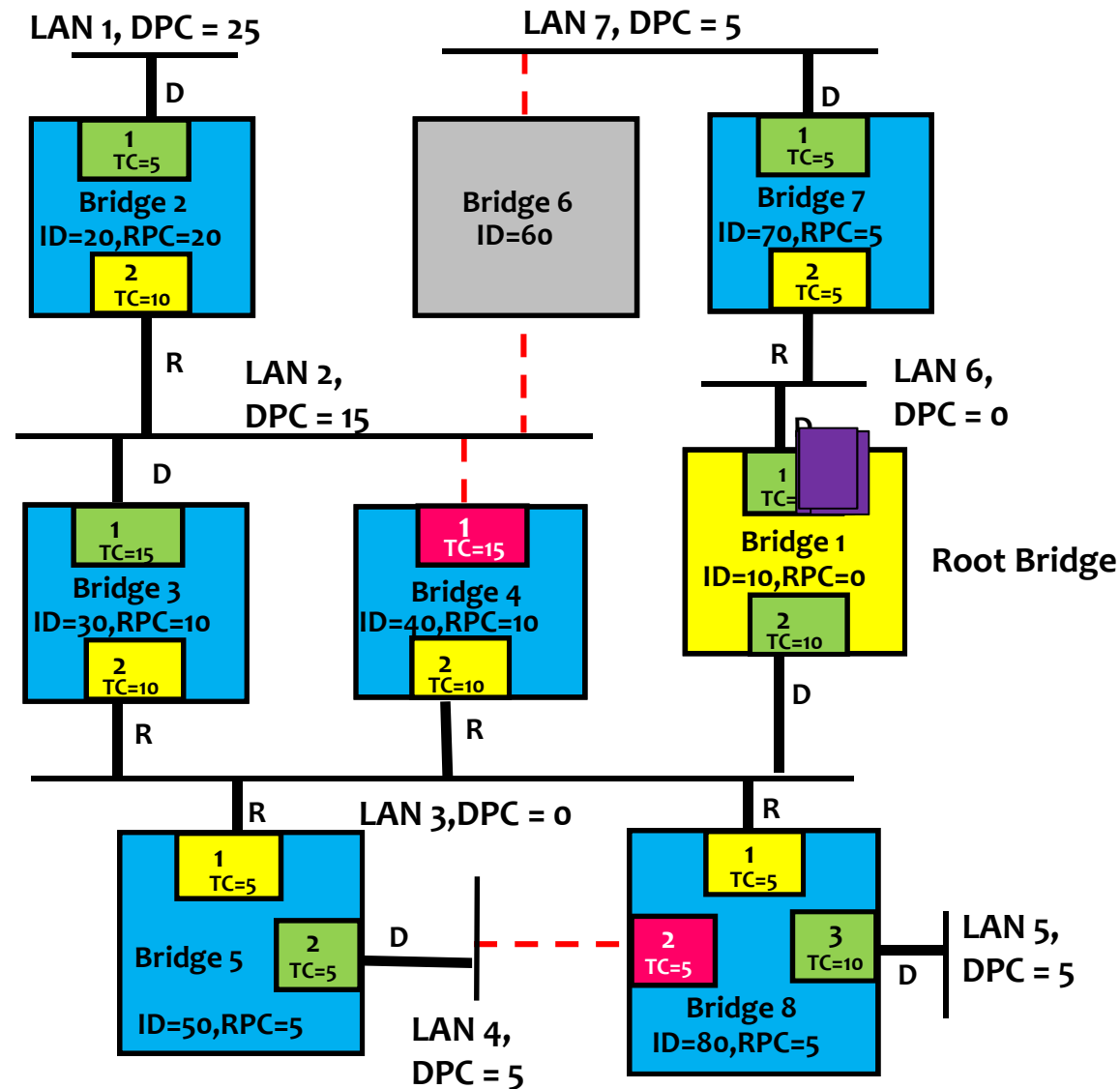
# Spanning Tree Maintenance

## Example 1 (Bridge Faults)

---

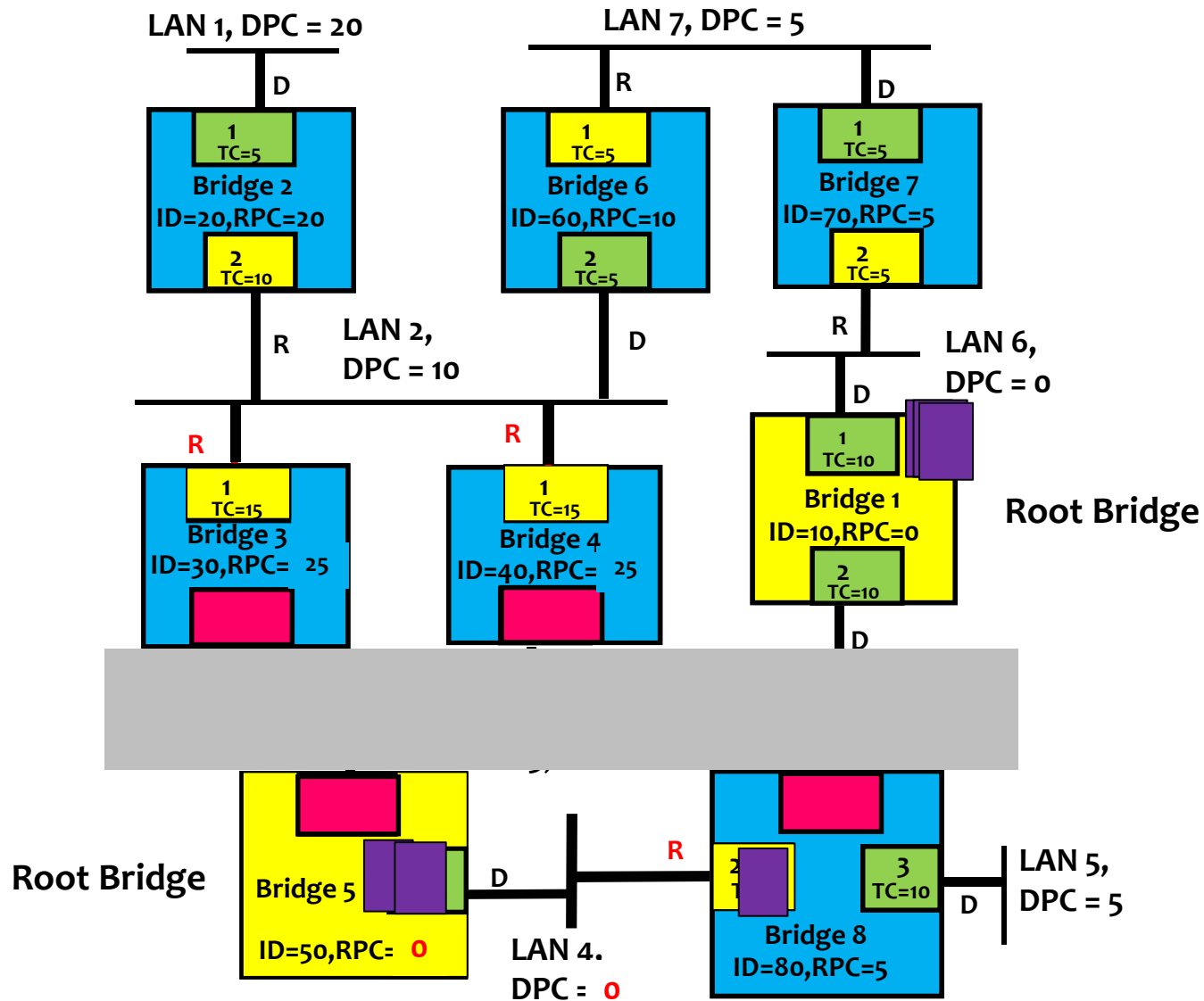
- Then bridge 3 will return another BPDU with RPC=15 since it's priority is higher than bridge 4 (same RPC, smaller ID).
- After two forwarding delays, bridge 3 will become the new Designated bridge of LAN2 and the DPC becomes 15.
- Also the DPC of LAN 1 is changed from 20 to 25.
- Bridge 3 then sends a Topology Change Notification (TCN) BPDU to root bridge.
- The root will set the Topology Change flag in all configuration messages transmitted for some time.

# Final Configuration of Example 1



# Spanning Tree Maintenance

## Example 2 (LAN Faults)



# Spanning Tree Maintenance

## Example 2 (LAN Faults)

---

- Assume LAN 3 faults.
- Then all the Hello BPDUs sent from root bridge to LAN 3 will be lost.
- All the ports connected to LAN 3, including port 2 of bridge 3, port 2 of bridge 4, port 1 of bridge 5, and port 1 of bridge 8, will become **“blocked” state** from “forwarding” state.
- All these bridges are now don't have “R” port (root port) and then try to be a root bridge.
- Bridges 3 and 4 still can receive the Hello BPDU from port 1, so they will change their root port to port 1.

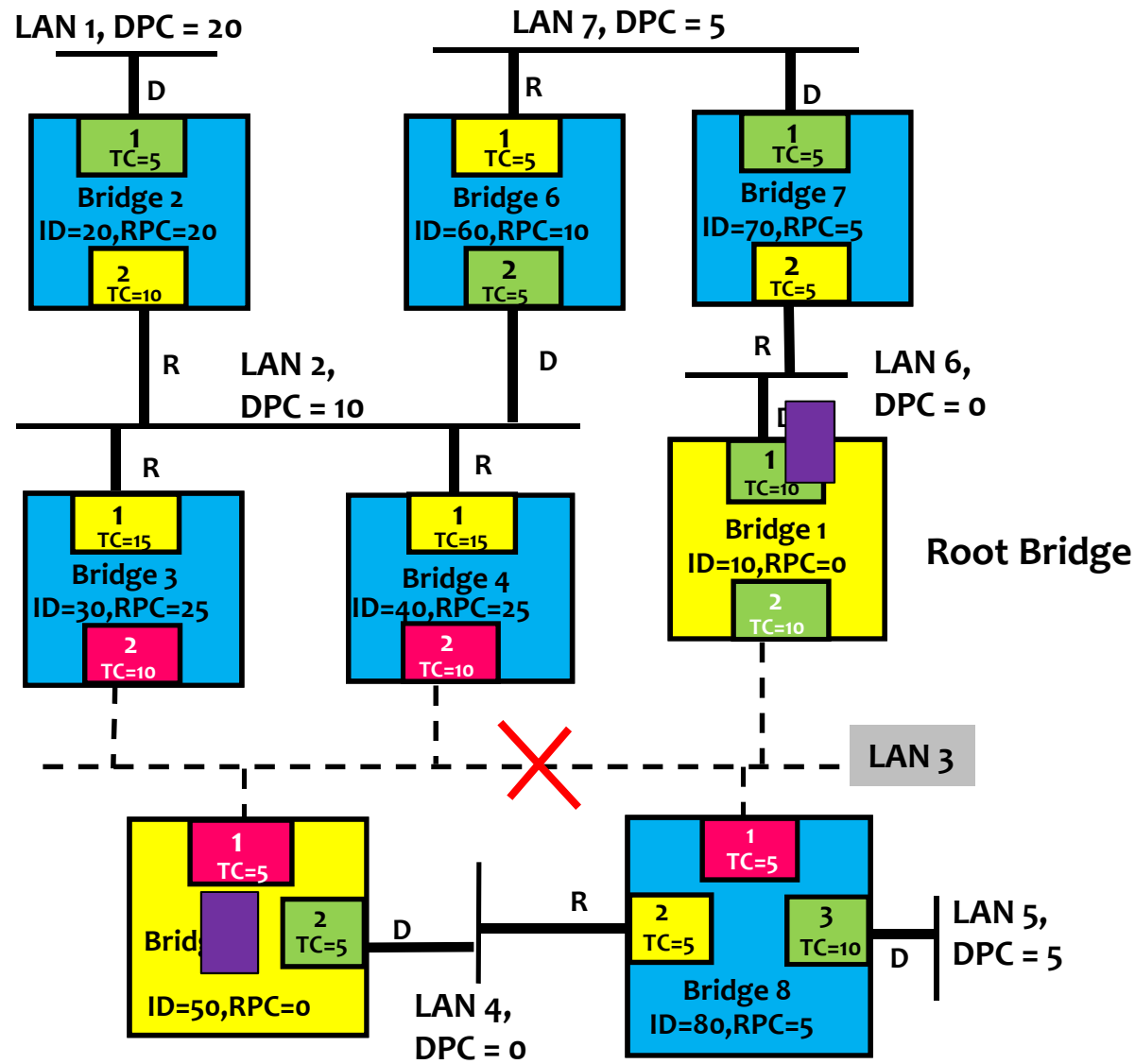
# Spanning Tree Maintenance

## Example 2 (LAN Faults)

---

- Bridges 5 and 8 will exchange BPDU to compete as a new root follow the STP protocol.
- Assume bridge 8 sends the BPDU first with a  $RPC = 0$ .
- Then bridge 5 will return another BPDU with  $RPC=0$  since it's priority is higher than bridge 8 (smaller ID).
- After two forwarding delays, bridge 5 will become the new root bridge and the port 1 of bridge 8 will become a root port.
- Finally, we have two separated (disconnected) spanning trees.

# Final Configuration of Example 2





# In Summary

---

- A bridge is a **layer 2 device** which relays frames among physically separated LANs and makes the physical LANs appear as **one logical LAN** to the end stations
- Basic functions of a bridge:
  - Frame **Forwarding** and **Filtering**
  - Address **Learning**
  - Resolving Possible **Loops** in the Topology
- The spanning tree constructed by the IEEE 802.1D algorithm has the features that **for each bridge, the shortest path (minimum root path cost, RPC) to the root bridge is included.**

# In Summary

---

- For each LAN, the shortest path (minimum root path cost, RPC) to the root bridge via the designated bridge is included.
- The spanning tree of a BLAN (or switches connected network) is predictable or deterministic.
- Thus, given a BLAN topology (with any loops), the spanning tree of the BLAN can be calculated manually.
- The spanning tree algorithm has the ability to maintain the spanning tree by handling the bridge faults as well as LAN faults.