Cisco MWR 2941 Mobile Wireless Edge Router Software Configuration Guide, Release 15.0(1)MR

March 2011

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Text Part Number: OL-23889-01

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The following information is for FCC compliance of Class A devices: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio-frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case users will be required to correct the interference at their own expense.

The following information is for FCC compliance of Class B devices: The equipment described in this manual generates and may radiate radio-frequency energy. If it is not installed in accordance with Cisco installation instructions, it may cause interference with radio and television reception. This equipment has been tested and found to comply with the limits for a Class B digital device in accordance with the specifications in part 15 of the FCC rules. These specifications are designed to provide reasonable protection against such interference in a residential installation. However, there is no guarantee that interference will not occur in a particular installation.

Modifying the equipment without Cisco written authorization may result in the equipment no longer complying with FCC requirements for Class A or Class B digital devices. In that event, your right to use the equipment may be limited by FCC regulations, and you may be required to correct any interference to radio or television communications at your own expense.

You can determine whether your equipment is causing interference by turning it off. If the interference stops, it was probably caused by the Cisco equipment or one of its peripheral devices. If the equipment causes interference to radio or television reception, try to correct the interference by using one or more of the following measures:

• Turn the television or radio antenna until the interference stops.

• Move the equipment to one side or the other of the television or radio.

• Move the equipment farther away from the television or radio.

• Plug the equipment into an outlet that is on a different circuit from the television or radio. (That is, make certain the equipment and the television or radio are on circuits controlled by different circuit breakers or fuses.)

Modifications to this product not authorized by Cisco Systems, Inc. could void the FCC approval and negate your authority to operate the product.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Cisco MWR 2941 Mobile Wireless Edge Router Software Configuration Guide, Release 15.0(1)MR

CONTENTS

CHAPTER 27

About This Guide

This section describes the objectives, audience, organization, and conventions of this software configuration guide. It contains the following sections:

Document Revision History

The Document Revision History table below records technical changes to this document.

Document Number	Date	Change Summary
OL-23889-01	March 2011	Updated for Release 15.0(2)MR.
OL-23889-01	December 2010	Initial release for Release 15.0(1)MR.

Objectives

This guide explains how to configure software features on the Cisco MWR 2941-DC and MWR 2941-DC-A routers. Unless otherwise stated, features described in this guide apply to both the Cisco MWR 2941-DC and the Cisco MWR 2941-DC-A.

Audience

This publication is for the person responsible for configuring the router. This guide is intended for the following audiences:

- Customers with technical networking background and experience

- System administrators who are familiar with the fundamentals of router-based internetworking, but who may not be familiar with Cisco IOS software
- System administrators who are responsible for installing and configuring internetworking equipment, and who are familiar with Cisco IOS software

Organization

The major sections of this software configuration guide are listed in the following table:

Chapter	Title	Description
	About This Guide	
Chapter 1	Cisco MWR 2941 Router Overview	Describes the purpose of the Cisco MWR 2941 router and its unique software features
Chapter 2	Cisco IOS Software Basics	Describes what you need to know about the Cisco IOS software
Chapter 3	First-Time Configuration	Describes the actions to take before turning on your router for the first time
Chapter 4	Configuring Gigabit Ethernet Interfaces	Describes how to configure Gigabit Ethernet interfaces on the router
Chapter 5	Configuring Layer 2 Interfaces	Describes how to configure Layer 2 interfaces on the router
Chapter 6	Configuring HWIC-D-9ESW Interfaces	Describes how to configure interfaces on the HWIC-D-9ESW
Chapter 7	Configuring VLANs	Describes how to configure VLANs on the router
Chapter 8	Configuring IEEE 802.1Q Tunneling and Layer 2 Protocol Tunneling	Describes how to configure 802.1Q and L2TP on the router
Chapter 9	Configuring STP	Describes how to configure Spanning Tree Protocol (STP)
Chapter 10	Configuring MSTP	Describes how to configure Multiple STP on the router
Chapter 11	Configuring Optional Spanning-Tree Features	Describes how to configure additional STP features
Chapter 12	Managing the MAC Address Table	Describes how to manage the MAC address table
Chapter 13	Configuring Cisco Express Forwarding	Describes how to configure Cisco Express Forwarding
Chapter 14	Configuring Resilient Ethernet Protocol	Describes how to configure Resilient Ethernet Protocol
Chapter 15	Configuring Ethernet OAM, CFM, and E-LMI	Describes how to configure carrier Ethernet features including Ethernet OAM, CFM, and E–LMI
Chapter 16	Configuring Clocking and Timing	Describes how to configure clocking and timing features

Chapter	Title	Description
Chapter 17	Configuring Synchronous Ethernet ESMC and SSM	Describes how to configure ESMC and SSM
Chapter 18	Configuring MLPPP Backhaul	Describes how to configure MLPPP backhaul
Chapter 19	Configuring Multiprotocol Label Switching	Describes how to configure MPLS
Chapter 20	Configuring Routing Protocols	Describes how to configure routing
Chapter 21	Configuring Bidirectional Forwarding Detection	Describes how to configure BFD
Chapter 22	Configuring Pseudowire	Describes how to configure pseudowire
Chapter 23	Configuring MPLS VPNs	Describes how to configure MPLS VPNs, also known as L3VPNs
Chapter 24	Configuring Quality of Service	Describes how to configure QoS features on the router
Chapter 25	Configuring Link Noise Monitor	Describes how to configure Link Noise Monitor
Chapter 26	Configuring Cisco Discovery Protocol	Describes how to configure Cisco Discovery Protocol
Chapter 27	Monitoring and Managing the Cisco MWR 2941 Router	Describes how to configure monitoring and network management features

Conventions

This publication uses the following conventions to convey instructions and information.

Convention	Description
boldface font	Commands and keywords.
italic font	Variables for which you supply values.
[ ]	Keywords or arguments that appear within square brackets are optional.
{x | y | z}	A choice of required keywords appears in braces separated by vertical bars. You must select one.
`screen font`	Examples of information displayed on the screen.
`boldface screen font`	Examples of information the user enters.
< >	Nonprinting characters, for example passwords, appear in angle brackets.
[ ]	Default responses to system prompts appear in square brackets.

Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the manual.

Timesaver Means *the described action saves time*. You can save time by performing the action described in the paragraph.

Tip Means *the following information will help you solve a problem*. The tips information might not be troubleshooting or even an action, but could be useful information, similar to a Timesaver.

Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

Related Documentation

The following list includes documentation related to your product by implementation.

- Cisco MWR 2941 Mobile Wireless Edge Router Documents
  - *Cisco MWR 2941 Router Command Reference, Release 15.0(1)MR*
  - *Cisco MWR 2941 Mobile Wireless Edge Router Hardware Installation Guide*
  - *Regulatory Compliance and Safety Information for the Cisco MWR 2941 Routers*
- Cisco Interface Cards Installation Guides
  - *Quick Start Guide: Interface Cards*
  - Cisco Interface Cards Installation Guide
- Release Notes
  - *Release Notes for Cisco MWR 2941 Mobile Wireless Edge Router for Cisco IOS Release 15.0(1)MR*

Note To obtain the latest information, access the online documentation.

Obtaining Documentation, Obtaining Support, and Security Guidelines

For information on obtaining documentation, obtaining support, providing documentation feedback, security guidelines, and also recommended aliases and general Cisco documents, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

CHAPTER 1

Cisco MWR 2941 Router Overview

The Cisco MWR 2941 Mobile Wireless Router is cell-site access platforms specifically designed to aggregate and transport mixed-generation radio access network (RAN) traffic. The router is used at the cell site edge as a part of a 2G, 3G, or 4G radio access network (RAN). The Cisco MWR 2941 includes the following models:

- Cisco MWR 2941-DC
- Cisco MWR 2941-DC-A

The Cisco MWR 2941 router helps enable a variety of RAN solutions by extending IP connectivity to devices using Global System for Mobile Communications (GSM), General Packet Radio Service (GPRS), Node Bs using HSPA or LTE, base transceiver stations (BTSs) using Enhanced Data Rates for GSM Evolution (EDGE), Code Division Multiple Access (CDMA), CDMA-2000, EVDO, or WiMAX, and other cell-site equipment. It transparently and efficiently transports cell-site voice, data, and signaling traffic over IP using traditional T1/E1 circuits, including leased line, microwave, and satellite, as well as alternative backhaul networks, including Carrier Ethernet, DSL, Ethernet in the First Mile (EFM), and WiMAX. It also supports standards-based Internet Engineering Task Force (IETF) Internet protocols over the RAN transport network, including those standardized at the Third-Generation Partnership Project (3GPP) for IP RAN transport.

Custom designed for the cell site, the Cisco MWR 2941 features a small form factor, extended operating temperature, and cell-site DC input voltages.

Note The Cisco MWR 2941-DC and 2941-DC-A support the same features except for commands related to the 1PPS, 10Mhz, 2.048Mhz, and 1.544Mhz timing ports that are included on the 2941-DC-A. For more information, see the *Release Notes for Cisco MWR 2941-DC Mobile Wireless Edge Router for Cisco IOS Release 15.0(1)MR.*

Introduction

A typical RAN is composed of thousands of base transceiver stations (BTSs)/Node Bs, hundreds of base station controllers/radio network controllers (BSCs/RNCs), and several mobile switching centers (MSCs). The BTS/Node Bs and BSC/RNC are often separated by large geographic distances, with the BTSs/Node Bs located in cell sites uniformly distributed throughout a region, and the BSCs, RNCs, and MSCs located at suitably chosen Central Offices (CO) or mobile telephone switching offices (MTSO).

The traffic generated by a BTS/Node B is transported to the corresponding BSC/RNC across a network, referred to as the backhaul network, which is often a hub-and-spoke topology with hundreds of BTS/Node Bs connected to a BSC/RNC by point-to-point time division multiplexing (TDM) trunks. These TDM trunks may be leased-line T1/E1s or their logical equivalents, such as microwave links or satellite channels.

The following sections describe the features available on the Cisco MWR 2941:

- RAN Transport Solutions
- MLPPP Optimization Features
- Intelligent Cell Site IP Services

RAN Transport Solutions

The Cisco MWR 2941 Mobile Wireless Router supports a variety of RAN transport solutions, including the following:

- IP/Multiprotocol Label Switching (MPLS) RAN backhaul: Allows you to create a high-speed backhaul for a variety of traffic types, including GSM, CDMA, HSPA/LTE, CDMA, EVDO, and WiMAX networks.
- Cell-site operations support networks: Facilitates telemetry to cell sites for remote operations and network element management.
- Cell-site IP points of presence (POPs): Allows you to offer IP services and applications at cell sites.
- Carrier Ethernet features including Resilient Ethernet Protocol (REP), Ethernet Connectivity Fault Management (CFM), Ethernet Local Management Interface (E-LMI), and Ethernet Operations, Administration, and Maintenance (OAM)
- Network clocking features including PTP, pseudowire-based clocking, and synchronous Ethernet.
- Flexible backhaul transport including MLPPP over T1, E1, xDSL, and Ethernet

MLPPP Optimization Features

The Cisco MWR 2941 supports several features that improve the performance of Multilink Point-to-Point Protocol (MLPPP) connections and related applications such as PWE3 over MLPPP and IP over MLPPP.

Distributed Multilink Point-to-Point Protocol (dMLPPP) Offload

Distributed Multilink Point-to-Point Protocol (dMLPPP) allows you to combine T1 or E1 connections into a bundle that has the combined bandwidth of all of the connections in the bundle, providing improved capacity and CPU utilization over MLPPP. The dMLPPP offload feature improves the performance for traffic in dMLPPP applications such as PWE3 over MLPPP and IP over MLPPP by shifting processing of this traffic from the main CPU to the network processor.

The Cisco MWR 2941 supports up to four serial links per T1/E1 connection and up to 24 MLPPP bundles. You can use the fixed T1/E1 ports to create up to 64 MLPPP links; if you install two four-port T1/E1 HWICs, you can create up to 96 MLPPP links.

The MWR 2941 implementation of multilink (dMLPPP) uses interleaving to allow short, delay-sensitive packets to be transmitted within a predictable amount of time. Interleaving allows the MWR 2941 to interrupt the transmission of delay-insensitive packets in order to transmit delay-sensitive packets. You can also adjust the responsiveness of the MWR 2941 to delay-sensitive traffic by adjusting the maximum fragment size; this value determines the maximum delay that a delay-sensitive packet can encounter while the MWR 2941 transmits queued fragments of delay-insensitive traffic.

Multiclass MLPPP

The MWR 2941 implementation of dMLPPP also supports Multiclass MLPPP. Multiclass MLPPP is an extension to MLPPP functionality that allows you to divide traffic passing over a multilink bundle into several independently sequenced streams or classes. Each multiclass MLPPP class has a unique sequence number, and the receiving network peer processes each stream independently. The multiclass MLPPP standard is defined in RFC 2686.

The MWR 2941 supports the following multiclass MLPPP classes:

- Class 0- Data traffic that is subject to normal MLPPP fragmentation. Appropriate for non-delay-sensitive traffic.
- Class 1- Data traffic that can be interleaved but not fragmented. Appropriate for delay-sensitive traffic such as voice.

For instructions on how to configure MLPPP backhaul, see Chapter 18, "Configuring MLPPP Backhaul."

Note The Cisco MWR 2941 does not support some PPP and MLPPP options when the bundle is offloaded to the network processor; you can retain these options by disabling MLPPP and IPHC offloading for a given bundle. For more information, see MLPPP Offload, page 18-12.

Note The output for the **show ppp multilink** command for an offloaded MLPPP bundle differs from the output for a non-offloaded bundle. For more information, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.

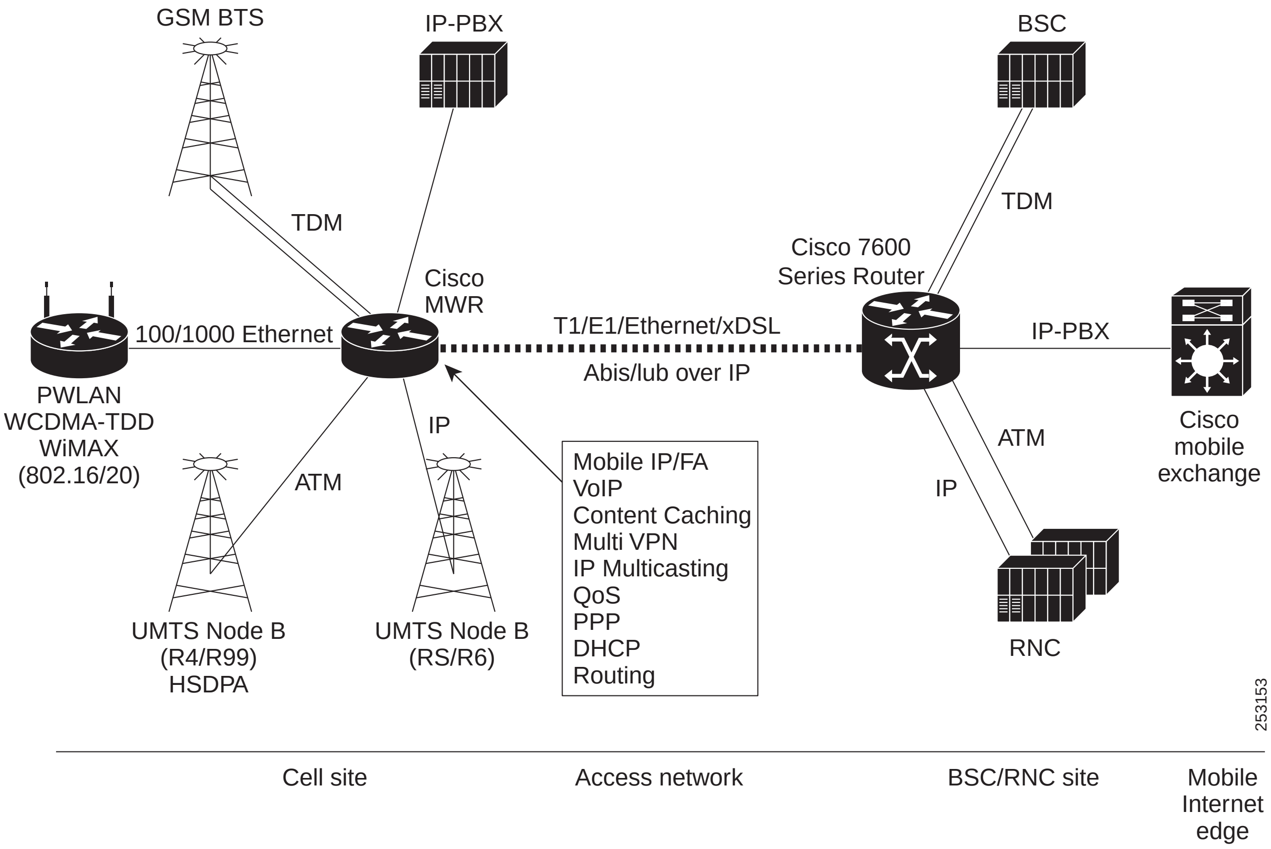
Intelligent Cell Site IP Services

The Cisco IP-RAN solutions allow you to deliver profit-enhancing services. This is achieved through the set of IP networking features supported in Cisco IOS software that extends to the cell site (see Figure 1-1 on page 1-4).

Cell Site Points-of-Presence

The cell site becomes a physical Point-of-Presence (POP) from which to offer hotspot services, or voice and wired ISP services, to nearby enterprises and residences. Because many cell sites are located in and around downtown areas, hotels, airports, and convention centers, they make attractive sites for co-locating public wireless LAN (PWLAN) access points and other wireless data overlays. Many of these wireless data radios are IP-based. IP networking features, like Mobile IP, VoIP, IP Multicast, VPN, and content caching, enable delivery of new revenue-generating services over these radios. The corresponding traffic "rides for free" on the spare backhaul bandwidth (Figure 1-1).

Figure 1-1 Cisco MWR 2941 Router in a Cell Site POP—Example



CHAPTER 2

Cisco IOS Software Basics

This chapter describes the basics of using the Cisco IOS software. Review this information before you configure the router using the command-line interface (CLI). This chapter includes the following sections:

- Getting Help, page 2-1
- Understanding Command Modes, page 2-2
- Undoing a Command or Feature, page 2-3
- Saving Configuration Changes, page 2-3

For additional information about using the Cisco IOS software, see the Configuration Fundamentals Configuration Guide, Cisco IOS Release 15.0S.

Getting Help

Use the question mark (?) and arrow keys to help you enter commands:

- For a list of available commands, enter a question mark:

  ```
  Router> ?
  ```

- To complete a command, enter a few known characters followed by a question mark (with no space):

  ```
  Router> s?
  ```

- For a list of command variables, enter the command followed by a space and a question mark:

  ```
  Router> show ?
  ```

- To redisplay a command that you previously entered, press the **Up Arrow** key. Continue to press the **Up Arrow** key to see more commands.

Understanding Command Modes

The Cisco IOS user interface is used in various command modes. Each command mode permits you to configure different components on your router. The commands available at any given time depend on which command mode you are in. Entering a question mark (**?**) at a prompt displays a list of commands available for that command mode. The following table lists the most common command modes.

Command Mode	Access Method	Router Prompt Displayed	Exit Method
User EXEC	Log in.	`Router>`	Use the **logout** command.
Privileged EXEC	From user EXEC mode, enter the **enable** command.	`Router#`	To exit to user EXEC mode, use the **disable**, **exit**, or **logout** command.
Global configuration	From the privileged EXEC mode, enter the **configure terminal** command.	`Router (config)#`	To exit to privileged EXEC mode, use the **exit** or **end** command, or press **Ctrl-Z**.
Interface configuration	From the global configuration mode, enter the **interface** *type number* command, such as **interface serial 0/0**.	`Router (config-if)#`	To exit to global configuration mode, use the **exit** command. To exit directly to privileged EXEC mode, press **Ctrl-Z**.

Timesaver Each command mode restricts you to a subset of commands. If you have trouble entering a command, check the prompt and enter the question mark (**?**) to see a list of available commands. You might be in the incorrect command mode or be using an incorrect syntax.

In the following example, notice how the prompt changes after each command to indicate a new command mode:

```
Router> enable
Password: <enable password>
Router# configure terminal
Router (config)# interface serial 0/0
Router (config-if)# line 0
Router (config-line)# controller t1 0
Router (config-controller)# exit
Router (config)# exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

The last message is normal and does not indicate an error. Press **Return** to get the `Router#` prompt.

Note You can press **Ctrl-Z** in any mode to immediately return to enable mode (`Router#`), instead of entering **exit**, which returns you to the previous mode.

Undoing a Command or Feature

If you want to undo a command that you entered or if you want to disable a feature, enter the keyword **no** before most commands; for example, **no ip routing**.

Saving Configuration Changes

To save your configuration changes to NVRAM, so that the changes are not lost during a system reload or power outage, enter the **copy running-config startup-config** command. For example:

```
Router# copy running-config startup-config
Building configuration...
```

It might take a few minutes to save the configuration to NVRAM. After the configuration has been saved, the following message appears:

```
[OK]
Router#
```

For additional information about using the Cisco IOS software, see the Configuration Fundamentals Configuration Guide, Cisco IOS Release 15.0S.

CHAPTER 3

First-Time Configuration

This chapter describes the actions to take before turning on your router for the first time. This chapter includes the following sections:

Understanding the Cisco MWR 2941 Router Interface Numbering

Each network interface on a Cisco MWR 2941 router is identified by a slot number and a port number.

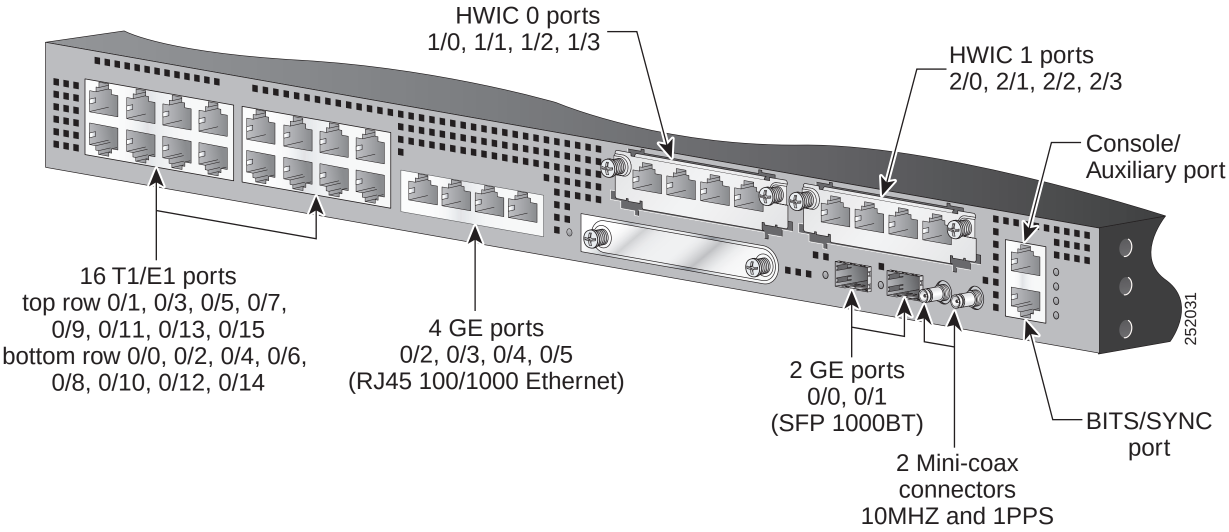
Figure 3-1 on page 3-2 shows an example of interface numbering on a Cisco MWR 2941 router:

- Two HWIC ports (HWICs are ordered separately)
- Two built-in Gigabit Ethernet small form-factor pluggable (SFP) interfaces (labeled GE0 and GE1)
- Four built-in Gigabit Ethernet interfaces (labeled L2–L5)
- 16 E1/T1 ports (labeled C1AL–C15AL)

Note The two HWIC cards shown in Figure 3-1 are not included with the Cisco MWR 2941 router; you must order them separately.

Note The mini-coax timing connectors shown in Figure 3-1 only apply to the Cisco MWR 2941-DC-A router; the Cisco MWR 2941-DC does not have these ports.

Figure 3-1 Cisco MWR 2941 Router Port Numbers



Slot and Port Numbering

The Cisco MWR 2941 router chassis contains the following interface types:

- 16 T1/E1 ports, labeled “T1/E1”
- 4 RJ-45 jacks for copper Ethernet ports, labeled “100/1000” Ethernet
- 2 HWIC slots, labeled “HWIC0” and “HWIC1”
- 1 compact FLASH Type-II connector, labeled “Compact Flash”
- 2 SFP connectors for optical GE ports, labeled “GE0” and “GE1”
- 2 miniature coaxial connectors for 10MHZ and 1PPS timing

Note Miniature coaxial timing connectors are not included on all versions of the Cisco MWR 2941. You can verify your hardware version with the VID label on the back of the router; routers labeled with a VID of V01 or V02 do not include the timing connectors, while routers with VID V03 and higher include the connectors.

- 1 RJ-45 connector for Console/Auxiliary, labeled “CON/AUX”
- 1 RJ-45 jack for BITS interface, labeled “BITS”

The logical slot numbers are 0 for all built-in interfaces.

The numbering format is:

```
Interface type Slot number/Interface number
```

Interface (port) numbers begin at logical 0 for each interface type.

Following is an explanation of the slot/port numbering:

- Logical interface numbering for the built-in T1/E1 ports runs from 0/0 through 0/15. Interfaces are hardwired; therefore, port 0 is always logical interface 0/0, port 1 is always logical interface 0/1, and so on. Built-in T1/E1 ports are numbered bottom to top, left to right (bottom row numbered 0-2-4-6-8-10-12-14, top row numbered 1-3-5-7-9-11-13-15).
- When the 2 HWIC slots are used to expand the T1/E1 port density to 20 or 24 ports, logical interface numbering continues from 1/0 through 1/3 and 2/0 through 2/3. Logical interfaces for HWIC0 are always 1/0 through 1/3 and logical interfaces for HWIC1 are always 2/0 through 2/3. Because the interfaces are hardwired, HWIC0 port 0 is always logical interface 1/0, HWIC0 port 1 is always logical interface 1/1, HWIC1 port 0 is always logical interface 2/0, HWIC1 port 1 is always logical interface 2/1, and so on. Ports are numbered left to right for each HWIC.
- Logical interface numbering for the built-in Ethernet ports runs from 0/0 through 0/5. Because the interfaces are hard-wired, ports correspond to logical interface numbers. For example, port 0 is always logical interface 0/0, and port 1 is always logical interface 0/1. SFP ports are numbered left to right, 0 and 1; 100/1000 Ethernet ports are numbered left to right, 2 through 5.
- Cisco IOS Setup Mode

Setup Mode

The **setup** mode guides you through creating a basic router configuration. If you prefer to configure the router manually or to configure a module or interface that is not included in **setup** mode, go to "Chapter 2, "Cisco IOS Software Basics" to familiarize yourself with the command-line interface (CLI).

Note Cisco Networking Services (CNS) is a collection of services that can provide remote configuration of Cisco IOS networking devices and remote execution of some command-line interface (CLI) commands. CNS allows a Cisco MWR 2941 deployed and powered on in the field to automatically download its configuration. For more information about CNS, see Cisco Networking Services (CNS), page 27-2.

Before Starting Your Router

Before you power on your router and begin using the **setup** mode, follow these steps:

Step 1 Set up the hardware and connect the console and network cables as described in the "Connecting Cables" section of the *Cisco MWR 2941-DC Router Hardware Installation Guide*.

Step 2 Configure your PC terminal emulation program for 9600 baud, 8 data bits, no parity, and 1 stop bit.

Using Setup Mode

The **setup** command facility appears in your PC terminal emulation program window. To create a basic configuration for your router, do the following:

- Complete the steps in the "Configuring Global Parameters" section on page 3-4
- Complete the steps in the "Completing the Configuration" section on page 3-6

Note If you make a mistake while using the setup command facility, you can exit the facility and run it again. Press **Ctrl-C**, and type **setup** at the enable mode prompt (`1900#`).

Configuring Global Parameters

Use the following procedure to configure global parameters.

Step 1 Power on the router. Messages appear in the terminal emulation program window.

Caution *Do not press any keys on the keyboard until the messages stop*. Any keys that you press during this time are interpreted as the first command entered after the messages stop, which might cause the router to power off and start over. Wait a few minutes. The messages stop automatically.

The messages look similar to the following:

Note The messages vary, depending on the Cisco IOS software image and interface modules in your router. This section is for reference only, and output might not match the messages on your console.

```
rommon 1 >boot
program load complete, entry point:0x80008000, size:0xc200

Initializing ATA monitor library.......
program load complete, entry point:0x80008000, size:0xc200

Initializing ATA monitor library.......
program load complete, entry point:0x80008000, size:0xc35eec
Self decompressing the image:
############################################################################
############################################################################
############################################################################
############################################################################
############################################################################
############################################################################
############################################################################
####################### [OK]

Smart Init is enabled
smart init is sizing iomem
    ID  MEMORY_REQTYPE
0035C   0X005F3C00 MWR2941 Mainboard
        0X000F3BB0 public buffer pools
        0X00843000 public particle pools
TOTAL:  0X06894CB0

If any of the above Memory requirements are "UNKNOWN", you may be using an
unsupported configuration or there is a software problem and system operation
may be compromised.
Rounded IOMEM up to: 104Mb.
Using 20 percent iomem. [104Mb/512Mb]

              Restricted Rights Legend

Use, duplication, or disclosure by the Government is
```

```
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.

           cisco Systems, Inc.
           170 West Tasman Drive
           San Jose, California 95134-1706

Cisco IOS Software, 2900 Software (MWR2900-IPRAN-M),
Experimental Version 12.4(20050412:070057),
Copyright (c) 1986-2009 by Cisco Systems, Inc.
Compiled Sat 10-Jan-09 03:19 by cbrezove
Image text-base:0x60008F60, data-base:0x6106A000

Cisco Systems, Inc. MWR-2941-DC (MPC8347E) processor (revision 0x400) with 41719
6K/107092K bytes of memory.
Processor board ID
MPC8347E CPU Rev: Part Number 0x8032, Revision ID 0x300
1 RTM Module: ASM-M2900-TOP daughter card
6 Gigabit Ethernet interfaces
1 terminal line
128K bytes of non-volatile configuration memory.
125440K bytes of ATA CompactFlash (Read/Write)

--- System Configuration Dialog ---
Would you like to enter the initial configuration dialog? [yes/no]: yes

At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.
```

Step 2 To begin the initial configuration dialog, enter **yes** when the following message appears:

```
Basic management setup configures only enough connectivity
for management of the system, extended setup will ask you
to configure each interface on the system

Would you like to enter basic management setup? [yes/no]:yes
Configuring global parameters:
```

Step 3 Enter a hostname for the router (this example uses 2941-1).

```
Configuring global parameters:

 Enter host name [Router]: 2941-1
```

Step 4 Enter an enable secret password. This password is encrypted (more secure) and cannot be seen when viewing the configuration.

```
The enable secret is a password used to protect access to
privileged EXEC and configuration modes. This password, after
entered, becomes encrypted in the configuration.
Enter enable secret: ciscoenable
```

Note When you enter the enable secret password, the password is visible while you type the it. After you enter the password, it becomes encrypted in the configuration.

Step 5 Enter an enable password that is different from the enable secret password. This password is *not* encrypted (less secure) and can be seen when viewing the configuration.

```
The enable password is used when you do not specify an
enable secret password, with some older software versions, and
some boot images.
Enter enable password: ciscoenable
```

Step 6 To prevent unauthenticated access to the router through ports other than the console port, enter the virtual terminal password.

```
The virtual terminal password is used to protect
access to the router over a network interface.
Enter virtual terminal password: ciscoterminal
```

Step 7 Respond to the following prompts as appropriate for your network:

```
Configure SNMP Network Management? [yes]:
    Community string [public]: public
```

Step 8 The summary of interfaces appears. This list varies, depending on the network modules installed in your router.

```
Current interface summary

Any interface listed with OK? value “NO” does not have a valid configuration

Interface               IP-Address      OK? Method Status        Protocol
GigabitEthernet0/0 unassigned      NO unset up up
GigabitEthernet0/1 unassigned      NO unset up up
```

Step 9 Specify the interface to be used to connect to the network management system.

```
Enter interface name used to connect to the
management network from the above interface summary: GigabitEthernet0/0
```

Step 10 Configure the specified interface as prompted.

```
Configuring interface GigabitEthernet0/0:
    Configure IP on this interface? [no]:
```

Completing the Configuration

When you have provided all of the information prompted for by the setup command facility, the configuration appears. Messages similar to the following appear:

```
The following configuration command script was created:

!
hostname 2941-1
enable secret 5 $1$5fH0$Z6Pr5EgtR5iNJ2nBg3i6y1 enable password ciscoenable line vty 0 4
password ciscoenablesnmp-server community public !
no ip routing

!
interface GigabitEthernet0/1
shutdown
!
end
```

To complete your router configuration, do the following:

Step 1 The setup command facility displays the following prompt.

```
[0] Go to the IOS command prompt without saving this config.
[1] Return back to the setup without saving this config.
[2] Save this configuration to nvram and exit.

Enter your selection [2]: 2
Building configuration...
[OK]

Use the enabled mode 'configure' command to modify this configuration.


Press RETURN to get started!
```

If you answer:

- **no—**The configuration information that you entered is *not* saved, and you return to the router enable prompt. To return to the system configuration dialog, enter **setup**.
- **yes—**The configuration is saved, and you return to the EXEC prompt.

Step 2 When the messages stop displaying in your window, press **Return** to view the command line prompt.

The `2941-1>` prompt indicates that you are now at the CLI and you have just completed a basic router configuration.

Note The basic configuration is *not* a complete configuration.

Verifying the Cisco IOS Software Version

To verify the version of Cisco IOS software, use the **show version** command. The **show version** command displays the configuration of the system hardware, the software version, the names and sources of the configuration files, and the boot images.

Configuring the Hostname and Password

First configure the hostname and set an encrypted password. Configuring a hostname allows you to distinguish multiple Cisco routers from each other. Setting an encrypted password allows you to prevent unauthorized configuration changes.

Note In the following procedure, press the **Return** key after each step unless otherwise noted. At any time, you can exit the privileged level and return to the user level by entering **disable** at the `Router#` prompt.

To configure a hostname and to set an encrypted password, follow these steps:

Step 1 Enter enable mode.

```
Router> enable
```

The Password prompt appears. Enter your password.

```
Password: password
```

When the prompt changes to `Router`, you have entered enable mode.

Step 2 Enter global configuration mode.

```
Router# configure terminal

Enter configuration commands, one per line. End with CNTL/Z.
```

When the prompt changes to `Router(config)`, you have entered global configuration mode.

```
Router(config)#
```

Step 3 Change the name of the router to a meaningful name. Substitute your hostname for `Router`.

```
Router(config)# hostname Router

Router(config)#
```

Step 4 Enter an enable secret password. This password provides access to privileged EXEC mode. When you type **enable** at the EXEC prompt (`Router>`), you must enter the enable secret password to access configuration mode. Enter your secret password.

```
Router(config)# enable secret secret password
```

Step 5 Exit back to global configuration mode.

```
Router(config)# exit
```

Verifying the Hostname and Password

To verify that you have correctly configured the hostname and password, follow these steps.

Step 1 Enter the **show config** command:

```
Router# show config
Using 1888 out of 126968 bytes
!
version XX.X
.
.
.
!
hostname Router
!
enable secret 5 $1$60L4$X2JYOwoDc0.kqa1loO/w8/
.
.
.
```

Step 2 Check the hostname and encrypted password, which appear near the top of the command output.

Step 3 Exit global configuration mode and attempt to re-enter it using the new enable password:

```
Router# exit
.
.Router con0 is now available
Press RETURN to get started.
Router> enable
Password: password
Router#
```

CHAPTER 4

Configuring Gigabit Ethernet Interfaces

To configure the Gigabit Ethernet (GE) interface on the Cisco MWR 2941, complete the following tasks:

- Configuring the Interface Properties, page 4-1
- Setting the Speed and Duplex Mode, page 4-2
- Enabling the Interface, page 4-3
- Creating Backup Switch Interfaces, page 4-4

Configuring the Interface Properties

Perform a basic Gigabit Ethernet IP Address configuration by specifying the port adapter and aligning an IP address and subnet mask of the interface as follows.

Note In the following procedure, press the **Return** key after each step unless otherwise noted. At any time, you can exit the privileged level and return to the user level by entering **disable** at the `Router#` prompt.

Note The spanning tree-related commands described in this section are optional.

To configure the GE interface, follow these steps while in global configuration mode:

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
	`Router#`
		Enter enable mode.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
	`Router(config)#`
		Enter configuration mode.

	Command	Purpose
Step 3	**interface gigabitethernet** *slot/port* **Example:** `Router(config)#` **`interface gigabitethernet 0/1`**	Specify the port adapter type and the location of the interface to be configured. The *slot* is always 0 and the *port* is the number of the port.

Step 4	**switchport mode {access | trunk}**
	Example:
	`Router(config-if)#`
	`switchport mode trunk`
		Specify the interface mode.

Step 5	**spanning-tree port-priority** *port_priority* **Example:** `Router(config-if)#` **`spanning-tree port-priority`** ***`port_priority`***	Specify an interface priority. You can use this value to prioritize an interface when two bridges compete for position as the root bridge.

Step 6	**spanning-tree cost** *port_cost*
	Example:
	`Router(config-if)#` **`spanning-tree`**
	`cost 10000000`
		To calculate the path cost of STP on an interface, use the **spanning-tree cost** command.

Step 7	**spanning-tree portfast**
	Example:
	`Router(config-if)#` **`spanning-tree`**
	`portfast`
		For interfaces that connect to end stations, you can use the **spanning-tree portfast** command to set the interface to move directly to the spanning-tree forwarding state when linkup occurs.

Step 8	**cdp enable** **Example:** `Router(config-if)#` **`cdp enable`**	To enable Cisco Discovery Protocol on the router, use the **cdp enable** command.

Step 9	**end**
	Example:
	`Router(config-if)#` **`end`**
	`Router#`
		Exit configuration mode.

Setting the Speed and Duplex Mode

The Gigabit Ethernet ports of the Cisco MWR 2941 router can run in full or half- duplex mode—100 Mbps or 1000 Mbps (1 Gbps). The Cisco MWR 2941 router has an autonegotiation feature that allows the router to negotiate the speed and duplex mode with the corresponding interface at the other end of the connection.

Autonegotiation is the default setting for the speed and transmission mode.

When you configure an interface speed and duplex mode, follow these guidelines:

- If both ends of the line support autonegotiation, use the default autonegotiation settings.
- When autonegotiation is turned on for either speed or duplex mode, it autonegotiates both speed and the duplex mode.
- If one interface supports autonegotiation, and the interface at the other end does not, configure the duplex mode and speed on both interfaces. If you use the autonegotiation setting on the supported side, the duplex mode setting is set at half-duplex.

Note In the following procedure, press the **Return** key after each step unless otherwise noted. At any time, you can exit the privileged level and return to the user level by entering **disable** at the `Router#` prompt.

To configure speed and duplex operation, follow these steps while in interface configuration mode:

	Command	Purpose
Step 1	**duplex** [**auto** | **half** | **full**] **Example:** `Router(config-if)#` **duplex auto**	Specify the duplex operation.
Step 2	**speed** [**auto** | **1000** | **100**] **Example:** `Router(config-if)#` **speed auto**	Specify the speed.

Enabling the Interface

Note In the following procedure, press the **Return** key after each step unless otherwise noted. At any time, you can exit the privileged level and return to the user level by entering **disable** at the `Router#` prompt.

	Command	Purpose
Step 1	**interface gigabitethernet** *slot/port* **Example:** `Router(config)#` **interface gigabitethernet 0/1**	Specify the port adapter type and the location of the interface to be configured. The *slot* is always 0 and the *port* is the number of the port.
Step 2	**no shutdown**	Enable the gigabit Ethernet interface using the **no shutdown** command.

Creating Backup Switch Interfaces

You can use the following command to create a backup switch interface:

	Command	Purpose
Step 1	**switchport backup interface** *interface_name* **preemption [forced | bandwidth | off] delay [time]**	Create a backup switch interface.

For more information about this command, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.

For instructions on how to create VLANs on GE interfaces, see Chapter 7, "Configuring VLANs."

CHAPTER 5

Configuring Layer 2 Interfaces

The Cisco MWR 2941 has an onboard layer 2 Gigabit Ethernet switch and supports HWICs with layer 2 interfaces.To configure the layer 2 interfaces on the Cisco MWR 2941, complete the following tasks:

- Configuring a Range of Interfaces, page 5-1
- Defining a Range Macro, page 5-2
- Configuring Layer 2 Optional Interface Features, page 5-2

Configuring a Range of Interfaces

The **interface range** command allows you to configure multiple interfaces at once. Follow these steps to configure an interface range.

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router>` **`enable`**
	`Router#`
		Enter enable mode.

Step 2	**`configure terminal`**
	Example:
	`Router#` **`configure terminal`**
	`Router(config)#`
		Enter configuration mode.

Step 3	**`interface range`** *`interface slot/port - port`*
	Example:
	`Router(config)#` **`interface range GigabitEthernet 0/1 - 3`**
		Use the **interface-range** command to select a range of interfaces to configure. You can specify a range that includes both VLANs and physical interfaces.

Defining a Range Macro

A range macro allows you to create a name that defines a range of interfaces on the Cisco MWR 2941. Follow these steps to configure an interface range macro.

	Command	Purpose

Step 1	**enable**
	
	Example:
	`Router> enable`
	`Router#`
		Enter enable mode.

Step 2	**configure terminal**
	
	Example:
	`Router#` **`configure terminal`**
	`Router(config)#`
		Enter configuration mode.

Step 3	**define interface-range macro** *interface slot/port - port* **Example:** `Router(config)#` **`define interface-range first_three GigabitEthernet0/1 - 2`**	Use the **define interface range** command to create the macro.

Configuring Layer 2 Optional Interface Features

- Interface Speed and Duplex Configuration Guidelines, page 5-2
- Configuring the Interface Speed, page 5-3
- Configuring the Interface Duplex Mode, page 5-3
- Configuring a Description for an Interface, page 5-4
- Configuring a Layer 2 Interface as a Layer 2 Trunk, page 5-5
- Configuring a Layer 2 Interface as Layer 2 Access, page 5-6

Interface Speed and Duplex Configuration Guidelines

Use the following guidelines when you configure an interface speed and duplex mode:

- Speed and duplex commands apply only to FastEthernet interfaces. They do not apply to the onboard Gigabit Ethernet ports.
- If both ends of the line support autonegotiation, use the default autonegotiation settings.
- If one interface supports auto negotiation and the other end does not, configure duplex and speed on both interfaces; do not use the auto setting on the supported side.
- Both ends of the line need to be configured to the same setting; for example, both hard-set or both auto-negotiate. Mismatched settings are not supported.

Caution Changing the interface speed and duplex mode configuration might shut down and re-enable the interface during the reconfiguration.

Configuring the Interface Speed

Follow these steps to configure the speed of a layer 2 interface.

	Command	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
	`Router#`
		Enter enable mode.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
	`Router(config)#`
		Enter configuration mode.

Step 3	**interface** *interface slot/port* **Example:** `Router(config)# interface fastethernet 1/0`	Enter configuration mode for the interface that you want to modify.

Step 4	`speed [10 | 100 | auto ]`
	Example:
	`Router(config-if)# speed auto`
		Specify the interface speed. You can set an interface to 10 Mbps, 100 Mbps, or autonegotiate.

Configuring the Interface Duplex Mode

Follow these steps below to set the duplex mode of a layer 2 interface.

	Command	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
	`Router#`
		Enter enable mode.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
	`Router(config)#`
		Enter configuration mode.

	Command	Purpose

Step 3	`interface` *`interface slot`*`/`*`port`*
	Example:
	`Router(config)# interface fastethernet 1/0`
		Enter configuration for the interface that you want to modify.

Step 4	`duplex [auto	full	half]` **Example:** `Router(config-if)# duplex auto`	Use the **duplex** command to set the interface to send traffic at full duplex, half duplex, or to autonegotiate its duplex setting.

Note If you set the port speed to auto on a 10/100-Mbps Ethernet interface, the interface auto-negotiates the speed and duplex settings. You cannot change the duplex mode of interfaces set to auto-negotiation.

Configuring a Description for an Interface

You can add a description of an interface to help you remember its function. The description appears in the output of the following commands: **show configuration**, **show running-config**, and **show interfaces**.

	Command	Purpose

Step 1	**enable**
	Example:
	`Router> enable`
	`Router#`
		Enter enable mode.

Step 2	**configure terminal**
	Example:
	`Router# configure terminal`
	`Router(config)#`
		Enter configuration mode.

Step 3	**interface** *interface slot/port* **Example:** `Router(config)# interface fastethernet 1/0`	Enter configuration for the interface that you want to modify.
Step 4	**description** *description* **Example:** `Router(config-if)# description newinterface`	Use the **description** command to assign a description to the interface.

Configuring a Layer 2 Interface as a Layer 2 Trunk

Follow these steps to configure an interface as a Layer 2 trunk.

	Command	Purpose

Step 1	**enable**
	Example:
	`Router> enable`
	`Router#`
		Enter enable mode.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
	`Router(config)#`
		Enter configuration mode.

Step 3	**`interface`** *`interface slot/port`*
	Example:
	`Router(config)# interface fastethernet 1/0`
		Enter configuration for the interface that you want to modify.

Step 4	**shutdown** **Example:** `Router(config-if)# shutdown`	Shut down the interface.
Step 5	**switchport mode trunk** **Example:** `Router(config-if)# switchport mode trunk`	Use the **switchport mode trunk** command to configure the interface as a Layer 2 trunk. **Note** The encapsulation is always set to dot1q.
Step 6	**switchport trunk native vlan** *vlan* **Example:** `Router(config-if)# switchport trunk native vlan 1`	If you are configuring an 802.1Q trunk, specify the native VLAN. Otherwise, proceed to the Step 7.
Step 7	**switchport trunk allowed vlan add** *vlan* **Example:** `Router(config-if)# switchport trunk allowed vlan add vlan1, vlan2, vlan3`	Use the **switchport trunk allowed vlan** command to configure the list of VLANs allowed on the trunk. The **add**, **except**, **none**, or **remove** keywords specify the action to take for the specified VLANs. **Note** All VLANs are allowed by default. You cannot remove any of the default VLANs from a trunk.

	Command	Purpose
Step 8	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Activate the interface.

Step 9	**end**
	Example:
	`Router(config-if)# end`
	`Router#`
		Exit configuration mode.
		You can use the **show running-configuration** command to verify the layer 2 trunk configuration.

Configuring a Layer 2 Interface as Layer 2 Access

Follow these steps to configure a Fast Ethernet interface as Layer 2 access.

	Command	Purpose

Step 1	**enable**
	Example:
	`Router> enable`
	`Router#`
		Enter enable mode.

Step 2	**configure terminal**
	Example:
	`Router# configure terminal`
	`Router(config)#`
		Enter configuration mode.

Step 3	**interface** *interface slot/port* **Example:** `Router(config)# interface fastethernet 1/0`	Enter configuration for the interface that you want to modify.
Step 4	**shutdown** **Example:** `Router(config-if)# shutdown`	Shut down the interface.
Step 5	**switchport mode access** **Example:** `Router(config-if)# switchport mode access`	Use the **switchport mode access** command to configure the interface as a layer 2 access.
Step 6	**switchport access vlan** *vlan* **Example:** `Router(config-if)# switchport access vlan 1`	Use the **switchport access vlan** command to specify an access VLAN for access ports.

	Command	Purpose
Step 7	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Activate the interface.

Step 8	**end**
	Example:
	`Router(config-if)# end`
	`Router#`
		Exit configuration mode.

Note You can use the **show running-config interface** command and the **show interfaces** command to verify layer 2 access configuration.

CHAPTER 6

Configuring HWIC-D-9ESW Interfaces

This chapter provides instructions for configuring interfaces on the HWIC-D-9ESW card. Follow these steps to configure a pair of ports on two different switch modules as stacking partners.

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface fastethernet** *interface-id* **Example:** `Router(config)# interface fastethernet 0/3/1`	Specifies the port to configure and enters interface configuration mode. • Enter the interface number.
Step 4	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Activates the interface. • This step is required only if you shut down the interface.
Step 5	**switchport stacking-partner interface fastethernet** *partner-interface-id* **Example:** `Router(config-if)# switchport stacking-partner interface FastEthernet partner-interface-id`	Selects and configures the stacking partner port. • Enter the partner interface number. • To restore the defaults, use the **no** form of this command.
Step 6	**exit** **Example:** `Router(config-if)# exit`	Returns to privileged configuration mode.

	Command or Action	Purpose
Step 7	**interface gigabitethernet** *partner-interface-id* **Example:** `Router# interface gigabitethernet 0/3/1`	Specifies the partner-interface, and enters interface configuration mode. • Enter the partner interface number. **Note** When you configure the FastEthernet port as a stacking partner, the corresponding GigabitEthernet interface is automatically configured as a stacking partner.
Step 8	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Activates the stacking partner interface.

Step 9	**end**
	Example:
	`Router(config-if)# end`
		Exits configuration mode.

When you have completed the configuration, connect a crossover Ethernet cable from FastEthernet port 8 of the HWIC-D-9ESW card to the GigabitEthernet port that you want to use as a stacking partner.

For more information about how to configure other features on the HWIC-D-9ESW card, see Chapter 5, "Configuring Layer 2 Interfaces."

Note Both stacking partner ports must have their **speed** and **duplex** parameters set to **auto**.

Caution If stacking is removed, stacked interfaces will go to **shutdown** state. Other nonstacked ports will be left unchanged.

Note For more detailed instructions on how to connect cables, see the *Cisco MWR 2941-DC Mobile Wireless Edge Router Hardware Installation Guide.*

CHAPTER 7

Configuring VLANs

This chapter describes how to configure normal-range VLANs (VLAN IDs 1 to 1005) and extended-range VLANs (VLAN IDs 1006 to 4094) on the Cisco MWR 2941 router. It includes information about VLAN membership modes, VLAN configuration modes, and VLAN trunks.

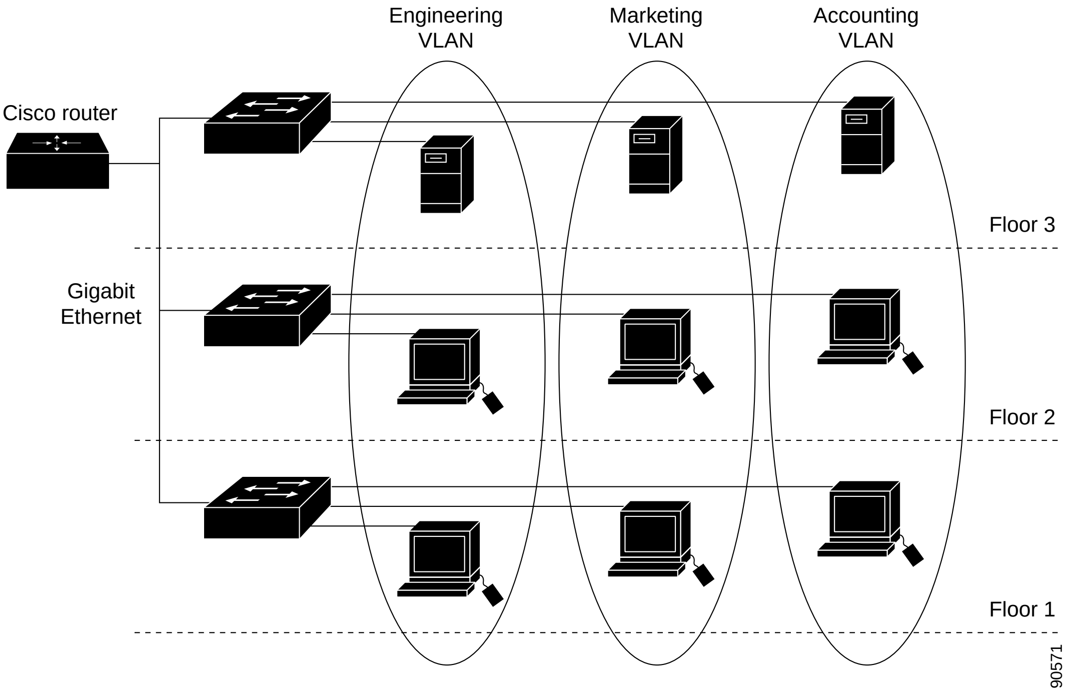
Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

Understanding VLANs

A VLAN is a switched network that is logically segmented by function, project team, or application, without regard to the physical locations of the users. VLANs have the same attributes as physical LANs, but you can group end stations even if they are not physically located on the same LAN segment. Any switch port can belong to a VLAN, and unicast, broadcast, and multicast packets are forwarded and flooded only to end stations in the VLAN. Each VLAN is considered a logical network, and packets destined for stations that do not belong to the VLAN must be forwarded through a router, as shown in Figure 7-1. Because a VLAN is considered a separate logical network, it contains its own bridge MIB information and can support its own implementation of spanning tree. See Chapter 9, "Configuring STP."

Figure 7-1 shows an example of VLANs segmented into logically defined networks.

Figure 7-1 VLANs as Logically Defined Networks



VLANs are often associated with IP subnetworks. For example, all the end stations in a particular IP subnet belong to the same VLAN. Interface VLAN membership on the switch is assigned manually on an interface-by-interface basis. When you assign switch interfaces to VLANs by using this method, it is known as interface-based, or static, VLAN membership.

Traffic between VLANs must be routed. Switches that are running the metro IP access image can route traffic between VLANs by using switch virtual interfaces (SVIs). To route traffic between VLANs, an SVI must be explicitly configured and assigned an IP address. For more information, see the *Interface and Hardware Component Configuration Guide, Cisco IOS Release 15.0S.*.

This section includes these topics:

- Supported VLANs, page 7-2
- Normal-Range VLANs, page 7-3
- Extended-Range VLANs, page 7-4
- VLAN Port Membership Modes, page 7-4

Supported VLANs

VLANs are identified with a number from 1 to 4094. VLAN IDs 1002 through 1005 are reserved for Token Ring and FDDI VLANs. VLAN IDs greater than 1005 are extended-range VLANs and are not stored in the VLAN database.

Although the router supports a total of 1005 (normal-range and extended-range) VLANs, the number of routed ports, SVIs, and other configured features affects the use of the switch hardware.

The router supports Per VLAN Spanning Tree Plus (PVST+) with a maximum of 128 spanning-tree instances. One spanning-tree instance is allowed per VLAN.

Note The router does not support Rapid PVST+.

Note Network node interfaces (NNIs) support STP by default. Enhanced network interfaces (ENIs) can be configured to support STP. User-network interfaces (UNIs) do not support STP and by default are always in a forwarding state.

See the "VLAN Configuration Guidelines" section on page 7-6 for more information about the number of spanning-tree instances and the number of VLANs. The switch supports IEEE 802.1Q trunking for sending VLAN traffic over Ethernet ports.

Normal-Range VLANs

Normal-range VLANs are VLANs with VLAN IDs 1 to 1005. You can add, modify or remove configurations for VLANs 2 to 1001 in the VLAN database. (VLAN IDs 1 and 1002 to 1005 are automatically created and cannot be removed.)

Configurations for VLAN IDs 1 to 1005 are written to the file vlan.dat (VLAN database), and you can display them by entering the **show vlan** privileged EXEC command. The *vlan.dat* file is stored in flash memory.

Caution You can cause inconsistency in the VLAN database if you try to manually delete the vlan.dat file. If you want to modify the VLAN configuration, use the commands described in this guide and in the command reference for this release.

You can set these parameters when you create a new normal-range VLAN or modify an existing VLAN in the VLAN database:

- VLAN ID
- VLAN name
- VLAN type (Ethernet, FDDI, FDDI network entity title [NET], TrBRF, or TrCRF, Token Ring, Token Ring-Net)

Note The switch supports only Ethernet VLANs. You can configure parameters for FDDI and Token Ring VLANs and view the results in the vlan.dat file, but these parameters are not used.

- VLAN state (active or suspended)
- Maximum transmission unit (MTU) for the VLAN
- Security Association ID (SAID)
- Bridge identification number for TrBRF VLANs
- Ring number for FDDI and TrCRF VLANs
- Parent VLAN number for TrCRF VLANs
- Spanning Tree Protocol (STP) type for TrCRF VLANs

- VLAN number to use when translating from one VLAN type to another

Note This chapter does not provide configuration details for most of these parameters. For complete information on the commands and parameters that control VLAN configuration, see the command reference for this release.

Extended-Range VLANs

You can create extended-range VLANs (in the range 1006 to 4094) to enable service providers to extend their infrastructure to a greater number of customers. The extended-range VLAN IDs are allowed for any **switchport** commands that allow VLAN IDs. Extended-range VLAN configurations are not stored in the VLAN database, but they are stored in the switch running configuration file, and you can save the configuration in the startup configuration file by using the **copy running-config startup-config** privileged EXEC command.

Note Although the switch supports 4094 VLAN IDs, the actual number of VLANs supported is 1005.

VLAN Port Membership Modes

You configure a port to belong to a VLAN by assigning a membership mode that specifies the kind of traffic that the port carries and the number of VLANs to which it can belong. Table 7-1 lists the membership modes and characteristics.

Table 7-1 Port Membership Modes

Membership Mode	VLAN Membership Characteristics
Static-access	A static-access port can belong to one VLAN and is manually assigned to that VLAN. For more information, see the “Assigning Static-Access Ports to a VLAN” section on page 7-8.
Trunk (IEEE 802.1Q)	A trunk port is a member of all VLANs by default, including extended-range VLANs, but membership can be limited by configuring the allowed-VLAN list. For information about configuring trunk ports, see the “Configuring an Ethernet Interface as a Trunk Port” section on page 7-13.
Tunnel (**dot1q-tunnel**)	Tunnel ports are used for IEEE 802.1Q tunneling to maintain customer VLAN integrity across a service-provider network. You configure a tunnel port on an edge switch in the service-provider network and connect it to an IEEE 802.1Q trunk port on a customer interface, creating an assymetric link. A tunnel port belongs to a single VLAN that is dedicated to tunneling. For more information about tunnel ports, see Chapter 8, “Configuring IEEE 802.1Q Tunneling and Layer 2 Protocol Tunneling.”

For more detailed definitions of access and trunk modes and their functions, see Table 7-4 on page 7-12.

When a port belongs to a VLAN, the switch learns and manages the addresses associated with the port on a per-VLAN basis. For more information, see the Chapter 12, “Managing the MAC Address Table.”.

Creating and Modifying VLANs

You use VLAN configuration mode, accessed by entering the **vlan** global configuration command to create VLANs and to modify some parameters. You use the interface configuration mode to define the port membership mode and to add and remove ports from VLANs. The results of these commands are written to the running-configuration file, and you can display the file by entering the **show running-config** privileged EXEC command.

These sections contain VLAN configuration information:

- Default Ethernet VLAN Configuration, page 7-5
- VLAN Configuration Guidelines, page 7-6
- Creating or Modifying an Ethernet VLAN, page 7-7
- Assigning Static-Access Ports to a VLAN, page 7-8
- Creating an Extended-Range VLAN with an Internal VLAN ID, page 7-9

For more efficient management of the MAC address table space available on the switch, you can control which VLANs learn MAC addresses by disabling MAC address learning on specific VLANs. See the "Managing the MAC Address Table" section on page 12-1 for more information.

Default Ethernet VLAN Configuration

The switch supports only Ethernet interfaces. Table 7-2 shows the default configuration for Ethernet VLANs.

Note On extended-range VLANs, you can change only the MTU size and the UNI-ENI VLAN configuration. All other characteristics must remain at the default conditions.

Table 7-2 Ethernet VLAN Defaults and Ranges

Parameter	Default	Range
VLAN ID	1	1–4094 **Note** Extended-range VLANs (VLAN IDs 1006 to 4094) are not saved in the VLAN database.
VLAN name	*VLANxxxx*, where *xxxx* represents four numeric digits (including leading zeros) equal to the VLAN ID number	No range
IEEE 802.10 SAID	100001 (100000 plus the VLAN ID)	1–4294967294
MTU size	1500	1500–9198
Translational bridge 1	0	0–1005
Translational bridge 2	0	0–1005

Table 7-2 Ethernet VLAN Defaults and Ranges (continued)

Parameter	Default	Range
VLAN state	active	active, suspend
UNI-ENI VLAN	UNI-ENI isolated VLAN	2–1001, 1006–4094 VLAN 1 is always a UNI-ENI isolated VLAN.

VLAN Configuration Guidelines

Follow these guidelines when creating and modifying VLANs in your network:

- The router supports 1005 VLANs.
- Normal-range Ethernet VLANs are identified with a number between 1 and 1001. VLAN numbers 1002 through 1005 are reserved for Token Ring and FDDI VLANs.
- The router does not support Token Ring or FDDI media. The switch does not forward FDDI, FDDI-Net, TrCRF, or TrBRF traffic.
- VLAN configurations for VLANs 1 to 1005 are always saved in the VLAN database and in the switch running configuration file.
- Configuration options for VLAN IDs 1006 through 4094 (extended-range VLANs) are limited to MTU. Extended-range VLANs are not saved in the VLAN database.
- STP is enabled by default only for NNIs on all VLANs. You can configure STP on ENIs. NNIs and ENIs in the same VLAN are in the same spanning-tree instance. The switch supports 128 spanning-tree instances. If a switch has more active VLANs than supported spanning-tree instances, spanning tree can be enabled on 128 VLANs and is disabled on the remaining VLANs. If you have already used all available spanning-tree instances on a switch, adding another VLAN creates a VLAN on that switch that is not running spanning tree. If you have the default allowed list on the trunk ports of that switch (which is to allow all VLANs), the new VLAN is carried on all trunk ports. Depending on the topology of the network, this could create a loop in the new VLAN that would not be broken, particularly if there are several adjacent switches that have run out of spanning-tree instances. You can prevent this by setting allowed lists on the trunk ports of switches that have used up their allocation of spanning-tree instances.

  If the number of VLANs on the switch exceeds the number of supported spanning-tree instances, configure the IEEE 802.1s Multiple STP (MSTP) on your switch to map multiple VLANs to a single spanning-tree instance. For more information about MSTP, see Chapter 10, "Configuring MSTP."

  Note MSTP is supported only on NNIs on ENIs on which STP has been enabled.

- Each routed port on the switch creates an internal VLAN for its use. These internal VLANs use extended-range VLAN numbers, and the internal VLAN ID cannot be used for an extended-range VLAN. If you try to create an extended-range VLAN with a VLAN ID that is already allocated as an internal VLAN, an error message is generated, and the command is rejected.
  - Because internal VLAN IDs are in the lower part of the extended range, we recommend that you create extended-range VLANs beginning from the highest number (4094) and moving to the lowest (1006) to reduce the possibility of using an internal VLAN ID.
  - Before configuring extended-range VLANs, enter the **show vlan internal usage** privileged EXEC command to see which VLANs have been allocated as internal VLANs.

  - If necessary, you can shut down the routed port assigned to the internal VLAN, which frees up the internal VLAN, and then create the extended-range VLAN and re-enable the port, which then uses another VLAN as its internal VLAN. See the "Creating an Extended-Range VLAN with an Internal VLAN ID" section on page 7-9.

- Although the switch supports a total of 1005 (normal-range and extended-range) VLANs, the number of routed ports, SVIs, and other configured features affects the use of the router hardware. If you try to create an extended-range VLAN and there are not enough hardware resources available, an error message is generated, and the extended-range VLAN is rejected.

Creating or Modifying an Ethernet VLAN

To access VLAN configuration mode, enter the **vlan** global configuration command with a VLAN ID. Enter a new VLAN ID to create a VLAN, or enter an existing VLAN ID to modify that VLAN. You can use the default VLAN configuration (Table 7-2) or enter commands to configure the VLAN.

Note Extended-range VLANs use the default Ethernet VLAN characteristics and the MTU and the UNI-ENI VLAN configurations are the only parameters that you can change.

For more information about commands available in VLAN configuration mode, see the **vlan** command description in the command reference for this release. When you have finished the configuration, you must exit VLAN configuration mode for the configuration to take effect. To display the VLAN configuration, enter the **show vlan** privileged EXEC command.

The configurations of VLAN IDs 1 to 1005 are always saved in the VLAN database (vlan.dat file) with a VLAN number and name and in the switch running configuration file. Extended-range VLANs are not saved in the VLAN database; they are saved in the switch running configuration file. You can save the VLAN configuration in the switch startup configuration file by using the **copy running-config startup-config** privileged EXEC command.

Note Before you create an extended-range VLAN, you can verify that the VLAN ID is not used internally by entering the **show vlan internal usage** privileged EXEC command. If the VLAN ID is used internally and you want to release it, go to the "Creating an Extended-Range VLAN with an Internal VLAN ID" section on page 7-9 before creating the extended-range VLAN.

Follow these steps to create or modify an Ethernet VLAN:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vlan** *vlan-id*	Enter a VLAN ID, and enter VLAN configuration mode. Enter a new VLAN ID to create a VLAN, or enter an existing VLAN ID to modify that VLAN. The available VLAN ID range for this command is 1–4094. **Note** When you create a new VLAN, by default the VLAN is a UNI-ENI isolated VLAN.
Step 3	**name** *vlan-name*	(Optional and supported only on normal-range VLANs) Enter a name for the VLAN. If no name is entered for the VLAN, the default in the VLAN database is to append the *vlan-id* with leading zeros to the word VLAN. For example, VLAN0004 is a default VLAN name for VLAN 4.

	Command	Purpose
Step 4	**mtu** *mtu-size*	(Optional) Change the MTU size.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show vlan** {**name** *vlan-name* | **id** *vlan-id*}	Verify your entries. The **name** option is valid only for VLAN IDs 1 to 1005.
Step 7	**copy running-config startup config**	(Optional) Save the configuration in the switch startup configuration file.

To delete a VLAN, use the **no vlan** *vlan-id* global configuration command. You cannot delete VLAN 1 or VLANs 1002 to 1005.

Caution When you delete a VLAN, any ports assigned to that VLAN become inactive. They remain associated with the VLAN (and thus inactive) until you assign them to a new VLAN.

To return the VLAN name to the default settings, use the **no name** or **no mtu** VLAN configuration command.

This example shows how to create Ethernet VLAN 20, name it *test20*, and add it to the VLAN database:

```
Router# configure terminal
Router(config)# vlan 20
Router(config-vlan)# name test20
Router(config-vlan)# end
```

This example shows how to create a new extended-range VLAN with all default characteristics, enter config-vlan mode, and save the new VLAN in the switch startup configuration file:

```
Router(config)# vlan 2000
Router(config-vlan)# end
Router# copy running-config startup config
```

Assigning Static-Access Ports to a VLAN

You can assign a static-access port to a VLAN.

Note If you assign an interface to a VLAN that does not exist, the new VLAN is created. (See the "Creating or Modifying an Ethernet VLAN" section on page 7-7.)

Beginning in privileged EXEC mode, follow these steps to assign a port to a VLAN in the VLAN database:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode
Step 2	**interface** *interface-id*	Enter the interface to be added to the VLAN.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
Step 4	**switchport mode access**	Define the VLAN membership mode for the port (Layer 2 access port).

	Command	Purpose
Step 5	**switchport access vlan** *vlan-id*	Assign the port to a VLAN. Valid VLAN IDs are 1 to 4094.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config interface** *interface-id*	Verify the VLAN membership mode of the interface.
Step 8	**show interfaces** *interface-id* **switchport**	Verify your entries in the Administrative Mode and the Access Mode VLAN fields of the display.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return an interface to its default configuration, use the **default interface** *interface-id* interface configuration command.

This example shows how to configure a port as an access port in VLAN 2:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# interface fastethernet0/1
Router(config-if)# switchport mode access
Router(config-if)# switchport access vlan 2
Router(config-if)# end
```

Creating an Extended-Range VLAN with an Internal VLAN ID

If you enter an extended-range VLAN ID that is already assigned to an internal VLAN, an error message appears, and the extended-range VLAN is rejected. To manually release an internal VLAN ID, you must temporarily shut down the routed port that is using the internal VLAN ID.

Beginning in privileged EXEC mode, follow these steps to release a VLAN ID that is assigned to an internal VLAN and to create an extended-range VLAN with that ID:

	Command	Purpose
Step 1	**show vlan internal usage**	Display the VLAN IDs being used internally by the switch. If the VLAN ID that you want to use is an internal VLAN, the display shows the routed port that is using the VLAN ID. Enter that port number in Step 3.
Step 2	**configure terminal**	Enter global configuration mode.
Step 3	**interface** *interface-id*	Specify the interface ID for the routed port that is using the VLAN ID, and enter interface configuration mode.
Step 4	**shutdown**	Shut down the port to release the internal VLAN ID.
Step 5	**exit**	Return to global configuration mode.
Step 6	**vlan** *vlan-id*	Enter the new extended-range VLAN ID, and enter config-vlan mode.
Step 7	**exit**	Exit from config-vlan mode, and return to global configuration mode.
Step 8	**interface** *interface-id*	Specify the interface ID for the routed port that you shut down in Step 4, and enter interface configuration mode.
Step 9	**no shutdown**	Re-enable the routed port. It will be assigned a new internal VLAN ID.
Step 10	**end**	Return to privileged EXEC mode.
Step 11	**copy running-config startup config**	(Optional) Save your entries in the switch startup configuration file.

Configuring VLAN Trunking Protocol

This section describes how to configure the VLAN Trunking Protocol (VTP) on an EtherSwitch HWIC, and contains the following tasks:

- Configuring a VTP Server
- Configuring a VTP Client
- Disabling VTP

Configuring a VTP Server

When a router is in VTP server mode, you can change the VLAN configuration and have it propagate throughout the network. Follow these steps to configure the router as a VTP server:

	Command	Purpose
Step 1	**enable**	Enter enable mode.
Step 2	**configure terminal**	Enter configuration mode.
Step 3	**vtp mode server**	Configure the router as a VTP server.
Step 4	**vtp domain**	Define the VTP domain name, which can be up to 32 characters long.
Step 5	**vtp password** *password*	(Optional) If you want to specify a password for the VTP domain, use **vtp password** command. The password can be from 8 to 64 characters long.
Step 6	**exit**	Exit configuration mode.

Configuring a VTP Client

When a router is in VTP client mode, you cannot change the VLAN configuration. A client switch receives VTP updates from a VTP server in the management domain and modifies its configuration accordingly. Follow these steps to configure the router as a VTP client.

	Command	Purpose
Step 1	**enable**	Enter enable mode.
Step 2	**configure terminal**	Enter configuration mode.
Step 3	**vtp mode client**	Configure the switch as a VTP client.
Step 4	**exit**	Exit configuration mode.

Disabling VTP

You can disable VTP on the router by configuring it to VTP transparent mode, meaning that the router does not send VTP updates or act on VTP updates received from other switches. Follow these steps to disable VTP on the router:

	Command	Purpose
Step 1	**enable**	Enter enable mode.
Step 2	**configure terminal**	Enter configuration mode.
Step 3	**vtp mode transparent**	Set the router in VTP transparent mode.
Step 4	**exit**	Exit configuration mode.

Note You can use the **show vtp status** command to verify the VTP status of the router.

Displaying VLANs

Use the **show vlan** privileged EXEC command to display a list of all VLANs on the router, including extended-range VLANs. The display includes VLAN status, ports, and configuration information. Table 7-3 lists other privileged EXEC commands for monitoring VLANs.

Table 7-3 VLAN Monitoring Commands

Command	Purpose
show interfaces [**vlan** *vlan-id*]	Display characteristics for all interfaces or for the specified VLAN configured on the router.
show vlan [**id** *vlan-id*]	Display parameters for all VLANs or the specified VLAN on the router.
show vlan [*vlan-name*] **uni-vlan type**	Display UNI-ENI isolated or UNI-ENI community VLANs by VLAN name.
show vlan uni-vlan	Display UNI-ENI community VLANs and associated ports on the router.
show vlan uni-vlan type	Display UNI-ENI isolated and UNI-ENI community VLANs on the router by VLAN ID.

For more details about the **show** command options and explanations of output fields, see the command reference for this release.

Configuring VLAN Trunks

Trunking Overview

A trunk is a point-to-point link between one or more Ethernet switch interfaces and another networking device such as a router or a switch. Ethernet trunks carry the traffic of multiple VLANs over a single link, and you can extend the VLANs across an entire network. The router supports the IEEE 802.1Q industry-standard trunking encapsulation.

Ethernet interfaces support different trunking modes (see Table 7-4). You can set an interface as trunking or nontrunking.

- If you do not intend to trunk across links, use the **switchport mode access** interface configuration command to disable trunking.
- To enable trunking, use the **switchport mode trunk** interface configuration command to change the interface to a trunk.

Table 7-4 Layer 2 Interface Modes

Mode	Function
switchport mode access	Puts the interface (access port) into permanent nontrunking mode and negotiates to convert the link into a nontrunk link. The interface becomes a nontrunk interface regardless of whether or not the neighboring interface is a trunk interface. This is the default mode.
switchport mode trunk	Puts the interface into permanent trunking mode and negotiates to convert the neighboring link into a trunk link. The interface becomes a trunk interface even if the neighboring interface is not a trunk interface.
switchport mode dot1q-tunnel	Configures the interface as a tunnel (nontrunking) port to be connected in an asymmetric link with an IEEE 802.1Q trunk port. The IEEE 802.1Q tunneling is used to maintain customer VLAN integrity across a service provider network. See Chapter 8, “Configuring IEEE 802.1Q Tunneling and Layer 2 Protocol Tunneling,” for more information on tunnel ports.

IEEE 802.1Q Configuration Considerations

The IEEE 802.1Q trunks impose these limitations on the trunking strategy for a network:

- In a network of Cisco switches connected through IEEE 802.1Q trunks, the switches maintain one spanning-tree instance for each VLAN allowed on the trunks. Non-Cisco devices might support one spanning-tree instance for all VLANs.

  When you connect a Cisco switch to a non-Cisco device through an IEEE 802.1Q trunk, the Cisco switch combines the spanning-tree instance of the VLAN of the trunk with the spanning-tree instance of the non-Cisco IEEE 802.1Q switch. However, spanning-tree information for each VLAN is maintained by Cisco switches separated by a cloud of non-Cisco IEEE 802.1Q switches. The non-Cisco IEEE 802.1Q cloud separating the Cisco switches is treated as a single trunk link between the switches.

- If the native VLAN on one end of the trunk is different from the native VLAN on the other end, spanning-tree loops might result. Make sure that the native VLAN for an IEEE 802.1Q trunk is the same on both ends of the trunk link.
- Disabling spanning tree on the native VLAN of an IEEE 802.1Q trunk without disabling spanning tree on every VLAN in the network can potentially cause spanning-tree loops. Leave spanning tree enabled on the native VLAN of an IEEE 802.1Q trunk or disable spanning tree on every VLAN in the network. Make sure that your network is loop-free before disabling spanning tree.

Default Layer 2 Ethernet Interface VLAN Configuration

Table 7-5 shows the default Layer 2 Ethernet interface VLAN configuration.

Table 7-5 Default Layer 2 Ethernet Interface VLAN Configuration

Feature	Default Setting
Interface mode	**switchport mode access**
Allowed VLAN range	VLANs 1–4094
Default VLAN (for access ports)	VLAN 1
Native VLAN (for IEEE 802.1Q trunks)	VLAN 1

Configuring an Ethernet Interface as a Trunk Port

- Interaction with Other Features, page 7-13
- Defining the Allowed VLANs on a Trunk, page 7-14
- Configuring the Native VLAN for Untagged Traffic, page 7-15

Interaction with Other Features

Trunking interacts with other features in these ways:

- A trunk port cannot be a secure port.
- A trunk port cannot be a tunnel port.
- If you try to enable IEEE 802.1x on a trunk port, an error message appears, and IEEE 802.1x is not enabled. If you try to change the mode of an IEEE 802.1x-enabled port to trunk, the port mode is not changed.

Configuring a Trunk Port

Beginning in privileged EXEC mode, follow these steps to configure a port as an IEEE 802.1Q trunk port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured for trunking, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
Step 4	**switchport mode trunk**	Configure the interface as a Layer 2 trunk.
Step 5	**switchport access vlan** *vlan-id*	(Optional) Specify the default VLAN, which is used if the interface stops trunking.
Step 6	**switchport trunk native vlan** *vlan-id*	Specify the native VLAN for IEEE 802.1Q trunks.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show interfaces** *interface-id* **switchport**	Display the switchport configuration of the interface in the Administrative Mode field of the display.
Step 9	**show interfaces** *interface-id* **trunk**	Display the trunk configuration of the interface.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return an interface to its default configuration, use the **default interface** *interface-id* interface configuration command. To reset all trunking characteristics of a trunking interface to the defaults, use the **no switchport trunk** interface configuration command. To disable trunking, use the **switchport mode access** interface configuration command to configure the port as a static-access port.

This example shows how to configure a port as an IEEE 802.1Q trunk with VLAN 33 as the native VLAN:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# interface fastethernet0/2
Router(config-if)# switchport mode trunk
Router(config-if)# switchport trunk native vlan 33
Router(config-if)# end
```

Defining the Allowed VLANs on a Trunk

By default, a trunk port sends traffic to and receives traffic from all VLANs. All VLAN IDs, 1 to 4094, are allowed on each trunk. However, you can remove VLANs from the allowed list, preventing traffic from those VLANs from passing over the trunk. To restrict the traffic a trunk carries, use the **switchport trunk allowed vlan remove** *vlan-list* interface configuration command to remove specific VLANs from the allowed list.

Note VLAN 1 is the default VLAN on all trunk ports in all Cisco switches, and it has previously been a requirement that VLAN 1 always be enabled on every trunk link. The VLAN 1 minimization feature allows you to disable VLAN 1 on any individual VLAN trunk link so that no user traffic (including spanning-tree advertisements) is sent or received on VLAN 1. You do this by removing VLAN 1 from the allowed VLAN list.

To reduce the risk of spanning-tree loops or storms, you can disable VLAN 1 on any individual VLAN trunk port by removing VLAN 1 from the allowed list. When you remove VLAN 1 from a trunk port, the interface continues to send and receive management traffic, for example, Cisco Discovery Protocol.

If a trunk port with VLAN 1 disabled is converted to a nontrunk port, it is added to the access VLAN. If the access VLAN is set to 1, the port is added to VLAN 1, regardless of the **switchport trunk allowed** setting. The same is true for any VLAN that has been disabled on the port.

A trunk port can become a member of a VLAN if the VLAN is enabled and if the VLAN is in the allowed list for the port.

Beginning in privileged EXEC mode, follow these steps to modify the allowed list of an IEEE 802.1Q trunk:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
Step 4	**switchport mode trunk**	Configure the interface as a VLAN trunk port.
Step 5	**switchport trunk allowed vlan** {**add** | **all** | **except** | **remove**} *vlan-list*	(Optional) Configure the list of VLANs allowed on the trunk. For explanations about using the **add**, **all**, **except**, and **remove** keywords, see the command reference for this release. The *vlan-list* parameter is either a single VLAN number from 1 to 4094 or a range of VLANs described by two VLAN numbers, the lower one first, separated by a hyphen. Do not enter any spaces between comma-separated VLAN parameters or in hyphen-specified ranges. All VLANs are allowed by default.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show interfaces** *interface-id* **switchport**	Verify your entries in the *Trunking VLANs Enabled* field of the display.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default allowed VLAN list of all VLANs, use the **no switchport trunk allowed vlan** interface configuration command.

This example shows how to remove VLAN 2 from the allowed VLAN list on a port:

```
Router(config)# interface fastethernet0/1
Router(config-if)# switchport trunk allowed vlan remove 2
Router(config-if)# end
```

Configuring the Native VLAN for Untagged Traffic

A trunk port configured with IEEE 802.1Q tagging can receive both tagged and untagged traffic. By default, the router forwards untagged traffic in the native VLAN configured for the port. The native VLAN is VLAN 1 by default.

Note The native VLAN can be assigned any VLAN ID.

For information about IEEE 802.1Q configuration issues, see the "IEEE 802.1Q Configuration Considerations" section on page 7-12.

Beginning in privileged EXEC mode, follow these steps to configure the native VLAN on an IEEE 802.1Q trunk:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Define the interface that is configured as the IEEE 802.1Q trunk, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
Step 4	**switchport trunk native vlan** *vlan-id*	Configure the VLAN that is sending and receiving untagged traffic on the trunk port. For *vlan-id*, the range is 1 to 4094.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **switchport**	Verify your entries in the Trunking Native Mode VLAN field.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default native VLAN, VLAN 1, use the **no switchport trunk native vlan** interface configuration command.

If a packet has a VLAN ID that is the same as the sending port native VLAN ID, the packet is sent untagged; otherwise, the router sends the packet with a tag.

Configuring Trunk Ports for Load Sharing

Load sharing divides the bandwidth supplied by parallel trunks that connect switches. To avoid loops, STP normally blocks all but one parallel link between switches. Using load sharing, you divide the traffic between the links according to the VLAN to which the traffic belongs.

You configure load sharing on trunk ports that have STP enabled by using STP port priorities or STP path costs. For load sharing using STP port priorities, both load-sharing links must be connected to the same switch. For load sharing using STP path costs, each load-sharing link can be connected to the same switch or to two different switches. For more information about STP, see Chapter 9, “Configuring STP.”

Load Sharing Using STP Port Priorities

When two ports on the same switch form a loop, the switch uses the STP port priority to decide which port is enabled and which port is in a blocking state. You can set the priorities on a parallel STP trunk port so that the port carries all the traffic for a given VLAN. The trunk port with the higher priority (lower values) for a VLAN is forwarding traffic for that VLAN. The trunk port with the lower priority (higher values) for the same VLAN remains in a blocking state for that VLAN. One trunk port sends or receives all traffic for the VLAN.

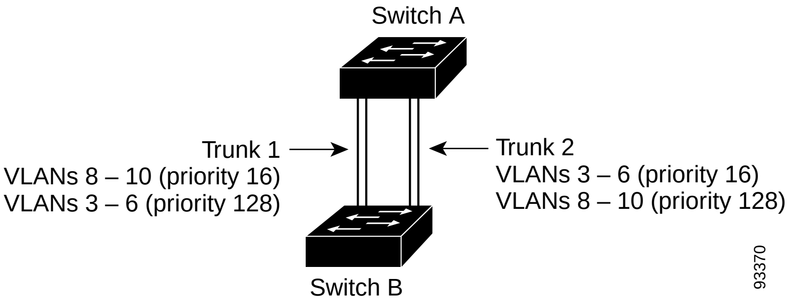
Figure 7-2 shows two trunks connecting supported switches. In this example, the switches are configured as follows:

- VLANs 8 through 10 are assigned a port priority of 16 on Trunk 1.
- VLANs 3 through 6 retain the default port priority of 128 on Trunk 1.
- VLANs 3 through 6 are assigned a port priority of 16 on Trunk 2.

- VLANs 8 through 10 retain the default port priority of 128 on Trunk 2.

In this way, Trunk 1 carries traffic for VLANs 8 through 10, and Trunk 2 carries traffic for VLANs 3 through 6. If the active trunk fails, the trunk with the lower priority takes over and carries the traffic for all of the VLANs. No duplication of traffic occurs over any trunk port.

Figure 7-2 Load Sharing by Using STP Port Priorities


Switch A
Trunk 1
VLANs 8 – 10 (priority 16)
VLANs 3 – 6 (priority 128)
Trunk 2
VLANs 3 – 6 (priority 16)
VLANs 8 – 10 (priority 128)
Switch B
93370

Beginning in privileged EXEC mode on Switch A, follow these steps to configure the network shown in Figure 7-2. Note that you can use any interface numbers; those shown are examples.

	Command	Purpose
Step 1	**show vlan**	Verify that the referenced VLANs exist on Switch A. If not, create the VLANs by entering the VLAN IDs.
Step 2	**configure terminal**	Enter global configuration mode.
Step 3	**interface gigabitethernet 0/1**	Define the interface to be configured as the Trunk 1 interface, and enter interface configuration mode.
Step 4	**switchport mode trunk**	Configure the port as a trunk port.
Step 5	**spanning-tree vlan 8-10 port-priority 16**	Assign the port priority of 16 for VLANs 8 through 10 on Trunk 1.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show interfaces gigabitethernet 0/1 switchport**	Verify the port configuration.
Step 8	**configure terminal**	Enter global configuration mode.
Step 9	**interface gigabitethernet 0/2**	Define the interface to be configured as the Trunk 2 interface, and enter interface configuration mode.
Step 10	**switchport mode trunk**	Configure the port as a trunk port.
Step 11	**spanning-tree vlan 3-6 port-priority 16**	Assign the port priority of 16 for VLANs 3 through 6 on Trunk 2.
Step 12	**end**	Return to privileged EXEC mode.
Step 13	**show interfaces gigabitethernet 0/2 switchport**	Verify the port configuration.
Step 14	**show running-config**	Verify your entries.
Step 15	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Follow the same steps on Switch B to configure the trunk port for Trunk 1 with a spanning-tree port priority of 16 for VLANs 8 through 10, and the configure trunk port for Trunk 2 with a spanning-tree port priority of 16 for VLANs 3 through 6.

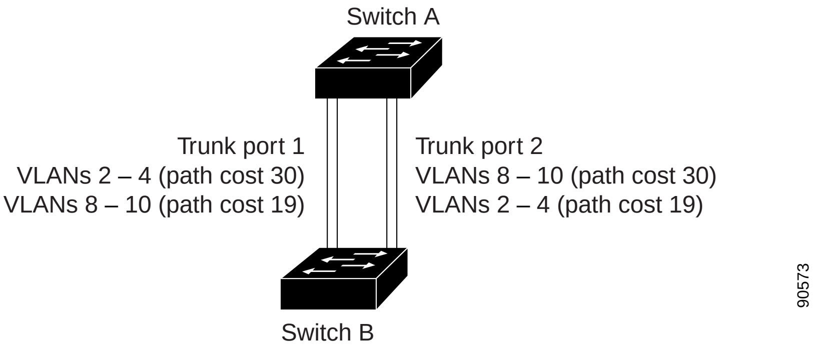
Load Sharing Using STP Path Cost

You can configure parallel trunks to share VLAN traffic by setting different path costs on a trunk and associating the path costs with different sets of VLANs, blocking different ports for different VLANs. The VLANs keep the traffic separate and maintain redundancy in the event of a lost link.

In Figure 7-3, Trunk ports 1 and 2 are configured as 100Base-T ports. These VLAN path costs are assigned:

- VLANs 2 through 4 are assigned a path cost of 30 on Trunk port 1.
- VLANs 8 through 10 retain the default 100Base-T path cost on Trunk port 1 of 19.
- VLANs 8 through 10 are assigned a path cost of 30 on Trunk port 2.
- VLANs 2 through 4 retain the default 100Base-T path cost on Trunk port 2 of 19.

Figure 7-3 Load-Sharing Trunks with Traffic Distributed by Path Cost


Switch A
Trunk port 1
VLANs 2 – 4 (path cost 30)
VLANs 8 – 10 (path cost 19)
Trunk port 2
VLANs 8 – 10 (path cost 30)
VLANs 2 – 4 (path cost 19)
Switch B
90573

Beginning in privileged EXEC mode, follow these steps to configure the network shown in Figure 7-3:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode on Switch A.
Step 2	**interface fastethernet0/1**	Define the interface to be configured as Trunk port 1, and enter interface configuration mode.
Step 3	**switchport mode trunk**	Configure the port as a trunk port.
Step 4	**exit**	Return to global configuration mode.
Step 5	**interface fastethernet0/2**	Define the interface to be configured as Trunk port 2, and enter interface configuration mode.
Step 6	**switchport mode trunk**	Configure the port as a trunk port.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show running-config**	Verify your entries. In the display, make sure that the interfaces configured in Steps 2 and 7 are configured as trunk ports.
Step 9	**show vlan**	Verify that VLANs 2 through 4 and 8 through 10 are configured on Switch A. If not, create these VLANs.
Step 10	**configure terminal**	Enter global configuration mode.
Step 11	**interface fastethernet0/1**	Enter interface configuration mode for Trunk port 2.
Step 12	**spanning-tree vlan 2-4 cost 30**	Set the spanning-tree path cost to 30 for VLANs 2 through 4.

	Command	Purpose
Step 13	**exit**	Return to global configuration mode.
Step 14	**interface fastethernet0/2**	Enter interface configuration mode for Trunk port 2.
Step 15	**spanning-tree vlan 8-10 cost 30**	Set the spanning-tree path cost to 30 for VLANs 2 through 4.
Step 16	**exit**	Return to global configuration mode.
Step 17		Repeat Steps 9 through 11 on the other configured trunk interface on Switch A, and set the spanning-tree path cost to 30 for VLANs 8, 9, and 10.
Step 18	**exit**	Return to privileged EXEC mode.
Step 19	**show running-config**	Verify your entries. In the display, verify that the path costs are set correctly for both trunk interfaces.
Step 20	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Follow the same steps on Switch B to configure the trunk port for Trunk 1 with a path cost of 30 for VLANs 2 through 4, and configure the trunk port for Trunk 2 with a path cost of 30 for VLANs 8 through 10.

CHAPTER 8

Configuring IEEE 802.1Q Tunneling and Layer 2 Protocol Tunneling

VPNs provide enterprise-scale connectivity on a shared infrastructure, often Ethernet-based, with the same security, prioritization, reliability, and manageability requirements of private networks. Tunneling is a feature designed for service providers who carry traffic of multiple customers across their networks and are required to maintain the VLAN and Layer 2 protocol configurations of each customer without impacting the traffic of other customers. The Cisco MWR 2941 router supports IEEE 802.1Q tunneling and Layer 2 protocol tunneling.

Note Release 15.0(1)MR does not support the 802.1ad standard.

Note For complete syntax and usage information for the commands used in this chapter, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.

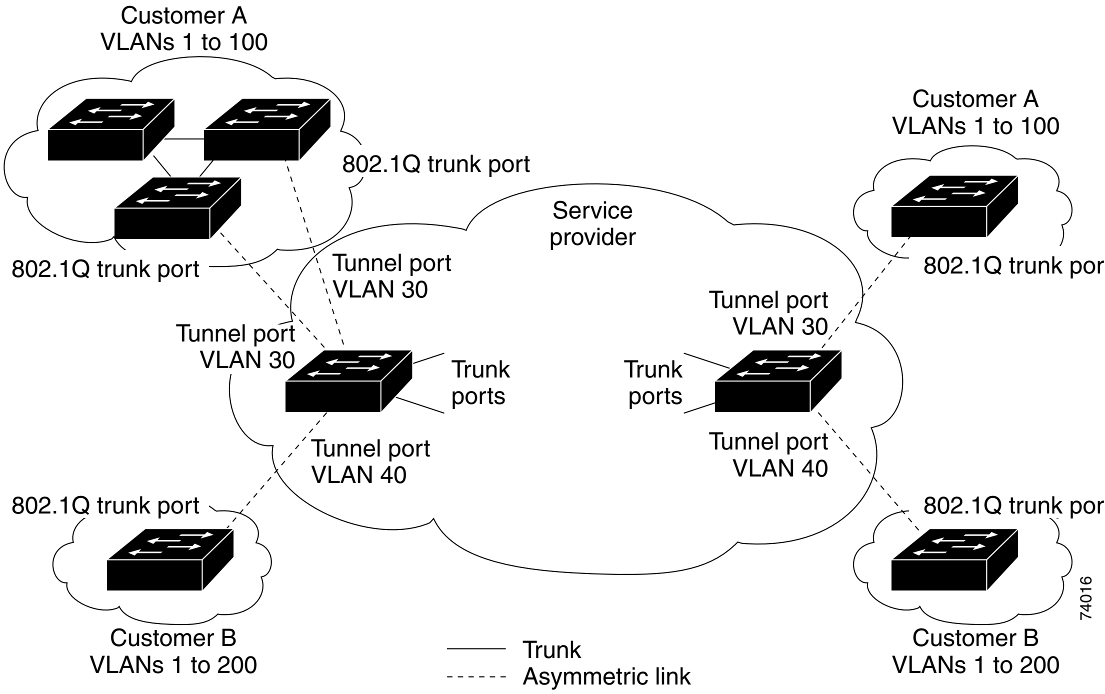
Understanding 802.1Q Tunneling

Business customers of service providers often have specific requirements for VLAN IDs and the number of VLANs to be supported. The VLAN ranges required by different customers in the same service-provider network might overlap, and traffic of customers through the infrastructure might be mixed. Assigning a unique range of VLAN IDs to each customer would restrict customer configurations and could easily exceed the VLAN limit (4096) of the 802.1Q specification.

Using the 802.1Q tunneling (QinQ) feature, service providers can use a single VLAN to support customers who have multiple VLANs. Customer VLAN IDs (C-VLANs) are preserved, and traffic from different customers is segregated within the service-provider network, even when they appear to be in the same VLAN. Using 802.1Q tunneling expands VLAN space by using a VLAN-in-VLAN hierarchy and retagging the tagged packets. A port configured to support 802.1Q tunneling is called a *tunnel port*. When you configure tunneling, you assign a tunnel port to a VLAN ID that is dedicated to tunneling. Each customer requires a separate service-provider VLAN ID (S-VLAN), but that VLAN ID supports all of the customer's VLANs. Configuring 802.1Q tunneling on a tunnel port is referred to as *traditional QinQ*.

Customer traffic tagged in the normal way with appropriate VLAN IDs comes from an 802.1Q trunk port on the customer device and into a tunnel port on the service-provider edge switch. The link between the customer device and the edge switch is asymmetric because one end is configured as an 802.1Q trunk port, and the other end is configured as a tunnel port. You assign the tunnel port interface to an access VLAN ID that is unique to each customer. See Figure 8-1.

Figure 8-1 802.1Q Tunnel Ports in a Service-Provider Network

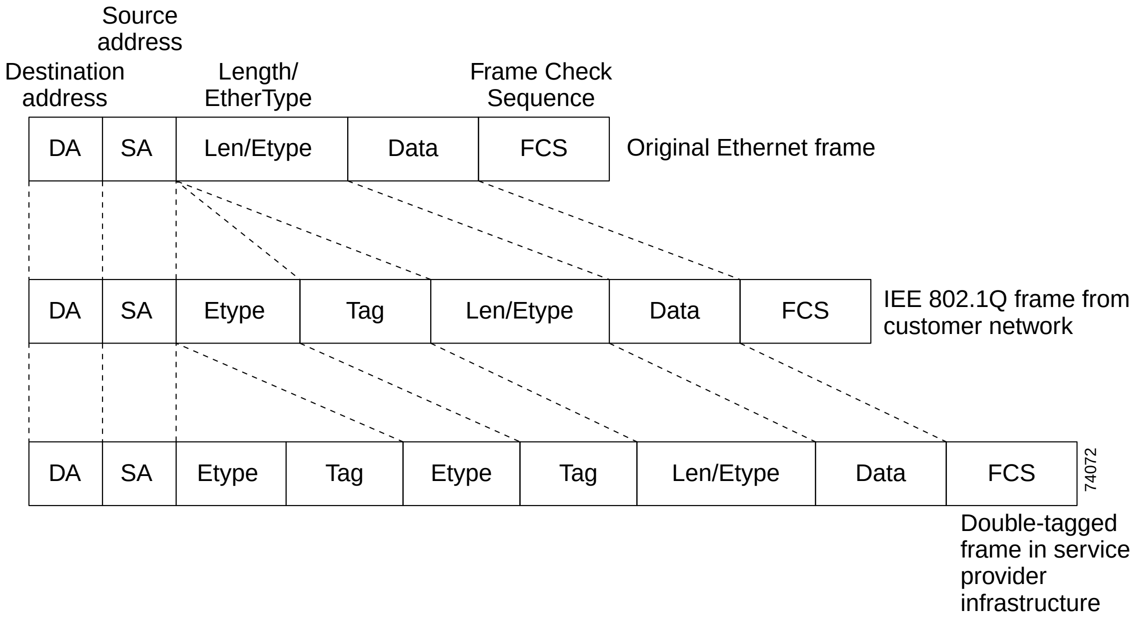


Packets coming from the customer trunk port into the tunnel port on the service-provider edge switch are normally 802.1Q-tagged with the appropriate VLAN ID. The the tagged packets remain intact inside the switch and when they exit the trunk port into the service-provider network, they are encapsulated with another layer of an 802.1Q tag (called the *metro tag*) that contains the VLAN ID that is unique to the customer. The original customer 802.1Q tag is preserved in the encapsulated packet. Therefore, packets entering the service-provider network are double-tagged, with the outer (metro) tag containing the customer's access VLAN ID, and the inner VLAN ID being that of the incoming traffic.

When the double-tagged packet enters another trunk port in a service-provider core switch, the outer tag is stripped as the switch processes the packet. When the packet exits another trunk port on the same core switch, the same metro tag is again added to the packet. Figure 8-2 shows the tag structures of the double-tagged packets.

Note Remove the Layer 2 protocol configuration from a trunk port because incoming encapsulated packets change that trunk port to error disabled. The outgoing encapsulated VTP (CDP and STP) packets are dropped on that trunk.

Figure 8-2 Original (Normal), 802.1Q, and Double-Tagged Ethernet Packet Formats



When the packet enters the trunk port of the service-provider egress switch, the outer tag is again stripped as the switch internally processes the packet. However, the metro tag is not added when the packet is sent out the tunnel port on the edge switch into the customer network. The packet is sent as a normal 802.1Q-tagged frame to preserve the original VLAN numbers in the customer network.

In Figure 8-1, Customer A was assigned VLAN 30, and Customer B was assigned VLAN 40. Packets entering the edge switch tunnel ports with 802.1Q tags are double-tagged when they enter the service-provider network, with the outer tag containing VLAN ID 30 or 40, appropriately, and the inner tag containing the original VLAN number, for example, VLAN 100. Even if both Customers A and B have VLAN 100 in their networks, the traffic remains segregated within the service-provider network because the outer tag is different. Each customer controls its own VLAN numbering space, which is independent of the VLAN numbering space used by other customers and the VLAN numbering space used by the service-provider network.

At the outbound tunnel port, the original VLAN numbers on the customer's network are recovered. It is possible to have multiple levels of tunneling and tagging.

Note The Cisco MWR 2941 currently supports only one level of tagging.

If traffic coming from a customer network is not tagged (native VLAN frames), these packets are bridged or routed as normal packets. All packets entering the service-provider network through a tunnel port on an edge switch are treated as untagged packets, whether they are untagged or already tagged with 802.1Q headers. The packets are encapsulated with the metro tag VLAN ID (set to the access VLAN of the tunnel port) when they are sent through the service-provider network on an 802.1Q trunk port. The priority field on the metro tag is set to match the CoS field of the inner tag (or customer tag) by default.

Configuring 802.1Q Tunneling

- Default 802.1Q Tunneling Configuration, page 8-4
- 802.1Q Tunneling Configuration Guidelines, page 8-4
- 802.1Q Tunneling and Other Features, page 8-6
- Configuring an 802.1Q Tunneling Port, page 8-6

Default 802.1Q Tunneling Configuration

By default, 802.1Q tunneling is disabled because the default switchport mode is access. Tagging of 802.1Q native VLAN packets on all 802.1Q trunk ports is also disabled. By default, VLANs on the router are dot1q tunnel ports.

802.1Q Tunneling Configuration Guidelines

When you configure 802.1Q tunneling, you should always use an asymmetrical link between the customer device and the edge switch, with the customer device port configured as an 802.1Q trunk port and the edge switch port configured as a tunnel port.

Assign tunnel ports only to VLANs that are used for tunneling.

The following sections explain the configuration requirements for native VLANs and maximum transmission units (MTUs).

Native VLANs

When configuring 802.1Q tunneling on an edge switch, you must use 802.1Q trunk ports for sending packets into the service-provider network. However, packets going through the core of the service-provider network can be carried through 802.1Q trunks or nontrunking links. When 802.1Q trunks are used in these core switches, the native VLANs of the 802.1Q trunks must not match any native VLAN of the nontrunking (tunneling) port on the same switch because traffic on the native VLAN would not be tagged on the 802.1Q sending trunk port.

See Figure 8-3. VLAN 40 is configured as the native VLAN for the 802.1Q trunk port from Customer X at the ingress edge switch in the service-provider network (Switch B). Switch A of Customer X sends a tagged packet on VLAN 30 to the ingress tunnel port of Switch B in the service-provider network, which belongs to access VLAN 40. Because the access VLAN of the tunnel port (VLAN 40) is the same as the native VLAN of the edge-switch trunk port (VLAN 40), the metro tag is not added to tagged packets received from the tunnel port. The packet carries only the VLAN 30 tag through the service-provider network to the trunk port of the egress-edge switch (Switch C) and is misdirected through the egress switch tunnel port to Customer Y.

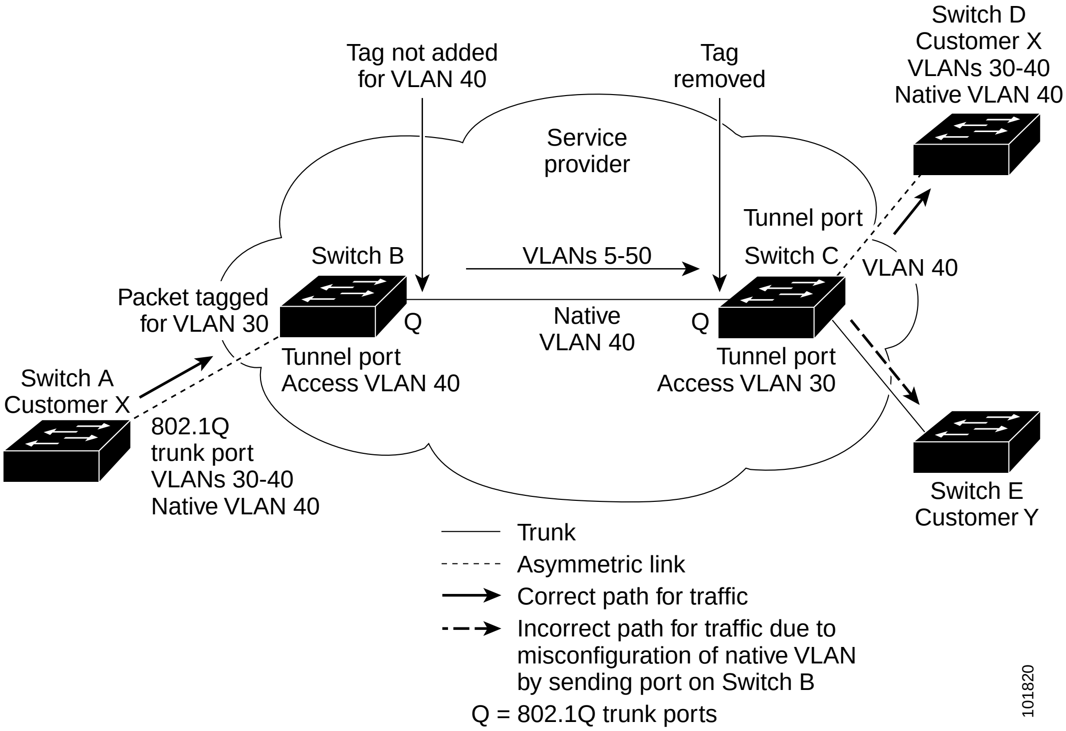
These are some ways to solve this problem:

- Use ISL trunks between core switches in the service-provider network. Although customer interfaces connected to edge switches must be 802.1Q trunks, we recommend using ISL trunks for connecting switches in the core layer.

Note The Cisco MWR 2941 router does not support ISL trunks.

- Use the **vlan dot1q tag native** global configuration command to configure the edge switch so that all packets going out an 802.1Q trunk, including the native VLAN, are tagged. If the switch is configured to tag native VLAN packets on all 802.1Q trunks, the switch accepts untagged packets, but sends only tagged packets.
- Ensure that the native VLAN ID on the edge-switch trunk port is not within the customer VLAN range. For example, if the trunk port carries traffic of VLANs 100 to 200, assign the native VLAN a number outside that range.

Figure 8-3 Potential Problem with 802.1Q Tunneling and Native VLANs



System MTU

The default MTU size for Gigabit Ethernet ports is 9216 bytes and is a fixed value. Release 15.0(1)MR does not support the **system mtu** or **system mtu jumbo** command.

802.1Q Tunneling and Other Features

Although 802.1Q tunneling works well for Layer 2 packet switching, there are incompatibilities between some Layer 2 features and Layer 3 switching.

Routing on VLANs with 802.1Q Tunnel Ports

IP routing is not supported on a VLAN that includes 802.1Q tunnel ports. Packets received from a tunnel port are forwarded based only on Layer 2 information. The Cisco MWR 2941 does not support routing on switch virtual interfaces (SVIs).

Configuring an 802.1Q Tunneling Port

Beginning in privileged EXEC mode, follow these steps to configure a port as an 802.1Q tunnel port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode for the interface to be configured as a tunnel port. This should be the edge port in the service-provider network that connects to the customer switch.
Step 3	**no shutdown**	Enable the port, if necessary. NNIs are enabled by default.
Step 4	**switchport access vlan** *vlan-id*	Specify the default VLAN, which is used if the interface stops trunking. This VLAN ID is specific to the particular customer.
Step 5	**switchport mode dot1q-tunnel**	Set the interface as an 802.1Q tunnel port.
Step 6	**exit**	Return to global configuration mode.
Step 7	**vlan dot1q tag native**	(Optional) Set the router to enable tagging of native VLAN packets on all 802.1Q trunk ports. When not set, and a customer VLAN ID is the same as the native VLAN, the trunk port does not apply a metro tag, and packets could be sent to the wrong destination.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show running-config** **show dot1q-tunnel**	Display the ports configured for 802.1Q tunneling. Display the ports that are in tunnel mode.
Step 10	**show vlan dot1q tag native**	Display 802.1Q native VLAN tagging status.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no switchport mode dot1q-tunnel** interface configuration command to return the port to the default state of access. Use the **no vlan dot1q tag native** global configuration command to disable tagging of native VLAN packets.

This example shows how to configure an interface as a tunnel port, enable tagging of native VLAN packets, and verify the configuration. In this configuration, the VLAN ID for the customer connected to Gigabit Ethernet interface 2 is VLAN 22.

```
Router(config)# interface gigabitethernet0/2
Router(config-if)# switchport access vlan 22
% Access VLAN does not exist. Creating vlan 22
Router(config-if)# switchport mode dot1q-tunnel
Router(config-if)# exit
Router(config)# vlan dot1q tag native
Router(config)# end
Router# show dot1q-tunnel interface gigabitethernet0/2
dot1q-tunnel mode LAN Port(s)
-----------------------------
Gi0/1

Router# show vlan dot1q tag native
dot1q native vlan tagging is enabled
```

There is no special configuration for provider trunk ports if the S-tags (outer VLAN tags) used in the provider network have a TPID of 0x8100, the TPID used in 802.1Q tags. The provider trunk ports are configured as normal trunk mode switch ports. If one of the other well known tunneling TPIDs is required the **dot1q tunneling ethertype tpid** interface configuration mode command is used to change it. Valid values for tpid are 0x88A8 (IEEE 802.1ad), 0x9100 and 0x9200. Use the **no** form of this command to set the TPID back to the default setting of 0x8100.

Note The Cisco MWR 2941 does not currently support IEEE 802.1ad.

Understanding VLAN Mapping

Another way to establish S-VLANs is to configure VLAN mapping (or VLAN ID translation) on trunk ports connected to a customer network to map customer VLANs to service-provider VLANs. Packets entering the port are mapped to a service provider VLAN (S-VLAN) based on the port number and the original customer VLAN-ID (C-VLAN) of the packet. Because the VLAN ID is mapped to the S-VLAN on ingress, all forwarding operations are performed based on S-VLAN information.

Note The Cisco MWR 2941 only supports VLAN mapping on 802.1q tunnel ports.

Note When you configure features on a port that has VLAN mapping configured, you always use the S-VLAN (translated VLAN) ID, not the customer VLAN-ID (C-VLAN).

On an interface configured for VLAN mapping, the specified C-VLAN packets are mapped to the specified S-VLAN when they enter the port. Symmetrical mapping back to the customer C-VLAN occurs when packets exit the port.

There are three types of VLAN mapping on 802.1q tunnel ports:

- One-to-one VLAN mapping—Occurs at the ingress and egress of the port and maps the customer C-VLAN ID in the 802.1Q tag to the service-provider S-VLAN ID. You can also specify that packets with all other VLAN IDs are dropped. The Cisco MWR 2941 does not currently support this type of VLAN mapping.
- Selective QinQ—Maps the specified customer VLANs entering the tunnel port to the specified S-VLAN ID. The S-VLAN is added to the incoming unmodified C-VLAN.

- Traditional 802.1Q tunneling (QinQ)—Performs all-to-one bundling of C-VLAN IDs to a single S-VLAN ID for the port. The S-VLAN is added to the incoming unmodified C-VLAN. You can configure the port as an 802.1Q tunnel port for traditional QinQ, or you can configure selective QinQ for a more flexible implementation. Mapping takes place at ingress and egress of the port. All packets on the port are bundled into the specified S-VLAN.

Untagged packets enter the router on the trunk with the native VLAN and are not mapped.

Note The Cisco MWR 2941 does not support one-to-one VLAN mapping.

Note The Cisco MWR 2941 does not currently support ingress classification and marking on dot1q interfaces.

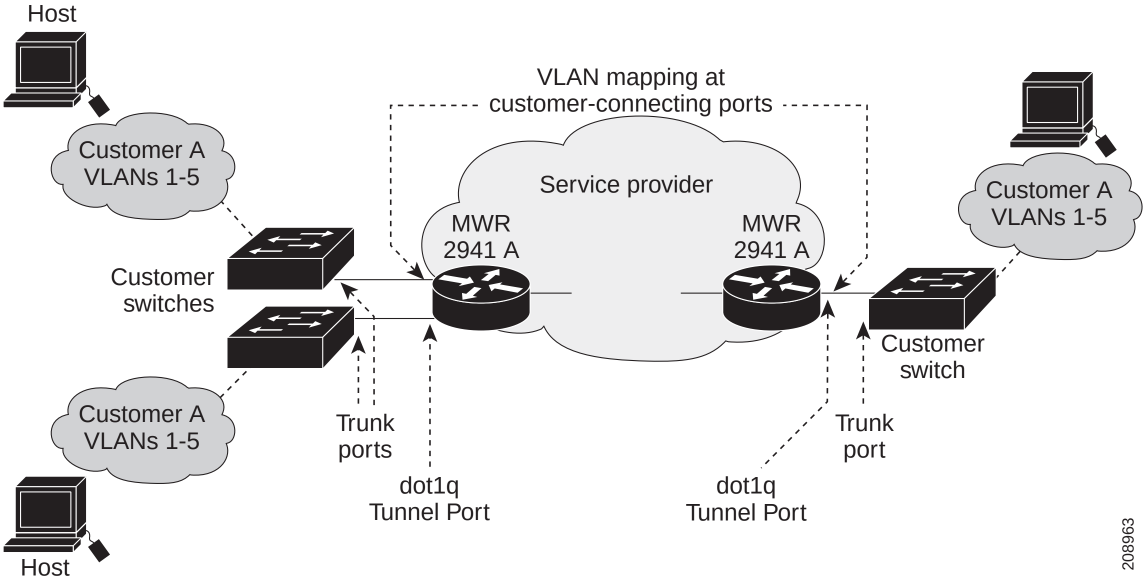
For information about Quality of Service, see the *Release Notes for Cisco MWR 2941 Mobile Wireless Edge Router for Cisco IOS Release 15.0(1)MR* and Chapter 24, "Configuring Quality of Service."

Mapping Customer VLANs to Service-Provider VLANs

Figure 8-4 shows a topology where a customer uses the same VLANs in multiple sites on different sides of a service-provider network. You map the customer VLAN IDs to service-provider VLAN IDs for packet travel across the service-provider backbone. The customer VLAN IDs are retrieved at the other side of the service-provider backbone for use in the other customer site. Configure the same set of VLAN mappings at a customer-connected port on each side of the service-provider network.

See the examples following the configuration steps for using one-to-one mapping, traditional QinQ, or selective QinQ to map customer VLANs 1 to 5 to service-provider VLANs.

Figure 8-4 Mapping Customer VLANs



Configuring VLAN Mapping

Default VLAN Mapping Configuration

By default, no VLAN mapping is configured.

VLAN Mapping Configuration Guidelines

- The Cisco MWR 2941 uses 802.1Q tunnel ports for both traditional and selective QinQ. VLAN mapping is only supported on 802.1Q tunnel ports.
- To avoid mixing customer traffic, when you configure traditional QinQ on a trunk port, you should configure the service provider S-VLAN ID as an allowed VLAN on the trunk port.
- When you configure selective QinQ to tunnel the traffic of two different customers on different S-VLANs, if the native VLAN (VLAN 1) is on one of the selective QinQ interfaces, untagged CDP and STP VLAN 1 packets are leaked to the other customer switches. The workaround is to use the **vlan dot1q tag native** configuration command to configure the native VLAN ID on an interface tunneling S-VLANs. For example, if you configured QinQ by entering the **switchport vlan mapping 1 500** command, you should also enter the **vlan dot1q tag native** command.

Configuring VLAN Mapping

These procedures show how to configure each type of VLAN mapping on trunk ports. To verify your configuration, enter the **show running-config interface** privileged EXEC command with the **interface** *type number* keyword. See the "Monitoring and Maintaining Tunneling and Mapping Status" section on page 8-16 for the syntax of these commands. For more information about all commands in this section, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.

Configuring Traditional QinQ on an 802.1Q Tunnel Port

Beginning in privileged EXEC mode, follow these steps to configure VLAN mapping for traditional QinQ on a tunnel port or tunneling by default. Configuring tunneling by default bundles all packets on the port into the configured S-VLAN.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode for the interface connected to the service-provider network.
Step 3	**switchport mode dot1q-tunnel**	Configure the interface as a 802.1q tunnel port.
Step 4	**switchport access vlan** *vlan-id*	Specify an outer VLAN ID to assign to all packets on the switch port.
Step 5	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 6	**show running-config interface** *type number*	Verify the configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example shows how to bundle all traffic on the port to leave the router with the S-VLAN ID of 100.

```
Router(config)# interface gigabiethernet0/1
Router(config-if)# switchport mode dot1q-tunnel
Router(config-if)# switchport access vlan 100
Router(config-if)# exit
```

Configuring Selective QinQ on a Tunnel Port

Beginning in privileged EXEC mode, follow these steps to configure VLAN mapping for selective QinQ on a trunk port. Note that you can configure one-to-one mapping and selective QinQ on the same interface, but you cannot use the same C-VLAN IDs in both configurations.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode for the interface connected to the service-provider network.
Step 3	**switchport mode dot1q-tunnel**	Configure the interface as a 802.1Q port.
Step 4	**switchport vlan mapping** *original-vlan-id translated-vlan-id*	Enter the VLAN IDs to be mapped: • *vlan-id*—the customer VLAN ID (C-VLAN) entering the router from the customer network. The range is from 1 to 4094. You can enter a string of VLAN-IDs. • *outer-vlan-id*—Enter the outer VLAN ID (S-VLAN) of the service-provider network. The range is from 1 to 4094.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config interface** *type number*	Verify the configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no switchport vlan mapping** *vlan-id outer vlan-id* command to remove the VLAN mapping configuration. Entering **no switchport vlan mapping all** deletes all mapping configurations.

This example shows how to configure selective QinQ mapping on the port so that traffic with a C-VLAN ID of 1 to 5 enters the router with an S-VLAN ID of 100. The traffic of any other VLAN IDs is tagged as VLAN 123.

```
Router(config)# interface gigabiethernet0/1
Router(config-if)# switchport mode dot1q-tunnel
Router(config-if)# switchport access vlan 123
Router(config-if)# switchport vlan mapping 1 100
Router(config-if)# switchport vlan mapping 2 100
Router(config-if)# switchport vlan mapping 3 100
Router(config-if)# switchport vlan mapping 4 100
Router(config-if)# switchport vlan mapping 5 100
Router(config-if)# exit
```

Understanding Layer 2 Protocol Tunneling

Customers at different sites connected across a service-provider network need to use various Layer 2 protocols to scale their topologies to include all remote sites, as well as the local sites. STP must run properly, and every VLAN should build a proper spanning tree that includes the local site and all remote sites across the service-provider network. Cisco Discovery Protocol (CDP) must discover neighboring Cisco devices from local and remote sites. VLAN Trunking Protocol (VTP) must provide consistent VLAN configuration throughout all sites in the customer network that are participating in VTP.

When protocol tunneling is enabled, edge switches on the inbound side of the service-provider network encapsulate Layer 2 protocol packets with a special MAC address and send them across the service-provider network. Core switches in the network do not process these packets but forward them as normal packets. Layer 2 protocol data units (PDUs) for CDP, STP, or VTP cross the service-provider network and are delivered to customer switches on the outbound side of the service-provider network. Identical packets are received by all customer ports on the same VLANs with these results:

- Users on each of a customer's sites can properly run STP, and every VLAN can build a correct spanning tree based on parameters from all sites and not just from the local site.
- CDP discovers and shows information about the other Cisco devices connected through the service-provider network.
- VTP provides consistent VLAN configuration throughout the customer network, propagating to all switches through the service provider that support VTP.

Note To provide interoperability with third-party vendors, you can use the Layer 2 protocol-tunnel bypass feature. Bypass mode transparently forwards control PDUs to vendor switches that have different ways of controlling protocol tunneling. You implement bypass mode by enabling Layer 2 protocol tunneling on the egress trunk port. When Layer 2 protocol tunneling is enabled on the trunk port, the encapsulated tunnel MAC address is removed and the protocol packets have their normal MAC address.

Layer 2 protocol tunneling can be used independently or can enhance 802.1Q tunneling. If protocol tunneling is not enabled on 802.1Q tunneling ports, remote switches at the receiving end of the service-provider network do not receive the PDUs and cannot properly run STP, CDP, and VTP. When protocol tunneling *is* enabled, Layer 2 protocols within each customer's network are totally separate from those running within the service-provider network. Customer switches on different sites that send traffic through the service-provider network with 802.1Q tunneling achieve complete knowledge of the customer's VLAN. If 802.1Q tunneling is not used, you can still enable Layer 2 protocol tunneling by connecting to the customer switch through access or trunk ports and enabling tunneling on the service-provider access or trunk port.

For example, in Figure 8-5, Customer X has four switches in the same VLAN, that are connected through the service-provider network. If the network does not tunnel PDUs, switches on the far ends of the network cannot properly run STP, CDP, and VTP. For example, STP for a VLAN on a switch in Customer X, Site 1, will build a spanning tree on the switches at that site without considering convergence parameters based on Customer X's switch in Site 2. This could result in the topology shown in Figure 8-6.

Figure 8-5 Layer 2 Protocol Tunneling

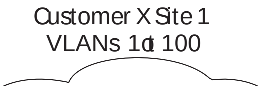
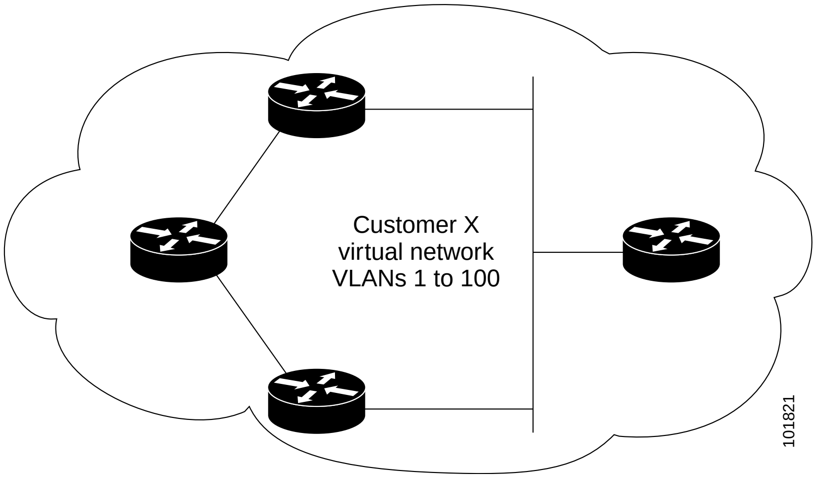


Figure 8-6 Layer 2 Network Topology without Proper Convergence



Configuring Layer 2 Protocol Tunneling

You can enable Layer 2 protocol tunneling (by protocol) on the ports that are connected to the customer in the edge switches of the service-provider network. The service-provider edge switches connected to the customer switch perform the tunneling process. Edge-switch tunnel ports are connected to customer 802.1Q trunk ports. Edge-switch access ports are connected to customer access ports. The edge switches connected to the customer switch perform the tunneling process.

You can enable Layer 2 protocol tunneling on ports that are configured as access ports, tunnel ports, or trunk ports. The switch supports Layer 2 protocol tunneling for CDP, STP, and VTP. The switch does not support PAgP, LACP, and UDLD protocols for emulated point-to-point network topologies or Layer 2 protocol tunneling for LLDP.

When the Layer 2 PDUs that entered the service-provider inbound edge switch through a Layer 2 protocol-enabled port exit through the trunk port into the service-provider network, the switch overwrites the customer PDU-destination MAC address with a well-known Cisco proprietary multicast address (01-00-0c-cd-cd-d0). If 802.1Q tunneling is enabled, packets are also double-tagged; the outer tag is the customer metro tag, and the inner tag is the customer's VLAN tag. The core switches ignore the inner tags and forward the packet to all trunk ports in the same metro VLAN. The edge switches on the outbound side restore the proper Layer 2 protocol and MAC address information and forward the packets to all Layer 2 protocol-enabled access ports, tunnel ports, and trunk ports in the same metro VLAN. Therefore, the Layer 2 PDUs remain intact and are delivered across the service-provider infrastructure to the other side of the customer network.

See Figure 8-5, with Customer X and Customer Y in access VLANs 30 and 40, respectively. Asymmetric links connect the customers in Site 1 to edge switches in the service-provider network. The Layer 2 PDUs (for example, BPDUs) coming into Switch B from Customer Y in Site 1 are forwarded to the infrastructure as double-tagged packets with the well-known MAC address as the destination MAC address. These double-tagged packets have the metro VLAN tag of 40, as well as an inner VLAN tag (for example, VLAN 100). When the double-tagged packets enter Switch D, the outer VLAN tag 40 is removed, the well-known MAC address is replaced with the respective Layer 2 protocol MAC address, and the packet is sent to Customer Y on Site 2 as a single-tagged frame in VLAN 100.

You can also enable Layer 2 protocol tunneling on access ports on the edge switch connected to access or trunk ports on the customer switch. In this case, the encapsulation and decapsulation process is the same as described in the previous paragraph, except that the packets are not double-tagged in the service-provider network. The single tag is the customer-specific access VLAN tag.

These sections contain this configuration information:

- Default Layer 2 Protocol Tunneling Configuration, page 8-13
- Layer 2 Protocol Tunneling Configuration Guidelines, page 8-14
- Configuring Layer 2 Protocol Tunneling, page 8-15

Default Layer 2 Protocol Tunneling Configuration

Table 8-1 shows the default Layer 2 protocol tunneling configuration.

Table 8-1 Default Layer 2 Ethernet Interface VLAN Configuration

Feature	Default Setting
Layer 2 protocol tunneling	Disabled.
Shutdown threshold	None set.

Table 8-1 Default Layer 2 Ethernet Interface VLAN Configuration (continued)

Feature	Default Setting
Drop threshold	None set.
CoS value	If a CoS value is configured on the interface, that value is used to set the BPDU CoS value for Layer 2 protocol tunneling. If no CoS value is configured at the interface level, the default value for CoS marking of L2 protocol tunneling BPDUs is 5. This does not apply to data traffic.

Layer 2 Protocol Tunneling Configuration Guidelines

These are some configuration guidelines and operating characteristics of Layer 2 protocol tunneling:

- The router supports tunneling of CDP, STP, including multiple STP (MSTP), and VTP. Protocol tunneling is disabled by default but can be enabled for the individual protocols on 802.1Q tunnel ports, access ports. or trunk ports.
- The edge switches on the outbound side of the service-provider network restore the proper Layer 2 protocol and MAC address information and forward the packets to all Layer 2 protocol-enabled tunnel, access, and trunk ports in the same metro VLAN.
- For interoperability with third-party vendor switches, the switch supports a Layer 2 protocol-tunnel bypass feature. Bypass mode transparently forwards control PDUs to vendor switches that have different ways of controlling protocol tunneling.When Layer 2 protocol tunneling is enabled on ingress ports on a switch, egress trunk ports forward the tunneled packets with a special encapsulation. If you also enable Layer 2 protocol tunneling on the egress trunk port, this behavior is bypassed, and the switch forwards control PDUs without any processing or modification.
- If an encapsulated PDU (with the proprietary destination MAC address) is received from a tunnel port or access or trunk port with Layer 2 tunneling enabled, the tunnel port is shut down to prevent loops. The port also shuts down when a configured shutdown threshold for the protocol is reached. You can manually re-enable the port (by entering a **shutdown** and a **no shutdown** command sequence). If errdisable recovery is enabled, the operation is retried after a specified time interval.
- Only decapsulated PDUs are forwarded to the customer network. The spanning-tree instance running on the service-provider network does not forward BPDUs to tunnel ports. CDP packets are not forwarded from tunnel ports.
- When protocol tunneling is enabled on an interface, you can set a per-protocol, per-port, shutdown threshold for the PDUs generated by the customer network. If the limit is exceeded, the port shuts down. You can also limit BPDU rate by using QoS ACLs and policy maps on a tunnel port.
- When protocol tunneling is enabled on an interface, you can set a per-protocol, per-port, drop threshold for the PDUs generated by the customer network. If the limit is exceeded, the port drops PDUs until the rate at which it receives them is below the drop threshold.
- Because tunneled PDUs (especially STP BPDUs) must be delivered to all remote sites so that the customer virtual network operates properly, you can give PDUs higher priority within the service-provider network than data packets received from the same tunnel port. By default, the PDUs use the same CoS value as data packets.

Configuring Layer 2 Protocol Tunneling

Beginning in privileged EXEC mode, follow these steps to configure a port for Layer 2 protocol tunneling:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and enter the interface to be configured as a tunnel port. This should be the edge port in the service-provider network that connects to the customer switch.
Step 3	**no shutdown**	Enable the port, if necessary. NNIs are enabled by default.
Step 4	**switchport mode access** or **switchport mode dot1q-tunnel** or **switchport mode trunk**	Configure the interface as an access port, an 802.1Q tunnel port or a trunk port. The default switchport mode is access.
Step 5	**l2protocol-tunnel** [**cdp** | **stp** | **vtp**]	Enable protocol tunneling for the desired protocol. If no keyword is entered, tunneling is enabled for all three Layer 2 protocols.
Step 6	**l2protocol-tunnel shutdown-threshold** [**cdp** | **stp** | **vtp**] *value*	(Optional) Configure the threshold for packets-per-second accepted for encapsulation. The interface is disabled if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured. **Note** If you also set a drop threshold on this interface, the shutdown-threshold value must be greater than or equal to the drop-threshold value.
Step 7	**l2protocol-tunnel drop-threshold** [**cdp** | **stp** | **vtp**] *value*	(Optional) Configure the threshold for packets-per-second accepted for encapsulation. The interface drops packets if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured. If you also set a shutdown threshold on this interface, the drop-threshold value must be less than or equal to the shutdown-threshold value.
Step 8	**exit**	Return to global configuration mode.
Step 9	**errdisable recovery cause l2ptguard**	(Optional) Configure the recovery mechanism from a Layer 2 maximum-rate error so that the interface is re-enabled and can try again. Errdisable recovery is disabled by default; when enabled, the default time interval is 300 seconds.
Step 10	**l2protocol-tunnel cos** *value*	(Optional) Configure the CoS value for all tunneled Layer 2 PDUs. The range is 0 to 7; the default is the default CoS value for the interface. If none is configured, the default is 5.
Step 11	**end**	Return to privileged EXEC mode.
Step 12	**show l2protocol**	Display the Layer 2 tunnel ports on the router, including the protocols configured, the thresholds, and the counters.
Step 13	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no l2protocol-tunnel** [**cdp** | **stp** | **vtp**] interface configuration command to disable protocol tunneling for one of the Layer 2 protocols or for all three. Use the **no l2protocol-tunnel shutdown-threshold** [**cdp** | **stp** | **vtp**] and the **no l2protocol-tunnel drop-threshold** [**cdp** | **stp** | **vtp**] commands to return the shutdown and drop thresholds to the default settings.

This example shows how to configure Layer 2 protocol tunneling for CDP, STP, and VTP and to verify the configuration.

```
Router(config)# interface gigatethernet0/1
Router(config-if)# l2protocol-tunnel cdp
Router(config-if)# l2protocol-tunnel stp
Router(config-if)# l2protocol-tunnel vtp
Router(config-if)# l2protocol-tunnel shutdown-threshold 1500
Router(config-if)# l2protocol-tunnel drop-threshold 1000
Router(config-if)# exit
Router(config)# l2protocol-tunnel cos 7
Router(config)# end
Router# show l2protocol
COS for Encapsulated Packets: 7
Port      Protocol Shutdown  Drop      Encapsulation Decapsulation Drop
                   Threshold Threshold Counter       Counter       Counter
-------   -------- --------- --------- ------------- ------------- -------------
Gi  0/1  cdp           1500      1000 2288          2282          0
          stp           1500      1000 116           13            0
          vtp           1500      1000 3             67            0
```

Monitoring and Maintaining Tunneling and Mapping Status

Table 8-2 shows the privileged EXEC commands for monitoring and maintaining 802.1Q and Layer 2 protocol tunneling and VLAN mapping.

Table 8-2 Commands for Monitoring and Maintaining Tunneling

Command	Purpose
clear l2protocol-tunnel counters	Clear the protocol counters on Layer 2 protocol tunneling ports.
show dot1q-tunnel	Display 802.1Q tunnel ports on the router.
show dot1q-tunnel interface *interface-id*	Verify if a specific interface is a tunnel port.
show running-config interface *type number*	Verify the configuration for a specified interface. You can use this command to display the mapping configuration for a VLAN interface.
show l2protocol-tunnel	Display information about Layer 2 protocol tunneling ports.
show errdisable recovery	Verify if the recovery timer from a Layer 2 protocol-tunnel error disable state is enabled.
show l2protocol-tunnel interface *interface-id*	Display information about a specific Layer 2 protocol tunneling port.
show l2protocol-tunnel summary	Display only Layer 2 protocol summary information.
show vlan dot1q tag native	Display the status of native VLAN tagging on the router.

For detailed information about these displays, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.

CHAPTER 9

Configuring STP

This chapter describes how to configure the Spanning Tree Protocol (STP) on port-based VLANs on the Cisco MWR 2941 router. The router can use the per-VLAN spanning-tree plus (PVST+) protocol based on the IEEE 802.1D standard and Cisco proprietary extensions, or the rapid per-VLAN spanning-tree plus (rapid-PVST+) protocol based on the IEEE 802.1w standard. On the Cisco MWR 2941 router, STP is enabled by default on network node interfaces (NNIs). It is disabled by default, but can be enabled, on enhanced network interfaces (ENIs). User network interfaces (UNIs) on the router do not participate in STP. UNIs and ENIs on which STP is not enabled immediately forward traffic when they are brought up.

For information about the Multiple Spanning Tree Protocol (MSTP) and how to map multiple VLANs to the same spanning-tree instance, see Chapter 10, "Configuring MSTP." For information about other spanning-tree features including Port Fast and root guard, see Chapter 11, "Configuring Optional Spanning-Tree Features."

For detailed information about the commands used in this chapter, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.

This chapter consists of these sections:

- Understanding Spanning-Tree Features, page 9-1
- Configuring Spanning-Tree Features, page 9-10
- Displaying the Spanning-Tree Status, page 9-23

Understanding Spanning-Tree Features

- STP Overview, page 9-2
- Spanning-Tree Topology and BPDUs, page 9-2
- Bridge ID, Switch Priority, and Extended System ID, page 9-3
- Spanning-Tree Interface States, page 9-4
- How a Switch or Port Becomes the Root Switch or Root Port, page 9-7
- Spanning Tree and Redundant Connectivity, page 9-7
- Spanning-Tree Address Management, page 9-8
- Accelerated Aging to Retain Connectivity, page 9-8
- Spanning-Tree Modes and Protocols, page 9-9
- Supported Spanning-Tree Instances, page 9-10

- Spanning-Tree Interoperability and Backward Compatibility, page 9-10
- STP and IEEE 802.1Q Trunks, page 9-10

For configuration information, see the "Configuring Spanning-Tree Features" section on page 9-10.

For information about optional spanning-tree features, see Chapter 11, "Configuring Optional Spanning-Tree Features."

STP Overview

STP is a Layer 2 link management protocol that provides path redundancy while preventing loops in the network. For a Layer 2 Ethernet network to function properly, only one active path can exist between any two stations. Multiple active paths among end stations cause loops in the network. If a loop exists in the network, end stations might receive duplicate messages. Switches might also learn end-station MAC addresses on multiple Layer 2 interfaces. These conditions result in an unstable network. Spanning-tree operation is transparent to end stations, which cannot detect whether they are connected to a single LAN segment or a switched LAN of multiple segments.

The STP uses a spanning-tree algorithm to select one switch of a redundantly connected network as the root of the spanning tree. The algorithm calculates the best loop-free path through a switched Layer 2 network by assigning a role to each port based on the role of the port in the active topology:

- Root—A forwarding port elected for the spanning-tree topology
- Designated—A forwarding port elected for every switched LAN segment
- Alternate—A blocked port providing an alternate path to the root bridge in the spanning tree
- Backup—A blocked port in a loopback configuration

The device that has *all* of its ports as the designated switch role or the backup role is the root switch. The device that has at least *one* of its ports in the designated role is called the designated switch.

Spanning tree forces redundant data paths into a standby (blocked) state. If a network segment in the spanning tree fails and a redundant path exists, the spanning-tree algorithm recalculates the spanning-tree topology and activates the standby path. Switches send and receive spanning-tree frames, called bridge protocol data units (BPDUs), at regular intervals. The switches do not forward these frames but use them to construct a loop-free path. BPDUs contain information about the sending switch and its ports, including switch and MAC addresses, switch priority, port priority, and path cost. Spanning tree uses this information to elect the root switch and root port for the switched network and the root port and designated port for each switched segment.

When two ports on a switch are part of a loop, the spanning-tree port priority and path cost settings control which port is put in the forwarding state and which is put in the blocking state. The spanning-tree port priority value represents the location of a port in the network topology and how well it is located to pass traffic. The path cost value represents the media speed.

Spanning-Tree Topology and BPDUs

The stable, active spanning-tree topology of a switched network is controlled by these elements:

- The unique bridge ID (switch priority and MAC address) associated with each VLAN on each switch.
- The spanning-tree path cost to the root switch.
- The port identifier (port priority and MAC address) associated with each Layer 2 interface.

When the switches in a network are powered up, each functions as the root switch. Each switch sends a configuration BPDU through all of its ports, or on the Cisco MWR 2941 router, only through the ports. The BPDUs communicate and compute the spanning-tree topology. Each configuration BPDU contains this information:

- The unique bridge ID of the switch that the sending switch identifies as the root switch
- The spanning-tree path cost to the root
- The bridge ID of the sending switch
- Message age
- The identifier of the sending interface
- Values for the hello, forward delay, and max-age protocol timers

When a switch receives a configuration BPDU that contains *superior* information (lower bridge ID, lower path cost, and so forth), it stores the information for that port. If this BPDU is received on the root port of the switch, the switch also forwards it with an updated message to all attached LANs for which it is the designated switch.

If a switch receives a configuration BPDU that contains *inferior* information to that currently stored for that port, it discards the BPDU. If the switch is a designated switch for the LAN from which the inferior BPDU was received, it sends that LAN a BPDU containing the up-to-date information stored for that port. In this way, inferior information is discarded, and superior information is propagated on the network.

A BPDU exchange results in these actions:

- One switch in the network is elected as the root switch (the logical center of the spanning-tree topology in a switched network).

  For each VLAN, the switch with the highest switch priority (the lowest numerical priority value) is elected as the root switch. If all switches are configured with the default priority (32768), the switch with the lowest MAC address in the VLAN becomes the root switch. The switch priority value occupies the most significant bits of the bridge ID, as shown in Table 9-1 on page 9-4.
- A root port is selected for each switch (except the root switch). This port provides the best path (lowest cost) when the switch forwards packets to the root switch.
- The shortest distance to the root switch is calculated for each switch based on the path cost.
- A designated switch for each LAN segment is selected. The designated switch incurs the lowest path cost when forwarding packets from that LAN to the root switch. The port through which the designated switch is attached to the LAN is called the designated port. For the Cisco MWR 2941 router, this only applies to NNIs or to ENIs on which STP has been specifically enabled.

All paths that are not needed to reach the root switch from anywhere in the switched network are placed in the spanning-tree blocking mode.

Bridge ID, Switch Priority, and Extended System ID

The IEEE 802.1D standard requires that each switch has an unique bridge identifier (bridge ID), which controls the selection of the root switch. Because each VLAN is considered as a different *logical bridge* with PVST+, the same switch must have as many different bridge IDs as VLANs configured on it. Each VLAN on the switch has a unique 8-byte bridge ID. The two most-significant bytes are used for the switch priority, and the remaining six bytes are derived from the switch MAC address.

The switch supports the IEEE 802.1t spanning-tree extensions, and some of the bits previously used for the switch priority are now used as the VLAN identifier. The result is that fewer MAC addresses are reserved for the switch, and a larger range of VLAN IDs can be supported, all while maintaining the uniqueness of the bridge ID. As shown in Table 9-1, the two bytes previously used for the switch priority are reallocated into a 4-bit priority value and a 12-bit extended system ID value equal to the VLAN ID.

Table 9-1 Switch Priority Value and Extended System ID

Switch Priority Value				**Extended System ID (Set Equal to the VLAN ID)**											
Bit 16	**Bit 15**	**Bit 14**	**Bit 13**	**Bit 12**	**Bit 11**	**Bit 10**	**Bit 9**	**Bit 8**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**
32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1

Spanning tree uses the extended system ID, the switch priority, and the allocated spanning-tree MAC address to make the bridge ID unique for each VLAN.

Support for the extended system ID affects how you manually configure the root switch, the secondary root switch, and the switch priority of a VLAN. For example, when you change the switch priority value, you change the probability that the switch will be elected as the root switch. Configuring a higher value decreases the probability; a lower value increases the probability. For more information, see the "Configuring the Router as a Root Switch" section on page 9-15, the "Configuring a Secondary Root Switch" section on page 9-16, and the "Configuring the Switch Priority of a VLAN" section on page 9-20.

Spanning-Tree Interface States

Propagation delays can occur when protocol information passes through a switched LAN. As a result, topology changes can take place at different times and at different places in a switched network. When an STP port transitions directly from nonparticipation in the spanning-tree topology to the forwarding state, it can create temporary data loops. Interfaces must wait for new topology information to propagate through the switched LAN before starting to forward frames. They must allow the frame lifetime to expire for forwarded frames that have used the old topology.

Each Layer 2 interface on a switch using spanning tree exists in one of these states:

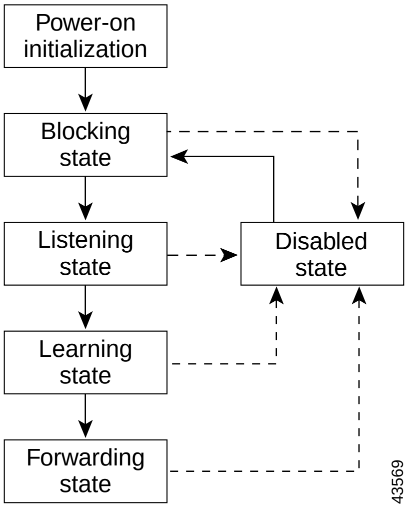
- Blocking—The interface does not participate in frame forwarding.
- Listening—The first transitional state after the blocking state when the spanning tree determines that the interface should participate in frame forwarding.
- Learning—The interface prepares to participate in frame forwarding.
- Forwarding—The interface forwards frames.
- Disabled—The interface is not participating in spanning tree because of a shutdown port, no link on the port, or no spanning-tree instance running on the port.

A port participating in spanning tree moves through these states:

- From initialization to blocking
- From blocking to listening or to disabled
- From listening to learning or to disabled
- From learning to forwarding or to disabled
- From forwarding to disabled

Figure 9-1 illustrates how an interface moves through the states.

Figure 9-1 Spanning-Tree Interface States



When you power up the router, spanning tree is enabled by default, and every NNI in the Cisco MWR 2941 router (and every ENI on which STP has been enabled), as well as any other port in other switches in the VLAN or network that are participating in spanning tree, goes through the blocking state and the transitory states of listening and learning. Spanning tree stabilizes each interface at the forwarding or blocking state.

When the spanning-tree algorithm places a Layer 2 spanning-tree interface in the forwarding state, this process occurs:

1. The interface is in the listening state while spanning tree waits for protocol information to transition the interface to the blocking state.
2. While spanning tree waits the forward-delay timer to expire, it moves the interface to the learning state and resets the forward-delay timer.
3. In the learning state, the interface continues to block frame forwarding as the router learns end-station location information for the forwarding database.
4. When the forward-delay timer expires, spanning tree moves the interface to the forwarding state, where both learning and frame forwarding are enabled.

Blocking State

A Layer 2 interface in the blocking state does not participate in frame forwarding. After initialization, a BPDU is sent to each switch interface, or to each switch STP port. A switch initially functions as the root until it exchanges BPDUs with other switches. This exchange establishes which switch in the network is the root or root switch. If there is only one switch in the network, no exchange occurs, the forward-delay timer expires, and the interface moves to the listening state. An interface participating in spanning tree always enters the blocking state after switch initialization.

An interface in the blocking state performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding

- Does not learn addresses
- Receives BPDUs

Listening State

The listening state is the first state a Layer 2 interface enters after the blocking state. The interface enters this state when the spanning tree decides that the interface should participate in frame forwarding.

An interface in the listening state performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Receives BPDUs

Learning State

A Layer 2 interface in the learning state prepares to participate in frame forwarding. The interface enters the learning state from the listening state.

An interface in the learning state performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Learns addresses
- Receives BPDUs

Forwarding State

A Layer 2 interface in the forwarding state forwards frames. The interface enters the forwarding state from the learning state.

An interface in the forwarding state performs these functions:

- Receives and forwards frames received on the interface
- Forwards frames switched from another interface
- Learns addresses
- Receives BPDUs

Disabled State

A Layer 2 interface in the disabled state does not participate in frame forwarding or in the spanning tree. An interface in the disabled state is nonoperational.

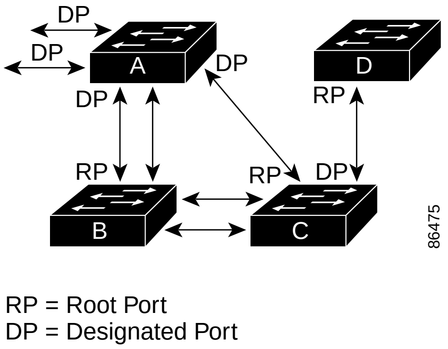
A disabled interface performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Does not receive BPDUs

How a Switch or Port Becomes the Root Switch or Root Port

If all switches in a network are enabled with default spanning-tree settings, the switch with the lowest MAC address becomes the root switch. In Figure 9-2, Switch A is elected as the root switch because the switch priority of all the switches is set to the default (32768) and Switch A has the lowest MAC address. However, because of traffic patterns, number of forwarding interfaces, or link types, Switch A might not be the ideal root switch. By increasing the priority (lowering the numerical value) of the ideal switch so that it becomes the root switch, you force a spanning-tree recalculation to form a new topology with the ideal switch as the root.

Figure 9-2 Spanning-Tree Topology



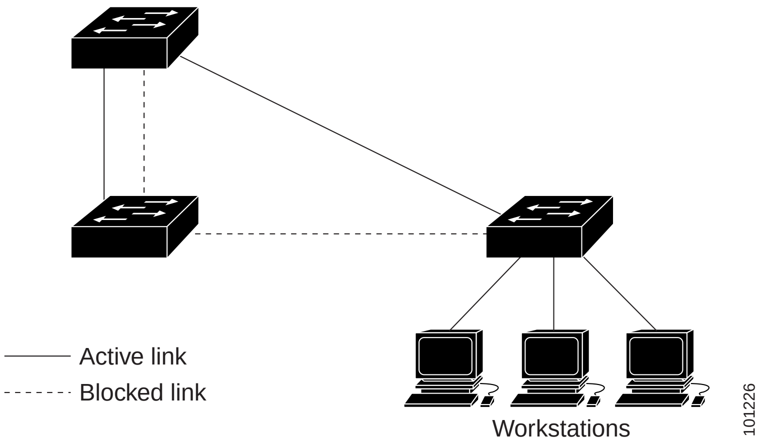
When the spanning-tree topology is calculated based on default parameters, the path between source and destination end stations in a switched network might not be ideal. For instance, connecting higher-speed links to an interface that has a higher number than the root port can cause a root-port change. The goal is to make the fastest link the root port.

For example, assume that one port on Switch B is a Gigabit Ethernet link and that another port on Switch B (a 10/100 link) is the root port. Network traffic might be more efficient over the Gigabit Ethernet link. By changing the spanning-tree port priority on the Gigabit Ethernet port to a higher priority (lower numerical value) than the root port, the Gigabit Ethernet port becomes the new root port.

Spanning Tree and Redundant Connectivity

You can create a redundant backbone with spanning tree by connecting two switch interfaces that are participating in spanning tree to another device or to two different devices, as shown in Figure 9-3. Spanning tree automatically disables one interface but enables it if the other one fails. If one link is high-speed and the other is low-speed, the low-speed link is always disabled. If the speeds are the same, the port priority and port ID are added together, and spanning tree disables the link with the lowest value.

Figure 9-3 Spanning Tree and Redundant Connectivity



Spanning-Tree Address Management

IEEE 802.1D specifies 17 multicast addresses, ranging from 0x00180C2000000 to 0x0180C2000010, to be used by different bridge protocols. These addresses are static addresses that cannot be removed.

Regardless of the spanning-tree state, each switch receives but does not forward packets destined for addresses between 0x0180C2000000 and 0x0180C200000F.

If spanning tree is enabled, the CPU on the switch receives packets destined for 0x0180C2000000 and 0x0180C2000010. If spanning tree is disabled, the switch forwards those packets as unknown multicast addresses.

Accelerated Aging to Retain Connectivity

The default for aging dynamic addresses is 5 minutes, the default setting of the **mac address-table aging-time** global configuration command. However, a spanning-tree reconfiguration can cause many station locations to change. Because these stations could be unreachable for 5 minutes or more during a reconfiguration, the address-aging time is accelerated so that station addresses can be dropped from the address table and then relearned. The accelerated aging is the same as the forward-delay parameter value (**spanning-tree vlan** *vlan-id* **forward-time** *seconds* global configuration command) when the spanning tree reconfigures.

Because each VLAN is a separate spanning-tree instance, the switch accelerates aging on a per-VLAN basis. A spanning-tree reconfiguration on one VLAN can cause the dynamic addresses learned on that VLAN to be subject to accelerated aging. Dynamic addresses on other VLANs can be unaffected and remain subject to the aging interval entered for the switch.

Spanning-Tree Modes and Protocols

The switch NNIs and ENIs with STP enabled support these spanning-tree modes and protocols:

- PVST+—This spanning-tree mode is based on the IEEE 802.1D standard and Cisco proprietary extensions. It is the default spanning-tree mode used on most Ethernet port-based VLANs. The PVST+ runs on each VLAN on the switch up to the maximum supported, ensuring that each has a loop-free path through the network.

  The PVST+ provides Layer 2 load balancing for the VLAN on which it runs. You can create different logical topologies by using the VLANs on your network to ensure that all of your links are used but that no one link is oversubscribed. Each instance of PVST+ on a VLAN has a single root switch. This root switch propagates the spanning-tree information associated with that VLAN to all other switches in the network. Because each switch has the same information about the network, this process ensures that the network topology is maintained.

- Rapid PVST+—This spanning-tree mode is the same as PVST+ except that is uses a rapid convergence based on the IEEE 802.1w standard. Rapid PVST+ is compatible with PVST+. To provide rapid convergence, the rapid PVST+ immediately deletes dynamically learned MAC address entries on a per-port basis upon receiving a topology change. By contrast, PVST+ uses a short aging time for dynamically learned MAC address entries.

  The rapid PVST+ uses the same configuration as PVST+ (except where noted), and the switch needs only minimal extra configuration. The benefit of rapid PVST+ is that you can migrate a large PVST+ install base to rapid PVST+ without having to learn the complexities of the MSTP configuration and without having to reprovision your network. In rapid-PVST+ mode, each VLAN runs its own spanning-tree instance up to the maximum supported.

Note The Cisco MWR 2941 does not currently support Rapid PVST+.

- MSTP—This spanning-tree mode is based on the IEEE 802.1s standard. You can map multiple VLANs to the same spanning-tree instance, which reduces the number of spanning-tree instances required to support a large number of VLANs. The MSTP runs on top of the RSTP (based on IEEE 802.1w), which provides for rapid convergence of the spanning tree by eliminating the forward delay and by quickly transitioning root ports and designated ports to the forwarding state. You cannot run MSTP without RSTP.

  The most common initial deployment of MSTP is in the backbone and distribution layers of a Layer 2 switched network. For more information, see Chapter 10, "Configuring MSTP."

For information about the number of supported spanning-tree instances, see the next section.

Supported Spanning-Tree Instances

In PVST+ mode, the switch supports up to 128 spanning-tree instances.

In MSTP mode, the switch supports up to 65 MST instances. You can map up to 255 VLANs to a particular MST instance.

Spanning-Tree Interoperability and Backward Compatibility

Table 9-2 lists the interoperability and compatibility among the supported spanning-tree modes in a network.

Table 9-2 PVST+ and MSTP Interoperability

	PVST+	MSTP
PVST+	Yes	Yes (with restrictions)
MSTP	Yes (with restrictions)	Yes

In a mixed MSTP and PVST+ network, the common spanning-tree (CST) root must be inside the MST backbone, and a PVST+ switch cannot connect to multiple MST regions.

STP and IEEE 802.1Q Trunks

The IEEE 802.1Q standard for VLAN trunks imposes some limitations on the spanning-tree strategy for a network. The standard requires only one spanning-tree instance for *all* VLANs allowed on the trunks. However, in a network of Cisco switches connected through IEEE 802.1Q trunks, the switches maintain one spanning-tree instance for *each* VLAN allowed on the trunks.

When you connect a Cisco MWR 2941 to a non-Cisco device through an IEEE 802.1Q trunk, the router uses PVST+ to provide spanning-tree interoperability.

All PVST+ information is maintained by Cisco switches separated by a cloud of non-Cisco 802.1Q switches. The non-Cisco 802.1Q cloud separating the Cisco switches is treated as a single trunk link between the switches.

PVST+ is automatically enabled on IEEE 802.1Q trunks, and no user configuration is required. The external spanning-tree behavior on access ports is not affected by PVST+.

For more information on IEEE 802.1Q trunks, see Chapter 7, “Configuring VLANs.”

Configuring Spanning-Tree Features

- Default Spanning-Tree Configuration, page 9-11
- Spanning-Tree Configuration Guidelines, page 9-11
- Enabling Spanning Tree on an ENI, page 9-13 (required)
- Disabling Spanning Tree, page 9-14 (optional)
- Configuring the Router as a Root Switch, page 9-15 (optional)
- Configuring a Secondary Root Switch, page 9-16 (optional)

- Configuring Port Priority, page 9-17 (optional)
- Configuring Path Cost, page 9-18 (optional)
- Configuring the Switch Priority of a VLAN, page 9-20 (optional)
- Configuring Spanning-Tree Timers, page 9-20 (optional)

Default Spanning-Tree Configuration

Table 9-3 shows the default spanning-tree configuration.

Table 9-3 Default Spanning-Tree Configuration

Feature	Default Setting
Enable state	Enabled on NNIs in VLAN 1. Disabled on ENIs. (Not supported on UNIs) For more information, see the "Supported Spanning-Tree Instances" section on page 9-10.
Spanning-tree mode	PVST+ PVST+ interoperates with PVST. MSTP is disabled.
Switch priority	32768.
Spanning-tree port priority (configurable on a per-interface basis)	128.
Spanning-tree port cost (configurable on a per-interface basis)	1000 Mbps: 4. 100 Mbps: 19. 10 Mbps: 100.
Spanning-tree VLAN port priority (configurable on a per-VLAN basis)	128.
Spanning-tree VLAN port cost (configurable on a per-VLAN basis)	1000 Mbps: 4. 100 Mbps: 19. 10 Mbps: 100.
Spanning-tree timers	Hello time: 2 seconds. Forward-delay time: 15 seconds. Maximum-aging time: 20 seconds.

Spanning-Tree Configuration Guidelines

If more VLANs are defined than there are spanning-tree instances, you can enable PVST+ on STP ports in only 128 VLANs on the switch. The remaining VLANs operate with spanning tree disabled. However, you can map multiple VLANs to the same spanning-tree instances by using MSTP. For more information, see Chapter 10, "Configuring MSTP."

If 255 instances of spanning tree are already in use, you can disable spanning tree on STP ports in one of the VLANs and then enable it on the VLAN where you want it to run. Use the **no spanning-tree vlan** *vlan-id* global configuration command to disable spanning tree on a specific VLAN, and use the **spanning-tree vlan** *vlan-id* global configuration command to enable spanning tree on the desired VLAN.

Note Removal of VLAN 1 should not be performed as this affects the operation of Spanning-tree BPDU processing.

Caution Switches that are not running spanning tree still forward BPDUs that they receive so that the other switches on the VLAN that have a running spanning-tree instance can break loops. Therefore, spanning tree must be running on enough switches to break all the loops in the network; for example, at least one switch on each loop in the VLAN must be running spanning tree. It is not absolutely necessary to run spanning tree on all devices in the VLAN. However, if you are running spanning tree only on a minimal set of switches, an incautious change to the network that introduces another loop into the VLAN can result in a broadcast storm.

If you have already used all available spanning-tree instances on your switch, adding another VLAN creates a VLAN that is not running spanning tree on that switch. If you have the default allowed list on the trunk ports of that switch, the new VLAN is carried on all trunk ports. Depending on the topology of the network, this could create a loop in the new VLAN that will not be broken, particularly if there are several adjacent switches that have all run out of spanning-tree instances. You can prevent this possibility by setting up allowed lists on the trunk ports of switches that have used up their allocation of spanning-tree instances. Setting up allowed lists is not necessary in many cases and can make it more labor-intensive to add another VLAN to the network.

Spanning-tree commands control the configuration of VLAN spanning-tree instances. You create a spanning-tree instance when you assign an STP port (an NNI or ENI with STP enabled) to a VLAN. The spanning-tree instance is removed when the last port is moved to another VLAN. You can configure switch and port parameters before a spanning-tree instance is created; these parameters are applied when the spanning-tree instance is created.

The switch supports PVST+ and MSTP, but only one version can be active at any time. (For example, all VLANs run PVST+, or all VLANs run MSTP.) For information about the different spanning-tree modes and how they interoperate, see the "Spanning-Tree Interoperability and Backward Compatibility" section on page 9-10.

Caution Loop guard works only on point-to-point links. We recommend that each end of the link has a directly connected device that is running STP.

Enabling Spanning Tree on an ENI

By default, spanning tree is enabled on all NNIs on the router and disabled on ENIs. Beginning in privileged EXEC mode, follow these steps to enable spanning tree an ENI:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary.
Step 4	**spanning-tree**	Enable spanning tree on the interface. The interface will belong to the switch spanning tree instance along with NNIs in the VLAN. **Note** This command is visible only on ENIs.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show spanning-tree interface** *interface-id*	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable spanning tree on an ENI, enter the **no spanning-tree** interface command.

Changing the Spanning-Tree Mode

The router supports two spanning-tree modes: PVST+ or MSTP. By default, the router runs the PVST+ protocol on all NNIs and ENIs on which spanning tree is enabled.

Beginning in privileged EXEC mode, follow these steps to change the spanning-tree mode. If you want to enable a mode that is different from the default mode, this procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mode** {**pvst** | **mst** }	Configure a spanning-tree mode on STP ports on the switch. • Select **pvst** to enable PVST+. • Select **mst** to enable MSTP (and RSTP). For more configuration steps, see Chapter 10, “Configuring MSTP.” The default setting is PVST+.
Step 3	**interface** *interface-id*	(Recommended only for rapid-PVST+ mode) Specify an STP port to configure, and enter interface configuration mode. Valid interfaces include physical NNIs or ENIs with spanning tree enabled, VLANs, and NNIs. The VLAN ID range is 1 to 4094. The port-channel range is 1 to 48. **Note** If the interface is a VLAN, only ports with spanning tree enabled in the VLAN will run spanning tree.

	Command	Purpose
Step 4	**spanning-tree link-type point-to-point**	(Recommended only for rapid-PVST+ mode) Specify that the link type for this port is point-to-point. If you connect this port to a remote port through a point-to-point link and the local port becomes a designated port, the router negotiates with the remote port and rapidly changes the local port to the forwarding state.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**clear spanning-tree detected-protocols**	(Recommended only for rapid-PVST+ mode) If any port on the router running spanning tree is connected to a port on a legacy IEEE 802.1D switch, restart the protocol migration process on the entire router.
Step 7	**show spanning-tree summary** and **show spanning-tree interface** *interface-id*	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no spanning-tree mode** global configuration command. To return the port to its default spanning-tree mode setting, use the **no spanning-tree link-type** interface configuration command.

Disabling Spanning Tree

Spanning tree is enabled by default on all NNIs in VLAN 1 and in all newly created VLANs up to the spanning-tree limit specified in the "Supported Spanning-Tree Instances" section on page 9-10. Spanning tree is disabled on ENIs on the router but can be enabled on a per-interface basis. Disable spanning tree only if you are sure there are no loops in the network topology.

⚠
Caution When spanning tree is disabled and loops are present in the topology, excessive traffic and indefinite packet duplication can drastically reduce network performance.

Beginning in privileged EXEC mode, follow these steps to disable spanning-tree on a per-VLAN basis. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no spanning-tree vlan** *vlan-id*	For *vlan-id*, the range is 1 to 4094.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To re-enable spanning-tree, use the **spanning-tree vlan** *vlan-id* global configuration command.

Configuring the Router as a Root Switch

The switch maintains a separate spanning-tree instance for each active VLAN configured on it. A bridge ID, consisting of the switch priority and the switch MAC address, is associated with each instance. For each VLAN, the switch with the lowest bridge ID becomes the root switch for that VLAN.

To configure a switch to become the root for the specified VLAN, use the **spanning-tree vlan** *vlan-id* **root** global configuration command to modify the switch priority from the default value (32768) to a significantly lower value. When you enter this command, the software checks the switch priority of the root switches for each VLAN. Because of the extended system ID support, the switch sets its own priority for the specified VLAN to 24576 if this value will cause this switch to become the root for the specified VLAN.

If any root switch for the specified VLAN has a switch priority lower than 24576, the switch sets its own priority for the specified VLAN to 4096 less than the lowest switch priority. (4096 is the value of the least-significant bit of a 4-bit switch priority value as shown in Table 9-1 on page 9-4.)

Note The **spanning-tree vlan** *vlan-id* **root** global configuration command fails if the value necessary to be the root switch is less than 1.

If your network consists of switches that both do and do not support the extended system ID, it is unlikely that the switch with the extended system ID support will become the root switch. The extended system ID increases the switch priority value every time the VLAN number is greater than the priority of the connected switches running older software.

Note The root switch for each spanning-tree instance should be a backbone or distribution switch. Do not configure an access switch as the spanning-tree primary root.

Use the **diameter** keyword to specify the Layer 2 network diameter (that is, the maximum number of switch hops between any two end stations in the Layer 2 network). When you specify the network diameter, the switch automatically sets an optimal hello time, forward-delay time, and maximum-age time for a network of that diameter, which can significantly reduce the convergence time. You can use the **hello** keyword to override the automatically calculated hello time.

Note After configuring the switch as the root switch, we recommend that you avoid manually configuring the hello time, forward-delay time, and maximum-age time through the **spanning-tree vlan** *vlan-id* **hello-time**, **spanning-tree vlan** *vlan-id* **forward-time**, and the **spanning-tree vlan** *vlan-id* **max-age** global configuration commands.

Beginning in privileged EXEC mode, follow these steps to configure a switch to become the root for the specified VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree vlan** *vlan-id* **root primary** [**diameter** *net-diameter* [**hello-time** *seconds*]]	Configure a switch to become the root for the specified VLAN. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • (Optional) For **diameter** *net-diameter*, specify the maximum number of switches between any two end stations. The range is 2 to 7. • (Optional) For **hello-time** *seconds*, specify the interval in seconds between the generation of configuration messages by the root switch. The range is 1 to 10; the default is 2.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree detail**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no spanning-tree vlan** *vlan-id* **root** global configuration command.

Configuring a Secondary Root Switch

When you configure a device as the secondary root, the switch priority is modified from the default value (32768) to 28672. The switch is then likely to become the root switch for the specified VLAN if the primary root switch fails. This is assuming that the other network switches use the default switch priority of 32768 and therefore are unlikely to become the root switch.

You can execute this command on more than one switch to configure multiple backup root switches. Use the same network diameter and hello-time values that you used when you configured the primary root switch with the **spanning-tree vlan** *vlan-id* **root primary** global configuration command.

Beginning in privileged EXEC mode, follow these steps to configure a switch to become the secondary root for the specified VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree vlan** *vlan-id* **root secondary** [**diameter** *net-diameter* [**hello-time** *seconds*]]	Configure a switch to become the secondary root for the specified VLAN. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • (Optional) For **diameter** *net-diameter*, specify the maximum number of switches between any two end stations. The range is 2 to 7. • (Optional) For **hello-time** *seconds*, specify the interval in seconds between the generation of configuration messages by the root switch. The range is 1 to 10; the default is 2. Use the same network diameter and hello-time values that you used when configuring the primary root switch. See the "Configuring the Router as a Root Switch" section on page 9-15.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree detail**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no spanning-tree vlan** *vlan-id* **root** global configuration command.

Configuring Port Priority

If a loop occurs, spanning tree uses the port priority when selecting a spanning-tree port to put into the forwarding state. You can assign higher priority values (lower numerical values) to ports that you want selected first and lower priority values (higher numerical values) to ones that you want selected last. If all spanning-tree ports have the same priority value, spanning tree puts the port with the lowest interface number in the forwarding state and blocks the other interfaces.

Beginning in privileged EXEC mode, follow these steps to configure the port priority of a spanning-tree port. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode. **Note** If the interface is a VLAN, only ports with spanning tree enabled in the VLAN will run spanning tree.

	Command	Purpose
Step 3	**spanning-tree port-priority** *priority*	Configure the port priority for the spanning-tree port. For *priority*, the range is 0 to 240, in increments of 16; the default is 128. Valid values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected. The lower the number, the higher the priority.
Step 4	**spanning-tree vlan** *vlan-id* **port-priority** *priority*	Configure the port priority for a VLAN. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For *priority*, the range is 0 to 240, in increments of 16; the default is 128. Valid values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected. The lower the number, the higher the priority.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show spanning-tree interface** *interface-id* or **show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note The **show spanning-tree interface** *interface-id* privileged EXEC command displays information only if the port is in a link-up operative state. Otherwise, you can use the **show running-config interface** privileged EXEC command to confirm the configuration.

To return to the default spanning-tree setting, use the **no spanning-tree** [**vlan** *vlan-id*] **port-priority** interface configuration command. For information on how to configure load sharing on trunk ports by using spanning-tree port priorities, see the "Configuring Trunk Ports for Load Sharing" section on page 7-16.

Configuring Path Cost

The spanning-tree path cost default value is derived from the media speed of an interface (port running spanning tree). If a loop occurs, spanning tree uses cost when selecting an interface to put in the forwarding state. You can assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last. If all NNIs have the same cost value, spanning tree puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

Beginning in privileged EXEC mode, follow these steps to configure the cost of an interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode. Valid interfaces include physical NNIs or ENIs with STP enabled and port-channel logical interfaces (**port-channel** *port-channel-number*) that contain only NNIs or ENIs.
Step 3	**spanning-tree cost** *cost*	Configure the cost for an interface. If a loop occurs, spanning tree uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission. For *cost*, the range is 1 to 200000000; the default value is derived from the media speed of the interface.
Step 4	**spanning-tree vlan** *vlan-id* **cost** *cost*	Configure the cost for a VLAN. If a loop occurs, spanning tree uses the path cost when selecting a spanning-tree port to place into the forwarding state. A lower path cost represents higher-speed transmission. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For *cost*, the range is 1 to 200000000; the default value is derived from the media speed of the interface.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show spanning-tree interface** *interface-id* or **show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note The **show spanning-tree interface** *interface-id* privileged EXEC command displays information only for ports that are in a link-up operative state. Otherwise, you can use the **show running-config** privileged EXEC command to confirm the configuration.

To return to the default setting, use the **no spanning-tree** [**vlan** *vlan-id*] **cost** interface configuration command. For information on how to configure load sharing on trunk ports by using spanning-tree path costs, see the "Configuring Trunk Ports for Load Sharing" section on page 7-16.

Configuring the Switch Priority of a VLAN

You can configure the switch priority and make it more likely that the router will be chosen as the root switch.

Note Exercise care when using this command. For most situations, we recommend that you use the **spanning-tree vlan** *vlan-id* **root primary** and the **spanning-tree vlan** *vlan-id* **root secondary** global configuration commands to modify the switch priority.

Beginning in privileged EXEC mode, follow these steps to configure the switch priority of a VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree vlan** *vlan-id* **priority** *priority*	Configure the switch priority of a VLAN. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For *priority*, the range is 0 to 61440 in increments of 4096; the default is 32768. The lower the number, the more likely the switch will be chosen as the root switch. Valid priority values are 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440. All other values are rejected.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no spanning-tree vlan** *vlan-id* **priority** global configuration command.

Configuring Spanning-Tree Timers

Table 9-4 describes the timers that affect the entire spanning-tree performance.

Table 9-4 Spanning-Tree Timers

Variable	Description
Hello timer	Controls how often the router broadcasts hello messages to other switches.
Forward-delay timer	Controls how long each of the listening and learning states last before the STP port begins forwarding.
Maximum-age timer	Controls the amount of time the router stores protocol information received on an STP port.

The following sections describe how to configure spanning-tree timers.

- Configuring the Hello Time
- Configuring the Forwarding-Delay Time for a VLAN
- Configuring the Maximum-Aging Time for a VLAN

Configuring the Hello Time

You can configure the interval between the generation of configuration messages by the root switch by changing the hello time.

Note Exercise care when using this command. For most situations, we recommend that you use the **spanning-tree vlan** *vlan-id* **root primary** and the **spanning-tree vlan** *vlan-id* **root secondary** global configuration commands to modify the hello time.

Beginning in privileged EXEC mode, follow these steps to configure the hello time of a VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree vlan** *vlan-id* **hello-time** *seconds*	Configure the hello time of a VLAN. The hello time is the interval between the generation of configuration messages by the root switch. These messages mean that the switch is alive. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For *seconds*, the range is 1 to 10; the default is 2.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no spanning-tree vlan** *vlan-id* **hello-time** global configuration command.

Configuring the Forwarding-Delay Time for a VLAN

Beginning in privileged EXEC mode, follow these steps to configure the forwarding-delay time for a VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree vlan** *vlan-id* **forward-time** *seconds*	Configure the forward time of a VLAN. The forward delay is the number of seconds a spanning-tree port waits before changing from its spanning-tree learning and listening states to the forwarding state. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For *seconds*, the range is 4 to 30; the default is 15.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no spanning-tree vlan** *vlan-id* **forward-time** global configuration command.

Configuring the Maximum-Aging Time for a VLAN

Beginning in privileged EXEC mode, follow these steps to configure the maximum-aging time for a VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree vlan** *vlan-id* **max-age** *seconds*	Configure the maximum-aging time of a VLAN. The maximum-aging time is the number of seconds a switch waits without receiving spanning-tree configuration messages before attempting a reconfiguration. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For *seconds*, the range is 6 to 40; the default is 20.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no spanning-tree vlan** *vlan-id* **max-age** global configuration command.

Displaying the Spanning-Tree Status

To display the spanning-tree status, use one or more of the privileged EXEC commands in Table 9-5:

Table 9-5 Commands for Displaying Spanning-Tree Status

Command	Purpose
show spanning-tree active	Displays spanning-tree information only on active spanning-tree interfaces.
show spanning-tree detail	Displays a detailed summary of interface information.
show spanning-tree interface *interface-id*	Displays spanning-tree information for the specified spanning-tree interface.
show spanning-tree summary [**totals**]	Displays a summary of interface states or displays the total lines of the STP state section.

You can clear spanning-tree counters by using the **clear spanning-tree** [**interface** *interface-id*] privileged EXEC command.

For information about other keywords for the **show spanning-tree** privileged EXEC command, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.

CHAPTER 10

Configuring MSTP

This chapter describes how to configure the Cisco implementation of the IEEE 802.1s Multiple STP (MSTP) on the Cisco MWR 2941 router. On the Cisco MWR 2941 router, user network interfaces (UNIs) do not participate in STP and immediately forward traffic when they are brought up. STP is enabled by default on network node interfaces (NNIs), and can be also be enabled on enhanced network interfaces (ENIs). If STP is not enabled on an ENI, the interface always forwards traffic.

Note The multiple spanning-tree (MST) implementation is a pre-standard implementation. It is based on the draft version of the IEEE standard.

The MSTP enables multiple VLANs to be mapped to the same spanning-tree instance, thereby reducing the number of spanning-tree instances needed to support a large number of VLANs. The MSTP provides for multiple forwarding paths for data traffic and enables load balancing. It improves the fault tolerance of the network because a failure in one instance (forwarding path) does not affect other instances (forwarding paths). The most common initial deployment of MSTP is in the backbone and distribution layers of a Layer 2 switched network. This deployment provides the highly available network required in a service-provider environment.

When the router is in the MST mode, the Rapid Spanning Tree Protocol (RSTP), which is based on IEEE 802.1w, is automatically enabled. The RSTP provides rapid convergence of the spanning tree through explicit handshaking that eliminates the IEEE 802.1D forwarding delay and quickly transitions root ports and designated ports to the forwarding state.

Both MSTP and RSTP improve the spanning-tree operation and maintain backward compatibility with equipment that is based on the (original) 802.1D spanning tree, with existing Cisco-proprietary Multiple Instance STP (MISTP), and with existing Cisco per-VLAN spanning-tree plus (PVST+). For information about PVST+ and rapid PVST+, see Chapter 9, "Configuring STP." For information about other spanning-tree features such as Port Fast, UplinkFast, root guard, and so forth, see Chapter 11, "Configuring Optional Spanning-Tree Features."

Note The does not currently support rapid per-VLAN spanning-tree plus (rapid PVST+).

Note For complete syntax and usage information for the commands used in this chapter, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.

- Configuring MSTP Features, page 10-14
- Displaying the MST Configuration and Status, page 10-26

Understanding MSTP

MSTP, which uses RSTP for rapid convergence, enables VLANs to be grouped into a spanning-tree instance, with each instance having a spanning-tree topology independent of other spanning-tree instances. This architecture provides multiple forwarding paths for data traffic, enables load balancing, and reduces the number of spanning-tree instances required to support a large number of VLANs.

- Multiple Spanning-Tree Regions, page 10-2
- IST, CIST, and CST, page 10-2
- Hop Count, page 10-5
- Boundary Ports, page 10-6
- IEEE 802.1s Implementation, page 10-6
- “Interoperability with IEEE 802.1D STP” section on page 10-8

For configuration information, see the “Configuring MSTP Features” section on page 10-14.

Multiple Spanning-Tree Regions

For switches to participate in multiple spanning-tree (MST) instances, you must consistently configure the switches with the same MST configuration information. A collection of interconnected switches that have the same MST configuration comprises an MST region as shown in Figure 10-1 on page 10-4.

The MST configuration controls to which MST region each switch belongs. The configuration includes the name of the region, the revision number, and the MST VLAN-to-instance assignment map. You configure the router for a region by using the **spanning-tree mst configuration** global configuration command, after which the router enters the MST configuration mode. From this mode, you can map VLANs to an MST instance by using the **instance** MST configuration command, specify the region name by using the **name** MST configuration command, and set the revision number by using the **revision** MST configuration command.

A region can have one member or multiple members with the same MST configuration; each member must be capable of processing RSTP bridge protocol data units (BPDUs). There is no limit to the number of MST regions in a network, but each region can support up to 65 spanning-tree instances. You can assign a VLAN to only one spanning-tree instance at a time.

IST, CIST, and CST

Unlike PVST+ in which all the spanning-tree instances are independent, the MSTP establishes and maintains two types of spanning trees:

- An internal spanning tree (IST), which is the spanning tree that runs in an MST region.

  Within each MST region, the MSTP maintains multiple spanning-tree instances. Instance 0 is a special instance for a region, known as the internal spanning tree (IST). All other MST instances are numbered from 1 to 4094.

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

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

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

- A common and internal spanning tree (CIST), which is a collection of the ISTs in each MST region, and the common spanning tree (CST) that interconnects the MST regions and single spanning trees.

  The spanning tree computed in a region appears as a subtree in the CST that encompasses the entire switched domain. The CIST is formed as a result of the spanning-tree algorithm running between switches that support the IEEE 802.1w, IEEE 802.1s, and IEEE 802.1D protocols. The CIST inside an MST region is the same as the CST outside a region.

For more information, see the "Operations Within an MST Region" section on page 10-3 and the "Operations Between MST Regions" section on page 10-3.

Note The implementation of the IEEE 802.1s standard changes some of the terminology associated with MST implementations. For a summary of these changes, see Table 10-1 on page 10-5.

Operations Within an MST Region

The IST connects all the MSTP switches in a region. When the IST converges, the root of the IST becomes the IST master (shown in Figure 10-1 on page 10-4), which is the switch within the region with the lowest bridge ID and path cost to the CST root. The IST master also is the CST root if there is only one region within the network. If the CST root is outside the region, one of the MSTP switches at the boundary of the region is selected as the IST master.

When an MSTP switch initializes, it sends BPDUs claiming itself as the root of the CST and the IST master, with both of the path costs to the CST root and to the IST master set to zero. The switch also initializes all of its MST instances and claims to be the root for all of them. If the switch receives superior MST root information (lower bridge ID, lower path cost, and so forth) than currently stored for the port, it relinquishes its claim as the IST master.

During initialization, a region might have many subregions, each with its own IST master. As switches receive superior IST information, they leave their old subregions and join the new subregion that might contain the true IST master. Thus all subregions shrink, except for the one that contains the true IST master.

For correct operation, all switches in the MST region must agree on the same IST master. Therefore, any two switches in the region synchronize their port roles for an MST instance only if they converge to a common IST master.

Operations Between MST Regions

If there are multiple regions or legacy IEEE 802.1D switches within the network, MSTP establishes and maintains the CST, which includes all MST regions and all legacy STP switches in the network. The MST instances combine with the IST at the boundary of the region to become the CST.

The IST connects all the MSTP switches in the region and appears as a subtree in the CST that encompasses the entire switched domain, with the root of the subtree being the IST master. The MST region appears as a virtual switch to adjacent STP switches and MST regions.

Figure 10-1 shows a network with three MST regions and a legacy IEEE 802.1D switch (D). The IST master for region 1 (A) is also the CST root. The IST master for region 2 (B) and the IST master for region 3 (C) are the roots for their respective subtrees within the CST. The RSTP runs in all regions.

Figure 10-1 MST Regions, IST Masters, and the CST Root

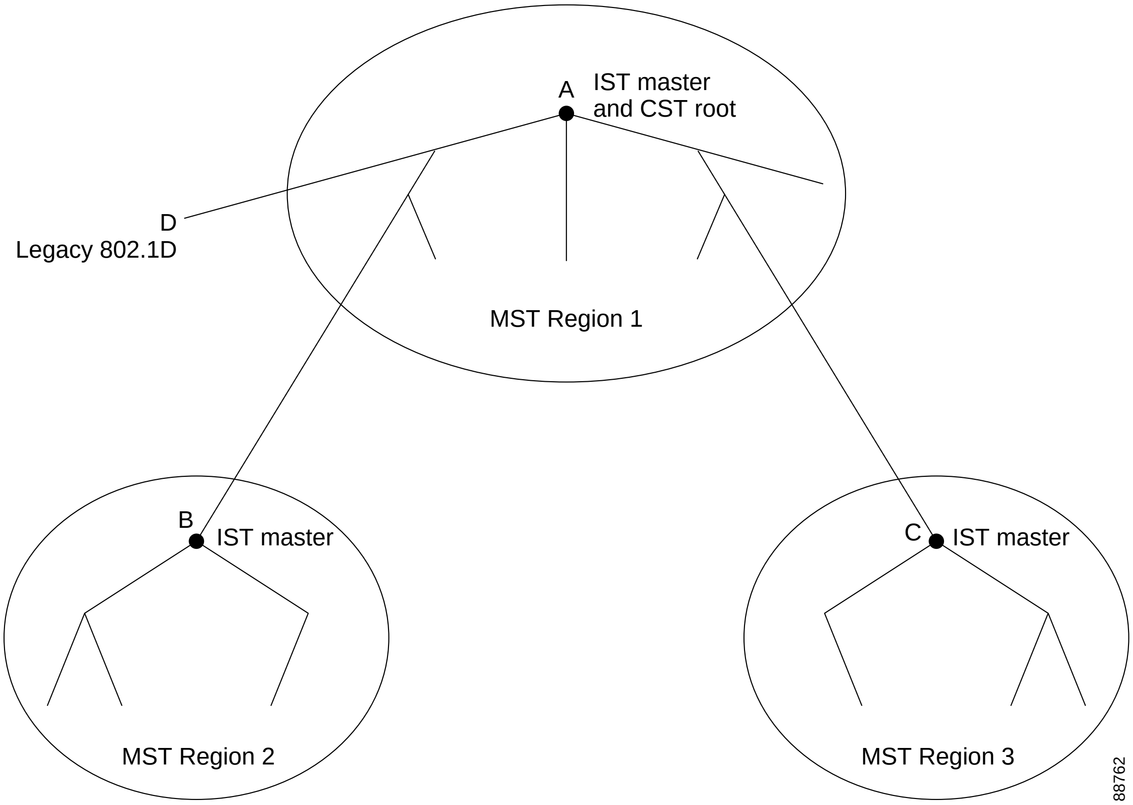


Figure 10-1 does not show additional MST instances for each region. Note that the topology of MST instances can be different from that of the IST for the same region.

Only the CST instance sends and receives BPDUs, and MST instances add their spanning-tree information into the BPDUs to interact with neighboring switches and compute the final spanning-tree topology. Because of this, the spanning-tree parameters related to BPDU transmission (for example, hello time, forward time, max-age, and max-hops) are configured only on the CST instance but affect all MST instances. Parameters related to the spanning-tree topology (for example, switch priority, port VLAN cost, port VLAN priority) can be configured on both the CST instance and the MST instance.

MSTP switches use Version 3 RSTP BPDUs or 802.1D STP BPDUs to communicate with legacy IEEE 802.1D switches. MSTP switches use MSTP BPDUs to communicate with MSTP switches.

IEEE 802.1s Terminology

Some MST naming conventions used in Cisco's prestandard implementation have been changed to identify some *internal* or *regional* parameters. These parameters are significant only within an MST region, as opposed to external parameters that are relevant to the whole network. Because the CIST is the only spanning-tree instance that spans the whole network, only the CIST parameters require the external rather than the internal or regional qualifiers.

- The CIST root is the root switch for the unique instance that spans the whole network, the CIST.
- The CIST external root path cost is the cost to the CIST root. This cost is left unchanged within an MST region. Remember that an MST region looks like a single switch for the CIST. The CIST external root path cost is the root path cost calculated between these virtual switches and switches that do not belong to any region.
- The CIST regional root was called the IST master in the prestandard implementation. If the CIST root is in the region, the CIST regional root is the CIST root. Otherwise, the CIST regional root is the closest switch to the CIST root in the region. The CIST regional root acts as a root switch for the IST.
- The CIST internal root path cost is the cost to the CIST regional root in a region. This cost is only relevant to the IST, instance 0.

Table 10-1 compares the IEEE standard and the Cisco prestandard terminology.

Table 10-1 Prestandard and Standard Terminology

IEEE Standard	Cisco Prestandard	Cisco Standard
CIST regional root	IST master	CIST regional root
CIST internal root path cost	IST master path cost	CIST internal path cost
CIST external root path cost	Root path cost	Root path cost
MSTI regional root	Instance root	Instance root
MSTI internal root path cost	Root path cost	Root path cost

Hop Count

The IST and MST instances do not use the message-age and maximum-age information in the configuration BPDU to compute the spanning-tree topology. Instead, they use the path cost to the root and a hop-count mechanism similar to the IP time-to-live (TTL) mechanism.

By using the **spanning-tree mst max-hops** global configuration command, you can configure the maximum hops inside the region and apply it to the IST and all MST instances in that region. The hop count achieves the same result as the message-age information (trigger a reconfiguration). The root switch of the instance always sends a BPDU (or M-record) with a cost of 0 and the hop count set to the maximum value. When a switch receives this BPDU, it decrements the received remaining hop count by one and propagates this value as the remaining hop count in the BPDUs it generates. When the count reaches zero, the switch discards the BPDU and ages the information held for the port.

The message-age and maximum-age information in the RSTP portion of the BPDU remain the same throughout the region, and the same values are propagated by the region's designated ports at the boundary.

Boundary Ports

In the Cisco prestandard implementation, a boundary port connects an MST region to a single spanning-tree region running RSTP, to a single spanning-tree region running PVST+ or to another MST region with a different MST configuration. A boundary port also connects to a LAN, the designated switch of which is either a single spanning-tree switch or a switch with a different MST configuration.

There is no definition of a boundary port in the IEEE 802.1s standard. The IEEE 802.1Q-2002 standard identifies two kinds of messages that a port can receive: internal (coming from the same region) and external. When a message is external, it is received only by the CIST. If the CIST role is root or alternate, or if the external BPDU is a topology change, it could have an impact on the MST instances. When a message is internal, the CIST part is received by the CIST, and each MST instance receives its respective M-record. The Cisco prestandard implementation treats a port that receives an external message as a boundary port. This means a port cannot receive a mix of internal and external messages.

An MST region includes both switches and LANs. A segment belongs to the region of its designated port. Therefore, a port in a different region than the designated port for a segment is a boundary port. This definition allows two ports internal to a region to share a segment with a port belonging to a different region, creating the possibility of receiving both internal and external messages on a port.

The primary change from the Cisco prestandard implementation is that a designated port is not defined as boundary, unless it is running in an STP-compatible mode.

Note If there is a legacy STP switch on the segment, messages are always considered external.

The other change from the prestandard implementation is that the CIST regional root switch ID field is now inserted where an RSTP or legacy IEEE 802.1Q switch has the sender switch ID. The whole region performs like a single virtual switch by sending a consistent sender switch ID to neighboring switches. In the example in Figure 10-1, switch C would receive a BPDU with the same consistent sender switch ID of root, whether or not A or B is designated for the segment.

IEEE 802.1s Implementation

The Cisco implementation of the IEEE MST standard includes features required to meet the standard, as well as some of the desirable prestandard functionality that is not yet incorporated into the published standard.

Port Role Naming Change

The boundary role is no longer in the final MST standard, but this boundary concept is maintained in Cisco's implementation. However, an MST instance port at a boundary of the region might not follow the state of the corresponding CIST port. Two cases exist now:

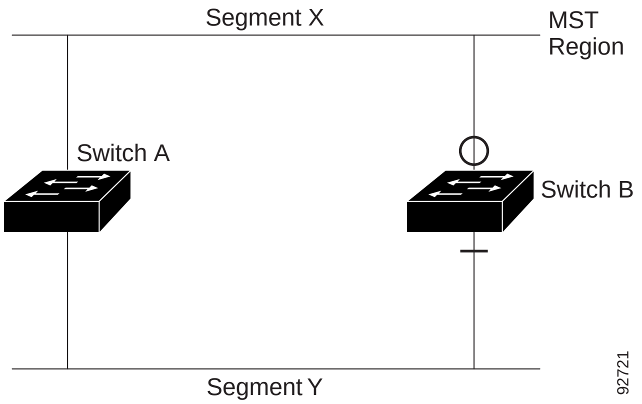
- The boundary port is the root port of the CIST regional root—When the CIST instance port is proposed and is in sync, it can send back an agreement and move to the forwarding state only after all the corresponding MSTI ports are in sync (and thus forwarding). The MSTI ports now have a special *master* role.
- The boundary port is not the root port of the CIST regional root—The MSTI ports follow the state and role of the CIST port. The standard provides less information, and it might be difficult to understand why an MSTI port can be alternately blocking when it receives no BPDUs (MRecords). In this case, although the boundary role no longer exists, the **show** commands identify a port as boundary in the *type* column of the output.

Interoperation Between Legacy and Standard Switches

Because automatic detection of prestandard switches can fail, you can use an interface configuration command to identify prestandard ports. A region cannot be formed between a standard and a prestandard switch, but they can interoperate by using the CIST. Only the capability of load balancing over different instances is lost in that particular case. The CLI displays different flags depending on the port configuration when a port receives prestandard BPDUs. A syslog message also appears the first time a switch receives a prestandard BPDU on a port that has not been configured for prestandard BPDU transmission.

Figure 10-2 illustrates this scenario. Assume that A is a standard switch and B a prestandard switch, both configured to be in the same region. A is the root switch for the CIST, and thus B has a root port (BX) on segment X and an alternate port (BY) on segment Y. If segment Y flaps, and the port on BY becomes the alternate before sending out a single prestandard BPDU, AY cannot detect that a prestandard switch is connected to Y and continues to send standard BPDUs. The port BY is thus fixed in a boundary, and no load balancing is possible between A and B. The same problem exists on segment X, but B might transmit topology changes.

Figure 10-2 Standard and Prestandard Switch Interoperation



Note We recommend that you minimize the interaction between standard and prestandard MST implementations.

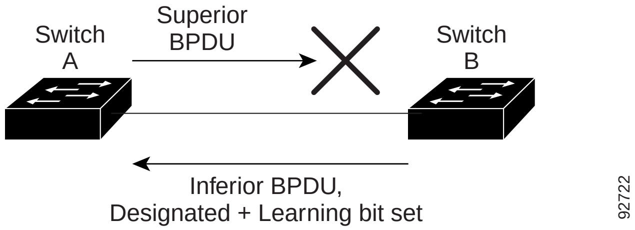
Detecting Unidirectional Link Failure

This feature is not yet present in the IEEE MST standard, but it is included in this Cisco IOS release. The software checks the consistency of the port role and state in the received BPDUs to detect unidirectional link failures that could cause bridging loops.

When a designated port detects a conflict, it keeps its role, but reverts to discarding state because disrupting connectivity in case of inconsistency is preferable to opening a bridging loop.

Figure 10-3 illustrates a unidirectional link failure that typically creates a bridging loop. Switch A is the root switch, and its BPDUs are lost on the link leading to switch B. RSTP and MST BPDUs include the role and state of the sending port. With this information, switch A can detect that switch B does not react to the superior BPDUs it sends and that switch B is the designated, not root switch. As a result, switch A blocks (or keeps blocking) its port, thus preventing the bridging loop.

Figure 10-3 Detecting Unidirectional Link Failure



Interoperability with IEEE 802.1D STP

A switch running MSTP supports a built-in protocol migration mechanism that enables it to interoperate with legacy IEEE 802.1D switches. If this switch receives a legacy IEEE 802.1D configuration BPDU (a BPDU with the protocol version set to 0), it sends only IEEE 802.1D BPDUs on that port. An MSTP switch also can detect that a port is at the boundary of a region when it receives a legacy BPDU, an MSTP BPDU (Version 3) associated with a different region, or an RSTP BPDU (Version 2).

However, the switch does not automatically revert to the MSTP mode if it no longer receives IEEE 802.1D BPDUs because it cannot detect whether the legacy switch has been removed from the link unless the legacy switch is the designated switch. Also, a switch might continue to assign a boundary role to a port when the switch to which this switch is connected has joined the region. To restart the protocol migration process (force the renegotiation with neighboring switches), use the **clear spanning-tree detected-protocols** privileged EXEC command.

If all the legacy switches on the link are RSTP switches, they can process MSTP BPDUs as if they are RSTP BPDUs. Therefore, MSTP switches send either a Version 0 configuration and TCN BPDUs or Version 3 MSTP BPDUs on a boundary port. A boundary port connects to a LAN, the designated switch of which is either a single spanning-tree switch or a switch with a different MST configuration.

Understanding RSTP

The RSTP takes advantage of point-to-point wiring and provides rapid convergence of the spanning tree. Reconfiguration of the spanning tree can occur in less than 1 second (in contrast to 50 seconds with the default settings in the IEEE 802.1D spanning tree), which is critical for networks carrying delay-sensitive traffic such as voice and video.

- Port Roles and the Active Topology, page 10-9
- Rapid Convergence, page 10-10
- Synchronization of Port Roles, page 10-11
- Bridge Protocol Data Unit Format and Processing, page 10-12

For configuration information, see the “Configuring MSTP Features” section on page 10-14.

Port Roles and the Active Topology

The RSTP provides rapid convergence of the spanning tree by assigning port roles and by learning the active topology. The RSTP builds upon the IEEE 802.1D STP to select the switch with the highest switch priority (lowest numerical priority value) as the root switch as described in the "Spanning-Tree Topology and BPDUs" section on page 9-2. Then the RSTP assigns one of these port roles to individual ports.

- Root port—Provides the best path (lowest cost) when the switch forwards packets to the root switch.
- Designated port—Connects to the designated switch, which incurs the lowest path cost when forwarding packets from that LAN to the root switch. The port through which the designated switch is attached to the LAN is called the designated port.
- Alternate port—Offers an alternate path toward the root switch to that provided by the current root port.
- Backup port—Acts as a backup for the path provided by a designated port toward the leaves of the spanning tree. A backup port can exist only when two ports are connected together in a loopback by a point-to-point link or when a switch has two or more connections to a shared LAN segment.
- Disabled port—Has no role within the operation of the spanning tree.

A port with the root or a designated port role is included in the active topology. A port with the alternate or backup port role is excluded from the active topology.

In a stable topology with consistent port roles throughout the network, the RSTP ensures that every root port and designated port immediately transition to the forwarding state while all alternate and backup ports are always in the discarding state (equivalent to blocking in 802.1D). The port state controls the operation of the forwarding and learning processes. Table 10-2 provides a comparison of IEEE 802.1D and RSTP port states.

Table 10-2 Port State Comparison

Operational Status	STP Port State (IEEE 802.1D)	RSTP Port State	Is Port Included in the Active Topology?
Enabled	Blocking	Discarding	No
Enabled	Listening	Discarding	No
Enabled	Learning	Learning	Yes
Enabled	Forwarding	Forwarding	Yes
Disabled	Disabled	Discarding	No

To be consistent with Cisco STP implementations, this guide documents the port state as *blocking* instead of *discarding*. Designated ports start in the listening state.

Rapid Convergence

The RSTP provides for rapid recovery of connectivity following the failure of a switch, a switch port, or a LAN. It provides rapid convergence for edge ports, new root ports, and ports connected through point-to-point links as follows:

- Edge ports—If you configure a port as an edge port on an RSTP switch by using the **spanning-tree portfast** interface configuration command, the edge port immediately transitions to the forwarding state. An edge port is the same as a Port Fast-enabled port, and you should enable it only on ports that connect to a single end station.

- Root ports—If the RSTP selects a new root port, it blocks the old root port and immediately transitions the new root port to the forwarding state.
- Point-to-point links—If you connect a port to another port through a point-to-point link and the local port becomes a designated port, it negotiates a rapid transition with the other port by using the proposal-agreement handshake to ensure a loop-free topology.

  As shown in Figure 10-4, Switch A is connected to Switch B through a point-to-point link, and all of the ports are in the blocking state. Assume that the priority of Switch A is a smaller numerical value than the priority of Switch B. Switch A sends a proposal message (a configuration BPDU with the proposal flag set) to Switch B, proposing itself as the designated switch.

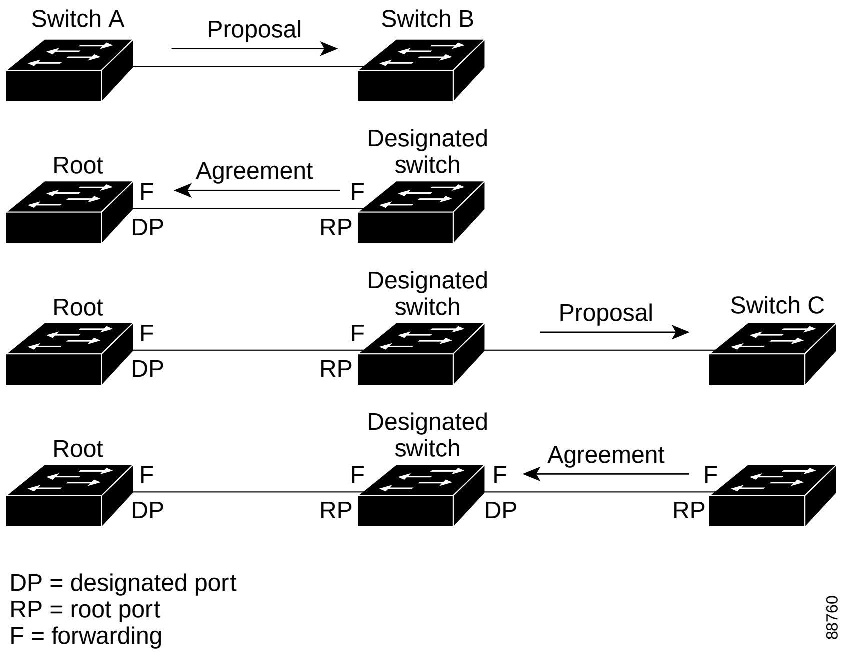
  After receiving the proposal message, Switch B selects as its new root port the port from which the proposal message was received, forces all nonedge ports to the blocking state, and sends an agreement message (a BPDU with the agreement flag set) through its new root port.

  After receiving Switch B's agreement message, Switch A also immediately transitions its designated port to the forwarding state. No loops in the network are formed because Switch B blocked all of its nonedge ports and because there is a point-to-point link between Switches A and B.

  When Switch C is connected to Switch B, a similar set of handshaking messages are exchanged. Switch C selects the port connected to Switch B as its root port, and both ends immediately transition to the forwarding state. With each iteration of this handshaking process, one more switch joins the active topology. As the network converges, this proposal-agreement handshaking progresses from the root toward the leaves of the spanning tree.

  The switch learns the link type from the port duplex mode: a full-duplex port is considered to have a point-to-point connection; a half-duplex port is considered to have a shared connection. You can override the default setting that is controlled by the duplex setting by using the **spanning-tree link-type** interface configuration command.

Figure 10-4 Proposal and Agreement Handshaking for Rapid Convergence



Synchronization of Port Roles

When the switch receives a proposal message on one of its ports and that port is selected as the new root port, the RSTP forces all other ports to synchronize with the new root information.

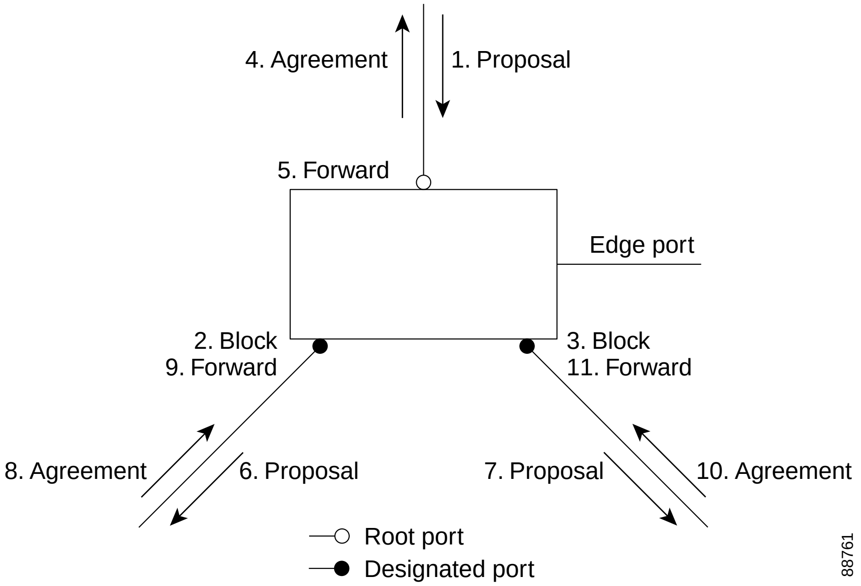
The switch is synchronized with superior root information received on the root port if all other ports are synchronized. An individual port on the switch is synchronized if

- That port is in the blocking state.
- It is an edge port (a port configured to be at the edge of the network).

If a designated STP port is in the forwarding state and is not configured as an edge port, it transitions to the blocking state when the RSTP forces it to synchronize with new root information. In general, when the RSTP forces a port to synchronize with root information and the port does not satisfy any of the above conditions, its port state is set to blocking.

After ensuring all of the ports are synchronized, the switch sends an agreement message to the designated switch corresponding to its root port. When the switches connected by a point-to-point link are in agreement about their port roles, the RSTP immediately transitions the port states to forwarding. The sequence of events is shown in Figure 10-5.

Figure 10-5 Sequence of Events During Rapid Convergence



Bridge Protocol Data Unit Format and Processing

The RSTP BPDU format is the same as the IEEE 802.1D BPDU format except that the protocol version is set to 2. A new one-byte Version 1 Length field is set to zero, which means that no version 1 protocol information is present. Table 10-3 shows the RSTP flag fields.

Table 10-3 RSTP BPDU Flags

Bit	Function
0	Topology change (TC)
1	Proposal
2–3:	Port role:
00	Unknown
01	Alternate port
10	Root port
11	Designated port
4	Learning
5	Forwarding
6	Agreement
7	Topology change acknowledgement (TCA)

The sending switch sets the proposal flag in the RSTP BPDU to propose itself as the designated switch on that LAN. The port role in the proposal message is always set to the designated port.

The sending switch sets the agreement flag in the RSTP BPDU to accept the previous proposal. The port role in the agreement message is always set to the root port.

The RSTP does not have a separate topology change notification (TCN) BPDU. It uses the topology change (TC) flag to show the topology changes. However, for interoperability with 802.1D switches, the RSTP switch processes and generates TCN BPDUs.

The learning and forwarding flags are set according to the state of the sending port.

Processing Superior BPDU Information

If a port receives superior root information (lower bridge ID, lower path cost, and so forth) than currently stored for the port, the RSTP triggers a reconfiguration. If the port is proposed and is selected as the new root port, RSTP forces all the other ports to synchronize.

If the BPDU received is an RSTP BPDU with the proposal flag set, the switch sends an agreement message after all of the other ports are synchronized. If the BPDU is an 802.1D BPDU, the switch does not set the proposal flag and starts the forward-delay timer for the port. The new root port requires twice the forward-delay time to transition to the forwarding state.

If the superior information received on the port causes the port to become a backup or alternate port, RSTP sets the port to the blocking state but does not send the agreement message. The designated port continues sending BPDUs with the proposal flag set until the forward-delay timer expires, at which time the port transitions to the forwarding state.

Processing Inferior BPDU Information

If a designated port receives an inferior BPDU (higher bridge ID, higher path cost, and so forth than currently stored for the port) with a designated port role, it immediately replies with its own information.

Topology Changes

This section describes the differences between the RSTP and the IEEE 802.1D in handling spanning-tree topology changes.

- Detection—Unlike IEEE 802.1D in which *any* transition between the blocking and the forwarding state causes a topology change, *only* transitions from the blocking to the forwarding state cause a topology change with RSTP (only an increase in connectivity is considered a topology change). State changes on an edge port do not cause a topology change. When an RSTP switch detects a topology change, it flushes the learned information on all of its nonedge ports except on those from which it received the TC notification.
- Notification—Unlike IEEE 802.1D, which uses TCN BPDUs, the RSTP does not use them. However, for 802.1D interoperability, an RSTP switch processes and generates TCN BPDUs.
- Acknowledgement—When an RSTP switch receives a TCN message on a designated port from an IEEE 802.1D switch, it replies with an 802.1D configuration BPDU with the TCA bit set. However, if the TC-while timer (the same as the topology-change timer in 802.1D) is active on a root port connected to an 802.1D switch and a configuration BPDU with the TCA bit set is received, the TC-while timer is reset.

  This behavior is only required to support IEEE 802.1D switches. The RSTP BPDUs never have the TCA bit set.
- Propagation—When an RSTP switch receives a TC message from another switch through a designated or root port, it propagates the change to all of its nonedge, designated ports and to the root port (excluding the port on which it is received). The switch starts the TC-while timer for all such ports and flushes the information learned on them.
- Protocol migration—For backward compatibility with IEEE 802.1D switches, RSTP selectively sends IEEE 802.1D configuration BPDUs and TCN BPDUs on a per-port basis.

  When a port is initialized, the migrate-delay timer is started (specifies the minimum time during which RSTP BPDUs are sent), and RSTP BPDUs are sent. While this timer is active, the switch processes all BPDUs received on that port and ignores the protocol type.

  If the switch receives an IEEE 802.1D BPDU after the port's migration-delay timer has expired, it assumes that it is connected to an IEEE 802.1D switch and starts using only IEEE 802.1D BPDUs. However, if the RSTP switch is using IEEE 802.1D BPDUs on a port and receives an RSTP BPDU after the timer has expired, it restarts the timer and starts using RSTP BPDUs on that port.

Configuring MSTP Features

- Default MSTP Configuration, page 10-14
- MSTP Configuration Guidelines, page 10-15
- Specifying the MST Region Configuration and Enabling MSTP, page 10-16 (required)
- Configuring the Router as a Root Switch, page 10-17 (optional)
- Configuring a Secondary Root Switch, page 10-18 (optional)
- Configuring Port Priority, page 10-19 (optional)
- Configuring Path Cost, page 10-20 (optional)
- Configuring the Device Priority, page 10-21 (optional)
- Configuring the Hello Time, page 10-22 (optional)

- Configuring the Forwarding-Delay Time, page 10-23 (optional)
- Configuring the Maximum-Aging Time, page 10-23 (optional)
- Configuring the Maximum-Hop Count, page 10-24 (optional)
- Specifying the Link Type to Ensure Rapid Transitions, page 10-24 (optional)
- Designating the Neighbor Type, page 10-25 (optional)
- Restarting the Protocol Migration Process, page 10-25 (optional)

Default MSTP Configuration

Table 10-4 shows the default MSTP configuration.

Table 10-4 Default MSTP Configuration

Feature	Default Setting
Spanning-tree mode	PVST+
Switch priority (configurable on a per-CIST port basis)	32768.
Spanning-tree port priority (configurable on a per-CIST port basis)	128.
Spanning-tree port cost (configurable on a per-CIST port basis)	1000 Mbps: 4. 100 Mbps: 19. 10 Mbps: 100.
Hello time	2 seconds.
Forward-delay time	15 seconds.
Maximum-aging time	20 seconds.
Maximum hop count	20 hops.

For information about the supported number of spanning-tree instances, see the "Supported Spanning-Tree Instances" section on page 9-10.

MSTP Configuration Guidelines

- On the Cisco MWR 2941 router, MSTP is supported only on NNIs or ENIs on which STP has been enabled. You enable STP on an ENI by entering the **spanning-tree** interface configuration command. UNIs do not participate in MSTP.
- When you enable MST by using the **spanning-tree mode mst** global configuration command, RSTP is automatically enabled.
- For two or more switches to be in the same MST region, they must have the same VLAN-to-instance map, the same configuration revision number, and the same name.
- The router supports up to 65 MST instances. You can map up to 255 VLANs to a particular MST instance.

- PVST+ and MSTP are supported, but only one version can be active at any time. (For example, all VLANs run PVST+ or all VLANs run MSTP.) For more information, see the "Spanning-Tree Interoperability and Backward Compatibility" section on page 9-10. For information on the recommended trunk port configuration, see the "Interaction with Other Features" section on page 7-13.
- You can manually configure the MST configuration (region name, revision number, and VLAN-to-instance mapping) on each switch within the MST region by using the command-line interface (CLI).
- For load balancing across redundant paths in the network to work, all VLAN-to-instance mapping assignments must match; otherwise, all traffic flows on a single link.
- All MST boundary ports must be forwarding for load balancing between a PVST+ and an MST cloud. For this to occur, the IST master of the MST cloud should also be the root of the CST. If the MST cloud consists of multiple MST regions, one of the MST regions must contain the CST root, and all of the other MST regions must have a better path to the root contained within the MST cloud than a path through the PVST+ cloud. You might have to manually configure the switches in the clouds.
- Partitioning the network into a large number of regions is not recommended. However, if this situation is unavoidable, we recommend that you partition the switched LAN into smaller LANs interconnected by routers or non-Layer 2 devices.

Specifying the MST Region Configuration and Enabling MSTP

For two or more switches to be in the same MST region, they must have the same VLAN-to-instance mapping, the same configuration revision number, and the same name.

A region can have one member or multiple members with the same MST configuration; each member must be capable of processing RSTP BPDUs. There is no limit to the number of MST regions in a network, but each region can support up to 65 spanning-tree instances. You can assign a VLAN to only one spanning-tree instance at a time.

Beginning in privileged EXEC mode, follow these steps to specify the MST region configuration and enable MSTP. This procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst configuration**	Enter MST configuration mode.
Step 3	**instance** *instance-id* **vlan** *vlan-range*	Map VLANs to an MST instance. • For *instance-id*, the range is 0 to 4094. • For **vlan** *vlan-range*, the range is 1 to 4094. When you map VLANs to an MST instance, the mapping is incremental, and the VLANs specified in the command are added to or removed from the VLANs that were previously mapped. To specify a VLAN range, use a hyphen; for example, **instance 1 vlan 1-63** maps VLANs 1 through 63 to MST instance 1. To specify a VLAN series, use a comma; for example, **instance 1 vlan 10, 20, 30** maps VLANs 10, 20, and 30 to MST instance 1.

	Command	Purpose
Step 4	**name** *name*	Specify the configuration name. The *name* string has a maximum length of 32 characters and is case sensitive.
Step 5	**revision** *version*	Specify the configuration revision number. The range is 0 to 65535.
Step 6	**show pending**	Verify your configuration by displaying the pending configuration.
Step 7	**exit**	Apply all changes, and return to global configuration mode.
Step 8	**spanning-tree mode mst**	Enable MSTP. RSTP is also enabled. **Caution** Changing spanning-tree modes can disrupt traffic because all spanning-tree instances are stopped for the previous mode and restarted in the new mode. You cannot run both MSTP and PVST+ at the same time.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show running-config**	Verify your entries.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default MST region configuration, use the **no spanning-tree mst configuration** global configuration command. To return to the default VLAN-to-instance map, use the **no instance** *instance-id* [**vlan** *vlan-range*] MST configuration command. To return to the default name, use the **no name** MST configuration command. To return to the default revision number, use the **no revision** MST configuration command.

This example shows how to enter MST configuration mode, map VLANs 10 to 20 to MST instance 1, name the region *region1*, set the configuration revision to 1, display the pending configuration, apply the changes, and return to global configuration mode:

```
Router(config)# spanning-tree mst configuration
Router(config-mst)# instance 1 vlan 10-20
Router(config-mst)# name region1
Router(config-mst)# revision 1
Router(config-mst)# show pending
Pending MST configuration
Name      [region1]
Revision  1
Instances configured  2
Instance  Vlans Mapped
--------  ---------------------
0         1-9,21-4094
1         10-20
-------------------------------

Router(config-mst)# exit
Router(config)#
```

Configuring the Router as a Root Switch

The router maintains a spanning-tree instance for the group of VLANs mapped to it. A bridge ID, consisting of the switch priority and the switch MAC address, is associated with each instance. For a group of VLANs, the switch with the lowest bridge ID becomes the root switch.

To configure the router to become the root switch, use the **spanning-tree mst** *instance-id* **root** global configuration command to modify the switch priority from the default value (32768) to a significantly lower value so that the switch becomes the root switch for the specified spanning-tree instance. When you enter this command, the switch checks the switch priorities of the root switches. Because of the extended system ID support, the switch sets its own priority for the specified instance to 24576 if this value will cause this switch to become the root for the specified spanning-tree instance.

If any root switch for the specified instance has a switch priority lower than 24576, the switch sets its own priority to 4096 less than the lowest switch priority. (4096 is the value of the least-significant bit of a 4-bit switch priority value as shown in Table 9-1 on page 9-4.)

If your network consists of switches that both do and do not support the extended system ID, it is unlikely that the switch with the extended system ID support will become the root switch. The extended system ID increases the switch priority value every time the VLAN number is greater than the priority of the connected switches running older software.

The root switch for each spanning-tree instance should be a backbone or distribution switch. Do not configure an access switch as the spanning-tree primary root.

Use the **diameter** keyword, which is available only for MST instance 0, to specify the Layer 2 network diameter (that is, the maximum number of switch hops between any two end stations in the Layer 2 network). When you specify the network diameter, the switch automatically sets an optimal hello time, forward-delay time, and maximum-age time for a network of that diameter, which can significantly reduce the convergence time. You can use the **hello** keyword to override the automatically calculated hello time.

Note After configuring the router as the root switch, we recommend that you avoid manually configuring the hello time, forward-delay time, and maximum-age time through the **spanning-tree mst hello-time**, **spanning-tree mst forward-time**, and the **spanning-tree mst max-age** global configuration commands.

Beginning in privileged EXEC mode, follow these steps to configure a switch as the root switch. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst** *instance-id* **root primary** [**diameter** *net-diameter* [**hello-time** *seconds*]]	Configure the router as the root switch. • For *instance-id*, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4094. • (Optional) For **diameter** *net-diameter*, specify the maximum number of switches between any two end stations. The range is 2 to 7. This keyword is available only for MST instance 0. • (Optional) For **hello-time** *seconds*, specify the interval in seconds between the generation of configuration messages by the root switch. The range is 1 to 10 seconds; the default is 2 seconds.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst** *instance-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the router to its default setting, use the **no spanning-tree mst** *instance-id* **root** global configuration command.

Configuring a Secondary Root Switch

When you configure the router with the extended system ID support as the secondary root, the switch priority is modified from the default value (32768) to 28672. The switch is then likely to become the root switch for the specified instance if the primary root switch fails. This is assuming that the other network switches use the default switch priority of 32768 and therefore are unlikely to become the root switch.

You can execute this command on more than one switch to configure multiple backup root switches. Use the same network diameter and hello-time values that you used when you configured the primary root switch with the **spanning-tree mst** *instance-id* **root primary** global configuration command.

Beginning in privileged EXEC mode, follow these steps to configure a switch as the secondary root switch. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst** *instance-id* **root secondary** [**diameter** *net-diameter* [**hello-time** *seconds*]]	Configure a switch as the secondary root switch. • For *instance-id*, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4094. • (Optional) For **diameter** *net-diameter*, specify the maximum number of switches between any two end stations. The range is 2 to 7. This keyword is available only for MST instance 0. • (Optional) For **hello-time** *seconds*, specify the interval in seconds between the generation of configuration messages by the root switch. The range is 1 to 10 seconds; the default is 2 seconds. Use the same network diameter and hello-time values that you used when configuring the primary root switch. See the "Configuring the Router as a Root Switch" section on page 10-17.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst** *instance-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the router to its default setting, use the **no spanning-tree mst** *instance-id* **root** global configuration command.

Configuring Port Priority

If a loop occurs, the MSTP uses the port priority when selecting an STP port to put into the forwarding state. You can assign higher priority values (lower numerical values) to STP ports that you want selected first and lower priority values (higher numerical values) that you want selected last. If all interfaces have the same priority value, the MSTP puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

Beginning in privileged EXEC mode, follow these steps to configure the MSTP port priority of an interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode. Valid interfaces include physical NNIs or ENIs with spanning tree enabled, VLANs, and NNIs. The VLAN ID range is 1 to 4094. The port-channel range is 1 to 48. **Note** If the interface is a VLAN, only ports with spanning tree enabled in the VLAN will run spanning tree.
Step 3	**spanning-tree mst** *instance-id* **port-priority** *priority*	Configure the port priority. • For *instance-id*, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4094. • For *priority*, the range is 0 to 240 in increments of 16. The default is 128. The lower the number, the higher the priority. The priority values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show spanning-tree mst interface** *interface-id* or **show spanning-tree mst** *instance-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note The **show spanning-tree mst interface** *interface-id* privileged EXEC command displays information only if the port is in a link-up operative state. Otherwise, you can use the **show running-config interface** privileged EXEC command to confirm the configuration.

To return the interface to its default setting, use the **no spanning-tree mst** *instance-id* **port-priority** interface configuration command.

Configuring Path Cost

The MSTP path cost default value is derived from the media speed of an STP port. If a loop occurs, the MSTP uses cost when selecting an interface to put in the forwarding state. You can assign lower cost values to STP ports that you want selected first and higher cost values that you want selected last. If all interfaces have the same cost value, the MSTP puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

Beginning in privileged EXEC mode, follow these steps to configure the MSTP cost of an interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode. Valid interfaces include physical NNIs or ENIs with spanning tree enabled, VLANs, and NNIs. The VLAN ID range is 1 to 4094. The port-channel range is 1 to 48. **Note** If the interface is a VLAN, only ports with spanning tree enabled in the VLAN will run spanning tree.
Step 3	**spanning-tree mst** *instance-id* **cost** *cost*	Configure the cost. If a loop occurs, the MSTP uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission. • For *instance-id*, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4094. • For *cost*, the range is 1 to 200000000; the default value is derived from the media speed of the interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show spanning-tree mst interface** *interface-id* or **show spanning-tree mst** *instance-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note The **show spanning-tree mst interface** *interface-id* privileged EXEC command displays information only for ports that are in a link-up operative state. Otherwise, you can use the **show running-config** privileged EXEC command to confirm the configuration.

To return the interface to its default setting, use the **no spanning-tree mst** *instance-id* **cost** interface configuration command.

Configuring the Device Priority

You can configure the device priority and make it more likely that a switch will be chosen as the root switch.

Note Exercise care when using this command. For most situations, we recommend that you use the **spanning-tree mst** *instance-id* **root primary** and the **spanning-tree mst** *instance-id* **root secondary** global configuration commands to modify the switch priority.

Beginning in privileged EXEC mode, follow these steps to configure the switch priority. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst** *instance-id* **priority** *priority*	Configure the switch priority. • For *instance-id*, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4094. • For *priority*, the range is 0 to 61440 in increments of 4096; the default is 32768. The lower the number, the more likely the switch will be chosen as the root switch. Priority values are 0, 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440. All other values are rejected.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst** *instance-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the router to its default setting, use the **no spanning-tree mst** *instance-id* **priority** global configuration command.

Configuring the Hello Time

You can configure the interval between the generation of configuration messages by the root switch by changing the hello time.

Note Exercise care when using this command. For most situations, we recommend that you use the **spanning-tree mst** *instance-id* **root primary** and the **spanning-tree mst** *instance-id* **root secondary** global configuration commands to modify the hello time.

Beginning in privileged EXEC mode, follow these steps to configure the hello time for all MST instances. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst hello-time** *seconds*	Configure the hello time for all MST instances. The hello time is the interval between the generation of configuration messages by the root switch. These messages mean that the switch is alive. For *seconds*, the range is 1 to 10; the default is 2.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the router to its default setting, use the **no spanning-tree mst hello-time** global configuration command.

Configuring the Forwarding-Delay Time

Beginning in privileged EXEC mode, follow these steps to configure the forwarding-delay time for all MST instances. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst forward-time** *seconds*	Configure the forward time for all MST instances. The forward delay is the number of seconds a port waits before changing from its spanning-tree learning and listening states to the forwarding state. For *seconds*, the range is 4 to 30; the default is 15.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the router to its default setting, use the **no spanning-tree mst forward-time** global configuration command.

Configuring the Maximum-Aging Time

Beginning in privileged EXEC mode, follow these steps to configure the maximum-aging time for all MST instances. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst max-age** *seconds*	Configure the maximum-aging time for all MST instances. The maximum-aging time is the number of seconds a switch waits without receiving spanning-tree configuration messages before attempting a reconfiguration. For *seconds*, the range is 6 to 40; the default is 20.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the router to its default setting, use the **no spanning-tree mst max-age** global configuration command.

Configuring the Maximum-Hop Count

Beginning in privileged EXEC mode, follow these steps to configure the maximum-hop count for all MST instances. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst max-hops** *hop-count*	Specify the number of hops in a region before the BPDU is discarded, and the information held for a port is aged. For *hop-count*, the range is 1 to 255; the default is 20.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the router to its default setting, use the **no spanning-tree mst max-hops** global configuration command.

Specifying the Link Type to Ensure Rapid Transitions

If you connect an STP port to another STP port through a point-to-point link and the local port becomes a designated port, the RSTP negotiates a rapid transition with the other port by using the proposal-agreement handshake to ensure a loop-free topology as described in the "Rapid Convergence" section on page 10-10.

By default, the link type is controlled from the duplex mode of the interface: a full-duplex port is considered to have a point-to-point connection; a half-duplex port is considered to have a shared connection. If you have a half-duplex link physically connected point-to-point to a single port on a remote switch running MSTP, you can override the default setting of the link type and enable rapid transitions to the forwarding state.

Beginning in privileged EXEC mode, follow these steps to override the default link-type setting. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode. Valid interfaces include physical NNIs or ENIs with spanning tree enabled, VLANs, and NNIs. The VLAN ID range is 1 to 4094. The port-channel range is 1 to 48. **Note** If the interface is a VLAN, only ports with spanning tree enabled in the VLAN will run spanning tree.
Step 3	**spanning-tree link-type point-to-point**	Specify that the link type of a port is point-to-point.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show spanning-tree mst interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the port to its default setting, use the **no spanning-tree link-type** interface configuration command.

Designating the Neighbor Type

A topology could contain both prestandard and IEEE 802.1s standard compliant devices. By default, ports can automatically detect prestandard devices, but they can still receive both standard and prestandard BPDUs. When there is a mismatch between a device and its neighbor, only the CIST runs on the interface.

You can choose to set a port to send only prestandard BPDUs. The prestandard flag appears in all the show commands, even if the port is in STP compatibility mode.

Beginning in privileged EXEC mode, follow these steps to override the default link-type setting. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode. Valid interfaces include physical NNIs or ENIs with spanning tree enabled, VLANs, and NNIs. The VLAN ID range is 1 to 4094. The port-channel range is 1 to 48. **Note** If the interface is a VLAN, only ports with spanning tree enabled in the VLAN will run spanning tree.
Step 3	**spanning-tree mst pre-standard**	Specify that the port can send only prestandard BPDUs.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show spanning-tree mst interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the port to its default setting, use the **no spanning-tree mst prestandard** interface configuration command.

Restarting the Protocol Migration Process

A device running MSTP supports a built-in protocol migration mechanism that enables it to interoperate with legacy 802.1D switches. If this switch receives a legacy 802.1D configuration BPDU (a BPDU with the protocol version set to 0), it sends only 802.1D BPDUs on that port. An MSTP switch also can detect that a port is at the boundary of a region when it receives a legacy BPDU, an MST BPDU (Version 3) associated with a different region, or an RST BPDU (Version 2).

However, the switch does not automatically revert to the MSTP mode if it no longer receives 802.1D BPDUs because it cannot detect whether the legacy switch has been removed from the link unless the legacy switch is the designated switch. A switch also might continue to assign a boundary role to a port when the switch to which it is connected has joined the region.

To restart the protocol migration process (force the renegotiation with neighboring switches) on the switch, use the **clear spanning-tree detected-protocols** privileged EXEC command.

To restart the protocol migration process on a specific interface, use the **clear spanning-tree detected-protocols interface** *interface-id* privileged EXEC command.

Displaying the MST Configuration and Status

To display the spanning-tree status, use one or more of the privileged EXEC commands in Table 10-5:

Table 10-5 Commands for Displaying MST Status

Command	Purpose
show spanning-tree mst configuration	Displays the MST region configuration.
show spanning-tree mst configuration digest	Displays the MD5 digest included in the current MSTCI.
show spanning-tree mst *instance-id*	Displays MST information for the specified instance.
show spanning-tree mst interface *interface-id*	Displays MST information for the specified interface.

For information about other keywords for the **show spanning-tree** privileged EXEC command, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.

CHAPTER 11

Configuring Optional Spanning-Tree Features

This chapter describes how to configure optional spanning-tree features on the Cisco MWR 2941router. You can configure all of these features when your router is running per-VLAN spanning-tree plus (PVST+). You can configure only the noted features when your router is running the Multiple Spanning Tree Protocol (MSTP) protocol. On the Cisco MWR 2941 router, STP is enabled by default on network node interfaces (NNIs). It is disabled by default, but can be enabled, on enhanced network interfaces (ENIs). User network interfaces (UNIs) on the switch do not participate in STP. UNIs and ENIs on which STP is not enabled immediately forward traffic when they are brought up.

For information on configuring the PVST+, see Chapter 9, "Configuring STP." For information about the Multiple Spanning Tree Protocol (MSTP) and how to map multiple VLANs to the same spanning-tree instance, see Chapter 10, "Configuring MSTP."

For detailed information about the commands used in this chapter, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.

- Understanding Optional Spanning-Tree Features, page 11-1
- Configuring Optional Spanning-Tree Features, page 11-5
- Displaying the Spanning-Tree Status, page 11-10

Understanding Optional Spanning-Tree Features

- Understanding Port Fast, page 11-1
- Understanding BPDU Guard, page 11-2
- Understanding BPDU Filtering, page 11-3
- Understanding Root Guard, page 11-3
- Understanding Loop Guard, page 11-4

Understanding Port Fast

Port Fast immediately brings an STP port configured as an access or trunk port to the forwarding state from a blocking state, bypassing the listening and learning states.

Note By default, STP is enabled on NNIs and disabled on ENIs. To configure an ENI as an STP port, enter the **spanning-tree** interface configuration command to configure the port as an STP port.

You can use Port Fast on STP ports connected to a single workstation or server, as shown in Figure 11-1, to allow those devices to immediately connect to the network, rather than waiting for the spanning tree to converge.

STP ports connected to a single workstation or server should not receive bridge protocol data units (BPDUs). An STP port with Port Fast enabled goes through the normal cycle of spanning-tree status changes when the switch is restarted.

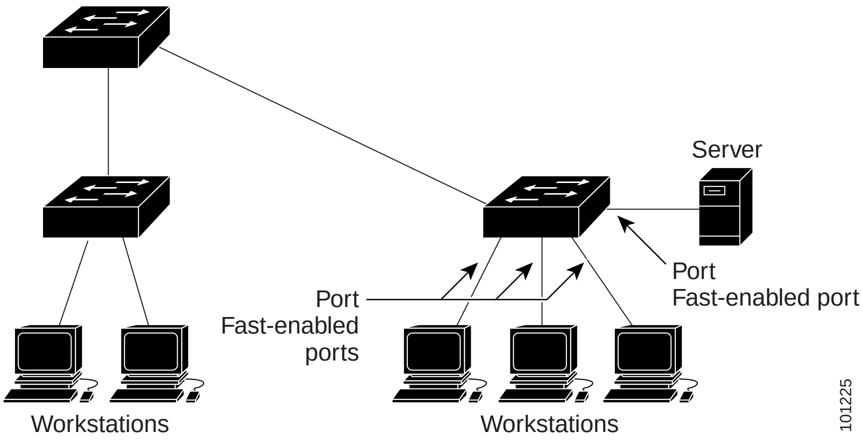
Note Because the purpose of Port Fast is to minimize the time interfaces must wait for spanning tree to converge, it is effective only when used on STP ports connected to end stations. If you enable Port Fast on an interface connecting to another switch, you risk creating a spanning-tree loop.

UNIs are typically customer-facing ports and do not participate in the spanning tree of the service provider. However, if you configure a customer-facing port as an ENI and enable spanning tree, the ENI could become the spanning tree root port unless you configure root guard on the port. See the "Understanding Root Guard" section on page 11-3. A customer-facing ENI with STP enabled participates in the same spanning tree as the service-provider facing NNI.

Note Exercise caution when enabling STP on a customer-facing ENI.

You can enable this feature by using the **spanning-tree portfast** interface configuration or the **spanning-tree portfast default** global configuration command.

Figure 11-1 Port Fast-Enabled Interfaces



Understanding BPDU Guard

The BPDU guard feature can be globally enabled on the router or can be enabled per interface, but the feature operates with some differences.

At the global level, you enable BPDU guard on Port Fast-enabled STP ports by using the **spanning-tree portfast bpduguard default** global configuration command. Spanning tree shuts down STP ports that are in a Port Fast-operational state if any BPDU is received on those ports. In a valid configuration, Port Fast-enabled STP ports do not receive BPDUs. Receiving a BPDU on a Port Fast-enabled port signals an invalid configuration, such as the connection of an unauthorized device, and the BPDU guard feature puts the interface in the error-disabled state.

At the interface level, you enable BPDU guard on any STP port by using the **spanning-tree bpduguard enable** interface configuration command without also enabling the Port Fast feature. When the STP port receives a BPDU, it is put in the error-disabled state.

The BPDU guard feature provides a secure response to invalid configurations because you must manually put the interface back in service. Use the BPDU guard feature in a service-provider network to prevent an access port from participating in the spanning tree.

You can enable the BPDU guard feature for the entire router or for an interface.

Understanding BPDU Filtering

The BPDU filtering feature can be globally enabled on the router or can be enabled per interface, but the feature operates with some differences.

At the global level, you can enable BPDU filtering on Port Fast-enabled STP ports by using the **spanning-tree portfast bpdufilter default** global configuration command. This command prevents interfaces that are in a Port Fast-operational state from sending or receiving BPDUs. The interfaces still send a few BPDUs at link-up before the router begins to filter outbound BPDUs. You should globally enable BPDU filtering on a switch so that hosts connected to these ports do not receive BPDUs. If a BPDU is received on a Port Fast-enabled STP port, the interface loses its Port Fast-operational status, and BPDU filtering is disabled.

At the interface level, you can enable BPDU filtering on any STP port by using the **spanning-tree bpdufilter enable** interface configuration command without also enabling the Port Fast feature. This command prevents the interface from sending or receiving BPDUs.

Caution Enabling BPDU filtering on an STP port is the same as disabling spanning tree on it and can result in spanning-tree loops.

You can enable the BPDU filtering feature for the entire router or for an STP port.

Understanding Root Guard

The Layer 2 network of a service provider (SP) can include many connections to switches that are not owned by the SP. In such a topology, the spanning tree can reconfigure itself and select a *customer switch* as the root switch, as shown in Figure 11-2. You can avoid this situation by enabling root guard on SP switch interfaces that connect to switches in your customer's network. If spanning-tree calculations cause an interface in the customer network to be selected as the root port, root guard then places the interface in the root-inconsistent (blocked) state to prevent the customer's switch from becoming the root switch or being in the path to the root.

If a switch outside the SP network becomes the root switch, the interface is blocked (root-inconsistent state), and spanning tree selects a new root switch. The customer's switch does not become the root switch and is not in the path to the root.

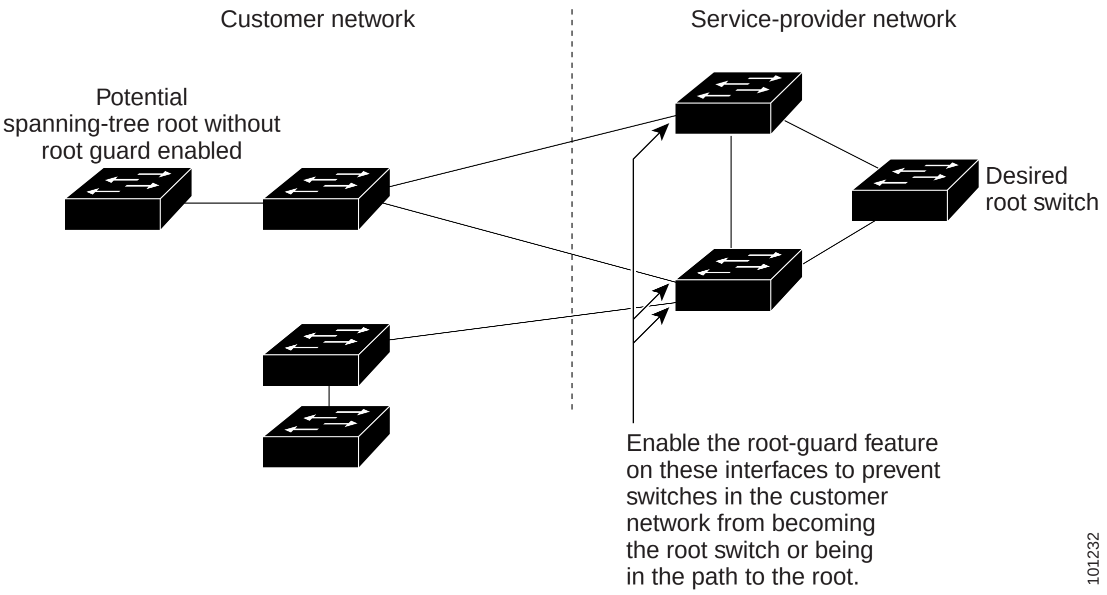
If the router is operating in multiple spanning-tree (MST) mode, root guard forces the interface to be a designated port. If a boundary port is blocked in an internal spanning-tree (IST) instance because of root guard, the interface also is blocked in all MST instances. A boundary port is an interface that connects to a LAN, the designated switch of which is either an 802.1D switch or a switch with a different MST region configuration.

Root guard enabled on an interface applies to all the VLANs to which the interface belongs. VLANs can be grouped and mapped to an MST instance.

You can enable this feature by using the **spanning-tree guard root** interface configuration command.

Caution Misuse of the root-guard feature can cause a loss of connectivity.

Figure 11-2 Root Guard in a Service-Provider Network



Understanding Loop Guard

You can use loop guard to prevent alternate or root ports from becoming designated ports because of a failure that leads to a unidirectional link. This feature is most effective when it is enabled on the entire switched network. Loop guard prevents alternate and root ports from becoming designated ports, and spanning tree does not send BPDUs on root or alternate ports.

You can enable this feature by using the **spanning-tree loopguard default** global configuration command.

When the router is operating in PVST+ mode, loop guard prevents alternate and root ports from becoming designated ports, and spanning tree does not send BPDUs on root or alternate ports.

When the router is operating in MST mode, BPDUs are not sent on nonboundary ports only if the interface is blocked by loop guard in all MST instances. On a boundary port, loop guard blocks the interface in all MST instances.

Configuring Optional Spanning-Tree Features

Default Optional Spanning-Tree Configuration

Table 11-1 shows the default optional spanning-tree configuration. Only NNIs or ENIs with STP enabled participate in STP on the router. UNIs and ENIs that have not been configured for STP are always in the forwarding state.

Table 11-1 Default Optional Spanning-Tree Configuration

Feature	Default Setting
Port Fast, BPDU filtering, BPDU guard	Globally disabled (unless they are individually configured per STP port).
Root guard	Disabled on all STP ports.
Loop guard	Disabled on all STP ports.

Optional Spanning-Tree Configuration Guidelines

You can configure PortFast, BPDU guard, BPDU filtering, root guard, or loop guard if your router is running PVST+ or MSTP.

Enabling Port Fast

An STP port with the Port Fast feature enabled is moved directly to the spanning-tree forwarding state without waiting for the standard forward-time delay.

Caution Use Port Fast *only* when connecting a single end station to an access or trunk port. Enabling this feature on an interface connected to a switch or hub could prevent spanning tree from detecting and disabling loops in your network, which could cause broadcast storms and address-learning problems.

You can enable this feature if your router is running PVST+ or MSTP.

Beginning in privileged EXEC mode, follow these steps to enable Port Fast. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an STP interface to configure, and enter interface configuration mode.
Step 3	**spanning-tree portfast** [**trunk**]	Enable Port Fast on an access port connected to a single workstation or server. By specifying the **trunk** keyword, you can enable Port Fast on a trunk port. **Note** To enable Port Fast on trunk ports, you must use the **spanning-tree portfast trunk** interface configuration command. The **spanning-tree portfast** command does not work on trunk ports. ⚠ **Caution** Make sure that there are no loops in the network between the trunk port and the workstation or server before you enable Port Fast on a trunk port. By default, Port Fast is disabled on all STP ports.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show spanning-tree interface** *interface-id* **portfast**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note You can use the **spanning-tree portfast default** global configuration command to globally enable the Port Fast feature on all nontrunking STP ports.

To disable the Port Fast feature, use the **spanning-tree portfast disable** interface configuration command.

Enabling BPDU Guard

When you globally enable BPDU guard on ports that are Port Fast-enabled (the ports are in a Port Fast-operational state), spanning tree continues to run on the ports. They remain up unless they receive a BPDU.

In a valid configuration, Port Fast-enabled interfaces do not receive BPDUs. Receiving a BPDU on a Port Fast-enabled interface signals an invalid configuration, such as the connection of an unauthorized device, and the BPDU guard feature puts the interface in the error-disabled state. The BPDU guard feature provides a secure response to invalid configurations because you must manually put the interface back in service. Use the BPDU guard feature in a service-provider network to prevent an access port from participating in the spanning tree.

Caution Configure Port Fast only on STP ports that connect to end stations; otherwise, an accidental topology loop could cause a data packet loop and disrupt switch and network operation.

You also can use the **spanning-tree bpduguard enable** interface configuration command to enable BPDU guard on any STP port without also enabling the Port Fast feature. When the interface receives a BPDU, it is put in the error-disabled state.

You can enable the BPDU guard feature if your router is running PVST+ or MSTP.

Beginning in privileged EXEC mode, follow these steps to globally enable the BPDU guard feature. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree portfast bpduguard default**	Globally enable BPDU guard. (By default, BPDU guard is disabled.) **Note** Globally enabling BPDU guard enables it only on STP ports; the command has no effect on ports that are not running STP.
Step 3	**interface** *interface-id*	Specify the interface connected to an end station, and enter interface configuration mode.
Step 4	**spanning-tree portfast**	Enable the Port Fast feature.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable BPDU guard, use the **no spanning-tree portfast bpduguard default** global configuration command.

You can override the setting of the **no spanning-tree portfast bpduguard default** global configuration command by using the **spanning-tree bpduguard enable** interface configuration command on an STP port.

Enabling BPDU Filtering

When you globally enable BPDU filtering on Port Fast-enabled STP ports, it prevents interfaces that are in a Port Fast-operational state from sending or receiving BPDUs. The interfaces still send a few BPDUs at link-up before the router begins to filter outbound BPDUs. You should globally enable BPDU filtering on a switch so that hosts connected to these interfaces do not receive BPDUs. If a BPDU is received on a Port Fast-enabled STP port, the interface loses its Port Fast-operational status, and BPDU filtering is disabled.

Caution Configure Port Fast only on STP ports that connect to end stations; otherwise, an accidental topology loop could cause a data packet loop and disrupt switch and network operation.

You can also use the **spanning-tree bpdufilter enable** interface configuration command to enable BPDU filtering on any STP port without also enabling the Port Fast feature. This command prevents the STP port from sending or receiving BPDUs.

Caution Enabling BPDU filtering on an STP port is the same as disabling spanning tree on it and can result in spanning-tree loops.

You can enable the BPDU filtering feature if your router is running PVST+ or MSTP.

Beginning in privileged EXEC mode, follow these steps to globally enable the BPDU filtering feature. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree portfast bpdufilter default**	Globally enable BPDU filtering. (By default, BPDU filtering is disabled.) **Note** Globally enabling BPDU filtering enables it only on STP ports; the command has no effect on UNIs or ENIs on which STP is not enabled.
Step 3	**interface** *interface-id*	Specify the interface connected to an end station, and enter interface configuration mode.
Step 4	**spanning-tree portfast**	Enable the Port Fast feature.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable BPDU filtering, use the **no spanning-tree portfast bpdufilter default** global configuration command.

You can override the setting of the **no spanning-tree portfast bpdufilter default** global configuration command by using the **spanning-tree bpdufilter enable** interface configuration command on an STP port.

Enabling Root Guard

Root guard enabled on an STP port applies to all the VLANs to which the port belongs.

Note You cannot enable both root guard and loop guard at the same time.

You can enable this feature if your router is running PVST+ or MSTP.

Beginning in privileged EXEC mode, follow these steps to enable root guard on an interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode.
Step 3	**spanning-tree guard root**	Enable root guard on the STP port. By default, root guard is disabled on all interfaces.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable root guard, use the **no spanning-tree guard** interface configuration command.

Enabling Loop Guard

You can use loop guard to prevent alternate or root ports from becoming designated ports because of a failure that leads to a unidirectional link. This feature is most effective when it is configured on the entire switched network. Loop guard operates only on STP ports that are considered point-to-point by the spanning tree.

Note You cannot enable both loop guard and root guard at the same time.

You can enable this feature if your router is running PVST+ or MSTP.

Beginning in privileged EXEC mode, follow these steps to enable loop guard. This procedure is optional.

	Command	Purpose
Step 1	**show spanning-tree active** or **show spanning-tree mst**	Verify which interfaces are alternate or root ports.
Step 2	**configure terminal**	Enter global configuration mode.
Step 3	**spanning-tree loopguard default**	Enable loop guard. By default, loop guard is disabled.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To globally disable loop guard, use the **no spanning-tree loopguard default** global configuration command. You can override the setting of the **no spanning-tree loopguard default** global configuration command by using the **spanning-tree guard loop** interface configuration command on an NNI.

Displaying the Spanning-Tree Status

To display the spanning-tree status, use one or more of the privileged EXEC commands in Table 11-2:

Table 11-2 Commands for Displaying the Spanning-Tree Status

Command	Purpose
show spanning-tree active	Displays spanning-tree information on active interfaces only.
show spanning-tree detail	Displays a detailed summary of interface information.
show spanning-tree interface *interface-id*	Displays spanning-tree information for the specified interface.
show spanning-tree mst interface *interface-id*	Displays MST information for the specified interface.
show spanning-tree summary [**totals**]	Displays a summary of interface states or displays the total lines of the spanning-tree state section.

You can clear spanning-tree counters by using the **clear spanning-tree** [**interface** *interface-id*] privileged EXEC command.

For information about other keywords for the **show spanning-tree** privileged EXEC command, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.

CHAPTER 12

Managing the MAC Address Table

The MAC address table contains address information that the switch uses to forward traffic between ports. All MAC addresses in the address table are associated with one or more ports. The address table includes these types of addresses:

- Dynamic address: a source MAC address that the switch learns and then ages when it is not in use.
- Static address: a manually entered unicast address that does not age and that is not lost when the switch resets.

The address table lists the destination MAC address, the associated VLAN ID, and port number associated with the address and the type (static or dynamic).

MAC address table management can be used with the STP, MSTP, and REP features. For more information about configuring SPAN, see Chapter 9, "Configuring STP." For more information about configuring RSPAN, see Chapter 10, "Configuring MSTP." For more information about configuring REP, see Chapter 14, "Configuring Resilient Ethernet Protocol."

The following sections describe how to manage the MAC address table:

- Disabling MAC Address Learning on an Interface or VLAN
- Displaying Address Table Entries

Disabling MAC Address Learning on an Interface or VLAN

By default, MAC address learning is enabled on all interfaces and VLANs on the router. You can control MAC address learning on an interface or VLAN to manage the available MAC address table space by controlling which interfaces or VLANs can learn MAC addresses. Before you disable MAC address learning, be sure that you are familiar with the network topology and the router system configuration. Disabling MAC address learning on an interface or VLAN could cause flooding in the network.

Follow these guidelines when disabling MAC address learning on an interface or VLAN:

- Use caution before disabling MAC address learning on an interface or VLAN with a configured switch virtual interface (SVI). The switch then floods all IP packets in the Layer 2 domain.
- You can disable MAC address learning on a single VLAN ID from 1 to 4094 (for example, **no mac address-table learning vlan 223**) or a range of VLAN IDs, separated by a hyphen or comma (for example, **no mac address-table learning vlan 1-10, 15**).
- We recommend that you disable MAC address learning only in VLANs with two ports. If you disable MAC address learning on a VLAN with more than two ports, every packet entering the switch is flooded in that VLAN domain.

- You cannot disable MAC address learning on a VLAN that is used internally by the router. If the VLAN ID that you enter is an internal VLAN, the switch generates an error message and rejects the command. To view internal VLANs in use, enter the **show vlan internal usage** privileged EXEC command.
- If you disable MAC address learning on a VLAN that includes a secure port, MAC address learning is not disabled on that port.

Beginning in privileged EXEC mode, follow these steps to disable MAC address learning on a VLAN:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no mac-address-table learning** {**vlan** *vlan-id* [**,***vlan-id* | **-***vlan-id*] | **interface** *interface slot*/*port*}	Disable MAC address learning on an interface or on a specified VLAN or VLANs. You can specify a single VLAN ID or a range of VLAN IDs separated by a hyphen or comma. Valid VLAN IDs 1 to 4094. It cannot be an internal VLAN.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show mac address-table learning** [**vlan** *vlan-id* | **interface** *interface slot*/*port*]	Verify the configuration.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To reenable MAC address learning on an interface or VLAN, use the **default mac address-table learning** global configuration command. You can also reenable MAC address learning on a VLAN by entering the **mac address-table learning** global configuration command. The first (**default**) command returns to a default condition and therefore does not appear in the output from the **show running-config** command. The second command causes the configuration to appear in the **show running-config** privileged EXEC command display.

This example shows how to disable MAC address learning on VLAN 200:

```
Router(config)# no mac address-table learning vlan 200
```

This example shows how to disable MAC-address learning for all modules on a specific routed interface:

```
Router(config)# no mac-address-table learning interface GigabitEthernet 0/5
Router(config)#
```

You can display the MAC address learning status of all VLANs or a specified VLAN by entering the **show mac-address-table learning** [**vlan** *vlan-id*] privileged EXEC command.

Displaying Address Table Entries

You can display the MAC address table by using one or more of the privileged EXEC commands described in Table 12-1:

Table 12-1 Commands for Displaying the MAC Address Table

Command	Description
show mac address-table address	Displays MAC address table information for the specified MAC address.
show mac address-table aging-time	Displays the aging time in all VLANs or the specified VLAN.

Table 12-1 Commands for Displaying the MAC Address Table (continued)

Command	Description
show mac address-table count	Displays the number of addresses present in all VLANs or the specified VLAN.
show mac address-table dynamic	Displays only dynamic MAC address table entries.
show mac address-table interface	Displays the MAC address table information for the specified interface.
show mac address-table learning	Displays MAC address learning status of all VLANs or the specified VLAN.
show mac address-table static	Displays only static MAC address table entries.
show mac address-table vlan	Displays the MAC address table information for the specified VLAN.

CHAPTER 13

Configuring Cisco Express Forwarding

This module contains information about Cisco Express Forwarding and describes the required and optional tasks for configuring a load-balancing scheme for Cisco Express Forwarding traffic. Load-balancing allows you to optimize resources by distributing traffic over multiple paths.

Cisco Express Forwarding is an advanced Layer 3 IP switching technology. It optimizes network performance and scalability for all kinds of networks: those that carry small amounts of traffic and those that carry large amounts of traffic in complex patterns, such as the Internet and networks characterized by intensive web-based applications or interactive sessions.

The following sections describe Cisco Express Forwarding:

- Information About Cisco Express Forwarding, page 1
- Configuring Cisco Express Forwarding, page 6
- Configuration Examples for Configuring a Load-Balancing Scheme for Cisco Express Forwarding Traffic, page 11

Information About Cisco Express Forwarding

Before using Cisco Express Forwarding or distributed Cisco Express Forwarding, you should understand the following:

- Cisco Express Forwarding Benefits—Improved Performance, Scalability, and Resilience, page 1
- Media Supported by Cisco Express Forwarding, page 2
- Main Components of Cisco Express Forwarding Operation, page 3
- FIB Overview, page 3
- Cisco Express Forwarding Adjacency Tables Overview, page 4
- Cisco Express Forwarding Operation Modes—Central and Distributed, page 5

Cisco Express Forwarding Benefits—Improved Performance, Scalability, and Resilience

Cisco Express Forwarding offers the following benefits:

- Improved performance—Cisco Express Forwarding is less CPU-intensive than fast switching route caching. As a result, more CPU processing power can be dedicated to Layer 3 services such as quality of service (QoS) and encryption.

- Scalability—Cisco Express Forwarding offers full switching capacity at each line card when distributed Cisco Express Forwarding mode is active. Distributed Cisco Express Forwarding is a distributed switching mechanism that scales linearly with the number of interface cards and the bandwidth installed in the router.

Note Distributed Cisco Express Forwarding is not currently supported on the Cisco MWR 2941.

- Resilience—Cisco Express Forwarding offers an unprecedented level of switching consistency and stability in large dynamic networks. In dynamic networks, fast-switched cache entries are frequently invalidated by routing changes. These changes can cause traffic to be process-switched through use of the routing table, rather than fast switched through use of the route cache. Because the forwarding information base (FIB) lookup table contains all known routes that exist in the routing table, it eliminates the need for route cache maintenance and the steps involved with fast-switch or process-switch forwarding. Cisco Express Forwarding can switch traffic more efficiently than typical demand caching schemes.

You can use Cisco Express Forwarding in any part of a network. For example, Figure 13-1 shows Cisco Express Forwarding being run on Cisco 12000 Series Internet routers at aggregation points at the core of a network where traffic levels are high and performance is critical.

Figure 13-1 Cisco Express Forwarding Example

In a typical high-capacity Internet service provider (ISP) environment, Cisco 12000 Series Internet routers function as aggregation devices at the core of the network and support links to Cisco 7500 series routers or other feeder devices. Cisco Express Forwarding in these platforms at the network core provides the performance and scalability that networks need to respond to continued growth and steadily increasing network traffic. Cisco Express Forwarding is a distributed switching mechanism that scales linearly with the number of interface cards and the bandwidth installed in the router.

Media Supported by Cisco Express Forwarding

Cisco Express Forwarding currently supports the following media:

- ATM/AAL5snap, ATM/AAL5mux, and ATM/AAL5nlpid
- Ethernet
- FDDI

- Frame Relay
- High-Level Data Link Control (HDLC)
- PPP
- Spatial Reuse Protocol (SRP)
- TokenRing
- Tunnels

Main Components of Cisco Express Forwarding Operation

Information conventionally stored in a route cache is stored in several data structures for Cisco Express Forwarding switching. The data structures provide optimized lookup for efficient packet forwarding. The two main components of Cisco Express Forwarding operation are the forwarding information base (FIB) and the adjacency tables.

The FIB is conceptually similar to a routing table or information base. A router uses this lookup table to make destination-based switching decisions during Cisco Express Forwarding operation. The FIB is updated when changes occur in the network and contains all routes known at the time. For more information, see the "FIB Overview" section on page 3.

Adjacency tables maintain Layer 2 next-hop addresses for all FIB entries. For more information, see the "Cisco Express Forwarding Adjacency Tables Overview" section on page 4.

This separation of the reachability information (in the Cisco Express Forwarding table) and the forwarding information (in the adjacency table), provides a number of benefits:

- The adjacency table can be built separately from the Cisco Express Forwarding table, allowing both to be built without any packets being process switched.
- The MAC header rewrite used to forward a packet is not stored in cache entries, so changes in a MAC header rewrite string do not require invalidation of cache entries.

FIB Overview

Cisco Express Forwarding uses a FIB to make IP destination prefix-based switching decisions.

The FIB contains the prefixes from the IP routing table structured in a way that is optimized for forwarding. When routing or topology changes occur in the network, the IP routing table is updated, and those changes are reflected in the FIB. The FIB maintains next-hop address information based on the information in the IP routing table.

Because there is a one-to-one correlation between FIB entries and routing table entries, the FIB contains all known routes and eliminates the need for the route cache maintenance that is associated with switching paths such as those used in fast switching and optimum switching.

Cisco Express Forwarding FIB and Load Balancing

Several paths can lead to a destination prefix. This occurs, for example, when a router is configured for simultaneous load balancing and redundancy. For each resolved path, the FIB contains a pointer for the adjacency corresponding to the next hop interface for that path.

Cisco Express Forwarding Adjacency Tables Overview

A node is said to be adjacent to another node if the node can be reached with a single hop across a link layer (Layer 2). Cisco Express Forwarding stores forwarding information (outbound interface and MAC header rewrite) for adjacent nodes in a data structure called the adjacency table. Cisco Express Forwarding uses adjacency tables to prepend Layer 2 addressing information to packets. The adjacency tables maintain Layer 2 next-hop addresses for all FIB entries.

The following sections provide additional information about adjacencies:

- Adjacency Discovery, page 4
- Adjacency Types That Require Special Handling, page 4
- Unresolved Adjacency, page 5

Adjacency Discovery

Each adjacency table is populated as adjacencies are discovered. Adjacencies are added to the table either through indirect manual configuration or dynamically—discovered through a mechanism like Address Resolution Protocol (ARP) or added through the use of a routing protocol, such as Border Gateway Protocol (BGP) or Open Shortest Path First (OSPF), which forms neighbor relationships. Each time an adjacency entry is created, a link-layer header for that adjacent node is computed and stored in the adjacency table.

The adjacency information is subsequently used for encapsulation during Cisco Express Forwarding switching of packets.

Adjacency Types That Require Special Handling

In addition to adjacencies associated with next hop interfaces (host-route adjacencies), other types of adjacencies are used to expedite switching when certain exception conditions exist. Prefixes requiring exception processing or special handling are cached with one of the special adjacencies listed in Table 13-1.

Table 13-1 Adjacency Types That Require Special Handling

Packets of This Adjacency Type	Receive This Processing
Null adjacency	Packets destined for a Null0 interface are dropped. Null adjacency can be used as an effective form of access filtering.
Glean adjacency	When a router is connected to a multiaccess medium, the FIB table on the router maintains a prefix for the subnet rather than for the individual host prefixes. The subnet prefix points to a glean adjacency. A glean adjacency entry indicates that a particular next hop should be directly connected, but there is no MAC header rewrite information available. When the router needs to forward packets to a specific host on a subnet, Cisco Express Forwarding requests an ARP entry for the specific prefix, ARP sends the MAC address, and the adjacency entry for the host is built.

Table 13-1 Adjacency Types That Require Special Handling (continued)

Packets of This Adjacency Type	Receive This Processing
Punt adjacency	The router forwards packets that require special handling or packets sent by features that are not yet supported in conjunction with Cisco Express Forwarding switching paths to the next higher switching level for handling.
Discard adjacency	The router discards the packets.
Drop adjacency	The router drops the packets.

Unresolved Adjacency

When a link-layer header is prepended to a packet, the FIB requires the prepended header to point to an adjacency corresponding to the next hop. If an adjacency was created by the FIB and not discovered through a mechanism such as ARP, the Layer 2 addressing information is not known and the adjacency is considered incomplete or unresolved. Once the Layer 2 information is known, the packet is forwarded to the RP, and the adjacency is determined through ARP. Thus, the adjacency is resolved.

Cisco Express Forwarding Operation Modes—Central and Distributed

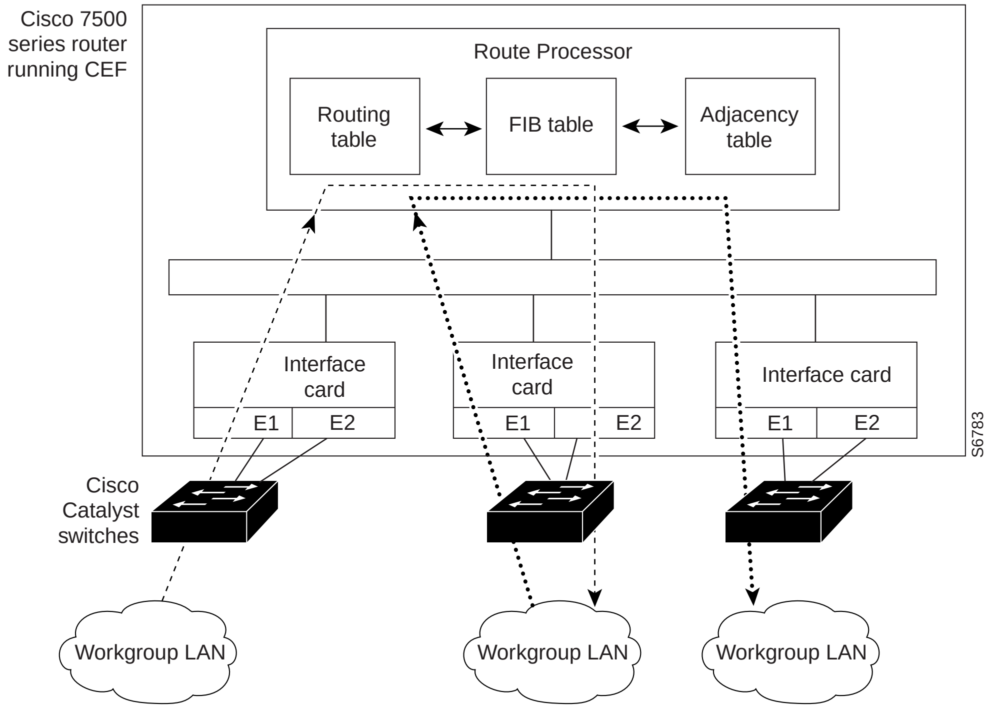
Cisco Express Forwarding can be enabled in Central or Distributed mode. The Cisco MWR 2941 supports Central mode but does not support Distributed mode.

Central Cisco Express Forwarding Mode Operation

You can use central Cisco Express Forwarding mode when line cards are not available for Cisco Express Forwarding switching, when you need to use features not compatible with distributed Cisco Express Forwarding switching, or when you are running on a nondistributed platform. When central Cisco Express Forwarding mode is enabled, the Cisco Express Forwarding FIB and adjacency tables reside on the RP, and the RP performs the express forwarding.

Figure 13-2 shows the relationship between the routing table, the FIB, and the adjacency table during central Cisco Express Forwarding mode operation. The Catalyst switches forward traffic from workgroup LANs to a Cisco 7500 series router on the enterprise backbone running central Cisco Express Forwarding. The RP performs the express forwarding.

Figure 13-2 Central Cisco Express Forwarding Mode Operation



Configuring Cisco Express Forwarding

Cisco Express Forwarding load balancing is based on a combination of source and destination packet information; it allows you to optimize resources by distributing traffic over multiple paths.

You can configure load balancing on a per-destination or per-packet basis. Because load-balancing decisions are made on the outbound interface, load balancing must be configured on the outbound interface.

Note The Cisco MWR 2941 does not currently support per-packet load balancing.

The following sections describe how to configure Cisco Express Forwarding.

- Supported Features, page 7
- How to Configure a Load-Balancing Scheme for Cisco Express Forwarding Traffic, page 7
- Enabling or Disabling Cisco Express Forwarding Per-Destination Load Balancing, page 8
- Selecting a Cisco Express Forwarding Load-Balancing Algorithm, page 8
- Selecting a Cisco Express Forwarding Load-Balancing Algorithm: Example, page 11

Supported Features

Cisco IOS Release 15.0(1)MR supports the following CEF features:

- Per-Destination Load Balancing for Cisco Express Forwarding Traffic
- Load-Balancing Algorithms for Cisco Express Forwarding Traffic

Per-Destination Load Balancing for Cisco Express Forwarding Traffic

Per-destination load balancing allows the router to use multiple paths to achieve load sharing across multiple source-destination host pairs. Packets for a given source-destination host pair are guaranteed to take the same path, even if multiple paths are available. Traffic streams destined for different pairs tend to take different paths.

Per-destination load balancing is enabled by default when you enable Cisco Express Forwarding. To use per-destination load balancing, you do not perform any additional tasks once Cisco Express Forwarding is enabled. Per-destination is the load-balancing method of choice for most situations.

Because per-destination load balancing depends on the statistical distribution of traffic, load sharing becomes more effective as the number of source-destination host pairs increases.

You can use per-destination load balancing to ensure that packets for a given host pair arrive in order. All packets intended for a certain host pair are routed over the same link (or links).

Typically, you disable per-destination load balancing when you want to enable per-packet load balancing.

Load-Balancing Algorithms for Cisco Express Forwarding Traffic

The following load-balancing algorithms are provided for use with Cisco Express Forwarding traffic. You select a load-balancing algorithm with the **ip cef load-sharing algorithm** command.

- Universal algorithm—The universal load-balancing algorithm allows each router on the network to make a different load sharing decision for each source-destination address pair, which resolves load-sharing imbalances. The router is set to perform universal load sharing by default.
- Include-ports algorithm—The include-ports algorithm allows you to use the Layer 4 source and destination ports as part of the load-balancing decision. This method benefits traffic streams running over equal cost paths that are not load shared because the majority of the traffic is between peer addresses that use different port numbers, such as Real-Time Protocol (RTP) streams.

Note Cisco IOS Release 15.0(1)MR does not support the **original** or **tunnel** algorithms.

How to Configure a Load-Balancing Scheme for Cisco Express Forwarding Traffic

Perform the following tasks to configure and fine-tune load balancing for Cisco Express Forwarding:

- Enabling or Disabling Cisco Express Forwarding Per-Destination Load Balancing, page 8 (optional)
- Selecting a Cisco Express Forwarding Load-Balancing Algorithm, page 8 (optional)

Enabling or Disabling Cisco Express Forwarding Per-Destination Load Balancing

Perform this task to enable or disable Cisco Express Forwarding per-destination load balancing.

Note The Cisco MWR 2941 router does not support per-packet load balancing.

Cisco Express Forwarding per-destination load balancing is enabled by default on the Cisco MWR 2941; therefore no configuration is required to use this feature.

Note You cannot disable per-destination load balancing.

To display information about CEF sessions, use the **show ip cef exact-route** command.

Selecting a Cisco Express Forwarding Load-Balancing Algorithm

Perform one of the following tasks to elect a Cisco Express Forwarding load-balancing algorithm.

- Selecting a Tunnel Load-Balancing Algorithm for Cisco Express Forwarding Traffic, page 8
- Selecting an Include-Ports Layer 4 Load-Balancing Algorithm for Cisco Express Forwarding Traffic, page 9

The router is set to perform universal load sharing by default.

Selecting a Tunnel Load-Balancing Algorithm for Cisco Express Forwarding Traffic

Perform the following task to select a tunnel load-balancing algorithm for Cisco Express Forwarding traffic. Select the tunnel algorithm when your network environment contains only a few source and destination pairs.

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose

Step 3	**ip cef load-sharing algorithm** {**universal** [*id*] **include-ports** {**source** [*id*] | [**destination**] [*id*] | **source** [*id*] **destination** [*id*]}}
	Example:
	`Router(config)# ip cef load-sharing algorithm universal 111`
		Selects a Cisco Express Forwarding load-balancing algorithm.
		• The **universal** keyword sets the load-balancing algorithm to one that uses a source and destination and an ID hash.
		• The **include-ports source** keyword sets the load-balancing algorithm to one that uses the source port.
		• The *id* argument is a fixed identifier.
		• The **include-ports destination** keyword sets the load-balancing algorithm to one that uses the destination port.
		• The **include-ports source destination** keyword sets the load-balancing algorithm to one that uses the source and destination ports.

Step 4	**end**
	Example:
	`Router(config)# end`
		Exits to privileged EXEC mode.

Step 5	**show ip cef** [**vrf** *vrf-name*] **exact-route** [**platform**] *source-address destination-address*	(Optional) Displays the exact route for a CEF session. You can use the **platform** keyword to display the exact route for a hardware session.

Selecting an Include-Ports Layer 4 Load-Balancing Algorithm for Cisco Express Forwarding Traffic

Perform the following task to select an include-ports load-balancing algorithm for Cisco Express Forwarding traffic. Select the include-port algorithm when your network environment contains traffic running over equal-cost paths that is not load shared because the majority of the traffic is between peer addresses with different port numbers, such as RTP streams.

Prerequisites

Your system must be using Cisco IOS Release 15.0(1)MR or a later release.

Restrictions

The Layer 4 load-balancing algorithm applies to software switched packets.

For platforms that switch traffic using a hardware forwarding engine, the hardware load-balancing decision might be different from the software load-balancing decision for the same traffic stream. You might want to override the configured algorithm.

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

Step 3	**ip cef load-sharing algorithm** {**universal** [*id*] **include-ports** {**source** [*id*] | [**destination**] [*id*] | **source** [*id*] **destination** [*id*]}}
	Example:
	`Router(config)# ip cef load-sharing algorithm universal 111`
		Selects a Cisco Express Forwarding load-balancing algorithm.
		• The **universal** keyword sets the load-balancing algorithm to one that uses a source and destination and an ID hash.
		• The **include-ports source** keyword sets the load-balancing algorithm to one that uses the source port.
		• The *id* argument is a fixed identifier.
		• The **include-ports destination** keyword sets the load-balancing algorithm to one that uses the destination port.
		• The **include-ports source destination** keyword sets the load-balancing algorithm to one that uses the source and destination ports.

Step 4	**end**
	Example:
	`Router(config)# end`
		Exits to privileged mode.

Configuration Examples for Configuring a Load-Balancing Scheme for Cisco Express Forwarding Traffic

This section provides the following examples for configuring a load-balancing scheme for Cisco Express Forwarding traffic.

Selecting a Cisco Express Forwarding Load-Balancing Algorithm: Example

The router is set to perform universal load balancing by default.

The following examples show how to select a different Cisco Express Forwarding load-balancing algorithm:

- Selecting a Tunnel Load-Balancing Algorithm for Cisco Express Forwarding Traffic: Example, page 11
- Selecting an Include-Ports Layer 4 Load-Balancing Algorithm for Cisco Express Forwarding Traffic: Example, page 11

Selecting a Tunnel Load-Balancing Algorithm for Cisco Express Forwarding Traffic: Example

The following example shows how to select a tunnel load-balancing algorithm for Cisco Express Forwarding:

```
configure terminal
!
ip cef load-sharing algorithm universal 111
end
```

The following example shows how to disable the tunnel load-balancing algorithm:

```
configure terminal
!
no ip cef load-sharing algorithm universal 111
end
```

Selecting an Include-Ports Layer 4 Load-Balancing Algorithm for Cisco Express Forwarding Traffic: Example

The following example shows how to select an include-ports Layer 4 load-balancing algorithm for Cisco Express Forwarding traffic:

```
configure terminal
!
ip cef load-sharing algorithm include-ports source
end
```

This example sets up load sharing that includes the source port in the load-balancing decision.

To disable the include-ports Layer 4 load-balancing algorithm and return to the default universal mode, enter the following commands:

```
configure terminal
!
no ip cef load-sharing algorithm
```

```
end
```

CHAPTER 14

Configuring Resilient Ethernet Protocol

Resilient Ethernet Protocol (REP) is a Cisco proprietary protocol that provides an alternative to Spanning Tree Protocol (STP) to control network loops, to respond to link failures, and to improve convergence time. REP controls a group of ports connected in a segment, ensures that the segment does not create any bridging loops, and responds to link failures within the segment. REP provides a basis for constructing more complex networks and supports VLAN load balancing.

The following sections describe how to configure REP:

- Understanding Resilient Ethernet Protocol, page 14-1
- Configuring Resilient Ethernet Protocol (REP), page 14-6
- Configuration Examples for REP, page 14-16

Understanding Resilient Ethernet Protocol

The following sections provide further information about REP:

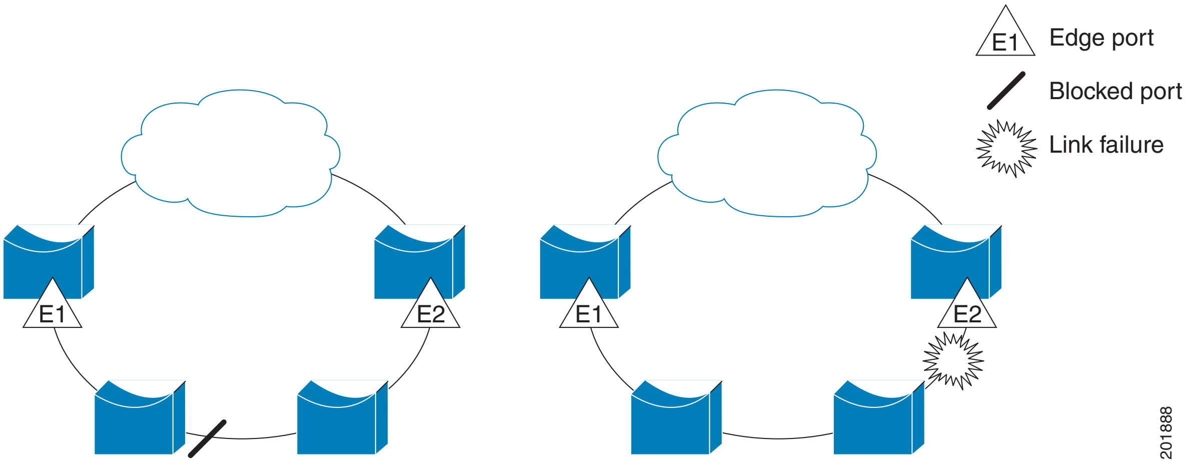
- Overview
- Link Integrity
- Fast Convergence
- VLAN Load Balancing
- REP Ports

Overview

A REP segment is a chain of ports connected to each other and configured with a segment ID. Each segment consists of standard (nonedge) segment ports and two user-configured edge ports. A switch can have only two ports belonging to the same segment, and each segment port can have only one external neighbor. A segment can go through a shared medium, but on any link, only two ports can belong to the same segment. REP is supported only on Layer 2 trunk interfaces.

Figure 14-1 shows an example of a segment consisting of six ports spread across four switches. Ports E1 and E2 are configured as edge ports. When all ports are operational (as in the segment on the left), a single port is blocked, shown by the diagonal line. When there is a network failure, as shown in the diagram on the right, the blocked port returns to the forwarding state to minimize network disruption.

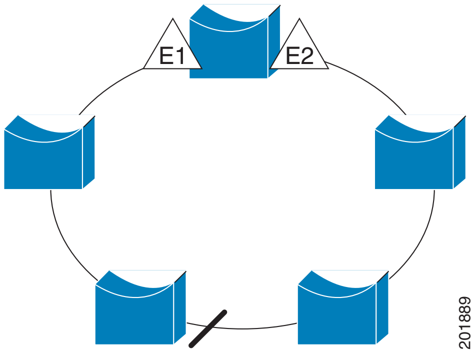
Figure 14-1 REP Open Segments



The segment shown in Figure 14-1 is an open segment; there is no connectivity between the two edge ports. The REP segment cannot cause a bridging loop, and you can safely connect the segment edges to any network. All hosts connected to switches inside the segment have two possible connections to the rest of the network through the edge ports, but only one connection is accessible at any time. If a host cannot access its usual gateway because of a failure, REP unblocks all ports to ensure that connectivity is available through the other gateway.

The segment shown in Figure 14-2, with both edge ports located on the same switch, is a ring segment. In this configuration, there is connectivity between the edge ports through the segment. With this configuration, you can create a redundant connection between any two switches in the segment.

Figure 14-2 REP Ring Segment



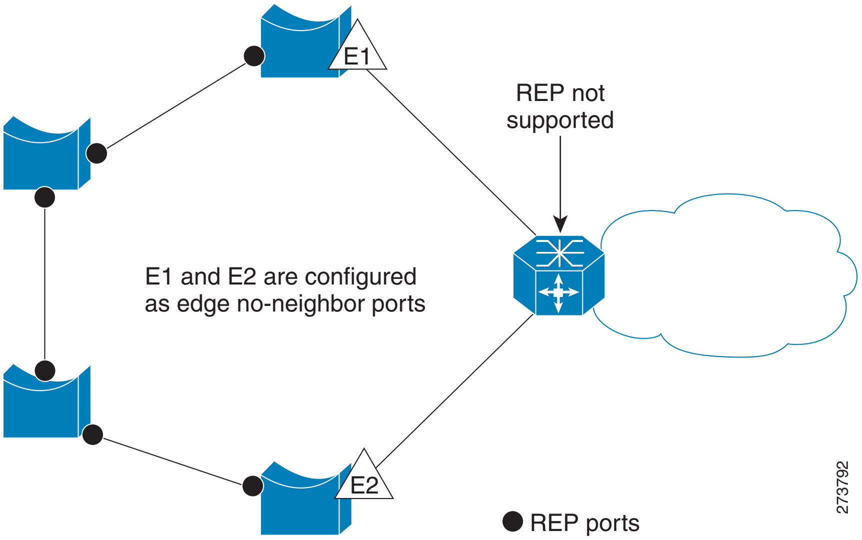
REP segments have these characteristics:

- If all ports in the segment are operational, one port (referred to as the *alternate* port) is in the blocked state for each VLAN.
- If VLAN load balancing is configured, two ports in the segment control the blocked state of VLANs.
- If one or more ports in a segment is not operational, causing a link failure, all ports forward traffic on all VLANs to ensure connectivity.
- In case of a link failure, the alternate ports are unblocked as quickly as possible. When the failed link comes back up, a logically blocked port per VLAN is selected with minimal disruption to the network.

You can construct almost any type of network based on REP segments. REP also supports VLAN load-balancing, controlled by the primary edge port but occurring at any port in the segment.

In access ring topologies, the neighboring switch might not support REP, as shown in Figure 14-3. In this case, you can configure the non-REP facing ports (E1 and E2) as edge no-neighbor ports. These ports inherit all properties of edge ports, and you can configure them the same as any edge port, including configuring them to send STP or REP topology change notices to the aggregation switch. In this case the STP topology change notice (TCN) that is sent is a multiple spanning-tree (MST) STP message.

Figure 14-3 No-neighbor Topology



REP has these limitations:

- You must configure each segment port; an incorrect configuration can cause forwarding loops in the networks.
- REP can manage only a single failed port within the segment; multiple port failures within the REP segment cause loss of network connectivity.
- You should configure REP only in networks with redundancy. Configuring REP in a network without redundancy causes loss of connectivity.

Link Integrity

REP does not use an end-to-end polling mechanism between edge ports to verify link integrity. It implements local link failure detection. The REP Link Status Layer (LSL) detects its REP-aware neighbor and establishes connectivity within the segment. All VLANs are blocked on an interface until it detects the neighbor. After the neighbor is identified, REP determines which neighbor port should become the alternate port and which ports should forward traffic.

Each port in a segment has a unique port ID. The port ID format is similar to that used by the spanning tree algorithm: a port number (unique on the bridge), associated to a MAC address (unique in the network). When a segment port is coming up, its LSL starts sending packets that include the segment ID and the port ID. The port is declared operational after it performs a three-way handshake with a neighbor in the same segment.

A segment port does not become operational if:

- No neighbor has the same segment ID.
- More than one neighbor has the same segment ID.
- The neighbor does not acknowledge the local port as a peer.

Each port creates an adjacency with its immediate neighbor. After the neighbor adjacencies are created, the ports negotiate to determine one blocked port for the segment, the alternate port. All other ports become unblocked. By default, REP packets are sent to a BPDU class MAC address. The packets can also be sent to the Cisco multicast address, which is used only to send blocked port advertisement (BPA) messages when there is a failure in the segment. The packets are dropped by devices not running REP.

Fast Convergence

Because REP runs on a physical link basis and not a per-VLAN basis, only one hello message is required for all VLANs, reducing the load on the protocol. We recommend that you create VLANs consistently on all switches in a given segment and configure the same allowed VLANs on the REP trunk ports. To avoid the delay introduced by relaying messages in software, REP also allows some packets to be flooded to a regular multicast address. These messages operate at the hardware flood layer (HFL) and are flooded to the whole network, not just the REP segment. Switches that do not belong to the segment treat them as data traffic. You can control flooding of these messages by configuring a dedicated administrative VLAN for the whole domain.

The estimated convergence recovery time on fiber interfaces is less than 200 ms for the local segment with 200 VLANs configured. Convergence for VLAN load balancing is 300 ms or less.

VLAN Load Balancing

One edge port in the REP segment acts as the primary edge port; the other as the secondary edge port. The primary edge port always participates in VLAN load balancing in the segment. REP VLAN balancing is achieved by blocking some VLANs at a configured alternate port and all other VLANs at the primary edge port. When you configure VLAN load balancing, you can specify the alternate port in one of three ways:

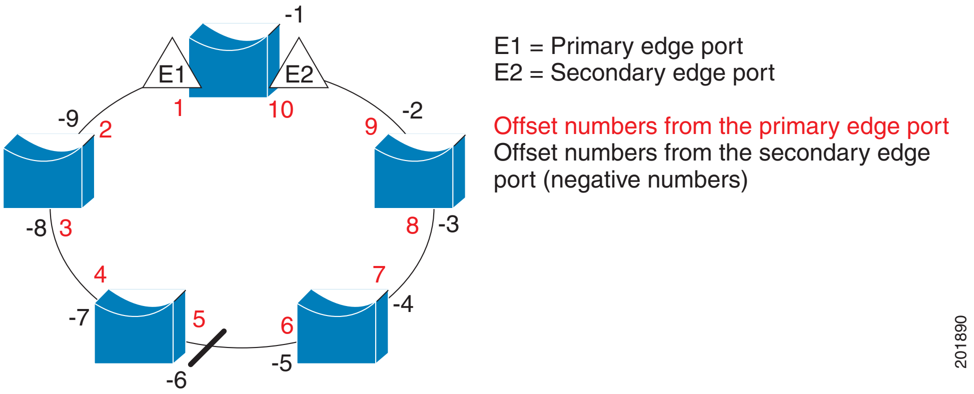
- Enter the port ID of the interface. To identify the port ID of a port in the segment, enter the **show interface rep detail** interface configuration command for the port.
- Enter the neighbor offset number of a port in the segment, which identifies the downstream neighbor port of an edge port. The neighbor offset number range is –256 to +256; a value of 0 is invalid. The primary edge port has an offset number of 1; positive numbers above 1 identify downstream neighbors of the primary edge port. Negative numbers identify the secondary edge port (offset number -1) and its downstream neighbors.

Note You configure offset numbers on the primary edge port by identifying the downstream position from the primary (or secondary) edge port. You would never enter an offset value of 1 because that is the offset number of the primary edge port itself.

Figure 14-4 shows neighbor offset numbers for a segment where E1 is the primary edge port and E2 is the secondary edge port. The red numbers inside the ring are numbers offset from the primary edge port; the black numbers outside the ring show the offset numbers from the secondary edge port. Note that you can identify all ports (except the primary edge port) by either a positive offset number (downstream position from the primary edge port) or a negative offset number (downstream position from the secondary edge port). If E2 became the primary edge port, its offset number would then be 1, and E1 would be -1.

- By entering the **preferred** keyword to select the port that you previously configured as the preferred alternate port with the **rep segment** *segment-id* **preferred** interface configuration command.

Figure 14-4 Neighbor Offset Numbers in a Segment


-1
E1
E2
1
10
-9
2
9
-2
-8
3
8
-3
4
7
-7
5
6
-4
-6
-5
E1 = Primary edge port
E2 = Secondary edge port
Offset numbers from the primary edge port
Offset numbers from the secondary edge port (negative numbers)
201890

When the REP segment is complete, all VLANs are blocked. When you configure VLAN load balancing, you must also configure triggers in one of two ways

- Manually trigger VLAN load balancing at any time by entering the **rep preempt segment** *segment-id* privileged EXEC command on the switch that has the primary edge port.
- Configure a preempt delay time by entering the **rep preempt delay** *seconds* interface configuration command. After a link failure and recovery, VLAN load balancing begins after the configured preemption time period elapses. Note that the delay timer restarts if another port fails before the time has elapsed.

Note When VLAN load balancing is configured, it does not start working until triggered by either manual intervention or a link failure and recovery.

When VLAN load balancing is triggered, the primary edge port sends a message to alert all interfaces in the segment about the preemption. When the secondary port receives the message, it is reflected into the network to notify the alternate port to block the set of VLANs specified in the message and to notify the primary edge port to block the remaining VLANs.

You can also configure a particular port in the segment to block all VLANs. Only the primary edge port initiates VLAN load balancing, which is not possible if the segment is not terminated by an edge port on each end. The primary edge port determines the local VLAN load balancing configuration.

Reconfigure the primary edge port to reconfigure load balancing. When you change the load balancing configuration, the primary edge port again waits for the **rep preempt segment** command or for the configured preempt delay period after a port failure and recovery before executing the new configuration. If you change an edge port to a regular segment port, the existing VLAN load balancing status does not change. Configuring a new edge port might cause a new topology configuration.

Note Do not configure VLAN load balancing on an interface that carries Ethernet over multiprotocol label switching (EoMPLS) traffic. VLAN load balancing across the REP ring might prevent forwarding some of the EoMPLS traffic.

Spanning Tree Interaction

REP does not interact with STP or with the Flex Link feature, but can coexist with both. A port that belongs to a segment is removed from spanning tree control and STP BPDUs are not accepted or sent from segment ports.

To migrate from an STP ring configuration to REP segment configuration, begin by configuring a single port in the ring as part of the segment, and continue by configuring contiguous ports to minimize the number of segments. Each segment always contains a blocked port, so multiple segments means multiple blocked ports and a potential loss of connectivity. When the segment has been configured in both directions to the edge ports, you then configure the edge ports.

REP Ports

Ports in REP segments are Failed, Open, or Alternate.

- A port configured as a regular segment port starts as a failed port.
- After the neighbor adjacencies are determined, the port changes to alternate port state, blocking all VLANs on the interface. Blocked port negotiations occur and when the segment settles, one blocked port remains in the alternate role, and all other ports become open ports.
- When a failure occurs in a link, all ports move to the open state. When the alternate port receives the failure notification, it changes to the open state, forwarding all VLANs.

A regular segment port converted to an edge port, or an edge port converted to a regular segment port, does not always result in a topology change. If you convert an edge port into a regular segment port, VLAN load balancing is not implemented unless it has been configured. For VLAN load balancing, you must configure two edge ports in the segment.

A segment port that is reconfigured as a spanning tree port restarts according the spanning tree configuration. By default, this is a designated blocking port. If PortFast is configured or if STP is disabled, the port goes into the forwarding state.

For instructions on how to configure REP, see Configuring Resilient Ethernet Protocol (REP), page 14-6.

Configuring Resilient Ethernet Protocol (REP)

A segment is a collection of ports connected one to the other in a chain and configured with a segment ID. To configure REP segments, you configure the REP administrative VLAN (or use the default VLAN 1) and then add the ports to the segment using interface configuration mode. You should configure two edge ports in the segment, one as the primary edge port and the other, by default, the secondary edge port. A segment has only one primary edge port. If you configure two ports in a segment as the primary edge port, for example ports on different switches, the REP selects one to serve as the segment primary edge port. You can also optionally configure where to send segment topology change notices (STCNs) and VLAN load balancing messages.

The following sections describe how to configure REP on the Cisco MWR 2941:

Default REP Configuration

REP is disabled on all interfaces. When enabled, the interface is a regular segment port unless it is configured as an edge port.

When REP is enabled, the sending of segment topology change notices (STCNs) is disabled, all VLANs are blocked, and the administrative VLAN is VLAN 1.

When VLAN load balancing is enabled, the default is manual preemption with the delay timer disabled. If VLAN load balancing is not configured, the default after manual preemption is to block all VLANs at the primary edge port.

REP Configuration Guidelines

Follow these guidelines when configuring REP:

- We recommend that you begin by configuring one port and then configure the contiguous ports to minimize the number of segments and the number of blocked ports.
- If more than two ports in a segment fail when no external neighbors are configured, one port goes into a forwarding state for the data path to help maintain connectivity during configuration. In the **show rep interface** privileged EXEC command output, the Port Role for this port shows as *Fail Logical Open*; the Port Role for the other failed port shows as *Fail No Ext Neighbor*. When the external neighbors for the failed ports are configured, the ports go through the alternate port state transitions and eventually go to an open state or remain as the alternate port, based on the alternate port election mechanism.
- REP ports must be Layer 2 trunk ports.
- Be careful when configuring REP through a Telnet connection. Because REP blocks all VLANs until another REP interface sends a message to unblock the VLAN, you might lose connectivity to the router if you enable REP in a Telnet session that accesses the router through the REP interface.
- You cannot run REP and STP or REP and Flex Links on the same segment or interface.
- If you connect an STP network to the REP segment, be sure that the connection is at the segment edge. An STP connection that is not at the edge could cause a bridging loop because STP does not run on REP segments. All STP BPDUs are dropped at REP interfaces.
- You must configure all trunk ports in the segment with the same set of allowed VLANs, or a misconfiguration occurs.
- REP ports follow these rules:
  - There is no limit to the number of REP ports on a switch; however, only two ports on a switch can belong to the same REP segment.

  - If only one port on a switch is configured in a segment, the port should be an edge port.
  - If two ports on a switch belong to the same segment, they must be both edge ports, both regular segment ports, or one regular port and one edge no-neighbor port. An edge port and regular segment port on a switch cannot belong to the same segment.
  - If two ports on a switch belong to the same segment and one is configured as an edge port and one as a regular segment port (a misconfiguration), the edge port is treated as a regular segment port.
- REP interfaces come up and remain in a blocked state until notified that it is safe to unblock. You need to be aware of this to avoid sudden connection losses.
- REP sends all LSL PDUs in untagged frames on the native VLAN. The BPA message sent to the Cisco multicast address is sent on the administration VLAN, which is VLAN 1 by default.
- You can configure how long a REP interface remains up without receiving a hello from a neighbor. You can use the **rep lsl-age-timer** *value* interface configuration command to set the time from 120 ms to 10000 ms. The LSL hello timer is then set to the age-timer value divided by three. In normal operation, three LSL hellos are sent before the age timer on the peer switch expires and searches for hello messages.
- You can configure how long a REP interface remains up without receiving a hello from a neighbor. You can use the **rep lsl-age-timer** value interface configuration command to set the time from 120 ms to 10000 ms. The LSL hello timer is then set to the age-timer value divided by three. In normal operation, three LSL hellos are sent before the age timer on the peer switch expires and searches for hello messages.
- REP ports cannot be configured as one of these port types:
  - SPAN destination port
  - Private VLAN
  - Tunnel port
  - Access port
- There is a maximum of 64 REP segments per switch.

Configuring the REP Administrative VLAN

To avoid the delay introduced by relaying messages in software for link-failure or VLAN-blocking notification during load balancing, REP floods packets at the hardware flood layer (HFL) to a regular multicast address. These messages are flooded to the whole network, not just the REP segment. You can control flooding of these messages by configuring an administrative VLAN for the whole domain.

Follow these guidelines when configuring the REP administrative VLAN:

- If you do not configure an administrative VLAN, the default is VLAN 1.
- There can be only one administrative VLAN on a switch and on a segment. However, this is not enforced by software.

Beginning in privileged EXEC mode, follow these steps to configure the REP administrative VLAN:

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**rep admin vlan** *vlan-id* **Example:** `Router(config)# rep admin vlan 1`	Configures a REP administrative VLAN. • Specify the administrative VLAN. The range is 1–4094. The default is VLAN 1.

Step 4	**end**
	Example:
	`Router(config)# end`
		Returns to privileged EXEC mode.

Step 5	**show interface** [*interface-id*] **rep** [**detail**] **Example:** `Router# show interface gigabitethernet0/1 rep detail`	Displays the REP configuration and status for a specified interface. • Enter the physical interface.
Step 6	**copy running-config startup config** **Example:** `Router# copy running-config startup config`	(Optional) Saves your entries in the router startup configuration file.

Configuring REP Interfaces

For REP operation, you need to enable it on each segment interface and to identify the segment ID. This step is required and must be done before other REP configuration. You must also configure a primary and secondary edge port on each segment. All other steps are optional.

Beginning in privileged EXEC mode, follow these steps to enable and configure REP on an interface:

	Command	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`interface` *`interface-id`*
	Example:
	`Router(config)# interface gigabitethernet0/1`
		Specifies the interface, and enters interface configuration mode.
		• Enter the interface ID. The interface can be a physical Layer 2 interface.

Step 4	`switchport mode trunk`
	Example:
	`Router(config-if)# switchport mode trunk`
		Configures the interface as a Layer 2 trunk port.

	Command	Purpose
Step 5	**rep segment** *segment-id* [**edge** [**no-neighbor**] [**primary**]] [**preferred**] **Example:** `Router(config-if)# rep segment 1 edge preferred`	Enables REP on the interface, and identifies a segment number. The segment ID range is from 1 to 1024. **Note** You must configure two edge ports, including one primary edge port for each segment. These optional keywords are available. • Enter **edge** to configure the port as an edge port. Entering **edge** without the **primary** keyword configures the port as the secondary edge port. Each segment has only two edge ports. • (Optional) Enter **no-neighbor** to configure a port with no external REP neighbors as an edge port. The port inherits all properties of edge ports, and you can configure them the same as any edge port. • On an edge port, enter **primary** to configure the port as the primary edge port, the port on which you can configure VLAN load balancing. **Note** Although each segment can have only one primary edge port, if you configure edge ports on two different switches and enter the **primary** keyword on both switches, the configuration is allowed. However, REP selects only one of these ports as the segment primary edge port. You can identify the primary edge port for a segment by entering the **show rep topology** privileged EXEC command. • Enter **preferred** to indicate that the port is the preferred alternate port or the preferred port for VLAN load balancing. **Note** Configuring a port as preferred does not guarantee that it becomes the alternate port; it merely gives it a slight edge among equal contenders. The alternate port is usually a previously failed port.
Step 6	**rep lsl-retries** *number-of-retries* **Example:** `Router(config-if)# rep lsl-retries 4`	Use the **rep lsl-retries** command to configure the REP link status layer (LSL) number of retries before the REP link is disabled.

Step 7	**rep stcn** {**interface** *interface-id* | **segment** *id-list* | **stp**}
	Example:
	`Router(config-if)# rep stcn segment 2-5`
		(Optional) Configures the edge port to send segment topology change notices (STCNs).
		• Enter **interface** *interface-id* to designate a physical interface to receive STCNs.
		• Enter **segment** *id-list* to identify one or more segments to receive STCNs. The range is from 1–1024.
		• Enter **stp** to send STCNs to STP networks.

	Command	Purpose

Step 8	**rep block port** {**id** *port-id* | *neighbor-offset* | **preferred**} **vlan** {*vlan-list* | **all**}
	Example:
	`Router(config-if)#` **`rep block port 0009001818D68700 vlan all`**
		(Optional) Configures VLAN load balancing on the primary edge port, identifies the REP alternate port in one of three ways, and configures the VLANs to be blocked on the alternate port.
		• Enter the **id** *port-id* to identify the alternate port by port ID. The port ID is automatically generated for each port in the segment. You can view interface port IDs by entering the **show interface** *interface-id* **rep** [**detail**] privileged EXEC command.
		• Enter a *neighbor-offset* number to identify the alternate port as a downstream neighbor from an edge port. The range is from –256 to 256, with negative numbers indicating the downstream neighbor from the secondary edge port. A value of 0 is invalid. Enter **-1** to identify the secondary edge port as the alternate port.
		Note Because you enter this command at the primary edge port (offset number 1), you would never enter an offset value of 1 to identify an alternate port.
		• Enter **preferred** to select the regular segment port previously identified as the preferred alternate port for VLAN load balancing.
		• Enter **vlan** *vlan-list* to block one VLAN or a range of VLANs.
		• Enter **vlan all** to block all VLANs.
		Note Enter this command only on the REP primary edge port.

Step 9	**rep preempt delay** *seconds* **Example:** `Router(config-if)#` **`rep preempt delay 60`**	(Optional) Configures a preempt time delay. Use this command if you want VLAN load balancing to automatically trigger after a link failure and recovery. The time delay range is 15 to 300 seconds. The default is manual preemption with no time delay. **Note** Use this command only on the REP primary edge port.
Step 10	**rep lsl-age-timer** *value* **Example:** `Router(config-if)` **`rep lsl-age-timer 5000`**	(Optional) Configure a time (in milliseconds) for which the REP interface remains up without receiving a hello from a neighbor. The range is from 120 to 10000 ms in 40-ms increments; the default is 5000 ms (5 seconds).

Step 11	**end**
	Example:
	`Router(config-if)# end`
		Returns to privileged EXEC mode.

Step 12	**show interface** [*interface-id*] **rep** [**detail**] **Example:** `Router(config-if)# show interface gigabitethernet0/1 rep detail`	Verifies the REP interface configuration. • Enter the interface ID and the optional **detail** keyword, if desired.

	Command	Purpose

Step 13	**`show rep topology [segment`** *`segment-id`***`]`**
	`[archive] [detail]`
	
	Example:
	`Router#` **`show rep topology segment 1`**
	`REP Segment 1`
	`BridgeName       PortName   Edge Role`
	`---------------- ---------- ---- ----`
	`sw1_multseg_3750 Gi1/1/1    Pri  Alt`
	`sw3_multseg_3400 Gi0/13          Open`
	`sw3_multseg_3400 Gi0/14          Alt`
	`sw4_multseg_3400 Gi0/13          Open`
	`sw4_multseg_3400 Gi0/14          Open`
	`sw5_multseg_3400 Gi0/13          Open`
	`sw5_multseg_3400 Gi0/14          Open`
	`sw2_multseg_3750 Gi1/1/2         Open`
	`sw2_multseg_3750 Gi1/1/1         Open`
	`sw1_multseg_3750 Gi1/1/2    Sec  Open`
		Indicates which port in the segment is the primary edge port.

Step 14	**`copy running-config startup config`**
	
	Example:
	`Router(config-if)# copy running-config`
	`startup config`
		(Optional) Saves your entries in the router startup configuration file.

Setting Manual Preemption for VLAN Load Balancing

If you do not enter the **rep preempt delay** *seconds* interface configuration command on the primary edge port to configure a preemption time delay, the default is to manually trigger VLAN load balancing on the segment. Be sure to complete all other segment configuration before manually preempting VLAN load balancing. When you enter the **rep preempt segment** *segment-id* command, a confirmation message appears before the command is executed because preemption can cause network disruption.

Note Do not configure VLAN load balancing on an interface that carries Ethernet over MultiprotocolLabel Switching (EoMPLS) traffic. VLAN load balancing across the REP ring might prevent forwarding some of the EoMPLS traffic.

Beginning in privileged EXEC mode, follow these steps on the switch that has the segment primary edge port to manually trigger VLAN load balancing on a segment:

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`rep preempt segment`** *`segment-id`*
	Example:
	`Router(config)# rep preempt segment 1`
		Manually triggers VLAN load balancing on the segment.
		• Enter the segment ID.
		Note You will be asked to confirm the action before the command is executed.

Step 4	**`end`**
	Example:
	`Router(config)# end`
		Returns to privileged EXEC mode.

Step 5	**`show rep topology`**
	Example:
	`Router# show rep topology`
		Views the REP topology information.

Configuring SNMP Traps for REP

You can configure the switch to send REP-specific traps to notify the SNMP server of link operational status changes and port role changes. Beginning in privileged EXEC mode, follow these steps to configure REP traps:

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

	Command or Action	Purpose

Step 3	**`snmp mib rep trap-rate`** *`value`*
	Example:
	`Router(config)# snmp mib rep trap-rate 500`
		Enables the router to send REP traps, and sets the number of traps sent per second.
		• Enter the number of traps sent per second. The range is from 0 to 1000. The default is 0 (no limit imposed; a trap is sent at every occurrence).
		Note To remove the traps, enter the **no snmp mib rep trap-rate** command.

Step 4	**`end`**
	Example:
	`Router(config)# end`
		Returns to privileged EXEC mode.

Step 5	**`show running-config`**
	Example:
	`Router# show running-config`
		(Optional) Displays the running configuration, which you can use to verify the REP trap configuration.

Step 6	**`copy running-config startup config`**
	Example:
	`Router# copy running-config startup config`
		(Optional) Saves your entries in the router startup configuration file.

Monitoring REP

To monitor the REP configuration, complete the following steps:

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`show interface`** `[`*`interface-id`*`]` **`rep [detail]`**
	Example:
	`Router# show interface gigabitethernet0/1 rep detail`
		(Optional) Displays the REP configuration and status for a specified interface.
		• Enter the physical interface and the optional **detail** keyword.

Step 3	**`show rep topology`** `[`**`segment`** *`segment-id`*`] [`**`archive`**`] [`**`detail`**`]`
	Example:
	`Router# show rep topology`
		(Optional) Displays REP topology information for a segment or for all segments, including the primary and secondary edge ports in the segment.
		• Enter the optional keywords and arguments, as desired.

Configuration Examples for REP

Configuring the REP Administrative VLAN: Example

This example shows how to configure the administrative VLAN as VLAN 100.

```
Router# configure terminal
Router(config)# rep admin vlan 100
Router(config-if)# end
```

Configuring a REP Interface: Example

This example shows how to configure an interface as the primary edge port for segment 1, to send Spanning Tree Topology Changes Notification (STCNs) to segments 2 through 5, and to configure the alternate port as the port with port ID 0009001818D68700 to block all VLANs after a preemption delay of 60 seconds after a segment port failure and recovery.

```
Router# configure terminal
Router(config)# interface gigabitethernet0/1
Router(config-if)# rep segment 1 edge primary
Router(config-if)# rep stcn segment 2-5
Router(config-if)# rep block port 0009001818D68700 vlan all
Router(config-if)# rep preempt delay 60
Switch (config-if)# rep lsl-age-timer 6000
Router(config-if)# end
```

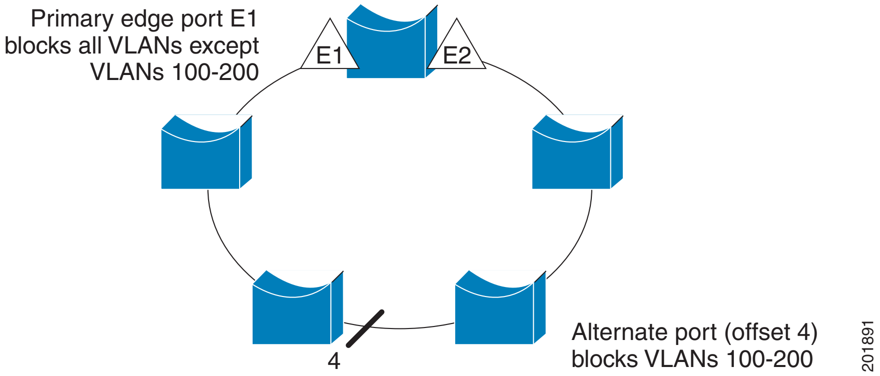
This example shows how to configure the same configuration when the interface has no external REP neighbor:

```
Router# configure terminal
Router(conf)# interface gigabitethernet0/1
Router(config-if)# rep segment 1 edge no-neighbor primary
Router(config-if)# rep stcn segment 2-5
Router(config-if)# rep block port 0009001818D68700 vlan all
Router(config-if)# rep preempt delay 60
Router(config-if)# rep lsl-age-timer 6000
```

This example shows how to configure the VLAN blocking configuration shown in Figure 5. The alternate port is the neighbor with neighbor offset number 4. After manual preemption, VLANs 100 to 200 are blocked at this port and all other VLANs are blocked at the primary edge port E1 (Gigabit Ethernet port 0/1).

```
Router# configure terminal
Router(config)# interface gigabitethernet0/1
Router(config-if)# rep segment 1 edge primary
Router(config-if)# rep block port 4 vlan 100-200
Router(config-if)# end
```

Figure 5 Example of VLAN Blocking



Setting the Preemption for VLAN Load Balancing: Example

The following is an example of setting the preemption for VLAN load balancing on a REP segment.

```
Router> enable
Router# configure terminal
Router(config)# rep preempt segment 1
Router(config)# end
```

Configuring SNMP Traps for REP: Example

This example configures the router to send REP traps at a rate of 10 traps per second:

```
Router> enable
Router# configure terminal
Router(config)# snmp mib rep trap-rate 10
Router(config)# end
```

Monitoring the REP Configuration: Example

The following is sample output of the **show interface rep detail** command. Use the **show interface rep detail** command on one of the REP interfaces to monitor and verify the REP configuration.

```
Router# show interface gigabitethernet0/1 rep detail

GigabitEthernet0/1 REP enabled
Segment-id: 2 (Edge)
PortID: 00010019E7144680
Preferred flag: No
Operational Link Status: TWO_WAY
Current Key: 0002001121A2D5800E4D
Port Role: Open
Blocked Vlan: <empty>
Admin-vlan: 100
Preempt Delay Timer: disabled
Load-balancing block port: none
Load-balancing block vlan: none
STCN Propagate to: none
LSL PDU rx: 3322, tx: 1722
HFL PDU rx: 32, tx: 5
BPA TLV rx: 16849, tx: 508
BPA (STCN, LSL) TLV rx: 0, tx: 0
BPA (STCN, HFL) TLV rx: 0, tx: 0
EPA-ELECTION TLV rx: 118, tx: 118
EPA-COMMAND TLV rx: 0, tx: 0
EPA-INFO TLV rx: 4214, tx: 4190
```

Sample MWR 2941 Topology: Example

The following configuration example shows two Cisco MWR 2941 routers and two Cisco 7600 series routers using a REP ring.

Note This section provides partial configurations intended to demonstrate a specific feature.

2941_1

```
interface GigabitEthernet0/0
 switchport trunk allowed vlan 1,2
 switchport mode trunk
 rep segment 1
!
interface GigabitEthernet0/1
 switchport trunk allowed vlan 1,2
 switchport mode trunk
 rep segment 1
!
interface GigabitEthernet0/3
 switchport access vlan 3
!
interface GigabitEthernet0/4
 switchport access vlan 4
!
interface Vlan1
 ip address 172.18.40.70 255.255.255.128
 no ptp enable
!
interface Vlan2
 ip address 1.1.1.1 255.255.255.0
 no ptp enable
!
interface Vlan3
 ip address 2.2.2.2 255.255.255.0
 no ptp enable
!
interface Vlan3
 ip address 4.4.4.2 255.255.255.0
 no ptp enable
!
ip route 3.3.3.0 255.255.255.0 1.1.1.4
ip route 5.5.5.0 255.255.255.0 1.1.1.4
```

2941_2

```
interface GigabitEthernet0/0
 switchport trunk allowed vlan 1,2
 switchport mode trunk
 rep segment 1
!
interface GigabitEthernet0/1
 switchport trunk allowed vlan 1,2
 switchport mode trunk
 rep segment 1
!
interface Vlan1
 ip address 172.18.44.239 255.255.255.0
 no ptp enable
!
interface Vlan2
 ip address 1.1.1.2 255.255.255.0
 no ptp enable
```

7600_1

```
interface Port-channel69
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,2
 switchport mode trunk
!
interface GigabitEthernet3/25
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,2
 switchport mode trunk
 channel-group 69 mode on
!
interface GigabitEthernet3/26
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,2
 switchport mode trunk
 channel-group 69 mode on
!
 interface GigabitEthernet3/35
 ip address 3.3.3.2 255.255.255.0
!
interface GigabitEthernet3/36
 ip address 5.5.5.2 255.255.255.0
!
interface GigabitEthernet5/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,2
 switchport mode trunk
 rep segment 1 edge
!
interface Vlan1
 no ip address
!
interface Vlan2
 ip address 1.1.1.4 255.255.255.0
!
ip route 2.2.2.0 255.255.255.0 1.1.1.1
ip route 4.4.4.0 255.255.255.0 1.1.1.1
```

7600_2

```
interface Port-channel69
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,2
 switchport mode trunk
!
interface GigabitEthernet5/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,2
 switchport mode trunk
 rep segment 1 edge
!
interface GigabitEthernet7/25
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,2
 switchport mode trunk
 channel-group 69 mode on
!
interface GigabitEthernet7/26
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,2
 switchport mode trunk
 channel-group 69 mode on
!
interface Vlan1
 no ip address
!
interface Vlan2
 ip address 1.1.1.3 255.255.255.0
!
```

CHAPTER 15

Configuring Ethernet OAM, CFM, and E-LMI

Ethernet Operations, Administration, and Maintenance (OAM) is a protocol for installing, monitoring, and troubleshooting Ethernet networks to increase management capability within the context of the overall Ethernet infrastructure. The Cisco MWR 2941 router supports IEEE 802.1ag Connectivity Fault Management (CFM), Ethernet Local Management Interface (E-LMI), and IEEE 802.3ah Ethernet OAM discovery, link monitoring, remote fault detection, and remote loopback. It also supports IP Service Level Agreements (SLAs) for CFM, and ITU-T Y.1731 fault management.

This chapter provides information about configuring CFM, E-LMI, and the Ethernet OAM protocol. It defines the differences between the ratified CFM 802.1ag standard (draft 8.1) and the previous version, Cisco IOS (draft 1.0). It also includes configuration information for CFM ITU-TY.1731 fault management support in this release.

Note Release 15.0(1)MR does not support the draft 1.0 version of CFM.

For complete command and configuration information for Ethernet OAM,CFM, E-LMI, and Y.1731, see the *Cisco IOS Carrier Ethernet Configuration Guide* at this URL:
http://www.cisco.com/en/US/docs/ios/cether/configuration/guide/12_2sr/ce_12_2sr_book.html

For complete syntax of the commands used in this chapter, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR* and the *Cisco IOS Carrier Ethernet Command Reference* at this URL:
http://www.cisco.com/en/US/docs/ios/cether/command/reference/ce_book.html

The Cisco MWR 2941 does not necessarily support all of the commands listed in the Cisco IOS Carrier Ethernet documentation.

This chapter contains these sections:

Understanding Ethernet CFM

Ethernet CFM is an end-to-end per-service-instance (per VLAN) Ethernet layer OAM protocol that includes proactive connectivity monitoring, fault verification, and fault isolation. End-to-end can be provider-edge-to-provider-edge (PE-to-PE) device or customer-edge-to-customer-edge (CE-to-CE) device. Ethernet CFM, as specified by IEEE 802.1ag, is the standard for Layer 2 ping, Layer 2 traceroute, and end-to-end connectivity check of the Ethernet network.

These sections contain conceptual information about Ethernet CFM:

CFM Domain

A CFM maintenance domain is a management space on a network that is owned and operated by a single entity and defined by a set of ports internal to it, but at its boundary. You assign a unique maintenance level (from 0 to 7) to define the hierarchical relationship between domains. The larger the domain, the higher the level. For example, as shown in Figure 15-1, a service-provider domain would be larger than an operator domain and might have a maintenance level of 6, while the operator domain maintenance level is 3 or 4.

As shown in Figure 15-2, domains cannot intersect or overlap because that would require management by more than one entity, which is not allowed. Domains can touch or nest (if the outer domain has a higher maintenance level than the nested domain). Nesting domains is useful when a service provider contract with one or more operators to provide Ethernet service. Each operator has its own maintenance domain and the service provider domain is a superset of the operator domains. Maintenance levels of nesting domains should be communicated among the administrating organizations. CFM exchanges messages and performs operations on a per-domain basis.

Figure 15-1 CFM Maintenance Domains

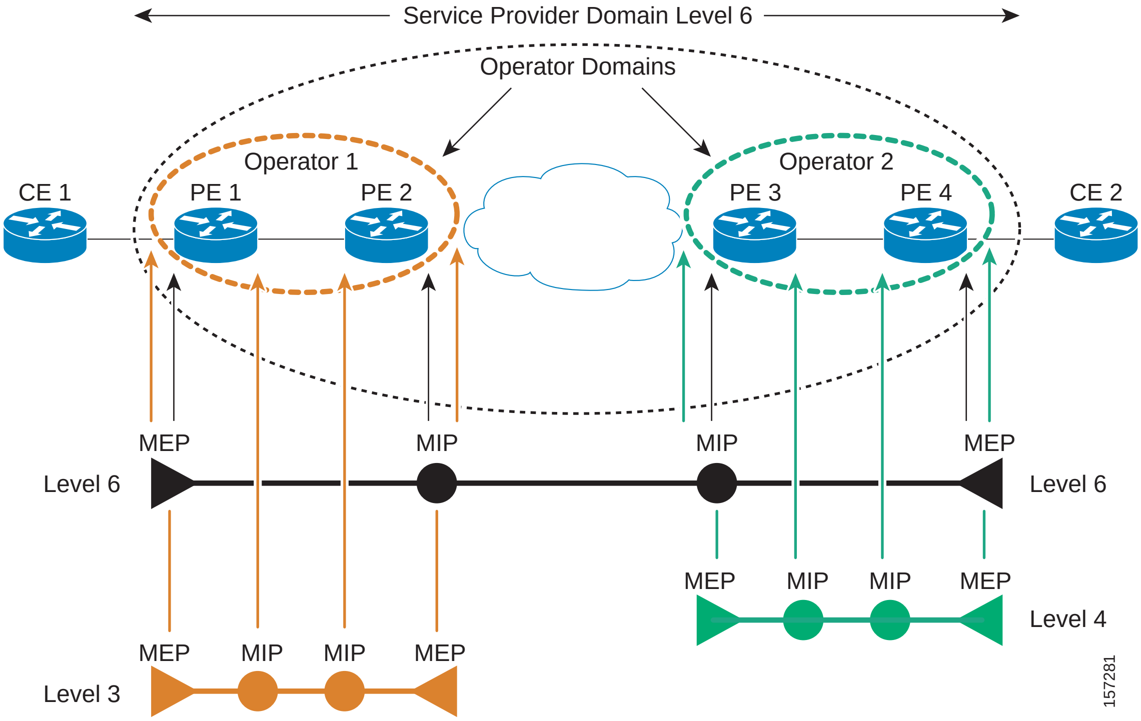
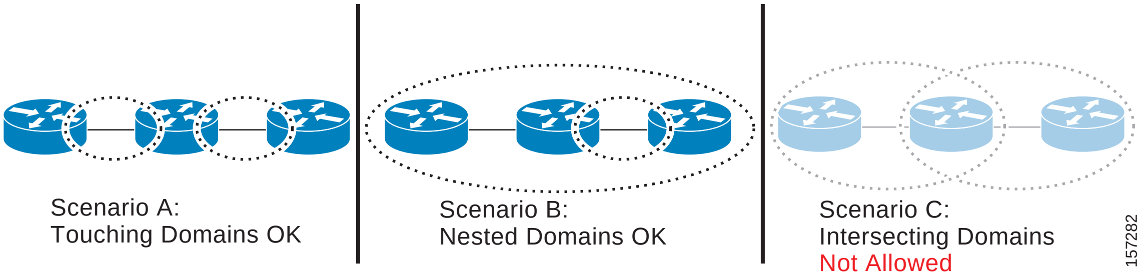


Figure 15-2 Allowed Domain Relationships



Maintenance Associations and Maintenance Points

A maintenance association (MA) identifies a service that can be uniquely identified within the maintenance domain. The CFM protocol runs within a maintenance association.A maintenance point is a demarcation point on an interface that participates in CFM within a maintenance domain. Maintenance points drop all lower-level frames and forward all higher-level frames. There are two types of maintenance points:

- Maintenance end points (MEPs) are points at the edge of the domain that define the boundaries and confine CFM messages within these boundaries. *Outward facing* or *Down* MEPs communicate through the wire side (connected to the port). *Inward facing* or *Up* MEPs communicate through the relay function side, not the wire side.

Note CFM draft 1 referred to inward and outward-facing MEPs. CFM draft 8.1 refers to up and down MEPs, respectively. This document uses the CFM 8.1 terminology for direction.

CFM draft 1 supported only up MEPs on a per-port or per-VLAN basis. CFM 802.1ag supports up and down per-VLAN MEPs, as well as port MEPs, which are untagged down MEPs that are not associated with a VLAN. Port MEPs are configured to protect a single hop and used to monitor link state through CFM. If a port MEP is not receiving continuity check messages from its peer (static remote MEP), for a specified interval, the port is put into an operational down state in which only CFM and OAM packets pass through, and all other data and control packets are dropped.

- An up MEP sends and receives CFM frames through the relay function. It drops all CFM frames at its level or lower that come from the wire side, except traffic going to the down MEP. For CFM frames from the relay side, it processes the frames at its level and drops frames at a lower level. The MEP transparently forwards all CFM frames at a higher level, regardless of whether they are received from the relay or wire side. If the port on which MEP is configured is blocked by STP, the MEP can still send or receive CFM messages through the relay function. CFM runs at the provider maintenance level (UPE-to-UPE), specifically with up MEPs at the user network interface (UNI).
- A down MEP sends and receives CFM frames through the wire connected to the port on which the MEP is configured. It drops all CFM frames at its level or lower that come from the relay side. For CFM frames from the wire side, it processes all CFM frames at its level and drops CFM frames at lower levels except traffic going to the other lower-level down MEP. The MEP transparently forwards all CFM frames at a higher level, regardless of whether they are received from the relay or through the wire.

- Maintenance intermediate points (MIPs) are internal to a domain, not at the boundary, and respond to CFM only when triggered by traceroute and loopback messages. They forward CFM frames received from MEPs and other MIPs, drop all CFM frames at a lower level (unless MIP filtering is enabled), and forward all CFM frames at a higher level and at a lower level and regardless of whether they are received from the relay or wire side. When MIP filtering is enabled, the MIP drops CFM frames at a lower level. MIPs also catalog and forward continuity check messages (CCMs), but do not respond to them.

  In the first draft of CFM, MIP filtering was always enabled. In draft 8.1, MIP filtering is disabled by default, and you can configure it to be enabled or disabled. When MIP filtering is disabled, all CFM frames are forwarded.

  You can manually configure a MIP or configure the router to automatically create a MIP. You can configure a MEP without a MIP. In case of a configuration conflict, manually created MIPs take precedence over automatically created MIPs.

  If port on which the MEP is configured is blocked by Spanning-Tree Protocol (STP), the MIP can receive and might respond to CFM messages from both the wire and relay side, but cannot forward any CFM messages. This differs from CFM draft 1, where STP blocked ports could not send or receive CFM messages.

CFM Messages

CFM uses standard Ethernet frames distinguished by EtherType or (for multicast messages) by MAC address. All CFM messages are confined to a maintenance domain and to a service-provider VLAN (S-VLAN). These CFM messages are supported:

- Continuity Check (CC) messages—multicast heartbeat messages exchanged periodically between MEPs that allow MEPs to discover other MEPs within a domain and allow MIPs to discover MEPs. CC messages are configured to a domain or VLAN. Enter the **continuity-check** Ethernet service configuration command to enable CCM.

  The default continuity check message (CCM) interval on the router is 10 seconds. You can set it to be 100 ms, 1 second, 1 minute, or 10 minutes by entering the **continuity-check interval** Ethernet service mode command. Because faster CCM rates are more CPU intensive, we do not recommend configuring a large number of MEPs running at 100 ms intervals.
- Loopback messages—unicast or multicast frames transmitted by a MEP at administrator request to verify connectivity to a particular maintenance point, indicating if a destination is reachable. A loopback message is similar to an Internet Control Message Protocol (ICMP) ping message. Refer to the **ping ethernet** privileged EXEC command.
- Traceroute messages—multicast frames transmitted by a MEP at administrator request to track the path (hop-by-hop) to a destination MEP. Traceroute messages are similar in concept to UDP traceroute messages. Refer to the **traceroute ethernet** privileged EXEC command.

Crosscheck Function and Static Remote MEPs

The crosscheck function is a timer-driven post-provisioning service verification between dynamically configured MEPs (using crosscheck messages) and expected MEPs (by configuration) for a service. It verifies that all endpoints of a multipoint service are operational. The crosscheck function is performed only one time and is initiated from the command-line interface (CLI).

CFM 802.1ag also supports static remote MEPs or static RMEP check. Unlike the crosscheck function, which is performed only once, configured static RMEP checks run continuously. To configure static RMEP check, enter the **continuity-check static rmep** Ethernet CFM service mode command.

SNMP Traps and Fault Alarms

The MEPs generate two types of SNMP traps: CC traps and crosscheck traps. Supported CC traps are MEP up, MEP down, cross-connect (a service ID does not match the VLAN), loop, and configuration error. The crosscheck traps are service up, MEP missing (an expected MEP is down), and unknown MEP.

Fault alarms are unsolicited notifications sent to alert the system administrator when CFM detects a fault. In CFM draft 1, fault alarms were sent instantaneously when detected. In CFM 802.1ag, you can configure the priority level of alarms that trigger an SNMP trap or syslog message. You can also configure a delay period before a fault alarm is sent and the time before the alarm is reset.

Configuration Error List

CFM configuration errors in CFM 802.1ag can be misconfigurations or extra configuration commands detected during MEP configuration. They can be caused by overlapping maintenance associations. For example, if you create a maintenance association with a VLAN list and a MEP on an interface, a potential leak error could occur if other maintenance associations associated with the same VLAN exist at a higher level without any MEPs configured. You can display the configuration error list, which is informational only, by entering the **show ethernet cfm errors configuration** privileged EXEC command.

CFM Version Interoperability

When customers upgrade their network from the Cisco CFM draft 1 to IEEE standardized 802.1ag CFM, they might not upgrade all equipment at the same time, which could result in a mix of Cisco CFM draft 1 and IEEE standardized CFM devices in the network. CFM areas are regions in a network running Cisco CFM draft 1 software. Internal area bridges are all Cisco devices running CFM draft 1, and external area bridges are devices (Cisco or third-party devices) running IEEE standardized 802.1ag CFM.

Devices at the edge of these areas perform message translation. Translation is not needed for maintenance domains that do not span different areas (that is, where CFM messages end on a port on the device) since the port can respond in the same message format as was received. However, for maintenance domains that span across two areas, the device must translate the CFM message appropriately before sending it on to the other area.

Note The Cisco MWR 2941 does not support translation between CFM draft 1.0 and IEEE standardized 802.1ag CFM.

When designing a network with CFM areas, follow these guidelines:

- Whenever possible, group devices with the same CFM version together.
- Minimize the number of boundaries between CFM clusters, minimizing the number of devices that must perform translation.
- Never mix CFM versions on a single segment.

IP SLAs Support for CFM

The router supports CFM with IP Service Level Agreements (SLAs), which provides the ability to gather Ethernet layer network performance metrics. Available statistical measurements for the IP SLAs CFM operation include round-trip time, jitter (interpacket delay variance), and packet loss. You can schedule multiple IP SLAs operations and use Simple Network Management Protocol (SNMP) trap notifications and syslog messages for proactive threshold violation monitoring.

For more information about IP SLAs, see Chapter 41, “Configuring Cisco IOS IP SLAs Operations.”

IP SLAs integration with CFM gathers Ethernet layer statistical measurements by sending and receiving Ethernet data frames between CFM MEPs. Performance is measured between the source MEP and the destination MEP. Unlike other IP SLAs operations that provide performance metrics for only the IP layer, IP SLAs with CFM provides performance metrics for Layer 2.

You can manually configure individual Ethernet ping or jitter operations. You can also configure an IP SLAs automatic Ethernet operation that queries the CFM database for all MEPs in a given maintenance domain and VLAN. The operation then automatically creates individual Ethernet ping or jitter operations based on the discovered MEPs.

Because IP SLAs is a Cisco proprietary feature, interoperability between CFM draft 1 and CFM 802.1ag is handled automatically by the router.

For more information about IP SLAs operation with CFM, see the *IP SLAs for Metro-Ethernet* feature module at this URL:

http://www.cisco.com/en/US/products/ps6922/products_feature_guide09186a00807d72f5.html

Configuring Ethernet CFM

Configuring Ethernet CFM requires configuring the CFM domain. You can optionally configure and enable other CFM features such as crosschecking, remote MEP, port MEPs, SNMP traps, and fault alarms. Note that some of the configuration commands and procedures differ from those used in CFM draft 1.

Default Ethernet CFM Configuration

CFM is globally disabled.

CFM is enabled on all interfaces when CFM is globally enabled.

A port can be configured as a flow point (MIP/MEP), a transparent port, or disabled (CFM disabled). By default, ports are transparent ports until configured as MEP, MIP, or disabled.

There are no MEPs or MIPs configured.

When configuring a MEP, if you do not configure direction, the default is up (inward facing).

Ethernet CFM Configuration Guidelines

- EtherChannels are not supported.
- CFM is not supported on and cannot be configured on routed ports.
- You cannot configure CFM on VLAN interfaces.
- CFM is supported on trunk ports and access ports with these exceptions:

- – Trunk ports configured as MEPs must belong to allowed VLANs
- – Access ports configured as MEPs must belong to the native VLAN.
- CFM is not supported on 802.1Q tunnel interfaces.
- You cannot configure CFM on an EoMPLS port.
- A REP port or FlexLink port can also be a service (VLAN) MEP or MIP, but it cannot be a port MEP.
- CFM is supported on ports running STP.
- You must configure a port MEP at a lower level than any service (VLAN) MEPs on an interface.

Configuring the CFM Domain

Beginning in privileged EXEC mode, follow these steps to configure the Ethernet CFM domain, configure a service to connect the domain to a VLAN, or configure a port to act as a MEP. You can also enter the optional commands to configure other parameters, such as continuity checks.

Note You do not need to enter the **ethernet cfm ieee** global configuration command to configure the CFM version as IEEE 802.1ag; the CFM version is always 802.1ag and the command is automatically generated when you enable CFM.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ethernet cfm global**	Globally enable Ethernet CFM on the router.
Step 3	**ethernet cfm traceroute cache** [**size** *entries* | **hold-time** *minutes*]	(Optional) Configure the CFM traceroute cache. You can set a maximum cache size or hold time. • (Optional) For **size**, enter the cache size in number of entry lines. The range is from 1 to 4095; the default is 100 lines. • (Optional) For **hold-time**, enter the maximum cache hold time in minutes. The range is from 1 to 65535; the default is 100 minutes.
Step 4	**ethernet cfm mip auto-create level** *level-id* **vlan** *vlan-id*	(Optional) Configure the router to automatically create MIPs for VLAN IDS that are not associated with specific maintenance associations at the specified level. The level range is 0 to 7. **Note** Configure MIP auto-creation only for VLANs that MIPs should monitor. Configuring for all VLANs can be CPU and memory-intensive.
Step 5	**ethernet cfm mip filter**	(Optional) Enable MIP filtering, which means that all CFM frames at a lower level are dropped. The default is disabled.
Step 6	**ethernet cfm domain** *domain-name* **level** *level-id*	Define a CFM domain, set the domain level, and enter ethernet-cfm configuration mode for the domain. The maintenance level number range is 0 to 7.

	Command	Purpose
Step 7	**id** {*mac-address domain_number* | **dns** *name* | **null**}	(Optional) Assign a maintenance domain identifier. • *mac-address domain_number*—Enter the MAC address and a domain number. The number can be from 0 to 65535. • **dns** *name*—Enter a DNS name string. The name can be a maximum of 43 characters. • **null**—Assign no domain name.
Step 8	**service** {*ma-name* | *ma-number* | *vpn-id vpn*} {**vlan** *vlan-id* [**direction down**] | **port**}	Define a customer service maintenance association (MA) name or number or VPN ID to be associated with the domain, a VLAN ID or port MEP, and enter ethernet-cfm-service configuration mode. • *ma-name*—a string of no more than 100 characters that identifies the MAID. • *ma-number*—a value from 0 to 65535. • *vpn-id vpn*—enter a VPN ID as the *ma-name*. • **vlan** *vlan-id*—VLAN range is from 1 to 4094. You cannot use the same VLAN ID for more than one domain at the same level. • (Optional) **direction down**—specify the service direction as down. • **port**—Configure port MEP, a down MEP that is untagged and not associated with a VLAN.
Step 9	**continuity-check**	Enable sending and receiving of continuity check messages.
Step 10	**continuity-check interval** *value*	(Optional) Set the interval at which continuity check messages are sent. The available values are 100 ms, 1 second, 10 seconds, 1 minute and 10 minutes. The default is 10 seconds. **Note** Because faster CCM rates are more CPU-intensive, we do not recommend configuring a large number of MEPs running at 100 ms intervals.
Step 11	**continuity-check loss-threshold** *threshold-value*	(Optional) Set the number of continuity check messages to be missed before declaring that an MEP is down. The range is 2 to 255; the default is 3.
Step 12	**maximum meps** *value*	(Optional) Configure the maximum number of MEPs allowed across the network. The range is from 1 to 65535. The default is 100.
Step 13	**sender-id** {**chassis** | **none**}	(Optional) Include the sender ID TLVs, attributes containing type, length, and values for neighbor devices. • **chassis**—Send the chassis ID (host name). • **none**—Do not include information in the sender ID.

	Command	Purpose
Step 14	**mip auto-create** [**lower-mep-only** | **none**]	(Optional) Configure auto creation of MIPs for the service. • **lower-mep-only**—Create a MIP only if there is a MEP for the service in another domain at the next lower active level. • **none** —No MIP auto-create.
Step 15	**exit**	Return to ethernet-cfm configuration mode.
Step 16	**mip auto-create** [**lower-mep-only**]	(Optional) Configure auto creation of MIPs for the domain. • **lower-mep-only**—Create a MIP only if there is a MEP for the service in another domain at the next lower active level.
Step 17	**mep archive-hold-time** *minutes*	(Optional) Set the number of minutes that data from a missing maintenance end point is kept before it is purged. The range is 1 to 65535; the default is 100 minutes.
Step 18	**exit**	Return to global configuration mode.
Step 19	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode.
Step 20	**switchport mode trunk**	(Optional) Configure the port as a trunk port.
Step 21	**ethernet cfm mip level** *level-id*	(Optional) Configure a customer level or service-provider level maintenance intermediate point (MIP) for the interface. The MIP level range is 0 to 7. **Note** This step is not required if you have entered the **ethernet cfm mip auto-create** global configuration command or the **mip auto-create** ethernet-cfm or ethernet-cfm-srv configuration mode.
Step 22	**ethernet cfm mep domain** *domain-name* **mpid** *identifier* {**vlan** *vlan-id* | **port**}	Configure maintenance end points for the domain, and enter Ethernet cfm mep mode. • **domain** *domain-name*—Specify the name of the created domain. • **mpid** *identifier*—Enter a maintenance end point identifier. The identifier must be unique for each VLAN (service instance). The range is 1 to 8191. • **vlan** *vlan-id*—Enter the service provider VLAN ID or IDs as a VLAN-ID (1 to 4094), a range of VLAN-IDs separated by a hyphen, or a series of VLAN IDs separated by comma. • **port**—Configure port MEP.
Step 23	**cos** *value*	(Optional) Specify the class of service (CoS) value to be sent with the messages. The range is 0 to 7.
Step 24	**end**	Return to privileged EXEC mode.
Step 25	**show ethernet cfm maintenance-points** {**local** | **remote**}	Verify the configuration.

	Command	Purpose
Step 26	**show ethernet cfm errors [configuration]**	(Optional) Display the configuration error list.
Step 27	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** versions of the commands to remove the configuration or return to the default configurations.

This is an example of the basic CFM configuration:

```
Router(config)# ethernet cfm ieee
Router(config)# ethernet cfm global
Router(config)# ethernet cfm domain abc level 3
Router(config-ecfm)# service test vlan 5
Router(config-ecfm-srv)# continuity-check
Router(config-ecfm-srv)# exit
Router(config-ecfm)# exit
Router(config)# interface gigabitethernet1/0/2
Router(config-if)# ethernet cfm mep domain abc mpid 222 vlan 5
Router(config-if-ecfm-mep)# exit
```

Configuring Ethernet CFM Crosscheck

Beginning in privileged EXEC mode, follow these steps to configure Ethernet CFM crosscheck:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ethernet cfm mep crosscheck start-delay** *delay*	Configure the number of seconds that the device waits for remote MEPs to come up before the crosscheck is started. The range is 1 to 65535; the default is 30 seconds.
Step 3	**ethernet cfm domain** *domain-name* **level** *level-id*	Define a CFM domain, set the domain level, and enter ethernet-cfm configuration mode for the domain. The maintenance level number range is 0 to 7.
Step 4	**service** {*ma-name* | *ma-number* | *vpn-id vpn*} {**vlan** *vlan-id*}	Define a customer service maintenance association name or number or VPN ID to be associated with the domain, and a VLAN ID, and enter ethernet-cfm-service configuration mode. • *ma-name*—a string of no more than 100 characters that identifies the MAID. • *ma-number*—a value from 0 to 65535. • *vpn-id vpn*—enter a VPN ID as the *ma-name*. • **vlan** *vlan-id*—VLAN range is from 1 to 4094. You cannot use the same VLAN ID for more than one domain at the same level.
Step 5	**mep mpid** *identifier*	Define the MEP maintenance end point identifier in the domain and service. The range is 1 to 8191
Step 6	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 7	**ethernet cfm mep crosscheck** {**enable** | **disable**} **domain** *domain-name* {**vlan** {*vlan-id* | **any**} | **port**}	Enable or disable CFM crosscheck for one or more VLANs or a port MEP in the domain. • **domain** *domain-name*—Specify the name of the created domain. • **vlan** {*vlan-id* | **any**}—Enter the service provider VLAN ID or IDs as a VLAN-ID (1 to 4094), a range of VLAN-IDs separated by a hyphen, or a series of VLAN IDs separated by comma. Enter **any** for any VLAN. • **port**—Identify a port MEP.
Step 8	**show ethernet cfm maintenance-points remote crosscheck**	Verify the configuration.
Step 9	**show ethernet cfm errors** [**configuration**]	Enter this command after you enable CFM crosscheck to display the results of the crosscheck operation. Enter the **configuration** keyword to display the configuration error list.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of each command to remove a configuration or to return to the default settings.

Configuring Static Remote MEP

Beginning in privileged EXEC mode, follow these steps to configure Ethernet CFM static remote MEP:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ethernet cfm domain** *domain-name* **level** *level-id*	Define a CFM domain, set the domain level, and enter ethernet-cfm configuration mode for the domain. The maintenance level number range is 0 to 7.

	Command	Purpose
Step 3	**service** {*ma-name* | *ma-num* | **vlan-id** *vlan-id* | **vpn-id** *vpn-id*} [**port** | **vlan** *vlan-id* [**direction down**]]	Define a customer service maintenance association name or number or a VPN ID to be associated with the domain, and a VLAN ID or peer MEP, and enter ethernet-cfm-service configuration mode. • *ma-name*—a string of no more than 100 characters that identifies the MAID. • *ma-number*—a value from 0 to 65535. • *vpn-id*—enter a VPN ID as the *ma-name*. **Note** The **vpn-id** keyword is not supported. • **vlan** *vlan-id*—VLAN range is from 1 to 4094. You cannot use the same VLAN ID for more than one domain at the same level. • (Optional) **direction down**—specify the service direction as down. • **port**—Configure port MEP, a down MEP that is untagged and not associated with a VLAN.
Step 4	**continuity-check**	Enable sending and receiving of continuity check messages.
Step 5	**mep mpid** *identifier*	Define the static remote maintenance end point identifier. The range is 1 to 8191
Step 6	**continuity-check static rmep**	Enable checking of the incoming continuity check message from a remote MEP that is configured in the MEP list.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show ethernet cfm maintenance-points remote static**	Verify the configuration.
Step 9	**show ethernet cfm errors** [**configuration**]	Enter this command after you enable CFM crosscheck to display the results of the crosscheck operation. Enter the **configuration** keyword to display the configuration error list.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of each command to remove a configuration or to return to the default settings.

Configuring a Port MEP

A port MEP is a down MEP that is not associated with a VLAN and that uses untagged frames to carry CFM messages. You configure port MEPs on two connected interfaces. Port MEPs are always configured at a lower domain level than native VLAN MEPs.

Beginning in privileged EXEC mode, follow these steps to configure Ethernet CFM port MEPs:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ethernet cfm domain** *domain-name* **level** *level-id*	Define a CFM domain, set the domain level, and enter ethernet-cfm configuration mode for the domain. The maintenance level number range is 0 to 7.
Step 3	**service** {*ma-name* | *ma-number* | *vpn-id*} **port**	Define a customer service maintenance association name or number or VPN ID to be associated with the domain, define a port MEP, and enter ethernet-cfm-service configuration mode. • *ma-name*—a string of no more than 100 characters that identifies the MAID. • *ma-number*—a value from 0 to 65535. • *vpn-id vpn*—enter a VPN ID as the *ma-name*.
Step 4	**mep mpid** *identifier*	Define the static remote maintenance end point identifier in the domain and service. The range is 1 to 8191
Step 5	**continuity-check**	Enable sending and receiving of continuity check messages.
Step 6	**continuity-check interval** *value*	(Optional) Set the interval at which continuity check messages are sent. The available values are 100 ms, 1 second, 10 seconds, 1 minute and 10 minutes. The default is 10 seconds. **Note** Because faster CCM rates are more CPU-intensive, we do not recommend configuring a large number of MEPs running at 100 ms intervals.
Step 7	**continuity-check loss-threshold** *threshold-value*	(Optional) Set the number of continuity check messages to be missed before declaring that an MEP is down. The range is 2 to 255; the default is 3.
Step 8	**continuity-check static rmep**	Enable checking of the incoming continuity check message from a remote MEP that is configured in the MEP list.
Step 9	**exit**	Return to ethernet-cfm configuration mode.
Step 10	**exit**	Return to global configuration mode.
Step 11	**interface** *interface-id*	Identify the port MEP interface and enter interface configuration mode.

	Command	Purpose
Step 12	**ethernet cfm mep domain** *domain-name* **mpid** *identifier* **port**	Configure the interface as a port MEP for the domain. • **domain** *domain-name*—Specify the name of the created domain. • **mpid** *identifier*—Enter a maintenance end point identifier. The identifier must be unique for each VLAN (service instance). The range is 1 to 8191.
Step 13	**end**	Return to privileged EXEC mode.
Step 14	**show ethernet cfm maintenance-points remote static**	Verify the configuration.
Step 15	**show ethernet cfm errors** [**configuration**]	Enter this command after you enable CFM crosscheck to display the results of the crosscheck operation. Enter the **configuration** keyword to display the configuration error list.
Step 16	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of each command to remove a configuration or to return to the default settings.

This is a sample configuration for a port MEP:

```
Router(config)# ethernet cfm domain abc level 3
Router(config-ecfm)# service PORTMEP port
Router(config-ecfm-srv)# mep mpid 222
Router(config-ecfm-srv)# continuity-check
Router(config-ecfm-srv)# continuity-check static rmep
Router(config-ecfm-srv)# exit
Router(config-ecfm)# exit
Router(config)# interface gigabitethernet1/0/1
Router(config-if)# ethernet cfm mep domain abc mpid 111 port
Router(config-if)# end
```

Configuring SNMP Traps

Beginning in privileged EXEC mode, follow these steps to configure traps for Ethernet CFM:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**snmp-server enable traps ethernet cfm cc** [**mep-up**] [**mep-down**] [**config**] [**loop**] [**cross-connect**]	(Optional) Enable Ethernet CFM continuity check traps.
Step 3	**snmp-server enable traps ethernet cfm crosscheck** [**mep-unknown**] [**mep-missing**] [**service-up**]	(Optional) Enable Ethernet CFM crosscheck traps.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of each command to remove a configuration or to return to the default settings.

Configuring Fault Alarms

Beginning in privileged EXEC mode, follow these steps to configure Ethernet CFM fault alarms. Note that you can configure fault alarms in either global configuration mode or Ethernet CFM interface MEP mode. In case of conflict, the interface MEP mode configuration takes precedence.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ethernet cfm alarm notification** {**all** | **error-xcon** | **mac-remote-error-xcon** | **none** | **remote-error-xcon** | **xcon**}	Globally enable Ethernet CFM fault alarm notification for the specified defects: • **all**—report all defects. • **error-xcon**—Report only error and connection defects. • **mac-remote-error-xcon**—Report only MAC-address, remote, error, and connection defects. • **none**—Report no defects. • **remote-error-xcon**—Report only remote, error, and connection defects. • **xcon**—Report only connection defects.
Step 3	**ethernet cfm alarm delay** *value*	(Optional) Set a delay period before a CFM fault alarm is sent. The range is 2500 to 10000 milliseconds (ms). The default is 2500 ms.
Step 4	**ethernet cfm alarm reset** *value*	(Optional) Specify the time period before the CFM fault alarm is reset. The range is 2500 to 10000 milliseconds (ms). The default is 10000 ms.
Step 5	**ethernet cfm logging alarm ieee**	Configure the router to generate system logging messages for the alarms.
Step 6	**interface** *interface-id*	(Optional) Specify an interface to configure, and enter interface configuration mode.
Step 7	**ethernet cfm mep domain** *domain-name* **mpid** *identifier* **vlan** *vlan-id*	Configure maintenance end points for the domain, and enter ethernet cfm interface mep mode. • **domain** *domain-name*—Specify the name of the created domain. • **mpid** *identifier*—Enter a maintenance end point identifier. The identifier must be unique for each VLAN (service instance). The range is 1 to 8191. • **vlan** *vlan-id*—Enter the service provider VLAN ID or IDs as a VLAN-ID (1 to 4094), a range of VLAN-IDs separated by a hyphen, or a series of VLAN IDs separated by comma.

	Command	Purpose
Step 8	**ethernet cfm alarm notification** {**all** | **error-xcon** | **mac-remote-error-xcon** | **none** | **remote-error-xcon** | **xcon**}	(Optional) Enable Ethernet CFM fault alarm notification for the specified defects on the interface. **Note** The Ethernet CFM interface MEP alarm configuration takes precedence over the global configuration.
Step 9	**ethernet cfm alarm** {**delay** *value* | **reset** *value*}	(Optional) Set an alarm delay period or a reset period. **Note** The Ethernet CFM interface MEP alarm configuration takes precedence over the global configuration.
Step 10	**end**	Return to privileged EXEC mode.
Step 11	**show running-config**	Verify your entries.
Step 12	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of each command to remove a configuration or to return to the default settings.

Configuring IP SLAs CFM Operation

You can manually configure an individual IP SLAs Ethernet ping or jitter echo operation or you can configure IP SLAs Ethernet operation with endpoint discovery. You can also configure multiple operation scheduling. For accurate one-way delay statistics, the clocks on the endpoint switches must be synchronized. You can configure the endpoint switches with Network Time Protocol (NTP) so that the switches are synchronized to the same clock source.

Note You cannot enable Precision Timing Protocol (PTP) if you enable NTP. For more information about PTP, see Configuring Clocking and Timing.

For more information about configuring IP SLAs Ethernet operation, see the IP SLAs Configuration Guide, Cisco IOS Release 15.0S. For detailed information about commands for IP SLAs, see the Cisco IOS IP SLAs Command Reference.

Note The Cisco MWR 2941 does not necessarily support all of the commands listed in the Cisco IOS IP SLA documentation.

This section includes these procedures:

- Manually Configuring an IP SLAs CFM Probe or Jitter Operation, page 15-18
- Configuring an IP SLAs Operation with Endpoint Discovery, page 15-19

Manually Configuring an IP SLAs CFM Probe or Jitter Operation

Beginning in privileged EXEC mode, follow these steps to manually configure an IP SLAs Ethernet echo (ping) or jitter operation:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip sla** *operation-number*	Create an IP SLAs operation, and enter IP SLAs configuration mode.
Step 3	**ethernet echo mpid** *identifier* **domain** *domain-name* **vlan** *vlan-id* or **ethernet jitter mpid** *identifier* **domain** *domain-name* **vlan** *vlan-id* [**interval** *interpacket-interval*] [**num-frames** *number-of frames transmitted*]	Configure the IP SLAs operation as an echo (ping) or jitter operation, and enter IP SLAs Ethernet echo configuration mode. • Enter **echo** for a ping operation or **jitter** for a jitter operation. • For **mpid** *identifier*, enter a maintenance endpoint identifier. The identifier must be unique for each VLAN (service instance). The range is 1 to 8191. • For **domain** *domain-name*, enter the CFM domain name. • For **vlan** *vlan-id*, the VLAN range is from 1 to 4095. • (Optional—for jitter only) Enter the **interval** between sending of jitter packets. • (Optional—for jitter only) Enter the **num-frames** and the number of frames to be sent.
Step 4	**cos** *cos-value*	(Optional) Set a class of service value for the operation.
Step 5	**frequency** *seconds*	(Optional) Set the rate at which the IP SLAs operation repeats. The range is from 1 to 604800 seconds; the default is 60 seconds.
Step 6	**history** *history-parameter*	(Optional) Specify parameters for gathering statistical history information for the IP SLAs operation.
Step 7	**owner** *owner-id*	(Optional) Configure the SNMP owner of the IP SLAs operation.
Step 8	**request-data-size** *bytes*	(Optional) Specify the protocol data size for an IP SLAs request packet. The range is from 0 to the maximum size allowed by the protocol being used; the default is 66 bytes.
Step 9	**tag** *text*	(Optional) Create a user-specified identifier for an IP SLAs operation.
Step 10	**threshold** *milliseconds*	(Optional) Specify the upper threshold value in milliseconds (ms0 for calculating network monitoring statistics. The range is 0 to 2147483647; the default is 5000.
Step 11	**timeout** *milliseconds*	(Optional) Specify the amount of time in ms that the IP SLAs operation waits for a response from its request packet. The range is 0 to 604800000; the default value is 5000.

	Command	Purpose
Step 12	**exit**	Return to global configuration mode.
Step 13	**ip sla schedule** *operation-number* [**ageout** *seconds*] [**life** {**forever** | *seconds*}] [**recurring**] [**start-time** {*hh:mm {:ss}* [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*}]	Schedule the time parameters for the IP SLAs operation. • *operation-number*—Enter the IP SLAs operation number. • (Optional) **ageout** *seconds*—Enter the number of seconds to keep the operation in memory when it is not actively collecting information. The range is 0 to 2073600 seconds. The default is 0 seconds. • (Optional) **life**—Set the operation to run indefinitely (**forever**) or for a specific number of *seconds*. The range is from 0 to 2147483647. The default is 3600 seconds (1 hour) • (Optional) **recurring**—Set the probe to be automatically scheduled every day. • (Optional) **start-time**—Enter the time for the operation to begin collecting information: – To start at a specific time, enter the hour, minute, second (in 24-hour notation), and day of the month. – Enter **pending** to select no information collection until a start time is selected. – Enter **now** to start the operation immediately. – Enter **after** *hh:mm:ss* to show that the operation should start after the entered time has elapsed.
Step 14	**end**	Return to privileged EXEC mode.
Step 15	**show ip sla configuration** [*operation-number*]	Show the configured IP SLAs operation.
Step 16	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove an IP SLAs operation, enter the no **ip sla** *operation-number* global configuration command.

Configuring an IP SLAs Operation with Endpoint Discovery

Beginning in privileged EXEC mode, follow these steps to use IP SLAs to automatically discover the CFM endpoints for a domain and VLAN ID. You can configure ping or jitter operations to the discovered endpoints.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip sla ethernet-monitor** *operation-number*	Begin configuration of an IP SLAs automatic Ethernet operation, and enter IP SLAs Ethernet monitor configuration mode.

	Command	Purpose
Step 3	**type echo domain** *domain-name* **vlan** *vlan-id* [**exclude-mpids** *mp-ids*] or **type jitter domain** *domain-name* **vlan** *vlan-id* [**exclude-mpids** *mp-ids*] [**interval** *interpacket-interval*] [**num-frames** *number-of frames transmitted*]	Configure the automatic Ethernet operation to create echo (ping) or jitter operation and enter IP SLAs Ethernet echo configuration mode. • Enter **type echo** for a ping operation or **type jitter** for a jitter operation. • For **mpid** *identifier*, enter a maintenance endpoint identifier. The range is 1 to 8191. • For **domain** *domain-name*, enter the CFM domain name. • For **vlan** *vlan-id*, the VLAN range is from 1 to 4095. • (Optional) Enter **exclude-mpids** *mp-ids* to exclude the specified maintenance endpoint identifiers. • (Optional—for jitter only) Enter the **interval** between sending of jitter packets. • (Optional—for jitter only) Enter the **num-frames** and the number of frames to be sent.
Step 4	**cos** *cos-value*	(Optional) Set a class of service value for the operation. Before configuring the **cos** parameter, you must globally enable QoS by entering the **mls qos** global configuration command.
Step 5	**owner** *owner-id*	(Optional) Configure the SNMP owner of the IP SLAs operation.
Step 6	**request-data-size** *bytes*	(Optional) Specify the protocol data size for an IP SLAs request packet. The range is from 0 to the maximum size allowed by the protocol being used; the default is 66 bytes.
Step 7	**tag** *text*	(Optional) Create a user-specified identifier for an IP SLAs operation.
Step 8	**threshold** *milliseconds*	(Optional) Specify the upper threshold value in milliseconds for calculating network monitoring statistics. The range is 0 to 2147483647; the default is 5000.
Step 9	**timeout** *milliseconds*	(Optional) Specify the amount of time in milliseconds that the IP SLAs operation waits for a response from its request packet. The range is 0 to 604800000; the default value is 5000.
Step 10	**exit**	Return to global configuration mode.

	Command	Purpose
Step 11	**ip sla schedule** *operation-number* [**ageout** *seconds*] [**life** {**forever** | *seconds*}] [**recurring**] [**start-time** {*hh:mm {:ss}* [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*}]	Schedule the time parameters for the IP SLAs operation. • *operation-number*—Enter the IP SLAs operation number. • (Optional) **ageout** *seconds*—Enter the number of seconds to keep the operation in memory when it is not actively collecting information. The range is 0 to 2073600 seconds. The default is 0 seconds. • (Optional) **life**—Set the operation to run indefinitely (**forever**) or for a specific number of *seconds*. The range is from 0 to 2147483647. The default is 3600 seconds (1 hour) • (Optional) **recurring**—Set the probe to be automatically scheduled every day. • (Optional) **start-time**—Enter the time for the operation to begin collecting information: – To start at a specific time, enter the hour, minute, second (in 24-hour notation), and day of the month. – Enter **pending** to select no information collection until a start time is selected. – Enter **now** to start the operation immediately. – Enter **after** *hh:mm:ss* to show that the operation should start after the entered time has elapsed.
Step 12	**end**	Return to privileged EXEC mode.
Step 13	**show ip sla configuration** [*operation-number*]	Show the configured IP SLAs operation.
Step 14	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove an IP SLAs operation, enter the **no ip sla** *operation-number* global configuration command.

Understanding CFM ITU-T Y.1731 Fault Management

The ITU-T Y.1731 feature provides new CFM functionality for fault and performance management for service providers in large network. The router supports Ethernet Alarm Indication Signal (ETH-AIS), Ethernet Remote Defect Indication (ETH-RDI), and Ethernet Multicast Loopback Message (MCAST-LBM) functionality for fault detection, verification, and isolation.

- Y.1731 Terminology, page 15-22
- Alarm Indication Signals, page 15-22
- Ethernet Remote Defect Indication, page 15-23
- Ethernet Locked Signal, page 15-23
- Multicast Ethernet Loopback, page 15-23

Y.1731 Terminology

- Server MEP—the combination of the server layer termination function and server or Ethernet adaptation layer termination function or server or Ethernet adaptation function, where the server layer termination function is expected to run OAM mechanisms specific to the server layer. The supported mechanisms are link up, link down, and 802.3ah.
- Server layer—a virtual MEP layer capable of detecting fault conditions.
- Defect conditions:
  - Loss of continuity (LOC): the MEP stopped receiving CCM frames from a peer MEP
  - Mismerge: the MEP received a CCM frame with a correct maintenance level (matching the MEP level) but an incorrect maintenance ID.
  - Unexpected MEP: the MEP received a CCM frame with the correct maintenance level (matching the MEP's level) and correct maintenance ID, but an unexpected MEP ID.
  - Unexpected maintenance level: the MEP received a CCM frame with an incorrect maintenance level.
  - Unexpected period: the MEP received a CCM frame with a correct maintenance level, a correct maintenance ID, a correct MEP ID, but a different transmission period field.
- Signal fail—the MEP declares a signal fail condition when it detects a defect condition.
- Alarm Indication Signal (AIS) condition—the MEP received an AIS frame.
- Remote Defect Indication (RDI) condition—The MEP received a CCM frame with the RDI field set.
- Locked Signal (LCK) condition—The MEP received an LCK frame.

Alarm Indication Signals

The Ethernet Alarm Signal function (ETH-AIS) is used to suppress alarms after defects are detected at the *server* (sub) layer, which is a virtual MEP layer capable of detecting fault conditions. A fault condition could be a signal fail condition, an AIS condition, or a LCK condition.

Note Although the configuration is allowed, you should not configure AIS in networks running STP. An STP configuration might cause AIS interruption or redirection.

When a MEP or a service MEP (SMEP) detects a connectivity fault at a specific maintenance association level, it multicasts AIS frames in the direction away from the detected failure at the client maintenance association level. The frequency of AIS frame transmission is based on the AIS transmission period. The first AIS frame is always sent immediately following the detection of the defect condition. We recommend a transition period of 1 second in a network of only a few VLANs to ensure that the first AIS frame is sent immediately following error detection. We recommend a 60-second interval in a network of multiple (up to 4094) VLANs to prevent stressing the network with 1-second transmissions.

A MEP that receives a frame with ETH-AIS information cannot determine the specific server with the defect condition or the set of peer MEPs for which it should suppress alarms. Therefore, it suppresses alarms for all peer MEPs, whether or not they are connected.

When a MEP receives an AIS frame, it examines it to be sure that the Maintenance Entity Group (MEG) level matches its own MEG and then detects the AIS default condition. (A MEG is Y.1731 terminology for maintenance association in 802.1ag.) After this detection, if no AIS frames are received for an interval of 3.5 times the AIS transmission period, the MEP clears the AIS defect condition. For example, if the AIS timer is set for 60 seconds, the AIS timer period expires after 3.5 times 60, or 210 seconds.

The AIS condition is terminated when a valid CCM is received with all error conditions cleared or when the AIS period timer expires (the default time is 60 seconds).

Ethernet Remote Defect Indication

When Ethernet OAM continuity check (ETH-CC) transmission is enabled, the Ethernet Remote Defect Indication (ETH-RDI) function uses a bit in the CFM CC message to communicate defect conditions to the MEP peers. For ETH-RDI functionality, you must configure the MEP MEG level, the ETH-CC transmission period, and the ETH-CC frame priority. ETH-RDI does not require any MIP configuration.

When a MEP receives frames with ETH-RDI information, it determines that its peer MEP has encountered a defect condition and sets the RDI files in the CCM frames for the duration of the defect condition. When the defect condition clears, the MEP clears the RDI field.

When a MEP receives a CCM frame, it examines it to ensure that its MEG level is the same and if the RDI field is set, it detects an RDI condition. For point-to-point Ethernet connections, a MEP can clear the RDI condition when it receives the first frame from its peer MEP with the RDI field cleared. However, for multipoint Ethernet connectivity, the MEP cannot determine the associated subset of peer MEPs with which the sending MEP has seen the defect condition. It can clear the RDI condition after it receives CCM frames with the RDI field cleared from its entire list of peer MEPs.

Ethernet Locked Signal

Note Ethernet locked signal is not supported in Release 15.0(1)MR.

The Ethernet Locked Signal (ETH-LCK) function communicates the administrative locking of a server MEP and interruption of data traffic being forwarded to the MEP expecting the traffic. A MEP that receives frames with ETH-LCK information can differentiate between a defect condition and an administrative locking. ETH-LCK relies on loopback information (local and remote). The default timer for ETH-LCK is 60 seconds and the default level is the MIP level.

When a MEP is administratively locked, it sends LCK frames in a direction opposite to its peer MEPs, based on the LCK transmission period, which is the same as the AIS transmission period. The first LCK frame is sent immediately following the administrative or diagnostic action.

A MEP receiving a LCK frame verifies that the maintenance level matches its configured maintenance level, and detects a LCK condition. When no LCK frames are received for an interval of 3.5 times the LCK transmission period, the MEP clears the LCK condition.

Multicast Ethernet Loopback

The multicast Ethernet loopback (ETH-LB) function verifies bidirectional connectivity of a MEP with its peer MEPs and is an on-demand OAM function. When the feature is invoked on a MEP by entering the **ping** privileged EXEC command, the MEP sends a multicast frame with ETH-LB request information to peer MEPs in the same MEG. The MEP expects to receive a unicast frame with ETH-LB reply information from its peer MEPs within a specified time period. A MEP receiving a multicast frame with ETH-LB request information validates the frame and transmits a frame with reply information.

To configure multicast ETH-LB, you configure the MEG level of the MEP and the priority of the multicast frames with ETH-LB requests. Multicast frames with ETH-LB request information are always marked as drop ineligible. No MIP configuration is required.

The MEP sends multicast LB message frames on an on-demand basis. After sending a multicast LBM frame, the MEP expects to receive LB reply frames within 5 seconds.

When a MEP receives a valid LBM frame, it generates an LB reply frame and sends it to the requested MEP after a random delay in the range of 0 to 1 second. The validity of the frame is determined on its having the correct MEG level.

When a MEP sends a multicast LBM frame and receives an LB reply frame within 5 seconds, the LB reply frame is valid.

Configuring Y.1731 Fault Management

To configure Y.1731 fault management, you must enable CFM and configure MIPs on the participating interfaces. AIS messages are generated only on interfaces with a configured MIP.

- Default Y.1731 Configuration, page 15-24
- Configuring ETH-AIS, page 15-24
- Using Multicast Ethernet Loopback, page 15-26

Default Y.1731 Configuration

ETH-AIS is enabled by default when CFM is enabled.

When you configure ETH-AIS, you must configure CFM before ETH-AIS is operational.

ETH-RDI is set automatically when continuity check messages are enabled.

Configuring ETH-AIS

Beginning in privileged EXEC mode, follow these steps to configure Ethernet AIS on the router:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ethernet cfm ais link-status global**	Configure AIS-specific SMEP commands by entering config-ais-link-cfm mode.
Step 3	**level** *level-id* or **disable**	Configure the maintenance level for sending AIS frames transmitted by the SMEP. The range is 0 to 7. or Disable generation of ETH-AIS frames.
Step 4	**period** *value*	Configure the SMEP AIS transmission period interval. Allowable values are 1 second or 60 seconds.

	Command	Purpose
Step 5	**exit**	Return to global configuration mode.
Step 6	**ethernet cfm domain** *domain-name* **level** *level-id*	Define a CFM domain, set the domain level, and enter ethernet-cfm configuration mode for the domain. The maintenance level number range is 0 to 7.
Step 7	**service** {*ma-name* | *ma-number* | *vpn-id vpn*} {**vlan** *vlan-id* [**direction down**] | **port**}	Define a customer service maintenance association (MA) name or number to be associated with the domain, or a VLAN ID or VPN-ID, and enter ethernet-cfm-service configuration mode. • *ma-name*—a string of no more than 100 characters that identifies the MAID. • *ma-number*—a value from 0 to 65535. • *vpn-id*—enter a VPN ID as the *ma-name*. • **vlan** *vlan-id*—VLAN range is from 1 to 4094. You cannot use the same VLAN ID for more than one domain at the same level. • (Optional) **direction down**—specify the service direction as down. • **port**—Configure port MEP, a down MEP that is untagged and not associated with a VLAN.
Step 8	**ais level** *level-id*	(Optional) Configure the maintenance level for sending AIS frames transmitted by the MEP. The range is 0 to 7.
Step 9	**ais period** *value*	(Optional) Configure the MEP AIS transmission period interval. Allowable values are 1 second or 60 seconds.
Step 10	**ais expiry-threshold** *value*	(Optional) Set the expiring threshold for the MA as an integer. The range is 2 to 255. The default is 3.5.
Step 11	**no ais suppress-alarms**	(Optional) Override the suppression of redundant alarms when the MEP goes into an AIS defect condition after receiving an AIS message.
Step 12	**exit**	Return to ethernet-cfm configuration mode.
Step 13	**exit**	Return to global configuration mode.
Step 14	**interface** *interface-id*	Specify an interface ID, and enter interface configuration mode.
Step 15	[**no**] **ethernet cfm ais link-status**	Enable or disable sending AIS frames from the SMEP on the interface.
Step 16	**ethernet cfm ais link-status period** *value*	Configure the ETH-AIS transmission period generated by the SMEP on the interface. Allowable values are 1 second or 60 seconds.
Step 17	**ethernet cfm ais link-status level** *level-id*	Configure the maintenance level for sending AIS frames transmitted by the SMEP on the interface. The range is 0 to 7.
Step 18	**end**	Return to privileged EXEC mode.
Step 19	**show ethernet cfm smep** [**interface** *interface-id*]	Verify the configuration.

	Command	Purpose
Step 20	**show ethernet cfm error**	Display received ETH-AIS frames and other errors.
Step 21	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of the commands to return to the default configuration or to remove a configuration. To disable the generation of ETH-AIS frames, enter the **disable** config-ais-link-cfm mode command.

This is an example of the output from the **show ethernet cfm smep** command when Ethernet AIS has been enabled:

```
Router# show ethernet cfm smep
SMEP Settings:
--------------
Interface: GigabitEthernet1/0/3
LCK-Status: Enabled
LCK Period: 60000 (ms)
Level to transmit LCK: Default
AIS-Status: Enabled
AIS Period: 60000 (ms)
Level to transmit AIS: Default
Defect Condition: AIS
```

Using Multicast Ethernet Loopback

You can use the **ping** privileged EXEC command to verify bidirectional connectivity of a MEP, as in this example:

```
Router# ping ethernet multicast domain CD vlan 10
Type escape sequence to abort.
Sending 5 Ethernet CFM loopback messages to 0180.c200.0037, timeout is 5 seconds:
Reply to Multicast request via interface FastEthernet1/0/3, from 001a.a17e.f880, 8 ms
Total Loopback Responses received: 1
```

Managing and Displaying Ethernet CFM Information

You can use the privileged EXEC commands in these tables to clear Ethernet CFM information.

Table 1 Clearing CFM Information

Command	Purpose
clear ethernet cfm ais domain *domain-name* **mpid** *id* {**vlan** *vlan-id* | **port**}	Clear MEPs with matching domain and VLAN ID out of AIS defect condition.
clear ethernet cfm ais link-status interface *interface-id*	Clear a SMEP out of AIS defect condition.
clear ethernet cfm error	Clear all CFM error conditions, including AIS.

You can use the privileged EXEC commands in Table 15-2 to display Ethernet CFM information.

Table 15-2 Displaying CFM Information

Command	Purpose
show ethernet cfm domain [brief]	Displays CFM domain information or brief domain information.
show ethernet cfm errors [configuration | domain-id]	Displays CFM continuity check error conditions logged on a device since it was last reset or the log was last cleared. When CFM crosscheck is enabled, displays the results of the CFM crosscheck operation.
show ethernet cfm maintenance-points local [detail | domain | interface | level | mep | mip]	Displays maintenance points configured on a device.
show ethernet cfm maintenance-points remote [crosscheck | detail | domain | static]	Displays information about a remote maintenance point domains or levels or details in the CFM database.
show ethernet cfm mpdb	Displays information about entries in the MIP continuity-check database.
show ethernet cfm smep [**interface** *interface-id*]	Displays Ethernet CFM SMEP information.
show ethernet cfm traceroute-cache	Displays the contents of the traceroute cache.
show platform cfm	Displays platform-independent CFM information.

This is an example of output from the **show ethernet cfm domain brief** command:

```
Router# show ethernet cfm domain brief
Domain Name                               Index Level Services Archive(min)
level5                                        1     5        1     100
level3                                        2     3        1     100
test                                          3     3        3     100
name                                          4     3        1     100
test1                                         5     2        1     100
lck                                           6     1        1     100Total Services : 1
```

This is an example of output from the **show ethernet cfm errors** command:

```
Router# show ethernet cfm errors
--------------------------------------------------------------------------------
MPID Domain Id                                      Mac Address     Type   Id  Lvl
     MAName                                         Reason                 Age
--------------------------------------------------------------------------------
6307 level3                                         0021.d7ee.fe80  Vlan   7    3
     vlan7                                          Receive RDI            5s
```

This is an example of output from the **show ethernet cfm maintenance-points local detail** command:

```
Router# show ethernet cfm maintenance-points local detail
Local MEPs:
----------
MPID: 7307
DomainName: level3
Level: 3
Direction: Up
Vlan: 7
Interface: Gi0/3
CC-Status: Enabled
CC Loss Threshold: 3
MAC: 0021.d7ef.0700
LCK-Status: Enabled
LCK Period: 60000(ms)
LCK Expiry Threshold: 3.5
Level to transmit LCK: Default
```

```
Defect Condition: No Defect
presentRDI: FALSE
AIS-Status: Enabled
AIS Period: 60000(ms)
AIS Expiry Threshold: 3.5
Level to transmit AIS: Default
Suppress Alarm configuration: Enabled
Suppressing Alarms: No


MIP Settings:
-------------
Local MIPs:
* = MIP Manually Configured
------------------------------------------------------------------------------
 Level Port           MacAddress     SrvcInst   Type    Id
------------------------------------------------------------------------------
*5     Gi0/3          0021.d7ef.0700 N/A        Vlan    2,7
```

This is an example of output from the **show ethernet cfm traceroute** command:

```
Router# show ethernet cfm traceroute
Current Cache-size: 0 Hops
Max Cache-size: 100 Hops
Hold-time: 100 Minutes
```

You can use the privileged EXEC commands in Table 15-3 to display IP SLAs Ethernet CFM information.

Table 15-3 Displaying IP SLAs CFM Information

Command	Purpose
show ip sla configuration [*entry-number*]	Displays configuration values including all defaults for all IP SLAs operations or a specific operation.
show ip sla ethernet-monitor configuration [*entry-number*]	Displays the configuration of the IP SLAs automatic Ethernet operation.
show ip sla statistics [*entry-number* | **aggregated** | **details**]	Display current or aggregated operational status and statistics.

Understanding the Ethernet OAM Protocol

The Ethernet OAM protocol for installing, monitoring, and troubleshooting Metro Ethernet networks and Ethernet WANs relies on an optional sublayer in the data link layer of the OSI model. Normal link operation does not require Ethernet OAM. You can implement Ethernet OAM on any full-duplex point-to-point or emulated point-to-point Ethernet link for a network or part of a network (specified interfaces).

OAM frames, called OAM protocol data units (OAM PDUs) use the slow protocol destination MAC address 0180.c200.0002. They are intercepted by the MAC sublayer and cannot propagate beyond a single hop within an Ethernet network. Ethernet OAM is a relatively slow protocol, with a maximum transmission rate of 10 frames per second, resulting in minor impact to normal operations. However, when you enable link monitoring, because the CPU must poll error counters frequently, the number of required CPU cycles is proportional to the number of interfaces that must be polled.

Ethernet OAM has two major components:

- The OAM client establishes and manages Ethernet OAM on a link and enables and configures the OAM sublayer. During the OAM discovery phase, the OAM client monitors OAM PDUs received from the remote peer and enables OAM functionality. After the discovery phase, it manages the rules of response to OAM PDUs and the OAM remote loopback mode.
- The OAM sublayer presents two standard IEEE 802.3 MAC service interfaces facing the superior and inferior MAC sublayers. It provides a dedicated interface for the OAM client to pass OAM control information and PDUs to and from the client. It includes these components:
  - The control block provides the interface between the OAM client and other OAM sublayer internal blocks.
  - The multiplexer manages frames from the MAC client, the control block, and the parser and passes OAM PDUs from the control block and loopback frames from the parser to the subordinate layer.
  - The parser classifies frames as OAM PDUs, MAC client frames, or loopback frames and sends them to the appropriate entity: OAM PDUs to the control block, MAC client frames to the superior sublayer, and loopback frames to the multiplexer.

OAM Features

These OAM features are defined by IEEE 802.3ah:

- Discovery
- Link Monitoring
- Remote Failure Indication
- Remote Loopback

Discovery

Discovery is the first phase of Ethernet OAM and it identifies the devices in the network and their OAM capabilities. Discovery uses information OAM PDUs. During the discovery phase, the following information is advertised within periodic information OAM PDUs:

- OAM mode—Conveyed to the remote OAM entity. The mode can be either active or passive and can be used to determine device functionality.
- OAM configuration (capabilities)—Advertises the capabilities of the local OAM entity. With this information a peer can determine what functions are supported and accessible; for example, loopback capability.
- OAM PDU configuration—Includes the maximum OAM PDU size for receipt and delivery. This information along with the rate limiting of 10 frames per second can be used to limit the bandwidth allocated to OAM traffic.
- Platform identity—A combination of an organization unique identifier (OUI) and 32-bits of vendor-specific information. OUI allocation, controlled by the IEEE, is typically the first three bytes of a MAC address.

Discovery includes an optional phase in which the local station can accept or reject the configuration of the peer OAM entity. For example, a node may require that its partner support loopback capability to be accepted into the management network. These policy decisions may be implemented as vendor-specific extensions.

Link Monitoring

Link monitoring in Ethernet OAM detects and indicates link faults under a variety of conditions. Link monitoring uses the event notification OAM PDU and sends events to the remote OAM entity when there are problems detected on the link. The error events include the following:

- Error Symbol Period (error symbols per second)—The number of symbol errors that occurred during a specified period exceeded a threshold. These errors are coding symbol errors.
- Error Frame (error frames per second)—The number of frame errors detected during a specified period exceeded a threshold.
- Error Frame Period (error frames per n frames)—The number of frame errors within the last n frames has exceeded a threshold.
- Error Frame Seconds Summary (error seconds per m seconds)—The number of error seconds (1-second intervals with at least one frame error) within the last m seconds has exceeded a threshold.

Since IEEE 802.3ah OAM does not provide a guaranteed delivery of any OAM PDU, the event notification OAM PDU may be sent multiple times to reduce the probability of a lost notification. A sequence number is used to recognize duplicate events.

Remote Failure Indication

Faults in Ethernet connectivity that are caused by slowly deteriorating quality are difficult to detect. Ethernet OAM provides a mechanism for an OAM entity to convey these failure conditions to its peer via specific flags in the OAM PDU. The following failure conditions can be communicated:

- Link Fault—Loss of signal is detected by the receiver; for instance, the peer's laser is malfunctioning. A link fault is sent once per second in the information OAM PDU. Link fault applies only when the physical sublayer is capable of independently transmitting and receiving signals.
- Dying Gasp—An unrecoverable condition has occurred; for example, a power failure. This type of condition is vendor specific. A notification about the condition may be sent immediately and continuously.
- Critical Event—An unspecified critical event has occurred. This type of event is vendor specific. A critical event may be sent immediately and continuously.

Remote Loopback

An OAM entity can put its remote peer into loopback mode using the loopback control OAM PDU. Loopback mode helps an administrator ensure the quality of links during installation or when troubleshooting. In loopback mode, every frame received is transmitted back on the same port except for OAM PDUs and pause frames. The periodic exchange of OAM PDUs must continue during the loopback state to maintain the OAM session.

The loopback command is acknowledged by responding with an information OAM PDU with the loopback state indicated in the state field. This acknowledgement allows an administrator, for example, to estimate if a network segment can satisfy a service-level agreement. Acknowledgement makes it possible to test delay, jitter, and throughput.

When an interface is set to the remote loopback mode the interface no longer participates in any other Layer 2 or Layer 3 protocols; for example Spanning Tree Protocol (STP) or Open Shortest Path First (OSPF). The reason is that when two connected ports are in a loopback session, no frames other than the OAM PDUs are sent to the CPU for software processing. The non-OAM PDU frames are either looped back at the MAC level or discarded at the MAC level.

From a user's perspective, an interface in loopback mode is in a link-up state.

Cisco Vendor-Specific Extensions

Ethernet OAM allows vendors to extend the protocol by allowing them to create their own type-length-value (TLV) fields.

OAM Messages

Ethernet OAM messages or OAM PDUs are standard length, untagged Ethernet frames within the normal frame length bounds of 64 to 1518 bytes. The maximum OAM PDU frame size exchanged between two peers is negotiated during the discovery phase.

OAM PDUs always have the destination address of slow protocols (0180.c200.0002) and an Ethertype of 8809. OAM PDUs do not go beyond a single hop and have a hard-set maximum transmission rate of 10 OAM PDUs per second. Some OAM PDU types may be transmitted multiple times to increase the likelihood that they will be successfully received on a deteriorating link.

Four types of OAM messages are supported:

- Information OAM PDU—A variable-length OAM PDU that is used for discovery. This OAM PDU includes local, remote, and organization-specific information.
- Event notification OAM PDU—A variable-length OAM PDU that is used for link monitoring. This type of OAM PDU may be transmitted multiple times to increase the chance of a successful receipt; for example, in the case of high-bit errors. Event notification OAM PDUs also may include a time stamp when generated.
- Loopback control OAM PDU—An OAM PDU fixed at 64 bytes in length that is used to enable or disable the remote loopback command.
- Vendor-specific OAM PDU—A variable-length OAM PDU that allows the addition of vendor-specific extensions to OAM.

For instructions on how to configure Ethernet Link OAM, see Setting Up and Configuring Ethernet OAM, page 15-31.

Setting Up and Configuring Ethernet OAM

This section includes this information:

- Default Ethernet OAM Configuration, page 15-32
- Ethernet OAM Configuration Guidelines, page 15-32
- Enabling Ethernet OAM on an Interface, page 15-32
- Enabling Ethernet OAM Remote Loopback, page 15-33
- Configuring Ethernet OAM Link Monitoring, page 15-34
- Configuring Ethernet OAM Remote Failure Indications, page 15-37
- Configuring Ethernet OAM Templates, page 15-37

Default Ethernet OAM Configuration

Ethernet OAM is disabled on all interfaces.

When Ethernet OAM is enabled on an interface, link monitoring is automatically turned on.

Remote loopback is disabled.

No Ethernet OAM templates are configured.

Ethernet OAM Configuration Guidelines

Follow these guidelines when configuring Ethernet OAM:

- The router does not support monitoring of egress frames sent with cyclic redundancy code (CDC) errors. The **ethernet oam link-monitor transmit crc** interface-configuration or template-configuration commands are visible but are not supported on the router. The commands are accepted, but are not applied to an interface.
- For a remote failure indication, the router does not generate Link Fault or Critical Event OAM PDUs. However, if these PDUs are received from a link partner, they are processed. The router supports generating and receiving Dying Gasp OAM PDUs when Ethernet OAM is disabled, the interface is shut down, the interface enters the error-disabled state, or the router is reloading. It can respond to, but not generate, Dying Gasp PDUs based on loss of power.

Enabling Ethernet OAM on an Interface

Beginning in privileged EXEC mode, follow these steps to enable Ethernet OAM on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Define an interface to configure as an EOM interface, and enter interface configuration mode.
Step 3	**ethernet oam**	Enable Ethernet OAM on the interface.

	Command	Purpose
Step 4	**ethernet oam** [**max-rate** *oampdus* | **min-rate** *seconds* | **mode** {**active** | **passive**} | **timeout** *seconds*]	You can configure these optional OAM parameters: • (Optional) Enter **max-rate** *oampdus* to configure the maximum number of OAM PDUs sent per second. The range is from 1 to 10. • (Optional) Enter **min-rate** *seconds* to configure the minimum transmission rate in seconds when one OAM PDU is sent per second. The range is from 1 to 10. • (Optional) Enter **mode active** to set OAM client mode to active. • (Optional) Enter **mode passive** to set OAM client mode to passive. **Note** When Ethernet OAM mode is enabled on two interfaces passing traffic, at least one must be in the active mode. • (Optional) Enter **timeout** *seconds* to set a time for OAM client timeout. The range is from 2 to 30.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ethernet oam status** [**interface** *interface-id*]	Verify the configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Enter the **no ethernet oam** interface configuration command to disable Ethernet OAM on the interface.

Enabling Ethernet OAM Remote Loopback

You must enable Ethernet OAM remote loopback on an interface for the local OAM client to initiate OAM remote loopback operations. Changing this setting causes the local OAM client to exchange configuration information with its remote peer. Remote loopback is disabled by default.

Remote loopback has these limitations:

- Internet Group Management Protocol (IGMP) packets are not looped back.
- If dynamic ARP inspection is enabled, ARP or reverse ARP packets are not looped or dropped.

Beginning in privileged EXEC mode, follow these steps to enable Ethernet OAM remote loopback on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Define an interface to configure as an EOM interface, and enter interface configuration mode.

	Command	Purpose
Step 3	**ethernet oam remote-loopback** {**supported** | **timeout** *seconds*}	Enable Ethernet remote loopback on the interface or set a loopback timeout period. • Enter **supported** to enable remote loopback. • Enter **timeout** *seconds* to set a remote loopback timeout period. The range is from 1 to 10 seconds.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**ethernet oam remote-loopback** {**start** | **stop**} {**interface** *interface-id*}	Turn on or turn off Ethernet OAM remote loopback on an interface.
Step 6	**show ethernet oam status** [**interface** *interface-id*]	Verify the configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ethernet oam remote-loopback** {**supported** | **timeout**} interface configuration command to disable remote loopback support or remove the timeout setting.

Configuring Ethernet OAM Link Monitoring

You can configure high and low thresholds for link-monitoring features. If no high threshold is configured, the default is **none** —no high threshold is set. If you do not set a low threshold, it defaults to a value lower than the high threshold.

Beginning in privileged EXEC mode, follow these steps to configure Ethernet OAM link monitoring on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Define an interface, and enter interface configuration mode.
Step 3	**ethernet oam link-monitor supported**	Enable the interface to support link monitoring. This is the default. You need to enter this command only if it has been disabled by previously entering the **no ethernet oam link-monitor supported** command.
Step 4	**ethernet oam link-monitor high-threshold action** {**error-disable-interface** | **failover**}	Use the **ethernet oam link-monitor high-threshold** command to configure an error-disable function on the Ethernet OAM interface when a high threshold for an error is exceeded. **Note** Release 15.0(1)MR does not support the **failover** keyword.

	Command	Purpose
Step 5	**ethernet oam link-monitor symbol-period** {**threshold** {**high** {*high symbols* | **none**} | **low** {*low-symbols*}} | **window** *symbols*} **Note** Repeat this step to configure both high and low thresholds.	(Optional) Configure high and low thresholds for an error-symbol period that trigger an error-symbol period link event. • Enter **threshold high** *high-symbols* to set a high threshold in number of symbols. The range is 1 to 65535. The default is **none**. • Enter **threshold high none** to disable the high threshold if it was set. This is the default. • Enter **threshold low** *low-symbols* to set a low threshold in number of symbols. The range is 0 to 65535. It must be lower than the high threshold. • Enter **window** *symbols* to set the window size (in number of symbols) of the polling period. The range is 1 to 65535 symbols.
Step 6	**ethernet oam link-monitor frame** {**threshold** {**high** {*high-frames* | **none**} | **low** {*low-frames*}} | **window** *milliseconds*} **Note** Repeat this step to configure both high and low thresholds.	(Optional) Configure high and low thresholds for error frames that trigger an error-frame link event. • Enter **threshold high** *high-frames* to set a high threshold in number of frames. The range is 1 to 65535. The default is **none**. • Enter **threshold high none** to disable the high threshold if it was set. This is the default. • Enter **threshold low** *low-frames* to set a low threshold in number of frames. The range is 0 to 65535. The default is 1. • Enter **window** *milliseconds* to set the a window and period of time during which error frames are counted. The range is 10 to 600 and represents the number of milliseconds in multiples of 100. The default is 100.
Step 7	**ethernet oam link-monitor frame-period** {**threshold** {**high** {*high-frames* | **none**} | **low** {*low-frames*}} | **window** *frames*} **Note** Repeat this step to configure both high and low thresholds.	(Optional) Configure high and low thresholds for the error-frame period that triggers an error-frame-period link event. • Enter **threshold high** *high-frames* to set a high threshold in number of frames. The range is 1 to 65535. The default is **none**. • Enter **threshold high none** to disable the high threshold if it was set. This is the default. • Enter **threshold low** *low-frames* to set a low threshold in number of frames. The range is 0 to 65535. The default is 1. • Enter **window** *frames* to set the a polling window size in number of frames. The range is 1 to 65535; each value is a multiple of 10000 frames. The default is 1000.

	Command	Purpose
Step 8	**ethernet oam link-monitor frame-seconds** {**threshold** {**high** {*high-frames* | **none**} | **low** {*low-frames*}} | **window** *milliseconds*} **Note** Repeat this step to configure both high and low thresholds.	(Optional) Configure high and low thresholds for the frame-seconds error that triggers an error-frame-seconds link event. • Enter **threshold high** *high-frames* to set a high error frame-seconds threshold in number of seconds. The range is 1 to 900. The default is none. • Enter **threshold high none** to disable the high threshold if it was set. This is the default. • Enter **threshold low** *low-frames* to set a low threshold in number of frames. The range is 1 to 900. The default is 1. • Enter **window** *frames* to set the a polling window size in number of milliseconds. The range is 100 to 9000; each value is a multiple of 100 milliseconds. The default is 1000.
Step 9	**ethernet oam link-monitor receive-crc** {**threshold** {**high** {*high-frames* | **none**} | **low** {*low-frames*}} | **window** *milliseconds*} **Note** Repeat this step to configure both high and low thresholds.	(Optional) Configure thresholds for monitoring ingress frames received with cyclic redundancy code (CRC) errors for a period of time. • Enter **threshold high** *high-frames* to set a high threshold for the number of frames received with CRC errors. The range is 1 to 65535 frames. • Enter **threshold high none** to disable the high threshold. • Enter **threshold low** *low-frames* to set a low threshold in number of frames. The range is 0 to 65535. The default is 1. • Enter **window** *milliseconds* to set the a window and period of time during which frames with CRC errors are counted. The range is 10 to 1800 and represents the number of milliseconds in multiples of 100. The default is 100.
Step 10	**ethernet oam link-monitor transmit-crc** {**threshold** {**high** {*high-frames* | **none**} | **low** *low-frames*} | **window** *milliseconds*} }	Use the **ethernet oam link-monitor transmit-crc** command to configure an Ethernet OAM interface to monitor egress frames with CRC errors for a period of time.
Step 11	**[no] ethernet link-monitor on**	(Optional) Start or stop (when the **no** keyword is entered) link-monitoring operations on the interface. Link monitoring operations start automatically when support is enabled.
Step 12	**end**	Return to privileged EXEC mode.
Step 13	**show ethernet oam status** [**interface** *interface-id*]	Verify the configuration.
Step 14	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

The **ethernet oam link-monitor transmit-crc** {**threshold** {**high** {*high-frames* | **none**} | **low** {*low-frames*}} | **window** *milliseconds*} command is visible on the router and you are allowed to enter it, but it is not supported.Enter the **no** form of the commands to disable the configuration. Use the **no** form of each command to disable the threshold setting.

Configuring Ethernet OAM Remote Failure Indications

You can configure an error-disable action to occur on an interface if one of the high thresholds is exceeded, if the remote link goes down, if the remote device is rebooted, or if the remote device disables Ethernet OAM on the interface.

Beginning in privileged EXEC mode, follow these steps to enable Ethernet OAM remote-failure indication actions on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Define an interface, and enter interface configuration mode.
Step 3	**ethernet oam remote-failure** {**critical-event** | **dying-gasp** | **link-fault**} **action error-disable-interface**	Configure the Ethernet OAM remote-failure action on the interface. You can configure disabling the interface for one of these conditions: • Select **critical-event** to shut down the interface when an unspecified critical event has occurred. • Select **dying-gasp** to shut down the interface when Ethernet OAM is disabled or the interface enters the error-disabled state. • Select **link-fault** to shut down the interface when the receiver detects a loss of signal.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ethernet oam status** [**interface** *interface-id*]	Verify the configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

The router does not generate Link Fault or Critical Event OAM PDUs. However, if these PDUs are received from a link partner, they are processed. The router supports sending and receiving Dying Gasp OAM PDUs when Ethernet OAM is disabled, the interface is shut down, the interface enters the error-disabled state, or the router is reloading. It can respond to, but not generate, Dying Gasp PDUs based on loss of power. Enter the **no ethernet remote-failure** {**critical-event** | **dying-gasp** | **link-fault**} **action** command to disable the remote failure indication action.

Configuring Ethernet OAM Templates

You can create a template for configuring a common set of options on multiple Ethernet OAM interfaces. The template can be configured to monitor frame errors, frame-period errors, frame-second errors, received CRS errors, and symbol-period errors and thresholds. You can also set the template to put the interface in error-disabled state if any high thresholds are exceeded. These steps are optional and can be performed in any sequence or repeated to configure different options.

Beginning in privileged EXEC mode, follow these steps to configure an Ethernet OAM template and to associate it with an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**template** *template-name*	Create a template, and enter template configuration mode.
Step 3	**ethernet oam link-monitor receive-crc** {**threshold** {**high** {*high-frames* | **none**} | **low** {*low-frames*}} | **window** *milliseconds*}	(Optional) Configure thresholds for monitoring ingress frames received with cyclic redundancy code (CRC) errors for a period of time. • Enter **threshold high** *high-frames* to set a high threshold for the number of frames received with CRC errors. The range is 1 to 65535 frames. • Enter **threshold high none** to disable the high threshold. • Enter **threshold low** *low-frames* to set a low threshold in number of frames. The range is 0 to 65535. The default is 1. • Enter **window** *milliseconds* to set the a window and period of time during which frames with CRC errors are counted. The range is 10 to 1800 and represents the number of milliseconds in multiples of 100. The default is 100.
Step 4	**ethernet oam link-monitor symbol-period** {**threshold** {**high** {*high symbols* | **none**} | **low** {*low-symbols*}} | **window** *symbols*}	(Optional) Configure high and low thresholds for an error-symbol period that triggers an error-symbol period link event. • Enter **threshold high** *high-symbols* to set a high threshold in number of symbols. The range is 1 to 65535. • Enter **threshold high none** to disable the high threshold. • Enter **threshold low** *low-symbols* to set a low threshold in number of symbols. The range is 0 to 65535. It must be lower than the high threshold. • Enter **window** *symbols* to set the window size (in number of symbols) of the polling period. The range is 1 to 65535 symbols.

	Command	Purpose
Step 5	**ethernet oam link-monitor frame** {**threshold** {**high** {*high-frames* | **none**} | **low** {*low-frames*}} | **window** *milliseconds*}	(Optional) Configure high and low thresholds for error frames that trigger an error-frame link event. • Enter **threshold high** *high-frames* to set a high threshold in number of frames. The range is 1 to 65535. You must enter a high threshold. • Enter **threshold high none** to disable the high threshold. • Enter **threshold low** *low-frames* to set a low threshold in number of frames. The range is 0 to 65535. The default is 1. • Enter **window** *milliseconds* to set the a window and period of time during which error frames are counted. The range is 10 to 600 and represents the number of milliseconds in a multiple of 100. The default is 100.
Step 6	**ethernet oam link-monitor frame-period** {**threshold** {**high** {*high-frames* | **none**} | **low** {*low-frames*}} | **window** *frames*}	(Optional) Configure high and low thresholds for the error-frame period that triggers an error-frame-period link event. • Enter **threshold high** *high-frames* to set a high threshold in number of frames. The range is 1 to 65535. You must enter a high threshold. • Enter **threshold high none** to disable the high threshold. • Enter **threshold low** *low-frames* to set a low threshold in number of frames. The range is 0 to 65535. The default is 1. • Enter **window** *frames* to set the a polling window size in number of frames. The range is 1 to 65535; each value is a multiple of 10000 frames. The default is 1000.
Step 7	**ethernet oam link-monitor frame-seconds** {**threshold** {**high** {*high-seconds* | **none**} | **low** {*low-seconds*}} | **window** *milliseconds*}	(Optional) Configure frame-seconds high and low thresholds for triggering an error-frame-seconds link event. • Enter **threshold high** *high-seconds* to set a high threshold in number of seconds. The range is 1 to 900. You must enter a high threshold. • Enter **threshold high none** to disable the high threshold. • Enter **threshold low** *low-frames* to set a low threshold in number of frames. The range is 1 to 900. The default is 1. • Enter **window** *frames* to set the a polling window size in number of frames. The range is 100 to 9000; each value is a multiple of 100 milliseconds. The default is 1000.

	Command	Purpose
Step 8	**ethernet oam link-monitor high threshold action error-disable-interface**	(Optional) Configure the router to put an interface in an error disabled state when a high threshold for an error is exceeded.
Step 9	**exit**	Return to global configuration mode.
Step 10	**interface** *interface-id*	Define an Ethernet OAM interface, and enter interface configuration mode.
Step 11	**source-template** *template-name*	Associate the template to apply the configured options to the interface.
Step 12	**end**	Return to privileged EXEC mode.
Step 13	**show ethernet oam status** [**interface** *interface-id*]	Verify the configuration.
Step 14	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

The router does not support monitoring egress frames with CRC errors. The **ethernet oam link-monitor transmit-crc** {**threshold** {**high** {*high-frames* | **none**} | **low** {*low-frames*}} | **window** *milliseconds*} command is visible on the router and you can enter it, but it is not supported. Use the **no** form of each command to remove the option from the template. Use the **no source-template** *template-name* to remove the source template association.

Displaying Ethernet OAM Protocol Information

You can use the privileged EXEC commands in Table 15-4 to display Ethernet OAM protocol information.

Table 15-4 Displaying Ethernet OAM Protocol Information

Command	Purpose
show ethernet oam discovery [**interface** *interface-id*]	Displays discovery information for all Ethernet OAM interfaces or the specified interface.
show ethernet oam statistics [**interface** *interface-id*]	Displays detailed information about Ethernet OAM packets.
show ethernet oam status [**interface** *interface-id*]	Displays Ethernet OAM configuration for all interfaces or the specified interface.
show ethernet oam summary	Displays active Ethernet OAM sessions on the router.

Understanding E-LMI

Ethernet Local Management Interface (E-LMI) is a protocol between the customer-edge (CE) device and the provider-edge (PE) device. It runs only on the PE-to-CE UNI link and notifies the CE device of connectivity status and configuration parameters of Ethernet services available on the CE port. E-LMI interoperates with an OAM protocol, such as CFM, that runs within the provider network to collect OAM status. CFM runs at the provider maintenance level (UPE to UPE with inward-facing MEPs at the UNI).

OAM manager, which streamlines interaction between any two OAM protocols, handles the interaction between CFM and E-LMI. This interaction is unidirectional, running only from OAM manager to E-LMI on the UPE side of the router. Information is exchanged either as a result of a request from E-LMI or triggered by OAM when it received notification of a change from the OAM protocol. This type of information is relayed:

- EVC name and availability status
- Remote UNI name and status
- Remote UNI counts

You can configure Ethernet virtual connections (EVCs), service VLANs, UNI ids (for each CE-to-PE link), and UNI count and attributes. You need to configure CFM to notify the OAM manager of any change to the number of active UNIs and or the remote UNI ID for a given S-VLAN domain.

You can configure the router as either the customer-edge device or the provider-edge device.

Note The Cisco MWR 2941 does not support Ethernet Virtual Connections (EVCs).

Note The Cisco MWR 2941 does not support OAM Manager.

Configuring E-LMI

For E-LMI to work with CFM, you configure Ethernet virtual connections (EVCs), Ethernet service instances (EFPs), and E-LMI customer VLAN mapping. Most of the configuration occurs on the PE device on the interfaces connected to the CE device. On the CE device, you only need to enable E-LMI on the connecting interface. Note that you must configure some OAM parameters, for example, EVC definitions, on PE devices on both sides of a metro network.

Note The Cisco MWR 2941 does not support Ethernet Virtual Connections (EVCs).

This section includes this information:

- Default E-LMI Configuration, page 15-41
- Enabling E-LMI, page 15-42
- Customer-Edge Device Configuration, page 15-43

Default E-LMI Configuration

Ethernet LMI is globally disabled by default. When enabled, the router is in provider-edge (PE) mode by default.

When you globally enable E-LMI by entering the **ethernet lmi global** global configuration command, it is automatically enabled on all interfaces. You can also enable or disable E-LMI per interface to override the global configuration. The E-LMI command that is given last is the command that has precedence.

There are no EVCs, EFP service instances, or UNIs defined.

UNI bundling service is bundling with multiplexing.

Enabling E-LMI

You can enable E-LMI globally or on an interface and you can configure the router as a PE or a CE device. Beginning in privileged EXEC mode, follow these steps to enable for E-LMI on the router or on an interface. Note that the order of the global and interface commands determines the configuration. The command that is entered last has precedence.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ethernet lmi global**	Globally enable E-LMI on all interfaces. By default, the router is a PE device.
Step 3	**ethernet lmi ce**	(Optional) Configure the router as an E-LMI CE device.
Step 4	**interface** *interface-id*	Define an interface to configure as an E-LMI interface, and enter interface configuration mode.
Step 5	**ethernet lmi interface**	Configure Ethernet LMI on the interface. If E-LMI is enabled globally, it is enabled on all interfaces unless you disable it on specific interfaces. If E-LMI is disabled globally, you can use this command to enable it on specified interfaces.
Step 6	**ethernet lmi** {**n391** *value* | **n393** *value* | **t391** *value*| **t392** *value*}	Configure E-LMI parameters for the UNI. The keywords have these meanings: • **n391** *value*—Set the event counter on the customer equipment. The counter polls the status of the UNI and all Ethernet virtual connections (EVCs). The range is from 1 to 65000; the default is 360. • **n393** *value*—Set the event counter for the metro Ethernet network. The range is from 1 to 10; the default is 4. • **t391** *value*—Set the polling timer on the customer equipment. A polling timer sends status enquiries and when status messages are not received, records errors. The range is from 5 to 30 seconds; the default is 10 seconds. • **t392** *value*—Set the polling verification timer for the metro Ethernet network or the timer to verify received status inquiries. The range is from 5 to 30 seconds, or enter 0 to disable the timer. The default is 15 seconds. **Note** The **t392** keyword is not supported when the router is in CE mode.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show ethernet lmi evc**	Verify the configuration.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ethernet lmi** global configuration command to globally disable E-LMI. Use the **no** form of the **ethernet lmi** interface configuration command with keywords to disable E-LMI on the interface or to return the timers to the default settings.

Use the **show ethernet lmi** commands to display information that was sent to the CE from the status request poll. Use the **show ethernet service** commands to show current status on the device.

Customer-Edge Device Configuration

This example shows the commands necessary to configure E-LMI on the CE device.

This example enables E-LMI globally, but you can also enable it only on a specific interface. However, if you do not enter the **ethernet lmi ce** global configuration command, the interface will be in PE mode by default.

```
Router# config t
Router(config)# ethernet lmi global
Router(config)# ethernet lmi ce
Router(config)# exit
```

Note For E-LMI to work, any VLANs used on the PE device must also be created on the CE device. Create a VLAN by entering the **vlan** *vlan-id* global configuration command on the CE device, where the *vlan-ids* match those on the PE device and configure these VLANs as allowed VLANs by entering the **switchport trunk allowed vlan** *vlan-ids* interface configuration command. Allowed VLANs can receive and send traffic on the interface in tagged format when in trunking mode.

Displaying E-LMI Information

You can use the privileged EXEC commands in Table 15-5 to display E-LMI information.

Table 15-5 Displaying E-LMI Information

Command	Purpose
show ethernet lmi evc [**detail** *evc-id* [**interface** *interface-id*] | **map interface** *type number*]	Displays details sent to the CE from the status request poll about the E-LMI EVC.
show ethernet lmi parameters interface *interface-id*	Displays Ethernet LMI interface parameters sent to the CE from the status request poll.
show ethernet lmi statistics interface *interface-id*	Displays Ethernet LMI interface statistics sent to the CE from the status request poll.
show ethernet lmi uni map interface [*interface-id*]	Displays information about the E-LMI UNI VLAN map sent to the CE from the status request poll.
show ethernet service instance {**detail | id** *efp-identifier* **interface** *interface-id* | **interface** *interface-id*}	Displays information relevant to the specified Ethernet service instances (EFPs).

Enabling Ethernet OAM

Beginning in privileged EXEC mode, follow these steps to enable Ethernet OAM on an interface.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Define an interface to configure as an Ethernet OAM interface and enter interface configuration mode.
Step 3	**ethernet oam** [**max-rate** *oampdus* | **min-rate** *seconds* | **mode** {**active** | **passive**} | **timeout** *seconds*]	Enable Ethernet OAM on the interface • (Optional) Enter **max-rate** *oampdus to* set the maximum rate (per second) to send OAM PDUs. The range is 1 to 10 PDUs per second; the default is 10. • (Optional) Enter **min-rate** *seconds* to set the minimum rate in seconds.The range is 1 to 10 seconds. • (Optional) Set the OAM client **mode** as **active** *or* **passive.** The default is **active.** • (Optional) Enter **timeout** *seconds* to set the time after which a device declares the OAM peer to be nonoperational and resets its state machine. The range is 2 to 30 seconds; the default is 5 seconds.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.
Step 6	**show ethernet cfm maintenance points remote**	(Optional) Display the port states as reported by Ethernet OAM.

Understanding Microwave 1+1 Hot Standby Protocol

The following sections describe the Microwave 1+1 Hot Standby Protocol (HSBY) protocol:

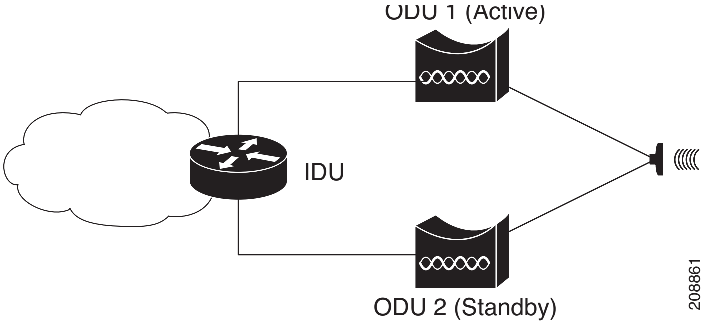
- Overview, page 15-44
- HSBY Maintenance Associations, page 15-45

Overview

Microwave 1+1 Hot Standby Protocol (HSBY) is a link protection protocol developed by Nokia Siemens Networks. HSBY extends the functionality of CFM Continuity Check messages to enable detection and handling of hardware failures in microwave devices in order to provide redundancy. HSBY provides link protection support for indoor units (IDUs) and outdoor units (ODUs).

Figure 15-3 shows a sample physical topology for HSBY using two ODUs (active and standby) and one IDU.

Figure 15-3 HSBY Link Protection Physical Topology



In this topology, the IDU is connected to an active and a standby ODU. While only the active ODU handles data traffic, both ODUs process CFM and management traffic at all times. The HSBY implementation of CFM detects connectivity failures between the IDU and each ODU and indicates which ODU is active and handling traffic. In the event of a failure, the standby ODU assumes the role of the active ODU.

Suspending Continuity Check Messages

Under some circumstances such as a software upgrade or a device reload, it is necessary to temporarily suspend continuity check messages between the ODU and IDU in order to prevent unnecessary link protection action such as a failover. In this case, the ODU sets a suspend flag within the continuity check messages sent to the IDU indicating the amount of time until continuity check messages resume. The IDU resumes exchanging continuity check messages with the ODU after the suspend interval has passed or after the ODU recovers sends a continuity check message.

Note While the Cisco MWR 2941 processes continuity check suspend messages from the IDU, configuration of continuity check messages on the Cisco MWR 2941 is not supported.

HSBY Maintenance Associations

HSBY protocol uses two types of CFM continuity check messages:

- E–CCM—An IDU-to-ODU continuity check message that functions at Ethernet CFM domain level 0. There are two active E–CCM sessions when HSBY is configured.
- P–CCM—An ODU-to-ODU continuity check message that functions at Ethernet CFM domain level 4.

Note The IDU is only associated with the E–CCM sessions; it has outward-facing MEPs configured for each session.

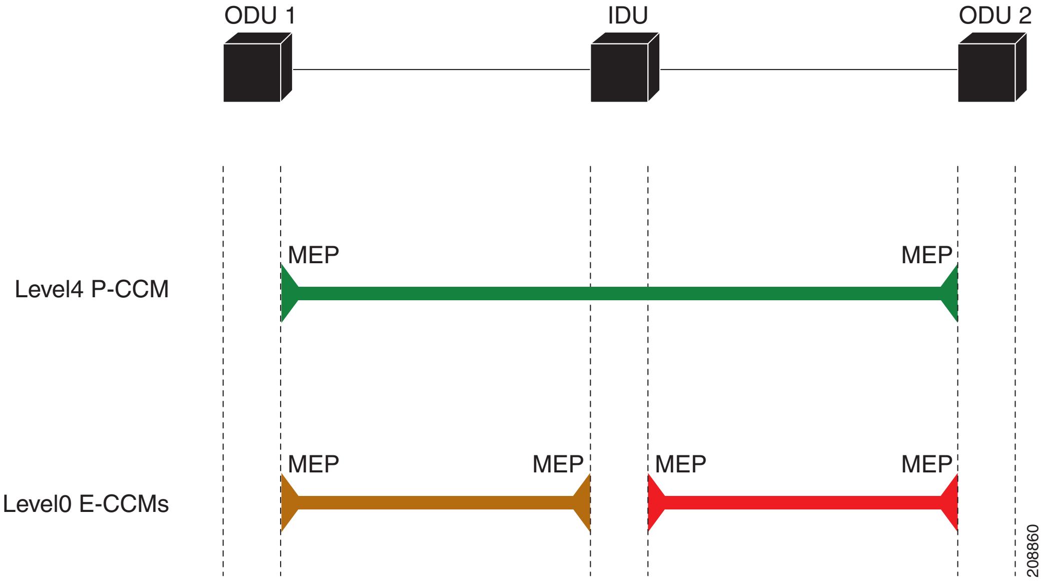
Thus, the HSBY configuration shown in Figure 15-3 consists of five separate traffic flows:

- CFM traffic between the IDU and ODU 1

- CFM traffic between the IDU and the ODU 2
- CFM traffic between ODU 1 and ODU 2. This traffic passes through IDU.
- Data traffic between the WAN and ODU 1. This traffic passes through the IDU.

Figure 15-4 provides a logical view of the maintenance associations used in this HSBY topology.

Figure 15-4 HSBY Protocol CFM Maintenance Associations



Note To prevent switching loops on the management VLAN, we recommend that you enable RSTP on the management VLAN. For more information about how to configure RSTP, see "Understanding RSTP" section on page 10-8.

Configuring Microwave 1+1 Hot Standby Protocol

The following sections describe how to configure Microwave 1+1 Hot Standby Protocol (HSBY) on the Cisco MWR 2941.

- ODU Configuration Values, page 15-47
- IDU Configuration Values, page 15-47
- Configuring HSBY, page 15-47

ODU Configuration Values

HSBY protocol specifies that some values on the ODU are configurable while others utilize fixed values. Table 15-6 summarizes the permitted values for an ODU using HSBY protocol.

Table 15-6 HSBY ODU Configuration Parameters Summary

Parameter	Default Value	Permitted Values
Short MA Name	Learned	0–65535
MPID	2	Fixed
MA VLAN-ID (E-CCM)	None	16–50

IDU Configuration Values

HSBY protocol specifies that some values on the IDU are configurable while others utilize fixed values. Table 15-7 summarizes the permitted values for an IDU using HSBY protocol.

Table 15-7 HSBY IDU Configuration Parameters Summary

Parameter	Default Value	Permitted Values
Domain Name	Null	Fixed
Domain Level	0	Fixed
Short MA Name	None	0–65535
MPID	1	Fixed
MA VLAN-ID (E-CCM)	None	1–15
CC Interval	100 ms	10 ms, 100 ms, and 1000 ms **Note** Release 15.0(1)MR does not support 10ms CC intervals.
Suspend Interval	160 seconds	80 s, 160 s, 240 s, and 320 s

Configuring HSBY

Follow these steps to configure HSBY protocol on the Cisco MWR 2941.

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**interface gigabitethernet** *slot/port*	Enters configuration for the interface connected to ODU 1. **Note** HSBY is permitted only on Gigabit Ethernet interfaces 0/0–0/5.
Step 3	**ethernet cfm mep domain** *domain-name* **mpid** *mpid* {**port** | **vlan** *vlan-id*}	Defines a CFM MEP domain for ODU 1.

	Command	Purpose
Step 4	**link-protection group** *group-number* **pccm vlan** *vlan-id*	Specifies a link protection group for ODU 1.
Step 5	**interface gigabitethernet** *slot/port*	Enters configuration for the interface connected to ODU 2
Step 6	**ethernet cfm mep domain** *domain-name* **mpid** *mpid* {**port** | **vlan** *vlan-id*}	Defines a CFM MEP domain for ODU 2.
Step 7	**link-protection group** *group-number* **pccm vlan** *vlan-id*	Specifies a link protection group for ODU 2.
Step 8	**ethernet cfm domain** *domain-name* **level** *level-id* [**direction outward**]	Configures the CFM MEP domain for ODU 1.

Step 9	**id** {*mac-address domain-number* | **dns** *dns-name* | **null**}
	Example:
	`Router(config)# id null`
		Defines a the maintenance domain identifier (MDID) for ODU 1 as null.

Step 10	**service** {*ma-name* | *ma-num* | **vlan-id** *vlan-id* | **vpn-id** *vpn-id*} [**port** | **vlan** *vlan-id* [**direction down**]]	Defines a maintenance association for ODU 1.
Step 11	**continuity-check**	Enables transmission of continuity check messages (CCMs) within the ODU 1 maintenance association.
Step 12	**continuity-check** [**interval** *time* | **loss-threshold** *threshold* | **static rmep**]	Defines a continuity-check interval for the ODU 1 maintenance association.
Step 13	**ethernet cfm domain** *domain-name* **level** *level-id* [**direction outward**]	Configures the CFM MEP domain for ODU 2.

Step 14	**id** {*mac-address domain-number* | **dns** *dns-name* | **null**}
	Example:
	`Router(config)# id null`
		Defines a the MDID for ODU 2 as null.

Step 15	**service** {*ma-name* | *ma-num* | **vlan-id** *vlan-id* | **vpn-id** *vpn-id*} [**port** | **vlan** *vlan-id* [**direction down**]]	Defines a maintenance association for ODU 2.
Step 16	**continuity-check**	Enables transmission of CCMs within the ODU 2 maintenance association.
Step 17	**continuity-check** [**interval** *time* | **loss-threshold** *threshold* | **static rmep**]	Defines a continuity-check interval for the ODU 2 maintenance association.
Step 18	**link-protection enable**	Globally enables link protection on the router.
Step 19	**link-protection group management vlan** *vlan-id*	Defines the management VLAN used for link protection.
Step 20	**ethernet cfm ieee**	Enables the Ethernet Connectivity Fault Management 802.1ag Standard (CFM IEEE) version of CFM.
Step 21	**ethernet cfm global**	Enables Ethernet connectivity fault management (CFM) globally.
Step 22	**end**	Returns to privileged EXEC mode.
Step 23	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

	Command	Purpose
Step 24	**show link-protection [detail [group** *group-number*]]	(Optional) Displays the status of configured link protection groups.
Step 25	**show link-protection statistics [interface** *interface-name slot/port*]	(Optional) Displays the counters for each link protection port.

Configuration Examples

- Ethernet OAM and CFM Configuration: Example
- CFM and ELMI Sample Configuration: Example
- HSBY Sample Configuration: Example

Ethernet OAM and CFM Configuration: Example

These are example configurations of the interworking between Ethernet OAM and CFM in a sample service provider network with a provider-edge device connected to a customer edge device at each endpoint. You must configure CFM, E-LMI, and Ethernet OAM between the customer edge and the provider edge devices.

Customer-edge device 1 (CE1) configuration:

```
Router# config t
Router(config)# interface gigabitethernet1/0/1
Router(config-if)# switchport trunk allowed vlan 10
Router(config-if)# switchport mode trunk
Router(config-if)# ethernet oam remote-loopback supported
Router(config-if)# ethernet oam
Router(config-if)# exit
```

Provider-edge device 1 (PE1) configuration:

```
Router# config t
Router(config)# interface gigabitethernet1/0/20
Router(config-if)# switchport trunk encapsulation dot1q
Router(config-if)# switchport mode trunk
Router(config-if)# ethernet cfm mip
Router(config-if)# ethernet cfm mep mpid 100 vlan 10
Router(config-if)# ethernet uni id 2004-20
Router(config-if)# ethernet oam remote-loopback supported
Router(config-if)# ethernet oam
Router(config-if)# service instance 10 ethernet BLUE
Router(config-if-srv)# ethernet lmi ce-vlan map 10
Router(config-if-srv)# exit
```

Provider-edge device 2 (PE2) configuration:

```
Router# config t
Router(config)# interface gigabitethernet1/1/20
Router(config-if)# switchport mode trunk
Router(config-if)# ethernet cfm mip
Router(config-if)# ethernet cfm mep mpid 101 vlan 10
Router(config-if)# ethernet uni id 2004-20
Router(config-if)# ethernet oam remote-loopback supported
Router(config-if)# ethernet oam
Router(config-if)# service instance 10 ethernet BLUE
Router(config-if-srv)# ethernet lmi ce-vlan map 10
Router(config-if-srv)# exit
```

Customer-edge device 2 (CE2) configuration:

```
Router# config t
Router(config)# interface gigabitethernet1/0/1
Router(config-if)# switchport trunk allowed vlan 10
Router(config-if)# switchport mode trunk
Router(config-if)# ethernet oam remote-loopback supported
Router(config-if)# ethernet oam
Router(config-if)# exit
```

These are examples of the output showing provider-edge switch port status of the configuration. Port status shows as *UP* at both switches.

PE1:

```
Router# show ethernet cfm maintenance points remote
MPID  Level Mac Address    Vlan PortState InGressPort     Age(sec) Service ID
101 * 4     0015.633f.6900 10   UP        Gi1/1/1         27       blue
```

PE2:

```
Router# show ethernet cfm maintenance points remote
MPID  Level Mac Address    Vlan PortState InGressPort     Age(sec) Service ID
100 * 4     0012.00a3.3780 10   UP        Gi1/1/1         8        blue
Total Remote MEPs: 1
```

This example shows the outputs when you start remote loopback on CE1 (or PE1). The port state on the remote PE switch shows as *Test* and the remote CE switch goes into error-disable mode.

```
Router# ethernet oam remote-loopback start interface gigabitEthernet 0/1
This is a intrusive loopback.
Therefore, while you test Ethernet OAM MAC connectivity,
you will be unable to pass traffic across that link.
Proceed with Remote Loopback? [confirm]
```

PE1:

```
Router# show ethernet cfm maintenance points remote
MPID  Level Mac Address    Vlan PortState InGressPort     Age(sec) Service ID
101 * 4     0015.633f.6900 10   UP        Gi1/1/1         27       blue
```

PE2:

```
Router# show ethernet cfm maintenance points remote
MPID  Level Mac Address    Vlan PortState InGressPort     Age(sec) Service ID
100 * 4     0012.00a3.3780 10   TEST      Gi1/1/1         8        blue
Total Remote MEPs: 1
```

In addition, if you shut down the CE1 interface that connects to PE1, the remote PE2 port will show a PortState of *Down*.

CFM and ELMI Sample Configuration: Example

The following sample configuration uses CFM and ELMI with three inward facing MEPs, two MIPs, and three maintenance domains.

Note This section provides partial configurations intended to demonstrate a specific feature.

```
!
```

```
ethernet cfm ieee
ethernet cfm global
ethernet cfm traceroute cache
ethernet cfm traceroute cache size 112
ethernet cfm domain CISCO_7
 service L7 vlan 700
  continuity-check
!
ethernet cfm domain CISCO_ENG
 service ce28 vlan 600
  continuity-check
!
ethernet cfm domain CISCO_5
 service L5 vlan 1
  continuity-check
!
ethernet lmi global

!
interface GigabitEthernet0/2
 switchport access vlan 600
 shutdown
 ethernet cfm mip vlan 600
 ethernet cfm mep domain CISCO_ENG mpid 629 vlan 600
!
interface GigabitEthernet0/3
 switchport mode trunk
 shutdown
 ethernet cfm mep domain CISCO_5 mpid 529 vlan 1
!
interface GigabitEthernet0/4
 switchport access vlan 700
 shutdown
 ethernet cfm mep domain CISCO_7 mpid 729 vlan 700
!
interface GigabitEthernet0/5
 switchport mode trunk
 ethernet cfm mip vlan 1-2,100,600,700
!
```

HSBY Sample Configuration: Example

```
!
link-protection enable
link-protection management vlan 51
link-protection group 2 pccm vlan 16

!
ethernet cfm ieee
ethernet cfm global
!
ethernet cfm domain LPG1 level 0
 id null
 service number 100 vlan 10 direction down
  continuity-check
  continuity-check interval 100ms
!
ethernet cfm domain LPG2 level 0
 id null
 service number 200 vlan 11 direction down
  continuity-check
  continuity-check interval 10ms
!
interface GigabitEthernet0/3
 ethernet cfm mep domain LPG1 mpid 1 vlan 10
   link-protection group 12
!
interface GigabitEthernet0/4
 ethernet cfm mep domain LPG2 mpid 1 vlan 11
    link-protection group 12
!
```

CHAPTER 16

Configuring Clocking and Timing

Clock synchronization is important for a variety of applications, including synchronization of radio cell towers. While legacy TDM protocols incorporate timing features, packet-switched networks such as Ethernet do not natively include these features. The Cisco MWR 2941 supports legacy TDM technologies while supporting a variety of technologies that distribute clocking information over packet-switched networks.

The following sections describe the clocking and timing features available on the Cisco MWR 2941.

- Network Clocking Overview
- Configuring Clocking and Timing
- Clocking Sample Configurations

Network Clocking Overview

Clocking is typically distributed from the core network outward to the BTS or Node B at the network edge. The Cisco MWR 2941 receives and transmits clocking information using any of the following ports:

- T1/E1
- Ethernet (GigabitEthernet and FastEthernet)
- DSL
- BITS/SYNC port
- 1PPS
- 1.544Mhz
- 2.048Mhz
- 10Mhz

The Cisco MWR 2941 supports the following clocking types:

- Precision Timing Protocol (PTP)
- Pseudowire-Based Clocking
- Synchronous Ethernet

Precision Timing Protocol (PTP)

The Cisco MWR 2941 supports the Precision Time Protocol (PTP) as defined by the IEEE 1588-2008 standard. PTP provides for accurate time synchronization on over packet-switched networks. Nodes within a PTP network can act in one of the following roles:

- Grandmaster—A device on the network physically attached to the primary time source. All other clocks are ultimately synchronized to the grandmaster clock.
- Ordinary clock—An ordinary clock is a 1588 clock with a single PTP port that can serve in one of the following roles:
  - Master mode—Distributes timing information over the network to one or more slave clocks, thus allowing the slave to synchronize its clock to the master.
  - Slave mode—Synchronizes its clock to a master clock. You can enable slave clocking on up to two interfaces simultaneously in order to connect to two different master clocks.
- Boundary clock—The device participates in selecting the best master clock and can act as the master clock if no better clocks are detected.
- Transparent clock—A device such as a switch that calculates the time it requires to forward traffic and updates the PTP time correction field to account for the delay, making the device transparent in terms of timing calculations.

Note The Cisco MWR 2941 does not currently act as a transparent clock.

Note The 1588-2008 standard defines other clocking devices that are not described here.

PTP Domains

PTP devices use a best master clock algorithm to determine the most accurate clock on a network and construct a clocking hierarchy based on the grandmaster clock. A given clocking hierarchy is called a PTP domain.

Clock Synchronization

PTP master devices periodically launch an exchange of messages with slave devices to help each slave clock recompute the offset between its clock and the master clock. Periodic clock synchronization mitigates any drift between the master and slave clocks.

PTP Redundancy

The Cisco MWR 2941 supports the multicast- and unicast-based timing as specified in the 1588-2008 standard. The Cisco MWR 2941 can use multicast routing to establish redundant paths between an external PTP client and one or more PTP multicast master clocks. The Cisco MWR 2941 functions as a multicast router only for PTP traffic and only allows multicast traffic to pass from the PTP master clocks to the PTP client (the PTP client can send unicast traffic).

When configured as a multicast PTP router, the Cisco MWR 2941 selects the best path toward a Rendezvous Point (RP) using the active routing protocol, sends a Cisco Protocol Independent Multicast (PIM) join message to the RP, and forwards PTP multicast messages to the PTP client. The Cisco MWR 2941 also supports PIM forwarding. For instructions on how to configure PTP redundancy using multicast, see Configuring PTP Redundancy, page 16-10.

Hot Standby Master Clock

The Cisco MWR 2941 supports a hot standby master clock for PTP clocking; the Cisco MWR 2941 selects the best clock source between two PTP master clocks and switches dynamically between them if the clock quality of the standby clock is greater than that of the current master clock. For instructions on how to configure a hot standby master clock, see Configuring PTP Clocking.

Hybrid Clocking

The Cisco MWR 2941 supports a hybrid clocking mode that uses clock frequency obtained from the synchronous Ethernet port while using phase (ToD or 1PPS) obtained using PTP. For instructions on how to configure hybrid clocking, see Configuring PTP Clocking.

Pseudowire-Based Clocking

Pseudowire-based clocking allows the Cisco MWR 2941 router to

- Transmit and receive clocking information over a pseudowire interface
- Receive clocking over a virtual pseudowire interface.

The Cisco MWR 2941 can transmit clocking information within packet headers (in-band) or as a separate packet stream (out-of-band).

Pseudowire-based clocking also supports adaptive clock recovery (ACR), which allows the Cisco MWR 2941 to recover clocking from the headers of a packet stream. For instructions on how to configure pseudowire-based clocking, see Configuring Clocking and Timing.

Synchronous Ethernet

Synchronous Ethernet is a timing technology that allows the Cisco MWR 2941 to transport frequency and time information over Ethernet. Because frequency and time are embedded in Ethernet packets, synchronous Ethernet must be supported by each network element in the synchronization path. Synchronous Ethernet is defined in the ITU-T G.781, G.8261, G.8262, and G.8264, Telcordia GR-253-CORE, and Telcordia GR-1244-CORE standards.

You can use synchronous Ethernet in conjunction with an external timing technology such as GPS to synchronize timing across the network. For instructions on how to configure synchronous Ethernet, see Configuring Clocking and Timing.

Synchronous Ethernet ESMC and SSM

The Cisco MWR 2941 supports Ethernet Synchronization Message Channel (ESMC) and Synchronization Status Message (SSM) to provide clock synchronization on Synchronous Ethernet. For more information about Ethernet ESMC and SSM, see Chapter 17, “Configuring Synchronous Ethernet ESMC and SSM.”

Configuring Clocking and Timing

The Cisco MWR 2941 supports the following network clocking types:

- Precision Time Protocol (PTP)—Clocking and clock recovery based on the IEEE 1588-2008 standard; allows the Cisco MWR 2941 router to receive clocking from another PTP-enabled device or provide clocking to a PTP-enabled device. To configure PTP clocking, see Configuring PTP Clocking. If you want to enable PTP redundancy, you must also complete the steps in the Configuring PTP Redundancy section.
- Pseudowire-based clocking—Allows the Cisco MWR 2941 router to use clocking using a pseudowire or virtual pseudowire interface. Pseudowire-based clocking supports adaptive clock recovery, which allows the Cisco MWR 2941 to recover clocking from the headers of a packet stream. To configure pseudowire-based clocking, see Configuring Pseudowire-Based Clocking with Adaptive Clock Recovery.
- Synchronous Ethernet—Allows the network to transport frequency and time information over Ethernet. To configure synchronous Ethernet, see Configuring Synchronous Ethernet.
- Verifying Clock Settings—To verify a clocking configuration, see Verifying Clock-Related Settings.

Note The Cisco MWR 2941 does not support the use of PTP and PWE-based clocking at the same time.

Configuring PTP Clocking

This section describes how to configure PTP-based clocking on the Cisco MWR 2941. For more information about the PTP commands, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.

Note The settings shown in this section are an example only; you must determine the appropriate PTP settings based upon your network clocking design.

Note The configuration sections describing the 1PPS and 10Mhz timing ports only apply to the Cisco MWR 2941-DC-A; the Cisco MWR-DC router does not have these timing ports.

Configuring Global PTP Settings

Enter the following commands to configure the global PTP settings:

	Command	Purpose
Step 1	**enable** **Example:** Router> enable	Enables privileged EXEC mode. - Enter your password if prompted.
Step 2	**configure terminal** **Example:** Router# configure terminal	Enters global configuration mode.
Step 3	**ptp mode ordinary** **Example:** Router(config)# **ptp mode ordinary**	Specifies the PTP mode; you can configure **ordinary** or **boundary** clock mode.
Step 4	Router(config)# **ptp priority1 128**	Configures the preference level for a clock; slave devices use the priority1 value when selecting a master clock.
Step 5	Router(config)# **ptp priority2 128**	Sets a secondary preference level for a clock; slave devices use the priority2 value when selecting a master clock.
Step 6	Router(config)# **ptp domain 6**	Specifies the PTP domain number that the router uses. PTP domains allow you to use multiple independent PTP clocking subdomains on a single network.
Step 7	**exit** **Example:** Router(config)# **exit** Router#	Exits configuration mode.

Note If you want to use PTP redundancy, you must also complete the tasks in the Configuring PTP Redundancy, page 16-10.

Configuring the PTP Interface Settings

Table 16-1 summarizes the PTP interface commands that you can use on the Cisco MWR 2941.

Note If you want to use PTP redundancy, you must also complete the tasks in the Configuring PTP Redundancy, page 16-10.

Table 16-1 PTP Interface Commands

Command	Purpose
ptp announce	Sets interval and timeout values for PTP announcement packets.
ptp boundary	Sets the interface in boundary clock mode; you can specify the interface to use multicast or unicast negotiation.
ptp clock-destination	Specifies the IP address of a clock destination. This command applies only when the router is in PTP master unicast mode.
ptp clock-source	Specifies the IP address of the clock source. This command applies only when the router is in PTP slave mode.
ptp delay-req interval	Specifies the delay request interval, the time recommended to member devices to send delay request messages when an interface is in PTP master mode.
ptp delay-req unicast	Configures the Cisco MWR 2941 to send unicast PTP delay request messages while in multicast mode. This command helps reduce unnecessary PTP delay request traffic.
ptp enable	Enables PTP mode on an interface. You can enable PTP slave mode on two VLAN interfaces simultaneously.
ptp master	Sets an interface in master clock mode for PTP clocking. **Note** PTP master mode is intended only for trial use and is not for use in a production network.
ptp slave	Sets an interface to slave clock mode for PTP clocking. You can enable slave mode on two interfaces simultaneously to connect to two different master clocks.
ptp sync	Specifies the interval that the router uses to send PTP synchronization messages.

The following examples demonstrate how to use these commands to configure each of the PTP modes. Use the appropriate section based on the PTP mode that you want to configure on the Cisco MWR 2941.

- PTP multicast master mode—Sets the Cisco MWR 2941 to act as the master PTP clock. Multicast specifies that the router sends PTP messages to all the slaves listening on the PTP multicast group.

Note PTP master mode is intended for trial use only and is not for use in a production network.

```
Router(config)# interface Vlan10
Router(config-if)# ip address 192.168.52.38 255.255.255.0
Router(config-if)# ip igmp join-group 224.0.1.129
Router(config-if)# ptp announce interval 0
Router(config-if)# ptp sync interval -6
Router(config-if)# ptp delay-req interval -4
Router(config-if)# ptp master multicast
Router(config-if)# ptp enable
```

- PTP multicast slave mode—Sets the Cisco MWR 2941 to receive clocking from a PTP master device in multicast mode.

```
Router(config)# interface Vlan10
Router(config-if)# ip address 192.168.52.38 255.255.255.0
Router(config-if)# ip igmp join-group 224.0.1.129
Router(config-if)# ptp announce interval 0
Router(config-if)# ptp sync interval -6
Router(config-if)# ptp delay-req interval -4
Router(config-if)# ptp slave multicast
Router(config-if)# ptp enable
```

- PTP multicast slave mode (with hybrid clocking)—Sets the Cisco MWR 2941 to receive phase from a PTP master device in multicast mode while using clock frequency obtained from the synchronous Ethernet port.

```
Router(config)# interface Vlan10
Router(config-if)# ip address 192.168.52.38 255.255.255.0
Router(config-if)# ip igmp join-group 224.0.1.129
Router(config-if)# ptp announce interval 0
Router(config-if)# ptp sync interval -6
Router(config-if)# ptp delay-req interval -4
Router(config-if)# ptp slave multicast hybrid
Router(config-if)# ptp enable
```

Note You can use the **ptp delay-req unicast** command to set the Cisco MWR 2941 to send unicast PTP Delay_Req messages while in multicast mode in order to eliminate unnecessary multicast traffic. For more information about this command, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.

- PTP unicast master mode—Sets the Cisco MWR 2941 to act as the master PTP clock. Unicast specifies that the router sends PTP messages to a single slave host.

```
Router(config)# interface Vlan2
Router(config-if)# ip address 192.168.52.38 255.255.255.0
Router(config-if)# ptp announce interval 0
Router(config-if)# ptp sync interval -6
Router(config-if)# ptp delay-req interval -4
Router(config-if)# ptp master unicast
Router(config-if)# ptp clock-destination 192.168.52.201
Router(config-if)# ptp enable
```

- PTP unicast master mode (with negotiation enabled)—Sets the Cisco MWR 2941 to send clocking to a single PTP slave device; the router allows the slave devices to negotiate their master clock device. When in the router is in PTP unicast master mode, you can specify up to 128 PTP clock destination devices.

Note If you set the router to PTP master unicast mode with negotiation, you do not specify PTP clock destinations because the router negotiates to determine the IP addresses of the PTP slave devices.

Note We recommend that you determine the number of destination devices to assign to a master clock based on traffic rates and available bandwidth.

```
Router(config)# interface Vlan2
Router(config-if)# ip address 192.168.52.38 255.255.255.0
Router(config-if)# ptp announce interval 0
Router(config-if)# ptp sync interval -6
Router(config-if)# ptp delay-req interval -4
Router(config-if)# ptp master unicast negotiation
Router(config-if)# ptp enable
```

- PTP unicast slave mode—Sets the Cisco MWR 2941 to receive clocking from a single PTP master device.

```
Router(config)# interface Vlan2
Router(config-if)# ip address 192.168.52.38 255.255.255.0
Router(config-if)# ptp announce interval 3
Router(config-if)# ptp announce timeout 2
Router(config-if)# ptp sync interval -6
Router(config-if)# ptp delay-req interval -4
Router(config-if)# ptp slave unicast
Router(config-if)# ptp clock-source 192.168.52.10
Router(config-if)# ptp enable
```

- PTP unicast slave mode (with negotiation enabled)—Sets the Cisco MWR 2941 to receive clocking from a PTP master device; the router negotiates between up to 128 PTP master devices.

```
Router(config)# interface Vlan2
Router(config-if)# ip address 192.168.52.38 255.255.255.0
Router(config-if)# ptp announce interval 3
Router(config-if)# ptp announce timeout 2
Router(config-if)# ptp sync interval -6
Router(config-if)# ptp delay-req interval -4
Router(config-if)# ptp slave unicast negotiation
Router(config-if)# ptp clock-source 192.168.52.10
Router(config-if)# ptp enable
```

- PTP unicast slave mode (with hybrid clocking)—Sets the Cisco MWR 2941 to receive phase (ToD or 1PPS) from a single PTP master device while using clock frequency obtained from the synchronous Ethernet port.

```
Router(config)# interface Vlan2
Router(config-if)# ip address 192.168.52.38 255.255.255.0
Router(config-if)# ptp announce interval 3
Router(config-if)# ptp announce timeout 2
Router(config-if)# ptp sync interval -6
Router(config-if)# ptp delay-req interval -4
Router(config-if)# ptp slave unicast negotiation hybrid
Router(config-if)# ptp clock-source 192.168.52.10
Router(config-if)# ptp enable
```

- PTP unicast slave mode (with hot standby master clock)—Sets the Cisco MWR 2941 to receive clocking from a single PTP master device and enables a standby master clock. When you enable a standby master clock, the Cisco MWR 2941 selects the best clock source between two PTP master clocks and switches dynamically between them if the clock quality of the standby clock is greater than that of the current master clock. If you define a standby master clock, both clock sources must be in the same VLAN. Setting a standby master clock in unicast mode is optional.

```
Router(config)# interface Vlan2
Router(config-if)# ip address 192.168.52.38 255.255.255.0
Router(config-if)# ptp announce interval 3
Router(config-if)# ptp announce timeout 2
Router(config-if)# ptp sync interval -6
Router(config-if)# ptp delay-req interval -4
Router(config-if)# ptp slave unicast negotiation hybrid
Router(config-if)# ptp clock-source 192.168.52.10
Router(config-if)# ptp clock-source 192.168.52.150
Router(config-if)# ptp enable
```

Enabling PTP on Multiple VLANs

You can enable PTP on up to three VLANs at a time. The following restrictions apply:

- All PTP-enabled VLANs must use PTP master or PTP slave; you cannot configure PTP master and PTP slave VLANs at the same time.
- All PTP-enabled VLANs must use multicast or unicast, but not both.

Configure the Global Network Clock

Use the **network-clock-select** command to configure clock selection for the entire network.

- If you configured the router for PTP master mode, set one or more external clock sources using the **network-clock-select** command with the synchronous Ethernet, bits, E1, or T1 interface parameters:

```
Router(config)# network-clock-select 1 BITS
Router(config)# network-clock-select 2 SYNC 0
Router(config)# network-clock-select 3 E1 0/0
```

- If you configured the router for PTP slave mode, enter the following commands:

```
Router(config)# network-clock-select 1 PACKET-TIMING
Router(config)# network-clock-select hold-timeout 900
```

Note The **network-clock-select hold-timeout** command is optional; the minimum recommended value in the slave mode is 900 seconds or 15 minutes. For more information about this command, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.

Configuring PTP Input and Output

You can use the 1pps and 10Mhz timing ports on the Cisco MWR 2941-DC-A to do the following:

- Provide or receive 1PPS time of day messages
- Provide output clocking at 10Mhz, 2.048Mhz, and 1.544Mhz
- Receive input clocking at 10Mhz, 2.048Mhz, and 1.544Mhz

Note This section applies only to the Cisco MWR 2941-DC-A.

The following section describes how to configure time of day messages, output clocking, and input clocking.

- If you want to configure PTP input clocking using the 10Mhz timing port, complete the following steps:
  - Use the **ptp input** command to enable PTP input clocking at 10Mhz, 2.048Mhz, or 1.544Mhz.

    ```
    Router(config)# ptp input 10M
    ```

  - Use the **network-clock-select** command to select the port to use for input clocking.

    ```
    Router(config)# network-clock-select 10 10M
    ```

  Input clocking applies when the router is in PTP master mode.

- To configure output clocking using the 10Mhz timing port, use the **ptp output** command to specify 10Mhz, 2.048Mhz, or 1.544Mhz output. Use this command when the router is in PTP slave mode.

```
Router(config)# ptp output 2.048M
```

- To configure the router to send time of day messages using the 1PPS port, use the **ptp 1pps** command. Use the **input** or **output** parameters to specify the direction and the **pulse-width** parameter to specify the pulse width value.

```
Router(config)# ptp output 1pps pulse-width 2000 ms
```

Note The **pulse-width** parameter is only supported for PTP output.

- To configure the time of day message format, use the **ptp tod** command.

```
Router(config)# ptp tod ubx delay 400
```

- To configure the router to periodically update the system calendar with PTP clock time, use the **ptp update-calendar** command.

```
Rounter(config)# ptp update-calendar
```

Note To see configuration examples for input and output timing, see Clocking Sample Configurations.

Configuring PTP Redundancy

The Cisco MWR 2941 supports PTP redundancy by allowing the Cisco MWR 2941 to act as a multicast router for PTP traffic between an external PTP client and one or more multicast PTP master clocks. Follow these steps to configure PTP redundancy on the Cisco MWR 2941.

Note Special IP routing protocols such as OSPF and IS-IS will be recognized based on protocol type and forwarded to the PPC host. This a basic routing requirement for these protocols and is not related to PTP redundancy.

	Command	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)# ip multicast-routing`	Enables multicast routing.

	Command	Purpose
Step 4	Router(config)# **interface vlan 100** Router(config-if)# **description PTP client interface** Router(config-if)# **ip address 10.1.1.1 255.255.255.252**	Enter these commands to configure a VLAN for the PTP client.
Step 5	Router(config-if)# **ip pim sparse-dense-mode**	Enables Protocol Independent Multicast (PIM) on the VLAN interface. You can specify the interface to use sparse, dense, or sparse-dense mode. For more information about the **ip pim** command, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.
Step 6	Router(config)# **interface vlan 401** Router(config-if)# **description Network interface sourcing PTP multicast** Router(config-if)# **ip address 10.1.2.1 255.255.255.0**	Enter the following commands to configure a VLAN for the first multicast PTP clock source.
Step 7	Router(config-if)# **ip pim sparse-dense-mode**	Enables Protocol Independent Multicast (PIM) on the VLAN interface. You can specify the interface to use sparse, dense, or sparse-dense mode. For more information about the **ip pim** command, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.
Step 8	Router(config)# **interface vlan 402** Router(config-if)# **description Network interface sourcing PTP multicast** Router(config-if)# **ip address 10.1.3.1 255.255.255.0**	Configure a VLAN for the second multicast PTP clock source.
Step 9	Router(config-if)# **ip pim sparse-dense-mode**	Enables Protocol Independent Multicast (PIM) on the VLAN interface. You can specify the interface to use sparse, dense, or sparse-dense mode. For more information about the **ip pim** command, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.
Step 10	Router(config)# **interface gigabitethernet 0/5** Router(config-if)# **description physical interface to PTP client** Router(config-if)# **switchport access vlan 100**	Configures the gigabit Ethernet interface connected to the PTP client.
Step 11	Router(config)# **interface gigabitethernet 0/0** Router(config-if)# **description physical interface to PTP multicast source** Router(config-if)# **switchport trunk allowed vlan 1,2,401,1002-1005** Router(config-if)# **switchport mode trunk**	Configures the first gigabit Ethernet interface connected to the multicast PTP clock source.

	Command	Purpose
Step 12	Router(config)# **interface gigabitethernet 0/1** Router(config-if)# **description physical interface to PTP multicast source** Router(config-if)# **switchport trunk allowed vlan 1,2,402,1002-1005** Router(config-if)# **switchport mode trunk**	Configures the second gigabit Ethernet interface connected to the multicast PTP clock source.
Step 13	Router(config-if)# **exit** Router(config)#	Exits the gigabitEthernet interface.
Step 14	Router(config)# **ip pim rp-address 10.2.1.1 5 override**	If you need to statically configure a PIM rendezvous point (RP) for a multicast group, use the **ip pim rp-address** command in global configuration mode.
Step 15	Router(config)# **access-list 5 permit 224.0.1.129**	Creates an access list entry to allow PTP traffic from the router's PTP multicast address.
Step 16	Router(config)# **exit** Router#	Exits configuration mode.
Step 17	Router# **show ip mroute**	Use the **show ip mroute** command to verify your configuration.
Step 18	**exit** **Example:** Router(config)# exit Router#	Exits configuration mode.

Note To view a sample configuration of multicast/PTP redundancy, see Clocking Sample Configurations, page 16-17.

Configuring Pseudowire-Based Clocking with Adaptive Clock Recovery

The Cisco MWR 2941 supports the following adaptive clock recovery modes:

- In-band master mode—The Cisco MWR 2941 provides clocking to slave devices using the headers in a packet stream. To configure this clocking mode, see Configuring In-Band Master Mode.
- In-band slave mode—The Cisco MWR 2941 receives clocking from a master clock using the headers from a packet stream. To configure this clocking mode, see Configuring In-Band Slave Mode.
- Out-of-band slave mode—The Cisco MWR 2941 receives clocking from a master clock using dedicated packets for timing. To configure this clocking mode, see Configuring Out-of-Band Slave Mode.

Note The Cisco MWR 2941 currently does not support out-of-band master mode.

Configuring In-Band Master Mode

Use the following steps to configure in-band master mode.

	Command	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	The following example shows how to configure SAToP.
	`Router(config)# controller e1 0/0`
	`Router(config-controller)# clock source internal`
	`Router(config-controller)# cem-group 0 unframed`
	The following example shows how to configure CES.
	`Router(config)# controller e1 0/0`
	`Router(config-controller)# clock source internal`
	`Router(config-controller)# cem-group 3 timeslots 1-31`
		To configure in-band ACR master mode, you must configure Structure-agnostic TDM over Packet (SAToP) or Circuit Emulation Service (CES).

Step 4	`Router(config)# interface Loopback`
	`Router(config-if)# ip address 10.88.88.99 255.255.255.255`
		Configures the loopback interface.

Step 5	`Router(config)# interface Vlan1`
	`Router(config-if)# ip address 192.168.52.2 255.255.255.0`
	`Router(config-if)# no ptp enable`
	`Router(config-if)# mpls ip`
		Configures the VLAN interface.

Step 6	`Router(config)# mpls ldp router-id Loopback0 force`	Configures MPLS.

Step 7	`Router(config)# interface cem 0/1`
	`Router(config-if)# cem 0`
	`Router(config-if-cem)# xconnect 10.10.10.2 7600 encap mpls`
		Configures the CEM interface.

Step 8	`Router(config)# network-clock-select 1 BITS`	Sets one or more external clock sources using the **synce**, **bits**, **e1**, or **t1** interface parameters:

Step 9	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Configuring In-Band Slave Mode

Use the following steps to configure in-band slave mode.

	Command	Purpose
Step 1	**enable** **Example:** Router> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Router# configure terminal	Enters global configuration mode.
Step 3	The following example shows how to configure SAToP. Router(config)# **controller e1 0/0** Router(config-controller)# **clock source internal** Router(config-controller)# **cem-group 0 unframed** The following example shows how to configure CES. Router(config)# **controller e1 0/0** Router(config-controller)# **clock source internal** Router(config-controller)# cem-group 3 timeslots 1-31	To configure in-band ACR slave mode, you must configure Structure-agnostic TDM over Packet (SAToP) or Circuit Emulation Service (CES).
Step 4	Router(config)# **interface Loopback** Router(config-if)# **ip address 10.88.88.99 255.255.255.255**	Configures the loopback interface.
Step 5	Router(config)# **interface Vlan1** Router(config-if)# **ip address 192.168.52.10.2 255.255.255.0** Router(config-if)# **no ptp enable** Router(config-if)# **mpls ip**	Configures the VLAN interface.
Step 6	Router(config)# **mpls ldp router-id Loopback0 force**	Configures MPLS.
Step 7	Router(config)# **interface cem 0/0** Router(config-if)# **cem 0** Router(config-if-cem)# **xconnect 10.10.10.2 7600 encap mpls**	Configures the CEM interface.
Step 8	Router(config)# **recovered-clock recovered adaptive cem 0 0 0**	Configures adaptive clock recovery using a circuit emulation (CEM) interface.

	Command	Purpose

Step 9	`Router(config)#` **`network-clock-select 1 PACKET-TIMING`**
	`Router(config)#` **`network-clock-select hold-timeout 900`**
		Configures the network clock:

Step 10	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Configuring Out-of-Band Slave Mode

Use the following steps to configure out-of-band slave mode.

Note When configuring out-of-band clocking, verify that the edge router (such as the Cisco 7600 Series Router) has matching settings for out-of-band clocking.

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)#` **`recovered-clock slave`**	Configures clock recovery in slave mode:

Step 4	`Router(config)#` **`interface Loopback`**
	`Router(config-if)#` **`ip address 10.88.88.99 255.255.255.255`**
		Configures the loopback interface.

Step 5	`Router(config)#` **`interface Vlan1`**
	`Router(config-if)#` **`ip address 192.168.52.10.2 255.255.255.0`**
	`Router(config-if)#` **`no ptp enable`**
	`Router(config-if)#` **`mpls ip`**
		Configures the VLAN interface.

Step 6	`Router(config)#` **`mpls ldp router-id Loopback0 force`**	Configures MPLS.

Step 7	`Router(config)#` **`interface virtual-cem 0/24`**
	`Router(config-if)#` **`payload-size 486`**
	`Router(config-if)#` **`cem 0`**
	`Router(config-if-cem)#` **`xconnect 10.10.10.2 7600 encap mpls`**
		Configures the CEM interface.
		Note The Cisco MWR 2941 only supports a payload size of 486 (625 packets per second) or 243 (1250 packets per second). This value affects the payload size only and does not alter the packet size, which is constant regardless of payload value.

	Command	Purpose

Step 8	`Router(config)#`
	`network-clock-select 1 PACKET-TIMING`
	`Router(config)#`
	`network-clock-select hold-timeout 900`
		Configures the network clock.

Step 9	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Configuring Synchronous Ethernet

The following sections describe how to configure synchronous Ethernet timing on the Cisco MWR 2941.

Configuring an External Clock Source

To configure an external clock source using Synchronous Ethernet, use the **network-clock select** command.

```
Router(config)# network-clock-select 2 SYNC 0
```

Configuring Synchronous Ethernet ESMC and SSM

For instructions on how to configure synchronous Ethernet Synchronization Message Channel (ESMC) and Synchronization Status Message (SSM), see Chapter 17, "Configuring Synchronous Ethernet ESMC and SSM."

Verifying Clock-Related Settings

Use the following commands to verify the clock settings:

- **show network-clocks**—Displays information about the network clocks
- **show controller**—Displays the status of the controller, including clocking information.
- **show ptp clock**—Displays ptp clock information
- **show ptp foreign-master-record**—Displays PTP foreign master records
- **show ptp parent**—Displays PTP parent properties
- **show ptp port**—Displays PTP port properties
- **show ptp time-property**—Displays PTP clock time properties
- **show interface virtual-cem 0/24**—Displays the status of the CEM interface
- **show cem circuit**—Displays information about the CEM circuit
- **show platform hardware**—Displays the status of hardware devices on the Cisco MWR 2941.

- **show platform hardware rtm**—Displays the current status of the TOP module

For more information about these commands, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.

Clocking Sample Configurations

The following sections show a sample configurations for PTP. For more information about how to configure PTP, see Chapter 16, "Configuring Clocking and Timing."

- PTP Slave Mode with Redundancy
- PTP Redundancy
- PTP Boundary Clock
- PTP with Multiple VLANs
- PTP Hybrid Mode
- PTP Hot Standby Master Clock
- PTP Input Timing
- PTP Output Timing

PTP Slave Mode with Redundancy

The following configuration implements PTP slave mode and PTP redundancy.

```
!
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname MWR_2
!
boot-start-marker
boot system flash mwr2941-ipran-mz.ptp
boot-end-marker
!
card type e1 0 0
enable secret 5 mysecret
!
no aaa new-model
ip source-route
!
!
ip cef
no ip domain lookup
ip multicast-routing
ptp mode ordinary
ptp priority1 128
ptp priority2 128
ptp domain 0
multilink bundle-name authenticated
!
mpls label protocol ldp
!
```

```
!
ipran-mib snmp-access outOfBand
archive
 log config
  hidekeys
!
!
controller E1 0/0
 clock source internal
 cem-group 0 unframed
 description TDM Shorthaul for SAToP PW
!
controller E1 0/1
 framing NO-CRC4
 clock source internal
 cem-group 0 timeslots 1-31
 description TDM Shorthaul for CESoPSN PW
!
controller E1 0/2
 clock source internal
!
controller E1 0/3
 clock source internal
!
controller E1 0/4
 clock source line
!
controller E1 0/5
 clock source line
!
controller E1 0/6
 clock source line
!
controller E1 0/7
 clock source line
!
controller E1 0/8
 clock source internal
 ima-group 0 scrambling-payload
 description ATM Shorthaul for ATMoMPLS PW
!
controller E1 0/9
 clock source internal
 ima-group 0 scrambling-payload
 description ATM Shorthaul for ATMoMPLS PW
!
controller E1 0/10
 clock source internal
 ima-group 0 scrambling-payload
 description ATM Shorthaul for ATMoMPLS PW
!
controller E1 0/11
 clock source internal
!
controller E1 0/12
 clock source internal
!
controller E1 0/13
 clock source internal
!
controller E1 0/14
 clock source internal
!
controller E1 0/15
```

```
 clock source internal
!
controller BITS
  applique E1
!
!
pseudowire-class My_MPLS
 encapsulation mpls
 sequencing both
!
!
interface Loopback0
 ip address 10.1.1.22 255.255.255.255
!
interface GigabitEthernet0/0
 switchport access vlan 11
!
interface GigabitEthernet0/1
 switchport access vlan 12
!
interface GigabitEthernet0/2
 switchport access vlan 30
!
interface GigabitEthernet0/3
 shutdown
!
interface GigabitEthernet0/4
 switchport mode trunk
 shutdown
!
interface GigabitEthernet0/5
 switchport access vlan 5
 duplex full
 speed 1000
!
interface CEM0/0
 description SAToP PW
 no ip address
 cem 0
  xconnect 10.10.10.36 5200 encapsulation mpls
 !
!
interface CEM0/1
 description CESoPSN PW
 no ip address
 cem 0
  xconnect 10.10.10.36 5201 encapsulation mpls
 !
!
interface ATM0/IMA0
 description ATMoMPLS N:1 VCC Mode (where N=1)
 no ip address
 ima group-id 0
 atm bandwidth dynamic
 no atm ilmi-keepalive
 pvc 1/32 l2transport
  encapsulation aal5
  xconnect 10.10.10.36 5232 encapsulation mpls
 !
 pvc 1/36 l2transport
  encapsulation aal0
  xconnect 10.10.10.36 5236 encapsulation mpls
 !
 pvc 1/37 l2transport
```

```
  encapsulation aal0
  xconnect 10.10.10.36 5237 encapsulation mpls
 !
 pvc 1/38 l2transport
  encapsulation aal0
  xconnect 10.10.10.36 5238 encapsulation mpls
 !
 pvc 1/39 l2transport
  encapsulation aal0
  xconnect 10.10.10.36 5239 encapsulation mpls
 !
!
interface Vlan1
 no ip address
 shutdown
 no ptp enable
!
interface Vlan3
 description 7600/2941 MPLS Backhaul VLAN
 ip address 192.22.2.2 255.255.255.0
 ip pim sparse-mode
 ptp sync interval -6
 ptp delay-req interval -4
 ptp slave multicast
 ptp enable
 mpls ip
!
interface Vlan5
 ip address 192.18.75.38 255.255.255.0
 no ptp enable
!
interface Vlan11
 description Link to 7600-PE1
 ip address 10.100.11.2 255.255.255.252
 ip pim sparse-mode
 ip ospf 1 area 0
 no ptp enable
 mpls ip
!
interface Vlan12
 description Link to 7600-PE2
 ip address 10.100.12.2 255.255.255.252
 ip pim sparse-mode
 ip igmp join-group 224.0.1.129 source 10.100.2.2
 ip igmp join-group 224.0.1.129 source 10.100.3.2
 ip ospf 1 area 0
 no ptp enable
 mpls ip
!
interface Vlan30
 description Link to PTP client
 ip address 10.100.30.1 255.255.255.0
 ip pim sparse-mode
 no ptp enable
!
router ospf 1
 router-id 10.1.1.22
 log-adjacency-changes
 redistribute connected subnets
 network 10.1.1.22 0.0.0.0 area 0
 network 10.1.11.0 0.0.0.3 area 0
 network 10.1.12.0 0.0.0.3 area 0
 network 10.100.30.0 0.0.0.255 area 0
!
```

```
ip forward-protocol nd
ip route 0.0.0.0 0.0.0.0 172.18.75.1
ip route 10.1.1.201 255.255.255.255 10.100.11.1
ip route 10.1.1.202 255.255.255.255 10.100.12.1
!
!
ip http server
ip pim rp-address 10.2.1.1 5 override
!
access-list 5 permit 224.0.1.129
snmp-server community public RO 1
snmp-server ifindex persist
snmp-server trap link ietf
no snmp-server sparse-tables
snmp-server queue-limit notification-host 100
snmp-server enable traps snmp linkdown linkup coldstart warmstart
snmp-server enable traps cpu threshold
snmp-server enable traps syslog
snmp-server enable traps ipran
snmp-server host 10.10.10.10 version 2c V2C
!
!
!
mpls ldp router-id Loopback0 force
!
control-plane
!
!
!
!
!
!
line con 0
 logging synchronous
 no modem enable
line aux 0
line vty 0 4
 password mypassword
 login
!
exception data-corruption buffer truncate
ntp clock-period 17180198
ntp peer 10.81.254.131
network-clock-select hold-timeout 600
network-clock-select mode nonrevert
network-clock-select 1 PACKET-TIMING
end
```

PTP Redundancy

The following configurations use PTP with PTP redundancy.

Note This section provides partial configurations intended to demonstrate a specific feature.

MWR_A

```
!
interface Loopback0
ip address 6.6.6.3 255.255.255.255
```

```
end
!
interface GigabitEthernet0/0
switchport access vlan 10
!
interface GigabitEthernet0/1
switchport access vlan 5
!
interface Vlan5
ip address 5.5.5.2 255.255.255.0
ip router isis
ip pim sparse-mode
no ptp enable
!
interface Vlan10
ip address 10.10.10.2 255.255.255.0
ip router isis
ip pim sparse-mode
no ptp enable
!
router isis
net 49.0001.1720.1600.3003.00
passive-interface Loopback0
!
ip pim rp-address 6.6.6.1 override
!
```

MWR_B

```
!
interface Loopback0
ip address 6.6.6.2 255.255.255.255
ip pim sparse-mode
end
!
interface GigabitEthernet0/0
switchport access vlan 10
!
interface GigabitEthernet0/4
switchport access vlan 4
load-interval 30
!
!
interface Vlan4
ip address 7.7.7.2 255.255.255.0
ip router isis
ip pim sparse-mode
no ptp enable
!
!
interface Vlan10
ip address 10.10.10.1 255.255.255.0
ip router isis
ip pim sparse-mode
no ptp enable
!
router isis
net 49.0001.1720.1600.9009.00
passive-interface Loopback0
!
ip pim rp-address 6.6.6.1 override
```

PTP Boundary Clock

The following configurations show how to use PTP boundary clock:

Note This section provides partial configurations intended to demonstrate a specific feature.

Boundary Node

```
ptp mode boundary
ptp priority1 128
ptp priority2 128
ptp domain 1

interface Vlan1
 ip address 192.168.1.2 255.255.255.0
 ptp announce interval 3
 ptp announce timeout 2
 ptp sync interval -4
 ptp delay-req interval -4
 ptp slave unicast
 ptp clock-source 192.168.1.1
 ptp enable

interface Vlan2
 ip address 172.18.52.38 255.255.255.0
 ip igmp join-group 224.0.1.129
 ptp announce interval 0
 ptp sync interval -4
 ptp delay-req interval -4
 ptp master unicast negotiation
 ptp enable

network-clock-select 1 PACKET_TIMING
```

Multicast Boundary Clock

```
ptp mode boundary
ptp priority1 128
ptp priority2 128
ptp domain 1

interface Vlan1
 ip address 192.168.1.2 255.255.255.0
 ptp announce interval 3
 ptp announce timeout 2
 ptp sync interval -4
 ptp delay-req interval -4
 ptp boundary multicast
 ptp enable

interface Vlan2
 ip address 172.18.52.38 255.255.255.0
 ip igmp join-group 224.0.1.129
 ptp announce interval 0
 ptp sync interval -4
 ptp delay-req interval -4
 ptp boundary multicast
 ptp enable

network-clock-select 1 PACKET_TIMING
```

Unicast Boundary Clock

```
ptp mode boundary
ptp priority1 128
ptp priority2 128
ptp domain 1

interface Vlan1
 ip address 192.168.1.2 255.255.255.0
 ptp announce interval 3
 ptp announce timeout 2
 ptp sync interval -4
 ptp delay-req interval -4
 ptp boundary unicast-negotiation
 ptp clock-source 192.168.1.1
 ptp enable

interface Vlan2
 ip address 172.18.52.38 255.255.255.0
 ptp announce interval 0
 ptp sync interval -4
 ptp delay-req interval -4
 ptp boundary unicast-negotiation
 ptp clock-source 172.18.52.39
 ptp enable

network-clock-select 1 PACKET_TIMING
```

PTP with Multiple VLANs

The configuration in this section consists of three Cisco MWR 2941 routers:

- MWR A—Router acting as a PTP master clock on a single VLAN
- MWR B—Router acting as a PTP master clock on a single VLAN
- MWR C—A router acting as a PTP slave clock and receiving clocking on two VLANs

Note This section provides partial configurations intended to demonstrate a specific feature.

MWR A

```
ptp mode ordinary
ptp priority1 128
ptp priority2 128
ptp domain 0

Vlan $vlan1

interface Vlan $vlan1
 ip address 192.168.10.1 255.255.255.0
 ptp announce interval 0
 ptp announce timeout 10
 ptp sync interval -6
 ptp delay-req interval -4
 ptp master unicast
 ptp clock-destination 192.168.10.2
 ptp enable
```

MWR B

```
ptp mode ordinary
ptp priority1 128
ptp priority2 128
ptp domain 0

Vlan $vlan2

interface Vlan $vlan2
 ip address 192.168.20.1 255.255.255.0
 ptp announce interval 0
 ptp announce timeout 10
 ptp sync interval -6
 ptp delay-req interval -4
 ptp master unicast
 ptp clock-destination 192.168.20.2
 ptp enable
```

MWR C

```
ptp mode ordinary
ptp priority1 128
ptp priority2 128
ptp domain 0

Vlan $vlan1
Vlan $vlan2

interface Vlan $vlan1
 ip address 192.168.10.2 255.255.255.0
 ptp announce interval 0
 ptp announce timeout 10
 ptp sync interval -6
 ptp delay-req interval -4
 ptp slave unicast
 ptp clock-source 192.168.10.1
 ptp enable
!
interface Vlan $vlan2
 ip address 192.168.20.2 255.255.255.0
 ptp announce interval 0
 ptp announce timeout 10
 ptp sync interval -6
 ptp delay-req interval -4
 ptp slave unicast
 ptp clock-source 192.168.20.1
 ptp enable

network-clock-select 1 PACKET-TIMING
```

PTP Hybrid Mode

The following section shows a sample PTP configuration that uses hybrid mode. For more information about how to configure PTP hybrid mode, see "Hybrid Clocking" section on page 16-3.

Note This section provides a partial configuration intended to demonstrate a specific feature.

```
ptp mode ordinary
ptp priority1 128
ptp priority2 128
ptp domain 1

interface Vlan1
 ip address 192.168.1.2 255.255.255.0
 ptp announce interval 3
 ptp announce timeout 2
 ptp sync interval -4
 ptp delay-req interval -4
 ptp slave multicast hybrid
 ptp enable

network-clock-select 1 SYNCE 0/1
```

PTP Hot Standby Master Clock

The following section shows a sample PT P configuration that uses a hot standby master clock. For more information about how to configure a PTP hot standby master clock, see "Configuring PTP Clocking" section on page 16-4.

Note This section provides a partial configuration intended to demonstrate a specific feature.

```
ptp mode ordinary
ptp priority1 128
ptp priority2 128
ptp domain 1
ptp best-recovered-quality 2 30

interface Vlan1
 ip address 192.168.1.2 255.255.255.0
 ptp announce interval 3
 ptp announce timeout 2
 ptp sync interval -4
 ptp delay-req interval -4
 ptp slave unicast negotiation
 ptp clock-source 10.0.1.2
 ptp clock-source 10.0.1.3
 ptp enable

network-clock-select 1 PACKET_TIMING
```

PTP Input Timing

The following sample configuration sets the router as a PTP master clock with input timing enabled using the 10Mhz timing port.

Note This section only applies to the Cisco MWR 2941-DC-A router; the Cisco MWR-DC router does not have the timing ports used in this example.

Note This section provides a partial configuration intended to demonstrate a specific feature.

```
ptp mode ordinary
ptp priority1 128
ptp priority2 128
ptp domain 0
ptp input 10M 1pps
ptp tod iso
ptp update-calendar

interface GigabitEthernet 0/0
    switchport access vlan 1588

interface vlan 1588
    ip address 192.168.15.89 255.255.255.0
    ip igmp join-group 224.0.1.129
    ptp sync interval -6
    ptp delay-req interval -4
    ptp master multicast
    ptp enable

network-clock-select hold-timeout 3600
network-clock-select 1 10M
```

PTP Output Timing

The following sample configuration sets the router as a PTP slave clock with output timing enabled on the 10M timing port.

Note This section only applies to the Cisco MWR 2941-DC-A router.; the Cisco MWR-DC router does not have the timing ports used in this example.

Note This section provides a partial configuration intended to demonstrate a specific feature.

```
ptp mode ordinary
ptp priority1 128
ptp priority2 128
ptp domain 0
ptp output 10M 1pps
ptp tod ubx delay 1
ptp update-calendar
```

```
interface GigabitEthernet 0/0
    switchport access vlan 1588

interface vlan 1588
    ip address 192.168.15.88 255.255.255.0
    ip igmp join-group 224.0.1.129
    ptp sync interval -6
    ptp delay-req interval -4
    ptp slave multicast
    ptp enable

network-clock-select hold-timeout 1000
network-clock-select 1 PACKET-TIMING
 enable 10M
```

CHAPTER 17

Configuring Synchronous Ethernet ESMC and SSM

With Ethernet equipment gradually replacing Synchronous Optical Networking (SONET) and Synchronous Digital Hierarchy (SDH) equipment in service-provider networks, frequency synchronization is required to provide high-quality clock synchronization over Ethernet ports.

Synchronous Ethernet (SyncE) provides the required synchronization at the physical level. In SyncE, Ethernet links are synchronized by timing their bit clocks from high-quality, stratum-1-traceable clock signals in the same manner as SONET/SDH. Operation messages maintain SyncE links, and ensure a node always derives timing from the most reliable source.

The SyncE synchronizes clock frequency over an Ethernet port. In SONET/SDH the communication channel for conveying clock information is Synchronization Status Message (SSM), and in SyncE it is the Ethernet Synchronization Message Channel (ESMC).

Note For information about how to configure synchronous Ethernet, see "Configuring Clocking and Timing".

Contents

Prerequisites for Synchronous Ethernet (SyncE): ESMC and SSM

You need to first configure the network clock for SyncE configuration. Automatic synchronization of the network clock should be enabled. Ensure the **network-clock-select** and **network-clock-participate** commands do not exist in the configuration in order to continue with the SyncE configuration.

Restrictions for Synchronous Ethernet (SyncE): ESMC and SSM

- To use the **network-clock synchronization ssm option** command, the following conditions are required:
  - No input source is in the configuration.
  - No network clock quality level is in the configuration.
  - No network clock source quality source is set under any synchronous Ethernet interface.
- The **network-clock synchronization ssm option** command must be compatible with the **network-clock eec** command in the configuration.
- The **esmc process** and **synchronous mode** commands can be used only if the SyncE capable interface is installed on the router.

Information About Synchronous Ethernet (SyncE): ESMC and SSM

- Synchronous Ethernet (SyncE): ESMC and SSM, page 17-2

Synchronous Ethernet (SyncE): ESMC and SSM

Customers using a packet network find it difficult to provide timing to multiple remote network elements (NEs) through an external time division multiplexed (TDM) circuit. The SyncE feature helps to overcome this problem by providing effective timing to the remote NEs through a packet network. SyncE leverages the physical layer of Ethernet to transmit frequency to the remote sites. SyncE's functionality and accuracy resemble the SONET/SDH network because of its physical layer characteristic. SyncE uses ESMC to allow the best clock source traceability, to correctly define the timing source, and to help prevent a timing loop.

SONET/SDH use 4 bits from the two S bytes in the SONET/SDH overhead frame for message transmission. Ethernet relies on ESMC that is based on an IEEE 802.3 organization-specific slow protocol for message transmission. Each NE along the synchronization path supports SyncE, and SyncE effectively delivers frequency in the path. SyncE do not support relative time (for example, phase alignment) or absolute time (Time of Day).

SyncE provides the Ethernet physical layer network (ETY) level frequency distribution of known common precision frequency references. Clocks for use in SyncE are compatible with the clocks used in the SONET/SDH synchronization network. To achieve network synchronization, synchronization information is transmitted through the network via synchronous network connections with performance of egress clock. In SONET/SDH the communication channel for conveying clock information is Synchronization Status Message (SSM), and in SyncE it the Ethernet Synchronization Message Channel (ESMC).

ESMC carries a Quality Level (QL) identifier that identifies the timing quality of the synchronization trail. QL values in QL-TLV are the same as QL values defined for SONET and SDH SSM. Information provided by SSM QLs during the network transmission helps a node derive timing from the most reliable source and prevents timing loops. ESMC is used with the synchronization selection algorithms. Because Ethernet networks are not required to be synchronous on all links or in all locations, the ESMC channel provides this service. ESMC is composed of the standard Ethernet header for an organization-specific slow protocol; the ITU-T OUI, a specific ITU-T subtype; an ESMC-specific header; a flag field; and a type, length, value (TLV) structure. The use of flags and TLVs improves the management of SyncE links and the associated timing change.

How to Configure Synchronous Ethernet (SyncE): ESMC and SSM

Perform this task to configure SyncE using ESMC and SSM.

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`controller BITS`**
	Example:
	`Router(config)# controller BITS`
		Enters BITS controller configuration mode.

Step 4	**`applique {E1 | T1}`**
	Example:
	`Router(config-controller)# applique e1`
		Specifies the BITS controller type.

Step 5	**E1 controller**
	`framing {crc4 | no-crc4 | none}`
	T1 controller
	`framing {esf | none | sf}`
	Example: E1 Controller
	`Router(config-controller)# framing crc4`
	Example: T1 Controller
	`Router(config-controller)# framing esf`
		Specify the framing type for the E1 or T1 BITS interface.
		For an E1 interface, ensure that the controller is set to use crc4 framing; CRC4 is the default setting.
		For a T1 interface, configure the controller to use ESF framing; ESF is not the default setting.

Step 6	**`ssm`**
	Example:
	`Router(config-controller)# ssm`
		Enables SSM on the T1 or E1 BITS interface.

	Command or Action	Purpose
Step 7	**sabit** **Example:** `Router(config-controller)# sabit 4`	(Optional) Specifies the San synchronization status bit used to indicate the clock quality level. Valid values are 4–8. **Note** This command only applies to the E1 controller.

Step 8	**exit**
	Example:
	`Router(config-controller)# exit`
	`Router(config)#`
		Exits controller configuration mode and returns to configuration mode.

Step 9	**network-clock synchronization automatic** **Example:** `Router(config)# network-clock synchronization automatic`	Enables the network clock selection algorithm. This command disables the Cisco-specific network clock process and turns on the G.781-based automatic clock selection process.

Step 10	**network-clock eec {1 | 2}**
	Example:
	`Router(config)# network-clock eec 1`
		Configures the clocking system hardware with the desired parameters. These are the options:
		• For option 1, the default value is EEC-Option 1 (2048).
		• For option 2, the default value is EEC-Option 2 (1544).

Step 11	**network-clock synchronization ssm option** {**1** | **2** {**GEN1** | **GEN2**}} **Example:** `Router(config)# network-clock synchronization ssm option 2 GEN2`	Configures the router to work in a synchronization network. • Option 1 refers to synchronization networks designed for Europe. This is the default value. • Option 2 refers to synchronization networks designed for United States.

Step 12	**network-clock input-source** *priority* {**controller BITS** | **E1**} | {**interface** *type slot/card/port*] {**external** [**2m** | **10m**]}}
	Example:
	`Router(config)# network-clock input-source 1 interface GigabitEthernet 0/1`
		Enables you to select an interface as an input clock for the router. You can select the BITS, Gigabit Ethernet 0/0, Gigabit Ethernet 0/1 interfaces, or GPS interfaces.

Step 13	**network-clock synchronization mode ql-enabled** **Example:** `Router(config)# network-clock synchronization mode ql-enabled`	Configure the automatic selection process ql-enabled mode. • QL is disabled by default. • ql-enabled mode can be used only when the synchronization interface is capable to send SSM.

Step 14	**network-clock hold-off** {**0** | *milliseconds*}
	Example:
	`Router(config)# network-clock hold-off 0`
		(Optional) Configures hold-off timer for the interface.

Step 15	**network-clock wait-to-restore** *seconds* **Example:** `Router(config)# network-clock wait-to-restore 70`	(Optional) Configures wait-to-restore timer for the SyncE interface.

	Command or Action	Purpose

Step 16	**network-clock-select mode** {**revert** | **nonrevert**}
	Example:
	`Router(config)# network-clock-select mode revert`
		(Optional) Specifies the router switching mode when recovering from a failure.

Step 17	**network-clock-select hold-timeout** {*timeout* | **infinite**}
	Example:
	`Router(config)# network-clock-select hold-timeout 2000`
		(Optional) Specifies how long the router waits before reevaluating the network clock entry.

Step 18	**esmc process** **Example:** `Router(config)# esmc process`	Enables the ESMC process.

Step 19	**network-clock external** *slot/card/port* **hold-off** {**0** | *milliseconds*}
	Example:
	`Router(config)# network-clock external 0/1/0 hold-off 0`
		Overrides the hold-off timer value for the external interface.

Step 20	**network-clock quality-level** {**tx** | **rx**} *value* {**interface** *type slot/card/port* | **external** {**2m** | **10m**} | **controller** {**BITS** | **E1**}
	Example:
	`Router(config)# network-clock quality-level rx QL-STU GigabitEthernet 0/0`
		Forces the QL value for line or external timing input and output.

Step 21	**interface** *type number* **Example:** `Router(config)# interface GigabitEthernet 0/0`	Enters interface configuration mode.
Step 22	**synchronous mode** **Example:** `Router(config-if)# synchronous mode`	Configures the Ethernet interface to synchronous mode and automatically enables the ESMC and QL process on the interface.
Step 23	**esmc mode** [**ql-disabled** | **tx** | **rx**] *value* **Example:** `Router(config-if)# esmc mode rx QL-STU`	(Optional) Enables the ESMC process on the interface.
Step 24	**network-clock source quality-level** *value* {**tx** | **rx**} **Example:** Router(config-if)# **network-clock source quality-level ql-prc tx**	(Optional) Provides the forced QL value to the local clock selection process.

	Command or Action	Purpose

Step 25	**network-clock hold-off** {*0* | *milliseconds*}
	Example:
	`Router(config-if)# network-clock hold-off 0`
		(Optional) Configures the hold-off timer for the interface.

Step 26	**network-clock wait-to-restore** *seconds* **Example:** `Router(config-if)# network-clock wait-to-restore 70`	(Optional) Configures wait-to-restore timer for the SyncE interface.

Step 27	**end**
	Example:
	`Router(config-if)# end`
		Exits interface configuration mode and returns to privileged EXEC mode.

You can use the **show network-clocks** command to verify your configuration.

Configuration Examples for Synchronous Ethernet (SyncE): ESMC and SSM

Example: Synchronous Ethernet (SyncE): ESMC and SSM

The following examples show the SyncE configuration sequence (configuring an interface with two SyncE interfaces and two external interfaces):

```
Interface GigabitEthernet0/0
    synchronous mode
    network-clock wait-to-restore 720
!
Interface GigabitEthernet0/1
    synchronous mode
!
controller BITS
ssm

!
network-clock synchronization automatic
network-clock input-source 1 controller BITS
network-clock input-source 1 gigabitethernet 0/0
network-clock input-source 2 gigabitethernet 0/1
network-clock synchronization mode QL-enabled
network-clock-select hold-timeout infinite
network-clock-select mode nonrevert
```

The following examples shows how to verify whether ESMC is enabled or not:

```
Router# show esmc

Interface: GigabitEthernet0/0
Administrative configurations:
  Mode: Synchronous
  ESMC TX: Enable
  ESMC RX : Enable
  QL RX configured : NA
  QL TX configured : NA
Operational status:
  Port status: UP
  QL Receive: QL-SSU-B
  ESMC Information rate : 1 packet/second
  ESMC Expiry: 5 second
```

The following examples shows how to view the network clock synchronization details:

```
Router# show network-clock synchronization detail

Automatic selection process : Enable
Equipment Clock : 2048 (EEC-Option1)
Clock Mode : QL-Enable
ESMC : Disabled
SSM Option : 1
T0 : Internal
Hold-off (global) : 300 ms
Wait-to-restore (global) : 300 sec
Revertive : No
Force Switch: FALSE
Manual Switch: FALSE
Number of synchronization sources: 1
Secondary src: Ethernet0/0
Slots disabled 0x0
Monitor source(s):  Ethernet0/0
Selected QL: QL-SEC
sm(netsync_ql_dis NETCLK_QL_ENABLE), running yes, state 1A
Last transition recorded: (begin)-> 1A (ql_mode_enable)-> 1A (src_added)-> 1A


Nominated Interfaces

 Interface           SigType     Mode/QL      Prio  QL_IN     ESMC Tx   ESMC Rx
*Internal            NA          NA/Dis       251   QL-SEC    NA        NA
 Et0/0               NA          Sync/En      2     QL-DNU    -         -

Interface:
---------------------------------------------
Local Interface: Internal
Signal Type: NA
Mode: NA(Ql-enabled)
SSM Tx: Disable
SSM Rx: Disable
Priority: 251
QL Receive: QL-SEC
QL Receive Configured: -
QL Receive Overrided: -
QL Transmit: -
QL Transmit Configured: -
Hold-off: 0
Wait-to-restore: 0
Lock Out: FALSE
Signal Fail: FALSE
Alarms: FALSE
Slot Disabled: FALSE

Local Interface: Et0/0
Signal Type: NA
Mode: Synchronous(Ql-enabled)
ESMC Tx: Enable
ESMC Rx: Enable
Priority: 2
QL Receive: QL-DNU
QL Receive Configured: -
QL Receive Overrided: -
QL Transmit: -
QL Transmit Configured: -
Hold-off: 300
Wait-to-restore: 300
Lock Out: FALSE
Signal Fail: FALSE
Alarms: FALSE
Slot Disabled: FALSE
Dont Use: FALSE
Configured Priority: 2
Force Switch: FALSE
Manual Switch: FALSE
Manual Switch In progress: FALSE
Holdoff_cfg: FALSE
Wtr_cfg: FALSE
Reason for alarm flag: 0
Msw in progress: FALSE
Intf_sig_nv: 0
Hold off Timer: Stopped
Wait to restore Timer: Stopped
Switchover Timer: Stopped
ESMC Tx Timer: Stopped
ESMC Rx Timer: Stopped
Tsm Delay Timer: Stopped
```

Additional References

Related Documents

Related Topic	Document Title
SyncE configuration commands	*Cisco IOS Interface and Hardware Component Command Reference*

Standards

Standard	Title
ITU-T G.8262	*Timing characteristics of synchronous Ethernet equipment slave clock (EEC)*
ITU-T G.8264	*Timing distribution through Packet Networks*
ITU-T G.781	*Synchronization layer functions*

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
None	

CHAPTER 18

Configuring MLPPP Backhaul

To configure an MLPPP backhaul, complete the following tasks:

- Configuring the Card Type, page 18-1
- Configuring E1 Controllers, page 18-2
- Configuring T1 Controllers, page 18-4
- Configuring a Multilink Backhaul Interface, page 18-7

Configuring the Card Type

Perform a basic card type configuration by enabling the router, enabling an interface, and specifying the card type as described below. You might also need to enter other configuration commands, depending on the requirements for your system configuration and the protocols you plan to route on the interface.

Note In the following procedure, press the **Return** key after each step unless otherwise noted. At any time, you can exit the privileged level and return to the user level by entering **disable** at the `Router#` prompt.

To select and configure a card type, follow these steps:

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

	Command	Purpose
Step 3	**card type {e1 | t1}** *slot subslot* **Example:** `Router(config)# card type e1 0 1`	Sets the card type. The command has the following syntax: • *slot*—Slot number of the interface. • *subslot*—VWIC slot number. The example shows how to configure a T1/E HWIC in the first HWIC slot as an E1 card. When the command is used for the first time, the configuration takes effect immediately. A subsequent change in the card type does not take effect unless you enter the **reload** command or reboot the router. **Note** When you are using the **card type** command to change the configuration of an installed card, you must first enter the **no card type** {**e1** | **t1**} *slot subslot* command. Then enter the **card type** {**e1** | **t1**} *slot subslot* command for the new configuration information.

Step 4	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exit configuration mode.

Configuring E1 Controllers

Perform a basic E1 controller configuration by specifying the E1 controller, entering the clock source, specifying the channel-group, configuring the serial interface, configuring PPP encapsulation, and enabling keepalive packets. You might also need to enter other configuration commands, depending on the requirements for your system configuration and the protocols you plan to route on the interface.

Note In the following procedure, press the **Return** key after each step unless otherwise noted. At any time, you can exit the privileged level and return to the user level by entering **disable** at the `Router#` prompt.

To configure the E1 controllers, follow these steps in global configuration mode:

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command	Purpose

Step 3	`controller e1` *`slot/port`*
	Example:
	`Router(config)# controller e1 0/0`
	`Router(config-controller)#`
		Specifies the controller that you want to configure. Controller E1 0/0 maps to the T1/E1 HWIC card in HWIC slot 0.
		The example shows how to specify the E1 controller as the first port of the T1/E1 HWIC card in slot 0.

Step 4	`framing {crc4 | no-crc4}`
	Example:
	`Router(config-controller)# framing crc4`
		Specifies the framing type.

Step 5	`linecode {ami | hdb3}`
	Example:
	`Router(config-controller)# linecode ami`
		Specifies the line code format.

Step 6	`mode {atm | cas}`
	Example:
	`Router(config-controller)# mode cas`
		Sets the controller in ATM or channel-associated signaling (CAS) mode.

Step 7	`clock source {line | internal} [bits]`
	Example:
	`Router(config-controller)# clock source line`
		Specifies the clocking source. The syntax is:
		• *line*—Specifies the E1 line from which the clocking is taken.
		• *internal*—Internal clocking.
		• bits—Building Integrated Timing Supply (BITS) clocking.
		The example shows how to configure the clock source for the E1 controller.
		Note When you are using the **clock source** command to change the configuration of an installed card, you must enter the **no clock source** command first. Then enter the **clock source** command for the new configuration information.

	Command	Purpose
Step 8	Router(config-controller)# **channel-group** *channel-no* **timeslots** *timeslot-list* **speed {64}** **Example:** Router(config-controller)# **channel-group 0 timeslots 1-31 speed 64**	Specifies the channel-group and time slots to be mapped. After you configure a channel-group, the serial interface is automatically created. The syntax is: • *channel-no*—ID number to identify the channel group. The valid range is from 0–30. • *timeslot-list*—Timeslots (DS0s) to include in this channel-group. The valid time slots are from 1–31. • **speed {64}**—The speed of the DS0. The example configures the channel-group and time slots for the E1 controller: **Note** When you are using the **channel-group** *channel-no* **timeslots** *timeslot-list* **{64}** command to change the configuration of an installed card, you must enter the **no channel-group** *channel-no* **timeslots** *timeslot-list* **speed {64}** command first. Then enter the **channel-group** *channel-no* **timeslots** *timeslot-list* **{64}** command for the new configuration information.
Step 9	Router(config-controller)# **exit** Router(config)#	Exits controller configuration mode.
Step 10	**interface serial** *slot*/*port*:**channel** **Example:** Router(config)# **interface serial 0/0:1** Router(config-if)#	Configures the serial interface. Specify the E1 slot, port number, and channel-group. When the prompt changes to `Router(config-if)`, you have entered interface configuration mode. **Note** To see a list of the configuration commands available to you, enter **?** at the prompt or press the **Help** key while in the configuration mode.
Step 11	Router(config-if)# **encapsulation ppp**	Specifies PPP encapsulation on the interface.
Step 12	keepalive [period [retries]] **Example:** Router(config-if)# **keepalive** [*period* [*retries*]]	Enables keepalive packets on the interface and specify the number of times keepalive packets are sent without a response before the router disables the interface.
Step 13	Router(config-if)# **end** Router#	Exits interface configuration mode.

Configuring T1 Controllers

Use the following steps to perform a basic T1 controller configuration: specifying the T1 controller, specifying the framing type, specifying the line code form, specifying the channel-group and time slots to be mapped, configuring the cable length, configuring the serial interface, configuring PPP encapsulation, and enabling keepalive packets. You might also need to enter other configuration commands, depending on the requirements for your system configuration and the protocols you plan to route on the interface.

Note In the following procedure, press the **Return** key after each step unless otherwise noted. At any time, you can exit the privileged level and return to the user level by entering **disable** at the `Router#` prompt.

To configure the T1 interfaces, follow these steps in the global configuration mode:

	Command	Purpose
Step 1	**enable** **Example:** Router> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** Router# configure terminal	Enters global configuration mode.
Step 3	**card type {e1 | t1}** *slot subslot* **Example:** Router(config)# **card type t1 0 1**	Sets the card type. The command has the following syntax: • *slot*—Slot number of the interface. • *subslot*—The VWIC slot number. Controller T1 0/0 maps to the T1/E1 HWIC card in HWIC slot 0. The example shows how to configure a T1/E HWIC in the first HWIC slot as an T1 card. When the command is used for the first time, the configuration takes effect immediately. A subsequent change in the card type does not take effect unless you enter the **reload** command or reboot the router. **Note** When you are using the **card type** command to change the configuration of an installed card, you must first enter the **no card type** {**e1** | **t1**} slot subslot command. Then enter the **card type** {**e1** | **t1**} *slot subslot* command for the new configuration information.
Step 4	Router(config-controller)# **framing esf**	Specifies the framing type.
Step 5	Router(config-controller)# **linecode b8zs**	Specifies the line code format.
Step 6	Router(config-controller)# **mode** {atm | cas}	Set the controller in ATM or channel-associated signaling (CAS) mode.
Step 7	Router(config-controller)# **channel-group 0 timeslots 1-24 speed 56**	Specifies the channel-group and time slots to be mapped. After you configure a channel-group, the serial interface is automatically created. **Note** The default speed of the channel-group is 56.
Step 8	Router(config-controller)# **cablelength {long [-15db | -22.5db | -7.5db | 0db] short [110ft | 220ft | 330ft| 440ft | 550ft | 600ft]}**	Configures the cable length.
Step 9	Router(config-controller)# **exit**	Exits controller configuration mode.
Step 10	Router(config)# **interface serial** *slot/port*:**channel**	Configures the serial interface. Specify the T1 slot (always 0), port number, and channel-group.

	Command	Purpose
Step 11	`Router(config-if)#` **`encapsulation ppp`**	Enters the following command to configure PPP encapsulation.
Step 12	`Router(config-if)#` **`keepalive`** `[`*`period`* `[`*`retries`*`]]`	Enables keepalive packets on the interface and specify the number of times that keepalive packets will be sent without a response the interface is brought down:

Step 13	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Configuring ATM IMA

Inverse multiplexing provides the capability to transmit and receive a single high-speed data stream over multiple slower-speed physical links. In Inverse Multiplexing over ATM (IMA), the originating stream of ATM cells is divided so that complete ATM cells are transmitted in round-robin order across the set of ATM links. Follow these steps to configure ATM IMA on the Cisco MWR 2941.

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)#` **`card type e1 0 0`**	Specifies the slot and port number of the E1 or T1 interface.

Step 4	`Router(config)#` **`controller E1 0/4`**
	`Router(config-controller)#`
		Specifies the controller interface on which you want to enable IMA.

Step 5	`Router(config-controller)#` **`clock source internal`**	Set sthe clock source to internal.
Step 6	`Router(config-controller)#` **`ima-group 0 scrambling-payload`**	Assigns the interface to an IMA group, and set the scrambling-payload parameter to randomize the ATM cell payload frames. This command assigns the interface to IMA group 0. **Note** This command automatically creates an ATM0/IMAx interface.
Step 7		To add another member link, repeat Step 3 to Step 6.

Step 8	`Router(config-controller)#` **`exit`**
	`Router(config)#`
		Exits the controller interface.

	Command	Purpose
Step 9	**interface ATM***slot***/IMA***<group-number>* **Example:** Router(config-if)# **interface atm0/ima0**	Specify the slot location and port of IMA interface group. • *slot*—The slot location of the ATM IMA port adapter. • *group-number*—The group number of the IMA group. The example specifies the slot number as 0 and the group number as 0. **Note** To explicitly configure the IMA group ID for the IMA interface, you may use the optional **ima group-id** command. You cannot configure the same IMA group ID on two different IMA interfaces; therefore, if you configure an IMA group ID with the system-selected default ID already configured on an IMA interface, the system toggles the IMA interface to make the user-configured IMA group ID the effective IMA group ID. At the same, the system toggles the original IMA interface to select a different IMA group ID.
Step 10	Router(config-if)# **no ip address**	Disables the IP address configuration for the physical layer interface.
Step 11	Router(config-if)# **atm bandwidth dynamic**	Specifies the ATM bandwidth as dynamic.
Step 12	Router(config-if)# **no atm ilmi-keepalive**	Disables the Interim Local Management Interface (ILMI) keepalive parameters.
Step 13	**exit** **Example:** Router(config)# exit Router#	Exits configuration mode.

Note The above configuration has one IMA shorthaul with two member links (atm0/0 and atm0/1).

Configuring a Multilink Backhaul Interface

A multilink interface is a virtual interface that represents a multilink PPP bundle. The multilink interface coordinates the configuration of the bundled link, and presents a single object for the aggregate links. However, the individual PPP links that are aggregated must also be configured. Therefore, to enable multilink PPP on multiple serial interfaces, you first need to set up the multilink interface, and then configure each of the serial interfaces and add them to the same multilink interface.

Note In the following procedure, press the **Return** key after each step unless otherwise noted. At any time, you can exit the privileged level and return to the user level by entering **disable** at the `Router#` prompt.

The Cisco MWR 2941 router can support up to 16 E1/T1 connections through the multilink interface, ranging from 12 bundles of 1 E1/T1 each to a single bundle containing 16 E1/T1 bundles.

Complete the following tasks to configure a multilink backhaul interface.

- Creating a Multilink Bundle
- Configuring PFC and ACFC, page 18-9
- Enabling Multilink and Identifying the Multilink Interface, page 18-11
- Enabling Real-Time Transport Protocol (RTP) Header Compression, page 18-13

Creating a Multilink Bundle

To create a multilink bundle, follow these steps while in the global configuration mode:

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)#` **`interface multilink`** *`group-number`*
	Example:
	`Router(config)#` **`interface multilink5`**
	`Router(config-if)#`
		Creates a multilink bundle and enter the interface configuration mode:
		• *group-number*—Number of the multilink bundle.
		The example creates a multilink bundle 5.
		To remove a multilink bundle, use the **no** form of this command.
		Note To see a list of the configuration commands available to you, enter **?** at the prompt or press the **Help** key while in the configuration mode.

Step 4	`Router(config-if)#` **`ip address`** *`address [subnet mask]`*
	Example:
	`Router(config-if)#` **`ip address 10.10.10.2 255.255.255.0`**
		Assigns an IP address to the multilink interface.
		• *address*— IP address.
		• *subnet mask*—Network mask of IP address.
		The example configures an IP address and subnet mask.

Step 5	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Configuring PFC and ACFC

Protocol-Field-Compression (PFC) and Address-and-Control-Field-Compression (AFC) are PPP compression methods defined in RFCs 1661 and 1662. PFC allows for compression of the PPP Protocol field; ACFC allows for compression of the PPP Data Link Layer Address and Control fields.

Follow these steps to configure PFC and ACFC handling during PPP negotiation to be configured. By default, PFC/ACFC handling is not enabled.

Note The recommended PFC and ACFC handling in the Cisco MWR 2941 router is: **acfc local request, acfc remote apply, pfc local request, and pfc remote apply**.

Configuring PFC

To configure PFC handling during PPP negotiation, follow these steps:

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config-if)# ppp pfc local {request | forbid}`
	Example:
	`Router(config-if)# ppp pfc local request`
		Configures how the router handles PFC in its outbound configuration requests, use the **ppp pfc local** command. The syntax is as follows:
		• **request**—The PFC option is included in outbound configuration requests.
		• **forbid**—The PFC option is not sent in outbound configuration requests, and requests from a remote peer to add the PFC option are not accepted.
		The example shows how to create a method for the router to manage PFC.

	Command	Purpose
Step 4	Router(config-if)# **ppp pfc remote** {**apply** | **reject** | **ignore**} **Example:** Router(config)# **ppp pfc remote apply**	Specifies how the router manages the PFC option in configuration requests received from a remote peer. The syntax is as follows: • **apply**—Specifies that PFC options are accepted and ACFC may be performed on frames sent to the remote peer. • **reject**—Specifies that PFC options are explicitly ignored. • **ignore**—Specifies that PFC options are accepted, but ACFC is not performed on frames sent to the remote peer. The example shows how to allow PFC options to be accepted.
Step 5	**exit** **Example:** Router(config)# exit Router#	Exits configuration mode.

Configuring ACFC

To configure ACFC handling during PPP negotiation, follow these steps, while in interface configuration mode:

	Command	Purpose
Step 1	**enable** **Example:** Router> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Router# configure terminal	Enters global configuration mode.
Step 3	Router(config-if)# **ppp acfc local** {**request** | **forbid**} **Example:** Router(config-if)# **ppp acfc local request**	Specifies how the router handles ACFC in outbound configuration requests. The syntax is as follows: • **request**—Specifies that the ACFC option is included in outbound configuration requests. • **forbid**—Specifies that the ACFC option is not sent in outbound configuration requests, and requests from a remote peer to add the ACFC option are not accepted.

	Command	Purpose

Step 4	`Router(config-if)#` **`ppp acfc remote`** `{`**`apply`** `|` **`reject`** `|` **`ignore`**`}`
	Example:
	`Router(config-if)#` **`ppp acfc remote apply`**
		Specifies how the router handles the ACFC option in configuration requests received from a remote peer. The syntax is as follows:
		• **apply**—ACFC options are accepted and ACFC may be performed on frames sent to the remote peer.
		• **reject**—ACFC options are explicitly ignored.
		• **ignore**—ACFC options are accepted, but ACFC is not performed on frames sent to the remote peer.
		The example allows ACFC options to be accepted.

Step 5	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exit configuration mode.

Enabling Multilink and Identifying the Multilink Interface

To enable multilink and identify the multilink interface, follow these steps, while in interface configuration mode:

Note If you modify parameters for an MLPPP bundle while it is active, the changes do not take effect until the Cisco MWR 2941 renegotiates the bundle connection.

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config-if)#` **`ppp multilink`**	Enables multilink PPP operation.

Step 4	`Router(config-if)#` **`ppp multilink group`** *`group-number`*
	Example:
	`Router(config-if)#` **`ppp multilink group 5`**
		Configures the identification number for the multilink interface. The syntax is as follows:
		• ***group-number***—Multilink group number.
		The example restricts (identifies) the multilink interface that can be negotiated to multilink interface 5.

	Command	Purpose

Step 5	`Router(config-if)# keepalive [period [retries]]`
	Example:
	`Router(config-if)# keepalive 1 5`
		Enables keepalive packets on the interface and specifies the number of times the keepalive packets are sent without a response before the router disables the interface. The syntax is as follows:
		• ***period***—(Optional) Integer value in seconds greater than 0. The default is 10.
		• ***retries***—(Optional) Specifies the number of times that the device will continue to send keepalive packets without response before bringing the interface down. Integer value greater than 1 and less than 255. If omitted, the value that was previously set is used; if no value was specified previously, the default of 5 is used.

Step 6	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

MLPPP Offload

By default, the Cisco MWR 2941 offloads processing for distributed MLPPP (dMLPPP) to the network processor for improved performance. However, the Cisco MWR 2941 does not support some dMLPPP settings on offloaded bundles. The Cisco MWR 2941 does not support the following options on offloaded dMLPPP bundles:

- **ppp multilink idle-link**
- **ppp multilink queue depth**
- **ppp multilink fragment maximum**
- **ppp multilink slippage**
- **ppp timeout multilink lost-fragment**

Note If you have a bundle that requires the use of these options, contact Cisco support for assistance.

For more information about MLPPP offload, see MLPPP Optimization Features, page 1-2.

Configuring Additional MLPPP Settings

You can perform a variety of other configurations on an MLPPP bundle, including the following:

- Modifying the maximum fragment size
- Modifying fragmentation settings
- Enabling or disabling fragmentation
- Enabling or disabling interleaving
- Configuring distributed MLPPP (dMLPPP)
- Configuring multiclass MLPPP

For more information about configuring MLPPP, see the Dial Configuration Guide, Cisco IOS Release 15.0S.

Enabling Real-Time Transport Protocol (RTP) Header Compression

To enable RTP header compression, follow these steps while in the interface configuration mode:

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config-if)#` **`ip rtp header-compression [ietf-format] [periodic-refresh]`**
	Example:
	`Router(config-if)#` **`ip rtp header-compression ietf-format periodic-refresh`**
		Enable RTP header-compression using the **ip rtp header-compression** command. The syntax is as follows:
		• **ietf-format**—(Optional) Indicates that the Internet Engineering Task Force (IETF) format of header compression will be used.
		• **periodic-refresh**—(Optional) Indicates that the compressed IP header will be refreshed periodically.
		The example enables RTP header-compression in the Internet Engineering Task Force (IETF) format by suppressing the IP ID in the RTP/UDP header compression.
		Note IP header compression is only supported when MLPPP operates on the host processor; it is not supported when MLPPP is offloaded.

Step 4	**exit**
	Example:
	`Router(config)#` **`exit`**
	`Router#`
		Exit configuration mode.

CHAPTER 19

Configuring Multiprotocol Label Switching

Several technologies such as pseudowires utilize MPLS for packet transport. For more information about how to configure MPLS, see the *MPLS Configuration Guide, Cisco IOS Release 15.0S*.

Note The Cisco MWR 2941 does not necessarily support all of the commands listed in the Release 15.0S documentation.

CHAPTER 20

Configuring Routing Protocols

In addition to static routing, the Cisco MWR 2941 supports the following routing protocols:

- OSPF—An Interior Gateway Protocol (IGP) designed expressly for IP networks that supports IP subnetting and tagging of externally derived routing information. OSPF also allows packet authentication and uses IP multicast when sending and receiving packets. For instructions on how to configure OSPF, see the IP Routing: OSPF Configuration Guide, Cisco IOS Release 15.0S.
- IS-IS—An Open System Interconnection (OSI) protocol that specifies how routers communicate with routers in different domains. For instructions on how to configure IS–IS, see the IP Routing: ISIS Configuration Guide, Cisco IOS Release 15.0S.
- BGP—An interdomain routing protocol designed to provide loop-free routing between separate routing domains that contain independent routing policies (autonomous systems). For instructions on how to configure BGP, see the IP Routing: BGP Configuration Guide, Cisco IOS Release 15.0S.

Note For information about Bidirectional Forwarding Detection (BFD) including sample routing configurations with BFD, see Chapter 21, "Configuring Bidirectional Forwarding Detection."

CHAPTER 21

Configuring Bidirectional Forwarding Detection

Bidirectional Forwarding Detection (BFD) provides a low-overhead, short-duration method of detecting failures in the forwarding path between two adjacent routers, including the interfaces, data links, and forwarding planes. BFD is a detection protocol that you enable at the interface and routing protocol levels.

The following sections describe how to configure BFD on the Cisco MWR 2941:

- Understanding BFD, page 21-1
- Configuring BFD, page 21-1
- Configuration Examples for BFD, page 21-6

Understanding BFD

Cisco supports the BFD asynchronous mode, in which two routers exchange BFD control packets to activate and maintain BFD neighbor sessions. To create a BFD session, you must configure BFD on both systems (or BFD peers). After you have enabled BFD on the interface and the router level for the appropriate routing protocols, a BFD session is created, BFD timers are negotiated, and the BFD peers begin to send BFD control packets to each other at the negotiated interval.

Configuring BFD

The following sections describe how to configure BFD for each routing protocol:

- Configuring BFD for OSPF, page 21-2
- Configuring BFD for BGP, page 21-3
- Configuring BFD for IS-IS, page 21-4
- Configuring BFD for Static Routes, page 21-6

For more information about BFD, refer to the *IP Routing: BFD Configuration Guide, Cisco IOS Release 15.0S*. For a sample BFD configurations, see Configuration Examples for BFD.

Configuring BFD for OSPF

This section describes how to configure BFD on the Cisco MWR 2941.

Configuring BFD for OSPF on One of More Interfaces

Follow these steps to configure BFD for OSPF on a single interface.

	Command	Purpose
Step 1	**enable** **Example:** Router> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Router# **configure terminal**	Enters global configuration mode.
Step 3	Router(config)# **interface vlan1** Router(config-if)#	Specifies an interface to configure.
Step 4	Router(config-if)# **ip ospf bfd**	Enables BFD for OSPF on the interface.
Step 5	Router(config-if)# **bfd interval 50 min_rx 50 multiplier 3**	Specifies the BFD session parameters.
Step 6	**end** **Example:** Router(config-if)# **end** Router#	Exits configuration mode.

Note You can also use the **show bfd neighbors** and **show ip ospf** commands to display troubleshooting information about BFD and OSPF.

Configuring BFD for OSPF on All Interfaces

Follow these steps to configure BFD for OSPF on all interfaces.

	Command	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)#` **`router ospf 100`**	Creates a configuration for an OSPF process.
Step 4	`Router(config)#` **`bfd all-interfaces`**	Enables BFD globally on all interfaces associated with the OSPF routing process.

Step 5	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Note You can disable BFD on a single interface using the **ip ospf bfd disable** command when configuring the relevant interface.

Configuring BFD for BGP

Follow these steps to configure BFD for BGP.

	Command	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)#` **`router bgp`** *`as-tag`*	Specifies a BGP process and enter router configuration mode.
Step 4	`Router(config)#` **`neighbor`** *`ip-address`* **`fall-over bfd`**	Enables support for BFD failover.

	Command	Purpose

Step 5	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Step 6	**show bfd neighbors [details]** **show ip bgp neighbor**	Use the following commands to verify the BFD configuration: • **show bfd neighbors [details]** —Verifies that the BFD neighbor is active and displays the routing protocols that BFD has registered. • **show ip bgp neighbor**—Displays information about BGP and TCP connections to neighbors.

Configuring BFD for IS-IS

This section describes how to configure BFD for IS-IS routing.

Configuring BFD for IS-IS on a Single Interface

Follow these steps to configure BFD for IS-IS on a single interface.

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)# interface vlan1`
	`Router(config-if)#`
		Enters interface configuration mode.

Step 4	`Router(config-if) ip router isis [tag]`	Enables support for IPv4 routing on the interface.

Step 5	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Note You can use the **show bfd neighbors** and **show clns interface** commands to verify your configuration.

Configuring BFD for IS-IS for All Interfaces

Follow these steps to configure BFD for IS-IS on all interfaces.

	Command	Purpose
Step 1	**enable** **Example:** Router> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Router# **configure terminal**	Enters global configuration mode.
Step 3	Router(config)# **interface vlan1** Router(config-if)#	Enters interface configuration mode.
Step 4	Router(config-if) **ip router isis** [tag]	Enables support for IPv4 routing on the interface.
Step 5	Router(config-router)# **bfd all-interfaces**	Enables BFD globally on all interfaces associated with the IS-IS routing process.
Step 6	Router(config-router)# **exit** Router(config)#	Exits the interface.
Step 7	Router(config)# **interface vlan1** Router(config-if) **ip router isis** [tag] Router(config-if)# **isis bfd**	If you want to enable BFD on a per-interface basis for one or more interfaces associated with the IS-IS routing process, complete the following steps: **a.** Use the **interface** command to enter interface configuration mode. **b.** Use the **ip router isis** command to enables support for IPv4 routing on the interface. **c.** Use the **isis bfd** command to enable BFD on the interface.
Step 8	**exit** **Example:** Router(config)# **exit** Router#	Exit configuration mode.

Note You can use the **show bfd neighbors** and **show clns interface** commands to verify your configuration.

Configuring BFD for Static Routes

Follow these steps to configure BFD for static routes.

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)#` **`interface serial 2/0`**	Specifies an interface and enters interface configuration mode.
Step 4	`Router(config-if)#` **`ip address 10.201.201.1 255.255.255.0`**	Configures an IP address for the interface.
Step 5	`Router(config-if)#` **`bfd interval 500 min_rx 500 multiplier 5`**	Enables BFD on the interface.
Step 6	`Router(config-if)#` **`ip route static bfd Serial 2/0 10.201.201.2`**	Specifies a static route BFD neighbor.

Step 7	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Note You can use the **show ip static route** command to verify your configuration.

Configuration Examples for BFD

The following section contains sample configurations for each routing protocol using BFD.

- OSPF with BFD, page 21-6
- BGP with BFD, page 21-10
- IS-IS with BFD, page 21-13

For more information about how to configure routing on the Cisco MWR 2941, see Chapter 20, "Configuring Routing Protocols."

OSPF with BFD

```
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname BFD2941
!
boot-start-marker
boot-end-marker
!
card type t1 0 0
logging buffered 1000000
no logging console
!
no aaa new-model
ip source-route
!
!
ip cef
no ip domain lookup
ip host tftp 64.102.116.25
ptp mode ordinary
ptp priority1 128
ptp priority2 128
ptp domain 0
multilink bundle-name authenticated
!
archive
 log config
  hidekeys
!
controller T1 0/0
 mode atm
 clock source line
!
controller T1 0/1
 clock source line
 cem-group 0 timeslots 1-31
!
controller T1 0/2
 clock source internal
!
controller T1 0/3
 clock source internal
!
controller T1 0/4
 clock source internal
!
controller T1 0/5
 clock source internal
!
controller T1 0/6
 clock source internal
!
controller T1 0/7
 clock source internal
!
controller T1 0/8
 clock source internal
!
controller T1 0/9
 clock source internal
!
controller T1 0/10
 clock source internal
!
```

```
controller T1 0/11
 clock source internal
!
controller T1 0/12
 clock source internal
!
controller T1 0/13
 clock source internal
!
controller T1 0/14
 clock source internal
!
controller T1 0/15
 clock source internal
!
controller BITS
  applique E1
!
!
interface Loopback0
 ip address 88.88.88.150 255.255.255.255
!
interface GigabitEthernet0/0
 switchport trunk allowed vlan 1-9,11-4094
 switchport mode trunk
!
interface GigabitEthernet0/1
!
interface GigabitEthernet0/2
 switchport access vlan 10
!
interface GigabitEthernet0/3
!
interface GigabitEthernet0/4
!
interface GigabitEthernet0/5
!
interface ATM0/0
 no ip address
 scrambling-payload
 atm pvp 1 l2transport
  xconnect 10.10.10.2 10001 encapsulation mpls
 no atm ilmi-keepalive
 pvc 0/20 l2transport
  vc-hold-queue 80
  encapsulation aal0
  xconnect 10.10.10.2 10020 encapsulation mpls
 !
 pvc 0/30 l2transport
  encapsulation aal5
  xconnect 10.10.10.2 10030 encapsulation mpls
 !
 pvc 0/40
  vc-hold-queue 50
  encapsulation aal5snap
 !
!
interface CEM0/1
 no ip address
 cem 0
  xconnect 10.10.10.2 222 encapsulation mpls
 !
!
interface Vlan1
```

```
 no ip address
 shutdown
 no ptp enable
!
interface Vlan10
 ip address 192.168.52.88 255.255.255.0
 no ptp enable
!
interface Vlan100
 description Primary EVC
 ip address 172.22.41.2 255.255.255.0
 ip ospf cost 4
 ip ospf hello-interval 1
 ip ospf dead-interval 3
 no ptp enable
 mpls ip
 bfd interval 50 min_rx 50 multiplier 3
!
interface Vlan200
 description Secondary EVC
 ip address 172.22.42.2 255.255.255.0
 ip ospf cost 5
 ip ospf hello-interval 1
 ip ospf dead-interval 3
 no ptp enable
 mpls ip
!
router ospf 100
 router-id 88.88.88.150
 log-adjacency-changes
 timers throttle spf 50 50 1000
 timers throttle lsa all 0 25 10000
 timers lsa arrival 0
 timers pacing flood 20
 timers pacing retransmission 30
 redistribute static subnets
 network 88.88.88.150 0.0.0.0 area 0
 network 172.22.41.0 0.0.0.255 area 0
 network 172.22.42.0 0.0.0.255 area 0
 bfd all-interfaces
!
ip default-gateway 192.168.52.1
ip forward-protocol nd
ip route 0.0.0.0 0.0.0.0 192.168.52.1
ip route 64.102.116.25 255.255.255.255 192.168.52.1
!
!
ip http server
no ip http secure-server
!
control-plane
!
line con 0
 exec-timeout 0 0
 no modem enable
line aux 0
line vty 0 4
 exec-timeout 0 0
 privilege level 15
 password xxxxx
 login
!
exception data-corruption buffer truncate
network-clock-select hold-timeout infinite
```

```
network-clock-select mode nonrevert
network-clock-select 1 E1 0/0
end
```

BGP with BFD

```
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname BFD2941
!
boot-start-marker
boot-end-marker
!
card type t1 0 0
logging buffered 1000000
no logging console
!
no aaa new-model
ip source-route
!
!
ip cef
no ip domain lookup
ip host tftp 64.102.116.25
ptp mode ordinary
ptp priority1 128
ptp priority2 128
ptp domain 0
multilink bundle-name authenticated
!
archive
 log config
  hidekeys
!
controller T1 0/0
 mode atm
 clock source line
!
controller T1 0/1
 clock source line
 cem-group 0 timeslots 1-31
!
controller T1 0/2
 clock source internal
!
controller T1 0/3
 clock source internal
!
controller T1 0/4
 clock source internal
!
controller T1 0/5
 clock source internal
!
controller T1 0/6
 clock source internal
!
```

```
controller T1 0/7
 clock source internal
!
controller T1 0/8
 clock source internal
!
controller T1 0/9
 clock source internal
!
controller T1 0/10
 clock source internal
!
controller T1 0/11
 clock source internal
!
controller T1 0/12
 clock source internal
!
controller T1 0/13
 clock source internal
!
controller T1 0/14
 clock source internal
!
controller T1 0/15
 clock source internal
!
controller BITS
  applique E1
!
interface Loopback0
 ip address 20.20.20.20 255.255.255.255
!
interface GigabitEthernet0/2
 switchport access vlan 10
 load-interval 30
 duplex full
 speed 100
!
interface GigabitEthernet0/3
 switchport access vlan 200
 load-interval 30
 duplex full
 speed 100
!
interface GigabitEthernet0/4
 switchport access vlan 4
 load-interval 30
 duplex full
 speed 100
!
interface GigabitEthernet0/5
 switchport access vlan 100
 load-interval 30
 duplex full
 speed 100
!
interface ATM0/0
 no ip address
 scrambling-payload
 atm bandwidth dynamic
 pvc 0/100 l2transport
 !
!
```

```
interface ATM0/0.1 multipoint
 pvc 1/5 l2transport
  encapsulation aal0
  xconnect 10.10.10.10 10010 encapsulation mpls
 !
 pvc 1/6 l2transport
  encapsulation aal5
  xconnect 10.10.10.10 10020 encapsulation mpls
!
!
interface ATM0/0.2 multipoint
 xconnect 10.10.10.10 10030 encapsulation mpls
 pvc 2/5 l2transport
  encapsulation aal0
 !
 pvc 2/6 l2transport
  encapsulation aal0
 !
!
interface ATM0/1
 no ip address
 scrambling-payload
 no atm ilmi-keepalive
 pvc 0/100 l2transport
 !
!
interface Vlan4 (connected to 7600)
 ip address 11.1.1.2 255.255.255.0
 no ptp enable
 bfd interval 50 min_rx 50 multiplier 3
!
interface Vlan10
 ip address 192.168.40.61 255.255.255.128
 no ptp enable
 mpls ip
!
interface Vlan100
 ip address 12.1.1.2 255.255.255.0
 no ptp enable
 mpls bgp forwarding
 mpls ip
 bfd interval 50 min_rx 50 multiplier 3
!
interface Vlan200
 ip address 12.1.2.2 255.255.255.0
 no ptp enable
 mpls bgp forwarding
 mpls ip
 bfd interval 50 min_rx 50 multiplier 3
!
router bgp 200
 no synchronization
 bgp log-neighbor-changes
 network 11.1.1.0
 network 12.1.1.0
 network 12.1.2.0
 redistribute connected
 neighbor 11.1.1.1 remote-as 100
 neighbor 11.1.1.1 fall-over bfd
 neighbor 11.1.1.1 send-label
 neighbor 12.1.1.1 remote-as 300
 neighbor 12.1.1.1 fall-over bfd
 neighbor 12.1.1.1 send-label
 neighbor 12.1.2.1 remote-as 300
 neighbor 12.1.2.1 fall-over bfd
 neighbor 12.1.2.1 send-label
 no auto-summary
!
connect atmcellsw ATM0/0 0/100 ATM0/1 0/100
 !
!
mpls ldp router-id Loopback0 force
!
```

IS-IS with BFD

```
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname BFD2941
!
boot-start-marker
boot-end-marker
!
card type t1 0 0
logging buffered 1000000
no logging console
!
no aaa new-model
ip source-route
!
!
ip cef
no ip domain lookup
ip host tftp 64.102.116.25
ptp mode ordinary
ptp priority1 128
ptp priority2 128
ptp domain 0
multilink bundle-name authenticated
!
archive
 log config
  hidekeys
!
controller T1 0/0
 mode atm
 clock source line
!
controller T1 0/1
 clock source line
 cem-group 0 timeslots 1-31
!
controller T1 0/2
 clock source internal
!
controller T1 0/3
 clock source internal
!
controller T1 0/4
 clock source internal
!
```

```
controller T1 0/5
 clock source internal
!
controller T1 0/6
 clock source internal
!
controller T1 0/7
 clock source internal
!
controller T1 0/8
 clock source internal
!
controller T1 0/9
 clock source internal
!
controller T1 0/10
 clock source internal
!
controller T1 0/11
 clock source internal
!
controller T1 0/12
 clock source internal
!
controller T1 0/13
 clock source internal
!
controller T1 0/14
 clock source internal
!
controller T1 0/15
 clock source internal
!
controller BITS
  applique E1
!
interface Loopback0
 ip address 20.20.20.20 255.255.255.255
!
interface GigabitEthernet0/2
 switchport access vlan 10
 load-interval 30
 duplex full
 speed 100
!
interface GigabitEthernet0/3
 switchport access vlan 200
 load-interval 30
 duplex full
 speed 100
!
interface GigabitEthernet0/4
 switchport access vlan 4
 load-interval 30
 duplex full
 speed 100
!
interface GigabitEthernet0/5
 switchport access vlan 100
 load-interval 30
 duplex full
 speed 100
!
interface ATM0/0
```

```
 no ip address
 scrambling-payload
 atm bandwidth dynamic
pvc 0/100 l2transport
 !
!
interface ATM0/0.1 multipoint
 pvc 1/5 l2transport
  encapsulation aal0
  xconnect 10.10.10.10 10010 encapsulation mpls
 !
 pvc 1/6 l2transport
  encapsulation aal5
  xconnect 10.10.10.10 10020 encapsulation mpls
 !
!
interface ATM0/0.2 multipoint
 xconnect 10.10.10.10 10030 encapsulation mpls
 pvc 2/5 l2transport
  encapsulation aal0
 !
 pvc 2/6 l2transport
  encapsulation aal0
 !
!
interface ATM0/1
 no ip address
 scrambling-payload
 no atm ilmi-keepalive
 pvc 0/100 l2transport
 !
!
interface Vlan4
 ip address 11.1.1.2 255.255.255.0
 ip router isis test_ip_isis
 no ptp enable
 isis bfd
!
interface Vlan10
 ip address 192.168.40.61 255.255.255.128
 no ptp enable
 mpls ip
!
interface Vlan100
 ip address 12.1.1.2 255.255.255.0
 ip router isis test_ip_isis
 no ptp enable
 mpls ip
 bfd interval 50 min_rx 50 multiplier 3
 isis bfd
!
interface Vlan200
 ip address 12.1.2.2 255.255.255.0
 ip router isis test_ip_isis
 no ptp enable
 mpls ip
 bfd interval 50 min_rx 50 multiplier 3
 isis bfd
!
router isis test_ip_isis
 net 47.0004.004d.0055.0000.0c00.0002.00
 net 47.0004.004d.0056.0000.0c00.0002.00
 is-type level-2-only
 redistribute connected
 bfd all-interfaces
!
```

CHAPTER 22

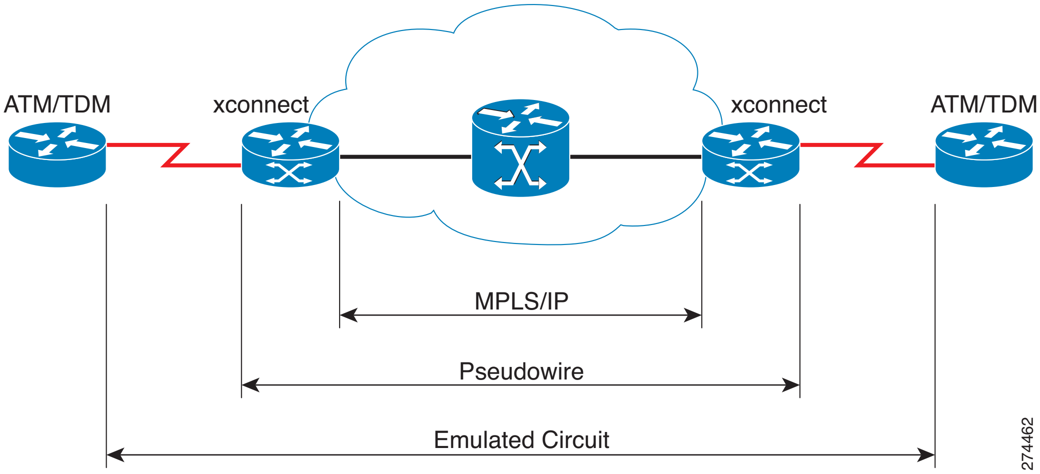
Configuring Pseudowire

Cisco Pseudowire Emulation Edge-to-Edge (PWE3) allows you to transport traffic using traditional services such as E1/T1 over a packet-based backhaul technology such as MPLS or IP. A pseudowire (PW) consists of a connection between two provider edge (PE) devices that connects two attachment circuits (ACs), such as ATM VPIs/VCIs or E1/T1 links.

The following sections describe how to configure pseudowire on the Cisco MWR 2941:

- Understanding Pseudowire, page 22-1
- Configuring Pseudowire, page 22-3
- Configuration Examples for Pseudowire, page 22-18

Figure 22-1 Cisco MWR 2941 Router in a PWE3—Example



Understanding Pseudowire

PWs manage encapsulation, timing, order, and other operations in order to make it transparent to users; the PW tunnel appears as an unshared link or circuit of the emulated service.

There are limitations that impede some applications from utilizing a PW connection. For more information, see the section describing the PW service.

Cisco supports the following standards-based PWE types:

Structure-Agnostic TDM over Packet

SAToP encapsulates TDM bit-streams (T1, E1, T3, E3) as PWs over PSNs. It disregards any structure that may be imposed on streams, in particular the structure imposed by the standard TDM framing.

The protocol used for emulation of these services does not depend on the method in which attachment circuits are delivered to the PEs. For example, a T1 attachment circuit is treated the same way for all delivery methods, including: PE on copper, multiplex in a T3 circuit, mapped into a virtual tributary of a SONET/SDH circuit, or carried over a network using unstructured Circuit Emulation Service (CES). Termination of specific carrier layers used between the PE and circuit emulation (CE) is performed by an appropriate network service provider (NSP).

For instructions on how to configure SAToP, see Configuring Structure-Agnostic TDM over Packet (SAToP). For a sample SAToP configuration, see Configuration Examples for Pseudowire.

Structure-Aware TDM Circuit Emulation Service over Packet-Switched Network

CESoPSN encapsulates structured (NxDS0) TDM signals as PWs over PSNs. It complements similar work for structure-agnostic emulation of TDM bit-streams, such as PWE3-SAToP.

Emulation of NxDS0 circuits saves PSN bandwidth and supports DS0-level grooming and distributed cross-connect applications. It also enhances resilience of CE devices due to the effects of loss of packets in the PSN.

CESoPSN supports channel-associated signaling (CAS) for E1 and T1 interfaces. CAS provides signaling information within each DS0 channel as opposed to using a separate signaling channel. CAS also referred to as in-band signaling or robbed bit signaling.

For instructions on how to configure SAToP, see Configuring Circuit Emulation Service over Packet-Switched Network (CESoPSN). For a sample SAToP configuration, see Configuration Examples for Pseudowire.

Transportation of Service Using ATM over MPLS

An Asynchronous Transfer Mode (ATM) over MPLS PW is used to carry ATM cells over an MPLS network. It is an evolutionary technology that allows you to migrate packet networks from legacy networks, yet provides transport for legacy applications. ATM over MPLS is particularly useful for transporting 3G voice traffic over MPLS networks.

You can configure ATM over MPLS in the following modes:

- N-to-1 Cell Mode—Maps one or more ATM virtual channel connections (VCCs) or virtual permanent connection (VPCs) to a single pseudowire.
- 1-to-1 Cell Mode—Maps a single ATM VCC or VPC to a single pseudowire.
- Port Mode—Map one physical port to a single pseudowire connection.

The Cisco MWR 2941 also supports cell packing and PVC mapping for ATM over MPLS pseudowires.

Note Release 15.0(1)MR does not support ATM over MPLS N-to-1 Cell Mode or 1-to-1 Cell Mode.

For more information about how to configure ATM over MPLS, see Configuring Transportation of Service Using ATM over MPLS. For sample ATM over MPLS configurations, see Configuration Examples for Pseudowire.

Transportation of Service Using Ethernet over MPLS

Ethernet over MPLS (EoMPLS) PWs provide a tunneling mechanism for Ethernet traffic through an MPLS-enabled Layer 3 core network. EoMPLS PWs encapsulate Ethernet protocol data units (PDUs) inside MPLS packets and use label switching to forward them across an MPLS network. EoMPLS PWs are an evolutionary technology that allows you to migrate packet networks from legacy networks while providing transport for legacy applications. EoMPLS PWs also simplify provisioning, since the provider edge equipment only requires Layer 2 connectivity to the connected customer edge (CE) equipment. The Cisco MWR 2941 implementation of EoMPLS PWs is compliant with the RFC 4447 and 4448 standards.

For instructions on how to create an EoMPLS PW, see Configuring Transportation of Service Using Ethernet over MPLS.

Limitations

When configuring an EoMPLS pseudowire on the Cisco MWR 2941, you cannot configure an IP address on the same interface as the pseudowire.

Configuring Pseudowire

This section describes how to configure pseudowire on the Cisco MWR 2941. The Cisco MWR 2941 supports pseudowire connections using SATоP, CESoPSN, and ATM over MPLS. The following sections describe how to configure pseudowire connections on the Cisco MWR 2941.

- Using Pseudowire Classes, page 22-4
- Using CEM Classes, page 22-5
- Configuring a Backup Peer, page 22-6
- Configuring Structure-Agnostic TDM over Packet (SAToP), page 22-7
- Configuring Circuit Emulation Service over Packet-Switched Network (CESoPSN), page 22-7
- Configuring Transportation of Service Using ATM over MPLS, page 22-10
- Configuring Transportation of Service Using Ethernet over MPLS, page 22-17

For full descriptions of each command, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*. For pseudowire configuration examples, see Configuration Examples for Pseudowire, page 22-18

Using Pseudowire Classes

A pseudowire class allows you to create a single configuration template for multiple pseudowire connections. You can apply pseudowire classes to all pseudowire types. Follow these steps to configure a pseudowire class:

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)# pseudowire-class newclass`	Creates a new pseudowire class.
Step 4	`Router(config-pw-class)# encapsulation mpls`	Sets an encapsulation type. For an ATM over MPLS pseudowire, use **mpls**. For a CESoPSN pseudowire using UDP encapsulation, use **udp**.
Step 5	`Router(config-pw-class)# mpls experimental 5`	Specifies the 3-bit EXP field in the MPLS label used for pseudowire packets. **Note** For more information about the **mpls experimental** command, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.
Step 6	`Router(config-pw-class)# preferred-path peer 50.0.0.1`	Specifies a preferred path if there are multiple paths that traffic can cross within the pseudowire class. **Note** This command applies only to MPLS pseudowires.

Step 7	`Router(config)# interface atm0/ima0`
	`Router(config-if)# pvc 0/40 l2transport`
	`Router(cfg-if-atm-l2trans-pvc)# encapsulation aal0`
		Configures the pseudowire interface to use for the new pseudowire class. This example shows an ATM IMA interface.

Step 8	`Router(cfg-if-atm-l2trans-pvc)# xconnect 1.1.1.1 40 pw-class myclass`	Binds an attachment circuit to the ATM IMA interface to create an ATM pseudowire. Use the **pw-class** parameter to specify the pseudowire class that the ATM pseudowire interface uses.

Step 9	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Note You cannot use the encapsulation **mpls** parameter with the **pw-class** parameter.

Note The use of the **xconnect** command can vary depending on the type of pseudowire you are configuring.

Using CEM Classes

A CEM class allows you to create a single configuration template for multiple CEM pseudowires. Follow these steps to configure a CEM class:

Note You cannot apply a CEM class to other pseudowire types such as ATM over MPLS.

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)#` **`class cem mycemclass`**	Creates a new CEM class

Step 4	`Router(config-cem-class)#` **`payload-size 512`**
	`Router(config-cem-class)#` **`dejitter-buffer 10`**
	`Router(config-cem-class)#` **`idle-pattern 0x55`**
		Enter the configuration commands common to the CEM class. This example specifies a sample rate, payload size, dejitter buffer, and idle pattern.

Step 5	`Router(config-cem-class)#` **`exit`**	Returns to the config prompt.

Step 6	`Router(config)#` **`interface cem 0/0`**
	`Router(config-if)#` **`no ip address`**
	`Router(config-if)#` **`cem 0`**
	`Router(config-if-cem)#` **`cem class mycemclass`**
	`Router(config-if-cem)#` **`xconnect 10.10.10.10 200 encapsulation mpls`**
		Configure the CEM interface that you want to use for the new CEM class.
		Note The use of the **xconnect** command can vary depending on the type of pseudowire you are configuring.

Step 7	`Router(config-if-cem)#` **`exit`**
	`Router(config-if)#`
		Exits the CEM interface.

Step 8	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Configuring a Backup Peer

A backup peer provides a redundant pseudowire (PW) connection in the case that the primary PW loses connection; if the primary PW goes down, the Cisco MWR 2941 diverts traffic to the backup PW. Follow these steps to configure a backup peer.

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# `**`configure terminal`**
		Enters global configuration mode.

Step 3	`Router(config)# `**`backup peer`** *`peer-router-ip-address vcid`* `[`**`pw-class`** *`pw-class name`*`]`	Defines the address and VC of the backup peer.
Step 4	`Router(config)# `**`backup delay`** *`enable-delay`* `{`*`disable-delay`* `|` **`never`**`}`	Specifies the delay before the router switches pseudowire traffic to the backup peer VC. Where: • *enable-delay*—Time before the backup PW takes over for the primary PW. • *disable-delay*—Time before the restored primary PW takes over for the backup PW. • **never**—Disables switching from the backup PW to the primary PW.

Step 5	**exit**
	Example:
	`Router(config)# `**`exit`**
	`Router#`
		Exits configuration mode.

Configuring Structure-Agnostic TDM over Packet (SAToP)

Follow these steps to configure SAToP on the Cisco MWR 2941:

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router>` **`enable`**
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router#` **`configure terminal`**
		Enters global configuration mode.

Step 3	`Router(config)#` **`controller [T1|E1] 0/4`**
	`Router(config-controller)#`
		Configures the T1 or E1 interface.

Step 4	`Router(config-if)#` **`cem-group 4 unframed`**	Assigns channels on the T1 or E1 circuit to the CEM channel. This example uses the **unframed** parameter to assign all the T1 timeslots to the CEM channel.

Step 5	`Router(config)#` **`interface CEM0/4`**
	`Router(config-if)#` **`no ip address`**
	`Router(config-if)#` **`cem 4`**
		Defines a CEM group.

Step 6	`Router(config-if)#` **`xconnect 30.30.30.2 304 encapsulation mpls`**	Binds an attachment circuit to the CEM interface to create a pseudowire. This example creates a pseudowire by binding the CEM circuit 304 to the remote peer 30.30.2.304.

Step 7	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Note When creating IP routes for a pseudowire configuration, we recommend that you build a route from the xconnect address (LDP router-id or loopback address) to the next hop IP address, such as **ip route 30.30.30.2 255.255.255.255 1.2.3.4**.

Configuring Circuit Emulation Service over Packet-Switched Network (CESoPSN)

Follow these steps to configure CESoPSN on the Cisco MWR 2941.

Note To configure a CESoPSN pseudowire with UDP encapsulation, see Configuring a CESoPSN Pseudowire with UDP Encapsulation, page 22-9.

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

Step 3	`Router(config)#` **`controller [e1|t1] 0/0`**
	`Router(config-controller)#`
		Enters configuration mode for the E1 or T1 controller.

Step 4	`Router(config-controller)#` **`mode {atm	cas}`**	Sets the controller in asynchronous transfer mode (ATM) or channel-associated signaling (CAS) mode.

Step 5	`Router(config-controller)#`
	`cem-group 5 timeslots 1-24`
		Assigns channels on the T1 or E1 circuit to the circuit emulation (CEM) channel. This example uses the **timeslots** parameter to assign specific timeslots to the CEM channel.

Step 6	`Router(config-controller)#` **`exit`**
	`Router(config)#`
		Exits controller configuration.

Step 7	`Router(config)#` **`interface CEM0/5`**
	`Router(config-if-cem)#` **`cem 5`**
	`Router(config-if-cem)#` **`signaling inband-cas`**
		Defines a CEM channel.

Step 8	`Router(config-if-cem)#` **`xconnect 30.30.30.2 305 encapsulation mpls`**	Binds an attachment circuit to the CEM interface to create a pseudowire. This example creates a pseudowire by binding the CEM circuit 5 to the remote peer 30.30.30.2. **Note** When creating IP routes for a pseudowire configuration, we recommend that you build a route from the xconnect address (LDP router-id or loopback address) to the next hop IP address, such as **`ip route 30.30.30.2 255.255.255.255 1.2.3.4`**.

Step 9	`Router(config-if-cem)#` **`exit`**
	`Router(config)#`
		Exits the CEM interface.

Step 10	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Configuring a CESoPSN Pseudowire with UDP Encapsulation

Follow these steps to configure a CESoPSN pseudowire with UDP encapsulation:

	Command	Purpose
Step 1	**enable** **Example:** Router> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Router# **configure terminal**	Enters global configuration mode.
Step 3	Router(config)# **pseudowire-class udpClass**	Creates a new pseudowire class.
Step 4	Router(config-pw-class)# **encapsulation udp**	Specifies the UDP transport protocol.
Step 5	Router(config-pw-class)# **ip local interface Loopback1**	Configures the IP address of the provider edge (PE) router interface to be used as the source IP address for sending tunneled packets.
Step 6	Router(config-pw-class)# **ip tos value 100**	Specifies the type of service (ToS) level for IP traffic in the pseudowire.
Step 7	Router(config-pw-class)# **ip ttl 100**	Specifies a value for the time-to-live (TTL) byte in the IP headers of Layer 2 tunneled packets.
Step 8	Router(config-pw-class)# **exit** Router(config)#	Exits pseudowire-class configuration mode.
Step 9	Router(config)# **controller [e1|t1] 0/0** Router(config-controller)#	Enters E1 or T1 controller configuration mode.
Step 10	Router(config-controller)# **cem-group 5 timeslots 1-24**	Assigns channels on the T1 or E1 circuit to the circuit emulation (CEM) channel. This example uses the **timeslots** parameter to assign specific timeslots to the CEM channel.
Step 11	Router(config-controller)# **exit** Router(config)#	Exits controller configuration.
Step 12	Router(config)# **interface CEM0/5** Router(config-if-cem)# **cem 5**	Defines a CEM channel.
Step 13	Router(config-if-cem)# **xconnect 30.30.30.2 305 pw-class udpClass**	Binds an attachment circuit to the CEM interface to create a pseudowire. This example creates a pseudowire by binding the CEM circuit 5 to the remote peer 30.30.30.2. **Note** When creating IP routes for a pseudowire configuration, we recommend that you build a route from the xconnect address (LDP router-id or loopback address) to the next hop IP address, such as **ip route 30.30.30.2 255.255.255.255 1.2.3.4**.
Step 14	Router(config-if-cem)# **udp port local 50000 remote 55000**	Specifies a local and remote UDP port for the connection. **Note** Valid port values for CESoPSN pseudowires using UDP are from 49152–57343.

	Command	Purpose

Step 15	`Router(config-if-cem)# exit`
	`Router(config)#`
		Exits the CEM interface.

Step 16	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Configuring Transportation of Service Using ATM over MPLS

ATM over MPLS pseudowires allow you to encapsulate and transport ATM traffic across an MPLS network. This service allows you to deliver ATM services over an existing MPLS network.

The following sections describe how to configure transportation of service using ATM over MPLS:

- Configuring the Controller
- Configuring an IMA Interface
- Configuring the ATM over MPLS Pseudowire Interface

Note For sample configurations for ATM over MPLS, see Configuration Examples for Pseudowire.

Configuring the Controller

Follow these steps to configure the controller.

	Command	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)# card type e1 0 0`	Configures IMA on an E1 or T1 interface.

Step 4	`Router(config)# controller E1 0/4`
	`Router(config-controller)#`
		Specifies the controller interface on which you want to enable IMA.

Step 5	`Router(config-controller)# clock source internal`	Sets the clock source to internal.

	Command	Purpose

Step 6	`Router(config-controller)#`
	`ima-group 0 scrambling-payload`
		If you want to configure an ATM IMA backhaul, use the **ima-group** command to assign the interface to an IMA group. For a T1 connection, use the **no-scrambling-payload** to disable ATM-IMA cell payload scrambling; for an E1 connection, use the **scrambling-payload** parameter to enable ATM-IMA cell payload scrambling.
		The example assigns the interface to IMA group 0 and enables payload scrambling.

Step 7	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Note For more information about configuring IMA groups, see the "Configuring ATM IMA" section on page 18-6.

Configuring an IMA Interface

If you want to use ATM IMA backhaul, follow these steps to configure the IMA interface.

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

	Command	Purpose

Step 3	`Router(config-controller)#`
	`interface ATM`*`slot`***`/IMA`***`group-number`*
	
	Example:
	`Router(config-controller)#`
	`interface atm0/ima0`
	`Router(config-if)#`
		Specifies the slot location and port of IMA interface group. The syntax is as follows:
		• *slot*—The slot location of the ATM IMA port adapter.
		• *group-number*—The group number of the IMA group.
		The example specifies the slot number as 0 and the group number as 0.
		Note To explicitly configure the IMA group ID for the IMA interface, you may use the optional **ima group-id** command. You cannot configure the same IMA group ID on two different IMA interfaces; therefore, if you configure an IMA group ID with the system-selected default ID already configured on an IMA interface, the system toggles the IMA interface to make the user-configured IMA group ID the effective IMA group ID. At the same, the system toggles the original IMA interface to select a different IMA group ID.

Step 4	`Router(config-if)#` **`no ip address`**	Disables the IP address configuration for the physical layer interface.
Step 5	`Router(config-if)#` **`atm bandwidth dynamic`**	Specifies the ATM bandwidth as dynamic.
Step 6	`Router(config-if)#` **`no atm ilmi-keepalive`**	Disables the ILMI keepalive parameters.

Step 7	**exit**
	
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

For more information about configuring IMA groups, see the "Configuring ATM IMA" section on page 18-6.

Configuring the ATM over MPLS Pseudowire Interface

You can configure ATM over MPLS is several modes according to the needs of your network. Use the appropriate section according to the needs of your network. You can configure the following ATM over MPLS pseudowire types:

- Configuring N-to-1 VCC Cell Transport Pseudowire—Maps multiple VCCs to a single pseudowire
- Configuring N-to-1 VPC Cell Transport—Maps multiple VPCs to a single pseudowire
- Configuring ATM AAL5 SDU VCC Transport—Maps a single ATM PVC to another ATM PVC
- Configuring a Port Mode Pseudowire—Maps one physical port to a single pseudowire connection
- Optional Configurations

Note Release 15.0(1)MR does not support N-to-1 VCC Cell Transport for mapping multiple PVCs, 1-to-1 VCC Cell Mode, or PVC mapping.

Note When creating IP routes for a pseudowire configuration, build a route from the xconnect address (LDP router-id or loopback address) to the next hop IP address, such as **ip route 30.30.30.2 255.255.255.255 1.2.3.4**.

Configuring N-to-1 VCC Cell Transport Pseudowire

An N-to-1 VCC cell transport pseudowire maps one or more ATM virtual channel connections (VCCs) to a single pseudowire. Follow these steps to configure an N-to-1 pseudowire.

You can use the following methods to configure an N-to-1 VCC Cell Transport pseudowire.

Mapping a Single PVC to a Pseudowire

To map a single PVC to an ATM over MPLS pseudowire, apply the **xconnect** command at the PVC level. This configuration type only uses AAL0 encapsulation. Follow these steps to map a single PVC to an ATM over MPLS pseudowire.

Note Release 15.0(1)MR does not support mapping multiple VCCs to a pseudowire.

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)#` **`interface atm0/ima0`**	Configures the ATM IMA interface.

Step 4	`Router(config-if)#` **`pvc 0/40 l2transport`**
	`Router(cfg-if-atm-l2trans-pvc)#`
		Defines a PVC. Use the **l2transport** keyword to configure the PVC as layer 2 virtual circuit.

Step 5	`Router(cfg-if-atm-l2trans-pvc)#` **`encapsulation aal0`**	Defines the encapsulation type for the PVC.

Step 6	`Router(config-if)#` **`xconnect 1.1.1.1 40 encapsulation mpls`**
	`Router(cfg-if-atm-l2trans-pvc-xconn)#`
		Binds an attachment circuit to the ATM IMA interface to create a pseudowire. This example creates a pseudowire by binding PVC 40 to the remote peer 1.1.1.1.

Step 7	`Router(cfg-if-atm-l2trans-pvp-xconn)#` **`end`**
	`Router#`
		Exits configuration mode.

Configuring N-to-1 VPC Cell Transport

An N-to-1 VPC cell transport pseudowire maps one or more ATM virtual path connections (VPCs) to a single pseudowire. While the configuration is similar to one-to-one VPC cell mode, this transport method uses the N-to-1 VPC Pseudowire protocol and format defined in RFCs 4717 and 4446. Follow these steps to configure an N-to-1 VPC pseudowire.

Note Release 15.0(1)MR does not support mapping multiple VPCs to a pseudowire.

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		- Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)#` **`interface atm0/ima0`**
	`Router(config-if)#`
		Configures the ATM IMA interface.

Step 4	`Router(config-if)#` **`atm pvp 10 l2transport`**
	`Router(cfg-if-atm-l2trans-pvp)#`
		Maps a PVP to a pseudowire

Step 5	`Router(cfg-if-atm-l2trans-pvp)#`
	`xconnect 30.30.30.2 305 encapsulation mpls`
	`Router(cfg-if-atm-l2trans-pvp-xconn)#`
		Binds an attachment circuit to the ATM IMA interface to create a pseudowire. This example creates a pseudowire by binding the ATM circuit 305 to the remote peer 30.30.30.2.

Step 6	`Router(cfg-if-atm-l2trans-pvp-xconn)#`
	`end`
	`Router#`
		Exits configuration mode.

Configuring ATM AAL5 SDU VCC Transport

An ATM AAL5 SDU VCC transport pseudowire maps a single ATM PVC to another ATM PVC. Follow these steps to configure an ATM AAL5 SDU VCC transport pseudowire.

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		- Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)# interface atm 0/ima0`
	`Router(config-if)#`
		Configures the ATM IMA interface.

	Command	Purpose

Step 4	`Router(config-if)# pvc 0/12 l2transport`
	`Router(cfg-if-atm-l2trans-pvc)#`
		Configures a PVC and specify a VCI/VPI.

Step 5	`Router(cfg-if-atm-l2trans-pvc)# encapsulation aal5`	Sets the PVC encapsulation type to AAL5. **Note** You must use AAL5 encapsulation for this transport type.
Step 6	`Router(cfg-if-atm-l2trans-pvc)# xconnect 25.25.25.25 125 encapsulation mpls`	Binds an attachment circuit to the ATM IMA interface to create a pseudowire. This example creates a pseudowire by binding the ATM circuit 125 to the remote peer 25.25.25.25.

Step 7	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Configuring a Port Mode Pseudowire

A port mode pseudowire allows you to map an entire ATM interface to a single pseudowire connection. Follow these steps to configure a port mode pseudowire:

	Command	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)# interface atm 0/ima0`	Configures the ATM interface.
Step 4	`Router(cfg-if)# xconnect 25.25.25.25 2000 encapsulation mpls`	Binds an attachment circuit to the ATM IMA interface to create a pseudowire. This example creates a pseudowire by binding the ATM circuit 200 to the remote peer 25.25.25.25.

Step 5	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Optional Configurations

You can apply the following optional configurations to a pseudowire link.

Configuring Cell Packing

Cell packing allows you to improve the efficiency of ATM-to-MPLS conversion by packing multiple ATM cells into a single MPLS packet. Follow these steps to configure cell packing.

	Command	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		- Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)# int atm1/0`	Configures the ATM interface.

Step 4	`Router(config)# int atm1/0`
	`Router(config-if)# atm mcpt-timers 1000 2000 3000`
		Defines the three Maximum Cell Packing Timeout (MCPT) timers under an ATM interface. The three independent MCPT timers specify a wait time before forwarding a packet.

Step 5	`Router(config)# pvc 0/11 l2transport`
	`Router(cfg-if-atm-l2trans-pvc)# encapsulation aal0`
	`Router(cfg-if-atm-l2trans-pvc)# cell-packing 20 mcpt-timer 3`
		Specifies the maximum number of cells in PW cell pack and the cell packing timer that the Cisco MWR 2941 uses. This example specifies 20 cells per pack and the third MCPT timer.

Step 6	**end**
	Example:
	`Router(cfg-if-atm-l2trans-pvc)# end`
	`Router#`
		Exits configuration mode.

Configuring Transportation of Service Using Ethernet over MPLS

Ethernet over MPLS PWs allow you to transport Ethernet traffic over an existing MPLS network. For an overview of Ethernet over MPLS pseudowires, see Transportation of Service Using Ethernet over MPLS, page 22-3.

Configuring VLAN Mode

An Ethernet over MPLS pseudowire in VLAN mode creates a connection based on an existing VLAN ID on the Cisco MWR 2941. Follow these steps to configure an Ethernet over MPLS pseudowire in VLAN mode.

	Command	Purpose
Step 1	**enable** **Example:** Router> enable	Enables privileged EXEC mode. - Enter your password if prompted.
Step 2	**configure terminal** **Example:** Router# configure terminal	Enters global configuration mode.
Step 3	Router(config)# interface vlan 100	Creates the VLAN interface to bind to a pseudowire.
Step 4	Router(config-if)# **xconnect 1.1.1.2 101 encapsulation mpls**	Binds the Ethernet port interface to an attachment circuit to create a pseudowire. This example uses virtual circuit (VC) 101 to uniquely identify the PW. Ensure that the remote VLAN is configured with the same VC. **Note** When creating IP routes for a pseudowire configuration, we recommend that you build a route from the xconnect address (LDP router-id or loopback address) to the next hop IP address, such as **ip route 30.30.30.2 255.255.255.255 1.2.3.4**.
Step 5	Router(config-if)# **interface GigabitEthernet 0/1** Router(config-if)# **switchport trunk allowed vlan 100** Router(config-if)# **switchport mode trunk**	Adds the GigabitEthernet interface to the VLAN.
Step 6		Creates a corresponding configuration on the remote router with the same VCID value. This configuration uses VCID 101.
Step 7	**exit** **Example:** Router(config)# exit Router#	Exits configuration mode.

Note The Cisco MWR 2941 supports VLAN rewriting on EoMPLS PWs. If the two networks use different VLAN IDs, the router rewrites PW packets using the appropriate VLAN number for the local network.

Note For more information about configuring VLANs on the Cisco MWR 2941, see the "Configuring VLANs" section on page 7-1.

Configuration Examples for Pseudowire

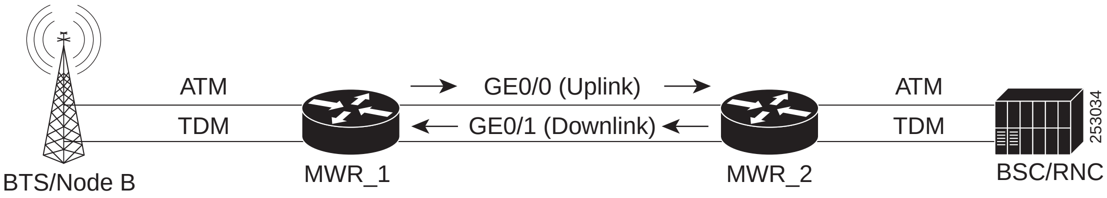
The following sections contain full configuration examples for pseudowire connections.

- Asymmetric PWE3 Configuration, page 22-18
- PWE3 Redundancy Configuration, page 22-26
- TDM over MPLS Configuration, page 22-30
- ATM over MPLS Configuration, page 22-34
- Ethernet over MPLS Configuration, page 22-39

Asymmetric PWE3 Configuration

The following example shows an Asymmetric PWE3 configuration (Figure 22-2).

Figure 22-2 Asymmetric PWE3 Configuration



MWR_1

```
version 12.4
service timestamps debug datetime msec localtime
service timestamps log datetime msec localtime

!
hostname MWR1
!
boot-start-marker
boot-end-marker
!
card type e1 0 0
card type e1 0 1
!
!
ip cef
!
!
controller E1 0/0
 clock source internal
 cem-group 1 unframed
```

```
!
controller E1 0/1
 clock source internal
 cem-group 20 unframed
!
controller E1 0/2
 clock source internal
 cem-group 12 unframed
!
controller E1 0/3
 clock source internal
 cem-group 30 unframed
!
controller E1 0/4
 clock source internal
 cem-group 8 unframed
!
controller E1 0/5
 clock source internal
 cem-group 25 unframed
!
controller E1 1/0
 mode atm
 clock source internal
!
controller E1 1/1
 mode atm
 clock source internal
!
controller E1 1/2
 mode atm
 clock source internal
!
controller E1 1/3
!
!
pseudowire-class mpls
 encapsulation mpls
 preferred-path peer 50.0.0.2
!
!
interface Loopback50
 ip address 50.0.0.1 255.255.255.255
!
interface CEM0/0
 no ip address
 cem 1
  xconnect 50.0.0.2 1 encapsulation mpls
 !
!
interface Vlan 20
 ip address 20.0.0.1 255.0.0.0
 mpls ip
!
interface CEM0/1
 no ip address
 cem 20
  xconnect 50.0.0.2 2 encapsulation mpls
!
interface Vlan 60
 ip address 60.0.0.1 255.0.0.0
 mpls ip
!
interface CEM0/2
```

```
 no ip address
 cem 12
  xconnect 50.0.0.2 3 encapsulation mpls
 !
!
interface CEM0/3
 no ip address
 cem 30
  xconnect 50.0.0.2 4 encapsulation mpls
!
interface CEM0/4
 no ip address
 cem 8
  xconnect 50.0.0.2 5 encapsulation mpls
 !
!
interface CEM0/5
 no ip address
 cem 25
  xconnect 50.0.0.2 6 encapsulation mpls
!
interface GigabitEthernet0/0
switchport access vlan 20
duplex auto
speed auto
!
interface GigabitEthernet0/1
switchport access vlan 60
duplex auto
speed auto
!
interface ATM1/0
 no ip address
 load-interval 30
 scrambling-payload
  mcpt-timers 1000 5000 10000
 no ilmi-keepalive
 pvc 0/5 l2transport
  encapsulation aal0
  cell-packing 10 mcpt-timer 3
  xconnect 50.0.0.2 10 pw-class mpls
!
 pvc 0/6 l2transport
  xconnect 50.0.0.2 20 pw-class mpls
 !
 pvc 0/7 l2transport
  encapsulation aal0
  cell-packing 28 mcpt-timer 3
  xconnect 50.0.0.2 30 encapsulation mpls pw-class mpls one-to-one
 !
 pvc 0/8 l2transport
  xconnect 50.0.0.2 40 pw-class mpls
 !
 pvc 0/9 l2transport
  encapsulation aal0
  xconnect 50.0.0.2 50 pw-class mpls one-to-one
 !
!
interface ATM1/0.1 point-to-point
 pvc 0/15 l2transport
  xconnect 50.0.0.2 13 pw-class mpls
!
interface ATM1/0.2 multipoint
  cell-packing 2 mcpt-timer 1
 xconnect 50.0.0.2 12 encapsulation mpls
 pvc 0/10 l2transport
  encapsulation aal0
 !
 pvc 0/11 l2transport
  encapsulation aal0
 !
 pvc 0/12 l2transport
  encapsulation aal0
 !
 pvc 0/13 l2transport
  encapsulation aal0
 !
!
interface ATM1/0.3 point-to-point
 pvc 0/16 l2transport
  encapsulation aal0
  xconnect 50.0.0.2 14 encapsulation mpls
 !
!
interface ATM1/0.4 point-to-point
 pvc 0/17 l2transport
  encapsulation aal0
  xconnect 50.0.0.2 15 pw-class mpls one-to-one
 !
!
interface ATM1/0.6 multipoint
 pvc 0/26 l2transport
  xconnect 50.0.0.2 16 pw-class mpls
 !
 pvc 0/27 l2transport
  encapsulation aal0
  cell-packing 8 mcpt-timer 3
  xconnect 50.0.0.2 17 pw-class mpls
 !
 pvc 0/28 l2transport
  encapsulation aal0
  cell-packing 16 mcpt-timer 2
  xconnect 50.0.0.2 18 pw-class mpls
 !
!
interface ATM1/0.7 multipoint
!
interface ATM1/1
 no ip address
 scrambling-payload
  mcpt-timers 1000 5000 10000
 no  ilmi-keepalive
  cell-packing 20 mcpt-timer 2
 xconnect 50.0.0.2 11 encapsulation mpls
 pvc 0/21 l2transport
  encapsulation aal0
 !
 pvc 0/22 l2transport
  encapsulation aal0
 !
 pvc 0/23 l2transport
  encapsulation aal0
 !
!
interface ATM1/1.1 point-to-point
!
interface ATM1/1.2 multipoint
!
```

```
interface ATM1/2
 no ip address
 scrambling-payload
 ima-group 0
 no  ilmi-keepalive
!
ip route 50.0.0.2 255.255.255.255 20.0.0.2
!
ip http server
no ip http secure-server
!
!
mpls ldp router-id Loopback50 force
!
!
line con 0
 exec-timeout 0 0
line aux 0
line vty 0 4
 login
!
network-clock-select 1 BITS
!
end
```

MWR_2

```
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
!
hostname MWR2
!
boot-start-marker
boot-end-marker
!
card type e1 0 0
card type e1 0 1
!
enable password mypassword
!
no aaa new-model
!
ip cef
!
!
controller E1 0/0
 cem-group 1 unframed
!
controller E1 0/1
 cem-group 20 unframed
!
controller E1 0/2
 cem-group 12 unframed
!
controller E1 0/3
 cem-group 30 unframed
!
controller E1 0/4
 cem-group 8 unframed
!
controller E1 0/5
 cem-group 25 unframed
```

```
!
controller E1 1/0
 mode  atm
 clock source internal
!
controller E1 1/1
 mode  atm
 clock source internal
!
controller E1 1/2
 mode  atm
 clock source internal
!
controller E1 1/3
 clock source internal
!
pseudowire-class mpls
 encapsulation mpls
 preferred-path peer 50.0.0.1
!
!
interface Loopback50
 ip address 50.0.0.2 255.255.255.255
!
interface CEM0/0
 no ip address
 cem 1
  xconnect 50.0.0.1 1 encapsulation mpls
 !
!
interface Vlan20
ip address 20.0.0.2 255.0.0.0
mpls ip
!
interface Vlan60
ip address 60.0.0.2 255.0.0.0
mpls ip
!
interface GigabitEthernet0/0
switchport access vlan 20
duplex auto
speed auto
!
interface GigabitEthernet0/1
switchport access vlan 60
duplex auto
speed auto
!
!
interface CEM0/1
 no ip address
 cem 20
  xconnect 50.0.0.1 2 encapsulation mpls
 !
!
interface CEM0/2
 no ip address
 cem 12
  xconnect 50.0.0.1 3 encapsulation mpls
 !
!
interface CEM0/3
 no ip address
 cem 30
```

```
  xconnect 50.0.0.1 4 encapsulation mpls
 !
!
interface CEM0/4
 no ip address
 cem 8
  xconnect 50.0.0.1 5 encapsulation mpls
 !
!
interface CEM0/5
 no ip address
 cem 25
  xconnect 50.0.0.1 6 encapsulation mpls
 !
!
interface ATM1/0
 ip address 1.1.1.2 255.0.0.0
 load-interval 30
 scrambling-payload
  mcpt-timers 1000 5000 10000
 no  ilmi-keepalive
 pvc 0/5 l2transport
  encapsulation aal0
  cell-packing 25 mcpt-timer 3
  xconnect 50.0.0.1 10 pw-class mpls
 !
 pvc 0/6 l2transport
  xconnect 50.0.0.1 20 pw-class mpls
 !
 pvc 0/7 l2transport
  encapsulation aal0
  cell-packing 12 mcpt-timer 2
  xconnect 50.0.0.1 30 encapsulation mpls pw-class mpls one-to-one
 !
 pvc 0/8 l2transport
  xconnect 50.0.0.1 40 pw-class mpls
 !
 pvc 0/9 l2transport
  encapsulation aal0
  xconnect 50.0.0.1 50 pw-class mpls one-to-one
 !
 pvc 0/99
  protocol ip 1.1.1.1 broadcast
  encapsulation aal5snap
 !
!
interface ATM1/0.1 point-to-point
 pvc 0/15 l2transport
  xconnect 50.0.0.1 13 pw-class mpls
 !
!
interface ATM1/0.2 multipoint
  cell-packing 10 mcpt-timer 2
 xconnect 50.0.0.1 12 encapsulation mpls
 pvc 0/10 l2transport
  encapsulation aal0
 !
 pvc 0/11 l2transport
  encapsulation aal0
 !
 pvc 0/12 l2transport
  encapsulation aal0
 !
 pvc 0/13 l2transport
```

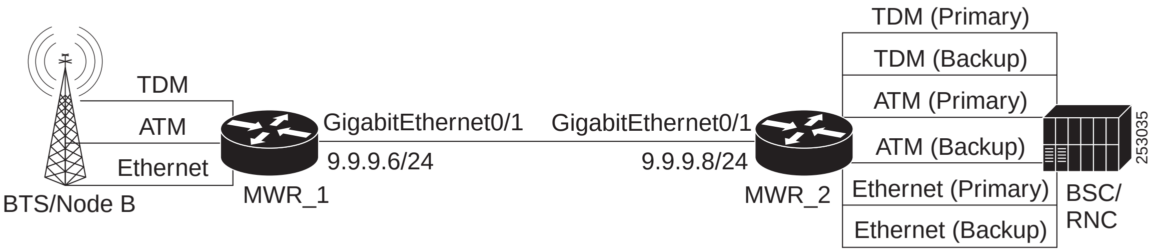
```
  encapsulation aal0
 !
!
interface ATM1/0.3 point-to-point
 pvc 0/16 l2transport
  encapsulation aal0
  xconnect 50.0.0.1 14 encapsulation mpls
 !
!
interface ATM1/0.4 point-to-point
 pvc 0/17 l2transport
  encapsulation aal0
  xconnect 50.0.0.1 15 pw-class mpls one-to-one
 !
!
interface ATM1/0.6 multipoint
 pvc 0/26 l2transport
  xconnect 50.0.0.1 16 pw-class mpls
 !
 pvc 0/27 l2transport
  encapsulation aal0
  cell-packing 18 mcpt-timer 3
  xconnect 50.0.0.1 17 pw-class mpls
 !
 pvc 0/28 l2transport
  encapsulation aal0
  cell-packing 24 mcpt-timer 2
  xconnect 50.0.0.1 18 pw-class mpls
 !
!
interface ATM1/0.7 multipoint
!
interface ATM1/1
 no ip address
 scrambling-payload
  mcpt-timers 1000 5000 10000
 no  ilmi-keepalive
  cell-packing 20 mcpt-timer 2
 xconnect 50.0.0.1 11 encapsulation mpls
 pvc 0/21 l2transport
  encapsulation aal0
 !
 pvc 0/22 l2transport
  encapsulation aal0
 !
 pvc 0/23 l2transport
  encapsulation aal0
 !
!
interface ATM1/2
 no ip address
 scrambling-payload
 ima-group 0
 no  ilmi-keepalive
!
ip route 50.0.0.1 255.255.255.255 60.0.0.1
!
!
ip http server
no ip http secure-server
!
!
mpls ldp router-id Loopback50 force
!
```

```
!
!
line con 0
 exec-timeout 0 0
line aux 0
line vty 0 4
 exec-timeout 0 0
 login
!
network-clock-select 1 BITS
!
end
```

PWE3 Redundancy Configuration

The following example shows a PWE3 Redundancy configuration (Figure 22-3).

Figure 22-3 PWE3 Redundancy Configuration



MWR_1

```
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
!
hostname mwr-1
!
boot-start-marker
boot-end-marker
!
card type e1 0 1
card type e1 0 2
!
ip cef
!
controller E1 0/0
 clock source internal
 cem-group 0 unframed
!
controller E1 0/1
!
controller E1 0/2
!
controller E1 0/3
 clock source internal
!
controller E1 1/0
 mode atm
 clock source internal
!
```

```
controller E1 1/1
!
controller E1 1/2
!
controller E1 1/3
 clock source internal
!
interface CEM0/0
cem 0
 xconnect 2.2.2.2 1 encapsulation mpls
 backup peer 2.2.2.2 2
 backup delay 20 20
!
interface ATM1/0
 no ip address
 scrambling-payload
 no  ilmi-keepalive
pvc 0/1 l2transport
  encapsulation aal0
xconnect 2.2.2.2 3 encapsulation mpls
 backup peer 2.2.2.2 4
 backup delay 20 20
!
interface Loopback0
 no ip address
!
interface Loopback1
 ip address 1.1.1.1 255.255.255.255
 load-interval 30
!
interface Loopback101
 no ip address
!
!
interface Vlan 9
 ip address 9.9.9.6 255.255.255.0
 mpls ip
!
interface Vlan 10
no ip address
no ptp enable
xconnect 2.2.2.2 10 encapsulation mpls
backup peer 2.2.2.2 20
!
interface GigabitEthernet0/1
switchport access vlan 9
duplex auto
speed auto
!
interface GigabitEthernet0/2
switchport access vlan 10
duplex auto
speed auto
!
!
ip forward-protocol nd
ip route 2.2.2.2 255.255.255.255  9.9.9.8
!

!
control-plane
!
!
line con 0
```

```
 exec-timeout 0 0
 logging synchronous
line aux 0
line vty 0 4
 exec-timeout 0 0
 password mypassword
 login
!
exception data-corruption buffer truncate
!
end
```

MWR_2

```
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
!
hostname mwr-pe2
!
boot-start-marker
boot-end-marker
!
card type e1 0 0
card type e1 0 1
card type e1 0 2
!
!
ip cef
!
!
controller E1 0/0
 cem-group 0 unframed
!
controller E1 0/1
 clock source internal
 cem-group 0 unframed
!
controller E1 0/2
!
controller E1 0/3
 clock source internal
!
controller E1 0/4
 clock source internal
 !
controller E1 0/5
!
controller E1 1/0
 mode  atm
 clock source internal
!
controller E1 1/1
 clock source internal
 !
controller E1 1/2
 clock source internal
 !
controller E1 1/3
 mode  atm
 clock source internal
!
! Primary
```

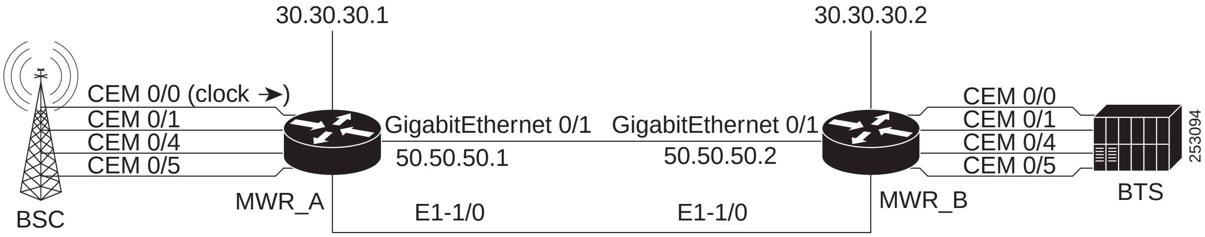
```
interface CEM0/0
cem 0
 xconnect 1.1.1.1 1 encapsulation mpls
!
! Backup
interface CEM0/1
cem 0
 xconnect 1.1.1.1 2 encapsulation mpls
!
! Primary
interface ATM1/0
 no ip address
 scrambling-payload
 no  ilmi-keepalive
pvc 0/1 l2transport
  encapsulation aal0
  xconnect 1.1.1.1 3 encapsulation mpls
 !
! Backup
interface ATM1/3
no ip address
 scrambling-payload
 no  ilmi-keepalive
pvc 0/1 l2transport
  encapsulation aal0
  xconnect 1.1.1.1 4 encapsulation mpls
 !
!
interface Loopback1
 ip address 2.2.2.2 255.255.255.255
!
!
interface Vlan 9
 ip address 9.9.9.8 255.255.255.0
 mpls ip
!
interface Vlan 10
no ip address
no ptp enable
xconnect 1.1.1.1 10 encapsulation mpls
!
interface Vlan 20
no ip address
no ptp enable
xconnect 1.1.1.1 20 encapsulation mpls
!
interface GigabitEthernet0/1
switchport access vlan 9
duplex auto
speed auto
!
interface GigabitEthernet0/2
switchport access vlan 10
duplex auto
speed auto
!
interface GigabitEthernet0/3
switchport access vlan 20
duplex auto
speed auto
!
!
ip forward-protocol nd
ip route 1.1.1.1 255.255.255.255  9.9.9.6
```

```
!
!
mpls ldp router-id Loopback1 force
!
control-plane
!
no call rsvp-sync
!
!
!
line con 0
 exec-timeout 0 0
 logging synchronous
line aux 0
line vty 0 4
 exec-timeout 0 0
 password mypassword
 login
!
exception data-corruption buffer truncate
!
end
```

TDM over MPLS Configuration

Figure 22-4 shows a TDM over MPLS configuration. The configuration uses both SAToP and CESoPSN for E1 and T1.

Figure 22-4 TDM over MPLS Configuration



MWR_A

```
!
version 12.4
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
no service password-encryption
!
hostname mwr_A
!
boot-start-marker
boot-end-marker
!
card type e1 0 0
card type e1 0 1
enable password xxx
!
no aaa new-model
clock timezone est -5
!
```

```
ip cef
!
controller E1 0/0
cem-group 0 timeslots 1-31
description E1 CESoPSN example
!
controller E1 0/1
clock source internal
cem-group 1 unframed
description E1 SATOP example
!
controller E1 0/4
clock source internal
cem-group 4 unframed
description E1 SATOP example
!
controller E1 0/5
clock source internal
cem-group 5 timeslots 1-24
description E1 CESoPSN example
!
controller E1 1/0
clock source internal
!
controller E1 1/1
!
interface Loopback0
ip address 30.30.30.1 255.255.255.255
!
interface GigabitEthernet0/1
ip address 50.50.50.1 255.255.255.0
mpls ip
!
interface CEM0/0
no ip address
cem 0
  xconnect 30.30.30.2 300 encapsulation mpls
!
interface CEM0/1
no ip address
cem 1
  xconnect 30.30.30.2 301 encapsulation mpls
!
!
interface CEM0/4
no ip address
cem 4
  xconnect 30.30.30.2 304 encapsulation mpls
!
!
interface CEM0/5
no ip address
cem 5
  xconnect 30.30.30.2 305 encapsulation mpls
!
!
no ip classless
ip route 30.30.30.2 255.255.255.255 50.50.50.2
!
no ip http server
no ip http secure-server
!
line con 0
password xxx
```

```
login
line aux 0
password xxx
login
no exec
line vty 0 4
password xxx
login
!
network-clock-select 1 BITS
end
```

MWR_B

```
!
version 12.4
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
no service password-encryption
!
hostname mwr_B
!
boot-start-marker
boot-end-marker
!
card type e1 0 0
card type e1 0 1
enable password xxx
!
no aaa new-model
clock timezone est -5
!
ip cef
!
controller E1 0/0
clock source internal
cem-group 0 timeslots 1-31
description E1 CESoPSN example
!
controller E1 0/1
clock source internal
cem-group 1 unframed
description E1 SATOP example
!
controller E1 0/4
clock source internal
cem-group 4 unframed
description T1 SATOP example
!
controller E1 0/5
clock source internal
cem-group 5 timeslots 1-24
description T1 CESoPSN example
!
controller E1 1/0

!
controller E1 1/1
!
interface Loopback0
ip address 30.30.30.2 255.255.255.255
!
!
```

```
interface GigabitEthernet0/1
ip address 50.50.50.2 255.255.255.0
mpls ip
!
interface CEM0/0
no ip address
cem 0
  xconnect 30.30.30.1 300 encapsulation mpls
!
interface CEM0/1
no ip address
cem 1
  xconnect 30.30.30.1 301 encapsulation mpls
!
interface CEM0/4
no ip address
cem 4
  xconnect 30.30.30.1 304 encapsulation mpls
!
!
interface CEM0/5
no ip address
cem 5
  xconnect 30.30.30.1 305 encapsulation mpls
!
!
no ip classless
ip route 30.30.30.1 255.255.255.255 50.50.50.1
!
no ip http server
no ip http secure-server
!
line con 0
password xxx
login
line aux 0
password xxx
login
no exec
line vty 0 4
password xxx
login
!
network-clock-select 1 E1 1/0
end
```

CESoPSN with UDP Configuration

The following configuration uses CESoSPN with UDP encapsulation.

Note This section provides a partial configuration intended to demonstrate a specific feature.

```
interface Loopback0
ip address 2.2.2.8 255.255.255.255
!
pseudowire-class udpClass
encapsulation udp
```

```
protocol none
ip local interface Loopback 0
!
controller E1 0/13
clock source internal
cem-group 0 timeslots 1-31
!
interface cem 0/13
cem 0
xconnect 2.2.2.9 200 pw-class udpClass
udp port local 50000 remote 55000
```

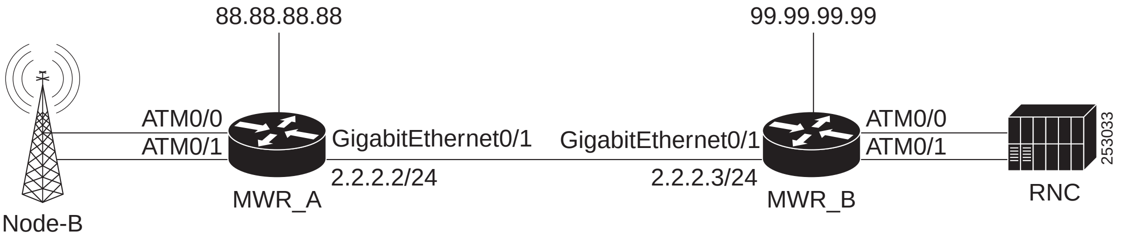
ATM over MPLS Configuration

This example shows how to accomplish the following configurations (Figure 22-5):

Note Release 15.0(1)MR does not support N-to-1 VCC Cell Transport for mapping multiple PVCs, 1-to-1 VCC Cell Mode, or PVC mapping.

- AAL5 SDU mode PW on 0/1 PVC 0/100
- N:1 VCC cell mode PW on 0/1 PVC 0/101
- Multiple PVCs N:1 VCC cell mode PW on 0/1.1
- 1:1 VCC cell mode PW on 0/1 PVC 0/102
- Cell-packing for port mode PWs
- VCC cell-relay mode PWs
- PVC mapping for 0/1.1 N:1 VCC cell relay PWs

Figure 22-5 ATM over MPLS Configuration



MWR_A

```
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
!
hostname mwr_A
!
boot-start-marker
boot-end-marker
!
card type e1 0 0
card type e1 0 1
```

```
logging buffered 4096
enable password mypassword
!
!
ip cef
!
!
no ip domain lookup
!
!
controller E1 0/0
 mode atm
 clock source internal
!
controller E1 0/1
 mode atm
 clock source internal
!
controller E1 0/2
 mode  atm
 clock source internal
!
controller E1 0/3
 mode  atm
 clock source internal
!
controller E1 0/4
!
controller E1 0/5
!
controller E1 1/0
!
controller E1 1/1
!
pseudowire-class mpls-exp-5
 encapsulation mpls
 mpls experimental 5
!
!
interface Loopback0
 ip address 88.88.88.88 255.255.255.255
!
interface ATM0/0
 no ip address
 scrambling-payload
  mcpt-timers 1000 2000 3000
 no  ilmi-keepalive
  cell-packing 28 mcpt-timer 3
 xconnect 99.99.99.99 100 encapsulation mpls
 pvc 1/35 l2transport
  encapsulation aal0
 !
 pvc 1/36 l2transport
  encapsulation aal0
 !
 pvc 1/37 l2transport
  encapsulation aal0
!
interface GigabitEthernet0/0
!
interface ATM0/1
 no ip address
 load-interval 30
 scrambling-payload
```

```
  mcpt-timers 1000 2000 3000
 no  ilmi-keepalive
 pvc 0/10
 !
 pvc 0/100 l2transport
  encapsulation aal5
  xconnect 99.99.99.99 1100 encapsulation mpls
 !
 pvc 0/101 l2transport
  encapsulation aal0
  cell-packing 28 mcpt-timer 3
  xconnect 99.99.99.99 1101 encapsulation mpls
 !
 pvc 0/102 l2transport
  encapsulation aal0
  cell-packing 28 mcpt-timer 3
  xconnect 99.99.99.99 1102 encapsulation mpls
 !
 pvc 0/103 l2transport
  encapsulation aal0
  cell-packing 28 mcpt-timer 3
  xconnect 99.99.99.99 1103 pw-class mpls-exp-5
 !
!
interface ATM0/1.1 multipoint
  cell-packing 28 mcpt-timer 3
 xconnect 99.99.99.99 1200 encapsulation mpls
 pvc 1/35 l2transport
  encapsulation aal0
  pw-pvc 2/135
 !
 pvc 1/36 l2transport
  encapsulation aal0
  pw-pvc 2/136
 !
 pvc 1/37 l2transport
  encapsulation aal0
  pw-pvc 2/137
 !
!
interface GigabitEthernet0/1
 description interface to 7600 fas 3/5
 ip address 2.2.2.2 255.255.255.0
 duplex auto
 speed auto
 mpls ip
 no keepalive
!
interface ATM0/2
 no ip address
 scrambling-payload
 no  ilmi-keepalive
!
interface ATM0/3
 no ip address
 scrambling-payload
 no  ilmi-keepalive
!
ip route 99.99.99.99 255.255.255.255 2.2.2.3
!
!
ip http server
no ip http secure-server
!
```

```
!
mpls ldp router-id Loopback0
!
!
line con 0
 exec-timeout 0 0
line aux 0
line vty 0 4
 exec-timeout 0 0
 privilege level 15
 password mypassword
 login
!
network-clock-select 1 E1 1/0
!
end
```

MWR_B

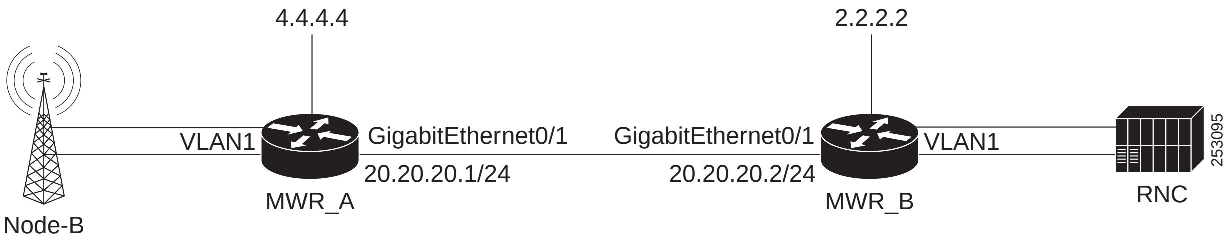
```
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
!
hostname mwr_B
!
boot-start-marker
boot-end-marker
!
card type e1 0 0
card type e1 0 1
logging buffered 4096
enable password mypassword
!
!
ip cef
!
!
no ip domain lookup
!
!
controller E1 0/0
 mode atm
!
controller E1 0/1
 mode atm
!
controller E1 0/2
 mode atm
!
controller E1 0/3
 mode atm
!
controller E1 0/4
!
controller E1 0/5
!
pseudowire-class mpls-exp-5
 encapsulation mpls
 mpls experimental 5
!
!
interface Loopback0
 ip address 99.99.99.99 255.255.255.255
```

```
!
interface ATM0/0
 no ip address
 scrambling-payload
  mcpt-timers 1000 2000 3000
 no  ilmi-keepalive
  cell-packing 28 mcpt-timer 3
 xconnect 88.88.88.88 100 encapsulation mpls
 pvc 1/35 l2transport
  encapsulation aal0
 !
 pvc 1/36 l2transport
  encapsulation aal0
 !
 pvc 1/37 l2transport
  encapsulation aal0
 !
!
interface GigabitEthernet0/0
!
interface ATM0/1
 no ip address
 scrambling-payload
  mcpt-timers 1000 2000 3000
 no  ilmi-keepalive
 pvc 0/2
 !
 pvc 0/100 l2transport
  encapsulation aal5
  xconnect 88.88.88.88 1100 encapsulation mpls
 !
 pvc 0/101 l2transport
  encapsulation aal0
  cell-packing 28 mcpt-timer 3
  xconnect 88.88.88.88 1101 encapsulation mpls
 !
 pvc 0/102 l2transport
  encapsulation aal0
  cell-packing 28 mcpt-timer 3
  xconnect 88.88.88.88 1102 encapsulation mpls
 !
 pvc 0/103 l2transport
  encapsulation aal0
  cell-packing 28 mcpt-timer 3
  xconnect 88.88.88.88 1103 pw-class mpls-exp-5
 !
interface ATM0/1.1 multipoint
  cell-packing 28 mcpt-timer 3
 xconnect 88.88.88.88 1200 encapsulation mpls
 pvc 2/135 l2transport
  encapsulation aal0
 !
 pvc 2/136 l2transport
  encapsulation aal0
 !
 pvc 2/137 l2transport
  encapsulation aal0
 !
!
interface GigabitEthernet0/1
 ip address 2.2.2.3 255.255.255.0
 duplex auto
 speed auto
 mpls ip
```

```
!
interface ATM0/2
 no ip address
 scrambling-payload
 ima-group 0
 no  ilmi-keepalive
!
interface ATM0/3
 no ip address
 scrambling-payload
 ima-group 0
 no  ilmi-keepalive
!
ip route 88.88.88.88 255.255.255.255 2.2.2.2
!
!
ip http server
no ip http secure-server
!
!
mpls ldp router-id Loopback0
!
!
line con 0
 exec-timeout 0 0
line aux 0
line vty 0 4
 exec-timeout 0 0
 password mypassword
 login
!
network-clock-select 1 E1 0/0
!
end
```

Ethernet over MPLS Configuration

The following configuration example shows an Ethernet pseudowire (aka EoMPLS) configuration.



MWR_A

```
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname mwr_A
!
```

```
boot-start-marker
boot-end-marker
!
card type e1 0 0
card type e1 0 1
logging buffered 4096
enable password mypassword
!
no aaa new-model
!
network-clock-select 1 E1 0/0
mmi polling-interval 60
no mmi auto-configure
no mmi pvc
mmi snmp-timeout 180
ip cef
!
no ip domain lookup
ip domain name cisco.com
multilink bundle-name authenticated
mpls label protocol ldp
vpdn enable
!
!
controller E1 0/0
 mode  aim 1
!
controller E1 0/1
 mode  aim 1
!
controller E1 0/2
 mode  aim 1
!
controller E1 0/3
 mode  aim 1
!
controller E1 0/4
!
controller E1 0/5
!
interface Loopback0
 ip address 4.4.4.4 255.255.255.255
!
interface GigabitEthernet0/4
 switchport trunk allowed vlan 1,2,20,1002-1005
 switchport mode trunk
!
interface GigabitEthernet0/5
 switchport trunk allowed vlan 1,2,40,1002-1005
 switchport mode trunk
!
interface Vlan20
 ip address 20.20.20.1 255.255.255.0
 no ptp enable
 mpls ip
!
interface Vlan40
 no ip address
 no ptp enable
 xconnect 2.2.2.2 10 encapsulation mpls
!
ip route 2.2.2.2 255.255.255.255 20.20.20.2
!
no ip http server
no ip http secure-server
!
!
mpls ldp router-id Loopback0
!
!
line con 0
 exec-timeout 0 0
line aux 0
line vty 0 4
 exec-timeout 0 0
 password mypassword
 login
!
end
```

MWR_B

```
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname mwr_B
!
boot-start-marker
boot-end-marker
!
card type e1 0 0
card type e1 0 1
logging buffered 4096
enable password mypassword
!
no aaa new-model
!
network-clock-select 1 E1 0/0
mmi polling-interval 60
no mmi auto-configure
no mmi pvc
mmi snmp-timeout 180
ip cef
!
no ip domain lookup
ip domain name cisco.com
multilink bundle-name authenticated
mpls label protocol ldp
vpdn enable
!
!
controller E1 0/0
 mode  aim 1
!
controller E1 0/1
 mode  aim 1
!
controller E1 0/2
 mode  aim 1
!
controller E1 0/3
 mode  aim 1
!
```

```
controller E1 0/4
!
controller E1 0/5
!
interface Loopback0
 ip address 2.2.2.2 255.255.255.255
!
interface GigabitEthernet0/4
 switchport trunk allowed vlan 1,2,20,1002-1005
 switchport mode trunk
!
interface GigabitEthernet0/5
 switchport trunk allowed vlan 1,2,40,1002-1005
 switchport mode trunk
!
interface Vlan20
 ip address 20.20.20.2 255.255.255.0
 no ptp enable
 mpls ip
!
interface Vlan40
 no ip address
 no ptp enable
 xconnect 4.4.4.4 10 encapsulation mpls
!
ip route 4.4.4.4 255.255.255.255 20.20.20.1
!
no ip http server
no ip http secure-server
!
!
mpls ldp router-id Loopback0
!
!
line con 0
 exec-timeout 0 0
line aux 0
line vty 0 4
 exec-timeout 0 0
 password mypassword
 login
!
end
```

CHAPTER 23

Configuring MPLS VPNs

A Virtual Private Network (VPN) is an IP-based network that delivers private network services over a public infrastructure. VPNs allow you to create a set of sites that can communicate privately over the Internet or other public or private networks.

The following sections describe how to configure MPLS VPNs on the Cisco MWR 2941:

- Understanding MPLS VPNs
- Configuring MPLS VPNs
- Sample MPLS VPN Configuration

Understanding MPLS VPNs

A conventional VPN consists of a full mesh of tunnels or permanent virtual circuits (PVCs) connecting all of the sites within the VPN. This type of VPN requires changes to each edge device in the VPN in order to add a new site. MPLS VPNs, also known as Layer 3 VPNs, are easier to manage and expand than conventional VPNs because they use layer 3 communication protocols and are based on a peer model. The peer model enables the service provider and customer to exchange Layer 3 routing information, enabling service providers to relay data between customer sites without customer involvement. The peer model also provides improved security of data transmission between VPN sites because data is isolated between improves security between VPN sites.

The Cisco MWR 2941 supports the following MPLS VPN types:

- Basic Layer 3 VPN—Provides a VPN private tunnel connection between customer edge (CE) devices in the service provider network. The provider edge (PE) router uses Multiprotocol Border Gateway Protocol (MP-BGP) to distribute VPN routes and MPLS Label Distribution Protocol (LDP) to distribute Interior Gateway Protocol (IGP) labels to the next-hop PE router.
- MPLS Carrier Supporting Carrier (CSC) VPN—Enables an MPLS VPN-based service provider to allow other service providers to use a segment of its backbone network. MPLS CSC VPNs use MPLS LDP to distribute MPLS labels and IGP to distribute routes.
- Inter-Autonomous System (AS) VPN—An inter-AS VPN allows service providers running separate networks to jointly offer MPLS VPN services to the same end customer; an inter-AS VPN can begin at one customer site and traverse multiple service provider backbones before arriving at another customer site.

Configuring MPLS VPNs

Layer 3 VPNs allow you to establish VPNs in a routed environment, improving the flexibility and ease of maintenance of VPNs. For instructions on how to configure layer 3 VPNs, see the *MPLS Configuration Guide, Cisco IOS Release 15.0S.*

Sample MPLS VPN Configuration

The following section shows a sample configuration for Layer 3 Virtual Private Network (VPN).

Note This section provides a partial configuration intended to demonstrate a specific feature.

```
!
-----------Customer definitions for 2 customers---------------------------------------
vrf definition customer_a
rd 192.168.1.1:100
route-target export 192.168.1.1:100
route-target import 192.168.1.1:100
!
address-family ipv4
exit-address-family
!
vrf definition customer_b
rd 192.168.2.1:200
route-target export 192.168.2.1:200
route-target import 192.168.2.1:200
!
address-family ipv4
exit-address-family
!
-------------------Loopback addresses for 2 customers-----------------------------------
interface Loopback100
vrf forwarding customer_a
ip address 192.169.1.3 255.255.255.255
!
interface Loopback101
vrf forwarding customer_b
ip address 192.168.100.1 255.255.255.255
!
------------------------Core-facing OSPF instance----------------------------
router ospf 1
log-adjacency-changes
network 100.0.0.0 0.255.255.255 area 0
network 192.168.0.0 0.0.255.255 area 0
network 192.169.0.0 0.0.255.255 area 0
!
----------------------VRF OSPF instances for 2 customers -------------------------------
router ospf 100 vrf customer_a
router-id 192.168.1.3
log-adjacency-changes
redistribute bgp 101 metric-type 1 subnets
network 192.168.0.0 0.0.255.255 area 0
network 192.169.0.0 0.0.255.255 area 0
!
router ospf 100 vrf customer_b
router-id 192.168.100.1
log-adjacency-changes
```

```
redistribute bgp 101 metric-type 1 subnets
network 192.168.0.0 0.0.255.255 area 0
network 192.169.0.0 0.0.255.255 area 0
!
----------------------MP-BGP with 2 VRF customers ---------------------------------
router bgp 101
bgp router-id 100.1.1.1
bgp log-neighbor-changes
neighbor 100.1.1.2 remote-as 101
neighbor 100.1.1.2 update-source Loopback1
!
address-family ipv4
redistribute connected
neighbor 100.1.1.2 activate
no auto-summary
no synchronization
exit-address-family
!
address-family vpnv4
neighbor 100.1.1.2 activate
neighbor 100.1.1.2 send-community extended
bgp scan-time import 5
exit-address-family
!
address-family ipv4 vrf customer_b
redistribute connected
neighbor 100.1.1.2 remote-as 101
neighbor 100.1.1.2 update-source Loopback1
neighbor 100.1.1.2 activate
no synchronization
exit-address-family
!
address-family ipv4 vrf customer_a
redistribute connected
neighbor 100.1.1.2 remote-as 101
neighbor 100.1.1.2 update-source Loopback1
neighbor 100.1.1.2 activate
no synchronization
exit-address-family
!
----------------MP-BGP loopback interface ---------------------------------
interface Loopback1
ip address 100.1.1.1 255.255.255.255
!
------------------Core-facing Vlan interface -------------------------------
interface GigabitEthernet0/1
switchport access vlan 20
switchport trunk allowed vlan 1,2,20-23,1002-1005
switchport mode trunk
load-interval 30
!
interface Vlan20
ip address 192.169.10.1 255.255.255.0
load-interval 30
no ptp enable
mpls ip
!
------------------CE-facing Vlan interfaces for 2 customers------------------------------
interface GigabitEthernet0/4
switchport access vlan 100
load-interval 30
duplex full
!
interface Vlan100
```

```
vrf forwarding customer_a
ip address 192.169.3.2 255.255.255.0
!
interface GigabitEthernet0/5
switchport access vlan 99
load-interval 30
duplex full
!
interface Vlan99
vrf forwarding customer_b
ip address 192.169.3.2 255.255.255.0
!
```

CHAPTER 24

Configuring Quality of Service

QoS refers to the ability of a network to provide improved service to selected network traffic over various underlying technologies including Frame Relay, ATM, Ethernet and 802.1 networks, SONET, and IP-routed networks. In particular, QoS features provide improved and more predictable network service by implementing the following services:

- Supporting guaranteed bandwidth
- Improving loss characteristics
- Avoiding and managing network congestion
- Shaping network traffic
- Setting traffic priorities across the network

The following sections describe how to configure Quality of Service on the Cisco MWR 2941:

- Understanding Quality of Service
- Configuring Quality of Service
- Sample Quality of Service Configurations

Understanding Quality of Service

This section describes the Quality of Service (QoS) features on the Cisco MWR 2941. The Cisco MWR 2941 supports the following QoS features.

- Traffic Classification
- Traffic Marking
- Traffic Queuing
- Traffic Shaping

Note The Cisco MWR 2941 support for QoS varies based on the interface and traffic type. For more information about the QoS limitations, see QoS Limitations.

For instructions on how to configure QoS on the Cisco MWR 2941, see Configuring Quality of Service.

Traffic Classification

Classifying network traffic allows you to organize packets into traffic classes based on whether the traffic matches specific criteria. Classifying network traffic is the foundation for enabling many QoS features on your network. For instructions on how to configure traffic classification, see Configuring Classification.

Traffic Marking

Marking network traffic allows you to set or modify the attributes for packets in a defined traffic class. You can use marking with traffic classification to configure variety of QoS features for your network. For instructions on how to configure traffic marking, see Configuring Marking.

Traffic Queuing

The Cisco MWR 2941 supports class-based WFQ (CBWFQ) for congestion management. CBWFQ extends the standard WFQ functionality to provide support for user-defined traffic classes. For CBWFQ, you define traffic classes based on match criteria such as input interface. Packets satisfying the match criteria for a class constitute the traffic for that class. For more instructions on how to configure traffic queuing, see Configuring Congestion Management.

Traffic Shaping

Regulating the packet flow on the network is also known as traffic shaping. Traffic shaping allows you to control the speed of traffic leaving an interface. This way, you can match the flow of the traffic to the speed of the interface receiving the packet.

The Cisco MWR 2941 supports Class-Based Traffic Shaping. Class-Based Traffic Shaping allows you to regulate the flow of packets leaving an interface on a per-traffic-class basis, matching the packet flow to the speed of the interface. For more instructions on how to configure traffic shaping, see Configuring Shaping.

For more information about Quality of Service, see the *Quality of Service Solutions Configuration Guide, Cisco IOS Release 15.0S*.

Configuring Quality of Service

The following sections describe how to configure the Quality of Service (QoS) features supported by the Cisco MWR 2941 router.

- QoS Limitations, page 24-3
- Sample QoS Configuration, page 24-8
- Configuring Classification, page 24-10
- Configuring Marking, page 24-12
- Configuring Congestion Management, page 24-16
- Configuring Shaping, page 24-18

- Configuring Ethernet Trusted Mode, page 24-20

QoS Limitations

The Cisco MWR 2941 offers different QoS support according to the physical interface and traffic type. The following sections describe the limitations for each QoS capability on the Cisco MWR 2941.

- General QoS Limitations
- Statistics Limitations
- Propagation Limitations
- Classification Limitations
- Marking Limitations
- Congestion Management Limitations
- Shaping Limitations

General QoS Limitations

The following general QoS limitations apply to the Cisco MWR 2941.

- You can create a maximum of 32 class maps including the class-default class map.
- You can create a maximum of 32 policy-maps.
- Congestion Avoidance, including Weighted Random Early Detection (WRED) is not supported.
- The following limitations apply to MLPPP interfaces:
  - Input MLPPP interfaces do not support QoS service policies.
  - You can apply only one output QoS service policy to an MLPPP interface.
  - You can create a maximum of 8 **match** statements within a class map in a service policy applied to an MLPPP interface.
  - When applying or modifying any aspect of a service-policy on an MLPPP interface, you must shut down and re-enable the interface.
  - You can create a maximum of 8 classes within a policy-map that is applied to an MLPPP interface. This number includes the default-class.
  - You can have only 1 priority class within a policy-map applied to an MLPPP interface.
- The following limitations apply to GigabitEthernet interfaces:
  - You can apply a maximum of 3 different service policies to GigabitEthernet interfaces
  - You can only use the class-default class for HQoS parent service policies applied to egress GigabitEthernet interfaces.

Statistics Limitations

- Input service policies on the GigabitEthernet interface support statistics based on class map and in terms of packets. Statistics based on filters and statistics in terms of bytes or rates are not supported.
- The **show policy-map** command displays inaccurate output for QoS counters due to ingress counter limitations on the router. The command displays a summary of QoS activity on the MWR 2941 that is limited as follows:

- The number of packets displayed below the Class-map name includes the number of packets matched and marked on the router.
- The Packets marked number for each QoS value always displays as 0.

The following example shows output for the **show policy-map** command:

```
Router# show policy-map interface gigabitethernet0/0 in
 GigabitEthernet0/0

  Service-policy input: INPUT-POLICY

    Class-map: DSCP-IN (match-any)
      2857393 packets
      Match: ip dscp af43 (38) ef (46) cs6 (48) 62
      QoS Set
        cos 5
          Packets marked 0
        qos-group 5
          Packets marked 0
```

- Output MLPPP interfaces support QoS statistics.
- Output service policies on the GigabitEthernet interface do not support statistics.

Propagation Limitations

The Cisco MWR 2941 has the following limitations when propagating QoS values between interfaces:

- The following limitations apply when traffic ingresses through a GigabitEthernet interface and egresses through a GigabitEthernet interface:
  - When traffic is routed at layer 3, the router maps the CoS bits to the QoS group value. The QoS group is not propagated through the L3 network processor.
  - When traffic is switched at layer 2, the QoS group is propagated through the router.
- The following limitations apply when traffic ingresses through any other interface type (host-generated, MLPPP, or HWIC) and egresses through the GigabitEthernet interface.
  - The Precedence bit value is propagated to the CoS bit. The CoS bit value is mapped 1:1 to the QoS group value.

See Sample QoS Configuration, page 24-8 for a sample QoS configuration that accounts for propagation limitations on the Cisco MWR 2941.

Note For more information about QoS restrictions for individual interface cards, see the documentation for Cisco Interface Cards.

Classification Limitations

Table 24-1 summarizes the values that you can use to classify traffic based on interface type. The values are parameters that you can use with the **match** command.

Table 24-1 QoS Classification Limitations by Interface

	GigabitEthernet		**HWIC-9ESW**		**MLPPP**		**HWIC-1GE-SFP**		**HWIC-ADSL**		**HWIC-SHDSL**	
Value	Ingress	Egress	Ingress	Egress	Ingress	Egress	Ingress	Egress	Ingress	Egress	Ingress	Egress
access-group												
all												
any	X					X	X					
any												
class-map												
cos	X						X					
destination-address												
discard-class												
dscp	X					X	X					
flow pdp												
frde												
frdlci												
ip dscp	X						X					
ip precedence												
ip rtp												
mpls experimental						X	X					
not												
packet length												
precedence												
protocol												
qos-group		X										
source-address												
vlan	X											

The following limitations also apply when configuring classification on the Cisco MWR 2941.

- The following limitations apply to input Gigabit Ethernet interface QoS policies:
  - You can use a the **match vlan** command with a maximum of 4 VLANs.
  - You can use the **match dcsp** command with a maximum of 4 DSCP values.

  - You cannot use the same match statement more than once in a single class map. For example, you cannot add two **match vlan** commands to a single class map.
  - You cannot use the **match cos** and **match dscp** commands together in a single class map.
  - Ingress VLAN classification is not supported on switchport interfaces configured as dot1q tunnels using the switchport mode dot1q-tunnel command. We recommend that you configure classification based on CoS, Exp bit, or DSCP.
- The following limitations apply to output Gigabit Ethernet interface QoS policies:
  - Class maps only support matching based on qos-group. This limitation does not apply to the class-default class map.
  - You cannot create two policy maps that match based on the same qos-group value.
- The following limitations apply to input MLPPP interfaces:
  - You can create up to 8 matches in a class-map using DSCP or MPLS Exp values.

Marking Limitations

Table 24-2 summarizes the values that you can use to mark traffic based on interface type. The values are parameters that you can use with the **set** command.

Table 24-2 QoS Marking Limitations by Interface

	GigabitEthernet		**HWIC-9ESW**		**MLPPP**		**HWIC-1GE-SFP**		**HWIC-ADSL**		**HWIC-SHDSL**	
Value	Ingress	Egress	Ingress	Egress	Ingress	Egress	Ingress	Egress	Ingress	Egress	Ingress	Egress
atm-clp												
cos	X		X									
discard-class												
dscp												
dscp-transmit												
ip dscp	X											
ip precedence												
mpls experimental												
mpls experimental imposition												
mpls experimental imposition qos-group												
precedence												
prec-transmit												
qos-group	X											

Congestion Management Limitations

The congestion management limitations for the Cisco MWR 2941 are described in the following sections:

- Queuing Limitations
- Rate Limiting Limitations

Queuing Limitations

The Cisco MWR 2941 uses Class-based fair weighted queuing (CBFQ) for congestion management. Table 24-3 summarizes the queuing commands that you can apply when using CBFQ according to interface type.

Table 24-3 QoS Queuing Limitations by Interface

	GigabitEthernet		HWIC-9ESW		MLPPP		HWIC-1GE-SFP		HWIC-ADSL		HWIC-SHDSL	
Value	Ingress	Egress	Ingress	Egress	Ingress	Egress	Ingress	Egress	Ingress	Egress	Ingress	Egress
bandwidth (kbps)												
bandwidth percent		X				X						
bandwidth remaining percent		X	X			X						
compression header ip												
drop												
fair-queue												
priority		X				X						
priority (kbps)												
priority (without queue-limit)												
priority percent		X				X						
queue-limit (cells)												
queue-limit (packets)		X				X						

Rate Limiting Limitations

You can use rate limiting for congestion management on the Cisco MWR 2941. Table 24-4 summarizes the rate limiting parameters that you can use with the **police** command according to interface type. The table uses the following terms:

- Rate—A speed of network traffic such as a committed information rate (CIR) or peak information rate (PIR).

- Actions—A defined action when traffic exceeds a rate, such as conform-action, exceed-action, or violate-action.

Table 24-4 QoS Rate Limiting Limitations by Interface

	GigabitEthernet		**HWIC-9ESW**		**MLPPP**		**HWIC-1GE-SFP**		**HWIC-ADSL**		**HWIC-SHDSL**	
Policing with	Ingress	Egress	Ingress	Egress	Ingress	Egress	Ingress	Egress	Ingress	Egress	Ingress	Egress
One rate												
One rate and two actions												
Two rates and two actions												
Two rates and three actions												

Shaping Limitations

Table 24-5 summarizes the values that you can use to mark traffic based on interface type. The values are parameters that you can use with the **shape** command.

Table 24-5 QoS Shaping Limitations by Interface

	GigabitEthernet		**HWIC-9ESW**		**MLPPP**		**HWIC-1GE-SFP**		**HWIC-ADSL**		**HWIC-SHDSL**	
Value	Ingress	Egress	Ingress	Egress	Ingress	Egress	Ingress	Egress	Ingress	Egress	Ingress	Egress
adaptive												
average		X						X				
fecn-adapt												
max-buffers												
peak												

The following limitations also apply to QoS shaping on the Cisco MWR 2941:

- The following limitations apply to input Gigabit Ethernet interfaces:
  - You cannot apply shaping to the class-default class unless you are using hierarchical policy maps and applying shaping to the parent policy map.
  - If you are using hierarchical policy maps, you can only apply the class-default class to the parent policy map.

Sample QoS Configuration

The following configuration demonstrates how to apply QoS given the hardware limitations. The Cisco MWR 2941 processes traffic between interfaces as follows:

- For layer 2 traffic passing between the GigabitEthernet 0/2 interface and the GigabitEthernet 0/0 interface, the output queue is determined by the QoS Group assigned in the in-qos policy map.

- For layer 3 traffic passing between GigabitEthernet 0/2 interface and the GigabitEthernet 0/0 interface, the output queue is determined based on the CoS value assigned in the in-qos policy map. (the CoS value is mapped 1:1 to the QoS group value.)
- For traffic passing between other interfaces, the output queue is determined based on the CS fields (top three bits) of the IP DSCP bits; these bits are copied to the CoS bits, which are mapped 1:1 to the QoS group value.

```
!
class-map match-all q0
 match qos-group 0
class-map match-all q1
 match qos-group 1
class-map match-all q2
 match qos-group 2
class-map match-all q3
 match qos-group 3
class-map match-all q4
 match qos-group 4
class-map match-all q5
 match qos-group 5
class-map match-all q6
 match qos-group 6
class-map match-all q7
 match qos-group 7
class-map match-any Voice
 match dscp ef
class-map match-any Signaling
 match dscp af41
class-map match-any HSDPA
 match dscp af11 af12
!
policy-map in-qos
 class Voice
  set cos 5
  set qos-group 5
 class control_plane
  set cos 4
  set qos-group 4
 class HSDPA
  set cos 1
  set qos-group 1
!
policy-map out-child
 class q5
    priority percent 20
 class q4
    bandwidth remaining percent 20
 class q1
    bandwidth remaining percent 59
!
!
policy-map out-parent
 class class-default
    shape average 100000000
  service-policy out-child
!
interface GigabitEthernet 0/2
  switchport access vlan 20
  service-policy input in-qos
!
interface GigabitEthernet 0/0
  switchport trunk allowed vlan 1,10-30,1002-1005
  switchport mode trunk
  service-policy output out-parent
```

Note This is a partial configuration intended to demonstrate the QoS feature.

To view other QoS sample configurations see Sample Quality of Service Configurations.

Configuring Classification

Classifying network traffic allows you to organize packets into traffic classes based on whether the traffic matches specific criteria. Classifying network traffic is the foundation for enabling many QoS features on your network.

Creating a Class Map for Classifying Network Traffic

Class maps allow you to define classes of network traffic to apply QoS features to each class. Follow these steps to create a class map:

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	`Router(config)#` **class-map class1**	Defines a new class map and enter class map configuration mode.
Step 4	`Router(config-cmap)#` **match qos-group 7**	Specifies the match criteria for the class map. You can define a variety of match criteria including CoS, DSCP, MPLS Exp, or QoS group value.

Step 5	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Creating a Policy Map for Applying a QoS Feature to Network Traffic

A policy map allows you to apply a QoS feature to network traffic based on the traffic classification. Follow these steps to create and configure a policy map that uses an existing class map:

	Command	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)# policy-map policy1`
	`Router(config-pmap)#`
		Defines a new policy map and enter policy map configuration mode.

Step 4	`Router(config-pmap)# class class1`
	`Router(config-pmap-c)#`
		Specifies a traffic class to which the policy applies. This command enters policy-map class configuration mode, which allows you to define the treatment for the traffic class.

Step 5	`Router(config-pmap-c)# bandwidth percent 50`	(Optional) Specifies the bandwidth allocated for a traffic class attached to the policy map. You can define the amount of bandwidth in kbps, a percentage of bandwidth, or an absolute amount of bandwidth. **Note** GigabitEthernet interfaces support only bandwidth defined as a percentage or remaining percent.

Step 6	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Note You can use the **show policy-map** command to verify your configuration.

Attaching the Policy Map to an Interface

After you create the policy map, you must attach it to an interface. Policy maps can be attached to either the input or output direction of the interface. Follow these steps to attach a policy map to an interface:

	Command	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)# interface gigabitEthernet0/1`	Specifies the interface to which you want to apply the policy map.

	Command	Purpose
Step 4	`Router(config-if)# service-policy output policy1`	Attaches the policy map to an interface. The **input** and **output** parameters specify the direction in which router applies the policy map.

Step 5	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Note You can use the **show policy map** interface command to verify your configuration.

For more information about configuring classification, see the *Quality of Service Solutions Configuration Guide, Cisco IOS Release 15.0S*.

Configuring Marking

Marking network traffic allows you to set or modify the attributes for packets in a defined traffic class. You can use marking with traffic classification to configure variety of QoS features for your network.

The Cisco MWR 2941 marking allows you to modify the following packet attributes:

- Differentiated services code point (DSCP) value
- Class of service (CoS) value
- MPLS Exp bit value
- Qos-group value (internal)

For instructions on how to configure marking for IP Precedence, DSCP, or CoS value, use the following sections:

- Creating a Class Map for Marking Network Traffic, page 24-13
- Creating a Policy Map for Applying a QoS Feature to Network Traffic, page 24-13
- Attaching the Policy Map to an Interface, page 24-14

For instructions on how to configure MPLS Exp bit marking, see Configuring MPLS Exp Bit Marking Using a Pseudowire.

Creating a Class Map for Marking Network Traffic

Class maps allow you to define classes of network traffic to apply QoS features to each class. Follow these steps to define a traffic class to mark network traffic.

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	`Router(config)# class-map class1`	Defines a new class map and enter class map configuration mode.
Step 4	`Router(config-cmap)# match qos-group 7`	Specifies the match criteria for the class map. You can define a variety of match criteria including CoS, DSCP, MPLS Exp, or QoS group value.

Step 5	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Creating a Policy Map for Applying a QoS Feature to Network Traffic

Policy maps allow you to apply the appropriate QoS feature to the network traffic based on the traffic classification. The follow sections describe how to create and configure a policy map to use a class map or table map.

The following restrictions apply when applying a QoS feature to network traffic:

- A policy map containing the **set qos-group** command can be attached only as an output traffic policy.
- A policy map containing the **set cos** command can be attached only as an input traffic policy.

Follow these steps to create a policy map.

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

Step 3	`Router(config)# policy-map policy1`
	`Router(config-pmap)#`
		Defines a policy map and enter policy map configuration mode.

	Command	Purpose

Step 4	`Router(config-pmap)# class class1`
	`Router(config-pmap-c)#`
		Specifies the traffic class for which you want to create a policy and enter policy map class configuration mode. You can also use the **class-default** parameter to define a default class.

Step 5	**`set cos`**
	`set dscp`
	`set qos-group`
		Defines a QoS treatment type; use one of the **set** commands listed in Table 6.

Step 6	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Table 6 set Commands Summary

set Commands	Traffic Attributes	Network Layer	Protocol
set cos	Layer 2 CoS value of the outgoing traffic	Layer 2	ATM
set dscp	DSCP value in the ToS byte	Layer 3	IP
set qos-group	QoS group ID	Layer 3	IP, MPLS

Note You can use the **show policy-map** or **show policy-map** *policy-map* **class** *class-name* commands to verify your configuration.

Attaching the Policy Map to an Interface

Follow these steps to attach a policy map to an interface.

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router#` **`configure terminal`**
		Enters global configuration mode.

Step 3	`Router(config)#` **`interface gigabitEthernet0/1`**	Specifies the interface to which to apply the policy map.

	Command	Purpose
Step 4	`Router(config-if)# service-policy input policy1`	Attaches the policy map to an interface. The **input** and **output** parameters specify the direction in which router applies the policy map.

Step 5	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Note You can use the **show policy map** interface command to verify your configuration.

Configuring MPLS Exp Bit Marking Using a Pseudowire

You can also configure MPLS Exp bit marking within an ATM over MPLS pseudowire interface using the **mpls experimental** command. Follow these steps to configure MPLS Exp bit marking using a pseudowire interface.

	Command	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)# pseudowire-class MPLS_3`	Creates a new pseudowire class.
Step 4	`Router(config-pw-class)# encapsulation mpls`	Configures MPLS encapsulation.
Step 5	`Router(config-pw-class)# mpls experimental 3`	Specifies the MPLS Exp bit value.

Step 6	`Router(config-pw-class)# exit`
	`Router(config)#`
		Exits the pseudowire-class interface.

Step 7	`Router(config)# interface ATM0/IMA0`
	`Router(config-if)#`
		Configures the ATM/IMA interface.

Step 8	`Router(config-if)# pvc 2/1 l2transport`
	`Router(cfg-if-atm-l2trans-pvc)#`
		Specifies a PVC.

Step 9	`Router(cfg-if-atm-l2trans-pvc)# encapsulation aal0`	Specifies an encapsulation type for the PVC.

	Command	Purpose
Step 10	`Router(cfg-if-atm-l2trans-pvc)#` **`xconnect 10.10.10.1 121 pw-class MPLS_3`**	Creates a pseudowire. Use the **pw-class** keyword to use the configuration defined in the pseudowire class.

Step 11	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

For more information about configuring marking, see the *Quality of Service Solutions Configuration Guide, Cisco IOS Release 15.0S*.

Note The Cisco MWR 2941 does not support all of the commands described in the IOS Release 15.0S documentation.

Configuring Congestion Management

The following sections describe how to configure congestion management on the Cisco MWR 2941.

- Configuring Low Latency Queueing (LLQ), page 24-16
- Configuring Class-Based Weighted Fair Queuing (CBFQ), page 24-17

Configuring Low Latency Queueing (LLQ)

Low latency queuing allows you to define a percentage of bandwidth to allocate to an interface or PVC as a percentage. You can define a percentage for priority or nonpriority traffic classes. Follow these steps to configure LLQ.

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)#` **`policy-map policy1`**	Use the **policy-map** command to define a policy map.

Step 4	`Router(config-pmap)#` **`class class1`**
	`Router(config-pmap-c)#`
		Use the **class** command to reference the class map that defines the traffic to which the policy map applies.

	Command	Purpose
Step 5	`Router(config-pmap-c)#` **`priority percent 10`**	Use the **priority** command to specify the priority percentage allocated to the traffic class assigned to the policy map. You can use the **burst** parameter to configures the network to accommodate temporary bursts of traffic.
Step 6	`Router(config-pmap-c)#` **`bandwidth percent 30`**	Use the **bandwidth** command to specify the bandwidth available to the traffic class within the policy map. You can specify the bandwidth in kbps or by a percentage of bandwidth.

Step 7	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exit configuration mode.

Note You can use the **show policy-map**, **show policy-map policy-map class** ***class-name, or*** **show policy-map interface** commands to verify your configuration.

Configuring Class-Based Weighted Fair Queuing (CBFQ)

The Cisco MWR 2941 supports class-based weighted fair queuing (CBWFQ) for congestion management. Follow these steps to configure CBWFQ.

Note The Cisco MWR 2941 does not support the **queue-limit** and **random-detect** commands.

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		- Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router#` **`configure terminal`**
		Enters global configuration mode.

Step 3	`Router(config)#` **`class-map class1`**
	`Router(config-cmap)#`
		Creates a class map.
		A class map contains match criteria against which a packet is checked to determine if it belongs to the class. You can use class maps to define criteria that are referenced in one or more policy maps.

Step 4	`Router(config-cmap)#` **`match qos-group 7`**	Specifies the match criteria for the class map. You can define a variety of match criteria including CoS, DSCP, MPLS Exp, or QoS group value.

Step 5	`Router(config-cmap)#` **`exit`**
	`Router(config)#`
		Exits class map configuration.

Step 6	`Router(config)#` **`policy-map policy1`**
	`Router(config-pmap)#`
		Defines a policy map.

	Command	Purpose

Step 7	`Router(config-pmap)#` **`class class1`**
	`Router(config-pmap-c)#`
		References the class map that defines the traffic to which the policy map is applied.

Step 8	`Router(config-pmap-c)#` **`bandwidth 3000`**	Specifies the bandwidth allocated for the traffic class.

Step 9	`Router(config-pmap)#` **`exit`**
	`Router(config)#`
		Exits the policy map configuration.

Step 10	`Router(config)#` **`interface atm0/ima0`**	Enters configuration for the interface to which you want to apply the policy map.
Step 11	`Router(config-if)#` **`service-policy output policy1`**	Applies the service policy to the interface.

Step 12	**exit**
	Example:
	`Router(config)#` **`exit`**
	`Router#`
		Exits configuration mode.

Configuring Shaping

The Cisco MWR 2941 supports class-based traffic shaping.

Class-based traffic shaping is configured using a hierarchical policy map structure; you enable traffic shaping on a primary level (parent) policy map and other QoS features such as queuing and policing on a secondary level (child) policy map.

The following sections describe how to configure shaping:

- Configuring Class-Based Traffic Shaping in a Primary-Level (Parent) Policy Map
- Configuring the Secondary-Level (Child) Policy Map

Configuring Class-Based Traffic Shaping in a Primary-Level (Parent) Policy Map

Follow these steps to configure a parent policy map for traffic shaping.

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)#` **`policy-map`** *`output-policy`*	Specifies the policy map for which you want to configure shaping and enter policy-map configuration mode.

Step 4	`Router(config-pmap)#` **`class class1`**
	`Router(config-pmap-c)#`
		Specifies the traffic class to which the policy map applies.

	Command	Purpose
Step 5	Router(config-pmap-c)# **shape** [**average** | **peak**] *mean-rate* [[*burst-size*] [*excess-burst-size*]]	Defines the algorithm and rate used for traffic shaping.
Step 6	Router(config-pmap-c)# service-policy *policy-map*	Attaches the policy map to the class map.
Step 7	**exit** **Example:** Router(config)# exit Router#	Exits configuration mode.

Note You can use the **show policy-map** command to verify your configuration.

For more information about configuring shaping, see the *Quality of Service Solutions Configuration Guide, Cisco IOS Release 15.0S.*

Note The Cisco MWR 2941 does not support all of the commands described in the IOS Release 15.0S documentation.

Configuring the Secondary-Level (Child) Policy Map

Follow these steps to create a child policy map for traffic shaping:

	Command	Purpose
Step 1	**enable** **Example:** Router> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Router# **configure terminal**	Enters global configuration mode.
Step 3	Router(config)# **policy-map** *output-policy*	Specifies the policy map for which you want to configure shaping and enter policy-map configuration mode.
Step 4	Router(config-pmap)# **class class1** Router(config-pmap-c)#	Specifies the traffic class to which the policy map applies.

	Command	Purpose
Step 5	`Router(config-pmap-c)# bandwidth percent 50`	Specifies the bandwidth allocated to the policy map. You can specify the bandwidth in kbps, a relative percentage of bandwidth, or an absolute amount of bandwidth.

Step 6	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

For more information about configuring shaping, see the *Quality of Service Solutions Configuration Guide, Cisco IOS Release 15.0S.*

Note The Cisco MWR 2941 does not support all of the commands described in the IOS Release 15.0S documentation.

Configuring Ethernet Trusted Mode

The Cisco MWR 2941 supports trusted and non-trusted mode for switch ports. Switch ports are set in non-trusted mode by default; if you want to set the Ethernet switch ports in trusted mode, use the global command **switch l2trust** to set all Ethernet ports to trusted mode.

```
Router(config)# switch l2trust
```

For more information about the **switch l2trust** command, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.

Configuring Switchport Priority

Follow these steps to configure priority bit values on incoming traffic on 9ESW HWIC interfaces.

	Command	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		- Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)# interface FastEthernet 1/7`	Enter interface configuration.

	Command	Purpose
Step 4	Router(config-if)# **switchport priority default** *priority* **Example:** Router(config-if)# **switchport priority default 7**	Configures a default priority value to apply to incoming traffic on the interface.
Step 5	Router(config-if)# **switchport priority override**	(Optional) Configures the interface to override the priority value set on inbound traffic.
Step 6	**exit** **Example:** Router(config)# **exit** Router#	Exits configuration mode.

Sample Quality of Service Configurations

The following sample configurations demonstrate how you can apply QoS configurations on the Cisco MWR 2941.

Note This section provides partial configurations intended to demonstrate a specific feature.

The following sections provide sample configurations for QoS on the Cisco MWR 2941.

- Switchport Priority
- Classification and Marking
- Priority Queuing

For more information about configuring QoS, see "Configuring Quality of Service" section on page 24-1.

Switchport Priority

The following sample configuration demonstrates how to mark P-bit values on incoming traffic on the 9ESW HWIC interface.

```
...............
interface GigabitEthernet0/2
no ip address
  switchport stacking-partner interface FastEthernet1/8
...............
interface FastEthernet1/7
switchport mode trunk
switchport priority default 7 ! sets all ingress traffic to priority 7
switchport priority override

interface FastEthernet1/7
switchport mode access
switchport access vlan 100
switchport priority default 5   ! set all ingress traffic to priority 5
```

```
interface FastEthernet1/8
no IP address
switchport stacking-partner interface GigabitEthernet0/2
```

Classification and Marking

The following configuration example marks the DSCP value of ingress Ethernet traffic and assigns it to a QoS group, and marks P-bits. Egress traffic is queued using WRR with bandwidth percentages allocated to each group.

```
! Note 1: these class-maps are applied on ingress
class-map match-any common-channels
 match  dscp af31  af32  af33
class-map match-any HSDPA
 match  dscp default
class-map match-any R99
 match  dscp af21  af22  af23
class-map match-any synchronization
 match  dscp ef  cs6
class-map match-any signaling
 match  dscp af41  af42  af43
!
! Note 2: these classp-maps are applied on egress
class-map match-any group1
 match qos-group 1
class-map match-any group2
 match qos-group 2
class-map match-any group3
 match qos-group 3
class-map match-any group4
 match qos-group 4
class-map match-any group5
 match qos-group 5
class-map match-any group6
 match qos-group 6

! Note 3:The input policy performs the DSCP match and all marking
policy-map input-policy
 class synchronization
  set qos-group 6
  set cos 6
 class signaling
  set qos-group 5
  set cos 5
 class common-channels
  set qos-group 4
  set cos 4
 class R99
  set qos-group 3
  set cos 3
 class HSDPA
  set qos-group 1
 class default
  set qos-group 1
!
! Note 4: the hierarchical output policy handles WRR and shaping
policy-map QOS-child
 class group6
  priority  percent 5
 class group5
  bandwidth percent 20
 class group4
  bandwidth percent 20
 class group3
  bandwidth percent 20
 class group1
  bandwidth percent 20
policy-map output-policy
 class class-default
  shape average 38000000
  service-policy QOS-child
!
Interface GigabitEthernet 0/0
 service-policy input  input-policy
Interface GigabitEthernet 0/1
 service-policy output output-policy
```

MPLS Bit Marking

The following configuration example marks MPLS Exp bits on traffic passing through pseudowire class UMTS_3. You can map the Exp bit value to a QoS group on an MLPPP egress interface or an MLPPP or layer 2 Ethernet queue.

```
!
pseudowire-class UMTS_3
encapsulation mpls
mpls experimental 3
!
interface ATM0/IMA0
 pvc 2/1 l2transport
 encapsulation aal0
 xconnect 10.10.10.1 121 pw-class UMTS_3
!
!
```

Priority Queuing

The following sample configuration places any traffic with a DSCP value of **ef** into the priority queue of the MLPPP multilink interface.

```
class-map match-any gsm-abis
 match  dscp ef
!
!
policy-map gsm-abis  ? note that without multiclass up to 4 queues supported
 class gsm-abis
  priority percent 99
 class class-default
  bandwidth remaining percent 1
!

 interface Multilink1
 ip address 50.50.50.49 255.255.255.0
 ip tcp header-compression ietf-format
 load-interval 30
 keepalive 1
 ppp pfc local request
 ppp pfc remote apply
 ppp acfc local request
 ppp acfc remote apply
 ppp multilink
 ppp multilink interleave
 ppp multilink group 1
 ppp multilink fragment delay 0 1
 ppp multilink multiclass
 ppp timeout multilink lost-fragment 1
 max-reserved-bandwidth 100
 service-policy output gsm-abis
 hold-queue 50 out
 ip rtp header-compression ietf-format
```

CHAPTER 25

Configuring Link Noise Monitor

Noise on T1 and E1 links that span between the BTS and central office can affect voice quality for mobile users to the point where it becomes unacceptable. To monitor the quality of individual links in a multilink bundle, you can configure the Link Noise Monitor (LNM) on your Cisco MWR 2941 router.

The LNM detects, alerts, and removes noisy links from a bundle based on user-defined thresholds and durations. In addition, the LNM notifies the operator once the quality of the line has improved, and restores the link service if the link has been removed.

To detect noise on a link, the LNM monitors the following two types of errors which make up the Bit Error Rate (BER) and compares the number of errors with the user-defined thresholds:

- Line Code Violation (LCV)—A Bi-Polar Violation (BPV) or Excessive Zeroes (EXZ) error has occurred.
- Path Code Violation (PCV)—A Cyclic Redundancy Check (CRC) error, which is generally caused by one or more LCV or logic errors, has occurred in a time slot.

The LNM provides the following types of noise monitors:

- Link Warning—Issues a warning when the noise level of a link exceeds a user-defined threshold and notifies the operator when the noise level improves to the point that it drops below a second user-defined threshold.
- Link Removal—Issues an error and removes a link from service when the noise level of the link exceeds a user-defined threshold and restores the link and provides notification when the noise level improves to the point that it drops below a second user-defined threshold.

Note If the noise level on the last active link in a multilink bundle exceeds the Link Removal threshold, an alert is issued but the link will not be removed from service. If this situation occurs, the standard T1 error rate is used to determine if the last active link must be removed from service.

To configure the LNM feature, issue the **span** command from controller configuration mode of each T1 or E1 link in the bundle that you want to monitor. To disable LNM on a link, issue the **no** version of the command from controller configuration mode of the link.

span { **warn** | **remove }** [{ [**lcv** *value* [**pcv** *value*]] [**duration** *seconds*] } **set** | **clear**]

where:

- **warn**—Enables Link Warning monitoring on the link.
- **remove**—Enables Link Removal monitoring on the link.

- **lcv** *value*—Threshold (in bit errors per second) that when exceeded for the configured duration when the **set** keyword has been specified, creates a condition (warning or link removal), or when fallen below for the configured duration when the **clear** keyword has been specified, clears the condition.

  For T1 links:

  – Valid range is 5 to 1544.

  – For Link Warning monitoring, the default is 15.

  – For Link Removal monitoring, the default is 154.

  For E1 links,

  – Valid range is 7 to 2048.

  – For Link Warning monitoring, the default is 20.

  – For Link Removal monitoring, the default is 205.

- **pcv** *value*—Number of time slots in errors per second. If not specified by the user, this value is calculated from the LCV threshold based on a Gaussian distribution that matches typical noise-induced errors.

  For T1 links:

  – Valid range is 3 to 320.

  – For Link Warning monitoring, the default is 15.

  – For Link Removal monitoring, the default is 145.

  For E1 links,

  – Valid range is 8 to 832.

  – For Link Warning monitoring, the default is 20.

  – For Link Removal monitoring, the default is 205.

- **duration** *seconds*—Number of seconds that a threshold must be exceeded to create a condition or fallen below to clear a condition. Valid range is 1 to 600. The default is 10.

  When specified with th**e lcv** keyword, the duration must be configured after the LCV threshold. For example, **span warn lcv 55 duration 20** is a correct way to issue the command; s**pan warn duration 20 lcv 55** is not.

- **set**—Specifies that the values configured for the **span** command are to be used to set a condition.
- **clear**—Specifies that the values configured for the **span** command are to be used to clear a condition.

Usage Notes

When configuring the LNM, please note the following:

- If the **warn** and **remove** keywords are specified without any other options, the LCV and PCV thresholds and duration defaults will be used to determine (**set**) and clear (**clear**) the condition.
- If the **span** command is issued with the **set** keyword specified (defining the LNM type and parameters to use to determine a condition exists) and the command is not issued again with the **clear** keyword specified (defining the parameters used to clear a condition), or vice versa, the values configured for the threshold and duration will be used for both.

- If the **span** command is issued without either the **set** or **clear** keywords specified, **set** is the default.
- The **set** and **clear** keywords can only be specified if the threshold and/or duration has been specified.
- If the PCV threshold is not configured (using the **pcv** keyword and value), the threshold is calculated using Gaussian probability distribution that is representative of most noise environments.
- The following SYSLOG messages have been added for fault notification:

```
- %LNM-4- WARNEXCEED:Controller <Controller IF>, exceeded noise warning threshold
  <int>, duration <int>
- %LNM-4- WARNIMPROVE:Controller <Controller IF>, noise improved below threshold
  <int>, duration <int>
- %LNM-2-REMOVE:Interface <Serial IF> removed, noise exceeded threshold <int>,
  duration <int>
- %LNM-2- RESTORE:Interface <Serial IF> restored, noise improved below threshold
  <int>, duration <int>
- %LNM-2- REMEXCEED:Interface <Serial IF>, noise exceeded threshold <int>,
  duration <int>
- %LNM-2- REMIMPROVE:Interface <Serial IF>, noise improved below threshold <int>,
  duration <int>
```

CHAPTER 26

Configuring Cisco Discovery Protocol

This chapter describes how to configure Cisco Discovery Protocol (CDP) on the Cisco MWR 2941 router.

Note For complete syntax and usage information for the commands used in this chapter, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR* and the *Configuration Fundamentals Configuration Guide, Cisco IOS Release 15.0S*.

Note The Cisco MWR 2941 does not necessarily support all of the commands described in the Release 15.0(1)S documentation.

Understanding CDP

CDP is a device discovery protocol that runs over Layer 2 (the data link layer) on all Cisco-manufactured devices (routers, bridges, access servers, and switches) and allows network management applications to discover Cisco devices that are neighbors of already known devices. With CDP, network management applications can learn the device type and the Simple Network Management Protocol (SNMP) agent address of neighboring devices running lower-layer, transparent protocols. This feature enables applications to send SNMP queries to neighboring devices.

CDP runs on all media that support Subnetwork Access Protocol (SNAP). Because CDP runs over the data-link layer only, two systems that support different network-layer protocols can learn about each other.

Each CDP-configured device sends periodic messages to a multicast address, advertising at least one address at which it can receive SNMP messages. The advertisements also contain time-to-live, or holdtime information, which is the length of time a receiving device holds CDP information before discarding it. Each device also listens to the messages sent by other devices to learn about neighboring devices.

For a router and connected endpoint devices running Cisco Medianet

- CDP identifies connected endpoints that communicate directly with the router.
- To prevent duplicate reports of neighboring devices, only one wired switch reports the location information.
- The wired switch and the endpoints both send and receive location information.

  For information, go to http://www.cisco.com/en/US/docs/ios/netmgmt/configuration/guide/nm_cdp_discover.html.

The router supports CDP Version 2.

Configuring CDP

- Default CDP Configuration, page 26-2
- Configuring the CDP Characteristics, page 26-2
- Disabling and Enabling CDP, page 26-3
- Disabling and Enabling CDP on an Interface, page 26-4

Default CDP Configuration

Table 26-1 shows the default CDP configuration.

Table 26-1 Default CDP Configuration

Feature	Default Setting
CDP global state	Enabled.
CDP interface state	Enabled only on NNIs; disabled on ENIs **Note** CDP is not supported on UNIs.
CDP timer (packet update frequency)	60 seconds
CDP holdtime (before discarding)	180 seconds
CDP Version-2 advertisements	Enabled

Configuring the CDP Characteristics

You can configure the frequency of CDP updates, the amount of time to hold the information before discarding it, and whether or not to send Version-2 advertisements.

Beginning in privileged EXEC mode, follow these steps to configure the CDP timer, holdtime, and advertisement type.

Note Steps 2 through 4 are all optional and can be performed in any order.

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**cdp timer** *seconds*	(Optional) Sets the transmission frequency of CDP updates in seconds. The range is from 5–254; the default is 60 seconds.
Step 3	**cdp holdtime** *seconds*	(Optional) Specifies the amount of time a receiving device should hold the information sent by your device before discarding it. The range is from 10–255 seconds; the default is 180 seconds.
Step 4	**cdp advertise-v2**	(Optional) Configures CDP to send Version-2 advertisements. This is the default state.
Step 5	**end**	Returns to privileged EXEC mode.
Step 6	**show cdp**	Verifies your settings.
Step 7	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Use the **no** form of the CDP commands to return to the default settings.

This example shows how to configure CDP characteristics.

```
Router# configure terminal
Router(config)# cdp timer 50
Router(config)# cdp holdtime 120
Router(config)# cdp advertise-v2
Router(config)# end
```

For additional CDP **show** commands, see the "Monitoring and Maintaining CDP" section on page 26-5.

Disabling and Enabling CDP

CDP is enabled by default on NNIs. It is disabled by default on ENIs but can be enabled.

Note Cisco devices (such as Cisco IP Phones) regularly exchange CDP messages with connected devices. Disabling CDP can interrupt device connectivity.

Beginning in privileged EXEC mode, follow these steps to globally disable the CDP device discovery capability:

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**no cdp run**	Disables CDP.
Step 3	**end**	Returns to privileged EXEC mode.

Beginning in privileged EXEC mode, follow these steps to globally enable CDP when it has been disabled:

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**cdp run**	Enables CDP after disabling it.
Step 3	**end**	Returns to privileged EXEC mode.

This example shows how to globally enable CDP if it has been disabled:

```
Router# configure terminal
Router(config)# cdp run
Router(config)# end
```

Disabling and Enabling CDP on an Interface

CDP is enabled by default on NNIs to send and to receive CDP information. You can enable CDP on ENIs, but it is not supported on UNIs. Beginning in privileged EXEC mode, follow these steps to disable CDP on a port:

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id*	Specifies the interface on which you are disabling CDP, and enter interface configuration mode.
Step 3	**no cdp enable**	Disables CDP on the interface.
Step 4	**end**	Returns to privileged EXEC mode.
Step 5	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Beginning in privileged EXEC mode, follow these steps to enable CDP on a port when it has been disabled:

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id*	Specifies the interface on which you are enabling CDP, and enter interface configuration mode.
Step 3	**cdp enable**	Enables CDP on the interface after disabling it.
Step 4	**end**	Returns to privileged EXEC mode.
Step 5	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

This example shows how to enable CDP on a port when it has been disabled:

```
Router# configure terminal
Router(config)# interface gigabitethernet0/1
Router(config-if)# cdp enable
Router(config-if)# end
```

This example shows how to change a UNI to an ENI and enable CDP on the port.

```
Router# configure terminal
Router(config)# interface fastethernet0/1
Router(config-if)# cdp enable
Router(config-if)# end
```

Monitoring and Maintaining CDP

To monitor and maintain CDP on your device, perform one or more of these tasks, beginning in privileged EXEC mode:

Command	Description
clear cdp counters	Resets the traffic counters to zero.
clear cdp table	Deletes the CDP table of information about neighbors.
show cdp	Displays global information, such as frequency of transmissions and the holdtime for packets being sent.
show cdp entry *entry-name* [**protocol** | **version**]	Displays information about a specific neighbor. You can enter an asterisk (*) to display all CDP neighbors, or you can enter the name of the neighbor about which you want information. You can also limit the display to information about the protocols enabled on the specified neighbor or information about the version of software running on the device.
show cdp interface [*interface-id*]	Displays information about interfaces where CDP is enabled. You can limit the display to the interface about which you want information.
show cdp neighbors [*interface-id*] [**detail**]	Displays information about neighbors, including device type, interface type and number, holdtime settings, capabilities, platform, and port ID. You can limit the display to neighbors of a specific interface or expand the display to provide more detailed information.
show cdp traffic	Displays CDP counters, including the number of packets sent and received and checksum errors.

CHAPTER 27

Monitoring and Managing the Cisco MWR 2941 Router

The Cisco MWR 2941 supports a variety of network management features, including Mobile Wireless Transport Manager (MTWM), Cisco Active Network Abstraction (ANA), SNMP, and Cisco Networking Services (CNS). The following sections describe the network management features on the Cisco MWR 2941.

- Understanding Network Management Features for the Cisco MWR 2941, page 27-1
- Configuring Network Management Features, page 27-2

Understanding Network Management Features for the Cisco MWR 2941

The following sections describe the network management features available on the Cisco MWR 2941.

- Cisco Mobile Wireless Transport Manager (MWTM), page 27-1
- Cisco Active Network Abstraction (ANA), page 27-1
- SNMP MIB Support, page 27-2
- Cisco Networking Services (CNS), page 27-2

Cisco Mobile Wireless Transport Manager (MWTM)

You can use Cisco Mobile Wireless Transport Manager (MWTM), to monitor and manage the Cisco MWR 2941. Cisco MWTM addresses the element-management requirements of mobile operators and provides fault, configuration, and troubleshooting capability. For more information about MWTM, see http://www.cisco.com/en/US/products/ps6472/tsd_products_support_series_home.html.

Cisco Active Network Abstraction (ANA)

You can also use Cisco Active Network Abstraction (ANA) to manage the Cisco MWR 2941. Cisco ANA is a powerful, next-generation network resource management solution designed with a fully distributed OSS mediation platform which abstracts the network, its topology and its capabilities from the physical elements. Its virtual nature provides customers with a strong and reliable platform for service activation, service assurance and network management. For more information about ANA, see http://www.cisco.com/en/US/products/ps6776/tsd_products_support_series_home.html.

SNMP MIB Support

To view the current MIBs that the Cisco MWR 2941 supports, see the *Release Notes for Cisco MWR 2941-DC Mobile Wireless Edge Router for Cisco IOS Release 12.4(20)MR*.

For instructions on how to configure MIBs on the Cisco MWR 2941, see Configuring SNMP Support and Enabling Remote Network Management.

Cisco Networking Services (CNS)

Cisco Networking Services (CNS) is a collection of services that can provide remote configuration of Cisco IOS networking devices and remote execution of some command-line interface (CLI) commands. CNS allows a Cisco MWR 2941 deployed and powered on in the field to automatically download its configuration.

Note The Cisco MWR 2941 only supports CNS over motherboard Ethernet interfaces. Other interface types do not support CNS.

For instructions on how to configure CNS, see Configuring Cisco Networking Services (CNS).

Configuring Network Management Features

The following sections describe how to configure network management features on the Cisco MWR 2941.

- Using Cisco Mobile Wireless Transport Manager (MWTM), page 27-2
- Configuring SNMP Support, page 27-3
- Enabling Remote Network Management, page 27-8
- Show Commands for Monitoring the Cisco MWR 2941 Router, page 27-9
- Configuring Cisco Networking Services (CNS), page 27-11

Using Cisco Mobile Wireless Transport Manager (MWTM)

You can use Cisco network management applications, such as Cisco Mobile Wireless Transport Manager (MWTM), to monitor and manage the Cisco MWR 2941. This Network Management tool provides monitoring and management capabilities to the RAN-O solution. The Cisco MWTM addresses the element-management requirements of mobile operators and provides fault, configuration, and troubleshooting capability. The Cisco MWTM provides the following key features:

- Event Monitoring
- Web-Based Reporting
- Autodiscovery and Topology

- Inventory
- OSS Integration
- Security
- Client/Server Architecture
- Multiple OS Support

The Cisco MWTM integrates with any SNMP-based monitoring system, such as Cisco Info Center products. In addition, the Cisco MWTM collects a large amount of performance data that can be exported or directly accessed from the database. This data can then be used by performance reporting applications. For more information about MWTM, see http://www.cisco.com/en/US/products/ps6472/tsd_products_support_series_home.html.

Configuring SNMP Support

Use the following instructions to configure SNMP support: setting up the community access, establishing a message queue for each trap host, enabling the router to send SNMP traps, enabling SNMP traps for alarms, and enabling SNMP traps for a specific environment. You might also need to enter other configuration commands, depending on the requirements for your system configuration and the protocols you plan to route on the interface.

Note To view the current MIBs that the Cisco MWR 2941 supports, see the *Release Notes for Cisco MWR 2941 Mobile Wireless Edge Router for Cisco IOS Release 12.2(33)MRA*.

Note In the following procedure, press the **Return** key after each step unless otherwise noted. At any time, you can exit the privileged level and return to the user level by entering **disable** at the `Router#` prompt.

To configure a Cisco MWR 2941 for SNMP, follow these steps while in the global configuration mode:

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

	Command	Purpose

Step 3	`Router(config)# snmp-server community string [view view-name] [ro | rw] [number]`
	Example:
	`Router(config)# snmp-server community xxxxx RO`
		Sets up the community access string to permit access to SNMP. The **no** form of this command removes the specified community string.
		The syntax is as follows:
		• *string*—Community string that acts like a password and permits access to the SNMP protocol.
		• **view** *view-name*—(Optional) Previously defined view. The view defines the objects available to the community.
		• **ro**—(Optional) Specifies read-only access. Authorized management stations are able only to retrieve MIB objects.
		• **rw**—(Optional) Specifies read-write access. Authorized management stations are able to both retrieve and modify MIB objects.
		• *number*—(Optional) Integer from 1 to 99 that specifies an access list of IP addresses that are allowed to use the community string to gain access to the SNMP agent.
		The example shows how to configure the community access string as xxxxx with read-only access.

Step 4	`Router(config)# snmp-server queue-length length`
	Example:
	`Router(config)# snmp-server queue-length 100`
		Establishes the message queue length for each trap host, use the **snmp-server queue-length** command. The syntax is as follows:
		• *length*—Integer that specifies the number of trap events that can be held before the queue must be emptied.
		The examples shows how to configure the number of trap events as 100.

	Command	Purpose
Step 5	Router(config)# **snmp-server enable traps** [***notification-type***] [***notification-option***] **Example:** Router(config)# **snmp-server enable traps snmp linkdown linkup coldstart warmstart**	Enables the router to send SNMP traps or notifications. Use the **no** form of this command to disable SNMP notifications. The syntax is as follows: • ***notification-type***—**snmp [authentication]**—Enables RFC 1157 SNMP notifications. Note that use of the **authentication** keyword produces the same effect as not using the **authentication** keyword. Both the **snmp-server enable traps snmp** and **snmp-server enable traps snmp authentication** forms of this command globally enable (or, if using the **no** form, disable) the following SNMP traps: – authentication failure – linkup – linkdown – coldstart – warmstart • ***notification-option***—(Optional) **atm pvc** [**interval** *seconds*] [**fail-interval** *seconds*]—The optional interval seconds keyword/argument combination specifies the minimum period between successive traps, in the range from 1 to 3600. Generation of PVC traps is dampened by the notification interval to prevent trap storms. No traps are sent until the interval lapses. The default interval is 30. The optional fail-interval seconds keyword/argument combination specifies the minimum period for storing the failed time stamp, in the range from 0 to 3600. The default fail-interval is 0. • **envmon** [**voltage** | **shutdown** | **supply** | **fan** | **temperature**]—When the **envmon** keyword is used, you can enable a specific environmental notification type, or accept all notification types from the environmental monitor system. If no option is specified, all environmental notifications are enabled. The option can be one or more of the following keywords: **voltage**, **shutdown**, **supply**, **fan**, and **temperature**. • **isdn** [**call-information** | **isdn u-interface**]—When the **isdn** keyword is used, you can specify the **call-information** keyword to enable an SNMP ISDN call information notification for the ISDN MIB subsystem, or you can specify the **isdnu-interface** keyword to enable an SNMP ISDN U interface notification for the ISDN U interface MIB subsystem. • **repeater** [**health** | **reset**]—When the **repeater** keyword is used, you can specify a repeater option. If no option is specified, all repeater notifications are enabled. The option can be one or more of the following keywords: – **health**—Enables IETF Repeater Hub MIB (RFC 1516) health notification. – **reset**—Enables IETF Repeater Hub MIB (RFC 1516) reset notification.

	Command	Purpose
Step 6	Router(config)# **snmp-server enable traps ipran**	Enables SNMP traps for all IP-RAN notifications. **Note** Besides enabling SNMP traps for all IP-RAN notifications, you can also enable traps for IP-RAN GSM alarms, UMTS alarms, and general information about the backhaul utilization. For descriptions on how to use these SNMP commands, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*.
Step 7	Router(config)# **snmp-server enable traps envmon**	Enables SNMP traps for a specific environment, use the **snmp-server enable traps envmon** command.
Step 8	Router(config)# **snmp-server host** *host-addr* [**traps** | **informs**] [**version** {**1** | **2c** | **3** [**auth** | **noauth** | **priv**]}] *community-string* [**udp-port port**] [***notification-type***] **Example:** Router(config)# **snmp-server host 10.20.30.40 version 2c**	Specifies the recipient of an SNMP notification operation. To remove the specified host, use the **no** form of this command. The syntax is as follows: • *host-addr*—Name or Internet address of the host (the targeted recipient). • **traps**—(Optional) Sends SNMP traps to this host. This is the default. • **informs**—(Optional) Sends SNMP informs to this host. • **version**—(Optional) Version of the SNMP used to send the traps. Version 3 is the most secure model because allows packet encryption with the **priv** keyword. If you use the version keyword, one of the following must be specified: – **1**—SNMPv1. This option is not available with informs. – **2c**—SNMPv2C. – **3**—SNMPv3. The following three optional keywords can follow the version 3 keyword: –**auth** (Optional). Enables Message Digest 5 (MD5) and Secure Hash Algorithm (SHA) packet authentication –**noauth** (Default). The noAuthNoPriv security level. This is the default if the [auth | noauth | priv] keyword choice is not specified. –**priv** (Optional). Enables Data Encryption Standard (DES) packet encryption (also called “privacy”). • *community-string*—Password-like community string sent with the notification operation. Though you can set this string using the **snmp-server host** command by itself, we recommend you define this string using the **snmp-server community** command before using the **snmp-server host** command. • **udp-port** *port*—UDP port of the host to use. The default is 162.

Command	Purpose
	• *notification-type*—(Optional) Type of notification to be sent to the host. If no type is specified, all notifications are sent. The notification type can be one or more of the following keywords: – **aaa_server**—Enable SNMP AAA Server traps. – **atm**—Enable SNMP atm Server traps. – **ccme**—Enable SNMP ccme traps. – **cnpd**—Enable NBAR Protocol Discovery traps. – **config**—Enable SNMP config traps. – **config-copy**—Enable SNMP config-copy traps. – **cpu**—Allow cpu related traps. – **dial**—Enable SNMP dial control traps. – **dnis**—Enable SNMP DNIS traps. – **ds0-busyout**—Enable ds0-busyout traps. – **ds1**—Enable SNMP DS1 traps. – **ds1-loopback**—Enable ds1-loopback traps. – **ds3**—Enable SNMP DS3 traps. – **dsp**—Enable SNMP dsp traps. – **eigrp**—Enable SNMP EIGRP traps. – **entity**—Enable SNMP entity traps. – **envmon**—Enable SNMP environmental monitor traps. – **flash**—Enable SNMP FLASH notifications. – **frame-relay**—Enable SNMP frame-relay traps. – **hsrp**—Enable SNMP HSRP traps. – **icsudsu**—Enable SNMP ICSUDSU traps. – **ipmulticast**—Enable SNMP ipmulticast traps. – **ipran**—Enable IP-RAN Backhaul traps. – **ipsla**—Enable SNMP IP SLA traps. – **isdn**—Enable SNMP isdn traps. – **12tun**—Enable SNMP L2 tunnel protocol traps. – **mpls**—Enable SNMP MPLS traps. – **msdp**—Enable SNMP MSDP traps. – **mvpn**—Enable Multicast Virtual Private Networks traps. – **ospf**—Enable OSPF traps. – **pim**—Enable SNMP PIM traps.

	Command	Purpose
		– **pppoe**—Enable SNMP pppoe traps. – **pw**—Enable SNMP PW traps. – **rsvp**—Enable RSVP flow change traps. – **snmp**—Enable SNMP traps. – **srst**—Enable SNMP srst traps. – **syslog**—Enable SNMP syslog traps. – **tty**—Enable TCP connection traps. – **voice**—Enable SNMP voice traps. – **vrrp**—Enable SNMP vrrp traps. – **vtp**—Enable SNMP VTP traps. – **xgcp**—Enable XGCP protocol traps. The example specifies a recipient of the SNMP operation with a host-address of 10.20.30.40 with a version SNMP of SNMPv2C.

Step 9	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits global configuration mode.

Enabling Remote Network Management

To enable remote network management of the Cisco MWR 2941, do the following:

	Command	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`Router(config)# ip host hostname ip_address`	Assigns a host name to each of the network management workstations, where *hostname* is the name assigned to the Operations and Maintenance (O&M) workstation and *ip_address* is the address of the network management workstation.

Step 4	`Router(config)# interface loopback number`
	`Router(config-if)# ip address ip_address subnet_mask`
		Creates a loopback interface for O&M.
		Note For more information about creating loopback interfaces, see Chapter 19, "Configuring Multiprotocol Label Switching."

Step 5	`Router(config-if)# exit`
	`Router(config)#`
		Exits interface configuration mode.

	Command	Purpose
Step 6	`Router(config)#` **`snmp-server host`** *`hostname`* `[`**`traps`** `|` **`informs`**`] [`**`version`** `{`**`1`** `|` **`2c`** `|` **`3`** `[`**`auth`** `|` **`noauth`** `|` **`priv`**`]}]` *`community-string`* `[`**`udp-port`** *`port`*`] [`*`notification-type`*`]`	Specifies the recipient of a Simple Network Management Protocol (SNMP) notification operation. The *hostname* is the name assigned to the Cisco Info Center workstation with the **ip host** command in Step 3.

Step 7	`Router(config)#` **`snmp-server community`** *`public`* **`RO`**
	`Router(config)#` **`snmp-server community`** *`private`* **`RW`**
		Specifies the public and private SNMP community names.

Step 8	`Router(config)#` **`snmp-server enable traps`**	Enables the transmission of SNMP traps.
Step 9	`Router(config)#` **`snmp-server trap-source loopback`** *`number`*	Specifies the loopback interface from which SNMP traps should originate, where *number* is the number of the loopback interface you configured for the O&M in Step 4.

Step 10	**exit**
	Example:
	`Router(config)# exit`
	`Router#`
		Exits configuration mode.

Show Commands for Monitoring the Cisco MWR 2941 Router

To monitor and maintain the Cisco MWR 2941 router, use the following commands:

Command	Purpose
show atm cell-packing	Information about Layer 2 transport ATM cell-packing.
show cem circuit	Summary about the CEM circuit state, including controller, interface, and AC. Also displays specific CEM circuit state, circuit parameters, and statistics/counters.
show cem platform	CEM errors and information.
show connection	Displays the status of interworking connections.
show controllers	All network modules and their interfaces. Also displays the status of the VWIC relays when a VWIC is installed.
show controllers gigabitethernet *slot/port*	Information about initialization block, transmit ring, receive ring, and errors for the Fast Ethernet controller chip.
show controllers e1	Information about controller status specific to the controller hardware. Also displays statistics about the E1 link. If you specify a slot and a port number, statistics for each 15-minute period appears.

Command	Purpose
show controllers t1	Information about cable length, framing, firmware, and errors associated with the T1. With the Cisco MWR 2941 router, this command also shows the status of the relays on the VWIC.
show dsl interface atm	Displays information specific to the asymmetric digital subscriber line (ADSL) for a specified ATM interface.
show gsm traffic	Traffic rates in bits per second at 1 second, 5 seconds, 1 minute, 5 minutes, and 1 hour intervals for GSM data transmitted and received over the backhaul.
show gsm-abis efficiency [history]	The history of the GSM efficiency averages for compression/decompression at 1-second, 5-second, 1-minute, 5-minute, and 1-hour intervals.
show gsm-abis errors	Error statistics counters of the GSM for compression/decompression.
show gsm-abis packets	Packet statistics counters of the GSM for compression/decompression.
show gsm-abis peering [details]	Peering status, statistics, and history of the GSM compression/decompression.
show interface *type slot/port*	Configuration and status of the specified interface.
show interface switchport backup	Status information about the backup switchport.
show interface virtual-cem *slot/port*	Status of the CEM interface.
show interface gigabitethernet *slot/port*	Status of the FE interface.
show ip mroute	Contents of the multicast routing (mroute) table. **Note** Multicast routing applies only to PTP redundancy.
show ip rtp header-compression	RTP header compression statistics.
show ip tcp header-compression	Transmission Control Protocol (TCP)/IP header compression statistics
show mpls l2transport vc	Information about Any Transport over MPLS (AToM) virtual circuits (VCs) that are enabled to route Layer 2 packets on a router.
show network-clocks	Network clocking configuration.
show platform hardware	Status of hardware devices on the Cisco MWR 2941 router.
show policy-map	Configuration of all classes for a specified service policy map or of all classes for all existing policy maps.
show policy-map interface	Statistics and the configurations of the input and output policies that are attached to an interface.

Command	Purpose
show ppp multilink	MLP and multilink bundle information.
show ppp multilink interface *number*	Multilink information for the specified interface.
show protocols	Protocols configured for the router and the individual interfaces.
show ptp clock	PTP clock information.
show ptp foreign-master-record	PTP foreign master records.
show ptp parent	PTP parent properties.
show ptp port	PTP port properties.
show ptp time-property	PTP clock time properties.
show xconnect all	xconnect information.

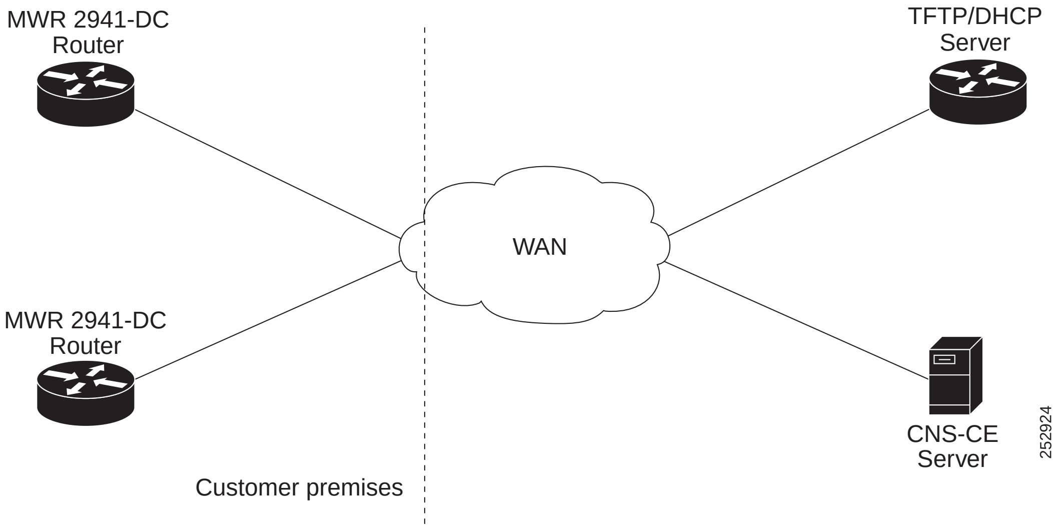
Configuring Cisco Networking Services (CNS)

Cisco Networking Services (CNS) is a collection of services that can provide remote configuration of Cisco IOS networking devices and remote execution of some command-line interface (CLI) commands. CNS allows a Cisco MWR 2941 deployed and powered on in the field to automatically download its configuration.

Note The Cisco MWR 2941 only supports CNS over motherboard Ethernet interfaces. Other interface types do not support CNS.

To enable CNS, you need the following items:

- A DHCP server (standalone or enabled on the carrier edge router)
- A TFTP server (standalone or enabled on the carrier edge router)
- A server running the Cisco Configuration Engine (formerly known as the CNS-CE server)



Note These devices must be connected through onboard Ethernet interfaces. CNS connections over Ethernet HWICs and non-Ethernet interfaces are not supported.

The following sections describe how to configure CNS on the Cisco MWR 2941.

- Process Overview
- Configuring a DHCP Server
- Configuring a TFTP Server
- Configuring the Cisco Configuration Engine
- Verifying the Configuration
- Zero Touch Deployment Sample Configuration

Process Overview

The following sections provide an overview of the steps that take place during a Cisco MWR 2941 zero-touch deployment and image download.

Zero-Touch Deployment

The following sequence of events takes place when a CNS-enabled Cisco MWR 2941 boots and receives a configuration.

1. The Cisco MWR 2941 boots and sends a DHCP Discover message
2. The DHCP Server replies with DHCP Offer
3. The Cisco MWR 2941 sends DHCP Request
4. The DHCP Server replies with option 150 for TFTP
5. The Cisco MWR 2941 requests network-confg file via TFTP
6. The TFTP server sends the Cisco MWR 2941 a network-config file
7. The Cisco MWR 2941 sends an HTTP request to the CNS-CE server
8. The CNS-CE server sends a configuration template to the Cisco MWR 2941
9. Successful event
10. Publish success event

Image Download

The following events take place when a CNS-enabled Cisco MWR 2941 downloads a new image.

1. The CNS-CE server requests inventory (disk/flash info) from the Cisco MWR 2941-DC
2. The Cisco MWR 2941-DC sends an inventory
3. The CNS-CE server sends an image location
4. The Cisco MWR 2941-DC sends an TFTP image request
5. The Cisco MWR 2941-DC downloads an image from the TFTP server
6. The Cisco MWR 2941-DC indicates that the image download is complete
7. The CNS-CE server reboots the Cisco MWR 2941-DC router

Configuring a DHCP Server

The Cisco MWR 2941 requires a DHCP server for zero-touch deployment. The DHCP server is typically implemented on the carrier edge router. You can use the following sample configuration to enable a DHCP server on the edge router.

```
ip dhcp excluded-address 30.30.1.6
ip dhcp excluded-address 30.30.1.20 30.30.1.255
!
ip dhcp pool mwrdhcp
network 30.30.1.0 255.255.255.0
option 150 ip 30.30.1.6
! Specifies the TFTP server address
!
default-router 30.30.1.6
```

Configuring a TFTP Server

You need to set up a TFTP server in order to provide a bootstrap image to 2941s when they boot.

Creating a Bootstrap Configuration

The TFTP server should store a configuration that the Cisco MWR 2941 uses to boot. The following sample configuration specifies 30.30.1.20 as the CNS server IP address and port 80 for the configuration service.

```
hostname test-2941
!
cns trusted-server all-agents 30.30.1.20
cns event 30.30.1.20 11011 keepalive 60 3
cns config initial 30.30.1.20 80
cns config partial 30.30.1.20 80
cns id hostname
cns id hostname event
cns id hostname image
cns exec 80
logging buffered 20000
!
end
```

For more information about the commands used in this configuration, see the *Cisco MWR 2941 Mobile Wireless Edge Router IOS Command Reference, Release 15.0(1)MR*. *Cisco Configuration Engine Installation & Configuration Guide* at
http://www.cisco.com/en/US/products/sw/netmgtsw/ps4617/tsd_products_support_series_home.html.

Enabling a TFTP Server on the Edge Router

The Cisco MWR 2941 requires a TFTP server for zero-touch deployment. The TFTP server is typically implemented on the carrier edge router. You can use the following global configuration commands enable a TFTP server on the edge router that can send a configuration to the Cisco MWR 2941 router.

```
tftp-server sup-bootflash:network-confg
tftp-server sup-bootflash:test-2941-confg
```

Once the Cisco MWR 2941 boots with this configuration, it can connect to the CNS-CE server.

Configuring the Cisco Configuration Engine

The Cisco Configuration Engine (formerly known as the Cisco CNS Configuration Engine) allows you to remotely manage configurations and IOS software images on Cisco devices including the Cisco MWR 2941.

Once the Cisco MWR 2941 downloads the bootstrap configuration and connects to the Cisco Configuration Engine server, you can use the server to download a full configuration to the router. You can also use the CNS-CE server to complete any of the following tasks:

- Manage configuration templates—The CNS-CE server can store and manage configuration templates.
- Download a new image—You can use the CNS-CE server to load a new IOS image on a Cisco MWR 2941 router.
- Loading a new config—You can use the CNS-CE server to load a new configuration file on a Cisco MWR 2941 router.
- Enable identification—You can use a unique CNS agent ID to verify the identity of a host device prior to communication with the CNS-CE server.
- Enable Authentication—You can configure the CNS-CE server to require a unique password from the 2941 router as part of any communication handshake.
- Enable encryption—You can enable Secure Socket Layer (SSL) encryption for the HTTP sessions between the CNS agent devices (Cisco MWR 2941 routers) and the CNS-CE server.

For instructions about how to use the CNS-CE server, see the *Cisco Configuration Engine Installation & Configuration Guide* at http://www.cisco.com/en/US/products/sw/netmgtsw/ps4617/tsd_products_support_series_home.html.

Verifying the Configuration

You can use the following IOS commands to verify the CNS configuration on the Cisco MWR 2941.

- **show cns event connection**
- **show cns image connection**
- **show cns image inventory**
- **debug cns all**

Zero Touch Deployment Sample Configuration

The following configuration example sets the Cisco MWR 2941 to boot using configurations stored on a CNS–CE server with the IP address 30.30.1.20.

Note This section provides partial configurations intended to demonstrate a specific feature.

```
hostname 2941
!
cns trusted-server all-agents 30.30.1.20
cns event 30.30.1.20 11011 keepalive 60 3
cns config initial 30.30.1.20 80
cns config partial 30.30.1.20 80
cns id hostname
cns id hostname event
cns id hostname image
cns exec 80
logging buffered 20000
!
end
```

INDEX