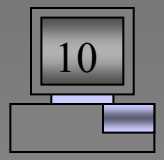
Transparent Bridging

Yudhie Kurnia M.

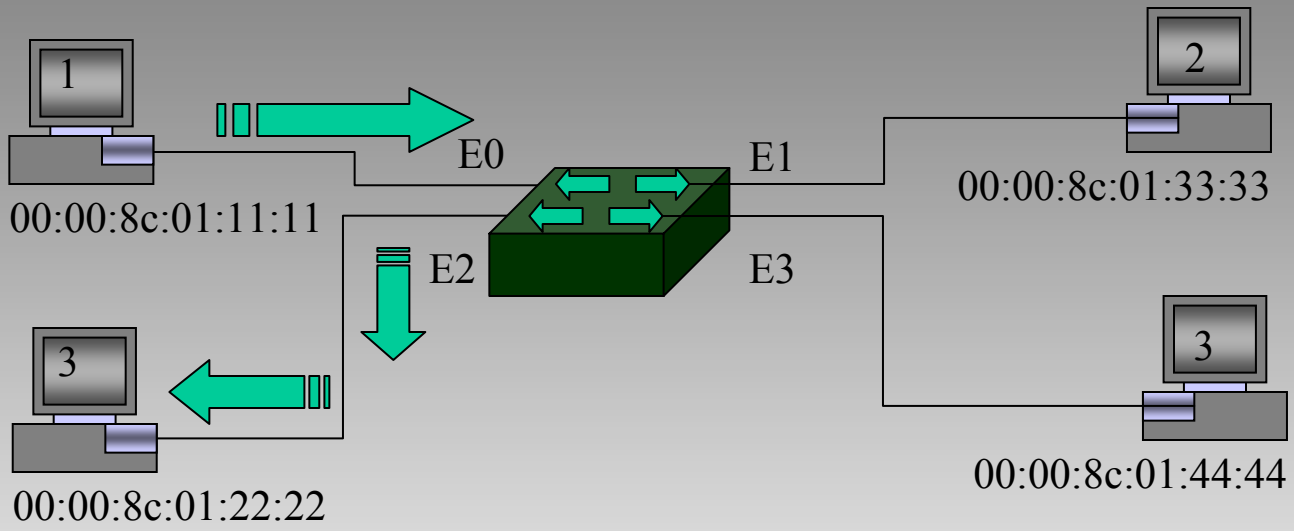
Transparent Bridging

- Transparent bridges were first developed at Digital Equipment Corporation (Digital) in the early 1980s.
- Work into the IEEE 802.1 standard. Transparent bridges are very popular in Ethernet/IEEE 802.3 networks.
- When transparent bridges are powered on, they learn the workstation locations by analyzing the source address of incoming frames from all attached networks.
- If a bridge sees a frame arrive on port 1 from Host A, the bridge concludes that Host A can be reached through the segment connected to port 1. Through this process, transparent bridges build a table



13:20:48:53:50:00

Address Learning



- Host 1 (MAC: 00:00:8c:01:11:11) mengirim frame ke Host 3 (MAC: 00:00:8c:01:22:22)
- Switch menerima frame tersebut melalui port E0. Switch kemudian tahu bahwa Host 1 dapat dicapai lewat E0. Hal ini dicatat ke dalam tabel.

Address Learning

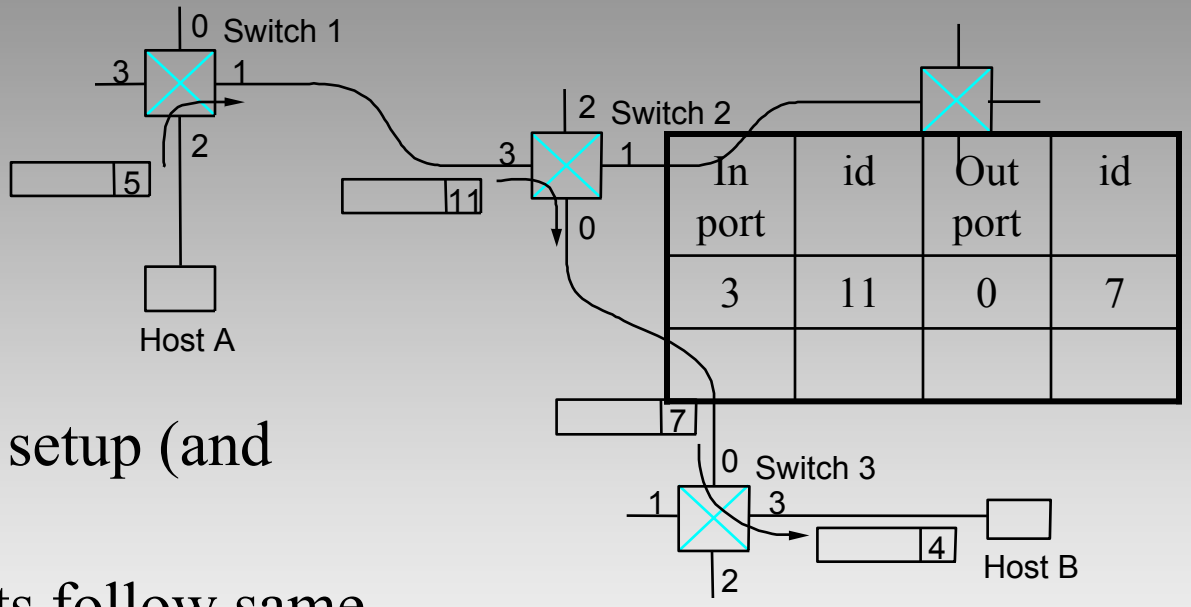
- Switch tidak tahu di mana letak Host 3. Switch mem-broadcast ke semua port-nya
- Host 3 menerima frame tersebut, dan memberi jawaban (response) ke Host 1
- Switch menerima jawaban dari Host 3 yang ditujukan ke Host 1 via port E2. Switch kemudian mencatat bahwa Host 3 terletak di port E2 ke dalam tabelnya.
- Host 1 dan Host 3 kemudian dapat kirim-kiriman frame (point-to-point conection) tanpa didengar oleh Host 2 dan Host 4.

Operation

- The bridge uses its table as the basis for traffic forwarding.
- When a frame is received on one of the bridge's interfaces, the bridge looks up the frame's destination address in its internal table.
- If the table contains an association between the destination address and any of the bridge's ports aside from the one on which the frame was received, the frame is forwarded out the indicated port.
- If no association is found, the frame is flooded to all ports except the inbound port. Broadcasts and multicasts also are flooded in this way.

Virtual Circuit Switching

In port	id	Out port	id
2	5	1	11



In port	id	Out port	id
3	11	0	7

- Explicit connection setup (and tear-down) phase
- Subsequence packets follow same circuit
- Sometimes called *connection-oriented* model

In port	id	Out port	id
0	7	3	4

Virtual Circuit Switching

- Connection setup

  Untuk menyambungkan antara A dan B, NA menset suatu nilai VCI yang belum dipakai. Sebagai contoh,

  VCI=5 diberikan untuk link dari A ke switch 1.

  VCI=11 untuk link dari switch 1 ke switch 2

  VCI=7 untuk link dari switch 2 ke switch 3

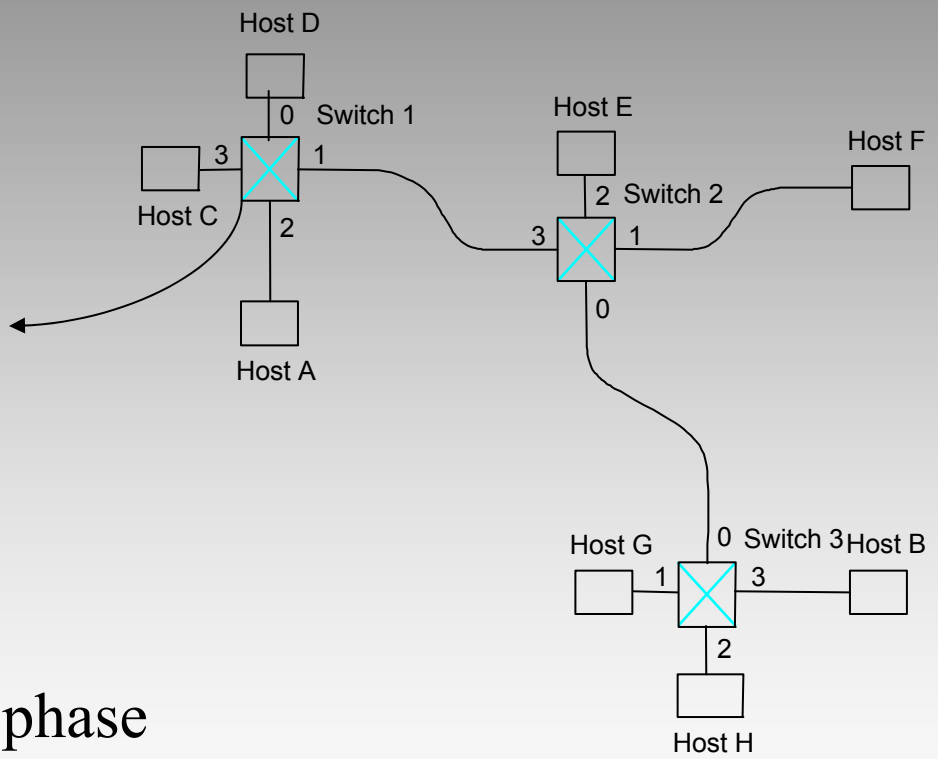
  VCI=4 untuk link dari switch 3 ke B

- Data Transfer

  Untuk setiap data yang akan dikirim dari A ke B, A harus mengisi nilai VCI=5 dan mengirim paket tersebut ke switch 1. Switch 1 menerima paket tersebut. Switch 1 menggunakan kombinasi inport dan VCI untuk menentukan entry tabel-nya.

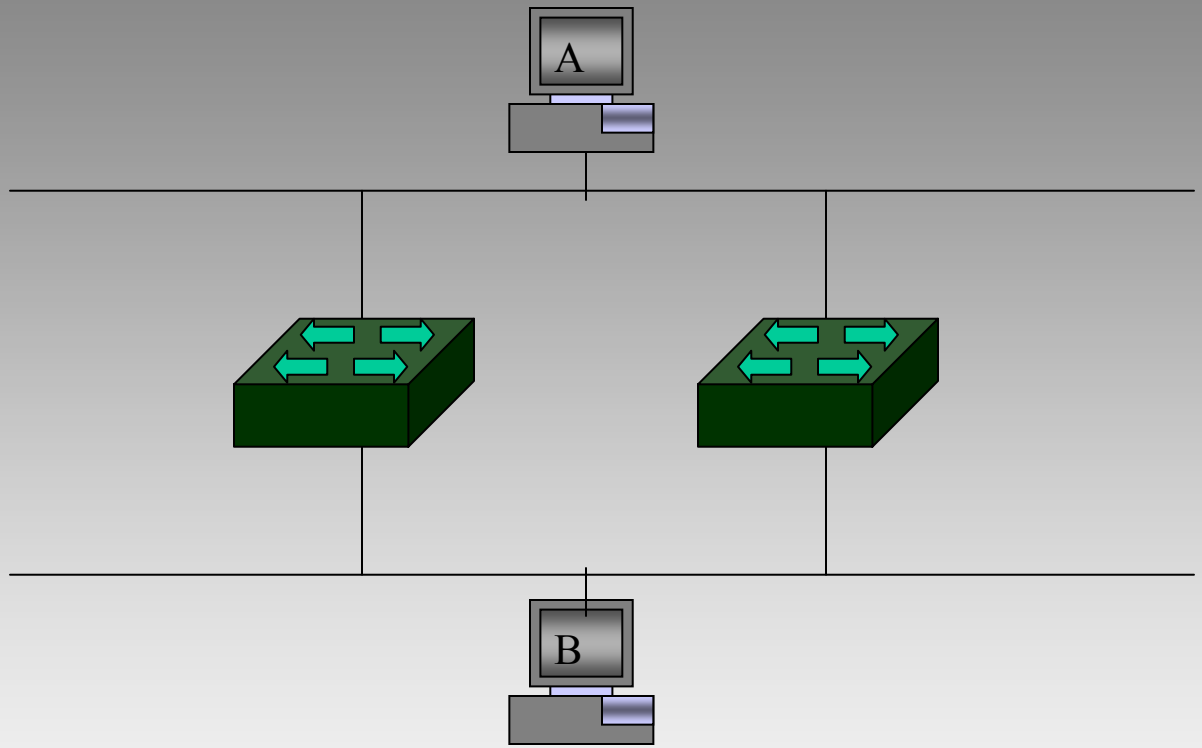
Datagram Switching

Dest	Port
A	2
B	1
C	3
D	0
E	1
F	1
G	1
H	1



- No connection setup phase
- Each packet forwarded independently
- Sometimes called *connectionless* model

Bridging Loops



- Without a bridge-to-bridge protocol, the transparent-bridge algorithm fails when multiple paths of bridges and local-area networks (LANs) exist between any two LANs in the internetwork

Bridging Loops

- Suppose that Host A sends a frame to Host B. Both bridges receive the frame and correctly learn that Host B is on segment 2. Each bridge then forwards the frame onto segment 2.
- Host B will receive **two copies** of the frame (once from bridge 1 and once from bridge 2) ☹
- Each bridge now believes that Host A resides on the same segment as Host B.
- When Host B replies to Host A's frame, both bridges will receive and subsequently **filter** (stop) the replies because the bridge table will indicate that the destination (Host A) is on the same network segment as the frame's source. ☹

Bridging Loops

- Host A's initial frame is a **broadcast**. Both bridges forward the frames endlessly, using all available network bandwidth and blocking the transmission of other packets on both segments. ☹☹
- A loop implies the existence of multiple paths through the internetwork, and a network with multiple paths from source to destination can increase overall network fault tolerance through improved topological flexibility. ☺

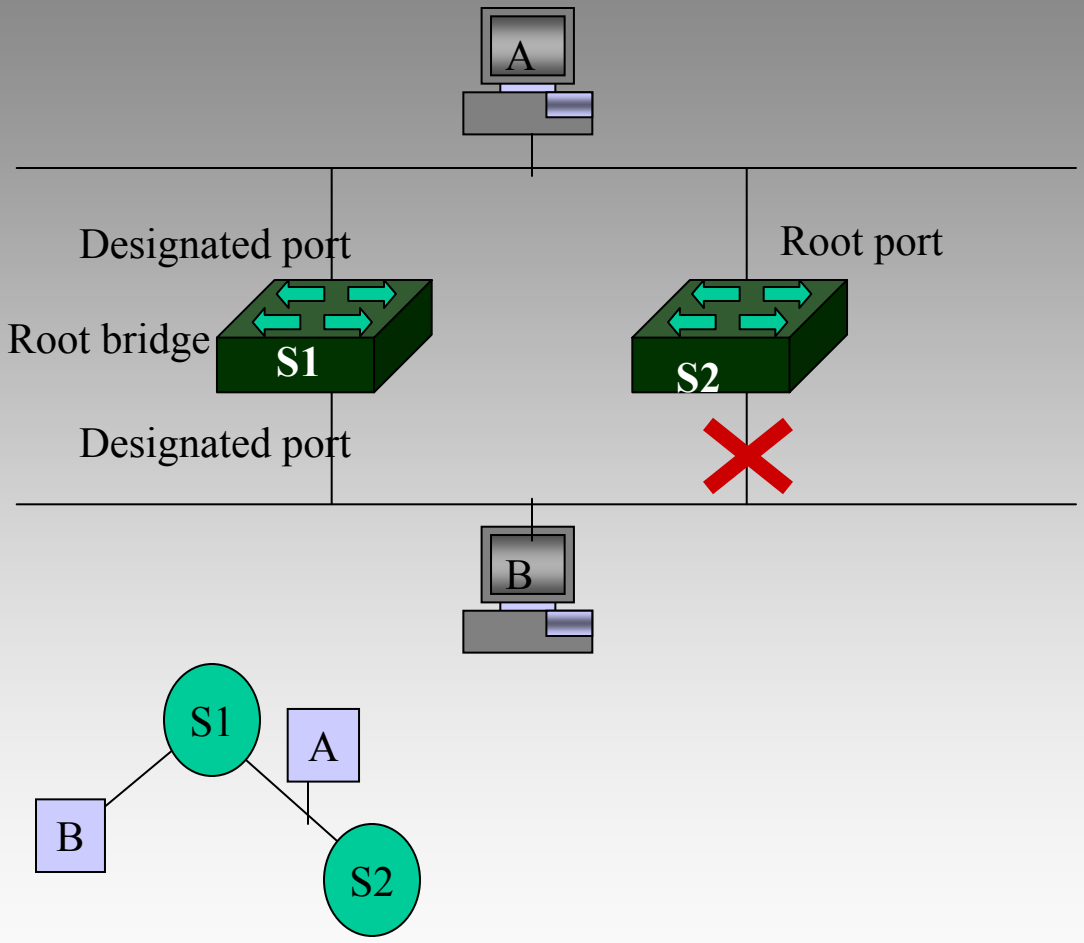
Spanning-Tree

- The *spanning-tree algorithm (STA)* was developed by Digital Equipment Corporation, and published in the IEEE 802.1d specification.
- The STA designates a loop-free subset of the network's topology by placing those bridge ports that, if active, would create loops into a standby (blocking) condition. The STA uses a conclusion from graph theory as a basis for constructing a loop-free subset of the network's topology. Graph theory states the following:
  - For any connected graph consisting of nodes and edges connecting pairs of nodes, a spanning tree of edges maintains the connectivity of the graph but contains no loops.

Spanning-Tree

- STA akan memilih satu bridge sebagai root di dalam network.
- The STA calls for each bridge to be assigned a unique identifier.
- Typically, this identifier is one of the bridge's **MAC addresses**, plus an administratively assigned **priority**.
- Each port in every bridge also is assigned a unique identifier (within that bridge), which is typically its own MAC address.
- Finally, each bridge port is associated with a path cost, which represents the **cost** of transmitting a frame onto a LAN through that port.

Spanning-Tree



Seleksi Root Bridge

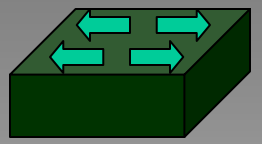
- Komunikasi antar bridge menggunakan BPDUs (Brige Protocol Data Units)
- Root bridge dipilih berdasarkan kombinasi nilai priority dan MAC address. Jika dua bridge mempunyai nilai pritoritas yang sama, maka bridge dengan MAC address lebih kecil akan dipilih sebagai root.
- BPDU dikirim setiap 2 detik by default.

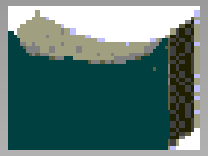
Seleksi Designated Port

- Untuk menentukan designated port, harus dilihat **path cost**-nya.
- STA cost adalah cost total berdasarkan pada bandwidth suatu link.

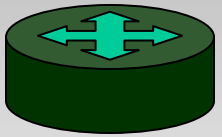
Speed	New IEEE Cost	Original IEEE Cost
10 Gbps	2	1
1 Gbps	4	1
100 Mbps	19	10
10 Mbps	100	100

Spanning-Tree





Switch/Bridge

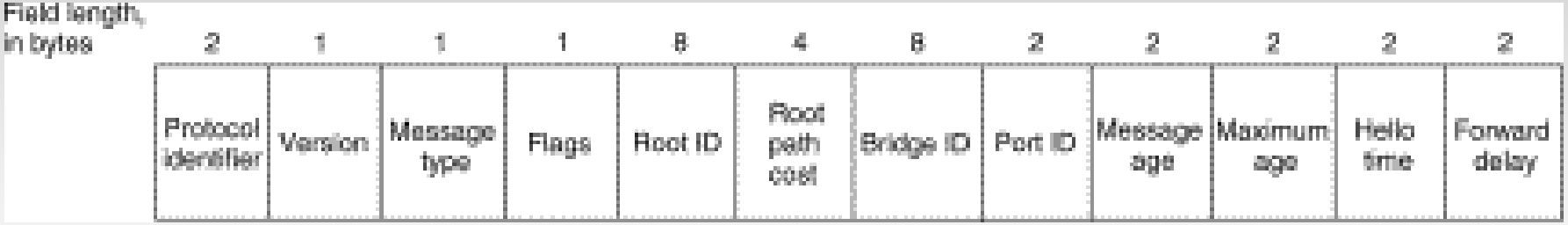


Router

- The spanning-tree calculation occurs when the bridge is powered up and whenever a topology change is detected.
- Configuration messages contain information identifying the bridge that is presumed to be the root (root identifier) and the distance from the sending bridge to the root bridge (root path cost).
- Configuration messages also contain the bridge and port identifier of the sending bridge, as well as the age of information contained in the configuration message.

BPDU Frame Format

- Transparent bridges exchange *configuration messages* and *topology-change messages*.
- Configuration messages are sent between bridges to establish a network topology.
- This is the messages format



Frame Format

- **Protocol Identifier**—Contains the value zero.
- **Version**—Contains the value zero.
- **Message Type**—Contains the value zero.
- **Flag**—Contains 1 byte, of which only 2 bits are used. The topology-change (TC) least significant bit signals a topology change. The topology-change acknowledgment (TCA) most significant bit is set to acknowledge receipt of a configuration message with the TC bit set.
- **Root ID**—Identifies the root bridge by listing its 2-byte priority followed by its
  6-byte ID.

Frame Format

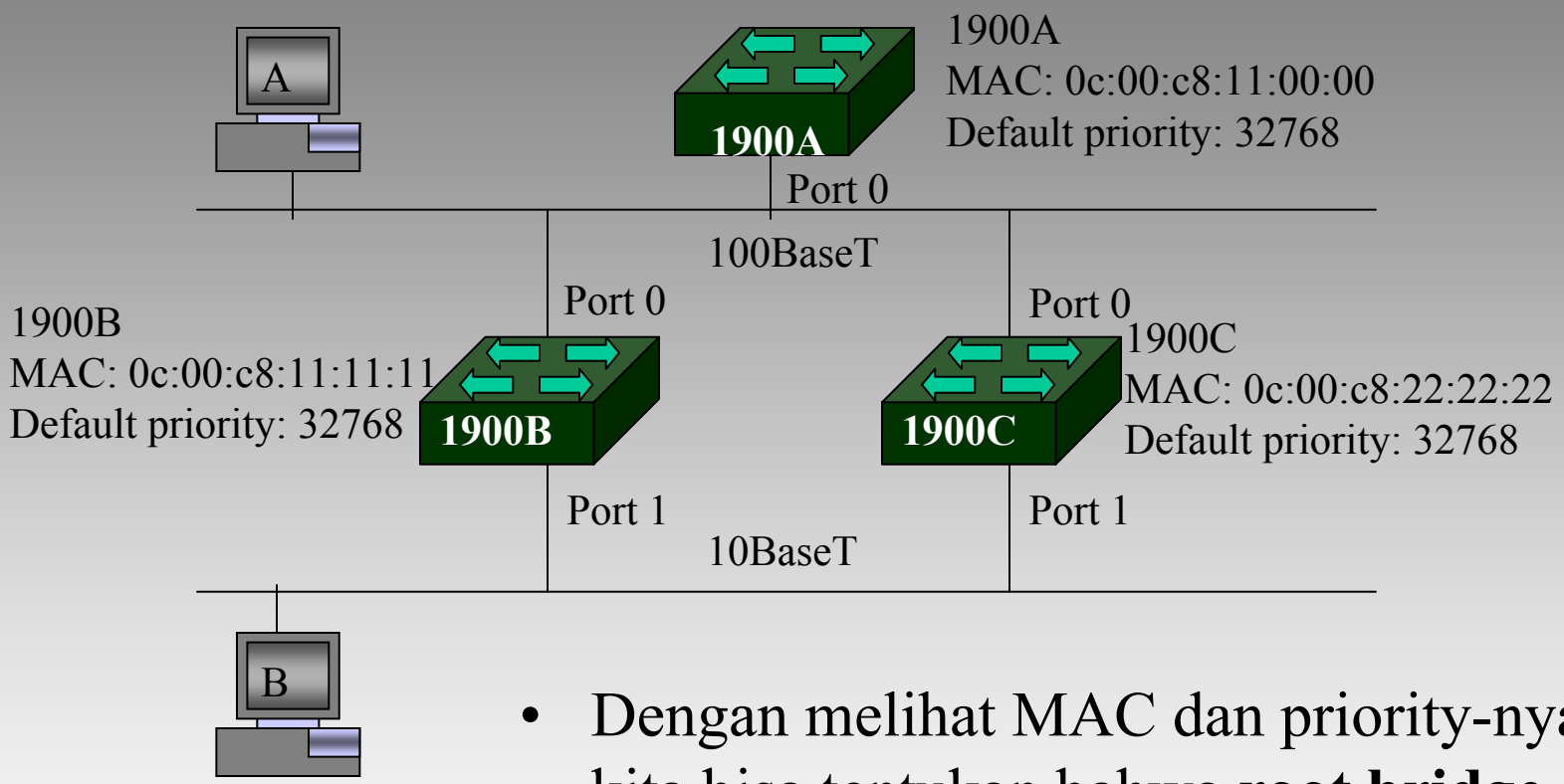
- **Root Path Cost**—Contains the cost of the path from the bridge sending the configuration message to the root bridge.
- **Bridge ID**—Identifies the priority and ID of the bridge sending the message.
- **Port ID**—Identifies the port from which the configuration message was sent. This field allows loops created by multiple attached bridges to be detected and handled.
- **Message Age**—Specifies the amount of time since the root sent the configuration message on which the current configuration message is based.

Frame Format

- **Maximum Age**—Indicates when the current configuration message should be deleted.
- **Hello Time**—Provides the time period between root bridge configuration messages.
- **Forward Delay**—Provides the length of time that bridges should wait before transitioning to a new state after a topology change. If a bridge transitions too soon, not all network links might be ready to change their state, and loops can result

Topology-change messages consist of only 4 bytes. These include a Protocol-Identifier field, which contains the value zero; a *Version* field, which contains the value zero; and a Message-Type field, which contains the value 128.

Contoh STA

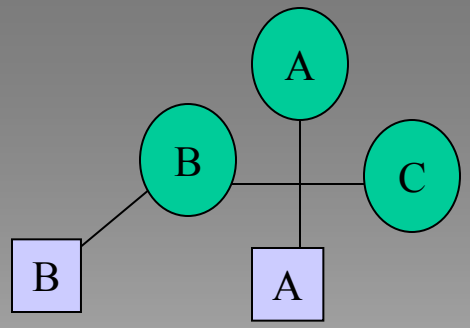


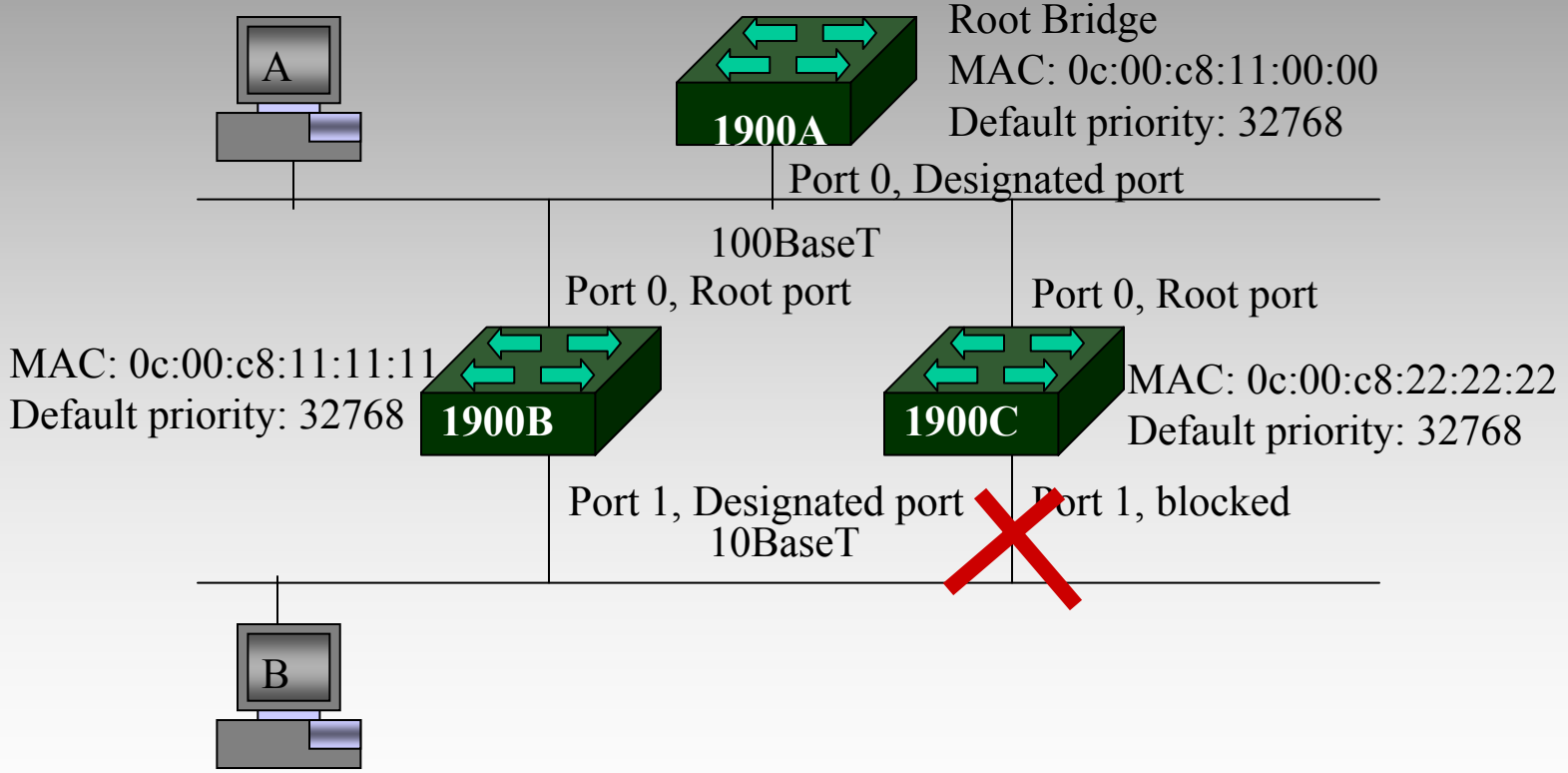
- Dengan melihat MAC dan priority-nya, kita bisa tentukan bahwa **root bridge** adalah 1900A

Contoh STA

- Untuk menentukan **root ports** pada switch 1900B dan 1900C, harus diperhatikan cost-nya. Karena dua-duanya lewat port 0 terhubung jalur 100Mbps (yang adalah yang terbaik), maka port 0 pada 1900B adalah root port, dan port 0 pada 1900C adalah juga root port.
- Penentuan **designated port**. Semua port milik root bridge adalah designated port. Walaupun 1900B dan 1900C mempunyai cost yang sama ke root bridge, designated port ditentukan pada 1900B, karena 1900B mempunyai ID yang lebih kecil.
- Karena 1900B adalah pemilik designated port, 1900C akan memblok port 1

Contoh STA


A
B
C
B
A


A
1900A
Root Bridge
MAC: 0c:00:c8:11:00:00
Default priority: 32768
Port 0, Designated port
100BaseT
Port 0, Root port
Port 0, Root port
MAC: 0c:00:c8:11:11:11
Default priority: 32768
1900B
1900C
MAC: 0c:00:c8:22:22:22
Default priority: 32768
Port 1, Designated port
10BaseT
Port 1, blocked
B

Spanning Tree States

- **Blocking**, tidak mem-forward frame, tapi mendengarkan BPDUs. (Jika switch baru saja on, semua port dalam status blocking)
- **Listening**, mendengarkan BPDU
- **Learning**, mempelajari MAC address dan membangun tabel filter/forwarding, tapi belum mem-forward frame
- **Forwarding**, mengirim dan menerima frame.

Mode switch LAN

- Store and Forward. Suatu frame telah diterima lengkap di dalam buffer Switch, CRC dijalankan, dan alamat tujuan dilihat dari tabel untuk diteruskan
- Cut-through. Switch menerima alamat tujuan, melihat ke dalam tabelnya, kemudian berdasar tabel itu memforwardkan frame.
- Fragment-free. Kadang disebut dengan modified Cut-through. Mencheck 64 bytes pertama (karena ada kemungkinan collision) sebelum mem-forward.