Spanning-Tree Protocol

Spanning Tree Protocol (IEEE 802.1D 1998),
Rapid STP (IEEE 802.1D 2004), Cisco PVST+, MSTP

Agenda

- **Spanning Tree Protocol (STP)**
  - Introduction
  - Details
  - Convergence
  - Some more details
- **Rapid Spanning Tree Protocol (RSTP)**
- **Cisco PVST, PVST+**
- **Multiple Spanning Tree Protocol (MSTP)**

Problem Description

- **We want redundant links in bridged networks**
- **But transparent bridging cannot deal with redundancy**
  - Broadcast storms and other problems
- **Solution: STP (Spanning Tree Protocol)**
  - Allows for redundant paths
  - Ensures non-redundant active paths
- **Invented by *Radia Perlman* as general "mesh-to-tree" algorithm**
- **Only one purpose:**
  cut off redundant paths with highest costs

L07 - Spanning-Tree Details (v5.0)

Algorhyme

I think that I shall never see
a graph more lovely than a tree
a graph whose crucial property
is loop-free connectivity.
A tree which must be sure to span
so packets can reach every lan.
first the root must be selected
by ID it is elected.
least cost paths to root are traced,
and in the tree these paths are place.
mesh is made by folks like me;
bridges find a spanning tree.

Radia Perlman

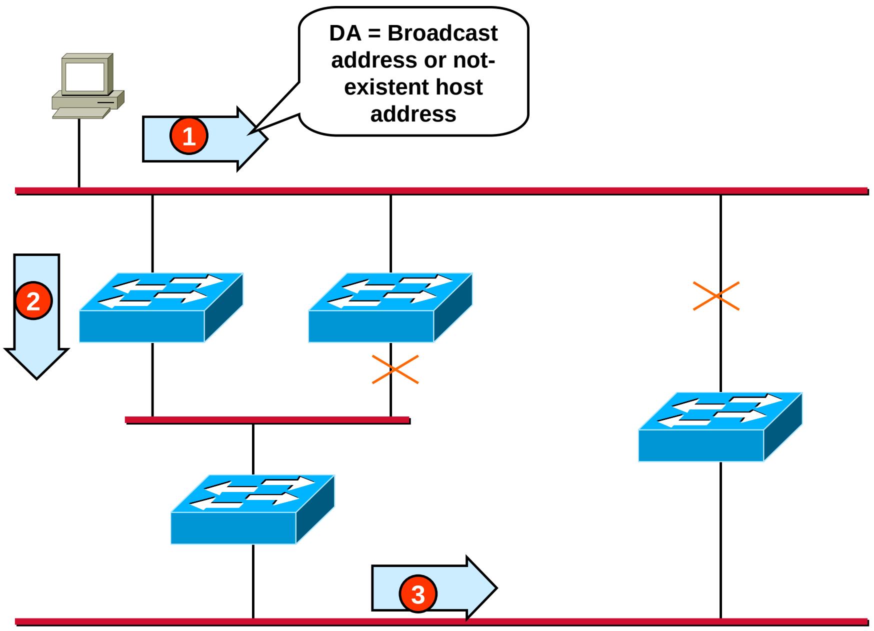
 Spanning-Tree Details, v5.0 4

Radia Perlman, PhD computer science 1988, MIT * MS math 1976, MIT * BA math 1973, MIT

Radia Perlman specializes in network and security protocols. She is the inventor of the spanning tree algorithm used by bridges, and the mechanisms that make modern link state protocols efficient and robust. She is the author of two textbooks, and has a PhD from MIT in computer science.

Her thesis on routing in the presence of malicious failures remains the most important work in routing security. She has made contributions in diverse areas such as, in network security, credentials download, strong password protocols, analysis and redesign of IPSec IKE protocols, PKI models, efficient certificate revocation, and distributed authorization. In routing, her contributions include making link state protocols robust and scalable, simplifying the IP multicast model, and routing with policies.

STP in Action (1)
No Broadcast Storm



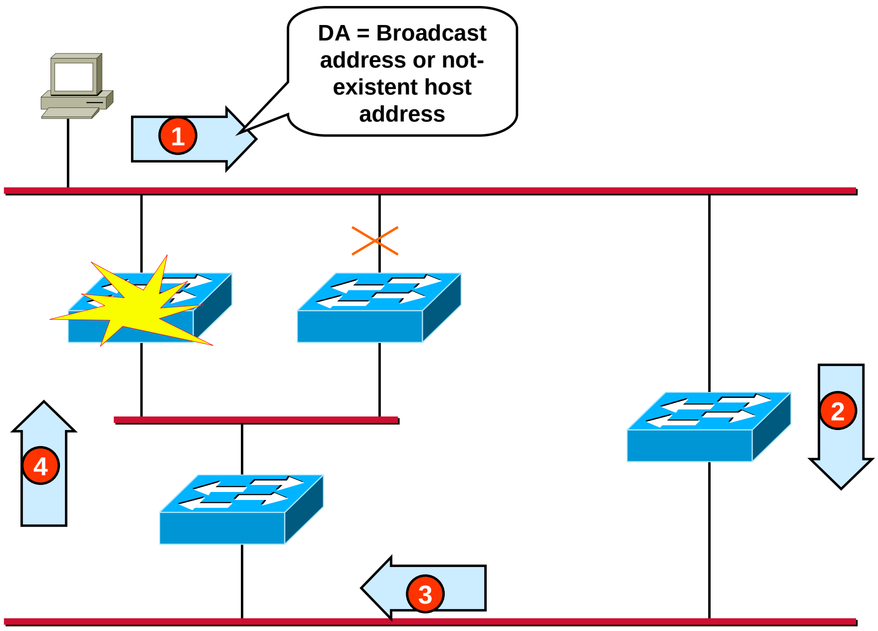
© 2012, D.I. Lindner / D.I. Haas Spanning-Tree Details, v5.0 5

STP eliminates redundancy in a LAN bridged environment by cutting of certain paths which are determined by the STP parameters Bridge ID, Bridge Priority and interface Port Costs. An easy way to achieve this is built a tree topology. A tree has per default no redundancy or have you ever seen leafs of a tree which are connected via two or more branches to the same tree?

Spanning Tree Protocol (STP) takes care that there is always exact only one active path between any 2 stations implemented by a special communication protocol between the bridges using BPDU (Bridge Protocol Data Unit) frames with MAC-multicast address. The failure of an active path causes activation of a new redundant path resulting in new tree topology.

L07 - Spanning-Tree Details (v5.0)

STP in Action (2)
Bridge Failure – New STP Topology



 Spanning-Tree Details, v5.0 6

Additional task of STP is to recognizes any failures of bridges and to automatically build a new STP topology allowing any-to-any communication again.

Here you can also see one main disadvantage of STP: Redundant lines or redundant network components cannot be used for load balancing. Redundant lines and components come only into action if something goes wrong with the current active tree.

Agenda

- **Spanning Tree Protocol (STP)**
  - Introduction
  - Details
  - Convergence
  - Some more details
- **Rapid Spanning Tree Protocol (RSTP)**
- **Cisco PVST, PVST+**
- **Multiple Spanning Tree Protocol (MSTP)**

Spanning Tree Protocol

- **Takes care that there is always exact only one active path between any 2 stations**
- **Implemented by a special communication protocol between the bridges**
  - Using BPDU (Bridge Protocol Data Unit) frames with MAC-multicast address as destination address
- **Three important STP parameters determine the resulting tree topology in a meshed network:**
  - Bridge-ID
  - Interface-Cost
  - Port-ID

 Spanning-Tree Details, v5.0 8

What do we need for STP to work? First of all this protocol needs a special messaging means, realized in so-called **Bridge Protocol Data Units (BPDUs).** BPDUs are simple messages contained in Ethernet frames containing several parameters described in the next pages.

Parameters for STP 1

- **Bridge Identifier (Bridge ID)**
  - Consists of a priority number and the MAC-address of a bridge
    - Bridge-ID = Priority# (2 Byte) + MAC# (6 Byte)
  - Priority number may be configured by the network administrator
    - Default value is 32768
  - Lowest Bridge ID has highest priority
  - If you keep default values
    - The bridge with the lowest MAC address will have the highest priority

 Spanning-Tree Details, v5.0 9

Each bridge is assigned one unique **Bridge-ID** which is a combination of a 16 bit priority number and the lowest MAC address found on any port on this bridge. The Bridge-ID is determined automatically using the default priority 32768. Note: Although bridge will not be seen by end systems, for bridge communication and management purposes a bridge will listen to one or more dedicated (BIA) MAC addresses. Typically, the lowest MAC-address is used for that. The Bridge-ID is used by STP algorithm to determine root bridge and as tie-breaker to when determine the designated port.

Parameters for STP 2

- **Port Cost (C)**
  - Costs in order to access local interface
  - Inverse proportional to the transmission rate
  - Default cost = 1000 / transmission rate in Mbit/s
    - With occurrence of 1Gbit/s Ethernet the rule was slightly adapted
    - May be configured to a different value by the network administrator
- **Port Identifier (Port ID)**
  - Consists of a priority number and the port number
    - Port-ID = port priority#.port#
    - Default value for port priority is 128
    - Port priority may be configured to a different value by the network administrator

Each port is assigned a **Port Cost**. Again this value is determined automatically using the simple formula Port Cost = 1000 / BW, where BW is the bandwidth in Mbit/s. Of course the Port Cost can be configured manually. Port Cost are used by STP algorithm to calculate Root Path Cost in order to determine the root port and the designated port

Each port is assigned a **Port Identifier.** Only used by STP algorithm as tie-breaker if the same Bridge-ID and the same Path Cost is received on multiple ports.

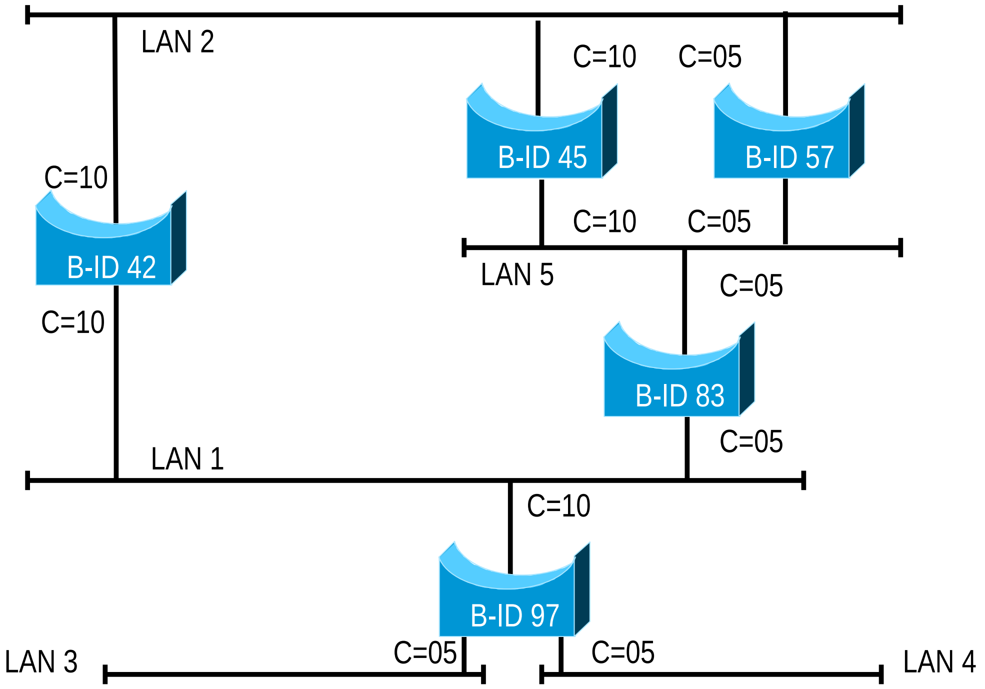
Comparison Table For Port Costs:

Speed [Mbit/s]	OriginalCost (1000/Speed)	802.1D-1998	802.1D-2004
10	100	100	2000000
100	10	19	200000
155	6	14	(129032 ?)
622	1	6	(32154 ?)
1000	1	4	20000
10000	1	2	2000

- **Also different cost values might be used**
  - See recommendations in the IEEE 802.1D-2004 standard to comply with RSTP and MSTP
  - 802.1D-2004 operates with 32-bit cost values instead of 16-bit

L07 - Spanning-Tree Details (v5.0)

STP Parameter Example (1)

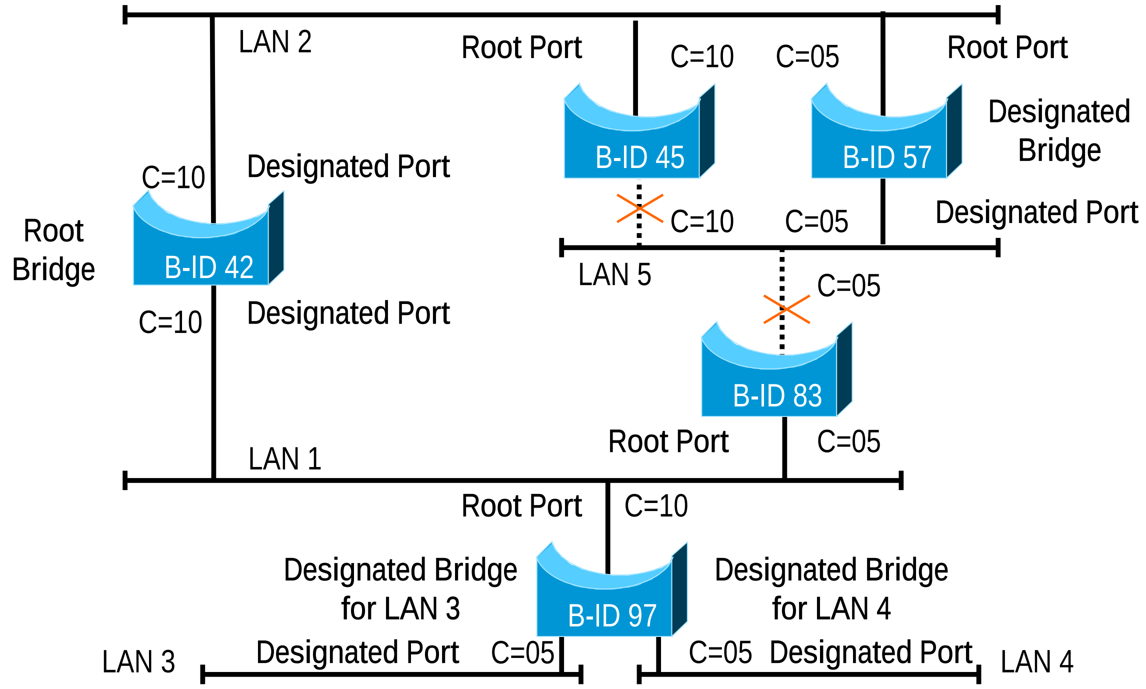


Spanning Tree Algorithm Summary

- **Select the root bridge**
  - Bridge with the lowest Bridge Identifier
- **Select the root ports**
  - By computation of the shortest path from any non-root bridge to the root bridge
  - Root port points to the shortest path towards the root
- **Select one designated bridge for every LAN segment which can be reached by more than one bridge**
  - Bridge with lowest root path costs on the root port side
  - Corresponding port on other side is called designated port
- **Set the designated and root ports in forwarding state**
- **Set all other ports in blocking state**

These creates single paths from the root to all leaves (LAN segments) of the network.

STP Parameter Example (2)



 Spanning-Tree Details, v5.0 14

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**

Prot. ID	Prot. Vers.	BPDU Type	Flags	Root ID (R-ID)	Root Path Costs (RPC)	Bridge ID (O-ID)	Port ID (P-ID)	Msg Age	Max Age	Hello Time	Fwd. Delay
2 Byte	1 Byte	1 Byte	1 Byte	8 Byte	4 Byte	8 Byte	2 Byte	2 Byte	2 Byte	2 Byte	2 Byte

Root ID: The Bridge I regard as root

Root Path Costs: The total cost I see toward the root

Bridge ID: My own ID

Just for your interest, the above picture shows the structure of BPDUs. You see, there is no magic in here, and the protocol is very simple. There are no complicated protocol procedures. BPDUs are sent periodically and contain all involved parameters. Each bridge enters its own "opinion" there or adds its root path costs to the appropriate field. Note that some parameters are transient and others are not.

The other parameters will be explained in the next slides.

BPDU Fields in Detail (1)

- Protocol Identifier:
  - **0000 (hex) for STP 802.1D**
- Protocol Version:
  - **00 (hex) for version 802.1D (1998)**
  - **02 (hex) for version 802.1D (2004) - RSTP**
- BPDU Type:
  - **00 (hex) for Configuration BPDU**
  - **80 (hex) for Topology Change Notification (TCN) BPDU**
- Root Identifier:
  - **2 bytes for priority (default 32768)**
  - **6 bytes for MAC-address**
- Root Path Costs in binary representation:
  - **range 1-65535**
- Bridge Identifier:
  - **Structure like Root Identifier**

BPDU Fields in Detail (2)

- Port Identifier:
  - **1 byte priority (default 128)**
  - **1 byte port number**
- Message Age (range 1-10s):
  - **Age of Configuration BPDU**
  - **Transmitted by root-bridge initially using zero value, each passing-on (by designated bridge) increases this number**
- Max Age (range 6-40s):
  - **Aging limit for information obtained from Configuration BPDU**
  - **Basic parameter for detecting idle failures (e.g. root bridge = dead)**
  - **Default 20 seconds**
- Hello Time (range 1-10s):
  - **Time interval for generation of periodic Configuration BPDUs by root bridge**
  - **Default 2 seconds**

BPDU Fields in Detail (3)

- Forward Delay (range 4-30s):
  - **Time delay for putting a port in the forwarding state**
  - **Default 15 seconds**
  - **That actually means:**
  - **15 seconds LISTENING for allowing STP topology to converge after a topology change**
  - **plus**
  - **15 seconds LEARNING to fill the empty MAC address table with locally seen MAC addresses in order to avoid flooding for any local MAC addresses**
  - **After that the ports are set to forwarding**
- Hello Time, Max Age, Forward Delay are specified by Root-Bridge
- Maximum Bridge Diameter
  - Maximum number of bridges between any two end systems is 7 using default values for hello time, forward delay and max age

BPDU Fields in Detail (4)

- Flags (a "1" indicates the function):
  - **Bit 8 ... Topology Change Acknowledgement (TCA)**
  - **Bit 1 ... Topology Change (TC)**
  - **Used in TCN BPDUs for signaling topology changes**
    - **TCN … Topology Change Notification**
    - **The bridge recognizing the topology change sends a TCN BPDU on the root port until 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**
    - **Now the new location (port) can be dynamically relearned by the actual user traffic**
    - **Note: 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**

BPDU MAC Addresses / LLC DSAP-SSAP

- **Bridges use for STP-communication:**
  - Multicast address:

    0180 C200 0000 hex

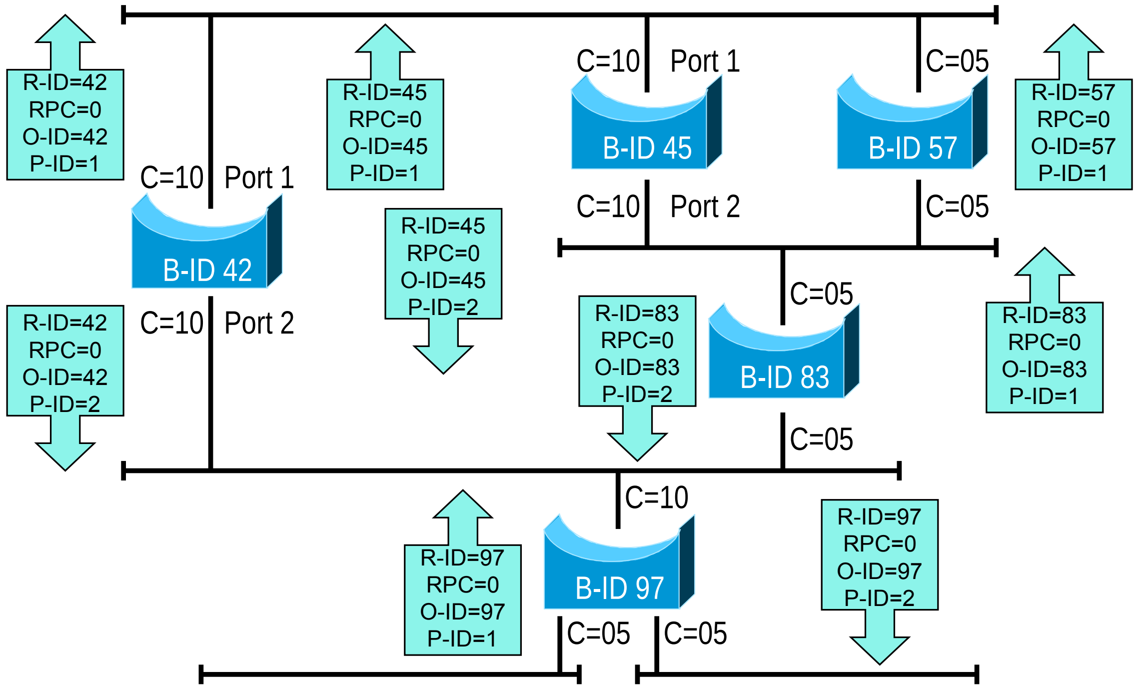
    0180 C200 0001 to 0180 C200 000F are reserved

    0180 C200 0010 hex All LAN Bridges Management Group Address
  - Note :
    - All addresses in Ethernet canonical format
  - The DSAP/SSAP of LLC header

    42 hex … Bridge Spanning Tree Protocol

L07 - Spanning-Tree Details (v5.0)

Selection of Root Bridge



 Spanning-Tree Details, v5.0 21

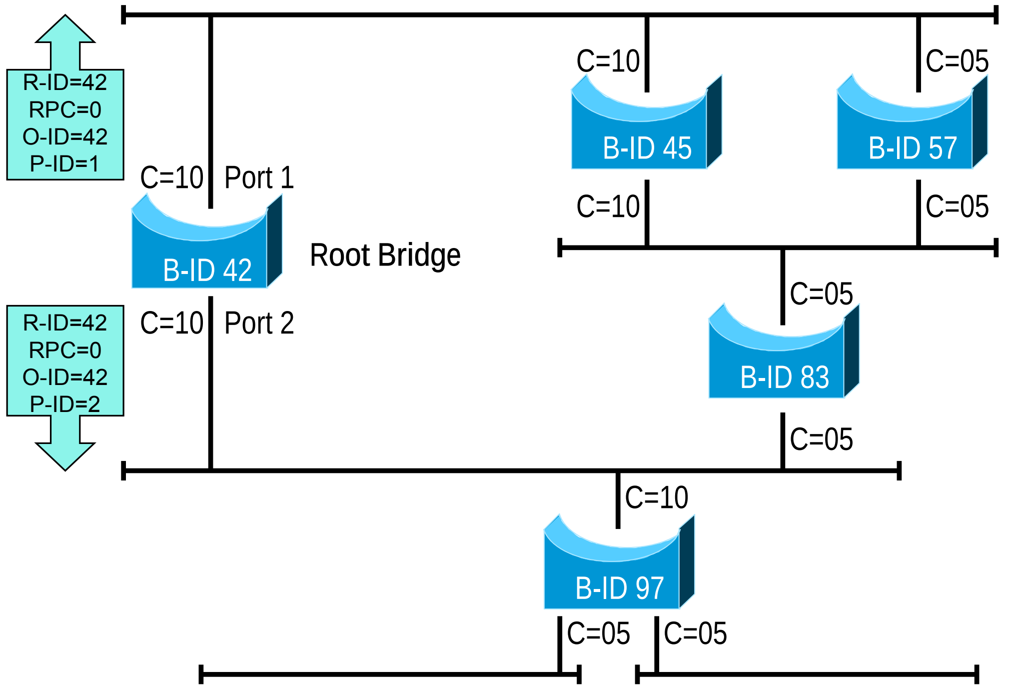
After power up all ports are set in a Blocking State and every bridge tries to become the Root Bridge (RB) of the Spanning Tree by sending Configuration BPDUs.

Blocking state means: End station Ethernet frames are not received and forwarded on such a port but BDPU frames can still be received, manipulated by the bridge and transmitted on such port. BPDU frames are actually filtered based on the well-known multicast address and are given to the CPU of the bridge.

Using such Configuration BPDUs, a bridge tells, which bridge actually is seen as RB, which path costs exist to this RB (Root Path Cost) and its own Bridge ID and Port ID.

L07 - Spanning-Tree Details (v5.0)

Root Bridge Selected, Triggers RPC Calculation



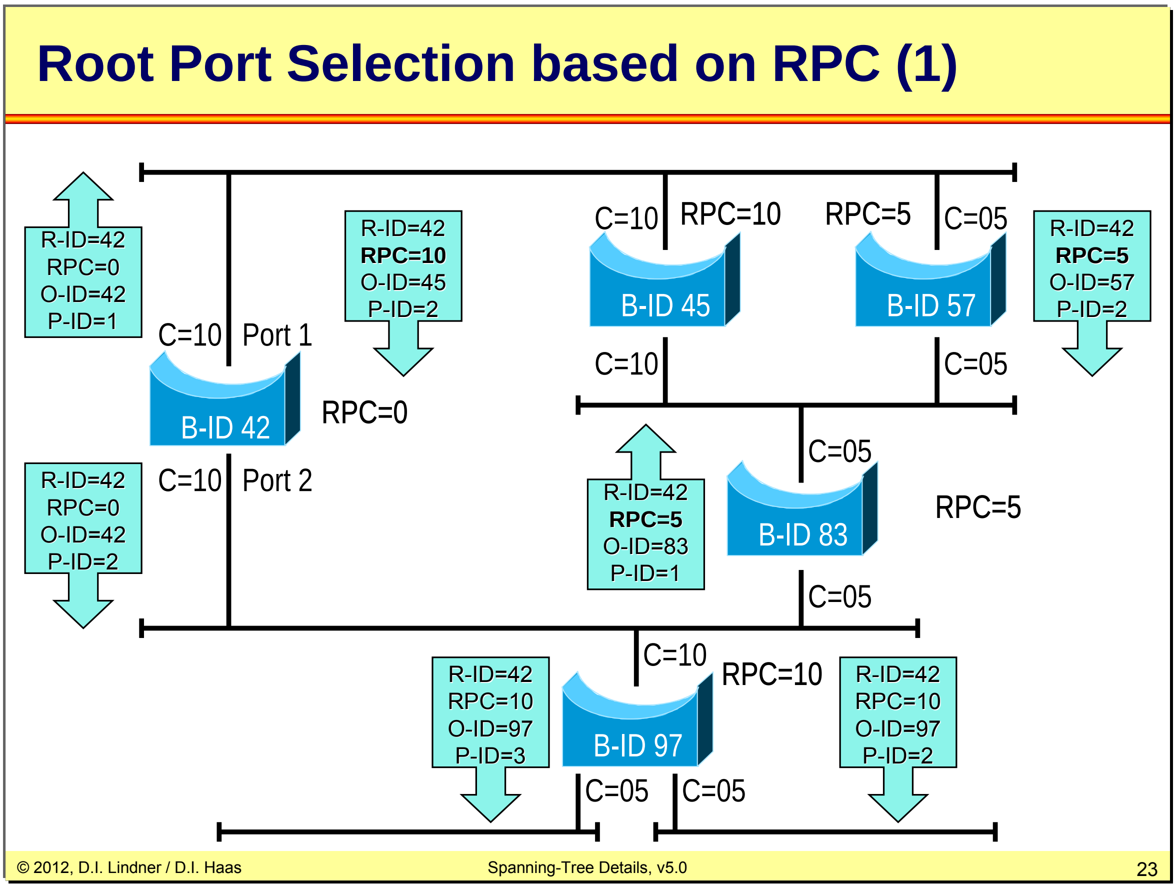
 Spanning-Tree Details, v5.0 22

Bridge with the lowest Bridge ID becomes RB. after selection of the RB all sending of Configuration BPDUs are exclusively triggered by the RB. Other bridges just move such BPDUs on after actualizing the corresponding BPDU fileds.

Strategy to determinate the RB :

If bridge receives a Configuration BPDU with *lower* Root Bridge ID as own Bridge ID the bridge stops sending Configuration BPDUs on this port and the received and adapted Configuration BPDU is forwarded to all other ports.

If bridge receives Configuration BPDU with *higher* Root Bridge ID as own Bridge ID the bridge continues sending Configuration BPDUs with own Bridge ID as proposed Root Bridge ID on all ports, the other bridges should give up.

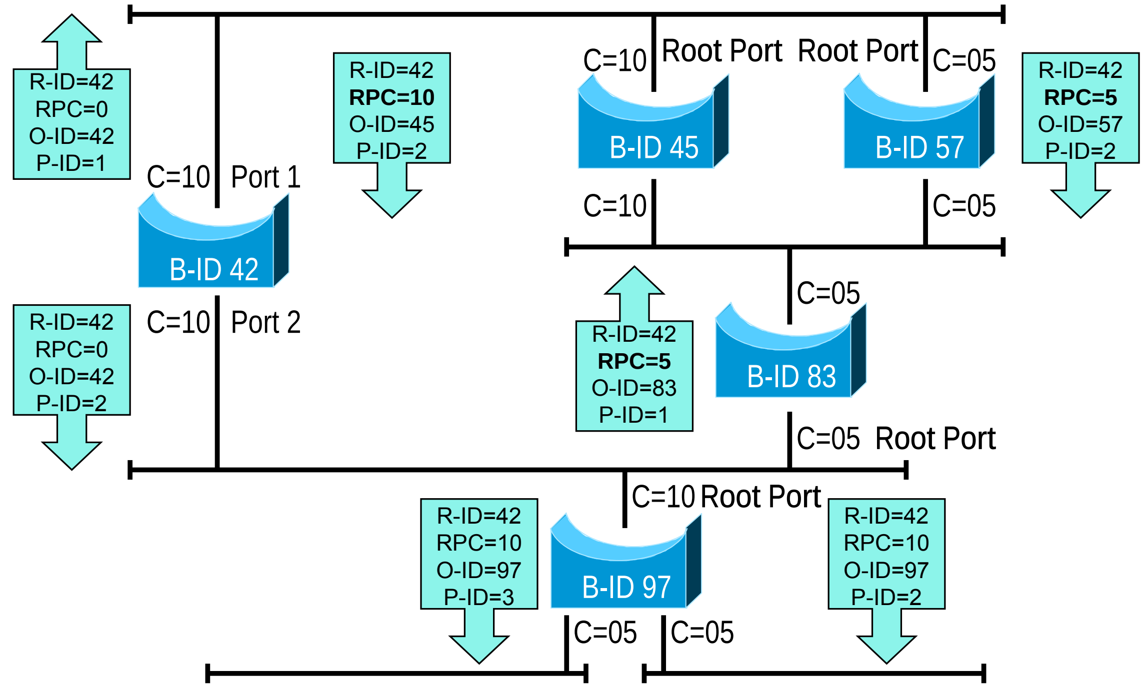


Now, every bridge determines which of its ports has the lowest Root Path Cost. Root Path Cost = sum of all port costs from this bridge to the RB, including port costs of all intermediate bridges. This port becomes the Root Port. In case of equal costs the port ID decides (lower means better).

The principle calculation method: Root Path Cost received in BPDU + port cost of the local port receiving that BPDU.

Similar to Root Bridge selection, a Designated Bridge (DB) is selected for each LAN-segment which is the bridge with the lowest Root Path Cost on its Root Port. In case of equal costs the bridge with the lowest Bridge ID wins again.

Root Port Selection based on RPC (2)



 Spanning-Tree Details, v5.0 24

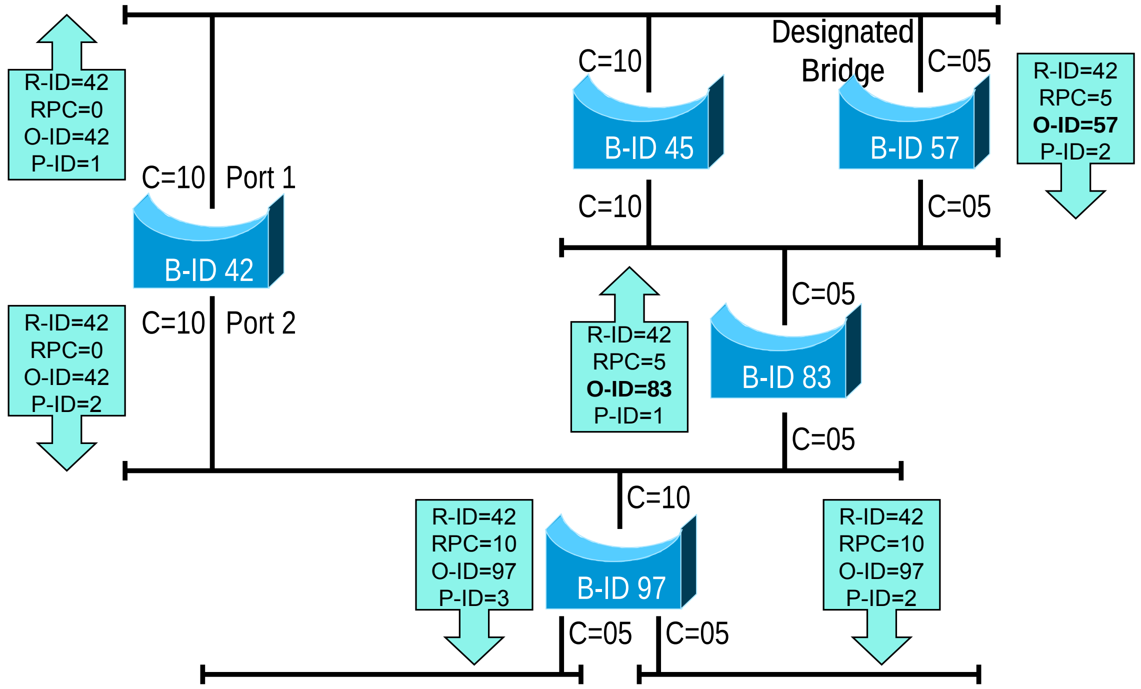
Using the Root Path Cost field in the Configuration BPDU, a bridge indicates its distance to the RB.

Strategy for decision:

If a bridge receives a Configuration BPDU from a bridge which is closer to the RB, the receiving bridge adds its own port costs to the Configuration BPDU and forwards this message to all other ports.

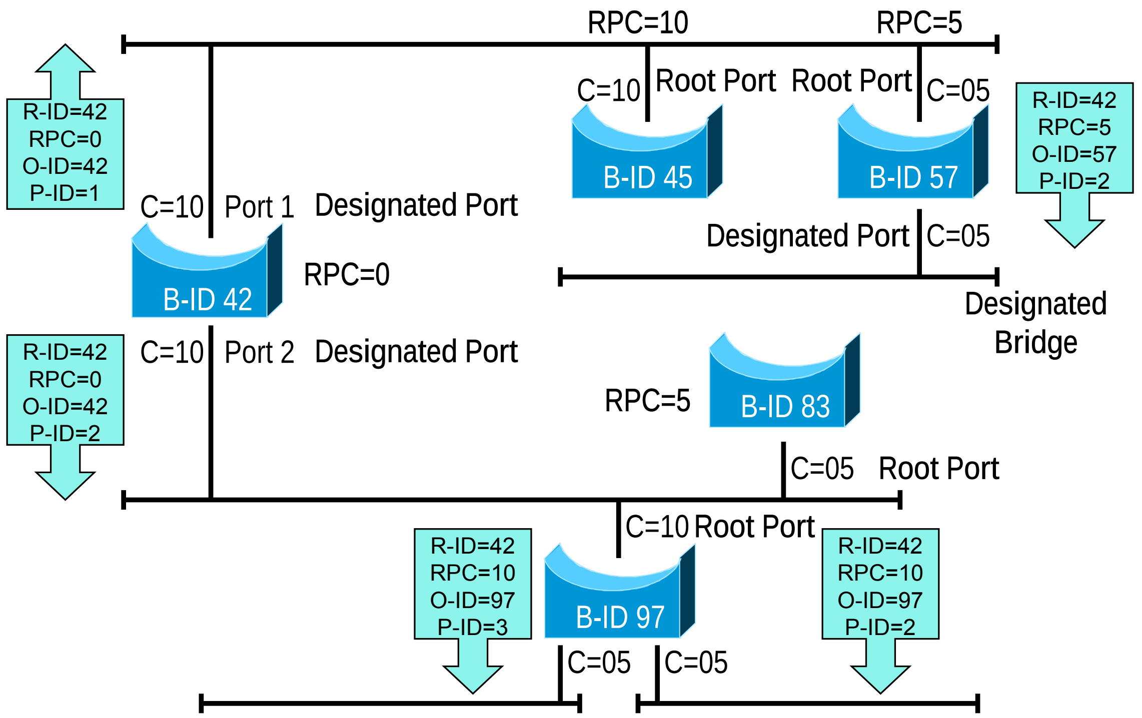
If a bridge receives a Configuration BPDU from a bridge which is more distant to the RB, the receiving bridge drops the message and sends its own Configuration BPDU on this port containing its own Root Path Cost.

Designated Bridge Selection (based on O-ID)



L07 - Spanning-Tree Details (v5.0)

Final Topology



 Spanning-Tree Details, v5.0 26

Procedure Parameters Summary:

Root Bridge -> lowest Bridge ID.

Root Ports via Root Path Costs -> which sum of costs contained in the Configuration BPDU and the receiving interface Port Costs.

Designated Bridge -> lowest Root Path Costs for a given LAN segment.

Root switch has only Designated Ports, all of them are in forwarding state.

Other switches have exactly one Root Port (RP) upstream, zero or more Designated Ports (DP) downstream and zero or more Nondesignated Ports (blocked).

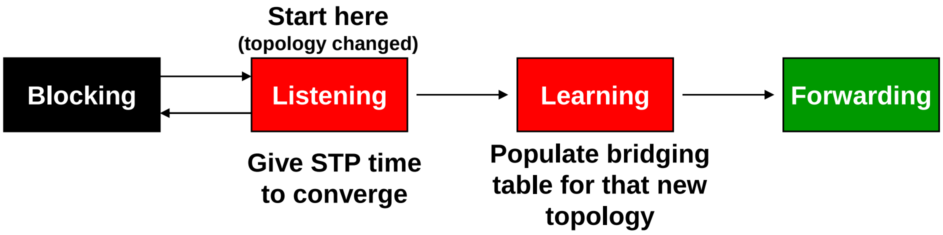
Now every designated bridge declares its ports as designated ports and puts them (together with the Root Port) in the Forwarding State.

All other bridges keep their non-RP and non-DP ports in the Blocking State.

From this moment on, the normal network operation is possible and there is only one path between any two arbitrary end systems.

Redundant links remain in active stand-by mode. If root port fails, other root port becomes active. Still it is reasonable to establish parallel paths in a switched network in order to utilize this redundancy in an event of failure. The STP automatically activates redundant paths if the active path is broken. Note that BPDUs are always sent or received on blocking ports. Note that (very-) low price switches might not support the STP and should not be used in high performance and redundant configurations.

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**

Agenda

- Spanning Tree Protocol (STP)
  - Introduction
  - Details
  - Convergence
  - Some more details
- Rapid Spanning Tree Protocol (RSTP)
- Cisco PVST, PVST+
- Multiple Spanning Tree Protocol (MSTP)

STP Error Detection

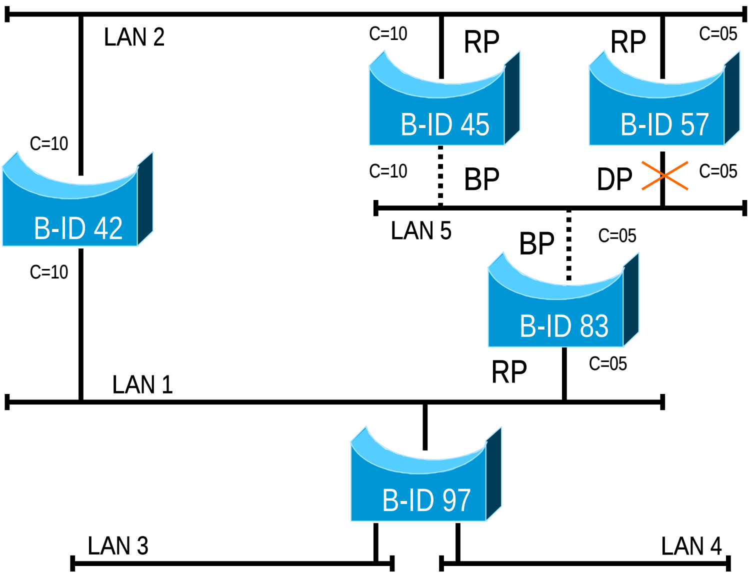
- **The root bridge generates (triggers)**
  - Every 1-10 seconds (hello time interval) a Configuration BPDU to be received on the root port of every other bridge and carried on through the designated ports
  - Bridges which are not designated are still listening to such messages on blocked ports
- **If 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 Details, v5.0 29

Under normal conditions the root bridge generates every hello-time period a "´Heartbeat"-BPDU. All other bridges expected to hear the heartbeat and they have to pass it on in case it is received. If the heartbeat disappears – for whatever reason – however a new STP will be built. During the time of convergence (between 30 and 50 seconds for the old STP, about up to 3-5 seconds for the RSTP) any-to-any connectivity ind the LAN will be disturbed or prevented, hence we have an outage time in the network.

Old STP which is covered in this section is described in the IEEE 802.1D-1998 standard.

STP Convergence Time – Failure at Designated Bridge



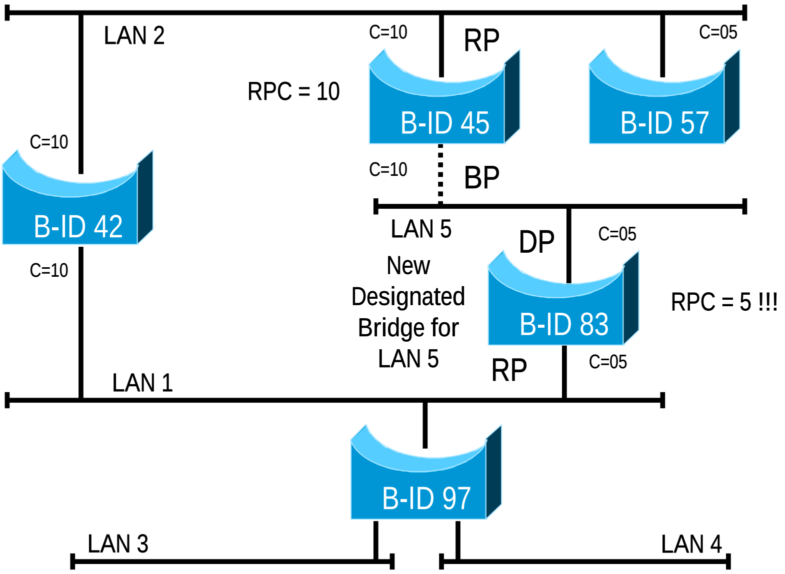
- **Time = max age (20 sec) to be waited until new STP is triggered**

 Spanning-Tree Details, v5.0 30

Scenario 1: Designated port (DP) of Bridge 57 fails. Bridge 45 and bridge 83 do not receive the heartbeat on their blocked ports (BP) anymore although heartbeat is seen on their root ports (RP). After max-age time (20 seconds) a new STP is triggered by bridge 45 and bridge 83.

L07 - Spanning-Tree Details (v5.0)

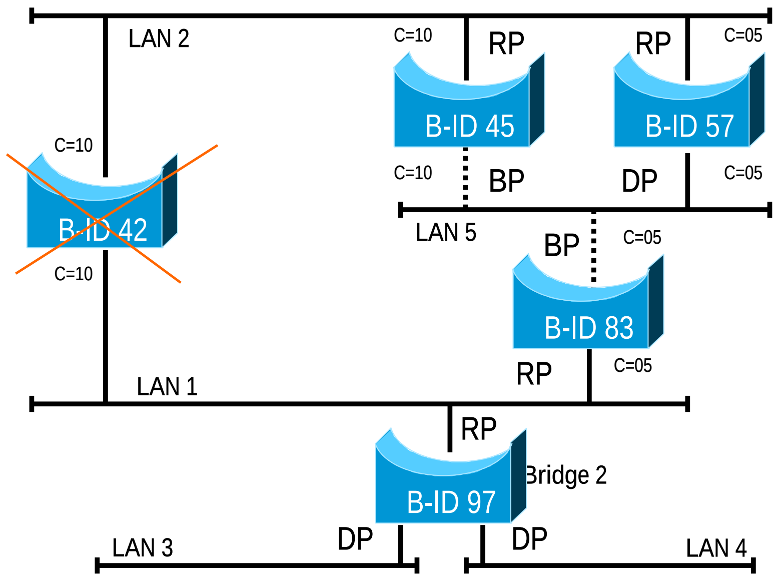
STP Convergence Time – Failure at Designated Bridge – New Topology



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

Scenario 1: Here you see the new topology. Bridge 83 became the designated bridge for LAN5.

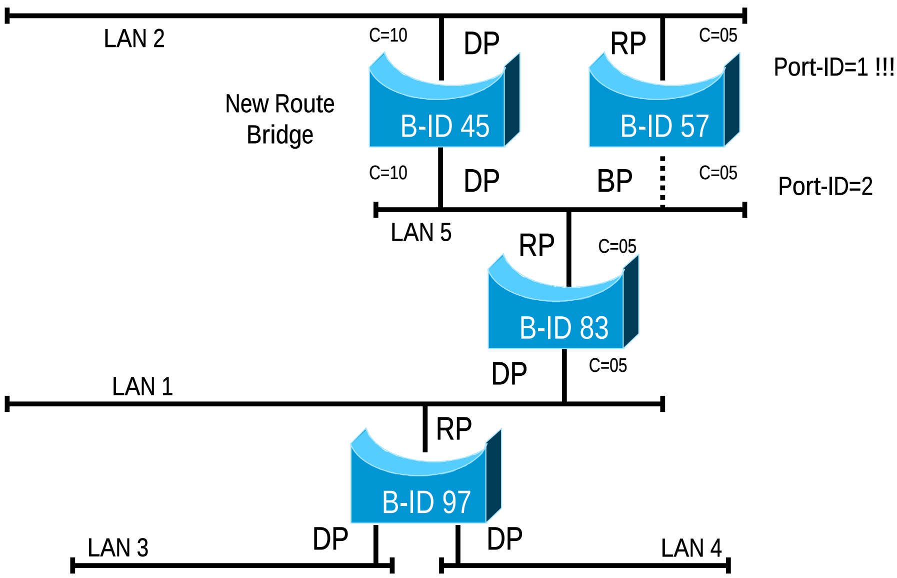
STP Convergence Time – Failure of Root Bridge



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

Scenario 2: Root bridge 42 fails. All other bridges do not receive the heartbeat neither on their root ports nor on their blocked ports (BP). After max-age time (20 seconds) a new STP is triggered by all remaining bridges 45.

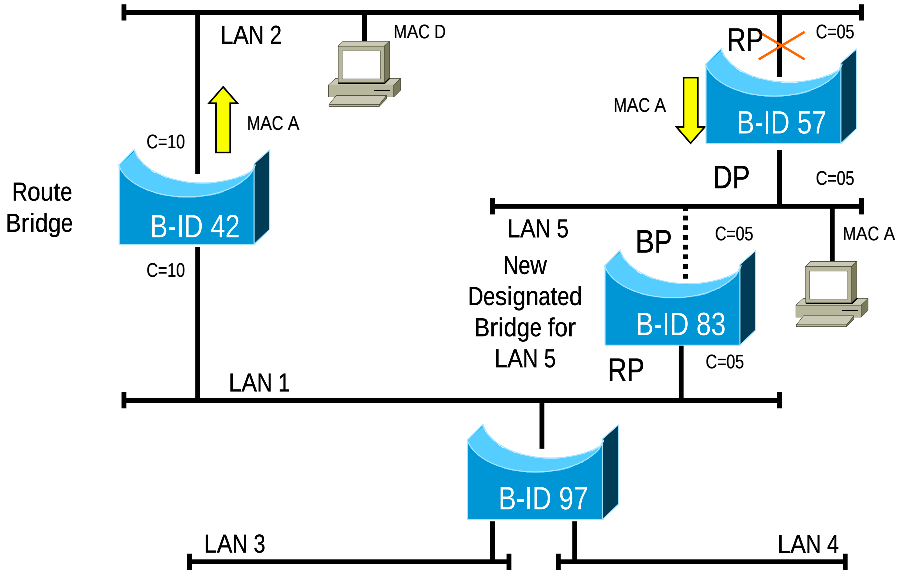
STP Convergence Time – Failure of Root Bridge – New Topology



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

Scenario 2: Here you see the new topology. Bridge 45 became the new root bridge. Bridge 57 has equal RPC on both ports hence the port-id decides which is RP and which is BP.

STP Convergence Time – Failure of Root Port



- **Time = max age (20 sec) has not to be waited until new STP is triggered**

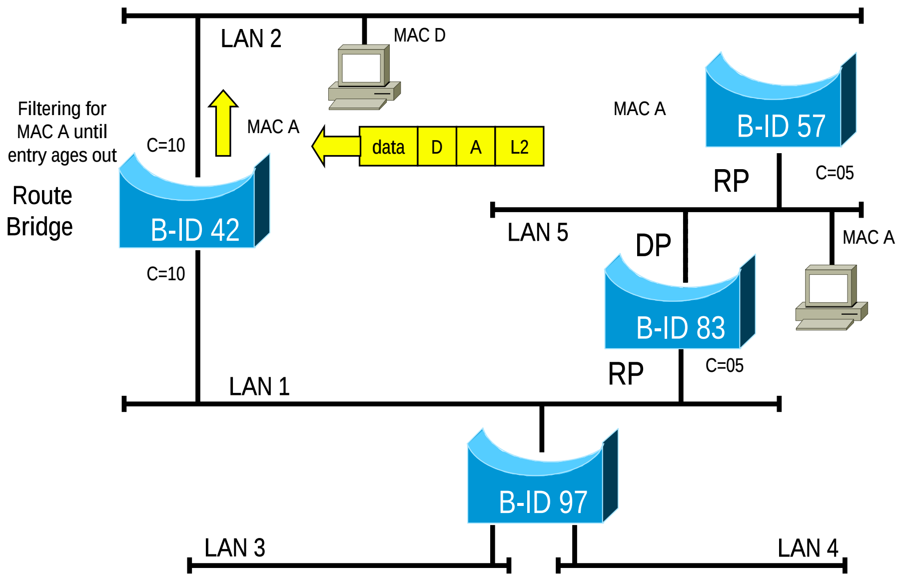
 Spanning-Tree Details, v5.0 34

Scenario 3: RP of Bridge 57 fails. In that case bridge 57 has not to wait for max-age period before triggering the new STP. Reason: Bridge is designated bridge but RP fails and there is no other connectivity to the root bridge possible

Yellow arrows show the signposts in the bridging table to reach MAC address A before the failure.

L07 - Spanning-Tree Details (v5.0)

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



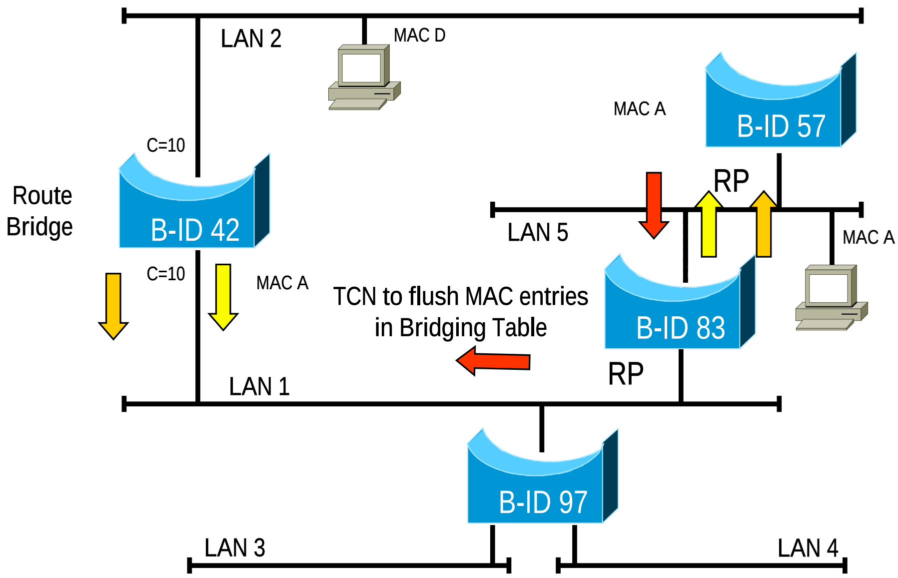
- **Convergence Time = 2*forward delay (15 sec Listening + 15 sec Learning) = 30 sec**

Scenario 3: Here you see the new topology. Bridge 83 became the new designated bridge for LAN5.

Recognize what happens if station D sends a frame to station A. The pointer in bridge 42 still points in the wrong direction and the frame will be filtered by bridge 42 until the entry times out after 5 minutes. Of course if A would send a broadcast frame the table would immediately be repaired but what if not.

Hence bridges should install an additional procedure to overcome such situations without interaction of end-system functionality like the mentioned broadcast of A. This procedure is called topology notification.

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



 Spanning-Tree Details, v5.0 36

Bridge 57 and 83 send TCN BPDUs out on their Root Ports (red arrows: TC bit set). After such a message is received by an upstream bridge it will be locally acknowledged by the upstream bridge in the reverse direction (yellow arrows: TCA bit set). If that finally appears to the root bridge, the root will sent a Conf BPDU with both flags set (orange arrows: TC and TCA bit set) for 35 seconds which has to be passed on downstream by the other bridges. All switches receiving TC+TCA=1 will age out (flush) their bridging tables in 15 seconds instead of waiting for 3 minutes.

STP Disadvantages

- **Active paths are always calculated from the root, but the actual information flow of the network may use other paths**
  - Note: network-manager can control this via Bridge Priority, Path Costs und Port Priority to achieve a certain topology under normal operation
  - Hence STP should be designed to overcome plug and play behavior resulted by default values
- **Redundant paths cannot be used for load balancing**
  - Redundant bridges may be never used if there is no failure of the currently active components
  - For remote bridging via WAN the same is true for redundant WAN links
- **Convergence time between 30 and 50 seconds**
  - Note: in order to improve convergence time Rapid Spanning Tree Protocol has been developed (802.1D version 2004)

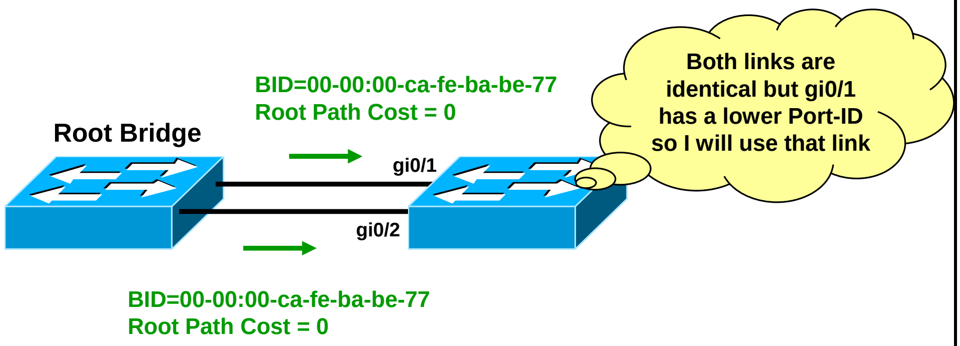
Note: Old STP which is covered in this section is described in the IEEE 802.1D-1998 standard.

Agenda

- **Spanning Tree Protocol (STP)**
  - Introduction
  - Details
  - Convergence
  - Some more details
- **Rapid Spanning Tree Protocol (RSTP)**
- **Cisco PVST, PVST+**
- **Multiple Spanning Tree Protocol (MSTP)**

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



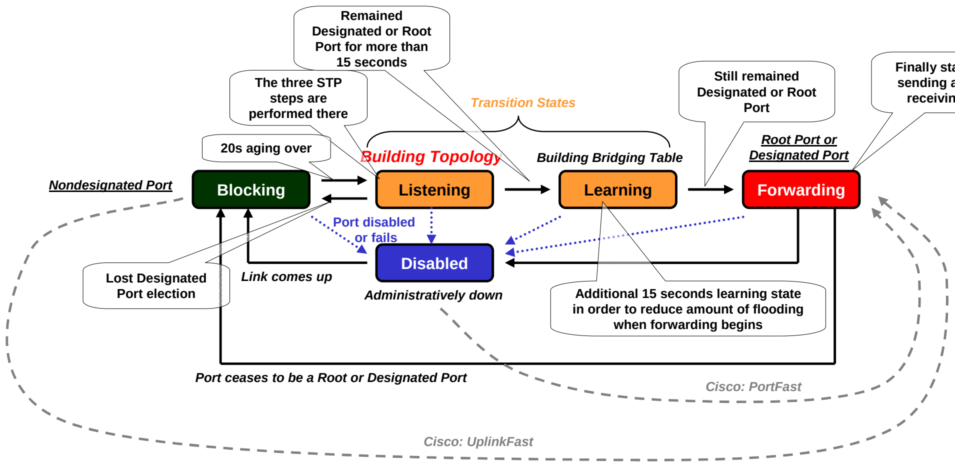
Importance of details…

- **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	Port States
Root	Disabled
Designated	Blocking
Nondesignated	Listening
	Learning
	Forwarding

 Spanning-Tree Details, v5.0 42

A specific port role is a long-term "destiny" for a port, while port states denote transient situations. The maximum-aging time is the number of seconds a switch waits without receiving spanning-tree configuration messages before attempting a reconfiguration.

From the 802.1D-1998 standard:

If the Bridge times out the information held for a Port, it will attempt to become the Designated Bridge for the LAN to which that Port is attached, and will transmit protocol information received from the Root on its Root Port on to that LAN.

If the Root Port of the Bridge is timed out, then another Port may be selected as the Root Port. The information transmitted on LANs for which the Bridge is the Designated Bridge will then be calculated on the basis of information received on the new Root Port.

L07 - Spanning-Tree Details (v5.0)

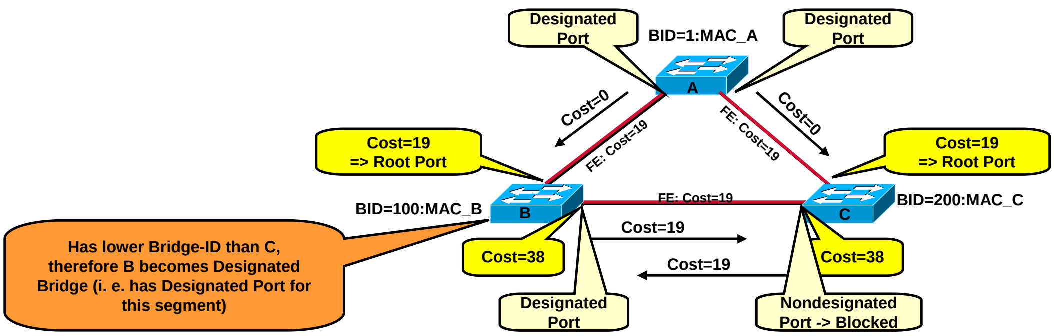
Example with L2 Switches

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



© 2012, D.I. Lindner / D.I. Haas Spanning-Tree Details, v5.0 43

Each segment has exactly one Designated Port. This simple rule actually breaks any loops.

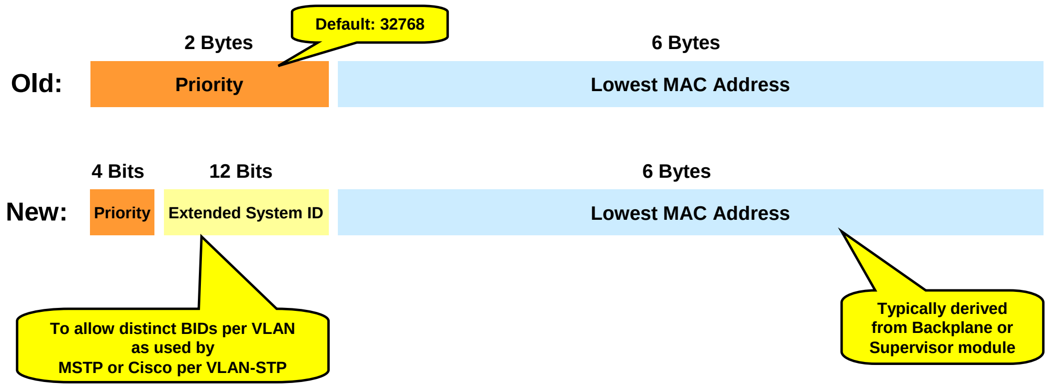
A nondesignated port receives a more useful BPDU than the one it would send out on its segment. Therefore it remains in the so-called blocking state.

Port ID - Contains a unique value for every port. Port 1/1 contains the value 0x8001, whereas Port 1/2 contains 0x8002. (Or in decimal: 128.1, 128.2, …)

From the 802.1D-1998 standard:

Each Configuration BPDU contains, among other parameters, the unique identifier of the Bridge that the transmitting Bridge believes to be the Root, the cost of the path to the Root from the transmitting Port, the identifier of the transmitting Bridge, and the identifier of the transmitting Port. This information is sufficient to allow a receiving Bridge to determine whether the transmitting Port has a better claim to be the Designated Port on the LAN on which the Configuration BPDU was received than the Port currently believed to be the Designated Port, and to determine whether the receiving Port should become the Root Port for the Bridge if it is not already.

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 …

802.1T spanning-tree extensions, and some of the bits previously used for the switch priority are now used for the extended system ID (VLAN identifier for the per-VLAN spanning-tree plus [PVST+] and for rapid PVST+ or an instance identifier for the multiple spanning tree [MSTP]).

Before this, spanning tree used one MAC address per VLAN to make the bridge ID unique for each VLAN.

Extended system IDs are VLAN IDs between 1025 and 4096. Cisco IOS Releases 12.1(14)E1 and later releases support a 12-bit extended system ID field as part of the bridge ID.

L07 - Spanning-Tree Details (v5.0)

Detailed BPDU Format

Field	Bytes	Description
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)

A TCN-BPDU only consists of these 3 fields !!! (Protocol ID, Version, Message Type)

When first booted, Root-ID == BID (Root ID)

If value increases, then the originating bridge lost connectivity to Root Bridge (Message Age)

Maximum Age, Hello Time, Forward Delay:
- Predetermined by root bridge
- Affect convergence time
- Misconfigurations cause loops

- **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

 Spanning-Tree Details, v5.0 45

In normal stable operation, the regular transmission of Configuration Messages by the Root ensures that topology information is not timed out. To allow for reconfiguration of the Bridged LAN when components are removed or when management changes are made to parameters determining the topology, the topology information propagated throughout the Bridged LAN has a limited lifetime. This is effected by transmitting the age of the information conveyed (the time elapsed since the Configuration Message originated from the Root) in each Configuration BPDU. Every Bridge stores the information from the Designated Port on each of the LANs to which its Ports are connected, and monitors the age of that information.

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!
  - It is transported as TCN BPDU
- **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)**
  - Which is actually a Conf BPDU from the upstream bridge

L07 - Spanning-Tree Details (v5.0)

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 and TCA 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

Main idea: To avoid 5 minute age timer upon topology change! Some destinations may not be reachable any more!

Normally, all Configuration BPDUs are (periodically) sent by the root bridge. Other bridges never send out a BPDU toward the root bridge! Therefore dedicated TCN messages have been defined to allow a non-root bridge to announce topology changes.

TCN BPDUs are sent on the root port until acknowledged by the upstream bridge (BPDU with the topology change acknowledgement (TCA) bit set). The TCN is sent every hello-time which is a locally configured value (not the hello-time specified in configuration BPDUs)

Reasons to send TCNs:

1. When a port changes from "Forwarding" to any other state

2. When a port transitions to forwarding and the bridge has a designated port (that is the bridge is not standalone).

Then a TCN is sent upstream to the root bridge (i. e. only sent through the root port) which 'broadcasts' this information downstream to all other bridges.

1. These downstream TCNs are not acknowledged

2. The TC bit is set by the root for a period of max-age + forward-delay seconds, which is 20+15=35 seconds by default.

3. Every bridge now reduces the aging time of every existing bridging table entry to 15 seconds (more precisely: the actual value of forward-delay) This is done (also for new entries) for the duration of 35 seconds (more precisely: max-age + forward-delay).

Cisco 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 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

Any connectivity problems after cold booting a PC in the morning but NOT after warm-booting during the day?

Cisco Uplink Fast (1)

- **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)

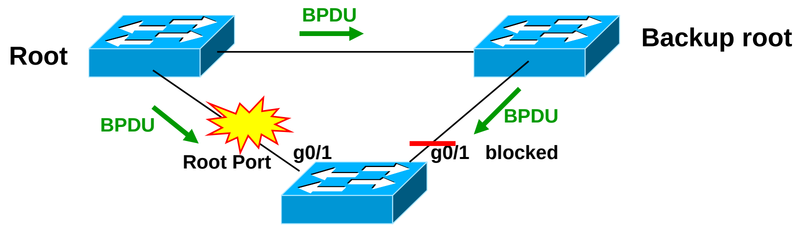
UplinkFast is actually a root port optimization.

The standard Cisco mcast address 01-00-0C-CC-CC-CC, which is used for CDP, VTP, DTP, and DISL cannot be used, because all Cisco devices are programmed to not flood these frames (rather consume it).

Note that only MACs not learned over the uplinks are flooded.

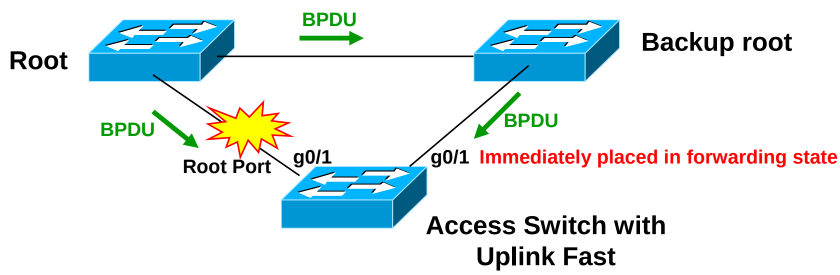
Cisco Uplink Fast (2): The Problem

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



Cisco Uplink Fast (3): Idea

- **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

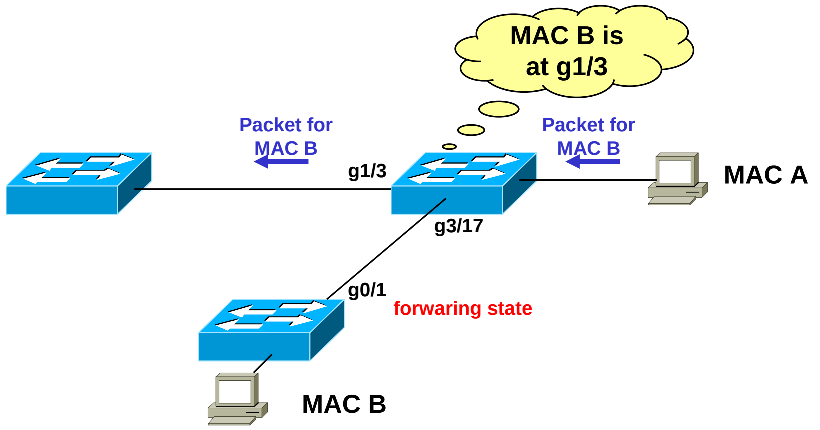


 Spanning-Tree Details, v5.0 51

The UplinkFast feature is based on the definition of an uplink group. On a given switch, the uplink group consists in the root port and all the ports that provide an alternate connection to the root bridge. If the root port fails, which means if the primary uplink fails, a port with next lowest cost from the uplink group is selected to immediately replace it.

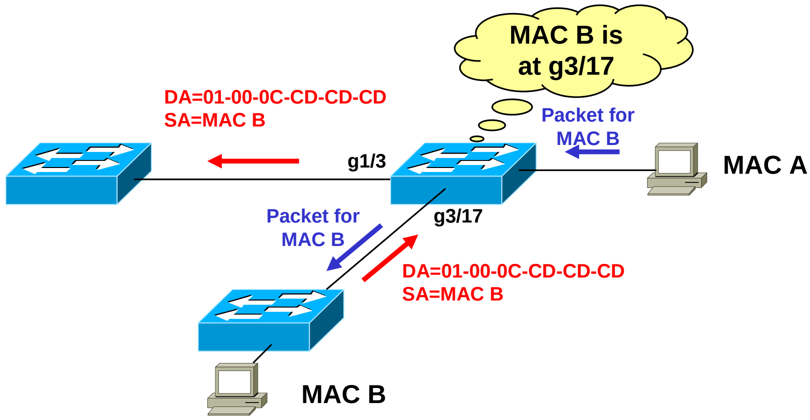
Cisco Uplink Fast (4): Incorrect Bridging Tables

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



Cisco Uplink Fast (5): 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



Cisco Uplink Fast (6): Additional 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)

1100xxxx xxxxxxxx = 49152=2^15+2^14

Cisco Backbone Fast (1)

- **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!**

 Spanning-Tree Details, v5.0 55

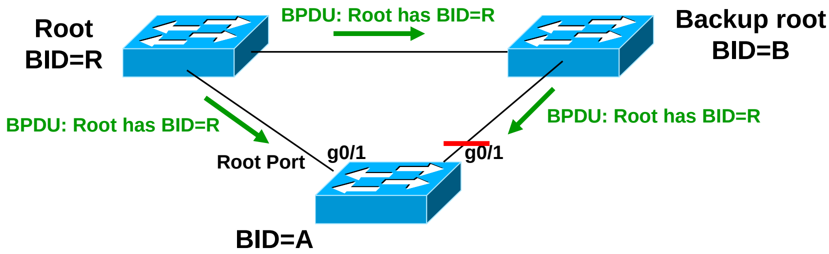
BackboneFast is actually a Max Age optimization.

Upon Root Port failure, a switch assumes it Root role and generates own Configuration BPDUs, which are treated as "inferior" BPDUs, because most switches might still receive the BPDUs from the original Root Bridge.

The request/response mechanism involves a so-called Root Link Query (RLQ) protocol, that is, RLQ-requests are sent to upstream bridges to check whether their connection to the Root Bridge is stable. Upstream bridges reply with RLQ-responses. If the upstream bridge does not know about any problems, it forwards the RLQ-request further upwards, until the problem is solved. If the RLQ-response is received by the downstream bridge on a non-Root Port, then this bridge knows, that it has lost its connection to the Root Bridge and can immediately expire the Max Age timer.

Cisco Backbone Fast (2): The Problem

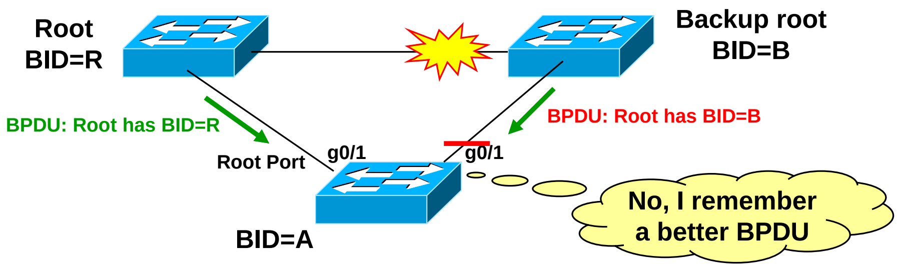
- Consider initial situation
- Note that blocked port (g0/1) always remembers "best seen" BPDU – which has best (=lowest) Root-BID



L07 - Spanning-Tree Details (v5.0)

Cisco Backbone Fast (3): The 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



 Spanning-Tree Details, v5.0 57

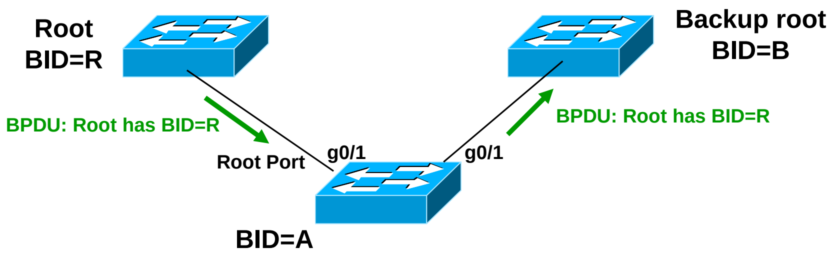
Note that the key problem is this:

1) Direct link failures would immediately set the bridge in listening mode (i. e. all of its ports).

2) But indirect link failures always includes the max-age timer (20 s) before entering the listening state.

Cisco Backbone Fast (4): The Problem (cont.)

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



Cisco Backbone Fast (5): The 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

Other CISCO 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**

Agenda

- **Spanning Tree Protocol (STP)**
  - Introduction
  - Details
  - Convergence
  - Some more details
- **Rapid Spanning Tree Protocol (RSTP)**
- **Cisco PVST, PVST+**
- **Multiple Spanning Tree Protocol (MSTP)**

Introduction

- **RSTP is part of the IEEE 802.1D-2004 standard**
  - Originally defined in IEEE 802.1w
  - Old STP IEEE 802.1D-1998 is now superseded by RSTP
- **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
- **Designed to be compatible and interoperable with the traditional STP – without additional management requirements**

 Spanning-Tree Details, v5.0 62

RSTP is designed to be compatible and interoperable with the traditional STP (IEEE 802.1D version 1998) – without additional management requirements. If an RSTP-enabled bridge is connected to an STP bridge, only Configuration-BPDUs and Topology-Change BPDUs are sent but no port role negotiation is supported.

An RSTP Bridge Port automatically adjusts to provide interoperability, if it is attached to the same LAN as an STP Bridge. Protocol operation on other ports is unchanged. Configuration and Topology Change Notification BPDUs are transmitted instead of RST BPDUs which are not recognized by STP Bridges. Port state transition timer values are increased to ensure that temporary loops are not created through the STP Bridge. Topology changes are propagated for longer to support the different FilteringDatabase flushing paradigm used by STP. It is possible that RSTP's rapid state transitions will increase rates of frame duplication and misordering.

BPDUs convey Configuration and Topology Change Notification (TCN) Messages. A Configuration Message can be encoded and transmitted as a Configuration BPDU or as an RST BPDU. A TCN Message can be encoded as a TCN BPDU or as an RST BPDU with the TC flag set. The Port Protocol Migration state machine determines the BPDU types used.

In most cases, RSTP performs better than Cisco's proprietary extensions (Port-Fast, Uplink-Fast, Backbone-Fast) without any additional configuration. 802.1w is also capable of reverting back to 802.1d in order to interoperate with legacy bridges (thus dropping the benefits it introduces) on a per-port basis.

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 (typically within 1 – 5 seconds)
- **Better scalability**
  - No network diameter limit
- **New port roles and port states**
  - Non-Designated Port role split in Alternate and Backup
  - Root Port and Designated Port role still remain the same
  - Port state discarding instead of disabled, learning and blocking

Remember:

Root Port Role: Receives the best BPDU (so it is closest to the root bridge).

Designated Port Role: A port is designated if it can send the best BDPU on the segment to which it is connected. On a given segment, there can be only one path towards the root-bridge.

Port States Comparison

STP (802.1d) Port State	RSTP (802.1w) Port State	Is Port included in active Topology?	Is Port learning MAC addresses?
disabled	discarding	No	No
blocking	discarding	No	No
listening	discarding	Yes	No
learning	learning	Yes	Yes
forwarding	forwarding	Yes	Yes

There are only 3 port states left in RSTP, corresponding to the 3 possible operational states. The 802.1d states disabled, blocking and listening have been merged into a unique 802.1w discarding state.

There is no difference between a port in blocking state and a port in listening state; they both discard frames and do not learn MAC addresses. The real difference lies in the role the spanning tree assigns to the port. It can safely be assumed that a listening port will be either a designated or root and is on its way to the forwarding state. Unfortunately, once in forwarding state, there is no way to detect from the port state whether the port is root or designated, which contributes to demonstrating the failure of this state-based terminology. RSTP addresses this by decoupling the role and the state of a port.

The role is now a variable assigned to a given port. The root port and designated port roles remain, while the blocking port role is now split into the backup and alternate port roles.

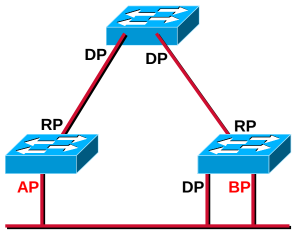
A non-designated port is a blocked port that receives a more useful BPDU than the one it would send out on its segment. The "more useful BPDU" can be received from the same switch (on another port on the same LAN segment) or from another switch (also on the same LAN segment). The first is called a **backup** port, the latter an **alternate** port.

Note: To make the confusion even worse -> The name *blocking* is used for the *discarding state* in Cisco implementations!!!

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:

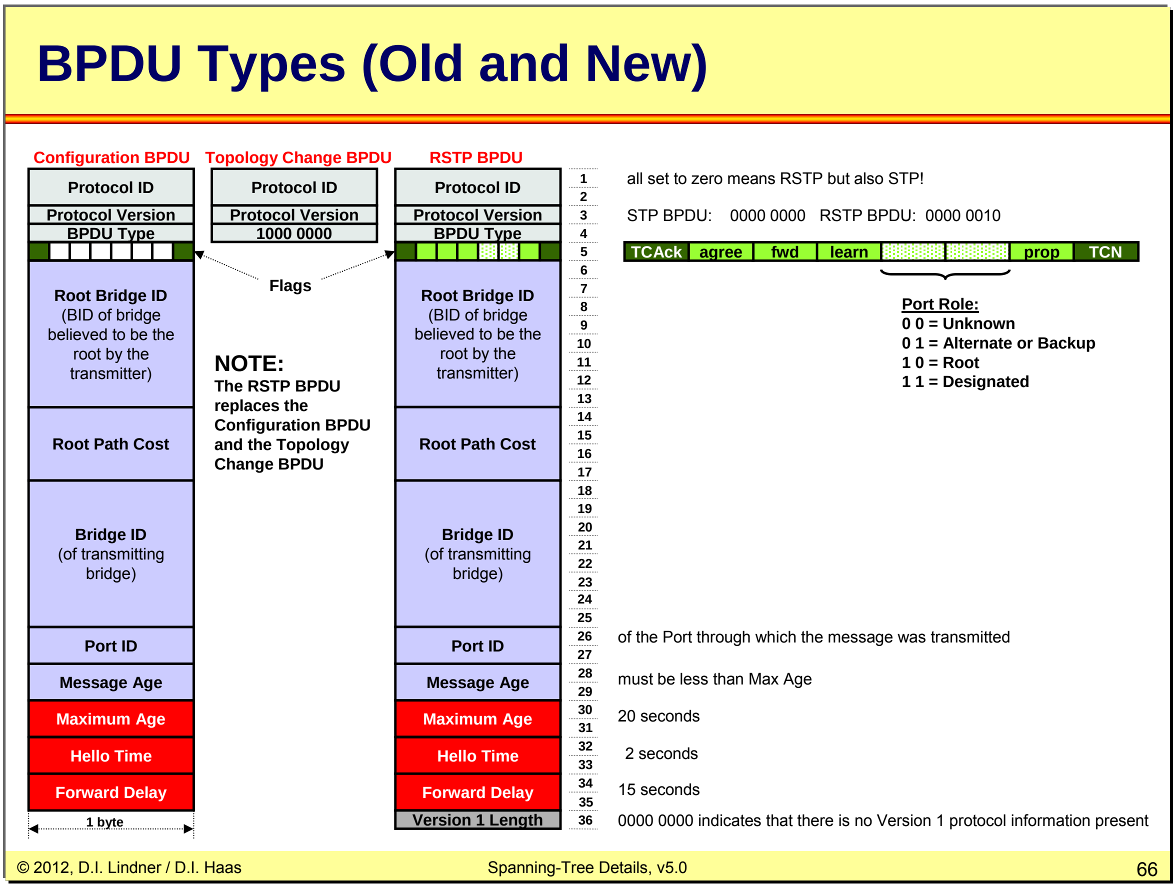


AP alternate port, BP is now backup port.

Alternate Port: A port blocked by receiving better BPDUs from a different bridge. It provides an alternate path to the root bridge

Backup Port: A port blocked by receiving better BPDUs from the same bridge. Provides a redundant connectivity to the same segment.

L07 - Spanning-Tree Details (v5.0)



Note1: A Configuration BPDU has same structure than a RSTP BPDU with the following exceptions:

1) A Configuration BPDU is only 35 byte long, that is, there is no "Version 1 length" field

2) A Configuration BPDU only uses two flags, that is, TCAck (bit 7) and TCN (bit 0)

3) BPDU type differentiate between CONF BPD and TCN BPDU

Note2: If the Unknown value of the Port Role parameter is received, the state machines will effectively treat the RST

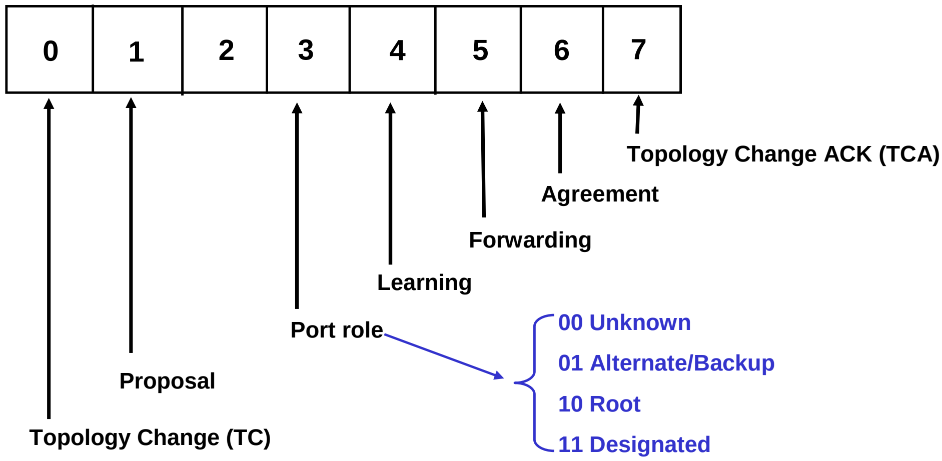
BPDU as if it were a Configuration BPDU.

Flags:

- TCN (bit 1)
- Proposal (bit 2)
- Port Role (bits 3, 4)
- Learning (bit 5)
- Forwarding (bit 6)
- Agreement (bit 7)
- Topology Change Acknowledgment (bit 8)

BPDU Flag Field – New Values

- TC and TCA were already introduced by old STP
- Other bits were unused by old STP
- RSTP also uses the 6 remaining bits

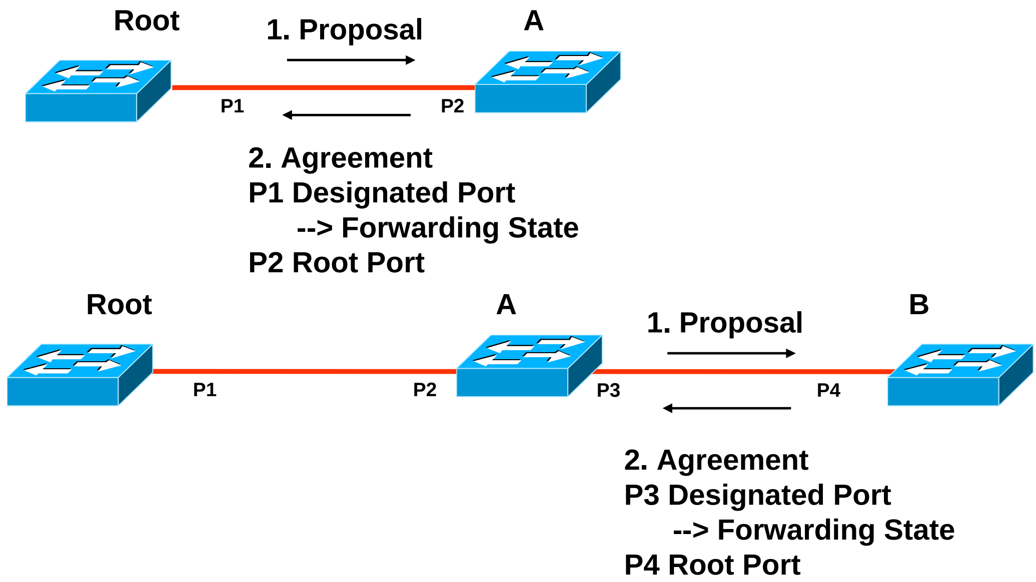


 Spanning-Tree Details, v5.0 67

The new bits encode the role state of the port originating the BPDU and handle the proposal/agreement mechanism.

Proposal/Agreement Sequence

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



 Spanning-Tree Details, v5.0 68

There is an explicit handshake between bridges upon link up event. The bridge sends a proposal to become designated for that segment. The remote bridge responses with an agreement if the port on which it received the proposal is the root port of the remote bridge. As soon as receiving an agreement, the bridge moves the port to the forwarding state. If the remote bridge has a better role like it is nearer to the root bridge or is the root bridge itself, it will not accept the proposal but will send an own proposal. Whatever is that case, the role and state of the ports is settled within exchange of 2 or 4 messages.

NEW BPDU Handling

- **Faster Failure Detection**
  - BPDUs acting now as keepalives messages
    - Different to the 802.1D STP a bridge now sends a BPDU with its current information every <hello-time> seconds (2 by default), even if it does not receive any BPDU from the root bridge
  - If hellos are not received for 3 consecutive times, port information is invalidated
    - Because BPDU's are now used as keepalive mechanism between bridges
    - If a bridge fails to receive BPDUs from a neighbor, the connection has been lost
  - Max age not used anymore
    - For listening and waiting for STP to converge

Rapid Transition to Forwarding State is the most important feature in 802.1w. The legacy STP was passively waiting for the network to converge before turning a port into the forwarding state

New RSTP is able to actively confirm that a port can safely transition to forwarding. It is a real feedback mechanism, that takes place between RSTP-compliant bridges through proposal / agreement sequence.

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 is necessary!

On a given port, if hellos are not received three consecutive times, protocol information can be immediately aged out (or if max-age expires). Because of the previously mentioned protocol modification, BPDUs are now used as a keepalive mechanism between bridges. A bridge considers that it loses connectivity to its direct neighbor root or designated bridge if it misses three BPDUs in a row. This fast aging of the information allows quick failure detection. If a bridge fails to receive BPDUs from a neighbor, it is certain that the connection to that neighbor is lost. This is opposed to 802.1D where the problem might have been anywhere on the path to the root.

Rapid transition is the most important feature introduced by 802.1w. The legacy STP passively waited for the network to converge before it turned a port into the forwarding state. The achievement of faster convergence was a matter of tuning the conservative default parameters (forward delay and max-age timers) and often put the stability of the network at stake. The new rapid STP is able to actively confirm that a port can safely transition to the forwarding state without having to rely on any timer configuration. There is now a real feedback mechanism that takes place between RSTP-compliant bridges. In order to achieve fast convergence on a port, the protocol relies upon two new variables: edge ports and link type.

Link Types and Edge Port

- **Shared Link (Half Duplex !!!)**
  - Are not supported by RSTP (ambiguous negotiations could result)
  - Instead slow standard STP is used here
- **Point-to-point Link (Full Duplex !!!)**
  - Supports proposal-agreement process
- **Edge Port**
  - Hosts reside 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
- **RSTP fast transition**
  - Only possible on edge ports or point-to-point links

 Spanning-Tree Details, v5.0 71

RSTP can only achieve rapid transition to forwarding: on edge ports (either full-duplex or half-duplex) or on point-to-point links (trunks between L2 switches using full-duplex), but not on shared links.

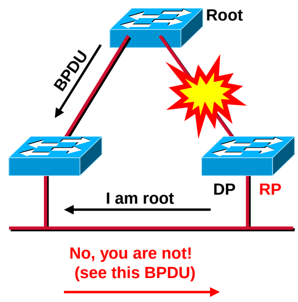
Edge ports, which are directly connected to end stations, cannot create bridging loops in the network and can thus directly perform on link setup transition to forwarding, skipping the listening and learning states of old STP.

Link type shared or point-to-point is automatically derived from the physical duplex mode of a port: A port operating in full-duplex will be assumed to be point-to-point, a port operating in half-duplex will be assumed to be a shared port.

Main Differences to STP

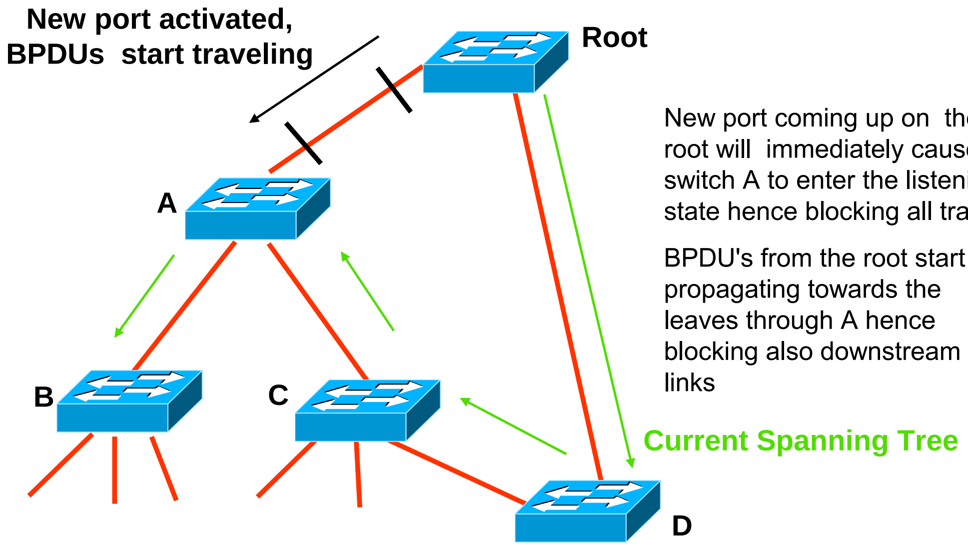
- **BPDUs are sent every hello-time, and not simply relayed anymore**
  - Immediate aging if three consecutive BPDUs are missing
- **When a bridge receives better information ("I am root") from its DB, it immediately accepts it and replaces the one previously stored**
  - But if the RB is still alive,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

BackboneFast-like behavior:



Slow Convergence with Legacy STP 1

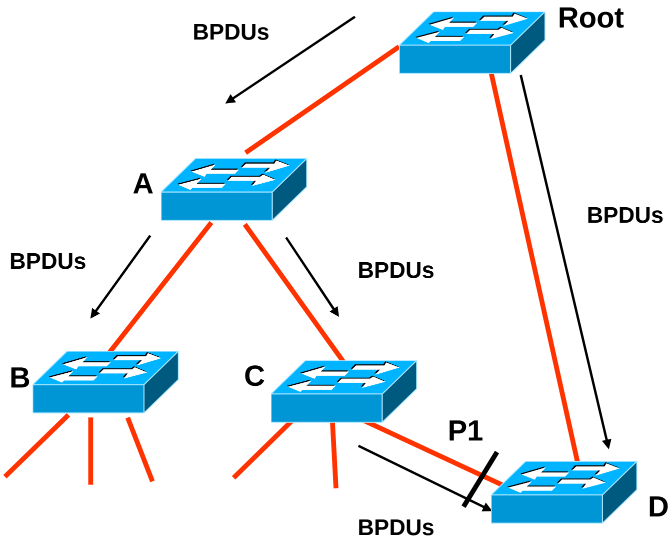
A new link between A and Root is being added to the bridged network



New port coming up on the root will immediately cause switch A to enter the listening state hence blocking all traffic

BPDU's from the root start propagating towards the leaves through A hence blocking also downstream links

Slow Convergence with Legacy STP 2

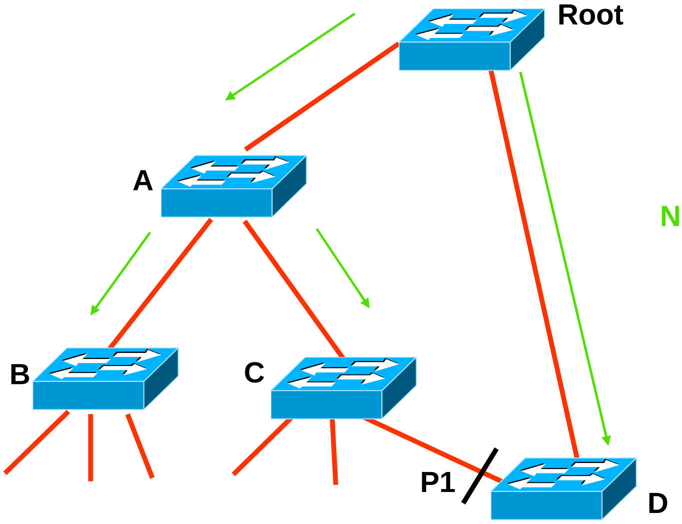


Very quickly, the BPDUs from the root bridge reach D that immediately blocks its port P1.

The topology has now converged, but the network is disrupted for twice forward delay because all switches needs time for listening (STP convergence time) and learning

30 seconds no network connectivity !!!

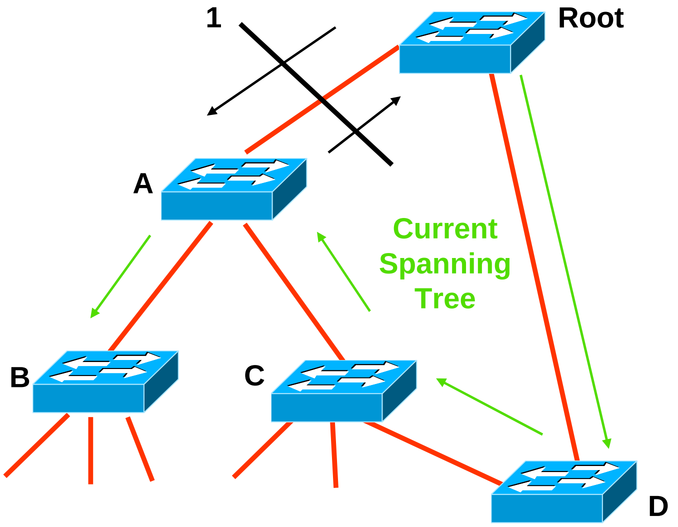
Slow Convergence with Legacy STP 3



 Spanning-Tree Details, v5.0 75

Fast Convergence with RSTP 1

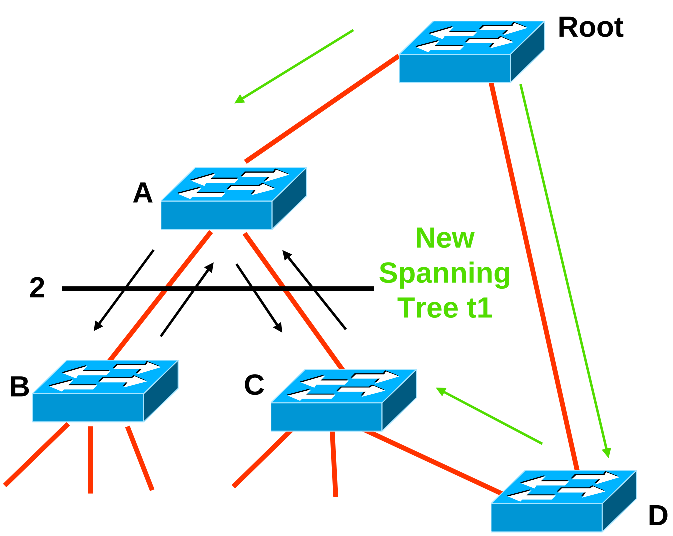
A new link between A and Root is being added to the bridged network



Both ports on link between A and the root are put in so called designated blocking as soon as they come up.

As soon as A receives the roots BPDU, it blocks its non-edge designated ports until synchronization is achieved. Through the agreement A explicitly authorizes the root bridge to put its port in forwarding

Fast Convergence with RSTP 2

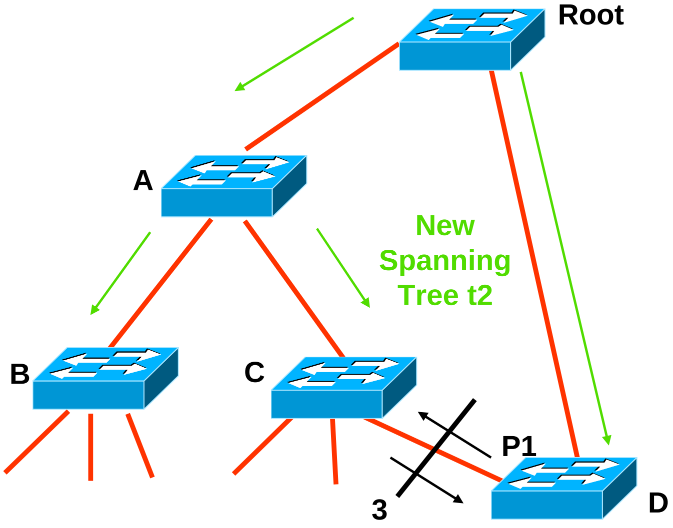


Now the link between switch A and the root is put in forwarding state.

The network below switch A is still blocking until port roles are negotiated at the next stage between switch A and switch B or A and C.

Switch B and C will enter the new spanning tree and A will put its ports in the forwarding state and the negotiations will proceed between C and D

Fast Convergence with RSTP 3



Switch C blocks its port to D because it root path costs of D are better than the root path costs of C

We have reached the final topology, which means that port P1 on D ends up blocking. It's the same final topology as for the STP example.

But we got this topology just time necessary for the new BPDU's to travel down the tree. No timer has been involved in this quick convergence.

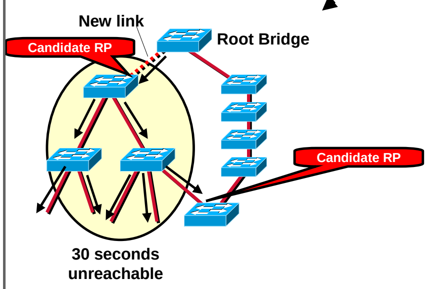
Convergence Time < 1 second

L07 - Spanning-Tree Details (v5.0)

Rapid Transition in Detail

Basic Principle

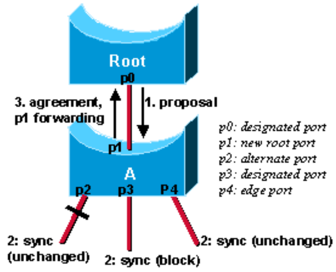
- **The new rapid STP is able to actively confirm that a port can safely transition to forwarding without relying on any timer configuration**
  - Feedback mechanism
- **Edge Ports connect hosts**
  - Cannot create bridging loops
  - Immediate transition to forwarding possible
  - No more Edge Port upon 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 in between 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.

 Spanning-Tree Details, v5.0 79

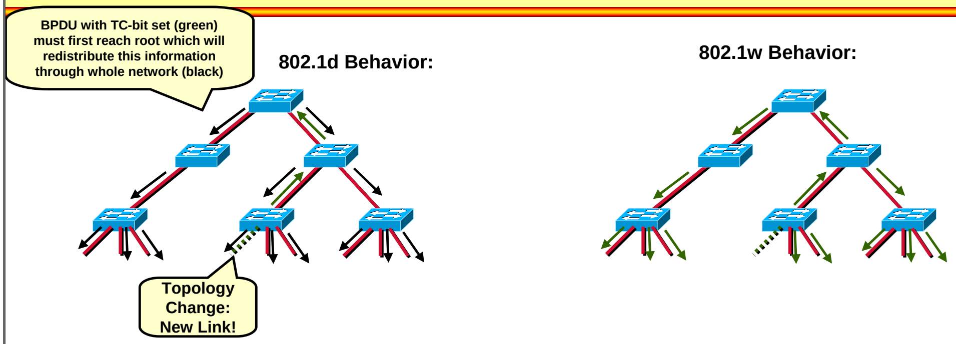
The edge port concept is already well known from Cisco's PortFast feature. Neither edge ports nor PortFast enabled ports generate topology changes when the link toggles. Unlike PortFast, an edge port that receives a BPDU immediately loses its edge port status and becomes a normal spanning tree port.

Note: Cisco's implementation maintains the PortFast keyword be used for edge port configuration, thus making the transition to RSTP simpler.

RSTP can only achieve rapid transition to forwarding on edge ports and on point-to-point links. A port operating in full-duplex will be assumed to be point-to-point, while a half-duplex port will be considered as a shared port by default.

Sync Operation: The final network topology is reached just in the time necessary for the new BPDUs to travel down the tree. No timer has been involved in this quick convergence. The only new mechanism introduced by RSTP is the acknowledgment that a switch can send on its new root port in order to authorize immediate transition to forwarding, bypassing the twice-the-forward-delay long listening and learning stages.

Topology Change

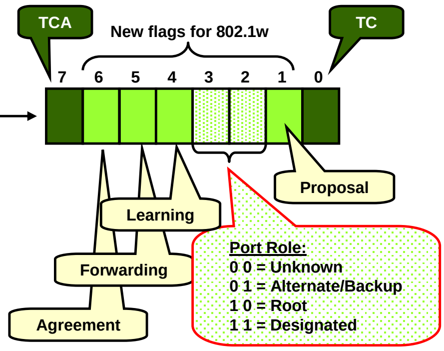


- **802.1d: When a bridge detects a topology change**
  - A TCN is sent to towards the root
  - Root sends Conf-BPDU with TC-bit downstream (for 10 BPDUs)
  - All other bridges can receive it and will reduce their bridging-table aging time to forward-delay seconds, ensuring a relatively quick flushing 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

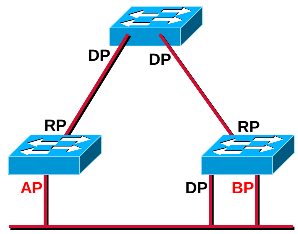
There is no need to wait for the root bridge to be notified and then maintain the topology change state for the whole network for <max age plus forward delay> seconds. In just a few seconds (a small multiple of hello times), most of the entries in the CAM tables of the entire network (VLAN) are flushed. This approach results in potentially more temporary flooding, but on the other hand it clears potential stale information that prevents rapid connectivity restitution.

RSTP Summary

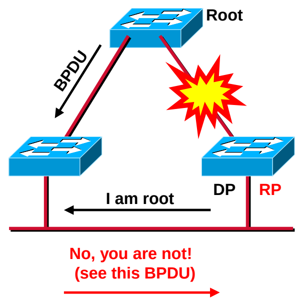
Bytes	Field
2	Protocol ID
1	Version
1	Message Type
1	Flags
8	Root ID
4	Root Path Cost
8	Bridge ID
2	Port ID
2	Message Age
2	Maximum Age = 20
2	Hello Time = 2
2	Forward Delay = 15



Backup and Alternate Ports:



Backbone Fast-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.

 Spanning-Tree Details, v5.0 81

RSTP is able to interoperate with legacy STP protocols. However, it is important to note that 802.1w's inherent fast convergence benefits are lost when interacting with legacy bridges. Each port maintains a variable defining the protocol to run on the corresponding segment. A migration delay timer of three seconds is also started when the port comes up. When this timer is running, the current (STP or RSTP) mode associated to the port is locked. As soon as the migration delay has expired, the port will adapt to the mode corresponding to the next BPDU it receives. If the port changes its operating mode as a result of receiving a BPDU, the migration delay is restarted, limiting the possible mode change frequency.

Agenda

- **Spanning Tree Protocol (STP)**
  - Introduction
  - Details
  - Convergence
  - Some more details
- **Rapid Spanning Tree Protocol (RSTP)**
- **Cisco PVST, PVST+**
- **Multiple Spanning Tree Protocol (MSTP)**

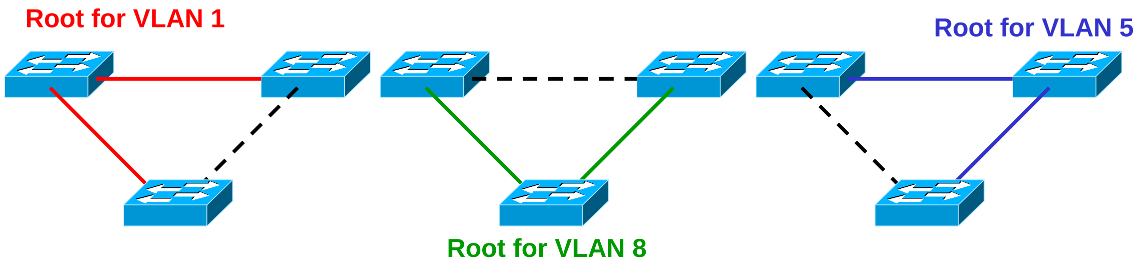
About

- **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)

VLANs (Virtual LANs) will be covered in the next chapter in more detail. Base idea: Multiplexing of several (virtual) LANs over the same LAN switching infrastructure consisting of Ethernet switches and trunk connections between Ethernet switches, A station connected to one VLAN has no access to a station on another VLAN, hence LANs are kept separated.

L07 - Spanning-Tree Details (v5.0)

Example

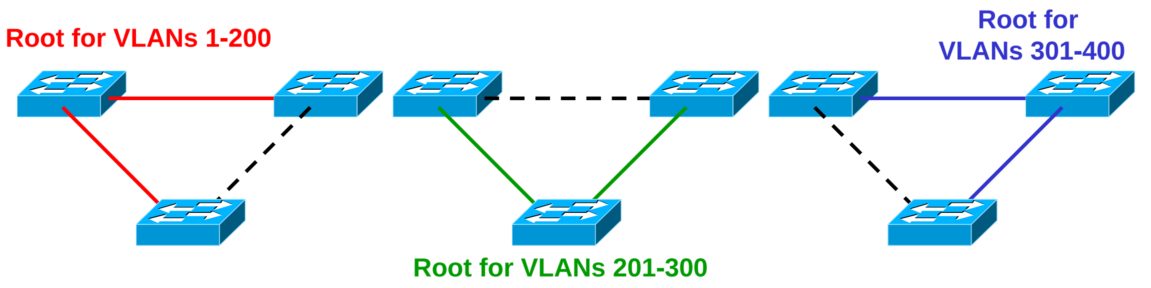


- **Remember that root bridge should realize the center of the LAN**
  - Attracts all traffic
  - Typically servers or Internet-connectivty resides there
- **Different VLANs might have different cores**
- **PVST+ allows for different topologies**
  - Admin should at least configure ideal root bridge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

 Spanning-Tree Details, v5.0 84

The picture shows a physical topology of three Ethernet switches, which are used for building three different VLANs.

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)

CST (Common Spanning-Tree) means IEEE 802.1D – 1998. ISL is VLAN trunking protocol.

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**

TLV (Type Length Value) is a technique to expand protocols by just defining what (type) is following the TLV field, how many bytes are following (length) and the type-corresponding data (value).

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

Native VLAN has number 1 in Cisco switches per default.

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…**

Agenda

- **Spanning Tree Protocol (STP)**
  - Introduction
  - Details
  - Convergence
  - Some more details
- **Rapid Spanning Tree Protocol (RSTP)**
- **Cisco PVST, PVST+**
- **Multiple Spanning Tree Protocol (MSTP)**

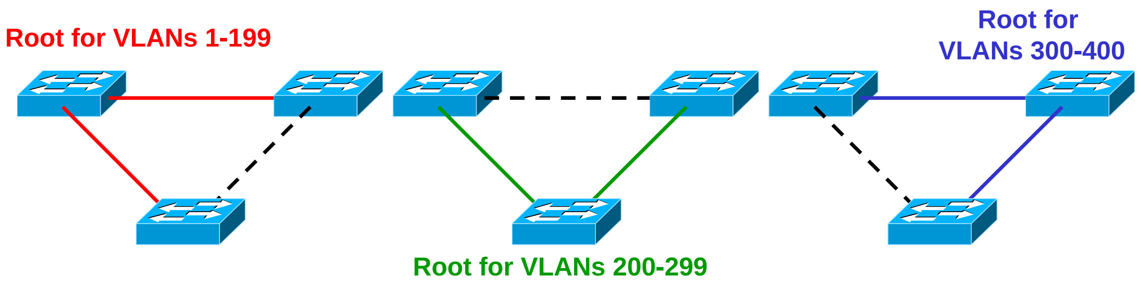
Overview

- **Also the MSTP standard contains contributions from Cisco**
  - IEEE 802.1Q-2003 (former 802.1s)
- **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**

The MSTP is defined in IEEE 802.1Q – 2003 which also defines VLAN tags and VLAN trunking

L07 - Spanning-Tree Details (v5.0)

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

 Spanning-Tree Details, v5.0 92

In this picture we need only three logical STP topologies, which are overlaid on the physical topology to use all link for VLAN traffic.

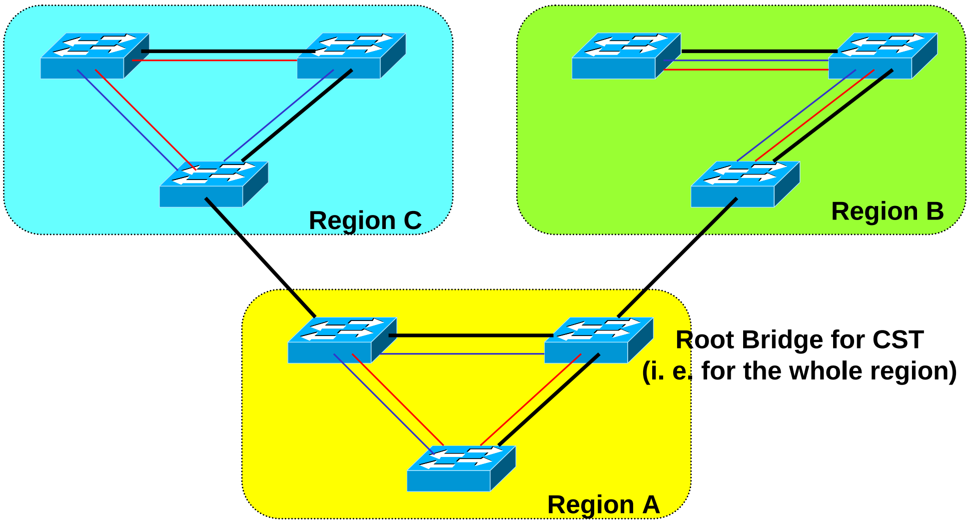
MSTP Details

- **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