Introduction to Computer Networks

IEEE 802.1D Spanning Tree Algorithm
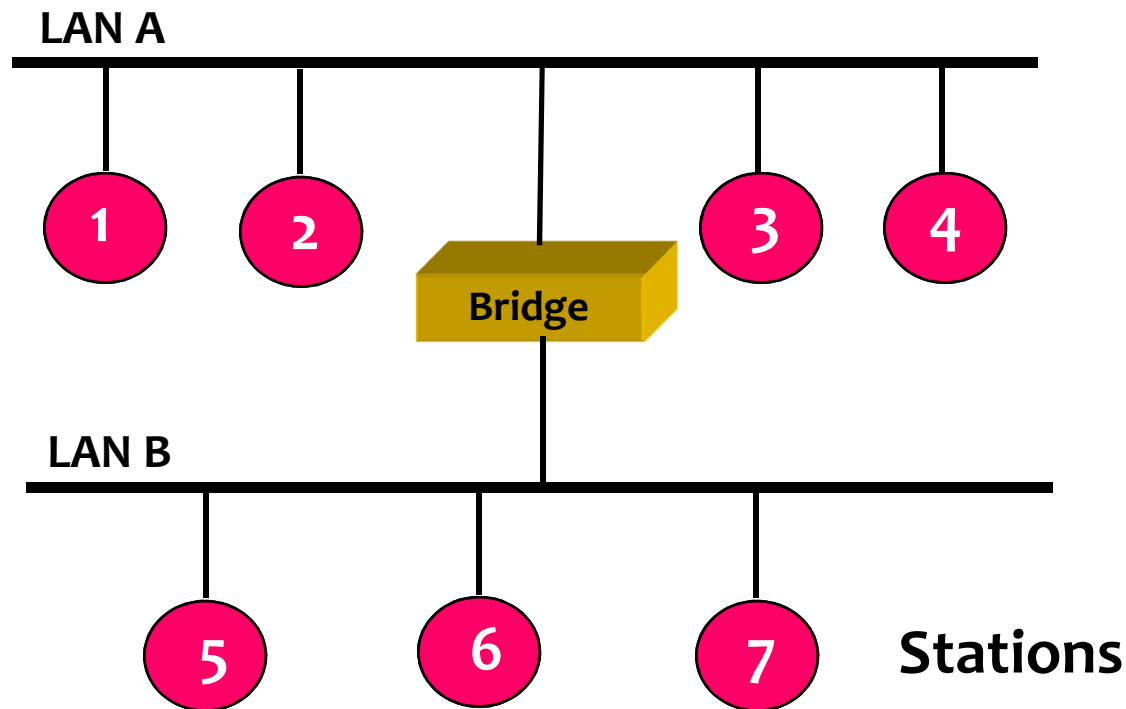
Outline

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

A Simple Bridge Example
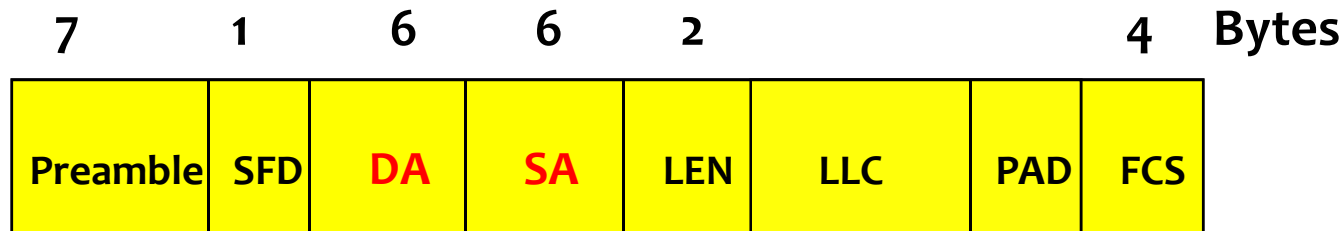
What is a Bridge ?

■ **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**

7	1	6	6	2			4	Bytes
Preamble	SFD	DA	SA	LEN	LLC	PAD	FCS	

The Concept of One Logical LAN

LAN A

1 2 **Bridge** 3 4

One Logical LAN

LAN B

5 6 7 **Stations**

Preamble	SFD	DA	SA	LEN	LLC	PAD	FCS

MAC_3 MAC_4
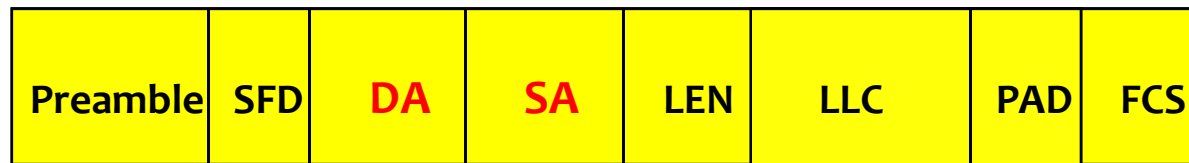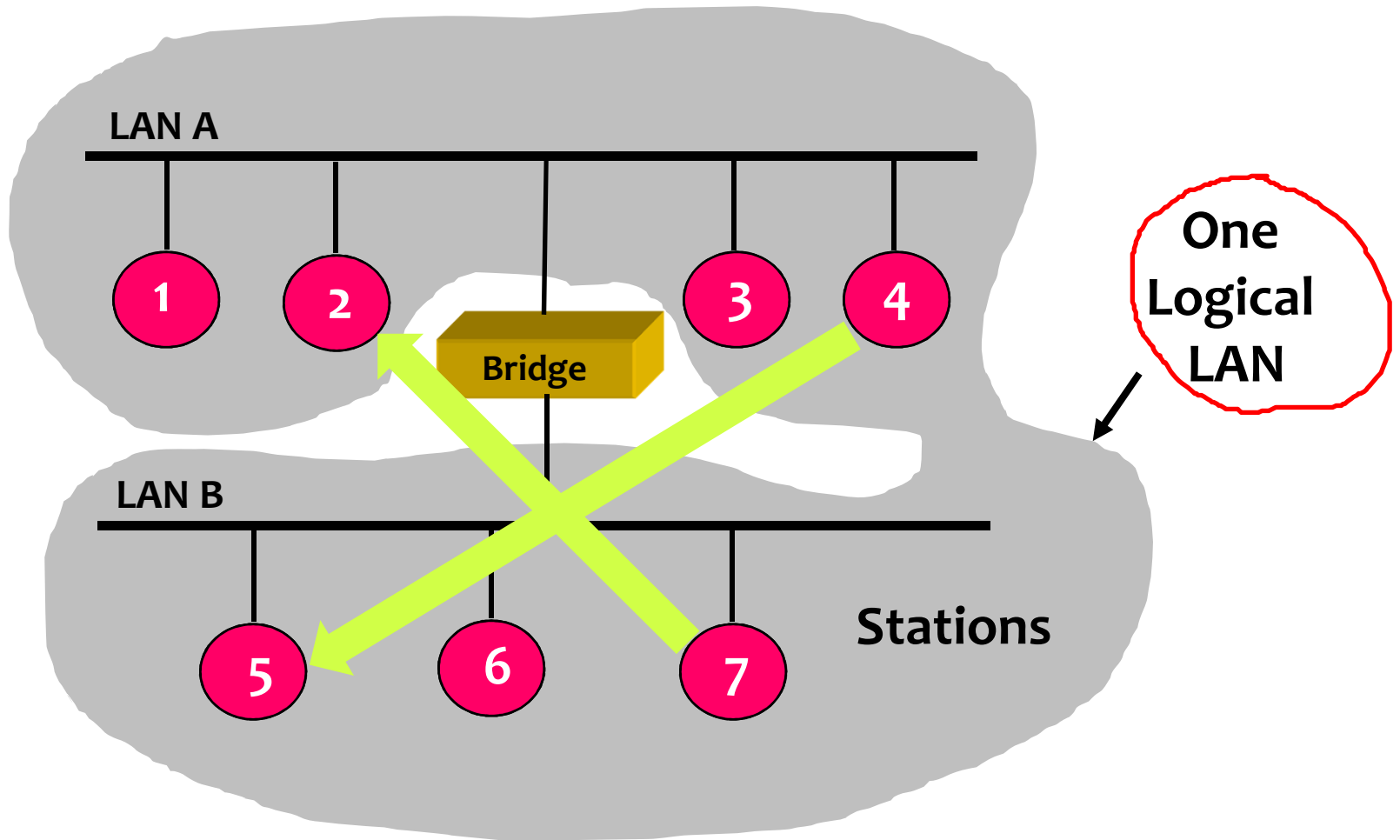
Functions of a Bridge

■ 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**
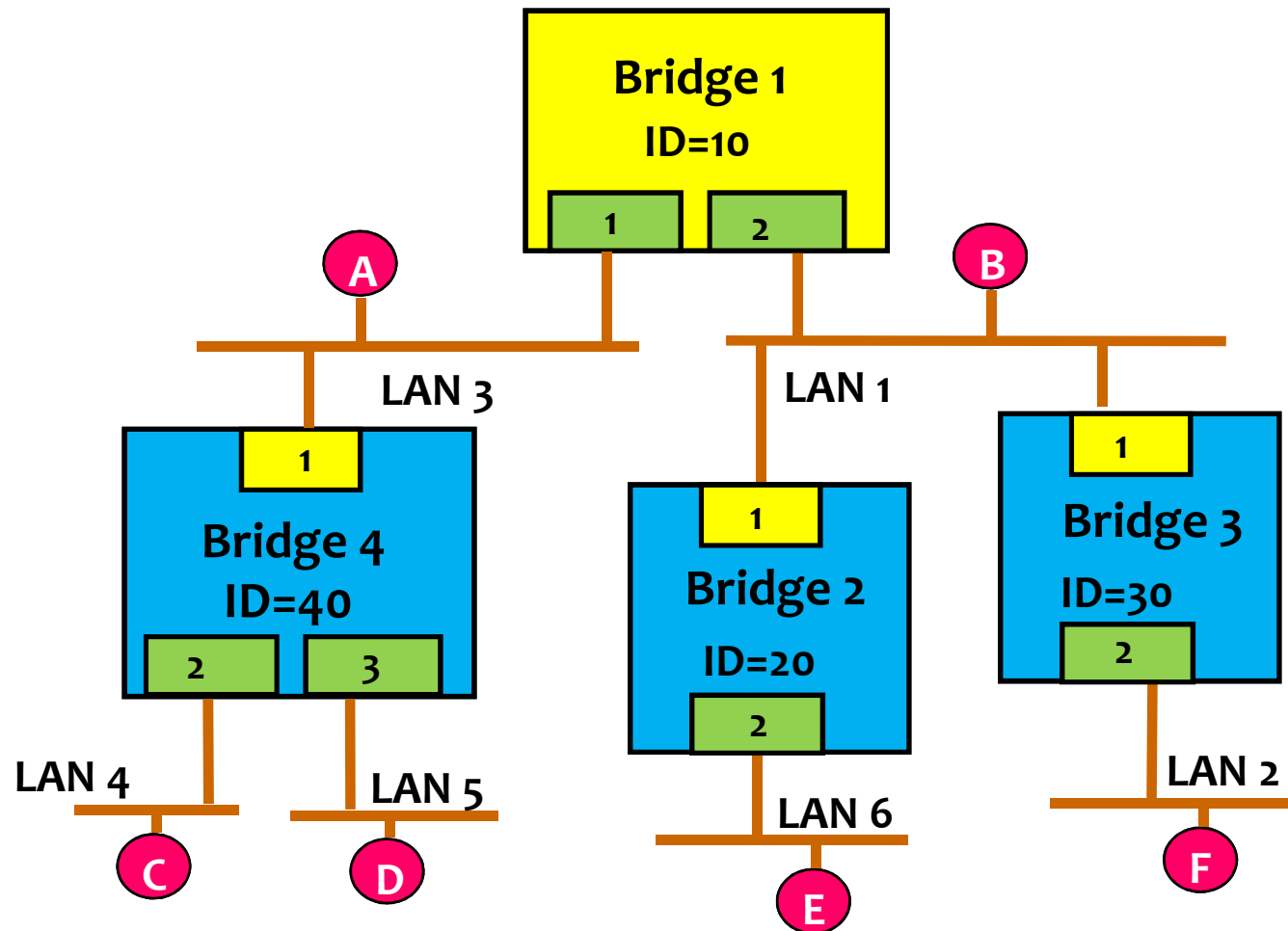 - **Security**
  - **Geography**

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
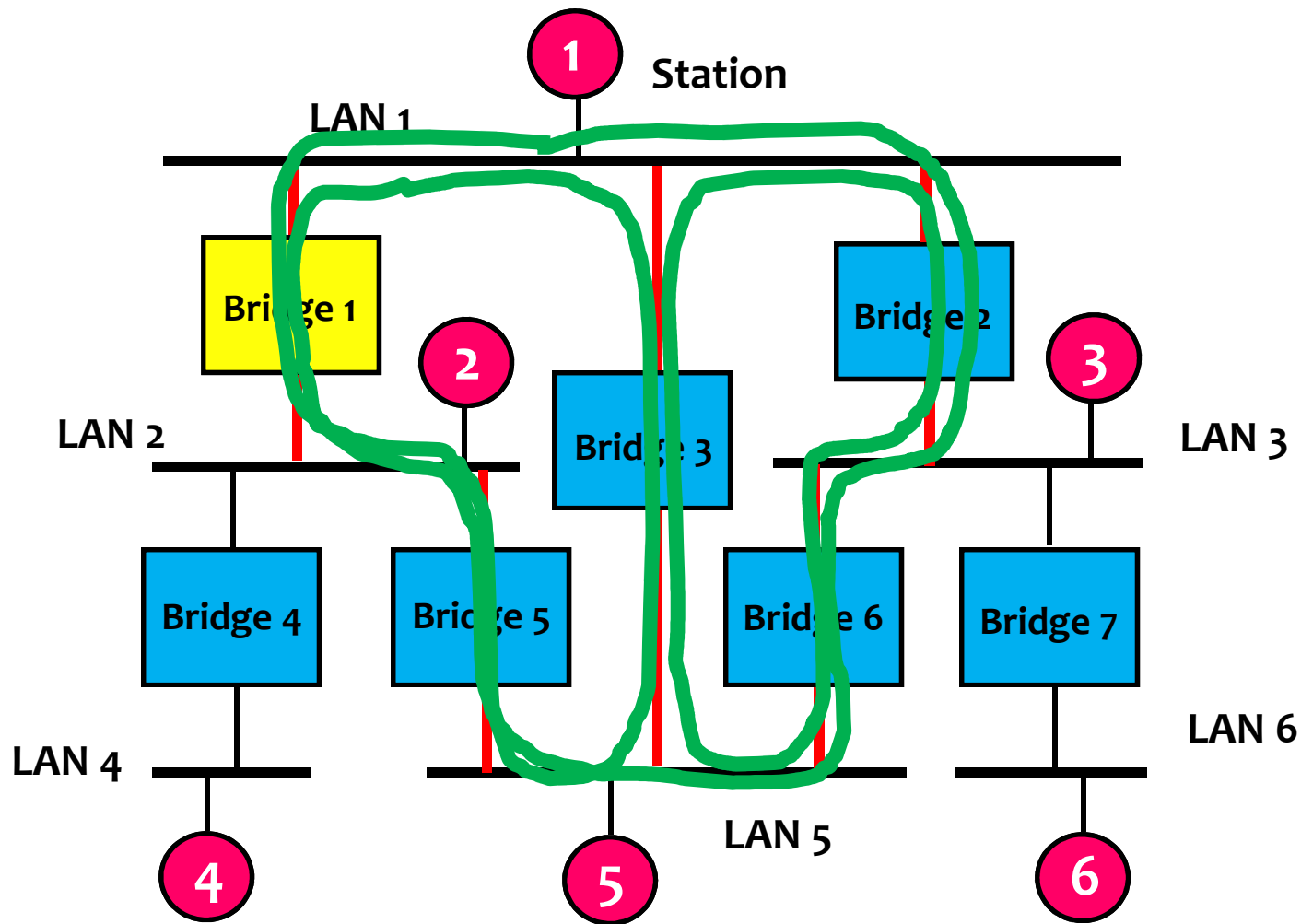
Bridge Protocol Architecture
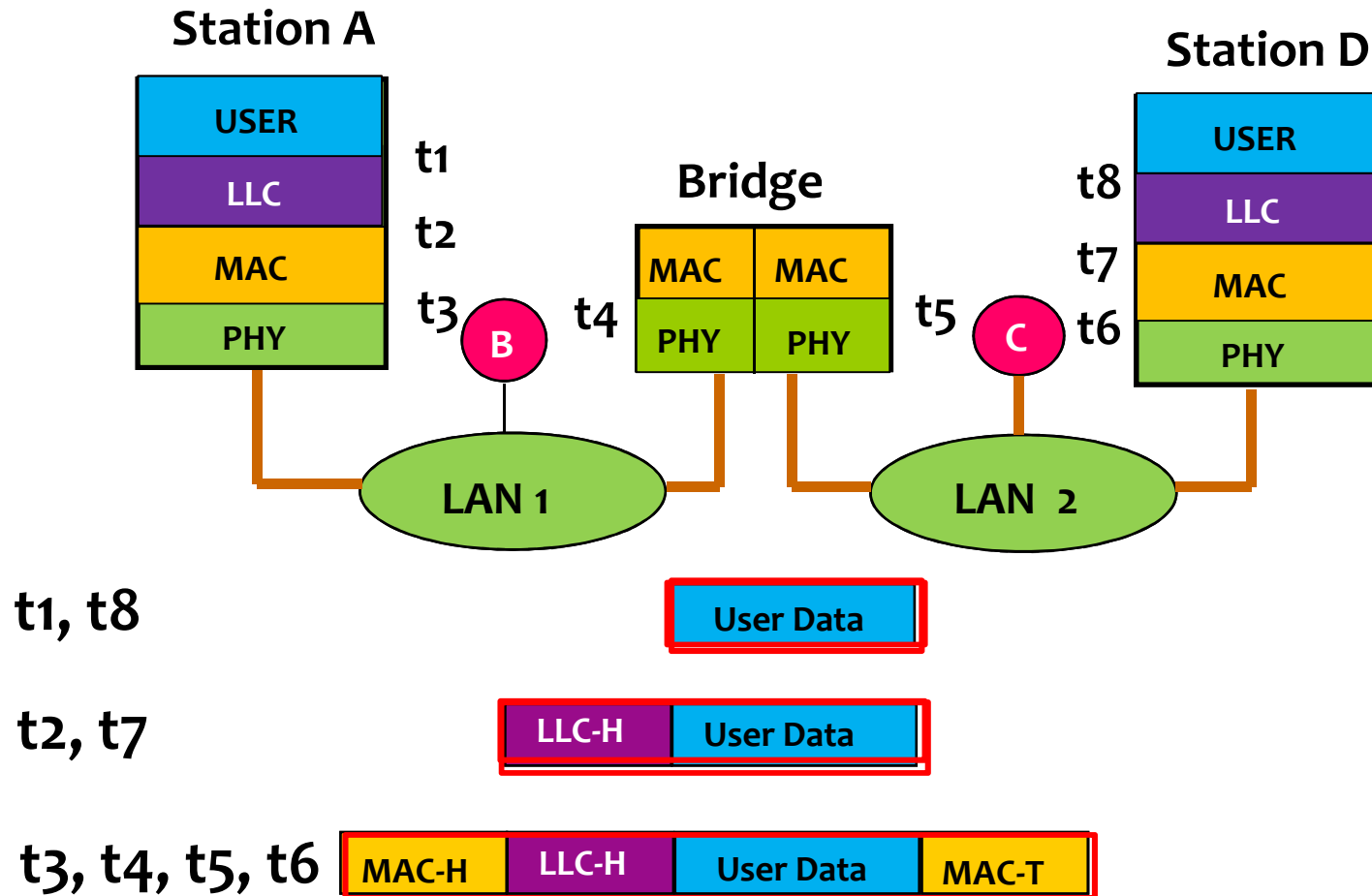
Spanning Tree Routing

■ **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**

- **If the element is not in the database, a new entry is created with its own timer**

7	1	6	6	2		4	Bytes
Preamble	SFD	DMAC	SMAC	LEN	LLC	PAD	FCS
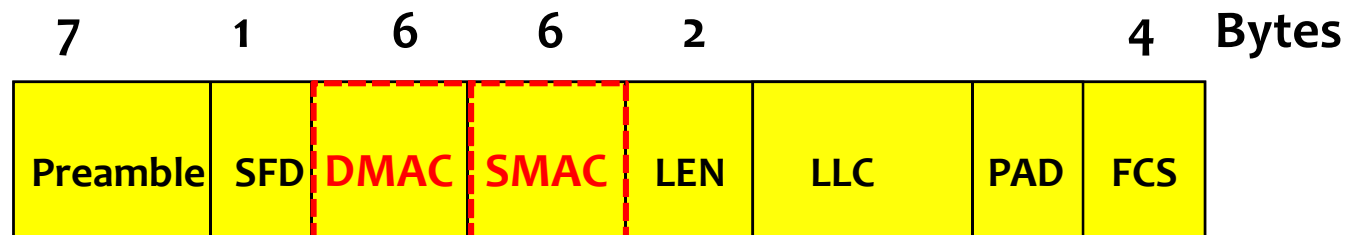
Filtering Database Examples
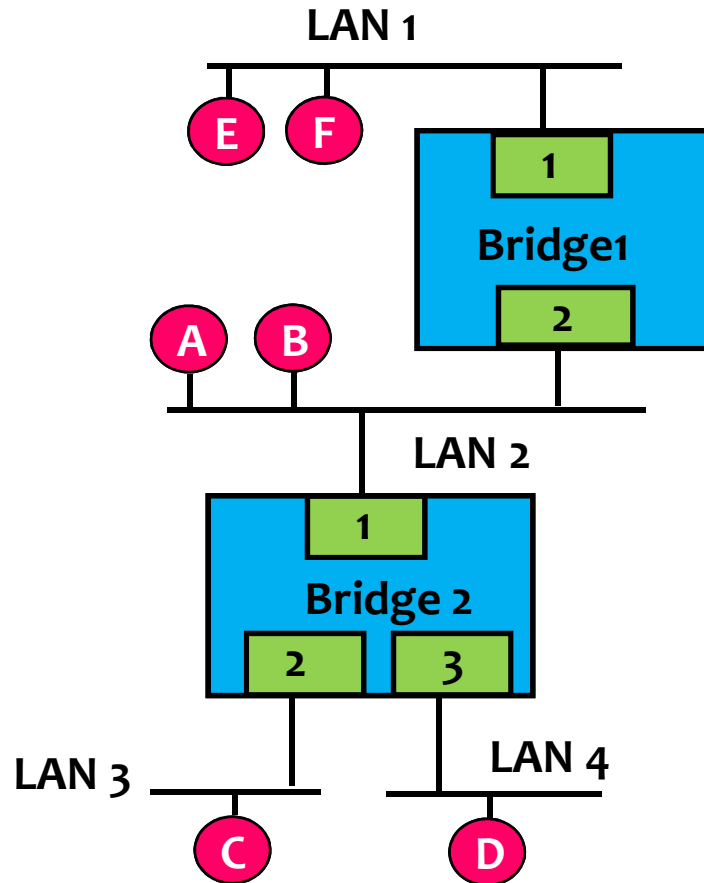
Filtering Database (Bridge 1)

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

LAN 1

Bridge1

1

2

A B

LAN 2

Bridge 2

1

2 3

LAN 3

LAN 4

C D

E F

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

Frame Forwarding

Frame from Port x

DMAC in FDB ?

N → Forward to all ports（except port x）

Y → Belong to Port x ?

Y → Filter

N → Forward to belonging Port

Address Learning

SMAC in FDB ?

Y → Change to port X, reset timer

N → Add SMAC, port (x) and Timer (0) into FDB

End
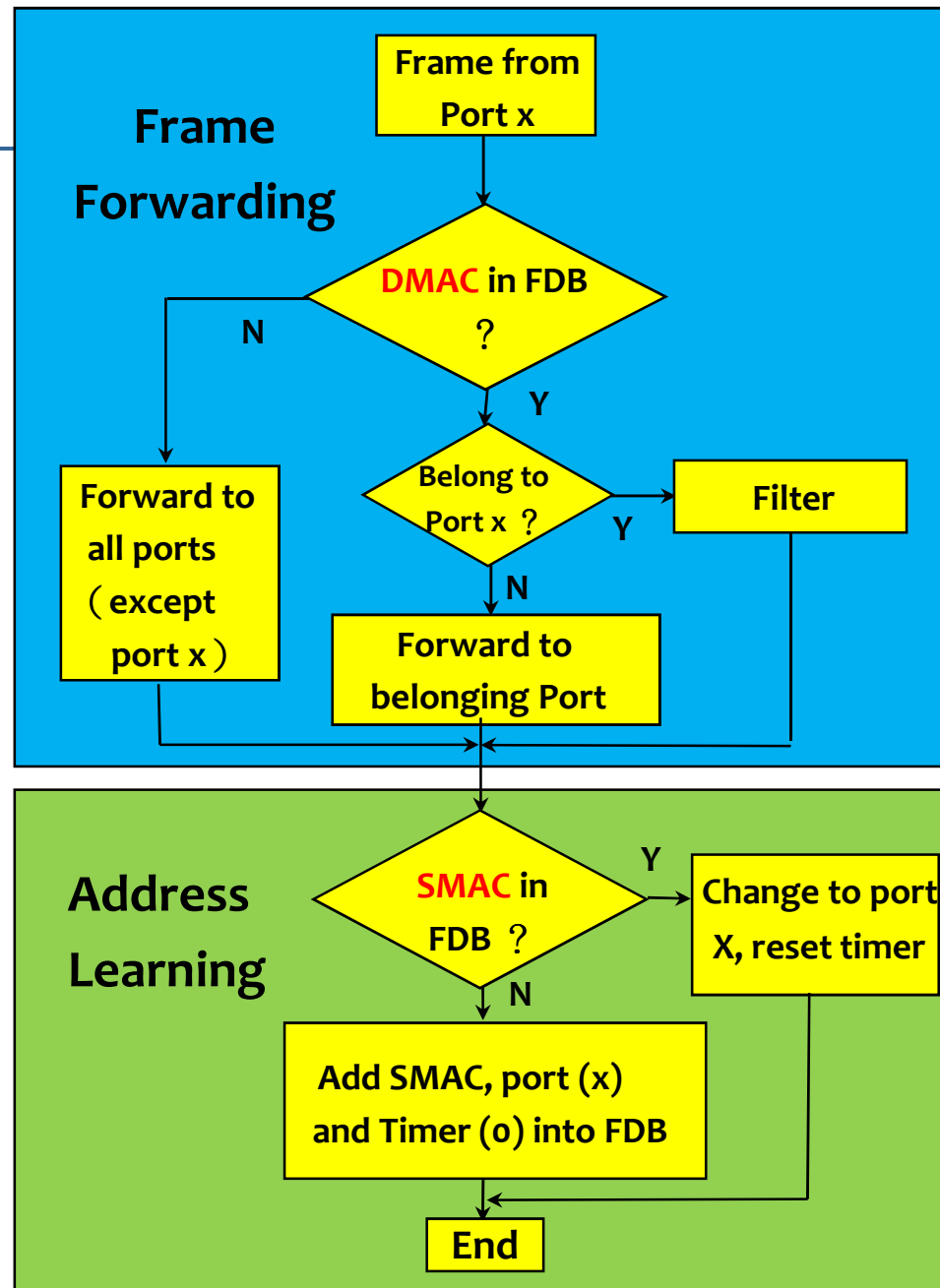
Addresses Learning Example

1. A -> E

2. B -> D

3. C -> B

4. D -> A

5. E -> C
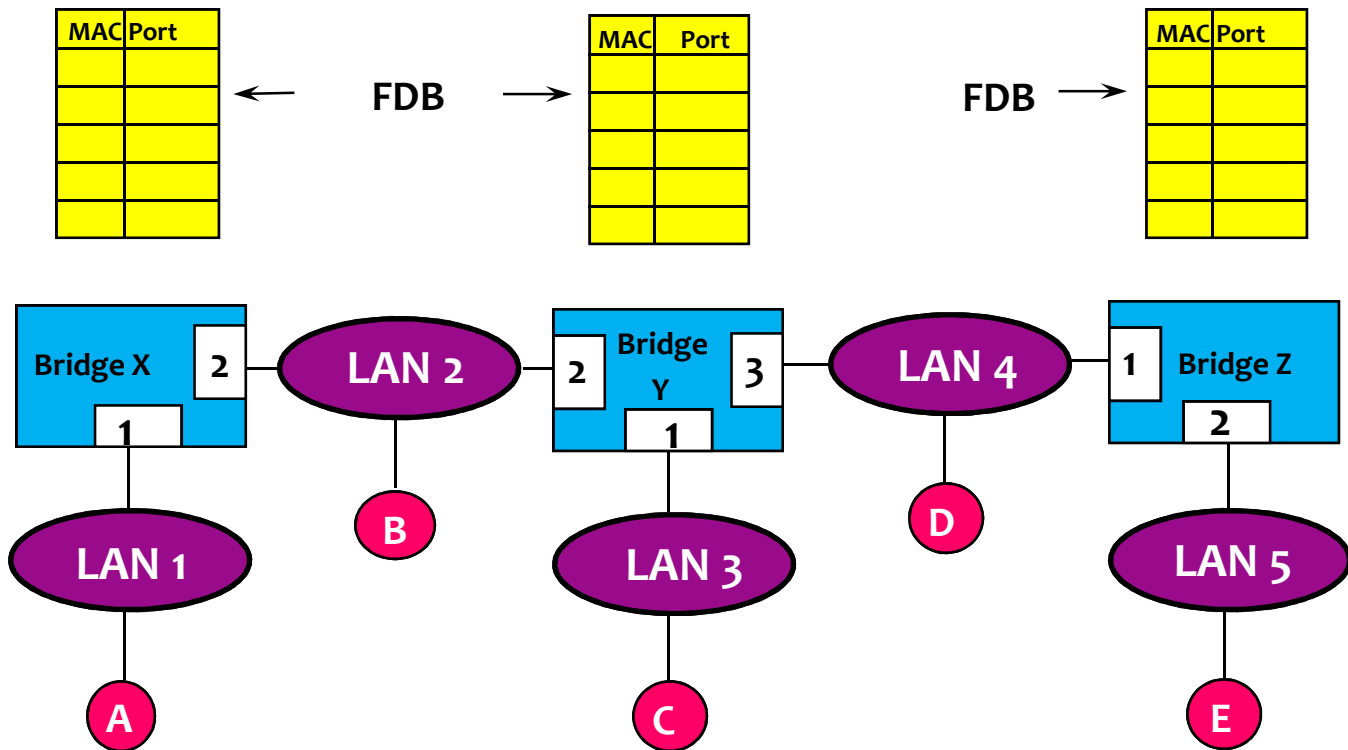
Addresses Learning Example (A→E)
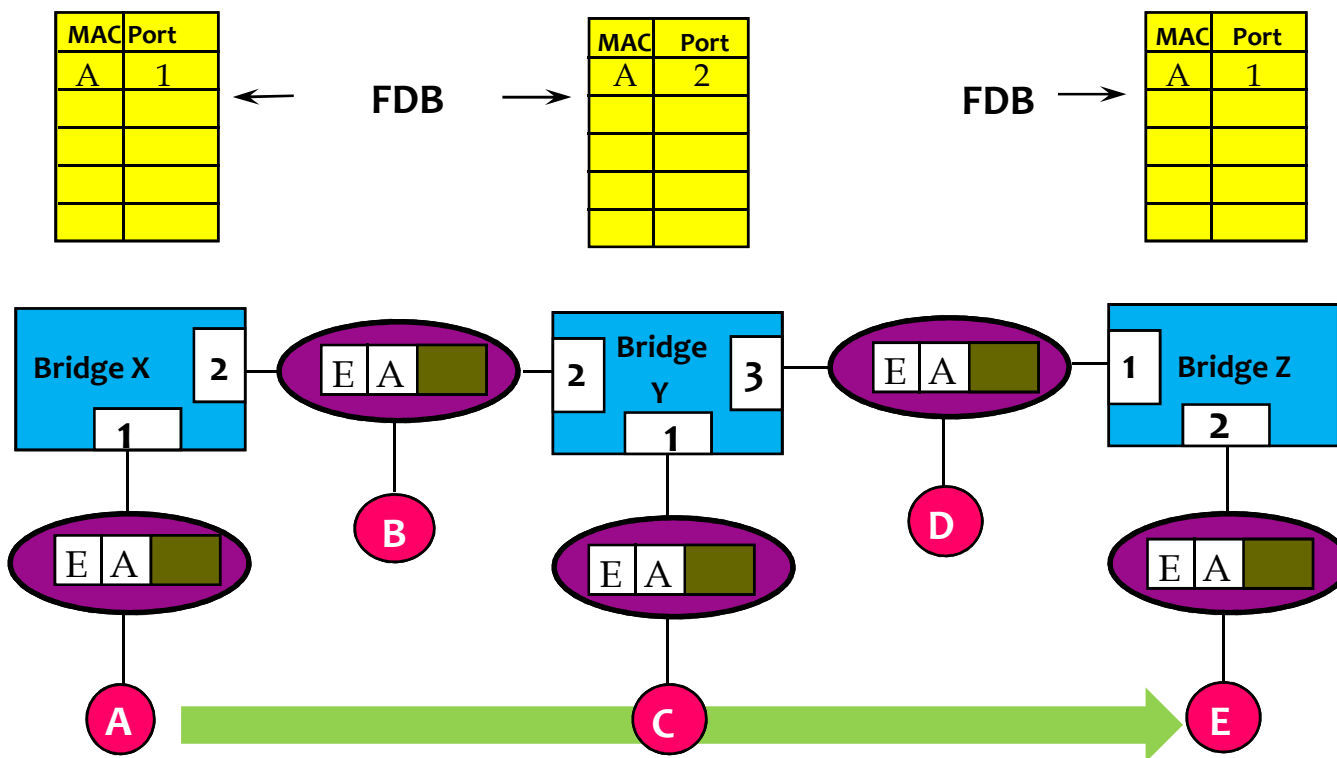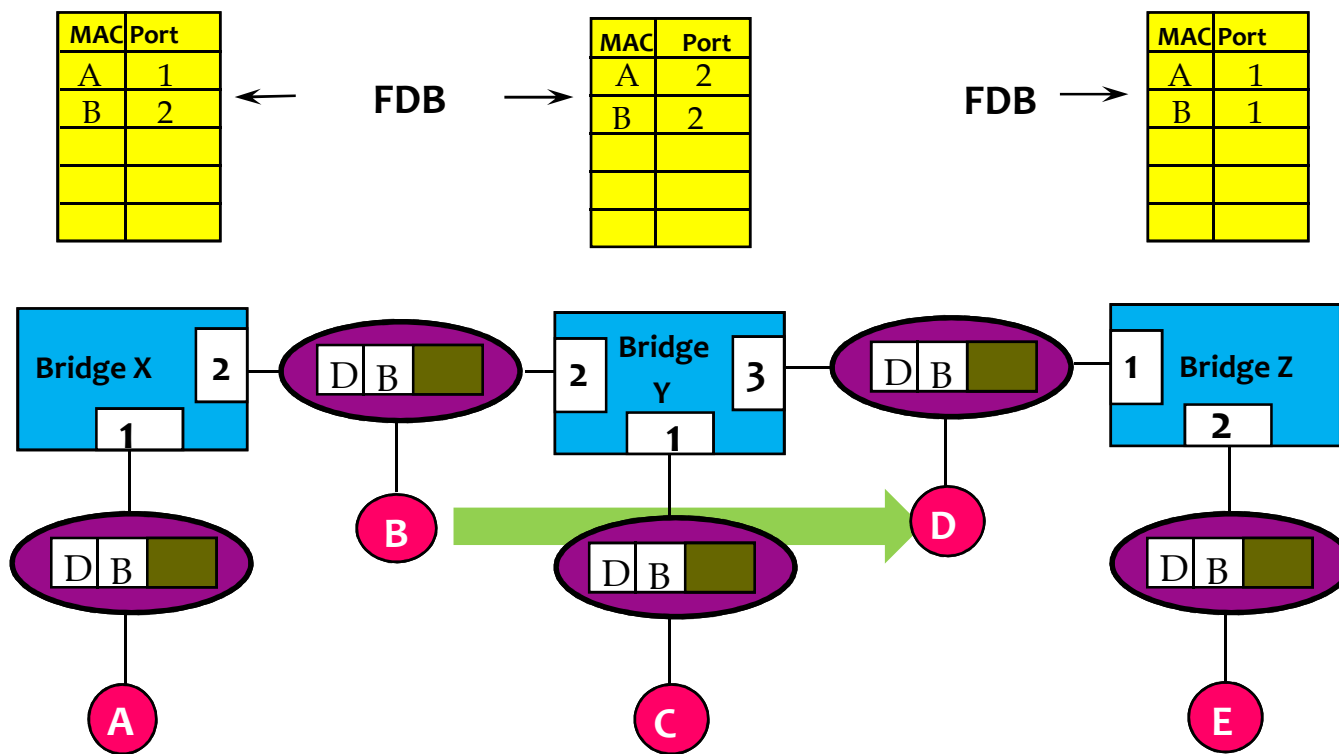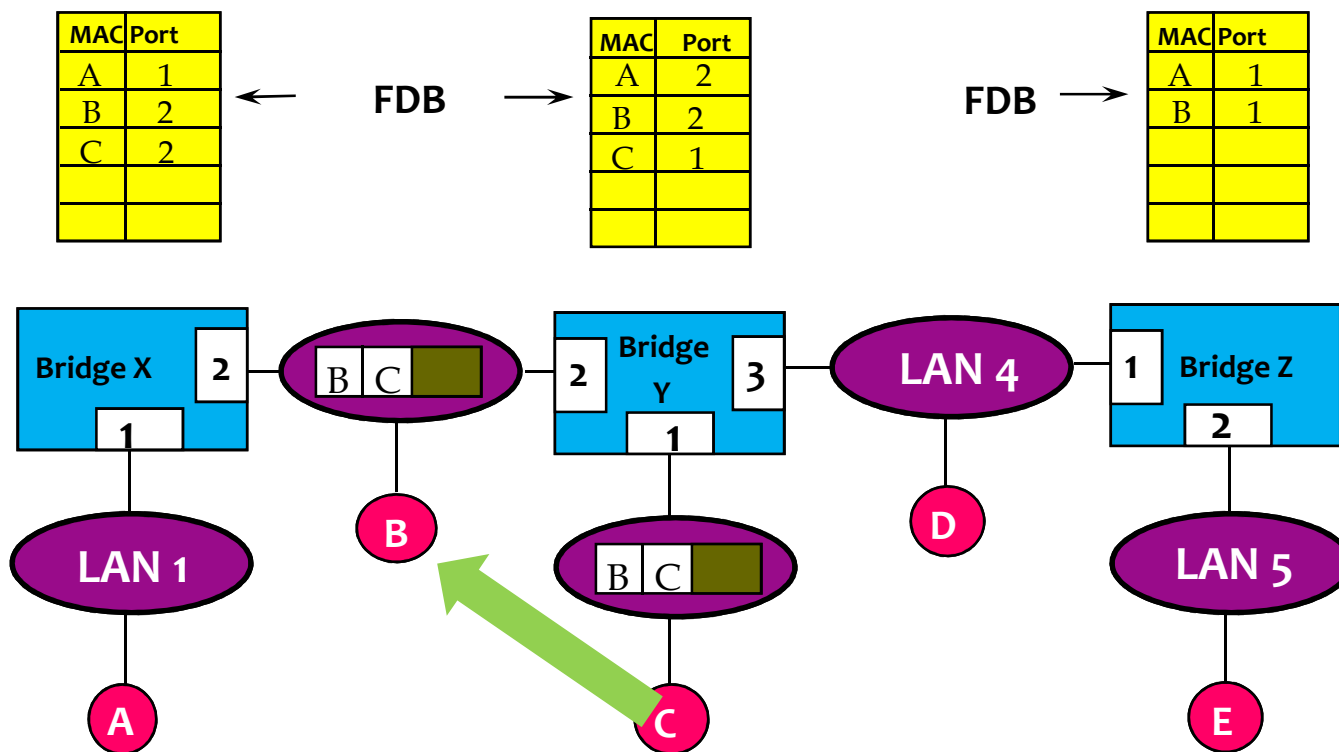
MAC	Port
A	1

← FDB →

MAC	Port
A	2

FDB →

MAC	Port
A	1

Bridge X 2 — E A — 2 **Bridge Y** 3 — E A — 1 **Bridge Z**

1

B

1

2

E A

A

E A

C

D

E A

E

Addresses Learning Example (B→D)

MAC	Port
A	1
B	2

← **FDB** →

MAC	Port
A	2
B	2

FDB →

MAC	Port
A	1
B	1

Bridge X 2
1

D B

Bridge Y 2 3
1

D B

Bridge Z 1
2

D B

D B

B → D

D B

D B

A

C

E

Addresses Learning Example (C→B)

MAC	Port
A	1
B	2
C	2

FDB

MAC	Port
A	2
B	2
C	1

FDB

MAC	Port
A	1
B	1

Bridge X — 2 — B C — 2 — Bridge Y — 3 — LAN 4 — 1 — Bridge Z

1 — LAN 1 — A

B

1 — B C

D

2 — LAN 5 — E

C

Addresses Learning Example (D→A)

MAC	Port
A	1
B	2
C	2
D	2

← **FDB** →

MAC	Port
A	2
B	2
C	1
D	3

FDB →

MAC	Port
A	1
B	1
D	1

Bridge X — 2 — | A | D | | — 2 — Bridge Y — 3 — | A | D | | — 1 — Bridge Z

1 — | A | D | |

B

LAN 3

D

1 — 2

LAN 5

A — C — E

Addresses Learning Example (E→C)

MAC	Port
A	1
B	2
C	2
D	2

← **FDB** →

MAC	Port
A	2
B	2
C	1
D	3
E	3

FDB →

MAC	Port
A	1
B	1
D	1
E	2

Bridge X 2

LAN 2 2 Bridge Y 3 C E

1 Bridge Z 2

1

LAN 1

B

C E

D

C E

A

C ← E
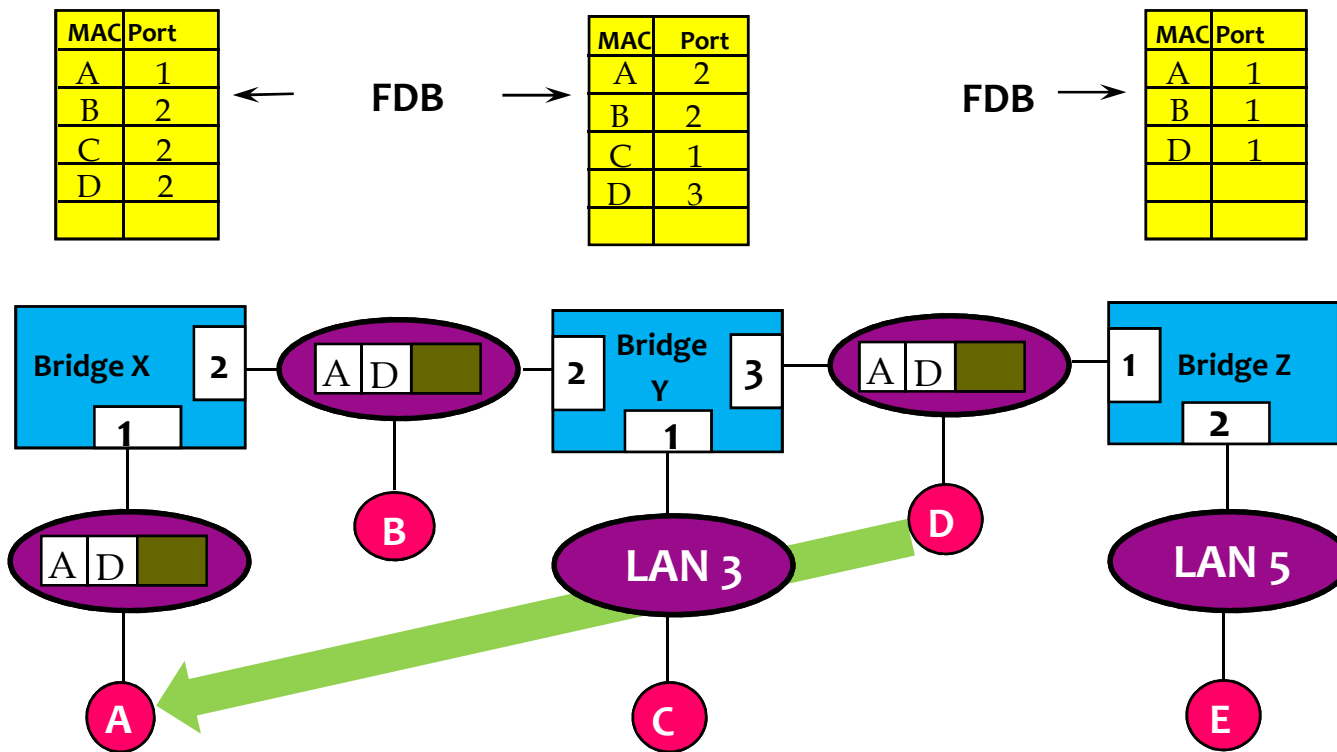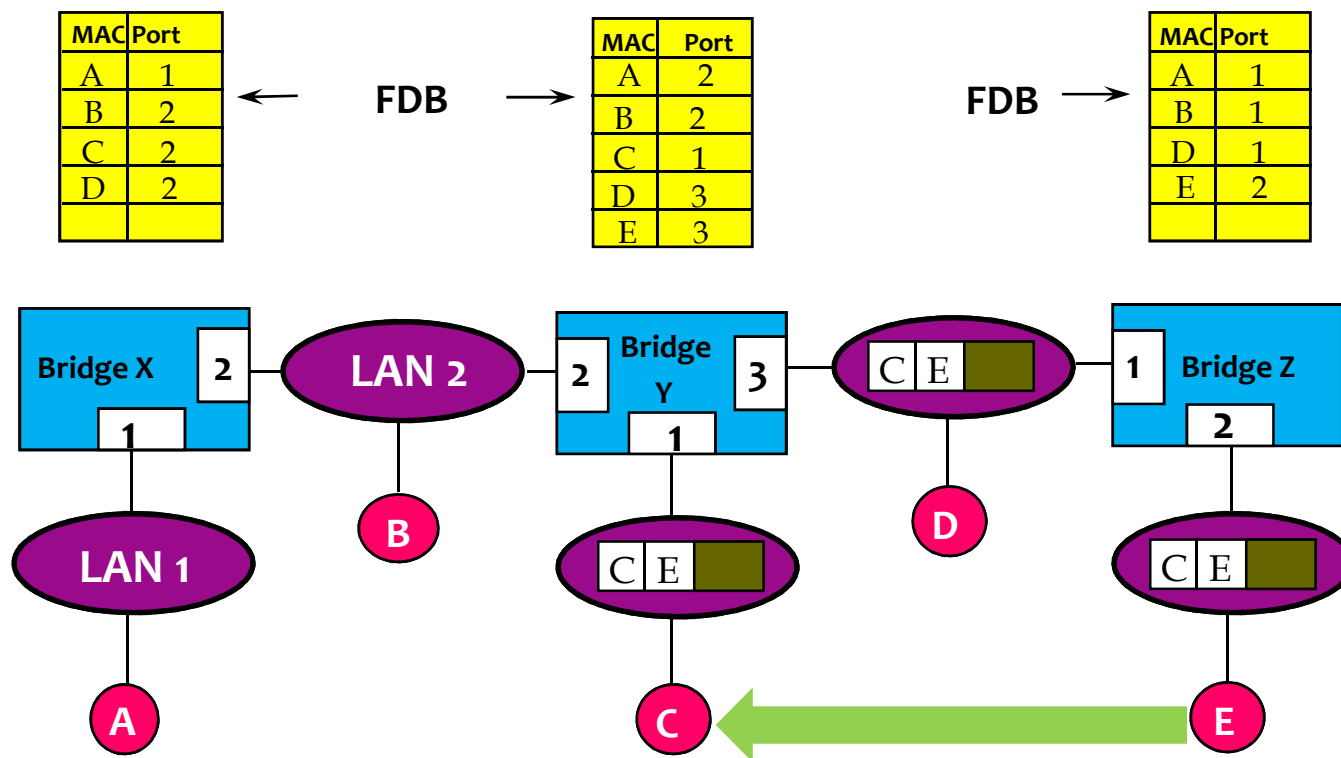
Outline

■ **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

LAN

Spanning Tree

Bridge
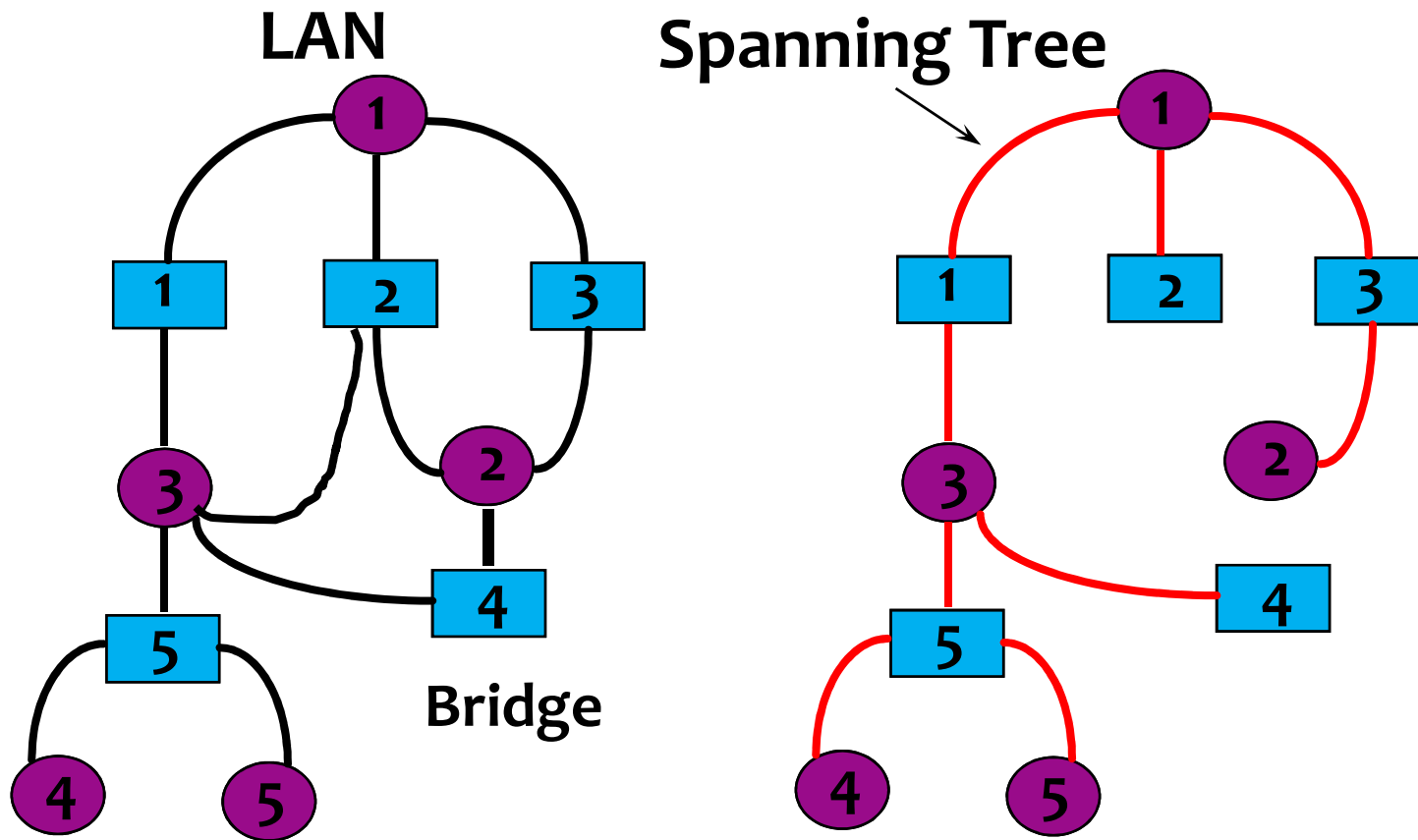
Spanning Tree Example 1
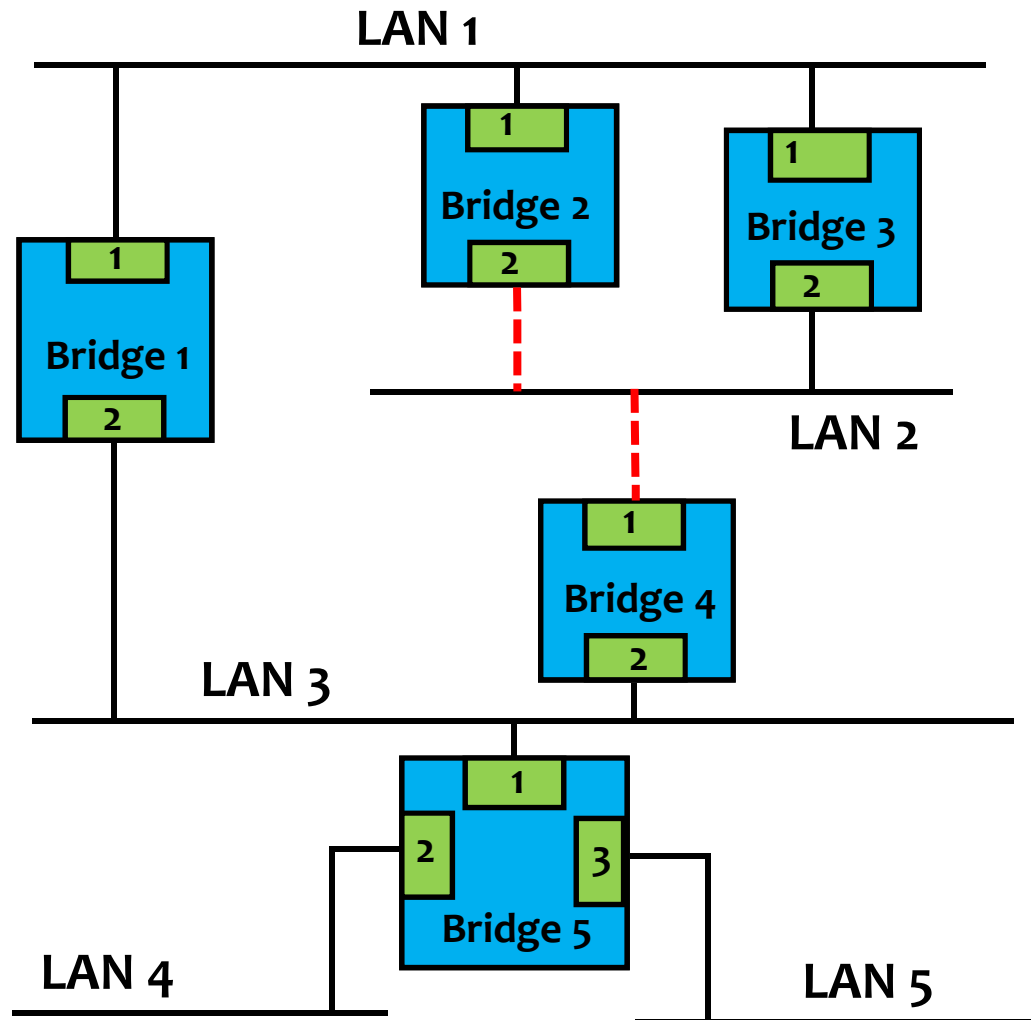
Spanning Tree Example 1 (Continued)
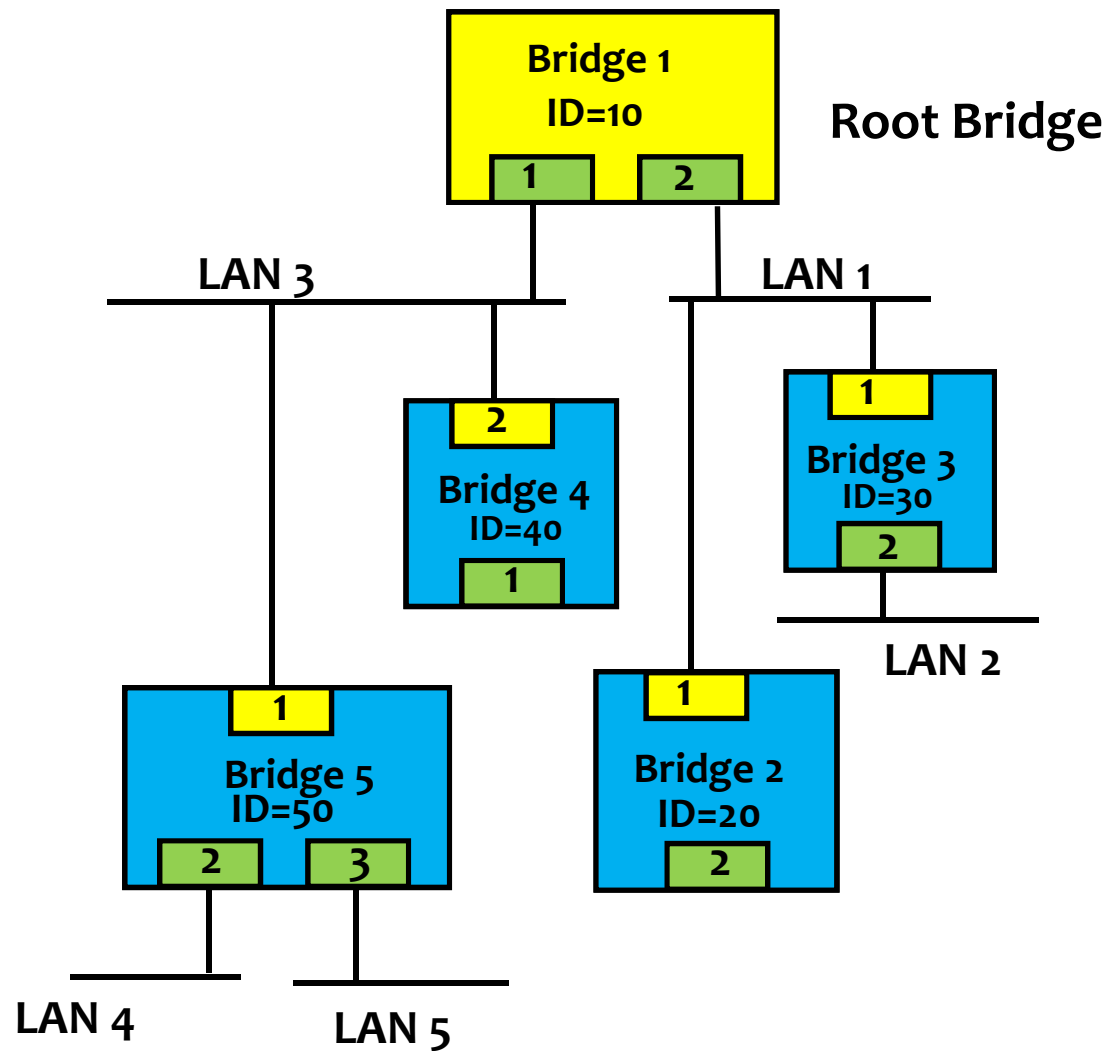
Outline

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

Spanning Tree Algorithm (requirements)

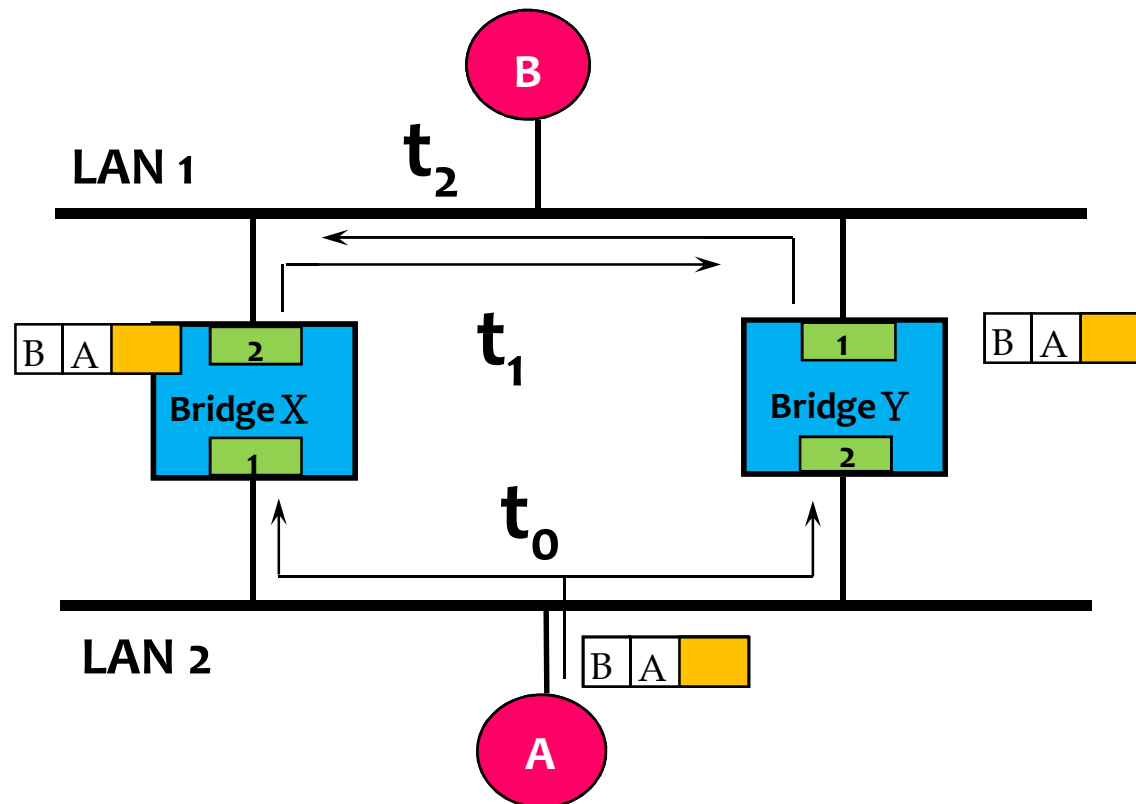## Bridges

- 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

LAN 1

Bridge 2
ID=20
1 TC=10 — R
2 TC=10

Bridge 3
ID=30
1 TC=5 — R
2 TC=5
D

Bridge 1
ID=10
1 TC=10 — D
2 TC=10 — D
Root Bridge

LAN 2

Bridge 4
ID=40
1 TC=5
2 TC=5 — R

LAN 3

Bridge 5
ID=50
1 TC=10 — R
2 TC=5 — D
3 TC=5 — D

LAN 4

LAN 5

TC: Transmission Cost (Path cost)
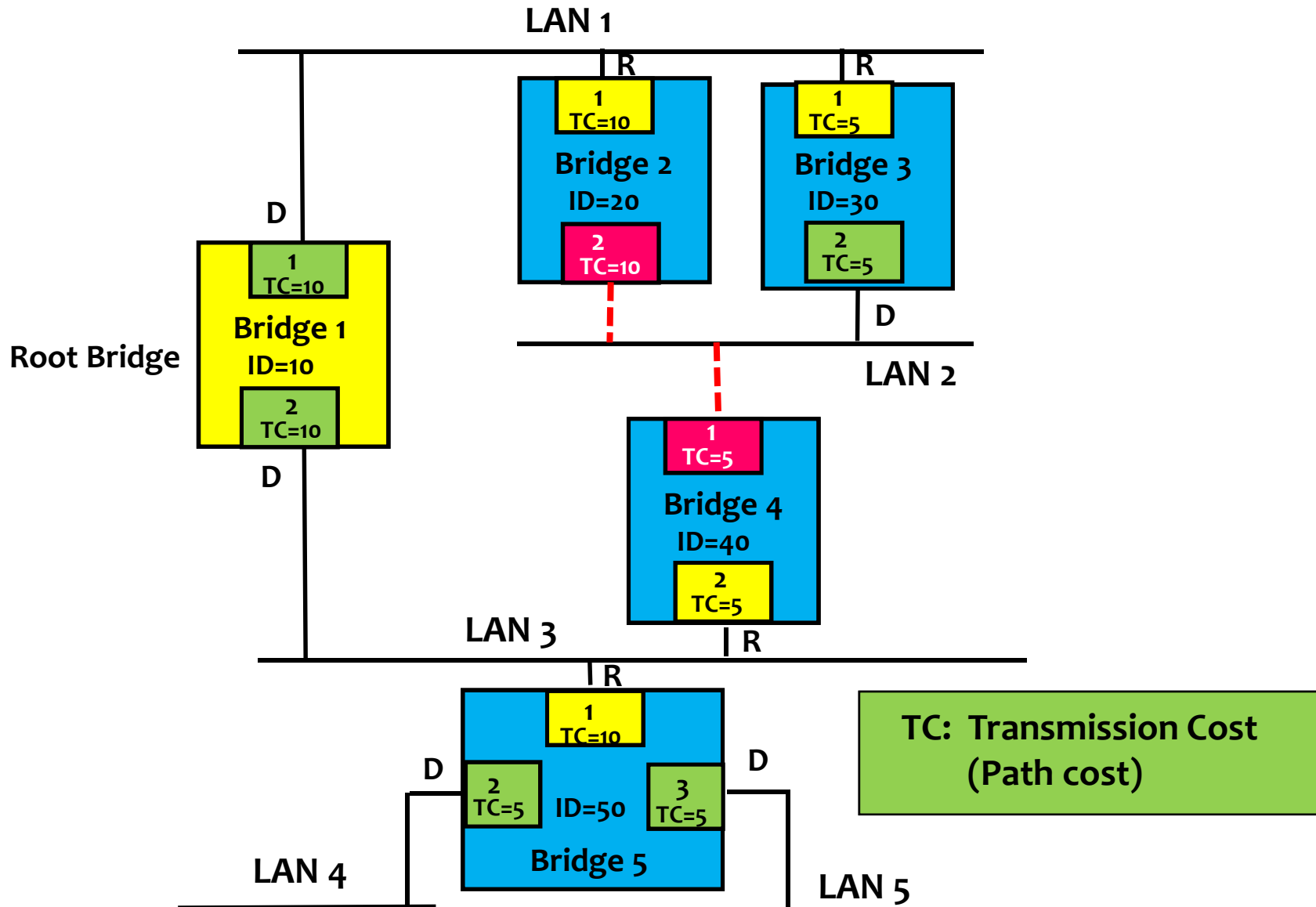
Spanning Tree Example 2 (continued)

Bridge 1
ID=10, RPC=0

1 TC=10 | 2 TC=10

Root Bridge

LAN 3

D | D | LAN 1

Bridge 4
ID=40,RPC=5

2 TC=5

1 TC=5

R

Bridge 3
ID=30,RPC=5

1 TC=5

2 TC=5

R

LAN 2

D

Bridge 5
ID=50, RPC=10

1 TC=10

2 TC=5 | 3 TC=5

R

LAN 4 | LAN 5

D | D

Bridge 2
ID=20,RPC=10

1 TC=10

2 TC=10

R

RPC: Root Path Cost
TC: Transmission Cost
D: Designated Port
R: Root Port
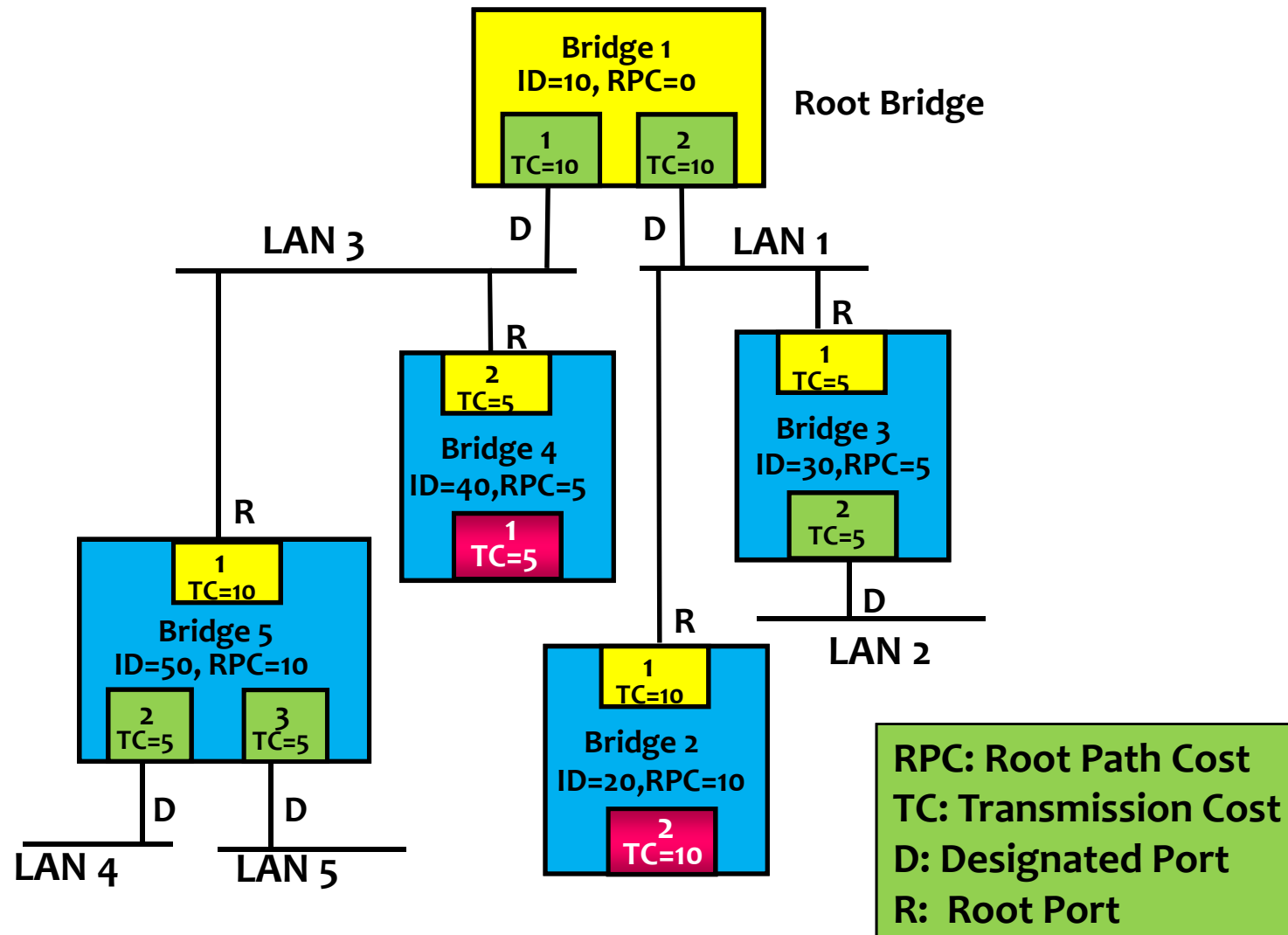
Spanning Tree Algorithm

■ **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

Blocking

Selected as
a D or R port

Cancel   Cancel   Cancel

Listening   **Learning**   **Forwarding**

After a forward delay time
(15 seconds)

After a forward delay time
(15 seconds)
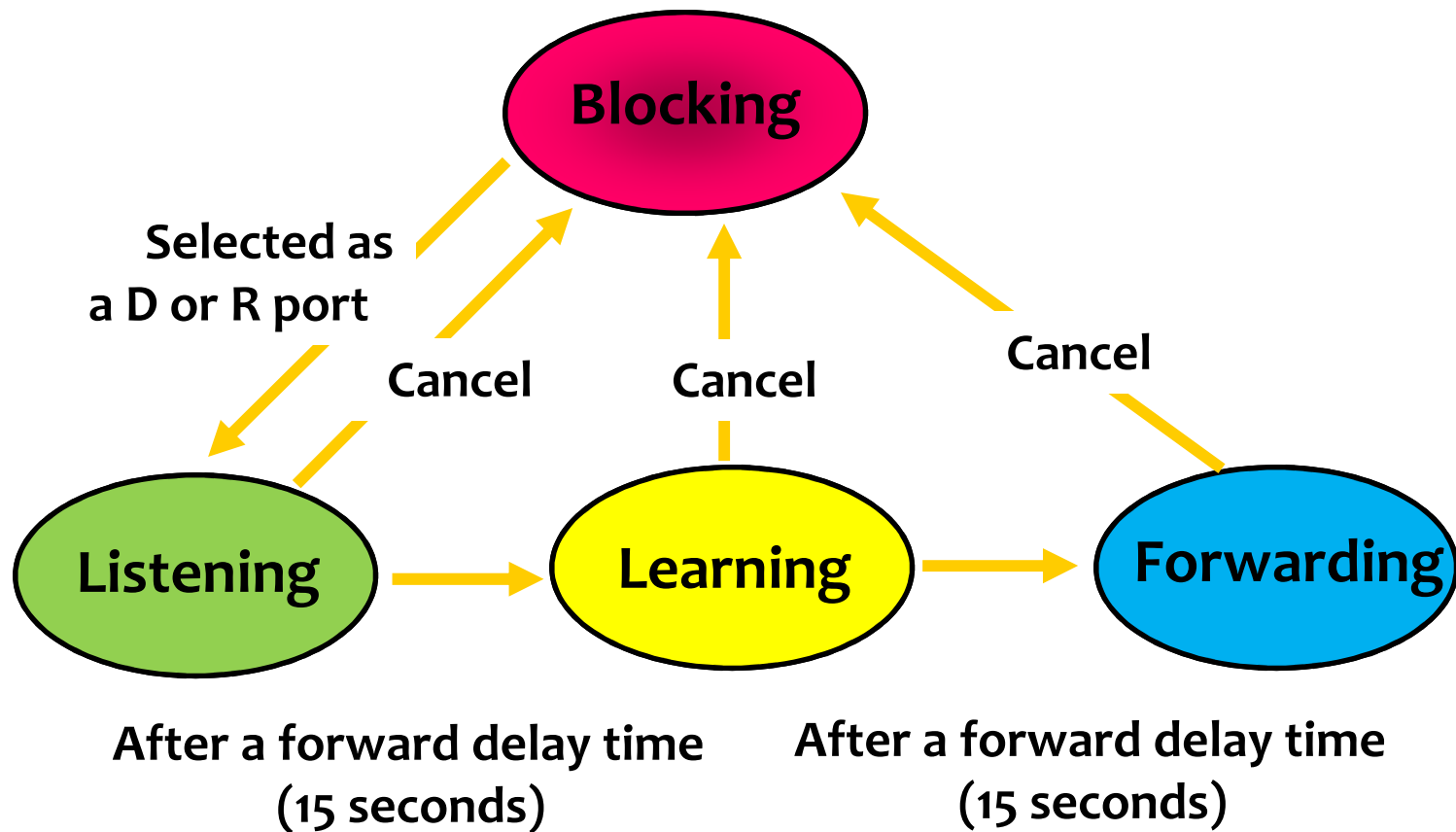
Bridge Protocol Data Unit (BPDU)

Bytes

Bytes	
2	Protocol ID
1	Version ID
1	BPDU Type
1	Flag
8	Root Bridge ID
4	RPC
8	Bridge ID
2	Root Port ID
2	Message Age
2	Time Limit
2	Hello Time
2	Forward delay

Bytes

Bytes	
2	Protocol ID
1	Version ID
1	BPDU Type

(b) Topology Change BPDU
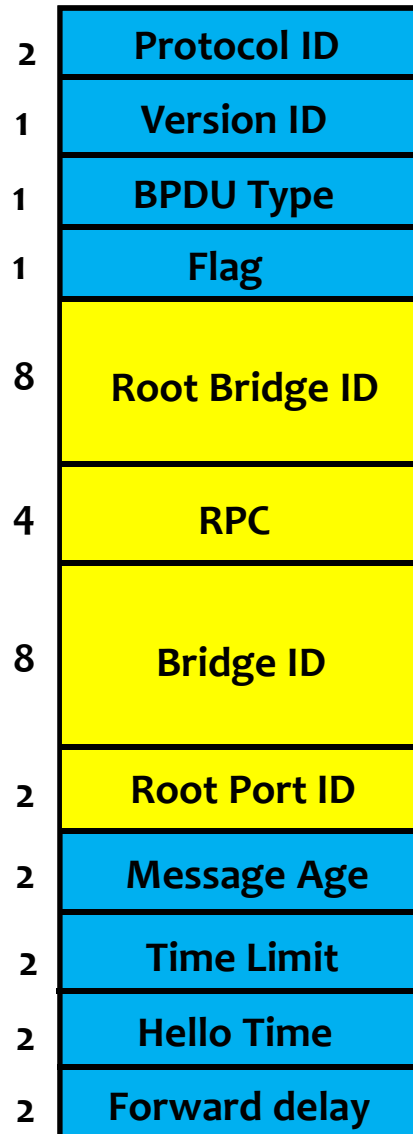
(a) Network Configuration BPDU

Spanning Tree Algorithm Example

(5) RPC = 20

i TC=15

Bridge X

j TC=10

(4) RPC = 58, R = j
(6) RPC = 35, R = i, D(W) = j
(11) RPC = 35, R = i

(7) RPC = 35

(9) RPC = 25

l TC=5

Bridge Y

k TC=5

(4) RPC = 53, R = k
(8) RPC = 40, R = k
(10) RPC = 30, R = l, D(W) = k

(11) RPC = 30

LAN W

(3) RPC = 48

m TC=10

Bridge Z

n TC=10

(2) RPC = 48, R = n, D(W) = m
(8) RPC = 45, R = m
(12) RPC = 40, R = m

D(W): Designated Port of LAN W

(1) RPC = 38
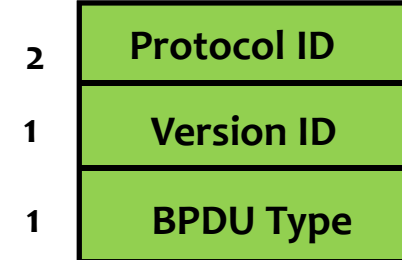
Spanning Tree Algorithm - 35
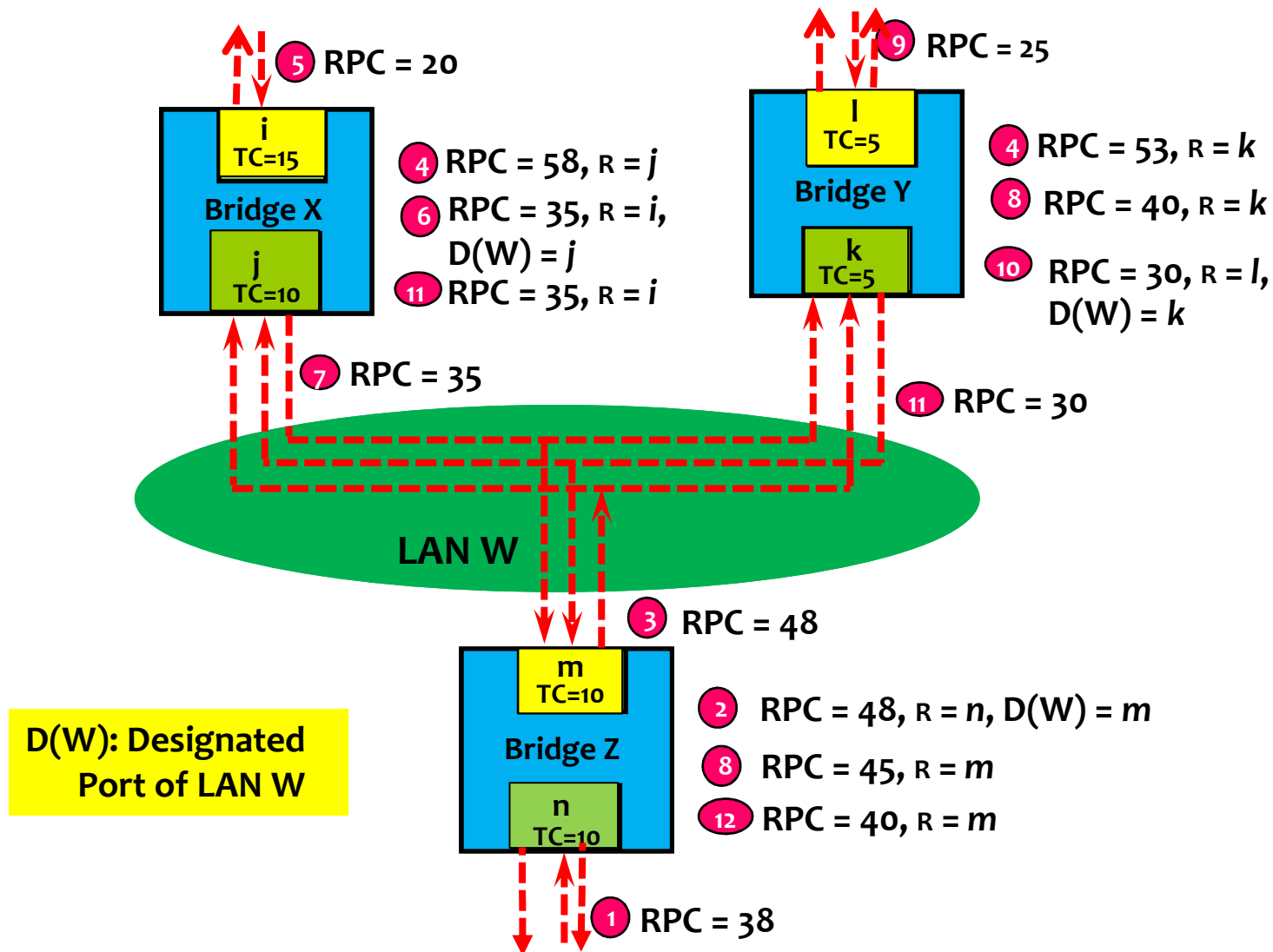
Spanning Tree Algorithm Example (Continued)
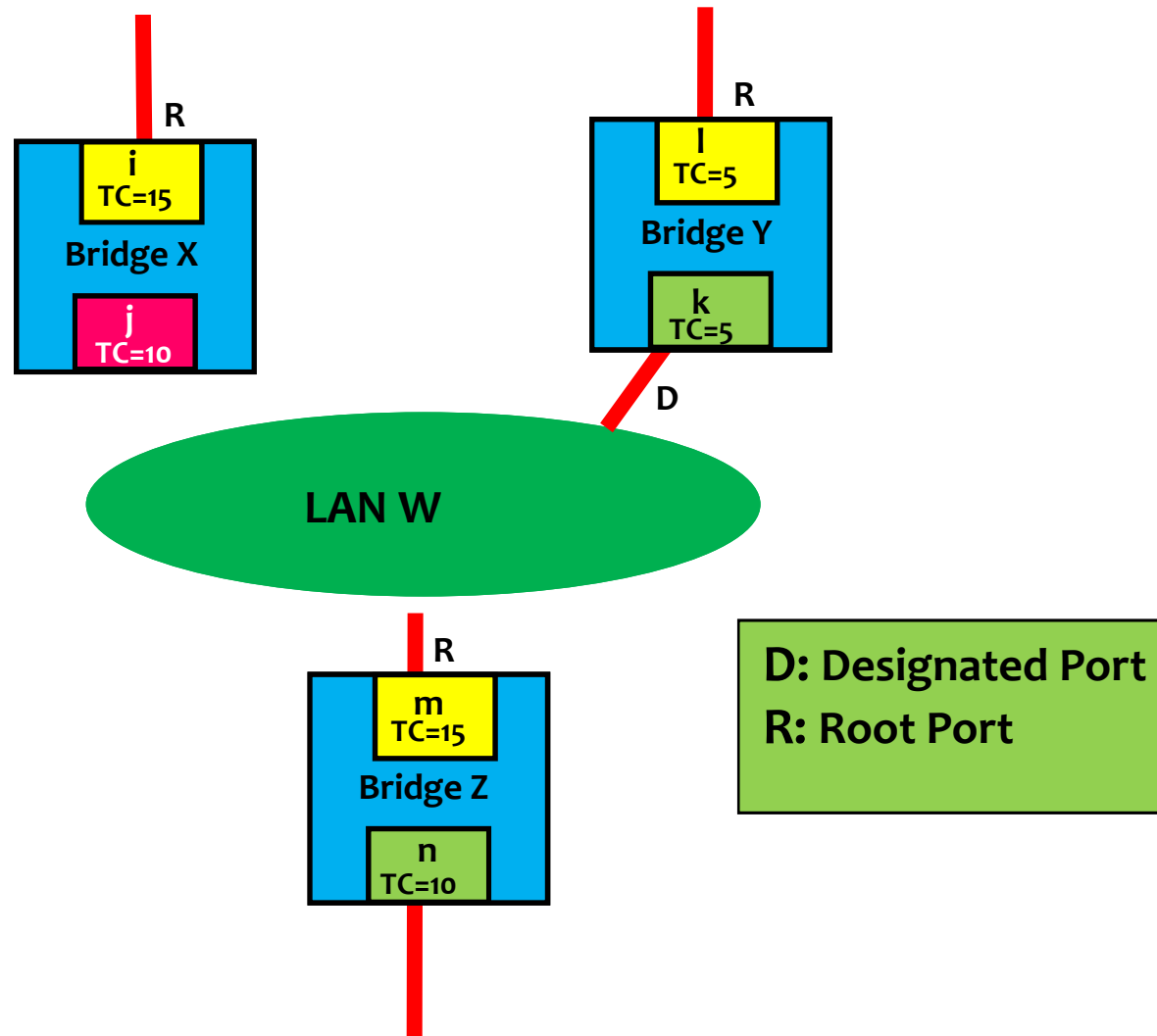
Spanning Tree Features

- 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.

The Spanning Tree is Predictable
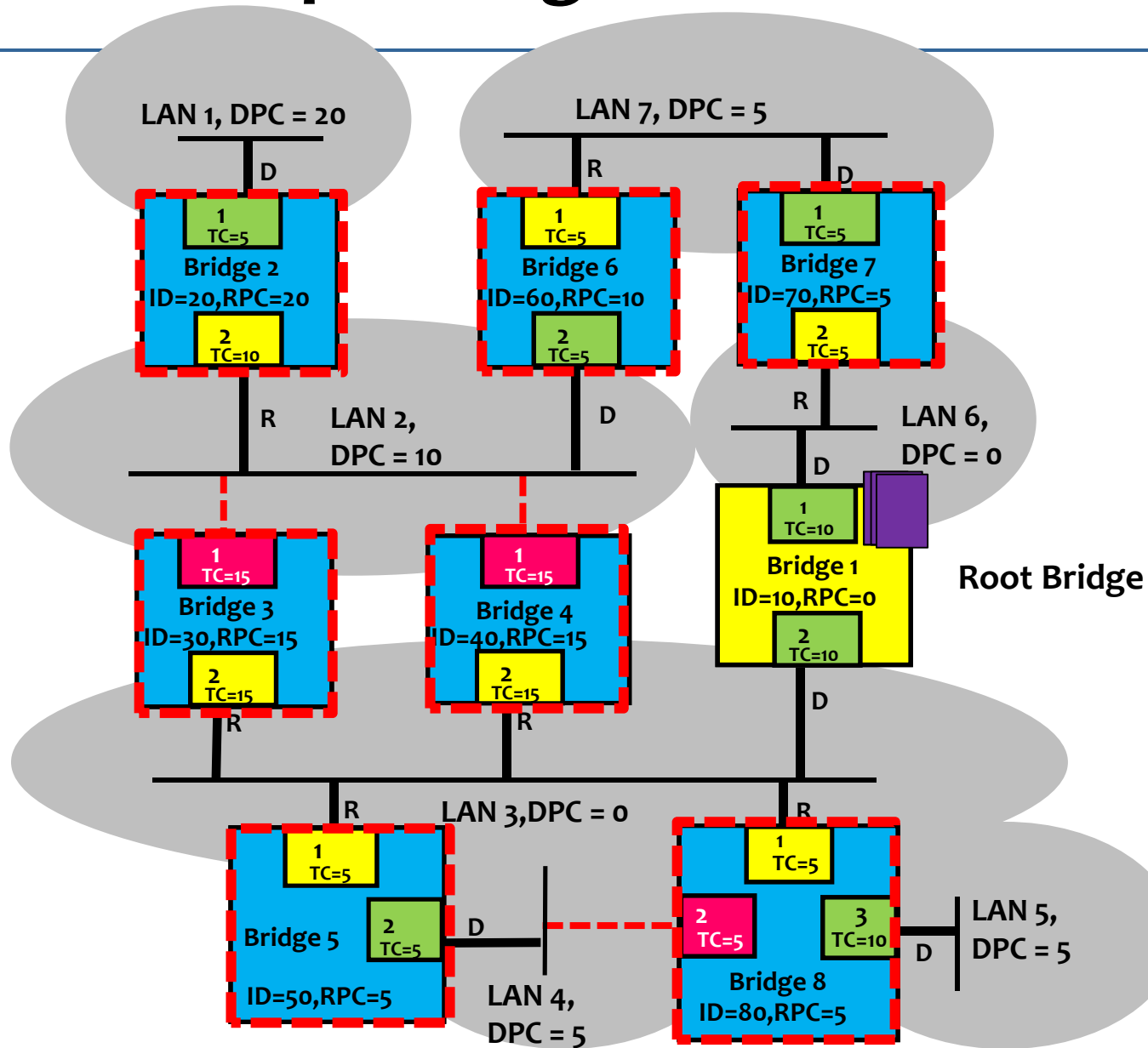
LAN 1, DPC = 20

D

1 TC=5
Bridge 2 ID=20,RPC=20
2 TC=10

LAN 7, DPC = 5

R

1 TC=5
Bridge 6 ID=60,RPC=10
2 TC=5

D

1 TC=5
Bridge 7 ID=70,RPC=5
2 TC=5

R

LAN 2, DPC = 10

D

1 TC=15
Bridge 3 ID=30,RPC=15
2 TC=15

R

1 TC=15
Bridge 4 ID=40,RPC=15
2 TC=15

R

LAN 6, DPC = 0

D

1 TC=10
Bridge 1 ID=10,RPC=0
2 TC=10

Root Bridge

D

LAN 3,DPC = 0

R

1 TC=5
Bridge 5 **2** TC=5
ID=50,RPC=5

D

LAN 4, DPC = 5

R

1 TC=5
2 TC=5 **3** TC=10
Bridge 8 ID=80,RPC=5

D

LAN 5, DPC = 5

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)

LAN 1, DPC = 25

LAN 7, DPC = 5

D

1
TC=5

Bridge 2
ID=20,RPC=20

2
TC=10

D

1
TC=5

Bridge 7
ID=70,RPC=5

2
TC=5

R

LAN 2.
DPC = 15

R

LAN 6,
DPC = 0

D

D

1
TC=10

Bridge 1
ID=10,RPC=0

2
TC=10

Root Bridge

1
TC=15

Bridge 3
ID=30,RPC=15

1
TC=1

Bridge 4
ID=40,RPC=15

2
TC=15

R

R

D

R

LAN 3,DPC = 0

R

1
TC=5

Bridge 5

2
TC=5

D

1
TC=5

2
TC=5

3
TC=10

D

LAN 5,
DPC = 5

ID=50,RPC=5

LAN 4,
DPC = 5

Bridge 8
ID=80,RPC=5

Spanning Tree Algorithm - 41
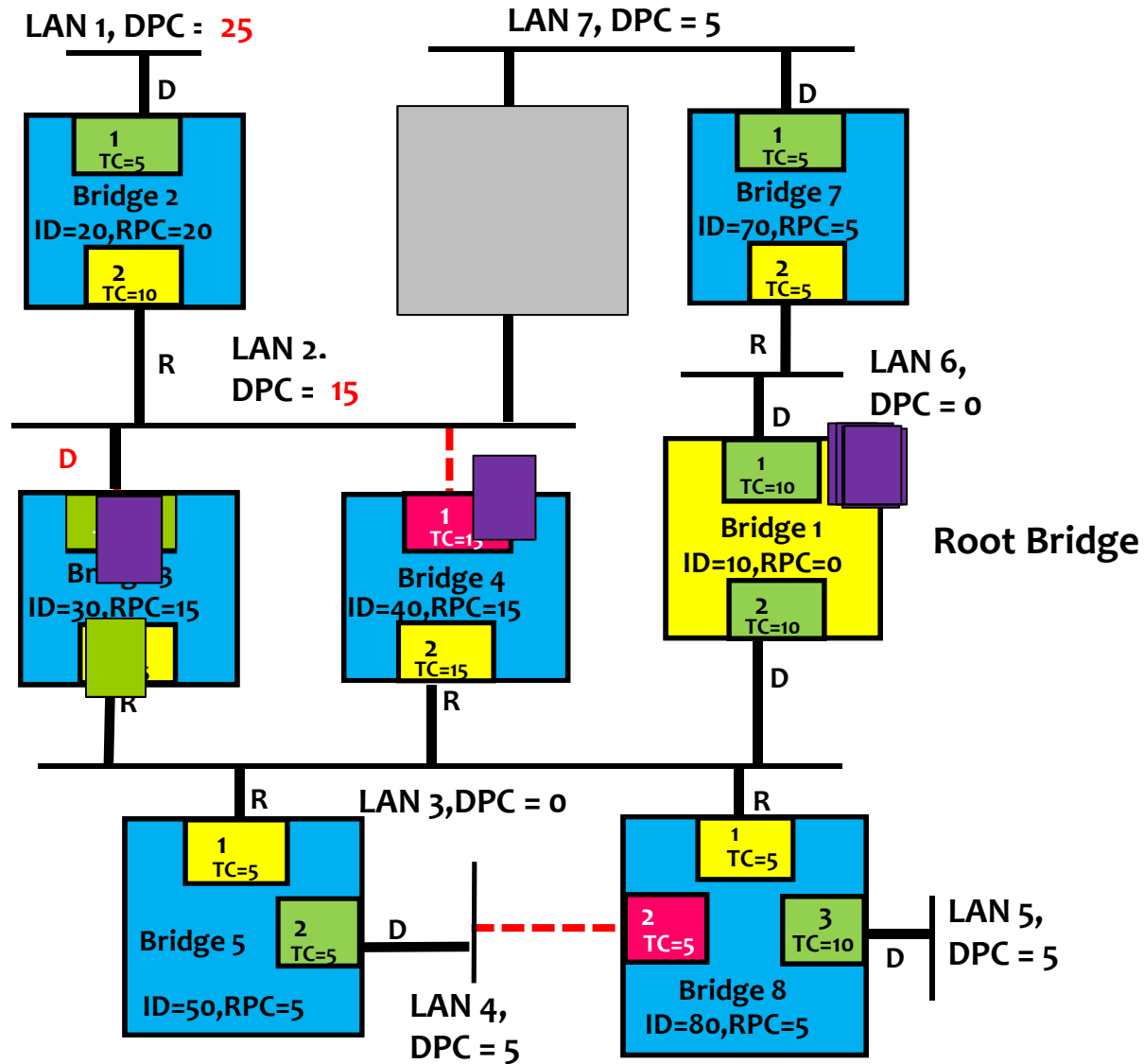
Spanning Tree Maintenance
Example 1 (Bridge Faults)

- **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

LAN 1, DPC = 25

LAN 7, DPC = 5

D

1
TC=5

Bridge 2
ID=20,RPC=20

2
TC=10

Bridge 6
ID=60

D

1
TC=5

Bridge 7
ID=70,RPC=5

2
TC=5

R

LAN 2,
DPC = 15

R

LAN 6,
DPC = 0

D

1
TC=15

Bridge 3
ID=30,RPC=10

2
TC=10

1
TC=15

Bridge 4
ID=40,RPC=10

2
TC=10

1
TC=

Bridge 1
ID=10,RPC=0

2
TC=10

Root Bridge

R

R

D

R

LAN 3,DPC = 0

R

1
TC=5

Bridge 5

2
TC=5

D

ID=50,RPC=5

LAN 4,
DPC = 5

1
TC=5

2
TC=5

3
TC=10

D

LAN 5,
DPC = 5

Bridge 8
ID=80,RPC=5

Spanning Tree Maintenance
Example 2 (LAN Faults)

LAN 1, DPC = 20

D

1
TC=5

Bridge 2
ID=20,RPC=20

2
TC=10

LAN 7, DPC = 5

R

1
TC=5

Bridge 6
ID=60,RPC=10

2
TC=5

D

1
TC=5

Bridge 7
ID=70,RPC=5

2
TC=5

R   **LAN 2,**
      DPC = 10

D

R   **LAN 6,**
      DPC = 0

R

1
TC=15

Bridge 3
ID=30,RPC= 25

R

1
TC=15

Bridge 4
ID=40,RPC= 25

D

1
TC=10

Bridge 1
ID=10,RPC=0

2
TC=10

Root Bridge

D

Root Bridge

Bridge 5

D

ID=50,RPC= 0

LAN 4.
DPC = 0

R

2

3
TC=10

D

Bridge 8
ID=80,RPC=5

LAN 5,
DPC = 5

Spanning Tree Algorithm - 45

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

LAN 1, DPC = 20

LAN 7, DPC = 5

D

R

D

1 TC=5
Bridge 2 ID=20,RPC=20
2 TC=10

1 TC=5
Bridge 6 ID=60,RPC=10
2 TC=5

1 TC=5
Bridge 7 ID=70,RPC=5
2 TC=5

R LAN 2, DPC = 10

D

R LAN 6, DPC = 0

R

R

D

1 TC=15
Bridge 3 ID=30,RPC=25
2 TC=10

1 TC=15
Bridge 4 ID=40,RPC=25
2 TC=10

1 TC=10
Bridge 1 ID=10,RPC=0
2 TC=10

Root Bridge

LAN 3

1 TC=5
Brid ID=50,RPC=0
2 TC=5

D

R

1 TC=5
Bridge 8 ID=80,RPC=5
2 TC=5	3 TC=10

LAN 5, DPC = 5

D

LAN 4, DPC = 0

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.**