Introduction to interconnecting devices and BRIDGES

Connecting Devices

Why Connecting Devices in LANs?

(1) LANs do not normally operate in isolation – they are connected to one another or to the Internet to enable sharing of CPUs, data-bases, programs, etc.

(2) as # of devices in a single LAN grows, MAC and error-&-flow control protocols become less effective

- way to avoid bottlenecks is to divide LAN into multiple LANs, thus reducing # of devices per LAN

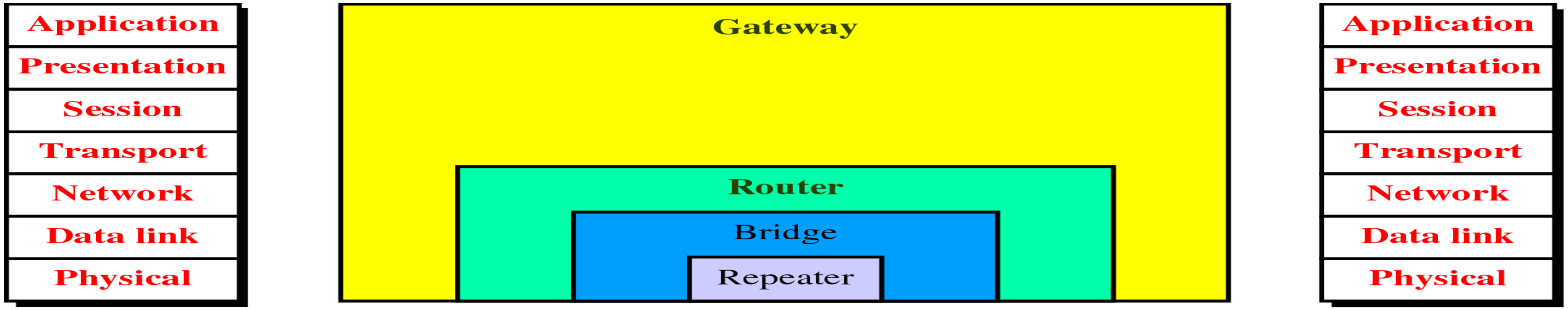
Types of Connecting Devices

- connecting devices can operate in different layers of the Internet model

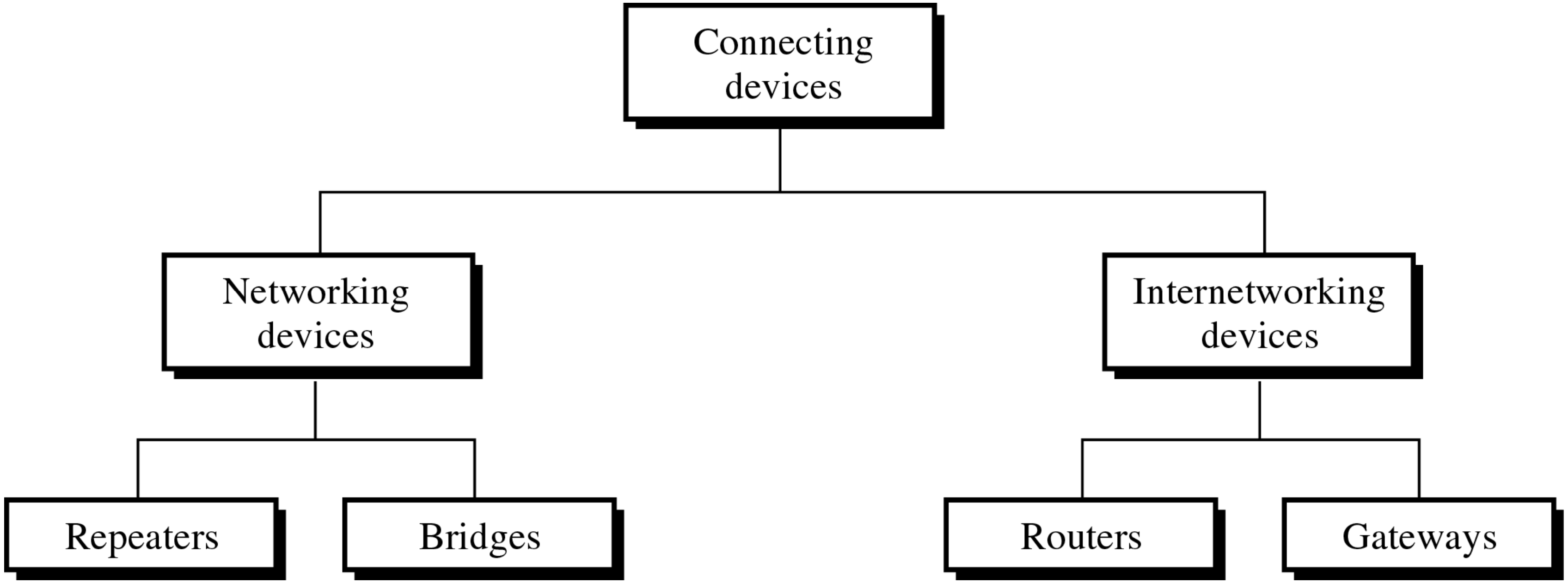
(1) **repeaters** and **hubs** operate in the <u>first layer</u>

(2) **bridges or 2 layer Switches** operate in the <u>first two layers</u>

(3) **routers** operate in the <u>first three layers</u>



Connecting Devices



Connecting devices

FOUR kinds of connecting devices:

- **repeaters (or hubs)**
- **bridges (or two-layer switches)**
- **routers (or three-layer switches)**
- **Gateways**

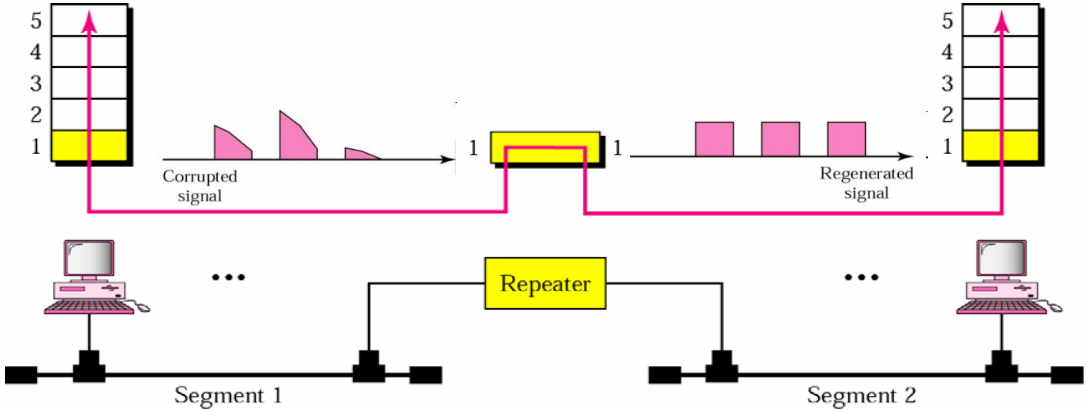
Repeaters and hubs operate in the first layer of the Internet model.
Bridges and two-layer switches operate in the first two layers.
Routers and three-layer switches operate in the first three layers
Gateways operate in the higher layers

Connecting Devices: Repeaters

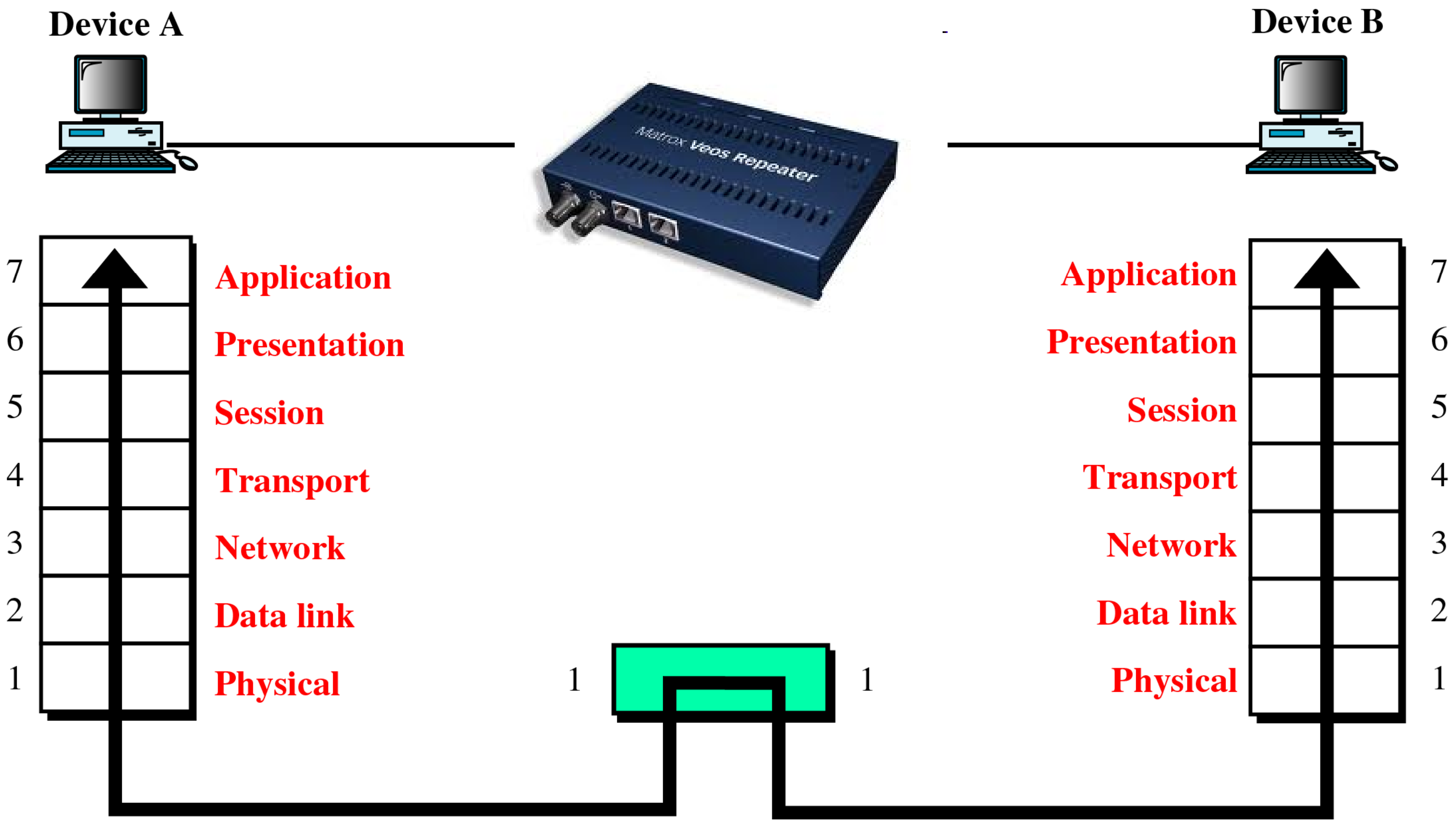
Repeater – **connecting device that operates only in the physical layer:**

(1)receive signal on one end →
(2)regenerate original bit patterns →
(3)send refreshed signal on the other end

- **connects only segments of the same LAN**
  - **segments must run the same protocol**
- **has no filtering capability**
  - **every frame received will be regenerated (not amplified) and forwarded**
- **location of a repeater is crucial – repeater must be placed so that a signal reaches it before noise changes the meaning of any of its bits**



A Repeater in the OSI Model



Device A

7 Application
6 Presentation
5 Session
4 Transport
3 Network
2 Data link
1 Physical

1 1

Device B

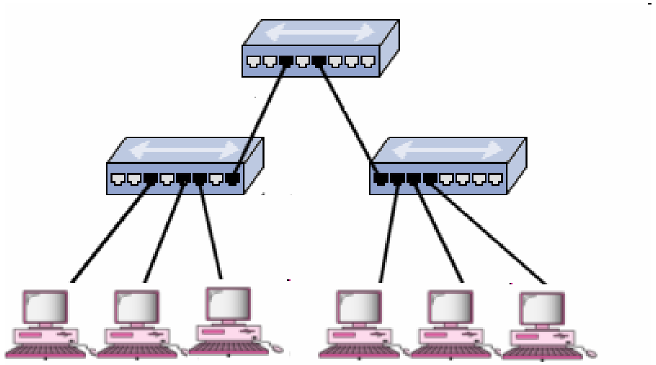
Application 7
Presentation 6
Session 5
Transport 4
Network 3
Data link 2
Physical 1

Connecting Devices: Hubs

Hub – multiport repeater !!!

(1)receive signal on one end →
(2)regenerate original bit patterns →
(3)send refreshed signal over all other ports

- **passive hubs: simply send signal to all connected hosts, without amplifying it**
- **active hubs: are connected to electric power source, and are used to refresh the signal sent to all ports**



Repeaters and hubs primarily extend the physical reach of a network, but at the same time they can create problems.

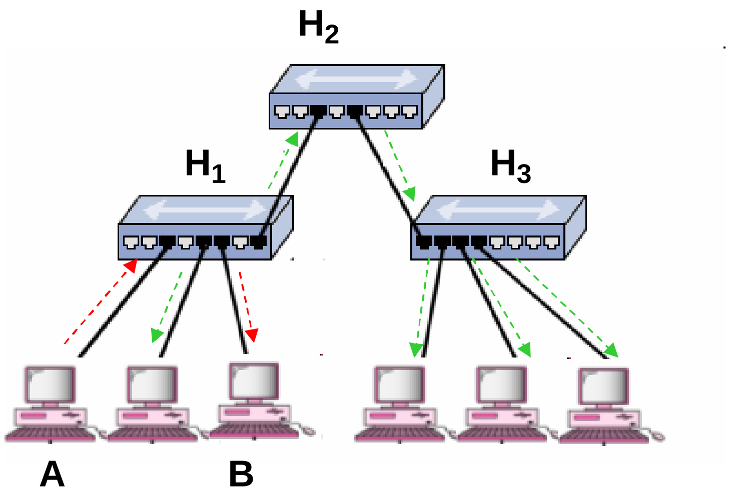
more devices access the medium ⇒ more traffic ⇒ degraded LAN performance

Connecting Devices: Hubs (cont.)

Example **[unnecessary frame flooding in LANs with hubs]**

If node A sends a frame to node B, hubs H_1, H_2, and H_3 forward the frame to all possible location.

H2, and H3 do not have built-in logic to know that A and B are on the same LAN and connected to the same hub, and that repeating the frame is pointless.



The problem of frame flooding can be resolved by filtering out (not forwarding) frames that have both 'source' and 'destination' address on the same LAN.

Physical layer solutions not satisfactory

Physical layer devices – repeater, hub – do not solve the more interesting problems

E.g., how to handle load

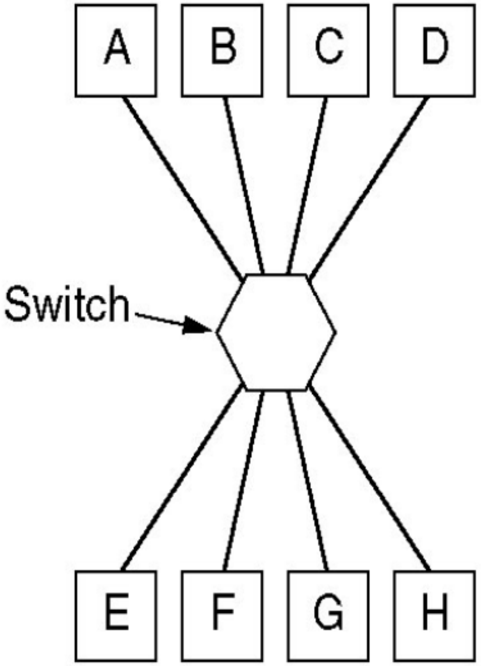
- Some knowledge of the data link layer structure is necessary
- To be able to inspect the content of the packets/frames and *do* something with that knowledge

Link-layer solutions

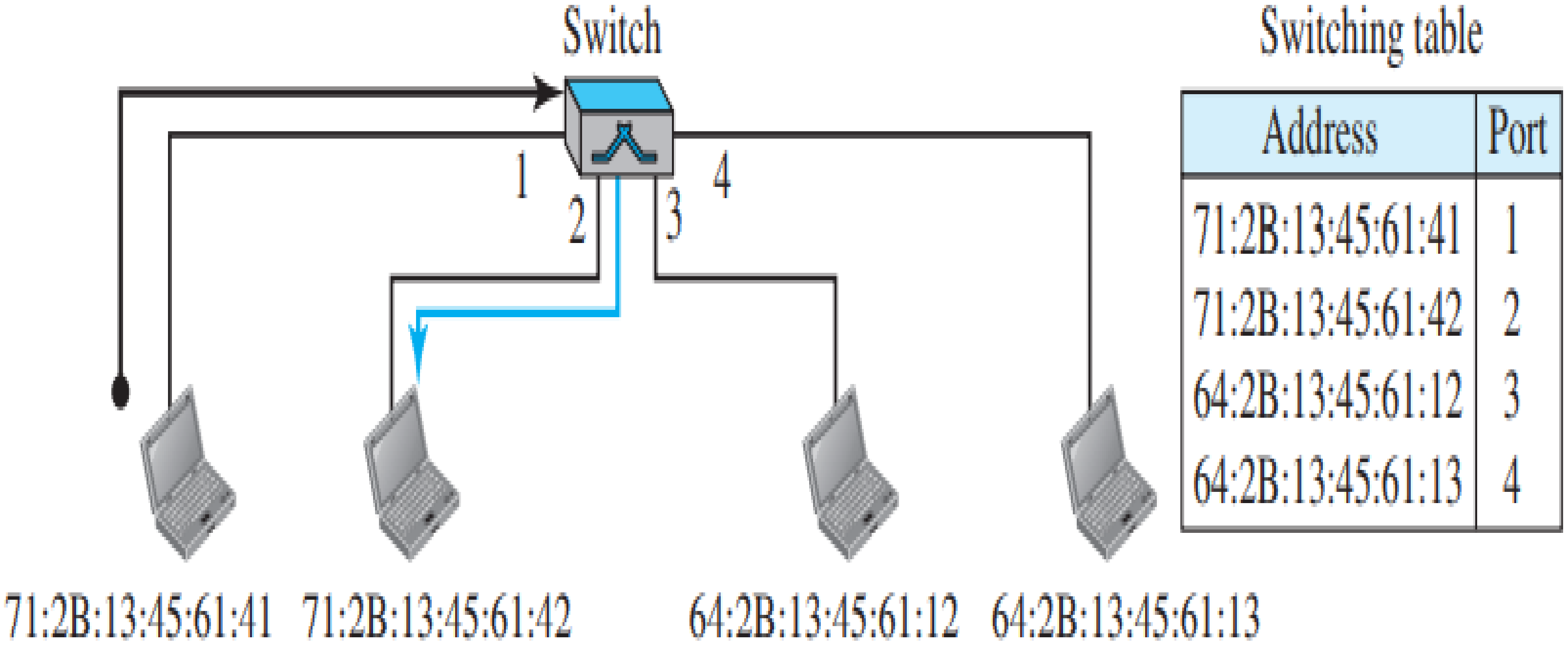
- Bridge & switch
- Switch: Interconnect several *terminals*
- Bridge: Interconnect several *networks*
-

SWITCHES

- Use a switch to connect several terminals in a single lan
- Switch inspects an arriving packet's destination addresses and forwards its *only* on the right cable
- ***uses Backward learning-*** to obtain knowledge about directions (Broadcast and unicast)



Switch Learning



Switching table

Address	Port
71:2B:13:45:61:41	1
71:2B:13:45:61:42	2
64:2B:13:45:61:12	3
64:2B:13:45:61:13	4

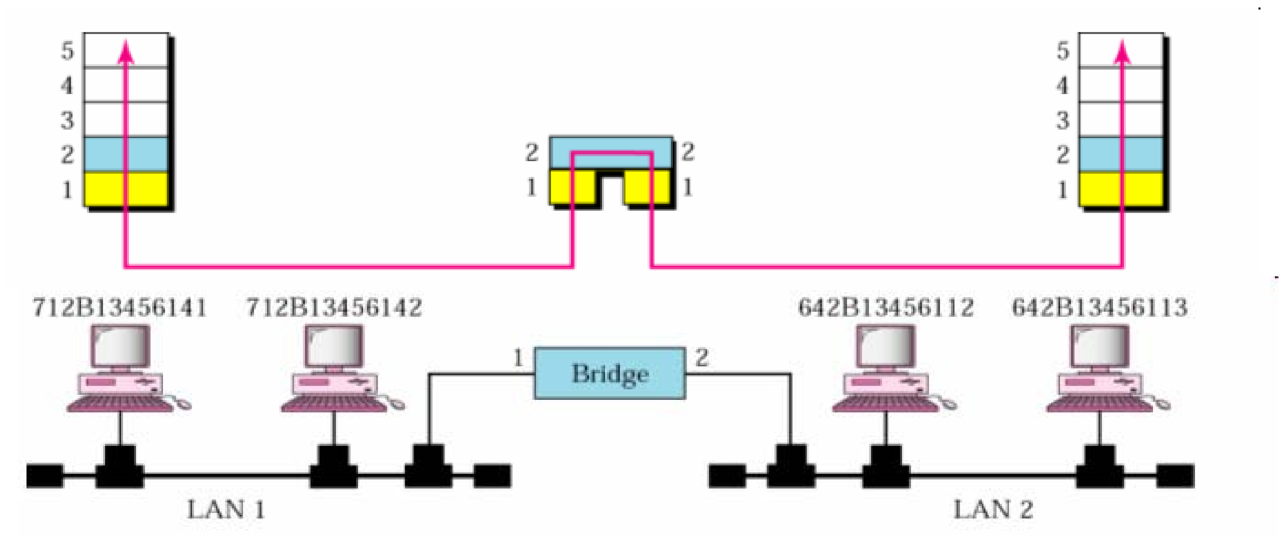
Connecting Devices: Bridges

Bridge – connecting device that <u>operates in both physical & data link layer</u>

- **as a physical-layer device, bridge regenerates the signal it receives**

- **as a data link layer device, bridge checks physical / MAC addresses (both source and destination) in frames**
  - **if frame sent in LAN 1 is destined for a device on LAN 2 – receive and forward the frame; otherwise ignore the frame**

- **to be able to properly forward / filter frames, bridge must build / learn a 'forwarding table', aka 'forwarding database'**

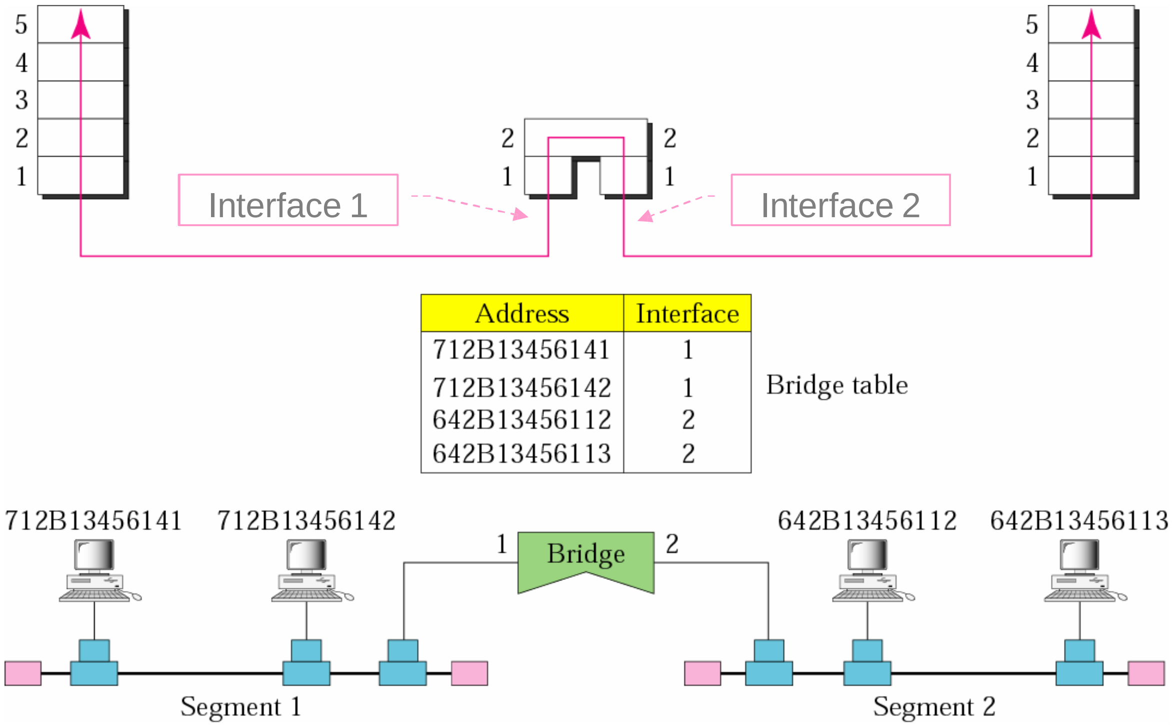


Bridges (cont.)

Example [filtering with bridges]

Assume the bridge has a table that maps addresses to ports, i.e. maps the address of each host to the bridge port # through which frames from the given host arrive.

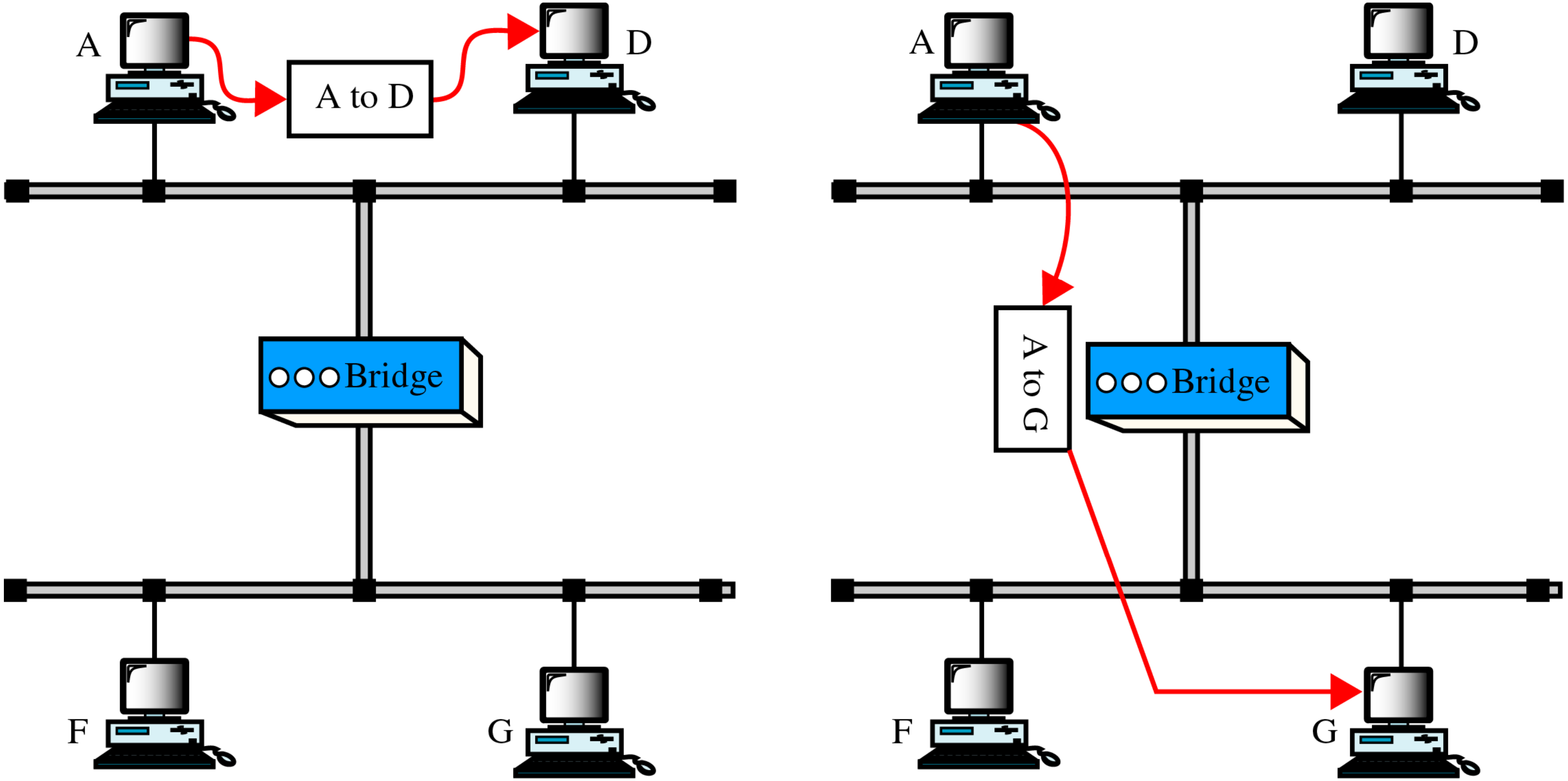
If a frame destined for station 712B1345142 arrives at port 1, the bridge consults its table to find the departing port. As frames for 712B1345142 leave through port 1, there is no need for frame forwarding.



Address	Interface
712B13456141	1
712B13456142	1
642B13456112	2
642B13456113	2

Bridge table

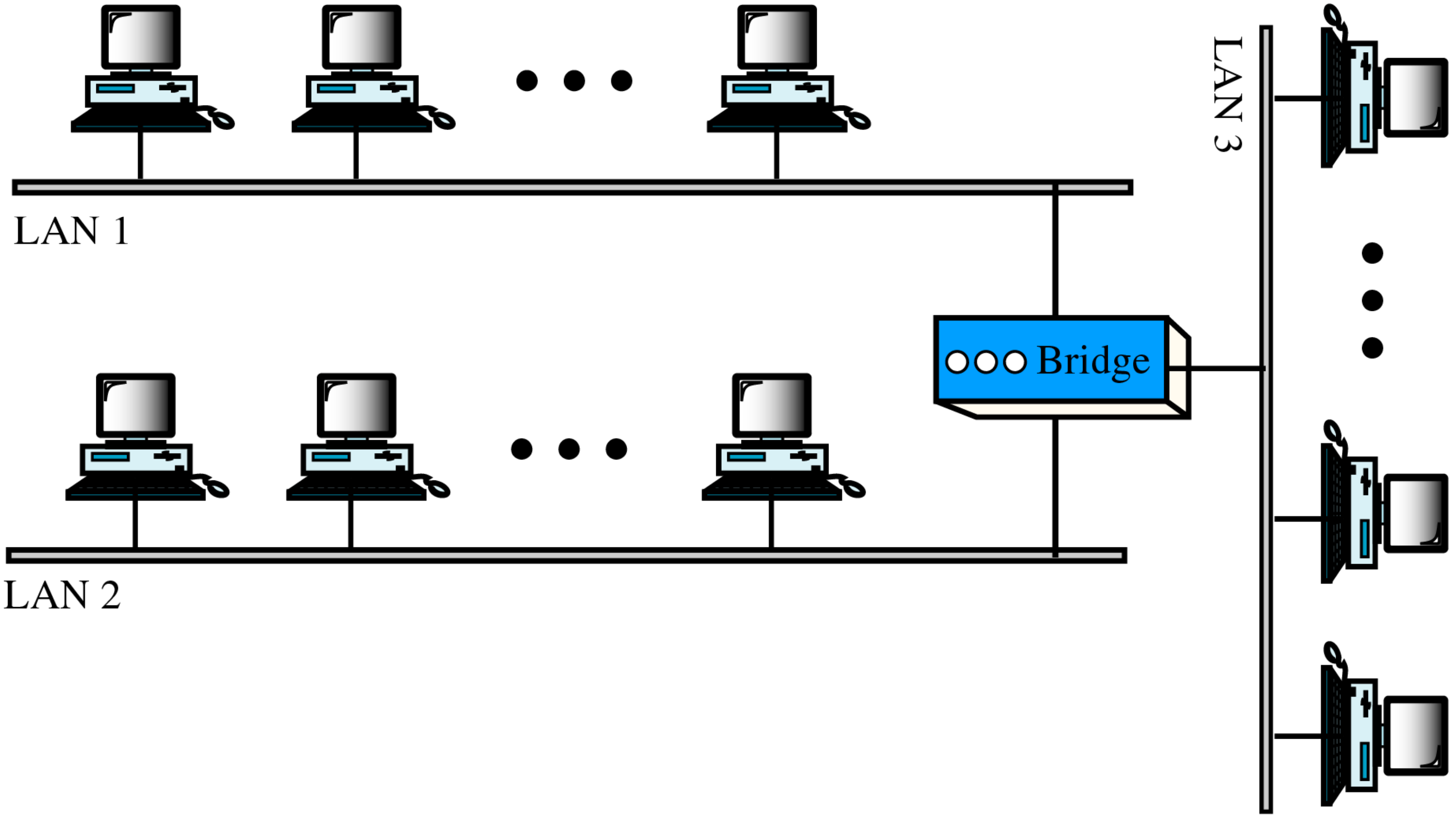
Function of a Bridge



a. A packet from A to D

b. A packet from A to G

Multiport Bridge



Bridge Learning

- **earliest bridges used static forwarding tables**
  - **system administrators would manually enter each table entry**
  - **simple but impractical process – whenever a new station was added or removed, the table had to be modified manually**

- **dynamic forwarding tables – bridge learns the location of all stations gradually, as it operates, and builds forwarding table automatically**

- **learning process:**
  - **bridge inspects both source and** destination address of each received frame

(a)source address is compared with each entry in table

- **if a match is not found, add source address together with port number on which frame was received to table**
- **if a match is found, do nothing**

(b)destination address is compared with each entry in the table

- **if a match is not found, flood frame on all ports except the one on which the frame was received**
- **if a match is found and port is one on which frame was received, do nothing; otherwise, forward frame to port indicated in table**

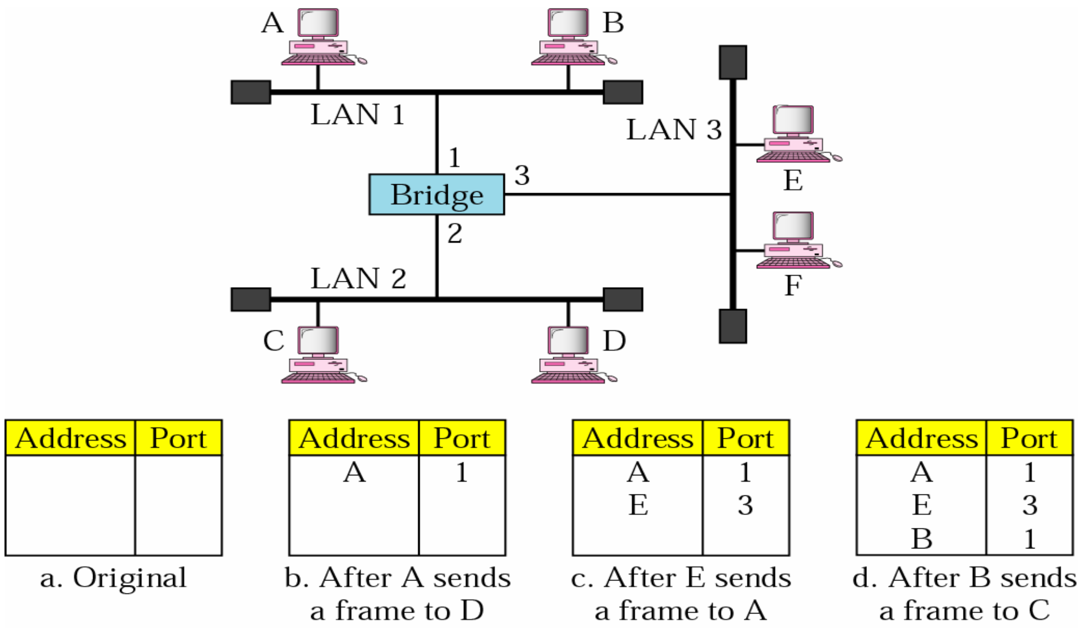
Bridge Learning

(a) **When station A sends a frame to station D, the bridge does not have any entry for either A or D. Hence,**

- **frame is flooded on ports 2 and 3**
- **by looking at the source address, the bridge learns that station A must be located on the LAN connected to port 1 (LAN 1) ⇒ frames destined for A must be sent out through port 1.**

(b) **When station E sends a frame to station A, the bridge has an entry for A. Hence**

- **the frame is forwarded only to port 1**
- **the source address of the frame is added as a second entry to the table**



Address	Port

a. Original

Address	Port
A	1

b. After A sends a frame to D

Address	Port
A	1
E	3

c. After E sends a frame to A

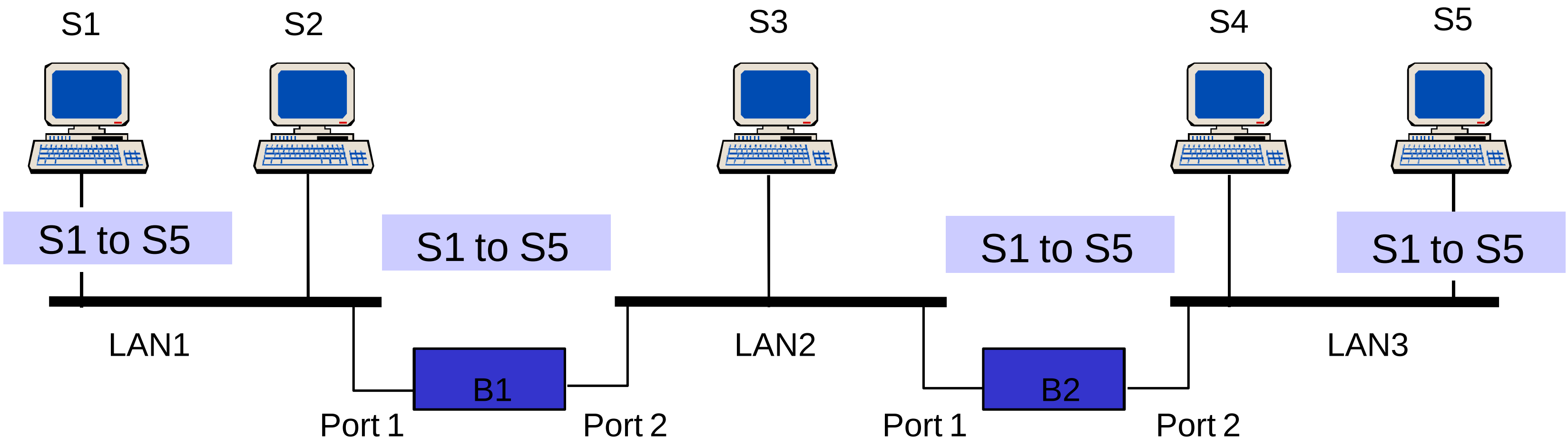
Address	Port
A	1
E	3
B	1

d. After B sends a frame to C

Bridges (cont.)

Example **[bridge learning]**

S_1 sends a frame to S_5.

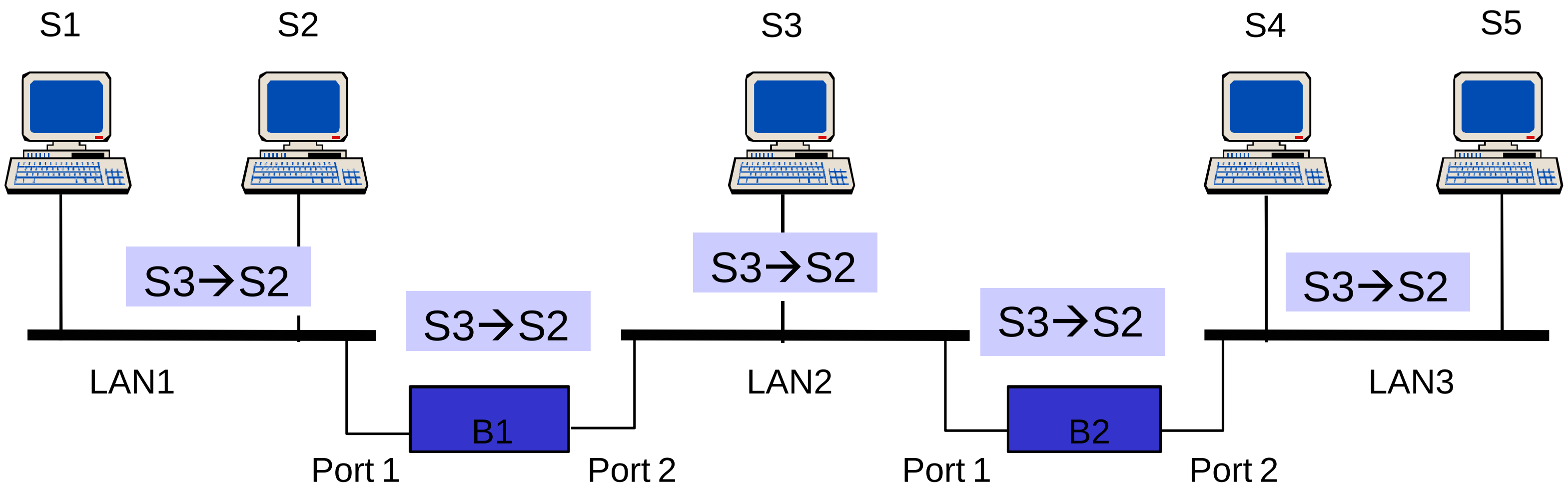


Address	Port
S1	1

Address	Port
S1	1

Bridges (cont.)

S_3 sends a frame to S_2.

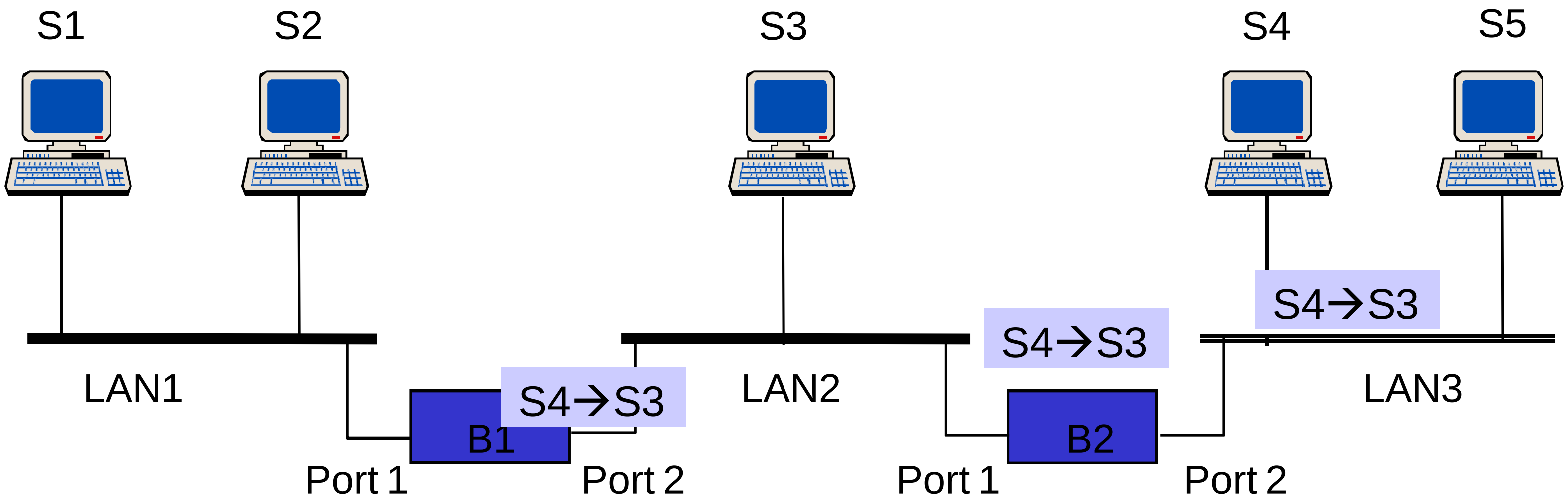


Address	Port
S1	1
S3	2

Address	Port
S1	1
S3	1

Connecting Devices: Bridges (cont.)

S_4 sends a frame to S_3.

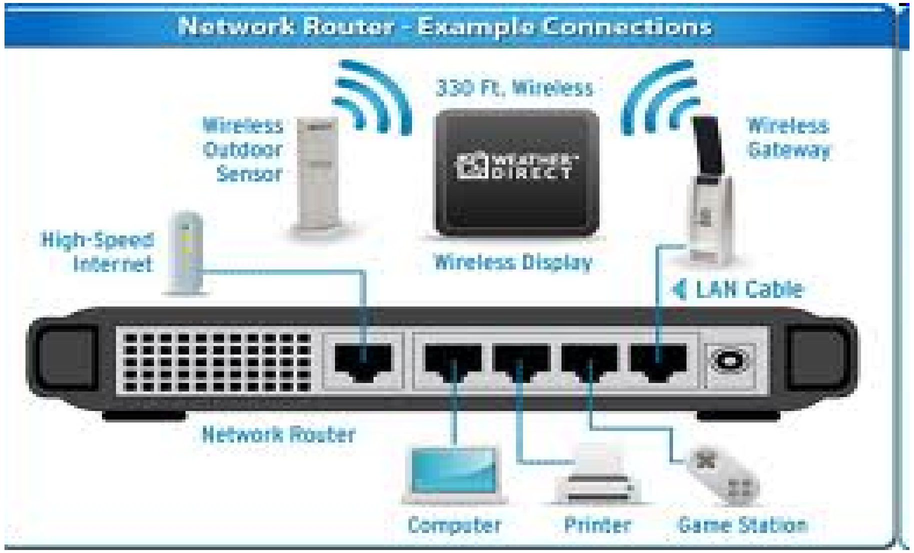


Address	Port
S1	1
S3	2
S4	2

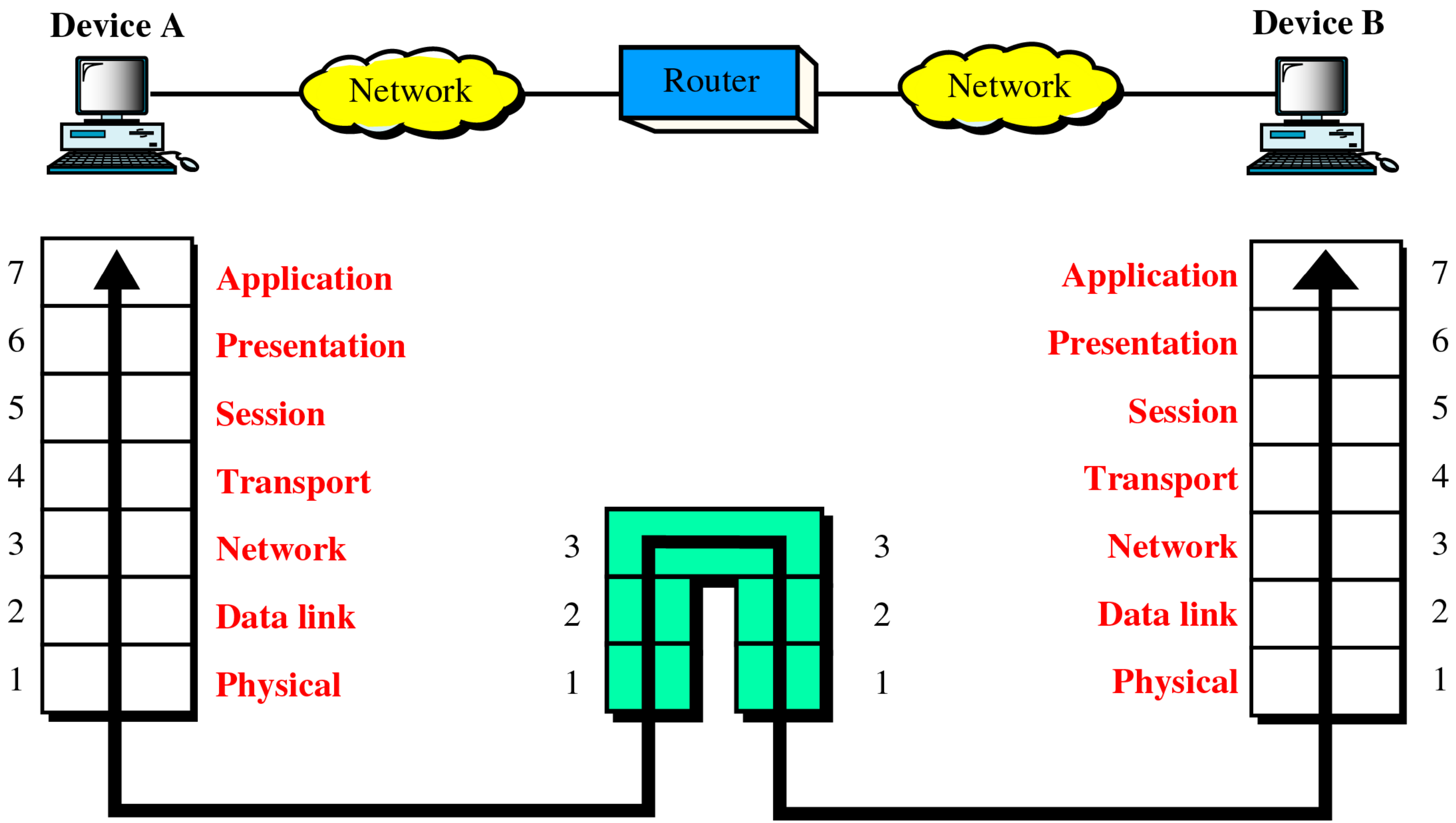
Address	Port
S1	1
S3	1
S4	2

ROUTER

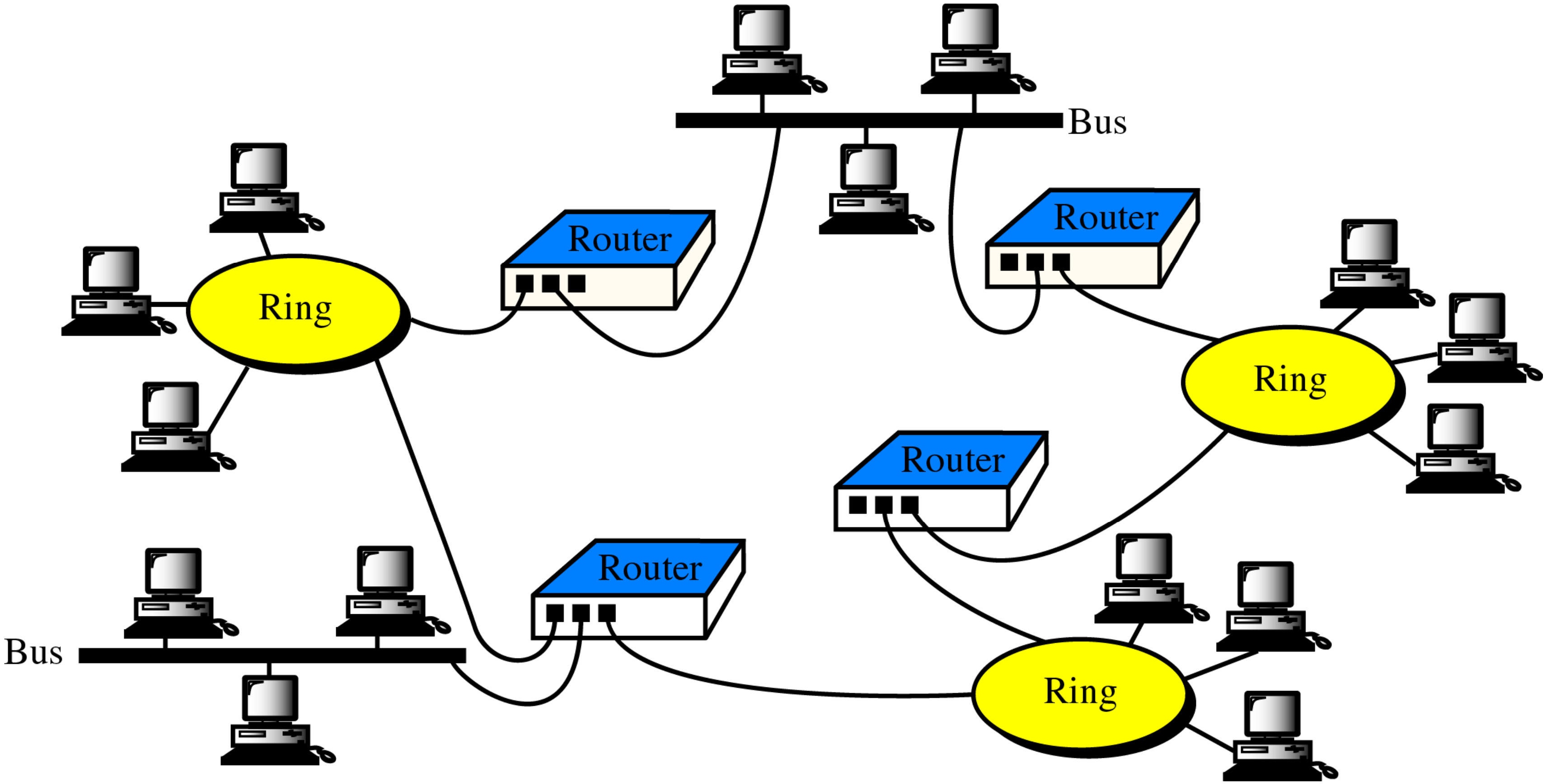
- Router connects lans with different protocols (Ethernet lan ,wifi lan)
- For interconnection *outside* a single LAN/connection of LAN and WAN, these simple addresses are insufficient
- Need ip addressing(logical)



A Router in the OSI Model



Routers in an Internet



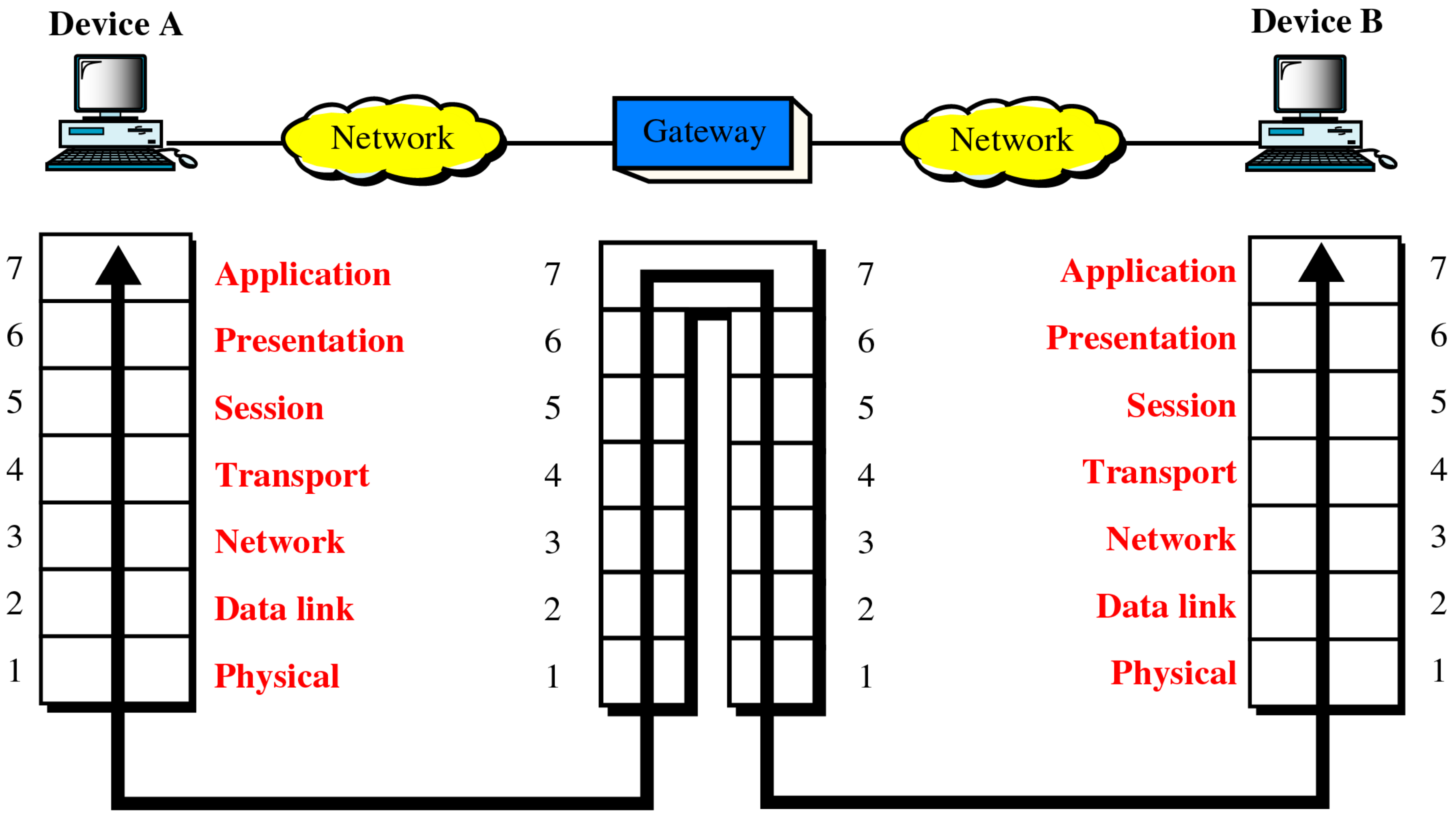
GATEWAY

- If even routers will not do, higher-layer interconnection is necessary: ***Gateways***

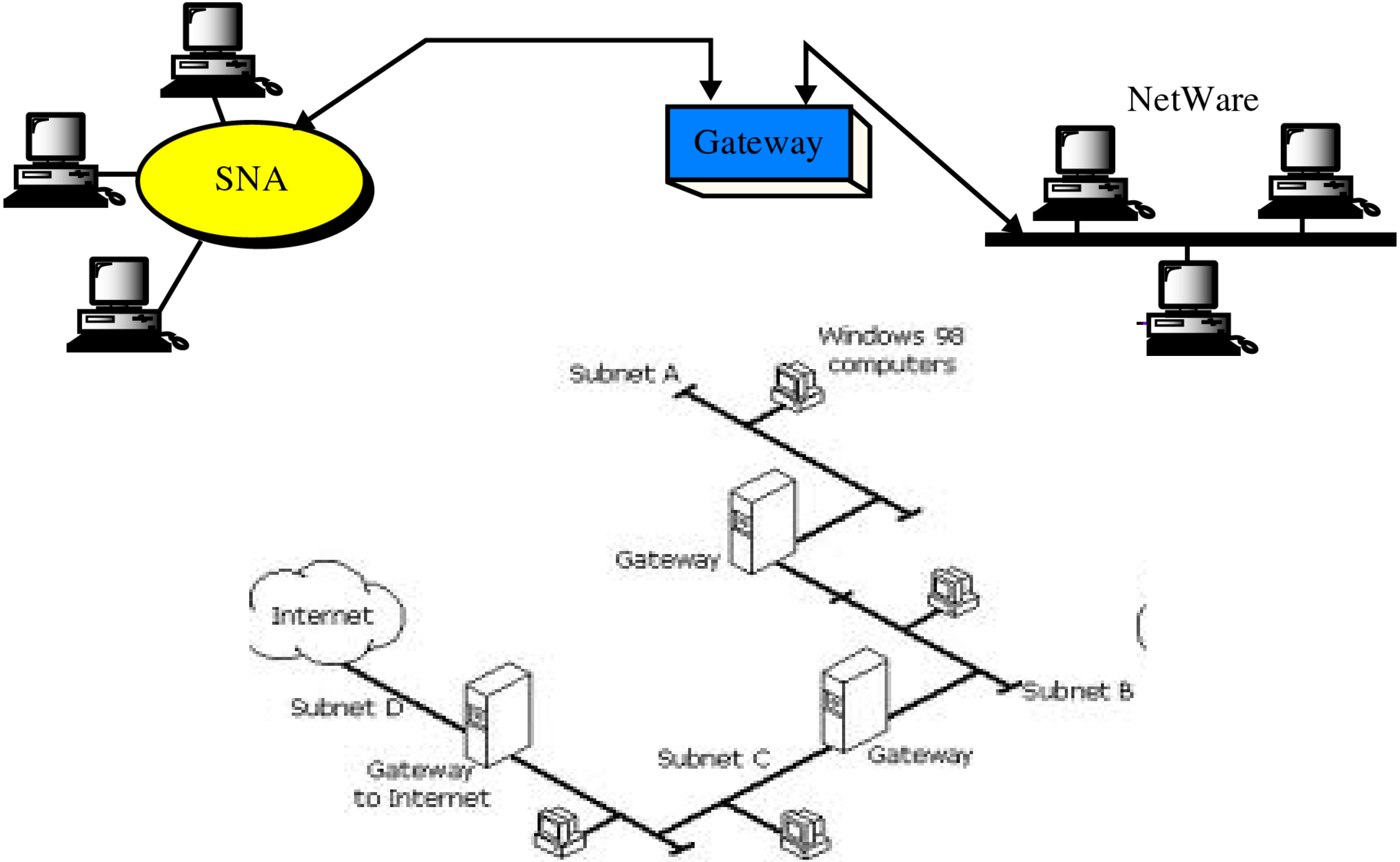
- Work at transport level and upwards

- A **network gateway** is an *internetworking* system capable of joining together two networks that use different base protocols.
- A network gateway can be implemented completely in software, completely in hardware, or as a combination of both.
- E.g., application gateways transforming between HTML ↔ WML or HTTP ↔ WAP

A Gateway in the OSI Model

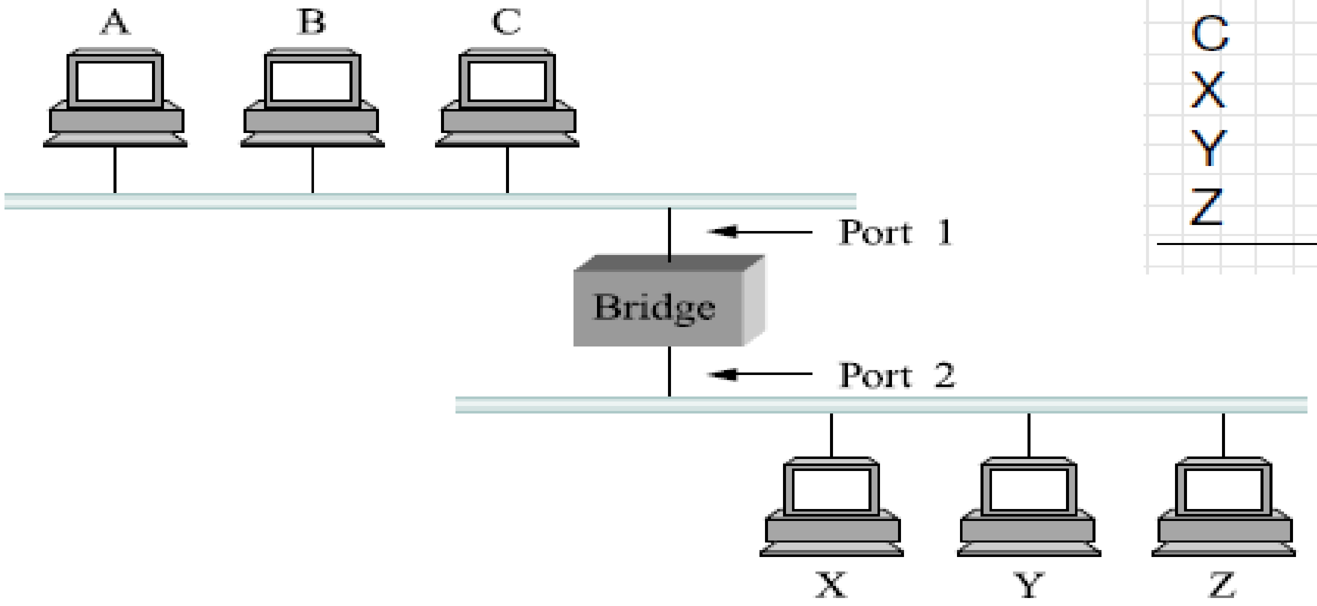


A Gateway


SNA
Gateway
NetWare
Windows 98 computers
Subnet A
Gateway
Internet
Subnet D
Subnet B
Gateway
Subnet C
Gateway to Internet

Bridges –in detail

Bridge installation breaks LAN into LAN segments
A bridge has a bridge table.



Host	Port
A	1
B	1
C	1
X	2
Y	2
Z	2

- The earliest bridges had forwarding tables that were static.
- The systems administrator would manually enter each table entry during bridge setup.
- process was simple, it was not practical.
- If a station was added or deleted, the table had to be modified manually. The same was true if a station's MAC address changed
- A better solution to the static table is a dynamic table that maps addresses to ports automatically.
- To make a table dynamic, we need a bridge that gradually learns from the frame movements.

- Bridge needs a forwarding table: which host is on which port

- ***A bridge does not change the physical(MAC) addresses in a frame.***

- How do bridges learn the location of the stations?

- For each arriving frame:

  1)Add source address (together with port number)to the forwarding table

  2) Check if destination address is in the forwarding table?

    a)if it is, transmit the frame on the respective port

    b)Otherwise, broadcast the frame on all ports

Transparent Bridges

- The stations are completely unaware of the bridge's existence

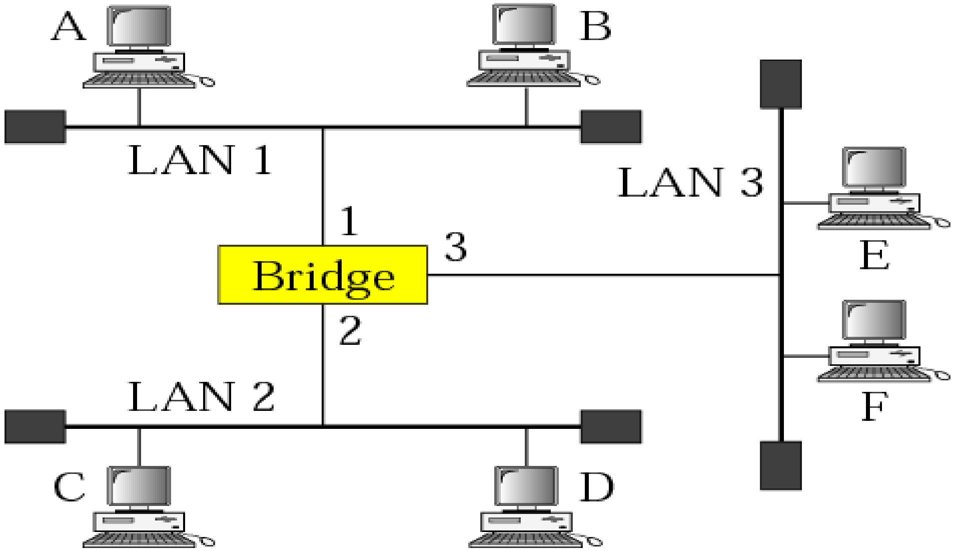
  - the stations does not reconfigured when a bridge is added or deleted

- A system equipped with transparent bridges must meet three criteria:

  1) **Frame must be forwarded; one station to another.**
  2) **The forwarding table is automatically made by learning frame movements in the network.**
  3) **Loops in the system must be prevented.**

Learning BRIDGE

- A sends to D



Address	Port

a. Original

Address	Port
A	1

b. After A sends a frame to D

Address	Port
A	1
E	3

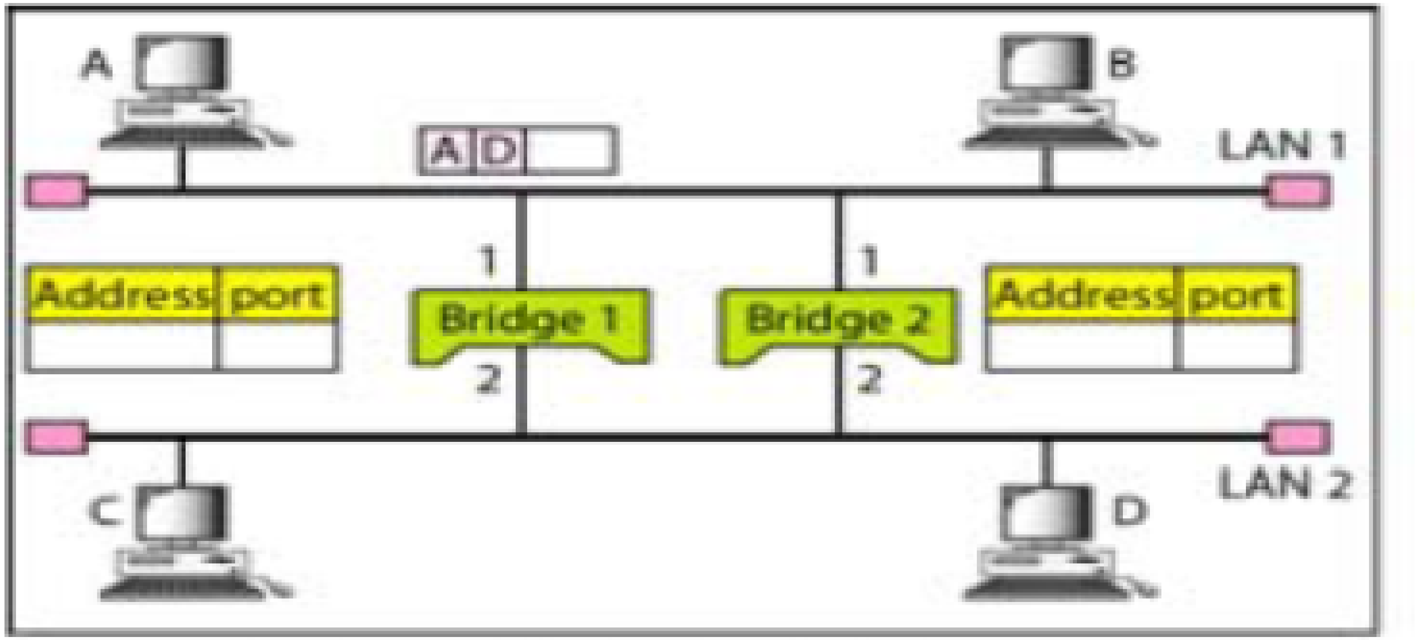
c. After E sends a frame to A

Address	Port
A	1
E	3
B	1

d. After B sends a frame to C

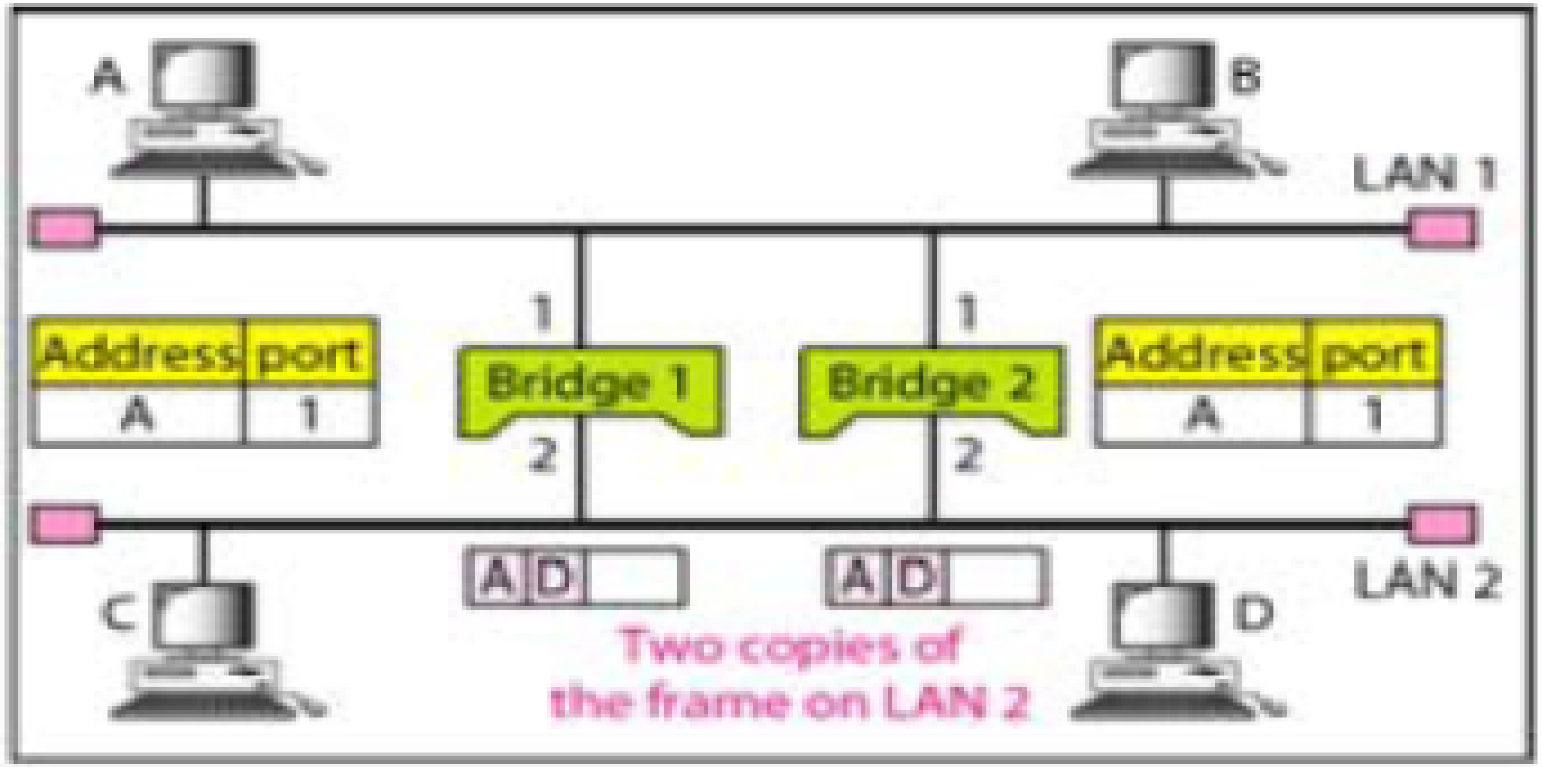
LOOPING

bridges are normally installed redundantly to make the system more reliable;
if a two LANs are connected by more than one bridge they may create a loop.



a. Station A sends a frame to station D

LOOPING-contd

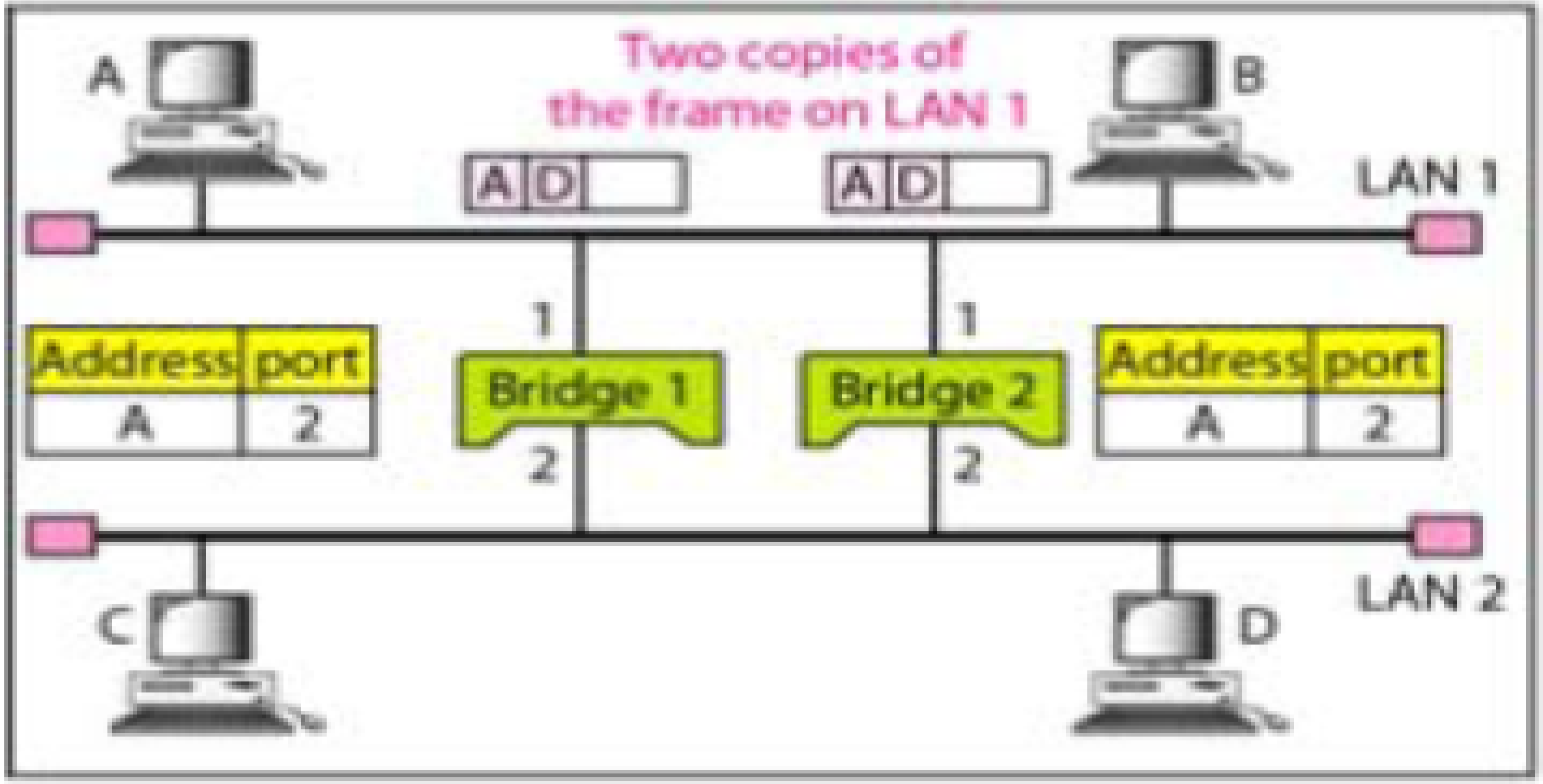


Address	port
A	1

Address	port
A	1

b. Both bridges forward the frame

LOOPING-contd

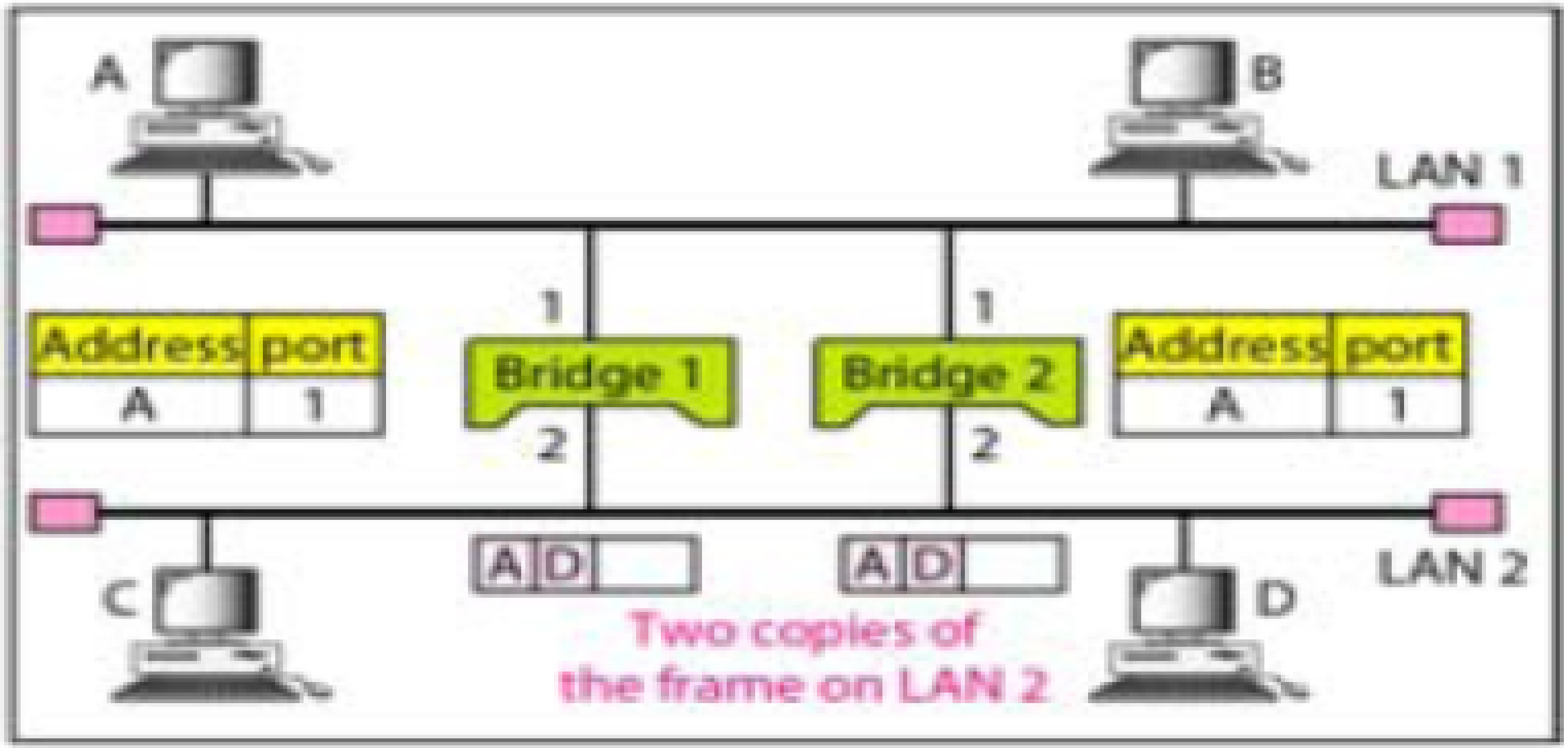


Address	port
A	2

Address	port
A	2

c. Both bridges forward the frame

LOOPING-contd



Address	port
A	1

Address	port
A	1

d. Both bridges forward the frame

Loop Resolving

- The simple learning mechanism described fails in presence of loops in the LAN
- Loops may be present by mistake, or deliberately provided for redundancy
- This problem is resolved by running a distributed spanning tree algorithm

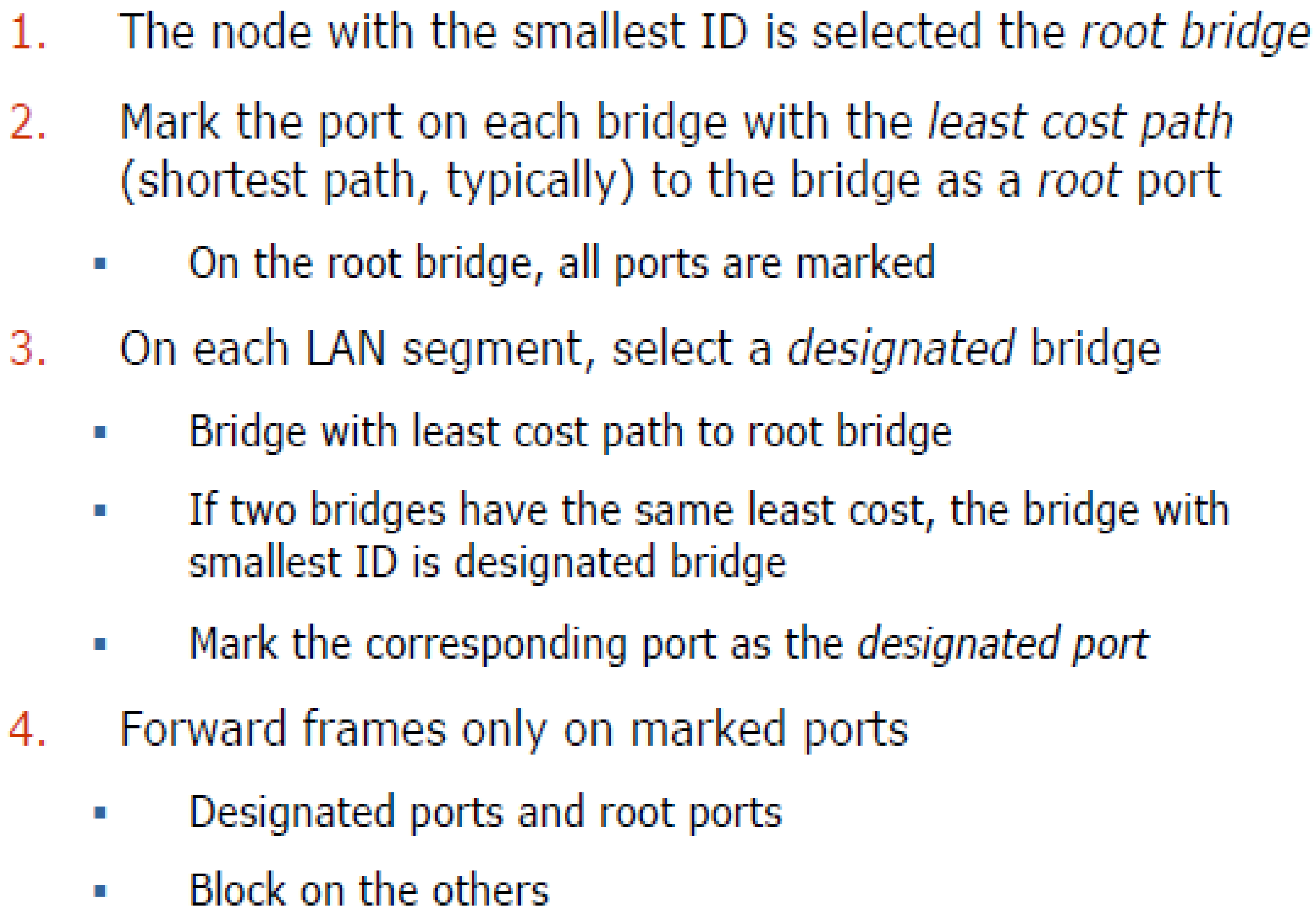
Spanning tree

- Think of the LAN as a graph that possibly has loops (LAN segments as nodes, bridges as edges)
- The spanning tree is a subgraph of this graph that covers all vertices (LAN segments), but contains no cycles.
- A graph in which there is no loop
- Create a topology in which each LAN can be reached from any other LAN
- through one path only (no loop)
- Create a logical topology that overlays physical topology which can not be changed

Spanning tree algorithm

- – Spanning tree algorithm is a protocol used by a set of bridges to agree upon a spanning tree for a particular extended LAN.
- – Essentially, this means that each bridge decides the ports over which it is and is not willing to forward packets.
- – Some ports (or even entire bridges) may not participate in a spanning tree
- – How does the bridge select the ports to include (/exclude)?

Spanning Tree Algorithm



1. The node with the smallest ID is selected the *root bridge*
2. Mark the port on each bridge with the *least cost path* (shortest path, typically) to the bridge as a *root* port
   - On the root bridge, all ports are marked
3. On each LAN segment, select a *designated* bridge
   - Bridge with least cost path to root bridge
   - If two bridges have the same least cost, the bridge with smallest ID is designated bridge
   - Mark the corresponding port as the *designated port*
4. Forward frames only on marked ports
   - Designated ports and root ports
   - Block on the others

STA-contd

To find the spanning tree

Assign a cost (metric) to each arc according to:

- Minimum hops,
- Minimum delay, or
- minimum bandwidth

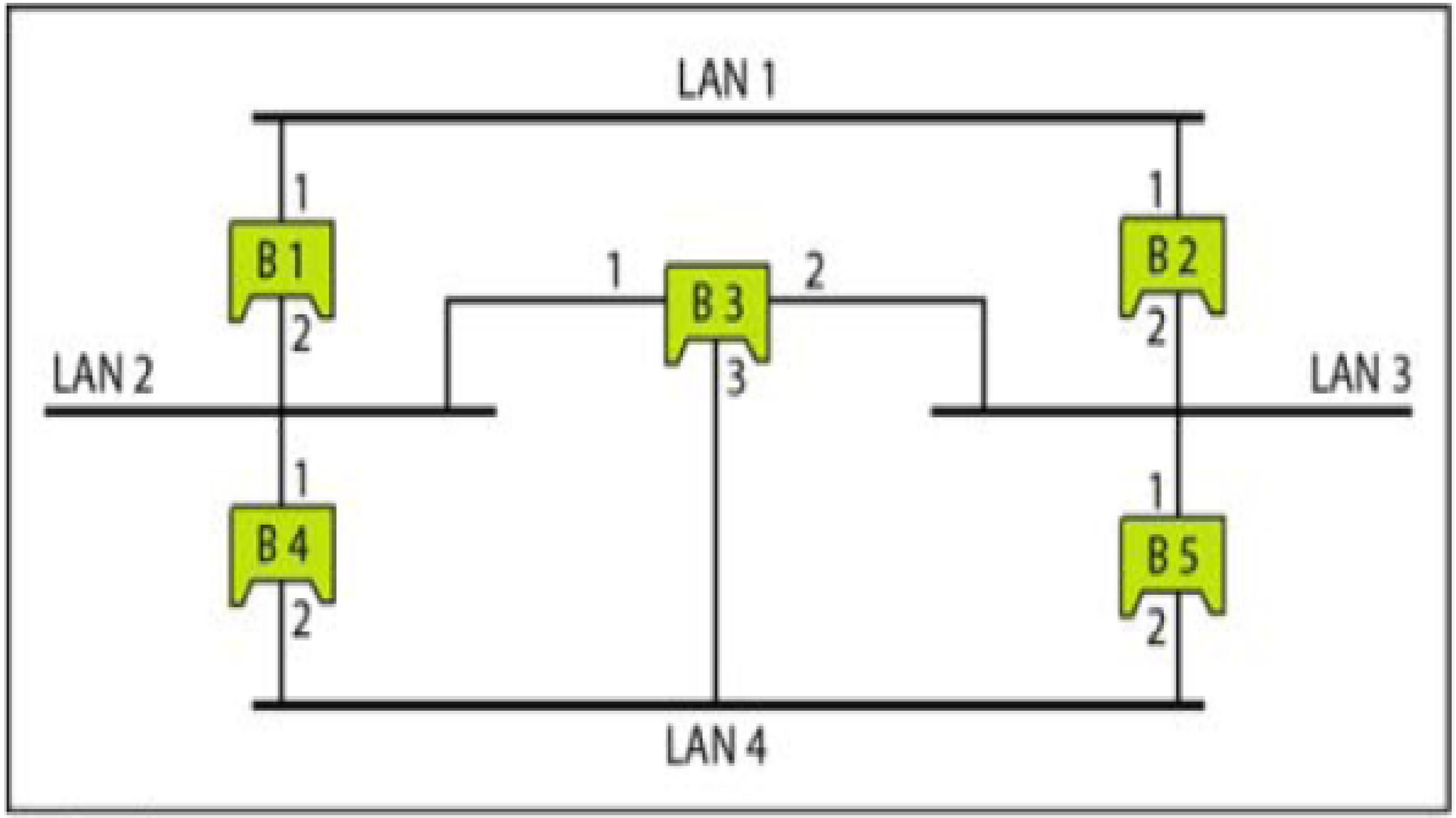
Minimum hops

The hop count is normally 1 from a bridge to the LAN and 0 in the reverse direction

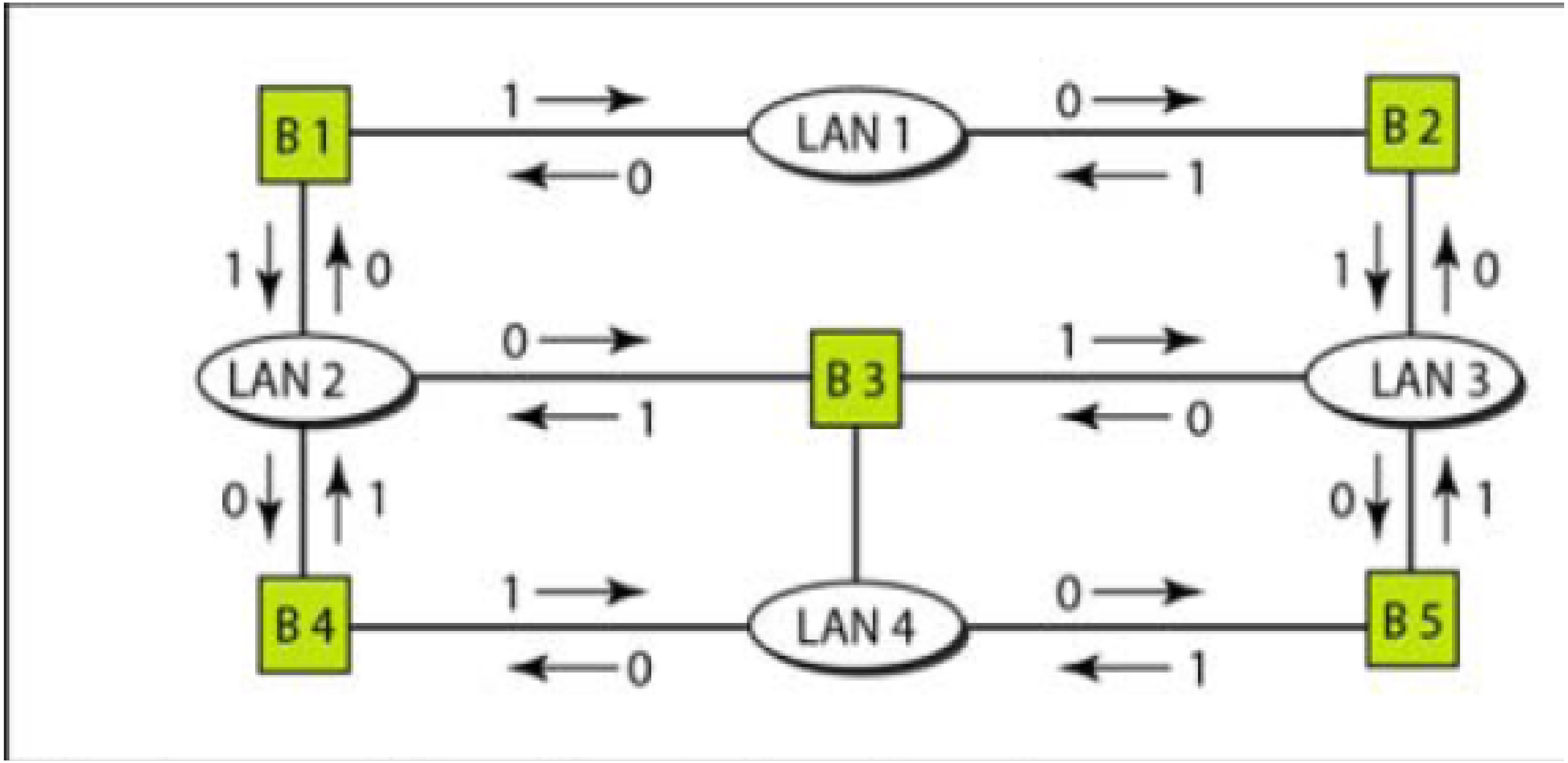
Bridge – ▬

LAN- ▬

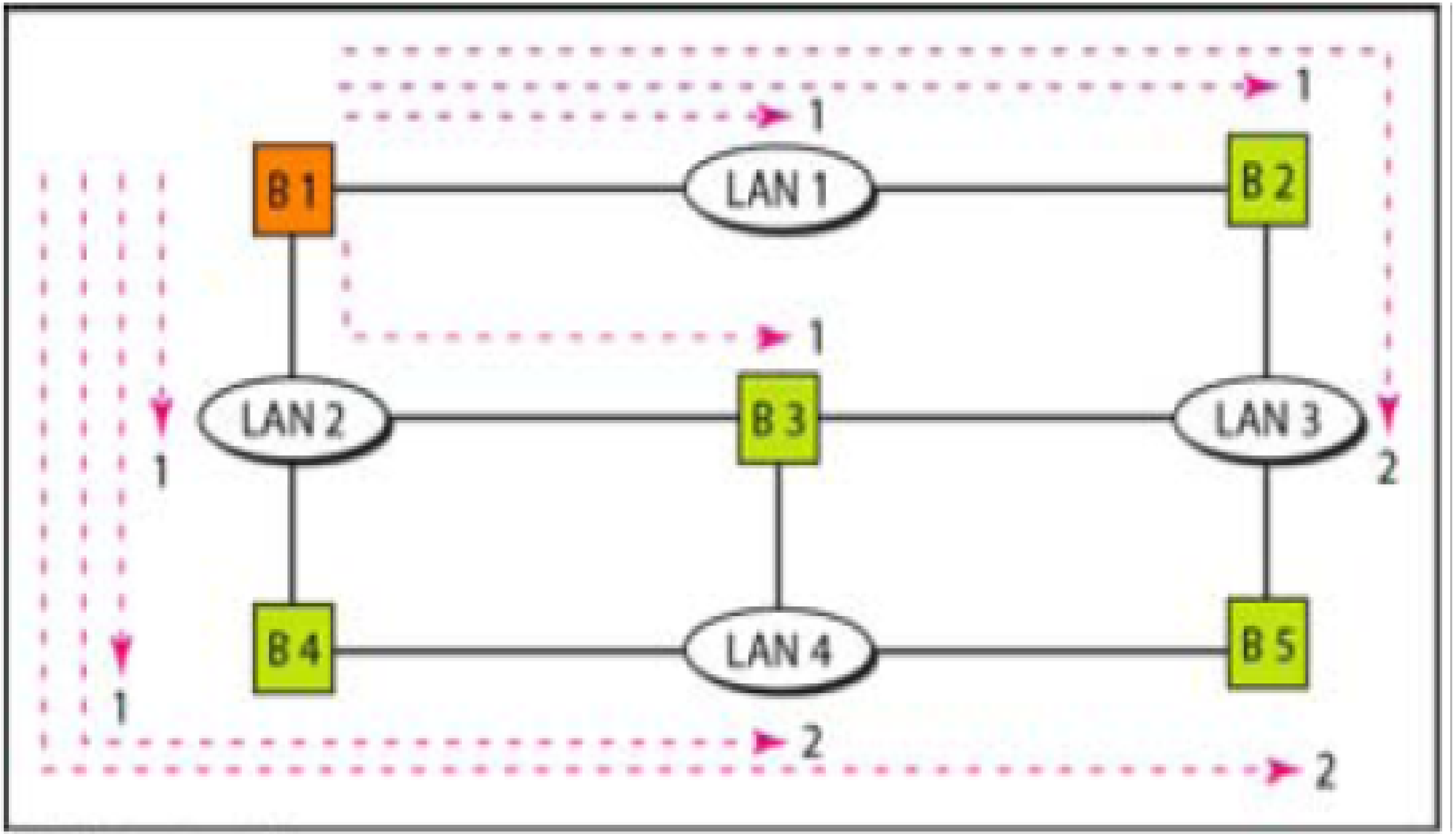
Example



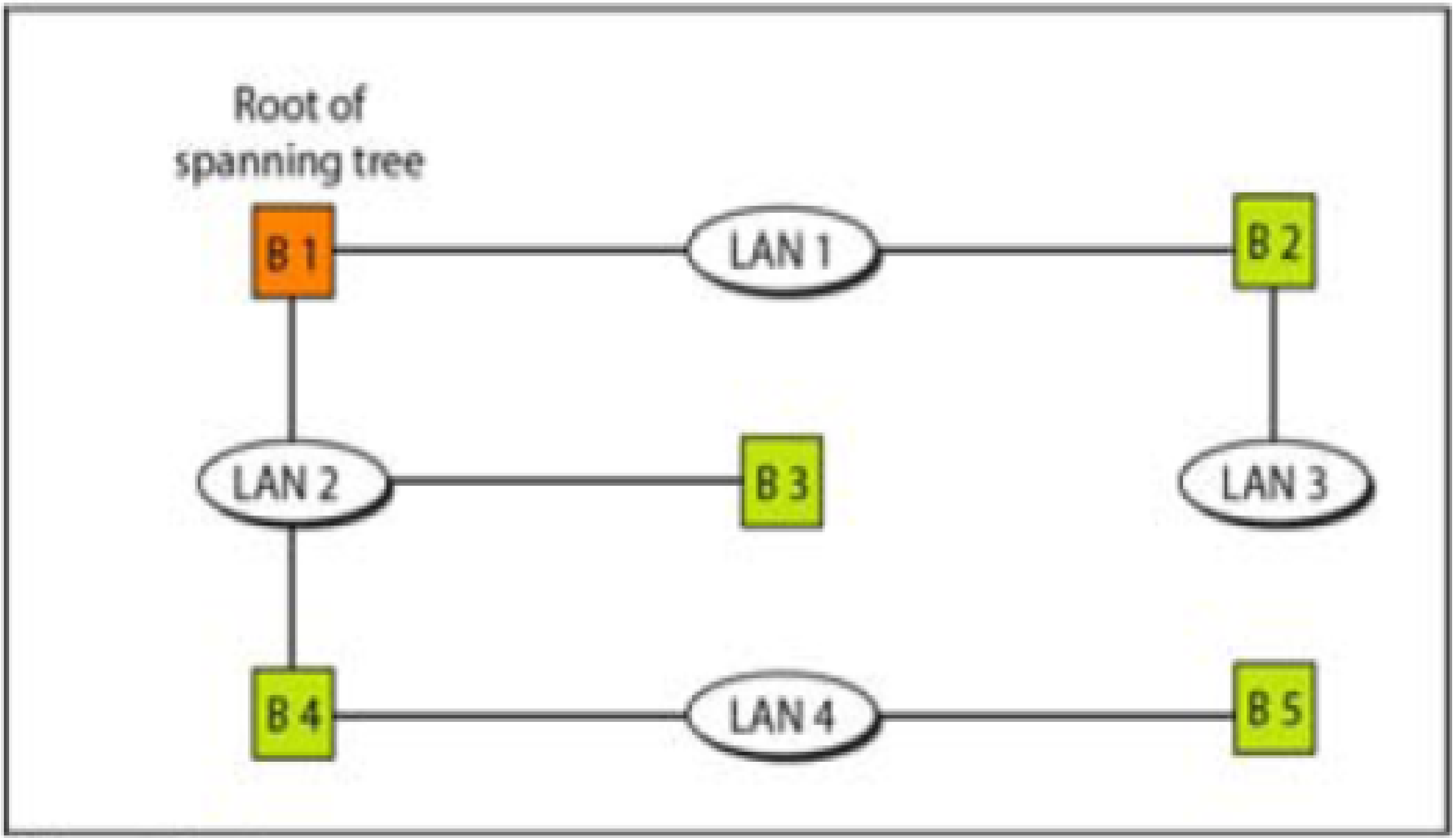
Assigning cost



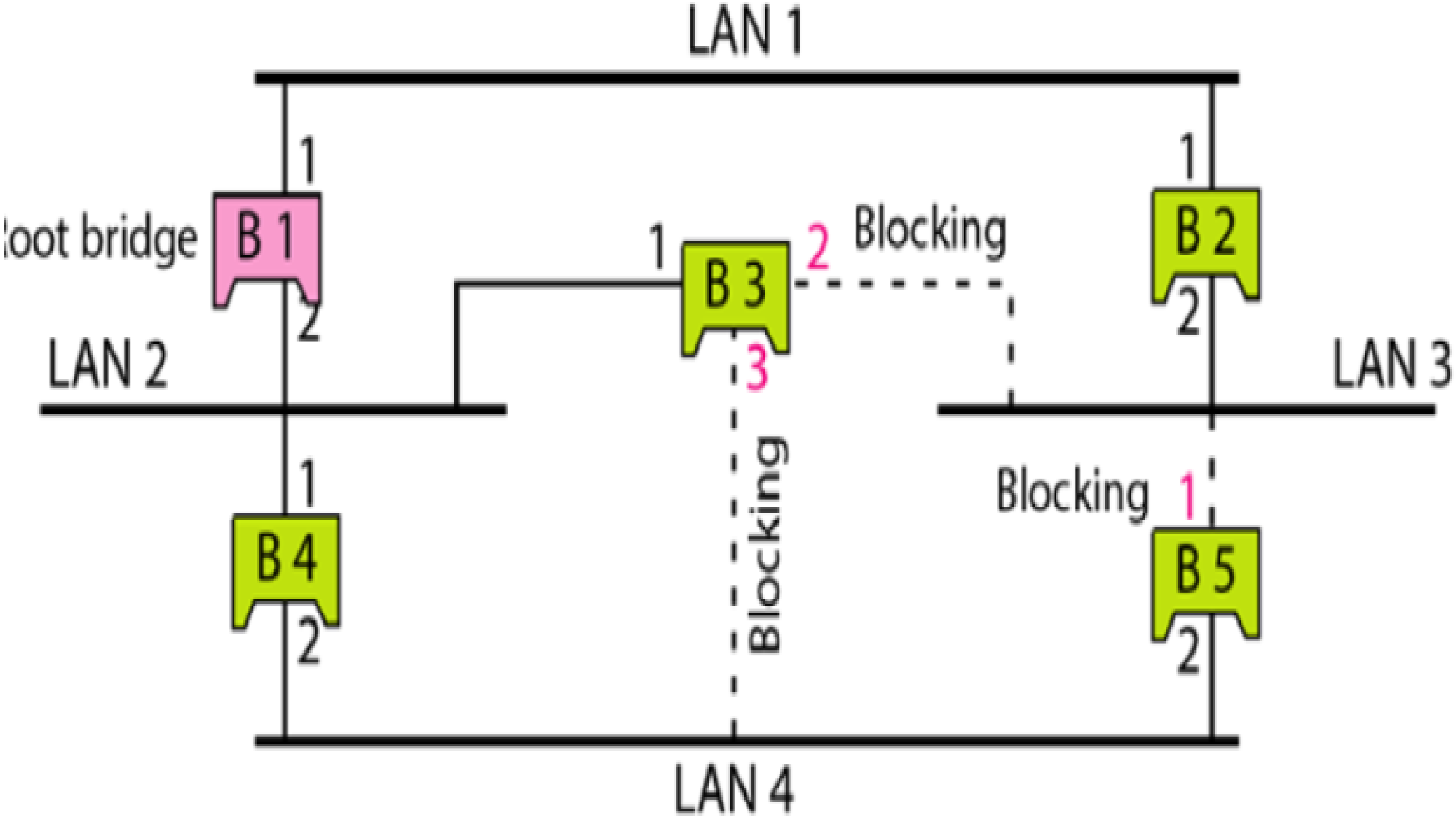
Finding the shortest paths



Finding the spanning tree

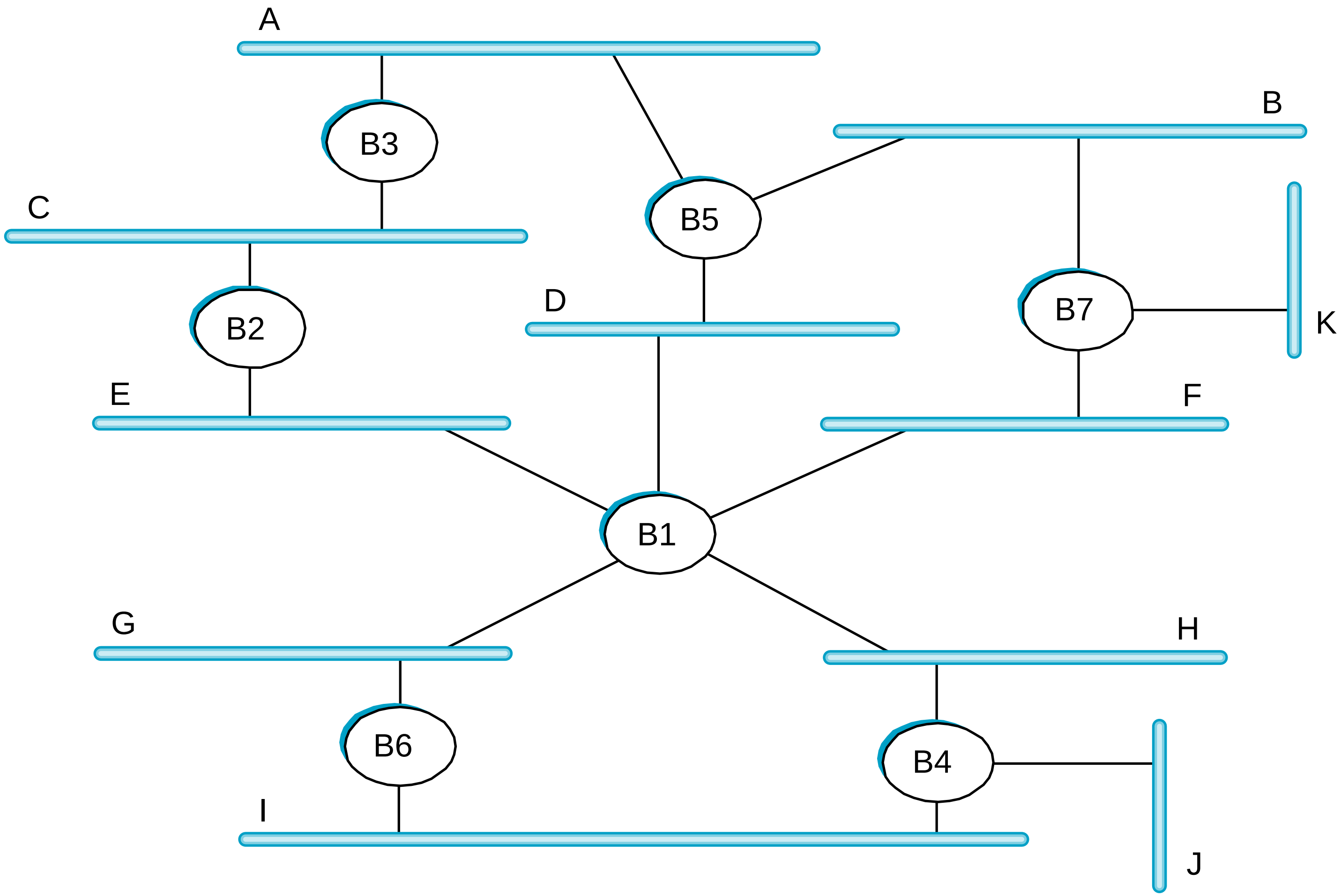


Forwarding and blocking ports



Ports 2 and 3 of bridge B3 are blocking ports (no frame is sent out of these ports). Port 1 of bridge B5 is also a blocking port (no frame is sent out of this port).

Exercise



Source Routing Bridges

Another way **to prevent loops** in a system.

- Sending station defines the bridges that the frame must visit.
- The addresses of these bridges are included in the frame.
- The frame contain the source and destination address,and the address of all the bridges to be visited
- Used with Token Ring LANs (not very common)