ProCurve
Networking by HP

Advanced Traffic Management Guide

2810

ProCurve Series 2810 Switches

N.11.XX

www.procurve.com

invent

ProCurve Series 2810 Switches

July 2007

Advanced Traffic Management Guide

Publication Number

5991-4733
July 2007

Applicable Products

ProCurve Switch 2810-24G - (J9021A)
ProCurve Switch 2810-48G - (J9022A)

Trademark Credits

Microsoft, Windows, and Windows NT are US registered trademarks of Microsoft Corporation.

Disclaimer

HEWLETT-PACKARD COMPANY MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Hewlett-Packard assumes no responsibility for the use or reliability of its software on equipment that is not furnished by Hewlett-Packard.

Warranty

See the Customer Support/Warranty booklet included with the product.

A copy of the specific warranty terms applicable to your Hewlett-Packard products and replacement parts can be obtained from your HP Sales and Service Office or authorized dealer.

Hewlett-Packard Company
8000 Foothills Boulevard, m/s 5551
Roseville, California 95747-5551
http://www.procurve.com

Contents

Product Documentation

1 Getting Started

2 Static Virtual LANs (VLANs)

3 GVRP

4 Multimedia Traffic Control with IP Multicast (IGMP)

5 Multiple Instance Spanning-Tree Operation

6 Quality of Service (QoS): Managing Bandwidth More Effectively

7 ProCurve Stack Management

Product Documentation

About Your Switch Manual Set

The switch manual set includes the following:

- **Read Me First** - a printed guide shipped with your switch. Provides software update information, product notes, and other information.
- **Installation and Getting Started Guide** - a printed guide shipped with your switch. This guide explains how to prepare for and perform the physical installation and connection to your network.
- **Management and Configuration Guide** - a PDF file on the ProCurve Networking website. This guide describes how to configure, manage, and monitor basic switch operation.
- **Advanced Traffic Management Guide** - a PDF file on the ProCurve Networking website. This guide explains the configuration and operation of traffic management features such as spanning tree and VLANs.
- **Access Security Guide** - a PDF file on the ProCurve Networking website. This guide explains the configuration and operation of access security and user authentication features on the switch.
- **Release Notes** - posted on the ProCurve web site to provide information on software updates. The release notes describe new features, fixes, and enhancements that become available between revisions of the above guides.

Note

For the latest version of all ProCurve switch documentation, including release notes covering recently added features, visit the ProCurve Networking website at **http://www.procurve.com**. Click on **Technical support**, and then click on **Product manuals**.

Feature Index

For the manual set supporting your switch model, the following feature index indicates which manual to consult for information on a given software feature.

Feature	Management and Configuration	Advanced Traffic Management	Access Security Guide
802.1Q VLAN Tagging	-	X	-
802.1p Priority	X	-	-
802.1X Authentication	-	-	X
Authorized IP Managers	-	-	X
Config File	X	-	-
Copy Command	X	-	-
Debug	X	-	-
DHCP Configuration	-	X	-
DHCP/Bootp Operation	X	-	-
Diagnostic Tools	X	-	-
Downloading Software	X	-	-
Event Log	X	-	-
Factory Default Settings	X	-	-
File Management	X	-	-
File Transfers	X	-	-
GVRP	-	X	-
IGMP	-	X	-
Interface Access (Telnet, Console/Serial, Web)	X	-	-
IP Addressing	X	-	-
LACP	X	-	-
Link	X	-	-

Feature	Management and Configuration	Advanced Traffic Management	Access Security Guide
LLDP	X	-	-
MAC Address Management	X	-	-
MAC Lockdown	-	-	X
MAC Lockout	-	-	X
MAC-based Authentication	-	-	X
Monitoring and Analysis	X	-	-
Multicast Filtering	-	X	-
Network Management Applications (LLDP, SNMP)	X	-	-
Passwords	-	-	X
Ping	X	-	-
Port Configuration	X	-	-
Port Security	-	-	X
Port Status	X	-	-
Port Trunking (LACP)	X	-	-
Port-Based Access Control	-	-	X
Port-Based Priority (802.1Q)	X	-	-
Quality of Service (QoS)	-	X	-
RADIUS Authentication and Accounting	-	-	X
Secure Copy	X	-	-
SFTP	X	-	-
SNMP	X	-	-
Software Downloads (SCP/SFTP, TFTP, Xmodem)	X	-	-
Source-Port Filters	-	-	X
Spanning Tree (MSTP)	-	X	-
SSH (Secure Shell) Encryption	-	-	X
SSL (Secure Socket Layer)	-	-	X

Feature	Management and Configuration	Advanced Traffic Management	Access Security Guide
Stack Management (Stacking)	-	X	-
Syslog	X	-	-
System Information	X	-	-
TACACS+ Authentication	-	-	X
Telnet Access	X	-	-
TFTP	X	-	-
Time Protocols (TimeP, SNTP)	X	-	-
Traffic/Security Filters	-	-	X
Troubleshooting	X	-	-
VLANs	-	X	-
Web-based Authentication	-	-	X
Xmodem	X	-	-

1

Getting Started

Contents

Introduction

This *Advanced Traffic Management Guide* describes how to manage and configure advanced traffic management features on your switch. It supports the following switches:

- ProCurve Switch 2810

For an overview of other product documentation for the above switches, refer to "Product Documentation" on page xi.

You can download a copy from the ProCurve Networking website, **http://www.procurve.com**.

Conventions

This guide uses the following conventions for command syntax and displayed information.

Feature Descriptions by Model

In cases where a software feature is not available in all of the switch models covered by this guide, the section heading specifically indicates which product or product series offer the feature.

For example (the switch model is highlighted here in ***bold italics***):

"Jumbo Packet Support on the ***2810 Switch***".

Command Syntax Statements

Syntax: aaa port-access authenticator < *port-list* >
[control < authorized | auto | unauthorized >]

- Vertical bars (|) separate alternative, mutually exclusive elements.
- Square brackets ([ ]) indicate optional elements.
- Braces (< >) enclose required elements.

- Braces within square brackets ([< >]) indicate a required element within an optional choice.
- Boldface indicates use of a CLI command, part of a CLI command syntax, or other displayed element in general text. For example:

  “Use the **copy tftp** command to download the key from a TFTP server.”
- Italics indicate variables for which you must supply a value when executing the command. For example, in this command syntax, **<*port-list*>** indicates that you must provide one or more port numbers:

  Syntax: aaa port-access authenticator < *port-list* >

Command Prompts

In the default configuration, your switch displays one of the following CLI prompts:

```
ProCurve Switch 2810#
```

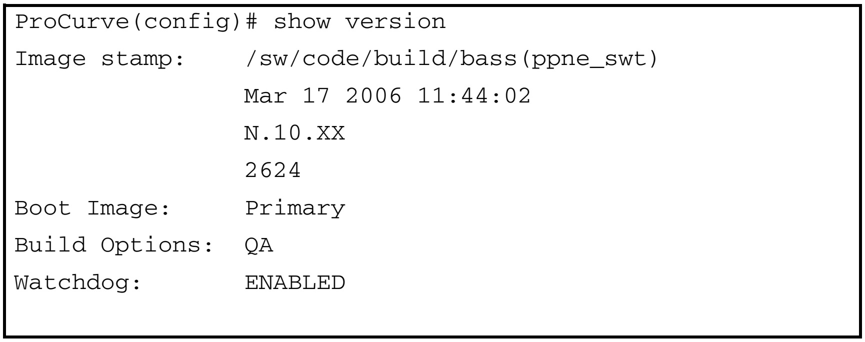
To simplify recognition, this guide uses `ProCurve` to represent command prompts for all models. For example:

```
ProCurve#
```

(You can use the **hostname** command to change the text in the CLI prompt.)

Screen Simulations

Figures containing simulated screen text and command output look like this:



```
ProCurve(config)# show version
Image stamp:     /sw/code/build/bass(ppne_swt)
                 Mar 17 2006 11:44:02
                 N.10.XX
                 2624
Boot Image:      Primary
Build Options:   QA
Watchdog:        ENABLED
```

Figure 1-1. Example of a Figure Showing a Simulated Screen

In some cases, brief command-output sequences appear outside of a numbered figure. For example:

```
ProCurve(config)# ip default-gateway 18.28.152.1/24
ProCurve(config)# vlan 1 ip address 18.28.36.152/24
ProCurve(config)# vlan 1 ip igmp
```

Port Identity Examples

This guide describes software applicable to both chassis-based and stackable ProCurve switches. Where port identities are needed in an example, this guide uses the chassis-based port identity system, such as “A1”, “B3 - B5”, “C7”, etc. However, unless otherwise noted, such examples apply equally to the stackable switches, which for port identities typically use only numbers, such as “1”, “3-5”, “15”, etc.

Sources for More Information

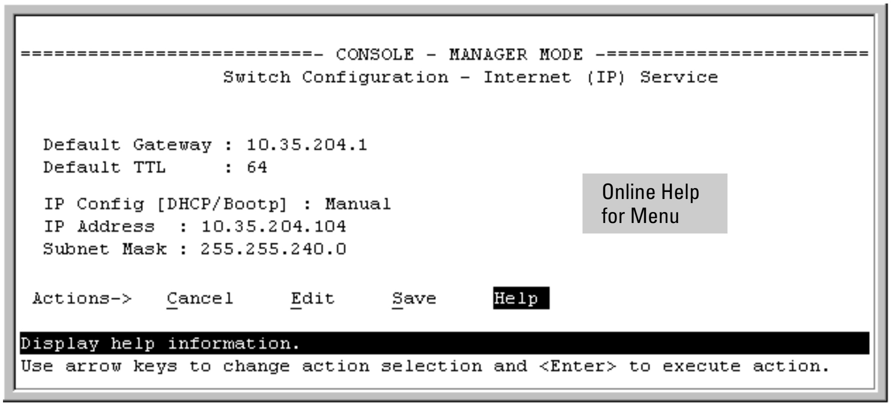
For additional information about switch operation and features not covered in this guide, consult the following sources:

- For information on which product manual to consult on a given software feature, refer to “Product Documentation” on page xi.

Note

For the latest version of all ProCurve switch documentation, including release notes covering recently added features, visit the ProCurve Networking website at **http://www.procurve.com**. Click on **Technical support**, and then click on **Product manuals**.

- For information on specific parameters in the menu interface, refer to the online help provided in the interface. For example:

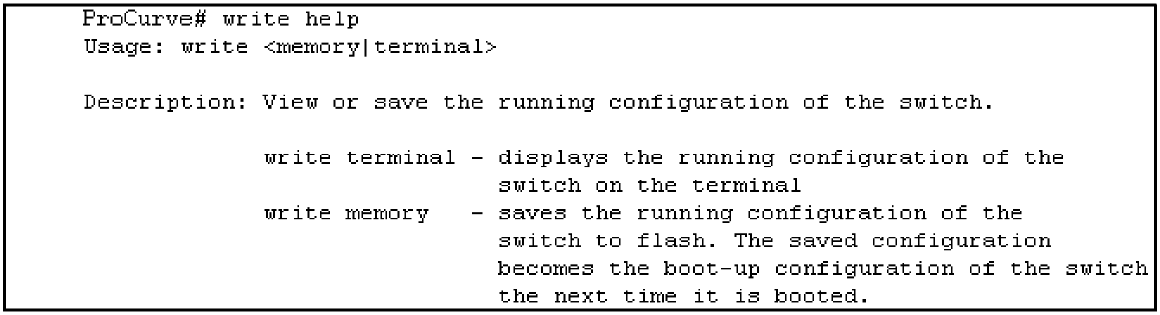


```
=========================- CONSOLE - MANAGER MODE -=======================
                 Switch Configuration - Internet (IP) Service


  Default Gateway : 10.35.204.1
  Default TTL     : 64

  IP Config [DHCP/Bootp] : Manual
  IP Address  : 10.35.204.104
  Subnet Mask : 255.255.240.0

 Actions->   Cancel     Edit     Save     Help

Display help information.
Use arrow keys to change action selection and <Enter> to execute action.
```

Online Help for Menu

Figure 1-2. Getting Help in the Menu Interface

- For information on a specific command in the CLI, type the command name followed by "help". For example:



```
ProCurve# write help
Usage: write <memory|terminal>

Description: View or save the running configuration of the switch.

                write terminal - displays the running configuration of the
                                 switch on the terminal
                write memory   - saves the running configuration of the
                                 switch to flash. The saved configuration
                                 becomes the boot-up configuration of the switch
                                 the next time it is booted.
```

Figure 1-3. Getting Help in the CLI

- For information on specific features in the Web browser interface, use the online help. For more information, refer to the *Management and Configuration Guide* for your switch.

- For further information on ProCurve Networking switch technology, visit the ProCurve Networking website at:

  http://www.procurve.com

Need Only a Quick Start?

IP Addressing

If you just want to give the switch an IP address so that it can communicate on your network, or if you are not using multiple VLANs, ProCurve recommends that you use the Switch Setup screen to quickly configure IP addressing. To do so, do one of the following:

- Enter **setup** at the CLI Manager level prompt.

```
ProCurve# setup
```

- In the Main Menu of the Menu interface, select

  8. Run Setup

For more on using the Switch Setup screen, see the *Quick Installation Guide* you received with the switch.

To Set Up and Install the Switch in Your Network

Important!

Use the *Quick Installation Guide* shipped with your switch for the following:

- Notes, cautions, and warnings related to installing and using the switch
- Instructions for physically installing the switch in your network
- Quickly assigning an IP address and subnet mask, setting a Manager password, and (optionally) configuring other basic features.
- Interpreting LED behavior.

For the latest version of the *Installation and Getting Started Guide* and other documentation for your switch, visit the ProCurve Networking Web site. (Refer to “Product Documentation” on page xi of this guide for further details.)

Static Virtual LANs (VLANs)

Contents

Overview

This chapter describes how to configure and use static, port-based VLANs on the switches covered by this manual.

For general information on how to use the switch's built-in interfaces, refer to these chapters in the *Management and Configuration Guide* for your switch:

- Chapter 3, "Using the Menu Interface"
- Chapter 4, "Using the Command Line Interface (CLI)"
- Chapter 5, "Using the Web Browser Interface"
- Chapter 6, "Switch Memory and Configuration"

Port-Based Virtual LANs (Static VLANs)

VLAN Features

Feature	Default	Menu	CLI	Web
view existing VLANs	n/a	page 2-14 thru 2-19	page 2-20	page 2-25
configuring static VLANs	default VLAN with VID = 1	page 2-14 thru 2-19	page 2-19	page 2-25
configuring dynamic VLANs	disabled	See the chapter on GVRP in this manual.		

A VLAN is a group of ports designated by the switch as belonging to the same broadcast domain. (That is, all ports carrying traffic for a particular subnet address would normally belong to the same VLAN.)

Note

This chapter describes *static* VLANs, which are VLANs you manually configure with a name, VLAN ID (VID), and port assignments. (For information on *dynamic* VLANs, see chapter 3, “GVRP”.)

Using a VLAN, you can group users by logical function instead of physical location. This helps to control bandwidth usage by allowing you to group high-bandwidth users on low-traffic segments and to organize users from different LAN segments according to their need for common resources.

By default, 802.1Q VLAN support is enabled for eight VLANS. You can configure up to 256 VLANs on the switch.

(802.1Q compatibility enables you to assign each switch port to multiple VLANs, if needed, and the port-based nature of the configuration allows interoperation with older switches that require a separate port for each VLAN.)

General Use and Operation. Port-based VLANs are typically used to reduce broadcast traffic and to increase security. A group of network users assigned to a VLAN forms a broadcast domain that is separate from other VLANs that may be configured on a switch. On a given switch, packets are forwarded only between ports that belong to the same VLAN. Thus, all ports carrying traffic for a particular subnet address should be configured to the same VLAN. Cross-domain broadcast traffic in the switch is eliminated and bandwidth is saved by not allowing packets to flood out all ports. Separate VLANs on the switch can communicate with each other through an external router.

For example, referring to figure 2-1, if ports A1 through A4 belong to VLAN_1 and ports A5 through A8 belong to VLAN_2, traffic from end-node stations on ports A2 through A4 is restricted to only VLAN_1, while traffic from ports A5 through A7 is restricted to only VLAN_2. For nodes on VLAN_1 to communicate with VLAN_2, their traffic must go through an external router via ports A1 and A8.

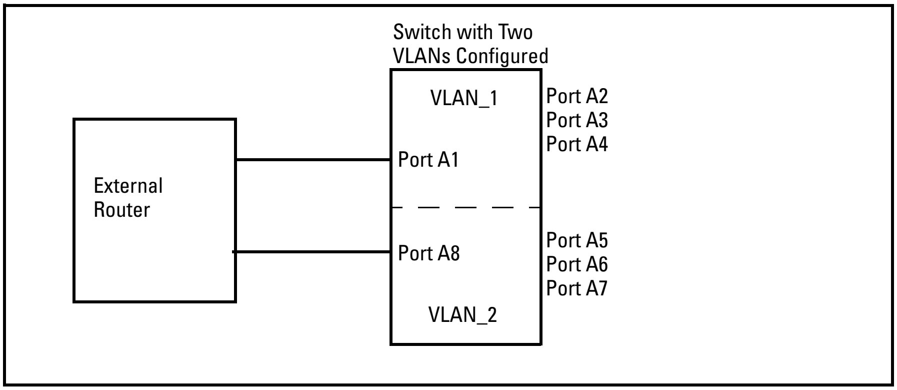


Figure 2-1. Example of Routing Between VLANs via an External Router

Overlapping (Tagged) VLANs. A port on the switch can be a member of more than one VLAN if the device to which it is connected complies with the 802.1Q VLAN standard. For example, a port connected to a central server using a network interface card (NIC) that complies with the 802.1Q standard can be a member of multiple VLANs, allowing members of multiple VLANs to use the server. Although these VLANs cannot communicate with each other through the server, they can all access the server *over the same connection from the switch.* Where VLANs overlap in this way, VLAN “tags” are used to distinguish between traffic from different VLANs.

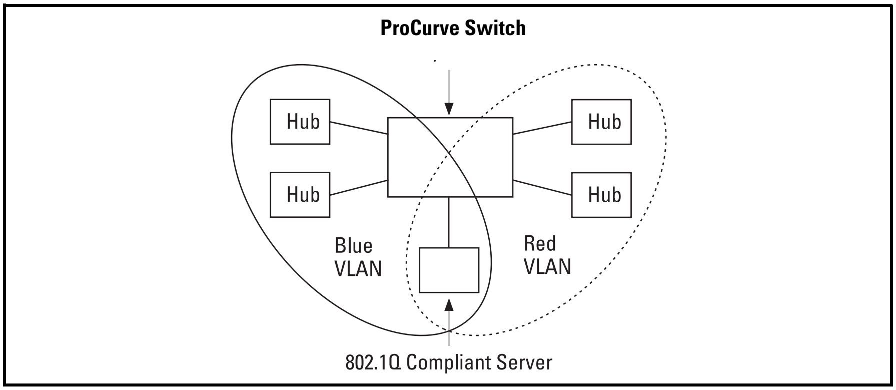


Figure 2-2. Example of Overlapping VLANs Using the Same Server

Similarly, using 802.1Q-compliant switches, you can connect multiple VLANs through a single switch-to-switch link.

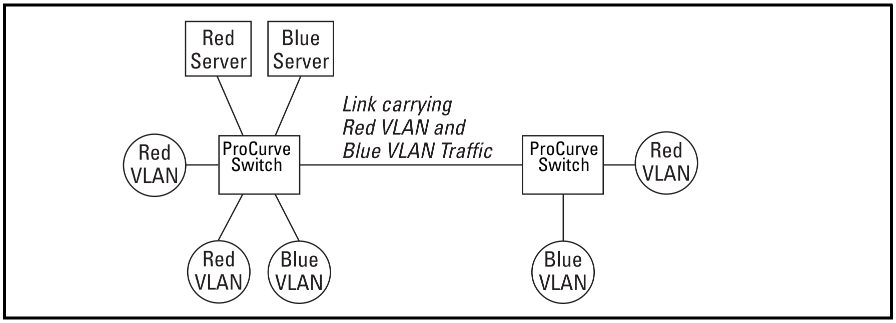


Figure 2-3. Example of Connecting Multiple VLANs Through the Same Link

Introducing Tagged VLAN Technology into Networks Running Legacy (Untagged) VLANs. You can introduce 802.1Q-compliant devices into networks that have built untagged VLANs based on earlier VLAN technology. The fundamental rule is that legacy/untagged VLANs require a separate link for each VLAN, while 802.1Q, or tagged VLANs can combine several VLANs in one link. This means that on the 802.1Q-compliant device, separate ports (configured as untagged) must be used to connect separate VLANs to non-802.1Q devices.

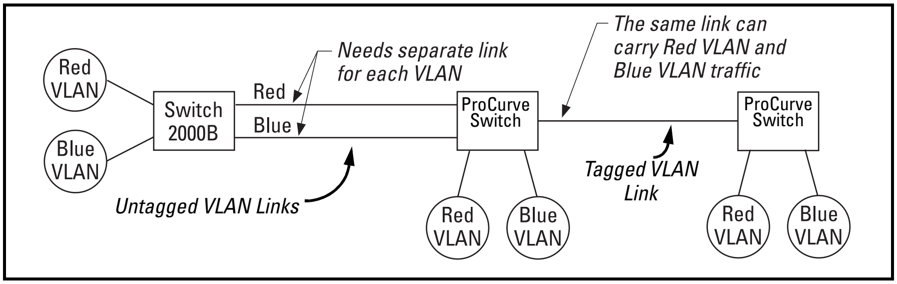


Figure 2-4. Example of Tagged and Untagged VLAN Technology in the Same Network

For more information on VLANs, refer to:

- "Overview of Using VLANs" (page 2-6)
- "Menu: Configuring VLAN Parameters (page 2-14)
- "CLI: Configuring VLAN Parameters" (page 2-14)
- "Web: Viewing and Configuring VLAN Parameters" (page 2-25)
- "VLAN Tagging Information" (page 2-26)
- "Effect of VLANs on Other Switch Features" (page 2-34)
- "VLAN Restrictions" (page 2-36)

Overview of Using VLANs

VLAN Support and the Default VLAN

In the factory default configuration, all ports on the switch belong to the default VLAN (named DEFAULT_VLAN). This places all ports in the switch into one physical broadcast domain. In the factory-default state, the default VLAN is the primary VLAN.

You can partition the switch into multiple virtual broadcast domains by adding one or more additional VLANs and moving ports from the default VLAN to the new VLANs. You can change the name of the default VLAN, but you cannot change the default VLAN's VID (which is always "1"). Although you can remove all ports from the default VLAN, this VLAN is always present; that is, you cannot delete it from the switch.

The Primary VLAN

Because certain features and management functions, such as single IP-address stacking, run on only one VLAN in the switch, and because DHCP and Bootp can run per-VLAN, there is a need for a dedicated VLAN to manage these features and ensure that multiple instances of DHCP or Bootp on different VLANs do not result in conflicting configuration values for the switch. The *primary* VLAN is the VLAN the switch uses to run and manage these features and data. In the factory-default configuration, the switch designates the default VLAN (DEFAULT_VLAN) as the primary VLAN. However, to provide more control in your network, you can designate another VLAN as primary. To summarize, *designating a non-default VLAN as primary* means that:

- The stacking feature runs on the switch's designated primary VLAN instead of the default VLAN
- The switch reads DHCP responses on the primary VLAN instead of on the default VLAN. (This includes such DHCP-resolved parameters as the TimeP server address, Default TTL, and IP addressing—including the Gateway IP address—when the switch configuration specifies DHCP as the source for these values.)
- The default VLAN continues to operate as a standard VLAN (except, as noted above, you cannot delete it or change its VID).
- Any ports not specifically assigned to another VLAN will remain assigned to the Default VLAN, regardless of whether it is the primary VLAN.

Candidates for primary VLAN include any static VLAN currently configured on the switch. (A dynamic—GVRP-learned—VLAN that has not been converted to a static VLAN cannot be the primary VLAN.) To display the current primary VLAN, use the CLI **show vlan** command.

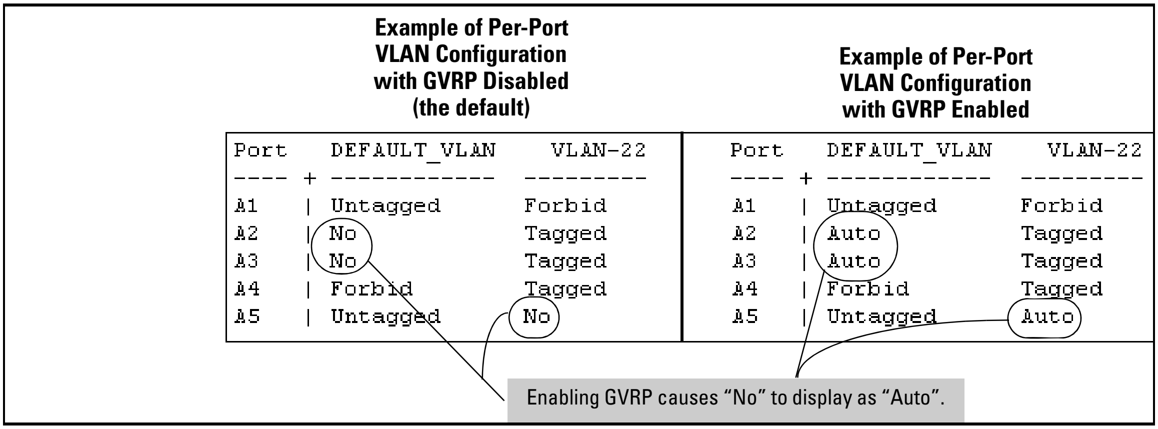
Note

If you configure a non-default VLAN as the primary VLAN, you cannot delete that VLAN unless you first select a different VLAN to act as primary.

If you manually configure a gateway on the switch, it will ignore any gateway address received via DHCP or Bootp.

Per-Port Static VLAN Configuration Options

The following figure and table show the options you have for assigning individual ports to a static VLAN. Note that GVRP, if configured, affects these options and VLAN behavior on the switch. The display below shows the per-port VLAN configuration options. Table 2-1 briefly describes these options.



Example of Per-Port VLAN Configuration with GVRP Disabled (the default)

```
Port   DEFAULT_VLAN    VLAN-22
---- + ------------  ---------
A1   | Untagged      Forbid
A2   | No            Tagged
A3   | No            Tagged
A4   | Forbid        Tagged
A5   | Untagged      No
```

Example of Per-Port VLAN Configuration with GVRP Enabled

```
Port   DEFAULT_VLAN    VLAN-22
---- + ------------  ---------
A1   | Untagged      Forbid
A2   | Auto          Tagged
A3   | Auto          Tagged
A4   | Forbid        Tagged
A5   | Untagged      Auto
```

Enabling GVRP causes “No” to display as “Auto”.

Figure 2-5. Comparing Per-Port VLAN Options With and Without GVRP

Table 2-1. Per-Port VLAN Configuration Options

Parameter	Effect on Port Participation in Designated VLAN
Tagged	Allows the port to join multiple VLANs.
Untagged	Allows VLAN connection to a device that is configured for an untagged VLAN instead of a tagged VLAN. The switch allows no more than one untagged VLAN assignment per port.
No *- or -* **Auto**	**No**: Appears when the switch is not GVRP-enabled; prevents the port from joining that VLAN. **Auto**: Appears when GVRP is enabled on the switch; allows the port to dynamically join any advertised VLAN that has the same VID
Forbid	Prevents the port from joining the VLAN, regardless of whether GVRP is enabled on the switch.

General Steps for Using VLANs

1. Plan your VLAN strategy and create a map of the logical topology that will result from configuring VLANs. Include consideration for the interaction between VLANs and other features such as Spanning Tree Protocol, load balancing, and IGMP. (Refer to "Effect of VLANs on Other Switch Features" on page 2-34.) If you plan on using dynamic VLANs, include the port configuration planning necessary to support this feature. (See chapter 3, "GVRP".)

   By default, VLAN support is enabled and the switch is configured for eight VLANs.

2. Configure at least one VLAN in addition to the default VLAN.
3. Assign the desired switch ports to the new VLAN(s).
4. If you are managing VLANs with SNMP in an IP network, each VLAN must have an IP address. Refer to the chapter on IP addressing in the *Management and Configuration Guide*.

VLAN Operating Notes

- If you are using DHCP/Bootp to acquire the switch's configuration, packet time-to-live, and TimeP information, you must designate the VLAN on which DHCP is configured for this purpose as the primary VLAN. (In the factory-default configuration, the DEFAULT_VLAN is the primary VLAN.)
- IGMP, and some other features operate on a "per VLAN" basis. This means you must configure such features separately for each VLAN in which you want them to operate.
- You can rename the default VLAN, but you cannot change its VID (1) or delete it from the switch.
- Any ports *not* specifically assigned to another VLAN will remain assigned to the DEFAULT_VLAN.
- To delete a VLAN from the switch, you must first remove from that VLAN any ports assigned to it.
- Changing the number of VLANs supported on the switch requires a reboot. Other VLAN configuration changes are dynamic.

Multiple VLAN Considerations

Switches use a *forwarding database* to maintain awareness of which external devices are located on which VLANs. Some switches, such as those covered by this guide, have a *multiple-forwarding database*, which means the switch allows multiple database entries of the same MAC address, with each entry showing the (different) source VLAN and source port. Other switch models have a *single-forwarding database*, which means they allow only one database entry of a unique MAC address, along with the source VLAN and source port on which it is found (see Table 2-6). Not all VLANs on a switch covered by this guide use the same MAC address (see "VLAN MAC Addresses" on page 2-35). Connecting multiple-forwarding database switch to a single-forwarding database switch where multiple VLANs exist imposes some cabling and port VLAN assignment restrictions. Table 2-6 illustrates the functional difference between the two database types.

Table 2-6. Example of Forwarding Database Content

Multiple-Forwarding Database			Single-Forwarding Database		
MAC Address	**Destination VLAN ID**	**Destination Port**	**MAC Address**	**Destination VLAN ID**	**Destination Port**
0004ea-84d9f4	1	A5	0004ea-84d9f4	100	A9
0004ea-84d9f4	22	A12	0060b0-880af9	105	A10
0004ea-84d9f4	44	A20	0060b0-880a81	107	A17
0060b0-880a81	33	A20			
This database allows multiple destinations for the same MAC address. If the switch detects a new destination for an existing MAC entry, it just **adds** a new instance of that MAC to the table.			This database allows only one destination for a MAC address. If the switch detects a new destination for an existing MAC entry, it **replaces** the existing MAC instance with a new instance showing the new destination.		

Table 2-7 lists the database structure of current ProCurve switch models.

Table 2-7. Forwarding Database Structure for Managed ProCurve Switches

Multiple-Forwarding Databases*	Single-Forwarding Database*
Switch 6108	Switch 1600M/2400M/2424M
Series 5300xl switches	Switch 4000M/8000M
Series 4100gl switches	Series 2500 switches
Series 3400cl switches	Switch 800T
Switch 2810	Switch 2000
Series 2800 switches	x
Series 2600/2600-PWR switches	

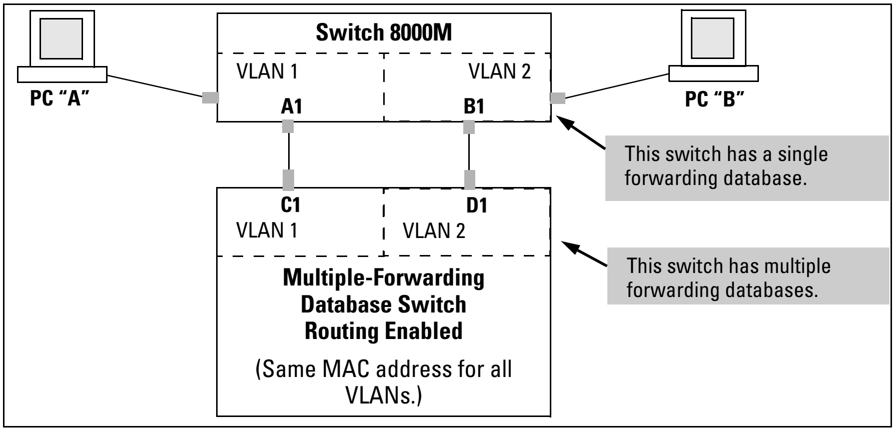
*To determine whether other vendors' devices use single-forwarding or multiple-forwarding database architectures, refer to the documentation provided for those devices.

Single-Forwarding Database Operation

When a packet arrives with a destination MAC address that matches a MAC address in the switch's forwarding table, the switch tries to send the packet to the port listed for that MAC address. But, if the destination port is in a different VLAN than the VLAN on which the packet was received, the switch drops the packet. This is not a problem for a switch with a multiple-forwarding database (refer to table 2-7, above) because the switch allows multiple instances of a given MAC address; one for each valid destination. However, a switch with a single-forwarding database allows only one instance of a given MAC address. If (1) you connect the two types of switches through multiple ports or trunks belonging to different VLANs, and (2) enable routing on the switch having the multiple-forwarding database; then, on the switch having the single-forwarding database, the port and VLAN record it maintains for the connected multiple-forwarding-database switch can frequently change. This causes poor performance and the appearance of an intermittent or broken connection.

Example of an Unsupported Configuration and How to Correct It

The Problem. In figure 2-1, the MAC address table for Switch 8000M will sometimes record the multiple-forwarding database switch as accessed on port A1 (VLAN 1), and other times as accessed on port B1 (VLAN 2):


Switch 8000M
VLAN 1
VLAN 2
A1
B1
PC "A"
PC "B"
C1
D1
VLAN 1
VLAN 2
Multiple-Forwarding Database Switch Routing Enabled
(Same MAC address for all VLANs.)

This switch has a single forwarding database.

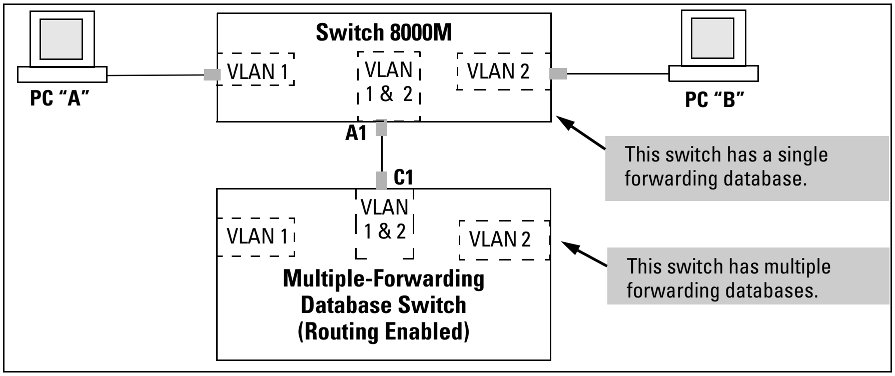
This switch has multiple forwarding databases.

Figure 2-1. Example of Invalid Configuration for Single-Forwarding to Multiple-Forwarding Database Devices in a Multiple VLAN Environment

In figure 2-1, PC "A" sends an IP packet to PC "B".

1. The packet enters VLAN 1 in the Switch 8000 with the multiple-forwarding database switch MAC address in the destination field. Because the 8000M has not yet learned this MAC address, it does not find the address in its address table, and floods the packet out all ports, including the VLAN 1 link (port "A1") to the multiple-forwarding database switch. The multiple-forwarding database switch then routes the packet through the VLAN 2 link to the 8000M, which forwards the packet on to PC "B". Because the 8000M received the packet from the multiple-forwarding database switch on VLAN 2 (port "B1"), the 8000M's single forwarding database records the multiple-forwarding database switch as being on port "B1" (VLAN 2).
2. PC "A" now sends a second packet to PC "B". The packet again enters VLAN 1 in the Switch 8000 with the multiple-forwarding database switch's MAC address in the destination field. However, this time the Switch 8000M's single forwarding database indicates that the multiple-forwarding database switch is on port B1 (VLAN 2), and the 8000M drops the packet instead of forwarding it.
3. Later, the multiple-forwarding database switch transmits a packet to the 8000M through the VLAN 1 link, and the 8000M updates its address table to indicate that the multiple-forwarding database switch is on port A1 (VLAN 1) instead of port B1 (VLAN 2). Thus, the 8000M's information on the location of the multiple-forwarding database switch changes over time. For this reason, the 8000M discards some packets directed through it for the multiple-forwarding database switch, resulting in poor performance and the appearance of an intermittent or broken link.

The Solution. To avoid the preceding problem, use only one cable or port trunk between the single-forwarding and multiple-forwarding database devices, and configure the link with multiple, tagged VLANs.



This switch has a single forwarding database.

This switch has multiple forwarding databases.

Figure 2-2. Example of a Solution for Single-Forwarding to Multiple-Forwarding Database Devices in a Multiple VLAN Environment

Now, the 8000M forwarding database always lists the multiple-forwarding database switch MAC address on port A1, and the 8000M will send traffic to either VLAN on the multiple-forwarding database switch.

To increase the network bandwidth of the connection between the devices, you can use a trunk of multiple physical links rather than a single physical link.

Multiple-Forwarding Database Operation

If you want to connect a switch covered by this guide to another switch that has a multiple-forwarding database, you can use either or both of the following connection options:

- A separate port or port trunk interface for each VLAN. This results in a forwarding database having multiple instances of the same MAC address with different VLAN IDs and port numbers. (See table 2-6.) The switches covered by this guide that use the same MAC address on all VLAN interfaces cause no problems.
- The same port or port trunk interface for multiple (tagged) VLANs. This results in a forwarding database having multiple instances of the same MAC address with different VLAN IDs, but the same port number.

Allowing multiple entries of the same MAC address on different VLANs enables topologies such as the following:

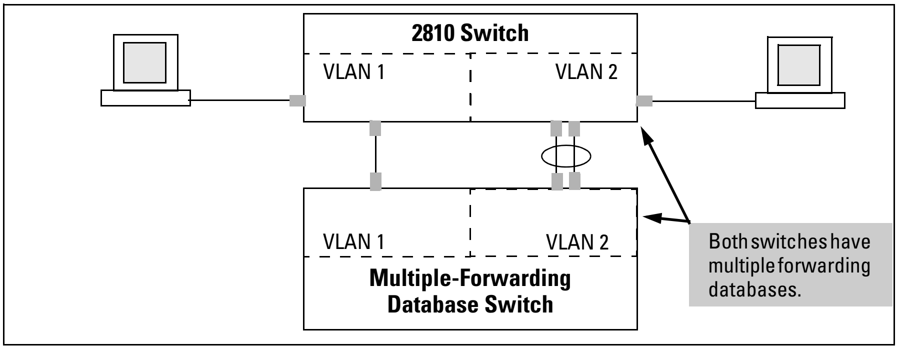

2810 Switch
VLAN 1
VLAN 2
VLAN 1
VLAN 2
Multiple-Forwarding Database Switch
Both switches have multiple forwarding databases.

Figure 2-3. Example of a Valid Topology for Devices Having Multiple-Forwarding Databases in a Multiple VLAN Environment

Menu: Configuring VLAN Parameters

In the factory default state, support is enabled for up to eight VLANs. (You can change the switch VLAN configuration to support additional VLANs. Also, all ports on the switch belong to the default VLAN (DEFAULT_VLAN) and are in the same broadcast/multicast domain. (The default VLAN is also the default primary VLAN—see “The Primary VLAN” on page 2-6.) In addition to the default VLAN, you can configure up to 29 other static VLANs by changing the “Maximum VLANs” parameter, adding new VLAN names and VIDs, and then assigning one or more ports to each VLAN. Note that each port can be assigned to multiple VLANs by using VLAN tagging. (See “802.1Q VLAN Tagging” on page 2-26.)

To Change VLAN Support Settings

This section describes:

- Changing the maximum number of VLANs to support
- Changing the primary VLAN selection (See “Changing the Primary VLAN” on page 2-22.)

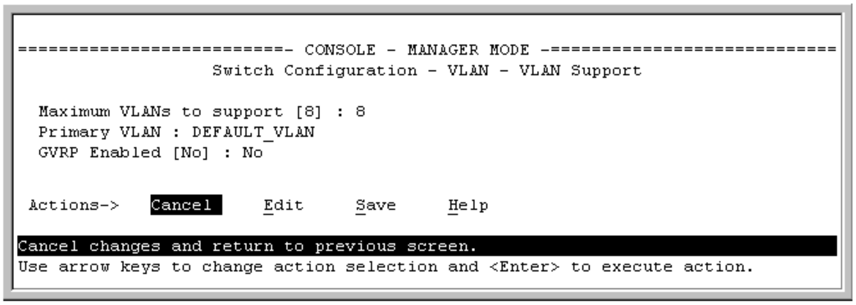
1. From the Main Menu select:

   2. Switch Configuration

   8. VLAN Menu . . .

   1. VLAN Support

   You will then see the following screen:



```
==========================- CONSOLE - MANAGER MODE -============================
                  Switch Configuration - VLAN - VLAN Support

  Maximum VLANs to support [8] : 8
  Primary VLAN : DEFAULT_VLAN
  GVRP Enabled [No] : No


 Actions->   Cancel     Edit     Save     Help

Cancel changes and return to previous screen.
Use arrow keys to change action selection and <Enter> to execute action.
```

Figure 2-8. The Default VLAN Support Screen

2. Press **[E]** (for **Edit)**, then do one or more of the following:

- To change the maximum number of VLANs, type the new number.
- To designate a different VLAN as the primary VLAN, select the **Primary VLAN** field and use the space bar to select from the existing options.

- To enable or disable dynamic VLANs, select the **GVRP Enabled** field and use the Space bar to toggle between options. (For GVRP information, see chapter 3, "GVRP".)

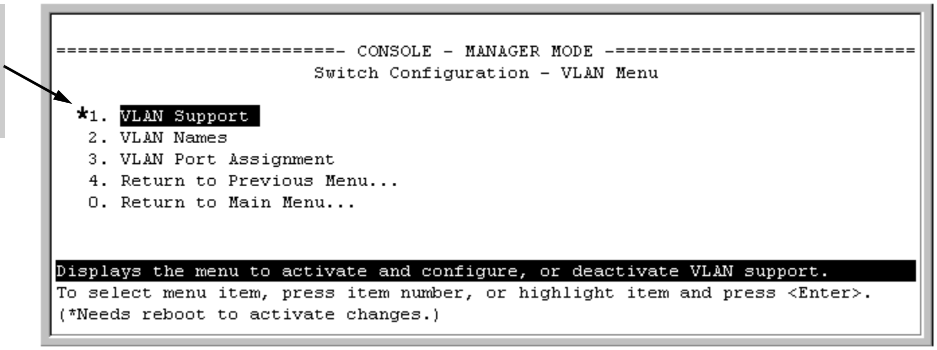
Note

For optimal switch memory utilization, set the number of VLANs at the number you will likely be using or a few more. If you need more VLANs later, you can increase this number, but a switch reboot will be required at that time.

3. Press **[Enter]** and then **[S]** to save the VLAN support configuration and return to the VLAN Menu screen.

   If you changed the value for **Maximum VLANs to support**, you will see an asterisk next to the **VLAN Support** option (see below).

An asterisk indicates you must reboot the switch to implement the new Maximum VLANs setting.



```
==========================- CONSOLE - MANAGER MODE -============================
                          Switch Configuration - VLAN Menu

  *1. VLAN Support
   2. VLAN Names
   3. VLAN Port Assignment
   4. Return to Previous Menu...
   0. Return to Main Menu...


Displays the menu to activate and configure, or deactivate VLAN support.
To select menu item, press item number, or highlight item and press <Enter>.
(*Needs reboot to activate changes.)
```

Figure 2-9. VLAN Menu Screen Indicating the Need To Reboot the Switch

   – If you changed the VLAN Support option, you must reboot the switch before the Maximum VLANs change can take effect. You can go on to configure other VLAN parameters first, but remember to reboot the switch when you are finished.
   – If you did not change the VLAN Support option, a reboot is not necessary.

4. Press **[0]** to return to the Main Menu.

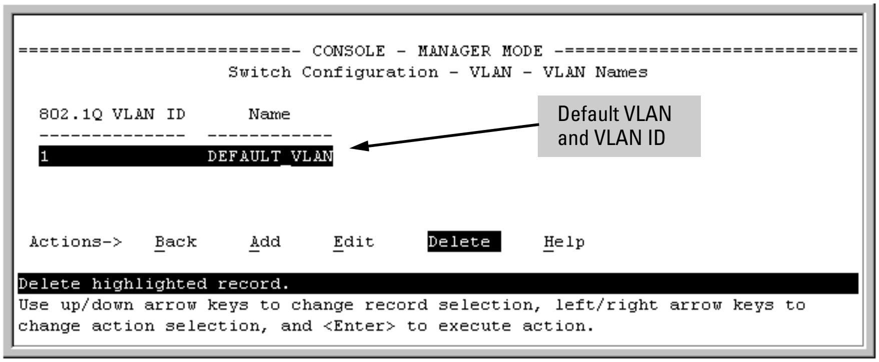
Adding or Editing VLAN Names

Use this procedure to add a new VLAN or to edit the name of an existing VLAN.

1. From the Main Menu select:

   2. Switch Configuration
   8. VLAN Menu . . .
   2. VLAN Names

   If multiple VLANs are not yet configured you will see a screen similar to figure 2-10:



```
=========================- CONSOLE - MANAGER MODE -===========================
                  Switch Configuration - VLAN - VLAN Names

  802.1Q VLAN ID      Name
  --------------   ------------
  1                DEFAULT_VLAN

 Actions->   Back     Add     Edit     Delete     Help

Delete highlighted record.
Use up/down arrow keys to change record selection, left/right arrow keys to
change action selection, and <Enter> to execute action.
```

Figure 2-10. The Default VLAN Names Screen

2. Press **[A]** (for **Add**). You will then be prompted for a new VLAN name and VLAN ID:

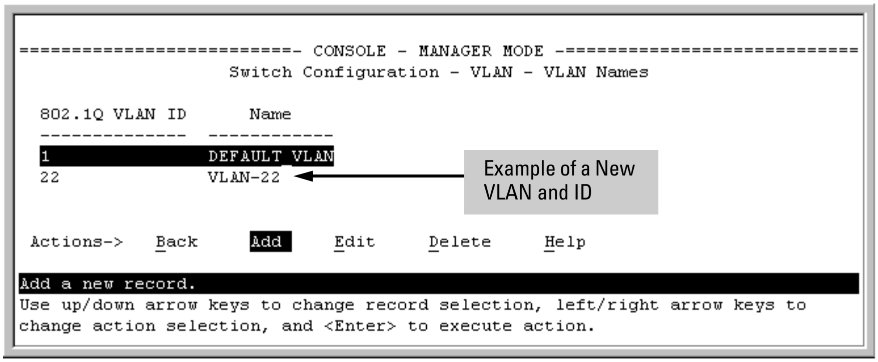
   802.1Q VLAN ID : 1
   Name : _

3. Type in a VID (VLAN ID number). This can be any number from 2 to 4094 that is not already being used by another VLAN. (The switch reserves “1” for the default VLAN.)

   Remember that a VLAN *must* have the same VID in every switch in which you configure that same VLAN. (GVRP dynamically extends VLANs with correct VID numbering to other switches. See chapter 3, “GVRP”.)

4. Press [↓] to move the cursor to the **Name** line and type the VLAN name (up to 12 characters, with no spaces) of a new VLAN that you want to add, then press **[Enter]**.
   (Avoid these characters in VLAN names: **@**, **#**, **$**, **^**, **&**, *****, **(**, and **)**.)

5. Press **[S]** (for **Save**). You will then see the VLAN Names screen with the new VLAN listed.



```
==========================- CONSOLE - MANAGER MODE -============================
                    Switch Configuration - VLAN - VLAN Names

  802.1Q VLAN ID      Name
  --------------  ------------
  1               DEFAULT_VLAN
  22              VLAN-22

 Actions->   Back     Add     Edit     Delete     Help

Add a new record.
Use up/down arrow keys to change record selection, left/right arrow keys to
change action selection, and <Enter> to execute action.
```

Example of a New VLAN and ID

Figure 2-11. Example of VLAN Names Screen with a New VLAN Added

6. Repeat steps 2 through 5 to add more VLANs.

   Remember that you can add VLANs until you reach the number specified in the **Maximum VLANs to support** field on the VLAN Support screen (see figure 2-8 on page 2-14). This includes any VLANs added dynamically due to GVRP operation.

7. Return to the VLAN Menu to assign ports to the new VLAN(s) as described in the next section, “Adding or Changing a VLAN Port Assignment”.

Adding or Changing a VLAN Port Assignment

Use this procedure to add ports to a VLAN or to change the VLAN assignment(s) for any port. (Ports not specifically assigned to a VLAN are automatically in the default VLAN.)

1. From the Main Menu select:

   2. Switch Configuration

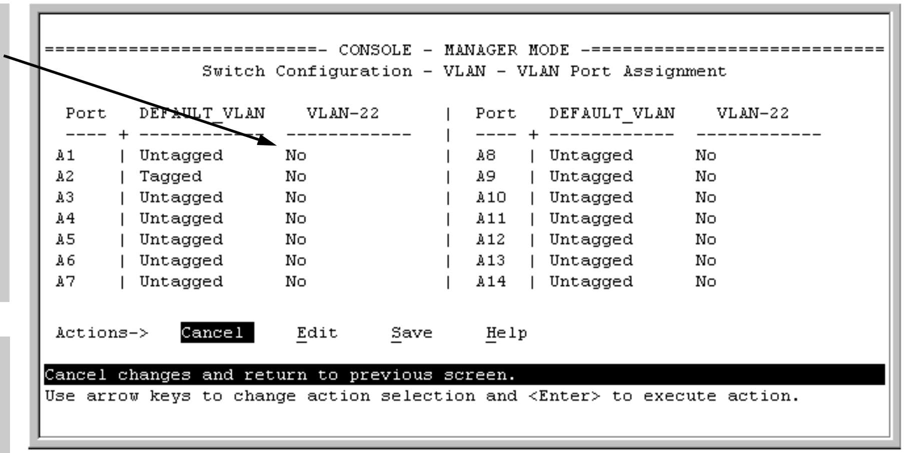
   8. VLAN Menu . . .

   3. VLAN Port Assignment

   You will then see a VLAN Port Assignment screen similar to the following:

Default: In this example, the "VLAN-22" has been defined, but no ports have yet been assigned to it. ("No" means the port is not assigned to that VLAN.)
Using GVRP? If you plan on using GVRP, any ports you don't want to join should be changed to "Forbid".

A port can be assigned to several VLANs, but only one of those assignments can be "Untagged".



```
=========================- CONSOLE - MANAGER MODE -===========================
               Switch Configuration - VLAN - VLAN Port Assignment

  Port   DEFAULT_VLAN    VLAN-22      |  Port   DEFAULT_VLAN    VLAN-22
  ---- + ------------  ------------   |  ---- + ------------  ------------
 A1    | Untagged      No             |  A8   | Untagged      No
 A2    | Tagged        No             |  A9   | Untagged      No
 A3    | Untagged      No             |  A10  | Untagged      No
 A4    | Untagged      No             |  A11  | Untagged      No
 A5    | Untagged      No             |  A12  | Untagged      No
 A6    | Untagged      No             |  A13  | Untagged      No
 A7    | Untagged      No             |  A14  | Untagged      No

 Actions->  Cancel     Edit     Save     Help

Cancel changes and return to previous screen.
Use arrow keys to change action selection and <Enter> to execute action.
```

Figure 2-12. Example of VLAN Port Assignment Screen

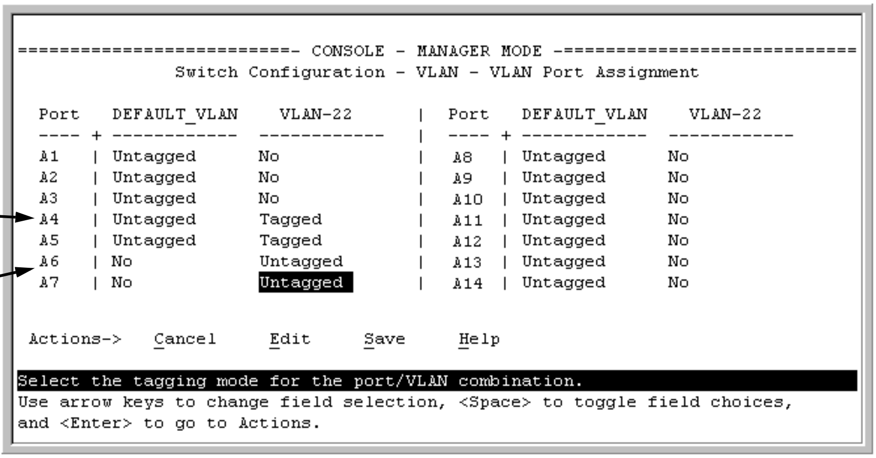
2. To change a port's VLAN assignment(s):
   a. Press **[E]** (for **Edit**).
   b. Use the arrow keys to select a VLAN assignment you want to change.
   c. Press the Space bar to make your assignment selection (**No**, **Tagged**, **Untagged**, or **Forbid**).

Note

For GVRP Operation: If you enable GVRP on the switch, "**No**" converts to "**Auto**", which allows the VLAN to dynamically join an advertised VLAN that has the same VID. See "Per-Port Options for Dynamic VLAN Advertising and Joining" on page 3-8.

Untagged VLANs: Only one untagged VLAN is allowed per port. Also, there must be at least one VLAN assigned to each port. In the factory default configuration, all ports are assigned to the default VLAN (DEFAULT_VLAN).

For example, if you want ports A4 and A5 to belong to both DEFAULT_VLAN and VLAN-22, and ports A6 and A7 to belong only to VLAN-22, you would use the settings in figure page 2-19. (This example assumes the default GVRP setting—disabled—and that you do not plan to enable GVRP later.)



Ports A4 and A5 are assigned to both VLANs.

Ports A6 and A7 are assigned only to VLAN-22.

All other ports are assigned only to the Default VLAN.

```
==========================- CONSOLE - MANAGER MODE -============================
               Switch Configuration - VLAN - VLAN Port Assignment

  Port   DEFAULT_VLAN    VLAN-22       |  Port   DEFAULT_VLAN    VLAN-22
  ---- + ------------  ------------    |  ---- + ------------  ------------
  A1   | Untagged      No              |  A8   | Untagged      No
  A2   | Untagged      No              |  A9   | Untagged      No
  A3   | Untagged      No              |  A10  | Untagged      No
  A4   | Untagged      Tagged          |  A11  | Untagged      No
  A5   | Untagged      Tagged          |  A12  | Untagged      No
  A6   | No            Untagged        |  A13  | Untagged      No
  A7   | No            Untagged        |  A14  | Untagged      No


 Actions->   Cancel     Edit     Save     Help

Select the tagging mode for the port/VLAN combination.
Use arrow keys to change field selection, <Space> to toggle field choices,
and <Enter> to go to Actions.
```

Figure 2-13. Example of VLAN Assignments for Specific Ports

For information on VLAN tags ("Untagged" and "Tagged"), refer to "802.1Q VLAN Tagging" on page 2-26.

d. If you are finished assigning ports to VLANs, press **[Enter]** and then **[S]** (for **Save**) to activate the changes you've made and to return to the Configuration menu. (The console then returns to the VLAN menu.)

3. Return to the Main menu.

CLI: Configuring VLAN Parameters

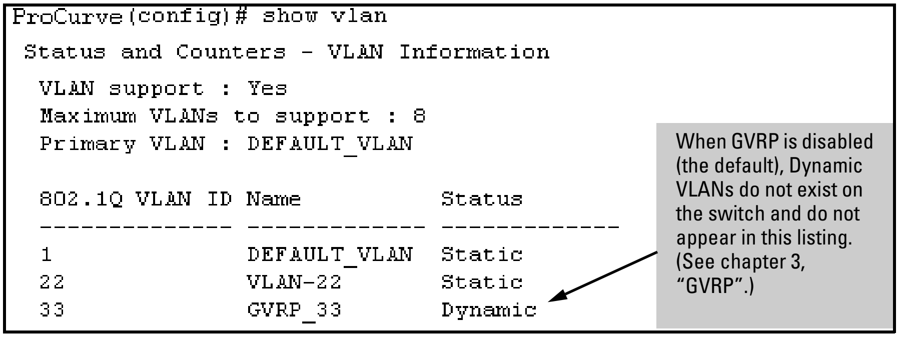
In the factory default state, all ports on the switch belong to the default VLAN (DEFAULT_VLAN) and are in the same broadcast/multicast domain. (The default VLAN is also the default primary VLAN—see "The Primary VLAN" on page 2-6.) You can configure additional static VLANs by adding new VLAN names, and then assigning one or more ports to each VLAN. Note that each port can be assigned to multiple VLANs by using VLAN tagging. (See "802.1Q VLAN Tagging" on page 2-26.)

VLAN Commands Used in this Section

show vlans	below
show vlan <*vlan-id*>	page 2-21
max-vlans	page 2-22
primary-vlan <*vlan-id*>	page 2-22
[no] vlan <*vlan-id*>	page 2-23
name <*vlan-name*>	page 2-24
[no] tagged <*port-list*>	page 2-24
[no] untagged <*port-list*>	page 2-24
[no] forbid	page 2-24
auto <*port-list*>	page 2-24 (Available if GVRP enabled.)
static-vlan <*vlan-id*>	page 2-23 (Available if GVRP enabled.)

Displaying the Switch's VLAN Configuration. The next command lists the VLANs currently running in the switch, with VID, VLAN name, and VLAN status. Dynamic VLANs appear only if the switch is running with GVRP enabled and one or more ports has dynamically joined an advertised VLAN. (In the default configuration, GVRP is disabled. (See chapter 3, "GVRP".)

Syntax: show vlan



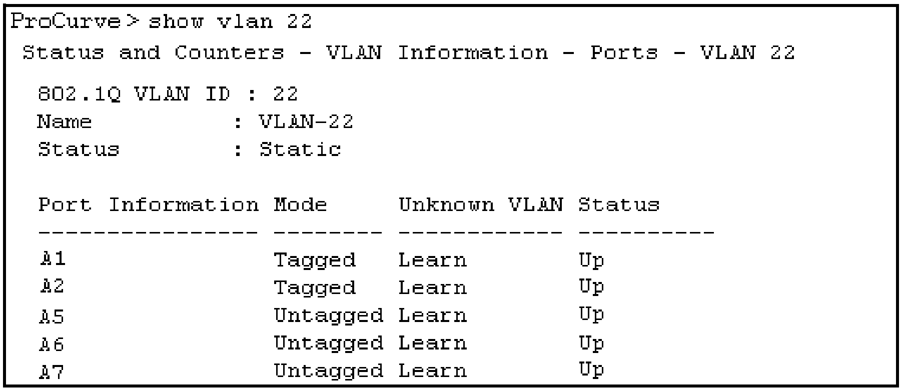
```
ProCurve(config)# show vlan
 Status and Counters - VLAN Information
  VLAN support : Yes
  Maximum VLANs to support : 8
  Primary VLAN : DEFAULT_VLAN

  802.1Q VLAN ID Name          Status
  -------------- ------------- -------------
  1              DEFAULT_VLAN  Static
  22             VLAN-22       Static
  33             GVRP_33       Dynamic
```

When GVRP is disabled (the default), Dynamic VLANs do not exist on the switch and do not appear in this listing. (See chapter 3, "GVRP".)

Figure 2-14. Example of "Show VLAN" Listing (GVRP Enabled)

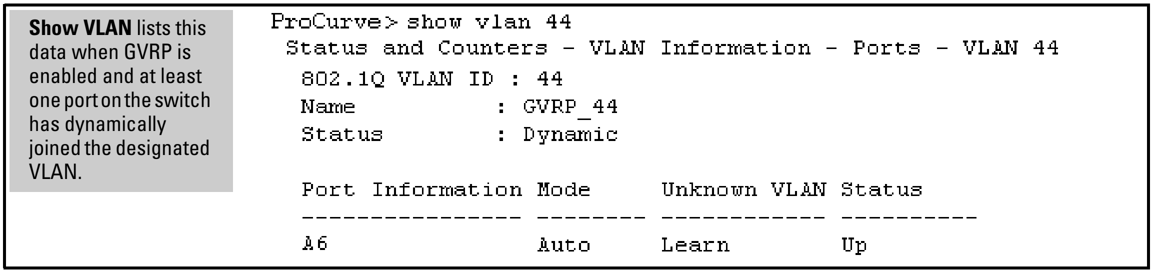
Displaying the Configuration for a Particular VLAN. This command uses the VID to identify and display the data for a specific static or dynamic VLAN.

Syntax: show vlan <*vlan-id*>



```
ProCurve> show vlan 22
 Status and Counters - VLAN Information - Ports - VLAN 22

  802.1Q VLAN ID : 22
  Name           : VLAN-22
  Status         : Static

  Port Information Mode     Unknown VLAN Status
  ---------------- -------- ------------ ----------
  A1               Tagged   Learn        Up
  A2               Tagged   Learn        Up
  A5               Untagged Learn        Up
  A6               Untagged Learn        Up
  A7               Untagged Learn        Up
```

Figure 2-15. Example of "Show VLAN" for a Specific Static VLAN



Show VLAN lists this data when GVRP is enabled and at least one port on the switch has dynamically joined the designated VLAN.

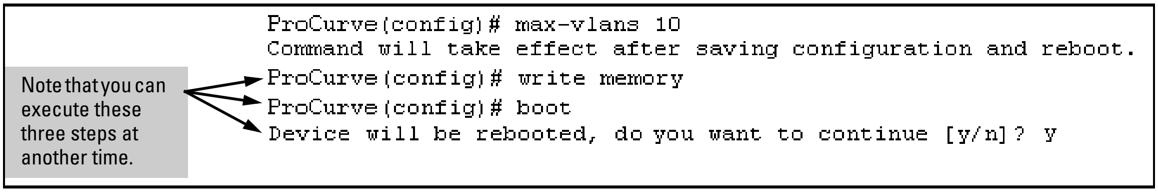
```
ProCurve> show vlan 44
 Status and Counters - VLAN Information - Ports - VLAN 44
  802.1Q VLAN ID : 44
  Name           : GVRP_44
  Status         : Dynamic

  Port Information Mode     Unknown VLAN Status
  ---------------- -------- ------------ ----------
  A6               Auto     Learn        Up
```

Figure 2-16. Example of "Show VLAN" for a Specific Dynamic VLAN

Changing the Number of VLANs Allowed on the Switch. By default, the switch allows a maximum of 8 VLANs. You can specify any value from 1 to the upper limit for the switch. If GVRP is enabled, this setting includes any dynamic VLANs on the switch. As part of implementing a new value, you must execute a write memory command (to save the new value to the startup-config file) and then reboot the switch.

Syntax: max-vlans <1... 256>

For example, to reconfigure the switch to allow 10 VLANs:



```
ProCurve(config)# max-vlans 10
Command will take effect after saving configuration and reboot.
ProCurve(config)# write memory
ProCurve(config)# boot
Device will be rebooted, do you want to continue [y/n]? y
```

Note that you can execute these three steps at another time.

Figure 2-17. Example of Command Sequence for Changing the Number of VLANs

Changing the Primary VLAN. In the factory-default configuration, the default VLAN (DEFAULT_VLAN) is the primary VLAN. However, you can designate any static VLAN on the switch as the primary VLAN. (For more on the primary VLAN, see "The Primary VLAN" on page 2-6.) To view the available VLANs and their respective VIDs, use **show vlan**.

Syntax: primary-vlan <*vlan-id*>

For example, to make VLAN 22 the primary VLAN:

```
ProCurve(config)# primary-vlan 22
```

Creating a New Static VLAN
Changing the VLAN Context Level.

With this command, entering a new VID creates a new static VLAN. Entering the VID or name of an existing static VLAN places you in the context level for that VLAN.

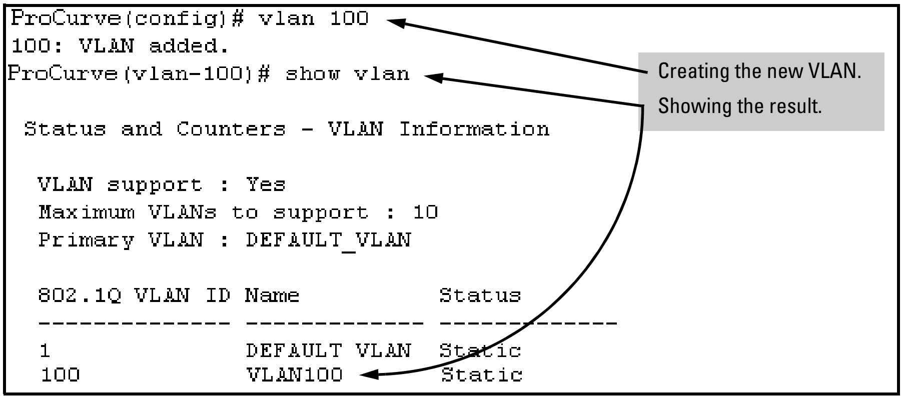
Syntax: vlan <*vlan-id*> [name <*name-str*>]

Creates a new static VLAN if a VLAN with that VID does not already exist, and places you in that VLAN's context level. If you do not use the name option, the switch uses "VLAN" and the new VID to automatically name the VLAN. If the VLAN already exists, the switch places you in the context level for that VLAN.

vlan <*vlan-name*>

Places you in the context level for that static VLAN.

For example, to create a new static VLAN with a VID of 100:



```
ProCurve(config)# vlan 100
100: VLAN added.
ProCurve(vlan-100)# show vlan

 Status and Counters - VLAN Information

  VLAN support : Yes
  Maximum VLANs to support : 10
  Primary VLAN : DEFAULT_VLAN

  802.1Q VLAN ID Name          Status
  -------------- ------------- -------------
  1              DEFAULT VLAN  Static
  100            VLAN100       Static
```

Creating the new VLAN.

Showing the result.

Figure 2-18. Example of Creating a New Static VLAN

To go to a different VLAN context level, such as to the default VLAN:

```
ProCurve(vlan-100)# vlan default_vlan
ProCurve(vlan-1) _
```

Converting a Dynamic VLAN to a Static VLAN. If GVRP is running on the switch and a port dynamically joins a VLAN, you can use the next command to convert the dynamic VLAN to a static VLAN. (For GVRP and dynamic VLAN operation, see chapter 3, "GVRP".) This is necessary if you want to make the VLAN permanent. After you convert a dynamic VLAN to static, you must configure the switch's per-port participation in the VLAN in the same way that you would for any static VLAN.

Syntax: static-vlan <*vlan-id*> (*Use* **show vlan** *to list current VIDs.*)

For example, suppose a dynamic VLAN with a VID of 125 exists on the switch. The following command converts the VLAN to a static VLAN.

```
ProCurve(config)# static-vlan 125
```

Configuring Static VLAN Name and Per-Port Settings. The **vlan <*vlan-id*>** command, used with the options listed below, changes the name of an existing static VLAN and changes the per-port VLAN membership settings.

Note

You can use these options from the configuration level by beginning the command with **vlan <*vlan-id*>**, or from the context level of the specific VLAN.

Syntax: name <*vlan-name*>

Changes the name of the existing static VLAN. (Avoid spaces and the following characters in the <vlan-name> entry: **2**, **#**, **$**, **^**, **&**, *, **(**, *and* **)**.)

[no] tagged <*port-list*>

Configures the indicated port(s) as **Tagged** *for the specified VLAN. The "***no***" version sets the port(s) to either* **No** *or (if GVRP is enabled) to* **Auto**.

[no] untagged <*port-list*>

Configures the indicated port(s) as **Untagged** *for the specified VLAN. The "***no***" version sets the port(s) to either* **No** *or (if GVRP is enabled) to* **Auto**.

[no] forbid <*port-list*>

*Configures the indicated port(s) as "forbidden" to participate in the designated VLAN. The "***no***" version sets the port(s) to either* **No** *or (if GVRP is enabled) to* **Auto**.

auto <*port-list*>

Available if GVRP is enabled on the switch. Returns the per-port settings for the specified VLAN to ***Auto*** *operation. Note that* ***Auto*** *is the default per-port setting for a static VLAN if GVRP is running on the switch. (For information on dynamic VLAN and GVRP operation, see* chapter 3, "GVRP".*)*

For example, if you have a VLAN named VLAN100 with a VID of 100, and all ports are set to **No** for this VLAN. To change the VLAN name to “Blue_Team” and set ports 1-5 to Tagged, you could do so with these commands:

```
ProCurve(config)# vlan 100 name Blue_Team
ProCurve(config)# vlan 100 tagged 1-5
```

To move to the vlan 100 context level and execute the same commands:

```
ProCurve(config)# vlan 100
ProCurve(vlan-100)# name Blue_Team
ProCurve(vlan-100)# tagged 1-5
```

Similarly, to change the tagged ports in the above examples to **No** (or **Auto**, if GVRP is enabled), you could use either of the following commands.

At the config level, use:

```
  ProCurve(config)# no vlan 100 tagged 1-5
```

- *or* -

At the VLAN 100 context level, use:

```
 ProCurve(vlan-100)# no tagged 1-5
```

Note

You cannot use these commands with dynamic VLANs. Attempting to do so results in the message “**VLAN already exists.**” and no change occurs.

Web: Viewing and Configuring VLAN Parameters

In the web browser interface you can do the following:

- Add VLANs
- Rename VLANs
- Remove VLANs
- Configure GVRP mode
- Select a new Primary VLAN

To configure static VLAN port parameters, you will need to use the menu interface (available by Telnet from the web browser interface) or the CLI.

1. Click on the **Configuration** tab.
2. Click on **VLAN Configuration**.
3. Click on **Add/Remove VLANs**.

For web-based Help on how to use the web browser interface screen, click on the **[?]** button provided on the web browser screen.

802.1Q VLAN Tagging

VLAN tagging enables traffic from more than one VLAN to use the same port. (Even when two or more VLANs use the same port they remain as separate domains and cannot receive traffic from each other without going through an external router.) As mentioned earlier, a "tag" is simply a unique VLAN identification number (VLAN ID, or VID) assigned to a VLAN at the time that you configure the VLAN name in the switch. The tag can be any number from 1 to 4094 that is not already assigned to a VLAN. When you subsequently assign a port to a given VLAN, you must implement the VLAN tag (VID) if the port will carry traffic for more than one VLAN. Otherwise, the port VLAN assignment can remain "untagged" because the tag is not needed. On a given switch, this means you should use the "Untagged" designation for a port VLAN assignment where the port is connected to non 802.1Q-compliant device or is assigned to only one VLAN. Use the "Tagged" designation when the port is assigned to more than one VLAN or the port is connected to a device that *does* comply with the 802.1Q standard.

For example, if port A7 on an 802.1Q-compliant switch is assigned to only the Red VLAN, the assignment can remain "untagged" because the port will forward traffic only for the Red VLAN. However, if both the Red and Green VLANs are assigned to port A7, then at least one of those VLAN assignments must be "tagged" so that Red VLAN traffic can be distinguished from Green VLAN traffic. The following illustration shows this concept:

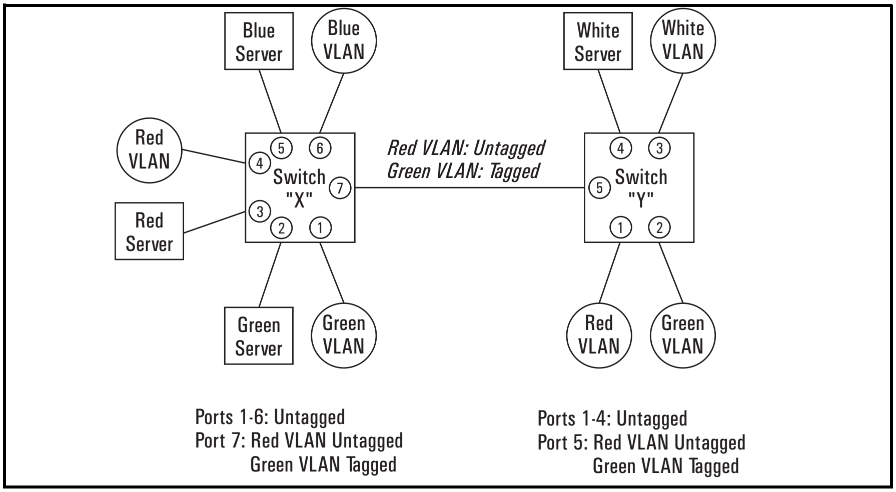
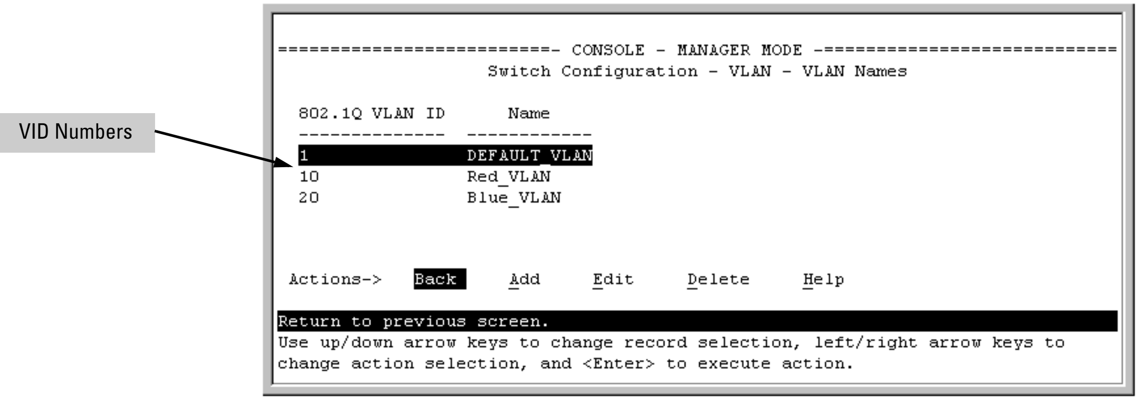


Figure 2-19. Example of Tagged and Untagged VLAN Port Assignments

- In switch X:
  - VLANs assigned to ports X1 - X6 can all be untagged because there is only one VLAN assignment per port. Red VLAN traffic will go out only the Red ports; Green VLAN traffic will go out only the Green ports, and so on. Devices connected to these ports do not have to be 802.1Q-compliant.
  - However, because both the Red VLAN and the Green VLAN are assigned to port X7, at least one of the VLANs must be tagged for this port.
- In switch Y:
  - VLANs assigned to ports Y1 - Y4 can all be untagged because there is only one VLAN assignment per port. Devices connected to these ports do not have to be 802.1Q-compliant.
  - Because both the Red VLAN and the Green VLAN are assigned to port Y5, at least one of the VLANs must be tagged for this port.
- In both switches: The ports on the link between the two switches must be configured the same. As shown in figure 2-19 (above), the Red VLAN must be untagged on port X7 and Y5 and the Green VLAN must be tagged on port X7 and Y5, or vice-versa.

Note

Each 802.1Q-compliant VLAN must have its own unique VID number, and that VLAN *must* be given the same VID in every device in which it is configured. That is, if the Red VLAN has a VID of 10 in switch X, then 10 must also be used for the Red VID in switch Y.



```
==========================- CONSOLE - MANAGER MODE -============================
                     Switch Configuration - VLAN - VLAN Names

  802.1Q VLAN ID      Name
  --------------  ------------
  1               DEFAULT_VLAN
  10              Red_VLAN
  20              Blue_VLAN


 Actions->   Back     Add     Edit     Delete     Help

Return to previous screen.
Use up/down arrow keys to change record selection, left/right arrow keys to
change action selection, and <Enter> to execute action.
```

Figure 2-20. Example of VLAN ID Numbers Assigned in the VLAN Names Screen

VLAN tagging gives you several options:

- Since the purpose of VLAN tagging is to allow multiple VLANs on the same port, any port that has only one VLAN assigned to it can be configured as "Untagged" (the default).
- Any port that has two or more VLANs assigned to it can have one VLAN assignment for that port as "Untagged". All other VLANs assigned to the same port must be configured as "Tagged". (There can be no more than one Untagged VLAN on a port.)
- If all end nodes on a port comply with the 802.1Q standard and are configured to use the correct VID, then, you can configure all VLAN assignments on a port as "Tagged" if doing so makes it easier to manage your VLAN assignments, or for security reasons.

For example, in the following network, switches X and Y and servers S1 and S2 are 802.1Q-compliant. (Server S3 could also be 802.1Q-compliant, but it makes no difference for this example.)

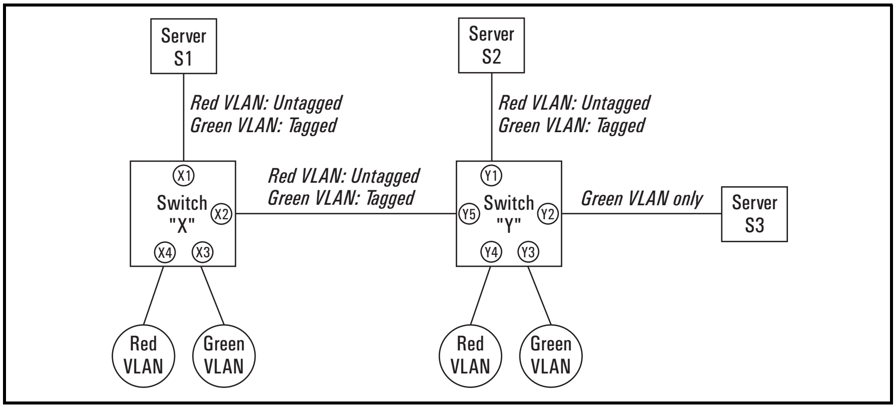


Figure 2-21. Example of Networked 802.1Q-Compliant Devices with Multiple VLANs on Some Ports

The VLANs assigned to ports X3, X4, Y2, Y3, and Y4 can all be untagged because there is only one VLAN assigned per port. Port X1 has multiple VLANs assigned, which means that one VLAN assigned to this port can be untagged and any others must be tagged. The same applies to ports X2, Y1, and Y5.

Switch X				**Switch Y**		
Port	**Red VLAN**	**Green VLAN**		**Port**	**Red VLAN**	**Green VLAN**
X1	Untagged	Tagged		Y1	Untagged	Tagged
X2	Untagged	Tagged		Y2	No*	Untagged
X3	No*	Untagged		Y3	No*	Untagged
X4	Untagged	No*		Y4	Untagged	No*
				Y5	Untagged	Tagged

*"No" means the port is not a member of that VLAN. For example, port X3 is not a member of the Red VLAN and does not carry Red VLAN traffic. Also, if GVRP were enabled, "Auto" would appear instead of "No".

Note

VLAN configurations on ports connected by the same link must match. Because ports X2 and Y5 are opposite ends of the same point-to-point connection, both ports must have the same VLAN configuration; that is, both ports configure the Red VLAN as "Untagged" and the Green VLAN as "Tagged".

To summarize:

VLANs Per Port	**Tagging Scheme**
1	Untagged or Tagged. If the device connected to the port is 802.1Q-compliant, then the recommended choice is "Tagged".
2 or More	1 VLAN Untagged; all others Tagged or All VLANs Tagged

A given VLAN *must* have the same VID on any 802.1Q-compliant device in which the VLAN is configured.

The ports connecting two 802.1Q devices should have identical VLAN configurations, as shown for ports X2 and Y5, above.

The Secure Management VLAN

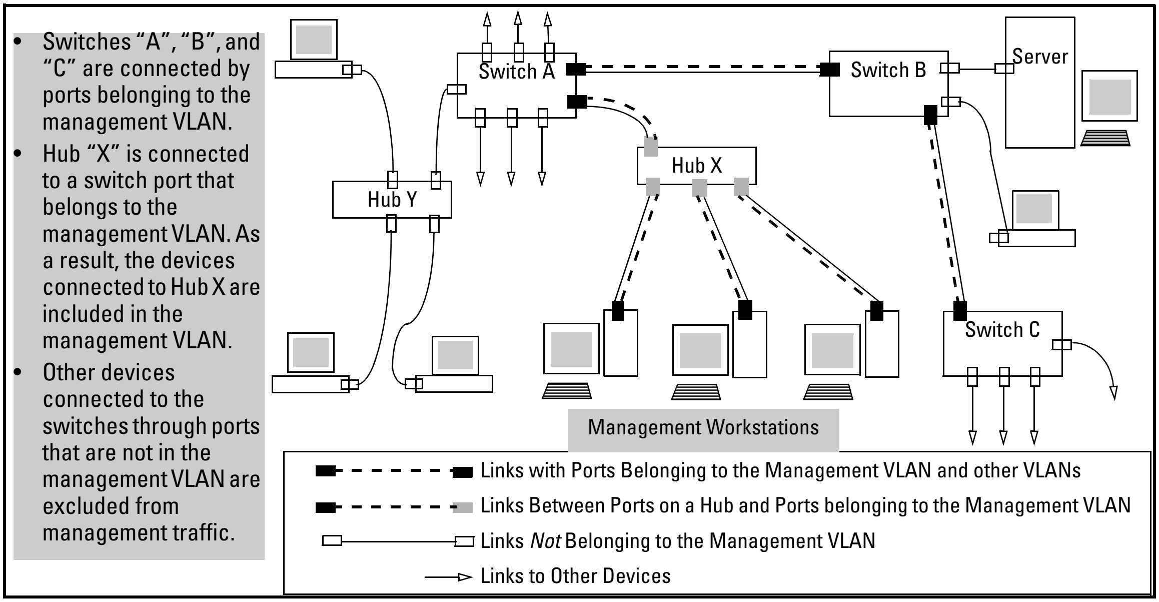
Configures a secure Management VLAN by creating an isolated network for managing the following ProCurve switches that support this feature:

- Series 2600 switches
- Series 2600-PWR switches
- Series 2800 switches
- Switch 2810
- Series 3400cl switches
- Series 4100gl switches
- Series 4200vl switches
- Series 5300xl switches
- Series 5400zl switches
- Switch 6108
- Series 6400cl switches

Access to this VLAN, and to the switch's management functions (Menu, CLI, and web browser interface) is available only through ports configured as members.

- ■ Multiple ports on the switch can belong to the Management VLAN. This allows connections for multiple management stations you want to have access to the Management VLAN, while at the same time allowing Management VLAN links between switches configured for the same Management VLAN.
- ■ Only traffic from the Management VLAN can manage the switch, which means that only the workstations and PCs connected to ports belonging to the Management VLAN can manage and reconfigure the switch.

Figure 2-22 illustrates use of the Management VLAN feature to support management access by a group of management workstations.

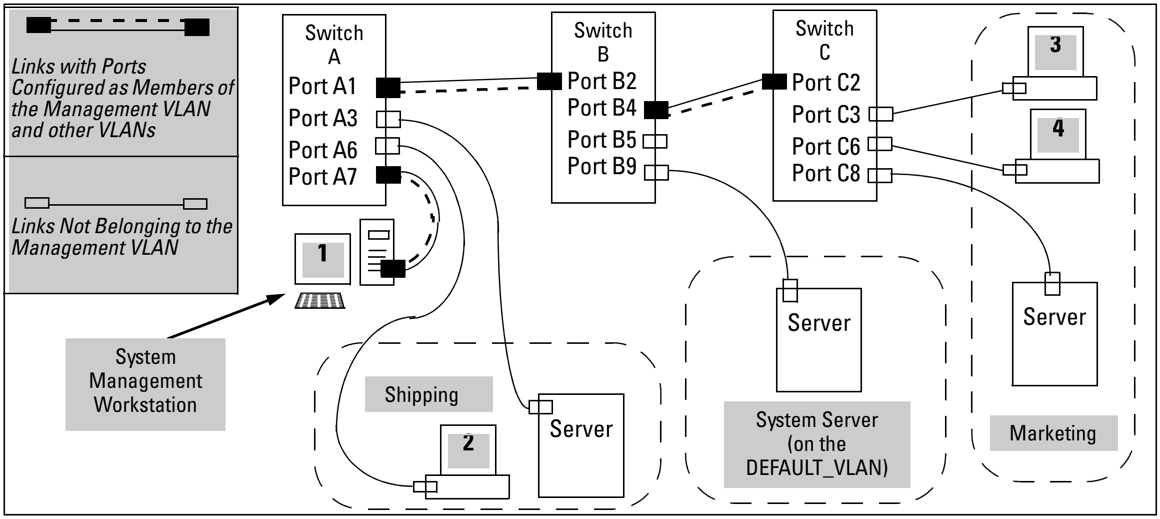


- Switches “A”, “B”, and “C” are connected by ports belonging to the management VLAN.
- Hub “X” is connected to a switch port that belongs to the management VLAN. As a result, the devices connected to Hub X are included in the management VLAN.
- Other devices connected to the switches through ports that are not in the management VLAN are excluded from management traffic.

Switch A
Switch B
Server
Hub X
Hub Y
Switch C
Management Workstations
Links with Ports Belonging to the Management VLAN and other VLANs
Links Between Ports on a Hub and Ports belonging to the Management VLAN
Links *Not* Belonging to the Management VLAN
Links to Other Devices

Figure 2-22. Example of Potential Security Breaches

In figure 2-23, Workstation 1 has management access to all three switches through the Management VLAN, while the PCs do not. This is because configuring a switch to recognize a Management VLAN automatically excludes attempts to send management traffic from any other VLAN.



Links with Ports Configured as Members of the Management VLAN and other VLANs
Links Not Belonging to the Management VLAN
Switch A
Port A1
Port A3
Port A6
Port A7
Switch B
Port B2
Port B4
Port B5
Port B9
Switch C
Port C2
Port C3
Port C6
Port C8
3
4
1
System Management Workstation
Shipping
2
Server
Server
System Server (on the DEFAULT_VLAN)
Server
Marketing

Figure 2-23. Example of Management VLAN Control in a LAN

Table 2-2. VLAN Membership in Figure 2-23

Switch	A1	A3	A6	A7	B2	B4	B5	B9	C2	C3	C6	C8
Management VLAN (VID = 7)	**Y**	N	N	**Y**	**Y**	**Y**	N	N	**Y**	N	N	N
Marketing VLAN (VID = 12)	N	N	N	N	N	N	N	N	N	**Y**	**Y**	**Y**
Shipping Dept. VLAN (VID = 20)	N	**Y**	**Y**	N	N	N	N	N	N	N	N	N
DEFAULT-VLAN (VID = 1)	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**

Preparation

1. Determine a VID and VLAN name suitable for your Management VLAN.
2. Determine the IP addressing for the Management VLAN (**DHCP/Bootp** or **Manual**.
3. Plan your Management VLAN topology to use ProCurve switches that support this feature. (See the list on page 2-30.) The ports belonging to the Management VLAN should be only the following:
   - Ports to which you will connect authorized management stations (such as Port A7 in figure 2-23.)
   - Ports on one switch that you will use to extend the Management VLAN to ports on other ProCurve switches (such as ports A1 and B2 or B4 and C2 in figure 2-23 on page 2-32.).

Hubs dedicated to connecting management stations to the Management VLAN can also be included in the above topology. Note that any device connected to a hub in the Management VLAN will also have Management VLAN access.

4. Configure the Management VLAN on the selected switch ports.
5. Test the management VLAN from all of the management stations authorized to use the Management VLAN, including any SNMP-based network management stations. Ensure that you include testing any Management VLAN links between switches.

Note

If you configure a Management VLAN on a switch by using a Telnet connection through a port that is not in the Management VLAN, then you will lose management contact with the switch if you log off your Telnet connection or execute **write memory** and **reboot** the switch.

Configuration

Syntax: [no] management-vlan < *vlan-id* | *vlan-name* >

Default: Disabled

To confirm the Management VLAN configuration, use the **show running-config** command.

For example, suppose you have already configured a VLAN named **My_VLAN** with a VID of 100. Now you want to configure the switch to do the following:

- Use **My_VLAN** as a Management VLAN (tagged, in this case) to connect port A1 on switch "A" to a management station. (The management station includes a network interface card with 802.1Q tagged VLAN capability.)
- Use port A2 to extend the Management VLAN to port B1 (which is already configured as a tagged member of **My_VLAN**) on an adjacent switch.

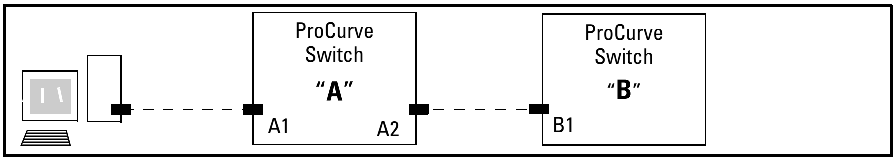


Figure 2-24. Illustration of Configuration Example

```
ProCurve(config)# management-vlan 100
ProCurve(config)# vlan 100 tagged a1
ProCurve(config)# vlan 100 tagged a2
```

Deleting the Management VLAN. You can disable the Secure Management feature without deleting the VLAN itself. For example, either of the following commands disables the Secure Management feature in the above example:

```
ProCurve(config)# no management-vlan 100
ProCurve(config)# no management-vlan my_vlan
```

Operating Notes for Management VLANs

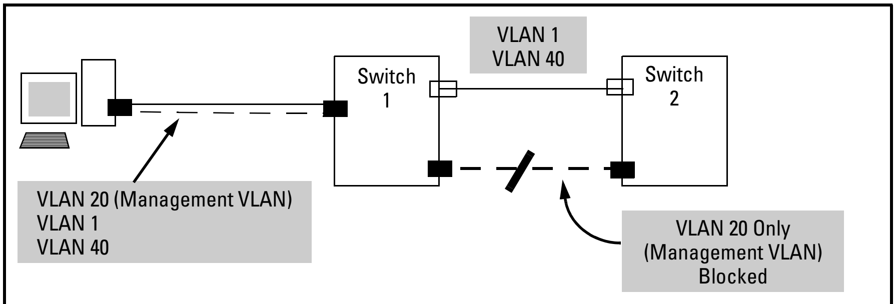
- Only one Management-VLAN can be active in the switch. If one Management-VLAN VID is saved in the startup-config file and you configure a different VID in the running-config file, the switch uses the running-config version until you either use the **write-memory** command or reboot the switch.

- During a Telnet session to the switch, if you configure the Management-VLAN to a VID that excludes the port through which you are connected to the switch, you will continue to have access only until you terminate the session by logging out or rebooting the switch.
- During a web browser session to the switch, if you configure the Management-VLAN to a VID that excludes the port through which you are connected to the switch, you will continue to have access only until you close the browser session or rebooting the switch.

Note

The Management-VLAN feature does not control management access through a direct connection to the switch's serial port.

- Enabling Spanning Tree where there are multiple links using separate VLANs, including the Management VLAN, between a pair of switches, Spanning Tree will force the blocking of one or more links. This may include the link carrying the Management VLAN, which will cause loss of management access to some devices.



Even though the ports on the Management VLAN link between Switch 1 and Switch 2 do not belong to the other VLANs connecting the two switches, enabling Spanning Tree will block one of the two links. This is because Spanning Tree operates per-switch and not per-VLAN.

Figure 2-25. Example of Inadvertently Blocking a Management VLAN Link by Implementing Spanning Tree

Effect of VLANs on Other Switch Features

Spanning Tree Operation with VLANs

Because the switch follows the 802.1Q VLAN recommendation to use single-instance spanning tree, Spanning Tree operates across all ports on the switch (regardless of VLAN assignments) instead of on a per-VLAN basis. This means that if redundant physical links exist between the switch and another 802.1Q device, all but one link will be blocked, regardless of whether the redundant links are in separate VLANs. However, you can use port trunking to prevent Spanning Tree from unnecessarily blocking ports (and to improve overall network performance).

Note that Spanning Tree operates differently in different devices. For example, in the (obsolete, non-802.1Q) ProCurve Switch 2000 and the ProCurve Switch 800T, Spanning Tree operates on a per-VLAN basis, allowing redundant physical links as long as they are in separate VLANs.

IP Interfaces

There is a one-to-one relationship between a VLAN and an IP network interface. Since the VLAN is defined by a group of ports, the state (up/down) of those ports determines the state of the IP network interface associated with that VLAN. When a VLAN comes up because one or more of its ports is up, the IP interface for that VLAN is also activated. Likewise, when a VLAN is deactivated because all of its ports are down, the corresponding IP interface is also deactivated.

VLAN MAC Addresses

Some switch models use the same MAC address for all configured VLANs, while other switch models use a different MAC address for each configured VLAN.

One (Same) MAC Address for all VLANs		Different MAC Address for Each VLAN
2600	3400cl	4100gl
2600-PWR	5300xl	6108
2810	6400cl	
2800		

You can send an 802.2 test packet to the VLAN MAC address to verify connectivity to the switch. Likewise, you can assign an IP address to the VLAN interface, and when you Ping that address, ARP will resolve the IP address to this MAC address.

Port Trunks

When assigning a port trunk to a VLAN, all ports in the trunk are automatically assigned to the same VLAN. You cannot split trunk members across multiple VLANs. Also, a port trunk is tagged, untagged, or excluded from a VLAN in the same way as for individual, untrunked ports.

Port Monitoring

If you designate a port on the switch for network monitoring, this port will appear in the Port VLAN Assignment screen and can be configured as a member of any VLAN. For information on how broadcast, multicast, and unicast packets are tagged inside and outside of the VLAN to which the monitor port is assigned, see the appendix on troubleshooting in the *Management and Configuration Guide*.

VLAN Restrictions

- A port must be a member of at least one VLAN. In the factory default configuration, all ports are assigned to the default VLAN (DEFAULT_VLAN; VID = 1).
- A port can be assigned to several VLANs, but only one of those assignments can be untagged. (The “Untagged” designation enables VLAN operation with non 802.1Q-compliant devices.)
- An external router must be used to communicate between tagged VLANs on the switch.
- Before you can delete a VLAN, you must first re-assign all ports in the VLAN to another VLAN.

Jumbo Packet Support

Jumbo packet support is enabled per-VLAN and applies to all ports belonging to the VLAN. For more information, refer to the chapter titled “Port Status and Basic Configuration” in the *Management and Configuration Guid*e for your switch.

GVRP

Contents

Overview

This chapter describes GVRP and how to configure it with the switch's built-in interfaces, and assumes an understanding of VLANs, which are described in Chapter 2, "Static Virtual LANs (VLANs)".

For general information on how to use the switch's built-in interfaces, refer to these chapters in the *Management and Configuration Guide* for your switch:

- Chapter 3, "Using the Menu Interface"
- Chapter 4, "Using the Command Line Interface (CLI)"
- Chapter 5, "Using the Web Browser Interface"
- Chapter 6, "Switch Memory and Configuration"

Introduction

Feature	Default	Menu	CLI	Web
view GVRP configuration	n/a	page 3-12	page 3-13	page 3-16
list static and dynamic VLANs on a GVRP-enabled switch	n/a	—	page 3-15	page 3-16
enable or disable GVRP	disabled	page 3-12	page 3-14	page 3-16
enable or disable GVRP on individual ports	enabled	page 3-12	page 3-14	—
control how individual ports will handle advertisements for new VLANs	Learn	page 3-12	page 3-14	page 3-16
convert a dynamic VLAN to a static VLAN	n/a	—	page 3-16	—
configure static VLANs	DEFAULT_VLAN (VID = 1)	page 2-14	page 2-19	page 2-25

GVRP—GARP VLAN Registration Protocol—is an application of the Generic Attribute Registration Protocol—GARP. GVRP is defined in the IEEE 802.1Q standard, and GARP is defined in the IEEE 802.1D-1998 standard.

Note

To understand and use GVRP you must have a working knowledge of 802.1Q VLAN tagging. (See “Port-Based Virtual LANs (Static VLANs)” on page 2-3.)

GVRP uses “GVRP Bridge Protocol Data Units” (“GVRP BPDUs”) to “advertise” static VLANs. In this manual, a GVRP BPDU is termed an *advertisement*. Advertisements are sent outbound from ports on a switch to the devices directly connected to those ports.

GVRP enables the switch to dynamically create 802.1Q-compliant VLANs on links with other devices running GVRP. This enables the switch to automatically create VLAN links between GVRP-aware devices. (A GVRP link can include intermediate devices that are not GVRP-aware.) This operation reduces the chances for errors in VLAN configuration by automatically providing VLAN ID (VID) consistency across the network. That is, you can use GVRP to propagate VLANs to other GVRP-aware devices instead of manually having to set up VLANs across your network. After the switch creates a dynamic VLAN, you can optionally use the CLI **static <*vlan-id*>** command to convert it to a static VLAN or allow it to continue as a dynamic VLAN for as long as needed. You can also use GVRP to dynamically enable port membership in static VLANs configured on a switch.

General Operation

When GVRP is enabled on a switch, the VID for any static VLANs configured on the switch is *advertised* (using BPDUs—Bridge Protocol Data Units) out all ports, regardless of whether a port is up or assigned to any particular VLAN. A GVRP-aware port on another device that receives the advertisements over a link can dynamically join the advertised VLAN.

A dynamic VLAN (that is, a VLAN learned through GVRP) is tagged on the port on which it was learned. Also, a GVRP-enabled port can forward an advertisement for a VLAN it learned about from other ports on the same switch (internal source), but the forwarding port will not itself join that VLAN until an advertisement for that VLAN is received through a link from another device (external source) on that specific port

Operating Note: When a GVRP-aware port on a switch learns a VID through GVRP from another device, the switch begins advertising that VID out all of its ports except the port on which the VID was learned.

Core switch with static VLANs (VID= 1, 2, & 3). Port 2 is a member of VIDs 1, 2, & 3.

1. Port 2 advertises VIDs 1, 2, & 3.

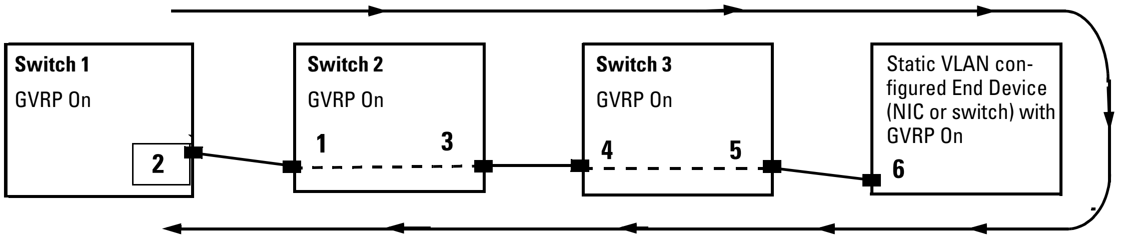
2. Port 1 receives advertisement of VIDs 1, 2, & 3 AND becomes a member of VIDs 1, 2, & 3.

3. Port 3 advertises VIDs 1, 2, & 3, but port 3 is NOT a member of VIDs 1, 2, & 3 at this point.

4. Port 4 receives advertisement of VIDs 1, 2, & 3 AND becomes a member of VIDs 1, 2, & 3.

5. Port 5 advertises VIDs 1, 2, & 3, but port 5 is NOT a member of VIDs 1, 2, & 3 at this point.

Port 6 is statically configured to be a member of VID 3.


Switch 1
GVRP On
2
Switch 2
GVRP On
1
3
Switch 3
GVRP On
4
5
Static VLAN configured End Device (NIC or switch) with GVRP On
6

11. Port 2 receives advertisement of VID 3. (Port 2 is already statically configured for VID 3.)

9. Port 3 receives advertisement of VID 3 AND becomes a member of VID 3. (Still not a member of VIDs 1 & 2.)

10. Port 1 advertises VID 3.

7. Port 5 receives advertisement of VID 3 AND becomes a member of VID 3. (Still not a member of VIDs 1 & 2.)

8. Port 4 advertises VID 3.

6. Port 6 advertises VID 3.

Figure 3-1. Example of Forwarding Advertisements and Dynamic Joining

Note that if a static VLAN is configured on at least one port of a switch, and that port has established a link with another device, then all other ports of that switch will send advertisements for that VLAN.

For example, in the following figure, Tagged VLAN ports on switch "A" and switch "C" advertise VLANs 22 and 33 to ports on other GVRP-enabled switches that can dynamically join the VLANs.

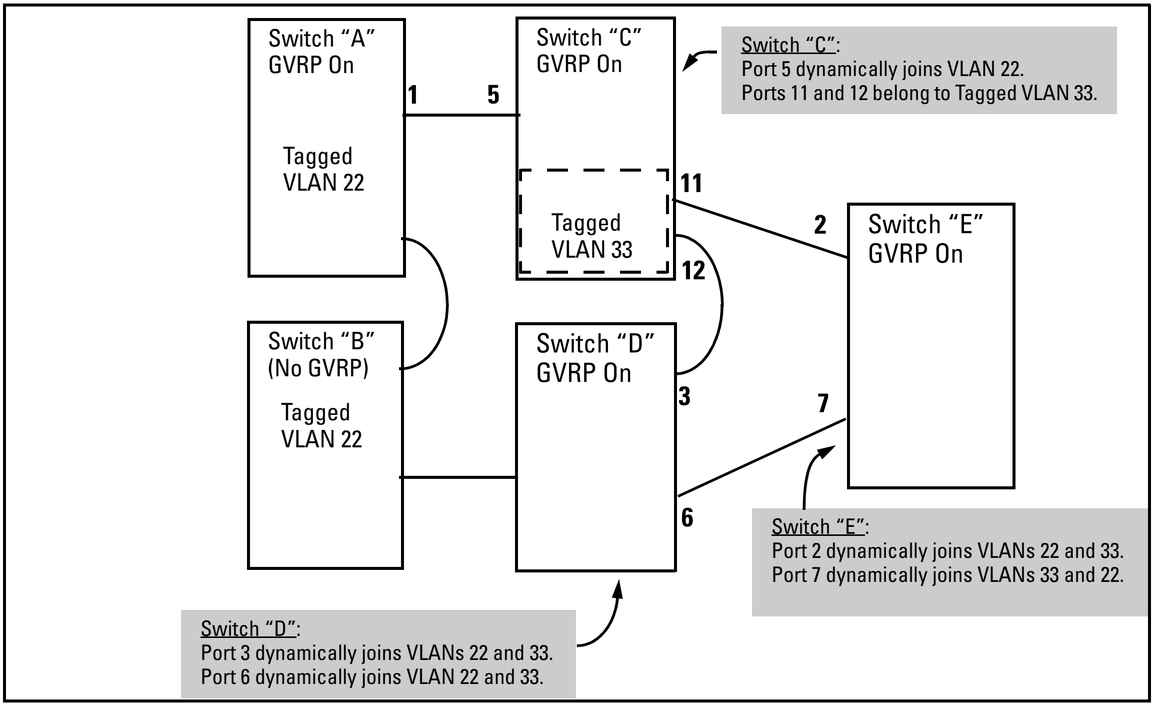

Switch "A"
GVRP On
Tagged
VLAN 22
1
5
Switch "C"
GVRP On
Tagged
VLAN 33
11
12
Switch "C":
Port 5 dynamically joins VLAN 22.
Ports 11 and 12 belong to Tagged VLAN 33.
2
Switch "E"
GVRP On
Switch "B"
(No GVRP)
Tagged
VLAN 22
Switch "D"
GVRP On
3
7
6
Switch "E":
Port 2 dynamically joins VLANs 22 and 33.
Port 7 dynamically joins VLANs 33 and 22.
Switch "D":
Port 3 dynamically joins VLANs 22 and 33.
Port 6 dynamically joins VLAN 22 and 33.

Figure 3-2. Example of GVRP Operation

Note

A port can learn of a dynamic VLAN through devices that are not aware of GVRP (Switch "B", above). VLANs must be disabled in GVRP-unaware devices to allow tagged packets to pass through.

A GVRP-aware port receiving advertisements has these options:

- If there is not already a static VLAN with the advertised VID on the receiving port, then dynamically create the VLAN and become a member.

- If the switch already has a static VLAN assignment with the same VID as in the advertisement, and the port is configured to **Auto** for that VLAN, then the port will dynamically join the VLAN and begin moving that VLAN's traffic. (For more detail on **Auto**, see "Per-Port Options for Dynamic VLAN Advertising and Joining" on page 3-8.)
- Ignore the advertisement for that VID.
- Don't participate in that VLAN.

Note also that a port belonging to a Tagged or Untagged static VLAN has these configurable options:

- Send VLAN advertisements, and also receive advertisements for VLANs on other ports and dynamically join those VLANs.
- Send VLAN advertisements, but ignore advertisements received from other ports.
- Avoid GVRP participation by not sending advertisements and dropping any advertisements received from other devices.

IP Addressing. A dynamic VLAN does not have an IP address, and moves traffic on the basis of port membership in VLANs. However, after GVRP creates a dynamic VLAN, you can convert it to a static VLAN. Note that it is then necessary to assign ports to the VLAN in the same way that you would for a static VLAN that you created manually. In the static state you can configure IP addressing on the VLAN and access it in the same way that you would any other static (manually created) VLAN.

Per-Port Options for Handling GVRP "Unknown VLANs"

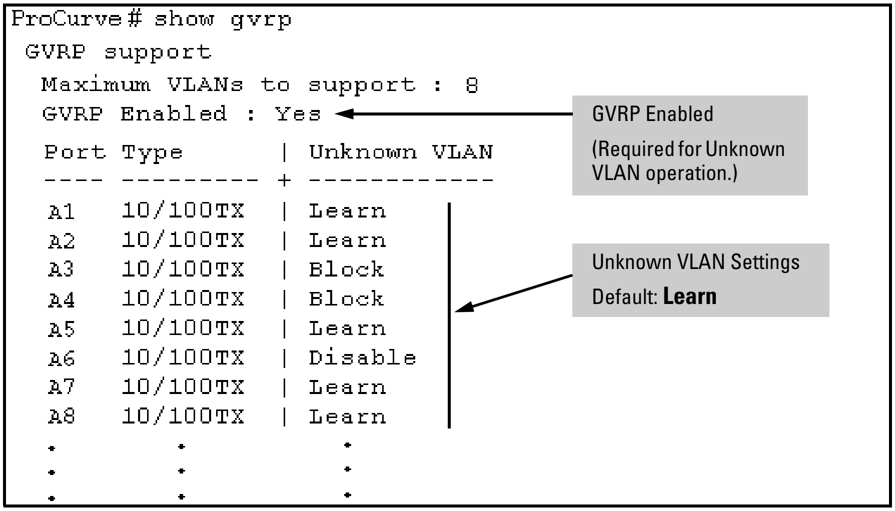
An "unknown VLAN" is a VLAN that the switch learns of by receiving an advertisement for that VLAN on a port that is not already a member of that VLAN. If the port is configured to learn unknown VLANs, then the VLAN is dynamically created and the port becomes a tagged member of the VLAN. For example, suppose that in figure 3-2 (page 3-5), port 1 on switch "A" is connected to port 5 on switch "C". Because switch "A" has VLAN 22 statically configured, while switch "C" does not have this VLAN statically configured (and does not "Forbid" VLAN 22 on port 5), VLAN 22 is handled as an "Unknown VLAN" on port 5 in switch "C". Conversely, if VLAN 22 was statically configured on switch C, but port 5 was not a member, port 5 would become a member when advertisements for VLAN 22 were received from switch "A".

When you enable GVRP on a switch, you have the per-port join-request options listed in Table 3-1.

Table 3-1. Options for Handling “Unknown VLAN” Advertisements:

Unknown VLAN Mode	Operation
Learn (the Default)	Enables the port to become a member of any unknown VLAN for which it receives an advertisement. Allows the port to advertise other VLANs that have at least one other port on the same switch as a member.
Block	Prevents the port from joining any new dynamic VLANs for which it receives an advertisement. Allows the port to advertise other VLANs that have at least one other port as a member.
Disable	Causes the port to ignore and drop all GVRP advertisements it receives and also prevents the port from sending any GVRP advertisements.

The CLI **show gvrp** command and the menu interface VLAN Support screen show a switch’s current GVRP configuration, including the Unknown VLAN settings.



```
ProCurve# show gvrp
 GVRP support

  Maximum VLANs to support : 8
  GVRP Enabled : Yes

  Port Type      | Unknown VLAN
  ---- --------- + ------------
  A1   10/100TX  | Learn
  A2   10/100TX  | Learn
  A3   10/100TX  | Block
  A4   10/100TX  | Block
  A5   10/100TX  | Learn
  A6   10/100TX  | Disable
  A7   10/100TX  | Learn
  A8   10/100TX  | Learn
  .        .          .
  .        .          .
  .        .          .
```

GVRP Enabled

(Required for Unknown VLAN operation.)

Unknown VLAN Settings

Default: **Learn**

Figure 3-3. Example of GVRP Unknown VLAN Settings

Per-Port Options for Dynamic VLAN Advertising and Joining

Initiating Advertisements. As described in the preceding section, to enable dynamic joins, GVRP must be enabled and a port must be configured to Learn (the default). However, to send advertisements in your network, one or more static (**Tagged**, **Untagged**, or **Auto**) VLANs must be configured on one or more switches (with GVRP enabled), depending on your topology.

Enabling a Port for Dynamic Joins. You can configure a port to dynamically join a static VLAN. The join will then occur if that port subsequently receives an advertisement for the static VLAN. (This is done by using the **Auto** and **Learn** options described in table 3-2, below.

Parameters for Controlling VLAN Propagation Behavior. You can configure an individual port to actively or passively participate in dynamic VLAN propagation or to ignore dynamic VLAN (GVRP) operation. These options are controlled by the GVRP "Unknown VLAN" and the static VLAN configuration parameters, as described in the following table:

Table 3-2. Controlling VLAN Behavior on Ports with Static VLANs

Per-Port "Unknown VLAN" (GVRP) Configuration	Static VLAN Options—Per VLAN Specified on Each Port [1]		
	Port Activity: Tagged or Untagged (Per VLAN)[2]	**Port Activity: Auto[2] (Per VLAN)**	**Port Activity: Forbid (Per VLAN)[2]**
Learn (the Default)	The port: • Belongs to specified VLAN. • Advertises specified VLAN. • Can become a member of dynamic VLANs for which it receives advertisements. • Advertises dynamic VLANs that have at least one other port (on the same switch) as a member.	The port: • Will become a member of specified VLAN if it receives advertisements for specified VLAN from another device. • Will advertise specified VLAN. • Can become a member of other, dynamic VLANs for which it receives advertisements. • Will advertise a dynamic VLAN that has at least one other port (on the same switch) as a member.	The port: 1. Will not become a member of the specified VLAN. 1. Will not advertise specified VLAN. 1. Can become a member of other dynamic VLANs for which it receives advertisements. 1. Will advertise a dynamic VLAN that has at least one other port on the same switch as a member.

Per-Port "Unknown VLAN" (GVRP) Configuration	**Static VLAN Options—Per VLAN Specified on Each Port** [1]		
	Port Activity: Tagged or Untagged (Per VLAN)[2]	**Port Activity: Auto**[2] **(Per VLAN)**	**Port Activity: Forbid (Per VLAN)**[2]
Block	The port: • Belongs to the specified VLAN. • Advertises this VLAN. • Will not become a member of new dynamic VLANs for which it receives advertisements. • Will advertise dynamic VLANs that have at least one other port as a member.	The port: • Will become a member of specified VLAN if it receives advertisements for this VLAN. • Will advertise this VLAN. • Will not become a member of new dynamic VLANs for which it receives advertisements. • Will advertise dynamic VLANs that have at least one other port (on the same switch) as a member.	The port: • Will not become a member of the specified VLAN. • Will not advertise this VLAN. • Will not become a member of dynamic VLANs for which it receives advertisements. • Will advertise dynamic VLANs that have at least one other port (on the same switch) as a member.
Disable	The port: • Is a member of the specified VLAN. • Will ignore GVRP PDUs. • Will not join any advertised VLANs. • Will not advertise VLANs.	The port: • Will not become a member of the specified VLAN. • Will ignore GVRP PDUs. • Will not join any dynamic VLANs. • Will not advertise VLANs.	The port: • Will not become a member of this VLAN. • Will ignore GVRP PDUs. • Will not join any dynamic VLANs. • Will not advertise VLANs.

[1] Each port on the switch must be a Tagged or Untagged member of at least one VLAN. Thus, any port configured for GVRP to Learn or Block will generate and forward advertisements for static VLAN(s) configured on the switch and also for dynamic VLANs the switch learns on other ports.

[2] To configure tagging, **Auto**, or **Forbid**, see "Configuring Static VLAN Name and Per-Port Settings" on page 2-24 (for the CLI) or "Adding or Changing a VLAN Port Assignment" on page 2-17 (for the menu).

As the preceding table indicates, when you enable GVRP, a port that has a Tagged or Untagged static VLAN has the option for both generating advertisements and dynamically joining other VLANs.

Note

In table 3-2, above, the Unknown VLAN parameters are configured on a per-port basis using the CLI. The Tagged, Untagged, Auto, and Forbid options are configured per static VLAN on every port, using either the menu interface or the CLI.

Because dynamic VLANs operate as Tagged VLANs, and because a tagged port on one device cannot communicate with an untagged port on another device, ProCurve recommends that you use Tagged VLANs for the static VLANs you will use to generate advertisements.

GVRP and VLAN Access Control

When you enable GVRP on a switch, the default GVRP parameter settings allow all of the switch's ports to transmit and receive dynamic VLAN advertisements (GVRP advertisements) and to dynamically join VLANs. The two preceding sections describe the per-port features you can use to control and limit VLAN propagation. To summarize, you can:

- Allow a port to advertise and/or join dynamic VLANs (Learn mode—the default).
- Allow a port to send VLAN advertisements, but not receive them from other devices; that is, the port cannot dynamically join a VLAN but other devices can dynamically join the VLANs it advertises (Block mode).
- Prevent a port from participating in GVRP operation (Disable mode).

Port-Leave From a Dynamic VLAN

A dynamic VLAN continues to exist on a port for as long as the port continues to receive advertisements of that VLAN from another device connected to that port or until you:

- Convert the VLAN to a static VLAN (See "Converting a Dynamic VLAN to a Static VLAN" on page 2-23.)
- Reconfigure the port to **Block** or **Disable**
- Disable GVRP
- Reboot the switch

The time-to-live for dynamic VLANs is 10 seconds. That is, if a port has not received an advertisement for an existing dynamic VLAN during the last 10 seconds, the port removes itself from that dynamic VLAN.

Planning for GVRP Operation

These steps outline the procedure for setting up dynamic VLANs for a segment.

1. Determine the VLAN topology you want for each segment (broadcast domain) on your network.
2. Determine the VLANs that must be static and the VLANs that can be dynamically propagated.
3. Determine the device or devices on which you must manually create static VLANs in order to propagate VLANs throughout the segment.

4. Determine security boundaries and how the individual ports in the segment will handle dynamic VLAN advertisements. (See table 3-1 on page 3-7 and table 3-2 on page 3-8.)
5. Enable GVRP on all devices you want to use with dynamic VLANs and configure the appropriate “Unknown VLAN” parameter (**Learn**, **Block**, or **Disable**) for each port.
6. Configure the static VLANs on the switch(es) where they are needed, along with the per-VLAN parameters (**Tagged**, **Untagged**, **Auto**, and **Forbid**—see table 3-2 on page 3-8) on each port.
7. Dynamic VLANs will then appear automatically, according to the configuration options you have chosen.
8. Convert dynamic VLANs to static VLANs where you want dynamic VLANs to become permanent.

Configuring GVRP On a Switch

The procedures in this section describe how to:

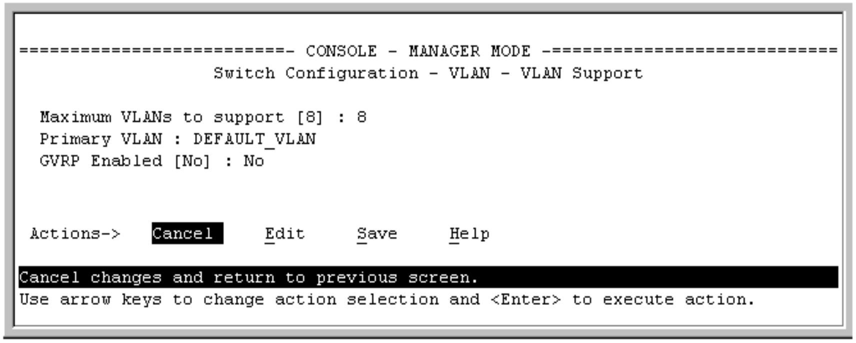
- View the GVRP configuration on a switch
- Enable and disable GVRP on a switch
- Specify how individual ports will handle advertisements

To view or configure static VLANs for GVRP operation, refer to “Per-Port Static VLAN Configuration Options” on page 2-8.

Menu: Viewing and Configuring GVRP

1. From the Main Menu, select:

   2. Switch Configuration . . .
       8. VLAN Menu . . .
           1. VLAN Support

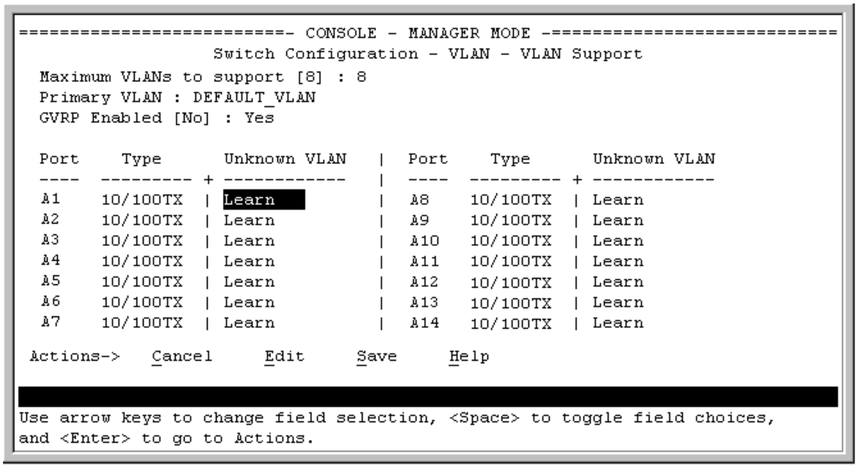


```
=========================- CONSOLE - MANAGER MODE -===========================
                   Switch Configuration - VLAN - VLAN Support

  Maximum VLANs to support [8] : 8
  Primary VLAN : DEFAULT_VLAN
  GVRP Enabled [No] : No


 Actions->   Cancel     Edit     Save     Help

Cancel changes and return to previous screen.
Use arrow keys to change action selection and <Enter> to execute action.
```

Figure 3-4. The VLAN Support Screen (Default Configuration)

2. Do the following to enable GVRP and display the Unknown VLAN fields:
   a. Press **[E]** (for **Edit**).
   b. Use [↓] to move the cursor to the **GVRP Enabled** field.
   c. Press the Space bar to select **Yes**.
   d. Press [↓] again to display the **Unknown VLAN** fields.

The Unknown VLAN fields enable you to configure each port to:
- Learn - Dynamically join any advertised VLAN and advertise all VLANs learned through other ports.
- Block - Do not dynamically join any VLAN, but still advertise all VLANs learned through other ports.
- Disable - Ignore and drop all incoming advertisements and do not transmit any advertisements.



```
=========================- CONSOLE - MANAGER MODE -===========================
                   Switch Configuration - VLAN - VLAN Support
  Maximum VLANs to support [8] : 8
  Primary VLAN : DEFAULT_VLAN
  GVRP Enabled [No] : Yes

  Port   Type      Unknown VLAN   |  Port   Type      Unknown VLAN
  ----  --------- + ------------  |  ----  --------- + ------------
  A1    10/100TX  | Learn         |  A8    10/100TX  | Learn
  A2    10/100TX  | Learn         |  A9    10/100TX  | Learn
  A3    10/100TX  | Learn         |  A10   10/100TX  | Learn
  A4    10/100TX  | Learn         |  A11   10/100TX  | Learn
  A5    10/100TX  | Learn         |  A12   10/100TX  | Learn
  A6    10/100TX  | Learn         |  A13   10/100TX  | Learn
  A7    10/100TX  | Learn         |  A14   10/100TX  | Learn

 Actions->   Cancel     Edit     Save     Help


Use arrow keys to change field selection, <Space> to toggle field choices,
and <Enter> to go to Actions.
```

Figure 3-5. Example Showing Default Settings for Handling Advertisements

3. Use the arrow keys to select the port you want, and the Space bar to select Unknown VLAN option for any ports you want to change.
4. When you finish making configuration changes, press **[Enter]**, then **[S]** (for **Save**) to save your changes to the Startup-Config file.

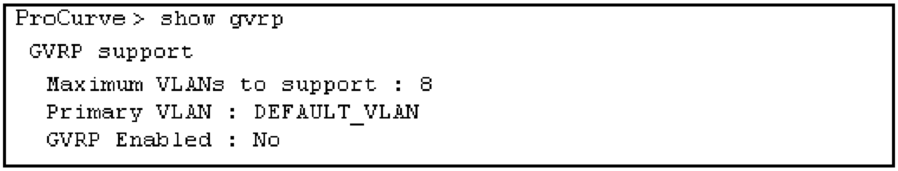
CLI: Viewing and Configuring GVRP

GVRP Commands Used in This Section

show gvrp	below
gvrp	page 3-14
unknown-vlans	page 3-14

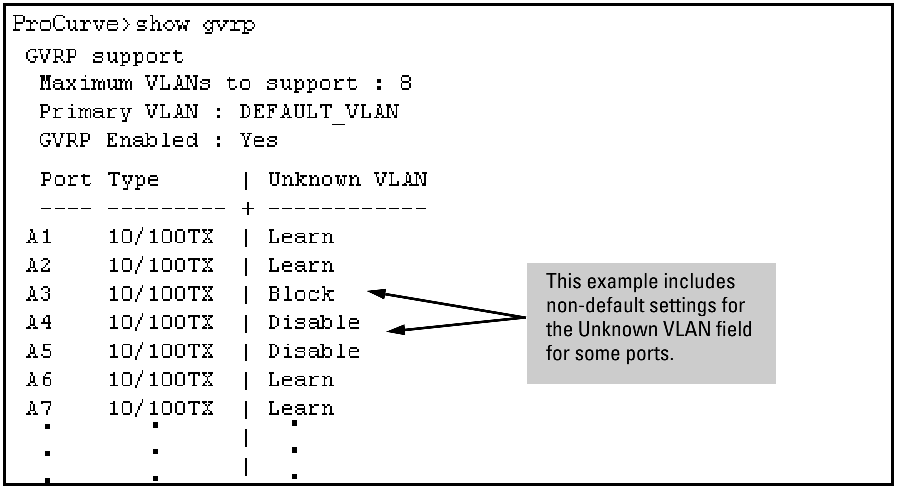
Displaying the Switch's Current GVRP Configuration. This command shows whether GVRP is disabled, along with the current settings for the maximum number of VLANs and the current Primary VLAN. (For more on the last two parameters, see "Port-Based Virtual LANs (Static VLANs)" on page 2-3.)

Syntax: show gvrp *Shows the current settings.*



```
ProCurve > show gvrp
 GVRP support
  Maximum VLANs to support : 8
  Primary VLAN : DEFAULT_VLAN
  GVRP Enabled : No
```

Figure 3-6. Example of "Show GVRP" Listing with GVRP Disabled



```
ProCurve>show gvrp
 GVRP support
  Maximum VLANs to support : 8
  Primary VLAN : DEFAULT_VLAN
  GVRP Enabled : Yes

  Port Type      | Unknown VLAN
  ---- --------- + ------------
 A1    10/100TX  | Learn
 A2    10/100TX  | Learn
 A3    10/100TX  | Block
 A4    10/100TX  | Disable
 A5    10/100TX  | Disable
 A6    10/100TX  | Learn
 A7    10/100TX  | Learn
  .       .      |   .
  .       .      |   .
  .       .      |   .
```

This example includes non-default settings for the Unknown VLAN field for some ports.

Figure 3-7. Example of Show GVRP Listing with GVRP Enabled

Enabling and Disabling GVRP on the Switch. This command enables GVRP on the switch.

Syntax: gvrp

This example enables GVRP:

```
ProCurve(config)# gvrp
```

This example disables GVRP operation on the switch:

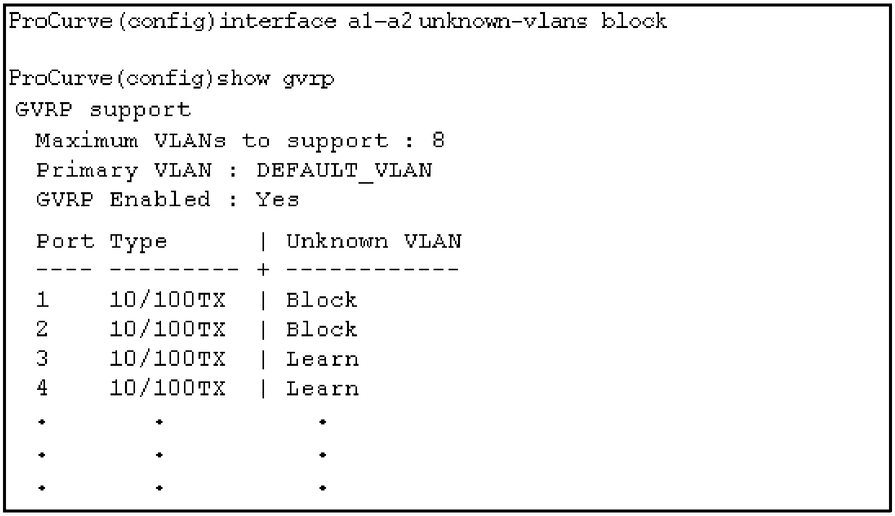
```
ProCurve(config)# no gvrp
```

Enabling and Disabling GVRP On Individual Ports. When GVRP is enabled on the switch, use the **unknown-vlans** command to change the Unknown VLAN field for one or more ports. You can use this command at either the Manager level or the interface context level for the desired port(s).

Syntax: interface <*port-list*> unknown-vlans < learn | block | disable >

Changes the Unknown VLAN field setting for the specified port(s).

For example, to change and view the configuration for ports A1-A2 to **Block**:



```
ProCurve(config)interface a1-a2 unknown-vlans block

ProCurve(config)show gvrp
 GVRP support
   Maximum VLANs to support : 8
   Primary VLAN : DEFAULT_VLAN
   GVRP Enabled : Yes

   Port Type      | Unknown VLAN
   ---- --------- + ------------
   1    10/100TX  | Block
   2    10/100TX  | Block
   3    10/100TX  | Learn
   4    10/100TX  | Learn
   .       .           .
   .       .           .
   .       .           .
```

Figure 3-8. Example of Preventing Specific Ports from Joining Dynamic VLANs

Displaying the Static and Dynamic VLANs Active on the Switch. The **show vlans** command lists all VLANs present in the switch.

Syntax: show vlans

For example, in the following illustration, switch "B" has one static VLAN (the default VLAN), with GVRP enabled and port 1 configured to **Learn** for Unknown VLANs. Switch "A" has GVRP enabled and has three static VLANs: the default VLAN, VLAN-222, and VLAN-333. In this scenario, switch B will dynamically join VLAN-222 and VLAN-333:

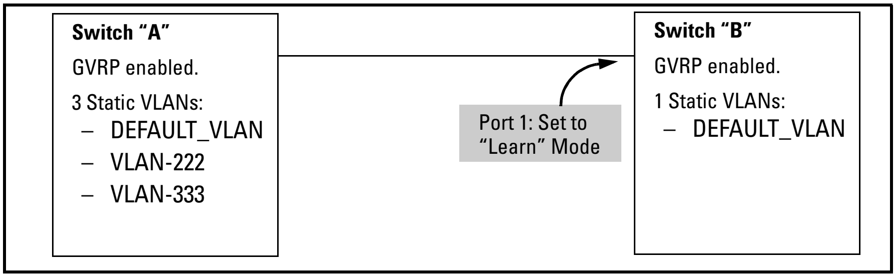
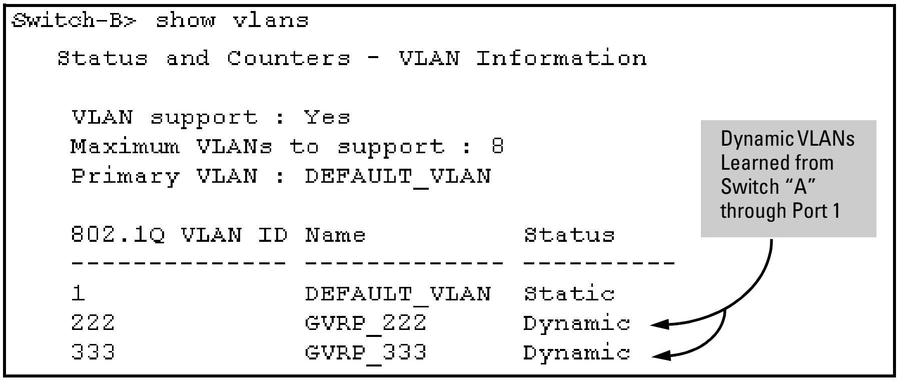


Figure 3-9. Example of Switches Operating with GVRP Enabled

The **show vlans** command lists the dynamic (and static) VLANs in switch "B" after it has learned and joined VLAN-222 and VLAN-333.



```
Switch-B> show vlans
   Status and Counters - VLAN Information

    VLAN support : Yes
    Maximum VLANs to support : 8
    Primary VLAN : DEFAULT_VLAN

    802.1Q VLAN ID Name          Status
    -------------- ------------- ----------
    1              DEFAULT_VLAN  Static
    222            GVRP_222      Dynamic
    333            GVRP_333      Dynamic
```

Dynamic VLANs Learned from Switch "A" through Port 1

Figure 3-10. Example of Listing Showing Dynamic VLANs

Converting a Dynamic VLAN to a Static VLAN. If a port on the switch has joined a dynamic VLAN, you can use the following command to convert that dynamic VLAN to a static VLAN:

Syntax: static <*dynamic-vlan-id*>

For example, to convert dynamic VLAN 333 (from the previous example) to a static VLAN:

```
ProCurve(config)# static 333
```

When you convert a dynamic VLAN to a static VLAN, all ports on the switch are assigned to the VLAN in Auto mode.

Web: Viewing and Configuring GVRP

To view, enable, disable, or reconfigure GVRP:

1. Click on the **Configuration** tab.
2. Click on **VLAN Configuration** and do the following:
   a. To enable or disable GVRP, click on **GVRP Enabled**.
   b. To change the Unknown VLAN field for any port:
      i. Click on **GVRP Security** and make the desired changes.
      ii. Click on **Apply** to save and implement your changes to the Unknown VLAN fields.

For web-based Help on how to use the web browser interface screen, click on the **[?]** button provided on the web browser screen.

GVRP Operating Notes

- A dynamic VLAN must be converted to a static VLAN before it can have an IP address.
- The total number of VLANs on the switch (static and dynamic combined) cannot exceed the current Maximum VLANs setting. For example, in the factory default state, the switch supports eight VLANs. Thus, in a case where four static VLANs are configured on the switch, the switch can accept up to four additional VLANs in any combination of static and dynamic. Any additional VLANs advertised to the switch will not be added unless you first increase the Maximum VLANs setting. In the Menu interface, click on **2. Switch Configuration ... | 8. VLAN Menu | 1. VLAN Support.** In the global config level of the CLI, use **max-vlans.**
- Converting a dynamic VLAN to a static VLAN and then executing the **write memory** command saves the VLAN in the startup-config file and makes it a permanent part of the switch's VLAN configuration.
- Within the same broadcast domain, a dynamic VLAN can pass through a device that is not GVRP-aware. This is because a hub or a switch that is not GVRP-ware will flood the GVRP (multicast) advertisement packets out all ports.
- GVRP assigns dynamic VLANs as Tagged VLANs. To configure the VLAN as Untagged, you must first convert it to a static VLAN.
- Rebooting a switch on which a dynamic VLAN exists deletes that VLAN. However, the dynamic VLAN re-appears after the reboot if GVRP is enabled and the switch again receives advertisements for that VLAN through a port configured to add dynamic VLANs.
- By receiving advertisements from other devices running GVRP, the switch learns of static VLANs on those other devices and dynamically (automatically) creates tagged VLANs on the links to the advertising devices. Similarly, the switch advertises its static VLANs to other GVRP-aware devices, as well as the dynamic VLANs the switch has learned.
- A GVRP-enabled switch does not advertise any GVRP-learned VLANs out of the port(s) on which it originally learned of those VLANs.

4

Multimedia Traffic Control with IP Multicast (IGMP)

Contents

Overview

This chapter describes Multimedia Traffic Control with IP Multicast (IGMP), and explains how to configure IGMP controls to reduce unnecessary bandwidth usage on a per-port basis.

For the latest information on IGMP, see the software release notes posted on the ProCurve Networking support web site at http://www.procurve.com.

For general information on how to use the switch's built-in interfaces, refer to these chapters in the *Management and Configuration Guide* for your switch:

- Chapter 3, "Using the Menu Interface"
- Chapter 4, "Using the Command Line Interface (CLI)"
- Chapter 5, "Using the Web Browser Interface"
- Appendix C, "Switch Memory and Configuration"

General Operation and Features

IGMP Features

Feature	Default	Menu	CLI	Web
view igmp configuration	n/a	—	page 4-6	—
show igmp status for multicast groups used by the selected VLAN	n/a	—	Yes	—
enabling or disabling IGMP (Requires VLAN ID Context)	disabled	—	page 4-8	page 4-11
per-port packet control	auto	—	page 4-9	—
IGMP traffic priority	normal	—	page 4-10	—
querier	enabled	—	page 4-10	—
fast-leave	disabled	—	page 4-14	—

In a network where IP multicast traffic is transmitted for various multimedia applications, you can use the switch to reduce unnecessary bandwidth usage on a per-port basis by configuring IGMP (Internet Group Management Protocol controls). In the factory default state (IGMP disabled), the switch simply floods all IP multicast traffic it receives on a given VLAN through all ports on that VLAN (except the port on which it received the traffic). This can result in significant and unnecessary bandwidth usage in networks where IP multicast traffic is a factor. Enabling IGMP allows the ports to detect IGMP queries and report packets and manage IP multicast traffic through the switch.

IGMP is useful in multimedia applications such as LAN TV, desktop conferencing, and collaborative computing, where there is multipoint communication; that is, communication from one to many hosts, or communication originating from many hosts and destined for many other hosts. In such multipoint applications, IGMP will be configured on the hosts, and multicast traffic will be generated by one or more servers (inside or outside of the local network). Switches in the network (that support IGMP) can then be configured to direct the multicast traffic to only the ports where needed. If multiple VLANs are configured, you can configure IGMP on a per-VLAN basis.

Enabling IGMP allows detection of IGMP queries and report packets in order to manage IP multicast traffic through the switch. If no other querier is detected, the switch will then also function as the querier. (If you need to disable the querier feature, you can do so through the IGMP configuration MIB. Refer to “Changing the Querier Configuration Setting” on page 4-10.)

Note

IGMP configuration on the switch operates at the VLAN context level. If you are not using VLANs, then configure IGMP in VLAN 1 (the default VLAN) context.

IGMP Terms

- **IGMP Device:** A switch or router running IGMP traffic control features.
- **IGMP Host:** An end-node device running an IGMP (multipoint, or multicast communication) application.
- **Querier:** A required IGMP device that facilitates the IGMP protocol and traffic flow on a given LAN. This device tracks which ports are connected to devices (IGMP clients) that belong to specific multicast groups, and triggers updates of this information. A querier uses data received from the queries to determine whether to forward or block multicast traffic on specific ports. When the switch has an IP address on a given VLAN, it automatically operates as a Querier for that VLAN if it does not detect a multicast router or another switch functioning as a Querier. When enabled (the default state), the switch’s querier function eliminates the need for a multicast router. In most cases, ProCurve recommends that you leave this parameter in the default “enabled” state even if you have a multicast router performing the querier function in your multicast group. For more information, see “How IGMP Operates” on page 4-11.

IGMP Operating Features

Basic Operation

In the factory default configuration, IGMP is disabled. If multiple VLANs are not configured, you must configure IGMP on the default VLAN (DEFAULT_VLAN; VID = 1). If multiple VLANs are configured, you must configure IGMP on a per-VLAN basis for every VLAN where this feature is desired.

Enhancements

With the CLI, you can configure these additional options:

- **Forward with High Priority.** Disabling this parameter (the default) causes the switch or VLAN to process IP multicast traffic, along with other traffic, in the order received (usually, normal priority). Enabling this parameter causes the switch or VLAN to give a higher priority to IP multicast traffic than to other traffic.
- **Auto/Blocked/Forward:** You can use the console to configure individual ports to any of the following states:
  - **Auto** (the default): Causes the switch to interpret IGMP packets and to filter IP multicast traffic based on the IGMP packet information for ports belonging to a multicast group. This means that IGMP traffic will be forwarded on a specific port only if an IGMP host or multicast router is connected to the port.
  - **Blocked:** Causes the switch to drop all IGMP transmissions received from a specific port and to block all outgoing IP Multicast packets for that port. This has the effect of preventing IGMP traffic from moving through specific ports.
  - **Forward:** Causes the switch to forward all IGMP and IP multicast transmissions through the port.
- **Operation With or Without IP Addressing:** This feature helps to conserve IP addresses by enabling IGMP to run on VLANs that do not have an IP address. See “Operation With or Without IP Addressing” on page 4-13.
- **Querier Capability:** The switch performs this function for IGMP on VLANs having an IP address when there is no other device in the VLAN acting as querier. See “Querier Operation” on page 4-21.

Notes

Whenever IGMP is enabled, the switch generates an Event Log message indicating whether querier functionality is enabled.

IP multicast traffic groups are identified by IP addresses in the range of 224.0.0.0 to 239.255.255.255. Also, incoming IGMP packets intended for reserved, or "well-known" multicast addresses automatically flood through all ports (except the port on which the packets entered the switch). For more on this topic, see "Excluding Multicast Addresses from IP Multicast Filtering" on page 4-22.

For more information, refer to "How IGMP Operates" on page 4-11.

CLI: Configuring and Displaying IGMP

IGMP Commands Used in This Section

show ip igmp configuration config vid [config] group <ip address>	page 4-7
ip igmp	page 4-8
high-priority-forward	page 4-10
auto <[ethernet] <port-list>	page 4-9
blocked <[ethernet] <port-list>	page 4-9
forward <[ethernet] <port-list>	page 4-9
querier	page 4-10
show ip igmp	See the appendix on monitoring and analyzing switch operation in the *Management and Configuration Guide*.

Viewing the Current IGMP Configuration. This command lists the IGMP configuration for all VLANs configured on the switch or for a specific VLAN.

Syntax: show ip igmp config

IGMP configuration for all VLANs on the switch.

Syntax: show ip igmp < *vid* > config

IGMP configuration for a specific VLAN on the switch, including per-port data

Syntax: show ip igmp group < *ip-address* >

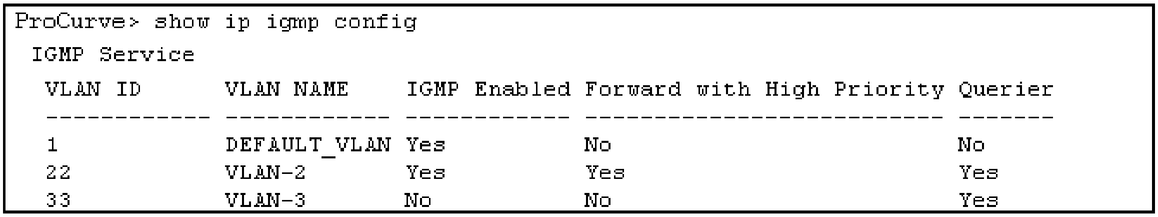
Lists the ports on which the specified multicast group IP address is registered.

(For IGMP operating status, see the appendix on monitoring and analyzing switch operation in the *Management and Configuration Guide*.)

For example, suppose you have the following VLAN and IGMP configurations on the switch:

VLAN ID	VLAN Name	IGMP Enabled	Forward with High Priority	Querier
1	DEFAULT_VLAN	Yes	No	No
22	VLAN-2	Yes	Yes	Yes
33	VLAN-3	No	No	No

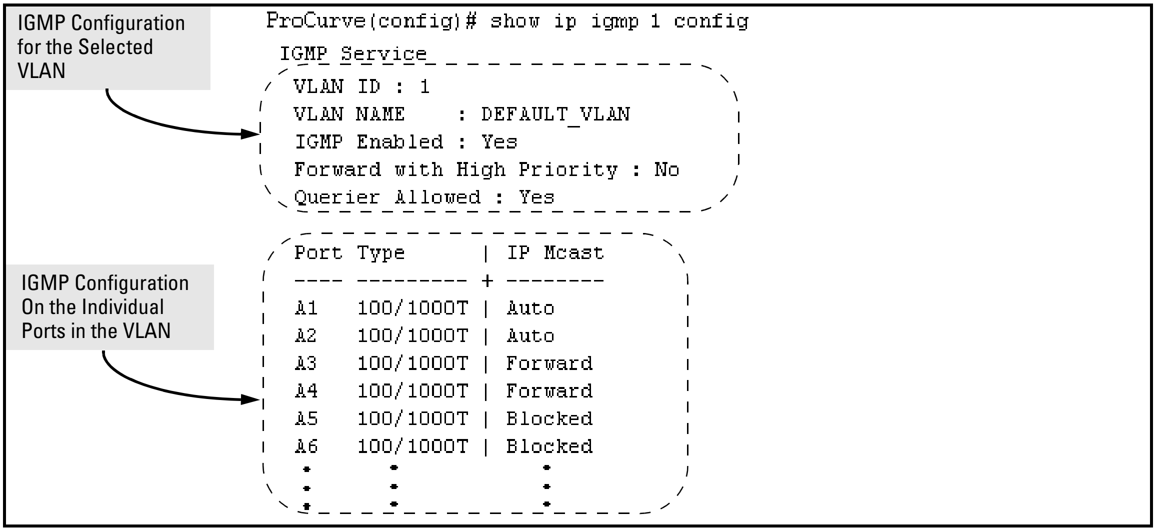
You could use the CLI to display this data as follows:



```
ProCurve> show ip igmp config
 IGMP Service
  VLAN ID      VLAN NAME    IGMP Enabled Forward with High Priority Querier
  ------------ ------------ ------------ -------------------------- -------
  1            DEFAULT_VLAN Yes          No                         No
  22           VLAN-2       Yes          Yes                        Yes
  33           VLAN-3       No           No                         Yes
```

Figure 4-1. Example Listing of IGMP Configuration for All VLANs in the Switch

The following version of the **show ip igmp** command includes the VLAN ID (*vid)* designation, and combines the above data with the IGMP per-port configuration:



IGMP Configuration for the Selected VLAN

IGMP Configuration On the Individual Ports in the VLAN

```
ProCurve(config)# show ip igmp 1 config
 IGMP Service
  VLAN ID : 1
  VLAN NAME    : DEFAULT_VLAN
  IGMP Enabled : Yes
  Forward with High Priority : No
  Querier Allowed : Yes

  Port Type      | IP Mcast
  ---- --------- + --------
  A1   100/1000T | Auto
  A2   100/1000T | Auto
  A3   100/1000T | Forward
  A4   100/1000T | Forward
  A5   100/1000T | Blocked
  A6   100/1000T | Blocked
   .      .          .
   .      .          .
   .      .          .
```

Figure 4-2. Example Listing of IGMP Configuration for A Specific VLAN

Enabling or Disabling IGMP on a VLAN. You can enable IGMP on a VLAN, along with the last-saved or default IGMP configuration (whichever was most recently set), or you can disable IGMP on a selected VLAN. Note that this command must be executed in a VLAN context.

Syntax: [no] ip igmp

For example, here are methods to enable and disable IGMP on the default VLAN (VID = 1).

```
ProCurve(config)# vlan 1 ip igmp
```

Enables IGMP on VLAN 1.

```
ProCurve(vlan-1)# ip igmp
```

Same as above.

```
ProCurve(config)# no vlan 1 ip igmp
```

Disables IGMP on VLAN 1.

Note

If you disable IGMP on a VLAN and then later re-enable IGMP on that VLAN, the switch restores the last-saved IGMP configuration for that VLAN. For more on how switch memory operates, see the chapter on switch memory and configuration in the *Management and Configuration Guide.*

You can also combine the **ip igmp** command with other IGMP-related commands, as described in the following sections.

Configuring Per-Port IGMP Packet Control. Use this command in the VLAN context to specify how each port should handle IGMP traffic.

Syntax: vlan < *vid* > ip igmp
[auto <*port-list*> | blocked <*port-list*> | forward <*port-list*>]

Syntax: vlan < vid > ip igmp

Enables IGMP on the specified VLAN. In a VLAN context, use only **ip igmp** *without the VLAN specifier.*

auto < *port-list* > (Default operation)

Filter multicast traffic on the specified ports. Forward IGMP traffic to hosts on the ports that belong to the multicast group for which the traffic is intended. (Also forward any multicast traffic through any of these ports that is connected to a multicast router.) This is the default IGMP port configuration.

blocked < *port-list* >

Drop all multicast traffic received from devices on the specified ports, and prevent any outgoing multicast traffic from moving through these ports.

forward < *port-list* >

Forward all multicast traffic through the specified port.

For example, suppose you wanted to configure IGMP as follows for VLAN 1 on ports A1 - A6:

- Ports A1 - A2: Auto
- Ports A3 - A4: Forward
- Ports A5 - A6: Block

Depending on the privilege level, you could use one of the following commands to configure IGMP on VLAN 1 with the above settings:

```
ProCurve(config)# vlan 1
ProCurve(vlan-1)# ip igmp auto a1,a2
ProCurve(vlan-1)# ip igmp forward a3,a4
ProCurve(vlan-1)# ip igmp blocked a5,a6
```

The following command displays the VLAN and per-port configuration resulting from the above commands.

```
ProCurve> show ip igmp 1 config
```

Configuring IGMP Traffic Priority. This command allows you to prioritize IGMP traffic as either "high" or "normal" (the default).

Syntax: [no] vlan < *vid* > ip igmp high-priority-forward

> *Assigns "high" priority to IGMP traffic. The "***no***" form returns a high-priority setting to (the default) "normal" priority. (The switch services the traffic at its inbound priority.)*

```
ProCurve(config)# vlan 1 ip igmp high-priority-forward
```

> *This example configures high priority for IGMP traffic on VLAN 1.*

```
ProCurve(vlan-1)# ip igmp high-priority-forward
```

> *Same as above command, but in the VLAN 1 context level.*

```
ProCurve(vlan 1)# no ip igmp high-priority-forward
```

> *Returns IGMP traffic to "normal" priority.*

```
ProCurve> show ip igmp config
```

> *Show command to display results of above high-priority commands.*

Configuring the Querier Function. In its default configuration, the switch is capable of operating as an IGMP querier. This command lets you disable or re-enable this function.

Syntax: [no] vlan <vid> ip igmp querier

> *Disables or re-enables the ability for the switch to become querier, if necessary, on the specified VLAN. The default querier capability is "enabled".*

```
ProCurve(config)# no vlan 1 ip igmp querier
```

> *Disables the querier function on VLAN 1.*

```
ProCurve> show ip igmp config
```

> *This is the show command used to display results of the above querier command.*

Web: Enabling or Disabling IGMP

In the web browser interface you can enable or disable IGMP on a per-VLAN basis. To configure other IGMP features, telnet to the switch console and use the CLI.

To Enable or Disable IGMP:

1. Click on the **Configuration** tab.
2. Click on the **Device Features** button.
3. If more than one VLAN is configured, use the VLAN pull-down menu to select the VLAN on which you want to enable or disable IGMP.
4. Use the Multicast Filtering (IGMP) menu to enable or disable IGMP.
5. Click on **Apply Changes** button to implement the configuration change.

For web-based help on how to use the web browser interface screen, click on the **?** button provided on the web browser screen.

How IGMP Operates

The Internet Group Management Protocol (IGMP) is an internal protocol of the Internet Protocol (IP) suite. IP manages multicast traffic by using switches, multicast routers, and hosts that support IGMP. (In ProCurve's implementation of IGMP, a multicast router is not necessary as long as a switch is configured to support IGMP with the **querier** feature enabled.) A set of hosts, routers, and/or switches that send or receive multicast data streams to or from the same source(s) is termed a *multicast group*, and all devices in the group use the same multicast group address.

Message Types

The multicast group running version 2 of IGMP uses three fundamental types of messages to communicate:

- **Query:** A message sent from the querier (multicast router or switch) asking for a response from each host belonging to the multicast group. If a multicast router supporting IGMP is not present, then the switch must assume this function in order to elicit group membership information from the hosts on the network. (If you need to disable the querier feature, you can do so through the CLI, using the IGMP configuration MIB. See “Configuring the Querier Function” on page 4-10.)

- **Report (Join):** A message sent by a host to the querier to indicate that the host wants to be or is a member of a given group indicated in the report message.
- **Leave Group:** A message sent by a host to the querier to indicate that the host has ceased to be a member of a specific multicast group.

IGMP Operating Notes

IGMP identifies members of a multicast group (within a subnet) and allows IGMP-configured hosts (and routers) to join or leave multicast groups based on the following process.

- An IP multicast packet includes the multicast group (address) to which the packet belongs.
- When an IGMP client connected to a switch port needs to receive multicast traffic from a specific group, it joins the group by sending an IGMP report (join request) to the network. (The multicast group specified in the join request is determined by the requesting application running on the IGMP client.)
- When a networking device with IGMP enabled receives the join request for a specific group, it forwards any IP multicast traffic it receives for that group through the port on which the join request was received.
- When the client is ready to leave the multicast group, it sends a Leave Group message to the network and ceases to be a group member.
- When the leave request is detected, the appropriate IGMP device will cease transmitting traffic for the designated multicast group through the port on which the leave request was received (as long as there are no other current members of that group on the affected port).

Displaying IGMP Data.

To display data showing active group addresses, reports, queries, querier access port, and active group address data (port, type, and access), see the appendix on monitoring and analyzing switch operation in the *Management and Configuration Guide*.

Supported Standards and RFCs

ProCurve's implementation of IGMP supports the following standards and operating capabilities:

- RFC2236 (IGMP V.2, with backwards support for IGMP V.1)
- IETF draft for IGMP and MLD snooping switches (for IGMP V1, V2 V3)
- Full IGMPv2 support as well as full support for IGMPv1 Joins.
- Ability to operate in IGMPv2 Querier mode on VLANs with an IP address.

The ProCurve implementation is subject to the following restrictions:

- Interoperability with RFC3376 (IGMPv3)
- Interoperability with IGMPv3 Joins. When the switch receives an IGMPv3 Join, it accepts the host request and begins forwarding the IGMP traffic. This means ports that have not joined the group and are not connected to routers or the IGMP Querier will not receive the group's multicast traffic.
- No support for the IGMPv3 "Exclude Source" or "Include Source" options in the Join Reports. Rather, the group is simply joined from all sources.
- No support for becoming a version 3 Querier. The switch will become a version 2 Querier in the absence of any other Querier on the network.

Note

IGMP is supported in the HP MIB, rather than the standard IGMP MIBs, as the latter reduce Group Membership detail in switched environments.

Operation With or Without IP Addressing

You can configure IGMP on VLANs that do not have IP addressing. The benefit of IGMP without IP addressing is a reduction in the number of IP addresses you have to use and configure. This can be significant in a network with a large number of VLANs. The limitation on IGMP without IP addressing is that the switch cannot become Querier on any VLANs for which it has no IP address—so the network administrator must ensure that another IGMP device will act as Querier. It is also advisable to have an additional IGMP device available as a backup Querier. See the following table.

Table 4-1.Comparison of IGMP Operation With and Without IP Addressing

IGMP Function Available With IP Addressing Configured on the VLAN	Available *Without* IP Addressing?	Operating Differences Without an IP Address
Forward multicast group traffic to any port on the VLAN that has received a join request for that multicast group.	Yes	None
Forward join requests (reports) to the Querier.	Yes	None
Configure individual ports in the VLAN to **Auto** (the default)/**Blocked**, or **Forward**.	Yes	None
Configure IGMP traffic forwarding to normal or high-priority forwarding.	Yes	None
Age-Out IGMP group addresses when the last IGMP client on a port in the VLAN leaves the group.	Yes	Requires that another IGMP device in the VLAN has an IP address and can operate as Querier. This can be a multicast router or another switch configured for IGMP operation. (ProCurve recommends that the VLAN also include a device operating as a backup Querier in case the device operating as the primary Querier fails for any reason.
Support Fast-Leave IGMP (below) and Forced Fast-Leave IGMP (page 4-14).	Yes	
Support automatic Querier election.	No	Querier operation not available.
Operate as the Querier.	No	Querier operation not available.
Available as a backup Querier.	No	Querier operation not available.

Automatic Fast-Leave IGMP

IGMP Operation Presents a "Delayed Leave" Problem. Where multiple IGMP clients are connected to the same port on an IGMP device (switch or router), if only one IGMP client joins a given multicast group, then later sends a Leave Group message and ceases to belong to that group, the IGMP device retains that IGMP client in its IGMP table and continues forwarding IGMP traffic to the IGMP client until the Querier triggers confirmation that no other group members exist on the same port. This means that the switch continues to transmit unnecessary multicast traffic through the port until the Querier renews the multicast group status.

Fast-Leave IGMP. Depending on the switch model, Fast-Leave is enabled or disabled in the default configuration.

Table 4-2.IGMP: Data-Driven and Non-Data Driven Behavior

Switch Model or Series	Data-Driven IGMP Included?	IGMP Fast-Leave Setting	Default IGMP Behavior
Switch 5300 Switch 2800 Switch 2810 Switch 2500	Yes	Always Enabled	Drops unjoined multicast traffic except for always-forwarded traffic toward the Querier or multicast routers, and out of IGMP-forward ports. Selectively forwards joined multicast traffic.
Switch 2600 Switch 2600-PWR Switch 4100 Switch 6108	No	Disabled in the Default Configuration	IGMP Fast-Leave disabled in the default configuration. Floods unjoined multicast traffic to all ports. Selectively forwards joined multicast traffic.

On switches that do not support Data-Driven IGMP, unregistered multicast groups are flooded to the VLAN rather than pruned. In this scenario, Fast-Leave IGMP can actually increase the problem of multicast flooding by removing the IGMP group filter before the Querier has recognized the IGMP Leave. The Querier will continue to transmit the multicast group during this short time, and because the group is no longer registered the switch will then flood the multicast group to all ports.

On ProCurve switches that do support Data-Driven IGMP ("Smart" IGMP), when unregistered multicasts are received the switch automatically filters (drops) them. Thus, the sooner the IGMP Leave is processed, the sooner this multicast traffic stops flowing.

Because of the multicast flooding problem mentioned above, the IGMP FastLeave feature is disabled by default on all ProCurve switches that *do not* support Data-Driven IGMP. (See table 4-2, above.) The feature can be enabled on these switches via an SNMP set of this object:

```
hpSwitchIgmpPortForceLeaveState.< vid >.< port number>
```

However, this is not recommended as this will increase the amount of multicast flooding during the period between the client's IGMP Leave and the Querier's processing of that Leave. For more on this topic, refer to "Forced Fast-Leave IGMP" on page 4-17.

ProCurve recommends that the following settings be used.

- Use Delayed Group Flush on the Series 2600 switches whenever Fast Leave or Forced Fast Leave are set on a port (see page 4-17).
- Forced fast leave can be used when there are multiple devices attached to a port.

Automatic Fast-Leave Operation. If a switch port is:

a. Connected to only one end node

b. The end node currently belongs to a multicast group; i.e. is an IGMP client

c. The end node subsequently leaves the multicast group

Then the switch does not need to wait for the Querier status update interval, but instead immediately removes the IGMP client from its IGMP table and ceases transmitting IGMP traffic to the client. (If the switch detects multiple end nodes on the port, automatic Fast-Leave does not activate—regardless of whether one or more of these end nodes are IGMP clients.)

In the next figure, automatic Fast-Leave operates on the switch ports for IGMP clients "3A" and "5A", but not on the switch port for IGMP clients "7A" and 7B, Server "7C", and printer "7D".

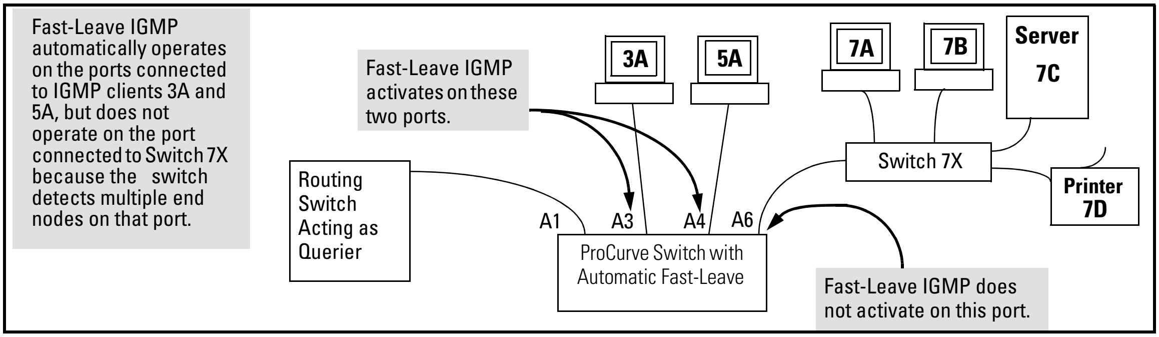

Fast-Leave IGMP automatically operates on the ports connected to IGMP clients 3A and 5A, but does not operate on the port connected to Switch 7X because the switch detects multiple end nodes on that port.
Fast-Leave IGMP activates on these two ports.
3A
5A
7A
7B
Server 7C
Switch 7X
Printer 7D
Routing Switch Acting as Querier
A1
A3
A4
A6
ProCurve Switch with Automatic Fast-Leave
Fast-Leave IGMP does not activate on this port.

Figure 4-3. Example of Automatic Fast-Leave IGMP Criteria

When client "3A" running IGMP is ready to leave the multicast group, it transmits a Leave Group message. Because the switch knows that there is only one end node on port A3, it removes the client from its IGMP table and halts multicast traffic (for that group) to port A3. If the switch is not the Querier, it does not wait for the actual Querier to verify that there are no other group members on port A3. If the switch itself is the Querier, it does not query port A3 for the presence of other group members.

Note that Fast-Leave operation does not distinguish between end nodes on the same port that belong to different VLANs. Thus, for example, even if all of the devices on port A6 in figure 4-3 belong to different VLANs, Fast-Leave does not operate on port A6.

Using Delayed Group Flush

This feature continues to filter IGMP-Left groups for a specified additional period of time. The delay in flushing the group filter prevents stale traffic from being forwarded by the server. Delayed Group Flush is enabled or disabled for the entire switch.

Syntax: igmp delayedflush <*time period*>

Enables the switch to continue to flush IGMP-Left groups for a specified period of time (0 - 255 seconds). The default setting is **Disabled**. *To disable, reset the time period to zero.*

Syntax: Show igmp delayedflush

Displays the current setting for the switch.

Forced Fast-Leave IGMP

Forced Fast-Leave IGMP speeds up the process of blocking unnecessary IGMP traffic to a switch port that is connected to multiple end nodes. (This feature does not activate on ports where the switch detects only one end node). For example, in figure 4-3, even if you configured Forced Fast-Leave on all ports in the switch, the feature would activate only on port A6 (which has multiple end nodes) when a Leave Group request arrived on that port.

When a port having multiple end nodes receives a Leave Group request from one end node for a given multicast group "X", Forced Fast-Leave activates and waits a small amount of time to receive a join request from any other group "X" member on that port. If the port does not receive a join request for that group within the forced-leave interval, the switch then blocks any further group "X" traffic to the port.

Setting Fast-Leave and Forced Fast-Leave from the CLI

In previous software versions, Fast-Leave and Forced Fast-Leave options for a port were set exclusively through the MIB. The following commands now allow a port to be configured for Fast-Leave or Forced Fast-leave operation from the CLI. Note that these commands must be executed in a VLAN context

Syntax: [no] ip igmp fastleave <*port-list*>

Enables IGMP Fast-Leaves on the specified ports in the VLAN (the default setting). In the Config context, use the VLAN specifier, for example, **vlan <** *vid* **> ip igmp fastleave <***port-list***>**. *The "no" form disables Fast-Leave on the specified ports.*

[no] ip igmp forcedfastleave <*port-list*>

Forces IGMP Fast-Leaves on the specified ports in the VLAN, even if they are cascaded.

To view the IGMP Fast-Leave status of a port use the **show running-config** or **show configuration** commands.

Setting Forced Fast-Leave Using the MIB

Fast-Leave and Forced Fast-Leave options for a port can also be set through the switch's MIB (Management Information Base).

Feature	Default	Settings	Function
Forced Fast-Leave state	**2** (disabled)	**1** (enabled) **2** (disabled)	Uses the **setmib** command to enable or disable Forced Fast-Leave on individual ports. When enabled on a port, Forced Fast-Leave operates only if the switch detects multiple end nodes (and at least one IGMP client) on that port.

Note on VLAN Numbers

In the ProCurve switches covered in this guide, the **walkmib** and **setmib** commands use an internal VLAN number (and not the VLAN ID, or VID) to display or change many per-vlan features, such as the Forced Fast-Leave state. Because the internal VLAN number for the default VLAN is always 1 (regardless of whether VLANs are enabled on the switch), and because a discussion of internal VLAN numbers for multiple VLANs is beyond the scope of this manual, this section concentrates on examples that use the default VLAN.

Listing the MIB-Enabled Forced Fast-Leave Configuration

The Forced Fast-Leave configuration data is available in the switch's MIB, and includes the state (enabled or disabled) for each port and the Forced-Leave Interval for all ports on the switch.

To List the Forced Fast-Leave State for all Ports in the Switch. In the CLI, use the **walkmib** command, as shown below.

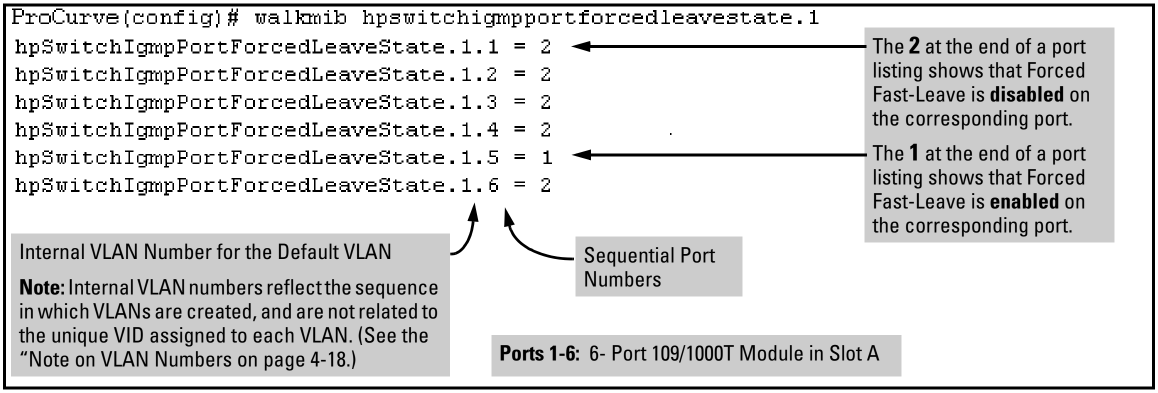
1. Enter either of the following walkmib command options:

```
walkmib hpSwitchIgmpPortForcedLeaveState
```

- OR -

```
walkmib 1.3.6.1.4.1.11.2.14.11.5.1.7.1.15.3.1.5
```

The resulting display lists the Forced Fast-Leave state for all ports in the switch, by VLAN. (A port belonging to more than one VLAN will be listed once for each VLAN, and if multiple VLANs are *not* configured, all ports will be listed as members of the default VLAN.) The following command produces a listing such as that shown in figure 4-4:



```
ProCurve(config)# walkmib hpswitchigmpportforcedleavestate.1
hpSwitchIgmpPortForcedLeaveState.1.1 = 2
hpSwitchIgmpPortForcedLeaveState.1.2 = 2
hpSwitchIgmpPortForcedLeaveState.1.3 = 2
hpSwitchIgmpPortForcedLeaveState.1.4 = 2
hpSwitchIgmpPortForcedLeaveState.1.5 = 1
hpSwitchIgmpPortForcedLeaveState.1.6 = 2
```

The **2** at the end of a port listing shows that Forced Fast-Leave is **disabled** on the corresponding port.

The **1** at the end of a port listing shows that Forced Fast-Leave is **enabled** on the corresponding port.

Internal VLAN Number for the Default VLAN

Note: Internal VLAN numbers reflect the sequence in which VLANs are created, and are not related to the unique VID assigned to each VLAN. (See the "Note on VLAN Numbers on page 4-18.)

Sequential Port Numbers

Ports 1-6: 6- Port 109/1000T Module in Slot A

Figure 4-4. Example of a Forced Fast-Leave Listing where all Ports are Members of the Default VLAN

To List the Forced Fast-Leave State for a Single Port. (See the "Note on VLAN Numbers" on page 4-18.)

Go to the switch's command prompt and use the **getmib** command, as shown below.

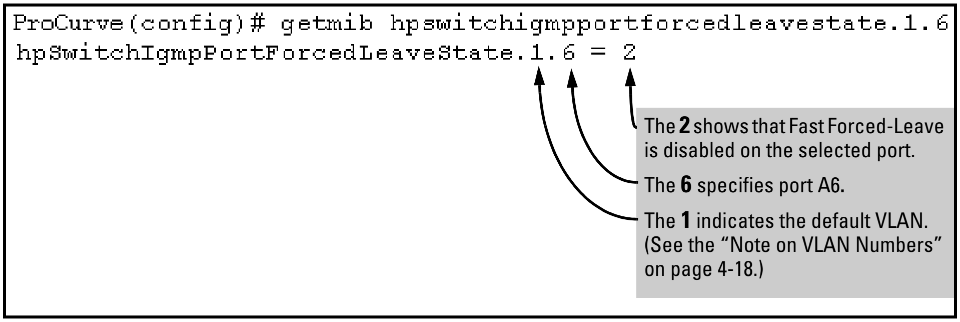
Syntax:

getmib hpSwitchIgmpPortForcedLeaveState.<*vlan number*><*.port number*>

- OR -

getmib 1.3.6.1.4.1.11.2.14.11.5.1.7.1.15.3.1.5.<*vlan number*><*.port number*>

For example, the following command to list the state for port A6 (which, in this case, belongs to the default VLAN) produces the indicated listing:



```
ProCurve(config)# getmib hpswitchigmpportforcedleavestate.1.6
hpSwitchIgmpPortForcedLeaveState.1.6 = 2
```

The **2** shows that Fast Forced-Leave is disabled on the selected port.

The **6** specifies port A6.

The **1** indicates the default VLAN. (See the "Note on VLAN Numbers" on page 4-18.)

Figure 4-5. Example Listing the Forced Fast-Leave State for a Single Port on the Default VLAN

Configuring Per-Port Forced Fast-Leave IGMP

In the factory-default configuration, Forced Fast-Leave is disabled for all ports on the switch. To enable (or disable) this feature on individual ports, use the switch's **setmib** command, as shown below.

Configuring Per-Port Forced Fast-Leave IGMP on Ports. This procedure enables or disables Forced Fast-Leave on ports in a given VLAN. (See the "Note on VLAN Numbers" on page 4-18.)

Syntax:

setmib hpSwitchIgmpPortForcedLeaveState.< *vlan number* >< *.port number* > -i < 1 | 2 >

- OR -

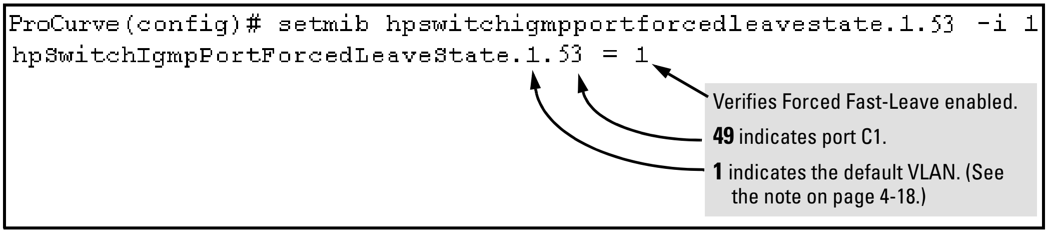
setmib 1.3.6.1.4.1.11.2.14.11.5.1.7.1.15.3.1.5.< *vlan number* >< *.port number* > -i < 1 | 2 >

where:

1 = Forced Fast-Leave enabled

2 = Forced Fast-Leave disabled

For example, suppose that your switch has a six-port gigabit module in slot A, and port C1 is a member of the default VLAN. In this case, the port number is "49" (In the MIB, slot A = ports 1-24; slot B = ports 25-48; slot C = ports 49-72, and so on.) To enable Forced Fast-Leave on C6 (53), you would execute the following command and see the indicated result:



```
ProCurve(config)# setmib hpswitchigmpportforcedleavestate.1.53 -i 1
hpSwitchIgmpPortForcedLeaveState.1.53 = 1
```

Verifies Forced Fast-Leave enabled.

49 indicates port C1.

1 indicates the default VLAN. (See the note on page 4-18.)

Figure 4-6. Example of Changing the Forced Fast-Leave Configuration on Port 49

Using the Switch as Querier

Querier Operation

The function of the IGMP Querier is to poll other IGMP-enabled devices in an IGMP-enabled VLAN to elicit group membership information. The switch performs this function if there is no other device in the VLAN, such as a multicast router, to act as Querier. Although the switch automatically ceases Querier operation in an IGMP-enabled VLAN if it detects another Querier on the VLAN, you can also use the Command Prompt to disable the Querier capability for that VLAN.

Note

A Querier is required for proper IGMP operation. For this reason, if you disable the Querier function on a switch, ensure that there is an IGMP Querier (and, preferably, a backup Querier) available on the same VLAN.

If the switch becomes the Querier for a particular VLAN (for example, the DEFAULT_VLAN), then subsequently detects queries transmitted from another device on the same VLAN, the switch ceases to operate as the Querier for that VLAN. If this occurs, the switch Event Log lists a pair of messages similar to these:

```
I 01/15/01 09:01:13 igmp: DEFAULT_VLAN: Other Querier detected

I 01/15/01 09:01:13 igmp: DEFAULT_VLAN: This switch is no longer
Querier
```

In the above scenario, if the other device ceases to operate as a Querier on the default VLAN, then the switch detects this change and can become the Querier as long as it is not pre-empted by some other IGMP Querier on the VLAN. In this case, the switch Event Log lists messages similar to the following to indicate that the switch has become the Querier on the VLAN:

```
I 01/15/01 09:21:55 igmp: DEFAULT_VLAN: Querier Election in
process

I 01/15/01 09:22:00 igmp: DEFAULT_VLAN: This switch has been
elected as Querier
```

Excluding Multicast Addresses from IP Multicast Filtering

Each multicast host group is identified by a single IP address in the range of 224.0.0.0 through 239.255.255.255. Specific groups of consecutive addresses in this range are termed “well-known” addresses and are reserved for predefined host groups. IGMP does not filter these addresses, so any packets the switch receives for such addresses are flooded out all ports assigned to the VLAN on which they were received (except the port on which the packets entered the VLAN).

The following table lists the 32 well-known address groups (8192 total addresses) that IGMP does not filter on the switches covered in this guide, as well as on the 1600M, 2400M, 2424M, 2650M, 4000M, 6108M, 8000M, and Switch 2500 Series devices.

Table 4-3.IP Multicast Address Groups Excluded from IGMP Filtering

Groups of Consecutive Addresses in the Range of 224.0.0.*X* to 239.0.0.*X**		Groups of Consecutive Addresses in the Range of 224.128.0.*X* to 239.128.0.*X**	
224.0.0.x	232.0.0.x	224.128.0.x	232.128.0.x
225.0.0.x	233.0.0.x	225.128.0.x	233.128.0.x
226.0.0.x	234.0.0.x	226.128.0.x	234.128.0.x
227.0.0.x	235.0.0.x	227.128.0.x	235.128.0.x
228.0.0.x	236.0.0.x	228.128.0.x	236.128.0.x
229.0.0.x	237.0.0.x	229.128.0.x	237.128.0.x
230.0.0.x	238.0.0.x	230.128.0.x	238.128.0.x
231.0.0.x	239.0.0.x	231.128.0.x	239.128.0.x

* X is any value from 0 to 255.

Notes:

IP Multicast Filters. *This operation applies to the ProCurve Switch 1600M, 2400M, 2424M, 4000M, and 8000M, but not to the Switch 2500, 2600, 2600-PWR, 2800, 2810, 4100, and 5300 Series devices or the Switch 6108 (which do not have static multicast traffic/security filters).*

IP multicast addresses occur in the range from 224.0.0.0 through 239.255.255.255 (which corresponds to the Ethernet multicast address range of 01005e-000000 through 01005e-7fffff). Where a switch has a static Traffic/ Security filter configured with a "Multicast" filter type and a "Multicast Address" in this range, the switch will use the static filter unless IGMP learns of a multicast group destination in this range. In this case, IGMP dynamically takes over the filtering function for the multicast destination address(es) for as long as the IGMP group is active. If the IGMP group subsequently deactivates, the switch returns filtering control to the static filter.

Reserved Addresses Excluded from IP Multicast (IGMP) Filtering. Traffic to IP multicast groups in the IP address range of 224.0.0.0 to 224.0.0.255 will always be flooded because addresses in this range are "well known" or "reserved" addresses. Thus, if IP Multicast is enabled and there is an IP multicast group within the reserved address range, traffic to that group will be flooded instead of filtered by the switch.

Number of IP Multicast Addresses Allowed. Multicast filters and IGMP filters (addresses) together can total up to 255 in the switch. If multiple VLANs are configured, then each filter is counted once per VLAN in which it is used.

5

Multiple Instance Spanning-Tree Operation

Contents

Overview

MSTP Features

802.1s Spanning Tree Protocol	Default	Menu	CLI	Web
Viewing the MSTP Status and Configuration	n/a	—	page 5-40	—
Enable/Disable MSTP and Configure Global Parameters	Disabled	—	page 5-19	—
Configuring Basic Port Connectivity Parameters	admin-edge-port: No-disabled auto-edge-port: Yes-enabled bpdu-filter: No-disabled bpdu-protection: No-disabled mcheck: Yes hello-time: 2 path-cost: auto point-to-point MAC: Force-True priority: 128 (multiplier: 8) root-guard: No-disabled tcn-guard: No-disabled loop-protection: Send disable	—	page 5-23 and following	—
Configuring MSTP Instance Parameters	instance (MSTPI): none priority: 32768 (multiplier: 8)	—	page 5-32	—
Configuring MSTP Instance Per-Port Parameters	Auto	—	page 5-35	—
Enabling/Disabling MSTP Spanning Tree Operation	Disabled	—	page 5-38	—
Enabling an Entire MST Region at Once	n/a	—	page 5-38	—

Without spanning tree, having more than one active path between a pair of nodes causes loops in the network, which can result in duplication of messages, leading to a “broadcast storm” that can bring down the network.

Multiple-Instance spanning tree operation (802.1s) ensures that only one active path exists between any two nodes in a spanning-tree *instance*. A spanning-tree instance comprises a unique set of VLANs, and belongs to a specific spanning-tree *region*. A region can comprise multiple spanning-tree instances (each with a different set of VLANs), and allows one active path among regions in a network. Applying VLAN tagging to the ports in a multiple-instance spanning-tree network enables blocking of redundant links in one instance while allowing forwarding over the same links for non-redundant use by another instance. For example, suppose you have three switches in a region configured with VLANs grouped into two instances, as follows:

VLANs	Instance 1	Instance 2
10, 11, 12	Yes	No
20, 21, 22	No	Yes

The logical and physical topologies resulting from these VLAN/Instance groupings result in blocking on different links for different VLANs:

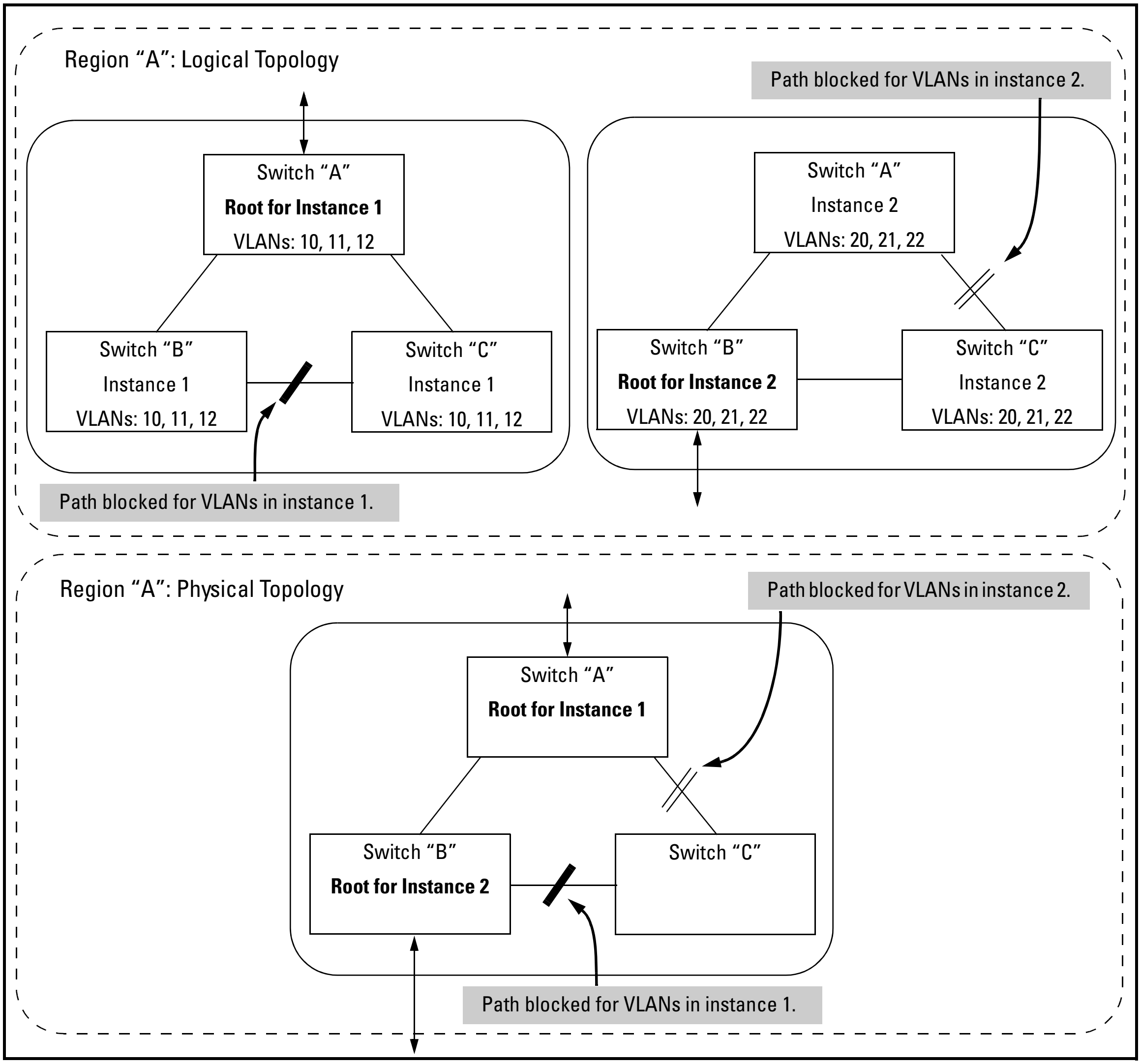


Figure 5-1. Example of a Multiple Spanning-Tree Application

Note on Path Cost

RSTP and MSTP implements a greater range of path costs and new default path cost values to account for higher network speeds. These values are different than the values defined by 802.1D STP as shown below.

Port Type	802.1D STP Path Cost	RSTP and MSTP Path Cost
10 Mbps	100	2 000 000
100 Mbps	10	200 000
1 Gbps	5	20 000

Because the maximum value for the path cost allowed by 802.1D STP is 65535, devices running that version of spanning tree cannot be configured to match the values defined by MSTP, at least for 10 Mbps and 100 Mbps ports. In LANs where there is a mix of devices running 802.1D STP, RSTP, and/or MSTP, you should reconfigure the devices so the path costs match for ports with the same network speeds.

802.1s Multiple Spanning Tree Protocol (MSTP)

The 802.1D and 802.1w spanning tree protocols operate without regard to a network's VLAN configuration, and maintain one common spanning tree throughout a bridged network. Thus, these protocols map one loop-free, logical topology on a given physical topology. The 802.1s Multiple Spanning Tree protocol (MSTP) uses VLANs to create multiple spanning trees in a network, which significantly improves network resource utilization while maintaining a loop-free environment.

While the per-VLAN spanning tree approach adopted by some vendors overcomes the network utilization problems inherent in using STP or RSTP, using a per-VLAN technology with multiple VLANs can overload the switch's CPU. MSTP on the switches covered by this guide complies with the IEEE 802.1s standard, and extends STP and RSTP functionality to map multiple independent spanning tree instances onto a physical topology. With MSTP, each spanning tree instance can include one or more VLANs and applies a separate, per-instance forwarding topology. Thus, where a port belongs to multiple VLANs, it may be dynamically blocked in one spanning tree instance, but forwarding in another instance. This achieves load-balancing across the network while keeping the switch's CPU load at a moderate level (by aggregating multiple VLANs in a single spanning tree instance). Like RSTP, MSTP provides fault tolerance through rapid, automatic reconfiguration if there is a failure in a network's physical topology.

Caution

Spanning tree interprets a switch mesh as a single link. Because the switch automatically gives faster links a higher priority, the default MSTP parameter settings are usually adequate for spanning tree operation. Also, because incorrect MSTP settings can adversely affect network performance, you should not change the MSTP settings from their default values unless you have a strong understanding of how spanning tree operates.

In a mesh environment, the default MSTP timer settings (**Hello Time** and **Forward Delay**) are usually adequate for MSTP operation. Because a packet crossing a mesh may traverse several links within the mesh, using smaller-than-default settings for the MSTP **Hello Time** and **Forward Delay** timers can cause unnecessary topology changes and end-node connectivity problems.

For MSTP information beyond what is provided in this manual, refer to the IEEE 802.1s standard.

MSTP Structure

MSTP maps active, separate paths through separate spanning tree instances and between MST regions. Each MST region comprises one or more MSTP switches. Note that MSTP recognizes an STP or RSTP LAN as a distinct spanning-tree region.

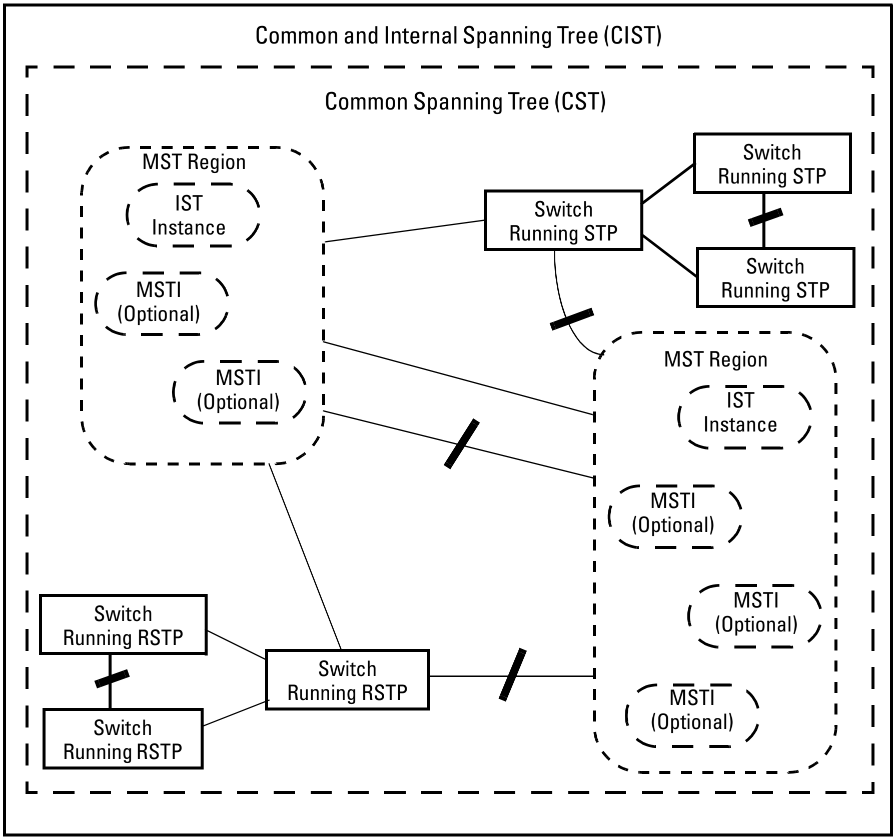


Figure 5-2. Example of MSTP Network with Legacy STP and RSTP Devices Connected

Common and Internal Spanning Tree (CIST): The CIST identifies the regions in a network and administers the CIST root bridge for the network, the root bridge for each region, and the root bridge for each spanning-tree instance in each region.

Common Spanning Tree (CST): The CST administers the connectivity among the MST regions, STP LANs, and RSTP LANs in a bridged network.

MST Region: An MST region comprises the VLANs configured on physically connected MSTP switches. All switches in a given region must be configured with the same VLANs and Multiple Spanning Tree Instances (MSTIs).

Internal Spanning Tree (IST): The IST administers the topology within a given MST region. When you configure a switch for MSTP operation, the switch automatically includes all of the static VLANs configured on the switch in a single, active spanning tree topology (instance) within the IST. This is termed the "IST instance". Any VLANs you subsequently configure on the switch are added to this IST instance. To create separate forwarding paths within a region, group specific VLANs into different Multiple Spanning Tree Instances (MSTIs). (Refer to "Multiple Spanning Tree Instance", below.)

Types of Multiple Spanning Tree Instances: A multiple spanning tree network comprises separate spanning-tree instances existing in an MST region. (There can be multiple regions in a network.) Each instance defines a single forwarding topology for an exclusive set of VLANs. By contrast, an STP or RSTP network has only one spanning tree instance for the entire network, and includes all VLANs in the network. (An STP or RSTP network operates as a single-instance network.) A region can include two types of STP instances:

- **Internal Spanning-Tree Instance (IST Instance):** This is the default spanning tree instance in any MST region. It provides the root switch for the region and comprises all VLANs configured on the switches in the region that are not specifically assigned to Multiple Spanning Tree Instances (MSTIs, described below). All VLANs in the IST instance of a region are part of the same, single spanning tree topology, which allows only one forwarding path between any two nodes belonging to any of the VLANs included in the IST instance. All switches in the region must belong to the set of VLANs that comprise the IST instance. Note that the switch automatically places dynamic VLANs (resulting from GVRP operation) in the IST instance. Dynamic VLANs cannot exist in an MSTI (described below).
- **MSTI (Multiple Spanning Tree Instance):** This type of configurable spanning tree instance comprises all static VLANs you specifically assign to it, and must include at least one VLAN. The VLAN(s) you assign to an MSTI must initially exist in the IST instance of the same MST region. When you assign a static VLAN to an MSTI, the switch removes the VLAN from the IST instance. (Thus, you can assign a VLAN to only one MSTI in a given region.) All VLANs in an MSTI operate as part of the same single spanning tree topology. (The switch does not allow dynamic VLANs in an MSTI.)

Caution

When you enable MSTP on the switch, the default MSTP spanning tree configuration settings comply with the values recommended in the IEEE 802.1s Multiple Spanning Tree Protocol (MSTP) standard. Note that inappropriate changes to these settings can result in severely degraded network performance. For this reason, *ProCurve strongly recommends that changing these default settings be reserved only for experienced network administrators who have a strong understanding of the IEEE 802.1D/w/s standards and operation.*

How MSTP Operates

In the factory default configuration, spanning tree operation is off. Also, the switch retains its currently configured spanning tree parameter settings when disabled. Thus, if you disable spanning tree, then later re-enable it, the parameter settings will be the same as before spanning tree was disabled. The switch also includes a "Pending" feature that enables you to exchange MSTP configurations with a single command. (Refer to "Enabling an Entire MST Region at Once or Exchanging One Region Configuration for Another" on page 5-38.)

Note

The switch automatically senses port identity and type, and automatically defines spanning-tree parameters for each type, as well as parameters that apply across the switch. Although these parameters can be adjusted, *ProCurve strongly recommends leaving these settings in their default configurations unless the proposed changes have been supplied by an experienced network administrator who has a strong understanding of the IEEE 802.1D/w/s standards and operation.*

MST Regions

All MSTP switches in a given region must be configured with the same VLANs. Also, each MSTP switch within the same region must have the same VLAN-to-instance assignments. (A VLAN can belong to only one instance within any region.) Within a region:

- All of the VLANs belonging to a given instance compose a single, active spanning-tree topology for that instance.
- Each instance operates independently of other regions.

Between regions there is a single, active spanning-tree topology.

How Separate Instances Affect MSTP Operation. Assigning different groups of VLANs to different instances ensures that those VLAN groups use independent forwarding paths. For example, in figure 5-3 each instance has a different forwarding path.

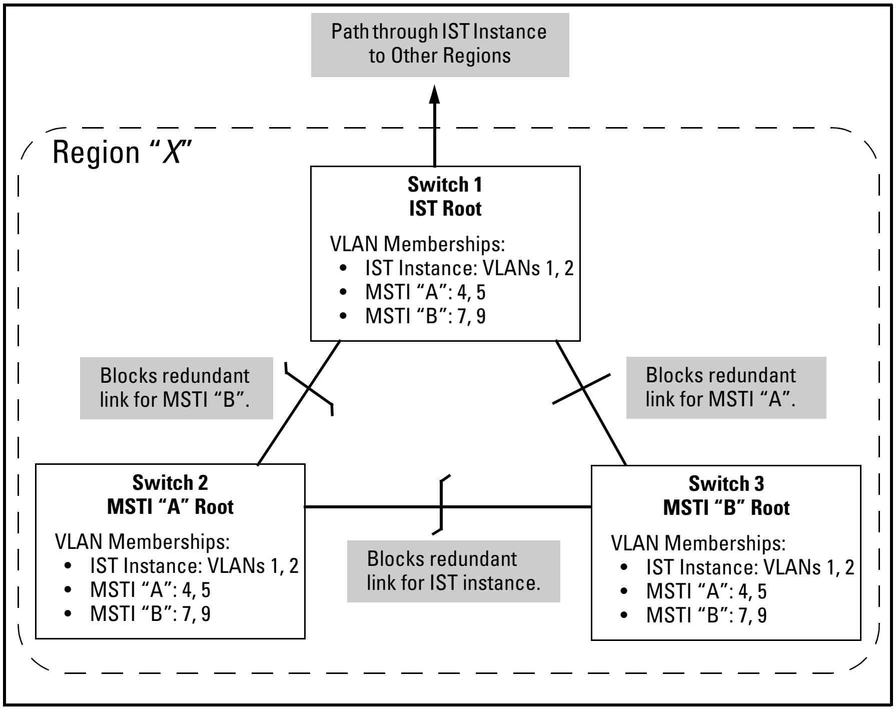

Path through IST Instance to Other Regions
Region "X"
Switch 1
IST Root
VLAN Memberships:
• IST Instance: VLANs 1, 2
• MSTI "A": 4, 5
• MSTI "B": 7, 9
Blocks redundant link for MSTI "B".
Blocks redundant link for MSTI "A".
Switch 2
MSTI "A" Root
VLAN Memberships:
• IST Instance: VLANs 1, 2
• MSTI "A": 4, 5
• MSTI "B": 7, 9
Blocks redundant link for IST instance.
Switch 3
MSTI "B" Root
VLAN Memberships:
• IST Instance: VLANs 1, 2
• MSTI "A": 4, 5
• MSTI "B": 7, 9

Figure 5-3. Active Topologies Built by Three Independent MST Instances

While allowing only one active path through a given instance, MSTP retains any redundant physical paths in the instance to serve as backups (blocked) paths in case the existing active path fails. Thus, if an active path in an instance fails, MSTP automatically activates (unblocks) an available backup to serve as the new active path through the instance for as long as the original active path is down. Note also that a given port may simultaneously operate in different states (forwarding or blocking) for different spanning-tree instances within the same region. This depends on the VLAN memberships to which the port is assigned. For example, if a port belongs to VLAN 1 in the IST instance of a region and also belongs to VLAN 4 in MSTI "*x*" in the same region, the port may apply different states to traffic for these two different instances.

Within a region, traffic routed between VLANs in separate instances can take only one physical path. To ensure that traffic in all VLANs within a region can travel between regions, all of the boundary ports for each region should belong to all VLANs configured in the region. Otherwise, traffic from some areas within a region could be blocked from moving to other regions.

All MSTP switches (as well as STP and RSTP switches) in a network use BPDUs (Bridge Protocol Data Units) to exchange information from which to build multiple, active topologies in the individual instances within a region and between regions. From this information:

- The MSTP switches in each LAN segment determine a designated bridge and designated port or trunk for the segment.
- The MSTP switches belonging to a particular instance determine the root bridge and root port or trunk for the instance.
- For the IST instance within a region, the MSTP switches linking that region to other regions (or to STP or RSTP switches) determine the IST root bridge and IST root port or trunk for the region. (For any Multiple Spanning-Tree instance—MSTI—in a region, the regional root may be a different switch that is not necessarily connected to another region.)
- The MSTP switches block redundant links within each LAN segment, across all instances, and between regions, to prevent any traffic loops.

As a result, each individual instance (spanning tree) within a region determines its regional root bridge, designated bridges, and designated ports or trunks.

Regions, Legacy STP and RSTP Switches, and the Common Spanning Tree (CST)

The IST instance and any MST instances in a region exist only within that region. Where a link crosses a boundary between regions (or between a region and a legacy STP or RSTP switch), traffic is forwarded or blocked as determined by the Common Spanning Tree (CST). The CST ensures that there is only one active path between any two regions, or between a region and a switch running STP and RSTP. (Refer to figure 5-2 on page 7.)

MSTP Operation with 802.1Q VLANs

As indicated in the preceding sections, within a given MST instance, a single spanning tree is configured for all VLANs included in that instance. This means that if redundant physical links exist in separate VLANs within the same instance, MSTP blocks all but one of those links. However, you can prevent the bandwidth loss caused by blocked redundant links for different VLANs in an instance by using a port trunk. The following example shows how you can use a port trunk with 802.1Q (tagged) VLANs and MSTP without unnecessarily blocking any links or losing any bandwidth.

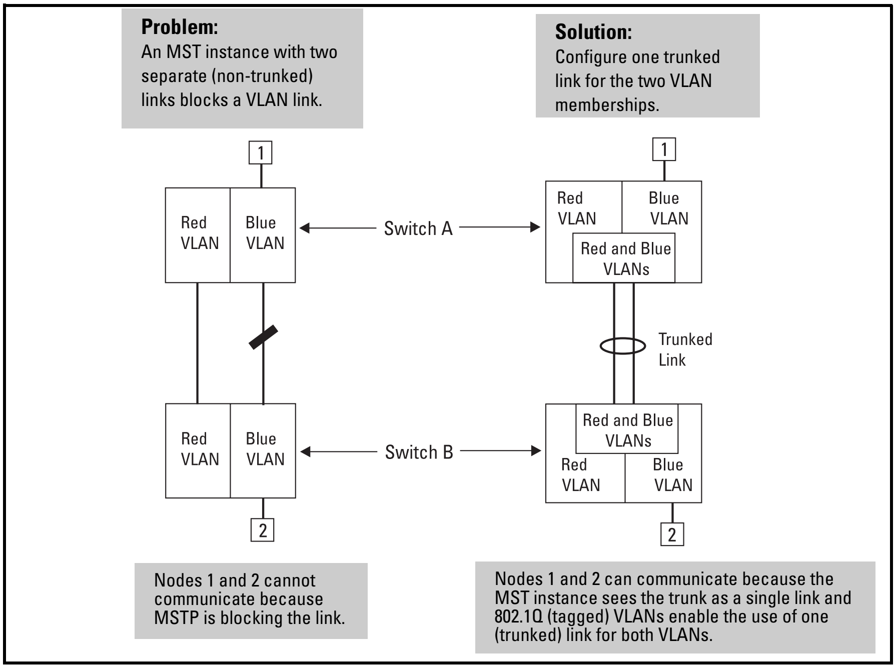

Problem:
An MST instance with two separate (non-trunked) links blocks a VLAN link.
Solution:
Configure one trunked link for the two VLAN memberships.
1
Red VLAN
Blue VLAN
Switch A
Red VLAN
Blue VLAN
Red and Blue VLANs
Trunked Link
Red and Blue VLANs
Red VLAN
Blue VLAN
Switch B
Red VLAN
Blue VLAN
2
Nodes 1 and 2 cannot communicate because MSTP is blocking the link.
Nodes 1 and 2 can communicate because the MST instance sees the trunk as a single link and 802.1Q (tagged) VLANs enable the use of one (trunked) link for both VLANs.

Figure 5-4. Example of Using a Trunked Link To Support Multiple VLAN Connectivity within the Same MST Instance

Note

All switches in a region should be configured with the VLANs used in that region, and all ports linking MSTP switches together should be members of all VLANs in the region. Otherwise, the path to the root for a given VLAN will be broken if MSTP selects a spanning tree through a link that does not include that VLAN.

Terminology

Bridge: See “MSTP Bridge”.

Common and Internal Spanning Tree (CIST): Comprises all LANs, STP, and RSTP bridges and MSTP regions in a network. The CIST automatically determines the MST regions in a network and defines the root bridge (switch) and designated port for each region. The CIST includes the Common Spanning Tree (CST), the Internal Spanning Tree (IST) within each region, and any multiple spanning-tree instances (MSTIs) in a region.

Common Spanning Tree (CST): Refers to the single forwarding path the switch calculates for STP (802.1D) and RSTP (802.1w) topologies, and for inter-regional paths in MSTP (802.1s) topologies. Note that all three types of spanning tree can interoperate in the same network. Also, the MSTP switch interprets a device running 802.1D STP or 802.1w RSTP as a separate region. (Refer to figure 5-2 on page 7.)

Internal Spanning Tree (IST): Comprises all VLANs within a region that are not assigned to a multiple spanning-tree instance configured within the region. All MST switches in a region should belong to the IST. In a given region "X", the IST root switch is the regional root switch and provides information on region "X" to other regions.

MSTP (Multiple Spanning Tree Protocol): A network supporting MSTP allows multiple spanning tree instances within configured regions, and a single spanning tree among regions, STP bridges, and RSTP bridges.

MSTP BPDU (MSTP Bridge Protocol Data Unit): These BPDUs carry region-specific information, such as the region identifier (region name and revision number). If a switch receives an MSTP BPDU with a region identifier that differs from its own, then the port on which that BPDU was received is on the boundary of the region in which the switch resides.

MSTP Bridge: In this manual, an MSTP bridge is a 2810 Switch (or another 802.1s-compatible device) configured for MSTP operation.

MST Region: An MST region forms a multiple spanning tree domain and is a component of a single spanning-tree domain within a network. For switches internal to the MST region:

- All switches have identical MST configuration identifiers (region name and revision number).
- All switches have identical VLAN assignments to the region's IST and (optional) MST instances.
- One switch functions as the designated bridge (IST root) for the region.
- No switch has a point-to-point connection to a bridging device that cannot process RSTP BPDUs.

Operating Rules

- All switches in a region must be configured with the same set of VLANs, as well as the same MST configuration name and MST configuration number.
- Within a region, a VLAN can be allocated to either a single MSTI or to the region's IST instance.
- All switches in a region must have the same VID-to-MST instance and VID-to-IST instance assignments.
- There is one root MST switch per configured MST instance.
- Within any region, the root switch for the IST instance is also the root switch for the region. Because boundary ports provide the VLAN connectivity between regions, all boundary ports on a region's root switch should be configured as members of all static VLANs defined in the region.
- There is one root switch for the Common and Internal Spanning Tree (CIST). Note that the per-port **hello-time** parameter assignments on the CIST root switch propagate to the ports on downstream switches in the network and override the **hello-time** configured on the downstream switch ports.
- Where multiple MST regions exist in a network, there is only one active, physical communication path between any two regions, or between an MST region and an STP or RSTP switch. MSTP blocks any other physical paths as long as the currently active path remains in service.
- Within a network, an MST region appears as a virtual RSTP bridge to other spanning tree entities (other MST regions, and any switches running 802.1D or 802.1w spanning-tree protocols).
- Within an MSTI, there is one spanning tree (one physical, communication path) between any two nodes. That is, within an MSTI, there is one instance of spanning tree, regardless of how many VLANs belong to the MSTI. Within an IST instance, there is also one spanning tree across all VLANs belonging to the IST instance.
- An MSTI comprises a unique set of VLANs and forms a single spanning-tree instance within the region to which it belongs.
- Communication between MST regions uses a single spanning tree.
- If a port on a switch configured for MSTP receives a legacy (STP/802.1D or RSTP/802.1w) BPDU, it automatically operates as a legacy port. In this case, the MSTP switch interoperates with the connected STP or RSTP switch as a separate MST region.
- Within an MST region, there is one logical forwarding topology per instance, and each instance comprises a unique set of VLANs. Where multiple paths exist between a pair of nodes using VLANs belonging to

the same instance, all but one of those paths will be blocked for that instance. However, if there are different paths in different instances, all such paths are available for traffic. Separate forwarding paths exist through separate spanning tree instances.

- A port can have different states (forwarding or blocking) for different instances (which represent different forwarding paths).
- MSTP interprets a switch mesh as a single link.
- A dynamic VLAN learned by GVRP will always be placed in the IST instance and cannot be moved to any configured MST instance.

Transitioning from STP or RSTP to MSTP

IEEE 802.1s MSTP includes RSTP functionality and is designed to be compatible with both IEEE 802.1D and 802.1w spanning-tree protocols. Even if all the other devices in your network are using STP, you can enable MSTP on the switches covered by this guide. Also, using the default configuration values, your switches will interoperate effectively with STP and RSTP devices. MSTP automatically detects when the switch ports are connected to non-MSTP devices in the spanning tree and communicates with those devices using 802.1D or 802.1w STP BPDU packets, as appropriate.

Because MSTP is so efficient at establishing the network path, ProCurve highly recommends that you update all of your switches to support 802.1s/MSTP. (For switches that do not support 802.1s/MSTP, ProCurve recommends that you update to RSTP to benefit from the convergence times of less than one second under optimal circumstances.) To make the best use of MSTP and achieve the fastest possible convergence times, there are some changes that you should make to the MSTP default configuration.

Note

Under some circumstances, it is possible for the rapid state transitions employed by MSTP and RSTP to result in an increase in the rates of frame duplication and misordering in the switched LAN. In order to allow MSTP and RSTP switches to support applications and protocols that may be sensitive to frame duplication and misordering, setting the Force Protocol Version parameter to **STP-compatible** allows MSTP and RSTP to operate with the rapid transitions disabled. The value of this parameter applies to all ports on the switch. See information on **force version** on page 5-17.

As indicated above, one of the benefits of MSTP and RSTP is the implementation of a larger range of port path costs, which accommodates higher network speeds. New default values have also been implemented for the path costs associated with the different network speeds. This can create some incompatibility between devices running the older 802.1D STP and your switch running MSTP or RSTP. Please see the “Note on Path Cost” on page 5-5 for more information on adjusting to this incompatibility.

Tips for Planning an MSTP Application

- Ensure that the VLAN configuration in your network supports all of the forwarding paths necessary for the desired connectivity. All ports connecting one switch to another within a region and one switch to another between regions should be configured as members of all VLANs configured in the region.
- All ports or trunks connecting one switch to another within a region should be configured as members of all VLANs in the region. Otherwise, some VLANs could be blocked from access to the spanning-tree root for an instance or for the region.
- Plan individual regions based on VLAN groupings. That is, plan on all MSTP switches in a given region supporting the same set of VLANs. Within each region, determine the VLAN membership for each spanning-tree instance. (Each instance represents a single forwarding path for all VLANs in that instance.)
- There is one logical spanning-tree path through the following:
  - Any inter-regional links
  - Any IST or MST instance within a region
  - Any legacy (802.1D or 802.1w) switch or group of switches. (Where multiple paths exist between an MST region and a legacy switch, expect the CST to block all but one such path.)
- Determine the root bridge and root port for each instance.
- Determine the designated bridge and designated port for each LAN segment.
- Determine which VLANs to assign to each instance, and use port trunks with 802.1Q VLAN tagging where separate links for separate VLANs would result in a blocked link preventing communication between nodes on the same VLAN. (Refer to “MSTP Operation with 802.1Q VLANs” on page 5-11.)
- Identify the edge ports connected to end nodes and enable the **admin-edge-port** setting for these ports. Leave the **admin-edge-port** setting disabled for ports connected to another switch, a bridge, or a hub.

Note on MSTP Rapid State Transitions

Under some circumstances the rapid state transitions employed by MSTP (and RSTP) can increase the rates of frame duplication and misordering in the switched LAN. To allow MSTP switches to support applications and protocols that may be sensitive to frame duplication and misordering, setting the Force Protocol Version (**force-version**) parameter to **stp-compatible** allows MSTP to operate with rapid transitions disabled. The value of this parameter applies to all ports on the switch. See the information on **force-version** on page 21.

Steps for Configuring MSTP

This section outlines the general steps for configuring MSTP operation in your network, and assumes you have already planned and configured the VLANs you want MSTP to use. The actual MSTP parameter descriptions are in the following sections.

Note

The switch supports MSTP configuration through the CLI. After you specify MSTP and reboot the switch as described above, the switch removes the **Spanning Tree** option from the Menu interface. If you later reconfigure the switch to use STP or RSTP, the switch returns the **Spanning Tree** option to the Menu interface.

This section assumes that you have already

1. Configured MSTP global parameters. This step involves configuring the following:
   - Required parameters for MST region identity:

     Region Name: **spanning-tree config-name**

     Region Revision Number: **spanning-tree config revision**
   - Optional MSTP parameter changes for region settings:

     ProCurve recommends that you leave these parameters at their default settings for most networks. Refer to the "Caution" on page 9.

     – The maximum number of hops before the MSTP BPDU is discarded (default: 20)
     spanning-tree max-hops

     – Force-Version operation
     spanning-tree force-version

     – Forward Delay
     spanning-tree forward-delay

- Hello Time (used if the switch operates as the root device.)
  spanning-tree hello-time

- Maximum age to allow for STP packets before discarding
  spanning-tree maximum-age

- Device spanning-tree priority. Specifies the priority value used along with the switch MAC address to determine which device is root. The lower a priority value, the higher the priority.
  spanning-tree priority

2. Configure MST instances.
   - Configure one instance for each VLAN group that you want to operate as an active topology within the region to which the switch belongs. When you create the instance, you must include a minimum of one VID. You can add more VIDs later if desired.
     spanning-tree instance < 1 - 16 > vlan < ***vid*** **>**

     To move a VLAN from one instance to another, first use **no spanning-tree instance <** ***n*** **> vlan <** ***vid*** **>** to unmap the VLAN from the current instance, then add the VLAN to the other instance. (While the VLAN is unmapped from an MSTI, it is associated with the region's IST instance.)

3. Configure the priority for each instance.

   spanning-tree instance < *n* > **priority** < *n* >

4. Configure MST instance port parameters. Enable **admin-edge-port** for ports connected to end nodes (page 23), but leave it disabled (the default) for connections to another switch, a bridge, or a hub. Set the path cost value for the port(s) used by a specific MST instance. Leaving this setting at the default auto allows the switch to calculate the path-cost from the link speed.
   spanning-tree instance < 1 - 16 | ist > port-list < port-list >

5. Enable spanning-tree operation on the switch.
   spanning-tree

Configuring MSTP Operation Mode and Global Parameters

Command	**Page**
spanning-tree config-name < *ascii-string* >	page 5-20
spanning-tree config-revision < *revision-number* >	page 5-20
spanning-tree max-hops < *hop-count* >	page 5-21
spanning-tree maximum-age	page 5-21
spanning-tree force-version < stp-compatible | rstp-operation | mstp-operation>	page 5-21
spanning-tree hello-time < 1..10 >	page 5-22
spanning-tree legacy-mode	page 5-22
spanning-tree legacy-path-cost	page 5-22

The commands in this section apply on the switch level, and do not affect individual port configurations.

Syntax: [no] spanning-tree config-name < *ascii-string* >

This command resets the configuration name of the MST region in which the switch resides. This name can include up to 32 non-blank characters and is case-sensitive. On all switches within a given MST region, the configuration names must be identical. Thus, if you want more than one MSTP switch in the same MST region, you must configure the identical region name on all such switches. If you retain the default configuration name on a switch, it cannot exist in the same MST region with another switch. (Default Name: A text string using the hexadecimal representation of the switch's MAC address)

The **no** *form of the command overwrites the currently configured name with the default name.*

> ***Note:*** *This option is available only when the switch is configured for MSTP operation. Also, there is no defined limit on the number of regions you can configure.*

Syntax: spanning-tree config-revision < *revision-number* >

This command configures the revision number you designate for the MST region in which you want the switch to reside. This setting must be the same for all switches residing in the same region. Use this setting to differentiate between region configurations in situations such as the following:

- *Changing configuration settings within a region where you want to track the configuration versions you use*
- *Creating a new region from a subset of switches in a current region and want to maintain the same region name.*
- *Using the* **pending** *option to maintain two different configuration options for the same physical region.*

Note that this setting must be the same for all MSTP switches in the same MST region. (Range: **0 - 65535***; Default:* **0***)*

> ***Note:*** *This option is available only when the switch is configured for MSTP operation.*

Syntax: spanning-tree max-hops < *hop-count* >

This command resets the number of hops allowed for BPDUs in an MST region. When an MSTP switch receives a BPDU, it decrements the hop-count setting the BPDU carries. If the hop-count reaches zero, the receiving switch drops the BPDU. Note that the switch does not change the message-age and maximum-age data carried in the BPDU as it moves through the MST region and is propagated to other regions. (Range: 1 - 40; Default: 20)

Syntax: spanning-tree maximum-age <6 - 40 seconds>

Maximum age to allow for STP packets before discarding. Default is 20 seconds.

Syntax: spanning-tree force-version < stp-compatible | rstp-operation | mstp-operation >

Sets the spanning-tree compatibility mode. When the switch is configured with MSTP mode, this command forces the switch to emulate behavior of earlier versions of spanning tree protocol or return to MSTP behavior. The command is useful in test or debug applications, and removes the need to reconfigure the switch for temporary changes in spanning-tree operation.

stp-compatible: *The switch applies 802.1D STP operation on all ports.*

rstp-operation: *The switch applies 802.1w operation on all ports except those ports where it detects a system using 802.1D Spanning Tree.*

mstp-operation: *The switch applies 802.1s MSTP operation on all ports where compatibility with 802.1D or 802.1w spanning tree protocols is not required.*

This command is available when the protocol version is set to **mstp** *(see 'protocol-version' later).*

Note that even when mstp-operation is selected, if the switch detects an 802.1D BPDU or an 802.1w BPDU on a port, it communicates with the device linked to that port using STP or RSTP BPDU packets. Also, if errors are encountered as described in the "Note on MSTP Rapid State Transitions" on page 17, setting **force-version** *to* **stp-compatible** *forces the MSTP switch to communicate out all ports using operations that are compatible with IEEE 802.1D STP.*

Syntax: spanning-tree hello-time < 1..10 >

If MSTP is running and the switch is operating as the CIST root for your network, this command specifies the time in seconds between transmissions of BPDUs for all ports on the switch configured with **Use Global** *(the default). This parameter applies in MSTP, RSTP and STP modes. During MSTP operation, you can override this global setting on a per-port basis with this command:* **spanning-tree <** ***port-list*** **> hello-time < 1..10 >** (page 22). (*Default:* **2**.)

Syntax: spanning-tree legacy-mode

Set spanning tree protocol to operate either in 802.1d legacy mode or in 802.1s native mode.

Syntax: spanning-tree legacy-path-cost

Set 802.1d (legacy) or 802.1t (not legacy) default pathcost values.

Configuring MSTP Per Port

The basic port connectivity parameters affect spanning-tree links at the global level. In most cases, ProCurve recommends that you use the default settings for these parameters and apply changes on a per-port basis only where a nondefault setting is clearly indicated by the circumstances of individual links.

Command	**Page**
spanning-tree < *port-list* >	
admin-edge-port	page 5-23
auto-edge-port	page 5-23
bpdu-filter	page 5-26
bpdu-protection	page 5-28
spanning-tree mcheck	page 5-24
hello-time < global | 1..10 >	page 5-24
spanning-tree path-cost < auto | 200000000 >	page 5-22
spanning-tree point-to-point-mac < force-true | force-false | auto>	page 5-25

Command	Page
spanning-tree priority	page 5-25
root-guard	
tcn-guard	
loop-protect	

Configuring Per Port Parameters

Syntax: [no] spanning-tree <port-list> admin-edge-port

Enable **admin-edge-port** *on ports connected to end nodes. During spanning tree establishment, ports with* **admin-edge-port** *enabled transition immediately to the forwarding state. If a bridge or switch is detected on the segment, the port automatically operates as* non-edge, *not enabled.* (Default: **No** - *disabled*)

If **admin-edge-port** *is disabled on a port and* **auto-edge-port** *has not been disabled, the* **auto-edge-port** *setting controls the behavior of the port.*

The **no spanning-tree < port-list > admin-edge-port** *command disables edge-port operation on the specified ports.*

Syntax: [no] spanning-tree < *port-list* > auto-edge-port

Supports the automatic identification of edge ports. The port will look for BPDUs for 3 seconds; if there are none it begins forwarding packets. If **admin-edge-port** *is enabled for a port, the setting for* **auto-edge-port** *is ignored whether set to yes or no. If* **admin-edge-port** *is set to* **No**, *and* **auto-edge-port** *has not been disabled (set to* **No***), then the* **auto-edge-port** *setting controls the behavior of the port. (Default:* **Yes - enabled***)*

The **no spanning-tree <** ***port-list*** **> auto-edge-port** *command disables* **auto-edge-port** *operation on the specified ports.*

Syntax: [no] spanning-tree < *port-list* > mcheck

Forces a port to send RSTP BPDUs for 3 seconds. This allows for another switch connected to the port and running RSTP to establish its connection quickly and for identifying switches running 802.1D STP. If the whole-switch force-version parameter is set to stp-compatible, the switch ignores the mcheck setting and sends 802.1D STP BPDUs out all ports. Disable this feature on all ports that are known to be connected to devices that are running 802.1D STP.

The **no spanning-tree <** ***port-list*** **> mcheck** *command disables mcheck.*

(Default: **Yes** *– mcheck is enabled)*

Syntax: spanning-tree < *port-list* > hello-time < global | 1 - 10 >

When the switch is the CIST root, this parameter specifies the interval (in seconds) between periodic BPDU transmissions by the designated ports. This interval also applies to all ports in all switches downstream from each port in the < port-list*>. A setting of* **global** *indicates that the ports in <* port-list*> on the CIST root are using the value set by the global spanning-tree* **hello-time** *value (page 5-24).*

When a given switch "X" is not the CIST root, the per-port **hello-time** *for all active ports on switch "X" is propagated from the CIST root, and is the same as the* **hello-time** *in use on the CIST root port in the currently active path from switch "X" to the CIST root. (That is, when switch "X" is not the CIST root, then the upstream CIST root's port* **hello-time** *setting overrides the* **hello-time** *setting configured on switch "X").*

(Default Per-Port setting: **Use Global**.
Default Global Hello-Time: **2**.*)*

Syntax: spanning-tree < *port-list* > path-cost < auto | 1..200000000 >

Assigns an individual port cost that the switch uses to determine which ports are forwarding ports in a given spanning tree. In the default configuration (auto) the switch determines a port's path cost by the port's type:

- *10 Mbps:* **2000000**
- *100 Mbps:* **200000**
- *1 Gbps:* **20000**

Refer to "Note on Path Cost" on page 5-5 for information on compatibility with devices running 802.1D STP for the path cost values

(Default: Auto).

Syntax: spanning-tree < *port-list* > point-to-point-mac < force-true | force-false | auto >

This parameter informs the switch of the type of device to which a specific port connects.

Force-True ***(default)****: Indicates a point-to-point link to a device such as a switch, bridge, or end-node.*

Force-False*: Indicates a connection to a hub (which is a shared LAN segment).*

Auto: *Causes the switch to set Force-False on the port if it is not running at full duplex. (Connections to hubs are half-duplex.)*

Syntax: spanning-tree < *port-list* > priority < priority-multiplier >

MSTP uses this parameter to determine the port(s) to use for forwarding. The port with the lowest priority number has the highest priority. The range is 0 to 240, and is configured by specifying a multiplier in the range of 0 - 15. That is, when you specify a priority multiplier of 0 - 15, the actual priority assigned to the switch is:

(*priority-multiplier*) *x* 16

*For example, if you configure "***2***" as the priority multiplier on a given port, then the actual* **Priority** *setting is 32. Thus, after you specify the port priority multiplier, the switch displays the actual port priority (and not the multiplier) in the* **show spanning-tree** *or* **show spanning-tree <** ***port-list*** **>** *displays.*

You can view the actual multiplier setting for ports by executing **show running** *and looking for an entry in this format:*

spanning-tree < *port-list* > priority < *priority-multiplier* >

For example, configuring port A2 with a priority multiplier of "3" results in this line in the **show running** *output:*

```
spanning-tree A2 priority 3
```

Syntax: spanning-tree < *port-list* > root-guard

MSTP only. When a port is enabled as **root-guard***, it cannot be selected as the root port even if it receives superior STP BPDUs. The port is assigned an "alternate" port role and enters a blocking state if it receives superior STP BPDUs. The BPDUs received on a port enabled as* **root-guard** *are ignored. All other BPDUs are accepted and the external devices may belong to the spanning tree as long as they do not claim to be the Root device.*

Syntax: spanning-tree < *port-list* > tcn-guard

When **tcn-guard** *is enabled for a port, it causes the port to stop propagating received topology change notifications and topology changes to other ports.*

(Default: **No** *- disabled)*

Configuring BPDU Filtering

The STP BPDU filter feature allows control of spanning-tree participation on a per-port basis. It can be used to exclude specific ports from becoming part of spanning tree operations. A port with the BPDU filter enabled will ignore incoming BPDU packets and stay locked in the spanning-tree forwarding state. All other ports will maintain their role.

Here are some sample scenarios in which this feature may be used:

- To have STP operations running on selected ports of the switch rather than every port of the switch at a time.
- To prevent the spread of errant BPDU frames.
- To eliminate the need for a topology change when a port's link status changes. For example, ports that connect to servers and workstations can be configured to remain outside of spanning-tree operations.
- To protect the network from denial of service attacks that use spoofing BPDUs by dropping incoming BPDU frames. For this scenario, BPDU protection offers a more secure alternative, implementing port shut down and a detection alert when errant BPDU frames are received (see page page 5-27 for details).

Caution

Ports configured with the BPDU filter mode remain active (learning and forward frames); however, spanning-tree cannot receive or transmit BPDUs on the port. The port remains in a forwarding state, permitting all broadcast traffic. This can create a network storm if there are any loops (that is, trunks or redundant links) using these ports. If you suddenly have a high load, disconnect the link and disable the bpdu-filter (using the **no** command).

Command Syntax and Example. The following command is used to configure BPDU filters.

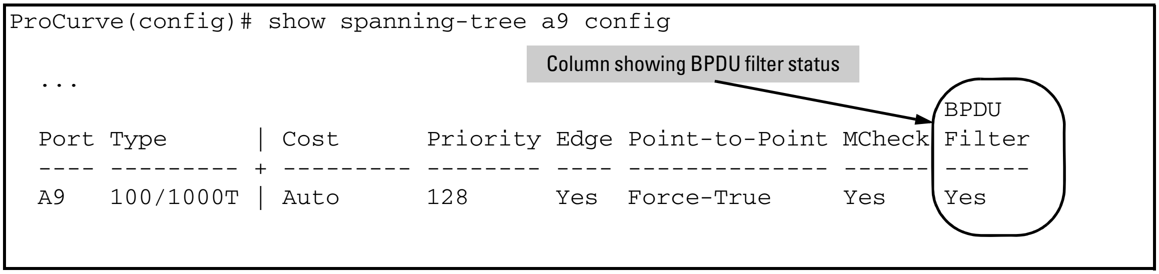
Syntax: [no] spanning-tree <port-list | all> bpdu-filter

Enables/disables the BPDU filter feature on the specified port(s). The bpdu-filter option forces a port to ***always*** *stay in the forwarding state and be excluded from standard STP operation.*

For example, to configure BPDU filtering on port a9, enter:

```
ProCurve(config)# spanning-tree a9 bpdu-filter
```

Viewing BPDU Filtering. The **spanning-tree show** < port> **configuration** command displays the BPDU's filter state.

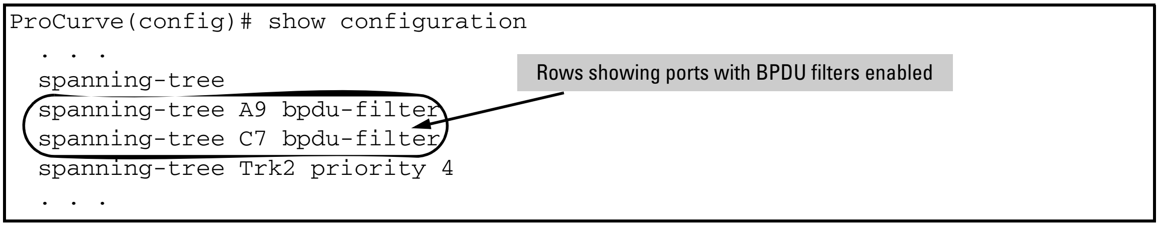


```
ProCurve(config)# show spanning-tree a9 config

  ...
                                                              BPDU
  Port Type      | Cost      Priority Edge Point-to-Point MCheck Filter
  ---- --------- + --------- -------- ---- -------------- ------ ------
  A9   100/1000T | Auto      128      Yes  Force-True     Yes    Yes
```

Column showing BPDU filter status

Figure 5-5. Example of BPDU Filter in Show Spanning Tree Configuration Command

BPDU filters per port are displayed as separate entries of the spanning tree category within the configuration file.



```
ProCurve(config)# show configuration
  . . .
  spanning-tree
  spanning-tree A9 bpdu-filter
  spanning-tree C7 bpdu-filter
  spanning-tree Trk2 priority 4
  . . .
```

Rows showing ports with BPDU filters enabled

Figure 5-6. Example of BPDU Filters in the Show Configuration Command

Configuring BPDU Protection

BPDU protection is a security feature designed to protect the active STP topology by preventing spoofed BPDU packets from entering the STP domain. In a typical implementation, BPDU protection would be applied to edge ports connected to end user devices that do not run STP. If STP BPDU packets are received on a protected port, the feature will disable that port and alert the network manager via an SNMP trap as shown in Figure 5-7.

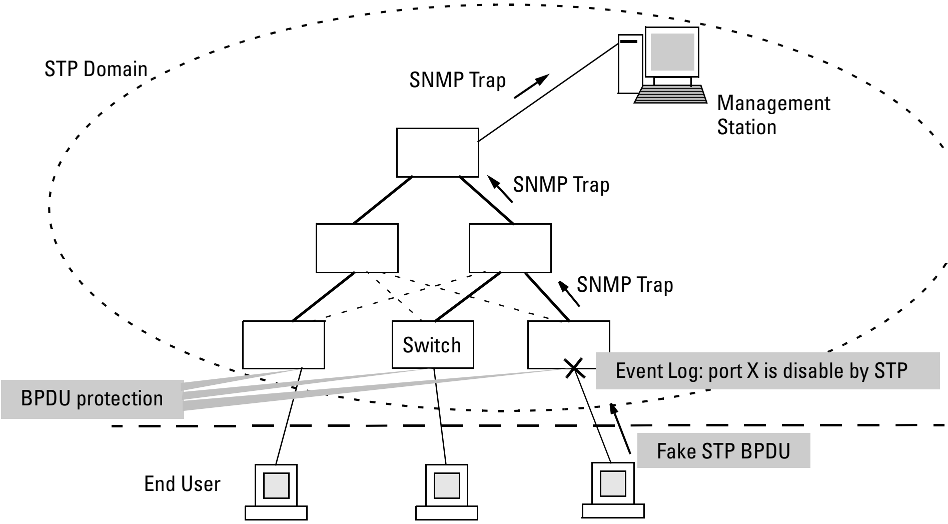


Figure 5-7. Example of BPDU Protection Enabled at the Network Edge

The following commands allow you to configure BPDU protection.

Syntax: [no] spanning-tree <*port-list*> bpdu-protection

Enables/disables the BPDU protection feature on a port

Syntax: [no] spanning-tree <*port-list*> bpdu-protection-timeout <*timeout*>

Configures the duration of time when protected ports receiving unauthorized BPDUs will remain disabled. The default value of zero sets an infinite timeout (that is, ports that are disabled by bpdu-protection are not, by default, re-enabled automatically).

(Range: 0-65535 seconds; Default: 0)

Syntax: [no] spanning-tree trap errant-bpdu

Enables/disables the sending of errant BPDU traps.

Caution

This command should only be used to guard edge ports that are not expected to participate in STP operations. Once BPDU protection is enabled, it will disable the port as soon as any BPDU packet is received on that interface.

Example. To configure BPDU protection on ports 1 to 10 with SNMP traps enabled, enter:

```
ProCurve(config)# spanning-tree 1-10 bpdu protection
ProCurve(config)# spanning-tree trap errant-bpdu
```

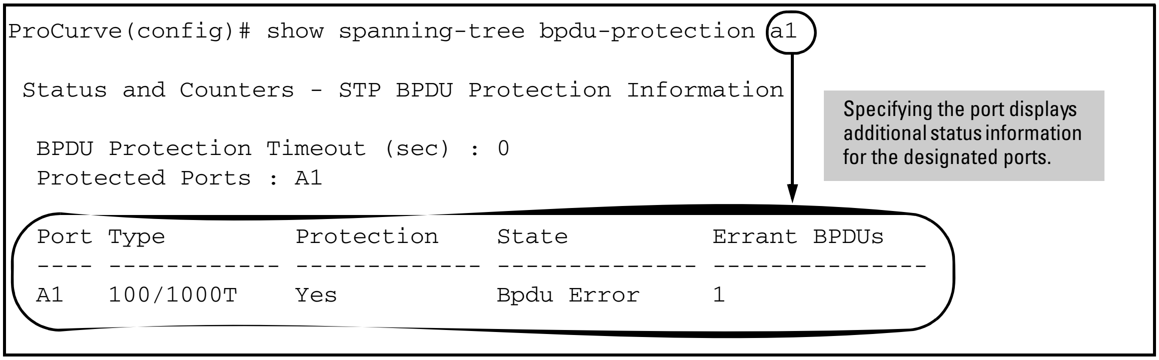
The following steps will then be set in process:

1. When an STP BPDU packet is received on ports 1-10, STP treats it as an unauthorized transmission attempt and shuts down the port that the BPDU came in on.
2. An event message is logged and an SNMP notification trap is generated.
3. The port remains disabled until re-enabled manually by a network administrator using the **interface** <*port-list*> **enable** command.

Note

To re-enable the bpdu-protected ports automatically, configure a timeout period using the **spanning-tree bpdu-protection-timeout** command.

Viewing BPDU Protection Status. The **show spanning-tree bpdu-protection** command displays a summary listing of ports with BPDU protection enabled. To display detailed per port status information, enter the specific port number(s) as shown in figure 5-4 below.

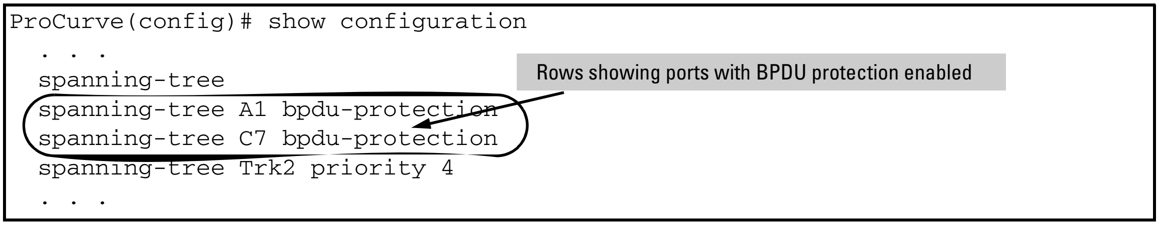


```
ProCurve(config)# show spanning-tree bpdu-protection a1

 Status and Counters - STP BPDU Protection Information

  BPDU Protection Timeout (sec) : 0
  Protected Ports : A1

  Port Type         Protection    State          Errant BPDUs
  ---- ------------ ------------- -------------- ---------------
  A1   100/1000T    Yes           Bpdu Error     1
```

Specifying the port displays additional status information for the designated ports.

Figure 5-8. Example of Show Spanning Tree BPDU Protection Command

BPDU protected ports are displayed as separate entries of the spanning tree category within the configuration file.



```
ProCurve(config)# show configuration
  . . .
  spanning-tree
  spanning-tree A1 bpdu-protection
  spanning-tree C7 bpdu-protection
  spanning-tree Trk2 priority 4
  . . .
```

Rows showing ports with BPDU protection enabled

Figure 5-9. Example of BPDU Filters in the Show Configuration Command

Configuring Loop Protection

You can use BPDU protection for systems that have spanning tree enabled (See "Configuring BPDU Protection" on page 5-27), however, the BPDU protection feature cannot detect the formation of loops when an unmanaged device on the network drops spanning tree packets. To protect against the formation of loops in these cases, you can enable the Loop Protection feature, which provides protection by transmitting loop protocol packets out ports on which loop protection has been enabled. When the switch sends out a loop protocol packet and then receives the same packet on a port that has **send-disable** configured, it shuts down the port from which the packet was sent.

You can configure the **disable-timer** parameter for the amount of time you want the port to remain disabled (0 to 604800 seconds). If you configure a value of zero, the port will not be re-enabled.

To enable loop protection, enter this command:

```
ProCurve(config)# loop-protect <port-list>
```

Syntax: [no] loop-protect <port-list> [receiver-action <send-disable | no-disable> |] [transmit-interval <1-10>] | [disable-timer <0-604800>] | [trap <loop-detected>]

Allows you to configure per-port loop protection on the switch.

[receiver-action <send-disable | no-disable>]

Sets the action to be taken when a loop is detected on the port. The port that received the loop protection packet determines what action is taken. If send-disable is configured, the port that transmitted the packet is disabled. If no-disable is configured, the port is not disabled.
Default: send-disable

[trap <loop-detected>]

Allows you to configure loop protection traps The "loop-detected" trap indicates that a loop was detected on a port.

[disable-timer <0-604800>]

How long (in seconds) a port is disabled when a loop has been detected. A value of zero disables the auto re-enable functionality.
Default: Timer is disabled

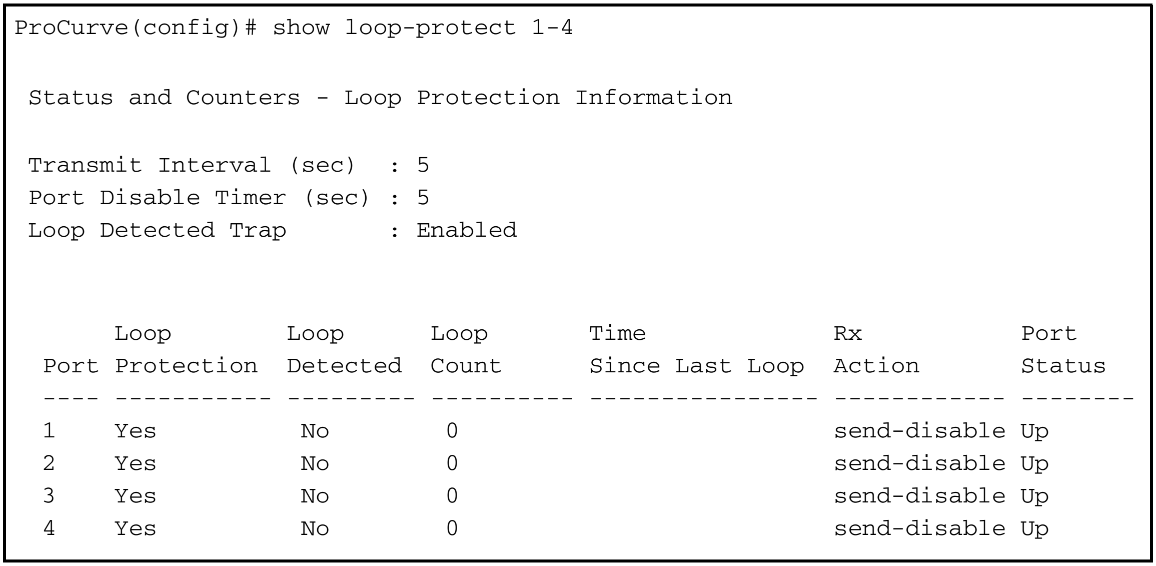
[transmit-interval <1-10>]

Allows you to configure the time in seconds between the transmission of loop protection packets.
Default: 5 seconds

To display information about ports with loop protection, enter this command.

Syntax: show loop-protect <*port-list*>

Displays the loop protection status. If no ports are specified, the information is displayed only for the ports that have loop protection enabled.



```
ProCurve(config)# show loop-protect 1-4

 Status and Counters - Loop Protection Information

 Transmit Interval (sec)   : 5
 Port Disable Timer (sec)  : 5
 Loop Detected Trap        : Enabled


       Loop         Loop      Loop       Time              Rx           Port
  Port Protection  Detected  Count      Since Last Loop   Action       Status
  ---- ----------- --------- ---------- ---------------- ------------ --------
  1    Yes          No        0                           send-disable Up
  2    Yes          No        0                           send-disable Up
  3    Yes          No        0                           send-disable Up
  4    Yes          No        0                           send-disable Up
```

Figure 5-10. Example of Show Loop Protect Display

Configuring MST Instance Parameters

Command	Page
spanning-tree instance < 1..16 > vlan < *vid*> [*vid..vid*] no spanning-tree instance < 1..16 >	page 5-32
spanning-tree instance < 1..16 > priority < 0..15 >	page 5-33
spanning-tree priority < 0..15 >	page 5-34

Syntax: spanning-tree instance < 1..16 > vlan < *vid* [*vid..vid*] >
no spanning-tree instance < 1..16 >

Configuring MSTP on the switch automatically configures the IST instance and places all statically configured VLANs on the switch into the IST instance. This command creates a new MST instance (MSTI) and moves the VLANs you specify from the IST to the MSTI. At least one VLAN must be mapped to a MSTI when you create it. (A VLAN cannot be mapped to more than one instance at a time.) You can create up to 16 MSTIs in a region. Use the no form of the command to remove a VLAN from an MSTI. (Removing a VLAN from an MSTI returns the VLAN to the IST instance, where it can either remain or be re-assigned to another MSTI configured in the region.)

The **no** *form of the command deletes the specified MSTI and returns all VLAN assignments to the region's IST instance.*

Syntax: spanning-tree instance < 1..16 > priority < 0 .. 15 >

This command sets the switch (bridge) priority for the designated instance. This priority is compared with the priorities of other switches in the same instance to determine the root switch for the instance. The lower the priority value, the higher the priority. (If there is only one switch in the instance, then that switch is the root switch for the instance.) The root bridge in a given instance provides the path to connected instances in other regions that share one or more of the same VLAN(s). (Traffic in VLANs assigned to a numbered STP instance in a given region moves to other regions through the root switch for that instance.)

The priority range for an MSTP switch is 0-61440. However, this command specifies the priority as a multiplier (0-15) of 4096. That is, when you specify a priority multiplier value of 0-15, the actual priority assigned to the switch for the specified MST instance is:

(*priority-multiplier*) x 4096

*For example, if you configure "***5***" as the priority-multiplier for MST Instance 1 on a given MSTP switch, then the* **Switch Priority** *setting is 20,480 for that instance in that switch.*

Note: *If multiple switches in the same MST instance have the same priority setting, then the switch with the lowest MAC address becomes the root switch for that instance.*

Syntax: spanning-tree priority < 0 .. 15 >

Every switch running an instance of MSTP has a Bridge Identifier, which is a unique identifier that helps distinguish this switch from all others. The switch with the lowest Bridge Identifier is elected as the root for the tree.

The Bridge Identifier is composed of a configurable Priority component (2 bytes) and the bridge's MAC address (6 bytes) The ability to change the Priority component provides flexibility in determining which switch will be the root for the tree, regardless of its MAC address.

This command sets the switch (bridge) priority for the designated region in which the switch resides. The switch compares this priority with the priorities of other switches in the same region to determine the root switch for the region. The lower the priority value, the higher the priority. (If there is only one switch in the region, then that switch is the root switch for the region.) The root bridge in a region provides the path to connected regions for the traffic in VLANs assigned to the region's IST instance. (Traffic in VLANs assigned to a numbered STP instance in a given region moves to other regions through the root switch for that instance.)

The priority range for an MSTP switch is 0-61440. However, this command specifies the priority as a multiplier (0 - 15) of 4096. That is, when you specify a priority multiplier value of 0 - 15, the actual priority assigned to the switch is:

(*priority-multiplier*) x 4096

*For example, if you configure "***2***" as the priority-multiplier on a given MSTP switch, then the* **Switch Priority** *setting is 8,192.*

Note: *If multiple switches in the same MST region have the same priority setting, then the switch with the lowest MAC address becomes the root switch for that region.*

Configuring MST Instance Per-Port Parameters

Command	Page
spanning-tree instance < 1..16 > < port-list > path-cost < auto | 1..200000000 >	page 5-35
spanning-tree instance < 1..16 > < *port-list* > priority < *priority-multiplier* >	page 5-36
spanning-tree < *port-list* > priority < *priority-multiplier* >	page 5-37

Syntax: spanning-tree instance < 1..16 > [e] < *port-list* > path-cost < auto | 1..200000000 >

This command assigns an individual port cost for the specified MST instance. (For a given port, the path cost setting can be different for different MST instances to which the port may belong.) The switch uses the path cost to determine which ports are the forwarding ports in the instance; that is which links to use for the active topology of the instance and which ports to block. The settings are either **auto** *or in a range from 1 to 200,000,000. With the* **auto** *setting, the switch calculates the path cost from the link speed:*

10 Mbps — 2000000

100 Mbps — 200000

1 Gbps — 20000

(Default: **Auto***)*

Syntax: spanning-tree instance < 1..16 > [e] < *port-list*> priority <*priority-multiplier*>

This command sets the priority for the specified port(s) in the specified MST instance. (For a given port, the priority setting can be different for different MST instances to which the port may belong.) The priority range for a port in a given MST instance is 0-255. However, this command specifies the priority as a multiplier (0 - 15) of 16. That is, when you specify a priority multiplier of 0 - 15, the actual priority assigned to the switch is:

(*priority-multiplier*) *x* 16

*For example, if you configure "***2***" as the priority multiplier on a given port in an MST instance, then the actual* **Priority** *setting is 32. Thus, after you specify the port priority multiplier in an instance, the switch displays the actual port priority (and not the multiplier) in the* **show spanning-tree instance < 1..16 >** *or* **show spanning-tree <** ***port-list*** **> instance < 1..16 >** *displays.*

You can view the actual multiplier setting for ports in the specified instance by executing **show running** *and looking for an entry in this format:*

spanning-tree instance < 1..15 > < *port-list* > priority < *priority-multiplier* >

For example, configuring port A2 with a priority multiplier of "3" in instance 1, results in this line in the **show running** *output:*

```
spanning-tree instance 1 A2 priority 3
```

Syntax: spanning-tree [e] < *port-list* > priority < *priority-multiplier* >

This command sets the priority for the specified port(s) for the IST (that is, Instance 0) of the region in which the switch resides. The "priority" component of the port's "Port Identifier" is set. The Port Identifier is a unique identifier that helps distinguish this switch's ports from all others. It consists of the Priority value with the port number extension—PRIORITY:PORT_NUMBER. A port with a lower value of Port Identifier is more likely to be included in the active topology.

This priority is compared with the priorities of other ports in the IST to determine which port is the root port for the IST instance. The lower the priority value, the higher the priority. The IST root port (or trunk) in a region provides the path to connected regions for the traffic in VLANs assigned to the region's IST instance.

The priority range for a port in a given MST instance is 0-240. However, this command specifies the priority as a multiplier (0 - 15) of 16. That is, when you specify a priority multiplier of 0 - 15, the actual priority assigned to the switch is:

(*priority-multiplier*) x 16

*For example, configuring "***5***" as the priority multiplier on a given port in the IST instance for a region creates an actual* **Priority** *setting of* **80***. Thus, after you specify the port priority multiplier for the IST instance, the switch displays the actual port priority (and not the multiplier) in the* **show spanning-tree instance ist** *or* **show spanning-tree <** ***port-list*** **> instance ist** *displays. You can view the actual multiplier setting for ports in the IST instance by executing* **show running** *and looking for an entry in this format:*

spanning-tree < *port-list* > priority < *priority-multiplier* >

For example, configuring port A2 with a priority multiplier of "2" in the IST instance, results in this line in the **show running** *output:*

```
spanning-tree A2 priority 2
```

Enabling or Disabling Spanning Tree Operation

This command enables or disables spanning tree operation for any spanning tree protocol enabled on the switch. Before using this command to enable spanning tree, ensure that the version you want to use is active on the switch.

Syntax: [no] spanning-tree

Enabling spanning tree with MSTP configured implements MSTP for all physical ports on the switch, according to the VLAN groupings for the IST instance and any other configured instances. Disabling MSTP removes protection against redundant loops that can significantly slow or halt a network.

This command simply turns spanning tree on or off. It does not change the existing spanning tree configuration.

Enabling an Entire MST Region at Once or Exchanging One Region Configuration for Another

Command	Page
spanning-tree pending < apply | config-name | config-revision | instance | reset >	page 5-39

This operation exchanges the currently active MSTP configuration with the currently pending MSTP configuration. It enables you to implement a new MSTP configuration with minimal network disruption or to exchange MSTP configurations for testing or troubleshooting purposes.

When you configure or reconfigure MSTP, the switch re-calculates the corresponding network paths. This can have a ripple effect throughout your network as adjacent MSTP switches recalculate network paths to support the configuration changes invoked in a single switch. Although MSTP employs RSTP operation, the convergence time for implementing MSTP changes can be disruptive to your network. However, by using the spanning-tree **pending** feature, you can set up an MSTP on the switch and then invoke all instances of the new configuration at the same time, instead of one at a time.

To Create a Pending MSTP Configuration. This procedure creates a pending MSTP configuration and exchanges it with the active MSTP configuration.

1. Configure the VLANs you want included in any instances in the new region. When you create the pending region, all VLANs configured on the switch will be assigned to the pending IST instance unless assigned to other, pending MST instances.
2. Configure MSTP as the spanning-tree protocol, then execute **write mem** and reboot. (The pending option is available only with MSTP enabled.)
3. Configure the pending region name to assign to the switch.
4. Configure the pending **config-revision** number for the region name.
5. If you want an MST instance other than the IST instance, configure the instance number and assign the appropriate VLANs (VIDs). (The **pending** command creates the region's IST instance automatically.)
6. Repeat step 5 for each additional MST instance you want to configure.
7. Use the **show spanning-tree pending** command to review your pending configuration (page 46).
8. Use the **spanning-tree pending apply** command to exchange the currently active MSTP configuration with the pending MSTP configuration.

Syntax: spanning-tree pending < apply | *config-name* | *config-revision* | instance | reset >

apply

Exchanges the currently active MSTP configuration with the pending MSTP configuration.

config-name

Specifies the pending MST region name. Must be the same for all MSTP switches in the region. (Default: The switch's MAC address.)

config-revision

Specifies the pending MST region configuration revision number. Must be the same for all MSTP switches in the region. (Default: **0***).*

instance < 1..16 > vlan [< *vid* | *vid-range* >

Creates the pending instance and assigns one or more VLANs to the instance.

reset

Copies the switch's currently active MSTP configuration to the pending configuration. This is useful when you want to experiment with the current MSTP configuration while maintaining an unchanged version.

9. To view the current pending MSTP configuration, use the **show spanning-tree pending** command (page page 5-46).

Displaying MSTP Statistics and Configuration

Command	Page
MSTP Statistics:	
show spanning-tree [< *port-list* >]	below
show spanning-tree instance < ist I 1..16 >	42
MSTP Configuration	
show spanning-tree [*port-list*] config	43
show spanning-tree [port-list] config instance < ist I 1..16 >	44
show spanning-tree mst-config	45
show spanning-tree pending< < instance I ist > I mst-config >	page 5-46
show spanning-tree root-history <cst I ist I msti>	page 5-47

Displaying MSTP Statistics

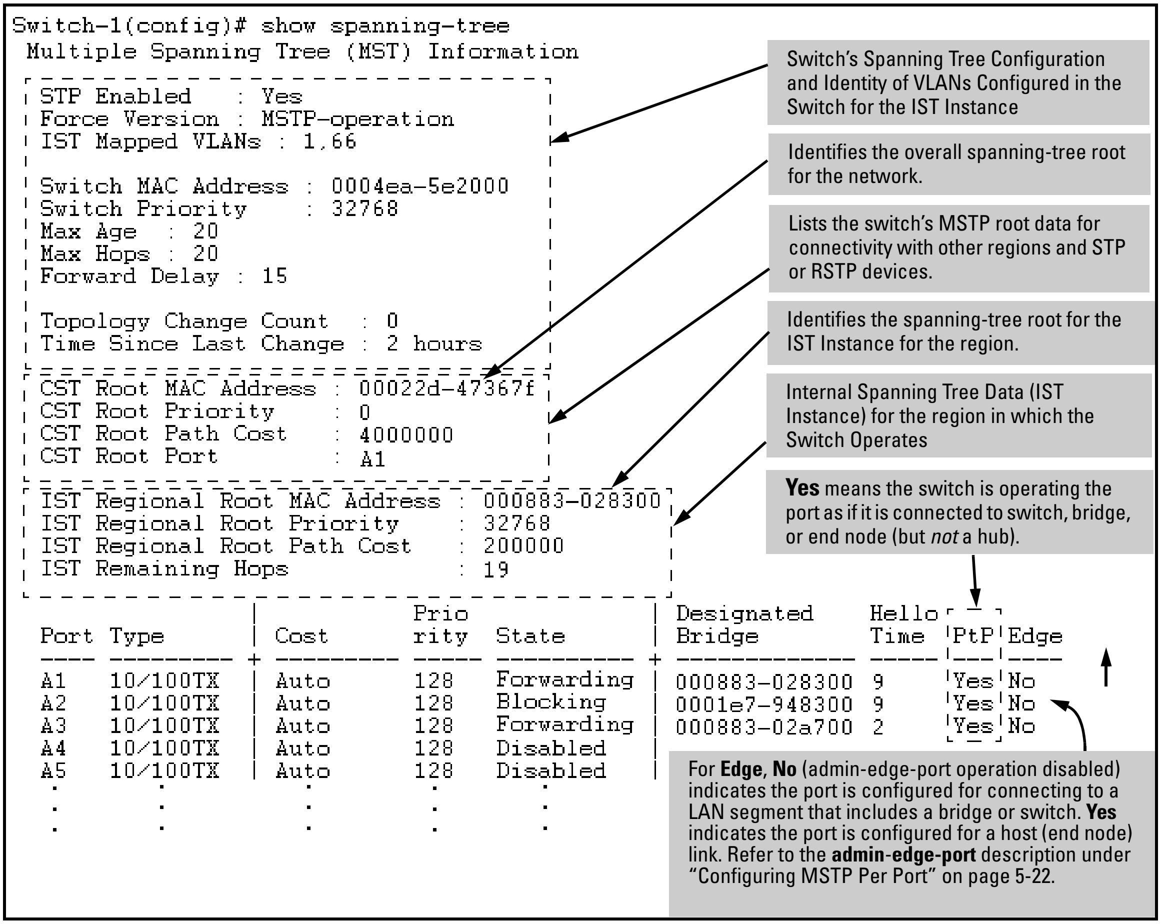
Displaying Switch Statistics for the Common Spanning Tree. This command displays the MSTP statistics for the connections between MST regions in a network.

Syntax: show spanning-tree

This command displays the switch's global and regional spanning-tree status, plus the per-port spanning-tree operation at the regional level. Note that values for the following parameters appear only for ports connected to active devices: **Designated Bridge**, **Hello Time**, **PtP**, *and* **Edge**.

Syntax: show spanning-tree < *port-list* >

This command displays the spanning-tree status for the designated port(s). You can list data for a series of ports and port trunks by specifying the first and last port or trunk of any consecutive series of ports and trunks. For example, to display data for port A20-A24 and trk1, you would use this command: **show spanning-tree a20-trk1**



```
Switch-1(config)# show spanning-tree
 Multiple Spanning Tree (MST) Information

  STP Enabled    : Yes
  Force Version : MSTP-operation
  IST Mapped VLANs : 1,66

  Switch MAC Address : 0004ea-5e2000
  Switch Priority    : 32768
  Max Age  : 20
  Max Hops : 20
  Forward Delay : 15

  Topology Change Count  : 0
  Time Since Last Change : 2 hours

  CST Root MAC Address : 00022d-47367f
  CST Root Priority    : 0
  CST Root Path Cost   : 4000000
  CST Root Port        : A1

  IST Regional Root MAC Address : 000883-028300
  IST Regional Root Priority    : 32768
  IST Regional Root Path Cost   : 200000
  IST Remaining Hops            : 19

                   |            Prio              | Designated    Hello
  Port Type        | Cost       rity  State       | Bridge        Time  PtP Edge
  ---- ---------- + ---------- ----- ---------- + ------------- ----- --- ----
  A1   10/100TX   | Auto       128   Forwarding | 000883-028300 9     Yes No
  A2   10/100TX   | Auto       128   Blocking   | 0001e7-948300 9     Yes No
  A3   10/100TX   | Auto       128   Forwarding | 000883-02a700 2     Yes No
  A4   10/100TX   | Auto       128   Disabled   |
  A5   10/100TX   | Auto       128   Disabled   |
   .       .          .          .        .
   .       .          .          .        .
   .       .          .          .        .
```

Switch's Spanning Tree Configuration and Identity of VLANs Configured in the Switch for the IST Instance

Identifies the overall spanning-tree root for the network.

Lists the switch's MSTP root data for connectivity with other regions and STP or RSTP devices.

Identifies the spanning-tree root for the IST Instance for the region.

Internal Spanning Tree Data (IST Instance) for the region in which the Switch Operates

Yes means the switch is operating the port as if it is connected to switch, bridge, or end node (but *not* a hub).

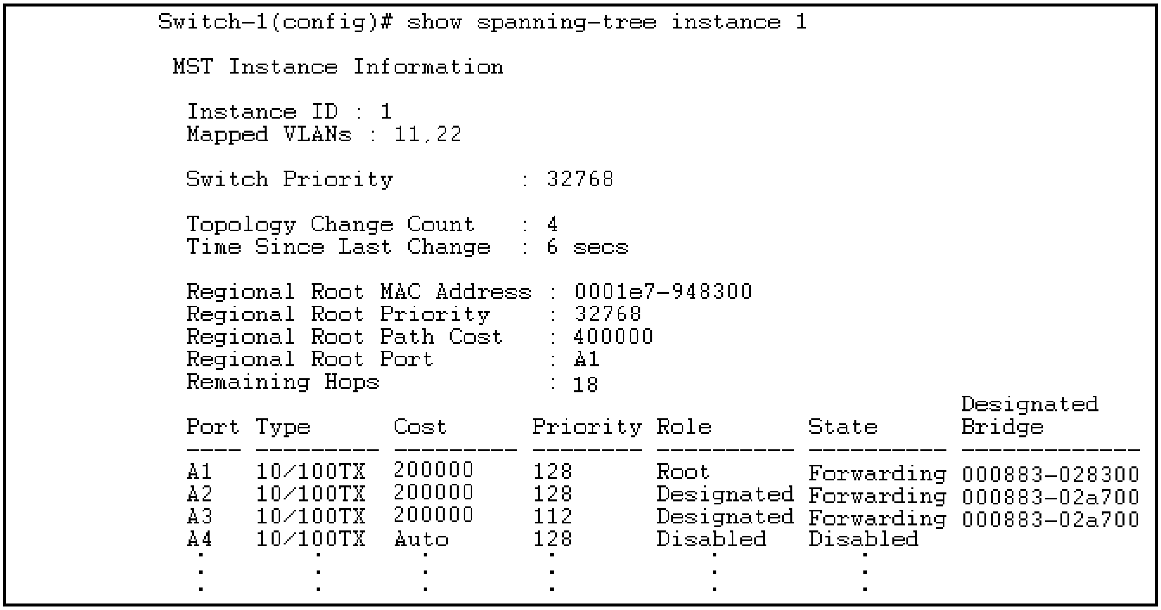
For **Edge**, **No** (admin-edge-port operation disabled) indicates the port is configured for connecting to a LAN segment that includes a bridge or switch. **Yes** indicates the port is configured for a host (end node) link. Refer to the **admin-edge-port** description under "Configuring MSTP Per Port" on page 5-22.

Figure 5-11. Example of Common Spanning Tree Status on an MSTP Switch

Displaying Switch Statistics for a Specific MST Instance.

Syntax: show spanning-tree instance < ist | 1..16 >

This command displays the MSTP statistics for either the IST instance or a numbered MST instance running on the switch.



```
Switch-1(config)# show spanning-tree instance 1

 MST Instance Information

  Instance ID : 1
  Mapped VLANs : 11,22

  Switch Priority          : 32768

  Topology Change Count    : 4
  Time Since Last Change   : 6 secs

  Regional Root MAC Address : 0001e7-948300
  Regional Root Priority    : 32768
  Regional Root Path Cost   : 400000
  Regional Root Port        : A1
  Remaining Hops            : 18
                                                                Designated
  Port Type      Cost      Priority Role       State      Bridge
  ---- --------- --------- -------- ---------- ---------- -------------
  A1   10/100TX  200000    128      Root       Forwarding 000883-028300
  A2   10/100TX  200000    128      Designated Forwarding 000883-02a700
  A3   10/100TX  200000    112      Designated Forwarding 000883-02a700
  A4   10/100TX  Auto      128      Disabled   Disabled
   .        .       .        .          .          .
   .        .       .        .          .          .
   .        .       .        .          .          .
```

Figure 5-12. Example of MSTP Statistics for a Specific Instance on an MSTP Switch

Displaying the MSTP Configuration

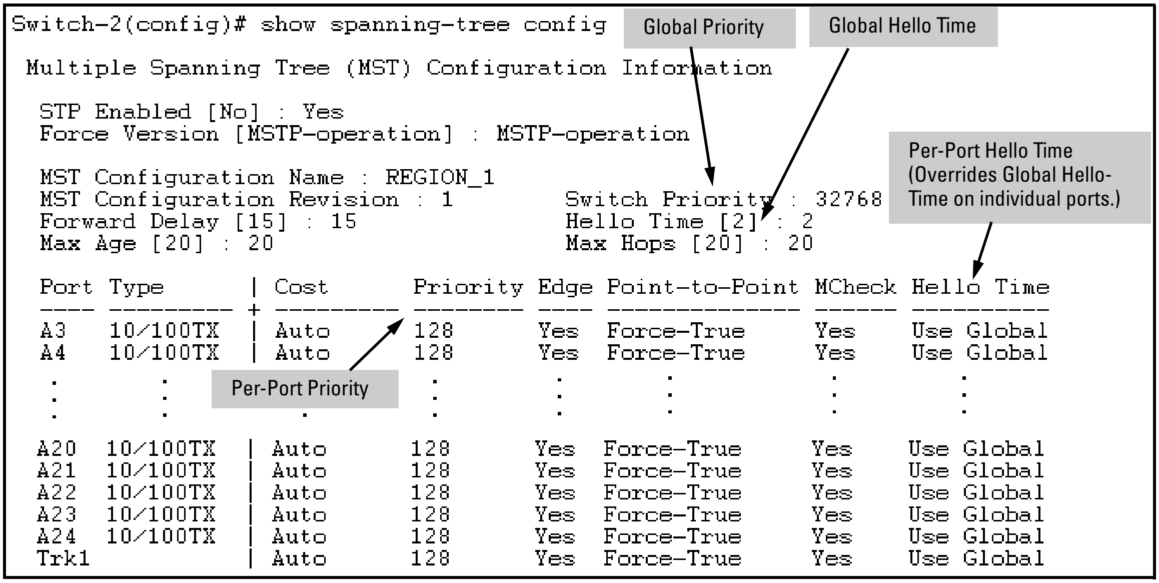
Displaying the Global MSTP Configuration. This command displays the switch's basic and MST region spanning-tree configuration, including basic port connectivity settings.

Syntax: show spanning-tree config

The upper part of this output shows the switch's global spanning-tree configuration that applies to the MST region. The port listing shows the spanning-tree port parameter settings for the spanning-tree region operation (configured by the **spanning-tree <** ***port-list*** **>** *command). For information on these parameters, refer to "Configuring MSTP Per Port" on page 5-22.*

Syntax: show spanning-tree < *port-list* > config

This command shows the same data as the above command, but lists the spanning-tree port parameter settings for only the specified port(s) and/or trunk(s). You can list data for a series of ports and port trunks by specifying the first and last port or trunk of any consecutive series of ports and trunks. For example, to display data for port A20-A24 and trk1, use this command: **show spanning-tree a20-trk1 config**



```
Switch-2(config)# show spanning-tree config

 Multiple Spanning Tree (MST) Configuration Information

  STP Enabled [No] : Yes
  Force Version [MSTP-operation] : MSTP-operation

  MST Configuration Name : REGION_1
  MST Configuration Revision : 1       Switch Priority : 32768
  Forward Delay [15] : 15              Hello Time [2] : 2
  Max Age [20] : 20                    Max Hops [20] : 20

  Port Type      | Cost      Priority Edge Point-to-Point MCheck Hello Time
  ---- --------- + --------- -------- ---- -------------- ------ ----------
  A3   10/100TX  | Auto      128      Yes  Force-True     Yes    Use Global
  A4   10/100TX  | Auto      128      Yes  Force-True     Yes    Use Global
  .        .         .        .        .        .          .        .
  .        .         .        .        .        .          .        .

  A20  10/100TX  | Auto      128      Yes  Force-True     Yes    Use Global
  A21  10/100TX  | Auto      128      Yes  Force-True     Yes    Use Global
  A22  10/100TX  | Auto      128      Yes  Force-True     Yes    Use Global
  A23  10/100TX  | Auto      128      Yes  Force-True     Yes    Use Global
  A24  10/100TX  | Auto      128      Yes  Force-True     Yes    Use Global
  Trk1           | Auto      128      Yes  Force-True     Yes    Use Global
```

Global Priority
Global Hello Time
Per-Port Hello Time (Overrides Global Hello-Time on individual ports.)
Per-Port Priority

Figure 5-13. Example of Displaying the Switch's Global Spanning-Tree Configuration

Displaying Per-Instance MSTP Configurations. These commands displays the per-instance port configuration and current state, along with instance identifiers and regional root data.

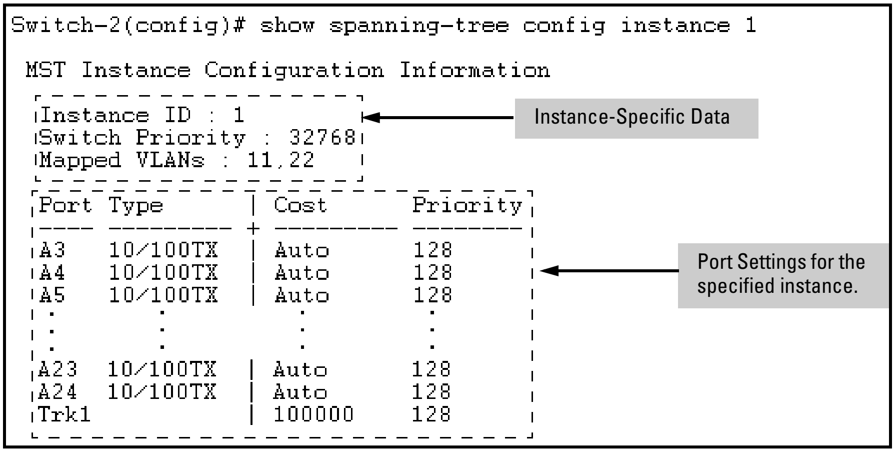
Syntax: show spanning-tree config instance < ist | 1..16 >

The upper part of this output shows the instance data for the specified instance. The lower part of the output lists the spanning-tree port settings for the specified instance.

Syntax: show spanning-tree < *port-list* > config instance < ist | 1..16 >

This command shows the same data as the above command, but lists the spanning-tree port parameter settings for only the specified port(s) and/or trunk(s). You can list data for a series of ports and port trunks by specifying the first and last port or trunk of any consecutive series of ports and trunks. For example, to display data for port A20-A24 and trk1, use this command:

show spanning-tree a20-trk1 config instance 1



```
Switch-2(config)# show spanning-tree config instance 1

 MST Instance Configuration Information

  Instance ID : 1
  Switch Priority : 32768
  Mapped VLANs : 11,22

  Port Type      | Cost      Priority
  ---- --------- + --------- --------
  A3   10/100TX  | Auto      128
  A4   10/100TX  | Auto      128
  A5   10/100TX  | Auto      128
  .        .          .        .
  .        .          .        .
  .        .          .        .
  A23  10/100TX  | Auto      128
  A24  10/100TX  | Auto      128
  Trk1           | 100000    128
```

Instance-Specific Data

Port Settings for the specified instance.

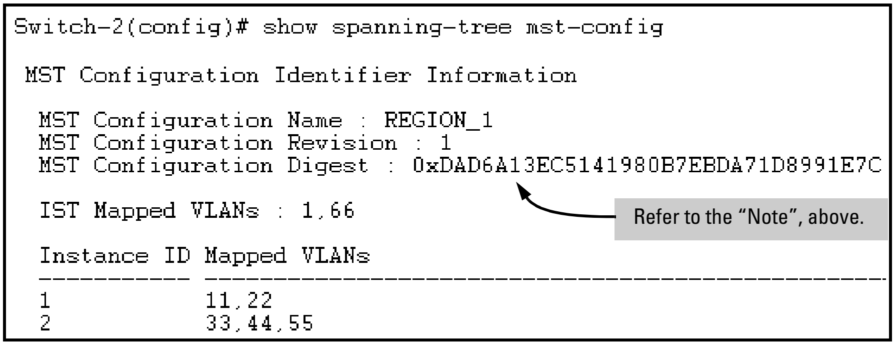
Figure 5-14. Example of the Configuration Listing for a Specific Instance

Displaying the Region-Level Configuration in Brief. This command output is useful for quickly verifying the allocation of VLANs in the switch's MSTP configuration and for viewing the configured region identifiers.

Syntax: show spanning-tree mst-config

This command displays the switch's regional configuration.

***Note:** The switch computes the **MSTP Configuration Digest** from the VID to MSTI configuration mappings on the switch itself. As required by the 802.1s standard, all MSTP switches within the same region must have the same VID to MSTI assignments, and any given VID can be assigned to either the IST or one of the MSTIs within the region. Thus, the MSTP Configuration Digest must be identical for all MSTP switches intended to belong to the same region. When comparing two MSTP switches, if their Digest identifiers do not match, then they cannot be members of the same region.*



```
Switch-2(config)# show spanning-tree mst-config

 MST Configuration Identifier Information

  MST Configuration Name : REGION_1
  MST Configuration Revision : 1
  MST Configuration Digest : 0xDAD6A13EC5141980B7EBDA71D8991E7C

  IST Mapped VLANs : 1,66

  Instance ID Mapped VLANs
  ----------- ------------------------------------------------------
  1           11,22
  2           33,44,55
```

Refer to the "Note", above.

Figure 5-15. Example of a Region-Level Configuration Display

Displaying the Pending MSTP Configuration. This command displays the MSTP configuration the switch will implement if you execute the spanning-tree pending apply command (Refer to "Enabling an Entire MST Region at Once or Exchanging One Region Configuration for Another" on page 5-38.)

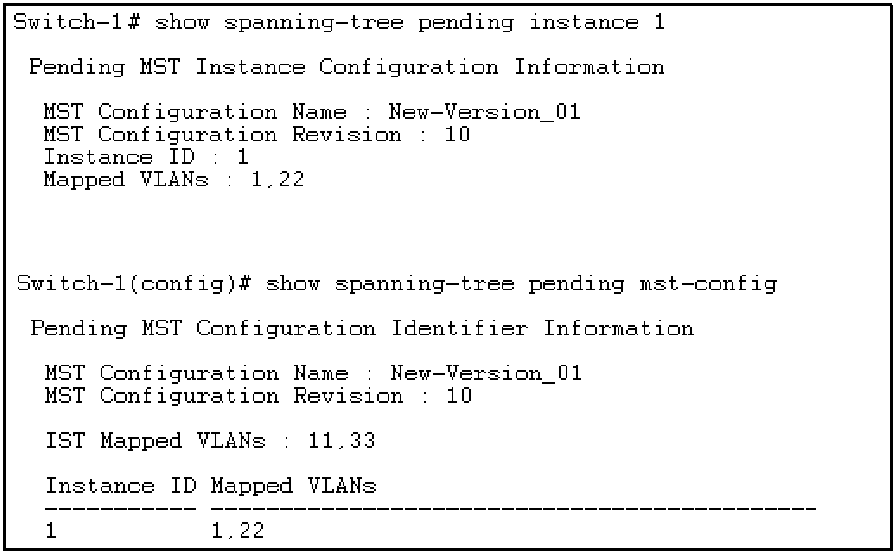
Syntax: show spanning-tree pending < instance | mst-config >

instance < 1..16 | ist >

Lists region, instance I.D. and VLAN information for the specified, pending instance.

mst-config

Lists region, IST instance VLAN(s), numbered instances, and assigned VLAN information for the pending MSTP configuration.



```
Switch-1# show spanning-tree pending instance 1

 Pending MST Instance Configuration Information

  MST Configuration Name : New-Version_01
  MST Configuration Revision : 10
  Instance ID : 1
  Mapped VLANs : 1,22



Switch-1(config)# show spanning-tree pending mst-config

 Pending MST Configuration Identifier Information

  MST Configuration Name : New-Version_01
  MST Configuration Revision : 10

  IST Mapped VLANs : 11,33

  Instance ID Mapped VLANs
  ----------- ---------------------------------------------
  1           1,22
```

Figure 5-16. Example of Displaying a Pending Configuration

Displaying the Root History. This command displays the spanning-tree root changes history information.

Syntax: show spanning-tree root-history < cst | ist | msti>

cst

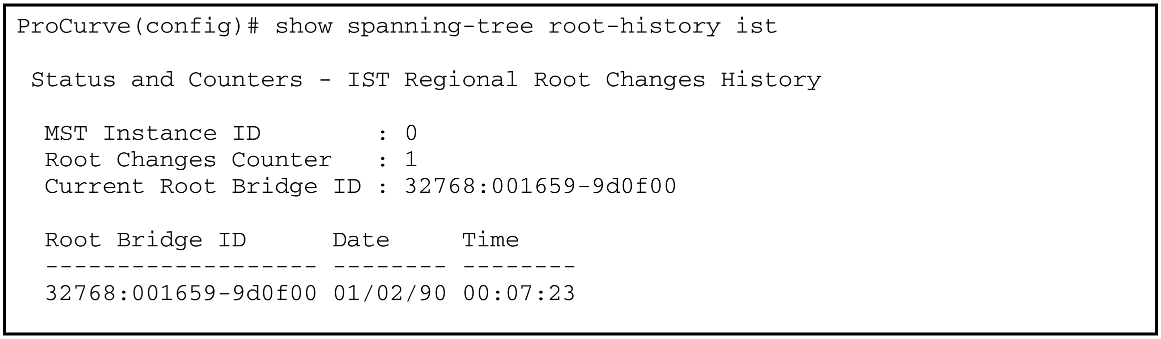
Displays the CST root changes history.

ist

Displays the IST root changes history.

msti

Displays the MSTI root changes history.



```
ProCurve(config)# show spanning-tree root-history ist

 Status and Counters - IST Regional Root Changes History

  MST Instance ID        : 0
  Root Changes Counter   : 1
  Current Root Bridge ID : 32768:001659-9d0f00

  Root Bridge ID      Date     Time
  ------------------- -------- --------
  32768:001659-9d0f00 01/02/90 00:07:23
```

Figure 5-17. Example of Displaying the IST Root Changes History

Operating Notes

SNMP MIB Support for MSTP. MSTP is a superset of the STP/802.1D and RSTP/802.1w protocols and uses the MIB objects defined for these two protocols. Also, as of December, 2003, there has been no formal MIB definition published for 802.1s MSTP managed objects.

Troubleshooting

Duplicate packets on a VLAN, or packets not arriving on a LAN at all. The allocation of VLANs to MSTIs may not be identical among all switches in a region.

A Switch Intended To Operate Within a Region Does Not Receive Traffic from Other Switches in the Region. An MSTP switch intended for a particular region may not have the same configuration name or region revision number as the other switches intended for the same region. The MSTP Configuration Name and MSTP Configuration Revision number must be identical on all MSTP switches intended for the same region. Another possibility is that the set of VLANs configured on the switch may not match the set of VLANs configured on other switches in the intended region.

6

Quality of Service (QoS): Managing Bandwidth More Effectively

Contents

Introduction

.

QoS Feature	Default	Menu	CLI	Web
UDP/TCP Priority	Disabled	—	page 6-20	Refer to the Online Help.
IP-Device Priority	Disabled	—	page 6-27	"
IP Type-of-Service Priority	Disabled	—	page 6-33	"
VLAN-ID Priority	Disabled	—	page 6-46	"
Source-Port Priority	Disabled	—	page 6-52	"
DSCP Policy Table	Various	—	page 6-58	"

As the term suggests, *network policy* refers to the network-wide controls you can implement to:

- Ensure uniform and efficient traffic handling throughout your network, while keeping the most important traffic moving at an acceptable speed, regardless of current bandwidth usage.
- Exercise control over the priority settings of inbound traffic arriving in and travelling through your network.

Adding bandwidth is often a good idea, but it is not always feasible and does not completely eliminate the potential for network congestion. There will always be points in the network where multiple traffic streams merge or where network links will change speed and capacity. The impact and number of these congestion points will increase over time as more applications and devices are added to the network.

When (not *if*) network congestion occurs, it is important to move traffic on the basis of relative importance. However, without *Quality of Service* (QoS) prioritization, less important traffic can consume network bandwidth and slow down or halt the delivery of more important traffic. That is, without QoS, most traffic received by the switch is forwarded with the same priority it had upon entering the switch. In many cases, such traffic is "normal" priority and competes for bandwidth with all other normal-priority traffic, regardless of its relative importance to your organization's mission. This section gives an overview of QoS operation and benefits, and describes how to configure QoS in the console interface.

Quality of Service is a general term for classifying and prioritizing traffic throughout a network. That is, QoS enables you to establish an end-to-end traffic priority policy to improve control and throughput of important data. You can manage available bandwidth so that the most important traffic goes first. For example, you can use Quality of Service to:

- Upgrade or downgrade traffic from various servers.
- Control the priority of traffic from dedicated VLANs or applications.
- Change the priorities of traffic from various segments of your network as your business needs change.
- Set priority policies in edge switches in your network to enable traffic-handling rules across the network.

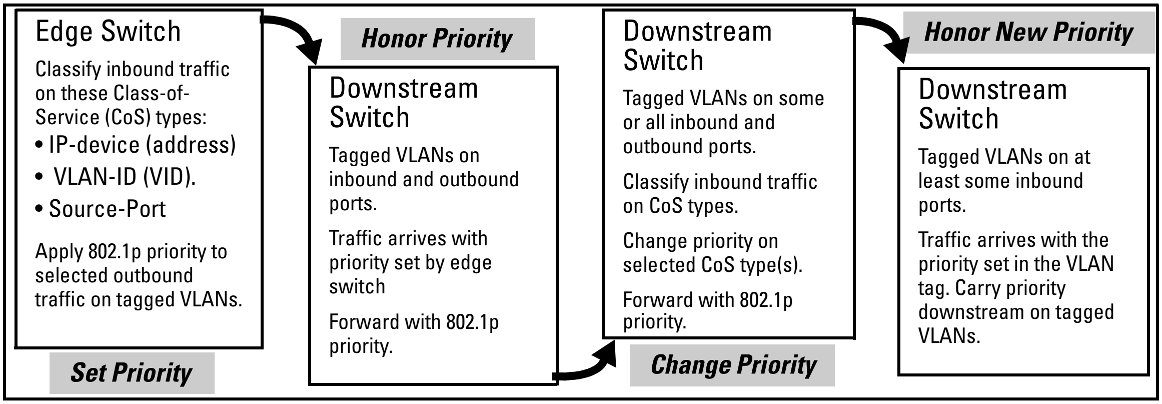

Edge Switch
Classify inbound traffic on these Class-of-Service (CoS) types:
• IP-device (address)
• VLAN-ID (VID).
• Source-Port
Apply 802.1p priority to selected outbound traffic on tagged VLANs.
Set Priority
Honor Priority
Downstream Switch
Tagged VLANs on inbound and outbound ports.
Traffic arrives with priority set by edge switch
Forward with 802.1p priority.
Downstream Switch
Tagged VLANs on some or all inbound and outbound ports.
Classify inbound traffic on CoS types.
Change priority on selected CoS type(s).
Forward with 802.1p priority.
Change Priority
Honor New Priority
Downstream Switch
Tagged VLANs on at least some inbound ports.
Traffic arrives with the priority set in the VLAN tag. Carry priority downstream on tagged VLANs.

Figure 6-1. Example of 802.1p Priority Based on CoS (Class-of-Service) Types and Use of VLAN Tags

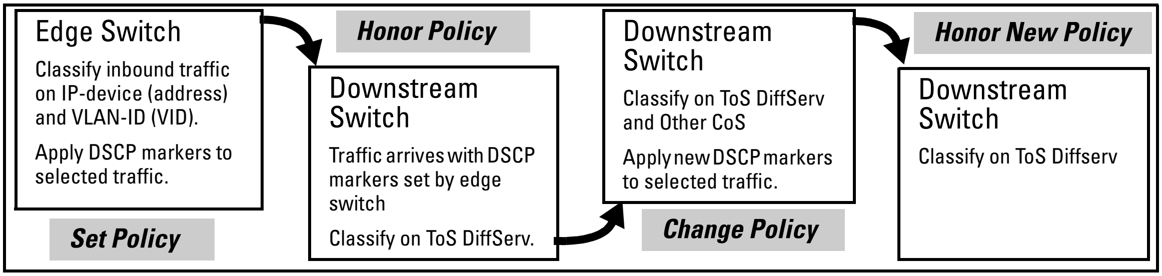

Edge Switch
Classify inbound traffic on IP-device (address) and VLAN-ID (VID).
Apply DSCP markers to selected traffic.
Set Policy
Honor Policy
Downstream Switch
Traffic arrives with DSCP markers set by edge switch
Classify on ToS DiffServ.
Downstream Switch
Classify on ToS DiffServ and Other CoS
Apply new DSCP markers to selected traffic.
Change Policy
Honor New Policy
Downstream Switch
Classify on ToS Diffserv

Figure 6-2. Example Application of Differentiated Services Codepoint (DSCP) Policies

At the edge switch, QoS classifies certain traffic types and in some cases applies a DSCP policy. At the next hop (downstream switch) QoS honors the policies established at the edge switch. Further downstream, another switch may reclassify some traffic by applying new policies, and yet other downstream switches can be configured to honor the new policies.

QoS is implemented in the form of rules or policies that are configured on the switch. While you can use QoS to prioritize only the outbound traffic while it is moving through the switch, you derive the maximum benefit by using QoS in an 802.1Q VLAN environment (with 802.1p priority tags) or in an untagged VLAN environment (with DSCP policies) where QoS can set priorities that downstream devices can support without re-classifying the traffic.

By prioritizing traffic, QoS supports traffic growth on the network while optimizing the use of existing resources—and delaying the need for further investments in equipment and services. That is, QoS enables you to:

- Specify which traffic has higher or lower priority, regardless of current network bandwidth or the relative priority setting of the traffic when it is received on the switch.
- Change (upgrade or downgrade) the priority of outbound traffic.
- Override "illegal" packet priorities set by upstream devices or applications that use 802.1Q VLAN tagging with 802.1p priority tags.
- Avoid or delay the need to add higher-cost NICs (network interface cards) to implement prioritizing. (Instead, control priority through network policy.)

QoS on the switches covered by this guide supports these types of traffic marking:

- **802.1p prioritization:** Controls the outbound port queue priority for traffic leaving the switch, and (if traffic exits through a VLAN-tagged port) sends the priority setting with the individual packets to the downstream devices.
- **IP Type-of-Service (ToS):** Enables the switch to set, change, and honor prioritization policies by using the Differentiated Services (diffserv) bits in the ToS byte of IPv4 packet headers.

Terminology

Term	Use in This Document
802.1p priority	A traffic priority setting carried by a VLAN-tagged packet moving from one device to another through ports that are tagged members of the VLAN to which the packet belongs. This setting can be from 0 - 7. The switch handles an outbound packet on the basis of its 802.1p priority. However, if the packet leaves the switch through a VLAN on which the port is an untagged member, this priority is dropped, and the packet arrives at the next, downstream device without an 802.1p priority assignment.
802.1Q field	A four-byte field that is present in the header of Ethernet packets entering or leaving the switch through a port that is a tagged member of a VLAN. This field includes an 802.1p priority setting, a VLAN tag, or ID number (VID), and other data. A packet entering or leaving the switch through a port that is an untagged member of the outbound VLAN does not have this field in its header and thus does not carry a VID or an 802.1p priority. See also "802.1p priority".
codepoint	Refer to DSCP, below.
downstream device	A device linked directly or indirectly to an outbound switch port. That is, the switch sends traffic to downstream devices.
DSCP	***Differentiated Services Codepoint.*** (Also termed ***codepoint***.) A DSCP is comprised of the upper six bits of the ToS (Type-of-Service) byte in IP packets. There are 64 possible codepoints. In the default QoS configuration for the switches covered in this chapter, one codepoint (101110) is set for Expedited Forwarding. All other codepoints are unused (and listed with **No-override** for a priority).
DSCP policy	A DSCP configured with a specific 802.1p priority (0- 7). (Default: **No-override**). Using a DSCP policy, you can configure the switch to assign priority to IP packets. That is, for an IP packet identified by the specified QoS type, you can assign a new DSCP and an 802.1p priority (0-7). For more on DSCP, refer to "Details of QoS IP Type-of-Service" on page 6-43. For the DSCP map, see figure 6-18 on page 6-44.
edge switch	In the QoS context, this is a switch that receives traffic from the edge of the LAN or from outside the LAN and forwards it to devices within the LAN. Typically, an edge switch is used with QoS to recognize packets based on QoS types such as TCP/UDP application type, IP-device (address), VLAN-ID (VID), and Source-Port (although it can also be used to recognize packets on the basis of ToS bits). Using this packet recognition, the edge switch can be used to set 802.1p priorities or DSCP policies that downstream devices will honor.
inbound port	Any port on the switch through which traffic enters the switch.
IP Options	In an IPv4 packet these are optional, extra fields in the packet header.
IP-precedence bits	The upper three bits in the Type of Service (ToS) field of an IP packet.
IPv4	Version 4 of the IP protocol.
IPv6	Version 6 of the IP protocol.
outbound packet	A packet leaving the switch through any LAN port.
outbound port	Any port on the switch through which traffic leaves the switch.

Term	Use in This Document
outbound port queue	For any port, a buffer that holds outbound traffic until it can leave the switch through that port. There are four outbound queues for each port in the switch: high, medium, normal, and low. Traffic in a port's high priority queue leaves the switch before any traffic in the port's medium priority queue, and so-on.
re-marking (DSCP re-marking)	Assigns a new QoS policy to an outbound packet by changing the DSCP bit settings in the ToS byte.
tagged port membership	Identifies a port as belonging to a specific VLAN and enables VLAN-tagged packets belonging to that VLAN to carry an 802.1p priority setting when outbound from that port. Where a port is an untagged member of a VLAN, outbound packets belonging to that VLAN do not carry an 802.1p priority setting.
Type-of-Service (ToS) byte	Comprised of a three-bit (high-order) precedence field and a five-bit (low-order) Type-of-Service field. Later implementations may use this byte as a six-bit (high-order) Differentiated Services field and a two-bit (low-order) reserved field. See also "IP-precedence bits" and DSCP elsewhere in this table.
upstream device	A device linked directly or indirectly to an inbound switch port. That is, the switch receives traffic from upstream devices.

Overview

QoS settings operate on two levels:

- **Controlling the priority of outbound packets moving through the switch:** Each switch port has four outbound traffic queues; "low", "normal", "medium", and "high" priority. Packets leave the switch port on the basis of their queue assignment and whether any higher queues are empty:

Table 6-1. Port Queue Exit Priorities

Port Queue and 802.1p Priority Values	Priority for Exiting From the Port
Low (1 - 2)	Fourth
Normal (0, 3)	Third
Medium (4 - 5)	Second
High (6 - 7)	First

A QoS configuration enables you to set the outbound priority queue to which a packet is sent. (In an 802.1Q VLAN environment with VLAN-tagged ports, if QoS is *not* configured on the switch, but *is* configured on an upstream device, the priorities carried in the packets determine the forwarding queues in the switch.)

- **Configuring a priority for outbound packets and a service (priority) policy for use by downstream devices:**
  - **DSCP Policy:** This feature enables you to set a priority policy in outbound IP packets. (You can configure downstream devices to read and use this policy.) This method is not dependent on VLAN-tagged ports to carry priority policy to downstream devices, and can:
    - Change the codepoint (the upper six bits) in the ToS byte.
    - Set a new 802.1p priority for the packet.

    (Setting DSCP policies requires IPv4 inbound packets. Refer to the "IPv4" entry under "Terminology" on page 6-6.)
  - **802.1p Priority Rules:** An outbound, VLAN-tagged packet carries an 802.1p priority setting that was configured (or preserved) in the switch. This priority setting ranges from 0 to 7, and can be used by downstream devices having up to eight outbound port queues. Thus, while packets within the switch move at the four priority levels shown in table 6-1, above, they still can carry an 802.1p priority that can be used by downstream devices having more or less than the four priority levels in the switches covered by this guide. Also, if the packet enters the switch with an 802.1p priority setting, QoS can override this setting if configured with an 802.1p priority rule to do so.

Note

If your network uses only one VLAN (and therefore does not require VLAN-tagged ports) you can still preserve 802.1p priority settings in your traffic by configuring the ports as tagged VLAN members on the links between devices you want to honor traffic priorities.

Rule and Policy Limits: The switches covered by this chapter allow the following maximum number of priority rules and/or DSCP policies in any combination:

•2810 Switches: 120

You can configure a QoS priority of 0 through 7 for an outbound packet. When the packet is then sent to a port, the QoS priority determines which outbound queue the packet uses:

Table 6-2. QoS Priority Settings and Operation

QoS Priority Setting	Outbound Port Queue
1 - 2	low priority
0 - 3	normal priority
4 - 5	medium priority
6 - 7	high priority

If a packet is not in a VLAN-tagged port environment, then the QoS settings in table 6-2 control only to which outbound queue the packet goes. Without VLAN tagging, no 802.1p priority is added to the packet for downstream device use. But if the packet is in a VLAN-tagged environment, then the above setting is also added to the packet as an 802.1p priority for use by downstream devices and applications (shown in table 6-3). In either case, an IP packet can also carry a priority policy to downstream devices by using DSCP-marking in the ToS byte.

Table 6-3. Mapping Switch QoS Priority Settings to Device Queues

Priority Setting	Outbound Port Queues in the Switch	802.1p Priority Setting Added to Tagged VLAN Packets Leaving the Switch	Queue Assignment in Downstream Devices With:		
			8 Queues	**3 Queues**	**2 Queues**
1	Queue 1	1 (low priority)	Queue 1	Queue 1	Queue 1
2		2	Queue 2		
0	Queue 2	0 (normal priority)	Queue 3	Queue 2	
3		3	Queue 4		
4	Queue 3	4 (medium priority)	Queue 5	Queue 3	Queue 2
5		5	Queue 6		
6	Queue 4	6 (high priority)	Queue 7		
7		7	Queue 8		

QoS Types for Prioritizing Outbound Packets

The QoS types used in the switches covered in this chapter are a subset of the types used in the Series 5300xl switches (they are referred to as "classifiers"). Also, the 2810 switch searches for type matches in the opposite order of that used in the 5300xl switches. (For more on QoS operation for the 5300xl switches and the 3400cl switches, refer to the *Advanced Traffic Management Guide* for these switch models.)

Note On Using Multiple Criteria

ProCurve recommends that you configure a minimum number of the available QoS types for prioritizing any given packet type. Increasing the number of active type options for a packet type increases the complexity of the possible outcomes and consumes switch resources.

Packet Types and Evaluation Order

The switches covered by this chapter provide five QoS types (packet-matching criteria) you can use to configure QoS priority.

Table 6-4. Switch Type Search Order and Precedence

Search Order	Precedence	QoS type
1	5	Interface - Incoming source-port on the switch
2	4	VLAN Priority
3	3	IP Type of Service (ToS) field (IP packets only)
4	2	Device Priority (destination or source IP address)
5	1 (highest)	UDP/TCP Application Type (port)

The switches use the lowest-to-highest search order shown in table 6-4 to identify the highest-precedence type to apply to any given packet. (Note that this is the opposite of the order used in the 5300xl switches.) If there is only one configured type that matches a given packet, then the switch applies the QoS policy specified in that type. If multiple configured types match a given packet, the switch applies each one in turn to the packet and concludes with the QoS policy for the highest-precedence type. Note that if the highest precedence type is configured to apply a DSCP policy, then both the DSCP in the packet and the 802.1p priority applied to the packet can be changed. However, if the highest precedence type is configured to apply an 802.1p priority rule, only the 802.1p priority in the final QoS match for the packet is changed.

Note

Intermixing lower-precedence types configured with DSCP policies and higher-precedence types configured with 802.1p priority rules is not recommended, as this can result in a packet with an 802.1p priority assigned by one type and a DSCP policy by another type. This is because the search order would allow a lower precedence type configured with a DSCP policy to change both the DSCP and the 802.1p setting in a packet, and then would allow a subsequent, higher precedence type configured with an 802.1p priority rule to change only the 802.1p setting. *To avoid this problem, a DSCP policy option should be applied only on the highest-precedence type in use on the switch or applied to all QoS types in use on the switch.*

In general, the precedence of QoS types should be considered when configuring QoS policies. For example, suppose that a system administrator has used an 802.1p priority rule to assign a high priority for packets received on VLAN 100, but has also used another 802.1p priority rule to assign a normal priority for TCP port 80 packets received on the switch. Since TCP/UDP port precedence supersedes VLAN precedence, all TCP port 80 packets on VLAN 100 will be set to normal priority. For a type precedence listing, see table 6-4, "Switch Type Search Order and Precedence", on page 6-10.

Table 6-5. Precedence Criteria for QoS types

Precedence	Criteria	Overview
1	UDP/TCP	Takes precedence based on a layer 4 UDP or TCP application, with a user-specified application port number (for example, Telnet). **Default state:** Disabled
	If a packet does not meet the criteria for UDP/TCP priority, then precedence defaults to the Device Priority type, below.	
2	Device Priority (IP Address)	Takes precedence based on an inbound packet having a particular destination or source IP address. QoS applies the following IP address limits: • Up to 60 IP addresses If a given packet has a destination IP address matching a QoS configuration, this packet takes precedence over another packet that has the matching IP address as a source address. (This can occur, for example, on an outbound port in a switch mesh environment.) Also, if the source and destination IP addresses (SA and DA) in the same packet match for different QoS policies, the DA takes precedence. **Default state:** No IP address prioritization.
	If a packet does not meet the criteria for device priority, then precedence defaults to the IP Type of Service (ToS) type, below.	
3	IP Type-of-Service (IP ToS)	Takes precedence based on the TOS field in IP packets. (Applies only to IP packets.) The ToS field is configured by an upstream device or application before the packet enters the switch. • **IP Precedence Mode:** QoS reads an inbound packet's IP precedence (upper three) bits in the Type-of-Service (ToS) byte and automatically assigns an 802.1p priority to the packet (if specified in the QoS configuration) for outbound transmission. • **Differentiated Services (Diffserve) Mode:** QoS reads an inbound IP packet's differentiated services, or codepoint (upper six), bits of the Type-of-Service (TOS) byte. Packet prioritization depends on the configured priority for the codepoint. (Some codepoints default to the DSCP standard, but can be overridden.) For more on IP ToS, see "QoS IP Type-of-Service (ToS) Policy and Priority" on page 6-33. **Default state:** Disabled.
	If a packet does not meet the criteria for ToS priority, then precedence defaults to the VLAN type	
4	VLAN Priority	Takes precedence based on the ID number of the VLAN in which the inbound packet exists. For example, if the default VLAN (VID = 1) and the "Blue" VLAN (with a VID of 20) are both assigned to a port, and Blue VLAN traffic is more important, you can configure QoS to give Blue VLAN traffic a higher priority than default VLAN traffic. (Priority is applied on the outbound port.) **Default state:** No-override.
	If a packet does not meet the criteria for VLAN priority, then precedence defaults to the Interface (Source-Port) type, below.	

<table>
<tr><th>Precedence</th><th>Criteria</th><th>Overview</th></tr>
<tr><td>5</td><td>Interface (Source-Port)</td><td>Takes precedence based on the Interface (that is, the port on which the packet entered the switch).</td></tr>
<tr><td></td><td colspan="2">If a packet does not meet the criteria for source-port priority, then precedence defaults to Incoming 802.1p criteria, below</td></tr>
<tr><td>6</td><td>Incoming 802.1p Priority</td><td>Where a VLAN-tagged packet enters the switch through a port that is a tagged member of that VLAN, if QoS is not configured to override the packet’s priority setting, the switch uses the packet’s existing 802.1p priority (assigned by an upstream device or application) to determine which inbound and outbound port queue to use. If there is no QoS policy match on the packet, and it then leaves the switch through a port that is a tagged member of the VLAN, then there is no change to its 802.1p priority setting. If the packet leaves the switch through a port that is an untagged member of the VLAN, the 802.1p priority is dropped.
<table>
<tr><th>Entering (Inbound) 802.1p Priority</th><th>Outbound Port Queue</th><th>Exiting (Outbound) 802.1p Priority</th></tr>
<tr><td>1 - 2</td><td>Low</td><td>1 - 2</td></tr>
<tr><td>0 - 3</td><td>Normal</td><td>0 - 3</td></tr>
<tr><td>4 - 5</td><td>Medium</td><td>4 - 5</td></tr>
<tr><td>6 - 7</td><td>High</td><td>6 - 7</td></tr>
</table></td></tr>
<tr><td></td><td colspan="2">If a packet does not meet the criteria for Incoming 802.1p priority, then the packet goes to the “normal” outbound queue of the appropriate port. If the packet entered the switch through a port that is an untagged member of a VLAN, but exits through a VLAN-tagged port, then an 802.1Q field, including an 802.1p priority, is added to the packet header. If no QoS policy is configured or applied to the packet, then the 802.1p priority of 0 (normal) is assigned to the packet for outbound transmission.</td></tr>
</table>

Preparation for Configuring QoS

QoS operates in VLAN-tagged and VLAN-untagged environments. If your network does not use multiple VLANs, you can still implement the 802.1Q VLAN capability for packets to carry their 802.1p priority to the next downstream device. To do so, configure ports as VLAN-tagged members on the links between switches and routers in your network infrastructure.

Table 6-6. Summary of QoS Capabilities

Outbound Packet Options	**Port Membership in VLANs**	
	Tagged	**Untagged**
Control Port Queue Priority for Packet Types	Yes	Yes
Carry 802.1p Priority Assignment to Next Downstream Device	Yes	No
Carry DSCP Policy to Downstream Devices. The policy includes:	Yes [1]	Yes [1]
Assigning a ToS Codepoint		
Assigning an 802.1p Priority [2] to the Codepoint		

[1] Except for non-IPv4 packets or packets processed using the QoS IP-Precedence method, which does not include the DSCP policy option. Also, to use a service policy in this manner, the downstream devices must be configured to interpret and use the DSCP carried in the IP packets.

[2] This priority corresponds to the 802.1p priority scheme and is used to determine the packet's port queue priority. When used in a VLAN-tagged environment, this priority is also assigned as the 802.1p priority carried outbound in packets having an 802.1Q field in the header.

Steps for Configuring QoS on the Switch

1. Determine the QoS policy you want to implement. This includes analyzing the types of traffic flowing through your network and identifying one or more traffic types to prioritize. In order of QoS precedence, these are:

   a. UDP/TCP applications

   b. Device Priority—destination or source IP address (Note that destination has precedence over source. See Table 6-5.)

   c. IP Type-of-Service Precedence Bits (Leftmost three bits in the ToS field of IP packets)

   d. VLAN

   e. Interface

   For more on how QoS operates with the preceding traffic types, see "Precedence Criteria for QoS types", on page 6-11.)

2. Select the QoS option you want to use. Table 6-7 lists the traffic types (QoS types) and the QoS options you can use for prioritizing or setting a policy on these traffic types:

Table 6-7. Applying QoS Options to Traffic Types Defined by QoS Types

QoS Options for Prioritizing Outbound Traffic		QoS Types				
		UDP/ TCP	IP Device	IP-ToS Precedence	VLAN -ID	Source -Port
Option 1: Configure 802.1p Priority Rules Only	Prioritize traffic by sending specific packet types (determined by QoS type) to different outbound port queues on the switch. Rely on VLAN-tagged ports to carry packet priority as an 802.1p value to downstream devices.	Yes	Yes	Yes [1]	Yes	Yes
Option 2: Configure ToS DSCP Policies with 802.1p Priorities	Prioritize traffic by sending specific packet types (determined by QoS type) to different outbound port queues on the switch. Propagate a service policy by reconfiguring the DSCP in outbound IP packets according to packet type. The packet is placed in an outbound port queue according to the 802.1p priority configured for that DSCP policy. (The policy assumes that downstream devices can be configured to recognize the DSCP in IP packets and implement the service policy it indicates.) Use VLAN-tagged ports to include packet priority as an 802.1p value to downstream devices.	Yes	Yes	No	Yes	Yes

[1] In this mode the configuration is fixed. You cannot change the automatic priority assignment when using IP-ToS Precedence as a QoS type.

3. If you want to include 802.1p priority settings in outbound packets, ensure that tagged VLANs are configured on the appropriate downstream links.

4. Determine the actual QoS configuration changes you will need to make on each QoS-capable device in your network in order to implement the desired policy. Also, if you want downstream devices to read and use DSCPs in IP packets from the switch, configure them to do so by enabling ToS Differentiated Service mode and making sure the same DSCP policies are configured.

5. Before configuring QoS on a switch, refer to the next section, “Planning a QoS Configuration” for information on per-port QoS resource use.

Planning a QoS Configuration

QoS uses resources in a way that requires attention to rule usage when planning a QoS configuration. Otherwise, there is an increased possibility of oversubscribing resources, which means that at some point the switch would not support further QoS configuration.

Prioritizing and Monitoring QoS Configuration Options

Plan and implement your QoS configuration in descending order of feature importance. This helps to ensure that the most important features are configured first. Also, if insufficient rule resources become a problem, this approach can help you recognize how to distribute the desired feature implementations across multiple switches to achieve your objectives. For example, a given type of traffic may be of higher importance than other traffic types you want to expedite by using QoS. In this case you should plan and configure your QoS resource usage first for the most important traffic type before configuring QoS resource usage for other traffic types. If insufficient resources remain for all of the QoS implementation you want, try spreading this implementation across multiple switches.

QoS Resource Usage and Monitoring

Each of the five types of QoS features has a maximum number of rules that can be configured for that type.

The following table shows the maximum number of rules available for each type and how many resources are used each time you configure a rule. You can use a maximum of five QoS resources, after which you will not be able to configure more rules.

You can also configure rules for trunks, and the rule will apply to all the ports in that trunk. If you add a port that has a rule configured for it to a trunk with a configured rule or rules, the port “drops” its rule and follows the rules for the trunk.

Table 6-8. Rule Maximums and QoS Resources Required

QoS Type	Rules Maximum	QoS Resources Required per Type
Interface	96	1
VLAN	120	1
Type-of-Service	64	1
Device priority	120	2
TCP/UDP Layer 4	30	2

Planning and Monitoring Rule Usage

The following two CLI commands are useful for planning and monitoring rule usage in a QoS configuration.

Syntax qos resources help

Provides a quick reference on how QoS uses rule resources for each configuration option.

Syntax show qos resources

Shows the maximum number of rules available for each type, the number of resources required by each type, and the resources that have been used so far. This command is useful for verifying rule availability as you proceed with configuring QoS.

Managing QoS Resource Consumption

Extensive QoS configurations can either fully subscribe the resources available or leave an insufficient number of rules available for configuring another QoS policy of that type. Problems with an insufficient number of rules available can occur in the following QoS scenarios:

- Attempting to configure a policy when there are an insufficient number of rules available for a particular type
- Attempting to configure a policy when an additional resource is required and five QoS resources have already been used
- Attempting to add a port to a QoS-configured VLAN where the policy already on the VLAN requires more rule resources than are available.

The policy will not be configured on any ports subject to the command. The CLI displays a message similar to the following:

```
Unable to add this QoS rule. Maximum number already
reached.
```

Adding a Port to a QoS-Configured VLAN Without Enough Rules Available. When you add a port to an existing, QoS-configured VLAN, the switch attempts to apply the VLAN's QoS configuration to the port. If there are insufficient rule resources to add the VLAN's QoS configuration:

- The port *is* added to the VLAN.
- The QoS types configured on the VLAN are *not* added to the port, which means that the port does not honor the QoS policies configured for the VLAN.

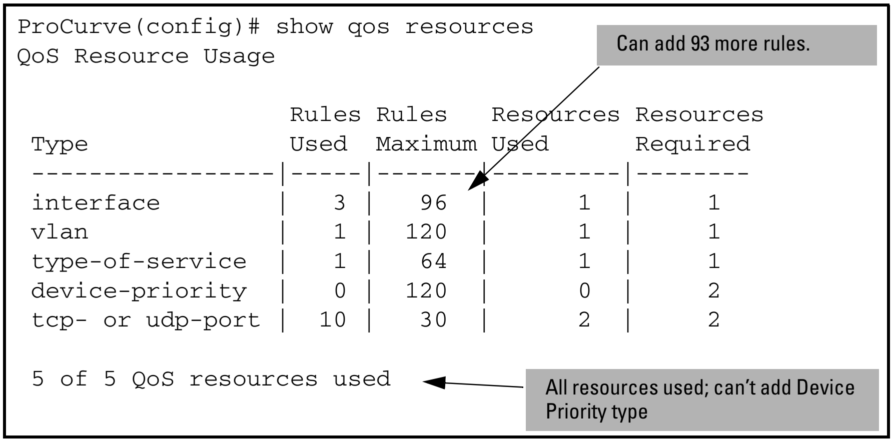
Troubleshooting a Shortage of Resources

The inability to configure more rules is caused by:

- existing QoS configurations consuming all the rules
- all the QoS resources being used.

Do the following to enable configuration of the desired policy:

1. Use the **show qos resources** command to view your current rule and resource usage.



```
ProCurve(config)# show qos resources
QoS Resource Usage

                   Rules Rules   Resources Resources
 Type              Used  Maximum Used      Required
 -----------------|-----|-------|---------|--------
 interface        |   3 |   96  |       1 |      1
 vlan             |   1 |  120  |       1 |      1
 type-of-service  |   1 |   64  |       1 |      1
 device-priority  |   0 |  120  |       0 |      2
 tcp- or udp-port |  10 |   30  |       2 |      2

 5 of 5 QoS resources used
```

Can add 93 more rules.

All resources used; can't add Device Priority type

Figure 6-3. Example of Inspecting Available Rule Resources

2. Use the **show qos** commands to identify the currently configured QoS policies.

3. Determine which of the existing policies you can remove to free up rule resources for the QoS policy you want to implement. Depending on your network topology and configuration, you can free up rule resources by moving some policies to other devices. Another alternative is to inspect the switch's existing QoS configuration for unnecessary entries or inefficient applications that could be removed or revised to achieve the desired policies with less resource usage. Table 6-8 on page 6-15, or the information displayed by the **qos resources help** command, can help you to determine the resource usage of QoS policies.

Using QoS Types To Configure QoS for Outbound Traffic

QoS Feature	Default	Menu	CLI	Web
UDP/TCP Priority	Disabled	—	page 6-20	Refer to Online Help.
IP-Device Priority	Disabled	—	page 6-27	“
IP Type-of-Service Priority	Disabled	—	page 6-33	“
VLAN-ID Priority	Disabled	—	page 6-46	“
Interface (Source-Port) Priority	Disabled	—	page 6-52	“

Note

In addition to the information in this section on the various QoS types, refer to “QoS Operating Notes and Restrictions” on page 6-66.

Viewing the QoS Configuration

Examples of the **show qos** output are included with the example for each priority type.

Syntax: show qos < *priority-type* >

tcp-udp-port-priority

Displays the current TCP/UDP port priority configuration. Refer to figure 6-8 on page 6-26.

device-priority

Displays the current device (IP address) priority configuration. Refer to figure 6-9 on page 6-29.

type-of-service

Displays the current type-of-service priority configuration. The display output differs according to the ToS option used:

- *IP Precedence: Refer to figure 6-13 on page 6-34.*
- *Diffserve: Refer to figure 6-15 on page 6-38.*

vlan-priority

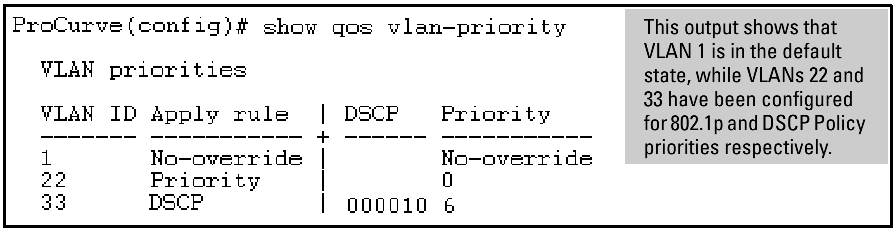
Displays the current VLAN priority configuration. Refer to figure 6-22 on page 6-48.

port-priority

Displays the current Interface (source-port) priority configuration. Refer to figure 6-27 on page 6-53.

No Override

By default, the IP ToS, VLAN-ID, and interface **show** outputs automatically list **No-override** for priority options that have not been configured. This means that if you do not configure a priority for a specific option, QoS does not prioritize packets to which that option applies, resulting in the **No override** state. In this case, IP packets received through a VLAN-tagged port receive whatever 802.1p priority they carry in the 802.1Q tag in the packet's header. VLAN-Tagged packets received through an untagged port are handled in the switch with "normal" priority. For example, figure 6-4 below shows a qos VLAN priority output in a switch where nondefault priorities exist for VLANs 22 and 33, while VLAN 1 remains in the default configuration.



```
ProCurve(config)# show qos vlan-priority

  VLAN priorities

  VLAN ID Apply rule   | DSCP   Priority
  ------- ------------ + ------ -----------
  1       No-override  |        No-override
  22      Priority     |        0
  33      DSCP         | 000010 6
```

This output shows that VLAN 1 is in the default state, while VLANs 22 and 33 have been configured for 802.1p and DSCP Policy priorities respectively.

Figure 6-4. Example of the Show QoS Output for VLAN Priority

QoS UDP/TCP Priority

QoS Type Precedence: 1

When you use UDP or TCP (Layer 4) as a QoS type, traffic carrying the specified UDP/TCP port number(s) is marked with the UDP/TCP type's configured priority level.

The UDP/TCP type uses two resources instead of one—one resource for the source IP address and one for the destination IP address. Destination IP addresses have precedence over source IP addresses. You can configure a maximum of 30 rules, or 15 rulesets (source and destination IP addresses comprise a ruleset).

Note

UDP/TCP QoS applications do not support IPv4 packets with IP options or layer-2 SAP encapsulation. For more information on packet-type restrictions, refer to "Details of Packet Criteria and Restrictions for QoS Support", on page 6-66.

Options for Assigning Priority. Priority control options for TCP or UDP packets carrying a specified TCP or UDP port number include:

- 802.1p priority
- DSCP policy (Assigning a new DSCP and an associated 802.1p priority; inbound packets must be IPv4.)

For a given TCP or UDP port number, you can use only one of the above options at a time. However, for different port numbers, you can use different options.

TCP/UDP Port Number Ranges. There are three ranges:

- Well-Known Ports: 0 - 1023
- Registered Ports: 1024 - 49151
- Dynamic and/or Private Ports: 49152 - 65535

For more information, including a listing of UDP/TCP port numbers, go to the *Internet Assigned Numbers Authority* (IANA) website at:

http://www.iana.org

Then click on:

Protocol Number Assignment Services

P (Under "Directory of General Assigned Numbers" heading)

Port Numbers

Assigning 802.1p Priority Based on TCP or UDP Port Number

This option assigns an 802.1p priority to (IPv4) TCP or UDP packets as described below.

Syntax: qos < udp-port | tcp-port > < *tcp or udp port number* > priority < 0 - 7 >

Configures an 802.1p priority for outbound packets having the specified TCP or UDP application port number. This priority determines the packet's queue in the outbound port to which it is sent. If the packet leaves the switch on a tagged port, it carries the 802.1p priority with it to the next downstream device. (Default: Disabled)

Note: *On switches covered in this guide, this feature is not supported for IPv4 packets with IP options. For more information on packet-type restrictions, refer to table 6-12 on page 6-66.*

no qos < udp-port | tcp-port > < *tcp-udp port number* >

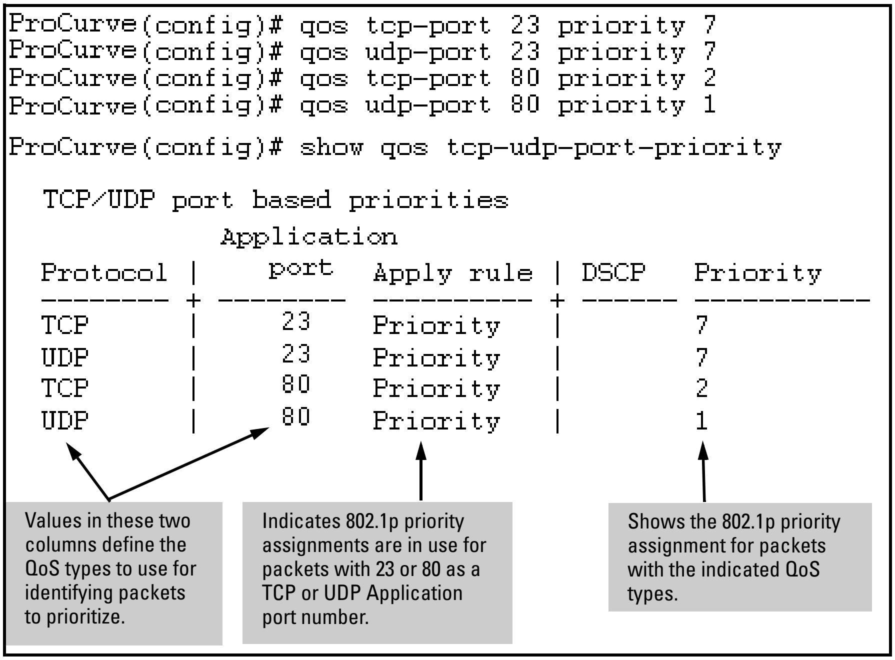
Deletes the specified UDP or TCP port number as a QoS type.

show qos tcp-udp-port-priority

Displays a listing of all TCP and UDP QoS types currently in the running-config file.

For example, configure and list 802.1p priority for the following UDP and TCP port prioritization:

TCP/UDP Port	802.1p Priority for TCP	802.1p Priority for UDP
TCP Port 23 (Telnet)	7	7
UDP Port 23 (Telnet)	7	7
TCP Port 80 (World Wide Web HTTP)	2	2
UDP Port 80 (World Wide Web HTTP)	1	1



```
ProCurve(config)# qos tcp-port 23 priority 7
ProCurve(config)# qos udp-port 23 priority 7
ProCurve(config)# qos tcp-port 80 priority 2
ProCurve(config)# qos udp-port 80 priority 1

ProCurve(config)# show qos tcp-udp-port-priority

  TCP/UDP port based priorities

             Application
  Protocol |    port    Apply rule | DSCP   Priority
  -------- + --------  ---------- + ------ -----------
  TCP      |    23      Priority   |        7
  UDP      |    23      Priority   |        7
  TCP      |    80      Priority   |        2
  UDP      |    80      Priority   |        1
```

Values in these two columns define the QoS types to use for identifying packets to prioritize.

Indicates 802.1p priority assignments are in use for packets with 23 or 80 as a TCP or UDP Application port number.

Shows the 802.1p priority assignment for packets with the indicated QoS types.

Figure 6-5. Example of Configuring and Listing 802.1p Priority Assignments on TCP/UDP Ports

Assigning a DSCP Policy Based on TCP or UDP Port Number

This option assigns a previously configured DSCP policy (codepoint and 802.1p priority) to (IPv4) TCP or UDP packets having the specified port number. That is, the switch:

1. Selects an incoming IP packet if the TCP or UDP port number it carries matches the port number specified in the TCP or UDP type (as shown in figure 6-5, above).

2. Overwrites (re-marks) the packet's DSCP with the DSCP configured in the switch for such packets.
3. Assigns the 802.1p priority configured in the switch for the new DSCP. (Refer to "Differentiated Services Codepoint (DSCP) Mapping" on page 6-58.)
4. Forwards the packet through the appropriate outbound port queue.

Note on Combining Policy Types

"Mixing" ToS DSCP policies and 802.1p priorities is not recommended. Refer to the Note on page 6-10.

For more on DSCP, refer to "Terminology" on page 6-6.

Steps for Creating a DSCP Policy Based on TCP/UDP Port Number Types. This procedure creates a DSCP policy for IPv4 packets carrying the selected UDP or TCP port-number type.

1. Identify the TCP or UDP port-number type you want to use for assigning a DSCP policy.
2. Determine the DSCP policy for packets carrying the selected TCP or UDP port number.
   a. Determine the DSCP you want to assign to the selected packets. (This codepoint will be used to overwrite (re-mark) the DSCP carried in packets received from upstream devices.)
   b. Determine the 802.1p priority you want to assign to the DSCP.
3. Configure the DSCP policy by using **qos dscp-map** to configure the priority to the codepoint you selected in step 2a. (For details, refer to the example later in this section, and to "Differentiated Services Codepoint (DSCP) Mapping" on page 6-58.)

Note

A codepoint must have an 802.1p priority assignment (0 - 7) before you can configure a policy for prioritizing packets by TCP or UDP port numbers. If a codepoint you want to use shows **No-override** in the **Priority** column of the DSCP map (**show qos dscp-map**), then you must assign a 0 - 7 priority before proceeding.

4. Configure the switch to assign the DSCP policy to packets with the specified TCP or UDP port number.

Syntax: qos dscp-map < *codepoint* > priority < 0 - 7 >

This command is optional if a priority has already been assigned to the < ***codepoint*** *>. The command creates a DSCP policy by assigning an 802.1p priority to a specific DSCP. When the switch applies this policy to a packet, the priority determines the packet's queue in the outbound port to which it is sent. If the packet leaves the switch on a tagged port, it carries the 802.1p priority with it to the next downstream device. For IPv4 packets, the DSCP will be replaced by the codepoint specified in this command. (Default:* **No-override** *for most codepoints. See table 6-11 on page 6-59.)*

Syntax: qos < udp-port | tcp-port > < *tcp or udp port number* > dscp < *codepoint* >

Assigns a DSCP policy to outbound packets having the specified TCP or UDP application port number and overwrites the DSCP in these packets with the assigned ***<codepoint>*** *value. This policy includes an 802.1p priority and determines the packet's queue in the outbound port to which it is sent. (The* ***<codepoint>*** *must be configured with an 802.1p setting. See step 3 on page 6-23.) If the packet leaves the switch on a tagged port, it carries the 802.1p priority with it to the next downstream device. (Default:* **No-override***)*

no qos < udp-port | tcp-port > < *tcp-udp port number* >

Deletes the specified UDP or TCP port number as a QoS type.

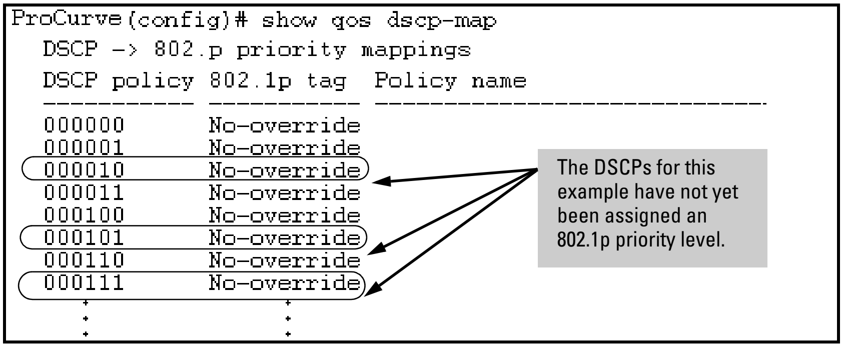
show qos tcp-udp-port-priority

Displays a listing of all TCP and UDP QoS types currently in the running-config file.

For example, suppose you wanted to assign these DSCP policies to the packets identified by the indicated UDP and TDP port applications:

Port Applications	DSCP Policies	
	DSCP	Priority
23-UDP	000111	7
80-TCP	000101	5
914-TCP	000010	1
1001-UDP	000010	1

1. Determine whether the DSCPs already have priority assignments, which could indicate use by existing applications. (Also, a DSCP must have a priority configured before you can assign any QoS types to use it.)

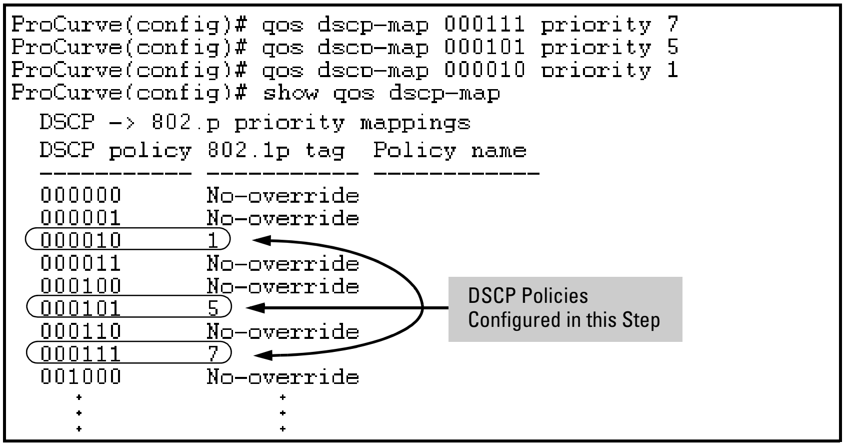


```
ProCurve(config)# show qos dscp-map
  DSCP -> 802.p priority mappings
  DSCP policy 802.1p tag  Policy name
  ----------- ----------- -------------------------------
  000000      No-override
  000001      No-override
  000010      No-override
  000011      No-override
  000100      No-override
  000101      No-override
  000110      No-override
  000111      No-override
     .            .
     .            .
     .            .
```

The DSCPs for this example have not yet been assigned an 802.1p priority level.

Figure 6-6. Display the Current DSCP-Map Configuration

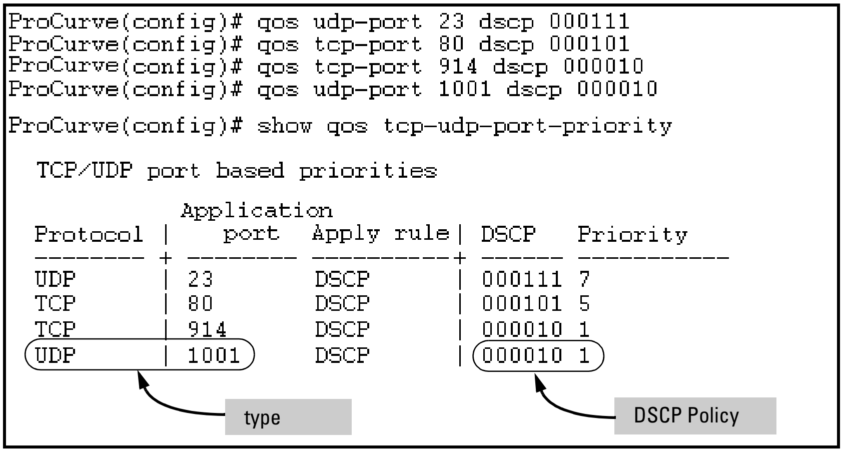
2. Configure the DSCP policies for the codepoints you want to use.



```
ProCurve(config)# qos dscp-map 000111 priority 7
ProCurve(config)# qos dscp-map 000101 priority 5
ProCurve(config)# qos dscp-map 000010 priority 1
ProCurve(config)# show qos dscp-map
  DSCP -> 802.p priority mappings
  DSCP policy 802.1p tag  Policy name
  ----------- ----------- ------------
  000000      No-override
  000001      No-override
  000010      1
  000011      No-override
  000100      No-override
  000101      5
  000110      No-override
  000111      7
  001000      No-override
     .              .
     .              .
     .              .
```

Figure 6-7. Assign Priorities to the Selected DSCPs

3. Assign the DSCP policies to the selected UDP/TCP port applications and display the result.



```
ProCurve(config)# qos udp-port 23 dscp 000111
ProCurve(config)# qos tcp-port 80 dscp 000101
ProCurve(config)# qos tcp-port 914 dscp 000010
ProCurve(config)# qos udp-port 1001 dscp 000010

ProCurve(config)# show qos tcp-udp-port-priority

  TCP/UDP port based priorities

             Application
  Protocol |    port   Apply rule| DSCP   Priority
  -------- + --------  ----------+ ------ -----------
  UDP      | 23        DSCP      | 000111 7
  TCP      | 80        DSCP      | 000101 5
  TCP      | 914       DSCP      | 000010 1
  UDP      | 1001      DSCP      | 000010 1
```

Figure 6-8. The Completed DSCP Policy Configuration for the Specified UDP/TCP Port Applications

The switch will now apply the DSCP policies in figure 6-8 to IPv4 packets received in the switch with the specified UDP/TCP port applications. This means the switch will:

- Overwrite the original DSCPs in the selected packets with the new DSCPs specified in the above policies.
- Assign the 802.1p priorities in the above policies to the selected packets.

QoS IP-Device Priority

QoS Type Precedence: 2

The IP device-priority option, which applies only to IPv4 packets, uses two resources instead of one—one resource for the source IP address and one for the destination IP address. Destination IP addresses have precedence over source IP addresses. You can configure a maximum of 120 rules, or 60 rulesets (source and destination IP addresses comprise a ruleset).

Where a particular device-IP address type has the highest precedence in the switch for traffic addressed to or from that device, then traffic received on the switch with that address is marked with the IP address type's configured priority level.

Note

The switch does not allow a QoS IP-device priority for the Management VLAN IP address, if configured. If there is no Management VLAN configured, then the switch does not allow configuring a QoS IP-device priority for the Default VLAN IP address.

IP address QoS does not support layer-2 SAP encapsulation.   For more information on packet-type restrictions, refer to "Details of Packet Criteria and Restrictions for QoS Support", on page 6-66.

Options for Assigning Priority. Priority control options for packets carrying a specified IP address include:

- 802.1p priority
- DSCP policy (Assigning a new DSCP and an 802.1p priority; inbound packets must be IPv4.)

(For operation when other QoS types apply to the same traffic, refer to "QoS Types for Prioritizing Outbound Packets" on page 6-9.)

For a given IP address, you can use only one of the above options at a time. However, for different IP addresses, you can use different options.

Assigning a Priority Based on IP Address

This option assigns an 802.1p priority to all IPv4 packets having the specified IP address as either a source or destination. (If both match, the priority for the IP destination address has precedence.)

Syntax: qos device-priority < *ip-address* > priority < 0 - 7 >

Configures an 802.1p priority for outbound packets having the specified IP address. This priority determines the packet's queue in the outbound port to which it is sent. If the packet leaves the switch on a tagged port, it carries the 802.1p priority with it to the next downstream device. (Default: Disabled)

no qos device-priority < *ip-address* >

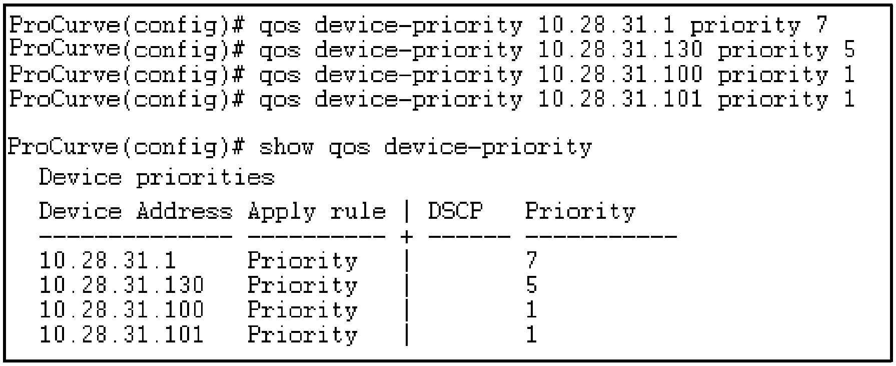
Removes the specified IP device-priority QoS type and resets the priority for that VLAN to **No-override**.

show qos device-priority

Displays a listing of all IP device-priority QoS types currently in the running-config file.

For example, configure and list the 802.1p priority for packets carrying the following IP addresses:

IP Address	802.1p Priority
10.28.31.1	7
10.28.31.130	5
10.28.31.100	1
10.28.31.101	1



```
ProCurve(config)# qos device-priority 10.28.31.1 priority 7
ProCurve(config)# qos device-priority 10.28.31.130 priority 5
ProCurve(config)# qos device-priority 10.28.31.100 priority 1
ProCurve(config)# qos device-priority 10.28.31.101 priority 1

ProCurve(config)# show qos device-priority
  Device priorities

  Device Address Apply rule | DSCP   Priority
  -------------- ---------- + ------ -----------
  10.28.31.1     Priority   |        7
  10.28.31.130   Priority   |        5
  10.28.31.100   Priority   |        1
  10.28.31.101   Priority   |        1
```

Figure 6-9. Example of Configuring and Listing 802.1p Priority Assignments for Packets Carrying Specific IP Addresses

Assigning a DSCP Policy Based on IP Address

This option assigns a previously configured DSCP policy (codepoint and 802.1p priority) to outbound IP packets having the specified IP address (either source or destination). That is, the switch:

1. Selects an incoming IPv4 packet on the basis of the source or destination IP address it carries.
2. Overwrites the packet’s DSCP with the DSCP configured in the switch for such packets, and assigns the 802.1p priority configured in the switch for the new DSCP. (Refer to “Differentiated Services Codepoint (DSCP) Mapping” on page 6-58.)
3. Forwards the packet through the appropriate outbound port queue.

Note on Combining Policy Types

“Mixing” ToS DSCP policies and 802.1p priorities is not recommended. Refer to the Note on page 6-10.

For more on DSCP, refer to “Terminology” on page 6-6.

Steps for Creating a Policy Based on IP Address. This procedure creates a DSCP policy for IPv4 packets carrying the selected IP address (source or destination).

1. Identify the IP address you want to use as a type for assigning a DSCP policy.

2. Determine the DSCP policy for packets carrying the selected IP address:
   a. Determine the DSCP you want to assign to the selected packets. (This codepoint will be used to overwrite the DSCP carried in packets received from upstream devices.)
   b. Determine the 802.1p priority you want to assign to the DSCP.
3. Configure the DSCP policy by using **dscp-map** to configure the priority to the codepoint you selected in step 2a. (For details, refer to "Differentiated Services Codepoint (DSCP) Mapping" on page 6-58.)

Notes

A codepoint must have an 802.1p priority assignment (0 - 7) before you can configure a policy for prioritizing packets by IP address. If a codepoint you want to use shows **No-override** in the **Priority** column of the DSCP map (**show qos dscp-map**), then you must assign a 0 - 7 priority before proceeding.

4. Configure the switch to assign the DSCP policy to packets with the specified IP address.

Syntax: qos dscp-map < *codepoint* > priority < 0 - 7 >

This command is optional if a priority has already been assigned to the < ***codepoint*** *>. The command creates a DSCP policy by assigning an 802.1p priority to a specific DSCP. When the switch applies this policy to a packet, the priority determines the packet's queue in the outbound port to which it is sent. If the packet leaves the switch on a tagged port, it carries the 802.1p priority with it to the next downstream device. If the packet is IPv4, the packet's DSCP will be replaced by the codepoint specified in this command. (Default: For most codepoints,* **No-override**. *See figure 6-11 on page 6-59.)*

Syntax: qos device-priority < *ip-address* > dscp < *codepoint* >

Assigns a DSCP policy to packets carrying the specified IP address, and overwrites the DSCP in these packets with the assigned ***< codepoint >*** *value. This policy includes an 802.1p priority and determines the packet's queue in the outbound port to which it is sent. If the packet leaves the switch on a tagged port, it carries the 802.1p priority with it to the next downstream device. (Default:* **No-override***)* no qos device-priority < *ip-address* >

Deletes the specified IP address as a QoS type.

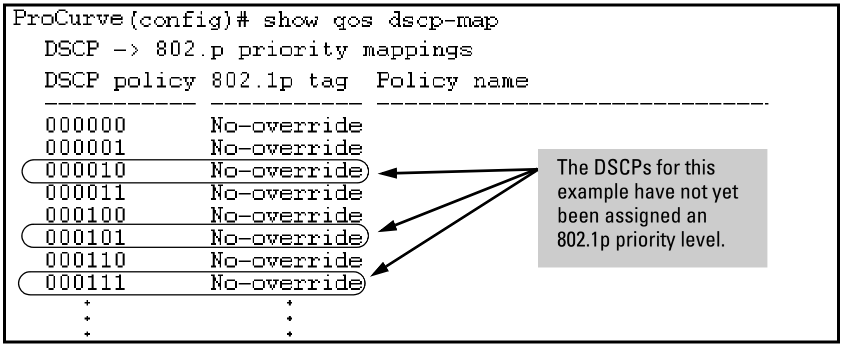
show qos device-priority

Displays a listing of all QoS Device Priority types currently in the running-config file.

For example, suppose you wanted to assign these DSCP policies to the packets identified by the indicated IP addresses:

IP Address	DSCP Policies	
	DSCP	**Priority**
10.28.31.1	000111	7
10.28.31.130	000101	5
10.28.31.100	000010	1
10.28.31.101	000010	1

1. Determine whether the DSCPs already have priority assignments, which could indicate use by existing applications. This is not a problem if the configured priorities are acceptable for all applications using the same DSCP. (Refer to the “Note On Changing a Priority Setting” on page 6-61. Also, a DSCP must have a priority configured before you can assign any QoS types to use it.)

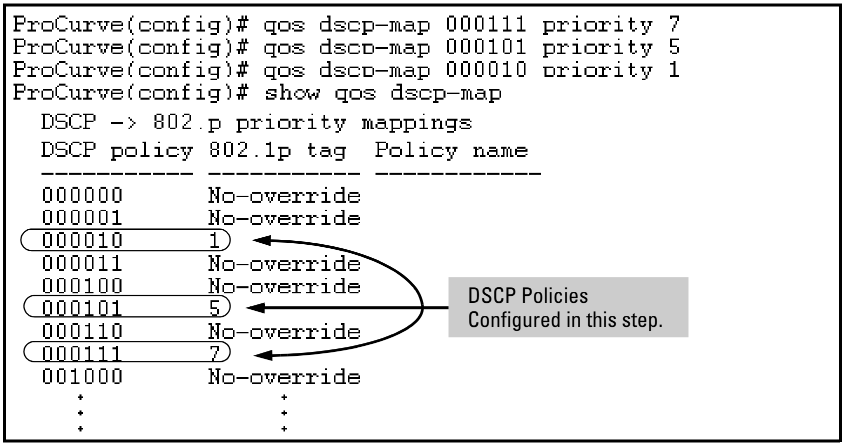


```
ProCurve(config)# show qos dscp-map
  DSCP -> 802.p priority mappings
  DSCP policy 802.1p tag  Policy name
  ----------- ----------- ------------------------------
  000000      No-override
  000001      No-override
  000010      No-override
  000011      No-override
  000100      No-override
  000101      No-override
  000110      No-override
  000111      No-override
     .            .
     .            .
     .            .
```

The DSCPs for this example have not yet been assigned an 802.1p priority level.

Figure 6-10. Display the Current DSCP-Map Configuration

2. Configure the priorities for the DSCPs you want to use.

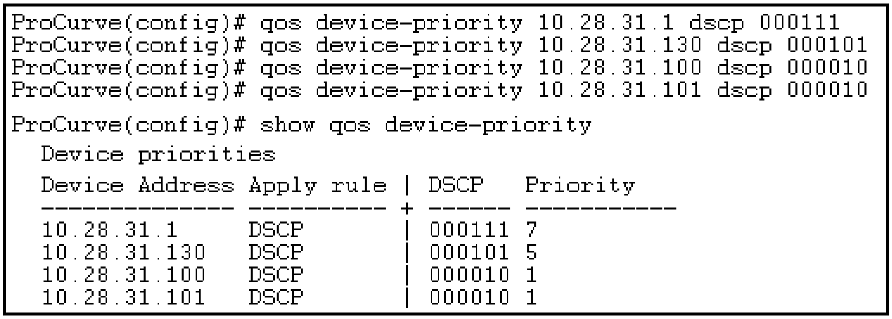


```
ProCurve(config)# qos dscp-map 000111 priority 7
ProCurve(config)# qos dscp-map 000101 priority 5
ProCurve(config)# qos dscp-map 000010 priority 1
ProCurve(config)# show qos dscp-map
  DSCP -> 802.p priority mappings
  DSCP policy 802.1p tag  Policy name
  ----------- ----------- ------------
  000000      No-override
  000001      No-override
  000010      1
  000011      No-override
  000100      No-override
  000101      5
  000110      No-override
  000111      7
  001000      No-override
     .              .
     .              .
     .              .
```

DSCP Policies Configured in this step.

Figure 6-11. Assigning 802.1p Priorities to the Selected DSCPs

3. Assign the DSCP policies to the selected device IP addresses and display the result.



```
ProCurve(config)# qos device-priority 10.28.31.1 dscp 000111
ProCurve(config)# qos device-priority 10.28.31.130 dscp 000101
ProCurve(config)# qos device-priority 10.28.31.100 dscp 000010
ProCurve(config)# qos device-priority 10.28.31.101 dscp 000010

ProCurve(config)# show qos device-priority
  Device priorities
  Device Address Apply rule | DSCP   Priority
  -------------- ---------- + ------ -----------
  10.28.31.1     DSCP       | 000111 7
  10.28.31.130   DSCP       | 000101 5
  10.28.31.100   DSCP       | 000010 1
  10.28.31.101   DSCP       | 000010 1
```

Figure 6-12. The Completed Device-Priority/Codepoint Configuration

The switch will now apply the DSCP policies in figure 6-11 to IPv4 packets received on the switch with the specified IP addresses (source or destination). This means the switch will:

- Overwrite the original DSCPs in the selected packets with the new DSCPs specified in the above policies.
- Assign the 802.1p priorities in the above policies to the appropriate packets.

QoS IP Type-of-Service (ToS) Policy and Priority

QoS Type Precedence: 3

This feature applies only to IPv4 traffic and performs either of the following:

- **ToS IP-Precedence Mode:** All IP packets generated by upstream devices and applications include precedence bits in the ToS byte. Using this mode, the switch uses these bits to compute and assign the corresponding 802.1p priority.
- **ToS Differentiated Services (Diffserv) Mode:** This mode requires knowledge of the codepoints set in IP packets by the upstream devices and applications. It uses the ToS codepoint in IP packets coming from upstream devices and applications to assign 802.1p priorities to the packets. You can use this option to do both of the following:
  - **Assign a New Prioritization Policy:** A "policy" includes both a codepoint and a corresponding 802.1p priority. This option selects an incoming IPv4 packet on the basis of its codepoint and assigns a new codepoint and corresponding 802.1p priority. (Use the **qos dscp-map** command to specify a priority for any codepoint—page 6-58.)
  - **Assign an 802.1p Priority:** This option reads the DSCP of an incoming IPv4 packet and, without changing this codepoint, assigns the 802.1p priority to the packet, as configured in the DSCP Policy Table (page 6-58). This means that a priority value of 0 - 7 must be configured for a DSCP before the switch will attempt to perform a QoS match on the packet's DSCP bits.

  Before configuring the ToS Diffserv mode, you must use the **dscp-map** command to configure the desired 802.1p priorities for the codepoints you want to use for either option. This command is illustrated in the following examples and is described under "Differentiated Services Codepoint (DSCP) Mapping" on page 6-58.

Unless IP-Precedence mode and Diffserv mode are both disabled (the default setting), enabling one automatically disables the other. *For more on ToS operation, refer to "Details of QoS IP Type-of-Service" on page 6-43.*

Notes

"Mixing" ToS DSCP policies and 802.1p priorities is not recommended. Refer to the Note on page 6-10.

IP-ToS QoS does not support layer-2 SAP encapsulation. For more information on packet-type restrictions, refer to "Details of Packet Criteria and Restrictions for QoS Support", on page 6-66.

Assigning an 802.1p Priority to IPv4 Packets on the Basis of the ToS Precedence Bits

If a device or application upstream of the switch sets the precedence bits in the ToS byte of IPv4 packets, you can use this feature to apply that setting for prioritizing packets for outbound port queues. If the outbound packets are in a tagged VLAN, this priority is carried as an 802.1p value to the adjacent downstream devices.

Syntax: qos type-of-service ip-precedence

Causes the switch to automatically assign an 802.1p priority to all IPv4 packets by computing each packet's 802.1p priority from the precedence bits the packet carries. This priority determines the packet's queue in the outbound port to which it is sent. If the packet leaves the switch on a tagged port, it carries the 802.1p priority with it to the next downstream device. (ToS IP Precedence Default: Disabled)

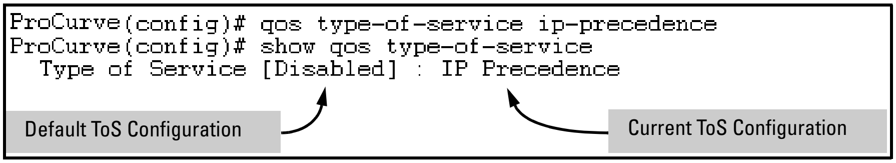
no qos type-of-service

Disables all ToS type operation, including prioritization using the precedence bits.

show qos type-of-service

When ip-precedence is enabled (or if neither ToS option is configured), shows the ToS configuration status. If diff-services is enabled, lists codepoint data as described under "Assigning a DSCP Policy on the Basis of the DSCP in IPv4 Packets Received from Upstream Devices" on page 6-39.

With this option, prioritization of outbound packets relies on the IP-Precedence bit setting that IP packets carry with them from upstream devices and applications. To configure and verify this option:



```
ProCurve(config)# qos type-of-service ip-precedence
ProCurve(config)# show qos type-of-service
  Type of Service [Disabled] : IP Precedence
```

Figure 6-13. Example of Enabling ToS IP-Precedence Prioritization

To replace this option with the ToS diff-services option, just configure **diff-services** as described below, which automatically disables IP-Precedence. To disable IP-Precedence without enabling the diff-services option, use this command:

```
ProCurve(config)# no qos type-of-service
```

Assigning an 802.1p Priority to IPv4 Packets on the Basis of Incoming DSCP

One of the best uses for this option is on an interior switch where you want to honor (continue) a policy set on an edge switch. That is, it enables you to select incoming packets having a specific DSCP and forward these packets with the desired 802.1p priority. For example, if an edge switch "A" marks all packets received on port A5 with a particular DSCP, you can configure a downstream (interior) switch "B" to handle such packets with the desired priority (regardless of whether 802.1Q tagged VLANs are in use).

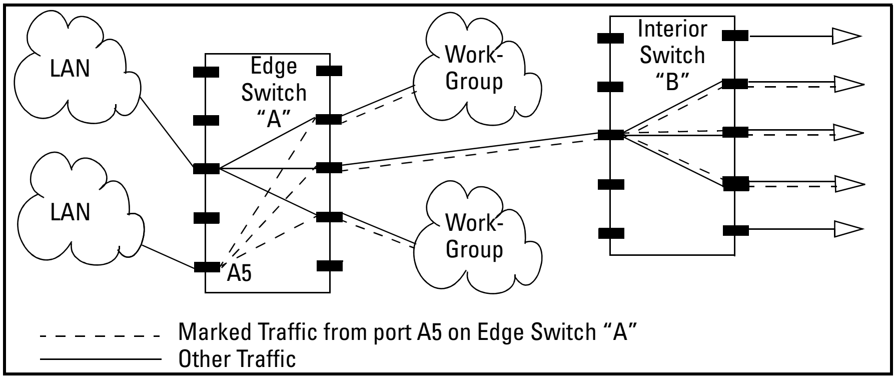


Figure 6-14. Interior Switch "B" Honors the Policy Established in Edge Switch "A"

To do so, assign the desired 802.1p priority to the same codepoint that the upstream or edge switch assigns to the selected packets. When the downstream switch receives an IPv4 packet carrying one of these codepoints, it assigns the configured priority to the packet and sends it out the appropriate priority queue. (The packet retains the codepoint it received from the upstream or edge switch). You can use this option concurrently with the diffserv DSCP Policy option (described later in this section), as long as the DSCPs specified in the two options do not match.

Note on DSCP Use

Different applications may use the same DSCP in their IP packets. Also, the same application may use multiple DSCPs if the application originates on different clients, servers, or other devices. Using an edge switch enables you to select the packets you want and mark them with predictable DSCPs that can be used by downstream switches to honor policies set in the edge switch.

When enabled, the switch applies direct 802.1p prioritization to all packets having codepoints that meet these criteria:

- The codepoint is configured with an 802.1p priority in the DSCP table. (Codepoints configured with **No-override** are not used.)
- The codepoint is not configured for a new DSCP policy assignment.

Thus, the switch does not allow the same incoming codepoint (DSCP) to be used simultaneously for directly assigning an 802.1p priority and also assigning a DSCP policy. For a given incoming codepoint, if you configure one option and then the other, the second overwrites the first.

To use this option:

1. Identify a DSCP used to set a policy in packets received from an upstream or edge switch.
2. Determine the 802.1p priority (0 - 7) you want to apply to packets carrying the identified DSCP. (You can either maintain the priority assigned in the upstream or edge switch, or assign a new priority.)
3. Use **qos dscp-map <** ***codepoint*** **> priority < 0 - 7 >** to assign the 802.1p priority you want to the specified DSCP. (For more on this topic, refer to "Differentiated Services Codepoint (DSCP) Mapping" on page 6-58.)
4. Enable **diff-services.**

.

Syntax: qos type-of-service diff-services < *codepoint*>

Causes the switch to read the ***< codepoint >*** *(DSCP) of an incoming IPv4 packet and, when a match occurs, assign a corresponding 802.1p priority, as configured in the switch's DSCP table (page 6-59).*

no qos type-of-service

Disables all ToS type operation.

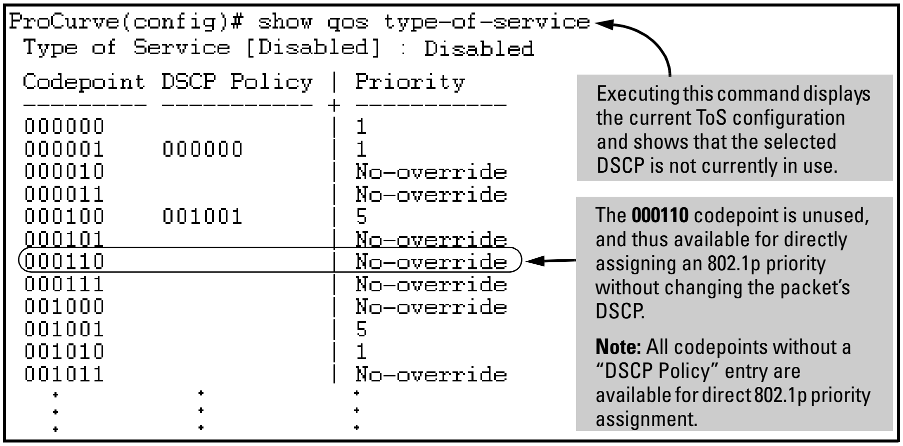
no qos dscp-map < *codepoint*>

Disables direct 802.1p priority assignment to packets carrying the < ***codepoint*** *> by reconfiguring the codepoint priority assignment in the DSCP table to* **No-override**. *Note that if this codepoint is in use as a DSCP policy for another diffserv codepoint, you must disable or redirect the other diffserv codepoint's DSCP policy before you can disable or change the codepoint. For example, in figure 6-15 you cannot change the priority for the 000000 codepoint until you redirect the DSCP policy for 000001 away from using 000000 as a policy. (Refer to "Note On Changing a Priority Setting" on page 6-61. Refer also to "Differentiated Services Codepoint (DSCP) Mapping" on page 6-58.)*

show qos type-of-service

Displays current Type-of-Service configuration. In diffserv mode it also shows the current direct 802.1p assignments and the current DSCP assignments covered later in this section.

For example, an edge switch "A" in an untagged VLAN assigns a DSCP of 000110 on IP packets it receives on port A6, and handles the packets with high priority (7). When these packets reach interior switch "B" you want the switch to handle them with the same high priority. To enable this operation you would configure an 802.1p priority of 7 for packets received with a DSCP of **000110**, and then enable **diff-services**:



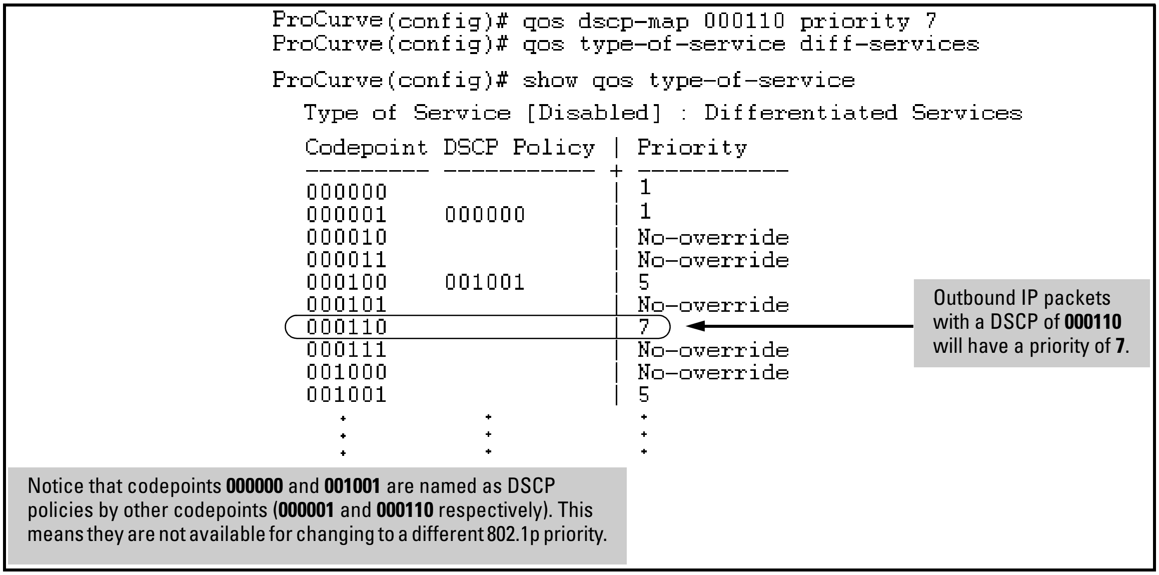
```
ProCurve(config)# show qos type-of-service
 Type of Service [Disabled] : Disabled

 Codepoint DSCP Policy | Priority
 --------- ----------- + -----------
 000000                | 1
 000001    000000      | 1
 000010                | No-override
 000011                | No-override
 000100    001001      | 5
 000101                | No-override
 000110                | No-override
 000111                | No-override
 001000                | No-override
 001001                | 5
 001010                | 1
 001011                | No-override
   .          .          .
   .          .          .
   .          .          .
```

Executing this command displays the current ToS configuration and shows that the selected DSCP is not currently in use.

The **000110** codepoint is unused, and thus available for directly assigning an 802.1p priority without changing the packet's DSCP.

Note: All codepoints without a "DSCP Policy" entry are available for direct 802.1p priority assignment.

Figure 6-15. Example Showing Codepoints Available for Direct 802.1p Priority Assignments



```
ProCurve(config)# qos dscp-map 000110 priority 7
ProCurve(config)# qos type-of-service diff-services

ProCurve(config)# show qos type-of-service

  Type of Service [Disabled] : Differentiated Services

  Codepoint DSCP Policy | Priority
  --------- ----------- + -----------
  000000                | 1
  000001    000000      | 1
  000010                | No-override
  000011                | No-override
  000100    001001      | 5
  000101                | No-override
  000110                | 7
  000111                | No-override
  001000                | No-override
  001001                | 5
     .         .           .
     .         .           .
     .         .           .
```

Outbound IP packets with a DSCP of **000110** will have a priority of **7**.

Notice that codepoints **000000** and **001001** are named as DSCP policies by other codepoints (**000001** and **000110** respectively). This means they are not available for changing to a different 802.1p priority.

Figure 6-16. Example of a Type-of-Service Configuration Enabling Both Direct 802.1p Priority Assignment and DSCP Policy Assignment

Assigning a DSCP Policy on the Basis of the DSCP in IPv4 Packets Received from Upstream Devices

The preceding section describes how to forward a policy set by an edge (or upstream) switch. This option changes a DSCP policy in an IPv4 packet by changing its IP ToS codepoint and applying the priority associated with the new codepoint. (A DSCP policy consists of a differentiated services codepoint and an associated 802.1p priority.) You can use this option concurrently with the diffserv 802.1p priority option (above), as long as the DSCPs specified in the two options do not match.

To use this option to configure a change in policy:

1. Identify a DSCP used to set a policy in packets received from an upstream or edge switch.
2. Create a new policy by using **qos dscp-map <** ***codepoint*** **> priority < 0 - 7 >** to configure an 802.1p priority for the codepoint you will use to overwrite the DSCP the packet carries from upstream. (For more on this topic, refer to “Differentiated Services Codepoint (DSCP) Mapping” on page 6-58.)

3. Use **qos type-of-service diff-services <** ***incoming-DSCP*** **> dscp <** ***outgoing-DSCP*** **>** to change the policy on packets coming from the edge or upstream switch with the specified incoming DSCP.

   (Figure 6-14 on page 6-35 illustrates this scenario.)

Notes

"Mixing" ToS DSCP policies and 802.1p priorities is not recommended. Refer to the Note on page 6-10.

Syntax: qos type-of-service diff-services

Enables ToS diff-services.

qos type-of-service diff-services < *current-codepoint* > dscp < *new-codepoint* >

*Configures the switch to select an incoming IP packet carrying the <****current-codepoint****> and then use the <****new-codepoint****> to assign a new, previously configured DSCP policy to the packet. The policy overwrites the <****current-codepoint****> with the < ****new-codepoint****> and assigns the 802.1p priority specified by the policy. (Use the* **qos dscp-map** *command to define the priority for the DSCPs—page 6-58.)*

no qos type-of-service

Disables all ToS type operation. Current ToS DSCP policies and priorities remain in the configuration and will become available if you re-enable ToS diff-services.

no qos type-of-service [diff-services < *codepoint* >]

*Deletes the DSCP policy assigned to the < ****codepoint**** > and returns the < ****codepoint**** > to the 802.1p priority setting it had before the DSCP policy was assigned. (This will be either a value from 0 - 7 or* **No-override***.)*

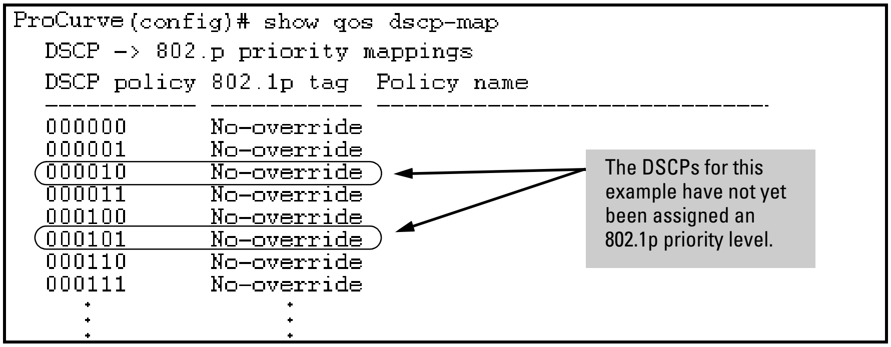
show qos type-of-service

Displays a listing of codepoints, with any corresponding DSCP policy re-assignments for outbound packets. Also lists the (802.1p) priority for each codepoint that does not have a DSCP policy assigned to it.

For example, suppose you want to configure the following two DSCP policies for packets received with the indicated DSCPs.

Received DSCP	Policy DSCP	802.1p Priority	Policy Name (Optional)
001100	000010	6	Level 6
001101	000101	4	Level 4

1. Determine whether the DSCPs already have priority assignments, which could indicate use by existing applications. This is not a problem as long as the configured priorities are acceptable for all applications using the same DSCP. (Refer to the “Note On Changing a Priority Setting” on page 6-61. Also, a DSCP must have a priority configured before you can assign any QoS types to use it.)

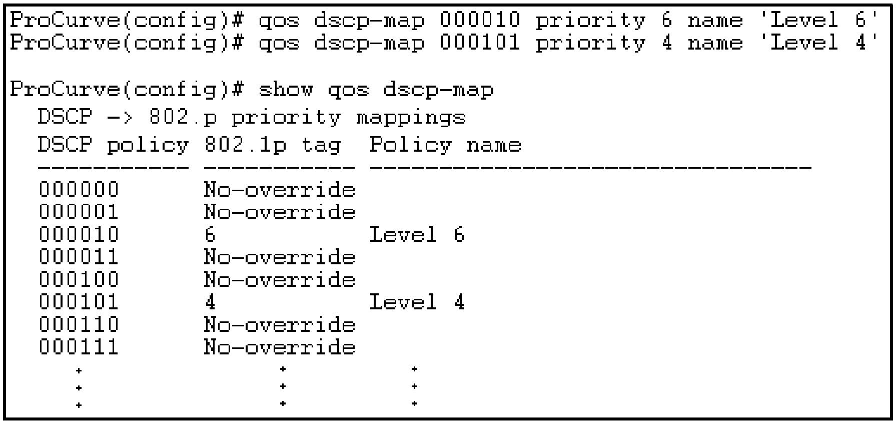


```
ProCurve(config)# show qos dscp-map
  DSCP -> 802.p priority mappings
  DSCP policy 802.1p tag  Policy name
  ----------- ----------- -----------------------------
  000000      No-override
  000001      No-override
  000010      No-override
  000011      No-override
  000100      No-override
  000101      No-override
  000110      No-override
  000111      No-override
     :           :
```

The DSCPs for this example have not yet been assigned an 802.1p priority level.

Figure 6-17. Display the Current DSCP-Map Configuration

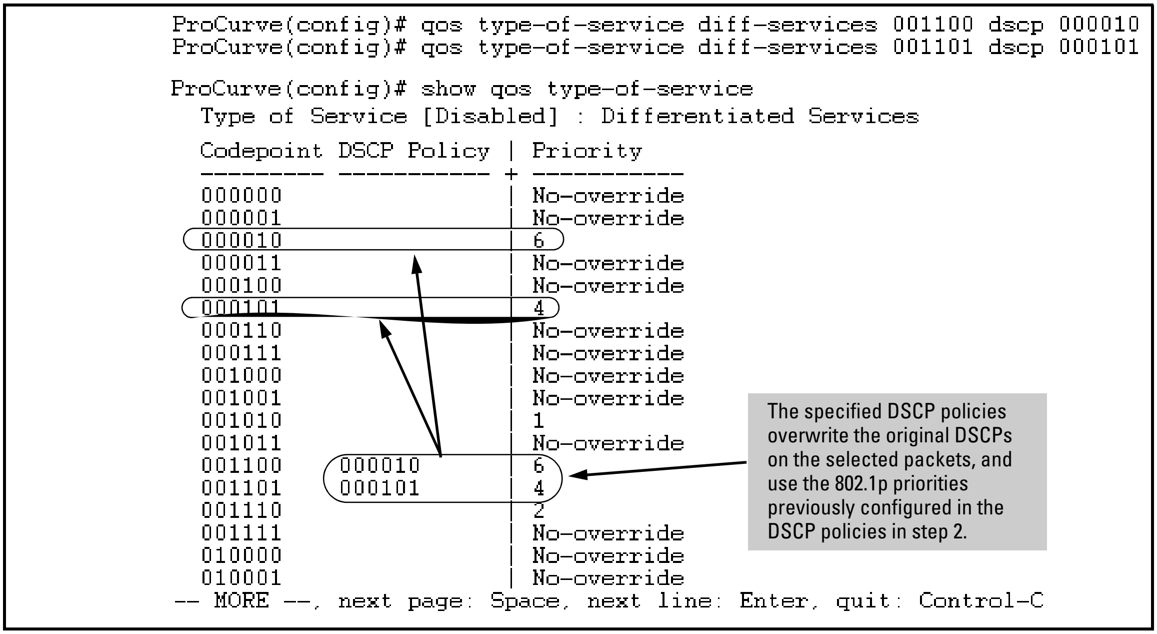
2. Configure the policies in the DSCP table:



```
ProCurve(config)# qos dscp-map 000010 priority 6 name 'Level 6'
ProCurve(config)# qos dscp-map 000101 priority 4 name 'Level 4'

ProCurve(config)# show qos dscp-map
  DSCP -> 802.p priority mappings
  DSCP policy 802.1p tag  Policy name
  ----------- ----------- -------------------------------
  000000      No-override
  000001      No-override
  000010      6           Level 6
  000011      No-override
  000100      No-override
  000101      4           Level 4
  000110      No-override
  000111      No-override
     .              .         .
     .              .         .
     .              .         .
```

Figure 6-18. Example of Policies Configured (with Optional Names) in the DSCP Table

3. Assign the policies to the codepoints in the selected packet types.



```
ProCurve(config)# qos type-of-service diff-services 001100 dscp 000010
ProCurve(config)# qos type-of-service diff-services 001101 dscp 000101

ProCurve(config)# show qos type-of-service
  Type of Service [Disabled] : Differentiated Services

  Codepoint DSCP Policy | Priority
  --------- ----------- + -----------
  000000                | No-override
  000001                | No-override
  000010                | 6
  000011                | No-override
  000100                | No-override
  000101                | 4
  000110                | No-override
  000111                | No-override
  001000                | No-override
  001001                | No-override
  001010                | 1
  001011                | No-override
  001100    000010      | 6
  001101    000101      | 4
  001110                | 2
  001111                | No-override
  010000                | No-override
  010001                | No-override
-- MORE --, next page: Space, next line: Enter, quit: Control-C
```

The specified DSCP policies overwrite the original DSCPs on the selected packets, and use the 802.1p priorities previously configured in the DSCP policies in step 2.

Figure 6-19. Example of Policy Assignment to Outbound Packets on the Basis of the DSCP in the Packets Received from Upstream Devices

Details of QoS IP Type-of-Service

IP packets include a Type of Service (ToS) byte. The ToS byte includes:

- **A Differentiated Services Codepoint (DSCP):** This element is comprised of the upper six bits of the ToS byte). There are 64 possible codepoints. In the switches covered by this manual, the default **qos** configuration includes the codepoint having the 802.1p priority setting for Expedited Forwarding, while all others, including the Assured-Forwarding codepoints, are unused (and listed with **No-override** for a Priority).

  Refer to figure 6-11 on page 6-59 for an illustration of the default DSCP policy table.

  Using the **qos dscp map** command, you can configure the switch to assign different prioritization policies to IPv4 packets having different codepoints. As an alternative, you can configure the switch to assign a new codepoint to an IPv4 packet, along with a corresponding 802.1p priority (0-7). To use this option in the simplest case, you would:

  a. Configure a specific DSCP with a specific priority in an edge switch.

  b. Configure the switch to mark a specific type of inbound traffic with that DSCP (and thus create a policy for that traffic type).

  c. Configure the internal switches in your LAN to honor the policy.

  (For example, you could configure an edge switch to assign a codepoint of 000001 to all packets received from a specific VLAN, and then handle all traffic with that codepoint at high priority.)

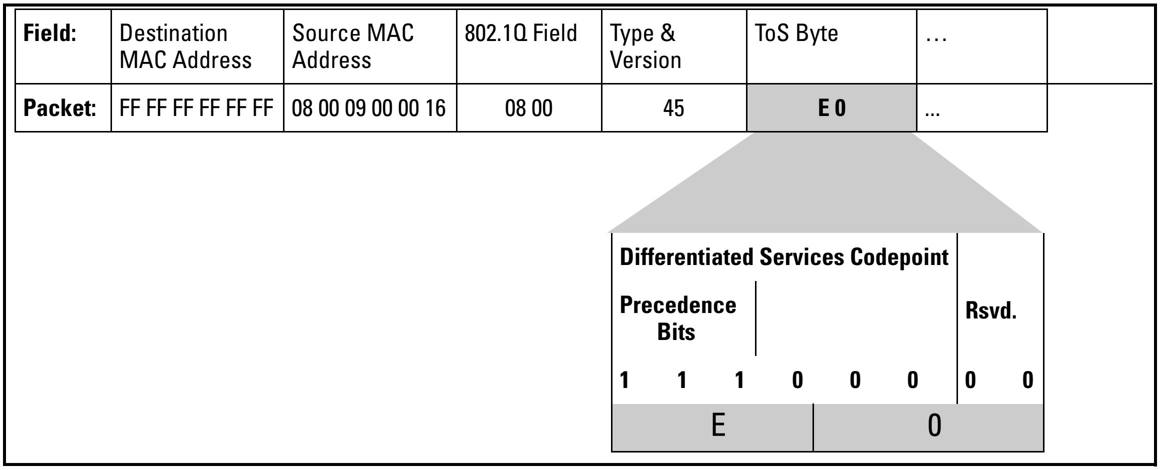
  For a codepoint listing and the commands for displaying and changing the DSCP Policy table, refer to “Differentiated Services Codepoint (DSCP) Mapping” on page 6-58.

Notes

“Mixing” ToS DSCP policies and 802.1p priorities is not recommended. Refer to the Note on page 6-10.

- **Precedence Bits:** This element is a subset of the DSCP and is comprised of the upper three bits of the ToS byte. When configured to do so, the switch uses the precedence bits to determine a priority for handling the associated packet. (The switch does not change the setting of the precedence bits.) Using the ToS Precedence bits to prioritize IPv4 packets relies on priorities set in upstream devices and applications.

Figure 6-20 shows an example of the ToS byte in the header for an IPv4 packet, and illustrates the diffserv bits and precedence bits in the ToS byte. (Note that the Precedence bits are a subset of the Differentiated Services bits.)



Field:	Destination MAC Address	Source MAC Address	802.1Q Field	Type & Version	ToS Byte	. . .
Packet:	FF FF FF FF FF FF	08 00 09 00 00 16	08 00	45	**E 0**	...

Differentiated Services Codepoint							
Precedence Bits						**Rsvd.**	
1	**1**	**1**	**0**	**0**	**0**	**0**	**0**
E				0			

Figure 6-20. The ToS Codepoint and Precedence Bits

Table 6-9. How the Switch Uses the ToS Configuration

Outbound Port	**ToS Option:**	
	IP Precedence (Value = 0 - 7)	**Differentiated Services**
IP Packet Sent Out an Untagged Port in a VLAN	Depending on the value of the IP Precedence bits in the packet's ToS field, the packet will go to one of four outbound port queues in the switch: 1 - 2 = low priority 0 - 3 = normal priority 4 - 5 = high priority 6 - 7 = high priority	For a given packet carrying a ToS codepoint that the switch has been configured to detect: • Change the codepoint according to the configured policy and assign the 802.1p priority specified for the new codepoint in the DSCP Policy Table (page 6-58). • Do not change the codepoint, but assign the 802.1p priority specified for the existing codepoint in the DSCP Policy Table (page 6-58). Depending on the 802.1p priority used, the packet will leave the switch through one of the following queues: 1 - 2 = low priority 0 - 3 = normal priority 4 - 5 = high priority 6 - 7 = high priority If **No-override** (the default) has been configured for a specified codepoint, then the packet is not prioritized by ToS and, by default, is sent to the "normal priority" queue.

Outbound Port	ToS Option:	
	IP Precedence (Value = 0 - 7)	Differentiated Services
IP Packet Sent Out an Untagged Port in a VLAN	Same as above, plus the IP Precedence value (0 - 7) will be used to set a corresponding 802.1p priority in the VLAN tag carried by the packet to the next downstream device. Refer to table 6-10, below.	Same as above, plus the Priority value (0 - 7) will be used to set a corresponding 802.1p priority in the VLAN tag carried by the packet to the next downstream device. Where **No-override** is the assigned priority, the VLAN tag carries a "0" (normal priority) 802.1p setting if not prioritized by other QoS types.

Table 6-10. ToS IP-Precedence Bit Mappings to 802.1p Priorities

ToS Byte IP Precedence Bits	Corresponding 802.1p Priority	Service Priority Level
000	1	Lowest
001	2	Low
002	0	Normal
003	3	
004	4	
005	5	
006	6	
007	7	Highest

QoS VLAN-ID (VID) Priority

QoS Type Precedence: 5

The VLAN QoS type allows you to configure one rule for each VLAN, up to a maximum of 120 VLANs.

Where a particular VLAN-ID type has the highest precedence in the switch for traffic in that VLAN, then traffic received in that VLAN is marked with the VID type's configured priority level.

Options for Assigning Priority. Priority control options for packets carrying a specified VLAN-ID include:

- 802.1p priority
- DSCP policy (Assigning a new DSCP and an associated 802.1p priority; inbound packets must be IPv4.)

(For operation when other QoS types apply to the same traffic, refer to "QoS Types for Prioritizing Outbound Packets" on page 6-9.)

Note

QoS with VID priority applies to static VLANs only, and applying QoS to dynamic VLANs created by GVRP operation is not supported. A VLAN must exist while a subject of a QoS configuration, and eliminating a VLAN from the switch causes the switch to clear any QoS features configured for that VID.

Assigning a Priority Based on VLAN-ID

This option assigns a priority to all outbound packets having the specified VLAN-ID (VID). You can configure this option by either specifying the VLAN-ID ahead of the **qos** command or moving to the VLAN context for the VLAN you want to configure for priority.

Syntax: vlan < *vid* > qos priority < 0 - 7 >

Configures an 802.1p priority for outbound packets belonging to the specified VLAN. This priority determines the packet's queue in the outbound port to which it is sent. If the packet leaves the switch on a tagged port, it carries the 802.1p priority with it to the next downstream device. You can configure one QoS type for each VLAN-ID. (Default: **No-override***)*

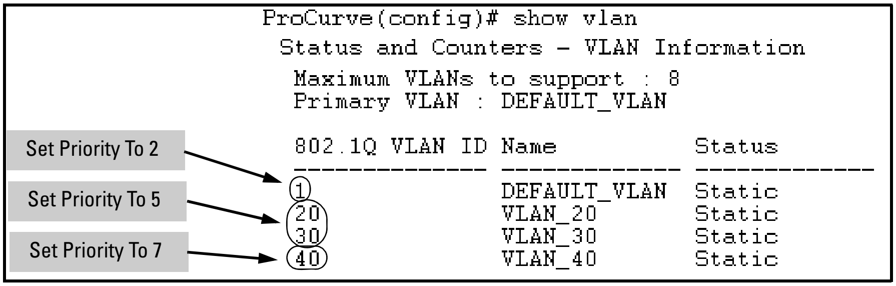
no vlan < *vid* > qos

Removes the specified VLAN-ID as a QoS type and resets the priority for that VLAN to **No-override**.

show qos vlan-priority

Displays a listing of the QoS VLAN-ID types currently in the running-config file, with their priority data.

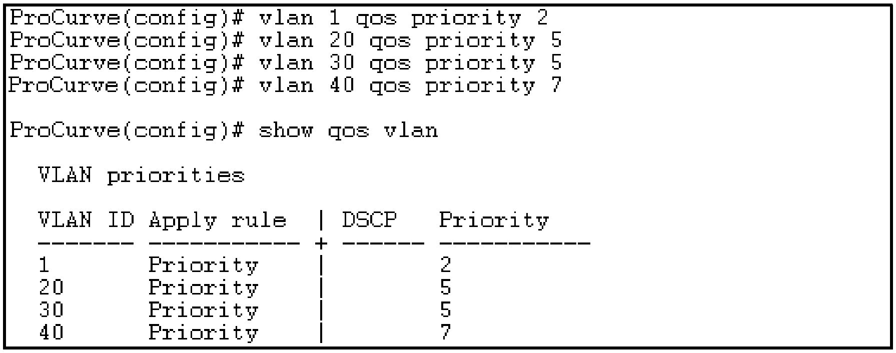
1. For example, suppose that you have the following VLANs configured on the switch and want to prioritize them as shown:



```
ProCurve(config)# show vlan
 Status and Counters - VLAN Information
  Maximum VLANs to support : 8
  Primary VLAN : DEFAULT_VLAN

  802.1Q VLAN ID Name          Status
  -------------- ------------- -------------
  1              DEFAULT_VLAN  Static
  20             VLAN_20       Static
  30             VLAN_30       Static
  40             VLAN_40       Static
```

Figure 6-21. Example of a List of VLANs Available for QoS Prioritization

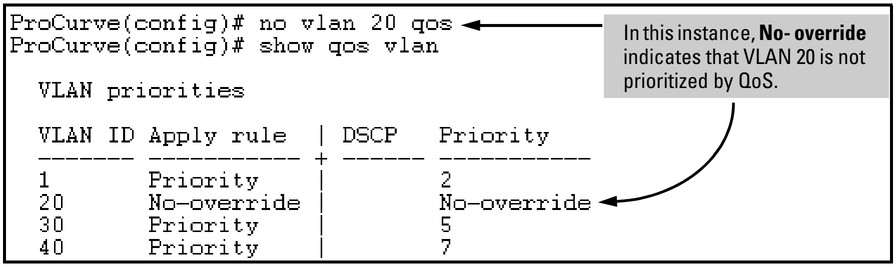
2. You would then execute the following commands to prioritize the VLANs by VID:



```
ProCurve(config)# vlan 1 qos priority 2
ProCurve(config)# vlan 20 qos priority 5
ProCurve(config)# vlan 30 qos priority 5
ProCurve(config)# vlan 40 qos priority 7

ProCurve(config)# show qos vlan

  VLAN priorities

  VLAN ID Apply rule  | DSCP   Priority
  ------- ----------- + ------ -----------
  1       Priority    |        2
  20      Priority    |        5
  30      Priority    |        5
  40      Priority    |        7
```

Figure 6-22. Configuring and Displaying QoS Priorities on VLANs

If you then decided to remove VLAN_20 from QoS prioritization:



```
ProCurve(config)# no vlan 20 qos
ProCurve(config)# show qos vlan

  VLAN priorities

  VLAN ID Apply rule  | DSCP   Priority
  ------- ----------- + ------ -----------
  1       Priority    |        2
  20      No-override |        No-override
  30      Priority    |        5
  40      Priority    |        7
```

In this instance, **No- override** indicates that VLAN 20 is not prioritized by QoS.

Figure 6-23. Returning a QoS-Prioritized VLAN to "No-override" Status

Assigning a DSCP Policy Based on VLAN-ID (VID)

This option assigns a previously configured DSCP policy (codepoint and 802.1p priority) to outbound IP packets having the specified VLAN-ID (VID). That is, the switch:

1. Selects an incoming IP packet on the basis of the VLAN-ID it carries.
2. Overwrites the packet's DSCP with the DSCP configured in the switch for such packets.
3. Assigns the 802.1p priority configured in the switch for the new DSCP. (Refer to "Differentiated Services Codepoint (DSCP) Mapping" on page 6-58.)
4. Forwards the packet through the appropriate outbound port queue.

Note

"Mixing" ToS DSCP policies and 802.1p priorities is not recommended. Refer to the Note on page 6-10.

For more on DSCP, refer to "Terminology" on page 6-6.

Steps for Creating a Policy Based on VLAN-ID Type.

1. Determine the VLAN-ID type to which you want to assign a DSCP policy.
2. Determine the DSCP policy for packets carrying the selected VLAN-ID:
   a. Determine the DSCP you want to assign to the selected packets. (This codepoint will be used to overwrite the DSCP carried in packets received from upstream devices.)
   b. Determine the 802.1p priority you want to assign to the DSCP.
3. Configure the DSCP policy by using **qos dscp-map** to configure the priority for each codepoint. (For details, see the example later in this section, and to "Differentiated Services Codepoint (DSCP) Mapping" on page 6-58.)

Note

A codepoint must have an 802.1p priority (0 - 7) before you can configure the codepoint for use in prioritizing packets by VLAN-ID. If a codepoint you want to use shows **No-override** in the **Priority** column of the DSCP Policy table (**show qos dscp-map**), then assign a priority before proceeding.

4. Configure the switch to assign the DSCP policy to packets with the specified VLAN-ID.

Syntax: qos dscp-map < *codepoint* > priority < 0 - 7 >

This command is optional if a priority has already been assigned to the < ***codepoint*** *>. The command creates a DSCP policy by assigning an 802.1p priority to a specific DSCP. When the switch applies this priority to a packet, the priority determines the packet's queue in the outbound port to which it is sent. If the packet leaves the switch on a tagged port, it carries the 802.1p priority with it to the next downstream device. If the packet is IPv4, the packet's DSCP will be replaced by the codepoint specified in this command. (Default: For most codepoints,* **No-override**. *See figure 6-11 on page 6-59 on page 6-59.)*

Syntax: vlan < *vid* > qos dscp < *codepoint* >

> *Assigns a DSCP policy to packets carrying the specified IP address, and overwrites the DSCP in these packets with the assigned <* ***codepoint*** *> value. This policy includes an 802.1p priority and determines the packet's queue in the outbound port to which it is sent. If the packet leaves the switch on a tagged port, it carries the 802.1p priority with it to the next downstream device. (Default:* **No-override***)*

no vlan < *vid* > qos

> *Removes QoS type for the specified VLAN.*

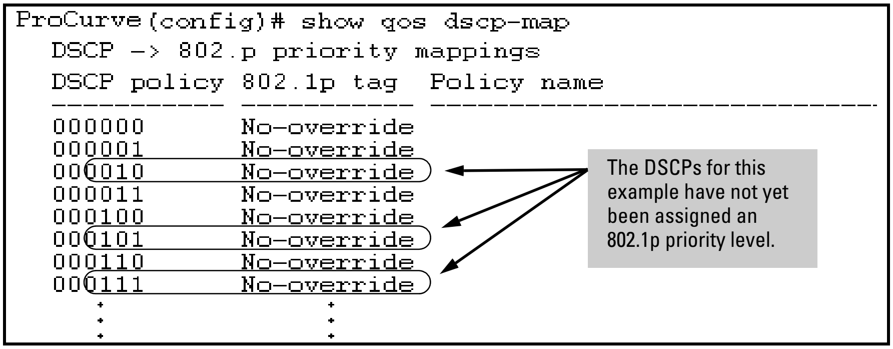
show qos vlan-priority

> *Displays a listing of all QoS VLAN-ID types currently in the running-config file.*

For example, suppose you wanted to assign this set of priorities:

VLAN-ID	DSCP	Priority
40	000111	7
30	000101	5
20	000010	1
1	000010	1

1. Determine whether the DSCPs already have priority assignments, which could indicate use by existing applications. This is not a problem as long as the configured priorities are acceptable for all applications using the same DSCP. (Refer to the "Note On Changing a Priority Setting" on page 6-61. Also, a DSCP must have a priority configured before you can assign any QoS types to use it.)

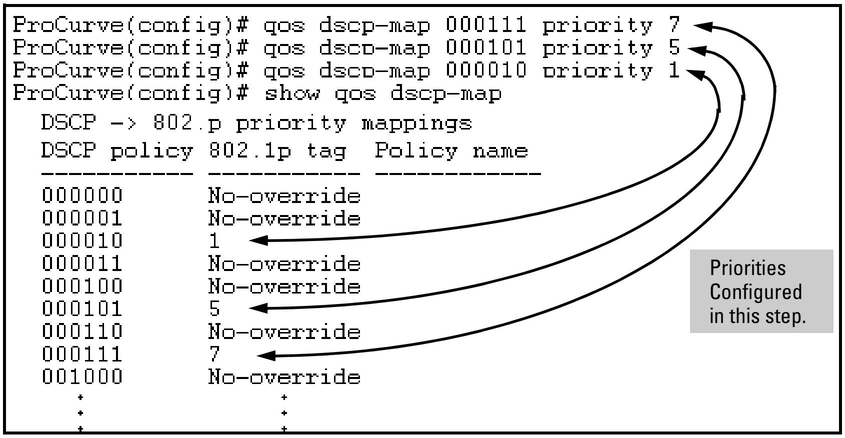


```
ProCurve(config)# show qos dscp-map
  DSCP -> 802.p priority mappings
  DSCP policy 802.1p tag  Policy name
  ----------- ----------- -------------------------------
  000000      No-override
  000001      No-override
  000010      No-override
  000011      No-override
  000100      No-override
  000101      No-override
  000110      No-override
  000111      No-override
     .             .
     .             .
     .             .
```

The DSCPs for this example have not yet been assigned an 802.1p priority level.

Figure 6-24. Display the Current Configuration in the DSCP Policy Table

2. Configure the priorities for the DSCPs you want to use.

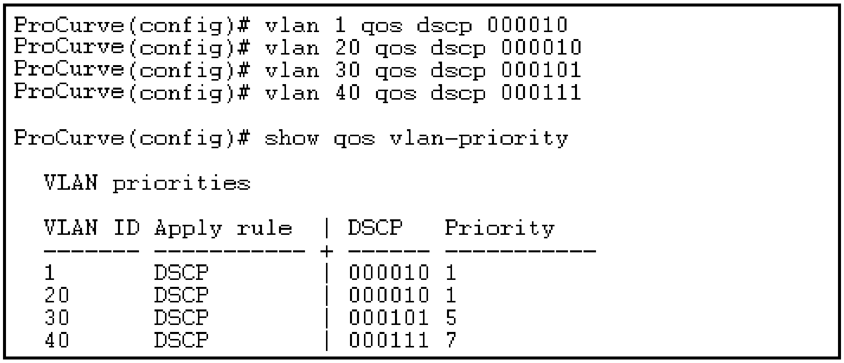


```
ProCurve(config)# qos dscp-map 000111 priority 7
ProCurve(config)# qos dscp-map 000101 priority 5
ProCurve(config)# qos dscp-map 000010 priority 1
ProCurve(config)# show qos dscp-map
  DSCP -> 802.p priority mappings
  DSCP policy 802.1p tag  Policy name
  ----------- ----------- ------------
  000000      No-override
  000001      No-override
  000010      1
  000011      No-override
  000100      No-override
  000101      5
  000110      No-override
  000111      7
  001000      No-override
     .              .
     .              .
     .              .
```

Priorities Configured in this step.

Figure 6-25. Assign Priorities to the Selected DSCPs

Assign the DSCP policies to the selected VIDs and display the result.



```
ProCurve(config)# vlan 1 qos dscp 000010
ProCurve(config)# vlan 20 qos dscp 000010
ProCurve(config)# vlan 30 qos dscp 000101
ProCurve(config)# vlan 40 qos dscp 000111

ProCurve(config)# show qos vlan-priority

  VLAN priorities

  VLAN ID Apply rule  | DSCP   Priority
  ------- ----------- + ------ -----------
  1       DSCP        | 000010 1
  20      DSCP        | 000010 1
  30      DSCP        | 000101 5
  40      DSCP        | 000111 7
```

Figure 6-26. The Completed VID-DSCP Priority Configuration

The switch will now apply the DSCP policies in figure 6-26 to packets received on the switch with the specified VLAN-IDs. This means the switch will:

- Overwrite the original DSCPs in the selected packets with the new DSCPs specified in the above policies.
- Assign the 802.1p priorities in the above policies to the appropriate packets.

QoS Interface (Source-Port) Priority

QoS Type Precedence: 6

The QoS Interface option enables you to use a packet's source-port on the switch as a QoS type. Where a particular source-port type has the highest precedence in the switch for traffic entering through that port, then traffic received from the port is marked with the source-port type's configured priority level.

Options for Assigning Priority. Priority control options for packets from a specified source-port include:

- 802.1p priority
- DSCP policy (Assigning a new DSCP and an associated 802.1p priority; inbound packets must be IPv4.)

(For operation when other QoS types apply to the same traffic, refer to "QoS Types for Prioritizing Outbound Packets" on page 6-9.)

Assigning a Priority Based on Source-Port

This option assigns a priority to outbound packets having the specified source-port. Configure this option by either specifying the source-port ahead of the **qos** command or moving to the port context for the port you want to configure for priority. (For configuring multiple source-ports with the same priority, you may find it easier to use the **interface <** ***port-list*** **>** command to go to the port context instead of individually configuring the priority for each port.)

Syntax: interface < *port-list* > qos priority < 0 - 7 >

Configures an 802.1p priority for packets entering the switch through the specified (source) ports. This priority determines the packet queue in the outbound port(s) to which traffic is sent. If a packet leaves the switch on a tagged port, it carries the 802.1p priority with it to the next downstream device. You can configure one QoS type for each source-port or group of source-ports. (Default: **No-override***)*

no interface < *port-list* > qos

Disables use of the specified source-port(s) for QoS type(s) and resets the priority for the specified source-port(s) to **No-override**.

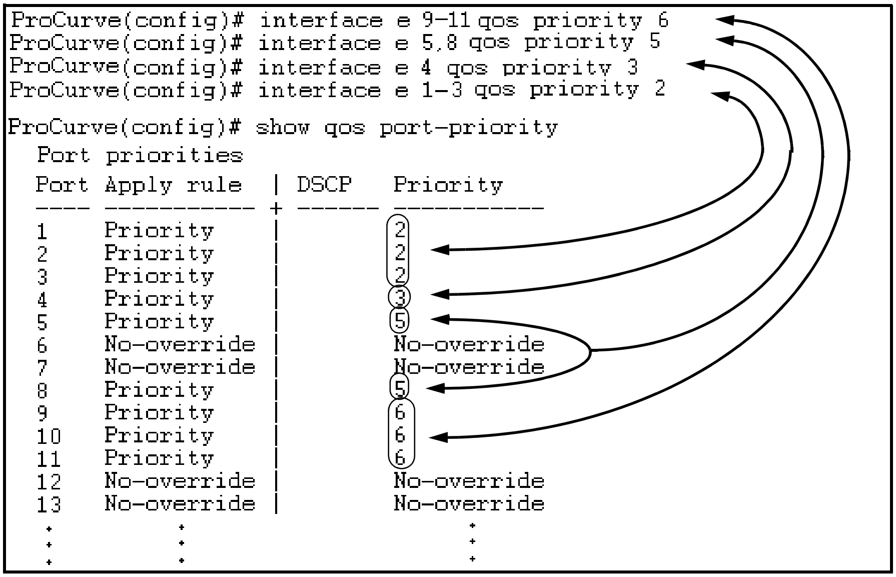
show qos port-priority

Lists the QoS port-priority types with their priority data.

For example, suppose that you want to prioritize inbound traffic on the following source-ports:

Interface (Source Port)	Priority
1 - 3	2
4	3
5, 8	5
9 - 11	6

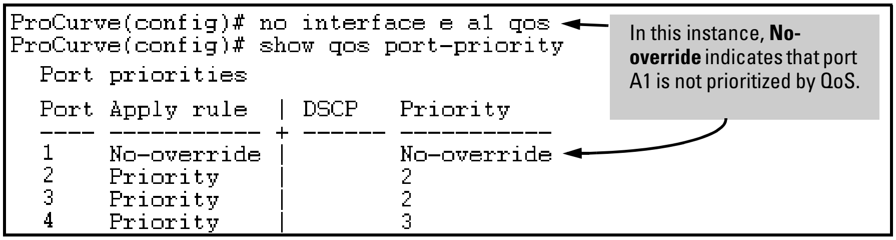
You would then execute the following commands to prioritize traffic received on the above ports:



```
ProCurve(config)# interface e 9-11 qos priority 6
ProCurve(config)# interface e 5,8 qos priority 5
ProCurve(config)# interface e 4 qos priority 3
ProCurve(config)# interface e 1-3 qos priority 2

ProCurve(config)# show qos port-priority
  Port priorities
  Port Apply rule  | DSCP   Priority
  ---- ----------- + ------ -----------
  1    Priority    |        2
  2    Priority    |        2
  3    Priority    |        2
  4    Priority    |        3
  5    Priority    |        5
  6    No-override |        No-override
  7    No-override |        No-override
  8    Priority    |        5
  9    Priority    |        6
  10   Priority    |        6
  11   Priority    |        6
  12   No-override |        No-override
  13   No-override |        No-override
  .         .                     .
  .         .                     .
  .         .                     .
```

Figure 6-27. Configuring and Displaying Source-Port QoS Priorities

If you then decided to remove port 1 from QoS prioritization:



```
ProCurve(config)# no interface e a1 qos
ProCurve(config)# show qos port-priority
  Port priorities
  Port Apply rule  | DSCP   Priority
  ---- ----------- + ------ -----------
  1    No-override |        No-override
  2    Priority    |        2
  3    Priority    |        2
  4    Priority    |        3
```

In this instance, **No-override** indicates that port A1 is not prioritized by QoS.

Figure 6-28. Returning a QoS-Prioritized VLAN to "No-override" Status

Assigning a DSCP Policy Based on the Source-Port

This option assigns a previously configured DSCP policy (codepoint and 802.1p priority) to outbound IP packets (received from the specified source-ports). That is, the switch:

1. Selects an incoming IP packet on the basis of its source-port on the switch.
2. Overwrites the packet's DSCP with the DSCP configured in the switch for such packets.
3. Assigns the 802.1p priority configured in the switch for the new DSCP. (Refer to "Differentiated Services Codepoint (DSCP) Mapping" on page 6-58.)
4. Forwards the packet through the appropriate outbound port queue.

Note

"Mixing" ToS DSCP policies and 802.1p priorities is not recommended. Refer to the Note on page 6-10.

For more on DSCP, refer to "Terminology" on page 6-6.

Steps for Creating a Policy Based on Source-Port Types.

Note

You can select one DSCP per source-port. Also, configuring a new DSCP for a source-port automatically overwrites (replaces) any previous DSCP or 802.1p priority configuration for that port.)

1. Identify the source-port type to which you want to assign a DSCP policy.
2. Determine the DSCP policy for packets having the selected source-port:
   a. Determine the DSCP you want to assign to the selected packets. (This codepoint will be used to overwrite the DSCP carried in packets received through the source-port from upstream devices.)
   b. Determine the 802.1p priority you want to assign to the DSCP.
3. Configure the DSCP policy by using **qos dscp-map** to configure the priority for each codepoint. (For details, refer to the example later in this section and to "Differentiated Services Codepoint (DSCP) Mapping" on page 6-58.)

Note

A codepoint must have an 802.1p priority assignment (0 - 7) before you can configure that codepoint as a criteria for prioritizing packets by source-port. If a codepoint shows **No-override** in the **Priority** column of the DSCP Policy Table (**show qos dscp-map**), then you must assign a 0 - 7 priority before proceeding.

4. Configure the switch to assign the DSCP policy to packets from the specified source-port.

Syntax: qos dscp-map < *codepoint* > priority < 0 - 7 >

This command is optional if a priority has already been assigned to the < **codepoint** *>. The command creates a DSCP policy by assigning an 802.1p priority to a specific DSCP. When the switch applies this priority to a packet, the priority determines the packet's queue in the outbound port to which it is sent. If the packet leaves the switch on a tagged port, it carries the 802.1p priority with it to the next downstream device. (Default: For most codepoints,* **No-override***. See figure 6-11 on page 6-59 on page 6-59.)*

Syntax: interface < *port-list* > qos dscp < *codepoint* >

Assigns a DSCP policy to packets from the specified source-port(s), and overwrites the DSCP in these packets with the assigned < **codepoint** *> value. This policy includes an 802.1p priority and determines the packet's queue in the outbound port to which it is sent. If the packet leaves the switch on a tagged port, it carries the 802.1p priority with it to the next downstream device. (Default:* **No-override***)*

no interface < *port-list* > qos

Removes QoS type for the specified source-port(s).

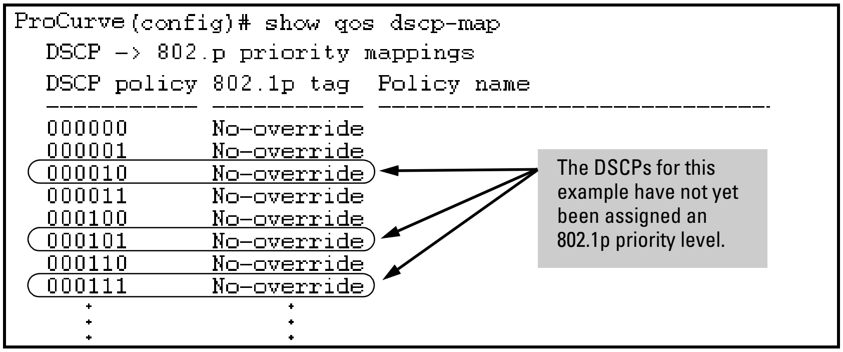
show qos port-priority

Displays a listing of all source-port QoS types currently in the running-config file.

For example, suppose you wanted to assign this set of priorities:

Source-Port	DSCP	Priority
2	000111	7
5 - 7	000101	5
8, 10	000010	1

1. Determine whether the DSCPs already have priority assignments, which could indicate use by existing applications. This is not a problem as long as the configured priorities are acceptable for all applications using the same DSCP. (Refer to the “Note On Changing a Priority Setting” on page 6-61. Also, a DSCP must have a priority configured before you can assign any QoS types to use it.)

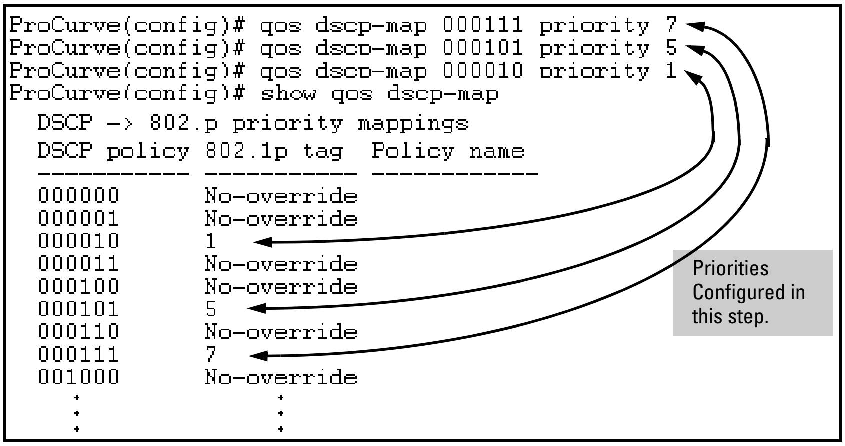


```
ProCurve(config)# show qos dscp-map
  DSCP -> 802.p priority mappings
  DSCP policy 802.1p tag  Policy name
  ----------- ----------- ------------------------------
  000000      No-override
  000001      No-override
  000010      No-override
  000011      No-override
  000100      No-override
  000101      No-override
  000110      No-override
  000111      No-override
     .            .
     .            .
     .            .
```

The DSCPs for this example have not yet been assigned an 802.1p priority level.

Figure 6-29. Display the Current Configuration in the DSCP Policy Table

2. Configure the priorities for the DSCPs you want to use.

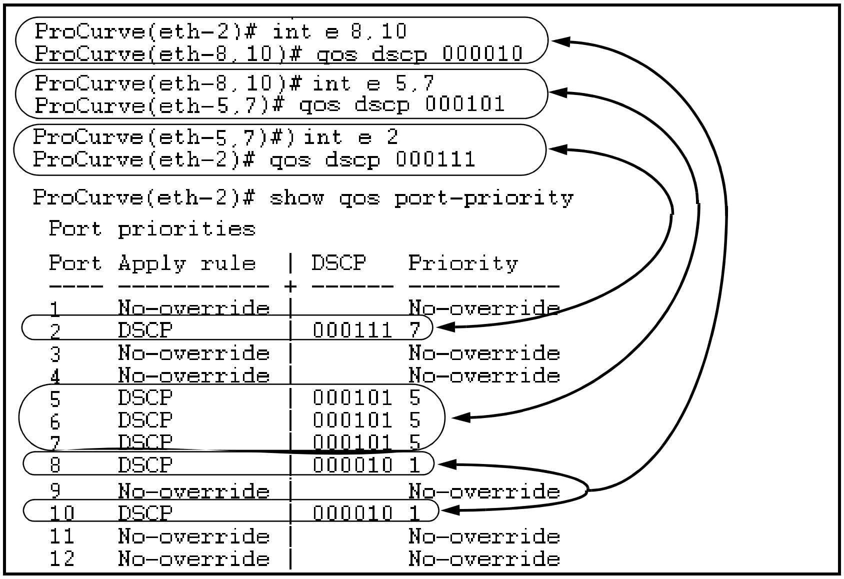


```
ProCurve(config)# qos dscp-map 000111 priority 7
ProCurve(config)# qos dscp-map 000101 priority 5
ProCurve(config)# qos dscp-map 000010 priority 1
ProCurve(config)# show qos dscp-map
  DSCP -> 802.p priority mappings
  DSCP policy 802.1p tag  Policy name
  ----------- ----------- ------------
  000000      No-override
  000001      No-override
  000010      1
  000011      No-override
  000100      No-override
  000101      5
  000110      No-override
  000111      7
  001000      No-override
     .            .
     .            .
     .            .
```

Priorities Configured in this step.

Figure 6-30. Assign Priorities to the Selected DSCPs

3. Assign the DSCP policies to the selected source-ports and display the result.



```
ProCurve(eth-2)# int e 8,10
ProCurve(eth-8,10)# qos dscp 000010
ProCurve(eth-8,10)# int e 5,7
ProCurve(eth-5,7)# qos dscp 000101
ProCurve(eth-5,7)#) int e 2
ProCurve(eth-2)# qos dscp 000111

ProCurve(eth-2)# show qos port-priority
 Port priorities

 Port Apply rule   | DSCP   Priority
 ---- ----------- + ------ -----------
 1    No-override  |        No-override
 2    DSCP         | 000111 7
 3    No-override  |        No-override
 4    No-override  |        No-override
 5    DSCP         | 000101 5
 6    DSCP         | 000101 5
 7    DSCP         | 000101 5
 8    DSCP         | 000010 1
 9    No-override  |        No-override
 10   DSCP         | 000010 1
 11   No-override  |        No-override
 12   No-override  |        No-override
```

Figure 6-31. The Completed Source-Port DSCP-Priority Configuration

Differentiated Services Codepoint (DSCP) Mapping

The DSCP Policy Table associates an 802.1p priority with a specific ToS byte codepoint in an IPv4 packet. This enables you to set a LAN policy that operates independently of 802.1Q VLAN-tagging.

In the default state, most of the 64 codepoints do not assign an 802.1p priority, as indicated by **No-override** in table 6-11 on page 6-59.

You can list the current DSCP Policy table, change the codepoint priority assignments, and assign optional names to the codepoints.

Syntax: show qos dscp-map

Displays the DSCP Policy Table.

qos dscp-map **<** *codepoint* > priority < 0 - 7 > [name < *ascii-string* >]

Configures an 802.1p priority for the specified codepoint and, optionally, an identifying (policy) name.

no qos dscp-map < *codepoint* >

*Reconfigures the 802.1p priority for <****codepoint****> to* **No-override**. *Also deletes the codepoint policy name, if configured.*

no qos dscp-map < *codepoint* > name

Deletes only the ***policy name, if*** *configured, for <* ***codepoint*** *>.*

Table 6-11. The Default DSCP Policy Table

DSCP Policy	802.1p Priority	DSCP Policy	802.1p Priority	DSCP Policy	802.1p Priority
000000	No-override	010110	3*	101011	No-override
000001	No-override	010111	No-override	101100	No-override
000010	No-override	011000	No-override	101101	No-override
000011	No-override	011001	No-override	101110	7[+]
000100	No-override	011010	4*	101111	No-override
000101	No-override	011011	No-override	110000	No-override
000110	No-override	011100	4*	110001	No-override
000111	No-override	011101	No-override	110010	No-override
001000	No-override	011110	5*	110011	No-override
001001	No-override	011111	No-override	110100	No-override
001010	1*	100000	No-override	110101	No-override
001011	No-override	100001	No-override	110110	No-override
001100	1*	100010	6*	110111	No-override
001101	No-override	100011	No-override	111000	No-override
001110	2*	100100	6*	111001	No-override
001111	No-override	100101	No-override	111010	No-override
010000	No-override	100110	7*	111011	No-override
010001	No-override	100111	No-override	111100	No-override
010010	0 *	101000	No-override	111101	No-override
010011	No-override	101001	No-override	111110	No-override
010100	0 *	101010	No-override	111111	No-override
010101	No-override				

*Assured Forwarding codepoints; configured by default on the Series 5304xl switches. These codepoints are configured as "No-override" in the Series 3400cl, 2600/2600-PWR, 2800 and 2810 switches.

[+] Expedited Forwarding codepoint configured by default.

Default Priority Settings for Selected Codepoints

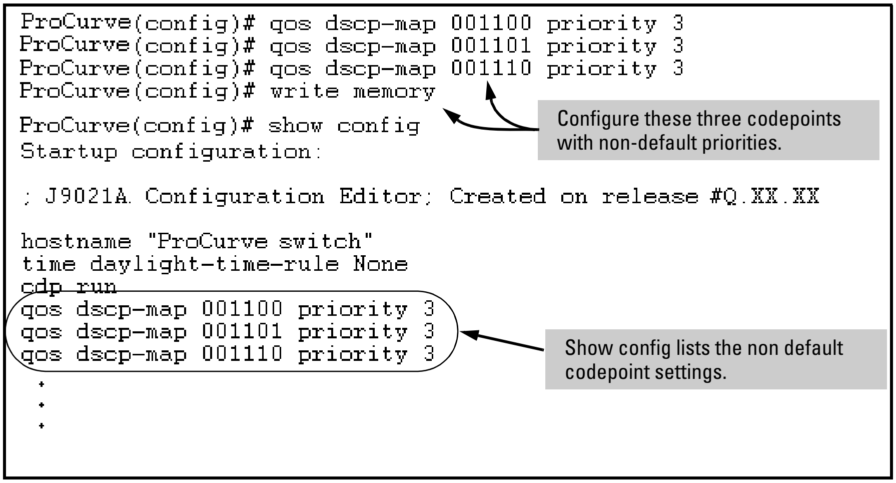
In a few cases, such as 001010 and 001100, a default policy (implied by the DSCP standards for Assured-Forwarding and Expedited-Forwarding) is used. You can change the priorities for the default policies by using **qos dscp-map <*codepoint*> priority < 0 - 7 >)**.(These policies are not in effect unless you have either applied the policies to a QoS type or configured QoS Type-of-Service to be in **diff-services** mode.)

Quickly Listing Non-Default Codepoint Settings

Table 6-11 lists the switch's default codepoint/priority settings. If you change the priority of any codepoint setting to a non-default value and then execute **write memory**, the switch will list the non-default setting in the show config display. For example, in the default configuration, the following codepoint settings are true:

Codepoint	Default Priority
001100	1
001101	No-override
001110	2

If you change all three settings to a priority of 3, and then execute **write memory**, the switch will reflect these changes in the show config listing:



```
ProCurve(config)# qos dscp-map 001100 priority 3
ProCurve(config)# qos dscp-map 001101 priority 3
ProCurve(config)# qos dscp-map 001110 priority 3
ProCurve(config)# write memory
ProCurve(config)# show config
Startup configuration:

; J9021A Configuration Editor; Created on release #Q.XX.XX

hostname "ProCurve switch"
time daylight-time-rule None
cdp run
qos dscp-map 001100 priority 3
qos dscp-map 001101 priority 3
qos dscp-map 001110 priority 3
  .
  .
  .
```

Configure these three codepoints with non-default priorities.

Show config lists the non default codepoint settings.

Figure 6-32. Example of Show Config Listing with Non-Default Priority Settings in the DSCP Table

Effect of "No-override". In the QoS Type-of-Service differentiated services mode, a **No-override** assignment for the codepoint of an outbound packet means that QoS is effectively disabled for such packets. That is, QoS does not affect the packet queuing priority or VLAN tagging. In this case, the packets are handled as follows (as long as no other QoS feature creates priority assignments for them):

802.1Q Status	Outbound 802.1p Priority
Received and Forwarded on a tagged port member of a VLAN.	Unchanged
Received on an Untagged port member of a VLAN; Forwarded on a tagged port member of a VLAN.	0 (zero)—"normal"
Forwarded on an Untagged port member of a VLAN.	None

Note On Changing a Priority Setting

If a QoS type is using a policy (codepoint and associated priority) in the DSCP Policy table, you must delete or change this usage before you can change the priority setting on the codepoint. Otherwise the switch blocks the change and displays this message:

```
Cannot modify DSCP Policy < codepoint > - in use by other
qos rules.
```

In this case, use **show qos <** ***type*** **>** to identify the specific types using the policy you want to change; that is:

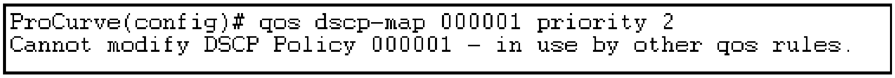
```
show qos device-priority
show qos port-priority
show qos tcp-udp-port-priority
show qos vlan-priority
show qos type-of-service
```

For example, suppose that the 000001 codepoint has a priority of 6, and several types use the 000001 codepoint to assign a priority to their respective types of traffic. If you wanted to change the priority of codepoint 000001 you would do the following:

1. Identify which QoS types use the codepoint.
2. Change the type configurations by assigning them to a different DSCP policy, or to an 802.1p priority, or to **No-override**.
3. Reconfigure the desired priority for the 000001 codepoint.
4. Either reassign the types to the 00001 codepoint policy or leave them as they were after step 2, above.

Example of Changing the Priority Setting on a Policy When One or More QoS Types Are Currently Using the Policy

Suppose that codepoint 000001 is in use by one or more types. If you try to change its priority, you see a result similar to the following:

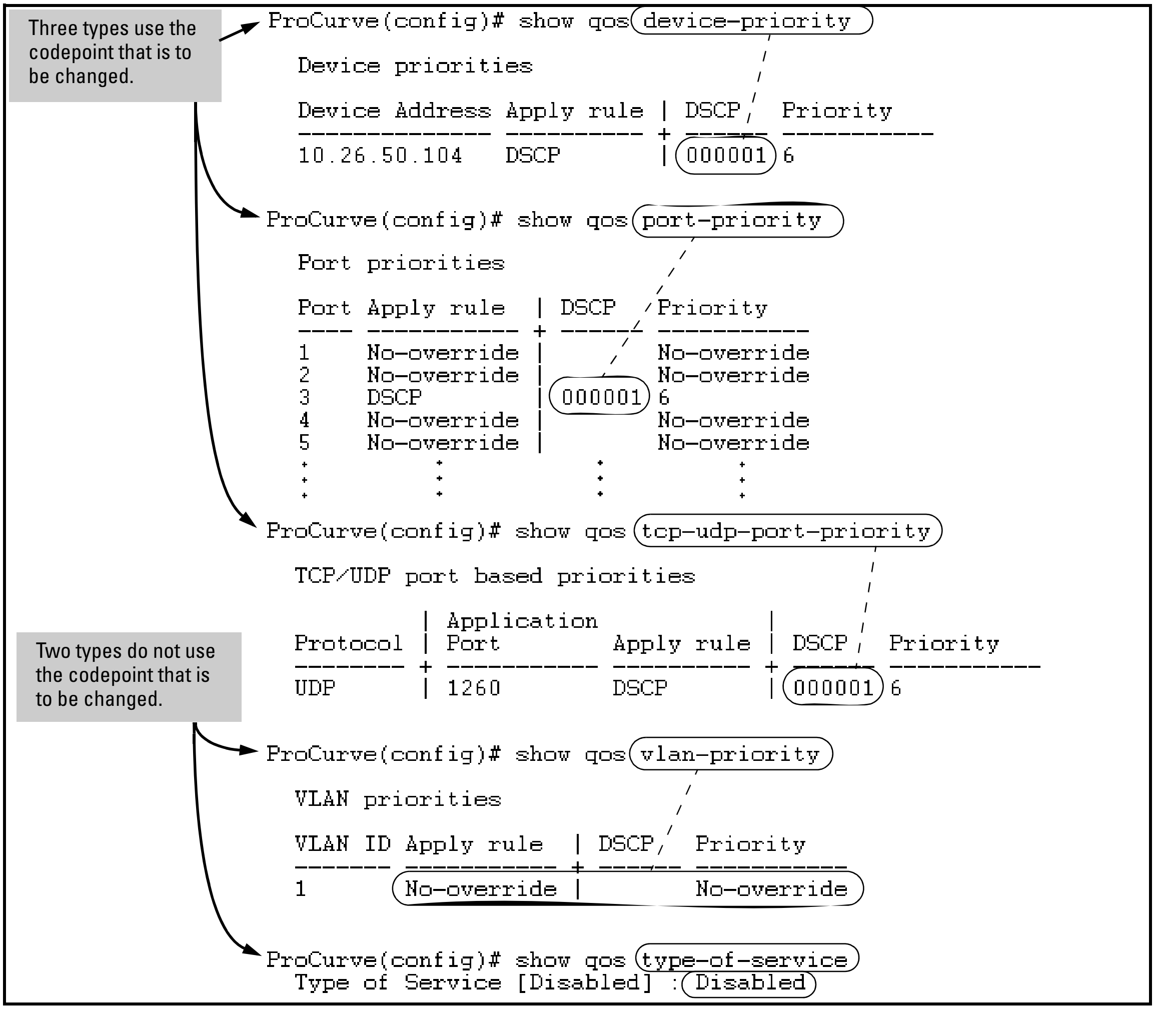


```
ProCurve(config)# qos dscp-map 000001 priority 2
Cannot modify DSCP Policy 000001 - in use by other qos rules.
```

Figure 6-33. Example of Trying To Change the Priority on a Policy In Use by a QoS Type

In this case, you would use steps similar to the following to change the priority.

1. Identify which types use the codepoint you want to change.



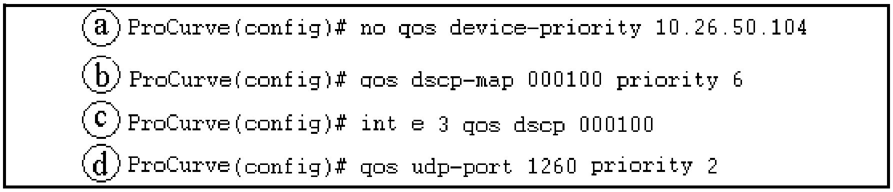
Three types use the codepoint that is to be changed.

Two types do not use the codepoint that is to be changed.

```
ProCurve(config)# show qos device-priority

  Device priorities

  Device Address Apply rule | DSCP   Priority
  -------------- ---------- + ------ -----------
  10.26.50.104   DSCP       | 000001 6

ProCurve(config)# show qos port-priority

  Port priorities

  Port Apply rule  | DSCP   Priority
  ---- ----------- + ------ -----------
  1    No-override |        No-override
  2    No-override |        No-override
  3    DSCP        | 000001 6
  4    No-override |        No-override
  5    No-override |        No-override
  .         .           .         .
  .         .           .         .
  .         .           .         .
ProCurve(config)# show qos tcp-udp-port-priority

  TCP/UDP port based priorities

           | Application            |
  Protocol | Port        Apply rule | DSCP   Priority
  -------- + ----------- ---------- + ------ -----------
  UDP      | 1260        DSCP       | 000001 6

ProCurve(config)# show qos vlan-priority

  VLAN priorities

  VLAN ID Apply rule  | DSCP   Priority
  ------- ----------- + ------ -----------
  1       No-override |        No-override

ProCurve(config)# show qos type-of-service
  Type of Service [Disabled] :Disabled
```

Figure 6-34. Example of a Search to Identify QoS Types Using a Codepoint You Want To Change

2. Change the type configurations by assigning them to a different DSCP policy, or to an 802.1p priority, or to **No-override**. For example:
   a. Delete the policy assignment for the **device-priority** type. (That is, assign it to **No-override**.)
   b. Create a new DSCP policy to use for re-assigning the remaining types.
   c. Assign the **port-priority** type to the new DSCP policy.
   d. Assign the **udp-port 1260** type to an 802.1p priority.



```
a ProCurve(config)# no qos device-priority 10.26.50.104
b ProCurve(config)# qos dscp-map 000100 priority 6
c ProCurve(config)# int e 3 qos dscp 000100
d ProCurve(config)# qos udp-port 1260 priority 2
```

3. Reconfigure the desired priority for the 000001 codepoint.

```
ProCurve(config)# qos dscp-map 000001 priority 4
```

4. You could now re-assign the types to the original policy codepoint or leave them as currently configured.

IP Multicast (IGMP) Interaction with QoS

IGMP high-priority-forward causes the switch to service the subscribed IP multicast group traffic at high priority, even if QoS on the switch has relegated the traffic to a lower priority. This does not affect any QoS priority settings, so the QoS priority is honored by downstream devices. However, QoS does take precedence over IGMP normal-priority traffic.

The switch's ability to prioritize IGMP traffic for either a normal or high priority outbound queue overrides any QoS criteria, and does not affect any 802.1p priority settings the switch may assign. For a given packet, if both IGMP high priority and QoS are configured, the QoS classification occurs and the switch marks the packet for downstream devices, but the packet is serviced by the high-priority queue when leaving the switch.

IGMP High Priority	QoS Configuration Affects Packet	Switch Port Output Queue	Outbound 802.1p Setting (Requires Tagged VLAN)
Not Enabled	Yes	Determined by QoS	Determined by QoS
Enabled	See above paragraph.	High	As determined by QoS if QoS is active.

QoS Messages in the CLI

Message	Meaning
DSCP Policy < ***decimal-codepoint*** > not configured	You have attempted to map a QoS type to a codepoint for which there is no configured priority (**No-override**). Use the **qos dscp-map** command to configure a priority for the codepoint, then map the type to the codepoint.
Cannot modify DSCP Policy < ***codepoint*** > - in use by other qos rules.	You have attempted to map a QoS type to a codepoint that is already in use by other QoS types. Before remapping the codepoint to a new priority, you must reconfigure the other QoS types so that they do not use this codepoint. You can have multiple QoS types use this same codepoint as long as it is acceptable for all such types to use the same priority.

QoS Operating Notes and Restrictions

Table 6-12. Details of Packet Criteria and Restrictions for QoS Support

Packet Criteria or Restriction	QoS Types						DSCP Overwrite (Re-Marking)
	UDP/TCP	Device Priority (IP Address)	IP Type-of-Service	VLAN	Source Port	Incoming 802.1p	
Restricted to IPv4 Packets Only	Yes	Yes	Yes	No	No	No	Yes
Allow Packets with IP Options[1]	No	Yes	Yes	Yes	Yes	Yes	Yes
Support IPv6 Packets[2]	No	No	No	Yes	Yes	Yes	No
Support Layer-2 SAP Encapsulation	No	No	No	Yes	Yes	Yes	No

[1] An "IP Option" is an optional, extra field in the header of an IP packet. If the switch is configured with a UDP/TCP type and a packet with an IP option is received, the switch uses the next-highest type, if configured.

[2] All Switches: For explicit QoS support of IPv6 packets, force IPv6 traffic into its own set of VLANs and then configure VLAN-based types for those VLANs.

- **All Switches:** For explicit QoS support of IP subnets, ProCurve recommends forcing IP subnets onto separate VLANs and then configuring VLAN-based types for those VLANs.
- **For Devices that Do Not Support 802.1Q VLAN-Tagged Ports:** For communication between these devices and the switch, connect the device to a switch port configured as **Untagged** for the VLAN in which you want the device's traffic to move.
- **Port Tagging Rules:** For a port on the switch to be a member of a VLAN, the port must be configured as either **Tagged** or **Untagged** for that VLAN. For more on VLANs, refer to chapter 2, "Static Virtual LANs (VLANs)".
- **SAP-Encapsulated Packet Restriction:** Except for source-port QoS and VLAN QoS, the switches covered in this guide do not support QoS operation for SAP-Encapsulated packets. Thus, the switch can use only VLAN QoS and source-port QoS to prioritize SAP-encapsulated packets.
- **Packets with IP Option Fields in the Header:** UDP/TCP QoS is not supported for IP packets carrying optional fields in their headers.

- **RADIUS Authentication:** RADIUS authentication allowing traffic through a given port may override the port's QoS configuration, which generates an Event Log message. When the authenticated host disconnects, the port returns to the static QoS configuration.
- **Not Supported:** Use of an inbound 802.1p packet priority as a type for remapping a packet's outbound priority to different 802.1p priority. For example, where inbound packets carry an 802.1p priority of 1, QoS cannot be configured use this priority as a type for changing the outbound priority to 0.

ProCurve Stack Management

Contents

Overview

This chapter describes how to use your network to stack switches without the need for any specialized cabling. For an overview of stacking features, refer to the table on page 7-4.

For general information on how to use the switch's built-in interfaces, see:

- Chapter 3, "Using the Menu Interface"
- Chapter 4, "Using the Command Line Interface (CLI)"
- Chapter 5, "Using the Web Browser Interface
- Chapter 6, "Switch Memory and Configuration"

Operation

Stacking Features

Feature	Default	Menu	CLI	Web
view stack status				
view status of a single switch	n/a	page 7-28 thru page 7-30	page 7-33	page 7-47
view candidate status	n/a	↑	page 7-33	↑
view status of commander and its stack	n/a		page 7-34	
view status of all stacking-enabled switches in the ip subnet	n/a		page 7-34	
configure stacking				
enable/disable candidate Auto-Join	enabled/Yes	page 7-17	page 7-39	
“push” a candidate into a stack	n/a	page 7-17	page 7-40	
configure a switch to be a commander	n/a	page 7-14	page 7-35	
“push” a member into another stack	n/a	page 7-26	page 7-41	
remove a member from a stack	n/a	page 7-23	page 7-42 or page 7-43	
“pull” a candidate into a stack	n/a	page 7-19	page 7-38	
“pull” a member from another stack	n/a	page 7-21	page 7-40	
convert a commander or member to a member of another stack	n/a	page 7-26	page 7-41	
access member switches for configuration and traffic monitoring	n/a	page 7-25	page 7-44	
disable stacking	enabled	page 7-17	page 7-46	
transmission interval	60 seconds	page 7-14	page 7-46	

ProCurve Stack Management (termed *stacking*) enables you to use a single IP address and standard network cabling to manage a group of up to 16 total switches in the same IP subnet (broadcast domain). Using stacking you can:

- Reduce the number of IP addresses needed in your network.

- Simplify management of small workgroups or wiring closets while scaling your network to handle increased bandwidth demand.
- Eliminate any specialized cables for stacking connectivity and remove the distance barriers that typically limit your topology options when using other stacking technologies.
- Add switches to your network without having to first perform IP addressing tasks.

Which Devices Support Stacking?

As of April, 2006, the following ProCurve devices support stacking:

- ProCurve Switch 6108
- ProCurve Switch 4104GL
- ProCurve Switch 4108GL
- ProCurve Switch 2810
- ProCurve Switch 2650
- ProCurve Switch 2626
- ProCurve Switch 2512
- ProCurve Switch 2524
- ProCurve Switch 8000M*
- ProCurve Switch 4000M*
- ProCurve Switch 2424M*
- ProCurve Switch 2400M*
- ProCurve Switch 1600M*

*Requires software release C.08.03 or later, which is included with the 8000M, 4000M, 2424M, and 1600M models as of July, 2000. Release C.08.03 or a later version is also available on the ProCurve website at **www.procurve.com**. (Click on **software**.)

Components of ProCurve Stack Management

Table 7-1. Stacking Definitions

Stack	Consists of a Commander switch and any Member switches belonging to that Commander's stack.
Commander	A switch that has been manually configured as the controlling device for a stack. When this occurs, the switch's stacking configuration appears as **Commander**.
Candidate	A switch that is ready to join (become a Member of) a stack through either automatic or manual methods. A switch configured as a Candidate is not in a stack.
Member	A switch that has joined a stack and is accessible from the stack Commander.

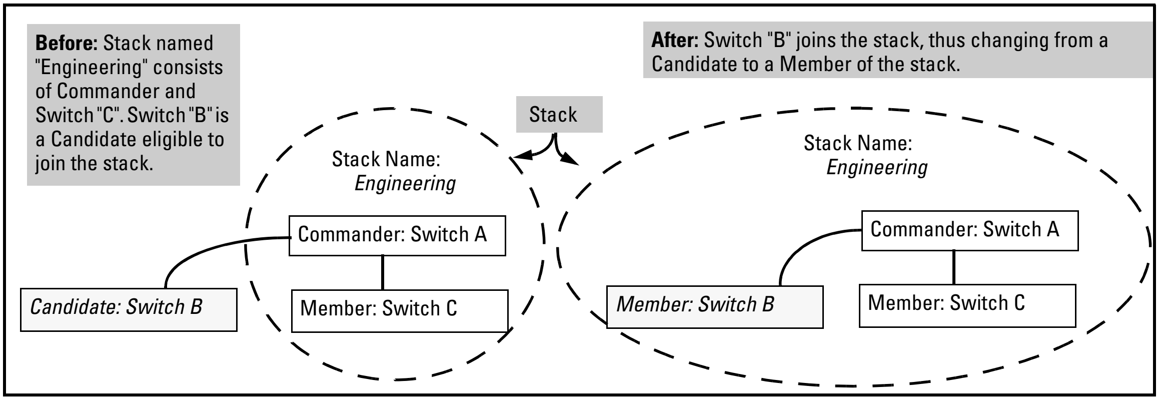

Before: Stack named "Engineering" consists of Commander and Switch "C". Switch "B" is a Candidate eligible to join the stack.
After: Switch "B" joins the stack, thus changing from a Candidate to a Member of the stack.
Stack
Stack Name: *Engineering*
Commander: Switch A
Member: Switch C
Candidate: Switch B
Stack Name: *Engineering*
Commander: Switch A
Member: Switch B
Member: Switch C

Figure 7-1. Illustration of a Switch Moving from Candidate to Member

General Stacking Operation

After you configure one switch to operate as the Commander of a stack, additional switches can join the stack by either automatic or manual methods. After a switch becomes a Member, you can work through the Commander switch to further configure the Member switch as necessary for all of the additional software features available in the switch.

The Commander switch serves as the in-band entry point for access to the Member switches. For example, the Commander's IP address becomes the path to all stack Members and the Commander's Manager password controls access to all stack Members.

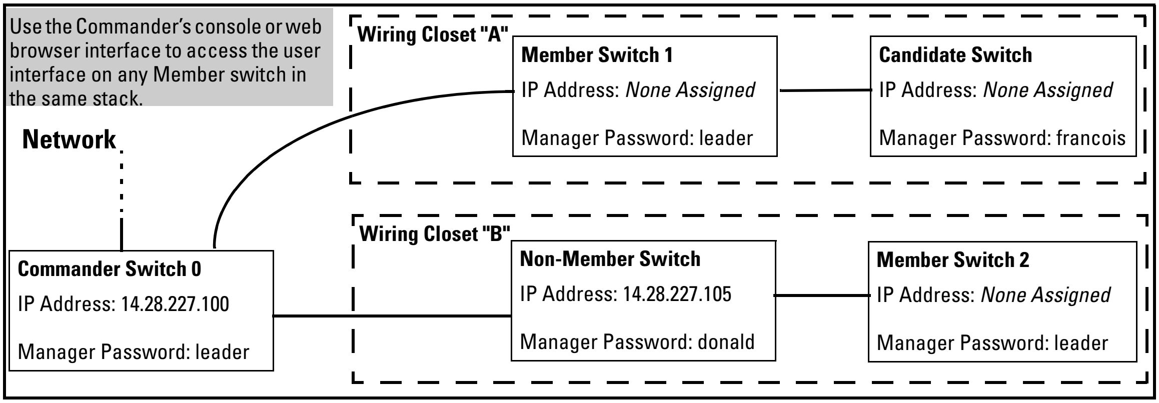

Use the Commander's console or web browser interface to access the user interface on any Member switch in the same stack.
Network
Wiring Closet "A"
Member Switch 1
IP Address: *None Assigned*
Manager Password: leader
Candidate Switch
IP Address: *None Assigned*
Manager Password: francois
Commander Switch 0
IP Address: 14.28.227.100
Manager Password: leader
Wiring Closet "B"
Non-Member Switch
IP Address: 14.28.227.105
Manager Password: donald
Member Switch 2
IP Address: *None Assigned*
Manager Password: leader

Figure 7-2. Example of Stacking with One Commander Controlling Access to Wiring Closet Switches

Interface Options. You can configure stacking through the switch's menu interface, CLI, or the web browser interface. For information on how to use the web browser interface to configure stacking, see the online Help for the web browser interface.

Web Browser Interface Window for Commander Switches. The web browser interface window for a Commander switch differs in appearance from the same window for non-commander switches. See figure 7-38 on page 7-47.

Operating Rules for Stacking

General Rules

- Stacking is an optional feature (enabled in the default configuration) and can easily be disabled. Stacking has no effect on the normal operation of the switch in your network.
- A stack requires one Commander switch. (Only one Commander allowed per stack.)
- All switches in a particular stack must be in the same IP subnet (broadcast domain). A stack cannot cross a router.
- A stack accepts up to 16 switches (numbered 0-15), including the Commander (always numbered 0).
- The stacking feature supports up to 100 switches in the same IP subnet (broadcast domain). A switch can belong to only one stack. In the event that the 100 switch limit is exceeded, it may take multiple attempts to add or move a member to any given stack. Once a member is added to a stack, it is not "forgotten" by the Commander.
- Viewing Stack Status (all) only displays up to 100 devices. Devices that are not members of a given stack may periodically "drop out" of the list when viewing Stack Status (all).
- If multiple VLANs are configured, stacking uses only the primary VLAN on any switch. In the factory-default configuration, the DEFAULT_VLAN is the primary VLAN. (See "Stacking Operation with Multiple VLANs Configured" on page 7-46 and "The Primary VLAN" on page 2-6.)
- Stacking allows intermediate devices that do not support stacking. This enables you to include switches that are distant from the Commander.

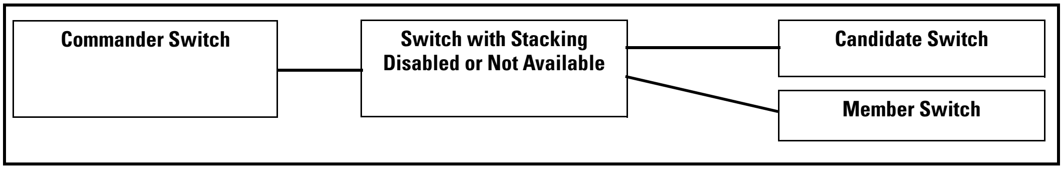


Figure 7-3. Example of a Non-Stacking Device Used in a Stacking Environment

Specific Rules

Table 7-2. Specific Rules for Commander, Candidate, and Member Switch

	IP Addressing and Stack Name	Number Allowed Per Stack	Passwords	SNMP Communities
Commander	**IP Addr:** Requires an assigned IP address and mask for access via the network. **Stack Name:** Required	Only one Commander switch is allowed per stack.	The Commander's Manager and Operator passwords are assigned to any switch becoming a Member of the stack. If you change the Commander's passwords, the Commander propagates the new passwords to all stack Members.	Standard SNMP community operation. The Commander also operates as an SNMP proxy to Members for all SNMP communities configured in the Commander.
Candidate	**IP Addr:** Optional. Configuring an IP address allows access via Telnet or web browser interface while the switch is not a stack member. In the factory default configuration the switch automatically acquires an IP address if your network includes DHCP service. **Stack Name:** N/A	n/a	Passwords optional. If the Candidate becomes a stack Member, it assumes the Commander's Manager and Operator passwords. If a candidate has a password, it cannot be automatically added to a stack. In this case, if you want the Candidate in a stack, you must manually add it to the stack.	Uses standard SNMP community operation if the Candidate has its own IP addressing.
Member	**IP Addr:** Optional. Configuring an IP address allows access via Telnet or web browser interface without going through the Commander switch. This is useful, for example, if the stack Commander fails and you need to convert a Member switch to operate as a replacement Commander. **Stack Name:** N/A	Up to 15 Members per stack.	When the switch joins the stack, it automatically assumes the Commander's Manager and Operator passwords and discards any passwords it may have had while a Candidate. **Note:** If a Member leaves a stack for any reason, it retains the passwords assigned to the stack Commander at the time of departure from the stack.	Belongs to the same SNMP communities as the Commander (which serves as an SNMP proxy to the Member for communities to which the Commander belongs). To join other communities that *exclude* the Commander, the Member must have its own IP address. Loss of stack membership means loss of membership in any community that is configured only in the Commander. See "SNMP Community Operation in a Stack" on page 7-45.

Note

In the default stack configuration, the Candidate **Auto Join** parameter is enabled, but the Commander **Auto Grab** parameter is disabled. This prevents Candidates from automatically joining a stack prematurely or joining the wrong stack (if more than one stack Commander is configured in a subnet or broadcast domain). If you plan to install more than one stack in a subnet, ProCurve recommends that you leave **Auto Grab** disabled on all Commander switches and manually add Members to their stacks. Similarly, if you plan to install a stack in a subnet (broadcast domain) where stacking-capable switches are not intended for stack membership, you should set the **Stack State** parameter (in the Stack Configuration screen) to **Disabled** on those particular switches.

Configuring Stack Management

Overview of Configuring and Bringing Up a Stack

This process assumes that:

- All switches you want to include in a stack are connected to the same subnet (broadcast domain).
- If VLANs are enabled on the switches you want to include in the stack, then the ports linking the stacked switches must be on the primary VLAN in each switch (which, in the default configuration, is the default VLAN). If the primary VLAN is tagged, then each switch in the stack must use the same VLAN ID (VID) for the primary VLAN. (Refer to "The Primary VLAN" on page 2-6, and "Stacking Operation with Multiple VLANs Configured" on page 7-46.)
- *If you are including an ProCurve Switch 8000M, 4000M, 2424M, 2400M, or 1600M in a stack, you must first update all such devices to software version C.08.03 or later.* (You can get a copy of the latest software version from the ProCurve Networking website and/or copy it from one switch to another. For downloading instructions, see appendix A, "File Transfers", in the *Management and Configuration Guide* for these switch models.)

Options for Configuring a Commander and Candidates. Depending on how Commander and Candidate switches are configured, Candidates can join a stack either automatically or by a Commander manually adding ("pulling") them into the stack. In the default configuration, a Candidate joins only when *manually* pulled by a Commander. You can reconfigure a Commander to *automatically* pull in Candidates that are in the default stacking configuration. You can also reconfigure a Candidate switch to either "push" itself into a particular Commander's stack, convert the Candidate to a Commander (for a stack that does not already have a Commander), or to operate as a standalone switch without stacking. The following table shows your control options for adding Members to a stack.

Table 7-3. Stacking Configuration Guide

Join Method[1]	**Commander** **(IP Addressing Required)**	**Candidate** **(IP Addressing Optional)**	
	Auto Grab	**Auto Join**	**Passwords**
Automatically add Candidate to Stack (Causes the first 15 eligible, discovered switches in the subnet to automatically join a stack.)	**Yes**	**Yes** *(default)*	No *(default)****
Manually add Candidate to Stack (Prevent automatic joining of switches you don't want in the stack)	**No** *(default)*	**Yes** *(default)*	Optional*
	Yes	**No**	Optional*
	Yes	**Yes** *(default)* or **No**	Configured
Prevent a switch from being a Candidate	**N/A**	**Disabled**	Optional

*The Commander's Manager and Operator passwords propagate to the candidate when it joins the stack.

The easiest way to *automatically* create a stack is to:

1. Configure a switch as a Commander.
2. Configure IP addressing and a stack name on the Commander.
3. Set the Commander's **Auto Grab** parameter to **Yes**.
4. Connect Candidate switches (in their factory default configuration) to the network.

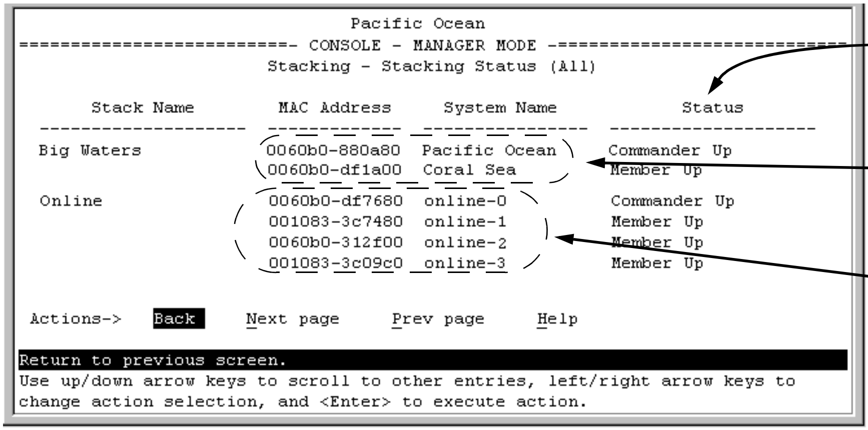
This approach automatically creates a stack of up to 16 switches (including the Commander). However this replaces manual control with an automatic process that may bring switches into the stack that you did not intend to include. With the Commander's **Auto Grab** parameter set to **Yes**, *any switch* conforming to all four of the following factors automatically becomes a stack Member:

- Default stacking configuration (**Stack State** set to **Candidate**, and **Auto Join** set to **Yes**)
- Same subnet (broadcast domain) and default VLAN as the Commander (If VLANs are used in the stack environment, see “Stacking Operation with a Tagged VLAN” on page 7-46.)
- No Manager password
- 14 or fewer stack members at the moment

General Steps for Creating a Stack

This section describes the general stack creation process. For the detailed configuration processes, see pages 7-14 through 7-38 for the menu interface and pages 7-31 through 7-43 for the CLI.

1. Determine the naming conventions for the stack. You will need a stack name. Also, to help distinguish one switch from another in the stack, you can configure a unique system name for each switch. Otherwise, the system name for a switch appearing in the Stacking Status screen appears as the stack name plus an automatically assigned switch number. For example:



```
                                  Pacific Ocean
=========================- CONSOLE - MANAGER MODE -=========================
                            Stacking - Stacking Status (All)

      Stack Name          MAC Address     System Name          Status
  --------------------  -------------  ---------------  --------------------
  Big Waters            0060b0-880a80  Pacific Ocean    Commander Up
                        0060b0-df1a00  Coral Sea        Member Up

  Online                0060b0-df7680  online-0         Commander Up
                        001083-3c7480  online-1         Member Up
                        0060b0-312f00  online-2         Member Up
                        001083-3c09c0  online-3         Member Up


 Actions->  Back     Next page     Prev page     Help

Return to previous screen.
Use up/down arrow keys to scroll to other entries, left/right arrow keys to
change action selection, and <Enter> to execute action.
```

For status descriptions, see the table on page 7-48.

Stack with unique system name for each switch.

Stack named "Online" with no previously configured system names assigned to individual switches.

Figure 7-4. Using the System Name to Help Identify Individual Switches

2. Configure the Commander switch. Doing this first helps to establish consistency in your stack configuration, which can help prevent startup problems.

   - A stack requires one Commander switch. If you plan to implement more than one stack in a subnet (broadcast domain), the easiest way to avoid unintentionally adding a Candidate to the wrong stack is to manually control the joining process by leaving the Commander's **Auto Grab** parameter set to **No** (the default).
   - The Commander assigns its Manager and Operator passwords to any Candidate switch that joins the stack.
   - The Commander's SNMP community names apply to members.

3. For automatically or manually pulling Candidate switches into a stack, you can leave such switches in their default stacking configuration. If you need to access Candidate switches through your network before they join the stack, assign IP addresses to these devices. Otherwise, IP addressing is optional for Candidates and Members. (Note that once a Candidate becomes a member, you can access it through the Commander to assign IP addressing or make other configuration changes.)

4. Make a record of any Manager passwords assigned to the switches (intended for your stack) that are not currently members. (You will use these passwords to enable the protected switches to join the stack.)

5. If you are using VLANs in the stacking environment, you must use the default VLAN for stacking links. For more information, see "Stacking Operation with a Tagged VLAN" on page 7-46.

6. Ensure that all switches intended for the stack are connected to the same subnet (broadcast domain). As soon as you connect the Commander, it will begin discovering the available Candidates in the subnet.

   - If you configured the Commander to automatically add Members (**Auto Grab** = **Yes**), the first fifteen discovered Candidates meeting both of the following criteria will automatically join the stack:
     – **Auto Join** parameter set to **Yes** (the default)
     – Manager password not configured
   - If you configured the Commander to manually add Members (**Auto Grab** set to **No**—the default), you can begin the process of selecting and adding the desired Candidates.

7. Ensure that all switches intended for the stack have joined.

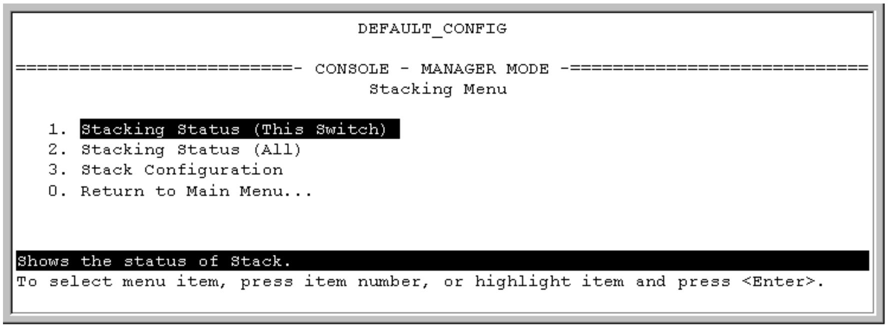
8. If you need to do specific configuration or monitoring tasks on a Member, use the console interface on the Commander to access the Member.

Using the Menu Interface To View Stack Status and Configure Stacking

Using the Menu Interface To View and Configure a Commander Switch

1. Configure an IP address and subnet mask on the Commander switch. (See the chapter on IP addressing in the *Management and Configuration Guide*.)

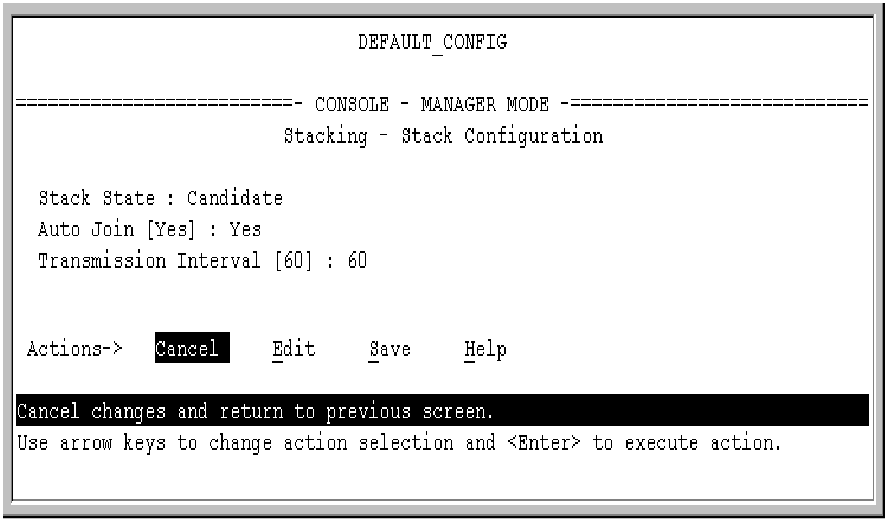
2. Display the Stacking Menu by selecting **Stacking** in the Main Menu.



```
                                    DEFAULT_CONFIG

==========================- CONSOLE - MANAGER MODE -============================
                                     Stacking Menu

   1. Stacking Status (This Switch)
   2. Stacking Status (All)
   3. Stack Configuration
   0. Return to Main Menu...


Shows the status of Stack.
To select menu item, press item number, or highlight item and press <Enter>.
```

Figure 7-5. The Default Stacking Menu

3. Display the Stack Configuration menu by pressing **[3]** to select **Stack Configuration**.

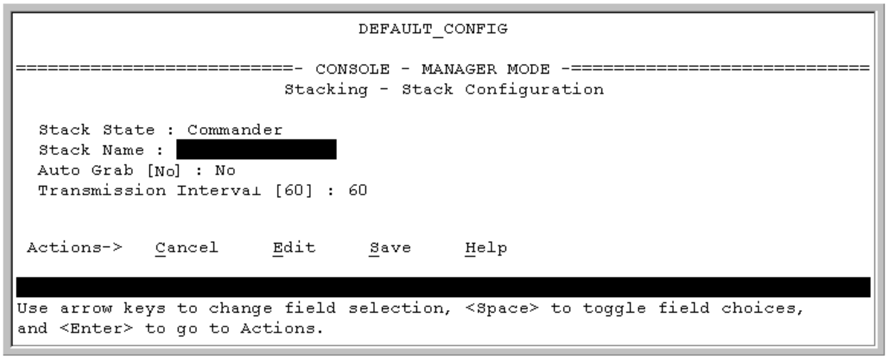


```
                               DEFAULT_CONFIG

==========================- CONSOLE - MANAGER MODE -============================
                        Stacking - Stack Configuration

  Stack State : Candidate
  Auto Join [Yes] : Yes
  Transmission Interval [60] : 60


 Actions->   Cancel     Edit     Save     Help

Cancel changes and return to previous screen.
Use arrow keys to change action selection and <Enter> to execute action.
```

Figure 7-6. The Default Stack Configuration Screen

4. Move the cursor to the Stack State field by pressing **[E]** (for **Edit**). Then use the Space bar to select the **Commander** option.

5. Press the downarrow key to display the Commander configuration fields in the Stack Configuration screen.



```
                               DEFAULT_CONFIG

==========================- CONSOLE - MANAGER MODE -============================
                        Stacking - Stack Configuration

  Stack State : Commander
  Stack Name : 
  Auto Grab [No] : No
  Transmission Interval [60] : 60


 Actions->   Cancel     Edit     Save     Help


Use arrow keys to change field selection, <Space> to toggle field choices,
and <Enter> to go to Actions.
```

Figure 7-7. The Default Commander Configuration in the Stack Configuration Screen

6. Enter a unique stack name (up to 15 characters; no spaces) and press the downarrow key.

7. Ensure that the Commander has the desired **Auto Grab** setting, then press the downarrow key:
   - **No** (the default) prevents automatic joining of Candidates that have their **Auto Join** set to **Yes**.
   - **Yes** enables the Commander to automatically take a Candidate into the stack as a Member if the Candidate has **Auto Join** set to **Yes** (the default Candidate setting) and does not have a previously configured password.
8. Accept or change the transmission interval (default: 60 seconds), then press **[Enter]** to return the cursor to the **Actions** line.
9. Press **[S]** (for **Save**) to save your configuration changes and return to the Stacking menu.

Your Commander switch should now be ready to automatically or manually acquire Member switches from the list of discovered Candidates, depending on your configuration choices.

Using the Menu To Manage a Candidate Switch

Using the menu interface, you can perform these actions on a Candidate switch:

- Add ("push") the Candidate into an existing stack
- Modify the Candidate's stacking configuration (**Auto Join** and **Transmission Interval**)
- Convert the Candidate to a Commander
- Disable stacking on the Candidate so that it operates as a standalone switch

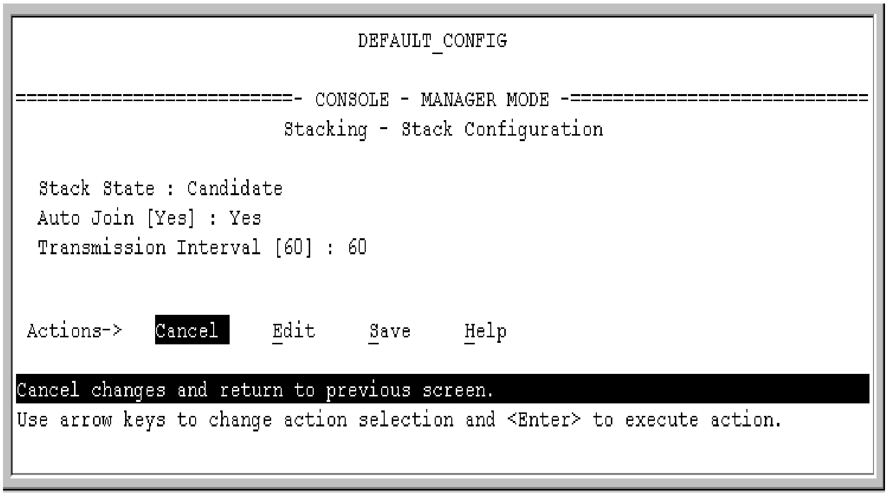
In its default stacking configuration, a Candidate switch can either automatically join a stack or be manually added ("pulled") into a stack by a Commander, depending on the Commander's **Auto Grab** setting. The following table lists the Candidate's configuration options:

Table 7-4. Candidate Configuration Options in the Menu Interface

Parameter	Default Setting	Other Settings
Stack State	Candidate	Commander, Member, or Disabled
Auto Join	Yes	No
Transmission Interval	60 Seconds	Range: 1 to 300 seconds

Using the Menu To "Push" a Switch Into a Stack, Modify the Switch's Configuration, or Disable Stacking on the Switch. Use Telnet or the web browser interface to access the Candidate if it has an IP address. Otherwise, use a direct connection from a terminal device to the switch's console port. (For information on how to use the web browser interface, see the online Help provided for the browser.)

1. Display the Stacking Menu by selecting **Stacking** in the console Main Menu.
2. Display the Stack Configuration menu by pressing **[3]** to select **Stack Configuration**.



```
                                DEFAULT_CONFIG

==========================- CONSOLE - MANAGER MODE -============================
                        Stacking - Stack Configuration

  Stack State : Candidate
  Auto Join [Yes] : Yes
  Transmission Interval [60] : 60


 Actions->   Cancel     Edit     Save     Help

Cancel changes and return to previous screen.
Use arrow keys to change action selection and <Enter> to execute action.
```

Figure 7-8. The Default Stack Configuration Screen

3. Move the cursor to the Stack State field by pressing **[E]** (for **Edit**).
4. Do one of the following:
   - To disable stacking on the Candidate, use the Space bar to select the **Disabled** option, then go to step 5.

     Note: Using the menu interface to disable stacking on a Candidate removes the Candidate from all stacking menus.
   - To insert the Candidate into a specific Commander's stack:

     i. Use the space bar to select Member.

     ii. Press **[Tab]** once to display the **Commander MAC Address** parameter, then enter the MAC address of the desired Commander.

- To change **Auto Join** or **Transmission Interval**, use **[Tab]** to select the desired parameter, and:
  - To change **Auto Join**, use the Space bar.
  - To change **Transmission Interval**, type in the new value in the range of 1 to 300 seconds.

    Note: All switches in the stack must be set to the same transmission interval to help ensure proper stacking operation. ProCurve recommends that you leave this parameter set to the default 60 seconds.

  Then go to step 5.

5. Press **[Enter]** to return the cursor to the **Actions** line.
6. Press **[S]** (for **Save**) to save your configuration changes and return to the Stacking menu.

Using the Commander To Manage The Stack

The Commander normally operates as your stack manager and point of entry into other switches in the stack. This typically includes:

- Adding new stack members
- Moving members between stacks
- Removing members from a stack
- Accessing stack members for individual configuration changes and traffic monitoring

The Commander also imposes its passwords on all stack members and provides SNMP community membership to the stack. (See “SNMP Community Operation in a Stack” on page 7-45.)

Using the Commander's Menu To Manually Add a Candidate to a Stack. In the default configuration, you must manually add stack Members from the Candidate pool. Reasons for a switch remaining a Candidate instead of becoming a Member include any of the following:

- **Auto Grab** in the Commander is set to **No** (the default).
- **Auto Join** in the Candidate is set to **No**.

  Note: When a switch leaves a stack and returns to Candidate status, its **Auto Join** parameter resets to **No** so that it will not immediately rejoin a stack from which it has just departed.
- A Manager password is set in the Candidate.
- The stack is full.

Unless the stack is already full, you can use the Stack Management screen to manually convert a Candidate to a Member. If the Candidate has a Manager password, you will need to use it to make the Candidate a Member of the stack.

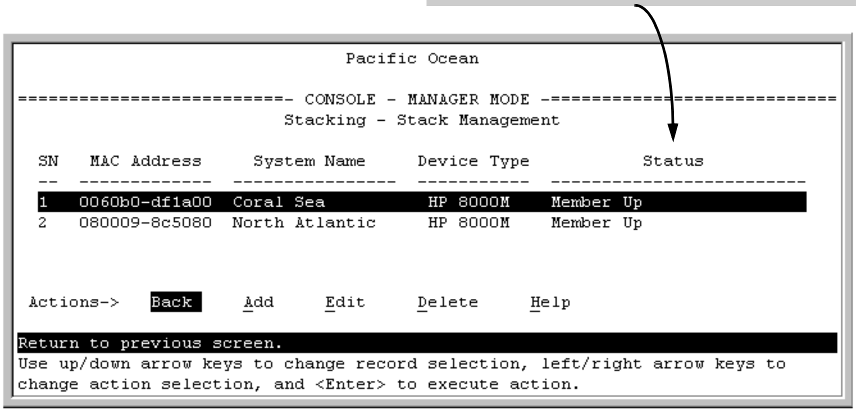
1. To add a Member, start at the Main Menu and select:

   9. Stacking...

   4. Stack Management

   You will then see the Stack Management screen:

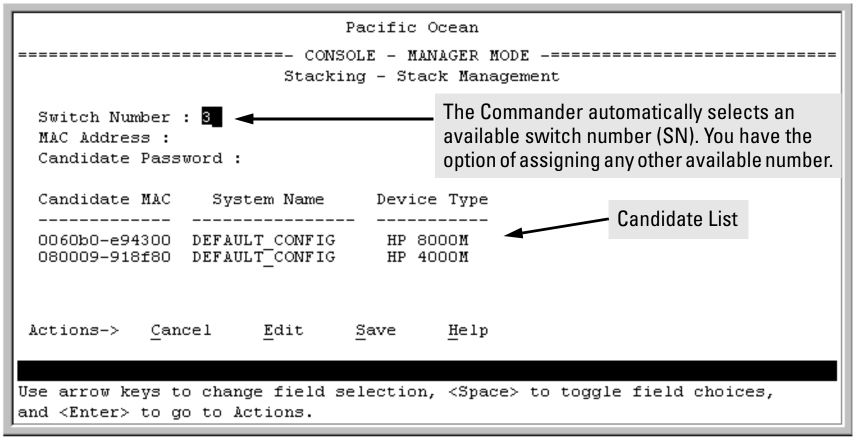
For status descriptions, see the table on page 7-48.



```
                              Pacific Ocean

========================- CONSOLE - MANAGER MODE -============================
                        Stacking - Stack Management

  SN   MAC Address     System Name     Device Type          Status
  --  -------------  ----------------  -----------  -------------------------
  1   0060b0-df1a00  Coral Sea          HP 8000M    Member Up
  2   080009-8c5080  North Atlantic     HP 8000M    Member Up


 Actions->   Back     Add     Edit     Delete     Help

Return to previous screen.
Use up/down arrow keys to change record selection, left/right arrow keys to
change action selection, and <Enter> to execute action.
```

Figure 7-9. Example of the Stack Management Screen

2. Press **[A]** (for **Add**) to add a Candidate. You will then see this screen listing the available Candidates:



```
                                  Pacific Ocean
==========================- CONSOLE - MANAGER MODE -============================
                          Stacking - Stack Management

  Switch Number : 3
  MAC Address :
  Candidate Password :

  Candidate MAC     System Name       Device Type
  -------------   ----------------   -----------
  0060b0-e94300   DEFAULT_CONFIG       HP 8000M
  080009-918f80   DEFAULT_CONFIG       HP 4000M


 Actions->    Cancel      Edit      Save      Help

Use arrow keys to change field selection, <Space> to toggle field choices,
and <Enter> to go to Actions.
```

The Commander automatically selects an available switch number (SN). You have the option of assigning any other available number.

Candidate List

Figure 7-10. Example of Candidate List in Stack Management Screen

3. Either accept the displayed switch number or enter another available number. (The range is 0 - 15, with 0 reserved for the Commander.)

4. Use the downarrow key to move the cursor to the MAC Address field, then type the MAC address of the desired Candidate from the Candidate list in the lower part of the screen.

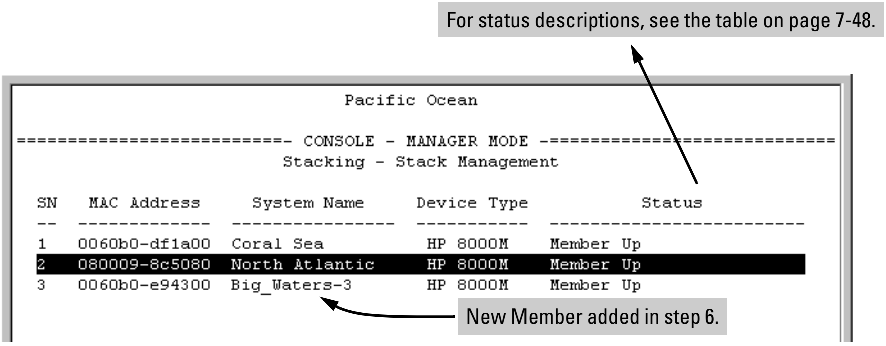
5. Do one of the following:

   - If the desired Candidate has a Manager password, press the downarrow key to move the cursor to the Candidate Password field, then type the password.

   - If the desired Candidate does not have a password, go to step 6.

6. Press **[Enter]** to return to the Actions line, then press **[S]** (for **Save**) to complete the Add process for the selected Candidate. You will then see a screen similar to the one in figure 7-11, below, with the newly added Member listed.

   Note: If the message **Unable to add stack member: Invalid Password** appears in the console menu's Help line, then you either omitted the Candidate's Manager password or incorrectly entered the Manager password.



For status descriptions, see the table on page 7-48.

```
                                Pacific Ocean

==========================- CONSOLE - MANAGER MODE -============================
                          Stacking - Stack Management

  SN   MAC Address     System Name     Device Type           Status
  --  -------------  ----------------  -----------  -------------------------
  1   0060b0-df1a00  Coral Sea           HP 8000M    Member Up
  2   080009-8c5080  North Atlantic      HP 8000M    Member Up
  3   0060b0-e94300  Big_Waters-3        HP 8000M    Member Up
```

New Member added in step 6.

Figure 7-11. Example of Stack Management Screen After New Member Added

Using the Commander's Menu To Move a Member From One Stack to Another. Where two or more stacks exist in the same subnet (broadcast domain), you can easily move a Member of one stack to another stack if the destination stack is not full. (If you are using VLANs in your stack environment, see "Stacking Operation with a Tagged VLAN" on page 7-46.) This procedure is nearly identical to manually adding a Candidate to a stack (page 7-19). (If the stack from which you want to move the Member has a Manager password, you will need to know the password to make the move.)

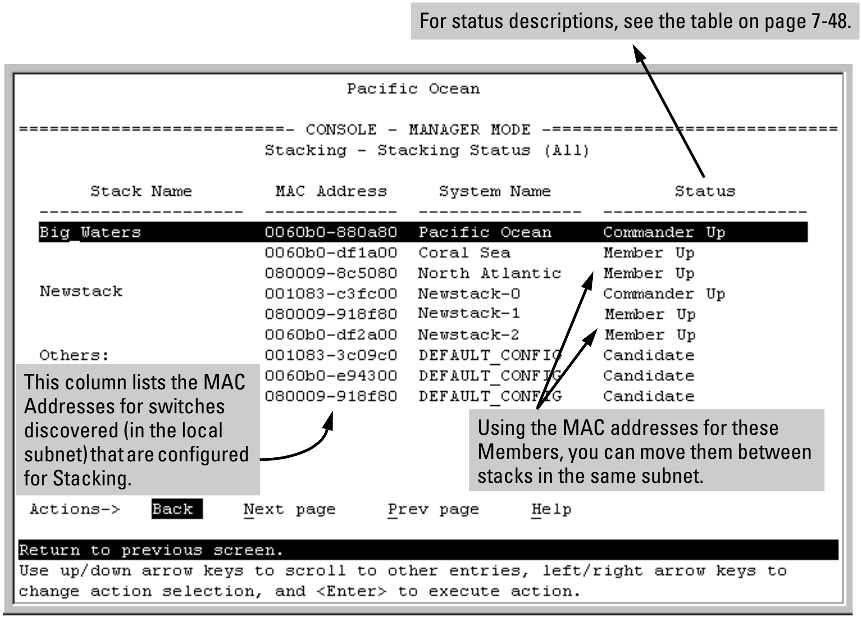
1. To move a Member from one stack to another, go to the Main Menu of the Commander in the destination stack and display the Stacking Menu by selecting

   9. Stacking...

2. To learn or verify the MAC address of the Member you want to move, display a listing of all Commanders, Members, and Candidates in the subnet by selecting:

   2. Stacking Status (All)

You will then see the Stacking Status (All) screen:



For status descriptions, see the table on page 7-48.

```
                              Pacific Ocean

========================- CONSOLE - MANAGER MODE -===========================
                      Stacking - Stacking Status (All)

      Stack Name         MAC Address     System Name           Status
  ------------------   -------------  ---------------  -------------------
  Big_Waters           0060b0-880a80  Pacific Ocean    Commander Up
                       0060b0-df1a00  Coral Sea        Member Up
                       080009-8c5080  North Atlantic   Member Up
  Newstack             001083-c3fc00  Newstack-0       Commander Up
                       080009-918f80  Newstack-1       Member Up
                       0060b0-df2a00  Newstack-2       Member Up
  Others:              001083-3c09c0  DEFAULT_CONFIG   Candidate
                       0060b0-e94300  DEFAULT_CONFIG   Candidate
                       080009-918f80  DEFAULT_CONFIG   Candidate



 Actions->   Back     Next page     Prev page     Help

Return to previous screen.
Use up/down arrow keys to scroll to other entries, left/right arrow keys to
change action selection, and <Enter> to execute action.
```

This column lists the MAC Addresses for switches discovered (in the local subnet) that are configured for Stacking.

Using the MAC addresses for these Members, you can move them between stacks in the same subnet.

Figure 7-12. Example of How the Stacking Status (All) Screen Helps You Find Member MAC Addresses

3. In the Stacking Status (All) screen, find the Member switch that you want to move and note its MAC address, then press **[B]** (for **Back**) to return to the Stacking Menu.

4. Display the Commander's Stack Management screen by selecting

   4. Stack Management

   (For an example of this screen, see figure 7-9 on page 7-19.)

5. Press **[A]** (for **Add**) to add the Member. You will then see a screen listing any available candidates. (See figure 7-10 on page 7-20.) Note that you will not see the switch you want to add because it is a Member of another stack and not a Candidate.)

6. Either accept the displayed switch number or enter another available number. (The range is 0 - 15, with 0 reserved for the Commander.)

7. Use the downarrow key to move the cursor to the MAC Address field, then type the MAC address of the desired Member you want to move from another stack.

8. Do one of the following:
   - If the stack containing the Member you are moving has a Manager password, press the downarrow key to select the Candidate Password field, then type the password.
   - If the stack containing the Member you want to move does not have a password, go to step 9.
9. Press **[Enter]** to return to the Actions line, then press **[S]** (for **Save**) to complete the Add process for the selected Member. You will then see a screen similar to the one in figure 7-9 on page 7-19, with the newly added Member listed.

Note: If the message **Unable to add stack member: Invalid Password** appears in the console menu's Help line, then you either omitted the Manager password for the stack containing the Member or incorrectly entered the Manager password.

You can "push" a Member from one stack to another by going to the Member's interface and entering the MAC address of the destination stack Commander in the Member's **Commander MAC Address** field. Using this method moves the Member to another stack without a need for knowing the Manager password in that stack, but also blocks access to the Member from the original Commander.

Using the Commander's Menu To Remove a Stack Member. These rules affect removals from a stack:

- When a Candidate becomes a Member, its **Auto Join** parameter is automatically set to **No**. This prevents the switch from automatically rejoining a stack as soon as you remove it from the stack.
- When you use the Commander to remove a switch from a stack, the switch rejoins the Candidate pool for your IP subnet (broadcast domain), with **Auto Join** set to **No**.
- When you remove a Member from a stack, it frees the previously assigned switch number (**SN**), which then becomes available for assignment to another switch that you may subsequently add to the stack. The default switch number used for an add is the lowest unassigned number in the Member range (1 - 15; 0 is reserved for the Commander).

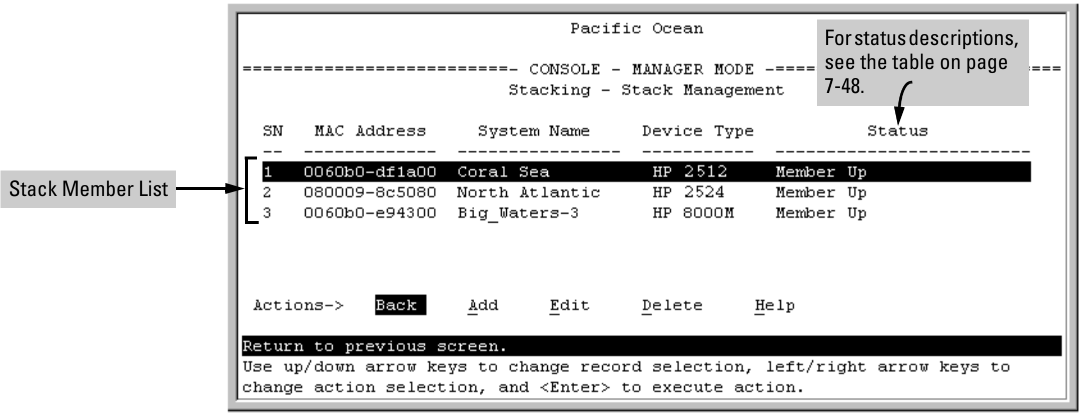
To remove a Member from a stack, use the Stack Management screen.

1. From the Main Menu, select:

   9. Stacking...

      4. Stack Management

   You will then see the Stack Management screen:



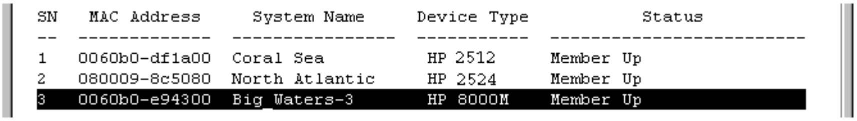
```
                            Pacific Ocean

=========================- CONSOLE - MANAGER MODE -====                    ===
                         Stacking - Stack Management

  SN   MAC Address     System Name      Device Type           Status
  --   -------------   ----------------  -----------  -------------------------
  1    0060b0-df1a00  Coral Sea          HP 2512      Member Up
  2    080009-8c5080  North Atlantic     HP 2524      Member Up
  3    0060b0-e94300  Big_Waters-3       HP 8000M     Member Up




 Actions->    Back      Add      Edit      Delete     Help

Return to previous screen.
Use up/down arrow keys to change record selection, left/right arrow keys to
change action selection, and <Enter> to execute action.
```

For status descriptions, see the table on page 7-48.

Stack Member List

Figure 7-13. Example of Stack Management Screen with Stack Members Listed

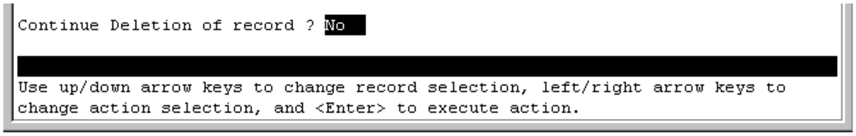
2. Use the downarrow key to select the Member you want to remove from the stack.



```
  SN   MAC Address     System Name      Device Type           Status
  --   -------------   ----------------  -----------  -------------------------
  1    0060b0-df1a00  Coral Sea          HP 2512      Member Up
  2    080009-8c5080  North Atlantic     HP 2524      Member Up
  3    0060b0-e94300  Big_Waters-3       HP 8000M     Member Up
```

Figure 7-14. Example of Selecting a Member for Removal from the Stack

3. Type **[D]** (for **Delete**) to remove the selected Member from the stack. You will then see the following prompt:



```
Continue Deletion of record ? No


Use up/down arrow keys to change record selection, left/right arrow keys to
change action selection, and <Enter> to execute action.
```

Figure 7-15. The Prompt for Completing the Deletion of a Member from the Stack

4. To continue deleting the selected Member, press the Space bar once to select **Yes** for the prompt, then press **[Enter]** to complete the deletion. The Stack Management screen updates to show the new stack Member list.

Using the Commander To Access Member Switches for Configuration Changes and Monitoring Traffic

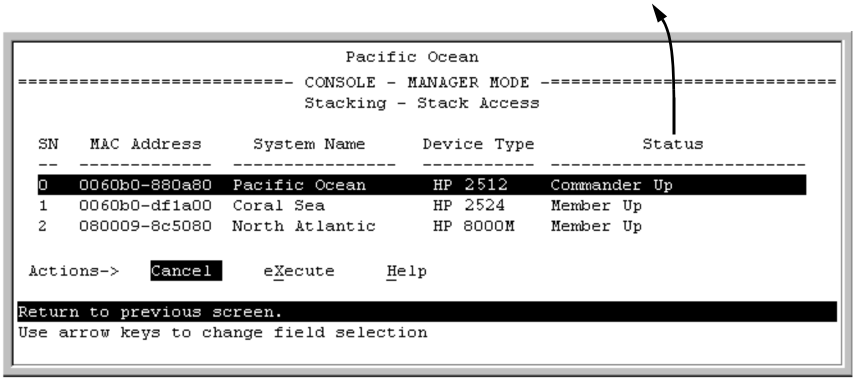
After a Candidate becomes a stack Member, you can use that stack's Commander to access the Member's console interface for the same configuration and monitoring that you would do through a Telnet or direct-connect access.

1. From the Main Menu, select:

   9. Stacking...
   5. Stack Access

   You will then see the Stack Access screen:

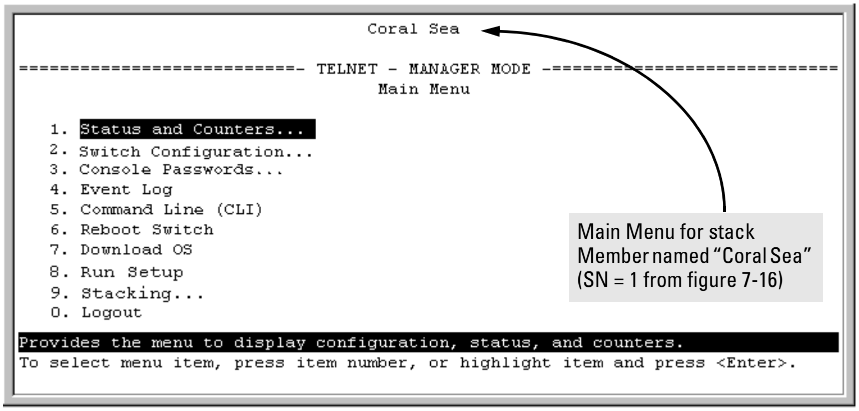
For status descriptions, see the table on page 7-48.



```
                                    Pacific Ocean
==========================- CONSOLE - MANAGER MODE -=============================
                                Stacking - Stack Access

  SN   MAC Address     System Name      Device Type          Status
  --   -------------   ----------------  -----------  -------------------------
  0    0060b0-880a80  Pacific Ocean       HP 2512     Commander Up
  1    0060b0-df1a00  Coral Sea           HP 2524     Member Up
  2    080009-8c5080  North Atlantic      HP 8000M    Member Up

 Actions->   Cancel     eXecute     Help

Return to previous screen.
Use arrow keys to change field selection
```

Figure 7-16. Example of the Stack Access Screen

Use the downarrow key to select the stack Member you want to access, then press **[X]** (for **eXecute**) to display the console interface for the selected Member. For example, if you selected switch number 1 (system name: **Coral Sea**) in figure 7-16 and then pressed **[X]**, you would see the Main Menu for the switch named Coral Sea.



```
                                Coral Sea

==========================- TELNET - MANAGER MODE -============================
                                Main Menu

     1. Status and Counters...
     2. Switch Configuration...
     3. Console Passwords...
     4. Event Log
     5. Command Line (CLI)
     6. Reboot Switch
     7. Download OS
     8. Run Setup
     9. Stacking...
     0. Logout

Provides the menu to display configuration, status, and counters.
To select menu item, press item number, or highlight item and press <Enter>.
```

Main Menu for stack Member named "Coral Sea" (SN = 1 from figure 7-16)

Figure 7-17. The eXecute Command Displays the Console Main Menu for the Selected Stack Member

2. You can now make configuration changes and/or view status data for the selected Member in the same way that you would if you were directly connected or telnetted into the switch.

3. When you are finished accessing the selected Member, do the following to return to the Commander's Stack Access screen:

   a. Return to the Member's Main Menu.

   b. Press **[0]** (for Logout), then **[Y]** (for Yes).

   c. Press **[Return]**.

   You should now see the Commander's Stack Access screen. (For an example, see figure 7-16 on page 7-25.)

Converting a Commander or Member to a Member of Another Stack

When moving a commander, the following procedure returns the stack members to Candidate status (with Auto-Join set to "**No**") and converts the stack Commander to a Member of another stack. When moving a member, the procedure simply pulls a Member out of one stack and pushes it into another.

1. From the Main Menu of the switch you want to move, select

   9. Stacking

2. To determine the MAC address of the destination Commander, select

   2. Stacking Status (All)

3. Press **[B]** (for **Back**) to return to the Stacking Menu.
4. To display Stack Configuration menu for the switch you are moving, select

   3. Stack Configuration

5. Press **[E]** (for **Edit**) to select the Stack State parameter.
6. Use the Space bar to select **Member**, then press [↓] to move to the **Commander MAC Address** field.
7. Enter the MAC address of the destination Commander and press **[Enter]**.
8. Press **[S]** (for **Save**).

Monitoring Stack Status

Using the stacking options in the menu interface for any switch in a stack, you can view stacking data for that switch or for all stacks in the subnet (broadcast domain). (If you are using VLANs in your stack environment, see "Stacking Operation with a Tagged VLAN" on page 7-46.) This can help you in such ways as determining the stacking configuration for individual switches, identifying stack Members and Candidates, and determining the status of individual switches in a stack. See table 7-5 on page 7-27.

Table 7-5. Stack Status Environments

Screen Name	Commander	Member	Candidate
Stack Status (This Switch)	• Commander's stacking configuration • Data on stack Members: – Switch Number – MAC Address – System Name – Device Type – Status	• Member's stacking configuration • Member Status • Data identifying Member's Commander: – Commander Status – Commander IP Address – Commander MAC Address	Candidate's stacking configuration
Stack Status (All)	Lists devices by stack name or Candidate status (if device is not a stack Member). Includes: • Stack Name • MAC Address • System Name • Status	Same as for Commander.	Same as for Commander.

Using Any Stacked Switch To View the Status for All Switches with Stacking Enabled. This procedure displays the general status of all switches in the IP subnet (broadcast domain) that have stacking enabled.

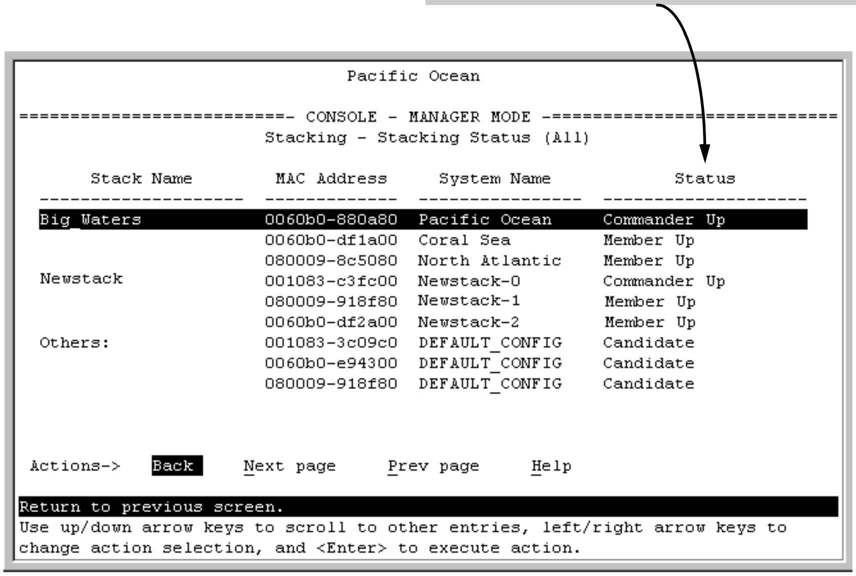
1. Go to the console Main Menu for any switch configured for stacking and select:

   9. Stacking ...

   2. Stacking Status (All)

   You will then see a Stacking Status screen similar to the following:

For status descriptions, see the table on page 7-48.



```
                                Pacific Ocean

==========================- CONSOLE - MANAGER MODE -============================
                        Stacking - Stacking Status (All)

      Stack Name          MAC Address      System Name           Status
  -------------------- -------------  ----------------  --------------------
  Big_Waters           0060b0-880a80  Pacific Ocean     Commander Up
                       0060b0-df1a00  Coral Sea         Member Up
                       080009-8c5080  North Atlantic    Member Up
  Newstack             001083-c3fc00  Newstack-0        Commander Up
                       080009-918f80  Newstack-1        Member Up
                       0060b0-df2a00  Newstack-2        Member Up
  Others:              001083-3c09c0  DEFAULT_CONFIG    Candidate
                       0060b0-e94300  DEFAULT_CONFIG    Candidate
                       080009-918f80  DEFAULT_CONFIG    Candidate



 Actions->   Back     Next page     Prev page     Help

Return to previous screen.
Use up/down arrow keys to scroll to other entries, left/right arrow keys to
change action selection, and <Enter> to execute action.
```

Figure 7-18. Example of Stacking Status for All Detected Switches Configured for Stacking

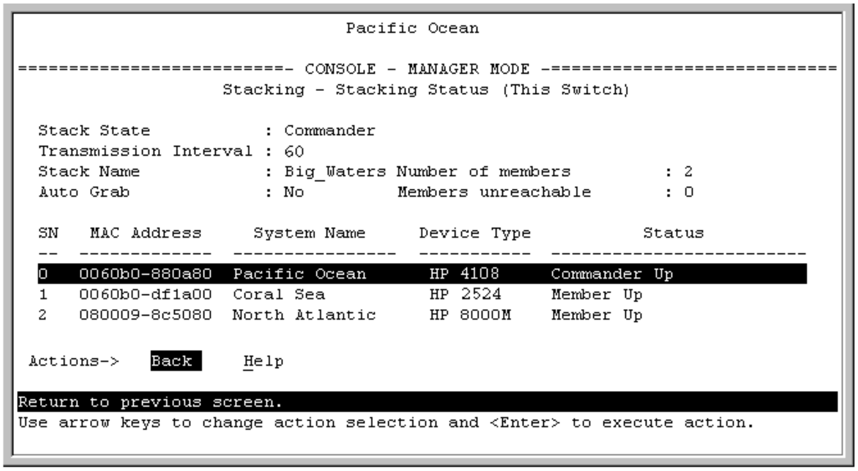
Viewing Commander Status. This procedure displays the Commander and stack configuration, plus information identifying each stack member.

To display the status for a Commander, go to the console Main Menu for the switch and select:

9. Stacking ...

1. Stacking Status (This Switch)

You will then see the Commander's Stacking Status screen:



```
                                 Pacific Ocean

==========================- CONSOLE - MANAGER MODE -============================
                     Stacking - Stacking Status (This Switch)

  Stack State           : Commander
  Transmission Interval : 60
  Stack Name            : Big_Waters Number of members         : 2
  Auto Grab             : No         Members unreachable       : 0

  SN   MAC Address     System Name      Device Type          Status
  --  -------------   ----------------  -----------  -------------------------
  0   0060b0-880a80  Pacific Ocean       HP 4108     Commander Up
  1   0060b0-df1a00  Coral Sea           HP 2524     Member Up
  2   080009-8c5080  North Atlantic      HP 8000M    Member Up

 Actions->   Back    Help

Return to previous screen.
Use arrow keys to change action selection and <Enter> to execute action.
```

Figure 7-19. Example of the Commander's Stacking Status Screen

Viewing Member Status. This procedure displays the Member's stacking information plus the Commander's status, IP address, and MAC address.

To display the status for a Member:

1. Go to the console Main Menu of the Commander switch and select

   9. Stacking ...

   5. Stack Access

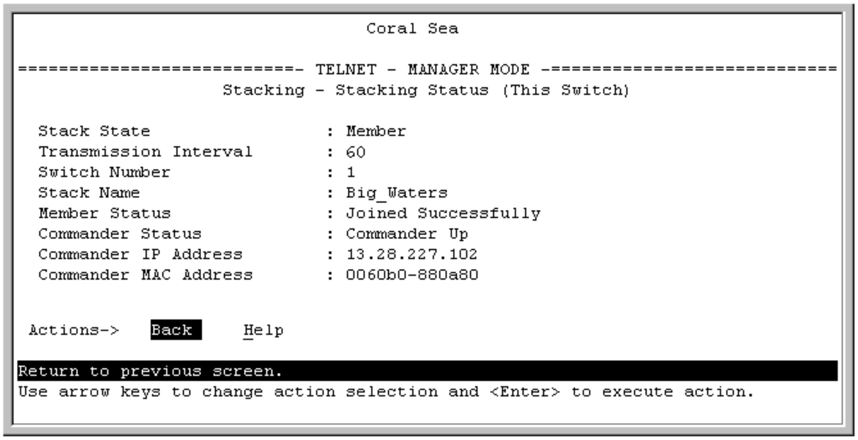
2. Use the downarrow key to select the Member switch whose status you want to view, then press **[X]** (for **eXecute**). You will then see the Main Menu for the selected Member switch.

3. In the Member's Main Menu screen, select

   9. Stacking ...

   1. Stacking Status (This Switch)

   You will then see the Member's Stacking Status screen:



```
                                    Coral Sea

============================- TELNET - MANAGER MODE -=============================
                      Stacking - Stacking Status (This Switch)

  Stack State                  : Member
  Transmission Interval        : 60
  Switch Number                : 1
  Stack Name                   : Big_Waters
  Member Status                : Joined Successfully
  Commander Status             : Commander Up
  Commander IP Address         : 13.28.227.102
  Commander MAC Address        : 0060b0-880a80


 Actions->   Back     Help

Return to previous screen.
Use arrow keys to change action selection and <Enter> to execute action.
```

Figure 7-20. Example of a Member's Stacking Status Screen

Viewing Candidate Status. This procedure displays the Candidate's stacking configuration.

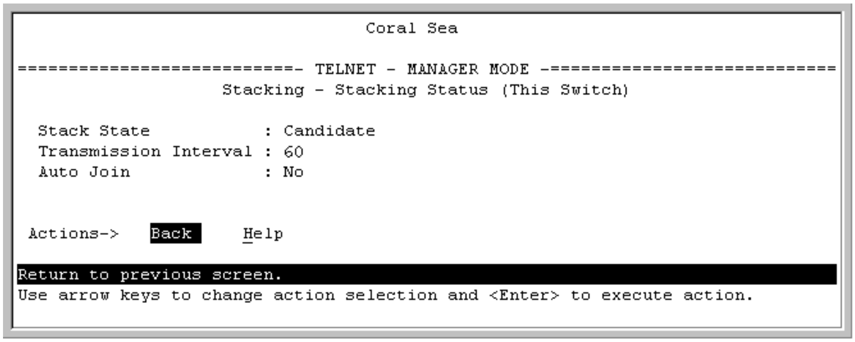
To display the status for a Candidate:

1. Use Telnet (if the Candidate has a valid IP address for your network) or a direct serial port connection to access the menu interface Main Menu for the Candidate switch and select

   9. Stacking ...

   1. Stacking Status (This Switch)

   You will then see the Candidate's Stacking Status screen:



```
                                    Coral Sea

============================- TELNET - MANAGER MODE -=============================
                      Stacking - Stacking Status (This Switch)

  Stack State           : Candidate
  Transmission Interval : 60
  Auto Join             : No


 Actions->   Back     Help

Return to previous screen.
Use arrow keys to change action selection and <Enter> to execute action.
```

Figure 7-21. Example of a Candidate's Stacking Screen

Using the CLI To View Stack Status and Configure Stacking

The CLI enables you to do all of the stacking tasks available through the menu interface.)

Table 7-6. CLI Commands for Configuring Stacking on a Switch

CLI Command	Operation
show stack **[candidates | view | all]**	**Commander:** Shows Commander's stacking configuration and lists the stack members and their individual status. **Member:** Lists Member's stacking configuration and status, and the status and the IP address and subnet mask of the stack Commander. Options: **candidates:** (Commander only) Lists stack Candidates. **view:** (Commander only) Lists current stack Members and their individual status. **all:** Lists all stack Commanders, Members and Candidates, with their individual status.
[no] stack	**Any Stacking-Capable Switch:** Enables or disables stacking on the switch. **Default:** Stacking Enabled
[no] stack commander *<stack name>*	**Candidate or Commander:** Converts a Candidate to a Commander or changes the stack name of an existing commander. "**No**" form eliminates named stack and returns Commander and stack Members to Candidate status with **Auto Join** set to **No**. "**No**" form prevents the switch from being discovered as a stacking-capable switch. **Default:** Switch Configured as a Candidate
[no] stack auto-grab	**Commander:** Causes Commander to automatically add to its stack any discovered Candidate in the subnet that does not have a Manager password and has **Auto-Join** set to **Yes**. **Default:** Disabled **Note:** If the Commander's stack already has 15 members, the Candidate cannot join until an existing member leaves the stack.

CLI Command	Operation
[no] stack member *<switch-num>* **mac-address** *<mac-addr>* **[password** *<password-str>***]**	**Commander:** Adds a Candidate to stack membership. "No" form removes a Member from stack membership. To easily determine the MAC address of a Candidate, use the **show stack candidates** command. To determine the MAC address of a Member you want to remove, use the **show stack view** command.The password (*password-str*) is required only when adding a Candidate that has a Manager password.
telnet *<1..15>* *Used In:* Commander Only	**Commander:** Uses the **SN** (switch number— assigned by the stack Commander) to access the console interface (menu interface or CLI) of a stack member. To view the list of **SN** assignments for a stack, execute the **show stack** command in the Commander's CLI.
[no] stack join *<mac-addr>*	**Candidate:** Causes the Candidate to join the stack whose Commander has the indicated MAC address. "No" form is used in a Member to remove it from the stack of the Commander having the specified address. **Member:** "Pushes" the member to another stack whose Commander has the indicated MAC address.
[no] stack auto-join	**Candidate:** Enables Candidate to automatically join the stack of any Commander in the IP subnet that has **Auto Grab** enabled, or disables **Auto-Join** in the candidate. **Default: Auto Join** enabled. **Note:** If the Candidate has a Manager password or if the available stack(s) already have the maximum of 15 Members, the automatic join will not occur.
stack transmission-interval	**All Stack Members:** specifies the interval in seconds for transmitting stacking discovery packets. **Default:** 60 seconds

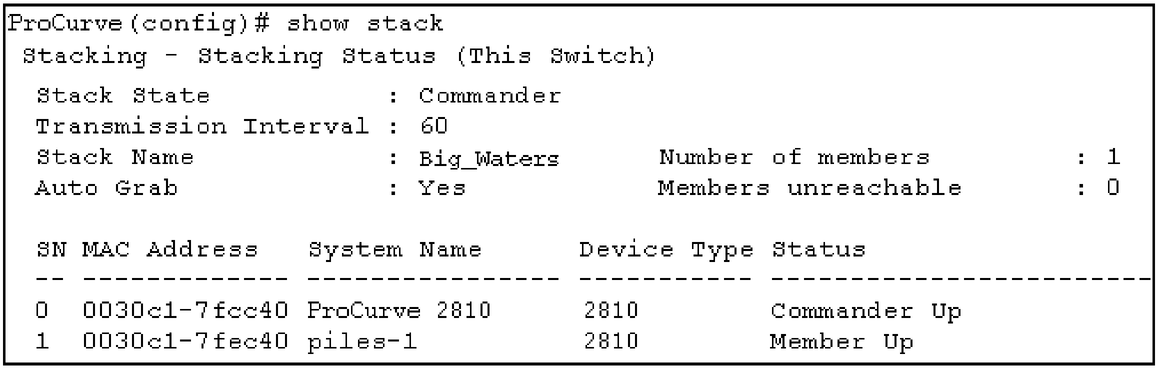
Using the CLI To View Stack Status

You can list the stack status for an individual switch and for other switches that have been discovered in the same subnet.

Syntax: show stack [candidates | view | all]

Viewing the Status of an Individual Switch. The following example illustrates how to use the CLI in a to display the stack status for that switch. In this case, the switch is in the default stacking configuration.

Syntax: show stack



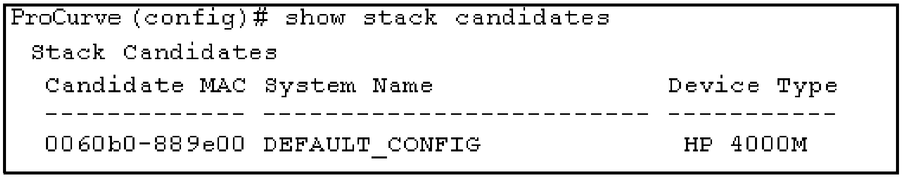
```
ProCurve(config)# show stack
 Stacking - Stacking Status (This Switch)

  Stack State           : Commander
  Transmission Interval : 60
  Stack Name            : Big_Waters      Number of members        : 1
  Auto Grab             : Yes             Members unreachable      : 0

  SN MAC Address    System Name      Device Type Status
  -- ------------- ---------------- ----------- ------------------------
  0  0030c1-7fcc40 ProCurve 2810     2810        Commander Up
  1  0030c1-7fec40 piles-1           2810        Member Up
```

Figure 7-22. Example of Using the Show Stack Command To List the Stacking Configuration for an Individual Switch

Viewing the Status of Candidates the Commander Has Detected.

This example illustrates how to list stack candidates the Commander has discovered in the ip subnet (broadcast domain).

Syntax: show stack candidates

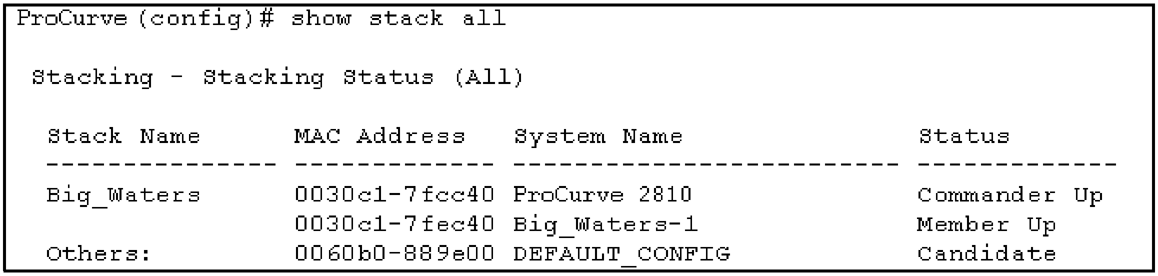


```
ProCurve(config)# show stack candidates
 Stack Candidates
  Candidate MAC System Name                Device Type
  ------------- -------------------------- -----------
  0060b0-889e00 DEFAULT_CONFIG              HP 4000M
```

Figure 7-23. Example of Using the Show Stack Candidates Command To List Candidates

Viewing the Status of all Stack-Enabled Switches Discovered in the IP Subnet. The next example lists all the stack-configured switches discovered in the IP subnet. Because the switch on which the **show stack all** command was executed is a candidate, it is included in the "Others" category.

Syntax: show stack all

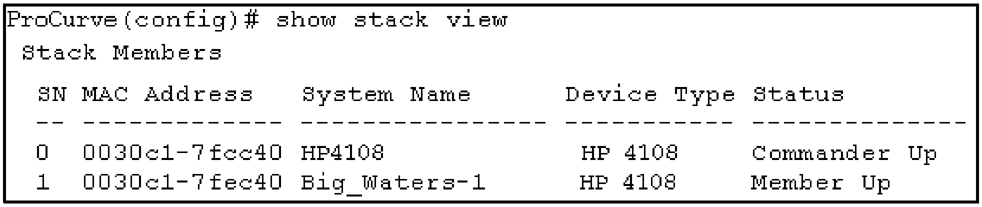


```
ProCurve(config)# show stack all

 Stacking - Stacking Status (All)

  Stack Name      MAC Address   System Name                Status
  --------------- ------------- -------------------------- -------------
  Big_Waters      0030c1-7fcc40 ProCurve 2810              Commander Up
                  0030c1-7fec40 Big_Waters-1               Member Up
  Others:         0060b0-889e00 DEFAULT_CONFIG             Candidate
```

Figure 7-24. Result of Using the Show Stack All Command To List Discovered Switches in the IP Subnet

Viewing the Status of the Commander and Current Members of the Commander's Stack. The next example lists all switches in the stack of the selected switch.

Syntax: show stack view



```
ProCurve(config)# show stack view
 Stack Members

  SN MAC Address   System Name      Device Type Status
  -- ------------- ---------------- ----------- --------------
  0  0030c1-7fcc40 HP4108            HP 4108    Commander Up
  1  0030c1-7fec40 Big_Waters-1      HP 4108    Member Up
```

Figure 7-25. Example of the Show Stack View Command To List the Stack Assigned to the Selected Commander

Using the CLI To Configure a Commander Switch

You can configure any stacking-enabled switch to be a Commander as long as the intended stack name does not already exist on the broadcast domain. (When you configure a Commander, you automatically create a corresponding stack.)

Before you begin configuring stacking parameters:

1. Configure IP addressing on the switch intended for stack commander and, if not already configured, on the primary VLAN. (For more on configuring IP addressing, see the chapter on IP addressing in the *Management and Configuration Guide*.)

Note

The primary VLAN must have an IP address in order for stacking to operate properly. For more on the primary VLAN, see “The Primary VLAN” on page 2-6.

2. Configure a Manager password on the switch intended for commander. (The Commander’s Manager password controls access to stack Members.) For more on passwords, see the local manager and operator password information in the *Access Security Guide* for your switch.

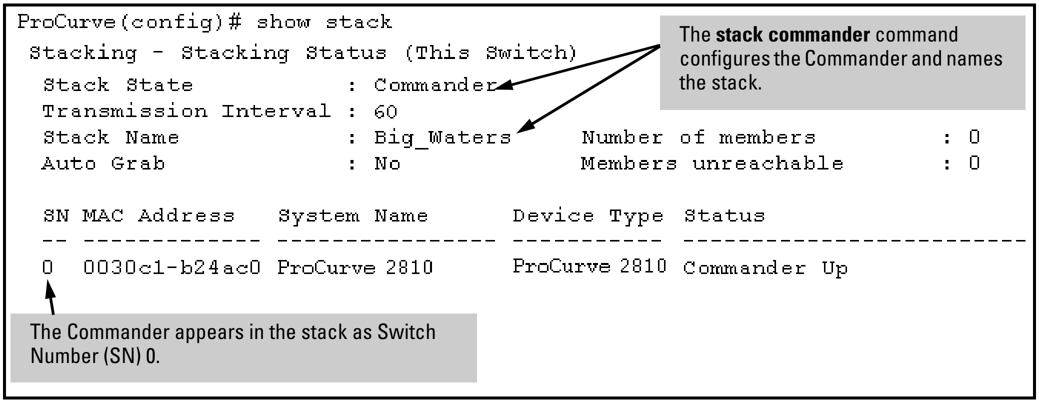
Configure the Stack Commander. Assigning a stack name to a switch makes it a Commander and automatically creates a stack.

Syntax: stack commander < *name-str* >

This example creates a Commander switch with a stack name of **Big_Waters**. (Note that if stacking was previously disabled on the switch, this command also enables stacking.)

```
ProCurve(config)# stack commander Big_Waters
```

As the following **show stack** display shows, the Commander switch is now ready to add members to the stack.



```
ProCurve(config)# show stack
 Stacking - Stacking Status (This Switch)
  Stack State           : Commander
  Transmission Interval : 60
  Stack Name            : Big_Waters     Number of members          : 0
  Auto Grab             : No             Members unreachable        : 0

  SN MAC Address    System Name      Device Type Status
  -- ------------- ---------------- ----------- -------------------------
  0  0030c1-b24ac0 ProCurve 2810     ProCurve 2810 Commander Up
```

The **stack commander** command configures the Commander and names the stack.

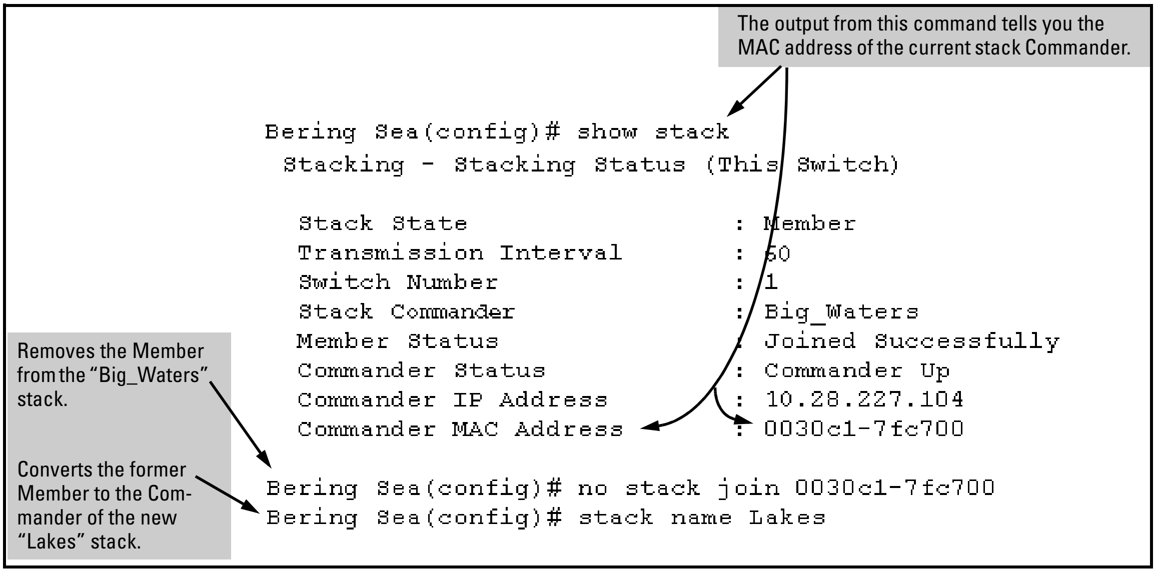
The Commander appears in the stack as Switch Number (SN) 0.

Figure 7-26. Example of the Commander's Show Stack Screen with Only the Commander Discovered

Using a Member's CLI to Convert the Member to the Commander of a New Stack. This procedure requires that you first remove the Member from its current stack, then create the new stack. If you do not know the MAC address for the Commander of the current stack, use **show stack** to list it.

Syntax: no stack
stack commander < *stack name* >

Suppose, for example, that a ProCurve switch named "Bering Sea" is a Member of a stack named "Big_Waters". To use the switch's CLI to convert it from a stack Member to the Commander of a new stack named "Lakes", you would use the following commands:



```
Bering Sea(config)# show stack
 Stacking - Stacking Status (This Switch)

  Stack State                    : Member
  Transmission Interval          : 60
  Switch Number                  : 1
  Stack Commander                : Big_Waters
  Member Status                  : Joined Successfully
  Commander Status               : Commander Up
  Commander IP Address           : 10.28.227.104
  Commander MAC Address          : 0030c1-7fc700

Bering Sea(config)# no stack join 0030c1-7fc700
Bering Sea(config)# stack name Lakes
```

The output from this command tells you the MAC address of the current stack Commander.

Removes the Member from the “Big_Waters” stack.

Converts the former Member to the Commander of the new “Lakes” stack.

Figure 7-27. Example of Using a Member’s CLI To Convert the Member to the Commander of a New Stack

Adding to a Stack or Moving Switches Between Stacks

You can add switches to a stack by adding discovered Candidates or by moving switches from other stacks that may exist in the same subnet. (You cannot add a Candidate that the Commander has not discovered.)

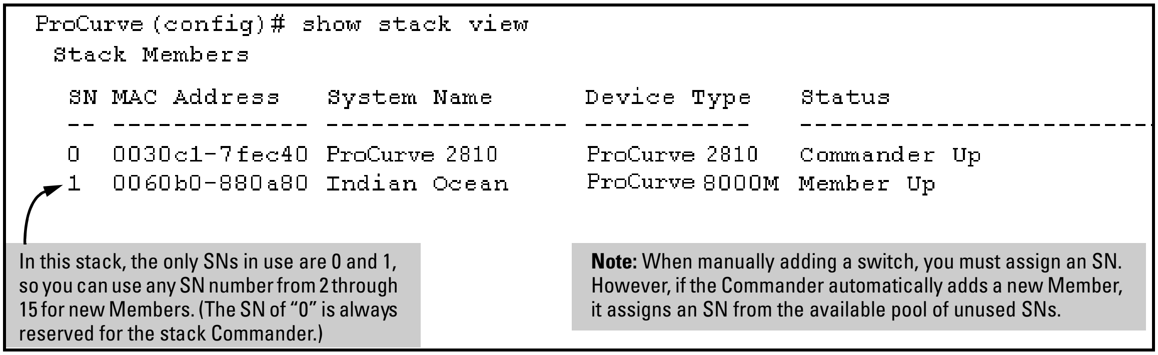
In its default configuration, the Commander’s **Auto-Grab** parameter is set to **No** to give you manual control over which switches join the stack and when they join. This prevents the Commander from automatically trying to add every Candidate it finds that has **Auto Join** set to **Yes** (the default for the Candidate).

(If you want any eligible Candidate to automatically join the stack when the Commander discovers it, configure **Auto Grab** in the Commander to **Yes**. When you do so, *any* Candidate discovered with **Auto Join** set to **Yes** (the default) and no Manager password will join the stack, up to the limit of 15 Members.)

Using the Commander's CLI To Manually Add a Candidate to the Stack. To manually add a candidate, you will use:

- A switch number (**SN**) to assign to the new member. Member SNs range from 1 to 15. To see which SNs are already assigned to Members, use **show stack view**. You can use any SN not included in the listing. (SNs are viewable only on a Commander switch.)
- The MAC address of the discovered Candidate you are adding to the stack. To see this data, use the **show stack candidates** listing.

For example:



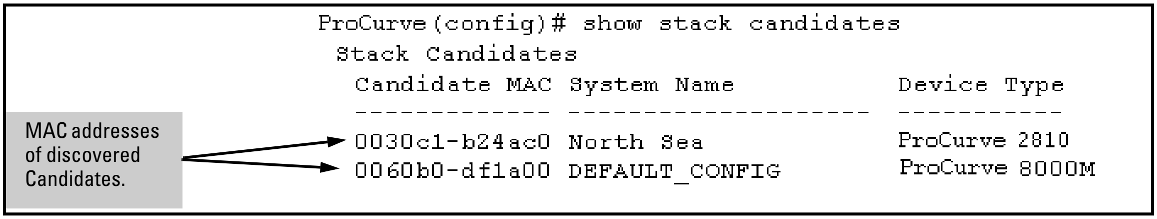
```
ProCurve(config)# show stack view
 Stack Members

  SN MAC Address   System Name      Device Type   Status
  -- ------------- ---------------- -----------   ------------------------
  0  0030c1-7fec40 ProCurve 2810    ProCurve 2810  Commander Up
  1  0060b0-880a80 Indian Ocean     ProCurve 8000M Member Up
```

In this stack, the only SNs in use are 0 and 1, so you can use any SN number from 2 through 15 for new Members. (The SN of "0" is always reserved for the stack Commander.)

Note: When manually adding a switch, you must assign an SN. However, if the Commander automatically adds a new Member, it assigns an SN from the available pool of unused SNs.

Figure 7-28. Example of How To Determine Available Switch Numbers (SNs)

To display all discovered Candidates with their MAC addresses, execute **show stack candidates** from the Commander's CLI. For example, to list the discovered candidates for the above Commander:



```
ProCurve(config)# show stack candidates
 Stack Candidates
  Candidate MAC System Name          Device Type
  ------------- -------------------- -----------
  0030c1-b24ac0 North Sea            ProCurve 2810
  0060b0-df1a00 DEFAULT_CONFIG       ProCurve 8000M
```

MAC addresses of discovered Candidates.

Figure 7-29. Example of How To Determine MAC Addresses of Discovered Candidates

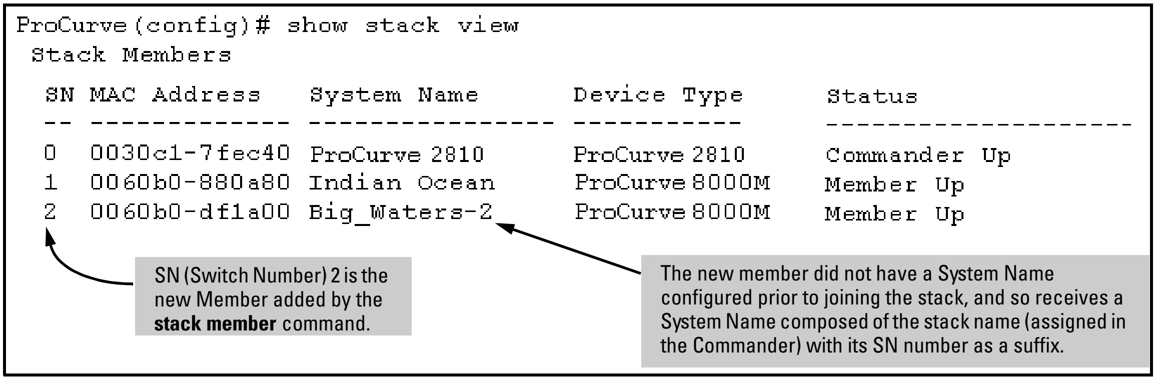
Knowing the available switch numbers (**SN**s) and Candidate MAC addresses, you can proceed to manually assign a Candidate to be a Member of the stack:

Syntax: stack member < *switch-number* > mac-address < *mac-addr* >
[password < *password-str* >]

For example, if the ProCurve 8000M in the above listing did not have a Manager password and you wanted to make it a stack Member with an **SN** of **2**, you would execute the following command:

```
ProCurve(config)# stack member 2 mac-address 0060b0-
df1a00
```

The **show stack view** command then lists the Member added by the above command:



```
ProCurve(config)# show stack view
 Stack Members

  SN MAC Address    System Name      Device Type      Status
  -- ------------- ---------------- -----------      --------------------
  0  0030c1-7fec40 ProCurve 2810    ProCurve 2810    Commander Up
  1  0060b0-880a80 Indian Ocean     ProCurve 8000M   Member Up
  2  0060b0-df1a00 Big_Waters-2     ProCurve 8000M   Member Up
```

SN (Switch Number) 2 is the new Member added by the **stack member** command.

The new member did not have a System Name configured prior to joining the stack, and so receives a System Name composed of the stack name (assigned in the Commander) with its SN number as a suffix.

Figure 7-30. Example Showing the Stack After Adding a New Member

Using Auto Join on a Candidate. In the default configuration, a Candidate's Auto Join parameter is set to "Yes", meaning that it will automatically join a stack if the stack's Commander detects the Candidate and the Commander's Auto Grab parameter is set to "Yes". You can disable Auto Join on a Candidate if you want to prevent automatic joining in this case. There is also the instance where a Candidate's Auto Join is disabled, for example, when a Commander leaves a stack and its members automatically return to Candidate status, or if you manually remove a Member from a stack. In this case, you may want to reset Auto Join to "Yes".

Status: [no] stack auto-join

```
ProCurve(config)# no stack auto-join
```

Disables Auto Join on a Candidate.

```
ProCurve(config)# stack auto-join
```

Enables Auto Join on a Candidate.

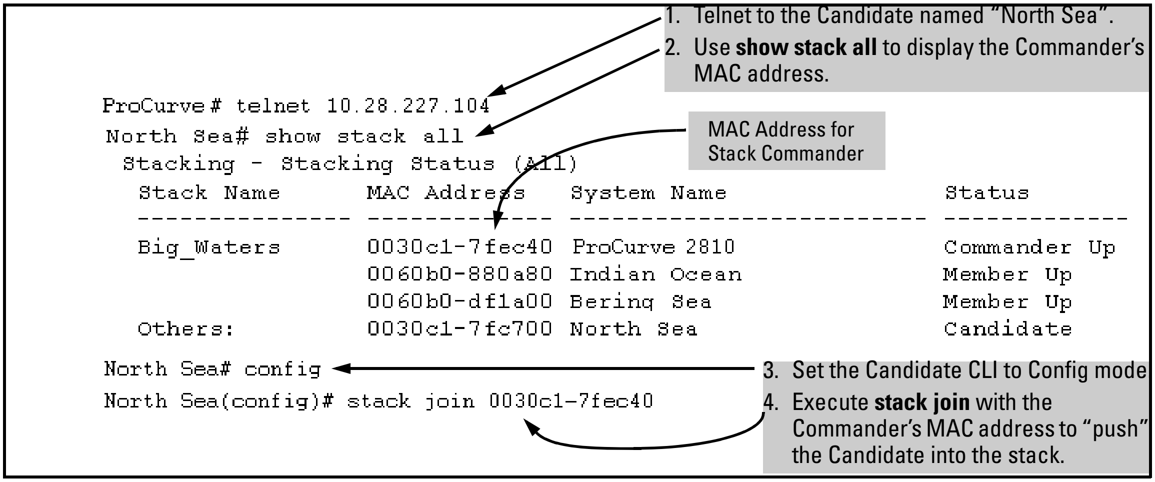
Using a Candidate CLI To Manually "Push" the Candidate Into a Stack . Use this method if any of the following apply:

- The Candidate's **Auto Join** is set to **Yes** (and you do not want to enable **Auto Grab** on the Commander) or the Candidate's **Auto Join** is set to **No**.
- Either you know the MAC address of the Commander for the stack into which you want to insert the Candidate, or the Candidate has a valid IP address and is operating in your network.

Syntax: stack join < *mac-addr* >

where: < *mac-addr* > is the MAC address of the Commander in the destination stack.

Use Telnet (if the Candidate has an IP address valid for your network) or a direct serial port connection to access the CLI for the Candidate switch. For example, suppose that a Candidate named "North Sea" with **Auto Join** off and a valid IP address of 10.28.227.104 is running on a network. You could Telnet to the Candidate, use **show stack all** to determine the Commander's MAC address, and then "push" the Candidate into the desired stack.



```
ProCurve# telnet 10.28.227.104
North Sea# show stack all
  Stacking - Stacking Status (All)
    Stack Name      MAC Address   System Name              Status
    --------------- ------------- ------------------------ -------------
    Big_Waters      0030c1-7fec40 ProCurve 2810            Commander Up
                    0060b0-880a80 Indian Ocean             Member Up
                    0060b0-df1a00 Bering Sea               Member Up
    Others:         0030c1-7fc700 North Sea                Candidate

North Sea# config
North Sea(config)# stack join 0030c1-7fec40
```

1. Telnet to the Candidate named "North Sea".
2. Use **show stack all** to display the Commander's MAC address.

MAC Address for Stack Commander

3. Set the Candidate CLI to Config mode
4. Execute **stack join** with the Commander's MAC address to "push" the Candidate into the stack.

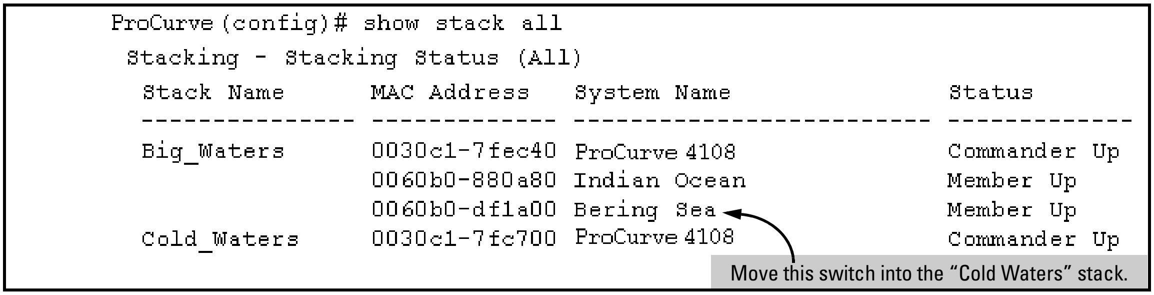
Figure 7-31. Example of "Pushing" a Candidate Into a Stack

To verify that the Candidate successfully joined the stack, execute **show stack all** again to view the stacking status.

Using the Destination Commander CLI To "Pull" a Member from Another Stack. This method uses the Commander in the destination stack to "pull" the Member from the source stack.

Syntax: stack member < *switch-number* >
mac-address < *mac-addr* >
[password < *password-str* >]

In the destination Commander, use **show stack all** to find the MAC address of the Member you want to pull into the destination stack. For example, suppose you created a new Commander with a stack name of "Cold_Waters" and you wanted to move a switch named "Bering Sea" into the new stack:



```
ProCurve(config)# show stack all
 Stacking - Stacking Status (All)
  Stack Name      MAC Address   System Name               Status
  --------------- ------------- ------------------------- -------------
  Big_Waters      0030c1-7fec40 ProCurve 4108             Commander Up
                  0060b0-880a80 Indian Ocean              Member Up
                  0060b0-df1a00 Bering Sea                Member Up
  Cold_Waters     0030c1-7fc700 ProCurve 4108             Commander Up
```

Move this switch into the "Cold Waters" stack.

Figure 7-32. Example of Stack Listing with Two Stacks in the Subnet

You would then execute the following command to pull the desired switch into the new stack:

```
ProCurve(config)# stack member 1 mac-address 0060b0-
df1a00
```

Where **1** is an unused switch number (**SN**).

Since a password is not set on the Candidate, a password is not needed in this example.

You could then use **show stack all** again to verify that the move took place.

Using a Member CLI To "Push" the Member into Another Stack. You can use the Member's CLI to "push" a stack Member into a destination stack if you know the MAC address of the destination Commander.

Syntax: **stack join *<mac-addr>***

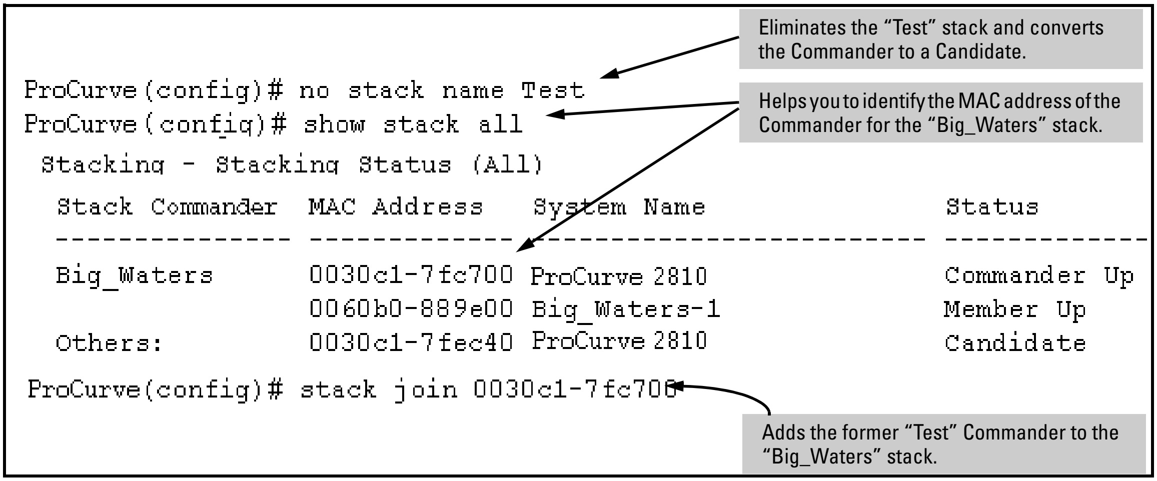
where: **< *mac-addr* >** is the MAC address of the Commander for the destination stack.

Converting a Commander to a Member of Another Stack. Removing the Commander from a stack eliminates the stack and returns its Members to the Candidate pool with **Auto Join** disabled.

Syntax: no stack name < *stack name*>
stack join < *mac-address* >

If you don't know the MAC address of the destination Commander, you can use **show stack all** to identify it.

For example, suppose you have a switch operating as the Commander for a temporary stack named "Test". When it is time to eliminate the temporary "Test" stack and convert the switch into a member of an existing stack named "Big_Waters", you would execute the following commands in the switch's CLI:



```
ProCurve(config)# no stack name Test
ProCurve(config)# show stack all
 Stacking - Stacking Status (All)

  Stack Commander  MAC Address    System Name               Status
  ---------------- ------------- ------------------------- ------------
  Big_Waters       0030c1-7fc700 ProCurve 2810             Commander Up
                   0060b0-889e00 Big_Waters-1              Member Up
  Others:          0030c1-7fec40 ProCurve 2810             Candidate

ProCurve(config)# stack join 0030c1-7fc700
```

Eliminates the "Test" stack and converts the Commander to a Candidate.

Helps you to identify the MAC address of the Commander for the "Big_Waters" stack.

Adds the former "Test" Commander to the "Big_Waters" stack.

Figure 7-33. Example of Command Sequence for Converting a Commander to a Member

Using the CLI To Remove a Member from a Stack

You can remove a Member from a stack using the CLI of either the Commander or the Member.

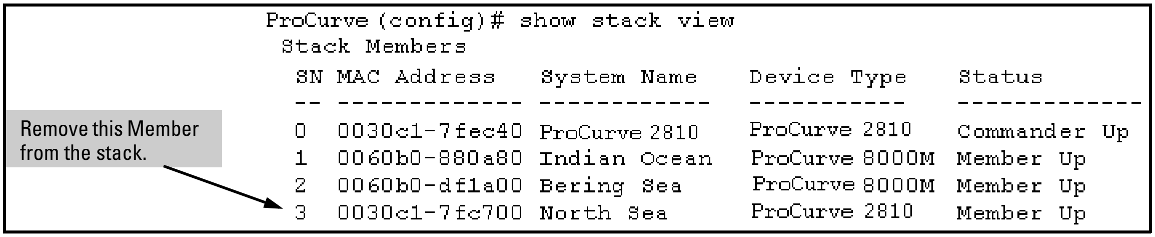
Note

When you remove a Member from a stack, the Member's **Auto Join** parameter is set to **No**.

Using the Commander CLI To Remove a Stack Member. This option requires the switch number (SN) and the MAC address of the switch to remove. (Because the Commander propagates its Manager password to all stack members, knowing the Manager password is necessary only for gaining access to the Commander.)

Syntax: [no] stack member <*switch-num*> mac-address <*mac-addr*>

Use **show stack view** to list the stack Members. For example, suppose that you wanted to use the Commander to remove the "North Sea" Member from the following stack:



```
ProCurve(config)# show stack view
 Stack Members

  SN MAC Address   System Name    Device Type    Status
  -- ------------- ------------   -----------    -------------
  0  0030c1-7fec40 ProCurve 2810  ProCurve 2810  Commander Up
  1  0060b0-880a80 Indian Ocean   ProCurve 8000M Member Up
  2  0060b0-df1a00 Bering Sea     ProCurve 8000M Member Up
  3  0030c1-7fc700 North Sea      ProCurve 2810  Member Up
```

Remove this Member from the stack.

Figure 7-34. Example of a Commander and Three Switches in a Stack

You would then execute this command to remove the "North Sea" switch from the stack:

```
ProCurve(config)# no stack member 3 mac-address 0030c1-
7fc700
```

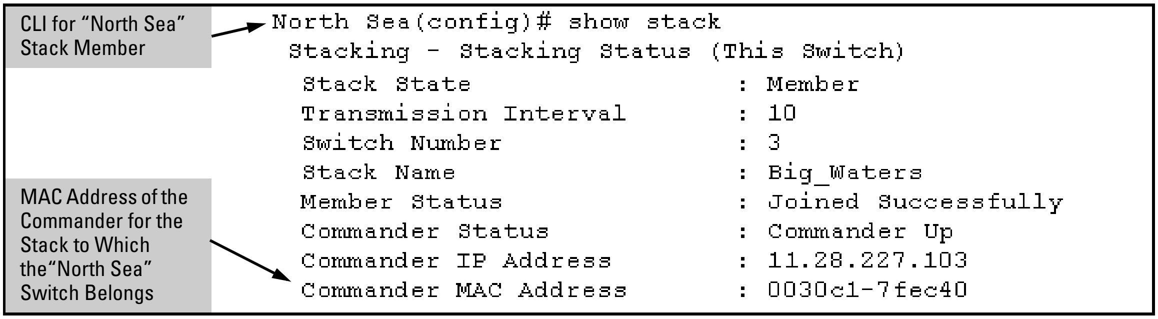
where:

- **3** is the "North Sea" Member's switch number (**SN**)
- **0030c1-7fc700** is the "North Sea" Member's MAC address

Using the Member's CLI To Remove the Member from a Stack.

Syntax: no stack join <*mac-addr*>

To use this method, you need the Commander's MAC address, which is available using the show stack command in the Member's CLI. For example:



```
North Sea(config)# show stack
 Stacking - Stacking Status (This Switch)

  Stack State                   : Member
  Transmission Interval         : 10
  Switch Number                 : 3
  Stack Name                    : Big_Waters
  Member Status                 : Joined Successfully
  Commander Status              : Commander Up
  Commander IP Address          : 11.28.227.103
  Commander MAC Address         : 0030c1-7fec40
```

CLI for "North Sea" Stack Member

MAC Address of the Commander for the Stack to Which the"North Sea" Switch Belongs

Figure 7-35. Example of How To Identify the Commander's MAC Address from a Member Switch

You would then execute this command in the “North Sea” switch’s CLI to remove the switch from the stack:

```
North Sea(config)# no stack join 0030c1-7fec40
```

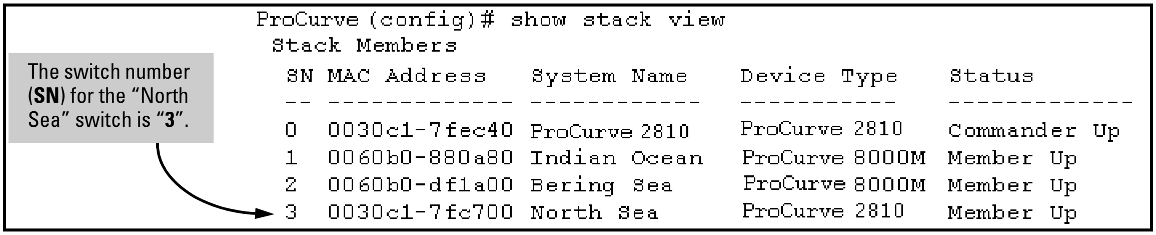
Using the CLI To Access Member Switches for Configuration Changes and Traffic Monitoring

After a Candidate becomes a Member, you can use the telnet command from the Commander to access the Member’s CLI or console interface for the same configuration and monitoring that you would do through a Telnet or direct-connect access from a terminal.

Syntax: telnet <switch-number>

where: unsigned integer is the switch number (**SN**) assigned by the Commander to each member (range: **1** - **15**).

To find the switch number for the Member you want to access, execute the **show stack view** command in the Commander’s CLI. For example, suppose that you wanted to configure a port trunk on the switch named “North Sea” in the stack named “Big_Waters”. Do do so you would go to the CLI for the “Big_Waters” Commander and execute show stack view to find the switch number for the “North Sea” switch:



```
ProCurve (config)# show stack view
 Stack Members
  SN MAC Address   System Name    Device Type   Status
  -- ------------- ------------   -----------   -------------
  0  0030c1-7fec40 ProCurve 2810  ProCurve 2810 Commander Up
  1  0060b0-880a80 Indian Ocean   ProCurve 8000M Member Up
  2  0060b0-df1a00 Bering Sea     ProCurve 8000M Member Up
  3  0030c1-7fc700 North Sea      ProCurve 2810 Member Up
```

The switch number (**SN**) for the “North Sea” switch is “**3**”.

Figure 7-36. Example of a Stack Showing Switch Number (SN) Assignments

To access the “North Sea” console, you would then execute the following **telnet** command:

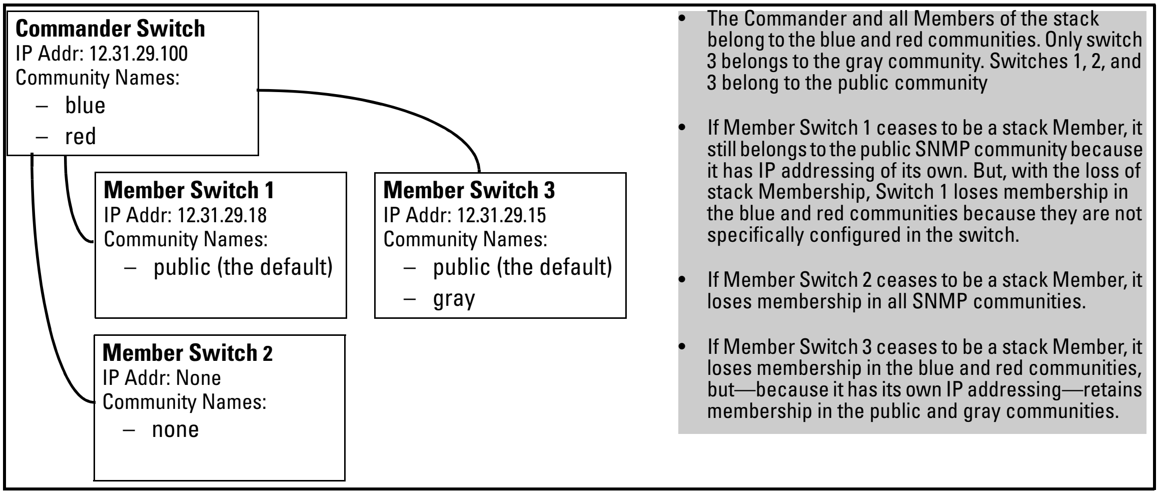
```
ProCurve(config)# telnet 3
```

You would then see the CLI prompt for the “North Sea” switch, allowing you to configure or monitor the switch as if you were directly connected to the console.

SNMP Community Operation in a Stack

Community Membership

In the default stacking configuration, when a Candidate joins a stack, it automatically becomes a Member of any SNMP community to which the Commander belongs, even though any community names configured in the Commander are not propagated to the Member's SNMP Communities listing. However, if a Member has its own (optional) IP addressing, it can belong to SNMP communities to which other switches in the stack, including the Commander, do not belong. For example:



- The Commander and all Members of the stack belong to the blue and red communities. Only switch 3 belongs to the gray community. Switches 1, 2, and 3 belong to the public community
- If Member Switch 1 ceases to be a stack Member, it still belongs to the public SNMP community because it has IP addressing of its own. But, with the loss of stack Membership, Switch 1 loses membership in the blue and red communities because they are not specifically configured in the switch.
- If Member Switch 2 ceases to be a stack Member, it loses membership in all SNMP communities.
- If Member Switch 3 ceases to be a stack Member, it loses membership in the blue and red communities, but—because it has its own IP addressing—retains membership in the public and gray communities.

Figure 7-37. Example of SNMP Community Operation with Stacking

SNMP Management Station Access to Members Via the Commander.

To use a management station for SNMP Get or Set access through the Commander's IP address to a Member, you must append **@sw<switch number>** to the community name. For example, in figure 7-37, you would use the following command in your management station to access Switch 1's MIB using the blue community:

```
snmpget< MIB variable > 10.31.29.100 blue@sw1
```

Note that because the gray community is only on switch 3, you could not use the Commander IP address for gray community access from the management station. Instead, you would access switch 3 directly using the switch's own IP address. For example:

```
snmpget < MIB variable > 10.31.29.15 gray
```

Note that in the above example (figure 7-37) you cannot use the public community through the Commander to access any of the Member switches. For example, you can use the public community to access the MIB in switches 1 and 3 by using their unique IP addresses. However, you must use the red or blue community to access the MIB for switch 2.

```
snmpget < MIB variable > 10.31.29.100 blue@sw2
```

Using the CLI To Disable or Re-Enable Stacking

In the default configuration, stacking is enabled on the switch. You can use the CLI to disable stacking on the switch at any time. Disabling stacking has the following effects:

- **Disabling a Commander:** Eliminates the stack, returns the stack Members to Candidates with **Auto Join** disabled, and changes the Commander to a stand-alone (nonstacking) switch. You must re-enable stacking on the switch before it can become a Candidate, Member, or Commander.
- **Disabling a Member:** Removes the Member from the stack and changes it to a stand-alone (nonstacking) switch. You must re-enable stacking on the switch before it can become a Candidate, Member, or Commander.
- **Disabling a Candidate:** Changes the Candidate to a stand-alone (nonstacking) switch.

Syntax: no stack (*Disables stacking on the switch.*)
stack (*Enables stacking on the switch.*)

Transmission Interval

All switches in the stack must be set to the same transmission interval to help ensure proper stacking operation. ProCurve recommends that you leave this parameter set to the default 60 seconds.

Syntax: stack transmission-interval < *seconds* >

Stacking Operation with Multiple VLANs Configured

Stacking uses the primary VLAN in a switch. In the factory-default configuration, the DEFAULT_VLAN is the primary VLAN. However, you can designate any VLAN configured in the switch as the primary VLAN. (See “The Primary VLAN” on page 2-6.)

When using stacking in a multiple-VLAN environment, the following criteria applies:

- Stacking uses only the primary VLAN on each switch in a stack.
- The primary VLAN can be tagged or untagged as needed in the stacking path from switch to switch.
- The same VLAN ID (VID) must be assigned to the primary VLAN in each stacked switch.

Web: Viewing and Configuring Stacking

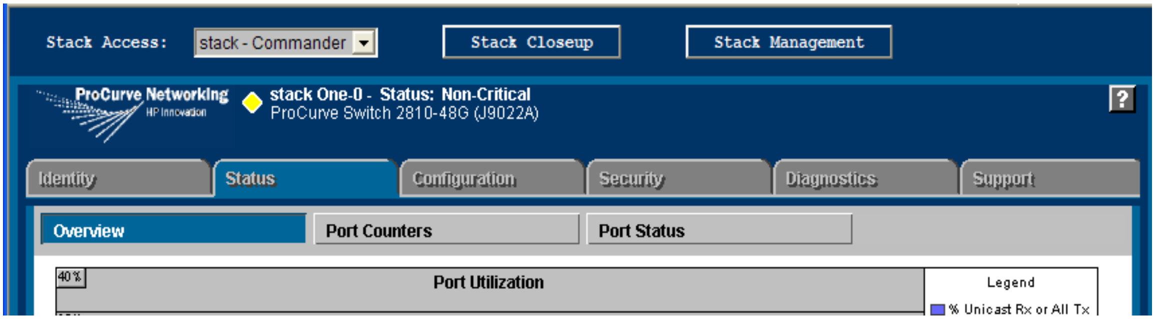


Figure 7-38. Example of the Web Browser Interface for a Commander

The web browser interface for a Commander appears as shown above. The interface for Members and Candidates appears the same as for a non-stacking switches.

To view or configure stacking on the web browser interface:

1. Click on the **Configuration** tab.
2. Click on **Stacking** to display the stacking configuration for an individual switch, and make any configuration changes you want for that switch.
3. Click on **Apply Changes** to save any configuration changes for the individual switch.
4. If the switch is a Commander, use the **Stack Closeup** and **Stack Management** buttons for viewing and using stack features.

To access the web-based Help provided for the switch, click on **[?]** in the web browser screen.

Status Messages

Stacking screens and listings display these status messages:

Message	Condition	Action or Remedy
Candidate Auto-join	Indicates a switch configured with Stack State set to **Candidate, Auto Join** set to **Yes** (the default), and no Manager password.	None required
Candidate	Candidate cannot automatically join the stack because one or both of the following conditions apply: • Candidate has **Auto Join** set to **No**. • Candidate has a Manager password.	Manually add the candidate to the stack.
Commander Down	Member has lost connectivity to its Commander.	Check connectivity between the Commander and the Member.
Commander Up	The Member has stacking connectivity with the Commander.	None required.
Mismatch	This may be a temporary condition while a Candidate is trying to join a stack. If the Candidate does not join, then stack configuration is inconsistent.	Initially, wait for an update. If condition persists, reconfigure the Commander or the Member.
Member Down	A Member has become detached from the stack. A possible cause is an interruption to the link between the Member and the Commander.	Check the connectivity between the Commander and the Member.
Member Up	The Commander has stacking connectivity to the Member.	None required.
Rejected	The Candidate has failed to be added to the stack.	The candidate may have a password. In this case, manually add the candidate. Otherwise, the stack may already be full. A stack can hold up to 15 Members (plus the Commander).

Index

ProCurve
Networking by HP

Technical information in this document
is subject to change without notice.

July 2007

Manual Part Number
5991-4733