The Spanning Tree

802.1D (2004)

RSTP

MSTP

Problem Description

- We want redundant links in bridged networks
- But transparent bridging cannot deal with redundancy
  - Broadcast storms and other problems (see later)
- Solution: the spanning tree protocol
  - Allows for redundant paths
  - Ensures non-redundant active paths

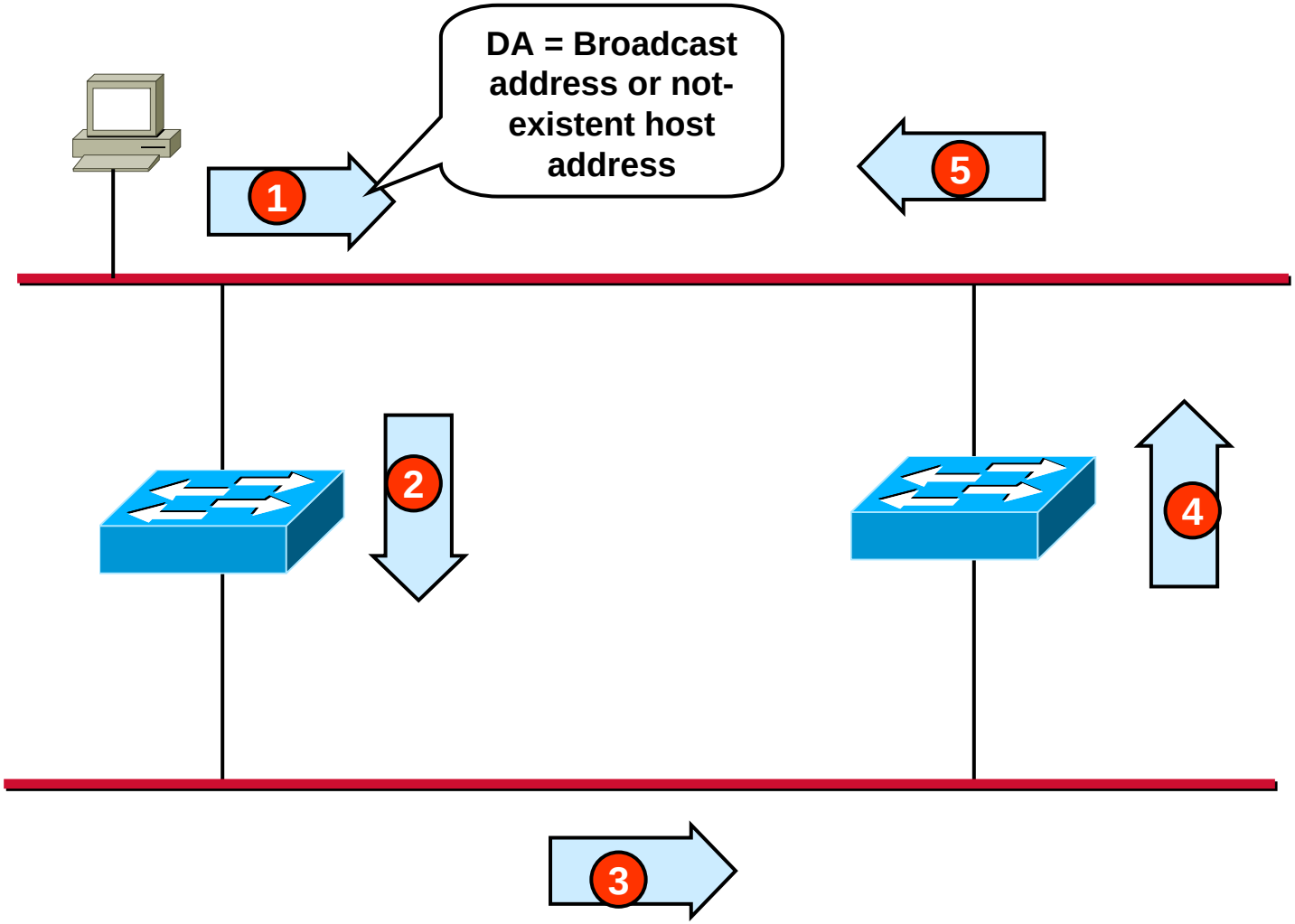
Standard STP

A short repetition of why and how

Bridging Problems

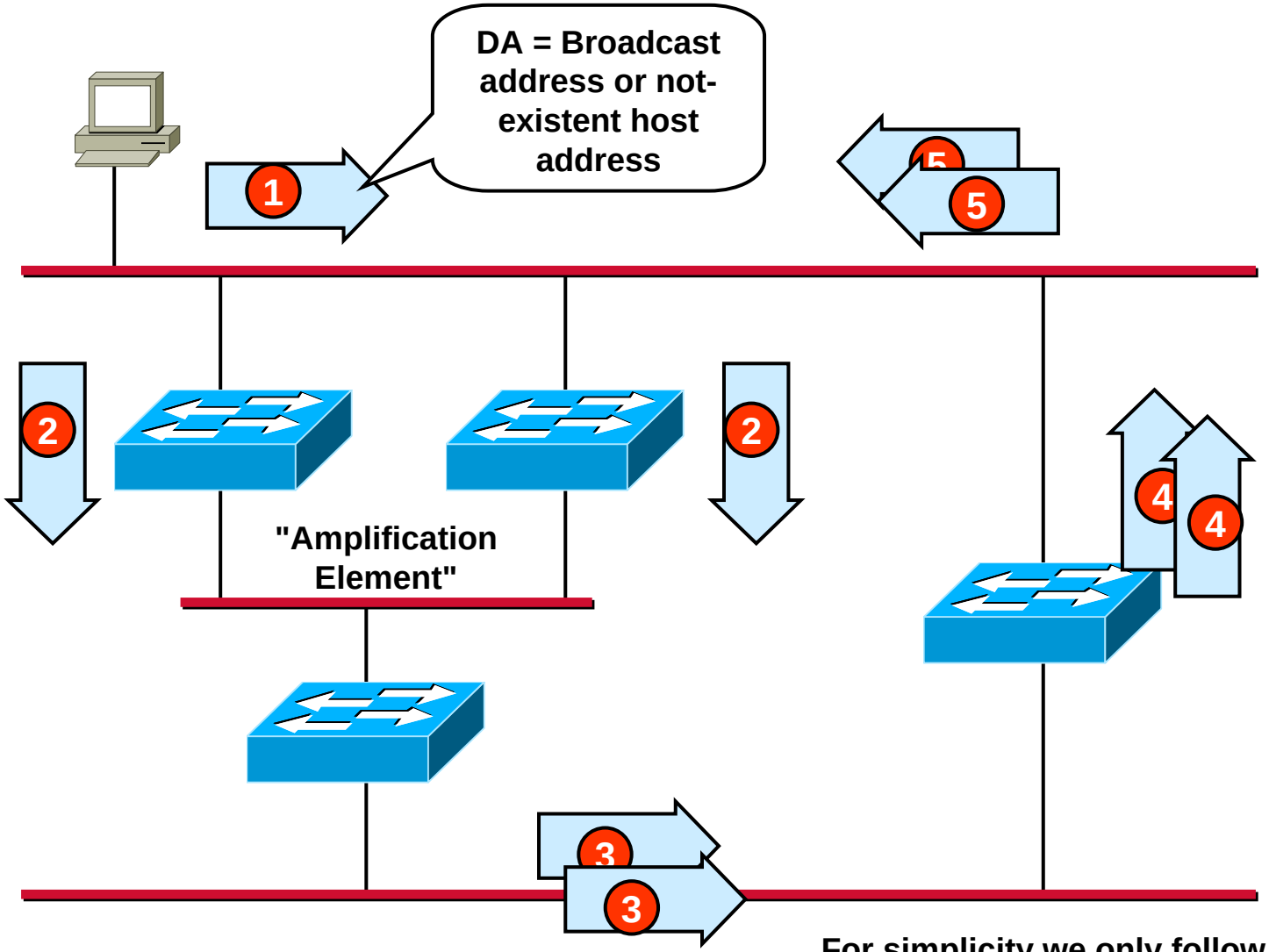
- Redundant paths lead to
  - Broadcast storms
  - Endless cycling
  - Continuous table rewriting
- No load sharing possible
- No ability to select best path

Endless Circling



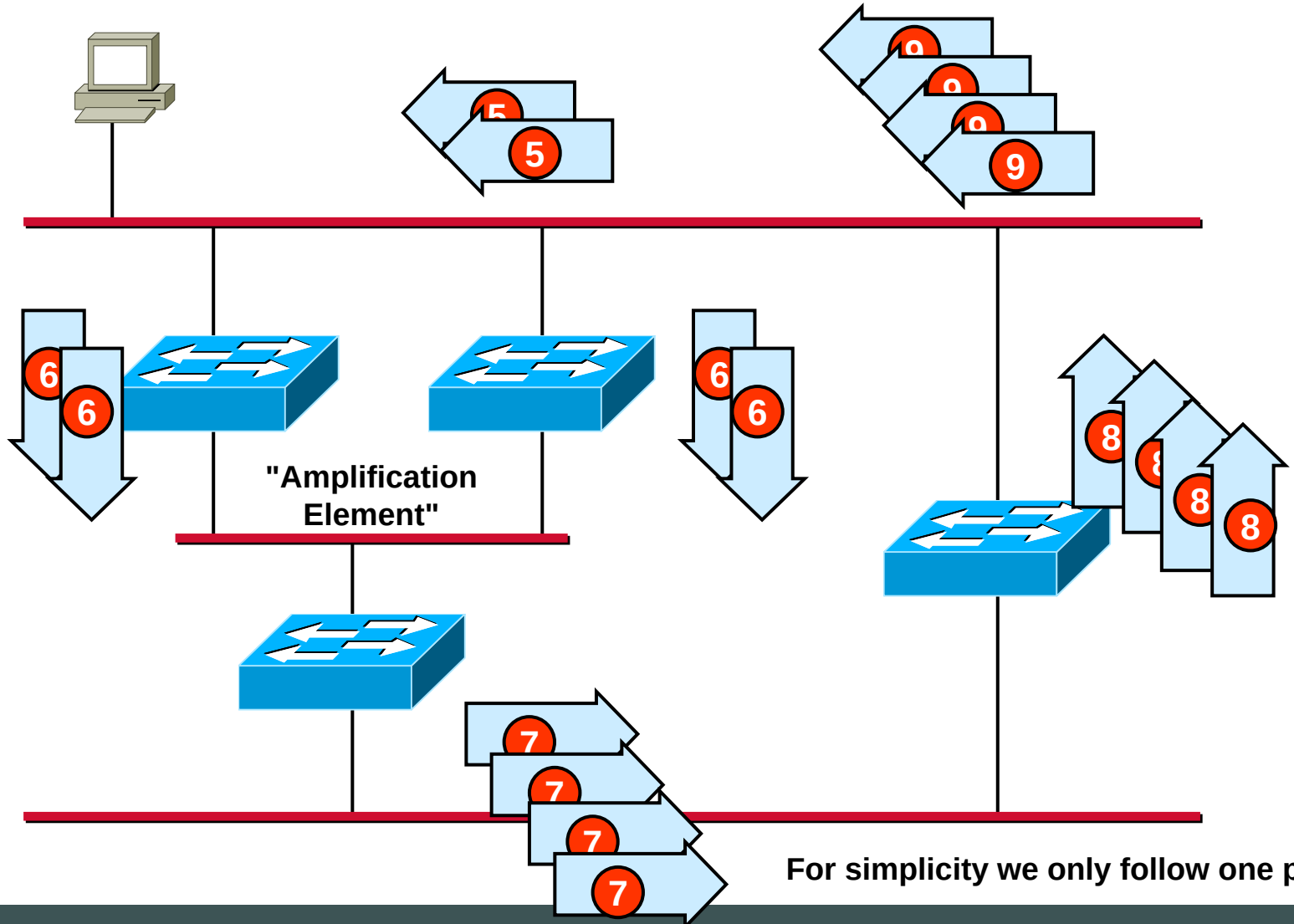
For simplicity we only follow one path

Broadcast Storm (1)

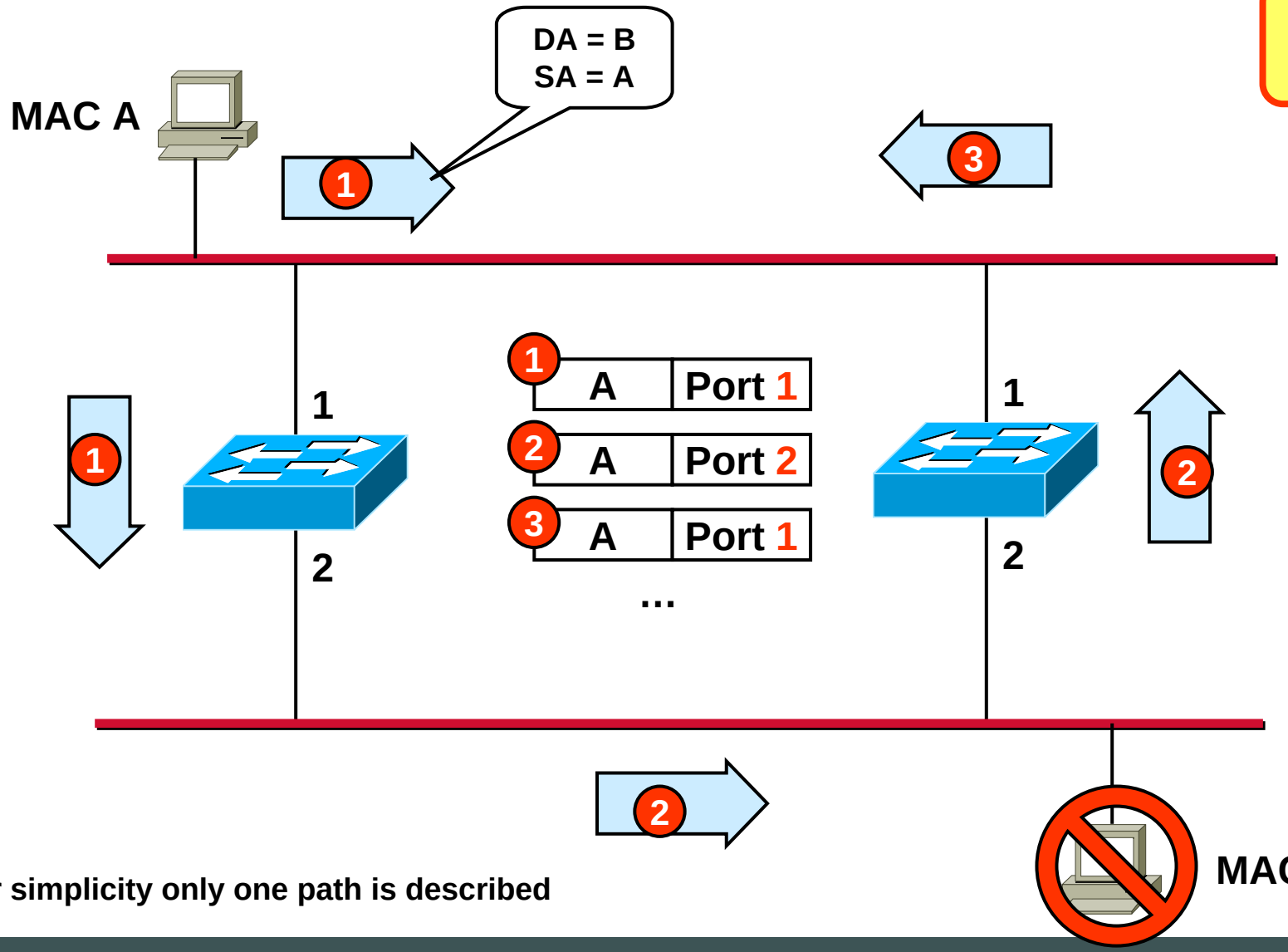


For simplicity we only follow one path

Broadcast Storm (2)



Mutual Table Rewriting



For simplicity only one path is described

The Spanning Tree

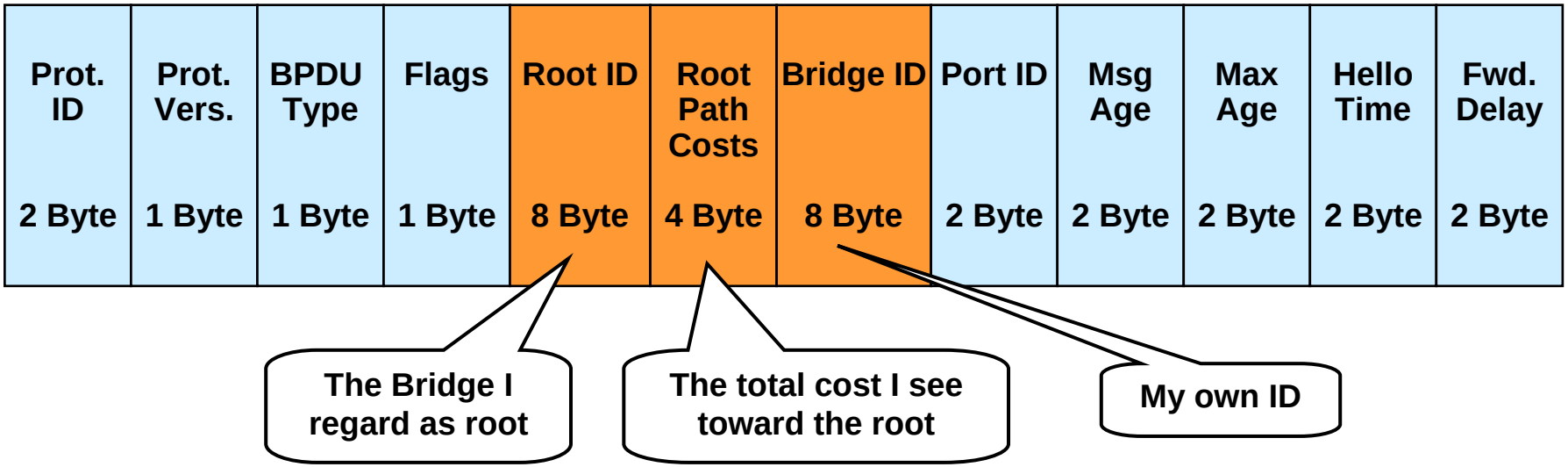
IEEE 802.1D-2004

Spanning Tree

- Invented by *Radia Perlman* as general "mesh-to-tree" algorithm
- A must in bridged networks with redundant paths
- Only one purpose: Cut off redundant paths with highest costs
- Special STP frames: Bridge Protocol Data Units (BPDUs)

BPDU Format

- Each bridge sends periodically BPDUs carried in Ethernet multicast frames
  - Hello time default: 2 seconds
- Contains all information necessary for building Spanning Tree



Three STP Parameters

- 8 byte Bridge-ID for each bridge
  - Consists of 2 byte Priority value (default 32768) and 6 byte (lowest) MAC address
  - Used to determine root bridge and as tie-breaker to when determing designated port
- 4 byte Port Cost for each port
  - Old (still used) standard method: 1000 / Port_BW_in_Mbits
    - E. g. 10 Mbit/s ➔ Cost=100
  - Used to calculate Root Path Cost to determine root port and designated port
- 2 byte Port-ID for each port
  - Consists of 1 byte Priority value (default 128) and 1 byte port number
  - Only used as tie-breaker if the same Bridge-ID and the same Path Cost is received on multiple ports

STP Port Cost

Speed [Mbit/s]	Old Cost (1000/Speed)	New Cost	802.1T
10	100	100	2,000,000
100	10	19	200,000
155	6	14	(129032 ?)
622	1	6	(32154 ?)
1000	1	4	20,000
10000	1	2	2,000

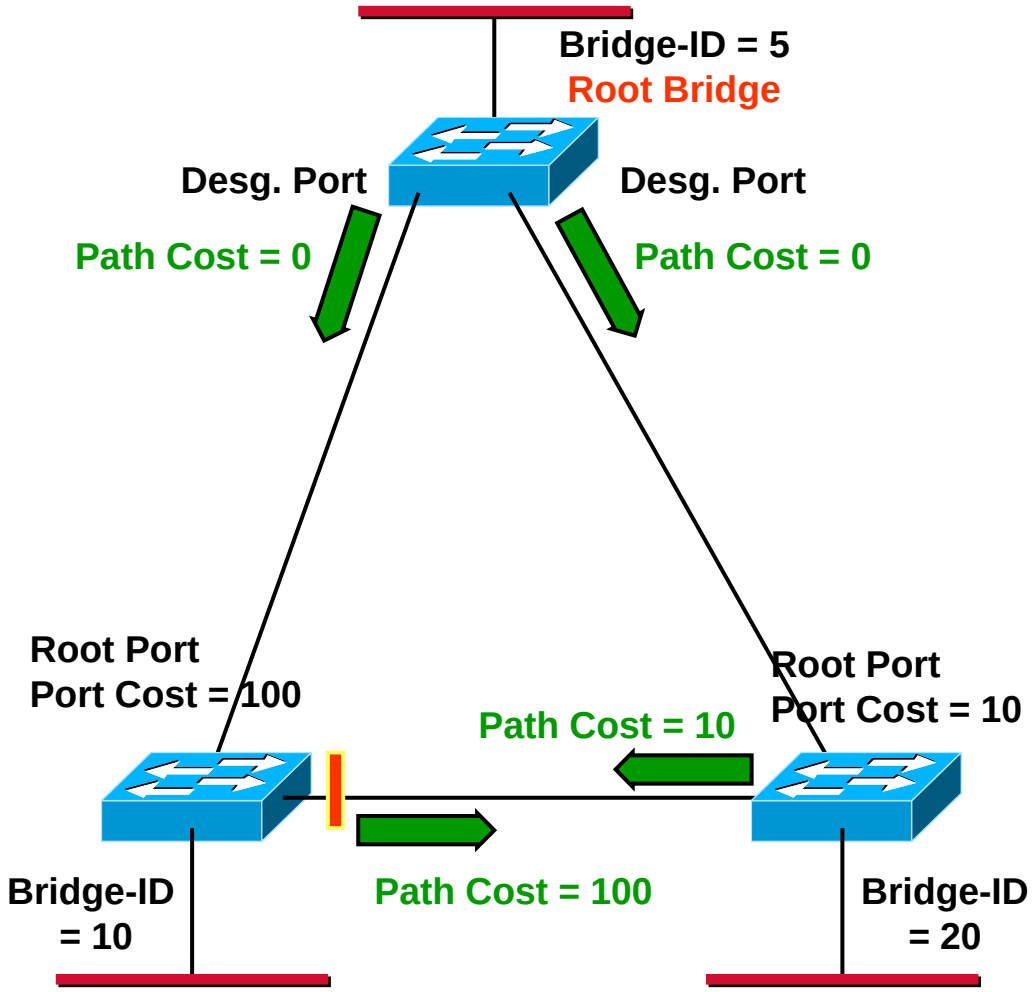
- **Also different cost values might be used**
  - **See recommendations in the IEEE 802.1D-2004 standard to comply with RSTP and MSTP**

802.1T Excerpt

Link Speed	Recommended value	Recommended range	Range
<=100 Kb/s	200 000 000*	20 000 000–200 000 000	1–200 000 000
1 Mb/s	20 000 000[a]	2 000 000–200 000 000	1–200 000 000
10 Mb/s	2 000 000[a]	200 000–20 000 000	1–200 000 000
100 Mb/s	200 000[a]	20 000–2 000 000	1–200 000 000
1 Gb/s	20 000	2 000–200 000	1–200 000 000
10 Gb/s	2 000	200–20 000	1–200 000 000
100 Gb/s	200	20–2 000	1–200 000 000
1 Tb/s	20	2–200	1–200 000 000
10 Tb/s	2	1–20	1–200 000 000

*Bridges conformant to IEEE Std 802.1D, 1998 Edition, i.e., that support only 16-bit values for Path Cost, should use 65 535 as the Path Cost for these link speeds when used in conjunction with Bridges that support 32-bit Path Cost values.

STP Basic Principle

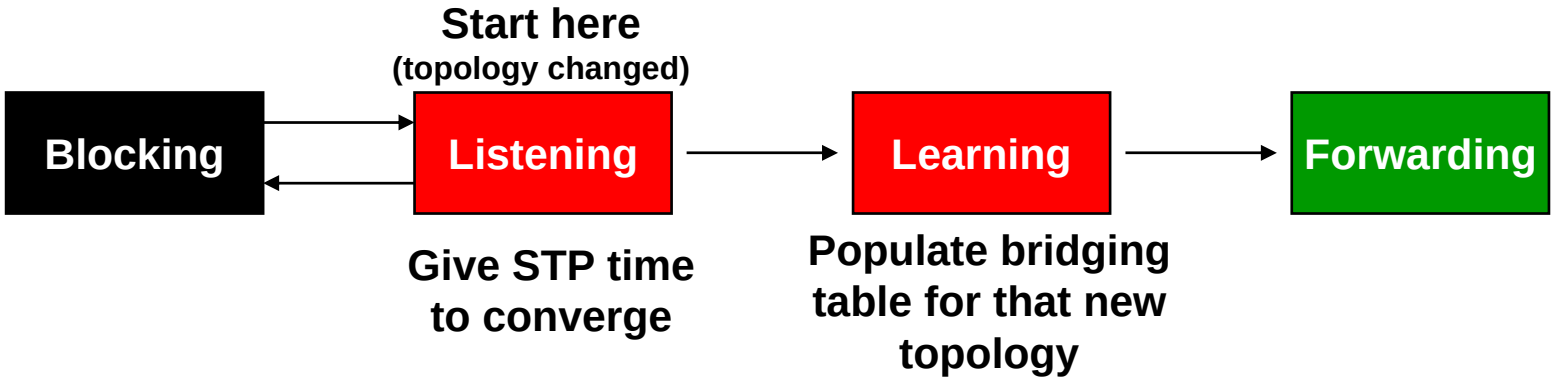


- First the Root Bridge is determined
  - Initially every bridge assumes itself as root
  - The bridge with lowest Bridge-ID wins
- Then the root bridge triggers transmissions of BDPUs
  - In hello time intervals (2 s)
  - Received at "Root Ports" by other bridges
  - Every bridge adds its own port cost to the advertised path cost and forwards the BPDU
- On each LAN segment one bridge becomes Designated Bridge
  - Having lowest root path cost
  - Other bridges set their (redundant) ports in blocking state

Final situation

- **Root switch**
  - **Has only Designated Ports**
  - **All in forwarding state**
- **Other switches have**
  - **Exactly one Root Port (upstream)**
  - **Zero or more Designated Ports (downstream)**
  - **Zero or more Nondesignated Ports (blocked)**

Port States



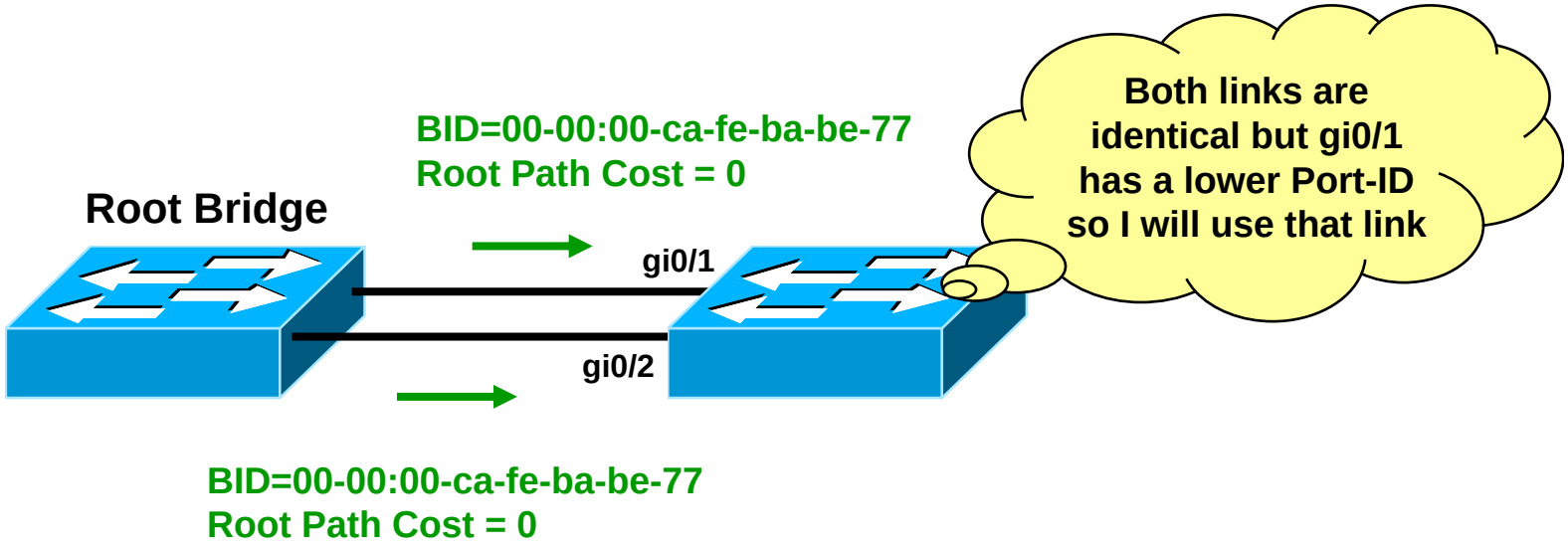
- At each time, a port is in one of the following states:
  - Blocking, Listening, Learning, Forwarding, or Disabled
- Only Blocking or Forwarding are final states (for enabled ports)
- Transition states
  - 15 s Listening state is used to converge STP
  - 15 s Learning state is used to learn MAC addresses for the new topology
- Therefore it lasts 30 seconds until a port is placed in forwarding state

Note

- Redundant links remain in active stand-by mode
  - If root port fails, other root port becomes active
- Only 7 bridges per path allowed according standard (!)
  - Because of 15 seconds listening state and 2 seconds hello timers

Usage for a Port-ID

- The Port-ID is only used as last tie-breaker
- Typical situation in highly redundant topologies: Multiple links between each two switches
  - Same BID and Costs announced on each link
  - Only local Port-ID can choose a single link



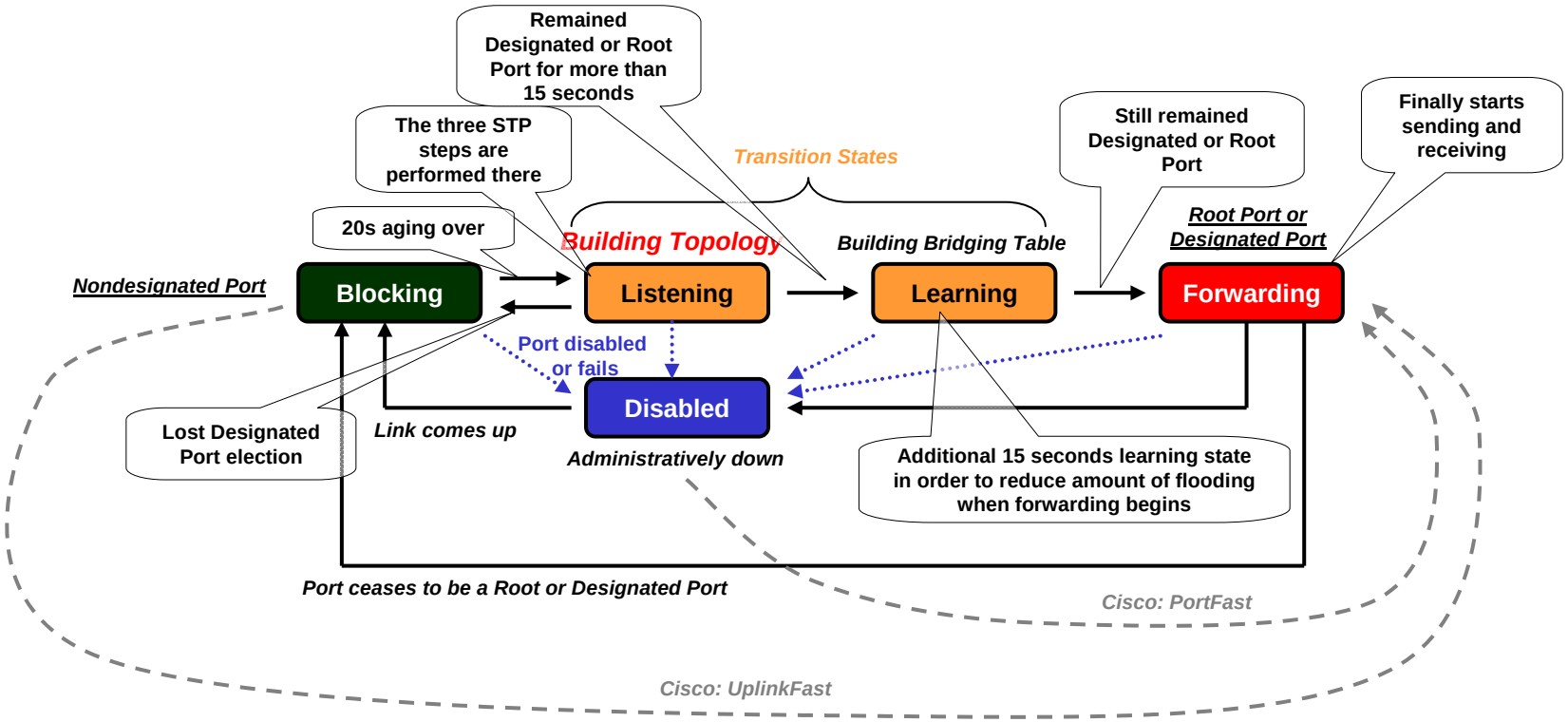
Importance of details…

- Many people think STP is a simple thing – until they encounter practical problems in real networks
- Important Details
  - STP State Machine
  - BPDU format details
  - TCN mechanism
  - RSTP
  - MSTP

Note: STP is a port-based algorithm

- Only the root-bridge election is done on the bridge-level
- All other processing is port-based
  - To establish the spanning tree, each enabled port is either forwarding or blocking
  - Additionally two transition states have been defined

STP State Machine: Port Transition Rules



- **STP is completely performed in the Listening state**
  - Blocking ports still receive BPDUs (but don‘t send)
- **Default convergence time is 30-50 s**
  - 20s aging, (15+15)s transition time
- **Timer tuning: Better don't do it !**
  - Only modify timers of the root bridge
  - Don't forget values on supposed backup root bridge

802.1d defines port roles and states:

Port Roles
Root
Designated
Nondesignated

Port States
Disabled
Blocking
Listening
Learning
Forwarding

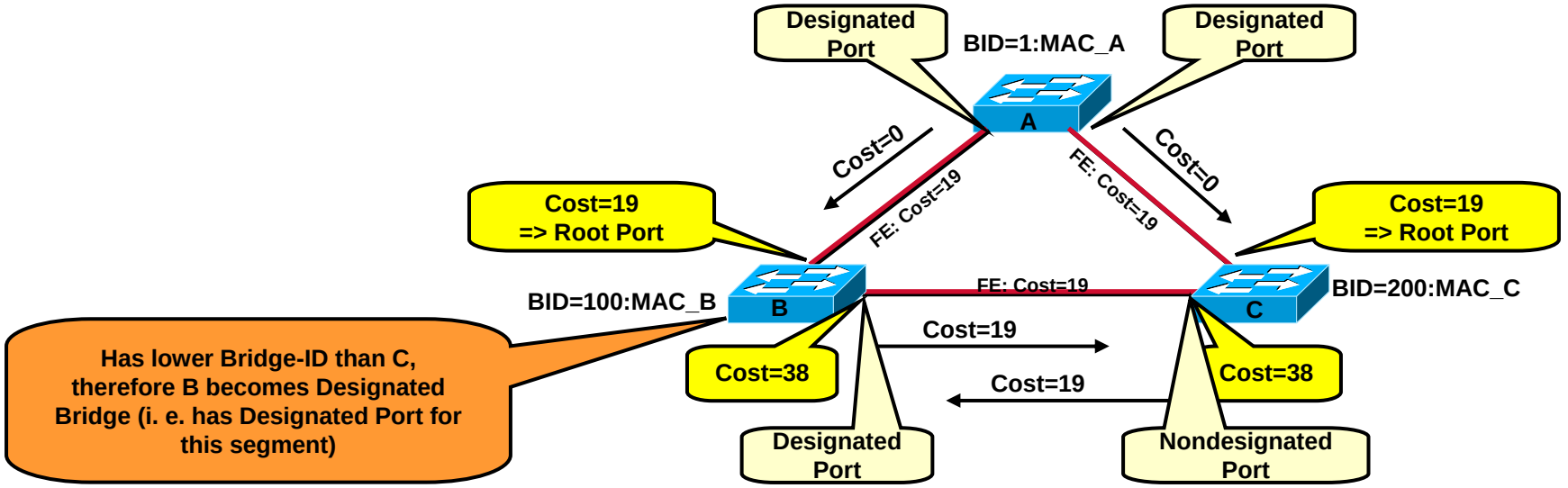
Another Example

Three steps to create spanning tree:

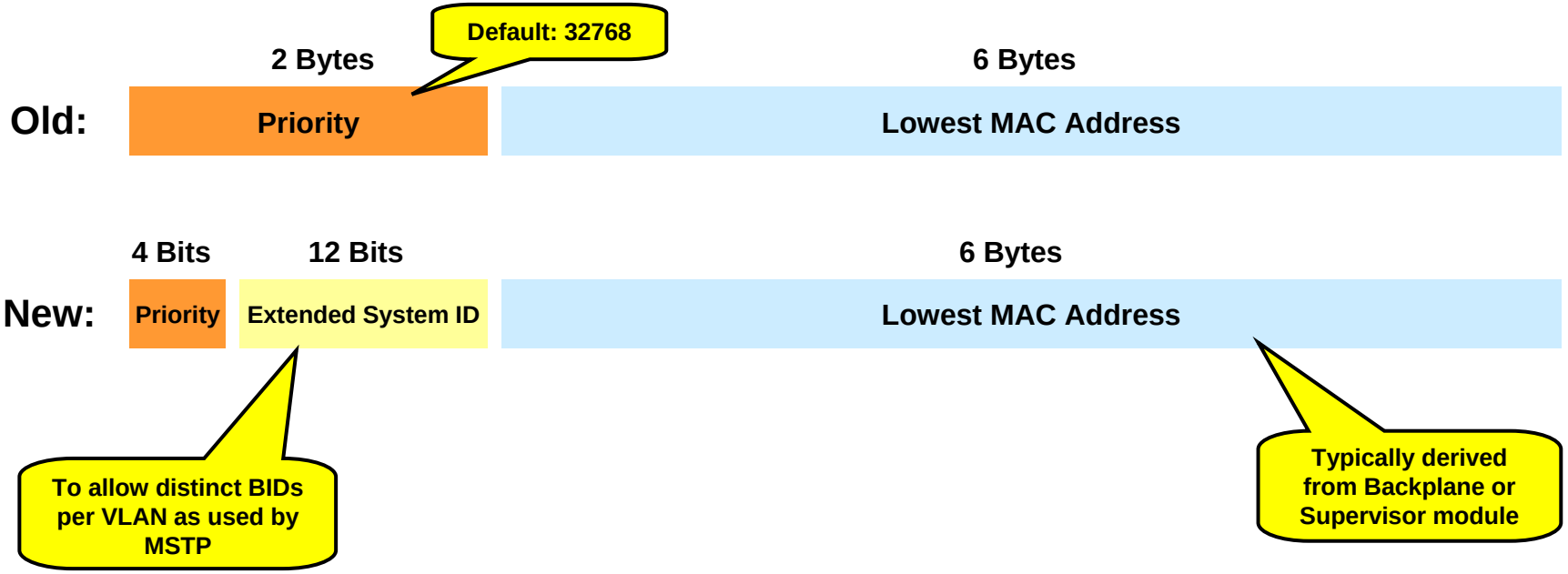
1. Elect Root Bridge (Each L2-network has exactly one Root Bridge)
2. Elect Root Ports (Each non-root bridge has exactly one Root Port)
3. Elect Designated Ports (Each segment has exactly one Designated Port)

To determine root port and designated port:

1. Determine lowest (cumulative) Path Cost to Root Bridge
2. Determine lowest Bridge ID
3. Determine lowest Port ID

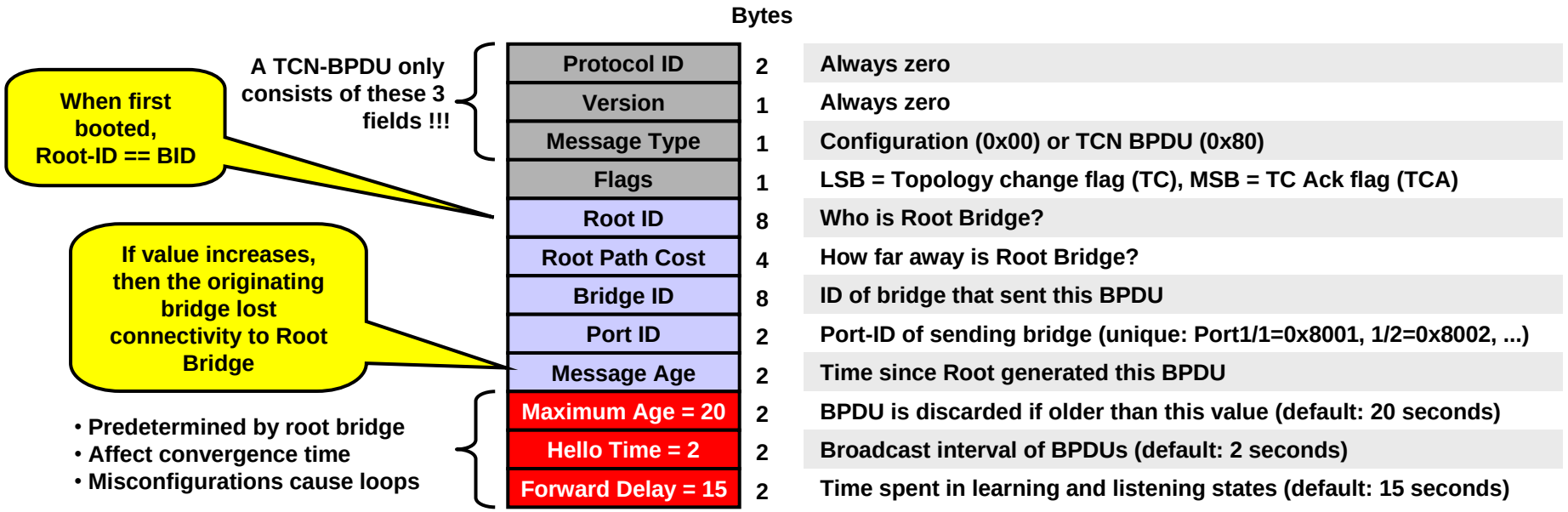


Components of the Bridge-ID



- The recent 802.1D-2004 standard requires only 4-bits for priority and 12 bits to distinguish multiple STP instances
  - Typically used for MSTP, where each set of VLANs has its own STP topology
- Therefore, ascending priority values are 0, 4096, 8192, …
  - Typically still configured as 0, 1, 2, 3 …

Detailed BPDU Format



	Bytes	
Protocol ID	2	Always zero
Version	1	Always zero
Message Type	1	Configuration (0x00) or TCN BPDU (0x80)
Flags	1	LSB = Topology change flag (TC), MSB = TC Ack flag (TCA)
Root ID	8	Who is Root Bridge?
Root Path Cost	4	How far away is Root Bridge?
Bridge ID	8	ID of bridge that sent this BPDU
Port ID	2	Port-ID of sending bridge (unique: Port1/1=0x8001, 1/2=0x8002, ...)
Message Age	2	Time since Root generated this BPDU
Maximum Age = 20	2	BPDU is discarded if older than this value (default: 20 seconds)
Hello Time = 2	2	Broadcast interval of BPDUs (default: 2 seconds)
Forward Delay = 15	2	Time spent in learning and listening states (default: 15 seconds)

- BPDUs are sent in 802.3 frames
  - DA = 01-80-C2-00-00-00
  - LLC has DSAP=SSAP = 0x42 ("the answer")
- Configuration BPDUs
  - Originated by Root Bridge periodically (2 sec Hello Time), flow downstream

Topology Change Notification (TCN)

- **Special BPDUs, used as alert by any bridge**
  - **Flow upstream (through Root Port)**
  - **Only consists of the first three standard header fields!**
- **Sent upon**
  - **Transition of a port into Forwarding state and at least one Designated Port exists**
  - **Transition of a port into Blocking state (from either Forwarding or Learning state)**
- **Sent until acknowledged by TC Acknowledge (TCA)**

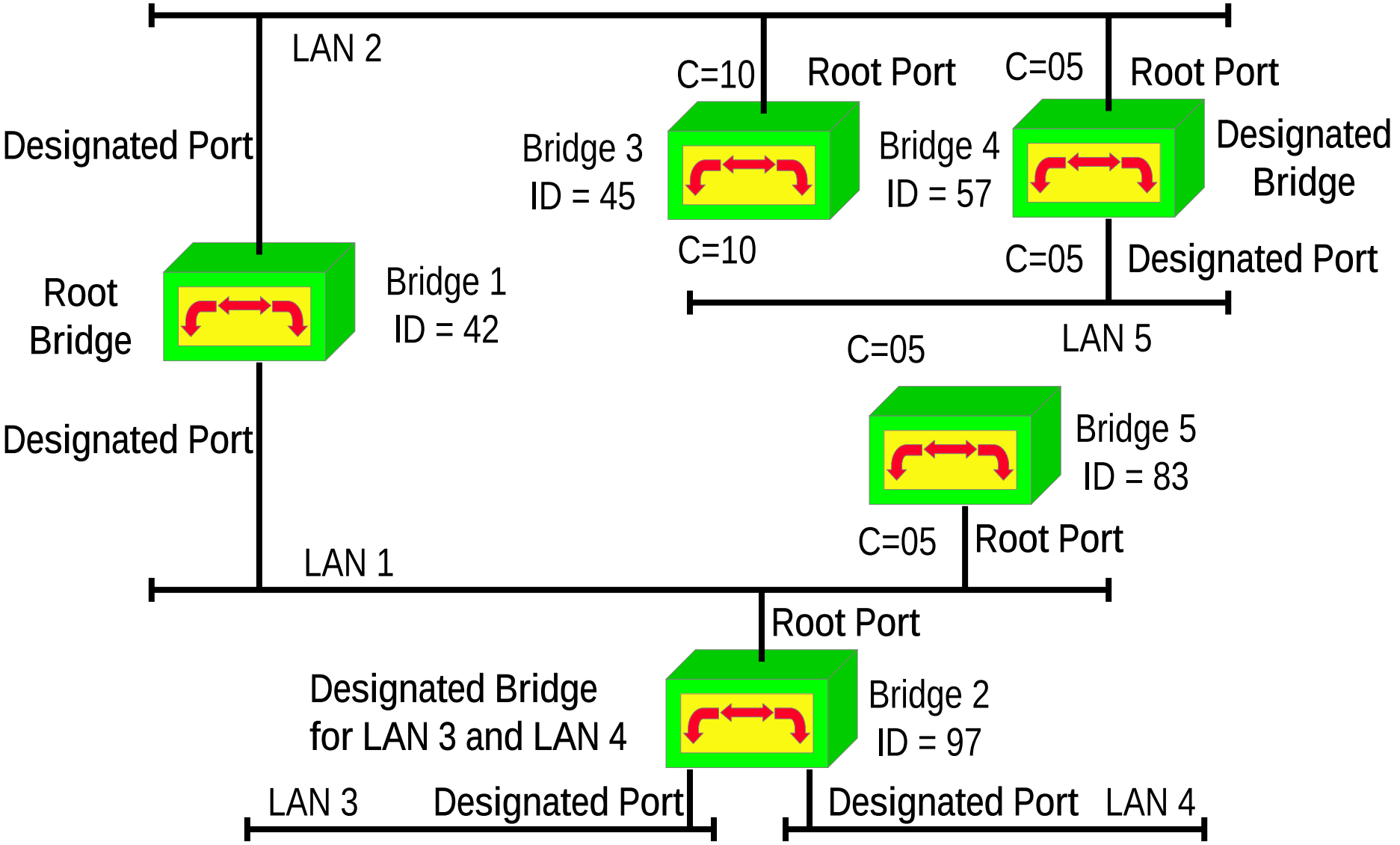
Topology Change Notification (TCN)

- Only the Designated Ports of upstream bridges processes TCN-BPDUs and send TC-Ack (TCA) downstream
- Finally the Root Bridge receives the TC and sends Configuration BPDUs with the TC flag set to 1 (=TCA) downstream for (Forward Delay + Max Age = 35) seconds
  - This instructs all bridges to reduce the default bridging table aging (300 s) to the current Forward Delay value (15 s)
  - Thus bridging tables can adapt to the new topology

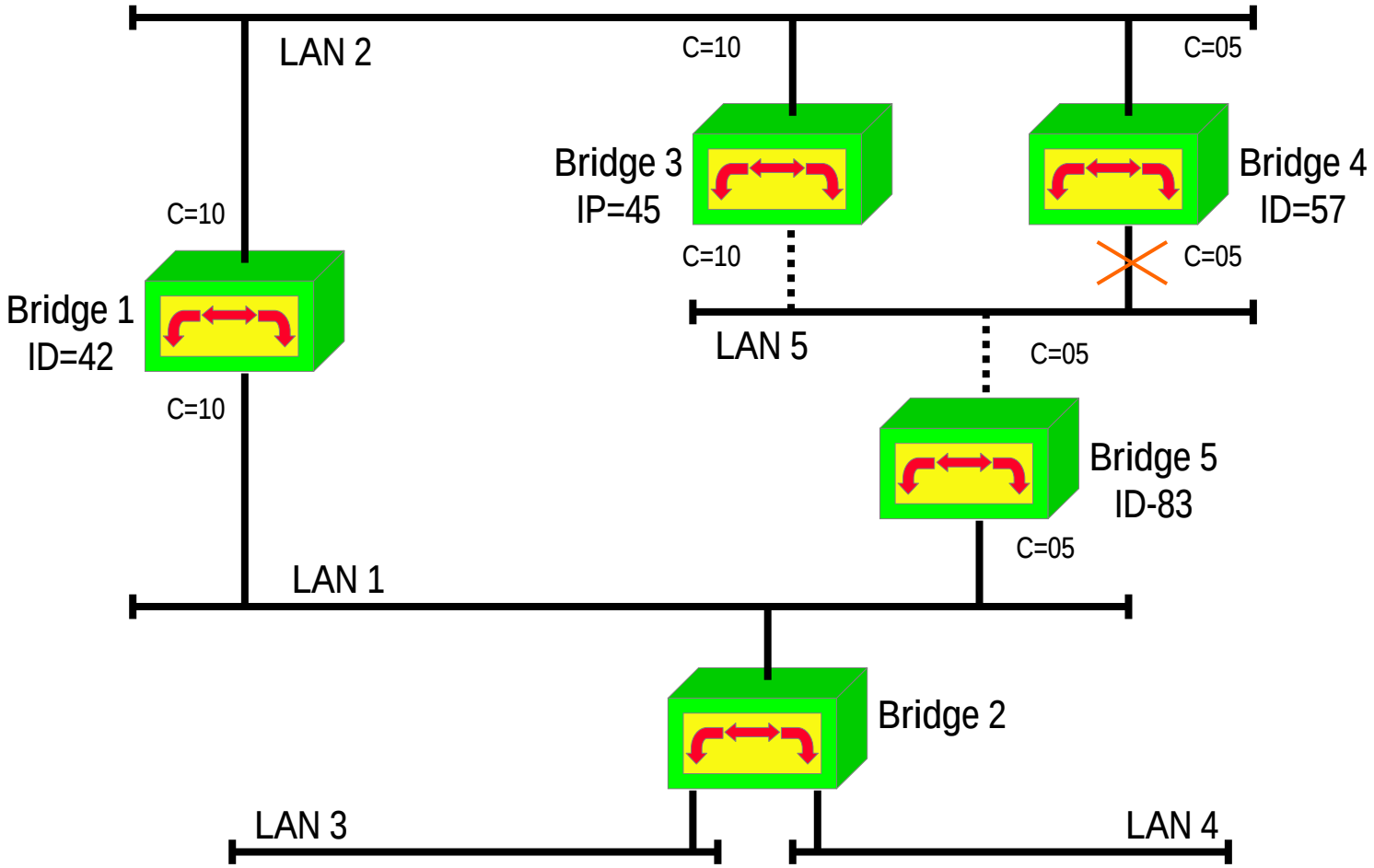
STP Error Detection

- **normally the root bridge generates (triggers)**
  - every 1-10 seconds (hello time interval) a Configuration BPDU to be received on the root port of every other bridge and carried on through the designated ports
  - bridges which are not designated are still listening to such messages on blocked ports
- **if triggering ages out two scenarios are possible**
  - root bridge failure
    - a new root bridge will be selected based on the lowest Bridge-ID and the whole spanning tree may be modified
  - designated bridge failure
    - if there is an other bridge which can support a LAN segment this bridge will become the new designated bridge

Spanning Tree Applied

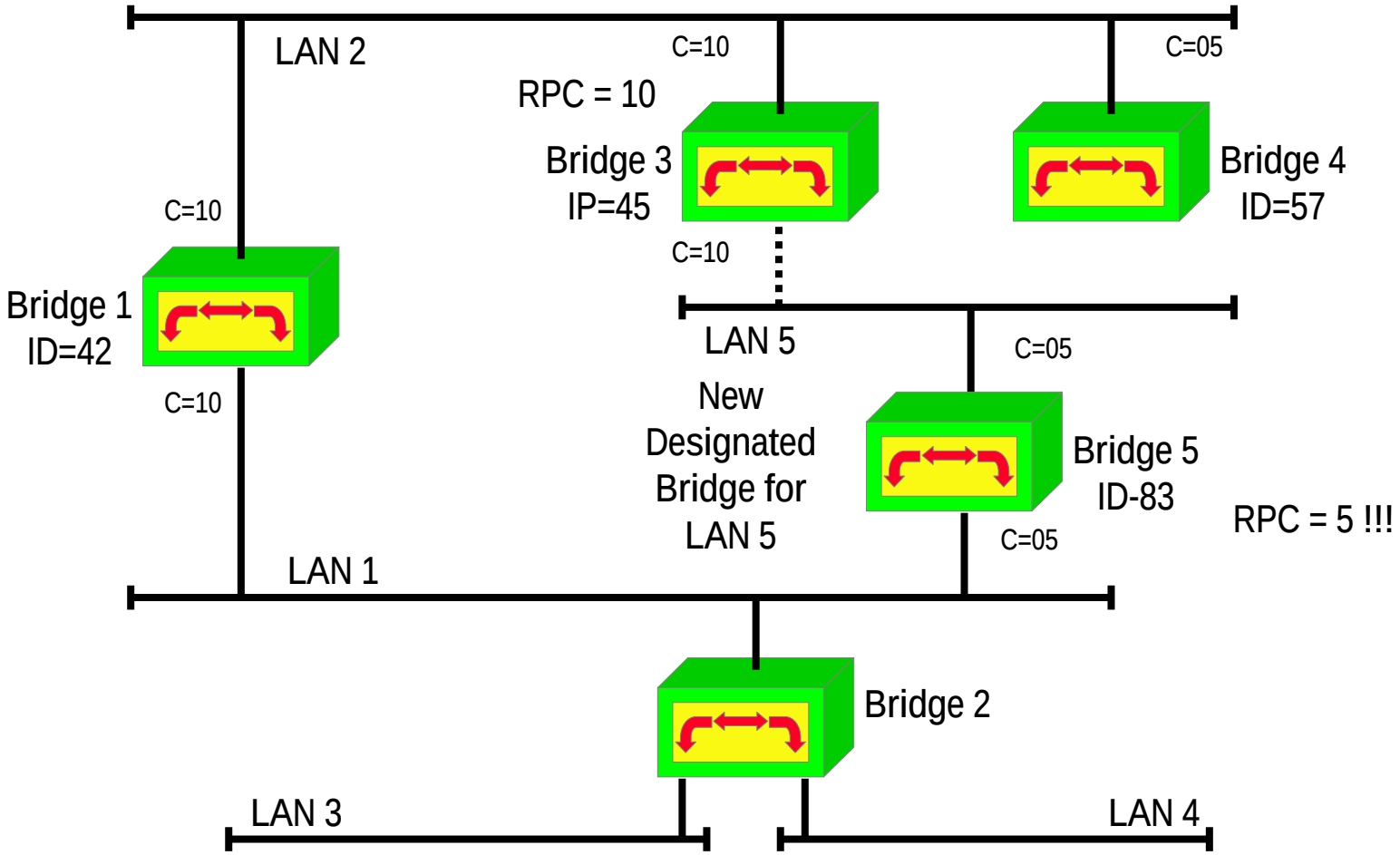


STP Convergence Time – Failure at Designated Bridge



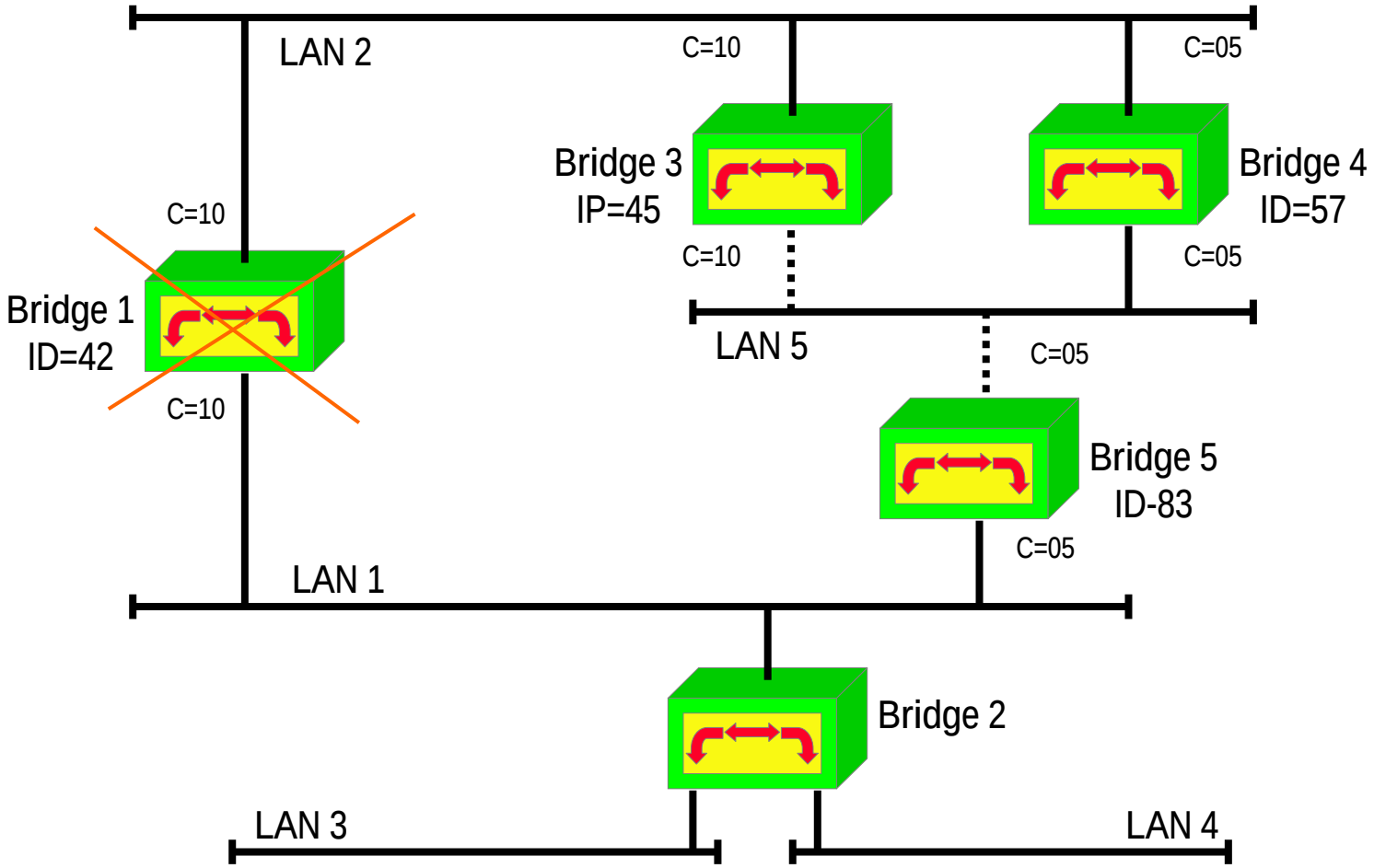
- **Time = max age (20 sec) + 2*forward delay (15 sec Listening + 15 sec Learning) = 50 sec**

STP Convergence Time – Failure at Designated Bridge – New Topology



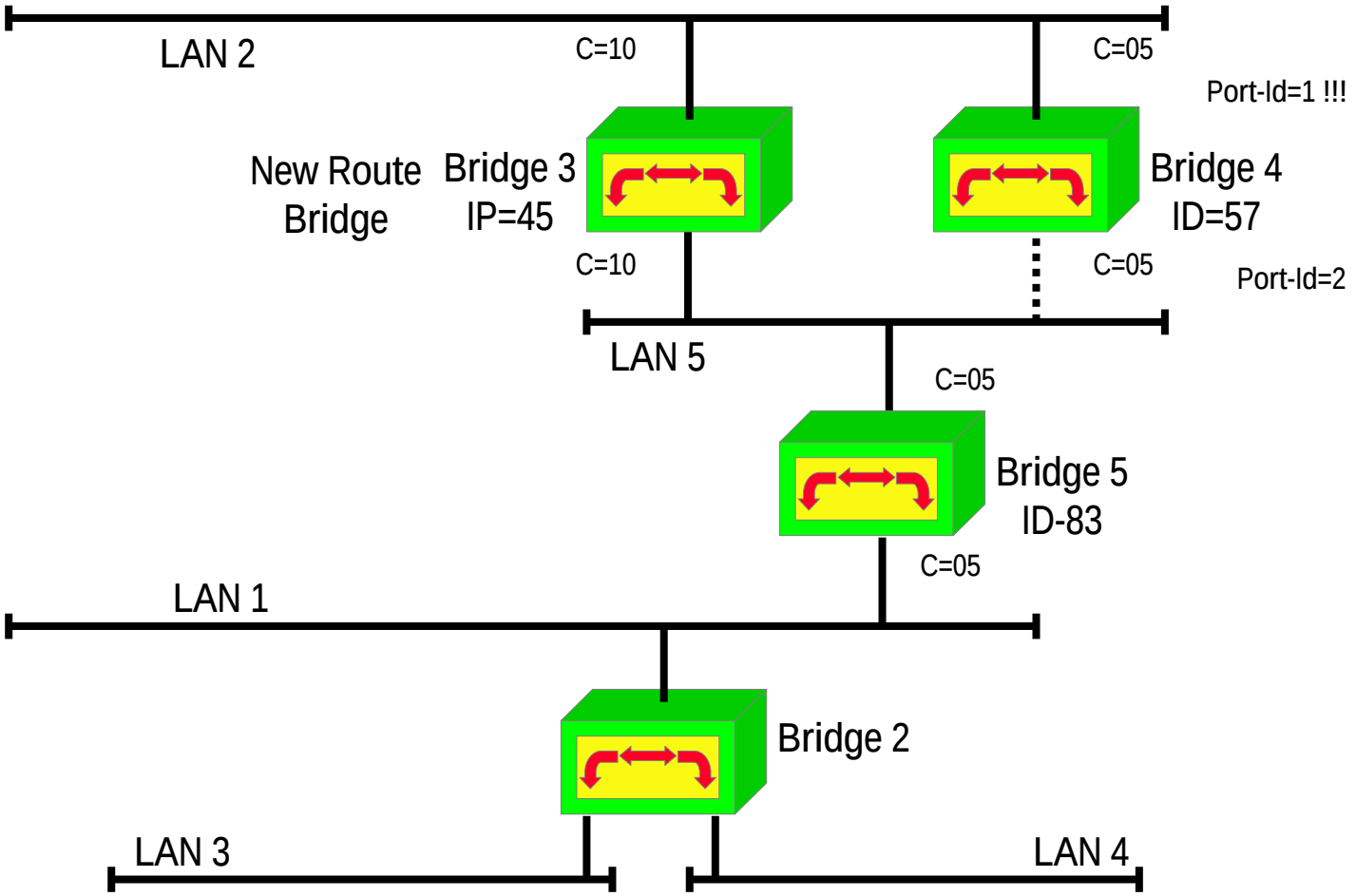
- **Time = max age (20 sec) + 2*forward delay (15 sec Listening + 15 sec Learning) = 50 sec**

STP Convergence Time – Failure of Root Bridge



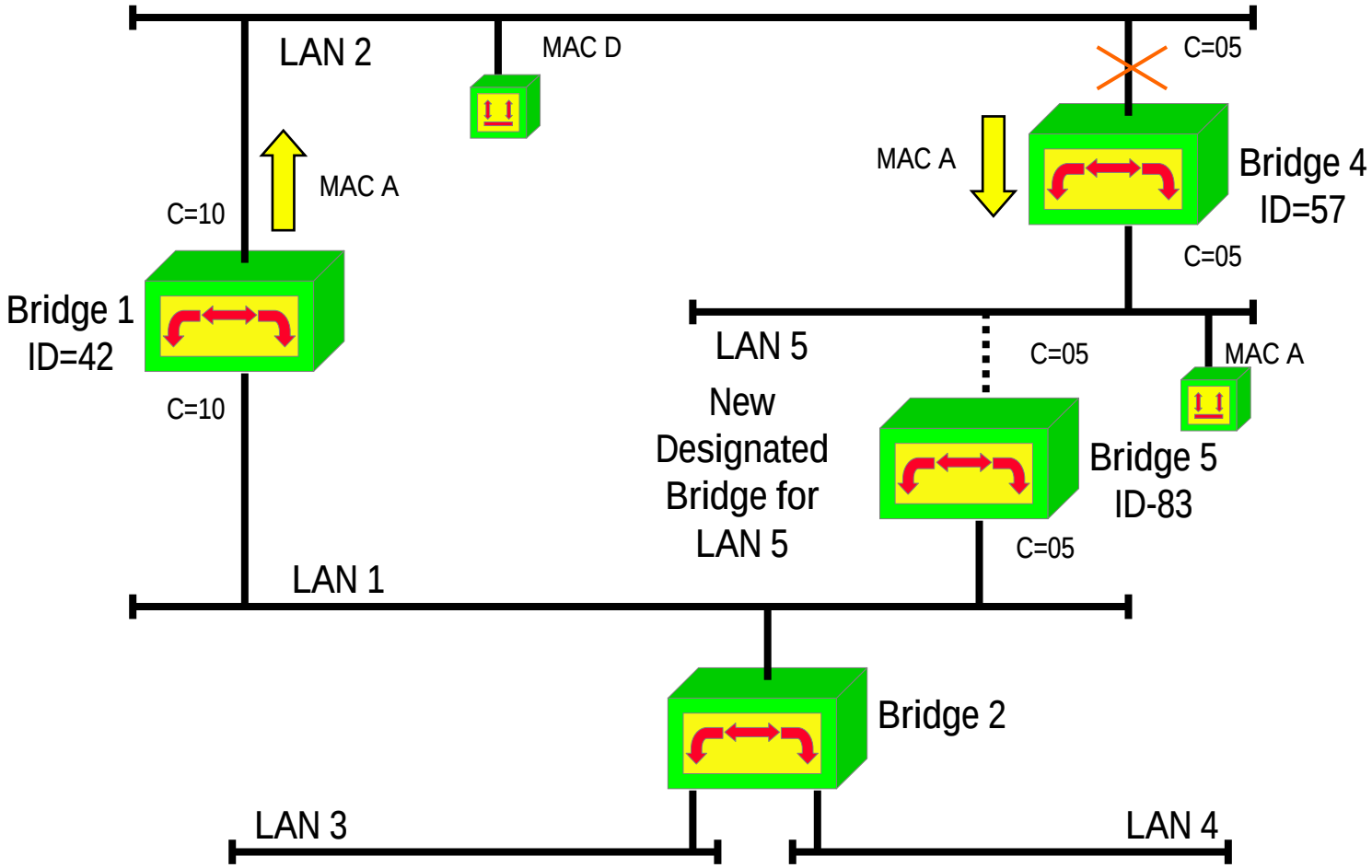
- **Time = max age (20 sec) + 2*forward delay (15 sec Listening + 15 sec Learning) = 50 sec**

STP Convergence Time – Failure of Root Bridge – New Topology



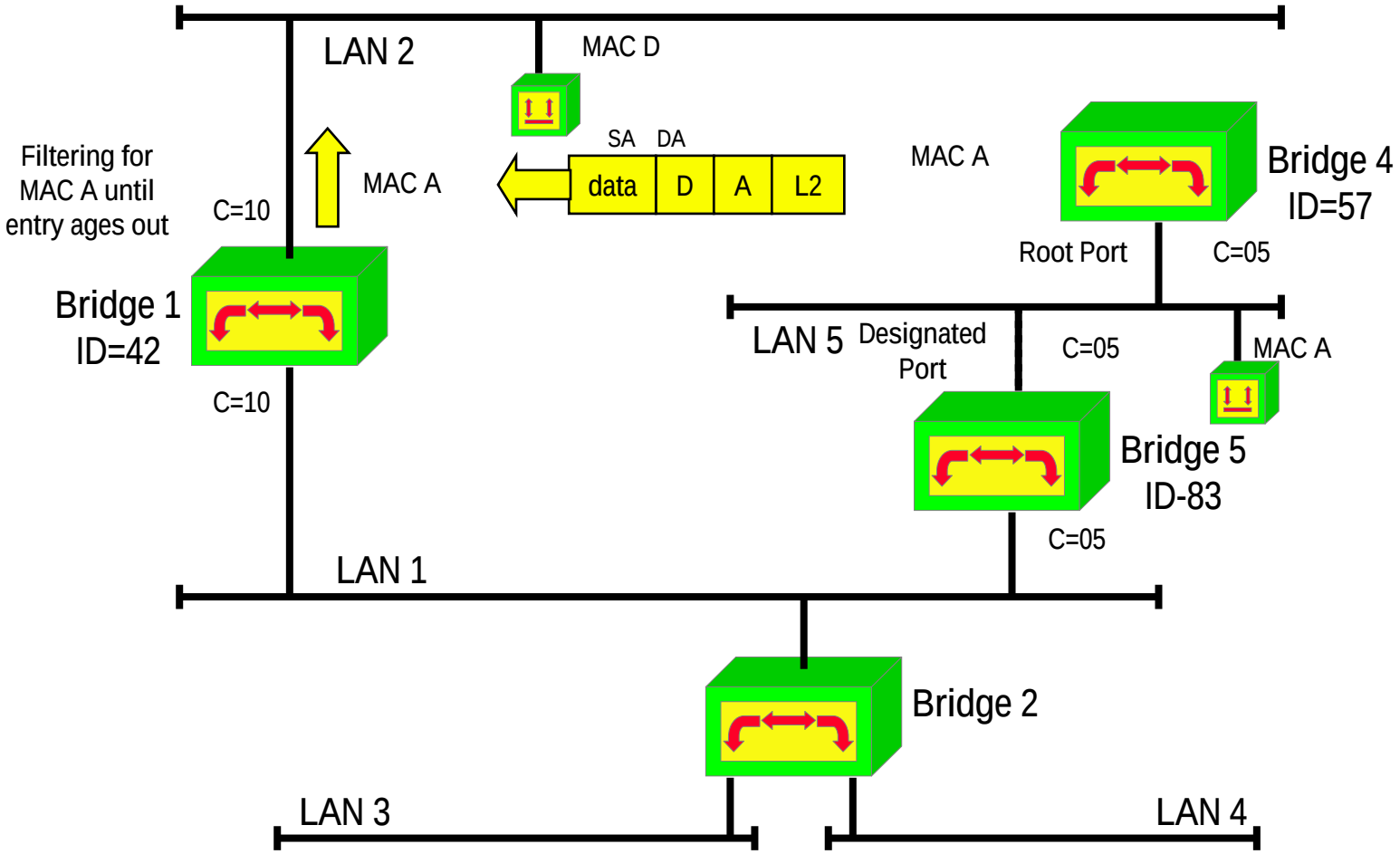
- **Time = max age (20 sec) + 2*forward delay (15 sec Listening + 15 sec Learning) = 50 sec**

STP Convergence Time – Failure of Root Port



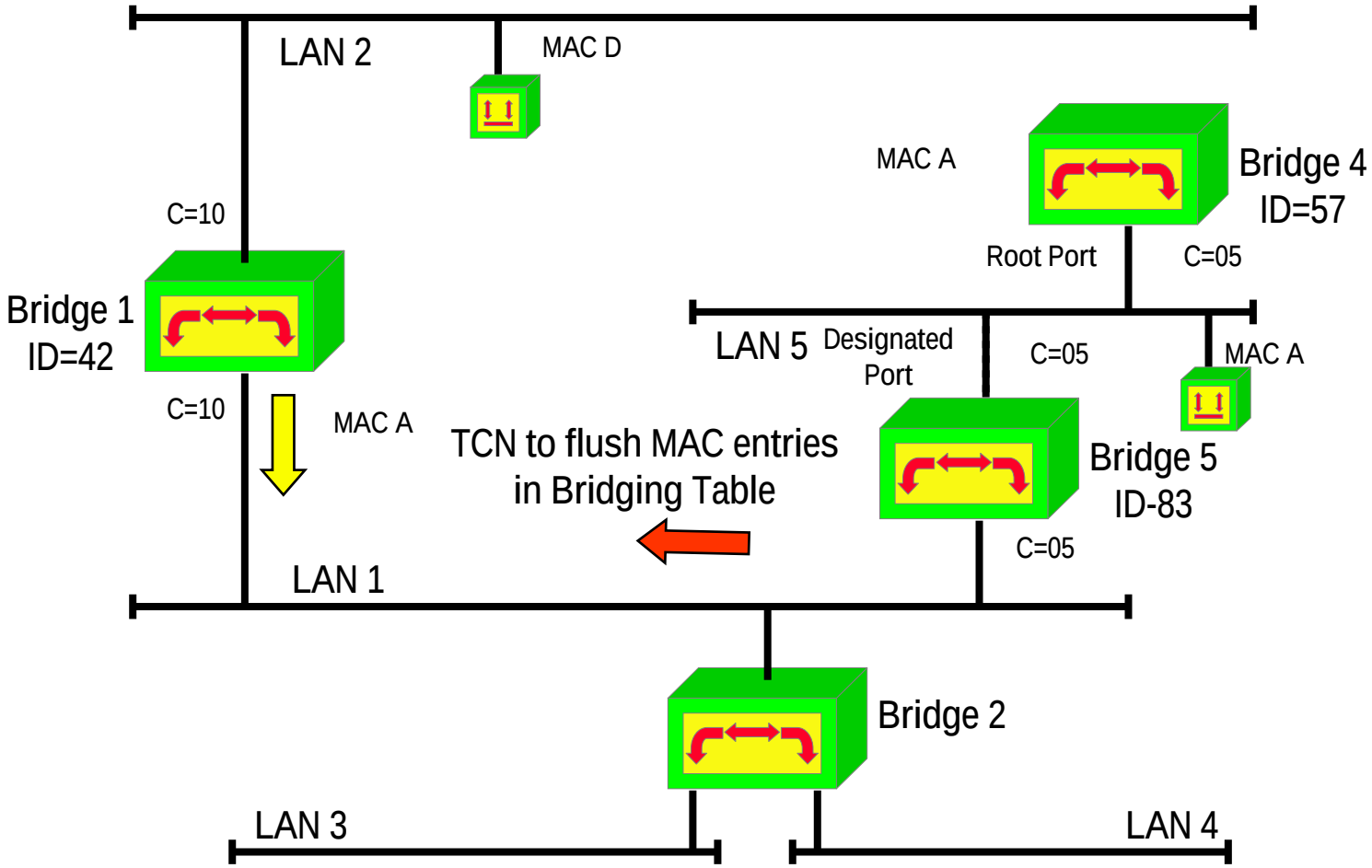
- **Time = 2*forward delay (15 sec Listening + 15 sec Learning) = 30 sec**

STP Convergence Time – Failure of Root Port - Interruption of Connectivity D->A



- **Time = 2*forward delay (15 sec Listening + 15 sec Learning) = 30 sec**

STP Convergence Time – Failure of Root Port – Topology Change Notification (TCN)



- **Time = 2*forward delay (15 sec Listening + 15 sec Learning) = 30 sec**

TCN Flags

- Flags (a "1" indicates the function):
  - **bit 8 ... Topology Change Acknowledgement (TCA)**
  - **bit 1 ... Topology Change (TC)**
  - **used in TCN BPDU's for signalling topology changes**
    - **TCN … Topology Change Notification**
    - **in case of a topology change the MAC addresses should change quickly to another port of the corresponding bridging table (convergence) in order to avoid forwarding of frames to the wrong port/direction and not waiting for the natural timeout of the dynamic entry**
    - **the bridge recognizing the topology change sends a TCN BPDU on the root port as long as a CONF BPDU with TCA is received on its root port**
    - **bridge one hop closer to the root passes TCN BPDU on towards the root bridge and acknowledges locally to the initiating bridge by usage of CONF BPDU with TCA**
    - **when the root bridge is reached a flushing of all bridging table is triggered by the root bridge by usage of CONF BPDUs with TC and TCA set**
    - **the new location (port) is dynamically relearned by the actual user traffic**

Configuration on Cisco switches

```
Switch(config)# spanning-tree vlan 200
```

Enable SPT on a specific VLAN

```
Switch(config)# spanning-tree vlan 200 priority 0
```

Enforcing Root Bridge

```
Switch(config-if)# spanning-tree cost 18
```

Manipulate Port Costs

```
Switch(config-if)# spanning-tree vlan 200 cost 15
```

Manipulate Port Costs for a specific VLAN

```
Switch# show spanning-tree vlan 200

VLAN0200
  Spanning tree enabled protocol ieee
  Root ID     Priority    49352
              Address     0008.2199.2bc0
              This bridge is the root
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID   Priority    49352  (priority 49152 sys-id-ext 200)
              Address     0008.2199.2bc0
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
              Aging Time 300
  Uplinkfast enabled

Interface        Port ID                    Designated                Port ID
Name             Prio.Nbr      Cost Sts     Cost Bridge ID            Prio.Nbr
---------------- -------- --------- --- --------- ------------------- --------
Fa0/1            128.1         3019 LIS         0 49352 0008.2199.2bc0 128.1
Fa0/2            128.2         3019 LIS         0 49352 0008.2199.2bc0 128.2
```

STP Optimizations

Port Fast
Uplink Fast
Backbone Fast

Port Fast

- Optimizes switch ports connected to end-station devices
  - Usually, if PC boots, NIC establishes L2-link, and switch port goes from Disabled=>Blocking=>Listening=>Learning=>Forwarding state ...30 seconds!!!
- Port Fast allows a port to immediately enter the Forwarding state
  - STP is NOT disabled on that port!

Port Fast

- Port Fast only works once after link comes up!
  - If port is then forced into Blocking state and later returns into Forwarding state, then the normal transition takes place!
  - Ignored on trunk ports
- Alternatives:
  - Disable STP (often a bad idea)
  - Use a hub in between => switch port is always active

PortFast Configuration

```
Switch(config-if)# spanning-tree portfast
```

Enables PortFast on an interface

```
Switch#show running-config interface fastethernet 5/8
Building configuration...
Current configuration:
!
interface FastEthernet5/8
 no ip address
 switchport
 switchport access vlan 200
 switchport mode access
 spanning-tree portfast
end
```

Verify PortFast

STP Optimizations

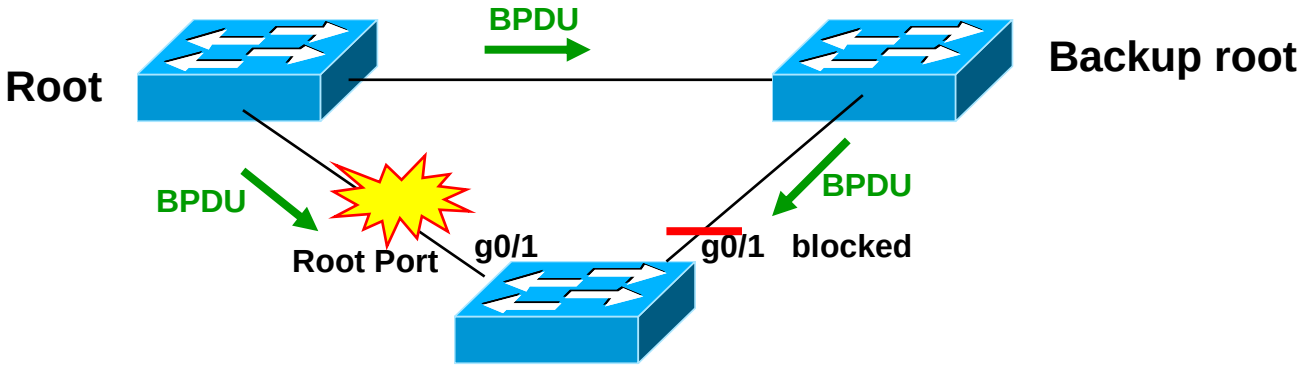
Port Fast
Uplink Fast
Backbone Fast

Uplink Fast

- Accelerates STP to converge within 1-3 seconds
  - Cisco patent
  - Marks some blocking ports as backup uplink
- Typically used on access layer switches
  - Only works on non-root bridges
  - Requires some blocked ports
  - Enabled for entire switch (and not for individual VLANs)

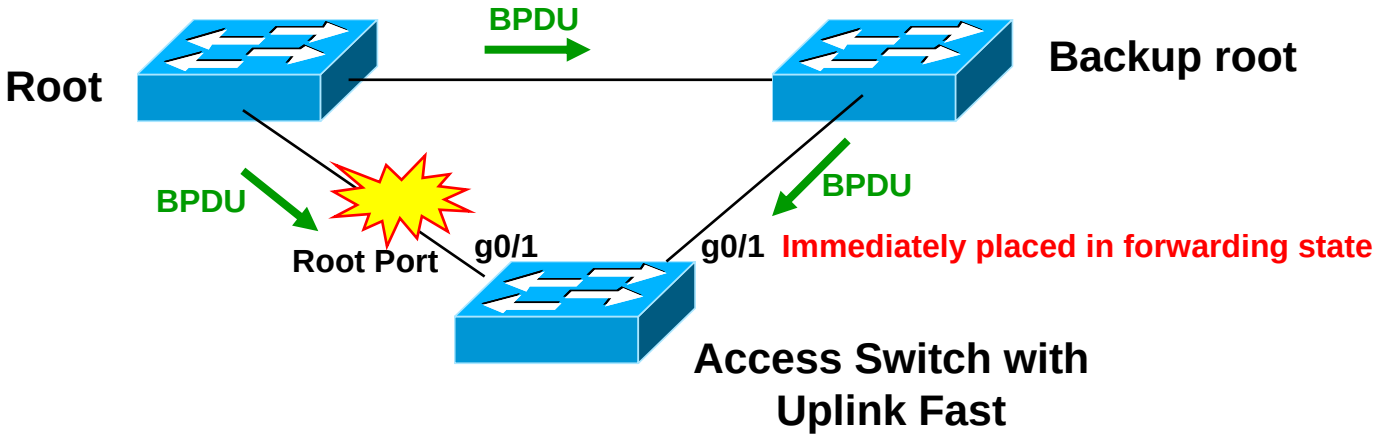
Problem

- When link to root bridge fails, STP requires (at least) 30 seconds until alternate root port becomes active



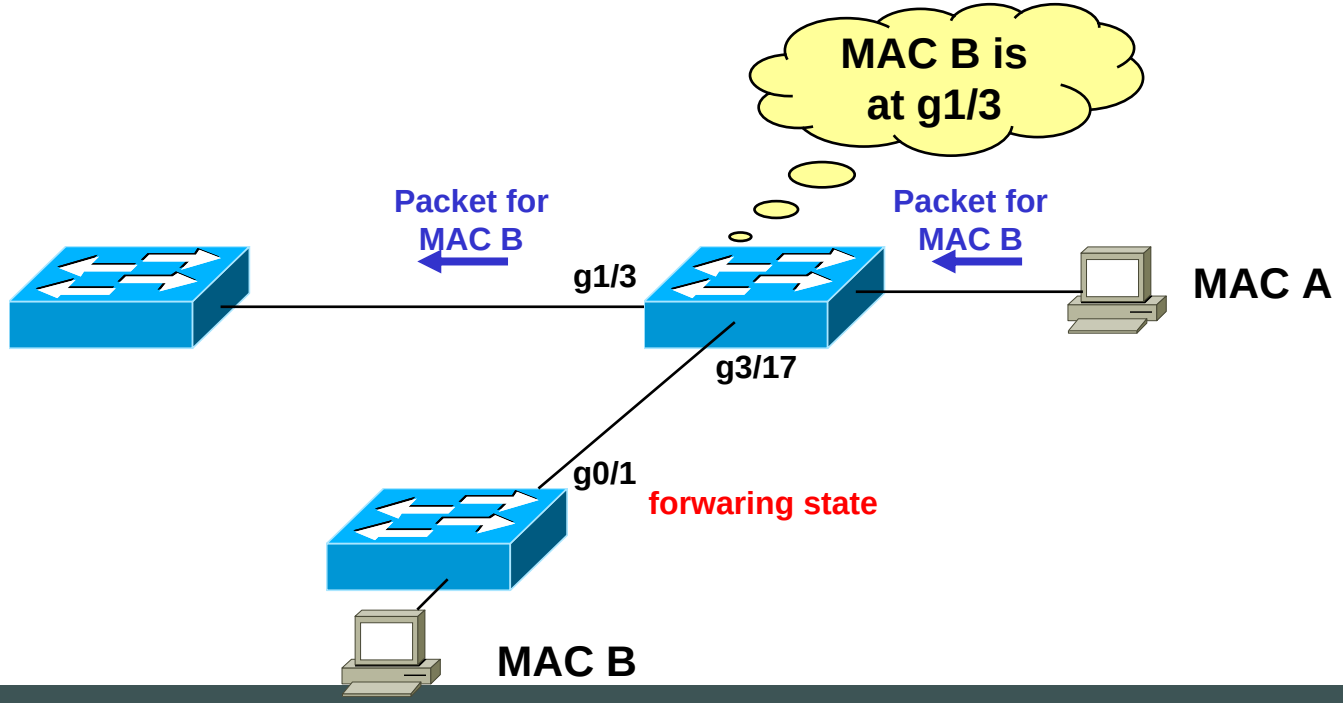
Idea of Uplink Fast

- When a port receives a BPDU, we know that it has a path to the root bridge
  - Put all root port candidates to a so-called "Uplink Group"
- Upon uplink failure, immediately put best port of Uplink group into forwarding state
  - There cannot be a loop because previous uplink is still down



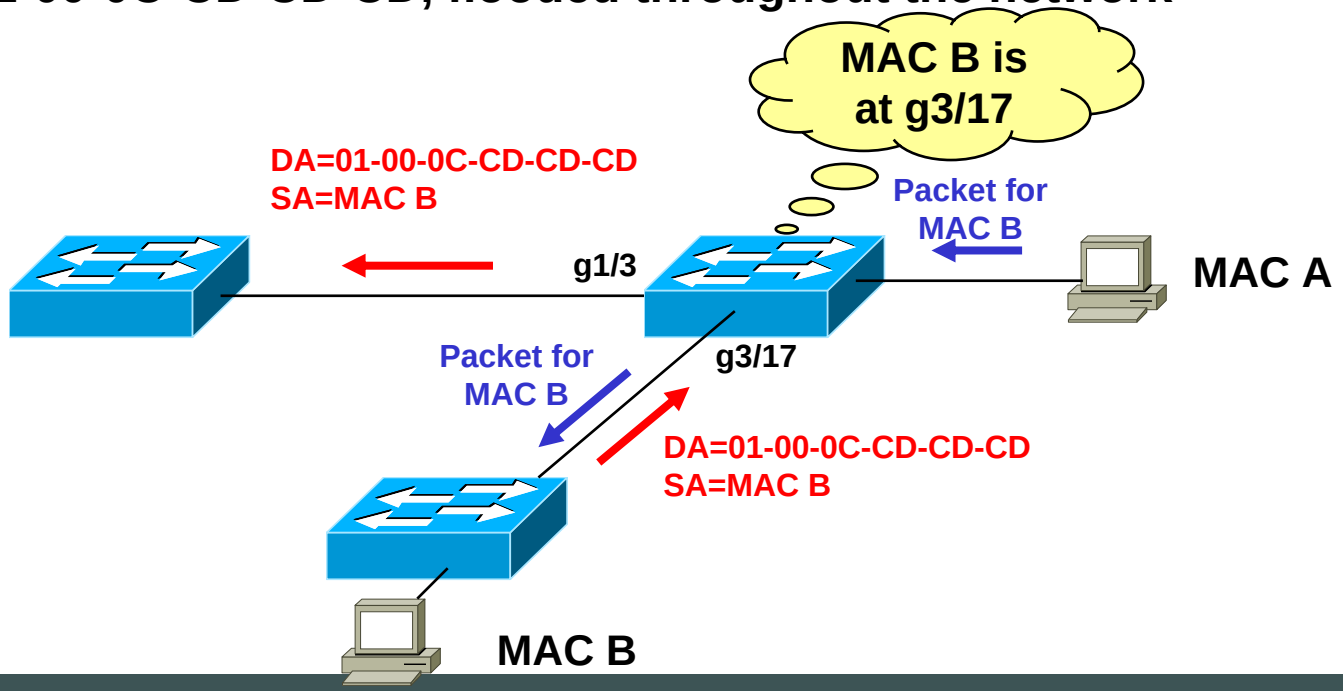
Incorrect Bridging Tables

- But upstream bridges still require 30 s to learn new topology
- Bridging table entries in upstream bridges may be incorrect



Actively correct tables

- Uplink Fast corrects the bridging tables of upstream bridges
- Sends 15 multicast frames (one every 100 ms) for each MAC address in its bridging table (i. e. for each downstream hosts)
  - Using SA=MAC: All other bridges quickly reconfigure their tables; dead links are no longer used
  - DA=01-00-0C-CD-CD-CD, flooded throughout the network



Addional Details

- When broken link becomes up again, Uplink Fast waits until traffic is seen
  - That is, 30 seconds plus 5 seconds to support other protocols to converge (e. g. Etherchannel, DTP, …)
- Flapping links would trigger uplink fast too often which causes too much additional traffic
  - Therefore the port is "hold down" for another 35 seconds before Uplink Fast mechanism is available for that port again
- Several STP parameters are modified automatically
  - Bridge Priority = 49152 (don't want to be root)
  - All Port Costs += 3000 (don't want to be designated port)

UplinkFast - Configuration

```
Switch(config)# spanning-tree uplinkfast [max-update-rate max_update_rate]
```

```
Switch# show spanning-tree uplinkfast
UplinkFast is enabled
Station update rate set to 150 packets/sec.
UplinkFast statistics
-----------------------
Number of transitions via uplinkFast (all VLANs)            :9
Number of proxy multicast addresses transmitted (all VLANs) :5308
Name                 Interface List
-------------------- ------------------------------------
VLAN1                Fa6/9(fwd), Gi5/7
VLAN2                Gi5/7(fwd)
VLAN3                Gi5/7(fwd)
VLAN4
VLAN5
VLAN1002             Gi5/7(fwd)
VLAN1003             Gi5/7(fwd)
VLAN1004             Gi5/7(fwd)
VLAN1005             Gi5/7(fwd)
```

STP Optimizations

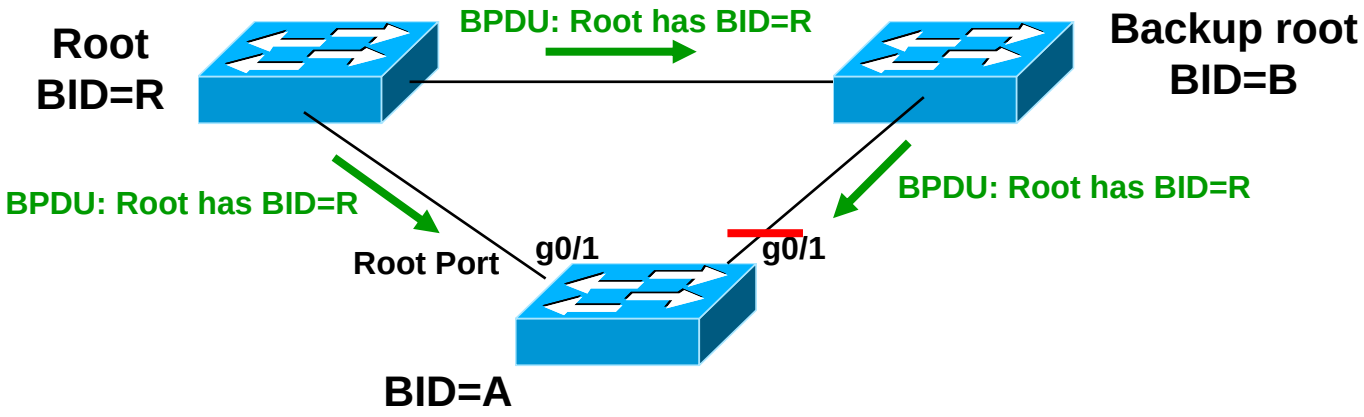
Port Fast
Uplink Fast
Backbone Fast

Backbone Fast

- Complementary to Uplink Fast
- Safes 20 seconds when recovering from indirect link failures in core area
  - Issues Max Age timer expiration
  - Reduce failover performance from 50 to 30 seconds
  - Cannot eliminate Forwarding Delay
- Should be enabled on every switch!

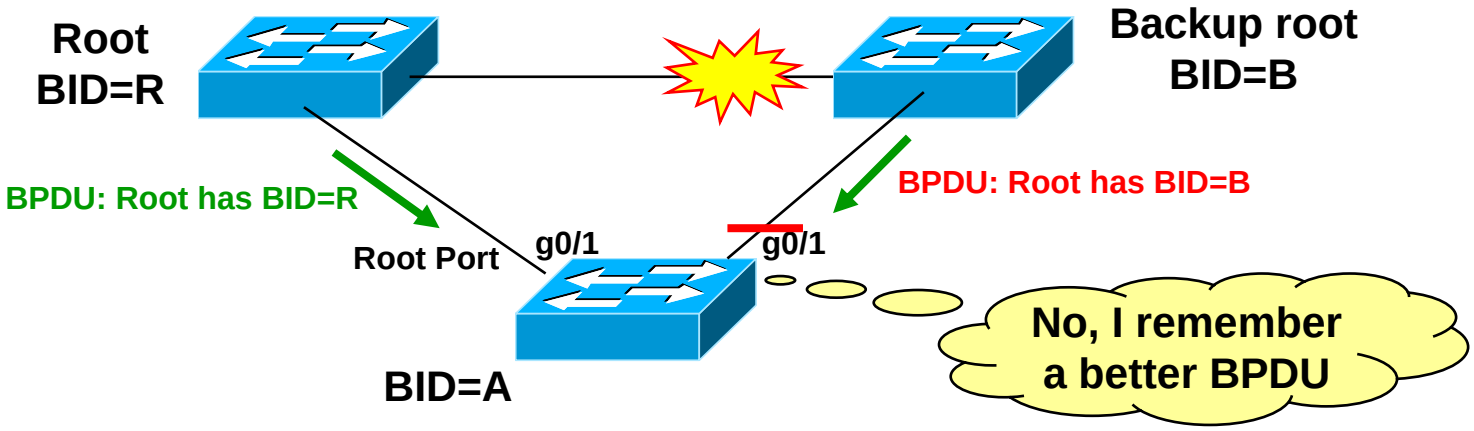
Problem

- Consider initial situation
- Note that blocked port (g0/1) always remembers "best seen" BPDU – which has best (=lowest) Root-BID



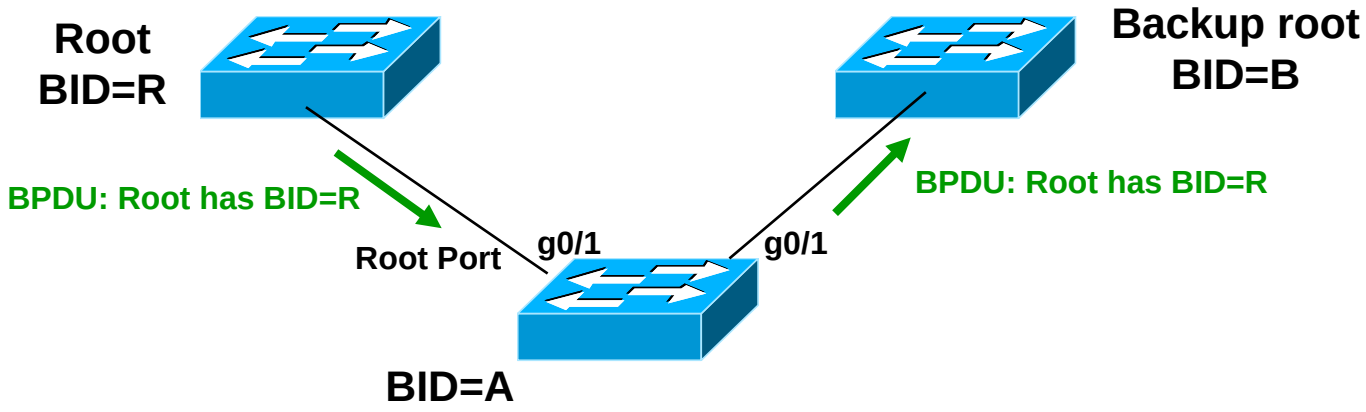
Problem (cont.)

- Now backup-root bridge looses connectivity to root bridge and assumes root role
- Port g0/1 does not see the BPDUs from the original root bridge any more
- But for MaxAge=20 seconds, any inferior BPDU is ignored



Problem (cont.)

- Only after 20 seconds port g0/1 enters listening state again
- Finally, bridge A unblocks g0/1 and forwards the better BPDUs to bridge B
- Total process lasts 20+15+15 seconds



Solution

- If an inferior BPDU is originated from the local segment's Designated Bridge, then this probably indicates an indirect failure
  - (Bridge B was Designated Bridge in our example)
- To be sure, we ask other Designated Bridges (over our ***other*** blocked ports and the root port) what they think which bridge the root is
  - Using Root Link Query (RLQ) BPDU
- If at least one reply contains the "old" root bridge, we know that an indirect link failure occurred
  - Immediately expire Max Age timer and enter Listening state

BackboneFast - Configuration

```
Switch(config)# spanning-tree backbonefast
```

```
Switch# show spanning-tree backbonefast
BackboneFast is enabled

BackboneFast statistics
-----------------------
Number of transition via backboneFast (all VLANs) : 0
Number of inferior BPDUs received (all VLANs)     : 0
Number of RLQ request PDUs received (all VLANs)   : 0
Number of RLQ response PDUs received (all VLANs)  : 0
Number of RLQ request PDUs sent (all VLANs)       : 0
Number of RLQ response PDUs sent (all VLANs)      : 0
```

Other STP Tuning Options

- BPDU Guard
  - Shuts down PortFast-configured interfaces that receive BPDUs, preventing a potential bridging loop
- Root Guard
  - Forces an interface to become a designated port to prevent surrounding switches from becoming the root switch
- BPDU Filter
- BPDU Skew Detection
  - Report late BPDUs via Syslog
  - Indicate STP stability issues, usually due to CPU problems
- Unidirectional Link Detection (UDLD)
  - Detects and shuts down unidirectional links
- Loop Guard

Rapid Spanning Tree (RSTP)

IEEE 802.1D – 2004

(Formerly known as 802.1w)

Introduction

- RSTP is now an add-on to the IEEE 802.1D-2004 standard
  - Contains contributions from Cisco
- Computation of the Spanning Tree is identical between STP and RSTP
  - Conf-BPDU and TCN-BPDU still remain
  - New BPDU type "RSTP" has been added
    - Version=2, type=2
- RSTP BPDUs can be used to negotiate port roles on a particular link
  - Only done if neighbor bridge supports RSTP (otherwise only Conf-BPDUs are sent
  - Using a Proposal/Agreement handshake

Major Features

- BPDUs are no longer triggered by root bridge
  - Instead, each bridge can generate BPDUs independently and immediately (on-demand)
- Much faster convergence
  - Few seconds
- Better scalability
  - No network diameter limit

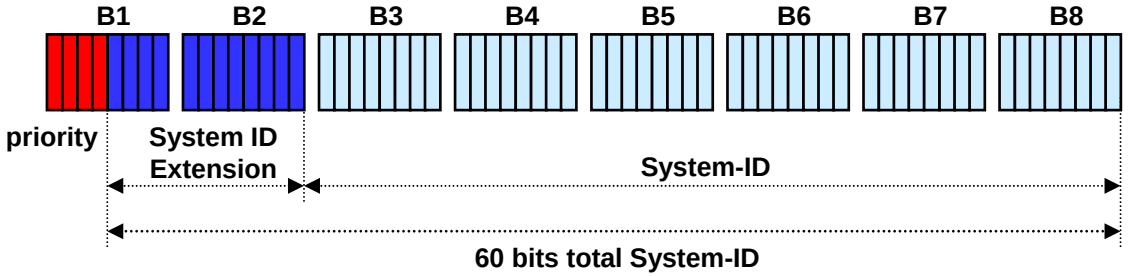
Compatibility

- RSTP is designed to be compatible and interoperable with the traditional STP – without additional management requirements!
- If an RSTP-enabled bridge is connected to an STP bridge, only Configuration-BPDUs and Topology-Change BPDUs are sent
  - (No port role negotiation)
- Memory requirements per bridge port independent of number of bridges

Basic Parameters

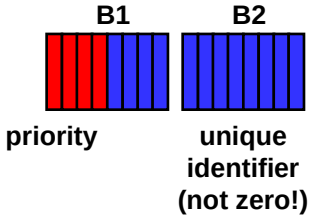
Bridge-ID

(the lesser the better)



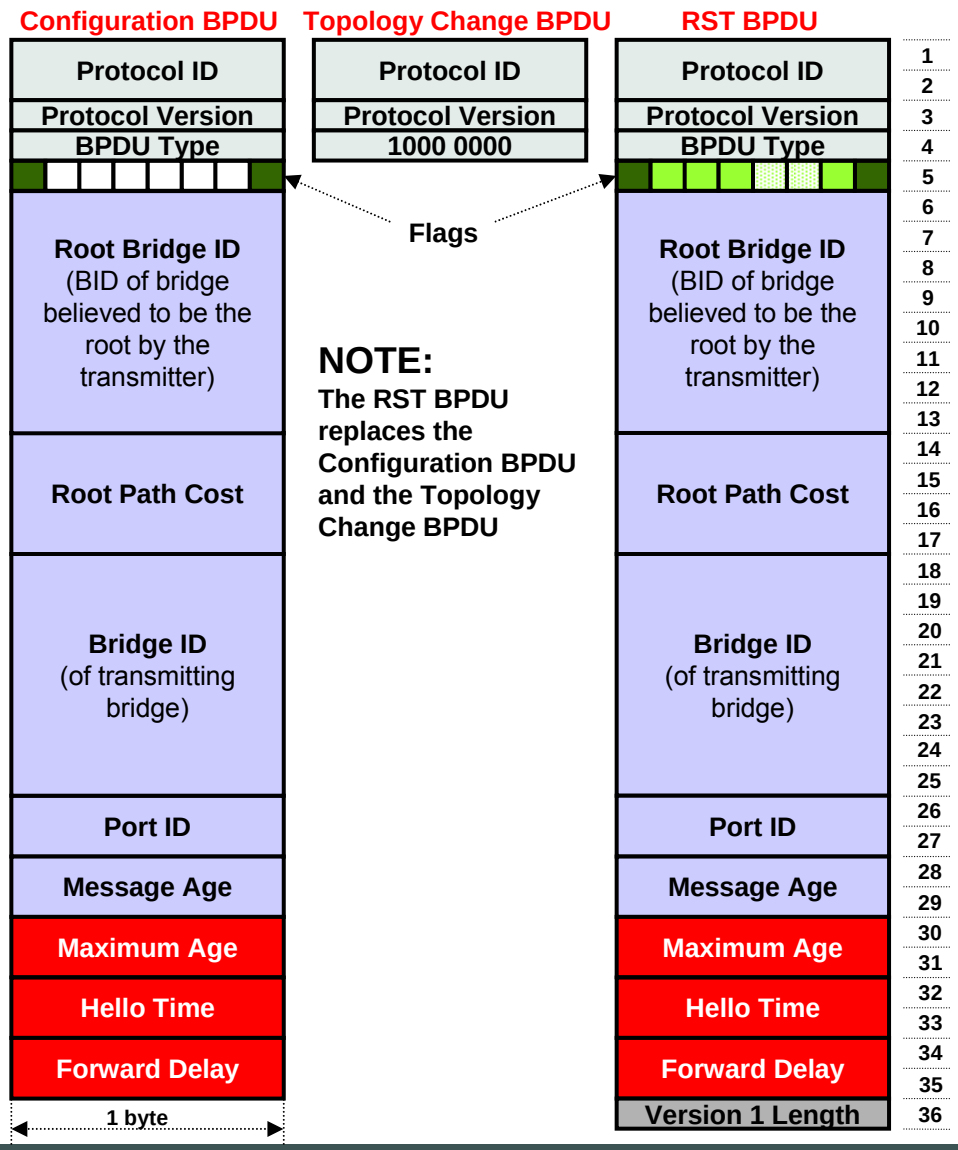
Port-ID

(the lesser the better)



Unit time value: **1/256 s**

BPDU Types (Old and New)



NOTE:
The RST BPDU replaces the Configuration BPDU and the Topology Change BPDU

all set to zero means RSTP but also STP!

RSTP BPDU: 0000 0010
RSTP BPDU: 0000 0010

Port Role:
0 0 = Unknown
0 1 = Alternate or Backup
1 0 = Root
1 1 = Designated

of the Port through which the message was transmitted

must be less than Max Age

20 seconds

2 seconds

15 seconds

0000 0000 indicates that there is no Version 1 protocol information present

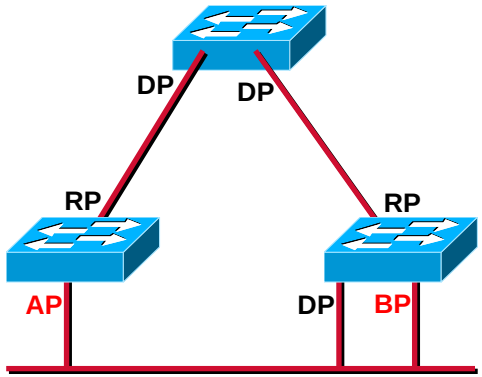
Same simple basic rules

- Bridge with lowest BID becomes Root Bridge
  - Has only Designated Ports
- Every other bridge has exactly one Root Port
  - Providing a least cost path to the Root Bridge
  - Local tie-breaker is the Port Identifier
- A Designated Bridge provides the lowest Root Path Cost for a LAN
  - Tie-breaker between multiple bridges is BID
  - Local tie-breaker is the Port Identifier

Backup and Alternate Ports

- If a port is neither Root Port nor Designated Port
  - It is a Backup Port – if this bridge is a Designated Bridge for that LAN
  - Or an Alternate Port otherwise

Backup and Alternate Ports:



Port Types

- **Shared Ports (Half Duplex !!!)**
  - Are not supported (ambiguous negotiations)
  - Uses standard STP here
- **Point-to-point ports (Full Duplex !!!)**
  - Usual and required port types
  - Supports proposal-agreement process
- **Edge Port**
  - Hosts resides here
  - Transitions directly to the Forwarding Port State, since there is no possibility of it participating in a loop
  - May change their role as soon as a BPDU is seen

Algorithm Overview

- Designated Ports transmit Configuration BPDUs periodically to detect and repair failures
  - Blocking (aka Discarding) ports send Conf-BPDUs only upon topology change
- Every Bridge accepts "better" BPDUs from any Bridge on a LAN or revised information from the prior Designated Bridge for that LAN
- To ensure that old information does not endlessly circulate through redundant paths in the network and prevent propagation of new information, each Configuration Message includes a message age and a maximum age
- Transitions to Forwarding is now confirmed by downstream bridge – therefore no Forward-Delay necessary!

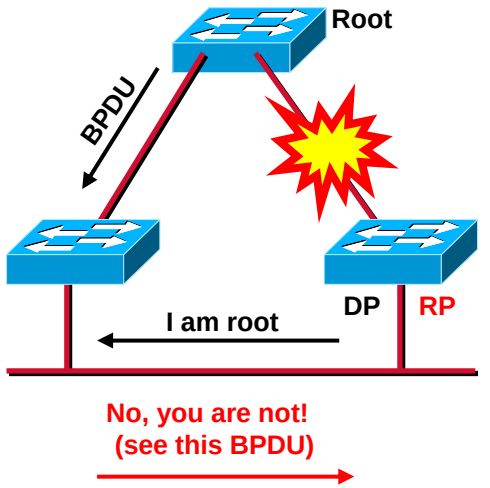
Main Differences to STP (1)

- The three 802.1d states *disabled*, *blocking*, and *listening* have been merged into a unique 802.1w discarding state
- Non-designated ports on a LAN segment are split into *alternate* ports and *backup* ports
  - A backup port receives better BPDUs from the same switch
  - An alternate port receives better BPDUs from another switch

Main Differences to STP (2)

- BPDUs are sent every hello-time, and not simply relayed anymore
  - Immediate aging if three consecutive BPDUs are missing
- When a bridge receives better information ("I am root") from its DB, it immediately accepts it and replaces the one previously stored
  - But if the RB is still alive, this bridge will notify the other via BPDUs

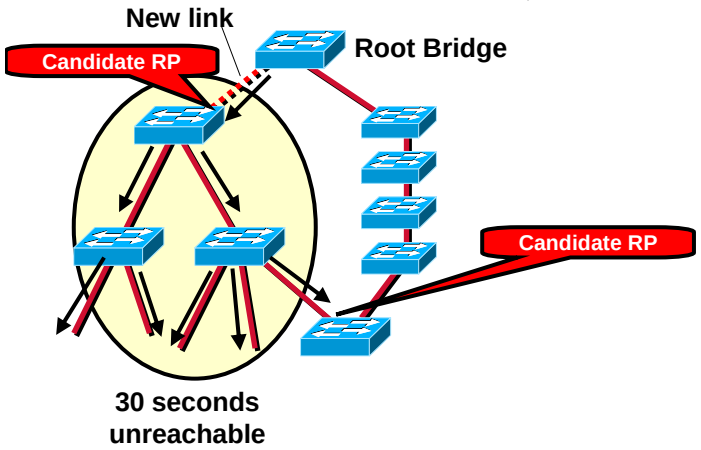
BackboneFast-like behavior:



Rapid Transition Details

Basic Principle

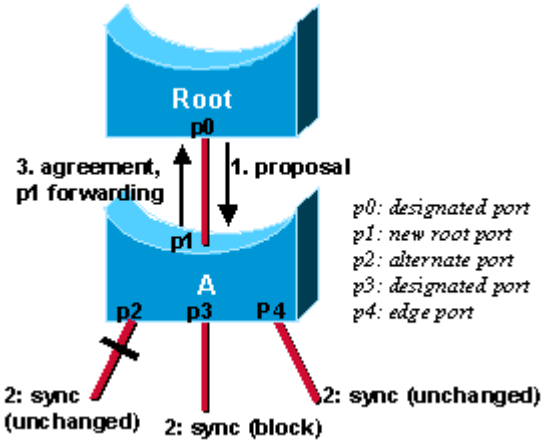
- The new rapid STP is able to actively confirm that a port can safely transition to forwarding without relying on any timer configuration
  - Feedback mechanism
- Edge Ports connect hosts
  - Cannot create bridging loops
  - Immediate transition to forwarding possible
  - No more Edge Port upon receiving BPDU
- Rapid transition only possible if Link Type is point-to-point
  - No half-duplex (=shared media)



Details

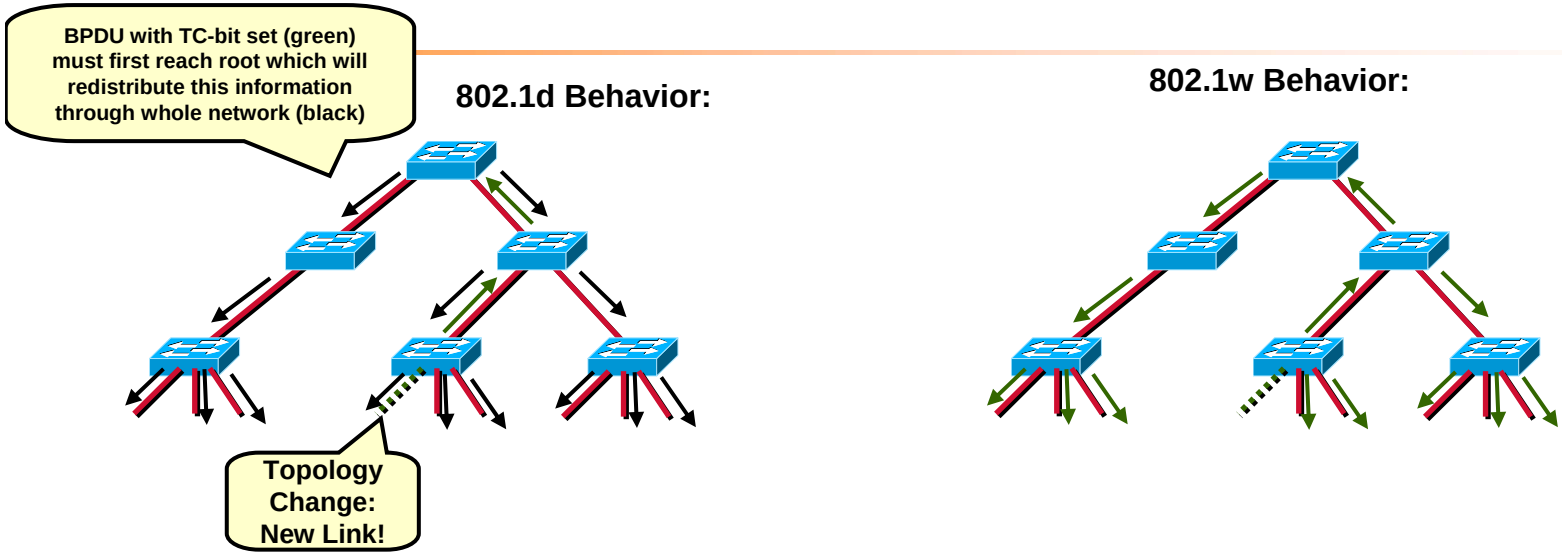
- Legacy STP:
  - Upon receiving a (better) BPDU on a blocked/previously-disabled port, 15+15 seconds transition time needed until forwarding state reached
  - But received BPDUs are propagated immediately downstream: some bridges below may detect a new Root Port candidate and also require 15+15 seconds transition time
  - Network inbetween is unreachable for 30 seconds!!!
- NEW: Sync Operation
  - Not the Root Port candidates are blocked, but the designated ports downstream—this avoids potential loops, too!
  - Bridge explicitly authorizes upstream bridge to put Designated Port in forwarding state (sync)
  - Then the sync-procedure propagates downstream

More Details



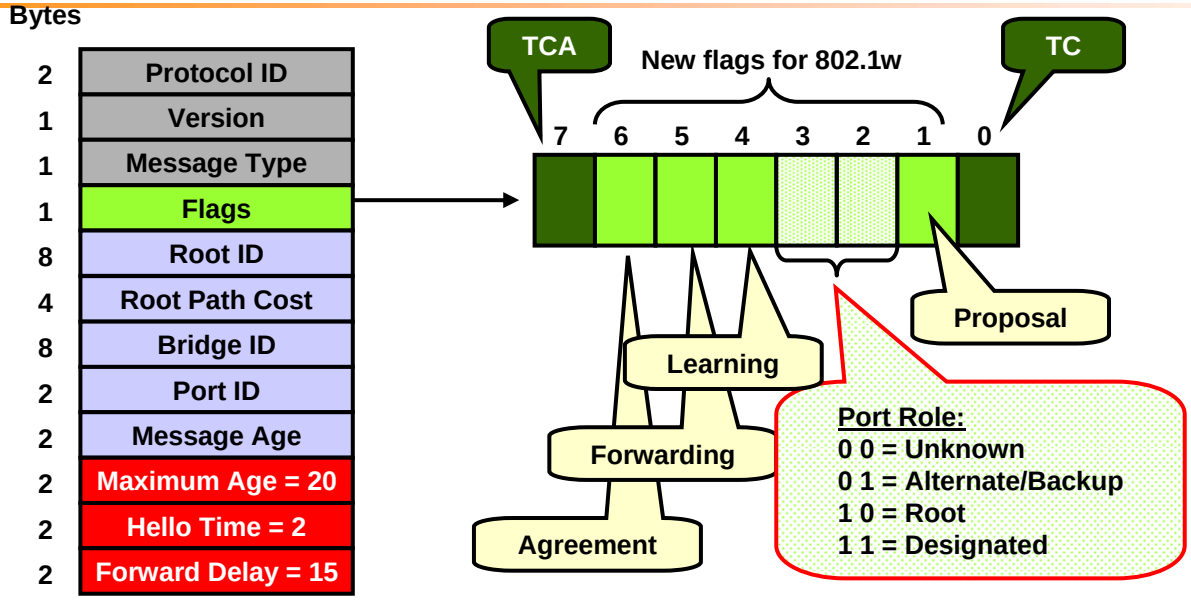
1) A new link is created between the root and Switch A.
2) Both ports on this link are put in a designated blocking state until they receive a BPDU from their counterpart.
3) Port p0 of the root bridge sets "proposal bit" in the BPDU (step 1)
4) Switch A then starts a sync to ensure that all of its ports are in-sync with this new information (only blocking and edge-ports are currently in-sync). Switch A just needs to block port p3, assigning it the discarding state (step 2).
5) Switch A can now unblock its newly selected root port p1 and reply to the root by sending an agreement message (Step 3, same BPDU with agreement bit set)
6) Once p0 receives that agreement, it can immediately transition to forwarding.
7) Now port 3 will send a proposal downwards, and the same procedure repeats.

Topology Change

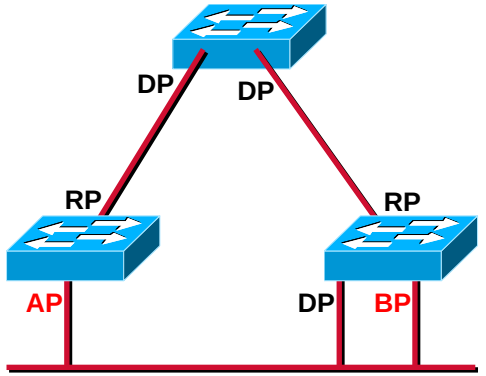


- 802.1d: When a bridge detects a topology change
  - A TCN is sent to towards the root
  - Root sends Conf-BPDU with TC-bit downstream (for 10 BPDUs)
  - All other bridges can receive it and will reduce their bridging-table aging time to *forward_delay* seconds, ensuring a relatively quick flushing of stale information
- RSTP: Only non-edge ports moving to the forwarding state cause a TCN
  - Loss of connectivity NOT regarded as topology change any more
  - TCN is immediately flooded throughout whole domain
  - Every bridge flushes MAC addresses and sends TCN upstream (RP) and downstream (DPs)
  - Other bridges do the same: Now, the TCN-process is a one-step procedure, as the TCNs do not need to reach the root first and require the root for re-origination downstream

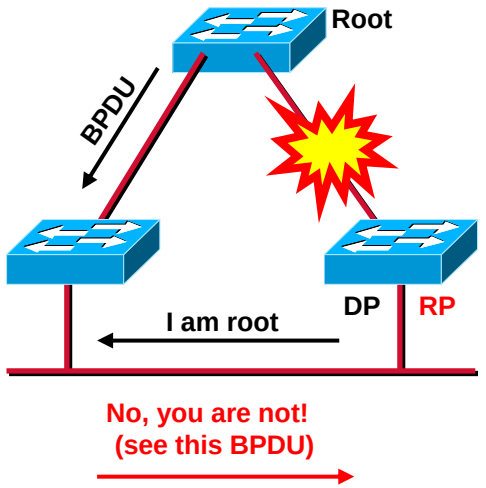
RSTP Summary



Backup and Alternate Ports:



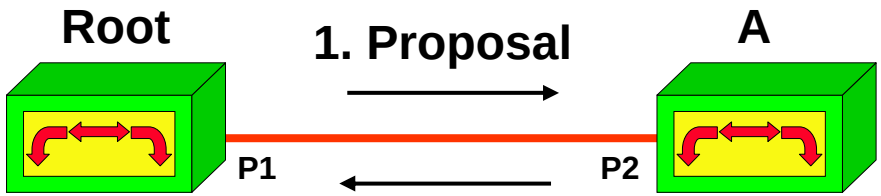
BackboneFast-like behavior:



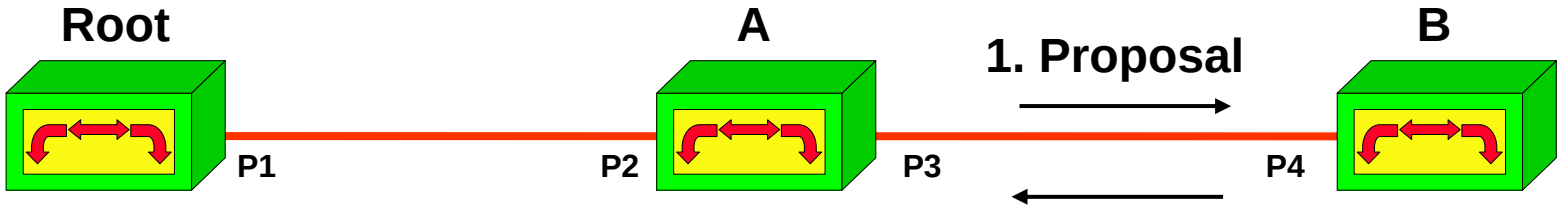
- IEEE 802.1w is an improvement of 802.1d
  - Vendor-independent (Cisco's Uplink Fast, Backbone Fast, and Port Fast are proprietary)
- The three 802.1d states *disabled*, *blocking*, and *listening* have been merged into a unique 802.1w discarding state
- Nondesignated ports on a LAN segment are split into *alternate* ports and *backup* ports
  - A backup port receives better BPDUs from the same switch
  - An alternate port receives better BPDUs from another switch
- Other changes:
  - BPDU are sent every hello-time, and not simply relayed anymore.
  - Immediate aging if three consecutive BPDUs are missing
  - When a bridge receives inferior information ("I am root") from its DB, it immediately accepts it and replaces the one previously stored. If the RB is still alive, this bridge will notify the other via BPDUs.

Proposal/Agreement Sequence

- **Suppose a new link is created between the root and switch A and a new switch B is inserted**



2. Agreement
P1 Designated Port -> Forwarding State
P2 Root Port



2. Agreement
P3 Designated Port
-> Forwarding State
P4 Root Port

Other

- There is no 15-sec forwarding delay anymore
  - TCN ensures that all tables are immediately flushed
- Protection against misordering and duplication
  - Port state transitions to Learning and Forwarding are delayed
  - Ports can temporarily transition to the Discarding state
- RSTP provides rapid recovery to minimize frame loss

Note

- A bridge must first receive a BPDU from the Root Bridge until BPDUs from Non-Root-Bridges can be forwarded
- Every bridge sends BPDUs periodically (by default every 2 seconds) and the neighbor bridge is declared dead when three subsequent BPDUs are missing
- Upon a topology change (e. g. neighbor dead) the bridge sends BPDUs with the Proposal Bit set which triggers a recalculation of the STP

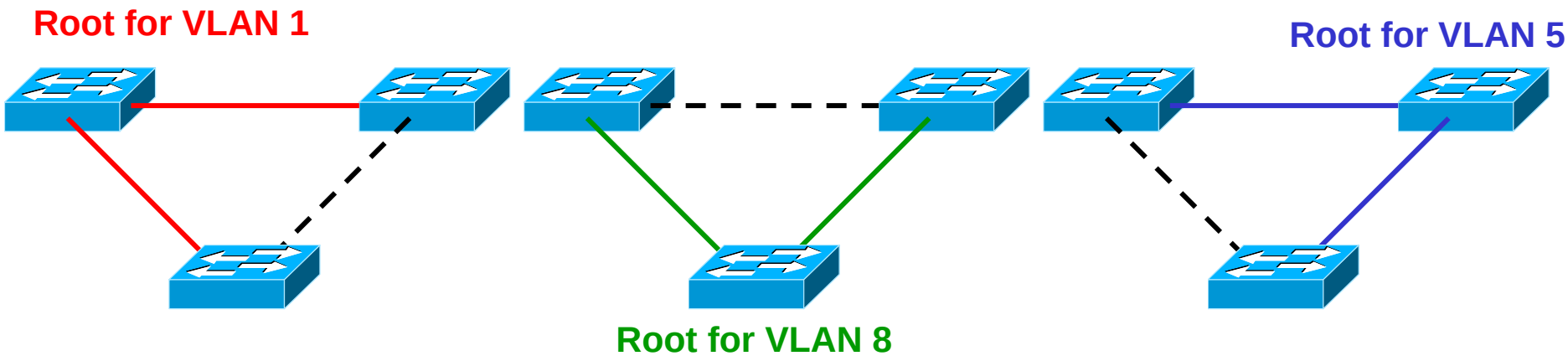
Cisco Extensions: PVST(+)

Per-VLAN Spanning Tree

About

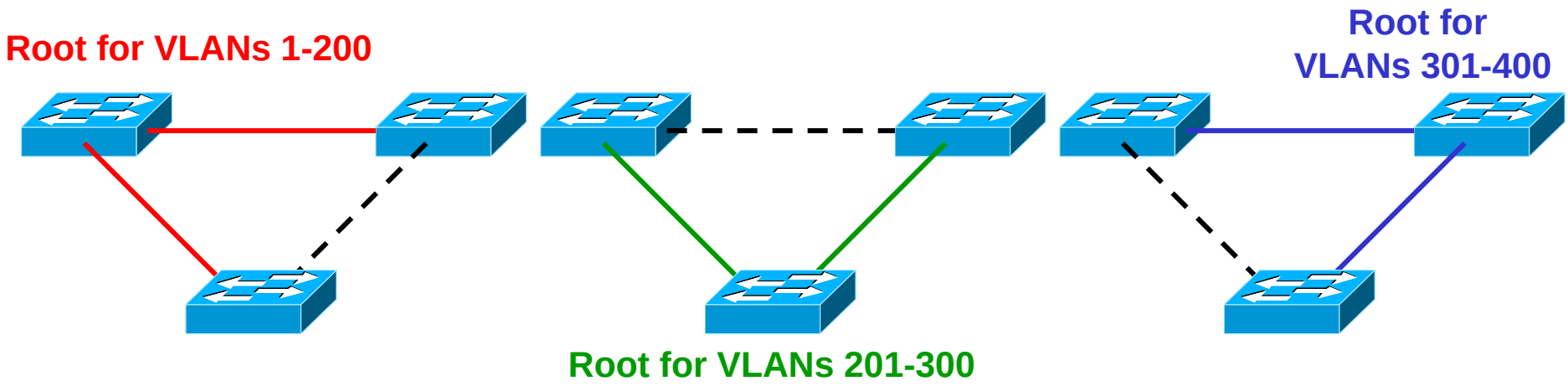
- In over 70% of all enterprise networks you will encounter Cisco switches
- Cisco extended STP and RSTP with a per-VLAN approach: "Per-VLAN Spanning Tree"
- Advantages:
  - Better (per-VLAN) topologies possible
  - STP-Attacks only affect current VLAN
- Disadvantages:
  - Interoperability problems might occur
  - Resource consumption (800 VLANs means 800 STP instances)

Example



- Remember that root bridge should realize the center of the LAN
  - Attracts all traffic
  - Typically servers or Internet-connectivty resides there
- Different VLANs might have different cores
- PVST+ allows for different topologies
  - Admin should at least configure ideal root bridge BID manually

Scalability Problem



- Typically the number of VLANs is much larger than the number of switches
- Results in many identical topologies
- In the above example we have 400 VLANs but only three different logical topologies
  - 400 Spanning Tree instances
  - 400 times more BPDUs running over the network

PVST (Classical, OLD!)

- Cisco proprietary (of course)
- Interoperability problems when also standard CST is used in the network (different trunking requirements)
- Provides dedicated STP for every VLAN
- Requires ISL
  - Inter Switch Link (Cisco's alternative to 802.1Q)

PVST+

- Today standard in Cisco switches
  - Default mode
  - Interoperable with CST
- The PVST BPDUs are also called SSTP BPDUs
- The messages are identical to the 802.1d BPDU but uses SNAP instead of LLC plus a special TLV at the end

PVST+ Protocol Details

- For native VLAN on trunk, normal (untagged) 802.1d BPDUs are sent
  - Also to the IEEE destination address 0180.c200.0000
- For tagged VLANs, PVST+ BPDUs use
  - SNAP, OID=00:00:0C, and EtherType 0x010B
  - Destination address 01-00-0c-cc-cc-cd
  - Plus 802.1Q tag
- Additionally a "PVID" TLV field is added at the end of the frame
  - This PVID TLV identifies the VLAN ID of the source port
  - The TLV has the format:
    - type (2 bytes) = 0x00 0x34
    - length (2 bytes) = 0x00 0x02
    - VLAN ID (2 bytes)
    - Also usually some padding is appended

PVST+ Compatibility Issues

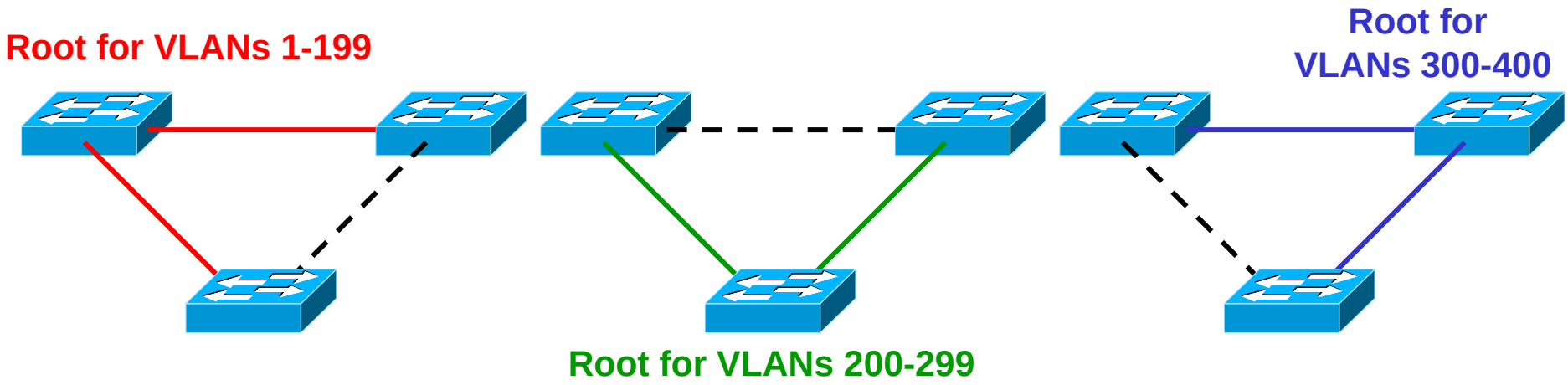
- PVST+ switches can act as translators between groups of Cisco PVST switches (using ISL) and groups of CST switches
  - Sent untagged over the native 802.1Q VLAN)
  - BPDUs of PVST-based VLANs are practically 'tunneled' over the CST-based switches using a special multicast address (the CST based switches will forward but not interpret these frames)
- Not important anymore…

MSTP

Overview

- Also the MSTP standard contains contributions from Cisco
- Solves the cardinality mismatch between the number of VLANs and the number of useful topologies
- Switches are organized in Regions
- In each Region sets of VLANs can be independently assigned to one out of 16 Spanning Tree Instances
- Each Instance has its own Spanning Tree topology

Example



- Compared to PVST+ only three Spanning Tree Topologies (=Instances) required
- Each STP instance has assigned 200 VLANs
  - Each VLAN can only be member of one instance of course

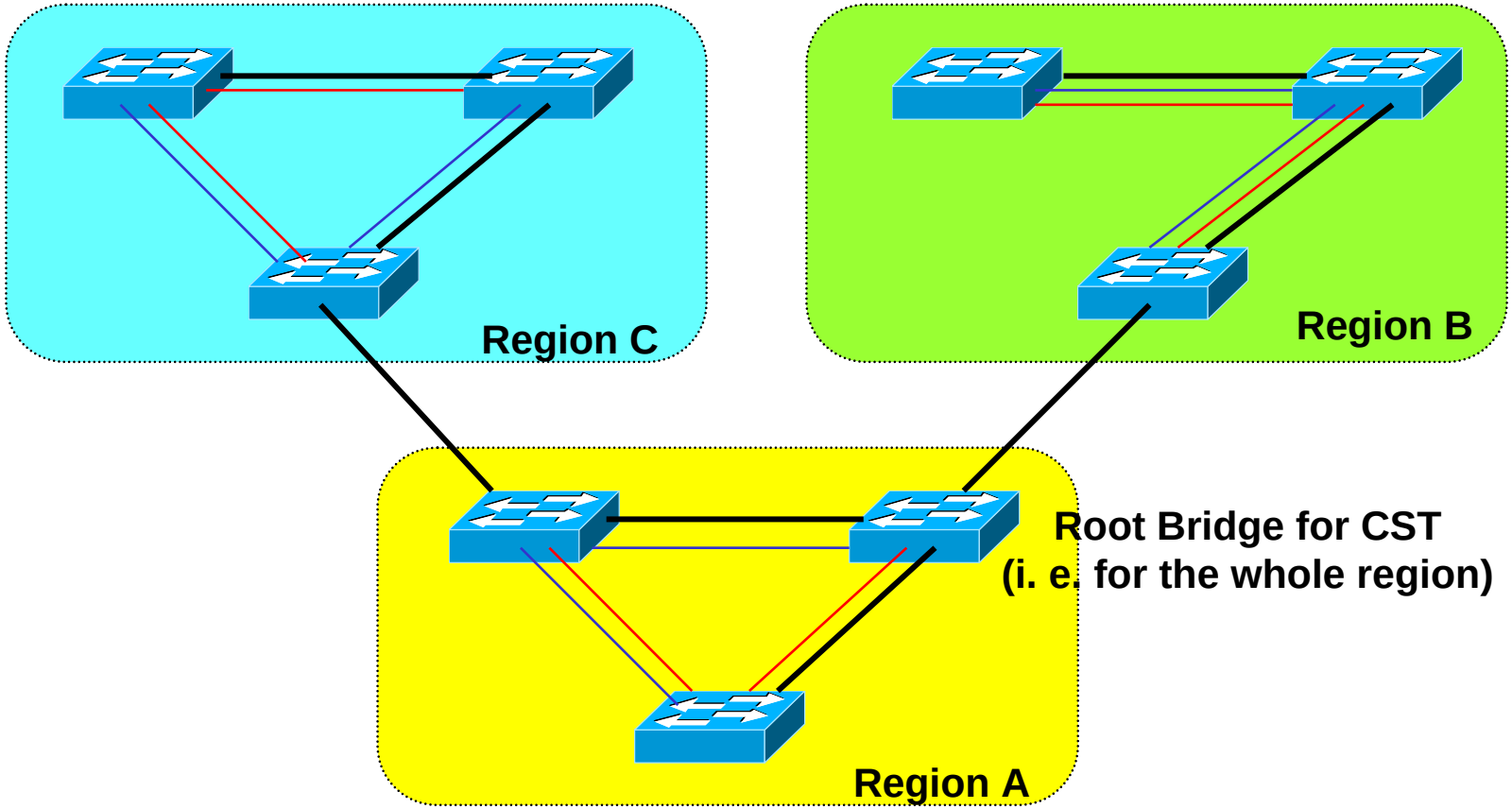
MSTP Details

- Each switch maintains its own MSTP configuration which contains the following mandatory attributes:
  - The Configuration name (32 chars),
  - The revision number (0..65535),
  - The element table which specifies the VLAN to Instance mapping
- All switches in a Region must have the same attributes

Regions

- The bridges checks attribute equivalence via a digest contained in the BPDUs
  - Note that the attributes must be configured manually and are NOT communicated via the BPDUs
- If digest does not match then we have a region boundary port
- Regions are only interconnected by the Common Spanning Tree (CST)
  - Instance 0
  - Uses traditional 802.1d STP

Region Example



- Only the logical STP topologies are shown (not the physical links)
- Each region has internal STP instances (red and blue)
- One CST instance interconnects all regions (black)

Note

- When enabling MSTP, per default the CST (instance zero) has all VLANs assigned
- Each region must be MSTP-aware
  - Since only a subset of VLANs is assigned to the CST
  - Old-STP switched always create a general (all-VLAN) topology
  - Don't let MSTP-unaware switch become root bridge

Any Questions?

THE ANSWER IS … FORTY-TWO!

The choice of 0x42 as the LLC SAP value for BPDUs has an interesting history. First, the chair and editor of the IEEE 802.1D Task Force (Mick Seaman) was British, and 42 is "The Answer to the Ultimate Question of Life, the Universe, and Everything" in *The Hitchhiker's Guide to the Galaxy*, a popular British book, radio, and television series [by Douglas Adams] at the time of the development of the original standard.

Even in the United States, the series was so popular that the original Digital Equipment Corp. bridge architecture specification was titled eXtended LAN Interface Interconnect, or XLII, the Roman representation of 42.

From Rich Seifert's Switch Book