

## Chapter 9

# Spanning Trees

Introduction .....	9-3
Spanning Tree Protocol (STP) .....	9-3
Multiple Spanning Trees and STP Interaction with VLANs .....	9-3
Spanning Tree Operation .....	9-4
The Forwarding Process .....	9-5
Electing a Root Bridge and Designated Bridge .....	9-6
Spanning Tree Modes .....	9-6
Rapid Mode Spanning Tree Types .....	9-7
Spanning Tree and Rapid Spanning Tree Port States .....	9-7
Configuring STP .....	9-8
Simple Meshed Network Example .....	9-12
Multiple Spanning Tree Protocol (MSTP) .....	9-13
Multiple Spanning Tree Regions .....	9-13
Bridge Protocol Data Units (BPDUs) .....	9-14
Compatibility with Previous Spanning Tree Protocols .....	9-15
Configuring MSTP .....	9-17
Common and Internal Spanning Tree (CIST) .....	9-23
The Relationship between Spanning Trees and Trunks .....	9-27
Command Reference .....	9-28
activate mstp migrationcheck port .....	9-28
add mstp msti vlan .....	9-29
add stp vlan .....	9-30
create mstp msti .....	9-31
create stp .....	9-33
delete mstp msti vlan .....	9-34
delete stp vlan .....	9-35
destroy mstp msti .....	9-36
destroy stp .....	9-37
disable mstp .....	9-38
disable mstp cist port .....	9-38
disable mstp debug .....	9-39
disable mstp msti port .....	9-40
disable mstp port .....	9-41
disable stp .....	9-42
disable stp debug .....	9-43
disable stp port .....	9-45
disable switch stpforward .....	9-46
enable mstp .....	9-46
enable mstp cist port .....	9-47
enable mstp debug .....	9-48
enable mstp msti port .....	9-50

enable mstp port .....	9-51
enable stp .....	9-52
enable stp debug .....	9-53
enable stp port .....	9-55
enable switch stpforward .....	9-56
purge mstp .....	9-57
purge stp .....	9-57
reset mstp counter port .....	9-58
reset stp .....	9-58
set mstp .....	9-59
set mstp cist .....	9-61
set mstp cist port .....	9-62
set mstp msti .....	9-64
set mstp msti port .....	9-65
set stp .....	9-67
set stp port .....	9-70
show mstp .....	9-74
show mstp cist .....	9-77
show mstp cist port .....	9-79
show mstp counter port .....	9-81
show mstp debug .....	9-82
show mstp msti .....	9-83
show mstp msti port .....	9-85
show stp .....	9-87
show stp counter .....	9-91
show stp debug .....	9-93
show stp port .....	9-94

## Introduction

---

The concept of the spanning tree protocol was devised to address broadcast storming. The spanning tree algorithm itself is defined by the IEEE in its 802.1D standard and its later revisions.

The IEEE Standard 802.1 uses the term “bridge” to define the spanning tree operation and uses terms such as Bridge Protocol Data Units, Root Bridge etc, when defining spanning tree protocol functions.

This Software Reference uses the term “switch” to mean a device that performs either the bridge or router function. However, for clarity and compatibility with the IEEE terminology, the term “bridge” is used in this chapter where text relates to these specific devices or their protocols as defined in the IEEE standard. The term “switch” is also used where this clearly represents a physical device that is operating the associated protocol or function.

When a bridge receives a frame, it reads the source and destination address fields. The bridge then enters the frame’s source address in its forwarding database. In doing this the bridge associates the frame’s source address with the network attached to the port on which the frame was received. The bridge also reads the destination address and if it can find this address in its forwarding database, it forwards the frame to the appropriate port. If the bridge does not recognise the destination address, it forwards the frame out from all its ports except for the one on which the frame was received, and then waits for a reply. This process is known as “flooding.”

Unfortunately, a significant problem arises where bridges connect via multiple paths. The arrival of a frame with an unknown destination address is flooded over all available paths. The arrival of these frames at another network via different paths and bridges produces major problems. The bridges can become confused about the location of the send and receive devices and begin sending frames in the wrong directions. This process feeds on itself and produces a condition known as a broadcast storm, where the increase of circulating frames can eventually overload the network.

## Spanning Tree Protocol (STP)

---

In order to avoid network loops, the Spanning Tree Protocol (STP) automatically disables redundant paths. The protocol can automatically enable these paths if a fault in the network means they are needed to keep traffic flowing.

## Multiple Spanning Trees and STP Interaction with VLANs

In a legacy network that has no VLANs configured, and has STP enabled, switches in the LAN run a distributed Spanning Tree Algorithm to create a single Spanning Tree.

In a network of switches with VLANs configured, all VLANs belong to a default Spanning Tree called *default*. Multiple Spanning Trees can be created with each Spanning Tree encompassing multiple VLANs. Spanning Tree Protocol entities, called STPs here, operate independently of each other; each

STP has its own root bridge and active path. Once an STP instance is created, one or more VLANs can be assigned to it. In operation, additional STP instances in the switch place no significant burden on the CPU.

If creating multiple STPs in a network, consider the following:

- A VLAN can only belong to a single STP.
- A port can belong to multiple STPs when the port is a member of more than one VLAN.

Where multiple spanning tree operation is required, you can use [“Multiple Spanning Tree Protocol \(MSTP\)” on page 9-13](#).

The concepts and examples in this section assume a single spanning tree, and either a single VLAN or multiple VLAN operation.

## Spanning Tree Operation

As explained in the section [“Introduction” on page 9-3](#), where a LAN’s topology results in more than one path existing between bridges, frames transmitted onto the extended LAN circulate in increasing numbers around the loop, decreasing performance and potentially overloading the network. However, multiple paths through the extended LAN are often required in order to provide redundancy and backup in the event of a bridge or link failure.

The spanning tree is created through the exchange of Bridge Protocol Data Units (BPDUs) between the bridges in the LAN. The spanning tree algorithm operates by:

- Automatically computing a loop-free portion of the topology, called a *spanning tree*. The topology is dynamically pruned to the spanning tree by declaring certain ports on a switch to be redundant, and placing them into a ‘Blocking’ state.
- Automatically recovering from a switch failure that would partition the extended LAN by reconfiguring the spanning tree to use redundant paths, if available.

The logical tree computed by the spanning tree algorithm has the following properties:

- A single bridge, is selected to become the spanning tree’s unique *root bridge*. This is the device that advertises the lowest Bridge ID. Each bridge is uniquely identified by its Bridge ID, which comprises the bridge’s *root priority* (a spanning tree parameter) followed by its MAC address.
- Each bridge or LAN in the tree, except the root bridge, has a unique parent, known as the *designated bridge*. Each LAN has a single bridge, called the *designated bridge*, that connects it to the next LAN on the path towards the root bridge.
- Each port connecting a bridge to a LAN has an associated *cost*, called the *root path cost*. This is the sum of the costs for each path between the particular bridge port and the root bridge. The designated bridge for a LAN is the one that advertises the lowest *root path cost*. If two bridges on the same LAN have the same lowest root path cost, then the switch with the lowest bridge ID becomes the designated bridge.

The spanning tree computation is a continuous, distributed process. The algorithm uses the following process to establish the spanning tree:

1. A unique *root bridge* is elected by the bridges on the LAN.
2. A *designated bridge* is elected for each LAN in the extended LAN by the switches in the LAN.
3. The spanning tree is computed and redundant paths are removed.

Once the spanning tree is established, it is maintained by:

1. Replacing a failed path with a redundant backup path, if one is available.
2. Detecting and removing loops by declaring ports redundant and removing them from the spanning tree.
3. Maintaining timers that control the ageing of the forwarding database entries (STP only).

The logical spanning tree, sometimes called the *active topology*, includes the root bridge and all designated bridges, meaning all ports that are to be used for communication within the STP. These ports are in the forwarding state. Ports removed from the logical spanning tree are not in the forwarding state. To implement the spanning tree algorithm, switches communicate with one another using the Spanning Tree Protocol. The primary protocol data unit (PDU) is the *Hello message* or *Configuration Bridge Protocol Data Unit* (BPDU). It includes the following information:

- The bridge ID of the root bridge.
- The distance (or cost) from this switch to the root bridge.
- The bridge ID of the designated bridge on this LAN.
- Hello messages are initiated at regular intervals by the root bridge and propagate through the extended LAN (STP only; in other spanning tree versions, the bridges themselves initiate their hello message transmissions).

## The Forwarding Process

The forwarding process forwards received frames that are to be relayed to other ports in the same VLAN, filtering out frames on the basis of information contained in the station map and on the state of the ports. When a frame is received on the port for a destination in a different VLAN, it is either Layer 3 switched if it is an IP packet, or looked up in the Layer 3 routing tables.

Forwarding occurs only when the port on which the frame was received is in the Spanning Tree forwarding or disabled states. The destination address is then looked up in the forwarding database for the VLAN. If the destination address is not found, the switch floods the frame on all ports in the VLAN except the port on which the frame was received. If the destination address is found, the switch discards the frame if the port is not in the STP forwarding or disabled states, if the destination address is on the same port as the source address, or if there is a static filter entry for the destination address set to **discard** (see [“Layer 2 Filtering” on page 8-31 of Chapter 8, Switching](#)). Otherwise, the frame is forwarded on the indicated port.

This whole process can further be modified by the action of static switch filters. These are configurable filters that allow switched frames to be checked against a number of entries.

The forwarding process provides storage for queued frames to be transmitted over a particular port or ports. More than one transmission queue may be provided for a given port. The transmission queue where a frame is sent is determined by the user priority tag in the Ethernet frame and the Quality of Service mapping (see [“Quality of Service” on page 8-31](#)).

## Electing a Root Bridge and Designated Bridge

Each spanning tree or spanning tree instance (in STP) has a *root bridge*, which controls the propagation rate of hello messages through the extended LAN, and sets the values of parameters that control the spanning tree computation process. Whereas, in RSTP and MSTP each bridge can control the transmission of their own periodic hello frames (BPDUs).

The bridge with the lowest bridge ID value becomes the root and is elected by the exchange of *hello* frames. When a switch receives a hello frame it compares the value of the root bridge ID in the message to the value of the root bridge ID parameter in its own spanning tree database. If the value in the message is better, the switch stores the new value in its database and sends hello messages with the new value out on its other ports. Otherwise, the switch continues to send hello messages with the value currently stored in its spanning tree database. By this process, all bridges in the extended LAN eventually learn the bridge ID of the root bridge.

Each LAN has a single *designated bridge* that logically connects one LAN to another LAN that is closer to the root bridge. The designated bridge for a LAN is the bridge that advertises the lowest root path cost and bridge ID. The designated bridge is elected by the exchange of hello messages, in the same way that the root bridge is elected. The election of a new root bridge due to a fault condition typically results in the election of a new designated bridge during the next few rounds of hello messages.

## Spanning Tree Modes

STP can run in *standard* mode or *rapid* mode. Rapid mode allows rapid configuration of the spanning tree. The Rapid Spanning Tree Protocol (RSTP) is specified in IEEE Standard 802.1w.

A spanning tree running in standard mode can take up to one minute to rebuild after a topology or configuration change. The RSTP algorithm provides for a faster recovery of connectivity following the failure of a bridge, bridge port, or a LAN. RSTP provides rapid recovery by including port roles in the computation of port states, and by allowing neighbouring bridges to explicitly acknowledge signals on a point-to-point link that indicate that a port wants to enter the forwarding mode.

In rapid mode, the rapid transition of a port to the forwarding state is possible when the port is considered to be part of a point-to-point link, or when the port is considered to be an *edge* port. An edge port is one that attaches to a LAN that has no other bridges attached.

To ensure that rapid transitions take place on an edge port, the port must be explicitly configured with the [set stp port command on page 9-70](#).

## Rapid Mode Spanning Tree Types

The RSTP algorithm has two types of operation: *normal* and *stp compatible*. If normal is specified, the algorithm uses rapid port role transitions and transmits and receives RST BPDUs. If STP compatible is specified, then rapid transitions are disabled and RST BPDUs are discarded. The default is normal. Setting the RSTP type to be STP compatible allows RSTP to support applications and protocols that may be sensitive to frame duplication and misordering, for example NetBeui.

Setting **rstptype** to **normal**, when normal has already been set, sets all ports to the “sending RSTP” state. This is referred to in IEEE Standard 802.1 as *mCheck*, and is useful for restoring full rapid mode operation when one or more ports on the switch has entered the “sending STP” state. RSTP capable devices - operating with RSTP set to **normal** - that receive the RST BPDUs, enter the “sending RSTP” state. After the mCheck operation, if an STP BPDU is received, either as a result of a device operating in rapid mode with **rstptype** set to **stpcompatible**, or as a result of a device operating in standard mode, the ports that received the STP BPDUs revert to the “sending STP” state.

Please note that mCheck is most effective on switches acting as designated bridges for LANs because they regularly propagate BPDUs. Other bridges in the LAN do not transmit BPDUs as frequently.

## Spanning Tree and Rapid Spanning Tree Port States

If STP is running in *standard* mode, then each port can be in one of five spanning tree states, and one of two switch states. If STP is running in *rapid* mode, then each port can be in one of four states. The state of a switch port is taken into account by STP. To be involved in STP negotiations, STP must be enabled on the switch, the port must be enabled on the switch, and enabled for the STP it belongs to.

The Spanning Tree port states ([Table 9-1](#) and [Table 9-2](#)) affect the behaviour of ports whose switch state is enabled.

Table 9-1: Spanning Tree port states

State	Meaning
DISABLED	STP operations are disabled on the port. The port does not participate in frame relay or the operation of the Spanning Tree Algorithm and Protocol. The port can still switch if its switch state is enabled.
BLOCKING	The forwarding process discards received frames and does not submit forwarded frames for transmission. This is the “standby” mode. The port does not participate in frame relay.
LISTENING	The port is enabled for receiving frames only. The port is preparing to participate in frame relay. The forwarding process discards received frames and does not submit forwarded frames for transmission.
LEARNING	The port is enabled for receiving frames only, and the Learning Process can add new source address information to the Forwarding Database.

Table 9-1: Spanning Tree port states (cont.)

State	Meaning
FORWARDING	The normal state for a switch port. The forwarding process and the Spanning Tree entity are enabled for transmit and receive operations on the port.

Table 9-2: Rapid Spanning Tree port states

State	Meaning
DISABLED	STP operations are disabled on the port.
DISCARDING	The port does not participate in frame relay. The forwarding process discards received frames and does not submit forwarded frames for transmission.
LEARNING	The port is enabled for receiving frames only, and the learning process can add new source address information to the forwarding database. The port does not forward any frames.
FORWARDING	The normal state for a switch port. The forwarding process and the Spanning Tree entity are enabled for transmit and receive operations on the port.

## Configuring STP

By default, the switch has one STP instance which cannot be destroyed. In most situations this one STP instance is sufficient. However, you can create additional STP instances by using the commands:

```
create stp=stp-name
destroy stp={stp-name|all}
```

By default all VLANs belong to the default STP instance, and because a VLAN can only be a member of one STP instance (unless you are running MSTP), all ports initially belong to the default STP instance. To add or delete a VLAN and all the ports belonging to it from any other STP, use the commands:

```
add stp vlan=stp-name vlan={vlan-name|2..4094}
delete stp vlan=stp-name vlan={vlan-name|2..4094|all}
```

The default STP is disabled by default at switch start up, and STPs created by a user are disabled by default when they are created. To enable or disable STPs, use the commands:

```
enable stp={stp-name|all}
disable stp={stp-name|all}
```

The Spanning Tree Protocol uses three configurable parameters for the time intervals that control the flow of STP information on which the dynamic STP topology depends: the **hellotime**, **forwarddelay**, and **maxage** parameters. All switches in the same spanning tree topology must use the same values for these parameters, but can themselves be configured with different, and potentially incompatible time intervals. The parameter values actually used by each switch are those sent by the root bridge, and forwarded to all other switches by the designated Bridges.

The **hellotime** parameter, with a default of 2 seconds, determines how often the switch sends hello messages containing spanning tree configuration



information if it is the *Root Bridge*, or is trying to become the Root Bridge in the network. Setting a shorter value for **hellotime** than the default of 2 seconds makes the network more robust; setting a longer time uses less processing overhead.

The **maxage** parameter, with a default of 20 seconds, determines the maximum time that dynamic STP configuration information is stored in the switch, before it is considered too old, and discarded. The value can be set at approximately two seconds for every hop across the network. If this value is too small, the STP may sometimes configure unnecessarily. If it is too long, there can be delays in adapting to a change in the topology, for instance when a fault occurs.

The **forwarddelay** parameter is used to prevent temporary loops in the network occurring in the briefly unstable topology while a topology change is propagated through the network. When STP is running in standard mode and a port that has been in the Blocking state is to move into the forwarding state, it must first pass through the listening and learning states. The **forwarddelay** parameter determines how long a port remains in each of these intermediate states before moving to the forwarding state in the active topology; that is, half the time between when it is decided that the port becomes part of the spanning tree, and when it is allowed to forward traffic. When STP is running in rapid mode, a port must pass from the discarding state through the learning state to reach the forwarding state. In this case, the **forwarddelay** parameter should be at least half the time it takes for a topology change message to reach the whole network. A value that is too short risks the temporary creation of loops, which can seriously degrade switch performance. A longer value can result in delays in the network after topology changes. The default **forwarddelay** value is 15 seconds.

The **forwarddelay**, **maxage** and **hellotime** parameters should be set according to the following formulae, as specified in IEEE 802.1d:

$$2 \times (\text{forwarddelay} - 1.0 \text{ seconds}) \geq \text{maxage}$$

$$\text{maxage} \geq 2 \times (\text{hellotime} + 1.0 \text{ seconds})$$

To modify the parameters controlling these time intervals, use the command:

```
set stp={stp-name|all} [forwarddelay=4..30] [hellotime=1..10]
[maxage=6..40]
```

The value of the **priority** parameter is used to set the writable portion of the bridge ID, i.e. the first two octets of the (8-octet long) Bridge Identifier. The remaining 6 octets of the bridge ID are given by the MAC address of the switches. The Bridge Identifier parameter is used in all configuration Spanning Tree Protocol packets transmitted by the switch. The first two octets, specified by the **priority** parameter, determine the switch's priority for becoming the *Root Bridge* or a *Designated Bridge* in the network, with a lower number indicating a higher priority. In fairly simple networks, for instance those with a small number of switches in a meshed topology, it may make little difference which switch is selected to be the Root Bridge, and no modifications may be needed to the default **priority** parameter, which has a default of 32768. In more complex networks, one or more switches are likely to be more suitable candidates for the root bridge role, for instance by virtue of being more central in the physical topology of the network. In these cases the **priority** parameters for at least one of the switches should be modified.

To change the STP priority value, use the command:

```
set stp={stp-name|all} priority=0..65535
```

To restore STP timer and priority defaults, use the command:

```
set stp={stp-name|all} default
```

Changing the **priority** using either of the previous commands initialises the STP, so that elections for the root bridge and designated bridges begin again, without resetting STP counters. To display general information about STPs on the switch, use the command:

```
show stp [= {stp-name|all}]
```

Each port has a port priority, with a default of 128, used to determine which port should be the root port for the STP if two ports are connected in a loop. A lower number indicates the higher priority.

```
set stp={stp-name|all} po={port-list|all} portpriority=0..255
```

Each port also has a path cost, which is used if the port is the root port for the STP on the switch. The path cost is added to the root path cost field in configuration messages received on the port to determine the total cost of the path to the Root Bridge. The default **pathcost** values and the range of recommended **pathcost** values depend on the port speed and mode ([Table 9-3](#) and [Table 9-4](#)). If the path cost for a port is not explicitly set, it varies as the speed of the port varies.

Table 9-3: Path cost values and port speed for standard mode

Port speed	Default PATHCOST	Recommended PATHCOST range
10Mbps	100	50-600
100Mbps	19	10-60
1Gbps	4	3-10

Table 9-4: Path cost values and port speed for rapid mode

Port Speed	Default PATHCOST	Recommended PATHCOST range
Less than 100 Kb/s	200,000,000	20,000,000-200,000,000
1Mbps	20,000,000	2,000,000-20,000,000
10Mbps	2,000,000	200,000-2,000,000
100 Mbps	200,000	20,000-200,000
1 Gbps	20,000	2,000-20,000
10 Gbps	2,000	200-2,000
100 Gbps	200	20-200
1Tbps	20	2-200
10 Tbps	2	2-20

Setting the path cost to a larger value on a particular port is likely to reduce the traffic over the LAN connected to it. This may be appropriate if the LAN has a lower bandwidth, or if there are reasons for limiting the traffic across it. To modify the STP port path cost, use the command:

```
set stp={stp-name|all} po={port-list|all}
pathcost=1..200000000
```

If the path cost of a port has been explicitly set to a particular value, it can be returned to its self-adjusting default path cost and priority, using the command:

```
set stp={stp-name|all} po={port-list|all} default
```

When an STP is enabled in a looped or meshed network, it disables and enables particular ports belonging to it dynamically, to eliminate redundant links. All ports in a VLAN belong to the same STP, and their participation in STP configuration, and hence the possibility of them being elected to the STP's active topology is enabled by default. To enable or disable particular ports, use the commands:

```
enable stp port={port-list|all}
disable stp port={port-list|all}
```

STP treats a trunk group configured on both ends of a link as a single path, and applies the default path cost based on the bit rate of single link within the trunk. For this reason we recommend that the path cost for trunk groups be manually configured with a path cost value based on the aggregate bit rate of the whole trunk group.

To display STP port information, use the command:

```
show stp[={stp-name|all}] port={port-list|all}
```

The spanning tree algorithm can be recalculated at any time, and all timers and counters be initialised, using the command:

```
reset stp={stp-name|all}
```

To display STP counters, use the command:

```
show stp={stp-name|all} counter
```

Enabling one or more STP debugging modes for a period of time displays information for STP troubleshooting ([Table 9-5](#)) to the port on which the switch received the command, or to the console.

Table 9-5: STP debugging options

Option	Debug Mode	Description
MSG	Message	Decoded display of received and transmitted STP packets
PKT	Packet	Raw ASCII display of received and transmitted STP packets
STATE	State	Port state transitions.
ALL	All	All debug options

To enable, disable or show the debug modes, use the commands:

```
enable stp={stp-name|all} debug={msg|pkt|state|all}
[output=console] [timeout={1..4000000000|none}]
enable stp debug={msg|pkt|state|all} port={port-list|all}
[output=console] [timeout={1..4000000000|none}]
disable stp={stp-name|all} debug={msg|pkt|state|all}
disable stp debug={msg|pkt|state|all} port={port-list|all}
show stp debug
```

STP debugging can be enabled or disabled for either a specific port or a specific STP. Using one of these commands overrides the other.

Set **output** to **console** if this command is in a script. Each of the debug modes can be enabled or disabled independently. Use the **timeout** parameter to prevent the switch or the display from being overloaded with debugging data.

If necessary, all the STP configuration that users create on a switch can be removed so that all STPs except the default STP are destroyed and all other defaults are restored. Use the command:

```
purge stp
```



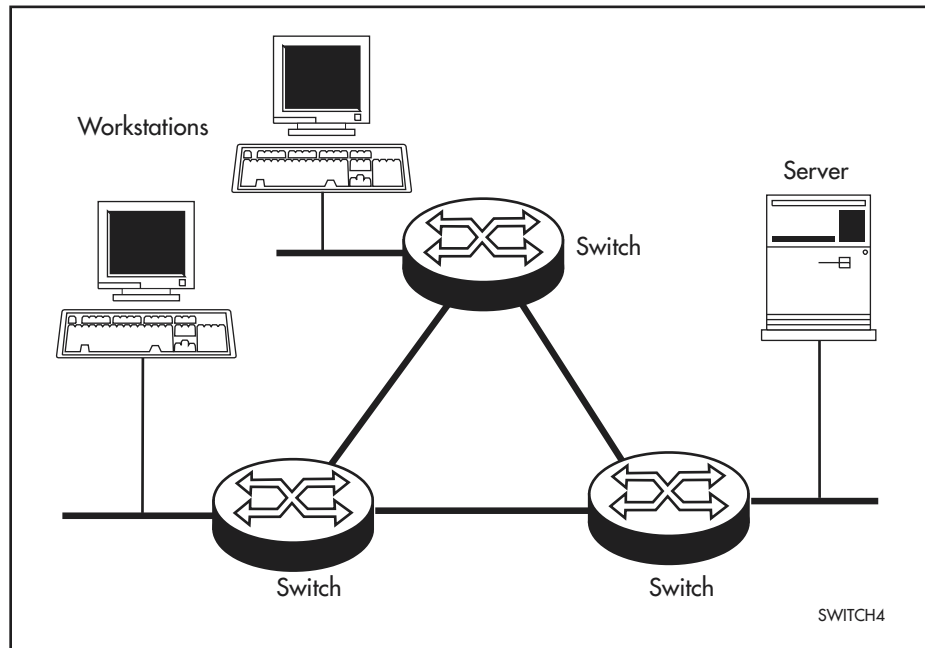
**Caution** The purge stp command should be used with caution, and generally only before major reconfiguration of the switch, as it removes all STP configuration entered on the switch.

## Simple Meshed Network Example

The following example shows how to configure STP on the switch. It assumes that the switch configuration begins from factory default settings.

The topology, shown in [Figure 9-1](#), has redundant links between the switches, and all ports belong only to the default VLAN. STP is needed because of the loop formed by the triangular links.

Figure 9-1: Example of switch with default configuration



### Configure all switches

The only software configuration required is to enable the default STP on each of the switches. Use the command:

```
enable stp=default
```

The switches begin switching as soon as they are physically connected and powered up.

## Multiple Spanning Tree Protocol (MSTP)

The multiple spanning tree protocol (MSTP) was developed to address the limitations in the existing spanning tree protocols, STP and RSTP. These limitations apply mainly to networks that use multiple VLANs with topologies employing alternative physical links. MSTP is defined in IEEE Standard 802.1Q 2003. The protocol builds on, and remains compatible with, the following previous standards:

- IEEE Standard 802.1w 2001, which defines the rapid spanning tree protocol (RSTP)
- IEEE Standard 802.1D 2004, which defines a draft standard for local and metropolitan area networks

## Multiple Spanning Tree Regions

Conceptually, MSTP views the total bridged network as one that comprises a number of *Multiple Spanning Tree Regions* (MSTRs), where each region can contain up to 64 spanning trees that operate locally, called *Multiple Spanning Tree Instances* (MSTIs). The task of assigning each bridge to a particular region is achieved by the member bridges each comparing their MST configuration identifiers. More information on configuration identifiers is provided in [Table 9-6 on page 9-13](#), but for the moment an *MST configuration identifier* can simply be thought of as an identifier that represents the mapping of VLANs to MSTIs within each bridge. Therefore, bridges with identical MST configuration identifiers, must have identical MSTI mapping tables.

While each MSTI can contain up to 4094 VLANs, each VLAN can be associated with only one MSTI. Once these associations have been made, the bridges in each region can transmit their spanning tree algorithms and advertise their MSTIs. This in turn establishes the active data paths between the bridges for each group of VLANs (i.e. for each MSTI) and block any duplicate paths. A particular advantage of this enhancement applies where a large number of VLANs share a few internetwork paths. In this situation there need only be as many Multiple Spanning Tree Instances (MSTIs) as there are source and destination bridge pairs, remembering that a pair of bridges probably has multiple paths between them.

In order to ensure that each bridge within a region maintains the same configuration information (particularly their VID to MSTI mappings) and to ensure each bridge's membership of a particular region, the bridges exchange configuration information in the form of "MST Configuration Identifiers." [Table 9-6 on page 9-13](#) provides a breakdown of an MST configuration identifier. A detailed explanation of bridge configuration identifiers can be found in Section 13.7 of the IEEE 802.1Q-2003 standard.

Table 9-6: MST Configuration identifier

Field Name	Description
Format Selector	A single octet field whose value of 0 indicates MSTP operation
Configuration Name	A name (up to 32 characters long) that identifies a particular MST region. The configuration name is defined using the <a href="#">set mstp command on page 9-59</a> .
Revision Level	A number representing the region's revision level. This value is normally set to 0.
Configuration Digest	A 16 octet (HMAC-MD5 based) signature created from the MST configuration table.

## Bridge Protocol Data Units (BPDUs)

The main function of bridge protocol data units is to enable MSTP to select its root bridges for the CIST (“[Common and Internal Spanning Tree \(CIST\)](#)” on [page 9-23](#)) and each MSTI. MSTP is compatible with earlier spanning tree versions; its Bridge Protocol Data Unit (BPDU) formats build on earlier versions (“[Compatibility with Previous Spanning Tree Protocols](#)” on [page 9-15](#)).

[Table 9-7](#) shows the standardised format for MSTP BPDU messages. The general format of the BPDUs comprise a common generic portion—octets 1 to 36—that are based on those defined in IEEE Standard 802.1D, 1998, followed by components that are specific to CIST—octets 37 to 102. Components specific to each MSTI are added to this BPDU data block.

Table 9-7: MST Bridge Protocol Data Units (BPDUs)

Field Name	Octets	Description
Protocol Identifier	1–2	Protocol being used. The value 0000 0000 0000 0000 identifies the spanning tree algorithm and protocol.
Protocol Version Identifier	3	Identifies the protocol version used.
BPDU Type	4	Value 0000 0000 specifies a configuration BPDU.
CIST Flags	5	Bit 1 is the topology change flag. Bit 2 conveys the CIST proposal flag in RST and MST BPDUs - unused in STP. Bits 3 & 4 convey the CIST port role in RST, and MST BPDUs - unused in STP. Bit 5 conveys the CIST learning flag in RST and MST BPDUs - unused in STP. Bit 6 conveys the CIST forwarding flag in RST and MST BPDUs - unused in STP. Bit 7 conveys the CIST agreement flag in RST and MST BPDUs - unused in STP. Bit 8 conveys the topology change acknowledge flag in STP configuration BPDUs - unused in RSTP and MSTP BPDUs.
CIST Root Identifier	6–13	The Bridge identifier of the CIST Root
CIST External Path Cost	14–17	The path cost between MST regions from the transmitting bridge to the CIST root.
CIST Regional Root Identifier	18–25	ID of the current CIST regional root bridge.
CIST Port Identifier	26–27	CIST port identifier of the transmitting bridge port.
Message Age	28–29	Message age timer value.
Max Age	30–31	Timeout value to be used by all bridges in the bridged network. This value is set by the root. Some implementations of MSTP may choose not to use this value.
Hello Time	32–33	Time interval between the generation of configuration BPDUs by the root bridge.
Forward Delay	34–35	A timeout value used to ensure forward delay timer consistency when transferring a port to the forwarding state. It is also used for ageing filtering database dynamic entries following changes in the active topology.

Table 9-7: MST Bridge Protocol Data Units (BPDUs) (cont.)

Field Name	Octets	Description
Version 1 Length	36	Used to convey the Version 1 length. It is always transmitted as 0.
Version 3 Length	37–38	Used to convey the Version 3 length. It is the number of octets taken by the parameters that follow in the BPDU.
MST Configuration Identifier	39–89	An identifier comprising elements of the following: Format Selector Configuration Name Revision Level Configuration Digest.
CIST Internal Root Path Cost	90–93	Path cost to the CIST regional root.
CIST Bridge Identifier	94–101	CIST bridge identifier of the transmitting bridge.
CIST Remaining Hops	102	Remaining hops which limits the propagation and longevity of received spanning tree information for the CIST.
MSTI Configuration Messages (may be absent)	103–39 plus Version 3 Length	See <a href="#">Table 9-8 on page 9-15</a> .

Table 9-8: MSTI configuration messages

Field Name	Octets	Description
MSTI Flags	1	Bits 1 through 8, convey the topology change flag, proposal flag, port role (two bits), Learning flag, forwarding flag, agreement flag, and master flag for this MSTI.
MSTI Regional Root Identifier	2–9	This includes the value of the MSTID for this configuration message encoded in bits 4 through 1 of octet 1, and bits 8 through 1 of octet 2.
MSTI Internal Root Path Cost	10-13	Internal Root Path Cost.
MSTI Bridge Priority	14	Bits 5 through 8 convey the value of the bridge identifier priority for this MSTI. Bits 1 through 4 of Octet 14 are transmitted as 0, and ignored on receipt.
MSTI Port Priority	15	Bits 5 through 8 are used to convey the value of the port identifier priority for this MSTI. Bits 1 through 4 are transmitted as 0, and ignored on receipt.
MSTI Remaining Hops	16	Value of remaining hops for this MSTI.

## Compatibility with Previous Spanning Tree Protocols

MSTP provides for compatibility with older spanning tree protocols in several ways. In addition to the MST region described in the previous section, the protocol provides for single spanning tree systems by employing a common and internal spanning tree (CIST) protocol. The CIST applies a common and

internal spanning tree protocol to the whole of the bridged network and is a direct equivalent the internal spanning tree (IST) protocol of earlier versions.

In common with legacy spanning tree systems, the CIST protocol first determines its root bridge from all the bridges on the network. This is the bridge that contains the lowest bridge identifier. The protocol then selects a regional root bridge for each MSTR. This is the bridge that provides the best path to the CIST root. After the MSTR root bridges have been chosen, they then act on the region's behalf in such a way that the region appears to the CST as a virtual bridge. So in addition to having multiple MSTIs, each region **must** operate as a bridge in a CST.



## Configuring MSTP

The configuration examples in this section are based on the network shown in [Figure 9-2](#). This simple network comprises three LAN bridges connected in a multi-linked mesh configuration.

The network is configured as a single MSTP region, called a MSTR, and given the name, Head Office. Two spanning tree instances (MSTIs) are created within this region called MSTI 2 and MSTI 3. For simplicity only two VLANs are configured VLAN 12 and VLAN 25; however, a typical MSTI network could have many more VLANs.

Two MSTIs are created (MSTI 2 and MSTI 3). MSTI 2 is assigned to VLAN12 and MSTI 3 is assigned to VLAN25. The network has several alternative links. By using MSTP each VLAN can be configured to use its own preferred set of links

Figure 9-2: Example configuration with MSTP

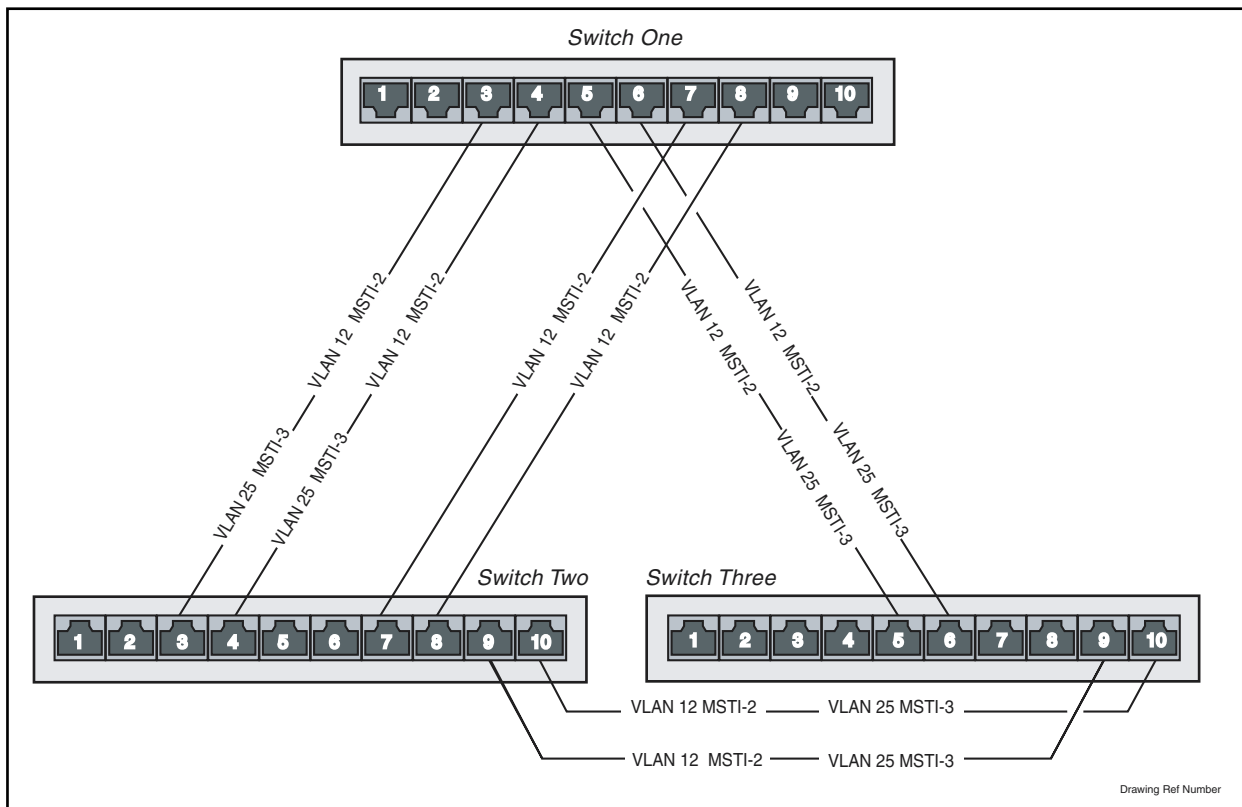


Figure 9-3: Example script for configuring Switch 1

```
# Configuring MSTP on Switch 1
# Name the system and set manager port
set system name=switch1
set manager asyn=0

# Create VLAN 12 and assign it a VID of 12
create vlan=vlan12 vid=12

# Create VLAN 25 and assign it a VID of 25
create vlan=vlan25 vid=25

# Add VLAN 12 to the required ports, as tagged ports
add vlan=12 po=3,4,5,6,7,8 frame=tagged

# Add VLAN 25 to the required ports, as tagged ports
add vlan=25 po=3,4,5,6 frame=tagged

# Set MSTP on Switch 1. Name the region Head Office and assign it a revision level
# of 0 (the value recommended in the IEEE standard)
set mstp configname=headoffice revision=0

# Enable static VLAN support on MSTP
set mstp staticvlans=on

# Create the MSTIs 2 and 3
create mstp msti=2
create mstp msti=3

# Add MSTI 2 to VLAN 12, and MSTI 3 to VLAN 25
add mstp msti=2 vlan=12
add mstp msti=3 vlan=25

# Assign priorities to each MSTI. These values are compared with those set on the #
# other switches in order to determine the root bridge for each MSTI
set mstp msti=2 prio=8192
set mstp msti=3 prio=8192

# Enable MSTP on the switch
ena mstp
```

Figure 9-4: Example script for configuring Switch 2

```
# Configuring MSTP on Switch 2
# Name the system and set manager port
set system name=switch2
set manager asyn=0

# Create VLAN 12 and assign it a VID of 12
create vlan=vlan12 vid=12

# Create VLAN 25 and assign it a VID of 25
create vlan=vlan25 vid=25

# Add VLAN 12 to the required ports, as tagged ports
add vlan=12 po=3,4,5,6,7,8,9,10 frame=tagged

# Add VLAN 25 to the required ports, as tagged ports
add vlan=25 po=3,4,9,10 frame=tagged

# Set MSTP on Switch 2. Name the region Head Office and assign it a revision level
# of 0 (the value recommended in the IEEE standard)
set mstp configname=headoffice revision=0

# Enable static VLAN support on MSTP
set mstp staticvlans=on

# Create the MSTIs 2 and 3
create mstp msti=2
create mstp msti=3

# Add MSTI 2 to VLAN 12, and MSTI 3 to VLAN 25
add mstp msti=2 vlan=12
add mstp msti=3 vlan=25

# Assign priorities to each MSTI. These values are compared with those set on the #
# other switches in order to determine the root bridge for each MSTI
set mstp msti=2 prio=8192
set mstp msti=3 prio=4096

# Enable MSTP on the switch
ena mstp
```

Figure 9-5: Example script for configuring Switch 3

```
# Configuring MSTP on Switch 3
# Name the system and set manager port
set system name=switch3
set manager asyn=0

# Create VLAN 12 and assign it a VID of 12
create vlan=vlan12 vid=12

# Create VLAN 25 and assign it a VID of 25
create vlan=vlan25 vid=25

# Add VLAN 12 to the required ports, as tagged ports
add vlan=12 po=5,6,8,10 frame=tagged

# Add VLAN 25 to the required ports, as tagged ports
add vlan=25 po=5,6,8,10 frame=tagged

# Set MSTP on Switch 3. Name the region Head Office and assign it a revision level
# of 0 (the value recommended in the IEEE standard)
set mstp configname=headoffice revision=0

# Enable static VLAN support on MSTP
set mstp staticvlans=on

# Create the MSTIs 2 and 3
create mstp msti=2
create mstp msti=3

# Add MSTI 2 to VLAN 12, and MSTI 3 to VLAN 25
add mstp msti=2 vlan=12
add mstp msti=3 vlan=25

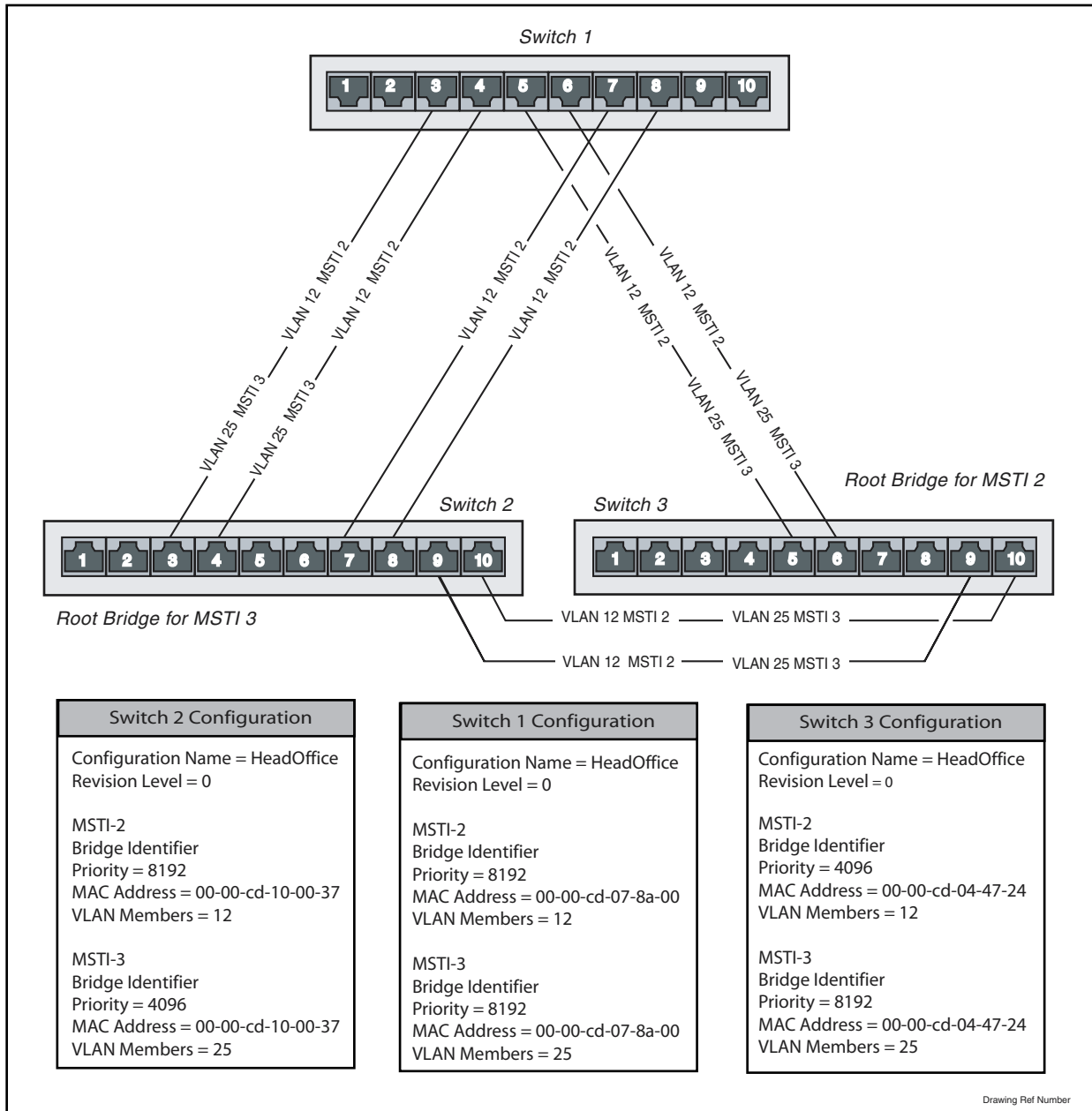
# Assign priorities to each MSTI. These values are compared with those set on the #
# other switches in order to determine the root bridge for each MSTI
set mstp msti=2 prio=4096
set mstp msti=3 prio=8192

# Enable MSTP on the switch
ena mstp
```

## Root bridge selection for MSTP MSTIs

The MSTP protocol selects its root bridges for each MSTI. It does this by selecting, for each MSTI, the bridge that contains (numerically) the lowest bridge identifier ([Figure 9-6](#)).

Figure 9-6: Example MSTP MSTI configuration



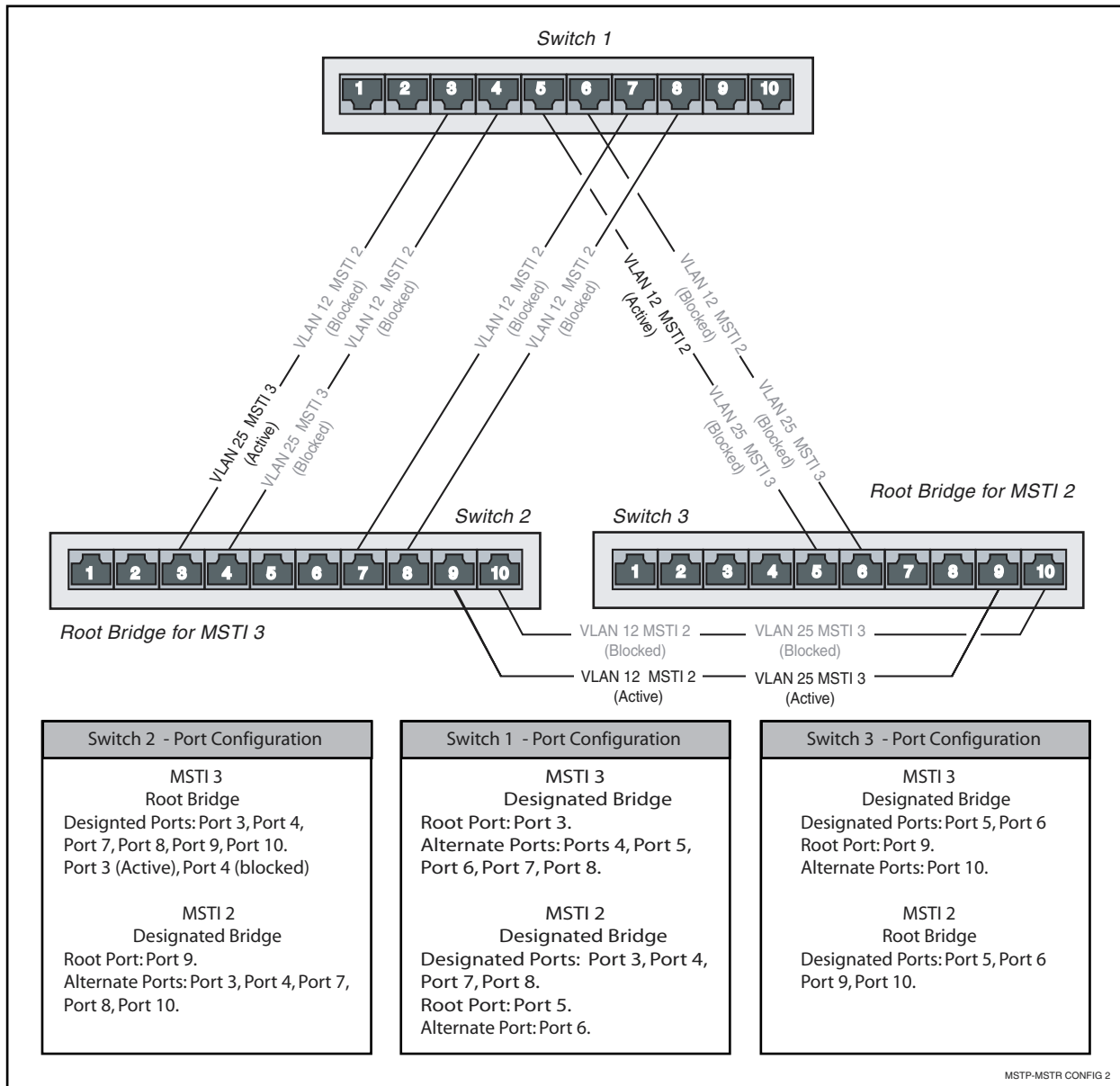
Notice that the root bridges are different for the two MSTIs. The root bridge for MSTI 2 is Switch 3 and the root bridge for MSTI 3 is Switch 2. This is because for MSTI 2 Switch 3 it has been given the lowest MSTI priority value, 4096, compared with 8192 for Switches 1 and 2.

Similarly, the root bridge for MSTI 3 is Switch 2 because its MSTI priority value has been set to 4096, compared with the value 8192 set for Switches 1 and 2. If all three bridges were configured with the same priority value for a particular MSTI, then Switch 3 would become the root bridge for that MSTI, because it has the lowest MAC address of the three switches.

## Path selection for MSTP MSTIs

After the protocol has selected its root bridge for each MSTI, it selects which are to be the active and blocked paths for each MSTI. The port with the best path to the root bridge is selected as the root port and becomes active. Other ports that also lead to the root bridge, but via a paths that have a higher cost than that via the root port, are considered alternate ports and are blocked to prevent loops. Ports that are connected to another port on the same switch, where that port has a better priority value, are backup ports and are blocked to prevent a loop. All other ports that are not disabled are selected as designated ports and are eventually made active. [Figure 9-7](#) shows which paths have been selected.

Figure 9-7: Example MSTP MSTI Path Configuration



For MSTI 3

Between Switches 1 and 2 there are two paths available, Port 3 to Port 3, and Port 4 to Port 4. Since no port priority has been explicitly applied, all port configurations have their defaults. Since all ports have the same speed (100 MBPS), each port has a Port Path Cost of 200,000. Since Port 3 is numerically lower than Port 4, the active path is the one between Switch 1 Port 3, and the other path is blocked. Similarly, the active path between Switches 2 and 3 is between Port 9 on each switch.

For MSTI 2

Between Switches 1 and 3 there are two paths available, Port 5 to Port 5, and Port 6 to Port 6. Since no port priority has been explicitly applied, all port configurations have their defaults. Since all ports have the same speed (100 MBPS) each port has a Port Path Cost of 200,000. Since Port 5 is numerically lower than Port 6, the active path is the one between Switch 1 Port 5 and Switch 2 Port 5, and the other path is blocked. Similarly, the active path between Switches 2 and 3, is between Port 9 on each switch.

If you want to make a particular path the active one, use the **set mstp msti port** command.

Example:

To balance the load between Switches 2 and 3, set the active path for MSTI 2 to be between Ports 10 and 10 of each switch. Use the following command to set the port path cost less than the present default of 200000:

For Switch 2

```
set mstp msti=2 port=10 pathcost=1000
```

For Switch 3

```
set mstp msti=2 port=10 pathcost=1000
```

## Configuration Check

To check the status of the paths and to see which are forwarding and which are blocked run the [show mstp msti port command on page 9-85](#), for a particular MSTI and port. From the output, note whether the port is a Root and whether its status is forwarding or blocking. If the port is a root port and is in the forwarding state, then its path is Active.

## Common and Internal Spanning Tree (CIST)

In addition to the individual MSTIs within each MSTR region, the MSTR contains a network-wide spanning tree called the Common and Internal Spanning Tree (CIST). Conceptually, each region represents a virtual bridge. Internal and external bridge connectivity are two independent functions.

Frames with VIDs allocated to the CIST are subject to the rules and path costs of the complete bridged LAN as determined by the CIST's vectors. Frames other than these are subject to the CIST when travelling outside their region, and subject to its particular MSTI inside the region.

The following operational rules apply:

- Each bridge can be a member of only one region.
- A data frame is associated with a single VID.
- Data frames with a given VID are associated with either the CIST or their particular MSTI, but not both.

The configuration examples in this section are based on the network shown in [Figure 9-8 on page 9-26](#). This simple network comprises six LAN bridges, and is basically two networks of the type used in the previous examples that are connected back to back.

## Configuring the CIST Example

Configuring this network involves the same basic steps used in the previous examples. Note that the only VLAN that is common to both regions is VLAN 12, which uses MSTI 3. These must be explicitly configured to Ports 1 and 10 of Switches 3 and 4.

For Switch 3

1. **Add VLAN 12 to the required ports, as tagged ports.**

```
add vlan=12 po=1,10 frame=tagged
set mstp msti=2 port=10 pathcost=1000
```

For Switch 4

1. **Add VLAN 12 to the required ports, as tagged ports.**

```
add vlan=12 po=1,10 frame=tagged
set mstp msti=2 port=10 pathcost=1000
```

If you configured the network using the steps in the previous example, and added the shared VLANs to the connecting ports as shown above, the network now has two regions: Region One representing a company's Head Office; and Region Two, representing the company's Manufacturing Plant. Note that although each network region is separate, with each of its MSTIs only having local significance within the region, the data itself still flows between the two networks and the VLANs in each are still recognised across MSTR boundaries.

The task of preventing loops within the wider network, is the role of CIST. By inspecting the example network, it is clear that there is a potential loop between the two regions that CIST must handle.

CIST first allocates root and designated bridges by selecting the bridge with the lowest identifier as the root. As far as the physical topology is concerned a good choice for the root bridge would be either of Switches 3 or 4. The network has been designed to force Switch 3 to become the root by assigning it the lowest priority identifier in the network (12288), and of course it is also the root bridge for Region One. Similarly, assigning Switch 4 the priority identifier of 20480 ensures that this bridge becomes the root bridge for Region 2 (because its priority identifier of 20480 is lower than any other bridge in its region). Switch 4 is also the CIST regional bridge since it offers the lowest path cost from Region 2 to Switch 3 (the CIST root bridge).

Note that the bridge identifier comprises two parts: a bridge priority part (more significant), and a bridge MAC address part (less significant). The multiple spanning tree algorithm uses the bridge identifier when determining the role of a switch within each spanning tree. The switch with a lower priority is considered to have better bridge identifier, and is therefore more likely to be chosen as the root bridge. You can set the CIST bridge priority using the **set mstp cist** command.

```
set mstp cist priority=20480
```



## CIST Vectors

Having selected the CIST Root and Designated bridge, the CIST then deals with any loops that exist between the regions. It does this by considering the following entities, called “vectors” in the following order:

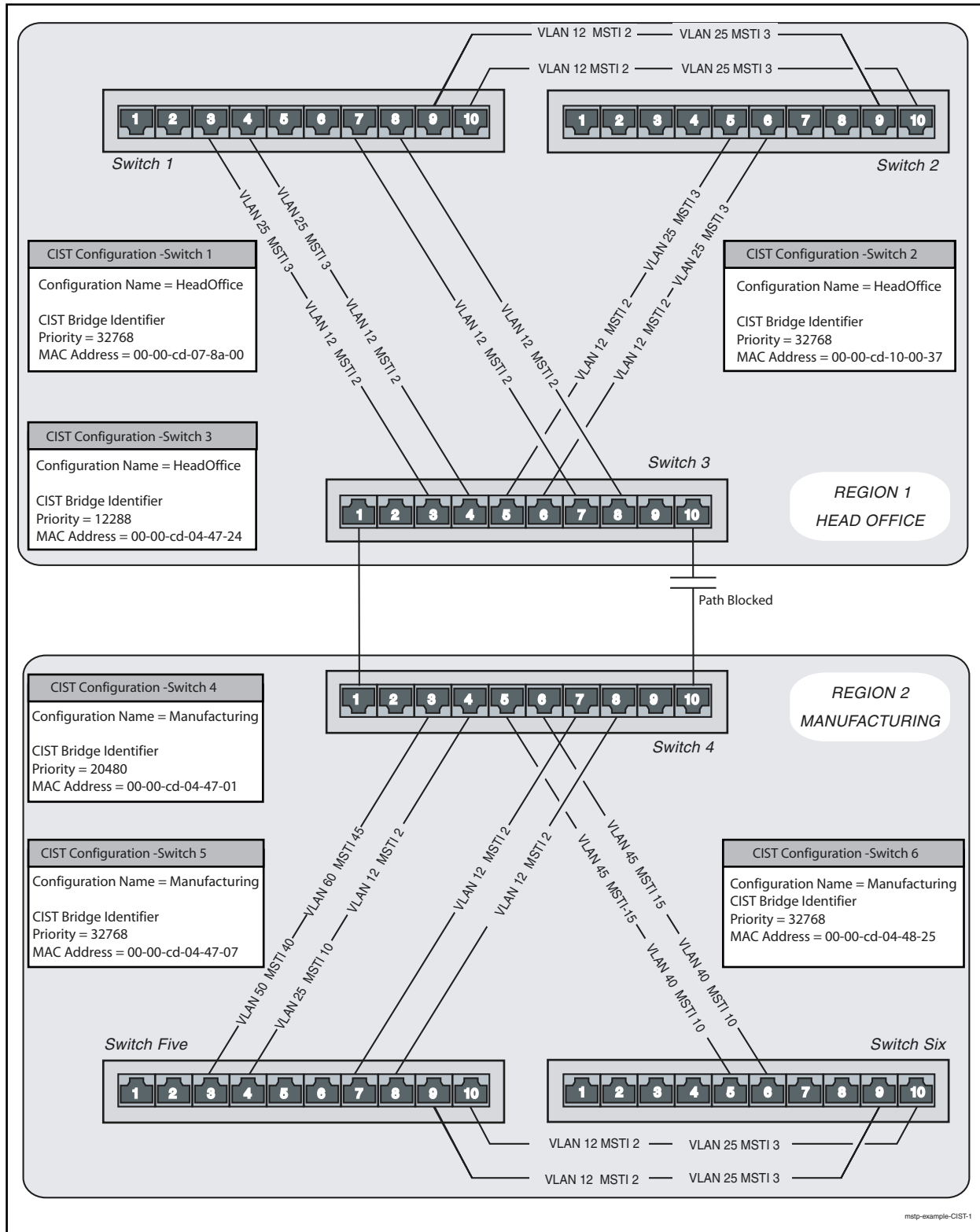
1. CIST External Root Path Cost
2. CIST Regional Root Identifier
3. CIST Internal Root Path Cost
4. CIST Designated Bridge Identifier
5. CIST Designated Port Identifier
6. CIST Receiving Port Identifier

Since there is clearly a loop condition between Switches 3 and 4, the CIST inspects each of the vectors. Assuming the two links from the same bridge have equal path costs, the active link is selected as the one from the port with the lowest port number. Hence the path between Port 10 on each switch is blocked.

Note the situation if the connections on Switch 4 were reversed, i.e. port 1 of Switch 3 being connected to port 10 of Switch 4, and port 1 of Switch 4 being connected to port 10 of Switch 3.

In the above situation, metric 5 above would apply (since metrics 1 through 4 would have the same value). The designated ports would be 1 and 10 on Switch 3, and since port 1 has the lower (numeric) value, this port would provide the active link, and the path from its port 10 would be blocked.

Figure 9-8: MSTP - CIST configuration example



## The Relationship between Spanning Trees and Trunks

If multiple links are trunked together, either manually or by using an automatic process such as LACP, the spanning tree application is notified and considers the links as a single logical path. Consequently, the spanning tree broadcast messages (BPDUs) only traverse the master trunk path.

Whether trunking offers a better solution depends on the individual network configuration. Users are recommended to consider both alternatives and select the option (Trunking or MSTP) that best meets the requirements of the particular network.

## Command Reference

---

This section describes the commands available to configure and manage the switching functions on the switch.

The shortest valid command is denoted by capital letters in the Syntax Section. See “Conventions” on page lxvi of [About this Software Reference](#) in the front of this manual for details of the conventions used to describe command syntax. See [Appendix A, Messages](#) for a complete list of messages and their meanings.

### activate mstp migrationcheck port

---

**Syntax** ACTivate MSTp MIGRationcheck Port={*port-list*|ALL}

where *port-list* is a port number, range (specified as *n-m*), or comma-separated list of port numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered Ethernet switch port, including uplink ports.

**Description** If an MSTP bridge detects the presence of STP data on one of its ports (from a legacy bridge) it automatically migrates the port to the STP protocol. Other MSTP and RSTP bridges connected to this port do the same. Thus all bridges that connect to this port revert to the STP protocol. However, this condition remains even after the original STP bridge has been removed.

Activating a migration check (mcheck) on such a port forces the bridge to migrate back to MSTP (or RSTP) and to transmit either MSTP (or RSTP) messages. After receiving these messages, other RSTP/MSTP bridges follow the same procedure. If no further STP bridge messages are received within a preset time period, then all the connected bridges remain in MSTP mode. The bridge decides whether to use RSTP or MSTP mode based on the setting of the **protocolversion** parameter of the MSTP command.

The **port** parameter specifies ports that are to have an mcheck applied to them. If **all** is specified, all ports in the switch are forced to the mcheck message. If the command would succeed on a subset of the ports specified, but cause an error on the others, then the command as a whole fails and has no effect.

The **port** parameter specifies the ports to transmit the mcheck messages. If **all** is specified, then all ports in the switch have an mcheck applied to them.

**Example** To transmit mcheck messages to all ports on the switch, use the command:

```
act mst migr po=all
```

**Related Commands** [show mstp](#)

## add mstp msti vlan

---

**Syntax** `ADD MSTP MSTI=instance VLAN={vlan-name|vlan-list|ALL}`

where:

- *instance* is an instance number from 1 to 4094 for a specific MSTI.
- *vlan-name* is a unique name for the VLAN, 1 to 32 characters in length. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen. The VLAN name cannot be a number or **all**.
- *vlan-list* is a VLAN number, range (specified as *n-m*), or comma-separated list of numbers and ranges. VLAN numbers start at 1 and end at 4094.

**Description** This command maps one or more VLANs to a specified multiple spanning tree instance (MSTI). The MST algorithm provides multiple spanning tree topologies within one MST region, so different VLANs can be forwarded in different paths.

All of the VLANs are mapped to the common internal spanning tree (CIST) by default. After a VLAN is mapped to a specified MSTI, it is removed from the CIST.

A VLAN can be mapped to only one MSTI or the CIST. One VLAN cannot be mapped to multiple spanning trees. A VLAN must be removed from one MSTI before it can be mapped to another. VLANs follow the CIST when operating between regions.

The **msti** parameter specifies the instance number of the spanning tree. The MSTI must already exist before any VLANs can be mapped to it. The command **create mstp msti** creates an MSTI.

The **vlan** parameter specifies a VLAN (or VLANs) to be mapped to the specified MSTI. If **all** is specified, then all VLANs are mapped to the MSTI. If a VLAN is already mapped to an MSTI other than the one specified in the command, then the command fails.

**Examples** To map a VLAN with VID of 1 to MSTI5, use the command:

```
add mst msti=5 vlan=1
```

**Related Commands**

- [delete mstp msti vlan](#)
- [create stp](#)
- [show mstp](#)
- [show mstp msti](#)

## add stp vlan

---

**Syntax** `ADD STP=stp-name VLAN={vlan-name|2..4094}`

where:

- *stp-name* is a character string 1 to 15 characters long. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen. The *stp-name* cannot be **all**.
- *vlan-name* is a unique name from 1 to 32 characters. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen. The *vlan-name* cannot be a number or **all**.

**Description** This command adds a VLAN to the specified STP. If as a result of the VLAN addition, ports are moved from one STP to another STP, the two affected STPs are initialised if they are currently enabled. Any previously disabled ports in the STPs are enabled.

The default VLAN cannot be added to an STP. The default VLAN always belongs to the default STP. A VLAN cannot be explicitly added to the default STP. A VLAN is implicitly added to the default STP when it is deleted from any other STP. Only a VLAN belonging to the default STP can be added to another STP. If the VLAN already belongs to another STP, it must first be deleted from its current STP (and so be returned to the default STP), and then added to the new STP.

If creating multiple STPs in a network, consider the following:

- A VLAN can only belong to a single STP.
- A port can belong to multiple STPs when the port is a member of more than one VLAN.

The **vlan** parameter specifies the name or the numerical VLAN Identifier of the VLAN to be added to the STP. The name is not case sensitive, although the case is preserved for display purposes. The VLAN specified must exist.

When a VLAN is added to an STP, the ports in the VLAN have default STP parameter values. The ports do not retain non-default STP configurations made when the VLAN was associated with any other STP.

**Examples** To add the *research* VLAN to the *company* STP, use the command:

```
add stp=company vlan=research
```

**Related Commands** [delete stp vlan](#)  
[show stp](#)

## create mstp msti

**Syntax** `CREate MSTp MSTI=instance [PRIOrity=0..65535]`

where *instance* is the instance number assigned to the new MSTI. It has the range 1-4094.

**Description** This command creates a new multiple spanning tree instance (MSTI) on the switch. The multiple spanning tree algorithm enables a collection of VLANs to be associated with a particular spanning tree instance. Within this instance, frames belonging to this VLAN group are forwarded over the active topology established by that particular instance's spanning tree. Frames for VLAN groups belonging to other instances each have their own active topologies.

Once an MSTI has been successfully created, VLANs can be added to it by using the command **add mstp msti vlan**.

Within each MST region, the MSTP maintains multiple spanning tree instances (MSTIs). A unique instance number identifies each single MSTI.

The MSTI parameter specifies the instance number of the multiple spanning tree instance (MSTI) being created. Although numbers can be assigned within the range 1 to 4094, the maximum number of MSTIs within each region, or switch, is 64. Instance number 0 is reserved for the common internal spanning tree (CIST) instance.

The MSTI number is very useful because it identifies a particular instance within an MST region.

The **priority** parameter sets the value of the priority field contained in the bridge identifier. The bridge identifier comprises two parts: a bridge priority part (more significant), and a bridge address part (less significant). The multiple spanning tree algorithm uses the bridge identifier when determining the role of a switch within each spanning tree. The switch with a lower priority is considered to have better bridge identifier, and is therefore more likely to be chosen as the root bridge. The CIST and each MSTI have their own individual **priority** parameter, so the roles of the same switch could be different in the CIST and each MSTI by tuning the bridge priority. The priority value operates in multiples of 4096. If you specify a value that is not a multiple of 4096, this is rounded down to the nearest multiple of 4096, see [Table 9-9 on page 9-31](#). The default switch priority is 32768.

Table 9-9: Rounding scheme for ranges of bridge priority parameter values

Lower Boundary	Upper Boundary	Rounded Bridge Value
0	4095	0
4096	8191	4096
8192	12287	8192
12288	16383	12288
16384	20479	16384
20480	24575	20480
24576	28671	24576
28672	32767	28672

Table 9-9: Rounding scheme for ranges of bridge priority parameter values (cont.)

Lower Boundary	Upper Boundary	Rounded Bridge Value
32768	36863	32768
36864	40959	36864
40960	45055	40960
45056	49151	45056
49152	53247	49152
53248	57343	53248
57344	61439	57344
61440	65535	61440

**Example** To create a new MSTI 5 with a priority of 8192, use the command:

```
cre mst msti=5 prio=8192
```

**Related Commands** [destroy mstp msti](#)  
[show mstp](#)  
[show mstp msti](#)



---

## create stp

---

**Syntax**    CREate STP=*stp-name*

where *stp-name* is a character string 1 to 15 characters long. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen. The *stp-name* cannot be **all** or **default**.

**Description**    This command creates a Spanning Tree Protocol entity with a unique name. The specified STP must not already exist. The name is not case sensitive, although the case is preserved for display purposes. The STP created is disabled by default. The maximum number of STPs that can be configured is 255.

A port can belong to more than one STP if the port is a member of two or more VLANs that belong to different STPs. The maximum number of STPs is 256.

**Example**    To create a new STP named *company*, use the command:

```
cre stp=company
```

**Related Commands**    [destroy stp](#)  
                          [enable stp](#)  
                          [set stp](#)  
                          [show stp](#)

## delete mstp msti vlan

---

**Syntax** DELEte MSTp MSTI=*instance* VLAN={*vlan-name*|*vlan-list*|ALL}

where:

- *instance* is an instance number from 1 to 4094 for a specific MSTI.
- *vlan-name* is a unique name for the VLAN, 1 to 32 characters in length. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen. The VLAN name cannot be a number or **all**.
- *vlan-list* is a VLAN number, range (specified as *n-m*), or comma-separated list of numbers and/or ranges. VLAN numbers start at 1 and end at 4094.

**Description** This command removes VLAN(s) from a specified MSTI. The removed VLANs are mapped to the CIST.

Once a VLAN is unmapped from a specified MSTI, the frames belonging to that VLAN are no longer forwarded along the spanning tree associated with that instance. The frames are forwarded along the CIST spanning tree.

The **msti** parameter specifies the instance number of the specified Multiple Spanning Tree Instance. Any VLANs that are not assigned to a specific MSTI explicitly are mapped to the CIST by default. There is no command to remove VLANs from the CIST.

The **vlan** parameter specifies the VLAN mapped to a specified MSTI. To unmap a VLAN from an MSTI it must have previously been mapped to the MSTI. If **all** is specified, all VLANs mapped to the MSTI are unmapped and re-mapped to the CIST.

**Examples** To delete the mapping of all VLANs from MSTI5, use the command:

```
del mst msti=5 vlan=all
```

**Related Commands**

- [add mstp msti vlan](#)
- [show mstp](#)
- [show mstp msti](#)

## delete stp vlan

---

**Syntax** `DELEte STP=stp-name VLAN={vlan-name|2..4094|ALL}`

where:

- *stp-name* is a character string 1 to 15 characters long. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen. The *stp-name* cannot be **all**.
- *vlan-name* is a unique name from 1 to 32 characters. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen. The *vlan-name* cannot be a number or **all**.

**Description** This command deletes one or all VLANs from the specified STP, and returns the VLANs to the default STP. A VLAN cannot be explicitly deleted from the default STP. The default VLAN cannot be deleted.

A port can belong to more than one STP after deletion. When a port belongs to multiple VLANs in the same STP, the port remains a member of this STP when a VLAN it was a member of is returned to the default STP.

If as a result of the VLAN deletion, ports are moved from one STP to another STP, the two affected STPs are initialised when they are currently enabled. Any previously disabled ports in the STPs are enabled.

When returned to the default STP, the ports of the VLAN have the default STP parameter values. The ports do not retain any non-default STP configuration that was made when the VLAN was associated with any other STP.

The **vlan** parameter specifies the name or numerical VLAN Identifier (VID) of the VLAN to be deleted. If **all** is specified, then all VLANs are deleted from the STP.

**Example** To delete the Research VLAN from the *company* STP, use the command:

```
del stp=company vlan=research
```

**Related Commands** [add stp vlan](#)  
[show stp](#)

## destroy mstp msti

---

**Syntax** DESTroy MSTp MSTI=*instance* [PRIOrity=0..65535]

where *instance* is the instance number assigned to the new MSTI. It has the range 1-4094.

**Description** This command destroys a specific multiple spanning tree instance (MSTI) on the switch. An MSTI cannot be destroyed when it still has VLANs mapped to it. Use the **delete mstp msti vlan=all** command to remove all VLANs from the specified MSTI.

**Example** To destroy an existing MSTI5, use the command:

```
dest mst msti=5
```

**Related Commands**

- [create stp](#)
- [delete mstp msti vlan](#)
- [show mstp](#)
- [show mstp msti](#)

---

## destroy stp

---

**Syntax** DESTroy STP={*stp-name*|ALL}

where *stp-name* is a character string 1 to 15 characters long. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen. The *stp-name* cannot be **all**.

**Description** This command destroys the specified Spanning Tree Protocol entity, or all STPs except the default STP. An STP cannot be destroyed if VLANs still belong to the STP.

The **stp** parameter specifies the name of the STP. The name is not case sensitive, although the case is preserved for display purposes. The **stp** specified must exist. The default STP cannot be destroyed. If **all** is specified, then all STPs except the default STP are destroyed. When **all** is specified and the command succeeds on a subset of STPs but causes errors on the others, then the command as a whole fails and has no effect.

**Examples** To destroy the *company* STP, use the command:

```
dest stp=company
```

To remove all user created STPs from the switch, none of which have VLANs belonging to them, use the command:

```
dest stp=all
```

**Related Commands**

- [create stp](#)
- [delete stp vlan](#)
- [disable stp](#)
- [enable stp](#)
- [set stp](#)
- [show stp](#)

## disable mstp

---

**Syntax** DISable MSTp

**Description** This command disables the multiple spanning tree operation on the switch. By default MSTP is disabled on switch start-up. This command overrides the following commands:

```
enable mstp cist port
disable mstp cist port
enable mstp msti port
disable mstp msti port
```

Once MSTP has been disabled, no port for the CIST or MSTIs can be enabled or disabled. MSTP must be disabled before any STP instances can be enabled.

**Examples** To disable MSTP, use the commands:

```
dis mst
```

**Related Commands** [enable mstp](#)  
[show mstp](#)

## disable mstp cist port

---

**Syntax** DISable MSTp CIST Port={*port-list*|ALL}

where *port-list* is a port number, range (specified as *n-m*), or comma-separated list of port numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered Ethernet switch port, including uplink ports.

**Description** This command disables operation of the Multiple Spanning Tree algorithm on specific ports or all ports for the Common Internal Spanning Tree. Disabled ports are placed in a discarding state and cannot forward frames. All of the ports are enabled for the CIST by default.

MSTP must be enabled first before any port for the CIST can be enabled or disabled.

The **port** parameter specifies a list of ports to be disabled for the CIST. If **all** is specified, all of the ports on the switch are disabled for the CIST. If the command would succeed on a subset of the ports specified, but cause an error on the others, then the command has no effect. Note that you can alternatively use the [disable mstp port](#) command to disable both the CIST and all MSTIs that are configured to a port or port range.

**Example** To disable port 2 in the CIST, use the command:

```
disable mstp cist port=2
```

**Related Commands** [show mstp msti](#)  
[show mstp cist port](#)

## disable mstp debug

---

**Syntax** `DISable MSTp DEBug={Msg|Pkt|State|All} MSTI={CIST|  
instance|ALL} [Port={port-list|ALL}]`

where:

- *instance* is the instance number of the selected MSTI in a range from 1 to 4094.
- *port-list* is a port number, range (specified as *n-m*), or comma-separated list of port numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered Ethernet switch port, including uplink ports.

**Description** This command disables MSTP debugging for a specified MST instance (MSTI) or all instances, or on specific port or ports.

The **msti** parameter specifies the instance for which the debugging mode is disabled. If **cist** is specified, then debug is disabled on the CIST. If an instance is specified, then debug is disabled on the MSTI. If ports are specified using the **port** parameter, then debug is disabled on the specified port on the specified instance. If **all** is specified and the ports are specified using the **port** parameter, then debug mode is disabled on all the instances for the listed ports.

The **debug** parameter specifies which debugging modes are to be disabled. If **all** is specified, then all debugging modes for the instances or ports are disabled. The other modes can be disabled independently of each other.

The **port** parameter specifies the ports on which the debug mode is disabled, or **all** ports on the switch.

**Example** To disable debugging on all ports in MSTI5, use the command:

```
dis mst msti=5 po=all
```

**Related Commands**

- [disable mstp debug](#)
- [show mstp msti](#)
- [show mstp msti port](#)
- [disable debug active](#) in Chapter 4, Configuring and Monitoring the System
- [show debug active](#) in Chapter 4, Configuring and Monitoring the System

## disable mstp msti port

---

**Syntax**    `DISable MSTp MSTI=instance Port={port-list|ALL}`

where:

- *instance* is the instance number of the specified MSTI in a range from 1 to 4094.
- *port-list* is a port number, range (specified as *n-m*), or comma-separated list of port numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered Ethernet switch port, including uplink ports.

**Description**    This command disables operation of the Multiple Spanning Tree algorithm on the specified ports or all ports for the specified Multiple Spanning Tree Instance. Disabled ports are placed in a discarding state and cannot forward frames. All ports are enabled for the specified **msti** by default.

MSTP must be enabled first before any port for the specified **msti** can be enabled or disabled.

The **msti** parameter specifies the instance number for the specified MSTI.

The **port** parameter specifies a list of ports to be disabled for the specified **msti**. If **all** is specified, all of the ports on the switch is disabled for the specified **msti**. Note that you can alternatively use the [disable mstp port](#) command to disable both the CIST and all MSTIs that are configured to a port or port range.

**Example**    To disable port 2 in MSTI5, use the command:

```
dis mst msti=5 po=2
```

**Related Commands**    [enable mstp](#)  
                          [enable mstp msti port](#)  
                          [show mstp msti](#)  
                          [show mstp msti port](#)



---

## disable mstp port

---

**Syntax** `DISable MSTP Port={port-list|ALL}`

where *port-list* is a port number, range (specified as *n-m*), or comma-separated list of port numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered Ethernet switch port, including uplink ports.

**Description** This command disables the Multiple Spanning Tree algorithm on the specified ports, or all ports, for both the CIST and all currently configured MSTIs. This command offers a shorter alternative to using the [disable mstp cist port](#) command, followed by the [disable mstp msti port](#) command.

**Example** To disable the CIST and all MSTIs on ports 10-15, use the command:

```
dis mstp po=10-15
```

**Related Commands**

- [disable mstp cist port](#)
- [disable mstp msti port](#)
- [enable mstp port](#)
- [show mstp cist](#)
- [show mstp msti](#)
- [show mstp cist port](#)
- [show mstp msti port](#)

## disable stp

---

**Syntax**    DISable STP={*stp-name*|ALL}

where *stp-name* is a character string 1 to 15 characters long. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen. The *stp-name* cannot be **all**.

**Description**    This command disables operation of the Spanning Tree Algorithm for the specified STP or for the entire switch. User created STPs are disabled by default. The default STP is disabled on switch start-up. An STP should be disabled only when its part of the LAN topology is free of loops. When there is a loop in the topology, the performance of the LAN can be significantly reduced.

This command overrides the **disable stp port** and **enable stp port** commands. Once an STP has been disabled by this command, no port belonging to that STP can be enabled or disabled. The STP must be enabled before ports belonging to the STP are enabled or disabled.

Disabling an STP does not affect the debug status of that STP set by the **enable stp debug** command. However, because the STP is disabled, STP debugging produces no information.

Disabling STP operation on a port may affect the operation of GARP. Each GARP application has a GIP component whose actions depend on whether the port is in the STP forwarding state.

**Examples**    To disable the *company* STP, use the command:

```
dis stp=company
```

To disable all STPs on the switch, use the command:

```
dis stp=all
```

**Related Commands**    [create stp](#)  
                          [destroy stp](#)  
                          [enable stp](#)  
                          [set stp](#)  
                          [show stp](#)

## disable stp debug

**Syntax** `DISable STP [= {stp-name | ALL}] DEBug = {MSG | PKT | STATE | ALL}  
Port = {port-list | ALL}`

`DISable STP DEBug = {MSG | PKT | STATE | ALL} Port = {port-list | ALL}`

where:

- *stp-name* is a character string 1 to 15 characters long. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen. The *stp-name* cannot be **all**.
- *port-list* is a port number, range (specified as *n-m*), or comma-separated list of numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered switch Ethernet port, including uplink ports.

**Description** This command disables STP debugging options for the specified STP or ports. The **debug** parameter specifies the debugging modes that are to be disabled ([Table 9-10 on page 9-43](#)).

A port can belong to more than one STP when the port is a member of two or more VLANs that belong to different STPs.

If **all** is specified, all debugging is disabled.

Table 9-10: STP debugging options

Option	Debug Mode	Description
MSG	Message	Decoded display of received and transmitted STP packets
PKT	Packet	Raw ASCII display of received and transmitted STP packets
STATE	State	Port state transitions.
ALL	All	All debug options

The **port** parameter specifies the ports where the debug mode is disabled.

The **port** parameter can be supplied with the STP name. If no STP name is provided, it assumes **all**. On the port parameter, the port list does not have to perfectly match all the STP port members so the command still succeeds as a whole.

The **stp** parameter specifies the STP for which the debugging mode is disabled. If an STP is specified, then the **port** parameter is invalid and all ports in the STP have the debug mode disabled.

The debug status of a port is not changed if the port is moved out of its current STP by one of the following commands:

- add vlan port
- delete vlan port
- add stp vlan, delete stp vlan

This command is effective on disabled ports or disabled STPs, but produces no debugging information until the ports and the STP are enabled.

**Examples**    To disable the state debugging mode for the STP named “company,” use the command:

```
dis stp=company deb=state
```

To disable all debug modes for all STPs, use the command:

```
dis stp=all deb=all
```

To disable the MSG debugging mode on ports 5 to 8, use the command:

```
dis stp deb=msg po=5-8
```

**Related Commands**    [enable stp debug](#)  
                          [show stp debug](#)  
                          [disable debug active](#) in Chapter 4, Configuring and Monitoring the System  
                          [show debug active](#) in Chapter 4, Configuring and Monitoring the System

## disable stp port

---

**Syntax** `DISable STP[={stp-name|ALL}] Port={port-list|ALL}`

where:

- *stp-name* is a character string 1 to 15 characters long. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen.
- *port-list* is a port number, range (specified as *n-m*), or comma-separated list of numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered switch Ethernet port, including uplink ports.

**Description** This command disables operation of the Spanning Tree Algorithm on the specified ports – normal switch processing continues. Disabled ports that are part of an enabled STP can still forward packets. This command is effective when the STP that the port belongs to is currently enabled. Disabling the operation of STP on a port does not affect the port's ability to receive and transmit frames.

A port can belong to multiple STPs when the port is a member of more than one VLAN.

Disabling the Spanning Tree Algorithm on one or more ports puts those ports in the Disabled state; all BPDUs received on these ports are discarded.

Disabling an STP port does not affect the debug status of the port as set by the [enable stp debug command on page 9-53](#). However, no STP debugging information is produced on a disabled port.

Disabling STP operation on a port may affect the operation of GARP. Each GARP application has a GIP component whose actions depend upon whether the port is in the STP forwarding state.

The STP parameter specifies the STP instance for which the port is disabled. If no value is provided, the default is **all**.

The **port** parameter specifies the ports. If **all** is specified, all ports in the switch are disabled. When the command succeeds on a subset of the specified ports but causes errors on the others, then the command as a whole fails and has no effect.

**Examples** To disable the Spanning Tree Algorithm from using port 4, use the command:

```
dis stp po=4
```

To disable STP on all ports, use the command:

```
dis stp po=all
```

To disable STP on just the administration network and only on port 4, use the command:

```
dis stp=admin po=4
```

**Related Commands** [enable stp port](#)  
[set stp port](#)  
[show stp port](#)

---

## disable switch stpforward

---

**Syntax**    DISable SWITch STPForward

**Description**    This command disables the switch from (transparently) forwarding spanning tree BPDUs. Forwarding is disabled by default.

**Example**    To disable switch STP forwarding, use the command:

```
dis swi stpf
```

**Related Commands**    [enable stp](#)  
                          [enable switch stpforward](#)  
                          [disable stp](#)

---

## enable mstp

---

**Syntax**    ENAbLe MSTp

**Description**    This command enables the operation of the multiple spanning tree algorithm on the switch. Multiple spanning tree protocol (MSTP) enables a number of VLANs to each use separate active topologies throughout a virtual bridged LAN. By default MSTP is disabled on switch start-up. MSTP must be enabled before the following commands can be used:

```
enable mstp cist port
```

```
disable mstp cist port
```

```
enable mstp msti port
```

```
disable mstp msti port
```

Once MSTP has been enabled, any port for the CIST and the existing MSTIs can be enabled or disabled. Enabling MSTP initialises the status for the switch and all of its ports. MSTP cannot be enabled while there are also STP instances enabled. All STP instances must be disabled before MSTP can be enabled.

**Examples**    To enable MSTP, use the commands:

```
ena mst
```

**Related Commands**    [disable mstp](#)  
                          [show mstp](#)

---

## enable mstp cist port

---

**Syntax** `ENABle MSTp CIST Port={port-list|ALL}`

where *port-list* is a port number, range (specified as *n-m*), or comma-separated list of port numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered Ethernet switch port, including uplink ports.

**Description** This command enables operation of the MST algorithm on specific ports or all ports for the CIST. All ports are enabled for the CIST by default. Note that you can alternatively use the [enable mstp port](#) command to enable both the CIST and all MSTIs that are configured to a port or port range.

MSTP must be enabled first before any port for the CIST can be enabled or disabled. If a port is a member of a trunk group but is not the master port then this command fails.

The **port** parameter specifies a list of ports to be enabled for the CIST. If **all** is specified, all ports on the switch are enabled for the CIST. If the command would succeed on a subset of the ports specified, but cause an error on the others, then the command as a whole fails and has no effect.

If a port is disabled with the [disable switch port command on page 8-82](#) or has a link status of down and this port is enabled, a message is displayed indicating the condition.

**Example** To enable all ports in the CIST, use the command:

```
ena mst cist po=all
```

**Related Commands** [show mstp cist](#)  
[show mstp cist port](#)

## enable mstp debug

**Syntax** `ENABle MSTp DEBUg={Msg|Pkt|State|All} MSTI={CIST|instance|ALL} [Port={port-list|ALL}] [Statemachine={PTM|PRX|PPM|PIM|PTX|PRS|PRT|PST|TCM|ALL}] [Output=Console] [Timeout=1..4000000000|None]`

where:

- *instance* is the instance number of the selected MSTI in a range from 1 to 4094.
- *port-list* is a port number, range (specified as *n-m*), or comma-separated list of port numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered Ethernet switch port, including uplink ports.

**Description** This command enables MSTP debugging for either a specified MSTP instance or all instances, or on specific port or ports.

The **msti** parameter specifies the spanning tree instance for which the debugging mode is enabled. If **cist** is specified, then debugging is enabled on the CIST. If an MST instance is specified, then debugging is enabled on the **msti** specified. If ports are specified using the **port** parameter, then the debug is enabled on the specified port on the specified instance. If **all** is specified and the ports are specified using the **port** parameter, then the debugging mode for the listed ports is enabled on all the instances with the listed ports.

The instance number is for the specified **msti**.

The **debug** parameter specifies which debugging modes are to be enabled. The debugging modes enabled by each option are shown in the following table. They can be enabled independently of each other.

Option	Description
MSG	Decoded display of BPDUs received and transmitted by MSTP
PKT	Raw ASCII display of BPDUs received and transmitted BY MSTP
STATE	Displays transitions for the MSTP state machines that are specified by the <b>statemachine</b> parameter
ALL	All debug modes for the instances or ports are enabled.

Setting the **output** parameter to **console** instructs the bridge to send the debugging information to the console. By default, the debugging data goes to the port that received the **enable mstp debug** command. This option should be selected if the **enable mstp debug** command is used in a script because a script is not received on a port.

The **port** parameter specifies which ports on the bridge for which the debug mode is enabled. If a value is not entered, the parameter defaults to **all**.

The **statemachine** parameter specifies which state machines has debugging enabled, see [Table 9-11 on page 9-49](#). This parameter is valid only when the debug mode is **state**. The default is **all**.

The value of this parameter is cleared only when the **disable mstp debug** command specifies the **debug** parameter as either **state** or **all**. When the debug mode is not **state** or **all**, the **statemachine** parameter is not cleared.



The **timeout** parameter specifies the time period, in seconds, during which debugging is enabled on the specified ports. Limiting the debugging time period reduces the risk of the switch and the display being overloaded with debugging information. Note that this parameter value overrides any previous MSTP debugging timeout values for these ports, even if they were specified for other debugging modes. If no **timeout** value is specified, then its value is **none**. When the timeout expires the following events occur:

- **output** is redirected to the console
- **debug** is disabled for all modes
- **statemachine** modes are all disabled
- **timeout** is set to **none**

Table 9-11: State Machine Mode in Debug and the State Machine

Option	Description
PTM	Port timer state machine
PRX	Port receive state machine
PPM	Port protocol migration state machine
PIM	Port information state machine
PTX	Port transmit state machine
PRS	Port role selection state machine
PRT	Port role transitions state machine
PST	Port state transition state machine
TCM	Topology change state machine

**Example** To enable debugging on all ports in MSTI5, use the command:

```
ena mst deb msti=5 po=all
```

**Related Commands**

- [disable mstp debug](#)
- [show mstp debug](#)
- [disable debug active](#) in Chapter 4, Configuring and Monitoring the System
- [show debug active](#) in Chapter 4, Configuring and Monitoring the System

## enable mstp msti port

---

**Syntax** ENABle MSTP MSTI=*instance* PORT={*port-list*|ALL}

where:

- *instance* is the instance number of the specified MSTI, having the range 1-4094.
- *port-list* is a port number, range (specified as *n-m*), or comma-separated list of port numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered Ethernet switch port, including uplink ports.

**Description** This command enables operation of the Multiple Spanning Tree algorithm on the specified ports or all ports for the specified Multiple Spanning Tree Instance. Note that alternatively you can use the [enable mstp port](#) command to enable both the CIST and all MSTIs that are configured to a port or port range.

MSTP must be enabled first before any port for the specified **msti** can be enabled or disabled.

The **msti** parameter specifies the instance number for the specified **msti**.

The **port** parameter specifies a list of ports to be enabled for the specified **msti**. If **all** is specified, all of the ports on the switch are enabled for the specified **msti**. If the command would succeed on a subset of the ports specified, but cause an error on the others, then the command has no effect.

If a port is a member of a trunk group but is not the master port, then the **enable mstp msti port** command has no effect.

If a port is either disabled by using the **disable switch port** command, or has a link status of *down* and the port is enabled, a message is displayed indicating the condition.

All of the ports are enabled for the specified **msti** by default.

**Example** To enable all ports in MSTI5, use the command:

```
ena mstp msti=5 port=all
```

**Related Commands** [show mstp msti](#)  
[show mstp msti port](#)

---

## enable mstp port

---

**Syntax**    ENABle MSTP PORt={*port-list*|ALL}

where *port-list* is a port number, range (specified as *n-m*), or comma-separated list of port numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered Ethernet switch port, including uplink ports.

**Description**    This command enables operation of the Multiple Spanning Tree algorithm on the specified ports, or all ports, for the both the CIST and all currently configured MSTIs. This command offers a shorter alternative to using the [enable mstp cist port](#), command, followed by the [enable mstp msti port](#) commands.

**Example**    To enable the CIST and all MSTIs on ports 10-15, use the command:

```
ena mstp po=10-15
```

**Related Commands**    [disable mstp port](#)  
                          [enable mstp cist port](#)  
                          [enable mstp msti port](#)  
                          [show mstp cist](#)  
                          [show mstp msti](#)  
                          [show mstp cist port](#)  
                          [show mstp msti port](#)

## enable stp

---

**Syntax** ENable STP{=*stp-name*|ALL}

where *stp-name* is a character string 1 to 15 characters long. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen. The *stp-name* cannot be **all**.

**Description** This enables operation of the Spanning Tree Algorithm for the specified **stp** or for the entire switch. If the Spanning Tree Algorithm is to be run on a VLAN, the VLAN must be added to an STP that is enabled. User created STPs are disabled by default. The default STP is disabled on switch start-up.

This command is required before the [disable stp port command on page 9-45](#) and [enable stp port command on page 9-55](#) commands can be used. Once an STP has been enabled by this command it is then possible to enable or disable any port belonging to that STP.

Enabling STP operation on a port may affect the operation of GARP. Each GARP application has a GIP component whose actions depend upon whether the port is in the STP forwarding state.

**Examples** To enable the *company* STP, use the command:

```
ena stp=company
```

To enable all STPs, use the following command:

```
ena stp=all
```

**Related Commands**

- [create stp](#)
- [destroy stp](#)
- [disable stp](#)
- [enable switch stpforward](#)
- [set stp](#)
- [show stp](#)

## enable stp debug

**Syntax** `ENable STP={stp-name|ALL} DEBug={MSG|PKT|STAtE|ALL}  
[OUTput=CONsole] [TIMEOut={1..4000000000|NONE}]`

`ENable STP={stp-name|ALL} DEBug={MSG|PKT|STAtE|ALL}  
Port={port-list|ALL} [OUTput=CONsole]  
[TIMEOut={1..4000000000|NONE}]`

where:

- *stp-name* is a character string 1 to 15 characters long. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen. The *stp-name* cannot be **all**.
- *port-list* is a port number, range (specified as *n-m*), or comma-separated list of numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered switch Ethernet port, including uplink ports.

**Description** This command enables STP debugging for the specified STP or ports. Be aware that enabling debug could flood the receiving Telnet session or asynchronous port with raw data.

The **stp** parameter specifies the STP for which the debugging mode is enabled. If an STP is specified and ports are specified with the **port** parameter, then debug is enabled on the specified port on the specified STP. If an STP is not specified or **all** is specified with the **stp** parameter, and ports are specified with the **port** parameter, then debug mode for the listed ports is enabled on the STPs with the listed port as a member.

The debug status of a port is not changed if the port is moved out of its current STP by one of the following commands:

- [add vlan port](#)
- [delete vlan port](#)
- [add stp vlan](#)
- [delete stp vlan](#)

This command is effective on disabled ports or disabled STPs but produces no debugging information until the ports and the STP are enabled.

The **debug** parameter specifies the debugging modes that are to be enabled. If **all** is specified, all debugging modes for the STP or ports are enabled. The other modes can be enabled independently of each other. The **debug** parameter must be specified before the **port** parameter. The debugging modes enabled by each option are shown in [Table 9-12 on page 9-53](#).

Table 9-12: STP debugging options

Option	Description
MSG	Decoded display of received and transmitted STP packets
PKT	Raw ASCII display of received and transmitted STP packets
STATE	Port state transitions. For RSTP, states for all state machines are displayed as well the current role of the port.
ALL	All debug options

The **output** parameter set to **console** specifies that the debugging information produced is sent to the console. The debugging data is by default sent to the port on which it received the **enable stp debug** command. Use this option if the **enable stp debug** command is used in a script, since a script is not received on a port.

The **timeout** parameter specifies the time in seconds that debugging is enabled on the specified ports. This reduces the risk of the switch and the display being overloaded with too much debugging information. This value overrides previous STP debugging timeout values for these ports, even if they were specified for other debugging modes. If **timeout** is not specified, the time out is the most recent **timeout** value set in an **enable stp debug** command, or **none** if none had been set.

The debug status of a port is not changed if the port is moved out of its current STP by one of the following commands: the **add vlan port**, **delete vlan port**, **add stp vlan**, **delete stp vlan**. This command is effective on disabled ports or disabled STPs, but produces no debugging information until the ports and the STP are enabled.

**Examples** To view state debugging information for the STP named “company” for the next 25 seconds, use the command:

```
ena stp=company deb=state timeo=25
```

To enable all debug modes for all STPs with output to the console and no timeout value, use this command:

```
ena stp=all deb=all out=console
```

To enable the message debug mode on ports 5 to 8 indefinitely, use the command:

```
ena stp deb=msg po=5-8 timeo=none
```

#### Related Commands

[disable stp debug](#)

[show stp debug](#)

[disable debug active](#) in Chapter 4, Configuring and Monitoring the System

[show debug active](#) in Chapter 4, Configuring and Monitoring the System

---

## enable stp port

---

**Syntax**    ENable STP[={*stp-name*|ALL}] Port={*port-list*|ALL}

where:

- *stp-name* is a character string 1 to 15 characters long. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen.
- *port-list* is a port number, range (specified as *n-m*), or comma-separated list of numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered switch Ethernet port, including uplink ports.

**Description**    This command enables the Spanning Tree algorithm on one or more ports, for one or all STPs. This command is effective only when the Spanning Tree algorithm is enabled for the STP to which the ports belong.

The **stp** parameter specifies the STP to have ports enabled. The default is **all**.

This command enables the Spanning Tree algorithm on specific ports or all ports on the switch. This command is effective only when the Spanning Tree algorithm is enabled for the STP to which the port belongs.

Enabling an STP port may cause reconfiguration of the Spanning Tree to which the port belongs because STP messages (BPDUs) are generated on the port.

Enabling STP operation on a port may affect the operation of GARP. Each GARP application has a GIP component whose actions depend upon whether the port is in the STP forwarding state.

The **disable stp** command overrides the results of the **disable stp port** and **enable stp port** commands. Once a STP has been disabled by this command it is not possible to enable or disable any port belonging to that STP. The STP must be enabled first before any port belonging to that STP can be enabled or disabled.

**Examples**    To enable the Spanning Tree Algorithm to use port 4, use the command:

```
ena atp po=4
```

To enable STP on all ports, use the command:

```
ena stp po=all
```

To enable STP on just the administration network and only on port 4, use the command:

```
ena stp=admin po=4
```

**Related Commands**    [disable stp port](#)  
                          [set stp port](#)  
                          [show stp port](#)

## enable switch stpforward

---

**Syntax**    ENAbLe SWItch STPForward

**Description**    This command enables the switch to (transparently) forward spanning tree protocol BPDUs (and rapid spanning tree protocol BPDUs). STP forwarding can only be enabled when no STP instance is enabled. Forwarding is disabled by default.

**Example**    To enable switch STP forwarding, use the command:

```
ena swi stpf
```

**Related Commands**    [disable stp port](#)  
[disable switch stpforward](#)  
[disable stp](#)  
[enable stp](#)



---

## purge mstp

---

**Syntax** PURge MSTp

**Description** This command returns MSTP to its status when the switch was first powered on. It purges all configuration information relating to MSTP. All user-created MSTIs are destroyed, and all VLANs are mapped to the CIST. It restores default values to all the configurable parameters.



---

**Caution** Use with extreme caution because all current configurations will be lost.

---

**Example** To purge the MSTP configuration, use the command:

```
purge mst
```

**Related Commands** [show mstp](#)  
[show mstp msti](#)

---

## purge stp

---

**Syntax** PURge STP

**Description** The debug parameters for all ports are reset to their defaults. It returns STP to its status when it was first powered on. This command destroys all user-created STPs, and restores the defaults to all the configurable parameters (**forwarddelay**, **hellotime**, **maxage** and **priority**) in the remaining default STP.

**Example** To purge all STPs, use the command:

```
purge stp
```

**Related Commands** [reset stp](#)  
[set stp](#)  
[set stp port](#)  
[show stp](#)  
[show stp counter](#)

---

## reset mstp counter port

---

**Syntax** RESET MSTp COUNTER Port={*port-list*|ALL}

where *port-list* is a port number, range (specified as *n-m*), or comma-separated list of port numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered Ethernet switch port, including uplink ports.

**Description** This command resets the counter value for a specified port or ports.

The **port** parameter specifies the ports. If **all** is specified, all port counters in the switch are reset. The default value is **all**.

**Example** To enable all ports in MSTI5, use the command:

```
reset mstp cou po=1
```

**Related Commands** [show mstp](#)  
[show mstp cist port](#)  
[show mstp msti port](#)

---

## reset stp

---

**Syntax** RESET STP={*stp-name*|ALL}

where *stp-name* is a character string 1 to 15 characters long. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen. The *stp-name* cannot be **all**.

**Description** This command resets operation of the Spanning Tree Algorithm for the specified STP, initialises all counters for the specified STP, and initialises all timers on all ports that are members of the STP. Ports remain in the state they were before the reset command was issued, for example, ports that were enabled remain enabled, ports that were disabled remain disabled.

**Example** To reset the *company* STP, use the command:

```
reset stp=company
```

**Related Commands** [purge stp](#)  
[set stp](#)  
[show stp](#)  
[show stp counter](#)  
[show switch port](#) in Chapter 8, Switching

---

## set mstp

---

**Syntax** SET MSTp [CONFigname=*name*] [REVIionlevel=*level*]  
[MAXHops=1..40] [MAXage=6..40] [HEllotime=1..10]  
[FORwarddelay=4..30] [PROToColversion={STP|RSTp|MSTp}]  
[STATIcVlans={YES|NO|ON|OFF|TRUE|FALSE}]

where:

- *name* is the MST configuration name. It is a string of up to 32 characters. Valid characters are uppercase and lowercase letters, digits, and the underscore. No other character types are allowed.
- *level* is the MST configuration revision level, having the range 0-65535.

**Description** This command sets the MST configuration identifier values and the state machine performance parameters. The configuration identifier contains:

- the configuration name
- the revision number
- a digest of the VLAN to MSTI configuration table

The state machine performance parameters are constants used by the CIST and MSTI state machines.

When the MST algorithm calculates the active topology, it does not consider the VLAN membership of the ports because the IEEE Standard 802.1Q-2003 assumes that the active topology is determined first, and the VLANs are then configured dynamically over the active topology via GVRP. GVRP configures the VLAN memberships of ports so that frames belonging to a VLAN are able to traverse the spanning tree (CIST or MSTI) to which the VLAN is assigned.

The process is reversed when statically configured VLANs are used. The VLAN memberships of ports are configured statically and then the active topology is calculated. However, the MST algorithm does not consider the VLAN memberships when calculating the active topology; it may choose a port that is not a member of any of the VLANs assigned to the spanning tree to be the root port, even though an alternate port that is a member of the VLANs may exist. This would partition the network, preventing frames belonging to a VLAN assigned to the spanning tree from traversing the network. In this situation it is desirable that the algorithm considers the VLAN memberships of ports and prevents partitioning where possible. It should choose the root port from the ports that are members of the VLANs assigned to the spanning tree.

When using statically configured VLANs, each VLAN assigned to a given spanning tree should have the same port membership; otherwise, partitioning may occur.

The MST configuration identifier determines which MST region a switch belongs to. The MST configuration identifier is conveyed in the MSTP BPDUs, so the switch can check whether it is allocating VIDs to the same spanning tree instance as a neighbouring switch. If the configuration identification of two switches matches they are from the same MST region.

MSTP assigns the switch a default MST configuration identification consisting of a unique default configuration name and a default revision level.

The **configname** parameter specifies the name for the MST region. All the switches in the same MST region have the same configuration name. If the configure name is not set explicitly by the command, the default name for the MST region is the switch's MAC address presented as a text string. All switches are in their own MST region by default because MAC addresses are unique.

The **revisionlevel** parameter specifies the revision level in the MST region. All the switches in the same MST region have the same revision number. If the revision level is not set explicitly by the command, the default revision level is 0.

The **forwarddelay** parameter sets a delay time, in seconds, that a port waits before changing its spanning tree state towards the forwarding state. Its purpose is to allow sufficient time for other ports to receive their spanning tree information. The delay determines the maximum time taken to transition from discarding to learning and from learning to forwarding. This value is only used when the switch is acting as the root bridge. Any switch not acting as the root bridge uses a dynamic value for the **forwarddelay** set by the root bridge. The **forwarddelay**, **maxage**, and **hellotime** parameters are interrelated. See the formulae below. The default for **forwarddelay** is 15 seconds.

The **hellotime** parameter sets the time period, in seconds, between the transmissions of spanning tree configuration messages. These messages are transmitted by ports with the 'designated port' role of the spanning tree, or are trying to become the root bridge. The default is 2 seconds.

The **maxage** parameter sets the maximum age, in seconds, that dynamic MSTP configuration information stored in the switch may reach before it is discarded. The default is 20 seconds.

The **forwarddelay**, **maxage**, and **hellotime** parameters should be set according to the following formulae, as specified in IEEE Standard 802.1d:

$$2 \times (\text{forwarddelay} - 1.0 \text{ seconds}) \geq \text{maxage}$$

$$\text{maxage} \geq 2 \times (\text{hellotime} + 1.0 \text{ seconds})$$

The **maxhop** parameter specifies the maximum hop count in transmitting information within an MST region. This is in order to ensure that old information does not endlessly circulate through redundant paths in the network, thus preventing the effective propagation of the new information. The hop count is decremented by each receiving port. Received information is discarded and the port is made a designated port if the hop count reaches 0. The default value for **maxhop** is 20.

The **protocolversion** parameter specifies which version of the spanning tree protocol the switch uses. If **mstp** is specified, the switch uses the full Multiple Spanning Tree protocol and sends MSTI BPDUs. If RSTP is specified, the switch uses the Rapid Spanning Tree protocol and sends RST BPDUs. The switch operates as though it is in a region of its own. If STP is specified, the switch emulates the Spanning Tree Protocol and transmits STP configuration BPDUs. Rapid port state transitions are disabled, and the switch operates as if in a region of its own.

The **staticvlans** parameter should be turned on when the ports that link to other switches have static VLAN memberships. In simple static VLAN configurations it may be possible to operate with this option turned off provided that redundant links between a pair of switches have the same VLAN memberships. If VLANs are being configured dynamically with GVRP, the **staticvlans** parameter should be **off**. The default is **off**.

**Example** To set MST configuration name to mstRegion1 and the revision level to 10, use the command:

```
set mst conf=mstregion1 revi=10
```

To set forward delay time to 20 seconds and max hop count to be 25, use the command:

```
set mst fo=20 maxh=25
```

To turn **staticvlans** on, use the command:

```
set mst stat=true
```

To set hello time to be 2 seconds and max message age to be 30 seconds, use the commands:

```
set mst he=2 ma=30
```

**Related Commands** [show mstp](#)

---

## set mstp cist

---

**Syntax** SET MSTp CIST [PRIOrity=0..65535]

**Description** This command sets parameters used by the MSTP algorithm to calculate the Common and Internal Spanning Tree (CIST). The bridge level parameters of the CIST can be modified to force the spanning tree configuration or to tune its topology.

The **priority** parameter sets the priority of the switch to become the root bridge in the CIST. The lower the value, the better the bridge identifier and the more likely it will be selected as the root. Although any value between 0 and 65,535 can be specified, the protocol requires the priority to be multiples of 4096; therefore, values are rounded down to the nearest multiple of 4096 (see the **create mstp cist** command for rounding scheme). The default is 32768.

**Example** To set CIST to a priority of 8192, use the command:

```
set mst cist prio=8192
```

**Related Commands** [create mstp msti](#)  
[show mstp](#)  
[show mstp cist](#)

## set mstp cist port

**Syntax** SET MSTp CIST Port={*port-list*|ALL} [PRIOrity=0..255]  
 [EXTPathcost=*extPathCost*] [INTPathcost=*intPathCost*]  
 [EDGEport={YES|NO|ON|OFF|True|False}]  
 [POINTtopoint={YES|NO|ON|OFF|True|False|Auto}]

where:

- *port-list* is a port number, range (specified as *n-m*), or comma-separated list of port numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered Ethernet switch port, including uplink ports.
- *extPathcost* is a value in the range 1 to 200,000,000
- *intPathcost* is a value from 1 to 200,000,000

**Description** This command sets the common internal spanning tree (CIST) tuning parameters for the specified ports. Modifying parameters for a switch port forces a recalculation of the CIST port rules.

The parameters assigned for the specified ports affect the network topology of only the CIST. They do not affect the topology of other spanning tree instances on the switch.

The **port** parameter specifies a list of ports to be configured for the CIST. If **all** is specified, then all of the ports are configured according to the new parameters for the CIST.

The **priority** parameter sets the value of the priority field contained in the port identifier. The MST algorithm uses the port priority when determining the root port for the switch in the CIST. The port with the lowest value is considered to have the highest priority and will be chosen as root port over a port - equivalent in all other aspects - but with a higher priority value. Any value in the range 0 to 255 can be entered, but the switch rounds the entered value down to the nearest multiple of 16 (for example, if 17 is entered, 16 is used). See the **priority** parameter of the [enable mstp msti port command on page 9-50](#) for a table of rounded values. The default value is 128.

The **extpathcost** parameter sets the external path cost for the ports. This parameter specifies a port's contribution to the cost of a path to the region containing the CIST root via that port. It applies when the port is a root port.

The **intpathcost** parameter sets the internal path cost for the ports. This parameter specifies a port's contribution to the cost of a path to the CIST regional root via that port. It applies when the port is a root port.

If the **extpathcost** or the **intpathcost** is not explicitly set by the user, or the default values have been restored to the port, then the default for the port varies as the speed of the port varies, See [Table 9-13 on page 9-66](#). However, deleting an existing **extpathcost** or **intpathcost** value does not re-apply the "no value" condition. To do this, enter the word "default."

The **edgeport** parameter specifies whether or not the port is an edge port. An edge port is a one that attaches to a LAN that is known to have no other bridges attached. If **no** is specified, then the port is not considered to be an edge port. The values **no**, **off** and **false** are equivalent. If **yes** is specified, then the port is considered to be an edge port. The values **yes**, **on**, and **true** are

equivalent. If **edgeport** is set to **yes** and an MSTP BPDU is received on the port, indicating that another bridge is connected to the LAN, then the port is no longer treated as an edge port. The default is **no**. Edge ports are permitted to make rapid transitions to the forwarding state, because they are known not to be connected to another bridge and therefore cannot form part of a network loop. Edge ports that are not configured as such must make slow transitions to the forwarding state. For optimal convergence all edge ports should be identified and have **edgeport** set to **yes**. A port should be set to edge port only when it connects to a single end station.

The **pointtopoint** parameter specifies whether or not the port has a point-to-point connection to another bridge. If **auto** is specified, then the status of point-to-point link is determined automatically by the switch. If **yes** is specified, then the port is treated as a point-to-point LAN segment. The values **yes**, **on** and **true** are equivalent. If **no** is specified, then the port is not treated as a point-to-point LAN segment. The values **no**, **off** and **false** are equivalent. If the port is considered as a point-to-point port, then it is permitted to make rapid transitions to the forwarding state, providing it receives an agreement message from the bridge at the other end of the segment. A port should be set to point-to-point only when it connects to exactly one other bridge. The default is **auto**.

**Example** To set port a priority of 16 for port 2 in the CIST, use the command:

```
set mstp cist po=2 priority=16
```

To set an external port path cost of 120 for port 2 in the CIST, use the command:

```
set mstp cist po=2 extpathcost=120
```

To set an internal port path cost of 200 for port 2 in the CIST, use the command:

```
set mstp cist po=2 intpathcost=200
```

To set port 2 in the CIST as an edge port, use the command:

```
set mstp cist po=2 edge=yes
```

To set port 2 in the CIST as a point to point link, use the command:

```
set mstp cist po=2 pointtopoint=yes
```

**Related Commands** [show mstp cist](#)  
[show mstp cist port](#)

## set mstp msti

---

**Syntax** SET MSTp MSTI=*instance* [PRIOrity=0..65535]

where *instance* is the instance number of a specific MSTI in a range from 1 to 4094.

**Description** This command sets parameters used by the Multiple Spanning Tree algorithm to calculate the spanning tree for a specified MSTI. The bridge level parameters of the MSTI can be modified in order to tune the spanning tree topology.

The **msti** parameter specifies the instance number for the specified Multiple Spanning Tree Instance.

The **priority** parameter sets the priority of the switch to become the Root Bridge in the specified MSTI. The lower the value of the bridge priority, the more likely it is to be selected as the root bridge. Although any value between 0 and 4096 can be specified, the switch processes only values that are multiples of 4096. Therefore, any value entered is rounded down to its nearest multiple of 4096, see [Table 9-9 on page 9-31](#). The default value for **priority** is 32768.

**Example** To set the **priority** to 8192 to MSTI5, use the command:

```
set mstp msti=5 priority=8192
```

**Related Commands** [show mstp](#)  
[show mstp msti](#)



## set mstp msti port

**Syntax** SET MSTp MSTI=*instance* Port={*port-list*|ALL}  
[PRIOrity=0..255] [PAthcost=*pathcost*]

where:

- *instance* is the instance number of the MSTI, from 1 to 4094.
- *port-list* is a port number, range (specified as *n-m*), or comma-separated list of port numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered Ethernet switch port, including uplink ports.
- *pathcost* is a value from 1 to 200,000,000.

**Description** This command sets tuning parameters for the specified ports or all ports for the specified multiple spanning tree instance (MSTI). Modifying parameters for a port forces a recalculation of the port roles for the specified **msti**.

The parameters assigned for the specified ports only affects the network topology of the specified **msti**, not any other spanning tree instances on the switch.

The **msti** parameter specifies the instance number for the selected **msti**.

The **port** parameter specifies a list of ports to be configured for the specified **msti**. If **all** is specified, all of the ports are configured according to the new parameter values for the specified **msti**.

The **priority** parameter sets the value of the priority field contained in the port identifier. The MST algorithm uses the port priority when determining the root port for the switch in the specified **msti**. The port with the lowest value is considered to have the highest priority and is chosen as root port over a port - equivalent in all other aspects - but with a higher priority value. Any value in the range 0 to 255 can be entered, but the switch rounds the entered value down to the nearest multiple of 16 (for example, if 17 is entered, 16 is used), as shown in the following table. The default value is 128.

Lower Boundary	Upper Boundary	Rounded Port Priority Value
0	0	0
16	31	16
32	47	32
48	63	48
64	79	64
80	95	80
96	127	96
128	143	128
144	159	144
160	175	160
176	191	176
192	207	192
208	223	208
224	239	224
240	254	240

The **pathcost** parameter sets the internal path cost for the each port. This parameter specifies a port's contribution to the cost of a path to the MSTI regional root via that port. It applies when the port is a root port. The **pathcost** for a LAN port should be set in the range of 1 to 200000000. The default **pathcost** values and the range of recommended **pathcost** values depend on the port speed.

If the **pathcost** of a port has not been explicitly set by the user, or the default values have been restored to the port, then the default **pathcost** for the port varies as the speed of the port varies. However, deleting an existing **pathcost** value does not reapply the "no value" condition. To reapply the "no value" condition, enter the word "default".

Table 9-13: Path cost values and port speed

Port Speed	Default pathcost	Recommended pathcost range
Less than 100 Kb/s	200,000,000	20,000,000-200,000,000
1Mbps	20,000,000	2,000,000-20,000,000
10Mbps	2,000,000	200,000-2,000,000
100 Mbps	200,000	20,000-200,000
1 Gbps	20,000	2,000-20,000
10 Gbps	2,000	200-2,000
100 Gbps	200	20-200
1Tbps	20	2-200
10 Tbps	2	2-20

**Example** To set port priority of 120 for port 2 in MSTI5, use the command:

```
set mstp msti=5 po=2 priority=120
```

To set port path cost of 200 for port 2 in MSTI5, use the command:

```
set mstp msti=5 po=2 pathcost=120
```

**Related Commands** [show mstp msti](#)  
[show mstp msti port](#)

## set stp

---

**Syntax** SET STP={*stp-name*|ALL} [Forwarddelay=4..30]  
 [Hellotime=1..10] [Maxage=6..40] [MODE={STANDARD|  
 RAPID}] [PRIOrity=0..65535] [RSTPtype={NORMAL|  
 STPCompatible}]

SET STP={*stp-name*|ALL} DEFault

where *stp-name* is a character string 1 to 15 characters long. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen. The *stp-name* cannot be **all**.

**Description** This command sets parameters used by the Spanning Tree Algorithm for the specified STP. If **all** is specified, then parameters for all STPs on the switch are set. When **all** is specified and the command succeeds on a subset of STPs but causes errors on the others, then the command as a whole fails and has no effect. Each STP has its own independent **forwarddelay**, **hellotime**, **maxage**, and **priority** parameters.

The **default** parameter sets the **forwarddelay**, **hellotime**, **maxage** and **priority** parameters back to their defaults. This parameter cannot be specified with either of the **forwarddelay**, **hellotime**, **maxage** or **priority** parameters.

The **forwarddelay** parameter sets the time in seconds to control how fast a port changes its spanning tree state when moving towards the forwarding state. If the mode is set to standard, the value determines how long the port stays in each of the listening and learning states which precede the forwarding state. If the mode is set to rapid, this value determines the maximum time taken to transition from discarding to learning and from learning to forwarding. This value is used only when the switch is acting as the root bridge. Switches not acting as the Root Bridge use a dynamic value for the **forwarddelay** set by the root bridge. The **forwarddelay**, **maxage**, and **hellotime** parameters are interrelated. See the formulas below. The default for **forwarddelay** is 15 seconds.

The **hellotime** parameter sets the time in seconds between the transmission of switch spanning tree configuration information when the switch is the Root Bridge of the spanning tree or is trying to become the Root Bridge. The default is 2 seconds.

The **maxage** parameter sets the maximum time in seconds that dynamic STP configuration information is stored in the switch before it is discarded. The default is 20 seconds.

The **forwarddelay**, **maxage**, and **hellotime** parameters should be set according to the following formulae, as specified in IEEE Standard 802.1d:

$$2 \times (\text{forwarddelay} - 1.0 \text{ seconds}) \geq \text{maxage}$$

$$\text{maxage} \geq 2 \times (\text{hellotime} + 1.0 \text{ seconds})$$

The **mode** parameter specifies whether the STP operates in standard or rapid mode. In standard mode, the Spanning Tree Algorithm is run. In rapid mode, the Rapid Spanning Tree Algorithm is run. The default is **standard**. If the mode is changed while the algorithm is running, the STP is reinitialised.

If the **mode** parameter has been set to **rapid**, values specified for the **priority** parameter must be multiples of 4096. If a value is specified that is not a

multiple of 4096, the value is rounded down to the nearest multiple of 4096. The rounding scheme is defined in [Table 9-14](#).

Table 9-14: Rounding scheme for ranges of **priority** parameter values when the **mode** parameter is set to **rapid**

Lower boundary	Upper boundary	Rounded RSTP Bridge Priority Value
0	4095	0
4096	8191	4096
8192	12287	8192
12288	16383	12288
16384	20479	16384
20480	24575	20480
24576	28671	24576
28672	32767	28672
32768	36863	32768
36864	40959	36864
40960	45055	40960
45056	49151	45056
49152	53247	49152
53248	57343	53248
57344	61439	57344
61440	65535	61440

The **priority** parameter sets the priority of the switch to become the Root Bridge. The lower the value of the Bridge Identifier, the higher the priority. If the **priority** parameter is set by specifying the **priority** or **default** parameters, the specified STP is initialised. Counters for the STP are not affected. The default for **priority** is 32768.

The **rstptype** parameter specifies how the RSTP algorithm operates. If **normal** is specified, then the algorithm uses rapid port role transitions and transmits and receives RST BPDUs. If **stpcompatible** is specified, then rapid transitions are disabled, standard BPDUs are transmitted and RST BPDUs are discarded. Setting **rstptype** to **stpcompatible** allows RSTP to support applications and protocols that may be sensitive to frame duplication and misordering, for example NetBeui. The default is **normal**.

Setting **rstptype** to **normal** when normal has already been set, sets all ports to the “sending RSTP” state. This is referred to in the IEEE Standard 802.1w standard as *mCheck* and is useful for restoring full rapid mode operation when one or more ports on the switch has entered the “sending STP” state. RSTP capable devices with **rstp** set to **normal** that receive the RST BPDUs enter the “sending RSTP” state. When an STP BPDU is received after the mCheck operation, either as a result of a device being in rapid mode with **rstptype** set to **stpcompatible** or as a result of a device in standard mode, the ports that received the STP BPDUs revert to the “sending STP” state. The default is **normal**. This parameter is available to STPs that are running with the **mode** parameter set to **rapid**.

**Examples** To set the forward delay to 22 seconds for the *company* STP, use the command:

```
set stp=company fo=22
```

To set the hello time to 3 seconds for the *company* STP, use the command:

```
set stp=company he=3
```

To set the maximum age to 19 seconds for the *company* STP, use the command:

```
set stp=company ma=19
```

To set the priority of the switch becoming the Root Bridge to 100 for the *company* STP, use the command:

```
set stp=company prio=100
```

To set the Forward Delay to 12 seconds for all STPs, assuming the **forwarddelay-maxage** criterion is met for all STPs, use the command:

```
set stp=all fo=12
```

To set the parameters for the *company* STP to their defaults, use the command:

```
set stp=company def
```

**Related Commands**

- [purge stp](#)
- [reset stp](#)
- [set stp port](#)
- [show stp](#)

## set stp port

**Syntax** SET STP[={*stp-name*|ALL}] Port={*port-list*|ALL}  
 [Pathcost=*pathcost*] [PORTPRiority=0..255]  
 [EDGEport={YES|NO|ON|OFF|True|False}] [PTP={Auto|ON|  
 OFF|YES|NO|True|False}]

SET STP[={*stp-name*|ALL}] Port={*port-list*|ALL} DEFault

where:

- *stp-name* is a character string 1 to 15 characters long. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen.
- *port-list* is a port number, range (specified as *n-m*), or comma-separated list of numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered switch Ethernet port, including uplink ports.
- *pathcost* is a value from 1 to 1,000,000 if STP is running in standard mode, and 1 to 200,000,000 if STP is running in rapid mode.

**Description** This command sets various parameters used by the Spanning Tree Algorithm for the specified ports, or all ports within the specified STP, or all STPs.

A port can belong to multiple STPs when the port is a member of more than one VLAN.

The STP parameter specifies an STP name. If no parameter is entered, the default is **all**.

Non-default STP parameters configured for a port are not retained when the VLAN that the port belongs to is moved to another STP using the [add stp vlan command on page 9-30](#) or [delete stp vlan command on page 9-35](#).

The **port** parameter specifies either a list of ports that belong to the specified STP, or **all** ports. A port can belong to multiple STPs if the port is a member of more than one VLAN. Where the **port** parameter specifies a list of ports belonging to a trunk group, the list must include “all” the ports belonging to the trunk group.

The **default** parameter sets the **pathcost** and **portpriority** parameters back to their defaults. This parameter cannot be specified with either of the **pathcost** and **portpriority** parameters. The **edgeport** and **ptp** parameters are not affected by this command.

The **pathcost** parameter sets a path cost value for the selected bridge port. This value then combines with others along the path to the root bridge in order to determine a total path cost value from the particular port to the root bridge. The lower the numeric value, the higher the priority of the path.

Table 9-15: Path cost values and port speed for standard mode

Port Speed	Default pathcost	Recommended pathcost range
10Mbps	100	50 - 600
100Mbps	19	10 -60
1Gbps	4	3 -10

Table 9-16: Path cost values and port speed for rapid mode

Port Speed	Default pathcost	Recommended pathcost range
Less than 100 Kb/s	200000000	20000000-200000000
1Mbps	20000000	2000000-20000000
10Mbps	2000000	200000-2000000
100 Mbps	200000	20000-200000
1 Gbps	20000	2000-20000
10 Gbps	2000	200-2000
100 Gbps	200	20-200
1Tbps	20	2-200
10 Tbps	2	2-20

When the STP mode is changed from standard to rapid, or rapid to standard, the **pathcost** parameter is mapped from one range to the other based on relative deviation from the nearest default. We recommend that the **pathcost** values be checked when changing mode to confirm that they are appropriate for the network configuration.

If the **pathcost** of a port has not been explicitly set by the user or the defaults have been restored to the port, then the default **pathcost** for the port varies as the speed of the port varies.

IEEE Standard 802.1d, limited the path cost parameter values to a range of 1-65000. The recommended values for various data rates based on this standard is shown in [Table 9-15 on page 9-70](#). A later version of the standard IEEE 802.1w, extended the range to 1-(4x10<sup>9</sup>) and revised the recommended values, see [Table 9-16 on page 9-71](#). This version of the standard recommends calculating each path cost by applying the formula,  $\text{pathcost} = 200000000000 / (\text{Link Speed in KB/s})$ . Both sets of path cost values allow for a network to have a maximum of 20 concatenated hops.

In LANs where the recommended values defined in IEEE Standard 802.1d and IEEE Standard 802.1w are required to interwork, one set of path cost values must be reconfigured so that both sets of bridges work to the same network metrics.

The **portpriority** parameter sets the value of the priority field contained in the port identifier. The Spanning Tree Algorithm uses the port priority when determining the root port for each switch. The port with the lowest value is considered to have the highest priority. The default is 128. Each STP has its own independent **portpriority** parameter for each member port.

If the STP mode is rapid, then the values specified for the **portpriority** parameter must be multiples of 16. If a user specifies a value that is not a multiple of 16, it is rounded down to the nearest multiple of 16. The rounding scheme is identified in [Table 9-17](#).

Table 9-17: Rounding scheme for **portpriority** value when the mode is rapid

Lower boundary	Upper boundary	Rounded Value
0	15	0
16	31	16
32	47	32
48	63	48
64	69	64
80	95	80
96	111	96
112	127	112
128	143	128
144	159	144
160	175	160
176	191	176
192	207	192
208	223	208
224	239	224
240	255	240

The **edgeport** parameter specifies whether the port is an edge port. An edge port is a port that attaches to a LAN that is known to have no other bridges attached. If **no** is specified, then the port is not considered to be an edge port. The values **no**, **off**, and **false** are equivalent. If **yes** is specified, then the port is considered to be an edgeport. The values **yes**, **on**, and **true** are equivalent. If **edgeport** is set to **yes** and an RST BPDU is received on the port, which indicates that another bridge is connected to the LAN, then the port is no longer treated as an edge port. The default is **no**. If STP is running in rapid mode, then the rapid transition of a port to the forwarding state depends on the port being considered an edgeport or part of a Point-to-Point link.

The **ptp** parameter specifies whether the port has a point-to-point connection with another bridge. If **auto** is specified, then the point-to-point status of the port is determined automatically by the switch. If **yes** is specified, then the port is treated as a point-to-point LAN segment. The values **yes**, **on**, and **true** are equivalent. If **no** is specified, then the port is not treated as a point-to-point LAN segment. The values **no**, **off**, and **false** are equivalent. If STP is running in rapid mode, then the rapid transition of a port to the forwarding state depends on the port being considered an edgeport or part of a Point-to-Point link. The default is **auto**.

**Examples** To set a port priority of 42 for port 10 in STP1, use the command:

```
set stp=1 po=10 portpri=42
```

To set a path cost of 120 for all ports on all STPs, use the command:

```
set stp=all po=all pa=120
```

To set the port parameters for ports 1 to 10 in STP3 to their standard defaults, use the command:

```
set stp=3 po=1-10 def
```



To set port 10 in STP3 as an edge, use the command:

```
set stp=3 po=10 edge=yes
```

To force port 10 in STP3 to be treated as if it were part of a point to point LAN segment, use the command:

```
set stp=3 po=10 ptp=yes
```

**Related Commands**

- [purge stp](#)
- [reset stp](#)
- [set stp](#)
- [show stp](#)

## show mstp

**Syntax** SHow MSTp [CONfigid] [TAbLe]

**Description** This command displays information about MSTP (Figure 9-9, Table 9-18).

If the **configid** parameter is specified, the MST Configuration Identification is displayed (Figure 9-10 on page 9-75, Table 9-19 on page 9-75).

If the **table** parameter is specified, the MST Configuration Table that contains the map between MSTIs and VLANs is displayed (Figure 9-11 on page 9-75, Table 9-20 on page 9-75).

Figure 9-9: Example output from the **show mstp** command

```

MSTP Information
-----
MSTP Status .....Enabled
MST Configuration Name ..... mstRegion1
MST Revision Level ..... 0
Number of MSTIs ..... 10
Hello Time ..... 2
Forward Delay ..... 15
Max Message Age ..... 100
Max Hops ..... 5
Protocol Version ..... MSTP
Support Static VLANs ..... Enabled
Transmission Limit ..... 3
Migrate Time ..... 8
-----

```

Table 9-18: Parameters in output of the **show mstp** command

Parameter	Description
MSTP Status	Whether MSTP is enabled.
MST Configuration Name	Name of the MST region.
MST Revision Level	Revision level of the MST region.
Number of MSTIs	Number of Multiple Spanning Tree instances.
Protocol Version	Spanning Tree Protocol version: STP, RSTP, or MSTP.
Max Hops	Maximum hop count in transmitting information within an MST region.
Transmission Limit	Number of bridge protocol messages (BPDUs) that may be transmitted in the interval specified by Hello Time.
Migrate Time	A constant timer value used as the initial value of the migration delay. The value of Migrate Time is 3 seconds.
Hello Time	The seconds between transmissions of spanning tree configuration information (BPDUs).
Forward Delay	Number of seconds that controls how fast a port changes its spanning tree state when moving towards the forwarding state.
Max Message Age	Maximum age of received bridge protocol message (BPDU) information before it is discarded.

Table 9-18: Parameters in output of the **show mstp** command (cont.)

Parameter	Description
Support Static VLAN	Whether a supporting static VLAN configuration is enabled.

Figure 9-10: Example output from the **show mstp configid** command

```

MST Configuration Identification
-----
Configuration Name ..... mstRegion1
Format Selector ..... 0
Revision Level ..... 12
Configuration Digest ..... AC36177F50283CD4B83821D8AB26D8AB
-----

```

Table 9-19: Parameters in output of the **show mstp configid** command

Parameter	Description
Configuration Name	Name of the MST region.
MST Configuration Name	Configuration Identifier Format Selector.
MST Revision Level	Revision level of the MST region.
Configuration digest	16-octet signature of type HMAC-MID5 created from the MST Configuration Table.

Figure 9-11: Example output from the **show mstp table** command

```

MST Configuration Table
-----
Multiple Spanning Tree Instance      VLAN Members
-----
CIST                                15-19, 31-4094
MSTI 1                             1, 2, 10, 20-30
MSTI 2                             3-9
MSTI 3                             11-14
-----

```

Table 9-20: Parameters in output of the **show mstp table** command

Option	Description
Multiple Spanning Tree Instance	Whether the instance of a spanning tree is CIST or an MSTI.
VLAN Members	List of the VLANs that are mapped to a specific MSTI.

**Example** To show information about MSTP, use the command:

```
sh mst
```

**Related Commands**

- enable mstp
- disable mstp
- create stp
- destroy mstp msti
- add mstp msti vlan
- delete mstp msti vlan
- set mstp
- set mstp cist
- set mstp msti

## show mstp cist

**Syntax** SHOW MSTp CIST

**Description** This command displays the information about the Common Internal Spanning Tree (Figure 9-12, Table 9-21).

Figure 9-12: Example output from the **show mstp cist** command

```
Common Internal Spanning Tree
-----
Bridge Identifier.....32768 : 00-00-cd-05-19-28
Bridge Role.....Root Bridge
VLAN Members.....1, 2-10, 20
CIST Root Bridge.....32768 : 00-00-cd-05-19-28
CIST Regional Root Bridge.....32768 : 00-00-cd-05-19-28
Designated Bridge.....32768 : 00-00-cd-05-19-28
Root Port.....N/A
External Root Path Cost.....0
Internal Root Path Cost.....0
Performance:
  Max Age.....20
  Hello Time.....2
  Forward Delay.....20
  Max Hops.....5
  Bridge Max Age.....20
  Bridge Hello Time.....20
  Bridge Forward Delay.....20
  Bridge Max Hops.....20
  Transmission Limit.....3
Topology Changes:
  Time Since Topology Change.....100
  Topology Change Count.....3
  Topology Change.....FALSE
-----
```

Table 9-21: Parameters in output of the **show mstp cist** command

Parameter	Meaning
Bridge Identifier	The unique bridge identifier of the switch. This parameter consists of two parts, one part is derived from the switch's unique MAC Address, and the other part is the priority value entered for the switch.
Bridge Role	The role of the bridge in the CIST. This can be either, the root bridge regional root bridge or designated bridge.
VLAN Members	A list of the VLANs that are mapped to the CIST.
CIST Root Bridge	The bridge identifier of the CIST Root of the bridged local area network.
CIST Regional Root Bridge	The bridge identifier of the root bridge for the CIST in an MST region (MSTR).
Designated Bridge	The bridge identifier of the bridge through which the root bridge may be reached from this device.
Root Port	The port number of the root port for the switch. This parameter is not valid if the switch is the root bridge. In this situation the output is shown as N/A.

Table 9-21: Parameters in output of the **show mstp cist** command (cont.)

Parameter	Meaning
External Root Path Cost	The path cost to the region containing the CIST root from this region.
Internal Root Path Cost	The path cost to the CIST Regional Root.
Max Age	The maximum age of received bridge protocol message (BPDU) information before it is discarded.
Hello Time	The time, in seconds, between transmissions of spanning tree configuration information (BPDUs)
Forward Delay	The maximum time taken to transition from the discarding state to the learning state, and from the learning state to the forwarding state.
Max Hops	Specifies the maximum hop count within an MST region for CIST information transmitted from this switch.
Bridge Max Age	The value of the Max Age parameter when the switch is either the root or is attempting to become the root. This parameter is set by the <b>maxage</b> parameter in the <b>set mstp</b> command.
Bridge Hello Time	The value of the Hello Time parameter when the switch is the root or is attempting to become the root. This parameter is set by the <b>hellotime</b> parameter in the <b>set mstp</b> command.
Bridge Forward Delay	The value of the Forward Delay parameter when this switch is the root or is attempting to become the root. This parameter is set by the <b>forwarddelay</b> parameter in the <b>set mstp</b> command.
Bridge Max Hops	The value of the Max Hops parameter when the switch is either the root or is attempting to become the root. This parameter is set by the <b>maxhops</b> parameter in the <b>set mstp</b> command.
Transmission Limit	The number of BPDUs that may be transmitted in the interval specified by the <b>hellotime</b> parameter. The value of this fixed parameter is 3.
Time Since Topology Change	The count in seconds of the time elapsed since the last topology changed.
Topology Change Count	The number of times the topology has changed since the bridge was powered or initialised.
Topology Change	Indicates whether the topology is in the middle of changing.

**Example** To display the current CIST information, use the command:

```
show mstp cist
```

**Related Commands**

- [disable mstp](#)
- [enable mstp](#)
- [set mstp cist](#)
- [set mstp cist port](#)
- [enable mstp cist port](#)
- [disable mstp cist port](#)

## show mstp cist port

**Syntax** `SHoW MSTp CIST Port [= {port-list | ALL}]`

where *port-list* is a port number, range (specified as *n-m*), or comma-separated list of port numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered Ethernet switch port, including uplink ports.

**Description** This command displays port information about the common internal spanning tree (CIST) (Figure 9-13, Table 9-22 and Figure 9-14 on page 9-80, Table 9-23 on page 9-80).

The **port** parameter specifies the ports to display. If **all** is specified, all ports in the switch are displayed.

Figure 9-13: Example output from the **show mstp cist port** command

```
CIST Port Information
-----
Port Number.....1
  Port Identifier.....127:1
  Port Role.....Designated Port
  Port State.....Forwarding

Port Number.....2
  Port Identifier.....127:2
  Port Role.....Designated Port
  Port State.....Forwarding

Port Number.....3
  Port Identifier.....127:3
  Port Role.....Designated Port
  Port State.....Forwarding
-----
```

Table 9-22: Parameters in output of the **show mstp cist port** command

Parameter	Meaning
Port Number	Number of the port in the switch.
Port Identifier	Unique identifier of the port. This parameter consists of two parts; one part is the port number, and the other is the priority configured for the port.
Port Role	Whether the role of the port is Disabled, Alternate, Backup, Designated, or Root.
Port State	Whether the port is Disabled, Discarding, Learning, or Forwarding.

Figure 9-14: Example output from the **show mstp cist port=1** command

```

CIST Port Information
-----
Port Number.....1
  Port Identifier.....128:1
  Port Role.....Disabled Port
  Port State.....Discarding
  Switch Port State.....Enabled
  Link Status.....Down
  Port Path Cost.....200000
  External Port Path Cost.....200000
  Designated Bridge.....32768 : 00-00-cd-08-35-e0
  Designated Port.....128:1
  Regional Root Path Cost.....0
  External Root Path Cost.....0
  Edge Port.....No
  Point to Point Link.....Yes (Auto)
  Boundary Port.....Yes
-----

```

Table 9-23: Parameters in output of the **show mstp cist port=1** command

Parameter	Meaning
Port Number	Number of the port in the switch.
Port Identifier	Unique identifier of the port. This parameter consists of two parts, one part is the port number, and the other is the priority configured for the port.
Port Role	Whether the role of the port is Disabled, Alternate, Backup, Designated, or Root.
Port State	Whether the state of the port is Disabled, Discarding, Learning, or Forwarding.
Switch Port State	Whether the port is enabled.
Link Status	Whether the link is up or down.
Port Path Cost	Path cost of the port within the region.
External Port Path Cost	Path cost of the port outside the region when the port is a boundary port
Edge Port	Whether this is an edge port that attaches to a LAN and known to have no other bridges attached.
Point to Point Link	Whether the port has a point-to-point connection with another bridge.
Boundary Port	Whether the port is a boundary port in the MST region.

**Example** To display port 1 information in the CIST, use the command:

```
sh mst cist po=1
```

**Related Commands**

- [enable mstp](#)
- [disable mstp](#)
- [set mstp cist](#)
- [enable mstp cist port](#)
- [show mstp](#)



## show mstp counter port

**Syntax** SHow MSTp COUnTer Port={*port-list*|ALL}

where *port-list* is a port number, range (specified as *n-m*), or comma-separated list of numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered Ethernet switch port, including uplink ports.

**Description** This command displays counter information for a specified port or ports (Figure 9-15, Table 9-24).

The **port** parameter specifies the ports to display. If **all** is specified, all ports on the switch are displayed.

Figure 9-15: Example output from the **show mstp counter port** command

MSTP Port Counters			
-----			
Port Number	1		
Receive:		Transmit:	
Total BPDUs	0	Total BPDUs	0
MSTP BPDUs	0	MSTP BPDUs	0
RSTP BPDUs	0	RSTP BPDUs	0
STP BPDUs	0	STP BPDUs	0
Invalid BPDUs	0		
Discarded:			
Port Disabled	0		
Invalid Protocol	0		
Invalid Type	0		
Invalid BPDU length	0		
-----			

Table 9-24: Parameters in output of the **show mstp counter port** command

Parameter	Meaning
Total BPDUs	Whether the role of the port is Disabled, Alternate, Backup, Designated, or Root.
MSTP BPDU	Number of received MSTP BPDUs
RSTP BPDUs	Number of received RSTP BPDUs
STP BPDUs	Number of received STP BPDUs
Invalid BPDUs	Number of received invalid BPDUs
Transmit	BPDUs transmitted
Total BPDUs	Total number of transmitted BPDUs.
MSTP BPDU	Number of transmitted MSTP BPDUs.
RSTP BPDUs	Number of transmitted RSTP BPDUs.
STP BPDUs	Number of transmitted STP BPDUs.
Discard	BPDUs discarded.
Port Disabled	Number of BPDUs discarded because the port that the BPDU was received on was disabled.
Invalid Protocol	Number of BPDUs that had an invalid Protocol Identifier field or invalid Protocol Version Identifier field.

Table 9-24: Parameters in output of the **show mstp counter port** command (cont.)

Parameter	Meaning
Invalid Type	Number of BPDUs that had an invalid Type field.
Invalid Message Age	Number of BPDUs that had an invalid message age.
Invalid BPDU Length	Number of BPDUs that had an incorrect length.

**Examples** To display the counters for port 1 to 3, use the command:

```
sh mst po=1-3 cou
```

**Related Commands** [enable mstp](#)  
[disable mstp](#)  
[reset mstp counter port](#)  
[set mstp cist](#)

## show mstp debug

**Syntax** SHow MSTp DEBug MSTI={CIST|*instance*|ALL}

where *instance* is an instance number from 1 to 4094 for a specific MSTI.

**Description** This command displays the MSTP debugging modes that are enabled on a specified MSTP instance or all instances ([Figure 9-16](#)).

Figure 9-16: Example output from the **show mstp debug msti** command

MSTP Instance	Port	Debug Modes State Machine Debug Modes	Output	Timeout
-----				
CIST	1	MSG, STATE	Asyn 0 (16)	None
		PTM, PIM, PST, PST		
	2	PKT	Asyn 0 (16)	1
		All		
	3	MSG, PKT, STATE	Asyn 0 (16)	2
		PRX, PPM, PTX, PRS, PRT, PST		
	4	MSG, STATE	Asyn 0 (16)	3
		PTM, PIM, PST, PST		
-----				

**Example** To display the debug mode for all MSTIs, use the command:

```
sh mst deb msti=all
```

**Related Commands** [enable mstp debug](#)  
[disable mstp debug](#)  
[enable mstp](#)  
[disable mstp](#)

## show mstp msti

**Syntax** `SHoW MSTp MSTI [= { instance | All }]`

where *instance* is an instance number from 1 to 4094 for a specific MSTI.

**Description** This command displays information about a specific Multiple Spanning Tree Instance or all instances. If **all** is specified, all MSTIs are displayed (Figure 9-17, Table 9-25).

The **msti** parameter specifies the instance number for a specific Multiple Spanning Tree Instance to be displayed. If no value is specified, summary information about all MSTIs is shown (Figure 9-18 on page 9-84, Table 9-26 on page 9-84).

Figure 9-17: Example output from the **show mstp msti** command

```

Multiple Spanning Tree Instances
-----
MSTI ..... 1
  Bridge Identifier ..... 32768 : 00-00-cd-05-19-28
  Bridge Role ..... Designated Bridge
  VLAN Members ..... 1,3-5,7,9

MSTI ..... 2
  Bridge Identifier ..... 32767 : 00-00-cd-05-19-28
  Bridge Role ..... Designated Bridge
  VLAN Members ..... 2,6,8,10-12

MSTI ..... 3
  Bridge Identifier ..... 32766 : 00-00-cd-05-19-28
  Bridge Role ..... Designated Bridge
  VLAN Members ..... 13-20,22
-----

```

Table 9-25: Parameters in output of the **show mstp msti** command

Parameter	Meaning
MSTI	Instance number of the spanning tree.
Bridge Identifier	Unique bridge identifier of the switch. This parameter consists of two parts: one is derived from the switch's unique MAC Address, and the other is the priority value entered for the switch.
Bridge Role	Whether the role of the bridge in the spanning tree is root bridge or designated bridge.
VLAN Members	A list of the VLANs that are mapped to a specified multiple spanning tree instance.

Figure 9-18: Example output from the **show mstp msti=1** command

```

Multiple Spanning Tree Instance
-----
MSTI ..... 1
  Bridge Identifier ..... 32768 : 00-00-cd-05-19-28
  Bridge Role ..... Root Bridge
  VLAN Members ..... vlan1, vlan2-vlan10, vlan20
  Regional Root Identifier ..... 32768 : 01-00-cd-05-19-28
  Designated Bridge ..... 32768 : 02-00-cd-05-19-28
  Root Path Cost ..... 32
  Root Port ..... 2
  Designated Port.....128:1
  Topology Changes:
    Time Since Topology Change .. 100
    Topology Change Count ..... 3
    Topology Change ..... FALSE
-----

```

Table 9-26: Parameters in output of the **show mstp msti=1** command

Parameter	Meaning
MSTI	Instance number of the spanning tree.
Bridge Identifier	Unique Bridge Identifier of the switch. This parameter consists of two parts: one part is derived from the switch's unique MAC Address, and the other part is the priority value entered for the switch.
Bridge Role	Whether the role of the bridge in the spanning tree is root bridge or designated bridge.
VLAN Members	A list of the VLANs that are mapped to a specified multiple spanning tree instance.
Regional Root Identifier	Bridge identifier of the root bridge for the MSTI in an MST region.
Designated Bridge	Bridge identifier for the transmitting bridge for the spanning tree.
Designated Port	The port on a designated bridge nominated to distribute BPDUs to its leaf networks. This output displays two components, a port priority component (lower number = higher priority) followed by a physical port number. So the display 128:1 represents a priority of 128 for physical port 1.
Root Path Cost	Path cost to the regional root.
Root Port	Port number of the root port for the switch. This parameter is not valid when the switch is the root bridge and n/a is displayed.
Time Since Topology Change	Seconds elapsed since the last topology change.
Topology Change Count	Number of times that the topology has changed since the bridge was powered on or initialised.
Topology Change	Whether the topology is in the middle of changing.

**Example** To display the information about a specified MSTI5, use the command:

```
sh mst msti=5
```

**Related Commands**

- [disable mstp](#)
- [enable mstp](#)
- [set mstp cist](#)
- [set mstp cist port](#)

## show mstp msti port

**Syntax** `SHoW MSTp MSTI=instance POrt={port-list|All}`

where:

- *instance* is the instance number of the specified MSTI in a range from 1 to 4094.
- *port-list* is a port number, range (specified as *n-m*), or comma-separated list of port numbers and/or ranges. Port numbers start at 1 and end at *m*, where *m* is the highest numbered Ethernet switch port, including uplink ports.

**Description** This command displays the port information of a specified multiple spanning tree instance (MSTI) (Figure 9-19, Table 9-27).

The **msti** parameter specifies the instance number for the specified MSTI to be displayed. The **port** parameter specifies the ports to display. If **all** is specified, all ports on the switch are displayed.

Figure 9-19: Example output from the **show mstp msti=1 port=1** command

```
MSTI 1 Port Information
-----
Port Number ..... 1
Port Identifier ..... 128:1
Port Role ..... Designated Port
Port State ..... Forwarding
Link Status ..... Forwarding
Port Path Cost.....200,000
Switch Port State .....Enabled
Port Path Cost ..... 200
Designated Bridge.....4096 : 00-00-cd-10-00-37
Designated Port.....128:1
-----
```

Table 9-27: Parameters in output of the **show mstp msti port** command

Parameter	Meaning
Port Number	The number of the port in the switch.
Port Identifier	The unique identifier of the port. This parameter consists of two parts, one part is the port number, and the other is the priority configured for the port.
Port Role	The role of the port, this can be either; <b>disabled</b> , <b>alternate</b> , <b>backup</b> , <b>designated</b> , or <b>root</b> .
Port State	The state of the port. The state can be either; Disabled, Discarding, Learning, or Forwarding.
Switch Port State	The state of the port. This can be either <b>enabled</b> or <b>disabled</b>
Link Status	The link state of the port. This can be either <b>up</b> or <b>down</b>
Port Path Cost	The path cost of the port.

Table 9-27: Parameters in output of the **show mstp msti port** command (cont.)

Parameter	Meaning
Designated Bridge	Either the unique Bridge Identifier of the switch, or the unique Bridge Identifier of the switch believed to be the Designated Bridge for the LAN to which the port is attached.
Designated Port	Port Identifier of the port on the Designated Bridge through which the Designated Bridge transmits Configuration BPDU information stored by this port.

**Example** To display the information of port 1 for MSTI5, use the command:

```
sh mst msti=5 po=1
```

**Related Commands**

- [disable mstp](#)
- [enable mstp](#)
- [set mstp cist](#)
- [set mstp cist port](#)

## show stp

**Syntax** `SHOW STP [= {stp-name | ALL}] [SUMmary]`

where *stp-name* is a character string 1 to 15 characters long. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen.

**Description** This command displays information about the specified Spanning Tree Protocol instance (STP), or all STPs (Figure 9-20, Table 9-28 on page 9-88).

If the **summary** parameter is specified, then a summary table of all configured STPs is displayed (Figure 9-21 on page 9-90, Figure 9-29 on page 9-90).

Figure 9-20: Example output from the **show stp** command

```
STP Information
-----
Name ..... grey
Mode ..... Rapid
RSTP Type ..... Normal
VLAN members ..... vlan4 (4)
Status ..... ON
Number of Ports ..... 2
    Number Enabled ..... 2
    Number Disabled ..... 0
Bridge Identifier ..... 32768 : 00-00-cd-05-19-28
Bridge Priority ..... 32768
Root Bridge ..... 32768 : 00-00-cd-05-19-28
Designated Bridge ..... 32768 : 00-00-cd-05-19-28
Root Port ..... (n/a)
Root Path Cost ..... 0
Max Age ..... 20
Hello Time ..... 2
Forward Delay ..... 15
Switch Max Age ..... 20
Switch Hello Time ..... 2
Switch Forward Delay .. 15
Transmission Limit .... 3
```

Figure 9-20: Example output from the **show stp** command (cont.)

```

Name ..... default
Mode ..... Standard
RSTP Type ..... (n/a)
VLAN members ..... default (1)
                        vlan5 (5)
                        vlan6 (6)
                        vlan7 (7)
                        vlan8 (8)
                        vlan9 (9)
                        vlan10 (10)
                        vlan11 (11)
                        vlan12 (12)
                        vlan13 (13)
                        vlan14 (14)
Status ..... ON
Number of Ports ..... 22
    Number Enabled ..... 10
    Number Disabled ..... 12
Bridge Identifier ..... 32768 : 00-00-cd-05-19-28
Bridge Priority ..... 32768
Designated Root ..... 32768 : 00-00-cd-05-19-28
Root Port ..... (n/a)
Root Path Cost ..... 0
Max Age ..... 20
Hello Time ..... 2
Forward Delay ..... 15
Switch Max Age ..... 20
Switch Hello Time ..... 2
Switch Forward Delay .. 15
Hold Time ..... 1
Number of TC ..... 6
Time since last TC .... 260
-----

```

Table 9-28: Parameters in output of the **show stp** command

Parameter	Meaning
STP Name	The name of the Spanning Tree Protocol entity.
Mode	Whether STP is running in standard, or rapid mode.
RSTP Type	Whether RSTP is operating normally, or as STP compatible. In STP compatible mode, the rapid transitions to forwarding do not occur.
VLAN members	A list of the VLANs that are members of the STP. VLAN Identifiers are shown in brackets.
Status	The status of the STP; either ON or OFF.
Number of Ports	The number of ports belonging to the STP.
Number Enabled	Number of ports enabled with the <b>enable stp</b> command and are being considered by the Spanning Tree Algorithm.
Number Disabled	Number of ports disabled with the <b>disable stp</b> command and are not being considered by the Spanning Tree Algorithm.
Bridge Identifier	A unique identifier of a specific bridge within a specific VLAN. This parameter consists of three parts: a MAC address, a bridge priority component, and a VLAN ID.



Table 9-28: Parameters in output of the **show stp** command (cont.)

Parameter	Meaning
Bridge Priority	A configurable, and more significant, portion of the bridge identifier. The bridge identifier, which comprises the bridge priority plus the bridge address, is used to select which bridge becomes the root. A lower number indicates a higher priority.
Designated Root	The unique Bridge Identifier of the bridge assumed to be the root (standard mode only).
Root Bridge	The unique Bridge Identifier of the bridge assumed to be the Root (rapid mode only).
Designated Bridge	The unique Bridge Identifier of the bridge assumed to be the designated bridge. Displayed when STP is in rapid mode.
Root Port	The port number of the root port for the switch. If the switch is the Root Bridge this parameter is not valid, and (n/a) is shown.
Root Path Cost	The cost of the path to the Root from this switch. If the switch is the Root Bridge this parameter is not valid and is not shown.
Max Age	The maximum age of received Configuration Message information before it is discarded.
Hello Time	The time interval (in seconds) between successive transmissions of the Configuration Message information by a switch that is the Root or is trying to become the Root.
Forward Delay	In STP standard mode, the time ports spend in the Listening state before moving to the Learning state and the Learning state before moving to the Forwarding state. In rapid mode, the maximum time taken to transition from discarding to learning and learning to forwarding. In both modes, the value is also used for the ageing timer for the dynamic entries in the forwarding database.
Switch Max Age	The value of the Max Age parameter when this switch is the Root or is attempting to become the root. This parameter is set by the <b>maxage</b> parameter in the <b>set stp</b> command.
Switch Hello Time	The value of the Hello Time parameter when this switch is the Root or is attempting to become the Root. This parameter is set by the <b>hellotime</b> parameter in the <b>set stp</b> command.
Switch Forward Delay	The value of the Forward Delay parameter when this switch is the Root or is attempting to become the Root. This parameter is set by the <b>forwarddelay</b> parameter in the <b>set stp</b> command.
Hold Time	The minimum time in seconds between the transmission of configuration BPDUs through a given LAN Port. The value of this fixed parameter is 1, as specified in IEEE Standard 802.1d. This parameter applies only to STP running in standard mode.
Transmission Limit	In rapid mode, this indicates the number of BPDUs that may be transmitted in the interval specified by Hello Time. The value of this fixed parameter is 3, as specified in IEEE Standard 802.1t.

Table 9-28: Parameters in output of the **show stp** command (cont.)

Parameter	Meaning
Number of TC	The number of topology changes since the switch last restarted.
Time since last TC	The number of seconds since the last topology change.

Figure 9-21: Example output from the **show stp summary** command

STP Name	Mode	Ports Enabled	Ports Disabled	Bridge Role
Rstp1	Rapid	0	2	Root Bridge
Default	Standard	0	21	Root Bridge

Table 9-29: Parameters in output of the **show stp summary** command

Parameter	Meaning
STP name	Name of the Spanning Tree Protocol entry.
Mode	Whether STP is running in standard or rapid mode.
Ports Enabled	Number of ports being considered by the Spanning Tree Algorithm.
Ports Disabled	Number of ports that have been disabled and are not active in the Spanning Tree Algorithm.
Bridge Role	Role of the bridge in the STP, either None, Designated, or Root.

**Example** To show the current settings of the company STP, use the command:

```
sh stp=company
```

**Related Commands**

- [create stp](#)
- [destroy stp](#)
- [disable stp](#)
- [enable stp](#)
- [set stp](#)
- [show stp counter](#)
- [show stp port](#)

## show stp counter

**Syntax** `SHoW STP[={stp-name|ALL}] COUnTer`

where *stp-name* is a character string 1 to 15 characters long. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen.

**Description** This command displays Spanning Tree Protocol counters for the specified STP or all STPs (Figure 9-22, Table 9-30). If no STP is specified, then counters for all STPs are displayed. If the port link status is **down**, then no STP BPDUs are transmitted on the port.

Figure 9-22: Example output from the **show stp counter** command

STP Counters			
-----			
STP Name: default			
Receive:		Transmit:	
Total STP Packets	56	Total STP Packets	2
Configuration BPDU	56	Configuration BPDU	0
TCN BPDU	0	TCN BPDU	0
RSTP TC-Flag	0	RSTP TC-Flag	0
RST BPDU	0	RSTP BPDU	0
Invalid BPDU	0		
Discarded:			
Port Disabled	0		
Invalid Protocol	0		
Invalid Type	0		
Invalid Message Age	0		
Config BPDU length	0		
TCN BPDU length	0		
RST BPDU length	0		
-----			

Table 9-30: Parameters in output of the **show stp counter** command

Parameter	Meaning
STP Name	Name of the STP.
<b>Receive</b>	STP packets received.
Total STP Packets	Total number of STP packets received. Valid STP packets comprise Configuration BPDUs and Topology Change Notification (TCN) BPDUs.
Configuration BPDU	Number of valid Configuration BPDUs received.
TCN BPDU	Number of valid Topology Change Notification BPDUs received.
RSTP TC-FLAG	Counts the number of times a BPDU has been received with the TC-flag set.
RST BPDU	Number of valid Rapid Spanning Tree BPDUs received (rapid mode only).
Invalid BPDU	Number of invalid STP packets received.
<b>Transmit</b>	STP packets transmitted.
Total STP packets	Total number of STP packets transmitted.

Table 9-30: Parameters in output of the **show stp counter** command (cont.)

Parameter	Meaning
Configuration BPDU	Number of Configuration BPDUs transmitted.
TCN BPDU	Number of Topology Change Notification BPDUs transmitted.
RST BPDU	Number of valid Rapid Spanning Tree BPDUs transmitted (rapid mode only).
<b>Discarded</b>	STP packets discarded.
Port Disabled	Number of BPDUs discarded because the port that the BPDU was received on was disabled.
Invalid Protocol	Number of STP packets that had an invalid Protocol Identifier field or invalid Protocol Version Identifier field.
Invalid Type	Number of STP packets that had an invalid Type field.
Invalid Message Age	Number of STP packets that had an invalid message age.
Config BPDU length	The number of Configuration BPDUs that had an incorrect length.
TCN BPDU length	Number of Topology Change Notification BPDUs that had an incorrect length.
RST BPDU length	Number of Rapid Spanning Tree BPDUs that had an incorrect length (rapid mode only).

**Example** To show the counters for all STPs, use the command:

```
sh stp cou
```

**Related Commands** [reset stp](#)  
[show stp](#)  
[show stp port](#)

## show stp debug

**Syntax** `SHoW STP [= {stp-name | ALL}] DEBuG`

where *stp-name* is a character string 1 to 15 characters long. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen.

**Description** This command shows the debugging modes enabled on each port ([Figure 9-23](#), [Table 9-31](#)).

An STP name can be specified. If no parameter is entered, the default is **all**.

Figure 9-23: Example output from the **show stp debug** command

STP Name	Port	Enabled Debug Modes	Output	Timeout
-----				
default				
	Port1	MSG, PKT, STATE	Console (16)	NONE
	Port2	STATE	Console (16)	12345
	Port3	None		
-----				
Admin				
	Port1	MSG, PKT, STATE	TTY (12)	100
-----				

Table 9-31: Parameters in output of the **show stp debug** command

Parameter	Meaning
Port	Port number on the switch.
Enabled Debug Modes	Whether the debugging option for the port is MSG, PKT, STATE, or NONE.
Output	Output device for the port.
Timeout	Time in seconds that the port stays in debug mode. If a timeout value is not set, "None" is shown.
STP name	Name of the STP instance.

**Example** To display the debug status for all ports in the switch, use the command:

```
sh stp deb
```

To show STP on just the ADMIN network, use the command:

```
sh stp=admin deb
```

**Related Commands**

- [disable stp debug](#)
- [enable stp debug](#)
- [show stp counter](#)
- [disable debug active](#) in Chapter 4, Configuring and Monitoring the System
- [show debug active](#) in Chapter 4, Configuring and Monitoring the System

## show stp port

**Syntax** SHow STP[={*stp-name*|ALL}] Port={*port-list*|ALL} [RSTPstate]

**Description** This command displays Spanning Tree Protocol port information for the specified ports, or all ports for the specified STP, or all STPs, ([Figure 9-24](#), [Table 9-32 on page 9-95](#)).

Parameter	Description
STP	The STP that port information is displayed for. Default: <b>all</b>
<i>stp-name</i>	A character string 1 to 15 characters long. Valid characters are uppercase and lowercase letters, digits, the underscore, and hyphen.
ALL	All STPs.
Port	The ports that port information is displayed for.
<i>port-list</i>	A port number, range (specified as <i>n-m</i> ), or comma-separated list of numbers and/or ranges. Port numbers start at 1 and end at <i>m</i> , where <i>m</i> is the highest numbered Ethernet switch port, including uplink ports.
ALL	All ports.
RSTPstate	Displays information about the status of RSTP state machines on the specified ports and STP instances.

Figure 9-24: Example output from the **show stp port** command

```

STP Port Information
-----
STP ..... stp20
  STP Status ..... ON

  Port ..... 14
    RSTP Port Role ..... Disabled
    State ..... Discarding
    Point To Point ..... No (Auto)
    Port Priority ..... 128
    Port Identifier ..... 800e
    Pathcost ..... 200000 (auto configured)
    Designated Root ..... 32768 : 00-00-cd-28-06-5a
    Designated Cost ..... 0
    Designated Bridge ... 32768 : 00-00-cd-28-06-5a
    Designated Port ..... 800e
    EdgePort ..... No
    VLAN membership ..... 1
    Send RSTP BPDU ..... TRUE
    Counters:
      Loopback Disabled ..... 0

```

Table 9-32: Parameters in output of the **show stp port** command

Parameter	Meaning
STP	Name of the STP of which the port is a member.
STP Status	Whether the STP is enabled.
Port	Port number.
RSTP Port Role	Role of the port, rapid mode only:
	Disabled
	Alternate A port that is presently blocked, but could offer an alternative path to the root bridge, if required.
	Backup A port that is presently blocked, but could offer an alternative path to the designated bridge, if required.
	Backup (Loopback Disabled) Same as Backup except all packets are dropped, including BPDUs. The port transmitted and received the same RSTP BPDU.
	Designated A port on a the designated bridge selected to convey data between a downstream LAN to which it connects, and the root bridge.
	Root A port on a bridge selected to receive frames on the least path cost from the root bridge.
State	Status of the port: For more information on port states see, <a href="#">For more information on port states see, "Spanning Tree and Rapid Spanning Tree Port States" on page 9-7.</a>
	Disabled Standard and rapid modes
	Blocking Standard mode
	Listening Standard mode
	Learning Standard and rapid modes
	Forwarding Standard and rapid modes
	Discarding Rapid mode
Point To Point	Whether the port has a point to point connection with another bridge (rapid mode only).
Port Priority	Priority of the port. Used as part of the Port Identifier field. In standard mode it forms the upper 8 bits of the Port Identifier field. In rapid mode it forms the upper 4 bits of the Port Identifier field
.Port Identifier	Unique identifier of the port. This parameter determines the root port or designated port of the switch.
Pathcost	Path cost of the port.
Designated Root	Unique Bridge Identifier of the Root Bridge, as recorded in the configuration BPDU.
Designated Cost	Designated Cost for the port.
Designated Bridge	Either the unique Bridge Identifier of the switch, or the unique Bridge Identifier of the switch believed to be the Designated Bridge for the LAN to which the port is attached.
Designated Port	Port Identifier of the port on the Designated Bridge through which the Designated Bridge transmits Configuration BPDU information stored by this port.

Table 9-32: Parameters in output of the **show stp port** command (cont.)

Parameter	Meaning
Edge Port	Whether this is an edge port, which is one that attaches to a LAN and is known to have no other bridges attached (rapid mode only).
VLAN membership	Number of VLANs the port is a member of within this STP instance.
Counters	The counters for the selected port.
Loopback Disabled	Number of transitions to the Backup (Loopback Disabled) RSTP port role.

Figure 9-25: Example output from the **show stp port rstpstate** command

```

RSTP State Information
-----
STP Name: EVA02
  Bridge Level State Machine ..... STATE
  Port Role Selection ..... Role Selection
  Port ..... 15
  Port State Machines ..... STATE
  Port Information ..... Current
  Port Role Transitions ..... Root Port
  Port State Transition ..... Forwarding
  Topology Change ..... Active
  Port Protocol Migration ..... Sending RSTP
  Port Transmit ..... Idle
  Port ..... 16
  Port State Machines ..... STATE
  Port Information ..... Disabled
  Port Role Transitions ..... Blocked Port
  Port State Transition ..... Discarding
  Topology Change ..... Inactive
  Port Protocol Migration ..... Init
  Port Transmit ..... Idle
-----

```



Table 9-33: Parameters in output of the **show stp port rstpstate** command

Parameter	Meaning
STP Name	Name of the Spanning Tree Protocol entity.
Bridge Level State Machine	The entry below this heading lists an aspect of the bridge level RSTP state machine. The current setting for the aspect is given in the column below the heading "STATE". See the IEEE Standard 802.1w-2001 for a full description of the states displayed.
Port Role Selection	The status of the Port Role Selection state machine. This displays whether RSTP is recalculating the port role for the bridge (Role Selection) or not (None).
Port	Port number.
Port State Machines	The entries below this heading list aspects of the port level RSTP state machines. The current setting for each aspect is given in the column below the heading "STATE". See the IEEE Standard 802.1w-2001 for a full description of the states displayed.
Port Information	The status of the Port Information state machine - whether it is receiving or propagating information, or (if neither) the status of the current information. RSTP calculates the status of the current information by aging the BPDUs sent to this port. It uses this to help determine whether it needs to change the port's role.
Port Role Transitions	The status of the Port Role Transitions state machine. This displays the status of this port within the RSTP bridge.
Port State Transition	The status of the Port State Transition state machine. This displays state the port is currently in, either forwarding, discarding, or learning.
Topology Change	The status of the Topology Change state machine. This displays whether RSTP is currently detecting and propagating topology changes on this port.
Port Protocol Migration	The status of the Port Protocol Migration state machine. This displays the type of BPDUs that RSTP transmits across this port. RSTP sets this to either STP or RSTP BPDUs, based on whether the receiving device is running STP or RSTP.
Port Transmit	The status of the Port Transmit state machine. This displays whether RSTP is currently transmitting any BPDUs.

**Example** To show STP information for port 2 on the STP named 'grey', use the command:

```
sh stp=grey po=2
```

**Related Commands**

- [disable stp port](#)
- [enable stp port](#)
- [set stp port](#)
- [show stp](#)

