HUBS, SWITCHES AND BRIDGES

CPSC 441 TUTORIAL – MARCH 21, 2012
TA: RUITING ZHOU

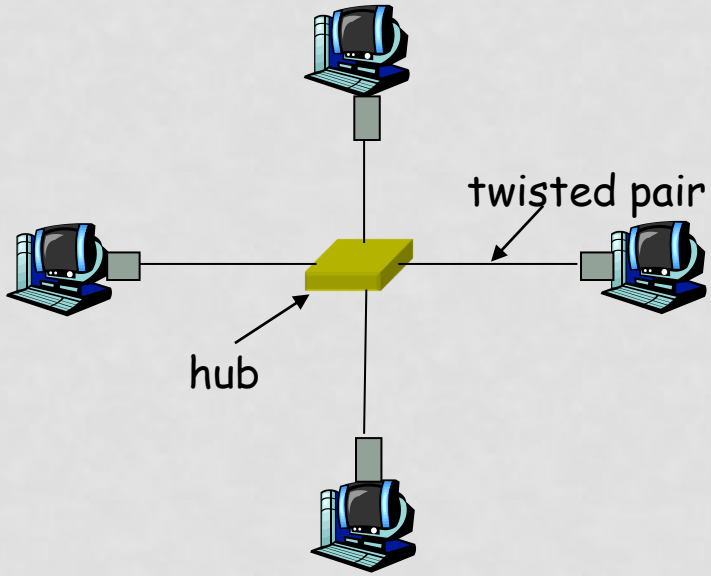
Parts of the slides contents are courtesy of the following people:
Jim Kurose, Keith Ross: http://www.aw-bc.com/kurose_ross/
Yishay Mansour: http://www.cs.tau.ac.il/~mansour/networking-course/lcc3.ppt

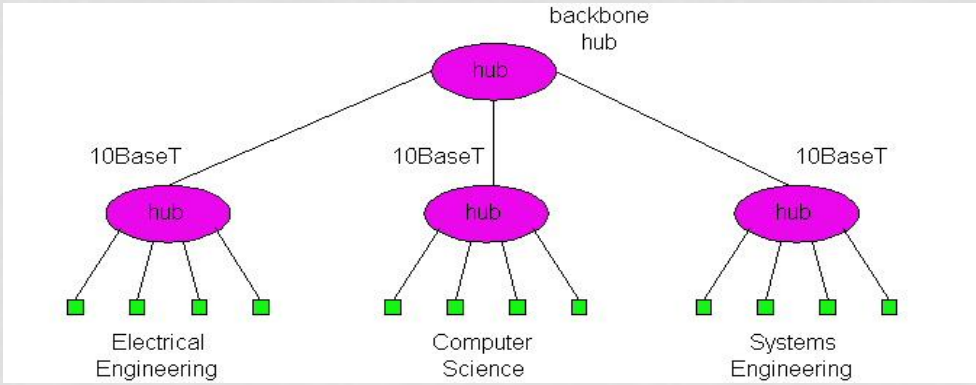
LAN INTERCONNECTION

- We need to break down big networks to sub-LANs
  - Limited amount of supportable traffic: on single LAN, all stations must share bandwidth
  - Limited length: 802.3 (Ethernet) specifies maximum cable length. For 10 Mbps:
    - Maximum length of the wire: 2,500 meter
  - Large "collision domain" (can collide with many stations)

HUBS

- **Physical Layer** devices
- Essentially repeaters operating at bit levels: repeat received bits on one interface to all other interfaces
- Hubs can be arranged in a hierarchy (or multi-tier design), with backbone hub at its top
- Each connected LAN referred to as LAN segment





HUBS: PROS

- Hub Advantages:
  - simple, inexpensive device
  - Multi-tier provides graceful degradation: portions of the LAN continue to operate if one hub malfunctions
  - extends maximum distance between node pairs (100m per Hub)

- limitations : Hubs do not isolate collision domains: node may collide with any node residing at any segment in LAN
  - Single collision domain results in no increase in max throughput
  - multi-tier throughput same as single segment throughput
  - Individual LAN restrictions pose limits on number of nodes in same collision domain and on total allowed geographical coverage
  - cannot connect different Ethernet types (e.g., 10BaseT and 100baseT) Why?

BRIDGES

- **Link-layer** devices:
  - store, forward Ethernet frames
  - examine incoming frame's MAC address, selectively forward frame based on its destination. When frame is to be forwarded on segment, bridge uses CSMA/CD to access segment and transmit
- Advantages:
  - Isolates collision domains resulting in higher total max throughput, and does not limit the number of nodes nor geographical coverage
  - Can connect different type Ethernet since it is a store and forward device
  - Transparent: no need for any change to hosts LAN adapters

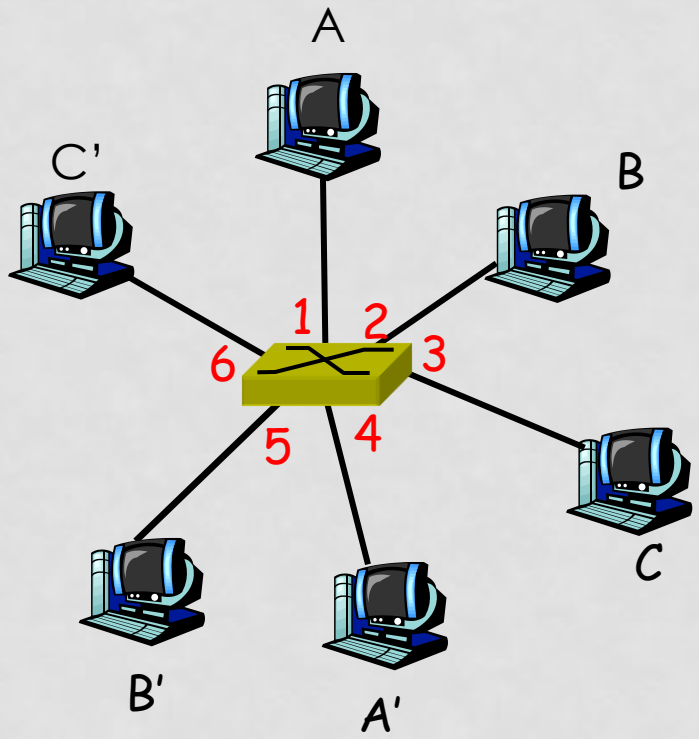
SWITCHES

- *A switch could be considered a bridge with numerous ports.* A bridge only has one incoming and one outgoing port.
- *Switch* or *Layer 2 switch* is often used interchangeably with *bridge*

- *Plug-and-play, self-learning*
  - switches do not need to be configured

SWITCH: ALLOWS *MULTIPLE* SIMULTANEOUS TRANSMISSIONS

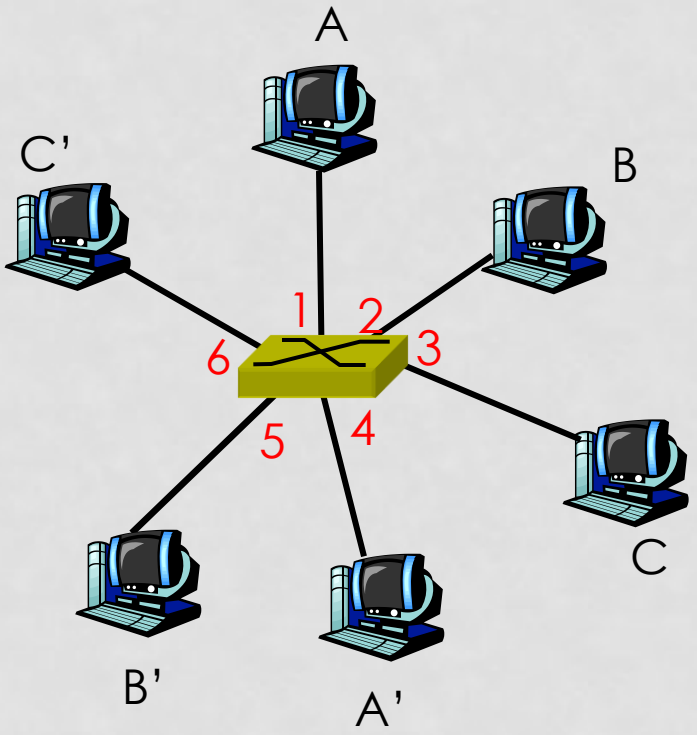
- hosts have dedicated, direct connection to switch
- switches buffer packets
- Ethernet protocol used on *each* incoming link, but no collisions; full duplex
  - each link is its own collision domain
- *switching:* A-to-A' and B-to-B' simultaneously, without collisions
  - not possible with dumb hub



switch with six interfaces (1,2,3,4,5,6)

SWITCH TABLE

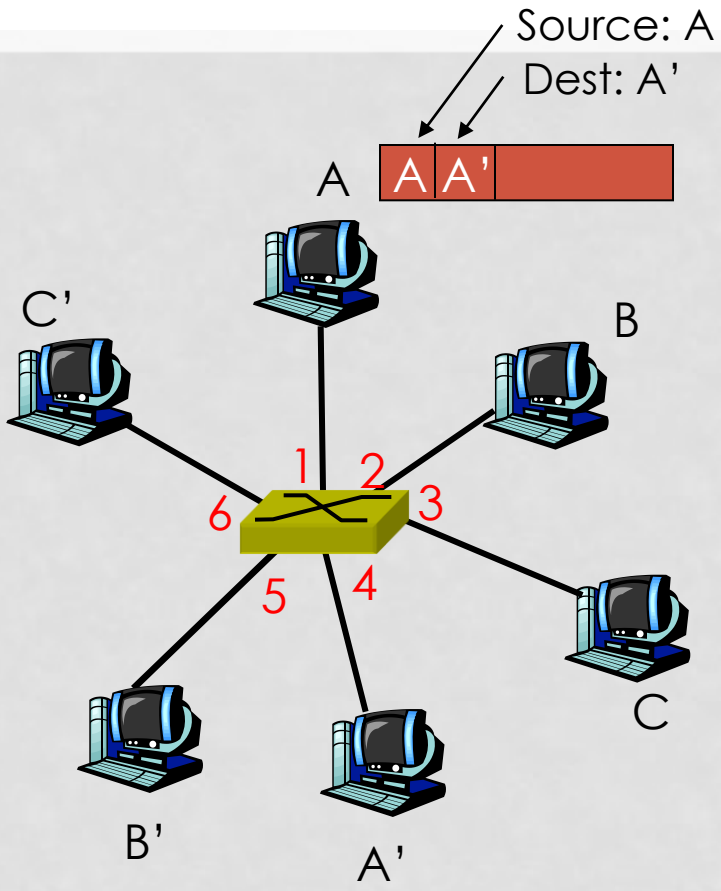
- Q: how does switch know that A' reachable via interface 4, B' reachable via interface 5?
- A: each switch has a switch table, each entry:
  - (MAC address of host, interface to reach host, time stamp)
- looks like a routing table!
- Q: how are entries created, maintained in switch table?
  - something like a routing protocol?



switch with six interfaces (1,2,3,4,5,6)

SWITCH: SELF-LEARNING

- switch *learns* which hosts can be reached through which interfaces
  - when frame received, switch "learns" location of sender: incoming LAN segment
  - records sender/location pair in switch table



MAC addr	interface	TTL
A	1	60

Switch table
(initially empty)

SWITCH: FRAME FILTERING/FORWARDING

When frame received:

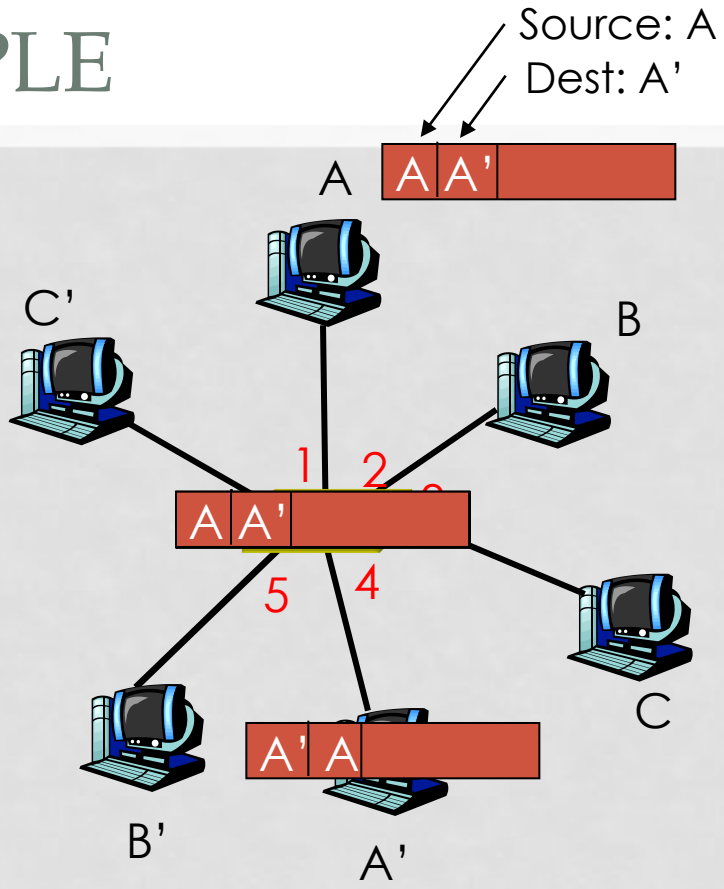
1. record link associated with sending host
2. index switch table using MAC dest address
3. **if** entry found for destination
   then {
   if dest on segment from which frame arrived
   then drop the frame
   else forward the frame on interface indicated
   }
   else flood

forward on all but the interface on which the frame arrived

SELF-LEARNING, FORWARDING: EXAMPLE

- frame destination unknown: flood
- destination A location known:

selective send

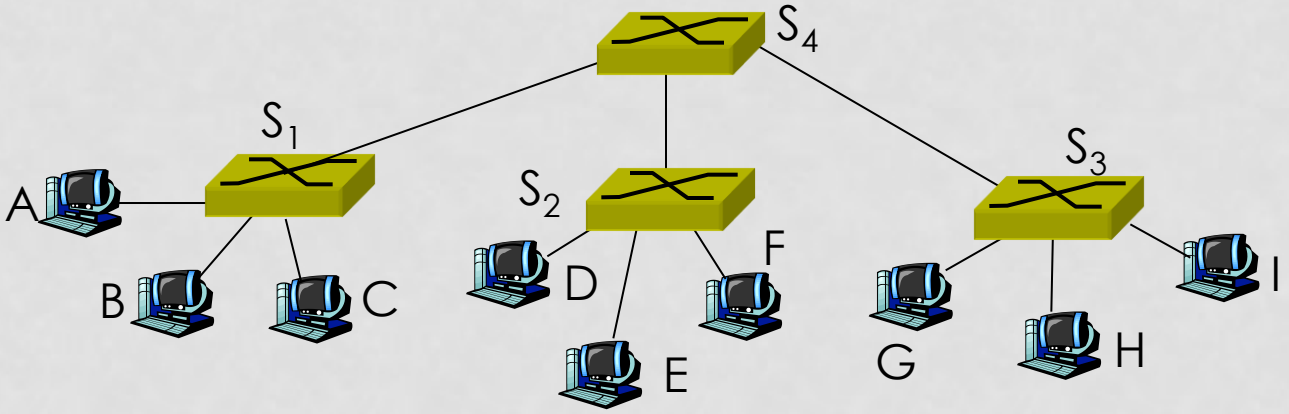


MAC addr	interface	TTL
A	1	60
A'	4	60

Switch table (initially empty)

INTERCONNECTING SWITCHES

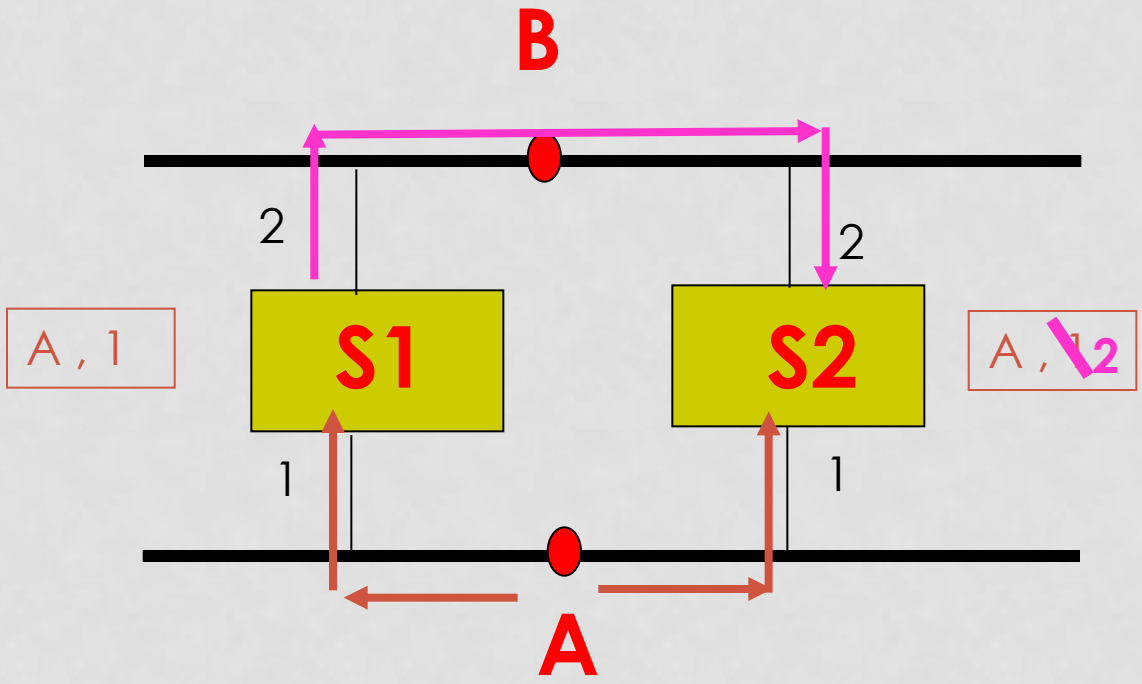
- switches can be connected together



- Q: sending from A to F - how does S_1 know to forward frame destined to F via S_4 and S_2?
- A: self learning! (works exactly the same as in single-switch case!)

WHAT WILL HAPPEN WITH LOOPS?

- *Incorrect learning*

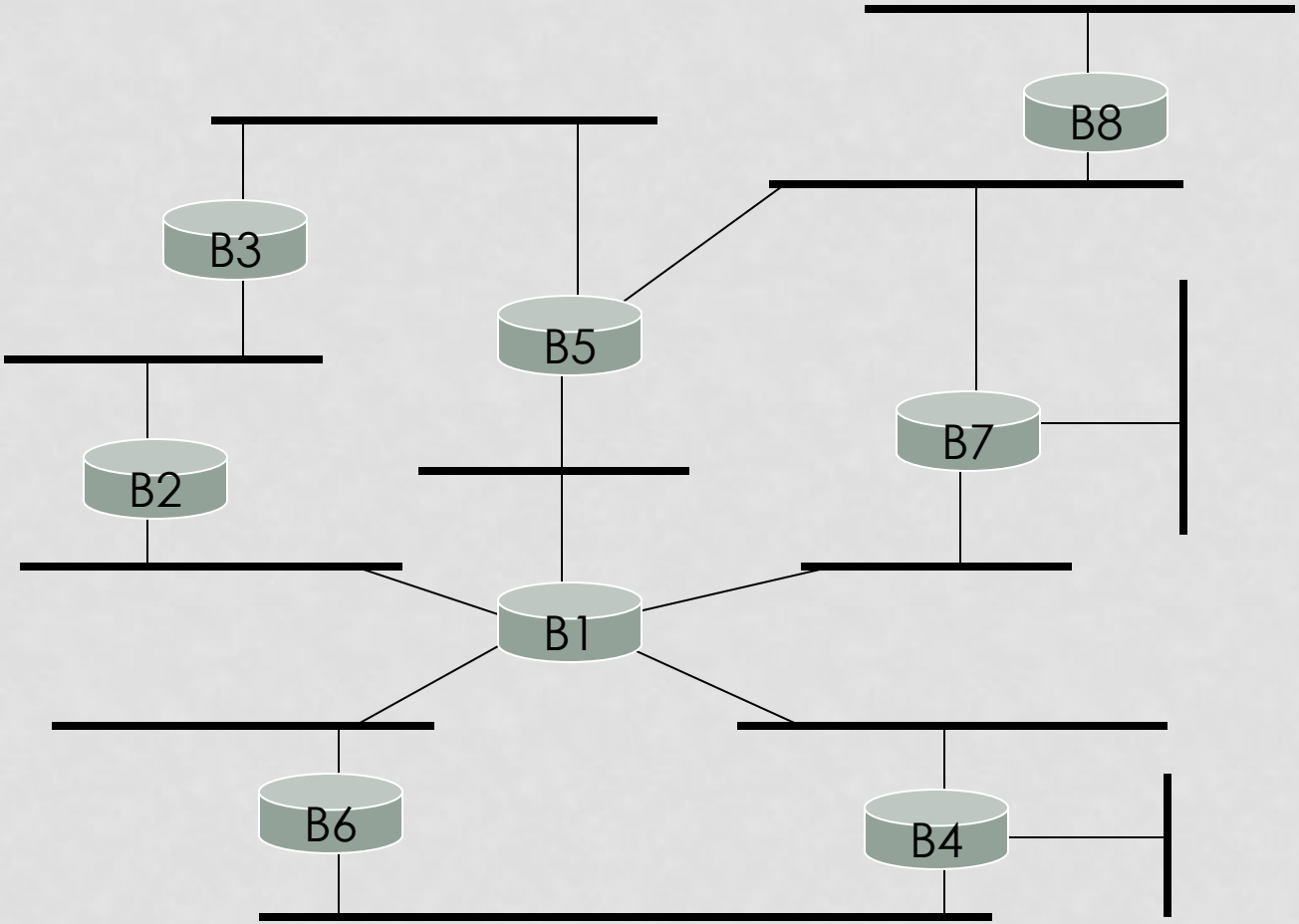


SPANNING TREES

- Allow a path between every LAN without causing loops (*loop-free environment*)
- Bridges communicate with special configuration messages (*BPDUs- Bridge Protocol Data Units*)
- Standardized by IEEE 802.1D

- Requirements:
  - Each bridge is assigned a unique identifier
  - A broadcast address for bridges on a LAN
  - A unique port identifier for all ports on all bridges
    - MAC address
    - Bridge id + port number

EXAMPLE SPANNING TREE



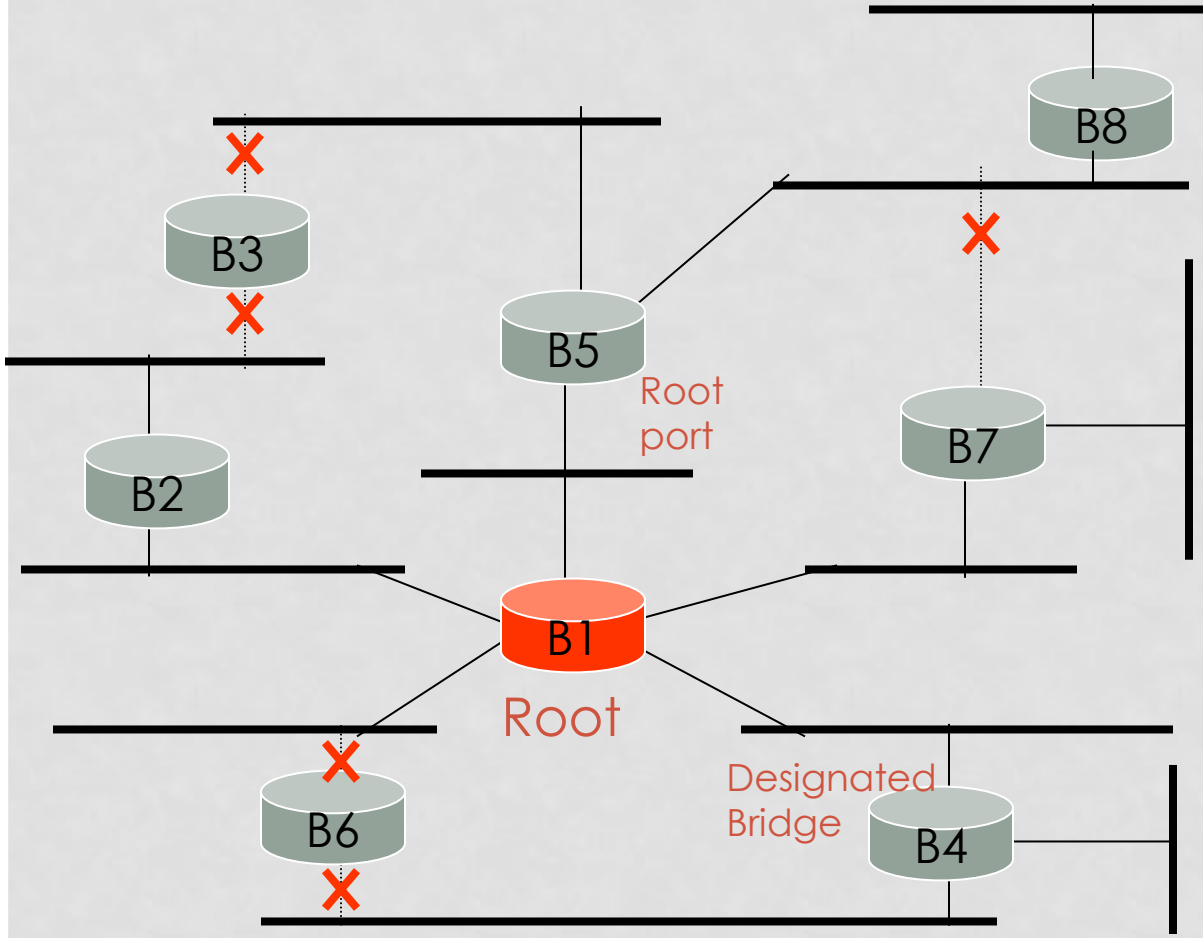
SPANNING TREE ALGORITHM: OVERVIEW

1. Determine the *root bridge* among all bridges

2. Each bridge determines its *root port*
   - The port in the direction of the root bridge

3. Determine the *designated bridge* on each LAN
   - The bridge which accepts frames to forward towards the root bridge
   - The frames are sent on the root port of the designated bridge

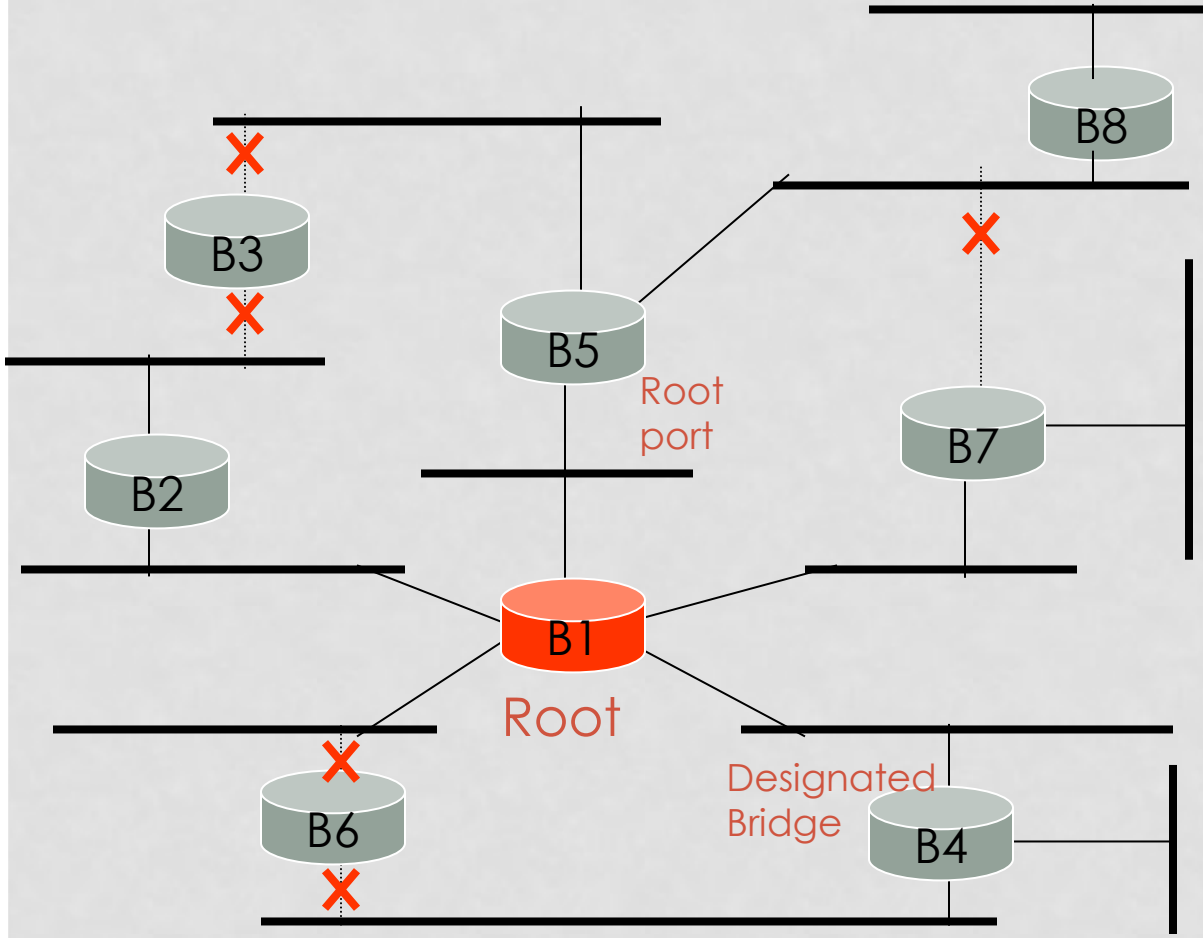
EXAMPLE SPANNING TREE



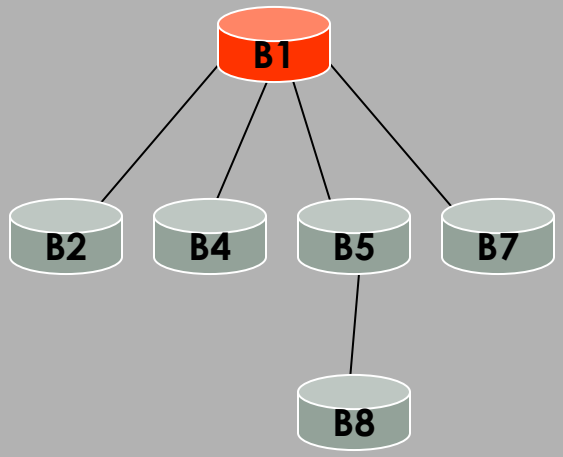
Protocol operation:

1. Picks a **root**
2. For each LAN, picks a **designated bridge** that is closest to the root.
3. All bridges on a LAN send packets towards the **root** via the **designated bridge**.

EXAMPLE SPANNING TREE



Spanning Tree:



SPANNING TREE ALGORITHM: SELECTING ROOT BRIDGE

- Initially, each bridge considers itself to be the root bridge
- Bridges send BDPU frames to its attached LANs
  - The bridge and port ID of the sending bridge
  - The bridge and port ID of the bridge--- the sending bridge considers the one is the root
  - The root path cost for the sending bridge
- Best one wins
  - (lowest root ID/cost/priority)

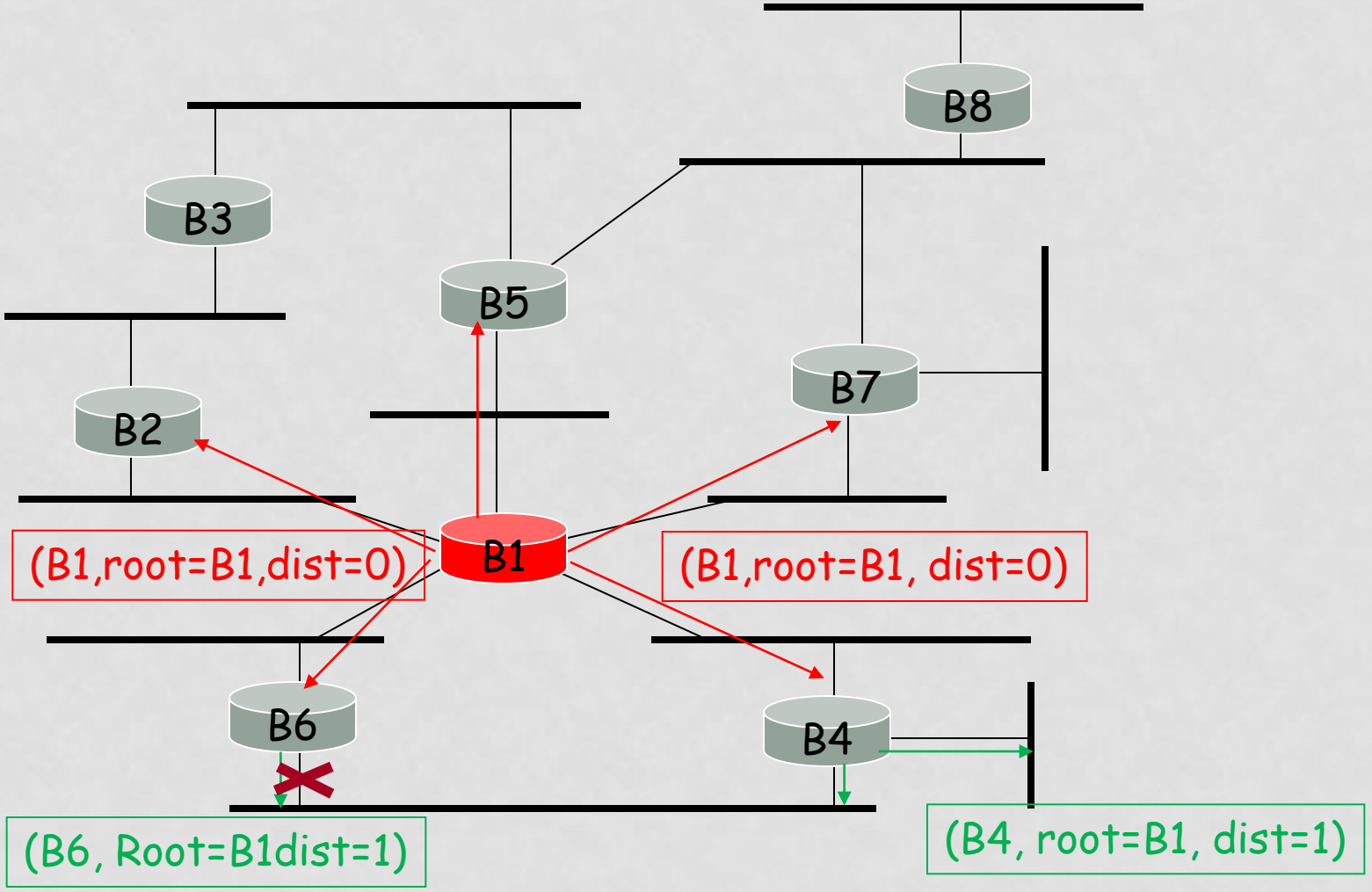
SPANNING TREE ALGORITHM: SELECTING ROOT PORTS

- Each bridge selects one of its ports which has the minimal cost to the root bridge
- When multiple paths from a bridge are least-cost paths, the chosen path uses the neighbor bridge with the lower bridge ID. The root port is thus the one connecting to the bridge with the lowest bridge ID.
- In case of another tie, two bridges are connected by multiple cables. In this case, the lowest port ID is used

SELECT DESIGNATED BRIDGES FORWARDING/BLOCKING STATE

- Same as selecting the root bridge:
- Initially, each bridge considers itself to be the designated bridge, send BDPU frames to attached LANs, best one wins!
- Root and designated bridges will *forward* frames to and from their attached LANs
- All other ports are in the *blocking* state

SPANNING TREE PROTOCOL: EXECUTION



SWITCHES VS. ROUTERS

- both store-and-forward devices
  - routers: network layer devices (examine network layer headers)
  - switches are link layer devices
- routers maintain routing tables, implement routing algorithms
- switches maintain switch tables, implement filtering, learning algorithms