

# Technology Overview

Spanning Tree Protocol  
on Juniper Networks MX Series 3D Universal Edge  
Routers

Release

10.4



Published: 2010-10-08

Revision 1

Juniper Networks, Inc.  
1194 North Mathilda Avenue  
Sunnyvale, California 94089  
USA  
408-745-2000  
www.juniper.net

This product includes the Envoy SNMP Engine, developed by Epilogue Technology, an Integrated Systems Company. Copyright © 1986-1997, Epilogue Technology Corporation. All rights reserved. This program and its documentation were developed at private expense, and no part of them is in the public domain.

This product includes memory allocation software developed by Mark Moraes, copyright © 1988, 1989, 1993, University of Toronto.

This product includes FreeBSD software developed by the University of California, Berkeley, and its contributors. All of the documentation and software included in the 4.4BSD and 4.4BSD-Lite Releases is copyrighted by the Regents of the University of California. Copyright © 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994. The Regents of the University of California. All rights reserved.

GateD software copyright © 1995, the Regents of the University. All rights reserved. Gate Daemon was originated and developed through release 3.0 by Cornell University and its collaborators. Gated is based on Kirton's EGP, UC Berkeley's routing daemon (routed), and DCN's HELLO routing protocol. Development of Gated has been supported in part by the National Science Foundation. Portions of the GateD software copyright © 1988, Regents of the University of California. All rights reserved. Portions of the GateD software copyright © 1991, D. L. S. Associates.

This product includes software developed by Maker Communications, Inc., copyright © 1996, 1997, Maker Communications, Inc.

Juniper Networks, Junos, Steel-Belted Radius, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. The Juniper Networks Logo, the Junos logo, and JunosE are trademarks of Juniper Networks, Inc. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners.

Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

Products made or sold by Juniper Networks or components thereof might be covered by one or more of the following patents that are owned by or licensed to Juniper Networks: U.S. Patent Nos. 5,473,599, 5,905,725, 5,909,440, 6,192,051, 6,333,650, 6,359,479, 6,406,312, 6,429,706, 6,459,579, 6,493,347, 6,538,518, 6,538,899, 6,552,918, 6,567,902, 6,578,186, and 6,590,785.

*Technology Overview Spanning Tree Protocol*  
*on Juniper Networks MX Series 3D Universal Edge Routers*  
Release 10.4  
Copyright © 2010, Juniper Networks, Inc.  
All rights reserved. Printed in USA.

Writing: Sridhar Selvarangam, Kumaraguru Radhakrishnan  
Editing: Marilyn Kerr, Katie Smith, Roy Spencer  
Illustration: Dawn Spencer  
Cover Design: Edmonds Design

Revision History  
October 2010—R1 Junos 10.4

The information in this document is current as of the date listed in the revision history.

#### YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. The Junos OS has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.

## END USER LICENSE AGREEMENT

**READ THIS END USER LICENSE AGREEMENT ("AGREEMENT") BEFORE DOWNLOADING, INSTALLING, OR USING THE SOFTWARE.** BY DOWNLOADING, INSTALLING, OR USING THE SOFTWARE OR OTHERWISE EXPRESSING YOUR AGREEMENT TO THE TERMS CONTAINED HEREIN, YOU (AS CUSTOMER OR IF YOU ARE NOT THE CUSTOMER, AS A REPRESENTATIVE/AGENT AUTHORIZED TO BIND THE CUSTOMER) CONSENT TO BE BOUND BY THIS AGREEMENT. IF YOU DO NOT OR CANNOT AGREE TO THE TERMS CONTAINED HEREIN, THEN (A) DO NOT DOWNLOAD, INSTALL, OR USE THE SOFTWARE, AND (B) YOU MAY CONTACT JUNIPER NETWORKS REGARDING LICENSE TERMS.

1. **The Parties.** The parties to this Agreement are (i) Juniper Networks, Inc. (if the Customer's principal office is located in the Americas) or Juniper Networks (Cayman) Limited (if the Customer's principal office is located outside the Americas) (such applicable entity being referred to herein as "Juniper"), and (ii) the person or organization that originally purchased from Juniper or an authorized Juniper reseller the applicable license(s) for use of the Software ("Customer") (collectively, the "Parties").

2. **The Software.** In this Agreement, "Software" means the program modules and features of the Juniper or Juniper-supplied software, for which Customer has paid the applicable license or support fees to Juniper or an authorized Juniper reseller, or which was embedded by Juniper in equipment which Customer purchased from Juniper or an authorized Juniper reseller. "Software" also includes updates, upgrades and new releases of such software. "Embedded Software" means Software which Juniper has embedded in or loaded onto the Juniper equipment and any updates, upgrades, additions or replacements which are subsequently embedded in or loaded onto the equipment.

3. **License Grant.** Subject to payment of the applicable fees and the limitations and restrictions set forth herein, Juniper grants to Customer a non-exclusive and non-transferable license, without right to sublicense, to use the Software, in executable form only, subject to the following use restrictions:

- a. Customer shall use Embedded Software solely as embedded in, and for execution on, Juniper equipment originally purchased by Customer from Juniper or an authorized Juniper reseller.
- b. Customer shall use the Software on a single hardware chassis having a single processing unit, or as many chassis or processing units for which Customer has paid the applicable license fees; provided, however, with respect to the Steel-Belted Radius or Odyssey Access Client software only, Customer shall use such Software on a single computer containing a single physical random access memory space and containing any number of processors. Use of the Steel-Belted Radius or IMS AAA software on multiple computers or virtual machines (e.g., Solaris zones) requires multiple licenses, regardless of whether such computers or virtualizations are physically contained on a single chassis.
- c. Product purchase documents, paper or electronic user documentation, and/or the particular licenses purchased by Customer may specify limits to Customer's use of the Software. Such limits may restrict use to a maximum number of seats, registered endpoints, concurrent users, sessions, calls, connections, subscribers, clusters, nodes, realms, devices, links, ports or transactions, or require the purchase of separate licenses to use particular features, functionalities, services, applications, operations, or capabilities, or provide throughput, performance, configuration, bandwidth, interface, processing, temporal, or geographical limits. In addition, such limits may restrict the use of the Software to managing certain kinds of networks or require the Software to be used only in conjunction with other specific Software. Customer's use of the Software shall be subject to all such limitations and purchase of all applicable licenses.
- d. For any trial copy of the Software, Customer's right to use the Software expires 30 days after download, installation or use of the Software. Customer may operate the Software after the 30-day trial period only if Customer pays for a license to do so. Customer may not extend or create an additional trial period by re-installing the Software after the 30-day trial period.
- e. The Global Enterprise Edition of the Steel-Belted Radius software may be used by Customer only to manage access to Customer's enterprise network. Specifically, service provider customers are expressly prohibited from using the Global Enterprise Edition of the Steel-Belted Radius software to support any commercial network access services.

The foregoing license is not transferable or assignable by Customer. No license is granted herein to any user who did not originally purchase the applicable license(s) for the Software from Juniper or an authorized Juniper reseller.

4. **Use Prohibitions.** Notwithstanding the foregoing, the license provided herein does not permit the Customer to, and Customer agrees not to and shall not: (a) modify, unbundle, reverse engineer, or create derivative works based on the Software; (b) make unauthorized copies of the Software (except as necessary for backup purposes); (c) rent, sell, transfer, or grant any rights in and to any copy of the Software, in any form, to any third party; (d) remove any proprietary notices, labels, or marks on or in any copy of the Software or any product in which the Software is embedded; (e) distribute any copy of the Software to any third party, including as may be embedded in Juniper equipment sold in the secondhand market; (f) use any 'locked' or key-restricted feature, function, service, application, operation, or capability without first purchasing the applicable license(s) and obtaining a valid key from Juniper, even if such feature, function, service, application, operation, or capability is enabled without a key; (g) distribute any key for the Software provided by Juniper to any third party; (h) use the

Software in any manner that extends or is broader than the uses purchased by Customer from Juniper or an authorized Juniper reseller; (i) use Embedded Software on non-Juniper equipment; (j) use Embedded Software (or make it available for use) on Juniper equipment that the Customer did not originally purchase from Juniper or an authorized Juniper reseller; (k) disclose the results of testing or benchmarking of the Software to any third party without the prior written consent of Juniper; or (l) use the Software in any manner other than as expressly provided herein.

5. **Audit.** Customer shall maintain accurate records as necessary to verify compliance with this Agreement. Upon request by Juniper, Customer shall furnish such records to Juniper and certify its compliance with this Agreement.

6. **Confidentiality.** The Parties agree that aspects of the Software and associated documentation are the confidential property of Juniper. As such, Customer shall exercise all reasonable commercial efforts to maintain the Software and associated documentation in confidence, which at a minimum includes restricting access to the Software to Customer employees and contractors having a need to use the Software for Customer's internal business purposes.

7. **Ownership.** Juniper and Juniper's licensors, respectively, retain ownership of all right, title, and interest (including copyright) in and to the Software, associated documentation, and all copies of the Software. Nothing in this Agreement constitutes a transfer or conveyance of any right, title, or interest in the Software or associated documentation, or a sale of the Software, associated documentation, or copies of the Software.

8. **Warranty, Limitation of Liability, Disclaimer of Warranty.** The warranty applicable to the Software shall be as set forth in the warranty statement that accompanies the Software (the "Warranty Statement"). Nothing in this Agreement shall give rise to any obligation to support the Software. Support services may be purchased separately. Any such support shall be governed by a separate, written support services agreement. TO THE MAXIMUM EXTENT PERMITTED BY LAW, JUNIPER SHALL NOT BE LIABLE FOR ANY LOST PROFITS, LOSS OF DATA, OR COSTS OR PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES, OR FOR ANY SPECIAL, INDIRECT, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THIS AGREEMENT, THE SOFTWARE, OR ANY JUNIPER OR JUNIPER-SUPPLIED SOFTWARE. IN NO EVENT SHALL JUNIPER BE LIABLE FOR DAMAGES ARISING FROM UNAUTHORIZED OR IMPROPER USE OF ANY JUNIPER OR JUNIPER-SUPPLIED SOFTWARE. EXCEPT AS EXPRESSLY PROVIDED IN THE WARRANTY STATEMENT TO THE EXTENT PERMITTED BY LAW, JUNIPER DISCLAIMS ANY AND ALL WARRANTIES IN AND TO THE SOFTWARE (WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE), INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT DOES JUNIPER WARRANT THAT THE SOFTWARE, OR ANY EQUIPMENT OR NETWORK RUNNING THE SOFTWARE, WILL OPERATE WITHOUT ERROR OR INTERRUPTION, OR WILL BE FREE OF VULNERABILITY TO INTRUSION OR ATTACK. In no event shall Juniper's or its suppliers' or licensors' liability to Customer, whether in contract, tort (including negligence), breach of warranty, or otherwise, exceed the price paid by Customer for the Software that gave rise to the claim, or if the Software is embedded in another Juniper product, the price paid by Customer for such other product. Customer acknowledges and agrees that Juniper has set its prices and entered into this Agreement in reliance upon the disclaimers of warranty and the limitations of liability set forth herein, that the same reflect an allocation of risk between the Parties (including the risk that a contract remedy may fail of its essential purpose and cause consequential loss), and that the same form an essential basis of the bargain between the Parties.

9. **Termination.** Any breach of this Agreement or failure by Customer to pay any applicable fees due shall result in automatic termination of the license granted herein. Upon such termination, Customer shall destroy or return to Juniper all copies of the Software and related documentation in Customer's possession or control.

10. **Taxes.** All license fees payable under this agreement are exclusive of tax. Customer shall be responsible for paying Taxes arising from the purchase of the license, or importation or use of the Software. If applicable, valid exemption documentation for each taxing jurisdiction shall be provided to Juniper prior to invoicing, and Customer shall promptly notify Juniper if their exemption is revoked or modified. All payments made by Customer shall be net of any applicable withholding tax. Customer will provide reasonable assistance to Juniper in connection with such withholding taxes by promptly: providing Juniper with valid tax receipts and other required documentation showing Customer's payment of any withholding taxes; completing appropriate applications that would reduce the amount of withholding tax to be paid; and notifying and assisting Juniper in any audit or tax proceeding related to transactions hereunder. Customer shall comply with all applicable tax laws and regulations, and Customer will promptly pay or reimburse Juniper for all costs and damages related to any liability incurred by Juniper as a result of Customer's non-compliance or delay with its responsibilities herein. Customer's obligations under this Section shall survive termination or expiration of this Agreement.

11. **Export.** Customer agrees to comply with all applicable export laws and restrictions and regulations of any United States and any applicable foreign agency or authority, and not to export or re-export the Software or any direct product thereof in violation of any such restrictions, laws or regulations, or without all necessary approvals. Customer shall be liable for any such violations. The version of the Software supplied to Customer may contain encryption or other capabilities restricting Customer's ability to export the Software without an export license.

12. **Commercial Computer Software.** The Software is "commercial computer software" and is provided with restricted rights. Use, duplication, or disclosure by the United States government is subject to restrictions set forth in this Agreement and as provided in DFARS 227.7201 through 227.7202-4, FAR 12.212, FAR 27.405(b)(2), FAR 52.227-19, or FAR 52.227-14 (ALT III) as applicable.

13. **Interface Information.** To the extent required by applicable law, and at Customer's written request, Juniper shall provide Customer with the interface information needed to achieve interoperability between the Software and another independently created program, on payment of applicable fee, if any. Customer shall observe strict obligations of confidentiality with respect to such information and shall use such information in compliance with any applicable terms and conditions upon which Juniper makes such information available.

14. **Third Party Software.** Any licensor of Juniper whose software is embedded in the Software and any supplier of Juniper whose products or technology are embedded in (or services are accessed by) the Software shall be a third party beneficiary with respect to this Agreement, and such licensor or vendor shall have the right to enforce this Agreement in its own name as if it were Juniper. In addition, certain third party software may be provided with the Software and is subject to the accompanying license(s), if any, of its respective owner(s). To the extent portions of the Software are distributed under and subject to open source licenses obligating Juniper to make the source code for such portions publicly available (such as the GNU General Public License ("GPL") or the GNU Library General Public License ("LGPL")), Juniper will make such source code portions (including Juniper modifications, as appropriate) available upon request for a period of up to three years from the date of distribution. Such request can be made in writing to Juniper Networks, Inc., 1194 N. Mathilda Ave., Sunnyvale, CA 94089, ATTN: General Counsel. You may obtain a copy of the GPL at <http://www.gnu.org/licenses/gpl.html>, and a copy of the LGPL at <http://www.gnu.org/licenses/lgpl.html>.

15. **Miscellaneous.** This Agreement shall be governed by the laws of the State of California without reference to its conflicts of laws principles. The provisions of the U.N. Convention for the International Sale of Goods shall not apply to this Agreement. For any disputes arising under this Agreement, the Parties hereby consent to the personal and exclusive jurisdiction of, and venue in, the state and federal courts within Santa Clara County, California. This Agreement constitutes the entire and sole agreement between Juniper and the Customer with respect to the Software, and supersedes all prior and contemporaneous agreements relating to the Software, whether oral or written (including any inconsistent terms contained in a purchase order), except that the terms of a separate written agreement executed by an authorized Juniper representative and Customer shall govern to the extent such terms are inconsistent or conflict with terms contained herein. No modification to this Agreement nor any waiver of any rights hereunder shall be effective unless expressly assented to in writing by the party to be charged. If any portion of this Agreement is held invalid, the Parties agree that such invalidity shall not affect the validity of the remainder of this Agreement. This Agreement and associated documentation has been written in the English language, and the Parties agree that the English version will govern. (For Canada: Les parties aux présentes confirment leur volonté que cette convention de même que tous les documents y compris tout avis qui s'y rattache, soient rédigés en langue anglaise. (Translation: The parties confirm that this Agreement and all related documentation is and will be in the English language)).



# Table of Contents

Overview of Spanning Tree Protocol on Juniper Networks MX Series 3D Universal Edge Routers . . . . .	1
Spanning Tree Protocol Operation . . . . .	3
Key Concepts in Spanning Tree Protocols . . . . .	7
Port Roles in STP . . . . .	9
Decision Sequence for a Loop-Free STP Topology . . . . .	11
Spanning Tree Protocol States . . . . .	13
Rapid Spanning Tree Protocol Port States and Port Roles . . . . .	17
Multiple Spanning Tree Protocol . . . . .	21
VLAN Spanning Tree Protocol . . . . .	27
STP Scaling and Performance on Juniper Networks MX Series 3D Universal Edge Routers . . . . .	31
Restrictions and Cautions for Implementing STP . . . . .	33





## Overview of Spanning Tree Protocol on Juniper Networks MX Series 3D Universal Edge Routers

---

The Spanning Tree Protocol (STP) is defined in the Institute of Electrical and Electronics Engineers (IEEE) Standard 802.1D. STP examines the network topology and calculates a spanning-tree structure that encompasses all bridges in a given Layer 2 network domain. It can disable redundant paths by pruning links that are not part of the tree, leaving a unique single path from each source to any other destination in the network.

STP consists of distance vector protocols that use distance or hop count as the primary metric for determining the best forwarding path. It uses maximum-age parameters to help avoid loops that count to infinity.

This document describes the major Spanning Tree Protocol versions supported on Juniper Networks MX Series 3D Universal Edge Routers using Junos OS Release 8.4 or later, including:

- Spanning Tree Protocol (STP), specified in IEEE 802.1D 1998
- Rapid Spanning Tree Protocol (RSTP), specified in IEEE 802.1w; standardized in IEEE 802.1D 2004
- Multiple Spanning Tree Protocol (MSTP), specified in IEEE 802.1s; standardized in IEEE 802.1Q 2003
- Virtual LAN Spanning Tree Protocol (VSTP), Juniper Networks solution for using STP with multiple VLANs

Both Juniper Networks EX Series Ethernet Switches and MX Series routers run Junos OS and support a similar set of Layer 2 features with some variations. The EX Series switches support three standard versions: STP, RSTP, and MSTP. The MX Series routers add support for VSTP, which is compatible with Per-VLAN Spanning Tree Plus (PVST+) and Rapid-PVST+ protocols supported on other vendors' routers and switches.

### Understanding the Requirement for Spanning Tree Protocols

MX Series routers are used for deploying Layer 2 bridged networks. STPs are essential for these types of deployments, where they are used for allowing Layer 2 bridged networks to include spare, redundant links that provide automatic backup paths if an active link fails. However, whenever there are redundant paths between a source and a destination, there is the potential for forwarding loops. Forwarding loops must be avoided because they result in broadcast storms in the network. STPs are required to block redundant forwarding paths to prevent forwarding loops.

This document helps you decide which STP version can and should be used on your MX Series routers to ensure a loop-free topology.

#### Related Documentation

- Decision Sequence for a Loop-Free STP Topology on page 11
- Key Concepts in Spanning Tree Protocols on page 7
- Multiple Spanning Tree Protocol on page 21

- Port Roles in STP on page 9
- Rapid Spanning Tree Protocol Port States and Port Roles on page 17
- Restrictions and Cautions for Implementing STP on page 33
- Spanning Tree Protocol Operation on page 3
- Spanning Tree Protocol States on page 13
- STP Scaling and Performance on Juniper Networks MX Series 3D Universal Edge Routers on page 31
- VLAN Spanning Tree Protocol on page 27

## Spanning Tree Protocol Operation

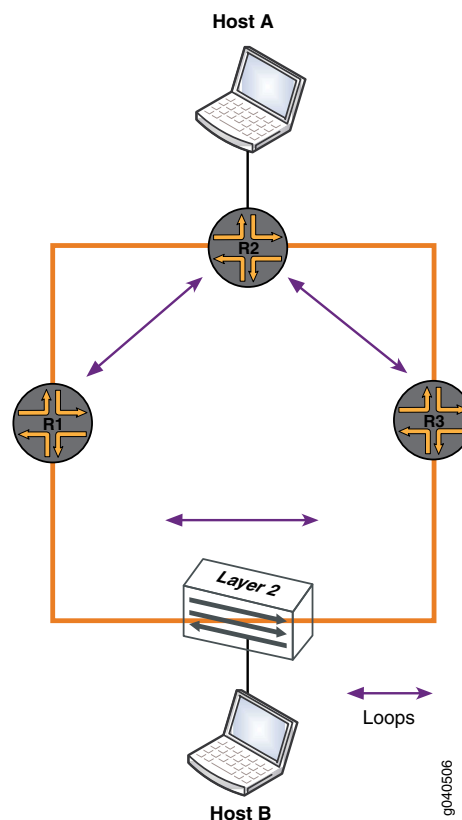
The Spanning Tree Protocol is used to create a loop-free topology in Layer 2 networks. It is a Layer 2 protocol that calculates the best path through a switched network that contains redundant paths.

It allows bridges to communicate with each other for the purpose of discovering physical loops in the network. When physical loops are found, the protocol specifies an algorithm that bridges can use to create a loop-free logical topology. Blocking loops reduces flooding in the network. The STP algorithm computes a tree structure of loop-free leaves and branches that spans the entire Layer 2 network.

Various Layer 2 STP control protocols take care of resolving the forwarding loops in a Layer 2 network. L2CPD is the Junos OS process responsible for all of the various STP versions in the Junos operating system.

Figure 1 on page 3 shows a typical Layer 2 network in which broadcast loops can occur.

**Figure 1: Flat Topology of Layer 2 Loops**



In this example, STP is not enabled. Host A sends a frame to Host B to the broadcast MAC address (ff-ff-ff-ff-ff-ff). Router R2 (operating as a switch) receives the frame and sends it out to Router R1 and Router R3. When Router R1 and Router R3 receive the frame, they forward the frame to the Layer 2 switch connected to Host B. The Layer 2 switch

then forwards the packet to the other router and to Host B. When Router R1 and Router R3 receive the frame, they forward the frame to Router R2. Router R2 forwards the frame to Host A, Router R1, and Router R2. Thus creating a forwarding loop.

Looping in Layer 2 Ethernet frame formats is handled differently than it is in IP headers. An IP header has a time-to-live (TTL) field that is set by the original host and is decremented at each router, allowing the router to prevent looped datagrams by discarding packets that reach TTL=0. This feature is not available for Layer 2 Ethernet frames.

Therefore, after a Layer 2 frame starts to loop in the network shown in Figure 1 on page 3, it continues looping forever until one of these events occurs:

- One of the bridges is turned off.
- A link is broken.
- A bridge is rebooted.

Bridging tables that are not configured with STP can be corrupted by the looping created by a broadcast storm. Looping can also be created by unicast traffic.

There are four Ethernet frame formats, each illustrated in Table 1, Table 2, Table 3, and Table 4:

**Table 1: Ethernet (Ethernet II) Frame Format**

Dst	Src	Type	Data
<-- 6 -->	<-- 6 -->	<-- 2 -->	<-- 46-1500 -->

**Table 2: IEEE 802.3 Frame Format**

Dst	Src	Length	Data
<-- 6 -->	<-- 6 -->	<-- 2 -->	<-- 46-1500 -->

**Table 3: IEEE 802.2 (IEEE 802.3 with IEEE 802.2 header) Frame Format**

Dst	Src	Length	DSAP	SSAP	Control	Data
			<-- 1 -->	<-- 1 -->	<-- 1 -->	<-- 43-1497 -->

**Table 4: SNAP (IEEE 802.3 with IEEE 802.2 and SNAP headers) Frame Format**

Dst	Src	Length	OxAA	OxAA	Ox03	Org Code	Type	Data
						<-- 3 -->	<-- 2 -->	<-- 38-1492 -->

**Related Documentation**

- Decision Sequence for a Loop-Free STP Topology on page 11
- Key Concepts in Spanning Tree Protocols on page 7

- Multiple Spanning Tree Protocol on page 21
- Overview of Spanning Tree Protocol on Juniper Networks MX Series 3D Universal Edge Routers on page 1
- Port Roles in STP on page 9
- Rapid Spanning Tree Protocol Port States and Port Roles on page 17
- Restrictions and Cautions for Implementing STP on page 33
- Spanning Tree Protocol States on page 13
- STP Scaling and Performance on Juniper Networks MX Series 3D Universal Edge Routers on page 31
- VLAN Spanning Tree Protocol on page 27



## Key Concepts in Spanning Tree Protocols

---

This section outlines key concepts important when using STPs.

### Bridge Protocol Data Units

Bridges use special frames known as bridge protocol data units (BPDUs) to transmit STP information between bridges. BPDUs carry information about bridge IDs and root path costs. STP uses BPDU packets to exchange information with other bridges. STP uses the information provided by the BPDUs to elect a root bridge, identify root ports for each bridge, identify designated ports for each physical LAN segment, and prune specific redundant links to create a loop-free tree topology.

There are two types of BPDUs:

- *Configuration BPDUs*—Under normal circumstances, once the root bridge is selected, configuration BPDUs are originated by the root bridge and flow outward along active paths radiating away from the root bridge.
- *Topology change notification BPDUs*—If there is any change in the topology, for example, a link is disconnected or a bridge fails, the bridge sends topology change notification (TCN) BPDUs to advertise the change. The TCN BPDUs flow upstream toward the root bridge to alert it that the active topology has changed.

### Root Bridge

The STP root bridge is the base of the spanning tree topology, much like roots are the base of a tree. All redundant paths to the root bridge within the spanning-tree network are disabled by putting them into a blocked mode. The root bridge is chosen by all of the bridges, based on the results of the BPDU exchange process.

### Bridge ID

A bridge ID (BID) is a single, 8-byte field that identifies a bridge through two subfields: the bridge priority and the MAC address of the bridge.

### Path Cost

Bridges use the concept of path cost to evaluate how close they are to other bridges. The path cost depends on the bandwidth of the link. STP uses algorithms to evaluate least-cost paths as most attractive.

### Port Roles

Bridges determine the root bridge and then compute the port roles based on the BPDU information, for any spanning-tree version.

#### Related Documentation

- Decision Sequence for a Loop-Free STP Topology on page 11
- Multiple Spanning Tree Protocol on page 21
- Overview of Spanning Tree Protocol on Juniper Networks MX Series 3D Universal Edge Routers on page 1

- Port Roles in STP on page 9
- Rapid Spanning Tree Protocol Port States and Port Roles on page 17
- Restrictions and Cautions for Implementing STP on page 33
- Spanning Tree Protocol Operation on page 3
- Spanning Tree Protocol States on page 13
- STP Scaling and Performance on Juniper Networks MX Series 3D Universal Edge Routers on page 31
- VLAN Spanning Tree Protocol on page 27



## Port Roles in STP

---

This section describes the function of the port roles in STP. The use and implementation of port roles differs slightly between different versions of STP.

Port roles include:

- *Designated port*—Responsible for forwarding the traffic coming from the spanning-tree root bridge onto this segment.
- *Backup port*—A backup for the designated port.
- *Root port*—The root port of a bridge is the port closest to the root bridge. It is responsible for carrying the traffic from this segment toward the root bridge, using the shortest path to reach the root.
- *Alternate port*—A backup for the root port.
- *Master port (MSTP only)*—The best possible path to reach the common internal spanning-tree (CIST) root bridge from any MSTP region.

### Related Documentation

- Decision Sequence for a Loop-Free STP Topology on page 11
- Key Concepts in Spanning Tree Protocols on page 7
- Multiple Spanning Tree Protocol on page 21
- Overview of Spanning Tree Protocol on Juniper Networks MX Series 3D Universal Edge Routers on page 1
- Rapid Spanning Tree Protocol Port States and Port Roles on page 17
- Restrictions and Cautions for Implementing STP on page 33
- Spanning Tree Protocol Operation on page 3
- Spanning Tree Protocol States on page 13
- STP Scaling and Performance on Juniper Networks MX Series 3D Universal Edge Routers on page 31
- VLAN Spanning Tree Protocol on page 27



## Decision Sequence for a Loop-Free STP Topology

---

STP always uses the same four-step decision sequence when creating a loop-free logical topology. Evaluations for computing the segments and branches of the spanning tree are performed in this order:

1. Lowest root identifier (RID).
2. Lowest path cost to root bridge.
3. Lowest sender bridge identifier (BID).
4. Lowest port ID.

Each bridge uses the four-step decision sequence to save a copy of the best BPDU for each port. When making this evaluation, it considers all of the BPDUs received on the port as well as the BPDU that is sent from that port. As each BPDU arrives, it is checked against the four-step sequence to see if it is lower in value, and thus more attractive, than the existing BPDU saved for that port.

When a bridge first becomes active, all of its ports send BPDUs every two seconds, the default time value. However, if a port hears a BPDU from another bridge that is more attractive than the BPDU it has been sending, the local port stops sending BPDUs. If the more attractive BPDU stops arriving from a neighbor for a period of time (default value is 20 seconds), the local port resumes sending BPDUs.

If a new or a locally-generated BPDU is more attractive than the existing BPDU, the existing value is replaced. Bridges continue sending configuration BPDUs until a more attractive BPDU is received.

### Steps of Initial Convergence

This section describes the algorithm that STP uses for initial convergence on a loop-free logical topology. Initial convergence consists of three steps:

1. Elect one root bridge.
2. Elect root ports.
3. Elect designated ports.

When the network first starts, all of the bridges are announcing a chaotic mix of BPDU information. The bridges immediately begin applying the four-step decision sequence in order to start computing a single spanning-tree for the entire network.

First, a single root bridge is elected for the entire domain. Next, all of the remaining bridges calculate a set of root ports and designated ports to build a loop-free topology. Every non-root bridge selects one root port. Specifically, bridges track the root path cost, which is the cumulative cost of all links to the root bridge. All of the ports in a root bridge are considered to be the root port, with a few exceptions. One exception is for the port that is connected back-to-back to the same root bridge.

Finally, the designated ports are selected. Each segment in a bridged network selects one designated port to be the single bridge port that exchanges traffic to and from that segment and the root bridge.

**Related  
Documentation**

- Key Concepts in Spanning Tree Protocols on page 7
- Multiple Spanning Tree Protocol on page 21
- Overview of Spanning Tree Protocol on Juniper Networks MX Series 3D Universal Edge Routers on page 1
- Port Roles in STP on page 9
- Rapid Spanning Tree Protocol Port States and Port Roles on page 17
- Restrictions and Cautions for Implementing STP on page 33
- Spanning Tree Protocol Operation on page 3
- Spanning Tree Protocol States on page 13
- STP Scaling and Performance on Juniper Networks MX Series 3D Universal Edge Routers on page 31
- VLAN Spanning Tree Protocol on page 27

## Spanning Tree Protocol States

This section describes essential features and operation of the original STP as defined in IEEE 802.1d.

After the bridges have classified the ports as root, designated, or non-designated, creating a loop-free topology is straightforward: root ports and designated ports forward traffic; non-designated ports block traffic.

### Spanning Tree Protocol States

When a bridge is first attached to a network segment, and before it can start forwarding data, it goes through a series of states while it processes BPDUs and learns the topology of the network. There are five states in STP, described in Table 5 on page 13.

Table 5: STP States

State	Purpose
Listening	Processing BPDUs and building active topology
Learning	Building bridging tables; no forwarding of data
Forwarding	Sending and receiving data; normal operation
Blocking	A port that would cause a loop if it were sending data, so it is only receiving BPDUs, until a topology change removes the possibility of a loop
Disabled	A port that is manually isolated from the network

### Notes on the Listening State

Initially, every bridge acts as if it is a root bridge, and enters the listening state to determine the active topology. An absence of BPDUs for a certain period of time can also cause the bridge to transition into the listening state. In the listening state, no user data is being passed; however, the port is sending and receiving BPDUs in an effort to determine the active topology. The three initial convergence steps take place in the listening state.

During initial convergence, any ports that are not elected as a designated or root port go into the blocking state.

After the default length of 15 seconds in the listening state, any ports that remain as designated or root ports progress into the learning state.

### Notes on the Learning State

The learning state is a 15-second interval during which the bridge does not pass user data frames while the bridge is building its bridging table. As the bridge receives frames, it places the source MAC address and port of each frame into the bridging table. The learning state reduces the amount of flooding required when data forwarding begins.

At the end of the learning state, if a port is still a designated port or a root port, it transitions into the forwarding state. The port starts sending and receiving user data frames only after it enters the forwarding state.

The bridge spends a default of 15 seconds in each of the listening and learning states.

### STP Timers

The Spanning Tree Protocol is controlled by three timers, as shown in Table 6.

**Table 6: STP Timers**

Timer	Primary Purpose	Default Time in Seconds	Description
Forward Delay	Duration of listening and learning states	15	Forward delay is the time that the bridge spends in the listening and learning states. It is a single value that controls both states. The default value of 15 seconds was originally derived by assuming a maximum network size of seven bridge/hub hops, a maximum of three lost BPDUs, and a Hello Time interval of two seconds.
Hello Time	Interval between sending of configuration BPDUs by the root bridge	2	The Hello Time controls the length of time between sending configuration BPDUs. The default value of 2 seconds only applies to configuration BPDUs, because they are generated at the root bridge. Other bridges propagate BPDUs from the root bridge as they are received. If BPDUs stop arriving for 2 through 20 seconds because of network disturbance, non-root bridges stop sending periodic BPDUs during this time.
Max Age	Length of time a BPDU is stored	20	Max Age is the length of time that a bridge stores a BPDU before discarding it.

### Processing of STP Topology Change Notification BPDUs

A bridge originates a topology change notification (TCN) BPDU in either of two conditions:

- When it transitions a port into the forwarding state and it has at least one designated port.
- When it transitions a port from either the forwarding or learning states to the blocking state.

Each of these two conditions changes the active topology, requiring notification to be sent to the root bridge. Assuming that the current bridge is not the root bridge, the current bridge begins the notification process by sending a TCN BPDU from its root port. It continues sending the TCN BPDU every Hello Time interval until the TCN message is acknowledged.

The upstream bridge receives the TCN BPDU. Although several bridges might hear the TCN BPDU, only the designated port accepts and processes it.

The upstream bridge sets the topology change acknowledgement (TCA) flag in the next configuration BPDU that it sends downstream from the designated port. This

acknowledges that the TCN BPDU was received and causes the originating bridge to stop generating TCN BPDUs.

The upstream bridge propagates the TCN BPDU out its root port, resulting in the TCN BPDU being one hop closer to the root bridge. The upstream bridges continue using this process until the root bridge receives the TCN BPDU.

The root bridge then sets the TCA flag to acknowledge the TCN BPDU sent by the previous bridge and also sets the TCA flag in the next configuration BPDU that it sends out.

The root bridge continues to set the TCA flag in all configuration BPDUs that it sends out for a total time equal to Forwarding Delay + Max Age seconds = 35 seconds (default). This flag instructs all the bridges to purge the MAC addresses in the bridge tables and start learning again as soon as the new loop-free topology is available.

### Time to Convergence

STP needs twice the length of the forwarding delay (15 seconds) to transition a port from the blocking state to the forwarding state, for a total of approximately 30 seconds to convergence. Typically, STP is configured as RSTP. If the original STP mode is needed, include the **force-version stp** statement at the **[edit protocols rstp]** hierarchy level.

### Related Documentation

- Decision Sequence for a Loop-Free STP Topology on page 11
- Key Concepts in Spanning Tree Protocols on page 7
- Multiple Spanning Tree Protocol on page 21
- Overview of Spanning Tree Protocol on Juniper Networks MX Series 3D Universal Edge Routers on page 1
- Port Roles in STP on page 9
- Rapid Spanning Tree Protocol Port States and Port Roles on page 17
- Restrictions and Cautions for Implementing STP on page 33
- Spanning Tree Protocol Operation on page 3
- STP Scaling and Performance on Juniper Networks MX Series 3D Universal Edge Routers on page 31
- VLAN Spanning Tree Protocol on page 27





## Rapid Spanning Tree Protocol Port States and Port Roles

The original Spanning Tree Protocol is defined in the IEEE 802.1D 1998 specification. A newer version called Rapid Spanning Tree Protocol (RSTP) was originally defined in the IEEE 802.1w draft specification and later incorporated into the IEEE 802.1D-2004 specification.

RSTP provides faster reconvergence time than the original STP by identifying certain links as point-to-point and by using protocol handshake messages rather than fixed timeouts. When a point-to-point link fails, the alternate link can transition to the forwarding state without waiting for any protocol timers to expire. Consequently, RSTP convergence is approximately 50 milliseconds for point-to-point links.

Port operation is similar between STP and RSTP. In both, the state of the port is variable, and determines if the port blocks or forwards traffic. Additionally, the role a port plays in the active topology varies, if it is calculated to be a root port, a designated port, and so on.

For example, in STP there are no operational differences between a port in the blocking state and a port in the listening state. Both port states discard frames and do not learn MAC addresses. The real difference is in the role the spanning tree assigns to the port. You can assume that a listening port is either a designated port or a root port, and is in the process of transitioning to the forwarding state. Once in the forwarding state, there is no way to infer from the port state whether the port is root or designated.

This is a weakness of the state-based terminology of STP. To address this issue, RSTP decouples the role and the state of a port.

Consequently, there are only three port states in RSTP that correspond to the operational states of STP. The disabled, blocking, and listening states of IEEE 802.1D STP are merged into a unique IEEE 802.1w RSTP discarding state, as shown in Table 7 on page 17.

**Table 7: Port States in STP and RSTP**

STP (IEEE 802.1D)	RSTP (IEEE 802.1w)	Port Included in Active Topology?	Port is Learning MAC Addresses?
Disabled	Discarding	No	No
Blocking	Discarding	No	No
Listening	Discarding	Yes	No
Learning	Learning	Yes	Yes
Forwarding	Forwarding	Yes	Yes

### Port Roles in RSTP

In RSTP, the port role is a variable assigned to a given port. The root port and designated port roles remain, and the blocking port role is replaced with the alternate and backup port roles. The Spanning Tree Algorithm (STA) determines the role of a port based on BPDUs.

#### Root Port and Root Bridge

The port closest to the root bridge in terms of least path cost (based on BPDU) is determined to be the root port. The STA elects a single root bridge in the entire bridged network of each VLAN. The root bridge sends BPDUs that have a lower bridge priority than the BPDUs that any other bridges send. The root bridge is the only bridge in the network that does not have a root port. All other bridges receive BPDUs on at least one port.

#### Designated Port

The designated port is the port that can send the best BPDU on the segment to which it is connected. The IEEE 802.1D bridges link different Ethernet segments to create a bridged domain. On a given segment, there can be only one path toward the root bridge. If there are two paths, there is a bridging loop in the network. All bridges connected to a given segment listen to the BPDUs of all bridges on that segment and agree on the bridge that sends the best BPDU as the designated bridge for the segment. The port on that bridge becomes the designated port for that segment.

#### Alternate and Backup Port

The alternate and backup port roles correspond to the blocking state of STP. A blocked port is defined as not being the designated or root port. A blocked port receives a more useful BPDU than the one it sends out on its segment. A port must receive BPDUs in order to stay blocked. For this purpose, RSTP introduces these two port roles:

- An alternate port receives more useful BPDUs from another bridge and is a blocked port.
- A backup port provides redundant connectivity to the same segment and cannot guarantee alternate connectivity to the root bridge.

When a port is selected by the STA to become a designated port in STP, the port still waits for two times the forward delay seconds ( $2 \times 15$  default) before it transitions to the forwarding state. In RSTP, this condition corresponds to a port with a designated role but with a blocking state that directly transitions to the forwarding state. It skips the listening and learning states, helping it converge faster than original STP.

RSTP has a backward compatibility mode in which it can fall back to STP operation on links.

The following is an RSTP configuration example.



NOTE: The `extended-system-id` statement is used to specify different bridge IDs for different STP or RSTP routing instances. In MSTP, each routing instance has its own bridge ID, so the `extended-system-id` statement is not used.



NOTE: To force RSTP to run in STP mode, include the `force-version stp` statement at the `[edit protocols rstp]` hierarchy level. Convergence time will then be the same as in original STP.

```
routing-instances {
  customer1 {
    instance-type virtual-switch;
    bridge-domains {
      ...
    }
  }
}
protocols {
  rstp {
    bpdu-block-on-edge;
    bpdu-destination-mac-address provider-bridge-group;
    bridge-priority priority;
    extended-system-id;
    force-version stp;
    forward-delay seconds;
    hello-time seconds;
    max-age seconds;
    interface interface-name {
      bpdu-timeout-action (alarm | block);
      cost cost;
      edge;
      mode (p2p | shared);
      no-root-port;
      priority interface-priority;
    }
    priority-hold-time seconds;
    traceoptions {
      file filename <files number> <size size> <world-readable | no-world-readable>;
      flag flag <flag-modifier> <disable>;
    }
  }
}
```

#### Related Documentation

- Decision Sequence for a Loop-Free STP Topology on page 11
- Key Concepts in Spanning Tree Protocols on page 7
- Multiple Spanning Tree Protocol on page 21
- Overview of Spanning Tree Protocol on Juniper Networks MX Series 3D Universal Edge Routers on page 1

- Port Roles in STP on page 9
- Restrictions and Cautions for Implementing STP on page 33
- Spanning Tree Protocol Operation on page 3
- Spanning Tree Protocol States on page 13
- STP Scaling and Performance on Juniper Networks MX Series 3D Universal Edge Routers on page 31
- VLAN Spanning Tree Protocol on page 27

## Multiple Spanning Tree Protocol

---

Multiple Spanning Tree Protocol (MSTP) was first specified in IEEE 802.1s and is standardized in IEEE 802.1Q. MSTP enables multiple VLANs to be mapped to the same spanning-tree instance, reducing the number of spanning-tree instances needed to support a large number of VLANs. MSTP provides multiple forwarding paths for data traffic and enables load balancing. It improves the fault tolerance of the network because a failure in one instance, or forwarding path, does not affect other instances.

It may be necessary to have different topologies for different VLANs, for load-sharing or other purposes. MSTP enables the grouping of multiple VLANs with the same topology requirements into one MST instance (MSTI). Instances are not supported in STP or RSTP, so those two versions have the same spanning-tree in common for all of the VLANs.

In MSTP, all of the interconnected bridges that have the same MSTP configuration comprise an MST region. An MSTP configuration consists of the configuration name, the configuration revision, and the mapping of VLANs to MSTIs.

There is a common spanning-tree (CST) instance that spans all regions and allows different regions to communicate between themselves. The CST is also used for traffic within the region for any VLANs not covered by an MSTI.

MSTP has a backward compatibility mode in which it can fall back to STP or RSTP operation on links with bridges that support only STP or RSTP.

The maximum number of MSTP instances supported in Juniper Networks MX Series 3D Universal Edge Routers is 64.

### Multiple Spanning-Tree Regions

MSTP provides the capability to logically divide a Layer 2 network into regions. Every region has a unique identifier and can contain multiple instances of spanning trees. All regions are bound together using a Common Instance Spanning Tree (CIST), which is responsible for creating a loop-free topology across regions, whereas the Multiple Spanning Tree Instance (MSTI) controls topology inside regions.

In order for bridges to participate in multiple spanning-tree (MST) instances, they must be consistently configured with the same MST configuration, the same revision level, and the same VLAN-to-instance assignment mapping information.

The inheritance model is based on the hierarchy level in which the priority costs are defined. If the priority costs are defined within an interface in MSTI hierarchy level, that configuration takes precedence. Otherwise, the values are derived from the values specified under the interface in the MSTP hierarchy level.

### Internal Spanning Tree, Common Internal Spanning Tree, and Common Spanning Tree

Within each MST region, the MSTP maintains multiple spanning-tree instances. All MST instances are numbered 0 through 4094. Instance 0 is a special instance for the region, known as the internal spanning tree (IST).

The IST is the only spanning-tree instance that sends and receives BPDUs. All of the other spanning-tree instance information is in M-records, which are encapsulated within MSTP BPDUs. Because the MSTP BPDU carries information for all instances, the number of BPDUs that need to be processed to support multiple spanning-tree instances is significantly reduced.

All MST instances within the same region share the same protocol timers, but each MST instance has its own topology parameters, such as root bridge ID, root path cost, and so on. By default, all VLANs are assigned to the IST.

An MST instance is local to its region; for example, MST instance 1 in region A is independent of MST instance 1 in region B, even if regions A and B are interconnected.

ISTs in different regions are interconnected through a common spanning tree (CST). The collection of the ISTs in each MST region, and the common spanning tree that interconnects the MST regions and single spanning trees are called the common and internal spanning tree (CIST).

The spanning tree computed in a region appears as a subtree in the CST that encompasses the entire bridged domain. The CIST is formed by the spanning-tree algorithm running among bridges that support the IEEE 802.1w, IEEE 802.1s, and IEEE 802.1D standards. The CIST inside an MST region is the same as the CST outside a region.

### **Operations Within an MST Region**

The IST connects all of the MSTP bridges in a region. When the IST converges, its root becomes the CIST regional root. The regional root is the bridge within the region that has the lowest bridge ID and path cost to the CIST root. The CIST regional root is also the CIST root if there is only one region in the network. If the CIST root is outside of the region, one of the MSTP bridges at the boundary of the region is selected as the CIST regional root.

When an MSTP bridge initializes, it sends BPDUs announcing itself as the CIST root and the CIST regional root, with each of the path costs to the CIST root and to the CIST regional root set to zero. The bridge also initializes all of its MST instances and announces itself as the root for all of them. If the bridge receives superior MST root information from another source, such as a lower bridge ID or lower path cost than it currently has stored for the port, it relinquishes its claim as the CIST regional root.

During initialization, a region might have many subregions, each with its own CIST regional root. As bridges receive superior IST information, they leave their old subregions and join the new subregion that contains the true CIST regional root. In this manner, all subregions shrink, except for the one containing the true CIST regional root.

All bridges in the MST region must agree on the same CIST regional root to operate correctly. Therefore, any two bridges in the region only synchronize their port roles for an MST instance if they converge to a common CIST regional root.

Bridge priority can be separately specified in the configuration for each MSTI.

Multiple bridge domains can correspond to the same MSTI. A maximum of 64 MSTIs can be configured. The logical ports (e.g. ge-0/0/0.1) in a bridge domain can belong to a range of VLANs; therefore, an MSTI can map to VLAN ranges.

Each MSTI only spans the VLANs of those physical interfaces specified under the interface that are within the STP configuration.

### Operations Between MST Regions

If there are multiple regions or legacy STP bridges within the network, MSTP establishes and maintains the CST, which includes all MST regions and all legacy STP bridges in the network. The MST instances combine with the IST at the boundary of the region to become the CST.

The IST connects all of the MSTP bridges in the region. It appears as a subtree in the CIST that encompasses the entire bridged domain. The root of the subtree is the CIST regional root. The MST region appears as a virtual switch to adjacent STP bridges and MST regions.

The following is an MSTP configuration example. In the example, MSTI 0 is configured as the CIST:

```
routing-instances {
  instance1 {
    instance-type virtual-switch;
    bridge-domains {
      bd-vlan-10 {
        domain-type bridge;
        routing-interface irb.0;
        interface ge-3/0/0.0;
        interface ge-3/0/1.0;
      }
      bd-vlan-11 {
        domain-type bridge;
        routing-interface irb.1;
        interface ge-4/0/0.0;
        interface ge-4/0/1.0;
      }
      bd-vlan-12 {
        domain-type bridge;
        routing-interface irb.2;
        interface ge-5/0/0.0;
        interface ge-5/0/1.0;
      }
      bd-vlan-13 {
        domain-type bridge;
        routing-interface irb.3;
        interface ge-6/0/0.0;
        interface ge-6/0/1.0;
      }
      bd-vlan-14 {
        domain-type bridge;
        routing-interface irb.4;
        interface ge-7/0/0.0;
        interface ge-7/0/1.0;
      }
    }
  }
}
```

```
    }
  }
}
protocols {
  mstp {
    bpdu-block-on-edge;
    bridge-priority priority;
    configuration-name configuration-name;
    disable;
    forward-delay seconds;
    hello-time seconds;
    max-age seconds;
    max-hops hops;
    priority-hold-time seconds;
    revision-level revision-level;
    interface interface-name {
      bpdu-timeout-action (alarm | block);
      cost cost;
      edge;
      mode (p2p | shared);
      no-root-port;
      priority interface-priority;
    }
    msti msti-id {
      bridge-priority priority;
      vlan vlan-id;
      interface interface-name {
        cost cost;
        edge;
        priority interface-priority;
      }
    }
  }
  traceoptions {
    file filename <files number> <size size> <world-readable | no-world-readable>;
    flag flag <flag-modifier> <disable>;
  }
}
```

**Related  
Documentation**

- Decision Sequence for a Loop-Free STP Topology on page 11
- Key Concepts in Spanning Tree Protocols on page 7
- Overview of Spanning Tree Protocol on Juniper Networks MX Series 3D Universal Edge Routers on page 1
- Port Roles in STP on page 9
- Rapid Spanning Tree Protocol Port States and Port Roles on page 17
- Restrictions and Cautions for Implementing STP on page 33
- Spanning Tree Protocol Operation on page 3
- Spanning Tree Protocol States on page 13
- STP Scaling and Performance on Juniper Networks MX Series 3D Universal Edge Routers on page 31



- [VLAN Spanning Tree Protocol on page 27](#)



## VLAN Spanning Tree Protocol

---

VLAN Spanning Tree Protocol (VSTP) is a proprietary protocol developed by Juniper Networks to allow each VLAN to have a completely independent STP configuration, providing STP control on a per-VLAN basis. For example, each VLAN can have its root bridge located in a different place. Cost values and priority values can be tuned on a per-VLAN basis. Per-VLAN control allows the network designer total flexibility when it comes to optimizing data flows within each VLAN. It also makes spanning-tree load-balancing possible. However, VSTP also creates a lot of chaotic traffic in the network as the BPDUs are exchanged for every VLAN.

### Interoperability with Other Solutions

VSTP on Juniper Networks routers has some interoperability with other proprietary per-VLAN vendor solutions; specifically, VSTP can interoperate with per-VLAN spanning tree (PVST) and Rapid-PVST on IEEE 802.1Q non-proprietary trunks.

In PVST, the IEEE 802.1Q BPDUs are sent untagged on the common spanning-tree VLAN 1 for interoperability with other vendors. The CST BPDUs are sent to the IEEE standard bridge group: MAC Address 01-80-c2-00-00-00, DSAP 42, SSAP 42.

PVST BPDUs are tagged and sent to MAC address 01-00-0c-cc-cc-cd (SNAP HDLC protocol type 0x010b) for each VLAN on a trunk. PVST per-VLAN BPDUs are tunneled by pure IEEE 802.1Q bridges.

### VLAN Spanning Tree Protocol Operation

In a Layer 2 bridging network with a single spanning-tree instance (SSTP), a single instance of STP or RSTP runs on all VLANs in a single bridged LAN environment.

There are certain network deployment scenarios that require different VLAN groups to be administered for trunk load-sharing or other purposes. MSTP can be used to achieve trunk load-balancing.

In Junos OS revision 9.0 and later, a single spanning tree running on each configured VLAN is supported to ensure that each VLAN has a loop-free data path through the Layer 2 network. VSTP and Rapid VSTP (RVSTP) can interoperate in these scenarios with other proprietary solutions such as PVST and Rapid PVST on non-proprietary trunks.

Virtual bridges can be defined within a routing instance by configuring the routing instance type as **virtual-switch**. Each virtual switch can run one instance of standard STP or RSTP common spanning-tree (CST) and zero or more STP or RSTP instances for each enabled VLAN for VSTP. BPDUs of the CST are sent without any IEEE 802.1Q tags (untagged frame) on a port. BPDUs for the VSTP are sent tagged with corresponding VLAN IDs.

Each instance of VSTP or RVSTP on a VLAN elects a single root bridge. The root bridge distributes the STP information associated with that VLAN to all other bridges in the network. Because different root bridges can be provisioned for each VLAN in the network, this feature enhancement can be used by the network administrator for load balancing.

To enable VSTP or RVSTP for a specific VLAN ID, the user must define a bridge domain or VPLS routing instance with a VLAN ID, and all of the logical interfaces assigned to each of them should have the same VLAN ID.

For each VSTP or RVSTP, you can separately specify **bridge-priority**, **max-age**, **hello-time**, **forward-delay**, **port-priority**, **port-cost**, **port-mode** and **port-edge**. Each of these parameters are the same as in standard STP or RSTP.

For interfaces configured at the global level, VSTP is enabled for all configured VLANs. If an interface is configured at both the global and the VLAN level, the configuration at the VLAN level overrides the global configuration.

Because VSTP is enabled per VLAN, a physical interface can be part of more than one VSTP instance under different virtual bridges, if each of those VSTP instances have different VLANs.

The Juniper Networks default implementation of STP is RSTP and the default operation of VSTP is RVSTP.

### **VSTP Configuration Constraints on Logical Interfaces**

VSTP has some restrictions. It cannot be configured on logical interfaces that have:

- VLAN translations
- The VLAN range statement included
- Dual VLAN tags

### **VSTP Interoperability with Other Spanning-Tree Configurations**

Virtual switches that have VLAN spanning trees enabled can still run standard STP and RSTP on the ports and can interoperate with standard IEEE 802.1Q bridges by sending untagged BPDUs to the IEEE standard bridge group (MAC address 01-80-c2-00-00-00).

VSTP does not directly interoperate with STP, RSTP, and MSTP. Instead, VSTP BPDUs are transparently tunneled by standard STP, RSTP, and MSTP bridges. To interoperate with standard IEEE 802.1Q bridges, RSTP must also be enabled on any ports where VSTP is enabled.

When interoperating with other vendor VLAN STP solutions, VSTP BPDUs that are tagged with another vendor's multicast MAC address (for example, 01-00-0c-cc-cc-cd) are tunneled across the IEEE 802.1Q CST. This allows VSTP spanning tree information to be maintained by Juniper Networks devices even if they are separated by a cloud of standard IEEE 802.1Q network devices from other vendors.

The following is a VSTP configuration example:



**NOTE:** To force VSTP to run in STP mode, include the `force-version stp` statement at the `[edit protocols vstp]` hierarchy level.

```
protocols {
  vstp {
    bpdu-block-on-edge;
    force-version stp;
    interface interface-name {
      bpdu-timeout-action (alarm | block);
      cost cost;
      edge;
      mode (p2p | shared);
      no-root-port;
      priority interface-priority;
    }
    priority-hold-time seconds;
    vlan vlan-id {
      bridge-priority priority;
      forward-delay seconds;
      hello-time seconds;
      max-age seconds;
      interface interface-name {
        bpdu-timeout-action (alarm | block);
        cost cost;
        edge;
        mode (p2p | shared);
        no-root-port;
        priority interface-priority;
      }
    }
    traceoptions {
      file filename <files number> <size size> <world-readable | no-world-readable>;
      flag flag <flag-modifier> <disable>;
    }
  }
}
```

#### Related Documentation

- Decision Sequence for a Loop-Free STP Topology on page 11
- Key Concepts in Spanning Tree Protocols on page 7
- Multiple Spanning Tree Protocol on page 21
- Overview of Spanning Tree Protocol on Juniper Networks MX Series 3D Universal Edge Routers on page 1
- Port Roles in STP on page 9
- Rapid Spanning Tree Protocol Port States and Port Roles on page 17
- Restrictions and Cautions for Implementing STP on page 33
- Spanning Tree Protocol Operation on page 3
- Spanning Tree Protocol States on page 13

- STP Scaling and Performance on Juniper Networks MX Series 3D Universal Edge Routers on page 31

## STP Scaling and Performance on Juniper Networks MX Series 3D Universal Edge Routers

These are the critical scaling and performance parameters for different STP versions on Juniper Networks MX Series 3D Universal Edge Routers:

- Up to 64 MSTI instances
- Up to 256 routing instances
- Maximum 480 ports
- RSTP switchover time of < 50 ms in point-to-point links in case of link failures
- RSTP switchover time of < 50 ms in case of root bridge failover
- STP switchover time of two times forwarding delay for link failures
- 8,000 STP indices in hardware; maximum number of port and MSTI pairs is 8,000
- Maximum number of MSTP routing instances (including default) is 31

### MAC Address Block

The bridge reserves a block of MAC addresses to be used as bridge MAC addresses for different STP, RSTP, and MSTP routing instances. Currently, 32 MAC addresses are reserved.

These rules apply to allocating MAC addresses to protocols running under different routing instances:

- All STP and RSTP routing instances share the same bridge MAC address.
- All MSTP routing instances get a unique bridge MAC address.
- All routing instances preserve their bridge MAC address across a restart or RE switchover.

### Related Documentation

- Decision Sequence for a Loop-Free STP Topology on page 11
- Key Concepts in Spanning Tree Protocols on page 7
- Multiple Spanning Tree Protocol on page 21
- Overview of Spanning Tree Protocol on Juniper Networks MX Series 3D Universal Edge Routers on page 1
- Port Roles in STP on page 9
- Rapid Spanning Tree Protocol Port States and Port Roles on page 17
- Restrictions and Cautions for Implementing STP on page 33
- Spanning Tree Protocol Operation on page 3
- Spanning Tree Protocol States on page 13
- VLAN Spanning Tree Protocol on page 27





## Restrictions and Cautions for Implementing STP

---

This section presents restrictions and cautions when implementing the different STP versions:

- STP can be enabled on a physical interface using only one of these configurations:
  - Different STP versions are configured within a routing instance of the type **virtual-switch**. Currently under L2CPD, a virtual switch can only be configured under the default **logical-router** hierarchy.
  - Creating a routing instance of type **layer2-control**
  - Different STP versions are configured within the default virtual switch of the default logical system.
- These are the restrictions when STP is enabled on a physical interface within a **virtual-switch** configuration:
  - No other virtual switch can have logical interfaces on this physical interface.
  - Logical interfaces on this physical interface cannot belong to any routing instance of the type **vpls**.
  - If other logical interfaces on this physical interface have routing/ccc/tcc enabled, then this traffic is not affected by the STP state of the physical interface.
- Different STP versions are supported on CE interfaces of VPLS **routing-instance** by creating a routing instance of type **layer2-control**, which cannot have any bridge domains defined.
- If STP is not configured on a physical interface, the restrictions listed above do not apply. Different logical interfaces on a physical interface can span virtual switches as well as routing instances of the type **vpls**.
- Virtual switches cannot have any logical interfaces corresponding to physical interfaces that are referenced within a routing instance of the type **layer2-control**, as shown in the following example configuration.

```
routing-instance l2control-4-green {
  instance-type layer2-control;
  protocols {
    rstp {
      interface ge-1/2/3 ;
      interface ge-2/2/3 ;
      interface ge-3/2/4 ;
    }
  }
}
routing-instance green {
  instance-type vpls ;
  interface ge-1/2/3.11 ;
  interface ge-2/2/3.1 ;
  interface ge-3/2/4.2 ;
```

}

- Any given logical interface of a bridge domain can be associated with only one MSTI. All VLANs that are part of a VLAN range of a logical interface must be associated with the same MSTI. An MSTI must be a super-set of all VLANs that are associated with a logical interface.
- A bridge domain on the MX Series is either associated with a specific VLAN or all VLANs (**vlan-all**); you cannot use **vlan-range**. If a bridge domain is configured as **vlan-all**, it represents multiple broadcast domains, and it can be associated with multiple MSTIs. Each VLAN within **vlan-all** is an independent and separate broadcast domain.
- If you include the **vlan** statement within a bridge domain and specify the **none** option, then MSTP cannot be enabled on physical interfaces corresponding to the logical interfaces defined under the bridge domain.
- MSTP cannot be configured on physical interfaces that have logical interfaces with VLAN translation. MSTP is not allowed within a bridge domain if none of the logical interfaces in the bridge domain has a **vlan-range** and:
  - All of the logical interfaces in the bridge domain do not have the same VLANOR
  - All of the logical interfaces in the bridge domain have the same VLAN, but the **vlan-id** defined for the bridge domain is different.
- STP and RSTP can be configured on physical interfaces that have logical interfaces with VLAN translation.

#### Related Documentation

- Decision Sequence for a Loop-Free STP Topology on page 11
- Key Concepts in Spanning Tree Protocols on page 7
- Multiple Spanning Tree Protocol on page 21
- Overview of Spanning Tree Protocol on Juniper Networks MX Series 3D Universal Edge Routers on page 1
- Port Roles in STP on page 9
- Rapid Spanning Tree Protocol Port States and Port Roles on page 17
- Spanning Tree Protocol Operation on page 3
- Spanning Tree Protocol States on page 13
- STP Scaling and Performance on Juniper Networks MX Series 3D Universal Edge Routers on page 31
- VLAN Spanning Tree Protocol on page 27