

STP & RSTP

(RAPID) SPANNING TREE PROTOCOL

**DESCRIPTION OF STP AND RSTP,
PROTOCOLS FOR LOOP FREE
LAN TOPOLOGIES**

Peter R. Egli
peteregli.net

Contents

1. Goal of STP: Loop-free topology for Ethernet networks
2. STP standards overview
3. IEEE 802.1D STP protocol
4. IEEE 802.1w RSTP Rapid STP
5. IEEE 802.1Q CST Common Spanning Tree
6. Cisco PVST+ and PVRST+
7. IEEE 802.1s MST Multiple Spanning Tree Protocol
8. STP Pros and Cons

1. Goal of STP: Loop-free topology for Ethernet networks

Ethernet bridges (or switches) must forward unknown unicast and broadcast Ethernet frames to all physical ports.

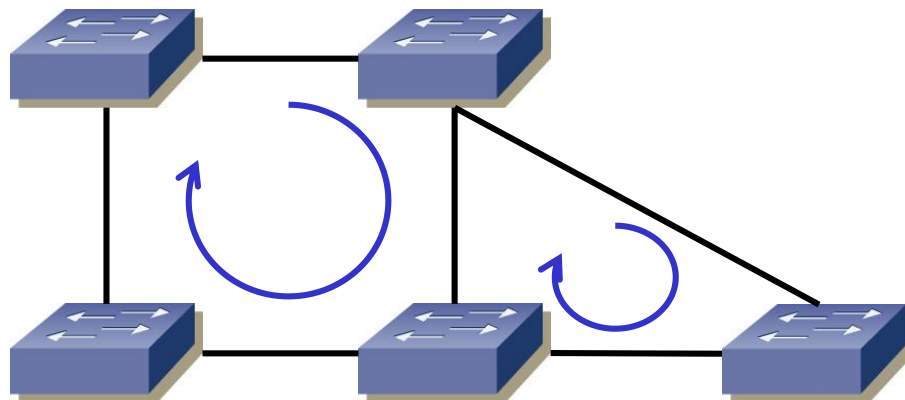
Therefore Ethernet networks require a **loop-free topology**, otherwise more and more broadcast and unknown unicast frames would swamp the network (creation of frame duplicates resulting in a broadcast storm).

Unknown unicast frame:

Frame with a destination Ethernet address that is not yet known by the receiving bridge.

Broadcast frame:

Ethernet frame with a broadcast destination target Ethernet address, e.g. for protocols such as ARP or BOOTP / DHCP.



Broadcast Ethernet frames and unknown unicast frames circle forever in an Ethernet network with loops.

2. STP standards overview:

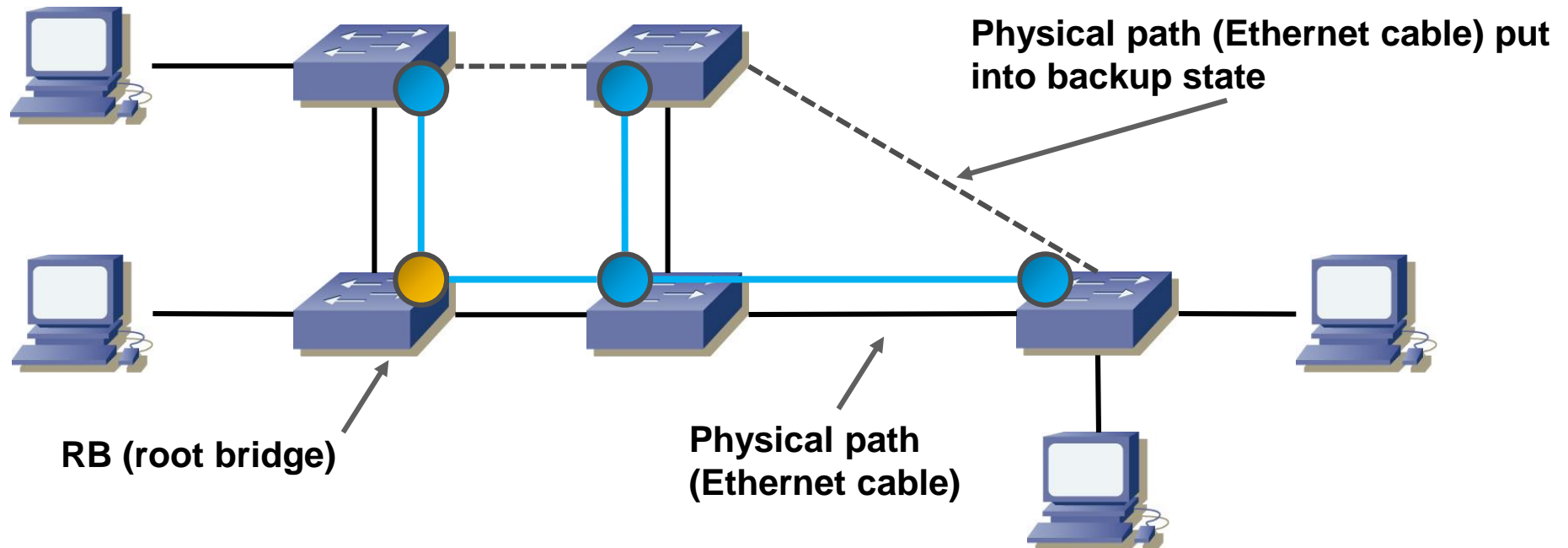
A number of different STP ‘standards’ and protocols evolved over time.

Standard	Description	Abbreviation
IEEE 802.1D	Spanning Tree Protocol <ul style="list-style-type: none">• Loop prevention.• Automatic reconfiguration of tree in case of topology changes (e.g. phys. link failure)• Slow convergence (up to 50 seconds)• Obsoleted by RSTP (IEEE 802.1w)	STP
IEEE 802.1w	Rapid Spanning Tree Protocol <ul style="list-style-type: none">• Improved STP with faster convergence• Backward compatible with STP	RSTP
IEEE 802.1Q	Virtual LAN <ul style="list-style-type: none">• Defines that 1 common spanning tree (CST) shall be used for all VLANs	CST
Cisco Proprietary	Per VLAN Spanning Tree <ul style="list-style-type: none">• 1 STP instance per VLAN• PVST+ is an improved variant of PVST	PVST PVST+
Cisco proprietary	Per VLAN Rapid Spanning Tree <ul style="list-style-type: none">• 1 RSTP instance per VLAN	PVRST+ or R-PVST+ or RPVST+
IEEE 802.1s	Multiple Spanning Tree Protocol or Multiple Instance Spanning Tree Protocol <ul style="list-style-type: none">• Multiple instances of VLAN mapped to 1 STP (tradeoff between IEEE 802.1Q CST and PVST)• Originally defined in 802.1s, then incorporated into IEEE 802.1Q-2005	MSTP or MISTP

3. IEEE 802.1D STP protocol (1/32):

3.1. Loop prevention:

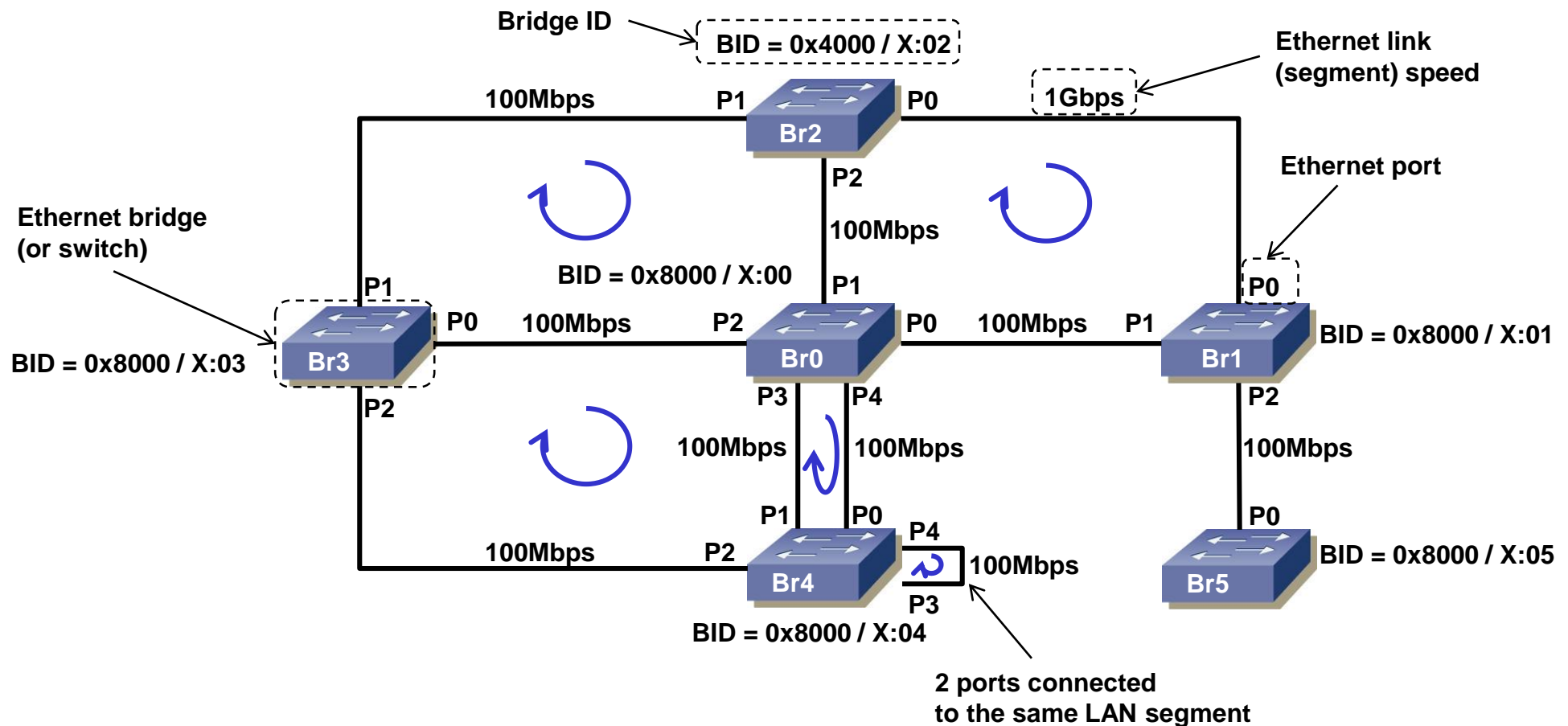
- STP provides loop prevention in bridged networks by establishing a loop-free tree of forwarding paths between any two bridges in a network with multiple physical paths.
- If a link fails, STP automatically establishes a new loop-free topology.
- STP does not necessarily converge to a tree with a minimal number of active links between segments because STP uses link speed as a preference (hop count is considered indirectly through root path count).



3. IEEE 802.1D STP protocol (2/32):

3.2. STP model:

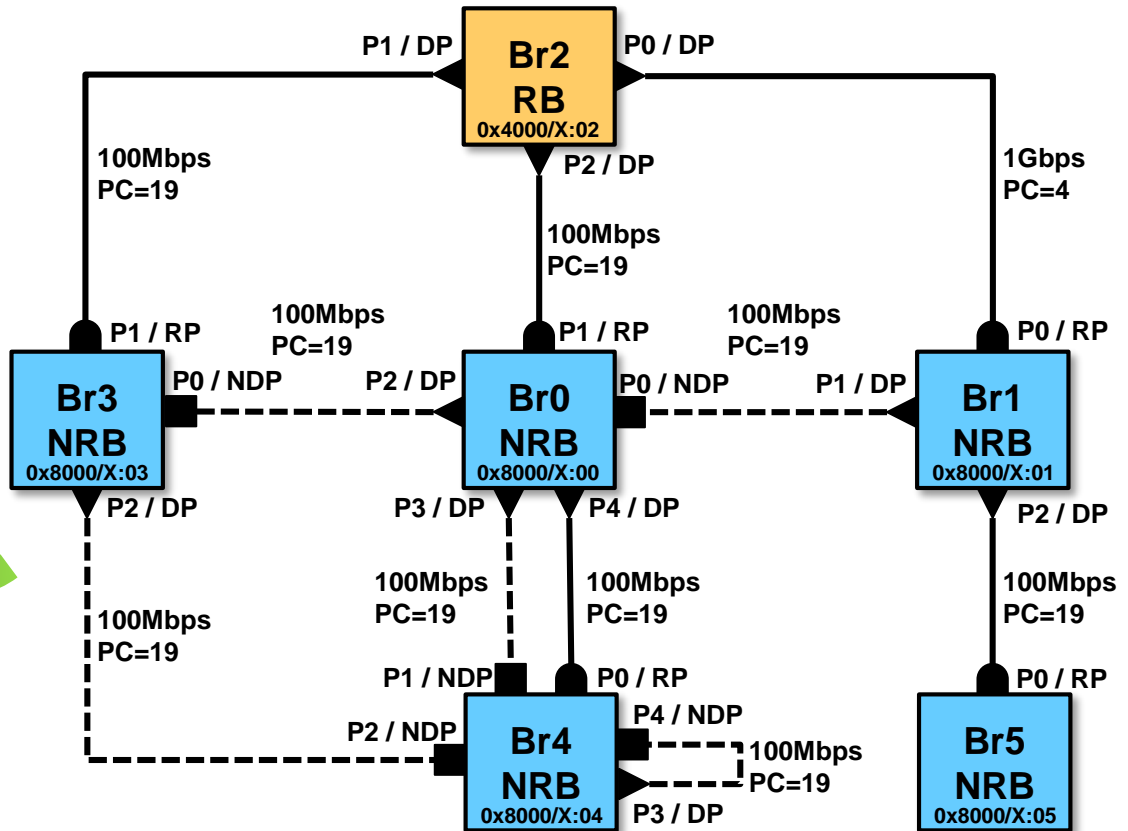
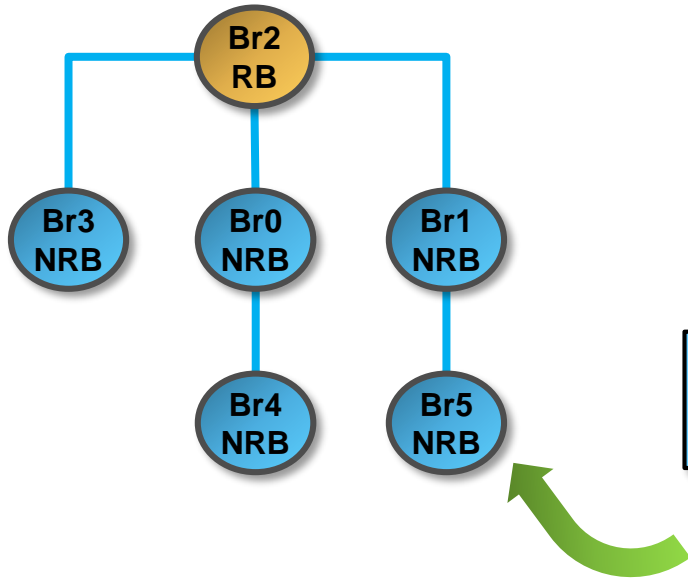
The following example network with 5 physical loops is used throughout the following chapters. The STP terms are explained and exemplified with this example network.



3. IEEE 802.1D STP protocol (3/32):

3.2. STP model:

The example network from the previous page is translated into the following loop-free and stable STP topology.



- Key:**
- P0 / RP
 - RP
 - DP
 - NDP
 -
 - PC
 - RB
 - NRB
 - Brx
 - 0x8000/X:03
 -
 -
- Port number and port role (RP, DP, NDP)
 - Port role = Root Port
 - ▲ Port role = Designated Port
 - Port role = Non Designated Port
 - No port role assigned yet
 - Path cost
 - Root bridge
 - Non-root bridge
 - Bridge (=switch)
 - Bridge ID (only last byte of MAC address shown)
 - Active link
 - Backup link (not used for forwarding)

3. IEEE 802.1D STP protocol (4/32):

3.2. STP model:

The different terms in the STP model are explained here. For the sake of brevity, acronyms are used throughout this document once they are defined.

Bridge:

A bridge connects to one or more LAN segments.

Today's networks are predominantly switch based. With regard to STP, the function of a switch is equal to a bridge. Therefore the terms bridge and switch are used interchangeably in this document.

Root bridge (RB):

The root bridge is the one bridge that provides an interconnection point for all segments. Every bridge in a LAN has a path to the root and thus any segment is reachable from any other segment through a path through the root bridge.

STP is able to automatically select the root bridge. However, STP's choice may be suboptimal. Therefore administrative action may be necessary to force a specific (powerful) bridge to become RB (see below).

Non-root bridge (NRB):

Any bridge that is not the RB is called non-root bridge.

3. IEEE 802.1D STP protocol (5/32):

3.2. STP model:

Bridge Protocol Data Unit (BPDU) (1/2):

BPDUs are used by bridges to exchange topology information. BPDUs are sent to the STP multicast target MAC address 01:80:C2:00:00:00. There exist 2 types of BPDUs:

A. Configuration BPDUs:

→ Used for exchanging topology information and establishing a loop-free topology.

	Bytes	Description
Protocol Identifier	2	Field that identifies frame as being STP (encoded as 0x0000).
Protocol Version Identifier	1	Protocol version identifier (encoded as 0x00).
BPDU Type	1	BPDU type (0x00= Configuration BPDU).
Flags	1	Topology change (bit 7) and topology change ack (bit 0).
Root Identifier	8	Root bridge identifier (root ID).
Root Path Cost	4	Path cost to root bridge through bridge and port that sent this BPDU.
Bridge Identifier	8	Bridge identifier of bridge that sent this BPDU (Bridge ID).
Port Identifier	2	Port identifier of port through which this BPDU is sent.
Message Age	2	Used for aging out old information (message age is incremented on receipt and discarded if > Max. Age).
Maximum Age	2	If message age exceeds max. age (default 20s), the information is discarded.
Hello Time	2	Interval between sending of configuration BPDUs.
Forward Delay	2	Delay for RPs and DPs to transition to the state <i>Forwarding</i> .

3. IEEE 802.1D STP protocol (6/32):

3.2. STP model:

Bridge Protocol Data Unit (BPDU) (2/2):

B. Topology Change Notification BPDU (TCN BPDU):

→ Used by NRBs to notify the RB of a topology change (trigger for re-establishing a loop-free topology).

	Bytes	Description
Protocol Identifier	2	Field that identifies packet as being STP (encoded as 0x0000).
Protocol Version Identifier	1	Protocol version identifier (encoded as 0x00).
BPDU Type	1	BPDU type (0x80=Topology Change BPDU).

3. IEEE 802.1D STP protocol (7/32):

3.2. STP model:

Segment:

In the old days of Ethernet, a segment used to be a multi-drop cable (thick Ethernet: 10Base5, thin Ethernet: 10Base2). Multiple bridge ports were connected to a single cable.

Since the introduction of shielded twisted pair and optical cabling, a segment is simply a point-to-point cable between 2 Ethernet ports of 2 bridges (1 RP and 1 DP).

Port role:

A bridge port has either of the following roles:

Root port (RP), designated port (DP) or non-designated port (NDP = role of ports that are neither RP nor DP and thus are blocked).

Port Role	Send BPDUs	Recieve BPDUs	Forward frames
NDP (blocked)	No	Yes	No
DP	Yes	Yes	Yes
RP	No	Yes	Yes

3. IEEE 802.1D STP protocol (8/32):

3.2. STP model:

Root port (RP):

The root port of a bridge is the port that leads towards the root bridge (kind of an «uplink»), i.e. the one port of a bridge that has the lowest path cost towards the RB.

Every NRB has exactly 1 RP.

The RB does not have RPs as every port is a DP (see below).

Designated port (DP):

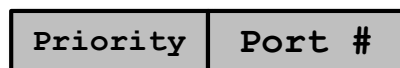
Every LAN segment needs to have 1 designated port. The bridge with the DP on a segment picks up frames sent to the segment and forwards the frames through its RP towards the RB or to any other bridge port as defined in the MAC address learning table.

The DP guarantees that every segment is connected to the STP tree topology (no islands of isolated segments without connectivity to the tree).

RB: All ports are DPs.

Port ID:

The port ID is used to determine the root port (RP). It consists of a configurable 1 byte priority value and a port number that is unique per bridge.



Port ID

3. IEEE 802.1D STP protocol (9/32):

3.2. STP model:

Path cost (PC):

In order to determine the best topology with regard to forwarding speed, STP uses a concept with path costs (there may be a better topology, though, depending on the selection of the RB). Every segment is given a path cost as per the following table (from IEEE 802.1D:1998):

Link speed	Path Cost Recommended Value	Path Cost Recommended Range	Range
4 Mbps	250	100-1000	1-65535
10Mbps	100	50-600	1-65535
16Mbps	62	40-400	1-65535
100Mbps	19	10-60	1-65535
1Gbps	4	3-10	1-65535
10Gbps	2	1-5	1-65535

Path cost ranges allow fine-tuning by configuring different values on bridge ports with equal link speed, thus preferring certain ports over others.

The root path cost value is encoded as 4 bytes in configuration BPDUs, but some bridges only support 16 bit values.

Path cost is calculated by adding the individual path cost values of each segment of the path. The table defines exponentially growing values in order to favor higher speed links over slower speed links. E.g. two 1Gbps links in sequence have a lower path cost ($4+4=8$) than one 100Mbps link (19) and thus are favored by the STP algorithm.

3. IEEE 802.1D STP protocol (10/32):

3.2. STP model:

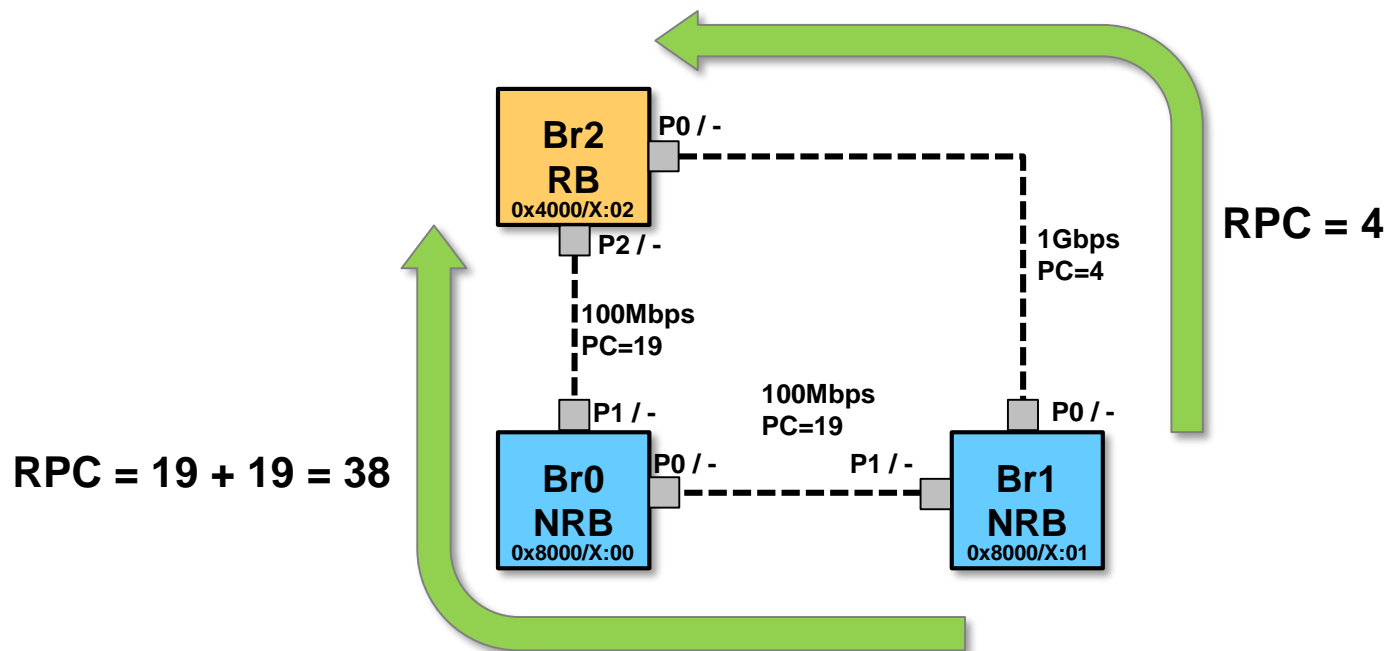
Root path cost (RPC):

Root path cost (RPC) of a bridge is the cost of the path with the lowest sum of individual segment costs towards the RB.

In the example network, Br1 has 2 physical paths towards the RB (Br2).

The single link path (1Gbps) has PC=4 while the path through Br0 has PC=19+19=38.

Br1 selects the path with the lower RPC for forwarding frames towards the RB.



3. IEEE 802.1D STP protocol (11/32):

3.2. STP model:

Designated path cost (DPC):

DPC is the path cost of designated ports.

By definition all ports of the RB have DPC=0 and thus are DP of their segment.

The DPC of designated ports of a non-root bridge is the RPC of the bridge (DPC = RPC).

The port with the lowest DPC (=RPC) on a segment is selected as DP. In case there are 2 ports with equal DPC, the port whose bridge has a lower bridge ID is selected as DP.

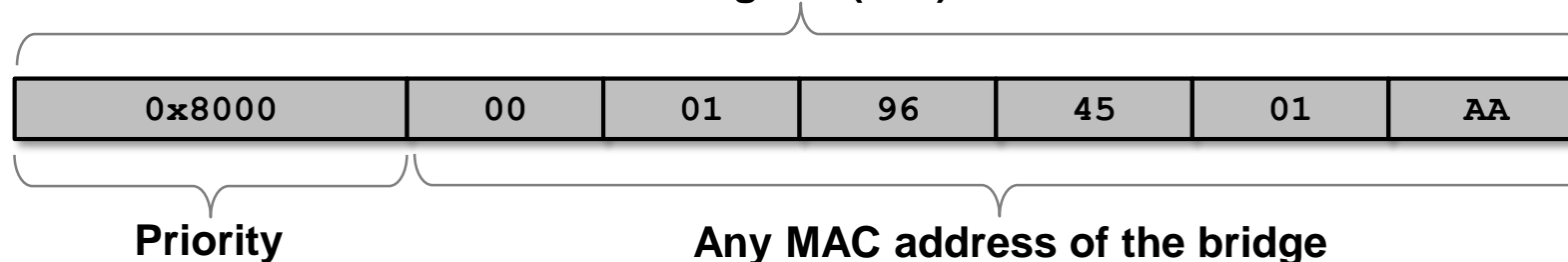
Bridge ID (BID):

The bridge ID is used for the selection of the RB and DPs.

The bridge ID is composed of a priority value (2 bytes, default 0x8000=32768) and one of the MAC addresses of the bridge (6 bytes). The MAC address serves as a tie breaker in case of equal priorities (which is usually the case with default priorities on all bridges).

Configuring the priority to a lower value allows the network administrator to force a specific bridge to become RB. Example BID: 0x8000 / 00:01:96:45:01:AA

Bridge ID (BID)

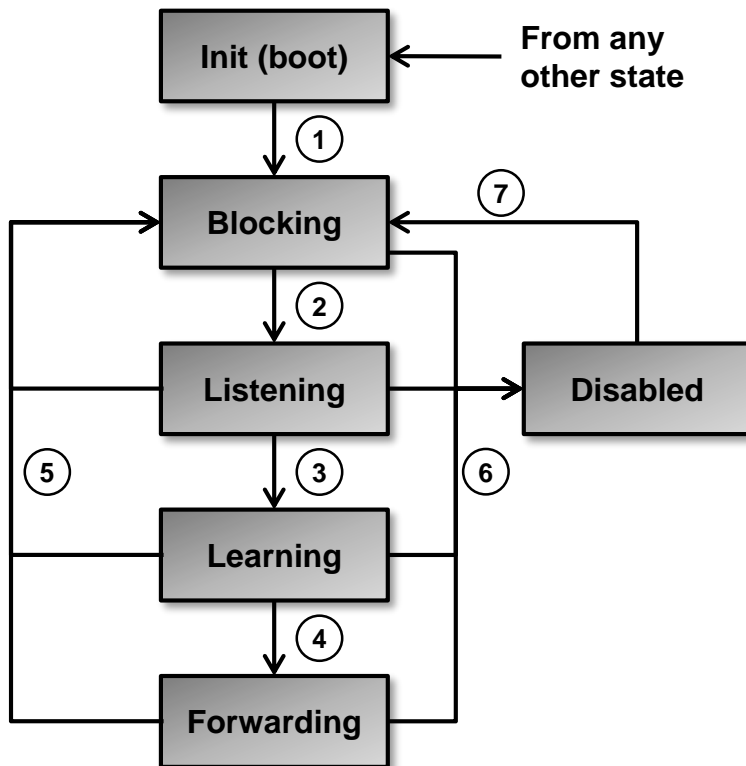


3. IEEE 802.1D STP protocol (12/32):

3.2. STP model:

Port state:

Ports states and transitions for STP are defined by the following state transition diagram.



1 (Init → Blocking):

A port is initialized and automatically transitions to *Blocking*.

2 (Blocking → Listening):

When *MaxAge* timer expires (up to 20 seconds), the port transitions to the state *Listening*.

3 (Listening → Learning):

When *ForwardDelay* timer expires (up to 15 seconds), a port transitions to the state *Learning*.

4 (Learning → Forwarding):

A port remains in *Learning* state (*ForwardDelay*, up to 15 seconds) until transitioning to the *Forwarding* state.

5 (any state → Blocking):

A topology change brings the port back to *Blocking* state.

6 (any state → Disabled):

A port is disabled by administrative action.

7 (Disabled → Blocking):

A port is enabled again by administrative action.

3. IEEE 802.1D STP protocol (13/32):

3.2. STP model:

Port state:

Ports process BPDUs and forward Ethernet frames as per the following table.

State	Description	Process BPDUs	Forward Ethernet frames	Learn MAC addresses
Init	Initialization of port (bootstrap). Actually not an STP port state.	No	No (discard frames)	No
Disabled	Administrative state. If disabled (shut down), a port does not participate in STP operation.	No	No (discard frames)	No
Blocking	The port does not forward Ethernet frames (discards them) and does not learn MAC addresses (backup state).	Yes (receive and process BPDUs only)	No (discard frames)	No
Listening	Computation of loop-free topology is carried out in this state and the port is assigned its role (RP, DP, NDP).	Yes (send and receive BPDUs)	No (discard frames)	No
Learning	Additional state to delay forwarding of Ethernet frames to avoid flooding the network.	Yes	No (discard frames)	Yes (populate MAC address table)
Forwarding	Normal operation of forwarding Ethernet frames (user traffic).	Yes	Yes	Yes

3. IEEE 802.1D STP protocol (14/32):

3.2. STP model:

Port state versus port role:

The following table shows which port states are possible for which port role.

N.B.: The port role “No role assigned” is used to denote the situation where a port has no assigned role yet.

Port role \ Port state	RP	DP	NDP (blocked)	No role assigned (yet)
Disabled	No	No	No	Yes
Init	No	No	No	Yes
Blocking	No	No	No	Yes
Listening	Yes	Yes	Yes	No
Learning	Yes	Yes	No	No
Forwarding	Yes	Yes	No	No

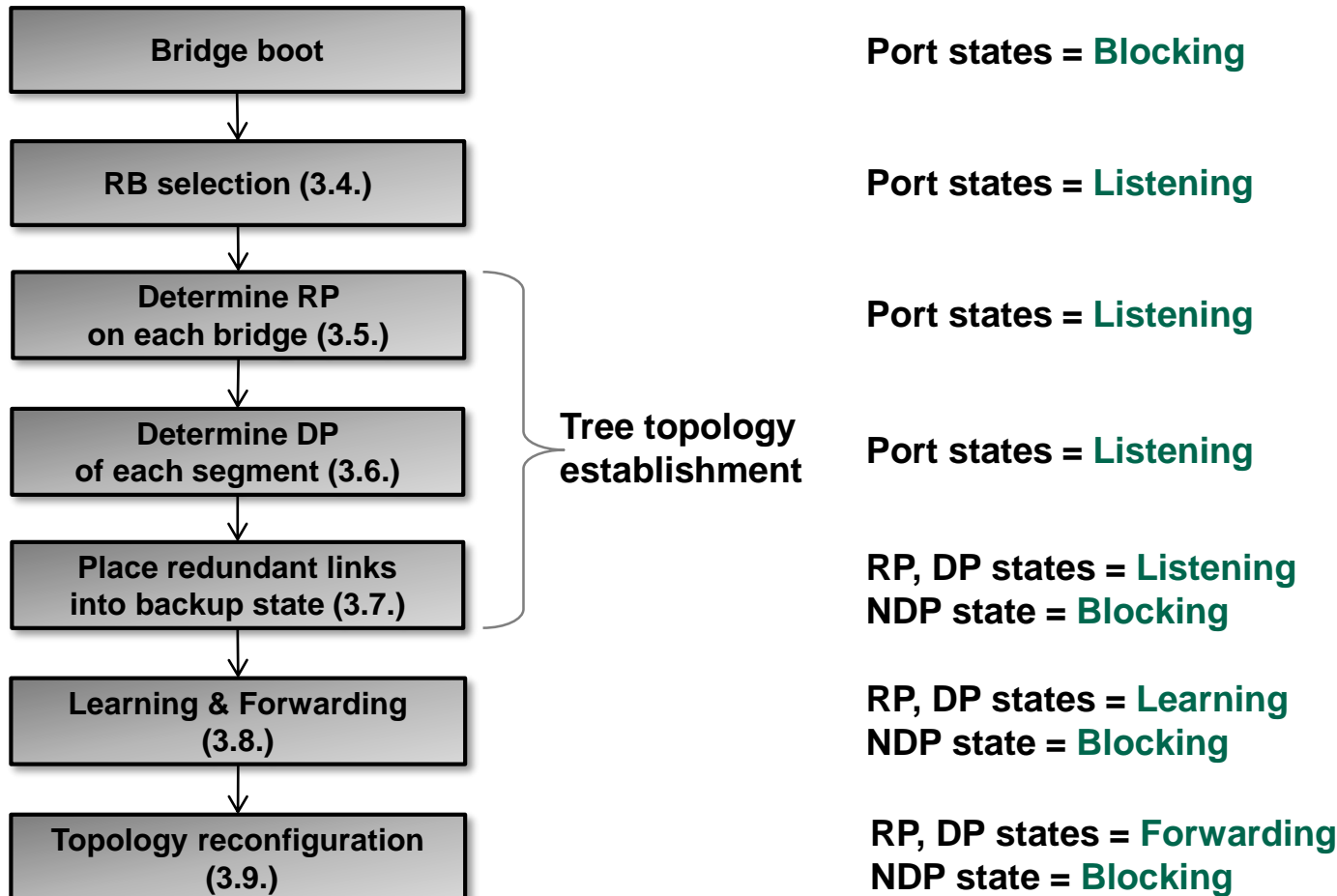
← The port role is assigned in the **Listening** state.

3. IEEE 802.1D STP protocol (15/32):

3.3. STP phases:

STP operation in a network may be split into different phases as shown below.

In reality, the ports may be in different transient states (**Blocking**, **Listening**, **Learning**) until reaching a stable state (**Forwarding** or **Blocking**).



3. IEEE 802.1D STP protocol (16/32):

3.4. Root bridge (RB) selection:

One bridge is elected as Root Bridge RB, i.e. becomes root of the spanning tree.

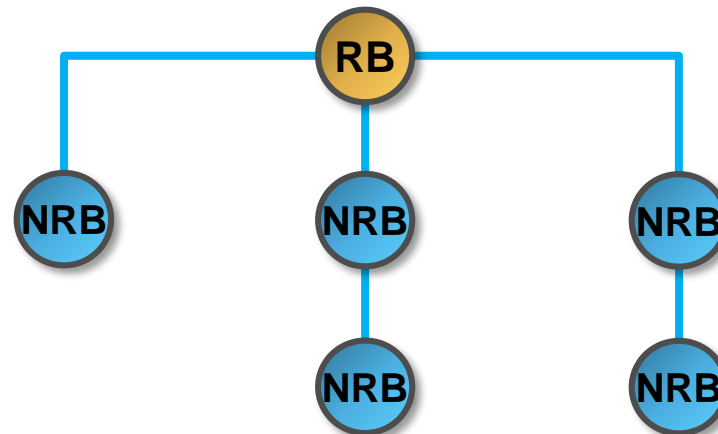
By definition, the bridge with the lowest bridge identifier (BID) in the network is elected as root bridge (STP tree root).

2 BIDs (BID1 and BID2) are compared as follows:

1. If either of the BIDs has a lower priority value, this bridge wins (has a chance to become RB).

2. If 2 BIDs have the same priority value, the bridge with the lower MAC address wins.

The RB is at the center of the spanning tree topology. Any segment is reachable from any other segment through the RB. As such the RB serves as an inter-connection point for all LAN segments.



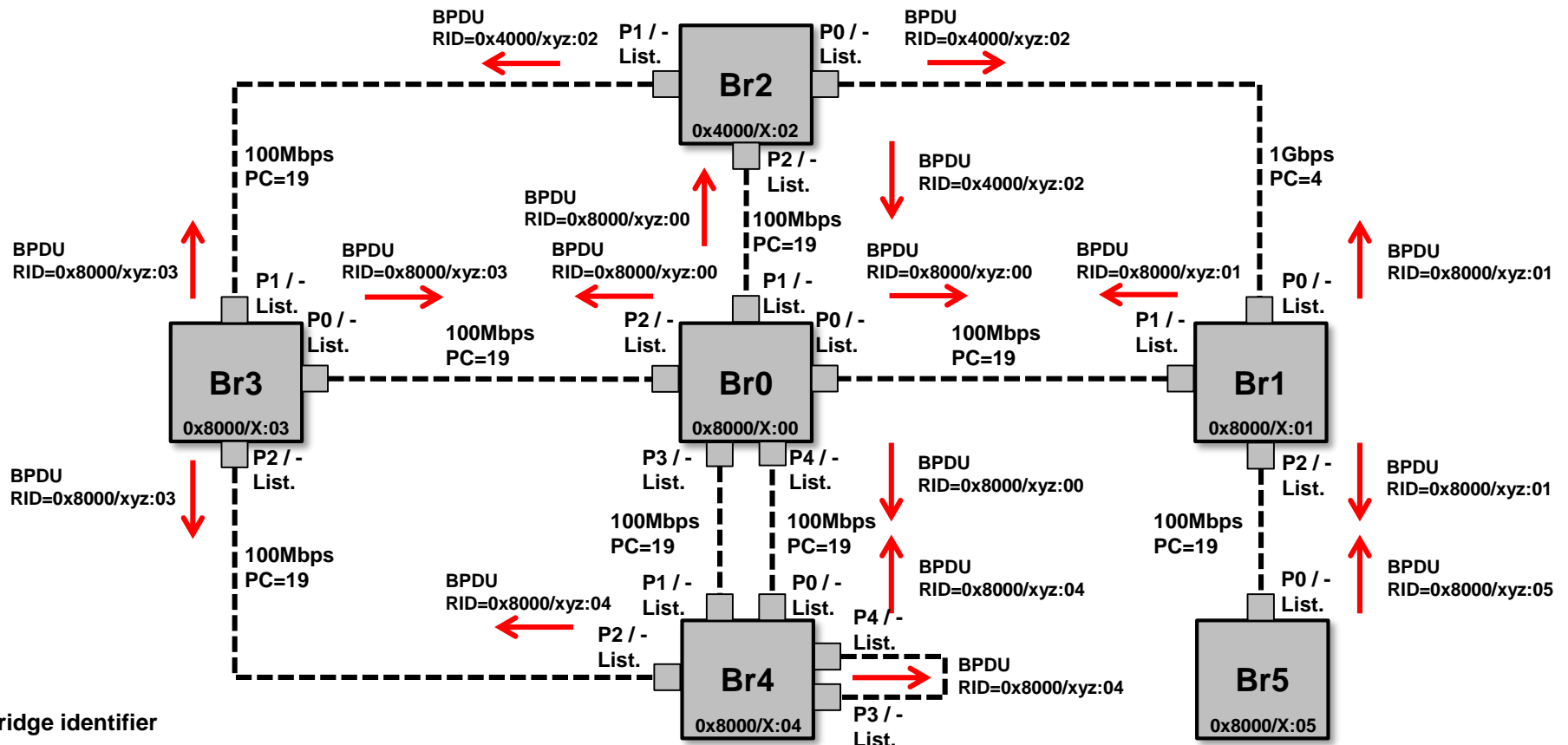
3. IEEE 802.1D STP protocol (17/32):

3.4. Root bridge (RB) selection:

All bridges send configuration BPDUs (assume they are the RB) with root ID = their own BID.

To force a specific bridge to become the RB (e.g. Br1), the administrator sets the priority of Br1 to a higher priority (lower value 0x4000).

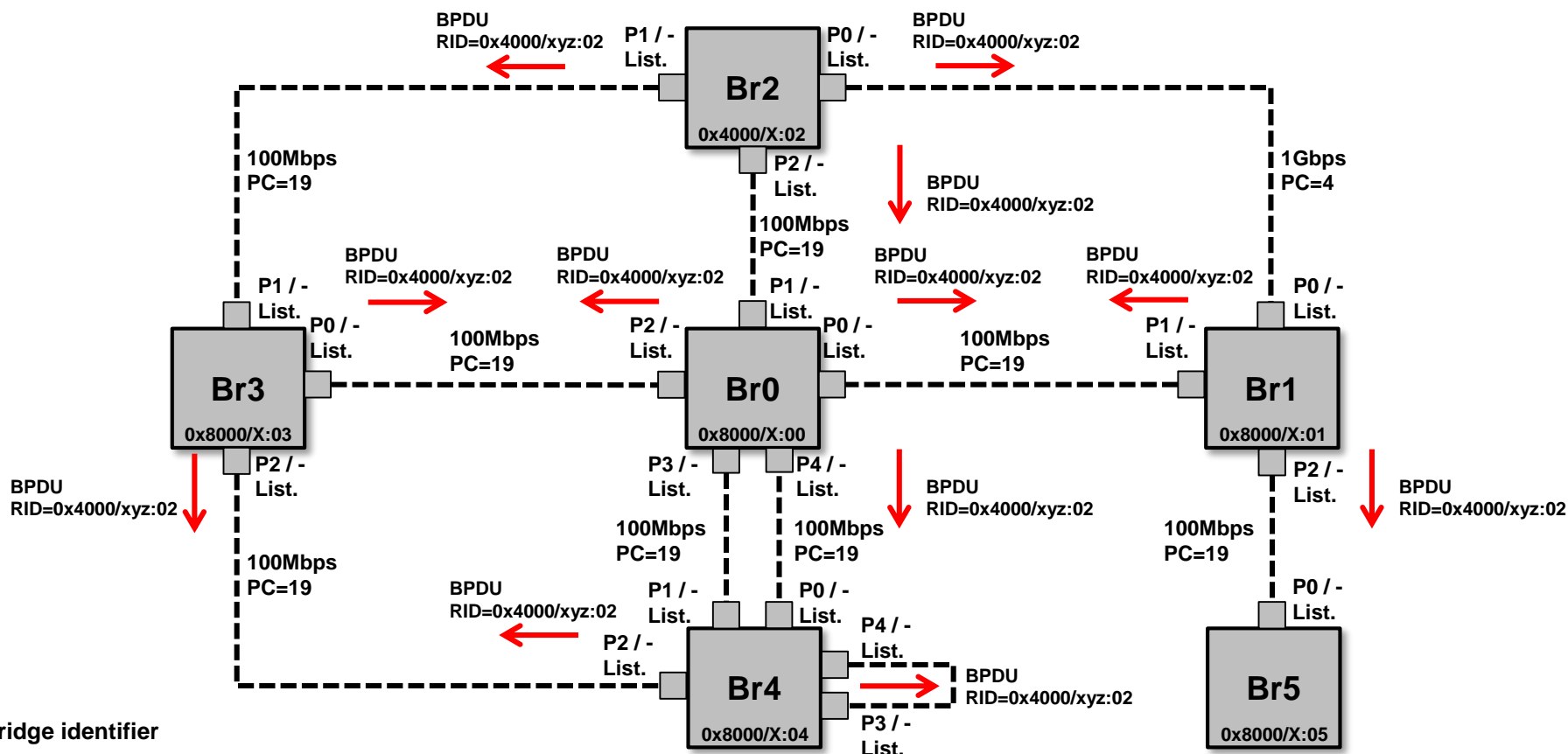
The interval between sending configuration BPDUs is defined by *HelloTime* (default 2 seconds).



3. IEEE 802.1D STP protocol (18/32):

3.4. Root bridge (RB) selection:

All bridges compare the received BPDU's root ID with their own BID. If the received root ID is lower, they stop sending own BPDUs but start forwarding received BPDUs with the better (=lower) root ID over all interfaces (except the one over which the BPDU was received).

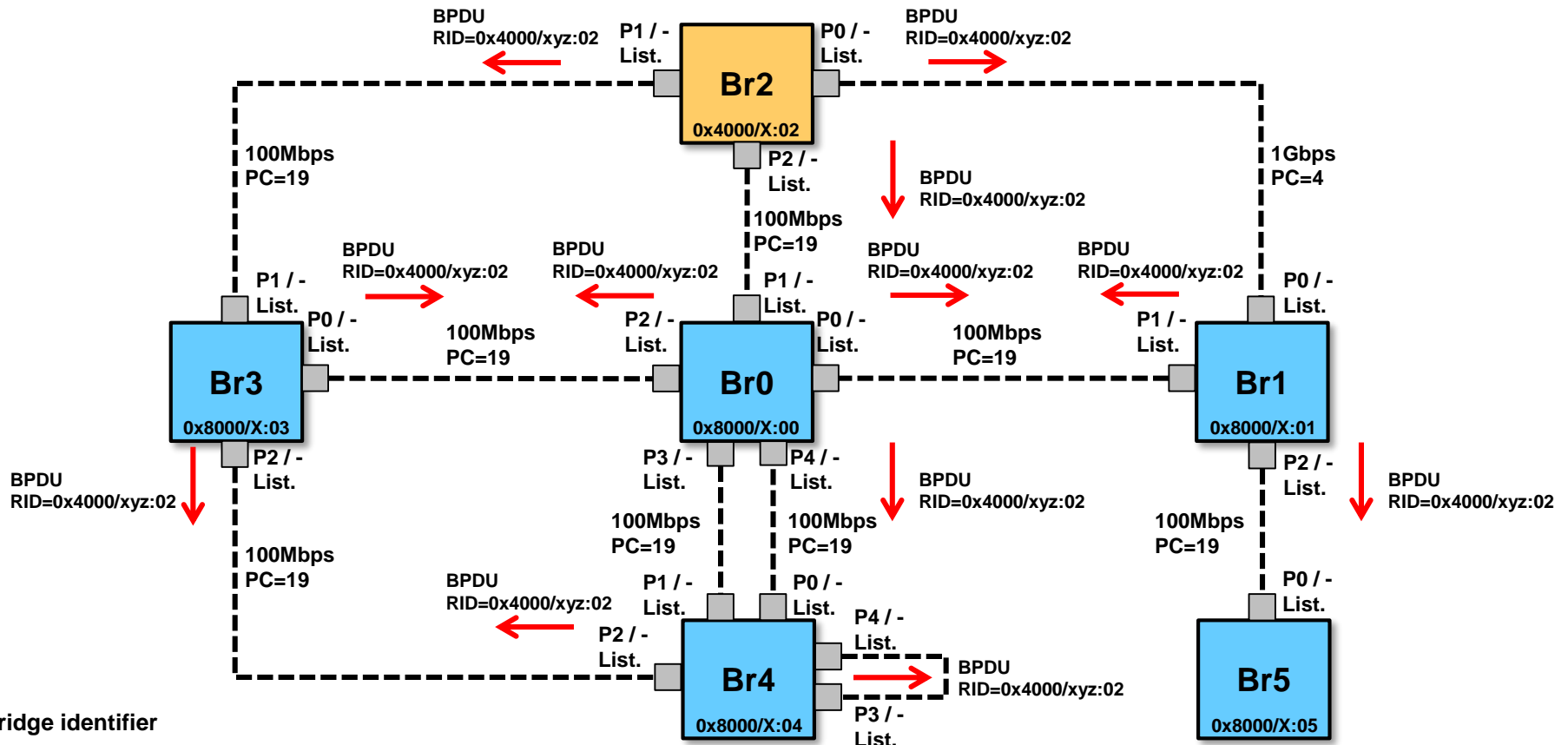


3. IEEE 802.1D STP protocol (19/32):

3.4. Root bridge (RB) selection:

Eventually, Br2's BPDUs will reach all corners of the LAN informing all bridges that it has the lowest root ID and thus becomes root bridge (RB).

Br2 continues sending BPDUs with interval *HelloTime*.

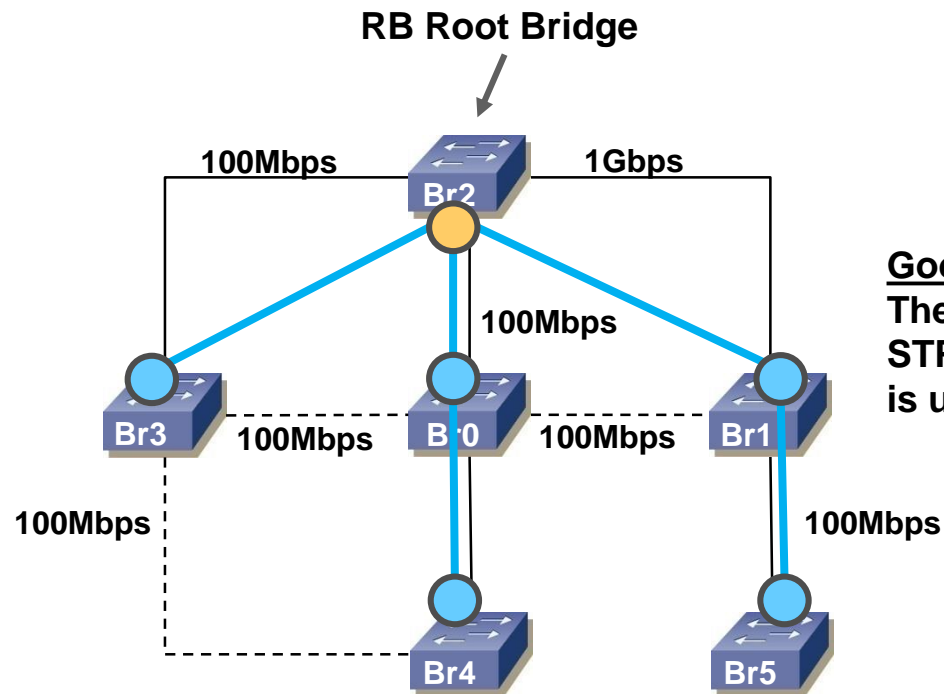


3. IEEE 802.1D STP protocol (20/32):

3.4. Root bridge (RB) selection:

The root bridge should be a powerful device (more traffic „routed“ through root bridge) and be positioned at the center of the network.

STP will find a loop-free topology in any network, but if an „edge“ bridge is chosen as RB, then the network will only have sub-optimal performance.



Good RB choice:

The RB (Br0) is at the center of the STP topology. The fast link with 1Gbps is used for frame forwarding.

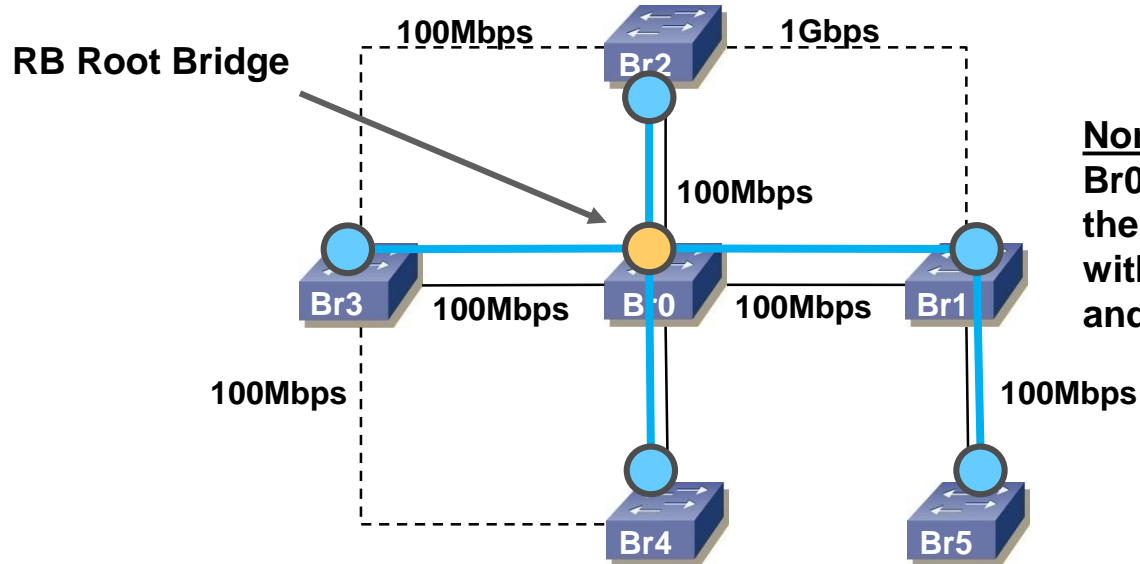
3. IEEE 802.1D STP protocol (21/32):

3.4. Root bridge (RB) selection:

Good selection of a RB may require administrative intervention.

Configuring an STP priority allows controlling which bridge becomes the root bridge in the network.

In the example below, the administrator would have to configure the priority value of Br2 so as to force STP to select Br2 as root bridge instead of Br0.



Non-optimal RB choice:

Br0 is elected as RB because it has the lowest BID. The fast link with 1Gbps is put into backup state and thus not used.

3. IEEE 802.1D STP protocol (22/32):

3.5. Determine root port (RP) on each bridge:

In this phase, each bridge except the RB determines its RP by calculating and comparing the RPC on its different ports. The port with the lowest RPC becomes RP.

In case of equal RPC, the neighbor BID and port ID are taken into consideration.

The root port is determined as per the following steps:

1. $RPC = \min. RPC_i$ of any port → Port with least root path cost becomes RP.
2. Equal RPC → Port to neighbor bridge with lowest BID becomes RP.
3. Equal RPC and 2 links to the same neighbor bridge → Port with lower port ID becomes RP.

3. IEEE 802.1D STP protocol (23/32):

3.5. Determine root port (RP) on each bridge:

Example lowest RPC determines RP (Br1):

Br2 sends configuration BPDUs with Root Path Cost field value = 0 (Br2 itself is RB, thus path cost to root = itself is 0).

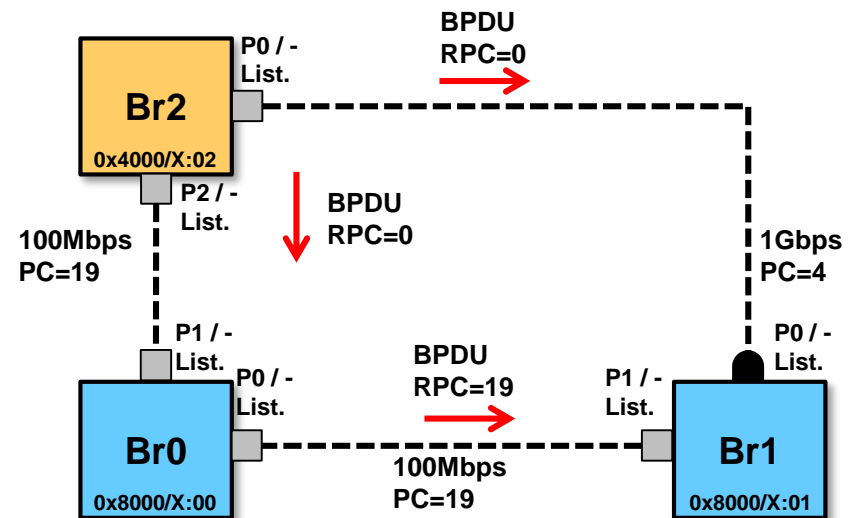
Br0 receives this BPDU, adds the segments cost to RPC and forwards the BPDU.

Br1 receives a BPDU from Br2 and Br0. Br1 adds the segment's path cost to the received BPDUs RPC.

Calculated RPC on Br1/P0 = 4.

Calculated RPC on Br1/P1 = 19 + 19 = 38.

P0 on Br1 has the lowest RPC, thus it becomes RP.



3. IEEE 802.1D STP protocol (24/32):

3.5. Determine root port (RP) on each bridge:

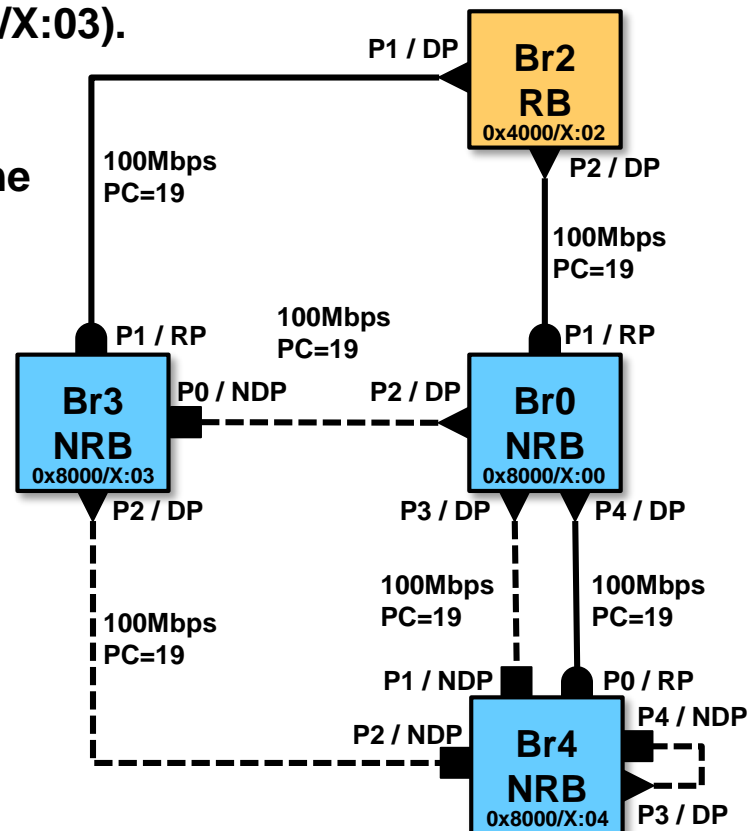
Example equal RPC (Br4):

RPC may not suffice to determine the root port. Multiple ports may have identical RPC.

Br4 has 3 ports with $RPC=19+19=38$ (P0, P1, P2).

Br4 favors P0 and P1 over P2 because the neighbor BID of P0 and P1 is $0x8000/X:00$ and thus better than the neighbor BID through P2 ($0x8000/X:03$).

Finally, Br4 selects P0 as RP because it has a better port ID than P1 (both ports have the same priority, but P0 has the lower port number).



3. IEEE 802.1D STP protocol (25/32):

3.6. Determine designated port (DP) of each segment:

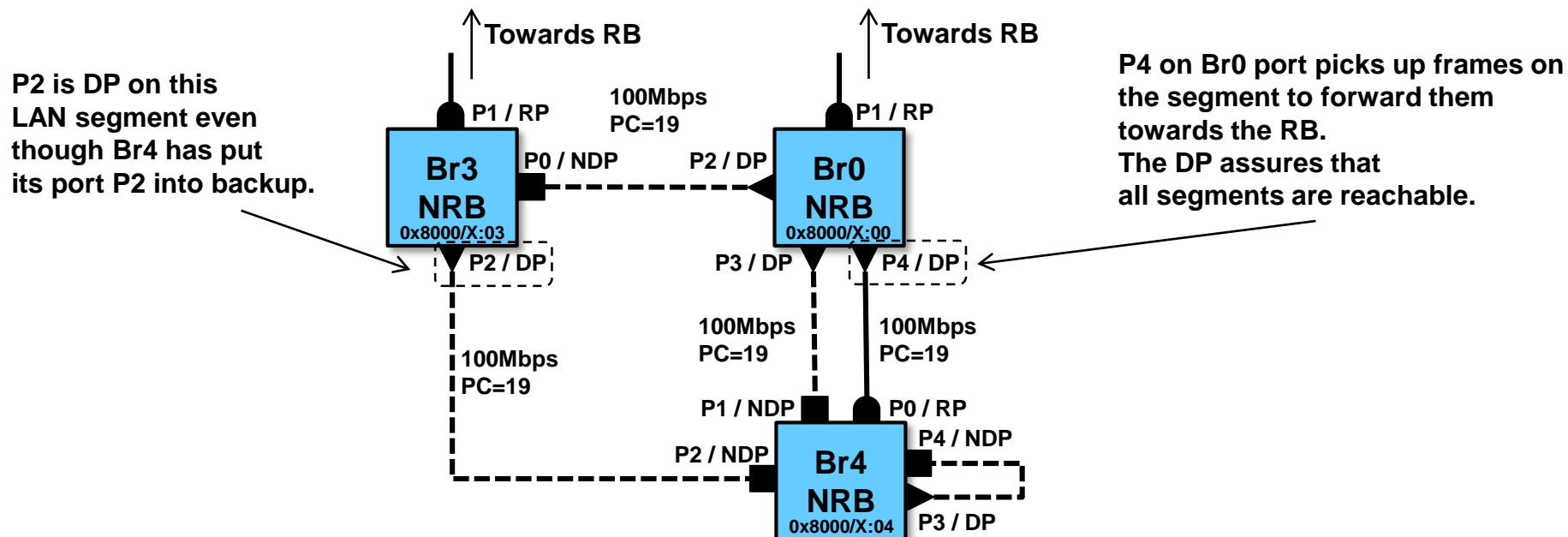
One port is designated to pick up frames (frames with target address reachable through RB) on that segment and send them to the RP.

By selecting RPC as DPC value, the bridge with the best RPC on a LAN segment becomes designated bridge.

On a RB, all ports by definition have DPC=0 and thus are DP of the segment.

Why is P2 on Br3 a DP even though the link on P2 is in backup state?

Br3 does not know that Br4 has placed its port P2 into backup state. Br3 must assume that at least one other bridge on the LAN segment has an RP on that segment.



3. IEEE 802.1D STP protocol (26/32):

3.6. Determine designated port (DP) of each segment:

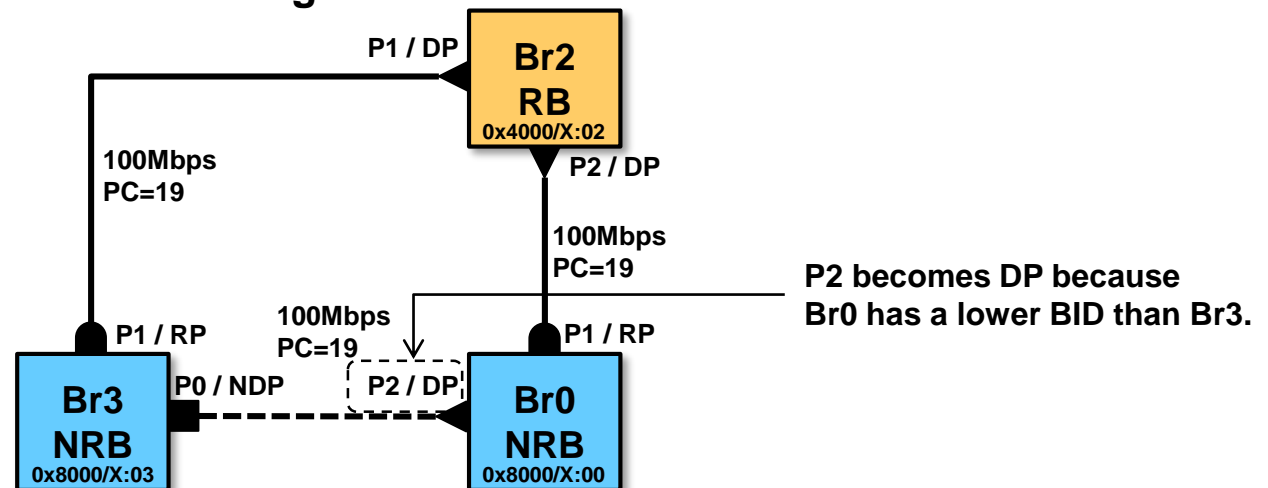
The designated port is determined as per the following steps:

1. DPC on segment = RPC of bridge. Port with lowest DPC (=RPC of bridge) is selected as DP.
2. If equal DPC → lowest BID wins.
3. If equal DPC and equal BID → lowest port ID wins.

Example equal RPC / DPC (Br0 and Br3):

Both Br0 and Br3 have RPC = 19.

Because Br0 has a lower and thus better BID (0x8000/X:00) than Br3, it becomes designated bridge and P2 designated port on the LAN segment towards Br3.



3. IEEE 802.1D STP protocol (27/32):

3.6. Determine designated port (DP) of each segment:

Example equal 2 ports on same bridge connected to the same segment:

P3 and P4 on Br4 are connected to the same segment.

The port with the lower port ID, i.e. P3, becomes DP on that segment.



3.7. Place redundant links into backup state:

All ports that are neither RP nor DP are placed into backup state, i.e. are put into port state **Blocking**.

STP has now converged to a stable topology (RB is selected, port roles are determined).

3.8. Learning & Forwarding:

RPs and DPs are placed into state **Learning** and, after the timer `ForwardDelay` expires, into the state **Forwarding**.

The network is now fully operational and starts forwarding frames.

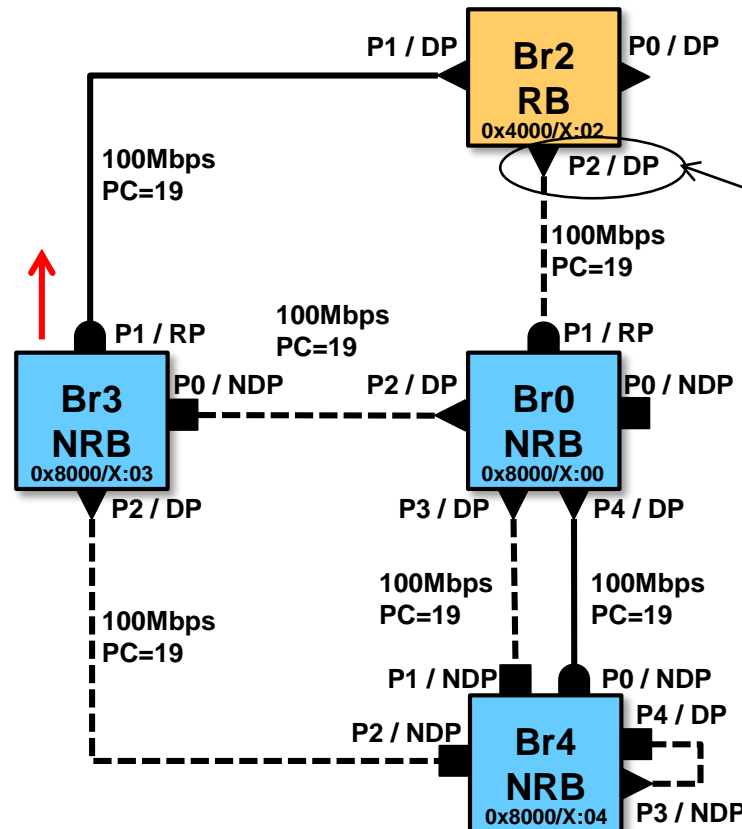
3. IEEE 802.1D STP protocol (28/32):

3.9. Topology reconfiguration upon topology changes:

Topology changes need to be communicated to other bridges in the network to re-establish a new loop-free topology as quickly as possible. Otherwise, hosts may not be reachable for up to 5 minutes (MAC address aging time in learning bridges).

Without topology change notification, Br3 in the example network would continue forwarding frames to Br4 through Br2 for 5 minutes (MAC address aging time) when P2 on Br2 goes down.

Without TCN, Br3 continues forwarding frames through P1 towards Br2 to reach Br4. These frames will not reach Br4 for up to 5 minutes.



P2 on Br2 goes down, e.g. through administrative action.

3. IEEE 802.1D STP protocol (29/32):

3.9. Topology reconfiguration upon topology changes:

Topology change notification BPDUs (TCN BPDUs) are sent by a bridge with a topology change on its RP towards the RB when either of the following is true:

A. Port in state **Forwarding** goes down, e.g. into state **Blocking**.

B. A port goes into state **Forwarding** and the bridge has a designated port (bridge port with an attached host that becomes active).

The upstream bridge responds with a TCA (Topology Change Ack) configuration BPDU with the Topology Change Ack bit set.

Again, the upstream bridge sends a TCN on its RP which again is acknowledged by the receiving bridge on its DP.

Finally, the TCNs reach the RB.

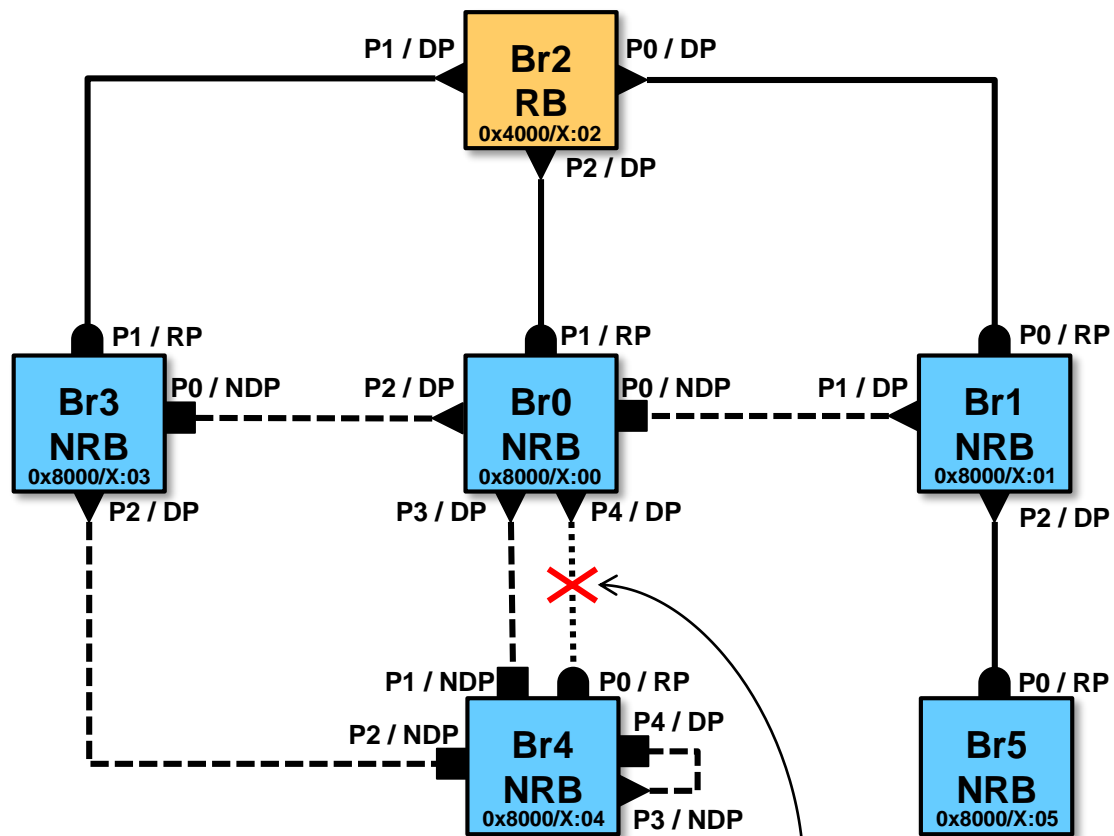
The RB then sends configuration BPDUs with the TC (Topology Change) bit set thus informing all bridges in the network of the topology change.

As a consequence, the bridges will temporarily reduce the aging time from 5 minutes to `ForwardDelay` (15 seconds) thus quickly aging out stale MAC table entries.

3. IEEE 802.1D STP protocol (30/32):

3.9. Topology reconfiguration upon topology changes:

Example with a link going down:

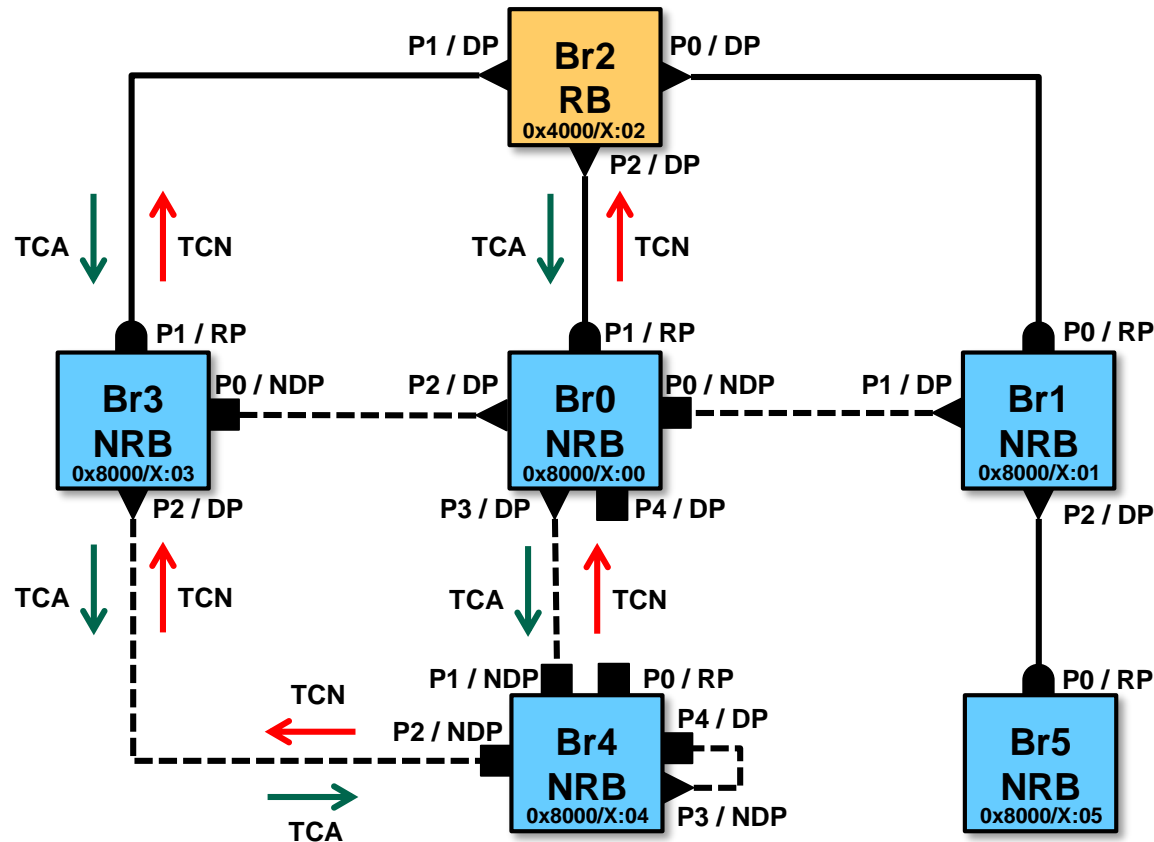


**Cable is pulled, thus ports
P4 on Br0 and P0 on Br4 go down.**

3. IEEE 802.1D STP protocol (31/32):

3.9. Topology reconfiguration upon topology changes:

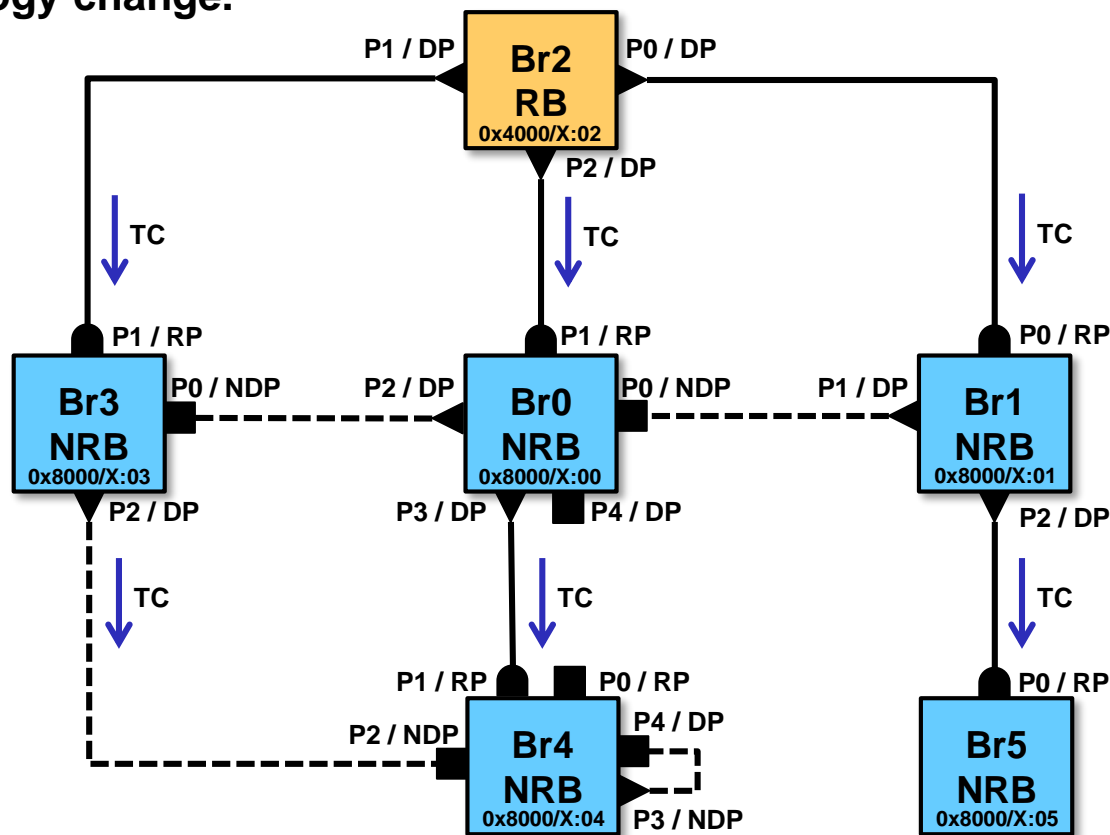
Br4 sends TCNs on all of its ports (root port became unavailable because cable was pulled). The receiving bridges acknowledge the TCNs with TCAs and forward the TCNs on their RPs until the RB is reached.



3. IEEE 802.1D STP protocol (32/32):

3.9. Topology reconfiguration upon topology changes:

Finally, the RB sends topology change notification BPDUs (TC, configuration BPDU with topology change bit set) on all of its ports that are forwarded by all bridges thus informing all bridges of the topology change.



4. IEEE 802.1w RSTP Rapid STP (1/17):

Why RSTP?

802.D is slow to converge in case of topology changes. RSTP introduces some mechanisms to speed up the convergence.

STP convergence time = MaxAge + ForwardDelay + ForwardDelay = 20s + 15s + 15s = 50s.

RSTP convergence time <= 10 seconds

IEEE 802.1w is incorporated in IEEE 802.1D:2004 which obsoletes STP and recommends to use RSTP instead.

Changes from STP to RSTP:

A number of changes were applied to RSTP for improved performance.

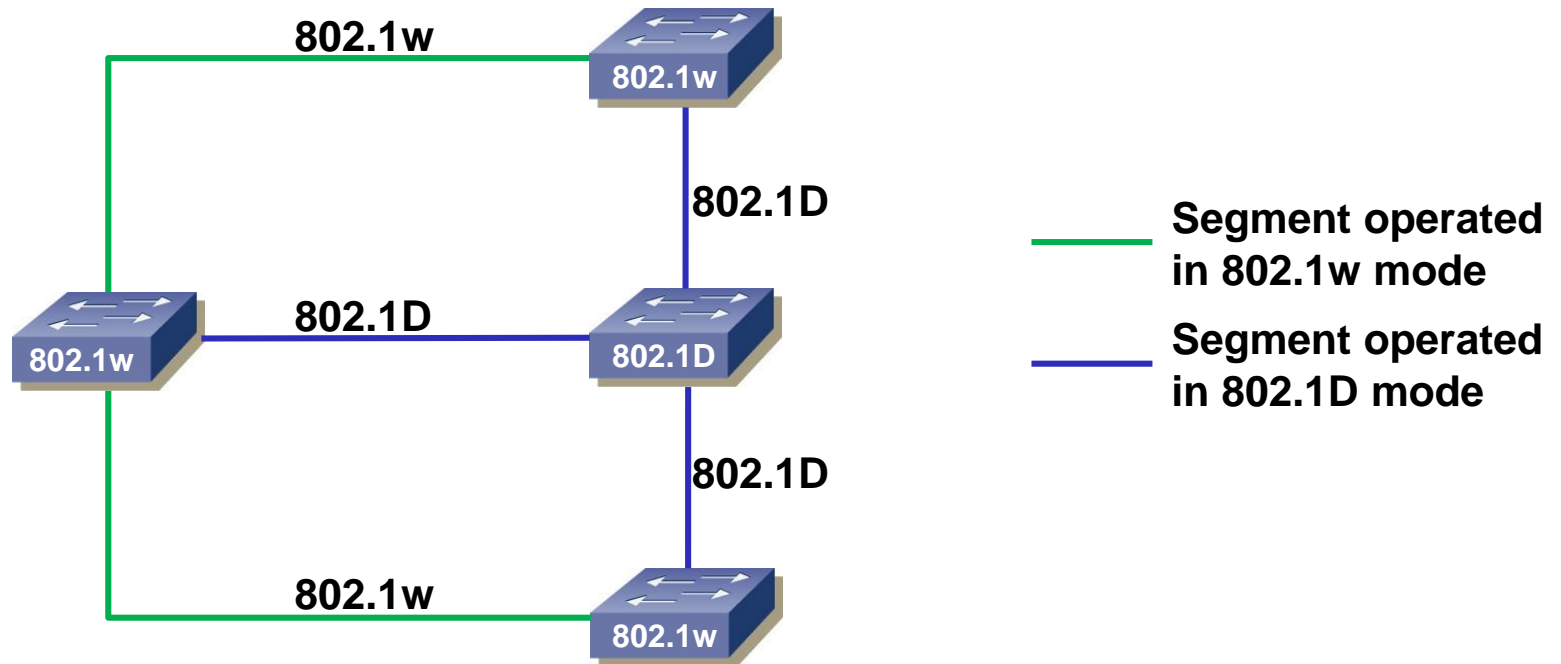
1. Redefined port states
2. Redefined port roles
3. Redefined path cost values
4. Slightly changed BPDU format
5. Rapid convergence and transition to forwarding state
6. Changed topology change notification mechanism

4. IEEE 802.1w RSTP Rapid STP (2/17):

4.1. Compatibility with STP:

RSTP bridges automatically fall back to STP (IEEE 802.1D) on a port to a neighbor bridge that only supports STP.

The convergence time for the entire topology is then determined by the slower operation of STP.



4. IEEE 802.1w RSTP Rapid STP (3/17):

4.2. New port states:

In RSTP, there are only 3 port states. The STP port states **Disabled**, **Blocking** and **Listening** were merged into the RSTP state **Discarding**.

Thus in RSTP, the port states express a port's function with regard to MAC frame processing, i.e. a port either discards MAC frames, learns MAC addresses from received frames but still discards frames or, when in **Forwarding** state, learns MAC addresses as well as forwards MAC frames.

802.1D STP Port State	802.1w RSTP Port State	Forward Ethernet frames	Learn MAC addresses
Init	Init	No (discard frames)	No
Disabled	Discarding	No (discard frames)	No
Blocking	Discarding	No (discard frames)	No
Listening	Discarding	No (discard frames)	No
Learning	Learning	No (discard frames)	Yes (populate MAC address table)
Forwarding	Forwarding	Yes	Yes

4. IEEE 802.1w RSTP Rapid STP (4/17):

4.3. New port roles:

The STP port role NDP (Non-Designated Port) was split into the 2 roles Alternate Port (AP) and Blocked Port (BP).

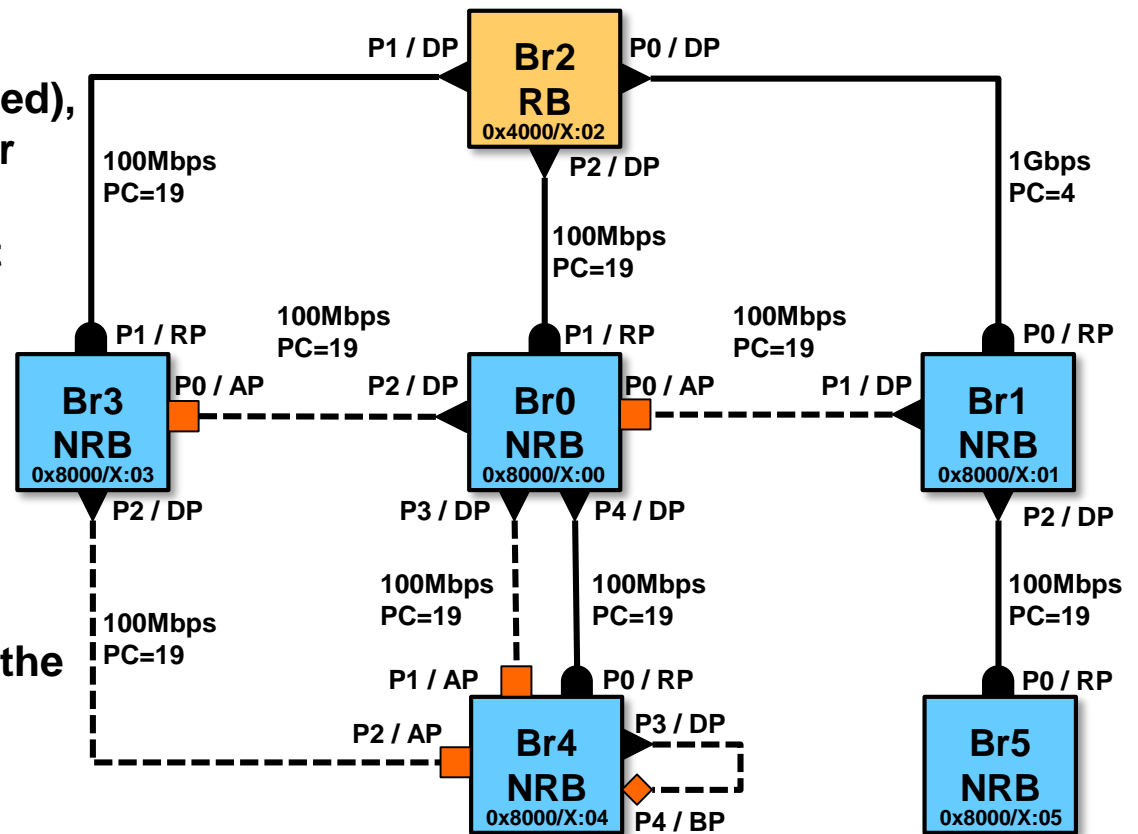
An alternate port provides an alternative path to the root bridge with fast switchover in case the root port on a bridge fails.

In case P0 on Br4 fails (e.g. cable pulled), Br4 immediately chooses P1 as RP for forwarding frames towards the RB.

If P1 fails too, Br4 chooses P2 as port towards the RB.

This failover feature greatly reduces the convergence time in case of failures.

P4 is a Blocked Port (BP) because it is on the same bridge as P3 and thus does not provide an alternate path to the root bridge.



Key:

- RP
- DP
- AP
- BP
- Port role = Root Port
- Port role = Designated Port
- Port role = Alternate Port
- Port role = Blocked Port

4. IEEE 802.1w RSTP Rapid STP (5/17):

4.4. New path cost:

802.1D encoded path cost in 16 bits thus supporting the range 1-65535.

The highest link speed supported by 802.1D was 10Gbps.

802.1w encodes path cost with 32 bit path cost values that support higher link speeds (up to 10 Tbps).

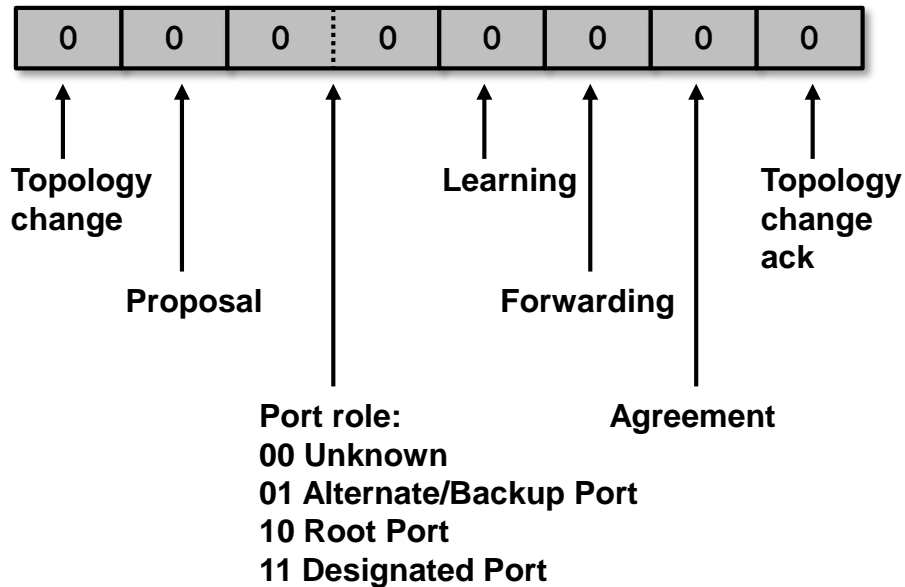
Link speed	802.1D Path Cost Recommended Value	802.1D Path Cost Recommended Range	802.1D Range	802.1w Path Cost Recommended Value	802.1w Path Cost Recommended Range	802.1w Range
4 Mbps	250	100-1000	1-65535	N/A	N/A	N/A
10Mbps	100	50-600	1-65535	2'000'000	200'000-20'000'000	1-200'000'000
16Mbps	62	40-400	1-65535	N/A	N/A	N/A
100Mbps	19	10-60	1-65535	200'000	20'000-2'000'000	1-200'000'000
1Gbps	4	3-10	1-65535	20'000	2'000-200'000	1-200'000'000
10Gbps	2	1-5	1-65535	2'000	200-20'000	1-200'000'000
100Gbps	N/A	N/A	N/A	200	20-2'000	1-200'000'000
1Tbps	N/A	N/A	N/A	20	2-200	1-200'000'000
10Tbps	N/A	N/A	N/A	2	1-20	1-200'000'000

4. IEEE 802.1w RSTP Rapid STP (6/17):

4.5. Changes to BPDU format:

In 802.1D, the configuration BPDU flag field only uses bits 7 (Topology change) and 0 (Topology change ack).

802.1w bridges also use the other 6 bits (Proposal, Port Role, Learning, Forwarding, Agreement) to exchange port state information with neighboring bridges thus speeding up the RSTP convergence (see 4.6. Rapid Convergence).



4. IEEE 802.1w RSTP Rapid STP (7/17):

4.6. Rapid convergence and transition to forwarding state:

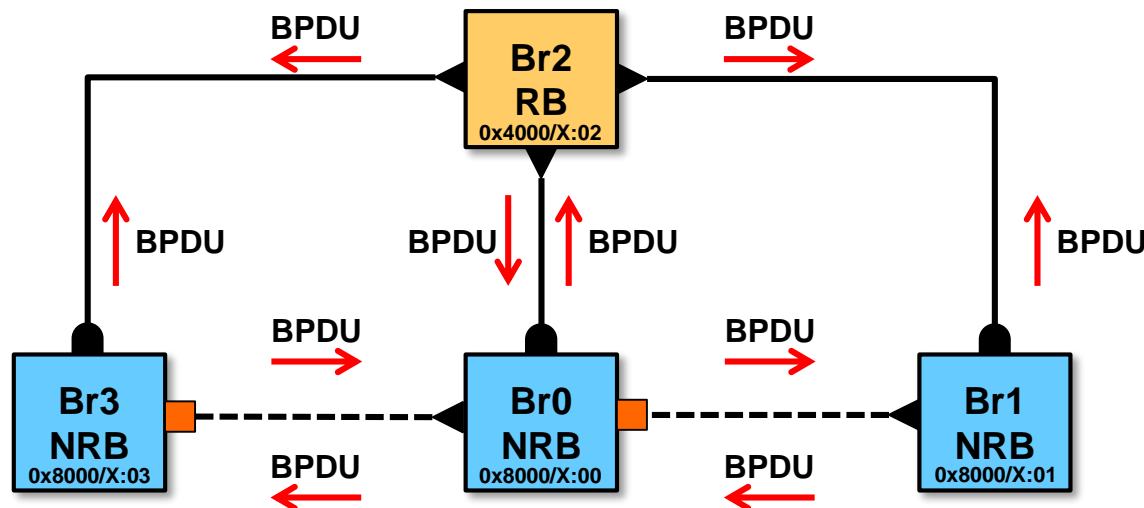
802.1w introduces a number of mechanisms to speed up the convergence process and re-establishment of a stable topology in case of topology changes.

Active sending of BPDUs:

In 802.1D in the converged state, bridges only forward BPDUs that emanate from the root bridge.

In 802.1w, bridges send BPDUs with their current information every *HelloTime* (2sec).

If no BPDU is received for 3 consecutive *HelloTime* intervals, bridges assume that connectivity to the peer bridge on that port is lost. In that case, protocol information can be quickly aged out (at the latest after *MaxAge* expires).

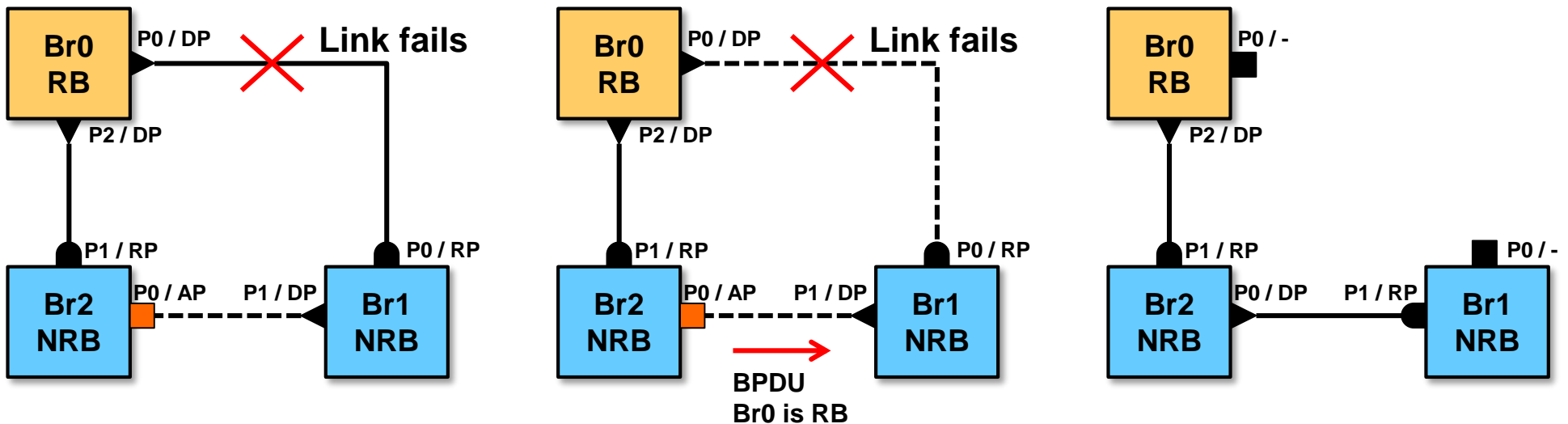


4. IEEE 802.1w RSTP Rapid STP (8/17):

4.6. Rapid convergence and transition to forwarding state:

Acceptance of inferior BPDUs:

Accepting inferior BPDUs allows bridges to quickly change the root port in case of link failures. In the example below, Br1 accepts inferior BPDUs on its port P1 as soon as the link to Br0 fails. Br2 sends BPDUs advertising that Br0 is the RB. When Br1 learns that the direct link to Br0 (=RB) fails, it immediately accepts inferior BPDUs from Br2 and makes port P1 the new root port.

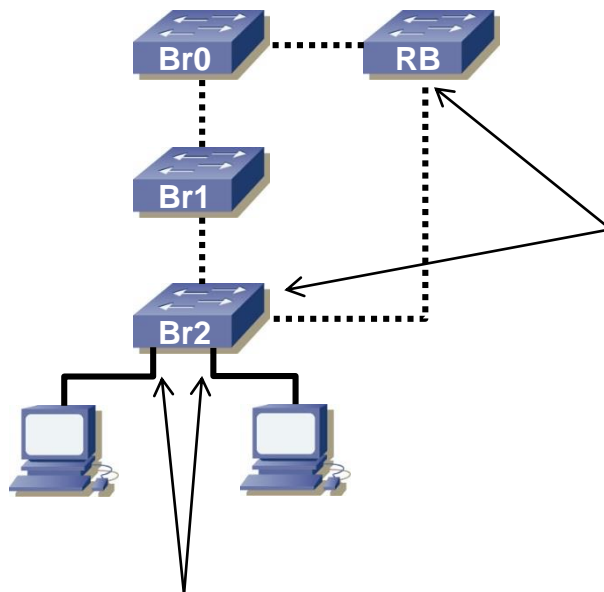


4. IEEE 802.1w RSTP Rapid STP (9/17):

4.6. Rapid convergence and transition to forwarding state:

Rapid transition to forwarding state:

802.1w actively exchanges information about port states (sync), thus allowing faster transition of ports into **Forwarding** state.



Spanning tree ports connect bridges and thus can create loops. 802.1w negotiates spanning tree port roles between neighboring bridges, thus speeding up the convergence and transition into **Forwarding** state.

Edge ports connect to hosts, thus edge ports cannot create loops and can directly transition to **Forwarding** state.

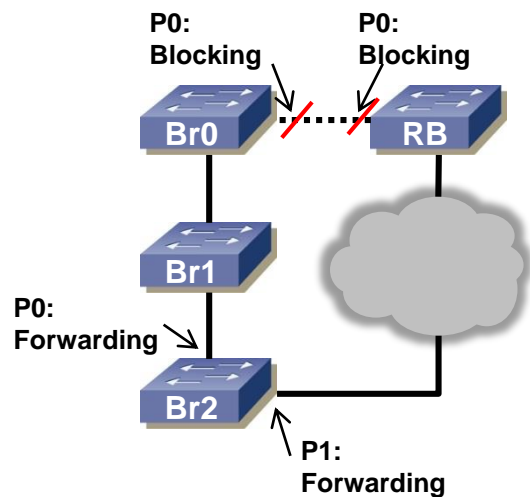
Edge ports are administratively defined (configuration).

4. IEEE 802.1w RSTP Rapid STP (10/17):

4.6. Rapid convergence and transition to forwarding state:

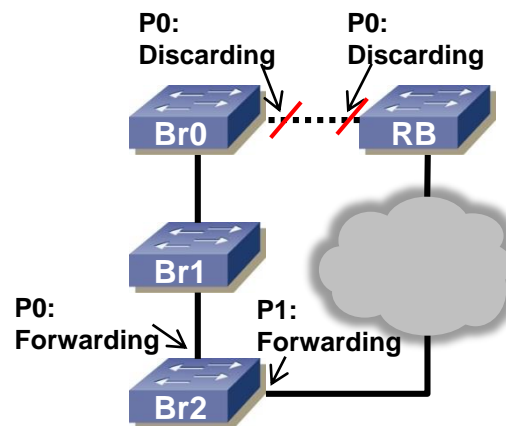
Comparison example between STP and RSTP. Br1 has an indirect connection to the RB. A new link is added between Br0 and RB.

802.1D STP: T + 0s



Both ports on BR0 and RB are put into the **Blocking** state.

802.1w RSTP: T + 0s

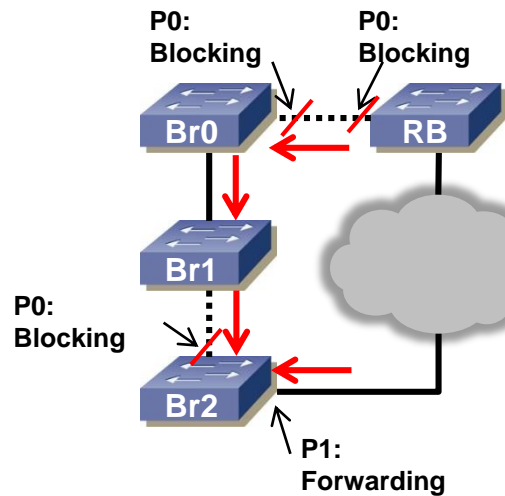


Both ports on BR0 and RB are put into the **Discarding** state.

4. IEEE 802.1w RSTP Rapid STP (11/17):

4.6. Rapid convergence and transition to forwarding state:

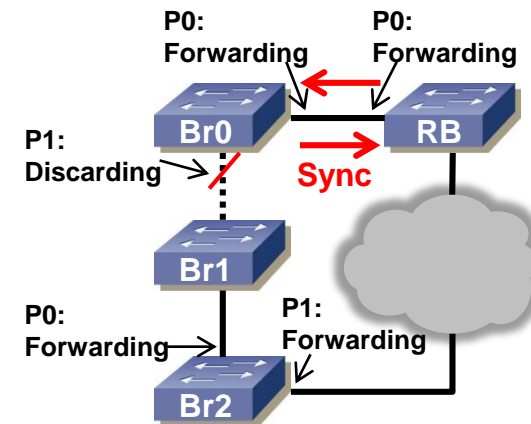
802.1D STP: T + HelloTime



BPDUs from the RB are quickly propagated through the new link to the leaves of the STP topology.

Br2 receives BPDUs on ports P0 and P1 and immediately blocks P0 due to the topology change.

802.1w RSTP: T + HelloTime



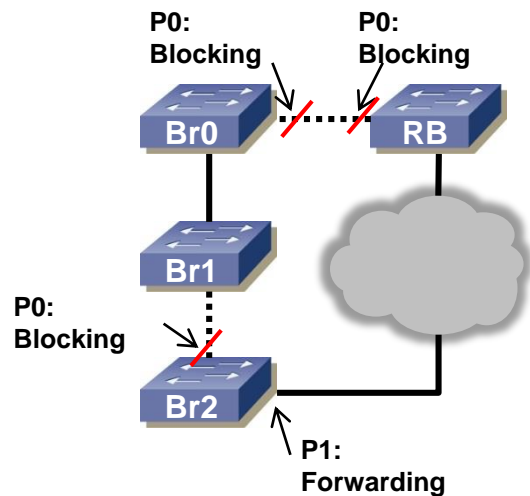
Br0 and RB negotiate the port roles (sync). P0 on Br0 becomes RP while P0 on RB becomes DP.

Br0 puts P1 into **Discarding** state to avoid loops before putting P0 into **Forwarding** state.

4. IEEE 802.1w RSTP Rapid STP (12/17):

4.6. Rapid convergence and transition to forwarding state:

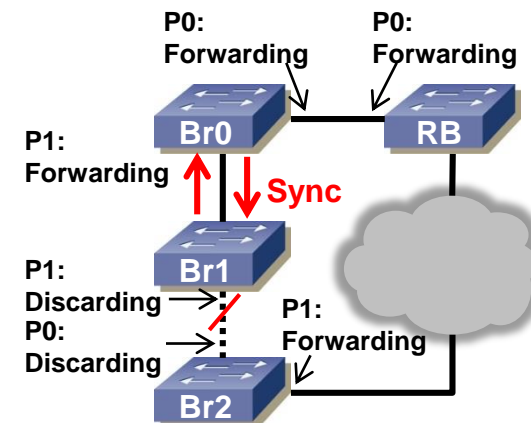
802.1D STP: T + a few seconds



The STP topology still blocks P0 on Br2 and the ports on the link between Br0 and RB.

Br0 and Br1 are still isolated from the rest of the network.

802.1w RSTP: T + max. 10 seconds



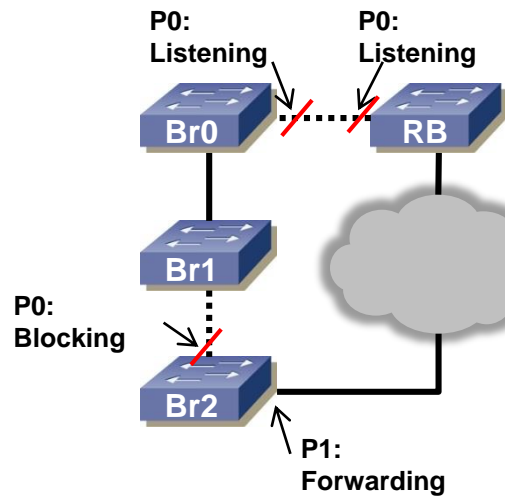
Quickly afterwards, the sync is taking place between Br0 and Br1. Before Br1 puts its port P0 to **Forwarding** state, it blocks its port P1 to avoid loops. Thus the cut travels down the tree.

The RSTP algorithm now converged to a new stable topology.

4. IEEE 802.1w RSTP Rapid STP (13/17):

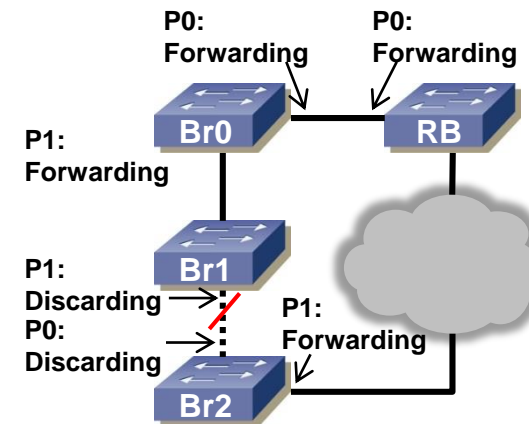
4.6. Rapid convergence and transition to forwarding state:

802.1D STP: T + MaxAge



After MaxAge expires, both ports on BR0 and RB are put into the **Listening** state.

802.1w RSTP: T > 10 seconds

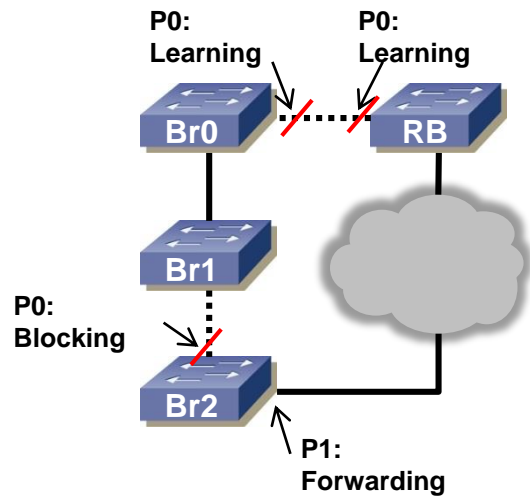


RSTP status: **Converged**.

4. IEEE 802.1w RSTP Rapid STP (14/17):

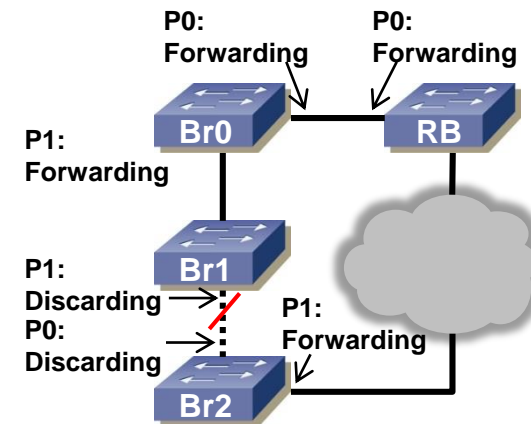
4.6. Rapid convergence and transition to forwarding state:

802.1D STP: $T + \text{MaxAge} + \text{ForwardDelay}$



After ForwardDelay expires, the ports go into the state **Learning**.

802.1w RSTP: $T > 10$ seconds

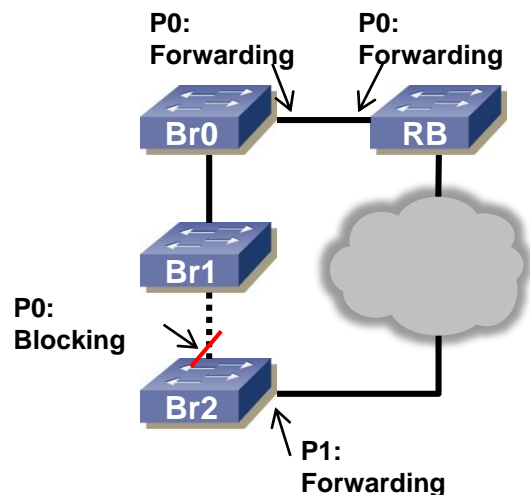


RSTP status: **Converged**.

4. IEEE 802.1w RSTP Rapid STP (15/17):

4.6. Rapid convergence and transition to forwarding state:

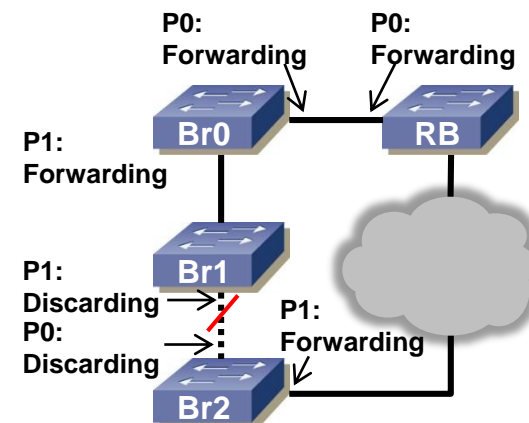
802.1D STP: $T + \text{MaxAge} + 2 * \text{ForwardDelay}$



After another ForwardDelay expires, the ports on RB, Br0 and Br2 go into the **Forwarding** state.

For up to 50 seconds, Br0 and Br1 were isolated from the rest of the network before frame forwarding was resumed in the entire network.

802.1w RSTP: $T > 10$ seconds



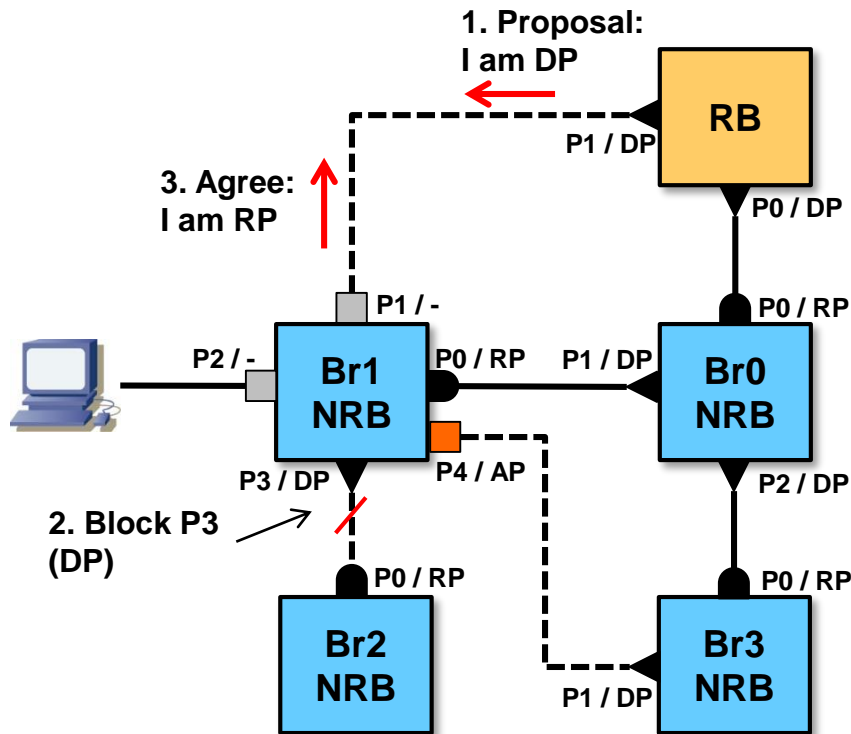
RSTP status: **Converged.**

4. IEEE 802.1w RSTP Rapid STP (16/17):

4.6. Rapid convergence and transition to forwarding state:

Proposal / agreement between bridges:

In the example below, a new link is added between the RB and Br1.
The RB makes its port P1 DP because all ports on the RB are DP.



The RB sends a proposal BPDU to the peer bridge. The peer bridge Br0 accepts the proposal because it sees that the proposal comes from the RB.

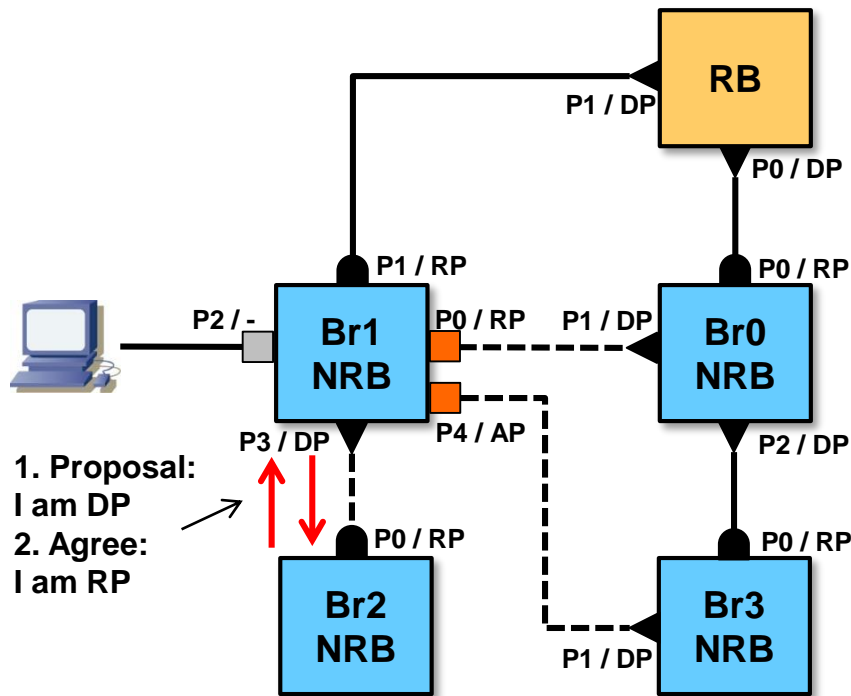
Before sending back an agreement, Br0 turns P0 from RP to AP (Br1 now has a better path to the RP through P1) and blocks P3 because now there may be a loop through P3.

Br1 does not need to block P2 (edge port) and P4 (alternate port) because no loop is possible through either of these ports. Both P2 and P4 are said to be in sync.

4. IEEE 802.1w RSTP Rapid STP (17/17):

4.6. Rapid convergence and transition to forwarding state:

Proposal / agreement between bridges:



Br1 sends out a proposal to Br2 that P3 is DP.

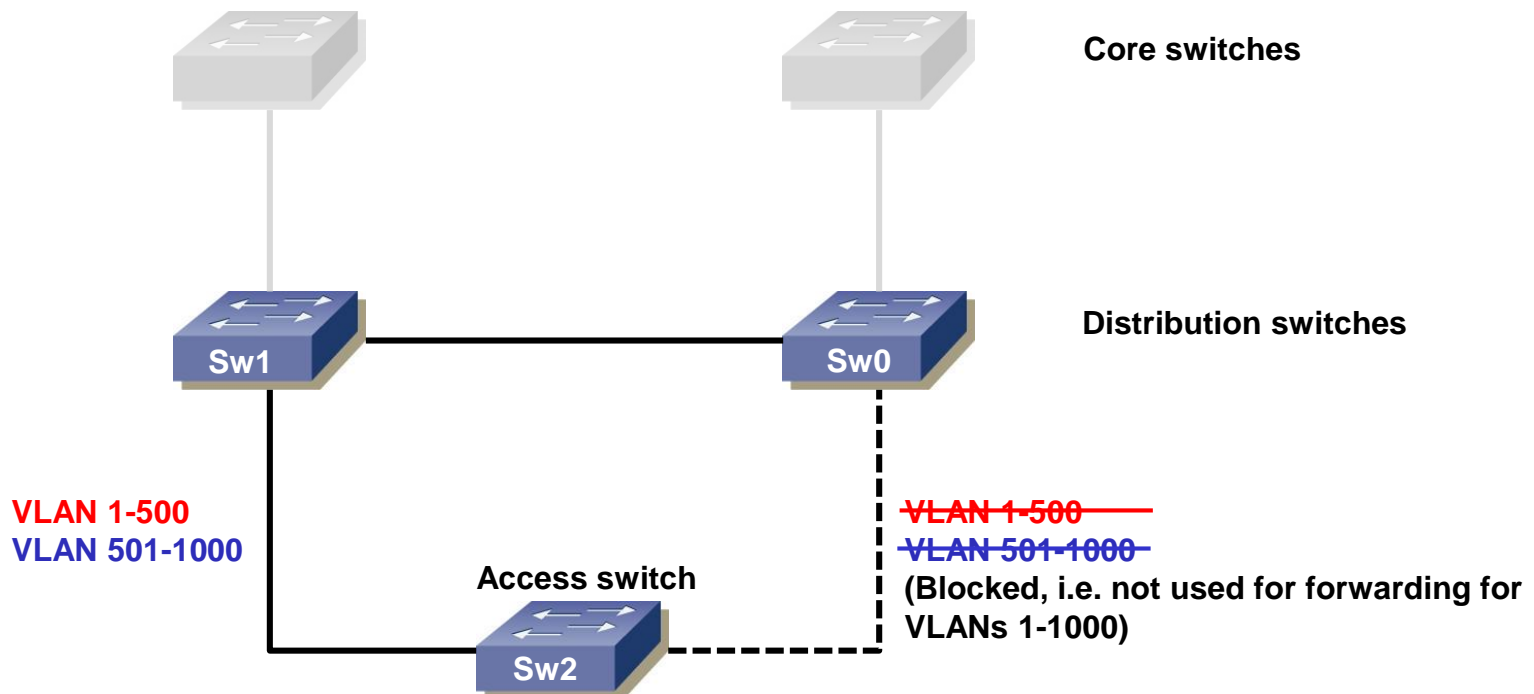
Br2 responds with an agreement that its port P0 is RP.

The sync process quickly assigned the new roles to the ports resulting in fast convergence.

5. IEEE 802.1Q CST Common Spanning Tree:

IEEE 802.1Q (VLAN) defines a common spanning tree for all VLANs in a physical network. In the setup below, an access switch (bridge in STP terminology) has 1000 VLANs and is connected to 2 distribution switches, both connected for failover purposes.

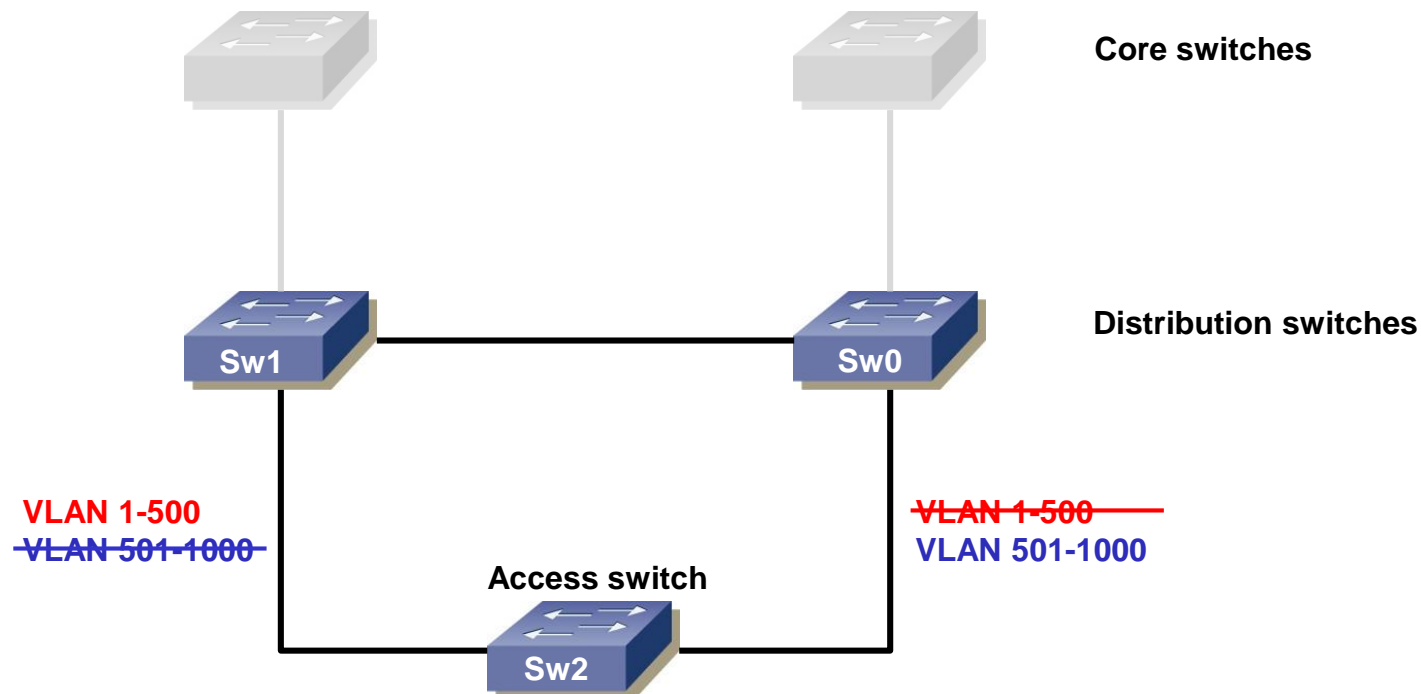
With only 1 instance of STP (or RSTP) for VLANs as defined in 802.1Q CST, only 1 of the links is used for forwarding traffic towards the distribution switches thus resulting in an inefficient use of links.



6. Cisco PVST+ and PVRST+:

Cisco's PVST+ and PVRST+ define a separate spanning tree instance for each VLAN. By defining Sw0 to be RB for VLANs 1-500 and Sw1 to be RB for VLANs 501-1000, respectively, load balancing can be achieved.

However, defining a separate spanning tree instance for each VLAN requires a lot of resources (CPU processing power, memory) and is therefore inefficient.



7. IEEE 802.1s MST Multiple Spanning Tree Protocol:

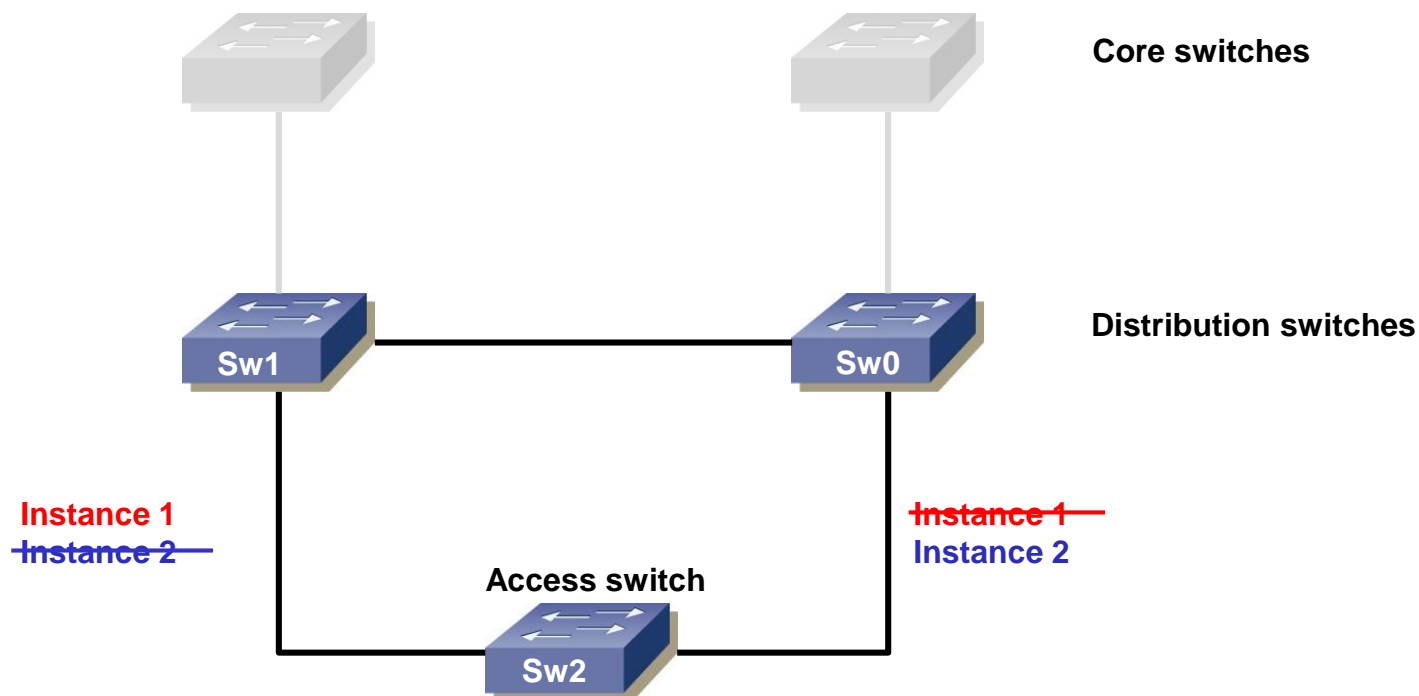
MSTP, originally defined in 802.1s and then merged to 802.1Q-2005, allows mapping multiple VLANs to a single spanning tree instance.

This reduces the resource requirements while preserving the advantages of having multiple spanning trees for load balancing purposes.

In the example below, the VLANs are mapped to 2 separate spanning tree instances as follows:

VLANs 1-500 → Spanning tree instance 1

VLANs 501-1000 → Spanning tree instance 2



8. STP Pros and Cons

Pros:

- 😊 Simple
- 😊 Proven and wide support by bridges and switches

Cons:

- 😞 Not optimal use of network infrastructure (links in backup state are not used for forwarding)
- 😞 RB represents a SPoF (Single Point of Failure)
- 😞 All traffic goes through RB (thus may become a bottleneck)
- 😞 Slow convergence in case of topology changes (fixed with RTSP)

Newer approaches like TRILL ([RFC6325](#)) or SPB (Shortest Path Bridging, IEEE 802.1aq) allow using all links thus improving network utilization while providing features like load distribution and balancing.