13. Spanning Tree Protocols

The spanning tree algorithms and protocols specified by this standard provide simple and full connectivity throughout a Bridged Local Area Network comprising arbitrarily interconnected bridges. Each bridge can use the Rapid Spanning Tree Protocol (RSTP), Multiple Spanning Tree Protocol (MSTP), or SPB (Shortest Path Bridging) protocols.

NOTE 1—The spanning tree protocols specified by this standard supersede the Spanning Tree Protocol (STP) specified in IEEE Std 802.1D revisions prior to 2004, but facilitate migration by interoperating with the latter without configuration restrictions beyond those previously imposed by STP. However networks that include bridges using STP can reconfigure slowly and constrain active topologies.

NOTE 2—Although the active topologies determined by the spanning tree protocols connect all the components of a Bridged Local Area Network, filtering (MVRP, etc.) can restrict frames to a subset of each active topology.

RSTP assigns all frames to a Common Spanning Tree (CST), without being aware of the active topology assignments made by MSTP or SPB that allow frames to follow separate paths within Multiple Spanning Tree (MST) or Shortest Path Tree (SPT) Regions. Each of these regions comprises MST or SPT Bridges that consistently assign any given frame to the same active topology (see ***8.4***) and the LANs that interconnect those bridges. These regions and other bridges and LANs are connected into the CST, to provide loop free network wide connectivity even if active topology assignments or spanning tree protocols differ locally.

MSTP and SPB connect all bridges and LANs with a single Common and Internal Spanning Tree (CIST) that supports the automatic determination of each region, choosing its maximum possible extent. The connectivity calculated for the CIST provides the CST for interconnecting the regions, and an Internal Spanning Tree (IST) within each region. MSTP calculates a number of independent Multiple Spanning Tree Instances (MSTIs) within each region, and ensures that frames with a given VID are assigned to one and only one of the MSTIs or the IST within the region (or reserved for use by SPB or PBB-TE), that the assignment is consistent among all bridges within the region, and that the stable connectivity of each MSTI and the IST at the boundary of the region matches that of the CST. SPB calculates symmetric sets of Shortest Path Trees (SPTs), each rooted at a bridge within a region, and ensures that frames for any given VLAN are assigned to the same symmetric SPT set within the region (or to the IST, an MSTI, or PBB-TE). Within an SPT Region, each shortest path bridged frame is assigned to the SPT rooted at the bridge providing ingress to the region for that frame, either by replacing the frame's Base VID (3.1) with one of a number of Shortest Path VIDs (SPVIDs) that support its VLAN within the region, or (when its VLAN is supported by shortest path backbone bridging) by using the frame's source address in conjunction with the Base VID. SPB protocols can dynamically allocate SPVIDs and ensure that their allocation is consistent within a region.

Spanning tree protocol entities transmit and receive BPDUs (Clause 14) to convey parameters used by RSTP and MSTP to calculate CST, CIST, and MSTI spanning trees. BPDUs also convey parameters that are used by all spanning tree protocols to interoperate with each other and with STP, that determine the extent of MST and SPT Regions, and that ensure that temporary loops are not created because neighbouring bridges are not yet acting on the same topology information. SPB uses ISIS (see Clause 29) to disseminate the information used to calculate the IST and SPTs, and to perform that calculation, and Tree Agreement Protocol (TAP, 13.17) to ensure loop-free connectivity. TAP Messages are conveyed within BPDUs.

This clause

a) Specifies protocol design and support requirements (13.1, 13.2) and design goals (13.3).
b) Provides an overview of RSTP (13.4), MSTP (13.5), SPB and SPBB (13.6) operation.
c) Describes how the spanning tree protocols interoperate and coexist (13.7)
d) Specifies how spanning tree priority vectors (13.9) are calculated (13.10, 13.11) and used to assign the Port Roles (13.12) that determine the Port States, i.e. forwarding and learning (***8.4***), for each tree.
e) Shows that RSTP, MSTP, and the SPB protocols provide stable connectivity (13.13).
f) Describes how spanning tree priority vectors are communicated (13.14) and changed (13.15).

g) Describes how Port Role are used to change Port States without introducing loops (13.16, 13.17).
h) Recommends defaults and ranges for the parameters that determine each tree's topology (13.18).
i) Describes the updating of learned station location information when a tree reconfigures (13.19).
j) Specifies additional controls that can speed reconfiguration or prevent unwanted outcomes (13.20).
k) Describes how loops are prevented when a LAN is only providing one-way connectivity (13.21), and can be prevented when the network includes bridges whose protocol operation can fail (13.22).
l) Describes how a bridge's protocol processing can be 'hot upgraded' in an active network (13.23).
m) Specifies RSTP, MSTP, and TAP using state machines (13.24–13.40).
n) Specifies the use and configuration of the spanning tree protocols for the special cases of a Provider Edge Bridge's Customer Edge Ports (13.41), a Backbone Edge Bridge's Virtual Instance Ports (13.42), and an L2 Gateway Port connecting a customer to a provider (13.43).

NOTE 3—Readers of this specification are urged to begin by familiarizing themselves with RSTP.

Clause 14 specifies the format of BPDUs. Clause 27 describes the uses of Shortest Path Bridging. ***Clause 29*** specifies ISIS-SPB. The text of this clause (Clause 13) takes precedence should any conflict be apparent between it and the text in other parts of this standard (in particular, Clause 12, Clause 14, and Annex A).

13.1 Protocol design requirements

The Spanning Tree Algorithm and its associated protocols operate in Bridged Local Area Networks of arbitrary physical topology comprising SPT, MST, RST, or STP Bridges connecting shared media or point-to-point LANs, so as to support, preserve, and maintain the quality of the MAC Service in all its aspects as specified by Clause 6.

RSTP configures the Port State (7.4) of each Bridge Port. MSTP configures the Port State for the CIST and each MSTI, and verifies the allocation of VIDs to FIDs and FIDs to trees. SPB protocols configure the Port State for the CIST and each SPT, the VID Translation Table for each Bridge Port, allocate FIDs to SPT sets, and assigns frames to SPTs by SPVIDs and/or source address. Operating both independently and together RSTP, MSTP, and SPB protocols meet the following requirements:

a) They configure one or more active topologies that fully connect all physically connected LANs and bridges, and stabilize (with high probability) within a short, known bounded interval after any change in the physical topology, maximising service availability (6.5.1).
b) The active topology for any given frame remains simply connected at all times (6.5.3, 6.5.4), and will (with high probability) continue to provide simple and full connectivity for frames even in the presence of administrative errors (e.g. in the allocation of VLANs to MSTIs).
c) The configured stable active topologies are unicast multicast congruent, downstream congruent, and reverse path congruent (symmetric) (3.2, 6.3).
d) The same symmetric active topology is used, in a stable network, for all frames using the same FID, i.e. between any two LANs all such frames are forwarded through the same Bridge Ports (6.3).
e) The active topology for a given VLAN can be chosen by the network administrator to be a common spanning tree, one of multiple spanning trees (if MSTP is implemented), or shortest path (if SPB protocols are implemented).
f) Each active topology is predictable, reproducible, and manageable, allowing Configuration Management (following traffic analysis) to meet Performance Management goals (6.3 and 6.3.10).
g) The configured network can support VLAN-unaware end stations, such that they are unaware of their attachment to a single LAN or a Bridged Local Area Network, or their use of a VLAN (6.2).
h) The communications bandwidth on any particular LAN is always a small fraction of the total available bandwidth (6.5.10).

NOTE—The spanning tree protocols cannot protect against temporary loops caused by the interconnection of LANs by devices other than bridges (e.g., LAN repeaters) that operate invisibly with respect to support of the MAC Internal Sublayer Service and the MAC_Operational status parameter (***6.6.2***).

13.2 Protocol support requirements

In order for the spanning tree protocols to operate, the following are required:

a) A unique Group MAC Address used by the Spanning Tree Protocol Entities (***8.10***) of participating bridges or bridge components (***5.2***), and recognized by all the bridges attached to a LAN.

b) An identifier for each bridge or bridge component, unique within the Bridged Local Area Network.

c) A identifier for each Bridge Port, unique within a bridge or bridge component.

Values for each of these parameters shall be provided by each bridge. The unique MAC Address that identifies the Spanning Tree Protocol Entities of MAC Bridges, VLAN Bridges (***5.9***), and C-VLAN components (***5.5***) is the Bridge Group Address (***Table 8-1***). The unique MAC Address that identifies the Spanning Tree Protocol Entities of S-VLAN components is the Provider Bridge Group Address (***Table 8-2***).

To allow management of active topology (for RSTP, MSTP, or SPB) means of assigning values to the following are required:

d) The relative Bridge Priority of each bridge in the network.

e) A Port Path Cost for each Bridge Port.

f) The relative Port Priority of each Bridge Port.

13.2.1 MSTP support requirements

MSTP does not require any additional configuration, provided that communication between end stations is supported by a number of VLANs. However, to realize the improved throughput and associated frame loss and transit delay performance improvements made possible by the use of multiple spanning trees, the following are required:

a) Assessment of the probable distribution of traffic between VLANs and between sets of communicating end stations using those VLANs.

b) Per MSTI assignment of Bridge Priority and Internal Port Path Costs to configure the MSTIs.

c) Consistent assignment of VIDs to MSTIDs within each potential MST Region.

d) Administrative agreement on the Configuration Name and Revision Level used to represent the assignments of VIDs to MSTIDs.

13.2.2 SPB support requirements

SPB protocols can provide both common spanning tree and shortest path support for VLANs without further configuration (except as required for ISIS-SPB). To support such operation this standard specifies:

a) A Configuration Name and Revision Level used to represent the assignment of VLAN 1 to the SPT Primary Set, VLAN 2 to the SPT Alternate Set, and all other VIDs to the CIST.

Use of Shortest Path Backbone Bridging (SPBB) requires the following:

b) Explicit identification of each VLAN to be shortest path backbone bridged.

13.3 Protocol design goals

All the spanning tree protocols meet the following goal, which simplifies operational practice:

a) Bridges do not have to be individually configured before being added to the network, other than having their MAC Addresses assigned through normal procedures.
b) In normal operation, the time taken to configure the active topology of a network comprising point-to-point LANs is independent of the timer values of the protocol.

RSTP and MSTP meet the following goal, which limits the complexity of bridges and their configuration:

c) The memory requirements associated with each Bridge Port are independent of the number of bridges and LANs in the network.

It is highly desirable that the operation of SPB protocols support updating of the SPB configuration, so that:

d) SPT Bridges can be added to a region without disrupting communication for existing VLANs.
e) Additional VLANs can be supported by SPB without disrupting communication for the VLANs that are already supported by SPB, or for those VLANs that supported by RSTP or MSTP.
f) Shortest path bridging support can be enabled or disabled for individual VLANs that are being used to support user communication, with the minimum of frame loss on those VLANs.

13.4 RSTP overview

The Rapid Spanning Tree Protocol (RSTP) configures the Port State (3.5, 8.4) of each Bridge Port in the Bridge Local Area Network. RSTP ensures that the stable connectivity provided by each bridge between its ports and by the individual LANs attached to those ports is predictable, manageable, full, simple, and symmetric. RSTP further ensures that temporary loops in the active topology do not occur if the network has to reconfigure in response to the failure, removal, or addition of a network component, and that erroneous station location information is removed from the Filtering Database after reconfiguration.

Each of the bridges in the network transmits Configuration Messages (13.14). Each message contains spanning tree priority vector (13.9) information that identifies one bridge as the Root Bridge of the network, and allows each bridge to compute its own lowest path cost to that Root Bridge before transmitting its own Configuration Messages. A Port Role (13.12) of Root Port is assigned to the one port on each bridge that provides that lowest cost path to the Root Bridge, and a Port Role of Designated Port to the one Bridge Port that provides the lowest cost path from the attached LAN to the Root Bridge. Alternate Port and Backup Port roles are assigned to Bridge Ports that can provide connectivity if other network components fail.

State machines associated with the Port Roles maintain and change the Port States that control forwarding (***8.6***) and learning (***8.7***) of frames. In a stable network, Root Ports and Designated Ports are Forwarding, while Alternate, Backup, and Disabled Ports are Discarding. Each Port's role can change if a Bridge, Bridge Port, or LAN fails, is added to, or removed from network. Port state transitions to Learning and Forwarding are delayed, and ports can temporarily transition to the Discarding state prevent loops and to ensure that misordering (6.5.3) and duplication (6.5.4) rates remain negligible.

RSTP provides rapid recovery of connectivity to minimize frame loss (6.5.2). A new Root Port, and Designated Ports attached to point-to-point LANs, can transition to Forwarding without waiting for protocol timers to expire. A Root Port can transition to Forwarding without transmitting or receiving messages from other bridges, while a Designated Port attached to a point-to-point LAN can transition when it receives an explicit agreement transmitted by the other bridge attached to that LAN. The forwarding transition delay used by a Designated Port attached to a shared media LAN is long enough for other bridges attached to that

LAN to receive and act on transmitted messages, but is independent of the overall network size. If all the LANs in a network are point-to-point, RSTP timers define worst-case delays that only occur if protocol messages are lost or rate transmission limits are exceeded.

A Bridge Port attached to a LAN that has no other bridges attached to it may be administratively configured as an Edge Port. RSTP monitors the LAN to ensure that no other bridges are connected, and may be configured to automatically detect an Edge Port. Each Edge Port transitions directly to the Forwarding Port State, since there is no possibility of it participating in a loop.

13.4.1 Computation of the active topology

The bridge with the best Bridge Identifier is selected as the Root Bridge. The unique Bridge Identifier for each bridge is derived, in part, from the Bridge Address (***8.13.8***) and, in part, from a manageable priority component (13.26). The relative priority of bridges is determined by the numerical comparison of the unique identifiers, with the lower numerical value indicating the better identifier.

Every bridge has a Root Path Cost associated with it. For the Root Bridge this is zero. For all other bridges, it is the sum of the Port Path Costs on the least cost path to the Root Bridge. Each port’s Path Cost may be managed, 13.18 recommends default values for ports attached to LANs of various speeds.

The Bridge Port on each bridge with the lowest Root Path Cost is assigned the role of Root Port for that bridge (the Root Bridge does not have a Root Port). If a bridge has two or more ports with the same Root Path Cost, then the port with the best Port Identifier is selected as the Root Port. Part of the Port Identifier is fixed and different for each port on a bridge, and part is a manageable priority component (13.26). The relative priority of Bridge Ports is determined by the numerical comparison of the unique identifiers, with the lower numerical value indicating the better identifier.

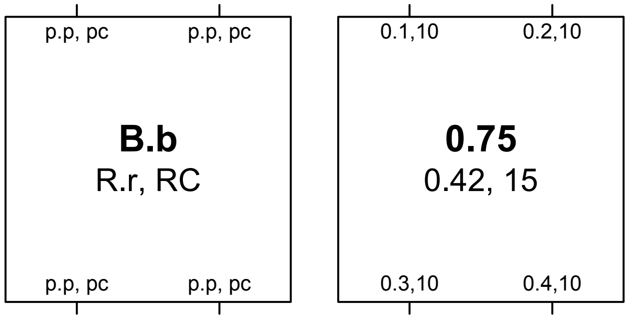
Each LAN in the Bridged Local Area Network also has an associated Root Path Cost. This is the Root Path Cost of the lowest cost bridge with a Bridge Port connected to that LAN. This bridge is selected as the Designated Bridge for that LAN. If there are two or more bridges with the same Root Path Cost, then the bridge with the best priority (least numerical value) is selected as the Designated Bridge. The Bridge Port on the Designated Bridge that is connected to the LAN is assigned the role of Designated Port for that LAN. If the Designated Bridge has two or more ports connected to the LAN, then the Bridge Port with the best priority Port Identifier (least numerical value) is selected as the Designated Port.

In a Bridged Local Area Network whose physical topology is stable, i.e RSTP has communicated consistent information throughout the network, every LAN has one and only one Designated Port, and every bridge with the exception of the Root Bridge has a single Root Port connected to a LAN. Since each bridge provides connectivity between its Root Port and its Designated Ports, the resulting active topology connects all LANs (is “spanning”) and will be loop free (is a “tree”).

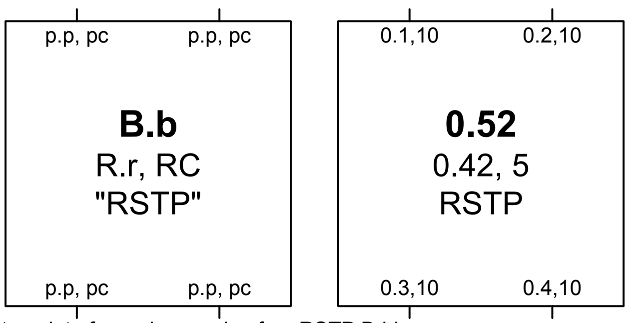
Any Bridge Port that is enabled, but not a Root or Designated Port, is a Backup Port if that bridge is the Designated Bridge for the attached LAN, and an Alternate Port otherwise. An Alternate Port offers an alternate path in the direction of the Root Bridge to that provided by the bridge’s own Root Port, whereas a Backup Port acts as a backup for the path provided by a Designated Port in the direction of the leaves of the Spanning Tree. Backup Ports exist only where there are two or more connections from a given bridge to a given LAN; hence, they (and the Designated Ports that they back up) can only exist where the bridge has two or more ports attached to a shared media LAN, or directly connected by a point-to-point LAN.

13.4.2 Example topologies

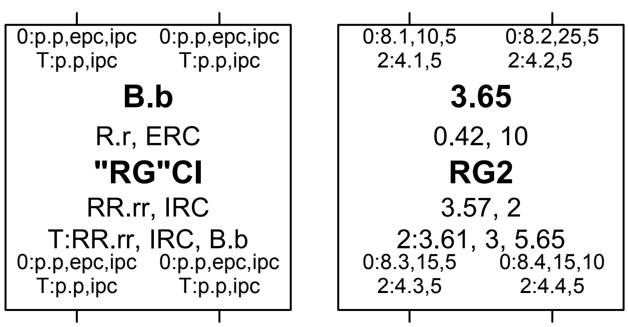
The spanning tree examples in this clause use the conventions of Figure 13-1.



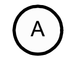
A template for and example of an STP Bridge. **B.b** is the Bridge Identifier (including the manageable priority component **B.**). **R.r** and **RC** are the Root Identifier, Root Path Cost, and the Designated Bridge Identifier, for the Root Port. **p.p, pc** are the Port Identifier (with manageable priority **p.**) and the Port Path Cost for a Bridge Port.



A template for and example of an RSTP Bridge.



A template for and an example of an MSTP Bridge. **B.b** is the CIST Bridge Identifier. **R.r**, **ERC**, **RR.rr** are the CIST Root Identifier, External Root Path Cost, and Regional Root Identifier. **CI** identifies the Configuration Identifier for the Bridge. **RR.rr, IRC** the CIST Regional Root Identifier and the Internal Root Path Cost. **T:RR.rr, IRC, B.b**, is the Regional Root Identifier, Internal Root Path Cost IRC, and Bridge Identifier for the MSTI with MSTID **T**.
-**p.p, epc, ipc** are the CIST Port Identifier, External Port Path Cost, and Internal Port Path Cost for a Bridge Port. **T:p.p, ipc** are the Port Identifiers and their Regional Costs for MSTI **T**.
Any of the above information may be selectively omitted if deemed irrelevant for the purposes of a diagram.



A LAN

Connections between Bridges and LANs indicate the Port Role and Port State by means of their end point symbols, and in some examples, may show the transmission of BPDUs from a Port onto a LAN by means of arrowheads, as shown in the following table.

Port Role	Port State	Legend
Designated	Discarding Learning Forwarding	
& operEdge	Forwarding	
Root Port or Master Port	Discarding Learning Forwarding	
Alternate	Discarding Learning Forwarding	
Backup	Discarding Learning Forwarding	
Disabled	-	
Transmitted Bpdus		
Designated Designated Proposal		
Root Root Agreement		

NOTE—These diagrammatic conventions allow the representation of Alternate and Backup Ports that are in Learning or Forwarding states; this can happen as a transitory condition due to implementation-dependent delays in switching off Learning and/or Forwarding on a Port that changes role from Designated or Root to Alternate or Backup.

Figure 13-1—Diagrammatic conventions for spanning tree topologies

Figure 13-2 shows a simple, redundantly connected, structured wiring configuration, with bridges connected by point-to-point LANs A through N, and a possible spanning tree active topology of the same network. bridge 111 has been selected as the Root (though one cannot tell simply by looking at the active topology which bridge is the Root).

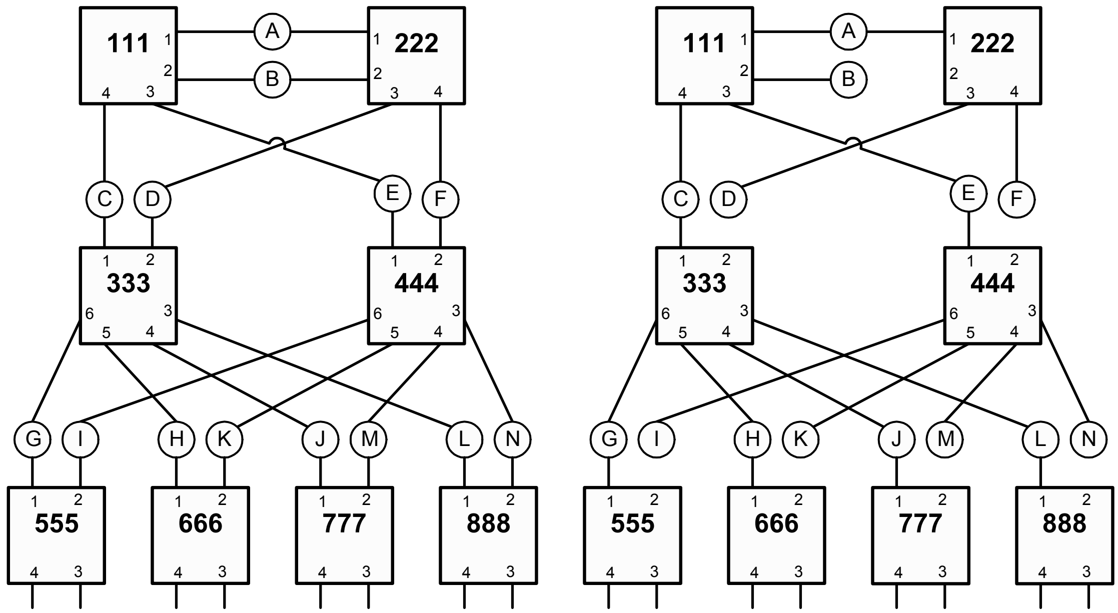


Figure 13-2—Physical topology and active topology

Figure 13-3 shows the Port Roles and Port States of each Bridge Port. It can be seen that bridge 111 is the Root, as its Ports are all Designated Ports, each of the remaining bridges have one Root Port.

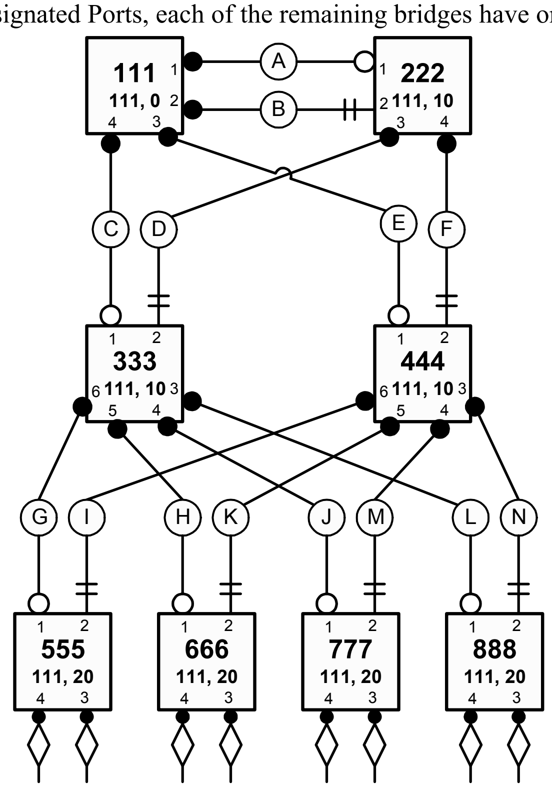


Figure 13-3—Port Roles and Port States

Figure 13-4 shows the result of connecting two of the ports of bridge 888 to the same LAN. As port 4 of bridge 888 has worse priority than port 3 and both offer the same Root Path Cost, port 4 will be assigned the Backup Port Role and will therefore be in the Discarding Port State. Should port 3 or its connection to LAN O fail, port 4 will be assigned the Designated Port Role and will transition to the Forwarding Port State.

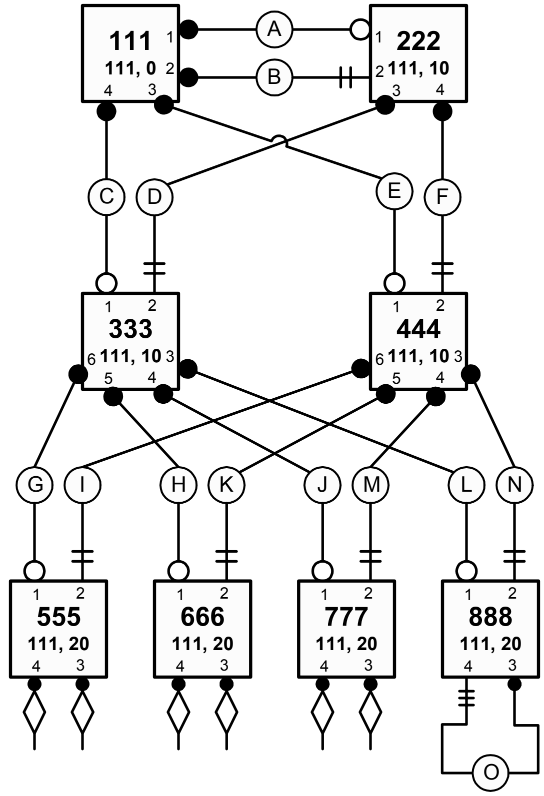


Figure 13-4—A Backup Port

Figure 13-5 shows a “ring” topology constructed from point-to-point links, as in some resilient backbone configurations. Bridge 111 is the Root, as in previous examples.

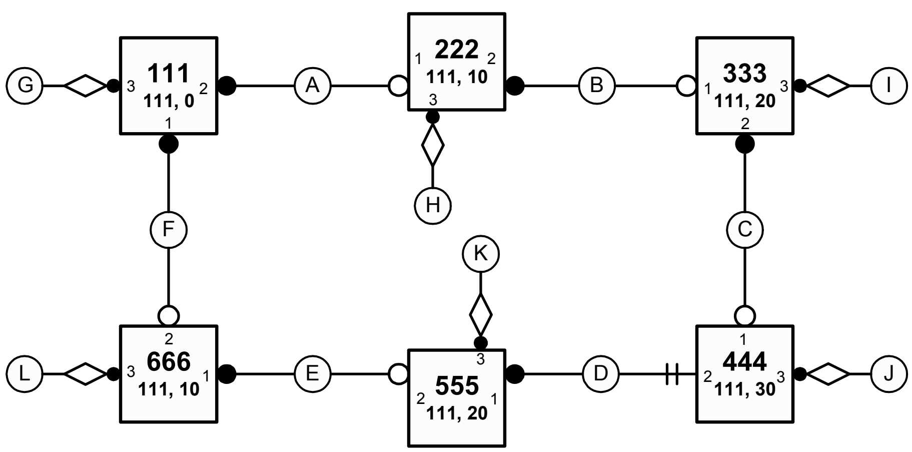


Figure 13-5—“Ring Backbone” example

13.5 MSTP overview

The Multiple Spanning Tree Protocol specifies:

a) An MST Configuration Identifier (13.8) that allows each bridge to advertise its assignment, to a specified MSTI or to the IST, of frames with any given VID.
b) A priority vector (13.9) that comprises bridge identifier and path cost information for constructing a deterministic and manageable single spanning tree active topology, the CIST, that:
    1) Fully and simply connects all bridges and LANs in a Bridged Local Area Network.
    2) Permits the construction and identification of MST Regions of bridges and LANs that are guaranteed fully connected by the bridges and LANs within each region.
    3) Ensures that paths within each MST Region are always preferred to paths outside the region.
c) An MSTI priority vector (13.9), comprising information for constructing a deterministic and independently manageable active topology for any given MSTI within each region.
d) Comparisons and calculations performed by each bridge in support of the distributed spanning tree algorithm (13.10). These select a CIST priority vector for each Bridge Port, based on the priority vectors and MST Configuration Identifiers received from other bridges and on an incremental Path Cost associated with each receiving port. The resulting priority vectors are such that in a stable network:
    1) One bridge is selected to be the CIST Root of the Bridged Local Area Network as a whole.
    2) A minimum cost path to the CIST Root is selected for each bridge and LAN, thus preventing loops while ensuring full connectivity.
    3) The one bridge in each MST Region whose minimum cost path to the Root is not through another bridge using the same MST Configuration Identifier is identified as its region's CIST Regional Root.
    4) Conversely, each bridge whose minimum cost path to the Root is through a bridge using the same MST Configuration Identifier is identified as being in the same region as that bridge.
e) Priority vector comparisons and calculations performed by each bridge for each MSTI (13.11). In a stable network:
    1) One bridge is independently selected for each MSTI to be the MSTI Regional Root.
    2) A minimum cost path to the MSTI Regional Root that lies wholly within the region is selected for each bridge and LAN.
f) CIST Port Roles (13.12) that identify the role in the CIST active topology played by each port on a bridge.
    1) The Root Port provides the minimum cost path from the bridge to the CIST Root (if the bridge is not the CIST Root) through the Regional Root (if the bridge is not a Regional Root).
    2) A Designated Port provides the least cost path from the attached LAN through the bridge to the CIST Root.
    3) Alternate or Backup Ports provide connectivity if other bridges, Bridge Ports, or LANs fail or are removed.
g) MSTI Port Roles (13.12) that identify the role played by each port on a bridge for each MSTI's active topology within and at the boundaries of a region.
    1) The Root Port provides the minimum cost path from the bridge to the MSTI Regional Root (if the bridge is not the Regional Root for this MSTI).
    2) A Designated Port provides the least cost path from the attached LAN though the bridge to the Regional Root.
    3) A Master Port provides connectivity from the region to a CIST Root that lies outside the region. The Bridge Port that is the CIST Root Port for the CIST Regional Root is the Master Port for all MSTIs.
    4) Alternate or Backup Ports provide connectivity if other bridges, Bridge Ports, or LANs fail or are removed.
h) State machines and state variables associated with each spanning tree (CIST or MSTI), port, and port role, to select and change the Port State (8.4, 13.24) that controls the processing and forwarding of frames assigned to that tree by a MAC Relay Entity (8.3).

13.5.1 Example topologies

Figure 13-6 is an example Bridged Local Area Network, using the conventions of Figure 13-1, and chosen to illustrate MSTP calculations rather than as an example of a common or desirable physical topology.

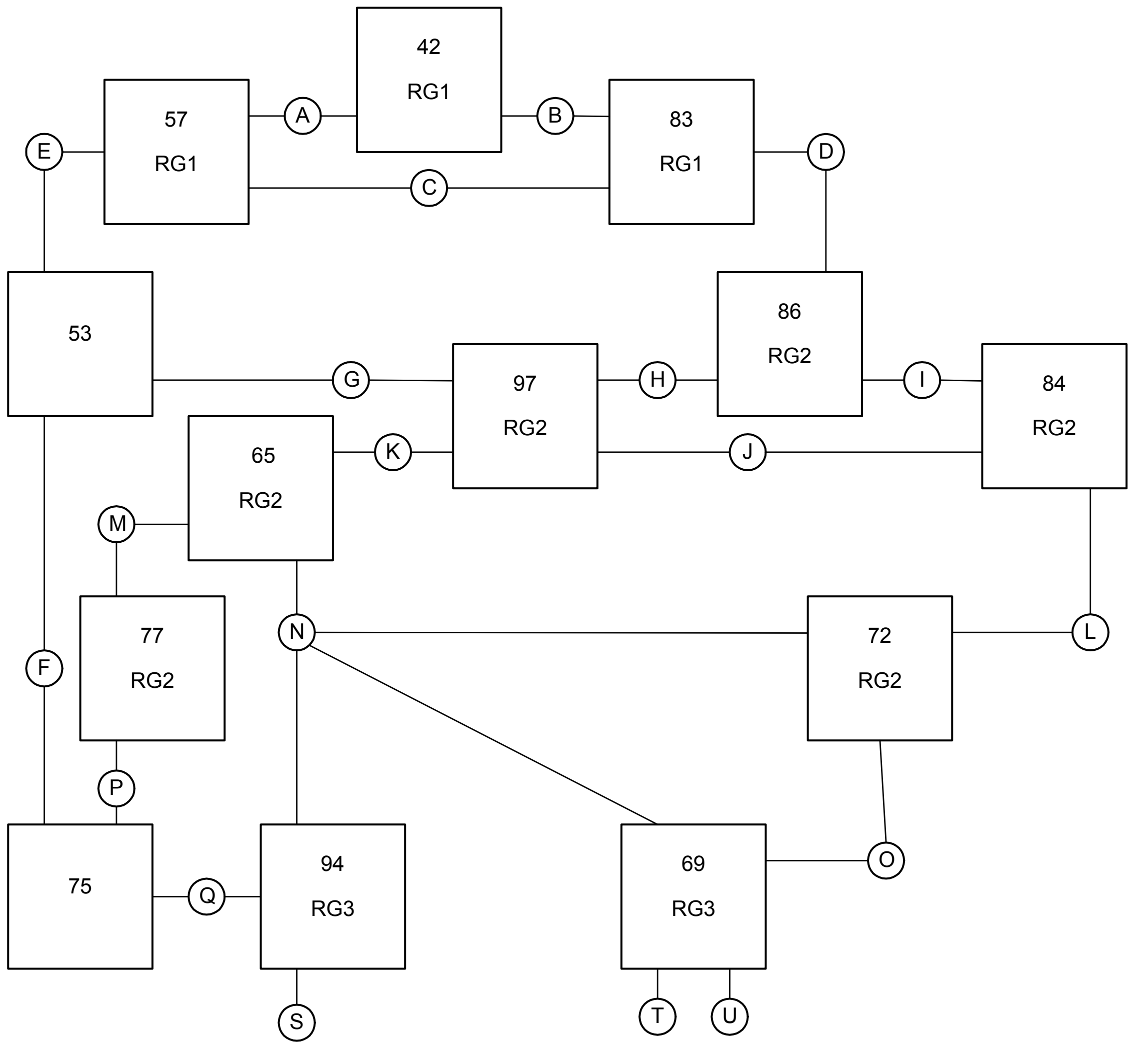


Figure 13-6—An MST Bridge network

Figure 13-8 is the same network showing bridges and LANs with better CIST spanning tree priorities higher on the page, and including CIST priority vectors, port roles, and MST Regions. In this example:

a) Bridge 0.42 is the CIST Root because it has the best (numerically lowest) Bridge Identifier.

b) Bridges 0.57 and 2.83 are in the same MST Region (1) as 0.42, because they have the same MST Configuration Identifier as the latter. Because they are in the same MST Region as the CIST Root, their External Root Path Cost is 0, and their CIST Regional Root is the CIST Root.

c) LANs A, B, C, and D are in Region 1 because their CIST Designated Bridge is a Region 1 MST Bridge, and no STP bridges are attached to those LANs. LAN E is not in an MST Region (or in its own region—an equivalent view) because it is attached to bridge 0.53, which is not an MST Bridge.

d) Bridges 0.77, 0.65, 0.97, 0.86, 3.84, and 3.72 are in the same MST Region (2) since they have the same MST Configuration Identifier and are interconnected by LANs for which one of them is the CIST Designated Bridge.

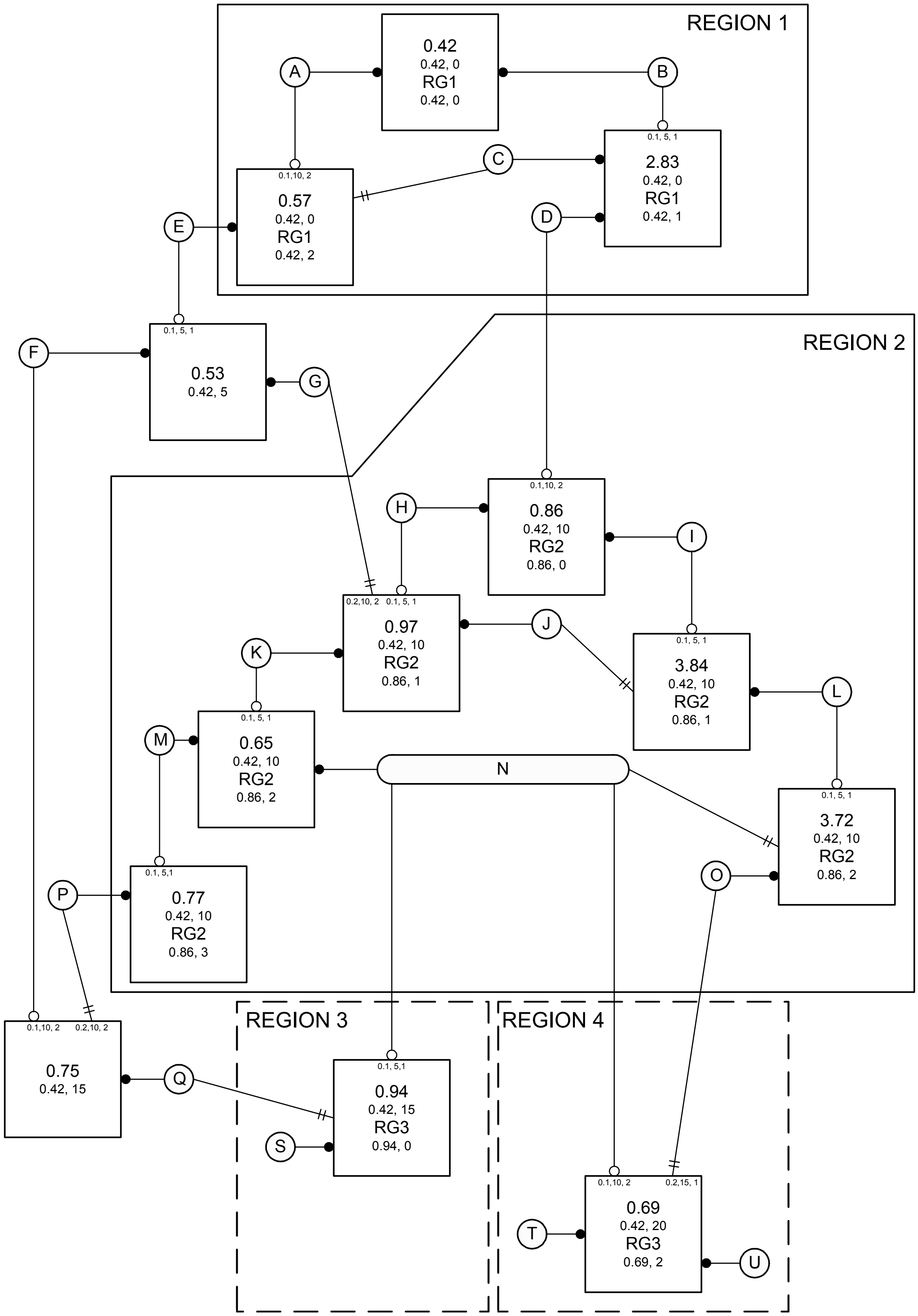


Figure 13-7—CIST Priority Vectors, Port Roles, and MST Regions

e) Bridge 0.86 is the CIST Regional Root for Region 2 because it is has the lowest External Root Path Cost through a Boundary Port.

f) LAN N is in Region 2 because its CIST Designated Bridge is in Region 2. Frames assigned to different MSTIDs may reach N from bridge 0.86 (for example) by either bridge 0.65 or bridge 3.72, even though bridges 0.94 and 0.69 with MST Configuration Identifiers that differ from those for bridges in Region 2 are attached to this shared LAN.

g) Bridges 0.94 and 0.69 are in different regions, even though they have the same MST Configuration Identifier, because the LAN that connects them (N) is in a different region.

Figure 13-8 shows a possible active topology of MSTI 2 within Region 2.

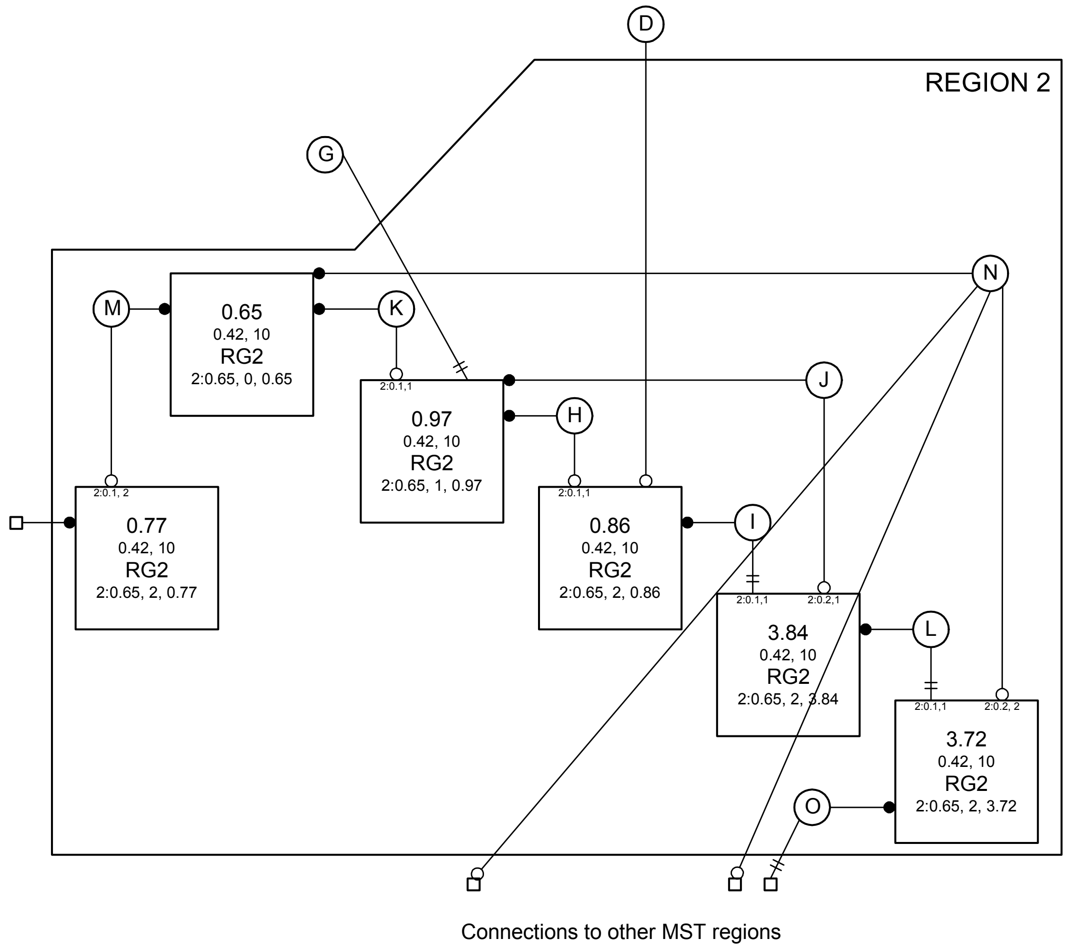


Figure 13-8—MSTI Active Topology in Region 2

h) Bridge 0.65 has been chosen as the MSTI Regional Root because it has the best (numerically the lowest) Bridge Identifier of all bridges in the region for this MSTI.

i) The connectivity between the whole of Region 2 and Region 1 is provided through a single Bridge Port, the Master Port on bridge 0.86. This port was selected for this role because it is the CIST Root Port on the CIST Regional Root for the region (see Figure 13-6).

j) The connectivity between the whole of Region 2 and LANs and bridges outside the region for the MSTI is the same as that for the CIST. This connectivity is similar to that which might result by replacing the entire region by a single SST Bridge. The region has a single Root Port (this port is the Master Port for each MSTI) and a number of Designated Ports.

13.5.2 Relationship of MSTP to RSTP

MSTP is based on RSTP, extended so frames for different VLANs can follow different trees within regions.

a) The same fundamental spanning tree algorithm selects the CIST Root Bridge and Port Roles, but extended priority vector components are used within (13.9, 13.10) in each region. As a result each region resembles a single bridge from the point of view of the CST as calculated by RSTP.
b) Each MSTI's Regional Root Bridge and Port Roles are also computed using the same fundamental spanning tree algorithm with modified priority vector components (13.11).
c) Different bridges may be selected as the Regional Root for different MSTIs by modifying the manageable priority component of the Bridge Identifier differently for the MSTIs.
d) MST Configuration Identification is specific to MSTP.
e) The Port Roles used by the CIST (Root, Designated, Alternate, Backup or Disabled Port) are the same as those of RSTP. The MSTIs use the additional port role Master Port. The Port States associated with each spanning tree and bridge port are the same as those of RSTP.
f) The state variables for each Bridge Port for each tree and for the bridge itself are those specified for RSTP as per bridge port and per bridge with a few exceptions, additions, and enhancements.
g) The performance parameters specified for RSTP apply to the CIST, with a few exceptions, additions, and enhancements. A simplified set of performance parameters apply to the MSTIs.
h) This standard specifies RSTP state machines and procedures as a subset of MSTP.

13.5.3 Modeling an MST or SPT Region as a single bridge

The nominal replacement of an entire region by a single RSTP Bridge leads to little impact on the remainder of the Bridged Local Area Network. This design is intended to assist those familiar with RSTP to comprehend and verify MSTP, and to administer networks using MSTP. Treating the MST Regions as single bridges provides the network administrator with a natural hierarchy. The internal management of MST Regions can be largely separated from the management of the active topology of the network as a whole.

The portion of the active topology of the network that connects any two bridges in the same MST Region traverses only MST Bridges and LANs in that region and never bridges of any kind outside the region; in other words, connectivity within the region is independent of external connectivity. This is because the protocol parameters that determine the active topology of the network as a whole, the Root Identifier and Root Path Cost (known in the MSTP specification as the CIST Root Identifier and CIST External Root Path Cost) are carried unchanged throughout and across the MST Region, so bridges within the region will always prefer spanning tree information that has been propagated within the region to information that has exited the region and is attempting to re-enter it.

NOTE 1—No LAN can be in more than one MST Region at a time, so two bridges (0.11 and 0.22 say) that would otherwise be in the same region by virtue of having the same MST Configuration and of being directly connected by a LAN, may be in distinct regions if that is a shared LAN with other bridges attached (having a different MST Configuration) and no other connectivity between 0.11 and 0.22 and lying wholly within their region is available. The region that the shared LAN belongs to may be dynamically determined. No such dynamic partitioning concerns arise with single bridges. Obviously the sharing of LANs between administrative regions militates against the partitioning of concerns and should only be done following careful analysis.

The Port Path Cost (MSTP's External Port Path Cost) is added to the Root Path Cost just once at the Root Port of the CIST Regional Root, the closest bridge in the region to the Root Bridge of the entire network. The Message Age used by STP and RSTP is also only incremented at this port. If the CIST Root is within an MST Region, it also acts as the Regional Root, and the Root Path Cost and Message Age advertised are zero, just as for a single bridge.

Within an MST Region, each MSTI operates in much the same way as an independent instance of RSTP with dedicated Regional Root Identifier, Internal Root Path Cost, and Internal Port Path Cost parameters.

Moreover, the overall spanning tree (the CIST) includes a fragment (the IST) within each MST Region that can be viewed as operating in the same way as an MSTI with the Regional Root as its root.

NOTE 2—Since an MST Region behaves like a single bridge and does not partition (except in the unusual configuration involving shared LANs noted above), it has a single Root Port in the CST active topology. Partitioning a network into two or more regions can therefore force non-optimal blocking of Bridge Ports at the boundaries of those regions.

13.6 SPB overview

Clause 27 provides a comprehensive overview of SPB and SPBB operation. This clause (13.6) summarizes aspects of SPB operation that relate to the transmission and reception of BPDUs, and their role in providing interoperability with MSTP, RSTP, and MSTP and in carrying the Tree Agreement Protocol (TAP) messages that ensure that each Shortest Path Tree (SPT) provides loop-free connectivity throughout an SPT Region. The details of TAP are considered in 13.17, the assignment of frames to SPTs in 8.4 and ***Clause 27***, and protocols and procedures to allow plug-and-play generation of SPVIDs in ***Clause 27***.

Considerations of backward and forward compatibility and interoperability figure largely in this clause (Clause 13) and to some extent these consideration revolve around the notion of MST or SPT Region, with each region capable of using different protocols or different configurations. These considerations should not be allowed to obscure the fact that the ideal network configuration (from the point of view of connectivity and bandwidth efficiency) comprises a single region, or possibly one core region with RSTP bridges attached, and that separate regions are most likely to arise when continued connectivity is being provided by the CST as configuration changes are made to bridges in the network. If an SPT Region is bounded by Backbone Edge Bridges or LAN with no other bridges attached, B-VLANs can be supported by SPBB, with frames assigned to SPTs using their source address, while other B-VLANs and S-VLANs can be supported by SPB, with frames assigned to each SPT by SPVID. In both cases, however, SPTs are calculated in the same way and the effects on this clause (Clause 13) are limited to allowing loop mitigation (***6.5.4.2***) for unicast frames supported by SPBB as well as loop prevention (***6.5.4.1***). The need for unicast multicast congruence (***3.2***), the simplification possible by not introducing differences between SPBB unicast forwarding and the other uses of SPTs, and the need for source address lookup to support loop mitigation means that all shortest path bridged frames are assigned to an SPT rooted at the source bridge and follow the standard bridging paradigm of multicast distribution with filtering, even when the frame has a single destination and the path to the destination is known through the use of routing protocol (ISIS-SPB).

SPT Bridges send and receive BPDUs in order to advertise their presence to, and recognize the presence of, other bridges. The requirement for loop-free SPT connectivity also means that SPT Bridges have to communicate with their nearest neighbors, when calculations that follow the reception of ISIS-SPB messages have been completed rather than as a part of the ISIS protocol itself. The Tree Agreement Protocol (TAP, 13.17) that satisfies this requirement can be separated conceptually from other uses of BPDUs, but the opportunity to share common information elements, synchronization of neighbor state, and an overall reduction in the number of protocol frames sent means that it is conveniently carried in BPDUs and its specification in terms of protocol variables, procedures, and state machines is integrated with the other provisions of this clause (13.24–13.40).

SPT Bridges use the configuration and active topology management parameters (Bridge Identifiers, Port Path Costs) already required for the CIST. They also retain MSTP capabilities as a subset of their operation, though it is always possible to configure zero MSTIs.

13.7 Compatibility and interoperability

RSTP, MSTP, and the SPB protocols are designed to interoperate with each other and with STP. This clause (13.7) reviews aspects of their design that are important to meeting that requirement. The SPB protocols and

SPT BPDUs include the functionality provided by MSTP and MST BPDUs, so the compatibility with RSTP and STP provided by the latter extends to SPB.

13.7.1 Designated Port selection

Correct operation of the spanning tree protocols requires that all Bridge Ports attached to any given LAN agree on a single CIST Designated Port after a short interval sufficient for any Bridge Port to receive a configuration message from that Designated Port.

A unique spanning tree priority (13.9) is required for each Bridge Port for STP, which has no other way of communicating port roles. Since port numbers on different bridges are not guaranteed to be unique, this necessitates the inclusion of the transmitting bridge's Bridge Identifier in the STP BPDU. RSTP and MSTP's Port Protocol Migration state machines (13.32) ensure that all bridges attached to any LAN with an attached STP bridge send and receive STP BPDUs exclusively.

NOTE 1—This behavior satisfies the requirement for unique, agreed Designated Port for LANs with attached STP bridges, but means that an MST Region cannot completely emulate a single bridge since the transmitted Designated Bridge Identifier can differ on Bridge Ports at the region's boundary.

MSTP transmits and receives the Regional Root Identifier and not the Designated Bridge Identifier in the BPDU fields recognized by RSTP (14.9) to allow both the MSTP and the RSTP Bridges potentially connected to a single LAN to perform comparisons (13.9, 13.10) between all spanning tree priority vectors transmitted that yield a single conclusion as to which RSTP Bridge or MST Region includes the Designated Port. MST and RST BPDUs convey the transmitting port's CIST Port Role. This is checked on receipt by RSTP when receiving messages from a Designated Bridge (17.21.8 of IEEE Std 802.1D), thus ensuring that an RSTP Bridge does not incorrectly identify one MST Bridge Port as being Designated rather than another, even while omitting the competing Bridge Ports' Designated Bridge Identifiers from comparisons.

NOTE 2—This ability of MSTP Bridges to communicate the full set of MSTP information on shared LANs to which RSTP Bridges are attached avoids the need for the Port Protocol Migration machines to detect RSTP Bridges. Two or more MSTP and one or more RSTP Bridges may be connected to a shared LAN, with full MSTP operation. This includes the possibility of different MSTI Designated Ports (see 13.5.1).

13.7.2 Force Protocol Version

A Force Protocol Version parameter, controlled by management, permits emulation of aspects of the behavior of earlier versions of spanning tree protocol that are not strictly required for interoperability. The value of this parameter applies to all Bridge Ports.

a) STP BPDUs, rather than MST BPDUs, are transmitted if Force Protocol Version is 0. RST BPDUs omit the MST Configuration Identifier and all MSTI Information.
b) RST BPDUs, rather than MST BPDUs, are transmitted if Force Protocol Version is 2. RST BPDUs omit the MST Configuration Identifier and all MSTI Information.
c) All received BPDUs are treated as being from a different MST Region if Force Protocol Version is 0 or 2.
d) Rapid transitions are disabled if Force Protocol Version is 0. This allows MSTP Bridges to support applications and protocols that can be sensitive to the increased rates of frame duplication and misordering that can arise under some circumstances, as discussed in Annex K of IEEE Std 802.1D.
e) The MSTP state machines allow full MSTP behavior if Force Protocol Version is 3 or more.
f) SPT BPDUs are transmitted if Force Protocol Version is 4 or more.

NOTE— Force Protocol Version does not support multiple spanning trees with rapid transitions disabled.

13.8 MST Configuration Identifier

It is essential that all bridges within an MST or SPT Region agree on the allocation of VIDs to spanning trees. If the allocation differs, frames for some VIDs may be duplicated or not delivered to some LANs at all. MST and SPT Bridges check that they are allocating VIDs to the same spanning trees as their neighbors in the same region by transmitting and receiving MST Configuration Identifiers in BPDUs. Each MST Configuration Identifier includes a Configuration Digest that is compact but designed so that two matching identifiers have a very high probability of denoting the same allocation of VIDs to MSTIDs (***3.n, 8.4***) even if the identifiers are not explicitly managed. Suitable management practices for equipment deployment and for choosing Configuration Names and Revision Levels (see below) can guarantee that the identifiers will differ if the VID to tree allocation differs within a single administrative domain.

An MST or SPT Region comprises one or more MST or SPT Bridges with the same MST Configuration Identifiers, interconnected by and including LANs for which one of those bridges is the Designated Bridge for the CIST and which have no bridges attached that cannot receive and transmit RST BPDUs.

SPT BPDUs are a superset of MST BPDUs received and validated by MST Bridges as if they were MST BPDUs, so MSTP operates within an SPT Region just as if it were an MST Region. However, each SPT Set is represented by a reserved MSTID value that is included in the Configuration Digest, so when shortest path bridging is used an SPT Region contains only SPT Bridges.

Each MST Configuration Identifier contains the following components:

1) A Configuration Identifier Format Selector, the value 0 encoded in a fixed field of one octet to indicate the use of the following components as specified in this standard.
2) The Configuration Name, a variable length text string encoded within a fixed field of 32 octets, conforming to RFC 2271’s definition of SnmpAdminString. If the Configuration Name is less than 32 characters, the text string should be terminated by the NUL character, with the remainder of the 32-octet field filled with NUL characters. Otherwise the text string is encoded with no terminating NUL character.
3) The Revision Level, an unsigned integer encoded within a fixed field of 2 octets.
4) The Configuration Digest, a 16-octet signature of type HMAC-MD5 (see IETF RFC 2104) created from the MST Configuration Table (3.86, 8.9). To calculate the digest, the table is considered to contain 4096 consecutive two octet elements, where each element of the table (with the exception of the first and last) contains an MSTID value encoded as a binary number, with the first octet being most significant. The first element of the table contains the value 0, the second element the MSTID value corresponding to VID 1, the third element the MSTID value corresponding to VID 2, and so on, with the next to last element of the table containing the MSTID value corresponding to VID 4094, and the last element containing the value 0. The key used to generate the signature consists of the 16-octet string specified in Table 13-1.

Table 13-1—Configuration Digest Signature Key

Parameter	Mandatory value
Configuration Digest Signature Key	0x13AC06A62E47FD51F95D2BA243CD0346

NOTE—The formulation of the signature as described above does not imply that a separate VID to MSTID translation table has to be maintained by the implementation; rather that it should be possible for the implementation to derive the logical contents of such a table, and the signature value as specified above, from the other configuration information maintained by the implementation, as described in Clause 12.

The Configuration Digests of some VID to MSTID translations are shown in Table 13-2 to help verify implementations of this specification.

Table 13-2—Sample Configuration Digest Signature Keys

VID to MSTID translation	Configuration Digest
All VIDs map to the CIST, no VID mapped to any MSTI	0xAC36177F50283CD4B83821D8AB26DE62
All VIDs map to MSTID 1	0xE13A80F11ED0856ACD4EE3476941C73B
Every VID maps to the MSTID equal to (VID modulo 32) + 1	0x9D145C267DBE9FB5D893441BE3BA08CE

It is recommended that MST and SPT Bridge implementations provide an easily selectable or default configuration comprising a Configuration Name of the Bridge Address as a text string using the Hexadecimal Representation specified in IEEE Std 802, a Revision Level of 0, and a Configuration Digest representing a VID to MSTID translation table containing the value 0 for every element. Such a table represents the mapping of all VLANs to the CIST. Since the Bridge Address is unique to each bridge, no two bridges using this default configuration will be identified as belonging to the same region.

13.9 Spanning Tree Priority Vectors

Priority vectors permit concise specification of each protocol's computation of the active topology, both in terms of the entire network and of the operation of individual bridges in support of the distributed algorithm. MST, RST, and STP bridges use *spanning tree priority vector* information in Configuration Messages (13.14), sent and received from neighbouring bridges, to assign Port Roles that determine each port's participation in a fully and simply connected active topology based on one or more spanning trees. SPT Bridges use ISS-SPB to disseminate the information necessary to calculate Port Roles throughout SPT Regions, and to perform those calculations, but also implement a Tree Agreement Protocol (TAP) that uses this Configuration Message information for the CIST to ensure that neighbouring bridges agree on that active topology, and to receive the information if the CIST Root lies outside their SPT Region.

CIST priority vectors comprise the following components:

a) CIST Root Identifier, the Bridge Identifier of the CIST Root;
b) CIST External Root Path Cost, the inter-regional cost from the transmitting bridge to the CIST Root;
c) CIST Regional Root Identifier, the Bridge Identifier of the single bridge in a region whose CIST Root Port connects to a LAN in a different region, or of the CIST Root if that is within the region;
d) CIST Internal Root Path Cost, the cost to the CIST Regional Root;
e) CIST Designated Bridge Identifier, the Bridge Identifier for the transmitting bridge for the CIST;
f) CIST Designated Port Identifier, the Port Identifier for the transmitting port for the CIST;
g) CIST Receiving Port Identifier (not conveyed in Configuration Messages, used as tie-breaker between otherwise equal priority vectors within a receiving bridge).

The first two components of the CIST priority vector are significant throughout the network. The CIST External Root Path Cost transmitted by a bridge is propagated along each path from the CIST Root, is added to at Bridge Ports that receive the priority vector from a bridge in a different region, and thus accumulates costs at the Root Ports of bridges that are not MST or SPT Bridges or are CIST Regional Roots and is constant within a region. The CIST Internal Root Path Cost is only significant and defined within a region. The last three components are used as locally significant tie breakers, not propagated within or between regions. The set of all CIST spanning tree priority vectors is thus totally ordered.

Since RSTP is not aware of regions, RSTP specifications also refer to the CIST Root Identifier and CIST External Root Path Cost simply as the Root Bridge Identifier and Root Path Cost, respectively, and omit the CIST Internal Root Path Cost (as does STP). MSTP encodes the CIST Regional Root Identifier in the BPDU field used by RSTP to convey the Designated Bridge Identifier (14.5), so an entire region appears to an RSTP capable bridge as a single bridge. RSTP's CST use of CIST priority vectors can be conveniently specified by the use of the zero for the Internal Root Path Cost and the same values for both the Regional Root Identifier and Designated Bridge Identifier.

NOTE 1—The path to the CIST Root from a bridge with a CIST Root Port within a region always goes to or through the CIST Regional Root.

NOTE 2—STP lacks the fields necessary for MST Bridges to communicate the Designated Bridge Identifier to resolve a potential priority vector tie, and MSTP BPDUs are not sent on a LAN to which an STP bridge is attached.

Each MSTI priority vector comprises the following components for the particular MSTI in a given region:

h) MSTI Regional Root Identifier, the Bridge Identifier of the MSTI Regional Root;
i) MSTI Internal Root Path Cost, the path cost to the MSTI Regional Root;
j) MSTI Designated Bridge Identifier, the Bridge Identifier for the transmitting bridge for this MSTI;
k) MSTI Designated Port Identifier, the Port Identifier for the transmitting port for this MSTI;
l) MSTI Receiving Port Identifier (not conveyed in Configuration Messages).

The set of priority vectors for a given MSTI is only defined within a region. Within each region they are totally and uniquely ordered. A CIST Root Identifier, CIST External Root Path Cost, and CIST Regional Root Identifier tuple defines the connection of the region to the external CST and is required to be associated with the source of the MSTI priority vector information when assessing the agreement of information for rapid transitions to forwarding, but plays no part in priority vector calculations.

As each bridge and Bridge Port receives priority vector information from bridges and ports closer to the Root, calculations and comparisons are made to decide which priority vectors to record, and what information to pass on. Decisions about a given port's role are made by comparing the priority vector components that could be transmitted with that received by the port. For all components, a lesser numerical value is better, and earlier components in the above lists are more significant. As each Bridge Port receives information from ports closer to the Root, additions are made to one or more priority vector components to yield a worse priority vector for potential transmission through other ports of the same bridge.

NOTE 3—The consistent use of lower numerical values to indicate better information is deliberate as the Designated Port that is closest to the Root Bridge, i.e., has a numerically lowest path cost component, is selected from among potential alternatives for any given LAN (13.9). Adopting the conventions that lower numerical values indicate better information, that where possible more significant priority components are encoded earlier in the octet sequence of a BPDU (14.3), and that earlier octets in the encoding of individual components are more significant (14.2) allow concatenated octets that compose a priority vector to be compared as if they were a multiple octet encoding of a single number, without regard to the boundaries between the encoded components. To reduce the confusion that naturally arises from having the lesser of two numerical values represent the better of the two, i.e., that chosen all other factors being equal, this clause uses the following consistent terminology. Relative numeric values are described as "least," "lesser," "equal," and "greater," and their comparisons as "less than," "equal to," or "greater than," while relative Spanning Tree priorities are described as "best," "better," "the same," "different," and "worse" and their comparisons as "better than," "the same as," "different from," and "worse than." The operators "<" and "=" represent less than and equal to, respectively. The terms "superior" and "inferior" are used for comparisons that are not simply based on priority but include the fact that a priority vector can replace an earlier vector transmitted by the same Bridge Port. All of these terms are defined for priority vectors in terms of the numeric comparison of components below (13.10, 13.11).

NOTE 4—To ensure that the CIST and each MSTI's view of the boundaries of each region remain in synchronization at all times, each BPDU carries priority vector information for the CIST as well as for MSTIs. Associating the CIST Root Identifier, External Path Cost, and Regional Root Identifier with the priority vector information for each MSTI does not therefore raise a requirement to transmit these components separately. A single bit per MSTI vector, the Agreement flag, satisfies the requirement to indicate that the vector beginning with the MSTI Regional Root Identifier for that specific MSTI has always been associated with the single CIST Root Identifier, etc. transmitted in the BPDU.

To allow the active topology to be managed for each tree through adjusting the relative priority of different bridges and Bridge Ports for selection as the CIST Root, a CIST or MSTI Regional Root, Designated Bridge, or Designated Port, the priority component of the bridge's Bridge Identifier can be independently chosen for the CIST and for each MSTI. The priority component used by the CIST for its CIST Regional Root Identifier can also be chosen independently of that used for the CIST Root Identifier. Independent configuration of Port Path Cost and Port Priority values for the CIST and for each MSTI can also be used to control selection of the various roles for the CIST and for each MSTI.

In principle an SPT priority vector could be defined within an SPT Region, comprising a Regional Root Identifier and Internal Root Path Cost, and reflects the construction of each SPT (lowest Internal Root Path Cost from each bridge and LAN to the Regional Root). However the set of such vectors cannot be totally ordered by the addition of purely local tie-breaker components: as each SPT Set has to be symmetric. Clause 29 specifies ISIS-SPB's calculation of SPTs.

13.10 CIST Priority Vector calculations

The *port priority vector* is the priority vector held for the port when the reception of BPDUs and any pending update of information has been completed:

port priority vector = *{RootID : ExtRootPathCost : RRootID : IntRootPathCost : DesignatedBridgeID : DesignatedPortID : RcvPortID}*

The *message priority vector* is the priority vector conveyed in a received Configuration Message. For a bridge with Bridge Identifier *B* receiving a Configuration Message on a port P_B from a Designated Port P_D on bridge *D* claiming a CIST Root Identifier of $R_{D,}$ a CIST External Root Path Cost of ERC_D, a CIST Regional Root Identifier of $RR_{D,}$ and a CIST Internal Root Path Cost of IRC_D:

$$message\ priority\ vector = \{R_D : ERC_D : RR_D : IRC_D : D : P_D : P_B\}$$

If *B* is not in the same region as *D*, the Internal Root Path Cost has no meaning to *B* and is set to 0.

NOTE—If a Configuration Message is received in an RST or STP BPDU, both the Regional Root Identifier and the Designated Bridge Identifier are decoded from the single BPDU field used for the Designated Bridge Parameter (the MST BPDU field in this position encodes the CIST Regional Root Identifier). An STP or RSTP bridge is always treated by MSTP as being in an region of its own, so the Internal Root Path Cost is decoded as zero.

The received CIST message priority vector is the same as B's port priority vector if:

$$(R_D == RootID)\ \&\&\ (ERC_D == ExtRootPathCost)\ \&\&\ (RR_D == RRootID)\ \&\&$$
$$(IRC_D == IntRootPathCost)\ \&\&\ (D == DesignatedBridgeID)\ \&\&\ (P_D == DesignatedPortID)$$

and is better if:

$$((R_D < RootID))\ ||$$
$$((R_D == RootID)\ \&\&\ (ERC_D < ExtRootPathCost))\ ||$$
$$((R_D == RootID)\ \&\&\ (ERC_D == ExtRootPathCost)\ \&\&\ (RR_D < RRootID))\ ||$$
$$((R_D == RootID)\ \&\&\ (ERC_D == ExtRootPathCost)\ \&\&\ (RR_D == RRootID)$$
$$\&\&\ (IRC_D < IntRootPathCost))\ ||$$
$$((R_D == RootID)\ \&\&\ (ERC_D == ExtRootPathCost)\ \&\&\ (RR_D == RRootID)$$
$$\&\&\ (IRC_D == IntRootPathCost)\ \&\&\ (D < DesignatedBridgeID))\ ||$$
$$((R_D == RootID)\ \&\&\ (ERC_D == ExtRootPathCost)\ \&\&\ (RR_D == RRootID)$$
$$\&\&\ (IRC_D == IntRootPathCost)\ \&\&\ (D == DesignatedBridgeID)$$
$$\&\&\ (P_D < DesignatedPortID))$$

A received CIST message priority vector is superior to the port priority vector if, and only if, the message priority vector is better than the port priority vector, or the Designated Bridge Identifier Bridge Address and Designated Port Identifier Port Number components are the same; in which case, the message has been transmitted from the same Designated Port as a previously received superior message, i.e., if:

$$(\{R_D : ERC_D : RR_D : IRC_D : D : P_D : P_B\}$$
$$\text{is better than}$$
$$\{RootID : ExtRootPathCost : RRootID : IntRootPathCost :$$
$$DesignatedBridgeID : DesignatedPortID : RcvPortID\}$$
$$)\,||\, ((D.BridgeAddress == DesignatedBridgeID.BridgeAddress)\ \&\&$$
$$(P_D.PortNumber == DesignatedPortID.PortNumber))$$

If the message priority vector received in a Configuration Message from a Designated Port is superior, it will replace the current port priority vector.

A *root path priority vector* for a port can be calculated from a port priority vector that contains information from a message priority vector, as follows.

If the port priority vector was received from a bridge in a different region (13.26.4), the External Port Path Cost EPC_{PB} is added to the External Root Path Cost component, and the Regional Root Identifier is set to the value of the Bridge Identifier for the receiving bridge. The Internal Root Path Cost component will have been set to zero on reception.

$$\text{root path priority vector} = \{R_D : ERC_D + EPC_{PB} : B : 0 : D : P_D : P_B)$$

If the port priority vector was received from a bridge in the same region (13.26.4), the Internal Port Path Cost IPC_{PB} is added to the Internal Root Path Cost component.

$$\text{root path priority vector} = \{R_D : ERC_D : RR_D : IRC_D + IPC_{PB} : D : P_D : P_B)$$

The *bridge priority vector* for a bridge *B* is the priority vector that would, with the Designated Port Identifier set equal to the transmitting Port Identifier, be used as the message priority vector in Configuration Messages transmitted on bridge *B's* Designated Ports if *B* was selected as the Root Bridge of the CIST.

$$\text{bridge priority vector} = \{B : 0 : B : 0 : B : 0 : 0\}$$

The *root priority vector* for bridge *B* is the best priority vector of the set of priority vectors comprising:

a) the bridge priority vector; plus
b) all root path priority vectors that:
   1) have a Designated Bridge Identifier *D* that is not equal to *B*, and
   2) were received from a Bridge Port attached to a LAN that is not in the same SPT Region as *B*; plus
c) the root path priority vector calculated by ISIS-SPB (if SPB is enabled, and the attached LAN is within the bridge's SPT Region).

NOTE—The BPDUs sent and received by all bridges attached to a LAN allow MST and SPT Bridges to determine whether each attached LAN is within their region independently of priority vector values. SPT Bridges take advantage of this fact by using ISIS-SPB to communicate the CST priority vector for each of SPT Region's potential Master Ports throughout the region, at the same time as ISIS-SPB calculates the Port Roles for each SPT (see ***Clause 29***).

If the bridge priority vector is the best of this set of priority vectors, Bridge *B* has been selected as the CIST Root. Otherwise the root priority vector will only be that calculated by ISIS-SPB if the Root Bridge or Regional Root is within the bridge's SPT Region.

The *designated priority vector* for a port Q on bridge B is the root priority vector with B*'s* Bridge Identifier B substituted for the *DesignatedBridgeID* and Q*'s* Port Identifier Q_B substituted for the *DesignatedPortID* and *RcvPortID* components. If Q is attached to a LAN that has one or more STP bridges attached (as determined by the Port Protocol Migration state machine), B*'s* Bridge Identifier B is also substituted for the the *RRootID* component.

If the designated priority vector is better than the port priority vector and the LAN attached to the port is not within the bridge's SPT Region (possibly because SPB is not enabled), the port will be the Designated Port for that LAN and the current port priority vector will be updated. If the attached LAN is within the bridge's SPT Region, then the Port Role is as calculated by ISIS-SPB and the port priority vector will be updated with the designated priority vector if and only if the port is a Designated Port. The message priority vector in Configuration Messages transmitted by a port always comprises the components of the designated priority vector for the port, even if the port is a Root Port.

13.11 MST Priority Vector calculations

The *port priority vector* is the priority vector held for the port when the reception of BPDUs and any pending update of information has been completed:

$$\textit{port priority vector} = \{RRootID : IntRootPathCost : DesignatedBridgeID : DesignatedPortID : RcvPortID\}$$

The *message priority vector* is the priority vector conveyed in a received Configuration Message. For a bridge with Bridge Identifier B receiving a Configuration Message on a Regional Port P_B from a Designated Port P_D on bridge D belonging to the same MST Region and claiming an Internal Root Path Cost of IRC_D:

$$\textit{message priority vector} = \{RR_D : IRC_D : D : P_D : P_B\}$$

An MSTI message priority vector received from a bridge not in the same MST Region is discarded.

An MSTI message priority vector received from a Bridge Port internal to the region is the same as the port priority vector if:

$$((RR_D == RRootID)\ \&\&\ (IRC_D == IntRootPathCost)\ \&\&\ (D == DesignatedBridgeID)\ \&\&\ (P_D == DesignatedPortID))$$

and is better if:

$$\begin{aligned}
&((RR_D < RRootID))\ \|\\
&((RR_D == RRootID)\ \&\&\ (IRC_D < IntRootPathCost))\ \|\\
&((RR_D == RRootID)\ \&\&\ (IRC_D == IntRootPathCost)\ \&\&\ (D < DesignatedBridgeID))\ \|\\
&((RR_D == RRootID)\ \&\&\ (IRC_D == IntRootPathCost)\ \&\&\ (D == DesignatedBridgeID)\\
&\qquad \&\&\ (P_D < DesignatedPortID))
\end{aligned}$$

An MSTI message priority vector is superior to the port priority vector if, and only if, the message priority vector is better than the port priority vector, or the Designated Bridge Identifier Bridge Address and Designated Port Identifier Port Number components are the same; in which case, the message has been transmitted from the same Designated Port as a previously received superior message, i.e., if:

$$\begin{aligned}
&(\{RR_D : IRC_D : D : P_D : P_B\}\\
&\qquad \textit{is better than}\\
&\{RRootID : IntRootPathCost : DesignatedBridgeID : DesignatedPortID : RcvPortID\}
\end{aligned}$$

$) \; || \; ((D.BridgeAddress == DesignatedBridgeID.BridgeAddress) \; \&\&$
$(P_D.PortNumber == DesignatedPortID.PortNumber))$

If the message priority vector received in a Configuration Message from a Designated Port for the MSTI is superior, it will replace the current port priority vector.

NOTE 1—The agree flag (13.27.4) for the port and this MSTI will be cleared if the CIST Root Identifier, CIST External Root Path Cost, and CIST Regional Root Identifier in the received BPDU are not better than or the same as those for the CIST designated priority vector for the port following processing of the received BPDU.

A *root path priority vector* for a given MSTI can be calculated for a port that has received a port priority vector from a bridge in the same region by adding the Internal Port Path Cost IPC_{PB} to the Internal Root Path Cost component.

$$root\ path\ priority\ vector = \{RR_D : IRC_D + IPC_{PB} : D : P_D : P_B)$$

NOTE 2—Internal Port Path Costs are independently manageable for each MSTI, as are the priority components of the Bridge and Port Identifiers. The ability to independently manage the topology of each MSTI without transmitting individual Port Path Costs is a key reason for retaining the use of a Distance Vector protocol for constructing MSTIs. A simple Link State Protocol requires transmission (or *a priori* sharing) of all Port Costs for all links.

The *bridge priority vector* for a bridge *B* is the priority vector that would, with the Designated Port Identifier set equal to the transmitting Port Identifier, be used as the message priority vector in Configuration Messages transmitted on bridge *B's* Designated Ports if *B* was selected as the Root Bridge of a given tree.

$$bridge\ priority\ vector = \{B : 0 : B : 0\}$$

The *root priority vector* for bridge *B* is the best priority vector of the set of priority vectors comprising the bridge priority vector plus all root path priority vectors whose Designated Bridge Identifier *D* is not equal to *B*. If the bridge priority vector is the best of this set of priority vectors, Bridge *B* has been selected as the Root of the tree.

The *designated priority vector* for a port *Q* on bridge *B* is the root priority vector with *B's* Bridge Identifier *B* substituted for the *DesignatedBridgeID* and *Q's* Port Identifier Q_B substituted for the *DesignatedPortID* and *RcvPortID* components.

If the designated priority vector is better than the port priority vector, the port will be the Designated Port for the attached LAN and the current port priority vector will be updated. The message priority vector in MSTP BPDUs transmitted by a port always comprises the components of the designated priority vector of the port, even if the port is a Root Port.

Figure 13-8 shows the priority vectors and the active topology calculated for an MSTI in a region of the example network of Figure 13-6.

13.12 Port Role assignments

Each bridge assigns CIST Port Roles (when new information becomes available as specified in this clause, 13.12) before assigning MSTI or SPT Port Roles. The calculations specified in 13.10 are used to assign a role to each Bridge Port that is enabled as follows:

a) If the bridge is not the CIST Root, the source of the root priority vector is the Root Port.
b) Each port whose port priority vector is the designated priority vector is a Designated Port.
c) Each port, other than the Root Port, with a port priority vector received from another bridge is a Alternate Port.

d) Each port with a port priority vector received from another port on this bridge is a Backup Port.

If the port is not enabled, i.e. its MAC_Operational status is FALSE or its Administrative Bridge Port state is Disabled (***8.4***), it is assigned the Disabled Port role for the CIST, all MSTIs, and all SPTs, to identify it as having no part in the operation of any of the spanning trees or the active topology of the network.

If the bridge is an MST or SPT Bridge, the calculations specified in 13.11 are used to assign a role to each enabled Bridge Port for each MSTI as follows:

e) If the port is the CIST Root Port and the CIST port priority vector was received from a bridge in another MST or SPT Region, the port is the Master Port.
f) If the bridge is not the MSTI Regional Root, the port that is the source of the MSTI root priority vector is the Root Port.
g) Each port whose port priority vector is the designated priority vector derived from the root priority vector is a Designated Port.
h) Each port, other than the Master Port or the Root Port, with a port priority vector received from another bridge or a CIST port priority vector from a bridge in another region, is an Alternate Port.
i) Each port that has a port priority vector that has been received from another port on this bridge is a Backup Port.

Independently of priority vector values and active topology calculations, each SPT Bridge Port determines from received BPDUs whether all the bridges attached to its LAN are in the same SPT Region. If not, the port is a Boundary Port, and its role for each SPT is determined by its CIST Port Role as follows:

j) If the port is the CIST Root Port, the port is the Master Port for all SPTs.
k) If the port is not the CIST Root Port, the port's role is the same as that for the CIST.

By excluding Boundary Ports from the physical topology used to calculate SPTs, and adopting CIST connectivity at those ports, ISIS-SPB ensures that the use of SPVIDs (rather than each VLAN's Base VID) is not required on shared media LANs attached to bridges in other regions.

SPT Bridges use ISIS-SPB to assign Port Roles for each SPT to non-Boundary Ports as follows:

l) If the bridge is not the SPT Root Bridge, the port that ISIS-SPB has calculated as providing the path for frames assigned to the SPT and forwarded to the bridge from that SPT Root is the Root Port.
m) Each port, other than the Root Port, that ISIS-SPB has calculated as providing a path for frames forwarded from the SPT Root to the attached LAN is a Designated Port.
n) Each port that is attached to the same LAN as a Designated Port for the SPT is a Backup Port.
o) Each port not assigned a Root, Designated, or Backup Port role is an Alternate Port.

13.13 Stable connectivity

This clause provides an analysis to show that RSTP, MSTP, and SPB protocols meets the goal of providing full and simple connectivity for frames assigned to any given VLAN in a stable network, i.e., where the physical topology has remained constant for long enough that the spanning tree information communicated and processed by bridges is not changing.

NOTE 1—The FDB can be configured to prevent connectivity, in particular this analysis assumes that every Bridge Port is a member of every VLAN's Member Set (8.8.9). Spanning tree protocol controls can also be used to prevent new connectivity (to allow for upgrades), or to disallow certain topologies (restricting the location of the CIST Root, for example). This analysis assumes that those controls are not being used, that all the bridges are using conformant protocol implementations and that the LANs are providing omnidirectional connectivity.

Every LAN provides connectivity for all frames between all attached Bridge Ports. Every bridge provides connectivity between and only between its CIST Root and Designated Ports for frames assigned to the CIST, between the Root, Designated, and Master Ports for a given MSTI for frames assigned to that MSTI, and between Root and Designated Ports for a given SPT for frames assigned to that SPT. Any given bridge does not assign frames to more than one tree and has one Root Port per tree, unless it is the Root of that tree.

Every LAN has one and only one CIST Designated Port, and every bridge apart from the CIST Root has one and only one CIST Root Port. The CIST spanning tree priority vector of the Designated Port attached to the LAN that is a bridge's Root Port is better than of any Designated Port of that bridge. The CIST thus connects all bridges and LANs (is "spanning") and loop free (is a "tree").

Each MST or SPT Region is bounded by CST Root and Alternate Ports. At the CST Root Ports connectivity for frames assigned to MSTIs or SPTs within the connected regions is the same as that for the CIST. Every region apart from that containing the CIST Root has a single CST Root Port, identified as the Master Port for each MSTI. The CIST spanning tree priority vector of the LAN attached to the region's CST Root Port is better than that of any CST Designated Port of a bridge in the region attached to a LAN also attached to the CST Root Port of another region. The CST thus provides loop free connectivity between all regions.

NOTE—The term "Common Spanning Tree (CST)" refers to the CIST connectivity between regions, and the term "Internal Spanning Tree (IST)" to the CIST connectivity within each region. An RSTP bridge and the LANs for which it is the Designated Bridge are conveniently considered as forming an MST region of limited extent.

Within each region each frame is consistently assigned to the CIST, an MSTI, or an SPT, and each of these spanning trees provides full loop-free connectivity to each of the bridges within the region, just as the CIST does for the network as a whole, including connectivity between the CST Root Port (Master Port) and the CST Designated Ports. Since each bridge or LAN is in one and only one region, and it has already been shown that loop free connectivity is provided between regions, loop free connectivity is thus provided between all the bridges and LANs in the network.

Figure 13-9 illustrates the above connectivity with the simple example of Region 1 from the example network of Figure 13-6 and Figure 13-8. Bridge 0.42 has been selected as the CIST Root and Regional Root, bridge 0.57 as the Regional Root for MSTI 1, and bridge 2.83 for MSTI 2 by management of the per MSTI Bridge Identifier priority component. The potential loop through the three bridges in the region is blocked at different Bridge Ports for the CIST, and each MSTI, but the connectivity across the region and from each LAN and bridge in the region through the boundaries of the region is the same in all cases.

13.14 Communicating Spanning Tree information

A Spanning Tree Protocol Entity transmit and receive group addressed BPDUs (***Clause 14***, ***8.13.5***) through each of its Bridge Ports to communicate with the Spanning Tree Protocol Entities of the other bridges attached to the same LAN. The group address used is one of a small number of addresses that identify frames that are not directly forwarded by bridges (8.6.3), but the information contained in the BPDU can be used by a bridge in calculating its own BPDUs to transmit and can stimulate that transmission.

BPDUs are used to convey the following types of messages:

a) Configuration Messages
b) Topology Change Notification (TCN) Messages
c) MST Configuration Identifiers
d) TAP Messages

Designated Ports also transmit BPDUs at intervals to guard against loss and to assist in the detection of failed components (LANs, bridges, or Bridge Ports), so all messages are designed to be idem potent.

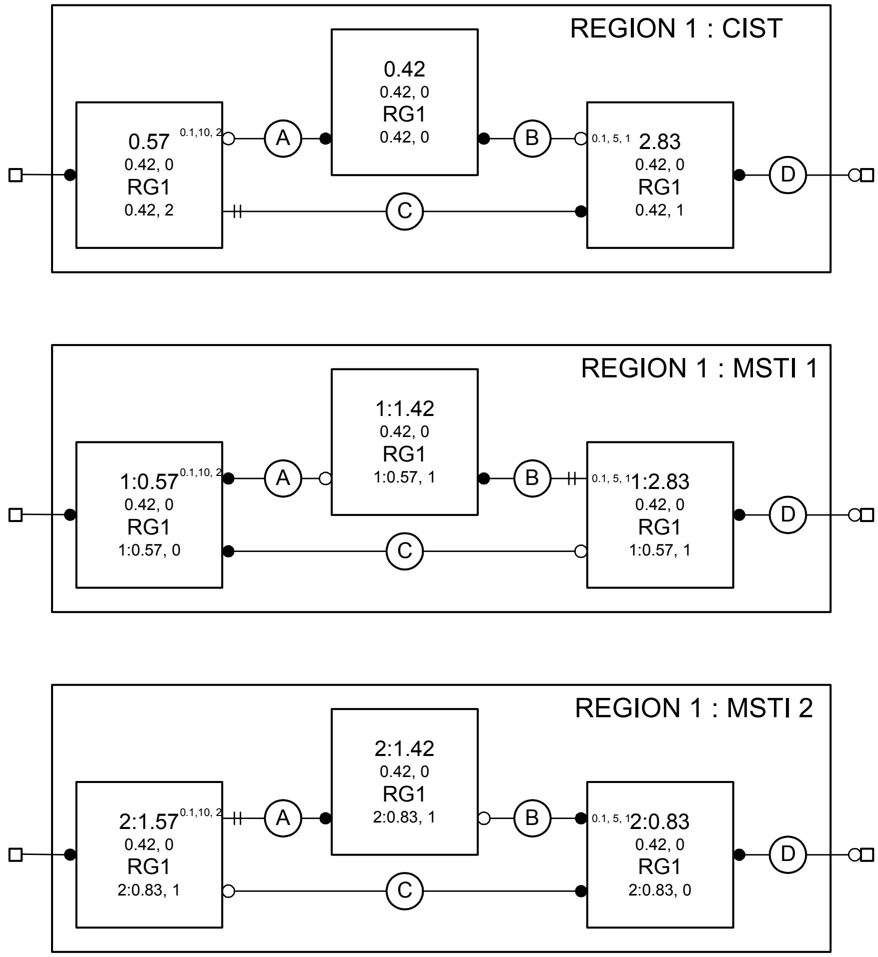


Figure 13-9—CIST and MSTI active topologies in Region 1 of the example network

A Configuration Message for the CIST can be encoded in an STP Configuration BPDU (***14.3.1***), an RST BPDU (***14.3.2***), an MST BPDU (***14.3.3***), or an SPT BPDU(<>). A TCN Message for the CIST can be encoded in an STP Topology Change Notification BPDU (***14.3.1***), or an RST, MST, or SPT BPDU with the TC flag set. Configuration and TCN Messages for the CIST and for all MSTIs in an MST Region are encoded in a single MST or SPT BPDU, as is the MST Configuration Identifier. No more than 64 MSTI Configuration Messages shall be encoded in an MST BPDU, and no more than 64 MSTIs shall be supported by an MST Bridge.

When SPB is enabled, ISIS-SPB is used to communicate CST priority vectors, IST topology information, and CST topology change information to and from the other bridges in the SPT Region and to calculate IST and SPT Port Roles and designated priority vectors. However the full CIST priority information is still conveyed in SPT BPDUs, partly to encode TAP message information for the IST, but principally to support interoperability at the boundaries of each region without having to assess whether the transmitting port is at a boundary before deciding what to encode in each BPDU. TAP messages are encoded in SPT BPDUs.

Configuration and Topology Change Notification BPDUs are distinguished from each other and from RST and MST BPDUs by their BPDU Type (***Clause 14***). RST and MST BPDUs share the same BPDU Type and are distinguished by their version identifiers.

Bridges implementing STP (Clause 8 of IEEE Std 802.1D, 1998 Edition) transmit and decode Configuration and Topology Change Notification BPDUs, and ignore RST and MST BPDUs on receipt. This ensures that connection of a Bridge Port of such a bridge to a LAN that is also attached to by a bridge implementing RSTP or MSTP is detected, as transmission of RSTP or MSTP BPDUs does not suppress regular transmissions by the STP bridge. This functionality is provided by the Port Protocol Migration state machine for RSTP (13.32). The Port Protocol Migration state machines select the BPDU types used to encode Spanning Tree messages so that all bridges attached to the same LAN participate in a spanning tree protocol, while maximizing the available functionality. If one or more attached bridges only implement STP, only Configuration and Topology Change Notification BPDUs will be used and the functionality provided by the protocol will be constrained.

13.15 Changing Spanning Tree information

Addition, removal, failure, or management of the parameters of bridges and LAN connectivity can change spanning tree information and require Port Role changes in all or part of the network (for the CIST) or all or part of an MST or SPT Region (for an MSTI or SPT Set). A CIST or MSTI configuration message received in a BPDU is considered superior to, and will replace, that recorded in the receiving port’s port priority vector if its message priority vector is better, or if it was transmitted by the same Designated Bridge and Designated Port and the message priority vector, timer, or hop count information differ from those recorded.

RSTP and MSTP propagate new information rapidly from bridge to bridge, superseding prior information and stimulating further transmissions until it reaches either Designated Ports that have already received the new information through redundant paths in the network or the leaves of the Spanning Tree, as defined by the new configuration. Configuration Message transmissions will then once more occur at regular intervals from ports selected as Designated Ports.

To ensure that old information does not endlessly circulate through redundant paths in the network, preventing the effective propagation of the new information, MSTP associates a hop count with the information for each spanning tree. The hop count is assigned by the CIST Regional Root or the MSTI Regional Root and decremented by each receiving port. Received information is discarded and the port made a Designated Port if the hop count reaches zero.

RSTP and MSTP’s CST processing, does not use an explicit hop count (for reasons of STP compatibility) but detect circulating aged information by treating the BPDU Message Age parameter as an incrementing hop count with Max Age as its maximum value. MSTP increments Message Age for information received at the boundary of an MST Region, discarding the information if necessary.

If a Bridge Port’s MAC_Operational parameter becomes FALSE, the port becomes a Disabled Port and received information is discarded. Spanning tree information for the tree can be recomputed, the bridge’s Port Roles changed, and new spanning tree information transmitted if necessary. Not all component failure conditions can be detected in this way, so each Designated Port transmits BPDUs at regular intervals and a receiving port will discard information and become a Designated Port if two transmissions are missed.

NOTE—Use of a separate hop count and message loss detection timer provides superior reconfiguration performance compared with the original use of Message Age and Max Age by STP. Connectivity loss detection is not compromised by the need to allow for the overall diameter of the network, nor does the time allowed extend the number of hops permitted to aged recirculating information. Management calculation of the necessary parameters for custom topologies is also facilitated, as no allowance needs to be made for relative timer jitter and accuracy in different bridges.

ISIS-SPB communicates CST, IST, and SPT information throughout an SPT Region using link state procedures specified in ***Clause 29***. In addition to the normal hop-by-hop distribution of this information, which is essential to guarantee its dissemination, each link state PDU is also distributed on the SPT rooted at the originating bridge, so new information can reach bridges in the region with the same delay as data.

13.16 Changing Port States with RSTP or MSTP

The Port State for the CIST and each MSTI for each Bridge Port is controlled by state machines whose goal is to maximize connectivity without introducing temporary loops in each of these active topologies. Root Ports, Master Ports, and Designated Ports are transitioned to the Forwarding Port State, and Alternate Ports and Backup Ports to the Discarding Port State, as rapidly as possible. Transitions to the Discarding Port State can be simply effected without the risk of data loops. This clause (13.16) describes the conditions for RSTP or MSTP to transition the Port State for a given spanning tree to Forwarding.

Starting with the assumption that any connected fragment of a network is composed of bridges, Bridge Ports, and connected LANs that form a subtree of a spanning tree, conditions are derived for transitioning ports with Root Port, Master Port, or Designated Port roles, such that the newly enlarged fragment continues to form either a subtree or the whole of the spanning tree. Since these conditions are applied every time a fragment is enlarged, it is possible to trace the growth of a fragment from a single bridge, which is clearly a consistent, if small, subtree of a spanning tree, to any sized fragment—thus justifying the initial assumption.

Port States in two subtrees, each bounded by ports that are not forwarding or are attached to LANs not attached to any other bridge, can be made consistent by waiting for any changes in the priority vector information used to assign Port Roles to reach all bridges in the network, thus ensuring that the subtrees are not, and are not about to be, joined by other Forwarding Ports. However, it can be shown that a newly selected Root Port can forward frames as soon as prior recent root ports on the same bridge cease to do so, without further communication from other bridges. Rapid transitions of Designated Ports and Master Ports do require an explicit Agreement from the bridges in the subtrees to be connected. The Agreement mechanism is described, together with a Proposal mechanism that forces satisfaction of the conditions if they have not already been met by blocking Designated Ports connecting lower subtrees that are not yet in agreement. The same Agreement mechanism is then used to transition the newly blocked ports back to forwarding, advancing any temporary cut in the active topology toward the edge of the network.

13.16.1 Subtree connectivity and priority vectors

Any given bridge B, the LANs connected through its Forwarding Designated Ports, the further bridges connected to those LANs through their Root Ports, the LANs connected to their Forwarding Designated Ports, and so on, recursively, comprise a subtree S_B. Any LAN L that is part of S_B will be connected to B through a Forwarding Designated Port P_{CL} on a bridge C also in S_B. L cannot be directly connected to any port P_B on bridge B unless B and C are one and the same, since the message priority vector for P_B is better than that of any port of any other bridge in S_B, and prior to Forwarding P_{CL} will have advertised its spanning port priority vector for long enough for it to receive any better message priority vector (within the design probabilities of protocol failure due to repeated BPDU loss) or will have engaged in an explicit confirmed exchange (see below) with all other Bridge Ports attached to that LAN.

NOTE—The analysis for the distance vector based RSTP and MSTP differs from that for the link state based ISIS-SPB (see 13.17). In the latter C's priority vector can become better than B's while C remains in S_B, without B being aware of the improvement first.

13.16.2 Root Port transition to Forwarding

It follows from the above that B's Root Port can be transitioned to Forwarding immediately whether it is attached to a LAN in S_B or in the rest of the network, provided that all prior recent Root Ports on B (that might be similarly arbitrarily attached) have been transitioned to Discarding and the Root Port was not a Backup Port recently (B and C the same as above).

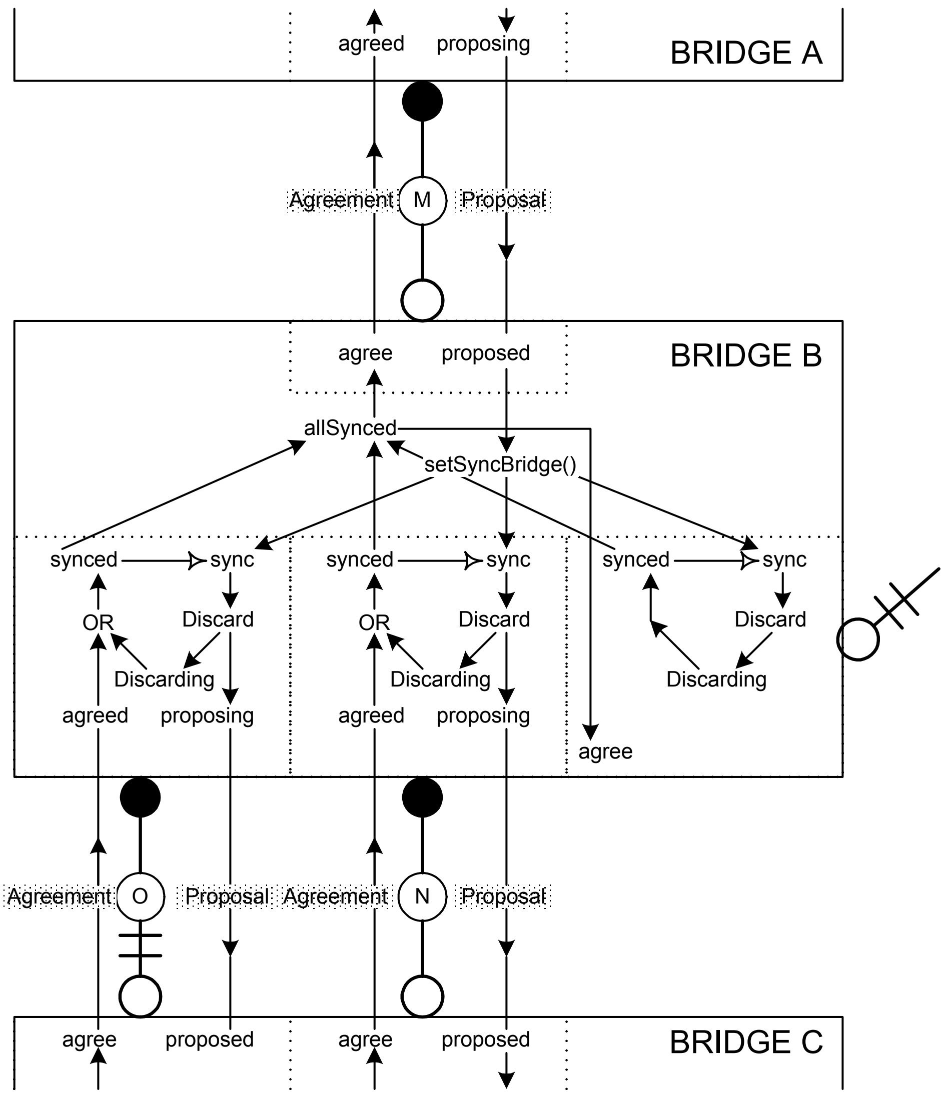


Figure 13-10—Agreements and Proposals

13.16.3 Designated Port transition to Forwarding

On any given bridge *A*, the Designated Port P_{AM} connected to a LAN *M* can be transitioned to Forwarding provided that the message priority advertised by the Designated Port P_{CL} on any LAN *L* in any subtree S_{M1}, S_{M2}, etc. connected to *M* is worse than that advertised by P_{AM}; that any bridge *D* attached to *L* has agreed that P_{CL} is the Designated Port; and that only the Root Port and Designated Ports on *D* are Forwarding. A sufficient condition for P_{AM} to transition to Forwarding is that *M* is a point-to-point link attached to the Root Port P_{BM} of a bridge *B*, that the port priority of P_{BM} is same as or worse than that of P_{AM}, and any port P_{BN} on B is Discarding or similarly attached to a bridge *C*. P_{BM} signals this condition to P_{AM} by setting the Agreement flag in a Configuration Message carrying P_{BM}'s designated priority and Port Role.

NOTE 1—RSTP and MSTP use adminPointToPointMAC and operPointToPointMAC (***6.6.3***) to allow the point-to-point status of LANs to be managed and used by the Port Role Transition state machines for Designated Ports. A newly selected Root Port can be transitioned to Forwarding rapidly, even if attached to a shared media LAN.

Figure 13-10 illustrates the generation of an Agreement at a bridge's Root Port from an Agreement received or a Port State of Discarding at each of its Designated Ports, and a Port State of Discarding at each of its

Alternate and Backup Ports. A bridge receiving a Proposal transitions any Designated Port not already synchronized to Discarding so it can send the Agreement, and that port solicits an Agreement by sending a Proposal in its turn.

NOTE 2—Agreements can be generated without prior receipt of a Proposal as soon as the necessary conditions are met. Subsequent receipt of a Proposal serves to elicit a further Agreement. If all other ports have already been synchronized (allSynced in Figure 13-10) and the Proposal's priority vector does not convey worse information, synchronization is maintained and there is no need to transition Designated Ports to Discarding once more, or to transmit further Proposals.

13.16.4 Master Port transition to Forwarding

While the connectivity of the CIST from the CIST Regional Root through the MST Region to the rest of the CIST comprises a subtree rooted in the CIST Regional Root, the connectivity of the MSTI from the Master Port includes both a subtree below the CIST Regional Root and a subtree rooted in the MSTI Regional Root and is connected to the CIST Regional Root by an MSTI Root Port. Figure 13-11 illustrates this connectivity for both part of the CIST and an MSTI through a region in the example network of Figure 13-6. (In the example, this latter subtree provides connectivity from the Master Port through LAN N to the subtree of the CIST outside the region). Prior to the Master Port's transition to Forwarding, it is possible that either MSTI subtree is providing connectivity to a prior Master Port. Before the Master Port can transition, the connectivity of both subtrees has to agree with the new CIST Regional Root.

NOTE 1—The physical layout shown in the two halves of Figure 13-11 differs in order to reflect the different priorities and logical topologies for the two spanning tree instances. The layout convention is that Designated Ports are shown as horizontal lines, Root Ports as vertical lines, and Alternate Ports as diagonal lines.

Figure 13-12 illustrates the extension of the Agreement mechanism to signal from Designated Ports to Root Ports as well as vice versa. To ensure that an MSTI does not connect alternate Master Ports, an Agreement is only recognized at an MSTI Port when the associated CIST Regional Root information matches that selected by the receiving port. Proposals, eliciting Agreements, necessarily flow from Designated Ports to Root Ports with the propagation of spanning tree information, so a new CIST Regional Root cannot transmit a Proposal directly on its MSTI Root Ports. However, updating a CIST Designated Port's port priority vector with a new Regional Root Identifier forces the port to discard frames for all MSTIs, thus initiating the Proposal from the first bridge nearer the MSTI Regional Root that learns of the new Regional Root.

When an Agreement A_{MR} is sent by a Root Port P_{MR} on a Regional Root *M,* it attests that the CIST Root Identifier and External Root Path Cost components of the message priority advertised on all LANs connected to the CIST by P_{MR} through M are the same as or worse than those accompanying A_{MR}. The connectivity provided by each MSTI can be independent of that provided by the CIST within the MST Region and can therefore connect P_{MR} and one or more CIST Root Ports external to but attached at the boundary of the region even as CIST connectivity within the region is interrupted in order to satisfy the conditions for generating A_{MR}. The Agreement cannot therefore be generated unless all MSTI subtrees as well as the CIST subtree internal to the region are in Agreement. To ensure that an MSTI does not connect to a CIST subtree external to the region that does not meet the constraints on the CST priority vector components, an Agreement received at an MSTI Designated Port from a Bridge Port not internal to the region is only recognized if the CIST Root Identifier and External Root Path Cost of the CIST root priority vector selected by the transmitting Bridge Port are equal to or worse than those selected by the receiver. Updating of a CIST Designated Port's port priority vector with a worse CIST Root Identifier and External Root Path Cost forces the port to discard frames for all MSTIs, thus initiating a Proposal that will elicit agreement.

NOTE 2—MSTI Designated Ports are prompted to discard frames, as required above, as follows. The CIST Port Information state machine sets sync for all MSTIs on a transition into the UPDATE state if updating the port priority with the designated priority changes the Regional Root Identifier or replaces the CIST Root Identifier or External Path Cost with a worse tuple. The MSTI's Port Role Transition machine acts on the sync, instructing the port to discard frames, and setting synced and cancelling sync when the port is discarding or an agreement is received.

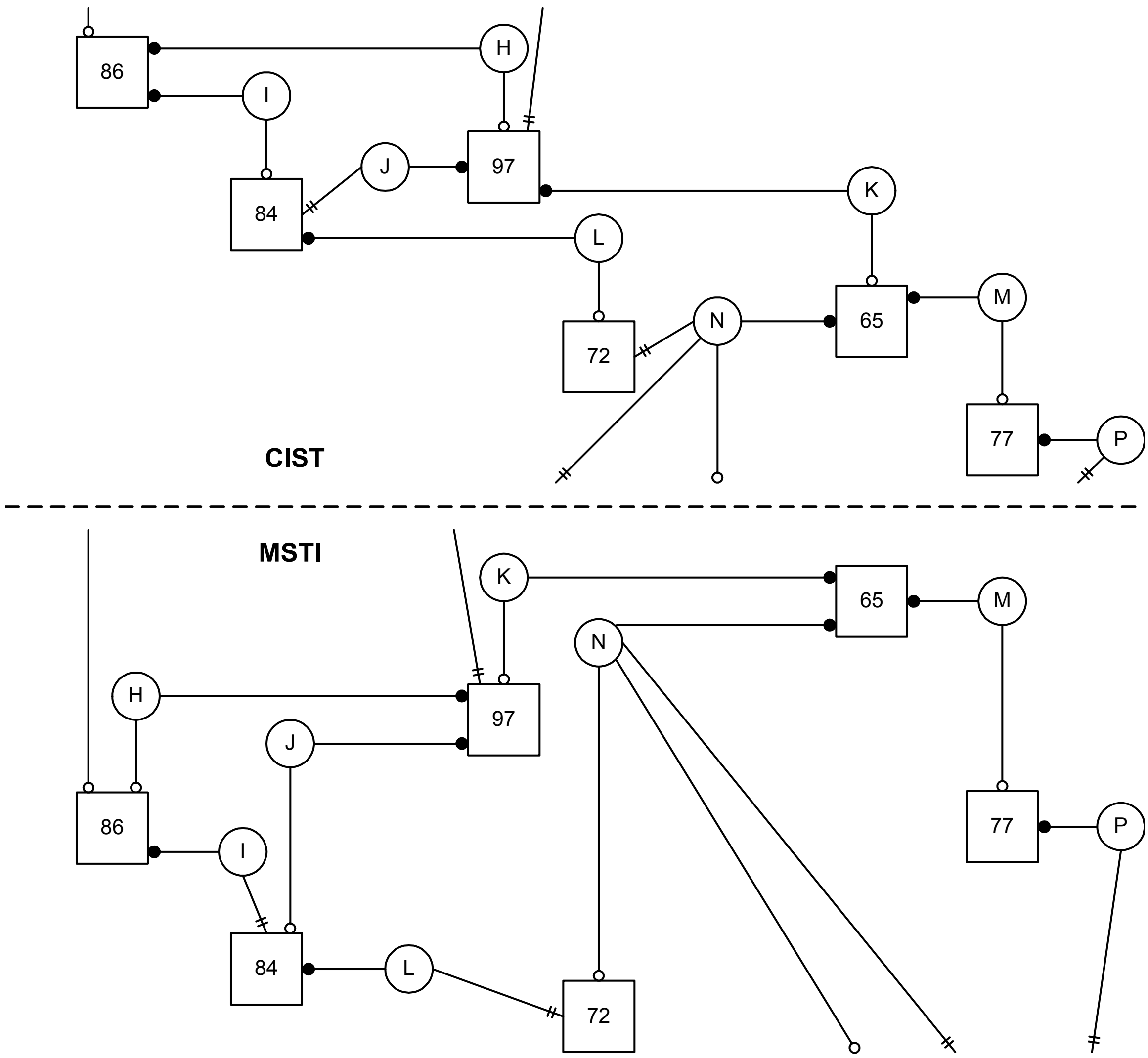


Figure 13-11—CIST and MSTI Active Topologies in Region 2 of Figure 13-6—

NOTE 3—A “cut” in an MSTI can be transferred to the CST, either at a Designated Port attached to the same LAN as an STP bridge or at the Root Port of a bridge in an adjacent region. If the CST priority components have already been synced, as is likely if the original cut was caused by changes in physical topology within the region, the cut will terminate there. Otherwise the transferred cut precedes a cut in the CIST, and the synced port may terminate the latter. In that way, cuts in the CST will proceed through an MST Region by the quickest tree that will carry them.

NOTE 4—In the important topology where the CIST Root Bridge is within an MST Region, cuts are not transferred from the region’s IST to any MSTI. CIST cuts propagating in the region will not disrupt MSTI connectivity.

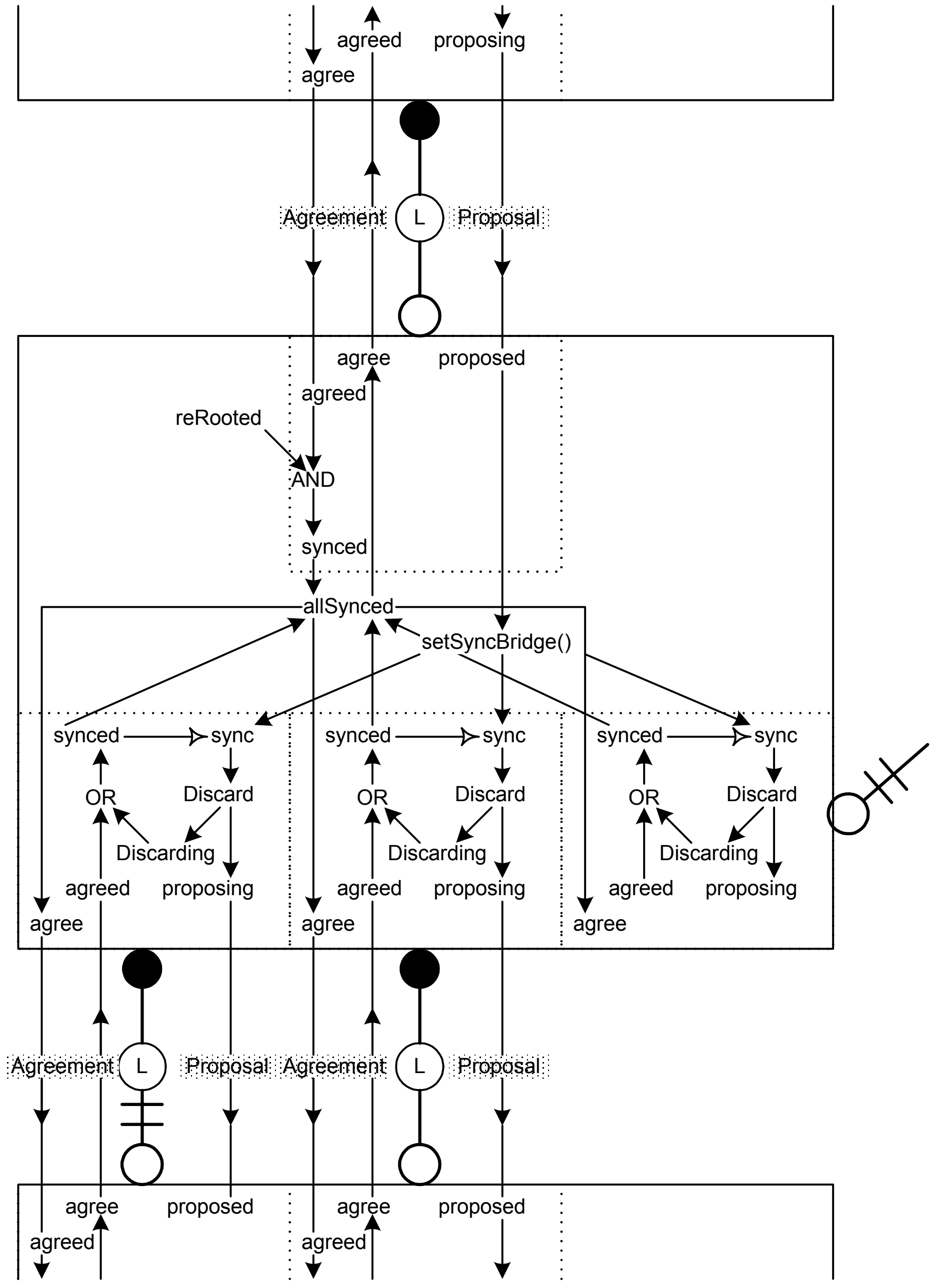


Figure 13-12—Enhanced Agreements

13.17 Changing Port States with SPB

While the link state routing protocols are less likely to create temporary loops than distance vector based protocols, loops are still possible if, for example, two links fail about the same time. Any potential loop is a serious problem for bridged networks, as a frame that is multicast or whose destination has not been learned and is travelling around a loop is copied to each of the connected subtrees on every circuit, consuming bandwidth throughout the network. Each SPT Bridge Port participates in Tree Agreement Protocol (TAP) that ensures that additions to each active topology do not occur unless there is sufficient agreement between neighbouring bridges (with some ports temporarily transitioned to Discarding if necessary) to ensure that a temporary loop will not be created.

Like RSTP and MSTP, TAP uses Agreements (see 13.16, Figure 13-12) to ensure that ports that transition to forwarding do not create loops. However, ISIS distributes information in parallel with computation, rather than computing a new active topology hop-by-hop as priority vector information is passed from the Root Bridge to members of a potential subtree. Moreover the distribution path is not restricted to the potential subtree. Arbitrary distribution and parallel computation means that different bridges can complete topology calculations at quite different times, with two major consequences as follows.

First, each bridge is not necessarily aware of priority vector improvements passed to any subtree connected through one of its Designated Ports before any other bridge in that subtree. TAP cannot rely (like RSTP) only on Agreements that propagate towards the Root of a tree, but needs (like MSTP) to use Agreements that propagate both up and down the tree. TAP ensures that new connectivity only occurs between a Designated Port (P_A, say) and a Root Port (P_B, say), when all Root Ports that are forwarding parents of P_A, i.e. connected through bridges from forwarding Designated Port to forwarding Root Port and through LANs from Root Port to Designated Port, are each associated with a better priority vector than any of the Root Ports that are forwarding children of P_B, i.e. that are connected through bridges from forwarding Root Port to forwarding Designated Port and through LANs from Designated Port to Root Port. Since a bridge has at most one forwarding Root Port these constraints are sufficient to ensure loop-free connectivity.

The second major consequence of arbitrary distribution and parallel computation, is that received Agreements that can appear to contradict the results of the last local link state computation still need to be held by the receiving port in case they become applicable after a future computation completes. This in turn means that a port P_A needs to know that any outdated Agreement sent to a neighbor P_B has been discarded by P_B, before P_A can take any action on an Agreement from P_B that would be inconsistent with the outdated Agreement. The explicit discard prevents two Designated Ports attached to the same LAN (for example) from both transitioning to forwarding, each using an old Agreement with a Root Port role from the other. To minimize the number of message exchanges necessary to create new connectivity, unsatisfactory received Agreements are voluntarily discarded when a link state topology calculation completes. TAP supports the necessary communication, allowing for the possibility of TAP message loss and buffering prior to transmission and before processing on receipt.

TAP supports the IST using CIST information present in both MST and SPT BPDUs, minimizing active topology disruption while allowing parallel adoption of a new topology by allowing neighbouring bridges to communicate IST priority vectors, propagate agreements, and signal the need for temporary cuts to speed transitions to forwarding, even while those neighbors' views of the entire active topology differ with differences in their knowledge of the latest physical topology changes. A single BPDU cannot convey the same per tree state for all possible SPTs, so the input to their link state calculation is summarized in a TAP Digest (13.17.1). If a received digest matches the bridge's own digest, the bridge can compute the implied received Agreements. Using the TAP Digest means neighbouring bridges cannot synchronize their local views of any SPT's active topology until they have synchronized their views of the physical topology of the entire SPT Region, and increases per tree computation, but reducing the number of messages is worthwhile.

NOTE—Management VLANs with modest bandwidth needs should be allocated to the IST in SPT Regions.

TAP messages contain a two-bit Agreement Number and a Discarded Agreement Number, that apply to IST Agreements (indicated by the CIST Agreement flag). Once an Agreement has been sent it is considered *outstanding* until a matching or more recent Discarded Agreement Number is received. SPT Agreements, implied by the transmission of a TAP Digest, are outstanding until the transmitting Bridge Port has received digests, from all the other bridges attached to the LAN, that match a subsequently calculated digest.

NOTE—TAP assumes that any two digests representing the same physical topology will differ with high probability, through inclusion of ISIS sequence number is the digest calculation, if the physical topology changed in the period between their generation, unless that period significantly exceeds any communication delays due to loss and buffering of TAP messages between nearest neighbors. Since a bridge only acts on received digests that match its own latest calculated digest, all digests can be treated as ordered in time.

The per-port per-tree state machine variable agreed is recomputed whenever ISIS-SPB's link state calculation provides a new root priority vector, designated priority vectors (13.10), and Port Roles (13.12) for the relevant tree (IST or SPT). If there are no outstanding Agreements for a different role (as recorded by a chgdRole parameter), then agreed is set for each Root and Alternate Port that holds a received Agreement (from a Designated Port) that is better than the bridge's root priority vector, and set for each Designated Port that holds a received Agreement (from a Root, Alternate, or Backup Port on every other bridge attached to the LAN) that is worse than any Agreement outstanding for that port and the root priority vector. Otherwise agreed is cleared. Received IST Agreements that would not result in setting agreed (if the receiving port had no outstanding agreements) are discarded, and a TAP message sent with the Discarded Agreement Number.

The agreed variables for each port are used, in conjunction with Discarding Port States and reRooted, to set synced for each port and hence to set allSynced, allowing the conditions for Port Role and Port State transitions and the transmission of Agreements to be the same as for MSTP (see Figure 13-12, 13-23, 13-24, and 13-24) with the additional requirement that allSynced be set for any port to transition to Learning or Forwarding, just as required by MSTP for Master Port state transitions. If agreed is set for an IST Root Port, but allSynced is not, and the Agreement for that Root Port was accompanied by a Proposal, then sync is set for the Designated Ports to force those without applicable Agreements to Discard, so that the Root Port can be allSynced and send an Agreement up the tree. The TAP Digest cannot signal per-SPT Agreements, so the Designated Ports that are not agreed are always made Discarding so that the digest (with its implied Agreements) can be sent.

13.17.1 TAP Digest

<<t.b.s. Should have format selector, like MST Configuration Digest.>>

13.18 Managing spanning tree topologies

The active topology of the CIST, and the topologies that can result after the failure or addition of network components, may be managed by assigning values to some or all of the following:

— The Bridge Priority component of the CIST Bridge Identifier for one or more bridges.
— The External Port Path Cost (also referred to as the Port Path Cost for RSTP) for some Bridge Ports.
— Components of the MST Configuration Identifier for bridges with the same Configuration Digest.
— The CIST Internal Port Path Cost Port (for MSTP and SPB protocols) for some Bridge Ports.
— The Port Priority component of the Port Identifier for some Bridge Ports.

Within an MST Region, the active topology of each MSTI may be managed by assigning values to some or all of the following:

— The Bridge Priority component of the MSTI Regional Root Identifier.
— The Internal Port Path Cost for the MSTI for some Bridge Ports,
— The Port Priority component of the MSTI's Port Identifier for some Bridge Ports.

In general topology management objectives can be met by modifying only a few parameter values in a few bridges in the network. Table 13-3 specifies default values and ranges for Bridge Priorities and Port Priorities. If these parameters can be updated by management, the bridge shall have the capability to use the full range of values with the granularity specified.

Table 13-3—Bridge and Port Priority values

Parameter	Recommended or default value	Range
Bridge Priority	32 768	0–61 440 in steps of 4096
Port Priority	128	0–240 in steps of 16

NOTE 1—The stated ranges and granularities for Bridge Priority and Port Priority differ from those in IEEE Std 802.1D, 1998 Edition and earlier revisions of that standard. Expressing these values in steps of 4096 and 16 allows consistent management of old and new implementations of this standard; the steps chosen ensure that bits that have been re-assigned are not modified, but priority values can be directly compared.

Table 13-4 recommends defaults and ranges for Port Path Cost and Internal Port Path Cost values, chosen according to the speed of the attached LAN, to minimize the administrative effort required to provide reasonable active topologies. If these values can be set by management, the bridge shall be able to use the full range of values in the parameter ranges specified, with a granularity of 1.

Table 13-4—Port Path Cost values

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

When two or more links are aggregated (see IEEE 802.1AX), Port Path Cost and Internal Port Path Cost values can be modified to reflect the actual throughput. However, as the primary purpose of Path Cost is to select active topologies, it can be inappropriate to track throughput too closely, as the resultant active topology could fluctuate or differ from that intended by the network administrator. For example, if the network administrator had chosen aggregated links for resilience (rather than for increased data rate), it would be inappropriate to change topology as a result of one of the links in an aggregation failing. Similarly, with links that can autonegotiate their data rate, reflecting such changes of data rate in changes to Path Cost is not necessarily appropriate. As a default behavior, dynamic changes of data rate should not automatically cause changes in Port Path Cost.

NOTE 2—BPDUs are capable of carrying 32 bits of Root Path Cost information, though IEEE Std 802.1D-1998 and its earlier revisions limited the range of the Port Path Cost parameter to a 16-bit unsigned integer value. Table 13-4 uses the full 32-bit range to extend the range of supported link speeds. Additional recommended values can be calculated as 20

000 000 000/(Link Speed in Kb/s). Limiting the range of the Path Cost parameter to 1–200 000 000 ensures that the accumulated Path Cost cannot exceed 32 bits over a concatenation of 20 hops. Where bridges using the IEEE Std 802.1D-1998 recommendations and others using Table 13-4 are mixed in the same Bridged Local Area Network, explicit configuration is likely to be necessary to obtain reasonable CST topologies.

13.19 Updating learned station location information

In normal stable operation, learned station location information held in the Filtering Database need only change as a consequence of the physical relocation of stations. It is therefore desirable to employ a long aging time for Dynamic Filtering Entries (***8.8.3***), especially as many end stations transmit frames following power-up causing the information to be relearned.

However, when the active topology reconfigures, stations can appear to move from the point of view of any given bridge even if that bridge's Port States have not changed. If a Bridge Port is no longer part of an active topology, stations are no longer reachable through that port, and its Dynamic Filtering Entries are removed from that bridge's Filtering Database. Conversely, stations formerly reachable through other ports might be reachable through a newly active port. Dynamic Filtering Entries for the other ports are removed, and RSTP and MSTP transmit Topology Change Notification Messages both through the newly active Port and through the other active Ports on that Bridge. TCNs signal additional connectivity, not just changes in connectivity as relearning a station's location is only possible if it can be reached, and if that is possible when a port is removed from the active topology another port will be added. A TCN is sent when a Bridge Port joins the active topology, and not before, so that bridges that can relearn removed station location information and minimize unnecessary flooding of frames. A bridge that receives a TCN on an active port removes Dynamic Filtering Entries for their other active Ports and propagates the TCN through those ports.

NOTE 1—STP allowed for the presence of LAN repeaters that could partition a shared media LAN, thus causing stations to appear to move when the partition was repaired later—with the only Bridge Port changing Port Role or Port State transitioning to Discarding at that time. This scenario does not occur with current technology, and its future likelihood does not justify the use of TCNs to signal connectivity loss. Bridge Ports that participate in the MAC status propagation protocol should be capable of originating TCNs when that protocol signals additional connectivity.

The Topology Change state machine (13.39) avoids removing learned information when ports temporarily revert to Discarding to suppress loops. It treats a port as joining the active topology when it becomes forwarding, and no longer active when it becomes an Alternate, Backup, or Disabled Port and stops forwarding and learning. The Topology Change state machine does not generate TCNs following Edge Port (operEdge,<>) Port State changes, as these do not affect connectivity or station location information in the rest of the network, nor does it remove Dynamic Filtering Entries for Edge Ports when TCNs are received.

Dynamic Filtering Entries for MAC addresses previously learned on a Root Port may be modified to move those addresses to an Alternate Port that becomes the new Root Port and a TCN sent only through the new Root Port (and not through other active ports), reducing the need to flood frames. This optimization is possible because a retiring Root Port that becomes Discarding temporarily partitions the active topology into two subtrees, one including all bridges and LANs hitherto reachable through the retiring Root Port, and the other including all the others. If the new Root Port simply provides a new path to the first of these subtrees, its stations will not appear to move from the point of view of bridges in the other subtree. Alternatively if a the tree reconfiguration is more complex one or more newly Designated Port will become active and will transmit the necessary TCNs.

NOTE 2—The rules described require removal of potentially invalid learned information for a minimum set of ports on each bridge. A bridge implementation can flush information from more ports than strictly necessary, removing (for example) all Dynamic Filtering Entries rather than just those for the specified ports. This does not result in incorrect operation, but will result in more flooding of frames with unknown destination addresses.

Changes in the active topology of any given MSTI or SPT do not change Dynamic Filtering Entries for the CIST or any other MSTI or SPT, unless the underlying changes in the physical topology that gave rise to the reconfiguration also cause those trees to reconfigure. Changes to the CST, i.e., the connectivity provided between regions, can cause end station location changes for all trees. Changes to an IST can cause CST end station location changes but do not affect MSTIs in that region unless those trees also reconfigure.

On receipt of a CIST TCN Message from a Bridge Port not internal to the region, or on a change in Port Role for a Bridge Port at the region boundary:

— TCN Messages are transmitted through each of the other ports of the receiving bridge for each MSTI and the Dynamic Filtering Entries for those ports are removed.
— ISIS-SPB communicates the change to each bridge within the SPT Region, and Dynamic Filtering Entries learned for all SPTs for each port that is a Designated Port for the SPT rooted at the bridge receiving the CIST TCN are removed.

NOTE 3—TCN Messages for the CIST are always encoded in the same way, irrespective of whether they are perceived to have originated from topology changes internal to the region or outside it. This allows RSTP Bridges whose Root Ports attach to a LAN within an MST Region to receive these TCN Messages correctly.

NOTE 4—The port receiving a CIST TCN Message from another Bridge Port external to the region can be a Master Port, a Designated Port attached to the same LAN as an STP bridge, or a Designated Port attached to a LAN that is within the region but is attached to by the Root Ports of bridges in other regions.

NOTE 5—Dynamic Filtering Entries created by ISIS-SPB as a result of that protocols direct knowledge of the location of some stations are not removed when a TCN is received, but can be changed by ISIS-SPB as result of its calculations.

NOTE 6—Topology changes for the IST are always propagated by TCNs received and transmitted in BPDUs, and are not injected as a result of ISIS-SPB calculations as the latter complete prior to new connectivity being established.

13.20 Managing reconfiguration

13.21 One-way connectivity

13.22 Fragile bridges

13.23 Hot upgrades

13.24 Spanning tree protocol state machines

Each Spanning Tree Protocol Entity’s operation of the protocols specified in this clause (Clause 13) is specified by the following state machines:

a) Port Role Selection (PRS, 13.36)

with the following state machines for each Bridge Port:

b) Port Timers (PTI, 13.30)

c) Port Protocol Migration (PPM, 13.32)
d) Port Receive (PRX, 13.31)
e) Port Transmit (PRT, 13.34)
f) Bridge Detection (BDM, 13.33)

and the following optional state machine for each Bridge Port:

g) Layer 2 Gateway Port Receive (L2GPRX, 13.40)

and the following state machines for each Bridge Port for the CIST and for each MSTI:

h) Port Information (PIM, 13.35)
i) A Topology Change (TCM, 13.39)

and the following state machines for each Bridge Port for the CIST, for each MSTI, and for each SPT:

j) Port Role Transitions (PRT, 13.37)
k) Port State Transition (PST, 13.38)

Each state machine and its associated variable and procedural definitions is specified in detail in 13.25 through 13.40. The state machine notation is specified in ***Annex P***. Figure 13-13 is not itself a state machine but provides an overview of the state machines, their state variables, and communication between machines. Figure 13-14 describes its notation.

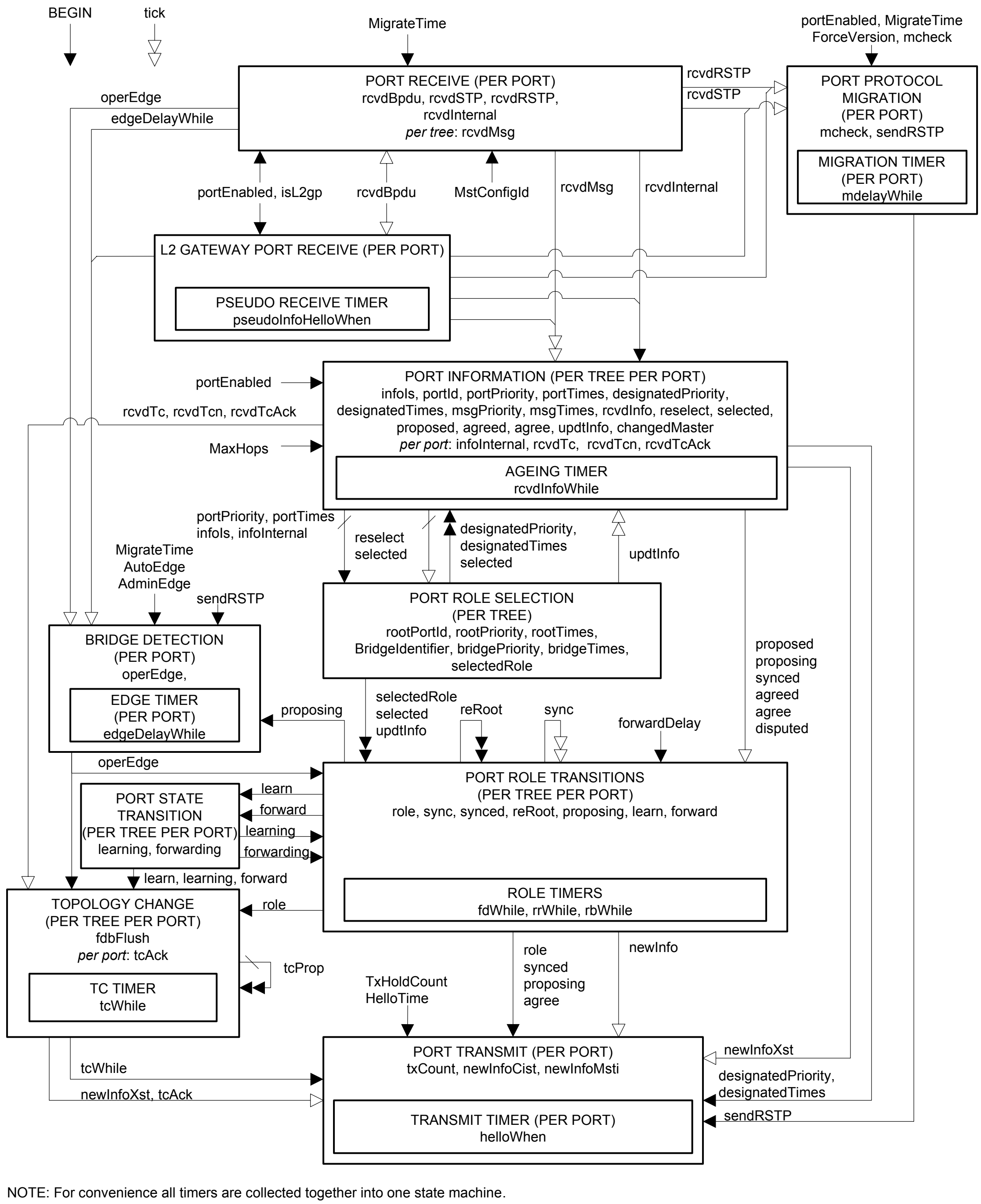


Figure 13-13—Spanning tree protocol state machines—overview and relationships

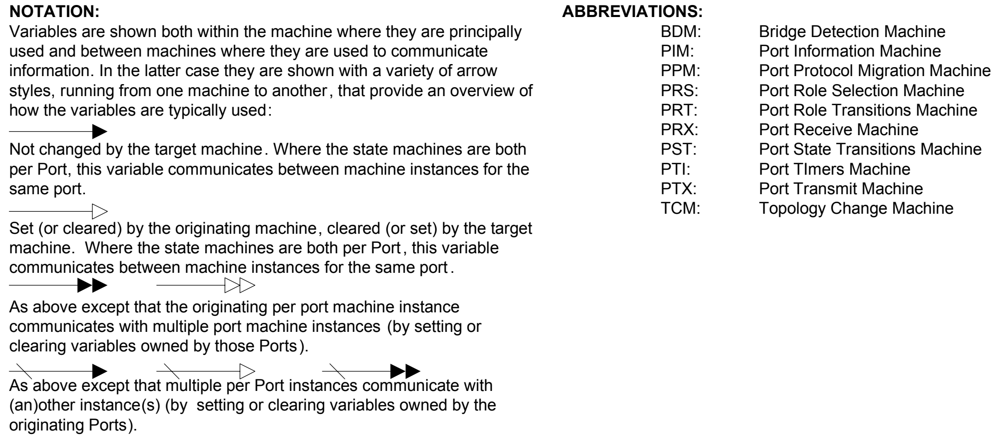


Figure 13-14—MSTP overview notation

13.25 State machine timers

The timer variables declared in this subclause are part of the specification. The accompanying descriptions are provided to aid in the comprehension of the protocol only, and are not part of the specification. Each timer variable represents an integral number of seconds before timer expiry.

One instance of the following shall be implemented per-port:

a) edgeDelayWhile (13.25.1)
b) helloWhen (13.25.3)
c) mdelayWhile (13.25.4)

One instance of the following shall be implemented per-port when L2GP functionality is provided:

d) pseudoInfoHelloWhen (13.25.9)

One instance per-port of the following shall be implemented for the CIST and one per-port for each MSTI:

e) fdWhile (13.25.1)
f) rrWhile (13.25.1)
g) rbWhile (13.25.1)
h) tcWhile (13.25.1)
i) rcvdInfoWhile (13.25.1)
j) tcDetected. The Topology Change timer for MRP application usage. New messages are sent while this timer is running (see 10.2).

Table 13-5 specifies values and ranges for the initial values of timers and for transmission rate limiting performance parameters. Default values are specified to avoid the need to set values prior to operation in most cases, and are widely applicable to networks using the spanning tree protocols specified in this standard. The table recommends Bridge Hello Time, Bridge Max Age, and Bridge Forward Delay values that maximise interoperability in networks that include bridges using STP as specified in IEEE Std 802.1D-1998. Ranges are specified to ensure that the protocols operate correctly.

Table 13-5—Timer and related parameter values

Parameter	Default	Permitted Range	Interoperability recommendations
Migrate Time	3.0	—[a]	—[a]
Hello Time	2.0	—[a]	—[a]
Bridge Max Age	20.0	6.0–40.0	20.0
Bridge Forward Delay	15.0	4.0–30.0	15.0
Transmit Hold Count	6	1–10	6
Max Hops	20	6–40	—

All times are in seconds. —[a] Not applicable, value is fixed.

NOTE—Changes to Bridge Forward Delay do not affect reconfiguration times, unless the network includes bridges that do not conform to this revision of this standard. Changes to Bridge Max Age can have an effect, as it is possible for old information to persist in loops in the physical topology for a number of “hops” equal to the value of Max Age in seconds, and thus exhaust the Transmit Hold Count in small loops.

Bridge Max Age, Bridge Forward Delay, and Transmit Hold Count may be set by management, if this capability is provided the bridge shall have the capability to use the full range of values in the parameter ranges specified in the Permitted Range column of Table 13-5, with a timer resolution of r seconds, where $0 < r <= 1$. To support interoperability with previous revisions of this standard and IEEE Std 802.1D, a bridge shall enforce the following relationships:

$$2 \times (Bridge\_Forward\_Delay - 1.0\ seconds) >= Bridge\_Max\_Age$$

$$Bridge\_Max\_Age >= 2 \times (Bridge\_Hello\_Time + 1.0\ seconds)$$

13.25.1 edgeDelayWhile

The Edge Delay timer. The time remaining, in the absence of a received BPDU, before this port is identified as an operEdgePort.

13.25.2 fdWhile

The Forward Delay timer. Used to delay Port State transitions until other Bridges have received spanning tree information.

13.25.3 helloWhen

The Hello timer. Used to ensure that at least one BPDU is transmitted by a Designated Port in each HelloTime period.

13.25.4 mdelayWhile

The Migration Delay timer. Used by the Port Protocol Migration state machine to allow time for another RSTP Bridge on the same LAN to synchronize its migration state with this Port before the receipt of a BPDU can cause this Port to change the BPDU types it transmits. Initialized to MigrateTime (13.26.5).

13.25.5 rbWhile

The Recent Backup timer. Maintained at its initial value, twice HelloTime, while the Port is a Backup Port.

13.25.6 rcvdInfoWhile

The Received Info timer. The time remaining before information, i.e. portPriority (13.27.32) and portTimes (13.27.33), received in a Configuration Message is aged out if a further message is not received.

13.25.7 rrWhile

The Recent Root timer.

13.25.8 tcWhile

The Topology Change timer. TCN Messages are sent while this timer is running.

13.25.9 pseudoInfoHelloWhen

The Pseudo Info Hello When timer. Used to ensure that at least one Pseudo Info BPDU is presented to the spanning tree (as if it had been received on the physical port on which preparePseudoInfo() was invoked) by a Layer Two Gateway Port in a designated role in each HelloTime period.

13.26 Per-bridge variables

The variables declared in this clause (13.26) are part of the specification. The accompanying descriptions are provided to aid in the comprehension of the protocol only, and are not part of the specification.

There is one instance per-bridge component of the following variable(s):

a) ForceProtocolVersion (13.7.2)
b) Transmit Hold Count ()
c) MigrateTime (13.26.5)

One instance of the following shall be implemented per-bridge component if MSTP or SPB protocols are implemented:

d) MstConfigId (13.26.6)

The above parameters ((a) through (d)) are not modified by the operation of the spanning tree protocols, but are treated as constants by the state machines. If ForceProtocolVersion or MSTConfigId are modified by management, BEGIN shall be asserted for all state machines.

There is one instance per-bridge of each of the following for the CIST, and one for each MSTI.

e) BridgeIdentifier (13.26.1)
f) BridgePriority (13.26.2)
g) BridgeTimes (13.26.3)
h) rootPortId (13.26.7)
i) rootPriority (13.26.8)
j) rootTimes (13.26.9)

BridgeIdentifier, BridgePriority, and BridgeTimes are not modified by the operation of the spanning tree protocols but are treated as constants by the state machines. If they are modified by management, spanning tree priority vectors and Port Role assignments for shall be recomputed, as specified by the operation of the Port Role Selection state machine (13.36) by clearing selected (13.27) and setting reselect (13.27) for all Bridge Port(s) for the relevant MSTI and for all trees if the CIST parameter is changed.

13.26.1 BridgeIdentifier

The unique Bridge Identifier assigned to this bridge for this tree (CIST or MSTI).

The 12-bit system ID extension component of a Bridge Identifier (9.2.5 of IEEE Std 802.1D) shall be set to zero for the CIST, and to the value of the MSTID for an MSTI, thus allocating distinct Bridge Identifiers to the CIST and each MSTI all based on the use of a single Bridge Address component value for the MST Bridge as a whole.

NOTE—This convention is used to convey the MSTID for each MSTI Configuration Message encoded in an MST BPDU.

The four most significant bits of the Bridge Identifier (the settable Priority component) for the CIST and for each MSTI can be modified independently of the setting of those bits for all other trees, as a part of allowing full and independent configuration control to be exerted over each Spanning Tree instance.

13.26.2 BridgePriority

For the CIST, the value of the CIST bridge priority vector, as defined in 13.10. The CIST Root Identifier, CIST Regional Root Identifier, and Designated Bridge Identifier components are all equal to the value of the CIST Bridge Identifier. The remaining components (External Root Path Cost, Internal Root Path Cost, and Designated Port Identifier) are set to zero.

For a given MSTI, the value of the MSTI bridge priority vector, as defined in 13.11. The MSTI Regional Root Identifier and Designated Bridge Identifier components are equal to the value of the MSTI Bridge Identifier (13.26.1). The remaining components (MSTI Internal Root Path Cost, Designated Port Identifier) are set to zero.

BridgePriority is used by updtRolesTree() in determining the value of the rootPriority variable (see 13.26.8).

13.26.3 BridgeTimes

For the CIST, BridgeTimes comprises:

a) The current values of Bridge Forward Delay and Bridge Max Age (13.25, Table 13-5).
b) A Message Age value of zero.
c) The current value of Max Hops (13.26.3). This parameter value is determined only by management.

For a given MSTI, BridgeTimes comprises:

d) The current value of MaxHops (Max Hops in Table 13-5), the initial value of remainingHops for MSTI information generated at the boundary of an MSTI region (see 13.26.9).

BridgeTimes is used by updtRolesTree() in determining the value of the RootTimes variable (13.26.9).

13.26.4 ForceProtocolVersion

The Force Protocol Version parameter for the Bridge (13.7.2).

13.26.5 MigrateTime

The value of the Migrate Time parameter as specified in Table 13-5. This value shall not be changed.

13.26.6 MstConfigId

The current value of the bridge's MST Configuration Identifier (13.8). Changes in this parameter cause BEGIN to be asserted for the state machines for the bridge, for all trees, and for each port.

13.26.7 rootPortId

For the CIST, the Port Identifier of the Root Port, and a component of the CIST root priority vector (13.10).

For a given MSTI, the Port Identifier of the Root Port, and a component of the MSTI root priority vector (13.11).

13.26.8 rootPriority

For the CIST: the Root Identifier, External Root Path Cost, Regional Root Identifier, Internal Root Path Cost, Designated Bridge Identifier, and Designated Port Identifier components of the bridge's CIST root priority vector (13.10).

For a given MSTI: the MSTI Regional Root Identifier, Internal Root Path Cost, Designated Bridge Identifier, and Designated Port Identifier components of the bridge's MSTI root priority vector (13.11).

13.26.9 rootTimes

For the CIST, the Bridge's timer parameter values (Message Age, Max Age, Forward Delay, and remainingHops). The values of these timers are derived (see 13.29.33) from the values stored in the CIST's portTimes parameter (13.27.33) for the Root Port or from BridgeTimes (13.26.3).

For a given MSTI, the value of remainingHops derived (13.29.33) from the value stored in the MSTI's portTimes parameter (13.27.33) for the Root Port or from BridgeTimes (13.26.3).

13.26.10 TxHoldCount

The value of Transmit Hold Count (Table 13-5) for the bridge. If this is modified, the value of txCount (13.27) for all ports shall be set to zero.

13.27 Per-port variables

The variables declared in this clause (13.27) are part of the specification. The accompanying descriptions are provided to aid in the comprehension of the protocol only, and are not part of the specification.

There is one instance per port of each of the following variables:

a) AdminEdge (13.27.1)
b) ageingTime (13.27.2)
c) AutoEdge (13.27.5)
d) enableBPDUrx (13.29)
e) enableBPDUtx (13.27.10)
f) isL2gp (13.27.12)
g) mcheck (13.27.23)
h) newInfo (13.27.26)
i) operEdge (13.27.28)
j) portEnabled (13.27.29)
k) portHelloTime (13.27.30)

l) rcvdBpdu (13.27.37)
m) rcvdRSTP (13.27.41)
n) rcvdSTP (13.27.42)
o) rcvdTcAck (13.27.44)
p) rcvdTcn (13.27.45)
q) restrictedRole (13.28.19)
r) restrictedTcn (13.27.49)
s) sendRSTP (13.27.53)
t) tcAck (13.27.53)
u) tick (13.27.58)
v) txCount (13.27.59)

If MSTP or the SPB protocols are implemented there is one instance per-port, applicable to the CIST and to all MSTIs and SPTs, of the following variable(s):

w) rcvdInternal (13.27.50)

If MSTP or the SPB protocols are implemented there is one instance per-port of each of the following variables for the CIST:

x) ExternalPortPathCost ()
y) infoInternal (13.27.16)
z) master (13.27.21)
aa) mastered (13.27.22)

A single per-port instance of the following variable(s) applies to all MSTIs:

ab) newInfoMsti (13.27.27)

If the SPB protocols are implemented there is one instance per-port, of the following variable(s):

ac) chgdRole ()
ad) discardedPromiseNumber ()
ae) promiseNumber ()
af) ownTAPDigest ()
ag) rcvdTAPDigest ()

There is one instance per-port of each of the following variables for the CIST, one per-port for each MSTI, and one per-port for each SPT:

ah) agree (13.27.3)
ai) agreed (13.27.4)
aj) designatedPriority (13.27.6)
ak) designatedTimes (13.27.7)
al) disputed (13.27.8)
s) fdbFlush (13.27.13)
t) forward (13.27.14)
u) forwarding (13.27.15)
v) infoIs (13.27.17)
w) InternalPortPathCost ()
x) learn (13.27.19)
y) learning (13.27.13)
z) msgPriority (13.27.24)
aa) msgTimes (13.27.25)

ab) portId (13.27.31)
ac) portPriority (13.27.32)
ad) portTimes (13.27.33)
ae) proposed (13.27.34)
af) proposing (13.27.35)
ag) pseudoRootId (13.29)
ah) rcvdInfo (13.27.38)
ai) rcvdMsg (13.27.40)
aj) rcvdTc (13.27.43)
ak) reRoot (13.27.46)
al) reselect (13.27.46)
am) role (13.27.50)
an) selected (13.27.51)
ao) selectedRole (13.27.52)
ap) sync (13.27.54)
aq) synced (13.27.55)
ar) tcProp (13.27.57)
as) updtInfo (13.27.60)

If the SPB protocol are implemented there is one instance per-port of the following variables for the CIST:

at) latestRole
au) latestContract

If the SPB protocol are implemented there is one instance per-port of the following variables for each SPT:

av) latestRole
aw) latestContract

13.27.1 AdminEdge

The AdminEdgePort parameter for the Port (***12.n.n***). The recommended default value is FALSE.

13.27.2 ageingTime

Filtering database entries for this Port are aged out after ageingTime has elapsed since they were first created or refreshed by the Learning Process. The value of this parameter is normally Ageing Time (***8.8.3***, ***Table 8-4***), and is changed to FwdDelay (13.28.8) for a period of FwdDelay after fdbFlush (13.27.13) is set by the topology change state machine if stpVersion (13.28.18) is TRUE.

13.27.3 agree

A Boolean. See 13.16.

13.27.4 agreed

A Boolean. Indicates that a Configuration Message has been received from another bridge attached to the same LAN indicating Agreement that all Port States for the given tree of all other bridges attached to the same LAN as this port are known to be likewise compatible with a loop free active topology determined by this bridge's priority vectors and, in the absence of further communication with this bridge, will remain compatible within the design probabilities of protocol failure due to repeated BPDU loss (13.16, 13.24).

13.27.5 AutoEdge

The AutoEdgePort parameter for the Port (***12.n.n***).

13.27.6 designatedPriority

For the CIST and a given port, the CIST Root Identifier, External Root Path Cost, Regional Root Identifier, Internal Root Path Cost, Designated Bridge Identifier, and Designated Port Identifier components of the port's CIST designated priority vector, as defined in 13.10.

For a given MSTI and port, the Regional Root Identifier, Internal Root Path Cost, Designated Bridge Identifier, and Designated Port Identifier components of the port's designated priority vector, as defined in 13.11.

13.27.7 designatedTimes

For the CIST and a given port, the set of timer parameter values (Message Age, Max Age, Forward Delay, and remainingHops) that are used to update Port Times when updtInfo is set. These timer parameter values are used in BPDUs transmitted from the port. The value of designatedTimes is copied from the CIST rootTimes Parameter (13.26.9) by the operation of the updtRolesTree() procedure.

For a given MSTI and port, the value of remainingHops used to update this MSTI's portTimes parameter when updtInfo is set. This timer parameter value is used in BPDUs transmitted from the port. The value of designatedTimes is copied from this MSTI's rootTimes parameter (13.26.9) by the operation of the updtRolesTree() procedure.

13.27.8 disputed

A Boolean. See 13.29.15.

13.27.9 enableBPDUrx

A Boolean. This per port management parameter is set by default, and should not be clear unless the port is configured as a Layer Two Gateway Port (i.e. isL2gp is set). When clear it can allow loops to be created, or can result in no connectivity. When cleared, BPDUs received on the port are discarded and not processed.

13.27.10 enableBPDUtx

A Boolean. This per port management parameter is set by default, and should not be clear unless the port is configured as a Layer Two Gateway Port (i.e. isL2gp is set). When clear it can allow loops to be created, or can result in no connectivity. When cleared, BPDUs received on the port are discarded and not processed.

13.27.11 ExternalPortPathCost

<>

13.27.12 isL2gp

A Boolean. Set by management to identify a port functioning as a Layer Two Gateway Port. This parameter is set to FALSE by default. When set, enableBPDUtx should be cleared.

13.27.13 fdbFlush

A Boolean. Set by the topology change state machine to instruct the filtering database to remove entries for this port, immediately if rstpVersion (17.20.9 of IEEE Std 802.1D) is TRUE, or by rapid ageing (17.19.1 of IEEE Std 802.1D) if stpVersion (17.20.10 of IEEE Std 802.1D) is TRUE. Reset by the filtering database once the entries are removed if rstpVersion is TRUE, and immediately if stpVersion is TRUE. Setting the fdbFlush variable does not result in removal of filtering database entries in the case that the port is an Edge Port (i.e., operEdge is TRUE). The filtering database removes entries only for those VLANs that have a fixed registration (see 10.7.2) on any port of the bridge that is not an Edge Port.

NOTE—If MVRP is in use, the topology change notification and flushing mechanisms defined in MRP (Clause 10) and MVRP (11.2.5) are responsible for filtering entries in the Filtering Database for VLANs that are dynamically registered using MVRP (i.e., for which there is no fixed registration in the bridge on non-Edge Ports).

13.27.14 forward

A Boolean. See 13.38.

13.27.15 forwarding

A Boolean. See 13.38.

13.27.16 infoInternal

If infoIs is Received, indicating that the port has received current information from the Designated Bridge for the attached LAN, infoInternal is set if that Designated Bridge is in the same MST Region as the receiving bridge and reset otherwise.

13.27.17 infoIs

A variable that takes the values Mine, Aged, Received, or Disabled, to indicate the origin/state of the Port’s Spanning Tree information (portInfo) held for the Port, as follows:

a) If infoIs is **Received**, the port has received current (not aged out) information from the Designated Bridge for the attached LAN (a point-to-point bridge link being a special case of a LAN).
b) If infoIs is **Mine**, information for the port has been derived from the Root Port for the Bridge (with the addition of root port cost information). This includes the possibility that the Root Port is “Port 0,” i.e., the bridge is the Root Bridge for the Bridged Local Area Network.
c) If infoIs is **Aged**, information from the Root Bridge has been aged out. Just as for “reselect” (see 13.27.47), the state machine does not formally allow the “Aged” state to persist. However, if there is a delay in recomputing the new root port, correct processing of a received BPDU is specified.
d) Finally if the port is disabled, infoIs is **Disabled**.

13.27.18 InternalPortPathCost

<>

13.27.19 learn

A Boolean. See 13.38.

13.27.20 learning

A Boolean. See 13.38.

13.27.21 master

A Boolean. Used to determine the value of the Master flag for this MSTI and port in transmitted MST BPDUs.

Set TRUE if the Port Role for the MSTI and Port is Root Port or Designated Port, and the bridge has selected one of its ports as the Master Port for this MSTI or the mastered flag is set for this MSTI for any other Bridge Port with a Root Port or Designated Port Role. Set FALSE otherwise.

13.27.22 mastered

A Boolean. Used to record the value of the Master flag for this MSTI and port in MST BPDUs received from the attached LAN.

NOTE—master and mastered signal the connection of the MSTI to the CST via the Master Port throughout the MSTI. These variables and their supporting procedures do not affect the connectivity provided by this standard but permit future enhancements to MSTP providing increased flexibility in the choice of Master Port without abandoning plug-and-play network migration. They are, therefore, omitted from the overviews of protocol operation, including Figure 13-13.

13.27.23 mcheck

A Boolean. May be set by management to force the Port Protocol Migration state machine to transmit RST (MST, or SPT) BPDUs for a MigrateTime (13.26.5, Table 13-5) period, to test whether all STP Bridges on the attached LAN have been removed and the Port can continue to transmit RSTP BPDUs. Setting mcheck has no effect if stpVersion (13.28.18) is TRUE.

13.27.24 msgPriority

For the CIST and a given port, the CIST Root Identifier, External Root Path Cost, Regional Root Identifier, Internal Root Path Cost, Designated Bridge Identifier, and Designated Port Identifier components of the CIST message priority vector conveyed in a received BPDU, as defined in 13.10.

For a given MSTI and port, the Regional Root Identifier, Internal Root Path Cost, Designated Bridge Identifier, and Designated Port Identifier components of the MSTI message priority vector, as defined in 13.11 and conveyed in a received BPDU for this MSTI.

13.27.25 msgTimes

For the CIST and a given port, the timer parameter values (Message Age, Max Age, Forward Delay, Hello Time, and remainingHops) conveyed in a received BPDU. If the BPDU is an STP or RST BPDU without MSTP parameters, remainingHops is set to the value of the MaxHops component of BridgeTimes (13.26.3).

For a given MSTI and port, the value of remainingHops received in the same BPDU as the message priority components of this MSTI’s msgPriority parameter.

13.27.26 newInfo

A Boolean. Set TRUE if a BPDU conveying changed CIST information is to be transmitted. It is set FALSE by the Port Transmit state machine.

13.27.27 newInfoMsti

A Boolean. Set TRUE if a BPDU conveying changed MSTI information is to be transmitted. It is set FALSE by the Port Transmit state machine.

13.27.28 operEdge

A Boolean. The value of the operEdgePort parameter, as determined by the operation of the Bridge Detection state machine (13.33).

13.27.29 portEnabled

A Boolean. Set if the Bridge's MAC Relay Entity and Spanning Tree Protocol Entity can use the MAC Service provided by the Bridge Port's MAC entity to transmit and receive frames to and from the attached LAN, i.e., portEnabled is TRUE if and only if:

a) MAC_Operational (***6.6.2***) is TRUE; and
b) Administrative Bridge Port State (***8.4***, 13.12) for the port is Enabled.

13.27.30 PortHelloTime

PortHelloTime takes the recommended default value given in Table 13-5

13.27.31 portId

The Port Identifier for this port. This variable forms a component of the port priority and designated priority vectors (13.10,13.11).

The four most significant bits of the Port Identifier (the settable Priority component) for the CIST and for each MSTI can be modified independently of the setting of those bits for all other trees, as a part of allowing full and independent configuration control to be exerted over each Spanning Tree instance.

13.27.32 portPriority

For the CIST and a given port, the CIST Root Identifier, External Root Path Cost, Regional Root Identifier, Internal Root Path Cost, Designated Bridge Identifier, and Designated Port Identifier components of the port's port priority vector (13.10).

For a given MSTI and port, the Regional Root Identifier, Internal Root Path Cost, Designated Bridge Identifier, and Designated Port Identifier components of the port's MSTI port priority vector (13.11).

13.27.33 portTimes

For the CIST and a given port, the port's timer parameter values (Message Age, Max Age, Forward Delay, Hello Time, and remainingHops). The Hello Time timer parameter value is used in transmitted BPDUs.

For a given MSTI and port, the value of remainingHops for this MSTI in transmitted BPDUs.

13.27.34 proposed

A Boolean. See 13.16.

13.27.35 proposing

A Boolean. See 13.16.

13.27.36 pseudoRootId

A Bridge Identifier configured by management on a per port and per instance basis. By default, it is set to the BridgeIdentifier (13.26.1). <<This “per instance basis” does not makes sense.>>

13.27.37 rcvdBPDU

A Boolean. Set by system dependent processes, this variable notifies the Port Receive state machine (13.31) when a valid (***Clause 14***) Configuration, TCN, RST, MST, or SPT BPDU (***14.3***) is received on the Port. Reset by the Port Receive state machine.

13.27.38 rcvdInfo

Set to the result of the rcvInfo() procedure (13.29.13).

13.27.39 rcvdInternal

A Boolean. Set TRUE by the Receive Machine if the BPDU received was transmitted by a bridge in the same MST Region as the receiving bridge.

13.27.40 rcvdMsg

A Boolean. See 13.31.

13.27.41 rcvdRSTP

A Boolean. See 13.31.

13.27.42 rcvdSTP

A Boolean. See 13.31.

13.27.43 rcvdTc

A Boolean. See 13.29.24 and 13.39.

13.27.44 rcvdTcAck

A Boolean. See 13.29.24 and 13.39.

13.27.45 rcvdTcn

A Boolean. See 13.29.24 and 13.39.

13.27.46 reRoot

A Boolean. See 13.38.

13.27.47 reselect

A Boolean. See 13.36.

13.27.48 restrictedRole

A Boolean. Set by management. If TRUE causes the port not to be selected as Root Port for the CIST or any MSTI, even it has the best spanning tree priority vector. Such a port will be selected as an Alternate Port after the Root Port has been selected. This parameter should be FALSE by default. If set, it can cause lack of spanning tree connectivity. It is set by a network administrator to prevent bridges external to a core region of the network influencing the spanning tree active topology, possibly because those bridges are not under the full control of the administrator.

13.27.49 restrictedTcn

A Boolean. Set by management. If TRUE causes the port not to propagate received topology change notifications and topology changes to other ports. This parameter should be FALSE by default. If set it can cause temporary loss of connectivity after changes in a spanning trees active topology as a result of persistent incorrectly learned station location information. It is set by a network administrator to prevent bridges external to a core region of the network, causing address flushing in that region, possibly because those bridges are not under the full control of the administrator or MAC_Operational for the attached LANs transitions frequently.

13.27.50 role

The current Port Role. DisabledPort, RootPort, DesignatedPort, AlternatePort, BackupPort, or MasterPort.

NOTE—The MasterPort role applies each MSTI when the CIST Port Role is RootPort and connects to another MST Region. An MSTI Master Port is part of the stable active topology for frames assigned to that MSTI, just as the CIST Root Port forwards frames for the IST. The Port State for each MSTI may differ as required to suppress temporary loops.

13.27.51 selected

A Boolean. See 13.36, 13.29.22.

13.27.52 selectedRole

A newly computed role for the port.

13.27.53 sendRSTP

A Boolean. See 13.32, 13.34.

13.27.54 sync

A Boolean. Set to force the Port State to be compatible with the loop free active topology determined by the priority vectors held by this bridge (13.16, 13.24) for this tree (CIST, or MSTI), transitioning the Port State to Discarding, and soliciting an Agreement if possible, if the port is not already synchronized (13.27.55).

13.27.55 synced

A Boolean. TRUE only if the Port State is compatible with the loop free active topology determined by the priority vectors held by this bridge for this tree (13.16, 13.24).

13.27.56 tcAck

A Boolean. Set to transmit a Configuration Message with a topology change acknowledge flag set.

13.27.57 tcProp

A Boolean. Set by the Topology Change state machine of any other port, to indicate that a topology change should be propagated through this port.

13.27.58 tick

A Boolean. See the Port Timers state machine (13.30).

13.27.59 txCount

A counter. Incremented by the Port Transmission (13.34) state machine on every BPDU transmission, and decremented used by the Port Timers state machine (13.30) once a second. Transmissions are delayed if txCount reaches TxHoldCount ().

13.27.60 updtInfo

A boolean. Set by the Port Role Selection state machine (13.36, 13.29.33) to tell the Port Information state machine that it should copy designatedPriority to portPriority and designatedTimes to portTimes.

13.28 State machine conditions and parameters

The following variable evaluations are defined for notational convenience in the state machines:

a) allSynced (13.28.1)
b) allTransmitReady (13.28.2)
c) cist (13.28.3)
d) cistRootPort (13.28.4)
e) cistDesignatedPort (13.28.5)
f) EdgeDelay (13.28.6)
g) forwardDelay (13.28.7)
h) FwdDelay (13.28.8)
i) HelloTime (13.28.9)
j) MaxAge (13.28.10)
k) mstiDesignatedOrTCpropagatingRootPort (13.28.11)
l) mstiMasterPort (13.28.12)
m) rcvdAnyMsg (13.28.13)
n) rcvdCistMsg (13.28.14)
o) rcvdMstiMsg (13.28.15)
p) reRooted (13.28.16)
q) rstpVersion (13.28.17)
r) stpVersion (13.28.18)
s) updtCistInfo (13.28.19)
t) updtMstiInfo (13.28.20)

13.28.1 allSynced

The condition allSynced is TRUE for a given port, for a given tree, if and only if

a) For all ports for the given tree, selected is TRUE, the port’s role is the same as its selectedRole, and updtInfo is FALSE; and
b) The role of the given port is

1) Root Port or Alternate Port and synced is TRUE for all ports for the given tree other than the Root Port; or
2) Designated Port and synced is TRUE for all ports for the given tree other than the given port; or
3) Master Port and synced is TRUE for all ports for the given tree other than the given port.

13.28.2 allTransmitReady

TRUE, if and only if, for the given port for all trees

a) selected is TRUE; and
b) updtInfo is FALSE.

13.28.3 cist

TRUE only for CIST state machines; i.e., FALSE for MSTI state machine instances.

13.28.4 cistRootPort

TRUE if the CIST role for the given port is RootPort.

13.28.5 cistDesignatedPort

TRUE if the CIST role for the given port is DesignatedPort.

13.28.6 EdgeDelay

Returns the value of MigrateTime if operPointToPointMAC is TRUE, and the value of MaxAge otherwise.

13.28.7 forwardDelay

Returns the value of HelloTime if sendRSTP is TRUE, and the value of FwdDelay otherwise.

13.28.8 FwdDelay

The Forward Delay component of the CIST's designatedTimes parameter (13.27.7).

13.28.9 HelloTime

The Hello Time component of the CIST's portTimes parameter (13.27.33) with the recommended default value given in Table 13-5.

13.28.10 MaxAge

The Max Age component of the CIST's designatedTimes parameter (13.27.7).

13.28.11 mstiDesignatedOrTCpropagatingRootPort

TRUE if the role for any MSTI for the given port is either:

a) DesignatedPort; or
b) RootPort, and the instance for the given MSTI and port of the tcWhile timer is not zero.

13.28.12 mstiMasterPort

TRUE if the role for any MSTI for the given port is MasterPort.

13.28.13 rcvdAnyMsg

TRUE for a given port if rcvdMsg is TRUE for the CIST or any MSTI for that port.

13.28.14 rcvdCistMsg

TRUE for a given port if and only if rcvdMsg is TRUE for the CIST for that port.

13.28.15 rcvdMstiMsg

TRUE for a given port and MSTI if and only if rcvdMsg is FALSE for the CIST for that port and rcvdMsg is TRUE for the MSTI for that port.

13.28.16 reRooted

TRUE if the rrWhile timer is clear (zero) for all Ports for the given tree other than the given Port.

13.28.17 rstpVersion

TRUE if ForceProtocolVersion (13.7.2) is greater than or equal to 2.

13.28.18 stpVersion

TRUE if Force Protocol Version (13.7.2) is less than 2.

13.28.19 updtCistInfo

TRUE for a given port if and only if updtInfo is TRUE for the CIST for that port.

13.28.20 updtMstiInfo

TRUE for a given port and MSTI if and only if updtInfo is TRUE for the MSTI for that port or updtInfo is TRUE for the CIST for that port.

NOTE—The dependency of rcvdMstiMsg and updtMstiInfo on CIST variables for the port reflects the fact that MSTIs exist in a context of CST parameters. The state machines ensure that the CIST parameters from received BPDUs are processed and updated prior to processing MSTI information.

13.29 State machine procedures

The following procedures perform the functions specified for both the CIST and the MSTI state machines or specifically for the CIST or a given MSTI:

a) betterorsameInfo(newInfoIs) (13.29.1)
b) checkBPDUconsistency() (13.29.2)
c) clearAllRcvdMsgs() (13.29.3)
d) clearReselectTree() (13.29.4)
e) disableForwarding(13.29.5)
f) disableLearning(13.29.6)
g) enableForwarding(13.29.7)

h) enableLearning(13.29.8)
i) fromSameRegion() (13.29.9)
j) newTcDetected() (13.29.10)
k) newTcWhile() (13.29.11)
j) checkBPDUConsistency() (13.29.2)
k) preparePseudoInfo() (13.29.12)
l) rcvInfo() (13.29.13)
m) recordAgreement() (13.29.14)
n) recordDispute() (13.29.15)
o) recordMastered() (13.29.16)
p) recordPriority()
q) recordProposal() (13.29.18)
r) recordTimes() (13.29.19)
s) setRcvdMsgs() (13.29.20)
t) setReRootTree(13.29.21)
u) setSelectedTree() (13.29.22)
v) setSyncTree() (13.29.23)
w) setTcFlags() (13.29.24)
x) setTcPropTree() (13.29.25)
y) syncMaster() (13.29.26)
z) txConfig() (13.29.27)
aa) txMstp() (13.29.28)
ab) txRstp() (13.29.29)
ac) txTcn() (13.29.30)
ad) updtBPDUVersion() (13.29.31)
ae) updtRcvdInfoWhile() (13.29.32)
af) updtRolesTree() (13.29.33)
ag) updtRolesDisabledTree() (13.29.34)

All references to named variables in the specification of procedures are to instances of the variables corresponding to the instance of the state machine using the function, i.e., to the CIST or the given MSTI as appropriate. References to forwarding and learning apply to frames assigned to the specified tree.

13.29.1 betterorsameInfo(newInfoIs)

Returns TRUE if, for a given port and tree (CIST, or MSTI), either

a) The procedure's parameter newInfoIs is Received, and infoIs is Received and the msgPriority vector is better than or the same as (13.10) the portPriority vector; or,
b) The procedure's parameter newInfoIs is Mine, and infoIs is Mine and the designatedPriority vector is better than or the same as (13.10) the portPriority vector.

Returns False otherwise.

NOTE—This procedure is not invoked (in the case of a MSTI) if the received BPDU carrying the MSTI information was received from another MST Region. In that event, the Port Receive Machine (using setRcvdMsgs()) does not set rcvdMsg for any MSTI, and the Port Information Machine's SUPERIOR_DESIGNATED state is not entered.

13.29.2 checkBPDUConsistency()

This procedure compares the message priority vector of the information received on the port with the port priority vector for the port, and:

a) If the received message priority vector is superior to the port priority vector; and

b) The BPDU is an STP BPDU (version 0 or version 1); or

c) The BPDU is an RST or MST BPDU (type 2, version 2 or above) and its Port Role values indicates Designated and its Learning flag is set;

then, for the CIST and all the MSTIs on this port the disputed flag is set and agreed is cleared.

13.29.3 clearAllRcvdMsgs()

Clears rcvdMsg for the CIST and all MSTIs, for this port.

13.29.4 clearReselectTree()

Clears reselect for the tree (the CIST or a given MSTI) for all ports of the bridge.

13.29.5 disableForwarding()

An implementation dependent procedure that causes the Forwarding Process (7.7) to stop forwarding frames through the Port. The procedure does not complete until forwarding has stopped.

13.29.6 disableLearning()

An implementation dependent procedure that causes the Learning Process (7.8) to stop learning from the source address of frames received on the Port. The procedure does not complete until learning has stopped.

13.29.7 enableForwarding()

An implementation dependent procedure that causes the Forwarding Process (7.7) to start forwarding frames through the Port. The procedure does not complete until forwarding has been enabled.

13.29.8 enableLearning()

An implementation dependent procedure that causes the Learning Process (7.8) to start learning from frames received on the Port. The procedure does not complete until learning has been enabled.

13.29.9 fromSameRegion()

Returns TRUE if rcvdRSTP is TRUE, and the received BPDU conveys a MST Configuration Identifier that matches that held for the bridge. Returns FALSE otherwise.

13.29.10 newTcDetected()

If the value of tcDetected is zero and sendRSTP is TRUE, this procedure sets the value of tcDetected to HelloTime plus one second. The value of HelloTime is taken from the CIST's portTimes parameter (13.27.33) for this port.

If the value of tcDetected is zero and sendRSTP is FALSE, this procedure sets the value of tcDetected to the sum of the Max Age and Forward Delay components of rootTimes.

Otherwise the procedure takes no action.

13.29.11 newTcWhile()

If the value of tcWhile is zero and sendRSTP is TRUE, this procedure sets the value of tcWhile to HelloTime plus one second and sets either newInfo TRUE for the CIST or newInfoMsti TRUE for a given MSTI. The value of HelloTime is taken from the CIST's portTimes parameter (13.27.33) for this port.

If the value of tcWhile is zero and sendRSTP is FALSE, this procedure sets the value of tcWhile to the sum of the Max Age and Forward Delay components of rootTimes and does not change the value of either newInfo or newInfoMsti.

Otherwise the procedure takes no action.

13.29.12 preparePseudoInfo()

Using local parameters, this procedure creates a Pseudo Info BPDU that will be presented to the spanning tree as if it had been received from the physical port on which it was invoked. The components of this BPDU are as follows:

a) This is an MST BPDU: Protocol Identifier 0, Protocol Version Identifier 3 and BPDU Type 2;
b) Message Age, Max Age, Hello Time and Forward Delay are derived from BridgeTimes (13.26.3);
c) The CIST information carries the message priority vector (13.10) with a value of {pseudoRootId, 0, pseudoRootId, 0, 0, 0};
d) CIST Flags with the Port Role flags indicating Designated, and the Learning and Forwarding flags set;
e) The Version 1 Length is 0 and Version 3 Length calculated appropriately;
f) For each MSTI configured on the bridge, the corresponding MSTI Configuration Message carries:
    1) a message priority vector with a value of {pseudoRootId, 0, 0, 0};
    2) MSTI Flags with the Port Role flags indicating Designated, and the Learning and Forwarding flags set;
    3) MSTI Remaining Hops set to the value of the MaxHops component of BridgeTimes (13.26.3).

13.29.13 rcvInfo()

Decodes received BPDUs. Sets rcvdTcn and sets rcvdTc for each and every MSTI if a TCN BPDU has been received, and extracts the message priority and timer values from the received BPDU storing them in the msgPriority and msgTimes variables.

Returns SuperiorDesignatedInfo if, for a given port and tree (CIST, or MSTI):

a) The received CIST or MSTI message conveys a Designated Port Role, and
    1) The message priority (msgPriority—13.27.24) is superior (13.10 or 13.11) to the port's port priority vector, or
    2) The message priority is the same as the port's port priority vector, and any of the received timer parameter values (msgTimes—13.27.25) differ from those already held for the port (portTimes—13.27.33).

Otherwise, returns RepeatedDesignatedInfo if, for a given port and tree (CIST, or MSTI):

b) The received CIST or MSTI message conveys a Designated Port Role, and
    1) A message priority vector and timer parameters that are the same as the port's port priority vector and timer values; and
    2) infoIs is Received.

Otherwise, returns InferiorDesignatedInfo if, for a given port and tree (CIST, or MSTI):

c) The received CIST or MSTI message conveys a Designated Port Role.

Otherwise, returns InferiorRootAlternateInfo if, for a given port and tree (CIST, or MSTI):

d) The received CIST or MSTI message conveys a Root Port, Alternate Port, or Backup Port Role and a CIST or MSTI message priority that is the same as or worse than the CIST or MSTI port priority vector.

Otherwise, returns OtherInfo.

NOTE—A Configuration BPDU implicitly conveys a Designated Port Role.

13.29.14 recordAgreement()

For the CIST and a given port, if rstpVersion is TRUE, operPointToPointMAC (6.6.3) is TRUE, and the received CIST Message has the Agreement flag set, the CIST agreed flag is set, and the CIST proposing flag is cleared. Otherwise the CIST agreed flag is cleared. Additionally, if the CIST message was received from a bridge in a different MST Region, i.e., the rcvdInternal flag is clear, the agreed and proposing flags for this port for all MSTIs are set or cleared to the same value as the CIST agreed and proposing flags. If the CIST message was received from a bridge in the same MST Region, the MSTI agreed and proposing flags are not changed.

For a given MSTI and port, if operPointToPointMAC (6.6.3) is TRUE, and

a) The message priority vector of the CIST Message accompanying the received MSTI Message (i.e., received in the same BPDU) has the same CIST Root Identifier, CIST External Root Path Cost, and Regional Root Identifier as the CIST port priority vector, and
b) The received MSTI Message has the Agreement flag set,

the MSTI agreed flag is set and the MSTI proposing flag is cleared. Otherwise the MSTI agreed flag is cleared.

NOTE—MSTI Messages received from bridges external to the MST Region are discarded and not processed by recordAgreeement() or recordProposal().

13.29.15 recordDispute()

For the CIST and a given port, if the CIST message has the learning flag set:

a) The disputed variable is set; and
b) The agreed variable is cleared.

Additionally, if the CIST message was received from a bridge in a different MST region (i.e., if the rcvdInternal flag is clear), then for all the MSTIs:

c) The disputed variable is set; and
d) The agreed variable is cleared.

For a given MSTI and port, if the received MSTI message has the learning flag set:

e) The disputed variable is set; and
f) The agreed variable is cleared.

13.29.16 recordMastered()

For the CIST and a given port, if the CIST message was received from a bridge in a different MST Region, i.e. the rcvdInternal flag is clear, the mastered variable for this port is cleared for all MSTIs.

For a given MSTI and port, if the MSTI message was received on a point-to-point link and the MSTI Message has the Master flag set, set the mastered variable for this MSTI. Otherwise reset the mastered variable.

13.29.17 recordPriority()

Sets the components of the portPriority variable to the values of the corresponding msgPriority components.

13.29.18 recordProposal()

For the CIST and a given port, if the received CIST Message conveys a Designated Port Role, and has the Proposal flag set, the CIST proposed flag is set. Otherwise the CIST proposed flag is not changed. Additionally, if the CIST Message was received from a bridge in a different MST Region, i.e., the rcvdInternal flag is clear, the proposed flags for this port for all MSTIs are set or cleared to the same value as the CIST proposed flag. If the CIST message was received from a bridge in the same MST Region, the MSTI proposed flags are not changed.

For a given MSTI and port, if the received MSTI Message conveys a Designated Port Role, and has the Proposal flag set, the MSTI proposed flag is set. Otherwise the MSTI proposed flag is not changed.

13.29.19 recordTimes()

For the CIST and a given port, sets portTimes' Message Age, Max Age, Forward Delay, and remainingHops to the received values held in msgTimes and portTimes' Hello Time to msgTimes' Hello Time if that is greater than the minimum specified in the Compatibility Range column of Table 17-1 of IEEE Std 802.1D, and to that minimum otherwise.

For a given MSTI and port, sets portTime's remainingHops to the received value held in msgTimes.

13.29.20 setRcvdMsgs()

Sets rcvdMsg for the CIST, and makes the received CST or CIST message available to the CIST Port Information state machines.

Additionally, and if and only if rcvdInternal is set, sets rcvdMsg for each and every MSTI for which a MSTI message is conveyed in the BPDU, and makes available each MSTI message and the common parts of the CIST message priority (the CIST Root Identifier, External Root Path Cost, and Regional Root Identifier) to the Port Information state machine for that MSTI.

13.29.21 setReRootTree()

Sets reRoot TRUE for this tree (the CIST or a given MSTI) for all ports of the bridge.

13.29.22 setSelectedTree()

Sets selected TRUE for this tree (the CIST or a given MSTI) for all ports of the bridge if reselect is FALSE for all ports in this tree. If reselect is TRUE for any port in this tree, this procedure takes no action.

13.29.23 setSyncTree()

Sets sync TRUE for this tree (the CIST or a given MSTI) for all ports of the bridge.

13.29.24 setTcFlags()

For the CIST and a given port:

a) If the Topology Change Acknowledgment flag is set for the CIST in the received BPDU, sets rcvdTcAck TRUE.
b) If rcvdInternal is clear and the Topology Change flag is set for the CIST in the received BPDU, sets rcvdTc TRUE for the CIST and for each and every MSTI.
c) If rcvdInternal is set, sets rcvdTc for the CIST if the Topology Change flag is set for the CIST in the received BPDU.

For a given MSTI and port, sets rcvdTc for this MSTI if the Topology Change flag is set in the corresponding MSTI message.

13.29.25 setTcPropTree()

If and only if restrictedTcn is FALSE for the port that invoked the procedure, sets tcProp TRUE for the given tree (the CIST or a given MSTI) for all other ports.

13.29.26 syncMaster()

For all MSTIs, for each port that has infoInternal set:

a) Clears the agree, agreed, and synced variables; and
b) Sets the sync variable.

13.29.27 txConfig()

Transmits a Configuration BPDU. The first four components of the message priority vector (13.27.24) conveyed in the BPDU are set to the value of the CIST Root Identifier, External Root Path Cost, Bridge Identifier, and Port Identifier components of the CIST's designatedPriority parameter (13.27.6) for this port. The topology change flag is set if (tcWhile != 0) for the port. The topology change acknowledgment flag is set to the value of TcAck for the port. The remaining flags are set to zero. The value of the Message Age, Max Age, and Fwd Delay parameters conveyed in the BPDU are set to the values held in the CIST's designatedTimes parameter (13.27.7) for the port. The value of the Hello Time parameter conveyed in the BPDU is set to the value held in the CIST's portTimes parameter (13.27.33) for the port.

13.29.28 txMstp()

Transmits a MST BPDU (14.5), encoded according to the specification contained in 14.9. The first six components of the CIST message priority vector (13.27.24) conveyed in the BPDU are set to the value of the CIST's designatedPriority parameter (13.27.6) for this port. The Port Role in the BPDU (14.2.1) is set to the current value of the role variable for the transmitting port (13.27.50). The Agreement and Proposal flags in the BPDU are set to the values of the agree (13.27 of this standard, 17.19 of IEEE Std 802.1D) and proposing (13.27 of this standard, 17.19.24 of IEEE Std 802.1D) variables for the transmitting port, respectively. The CIST topology change flag is set if (tcWhile != 0) for the port. The topology change acknowledge flag in the BPDU is never used and is set to zero. The learning and forwarding flags in the BPDU are set to the values of the learning (13.27 of this standard, 17.19.12 of IEEE Std 802.1D) and forwarding (13.27 of this standard, 17.19.9 of IEEE Std 802.1D) variables for the CIST, respectively. The value of the Message Age, Max Age, and Fwd Delay parameters conveyed in the BPDU are set to the values held in the CIST's designatedTimes parameter (13.27.7) for the port. The value of the Hello Time parameter conveyed in the BPDU is set to the value held in the CIST's portTimes parameter (13.27.33) for the port.

If the value of the Force Protocol Version parameter is less than 3, no further parameters are encoded in the BPDU and the protocol version parameter is set to 2 (denoting a RST BPDU). Otherwise, the protocol version parameter is set to 3 and the remaining parameters of the MST BPDU are encoded:

a) The version 3 length.
b) The MST Configuration Identifier parameter of the BPDU is the value of MstConfigId (13.26.6).
c) The CIST Internal Root Path Cost (13.27.6).
d) The CIST Bridge Identifier (CIST Designated Bridge Identifier—13.27.6).
e) The CIST Remaining Hops (13.27.7).
f) The parameters of each MSTI message, encoded in MSTID order.

NOTE—No more than 64 MSTIs may be supported. The parameter sets for all of these can be encoded in a standard-sized Ethernet frame.

13.29.29 txRstp()

Transmits an RST BPDU. The values of the message priority vector components conveyed in the BPDU are those of designatedPriority (13.27.6) for this Port. The Port Role in the BPDU (9.3.3) is set to the current value of the role variable for the transmitting port (13.27.50). The Agreement and Proposal flags in the BPDU are set to the values of the agree (13.27.3) and proposing (13.27.35) variables for the transmitting Port, respectively. The topology change flag is set if (tcWhile ! = 0) for the Port. The topology change acknowledge flag in the BPDU is never used and is set to zero. The Learning and Forwarding flags in the BPDU are set to the values of the learning (13.27.20) and forwarding (13.27.15) variables for the transmitting Port, respectively. The value of the Message Age, Max Age, Fwd Delay, and Hello Time parameters conveyed in the BPDU are set to the values held in designatedTimes (13.27.7) for the Port.

13.29.30 txTcn()

Transmits a TCN BPDU.

13.29.31 updtBPDUVersion()

Sets rcvdSTP TRUE if the BPDU received is a version 0 or version 1 TCN or a Config BPDU. Sets rcvdRSTP TRUE if the received BPDU is a RST BPDU or a MST BPDU.

13.29.32 updtRcvdInfoWhile()

Updates rcvdInfoWhile (13.25). The value assigned to rcvdInfoWhile is three times the Hello Time, if either:

a) Message Age, incremented by 1 second and rounded to the nearest whole second, does not exceed Max Age and the information was received from a bridge external to the MST Region (rcvdInternal FALSE);

or

b) remainingHops, decremented by one, is greater than zero and the information was received from a bridge internal to the MST Region (rcvdInternal TRUE);

and is zero otherwise.

The values of Message Age, Max Age, remainingHops, and Hello Time used in these calculations are taken from the CIST's portTimes parameter (13.27.33) and are not changed by this procedure.

13.29.33 updtRolesTree()

This procedure calculates the following priority vectors (13.9, 13.10 for the CIST, 13.11 for a MSTI) and timer values, for the CIST or a given MSTI:

a) The *root path priority vector* for each Bridge Port that is not Disabled and has a *port priority vector* (portPriority plus portId—see 13.27.32 and 13.27.31) that has been recorded from a received message and not aged out (infoIs == Received); and
b) The Bridge's *root priority vector* (rootPortId, rootPriority—13.26.7, 13.26.8), chosen as the best of the set of priority vectors comprising the bridge's own *bridge priority vector* (BridgePriority—13.26.2) plus all calculated root path priority vectors whose:
    1) DesignatedBridgeID Bridge Address component is not equal to that component of the bridge's own bridge priority vector (13.10) and,
    2) Port's restrictedRole parameter is FALSE; and
c) The bridge's *root times*, (rootTimes—13.26.9), set equal to:
    1) BridgeTimes (13.26.3), if the chosen root priority vector is the bridge priority vector; otherwise,
    2) portTimes (13.27.33) for the port associated with the selected root priority vector, with the Message Age component incremented by 1 second and rounded to the nearest whole second if the information was received from a bridge external to the MST Region (rcvdInternal FALSE), and with remainingHops decremented by one if the information was received from a bridge internal to the MST Region (rcvdInternal TRUE).
d) The *designated priority vector* (designatedPriority—13.27.6) for each port; and
e) The *designated times* (designatedTimes—13.27.7) for each port set equal to the value of *root times*.

If the root priority vector for the CIST is recalculated, and has a different Regional Root Identifier than that previously selected, and has or had a non-zero CIST External Root Path Cost, the syncMaster() procedure (13.29.26) is invoked.

NOTE—Changes in Regional Root Identifier will not cause loops if the Regional Root is within an MST Region, as is the case if and only if the MST Region is the Root of the CST. This important optimization allows the MSTIs to be fully independent of each other in the case where they compose the core of a network.

The CIST or MSTI port role for each port is assigned, and its port priority vector and timer information are updated as follows:

f) If the port is Disabled (infoIs = Disabled), selectedRole is set to DisabledPort.
g) Otherwise, if this procedure is invoked for a given MSTI:
    1) If the port is not Disabled, the selected CIST Port Role (calculated for the CIST prior to invoking this procedure for a given MSTI) is RootPort, and the CIST port priority information was received from a bridge external to the MST Region (infoIs == Received and infoInternal == FALSE), selectedRole is set to MasterPort. Additionally, updtInfo is set if the port priority vector differs from the designated priority vector or the port's associated timer parameter differs from the one for the Root Port;
    2) If the port is not Disabled, the selected CIST Port Role (calculated for the CIST prior to invoking this procedure for a given MSTI) is AlternatePort, and the CIST port priority information was received from a bridge external to the MST Region (infoIs == Received and infoInternal == FALSE), selectedRole is set to AlternatePort. Additionally, updtInfo is set if the port priority vector differs from the designated priority vector or the port's associated timer parameter differs from the one for the Root Port.

Otherwise, for each port of the CIST, or for each port of a given MSTI that is not Disabled and whose CIST port priority information was not received from a bridge external to the Region (infoIs != Received or infoInternal == TRUE), the CIST or MSTI port role for each port is assigned, and its port priority vector and timer information are updated as follows:

h) If the port priority vector information was aged (infoIs = Aged), updtInfo is set and selectedRole is set to DesignatedPort;

i) If the port priority vector was derived from another port on the bridge or from the bridge itself as the Root Bridge (infoIs = Mine), selectedRole is set to DesignatedPort. Additionally, updtInfo is set if the port priority vector differs from the designated priority vector or the port's associated timer parameter(s) differ(s) from the Root Port's associated timer parameters;

j) If the port priority vector was received in a Configuration Message and is not aged (infoIs == Received), and the root priority vector is now derived from it, selectedRole is set to RootPort, and updtInfo is reset;

k) If the port priority vector was received in a Configuration Message and is not aged (infoIs == Received), the root priority vector is not now derived from it, the designated priority vector is not better than the port priority vector, and the designated bridge and designated port components of the port priority vector do not reflect another port on this bridge, selectedRole is set to AlternatePort, and updtInfo is reset;

l) If the port priority vector was received in a Configuration Message and is not aged (infoIs == Received), the root priority vector is not now derived from it, the designated priority vector is not better than the port priority vector, and the designated bridge and designated port components of the port priority vector reflect another port on this bridge, selectedRole is set to BackupPort, and updtInfo is reset;

m) If the port priority vector was received in a Configuration Message and is not aged (infoIs == Received), the root priority vector is not now derived from it, the designated priority vector is better than the port priority vector, selectedRole is set to DesignatedPort, and updtInfo is set.

13.29.34 uptRolesDisabledTree()

This procedure sets selectedRole to DisabledPort for all ports of the bridge for a given tree (CIST or MSTI).

13.30 The Port Timers state machine

The Port Timers state machine shall implement the function specified by the state diagram in Figure 13-15 and the attendant definitions contained in 13.25 through 13.29.

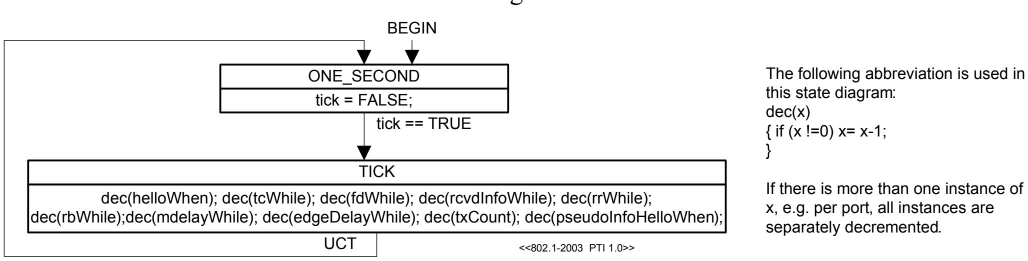


Figure 13-15—Port Timers state machine

The state machine uses tick (13.27), a signal set by an implementation-specific system clock function at one second intervals, to decrement the timer variables for the CIST and all MSTIs for the port. The state machine that uses a given timer variable is responsible for setting its initial value.

13.31 Port Receive state machine

The Port Receive state machine shall implement the function specified by the state diagram contained in Figure 13-16 and the attendant definitions contained in 13.25 through 13.29.

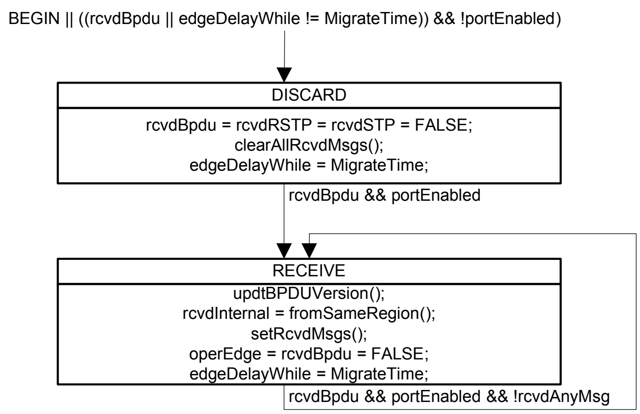


Figure 13-16—Port Receive state machine

This state machine is responsible for receiving each BPDU. The next BPDU is not processed until all rcvdMsg flags have been cleared by the per-tree state machines.

13.32 Port Protocol Migration state machine

The Port Protocol Migration state machine shall implement the function specified by the state diagram contained in Figure 13-18 and the attendant definitions contained in 13.25 through 13.29.

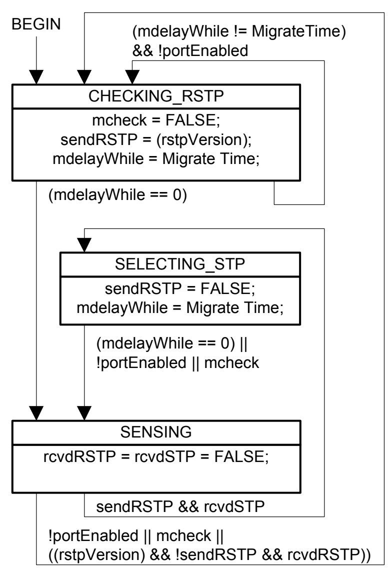


Figure 13-17—Port Protocol Migration state machine

13.33 Bridge Detection state machine

The Bridge Detection state machine shall implement the function specified by the state diagram contained in Figure 13-18 and the attendant definitions contained in 13.25 through 13.29.

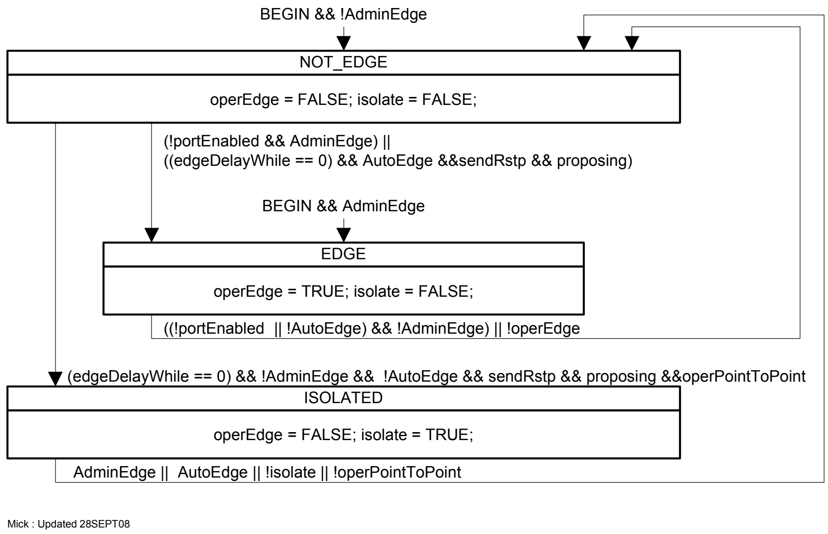


Figure 13-18—Bridge Detection state machine

13.34 Port Transmit state machine

The Port Transmit state machine shall implement the function specified by the state diagram contained in Figure 13-19 and the attendant definitions contained in 13.25 through 13.29.

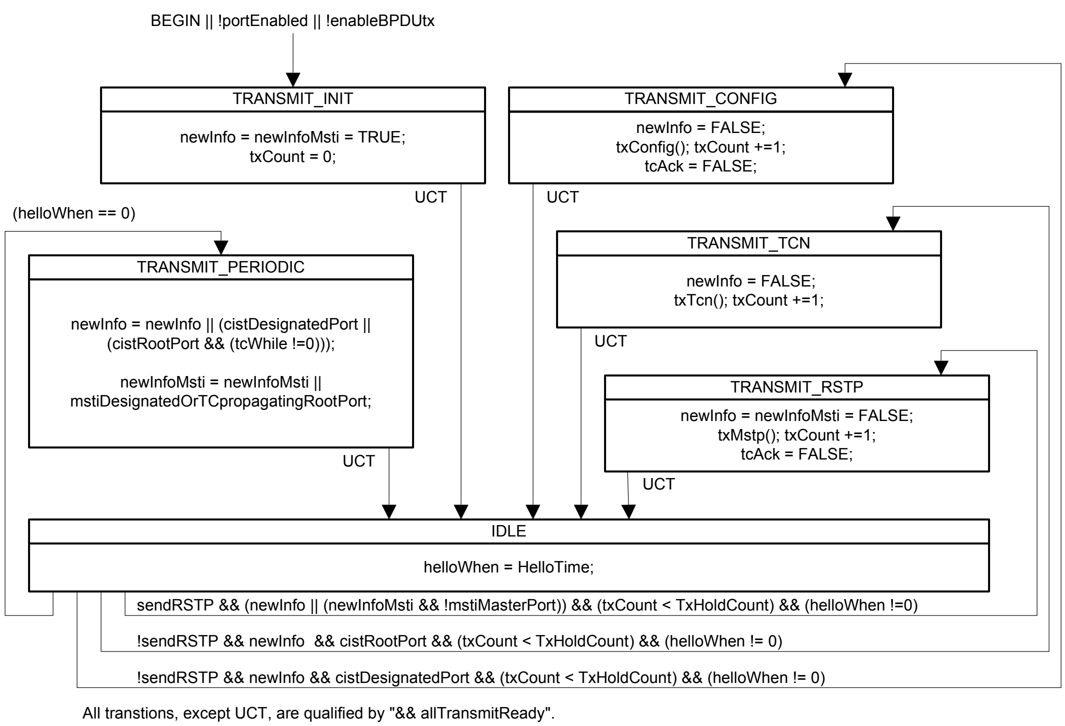


Figure 13-19—Port Transmit state machine

This state machine is responsible for transmitting BPDUs.

NOTE 1—Any single received BPDU that changes the CIST Root Identifier, CIST External Root Path Cost, or CIST Regional Root associated with MSTIs should be processed entirely, or not at all, before encoding BPDUs for transmission.This recommendation minimizes the number of BPDUs to be transmitted following receipt of a BPDU with new information. It is not required for correctness and has not therefore been incorporated into the state machines.

NOTE 2—If a CIST state machine sets newInfo, this machine will ensure that a BPDU is transmitted conveying the new CIST information. If MST BPDUs can be transmitted through the port, this BPDU will also convey new MSTI information for all MSTIs. If a MSTI state machine sets newInfoMsti, and MST BPDUs can be transmitted through the port, this machine will ensure that a BPDU is transmitted conveying information for the CIST and all MSTIs. Separate newInfo and newInfoMsti variables are provided to avoid requiring useless transmission of a BPDU through a port that can only transmit STP BPDUs (as required by the Force Protocol Version parameter or Port Protocol Migration machine) following a change in MSTI information without any change to the CIST.

13.35 Port Information state machine

The Port Information state machine for each tree shall implement the function specified by the state diagram contained in Figure 13-20 and the attendant definitions contained in 13.25 through 13.29.

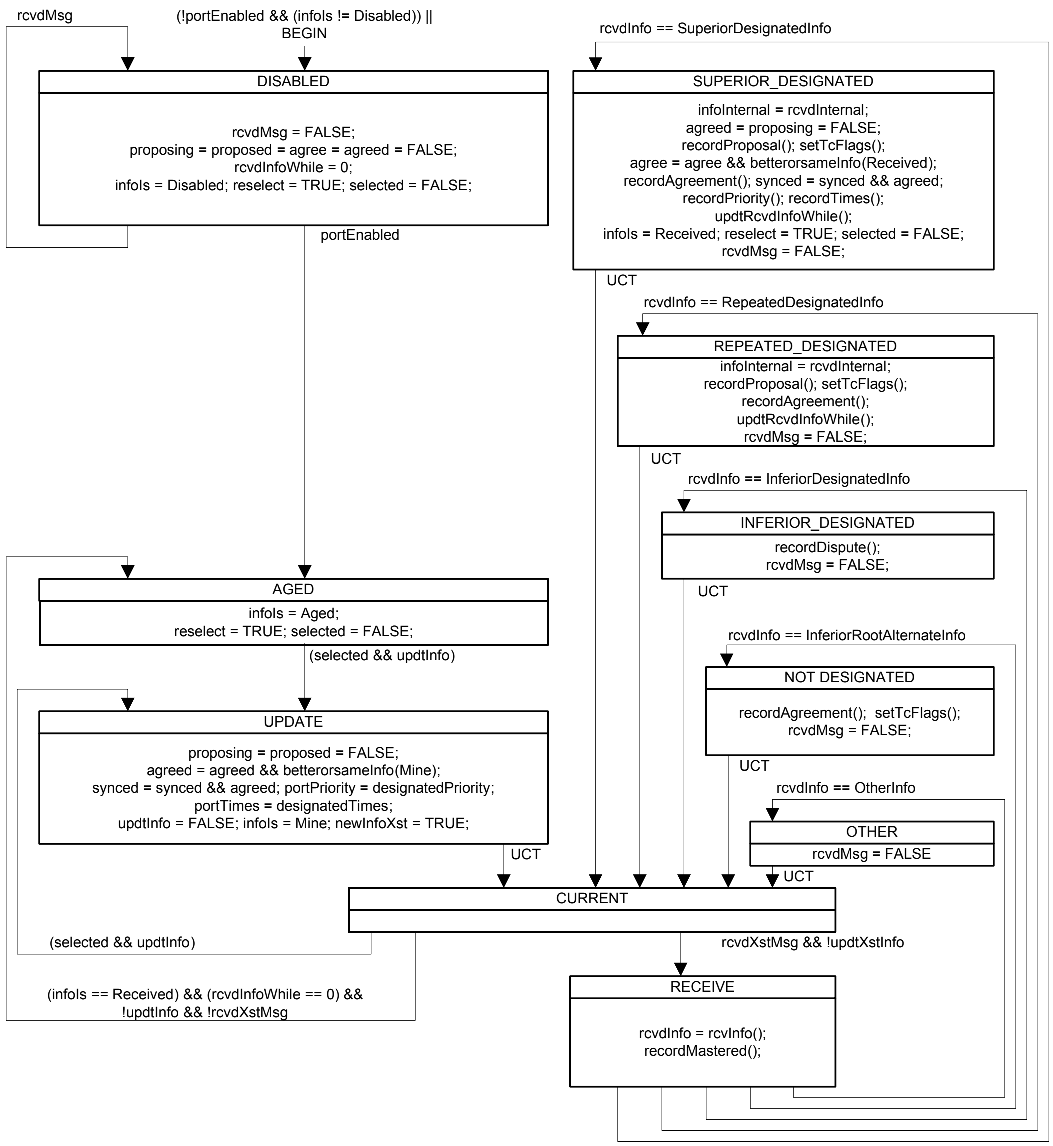


Figure 13-20—Port Information state machine

This state machine is responsible for recording the spanning tree information currently in use by the CIST or a given MSTI for a given port, ageing that information out if it was derived from an incoming BPDU, and recording the origin of the information in the infoIs variable. The selected variable is cleared and reselect set to signal to the Port Role Selection machine that port roles need to be recomputed. The infoIs and portPriority variables from all ports are used in that computation and, together with portTimes, determine new values of designatedPriority and designatedTimes. The selected variable is set by the Port Role Selection machine once the computation is complete.

13.36 Port Role Selection state machine

The Port Role Selection state machine shall implement the function specified by the state diagram contained in Figure 13-21 and the attendant definitions contained in 13.25 through 13.29.

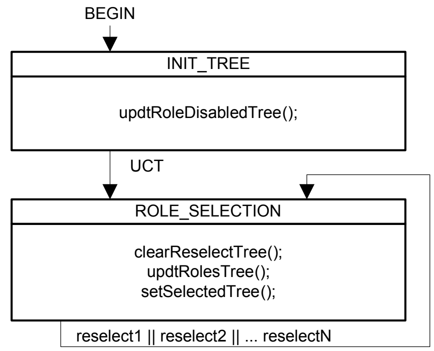


Figure 13-21—Port Role Selection state machine

13.37 Port Role Transitions state machine

The Port Role Transitions state machine shall implement the function specified by the state diagram contained in the following figures:

a) Part 1: Figure 13-22 for both the initialization of this state machine and the states associated with the DisabledPort role; and
b) Part 2: Figure 13-23 for the states associated with the MasterPort role; and
c) Part 3: Figure 13-24 for the states associated with the RootPort role; and
d) Part 4: Figure 13-25 for the states associated with the DesignatedPort role; and
e) Part 5: Figure 13-26 for the states associated with the AlternatePort and BackupPort roles;

and the attendant definitions contained in 13.25 through 13.29.

As Figure 13-22, Figure 13-23, Figure 13-24, Figure 13-25, and Figure 13-26 are component parts of the same state machine, the global transitions associated with these diagrams are possible exit transitions from the states shown in any of the diagrams.

Figure 13-22 and Figure 13-26 show the Port Roles for ports that do not form part of the active topology of the given tree.

Figure 13-23, Figure 13-24, and Figure 13-25 show the Port Roles that form part of the active topology.

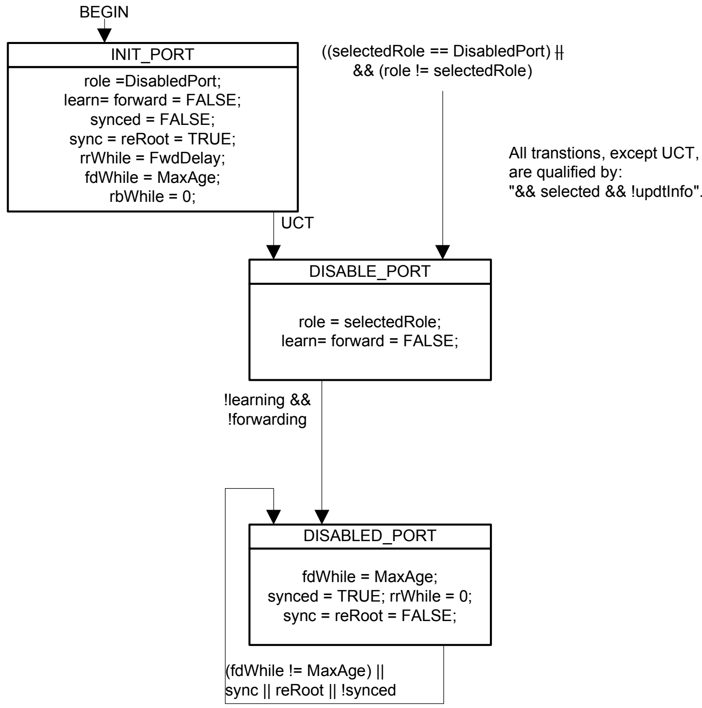


Figure 13-22—Disabled Port role transitions

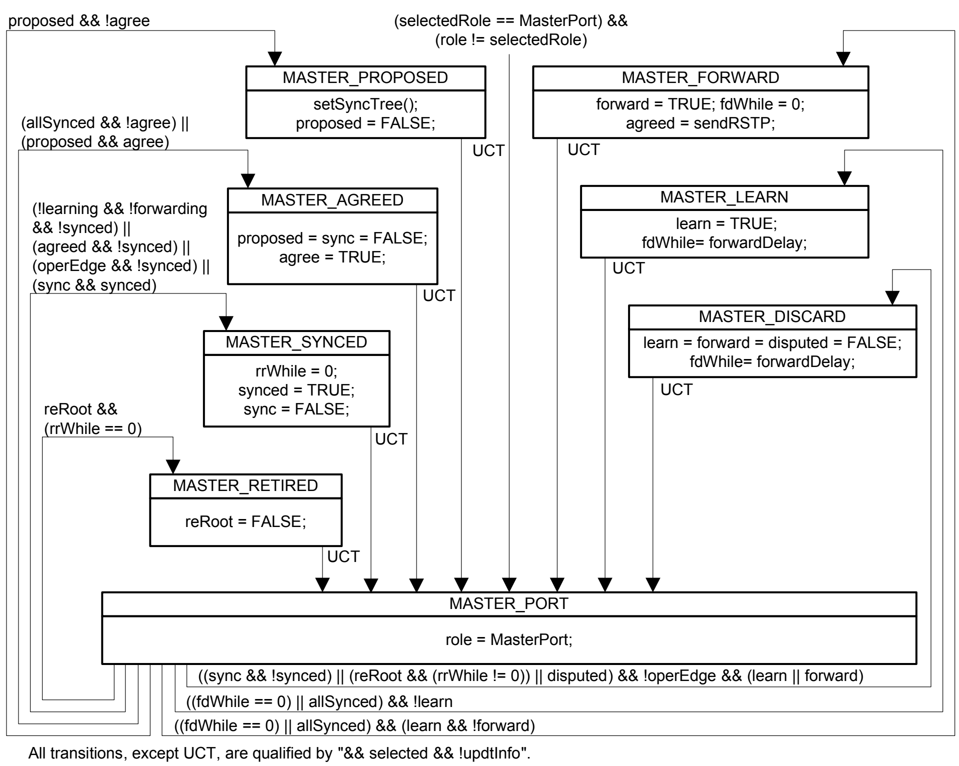
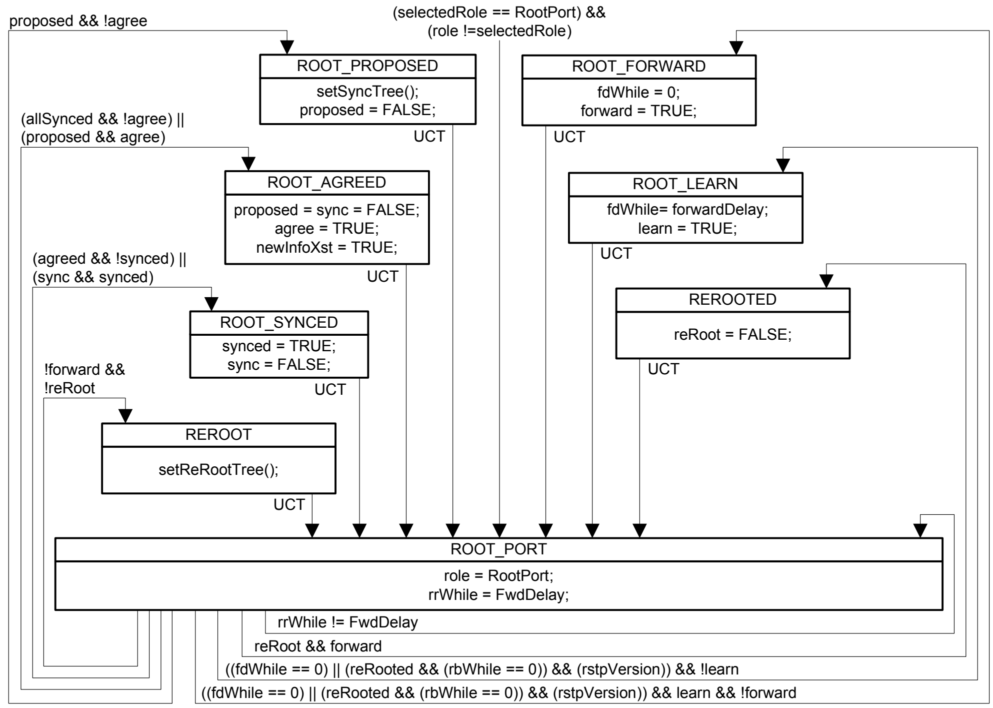


Figure 13-23—Port Role Transitions state machine—MasterPort



All transitions, except UCT, are qualified by "&& selected && !updtInfo".

Figure 13-24—Port Role Transitions state machine—RootPort

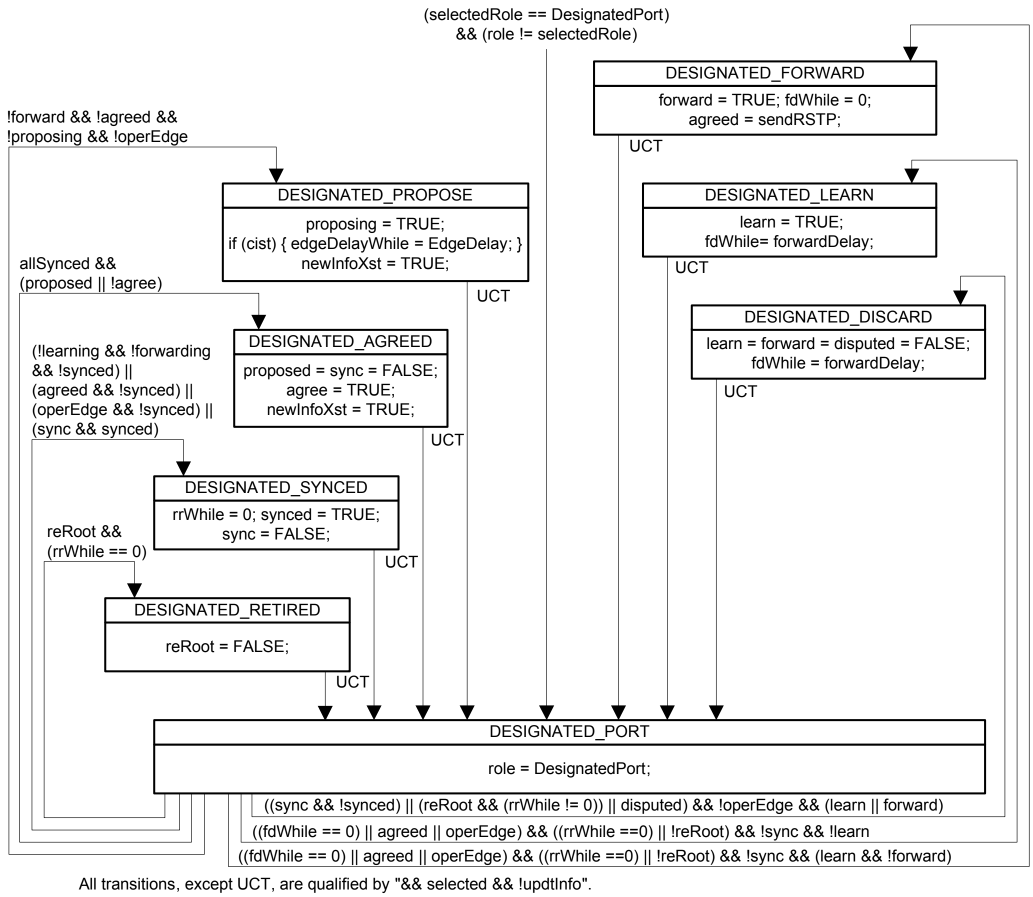


Figure 13-25—Port Role Transitions state machine—DesignatedPort

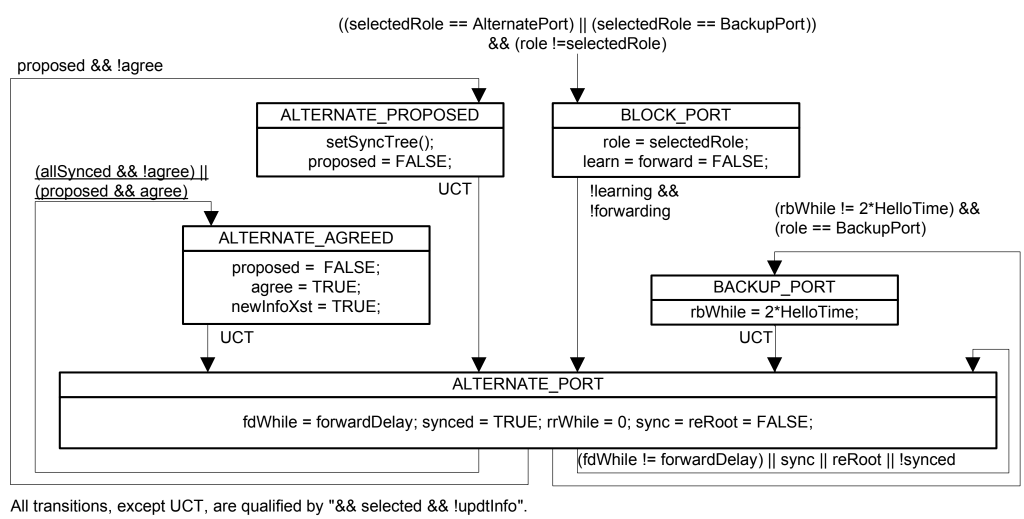


Figure 13-26—Port Role Transitions state machine—AlternatePort and BackupPort

13.38 Port State Transition state machine

The Port State Transition state machine shall implement the function specified by the state diagram contained in Figure 13-27 and the attendant definitions contained in 13.25 through 13.29.

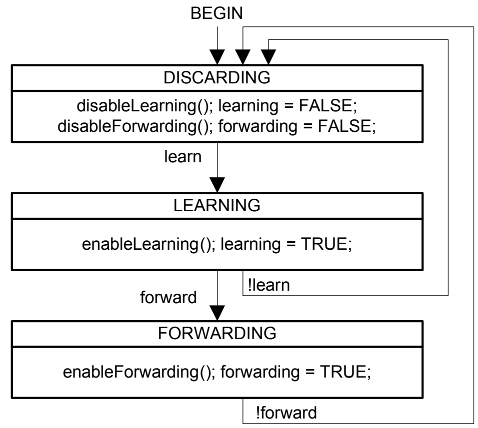


Figure 13-27—Port State Transition state machine

NOTE—A small system-dependent delay may occur on each of the transitions shown in the referenced state machine.

13.39 Topology Change state machine

The Topology Change state machine for each tree shall implement the function specified by the state diagram contained in Figure 13-28 and the attendant definitions contained in 13.25 through 13.29.

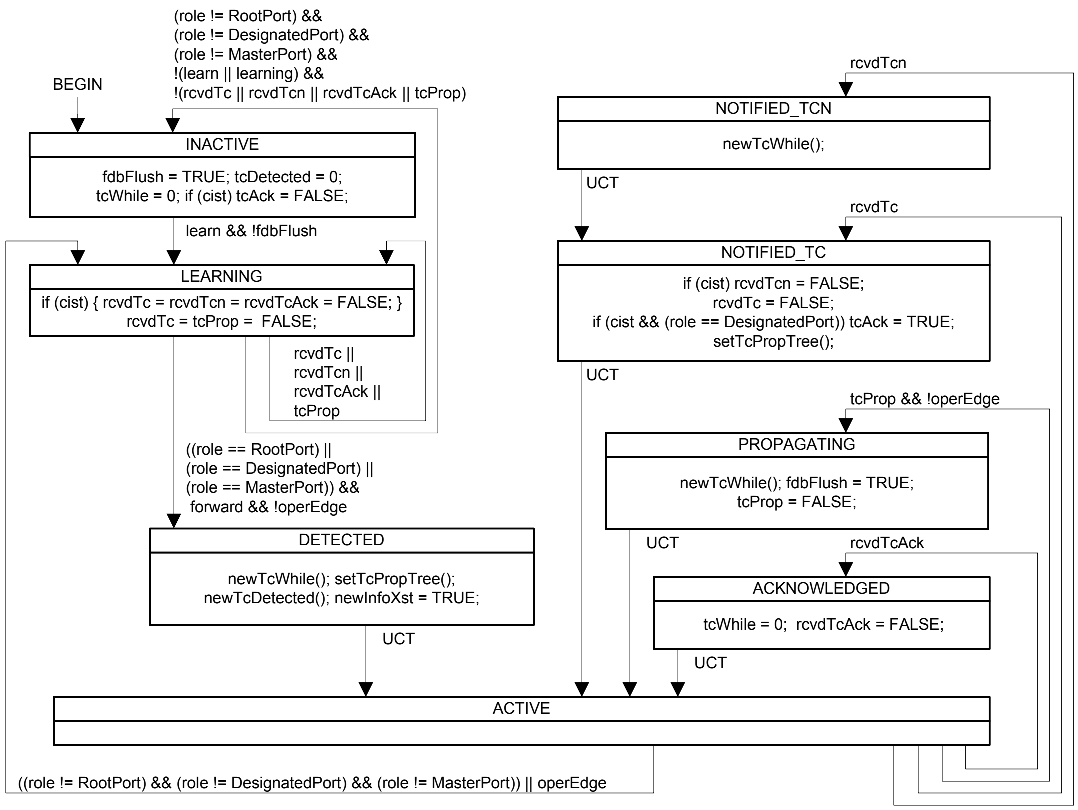


Figure 13-28—Topology Change state machine

NOTE—MRP (Clause 10) uses the tcDetected variable maintained by this state machine.

13.40 L2 Gateway Port Receive state machine

If implemented, the L2 Gateway Port state machine for each port shall implement the function specified by the state diagram contained in Figure 13-29 and the attendant definitions contained in 13.25 through 13.29. When activated, by setting isL2gp, it simulates continuous reception of BPDU carrying a spanning tree priority vector based on a configured pseudoRootID (13.27.36). As a result, an L2 Gateway Port that provides connectivity to a given service instance (***3.117***, ***3.7***) can only play one of two different roles:

a) If the information is the best in the instance, the Layer Two Gateway Port is elected root port and will be in forwarding state; <<This is just utterly and completely wrong. It’s backwards. If the information is better than any on the attached customer network the L2GP will be elected Rot Port and will be Forwarding.>>
b) Else, the port will have its disputed flag set and will remain in designated role discarding state.

A maximum of one Layer Two Gateway Port can thus be forwarding at a given time, provided its pseudo information is the best information advertised in the instance. A bridged network can be redundantly attached to a Provider Backbone Bridged Network (PBBN) (Clause 26) by the means of several Layer Two Gateway Ports. By way of communication within their region and without any influence from the PBBN, those ports will provide connectivity to the PBBN through a single Layer Two Gateway Port, thus avoiding any bridging loop between the PBBN and the instance.

Proper configuration of a Layer Two Gateway Port requires setting the isL2gp and clearing the enableBPDUtx on that port. The variable enableBPDUrx for the port can be cleared on the port to provide complete independence from the information received from the PBBN. Else, the state machine ensures that the CIST information received is inferior to the pseudo information the Layer Two Gateway Port presents to itself on the CIST, and will block all instances (CIST and MSTIs) otherwise. This mechanism will prevent a misconfiguration from introducing a loop in the instance, but adds a dependency to the information circulating outside of the region.

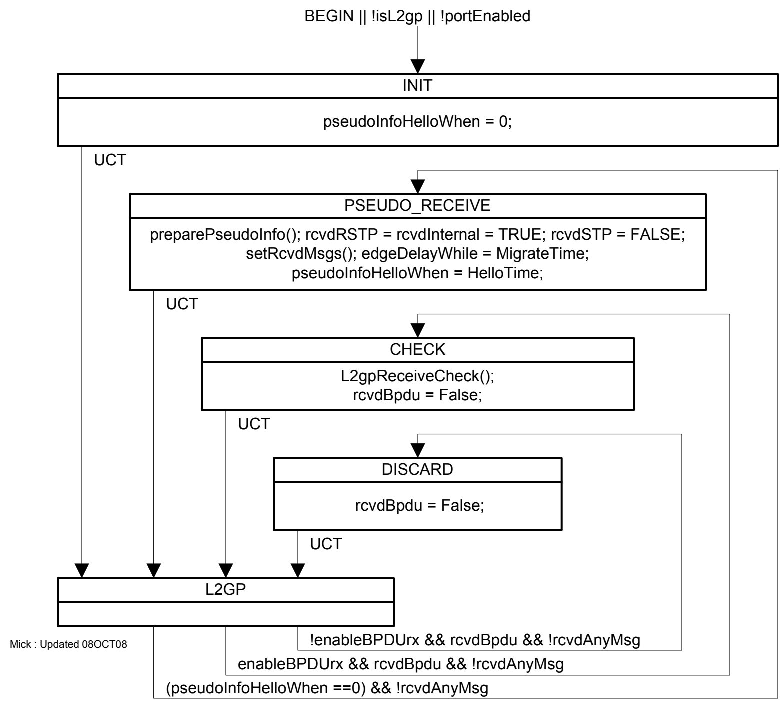


Figure 13-29—L2 Gateway Port Receive state machine

13.41 Customer Edge Port Spanning Tree operation

This subclause specifies the operation of the Spanning Tree Protocol Entity within a C-VLAN component that supports a Customer Edge Port (Figure 15-4) of a Provider Edge Bridge. The Customer Edge Port and each Provider Edge Port are treated as separate Bridge Ports by the spanning tree protocol.

If the C-VLAN component connects to the S-VLAN component with a single Provider Edge Port, and the associated service instance supports no more than two customer interfaces, then all frames (including Spanning Tree BPDUs) addressed to the Bridge Group Address may be relayed between the two ports of the C-VLAN component without modification. Otherwise, the Spanning Tree Protocol Entity shall execute RSTP, as modified by the provisions of this clause (13.41).

The RSTP enhancements specified do not reduce Provider Bridged Network connectivity between Customer Edge Ports to a single spanning tree of service instances but ensure that connectivity for frames assigned to any given C-VLAN is loop free. In this respect, the C-VLAN component's spanning tree protocol operation is equivalent to, but simpler to manage than, the operation of MSTP.

13.41.1 Provider Edge Port operPointToPointMAC and operEdge

The value of the adminPointToPointMAC parameter for a Provider Edge Port is always Auto, and no management control over its setting is provided. The value of the operPointToPointMAC parameter, used by the RSTP state machines, shall be true if the service instance corresponding to the Provider Edge Port connects at most two customer interfaces, and false otherwise.

The value of the adminEdge, autoEdge, and operEdge parameters for a Provider Edge Port are always false, true, and false, respectively. No management control over their setting is provided.

13.41.2 updtRolesTree()

The spanning tree priority vectors and timer values are calculated as specified for the updtRolesTree() procedure in IEEE Std 802.1D. The port role for each port, its port priority vector, and timer information are also updated as specified by IEEE Std 802.1D, with one exception. If selectedRole was to be set to AlternatePort, the port is an Provider Edge Port, and the root priority vector was derived from another Provider Edge Port, then the selectedRole shall be set to Root Port.

NOTE—The effect of this enhancement is to allow the C-VLAN component to have multiple Root Ports (just as if separate per S-VLAN trees were being provided), if they are all Provider Edge Ports. As the C-VLAN component assigns each frame to a single C-VLAN and maps any given C-VLAN to and from at most one Provider Edge Port, no loop is created.

13.41.3 setReRootTree(), setSyncTree(), setTcPropTree()

IEEE Std 802.1D specifies that the setReRootTree() and setSyncTree() procedures set the reRoot and sync variables for all ports of the bridge, and the setTcPropTree() sets the tcProp variable for all ports other than the port that invoked the procedure. If the port invoking the procedure is a Customer Edge Port, then this behavior is unchanged; if it is a Provider Edge Port, then the behavior of each procedure shall be as follows.

The setReRootTree() procedure sets reRoot for the port invoking the procedure and for the Customer Edge Port.

The setSyncTree() procedure sets sync for the port invoking the procedure and for the Customer Edge Port.

The setTcPropTree() procedure sets tcProp for the Customer Edge Port.

13.41.4 allSynced, reRooted

RSTP specifies a single value of the allSynced and reRooted state machine conditions for all Bridge Ports. This specification requires an independent value of each of these conditions for each port of the C-VLAN component. If that port is the Customer Edge Port, then allSynced shall be true if and only if synced is true for all Provider Edge Ports, and reRooted shall be true if and only if rrWhile is zero for all Provider Edge Ports. If the port for which the condition is being evaluated is a Provider Edge Port, then allSynced shall take the value of synced for the Customer Edge Port, and reRooted shall be true if and only if rrWhile is zero for the Customer Edge Port.

13.41.5 Configuration parameters

All configuration parameters for RSTP should be set to their recommended defaults, with the exception of the following, which are chosen to minimize the chance of interfering with the customer's configuration (e.g., by the C-VLAN component becoming the root of the customer spanning tree), as follows:

c) The Bridge Priority (13.18, Table 13-3, 13.26.1) should be set to 61 440. This sets the priority part of the Bridge Identifier (the most significant 4 bits) to hex F.
d) The following 12 bits (the Bridge Identifier system ID extension) should be set to hex FFF.
e) The Port Priority (13.18, Table 13-3, 13.27.32) should be set to 32. This sets the priority part of the Port Identifier (the most significant 4 bits) to hex 2, a higher priority than the default (128, or hex 8).
f) The Port Path Cost values for Provider Edge Ports should be set are to 128.

All BPDUs generated by the Spanning Tree Protocol Entity within a C-VLAN component use the MAC address of the Customer Edge Port as a source address. For each internal Provider Edge Port, the protocol uses the S-VID associated with the corresponding internal Customer Network Port on the S-VLAN component as a port number. For the Customer Edge Port, the value 0xFFF is used as the port number.

13.42 Virtual Instance Port Spanning Tree operation

This subclause specifies the operation of the Spanning Tree Protocol Entity within an I-component in a Backbone Edge Bridge. The Customer Network Ports (CNP) and Virtual Instance Ports (VIP) are treated as separate Bridge Ports by the spanning tree protocol.

If the I-component has a single CNP and a single VIP supported by a point-to-point backbone service instance, then all frames (including Spanning Tree BPDUs) addressed to the Bridge Group address may be relayed between the two ports of the I-component without modification. Otherwise, the Spanning Tree Protocol Entity shall execute RSTP, as modified by the provisions of this subclause.

The RSTP enhancements specified ensure that connectivity for frames assigned to any given S-VLAN is loop free.

The parameters and functions of the RSTP protocol used on the VIPs get the same values and functionality as defined for Provider Edge Ports of a C-VLAN component as defined in 13.41. The Bridge Identifier Priority and system ID extension get the values specified in 13.41.5. These changes in the RSTP protocol ensure that no VIP is blocked due to the operation of the RSTP protocol and the I-component will never be elected as root.

NOTE—The effect of not blocking any VIP in the I-component (never set the port role alternate to a VIP) will not cause a loop since the I-component maps any given S-VID to at most one VIP.

13.43 L2 Gateway Ports

<<place-holder, to be consistent with the other state machines the general description of L2GP should move here, just leaving the L2GPRX machine where it is>>

