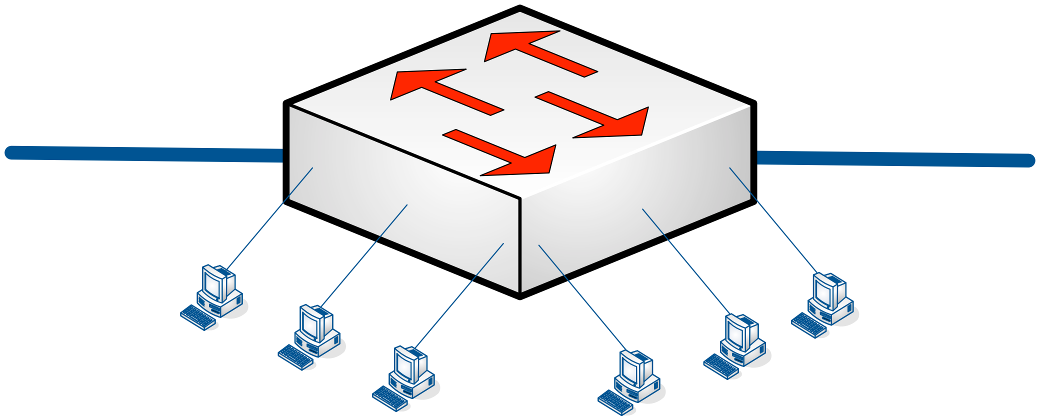
1

Switched LANs and Spanning Tree Protocol

Large LANs that offer redundancy and thereby avoid broadcast storms

LAN Switch

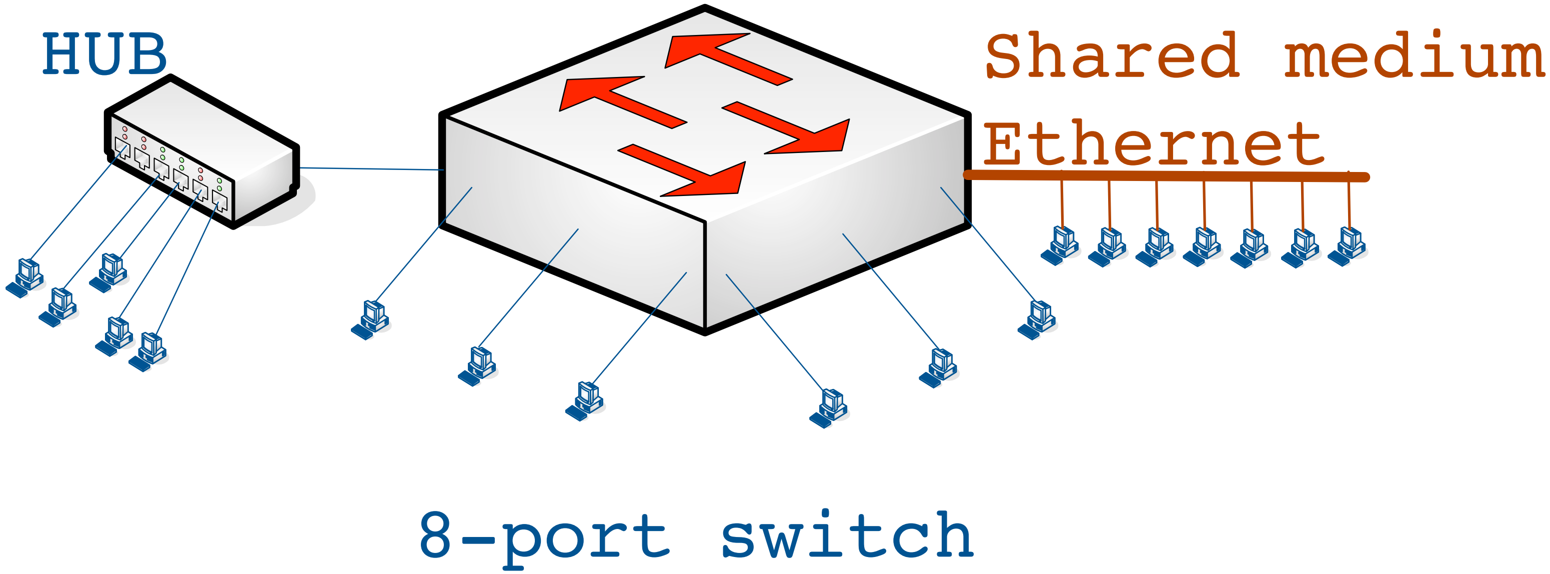
- S/F device
- 8 ports = 8 collision domains
- 1 broadcast domain = 1 Switched LAN
- Enlarge the Switched LAN, how?



8-port switch

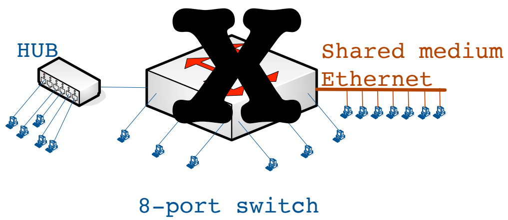
Enlarge the Switched LAN, how?

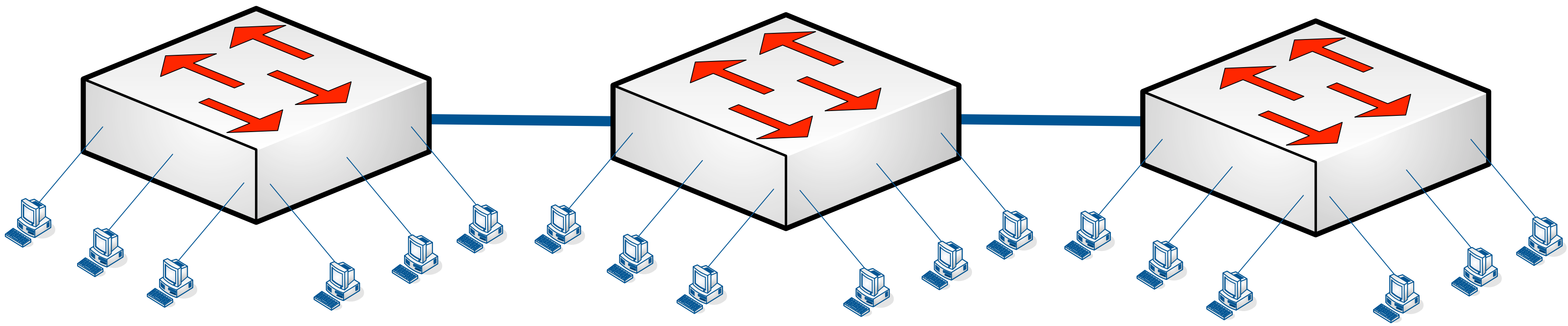
- Connect more hosts to each port
  - Recall: 1 port = 1 collision domain
  - Connect several stations to each port
  - Complex; damages performance; unreliable



Enlarge the Switched LAN, how?

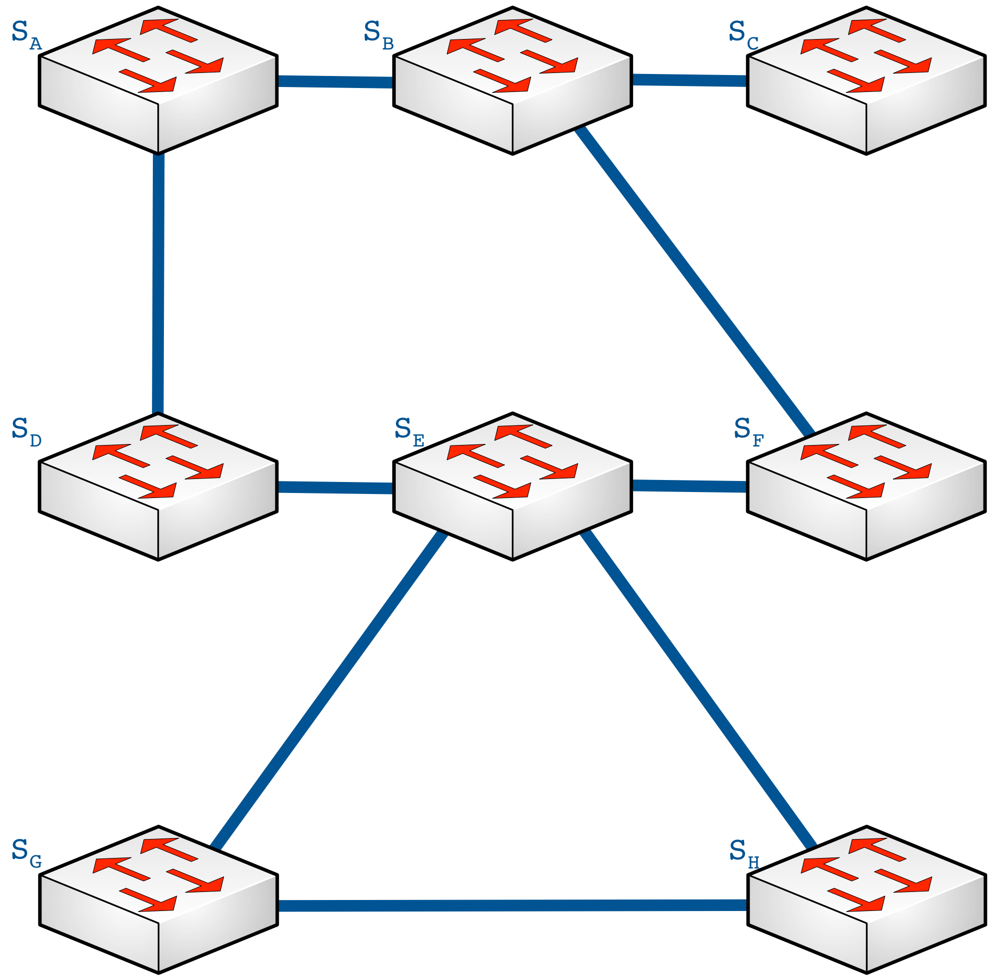
- Connect additional switches
- Preserves performance if well designed
- Even, create redundant connections
  - In case of link or cable faults





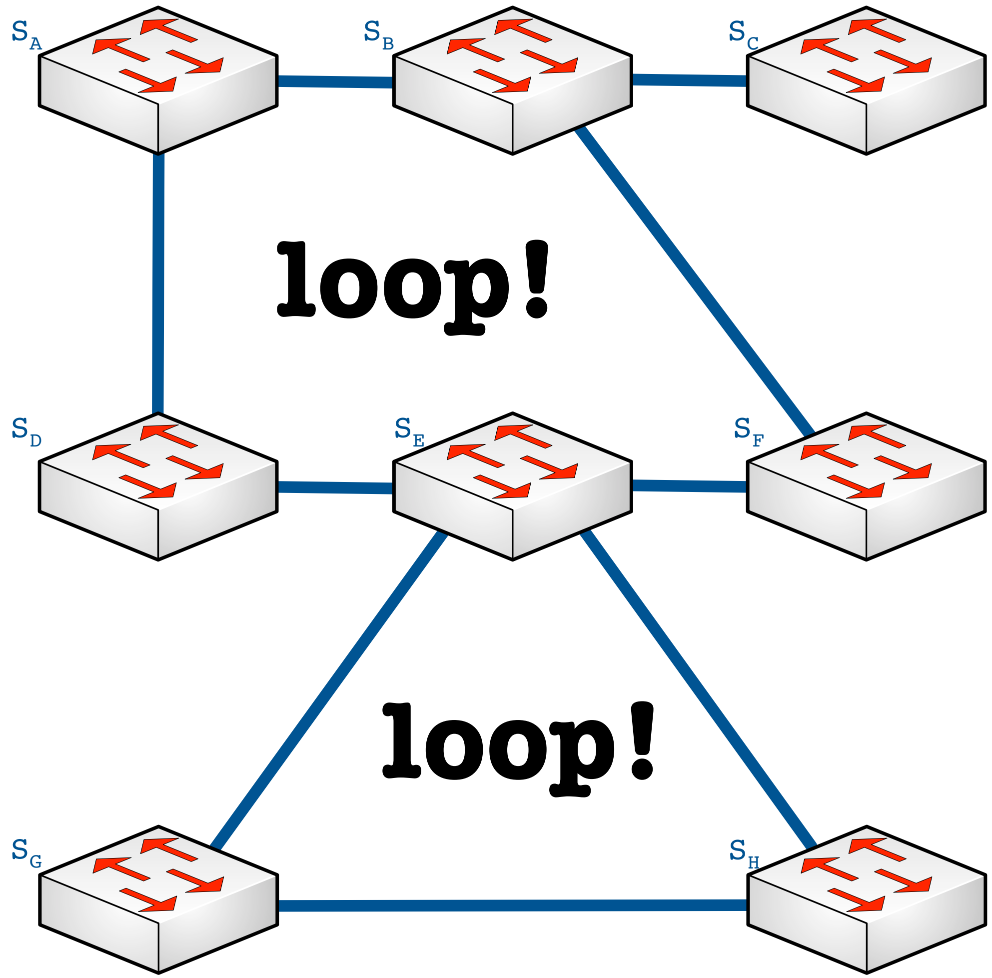
Enlarge the Switched LAN, how?

- Connect additional switches
- Preserves performance while increasing utilization
- Even, create redundant connections
  - Loops
  - Good for faults



Loops help with link faults

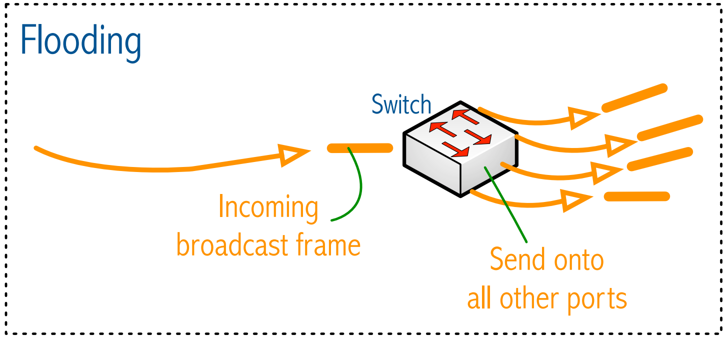
- Redundant paths
- Good when faults happen
- Drawbacks
  - Loops, when not properly managed, cause broadcast storms
  - Frames proliferate out of control
  - The whole switched LAN becomes useless
  - Hundreds of stations will not be able to communicate

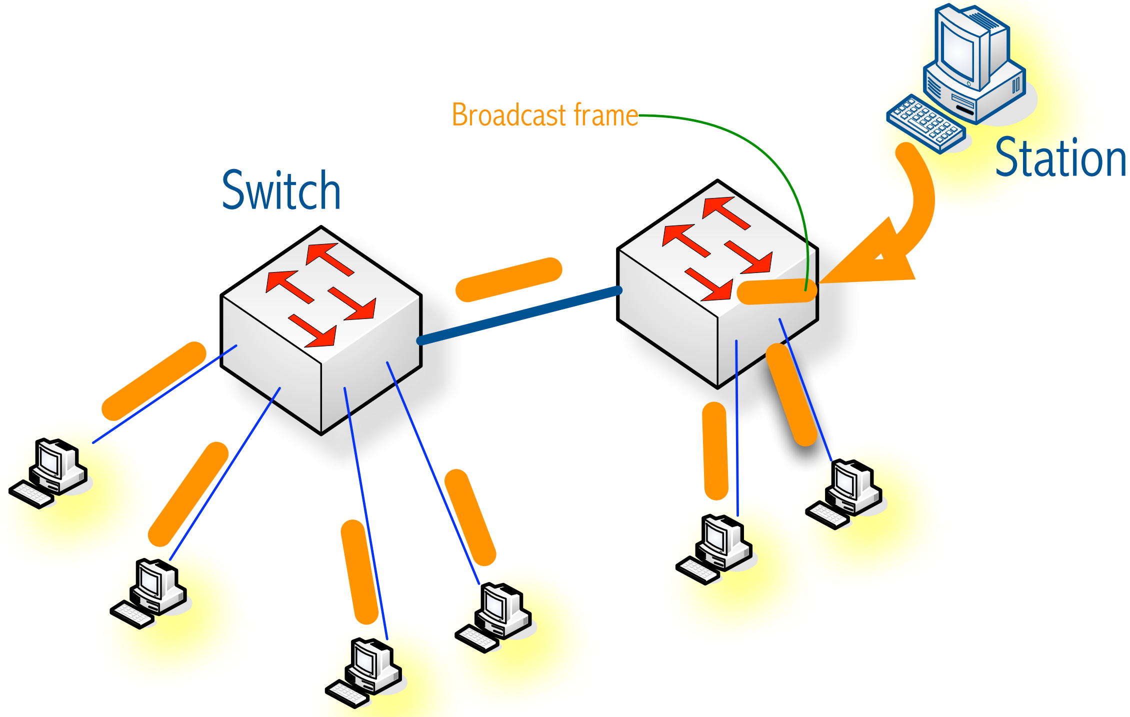


Review: LAN Switching Algorithm

- A switch floods every frame
  - Which Destination MAC is the broadcast address
  - Or, which destination address is not found in its FIB

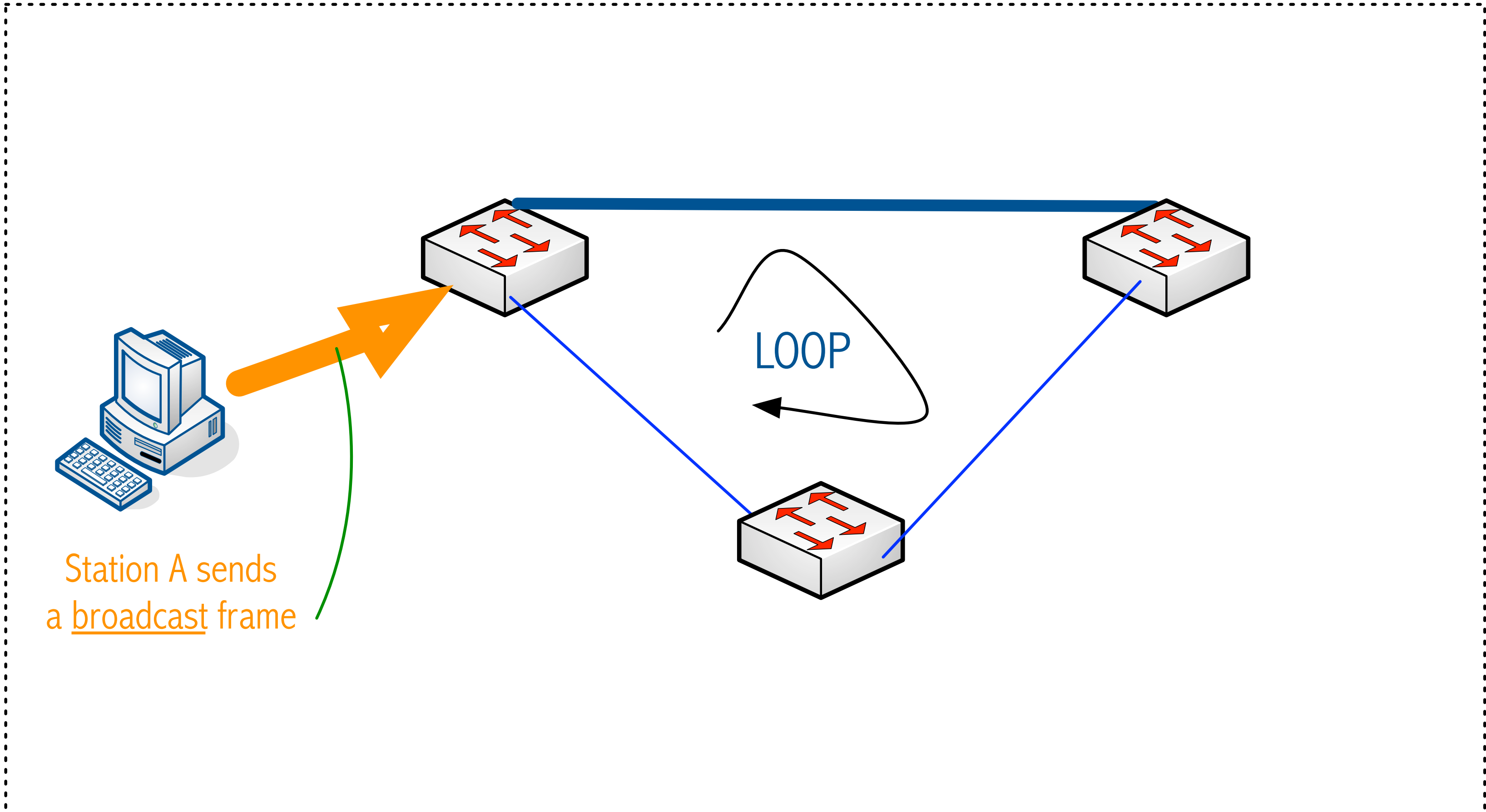
Essential!





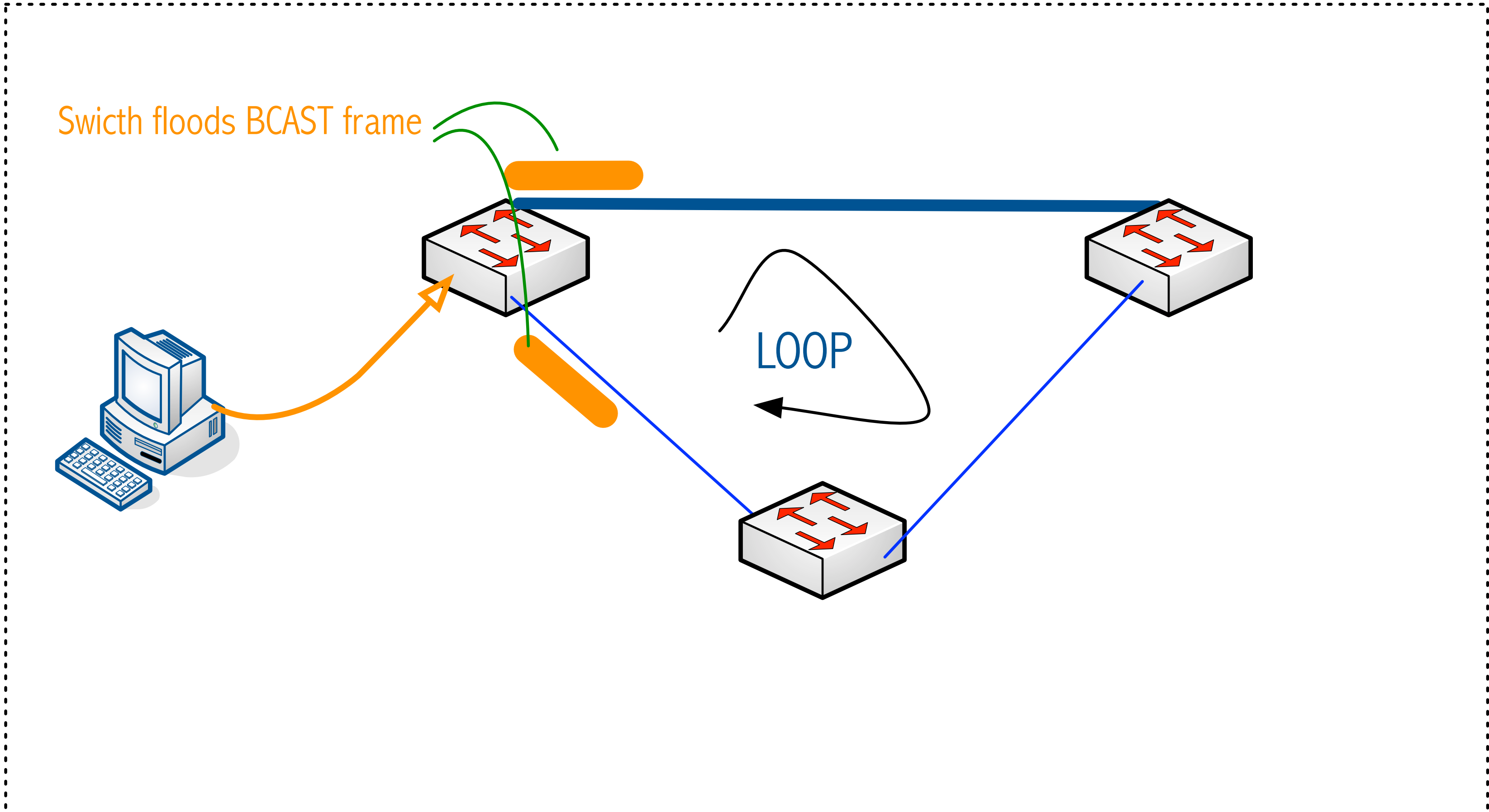
Switched LANs, broadcast and loops

- Station A sends a BROADCAST frame



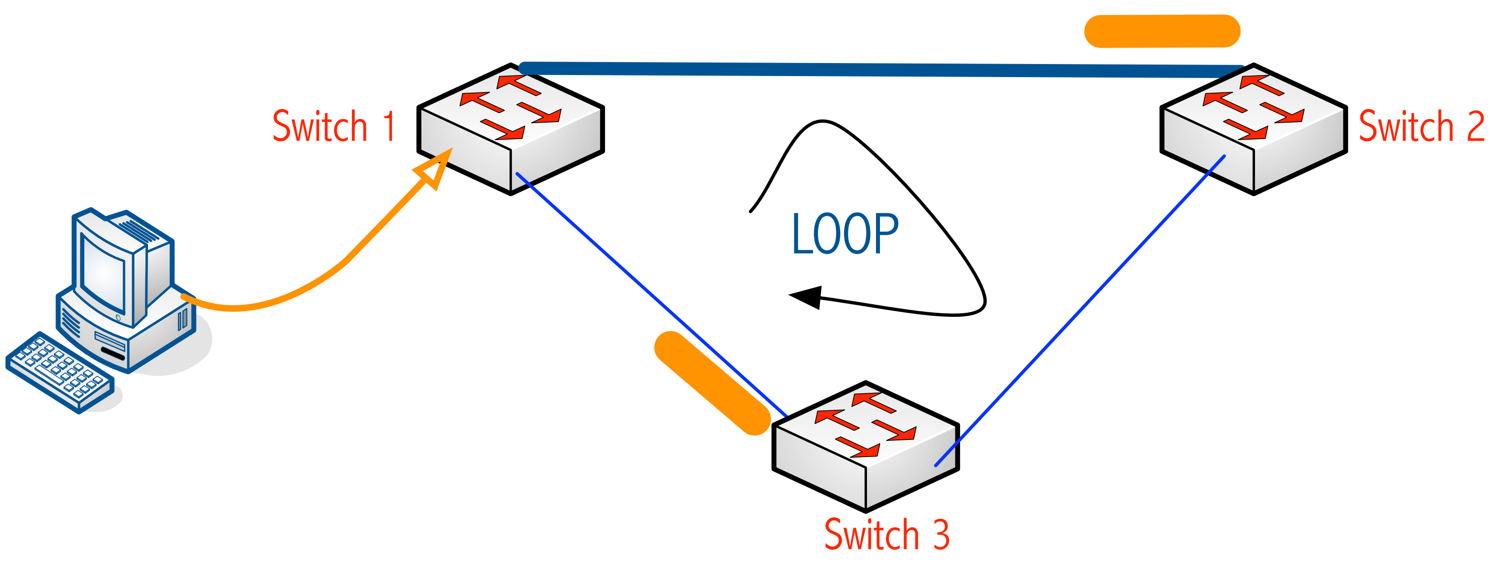
Switched LANs, broadcast and loops

- BCAST frame ingresses in switch
  - Switch will flood it: send it over all ports except the port over which it was received



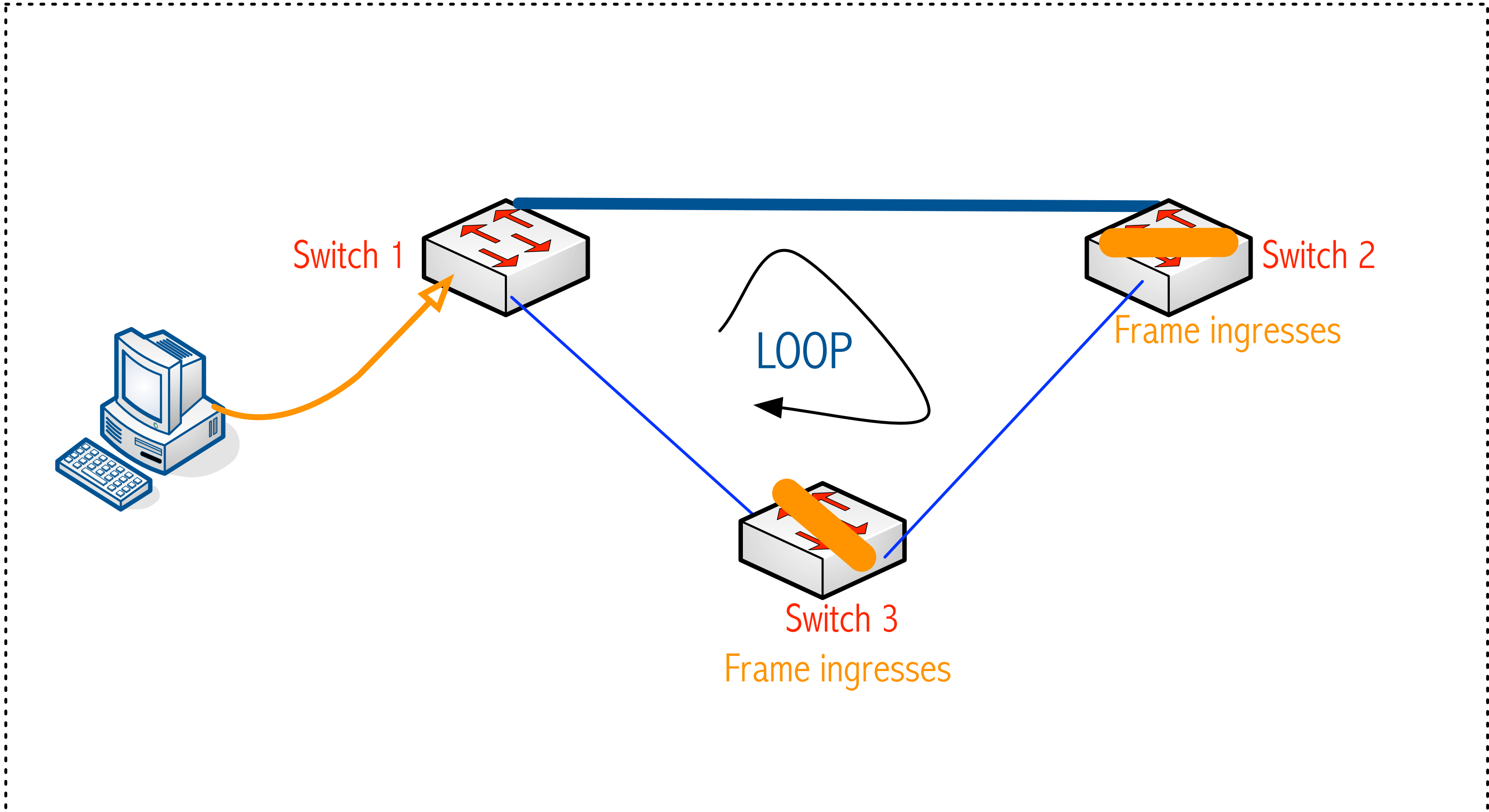
Switched LANs, broadcast and loops

- BCAST frame is received by Switch 2 and Switch 3



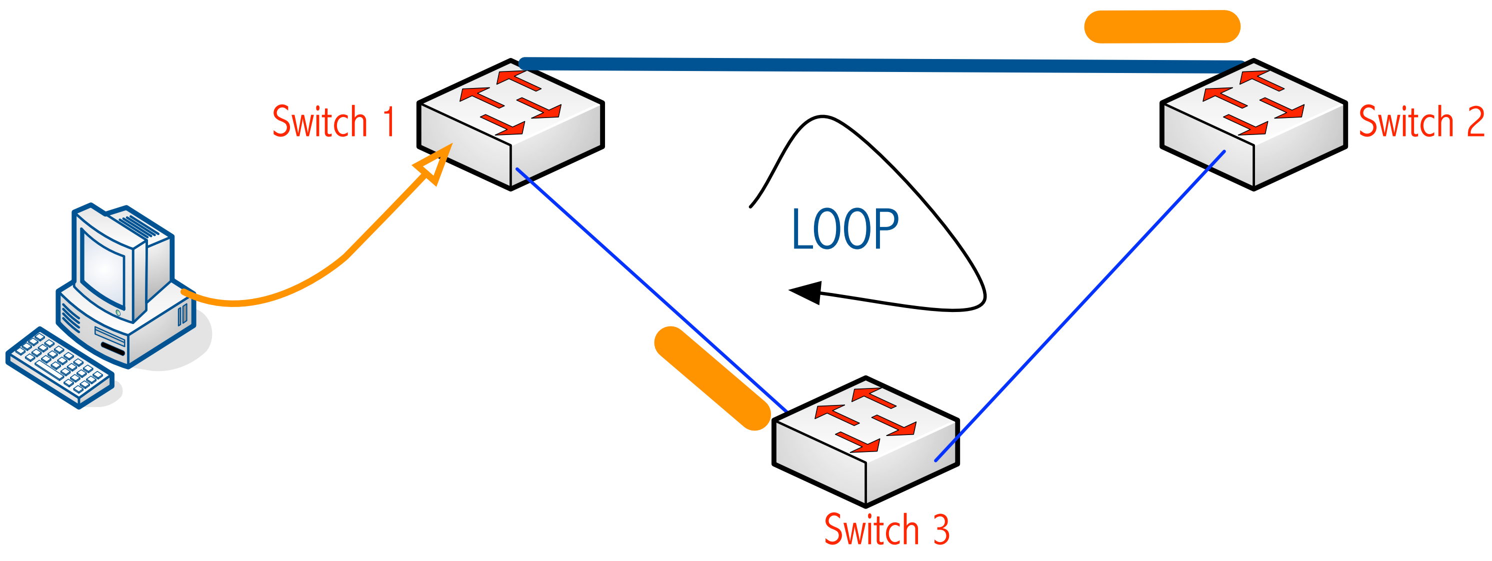
Switched LANs, broadcast and loops

- BCAST frame ingresses into Switch 2 and Switch 3



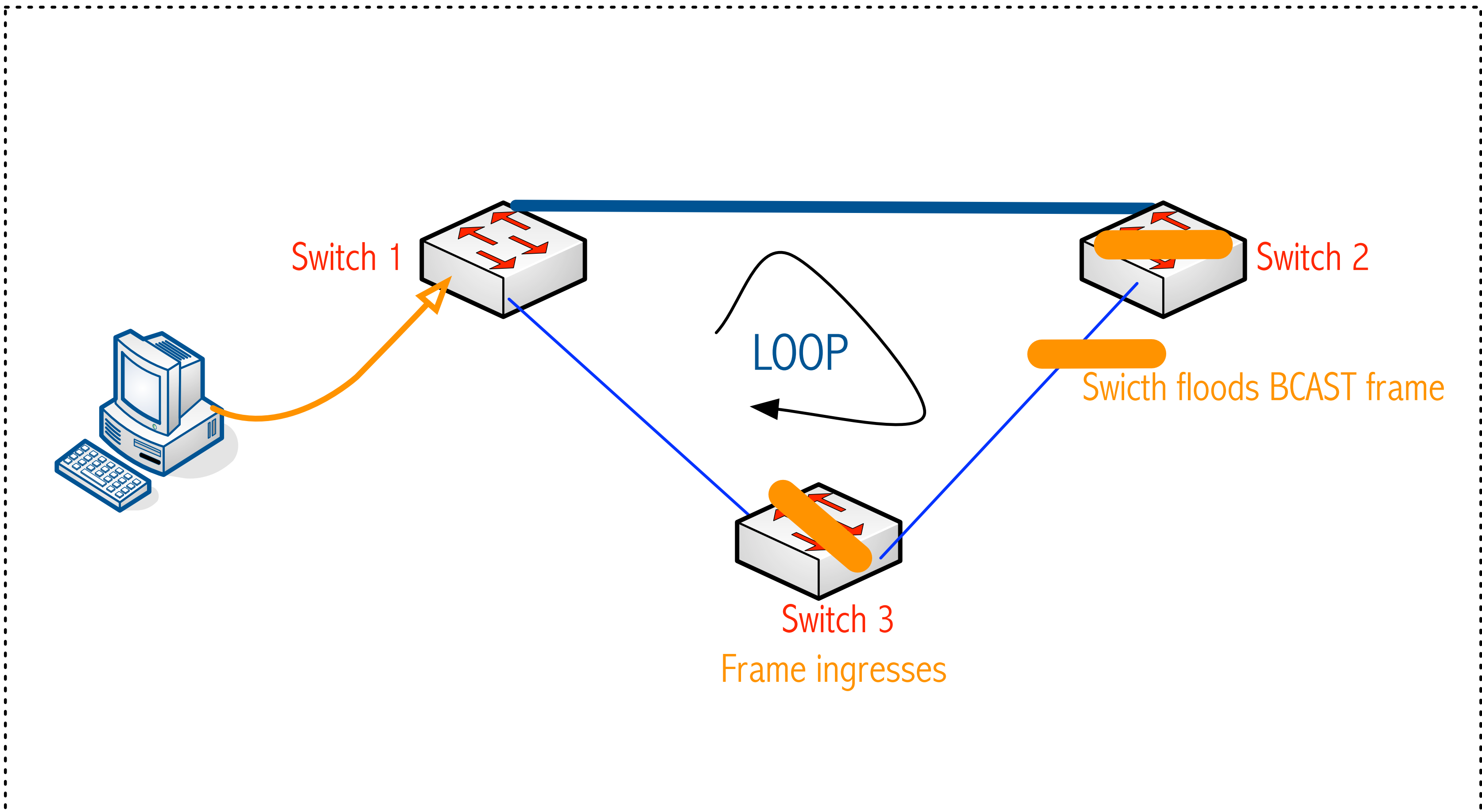
Switched LANs, broadcast and loops

- BCAST frame ingresses into Switch 2 and Switch 3



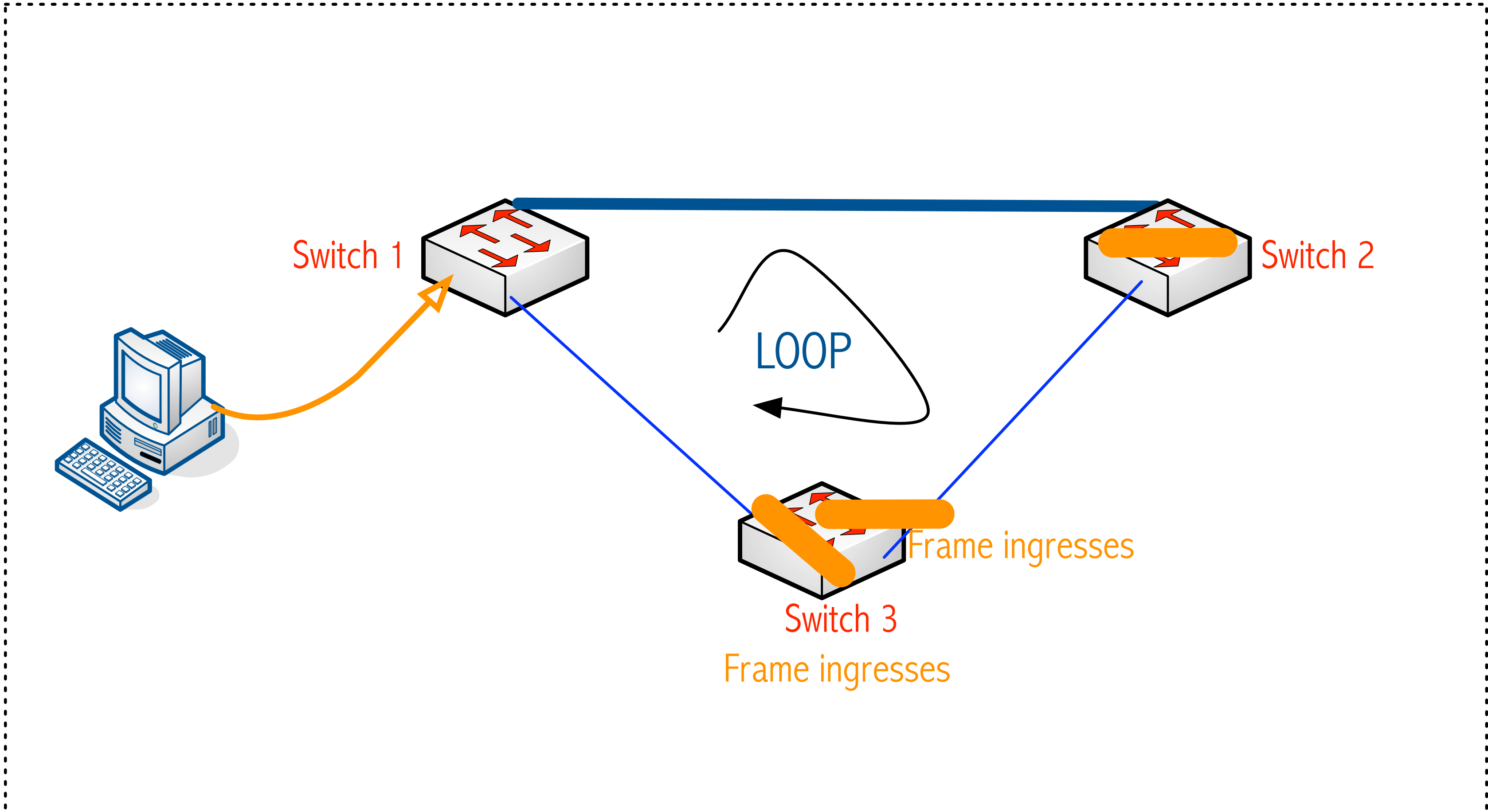
Switched LANs, broadcast and loops

- Switch 2 floods BCAST frame



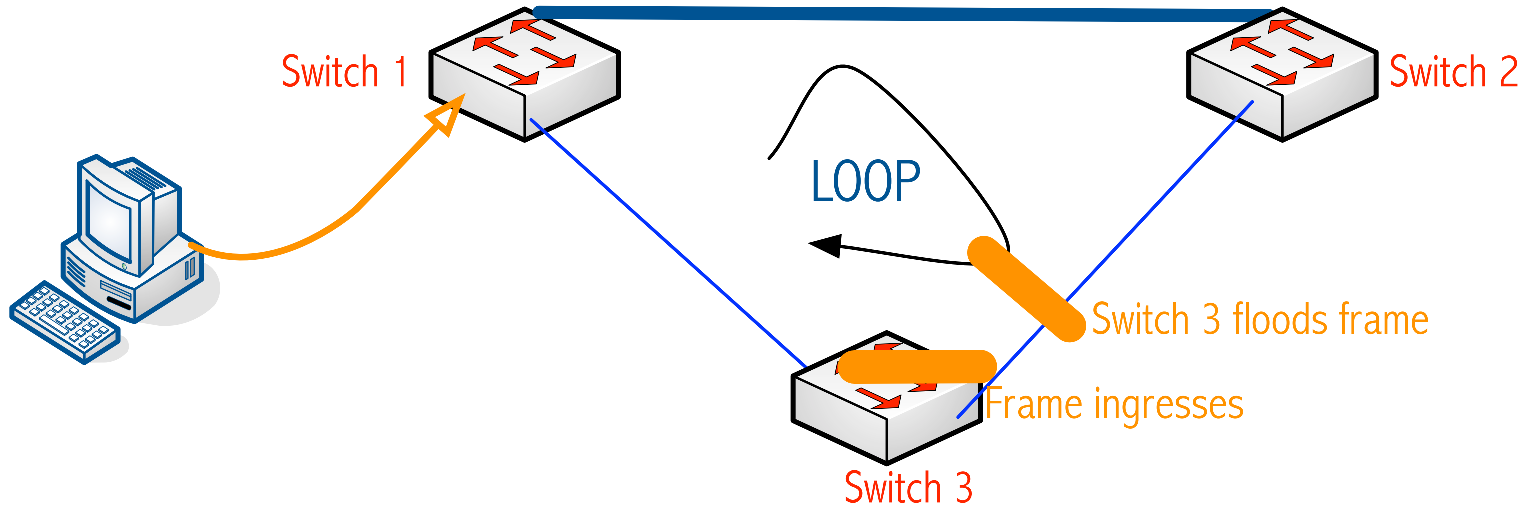
Switched LANs, broadcast and loops

- Switch 3 floods BCAST frame

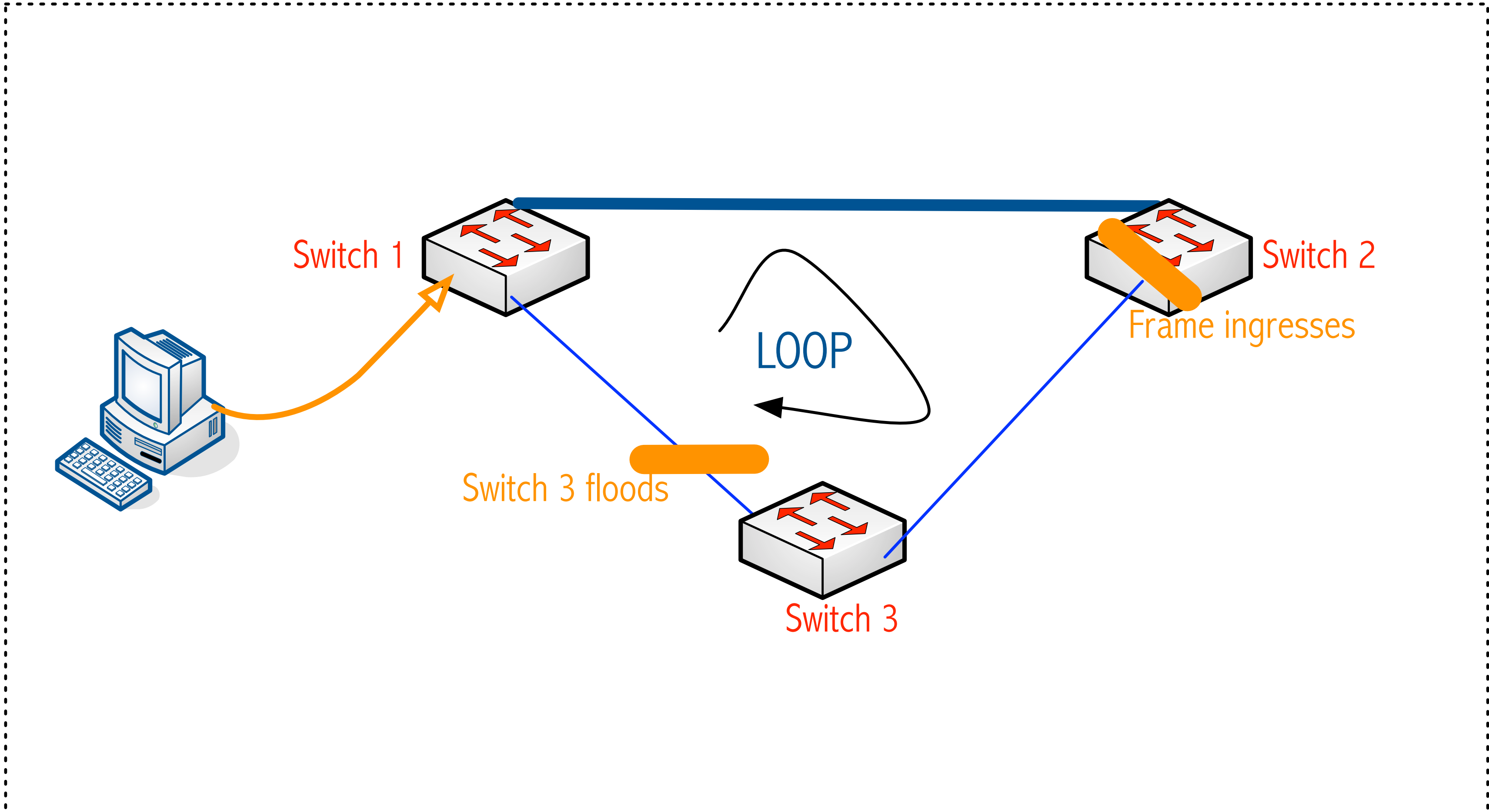


Switched LANs, broadcast and loops

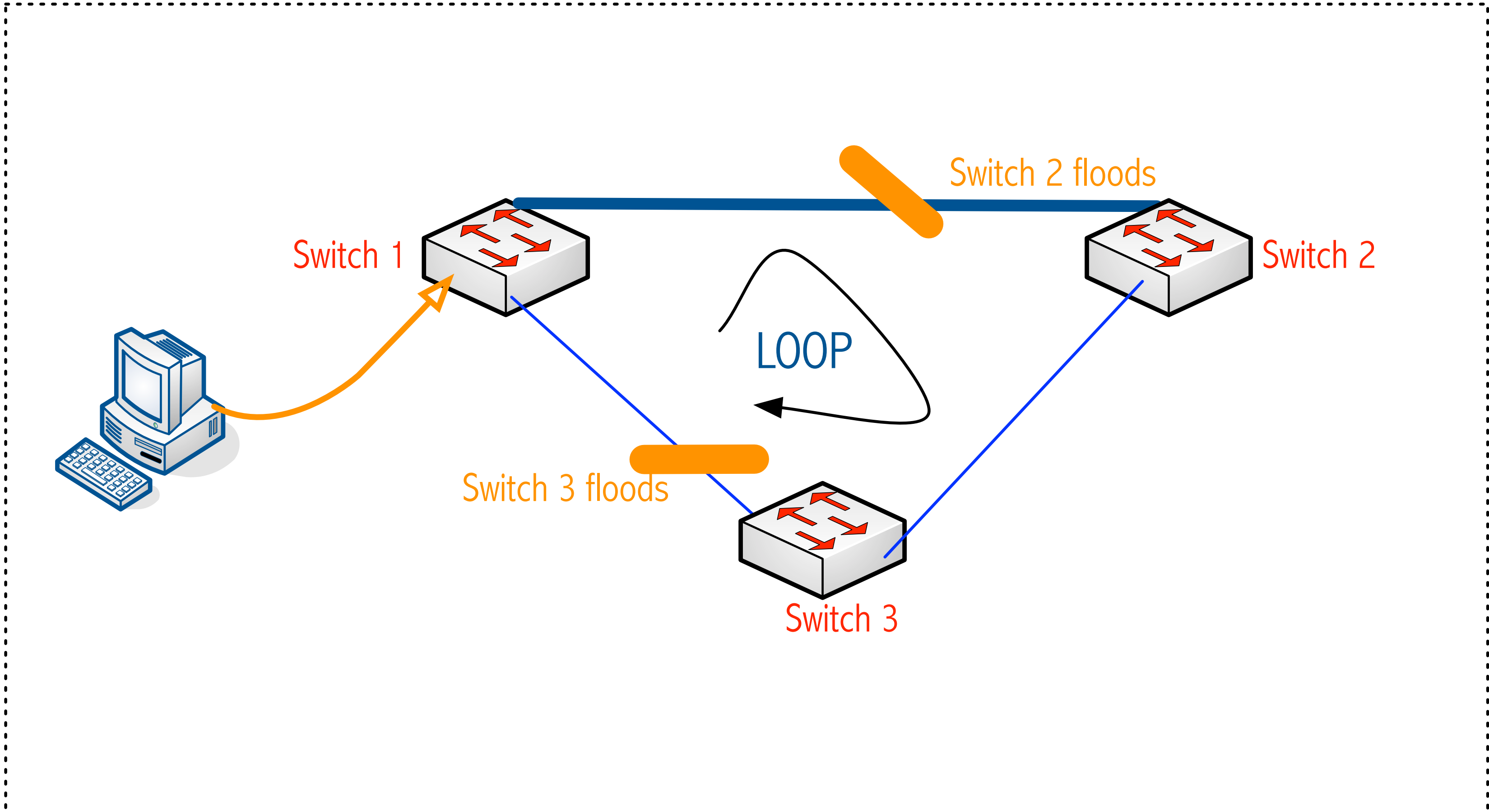
- New copy of frame goes towards Switch 2
- Switch 3 floods the new received copy of frame



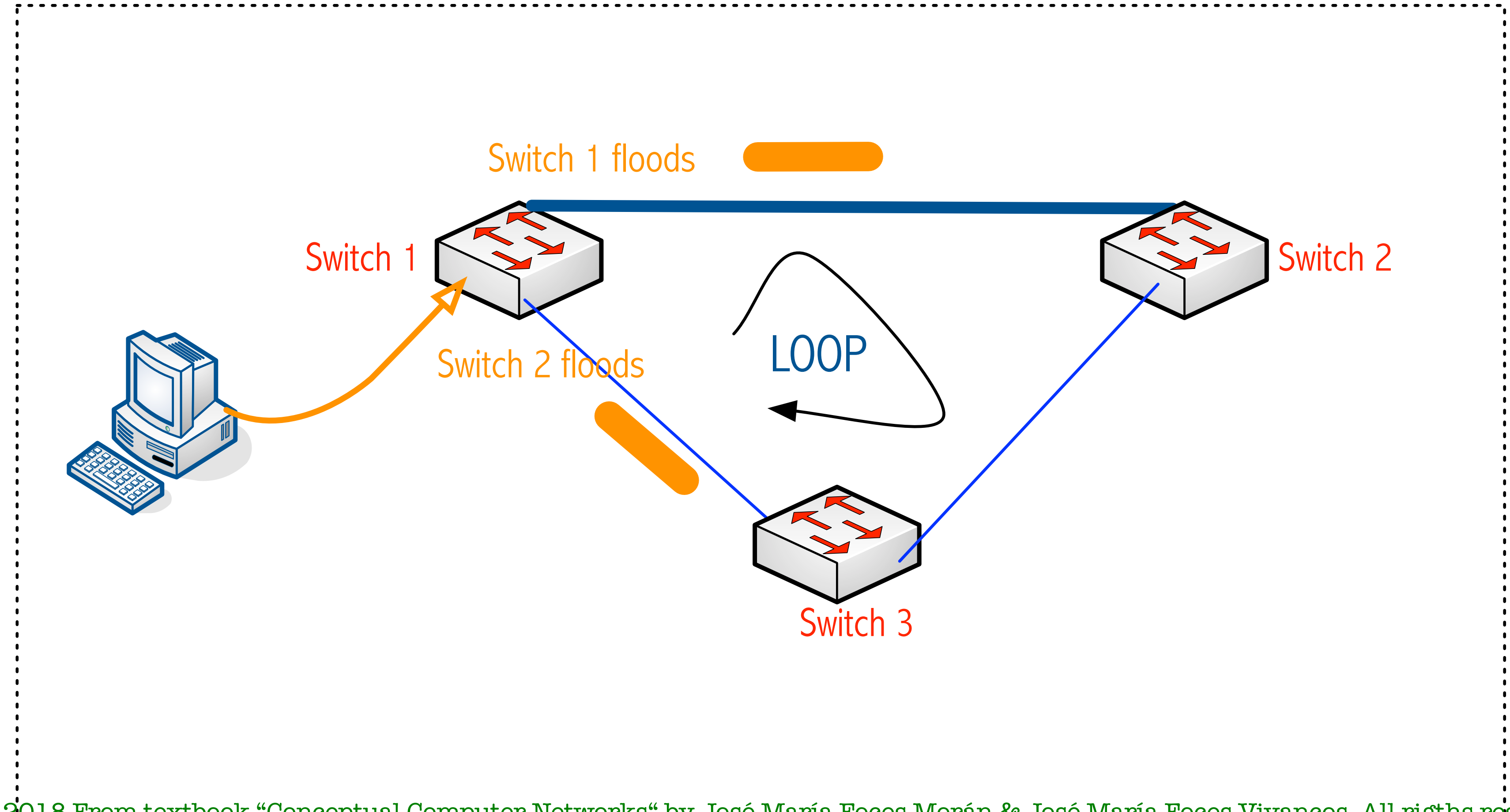
Switched LANs, broadcast and loops



Switched LANs, broadcast and loops

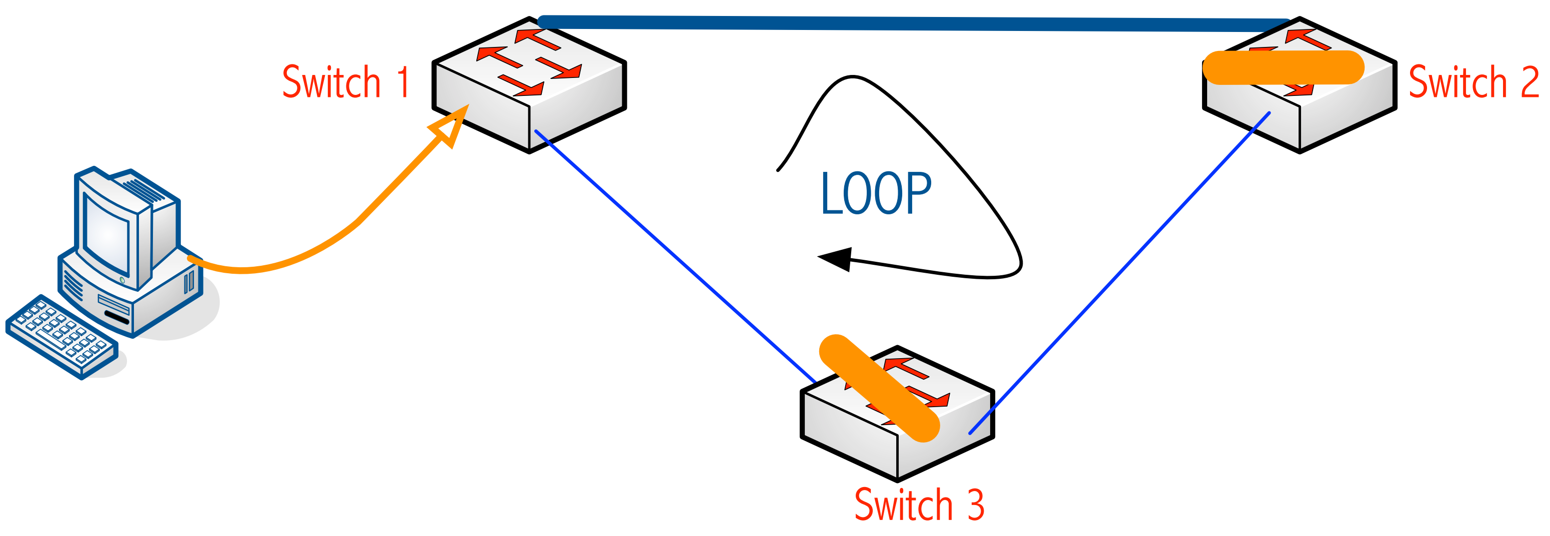


Switched LANs, broadcast and loops



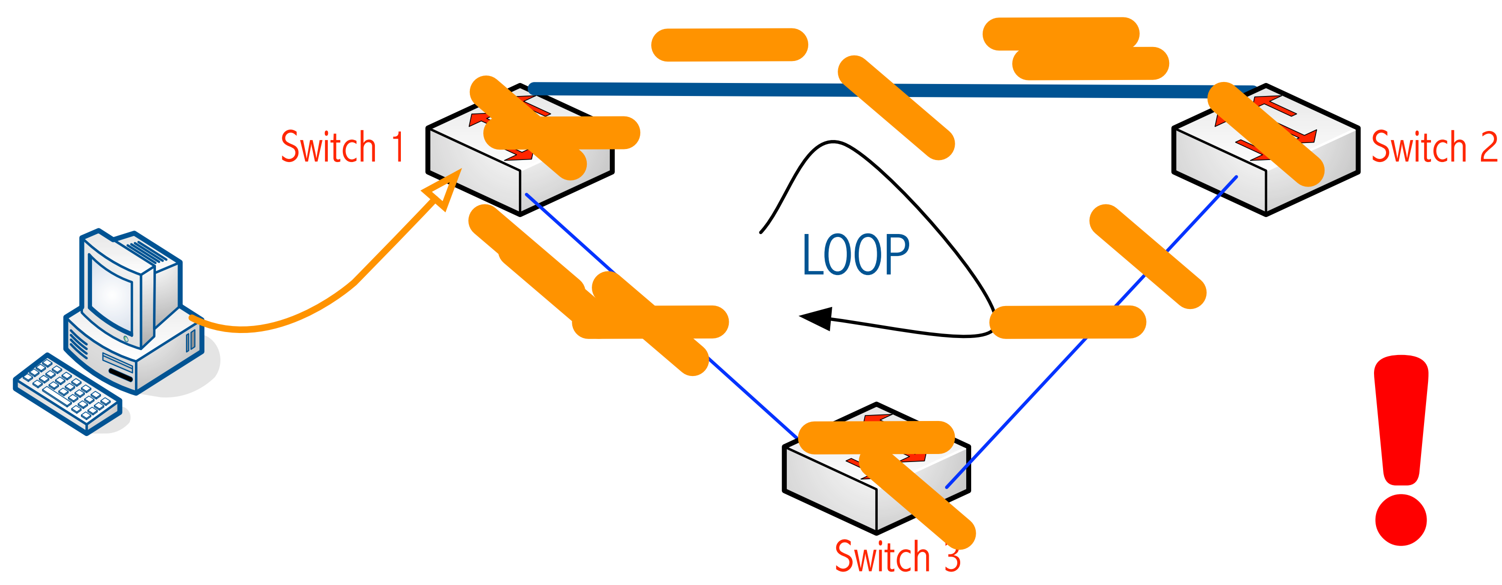
Switched LANs, broadcast and loops

- Frame keeps proliferating
- Endless cycle of receive-copy-send-recivey-copy-send
- Millions of copies of frame competing with just a few new, legitimate frames sent by stations



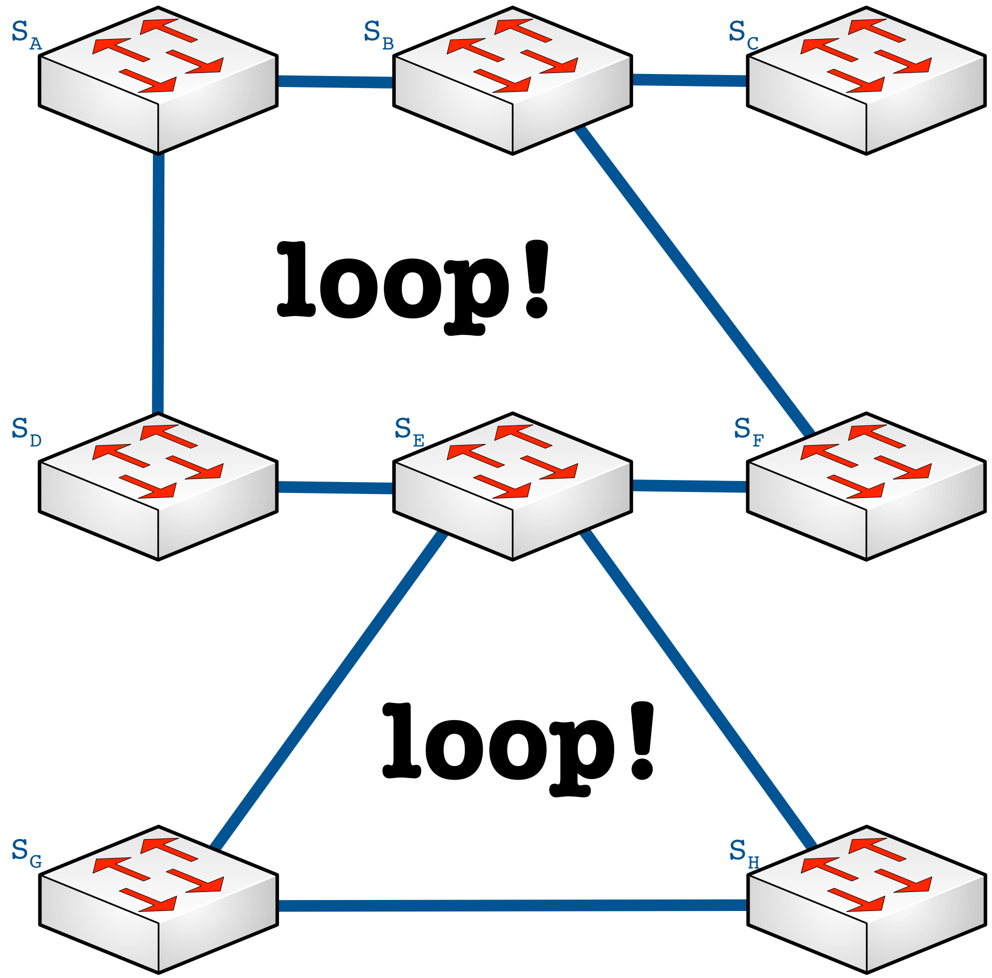
Switched LANs, broadcast and loops

- This process never ends and takes up all the network bandwidth!
- Loops provide redundant network paths in case of failures
- BUT, LOOPS CAUSE BROADCAST STORMS!!!



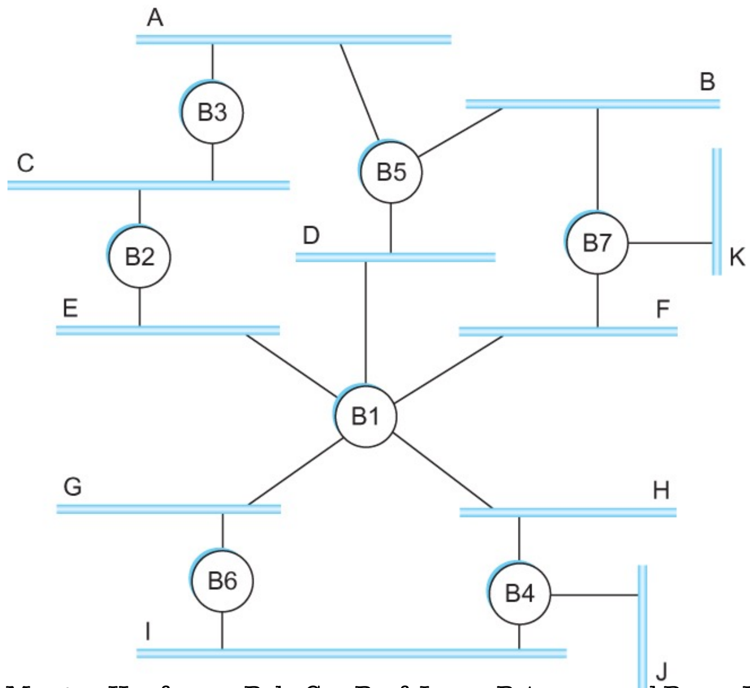
Can flooding live with loops?

- Great having loops and avoiding frame proliferation
- It's possible with:
  - Spanning Tree Protocol
  - STP
  - IEEE 802.3D



Switched Extended LANs

- Example:
  - What is this? A network: *ONE* network
  - B1, B2 … = Bridges
    - Remember: basically a switch
  - A, B, C: LAN segments (Collision domains)
    - Several collision domains
    - One broadcast domain: *The extended LAN*
  - Where are the end-nodes (hosts)?
    - Each end-node is connected to a segment



Spanning Tree Protocol (STP)

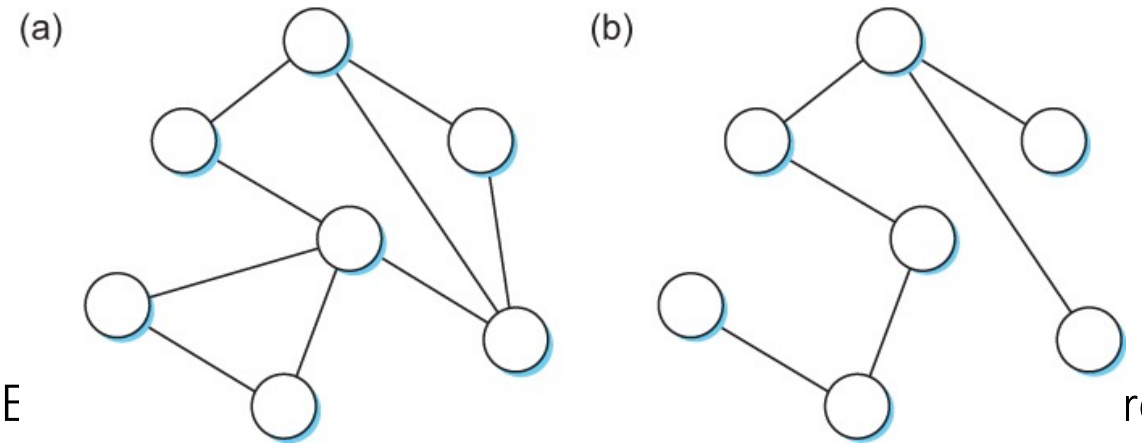
- Spanning Tree Algorithm is a distributed algorithm
- STP is based on it

Spanning Tree Algorithm

- The extended LAN may contain loops

- A **spanning tree** is a sub-graph of a graph that covers all its vertices but contains no cycles
  - It offers the same –*abstract*- connectivity but with no cycles

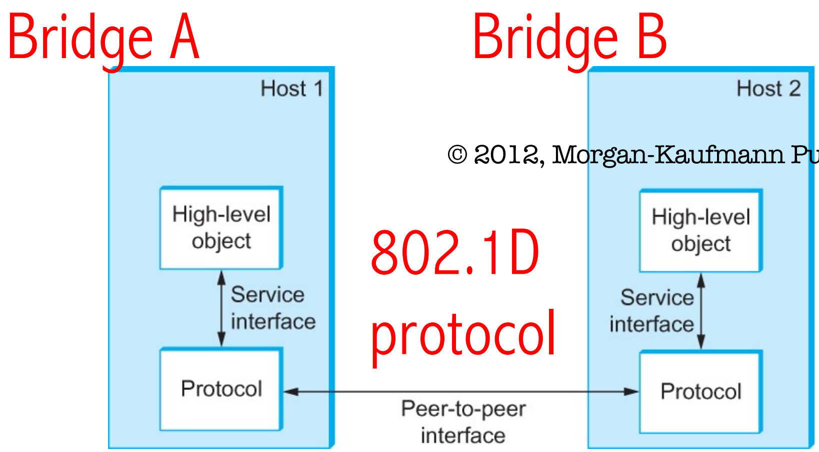
E



ree.

Spanning Tree Protocol (STP)

- A **protocol** used by a set of **bridges** to agree upon a spanning tree for a particular extended LAN
  - STP is based on the Spanning Tree Algorithm
- The IEEE 802.1D specification for LAN bridges is based on this algorithm
- Each bridge decides the ports over which it is and is not willing to forward frames
  - By removing ports from the topology the extended LAN is reduced to an acyclic tree
  - It is possible that an entire bridge will not participate in forwarding frames



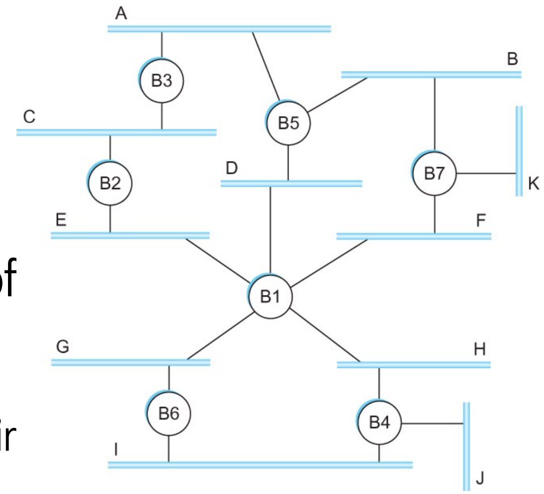
Spanning Tree Protocol (STP)

- Spanning Tree is executed in a distributed way (It's a distributed algorithm)
  - It is executed among a set of switches
  - The switches interchange STP messages (Look previous slide)

- The bridges are always ready to reconfigure themselves into a new spanning tree if some bridge or link fails

- Main idea
  - Each bridge selects the ports over which they will forward the frames

Spanning Tree *Algorithm*



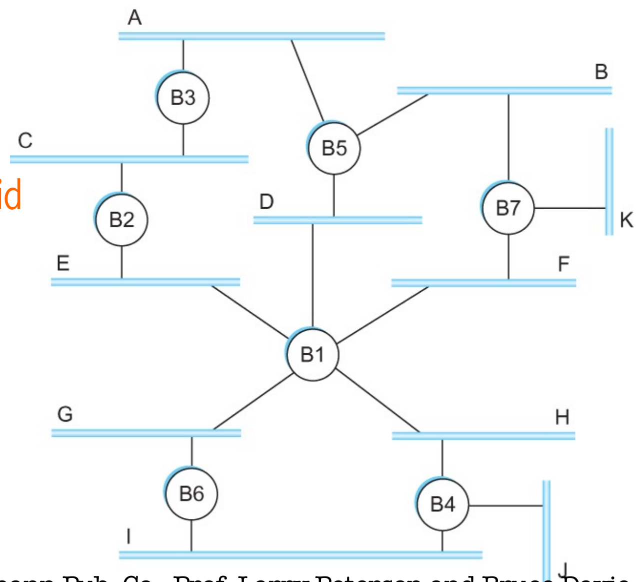
- The distributed algorithm selects ports as follows:

1. Each bridge has a unique identifier
   - B1, B2, B3… The Bridge with the smallest id becomes root of
   - The root bridge always forwards STP frames out over all of its ports
   - Actual switches use as ID the lowest MAC address allocated to their ports
2. RP = Root Port:
   - Each bridge computes the shortest path to the root and notes which of its ports is on this path
   - This port is selected as the bridge's preferred path to the root
3. DBP = Designated Bridge Port:
   - All bridges connected to a given LAN elect a single DBP
   - Responsible for forwarding frames toward the root bridge

Spanning Tree Algorithm

- Each bridge has a **root port (RP)**
  - The closest port to the root
  - Used for communication with the root

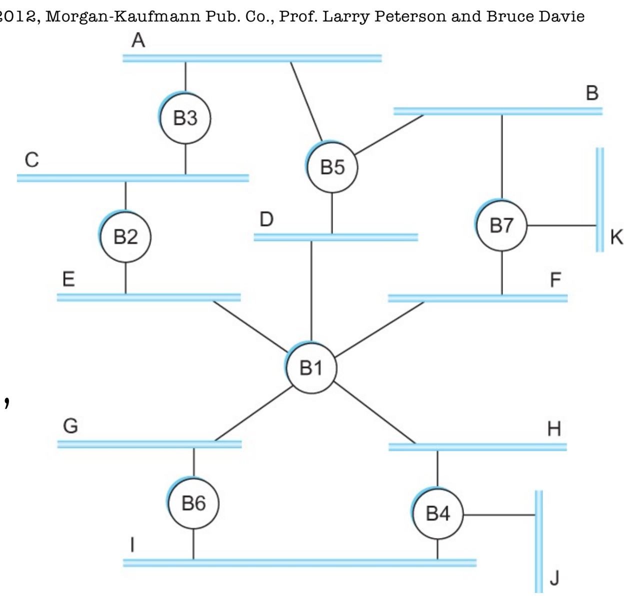
- If two or more ports are equally close to the root
  - *Break ties* by selecting the port with the smallest next-bridge id
  - If still equal cost, then *break ties* by choosing the port with lowest port id



- **Example: Which is B3's root port?**
  - B1 is root
  - Shortest distance from B3 to B1 (The root bridge)
    - Through A = 2
    - Through C = 2
    - Equal, then break ties:
      - A: Next bridge on least-cost path is B5
      - C: Next is B2 which has a lower ID than B5, <u>THEREFORE ROOT of B3 is its port C</u>

Spanning Tree Algorithm

- Each LAN has a **Designated Bridge Port (DBP)**
  - It's the one that is closest to the root

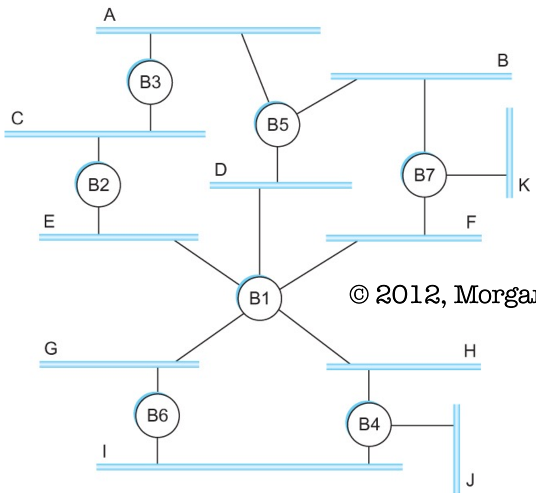


- If two or more bridges are equally close to the root,
  - *Break ties by* selecting the bridge with the smallest bridge id
  - If the bridge selected so far has two or more ports connected to a LAN, choose the port with lowest port id

- **Example: Which is the DBP of LAN B?**
  - Shortest distance from B → B1 (root) is 2 via B5 and via B7
  - Since B5 < B7, we select B5 as the *Designated Bridge on B,* specifically the port on the upper right of B5 is the Designated Bridge Port of LAN B

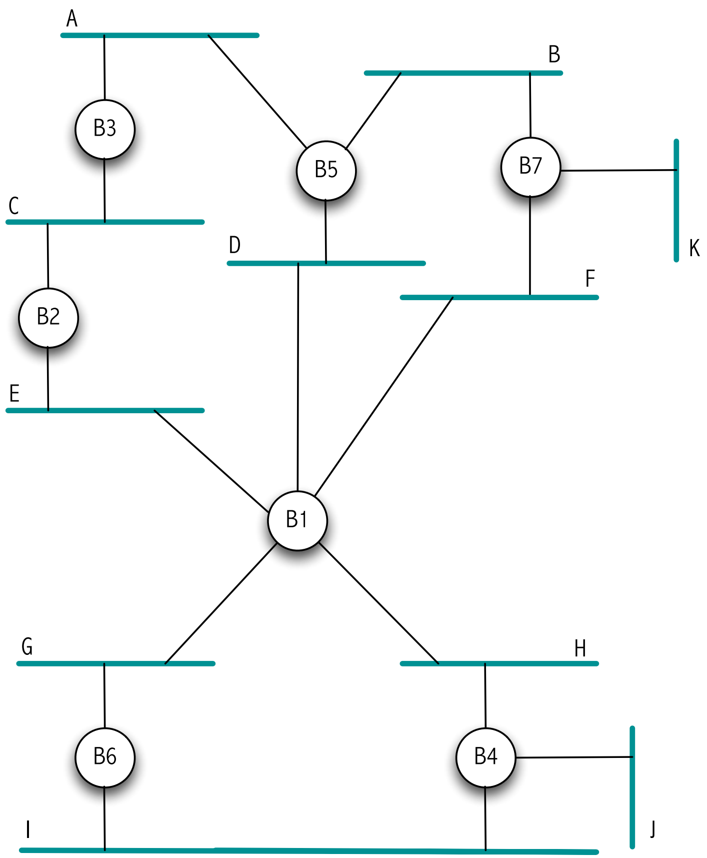
Spanning Tree Algorithm

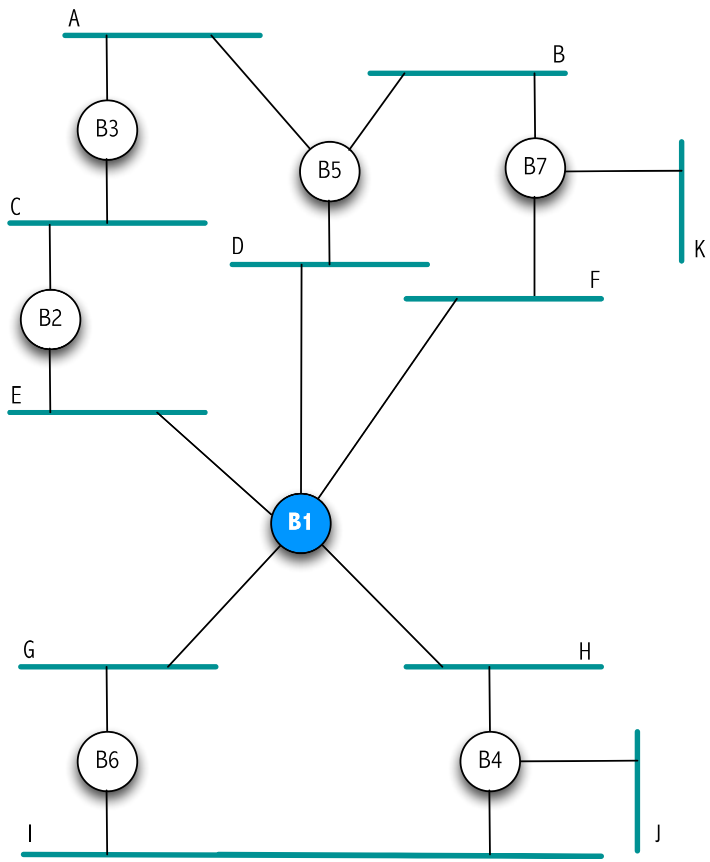
- Each LAN has a **Designated Bridge Port (DBP)**
  - It's the one that is closest to the root
  - *Each bridge is connected to more than one LAN*
  - *So it participates in the election of a designated bridge for each LAN it is connected to.*
  - *Each bridge decides if it is the designated bridge relative to each of its ports*
  - *The **bridge forwards frames** over those ports for which it is the designated bridge*



Spanning Tree Algorithm

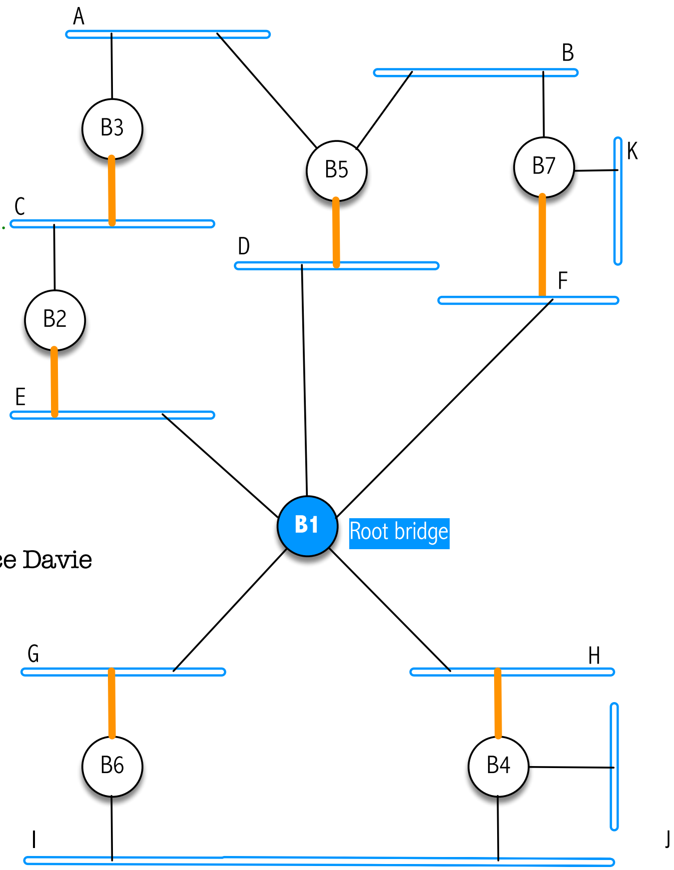
- Example from textbook pg. 194 (Fig. 3.10, P&D Ed. 5): Extended LAN with loops
- Step 1: Root bridge
  - B1 is the root bridge, the lowest numbered bridge





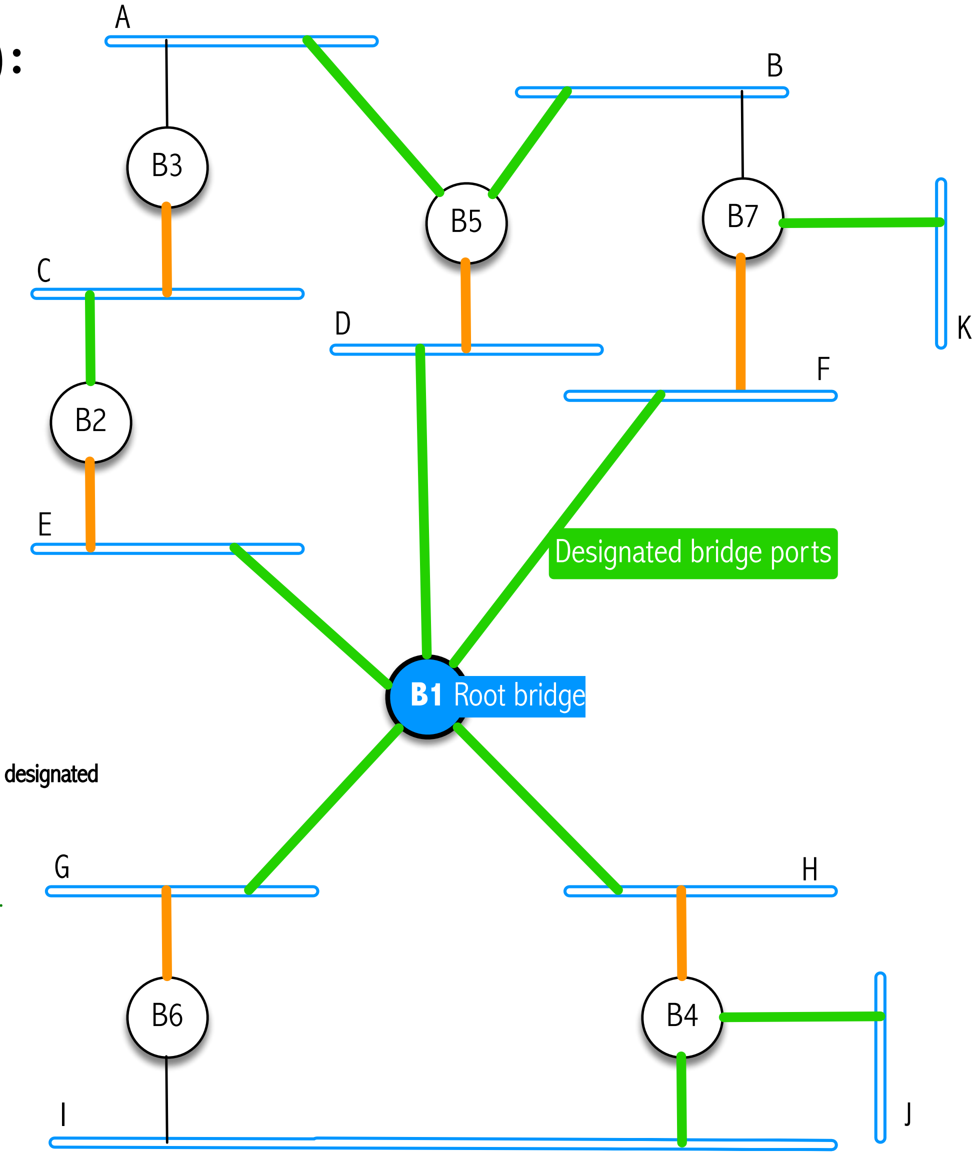
Spanning Tree Algorithm

- Textbook pg. 194 (Fig. 3.10; P&D Ed. 5): Extended LAN with loops
- Step 2: Root port (RP) of each bridge
  - B3 least cost to root is 2 (Via A and via C)
    - Break ties by lower label of next bridge: Choose B2 since label is lower numbered, B2 < B5
  - B4 least cost to root is 1 (Via H)
  - Calculate the root port of each bridge



Spanning Tree Algorithm

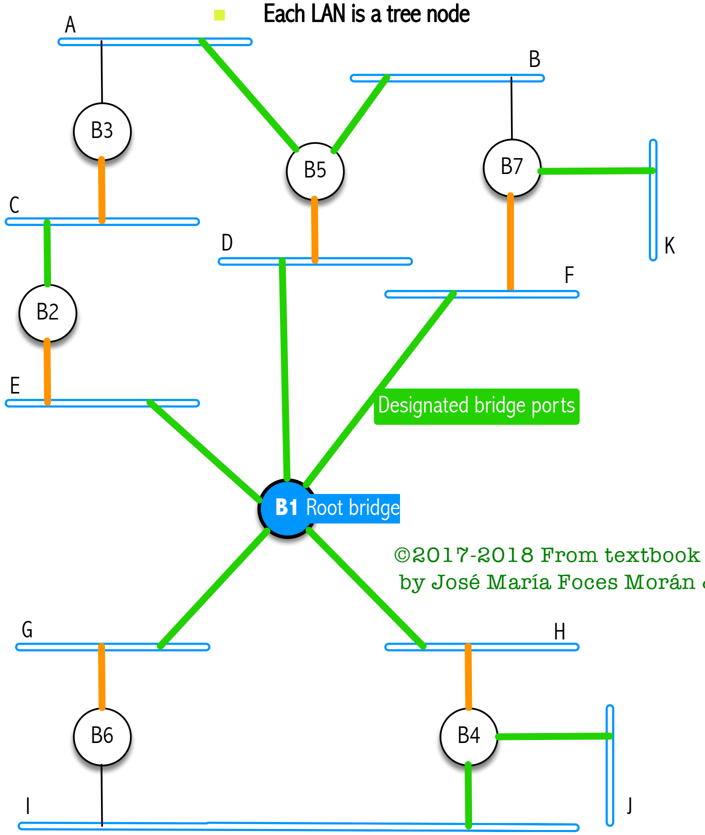
- **Textbook pg. 194 (Fig. 3.10; P&D Ed. 5): Extended LAN with loops**
- **Step 3: Designated Bridge at each LAN**
  - LAN A:
    - Cost to root via B3 = 3
    - Cost to root via B5 = 2
      - Choose bridge that is on the least cost path: B5
  - LAN J:
    - Connected to B4 only: Designated bridge is B4
  - LAN B:
    - Cost to root via B5 = 2
    - Cost to root via B7 = 2
      - Break ties by next bridge label, choose lower: B5 < B7, therefore designated bridge at LAN B is B5

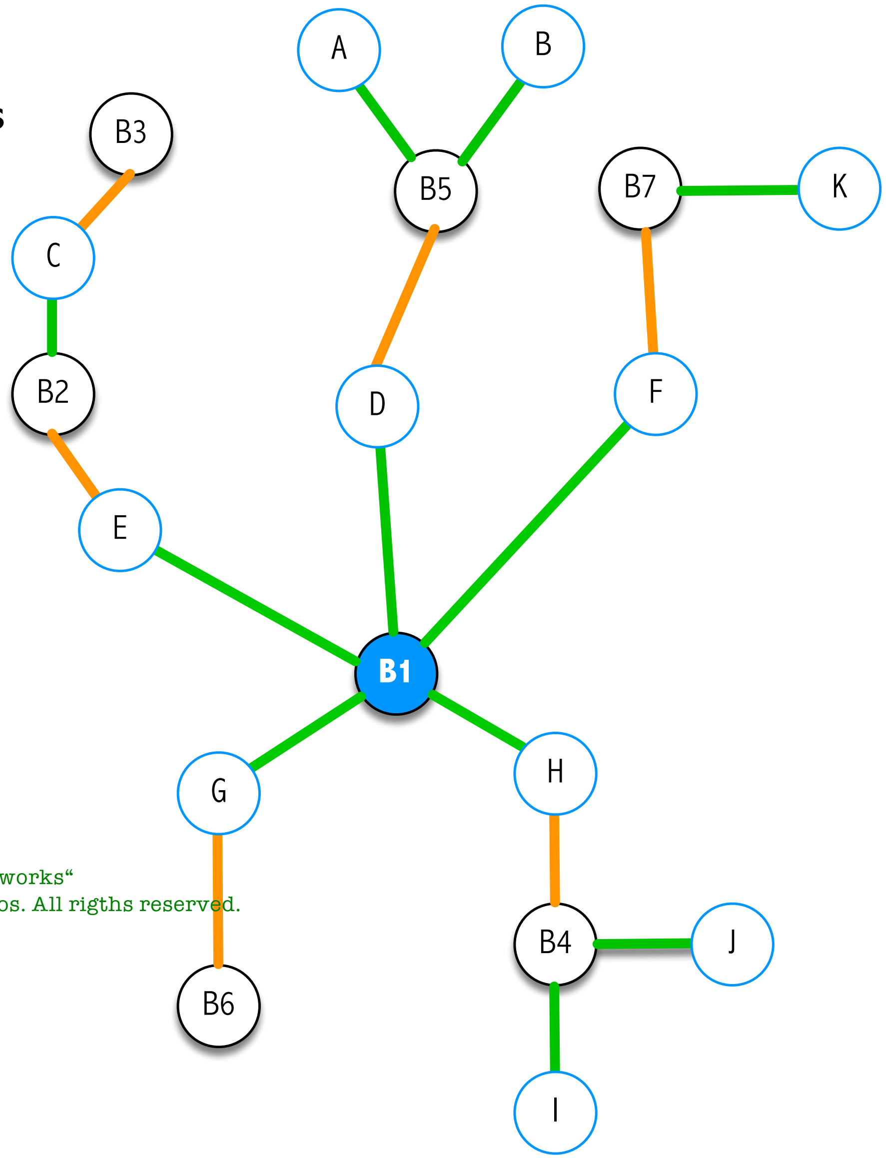


Spanning Tree Algorithm

Step 4: Obtain the spanning tree

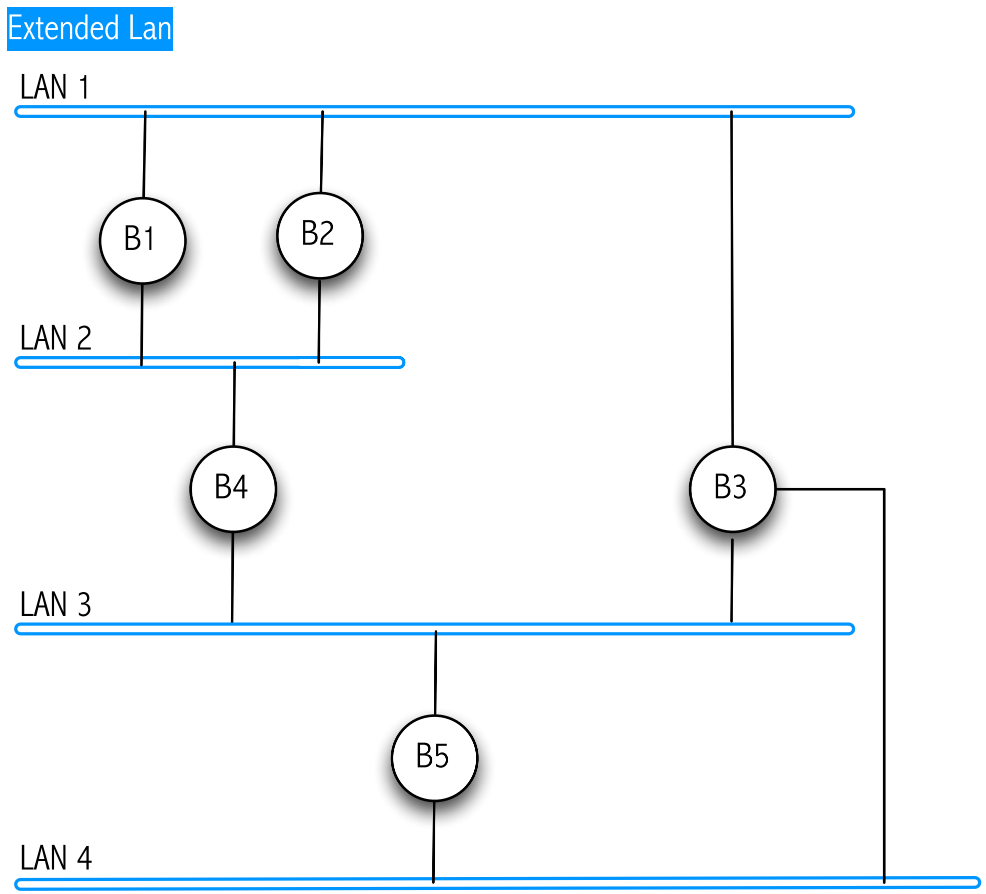
- All ports root or designated port result into active ports
  - Ports not RP nor DBP: disabled
- Spanning tree nodes:
  - Each bridge is a tree node
  - Each LAN is a tree node



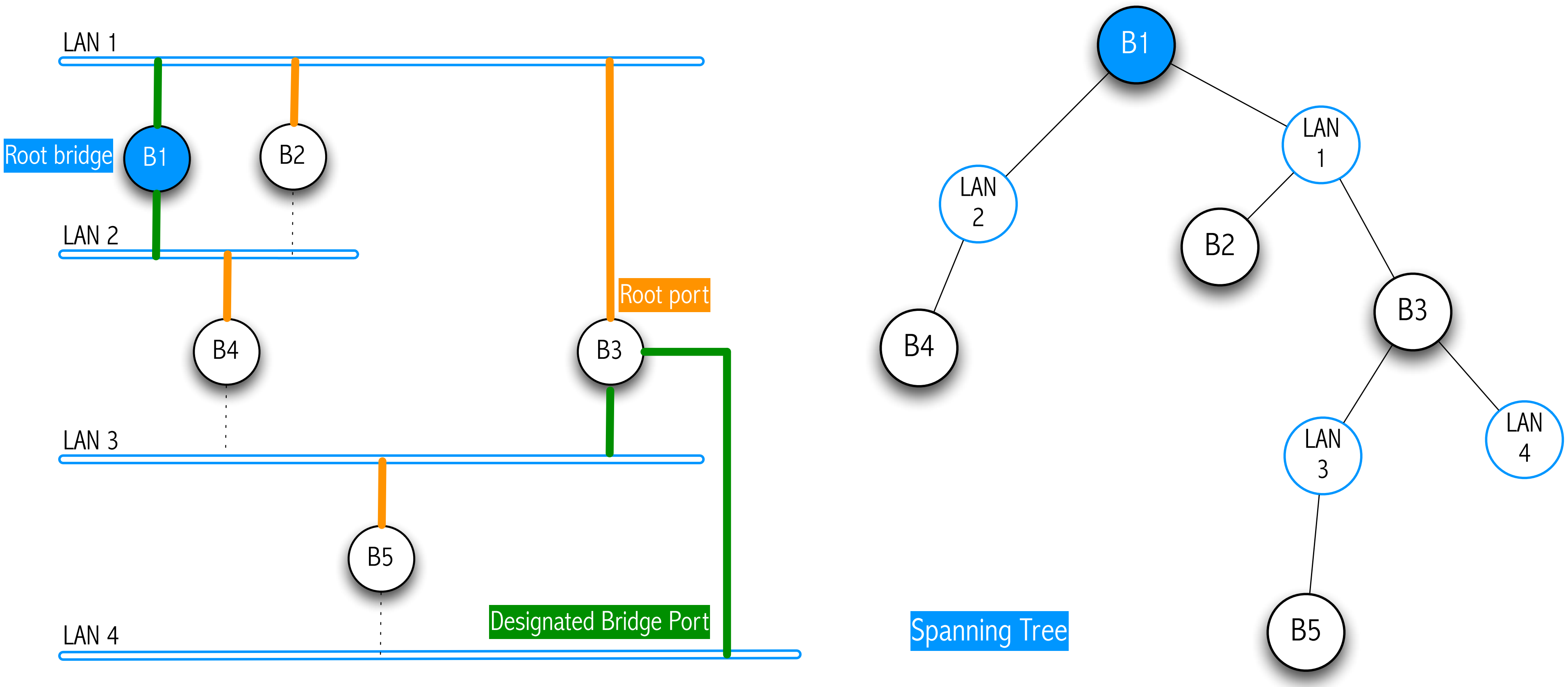


Spanning Tree Algorithm, example

- Obtain the Spanning Tree to the Extended Lan
  - 1. Root bridge
  - 2. RP
  - 3. DBP
  - ST



Spanning Tree Algorithm, example



STP messages

- The Spanning Tree Algorithm in STP is a distributed algorithm
  - It is executed among the switches of an Extended Lan by exchanging STP messages
- Initially each bridge thinks it is the root
  - It sends a configuration message on each of its ports identifying itself as the root and giving a distance to the root of 0
- Upon receiving a configuration message over a particular port a bridge checks to see if the new message is better than the current best configuration message recorded for that port
- The new configuration is better than the currently recorded information if
  - It identifies a root with a smaller id or
  - It identifies a root with an equal id but with a shorter distance or
  - The root id and distance are equal, but the sending bridge has a smaller id

STP messages

- If the new message is better than the currently recorded one,
  - The bridge discards the old information and saves the new information
  - It first adds 1 to the distance-to-root field

- When a bridge receives a configuration message indicating that it is not the root bridge (that is, a message from a bridge with smaller id)
  - The bridge stops generating configuration messages on its own
  - Only forwards configuration messages from other bridges after 1 adding to the distance field

STP messages

- When a bridge receives a configuration message that indicates it is not the designated bridge for that port

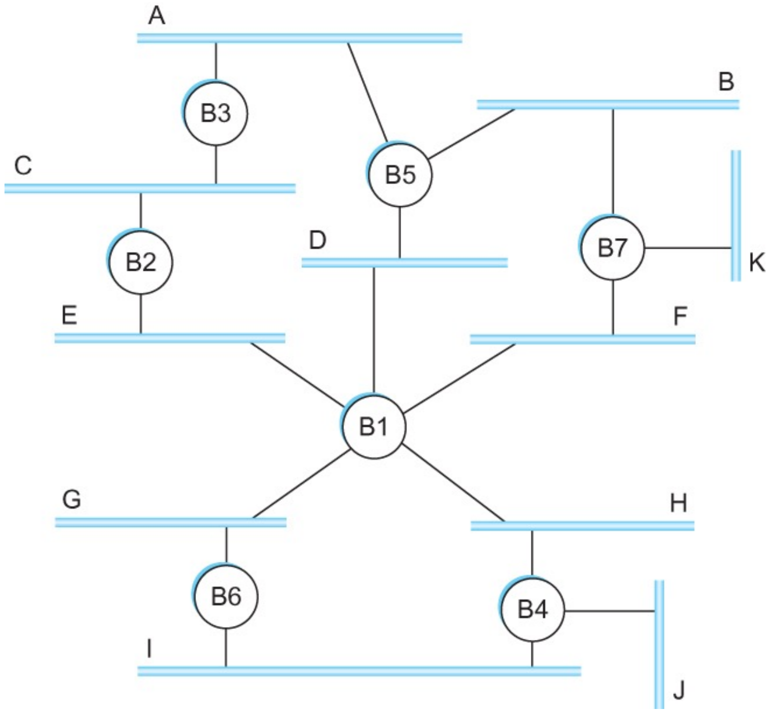
  => a message from a bridge that is closer to the root or equally far from the root but with a smaller id

  - The bridge stops sending configuration messages over that port

- When the system stabilizes,
  - Only the root bridge is still generating configuration messages.
  - Other bridges are forwarding these messages only over ports for which they are the designated bridge

STP messages

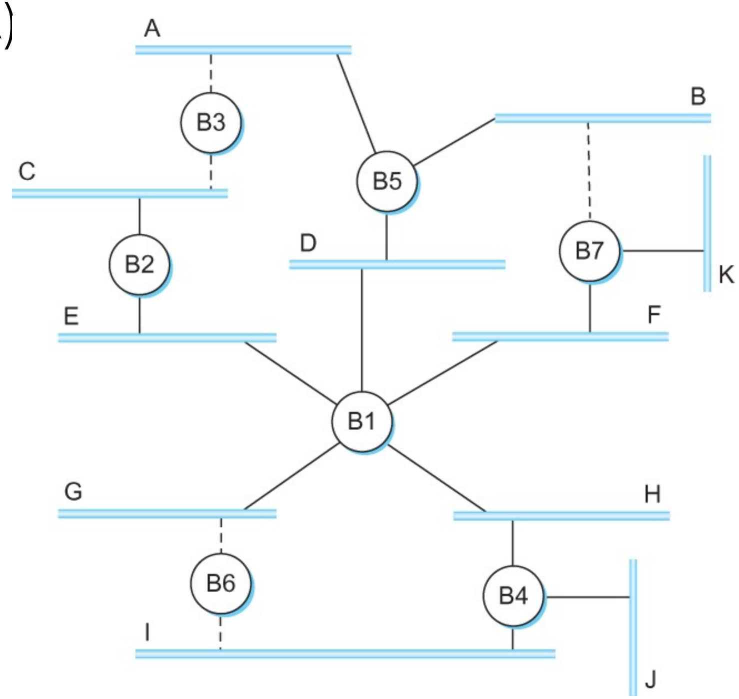
- Consider the situation when the power had just been restored to the building housing the following network



- All bridges would start off by claiming to be the root

STP messages

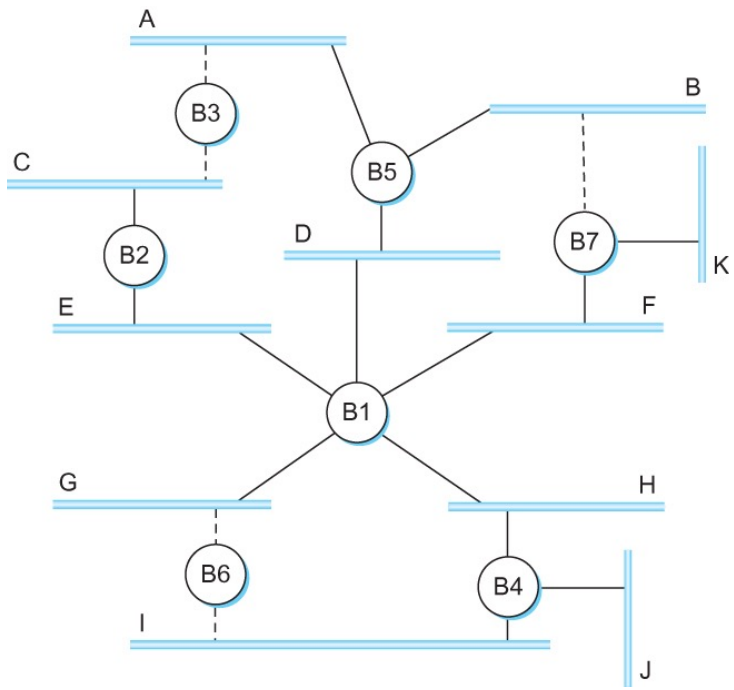
- Denote a configuration message from node X in which it claims to be distance d from the root node Y as (Y, d, X)



- Consider the activity at node B3

STP messages

- B3 receives (B2, 0, B2)
- Since 2 < 3, B3 accepts B2 as root
- B3 adds 1 to the distance advertised by B2 and sends (B2, 1, B3) to B5
- Meanwhile B2 accepts B1 as root because it has the lower id and it sends (B1, 1, B2) toward B3
- B5 accepts B1 as root and sends (B1, 1, B5) to B3
- B3 accepts B1 as root and it notes that both B2 and B5 are closer to the root than it is.
  - Thus B3 stops forwarding messages on both its interfaces
  - This leaves B3 with both ports not selected



Spanning Tree Algorithm

- Even after the system has stabilized, the root bridge continues to send configuration messages periodically
  - Other bridges continue to forward these messages
- When a bridge fails, the downstream bridges will not receive the configuration messages
- After waiting a specified period of time, they will once again claim to be the root and the algorithm starts again
- Note
  - Although the algorithm is able to reconfigure the spanning tree whenever a bridge fails, it is not able to forward frames over alternative paths for the sake of routing around a congested bridge

Spanning Tree Algorithm

- Broadcast and Multicast
  - Forward all broadcast/multicast frames
    - Current practice
  - Learn when no group members downstream
  - Accomplished by having each member of group G send a frame to bridge multicast address with G in source field

Spanning Tree Algorithm

- Limitation of Bridges
  - Do not scale
    - Spanning tree algorithm does not scale
    - Broadcast does not scale
  - Do not accommodate heterogeneity

The end