

Ethernet Card Software Feature and Configuration Guide

For the Cisco ONS 15454 SDH, Cisco ONS 15454, and Cisco ONS 15327
Cisco IOS Release 12.2(27)SV
CTC and Documentation Release 6.2
Last Updated: August 2012

Corporate 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: 408 526-4100



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's 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's 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 and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Ethernet Card Software Feature and Configuration Guide
Copyright 2012 Cisco Systems, Inc. All rights reserved.



About the Documentation xxiii

Revision History xxiii

Document Objectives xxiv

Audience xxiv

Document Organization xxiv

Related Documentation xxvi

Document Conventions xxvi

Obtaining Optical Networking Information xxxii

Where to Find Safety and Warning Information xxxii

Cisco Optical Networking Product Documentation CD-ROM xxxii

Obtaining Documentation and Submitting a Service Request xxxii

CHAPTER 1

ML-Series Card Overview 1-1

ML-Series Card Description 1-1

ML-Series Feature List 1-2

Key ML-Series Features 1-5

Cisco IOS 1-5

GFP-F Framing 1-5

DRPRI 1-5

EoMPLS 1-6

Link Aggregation (FEC, GEC, and POS) 1-6

RMON 1-6

RPR 1-6

SNMP 1-6

TL1 1-7

VRF Lite 1-7

CHAPTER 2

CTC Operations 2-1

Displaying ML-Series POS And Ethernet Statistics on CTC 2-1

Displaying ML-Series Ethernet Ports Provisioning Information on CTC 2-2

Displaying ML-Series POS Ports Provisioning Information on CTC 2-4

Provisioning Framing Mode 2-5

Managing SONET/SDH Alarms 2-6

Displaying Maintenance Information 2-7
 Provisioning SONET/SDH Circuits 2-8
 J1 Path Trace 2-8

CHAPTER 3

Initial Configuration 3-1

Hardware Installation 3-1
 Cisco IOS on the ML-Series Card 3-2
 Opening a Cisco IOS Session Using CTC 3-2
 Telnetting to the Node IP Address and Slot Number 3-3
 Telnetting to a Management Port 3-4
 ML-Series IOS CLI Console Port 3-4
 RJ-11 to RJ-45 Console Cable Adapter 3-5
 Connecting a PC or Terminal to the Console Port 3-5
 Startup Configuration File 3-7
 Manually Creating a Startup Configuration File Through the Serial Console Port 3-7
 Passwords 3-8
 Configuring the Management Port 3-8
 Configuring the Hostname 3-9
 CTC and the Startup Configuration File 3-9
 Loading a Cisco IOS Startup Configuration File Through CTC 3-10
 Database Restore of the Startup Configuration File 3-11
 Multiple Microcode Images 3-11
 Changing the Working Microcode Image 3-12
 Version Up Software Upgrade 3-14
 Node and Card Behavior During Version Up 3-14
 Enabling and Completing Version Up 3-14
 Cisco IOS Command Modes 3-17
 Using the Command Modes 3-18
 Exit 3-18
 Getting Help 3-19

CHAPTER 4

Configuring Interfaces 4-1

General Interface Guidelines 4-1
 MAC Addresses 4-2
 Interface Port ID 4-2
 Basic Interface Configuration 4-3
 Basic Fast Ethernet, Gigabit Ethernet, and POS Interface Configuration 4-4
 Configuring the Fast Ethernet Interfaces (ML100T-12) 4-4

Configuring the Gigabit Ethernet Interface (ML1000-2)	4-5
Configuring Gigabit Ethernet Remote Failure Indication (RFI)	4-6
Monitoring and Verifying Gigabit Ethernet Remote Failure Indication (RFI)	4-7
Configuring the POS Interfaces (ML100T-12 and ML1000-2)	4-10
CRC Threshold Configuration	4-10
Monitoring Operations on the Fast Ethernet and Gigabit Ethernet Interfaces	4-11

CHAPTER 5**Configuring POS 5-1**

POS on the ML-Series Card	5-1
ML-Series SONET and SDH Circuit Sizes	5-1
VCAT	5-2
SW-LCAS	5-3
Framing Mode, Encapsulation, and CRC Support	5-4
Configuring Framing Mode	5-4
Configuring Encapsulation Type	5-4
Configuring CRC Size in HDLC Framing	5-5
Setting the MTU Size	5-5
Configuring Keep Alive Messages	5-6
SONET/SDH Alarms	5-6
Configuring SONET/SDH Alarms	5-7
Configuring SONET/SDH Delay Triggers	5-7
Configuring SONET/SDH Alarms	5-7
C2 Byte and Scrambling	5-8
Third-Party POS Interfaces C2 Byte and Scrambling Values	5-9
Configuring SPE Scrambling	5-9
Monitoring and Verifying POS	5-9
POS Configuration Examples	5-11
ML-Series Card to ML-Series Card	5-11
ML-Series Card to Cisco 12000 GSR-Series Router	5-12
ML-Series Card to G-Series Card	5-13
ML-Series Card to ONS 15310 ML-100T-8 Card	5-14

CHAPTER 6**Configuring Bridges 6-1**

Understanding Bridging	6-1
Configuring Bridging	6-2
Monitoring and Verifying Bridging	6-3

CHAPTER 7

Configuring STP and RSTP 7-1

STP Features 7-1

STP Overview 7-2

Supported STP Instances 7-2

Bridge Protocol Data Units 7-2

Election of the Root Switch 7-3

Bridge ID, Switch Priority, and Extended System ID 7-4

Spanning-Tree Timers 7-4

Creating the Spanning-Tree Topology 7-4

Spanning-Tree Interface States 7-5

Blocking State 7-6

Listening State 7-7

Learning State 7-7

Forwarding State 7-7

Disabled State 7-7

Spanning-Tree Address Management 7-8

STP and IEEE 802.1Q Trunks 7-8

Spanning Tree and Redundant Connectivity 7-8

Accelerated Aging to Retain Connectivity 7-9

RSTP 7-9

Supported RSTP Instances 7-9

Port Roles and the Active Topology 7-9

Rapid Convergence 7-10

Synchronization of Port Roles 7-12

Bridge Protocol Data Unit Format and Processing 7-13

Processing Superior BPDU Information 7-14

Processing Inferior BPDU Information 7-14

Topology Changes 7-14

Interoperability with IEEE 802.1D STP 7-15

Configuring STP and RSTP Features 7-15

Default STP and RSTP Configuration 7-16

Disabling STP and RSTP 7-16

Configuring the Root Switch 7-17

Configuring the Port Priority 7-17

Configuring the Path Cost 7-18

Configuring the Switch Priority of a Bridge Group 7-19

Configuring the Hello Time 7-19

Configuring the Forwarding-Delay Time for a Bridge Group 7-20

Configuring the Maximum-Aging Time for a Bridge Group 7-20

Verifying and Monitoring STP and RSTP Status 7-20

CHAPTER 8

Configuring VLANs 8-1

Understanding VLANs 8-1

Configuring IEEE 802.1Q VLAN Encapsulation 8-2

IEEE 802.1Q VLAN Configuration 8-3

Monitoring and Verifying VLAN Operation 8-5

CHAPTER 9

Configuring IEEE 802.1Q Tunneling and Layer 2 Protocol Tunneling 9-1

Understanding IEEE 802.1Q Tunneling 9-1

Configuring IEEE 802.1Q Tunneling 9-4

IEEE 802.1Q Tunneling and Compatibility with Other Features 9-4

Configuring an IEEE 802.1Q Tunneling Port 9-4

IEEE 802.1Q Example 9-5

Understanding VLAN-Transparent and VLAN-Specific Services 9-6

VLAN-Transparent and VLAN-Specific Services Configuration Example 9-7

Understanding Layer 2 Protocol Tunneling 9-9

Configuring Layer 2 Protocol Tunneling 9-10

Default Layer 2 Protocol Tunneling Configuration 9-10

Layer 2 Protocol Tunneling Configuration Guidelines 9-11

Configuring Layer 2 Tunneling on a Port 9-11

Configuring Layer 2 Tunneling Per-VLAN 9-12

Monitoring and Verifying Tunneling Status 9-12

CHAPTER 10

Configuring Link Aggregation 10-1

Understanding Link Aggregation 10-1

Configuring EtherChannel 10-2

EtherChannel Configuration Example 10-3

Configuring POS Channel 10-4

POS Channel Configuration Example 10-5

Understanding Encapsulation over EtherChannel or POS Channel 10-7

Configuring Encapsulation over EtherChannel or POS Channel 10-7

Encapsulation over EtherChannel Example 10-7

Monitoring and Verifying EtherChannel and POS 10-9

CHAPTER 11

Configuring Networking Protocols 11-1

Basic IP Routing Protocol Configuration 11-1

RIP 11-2

- EIGRP 11-2
- OSPF 11-2
- BGP 11-3
- Enabling IP Routing 11-3
- Configuring IP Routing 11-4
 - Configuring RIP 11-4
 - RIP Authentication 11-7
 - Summary Addresses and Split Horizon 11-8
 - Configuring OSPF 11-9
 - OSPF Interface Parameters 11-13
 - OSPF Area Parameters 11-14
 - Other OSPF Behavior Parameters 11-16
 - Change LSA Group Pacing 11-18
 - Loopback Interface 11-19
 - Monitoring OSPF 11-19
 - Configuring EIGRP 11-20
 - EIGRP Router Mode Commands 11-22
 - EIGRP Interface Mode Commands 11-23
 - Configure EIGRP Route Authentication 11-25
 - Monitoring and Maintaining EIGRP 11-26
 - Border Gateway Protocol and Classless Interdomain Routing 11-27
 - Configuring BGP 11-27
 - Verifying the BGP Configuration 11-28
 - Configuring IS-IS 11-29
 - Verifying the IS-IS Configuration 11-30
- Configuring Static Routes 11-31
- Monitoring Static Routes 11-32
- Monitoring and Maintaining the IP Network 11-33
- Understanding IP Multicast Routing 11-33
 - Configuring IP Multicast Routing 11-34
 - Monitoring and Verifying IP Multicast Operation 11-35

CHAPTER 12

Configuring IRB 12-1

- Understanding Integrated Routing and Bridging 12-1
- Configuring IRB 12-2
- IRB Configuration Example 12-3
- Monitoring and Verifying IRB 12-4

CHAPTER 13**Configuring VRF Lite 13-1**

- Understanding VRF Lite 13-1
- Configuring VRF Lite 13-2
- VRF Lite Configuration Example 13-3
- Monitoring and Verifying VRF Lite 13-7

CHAPTER 14**Configuring Quality of Service 14-1**

- Understanding QoS 14-1
 - Priority Mechanism in IP and Ethernet 14-2
 - IP Precedence and Differentiated Services Code Point 14-2
 - Ethernet CoS 14-3
- ML-Series QoS 14-4
 - Classification 14-4
 - Policing 14-5
 - Marking and Discarding with a Policer 14-5
 - Queuing 14-6
 - Scheduling 14-6
 - Control Packets and L2 Tunneled Protocols 14-8
 - Egress Priority Marking 14-8
 - Ingress Priority Marking 14-8
 - QinQ Implementation 14-8
 - Flow Control Pause and QoS 14-9
- QoS on RPR 14-10
- Configuring QoS 14-10
 - Creating a Traffic Class 14-10
 - Creating a Traffic Policy 14-11
 - Attaching a Traffic Policy to an Interface 14-15
 - Configuring CoS-Based QoS 14-16
- Monitoring and Verifying QoS Configuration 14-16
- QoS Configuration Examples 14-17
 - Traffic Classes Defined Example 14-18
 - Traffic Policy Created Example 14-18
 - class-map match-any and class-map match-all Commands Example 14-19
 - match spr1 Interface Example 14-19
 - ML-Series VoIP Example 14-20
 - ML-Series Policing Example 14-21
 - ML-Series CoS-Based QoS Example 14-21
- Understanding Multicast QoS and Multicast Priority Queuing 14-23

- Default Multicast QoS 14-23
- Multicast Priority Queuing QoS Restrictions 14-24
- Configuring Multicast Priority Queuing QoS 14-24
- QoS not Configured on Egress 14-26
- ML-Series Egress Bandwidth Example 14-26
 - Case 1: QoS with Priority and Bandwidth Configured Without Priority Multicast 14-26
 - Case 2: QoS with Priority and Bandwidth Configured with Priority Multicast 14-27
- Understanding CoS-Based Packet Statistics 14-28
- Configuring CoS-Based Packet Statistics 14-28

CHAPTER 15

Configuring the Switching Database Manager 15-1

- Understanding the SDM 15-1
- SDM Regions 15-1
- Configuring SDM 15-2
 - Configuring SDM Regions 15-2
 - Configuring Access Control List Size in TCAM 15-3

CHAPTER 16

Configuring Access Control Lists 16-1

- Understanding ACLs 16-1
- ML-Series ACL Support 16-1
 - IP ACLs 16-2
 - Named IP ACLs 16-2
 - User Guidelines 16-2
 - Creating IP ACLs 16-3
 - Creating Numbered Standard and Extended IP ACLs 16-3
 - Creating Named Standard IP ACLs 16-4
 - Creating Named Extended IP ACLs (Control Plane Only) 16-4
 - Applying the ACL to an Interface 16-4
- Modifying ACL TCAM Size 16-5

CHAPTER 17

Configuring Resilient Packet Ring 17-1

- Understanding RPR 17-1
 - Role of SONET/SDH Circuits 17-2
 - Packet Handling Operations 17-2
 - Ring Wrapping 17-3
 - RPR Framing Process 17-5
 - MAC Address and VLAN Support 17-6
 - RPR QoS 17-6

CTM and RPR	17-6
Configuring RPR	17-7
Connecting the ML-Series Cards with Point-to-Point STS/STM Circuits	17-7
Configuring CTC Circuits for RPR	17-7
CTC Circuit Configuration Example for RPR	17-8
Configuring RPR Characteristics and the SPR Interface on the ML-Series Card	17-11
Assigning the ML-Series Card POS Ports to the SPR Interface	17-13
Creating the Bridge Group and Assigning the Ethernet and SPR Interfaces	17-14
RPR Cisco IOS Configuration Example	17-15
Verifying Ethernet Connectivity Between RPR Ethernet Access Ports	17-17
CRC threshold configuration and detection	17-17
Monitoring and Verifying RPR	17-17
Add an ML-Series Card into an RPR	17-18
Adding an ML-Series Card into an RPR	17-20
Delete an ML-Series Card from an RPR	17-22
Deleting an ML-Series Card from an RPR	17-25
Understanding RPR Link Fault Propagation	17-27
LFP Sequence	17-28
Propagation Delays	17-29
Configuring LFP	17-29
LFP Configuration Requirements	17-30
Monitoring and Verifying LFP	17-30
RPR Keep Alive	17-31
Configuring RPR Keep Alive	17-31
Monitoring and Verifying RPR Keep Alive	17-32
RPR Shortest Path	17-33
Configuring Shortest Path and Topology Discovery	17-35
Monitoring and Verifying Topology Discovery and Shortest Path Load Balancing	17-35
Understanding Dual RPR Interconnect	17-36
Configuring DRPRI	17-37
DRPRI IOS Configuration Example	17-39
Monitoring and Verifying DRPRI	17-43

CHAPTER 18**Configuring Ethernet over MPLS 18-1**

Understanding EoMPLS	18-1
EoMPLS Support	18-2
EoMPLS Restrictions	18-3
EoMPLS Quality of Service	18-3

- Configuring EoMPLS 18-4
 - EoMPLS Configuration Guidelines 18-6
 - VC Type 4 Configuration on PE-CLE Port 18-6
 - VC Type 5 Configuration on PE-CLE Port 18-7
 - EoMPLS Configuration on PE-CLE SPR Interface 18-8
 - EoMPLS Configuration on PE-CLE POS Interface 18-8
 - Bridge Group Configuration on MPLS Cloud-facing Port and SPR Interface 18-9
 - Bridge Group Configuration on MPLS Cloud-facing Port and POS Interface 18-10
 - Setting the Priority of Packets with the EXP 18-10
- EoMPLS Configuration Example 18-11
- Monitoring and Verifying EoMPLS 18-13
- Understanding MPLS-TE 18-14
 - RSVP on the ML-Series Card 18-14
 - Ethernet FCS Preservation 18-14
- Configuring MPLS-TE 18-15
 - Configuring an ML-Series Card for Tunnels Support 18-15
 - Configuring an Interface to Support RSVP-Based Tunnel Signalling and IGP Flooding 18-15
 - Configuring OSPF and Refresh Reduction for MPLS-TE 18-16
 - Configuring an MPLS-TE Tunnel 18-16
- MPLS-TE Configuration Example 18-17
- Monitoring and Verifying MPLS-TE and IP RSVP 18-19
- RPRW Alarm 18-20

CHAPTER 19

- Configuring RMON 19-1**
 - Understanding RMON 19-1
 - Configuring RMON 19-2
 - Default RMON Configuration 19-2
 - Configuring RMON Alarms and Events 19-2
 - Collecting Group History Statistics on an Interface 19-5
 - Collecting Group Ethernet Statistics on an Interface 19-6
 - Understanding ML-Series Card CRC Error Threshold 19-7
 - Configurable Threshold and Triggered Actions 19-7
 - SONET/GFP Suppression of CRC-ALARM 19-8
 - Clearing of CRC-ALARM 19-8
 - Unwrap Synchronization 19-8
 - Unidirectional Errors 19-8
 - Bidirectional Errors 19-10
 - Configuring the ML-Series Card CRC Error Threshold 19-13

Clearing the CRC-ALARM Wrap with the Clear CRC Error Command	19-14
Configuring ML-Series Card RMON for CRC Errors	19-16
Configuration Guidelines for CRC Thresholds on the ML-Series Card	19-16
Accessing CRC Errors Through SNMP	19-16
ifIndex Operation in the ML-Series Card	19-16
Configuring an SNMP Trap for the CRC Error Threshold Using Cisco IOS	19-17
Determining the ifIndex Number for an ML-Series Card	19-19
Manually Checking CRC Errors on the ML-Series Card	19-21
Displaying RMON Status	19-21

CHAPTER 20**Configuring SNMP 20-1**

Understanding SNMP	20-1
SNMP on the ML-Series Card	20-2
SNMP Versions	20-3
SNMP Manager Functions	20-3
SNMP Agent Functions	20-4
SNMP Community Strings	20-4
Using SNMP to Access MIB Variables	20-4
Supported MIBs	20-5
SNMP Notifications	20-5
Configuring SNMP	20-5
Default SNMP Configuration	20-6
SNMP Configuration Guidelines	20-6
Disabling the SNMP Agent	20-7
Configuring Community Strings	20-7
Configuring SNMP Groups and Users	20-8
Configuring SNMP Notifications	20-10
Setting the Agent Contact and Location Information	20-12
Limiting TFTP Servers Used Through SNMP	20-13
SNMP Examples	20-13
Displaying SNMP Status	20-14

CHAPTER 21**POS on ONS Ethernet Cards 21-1**

POS Overview	21-1
POS Interoperability	21-2
POS Encapsulation Types	21-4
IEEE 802.17b	21-4
LEX	21-5
PPP/BCP	21-6

- Cisco HDLC 21-6
- E-Series Proprietary 21-7
- POS Framing Modes 21-7
 - HDLC Framing 21-7
 - GFP-F Framing 21-7
- POS Characteristics of Specific ONS Ethernet Cards 21-7
 - ONS 15327 E-10/100-4 Framing and Encapsulation Options 21-8
 - ONS 15454 and ONS 15454 SDH E-Series Framing and Encapsulation Options 21-8
 - G-Series Encapsulation and Framing 21-9
 - ONS 15454, ONS 15454 SDH, ONS 15310-CL, and and ONS 15310-MA CE-Series Cards Encapsulation and Framing 21-10
 - ONS 15310 ML-100T-8 Encapsulation and Framing 21-10
 - ONS 15454 and ONS 15454 SDH ML-Series Protocol Encapsulation and Framing 21-10
- Ethernet Clocking Versus SONET/SDH Clocking 21-11

CHAPTER 22

E-Series and G-Series Ethernet Operation 22-1

- G-Series Application 22-1
 - G1K-4 and G1000-4 Comparison 22-2
 - G-Series Example 22-3
 - IEEE 802.3z Flow Control and Frame Buffering 22-3
 - Gigabit EtherChannel/IEEE 802.3ad Link Aggregation 22-4
 - Ethernet Link Integrity Support 22-5
 - Enhanced State Model for Gigabit Ethernet Ports 22-6
- G-Series Circuit Configurations 22-6
 - G-Series Point-to-Point Ethernet Circuits 22-7
 - G-Series Manual Cross-Connects 22-7
- G-Series Gigabit Ethernet Transponder Mode 22-8
 - Two-Port Bidirectional Transponder Mode 22-10
 - One-Port Bidirectional Transponder Mode 22-11
 - Two-Port Unidirectional Transponder Mode 22-11
 - G-Series Transponder Mode Characteristics 22-12
- E-Series Application 22-13
 - E-Series Modes 22-13
 - E-Series Multicard EtherSwitch Group 22-14
 - E-Series Single-Card EtherSwitch 22-14
 - Port-Mapped (Linear Mapper) 22-15
 - Available Circuit Sizes For E-Series Modes 22-16
 - Available Total Bandwidth For E-Series Modes 22-16
 - E-Series IEEE 802.3z Flow Control 22-16

E-Series VLAN Support	22-17
E-Series Q-Tagging (IEEE 802.1Q)	22-18
E-Series Priority Queuing (IEEE 802.1Q)	22-20
E-Series Spanning Tree (IEEE 802.1D)	22-21
E-Series Multi-Instance Spanning Tree and VLANs	22-22
Spanning Tree on a Circuit-by-Circuit Basis	22-22
E-Series Spanning Tree Parameters	22-22
E-Series Spanning Tree Configuration	22-23
E-Series Circuit Configurations	22-23
E-Series Circuit Protection	22-24
E-Series Point-to-Point Ethernet Circuits	22-24
E-Series Shared Packet Ring Ethernet Circuits	22-25
E-Series Hub-and-Spoke Ethernet Circuit Provisioning	22-26
E-Series Ethernet Manual Cross-Connects	22-27
Remote Monitoring Specification Alarm Thresholds	22-27

CHAPTER 23

CE-100T-8 Ethernet Operation	23-1
CE-100T-8 Overview	23-1
CE-100T-8 Ethernet Features	23-2
Autonegotiation, Flow Control, and Frame Buffering	23-2
Ethernet Link Integrity Support	23-3
Enhanced State Model for Ethernet and SONET/SDH Ports	23-4
IEEE 802.1Q CoS and IP ToS Queuing	23-4
RMON and SNMP Support	23-6
Statistics and Counters	23-6
CE-100T-8 SONET Circuits and Features	23-6
Available Circuit Sizes and Combinations	23-7
CE-100T-8 Pools and STS/VT Allocation Tab	23-9
CE-100T-8 VCAT Characteristics	23-10
CE-100T-8 POS Encapsulation, Framing, and CRC	23-10
CE-100T-8 Loopback, J1 Path Trace, and SONET Alarms	23-11

APPENDIX 24

Command Reference	24-1
--------------------------	-------------

APPENDIX 25

Unsupported CLI Commands	25-1
Unsupported Privileged Exec Commands	25-1
Unsupported Global Configuration Commands	25-1
Unsupported POS Interface Configuration Commands	25-3

Unsupported FastEthernet or GigabitEthernet Interface Configuration Commands 25-4
Unsupported Port-Channel Interface Configuration Commands 25-5
Unsupported BVI Interface Configuration Commands 25-6

APPENDIX 26

Using Technical Support 26-1

Gathering Information About Your Internetwork 26-1
Getting the Data from Your ML-Series Card 26-2
Providing Data to Your Technical Support Representative 26-3

INDEX



<i>Figure 2-1</i>	Displaying ML-Series POS Statistics	2-2
<i>Figure 2-2</i>	Displaying ML-Series Ethernet Port Provisioning Information	2-3
<i>Figure 2-3</i>	Displaying POS Port Provisioning Information	2-4
<i>Figure 2-4</i>	Displaying Framing Mode	2-6
<i>Figure 2-5</i>	Managing ML-Series SONET/SDH Alarms	2-7
<i>Figure 2-6</i>	Displaying Maintenance Information	2-8
<i>Figure 3-1</i>	CTC IOS Window	3-3
<i>Figure 3-2</i>	CTC Node View Showing IP Address and Slot Number	3-4
<i>Figure 3-3</i>	Console Cable Adapter	3-5
<i>Figure 3-4</i>	Connecting to the Console Port	3-6
<i>Figure 3-5</i>	Node Defaults Delayed Upgrade Settings	3-15
<i>Figure 3-6</i>	Software Activation Dialog	3-16
<i>Figure 5-1</i>	ML-Series Card to ML-Series Card POS Configuration	5-11
<i>Figure 5-2</i>	ML-Series Card to Cisco 12000 Series Gigabit Switch Router (GSR) POS Configuration	5-12
<i>Figure 5-3</i>	ML-Series Card to G-Series Card POS Configuration	5-14
<i>Figure 5-4</i>	ML-Series Card to ONS 15310 CE-100T-8 Card Configuration	5-14
<i>Figure 6-1</i>	Bridging Example	6-3
<i>Figure 7-1</i>	Spanning-Tree Topology	7-5
<i>Figure 7-2</i>	Spanning-Tree Interface States	7-6
<i>Figure 7-3</i>	Spanning Tree and Redundant Connectivity	7-8
<i>Figure 7-4</i>	Proposal and Agreement Handshaking for Rapid Convergence	7-12
<i>Figure 7-5</i>	Sequence of Events During Rapid Convergence	7-13
<i>Figure 8-1</i>	VLANs Spanning Devices in a Network	8-2
<i>Figure 8-2</i>	Bridging IEEE 802.1Q VLANs	8-4
<i>Figure 9-1</i>	IEEE 802.1Q Tunnel Ports in a Service-Provider Network	9-2
<i>Figure 9-2</i>	Normal, IEEE 802.1Q, and IEEE 802.1Q-Tunneled Ethernet Packet Formats	9-3
<i>Figure 9-3</i>	ERMS Example	9-7
<i>Figure 10-1</i>	Encapsulation over EtherChannel Example	10-3
<i>Figure 10-2</i>	POS Channel Example	10-6
<i>Figure 10-3</i>	Encapsulation over EtherChannel Example	10-8
<i>Figure 11-1</i>	IP Routing Protocol Example Using OSPF	11-11

Figure 12-1	Configuring IRB	12-3
Figure 13-1	VRF Lite—Sample Network Scenario	13-3
Figure 14-1	IP Precedence and DSCP	14-3
Figure 14-2	Ethernet Frame and the CoS Bit (IEEE 802.1p)	14-3
Figure 14-3	ML-Series QoS Flow	14-4
Figure 14-4	Dual Leaky Bucket Policer Model	14-5
Figure 14-5	Queuing and Scheduling Model	14-7
Figure 14-6	QinQ	14-9
Figure 14-7	ML-Series VoIP Example	14-20
Figure 14-8	ML-Series Policing Example	14-21
Figure 14-9	ML-Series CoS Example	14-22
Figure 14-10	QoS not Configured on Egress	14-26
Figure 17-1	RPR Packet Handling Operations	17-3
Figure 17-2	RPR Ring Wrapping	17-4
Figure 17-3	RPR Frame for ML-Series Card	17-5
Figure 17-4	RPR Frame Fields	17-5
Figure 17-5	Three Node RPR Example	17-8
Figure 17-6	CTC Card View for ML-Series Card	17-9
Figure 17-7	CTC Circuit Creation Wizard	17-10
Figure 17-8	RPR Bridge Group	17-15
Figure 17-9	Two-Node RPR Before the Addition	17-19
Figure 17-10	Three Node RPR After the Addition	17-19
Figure 17-11	Three Node RPR Before the Deletion	17-23
Figure 17-12	Two Node RPR After the Deletion	17-24
Figure 17-13	RPR Link Fault Propagation Example	17-28
Figure 17-14	Shortest and Longest Path	17-34
Figure 17-15	Dual RPR Interconnect Network and Paired Cards	17-36
Figure 18-1	EoMPLS Service Provider Network	18-2
Figure 18-2	EoMPLS Configuration Example	18-5
Figure 18-3	MPLS-TE Configuration Example	18-17
Figure 19-1	Remote Monitoring Example	19-2
Figure 19-2	Wrapped RPR with Unidirectional Excessive CRC Errors	19-9
Figure 19-3	Unwrapped RPR with Unidirectional Excessive CRC Errors	19-10
Figure 19-4	Wrapped RPR with Bidirectional Excessive CRC Errors	19-11
Figure 19-5	First Stage of Unwrapped RPR with Bidirectional Excessive CRC Errors	19-12

Figure 19-6	Second Stage of Unwrapped RPR with Bidirectional Excessive CRC Errors	19-13
Figure 20-1	SNMP on the ML-Series Card Example	20-2
Figure 20-2	SNMP Network	20-4
Figure 21-1	Ethernet to POS Process on ONS Node	21-2
Figure 21-2	RPR Data Frames	21-5
Figure 21-3	LEX Under HDLC Framing	21-5
Figure 21-4	BCP Under HDLC Framing	21-6
Figure 21-5	PPP Frame Under HDLC Framing	21-6
Figure 21-6	Cisco HDLC Under HDLC Framing	21-6
Figure 21-7	ONS 15327 E-Series Encapsulation and Framing Options	21-8
Figure 21-8	ONS 15454 and ONS 15454 SDH E-Series Encapsulation and Framing Options	21-9
Figure 21-9	ONS G-Series Encapsulation and Framing Options	21-9
Figure 21-10	ONS CE-100T-8 and ONS CE-1000-4 Encapsulation and Framing Options	21-10
Figure 21-11	ML-Series Card Framing and Encapsulation Options	21-11
Figure 22-1	Data Traffic on a G-Series Point-to-Point Circuit	22-3
Figure 22-2	G-Series Gigabit EtherChannel (GEC) Support	22-5
Figure 22-3	End-to-End Ethernet Link Integrity Support	22-5
Figure 22-4	G-Series Point-to-Point Circuit	22-7
Figure 22-5	G-Series Manual Cross-Connects	22-8
Figure 22-6	Card Level Overview of G-Series One-Port Transponder Mode Application	22-9
Figure 22-7	G-Series in Default SONET/SDH Mode	22-9
Figure 22-8	G-Series Card in Transponder Mode (Two-Port Bidirectional)	22-10
Figure 22-9	One-Port Bidirectional Transponder Mode	22-11
Figure 22-10	Two-Port Unidirectional Transponder	22-12
Figure 22-11	Multicard EtherSwitch Configuration	22-14
Figure 22-12	Single-Card EtherSwitch Configuration	22-15
Figure 22-13	E-Series Mapping Ethernet Ports to STS/VC Circuits	22-15
Figure 22-14	Edit Circuit Dialog Box Featuring Available VLANs	22-18
Figure 22-15	Q-tag Moving Through VLAN	22-19
Figure 22-16	Priority Queuing Process	22-20
Figure 22-17	STP Blocked Path	22-21
Figure 22-18	Spanning Tree Map on Circuit Window	22-22
Figure 22-19	Multicard EtherSwitch Point-to-Point Circuit	22-24
Figure 22-20	Single-Card EtherSwitch or Port-Mapped Point-to-Point Circuit	22-25
Figure 22-21	Shared Packet Ring Ethernet Circuit	22-26

<i>Figure 22-22</i>	Hub-and-Spoke Ethernet Circuit	22-26
<i>Figure 23-1</i>	CE-100T-8 Point-to-Point Circuit	23-1
<i>Figure 23-2</i>	Flow Control	23-3
<i>Figure 23-3</i>	End-to-End Ethernet Link Integrity Support	23-3
<i>Figure 23-4</i>	CE-100T-8 STS/VT Allocation Tab	23-9



TABLES

<i>Table 2-1</i>	ML-Series POS and Ethernet Statistics Fields and Buttons	2-2
<i>Table 3-1</i>	RJ-11 to RJ-45 Pin Mapping	3-5
<i>Table 3-2</i>	Microcode Image Feature Comparison	3-12
<i>Table 3-3</i>	Cisco IOS Command Modes	3-17
<i>Table 5-1</i>	SONET STS Circuit Capacity in Line Rate Mbps	5-2
<i>Table 5-2</i>	VCAT Circuit Sizes Supported by ML100T-12 and ML1000-2 Cards	5-2
<i>Table 5-3</i>	VCAT Circuit Sizes Supported by ML100T-12 and ML1000-2 Cards	5-3
<i>Table 5-4</i>	Supported Encapsulation, Framing, and CRC Sizes for ML-Series Cards on the ONS 15454 and ONS 15454 SDH	5-4
<i>Table 5-5</i>	Default MTU Size	5-6
<i>Table 5-6</i>	C2 Byte and Scrambling Default Values	5-8
<i>Table 5-7</i>	ML-Series Parameter Configuration for Connection to a Cisco 12000 GSR-Series Router	5-13
<i>Table 7-1</i>	Switch Priority Value and Extended System ID	7-4
<i>Table 7-2</i>	Spanning-Tree Timers	7-4
<i>Table 7-3</i>	Port State Comparison	7-10
<i>Table 7-4</i>	RSTP BPDU Flags	7-13
<i>Table 7-5</i>	Default STP and RSTP Configuration	7-16
<i>Table 7-6</i>	Commands for Displaying Spanning-Tree Status	7-20
<i>Table 9-1</i>	VLAN-Transparent Service Versus VLAN-Specific Services	9-6
<i>Table 9-2</i>	Default Layer 2 Protocol Tunneling Configuration	9-11
<i>Table 9-3</i>	Commands for Monitoring and Maintaining Tunneling	9-13
<i>Table 11-1</i>	Default RIP Configuration	11-5
<i>Table 11-2</i>	Default OSPF Configuration	11-10
<i>Table 11-3</i>	Show IP OSPF Statistics Commands	11-19
<i>Table 11-4</i>	Default EIGRP Configuration	11-21
<i>Table 11-5</i>	IP EIGRP Clear and Show Commands	11-26
<i>Table 11-6</i>	BGP Show Commands	11-28
<i>Table 11-7</i>	IS-IS Show Commands	11-30
<i>Table 11-8</i>	Routing Protocol Default Administrative Distances	11-32
<i>Table 11-9</i>	Commands to Clear IP Routes or Display Route Status	11-33
<i>Table 11-10</i>	IP Multicast Routing Show Commands	11-35
<i>Table 12-1</i>	Commands for Monitoring and Verifying IRB	12-5

Table 12-2	show interfaces irb Field Descriptions	12-6
Table 13-1	Commands for Monitoring and Verifying VRF Lite	13-7
Table 14-1	Traffic Class Commands	14-11
Table 14-2	Traffic Policy Commands	14-12
Table 14-3	CoS Commit Command	14-16
Table 14-4	Commands for QoS Status	14-17
Table 14-5	CoS Multicast Priority Queuing Command	14-25
Table 14-6	Packet Statistics on ML-Series Card Interfaces	14-28
Table 14-7	CoS-Based Packet Statistics Command	14-29
Table 14-8	Commands for CoS-Based Packet Statistics	14-29
Table 15-1	Default Partitioning by Application Region in TCAM	15-2
Table 15-2	Partitioning the TCAM Size for ACLs	15-3
Table 16-1	Commands for Numbered Standard and Extended IP ACLs	16-3
Table 16-2	Applying ACL to Interface	16-5
Table 17-1	Definitions of RPR Frame Fields	17-6
Table 18-1	Applicable EoMPLS QoS Statements and Actions	18-4
Table 18-2	Commands for Monitoring and Maintaining Tunneling	18-13
Table 18-3	Commands for Monitoring and Verifying MPLS-TE	18-19
Table 18-4	Commands for Monitoring and Verifying IP RSVP	18-20
Table 19-1	Port Numbers for the Interfaces of ML-Series Cards	19-20
Table 19-2	Port Numbers for the Interfaces of ML-Series Cards	19-20
Table 19-3	Commands for Displaying RMON Status	19-22
Table 20-1	SNMP Operations	20-3
Table 20-2	Default SNMP Configuration	20-6
Table 20-3	ML-Series Card Notification Types	20-10
Table 20-4	Commands for Displaying SNMP Information	20-14
Table 21-1	ONS SONET/SDH Ethernet Card Interoperability under HDLC Framing with Encapsulation Type and CRC	21-3
Table 21-2	ONS SONET/SDH Ethernet Card Interoperability under GFP-F Framing with Encapsulation Type	21-4
Table 22-1	ONS 15454 and ONS 15327 E-Series Ethernet Circuit Sizes	22-16
Table 22-2	ONS 15454 and ONS 15327 E-Series Total Bandwidth Available	22-16
Table 22-3	Priority Queuing	22-20
Table 22-4	Spanning Tree Parameters	22-23
Table 22-5	Spanning Tree Configuration	22-23
Table 22-6	Protection for E-Series Circuit Configurations	22-24
Table 23-1	IP ToS Priority Queue Mappings	23-5

<i>Table 23-2</i>	CoS Priority Queue Mappings	23-5
<i>Table 23-3</i>	CE-100T-8 Supported Circuit Sizes	23-7
<i>Table 23-4</i>	SONET Circuit Sizes and Ethernet Services	23-7
<i>Table 23-5</i>	CCAT High Order Circuit Size Combinations	23-7
<i>Table 23-6</i>	VCAT High Order Circuit Combinations for STS-1-3v and STS-1-2v	23-8
<i>Table 23-7</i>	CE-100T-8 Maximum Service Densities	23-8



About the Documentation



Note

The terms “Unidirectional Path Switched Ring” and “UPSR” may appear in Cisco literature. These terms do not refer to using Cisco ONS 15xxx products in a unidirectional path switched ring configuration. Rather, these terms, as well as “Path Protected Mesh Network” and “PPMN,” refer generally to Cisco’s path protection feature, which may be used in any topological network configuration. Cisco does not recommend using its path protection feature in any particular topological network configuration.

This section explains the objectives, intended audience, and organization of this publication and describes the conventions that convey instructions and other information.

Revision History

Date	Notes
March 2007	Revision History Table added for the first time
August 2007	<ul style="list-style-type: none">• Updated the “Enabling and Completing Version Up” section in Chapter 3, Initial Configuration.• Updated About the Documentation chapter.
January 2009	Added the following sections in Chapter 14, Configuring Quality of Service: <ul style="list-style-type: none">• QoS not Configured on Egress• ML-Series Egress Bandwidth Example
October 2009	Deleted the following caution from Chapter 4, Configuring Interfaces: <ul style="list-style-type: none">• Do not use the abbreviations g0 or g1 for Gigabit Ethernet user-defined abbreviations. This creates an unsupported group asynchronous interface.
February 2010	Updated image in the chapter, “Configuring VRF Lite”.
March 2011	Updated the section “IEEE 802.3z Flow Control and Frame Buffering” in the chapter “E-Series and G-Series Ethernet Operation”.
August 2012	The full length book-PDF was generated.

This section provides the following information:

- [Document Objectives](#)
- [Audience](#)

- [Document Organization](#)
- [Related Documentation](#)
- [Document Conventions](#)
- [Obtaining Optical Networking Information](#)
- [Obtaining Documentation and Submitting a Service Request](#)

Document Objectives

This guide covers the software features and operations of Ethernet cards for the Cisco ONS 15454, Cisco ONS 15454 SDH, and Cisco ONS 15327. It explains software features and configuration for Cisco IOS on the ML-Series card. The ML-Series card is a module in the Cisco ONS 15454 SONET or Cisco ONS 15454 SDH system. It also explains software feature and configuration for CTC on the E-Series, G-Series and CE-100T-8 cards. The E-Series cards and G-Series cards are modules in the Cisco ONS 15454, Cisco ONS 15454 SDH, and Cisco ONS 15327. The CE-100T-8 card is a module in the Cisco ONS 15454. The CE-100T-8 card is also available as module for the Cisco ONS 15310. The Cisco ONS 15310 version of the card is covered in the *Ethernet Card Software Feature and Configuration Guide* for the ONS 15310. Use this guide in conjunction with the appropriate publications listed in the [Related Documentation](#) section.

Audience

To use the ML-Series card chapters of this publication, you should be familiar with Cisco IOS and preferably have technical networking background and experience. To use the E-Series, G-Series and CE-100T-8 card chapters of this publication, you should be familiar with CTC and preferably have technical networking background and experience.

Document Organization

This *Ethernet Card Software Feature and Configuration Guide* is organized into the following chapters:

- [Chapter 1, “ML-Series Card Overview,”](#) provides a description of the ML-Series card, a feature list, and explanations of key features.
- [Chapter 2, “CTC Operations,”](#) provides details and procedures for using Cisco Transport Controller (CTC) software with the ML-Series card.
- [Chapter 3, “Initial Configuration,”](#) provides procedures to access the ML-Series card and create and manage startup configuration files.
- [Chapter 4, “Configuring Interfaces,”](#) provides information on the ML-Series card interfaces and basic procedures for the interfaces.
- [Chapter 5, “Configuring POS,”](#) provides information on the ML-Series card POS interfaces and advanced procedures for the POS interfaces.
- [Chapter 6, “Configuring Bridges,”](#) provides bridging examples and procedures for the ML-Series card.
- [Chapter 7, “Configuring STP and RSTP,”](#) provides spanning tree and rapid spanning tree examples and procedures for the ML-Series card.

- [Chapter 8, “Configuring VLANs,”](#) provides VLAN examples and procedures for the ML-Series card.
- [Chapter 9, “Configuring IEEE 802.1Q Tunneling and Layer 2 Protocol Tunneling,”](#) provides tunneling examples and procedures for the ML-Series card.
- [Chapter 10, “Configuring Link Aggregation,”](#) provides Etherchannel and packet-over-SONET/SDH (POS) channel examples and procedures for the ML-Series card.
- [Chapter 11, “Configuring Networking Protocols,”](#) provides network protocol examples and procedures for the ML-Series card.
- [Chapter 12, “Configuring IRB,”](#) provides integrated routing and bridging (IRB) examples and procedures for the ML-Series card.
- [Chapter 13, “Configuring VRF Lite,”](#) provides VPN Routing and Forwarding Lite (VRF Lite) examples and procedures for the ML-Series card.
- [Chapter 14, “Configuring Quality of Service,”](#) provides quality of service (QoS) examples and procedures for the ML-Series card.
- [Chapter 15, “Configuring the Switching Database Manager,”](#) provides switching database manager examples and procedures for the ML-Series card.
- [Chapter 16, “Configuring Access Control Lists,”](#) provides access control list (ACL) examples and procedures for the ML-Series card.
- [Chapter 17, “Configuring Resilient Packet Ring,”](#) provides resilient packet ring (RPR) examples and procedures for the ML-Series card.
- [Chapter 18, “Configuring Ethernet over MPLS,”](#) provides Ethernet over Multiprotocol Label Switching (EoMPLS) examples and procedures for the ML-Series card.
- [Chapter 19, “Configuring RMON,”](#) provides Remote Monitoring (RMON) examples and procedures for the ML-Series card.
- [Chapter 20, “Configuring SNMP,”](#) provides Simple Network Monitoring Protocol (SNMP) examples and procedures for the ML-Series card.
- [Chapter 21, “POS on ONS Ethernet Cards,”](#) details and explains POS on Ethernet cards for the ONS 15454, ONS 15454 SDH or ONS 15327 platform. It also details Ethernet card interoperability for the platform.
- [Chapter 22, “E-Series and G-Series Ethernet Operation,”](#) details and explains the features and operation of E-Series and G-Series Ethernet cards for the ONS 15454, ONS 15454 SDH and ONS 15327 platform.
- [Chapter 23, “CE-100T-8 Ethernet Operation,”](#) details and explains the features and operation of CE-Series Ethernet card for the ONS 15454.
- [Appendix A, “Command Reference,”](#) is an alphabetical listing of unique ML-Series card Cisco IOS commands with definitions and examples.
- [Appendix B, “Unsupported CLI Commands,”](#) is a categorized and alphabetized listing of Cisco IOS commands that the ML-Series card does not support.
- [Appendix C, “Using Technical Support,”](#) instructs the user on using the Cisco Technical Assistance Center (Cisco TAC) for ML-Series card problems.

Related Documentation

Use this *Ethernet Card Software Feature and Configuration Guide* in conjunction with the following general ONS 15454 or ONS 15454 SDH system publications:

- To install, turn up, provision, and maintain a Cisco ONS 15454 node and network, refer to the *Cisco ONS 15454 Procedure Guide*.
- For alarm clearing, general troubleshooting, and hardware replacement procedures for a Cisco ONS 15454 node, refer to the *Cisco ONS 15454 Troubleshooting Guide*.
- For detailed reference information on a Cisco ONS 15454 node, refer to the *Cisco ONS 15454 Reference Manual*.
- To install, turn up, provision, and maintain a Cisco ONS 15454 SDH node and network, refer to the *Cisco ONS 15454 SDH Procedure Guide*.
- For alarm clearing, general troubleshooting, and hardware replacement procedures for the Cisco ONS 15454 SDH node, refer to the *Cisco ONS 15454 SDH Troubleshooting Guide*.
- For detailed reference information on the Cisco ONS 15454 SDH node, refer to the *Cisco ONS 15454 SDH Reference Manual*.

The ML-Series card employs the Cisco IOS Modular QoS CLI (MQC). For more information on general MQC configuration, refer to the following Cisco IOS documents:

- Cisco IOS Quality of Service Solutions Configuration Guide, Release 12.2
- Cisco IOS Quality of Service Solutions Command Reference, Release 12.2

The ML-Series card employs Cisco IOS 12.2. For more general information on Cisco IOS 12.2, refer to the extensive Cisco IOS documentation at:

- <http://www.cisco.com/>

Document Conventions

This publication uses the following conventions:

Convention	Application
boldface	Commands and keywords in body text.
<i>italic</i>	Command input that is supplied by the user.
[]	Keywords or arguments that appear within square brackets are optional.
{ x x x }	A choice of keywords (represented by x) appears in braces separated by vertical bars. The user must select one.
Ctrl	The control key. For example, where Ctrl + D is written, hold down the Control key while pressing the D key.
screen font	Examples of information displayed on the screen.

Convention**Application****boldface screen font**

Examples of information that the user must enter.

< >

Command parameters that must be replaced by module-specific codes.

**Note**

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

**Caution**

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

**Warning****IMPORTANT SAFETY INSTRUCTIONS**

This warning symbol means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Use the statement number provided at the end of each warning to locate its translation in the translated safety warnings that accompanied this device. Statement 1071

SAVE THESE INSTRUCTIONS**Waarschuwing****BELANGRIJKE VEILIGHEIDSINSTRUCTIES**

Dit waarschuwingssymbool betekent gevaar. U verkeert in een situatie die lichamelijk letsel kan veroorzaken. Voordat u aan enige apparatuur gaat werken, dient u zich bewust te zijn van de bij elektrische schakelingen betrokken risico's en dient u op de hoogte te zijn van de standaard praktijken om ongelukken te voorkomen. Gebruik het nummer van de verklaring onderaan de waarschuwing als u een vertaling van de waarschuwing die bij het apparaat wordt geleverd, wilt raadplegen.

BEWAAR DEZE INSTRUCTIES**Varoitus****TÄRKEITÄ TURVALLISUUSOHJEITA**

Tämä varoitusmerkki merkitsee vaaraa. Tilanne voi aiheuttaa ruumiillisia vammoja. Ennen kuin käsittelet laitteistoa, huomioi sähköpiirien käsittelyyn liittyvät riskit ja tutustu onnettomuuksien yleisiin ehkäisytapoihin. Turvallisuusvaroitusten käännökset löytyvät laitteen mukana toimitettujen käännettyjen turvallisuusvaroitusten joukosta varoitusten lopussa näkyvien lausuntonumeroiden avulla.

SÄILYTÄ NÄMÄ OHJEET

Attention IMPORTANTES INFORMATIONS DE SÉCURITÉ

Ce symbole d'avertissement indique un danger. Vous vous trouvez dans une situation pouvant entraîner des blessures ou des dommages corporels. Avant de travailler sur un équipement, soyez conscient des dangers liés aux circuits électriques et familiarisez-vous avec les procédures couramment utilisées pour éviter les accidents. Pour prendre connaissance des traductions des avertissements figurant dans les consignes de sécurité traduites qui accompagnent cet appareil, référez-vous au numéro de l'instruction situé à la fin de chaque avertissement.

CONSERVEZ CES INFORMATIONS**Warnung WICHTIGE SICHERHEITSHINWEISE**

Dieses Warnsymbol bedeutet Gefahr. Sie befinden sich in einer Situation, die zu Verletzungen führen kann. Machen Sie sich vor der Arbeit mit Geräten mit den Gefahren elektrischer Schaltungen und den üblichen Verfahren zur Vorbeugung vor Unfällen vertraut. Suchen Sie mit der am Ende jeder Warnung angegebenen Anweisungsnummer nach der jeweiligen Übersetzung in den übersetzten Sicherheitshinweisen, die zusammen mit diesem Gerät ausgeliefert wurden.

BEWAHREN SIE DIESE HINWEISE GUT AUF.**Avvertenza IMPORTANTI ISTRUZIONI SULLA SICUREZZA**

Questo simbolo di avvertenza indica un pericolo. La situazione potrebbe causare infortuni alle persone. Prima di intervenire su qualsiasi apparecchiatura, occorre essere al corrente dei pericoli relativi ai circuiti elettrici e conoscere le procedure standard per la prevenzione di incidenti. Utilizzare il numero di istruzione presente alla fine di ciascuna avvertenza per individuare le traduzioni delle avvertenze riportate in questo documento.

CONSERVARE QUESTE ISTRUZIONI**Advarsel VIKTIGE SIKKERHETSINSTRUKSJONER**

Dette advarselssymbolet betyr fare. Du er i en situasjon som kan føre til skade på person. Før du begynner å arbeide med noe av utstyret, må du være oppmerksom på farene forbundet med elektriske kretser, og kjenne til standardprosedyrer for å forhindre ulykker. Bruk nummeret i slutten av hver advarsel for å finne oversettelsen i de oversatte sikkerhetsadvarslene som fulgte med denne enheten.

TA VARE PÅ DISSE INSTRUKSJONENE**Aviso INSTRUÇÕES IMPORTANTES DE SEGURANÇA**

Este símbolo de aviso significa perigo. Você está em uma situação que poderá ser causadora de lesões corporais. Antes de iniciar a utilização de qualquer equipamento, tenha conhecimento dos perigos envolvidos no manuseio de circuitos elétricos e familiarize-se com as práticas habituais de prevenção de acidentes. Utilize o número da instrução fornecido ao final de cada aviso para localizar sua tradução nos avisos de segurança traduzidos que acompanham este dispositivo.

GUARDE ESTAS INSTRUÇÕES

¡Advertencia! INSTRUCCIONES IMPORTANTES DE SEGURIDAD

Este símbolo de aviso indica peligro. Existe riesgo para su integridad física. Antes de manipular cualquier equipo, considere los riesgos de la corriente eléctrica y familiarícese con los procedimientos estándar de prevención de accidentes. Al final de cada advertencia encontrará el número que le ayudará a encontrar el texto traducido en el apartado de traducciones que acompaña a este dispositivo.

GUARDE ESTAS INSTRUCCIONES**Varning! VIKTIGA SÄKERHETSANVISNINGAR**

Denna varningssignal signalerar fara. Du befinner dig i en situation som kan leda till personskada. Innan du utför arbete på någon utrustning måste du vara medveten om farorna med elkretsar och känna till vanliga förfaranden för att förebygga olyckor. Använd det nummer som finns i slutet av varje varning för att hitta dess översättning i de översatta säkerhetsvarningar som medföljer denna anordning.

SPARA DESSA ANVISNINGAR**FONTOS BIZTONSÁGI ELOÍRÁSOK**

Ez a figyelmeztető jel veszélyre utal. Sérülésveszélyt rejtő helyzetben van. Mielőtt bármely berendezésen munkát végezte, legyen figyelemmel az elektromos áramkörök okozta kockázatokra, és ismerkedjen meg a szokásos balesetvédelmi eljárásokkal. A kiadványban szereplő figyelmeztetések fordítása a készülékhez mellékelt biztonsági figyelmeztetések között található; a fordítás az egyes figyelmeztetések végén látható szám alapján kereshető meg.

ORIZZE MEG EZEKET AZ UTASÍTÁSOKAT!**Предупреждение ВАЖНЫЕ ИНСТРУКЦИИ ПО СОБЛЮДЕНИЮ ТЕХНИКИ БЕЗОПАСНОСТИ**

Этот символ предупреждения обозначает опасность. То есть имеет место ситуация, в которой следует опасаться телесных повреждений. Перед эксплуатацией оборудования выясните, каким опасностям может подвергаться пользователь при использовании электрических цепей, и ознакомьтесь с правилами техники безопасности для предотвращения возможных несчастных случаев. Воспользуйтесь номером заявления, приведенным в конце каждого предупреждения, чтобы найти его переведенный вариант в переводе предупреждений по безопасности, прилагаемом к данному устройству.

СОХРАНИТЕ ЭТИ ИНСТРУКЦИИ**警告 重要的安全性说明**

此警告符号代表危险。您正处于可能受到严重伤害的工作环境中。在您使用设备开始工作之前，必须充分意识到触电的危险，并熟练掌握防止事故发生的标准工作程序。请根据每项警告结尾提供的声明号码来找到此设备的安全性警告说明的翻译文本。

请保存这些安全性说明

警告 安全上の重要な注意事項

「危険」の意味です。人身事故を予防するための注意事項が記述されています。装置の取り扱い作業を行うときは、電気回路の危険性に注意し、一般的な事故防止策に留意してください。警告の各国語版は、各注意事項の番号を基に、装置に付属の「Translated Safety Warnings」を参照してください。

これらの注意事項を保管しておいてください。

주의 중요 안전 지침

이 경고 기호는 위험을 나타냅니다. 작업자가 신체 부상을 일으킬 수 있는 위험한 환경에 있습니다. 장비에 작업을 수행하기 전에 전기 회로와 관련된 위험을 숙지하고 표준 작업 관례를 숙지하여 사고를 방지하십시오. 각 경고의 마지막 부분에 있는 경고문 번호를 참조하여 이 장치와 함께 제공되는 번역된 안전 경고문에서 해당 번역문을 찾으십시오.

이 지시 사항을 보관하십시오.

Aviso INSTRUÇÕES IMPORTANTES DE SEGURANÇA

Este símbolo de aviso significa perigo. Você se encontra em uma situação em que há risco de lesões corporais. Antes de trabalhar com qualquer equipamento, esteja ciente dos riscos que envolvem os circuitos elétricos e familiarize-se com as práticas padrão de prevenção de acidentes. Use o número da declaração fornecido ao final de cada aviso para localizar sua tradução nos avisos de segurança traduzidos que acompanham o dispositivo.

GUARDE ESTAS INSTRUÇÕES**Advarsel** VIGTIGE SIKKERHEDSANVISNINGER

Dette advarselssymbol betyder fare. Du befinder dig i en situation med risiko for legemeskade. Før du begynder arbejde på udstyr, skal du være opmærksom på de involverede risici, der er ved elektriske kredsløb, og du skal sætte dig ind i standardprocedurer til undgåelse af ulykker. Brug erklæringsnummeret efter hver advarsel for at finde oversættelsen i de oversatte advarsler, der fulgte med denne enhed.

GEM DISSE ANVISNINGER**تحذير****إرشادات الأمان الهامة**

يوضح رمز التحذير هذا وجود خطر. وهذا يعني أنك متواجد في مكان قد ينتج عنه التعرض لإصابات. قبل بدء العمل، احذر مخاطر التعرض للصدمات الكهربائية وكن على علم بالإجراءات القياسية للحيلولة دون وقوع أي حوادث. استخدم رقم البيان الموجود في آخر كل تحذير لتحديد مكان ترجمته داخل تحذيرات الأمان المترجمة التي تأتي مع الجهاز. قم بحفظ هذه الإرشادات

Upozorenje VAŽNE SIGURNOSNE NAPOMENE

Ovaj simbol upozorenja predstavlja opasnost. Nalazite se u situaciji koja može prouzročiti tjelesne ozljede. Prije rada s bilo kojim uređajem, morate razumjeti opasnosti vezane uz električne sklopove, te biti upoznati sa standardnim načinima izbjegavanja nesreća. U prevedenim sigurnosnim upozorenjima, priloženima uz uređaj, možete prema broju koji se nalazi uz pojedino upozorenje pronaći i njegov prijevod.

SAČUVAJTE OVE UPUTE**Upozornění DŮLEŽITÉ BEZPEČNOSTNÍ POKYNY**

Tento upozorňující symbol označuje nebezpečí. Jste v situaci, která by mohla způsobit nebezpečí úrazu. Před prací na jakémkoliv vybavení si uvědomte nebezpečí související s elektrickými obvody a seznamte se se standardními opatřeními pro předcházení úrazům. Podle čísla na konci každého upozornění vyhledejte jeho překlad v přeložených bezpečnostních upozorněních, která jsou přiložena k zařízení.

USCHOVEJTE TYTO POKYNY**Προειδοποίηση ΣΗΜΑΝΤΙΚΕΣ ΟΔΗΓΙΕΣ ΑΣΦΑΛΕΙΑΣ**

Αυτό το προειδοποιητικό σύμβολο σημαίνει κίνδυνο. Βρίσκεστε σε κατάσταση που μπορεί να προκαλέσει τραυματισμό. Πριν εργαστείτε σε οποιοδήποτε εξοπλισμό, να έχετε υπόψη σας τους κινδύνους που σχετίζονται με τα ηλεκτρικά κυκλώματα και να έχετε εξοικειωθεί με τις συνήθειες πρακτικές για την αποφυγή ατυχημάτων. Χρησιμοποιήστε τον αριθμό δήλωσης που παρέχεται στο τέλος κάθε προειδοποίησης, για να εντοπίσετε τη μετάφρασή της στις μεταφρασμένες προειδοποιήσεις ασφαλείας που συνοδεύουν τη συσκευή.

ΦΥΛΑΞΤΕ ΑΥΤΕΣ ΤΙΣ ΟΔΗΓΙΕΣ**אזהרה****הוראות בטיחות חשובות**

סימן אזהרה זה מסמל סכנה. אתה נמצא במצב העלול לגרום לפציעה. לפני שתעבוד עם ציוד כלשהו, עליך להיות מודע לסכנות הכרוכות במעגלים חשמליים ולהכיר את הנהלים המקובלים למניעת תאונות. השתמש במספר ההוראה המסופק בסופה של כל אזהרה כדי לאתר את התרגום באזהרות הבטיחות המתורגמות שמצורפות להתקן.

שמור הוראות אלה

Opomena **ВАЖНИ БЕЗБЕДНОСНИ НАПАТСТВИЈА**
 Симболот за предупредување значи опасност. Се наоѓате во ситуација што може да предизвика телесни повреди. Пред да работите со опремата, бидете свесни за ризикот што постои кај електричните кола и треба да ги познавате стандардните постапки за спречување на несреќни случаи. Искористете го бројот на изјавата што се наоѓа на крајот на секое предупредување за да го најдете неговиот период во преведените безбедносни предупредувања што се испорачани со уредот.
ЧУВАЈТЕ ГИ ОБИЕ НАПАТСТВИЈА

Ostrzeżenie **WAŻNE INSTRUKCJE DOTYCZĄCE BEZPIECZEŃSTWA**

Ten symbol ostrzeżenia oznacza niebezpieczeństwo. Zachodzi sytuacja, która może powodować obrażenia ciała. Przed przystąpieniem do prac przy urządzeniach należy zapoznać się z zagrożeniami związanymi z układami elektrycznymi oraz ze standardowymi środkami zapobiegania wypadkom. Na końcu każdego ostrzeżenia podano numer, na podstawie którego można odszukać tłumaczenie tego ostrzeżenia w dołączonym do urządzenia dokumencie z tłumaczeniami ostrzeżeń.

NINIEJSZE INSTRUKCJE NALEŻY ZACHOWAĆ

Obtaining Optical Networking Information

This section contains information that is specific to optical networking products. For information that pertains to all of Cisco, refer to the [Obtaining Documentation and Submitting a Service Request](#) section.

Where to Find Safety and Warning Information

For safety and warning information, refer to the *Cisco Optical Transport Products Safety and Compliance Information* document that accompanied the product. This publication describes the international agency compliance and safety information for the Cisco ONS 15454 system. It also includes translations of the safety warnings that appear in the ONS 15454 system documentation.

Cisco Optical Networking Product Documentation CD-ROM

Optical networking-related documentation, including Cisco ONS 15xxx product documentation, is available in a CD-ROM package that ships with your product. The Optical Networking Product Documentation CD-ROM is updated periodically and may be more current than printed documentation.

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, 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>

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS Version 2.0.



ML-Series Card Overview

This chapter provides an overview of the ML1000-2 and ML100T-12 cards for the ONS 15454 (SONET) and ONS 15454 SDH. It lists Ethernet and SONET/SDH capabilities and Cisco IOS and Cisco Transport Controller (CTC) software features, with brief descriptions of selected features.

This chapter contains the following major sections:

- [ML-Series Card Description, page 1-1](#)
- [ML-Series Feature List, page 1-2](#)
- [Key ML-Series Features, page 1-5](#)

ML-Series Card Description

The ML-Series cards are independent Gigabit Ethernet (ML1000-2) or Fast Ethernet (ML100T-12 and ML100X-8) Layer 3 switches that process up to 5.7 Mpps. The cards are integrated into the ONS 15454 SONET or the ONS 15454 SDH. An ONS 15454 SONET with a 10-Gigabit Cross-Connect (XC10G or XC-VXC-10G) card can host the card in any traffic card slot, but an ONS 15454 SONET with a Cross-Connect (XC) card or Cross Connect Virtual Tributary (XCVT) card can only host the ML-Series card in the four traffic slots. An ONS 15454 SDH can host the card in any traffic card slot with any cross-connect card.

The ML-Series card uses Cisco IOS Release 12.2(27)SV, and the Cisco IOS command-line interface (CLI) is the primary user interface for the ML-Series card. Most configuration for the card, such as Ethernet port, bridging, and VLAN, can be done only through the Cisco IOS CLI.

However, CTC, the ONS 15454 SONET/SDH graphical user interface (GUI), also supports the ML-Series card. SONET/SDH circuits cannot be provisioned through Cisco IOS, but must be configured through CTC or TL1. CTC offers ML-Series card status information, SONET/SDH alarm management, Cisco IOS Telnet session initialization, Cisco IOS configuration file management, provisioning, inventory, and other standard functions.

The ML100T-12 features twelve RJ-45 interfaces, and the ML100X-8 and ML1000-2 features two Small Form-factor Pluggable (SFP) slots supporting short wavelength (SX) and long wavelength (LX) optical modules. All three cards use the same hardware and software base and offer similar feature sets. For detailed card specifications, refer to the "Ethernet Cards" chapter of the Cisco ONS 15454 Reference Manual or the Cisco ONS 15454 SDH Reference Manual.

The ML-Series card features two virtual packet-over-SONET/SDH (POS) ports, which function in a manner similar to OC-N card ports. The SONET/SDH circuits are provisioned through CTC in the same manner as standard OC-N card circuits. The ML-Series POS ports support virtual concatenation (VCAT) of SONET/SDH circuits and a software link capacity adjustment scheme (SW-LCAS).

ML-Series Feature List

The ML100T-12, ML100X-8, and the ML1000-2 cards have the following features:

- Layer 1 data features:
 - 10/100BASE-TX half-duplex and full-duplex data transmission
 - 100BASE-FX full-duplex data transmission with Auto-MDIX (ML100X-8)
 - 1000BASE-SX, 1000BASE-LX full-duplex data transmission
 - IEEE 802.3z (Gigabit Ethernet) and 802.3x (Fast Ethernet) Flow Control
- SONET/SDH features:
 - High-level data link control (HDLC) or frame-mapped generic framing procedure (GFP-F) framing mechanism for POS
 - Two POS virtual ports
 - LEX, Cisco HDLC or Point-to-Point Protocol/Bridging Control Protocol (PPP/BCP) encapsulation for POS
 - VCAT with SW-LCAS
 - G-Series card and ONS 15327 E-Series card compatible (with LEX encapsulation only)
- Layer 2 bridging features:
 - Transparent bridging
 - MAC address learning, aging, and switching by hardware
 - Protocol tunneling
 - Multiple Spanning Tree (MST) protocol tunneling
 - 255 active bridge group maximum
 - 60,000 MAC address maximum per card and 8,000 MAC address maximum per bridge group
 - Integrated routing and bridging (IRB)
 - IEEE 802.1P/Q-based VLAN trunking
 - IEEE 802.1Q VLAN tunneling
 - IEEE 802.1D Spanning Tree Protocol (STP) and IEEE 802.1W Rapid Spanning Tree Protocol (RSTP)
 - IEEE 802.1D STP instance per bridge group
 - Dual RPR Interconnect (DRPRI)
 - Ethernet over Multiprotocol Label Switching (EoMPLS)
 - EoMPLS traffic engineering (EoMPLS-TE) with RSVP
 - VLAN-transparent and VLAN-specific services (Ethernet Relay Multipoint Service [ERMS])
- Resilient packet ring (RPR)
 - Ethernet FCS preservation for customers
 - CRC error alarm generation
 - FCS detection and threshold configuration
 - Shortest path determination
 - Keep alives

- Fast EtherChannel (FEC) features (ML100T-12):
 - Bundling of up to four Fast Ethernet ports
 - Load sharing based on source and destination IP addresses of unicast packets
 - Load sharing for bridge traffic based on MAC addresses
 - IRB
 - IEEE 802.1Q trunking
 - Up to 6 active FEC port channels
- Gigabit EtherChannel (GEC) features (ML1000-2):
 - Bundling the two Gigabit Ethernet ports
 - Load sharing for bridge traffic based on MAC addresses
 - IRB
 - IEEE 802.1Q trunking
 - Auto-negotiation with Remote Fault Indication (RFI)
- POS channel:
 - Bundling the two POS ports
 - LEX encapsulation only
 - IRB
 - IEEE 802.1Q trunking
- Layer 3 routing, switching, and forwarding:
 - Default routes
 - IP unicast and multicast forwarding
 - Simple IP access control lists (ACLs) (both Layer 2 and Layer 3 forwarding path)
 - Extended IP ACLs in software (control-plane only)
 - IP and IP multicast routing and switching between Ethernet ports
 - Reverse Path Forwarding (RPF) multicast (not RPF unicast)
 - Load balancing among equal cost paths based on source and destination IP addresses
 - Up to 18,000 IP routes
 - Up to 20,000 IP host entries
 - Up to 40 IP multicast groups
 - IRB routing mode support
- Supported routing protocols:
 - Virtual Private Network (VPN) Routing and Forwarding Lite (VRF Lite)
 - Intermediate System-to-Intermediate System (IS-IS) Protocol
 - Routing Information Protocol (RIP and RIP II)
 - Enhanced Interior Gateway Routing Protocol (EIGRP)
 - Open Shortest Path First (OSPF) Protocol
 - Protocol Independent Multicast (PIM)—Sparse, sparse-dense, and dense modes
 - Secondary addressing

- Static routes
- Local proxy ARP
- Border Gateway Protocol (BGP)
- Classless interdomain routing (CIDR)
- Quality of service (QoS) features:
 - Multicast priority queuing classes
 - Service level agreements (SLAs) with 1-Mbps granularity
 - Input policing
 - Guaranteed bandwidth (weighted round-robin [WDRR] plus strict priority scheduling)
 - Low latency queuing support for unicast Voice-over-IP (VoIP)
 - Class of service (CoS) based on Layer 2 priority, VLAN ID, Layer 3 Type of Service/DiffServ Code Point (TOS/DSCP), and port
 - CoS-based packet statistics
- Additional protocols:
 - Cisco Discovery Protocol (CDP) support on Ethernet ports
 - Dynamic Host Configuration Protocol (DHCP) relay
 - Hot Standby Router Protocol (HSRP) over 10/100 Ethernet, Gigabit Ethernet, FEC, GEC, and Bridge Group Virtual Interface (BVI)
 - Internet Control Message Protocol (ICMP)
- Management features:
 - Cisco IOS
 - CTC
 - Remote monitoring (RMON)
 - Simple Network Management Protocol (SNMP)
 - Transaction Language 1 (TL1)
- System features:
 - Automatic field programmable gate array (FPGA) Upgrade
 - Network Equipment Building Systems 3 (NEBS3) compliant
 - Multiple microcode images
 - Version up to independently upgrade individual ML-Series cards
- CTC features:
 - Framing Mode Provisioning
 - Standard STS/STM and VCAT circuit provisioning for POS virtual ports
 - SONET/SDH alarm reporting for path alarms and other ML-Series card specific alarms, including RPR-WRAP
 - Raw port statistics
 - Standard inventory and card management functions
 - J1 path trace

- Cisco IOS CLI Telnet sessions from CTC
- Cisco IOS startup configuration file management from CTC

Key ML-Series Features

This section describes selected key features and their implementation on the ML-Series cards.

Cisco IOS

Cisco IOS controls the data functions of the ML-Series cards and comes preloaded on the ONS 15454 SONET/SDH Advanced Timing, Communications, and Control (TCC2) card and Advanced Timing, Communications, and Control Plus (TCC2P) card. Users cannot update the ML-Series Cisco IOS image in the same manner as the Cisco IOS system image on a Cisco Catalyst Series. An ML-Series Cisco IOS image upgrade is accomplished only through the ONS 15454 SONET/SDH CTC, and Cisco IOS images for the ML-Series cards are available only as part of an ONS 15454 SONET or SDH software release. This Cisco IOS image is included on the standard ONS 15454 SONET/SDH System Software CD under the package file name M_I.bin and full file name ons15454m-i7-mz. The images are not available for download or shipped separately.

GFP-F Framing

GFP defines a standard-based mapping of different types of services onto SONET/SDH. The ML-Series and CE-Series support frame-mapped GFP (GFP-F), which is the PDU-oriented client signal adaptation mode for GFP. GFP-F maps one variable length data packet onto one GFP packet.

GFP is composed of common functions and payload specific functions. Common functions are those shared by all payloads. Payload-specific functions are different depending on the payload type. GFP is detailed in the ITU recommendation G.7041.

DRPRI

The bridge-group protocol DRPRI is an RPR mechanism that interconnects rings for protection from ONS node failure. The protocol provides two parallel connections of the rings linked by a special instance of RSTP. One connection is the active node and the other is the standby node. During a failure of the active node, link, or card, a proprietary algorithm detects the failure and causes a switchover to the standby node. DRPRI provides a less than 200-msec recovery time for Layer 2 bridged traffic when the ML-Series cards use the enhanced microcode image. The Layer 2 recovery time is up to 12 seconds for other microcode images. The recovery time for Layer 3 unicast and multicast traffic also depends on the convergence time of the routing protocol implemented regardless of the microcode image used.

EoMPLS

EoMPLS provides a tunneling mechanism for Ethernet traffic through an MPLS-enabled Layer 3 core. It encapsulates Ethernet protocol data units (PDUs) inside MPLS packets and using label stacking forwards them across the MPLS network. EoMPLS is an Internet Engineering Task Force (IETF) standard-track protocol based on the Martini draft. EoMPLS allows service providers to offer customers a virtual Ethernet line service or VLAN service using the service provider's existing MPLS backbone.

Link Aggregation (FEC, GEC, and POS)

The ML-Series offers Fast EtherChannel, Gigabit EtherChannel, and POS channel link aggregation. Link aggregation groups multiple ports into a larger logical port and provides resiliency during the failure of any individual ports. The ML-Series supports a maximum of four Ethernet ports in Fast EtherChannel, two Ethernet ports in Gigabit EtherChannel, and two SONET/SDH virtual ports in POS channel. POS channel is only supported with LEX encapsulation.

Traffic flows map to individual ports based on MAC source address (SA)/destination address (DA) for bridged packets and IP SA/DA for routed packets. There is no support for policing or class-based packet priorities when link aggregation is configured.

RMON

The ML-Series card features remote monitoring (RMON) that allows network operators to monitor the health of the network with a network management system (NMS). The ML-Series card Ethernet interfaces support RMON for statistics, utilization, and history. For general information about using Cisco IOS to manage RMON, refer to the “Configuring RMON Support” chapter of the Cisco IOS Configuration Fundamentals Configuration Guide.

The MIBs supported are:

- RFC-2819—RMON MIB
- RFC-2358—Ether-Like-MIB
- RFC-2233—IF MIB

RPR

RPR is an emerging network architecture designed for metro fiber ring networks. This new MAC protocol is designed to overcome the limitations of STP, RSTP, and SONET in packet-based networks. RPR convergence times are comparable to SONET and much faster than STP or RSTP. RPR operates at the Layer 2 level and is compatible with Ethernet and protected or unprotected SONET circuits.

SNMP

Both the ONS 15454 SONET/SDH and the ML-Series cards have SNMP agents and support SNMP Version 1 (SNMPv1) and SNMP Version 2c (SNMPv2c) sets and traps. The ONS 15454 SONET/SDH accepts, validates, and forwards get/getNext/set requests to the ML-Series

through a proxy agent. The ML-Series requests contain the slot identification of the ML-Series cards to distinguish the request from a general ONS 15454 SNMP request. Responses from the ML-Series are relayed by the ONS 15454 to the requesting SNMP agents.

The ML-Series card SNMP support includes:

- Spanning Tree Protocol (STP) traps from Bridge-MIB (RFC 1493)
- Authentication traps from RFC 1157
- Link-up and link-down traps for Ethernet ports from IF-MIB (RFC 1573)
- Export of QoS statistics through the CISCO-PORT-QOS-MIB extension

**Note**

The ML-Series card CISCO-PORT-QOS-MIB extension includes support for COS-based QoS indexing. It does not support configuration objects.

- For more information on how the ONS 15454 or ONS 15454 SDH implements SNMP, refer to the “SNMP” chapter of the *Cisco ONS 15454 Troubleshooting Manual* or the *Cisco ONS 15454 SDH Troubleshooting Manual*. For more information on specific MIBs, refer to the Cisco SNMP Object Navigator at <http://www.cisco.com>.

TL1

TL1 on the ML-Series cards can be used for card inventory, fault or alarm management, card provisioning, and retrieval of status information for both data and SONET ports. TL1 can also be used to provision SONET STS circuits and transfer a Cisco IOS startup configuration file to the TCC2/TCC2P card memory. For specific TL1 commands and general TL1 information, refer to the *Cisco ONS SONET TL1 Command Guide*.

VRF Lite

VPN Routing/Forwarding Lite (VRF Lite) is an ML-Series card-specific implementation of a VPN routing/forwarding instance (VRF). Unlike standard VRF, VRF Lite does not contain Multi-Protocol internal BGP (MP-iBGP).

Standard VRF is an extension of IP routing that provides multiple routing instances and separate IP routing and forwarding tables for each VPN. VRF is used in concert with internal MP-iBGP. MP-iBGP distributes the VRF information between routers to provide Layer 3 MPLS-VPN.

VRF Lite stores VRF information locally and does not distribute the VRF information to connected equipment. VRF information directs traffic to the correct interfaces and subinterfaces when the traffic is received from customer routers or from service provider router(s).

VRF Lite allows an ML-Series card, acting as customer equipment, to have multiple interfaces and subinterfaces with service provider equipment. The customer ML-Series card can then service multiple customers. Normal customer equipment serves a single customer.



CTC Operations

This chapter covers Cisco Transport Controller (CTC) operations of the ML-Series card. All operations described in the chapter take place at the card-level view of CTC. CTC shows provisioning information and statistics for both the Ethernet and packet-over-SONET/SDH (POS) ports of the ML-Series card. For the ML-Series cards, CTC manages SONET/SDH alarms and provisions STS/STM circuits in the same manner as other ONS 15454 SONET/SDH traffic cards.

Use CTC to load a Cisco IOS configuration file or to open a Cisco IOS command-line interface (CLI) session. See [Chapter 3, “Initial Configuration.”](#)

This chapter contains the following major sections:

- [Displaying ML-Series POS And Ethernet Statistics on CTC, page 2-1](#)
- [Displaying ML-Series Ethernet Ports Provisioning Information on CTC, page 2-2](#)
- [Displaying ML-Series POS Ports Provisioning Information on CTC, page 2-4](#)
- [Provisioning Framing Mode, page 2-5](#)
- [Managing SONET/SDH Alarms, page 2-6](#)
- [Displaying Maintenance Information, page 2-7](#)
- [Provisioning SONET/SDH Circuits, page 2-8](#)
- [J1 Path Trace, page 2-8](#)

Displaying ML-Series POS And Ethernet Statistics on CTC

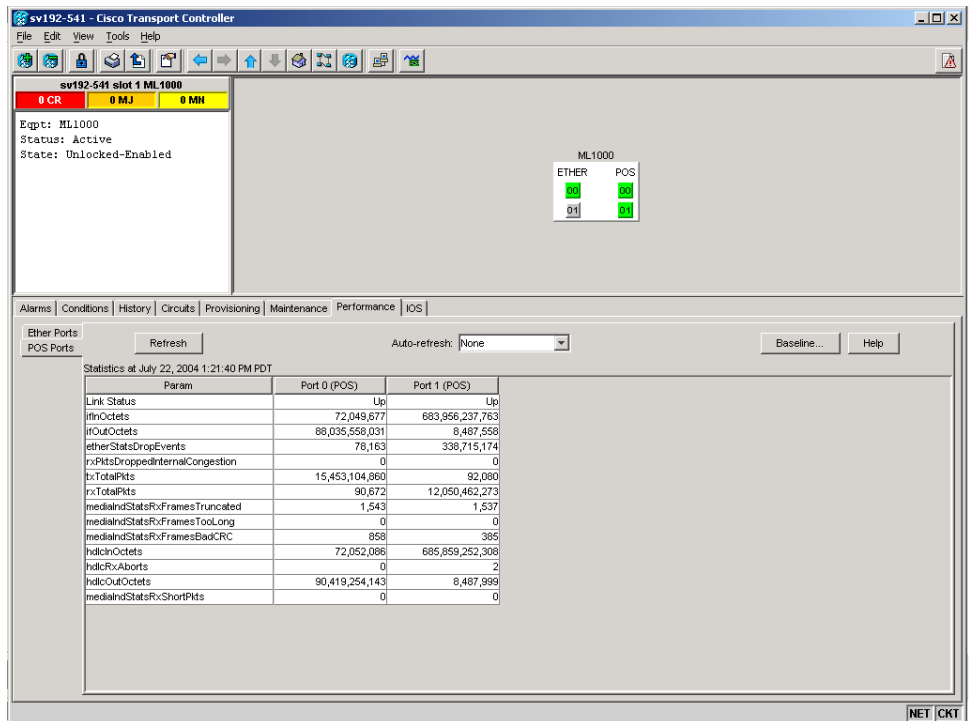
The POS statistics window lists POS port-level statistics ([Figure 2-1](#)). Display the CTC card view for the ML-Series card and click the **Performance > POS Ports** tabs to display the window.

The Ethernet statistics window lists Ethernet port-level statistics. It is similar in appearance to the POS statistics window. The ML-Series Ethernet ports are zero based. Display the CTC card view for the ML-Series card and click the **Performance > Ether Ports** tabs to display the window. [Table 2-1](#) describes the buttons in the POS Ports and Ether Ports window.

A different set of statistics appears for the ML-Series card depending on whether the card is using HDLC or GFP-F framing. For definitions of ML-Series card statistics, refer to the “Performance Monitoring” chapter of the *Cisco ONS 15454 SONET and DWDM Troubleshooting Guide* or the *Cisco ONS 15454 SDH Troubleshooting Guide*.

Table 2-1 ML-Series POS and Ethernet Statistics Fields and Buttons

Button	Description
Refresh	Manually refreshes the statistics.
Baseline	Resets the software counters (in that particular CTC client only) temporarily to zero without affecting the actual statistics on the card. From that point on, only counters displaying the change from the temporary baseline are displayed by this CTC client. These new baselined counters are shown only as long as the user displays the Performance window. If the user navigates to another CTC window and comes back to the Performance window, the true actual statistics retained by the card are shown.
Auto-Refresh	Sets a time interval for the automatic refresh of statistics.

Figure 2-1 Displaying ML-Series POS Statistics

Displaying ML-Series Ethernet Ports Provisioning Information on CTC

The Ethernet port provisioning window displays the provisioning status of the Ethernet ports (Figure 2-2). Click the **Provisioning > Ether Ports** tabs to display this window. For ML-Series cards, only the Port Name field can be provisioned from CTC. The user must configure ML-Series ports using the Cisco IOS CLI.

Figure 2-2 Displaying ML-Series Ethernet Port Provisioning Information

The screenshot shows the Cisco Transport Controller (CTC) interface for an Interop-551 slot 2 ML100T. The main window displays the status of the ML100T card and its ports. The ports are listed as follows:

Port	Port Name	Admin State	Link State	MTU	Speed	Duplex	Flow Control
0		Down	Down	1500	Auto	Auto	None
1		Down	Down	1500	Auto	Auto	None
2		Down	Down	1500	Auto	Auto	None
3		Down	Down	1500	Auto	Auto	None
4		Down	Down	1500	Auto	Auto	None
5		Down	Down	1500	Auto	Auto	None
6		Down	Down	1500	Auto	Auto	None
7		Down	Down	1500	Auto	Auto	None
8		Down	Down	1500	Auto	Auto	None
9		Down	Down	1500	Auto	Auto	None
10		Down	Down	1500	Auto	Auto	None
11		Down	Down	1500	Auto	Auto	None

The interface also shows a list of ports on the left side, including POS and ETHER ports, and a table of port configurations on the right side. The table includes columns for Port, Port Name, Admin State, Link State, MTU, Speed, Duplex, and Flow Control. The status of all ports is shown as Down.

The Provisioning > Ether Ports tab displays the following information:

- **Port**—The fixed number identifier for the specific port.
- **Port Name**—Configurable 12-character alphanumeric identifier for the port.
- **Admin State**—Configured port state, which is administratively active or inactive. Possible values are UP and DOWN.
- **Link State**—Status between signaling points at port and attached device. Possible values are UP and DOWN.
- **MTU**—(maximum transfer unit) Largest acceptable packet size configured for that port. Default value is 1500.
- **Speed**—ML1000-2 possible values are Auto or 1 Gbps. ML100T-12 possible values are Auto, 10Mbps, or 100Mbps.
- **Duplex**—Setting of the port. ML1000-2 possible values are Auto or Full. ML100T-12 possible values are Auto, Full, or Half.
- **Flow Control**—Negotiated flow control mode. ML1000-2 possible values are Asymmetrical, Symmetrical or None. ML100T-12 possible values are Symmetrical or None.
- **Optics**—Small form-factor pluggable (SFP) physical media type. Possible values are Unplugged, 1000 SX, or 1000 LX. (This information does not apply to the ML100T-12 card.)



Note

Auto indicates the port is set to autonegotiate capabilities with the attached link partner.

**Note**

The port name field configured in CTC and the port name configured in Cisco IOS are independent of each other. The name for the same port under Cisco IOS and CTC does not match, unless the same name is used to configure the port name in both CTC and Cisco IOS.

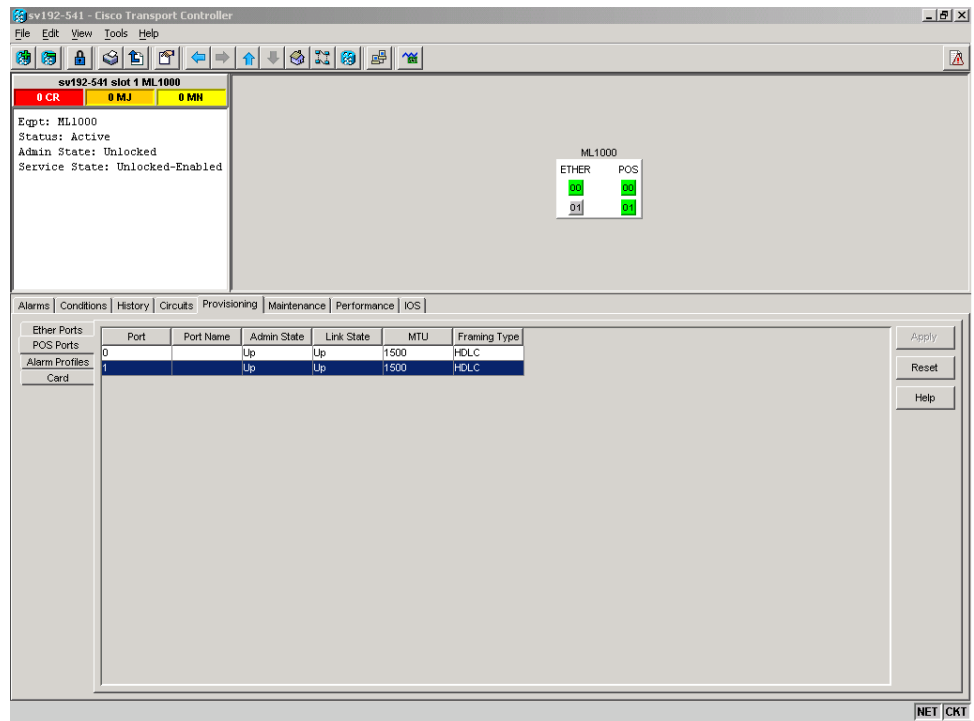
**Note**

When set to autonegotiate, the ML1000-2 might show Auto in the speed and duplex columns of the Ether ports provisioning screen. This indicates that the ML1000-2 is set to autonegotiate flow control with the link partner. It does not mean the speed or duplex mode of the card will vary from the 1-Gbps, full duplex characteristics of Gigabit Ethernet.

Displaying ML-Series POS Ports Provisioning Information on CTC

The POS ports provisioning window displays the provisioning status of the card's POS ports (Figure 2-3). Click the **Provisioning > POS Ports** tabs to display this window. For ML-Series cards, only the POS Port Name field can be provisioned from CTC. The user must configure ML-Series ports through the Cisco IOS CLI.

Figure 2-3 Displaying POS Port Provisioning Information



The **Provisioning > POS Ports** tab displays the following information:

- Port—Fixed number identifier for the specific port.
- Port Name—Configurable 12-character alphanumeric identifier for the port.

- Admin State—Configured administrative port state, which is active or inactive. Possible values are UP and DOWN. For the UP value to appear, a POS port must be both administratively active and have a SONET/SDH circuit provisioned.
- Link State—Status between signaling points at the port and an attached device. Possible values are UP and DOWN.
- MTU—The maximum transfer unit, which is the largest acceptable packet size configured for that port. The maximum setting is 9000. The default size is 1500 for the G-Series card compatible encapsulation (LEX) and 4470 for Cisco HDLC and Point-to-Point Protocol/Bridging Control Protocol (PPP/BCP) encapsulation.
- Framing Type—HDLC or frame-mapped generic framing procedure (GFP-F) framing type shows the POS framing mechanism being employed on the port.

**Note**

The port name field configured in CTC and the port name configured in Cisco IOS are independent of each other. The name for the same port under Cisco IOS and CTC does not match, unless the same name is used to configure the port name in both CTC and Cisco IOS.

Provisioning Framing Mode

The card mode provisioning window shows the framing mode employed by the ML-Series card and allows the user to change the framing mechanism to either HDLC or GFP-F. Click the **Provisioning > Card** tabs to display this window (Figure 2-4). HDLC is the default framing mode for the ONS 15454 or ONS 15454 SDH ML-Series card. For more information on framing mechanisms, see the “POS on ONS Ethernet Cards” chapter.

The user may also pre-provision the framing mode of an ML-Series card before the card is physically installed. The ML-Series card will then boot up into the pre-provisioned framing mode.

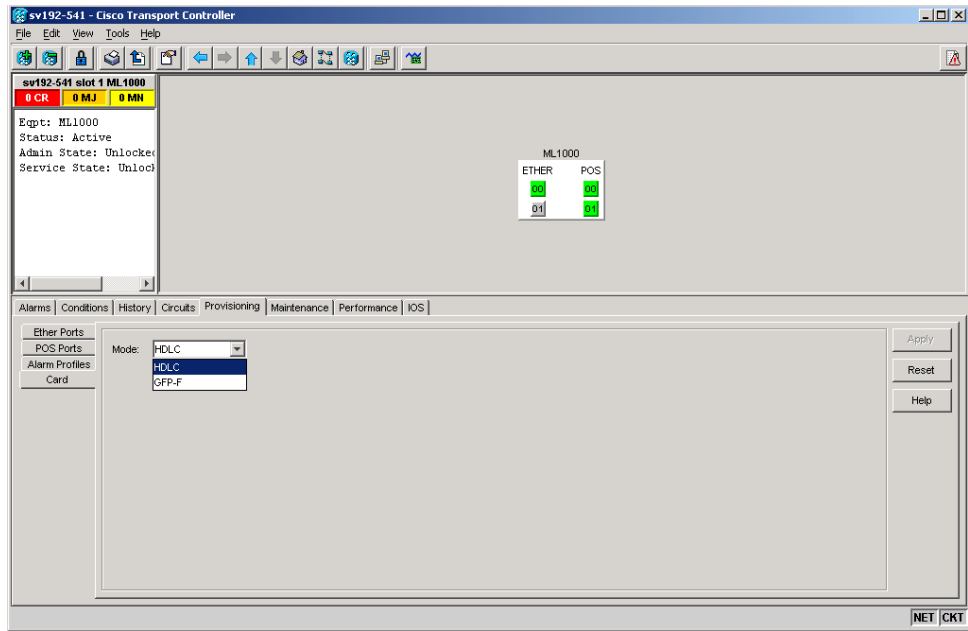
A connected POS port must match the framing mechanism of its peer port. You must delete all the existing STS/STM circuits on the ML-Series cards before changing the framing mode.

**Caution**

The ML-Series card reboots after the framing mode is changed.

Click the **Provisioning > Card** tabs to display this window. Use the **Mode** drop-down list and click **Apply** to provision the framing mode type. Click **Yes** at the Reset Card dialog box that appears.

Figure 2-4 Displaying Framing Mode

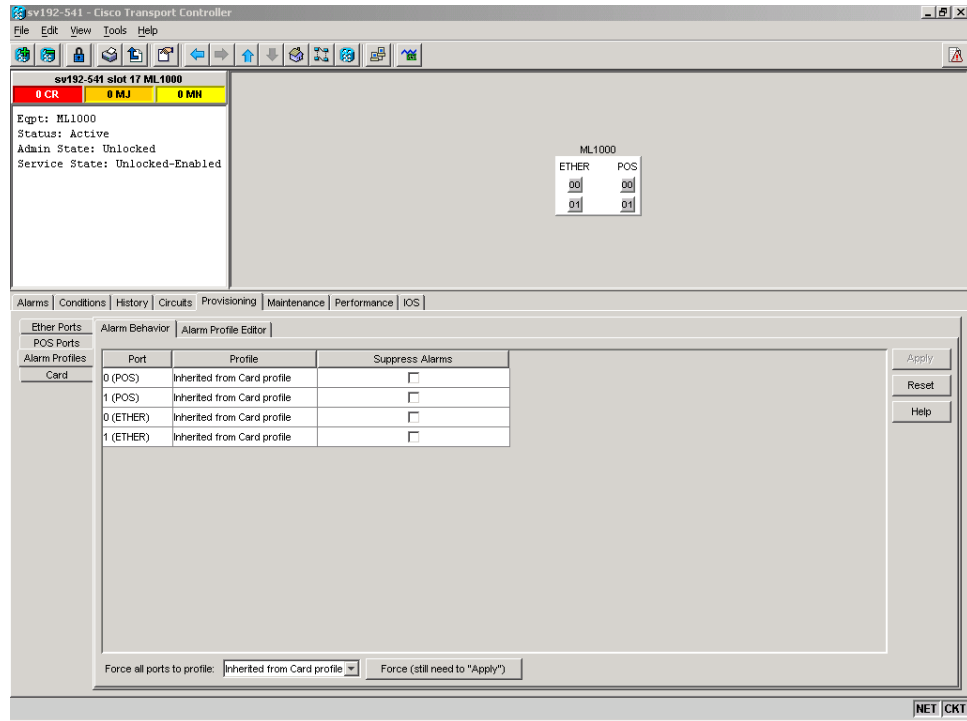


Managing SONET/SDH Alarms

CTC manages the ML-Series SONET/SDH alarm behavior in the same manner as it manages alarm behavior for other ONS 15454 SONET/SDH cards. Refer to the “Manage Alarms” chapter of the *Cisco ONS 15454 Procedure Guide* or the *Cisco ONS 15454 SDH Procedure Guide* for detailed information. For information on specific alarms, refer to the “Alarm Troubleshooting” chapter of the *Cisco ONS 15454 Troubleshooting Guide* or *Cisco ONS 15454 SDH Troubleshooting Guide* for detailed information.

To view the window, click the **Provisioning > Alarm Profiles** tabs for the Ethernet and POS port alarm profile information. [Figure 2-5](#) shows the alarm profile pane.

Figure 2-5 Managing ML-Series SONET/SDH Alarms



Displaying Maintenance Information

The maintenance information window displays the ML-Series card's field programmable gate array (FPGA) version (Figure 2-6). Click the **Maintenance > Info** tabs to display this window.

The FPGA on the ML100T-12 and ML1000-2 provides the interface and buffering between the card's network processor and the SONET/SDH cross-connect. There are two different FPGA images. FPGA Image Version 3.x supports HDLC framing, and FPGA Image Version 4.x supports GFP-F Framing. Both images support virtual concatenation (VCAT). In Release 5.0 and later, the correct FPGA is automatically loaded when the framing mode is changed by the user.



Note

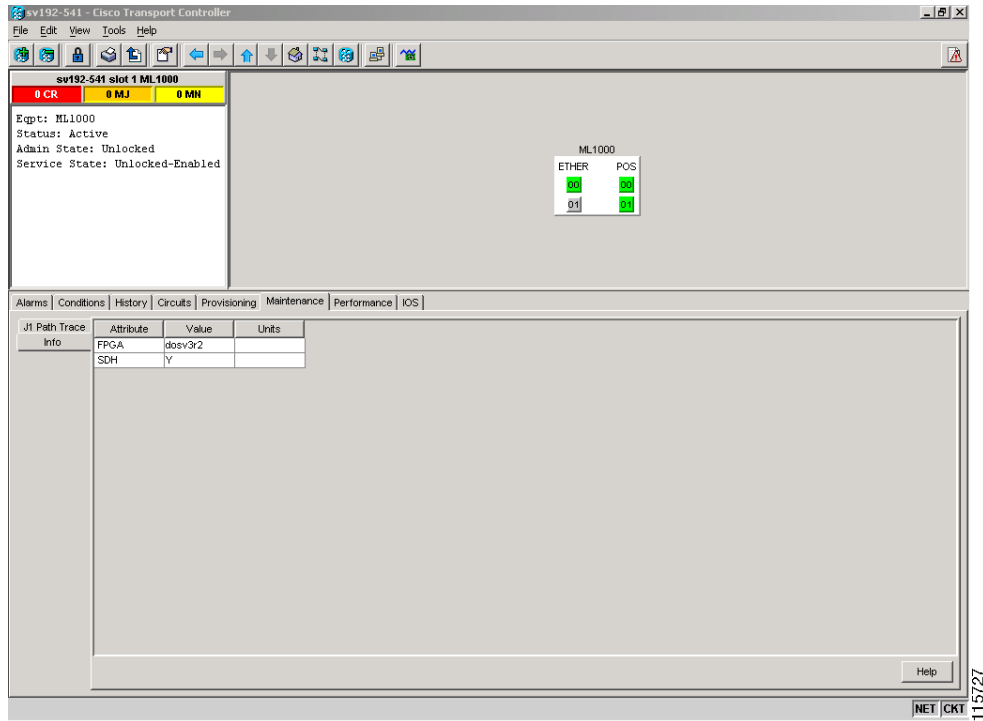
ML-Series cards manufactured prior to Software Release 4.6 need an updated version of the FPGA to support VCAT.



Caution

Do not attempt to use Release 5.0 FPGA images with an earlier CTC software release.

Figure 2-6 Displaying Maintenance Information



Provisioning SONET/SDH Circuits

CTC provisions and edits STS/STM level circuits for the two virtual SONET/SDH ports of the ML-Series card in the same manner as it provisions other ONS 15454 SONET/SDH OC-N cards. The ONS 15454 ML-Series card supports both contiguous concatenation (CCAT) and virtual concatenation (VCAT) circuits.

For step-by-step instructions to configure an ML-Series card SONET CCAT or VCAT circuit, refer to the “Create Circuits and VT Tunnels” chapter of the *Cisco ONS 15454 Procedure Guide*. For step-by-step instructions to configure an ML-Series card SDH CCAT or VCAT circuit, refer to the “Create Circuits and Tunnels” chapter of the *Cisco ONS 15454 SDH Procedure Guide*. For more general information on VCAT circuits, refer to the “Circuits and Tunnels” chapter of the *Cisco ONS 15454 Reference Manual* or the *Cisco ONS 15454 SDH Reference Manual*.

J1 Path Trace

The J1 Path Trace is a repeated, fixed-length string comprised of 64 consecutive J1 bytes. You can use the string to monitor interruptions or changes to SONET/SDH circuit traffic. For information on J1 Path Trace, refer to the *Cisco ONS 15454 Reference Manual* or the *Cisco ONS 15454 SDH Reference Manual*.



Initial Configuration

This chapter describes the initial configuration of the ML-Series card and contains the following major sections:

- [Hardware Installation, page 3-1](#)
- [Cisco IOS on the ML-Series Card, page 3-2](#)
- [Startup Configuration File, page 3-7](#)
- [Multiple Microcode Images, page 3-11](#)
- [Changing the Working Microcode Image, page 3-12](#)
- [Version Up Software Upgrade, page 3-14](#)
- [Cisco IOS Command Modes, page 3-17](#)
- [Using the Command Modes, page 3-18](#)

Hardware Installation

This section lists hardware installation tasks, including booting up the ML-Series card. Because ONS 15454 SONET/SDH card slots can be preprovisioned for an ML-Series line card, the following physical operations can be performed before or after the provisioning of the slot has taken place.

1. Install the ML-Series card into the ONS 15454 SONET/SDH. See Chapter 2, “Install Cards and Fiber-Optic Cable” of the *Cisco ONS 15454 Procedure Guide* or *Cisco ONS 15454 SDH Procedure Guide* for information.
2. Connect the cables to the front ports of the ML-Series card.
3. (Optional) Connect the console terminal to the ML-Series card.



Note

A NO-CONFIG condition is reported in CTC under the Alarms pane when an ML-Series card is inserted and no valid Cisco IOS startup configuration file exists. Loading or creating this file clears the condition. See the “[Startup Configuration File](#)” section on [page 3-7](#) for information on loading or creating the file.

Cisco IOS on the ML-Series Card

The Cisco IOS software image used by the ML-Series card is not permanently stored on the ML-Series card but in the flash memory of the TCC2/TCC2P card. During a hard reset, when a card is physically removed and reinserted or power is otherwise lost to the card, the Cisco IOS software image is downloaded from the flash memory of the TCC2/TCC2P to the memory cache of the ML-Series card. The cached image is then decompressed and initialized for use by the ML-Series card.

During a soft reset, when the ML-Series card is reset through CTC or the Cisco IOS CLI privilege level **reload** command, the ML-Series card checks its cache for a Cisco IOS image. If a valid and current Cisco IOS image exists, the ML-Series card decompresses and initializes the image. If the image does not exist, the ML-Series requests a new copy of the Cisco IOS image from the TCC2/TCC2P. Caching the Cisco IOS image provides a significant time savings when a warm reset is performed.

There are four ways to access the ML-Series card Cisco IOS configuration. The two out-of-band options are opening a Cisco IOS session on CTC and telnetting to the node IP Address and slot number plus 2000. The two-in-band signalling options are telnetting to a configured management interface and directly connecting to the console port.

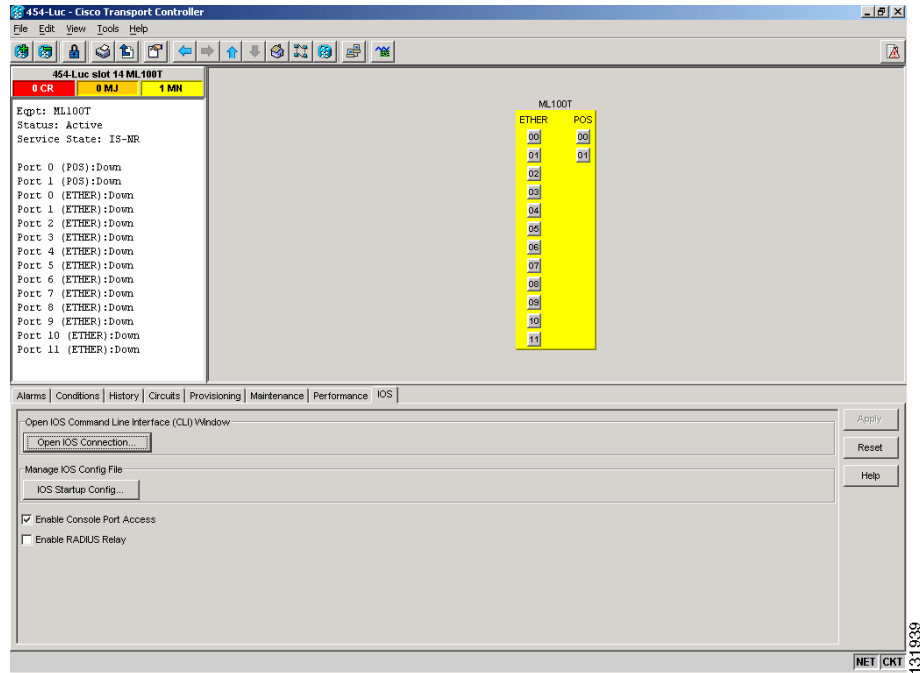
Opening a Cisco IOS Session Using CTC

Users can initiate a Cisco IOS CLI session for the ML-Series card using CTC. Click the **IOS** tab at the card-level CTC view, then click the **Open IOS Command Line Interface (CLI)** button (Figure 3-1). A window opens and a standard Cisco IOS CLI User EXEC command mode prompt appears.

**Note**

A Cisco IOS startup configuration file must be loaded and the ML-Series card must be installed and initialized prior to opening a Cisco IOS CLI session on CTC. See the [“Startup Configuration File” section on page 3-7](#) for more information.

Figure 3-1 CTC IOS Window



Telnetting to the Node IP Address and Slot Number

Users can telnet to the Cisco IOS CLI using the IP address and the slot number of the ONS 15454 SONET/SDH plus 2000.



Note

A Cisco IOS startup configuration file must be loaded and the ML-Series card must be installed and initialized prior to telnetting to the IP address and slot number plus 2000. See the [“Startup Configuration File” section on page 3-7](#) for more information.

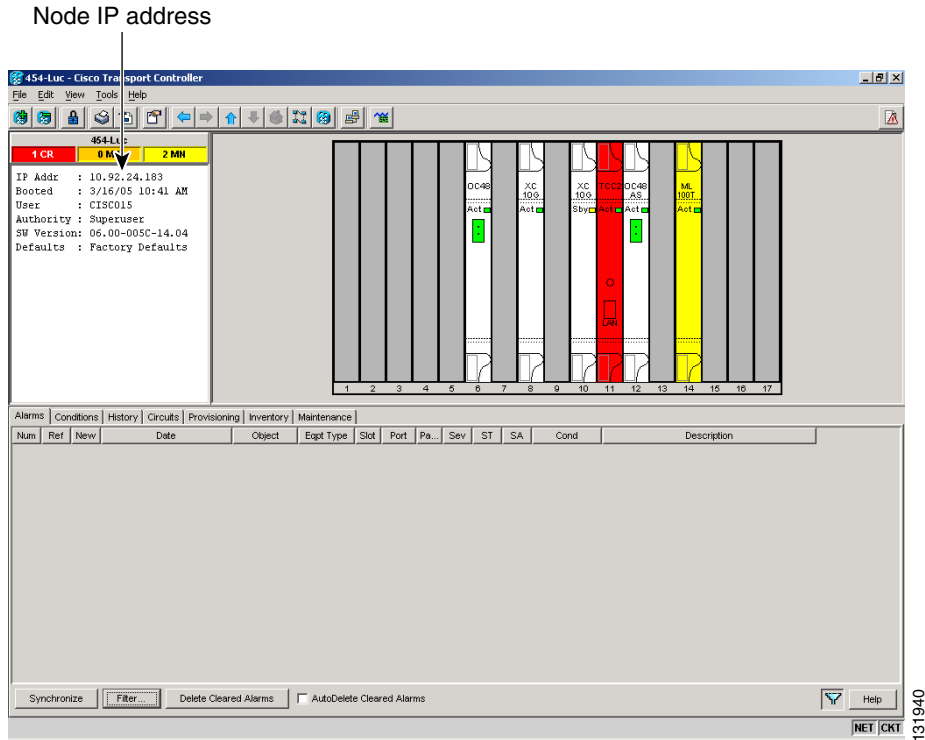


Note

If the ONS 15454 SONET/SDH node is set up as a proxy server, where one ONS 15454 SONET/SDH node in the ring acts as a gateway network element (GNE) for the other nodes in the ring, telnetting over the GNE firewall to the IP address and slot number of a non-GNE or end network element (ENE) requires the user’s Telnet client to be SOCKS v5 aware (RFC 1928). Configure the Telnet client to recognize the GNE as the Socks v5 proxy for the Telnet session and to recognize the ENE as the host.

- Step 1** Obtain the node IP address from the LCD on the front of the physical ONS 15454 SONET/SDH or the IP Addr field shown at the CTC node view (Figure 3-2).
- Step 2** Identify the slot number containing the targeted ML-Series card from either the physical ONS 15454 SONET/SDH or the CTC node view (Figure 3-2). For example, Slot 13.

Figure 3-2 CTC Node View Showing IP Address and Slot Number



- Step 3** Use the IP address and the total of the slot number plus 2000 as the Telnet address in your preferred communication program. For example, for an IP address of 10.92.18.124 and Slot 13, you would enter or telnet 10.92.18.124 2013.

Telnetting to a Management Port

Users can access the ML-Series through a standard Cisco IOS management port in the same manner as other Cisco IOS platforms. For further details about configuring ports and lines for management access, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide*.

As a security measure, the vty lines used for Telnet access are not fully configured. In order to gain Telnet access to the ML-Series card, you must configure the vty lines via the serial console connection or preload a startup-configuration file that configures the vty lines. A port on the ML-Series must first be configured as the management port; see “[Configuring the Management Port](#)” section on page 3-8.

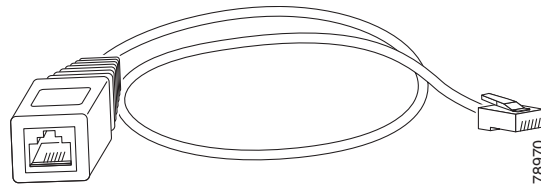
ML-Series IOS CLI Console Port

The ML-Series card has an RJ-11 serial console port on the card faceplate labeled CONSOLE. The console port is wired as data circuit-terminating equipment (DCE). It enables communication from the serial port of a PC or workstation running terminal emulation software to the Cisco IOS CLI on a specific ML-Series card.

RJ-11 to RJ-45 Console Cable Adapter

Due to space limitations on the ML-Series card faceplate, the console port is an RJ-11 modular jack instead of the more common RJ-45 modular jack. Cisco supplies an RJ-11 to RJ-45 console cable adapter (P/N 15454-CONSOLE-02) with each ML-Series card. After connecting the adapter, the console port functions like the standard Cisco RJ-45 console port. [Figure 3-3](#) shows the RJ-11 to RJ-45 console cable adapter.

Figure 3-3 Console Cable Adapter



[Table 3-1](#) shows the mapping of the RJ-11 pins to the RJ-45 pins.

Table 3-1 RJ-11 to RJ-45 Pin Mapping

RJ-11 Pin	RJ-45 Pin
1	1
2	2
3	3
4	4
None	5
5	6
None	7
6	8

Connecting a PC or Terminal to the Console Port

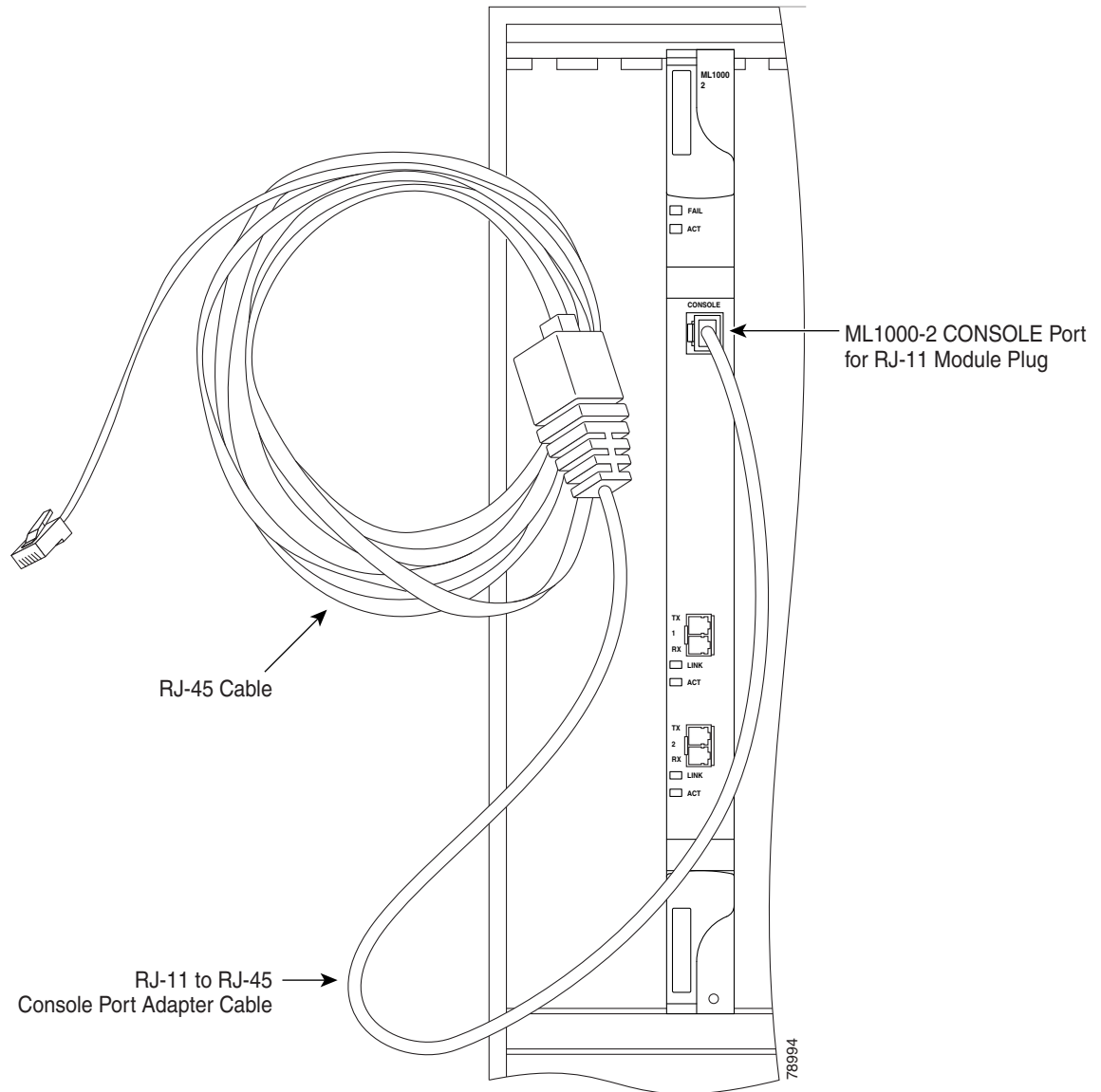
Use the supplied cable, an RJ-11 to RJ-45 console cable adapter, and a DB-9 adapter to connect a PC to the ML-Series console port.

The PC must support VT100 terminal emulation. The terminal-emulation software—frequently a PC application such as HyperTerminal or Procomm Plus—makes communication between the ML-Series and your PC or terminal possible during the setup program.

-
- Step 1** Configure the data rate and character format of the PC or terminal to match these console port default settings:
- 9600 baud
 - 8 data bits
 - 1 stop bit
 - No parity
- Step 2** Insert the RJ-45 connector of the supplied cable into the female end of the supplied console cable adapter.

- Step 3** Insert the RJ-11 modular plug end of the supplied console cable adapter into the RJ-11 serial console port, labeled CONSOLE, on the ML-Series card faceplate. [Figure 3-4](#) shows the ML1000-2 faceplate with console port. For the ML100T-12 and ML100X-8, the console port is at the bottom of the card faceplate.

Figure 3-4 Connecting to the Console Port



- Step 4** Attach the supplied RJ-45-to-DB-9 female DTE adapter to the nine-pin DB-9 serial port on the PC.
- Step 5** Insert the other end of the supplied cable in the attached adapter.

Startup Configuration File

The ML-Series card needs a startup configuration file in order to configure itself beyond the default configuration when the card is reset. If no startup configuration file exists in the TCC2/TCC2P flash memory, then the card boots up to a default configuration. Users can manually set up the startup configuration file through the serial console port and the Cisco IOS CLI configuration mode or load a Cisco IOS supplied sample startup configuration file through CTC. A running configuration becomes a startup configuration file when saved with a **copy running-config startup-config** command.

It is not possible to establish a Telnet connection to the ML-Series card until a startup configuration file is loaded onto the ML-Series card. Access is available through the console port.

**Caution**

The **copy running-config startup-config** command saves a startup configuration file to the flash memory on the ML-Series card. This operation is confirmed by the appearance of [OK] in the Cisco IOS CLI session. The startup configuration file is also saved to the ONS node's database restoration file after approximately 30 additional seconds.

**Caution**

Accessing the read-only memory monitor mode (ROMMON) on the ML-Series card without the assistance of Cisco personnel is not recommended. This mode allows actions that can render the ML-Series card inoperable. The ML-Series card ROMMON is preconfigured to boot the correct Cisco IOS software image for the ML-Series card.

**Caution**

The maximum size of the startup configuration file is 98356 bytes (characters).

**Note**

When the running configuration file is altered, a RUNCFG-SAVENEED condition appears in CTC. This condition is a reminder to enter a **copy running-config startup-config** command in the Cisco IOS CLI, or the changes will be lost when the ML-Series card reboots.

Manually Creating a Startup Configuration File Through the Serial Console Port

Configuration through the serial console port is familiar to those who have worked with other products using Cisco IOS. At the end of the configuration procedure, the **copy running-config startup-config** command saves a startup configuration file.

The serial console port gives the user visibility to the entire booting process of the ML-Series card. During initialization the ML-Series card first checks for a locally, valid cached copy of Cisco IOS. It then either downloads the Cisco IOS software image from the TCC2/TCC2P or proceeds directly to decompressing and initializing the image. Following Cisco IOS initialization the CLI prompt appears, at which time the user can enter the Cisco IOS CLI configuration mode and setup the basic ML-Series configuration.

Passwords

There are two types of passwords that you can configure for an ML-Series card: an enable password and an enable secret password. For maximum security, make the enable password different from the enable secret password.

- **Enable password**—The enable password is a non-encrypted password. It can contain any number of uppercase and lowercase alphanumeric characters. Give the enable password only to users permitted to make configuration changes to the ML-Series card.
- **Enable secret password**—The enable secret password is a secure, encrypted password. By setting an encrypted password, you can prevent unauthorized configuration changes. On systems running Cisco IOS software, you must enter the enable secret password before you can access global configuration mode.

An enable secret password can contain from 1 to 25 uppercase and lowercase alphanumeric characters. The first character cannot be a number. Spaces are valid password characters. Leading spaces are ignored; trailing spaces are recognized.

Passwords are configured in the [“Configuring the Management Port” section on page 3-8](#).

Configuring the Management Port

Because there is no separate management port on ML-Series cards, any Fast Ethernet interface (0-11 on the ML100T-12 card and 0-7 on the ML100X-8), any Gigabit Ethernet interface (0-1 on the ML1000-2 card), or any POS interface (0-1 on any ML-Series card) can be configured as a management port. For the packet over SONET (POS) interface to exist, an STS or STM circuit must first be created through CTC or TL1.

You can remotely configure the ML-Series card through the management port, but first you must configure an IP address so that the ML-Series card is reachable or load a startup configuration file. You can manually configure the management port interface from the Cisco IOS CLI via the serial console connection.

To configure Telnet for remote management access, perform the following procedure, beginning in user EXEC mode:

	Command	Purpose
Step 1	Router> enable Router#	Activates user EXEC (or enable) mode. The # prompt indicates enable mode.
Step 2	Router# configure terminal Router(config)#	Activates global configuration mode. You can abbreviate the command to confi g t . The Router(config)# prompt indicates that you are in global configuration mode.
Step 3	Router(config)# enable password <i>password</i>	Sets the enable password. See the “Passwords” section on page 3-8 .
Step 4	Router(config)# enable secret <i>password</i>	Allows you to enter an enable secret password. See the “Passwords” section on page 3-8 . A user must enter the enable secret password to gain access to global configuration mode.
Step 5	Router(config)# interface <i>type number</i> Router(config-if)#	Activates interface configuration mode on the interface.
Step 6	Router(config-if)# ip address <i>ip-address subnetmask</i>	Allows you to enter the IP address and IP subnet mask for the interface specified in Step 5.

	Command	Purpose
Step 7	Router(config-if)# no shutdown	Enables the interface.
Step 8	Router(config-if)# exit Router(config)#	Returns to global configuration mode.
Step 9	Router(config)# line vty line-number Router(config-line)#	Activates line configuration mode for virtual terminal connections. Commands entered in this mode control the operation of Telnet sessions to the ML-Series card.
Step 10	Router(config-line)# password password	Allows you to enter a password for Telnet sessions.
Step 11	Router(config-line)# end Router#	Returns to privileged EXEC mode.
Step 12	Router# copy running-config startup-config	(Optional) Saves your configuration changes to NVRAM.

After you have completed configuring remote management on the management port, you can use Telnet to remotely assign and verify configurations.

Configuring the Hostname

In addition to the system passwords and enable password, your initial configuration should include a hostname to easily identify your ML-Series card. To configure the hostname, perform the following task, beginning in enable mode:

	Command	Purpose
Step 1	Router# configure terminal Router(config)#	Activates global configuration mode.
Step 2	Router(config)# hostname name-string	Allows you to enter a system name. In this example, we set the hostname to “Router.”
Step 3	Router(config)# end Router#	Returns to privileged EXEC mode.
Step 4	Router# copy running-config startup-config	(Optional) Copies your configuration changes to NVRAM.

CTC and the Startup Configuration File

CTC allows a user to load the startup configuration file required by the ML-Series card. A Cisco-supplied sample Cisco IOS startup configuration file, named **Basic-IOS-startup-config.txt**, is available on the Cisco ONS 15454 SONET/SDH software CD. CISCO15 is the Cisco IOS CLI default line password and the enable password for this configuration. Users can also create their own startup configuration file, see the [“Manually Creating a Startup Configuration File Through the Serial Console Port”](#) section on page 3-7.

CTC can load a Cisco IOS startup configuration file into the TCC2/TCC2P card flash before the ML-Series card is physically installed in the slot. When installed, the ML-Series card downloads and applies the Cisco IOS software image and the preloaded Cisco IOS startup-configuration file. Preloading the startup configuration file allows an ML-Series card to immediately operate as a fully configured card when inserted into the ONS 15454 SONET/SDH.

If the ML-Series card is booted up prior to the loading of the Cisco IOS startup configuration file into TCC2/TCC2P card flash, then the ML-Series card must be reset to use the Cisco IOS startup configuration file or the user can issue the command **copy start run** at the Cisco IOS CLI to configure the ML-Series card to use the Cisco IOS startup configuration file.

Loading a Cisco IOS Startup Configuration File Through CTC

This procedure details the initial loading of a Cisco IOS Startup Configuration file through CTC.

-
- Step 1** At the card-level view of the ML-Series card, click the **IOS** tab.
The CTC IOS window appears (Figure 3-1 on page 3-3).
- Step 2** Click the **IOS startup config** button.
The config file dialog box appears.
- Step 3** Click the **Local -> TCC** button.
- Step 4** The sample Cisco IOS startup configuration file can be installed from either the ONS 15454 SONET/SDH software CD or from a PC or network folder:
- To install the Cisco supplied startup config file from the ONS 15454 SONET/SDH software CD, insert the CD into the CD drive of the PC or workstation. Using the CTC config file dialog, navigate to the CD drive of the PC or workstation and double-click the **Basic-IOS-startup-config.txt** file.
 - To install the Cisco supplied config file from a PC or network folder, navigate to the folder containing the desired Cisco IOS startup config file and double-click the desired Cisco IOS startup config file.
- Step 5** At the Are you sure? dialog box, click the **Yes** button.
The Directory and Filename fields on the configuration file dialog update to reflect that the Cisco IOS startup config file is loaded onto the TCC2/TCC2P.
- Step 6** Load the IOS startup config file from the TCC2/TCC2P to the ML-Series card:
- a. If the ML-Series card has already been installed, right-click on the ML-Series card at the node level or card level CTC view and select **Reset Card**.
After the reset, the ML-Series card runs under the newly loaded Cisco IOS startup config.
 - b. If the ML-Series card is not yet installed, installing the ML-Series card into the slot loads and runs the newly loaded Cisco IOS startup configuration on the ML-Series card.



Note When the Cisco IOS startup configuration file is downloaded and parsed at initialization, if there is an error in the parsing of this file, an ERROR-CONFIG alarm is reported and appears under the CTC alarms pane or in TL1. No other Cisco IOS error messages regarding the parsing of text are reported to the CTC or in TL1. An experienced Cisco IOS user can locate and troubleshoot the line in the startup configuration file that produced the parsing error by opening the Cisco IOS CLI and entering a **copy start run** command.

**Note**

A standard ONS 15454 SONET/SDH database restore reinstalls the Cisco IOS startup config file on the TCC2/TCC2P, but does not implement the Cisco IOS startup config on the ML-Series. See the “[Database Restore of the Startup Configuration File](#)” section on page 3-11 for additional information.

Database Restore of the Startup Configuration File

The ONS 15454 SONET/SDH includes a database restoration feature. Restoring the database will reconfigure a node and the installed line cards to the saved provisioning, except for the ML-Series card. The ML-Series card does not automatically restore the startup configuration file saved in the TCC2/TCC2P database.

A user can load the saved startup configuration file onto the ML-Series card in two ways. He can revert completely to the saved startup configuration and lose any additional provisioning in the unsaved running configuration, which is a restoration scheme similar to other ONS cards, or he can install the saved startup configuration file on top of the current running configuration, which is a merging restoration scheme used by many Cisco Catalyst devices.

To revert completely to the startup configuration file saved in the restored database, the user needs to reset the ML-Series card. Right-click the ML-Series card in CTC and choose **Reset** or use the Cisco IOS CLI **reload** command to reset the ML-Series card.

**Caution**

Resetting the ONS 15454 ML-Series card causes a loss of traffic and closes any Telnet sessions to the card.

To merge the saved startup configuration file with the running configuration, use the Cisco IOS CLI **copy startup-config running-config** command. This restoration scheme should only be used by experienced users with an understanding of the current running configuration and the Cisco IOS **copy** command. The **copy startup-config running-config** command will not reset the ML-Series card. The user also needs to use the Cisco IOS CLI **copy running-config startup-config** command to save the new merged running configuration to the startup configuration file.

Multiple Microcode Images

The primary packet processing and forwarding on the ML-Series card is done by the network processor, which is controlled by microcode. This microcode is a set of instructions (software) loaded into the network processor and executed at high speed. The network processor has limited microcode storage space.

Some of the ML-Series card features require significant amounts of microcode, and this additional microcode exceeds the storage capacity of the network processor. These features are added as new microcode images (separate microcode programs). The network processor can only hold one microcode image at a time, and changing the loaded microcode image requires resetting the network processor.

The user can choose from several microcode images for the ML-Series card. [Table 3-2](#) compares the features available with the different microcode images.

**Caution**


Configuring topology discovery or shortest path load balancing on an ML-Series card with the SW-RPR microcode image disables support for DRPRI.

Table 3-2 *Microcode Image Feature Comparison*

Features	Base	Enhanced	MPLS	SPR
Packet Classification	Yes	Yes	Yes	Yes
Policing and QoS	Yes	Yes	Yes	Yes
Layer 2 Bridging	Yes	Yes	Yes	Yes
IP Unicast Switching	Yes	Yes	Yes	Yes
IP Fragmentation	Yes	No	No	No
IP Multicast Switching	Yes	No	No	No
EoMPLS	No	No	Yes	No
RPR Encapsulation	Yes	Yes	Yes	Yes
DRPRI	No	Yes	No	Yes
RPR Keep Alive	No	No	Yes	Yes
RPR CRC Threshold Configuration, Detection, and Wrap	No	No	Yes	Yes
RPR Customer Ethernet FCS Preservation	No	No	Yes	Yes
RPR CRC Error Alarm Generation	No	No	Yes	Yes
RPR Shortest Path Determination and Topology Discovery	No	No	Yes	Yes
PPP/HDLC/LEX Encapsulation Support	Yes	Yes	Yes	No
Enhanced PM	No	Yes	No	Yes

Changing the Working Microcode Image

The user can change the microcode image with Cisco IOS CLI configuration and a reset of the ML-Series card. To configure a working microcode image, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# microcode { base enhanced fail system-reload spr }	<p>Configures the ML-Series card with the selected microcode image:</p> <p>base—(Default) Enables base features only. Base features include Multicast routing and IP fragmentation.</p> <p>enhanced—Enables ERMS, enhanced packet statistics, and enhanced DRPRI. Disables multicast routing and IP fragmentation.</p> <p>fail system-reload—This command and feature are specific to ML-Series card. In the event of a microcode failure, it configures the ML-Series card to save information to the flash memory and then reboot. The information is saved for use by the Cisco Technical Assistance Center (Cisco TAC).</p> <p>mpls—Enables EoMPLS. Disables IP multicast, IP fragmentation, and ERMS support.</p> <p>spr—Enables SPR encapsulation, Enhanced Packet Statistics, DRPRI, and Keepalives. Disables Multicast routing or IP Fragmentation.</p>
Step 2	Router(config)# exit	Exits global configuration mode.
Step 3	Router# copy running-config startup-config	Saves the configuration changes to Flash memory. The running configuration file configured with the new microcode image choice must be saved as a startup configuration file for the ML-Series card to reboot with the new microcode image choice.
Step 4	Router# reload	<p>Resets the ML-Series card and loads the new microcode image.</p> <p> Caution Resetting the ML-Series card causes a loss of traffic and closes any Telnet sessions to the card.</p>
Step 5	Router# show microcode	Shows the microcode image currently loaded and the microcode image that loads when the ML-Series card resets.

Version Up Software Upgrade

The ML Version Up feature allows a user to independently upgrade ML-Series cards as part of an overall software upgrade process. With this feature enabled, the user first upgrades all the cards in the node that are not ML-Series cards, then in a second pass updates the ML-Series cards. Version Up is disabled by default.

The user can initiate individual upgrades for each ML-Series card or upgrade all the ML-Series card-Series cards at the same time. In the case of redundant ML-Series cards, individual upgrades allow time to verify the proper operation of the first card before the second card is upgraded. No ML-Series cards are updated until the user specifically requests it.

The user can perform a Version Up upgrade with CTC or CTM. The Version Up feature is only supported on the ONS 15454 and ONS 15454 SDH platforms. TL-1 does not support Version Up, and you cannot enter TL-1 commands during the Version Up process.

Node and Card Behavior During Version Up

Between the upgrade of the non-ML-Series cards and the upgrade of the ML-Series cards, the node functions normally in regards to existing circuits but does not allow new provisioning or software downloads. Alarms still operate even with the ML-Series cards that are not yet upgraded.

The ML-Series card also continues to carry data traffic in the time span between the upgrade of the non-ML-Series cards and the upgrade of the ML-Series card, although this traffic drops when the ML-Series card resets to load the new software. You can Telnet to the ML-Series card and configure the card using the Cisco IOS CLI, but the new configuration only exists in the running configuration file and cannot be saved to the startup configuration file.

During the Version Up upgrade, a SwMismatch condition appears for any cards on different software, even for non-ML-Series cards awaiting their turn to reset. When the card resets and loads the new software, the condition clears. The SwMismatch condition disappears on the ML-Series cards, as they finish resetting and loading the new software. You can use the SwMismatch condition to keep track of ML-Series cards that still need upgrading. A SysBoot alarm is also raised during the upgrade. This alarm does not clear until all the ML-Series cards are upgraded.

Enabling and Completing Version Up

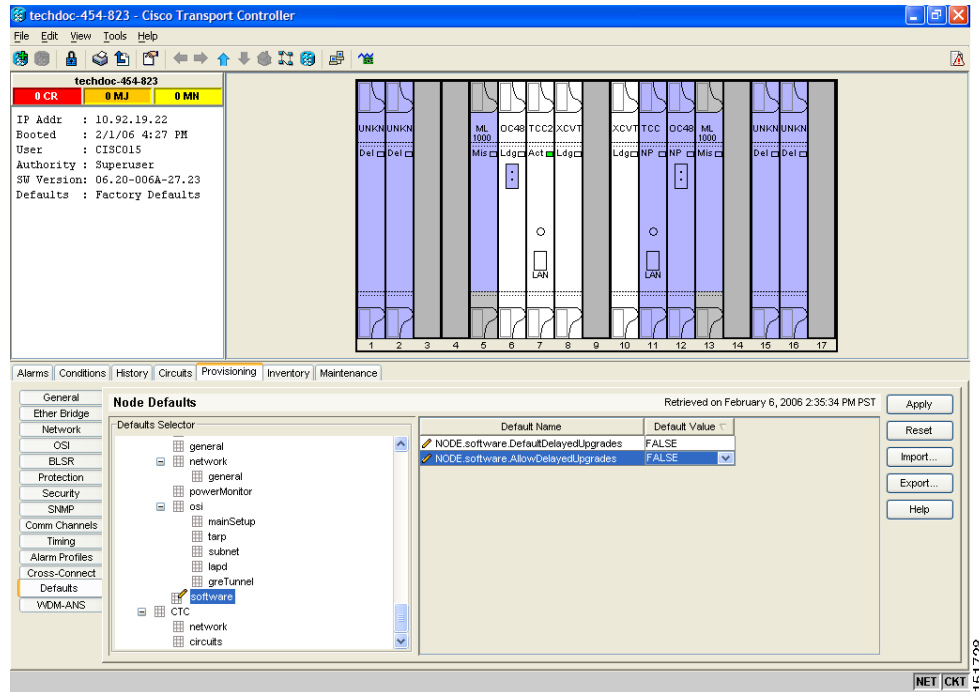
The default software upgrade behavior for the node is fully automatic. To enable Version Up, the NE-Defaults must be changed by a Superuser.

This procedure details enabling the version up feature through NE Defaults and completing the version up process.

Step 1 At the node-level view, click the **Provisioning > Defaults** tabs.

The Node Defaults appears ([Figure 3-5](#)).

Figure 3-5 Node Defaults Delayed Upgrade Settings



Step 2 Under the Defaults Selector field, click **NODE** and then click **software**.

Under the Default Name column, the *Node.Software.DefaultDelayedUpgrades* row and the *Node.Software.AllowDelayedUpgrades* row appear (Figure 3-5).

Step 3 Change the Default Value of the *Node.Software.DefaultDelayedUpgrades* row to TRUE.

Step 4 Change the Default Value of the *Node.Software.AllowDelayedUpgrades* row to TRUE.

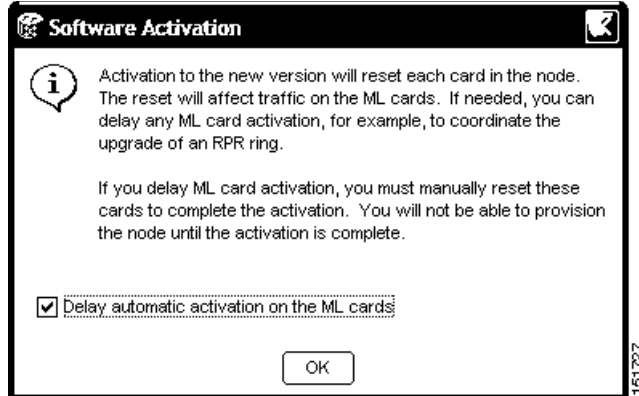
Step 5 Click the **Apply** button.

The NE Default is now set to enable Version Up.

Step 6 Begin the standard upgrade procedure for the node, detailed in *Upgrading the Cisco ONS 15454 to Release 6.2*.

After clicking the **Activate** button, the Software Activation dialog appears (Figure 3-6).

Figure 3-6 Software Activation Dialog



- Step 7** Select the “Delay automatic activation on the ML cards” checkbox and click the **OK** button to begin the Version Up activation.



Note Clearing the “Delay automatic activation on the ML cards” checkbox and clicking the OK button begins normal activation and upgrades all the cards in the node, including the ML-Series cards.

- Step 8** After the new software load is activated on the node and all the non-ML-Series cards, you can activate this load on the ML-Series cards by resetting the ML-Series cards.



Caution Resetting the ML-Series card causes a loss of traffic and closes any Telnet sessions to the card.

To reset the ML-Series card through CTC, go to the node level view and right click on the ML-Series card to reveal a drop down menu and click **Reset Card**.

- Step 9** After the ML-Series card reloads, verify the correct software build is on the card using the Cisco IOS CLI privilege level **show version** command.

[Example 3-1](#) shows a partial example of the **show version** command output with the Cisco IOS software in bold.

Example 3-1 Output from show version Command

```
ML_Series# show version
```

```
Cisco IOS Software, ONS M-Series Software (DAYTONA-I7K91-M), Experimental Version
12.2(20050912:041138) [BLD-IOS_MARINER_MARINER_BF5_BUILD_6.amoayed1 105]
```

Cisco IOS Command Modes

The Cisco IOS user interface has several different modes. The commands available to you depend on which mode you are in. To get a list of the commands available in a given mode, type a question mark (?) at the system prompt.

Table 3-3 describes the most commonly used modes, how to enter the modes, and the resulting system prompts. The system prompt helps you identify which mode you are in and, therefore, which commands are available to you.



Note

When a process makes unusually heavy demands on the CPU of the ML-Series card, it may impair CPU response time and cause a CPUHOG error message to appear on the console. This message indicates which process used a large number of CPU cycles, such as the updating of the routing table with a large number of routes due to an event. Seeing this message as a result of card reset or other infrequent events should not be a cause for concern.

Table 3-3 Cisco IOS Command Modes

Mode	What You Use It For	How to Access	Prompt
User EXEC	Connect to remote devices, change terminal settings on a temporary basis, perform basic tests, and display system information.	Log in.	Router>
Privileged EXEC (also called Enable mode)	Set operating parameters. The privileged command set includes the commands in user EXEC mode, as well as the configure command. Use this command mode to access the other command modes.	From user EXEC mode, enter the enable command and the enable password.	Router#
Global configuration	Configure features that affect the system as a whole.	From privileged EXEC mode, enter the configure terminal command.	Router(config)#

Table 3-3 Cisco IOS Command Modes (continued)

Mode	What You Use It For	How to Access	Prompt
Interface configuration	Enable features for a particular interface. Interface commands enable or modify the operation of a Fast Ethernet, Gigabit Ethernet or POS port.	From global configuration mode, enter the interface <i>type number</i> command. For example, enter interface fastethernet 0 for Fast Ethernet or interface gigabitethernet 0 for Gigabit Ethernet interfaces or interface pos 0 for Packet over SONET interfaces.	Router(config-if)#
Line configuration	Configure the console port or vty line from the directly connected console or the virtual terminal used with Telnet.	From global configuration mode, enter the line console 0 command to configure the console port or the line vty line-number command to configure a vty line.	Router(config-line)#

When you start a session on the ML-Series card, you begin in user EXEC mode. Only a small subset of the commands are available in user EXEC mode. To have access to all commands, you must enter privileged EXEC mode, also called Enable mode. From privileged EXEC mode, you can type in any EXEC command or access global configuration mode. Most of the EXEC commands are single-use commands, such as **show** commands, which show the current configuration status, and **clear** commands, which clear counters or interfaces. The EXEC commands are not saved across reboots of the ML-Series card.

The configuration modes allow you to make changes to the running configuration. If you later save the configuration, these commands are stored across ML-Series card reboots. You must start in global configuration mode. From global configuration mode, you can enter interface configuration mode, subinterface configuration mode, and a variety of protocol-specific modes.

ROMMON mode is a separate mode used when the ML-Series card cannot boot properly. For example, your ML-Series card might enter ROM monitor mode if it does not find a valid system image when it is booting, or if its configuration file is corrupted at startup.

Using the Command Modes

The Cisco IOS command interpreter, called the EXEC, interprets and executes the commands you enter. You can abbreviate commands and keywords by entering just enough characters to make the command unique from other commands. For example, you can abbreviate the **show** command to **sh** and the **configure terminal** command to **config t**.

Exit

When you type **exit**, the ML-Series card backs out one level. In general, typing **exit** returns you to global configuration mode. Enter **end** to exit configuration mode completely and return to privileged EXEC mode.

Getting Help

In any command mode, you can get a list of available commands by entering a question mark (?).

```
Router> ?
```

To obtain a list of commands that begin with a particular character sequence, type in those characters followed immediately by the question mark (?). Do not include a space. This form of help is called word help, because it completes a word for you.

```
Router# co?  
configure
```

To list keywords or arguments, enter a question mark in place of a keyword or argument. Include a space before the question mark. This form of help is called command syntax help, because it reminds you which keywords or arguments are applicable based on the command, keywords, and arguments you have already entered.

```
Router#configure ?  
memory          Configure from NV memory  
network         Configure from a TFTP network host  
overwrite-network Overwrite NV memory from TFTP network host  
terminal        Configure from the terminal  
<cr>
```

To redisplay a command you previously entered, press the Up Arrow key. You can continue to press the Up Arrow key to see more of the previously issued commands.



Tip

If you are having trouble entering a command, check the system prompt, and enter the question mark (?) for a list of available commands. You might be in the wrong command mode or using incorrect syntax.

You can press **Ctrl-Z** or type **end** in any mode to immediately return to privileged EXEC (enable) mode, instead of entering **exit**, which returns you to the previous mode.



Configuring Interfaces

This chapter describes basic interface configuration for the ML-Series card to help you get your ML-Series card up and running. Advanced POS interface configuration is covered in [Chapter 5, “Configuring POS.”](#) For more information about the Cisco IOS commands used in this chapter, refer to the *Cisco IOS Command Reference* publication.

This chapter contains the following major sections:

- [General Interface Guidelines, page 4-1](#)
- [Basic Interface Configuration, page 4-3](#)
- [Basic Fast Ethernet, Gigabit Ethernet, and POS Interface Configuration, page 4-4](#)
- [CRC Threshold Configuration, page 4-10](#)
- [Monitoring Operations on the Fast Ethernet and Gigabit Ethernet Interfaces, page 4-11](#)



Note

Complete the initial configuration of your ML-Series card before proceeding with configuring interfaces.

General Interface Guidelines

The main function of the ML-Series card is to relay packets from one data link to another. Consequently, you must configure the characteristics of the interfaces, which receive and send packets. Interface characteristics include, but are not limited to, IP address, address of the port, data encapsulation method, and media type.

Many features are enabled on a per-interface basis. Interface configuration mode contains commands that modify the interface operation (for example, of an Ethernet port). When you enter the **interface** command, you must specify the interface type and number.

The following general guidelines apply to all physical and virtual interface configuration processes:

- All interfaces have a name that is composed of an interface type (word) and a Port ID (number). For example, FastEthernet 2.
- Configure each interface with a bridge-group or IP address and IP subnet mask.
- VLANs are supported through the use of subinterfaces. The subinterface is a logical interface configured separately from the associated physical interface.
- Each physical interface, including the internal packet-over-SONET/SDH (POS) interfaces, has an assigned MAC address.

MAC Addresses

Every port or device that connects to an Ethernet network needs a MAC address. Other devices in the network use MAC addresses to locate specific ports in the network and to create and update routing tables and data structures.

To find MAC addresses for a device, use the **show interfaces** command, as follows:

```
Router# sh interfaces fastEthernet 0
FastEthernet0 is up, line protocol is up
  Hardware is epif_port, address is 0005.9a39.6634 (bia 0005.9a39.6634)
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, Auto Speed, 100BaseTX
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:01, output 00:00:18, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue :0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    11 packets input, 704 bytes
      Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
      0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
      0 watchdog, 11 multicast
    0 input packets with dribble condition detected
    3 packets output, 1056 bytes, 0 underruns
      0 output errors, 0 collisions, 0 interface resets
      0 babbles, 0 late collision, 0 deferred
      0 lost carrier, 0 no carrier
      0 output buffer failures, 0 output buffers swapped out
```

Interface Port ID

The interface port ID designates the physical location of the interface within the ML-Series card. It is the name that you use to identify the interface you are configuring. The system software uses interface port IDs to control activity within the ML-Series card and to display status information. Interface port IDs are not used by other devices in the network; they are specific to the individual ML-Series card and its internal components and software.

The ML100T-12 port IDs for the 12 Fast Ethernet interfaces are Fast Ethernet 0 through 11. The ML1000-2 port IDs for the two Gigabit Ethernet interfaces are Gigabit Ethernet 0 and 1. Both ML-Series cards feature two POS ports, and the ML-Series port IDs for the two POS interfaces are POS 0 and 1. You can use user-defined abbreviations such as f0 through f11 to configure the 12 Fast Ethernet interfaces, gi0 or gi1 to configure the two Gigabit Ethernet interfaces, and POS0 and POS1 to configure the two POS ports.

You can use Cisco IOS **show** commands to display information about any or all the interfaces of the ML-Series card.

Basic Interface Configuration

The following general configuration instructions apply to all interfaces. Before you configure interfaces, develop a plan for a bridge or routed network.

To configure an interface, do the following:

- Step 1** Enter the **configure EXEC** command at the privileged EXEC prompt to enter global configuration mode.

```
Router> enable
Password:
Router# configure terminal
Router(config)#
```

- Step 2** Enter the **interface** command, followed by the interface type (for example, fastethernet, gigabitethernet, or pos), and its interface port ID (see the “[Interface Port ID](#)” section on page 4-2).

For example, to configure a Gigabit Ethernet port, enter this command:

```
Router(config)# interface gigabitethernet number
```

- Step 3** Follow each **interface** command with the interface configuration commands required for your particular interface.

The commands you enter define the protocols and applications that will run on the interface. The ML-Series card collects and applies commands to the **interface** command until you enter another **interface** command or a command that is not an interface configuration command. You can also enter **end** to return to privileged EXEC mode.

- Step 4** Check the status of the configured interface by entering the EXEC **show interface** command.

```
Router# sh interface fastEthernet 0
FastEthernet0 is up, line protocol is up
Hardware is epif_port, address is 0005.9a39.6634 (bia 0005.9a39.6634)
MTU 1500 bytes, BW 100000 Bit, DLY 100 use,
    reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, loopback not set
Keepalive set (10 sec)
Full-duplex, Auto Speed, 100BaseTX
ARP type: ARPA, ARP Timeout 04:00:00
Last input 00:00:01, output 00:00:18, output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
Output queue :0/40 (size/max)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
    11 packets input, 704 bytes
    Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
    0 watchdog, 11 multicast
    0 input packets with dribble condition detected
    3 packets output, 1056 bytes, 0 underruns
    0 output errors, 0 collisions, 0 interface resets
    0 babbles, 0 late collision, 0 deferred
    0 lost carrier, 0 no carrier
    0 output buffer failures, 0 output buffers swapped out
```

Basic Fast Ethernet, Gigabit Ethernet, and POS Interface Configuration

ML-Series cards support Fast Ethernet, Gigabit Ethernet, and POS interfaces. This section provides some examples of configurations for all interface types.

To configure an IP address or bridge-group number on a Fast Ethernet, Gigabit Ethernet, or POS interface, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# interface <i>type number</i>	Activates interface configuration mode to configure either the Gigabit Ethernet interface, the Fast Ethernet interface, or the POS interface.
Step 2	Router(config-if)# { ip address <i>ip-address subnet-mask</i> bridge-group <i>bridge-group-number</i> }	Sets the IP address and IP subnet mask to be assigned to the interface. or Assigns a network interface to a bridge group.
Step 3	Router(config-if)# no shutdown	Enables the interface by preventing it from shutting down.
Step 4	Router(config)# end	Returns to privileged EXEC mode.
Step 5	Router# copy running-config startup-config	(Optional) Saves configuration changes to timing and control card (TCC2/TCC2P) flash database.

Configuring the Fast Ethernet Interfaces (ML100T-12)

To configure the IP address or bridge-group number, autonegotiation, and flow control on a Fast Ethernet interface, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# interface fastethernet <i>number</i>	Activates interface configuration mode to configure the Fast Ethernet interface.
Step 2	Router(config-if)# { ip address <i>ip-address subnet-mask</i> bridge-group <i>bridge-group-number</i> }	Sets the IP address and IP subnet mask to be assigned to the interface. or Assigns a network interface to a bridge group.
Step 3	Router(config-if)# [no] speed { 10 100 auto }	Configures the transmission speed for 10 or 100 Mbps. If you set the speed or duplex for auto , you enable autonegotiation on the system—the ML-Series card matches the speed and duplex mode of the partner node.
Step 4	Router(config-if)# [no] duplex { full half auto }	Sets full duplex, half duplex, or autonegotiate mode.

	Command	Purpose
Step 5	Router(config-if)# flowcontrol send {on off desired}	(Optional) Sets the send flow control value for an interface. Flow control works only with port-level policing. ML-Series card Fast Ethernet port flow control is IEEE 802.3x compliant.
Step 6	Router(config-if)# no shutdown	Enables the interface by preventing it from shutting down.
Step 7	Router(config)# end	Returns to privileged EXEC mode.
Step 8	Router# copy running-config startup-config	(Optional) Saves your configuration changes to TCC2/TCC2P flash database.

Example 4-1 shows how to do the initial configuration of a Fast Ethernet interface with an IP address, autonegotiated speed, and autonegotiated duplex.

Example 4-1 Initial Configuration of a Fast Ethernet Interface

```
Router(config)# interface fastethernet 1
Router(config-if)# ip address 10.1.2.4 255.0.0.0
Router(config-if)# speed auto
Router(config-if)# duplex auto
Router(config-if)# no shutdown
Router(config-if)# end
Router# copy running-config startup-config
```

Configuring the Gigabit Ethernet Interface (ML1000-2)

To configure IP address or bridge-group number, autonegotiation, and flow control on a Gigabit Ethernet interface, perform the following procedure, beginning in global configuration mode:



Note

The default setting for the negotiation mode is **auto** for the Gigabit Ethernet and Fast Ethernet interfaces. The Gigabit Ethernet port always operates at 1000 Mbps in full-duplex mode.

	Command	Purpose
Step 1	Router (config)# interface gigabitethernet number	Activates interface configuration mode to configure the Gigabit Ethernet interface.
Step 2	Router(config-if)# { ip address ip-address subnet-mask bridge-group bridge-group-number}	Sets the IP address and subnet mask. or Assigns a network interface to a bridge group.
Step 3	Router(config-if)# [no] negotiation auto	Sets negotiation mode to auto . The Gigabit Ethernet port attempts to negotiate the link with the partner port. If you want the port to force the link up no matter what the partner port setting is, set the Gigabit Ethernet interface to no negotiation auto .

	Command	Purpose
Step 4	Router(config-if)# flowcontrol {send receive} {on off desired}	(Optional) Sets the send or receive flow control value for an interface. Flow control works only with port-level policing. ML-Series card Gigabit Ethernet port flow control is IEEE 802.3z compliant.
Step 5	Router(config-if)# no shutdown	Enables the interface by preventing it from shutting down.
Step 6	Router(config)# end	Returns to privileged EXEC mode.
Step 7	Router# copy running-config startup-config	(Optional) Saves configuration changes to TCC2/TCC2P flash database.

Example 4-2 shows how to do an initial configuration of a Gigabit Ethernet interface with autonegotiation and an IP address.

Example 4-2 Initial Configuration of a Gigabit Ethernet Interface

```
Router(config)# interface gigabitethernet 0
Router(config-if)# ip address 10.1.2.3 255.0.0.0
Router(config-if)# negotiation auto
Router(config-if)# no shutdown
Router(config-if)# end
Router# copy running-config startup-config
```

Configuring Gigabit Ethernet Remote Failure Indication (RFI)

Remote Failure Indication (RFI) is part of the IEEE 802.3z standard and is sent to exchange failure information as part of link negotiation. This feature improves communication between non-Cisco equipment and the ML1000-2. RFI is not on by default but can be turned on by the user. Disabling RFI is sometimes necessary when a non-Cisco piece of equipment does not support the IEEE 802.3z standard implementation of RFI.

RFI on the ML-Series card supports bi-directional RFI. When there is a local fault on the ML-Series card, the ML-Series card will raise a local CARLOSS alarm and send its link partner an RFI. If an ML-Series card receives an RFI from its link partner, it raises the AUTONEG-RFI alarm and shuts down the Gigabit Ethernet port.

To enable RFI on a Gigabit Ethernet interface, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router (config)# interface gigabitethernet number	Activates interface configuration mode to configure the Gigabit Ethernet interface.
Step 2	Router(config-if)# [no] rfi auto	Enables IEEE 802.3z standard RFI. The no form of the command disables RFI.
Step 3	Router(config)# end	Returns to privileged EXEC mode.
Step 4	Router# copy running-config startup-config	(Optional) Saves configuration changes to TCC2/TCC2P flash database.

Example 4-2 shows how to do an initial configuration of RFI on a Gigabit Ethernet interface.

Example 4-3 RFI Configuration of a Gigabit Ethernet Interface

```
Router(config)# interface gigabitethernet 0
Router(config-if)# rfi auto
Router(config-if)# end
Router# copy running-config startup-config
```

Monitoring and Verifying Gigabit Ethernet Remote Failure Indication (RFI)

After RFI is configured, you can verify that RFI is enabled by using the global command **show running configuration**. Example 4-4 shows the output from this command, and the “rfi auto” line under each of the Gigabit Ethernet port’s output signifies RFI is enabled on these ports.

More specific RFI information is revealed with the global **show controller gigabit ethernet [0 | 1]** command:

- **Example 4-5** shows the full output from this command on a near-end ML-Series card when no faults are detected at the near-end or far-end. The Remote Fault Indication is 00 or no error, and the Local Fault Indication is 00 or no error.
- **Example 4-6** shows the partial output from this command on a near-end ML-Series card when a fault is detected at the near-end. The Remote Fault Indication is 00 or no error, but the Local Fault Indication is 01 or link error.
- **Example 4-7** shows the partial output from this command on a far-end ML-Series card when a fault is detected at the near-end. The Remote Fault Indication is 01 or link error, and the Local Fault Indication is 00 or no error.



Note

If the far-end link partner resets within approximately two minutes of the near-end ML-Series card sending an RFI signalling link error, the link partner will not display the RFI link error indication when back up.

Example 4-4 show run Command Output for RFI

```
Router# show running configuration
Building configuration...

Current configuration : 806 bytes
!
! No configuration change since last restart
!
version 12.2
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Interop-261-TOP-
!
boot-start-marker
boot-end-marker
!
enable password lab
!
clock timezone PST -8
```

```

clock summer-time PDT date Apr 2 2006 2:00 Oct 29 2006 2:00
ip subnet-zero
!
no mpls traffic-eng auto-bw timers frequency 0

interface GigabitEthernet0
no ip address
rfi auto
!
interface GigabitEthernet1
no ip address
rfi auto

```

Example 4-5 show controller Command Output for RFI on near-end card with no faults detected

```

Near_End# show controller gigabit ethernet 0
IF Name: GigabitEthernet0
Port Status UP
Port rxLosState Signal present
Remote Fault Indication 00 (no error)
Local Fault Indication 00 (no error)
Port 0 Gmac Loopback false
SFP EEPROM information
-----
0x0 : 03 04 07 00 00 00 02 12 00 01 01 01 0C 00 0A 64
0x10: 37 37 00 00 46 49 4E 49 53 41 52 20 43 4F 52 50
0x20: 2E 20 20 20 00 00 FFFFFFF90 65 46 54 52 4A 2D 31 33 31
0x30: 39 2D 37 44 2D 43 53 43 00 00 00 00 05 1E 00 00

GBIC Type: GBIC_1000BASE_LH
Send Flow Control: Enabled (Port level policing required to send pause frames)
Receive Flow Control : Enabled
CRC-ALARM: FALSE

MAC registers:
GCR: 0x0          CMCR : 0x00000803 (Tx Enabled, Rx Enabled)

MII registers of External GMAC:
Control Register          (0x00): 0x1140 (Auto negotiation Enabled)
Status Register          (0x01): 0x16D (Link Status Up)
Auto Neg. Advt. Register (0x04): 0x1A0 (Dir 1, Sym 1)
Auto Neg. Partner Ability Reg (0x05): 0x41A0 (Dir 1, Sym 1)
TR_IPG_TIME Register     (0x10): 0x7
PAUSE_TIME Register      (0x11): 0x100
PAUSE_SA1 Register       (0x13): 0x0
PAUSE_SA2 Register       (0x14): 0x0
PAUSE_SA3 Register       (0x15): 0x0
Pause Upper Threshold Reg. (0x19): 0x80
Pause Lower Threshold Reg. (0x1A): 0xFF
TX Full Threshold Register (0x1B): 0x40
Memory Address Register   (0x1C): 0xF008
Sync Status Register      (0x1D): 0x40
Sys Status Register       (0x1E): 0x98
Sys Control Register      (0x1F): 0x14
Auto Neg Ctrl Register    (0xF004): 0x7
Rx Uinfo Registerter-GMAC (0xF006): 0x0
RX control Register-GMAC  (0xF009): 0x3
RX Oversize Register-GMAC (0xF00A): 0x5F4
Statistics control register (0xF008): 0x1

Counters :
MAC receive conters:
Bytes          1952660

```



```

pkt64 0
pkts64to127 0
pkts128to255 0
pkts256to511 5485
pkts512to1023 0
pkts1024to1518 0
pkts1519to1530 0
pkts_good_giants 0
pkts_error_giants 0
pkts_good_runts 0
pkts_error_runts 0
pkts_ucast 0
pkts_mcast 5485
pkts_bcast 0
Rx Sync Loss 0
Overruns 0
FCS_errors 0
GMAC drop count 0
Symbol error 0
Rx Pause frames 0

```

MAC Transmit Counters

```

5d00h: %LINK-3-UPDOWN: Interface GigabitEthernet0, changed state to down
5d00h: %ETHERCHAN-5-MEMREMOVED: GigabitEthernet0 taken out of port-channel1
5d00h: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0, changed
  staBytes 1952660
pkt64 0
pkts65to127 0
pkts128to255 0
pkts256to511 5485
pkts512to1023 0
pkts1024to1518 0
pkts1519to1530 0
Good Giants 0
Unicast packets 0
Multicast packets 5485
Broadcast packets 0
FCS errors 0
Tx Pause frames 0
Ucode drops 0

```

Example 4-6 show controller Command Output for RFI on Near-end Card with Near-end Fault

```

Near_End# show controller gigabit ethernet 0
IF Name: GigabitEthernet0
Port Status DOWN
Port rxLosState No signal
Remote Fault Indication 00 (no error)
Local Fault Indication 01 (link error)
Port 0 Gmac Loopback false

```

Example 4-7 show controller Command Output for RFI on Far-end Card with Near-end Fault

```

Far_End# show controller gigabit ethernet 0
IF Name: GigabitEthernet0
Port Status DOWN
Port rxLosState Signal present
Remote Fault Indication 01 (link error)
Local Fault Indication 00 (no error)
Port 0 Gmac Loopback false

```

Configuring the POS Interfaces (ML100T-12 and ML1000-2)

Encapsulation changes on POS ports are allowed only when the interface is in a manual shutdown (ADMIN_DOWN). For advanced POS interface configuration, see [Chapter 5, “Configuring POS.”](#)

To configure the IP address, bridge group, or encapsulation for the POS interface, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# interface pos number	Activates interface configuration mode to configure the POS interface.
Step 2	Router(config-if)# { ip address ip-address subnet-mask bridge-group bridge-group-number }	Sets the IP address and subnet mask. or Assigns a network interface to a bridge group.
Step 3	Router(config-if)# shutdown	Manually shuts down the interface. Encapsulation changes on POS ports are allowed only when the interface is shut down (ADMIN_DOWN).
Step 4	Router(config-if)# encapsulation type	Sets the encapsulation type. Valid values are: <ul style="list-style-type: none"> • hdlc—Cisco HDLC • lex—(Default) LAN extension, special encapsulation for use with Cisco ONS Ethernet line cards • ppp—Point-to-Point Protocol
Step 5	Router(config-if)# no shutdown	Restarts the shutdown interface.
Step 6	Router(config)# end	Returns to privileged EXEC mode.
Step 7	Router# copy running-config startup-config	(Optional) Saves configuration changes to NVRAM.

CRC Threshold Configuration

You can configure a span shutdown when the ML-Series card receives CRC errors at a rate that exceeds the configured threshold and configured soak time. ML cards support CRC threshold configuration functionality on FE / GE / POS.

To enable and configure the triggers for CRC errors on POS, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	<pre>Router(config)#int pos0 Router(config-if)#trigger crc-error threshold <threshold_value></pre>	Sets the CRC threshold value. If the percentage of CRC errored frames received on this interface is greater than this value, we consider this interface as seeing excessive CRC. The valid values are 2, 3 and 4 indicating thresholds of 10e-2 (1%), 10e-3(0.1%) and 10e-4(.01%). The default value is 3.
Step 2	<pre>Router(config-if)#no trigger crc-error threshold < threshold_value></pre>	Sets the threshold value back to the default value 3.
Step 3	<pre>Router(config)#int pos0 Router(config-if)#trigger crc-error delay <soak_time_in_minutes></pre>	Sets the number of consecutive minutes for which excessive CRC errors should be seen to raise an excessive CRC indication. The valid values are from 3 minutes to 10 minutes. Default is 10minutes.
Step 4	<pre>Router(config-if)#no trigger crc-error delay <soak_time_in_minutes></pre>	Sets the soak value back to the default of 10 minutes.
Step 5	<pre>Router(config)#int pos0 Router(config-if)#trigger crc-error action</pre>	Enable trigger action. This configuration will bring the interface down on seeing CRC errors greater than configured <threshold value> for soak time period.
Step 6	<pre>Router(config-if)#no trigger crc-error action</pre>	Disables trigger action.

Monitoring Operations on the Fast Ethernet and Gigabit Ethernet Interfaces

To verify the settings after you have configured the interfaces, enter the **show interface** command. For additional information on monitoring the operations on POS interfaces, see the “[Configuring POS](#)” chapter.

Example 4-8 shows the output from the **show interface** command, which displays the status of the interface including port speed and duplex operation.

Example 4-8 show interface Command Output

```
Router# show interface fastEthernet 0
FastEthernet1 is administratively down, line protocol is down
Hardware is epif_port, address is 000d.bd5c.4c85 (bia 000d.bd5c.4c85)
MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, loopback not set
Keepalive set (10 sec)
Auto-duplex, Auto Speed, 100BaseTX
ARP type: ARPA, ARP Timeout 04:00:00
Last input never, output never, output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
```

```

Queueing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
0 packets input, 0 bytes
Received 0 broadcasts (0 IP multicast)
0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
0 watchdog, 0 multicast
0 input packets with dribble condition detected
0 packets output, 0 bytes, 0 underruns
0 output errors, 0 collisions, 0 interface resets
0 babbles, 0 late collision, 0 deferred
0 lost carrier, 0 no carrier
0 output buffer failures, 0 output buffers swapped out

```

Enter the **show controller** command to display information about the Fast Ethernet controller chip.

[Example 4-9](#) shows the output from the **show controller** command, which shows statistics, including information about initialization block information.

Example 4-9 *show controller Command Output*

```

Router# show controller fastEthernet 0
IF Name: FastEthernet0
Port Status DOWN
Send Flow Control      : Disabled
Receive Flow Control  : Enabled
MAC registers
CMCR : 0x0000042D (Tx Enabled, Rx Disabled)
CMPR : 0x150B0A80 (Long Frame Disabled)
FCR  : 0x0000A00B (Rx Pause detection Enabled)
MII registers:
Control Register          (0x0): 0x4000 (Auto negotiation disabled)
Status Register          (0x1): 0x7809 (Link status Down)
PHY Identification Register 1 (0x2): 0x40
PHY Identification Register 2 (0x3): 0x61D4
Auto Neg. Advertisement Reg (0x4): 0x1E1 (Speed 100, Duplex Full)
Auto Neg. Partner Ability Reg (0x5): 0x0 (Speed 10, Duplex Half)
Auto Neg. Expansion Register (0x6): 0x4
100Base-X Aux Control Reg (0x10): 0x2000
100Base-X Aux Status Register(0x11): 0x0
100Base-X Rcv Error Counter (0x12): 0x0
100Base-X False Carr. Counter(0x13): 0x0

```

Enter the **show run interface** [*type number*] command to display information about the configuration of the Fast Ethernet interface. The command is useful when there are multiple interfaces and you want to look at the configuration of a specific interface.

[Example 4-10](#) shows output from the **show run interface** [*type number*] command, which includes information about the IP or lack of IP address and the state of the interface.

Example 4-10 *show run interface Command Output*

```

daytona# show run interface FastEthernet 1
Building configuration...

Current configuration : 56 bytes
!
interface FastEthernet1
no ip address
shutdown

```

end



Configuring POS

This chapter describes advanced packet-over-SONET/SDH (POS) interface configuration for the ML-Series card. Basic POS interface configuration is included in [Chapter 4, “Configuring Interfaces.”](#) For more information about the Cisco IOS commands used in this chapter, refer to the *Cisco IOS Command Reference* publication. POS operation on ONS Ethernet cards, including the ML-Series card, is described in [Chapter 21, “POS on ONS Ethernet Cards.”](#)

This chapter contains the following major sections:

- [POS on the ML-Series Card, page 5-1](#)
- [Monitoring and Verifying POS, page 5-9](#)
- [POS Configuration Examples, page 5-11](#)

POS on the ML-Series Card

Ethernet and IP data packets need to be framed and encapsulated into SONET/SDH frames for transport across the SONET/SDH network. This framing and encapsulation process is known as POS and is done in the ML-Series card. [Chapter 5, “Configuring POS”](#) explains POS in greater detail.

The ML-Series card takes the standard Ethernet ports on the front of the card and the virtual POS ports and includes them all as switch ports. Under Cisco IOS, the POS port is an interface similar to the other Ethernet interfaces on the ML-Series card. It is usually used as a trunk port. Many standard Cisco IOS features, such as IEEE 802.1 Q VLAN configuration, are configured on the POS interface in the same manner as on a standard Ethernet interface. Other features and configurations are done strictly on the POS interface. The configuration of features limited to POS ports is shown in this chapter.

ML-Series SONET and SDH Circuit Sizes

SONET is an American National Standards Institute (ANSI) standard (T1.1051988) for optical digital transmission at hierarchical rates from 51.840 Mbps (STS-1) to 2.488 Gbps (STS-48) and greater. SDH is the international standard for optical digital transmission at hierarchical rates from 155.520 Mbps (STM-1) to 2.488 Gbps (STM-16) and greater.

Both SONET and SDH are based on a structure that has a basic frame and speed. The frame format used by SONET is the synchronous transport signal (STS), with STS-1 being the base level signal at 51.84 Mbps. A STS-1 frame can be carried in an OC-1 signal. The frame format used by SDH is the synchronous transport module (STM), with STM-1 being the base level signal at 155.52 Mbps. A STM-1 frame can be carried in an OC-3 signal.

Both SONET and SDH have a hierarchy of signaling speeds. Multiple lower level signals can be multiplexed together to form higher level signals. For example, three STS-1 signals can be multiplexed together to form a STS-3 signal, and four STM-1 signals can be multiplexed together to form a STM-4 signal.

SONET circuit sizes are defined as STS-n, where n is a multiple of 51.84 Mbps and n is equal to or greater than 1. SDH circuit sizes are defined as STM-n, where n is a multiple of 155.52 Mbps and n is equal to or greater than 0. [Table 5-1](#) shows STS and STM line rate equivalents.

Table 5-1 SONET STS Circuit Capacity in Line Rate Mbps

SONET Circuit Size	SDH Circuit Size	Line Rate in Mbps
STS-1 (OC-1)	VC-3 ¹	52 Mbps
STS-3c (OC-3)	STM-1 (VC4)	156 Mbps
STS-6c (OC-6)	STM-2 (VC4-2c)	311 Mbps
STS-9c (OC-9)	STM-3 (VC4-3c)	466 Mbps
STS-12c (OC-12)	STM-4 (VC4-4c)	622 Mbps
STS-24c (OC-24)	STM-8 (VC4-8c)	1244 Mbps (1.24 Gbps)

1. VC-3 circuit support requires an XCVL card to be installed.

For step-by-step instructions on configuring an ML-Series card SONET STS circuit, refer to the “Create Circuits and VT Tunnels” chapter of the *Cisco ONS 15454 Procedure Guide*. For step-by-step instructions on configuring an ML-Series card SDH STM circuit, refer to the “Create Circuits and Tunnels” chapter of the *Cisco ONS 15454 SDH Procedure Guide*.

VCAT

VCAT significantly improves the efficiency of data transport over SONET/SDH by grouping the synchronous payload envelopes (SPEs) of SONET/SDH frames in a nonconsecutive manner into VCAT groups. VCAT group circuit bandwidth is divided into smaller circuits called VCAT members. The individual members act as independent circuits.

Intermediate nodes treat the VCAT members as normal circuits that are independently routed and protected by the SONET/SDH network. At the terminating nodes, these member circuits are multiplexed into a contiguous stream of data. VCAT avoids the SONET/SDH bandwidth fragmentation problem and allows finer granularity for provisioning of bandwidth services.

The ONS 15454 SONET and ONS 15454 SDH ML-Series card VCAT circuits must also be routed over common fiber and be both bidirectional and symmetric. Only high order (HO) VCAT circuits are supported. The ML-Series card supports a maximum of two VCAT groups, with each group corresponding to one of the POS ports. Each VCAT group can contain two circuit members. [Table 5-2](#) shows supported VCAT circuit sizes for the ML-Series.

Table 5-2 VCAT Circuit Sizes Supported by ML100T-12 and ML1000-2 Cards

SONET VCAT Circuit Size	SDH VCAT Circuit Size
STS-1-2v	VC-3-2v
STS-3c-2v	VC-4-2v
STS-12c-2v	VC-4-4c-2v

A VCAT circuit originating from an ML-Series card must terminate on another ML-Series card or a CE-Series card. [Table 5-3](#) shows ML-Series card supported VCAT circuit sizes.

Table 5-3 VCAT Circuit Sizes Supported by ML100T-12 and ML1000-2 Cards

ML100T-12/ML1000-2 Connected to:	Supported SONET VCAT Circuit Sizes	Supported SDH VCAT Circuit Sizes
ML100T-12/ML1000-2	STS-1-2v, STS-3c-2v, or STS-12c-2v	VC-3-2v, VC-4-2v, or VC-4-4c-2v
ML-100T-8	STS-1-2v	Not applicable
CE-100T-8 (ONS 15310)	STS-1-2v	Not applicable
CE-100T-8 (ONS 15454 and ONS 15454 SDH)	STS-1-2v	Not applicable

For step-by-step instructions on configuring an ML-Series card SONET VCAT circuit, refer to the “Create Circuits and VT Tunnels” chapter of the *Cisco ONS 15454 Procedure Guide*. For step-by-step instructions on configuring an ML-Series card SDH VCAT circuit, refer to the “Create Circuits and Tunnels” chapter of the *Cisco ONS 15454 SDH Procedure Guide*. For more general information on VCAT circuits, refer to the “Circuits and Tunnels” chapter of the *Cisco ONS 15454 Reference Manual* or the *Cisco ONS 15454 SDH Reference Manual*.



Note

ML-Series card POS interfaces normally send PDI-P to the far-end when the POS link goes down or RPR wraps. ML-Series card POS interfaces do not send PDI-P to the far-end when PDI-P is detected, when RDI-P is being sent to the far-end or when the only defects detected are GFP LFD, GFP CSF, VCAT LOM or VCAT SQM.

SW-LCAS

A link capacity adjustment scheme (LCAS) increases VCAT flexibility by allowing the dynamic reconfiguration of VCAT groups without interrupting the operation of noninvolved members. Software link capacity adjustment scheme (SW-LCAS) is the software implementation of a LCAS-type feature. SW-LCAS differs from LCAS because it is not errorless and uses a different handshaking mechanism.

SW-LCAS on the ML-Series cards allows the automatic addition or removal of a VCAT group member in the event of a failure or recovery on a two-fiber bidirectional line switched ring (BLSR). The protection mechanism software operates based on ML-Series card link events. SW-LCAS allows service providers to configure VCAT member circuits on the ML-Series as protection channel access (PCA) circuits. This PCA traffic is dropped in the event of a protection switch, but is suitable for excess or noncommitted traffic and can double the total available bandwidth on the circuit.

For step-by-step instructions on configuring SW-LCAS, refer to the “Create Circuits and VT Tunnels” chapter of the *Cisco ONS 15454 Procedure Guide* or the “Create Circuits and Tunnels” chapter of the *Cisco ONS 15454 SDH Procedure Guide*. For more general information on SW-LCAS, refer to the “Circuits and Tunnels” chapter of the *Cisco ONS 15454 Reference Manual* or the *Cisco ONS 15454 SDH Reference Manual*.

Framing Mode, Encapsulation, and CRC Support

The ML-Series cards on the ONS 15454 and ONS 15454 SDH support two modes of the POS framing mechanism, GFP-F framing and HDLC framing (default). The framing mode, encapsulation, and CRC size on source and destination POS ports must match for a POS circuit to function properly. [Chapter 5, “Configuring POS”](#) explains the framing mechanisms, encapsulations, and cyclic redundancy check (CRC) bit sizes in detail.

Supported encapsulation and CRC sizes for the framing types are detailed in [Table 5-4](#).

Table 5-4 Supported Encapsulation, Framing, and CRC Sizes for ML-Series Cards on the ONS 15454 and ONS 15454 SDH

	Encapsulations for HDLC Framing	CRC Sizes for HDLC Framing	Encapsulations for GFP-F Framing	CRC Sizes for GFP-F Framing
ML-Series	LEX (default)	16-bit	LEX (default)	32-bit (default)
	Cisco HDLC	32-bit (default)	Cisco HDLC	
	PPP/BCP		PPP/BCP	



Note

ML-Series card POS interfaces normally send PDI-P to the far-end when the POS link goes down or RPR wraps. ML-Series card POS interfaces do not send PDI-P to the far-end when PDI-P is detected, when RDI-P is being sent to the far-end or when the only defects detected are GFP LFD, GFP CSF, VCAT LOM or VCAT SQM.

Configuring Framing Mode

You configure framing mode on an ML-Series card only through CTC. For more information on configuring framing mode in CTC, see [Chapter 2, “CTC Operations.”](#)

Configuring Encapsulation Type


To configure the encapsulation type for a ML-Series card, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# interface pos <i>number</i>	Activates interface configuration mode to configure the POS interface.
Step 2	Router(config-if)# shutdown	Manually shuts down the interface. Encapsulation changes on POS ports are allowed only when the interface is shut down (ADMIN_DOWN).

	Command	Purpose
Step 3	Router(config-if)# encapsulation <i>type</i>	Sets the encapsulation type. Valid values are: <ul style="list-style-type: none"> • hdlc—Cisco HDLC • lex—(default) LAN extension, special encapsulation for use with Cisco ONS Ethernet line cards • ppp—Point-to-Point Protocol
Step 4	Router(config-if)# no shutdown	Restarts the shutdown interface.
Step 5	Router(config)# end	Returns to privileged EXEC mode.
Step 6	Router# copy running-config startup-config	(Optional) Saves configuration changes to NVRAM.

Configuring CRC Size in HDLC Framing

To configure additional properties to match those of the interface at the far end, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# interface pos <i>number</i>	Activates interface configuration mode to configure the POS interface.
Step 2	Router(config-if)# crc {16 32}	Sets the CRC value for HDLC framing. If the device to which the POS module is connected does not support the default CRC value of 32, set both devices to use a value of 16.  Note The CRC value under GFP-F framing is fixed and non-configurable.
Step 3	Router(config-if)# end	Returns to the privileged EXEC mode.
Step 4	Router# copy running-config startup-config	(Optional) Saves configuration changes to NVRAM.

Setting the MTU Size

To set the maximum transmission unit (MTU) size, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# interface pos <i>number</i>	Activates interface configuration mode to configure the POS interface.
Step 2	Router(config-if)# mtu <i>bytes</i>	Configures the MTU size up to a maximum of 9000 bytes. See Table 5-5 for default MTU sizes.

	Command	Purpose
Step 3	Router(config-if)# end	Returns to the privileged EXEC mode.
Step 4	Router# copy running-config startup-config	(Optional) Saves configuration changes to NVRAM.

Table 5-5 shows the default MTU sizes.

Table 5-5 Default MTU Size

Encapsulation Type	Default Size
LEX (default)	1500
HDLC	4470
PPP	4470

Configuring Keep Alive Messages

To configure keep alive messages for the ML-Series card, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# interface pos number	Enters interface configuration mode and specifies the POS interface to configure.
Step 2	Router(config-if)# [no] keepalive	Configures keep alive messages. Keep alive messages are on by default and are recommended, but not required. The no form of this command turns off keep alive messages.
Step 3	Router(config-if)# end	Returns to the privileged EXEC mode.
Step 4	Router# copy running-config startup-config	(Optional) Saves configuration changes to NVRAM.

SONET/SDH Alarms

The ML-Series cards report SONET/SDH alarms under both Cisco IOS and CTC/TL1. A number of path alarms are reported in the Cisco IOS console. Configuring Cisco IOS console alarm reporting has no effect on CTC alarm reporting. The “[Configuring SONET/SDH Alarms](#)” procedure specifies the alarms reported to the Cisco IOS console.

CTC/TL1 has sophisticated SONET/SDH alarm reporting capabilities. As a card in the ONS node, the ML-Series card reports alarms to CTC/TL-1 like any other ONS card. On the ONS 15454 SONET, the ML-Series card reports Telcordia GR-253 SONET alarms in the Alarms panel of CTC. For more information on alarms and alarm definitions, refer to the “Alarm Troubleshooting” chapter of the *Cisco ONS 15454 Troubleshooting Guide*, or the *Cisco ONS 15454 SDH Troubleshooting Guide*.

CTC/TL1 has sophisticated SONET/SDH alarm reporting capabilities. As a card in the ONS node, the ML-Series card reports alarms to CTC/TL-1 like any other ONS card. On the ONS 15454 SONET, the ML-Series card reports Telcordia GR-253 SONET alarms in the Alarms panel of CTC. For more information on alarms and alarm definitions, refer to the “Alarm Troubleshooting” chapter of the *Cisco ONS 15454 Troubleshooting Guide*, or the *Cisco ONS 15454 SDH Troubleshooting Guide*.

Configuring SONET/SDH Alarms

All SONET/SDH alarms are logged on the Cisco IOS CLI by default. But to provision or disable the reporting of SONET/SDH alarms on the Cisco IOS CLI, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# interface pos <i>number</i>	Enters interface configuration mode and specifies the POS interface to configure.
Step 2	Router(config-if)# pos report { all encap pais plop ppdi pplm prdi ptim puneq sd-ber-b3 sf-ber-b3 }	Permits logging of selected SONET/SDH alarms. Use the no form of the command to disable reporting of a specific alarm. The alarms are as follows: <ul style="list-style-type: none"> • all—All alarms/signals • encap—Path encapsulation mismatch • pais—Path alarm indication signal • plop—Path loss of pointer • ppdi—Path payload defect indication • pplm—Payload label, C2 mismatch • prdi—Path remote defect indication • ptim—Path trace identifier mismatch • puneq—Path label equivalent to zero • sd-ber-b3—PBIP BER in excess of SD threshold • sf-ber-b3—PBIP BER in excess of SF threshold
Step 3	Router(config-if)# end	Returns to the privileged EXEC mode.
Step 4	Router# copy running-config startup-config	(Optional) Saves configuration changes to NVRAM.

To determine which alarms are reported on the POS interface and to display the bit error rate (BER) thresholds, use the **show controllers pos** command, as described in the “[Monitoring and Verifying POS](#)” section on page 5-9.



Note

Cisco IOS alarm reporting commands apply only to the Cisco IOS CLI. SONET/SDH alarms reported to the TCC2/TCC2P are not affected.

To configure path alarms as triggers and specify a delay, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# interface <i>pos number</i>	Enters interface configuration mode and specifies the POS interface to configure.
Step 2	Router(config-if)# pos trigger defect { <i>all</i> <i>ber_sf_b3</i> <i>encap</i> <i>pais</i> <i>plop</i> <i>ppdi</i> <i>pplm</i> <i>prdi</i> <i>ptim</i> <i>puneq</i> }	Configures certain path defects as triggers to bring down the POS interface. The configurable triggers are as follows: <ul style="list-style-type: none"> • all—All link down alarm failures • ber_sd_b3—PBIP BER in excess of SD threshold failure • ber_sf_b3—PBIP BER in excess of SD threshold failure (default) • encap—Path Signal Label Encapsulation Mismatch failure (default) • pais—Path Alarm Indication Signal failure (default) • plop—Path Loss of Pointer failure (default) • ppdi—Path Payload Defect Indication failure (default) • pplm—Payload label mismatch path (default) • prdi—Path Remote Defect Indication failure (default) • ptim—Path Trace Indicator Mismatch failure (default) • puneq—Path Label Equivalent to Zero failure (default)
Step 3	Router(config-if)# pos trigger delay <i>millisecond</i>	Sets waiting period before the line protocol of the interface goes down. Delay can be set from 200 to 2000 ms. If no time intervals are specified, the default delay is set to 200 ms.
Step 4	Router(config-if)# end	Returns to the privileged EXEC mode.
Step 5	Router# copy running-config startup-config	(Optional) Saves configuration changes to NVRAM.

C2 Byte and Scrambling

One of the overhead bytes in the SONET/SDH frame is the C2 byte. The SONET/SDH standard defines the C2 byte as the path signal label. The purpose of this byte is to communicate the payload type being encapsulated by the SONET framing overhead (FOH). The C2 byte functions similarly to EtherType and Logical Link Control (LLC)/Subnetwork Access Protocol (SNAP) header fields on an Ethernet network; it allows a single interface to transport multiple payload types simultaneously. The C2 byte is not configurable. Table 5-6 provides C2 byte hex values.

Table 5-6 C2 Byte and Scrambling Default Values

Signal Label	SONET/SDH Payload Contents
0x01	LEX Encapsulation with 32-bit CRC with or without scrambling
0x05	LEX Encapsulation with 16-bit CRC with or without scrambling
0xCF	Cisco HDLC or PPP/BCP without scrambling

Table 5-6 C2 Byte and Scrambling Default Values (continued)

Signal Label	SONET/SDH Payload Contents
0x16	Cisco HDLC or PPP/BCP with scrambling
0x1B	GFP-F

Third-Party POS Interfaces C2 Byte and Scrambling Values

If a Cisco POS interface fails to come up when connected to a third-party device, confirm the scrambling and cyclic redundancy check (CRC) settings as well as the advertised value in the C2 byte. On routers from Juniper Networks, configuring RFC 2615 mode sets the following three parameters:

- Scrambling enabled
- C2 value of 0x16
- CRC-32

Previously, when scrambling was enabled, these third-party devices continued to use a C2 value of 0xCF, which did not properly reflect the scrambled payload.

Configuring SPE Scrambling

SPE scrambling is on by default. To configure POS SONET/SDH Payload (SPE) scrambling, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	<code>Router(config)# interface pos number</code>	Enters interface configuration mode and specifies the POS interface to configure.
Step 2	<code>Router(config-if)# no pos scramble-spe</code>	Disables payload scrambling on the interface. Payload scrambling is on by default.
Step 3	<code>Router(config-if)# no shutdown</code>	Enables the interface with the previous configuration.
Step 4	<code>Router(config-if)# end</code>	Returns to the privileged EXEC mode.
Step 5	<code>Router# copy running-config startup-config</code>	(Optional) Saves configuration changes to NVRAM.

Monitoring and Verifying POS

The `show controller pos [0 | 1]` command (Example 5-1) outputs the receive and transmit values and the C2 value. Thus, changing the value on the local end does not change the value in the `show controller` command output.

Example 5-1 show controller pos [0 | 1] Command

```
ML_Series# sh controllers pos 0
Interface POS0
Hardware is Packet/Ethernet over Sonet
Framing Mode: HDLC
```

```

Concatenation: CCAT
Alarms reportable to CLI: PAIS PLOP PUNEQ PTIM PPLM ENCAP PRDI PPDI BER_SF_B3 BER_SD_B3
VCAT_OOU_TPT LOM SQM
Link state change defects: PAIS PLOP PUNEQ PTIM PPLM ENCAP PRDI PPDI BER_SF_B3
Link state change time : 200 (msec)
***** Path *****
Circuit state: IS
    PAIS      = 0          PLOP      = 0          PRDI      = 0          PTIM      = 0
    PPLM      = 0          PUNEQ     = 0          PPDI      = 0          PTIU      = 0
    BER_SF_B3 = 0          BER_SD_B3 = 0          BIP(B3)   = 0          REI       = 0
    NEWPTR    = 0          PSE       = 0          NSE       = 0          ENCAP     = 0
Active Alarms : PAIS
Demoted Alarms: None
Active Defects: PAIS
DOS FPGA channel number : 0
Starting STS (0 based) : 0
VT ID (if any) (0 based) : 255
Circuit size : STS-3c
RDI Mode : 1 bit
C2 (tx / rx) : 0x01 / 0x01
Framing : SONET
Path Trace
    Mode : off
    Transmit String :
    Expected String :
    Received String :
    Buffer : Stable
    Remote hostname :
    Remote interface:
    Remote IP addr :
B3 BER thresholds:
SFBER = 1e-4, SDBER = 1e-7
0 total input packets, 0 post-HDLC bytes
0 input short packets, 0 pre-HDLC bytes
0 input long packets , 0 input runt packets
0 input CRCerror packets , 0 input drop packets
0 input abort packets
0 input packets dropped by ucode
0 total output packets, 0 output pre-HDLC bytes
0 output post-HDLC bytes
Carrier delay is 200 msec

```

The **show interface pos {0 | 1}** command ([Example 5-2](#)) shows scrambling.

Example 5-2 *show interface pos [0 | 1] Command*

```

ML_Series# show interface pos 0
POS0 is administratively down, line protocol is down
  Hardware is Packet/Ethernet over Sonet, address is 0011.2130.b340 (bia 0011.2130.b340)
  MTU 1500 bytes, BW 145152 Kbit, DLY 100 usec,
  reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation: Cisco-EoS-LEX, crc 32, loopback not set
  Keepalive set (10 sec)
  Scramble enabled
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 01:21:02, output never, output hang never
  Last clearing of "show interface" counters 00:12:01
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
  0 packets input, 0 bytes

```



```

Received 0 broadcasts (0 IP multicast)
0 runts, 0 giants, 0 throttles
  0 parity
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
0 input packets with dribble condition detected
0 packets output, 0 bytes, 0 underruns
0 output errors, 0 applique, 0 interface resets
0 babbles, 0 late collision, 0 deferred
0 lost carrier, 0 no carrier
0 output buffer failures, 0 output buffers swapped out
0 carrier transitions

```

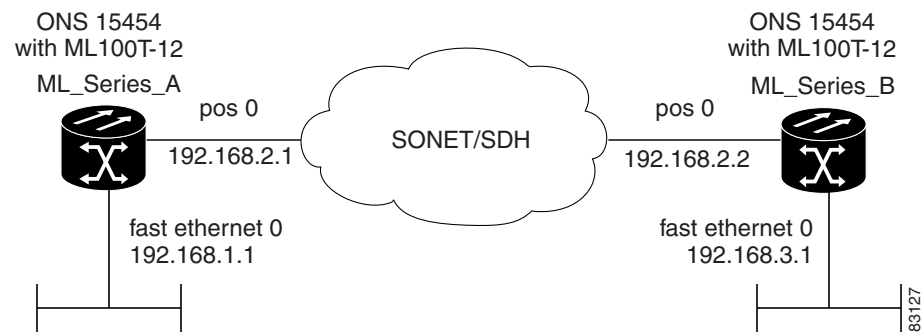
POS Configuration Examples

The following sections show ML-Series card POS configuration examples for connecting to other ONS Ethernet cards and POS-capable routers. These examples are only some of the ML-Series card configurations available to connect to other ONS Ethernet cards and POS-capable routers. For more specifics about the POS characteristics of ONS Ethernet cards, see [Chapter 21, “POS on ONS Ethernet Cards.”](#)

ML-Series Card to ML-Series Card

[Figure 5-1](#) illustrates a POS configuration between two ONS 15454 or ONS 15454 SDH ML-Series cards.

Figure 5-1 ML-Series Card to ML-Series Card POS Configuration



[Example 5-3](#) shows the code associated with the configuration of ML-Series card A.

Example 5-3 ML-Series Card A Configuration

```

hostname ML_Series_A
!
interface FastEthernet0
 ip address 192.168.1.1 255.255.255.0
!
interface POS0
 ip address 192.168.2.1 255.255.255.0
 crc 32
 pos flag c2 1
!
router ospf 1

```

```
log-adjacency-changes
network 192.168.1.0 0.0.0.255 area 0
network 192.168.2.0 0.0.0.255 area 0
```

Example 5-4 shows the code associated with the configuration of ML Series B.

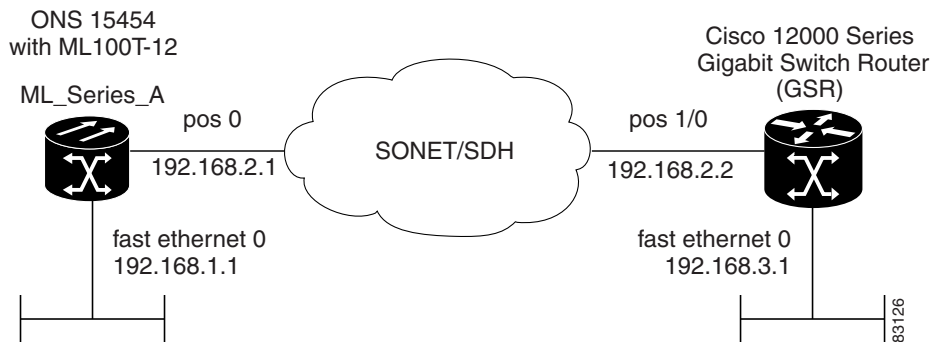
Example 5-4 ML-Series Card B Configuration

```
hostname ML_Series_B
!
interface FastEthernet0
 ip address 192.168.3.1 255.255.255.0
!
interface POS0
 ip address 192.168.2.2 255.255.255.0
 crc 32
 pos flag c2 1
!
router ospf 1
 log-adjacency-changes
 network 192.168.2.0 0.0.0.255 area 0
 network 192.168.3.0 0.0.0.255 area 0
!
```

ML-Series Card to Cisco 12000 GSR-Series Router

Figure 5-2 illustrates a POS configuration between an ML-Series card and a Cisco 12000 GSR-Series router. PPP/BCP encapsulation or Cisco HDLC encapsulation may be used for interoperation.

Figure 5-2 ML-Series Card to Cisco 12000 Series Gigabit Switch Router (GSR) POS Configuration



Example 5-5 shows the code associated with configuration of ML-Series card A.

Example 5-5 ML-Series Card A Configuration

```
hostname ML_Series_A
!
interface FastEthernet0
 ip address 192.168.1.1 255.255.255.0
!
!
interface POS0
 ip address 192.168.2.1 255.255.255.0
 encapsulation ppp
```

```

    crc 32
    !
router ospf 1
  log-adjacency-changes
  network 192.168.1.0 0.0.0.255 area 0
  network 192.168.2.0 0.0.0.255 area 0

```

Example 5-6 shows the code associated with the configuration of the GSR-12000.

Example 5-6 GSR-12000 Configuration

```

hostname GSR
!
interface FastEthernet1/0
  ip address 192.168.3.1 255.255.255.0
!
interface POS2/0
  ip address 192.168.2.2 255.255.255.0
  crc 32
  encapsulation PPP
  pos scramble-atm
!
router ospf 1
  log-adjacency-changes
  network 192.168.2.0 0.0.0.255 area 0
  network 192.168.3.0 0.0.0.255 area 0
!

```

The default encapsulation for the ML-Series card is LEX and the corresponding default MTU is 1500 bytes. When connecting to an external POS device, it is important to ensure that both the ML-Series switch and the external device uses the same configuration for the parameters listed in Table 5-7.

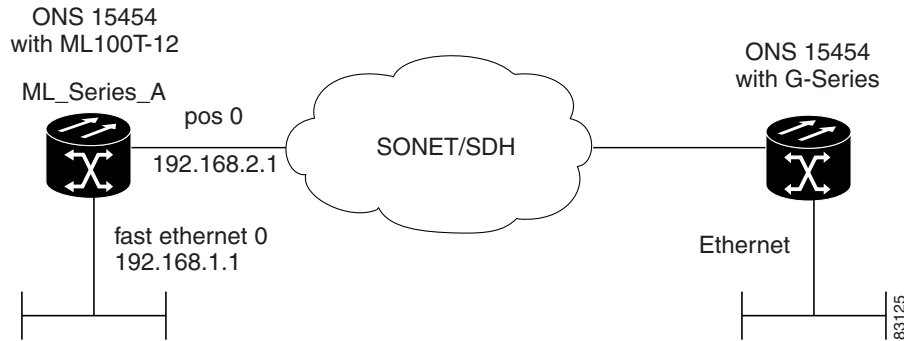
Table 5-7 ML-Series Parameter Configuration for Connection to a Cisco 12000 GSR-Series Router

Command	Parameter
Router(config-if)# encapsulation ppp or Router(config-if)# encapsulation hdlc	Encapsulation—Default encapsulation on the Cisco 12000 GSR Series is HDLC, which is supported by the ML-Series. PPP is also supported by both the ML-Series card and the Cisco 12000 GSR Series. The Cisco 12000 GSR Series does not support LEX, which is the default encapsulation on the ML-Series card.
Router(config-if)# show controller pos	C2 Byte—Use the show controller pos command to verify that the transmit and receive C2 values are the same.
Router(config-if)# pos flag c2 value	Sets the C2 byte value. Valid choices are 0 to 255 (decimal). The default value is 0x01 (hex) for LEX.

ML-Series Card to G-Series Card

Figure 5-3 illustrates a POS configuration between an ML-Series card and a G-Series card.

Figure 5-3 ML-Series Card to G-Series Card POS Configuration



Example 5-7 shows the code associated with the configuration of ML-Series card A.

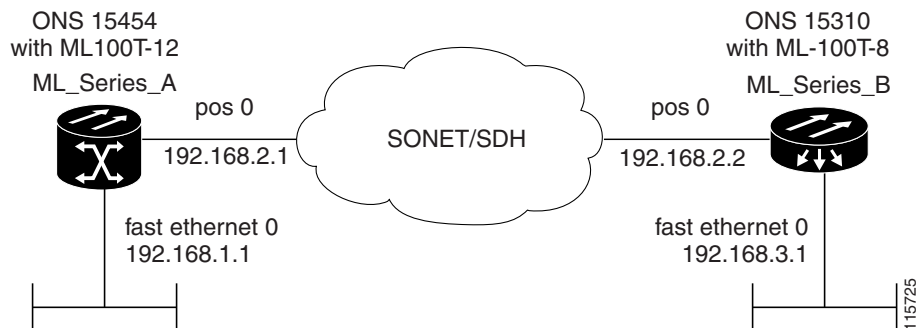
Example 5-7 ML-Series Card A Configuration

```
hostname ML_Series_A
!
interface FastEthernet0
 ip address 192.168.1.1 255.255.255.0
!
interface POS0
 ip address 192.168.2.1 255.255.255.0
 crc 32
!
router ospf 1
 log-adjacency-changes
 network 192.168.1.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0
```

ML-Series Card to ONS 15310 ML-100T-8 Card

Figure 5-3 illustrates a POS configuration between an ML-Series card and an ONS 15310 ML-100T-8 card. For step-by-step circuit configuration procedures for the connected ML-100T-8 card, refer to the *Cisco ONS 15310-CL and Cisco ONS 15310-MA Ethernet Card Software Feature and Configuration Guide*.

Figure 5-4 ML-Series Card to ONS 15310 CE-100T-8 Card Configuration



[Example 5-7](#) shows the code associated with the configuration of ML-Series card A.

Example 5-8 ML-Series Card A Configuration

```
hostname ML_Series_A
!
interface FastEthernet0
 ip address 192.168.1.1 255.255.255.0
!
interface POS0
 ip address 192.168.2.1 255.255.255.0
 crc 32
!
router ospf 1
 log-adjacency-changes
 network 192.168.1.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0
```




Configuring Bridges

This chapter describes how to configure bridging for the ML-Series card. For more information about the Cisco IOS commands used in this chapter, refer to the *Cisco IOS Command Reference* publication.

This chapter includes the following major sections:

- [Understanding Bridging, page 6-1](#)
- [Configuring Bridging, page 6-2](#)
- [Monitoring and Verifying Bridging, page 6-3](#)



Caution

Cisco Inter-Switch Link (ISL) and Cisco Dynamic Trunking Protocol (DTP) are not supported by the ML-Series cards, but the ML-Series broadcast forwards these formats. Using ISL or DTP on connecting devices is not recommended. Some Cisco devices attempt to use ISL or DTP by default.

Understanding Bridging

The ML-Series card supports transparent bridging for Fast Ethernet, Gigabit Ethernet and POS ports. It supports a maximum of 255 active bridge groups. Transparent bridging combines the speed and protocol transparency of a spanning-tree bridge, along with the functionality, reliability, and security of a router.

To configure bridging, you must perform the following tasks in the modes indicated:

- In global configuration mode:
 - Enable bridging of IP packets.
 - Select the type of Spanning Tree Protocol (STP) (optional).
- In interface configuration mode:
 - Determine which interfaces belong to the same bridge group.

The ML-Series card bridges all nonrouted traffic among the network interfaces comprising the bridge group. If spanning tree is enabled, the interfaces became part of the same spanning tree. Interfaces not participating in a bridge group cannot forward bridged traffic.

If the destination address of the packet is known in the bridge table, the packet is forwarded on a single interface in the bridge group. If the packet's destination is unknown in the bridge table, the packet is flooded on all forwarding interfaces in the bridge group. The bridge places source addresses in the bridge table as it learns them during the process of bridging.

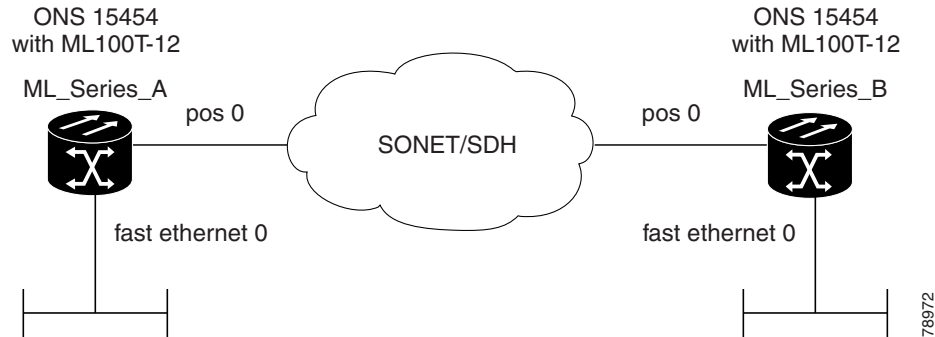
Spanning tree is not mandatory for an ML-Series card bridge group. But if it is configured, a separate spanning-tree process runs for each configured bridge group. A bridge group establishes a spanning tree based on the bridge protocol data units (BPDUs) it receives on only its member interfaces. The ML-Series card has a 255 active bridge group maximum.

Configuring Bridging

Use the following steps to configure bridging:

	Command	Purpose
Step 1	Router(config)# no ip routing	Enables bridging of IP packets. This command needs to be executed once per card, not once per bridge-group. This step is not done for integrated routing and bridging (IRB).
Step 2	Router(config)# bridge <i>bridge-group-number</i> [protocol { <i>drpri-rstp</i> <i>rstp</i> <i>ieee</i> }]	Assigns a bridge group number and defines the appropriate spanning-tree type: bridge-group-number can range from 1 to 4096. <i>drpri-rstp</i> is the protocol used to interconnect dual RPR to protect from node failure <i>rstp</i> is the IEEE 802.1W Rapid Spanning Tree. <i>ieee</i> is the IEEE 802.1D Spanning Tree Protocol . Note Spanning tree is not mandatory for an ML-Series card bridge group. But configuring spanning tree blocks network loops.
Step 3	Router(config)# bridge <i>bridge-group-number</i> priority <i>number</i>	(Optional) Assigns a specific priority to the bridge, to assist in the spanning-tree root definition. Lowering the priority of a bridge makes it more likely the bridge is selected as the root.
Step 4	Router(config)# interface <i>type</i> <i>number</i>	Enters interface configuration mode to configure the interface of the ML-Series card.
Step 5	Router(config-if)# bridge-group <i>bridge-group-number</i>	Assigns a network interface to a bridge group.
Step 6	Router(config-if)# no shutdown	Changes the shutdown state to up and enables the interface.
Step 7	Router(config-if)# end	Returns to privileged EXEC mode.
Step 8	Router# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Figure 6-1 shows a bridging example. Example 6-1 shows the code used to configure ML-Series A. Example 6-2 shows the code used to configure ML-Series B.

Figure 6-1 Bridging Example**Example 6-1 Router A Configuration**

```
bridge 1 protocol ieee
!
!
interface FastEthernet0
 no ip address
 bridge-group 1
!
interface POS0
 no ip address
 crc 32
 bridge-group 1
 pos flag c2 1
```

Example 6-2 Router B Configuration

```
bridge 1 protocol ieee
!
!
interface FastEthernet0
 no ip address
 bridge-group 1
!
interface POS0
 no ip address
 crc 32
 bridge-group 1
 pos flag c2 1
```

Monitoring and Verifying Bridging

After you have set up the ML-Series card for bridging, you can monitor and verify its operation by performing the following procedure in privileged EXEC mode:

	Command	Purpose
Step 1	Router# clear bridge <i>bridge-group-number</i>	Removes any learned entries from the forwarding database of a particular bridge group, clears the transmit, and receives counts for any statically configured forwarding entries.
Step 2	Router# show bridge { <i>bridge-group-number</i> <i>interface-address</i> }	Displays classes of entries in the bridge forwarding database.
Step 3	Router# show bridge verbose	Displays detailed information about configured bridge groups.
Step 4	ML_Series# show spanning-tree [<i>bridge-group-number</i>] [brief]	Displays detailed information about spanning tree. <i>bridge-group-number</i> restricts the spanning tree information to specific bridge groups. brief displays summary information about spanning tree.

Example 6-3 shows an example of the monitoring and verifying bridging.

Example 6-3 Monitoring and Verifying Bridging

```
ML-Series# show bridge

Total of 300 station blocks, 298 free
Codes: P - permanent, S - self

Bridge Group 1:

Maximum dynamic entries allowed: 1000
Current dynamic entry count: 2

      Address      Action  Interface
0000.0001.6000  forward FastEthernet0
0000.0001.6100  forward POS0

ML-Series# show bridge verbose

Total of 300 station blocks, 298 free
Codes: P - permanent, S - self

Maximum dynamic entries allowed: 1000
Current dynamic entry count: 2

BG Hash      Address      Action  Interface      VC   Age   RX count  TX co
unt
  1 60/0     0000.0001.6000 forward FastEthernet0   -
  1 61/0     0000.0001.6100 forward POS0      -

Flood ports
FastEthernet0
POS0

ML-Series# show spanning-tree brief

Bridge group 1
  Spanning tree enabled protocol ieee
  Root ID      Priority    32769
              Address    0005.9a39.6634
              This bridge is the root
```

```

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)
Address 0005.9a39.6634
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Aging Time 300

Interface      Role Sts Cost      Prio.Nbr Type
-----
Fa0            Desg FWD 19        128.3   P2p
PO0            Desg FWD 9         128.20  P2p
```




Configuring STP and RSTP

This chapter describes the IEEE 802.1D Spanning Tree Protocol (STP) and the ML-Series implementation of the IEEE 802.1W Rapid Spanning Tree Protocol (RSTP). It also explains how to configure STP and RSTP on the ML-Series card.

This chapter consists of these sections:

- [STP Features, page 7-1](#)
- [RSTP, page 7-9](#)
- [Interoperability with IEEE 802.1D STP, page 7-15](#)
- [Configuring STP and RSTP Features, page 7-15](#)
- [Verifying and Monitoring STP and RSTP Status, page 7-20](#)

STP Features

These sections describe how the spanning-tree features work:

- [STP Overview, page 7-2](#)
- [Supported STP Instances, page 7-2](#)
- [Bridge Protocol Data Units, page 7-2](#)
- [Election of the Root Switch, page 7-3](#)
- [Bridge ID, Switch Priority, and Extended System ID, page 7-4](#)
- [Spanning-Tree Timers, page 7-4](#)
- [Creating the Spanning-Tree Topology, page 7-4](#)
- [Spanning-Tree Interface States, page 7-5](#)
- [Spanning-Tree Address Management, page 7-8](#)
- [STP and IEEE 802.1Q Trunks, page 7-8](#)
- [Spanning Tree and Redundant Connectivity, page 7-8](#)
- [Accelerated Aging to Retain Connectivity, page 7-9](#)

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. 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.

When you create fault-tolerant internetworks, you must have a loop-free path between all nodes in a network. The spanning-tree algorithm calculates the best loop-free path throughout a switched Layer 2 network. 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 the frames to construct a loop-free path.

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 defines a tree with a root switch and a loop-free path from the root to all switches in the Layer 2 network. 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.

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

Supported STP Instances

The ML-Series card supports the per-VLAN spanning tree (PVST+) and a maximum of 255 spanning-tree instances.

Bridge Protocol Data Units

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

- Unique bridge ID (switch priority and MAC address) associated with each VLAN on each switch
- Spanning-tree path cost to the root switch
- 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. The BPDUs communicate and compute the spanning-tree topology. Each configuration BPDU contains this information:

- Unique bridge ID of the switch that the sending switch identifies as the root switch
- Spanning-tree path cost to the root
- Bridge ID of the sending switch
- Message age
- 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, etc.), 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.
- 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.
- Interfaces included in the spanning-tree instance are selected. Root ports and designated ports are put in the forwarding state.
- All interfaces not included in the spanning tree are blocked.

Election of the Root Switch

All switches in the Layer 2 network participating in the spanning tree gather information about other switches in the network through an exchange of BPDU data messages. This exchange of messages results in these actions:

- Election of a unique root switch for each spanning-tree instance
- Election of a designated switch for every switched LAN segment
- Removal of loops in the switched network by blocking Layer 2 interfaces connected to redundant links

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.

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.

The root switch is the logical center of the spanning-tree topology in a switched network. 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.

BPDU contains 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.

Bridge ID, Switch Priority, and Extended System ID

The IEEE 802.1D standard requires that each switch has a unique bridge identifier (bridge ID), which determines 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 ML-Series card supports the IEEE 802.1T spanning-tree extensions, and some of the bits previously used for the switch priority are now used as the bridge ID. 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 7-1](#), 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 bridge ID. In earlier releases, the switch priority is a 16-bit value.

Table 7-1 Switch Priority Value and Extended System ID

Switch Priority Value				Extended System ID (Set Equal to the Bridge 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. With earlier releases, spanning tree used one MAC address per VLAN to make the bridge ID unique for each VLAN.

Spanning-Tree Timers

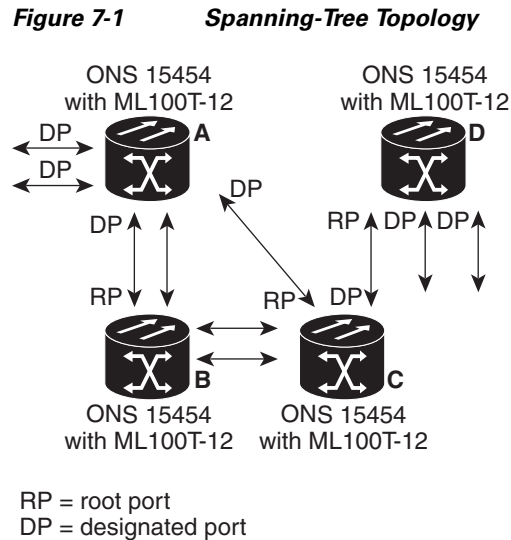
[Table 7-2](#) describes the timers that affect the entire spanning-tree performance.

Table 7-2 Spanning-Tree Timers

Variable	Description
Hello timer	When this timer expires, the interface sends out a Hello message to the neighboring nodes.
Forward-delay timer	Determines how long each of the listening and learning states last before the interface begins forwarding.
Maximum-age timer	Determines the amount of time the switch stores protocol information received on an interface.

Creating the Spanning-Tree Topology

In [Figure 7-1](#), 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.



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.

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 interface 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:

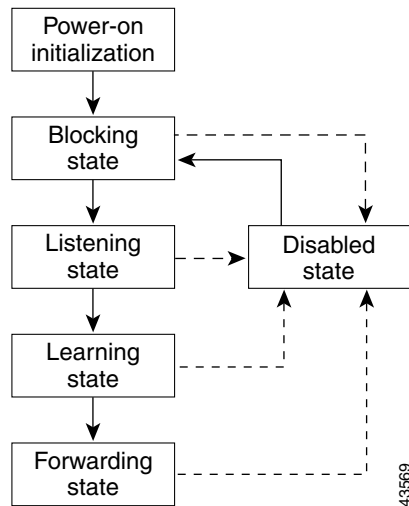
- **Blocking**—The interface does not participate in frame forwarding.
- **Listening**—The first transitional state after the blocking state when the spanning tree determines that the interface should participate in frame forwarding.
- **Learning**—The interface prepares to participate in frame forwarding.
- **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.

An interface moves through these states:

1. From initialization to blocking
2. From blocking to listening or to disabled
3. From listening to learning or to disabled
4. From learning to forwarding or to disabled
5. From forwarding to disabled

Figure 7-2 illustrates how an interface moves through the states.

Figure 7-2 *Spanning-Tree Interface States*



When you power up the switch, STP is enabled by default, and every interface in the switch, VLAN, or network 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 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 for 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 switch 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 interface in the switch. 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 interfaces move to the listening state. An interface always enters the blocking state after switch initialization.

An interface in the blocking state performs as follows:

- Discards frames received on the port
- 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 determines that the interface should participate in frame forwarding.

An interface in the listening state performs as follows:

- Discards frames received on the port
- 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 as follows:

- Discards frames received on the port
- 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 as follows:

- Receives and forwards frames received on the port
- Forwards frames switched from another port
- 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 as follows:

- Forwards frames switched from another interface for forwarding
- Learns addresses
- Does not receive BPDUs

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.

The ML-Series card switches supported BPDUs (0x0180C2000000 and 01000CCCCCD) when they are being tunneled via the protocol tunneling feature.

STP and IEEE 802.1Q Trunks

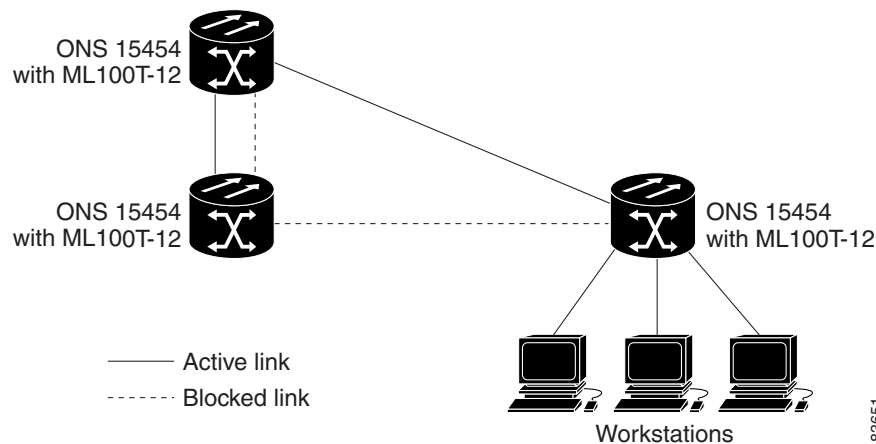
When you connect a Cisco switch to a non-Cisco device through an IEEE 802.1Q trunk, the Cisco switch uses PVST+ to provide spanning-tree interoperability. PVST+ is automatically enabled on IEEE 802.1Q trunks after users assign a protocol to a bridge group. The external spanning-tree behavior on access ports and Inter-Switch Link (ISL) trunk ports is not affected by PVST+.

For more information on IEEE 802.1Q trunks, see [Chapter 8, “Configuring VLANs.”](#)

Spanning Tree and Redundant Connectivity

You can create a redundant backbone with spanning tree by connecting two switch interfaces to another device or to two different devices. Spanning tree automatically disables one interface but enables it if the other one fails, as shown in [Figure 7-3](#). 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 7-3 Spanning Tree and Redundant Connectivity



You can also create redundant links between switches by using EtherChannel groups. For more information, see [Chapter 10, “Configuring Link Aggregation.”](#)

Accelerated Aging to Retain Connectivity

The default for aging dynamic addresses is 5 minutes, which is the default setting of the **bridge bridge-group-number 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.

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.

RSTP

RSTP provides rapid convergence of the spanning tree. 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 RSTP 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.

RSTP improves the operation of the spanning tree while maintaining backward compatibility with equipment that is based on the (original) IEEE 802.1D spanning tree.

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 2 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.

These sections describe how RSTP works:

- [Supported RSTP Instances, page 7-9](#)
- [Port Roles and the Active Topology, page 7-9](#)
- [Rapid Convergence, page 7-10](#)
- [Synchronization of Port Roles, page 7-12](#)
- [Bridge Protocol Data Unit Format and Processing, page 7-13](#)
- [Topology Changes, page 7-14](#)

Supported RSTP Instances

The ML Series supports per-VLAN rapid spanning tree (PVRST) and a maximum of 255 rapid spanning-tree instances.

Port Roles and the Active Topology

The RSTP provides rapid convergence of the spanning tree by assigning port roles and by determining 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 [“Election of the Root Switch” section on page 7-3](#). 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 IEEE 802.1D). The port state controls the operation of the forwarding and learning processes. [Table 7-3](#) provides a comparison of IEEE 802.1D and RSTP port states.

Table 7-3 Port State Comparison

Operational Status	STP Port State	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



Caution

STP edge ports are bridge ports that do not need STP enabled, where loop protection is not needed out of that port or an STP neighbor does not exist out of that port. For RSTP, it is important to disable STP on edge ports, which are typically front-side Ethernet ports, using the command **bridge bridge-group-number spanning-disabled** on the appropriate interface. If RSTP is not disabled on edge ports, convergence times will be excessive for packets traversing those ports.



Note

To be consistent with Cisco STP implementations, [Table 7-3](#) describes 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 switch, a switch port, or a LAN. It provides rapid convergence for new root ports, and ports connected through point-to-point links as follows:

- 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 7-4](#), 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.

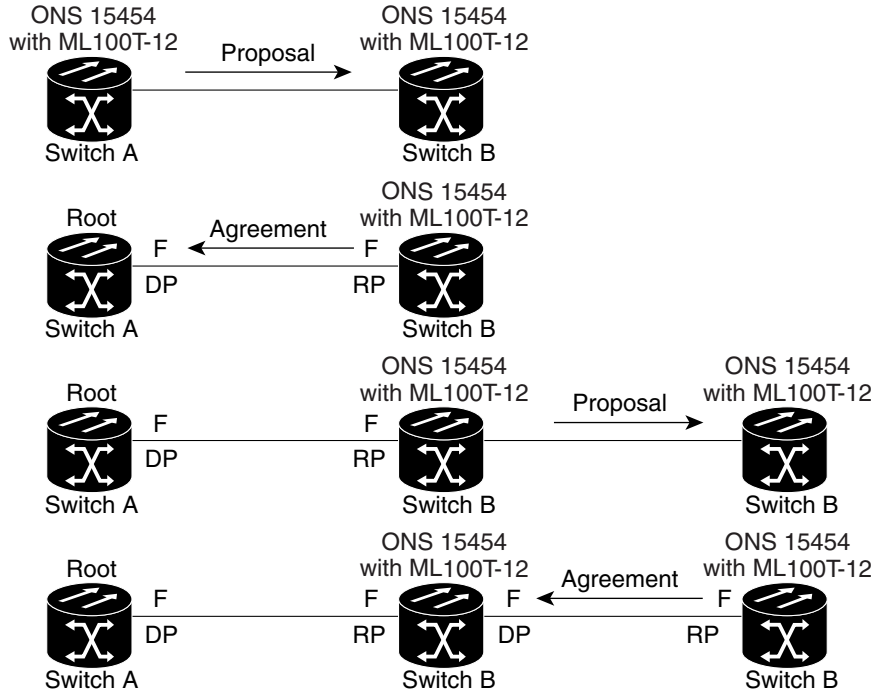
After receiving the proposal message, Switch B selects as its new root port the port from which the proposal message was received, forces all non edge ports to the blocking state, and sends an agreement message (a BPDU with the agreement flag set) through its new root port.

After receiving an agreement message from Switch B, 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 non edge 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 determines 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.

Figure 7-4 Proposal and Agreement Handshaking for Rapid Convergence



DP = designated port
 RP = root port
 F = forwarding

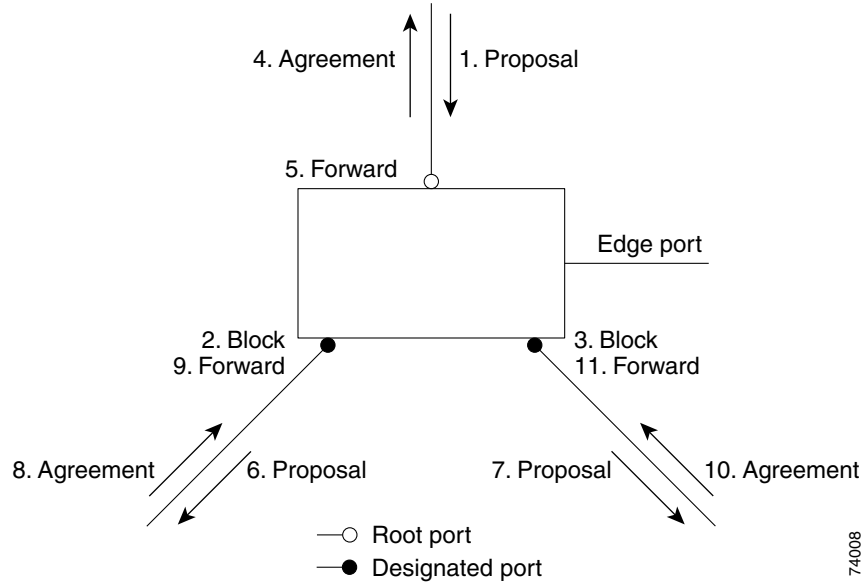
92178

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.

If a designated port is in the forwarding state, 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 7-5](#).

Figure 7-5 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 Length field is set to zero, which means that no version 1 protocol information is present. [Table 7-4](#) shows the RSTP flag fields.

Table 7-4 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

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 IEEE 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, etc.) 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 IEEE 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, etc.) 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 non edge ports.
- **Notification**—Unlike IEEE 802.1D, which uses TCN BPDUs, the RSTP does not use them. However, for IEEE 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 IEEE 802.1D configuration BPDU with the topology change acknowledgement bit set. However, if the TC-while timer (the same as the topology-change timer in IEEE 802.1D) is active on a root port connected to an IEEE 802.1D switch and a configuration BPDU with the topology change acknowledgement 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 topology change acknowledgement bit set.
- **Propagation**—When an RSTP switch receives a TC message from another switch through a designated or root port, it propagates the topology change to all of its non edge, edge, designated ports, and 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 timer is started (which 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.

Interoperability with IEEE 802.1D STP

A switch running RSTP 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.

However, the switch does not automatically revert to the RSTP mode if it no longer receives IEEE 802.1D BPDUs because it cannot determine 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.

Configuring STP and RSTP Features

These sections describe how to configure spanning-tree features:

- [Default STP and RSTP Configuration, page 7-16](#)
- [Disabling STP and RSTP, page 7-16](#)
- [Configuring the Root Switch, page 7-17](#)
- [Configuring the Port Priority, page 7-17](#)
- [Configuring the Path Cost, page 7-18](#)
- [Configuring the Switch Priority of a Bridge Group, page 7-19](#)
- [Configuring the Hello Time, page 7-19](#)
- [Configuring the Forwarding-Delay Time for a Bridge Group, page 7-20](#)
- [Configuring the Maximum-Aging Time for a Bridge Group, page 7-20](#)

Default STP and RSTP Configuration

Table 7-5 shows the default STP and RSTP configuration.

Table 7-5 Default STP and RSTP Configuration

Feature	Default Setting
Enable state	Up to 255 spanning-tree instances can be enabled.
Switch priority	32768 + Bridge ID
Spanning-tree port priority (configurable on a per-interface basis—used on interfaces configured as Layer 2 access ports)	128
Spanning-tree port cost (configurable on a per-interface basis)	1000 Mbps: 4 100 Mbps: 19 10 Mbps: 100 STS-1: 37 STS-3c: 14 STS-6c: 9 STS-9c: 7 STS-12c: 6 STS-24c: 3
Hello time	2 seconds
Forward-delay time	15 seconds
Maximum-aging time	20 seconds

Disabling STP and RSTP

STP is enabled by default on native VLAN 1 and on all newly created VLANs up to the specified spanning-tree limit of 255. Disable STP only if you are sure there are no loops in the network topology.



Caution

STP edge ports are bridge ports that do not need STP enabled, where loop protection is not needed out of that port or an STP neighbor does not exist out of that port. For RSTP, it is important to disable STP on edge ports, which are typically front-side Ethernet ports, using the command **bridge bridge-group-number spanning-disabled** on the appropriate interface. If RSTP is not disabled on edge ports, convergence times will be excessive for packets traversing those ports.



Caution

When STP 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 STP or RSTP on a per-VLAN basis:

	Command	Purpose
Step 1	Router# configure terminal	Enters the global configuration mode.
Step 2	Router(config)# interface <i>interface-id</i>	Enters the interface configuration mode.
Step 3	Router(config-if)# bridge-group <i>bridge-group-number</i> spanning disabled	Disables STP or RSTP on a per-interface basis.
Step 4	Router(config-if)# end	Returns to privileged EXEC mode.

To reenable STP, use the **no bridge-group** *bridge-group-number* **spanning disabled** interface-level configuration command.

Configuring the 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.



Note

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 bridge ID is greater than the priority of the connected switches that are running older software.

Configuring the Port Priority

If a loop occurs, spanning tree uses the port priority when selecting an interface to put into the forwarding state. You can assign higher priority values (lower numerical values) to interfaces 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, 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 port priority of an interface:

	Command	Purpose
Step 1	Router# configure terminal	Enters the global configuration mode.
Step 2	Router(config)# interface <i>interface-id</i>	Enters the interface configuration mode, and specifies an interface to configure. Valid interfaces include physical interfaces and port-channel logical interfaces (port-channel <i>port-channel-number</i>).

	Command	Purpose
Step 3	Router(config-if)# bridge-group <i>bridge-group-number priority-value</i>	Configures the port priority for an interface that is an access port. For the <i>priority-value</i> , the range is 0 to 255; the default is 128 in increments of 16. The lower the number, the higher the priority.
Step 4	Router(config-if)# end	Return to privileged EXEC mode.

To return the interface to its default setting, use the **no bridge-group id** *bridge-group-number priority-value* command.

Configuring the Path Cost

The spanning-tree path cost default value is derived from the media speed of an interface. 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 to interfaces that you want selected last. If all interfaces 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:

	Command	Purpose
Step 1	Router# configure terminal	Enters the global configuration mode.
Step 2	Router(config)# interface <i>interface-id</i>	Enters the interface configuration mode and specifies an interface to configure. Valid interfaces include physical interfaces and port-channel logical interfaces (port-channel <i>port-channel-number</i>).
Step 3	Router(config-if)# bridge-group <i>bridge-group-number path-cost</i> <i>cost</i>	Configures the cost for an interface that is an access port. 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 <i>cost</i> , the range is 0 to 65535; the default value is derived from the media speed of the interface.
Step 4	Router(config-if)# end	Returns to the privileged EXEC mode.



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 the interface to its default setting, use the **no bridge-group** *bridge-group-number path-cost cost* command.

Configuring the Switch Priority of a Bridge Group

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

Beginning in privileged EXEC mode, follow these steps to configure the switch priority of a bridge group:

	Command	Purpose
Step 1	Router# configure terminal	Enters the global configuration mode.
Step 2	Router(config)# bridge <i>bridge-group-number</i> priority <i>priority</i>	Configures the switch priority of a bridge group. For <i>priority</i> , 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. The value entered is rounded to the lower multiple of 4096. The actual number is computed by adding this number to the bridge group number.
Step 3	Router(config)# end	Return to the privileged EXEC mode.

To return the switch to its default setting, use the **no bridge** *bridge-group-number* **priority** *priority* 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.

Beginning in privileged EXEC mode, follow these steps to configure the hello time of a bridge group:

	Command	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# bridge <i>bridge-group-number</i> hello-time <i>seconds</i>	Configures the hello time of a bridge group. 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 <i>seconds</i> , the range is 1 to 10; the default is 2.
Step 3	Router(config)# end	Returns to privileged EXEC mode.

To return the switch to its default setting, use the **no bridge** *bridge-group-number* **hello-time** *seconds* command.

Configuring the Forwarding-Delay Time for a Bridge Group

Beginning in privileged EXEC mode, follow these steps to configure the forwarding-delay time for a bridge group:

	Command	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# bridge <i>bridge-group-number</i> forward-time <i>seconds</i>	Configures the forward time of a VLAN. 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 <i>seconds</i> , the range is 4 to 200; the default is 15.
Step 3	Router(config)# end	Returns to privileged EXEC mode.

To return the switch to its default setting, use the **no bridge** *bridge-group-number* **forward-time** *seconds* command.

Configuring the Maximum-Aging Time for a Bridge Group

Beginning in privileged EXEC mode, follow these steps to configure the maximum-aging time for a bridge group:

	Command	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# bridge <i>bridge-group-number</i> max-age <i>seconds</i>	Configures the maximum-aging time of a bridge group. The maximum-aging time is the number of seconds a switch waits without receiving spanning-tree configuration messages before attempting a reconfiguration. For <i>seconds</i> , the range is 6 to 200; the default is 20.
Step 3	Router(config)# end	Returns to privileged EXEC mode.

To return the switch to its default setting, use the **no bridge** *bridge-group-number* **max-age** *seconds* command.

Verifying and Monitoring STP and RSTP Status

To display the STP or RSTP status, use one or more of the privileged EXEC commands in [Table 7-6](#):

Table 7-6 Commands for Displaying Spanning-Tree Status

Command	Purpose
ML_Series# show spanning-tree	Displays detailed STP or RSTP information.
ML_Series# show spanning-tree brief	Displays summary of STP or RSTP information.

Table 7-6 *Commands for Displaying Spanning-Tree Status (continued)*

Command	Purpose
ML_Series# show spanning-tree interface <i>interface-id</i>	Displays STP or RSTP information for the specified interface.
ML_Series# show spanning-tree summary [totals]	Displays a summary of port states or displays the total lines of the STP or RSTP state section.

**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.

Examples of the **show spanning-tree** privileged EXEC command commands are shown here:

Example 7-1 *show spanning-tree Commands*

```
Router# show spanning-tree brief
```

```
Bridge group 1
Spanning tree enabled protocol ieee
Root ID    Priority    32769
           Address    0005.9a39.6634
           This bridge is the root
           Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

Bridge ID  Priority    32769 (priority 32768 sys-id-ext 1)
Address    0005.9a39.6634
Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
Aging Time 300

Interface      Role Sts Cost      Prio.Nbr Type
-----
Fa0             Desg FWD 19         128.3   P2p
PO0             Desg FWD 3          128.20  P2p
```

```
Router# show spanning-tree detail
```

```
Bridge group 1 is executing the ieee compatible Spanning Tree protocol
Bridge Identifier has priority 32768, sysid 1, address 0005.9a39.6634
Configured hello time 2, max age 20, forward delay 15
We are the root of the spanning tree
Topology change flag not set, detected flag not set
Number of topology changes 2 last change occurred 00:16:45 ago
from POS0
Times: hold 1, topology change 35, notification 2
hello 2, max age 20, forward delay 15
Timers: hello 0, topology change 0, notification 0, aging 300

Port 3 (FastEthernet0) of Bridge group 1 is forwarding
Port path cost 19, Port priority 128, Port Identifier 128.3.
Designated root has priority 32769, address 0005.9a39.6634
Designated bridge has priority 32769, address 0005.9a39.6634
Designated port id is 128.3, designated path cost 0
Timers: message age 0, forward delay 0, hold 0
Number of transitions to forwarding state: 1
Link type is point-to-point by default
BPDU: sent 641, received 0
```

```

Port 20 (POS0) of Bridge group 1 is forwarding
  Port path cost 3, Port priority 128, Port Identifier 128.20.
  Designated root has priority 32769, address 0005.9a39.6634
  Designated bridge has priority 32769, address 0005.9a39.6634
  Designated port id is 128.20, designated path cost 0
  Timers: message age 0, forward delay 0, hold 0
  Number of transitions to forwarding state: 6
  Link type is point-to-point by default
  BPDU: sent 582, received 15

```

```
Router# show spanning-tree interface fast 0
```

Bridge Group	Role	Sts	Cost	Prio.Nbr	Type
Bridge group 1	Desg	FWD	19	128.3	P2p

```
Router# show spanning-tree interface pos 0
```

Bridge Group	Role	Sts	Cost	Prio.Nbr	Type
Bridge group 1	Desg	FWD	3	128.20	P2p

```
Router# show spanning-tree summary totals
```

```
Switch is in pvst mode
Root bridge for: Bridge group 1
```

Name	Blocking	Listening	Learning	Forwarding	STP Active
1 bridge	0	0	0	2	2



Configuring VLANs

This chapter describes VLAN configurations for the ML-Series card. It describes how to configure IEEE 802.1Q VLAN encapsulation. For more information about the Cisco IOS commands used in this chapter, refer to the *Cisco IOS Command Reference* publication.

This chapter contains the following major sections:

- [Understanding VLANs, page 8-1](#)
- [Configuring IEEE 802.1Q VLAN Encapsulation, page 8-2](#)
- [IEEE 802.1Q VLAN Configuration, page 8-3](#)
- [Monitoring and Verifying VLAN Operation, page 8-5](#)



Note

Configuring VLANs is optional. Complete general interface configurations before proceeding with configuring VLANs as an optional step.

Understanding VLANs

VLANs enable network managers to group users logically rather than by physical location. A VLAN is an emulation of a standard LAN that allows secure intra-group data transfer and communication to occur without the traditional restraints placed on the network. It can also be considered a broadcast domain set up within a switch. With VLANs, switches can support more than one subnet (or VLAN) on each switch and give routers and switches the opportunity to support multiple subnets on a single physical link. A group of devices that belong to the same VLAN, but are part of different LAN segments, are configured to communicate as if they were part of the same LAN segment.

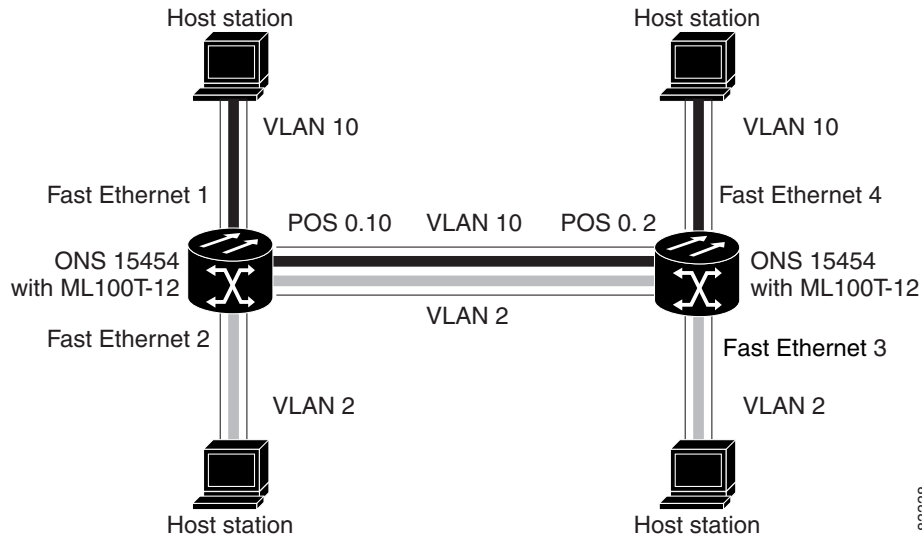
VLANs enable efficient traffic separation and provide excellent bandwidth utilization. VLANs also alleviate scaling issues by logically segmenting the physical LAN structure into different subnetworks so that packets are switched only between ports within the same VLAN. This can be very useful for security, broadcast containment, and accounting.

ML-Series software supports port-based VLANs and VLAN trunk ports, which are ports that carry the traffic of multiple VLANs. Each frame transmitted on a trunk link is tagged as belonging to only one VLAN.

ML-Series software supports VLAN frame encapsulation through the IEEE 802.1Q standard on both the ML100T-12 and the ML1000-2. The Cisco Inter-Switch Link (ISL) VLAN frame encapsulation is not supported. ISL frames are broadcast at Layer 2 or dropped at Layer 3.

ML-Series switching supports up to 900 VLAN subinterfaces per card (for example, 200 VLANs on four interfaces uses 800 VLAN subinterfaces). A maximum of 255 logical VLANs can be bridged per card (limited by the number of bridge-groups). Each VLAN subinterface can be configured for any VLAN ID in the full 1 to 4095 range. Figure 8-1 shows a network topology in which two VLANs span two ONS 15454s with ML-Series cards.

Figure 8-1 VLANs Spanning Devices in a Network



Configuring IEEE 802.1Q VLAN Encapsulation

You can configure IEEE 802.1Q VLAN encapsulation on either type of ML-Series card interfaces, Ethernet or Packet over SONET/SDH (POS). VLAN encapsulation is not supported on POS interfaces configured with HDLC encapsulation.

The native VLAN is always VLAN ID 1 on ML-Series cards. Frames on the native VLAN are normally transmitted and received untagged. On a trunk port, all frames from VLANs other than the native VLAN are transmitted and received tagged.

To configure VLANs using IEEE 802.1Q VLAN encapsulation, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# bridge <i>bridge-group-number</i> protocol <i>type</i>	Assigns a bridge group (VLAN) number and define the appropriate spanning tree type.
Step 2	Router(config)# interface <i>type number</i>	Enters interface configuration mode to configure the interface.
Step 3	Router(config-if)# no ip address	Disables IP processing.
Step 4	Router(config)# interface <i>type number.subinterface-number</i>	Enters subinterface configuration mode to configure the subinterface.

	Command	Purpose
Step 5	Router(config-subif)# encap dot1q <i>vlan-number</i>	Sets the encapsulation on the VLAN to IEEE 802.1Q.
Step 6	Router(config-subif)# bridge-group <i>bridge-group-number</i>	Assigns a network interface to a bridge group.
Step 7	Router(config-subif)# end	Returns to privileged EXEC mode.
Step 8	Router# copy running-config startup-config	(Optional) Saves your configuration changes to NVRAM.

**Note**

In a bridge group on the ML-Series card, the VLAN ID does not have to be uniform across interfaces that belong to that bridge group. For example, a bridge-group can connect from a VLAN ID subinterface to a subinterface with a different VLAN ID, and then frames entering with one VLAN ID can be changed to exit with a different VLAN ID. This is known as VLAN translation.

**Note**

IP routing is enabled by default. To enable bridging, enter the **no ip routing** or **bridge IRB** command.

**Note**

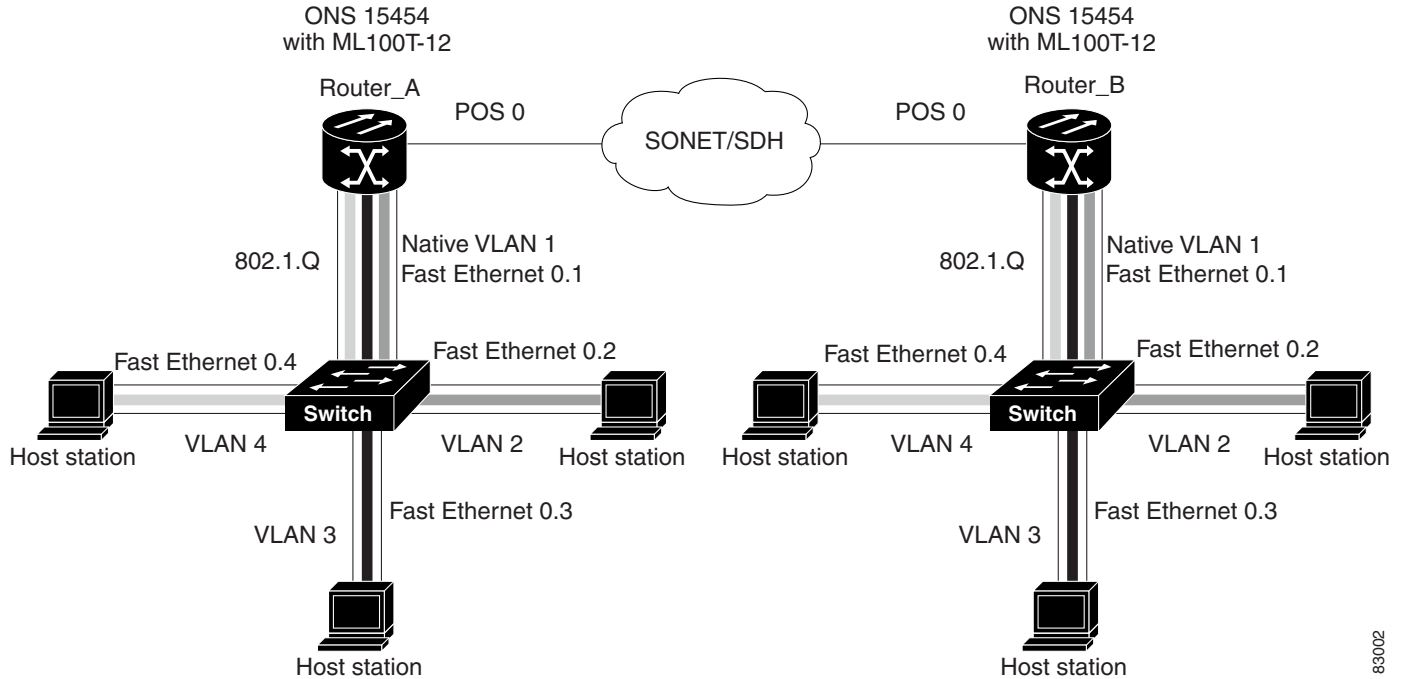
Native VLAN frames transmitted on the interface are normally untagged. All untagged frames received on the interface are associated with the native VLAN, which is always VLAN 1. Use the command **encapsulation dot1q 1 native**.

IEEE 802.1Q VLAN Configuration

The VLAN configuration example for the ML100T-12 shown in [Figure 8-2](#) depicts the following VLANs:

- Fast Ethernet subinterface 0.1 is in the IEEE 802.1Q native VLAN 1.
- Fast Ethernet subinterface 0.2 is in the IEEE 802.1Q VLAN 2.
- Fast Ethernet subinterface 0.3 is in the IEEE 802.1Q VLAN 3.
- Fast Ethernet subinterface 0.4 is in the IEEE 802.1Q VLAN 4.

Figure 8-2 Bridging IEEE 802.1Q VLANs



83002

[Example 8-1](#) shows how to configure VLANs for IEEE 802.1Q VLAN encapsulation. Use this configuration for both router A and router B. The example is shown in [Figure 8-2](#):

Example 8-1 Configure VLANs for IEEE 8021Q VLAN Encapsulation

```
bridge 1 protocol ieee
bridge 2 protocol ieee
bridge 3 protocol ieee
bridge 4 protocol ieee
!
!
interface FastEthernet0
no ip address
!
interface FastEthernet0.1
encapsulation dot1Q 1 native
bridge-group 1
!
interface FastEthernet0.2
encapsulation dot1Q 2
bridge-group 2
!
interface FastEthernet0.3
encapsulation dot1Q 3
bridge-group 3
!
interface FastEthernet0.4
encapsulation dot1Q 4
bridge-group 4
!
interface POS0
no ip address
crc 32
```

```
pos flag c2 1
!
interface POS0.1
 encapsulation dot1Q 1 native
 bridge-group 1
!
interface POS0.2
 encapsulation dot1Q 2
 bridge-group 2
!
interface POS0.3
 encapsulation dot1Q 3
 bridge-group 3
!
interface POS0.4
 encapsulation dot1Q 4
 bridge-group 4
```

Monitoring and Verifying VLAN Operation

After the VLANs are configured on the ML-Series card, you can monitor their operation by entering the privileged EXEC command **show vlans** *vlan-id*. This command displays information on all configured VLANs or on a specific VLAN (by VLAN ID number).



Caution

Two similar commands exist. The command **show vlans** gives information regarding IEEE 802.1Q VLANs configured on the ML-Series card. The command **show vlan** gives information regarding the VLAN tunnel. For more information on VLAN tunneling, see [Chapter 9, “Configuring IEEE 802.1Q Tunneling and Layer 2 Protocol Tunneling.”](#)



Configuring IEEE 802.1Q Tunneling and Layer 2 Protocol Tunneling

Virtual private networks (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 ML-Series cards support IEEE 802.1Q tunneling and Layer 2 protocol tunneling.

This chapter contains the following sections:

- [Understanding IEEE 802.1Q Tunneling, page 9-1](#)
- [Configuring IEEE 802.1Q Tunneling, page 9-4](#)
- [Understanding VLAN-Transparent and VLAN-Specific Services, page 9-6](#)
- [Understanding Layer 2 Protocol Tunneling, page 9-9](#)
- [Configuring Layer 2 Protocol Tunneling, page 9-10](#)

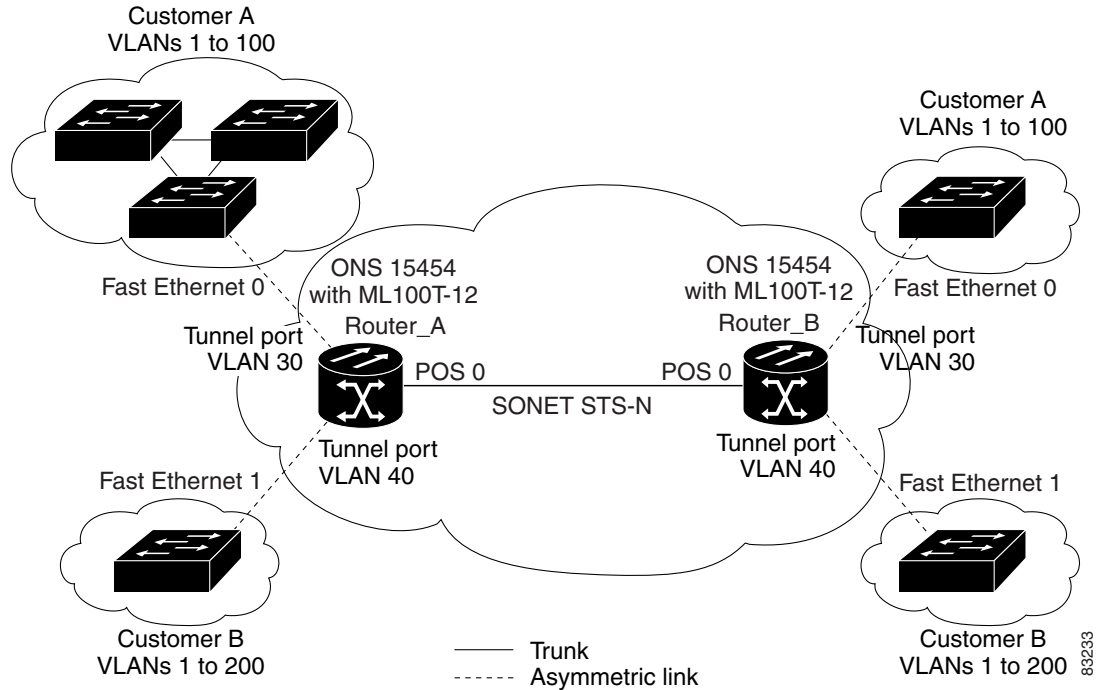
Understanding IEEE 802.1Q Tunneling

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 IEEE 802.1Q specification VLAN limit of 4096.

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

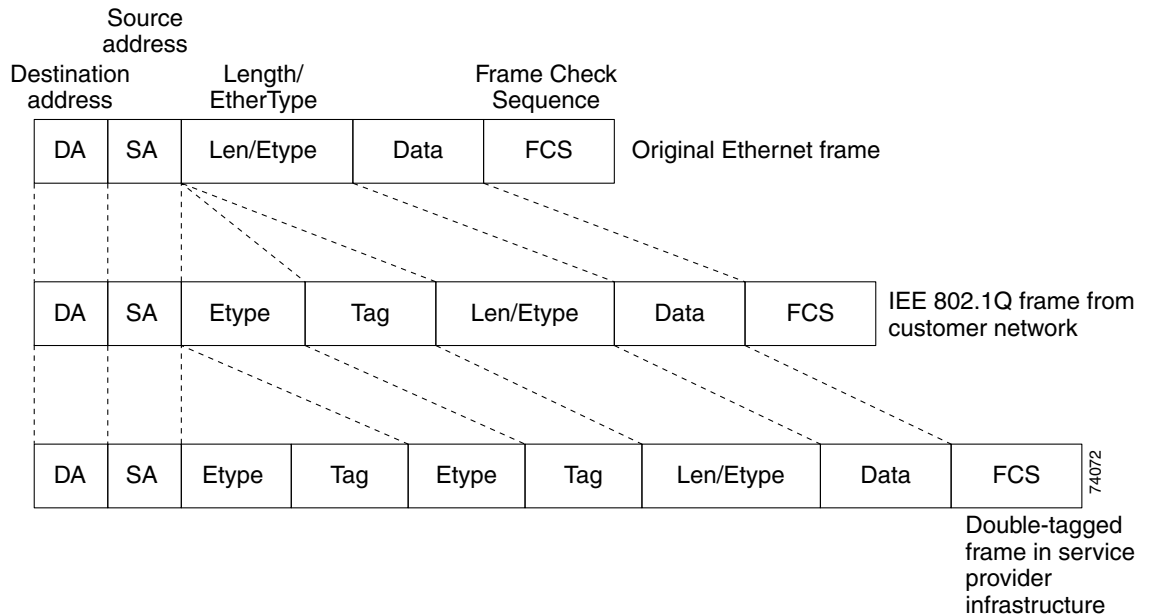
Customer traffic tagged in the normal way with appropriate VLAN IDs comes from an IEEE 802.1Q trunk port on the customer device and into a tunnel port on the ML-Series card. The link between the customer device and the ML-Series card is an asymmetric link because one end is configured as an IEEE 802.1Q trunk port and the other end is configured as a tunnel port. You assign the tunnel port interface to an access VLAN ID unique to each customer ([Figure 9-1](#)).

Figure 9-1 IEEE 802.1Q Tunnel Ports in a Service-Provider Network



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

When the double-tagged packet enters another trunk port in a service provider ML-Series card, the outer tag is stripped as the packet is processed inside the switch. When the packet exits another trunk port on the same core switch, the same metro tag is again added to the packet. Figure 9-2 shows the structure of the double-tagged packet.

Figure 9-2 Normal, IEEE 802.1Q, and IEEE 802.1Q-Tunneled Ethernet Packet Formats

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

In [Figure 9-1 on page 9-2](#), Customer A was assigned VLAN 30, and Customer B was assigned VLAN 40. Packets entering the ML-Series card tunnel ports with IEEE 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. With IEEE 802.1Q tunneling, 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. If the traffic coming from a customer network is not tagged (native VLAN frames), these packets are bridged or routed as if they were normal packets, and the metro tag is added (as a single-level tag) when they exit toward the service provider network.

If the native VLAN (VLAN 1) is used in the service provider network as a metro tag, this tag must always be added to the customer traffic, even though the native VLAN ID is not normally added to transmitted frames. If the VLAN 1 metro tag is not added on frames entering the service provider network, then the customer VLAN tag appears to be the metro tag, with disastrous results. The global configuration **vlan dot1q tag native** command must be used to prevent this by forcing a tag to be added to VLAN 1. Avoiding the use of VLAN 1 as a metro tag transporting customer traffic is recommended to reduce the risk of misconfiguration. A best practice is to use VLAN 1 as a private management VLAN in the service provider network.

The IEEE 802.1Q class of service (COS) priority field on the added metro tag is set to zero by default, but can be modified by input or output policy maps.

Configuring IEEE 802.1Q Tunneling

This section includes the following information about configuring IEEE 802.1Q tunneling:

- [IEEE 802.1Q Tunneling and Compatibility with Other Features, page 9-4](#)
- [Configuring an IEEE 802.1Q Tunneling Port, page 9-4](#)
- [IEEE 802.1Q Example, page 9-5](#)



Note

By default, IEEE 802.1Q tunneling is not configured on the ML-Series.

IEEE 802.1Q Tunneling and Compatibility with Other Features

Although IEEE 802.1Q tunneling works well for Layer 2 packet switching, there are incompatibilities with some Layer 2 features and with Layer 3 switching:

- A tunnel port cannot be a routed port.
- Tunnel ports do not support IP access control lists (ACLs).
- Layer 3 quality of service (QoS) ACLs and other QoS features related to Layer 3 information are not supported on tunnel ports. MAC-based QoS is supported on tunnel ports.
- EtherChannel port groups are compatible with tunnel ports as long as the IEEE 802.1Q configuration is consistent within an EtherChannel port group.
- Port Aggregation Protocol (PAgP) and Unidirectional Link Detection (UDLD) Protocol are not supported on IEEE 802.1Q tunnel ports.
- Dynamic Trunking Protocol (DTP) is not compatible with IEEE 802.1Q tunneling because you must manually configure asymmetric links with tunnel ports and trunk ports.
- Loopback detection is supported on IEEE 802.1Q tunnel ports.
- When a port is configured as an IEEE 802.1Q tunnel port, spanning tree bridge protocol data unit (BPDU) filtering is automatically disabled on the interface.

Configuring an IEEE 802.1Q Tunneling Port

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

	Command	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# bridge <i>bridge-number</i> protocol <i>bridge-protocol</i>	Creates a bridge number and specifies a protocol.
Step 3	Router(config)# interface fastEthernet <i>number</i>	Enters the interface configuration mode and 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. Valid interfaces include physical interfaces and port-channel logical interfaces (port channels 1 to 64).

	Command	Purpose
Step 4	Router(config-if)# bridge-group <i>number</i>	Assigns the tunnel port to a bridge-group. All traffic from the port (tagged and untagged) will be switched based on this bridge-group. Other members of the bridge-group should be VLAN subinterfaces on a provider trunk interface.
Step 5	Router(config-if)# mode dot1q-tunnel	Sets the interface as an IEEE 802.1Q tunnel port.
Step 6	Router(config)# end	Returns to privileged EXEC mode.
Step 7	Router# show dot1q-tunnel	Displays the tunnel ports on the switch.
Step 8	Router# copy running-config startup-config	(Optional) Saves your entries in the configuration file.



Note The VLAN ID (VID) range of 2 to 4095 is recommended for IEEE 802.1Q tunneling on the ML-Series card.



Note If VID 1 is required to be used as a metro tag, use the following command:

```
Router (config)# VLAN dot1q tag native
```

Use the **no mode dot1q-tunnel** interface configuration command to remove the IEEE 802.1Q tunnel from the interface.

IEEE 802.1Q Example

The following examples show how to configure the example in [Figure 9-1 on page 9-2](#). [Example 9-1](#) applies to Router A, and [Example 9-2](#) applies to Router B.

Example 9-1 Router A Configuration

```
bridge 30 protocol ieee
bridge 40 protocol ieee
!
!
interface FastEthernet0
no ip routing
no ip address
mode dot1q-tunnel
bridge-group 30
!
interface FastEthernet1
no ip address
mode dot1q-tunnel
bridge-group 40
!
interface POS0
no ip address
crc 32
pos flag c2 1
!
interface POS0.1
encapsulation dot1Q 30
bridge-group 30
```

```
!
interface POS0.2
 encapsulation dot1Q 40
 bridge-group 40
```

Example 9-2 Router B Configuration

```
bridge 30 protocol ieee
bridge 40 protocol ieee
!
!
interface FastEthernet0
no ip routing
no ip address
mode dot1q-tunnel
bridge-group 30
!
interface FastEthernet1
no ip address
mode dot1q-tunnel
bridge-group 40
!
interface POS0
no ip address
crc 32
pos flag c2 1
!
interface POS0.1
 encapsulation dot1Q 30
 bridge-group 30
!
interface POS0.2
 encapsulation dot1Q 40
 bridge-group 40
```

Understanding VLAN-Transparent and VLAN-Specific Services

The ML-Series card supports combining VLAN-transparent services and one or more VLAN-specific services on the same port. All of these VLAN-transparent and VLAN-specific services can be point-to-point or multipoint-to-multipoint.

This allows a service provider to combine a VLAN-transparent service, such as IEEE 802.1Q tunneling (QinQ), with VLAN-specific services, such as bridging specific VLANs, on the same customer port. For example, one customer VLAN can connect to Internet access and the other customer VLANs can be tunneled over a single provider VLAN to another customer site, all over a single port at each site.

[Table 9-1](#) outlines the differences between VLAN-transparent and VLAN-specific services.

Table 9-1 VLAN-Transparent Service Versus VLAN-Specific Services

VLAN-Transparent Services	VLAN-Specific Services
Bridging only	Bridging or routing
One service per port	Up to 254 VLAN-specific services per port
Applies indiscriminately to all VLANs on the physical interface	Applies only to specified VLANs

**Note**

VLAN-transparent service is also referred to as Ethernet Wire Service (EWS). VLAN-specific service is also referred to as QinQ tunneling trunk UNI in Metro Ethernet terminology.

A VLAN-specific service on a subinterface coexists with the VLAN-transparent service, often IEEE 802.1Q tunneling, on a physical interface. VLANs configured for a VLAN-transparent service and a VLAN-specific service follow the VLAN-specific service configuration. If you need to configure 802.1Q tunneling, configure this VLAN-transparent service in the normal manner, see the “[Configuring IEEE 802.1Q Tunneling](#)” section on page 9-4.

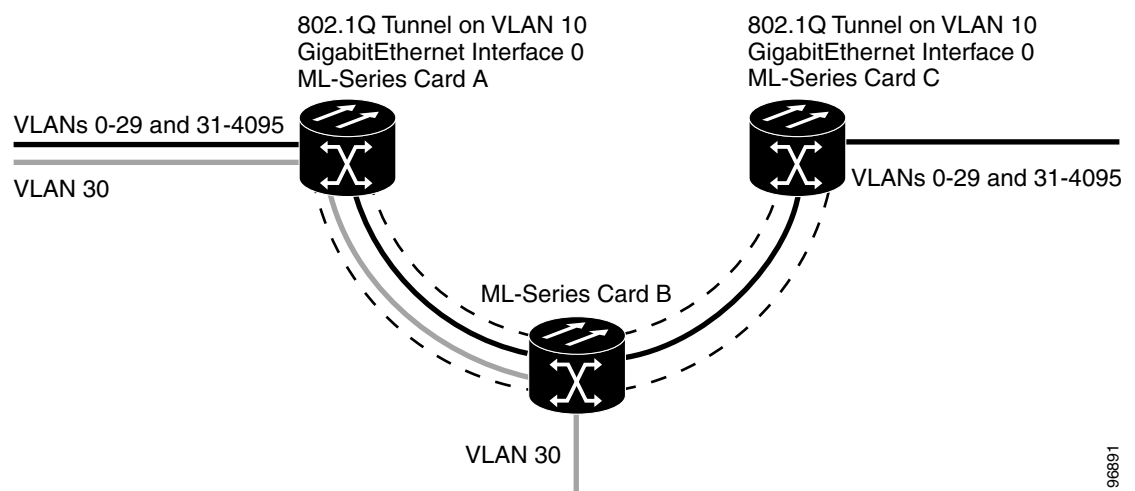
A VLAN-specific service can be any service normally applicable to a VLAN. To configure an ERMS VLAN-specific service, configure the service in the normal manner.

VLAN-Transparent and VLAN-Specific Services Configuration Example

In this example, the Gigabit Ethernet interface 0 on both the ML-Series card A and ML-Series card C are the trunk ports in an IEEE 802.1Q tunnel, a VLAN-transparent service. VLAN 10 is used for the VLAN-transparent service, which would normally transport all customer VLANs on the ML-Series card A’s Gigabit Ethernet interface 0. All unspecified VLANs and VLAN 1 would also be tunneled across VLAN 10.

VLAN 30 is prevented from entering the VLAN-transparent service and is instead forwarded on a specific-VLAN service, bridging Gigabit Ethernet interface 0 on ML-Series card A and Gigabit Ethernet interface 0 on ML-Series card B. [Figure 9-3](#) is used as an example to performing configuration examples 9-3, 9-4, and 9-5.

Figure 9-3 ERMS Example



[Example 9-3](#) applies to ML-Series card A.

Example 9-3 ML-Series Card A Configuration

```
hostname ML-A
bridge 10 protocol rstp
```

```

bridge 30 protocol ieee
!
!
interface GigabitEthernet0
  no ip address
  no ip route-cache
  mode dot1q-tunnel
  bridge-group 10
  bridge-group 10 spanning-disabled
!
interface GigabitEthernet0.3
  encapsulation dot1Q 30
  no ip route-cache
  bridge-group 30

!
interface POS0
  no ip address
  no ip route-cache
  crc 32
!
interface POS0.1
  encapsulation dot1Q 10
  no ip route-cache
  bridge-group 10
!
interface POS0.3
  encapsulation dot1Q 30
  no ip route-cache
  bridge-group 30

```

Example 9-4 applies to ML-Series card B.

Example 9-4 ML-Series Card B Configuration

```

hostname ML-B
!
bridge 10 protocol rstp
bridge 30 protocol ieee
!
!
interface GigabitEthernet0
  no ip address
!
interface GigabitEthernet0.3
  encapsulation dot1Q 30
  bridge-group 30
!
interface GigabitEthernet1
  no ip address
  shutdown
!
interface POS0
  no ip address
  crc 32
!
interface POS0.1
  encapsulation dot1Q 10
  bridge-group 10
!
interface POS0.3
  encapsulation dot1Q 30
  bridge-group 30

```



```

!
interface POS1
  no ip address
  crc 32
!
interface POS1.1
  encapsulation dot1Q 10
  bridge-group 10
!
interface POS1.3
  encapsulation dot1Q 30
  bridge-group 30

```

Example 9-5 applies to ML-Series card C.

Example 9-5 ML-Series Card C Configuration

```

hostname ML-C
bridge 10 protocol rstp
!
!
interface GigabitEthernet0
  no ip address
  no ip route-cache
  mode dot1q-tunnel
  bridge-group 10
  bridge-group 10 spanning-disabled
!
interface POS0
  no ip address
  no ip route-cache
  crc 32
!
interface POS0.1
  encapsulation dot1Q 10
  no ip route-cache
  bridge-group 10

```

Understanding Layer 2 Protocol Tunneling

Customers at different sites connected across a service-provider network need to run various Layer 2 protocols to scale their topology to include all remote sites, as well as the local sites. Spanning Tree Protocol (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 infrastructure. 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.

When protocol tunneling is enabled, edge switches on the inbound side of the service-provider infrastructure 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. CDP, STP, or VTP Layer 2 protocol data units (PDUs) cross the service-provider infrastructure 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 the following results:

- Users on each of a customer's sites are able to 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 through the service provider to all switches.

Layer 2 protocol tunneling can be used independently or to enhance IEEE 802.1Q tunneling. If protocol tunneling is not enabled on IEEE 802.1Q tunneling ports or on specific VLANs, 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 IEEE 802.1Q tunneling achieve complete knowledge of the customer's VLAN. If IEEE 802.1Q tunneling is not used, you can still enable Layer 2 protocol tunneling by connecting to the customer switch through access ports and enabling tunneling on the service-provider access port.

Configuring Layer 2 Protocol Tunneling

Layer 2 protocol tunneling (by protocol) is enabled on the tunnel ports or on specific tunnel VLANs that are connected to the customer by the edge switches of the service-provider network. ML-Series card tunnel ports are connected to customer IEEE 802.1Q trunk ports. The ML-Series card supports Layer 2 protocol tunneling for CDP, STP, and VTP at the interface and subinterface level. Multiple STP (MSTP) Tunneling support is achieved through subinterface protocol tunneling. The ML-Series cards connected to the customer switch perform the tunneling process.

When the Layer 2 PDUs that entered the inbound ML-Series switch through the tunnel port exit the switch 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 IEEE 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 VLAN tag. The core switches ignore the inner tags and forward the packet to all trunk ports in the same metro VLAN. The ML-Series switches on the outbound side restore the proper Layer 2 protocol and MAC address information and forward the packets. Therefore, the Layer 2 PDUs are kept intact and delivered across the service-provider infrastructure to the other side of the customer network.

This section contains the following information about configuring Layer 2 protocol tunneling:

- [Default Layer 2 Protocol Tunneling Configuration, page 9-10](#)
- [Layer 2 Protocol Tunneling Configuration Guidelines, page 9-11](#)
- [Configuring Layer 2 Tunneling on a Port, page 9-11](#)
- [Configuring Layer 2 Tunneling Per-VLAN, page 9-12](#)
- [Monitoring and Verifying Tunneling Status, page 9-12](#)

Default Layer 2 Protocol Tunneling Configuration

[Table 9-2](#) shows the default Layer 2 protocol tunneling configuration.

Table 9-2 Default Layer 2 Protocol Tunneling Configuration

Feature	Default Setting
Layer 2 protocol tunneling	Disabled for CDP, STP, and VTP.
Class of service (CoS) value	If a CoS value is configured on the interface for data packets, that value is the default used for Layer 2 PDUs. If none is configured, there is no default. This allows existing CoS values to be maintained, unless the user configures otherwise.

Layer 2 Protocol Tunneling Configuration Guidelines

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

- The ML-Series card supports Per-VLAN Protocol Tunneling (PVPT), which allows protocol tunneling to be configured and run on a specific subinterface (VLAN). PVPT configuration is done at the subinterface level.
- PVPT should be configured on VLANs that carry multi-session transport (MST) BPDUs on the connected devices.
- The ML-Series card supports tunneling of CDP, STP (including MSTP and VTP protocols). Protocol tunneling is disabled by default but can be enabled for the individual protocols on IEEE 802.1Q tunnel ports or on specific VLANs.
- Tunneling is not supported on trunk ports. If you enter the **l2protocol-tunnel** interface configuration command on a trunk port, the command is accepted, but Layer 2 tunneling does not take effect unless you change the port to a tunnel port.
- EtherChannel port groups are compatible with tunnel ports as long as the IEEE 802.1Q configuration is configured within an EtherChannel port group.
- If an encapsulated PDU (with the proprietary destination MAC address) is received from a tunnel port or access port with Layer 2 tunneling enabled, the tunnel port is shut down to prevent loops.
- 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. No CDP packets are forwarded from tunnel ports.
- Because tunneled PDUs (especially STP BPDUs) must be delivered to all remote sites for the customer virtual network to operate 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.
- Protocol tunneling has to be configured symmetrically at both the ingress and egress point. For example, if you configure the entry point to tunnel STP, CDP, VTP, then you must configure the egress point in the same way.

Configuring Layer 2 Tunneling on a Port

Beginning in privileged EXEC mode, follow these steps to configure a port as a Layer 2 tunnel port:

	Command	Purpose
Step 1	Router# configuration terminal	Enters global configuration mode.
Step 2	Router(config)# bridge <i>bridge-group-number protocol type</i>	Creates a bridge group number and specifies a protocol.
Step 3	Router(config)# l2protocol-tunnel cos <i>cos-value</i>	Associates a CoS value with the Layer 2 tunneling port. Valid numbers for a <i>cos-value</i> range from 0 to 7.
Step 4	Router(config)# interface type <i>number</i>	Enters interface configuration mode for the interface to be configured as a tunnel port.
Step 5	Router(config-if)# bridge-group <i>bridge-group-number</i>	Assigns a bridge group to the interface.
Step 6	Router(config-if)# mode dot1q tunnel	Sets the interface as an IEEE 802.1Q tunnel VLAN.
Step 7	Router(config-if)# l2protocol-tunnel { all cdp stp vtp }	Sets the interface as a Layer 2 protocol tunnel port and enables all three protocols or specifically enables CDP, STP, or VTP. These protocols are off by default.
Step 8	Router(config-if)# end	Returns to privileged EXEC mode.
Step 9	Router# show dot1q-tunnel	Displays the tunnel ports on the switch.
Step 10	Router# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Layer 2 Tunneling Per-VLAN

Beginning in privileged EXEC mode, follow these steps to configure a VLAN as a Layer 2 tunnel VLAN:

	Command	Purpose
Step 1	Router# configuration terminal	Enters global configuration mode.
Step 2	Router(config)# bridge <i>bridge-group-number protocol type</i>	Creates a bridge group number and specifies a protocol.
Step 3	Router(config)# l2protocol-tunnel cos <i>cos-value</i>	Associates a CoS value with the Layer 2 tunneling VLAN. Valid numbers for a <i>cos-value</i> range from 0 to 7.
Step 4	Router(config)# interface type <i>number.subinterface-number</i>	Enters subinterface configuration mode and the subinterface to be configured as a tunnel VLAN.
Step 5	Router(config-subif)# encapsulation dot1q <i>bridge-group-number</i>	Sets the subinterface as an IEEE 802.1Q tunnel VLAN.
Step 6	Router(config-subif)# bridge-group <i>bridge-group-number</i>	Assigns a bridge group to the interface.
Step 7	Router(config-subif)# end	Returns to privileged EXEC mode.
Step 8	Router# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring and Verifying Tunneling Status

Table 9-3 shows the privileged EXEC commands for monitoring and maintaining IEEE 802.1Q and Layer 2 protocol tunneling.

Table 9-3 **Commands for Monitoring and Maintaining Tunneling**

Command	Purpose
<code>show dot1q-tunnel</code>	Displays IEEE 802.1Q tunnel ports on the switch.
<code>show dot1q-tunnel interface <i>interface-id</i></code>	Verifies if a specific interface is a tunnel port.
<code>show l2protocol-tunnel</code>	Displays information about Layer 2 protocol tunneling ports.
<code>show vlan dot1q tag native</code>	Displays IEEE 802.1Q tunnel information.



Configuring Link Aggregation

This chapter describes how to configure link aggregation for the ML-Series cards, both EtherChannel and packet-over-SONET/SDH (POS) channel. For additional information about the Cisco IOS commands used in this chapter, refer to the *Cisco IOS Command Reference* publication.

This chapter contains the following major sections:

- [Understanding Link Aggregation, page 10-1](#)
- [Understanding Encapsulation over EtherChannel or POS Channel, page 10-7](#)
- [Monitoring and Verifying EtherChannel and POS, page 10-9](#)

Understanding Link Aggregation

The ML-Series card offers both EtherChannel and POS channel. Traditionally EtherChannel is a trunking technology that groups together multiple full-duplex IEEE 802.3 Ethernet interfaces to provide fault-tolerant high-speed links between switches, routers, and servers. EtherChannel is a logical aggregation of multiple Ethernet interfaces. EtherChannel forms a single higher bandwidth routing or bridging endpoint and was designed primarily for host-to-switch connectivity. The ML-Series card extends this link aggregation technology to bridged POS interfaces. POS channel is only supported with LEX encapsulation.

Link aggregation provides the following benefits:

- Logical aggregation of bandwidth
- Load balancing
- Fault tolerance

The EtherChannel interface, consisting of multiple Fast Ethernet, Gigabit Ethernet, or POS interfaces, is treated as a single interface, which is called a port channel. You must perform all EtherChannel configurations on the EtherChannel interface (port channel) rather than on the individual member Ethernet interfaces. You can create the EtherChannel interface by entering the **interface port-channel** interface configuration command. Each ML100T-12 supports up to 7 Fast EtherChannel (FEC) interfaces or port channels (6 Fast Ethernet and 1 POS). Each ML1000-2 supports up to 2 Gigabit EtherChannel (GEC) interfaces or port channels (1 Gigabit Ethernet and 1 POS.)

EtherChannel connections are fully compatible with IEEE 802.1Q trunking and routing technologies. IEEE 802.1Q trunking can carry multiple VLANs across an EtherChannel.

Cisco's FEC technology builds upon standards-based IEEE 802.3 full-duplex Fast Ethernet to provide a reliable high-speed solution for the campus network backbone. FEC provides bandwidth scalability within the campus by providing up to 400-Mbps full-duplex Fast Ethernet on the ML100-12.

Cisco's GEC technology provides bandwidth scalability by providing 2-Gbps full-duplex aggregate capacity on the ML1000-2.

Cisco's POS channel technology provide bandwidth scalability by providing up to 48 STSs or VC4-16c of aggregate capacity on either the ML100-12 or the ML1000-2.

**Caution**

The EtherChannel interface is the Layer 2/Layer 3 interface. Do not enable Layer 3 addresses on the physical interfaces. Do not assign bridge groups on the physical interfaces because doing so creates loops.

**Caution**

Before a physical interface is removed from an EtherChannel (port channel) interface, the physical interface must be disabled. To disable a physical interface, use the **shutdown** command in interface configuration mode.

**Note**

Link aggregation across multiple ML-Series cards is not supported.

**Note**

Policing is not supported on port channel interfaces.

**Note**

The ML-Series does not support the routing of Subnetwork Access Protocol (SNAP) or Inter-Switch Link (ISL) encapsulated frames.

Configuring EtherChannel

You can configure an FEC or a GEC by creating an EtherChannel interface (port channel) and assigning a network IP address. All interfaces that are members of a FEC or a GEC should have the same link parameters, such as duplex and speed.

To create an EtherChannel interface, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	<code>Router(config)# interface port-channel channel-number</code>	Creates the EtherChannel interface. You can configure up to 6 FECs on the ML100T-12 and 1 GEC on the ML1000-2.
Step 2	<code>Router(config-if)# ip address ip-address subnet-mask</code>	Assigns an IP address and subnet mask to the EtherChannel interface (required only for Layer 3 EtherChannel).
Step 3	<code>Router(config-if)# end</code>	Exits to privileged EXEC mode.
Step 4	<code>Router# copy running-config startup-config</code>	(Optional) Saves configuration changes to NVRAM.

For information on other configuration tasks for the EtherChannel, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide*.

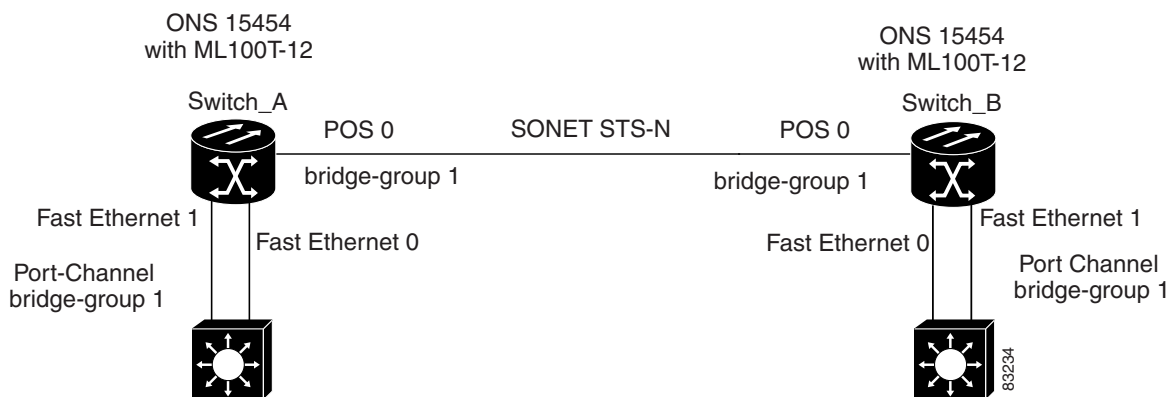
To assign Ethernet interfaces to the EtherChannel, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# interface fastethernet <i>number</i> or Router(config)# interface gigabitethernet <i>number</i>	Enters one of the interface configuration modes to configure the Fast Ethernet or Gigabit Ethernet interface that you want to assign to the EtherChannel. You can assign any Ethernet interface on the system to the EtherChannel, but both interfaces must be either FEC or GEC.
Step 2	Router(config-if)# channel-group <i>channel-number</i>	Assigns the Fast Ethernet or Gigabit Ethernet interfaces to the EtherChannel. The channel number must be the same channel number you assigned to the EtherChannel interface.
Step 3	Router(config-if)# end	Exits to privileged EXEC mode.
Step 4	Router# copy running-config startup-config	(Optional) Saves configuration changes to NVRAM.

EtherChannel Configuration Example

Figure 10-1 shows an example of encapsulation over EtherChannel. The associated commands are provided in Example 10-1 (Switch A) and Example 10-2 (Switch B).

Figure 10-1 Encapsulation over EtherChannel Example



Example 10-1 Switch A Configuration

```
hostname Switch A
!
bridge 1 protocol ieee
!
interface Port-channel 1
no ip address
```

```

bridge-group 1
hold-queue 150 in
!
interface FastEthernet 0
no ip address
channel-group 1
!
interface FastEthernet 1
no ip address
channel-group 1
!
interface POS 0
no ip routing
no ip address
crc 32
bridge-group 1
pos flag c2 1

```

Example 10-2 Switch B Configuration

```

hostname Switch B
!
bridge 1 protocol ieee
!
interface Port-channel 1
no ip routing
no ip address
bridge-group 1
hold-queue 150 in
!
interface FastEthernet 0
no ip address
channel-group 1
!
interface FastEthernet 1
no ip address
channel-group 1
!
interface POS 0
no ip address
crc 32
bridge-group 1
pos flag c2 1
!

```

Configuring POS Channel

You can configure a POS channel by creating a POS channel interface (port channel) and optionally assigning an IP address. All POS interfaces that are members of a POS channel should have the same port properties and be on the same ML-Series card.



Note

POS channel is only supported with G-Series card compatible (LEX) encapsulation.

To create a POS channel interface, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# interface port-channel <i>channel-number</i>	Creates the POS channel interface. You can configure one POS channel on the ML-Series card.
Step 2	Router(config-if)# ip address <i>ip-address</i> <i>subnet-mask</i>	Assigns an IP address and subnet mask to the POS channel interface (required only for the Layer 3 POS channel).
Step 3	Router(config-if)# end	Exits to privileged EXEC mode.
Step 4	Router# copy running-config startup-config	(Optional) Saves configuration changes to NVRAM.



Caution

The POS channel interface is the routed interface. Do not enable Layer 3 addresses on any physical interfaces. Do not assign bridge groups on any physical interfaces because doing so creates loops.

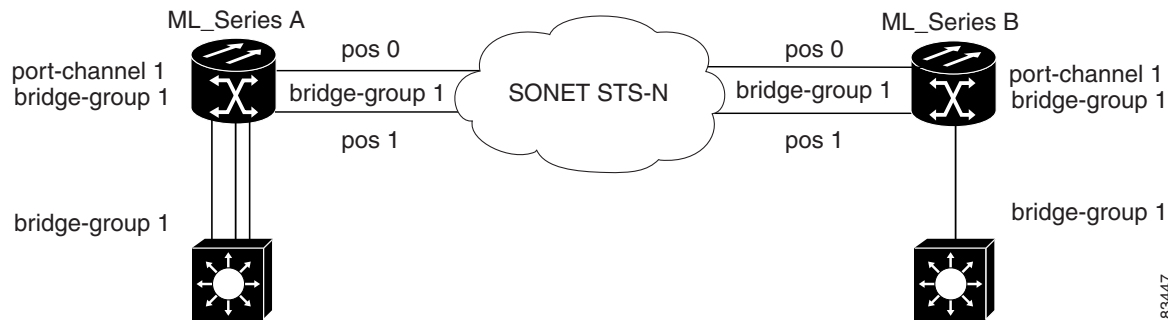
To assign POS interfaces to the POS channel, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# interface pos <i>number</i>	Enters the interface configuration mode to configure the POS interface that you want to assign to the POS channel.
Step 2	Router(config-if)# channel-group <i>channel-number</i>	Assigns the POS interface to the POS channel. The channel number must be the same channel number that you assigned to the POS channel interface.
Step 3	Router(config-if)# end	Exits to privileged EXEC mode.
Step 4	Router# copy running-config startup-config	(Optional) Saves the configuration changes to NVRAM.

POS Channel Configuration Example

Figure 10-2 shows an example of POS channel configuration. The associated code is provided in Example 10-3 (Switch A) and Example 10-4 (Switch B).

Figure 10-2 POS Channel Example



83447

Example 10-3 Switch A Configuration

```

bridge irb
bridge 1 protocol ieee
!
!
interface Port-channell
no ip address
no keepalive
bridge-group 1
!
interface FastEthernet0
no ip address
bridge-group 1
!
interface POS0
no ip address
channel-group 1
crc 32
pos flag c2 1
!
interface POS1
no ip address
channel-group 1
crc 32
pos flag c2 1

```

Example 10-4 Switch B Configuration

```

bridge irb
bridge 1 protocol ieee
!
!
interface Port-channell
no ip address
no keepalive
bridge-group 1
!
interface FastEthernet0
no ip address
bridge-group 1
!

```

```

interface POS0
  no ip address
  channel-group 1
  crc 32
pos flag c2 1
!
interface POS1
  no ip address
  channel-group 1
  crc 32
pos flag c2 1

```

Understanding Encapsulation over EtherChannel or POS Channel

When configuring encapsulation over FEC, GEC, or POS, be sure to configure IEEE 802.1Q on the port-channel interface, not its member ports. However, certain attributes of port channel, such as duplex mode, need to be configured at the member port levels. Also make sure that you do not apply protocol-level configuration (such as an IP address or a bridge group assignment) to the member interfaces. All protocol-level configuration should be on the port channel or on its subinterface. You must configure IEEE 802.1Q encapsulation on the partner system of the EtherChannel as well.

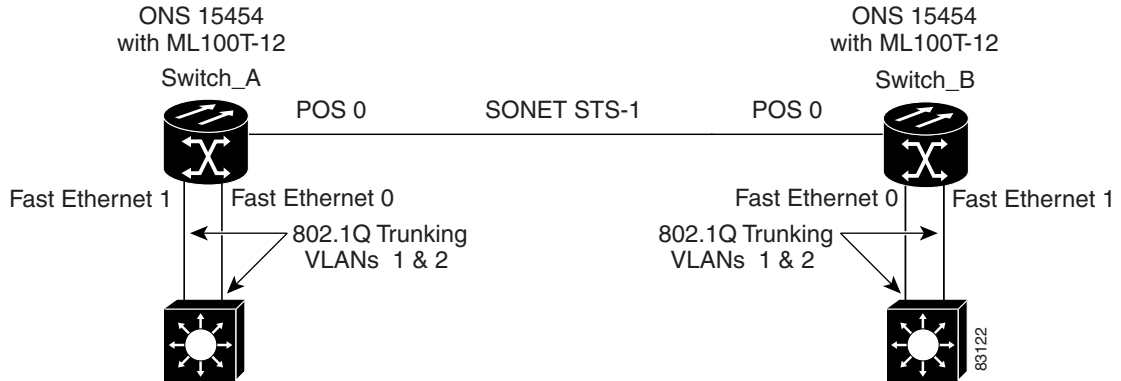
Configuring Encapsulation over EtherChannel or POS Channel

To configure encapsulation over the EtherChannel or POS channel, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# interface port-channel <i>channel-number.subinterface-number</i>	Configures the subinterface on the created port channel.
Step 2	Router(config-subif)# encapsulation dot1q <i>vlan-id</i>	Assigns the IEEE 802.1Q encapsulation to the subinterface.
Step 3	Router(config-subif)# bridge-group <i>bridge-group-number</i>	Assigns the subinterface to a bridge group.
Step 4	Router(config-subif)# end	Exits to privileged EXEC mode. Note Optionally, you can remain in interface configuration mode and enable other supported interface commands to meet your requirements.
Step 5	Router# copy running-config startup-config	(Optional) Saves the configuration changes to NVRAM.

Encapsulation over EtherChannel Example

Figure 10-3 shows an example of encapsulation over EtherChannel. The associated code is provided in Example 10-5 (Switch A) and Example 10-6 (Switch B).

Figure 10-3 Encapsulation over EtherChannel Example

This encapsulation over EtherChannel example shows how to set up two ONS 15454s with ML100T-12 cards (Switch A and Switch B) to interoperate with two switches that also support IEEE 802.1Q encapsulation over EtherChannel. To set up this example, use the configurations in the following sections for both Switch A and Switch B.

Example 10-5 Switch A Configuration

```
hostname Switch A
!
bridge irb
bridge 1 protocol ieee
bridge 2 protocol ieee
!
interface Port-channel1
no ip address
hold-queue 150 in
!
interface Port-channel1.1
encapsulation dot1Q 1 native
bridge-group 1
!
interface Port-channel1.2
encapsulation dot1Q 2
bridge-group 2
!
interface FastEthernet0
no ip address
channel-group 1
!
interface FastEthernet1
no ip address
channel-group 1
!
interface POS0
no ip address
crc 32
pos flag c2 1
!
interface POS0.1
encapsulation dot1Q 1 native
bridge-group 1
!
interface POS0.2
```

```
encapsulation dot1Q 2
bridge-group 2
```

Example 10-6 Switch B Configuration

```
hostname Switch B
!
bridge irb
bridge 1 protocol ieee
bridge 2 protocol ieee
!
interface Port-channel1
no ip address
hold-queue 150 in
!
interface Port-channel1.1
encapsulation dot1Q 1 native
bridge-group 1
!
interface Port-channel1.2
encapsulation dot1Q 2
bridge-group 2
!
interface FastEthernet0
no ip address
channel-group 1
!
interface FastEthernet1
no ip address
channel-group 1
!
interface POS0
no ip address
crc 32
pos flag c2 1
!
interface POS0.1
encapsulation dot1Q 1 native
bridge-group 1
!
interface POS0.2
encapsulation dot1Q 2
bridge-group 2
!
```

Monitoring and Verifying EtherChannel and POS

After FEC, GEC, or POS is configured, you can monitor its status using the **show interfaces port-channel** command.

Example 10-7 show interfaces port-channel Command

```
Router# show int port-channel 1
Port-channel1 is up, line protocol is up
Hardware is FEChannel, address is 0005.9a39.6634 (bia 0000.0000.0000)
MTU 1500 bytes, BW 200000 Kbit, DLY 100 usec,
reliability 255/255, txload 1/255, rxload 1/255
```

```
Encapsulation ARPA, loopback not set
Keepalive set (10 sec)
Unknown duplex, Unknown Speed
ARP type: ARPA, ARP Timeout 04:00:00
  No. of active members in this channel: 2
    Member 0 : FastEthernet0 , Full-duplex, Auto Speed
    Member 1 : FastEthernet1 , Full-duplex, Auto Speed
Last input 00:00:01, output 00:00:23, output hang never
Last clearing of "show interface" counters never
Input queue: 0/150/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
Output queue :0/80 (size/max)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
  820 packets input, 59968 bytes
  Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
  0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
  0 watchdog, 0 multicast
  0 input packets with dribble condition detected
  32 packets output, 11264 bytes, 0 underruns
  0 output errors, 0 collisions, 0 interface resets
  0 babbles, 0 late collision, 0 deferred
  0 lost carrier, 0 no carrier
  0 output buffer failures, 0 output buffers swapped out.
```




Configuring Networking Protocols

This chapter describes how to configure the ML-Series card for supported IP routing protocols. It is intended to provide enough information for a network administrator to get the protocols up and running. However, this section does not provide in-depth configuration detail for each protocol. For detailed information, refer to the *Cisco IOS IP and IP Routing Configuration Guide* and the *Cisco IOS IP and IP Routing Command Reference* publications.

This chapter contains the following major sections:

- [Basic IP Routing Protocol Configuration, page 11-1](#)
- [Configuring IP Routing, page 11-4](#)
- [Monitoring Static Routes, page 11-32](#)
- [Monitoring and Maintaining the IP Network, page 11-33](#)
- [Understanding IP Multicast Routing, page 11-33](#)
- [Configuring IP Multicast Routing, page 11-34](#)
- [Monitoring and Verifying IP Multicast Operation, page 11-35](#)

Basic IP Routing Protocol Configuration

IP routing is enabled by default on the ML-Series card.

For IP routing, you need the following to configure your interface:

- IP address
- IP subnet mask

You also need to do the following:

- Select a routing protocol.
- Assign IP network numbers to be advertised.

The ML Series supports the routing protocols listed and described in the following sections.

To configure IP routing protocols to run on a Fast Ethernet, Gigabit Ethernet, or Packet-over-SONET/SDH (POS) interface, perform one of the following procedures, depending on the protocol you are configuring.

RIP

To configure the Routing Information Protocol (RIP), perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# router rip	Enters router configuration mode, defines RIP as the routing protocol, and starts the RIP routing process.
Step 2	Router(config-router)# network <i>net-number</i>	Specifies a directly connected network based on the Internet Network Information Center (InterNIC) network number—not a subnet number or individual address. The routing process associates interfaces with the appropriate addresses and begins processing packets on the specified network.
Step 3	Router(config-router)# exit	Returns to global configuration mode.

EIGRP

To configure the Enhanced Interior Gateway Routing Protocol (EIGRP), perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# router eigrp <i>autonomous-system-number</i>	Defines EIGRP as the IP routing protocol. The autonomous system number is the autonomous system to which this ML-Series card belongs.
Step 2	Router(config-router)# network <i>net-number</i>	Defines the directly connected networks that run EIGRP. The network number is the number of the network that is advertised by this ML-Series card.
Step 3	Router(config-router)# exit	Returns to global configuration mode.

OSPF

To configure the Open Shortest Path First (OSPF) protocol, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# router ospf <i>process-ID</i>	Defines OSPF as the IP routing protocol. The process ID identifies a unique OSPF router process. This number is internal to the ML-Series card only; the process ID here does not have to match the process IDs on other routers.
Step 2	Router(config-router)# network <i>net-address wildcard-mask area area-ID</i>	Assigns an interface to a specific area. <ul style="list-style-type: none"> • The net-address is the address of directly connected networks or subnets. • The wildcard-mask is an inverse mask that compares a given address with interface addressing to determine whether OSPF uses this interface. • The area parameter identifies the interface as belonging to an area. • The area-ID specifies the area associated with the network address.
Step 3	Router(config-router)# end	Returns to privileged EXEC mode.

BGP

To configure the Border Gateway Protocol (BGP), perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# router bgp <i>autonomous-system-number</i>	Defines BGP as the IP routing protocol. The autonomous system number is the autonomous system to which this ML-Series card belongs.
Step 2	Router(config-router) # network <i>net-number</i>	Defines the directly connected networks that run BGP. The network number is the number of the network that is advertised by this ML-Series card.
Step 3	Router(config-router)# exit	Returns to global configuration mode.

Enabling IP Routing

Beginning in privileged EXEC mode, follow this procedure to enable IP routing:



Note

By default, IP routing is already enabled.

	Command	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# ip routing	Enables IP routing (default).
Step 3	Router(config)# router <i>ip-routing-protocol</i>	Specifies an IP routing protocol. This step might include other commands, such as specifying the networks to route with the network (RIP) router configuration command. For information about specific protocols, refer to sections later in this chapter and to the <i>Cisco IOS IP and IP Routing Configuration Guide</i> .
Step 4	Router(config-router)# end	Returns to privileged EXEC mode.
Step 5	Router(config)# show running-config	Verifies your entries.
Step 6	Router(config)# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Use the **no ip routing** global configuration command (Example 11-1) to disable routing.

Example 11-1 Enabling IP Routing Using RIP as the Routing Protocol

```
Router# configure terminal
Router(config)# ip routing
Router(config)# router rip
Router(config-router)# network 10.0.0.0
Router(config-router)# end
```

Configuring IP Routing

You can now set up parameters for the selected routing protocols as described in these sections:

- [Configuring RIP, page 11-4](#)
- [Configuring OSPF, page 11-9](#)
- [Configuring EIGRP, page 11-20](#)
- [Configuring BGP, page 11-27](#)
- [Configuring IS-IS, page 11-29](#)
- [Configuring Static Routes, page 11-31](#)

Configuring RIP

The Routing Information Protocol (RIP) is an Interior Gateway Protocol (IGP) created for use in small, homogeneous networks. It is a distance-vector routing protocol that uses broadcast User Datagram Protocol (UDP) data packets to exchange routing information. The protocol is documented in RFC 1058. You can find detailed information about RIP in *IP Routing Fundamentals*, published by Cisco Press.

Using RIP, the switch sends routing information updates (advertisements) every 30 seconds. If a router does not receive an update from another router for 180 seconds or more, it marks the routes served by that router as unusable. If there is still no update after 240 seconds, the router removes all routing table entries for the nonupdating router.

RIP uses hop counts to rate the value of different routes. The hop count is the number of routers that can be traversed in a route. A directly connected network has a hop count of zero; a network with a hop count of 16 is unreachable. This small range (0 to 15) makes RIP unsuitable for large networks.

If the router has a default network path, RIP advertises a route that links the router to the pseudo network 0.0.0.0. The 0.0.0.0 network does not exist; it is treated by RIP as a network to implement the default routing feature. The switch advertises the default network if a default was learned by RIP or if the router has a gateway of last resort and RIP is configured with a default metric. RIP sends updates to the interfaces in specified networks. If an interface's network is not specified, it is not advertised in any RIP update.

Table 11-1 shows the default RIP configuration.

Table 11-1 Default RIP Configuration

Feature	Default Setting
Auto summary	Enabled
Default-information originate	Disabled
Default metric	Built-in; automatic metric translations
IP RIP authentication key-chain	No authentication Authentication mode: clear text
IP RIP receive version	According to the version router configuration command
IP RIP send version	According to the version router configuration command
IP RIP triggered	According to the version router configuration command
IP split horizon	Varies with media
Neighbor	None defined
Network	None specified
Offset list	Disabled
Output delay	0 milliseconds
Timers basic	Update: 30 seconds Invalid: 180 seconds Hold-down: 180 seconds Flush: 240 seconds
Validate-update-source	Enabled
Version	Receives RIP Version 1 and Version 2 packets; sends Version 1 packets

To configure RIP, enable RIP routing for a network and optionally configure other parameters. Beginning in privileged EXEC mode, follow this procedure to enable and configure RIP:

	Command	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# ip routing	Enables IP routing. (Required only if IP routing is disabled.)
Step 3	Router(config)# router rip	Enables a RIP routing process, and enters router configuration mode.
Step 4	Router(config-router)# network <i>network-number</i>	Associates a network with a RIP routing process. You can specify multiple network commands. RIP routing updates are sent and received through interfaces only on these networks.
Step 5	Router(config-router)# neighbor <i>ip-address</i>	(Optional) Defines a neighboring router with which to exchange routing information. This step allows routing updates from RIP (normally a broadcast protocol) to reach nonbroadcast networks.
Step 6	Router(config-router)# offset list { <i>[access-list-number name]</i> } { in out } <i>offset [type-number]</i>	(Optional) Applies an offset list to routing metrics to increase incoming and outgoing metrics to routes learned through RIP. You can limit the offset list with an access list or an interface.
Step 7	Router(config-router)# timers basic <i>update invalid holddown flush</i>	(Optional) Adjusts routing protocol timers. Valid ranges for all timers are 0 to 4294967295 seconds. <ul style="list-style-type: none"> • update—The time (in seconds) between sending of routing updates. The default is 30 seconds. • invalid—The timer interval (in seconds) after which a route is declared invalid. The default is 180 seconds. • holddown—The time (in seconds) that must pass before a route is removed from the routing table. The default is 180 seconds. • flush—The amount of time (in seconds) for which routing updates are postponed. The default is 240 seconds.
Step 8	Router(config-router)# version { 1 2 }	(Optional) Configures the switch to receive and send only RIP Version 1 or RIP Version 2 packets. By default, the switch receives Version 1 and 2 but sends only Version 1. You can also use the interface commands ip rip {send receive} version {1 2 1 2} to control what versions are used for sending and receiving on interfaces.
Step 9	Router(config-router)# no auto summary	(Optional) Disables automatic summarization. By default, the switch summarizes subprefixes when crossing classful network boundaries. Disables summarization (RIP Version 2 only) to advertise subnet and host routing information to classful network boundaries.
Step 10	Router(config-router)# no validate-update-source	(Optional) Disables validation of the source IP address of incoming RIP routing updates. By default, the switch validates the source IP address of incoming RIP routing updates and discards the update if the source address is not valid. Under normal circumstances, disabling this feature is not recommended. However, if you have a router that is off-network and you want to receive its updates, you can use this command.
Step 11	Router(config-router)# output-delay <i>delay</i>	(Optional) Adds interpacket delay for RIP updates sent. By default, packets in a multiple-packet RIP update have no delay added between packets. If you are sending packets to a lower-speed device, you can add an interpacket delay in the range of 8 to 50 milliseconds.
Step 12	Router(config-router)# end	Returns to privileged EXEC mode.

	Command	Purpose
Step 13	Router# show ip protocols	Verifies your entries.
Step 14	Router# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

To turn off the RIP routing process, use the **no router rip** global configuration command.

To display the parameters and current state of the active routing protocol process, use the **show ip protocols** privileged EXEC command (Example 11-2).

Example 11-2 *show ip protocols Command Output (Showing RIP Processes)*

```
Router# show ip protocols
Routing Protocol is "rip"
  Sending updates every 30 seconds, next due in 15 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Redistributing: rip
  Default version control: send version 1, receive any version
    Interface         Send Recv Triggered RIP Key-chain
  FastEthernet0      1     1 2
  POS0                1     1 2
  Automatic network summarization is in effect
  Maximum path: 4
  Routing for Networks:
    192.168.2.0
    192.168.3.0
  Routing Information Sources:
    Gateway         Distance    Last Update
  192.168.2.1       120        00:00:23
  Distance: (default is 120)
```

Use the **show ip rip database** privileged EXEC command to display summary address entries in the RIP database (Example 11-3).

Example 11-3 *show ip rip database Command Output*

```
Router# show ip rip database
192.168.1.0/24    auto-summary
192.168.1.0/24
  [1] via 192.168.2.1, 00:00:24, POS0
192.168.2.0/24    auto-summary
192.168.2.0/24    directly connected, POS0
192.168.3.0/24    auto-summary
192.168.3.0/24    directly connected, FastEthernet0
```

RIP Authentication

RIP Version 1 does not support authentication. If you are sending and receiving RIP Version 2 packets, you can enable RIP authentication on an interface. The key chain determines the set of keys that can be used on the interface. If a key chain is not configured, no authentication is performed, not even the default.

The switch supports two modes of authentication on interfaces for which RIP authentication is enabled: plain text and message-digest key (MD5). The default is plain text.

Beginning in privileged EXEC mode, follow this procedure to configure RIP authentication on an interface:

	Command	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# interface <i>interface-id</i>	Enters interface configuration mode, and specifies the interface to configure.
Step 3	Router(config-if)# ip rip authentication key-chain <i>name-of-chain</i>	Enables RIP authentication.
Step 4	Router(config-if)# ip rip authentication mode {text md5}	Configures the interface to use plain text authentication (the default) or MD5 digest authentication.
Step 5	Router(config-if)# end	Returns to privileged EXEC mode.
Step 6	Router# show running-config interface [<i>interface-id</i>]	Verifies your entries.
Step 7	Router# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

To restore clear text authentication, use the **no ip rip authentication mode** interface configuration command. To prevent authentication, use the **no ip rip authentication key-chain** interface configuration command.

Summary Addresses and Split Horizon

Routers connected to broadcast-type IP networks and using distance-vector routing protocols normally use the split-horizon mechanism to reduce the possibility of routing loops. Split horizon blocks information about routes from being advertised by a router on any interface from which that information originated. This feature usually optimizes communication among multiple routers, especially when links are broken.



Note

In general, disabling split horizon is not recommended unless you are certain that your application requires it to properly advertise routes.

If you want to configure an interface running RIP to advertise a summarized local IP address pool on a network access server for dial-up clients, use the **ip summary-address rip** interface configuration command.

Beginning in privileged EXEC mode, follow these steps to set an interface to advertise a summarized local IP address pool and to disable split horizon on the interface:

	Command	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# interface <i>interface-id</i>	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	Router(config-if)# ip address <i>ip-address subnet-mask</i>	Configures the IP address and IP subnet.

	Command	Purpose
Step 4	Router(config-if)# ip summary-address rip <i>ip-address ip-network-mask</i>	Configures the IP address to be summarized and the IP network mask.
Step 5	Router(config-if)# no ip split horizon	Disables split horizon on the interface.
Step 6	Router(config-if)# end	Returns to privileged EXEC mode.
Step 7	Router# show ip interface <i>interface-id</i>	Verifies your entries.
Step 8	Router# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

To disable IP summarization, use the **no ip summary-address rip** router configuration command.



Note

If split horizon is enabled, neither autosummary nor interface summary addresses (those configured with the **ip summary-address rip** router configuration command) are advertised.

Configuring OSPF

This section briefly describes how to configure the Open Shortest Path First (OSPF) protocol. For a complete description of the OSPF commands, refer to the “OSPF Commands” chapter of the *Cisco IOS IP and IP Routing Command Reference* publication.

OSPF is an IGP designed expressly for IP networks, supporting IP subnetting and tagging of externally derived routing information. OSPF also allows packet authentication and uses IP multicast when sending and receiving packets. The Cisco implementation supports RFC 1253, the OSPF MIB.

The Cisco implementation conforms to the OSPF Version 2 specifications with these key features:

- Stub areas—Definition of stub areas is supported.
- Route redistribution—Routes learned through any IP routing protocol can be redistributed into another IP routing protocol. At the intradomain level, this means that OSPF can import and export routes learned through protocols such as EIGRP and RIP.
- Authentication—Plain text and MD5 authentication among neighboring routers within an area are supported.
- Routing interface parameter—Configurable parameters supported include interface output cost, retransmission interval, interface transmit delay, router priority, router dead and hello intervals, and authentication key.
- Virtual links—Virtual links are supported.
- Not-so-stubby-area (NSSA)—RFC 1587.

OSPF typically requires coordination among many internal routers, area border routers (ABRs) connected to multiple areas, and autonomous system boundary routers (ASBRs). The minimum configuration would use all default parameter values, no authentication, and interfaces assigned to areas. If you customize your environment, you must ensure coordinated configuration of all routers.

Table 11-2 shows the default OSPF configuration.

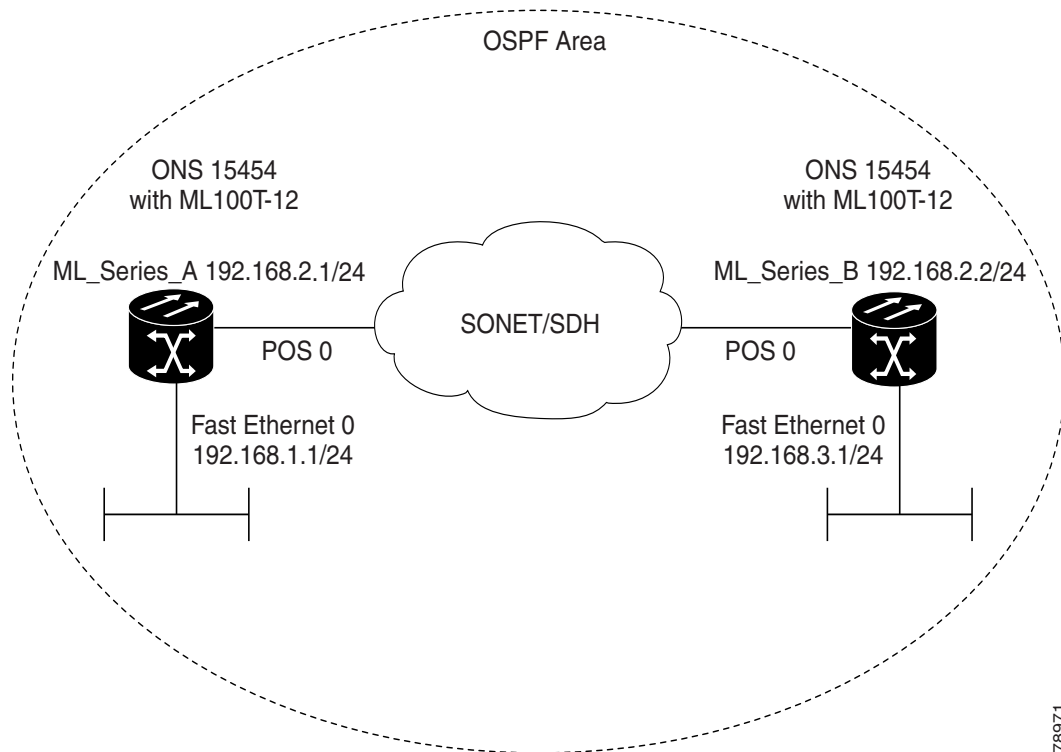
Table 11-2 **Default OSPF Configuration**

Feature	Default Setting
Interface parameters	Cost: No default cost predefined. Retransmit interval: 5 seconds. Transmit delay: 1 second. Priority: 1. Hello interval: 10 seconds. Dead interval: 4 times the hello interval. No authentication. No password specified. MD5 authentication disabled.
Area	Authentication type: 0 (no authentication). Default cost: 1. Range: Disabled. Stub: No stub area defined. NSSA: No NSSA area defined.
Auto cost	100 Mbps.
Default-information originate	Disabled. When enabled, the default metric setting is 10, and the external route type default is Type 2.
Default metric	Built-in, automatic metric translation, as appropriate for each routing protocol.
Distance OSPF	dist1 (all routes within an area): 110 dist2 (all routes from one area to another): 110 dist3 (routes from other routing domains): 110
OSPF database filter	Disabled. All outgoing link-state advertisements (LSAs) are flooded to the interface.
IP OSPF name lookup	Disabled.
Log adjacency changes	Enabled.
Neighbor	None specified.
Neighbor database filter	Disabled. All outgoing LSAs are flooded to the neighbor.
Network area	Disabled.
Router ID	No OSPF routing process defined.
Summary address	Disabled.
Timers LSA group pacing	240 seconds.

Table 11-2 Default OSPF Configuration (continued)

Feature	Default Setting
Timers shortest path first (spf)	spf delay: 5 seconds. spf-holdtime: 10 seconds.
Virtual link	No area ID or router ID defined. Hello interval: 10 seconds. Retransmit interval: 5 seconds. Transmit delay: 1 second. Dead interval: 40 seconds. Authentication key: No key predefined. MD5: No key predefined.

Figure 11-1 shows an example of an IP routing protocol using OSPF.

Figure 11-1 IP Routing Protocol Example Using OSPF

78971

Enabling OSPF requires that you create an OSPF routing process, specify the range of IP addresses to be associated with the routing process, and assign area IDs to be associated with that range.

Beginning in privileged EXEC mode, follow this procedure to enable OSPF:

	Command	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# router ospf <i>process-id</i>	Enables OSPF routing, and enters router configuration mode. The process ID is an internally used identification parameter that is locally assigned and can be any positive integer. Each OSPF routing process has a unique value.
Step 3	Router(config)# network <i>address</i> <i>wildcard-mask</i> area <i>area-id</i>	Defines an interface on which OSPF runs and the area ID for that interface. Use the wildcard-mask to use a single command to define one or more multiple interfaces to be associated with a specific OSPF area. The area ID can be a decimal value or an IP address.
Step 4	Router(config)# end	Returns to privileged EXEC mode.
Step 5	Router# show ip protocols	Verifies your entries.
Step 6	Router# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

To terminate an OSPF routing process, use the **no router ospf** *process-id* global configuration command.

[Example 11-4](#) shows an example of configuring an OSPF routing process. In the example, a process number of 1 is assigned. [Example 11-5](#) shows the output of the command used to verify the OSPF process ID.

Example 11-4 Configuring an OSPF Routing Process

```
Router(config)# router ospf 1
Router(config-router)# network 192.168.1.0 0.0.0.255 area 0
```

Example 11-5 show ip protocols Privileged EXEC Command Output

```
Router# show ip protocols
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 192.168.3.1
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    192.168.2.0 0.0.0.255 area 0
    192.168.3.0 0.0.0.255 area 0
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.3.1         110          00:03:34
    192.168.2.1         110          00:03:34
  Distance: (default is 110)
```

OSPF Interface Parameters

You can use the **ip ospf** interface configuration commands to modify interface-specific OSPF parameters. You are not required to modify any of these parameters, but some interface parameters (hello interval, dead interval, and authentication key) must be consistent across all routers in an attached network. If you modify these parameters, be sure all routers in the network have compatible values.



Note The **ip ospf** interface configuration commands are all optional.

Beginning in privileged EXEC mode, follow these steps to modify OSPF interface parameters:

	Command	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# interface <i>interface-id</i>	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	Router(config-if)# ip ospf cost	(Optional) Explicitly specifies the cost of sending a packet on the interface.
Step 4	Router(config-if)# ip ospf retransmit-interval <i>seconds</i>	(Optional) Specifies the number of seconds between link state advertisement transmissions. The range is 1 to 65535 seconds. The default is 5 seconds.
Step 5	Router(config-if)# ip ospf transmit-delay <i>seconds</i>	(Optional) Sets the estimated number of seconds to wait before sending a link state update packet. The range is 1 to 65535 seconds. The default is 1 second.
Step 6	Router(config-if)# ip ospf priority <i>number</i>	(Optional) Sets priority to help determine the OSPF designated router for a network. The range is from 0 to 255. The default is 1.
Step 7	Router(config-if)# ip ospf hello-interval <i>seconds</i>	(Optional) Sets the number of seconds between hello packets sent on an OSPF interface. The value must be the same for all nodes on a network. The range is 1 to 65535 seconds. The default is 10 seconds.
Step 8	Router(config-if)# ip ospf dead-interval <i>seconds</i>	(Optional) Sets the number of seconds after the last device hello packet was seen before its neighbors declare the OSPF router to be down. The value must be the same for all nodes on a network. The range is 1 to 65535 seconds. The default is 4 times the hello interval.
Step 9	Router(config-if)# ip ospf authentication-key <i>key</i>	(Optional) Assigns a password to be used by neighboring OSPF routers. The password can be any string of keyboard-entered characters up to 8 bytes in length. All neighboring routers on the same network must have the same password to exchange OSPF information.
Step 10	Router(config-if)# ip ospf message digest-key <i>keyid md5 key</i>	(Optional) Enables authentication. <ul style="list-style-type: none"> keyid—Identifier from 1 to 255. key—Alphanumeric password of up to 16 bytes.

	Command	Purpose
Step 11	Router(config-if)# ip ospf database-filter all out	(Optional) Blocks flooding of OSPF LSA packets to the interface. By default, OSPF floods new LSAs over all interfaces in the same area, except the interface on which the LSA arrives.
Step 12	Router(config-if)# end	Returns to privileged EXEC mode.
Step 13	Router# show ip ospf interface [interface-name]	Displays OSPF-related interface information.
Step 14	Router# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Use the **no** form of these commands to remove the configured parameter value or return to the default value. [Example 11-6](#) shows the output of the **show ip ospf interface** privileged EXEC command.

Example 11-6 show ip ospf interface Privileged EXEC Command Output

```
Router# show ip ospf interface
FastEthernet0 is up, line protocol is up
  Internet Address 192.168.3.1/24, Area 0
  Process ID 1, Router ID 192.168.3.1, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 192.168.3.1, Interface address 192.168.3.1
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:01
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 0
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 0, Adjacent neighbor count is 0
  Suppress hello for 0 neighbor(s)
POS0 is up, line protocol is up
  Internet Address 192.168.2.2/24, Area 0
  Process ID 1, Router ID 192.168.3.1, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 192.168.3.1, Interface address 192.168.2.2
  Backup Designated router (ID) 192.168.2.1, Interface address 192.168.2.1
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:05
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 2, maximum is 2
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 192.168.2.1 (Backup Designated Router)
  Suppress hello for 0 neighbor(s)
```

OSPF Area Parameters

You can optionally configure several OSPF area parameters. These parameters include authentication for password-based protection against unauthorized access to an area, stub areas, and NSSAs. Stub areas are areas into which information about external routes is not sent. Instead, the ABR generates a default external route into the stub area for destinations outside the autonomous system (AS). An NSSA does not flood all LSAs from the core into the area, but can import AS external routes within the area by redistribution.

Route summarization is the consolidation of advertised addresses into a single summary route to be advertised by other areas. If network numbers are contiguous, you can use the **area range** router configuration command to configure the ABR to advertise a summary route that covers all networks in the range.



Note The OSPF **area** router configuration commands are all optional.

Beginning in privileged EXEC mode, follow these steps to configure area parameters:

	Command	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# router ospf <i>process-id</i>	Enables OSPF routing, and enters router configuration mode.
Step 3	Router(config)# area area-id authentication	(Optional) Allows password-based protection against unauthorized access to the identified area. The identifier can be either a decimal value or an IP address.
Step 4	Router(config)# area area-id authentication message-digest	(Optional) Enables MD5 authentication on the area.
Step 5	Router(config)# area area-id stub [no-summary]	(Optional) Defines an area as a stub area. The no-summary keyword prevents an ABR from sending summary link advertisements into the stub area.
Step 6	Router(config)# area area-id nssa { no-redistribution default-information-originate no-summary }	(Optional) Defines an area as a not-so-stubby-area. Every router within the same area must agree that the area is NSSA. Select one of these keywords: <ul style="list-style-type: none"> • no-redistribution—Select when the router is an NSSA ABR and you want the redistribute command to import routes into normal areas, but not into the NSSA. • default-information-originate—Select on an ABR to allow importing type 7 LSAs into the NSSA. • no-redistribution—Select to not send summary LSAs into the NSSA.
Step 7	Router(config)# area area-id range <i>address-mask</i>	(Optional) Specifies an address range for which a single route is advertised. Use this command only with area border routers.
Step 8	Router(config)# end	Returns to privileged EXEC mode.
Step 9	Router# show ip ospf [<i>process-id</i>]	Displays information about the OSPF routing process in general or for a specific process ID to verify configuration.
Step 10	Router# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Use the **no** form of these commands to remove the configured parameter value or to return to the default value. [Example 11-7](#) shows the output of the **show ip ospf database** and the **show ip ospf** privileged EXEC commands.

Example 11-7 show ip ospf database and show ip ospf Privileged EXEC Command Outputs

```

Router# show ip ospf database

      OSPF Router with ID (192.168.3.1) (Process ID 1)

          Router Link States (Area 0)

Link ID          ADV Router      Age           Seq#           Checksum Link count
192.168.2.1     192.168.2.1     428          0x80000003    0x004AB8  2
192.168.3.1     192.168.3.1     428          0x80000003    0x006499  2

          Net Link States (Area 0)

Link ID          ADV Router      Age           Seq#           Checksum
192.168.2.2     192.168.3.1     428          0x80000001    0x00A4E0

Router# show ip ospf
Routing Process "ospf 1" with ID 192.168.3.1
Supports only single TOS(TOS0) routes
Supports opaque LSA
SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
Number of external LSA 0. Checksum Sum 0x000000
Number of opaque AS LSA 0. Checksum Sum 0x000000
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
External flood list length 0
  Area BACKBONE(0)
    Number of interfaces in this area is 2
    Area has no authentication
    SPF algorithm executed 4 times
    Area ranges are
    Number of LSA 3. Checksum Sum 0x015431
    Number of opaque link LSA 0. Checksum Sum 0x000000
    Number of DCbitless LSA 0
    Number of indication LSA 0
    Number of DoNotAge LSA 0
    Flood list length 0

```

Other OSPF Behavior Parameters

You can optionally configure other OSPF parameters in router configuration mode:

- **Route summarization**—When redistributing routes from other protocols, each route is advertised individually in an external LSA. To help decrease the size of the OSPF link state database, you can use the **summary-address** router configuration command to advertise a single router for all the redistributed routes included in a specified network address and mask.
- **Virtual links**—In OSPF, all areas must be connected to a backbone area. You can establish a virtual link in case of a backbone-continuity break by configuring two ABRs as endpoints of a virtual link. Configuration information includes the identity of the other virtual endpoint (the other ABR) and the nonbackbone link that the two routers have in common (the transit area). Virtual links cannot be configured through a stub area.
- **Default route**—When you specifically configure redistribution of routes into an OSPF routing domain, the route automatically becomes an ASBR. You can force the ASBR to generate a default route into the OSPF routing domain.

- Domain Name Server (DNS) names for use in all OSPF **show** privileged EXEC command displays make it easier to identify a router than displaying it by router ID or neighbor ID.
- Default metrics—OSPF calculates the OSPF metric for an interface according to the bandwidth of the interface. The metric is calculated as *ref-bw* divided by bandwidth, where *ref* is 10 by default, and bandwidth (*bw*) is determined by the **bandwidth** interface configuration command. For multiple links with high bandwidth, you can specify a larger number to differentiate the cost on those links.
- Administrative distance—This is a rating of the trustworthiness of a routing information source, an integer between 0 and 255, with a higher value meaning a lower trust rating. An administrative distance of 255 means that the routing information source cannot be trusted at all and should be ignored. OSPF uses three different administrative distances: routes within an area (intra-area), routes to another area (interarea), and routes from another routing domain learned through redistribution (external). You can change any of the distance values.
- Passive interfaces—Because interfaces between two devices on an Ethernet represent only one network segment, to prevent OSPF from sending hello packets for the sending interface, you must configure the sending device to be a passive interface. Both devices can identify each other through the hello packet for the receiving interface.
- Route calculation timers—You can configure the delay time between when OSPF receives a topology change and when it starts the shortest path first (SPF) calculation. You can also configure the hold time between two SPF calculations.
- Log neighbor changes—You can configure the router to send a syslog message when an OSPF neighbor state changes, providing a high-level view of changes in the router.

Beginning in privileged EXEC mode, follow this procedure to configure these OSPF parameters:

	Command	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# router ospf process-id	Enables OSPF routing, and enters router configuration mode.
Step 3	Router(config)# summary-address address-mask	(Optional) Specifies an address and IP subnet mask for redistributed routes so that only one summary route is advertised.
Step 4	Router(config)# area area-id virtual-link router-id [hello-interval seconds] [retransmit-interval seconds] [trans] {[authentication-key key] [message-digest-key key-id md5 key]}	(Optional) Establishes a virtual link and set its parameters. See the “OSPF Interface Parameters” section on page 11-13 for parameter definitions and Table 11-2 on page 11-10 for virtual link defaults.
Step 5	Router(config)# default-information originate [always] [metric metric-value] [metric-type type-value] [route-map map-name]	(Optional) Forces the ASBR to generate a default route into the OSPF routing domain. Parameters are all optional.
Step 6	Router(config)# ip ospf name-lookup	(Optional) Configures DNS name lookup. The default is disabled.
Step 7	Router(config)# ip auto-cost reference-bandwidth ref-bw	(Optional) Specifies an address range for which a single route will be advertised. Use this command only with area border routers.
Step 8	Router(config)# distance ospf {[inter-area dist1] [inter-area dist2] [external dist3]}	(Optional) Changes the OSPF distance values. The default distance for each type of route is 110. The range is 1 to 255.

	Command	Purpose
Step 9	Router(config)# passive-interface <i>type number</i>	(Optional) Suppresses the sending of hello packets through the specified interface.
Step 10	Router(config)# timers spf <i>spf-delay spf-holdtime</i>	(Optional) Configures route calculation timers. <ul style="list-style-type: none"> spf-delay—Enter an integer from 0 to 65535. The default is 5 seconds; 0 means no delay. spf-holdtime—Enter an integer from 0 to 65535. The default is 10 seconds; 0 means no delay.
Step 11	Router(config)# ospf log-adj-changes	(Optional) Sends syslog message when a neighbor state changes.
Step 12	Router(config)# end	Returns to privileged EXEC mode.
Step 13	Router# show ip ospf [<i>process-id</i> [<i>area-id</i>]] database	Displays lists of information related to the OSPF database for a specific router. For some of the keyword options, see to the “Monitoring OSPF” section on page 11-19 .
Step 14	Router# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Change LSA Group Pacing

The OSPF LSA group pacing feature allows the router to group OSPF LSAs and pace the refreshing, check-summing, and aging functions for more efficient router use. This feature is enabled by default with a four-minute default pacing interval, and you do not usually need to modify this parameter. The optimum group pacing interval is inversely proportional to the number of LSAs the router is refreshing, check-summing, and aging. For example, if you have approximately 10,000 LSAs in the database, decreasing the pacing interval would benefit you. If you have a very small database (40 to 100 LSAs), increasing the pacing interval to 10 to 20 minutes might benefit you slightly.

Beginning in privileged EXEC mode, follow this procedure to configure OSPF LSA pacing:

	Command	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# router ospf <i>process-id</i>	Enables OSPF routing, and enters router configuration mode.
Step 3	Router(config)# timers lsa-group-pacing <i>seconds</i>	Changes the group pacing of LSAs.
Step 4	Router(config)# end	Returns to privileged EXEC mode.
Step 5	Router# show running-config	Verifies your entries.
Step 6	Router# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

To return to the default value, use the **no timers lsa-group-pacing** router configuration command.

Loopback Interface

OSPF uses the highest IP address configured on the interfaces as its router ID. If this interface is down or removed, the OSPF process must recalculate a new router ID and resend all its routing information out of its interfaces. If a loopback interface is configured with an IP address, OSPF uses this IP address as its router ID, even if other interfaces have higher IP addresses. Because loopback interfaces never fail, this provides greater stability. OSPF automatically prefers a loopback interface over other interfaces, and it chooses the highest IP address among all loopback interfaces.

Beginning in privileged EXEC mode, follow this procedure to configure a loopback interface:

	Command	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# interface loopback 0	Creates a loopback interface, and enters interface configuration mode.
Step 3	Router(config)# ip address address mask	Assigns an IP address to this interface.
Step 4	Router(config)# end	Returns to privileged EXEC mode.
Step 5	Router# show ip interface	Verifies your entries.
Step 6	Router# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Use the **no interface loopback 0** global configuration command to disable the loopback interface.

Monitoring OSPF

You can display specific statistics such as the contents of IP routing tables, caches, and databases.

Table 11-3 lists some of the privileged EXEC commands for displaying statistics. For more **show ip ospf database** privileged EXEC command options and for explanations of fields in the resulting display, refer to the *Cisco IOS IP and IP Routing Command Reference*.

Table 11-3 Show IP OSPF Statistics Commands

Command	Purpose
Router(config)# show ip ospf [process-id]	Displays general information about OSPF routing processes.
Router(config)# show ip ospf [process-id] database [router] [link-state-id]	Displays lists of information related to the OSPF database.
Router(config)# show ip ospf border-routes	Displays the internal OSPF routing ABR and ASBR table entries.
Router(config)# show ip ospf interface [interface-name]	Displays OSPF-related interface information.
Router(config)# show ip ospf neighbor [interface-name] [neighbor-id] detail	Displays OSPF interface neighbor information.
Router(config)# show ip ospf virtual-links	Displays OSPF-related virtual links information.

Configuring EIGRP

Enhanced IGRP (EIGRP) is a Cisco proprietary enhanced version of the Interior Gateway Routing Protocol (IGRP). Enhanced IGRP uses the same distance vector algorithm and distance information as IGRP; however, the convergence properties and the operating efficiency of Enhanced IGRP are significantly improved.

The convergence technology employs an algorithm referred to as the Diffusing Update Algorithm (DUAL), which guarantees loop-free operation at every instant throughout a route computation and allows all devices involved in a topology change to synchronize at the same time. Routers that are not affected by topology changes are not involved in recomputations.

IP EIGRP provides increased network width. With RIP, the largest possible width of your network is 15 hops. When IGRP is enabled, the largest possible width is 224 hops. Because the EIGRP metric is large enough to support thousands of hops, the only barrier to expanding the network is the transport-layer hop counter. EIGRP increments the transport control field only when an IP packet has traversed 15 routers and the next hop to the destination was learned through EIGRP. When a RIP route is used as the next hop to the destination, the transport control field is incremented as usual.

EIGRP offers the following features:

- Fast convergence
- Incremental updates when the state of a destination changes, instead of sending the entire contents of the routing table, minimizing the bandwidth required for EIGRP packets
- Less CPU usage than IGRP because full update packets do not need to be processed each time they are received
- Protocol-independent neighbor discovery mechanism to learn about neighboring routers
- Variable-length subnet masks (VLSMs)
- Arbitrary route summarization
- EIGRP scales to large networks

EIGRP has four basic components:

- Neighbor discovery and recovery is the process that routers use to dynamically learn of other routers on their directly attached networks. Routers must also discover when their neighbors become unreachable or inoperative. Neighbor discovery and recovery is achieved with low overhead by periodically sending small hello packets. As long as hello packets are received, the Cisco IOS software can determine that a neighbor is alive and functioning. When this status is determined, the neighboring routers can exchange routing information.
- The reliable transport protocol is responsible for guaranteed, ordered delivery of EIGRP packets to all neighbors. It supports intermixed transmission of multicast and unicast packets. Some EIGRP packets must be sent reliably, and others need not be. For efficiency, reliability is provided only when necessary. For example, on a multiaccess network that has multicast capabilities (such as Ethernet), it is not necessary to send hellos reliably to all neighbors individually. Therefore, EIGRP sends a single multicast hello with an indication in the packet informing the receivers that the packet need not be acknowledged. Other types of packets (such as updates) require acknowledgment, which is shown in the packet. The reliable transport has a provision to send multicast packets quickly when there are unacknowledged packets pending. Doing so helps ensure that convergence time remains low in the presence of varying speed links.
- The DUAL finite state machine embodies the decision process for all route computations. It tracks all routes advertised by all neighbors. DUAL uses the distance information (known as a metric) to select efficient, loop-free paths. DUAL selects routes to be inserted into a routing table based on feasible successors. A successor is a neighboring router used for packet forwarding that has a

least-cost path to a destination that is guaranteed not to be part of a routing loop. When there are no feasible successors, but there are neighbors advertising the destination, a recomputation must occur. This is the process whereby a new successor is determined. The amount of time it takes to recompute the route affects the convergence time. Recomputation is processor-intensive; it is advantageous to avoid recomputation if it is not necessary. When a topology change occurs, DUAL tests for feasible successors. If there are feasible successors, it uses any it finds to avoid unnecessary recomputation.

- The protocol-dependent modules are responsible for network layer protocol-specific tasks. An example is the IP EIGRP module, which is responsible for sending and receiving EIGRP packets that are encapsulated in IP. It is also responsible for parsing EIGRP packets and informing DUAL of the new information received. EIGRP asks DUAL to make routing decisions, but the results are stored in the IP routing table. EIGRP is also responsible for redistributing routes learned by other IP routing protocols.

Table 11-4 shows the default EIGRP configuration.

Table 11-4 **Default EIGRP Configuration**

Feature	Default Setting
Auto summary	Enabled. Subprefixes are summarized to the classful network boundary when crossing classful network boundaries.
Default-information	Exterior routes are accepted and default information is passed between IGRP or EIGRP processes when doing redistribution.
Default metric	Only connected routes and interface static routes can be redistributed without a default metric. The metric includes: <ul style="list-style-type: none"> • Bandwidth: 0 or greater kbps. • Delay (tens of microseconds): 0 or any positive number that is a multiple of 39.1 nanoseconds. • Reliability: Any number between 0 and 255 (255 means 100 percent reliability). • Loading: Effective bandwidth as a number between 0 and 255 (255 is 100 percent loading). • MTU: Maximum transmission unit size of the route in bytes. 0 or any positive integer.
Distance	Internal distance: 90. External distance: 170.
EIGRP log-neighbor changes	Disabled. No adjacency changes logged.
IP authentication key-chain	No authentication provided.
IP authentication mode	No authentication provided.
IP bandwidth-percent	50 percent.
IP hello interval	For low-speed nonbroadcast multiaccess (NBMA) networks: 60 seconds; all other networks: 5 seconds.
IP hold-time	For low-speed NBMA networks: 180 seconds; all other networks: 15 seconds.
IP split-horizon	Enabled.
IP summary address	No summary aggregate addresses are predefined.


Table 11-4 Default EIGRP Configuration (continued)

Feature	Default Setting
Metric weights	tos: 0 k1 and k3: 1 k2, k4, and k5: 0
Network	None specified.
Offset-list	Disabled.
Router EIGRP	Disabled.
Set metric	No metric set in the route map.
Traffic-share	Distributed proportionately to the ratios of the metrics.
Variance	1 (equal-cost load balancing).

To create an EIGRP routing process, you must enable EIGRP and associate networks. EIGRP sends updates to the interfaces in the specified networks. If you do not specify an interface network, it is not advertised in any EIGRP update.

EIGRP Router Mode Commands

Beginning in privileged EXEC mode, follow these steps to configure EIGRP. Configuring the routing process is required; other steps are optional.

	Command	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# router eigrp <i>autonomous-system-number</i>	Enables an EIGRP routing process, and enters router configuration mode. The autonomous system number identifies the routes to other EIGRP routers and is used to tag routing information.
Step 3	Router(config)# network <i>network-number</i>	Associates networks with an EIGRP routing process. EIGRP sends updates to the interfaces in the specified networks. If an interface's network is not specified, it is not advertised in any IGRP or EIGRP update.
Step 4	Router(config)# eigrp log-neighbor-changes	(Optional) Enables logging of EIGRP neighbor changes to monitor routing system stability.
Step 5	Router(config)# metric weights tos <i>k1 k2 k3 k4 k5</i>	(Optional) Adjusts the EIGRP metric. Although the defaults have been carefully determined to provide excellent operation in most networks, you can adjust them.
		 Caution Determining metrics is complex and is not recommended without guidance from an experienced network designer.

	Command	Purpose
Step 6	Router(config)# offset list [{ <i>access-list-number</i> <i>name</i> }] { in out } <i>offset</i> [<i>type-number</i>]	(Optional) Applies an offset list to routing metrics to increase incoming and outgoing metrics to routes learned through EIGRP. You can limit the offset list with an access list or an interface.
Step 7	Router(config)# no auto-summary	(Optional) Disables automatic summarization of subnet routes into network-level routes.
Step 8	Router(config)# ip summary-address eigrp <i>autonomous-system-number</i> <i>address-mask</i>	(Optional) Configures a summary aggregate.
Step 9	Router(config)# end	Returns to privileged EXEC mode.
Step 10	Router# show ip protocols	Verifies your entries.
Step 11	Router# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Use the **no** forms of these commands to disable the feature or return the setting to the default value. [Example 11-8](#) shows the output for the **show ip protocols** privileged EXEC command.


Example 11-8 show ip protocols privileged EXEC Command Output (for EIGRP)

```
Router# show ip protocols
Routing Protocol is "eigrp 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  EIGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
  EIGRP maximum hopcount 100
  EIGRP maximum metric variance 1
  Redistributing: eigrp 1
  Automatic network summarization is in effect
  Automatic address summarization:
    192.168.3.0/24 for POS0
    192.168.2.0/24 for FastEthernet0
  Maximum path: 4
  Routing for Networks:
    192.168.2.0
    192.168.3.0
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.2.1           90          00:03:16
  Distance: internal 90 external 170
```

EIGRP Interface Mode Commands

Other optional EIGRP parameters can be configured on an interface basis.

Beginning in privileged EXEC mode, follow these steps:

	Command	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# interface <i>interface-id</i>	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	Router(config)# ip bandwidth-percent eigrp <i>autonomous-system-number percent</i>	(Optional) Configures the maximum percentage of bandwidth that can be used by EIGRP on an interface. The default is 50 percent.
Step 4	Router(config)# ip summary-address eigrp <i>autonomous-system-number address mask</i>	(Optional) Configures a summary aggregate address for a specified interface (not usually necessary if autosummary is enabled).
Step 5	Router(config)# ip hello-interval eigrp <i>autonomous-system-number seconds</i>	(Optional) Changes the hello time interval for an EIGRP routing process. The range is 1 to 65535 seconds. The default is 60 seconds for low-speed NBMA networks and 5 seconds for all other networks.
Step 6	Router(config)# ip hold-time eigrp <i>autonomous-system-number seconds</i>	(Optional) Changes the hold time interval for an EIGRP routing process. The range is 1 to 65535 seconds. The default is 180 seconds for low-speed NBMA networks and 15 seconds for all other networks.
		 <p>Caution Do not adjust the hold time without consulting Cisco technical support.</p>
Step 7	Router(config)# no ip split-horizon eigrp <i>autonomous-system-number</i>	(Optional) Disables split horizon to allow route information to be advertised by a router out any interface from which that information originated.
Step 8	Router# end	Returns to privileged EXEC mode.
Step 9	Router# show ip eigrp interface	Displays the interfaces that EIGRP is active on and information about EIGRP relating to those interfaces.
Step 10	Router# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Use the **no** forms of these commands to disable the feature or return the setting to the default value. [Example 11-9](#) shows the output of the **show ip eigrp interface** privileged EXEC command.

Example 11-9 show ip eigrp interface Privileged EXEC Command Output

```
Router# show ip eigrp interface
IP-EIGRP interfaces for process 1
```

Interface	Peers	Xmit Queue Un/Reliable	Mean SRTT	Pacing Time Un/Reliable	Multicast Flow Timer	Pending Routes
PO0	1	0/0	20	0/10	50	0
Fa0	0	0/0	0	0/10	0	0

Configure EIGRP Route Authentication

EIGRP route authentication provides MD5 authentication of routing updates from the EIGRP routing protocol to prevent the introduction of unauthorized or false routing messages from unapproved sources.

Beginning in privileged EXEC mode, follow these steps to enable authentication:

	Command	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# interface <i>interface-id</i>	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	Router(config-if)# ip authentication mode eigrp <i>autonomous-system-number md5</i>	Enables MD5 authentication in IP EIGRP packets.
Step 4	Router(config-if)# ip authentication key-chain eigrp <i>autonomous-system-number key-chain</i>	Enables authentication of IP EIGRP packets.
Step 5	Router(config-if)# exit	Returns to global configuration mode.
Step 6	Router(config)# key chain <i>name-of-chain</i>	Identifies a key chain and enter key-chain configuration mode. Match the name configured in Step 4.
Step 7	Router(config-keychain)# key <i>number</i>	In key-chain configuration mode, identifies the key number.
Step 8	Router(config-keychain)# key-string <i>text</i>	In key-chain key configuration mode, identifies the key string.
Step 9	Router(config-keychain-key)# accept-lifetime <i>start-time {infinite end-time duration seconds}</i>	(Optional) Specifies the time period during which the key can be received. The <i>start-time</i> and <i>end-time</i> syntax can be either <i>hh:mm:ss Month date year</i> or <i>hh:mm:ss date Month year</i> . The default <i>start-time</i> (and earliest acceptable day) is January 1, 1993. The default <i>end-time</i> and duration is infinite.
Step 10	Router(config-keychain-key)# send-lifetime <i>start-time {infinite end-time duration seconds}</i>	(Optional) Specifies the time period during which the key can be sent. The <i>start-time</i> and <i>end-time</i> syntax can be either <i>hh:mm:ss Month day year</i> or <i>hh:mm:ss day Month year</i> . The default <i>start-time</i> (and earliest acceptable day) is January 1, 1993. The default <i>end-time</i> and duration is infinite.
Step 11	Router(config)# end	Returns to privileged EXEC mode.
Step 12	Router# show key chain	Displays authentication key information.
Step 13	Router# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Use the **no** forms of these commands to disable the feature or to return the setting to the default value.

Monitoring and Maintaining EIGRP

You can delete neighbors from the neighbor table. You can also display various EIGRP routing statistics. [Table 11-5](#) lists the privileged EXEC commands for deleting neighbors and displaying statistics. For explanations of fields in the resulting display, refer to the *Cisco IOS IP and IP Routing Command Reference* publication.

Table 11-5 IP EIGRP Clear and Show Commands

Command	Purpose
Router# clear ip eigrp neighbors {ip-address interface}	Deletes neighbors from the neighbor table.
Router# show ip eigrp interface [interface] [as-number]	Displays information about interfaces configured for EIGRP.
Router# show ip eigrp neighbors [type-number]	Displays EIGRP discovered neighbors.
Router# show ip eigrp topology {autonomous-system-number [ip-address] mask}	Displays the EIGRP topology table for a given process.
Router# show ip eigrp traffic [autonomous-system-number]	Displays the number of packets sent and received for all or a specified EIGRP process.

[Example 11-10](#) shows the output of the **show ip eigrp interface** privileged EXEC command. [Example 11-11](#) shows the output of the **show ip eigrp neighbors** privileged EXEC command. [Example 11-12](#) shows the output of the **show ip eigrp topology** privileged EXEC command. [Example 11-13](#) shows the output of the **show ip eigrp traffic** privileged EXEC command.

Example 11-10 show ip eigrp interface Privileged EXEC Command Output

```
Router# show ip eigrp interface
IP-EIGRP interfaces for process 1

      Xmit Queue   Mean   Pacing Time   Multicast   Pending
Interface  Peers  Un/Reliable  SRTT  Un/Reliable  Flow Timer  Routes
PO0         1      0/0         20    0/10         50          0
Fa0         0      0/0         0     0/10         0           0
```

Example 11-11 show ip eigrp neighbors Privileged EXEC Command Output

```
Router# show ip eigrp neighbors
IP-EIGRP neighbors for process 1
H   Address                Interface   Hold Uptime   SRTT   RTO  Q  Seq Type
                               (sec)      (ms)        Cnt Num
0   192.168.2.1              PO0        13 00:08:15   20    200  0  2
```

Example 11-12 show ip eigrp topology Privileged EXEC Command Output

```
Router# show ip eigrp topology
IP-EIGRP Topology Table for AS(1)/ID(192.168.3.1)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status
```

```
P 192.168.1.0/24, 1 successors, FD is 30720
    via 192.168.2.1 (30720/28160), POS0
P 192.168.2.0/24, 1 successors, FD is 10752
    via Connected, POS0
P 192.168.3.0/24, 1 successors, FD is 28160
    via Connected, FastEthernet0
```

Example 11-13 show ip eigrp traffic Privileged EXEC Command Output

```
Router# show ip eigrp traffic
IP-EIGRP Traffic Statistics for process 1
  Hellos sent/received: 273/136
  Updates sent/received: 5/2
  Queries sent/received: 0/0
  Replies sent/received: 0/0
  Acks sent/received: 1/2
  Input queue high water mark 1, 0 drops
  SIA-Queries sent/received: 0/0
  SIA-Replies sent/received: 0/0
```

Border Gateway Protocol and Classless Interdomain Routing

Border Gateway Protocol (BGP) is an Exterior Gateway Protocol (EGP) that allows you to set up an interdomain routing system to automatically guarantee the loop-free exchange of routing information between autonomous systems. In BGP, each route consists of a network number, a list of autonomous systems that information has passed through (called the autonomous system path), and a list of other path attributes.

Layer 3 switching supports BGP version 4, including CIDR. CIDR lets you reduce the size of your routing tables by creating aggregate routes resulting in supernets. CIDR eliminates the concept of network classes within BGP and supports the advertising of IP prefixes. CIDR routes can be carried by OSPF, EIGRP, and RIP.

Configuring BGP

To configure BGP routing, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# ip routing	Enables IP routing (default).
Step 2	Router(config)# router bgp <i>autonomous-system</i>	Defines BGP as the routing protocol and starts the BGP routing process.
Step 3	Router(config-router)# network <i>network-number</i> [mask <i>network-mask</i>] [route-map <i>route-map-name</i>]	Flags a network as local to this autonomous system and enters it in the BGP table.
Step 4	Router(config-router)# end	Returns to privileged EXEC mode.

Example 11-14 shows an example of configuring BGP routing.

Example 11-14 Configuring BGP Routing

```
Router(config)# ip routing
```

```

Router(config)# router bgp 30
Router(config-router)# network 192.168.1.1
Router(config-router)# neighbor 192.168.2.1
Router(config-router)# end

```

For more information about configuring BGP routing, refer to the “Configuring BGP” chapter in the *Cisco IOS IP and IP Routing Configuration Guide*.

Verifying the BGP Configuration

Table 11-6 lists some common EXEC commands used to view the BGP configuration. Example 11-15 shows the output of the commands listed in Table 11-6.

Table 11-6 BGP Show Commands

Command	Purpose
Router# show ip protocols [summary]	Displays the protocol configuration.
Router# show ip bgp neighbor	Displays detailed information about the BGP and TCP connections to individual neighbors.
Router# show ip bgp summary	Displays the status of all BGP connections.
Router# show ip bgp	Displays the content of the BGP routing table.

Example 11-15 BGP Configuration Information

```

Router# show ip protocols
Routing Protocol is "bgp 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  IGP synchronization is enabled
  Automatic route summarization is enabled
  Redistributing: connected
  Neighbor(s):
    Address          FiltIn FiltOut DistIn DistOut Weight RouteMap
    192.168.2.1
  Maximum path: 1
  Routing for Networks:
  Routing Information Sources:
    Gateway          Distance      Last Update
  Distance: external 20 internal 200 local 200

Router# show ip bgp neighbor
BGP neighbor is 192.168.2.1, remote AS 1, internal link
  BGP version 4, remote router ID 192.168.2.1
  BGP state = Established, up for 00:08:46
  Last read 00:00:45, hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received(new)
    Address family IPv4 Unicast: advertised and received
  Received 13 messages, 0 notifications, 0 in queue
  Sent 13 messages, 0 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Default minimum time between advertisement runs is 5 seconds

For address family: IPv4 Unicast
  BGP table version 3, neighbor version 3
  Index 1, Offset 0, Mask 0x2
  2 accepted prefixes consume 72 bytes

```

```

Prefix advertised 2, suppressed 0, withdrawn 0
Number of NLRI in the update sent: max 2, min 0

Connections established 1; dropped 0
Last reset never
Connection state is ESTAB, I/O status: 1, unread input bytes: 0
Local host: 192.168.2.2, Local port: 179
Foreign host: 192.168.2.1, Foreign port: 11001

Enqueued packets for retransmit: 0, input: 0 mis-ordered: 0 (0 bytes)

Event Timers (current time is 0x45B7B4):
Timer           Starts    Wakeups      Next
Retrans         13         0            0x0
TimeWait        0          0            0x0
AckHold         13         9            0x0
SendWnd         0          0            0x0
KeepAlive       0          0            0x0
GiveUp          0          0            0x0
PmtuAger        0          0            0x0
DeadWait        0          0            0x0

iss: 3654396253  snduna: 3654396567  sndnxt: 3654396567  sndwnd: 16071
irs: 3037331955  rcvnx: 3037332269  rcvwnd: 16071  delrcvwnd: 313

SRTT: 247 ms, RTTO: 663 ms, RTV: 416 ms, KRRT: 0 ms
minRTT: 4 ms, maxRTT: 300 ms, ACK hold: 200 ms
Flags: passive open, nagle, gen tcbs

Datagrams (max data segment is 1460 bytes):
Rcvd: 15 (out of order: 0), with data: 13, total data bytes: 313
Sent: 22 (retransmit: 0), with data: 12, total data bytes: 313

Router# show ip bgp summary
BGP router identifier 192.168.3.1, local AS number 1
BGP table version is 3, main routing table version 3
3 network entries and 4 paths using 435 bytes of memory
2 BGP path attribute entries using 120 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP activity 3/6 prefixes, 4/0 paths, scan interval 60 secs

Neighbor      V    AS MsgRcvd MsgSent  TblVer  InQ OutQ Up/Down  State/PfxRcd
192.168.2.1   4    1     14     14      3     0   0 00:09:45      2

Router# show ip bgp
BGP table version is 3, local router ID is 192.168.3.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop           Metric LocPrf Weight Path
* i192.168.1.0      192.168.2.1         0     100     0 ?
* i192.168.2.0      192.168.2.1         0     100     0 ?
*>                 0.0.0.0             0                   32768 ?
*> 192.168.3.0      0.0.0.0             0                   32768 ?

```

Configuring IS-IS

To configure Intermediate System-to-Intermediate System (IS-IS) routing, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# router isis [tag]	Defines IS-IS as the IP routing protocol.
Step 2	Router(config-router)# net network-entity-title	Configures network entity titles (NETs) for the routing process; you can specify a name for a NET as well as an address.
Step 3	Router(config-router)# interface interface-type interface-id	Enters interface configuration mode.
Step 4	Router(config-if)# ip address ip-address mask	Assigns an IP address to the interface.
Step 5	Router(config-if)# ip router isis [tag]	Specifies that this interface should run IS-IS.
Step 6	Router(config-if)# end	Returns to privileged EXEC mode.

Example 11-16 shows an example of IS-IS routing configuration.

Example 11-16 Configuring IS-IS Routing

```
Router(config)# router isis
Router(config-router)# net 49.0001.0000.0000.000a.00
Router(config-router)# interface gigabitethernet 0
Router(config-if)# ip router isis
Router(config-if)# end
```

For more information about configuring IS-IS routing, refer to the “Configuring Integrated IS-IS” chapter in the *Cisco IOS IP and IP Routing Configuration Guide*.

Verifying the IS-IS Configuration

To verify the IS-IS configuration, use the EXEC commands listed in Table 11-7. Example 11-17 shows examples of the commands in Table 11-7 and their output.

Table 11-7 IS-IS Show Commands

Command	Purpose
Router# show ip protocols [summary]	Displays the protocol configuration.
Router# show isis database	Displays the IS-IS link-state database.
Router# show clns neighbor	Displays the ES and IS neighbors.



Note

The ML Series does not support Connectionless Network Service Protocol (CLNS) routing.

Example 11-17 IS-IS Configuration

```
Router# show ip protocols
Routing Protocol is "isis"
  Invalid after 0 seconds, hold down 0, flushed after 0
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Redistributing: isis
```

```

Address Summarization:
  None
Maximum path: 4
Routing for Networks:
  FastEthernet0
  POS0
Routing Information Sources:
  Gateway          Distance    Last Update
  192.168.2.1      115        00:06:48
Distance: (default is 115)

```

```
Router# show isis database
```

```

IS-IS Level-1 Link State Database:
LSPID                LSP Seq Num  LSP Checksum  LSP Holdtime  ATT/P/OL
Router_A.00-00       0x00000003   0xA72F        581            0/0/0
Router_A.02-00       0x00000001   0xA293        581            0/0/0
Router.00-00         * 0x00000004  0x79F9        582            0/0/0
IS-IS Level-2 Link State Database:
LSPID                LSP Seq Num  LSP Checksum  LSP Holdtime  ATT/P/OL
Router_A.00-00       0x00000004   0xF0D6        589            0/0/0
Router_A.02-00       0x00000001   0x328C        581            0/0/0
Router.00-00         * 0x00000004  0x6A09        586            0/0/0

```

```
Router# show clns neighbors
```

```

System Id      Interface  SNPA                State  Holdtime  Type Protocol
Router_A       PO0        0005.9a39.6790     Up     7          L1L2 IS-IS

```

Configuring Static Routes

Static routes are user-defined routes that cause packets moving between a source and a destination to take a specified path. Static routes can be important if the router cannot build a route to a particular destination. They are also useful for specifying a gateway of last resort to which all unroutable packets are sent.

Beginning in privileged EXEC mode, follow these steps to configure a static route:

	Command	Purpose
Step 1	Router# configure terminal	Enters global configuration mode.
Step 2	Router(config)# ip route <i>prefix mask</i> { <i>address</i> <i>interface</i> } [<i>distance</i>]	Establishes a static route. Illustrated in Example 11-18 .
Step 3	Router(config)# end	Returns to privileged EXEC mode.
Step 4	Router# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Example 11-18 Static Route

```
Router(config)# ip route 0.0.0.0 0.0.0.0 192.168.2.1
```

Use the **no ip route** *prefix mask* {*address* | *interface*} global configuration command to remove a static route. Use the show ip route privileged EXEC command to view information about the static IP route ([Example 11-19](#)).

Example 11-19 show ip route Privileged EXEC Command Output (with a Static Route Configured)

```

Router# show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route
Gateway of last resort is 192.168.2.1 to network 0.0.0.0

C    192.168.2.0/24 is directly connected, POS0
C    192.168.3.0/24 is directly connected, FastEthernet0
S*   0.0.0.0/0 [1/0] via 192.168.2.1

```

The output from the **show ip route** privileged EXEC command lists codes for the routing protocols. [Table 11-8](#) shows the default administrative distances for these routing protocols.

Table 11-8 Routing Protocol Default Administrative Distances

Route Source	Default Distance
Connected interface	0
Static route	1
EIRGP summary route	5
External BGP	20
Internal EIGRP	90
OSPF	110
RIP	120
External EIGRP	170
Internal BGP	200
Unknown	225

Monitoring Static Routes

You can display statistics about static routes with the **show ip route** command ([Example 11-20](#)). For more **show ip** privileged EXEC command options and for explanations of fields in the resulting display, refer to the *Cisco IOS IP and IP Routing Command Reference* publication.

Example 11-20 show ip route Command Output (with a Static Route Configured)

```

Router# show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route
Gateway of last resort is 192.168.2.1 to network 0.0.0.0

```



```

C    192.168.2.0/24 is directly connected, POS0
C    192.168.3.0/24 is directly connected, FastEthernet0
S*  0.0.0.0/0 [1/0] via 192.168.2.1

```

Monitoring and Maintaining the IP Network

You can remove all contents of a particular cache, table, or database. You can also display specific statistics. Use the privileged EXEC commands in [Table 11-9](#) to clear routes or display status.

Table 11-9 Commands to Clear IP Routes or Display Route Status

Command	Purpose
Router# clear ip route { <i>network</i> [<i>mask</i> *]}	Clears one or more routes from the IP routing table.
Router# show ip protocols	Displays the parameters and state of the active routing protocol process.
Router# show ip route [{ <i>address</i> [<i>mask</i>] [<i>longer-prefixes</i>] [<i>protocol</i> [<i>process-id</i>]}]	Displays the current state of the routing table.
Router# show ip interface <i>interface</i>	Displays detailed information about the interface.
Router# show ip interface brief	Displays summary status information about all interfaces.
Router# show ip route summary	Displays the current state of the routing table in summary form.
Router# show ip route supernets-only	Displays supernets.
Router# show ip cache	Displays the routing table used to switch IP traffic.
Router# show route-map [<i>map-name</i>]	Displays all route maps configured or only the one specified.

Understanding IP Multicast Routing

As networks increase in size, multicast routing becomes critically important as a means to determine which segments require multicast traffic and which do not. IP multicasting allows IP traffic to be propagated from one source to a number of destinations, or from many sources to many destinations. Rather than sending one packet to each destination, one packet is sent to the multicast group identified by a single IP destination group address.

A principal component of IP multicasting is the Internet Group Management Protocol (IGMP). Hosts identify their multicast group membership by sending IGMP messages to the ML-Series card. Traffic is sent to all members of a multicast group. A host can be a member of more than one group at a time. In addition, a host does not need to be a member of a group to send data to that group. When you enable Protocol Independent Multicast (PIM) on an interface, you will have enabled IGMP operation on that same interface.

The ML-Series cards support the protocol independent multicast (PIM) routing protocol and the Auto-RP configuration.

PIM includes three different modes of behavior for dense and sparse traffic environments. These are referred to as dense mode, sparse mode, and sparse-dense mode.

PIM dense mode assumes that the downstream networks want to receive the datagrams forwarded to them. The ML-Series card forwards all packets on all outgoing interfaces until pruning and truncating occur. Interfaces that have PIM dense mode enabled receive the multicast data stream until it times out. PIM dense mode is most useful under these conditions:

- When senders and receivers are in close proximity to each other
- When the internetwork has fewer senders than receivers
- When the stream of multicast traffic is constant

PIM sparse mode assumes that the downstream networks do not want to forward multicast packets for a group unless there is an explicit request for the traffic. PIM sparse mode defines a rendezvous point, which is used as a registration point to facilitate the proper routing of packets.

When a sender wants to send data, it first sends the data to the rendezvous point. When a ML-Series card is ready to receive data, it registers with the rendezvous point. After the data stream begins to flow from the sender to the rendezvous point and then to the receiver, ML-Series cards in the data path optimize the path by automatically removing any unnecessary hops, including the rendezvous point.

PIM sparse mode is optimized for environments in which there are many multipoint data streams and each multicast stream goes to a relatively small number of LANs in the internetwork. PIM sparse mode is most useful under these conditions:

- When there are few receivers in the group
- When senders and receivers are separated by WAN links
- When the stream of multicast traffic is intermittent

**Note**

The ML-Series card support Reverse Path Forwarding (RPF) multicast, but not RPF unicast.

Configuring IP Multicast Routing

To configure IP multicast routing, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# ip multicast-routing	Enables IP multicasting on the ML-Series card.
Step 2	Router(config)# interface <i>type number</i>	Enters interface configuration mode to configure any interface.
Step 3	Router(config-if)# ip pim {dense-mode sparse-mode sparse-dense-mode}	Runs IP multicast routing on each interface on which you enter this command. You must indicate dense mode, sparse mode, or sparse-dense mode.
Step 4	Router(config)# ip pim rp-address <i>rendezvous-point ip-address</i>	Configures a rendezvous point for the multicast group.
Step 5	Router(config-if)# end	Returns to privileged EXEC mode.
Step 6	Router# copy running-config startup-config	(Optional) Saves your configuration changes to NVRAM.

Monitoring and Verifying IP Multicast Operation

After IP multicast routing is configured, you can monitor and verify its operation by performing the commands listed in [Table 11-10](#), from privileged EXEC mode.

Table 11-10 IP Multicast Routing Show Commands

Command	Purpose
Router# <code>show ip mroute</code>	Shows the complete multicast routing table and the combined statistics of packets processed.
Router# <code>show ip pim neighbor</code>	When used in EXEC mode, lists the PIM neighbors discovered by the Cisco IOS software.
Router# <code>show ip pim interface</code>	Displays information about interfaces configured for PIM.
Router# <code>show ip pim rp</code>	When used in EXEC mode, displays the active rendezvous points (RPs) that are cached with associated multicast routing entries.



Configuring IRB

This chapter describes how to configure integrated routing and bridging (IRB) for the ML-Series card. For more information about the Cisco IOS commands used in this chapter, refer to the *Cisco IOS Command Reference* publication.

This chapter includes the following major sections:

- [Understanding Integrated Routing and Bridging, page 12-1](#)
- [Configuring IRB, page 12-2](#)
- [IRB Configuration Example, page 12-3](#)
- [Monitoring and Verifying IRB, page 12-4](#)



Caution

Cisco Inter-Switch Link (ISL) and Cisco Dynamic Trunking Protocol (DTP) are not supported by the ML-Series, but the ML-Series broadcast forwards these formats. Using ISL or DTP on connecting devices is not recommended. Some Cisco devices attempt to use ISL or DTP by default.

Understanding Integrated Routing and Bridging

Your network might require you to bridge local traffic within several segments and have hosts on the bridged segments reach the hosts or ML-Series card on routed networks. For example, if you are migrating bridged topologies into routed topologies, you might want to start by connecting some of the bridged segments to the routed networks.

Using the integrated routing and bridging (IRB) feature, you can route a given protocol between routed interfaces and bridge groups within a single ML-Series card. Specifically, local or unroutable traffic is bridged among the bridged interfaces in the same bridge group, while routable traffic is routed to other routed interfaces or bridge groups.

Because bridging is in the data link layer and routing is in the network layer, they have different protocol configuration models. With IP, for example, bridge group interfaces belong to the same network and have a collective IP network address. In contrast, each routed interface represents a distinct network and has its own IP network address. Integrated routing and bridging uses the concept of a Bridge Group Virtual Interface (BVI) to enable these interfaces to exchange packets for a given protocol.

A BVI is a virtual interface within the ML-Series card that acts like a normal *routed* interface. A BVI does not support bridging but actually represents the corresponding bridge group to routed interfaces within the ML-Series card. The interface number is the link between the BVI and the bridge group.

Before configuring IRB, consider the following:

- The default routing/bridging behavior in a bridge group (when IRB is enabled) is to bridge all packets. Make sure that you explicitly configure routing on the BVI for IP traffic.
- Packets of unroutable protocols such as local-area transport (LAT) are always bridged. You cannot disable bridging for the unroutable traffic.
- Protocol attributes should not be configured on the bridged interfaces when you are using IRB to bridge and route a given protocol. You can configure protocol attributes on the BVI, but you cannot configure bridging attributes on the BVI.
- A bridge links several network segments into one large, flat network. To bridge a packet coming from a routed interface among bridged interfaces, the bridge group should be represented by one interface.
- All ports in a BVI group must have matching maximum transmission unit (MUTT) settings.

Configuring IRB

The process of configuring integrated routing and bridging consists of the following tasks:

1. Configure bridge groups and routed interfaces.
 - a. Enable bridging.
 - b. Assign interfaces to the bridge groups.
 - c. Configure the routing.
2. Enable IRB.
3. Configure the BVI.
 - a. Enable the BVI to accept routed packets.
 - b. Enable routing on the BVI.
4. Configure IP addresses on the routed interfaces.
5. Verify the IRB configuration.

When you configure the BVI and enable routing on it, packets that come in on a routed interface destined for a host on a segment that is in a bridge group are routed to the BVI and forwarded to the bridging engine. From the bridging engine, the packet exits through a bridged interface. Similarly, packets that come in on a bridged interface but are destined for a host on a routed interface go first to the BVI. The BVI forwards the packets to the routing engine that sends them out on the routed interface.

To configure a bridge group and an interface in the bridge group, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# bridge <i>bridge-group</i> protocol {ieee rstp}	Defines one or more bridge groups.
Step 2	Router(config)# interface <i>type number</i>	Enters interface configuration mode.
Step 3	Router(config-if)# bridge-group <i>bridge-group</i>	Assigns the interface to the specified bridge group.
Step 4	Router(config-if)# end	Returns to privileged EXEC mode.

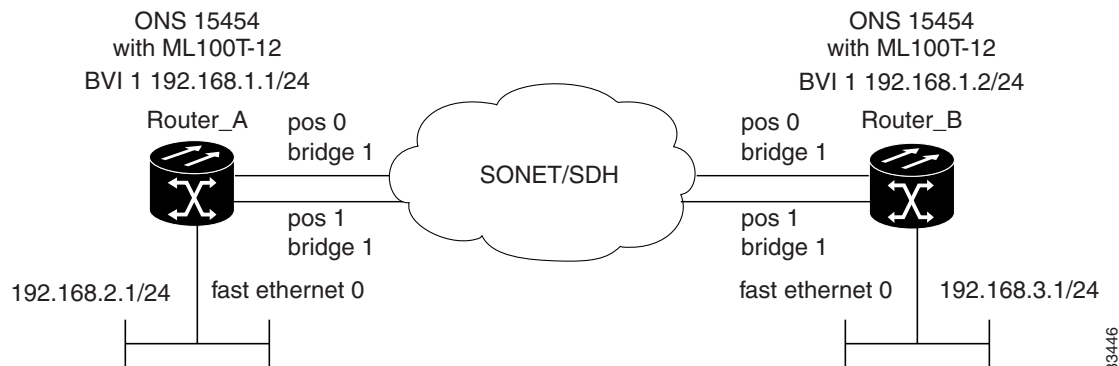
To enable and configure IRB and BVI, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# bridge irb	Enables IRB. Allows bridging of traffic.
Step 2	Router(config)# interface bvi <i>bridge-group</i>	Configures the BVI by assigning the number of the corresponding bridge group to the BVI. Each bridge group can have only one corresponding BVI.
Step 3	Router(config-if)# ip address <i>ip-address</i> <i>ip-address-subnet-mask</i>	Configures IP addresses on routed interfaces.
Step 4	Router(config-if)# exit	Exits the interface configuration mode.
Step 5	Router(config)# bridge <i>bridge-group</i> route <i>protocol</i>	Enables a BVI to accept and route routable packets received from its corresponding bridge group. Enter this command for each protocol that you want the BVI to route from its corresponding bridge group to other routed interfaces.
Step 6	Router(config)# end	Returns to the privileged EXEC mode.
Step 7	Router# copy running-config startup-config	(Optional) Saves your configuration changes to NVRAM.

IRB Configuration Example

Figure 12-1 shows an example of IRB configuration. Example 12-1 shows the configuration code for Router A, and Example 12-2 shows the configuration code for Router B.

Figure 12-1 Configuring IRB



Example 12-1 Configuring Router A

```
bridge irb
bridge 1 protocol ieee
  bridge 1 route ip
!
!
interface FastEthernet0
  ip address 192.168.2.1 255.255.255.0
```

```

!
interface POS0
  no ip address
  crc 32
bridge-group 1
  pos flag c2 1
!
interface POS1
  no ip address
  crc 32
bridge-group 1
  pos flag c2 1
!
interface BVI1
  ip address 192.168.1.1 255.255.255.0
!
router ospf 1
  log-adjacency-changes
  network 192.168.1.0 0.0.0.255 area 0
  network 192.168.2.0 0.0.0.255 area 0

```

Example 12-2 Configuring Router B

```

bridge irb
bridge 1 protocol ieee
  bridge 1 route ip
!
!
interface FastEthernet0
  ip address 192.168.3.1 255.255.255.0
!
interface POS0
  no ip address
  crc 32
bridge-group 1
  pos flag c2 1
!
interface POS1
  no ip address
  crc 32
bridge-group 1
  pos flag c2 1
!
interface BVI1
  ip address 192.168.1.2 255.255.255.0
!
router ospf 1
  log-adjacency-changes
  network 192.168.1.0 0.0.0.255 area 0
  network 192.168.3.0 0.0.0.255 area 0

```

Monitoring and Verifying IRB

Table 12-1 shows the privileged EXEC commands for monitoring and verifying IRB.

Table 12-1 Commands for Monitoring and Verifying IRB

Command	Purpose
Router# show interfaces bvi bvi-interface-number	Shows BVI information, such as the BVI MAC address and processing statistics. The bvi-interface-number is the number of the bridge group assigned to the BVI interface.
Router# show interfaces [type-number] irb	Shows BVI information for the following: <ul style="list-style-type: none"> • Protocols that this bridged interface can route to the other routed interface (if this packet is routable). • Protocols that this bridged interface bridges

The following is sample output from the **show interfaces bvi** and **show interfaces irb** commands:

Example 12-3 Monitoring and Verifying IRB

```

Router# show interfaces bvi1
BVI1 is up, line protocol is up
  Hardware is BVI, address is 0011.2130.b340 (bia 0000.0000.0000)
  Internet address is 100.100.100.1/24
  MTU 1500 bytes, BW 145152 Kbit, DLY 5000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 03:35:28, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    0 packets input, 0 bytes, 0 no buffer
    Received 0 broadcasts (0 IP multicast)
    0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    1353 packets output, 127539 bytes, 0 underruns
    0 output errors, 0 collisions, 0 interface resets
    0 output buffer failures, 0 output buffers swapped out

Router# show interfaces irb
BVI1
Software MAC address filter on BVI1
  Hash Len  Address      Matches Act   Type
  0x00:  0 ffff.ffff.ffff      0 RCV Physical broadcast
GigabitEthernet0
Bridged protocols on GigabitEthernet0:
  clns      ip
Software MAC address filter on GigabitEthernet0
  Hash Len  Address      Matches Act   Type
  0x00:  0 ffff.ffff.ffff      0 RCV Physical broadcast
  0x58:  0 0100.5e00.0006      0 RCV IP multicast
  0x5B:  0 0100.5e00.0005      0 RCV IP multicast
  0x65:  0 0011.2130.b344      0 RCV Interface MAC address
  0xC0:  0 0100.0ccc.cccc      0 RCV CDP
  0xC2:  0 0180.c200.0000      0 RCV IEEE spanning tree
POS0

```

```

Routed protocols on POS0:
  ip
Bridged protocols on POS0:
  clns      ip
Software MAC address filter on POS0
  Hash Len   Address      Matches  Act    Type
0x00:  0  ffff.ffff.ffff      9  RCV  Physical broadcast
0x58:  0  0100.5e00.0006      0  RCV  IP multicast
0x5B:  0  0100.5e00.0005     1313 RCV  IP multicast
0x61:  0  0011.2130.b340      38  RCV  Interface MAC address
0x61:  1  0011.2130.b340      0  RCV  Bridge-group Virtual Interface
0x65:  0  0011.2130.b344      0  RCV  Interface MAC address
0xC0:  0  0100.0ccc.cccc     224  RCV  CDP
0xC2:  0  0180.c200.0000      0  RCV  IEEE spanning tree
POS1
SPR1
Bridged protocols on SPR1:
  clns      ip
Software MAC address filter on SPR1
  Hash Len   Address      Matches  Act    Type
0x00:  0  ffff.ffff.ffff      0  RCV  Physical broadcast
0x60:  0  0011.2130.b341      0  RCV  Interface MAC address
0x65:  0  0011.2130.b344      0  RCV  Interface MAC address
0xC0:  0  0100.0ccc.cccc      0  RCV  CDP
0xC2:  0  0180.c200.0000      0  RCV  IEEE spanning tree

```

Table 12-1 describes significant fields shown in the display.

Table 12-2 *show interfaces irb Field Descriptions*

Field	Description
Routed protocols on...	List of the routed protocols configured for the specified interface.
Bridged protocols on...	List of the bridged protocols configured for the specified interface.
Software MAC address filter on...	Table of software MAC address filter information for the specified interface.
Hash	Hash key/relative position in the keyed list for this MAC-address entry.
Len	Length of this entry to the beginning element of this hash chain.
Address	Canonical (Ethernet ordered) MAC address.
Matches	Number of received packets matched to this MAC address.
Routed protocols on...	List of the routed protocols configured for the specified interface.
Bridged protocols on...	List of the bridged protocols configured for the specified interface.



Configuring VRF Lite

This chapter describes how to configure VPN Routing and Forwarding Lite (VRF Lite) for the ML-Series cards. For additional information about the Cisco IOS commands used in this chapter, refer to the *Cisco IOS Command Reference* publication. This chapter contains the following major sections:

- [Understanding VRF Lite, page 13-1](#)
- [Configuring VRF Lite, page 13-2](#)
- [VRF Lite Configuration Example, page 13-3](#)
- [Monitoring and Verifying VRF Lite, page 13-7](#)



Note

If you have already configured bridging, you may now proceed with configuring VRF Lite as an optional step.

Understanding VRF Lite

VRF is an extension of IP routing that provides multiple routing instances. It provides a separate IP routing and forwarding table to each VPN and is used in concert with MP-iBGP (Multi-Protocol internal BGP) between provider equipment (PE) routers to provide Layer 3 MPLS-VPN. However, ML-Series VRF implementation is without MP-iBGP. With VRF Lite, the ML Series is considered a PE-extension or a customer equipment (CE)-extension. VRF Lite is considered a PE-extension since it has VRF (but without MP-iBGP), and it is considered a CE-extension since this CE can have multiple VRFs and serves many customer with one CE box.

Under VRF Lite, an ML-Series CE can have multiple interfaces/subinterfaces with PE for different customers (while a normal CE is only for one customer). It holds VRFs (routing information) locally and it does not distribute the VRFs to its connected PE. It uses VRF information to direct traffic to the correct interfaces/subinterfaces when it receives traffic from customers' routers or from Internet service provider (ISP) PE router(s).

Configuring VRF Lite

Perform the following procedure to configure VRF Lite:

	Command	Purpose
Step 1	Router(config)# ip vrf <i>vrf-name</i>	Enters VRF configuration mode and assigns a VRF name.
Step 2	Router(config-vrf)# rd <i>route-distinguisher</i>	Creates a VPN route distinguisher (RD). An RD creates routing and forwarding tables and specifies the default route distinguisher for a VPN. The RD is added to the beginning of the customer's IPv4 prefixes to change them into globally unique VPN-IPv4 prefixes. Either RD is an ASN-relative RD, in which case it is composed of an autonomous system number and an arbitrary number, or it is an IP-address-relative RD, in which case it is composed of an IP address and an arbitrary number. You can enter a <i>route-distinguisher</i> in either of these formats: 16-bit AS number: your 32-bit number For example, 101:3. 32-bit IP address: your 16-bit number For example, 192.168.122.15:1.
Step 3	Router(config-vrf)# route-target { import export both } <i>route-distinguisher</i>	Creates a list of import and/or export route target communities for the specified VRF.
Step 4	Router(config-vrf)# import map <i>route-map</i>	(Optional) Associates the specified route map with the VRF.
Step 5	Router(config-vrf)# exit	Exits the current configuration mode and enters global configuration mode.
Step 6	Router(config)# interface type number	Specifies an interface and enters interface configuration mode.
Step 7	Router(config-vrf)# ip vrf forwarding <i>vrf-name</i>	Associates a VRF with an interface or subinterface.
Step 8	Router(config-if)# end	Exits to privileged EXEC mode.
Step 9	Router# copy running-config startup-config	(Optional) Saves configuration changes to NVRAM.

[Example 13-1](#) shows an example of configuring a VRF. In the example, the VRF name is `customer_a`, the route-distinguisher is `1:1`, and the interface type is Fast Ethernet, number `0.1`.

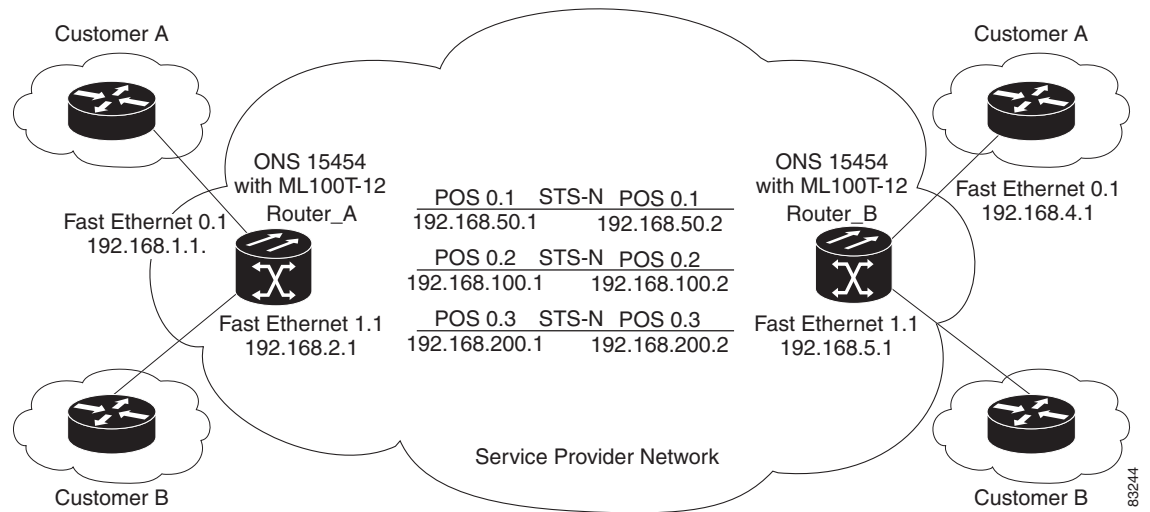
Example 13-1 Configuring a VRF

```
Router(config)# ip vrf customer_a
Router(config-vrf)# rd 1:1
Router(config-vrf)# route-target both 1:1
Router(config)# interface fastEthernet 0.1
Router(config-subif)# ip vrf forwarding customer_a
```

VRF Lite Configuration Example

Figure 13-1 shows an example of a VRF Lite configuration. The configurations for Router A and Router B are provided in Example 13-2 and Example 13-3 on page 13-4, respectively. The associated routing tables are shown in Example 13-4 on page 13-5 through Example 13-9 on page 13-7.

Figure 13-1 VRF Lite—Sample Network Scenario



Example 13-2 Router A Configuration

```
hostname Router_A
!
ip vrf customer_a
 rd 1:1
  route-target export 1:1
  route-target import 1:1
!
ip vrf customer_b
 rd 2:2
  route-target export 2:2
  route-target import 2:2
!
bridge 1 protocol ieee
bridge 2 protocol ieee
bridge 3 protocol ieee
!
!
interface FastEthernet0
 no ip address
!
interface FastEthernet0.1
 encapsulation dot1Q 2
 ip vrf forwarding customer_a
 ip address 192.168.1.1 255.255.255.0
 bridge-group 2
!
interface FastEthernet1
 no ip address
!
```

```

interface FastEthernet1.1
  encapsulation dot1Q 3
  ip vrf forwarding customer_b
  ip address 192.168.2.1 255.255.255.0
  bridge-group 3
!
interface POS0
  no ip address
  crc 32
  no cdp enable
  pos flag c2 1
!
interface POS0.1
  encapsulation dot1Q 1 native
  ip address 192.168.50.1 255.255.255.0
  bridge-group 1
!
interface POS0.2
  encapsulation dot1Q 2
  ip vrf forwarding customer_a
  ip address 192.168.100.1 255.255.255.0
  bridge-group 2
!
interface POS0.3
  encapsulation dot1Q 3
  ip vrf forwarding customer_b
  ip address 192.168.200.1 255.255.255.0
  bridge-group 3
!
router ospf 1
  log-adjacency-changes
  network 192.168.50.0 0.0.0.255 area 0
!
router ospf 2 vrf customer_a
  log-adjacency-changes
  network 192.168.1.0 0.0.0.255 area 0
  network 192.168.100.0 0.0.0.255 area 0
!
router ospf 3 vrf customer_b
  log-adjacency-changes
  network 192.168.2.0 0.0.0.255 area 0
  network 192.168.200.0 0.0.0.255 area 0
!

```

Example 13-3 Router_B Configuration

```

hostname Router_B
!
ip vrf customer_a
rd 1:1
  route-target export 1:1
  route-target import 1:1
!
ip vrf customer_b
rd 2:2
  route-target export 2:2
  route-target import 2:2
!
bridge 1 protocol ieee
bridge 2 protocol ieee
bridge 3 protocol ieee
!

```

```

!
interface FastEthernet0
  no ip address
!
interface FastEthernet0.1
  encapsulation dot1Q 2
  ip vrf forwarding customer_a
  ip address 192.168.4.1 255.255.255.0
  bridge-group 2
!
interface FastEthernet1
  no ip address
!
interface FastEthernet1.1
  encapsulation dot1Q 3
  ip vrf forwarding customer_b
  ip address 192.168.5.1 255.255.255.0
  bridge-group 3
!
interface POS0
  no ip address
  crc 32
  no cdp enable
  pos flag c2 1
!
interface POS0.1
  encapsulation dot1Q 1 native
  ip address 192.168.50.2 255.255.255.0
  bridge-group 1
!
interface POS0.2
  encapsulation dot1Q 2
  ip vrf forwarding customer_a
  ip address 192.168.100.2 255.255.255.0
  bridge-group 2
!
interface POS0.3
  encapsulation dot1Q 3
  ip vrf forwarding customer_b
  ip address 192.168.200.2 255.255.255.0
  bridge-group 3
!
router ospf 1
  log-adjacency-changes
  network 192.168.50.0 0.0.0.255 area 0
!
router ospf 2 vrf customer_a
  log-adjacency-changes
  network 192.168.4.0 0.0.0.255 area 0
  network 192.168.100.0 0.0.0.255 area 0
!
router ospf 3 vrf customer_b
  log-adjacency-changes
  network 192.168.5.0 0.0.0.255 area 0
  network 192.168.200.0 0.0.0.255 area 0
!

```

Example 13-4 Router_A Global Routing Table

```

Router_A# sh ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

```

```

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route

```

Gateway of last resort is not set

```
C 192.168.50.0/24 is directly connected, POS0.1
```

Example 13-5 Router_A customer_a VRF Routing Table

```

Router_A# show ip route vrf customer_a
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route

```

Gateway of last resort is not set

```

O 192.168.4.0/24 [110/2] via 192.168.100.2, 00:15:35, POS0.2
C 192.168.1.0/24 is directly connected, FastEthernet0.1
C 192.168.100.0/24 is directly connected, POS0.2

```

Example 13-6 Router_A customer_b VRF Routing Table

```

Router_A# show ip route vrf customer_b
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route

```

Gateway of last resort is not set

```

C 192.168.200.0/24 is directly connected, POS0.3
O 192.168.5.0/24 [110/2] via 192.168.200.2, 00:10:32, POS0.3
C 192.168.2.0/24 is directly connected, FastEthernet1.1

```

Example 13-7 Router_B Global Routing Table

```

Router_B# sh ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route

```

Gateway of last resort is not set

```
C 192.168.50.0/24 is directly connected, POS0.1
```


Example 13-8 Router_B customer_a VRF Routing Table

```

Router_B# sh ip route vrf customer_a
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.4.0/24 is directly connected, FastEthernet0.1
O    192.168.1.0/24 [110/2] via 192.168.100.1, 00:56:24, POS0.2
C    192.168.100.0/24 is directly connected, POS0.2

```

Example 13-9 Router_B customer_b VRF Routing Table

```

Router_B# show ip route vrf customer_b
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.200.0/24 is directly connected, POS0.3
C    192.168.5.0/24 is directly connected, FastEthernet1.1
O    192.168.2.0/24 [110/2] via 192.168.200.1, 00:10:51, POS0.3

```

Monitoring and Verifying VRF Lite

Table 13-1 shows the privileged EXEC commands for monitoring and verifying VRF Lite.

Table 13-1 Commands for Monitoring and Verifying VRF Lite

Command	Purpose
Router# show ip vrf	Displays the set of VRFs and interfaces.
Router# show ip route vrf vrf-name	Displays the IP routing table for a VRF.
Router# show ip protocols vrf vrf-name	Displays the routing protocol information for a VRF.
Router# ping vrf vrf-name ip ip-address	Pings an IP address that has a specific VRF.



Configuring Quality of Service

This chapter describes the Quality of Service (QoS) features built into your ML-Series card and how to map QoS scheduling at both the system and interface levels.

This chapter contains the following major sections:

- [Understanding QoS, page 14-1](#)
- [ML-Series QoS, page 14-4](#)
- [QoS on RPR, page 14-10](#)
- [Configuring QoS, page 14-10](#)
- [Monitoring and Verifying QoS Configuration, page 14-16](#)
- [QoS Configuration Examples, page 14-17](#)
- [Understanding Multicast QoS and Multicast Priority Queuing, page 14-23](#)
- [Configuring Multicast Priority Queuing QoS, page 14-24](#)
- [QoS not Configured on Egress, page 14-26](#)
- [ML-Series Egress Bandwidth Example, page 14-26](#)
- [Understanding CoS-Based Packet Statistics, page 14-28](#)
- [Configuring CoS-Based Packet Statistics, page 14-28](#)

The ML-Series card employs the Cisco IOS Modular QoS CLI (MQC). For more information on general MQC configuration, refer to the following Cisco IOS documents:

- *Cisco IOS Quality of Service Solutions Configuration Guide, Release 12.2* at this URL:
<http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122mindx/122index.htm>
- *Cisco IOS Quality of Service Solutions Command Reference, Release 12.2* at this URL:
http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fqos_r/index.htm

Understanding QoS

The ML-Series card multiplexes multiple IP/Ethernet services onto the SONET/SDH circuit and dynamically allocates transmission bandwidth to data services based on data service requirements. QoS enables significantly higher levels of network utilization. To support service-level agreements (SLAs), this dynamic allocation must accommodate the service elements of bandwidth, including loss and delay. The characteristics of these service elements make up QoS.

The QoS mechanism does three basic steps to meet these service elements. It classifies types of traffic, specifies what action to take against a type of traffic, and specifies where the action should take place. The following sections explain how the ML-Series card accomplishes these steps for unicast traffic. QoS for multicast traffic and traffic with unknown destination addresses is handled with a different mechanism detailed in the [“Understanding Multicast QoS and Multicast Priority Queuing”](#) section on [page 14-23](#).

Priority Mechanism in IP and Ethernet

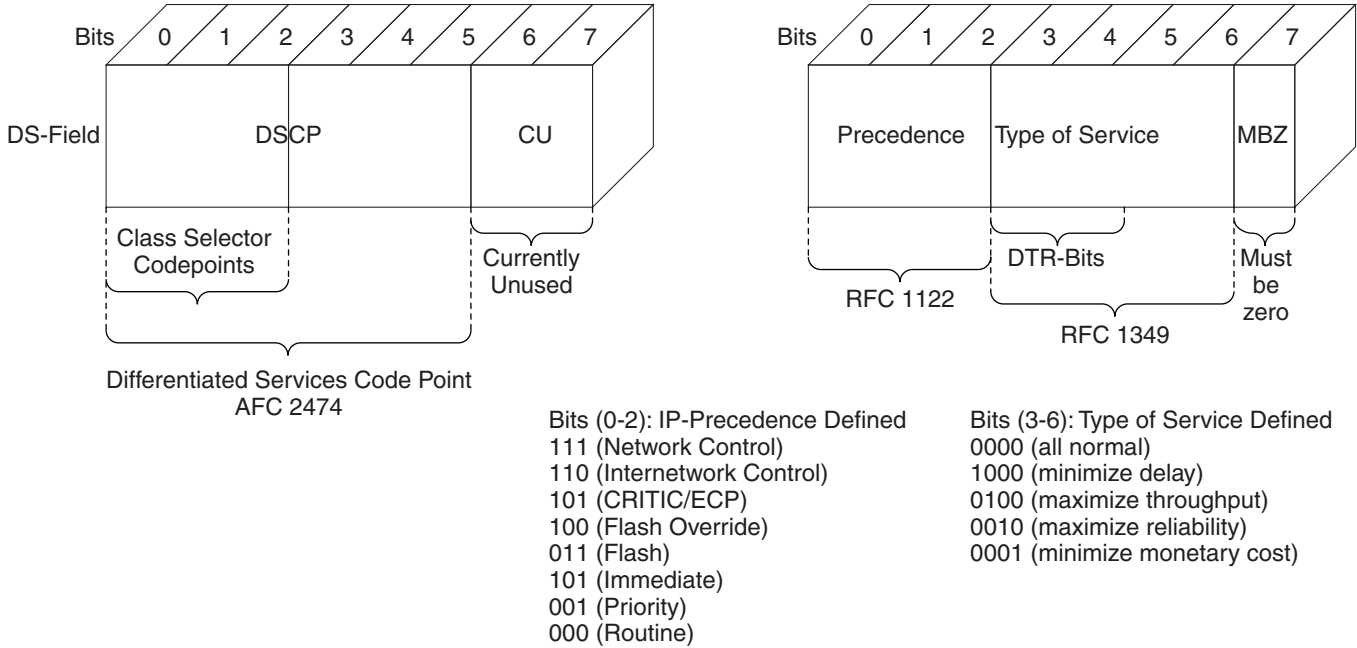
For any QoS service to be applied to data, there must be a way to mark or identify an IP packet or an Ethernet frame. When identified, a specific priority can be assigned to each individual IP packet or Ethernet frame. The IP Precedence or the IP Differentiated Services Code Point (DSCP) field prioritizes the IP packets, and the Ethernet class of service (IEEE 802.1p defined class of service [CoS]) is used for the Ethernet frames. IP precedence and Ethernet CoS are further described in the following sections.

IP Precedence and Differentiated Services Code Point

IP precedence uses the three precedence bits in the IPv4 header's ToS (type of service) field to specify class of service for each IP packet (RFC 1122). The most significant three bits on the IPv4 ToS field provides up to eight distinct classes, of which six are used for classifying services and the remaining two are reserved. On the edge of the network, the IP precedence is assigned by the client device or the router, so that each subsequent network element can provide services based on the determined policy or the service level agreement (SLA).

IP DSCP uses the six bits in the IPv4 header to specify class of service for each IP packet (RFC 2474). [Figure 14-1](#) illustrates IP precedence and DSCP. The DSCP field classifies packets into any of the 64 possible classes. On the network edge the IP DSCP is assigned by the client device or the router, so that each subsequent network element can provide services based on the determined policy or the SLA.

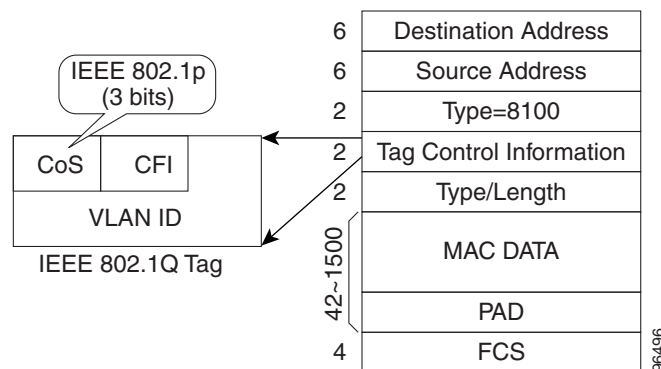
Figure 14-1 IP Precedence and DSCP



Ethernet CoS

Ethernet CoS refers to three bits within a four byte IEEE 802.1Q (VLAN) header used to indicate the priority of the Ethernet frame as it passes through a switched network. The CoS bits in the IEEE 802.1Q header are commonly referred to as the IEEE 802.1p bits. There are three CoS bits that provide eight classes, matching the number delivered by IP precedence. In many real-world networks, a packet might traverse both Layer 2 and Layer 3 domains. To maintain QoS across the network, the IP ToS can be mapped to the Ethernet CoS and vice versa, for example, in linear or one-to-one mapping, because each mechanism supports eight classes. Similarly, a set of DSCP values (64 classes) can be mapped into each of the eight individual Ethernet CoS values. Figure 14-2 is an IEEE 802.1Q Ethernet frame, which consists of a 2-byte Ethertype and a 2-byte tag (IEEE 802.1Q Tag) on the Ethernet protocol header.

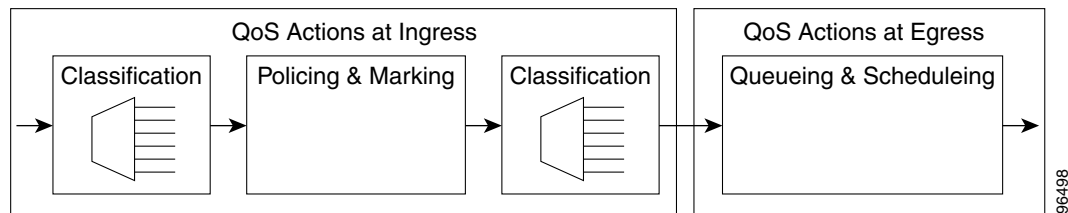
Figure 14-2 Ethernet Frame and the CoS Bit (IEEE 802.1p)



ML-Series QoS

The ML-Series QoS classifies each packet in the network based on its input interface, bridge group (VLAN), Ethernet CoS, IP precedence, IP DSCP, or RPR-CoS. After they are classified into class flows, further QoS functions can be applied to each packet as it traverses the card. Figure 14-3 illustrates the ML-Series QoS flow.

Figure 14-3 ML-Series QoS Flow



Policing provided by the ML-Series card ensures that attached equipment does not submit more than a predefined amount of bandwidth (Rate Limiting) into the network. The policing feature can be used to enforce the committed information rate (CIR) and the peak information rate (PIR) available to a customer at an interface. Policing also helps characterize the statistical nature of the information allowed into the network so that traffic engineering can more effectively ensure that the amount of committed bandwidth is available on the network, and the peak bandwidth is over-subscribed with an appropriate ratio. The policing action is applied per classification.

Priority marking can set the Ethernet IEEE 802.1p CoS bits or RPR-CoS bits as they exit the ML-Series card. The marking feature operates on the outer IEEE 802.1p tag, and provides a mechanism for tagging packets at the ingress of a QinQ packet. The subsequent network elements can provide QoS based only on this service-provider-created QoS indicator.

Per-class flow queuing enables fair access to excess network bandwidth, allows allocation of bandwidth to support SLAs, and ensures that applications with high network resource requirements are adequately served. Buffers are allocated to queues dynamically from a shared resource pool. The allocation process incorporates the instantaneous system load as well as the allocated bandwidth to each queue to optimize buffer allocation. Congestion management on the ML-Series is performed through a tail drop mechanism along with discard eligibility on the egress scheduler.

The ML-Series uses a Weighted Deficit Round Robin (WDRR) scheduling process to provide fair access to excess bandwidth as well as guaranteed throughput to each class flow.

Admission control is a process that is invoked each time that service is configured on the ML-Series card to ensure that QoS resources are not overcommitted. In particular, admission control ensures that no configurations are accepted, where a sum of the committed bandwidths on an interface exceeds total bandwidth on the interface.

Classification

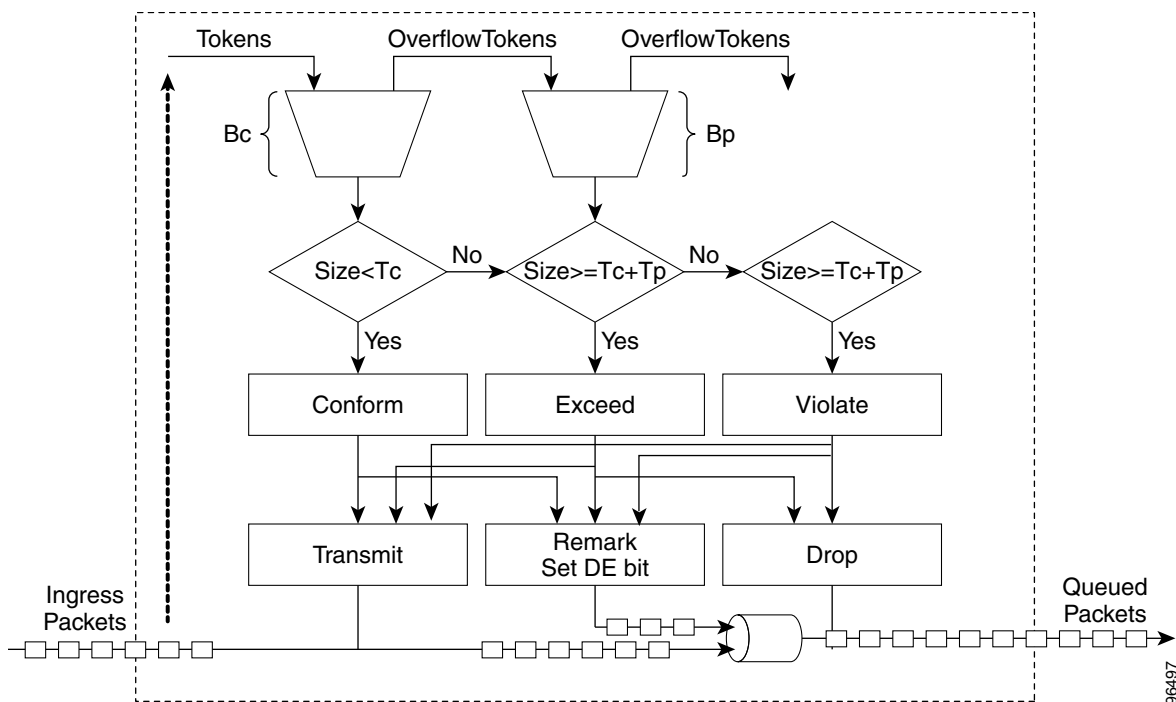
Classification can be based on any single packet classification criteria or a combination (logical AND and OR). A total of 254 classes, not including the class default, can be defined on the card. Classification of packets is configured using the Modular CLI **class-map** command. For traffic transiting the resilient packet ring (RPR), only the input interface and/or the RPR-CoS can be used as classification criteria.

Policing

Dual leaky bucket policer is a process where the first bucket (CIR bucket) is filled with tokens at a known rate (CIR), which is a parameter that can be configured by the operator. Figure 14-4 illustrates the dual leaky bucket policer model. The tokens fill the bucket up to a maximum level, which is the amount of burstable committed (BC) traffic on the policer. The nonconforming packets of the first bucket are the overflow packets, which are passed to the second leaky bucket (the PIR bucket). The second leaky bucket is filled with these tokens at a known rate (PIR), which is a parameter that can be configured by the operator. The tokens fill the PIR bucket up to a maximum level (BP), which is the amount of peak burstable traffic on the policer. The nonconform packets of the second bucket are the overflow packets, which can be dropped or marked according to the policer definition.

On the dual leaky bucket policer, the packets conforming to the CIR are conform packets, the packets not conforming to CIR but conforming to PIR are exceed packets, and the packets not conforming to either the PIR or CIR are violate packets.

Figure 14-4 Dual Leaky Bucket Policer Model



Marking and Discarding with a Policer

On the ML-Series card's policer, the conform packets can be transmitted or marked and transmitted. The exceed packets can be transmitted, marked and transmitted, or dropped. The violating packets can be transmitted, marked and transmitted, or dropped. The primary application of the dual-rate or three-color policer is to mark the conform packets with CoS bit 21, mark the exceed packet with CoS bit 1, and discard the violated packets so all the subsequent network devices can implement the proper QoS treatment per frame/packet basis based on these priority marking without knowledge of each SLA.

In some cases, it may be desirable to discard all traffic of a specific ingress class. This can be accomplished by using a police command of the following form with the class: **police 96000 conform-action drop exceed-action drop**.

If a marked packet has a provider-supplied Q-tag inserted before transmission, the marking only affects the provider Q-tag. If a Q-tag is received, it is re-marked. If a marked packet is transported over the RPR ring, the marking also affects the RPR-CoS bit.

If a Q-tag is inserted (QinQ), the marking affects the added Q-tag. If the ingress packet contains a Q-tag and is transparently switched, the existing Q-tag is marked. In case of a packet without any Q-tag, the marking does not have any significance.

The local scheduler treats all nonconforming packets as discard eligible regardless of their CoS setting or the global cos commit definition. For RPR implementation, the discard eligible (DE) packets are marked using the DE bit on the RPR header. The discard eligibility based on the CoS commit or the policing action is local to the ML-Series card scheduler, but it is global for the RPR ring.

Queuing

ML-Series card queuing uses a shared buffer pool to allocate memory dynamically to different traffic queues. The ML-Series card uses a total of 12 MB memory for the buffer pool. Ethernet ports share 6 MB of the memory, and packet-over-SONET/SDH (POS) ports share the remaining 6 MBs of memory. Memory space is allocated in 1500-byte increments.

Each queue has an upper limit on the allocated number of buffers based on the class bandwidth assignment of the queue and the number of queues configured. This upper limit is typically 30 percent to 50 percent of the shared buffer capacity. Dynamic buffer allocation to each queue can be reduced based on the number of queues needing extra buffering. The dynamic allocation mechanism provides fairness in proportion to service commitments as well as optimization of system throughput over a range of system traffic loads.

The Low Latency Queue (LLQ) is defined by setting the weight to infinity or committing 100 percent bandwidth. When a LLQ is defined, a policer should also be defined on the ingress for that specific class to limit the maximum bandwidth consumed by the LLQ; otherwise there is a potential risk of LLQ occupying the whole bandwidth and starving the other unicast queues.

The ML-Series includes support for 400 user-definable queues, which are assigned per the classification and bandwidth allocation definition. The classification used for scheduling classifies the frames/packet after the policing action, so if the policer is used to mark or change the CoS bits of the ingress frames/packet, the new values are applicable for the classification of traffic for queuing and scheduling. The ML-Series provides buffering for 4000 packets.

Scheduling

Scheduling is provided by a series of schedulers that perform a WDRR as well as priority scheduling mechanisms from the queued traffic associated with each egress port.

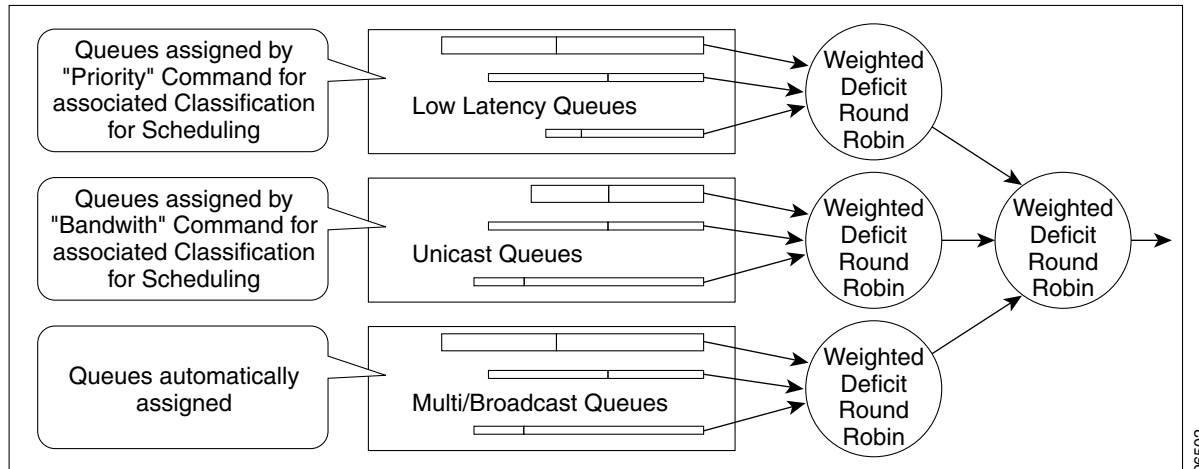
Though ordinary round robin servicing of queues can be done in constant time, unfairness occurs when different queues use different packet sizes. Deficit Round Robin (DRR) scheduling solves this problem. If a queue was not able to send a packet in its previous round because its packet size was too large, the remainder from the previous amount of credits a queue gets in each round (quantum) is added to the quantum for the next round.

WDRR extends the quantum idea from the DRR to provide weighted throughput for each queue. Different queues have different weights, and the quantum assigned to each queue in its round is proportional to the relative weight of the queue among all the queues serviced by that scheduler.

Weights are assigned to each queue as a result of the service provisioning process. When coupled with policing and policy mapping provisioning, these weights and the WDRR scheduling process ensure that QoS commitments are provided to each service flow.

Figure 14-5 illustrates the ML-Series card's queuing and scheduling.

Figure 14-5 Queuing and Scheduling Model



The weighting structure allows traffic to be scheduled at 1/2048 of the port rate. This equates to approximately 488 kbps for traffic exiting a Gigabit Ethernet port, approximately 293 kbps for traffic exiting an OC-12c port, and approximately 49 kbps for traffic exiting a FastEthernet port.

The unicast queues are created as the output service policy implementation on the egress ports. Each unicast queue is assigned with a committed bandwidth and the weight of the queue is determined by the normalization of committed bandwidth of all defined unicast queues for that port. The traffic beyond the committed bandwidth on any queue is treated by the scheduler according to the relative weight of the queue.

The LLQ is created as the output service policy implementation on the egress ports. Each LLQ queue is assigned with a committed bandwidth of 100 percent and is served with lower latency. To limit the bandwidth usage by the LLQ, a strict policer needs to be implemented on the ingress for the LLQ traffic classes.

The DE allows some packets to be treated as committed and some as discard-eligible on the scheduler. For the Ethernet frames, the CoS (IEEE 802.1p) bits are used to identify committed and discard eligible packets, where the RPR-CoS and the DE bits are used for RPR traffic. When congestion occurs and a queue begins to fill, the DE packets hit a lower tail-drop threshold than the committed packets. Committed packets are not dropped until the total committed load exceeds the interface output. The tail-drop thresholds adjust dynamically in the card to maximize use of the shared buffer pool while guaranteeing fairness under all conditions.

Control Packets and L2 Tunneled Protocols

The control packets originated by the ML-Series card have a higher priority than data packets. The external Layer 2 and Layer 3 control packets are handled as data packets and assigned to broadcast queues. Bridge protocol data unit (BPDU) prioritization in the ML-Series card gives Layer 2-tunneled BPDU sent out the multicast/broadcast queue a higher discard value and therefore a higher priority than other packets in the multicast/broadcast queue. The Ethernet CoS (IEEE 802.1p) for Layer 2-tunneled protocols can be assigned by the ML-Series card.

Egress Priority Marking

Egress priority marking allows the operator to assign the IEEE 802.1p CoS bits of packets that exit the card. This marking allows the operator to use the CoS bits as a mechanism for signaling to downstream nodes the QoS treatment the packet should be given. This feature operates on the outer-most IEEE 802.1p CoS field. When used with the QinQ feature, priority marking allows the user traffic (inner Q-tag) to traverse the network transparently, while providing a means for the network to internally signal QoS treatment at Layer 2.

Priority marking follows the classification process, and therefore any of the classification criteria identified earlier can be used as the basis to set the outgoing IEEE 802.1p CoS field. For example, a specific CoS value can be mapped to a specific bridge group.

Priority marking is configured using the MQC **set-cos** command. If packets would otherwise leave the card without an IEEE 802.1Q tag, then the **set-cos** command has no effect on that packet. If an IEEE 802.1Q tag is inserted in the packet (either a normal tag or a QinQ tag), the inserted tag has the set-cos priority. If an IEEE 802.1Q tag is present on packet ingress and retained on packet egress, the priority of that tag is modified. If the ingress interface is an QinQ access port, and the **set-cos** policy-map classifies based on ingress tag priority, this classifies based on the user priority. This is a way to allow the user-tag priority to determine the SP tag priority. When a packet does not match any **set-cos** policy-map, the priority of any preserved tag is unchanged and the priority of any inserted IEEE 802.1Q tag is set to 0.

The **set-cos** command on the output service policy is only applied to unicast traffic. Priority marking for multicast/broadcast traffic can only be achieved by the **set-cos** action of the policing process on the input service policy.

Ingress Priority Marking

Ingress priority marking can be done for all input packets of a port, for all input packets matching a classification, or based on a measured rate. Marking of all packets of an input class can also be done with a policing command of the form "**police 96000 conform-action set-cos-transmit exceed-action set-cos-transmit**". Using this command with a policy map that contains only the "class-default" will mark all ingress packets to the value. Rate based priority marking is discussed in the section "[Marking and Discarding with a Policer](#)" section on page 14-5.

QinQ Implementation

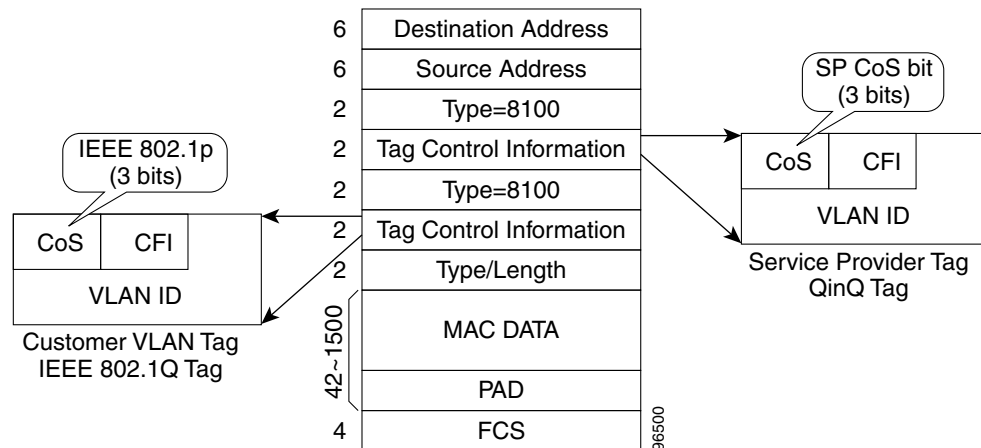
The hierarchical VLAN or IEEE 802.1Q tunneling feature enables the service provider to transparently carry the customer VLANs coming from any specific port (UNI) and transport them over the service provider network. This feature is also known as QinQ, which is performed by adding an additional IEEE 802.1Q tag on every customer frame.

Using the QinQ feature, service providers can use a single VLAN to support customers with multiple VLANs. QinQ preserves customer VLAN IDs and segregates traffic from different customers within the service-provider infrastructure, even when traffic from different customers originally shared the same VLAN ID. The QinQ also expands VLAN space by using a VLAN-in-VLAN hierarchy and tagging the tagged packets. When the service provider (SP) tag is added, the QinQ network typically loses any visibility to the IP header or the customer Ethernet IEEE 802.1Q tag on the QinQ encapsulated frames.

On the ML-Series cards, the QinQ access ports (IEEE 802.1Q tunnel ports or QinQ UNI ports) have visibility to the customer CoS and the IP precedence or IP DSCP values; therefore, the SP tag can be assigned with proper CoS bit which would reflect the customer IP precedence, IP DSCP, or CoS bits. In the QinQ network, the QoS is then implemented based on the IEEE 802.1p bit of the SP tag. The ML-Series cards do not have visibility into the customer CoS, IP precedence, or DSCP values after the packet is double-tagged (because it is beyond the entry point of the QinQ service).

Figure 14-6 illustrates the QinQ implementation on the ML-Series card.

Figure 14-6 QinQ



The ML-Series cards can be used as the IEEE 802.1Q tunneling device for the QinQ network and also provide the option to copy the customer frame's CoS bit into the CoS bit of the added QinQ tag. This way the service provider QinQ network can be fully aware of the necessary QoS treatment for each individual customer frame.

Flow Control Pause and QoS

If flow control and port-based policing are both enabled for an interface, flow control handles the bandwidth. If the policer gets noncompliant flow, then the policer drops or demarks the packets using the policer definition of the interface.



Note

QoS and policing are not supported on the ML-Series card interface when link aggregation is used.



Note

Egress shaping is not supported on the ML-Series cards.

QoS on RPR

For VLAN bridging over RPR, all ML-Series cards on the ring must be configured with the base RPR and RPR QoS configuration. SLA and bridging configurations are only needed at customer RPR access points, where IEEE 802.1Q VLAN CoS is copied to the RPR CoS. This IEEE 802.1Q VLAN CoS copying can be overwritten with a **set-cos** *action* command. The CoS commit rule applies at RPR ring ingress. Transit RPR ring traffic is classified on CoS only.

If the packet does not have a VLAN header, the RPR CoS for non-VLAN traffic is set using the following rules:

1. The default CoS is 0.
2. If the packet comes in with an assigned CoS, the assigned CoS replaces the default. If an IP packet originates locally, the IP precedence setting replaces the CoS setting.
3. The input policy map has a **set-cos** action.
4. The output policy map has a **set-cos** action (except for broadcast or multicast packets).

The RPR header contains a CoS value and DE indicator. The RPR DE is set for noncommitted traffic.

Configuring QoS

This section describes the tasks for configuring the ML-Series card QoS functions using the Modular Quality of Service Command-Line Interface (MQC). The ML-Series card does not support the full set of MQC functionality.

To configure and enable class-based QoS features, perform the procedures described in the following sections:

- [Creating a Traffic Class, page 14-10](#)
- [Creating a Traffic Policy, page 14-11](#)
- [Attaching a Traffic Policy to an Interface, page 14-15](#)
- [Configuring CoS-Based QoS, page 14-16](#)

For QoS configuration examples, see the “[QoS Configuration Examples](#)” section on page 14-17.

Creating a Traffic Class

The **class-map** global configuration command is used to create a traffic class. The syntax of the **class-map** command is as follows:

```
class-map [match-any | match-all] class-map-name
no class-map [match-any | match-all] class-map-name
```

The **match-all** and **match-any** options need to be specified only if more than one match criterion is configured in the traffic class. The **class-map match-all** command is used when all of the match criteria in the traffic class must be met for a packet to match the specified traffic class. The **class-map match-any** command is used when only one of the match criterion in the traffic class must be met for a packet to match the specified traffic class. If neither the **match-all** nor **match-any** keyword is specified, the traffic class behaves in a manner consistent with **class-map match-all** command.

To create a traffic class containing match criteria, use the **class-map** global configuration command to specify the traffic class name, and then use the following **match** commands in [Table 14-1](#), as needed:

Table 14-1 Traffic Class Commands

Command	Purpose
Router(config)# class-map <i>class-map-name</i>	Specifies the user-defined name of the traffic class. Names can be a maximum of 40 alphanumeric characters. If match-all or match-any is not specified, traffic must match all the match criteria to be classified as part of the traffic class. There is no default-match criteria. Multiple match criteria are supported. The command matches either all or any of the criteria, as controlled by the match-all and match-any subcommands of the class-map command.
Router(config)# class-map match-all <i>class-map-name</i>	Specifies that all match criteria must be met for traffic entering the traffic class to be classified as part of the traffic class.
Router(config)# class-map match-any <i>class-map-name</i>	Specifies that one of the match criteria must be met for traffic entering the traffic class to be classified as part of the traffic class.
Router(config-cmap)# match any	Specifies that all packets will be matched.
Router(config-cmap)# match bridge-group <i>bridge-group-number</i>	Specifies the bridge-group-number against whose contents packets are checked to determine if they belong to the class.
Router(config-cmap)# match cos <i>cos-number</i>	Specifies the CoS value against whose contents packets are checked to determine if they belong to the class.
Router(config-cmap)# match input-interface <i>interface-name</i>	Specifies the name of the input interface used as a match criterion against which packets are checked to determine if they belong to the class. The shared packet ring (SPR) interface used in RPR (SPR1) is a valid interface-name for the ML-Series card. For more information on the SPR interface, see Chapter 17, “Configuring Resilient Packet Ring.” The input-interface choice is not valid when applied to the INPUT of an interface (redundant).
Router(config-cmap)# match ip dscp <i>ip-dscp-value</i>	Specifies up to eight DSCP values used as match criteria. The value of each service code point is from 0 to 63.
Router (config-cmap)# match ip precedence <i>ip-precedence-value</i>	Specifies up to eight IP precedence values used as match criteria.

Creating a Traffic Policy

To configure a traffic policy, use the **policy-map** global configuration command to specify the traffic policy name, and use the following configuration commands to associate a traffic class, which was configured with the **class-map** command and one or more QoS features. The traffic class is associated with the traffic policy when the **class** command is used. The **class** command must be issued after entering policy-map configuration mode. After entering the **class** command, you are automatically in policy-map class configuration mode, which is where the QoS policies for the traffic policy are defined.

When the bandwidth or priority action is used on any class in a policy map, then there must be a class defined by the **match-any** command, which has a bandwidth or priority action in that policy map. This is to ensure that all traffic can be classified into a default class which has some assigned bandwidth. A minimum bandwidth can be assigned if the class is not expected to be used or no reserved bandwidth is desired for default traffic.

The QoS policies that can be applied in the traffic policy in policy-map class configuration mode are detailed in the following example:

The syntax of the **policy-map** command is:

```
policy-map policy-name
no policy-map policy-name
```

The syntax of the **class** command is:

```
class class-map-name
no class class-map-name
```

All traffic that fails to meet the matching criteria belongs to the default traffic class. The default traffic class can be configured by the user, but cannot be deleted.

To create a traffic policy, use the commands in [Table 14-2](#) as needed:

Table 14-2 Traffic Policy Commands

Command	Purpose
Router (config)# policy-map <i>policy-name</i>	Specifies the name of the traffic policy to configure. Names can be a maximum of 40 alphanumeric characters.
Router (config-pmap)# class <i>class-map-name</i>	Specifies the name of a predefined traffic class, which was configured with the class-map command, used to classify traffic to the traffic policy.
Router (config-pmap)# class class-default	Specifies the default class to be created as part of the traffic policy.

Table 14-2 Traffic Policy Commands (continued)

Command	Purpose
<pre>Router (config-pmap-c)# bandwidth {<i>bandwidth-kbps</i> percent <i>percent</i>}</pre>	<p>Specifies a minimum bandwidth guarantee to a traffic class in periods of congestion. A minimum bandwidth guarantee can be specified in kbps or by a percentage of the overall available bandwidth.</p> <p>Valid choices for the ML-Series cards are:</p> <ul style="list-style-type: none"> • Rate in kilobits per second • Percent of total available bandwidth (1 to 100) <p>If multiple classes and bandwidth actions are specified in a single policy map, they must use the same choice in specifying bandwidth (kilobits or percent).</p> <p>Note When using the bandwidth command, excess traffic (beyond the configured commit) is allocated any available bandwidth in proportion to the relative bandwidth commitment of its traffic class compared to other traffic classes. Excess traffic from two classes with equal commits has equal access to available bandwidth. Excess traffic from a class with a minimum commit might receive only a minimum share of available bandwidth compared to excess bandwidth from a class with a high commit.</p> <p>Note The true configureable bandwidth in kilobits or megabits per second is per port and depends on how the ML-Series card is configured. The show interface command shows the maximum bandwidth of a port (example <code>BW 100000 Kbit</code>). The sum of all bandwidth and priority actions applied to the interface, plus the cos priority-mcast bandwidth, is not allowed to exceed the maximum bandwidth of the port.</p>

Table 14-2 Traffic Policy Commands (continued)

Command	Purpose
<pre>Router (config-pmap-c)# police <i>cir-rate-bps</i> <i>normal-burst-byte</i> [<i>max-burst-byte</i>] [pir <i>pir-rate-bps</i>] [conform-action {set-cos-transmit transmit drop}] [exceed-action {set-cos-transmit drop}] [violate-action {set-cos-transmit drop}]</pre>	<p>Defines a policer for the currently selected class when the policy map is applied to input. Policing is supported only on ingress, not on egress.</p> <ul style="list-style-type: none"> • For <i>cir-rate-bps</i>, specify the average committed information rate (cir) in bits per second (bps). The range is 96000 to 800000000. • For <i>normal-burst-byte</i>, specify the cir burst size in bytes. The range is 8000 to 64000. • (Optional) For <i>maximum-burst-byte</i>, specify the peak information rate (pir) burst in bytes. The range is 8000 to 64000. • (Optional) For <i>pir-rate-bps</i>, specify the average pir traffic rate in bps where the range is 96000 to 800000000. • (Optional) Conform action options are: <ul style="list-style-type: none"> – Set a CoS priority value and transmit – Transmit packet (default) – Drop packet • (Optional) Exceed action options are: <ul style="list-style-type: none"> – Set a CoS value and transmit – Drop packet (default) • (Optional) The violate action is only valid if pir is configured. Violate action options are: <ul style="list-style-type: none"> – Set a CoS value and transmit – Drop packet (default)

Table 14-2 Traffic Policy Commands (continued)

Command	Purpose
<pre>Router (config-pmap-c)# priority kbps</pre>	<p>Specifies low latency queuing for the currently selected class. This command can only be applied to an output. When the policy-map is applied to an output, an output queue with strict priority is created for this class. The only valid rate choice is in kilobits per second.</p> <p>Note This priority command does not apply to the default class.</p> <p>Note When using the priority action, the traffic in that class is given a 100 percent CIR, regardless of the rate entered as the priority rate. To ensure that other bandwidth commitments are met for the interface, a policer must be configured on the input of all interfaces that might deliver traffic to this output class, limiting the peak rate to the priority rate entered.</p> <p>Note The true configureable bandwidth in kilobits or megabits per second is per port and depends on how the ML-Series card is configured. The show interface command shows the maximum bandwidth of a port (example <code>BW 100000 Kbit</code>). The sum of all bandwidth and priority actions applied to the interface, plus the cos priority-mcast bandwidth, is not allowed to exceed the maximum bandwidth of the port.</p>
<pre>Router (config-pmap-c)# set cos cos-value</pre>	<p>Specifies a CoS value or values to associate with the packet. The number is in the range from 0 to 7.</p> <p>This command can only be used in a policy-map applied to an output. It specifies the VLAN CoS priority to set for the outbound packets in the currently selected class. If QinQ is used, the top-level VLAN tag is marked. If outbound packets have no VLAN tag, the action has no effect. This action is applied to the packet after any set-cos action done by a policer, and therefore overrides the CoS set by a policer action.</p> <p>If a packet is marked by the policer and forwarded out an interface that also has a set-cos action assigned for the traffic class, the value specified by the police action takes precedence in setting the IEEE 802.1p CoS field.</p> <p>This command also sets the CoS value in the RPR header for packets exiting the ML-Series on the RPR interface.</p>

Attaching a Traffic Policy to an Interface

Use the **service-policy** interface configuration command to attach a traffic policy to an interface and to specify the direction in which the policy should be applied (either on packets coming into the interface or packets leaving the interface). Only one traffic policy can be applied to an interface in a given direction.

Use the **no** form of the command to detach a traffic policy from an interface. The **service-policy** command syntax is as follows:

```
service-policy {input | output} policy-map-name
no service-policy {input | output} policy-map-name
```

To attach a traffic policy to an interface, use the following commands in global configuration mode, as needed:

Step 1	Router(config)# interface <i>interface-id</i>	Enters interface configuration mode, and specifies the interface to apply the policy map. Valid interfaces are limited to physical Ethernet and POS interfaces. Note Policy maps cannot be applied to SPR interfaces, subinterfaces, port channel interfaces, or Bridge Group Virtual Interfaces (BVIS).
Step 2	Router(config-if)# service-policy output <i>policy-map-name</i>	Specifies the name of the traffic policy to be attached to the output direction of an interface. The traffic policy evaluates all traffic leaving that interface.
Step 3	Router(config-if)# service-policy input <i>policy-map-name</i>	Specifies the name of the traffic policy to be attached to the input direction of an interface. The traffic policy evaluates all traffic entering that interface.

Configuring CoS-Based QoS

The global **cos commit** *cos-value* command allows the ML-Series card to base the QoS treatment for a packet coming in on a network interface on the attached CoS value, rather than on a per-customer-queue policer.

CoS-based QoS is applied with a single global **cos commit** *cos-value* command, as shown in [Table 14-3](#):

Table 14-3 CoS Commit Command

Command	Purpose
Router(config)# cos-commit <i>cos-value</i>	Labels packets that come in with a CoS equal to or higher than the <i>cos-value</i> as CIR and packets with a lower CoS as DE.

Monitoring and Verifying QoS Configuration

After configuring QoS on the ML-Series card, the configuration of class maps and policy maps can be viewed through a variety of **show** commands. To display the information relating to a traffic class or traffic policy, use one of the following commands in EXEC mode, as needed. [Table 14-4](#) describes the commands that are related to QoS status.

Table 14-4 Commands for QoS Status

Command	Purpose
Router# show class-map <i>name</i>	Displays the traffic class information of the user-specified traffic class.
Router# show policy-map	Displays all configured traffic policies.
Router# show policy-map <i>name</i>	Displays the user-specified policy map.
Router# show policy-map interface <i>interface</i>	Displays configurations of all input and output policies attached to an interface. Statistics displayed with this command are unsupported and show zero.

[Example 14-1](#) show examples of the QoS commands.

Example 14-1 QoS Status Command Examples

```

Router# show class-map
Class Map match-any class-default (id 0)
  Match any
Class Map match-all policer (id 2)
  Match ip precedence 0

Router# show policy-map
Policy Map police_f0
  class policer
    police 1000000 10000 conform-action transmit exceed-action drop

Router# show policy-map interface

FastEthernet0

  service-policy input: police_f0

  class-map: policer (match-all)
    0 packets, 0 bytes
    5 minute offered rate 0 bps, drop rate 0 bps
    match: ip precedence 0

  class-map: class-default (match-any)
    0 packets, 0 bytes
    5 minute offered rate 0 bps, drop rate 0 bps
    match: any
      0 packets, 0 bytes
      5 minute rate 0 bps

```

QoS Configuration Examples

This section provides the specific command and network configuration examples:

- [Traffic Classes Defined Example](#)
- [Traffic Policy Created Example](#)
- [class-map match-any and class-map match-all Commands Example](#)
- [match spr1 Interface Example](#)

- [ML-Series VoIP Example](#)
- [ML-Series Policing Example](#)
- [ML-Series CoS-Based QoS Example](#)

Traffic Classes Defined Example

[Example 14-2](#) shows how to create a class map called class1 that matches incoming traffic entering interface fastethernet0.

Example 14-2 Class Interface Command Examples

```
Router(config)# class-map class1
Router(config-cmap)# match input-interface fastethernet0
```

[Example 14-3](#) shows how to create a class map called class2 that matches incoming traffic with IP-precedence values of 5, 6, and 7.

Example 14-3 Class IP-Precedence Command Examples

```
Router(config)# class-map match-any class2
Router(config-cmap)# match ip precedence 5 6 7
```



Note

If a class-map contains a match rule that specifies multiple values, such as 5 6 7 in this example, then the class-map must be match-any, not the default match-all. Without the match-any class-map, an error message is printed and the class is ignored. The supported commands that allow multiple values are **match cos**, **match ip precedence**, and **match ip dscp**.

[Example 14-4](#) shows how to create a class map called class3 that matches incoming traffic based on bridge group 1.

Example 14-4 Class Map Bridge Group Command Examples

```
Router(config)# class-map class3
Router(config-cmap)# match bridge-group 1
```

Traffic Policy Created Example

In [Example 14-5](#), a traffic policy called policy1 is defined to contain policy specifications, including a bandwidth allocation request for the default class and two additional classes—class1 and class2. The match criteria for these classes were defined in the traffic classes, see the [“Creating a Traffic Class” section on page 14-10](#).

Example 14-5 Traffic Policy Created Example

```
Router(config)# policy-map policy1
Router(config-pmap)# class class-default
Router(config-pmap-c)# bandwidth 1000
Router(config-pmap)# exit
```

```
Router(config-pmap)# class class1
Router(config-pmap-c)# bandwidth 3000
Router(config-pmap)# exit

Router(config-pmap)# class class2
Router(config-pmap-c)# bandwidth 2000
Router(config-pmap)# exit
```

class-map match-any and class-map match-all Commands Example

This section illustrates the difference between the **class-map match-any** command and the **class-map match-all** command. The **match-any** and **match-all** options determine how packets are evaluated when multiple match criteria exist. packets must either meet all of the match criteria (**match-all**) or one of the match criteria (**match-any**) in order to be considered a member of the traffic class.

[Example 14-6](#) shows a traffic class configured with the **class-map match-all** command.

Example 14-6 Class Map Match All Command Examples

```
Router(config)# class-map match-all cisco1
Router(config-cmap)# match cos 1
Router(config-cmap)# match bridge-group 10
```

If a packet arrives with a traffic class called `cisco1` configured on the interface, the packet is evaluated to determine if it matches the `cos 1` and `bridge group 10`. If both of these match criteria are met, the packet matches traffic class `cisco1`.

In traffic class called `cisco2`, the match criteria are evaluated consecutively until a successful match criterion is located. The packet is first evaluated to the determine whether `cos 1` can be used as a match criterion. If `cos 1` can be used as a match criterion, the packet is matched to traffic class `cisco2`. If `cos 1` is not a successful match criterion, then `bridge-group 10` is evaluated as a match criterion. Each matching criterion is evaluated to see if the packet matches that criterion. When a successful match occurs, the packet is classified as a member of traffic class `cisco2`. If the packet matches none of the specified criteria, the packet is classified as a member of the traffic class.

Note that the **class-map match-all** command requires that all of the match criteria must be met in order for the packet to be considered a member of the specified traffic class (a logical AND operator). In the example, `cos 1 AND bridge group 10` have to be successful match criteria. However, only one match criterion must be met for the packet in the **class-map match-any** command to be classified as a member of the traffic class (a logical OR operator). In the example, `cos 1 OR bridge group 10 OR ip dscp 5` have to be successful match criteria.

[Example 14-7](#) shows a traffic class configured with the **class-map match-any** command.

Example 14-7 Class Map Match Any Command Examples

```
Router(config)# class-map match-any cisco2
Router(config-cmap)# match cos 1
Router(config-cmap)# match bridge-group 10
Router(config-cmap)# match ip dscp 5
```

match spr1 Interface Example

In [Example 14-8](#), the SPR interface is specified as a parameter to the **match input-interface** CLI when defining a class-map.

Example 14-8 Class Map SPR Interface Command Examples

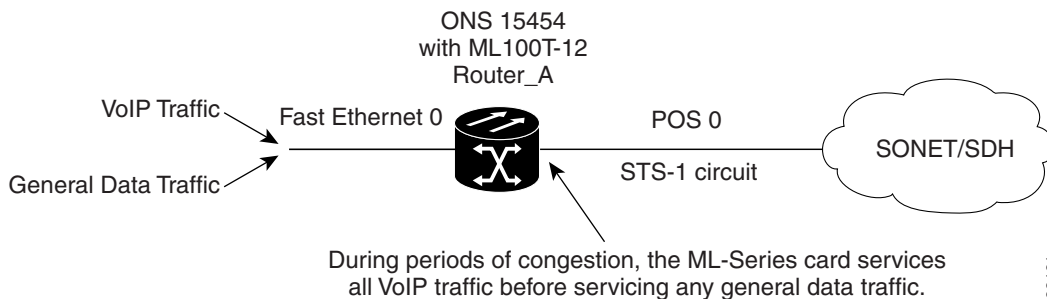
```

Router(config)# class-map spr1-cos1
Router(config-cmap)# match input-interface spr1
Router(config-cmap)# match cos 1
Router(config-cmap)# end
Router# sh class-map spr1-cos1
Class Map match-all spr1-cos1 (id 3)
  Match input-interface SPR1
  Match cos 1

```

ML-Series VoIP Example

Figure 14-7 shows an example of ML-Series QoS. The associated commands are provided in Example 14-9.

Figure 14-7 ML-Series VoIP Example

83181

Example 14-9 ML-Series VoIP Commands

```

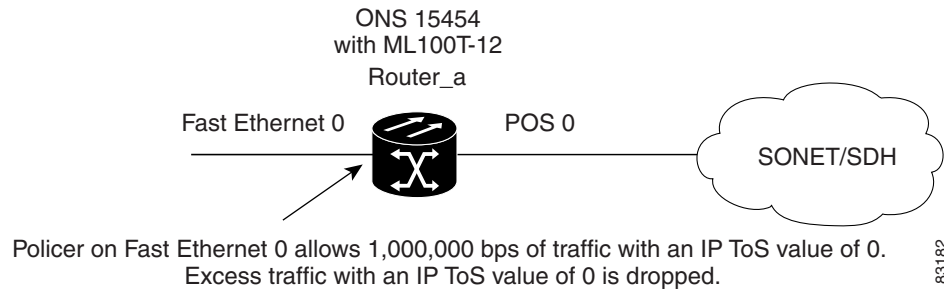
Router(config)# class-map match-all voip
Router(config-cmap)# match ip precedence 5
Router(config-cmap)# exit
Router(config)# class-map match-any default
Router(config-cmap)# match any
Router(config-cmap)# exit
Router(config)# policy-map pos0
Router(config-pmap)# class default
Router(config-pmap-c)# bandwidth 1000
Router(config-pmap-c)# class voip
Router(config-pmap-c)# priority 1000
Router(config-pmap-c)# interface FastEthernet0
Router(config-if)# ip address 1.1.1.1 255.255.255.0
Router(config-if)# interface POS0
Router(config-if)# ip address 2.1.1.1 255.255.255.0
Router(config-if)# service-policy output pos0
Router(config-if)# crc 32
Router(config-if)# no cdp enable
Router(config-if)# pos flag c2 1

```

ML-Series Policing Example

Figure 14-8 shows an example of ML-Series policing. The example shows how to configure a policer that restricts traffic with an IP precedence of 0 to 1,000,000 bps. The associated code is provided in Example 14-10.

Figure 14-8 ML-Series Policing Example



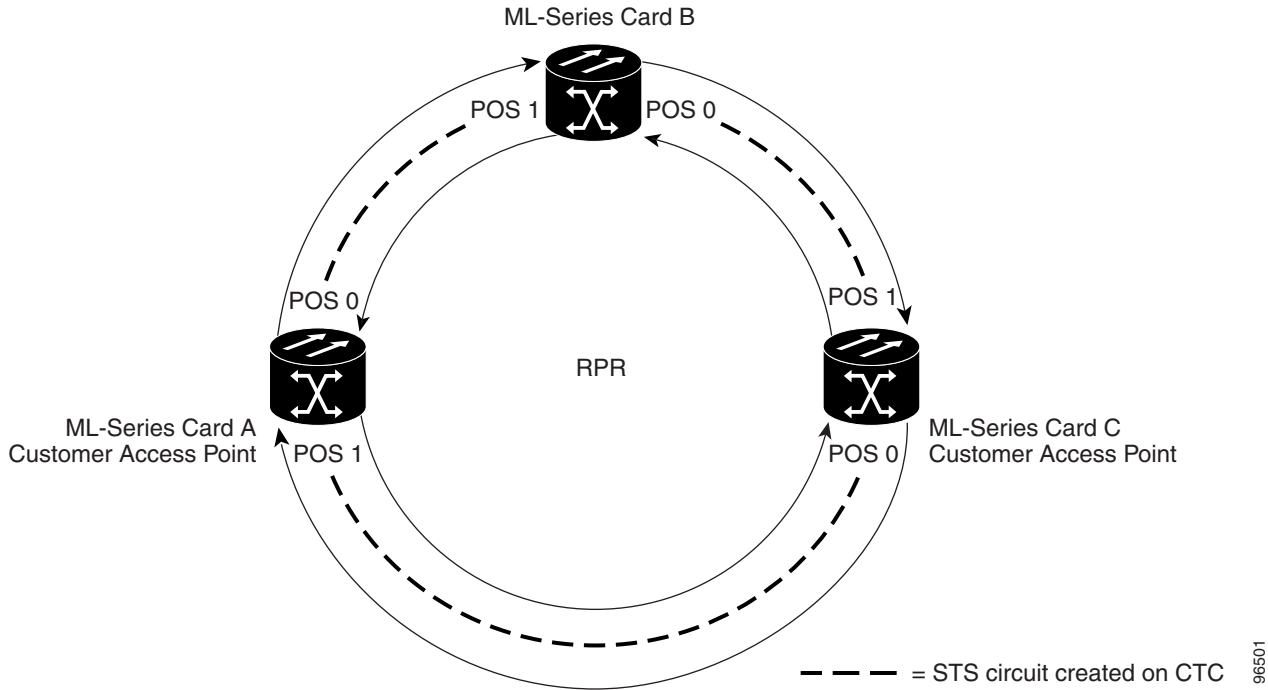
Example 14-10 ML-Series Policing Commands

```
Router(config)# class-map match-all policer
Router(config-cmap)# match ip precedence 0
Router(config-cmap)# exit
Router(config)# policy-map police_f0
Router(config-pmap)# class policer
Router(config-pmap-c)# police 1000000 10000 conform-action transmit exceed-action drop
Router(config-pmap-c)# interface FastEthernet0
Router(config-if)# service-policy input police_f0
```

ML-Series CoS-Based QoS Example

Figure 14-9 shows an example of ML-Series CoS-based QoS. The associated code is provided in the examples that follow the figure. The CoS example assumes that the ML-Series cards are configured into an RPR and that the ML-Series card POS ports are linked by point-to-point SONET circuits. ML-Series Card A and ML-Series Card C are customer access points. ML-Series Card B is not a customer access point. For more information on configuring RPR, see Chapter 17, “Configuring Resilient Packet Ring.”

Figure 14-9 ML-Series CoS Example



Example 14-11 shows the code used to configure ML-Series card A in Figure 14-9.

Example 14-11 ML-Series Card A Configuration (Customer Access Point)

```
ML_Series_A(config)# cos commit 2
ML_Series_A(config)# policy-map Fast5_in
ML_Series_A(config-pmap)# class class-default
ML_Series_A(config-pmap-c)# police 5000 8000 8000 pir 10000 conform-action
set-cos-transmit 2 exceed-action set-cos-transmit 1 violate-action drop
```

Example 14-12 shows the code used to configure ML-Series card B in Figure 14-9.

Example 14-12 ML-Series Card B Configuration (Not Customer Access Point)

```
ML_Series_B(config)# cos commit 2
```

Example 14-13 shows the code used to configure ML-Series card C in Figure 14-9.

Example 14-13 ML-Series Card C Configuration (Customer Access Point)

```
ML_Series_B(config)# cos commit 2
ML_Series_B(config)# policy-map Fast5_in
ML_Series_B(config-pmap)# class class-default
ML_Series_B(config-pmap-c)# police 5000 8000 8000 pir 10000 conform-action
set-cos-transmit 2 exceed-action set-cos-transmit 1 violate-action drop
```


Understanding Multicast QoS and Multicast Priority Queuing

ML-Series card QoS supports the creation of two priority classes for multicast traffic in addition to the default multiclass traffic class. Creating a multicast priority queuing class of traffic configures the ML-Series card to recognize an existing CoS value in ingress multicast traffic for priority treatment.

The multicast priority queuing CoS match is based on the “internal” CoS value of each packet. This value is normally the same as the egress CoS value (after policer marking if enabled) but differs in two cases. The “internal” CoS value is not the same as the egress value when dot1q-tunneling is used. Under dot1q-tunnel, the internal CoS value is always the value of the outer tag CoS, both when entering the dot1q tunnel and leaving the dot1q tunnel. The “internal” CoS value is also not the same as the egress value if a packet is transported over a vlan, and the vlan tag is removed on egress to send the packet untagged. In this case the internal CoS is the CoS of the removed tag (including ingress policing and marking if enabled).

The **cos priority-mcast** command does not modify the CoS of the multicast packets but only the bandwidth allocation for the multicast priority queuing class. The command guarantees a minimum amount of bandwidth and is queued separately from the default multicast/broadcast queue.

Creating a multicast priority queuing class allows for special handling of certain types of multiclass traffic. This is especially valuable for multicast video distribution and service provider multicast traffic. For example, a service provider may want to guarantee the protection of their own multicast management traffic. To do this, they could create a multicast priority queuing class on the ML-Series card for the CoS value of the multicast management traffic and guarantee its minimum bandwidth. For multicast video distribution, a multicast priority queuing class on the ML-Series card for the CoS value of the multicast video traffic enables networks to efficiently manage multicast video bandwidth demands on a network shared with VoIP and other Ethernet services.

**Note**

Multicast priority queuing traffic uses port-based load-balancing over RPR and EtherChannel. Default multicast traffic is load-balanced over RPR, but not over EtherChannel. On the ML1000-2, multicast load balancing maps GigabitEthernet Port 0 to POS Port 0 and GigabitEthernet Port 1 to POS Port 1. On the ML100T-12, multicast load balancing maps Fast Ethernet Port 0 and all even-numbered Fast Ethernet ports to POS 0 and all odd-numbered Fast Ethernet ports to POS 1.

**Note**

Multicast priority queuing bandwidth should not be oversubscribed for sustained periods with traffic from multiple sources. This can result in reduced multicast priority queuing throughput.

Default Multicast QoS

Default multicast traffic is any multicast traffic (including flooded traffic) that is not classified as multicast priority queuing. The default multicast class also includes broadcast data traffic, control traffic, L2 protocol tunneling, and flooding traffic of the unknown MAC during MAC learning.

With no QoS configured (no multicast priority queuing and no output policy map) on the ML-Series card, the default multicast bandwidth is a 10 percent minimum of total bandwidth.

When bandwidth is allocated to multicast priority queuing but no output policy map is applied, the default multicast congestion bandwidth is a minimum of 10 percent of the bandwidth not allocated to multicast priority queuing.

When an output policy-map is applied to an interface, default multicast and default unicast share the minimum bandwidth assigned to the default class. This default class is also known as the match-any class. The minimum bandwidth of default multicast is 10 percent of the total default class bandwidth.

Multicast Priority Queuing QoS Restrictions

The following restrictions apply to multicast priority queuing QoS:

- The bandwidth allocation and utilization configured for multicast priority queuing traffic is global and applies to all the ports on the ML-Series card, both POS and Fast Ethernet or Gigabit Ethernet, regardless of whether these ports carry multicast priority queuing traffic. The rate of traffic can be reduced for all ports on the ML-Series card when this feature is configured. Default multicast traffic uses bandwidth only on the ports where it egresses, not globally like multicast priority queuing.
- Multicast priority queuing QoS is supported only for Layer 2 bridging.
- The ML-Series card supports a maximum of two multicast priority queuing classes.
- Unlike the rest of the ML-Series card QoS, multicast priority queuing QoS is not part of the Cisco IOS Modular QoS CLI (MQC).
- Priority-mcast bandwidth allocation is per port and the maximum bandwidth configurable on an ML1000-2 with **cos priority-mcast** is 1000 mbps. But the load-balancing of multicast priority queuing increases the effective bandwidth. For example, with an ML1000-2 with GEC and STS-24c RPR circuits, the user can allocate 1000 mbps per port, but will be able to get 2000 mbps total effective bandwidth due to the load-balancing.

Configuring Multicast Priority Queuing QoS

To configure a priority class for multicast traffic, use the global configuration **cos priority-mcast** command defined in [Table 14-5](#).

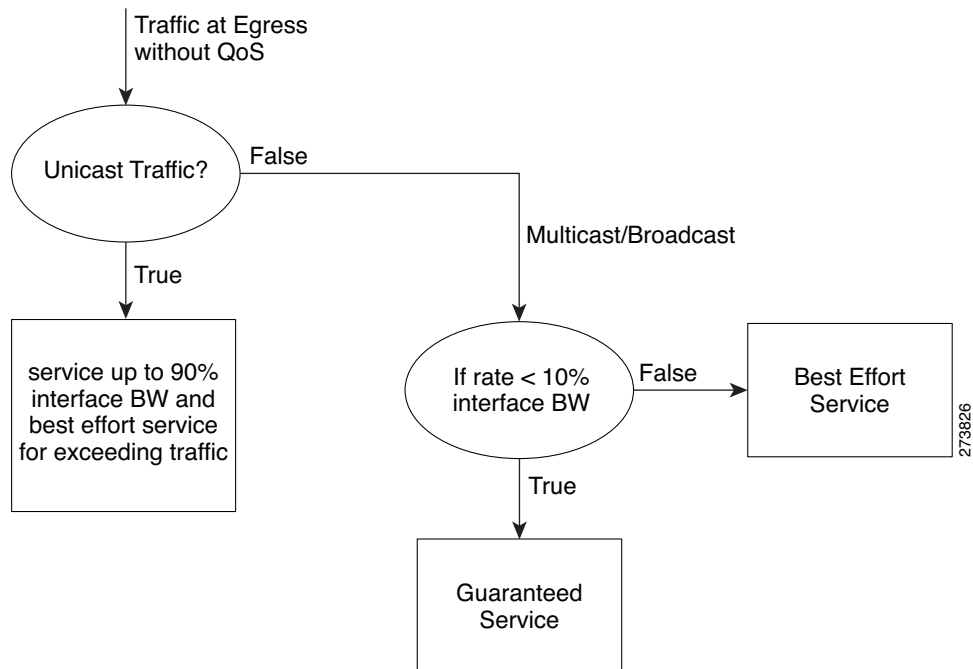
Table 14-5 CoS Multicast Priority Queuing Command

Command	Purpose
<pre>Router (config)# [no] cos priority-mcast <i>cos-value</i> {<i>bandwidth-kbps</i> mbps <i>bandwidth-mbps</i> percent <i>percent</i>}</pre>	<p><i>Creates a priority class of multicast traffic based on a multicast CoS value and specifies a minimum bandwidth guarantee to a traffic class in periods of congestion.</i></p> <p><i>cos-value</i> specifies the CoS value of multicast packets which will be given the bandwidth allocation. Matches only a single CoS of traffic (not a range). Supported CoS range is 0 to 7.</p> <p>A minimum bandwidth guarantee can be specified in kbps, in mbps, or by a percentage of the overall available bandwidth.</p> <p>Valid choices for the ML-Series card are:</p> <ul style="list-style-type: none"> • Rate in kilobits per second • Rate in megabits per second • Percent of total available port bandwidth (1 to 100) <p>Reentering the command with the same <i>cos-value</i> but a different bandwidth rate will modify the bandwidth of the existing class.</p> <p>Reentering the command with a different <i>cos-value</i> creates a separate multicast priority queuing class with a maximum of two multicast priority queuing classes.</p> <p>The no form of this command removes the multicast priority queuing class.</p> <p>Note The true configureable bandwidth in kilobits or megabits per second is per port and depends on how the ML-Series card is configured. The show interface command shows the maximum bandwidth of a port (example <code>BW 100000 Kbit</code>). The sum of all bandwidth and priority actions applied to the interface, plus the cos priority-mcast bandwidth, is not allowed to exceed the maximum bandwidth of the port.</p> <p>Attempting to configure a priority-mcast bandwidth which exceeds the true configureable bandwidth on any port will cause the priority-mcast configuration change to fail, and the multicast priority queuing bandwidth guarantee will not be changed.</p>

QoS not Configured on Egress

The QoS bandwidth allocation of multicast and broadcast traffic is handled separately from unicast traffic. On each interface, the aggregate multicast and broadcast traffic are given a fixed bandwidth commit of 10% of the interface bandwidth. This is the optimum bandwidth that can be provided for traffic exceeding 10% of the interface bandwidth.

Figure 14-10 QoS not Configured on Egress



ML-Series Egress Bandwidth Example

This section explains with examples the utilization of bandwidth across different queues with or without Priority Multicast.

Case 1: QoS with Priority and Bandwidth Configured Without Priority Multicast

Strict Priority Queue is always serviced first. The remaining interface bandwidth is utilized to service other configured traffic.

In the following example, after servicing unicast `customer_voice` traffic, the remaining interface bandwidth is utilized for other WRR queues such as `customer_core_traffic`, `customer_data`, and `class-default` in the ratio of 1:3:5.

At any given time, the sum of the bandwidth assigned cannot exceed the interface bandwidth (in kbps). The bandwidth share allocated to `class-default` will be utilized by default unicast traffic (in this example, unicast traffic with CoS values other than 2, 5, 7) and all multicast/broadcast traffic (all CoS values). The default unicast and all multicast/broadcast traffic will be serviced in the ratio of 9:1.

For example, if 18x bandwidth is available after servicing priority unicast traffic (CoS 5), then the remaining bandwidth will be allocated as follows:

Unicast traffic with CoS 2 : 2x
 Unicast traffic with CoS 7: 6x
 Unicast default (without CoS 2, CoS 5, CoS 7): 9x
 All multicast/broadcast (any CoS value): 1x

Example 14-14 QoS with Priority and Bandwidth Configured without Priority Multicast

```

!
class-map match-all customer_voice
  match cos 5
class-map match-all customer_data
  match cos 7
class-map match-all customer_core_traffic
  match cos 2
!
!
policy-map policy_egress_bandwidth
  class customer_core_traffic
    bandwidth 1000
  class customer_voice
    priority 1000
  class customer_data
    bandwidth 3000
  class class-default
    bandwidth 5000
!
!
interface POS0
  no ip address
  crc 32
  service-policy output policy_egress_bandwidth
!

```

Case 2: QoS with Priority and Bandwidth Configured with Priority Multicast

In this case, only multicast traffic of CoS 3 is allocated a guaranteed bandwidth. This multicast traffic will now participate in the queue along with other WRR queues. After servicing the `customer_voice` traffic, the remaining interface bandwidth is utilized for WRR queues, such as `customer_core_traffic`, `customer_data`, `class-default`, and multicast CoS 3 traffic in the ratio of 1:3:5:2.

At any given time, the sum of the bandwidth assigned cannot exceed the interface bandwidth (in kbps).

Example 14-15 QoS with Priority and Bandwidth configured with Priority Multicast

```

cos priority-mcast 3 2000
!
class-map match-all customer_voice
  match cos 5
class-map match-all customer_data
  match cos 7
class-map match-all customer_core_traffic
  match cos 2
!
!
policy-map policy_egress_bandwidth
  class customer_core_traffic
    bandwidth 1000

```

```

class customer_voice
  priority 1000
class customer_data
  bandwidth 3000
class class-default
  bandwidth 5000
!
!
interface POS0
  no ip address
  crc 32
  service-policy output policy_egress_bandwidth
!

```

Understanding CoS-Based Packet Statistics

Enhanced performance monitoring displays per-CoS packet statistics on the ML-Series card interfaces when CoS accounting is enabled. Per-CoS packet statistics are only supported for bridged services, not IP routing or Multiprotocol Label Switching (MPLS). CoS-based traffic utilization is displayed at the Fast Ethernet or Gigabit Ethernet interface or subinterface (VLAN) level, or the POS interface level. It is not displayed at the POS subinterface level. RPR statistics are not available at the SPR interface level, but statistics are available for the individual POS ports that make up the SPR interface. EtherChannel (port-channel) and BVI statistics are available only at the member port level. [Table 14-6](#) shows the types of statistics available at specific interfaces.

Table 14-6 Packet Statistics on ML-Series Card Interfaces

Statistics Collected	Gigabit/Fast Ethernet Interface	Gigabit/Fast Ethernet Subinterface (VLAN)	POS Interface	POS Subinterface
Input—Packets and Bytes	Yes	Yes	No	No
Output—Packets and Bytes	Yes	Yes	No	No
Drop Count—Packets and Bytes ¹	Yes	No	Yes	No

1. Drop counts only include discards caused by output congestion and are counted at the output interface.

CoS-based packet statistics are available through the Cisco IOS command-line interface (CLI) and Simple Network Management Protocol (SNMP), using an extension of the CISCO-PORT-QOS MIB. They are not available through Cisco Transport Controller (CTC).

Configuring CoS-Based Packet Statistics



Note

CoS-based packet statistics require the enhanced microcode image to be loaded onto the ML-Series card.

For information on the enhanced microcode image, see the [“Multiple Microcode Images”](#) section on [page 3-11](#).

To enable CoS-based packet statistics on an interface, use the interface configuration level command defined in [Table 14-7](#).

Table 14-7 CoS-Based Packet Statistics Command

Command	Purpose
Router(config-if)# cos accounting	Enables CoS-based packet statistics to be recorded at the specific interface and for all the subinterfaces of that interface. This command is supported only in interface configuration mode and not subinterface configuration mode. The no form of the command disables the statistics.

After configuring CoS-based packet statistics on the ML-Series card, the statistics can be viewed through a variety of **show** commands. To display this information, use one of the commands in [Table 14-8](#) in EXEC mode.

Table 14-8 Commands for CoS-Based Packet Statistics

Command	Purpose
Router# show interface <i>type number cos</i>	Displays the CoS-based packet statistics available for an interface.
Router# show interface <i>type number.subinterface-number cos</i>	Displays the CoS-based packet statistics available for a FastEthernet or Gigabit Ethernet subinterface. POS subinterfaces are not eligible.

[Example 14-16](#) shows examples of these commands.

Example 14-16 Commands for CoS-Based Packet Statistics Examples

```
Router# show interface gigabitethernet 0.5 cos
GigabitEthernet0.5
  Stats by Internal-Cos
  Input: Packets      Bytes
    Cos 0: 31         2000
    Cos 1:
    Cos 2: 5          400
    Cos 3:
    Cos 4:
    Cos 5:
    Cos 6:
    Cos 7:
  Output: Packets     Bytes
    Cos 0: 1234567890 1234567890
    Cos 1: 31          2000
    Cos 2:
    Cos 3:
    Cos 4:
    Cos 5:
    Cos 6: 10          640
    Cos 7:

Router# show interface gigabitethernet 0 cos
GigabitEthernet0
  Stats by Internal-Cos
  Input: Packets      Bytes
    Cos 0: 123         3564
    Cos 1:
    Cos 2: 3           211
```

```

Cos 3:
Cos 4:
Cos 5:
Cos 6:
Cos 7:
Output: Packets      Bytes
Cos 0: 1234567890  1234567890
Cos 1: 3            200
Cos 2:
Cos 3:
Cos 4:
Cos 5:
Cos 6: 1            64
Cos 7:
Output: Drop-pkts   Drop-bytes
Cos 0: 1234567890  1234567890
Cos 1:
Cos 2:
Cos 3:
Cos 4:
Cos 5: 1            64
Cos 6: 10           640
Cos 7:

```

```

Router# show interface pos0 cos
POS0
Stats by Internal-Cos
Output: Drop-pkts   Drop-bytes
Cos 0: 12           1234
Cos 1: 31           2000
Cos 2:
Cos 3:
Cos 4:
Cos 5:
Cos 6: 10           640
Cos 7:

```




Configuring the Switching Database Manager

This chapter describes the switching database manager (SDM) features built into the ML100T-12 and ML1000-2 cards.

This chapter contains the following major sections:

- [Understanding the SDM, page 15-1](#)
- [SDM Regions, page 15-1](#)
- [Configuring SDM, page 15-2](#)

Understanding the SDM

ML-Series cards use the forwarding engine and ternary content-addressable memory (TCAM) to implement high-speed forwarding. The high-speed forwarding information is maintained in TCAM. The SDM is the software subsystem that manages the switching information maintained in TCAM.

SDM organizes the switching information in TCAM into application-specific regions and configures the size of these application regions. SDM enables exact-match and longest-match address searches, which result in high-speed forwarding. SDM manages TCAM space by partitioning application-specific switching information into multiple regions.

TCAM identifies a location index associated with each packet forwarded and conveys it to the forwarding engine. The forwarding engine uses this location index to derive information associated with each forwarded packet.

The key benefits of SDM in switching are its ability to organize the switching information in TCAM into application-specific regions and its ability to configure the size of these application regions. SDM enables exact-match and longest-match address searches, which result in high-speed forwarding.

SDM Regions

SDM partitions TCAM space into multiple application-specific regions and interacts with the individual application control layers to store switching information. SDM consists of the following types of regions:

- **Exact-match region**—The exact-match region consists of entries for multiple application regions such as IP adjacencies.

- Longest-match region—Each longest-match region consists of multiple buckets or groups of Layer 3 address entries organized in decreasing order by mask length. All entries within a bucket share the same mask value and key size. The buckets can change their size dynamically by borrowing address entries from neighboring buckets. Although the size of the whole application region is fixed, you can reconfigure it.
- Weighted-exact-match region—The weighted-exact-match region consists of exact-match-entries with an assigned weight or priority. For example, with quality of service (QoS), multiple exact match entries might exist, but some have priority over others. The weight is used to select one entry when multiple entries match.

TCAM space consists of 65,536 entries, each entry being 64 bits wide. Because SDM is responsible for managing TCAM space, SDM partitions the entire TCAM space for each application region based on user configuration. Although the maximum size of all application regions is fixed, you can reconfigure the maximum size of each application region.

Table 15-1 lists default partitioning for each application region in TCAM.

Table 15-1 Default Partitioning by Application Region in TCAM

Application Region	Lookup Type	Key Size	Default Size	No. of TCAM Entries
IP Adjacency	Exact-match	64 bits	65536 (shared)	65536 (shared)
IP Prefix	Longest-match	64 bits	65536 (shared)	65536 (shared)
QoS Classifiers	Weighted exact-match	64 bits	65536 (shared)	65536 (shared)
IP VRF Prefix	Longest prefix match	64 bits	65536 (shared)	65536 (shared)
IP Multicast	Longest prefix match	64 bits	65536 (shared)	65536 (shared)
MAC Addr	Longest prefix match	64 bits	65536 (shared)	65536 (shared)
Access List	Weighted exact match	64 bits	65536 (shared)	65536 (shared)

Configuring SDM

This section describes the commands necessary to configure the SDM. It includes commands to configure the size of the SDM regions. The commands described in this section are unique to the switching software.

Configuring SDM Regions

TCAM space consists of 65,536 entries, each entry being 64 bits wide. Since SDM is responsible for managing TCAM space, SDM partitions the entire TCAM space for each application region based on user configuration. A change in the partition configuration takes effect the next time you reboot the system.

The application region size in SDM is represented by the number of 64-bit entries. The combined size of all the application regions should be calculated in terms of 64-bit TCAM entries and should not exceed 65,536 bytes, which is the total TCAM size.

To configure SDM maximum size for each application region, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# sdm size <i>region-name</i> [k-entries] <i>number-of-entries</i>	Sets the name of the application region whose size you want to configure. You can enter the size as multiples of 1K (that is, 1024) entries or in absolute number of entries.
Step 2	Router(config)# end	Exits to privileged EXEC mode.

An example of this is shown in [Example 15-1](#).

Example 15-1 Limiting the IP-Prefix Region to 2K Entries

```
Router # configure terminal
Router(config)# sdm size ip-prefix k-entries 2
Router(config)# end
```

To display the number of available TCAM entries, enter the **show sdm size** command from global configuration mode:

```
Router # show sdm size
Active Switching Database Region Maximum Sizes :
  IP Adjacency       : 65536 64-bit entries
  IP Prefix          : 204864-bit entries
  QoS Classifiers    : 65536 64-bit entries
  IP VRF Prefix      : 65536 64-bit entries
  IP Multicast       : 65536 64-bit entries
  MAC Addr           : 65536 64-bit entries
  Access List        : 6553664-bit entries
```

Configuring Access Control List Size in TCAM

The default maximum size of the access control list (ACL) is 65,536 64-bit entries. You can enter the **sdm access-list** command to limit the TCAM space for ACLs, as shown in [Table 15-2](#).

Table 15-2 Partitioning the TCAM Size for ACLs

Task	Command
sdm access-list <i>number-entries</i>	Sets the name of the application region for which you want to configure the size. You can enter the size as an absolute number of entries.

An example of this is shown in [Example 15-2](#).

Example 15-2 Configuring 8,192 Entries for the ACL Region in TCAM

```
Router# configure terminal
Router(config)# sdm access-list 8192
Router(config)# end
```




Configuring Access Control Lists

This chapter describes the access control list (ACL) features built into the ML-Series card.

This chapter contains the following major sections:

- [Understanding ACLs, page 16-1](#)
- [ML-Series ACL Support, page 16-1](#)
- [Modifying ACL TCAM Size, page 16-5](#)

Understanding ACLs

ACLs provide network control and security, allowing you to filter packet flow into or out of ML-Series interfaces. ACLs, which are sometimes called filters, allow you to restrict network use by certain users or devices. ACLs are created for each protocol and are applied on the interface for either inbound or outbound traffic. ACLs do not apply to outbound control plane traffic. Only one ACL filter can be applied per direction per subinterface.

When creating ACLs, you define criteria to apply to each packet processed by the ML-Series card; the ML-Series card decides whether to forward or block the packet based on whether or not the packet matches the criteria in your list. Packets that do not match any criteria in your list are automatically blocked by the implicit “deny all traffic” criteria statement at the end of every ACL.

ML-Series ACL Support

Both control-plane and data-plane ACLs are supported on the ML-Series card:

- **Control-plane ACLs:** ACLs used to filter control data that is processed by the CPU of the ML-Series card (for example, distribution of routing information, Internet Group Membership Protocol (IGMP) joins, and so on).
- **Data-plane ACLs:** ACLs used to filter user data being routed or bridged through the ML Series in hardware (for example, denying access to a host, and so on). These ACLs are applied to an interface in the input or output direction using the **ip access-group** command.

The following apply when using data-plane ACLs on the ML-Series card:

- ACLs are supported on all interface types, including bridged interfaces.
- Reflexive and dynamic ACLs are not supported on the ML-Series card.
- Access violations accounting is not supported on the ML-Series card.

- ACL logging is supported only for packets going to the CPU, not for switched packets.
- IP standard ACLs applied to bridged egress interfaces are not supported in the data-plane. When bridging, ACLs are only supported on ingress.

IP ACLs

The following ACL styles for IP are supported:

- Standard IP ACLs: These use source addresses for matching operations.
- Extended IP ACLs (control plane only): These use source and destination addresses for matching operations and optional protocol type and port numbers for finer granularity of control.
- Named ACLs: These use source addresses for matching operations.



Note

By default, the end of the ACL contains an implicit deny statement for everything if it did not find a match before reaching the end. With standard ACLs, if you omit the mask from an associated IP host address ACL specification, 0.0.0.0 is assumed to be the mask.

After creating an ACL, you must apply it to an interface, as shown in the [“Applying the ACL to an Interface” section on page 16-4](#).

Named IP ACLs

You can identify IP ACLs with a name, but it must be an alphanumeric string. Named IP ACLs allow you to configure more IP ACLs in a router than if you used numbered ACLs. If you identify your ACL with an alphabetic rather than a numeric string, the mode and command syntax are slightly different.

Consider the following before configuring named ACLs:

- A standard ACL and an extended ACL cannot have the same name.
- Numbered ACLs are also available, as described in the [“Creating Numbered Standard and Extended IP ACLs” section on page 16-3](#).

User Guidelines

Keep the following in mind when you configure IP network access control:

- You can program ACL entries into Ternary Content Addressable Memory (TCAM).
- You do not have to enter a deny everything statement at the end of your ACL; it is implicit.
- You can enter ACL entries in any order without any performance impact.
- For every eight TCAM entries, the ML-Series card uses one entry for TCAM management purposes.
- Do not set up conditions that result in packets getting lost. This situation can happen when a device or interface is configured to advertise services on a network that has ACLs that deny these packets.
- IP ACLs are not supported for double-tagged (QinQ) packets. They will however be applied to IP packets entering on a QinQ access port.

Creating IP ACLs

The following sections describe how to create numbered standard, extended, and named standard IP ACLs:

- [Creating Numbered Standard and Extended IP ACLs, page 16-3](#)
- [Creating Named Standard IP ACLs, page 16-4](#)
- [Creating Named Extended IP ACLs \(Control Plane Only\), page 16-4](#)
- [Applying the ACL to an Interface, page 16-4](#)

Creating Numbered Standard and Extended IP ACLs

Table 16-1 lists the global configuration commands used to create numbered standard and extended IP ACLs.

Table 16-1 Commands for Numbered Standard and Extended IP ACLs

Command	Purpose
Router(config)# access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>]	Defines a standard IP ACL using a source address and wildcard.
Router(config)# access-list <i>access-list-number</i> {deny permit} any	Defines a standard IP ACL using an abbreviation for the source and source mask of 0.0.0.0 255.255.255.255.
Router(config)# access-list <i>extended-access-list-number</i> {deny permit} <i>protocol source source-wildcard destination</i> <i>destination-wildcard</i> [precedence <i>precedence</i>] [tos tos]	Defines an extended IP ACL number and the access conditions.
Router(config)# access-list <i>extended-access-list-number</i> {deny permit} <i>protocol any any</i>	Defines an extended IP ACL using an abbreviation for a source and source wildcard of 0.0.0.0 255.255.255.255, and an abbreviation for a destination and destination wildcard of 0.0.0.0 255.255.255.255.
Router(config)# access-list <i>extended-access-list-number</i> {deny permit} <i>protocol host source host destination</i>	Defines an extended IP ACL using an abbreviation for a source and source wildcard of source 0.0.0.0, and an abbreviation for a destination and destination wildcard of destination 0.0.0.0.

Creating Named Standard IP ACLs

To create a named standard IP ACL, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# ip access-list standard <i>name</i>	Defines a standard IP ACL using an alphabetic name.
Step 2	Router(config-std-nacl)# deny { <i>source</i> [<i>source-wildcard</i>] any } or permit { <i>source</i> [<i>source-wildcard</i>] any }	In access-list configuration mode, specifies one or more conditions as permitted or denied. This determines whether the packet is passed or dropped.
Step 3	Router(config)# exit	Exits access-list configuration mode.

Creating Named Extended IP ACLs (Control Plane Only)

To create a named extended IP ACL, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# ip access-list extended <i>name</i>	Defines an extended IP ACL using an alphabetic name.
Step 2	Router(config-ext-nacl)# { deny permit } <i>protocol source source-wildcard destination destination-wildcard</i> [precedence <i>precedence</i>] [tos <i>tos</i>] or {deny permit} protocol any any or {deny permit} protocol host source host destination	In access-list configuration mode, specifies the conditions allowed or denied. Or: Defines an extended IP ACL using an abbreviation for a source and source wildcard of 0.0.0.0 255.255.255.255, and an abbreviation for a destination and destination wildcard of 0.0.0.0 255.255.255.255. Or: Defines an extended IP ACL using an abbreviation for a source and source wildcard of <i>source</i> 0.0.0.0, and an abbreviation for a destination and destination wildcard of <i>destination</i> 0.0.0.0.

Applying the ACL to an Interface

After you create an ACL or extended ACL, you can apply it to one or more interfaces. ACLs can be applied on either the inbound or the outbound direction of an interface. When controlling access to an interface, you can use a name or number. If a standard ACL is applied, the ML-Series card compares the source IP address with the ACL. To apply an ACL to one or more interfaces, use the command in [Table 16-2](#).

**Note**

IP standard ACLs applied to the ingress of a Bridge Group Virtual Interface (BVI) will be applied to all bridged IP traffic in the associated bridge-group, in addition to the BVI ingress traffic.

Table 16-2 Applying ACL to Interface

Command	Purpose
<code>ip access-group {access-list-number name} {in out}</code>	Controls access to an interface.

Modifying ACL TCAM Size

You can change the TCAM size by entering the `sdm access-list` command. For more information on ACL TCAM sizes, see the “[Configuring Access Control List Size in TCAM](#)” section on page 15-3.

[Example 16-1](#) provides an example of modifying and verifying ACLs.

**Note**

To increase the ACL TCAM size, you must decrease another region’s TCAM size, such as IP, IP multicast, or L2 switching.

**Caution**

You will need to increase the TCAM size if you see the following error message:

```
Warning:Programming TCAM entries failed
Please remove last ACL command to re-activate ACL operation.
!<ACL number or name> <IP or IPX> <INPUT_ACL or OUTPUT_ACL> from TCAM group for !<interface>
Please see the documentation to see if TCAM space can be
increased on this platform to alleviate the problem.
```

Example 16-1 Monitor and Verify ACLs

```
Router# show ip access-lists 1
Standard IP access list 1
  permit 192.168.1.1
  permit 192.168.1.2
```




Configuring Resilient Packet Ring



Note

The terms “Unidirectional Path Switched Ring” and “UPSR” may appear in Cisco literature. These terms do not refer to using Cisco ONS 15xxx products in a unidirectional path switched ring configuration. Rather, these terms, as well as “Path Protected Mesh Network” and “PPMN,” refer generally to Cisco’s path protection feature, which may be used in any topological network configuration. Cisco does not recommend using its path protection feature in any particular topological network configuration.

This chapter describes how to configure resilient packet ring (RPR), RPR Link Fault Propagation, and Dual RPR Interconnect (DRPRI) for the ML-Series card.

This chapter contains the following major sections:

- [Understanding RPR, page 17-1](#)
- [Configuring RPR, page 17-7](#)
- [Monitoring and Verifying RPR, page 17-17](#)
- [Add an ML-Series Card into an RPR, page 17-18](#)
- [Delete an ML-Series Card from an RPR, page 17-22](#)
- [Understanding RPR Link Fault Propagation, page 17-27](#)
- [Configuring LFP, page 17-29](#)
- [RPR Keep Alive, page 17-31](#)
- [RPR Shortest Path, page 17-33](#)
- [Understanding Dual RPR Interconnect, page 17-36](#)
- [Configuring DRPRI, page 17-37](#)

Understanding RPR

RPR is a new MAC protocol operating at the Layer 2 level. It is well suited for transporting Ethernet over a SONET/SDH ring topology and enables multiple ML-Series cards to become one functional network segment or shared packet ring (SPR). RPR overcomes the limitations of earlier schemes, such as IEEE 802.1D Spanning Tree Protocol (STP), IEEE 802.1W Rapid Spanning Tree Protocol (RSTP), and SONET/SDH, when used in this role. Although the IEEE 802.17 draft was used as reference for the Cisco ML-Series RPR implementation, the current ML-Series card RPR protocol does not comply with all clauses of IEEE 802.17.

Role of SONET/SDH Circuits

The ML-Series cards in an SPR must connect directly or indirectly through point-to-point STS/STM circuits. The point-to-point STS/STM circuits are configured on the ONS node and are transported over the ONS node's SONET/SDH topology with either protected or unprotected circuits.

On circuits unprotected by the SONET/SDH mechanism, RPR provides resiliency without using the capacity of the redundant protection path that a SONET/SDH protected circuit would require. This frees this capacity for additional traffic. RPR also utilizes the bandwidth of the entire ring and does not block segments like STP or RSTP.

Packet Handling Operations

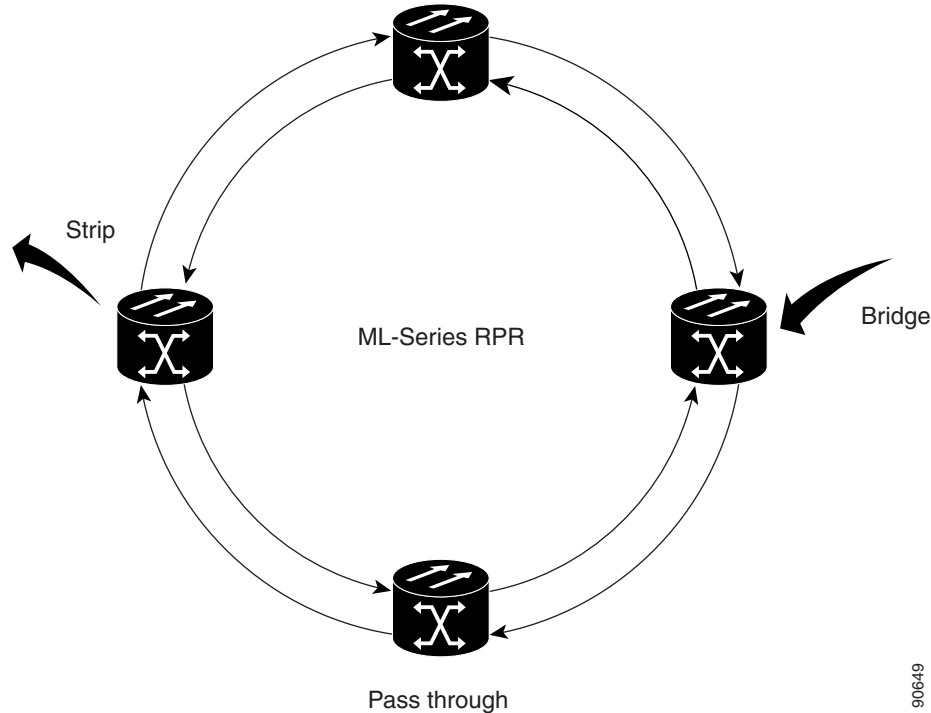
When an ML-Series card is configured with RPR and is made part of an SPR, the ML-Series card assumes a ring topology. If a packet is not destined for network devices bridged through the Ethernet ports of a specific ML-Series card, the ML-Series card simply continues to forward this transit traffic along the SONET/SDH circuit, relying on the circular path of the ring architecture to guarantee that the packet will eventually arrive at the destination. This eliminates the need to queue and process the packet flowing through the nondestination ML-Series card. From a Layer 2 or Layer 3 perspective, the entire RPR looks like one shared network segment.

An ML-Series card configured with RPR has three basic packet-handling operations: bridge, pass-through, and strip. [Figure 17-1](#) illustrates these operations. Bridging connects and passes packets between the Ethernet ports on the ML-Series and the packet-over-SONET/SDH (POS) ports used for the SONET/SDH circuit circling the ring. Pass-through lets the packets continue through the ML-Series card and along the ring, and stripping takes the packet off the ring and discards it.

The RPR protocol, using the transmitted packet's header information, allows the interfaces to quickly determine the operation that needs to be applied to the packet. It also uses both the source and destination addresses of a packet to choose a ring direction. Flow-based load sharing helps ensure that all packets populated with equal source-and destination-address pairs will be sent in the same direction, and arrive at their destination in the correct order. Ring direction also enables the use of spatial reuse to increase overall ring aggregate bandwidth. Unicast packets are destination stripped. Destination stripping provides the ability to have simultaneous flows of traffic between different parts of an RPR. Traffic can be concurrently transmitted bidirectionally between adjacent nodes. It can also span multiple nodes, effectively reusing the same ring bandwidth. Multicast packets are source stripped.

The ML-Series card supports three load-balancing schemes for unicast packets. The port-based load balancing option maps even ports to the POS 0 interface and odd ports to the POS 1 interface. The default auto option balances the load based on the MAC addresses or source and destination addresses of the IP packet. The ML-Series card also supports an RPR shortest path option. This option is covered in detail in the [“RPR Shortest Path”](#) section on page 17-33.

Figure 17-1 RPR Packet Handling Operations



906906

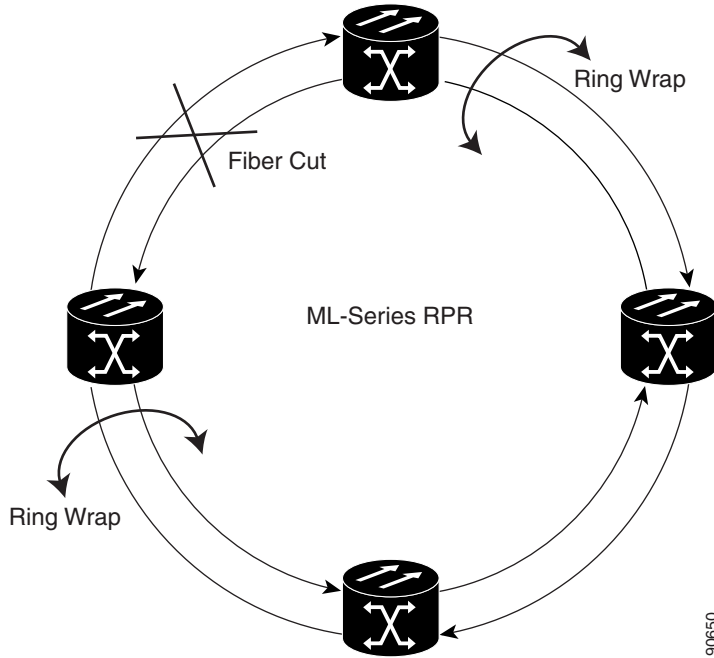
Ring Wrapping

RPR initiates ring wraps in the event of a fiber cut, node failure, node restoration, new node insertion, or other traffic problem. This protection mechanism redirects traffic to the original destination by sending it in the opposite direction around the ring after a link state change or after receiving SONET/SDH path level alarms. Ring wrapping on the ML-Series card allows convergence times of less than 50 ms for unicast and pass-through traffic. RPR convergence times are comparable to SONET/SDH and much faster than STP or RSTP.

RPR on the ML-Series card survives both unidirectional and bidirectional transmission failures within the ring. Unlike STP or RSTP, RPR restoration is scalable. Increasing the number of ML-Series cards in a ring does not increase the convergence time.

Ring wraps occur within 50 msec after the failure condition with the default **spr wrap immediate** configured. If **spr wrap delay** is configured, the wrap is delayed until the POS interface goes link-down. The link goes down after the time specified with the CLI **pos trigger delay <msec>**. If the circuits are VCAT then the Cisco IOS CLI command **pos vcat defect delayed** also needs to be configured. The delay helps ensure that when RPR is configured with SONET/SDH bandwidth protection, this Layer 1 protection has a chance to take effect before the Layer 2 RPR protection. If the interface goes down without a SONET error, then the carrier delay also take effect. [Figure 17-2](#) illustrates ring wrapping.

Figure 17-2 RPR Ring Wrapping



In case of a ring failure, the ML-Series cards connected to the failed section of the RPR detect the failure through the SONET/SDH path alarms. When any ML-Series card receives this path-AIS signal, it wraps the POS interface that received the signal.

**Note**

ML-Series card RPR convergence times might exceed 50 ms in the case of multiple failures in the same ring, if traffic passes through an ML-Series card configured with DRPRI (in active mode) during the reloading of the ML-Series card, or in the case of mismatched microcode images on ML-Series cards.

**Note**

If the carrier delay time is changed from the default, the new carrier delay time must be configured on all the ML-Series card interfaces, including the SPR, POS, and Gigabit Ethernet or Fast Ethernet interfaces.

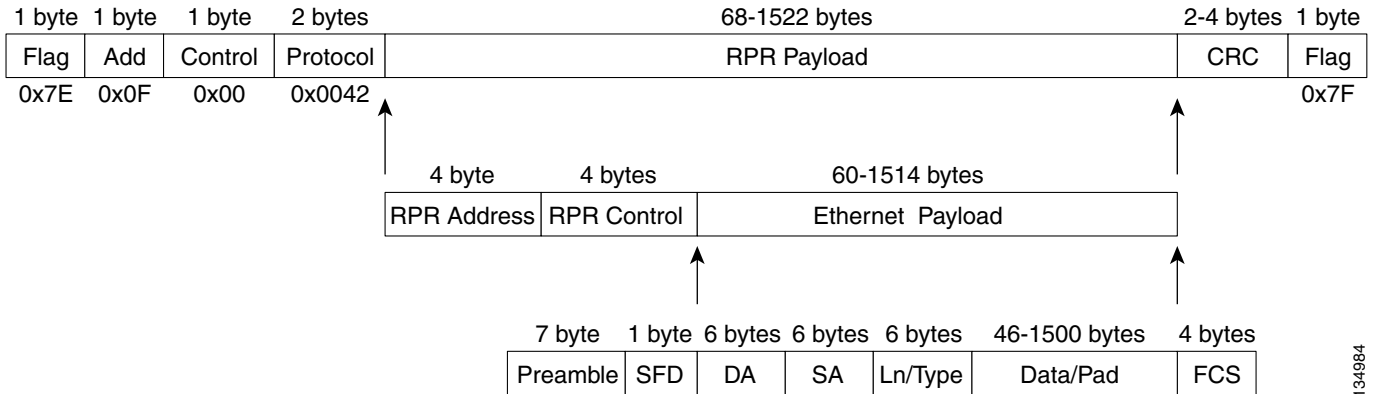
**Note**

ML-Series card POS interfaces normally send an alarm for signal label mismatch failure in the ONS 15454 STS path overhead (PDI-P) to the far end when the POS link goes down or when RPR wraps. ML-Series card POS interfaces do not send PDI-P to the far-end when PDI-P is detected, when a remote deflection indication alarm (RDI-P) is being sent to the far end, or when the only defects detected are generic framing procedure (GFP)-loss of frame delineation (LFD), GFP client signal fail (CSF), virtual concatenation (VCAT)-loss of multiframe (LOM), or VCAT-loss of sequence (SQM).

RPR Framing Process

The ML-Series card uses a proprietary RPR frame and HDLC or GFP-F framing. It attaches the RPR frame header to each Ethernet frame and encapsulates the RPR frame into the SONET/SDH payload for transport over the SONET/SDH topology. The RPR header is removed at the egress ML-Series card. [Figure 17-3](#) illustrates the RPR frame.

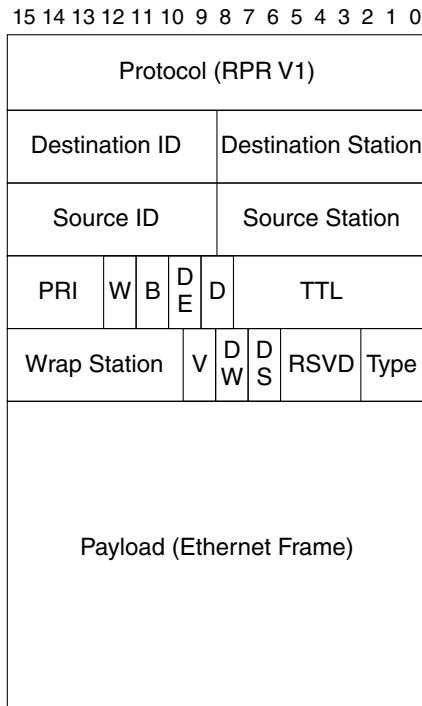
Figure 17-3 RPR Frame for ML-Series Card



134984

The RPR framing and header includes a number of fields, including four bytes for source and destination station information and another four bytes for RPR control and quality of service (QoS). [Figure 17-4](#) illustrates the RPR frame format. [Table 17-1](#) defines the most important fields.

Figure 17-4 RPR Frame Fields



134982

Table 17-1 Definitions of RPR Frame Fields

Destination Station	An eight-bit field specifying the MAC address of a specific ML-Series card in the RPR as the destination. It has two well-known addresses, 0xff for Multicast DA-MAC and 0x00 for Unknown DA-MAC.
Source Station	An eight-bit field specifying the MAC address of a specific ML-Series card in the RPR as the source.
PRI	A three-bit QoS class of service (CoS) field that establishes RPR priority.
DE	A one-bit field for the discard eligible flag.
TTL	A nine-bit field for the frame's time to live.
Type	A field indicating whether the packet is data or control.

MAC Address and VLAN Support

RPR increases the total number of MAC addresses supported because the MAC IDs of packets that pass through an ML-Series card are not recorded by that ML-Series card. The ML-Series card only records the MAC IDs of the packets that are bridged or stripped by that ML-Series card. This allows a greater number of MAC addresses in the collective address tables of the RPR.

VLANs on RPR require less interface configuration than VLANs on STP and RSTP, which require configuration on all the POS interfaces in the ring. RPR VLANs only require configuration on SPR interfaces that bridge or strip packets for that VLAN.

The ML-Series card still has an architectural maximum limit of 255 VLAN/bridge-group per ML-Series card. But because the ML-Series card only needs to maintain the MAC address of directly connected devices, a greater total number of connected devices are allowed on an RPR network.

RPR QoS

The ML-Series card's RPR relies on the QoS features of the ML-Series card for efficient bandwidth utilization with service level agreement (SLA) support. ML-Series card QoS mechanisms apply to all SONET/SDH traffic on the ML-Series card, whether passed-through, bridged, or stripped. For detailed RPR QoS information see the QoS on RPR section of [Chapter 14, "Configuring Quality of Service."](#)

CTM and RPR

The Cisco Transport Manager (CTM) is an element management system (EMS) designed to integrate into an overall network management system (NMS) and interface with other higher level management tools. CTM supports RPR provisioning on ML-Series cards. For more information, refer to the *Cisco Transport Manager User Guide* at:

http://www.cisco.com/en/US/products/sw/opticsw/ps2204/products_user_guide_list.html

Configuring RPR

You need to use both CTC and Cisco IOS to configure RPR for the ML-Series card. CTC is the graphical user interface (GUI) that serves as the enhanced craft tool for specific ONS node operations, including the provisioning of the point-to-point SONET/SDH circuits required for RPR. Cisco IOS is used to configure RPR on the ML-Series card and its interfaces.

Successfully creating an RPR requires several consecutive procedures:

1. [Connecting the ML-Series Cards with Point-to-Point STS/STM Circuits, page 17-7](#) (CTC or TL1)
2. [Configuring CTC Circuits for RPR, page 17-7](#) (CTC or TL1)
3. [Configuring RPR Characteristics and the SPR Interface on the ML-Series Card, page 17-11](#) (Cisco IOS)
4. [Assigning the ML-Series Card POS Ports to the SPR Interface, page 17-13](#) (Cisco IOS)
5. [Creating the Bridge Group and Assigning the Ethernet and SPR Interfaces, page 17-14](#) (Cisco IOS)
6. [Verifying Ethernet Connectivity Between RPR Ethernet Access Ports, page 17-17](#) (Cisco IOS)
7. [CRC threshold configuration and detection, page 17-17](#)

**Note**

Transaction Language One (TL1) can be used to provision the required SONET/SDH point-to-point circuits instead of CTC.

Connecting the ML-Series Cards with Point-to-Point STS/STM Circuits

You connect the ML-Series cards in an RPR through point-to-point STS/STM circuits. These circuits use the ONS 15454 SONET/SDH network and are provisioned using CTC in the normal manner for provisioning optical circuits.

Configuring CTC Circuits for RPR

These are the guidelines for configuring the CTC circuits required by RPR:

- Leave all CTC Circuit Creation Wizard options at their default settings, except **Fully Protected Path** in the Circuit Routing Preferences dialog box. **Fully Protected Path** provides SONET/SDH protection and should be unchecked. RPR normally provides the Layer 2 protection for SPR circuits.
- Check **Using Required Nodes and Spans** to route automatically in the Circuit Routing Preferences dialog box. If the source and destination nodes are adjacent on the ring, exclude all nodes except the source and destination in the Circuit Routing Preferences dialog box. This forces the circuit to be routed directly between source and destination and preserves STS/STM circuits, which would be consumed if the circuit routed through other nodes in the ring. If there is a node or nodes that do not contain an ML-Series card between the two nodes containing ML-Series cards, include this node or nodes in the included nodes area in the Circuit Routing Preference dialog box, along with the source and destination nodes.
- Keep in mind that ML-Series card STS/STM circuits do not support unrelated circuit creation options, such as the following check box titles in CTC, unidirectional traffic, creating cross-connects only (TL1-like), interdomain (unified control plane [UCP]), protected drops, subnetwork connection protection (SCNP), or path protection selectors.

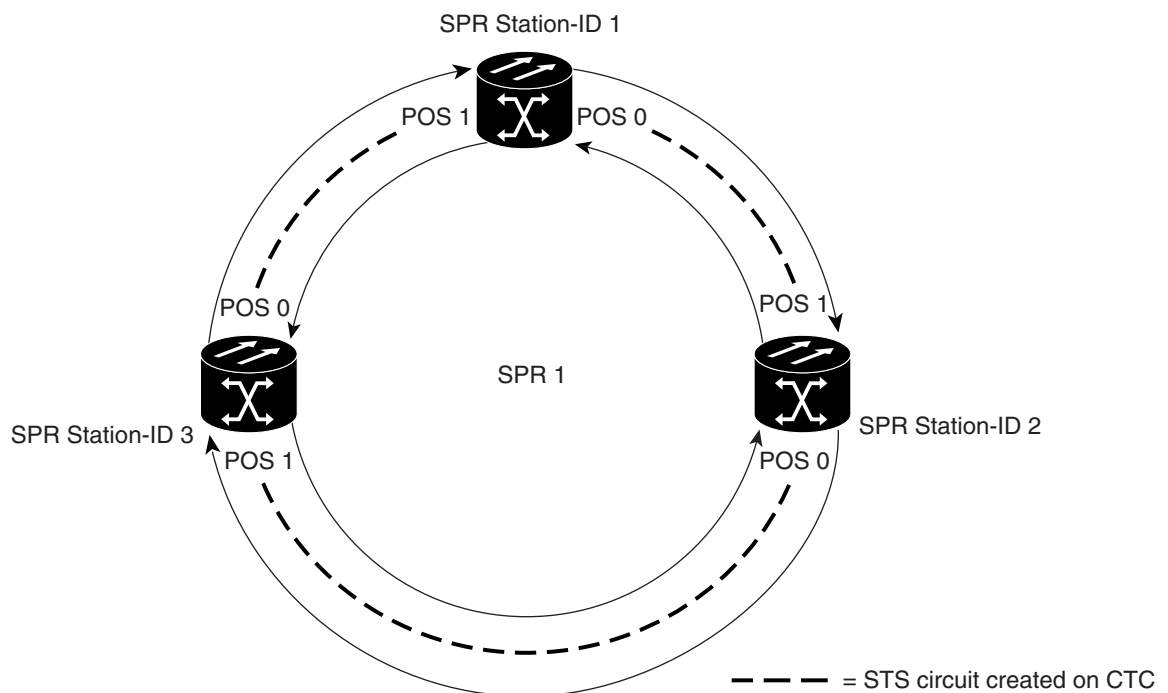
- A best practice is to configure SONET/SDH circuits in an east-to-west or west-to-east configuration, from Port 0 (east) to Port 1 (west) or Port 1 (east) to Port 0 (west), around the SONET/SDH ring. Do not configure Port 0 to Port 0 or Port 1 to Port 1. The east-to-west or west-to-east setup is also required in order for the CTM network management software to recognize the ML-Series configuration as an SPR.

Detailed CTC circuit procedures are available in the “Create Circuits and VT Tunnels” chapter of the *Cisco ONS 15454 Procedure Guide* and the “Create Circuits and Tunnels” chapter of the *Cisco ONS 15454 SDH Procedure Guide*.

CTC Circuit Configuration Example for RPR

Figure 17-5 illustrates an example of a three-node RPR.

Figure 17-5 Three Node RPR Example



The three-node RPR in Figure 17-5 is used for all of the examples in the consecutive RPR procedures. Combining the examples will give you an end-to-end example of creating an RPR. It is assumed that the SONET/SDH node and its network is already active.



Caution

The specific steps in the following procedure are for the topology shown in the example. Your own specific steps will vary according to your network. Do not attempt this procedure without obtaining a detailed plan or method of procedure from an experienced network architect.

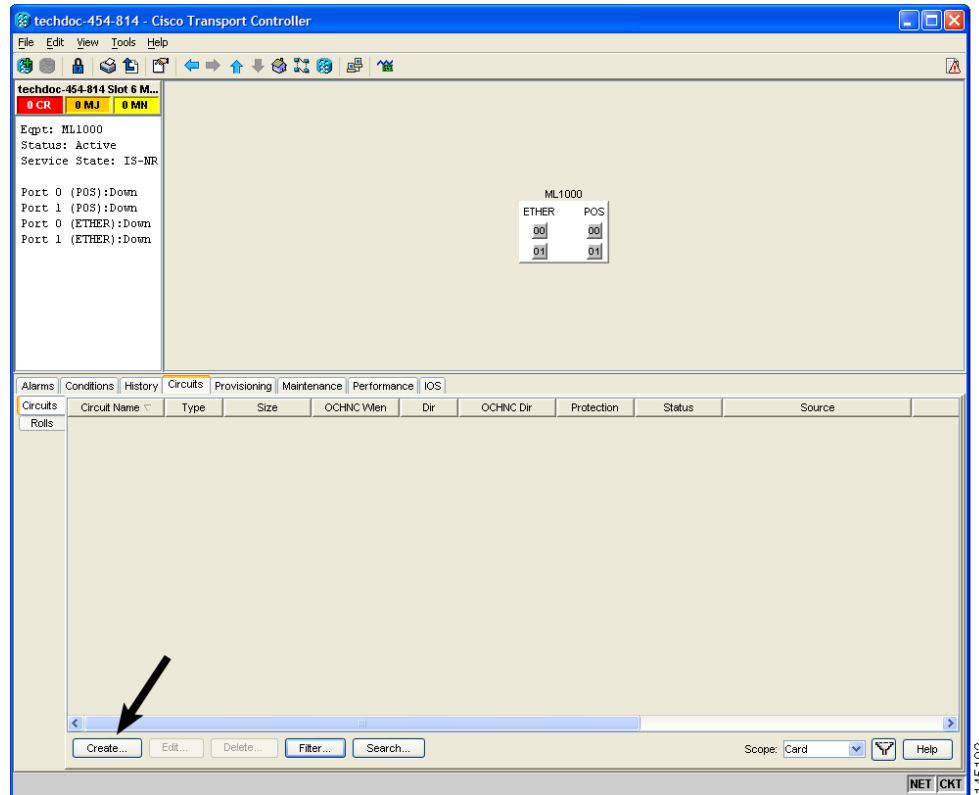
To configure the circuits, you need to create three circuits in CTC:

- Create a circuit from Node 1, POS Port 0 to Node 2, POS Port 1.
- Create a circuit from Node 2, POS Port 0 to Node 3, POS Port 1.

- Create a circuit from Node 3, POS Port 0 to Node 1, POS Port 1.

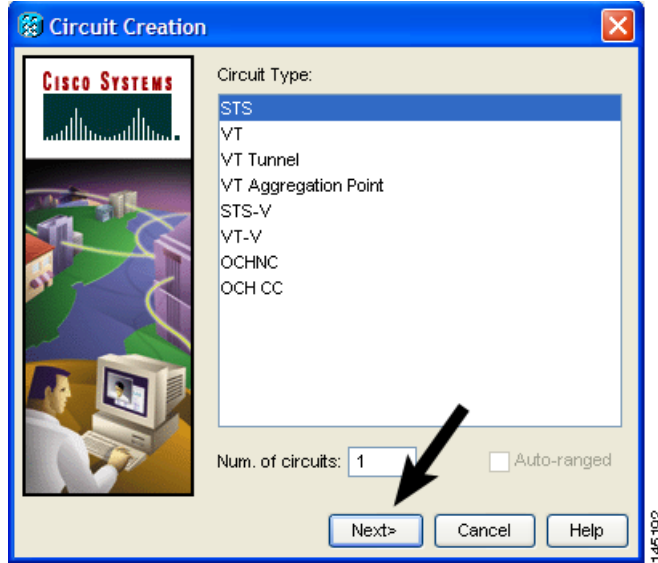
Step 1 In CTC, log into Node 1 and navigate to the CTC card view for the ML-Series card that will be in the RPR.

Figure 17-6 CTC Card View for ML-Series Card



Step 2 Click the **Circuits > Create** tabs.
The first page of the Circuit Creation wizard appears.

Figure 17-7 CTC Circuit Creation Wizard



Step 3 In the Circuit Type list, select **STS**.

Step 4 Click **Next**.

The Circuit Attributes page appears.

Step 5 Type a circuit name in the Name field.

Step 6 Select the relevant size of the circuit from the Size drop-down list, and the appropriate state from the State list.

Step 7 Verify the SD threshold is set to 1E-6 (default) or in the 1E-6 to 1E-9 range in the SD threshold field.

- a. If the SD threshold is the default 1E-6 or within the acceptable range, proceed to [Step 8](#).
- b. If the SD threshold is not the default 1E-6 or within the acceptable range, select 1E-6 or a threshold within the acceptable range from the menu.



Note

Lower SD thresholds increase the speed of CTC convergence, but also increase the possibility of interface flapping (repeatedly enabling and disabling) in some situations.

Step 8 Click **Next**.

The Source page appears.

Step 9 Select Node 1 as the source node from the node drop-down list.

Step 10 Select the ML-Series card from the Slot drop-down list, and choose 0 (POS) from the Port drop-down list.

Step 11 Click **Next**.

The Destination page appears.

Step 12 Select Node 2 as the destination node from the Node drop-down list.

Step 13 Select the ML-Series card from the Slot drop-down list, and choose 1 (POS) from the Port drop-down list.

Step 14 Click **Next**.

The Circuit Routing Preferences page appears.

Step 15 Uncheck the Fully Protected Path check box.

Step 16 Click **Next**.

The Circuit Constraints for Automatic Routing page appears.

Step 17 Click the Node 1 icon to select it and click **Next**.

The Route Review/Edit page appears.

Step 18 Click **Finish**.

You have now completed the initial circuit for the RPR.



Note

A TPTFAIL alarm might appear on CTC when the circuit is created. This alarm will disappear after the POS ports are enabled during the [“Assigning the ML-Series Card POS Ports to the SPR Interface” procedure on page 17-13](#).

Step 19 Build the second circuit between POS 0 on Node 2 and POS 1 on Node 3. Use the same procedure described in Steps 1 through 18, but substitute Node 2 for Node 1 and Node 3 for Node 2.

Step 20 Build the third circuit between POS 0 on Node 3 and POS 1 on Node 1. Use the same procedure described in Steps 1 through 18, but substitute Node 3 for Node 1 and Node 1 for Node 2.

Now all of the POS ports in all three nodes are connected by STS point-to-point circuits in an east-to-west pattern, as shown in [Figure 17-5 on page 17-8](#).

Step 21 The CTC circuit process is complete.

Configuring RPR Characteristics and the SPR Interface on the ML-Series Card

You configure RPR on the ML-Series cards by creating an SPR interface using the Cisco IOS command-line interface (CLI). The SPR interface is a virtual interface for the SPR. An ML-Series card supports a single SPR interface with a single MAC address. It provides all the normal attributes of a Cisco IOS virtual interface, such as support for default routes.

An SPR interface is configured similarly to a EtherChannel (port-channel) interface. Instead of using the **channel-group** command to define the members, you use the **spr-intf-id** command. Like the port-channel interface, you configure the virtual SPR interface instead of the physical POS interface. An SPR interface is considered a trunk port, and like all trunk ports, subinterfaces must be configured for the SPR interface for it to join a bridge group.

The physical POS interfaces on the ML-Series card are the only members eligible for the SPR interface. One POS port is associated with the SONET/SDH circuit heading east around the ring from the node, and the other POS port is associated with the circuit heading west. When the SPR interface is used and the POS ports are associated, RPR encapsulation is used on the SONET/SDH payload.



Caution

In configuring an SPR, if one ML-Series card is not configured with an SPR interface, but valid STS/STM circuits connect this ML-Series card to the other ML-Series cards in the SPR, no traffic will flow between the properly configured ML-Series cards in the SPR, and no alarms will indicate this condition. Cisco recommends that you configure all of the ML-Series cards in an SPR before sending traffic.

**Caution**

Do not use native VLANs for carrying traffic with RPR.

**Note**

RPR on the ML-Series card is only supported with the default LEX encapsulation, a special CISCO-EOS-LEX encapsulation for use with Cisco ONS Ethernet line cards.

RPR needs to be provisioned on each ML-Series card that is in the RPR. To provision RPR, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# bridge irb	Enables the Cisco IOS software to both route and bridge a given protocol on separate interfaces within a single ML-Series card.
Step 2	Router(config)# interface spr 1	Creates the SPR interface on the ML-Series card or enters the SPR interface configuration mode. The only valid SPR number is 1.
Step 3	Router(config-if)# spr station-id <i>station-ID-number</i>	Configures a station ID. The user must configure a different number for each SPR interface that attaches to the RPR. Valid station ID numbers range from 1 to 254.
Step 4	Router(config-if)# spr wrap { immediate delayed }	(Optional) Sets the RPR ring wrap mode to either wrap traffic the instant it detects a SONET/SDH path alarm or to wrap traffic after the 200 msec delay, which gives the SONET/SDH protection time to register the defect and declare the link down. Use immediate if RPR is running over unprotected SONET/SDH circuits. Use delayed for bidirectional line switched rings (BLSR), path protection, multiplex section-shared protection ring (MS-SPRing), or SNCP protected circuits. The default setting is immediate .
Step 5	Router(config-if)# carrier-delay msec <i>milliseconds</i>	(Optional) Sets the carrier delay time. The default setting is 200 milliseconds, which is optimum for SONET/SDH protected circuits. Note If the carrier delay time is changed from the default, the new carrier delay time must be configured on all the ML-Series card interfaces, including the SPR, POS, and Gigabit Ethernet or Fast Ethernet interfaces.
Step 6	Router(config-if)# [no] spr load-balance { auto port-based }	(Optional) Specifies the RPR load-balancing scheme for unicast packets. The port-based load balancing option maps even ports to the POS 0 interface and odd ports to the POS 1 interface. The default auto option balances the load based on the MAC addresses or source and destination addresses of the IP packet.
Step 7	Router(config-if)# end	Exits to privileged EXEC mode.
Step 8	Router# copy running-config startup-config	(Optional) Saves configuration changes to NVRAM.

Assigning the ML-Series Card POS Ports to the SPR Interface



Caution

The SPR interface is the routed interface. Do not enable Layer 3 addresses or assign bridge groups on the POS interfaces assigned to the SPR interface.




Caution

When traffic coming in on an SPR interface needs to be policed, the same input service policy needs to be applied to both POS ports that are part of the SPR interface.

The POS ports require LEX encapsulation to be used in RPR. The first step of RPR configuration is to set the encapsulation of POS 0 and POS 1 ports to LEX.

Each of the ML-Series card's two POS ports must also be assigned to the SPR interface. To configure LEX encapsulation and assign the POS interfaces on the ML-Series card to the SPR, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# interface pos 0	Enters the interface configuration mode to configure the first POS interface that you want to assign to the SPR.
Step 2	Router(config-if)# encapsulation lex	Sets POS interface encapsulation as LEX (default). RPR on the ML-Series card requires LEX encapsulation.
Step 3 (Router(config-if)# spr-intf-id <i>shared-packet-ring-number</i>	Assigns the POS interface to the SPR interface. The shared packet ring number must be 1, which is the only shared packet ring number that you can assign to the SPR interface.
Step 4	Router(config-if)# carrier-delay msec <i>milliseconds</i>	(Optional) Sets the carrier delay time. The default setting is 200 msec, which is optimum for SONET/SDH protected circuits. Note The default unit of time for setting the carrier delay is seconds. The msec command resets the time unit to milliseconds.
Step 5	Router(config-if)# pos trigger defect ber_sd-b3	(Optional) Configures a trigger to bring down the POS interface when the SONET/SDH bit error rate exceeds the threshold set for the signal degrade alarm. Bringing the POS interface down initiates the RPR wrap. This command is recommended for all RPR POS interfaces, since excessive SONET/SDH bit errors can cause packet loss on RPR traffic.  Note This command should not be used when a Cisco ONS 15310 is part of the ring. It may cause inconsistent RPR wrapping.
Step 6	Router(config-if)# no shutdown	Enables the POS port.
Step 7	Router(config-if)# interface pos 1	Enters the interface configuration mode to configure the second POS interface that you want to assign to the SPR.

	Command	Purpose
Step 8	Router(config-if)# encapsulation lex	Sets POS interface encapsulation as LEX (default). RPR on the ML-Series card requires LEX encapsulation.
Step 9	Router(config-if)# spr-intf-id <i>shared-packet-ring-number</i>	Assigns the POS interface to the SPR interface. The shared packet ring number must be 1 (the same shared packet ring number that you assigned in Step 3), which is the only shared packet ring number that you can assign to the SPR interface.
Step 10	Router(config-if)# carrier-delay msec <i>milliseconds</i>	(Optional) Sets the carrier delay time. The default setting is 200 milliseconds, which is optimum for SONET/SDH protected circuits.
Step 11	Router(config-if)# pos trigger defect ber_sd-b3	(Optional) Configures a trigger to bring down the POS interface when the SONET/SDH bit error rate exceeds the threshold set for the signal degrade alarm. Bringing the POS interface down initiates the RPR wrap. This command is recommended for all RPR POS interfaces since excessive SONET/SDH bit errors can cause packet loss on RPR traffic.
Step 12	Router(config-if)# no shutdown	Enables the POS port.
Step 13	Router(config-if)# end	Exits to privileged EXEC mode.
Step 14	Router# copy running-config startup-config	(Optional) Saves the configuration changes to NVRAM.

Creating the Bridge Group and Assigning the Ethernet and SPR Interfaces

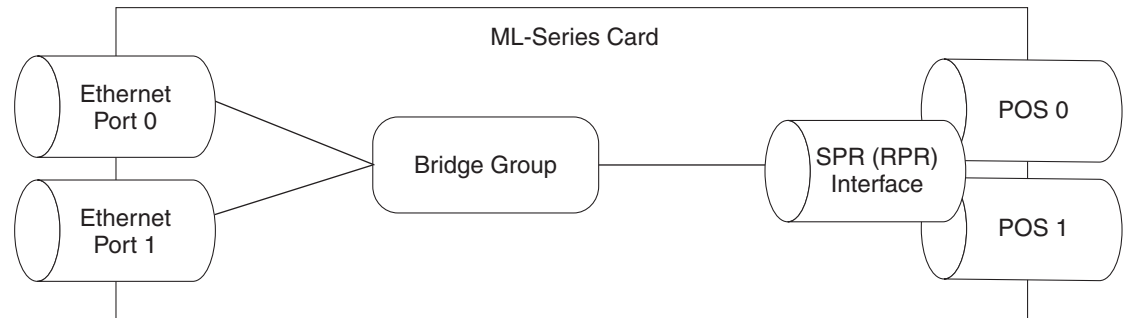
The default behavior of the ML-Series cards is that no traffic is bridged over the RPR even with the interfaces enabled. This is in contrast to many Layer 2 switches, including the Cisco Catalyst 6500 and the Cisco Catalyst 7600, which forward VLAN 1 by default. The ML-Series card will not forward any traffic by default, including untagged or VLAN 1 tagged packets.

For any RPR traffic to be bridged on an ML-Series card, a bridge group needs to be created for that traffic. Bridge groups maintain the bridging and forwarding between the interfaces on the ML-Series card and are locally significant. Interfaces not participating in a bridge group cannot forward bridged traffic.

To create a bridge group for RPR, you determine which Ethernet interfaces need to be in the same bridge group, create the bridge group, and associate these interfaces with the bridge group. Then associate the SPR interface with the same bridge group to provide transport across the RPR infrastructure.

[Figure 17-8](#) illustrates a bridge group spanning the ML-Series card interfaces, including the SPR virtual interface of RPR.

Figure 17-8 RPR Bridge Group



134983

**Caution**

All Layer 2 network redundant links (loops) in the connecting network, except the RPR topology, must be removed for correct RPR operation. Or if loops exist, you must configure STP/RSTP.

To configure the needed interfaces, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# interface <i>type number</i>	Enters interface configuration mode for the Ethernet interface joining the bridge group.
Step 2	Router(config-if)# no shutdown	Enables the interface.
Step 3	Router(config-if)# bridge-group <i>bridge-group-number</i>	Creates the specified bridge group and assigns the bridge group to the interface. Creating the bridge from the interface configuration disables STP or RSTP (spanning-disabled), which is recommended for RPR.
Step 4	Router(config)# interface spr1	Enters interface configuration mode for the SPR
Step 5	Router(config-subif)# bridge-group <i>bridge-group-number</i>	Associates the SPR interface to the specified bridge group.

RPR Cisco IOS Configuration Example

Figure 17-5 on page 17-8 shows a complete example of an RPR Cisco IOS configuration. The associated Cisco IOS code is provided in Examples 17-1, 17-2, and 17-3. The configuration assumes that ML-Series card POS ports are already linked by point-to-point SONET/SDH circuits configured through CTC.

Example 17-1 SPR Station-ID 1 Configuration

```
bridge irb

interface SPR1
no ip address
no keepalive
spr station-id 1
bridge-group 10
bridge-group 10 spanning-disabled
hold-queue 150 in
```

```

interface GigabitEthernet0
no ip address
bridge-group 10
bridge-group 10 spanning-disabled

interface GigabitEthernet1
no ip address
shutdown

interface POS0
no ip address
carrier-delay msec 0
spr-intf-id 1
crc 32

interface POS1
no ip address
carrier-delay msec 0
spr-intf-id 1
crc 32
!
```

Example 17-2 SPR Station-ID 2 Configuration

```

bridge irb

interface SPR1
no ip address
no keepalive
spr station-id 2
bridge-group 10
bridge-group 10 spanning-disabled

interface GigabitEthernet0
no ip address
bridge-group 10
bridge-group 10 spanning-disabled

interface GigabitEthernet1
no ip address
shutdown

interface POS0
no ip address
shutdown
spr-intf-id 1
crc 32

interface POS1
no ip address
spr-intf-id 1
crc 32
```

Example 17-3 SPR Station-ID 3 Configuration

```

bridge irb

interface SPR1
no ip address
no keepalive
spr station-id 3
```

```

bridge-group 10
bridge-group 10 spanning-disabled
hold-queue 150 in

interface GigabitEthernet0
no ip address
bridge-group 10
bridge-group 10 spanning-disabled

interface GigabitEthernet1
no ip address
shutdown

interface POS0
no ip address
spr-intf-id 1
crc 32

interface POS1
no ip address
spr-intf-id 1
crc 32
!
```

Verifying Ethernet Connectivity Between RPR Ethernet Access Ports

After successfully completing the procedures to provision an RPR, you can test Ethernet connectivity between the Ethernet access ports on the separate ML-Series cards using your standard Ethernet connectivity testing.

CRC threshold configuration and detection

You can configure a span shutdown when the ML-Series card receives CRC errors at a rate that exceeds the configured threshold and configured soak time. For this functionality to work in an SPR ring, make the configurations on the POS members of SPR interface as specified in [CRC Threshold Configuration, page 4-10](#).

Monitoring and Verifying RPR

After RPR is configured, you can monitor its status using the **show interface spr 1** command ([Example 17-4](#)) or the **show run interface spr 1** command ([Example 17-5](#)).

Example 17-4 Example of show interface spr 1 Output

```

ML-Series# show interfaces spr 1

SPR1 is up, line protocol is up
  Hardware is POS-SPR, address is 0005.9a39.77f8 (bia 0000.0000.0000)
  MTU 1500 bytes, BW 290304 Kbit, DLY 100 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation: Cisco-EoS-LEX, loopback not set
  Keepalive not set
  DTR is pulsed for 27482 seconds on reset, Restart-Delay is 65 secs
  ARP type: ARPA, ARP Timeout 04:00:00
```

```

No. of active members in this SPR interface: 2
  Member 0 : POS1
  Member 1 : POS0
Last input 00:00:38, output never, output hang never
Last clearing of "show interface" counters never
Input queue: 0/150/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
Output queue: 0/80 (size/max)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
  37385 packets input, 20993313 bytes
    Received 0 broadcasts (0 IP multicast)
    0 runts, 0 giants, 0 throttles
    0 parity
  2 input errors, 2 CRC, 0 frame, 0 overrun, 0 ignored
  0 input packets with dribble condition detected
37454 packets output, 13183808 bytes, 0 underruns
  0 output errors, 0 applique, 4 interface resets
  0 babbles, 0 late collision, 0 deferred
  0 lost carrier, 0 no carrier
  0 output buffer failures, 0 output buffers swapped out
  0 carrier transitions

```

Example 17-5 Example of show run interface spr 1 Output

```

ML-Series# show run interface spr 1

Building configuration...
Current configuration : 141 bytes
interface SPR1
 no ip address
 no keepalive
 spr station-id 2
 bridge-group 10
 bridge-group 10 spanning-disabled
 hold-queue 150 in
end

```

Add an ML-Series Card into an RPR

An existing RPR might need an ML-Series card added. This can be done without taking down data traffic due to the RPR wrapping capability and ring architecture. You can add the ML-Series card in concert with the addition of the node containing the card into the underlying SONET/SDH architecture. You can also add an ML-Series card to a node that is already part of the SONET/SDH topology.

The following example has a two-node RPR with two STS circuits connecting the ML-Series cards. One circuit will be deleted. The RPR will wrap traffic on the remaining circuit with as little as a one ping loss. The third node and ML-Series card are then added in, and the spans and circuits for this card are created.

Figure 17-9 shows the existing two-node RPR with the single STS circuit and span that will be deleted. Figure 17-10 shows the RPR after the third node is added with the two new STS circuits and spans that will be added.

Figure 17-9 Two-Node RPR Before the Addition

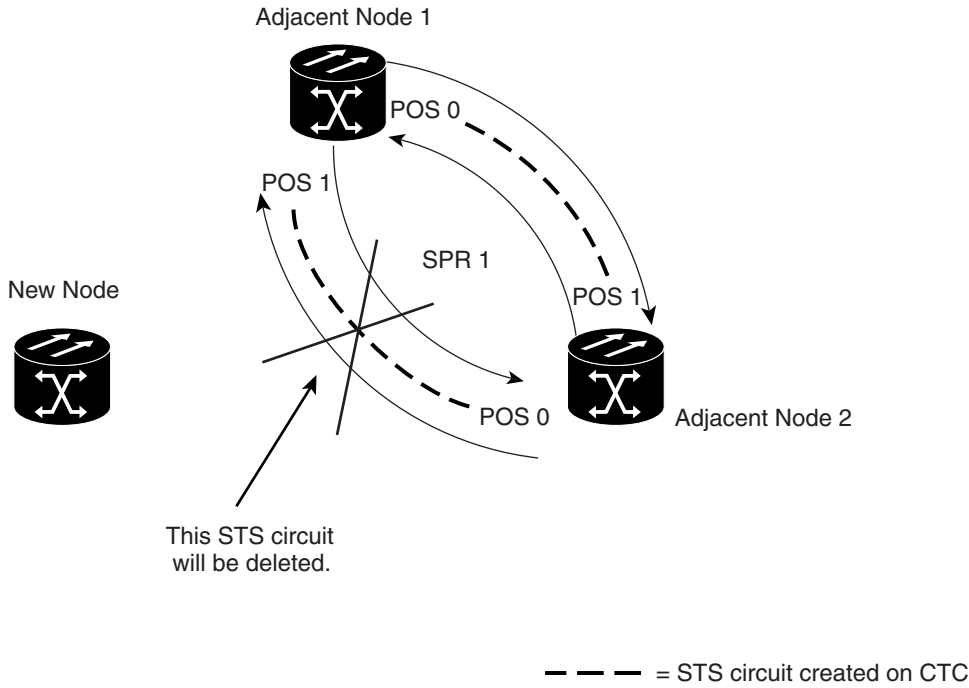
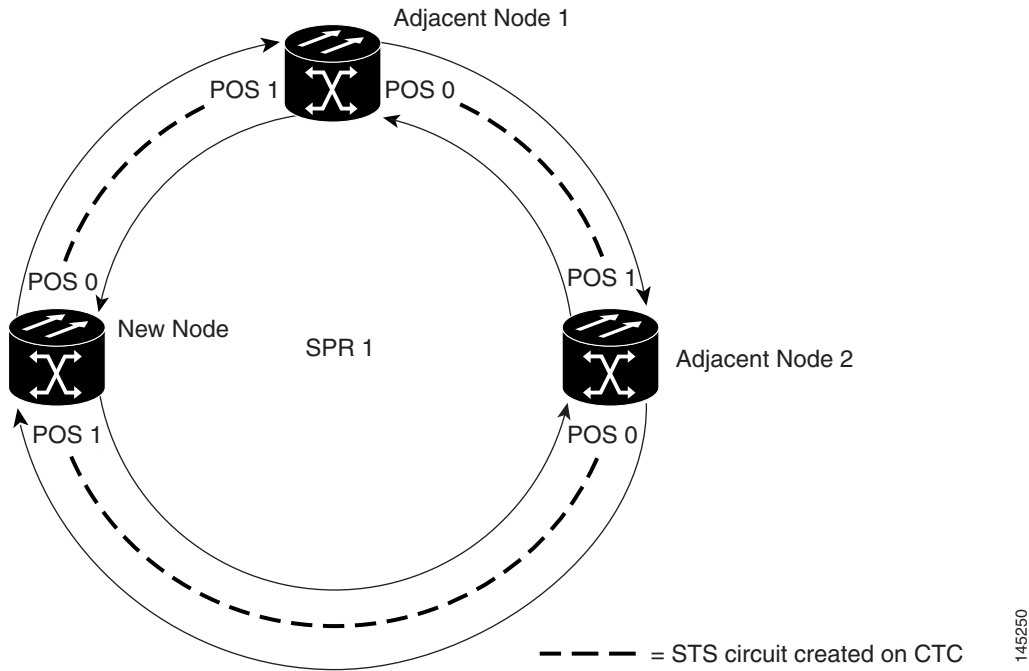


Figure 17-10 Three Node RPR After the Addition



To add an ML-Series card to the RPR, you need to complete several general actions:

- Force away any existing non-ML-Series card circuits, such as DS-1, that use the span that will be deleted.
- Shut down the POS ports on the adjacent ML-Series cards for the STS circuit that will be deleted to initiate the RPR wrap.
- Test Ethernet connectivity between the access ports on the existing adjacent ML-Series cards with a test set to ensure that the RPR wrapped successfully.
- Delete the STS circuit that will be replaced by the new circuits. (In [Figure 17-9](#), this is the circuit between Adjacent Node 2, POS 0 and Adjacent Node 1, POS 1.)
- Insert the new node into the ring topology if the node is not already part of the topology.
- Install the ML-Series card and load your initial configuration file or otherwise do an initial configuration of the ML-Series card.
- Ensure the new node is configured with RPR before its POS ports are manually enabled or enabled through the configuration file.
- Create an STS circuit from one of the POS ports of an existing adjacent ML-Series card to a POS port on the new ML-Series card. (In [Figure 17-10](#), this is the circuit between Adjacent Node 2, POS Port 0 and New Node, POS Port 1.)
- Create a second STS circuit from one of the POS ports of the other existing adjacent ML-Series card to the remaining POS port on the new ML-Series card. (In [Figure 17-10](#), this is the circuit between New Node, POS Port 0 and Adjacent Node 1, POS Port 1.)
- Configure the new ML-Series card to join the RPR and enable the POS ports, if the initial configuration file did not already do this.
- Enable the POS ports on the existing adjacent ML-Series cards that connect to the new ML-Series card. (In [Figure 17-10](#), these are Adjacent Node 1, POS Port 1 and Adjacent Node 2, POS Port 0.)
- Test Ethernet connectivity between the access ports on the new ML-Series card with a test set to validate the newly created three-node RPR.
- Monitor Ethernet traffic and existing routing protocols for at least an hour after the node insertion.

**Caution**

The specific steps in the following procedure are for the topology in the example. Your own steps will vary according to your network design. Do not attempt this procedure without obtaining a detailed plan or method of procedure from an experienced network architect.

Adding an ML-Series Card into an RPR

To add an ML-Series card to the RPR in the example, complete the following procedure:

- Step 1** Start a Cisco IOS CLI session for the ML-Series card in the first adjacent node. This is Adjacent Node 1 in [Figure 17-9](#).
- Step 2** Complete the following Cisco IOS configuration on the ML-Series card in the first adjacent node, beginning in global configuration mode:

a.	Router(config)# interface pos <i>interface-number</i>	Enters interface configuration mode for the POS port at one endpoint of the circuit to be deleted.
b.	Router(config-if)# shutdown	Closes the interface, which initiates the RPR wrap.

- Step 3** Start a Cisco IOS CLI session for the ML-Series card in Adjacent Node 2, as shown in [Figure 17-9](#).
- Step 4** Complete the following Cisco IOS configuration on the Adjacent Node 2 ML-Series card, beginning in global configuration mode:

a.	Router(config)# interface pos <i>interface-number</i>	Enters interface configuration mode for the POS port at one endpoint of the circuit to be deleted.
b.	Router(config-if)# shutdown	Closes the interface.

- Step 5** In CTC, log into Adjacent Node 1.
- Step 6** Double-click the ML-Series card in Adjacent Node 1.
The card view appears.
- Step 7** Click the **Circuits** tab.
- Step 8** Click the **Circuits** subtab.
- Step 9** Identify the appropriate STS circuit by looking under the source column and destination column for the circuit entry that matches the POS ports at the endpoints of the circuit to be deleted.
The circuit entry is in *node-name/card-slot/port-number* format, such as Node-1/s12(ML100T)/pPOS-0.
- Step 10** Click the circuit entry to highlight it.
- Step 11** Click **Delete**.
A confirmation dialog box appears.
- Step 12** Click **Yes**.
- Step 13** Use a test set to verify that Ethernet connectivity still exists between the Ethernet access ports on Adjacent Node 1 and Adjacent Node 2.



Note The SPR interface and the Ethernet interfaces on the ML-Series card must be in a bridge group in order for RPR traffic to bridge the RPR.

- Step 14** If the new node is not already an active node in the SONET/SDH ring topology, add the node to the ring. Refer to the “Add and Remove Nodes” chapter of the *Cisco ONS 15454 Procedure Guide* or the “Add and Remove Nodes” chapter of the *Cisco ONS 15454 SDH Procedure Guide* for procedures for installing ONS nodes.
- Step 15** If the ML-Series card in the new node is not already installed, install the card in the node. Refer to the “Install Cards and Fiber-Optic Cable” chapter of the *Cisco ONS 15454 Procedure Guide* or the “Install Cards and Fiber-Optic Cable” chapter of the *Cisco ONS 15454 SDH Procedure Guide* for procedures for installing cards in ONS nodes.
- Step 16** Upload the initial startup configuration file for the new ML-Series card (see the “[CTC and the Startup Configuration File](#)” section on page 3-9.) If you do not have a prepared startup configuration file, see the “[Manually Creating a Startup Configuration File Through the Serial Console Port](#)” section on page 3-7.



Caution Ensure the new node is configured with RPR before its POS ports are manually enabled or enabled through the configuration file.

Step 17 Build an STS circuit with a circuit state of In Service (IS) from the available POS port on Adjacent Node 1 to the New Node, as shown in [Figure 17-10](#). On the New Node, use the POS port with the interface-number that does not match the interface-number of the available POS port on Adjacent Node 1. For example, POS Port 0 on Adjacent Node 1 would connect to POS Port 1 on the New Node.

For detailed steps for building the circuit, see the “[Configuring CTC Circuits for RPR](#)” section on [page 17-7](#).



Note A best practice is to configure SONET/SDH circuits in an east-to-west or west-to-east configuration, from Port 0 (east) to Port 1 (west) or Port 1 (east) to Port 0 (west), around the SONET/SDH ring.

Step 18 Build an STS circuit with a circuit state of IS from the available POS port on Adjacent Node 2 to the remaining POS port on the New Node, as shown in [Figure 17-10](#).

Step 19 Start or resume a Cisco IOS CLI session for the ML-Series card in Adjacent Node 1, as shown in [Figure 17-9](#).

Step 20 Complete the following Cisco IOS configuration, beginning in global configuration mode:

a.	<code>Router(config)# interface pos interface-number</code>	Enters interface configuration mode for the POS port at one endpoint of the first newly created circuit.
b.	<code>Router(config-if)# no shutdown</code>	Enables the port.

Step 21 Start a Cisco IOS CLI session for the ML-Series card in Adjacent Node 2, as shown in [Figure 17-9](#).

Step 22 Complete the following Cisco IOS configuration on the Adjacent Node 2 ML-Series card, beginning in global configuration mode:

a.	<code>Router(config)# interface pos interface-number</code>	Enters interface configuration mode for the POS port at one endpoint of the second newly created circuit.
b.	<code>Router(config-if)# no shutdown</code>	Enables the port.

Step 23 Use a test set to verify that Ethernet connectivity exists on the RPR.

Step 24 Monitor Ethernet traffic and routing tables for at least one hour after the node insertion.

Stop. You have completed this procedure.

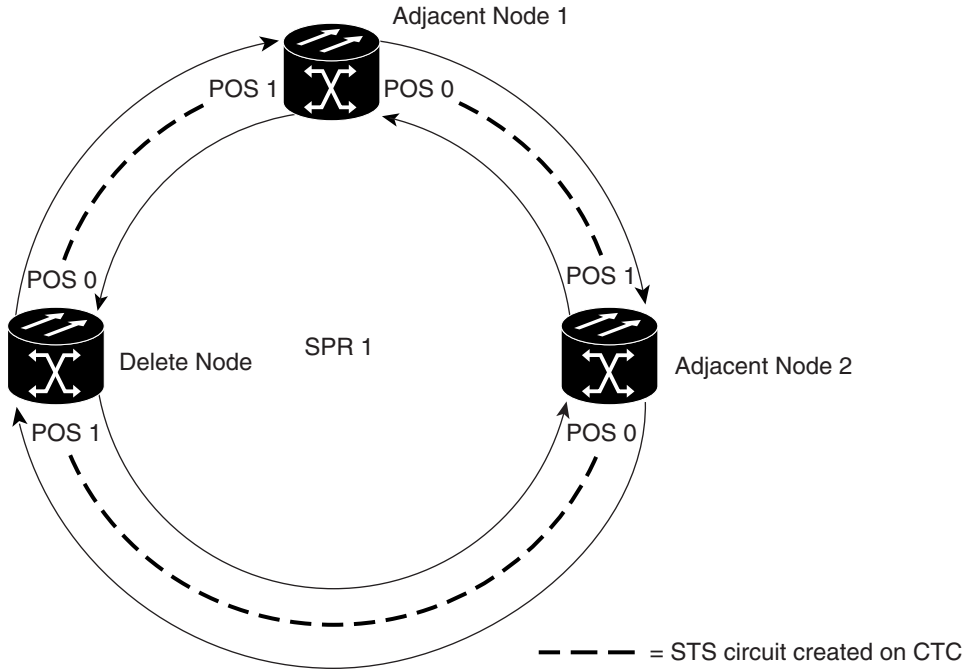
Delete an ML-Series Card from an RPR

An existing RPR might need an ML-Series card deleted. This can be done without taking down data traffic due to the RPR wrapping capability and ring architecture.

The following example has a three-node RPR with three STS circuits connecting the ML-Series cards. Two circuits will be deleted. The RPR will wrap traffic on the remaining circuit with as little as a one ping loss. The third node and ML-Series card are then deleted and a new STS circuit is created between the two remaining cards.

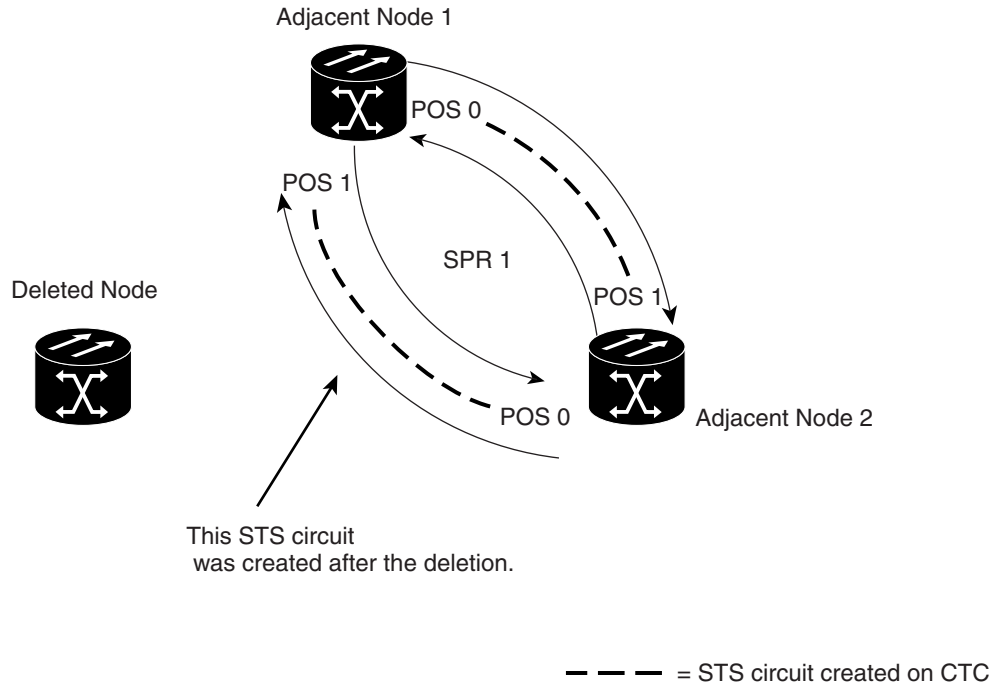
Figure 17-11 shows the existing three-node RPR with all three STS circuits and spans. Figure 17-12 shows the RPR after the third node, circuits, and spans are deleted and the new STS circuit and span are added.

Figure 17-11 Three Node RPR Before the Deletion



145251

Figure 17-12 Two Node RPR After the Deletion



To delete an ML-Series card from the RPR, you need to complete several general actions:

- Shut down the POS ports on the ML-Series card to be deleted with the POS interface level configuration **shut** command.
- Force away any existing non-ML-Series card circuits, such as DS-1, that use the spans that will be deleted.
- Shut down the POS ports on the adjacent ML-Series cards for the STS circuits that will be deleted to initiate the RPR wrap.
- Test Ethernet connectivity between the access ports on the existing adjacent ML-Series cards with a test set to ensure that the RPR wrapped successfully.
- Delete the two STS circuits that will be replaced by the new circuits. (In Figure 17-11, this is the circuit between the Delete Node and one Adjacent Node, and the circuit between the Delete Node and the other Adjacent Node.)
- Remove the Delete Node from the ring topology if desired.
- Physically remove the delete ML-Series card from the node if desired.
- Create an STS circuit from the available POS port of one of the remaining adjacent ML-Series cards to the available POS port on the other remaining adjacent ML-Series card. (In Figure 17-12, this is the circuit between Adjacent Node 2, POS Port 0 and Adjacent Node 1, POS Port 1.)
- Enable the POS ports on the existing adjacent ML-Series cards. (In Figure 17-12, this is the Adjacent Node 2, POS Port 0 and the Adjacent Node 1, POS Port 1.)
- Test Ethernet connectivity between the access ports on the adjacent ML-Series card with a test set to validate the two-node RPR.
- Monitor Ethernet traffic and existing routing protocols for at least an hour after the node deletion.

**Caution**

When taking an ML-Series card POS port out of an in-service RPR, you must first shut down the POS interface using the POS interface level configuration **shut** command. Do this before entering any addition commands.

**Caution**

The specific steps in the following procedure are for the topology in the example. Your own steps will vary according to your network design. Do not attempt this procedure without obtaining a detailed plan or method of procedure from an experienced network architect.

Deleting an ML-Series Card from an RPR

To delete an ML-Series card from an RPR, complete the following procedure:

- Step 1** Start a Cisco IOS CLI session for the ML-Series card on the delete node. This is the Delete Node in [Figure 17-11](#).
- Step 2** Complete the following Cisco IOS configuration on the ML-Series card in the node to be deleted, beginning in global configuration mode:
- | | | |
|----|--|--|
| a. | Router(config)# interface pos 0 | Enters interface configuration mode for the POS port 0 on the Delete Node. |
| b. | Router(config-if)# shutdown | Closes the interface. |
| c. | Router(config)# interface pos 1 | Enters interface configuration mode for the POS port 1 on the Delete Node. |
| d. | Router(config-if)# shutdown | Closes the interface. |
- Step 3** Start a Cisco IOS CLI session for the ML-Series card on the first adjacent node. This is Adjacent Node 1 in [Figure 17-11](#).
- Step 4** Complete the following Cisco IOS configuration on the ML-Series card in the first adjacent node, beginning in global configuration mode:
- | | | |
|----|---|---|
| a. | Router(config)# interface pos
<i>interface-number</i> | Enters interface configuration mode for the POS port at the end of the circuit directly connected to the Delete Node. |
| b. | Router(config-if)# shutdown | Closes the interface, which initiates the RPR wrap. |
- Step 5** Start a Cisco IOS CLI session for the ML-Series card in Adjacent Node 2, as shown in [Figure 17-11](#).
- Step 6** Complete the following Cisco IOS configuration on the Adjacent Node 2 ML-Series card, beginning in global configuration mode:
- | | | |
|----|---|---|
| a. | Router(config)# interface pos
<i>interface-number</i> | Enters interface configuration mode for the POS port at the end of the circuit directly connected to the Delete Node. |
| b. | Router(config-if)# shutdown | Closes the interface. |

- Step 7** Log into Adjacent Node 1 with CTC.
- Step 8** Double-click the ML-Series card in Adjacent Node 1.
The card view appears.
- Step 9** Click the **Circuits** tab.
- Step 10** Click the **Circuits** subtab.
- Step 11** Identify the appropriate STS circuit by looking under the source column and destination column for the circuit entry that matches the POS ports at the endpoints of the first circuit to be deleted.
The circuit entry is in *node-name/card-slot/port-number* format, such as Node-1/s12(ML100T)/pPOS-0.
- Step 12** Click the circuit entry to highlight it.
- Step 13** Click **Delete**.
A confirmation dialog box appears.
- Step 14** Click **Yes**.
- Step 15** Verify that Ethernet connectivity still exists between the Ethernet access ports on Adjacent Node 1 and Adjacent Node 2 by using a test set.



Note The SPR interface and the Ethernet interfaces on the ML-Series card must be in a bridge group in order for RPR traffic to bridge the RPR.

- Step 16** Log into Adjacent Node 2 with CTC.
- Step 17** Double-click the ML-Series card in Adjacent Node 2.
The card view appears.
- Step 18** Click the **Circuits** tab.
- Step 19** Click the **Circuits** subtab.
- Step 20** Identify the appropriate STS circuit by looking under the source column and destination column for the circuit entry that matches the POS ports at the endpoints of the second circuit to be deleted.
The circuit entry is in *node-name/card-slot/port-number* format, such as Node-1/s12(ML100T)/pPOS-0.
- Step 21** Click the circuit entry to highlight it.
- Step 22** Click **Delete**.
The confirmation dialog box appears.
- Step 23** Click **Yes**.
- Step 24** If the new node will no longer be an active node in the SONET/SDH ring topology, delete the node from the ring. Refer to the “Add and Remove Nodes” chapter of the *Cisco ONS 15454 Procedure Guide* or the “Add and Remove Nodes” chapter of the *Cisco ONS 15454 SDH Procedure Guide* for procedures for removing ONS nodes.
- Step 25** If the ML-Series card in the new node is to be deleted in CTC and physically removed, do so now. Refer to the “Install Cards and Fiber-Optic Cable” chapter of the *Cisco ONS 15454 Procedure Guide* or the “Install Cards and Fiber-Optic Cable” chapter of the *Cisco ONS 15454 SDH Procedure Guide* for procedures for installing cards in ONS nodes.
- Step 26** Build an STS circuit with a circuit state of IS from the available POS port on Adjacent Node 1 to the available POS port on Adjacent Node 2, as shown in [Figure 17-12](#). For detailed steps on building the circuit, see “[Configuring CTC Circuits for RPR](#)” section on page 17-7.



Note A best practice is to configure SONET/SDH circuits in an east-to-west or west-to-east configuration, from Port 0 (east) to Port 1 (west) or Port 1 (east) to Port 0 (west), around the SONET/SDH ring.

- Step 27** Start or resume a Cisco IOS CLI session for the ML-Series card in Adjacent Node 1.
- Step 28** Complete the following Cisco IOS configuration for the ML-Series card in Adjacent Node 1, beginning in global configuration mode:

a.	Router(config)# interface pos <i>interface-number</i>	Enters interface configuration mode for the POS port at one endpoint of the first newly created circuit.
b.	Router(config-if)# no shutdown	Enables the port.

- Step 29** Start a Cisco IOS CLI session for the ML-Series card in Adjacent Node 2.
- Step 30** Complete the following Cisco IOS configuration on the Adjacent Node 2 ML-Series card, beginning in global configuration mode:

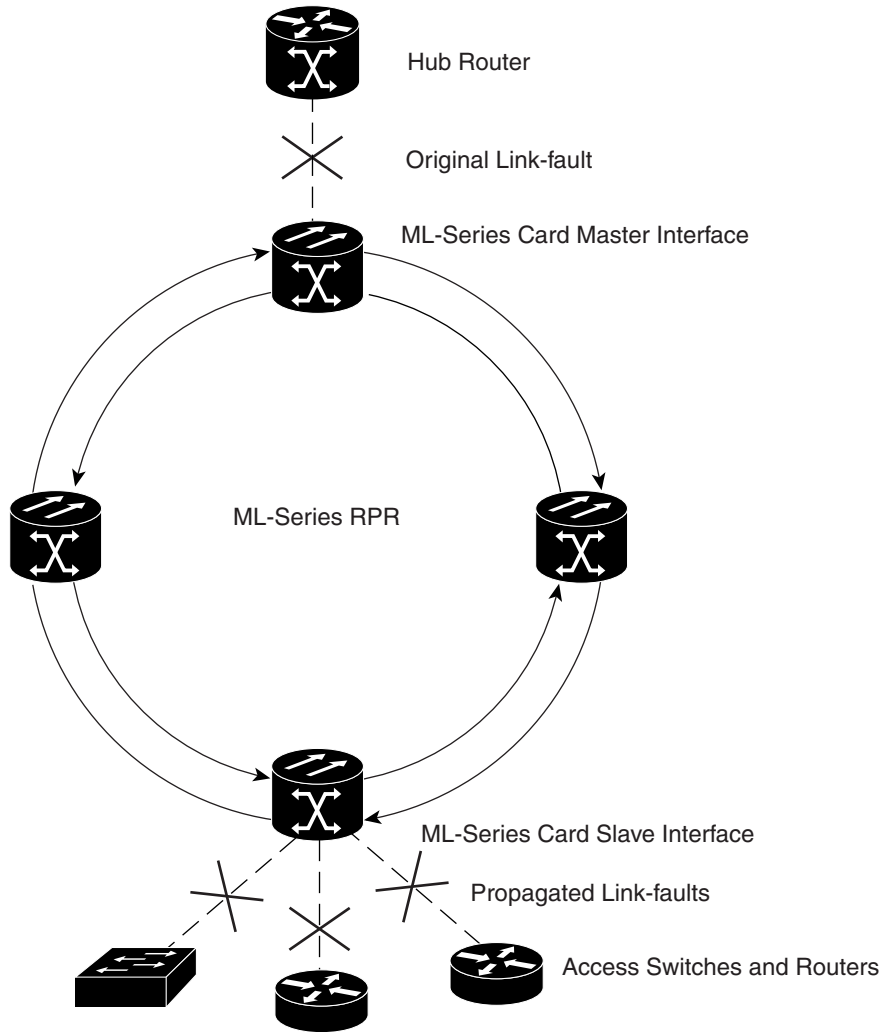
a.	Router(config)# interface pos <i>interface-number</i>	Enters interface configuration mode for the POS port at one endpoint of the second newly created circuit.
b.	Router(config-if)# no shutdown	Enables the port.

- Step 31** Use a test set to verify that Ethernet connectivity exists on the RPR.
- Step 32** Monitor Ethernet traffic and routing tables for at least one hour after the node deletion.
- Stop. You have completed this procedure.**

Understanding RPR Link Fault Propagation

Link fault propagation (LFP), also known as link pass-through, decreases convergence times in networks where routers interconnect through ML-Series card RPRs. It quickly relays link faults from a master Gigabit Ethernet link to a remote slave link, either Gigabit Ethernet or Fast Ethernet. LFP greatly improves the time it takes for a router connected to the slave link to fail over to an alternate path. Under normal protection schemes, convergence might take as long as forty seconds. Using LFP, the slave interface reflects the state of the master interface in less than a second. This feature is often used to enable a link failure at a far-end hub site in order to trigger a link down state at a near-end access site. [Figure 17-13](#) illustrates LFP.

Figure 17-13 RPR Link Fault Propagation Example



131696

LFP Sequence

LFP updates are done through a CDP packet extension. The update is sent periodically and immediately after the master interface goes into a link-down state. LFP updates are sent separately from normal Cisco discovery packets (CDP), and the two types do not interact. Configuring or disabling CDP on the interface has no effect on LFP updates.

When the master interface goes down, including an administrative shutdown, the slave interface is forced down. When the master interface goes up, the slave interface goes back up. Administrative shutdown on a slave interface will suspend the LFP function on that interface, and removing the shutdown will reactivate LFP.

A link-down fault is also forced onto the slave link if the connection from the master to the slave fails. Any of the following can cause a loss of connection:

- Removing or resetting the master ML-Series card.
- Shutdown or failure on both of the RPR paths between master and slave.

- Disabling LFP on the master interface.

Link faults only propagate from master to slave. Normal slave link faults are not propagated. RPR wrapping and unwrapping has no effect on LFP.

Propagation Delays

Propagation delay includes the carrier-delay time on the slave interface. The carrier-delay time is configurable and has a default of 200 ms. See the “[Configuring RPR](#)” section on page 17-7 for more information on configuring carrier-delay time.

Different propagation delays apply to different LFP scenarios:

- Propagation delay between master link-down and slave link-down is 50 ms plus the carrier-delay time on the slave interface.
- Propagation delay between master link-up and slave link-up has an additional built-in delay at the master interface to prevent interface flapping. Link-up propagation takes approximately 50 to 200 ms plus the carrier-delay time on the slave interface.
- Propagation delay from when the master-to-slave link fails until slave link-down occurs is approximately 600 ms plus the carrier-delay time on the slave interface.

Configuring LFP

[Figure 17-13 on page 17-28](#) illustrates an example of RPR configured with LFP. The process of configuring LFP consists of the following tasks:

1. Configure one ML-Series card Gigabit Ethernet interface as a master link.
2. Configure other ML-Series cards’ Gigabit Ethernet or Fast Ethernet interfaces as slave links.

To enable and configure the LFP master link, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router# interface gigabit ethernet <i>number</i>	Activates interface configuration mode to configure the Gigabit Ethernet interface.
Step 2	Router(config-if)# link-fault rpr-master	Enables link-fault master status on the interface. The no form of this command disables link-fault master status.
Step 3	Router(config-if)# no shutdown	Enables the interface by preventing it from shutting down.
Step 4	Router(config)# end	Returns to privileged EXEC mode.
Step 5	Router# copy running-config startup-config	(Optional) Saves configuration changes to the TCC2/TCC2P flash database.

To enable and configure the LFP slave link, perform the following procedure on an ML-Series card in the RPR other than the ML-Series card configured for the master link. Begin in global configuration mode:

	Command	Purpose
Step 1	Router# interface [gigabit ethernet fastethernet] <i>number</i>	Activates interface configuration mode to configure the Gigabit Ethernet or Fast Ethernet interface.
Step 2	Router(config-if)# link-fault rpr-slave	Enables link-fault slave status on the interface. The no form of this command disables link-fault slave status.
Step 3	Router(config-if)# no shutdown	Enables the interface by preventing it from shutting down.
Step 4	Router(config)# end	Returns to privileged EXEC mode.
Step 5	Router# copy running-config startup-config	(Optional) Saves configuration changes to the TCC2/TCC2P flash database.

LFP Configuration Requirements

LFP has these configuration requirement:

- A link-fault master and slave should not be configured on the same card.
- The ML-Series card must be running the Enhanced microcode image.
- All ML-Series cards in the RPR must be running Software Release 5.0 or later.
- ML-Series card configured for DRPRI should not be configured for LFP, and LFP on DRPRI is unsupported.
- Only ML-Series card Gigabit Ethernet interfaces are eligible to become link-fault masters.
- Only one link-fault master is allowed per RPR.
- Gigabit Ethernet and Fast Ethernet interfaces are both eligible to become link-fault slaves.
- There is no configuration limit on link-fault slaves on an RPR.

Monitoring and Verifying LFP

A slave interface in link-down state raises a carrier loss (CARLOSS) alarm in CTC. CTC does not distinguish between a local loss on the slave link and loss due to LFP. For more information on CARLOSS, refer to the “Alarm Troubleshooting” chapter of the *Cisco ONS 15454 Troubleshooting Guide* or the “Alarm Troubleshooting” chapter of the *Cisco ONS 15454 SDH Troubleshooting Guide*.

The Cisco IOS status of link-down interface is shown as protocol down/link down. Neither the **show controller** command nor the **show interface** command reveals the difference between a local loss on the link and an LFP loss.

After LFP is configured, you can monitor the LFP status of each master or slave link using the **show link-fault** command. Use this command to determine whether LFP caused the link down on a slave interface. [Example 17-6](#) illustrates the output from this command on a slave interface.

Example 17-6 Monitor and Verify LFP

```
Router# show link-fault
Link Fault Propagation Configuration:
-----
LFP Config Mode   : LFP_SLAVE
LFP Master State  : LFP_STATUS_DOWN
Interfaces configured for LFP:
    FastEthernet0 (down)
```

RPR Keep Alive

The keep alive mechanism for RPR POS interfaces sends keep alive packets onto SPR links connecting adjacent nodes. This mechanism protects against failures undetected by the SONET/SDH layer. Keep alive is off by default.

With this feature enabled, the RPR POS port wraps when it fails to receive three consecutive keep-alives. When a link is down due to an interruption in keep-alive reception, it raises a critical LINK-KEEPALIVE alarm on SONET/SDH. The RPR POS port unwraps and the alarm clears only after receiving ten consecutive keep-alive packets. Keep alive failures are not dependant on CRC errors. Keep-alive is also supported on DRPRI for POS interfaces. It is not supported on Gigabit EtherChannel (GEC).

Keep alive detection takes more than 50ms. This time is added to the standard under 50 ms switching time of SONET/SDH to result in a total recovery time of greater than 50 ms.

Configuring RPR Keep Alive



**Caution**

When enabling the keep alive feature on an RPR carrying traffic, it is highly recommended that users first set the underlying POS circuit to OOS,DSBLD, which will wrap the RPR in the standard sub-50 msec. They can then enable the keep alive and put the circuit back to IS state. This ensures that traffic experiences a sub-50 msec hit when RPR keep alives are enabled.

**Note**

The RPR keep alive requires the spr or mpls working microcode image for the ML-Series card.

To enable and configure the RPR keep alives, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router# interface pos 0	Activates interface configuration mode to configure the POS interface.
Step 2	Router(config-if)# [no] spr keepalive	Enables RPR keep alives on the POS interface. The no form of this command disables the RPR keep alives. You must shut down both ends of the link before disabling keep alives. The keepalive timer is not configurable in proprietary RPR it is by default set to 5 msec  Caution It is strongly recommended that keep alives are enabled on RPR.
Step 3	Router# interface pos 1	Activates interface configuration mode to configure the POS interface.
Step 4	Router(config-if)# [no] spr keepalive	Enables RPR keep alives on the POS interface. The no form of this command disables the RPR keep alives. You must shut down both ends of the link before disabling keep alives.  Caution It is strongly recommended that keep alives are enabled on RPR.
Step 5	Router(config-if)# no shutdown	Enables the interface by preventing it from shutting down.
Step 6	Router(config)# end	Returns to privileged EXEC mode.
Step 7	Router# copy running-config startup-config	(Optional) Saves configuration changes to the TCC2/TCC2P flash database.

Monitoring and Verifying RPR Keep Alives

After RPR keep alives are configured, you can monitor their status using the global command **show interface spr 1** and the global command **show ons spr keepalive-info pos <0-1>**. [Example 17-7](#) and [Example 17-8](#) illustrate the output of these commands.

Example 17-7 Monitor and Verify RPR keep alives

```
Router> show interface spr 1>
SPR1 is down, line protocol is down
  Hardware is POS-SPR, address is 0005.9a3b.c140 (bia 0000.0000.0000)
  MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec,
  reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation: Cisco-EoS-LEX, loopback not set
  Keepalive not set
  Unknown duplex, Unknown Speed, unknown media type
```

```

ARP type: ARPA, ARP Timeout 04:00:00
SPR Wrapped information:
  POS0 : SONET
  POS1 : SONET KEEPALIVE
No. of active members in this SPR interface: 0
Last input never, output never, output hang never
Last clearing of "show interface" counters never
Input queue: 0/0/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
Output queue: 0/0 (size/max)
5 minute input rate 0 bits/sec, 0 packets/sec

5 minute output rate 0 bits/sec, 0 packets/sec
0 packets input, 0 bytes
Received 0 broadcasts (0 IP multicast)

0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
0 watchdog, 0 multicast
0 input packets with dribble condition detected

0 packets output, 0 bytes, 0 underruns
0 output errors, 0 collisions, 0 interface resets
0 babbles, 0 late collision, 0 deferred
0 lost carrier, 0 no carrier
0 output buffer failures, 0 output buffers swapped out

```

Example 17-8 Monitor and Verify RPR keep alives

```

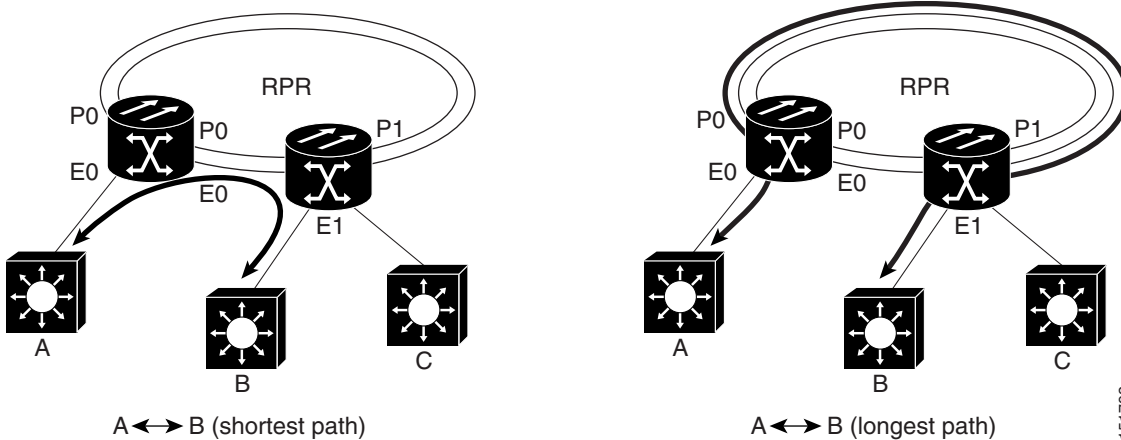
Router> show ons spr keepalive-info pos 1
Keep-alive is configured and operational
Keep-alive state is up
Interface State: UP           External Memory Location: 0xD
Num. KA pkts recvd: 461033198 Num KA pkts with KAF set: 930
StreamID: 79                 Src Node: 040
KA Dead Val: 3               KA Restore Val: 10
Curr FSM State: FULL         Prev FSM State: KAF COUNT
Prev FSM Event: RX KA        Wrap/Unwrap Event: STATE_UP
Defect Soak Count: 200      KA Fail Count: 0

```

RPR Shortest Path

The RPR shortest path feature determines the shorter hop-count of the two possible paths that can take traffic from location A to location B. [Figure 17-14](#) illustrates a shortest path and longest path for the same source and destination on an RPR. The shortest path from A to B on the RPR is counter-clockwise or east-to-west. The longest path is the opposite direction, clockwise or west-to-east along the RPR.

Figure 17-14 Shortest and Longest Path



By always using the shortest path, traffic between two nodes can achieve the lowest possible latency. This is especially important for delay-sensitive traffic such as VoIP or Video broadcast TV.

The ML-Series card implements shortest path through a hop-based topology discovery mechanism based on the IEEE 802.17 standard. Each RPR station bi-directionally floods local topology data around the RPR to its peers through a control frame. The topology state machine processes all the received control frames and builds a map based on the collective data. Layer 2 unicast traffic direction relies on this map.

The ML-Series card can also base its Layer 2 unicast traffic load-balancing on the shortest discovered path, although MAC address load-balancing is the default, and port-based load balancing is also an option.

The topology discovery mechanism also reroutes data during a protection event, such as a fiber cut. The station that detects the failure of an RPR link starts restoring the traffic with RPR wrapping. This causes partial traffic reorder and delivery through the reverse, sub-optimal path along the RPR to the destination. Once all the stations get updated with the new topology, which includes the link failure, they steer the traffic away from the failed span.

These are the guidelines for configuring RPR shortest path and topology discovery:

- You must configure `spr topology discovery-enable` on all the nodes on the RPR for accurate topology discovery.
- When topology discovery is configured, keep alives will automatically enable, if not already enabled.
- Disabling topology discovery does not disable keep alives, but a keep alive enabled warning message is displayed.
- DRPRI nodes do not support topology discovery.
- Shortest path load balancing requires the `spr` or `mpls` working microcode image for the ML-Series card. Attempting to enable shortest path load balancing with another microcode image, which doesn't support it, displays a warning message. The denied configuration is saved.
- If topology discovery is enabled, enable shortest path load balancing.
- You can enable shortest path load balancing on a per node basis. It is not dependent on the type of load balancing running on other nodes.

Configuring Shortest Path and Topology Discovery

To enable and configure shortest path and topology discovery, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router# interface spr1	Activates interface configuration mode to configure the POS interface.
Step 2	Router(config-if)# [no] spr topology discovery	Enables topology discovery on RPR.
Step 3	Router(config-if)# [no] spr load-balance shortest-hop	Configures shortest path load-balancing for L2 unicast packets being added to the ring.
Step 4	Router(config-if)# no shutdown	Enables the interface by preventing it from shutting down.
Step 5	Router(config)# end	Returns to privileged EXEC mode.
Step 6	Router# copy running-config startup-config	(Optional) Saves configuration changes to the TCC2/TCC2P flash database.

Monitoring and Verifying Topology Discovery and Shortest Path Load Balancing

After configuring topology discovery, you can monitor the status using the global command **show spr topology 1** to display the RPR topology information. [Example 17-7](#) and [Example 17-8](#) illustrate the output of these commands.

```
Router> show spr topology 1
***** ML-RPR Topology Map *****
Local Station Topology Info
Node Id : 40
West Span neighbor : 0
East Span neighbor : 0
Ring Topology: OPEN (UNSTABLE)
Advertised Protection requests:
ringlet0: IDLEringlet1: IDLE
Active Edges:
  ringlet0: NO ringlet1: NO
  Sequence Number: 0
=====
  East Interface: POS1   West Interface: POS0
  Number of nodes on east: 2
  Number of nodes on west: 2
  Active Topology Defects:
1. Topology Instability
Hops (POS 0)      Node Id      Edge W/E      Request W/E
    4              20          NO/NO         IDLE/IDLE
    6              40          NO/NO         IDLE/IDLE
Hops (POS 1)      Node Id      Edge W/E      Request W/E
    2              20          NO/NO         IDLE/IDLE
    6              40          NO/NO         IDLE/IDLE
```

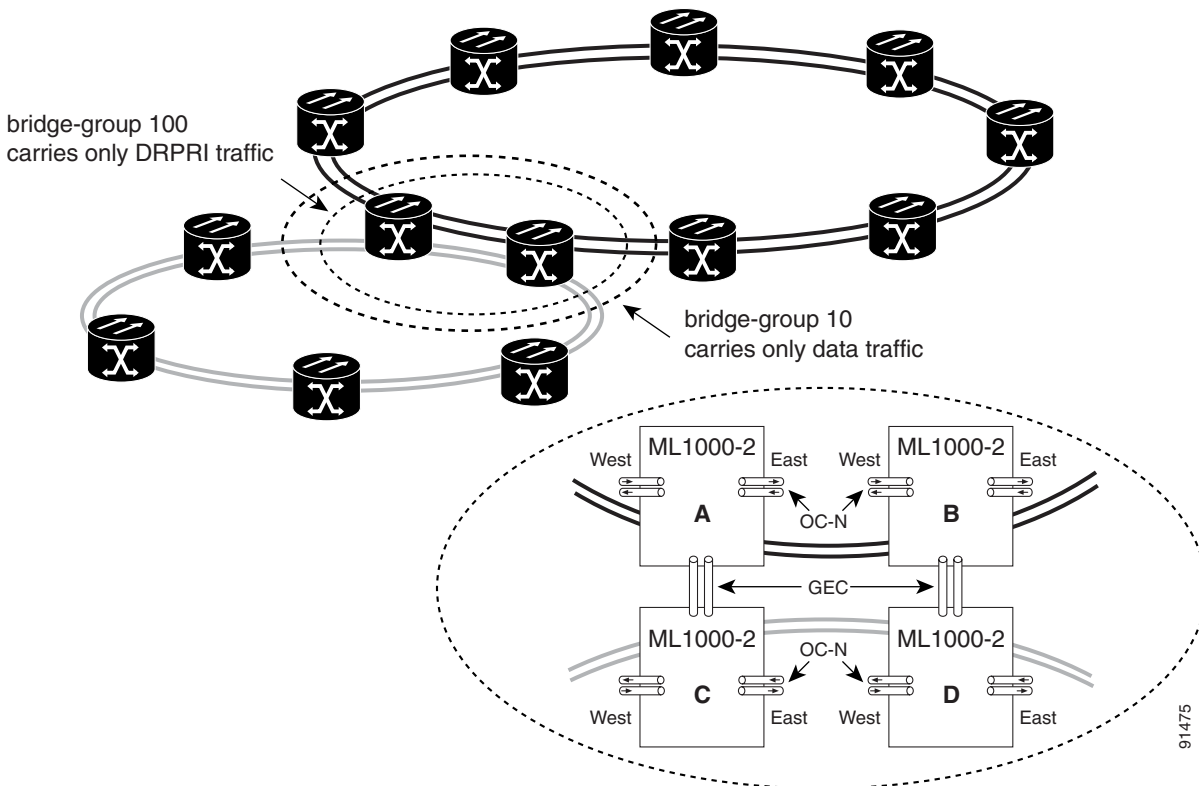
Understanding Dual RPR Interconnect

Cisco ML-Series RPR includes the bridge-group protocol DRPRI, which is a mechanism to interconnect rings for protection from node failure. DRPRI supports redundant pairs of back-to-back Ethernet connections between different RPR networks. One connection is the active node and the other is the standby node. During a failure of the active node, link, or card, a proprietary algorithm detects the failure and causes a switchover to the standby node.

DRPRI provides a recovery time of less than 200 ms for Layer 2 bridged traffic when the ML-Series card employs the enhanced microcode image. When the ML-Series card employs the spr or Multiprotocol Label Switching (MPLS) microcode image, the recovery time for Layer 2 bridged traffic is up to 12 seconds. With any microcode image, the recovery time for Layer 3 unicast and multicast traffic also depends on the convergence time of the routing protocol implemented. Customer routing protocols and spanning tree instances are not touched, regardless of DRPRI hops.

The paired ML1000-2 cards share the same station ID and are viewed by other members of the RPR as a single card. In Figure 17-15, paired cards A and B have the same SPR station ID, and paired cards C and D have the same station ID. The interconnected nodes do not need to be adjacent on the RPR. Bridging, IP routing, policing, and bandwidth allocations can still be provisioned on DRPRI ML1000-2 cards. Bridge group 100 in the example carries the DRPRI traffic. Bridge group 10 in the example carries data traffic.

Figure 17-15 Dual RPR Interconnect Network and Paired Cards



DRPRI has these characteristics:

- The DRPRI bridge-group cannot also be used to carry data traffic.


- The DRPRI bridge-group is limited to one protocol, so a bridge-group with DRPRI implemented cannot also implement RSTP or STP.
- Four ML1000-2 cards are required.
- All four ML1000-2 cards must be part of the same bridge-group (VLAN).
- Each paired set of ML1000-2 cards must have the same SPR station ID.
- The bridge-group must be configured on SPR subinterfaces.
- On each of the four ML1000-2 cards, both Gigabit Ethernet ports must be joined in a Gigabit EtherChannel (GEC) and the GEC interface must be included in the DRPRI bridge-group. Alternatively, one Gigabit Ethernet port must be shut down and the other one must be included in the DRPRI bridge-group. We recommend the GEC method.
- A manual shutdown on subinterfaces or the GEC interface included in the DRPRI bridge-group must be issued on the interfaces at both ends of the GEC or Ethernet connection between the rings.
- A DRPRI node can only be used for interconnecting two RPRs. The front ports of the cards should not be used to carry other traffic.
- Non-DRPRI bridge-groups carrying traffic between rings should not have STP or RSTP configured.
- Non-DRPRI bridge-groups carrying traffic between rings must be configured on each of the four ML-Series cards.
- 802.1 Q tunnels (QinQ) and protocol tunnels cannot be started on DRPRI nodes, but DRPRI nodes can bridge QinQ and protocol tunnels across the connected rings.
- Users should not change the pathcost of members of the DRPRI bridge-group. The pathcost is assigned by the ML-Series card to ensure proper operation of DRPRI. A user-configured pathcost is overwritten by the assigned default DRPRI pathcost.

Configuring DRPRI

DRPRI requires two pairs of ML-Series cards with one pair configured as an RPR and belonging to the first of two adjacent RPRs, and the second pair configured as an RPR and belonging to the second RPR (Figure 17-15 on page 17-36). DRPRI is configured on each of the four ML1000-2 cards that connect the two adjacent RPRs. The process of configuring DRPRI consists of the following major steps, which are detailed in the Cisco IOS procedure:

-
- Step 1** Configure a bridge-group with the DRPRI protocol.
 - Step 2** Configure the SPR interface.
 - a. Assign a station ID number.
 - b. Assign a DRPRI ID of 0 or 1.
 - Step 3** Create an SPR subinterface and assign the bridge-group to the subinterface.
 - Step 4** Create a GEC interface.
 - Step 5** Create a GEC subinterface and assign the bridge-group to the subinterface.

To enable and configure DRPRI, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# bridge crb	Concurrent routing and bridging is enabled. When concurrent routing and bridging has been enabled, the default behavior is to bridge all protocols that are not explicitly routed in a bridge group.
Step 2	Router(config)# bridge <i>bridge-group-number</i> protocol drpri-rstp	Creates the bridge-group number shared by the four ML1000-2 cards and assigns the protocol for DRPRI to the bridge-group. The same command using the same bridge group number must be given on each of the four cards.  Caution Do not use the DRPRI bridge group to carry data traffic. Data traffic on the DRPRI bridge group can cause instability and traffic hits.
Step 3	Router(config)# interface spr 1	Creates the SPR interface for RPR or enters the SPR interface configuration mode on a previously created SPR interface. The only valid SPR number is 1.
Step 4	Router(config-if)# spr station-ID <i>station-ID-number</i>	Configures a station identification number. The user must configure the same station ID on both the paired cards. Valid station ID numbers range from 1 to 254.
Step 5	Router(config-if)# spr drpri-ID {0 1}	Creates a DRPRI identification number of 0 or 1 to differentiate between the ML1000-2 cards paired for DRPRI. A DRPRI identification number of 0 is the default.
Step 6	Router(config-if)# interface spr <i>shared-packet-ring-subinterface-number</i>	Creates the SPR subinterface.
Step 7	Router(config-subif)# encapsulation dot1q <i>vlan-ID</i>	Sets the SPR subinterface encapsulation to IEEE 802.1Q.
Step 8	Router(config-subif)# bridge-group <i>bridge-group-number</i>	Assigns the SPR subinterface to the DRPRI bridge-group.
Step 9	Router(config)# interface port-channel <i>channel-number</i>	Creates the GEC interface or channel-group.
Step 10	Router(config-if)# interface Gigabit Ethernet <i>number</i>	Enters interface configuration mode for the first Gigabit Ethernet interface that you want to assign to the GEC subinterface.
Step 11	Router(config-if)# channel-group <i>channel-number</i>	Assigns the Gigabit Ethernet interfaces to the GEC. The channel number must be the same channel number that you assigned to the EtherChannel interface.

	Command	Purpose
Step 12	Router(config-if)# interface Gigabit Ethernet <i>number</i>	Enters interface configuration mode for the second Gigabit Ethernet interface that you want to assign to the GEC subinterface.
Step 13	Router(config-if)# channel-group <i>channel-number</i>	Assigns the Gigabit Ethernet interfaces to the GEC. The channel number must be the same channel number that you assigned to the EtherChannel interface.
Step 14	Router(config-subif)# interface port-channel <i>channel-sub-interface-number</i>	Creates the GEC subinterface.
Step 15	Router(config-subif)# encapsulation dot1q <i>vlan-ID</i>	Sets subinterface encapsulation to IEEE 802.1Q. The VLAN ID used should be the same VLAN ID used in Step 7.
Step 16	Router(config-subif)# bridge-group <i>bridge-group-number</i>	Assigns the GEC subinterface to the DRPRI bridge-group.
Step 17	Router(config-if)# end	Exits to privileged EXEC mode.
Step 18	Router# copy running-config startup-config	(Optional) Saves configuration changes to NVRAM.

DRPRI IOS Configuration Example

Figure 17-15 on page 17-36 shows an example of RPR configuration. The output from the **show run** command is provided in Examples 17-9, 17-10, 17-11, and 17-12.



Caution

Do not use the DRPRI bridge group to carry data traffic. Data traffic on the DRPRI bridge group can cause instability and traffic hits.



Note

To differentiate between the ML1000-2 cards paired for DRPRI, the cards have a DRPRI identification number of either 0 or 1. A **show run** command on a card with a DRPRI ID of 1 will display `spr drpri-ID 1` in the Cisco IOS CLI output. But a **show run** command on a card with a DRPRI ID of 0 will not display any DRPRI ID in the Cisco IOS CLI output.

Example 17-9 ML-Series Card A Configuration

```
hostname ML-Series A
bridge crb
bridge 100 protocol drpri-rstp
bridge 100 forward-time 4
!
!
interface SPR1
no ip address
no keepalive
spr station-id 1
spr drpri-id 0
hold-queue 150 in
!
interface SPR1.1
encapsulation dot1Q 100
```

```

bridge-group 100
!
interface SPR1.10
encapsulation dot1Q 10
bridge-group 10
bridge-group 10 spanning-disabled
!
interface Port-channel1
no ip address
hold-queue 150 in
!
interface Port-channel1.1
encapsulation dot1Q 100
bridge-group 100
bridge-group 100 path-cost 32000
!
interface Port-channel1.10
encapsulation dot1Q 10
bridge-group 10
bridge-group 10 spanning-disabled
!
interface GigabitEthernet0
no ip address
channel-group 1
!
interface GigabitEthernet1
no ip address
channel-group 1
!
interface POS0
no ip address
spr interface-id 1
crc 32
!
interface POS1
no ip address
spr interface-id 1
crc 32
!
ip classless
no ip http server

```

Example 17-10 ML-Series Card B Configuration

```

hostname ML-Series B
nodeB_ML1000#
bridge crb
bridge 100 protocol drpri-rstp
bridge 100 forward-time 4
!
!
interface SPR1
no ip address
no keepalive
spr station-id 1
spr drpri-id 1
hold-queue 150 in
!
interface SPR1.1
encapsulation dot1Q 100
bridge-group 100
!
interface SPR1.10

```

```

encapsulation dot1Q 10
bridge-group 10
bridge-group 10 spanning-disabled
!
interface Port-channel1
no ip address
hold-queue 150 in
!
interface Port-channel1.1
encapsulation dot1Q 100
bridge-group 100
bridge-group 100 path-cost 32000
!
interface Port-channel1.10
encapsulation dot1Q 10
bridge-group 10
bridge-group 10 spanning-disabled
!
interface GigabitEthernet0
no ip address
channel-group 1
!
interface GigabitEthernet1
no ip address
channel-group 1
!
interface POS0
no ip address
spr interface-id 1
crc 32
!
interface POS1
no ip address
spr interface-id 1
crc 32
!
ip classless
no ip http server

```

Example 17-11 ML-Series Card C Configuration

```

hostname ML-Series C
bridge crb
bridge 100 protocol drpri-rstp
bridge 100 forward-time 4
bridge 100 priority 0
!
!
interface SPR1
no ip address
no keepalive
spr station-id 2
spr drpri-id 0
hold-queue 150 in
!
interface SPR1.1
encapsulation dot1Q 100
bridge-group 100
!
interface SPR1.10
encapsulation dot1Q 10
bridge-group 10
bridge-group 10 spanning-disabled

```

```

!
interface Port-channel1
no ip address
hold-queue 150 in
!
interface Port-channel1.1
encapsulation dot1Q 100
bridge-group 100
bridge-group 100 path-cost 32000
!
interface Port-channel1.10
encapsulation dot1Q 10
bridge-group 10
bridge-group 10 spanning-disabled
!
interface GigabitEthernet0
no ip address
channel-group 1
!
interface GigabitEthernet1
no ip address
channel-group 1
!
interface POS0
no ip address
spr interface-id 1
crc 32
!
interface POS1
no ip address
spr interface-id 1
crc 32
!
ip classless
no ip http server

```

Example 17-12 ML-Series Card D Configuration

```

hostname ML-Series D
bridge crb
bridge 100 protocol drpri-rstp
bridge 100 forward-time 4
!
!
interface SPR1
no ip address
no keepalive
spr station-id 2
spr drpri-id 1
hold-queue 150 in
!
interface SPR1.1
encapsulation dot1Q 100
bridge-group 100
!
interface SPR1.10
encapsulation dot1Q 10
bridge-group 10
bridge-group 10 spanning-disabled
!
interface Port-channel1
no ip address
hold-queue 150 in

```

```
!  
interface Port-channel1.1  
encapsulation dot1Q 100  
bridge-group 100  
bridge-group 100 path-cost 65535  
!  
interface Port-channel1.10  
encapsulation dot1Q 10  
bridge-group 10  
bridge-group 10 spanning-disabled  
!  
interface GigabitEthernet0  
no ip address  
channel-group 1  
!  
interface GigabitEthernet1  
no ip address  
channel-group 1  
!  
interface POS0  
no ip address  
spr interface-id 1  
crc 32  
!  
interface POS1  
no ip address  
spr interface-id 1  
crc 32  
!  
ip classless  
no ip http server
```

Monitoring and Verifying DRPRI

After DRPRI is configured, you can monitor its status by using the **show bridge verbose** command (Example 17-13).

Example 17-13 *show bridge verbose Command*

```
Router# show bridge bridge-group-number verbose
```




Configuring Ethernet over MPLS

This chapter describes how to configure Ethernet over Multiprotocol Label Switching (EoMPLS) on the ML-Series card.

This chapter includes the following major sections:

- [Understanding EoMPLS, page 18-1](#)
- [Configuring EoMPLS, page 18-4](#)
- [EoMPLS Configuration Example, page 18-11](#)
- [Monitoring and Verifying EoMPLS, page 18-13](#)
- [Understanding MPLS-TE, page 18-14](#)
- [Configuring MPLS-TE, page 18-15](#)
- [MPLS-TE Configuration Example, page 18-17](#)
- [Monitoring and Verifying MPLS-TE and IP RSVP, page 18-19](#)

Understanding EoMPLS

EoMPLS provides a tunneling mechanism for transporting Layer 2 Ethernet frames through an MPLS-enabled Layer 3 core network. It encapsulates Ethernet protocol data units (PDUs) inside MPLS packets and using label stacking forwards them across the MPLS network. EoMPLS is an Internet Engineering Task Force (IETF) standard-track protocol based on the Martini draft, specifically the draft-martini-l2circuit-encap-mpls-01 and draft-martini-l2circuit-transport-mpls-05 sections.

EoMPLS allows service providers to offer customers a virtual Ethernet line service or VLAN service using the service provider's existing MPLS backbone. It also simplifies service provider provisioning, since the provider edge customer-leading edge (PE-CLE) equipment only needs to provide Layer 2 connectivity to the connected customer edge (CE) equipment.

[Figure 18-1](#) shows an example of EoMPLS implemented on a service provider network. In the example, the ML-Series card acts as PE-CLE equipment connecting to the Cisco GSR 12000 Series through an RPR access ring. Point-to-point service is provided to CE equipment in different sites that connect through ML-Series cards to the ML-Series card RPR access ring.

- SPR using HDLC-Framing
- MPLS tag switching is supported on SPR and POS interfaces.
- Class of service (CoS) values are mapped to the experimental (EXP) bits in the MPLS label, either statically or by using the IEEE 802.1p bits (default).
- Ingress PE-CLE ML-Series cards set the S bit of the VC label to 1 to indicate that the VC label is at the bottom of the stack.
- The Ethernet over MPLS feature is part of the Cisco Any Transport over MPLS (AToM) product set.
- The ML-Series card hosting the EoMPLS endpoint ports must be running the MPLS microcode image to support EoMPLS. For more information on multiple microcode images, see the “[Multiple Microcode Images](#)” section on page 3-11. Other ML-Series cards in the RPR are not restricted to the MPLS microcode image.

EoMPLS Restrictions

EoMPLS on the ML-Series card has the following restrictions:

- Zero hop or hairpin VCs are not supported. A single ML-Series card cannot be both the source and destination for a VC.
- MPLS control word for sequencing of data transmission is not supported. Packets must be received and transmitted without control word.
- Sequence checking or resequencing of EoMPLS traffic is not supported. Both depend on the control word to function.
- Maximum transmission unit (MTU) fragmentation is not supported.
- Explicit-null label for back-to-back LDP sessions is not supported.
- The ML-Series card supports a maximum of 650 VCs in EoMPLS configuration.



Caution

Since MTU fragmentation is not supported across the MPLS backbone, the network operator must make sure the MTU of all intermediate links between endpoints is sufficient to carry the largest Layer 2 PDU.



Caution

If the 650 VC maximum is exceeded, the user is not permitted to save the configuration file.

EoMPLS Quality of Service

The EXP is a 3-bit field and part of the MPLS header. It was created by the IETF on an experimental basis, but later became part of the standard MPLS header. The EXP bits in the MPLS header carry the packet priority. Each label switch router along the path honors the packet priority by queuing the packet into the proper queue and servicing the packet accordingly.

By default, the ML-Series card does not map the IEEE 802.1p bits in the VLAN tag header to the MPLS EXP bits. The MPLS EXP bits are set to a value of 0.

There is no straight copy between Layer 2 CoS and MPLS EXP, but the user can use the **set mpls experimental** action to set the MPLS EXP bit values based on a match to 802.1p bits. This mapping occurs at the entry point, the ingress of the network.

Quality of service (QoS) for EoMPLS traffic on ML-Series cards uses strict priority and/or weighted round robin scheduling in the egress interface of both imposition and disposition router. This requires selection of the service class queue that determines the type of scheduling. In the imposition router, the priority bits EXP or RPR CoS that are marked based on policing are used to select the service class queue and in the disposition router, the dot1p CoS bits (which are copied from EXP bits of the labels) are used to do the same. In addition to scheduling in the egress interface, the output policy action can also include remarking of EXP and RPR CoS bits.

EoMPLS on the ML-Series card uses the Cisco Modular Quality of Service Command-Line Interface (MQC), just like the standard QoS on the ML-Series card. But the full range of MQC commands are not available. [Table 18-1](#) lists the applicable MQC statements and actions for the ML-Series card interfaces.

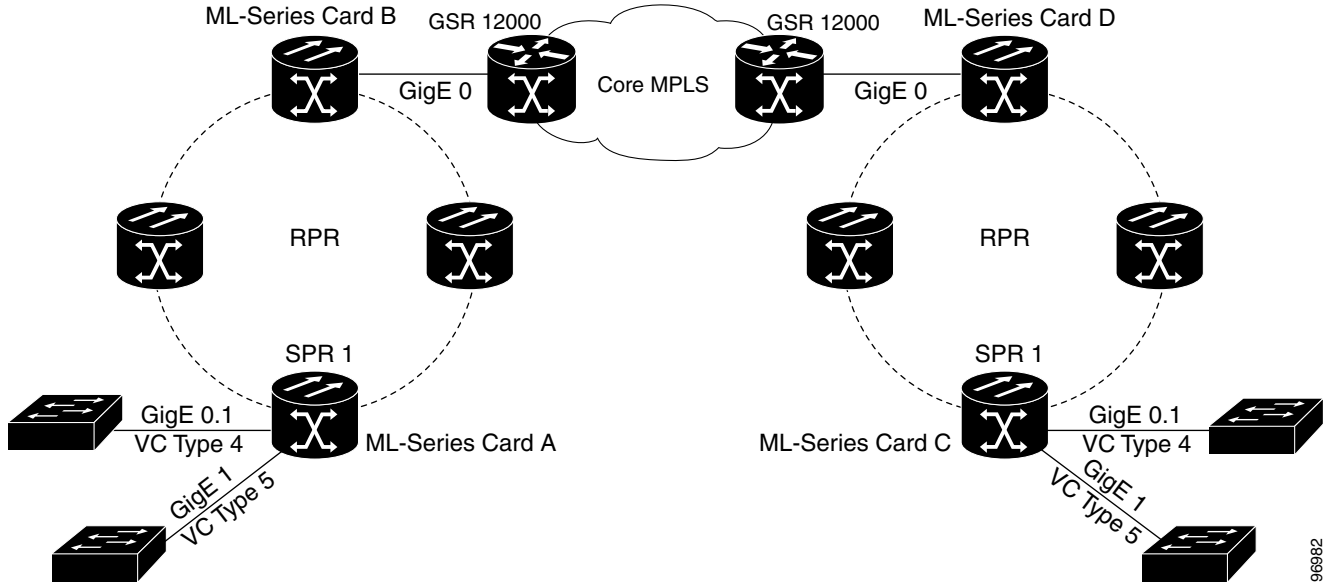
Table 18-1 Applicable EoMPLS QoS Statements and Actions

Interface	Applicable MQC Match Statements	Applicable MQC Actions
Imposition Ingress	match cos match ip precedence match ip dscp match vlan	police <i>cir</i> <i>cir-burst</i> [<i>pir-burst</i> pir <i>pir</i> conform [<i>set-mpls-exp</i> exceed [<i>set-mpls-exp</i>] violate <i>set-mpls-exp</i>]
Imposition Egress	match mpls exp	bandwidth { <i>bandwidth-kbps</i> percent <i>percent</i> } and priority <i>kbps</i> and [<i>set-mpls-exp</i>]
Disposition Ingress	Not applicable	Not applicable
Disposition Egress	match mpls exp	bandwidth { <i>bandwidth-kbps</i> percent <i>percent</i> } and priority <i>kbps</i> and set-cos <i>cos-value</i>

Configuring EoMPLS

Implementing EoMPLS on a service provider network requires ML-Series card interfaces to play three major roles. [Figure 18-2](#) illustrates EoMPLS on ML-Series cards and the interfaces that play these major roles.

Figure 18-2 EoMPLS Configuration Example



- ML-Series card interfaces connect the provider's network directly to the customer edge equipment and are known as the PE-CLE interfaces. This PE-CLE interface is configured to be an endpoint on the EoMPLS point-to-point session. The port on the ML-Series card connecting directly to the CE port is either Fast Ethernet on an ML100T-12 or Gigabit Ethernet on an ML1000-2.

The PE-CLE interface is either a VC type 4 or VC type 5. A VC type 4, which transport IEEE 802.1Q VLAN packets between two PE-CLE ML-Series cards, is configured on a subinterface, such as interface GigE 0.1 on card A and card C in Figure 18-2. A VC type 5, which transports the configured port's packets between two PE-CLE ML-Series cards, is configured on the main Fast Ethernet or Gigabit Ethernet interface. Interface GigE 1 on card A and card C plays the VC type 5 role in Figure 18-2.

For more information on the role of a VC type 4, see the “Understanding EoMPLS” section on page 18-1. For more information on the role of a VC type 5, see the “Understanding EoMPLS” section on page 18-1.

- An ML-Series card interface bridges the PE-CLE interface and the RPR network of ML-Series cards. This RPR/SPR interface contains POS ports and is configured for MPLS IP. To enable the RPR to act as an access ring for the MPLS cloud, you must provision the SPR interface on the same ML-Series card that hosts the EoMPLS PE-CLE FastEthernet or GigabitEthernet interfaces. Interface SPR 1 on card A and card C plays this role in Figure 18-2.

A POS interface is used in this role if the ML-Series cards on one side of the MPLS cloud use a point-to-point connection through POS interfaces instead of an RPR to connect. You provision the POS interface, instead of an SPR interface, to bridge the PE-CLE interface on the ML-Series card.

- An ML-Series card interface connects to the interface of a router that is part of the MPLS cloud. This interface is Gigabit Ethernet or Fast Ethernet and connects to the port of a Cisco GSR 12000 Series or similar device that is on the MPLS network. This MPLS cloud-facing interface bridges the SPR interface and the MPLS cloud. A bridge group must be created that contains this FastEthernet or GigabitEthernet port and the SPR interface. Interface GigE 0 on card B and card D plays this role in Figure 18-2.

A POS interface is used in the SPR interface role if the ML-Series cards on one side of the MPLS cloud use a point-to-point connection through POS interfaces instead of an RPR to connect. You provision the POS interface, instead of the SPR interface, as a member of the bridge group with MPLS cloud-facing interface on the ML-Series card.

Peer ML-Series card interface roles must be configured on both sides of the EoMPLS point-to-point service crossing the MPLS core.

EoMPLS Configuration Guidelines

The ML-Series peer cards on both endpoints of the EoMPLS point-to-point service must be configured. Perform the following configuration tasks to enable EoMPLS:


- [VC Type 4 Configuration on PE-CLE Port, page 18-6](#) (Either VC type 4 or VC type 5 is required.)
- [VC Type 5 Configuration on PE-CLE Port, page 18-7](#) (Either VC type 4 or VC type 5 is required.)
- [EoMPLS Configuration on PE-CLE SPR Interface, page 18-8](#) (Optional)
- [EoMPLS Configuration on PE-CLE POS Interface, page 18-8](#) (Optional)
- [Bridge Group Configuration on MPLS Cloud-facing Port and SPR Interface, page 18-9](#) (Optional)
- [Bridge Group Configuration on MPLS Cloud-facing Port and POS Interface, page 18-10](#) (Optional)
- [Setting the Priority of Packets with the EXP, page 18-10](#) (Optional)

These are the guidelines for configuring EoMPLS:

- Loopback addresses are used to specify the peer ML-Series card's IP address.
- LDP configuration is required. The default Tag Distribution Protocol (TDP) will not work.
- EoMPLS uses LDP targeted session between the ML-Series cards to create the EoMPLS VCs.
- The MPLS backbone must use an Interior Gateway Protocol (IGP) routing protocol, for example, Open Shortest Path First (OSPF).

VC Type 4 Configuration on PE-CLE Port


To provision a VC type 4, perform the following procedure on the CE facing port, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# mpls label protocol ldp	Specifies LDP as the label distribution protocol. LDP must be specified. The ML-Series card does not operate EoMPLS with the default TDP as the label distribution protocol.
Step 2	Router(config)# mpls label range 16 1039	Restricts the MPLS label range to correspond to the 1024 subinterfaces supported on the ML-Series card.  Caution Failure to restrict the MPLS label range allows the creation of MPLS labels outside the supported range. In certain circumstances, this may result in traffic loss.

	Command	Purpose
Step 3	Router(config)# interface loopback0	Enters loopback interface configuration mode.
Step 4	Router(config-if)# ip address ip-address 255.255.255.255	Assigns an IP address to the loopback interface. This loopback IP addresses is used to identify the peer in the EoMPLS point-to-point session.
Step 5	Router(config)# interface {GigabitEthernet FastEthernet} interface-number.sub-interface-number	Specifies the Ethernet subinterface for the imposition interface. Make sure the subinterface on the adjoining CE equipment is on the same VLAN as this subinterface.
Step 6	Router(config-subif)# no ip address	Disables the IP address if an IP address is assigned.
Step 7	Router(config-subif)# encapsulation dot1Q vlan-id	Enables the subinterface to accept 802.1q VLAN packets. Make sure the VLAN ID is the same as the VLAN ID on the adjoining CE equipment.
Step 8	Router(config-subif)# mpls l2transport route destination vc-id	<p>Specifies the VC to use to transport the VLAN packets. Initiates a remote LDP session with the peer point-to-point endpoint interface.</p> <p>The argument destination specifies the loopback address of the remote ML-Series at the other end of the VC (PE-CLE).</p> <p>The argument vc-id is a value you supply. It must be unique for each VC. The VC ID is used to connect the endpoints of the VC. Specify the same VC ID on both ends of the VC.</p>

VC Type 5 Configuration on PE-CLE Port

To provision a VC type 5, perform the following procedure on the customer facing port, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# mpls label protocol ldp	Specifies LDP as the label distribution protocol. LDP must be specified. The ML-Series card does not operate EoMPLS with the default TDP as the label distribution protocol.
Step 2	Router(config)# mpls label range 16 1039	<p>Restricts the MPLS label range to correspond to the 1024 subinterfaces supported on the ML-Series card.</p> <p> Caution Failure to restrict the MPLS label range allows the creation of MPLS labels outside the supported range. In certain circumstances this may result in traffic loss.</p>
Step 3	Router(config)# interface loopback0	Enters loopback interface configuration mode.
Step 4	Router(config-if)# ip address ip-address 255.255.255.255	Assigns an IP address to the loopback interface. This loopback IP addresses is used to identify the peer in the EoMPLS point-to-point session.

	Command	Purpose
Step 5	Router(config)# interface {GigabitEthernet FastEthernet} <i>interface-number</i>	Specifies the Ethernet interface for the imposition interface.
Step 6	Router(config-if)# no ip address	Disables the IP address if an IP address is assigned.
Step 7	Router(config-if)# mpls l2transport route destination vc-id	Specifies the VC to use to transport the VLAN packets. Initiates a remote LDP session with the peer point-to-point endpoint interface. destination specifies the loopback address of the remote ML-Series card at the other end of the VC (PE-CLE). vc-id is a common identifier used by the endpoints to identify the created EoMPLS VC. It must be unique for each VC and the same VC ID must be used on both ends of the VC.

EoMPLS Configuration on PE-CLE SPR Interface

To provision the SPR interface for MPLS, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# mpls label protocol ldp	Specifies LDP as the label distribution protocol. LDP must be specified. The ML-Series card does not operate EoMPLS with the default TDP as the label distribution protocol.
Step 2	Router(config)# interface spr 1	Enters RPR interface configuration mode.
Step 3	Router(config-if)# ip address ip-address mask	Assigns an IP address to the RPR interface for MPLS.
Step 4	Router(config-if)# mpls ip	Implements tag switching on the SPR interface.
Step 5	Router(config-if)# end	Exits interface configuration mode.
Step 6	Router# copy running-config startup-config	Saves the running configuration file to the startup configuration file.

EoMPLS Configuration on PE-CLE POS Interface

You can also configure a POS interface on the same ML-Series card that hosts the EoMPLS PE-CLE Fast Ethernet or Gigabit Ethernet interfaces. To provision the POS interface for MPLS, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# mpls label protocol ldp	Specifies LDP as the label distribution protocol. LDP must be specified. The ML-Series card does not operate EoMPLS with the default TDP as the label distribution protocol.
Step 2	Router(config)# interface pos0	Enters POS interface configuration mode.
Step 3	Router(config-if)# ip address ip-address mask	Assigns an IP address to the RPR interface for MPLS.
Step 4	Router(config-if)# mpls ip	Implements tag switching on the SPR interface.
Step 5	Router(config-if)# end	Exits interface configuration mode.
Step 6	Router# copy running-config startup-config	Saves the running configuration file to the startup configuration file.

Bridge Group Configuration on MPLS Cloud-facing Port and SPR Interface

To provision the MPLS cloud-facing port for EoMPLS, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# bridge 1 protocol {rstp ieee}	(Optional) Assigns a bridge group number and defines the appropriate spanning-tree type: either IEEE 802.1D Spanning Tree Protocol or IEEE 802.1W Rapid Spanning Tree.
Step 2	Router(config)# interface {GigabitEthernet FastEthernet} interface-number	Enters interface configuration mode to configure the MPLS cloud-facing FastEthernet or GigabitEthernet interface of the ML-Series card.
Step 3	Router(config-if)# bridge-group 1	Assigns the network interface to bridge group 1.
Step 4	Router(config-if)# no shutdown	Changes the shutdown state to up and enables the interface.
Step 5	Router(config)# interface spr 1.subinterface-number	Enters SPR subinterface mode.
Step 6	Router(config-subif)# encapsulation dot1q 1 native	Enables the IEEE 802.1Q native VLAN on the subinterface. EoMPLS on the ML-Series card uses the native VLAN (VLAN 1).
Step 7	Router(config-if)# bridge-group 1	Assigns the network interface to bridge group 1.
Step 8	Router(config-if)# end	Returns to privileged EXEC mode.
Step 9	Router# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Bridge Group Configuration on MPLS Cloud-facing Port and POS Interface

To provision the MPLS cloud-facing port for EoMPLS, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# bridge 1 protocol {rstp ieee}	(Optional) Assigns a bridge group number and defines the appropriate spanning-tree type: either IEEE 802.1D Spanning Tree Protocol or IEEE 802.1W Rapid Spanning Tree.
Step 2	Router(config)# interface {GigabitEthernet FastEthernet} interface-number	Enters interface configuration mode to configure the MPLS cloud-facing FastEthernet or GigabitEthernet interface of the ML-Series card.
Step 3	Router(config-if)# bridge-group 1	Assigns the network interface to bridge group 1.
Step 4	Router(config-if)# no shutdown	Changes the shutdown state to up and enables the interface.
Step 5	Router(config)# interface pos interface-number.subinterface-number	Enters POS subinterface mode.
Step 6	Router(config-subif)# encapsulation dot1q 1 native	Enables the IEEE 802.1Q native VLAN on the subinterface. EoMPLS on the ML-Series card uses the native VLAN (VLAN 1).
Step 7	Router(config-if)# bridge-group 1	Assigns the network interface to bridge group 1.
Step 8	Router(config-if)# end	Returns to privileged EXEC mode.
Step 9	Router# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Setting the Priority of Packets with the EXP

EoMPLS can provide QoS using the three EXP bits in a label to determine the priority of packets. Set the experimental bits in both the VC and tunnel labels to support QoS between ML-Series card endpoints.



Note

The ML-Series card also supports MPLS-TE. See [Understanding MPLS-TE, page 18-14](#) for more information.

To set the experimental bits, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# class-map class-name	Specifies the user-defined name of the traffic class.
Step 2	Router(config-cmap)# match any	Specifies that all packets will be matched.
Step 3	Router(config-cmap)# end	Returns to global configuration mode.

	Command	Purpose
Step 4	Router(config)# policy-map <i>policy-name</i>	Specifies the name of the traffic policy to configure.
Step 5	Router(config-pmap)# class <i>class-name</i>	Specifies the name of a predefined traffic class, which was configured with the class-map command, used to classify traffic to the traffic policy.
Step 6	Router (config-pmap-c)# set mpls experimental imposition <i>value</i>	Designates the value to which the MPLS bits are set if the packets match the specified policy map.
Step 7	Router(config)# interface GigabitEthernet <i>interface-number</i> or interface FastEthernet <i>interface-number</i>	Enters interface configuration mode.
Step 8	Router(config-if)# service-policy input <i>policy-name</i>	Attaches a traffic policy to an interface.

EoMPLS Configuration Example

Figure 18-2 on page 18-5 illustrates the sample network that the configuration commands reference. Examples 18-1, 18-2, 18-3, and 18-4 list relevant portions of the configuration files for enabling EoMPLS on ML-Series cards in a sample network.

Example 18-1 ML-Series Card A Configuration

```

microcode mpls
ip subnet-zero
no ip domain-lookup
!
mpls label protocol ldp
!
interface Loopback0

    ip address 10.10.10.10 255.255.255.255
    !
interface SPR1
    ip address 100.100.100.100 255.255.255.0
    no keepalive
spr station-id 1
mpls ip
    hold-queue 150 in
    !
interface GigabitEthernet0
    no ip address
    !
interface GigabitEthernet0.1
    encapsulation dot1Q 10
    mpls l2transport route 3.3.3.3 1
    !
interface GigabitEthernet1
    no ip address
    mpls l2transport route 4.4.4.4 2

```

```

!
interface POS0
  no ip address
  spr-intf-id 1
  crc 32
!
interface POS1
  no ip address
  spr-intf-id 1
  crc 32
router ospf 1
  log-adjacency-changes
  network 1.1.1.0 0.0.0.255 area 0
  network 10.10.10.0 0.0.0.255 area 0
!
ip classless
no ip http server

```

Example 18-2 ML-Series Card B Configuration

```

bridge 10 protocol ieee
!
!
interface SPR1
  no ip address
  no keepalive
  bridge-group 10
  hold-queue 150 in
!
interface GigabitEthernet0
  no ip address
  bridge-group 10

```

Example 18-3 ML-Series Card C Configuration

```

microcode mpls
ip subnet-zero
no ip domain-lookup
!
mpls label protocol ldp
!
interface Loopback0

  ip address 20.20.20.20 255.255.255.255
!
interface SPR1
  ip address 100.100.100.100 255.255.255.0
  no keepalive
  spr station-id 4
  mpls ip
  hold-queue 150 in
!
interface GigabitEthernet0
  no ip address
!
interface GigabitEthernet0.1
  encapsulation dot1Q 10
  mpls l2transport route 1.1.1.1 1
!
interface GigabitEthernet1
  no ip address
  mpls l2transport route 2.2.2.2 2

```

```

!
interface POS0
  no ip address
  spr-intf-id 1
  crc 32
!
interface POS1
  no ip address
  spr-intf-id 1
  crc 32
!
router ospf 1
  log-adjacency-changes
  network 1.1.1.0 0.0.0.255 area 0
  network 10.10.10.0 0.0.0.255 area 0
!
ip classless
no ip http server

```

Example 18-4 ML-Series Card D Configuration

```

bridge 20 protocol ieee
!
!
interface SPR1
  no ip address
  no keepalive
  bridge-group 20
  hold-queue 150 in
!
interface GigabitEthernet0
  no ip address
  bridge-group 20

```

Monitoring and Verifying EoMPLS

Table 18-2 shows the privileged EXEC commands for monitoring and verifying EoMPLS.

Table 18-2 Commands for Monitoring and Maintaining Tunneling

Command	Purpose
<code>show mpls l2transport vc</code>	Provides information about all EoMPLS tunnels.
<code>show mpls l2transport vc detail</code>	Provides detailed information about the EoMPLS tunnel.
<code>show mpls l2transport vc vc-id</code>	Provides information about a specific EoMPLS tunnel.

Understanding MPLS-TE

MPLS traffic is normally routed to the least cost path as calculated by OSPF or another IGP routing protocol. This routing gives little or no consideration to varying bandwidth demands or link loads. MPLS traffic engineering (MPLS-TE) overcomes this by mapping traffic flows to paths that take bandwidth demands into account. These paths are known as MPLS-TE tunnels, and they may deviate from the normal IGP calculated routes.

MPLS-TE (RFC 2702) allow service providers to create traffic engineered tunnels to reserve bandwidth for specific types of traffic and to provide point-to-point services for end customers. The ML-Series card supports a maximum of 24 MPLS-TE tunnels. MPLS-TE tunnels can carry a VC type 5, which tunnels an Ethernet port, or a VC type 4, which tunnels an 802.1Q VLAN.

For the ML-Series card to use MPLS-TE, you need to configure three main components. First, you must implement an IGP routing protocol that conveys and distributes information about the link resources throughout the MPLS network. For this purpose, the ML-Series card supports OSPF and OSPF-TE extensions (RFC 2328 and RFC 2370). MPLS-TE extensions for other routing protocols, such as IS-IS, are not supported on the ML-Series card.

Second, you need to configure a signalling protocol to reserve needed resources and establish LSPs across the MPLS network. MPLS-TE tunnels use Resource Reservation Protocol (RSVP) messages (RFC 2205 and RFC 3209) to accomplish this. The ML-Series card supports RSVP and the RSVP extensions for LSP tunnels on both POS interfaces and RPR (SPR) interfaces.

For the third component, you need to set up an MPLS-TE tunnel on the appropriate ML-Series card interface. This requires creating an MPLS tunnel interface with an IP address, destination, encapsulation, bandwidth, and explicit or dynamic path.

RSVP on the ML-Series Card

The ML-Series card uses RSVP to establish MPLS-TE tunnels and the associated tunnel labels. Targeted LDP is still used to establish the VC Labels. Also, RSVP is only used to guarantee the bandwidth on the intermediate nodes on the tunnel. On the ML-Series card, which will be the end-point of the MPLS-TE tunnel, RSVP is used only for bandwidth allocation.

You configure bandwidth guarantees on the ML-Series card ports using the Cisco Modular Quality of Service Command-Line Interface (MQC), just like the standard QoS on the ML-Series card. For more information, see the [“EoMPLS Quality of Service” section on page 18-3](#).

The ML-Series card does not use RSVP messages to carry the information for EoMPLS VCs. LDP sessions are still used to exchange VC information. Also RSVP does not guarantee bandwidth. It only allocates bandwidth.

The ML-Series card supports RSVP summary refresh and RSVP refresh reduction (RFC 2961). Refresh reduction is a set of extensions that reduce the messaging load imposed by RSVP. This helps RSVP scale to support larger numbers of flows. The global configuration command **ip rsvp signalling refresh reduction** enables this feature.

Ethernet FCS Preservation

You can configure the ML-Series card to encapsulate and preserve the customer's Ethernet FCS. The ML-Series card will carry the Ethernet FCS end-to-end and unmodified across EoMPLS or EoMPLS-TE tunnels. This end-to-end preservation of the original Ethernet FCS is useful for troubleshooting.

Ethernet FCS preservation is off by default on the ML-Series card. Configure Ethernet FCS preservation at the interface or sub-interface configuration level with the **[no] fcs-preservation-on** command. To operate correctly, both ends of the EoMPLS tunnel need to be configured for FCS preservation.

Configuring MPLS-TE

Perform the following tasks on the MPLS network before you enable MPLS-TE on the ML-Series card:

- Turn on MPLS tunnels
- Turn on OSPF

To configure MPLS-TE on the ML-Series card, perform the tasks described in the following sections:

- [Configuring an ML-Series Card for Tunnels Support](#)
- [Configuring an Interface to Support RSVP-Based Tunnel Signalling and IGP Flooding](#)
- [Configuring OSPF and Refresh Reduction for MPLS-TE](#)
- [Configuring an MPLS-TE Tunnel](#)



Note

The ML-Series card does not support MPLS-TE with IS-IS.



Note

Cisco Express Forwarding (CEF) is on by default on the ML-Series card.

Configuring an ML-Series Card for Tunnels Support

To configure an ML-Series card to support tunnels, use the following commands in global configuration mode:

	Command	Purpose
Step 1	Router(config)# mpls traffic-eng tunnels	Enables the MPLS-TE tunnel feature on a device.

Configuring an Interface to Support RSVP-Based Tunnel Signalling and IGP Flooding

To configure an interface to support RSVP-based tunnel signalling and IGP flooding, use the following commands in interface configuration mode:



Note

You must enable the tunnel feature on interfaces or sub-interfaces that you want to support MPLS-TE.

**Note**

A VC type 4 requires one POS interface to be configured for MPLS-TE tunnel and the other POS interface configured for the 802.1Q tunnel.

	Command	Purpose
Step 1	Router(config-if)# mpls traffic-eng tunnels	Enables MPLS-TE tunnels on an RPR (SPR) interface or on a POS interface.
Step 2	Router(config-if)# ip rsvp bandwidth <i>bandwidth</i>	Enables RSVP for IP on an interface and specifies the amount of bandwidth that will be reserved. For a description of the ip rsvp interface command syntax, see the <i>Cisco IOS Quality of Service Solutions Command Reference</i> .

Configuring OSPF and Refresh Reduction for MPLS-TE

For a description of the OSPF commands (excluding the OSPF traffic engineering commands), see the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols*

To configure OSPF and Refresh Reduction for MPLS-TE, use the following commands beginning in global configuration mode .

	Command	Purpose
Step 1	Router(config)# router ospf <i>process-id</i>	Configures an OSPF routing process for IP and places the router in configuration mode. The <i>process-id</i> argument is an internally used identification parameter for an OSPF routing process. It is locally assigned and can be any positive integer. Assign a unique value for each OSPF routing process.
Step 2	Router(config-router)# mpls traffic-eng area <i>area-id</i>	Turns on MPLS-TE for a specified OSPF area.
Step 3	Router(config-router)# mpls traffic-eng router-id <i>loopback0</i>	Specifies that the traffic engineering router identifier for the node is the IP address associated with interface loopback0.
Step 4	Router(config)# ip rsvp signalling refresh reduction	Reduces the messaging load imposed by RSVP

Configuring an MPLS-TE Tunnel

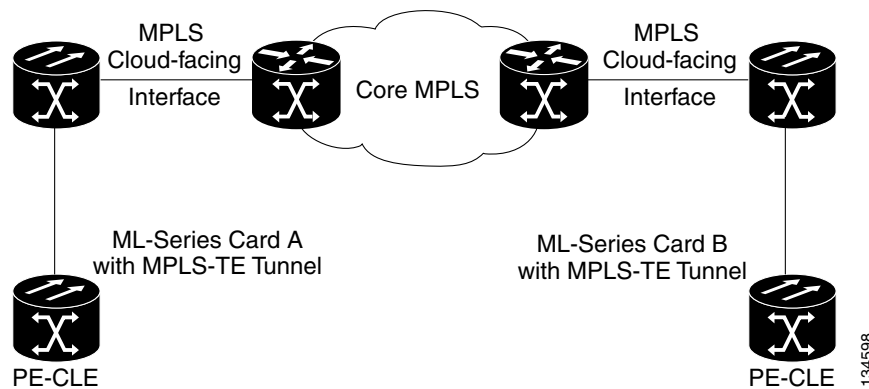
To configure an MPLS-TE tunnel, use the following commands, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# interface tunnel	Configures an interface type and enters interface configuration mode.
Step 2	Router(config)# ip unnumbered loopback0	Gives the tunnel interface an IP address. An MPLS-TE tunnel interface should be unnumbered because it represents a unidirectional link.
Step 3	Router(config-if)# tunnel destination A.B.C.D	Specifies the destination for a tunnel.
Step 4	Router(config-if)# tunnel mode mpls traffic-eng	Sets the tunnel encapsulation mode to MPLS-TE.
Step 5	Router(config-if)# tunnel mpls traffic-eng autoroute announce	Specifies IGP should use the tunnel (if the tunnel is up) in its enhanced shortest path first (SPF) calculation.
Step 6	Router(config-if)# tunnel mpls traffic-eng bandwidth bandwidth	Configures the bandwidth for the MPLS-TE tunnel.
Step 7	Router(config-if)# tunnel mpls traffic-eng path-option number {dynamic explicit}{name path-name path-number} [lockdown]	Configures the tunnel to use a named IP explicit path or a dynamic path.

MPLS-TE Configuration Example

Figure 18-3 illustrates the sample network that the configuration commands reference. Example 18-5 lists relevant portions of the configuration files for enabling MPLS-TE on ML-Series card A in the sample network. ML-Series card A is configured with an explicit path.

Figure 18-3 MPLS-TE Configuration Example



Example 18-5 ML-Series Card A Configuration

```

microcode mpls
ip subnet-zero
no ip domain-lookup
!
mpls label protocol ldp
mpls traffic-eng tunnels

```

```

no mpls traffic-eng auto-bw timers frequency 0
!
!
!
interface Loopback0
 ip address 222.222.222.222 255.255.255.255
!
interface Tunnel0
 ip unnumbered Loopback0
 tunnel destination 212.212.212.212
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng path-option 1 explicit identifier 1
!
interface Tunnel1
 ip unnumbered Loopback0
 tunnel destination 212.212.212.212
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng path-option 2 explicit identifier 2
!
interface GigabitEthernet0
 no ip address
 shutdown
!
interface GigabitEthernet1
 no ip address
!
interface GigabitEthernet1.1
 encapsulation dot1Q 10
 fcs-preservation-on
 mpls l2transport route 212.212.212.212 222
!
interface GigabitEthernet1.2
 encapsulation dot1Q 20
 mpls l2transport route 212.212.212.212 223
!
interface GigabitEthernet1.3
 encapsulation dot1Q 30
 mpls l2transport route 212.212.212.212 224
!
interface POS0
 ip address 170.170.170.172 255.255.255.0
 mpls traffic-eng tunnels
 tag-switching ip
 ip rsvp bandwidth 10000
!
interface POS1
 ip address 2.1.1.22 255.255.255.0
 mpls traffic-eng tunnels
 tag-switching ip
 ip rsvp bandwidth 10000
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 log-adjacency-changes
 network 2.1.1.22 0.0.0.0 area 0
 network 170.170.170.172 0.0.0.0 area 0
 network 222.222.222.222 0.0.0.0 area 0
!
ip classless
no ip http server
!

```



```

!
ip explicit-path identifier 1 enable
  next-address 2.1.1.1
  next-address 192.168.3.2
  next-address 192.168.3.1
  next-address 2.2.1.1
  next-address 2.2.1.2
  next-address 212.212.212.212
!
ip explicit-path identifier 2 enable
  next-address 170.170.170.171
  next-address 192.168.3.2
  next-address 192.168.3.1
  next-address 2.2.1.1
  next-address 2.2.1.2
  next-address 212.212.212.212
!
!
!
!
control-plane
!
!
!
line con 0
  exec-timeout 0 0
line vty 0 4
  exec-timeout 0 0
  password xxx
  no login

```

Monitoring and Verifying MPLS-TE and IP RSVP

Table 18-2 shows the privileged EXEC commands supported to monitor and verify the state of MPLS-TE tunnels on the ML-Series cards.

Table 18-3 Commands for Monitoring and Verifying MPLS-TE

Command	Purpose
<code>show mpls traffic-eng autoroute</code>	Displays tunnels announced to the Interior Gateway Protocol (IGP), including interface, destination, and bandwidth
<code>show mpls traffic-eng link-management admission-control</code>	Displays which tunnels were admitted locally and their parameters (such as, priority, bandwidth, incoming and outgoing interface, and state).
<code>show mpls traffic-eng link-management advertisements</code>	Displays local link information that MPLS traffic engineering link management is currently flooding into the global traffic engineering topology.
<code>show mpls traffic-eng link-management bandwidth-allocation</code>	Displays current local link information.

Table 18-3 *Commands for Monitoring and Verifying MPLS-TE (continued)*

Command	Purpose
<code>show mpls traffic-eng link-management igp-neighbors</code>	Displays IGP neighbors.
<code>show mpls traffic-eng link-management interfaces</code>	Displays interface resource and configuration information.
<code>show mpls traffic-eng link-management summary</code>	Displays a summary of link management information including link counts.
<code>show mpls traffic-eng topology</code>	Displays the MPLS-TE global topology as currently known at this node.
<code>show mpls traffic-eng tunnel</code>	Displays information about MPLS-TE tunnels, including LSP Tunnels Process and RSVP process.
<code>show mpls traffic-eng tunnel summary</code>	Displays condensed information about MPLS-TE tunnels.

Table 18-2 shows the privileged EXEC commands supported to monitor and verify the state of IP RSVP on the ML-Series cards.

Table 18-4 *Commands for Monitoring and Verifying IP RSVP*

Command	Purpose
<code>show ip rsvp interface [interface-type interface-number]</code>	Displays Resource Reservation Protocol (RSVP)-related information.
<code>show ip rsvp installed [interface-type interface-number]</code>	Displays RSVP-related installed filters and corresponding bandwidth information.
<code>show ip rsvp neighbor [interface-type interface-number]</code>	Displays current RSVP neighbors.
<code>show ip rsvp sender [interface-type interface-number]</code>	Displays RSVP path-related sender information currently in the database.
<code>show ip rsvp request [interface-type interface-number]</code>	Displays RSVP-related request information being requested upstream
<code>show ip rsvp reservation [interface-type interface-number]</code>	Displays RSVP-related receiver information currently in the database

RPRW Alarm

For information on the ONS 15454 RPRW alarm, refer to the online version of the *Cisco ONS 15454 Troubleshooting Guide, Release 5.0* at

<http://www.cisco.com/univercd/cc/td/doc/product/ong/15400/r50docs/r50tblsh/r50alts.htm>



Configuring RMON

This chapter describes how to configure remote network monitoring (RMON) on the ML-Series card for the ONS 15454 SONET/SDH.

RMON is a standard monitoring specification that defines a set of statistics and functions that can be exchanged between RMON-compliant console systems and network probes. RMON provides you with comprehensive network-fault diagnosis, planning, and performance-tuning information. The ML-Series card features RMON and is designed to work with a network management system (NMS).



Note

For complete syntax and usage information for the commands used in this chapter, see the “System Management Commands” section in the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.



Note

For general information about using Cisco IOS to manage RMON, refer to the “Configuring RMON Support” chapter of the Cisco IOS Configuration Fundamentals Configuration Guide.

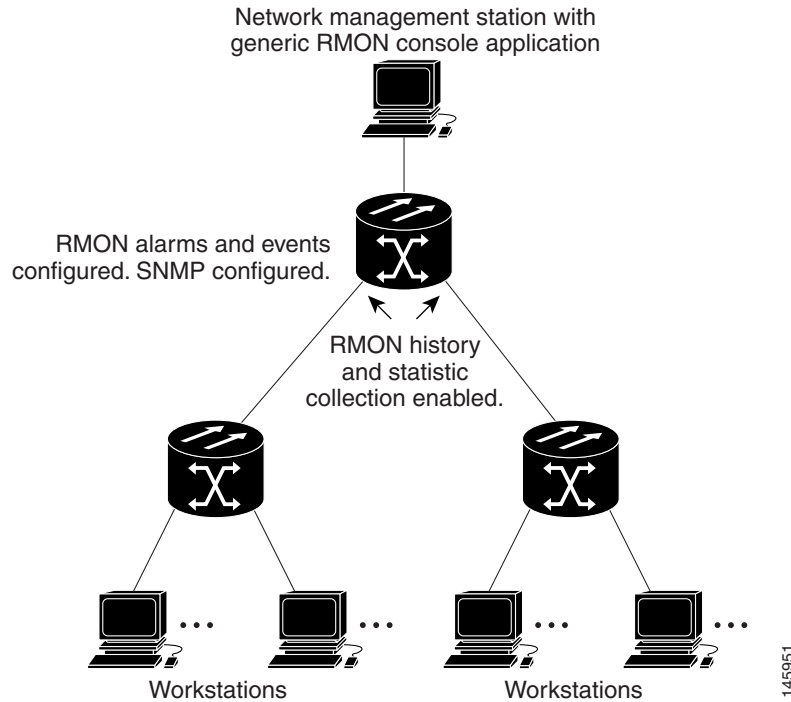
This chapter consists of these sections:

- [Understanding RMON, page 19-1](#)
- [Configuring RMON, page 19-2](#)
- [Understanding ML-Series Card CRC Error Threshold, page 19-7](#)
- [Configuring the ML-Series Card CRC Error Threshold, page 19-13](#)
- [Configuring ML-Series Card RMON for CRC Errors, page 19-16](#)
- [Determining the ifIndex Number for an ML-Series Card, page 19-19](#)
- [Displaying RMON Status, page 19-21](#)

Understanding RMON

RMON is an Internet Engineering Task Force (IETF) standard monitoring specification that allows various network agents and console systems to exchange network monitoring data. You can use the RMON feature with the Simple Network Management Protocol (SNMP) agent to monitor all the traffic flowing among ML-Series card and other switches on all connected LAN segments.

For information on the MIBs supported by the ML-Series card, see the “Supported MIBs” section on [page 20-5](#).

Figure 19-1 Remote Monitoring Example

Configuring RMON

These sections describe how to configure RMON on your ML-Series card:

- [Default RMON Configuration, page 19-2](#)
- [Configuring RMON Alarms and Events, page 19-2](#) (required)
- [Collecting Group History Statistics on an Interface, page 19-5](#) (optional)
- [Collecting Group Ethernet Statistics on an Interface, page 19-6](#) (optional)

Default RMON Configuration

RMON is disabled by default; no alarms or events are configured.

Configuring RMON Alarms and Events

You can configure your ML-Series card for RMON by using the command-line interface (CLI) or an SNMP-compatible network management station. We recommend that you use a generic RMON console application on the NMS to take advantage of RMON's network management capabilities. You must also configure SNMP on the ML-Series card to access RMON MIB objects. For more information about configuring SNMP, see [Chapter 20, "Configuring SNMP."](#)

Beginning in privileged EXEC mode, follow these steps to enable RMON alarms and events. This procedure is required.

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	rmon event <i>number</i> [description <i>string</i>] [log] [owner <i>string</i>] [trap <i>community</i>]	Add an event in the RMON event table that is associated with an RMON event number. <ul style="list-style-type: none"> For <i>number</i>, assign an event number. The range is 1 to 65535. (Optional) For description <i>string</i>, specify a description of the event. (Optional) Use the log keyword to generate an RMON log entry when the event is triggered. (Optional) For owner <i>string</i>, specify the owner of this event. (Optional) For trap <i>community</i>, enter the SNMP community string used for this trap.
Step 3	rmon alarm <i>number</i> <i>variable</i> <i>interval</i> { absolute delta } rising-threshold <i>value</i> [<i>event-number</i>] falling-threshold <i>value</i> [<i>event-number</i>] [owner <i>string</i>]	Set an alarm on a MIB object. <ul style="list-style-type: none"> For <i>number</i>, specify the alarm number. The range is 1 to 65535. For <i>variable</i>, specify the MIB object to monitor. For <i>interval</i>, specify the time in seconds that the alarm monitors the MIB variable. The range is 1 to 2147483647 seconds. Specify the absolute keyword to test each MIB variable directly. Specify the delta keyword to test the change between samples of a MIB variable. For <i>value</i>, specify a number at which the alarm is triggered and a number at which the alarm is reset. The range for the rising threshold and falling threshold values is -2147483648 to 2147483647. (Optional) For <i>event-number</i>, specify the event number to trigger when the rising or falling threshold exceeds its limit. (Optional) For owner <i>string</i>, specify the owner of the alarm.
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 an alarm, use the **no rmon alarm *number*** global configuration command on each alarm you configured. You cannot disable all the alarms that you configured by not specifying a specific number. You must disable each alarm separately. To disable an event, use the **no rmon event *number*** global configuration command. To learn more about alarms and events and how they interact with each other, see RFC 1757.

You can set an alarm on any MIB object. The following example configures RMON alarm number 10 by using the **rmon alarm** command. The alarm monitors the MIB variable *ifEntry.20.1* once every 20 seconds to check the change in the variable's rise or fall until the alarm is disabled. If the *ifEntry.20.1* value shows a MIB counter increase of 15 or more, such as from 100000 to 100015, the alarm is triggered. The alarm in turn triggers event number 1, which is configured with the **rmon event** command. Possible events can include a log entry or an SNMP trap. If the *ifEntry.20.1* value changes by 0, the alarm is reset and can be triggered again.

```
ML_Series(config)# rmon alarm 10 ifInErrors.65539 20 delta rising 15 1 fall 0
```


Note

The example does not trigger an optional event when the falling-threshold is 0.

Where 65539 is the SNMP IfIndex for interface POS 0. You can get the SNMP ifIndex for a given port with an SNMP get. In the example output, you can see that the SNMP ifIndex for POS0 is 65539:

```
tuvoks-view:128> getmany -v2c 10.92.56.97 tcc@1 ifDescr
ifDescr.65536 = GigabitEthernet0
ifDescr.65537 = GigabitEthernet1
ifDescr.65538 = Null10
ifDescr.65539 = POS0
ifDescr.65540 = POS1
ifDescr.65541 = SPR1
tuvoks-view:129>
```


The following example creates RMON event number 1 by using the **rmon event** command. The event is defined as *High ifOutErrors* and generates a log entry when the event is triggered by the alarm. The user *jjones* owns the row that is created in the event table by this command. This example also generates an SNMP trap when the event is triggered.

```
ML_Series(config)# rmon event 1 log trap eventtrap description "High ifOutErrors" owner
jjones
```

Collecting Group History Statistics on an Interface

You must first configure RMON alarms and events to display collection information.

Beginning in privileged EXEC mode, follow these steps to collect group history statistics on an interface. This procedure is optional.

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	interface <i>interface-id</i>	Specify the interface on which to collect history, and enter interface configuration mode.  Note Group history statistics do not work on packet-over-SONET/SDH (POS_ interfaces, only on Ethernet interfaces.
Step 3	rmon collection history <i>index</i> [buckets <i>bucket-number</i>] [interval <i>seconds</i>] [owner <i>ownername</i>]	Enable history collection for the specified number of buckets and time period. <ul style="list-style-type: none"> For <i>index</i>, identify the RMON group of statistics. The range is 1 to 65535. (Optional) For buckets <i>bucket-number</i>, specify the maximum number of buckets desired for the RMON collection history group of statistics. The range is 1 to 65535. The default is 50 buckets. (Optional) For interval <i>seconds</i>, specify the number of seconds in each polling cycle. The range is 1 to 3600. The default is 1800 seconds. (Optional) For owner <i>ownername</i>, enter the name of the owner of the RMON group of statistics.
Step 4	end	Return to privileged EXEC mode.
Step 5	show running-config	Verify your entries.
Step 6	show rmon history	Display the contents of the ML-Series card history table.
Step 7	copy running-config startup-config	(Optional) Save your entries in the configuration file.

To disable history collection, use the **no rmon collection history** *index* interface configuration command.

This example shows how to collect and show RMON history for the owner *root*:

```
ML_Series(config)# interface gigabitethernet1
ML_Series(config-if)# rmon collection history 2 owner root
ML_Series(config-if)# end
ML_Series# show rmon history
Entry 2 is active, and owned by root
Monitors ifIndex.393217 every 1800 second(s)
Requested # of time intervals, ie buckets, is 50,
```

Collecting Group Ethernet Statistics on an Interface

Beginning in privileged EXEC mode, follow these steps to collect group Ethernet statistics on an interface. This procedure is optional.

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	interface <i>interface-id</i>	Specify the interface on which to collect statistics, and enter interface configuration mode.
Step 3	rmon collection stats <i>index</i> [owner <i>ownername</i>]	Enable RMON statistic collection on the interface. <ul style="list-style-type: none"> For <i>index</i>, specify the RMON group of statistics. The range is from 1 to 65535. (Optional) For owner <i>ownername</i>, enter the name of the owner of the RMON group of statistics.
Step 4	end	Return to privileged EXEC mode.
Step 5	show running-config	Verify your entries.
Step 6	show rmon statistics	Display the contents of the ML-Series card statistics table.
Step 7	copy running-config startup-config	(Optional) Save your entries in the configuration file.

To disable the collection of group Ethernet statistics, use the **no rmon collection stats** *index* interface configuration command.

This example shows how to collect RMON statistics for the owner *root*:

```
ML_Series(config)# interface gigabitethernet1
ML_Series(config-if)# rmon collection stats 2 owner root
```


Understanding ML-Series Card CRC Error Threshold

The POS ports on the ML-Series card report alarms for SONET/SDH defects and GFP defects, including signal fail (SF) and signal degrade (SD) alarms. In most circumstances, these alarms alert the user to problems that also cause excessive CRC errors on the POS port. But there are situations where excessive CRC errors will occur on the POS port, but the link will not have any SONET defects or GFP defects to report. Examples of this situation include an ML-Series card at the other end of the link sending out packets with CRC errors or a bit error rate too low to trigger SF or SD defect, but high enough to cause a meaningful CRC packet error rate.

In these situations with a default ML-Series card RPR implementation and no reported SONET/SDH or GFP defects, the POS interface remains in the up state as a member of the shared packet ring (SPR) interface. Traffic is lost quietly and does not trigger any alarms or action.

The FCS threshold configuration and detection feature fixes this problem. The user can now configure the ML-Series card to raise an alarm if the percentage of packet loss due to CRC errors crosses a configurable threshold. The alarm raised is the CRC Threshold Crossing Alarm (CRC-ALARM), which is a service-affecting (SA) SONET/SDH alarm with a Major (MA) severity. Reported SONET/SDH alarms can be viewed under the Alarms tab of CTC.

The CRC_ALARM is raised on Ethernet interfaces when the rate of Ethernet frames with CRC errors exceeds a defined threshold. A raised CRC_Alarm can initiate an RPR wrap. For the wrap to occur, the RPR POS interface (SPR) needs to belong to an RPR, and the RPR wrap option must be configured on the the RPR wrap trigger.

Configurable Threshold and Triggered Actions

The configurable threshold is not set with a BER, since variable frame lengths and varying percentages of bandwidth can impair the usefulness of this measure. Instead, the users configure a more relevant measure using CRC error rate as a percentage of the traffic. The available triggering thresholds are:

- 10e-2 or 1% traffic (1 CRC error 100 packets)
- 10e-3 or 0.1% traffic (1 CRC error in 1000 packets) (default)
- 10e-4 or 0.01% traffic (1 CRC error in 10000 packets)

The default threshold is a CRC error rate of 0.1% of the traffic. For voice and video traffic, an error rate of 1% is typically a critical issue and 0.1% is a major issue. Voice and video needs to trigger a wrap if the error rate is higher than 0.1% (1 error every 1000 packets). For normal data traffic, an error rate of 10% traffic is a critical issue, requiring an immediate fix, and 1% traffic is a minor issue.

The following actions occur after the detection of excessive CRC errors:

1. The RPR wraps if this option is configured.
2. The link shuts down if this option is configured.
3. If the link shuts down, a path defect indication (PDI) is sent to the far-end ML-Series card port. This ensures the remote end wraps.
4. A CRC-ALARM is raised against the local end ML-Series card port. (If the remote end is also receiving excessive CRC errors, a CRC-ALARM is also raised against the far end ML-Series card port.

SONET/GFP Suppression of CRC-ALARM

This detection of excessive CRC errors is independent of SONET/GFP defects. A problem may have the potential to trigger both the SONET/GFP defects and the CRC-ALARM. In this scenario, the SONET/GFP defect will trigger before the CRC-WRAP alarm because CRC error threshold detection is a slower process. If the SONET/GFP defect causes the link to go down, this link-down happens before the CRC-ALARM is detected, and it suppresses the CRC-ALARM. If the SONET/GFP defect that causes CRC-ALARM is not a link-down trigger and the CRC-ALARM is configured to take the link down, the CRC-ALARM will report and trigger the link down.

Clearing of CRC-ALARM

When the trigger action is disabled (default), the CRC-ALARM automatically clears when the error rate falls below the threshold for a significant time period.

When the trigger action is enabled, a CRC-ALARM requires a manual clear from the user. This is required because the wrap or link down caused by the alarm blocks both traffic and the CRC errors in the traffic from the port. So with no CRC errors, an automatic clear would occur even though the underlying problem, such as dirty fiber or a defective ML-Series card, still exists. Interface flapping can occur in this situation.

Before doing a manual clear, the user needs to determine the root cause of a CRC-ALARM and correct it. After that, the user has several alternative methods to manually clear the alarm:

- Through the Cisco IOS CLI, enter the **clear crc alarm interface** *interface-type interface-number* command at the EXEC level.
- Through the Cisco IOS CLI, do an administrative **shutdown** on the linked ports and then a **no shutdown** to enable the ports.
- Through CTC or TL-1, disable and then re-enable the circuit.
- Through CTC or TL-1, delete the SONET/SDH circuit and create a replacement circuit with the same source and destination.

Unwrap Synchronization

The software on the ML-Series card raises the CRC-ALARM alarm on the POS interface that sees the errored frames. For unidirectional FCS errors, the user only needs to issue the unwrap command on the POS port at one end of the span, the one which raised the CRC-ALARM alarm. For bidirectional failures, both ends of the span raise the CRC-ALARM alarm and the user is required to issue the command once at each end of the span.

Since the POS ports at each end of the link are wrapped, removing the wrap (unwrapping) when the CRC-ALARM is cleared requires coordination. The software must also make sure that other errors that might cause wrapping are absent. The following examples illustrate this process for both unidirectional and bidirectional failures. For simplicity, the examples assume that excessive CRC errors is the only existing condition that might cause wrapping.

Unidirectional Errors

Figure 19-2 shows an RPR wrapped by excessive unidirectional CRC errors on POS port 0 of node E, which is also reporting the CRC-ALARM. This caused POS port 1 on node E and POS port 0 on node D to wrap. The figure captions further explain the process.

Figure 19-2 Wrapped RPR with Unidirectional Excessive CRC Errors

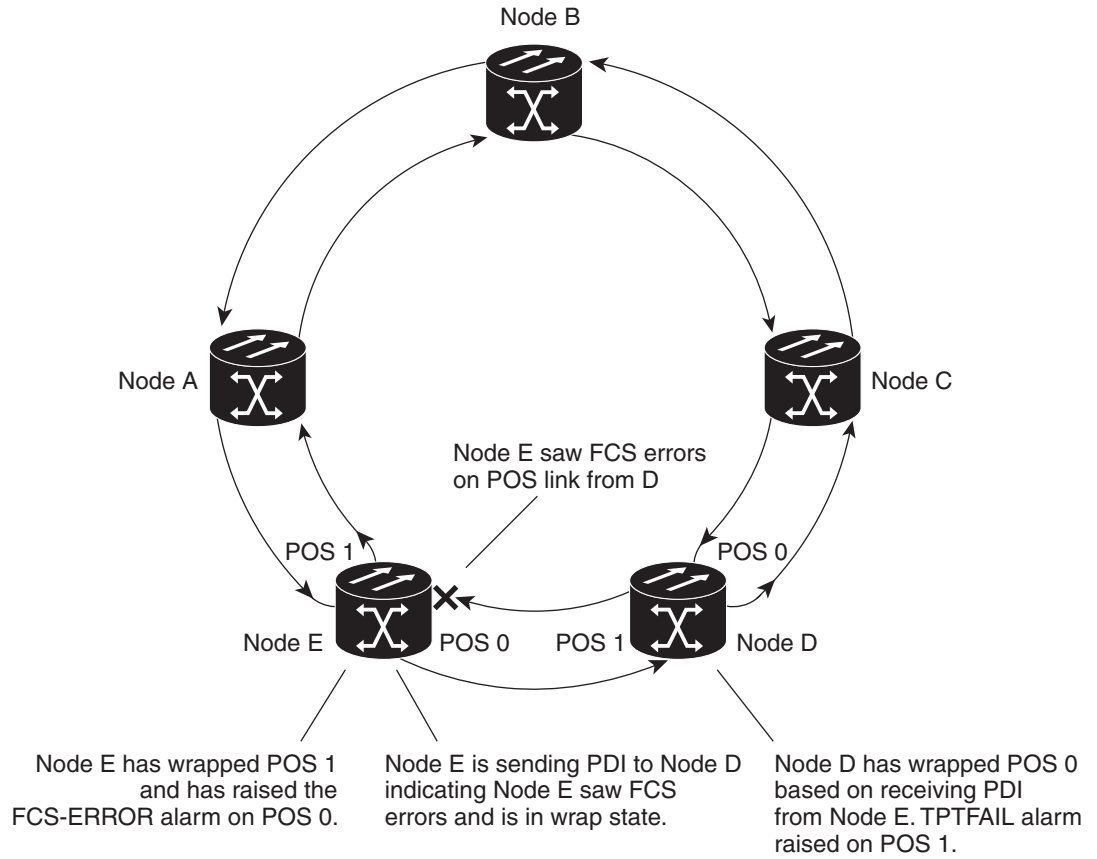
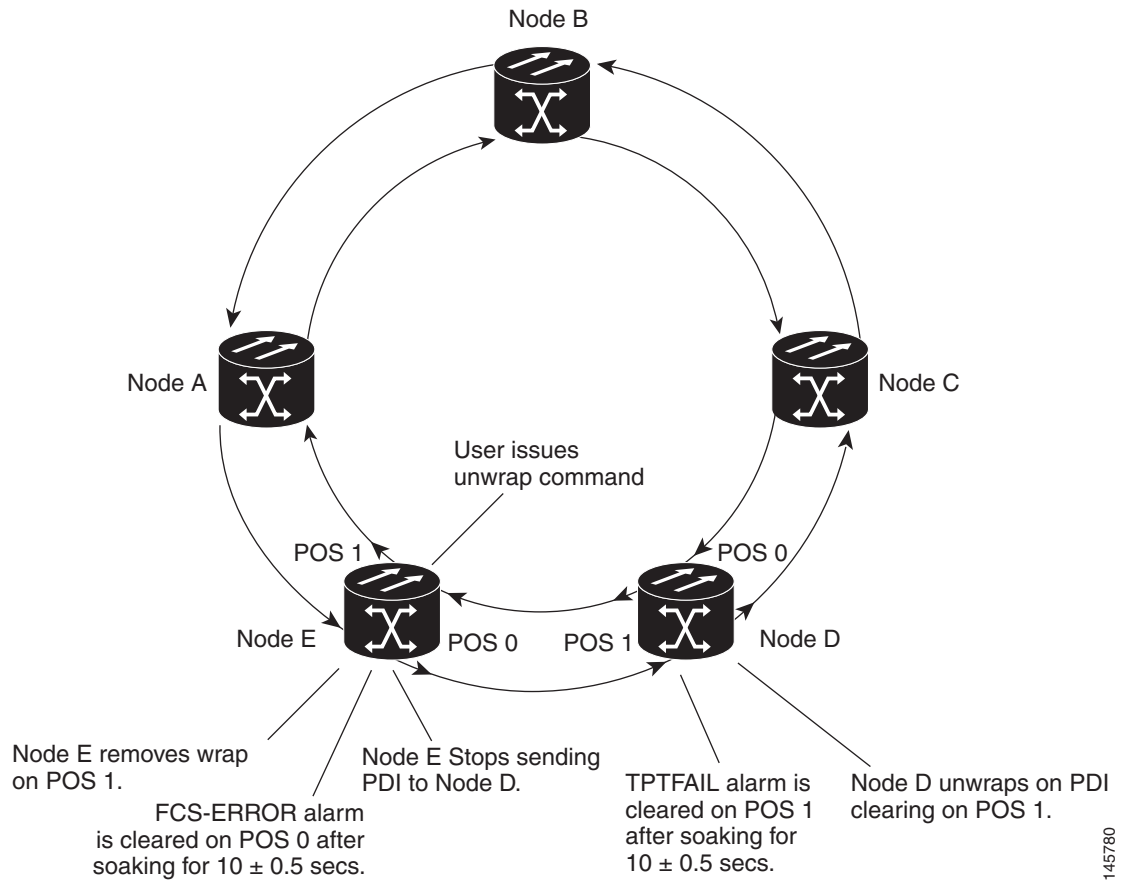


Figure 19-3 illustrates the unwrap sequence for Figure 19-2. The traffic hit for the unwrap is dependent on the soak time required to declare PDI cleared on node D.

Figure 19-3 Unwrapped RPR with Unidirectional Excessive CRC Errors



Bidirectional Errors

Figure 19-4 shows an RPR wrapped by excessive bidirectional CRC errors, both ports are reporting CRC-ALARMS. The figure captions further explain the process.

Figure 19-4 Wrapped RPR with Bidirectional Excessive CRC Errors

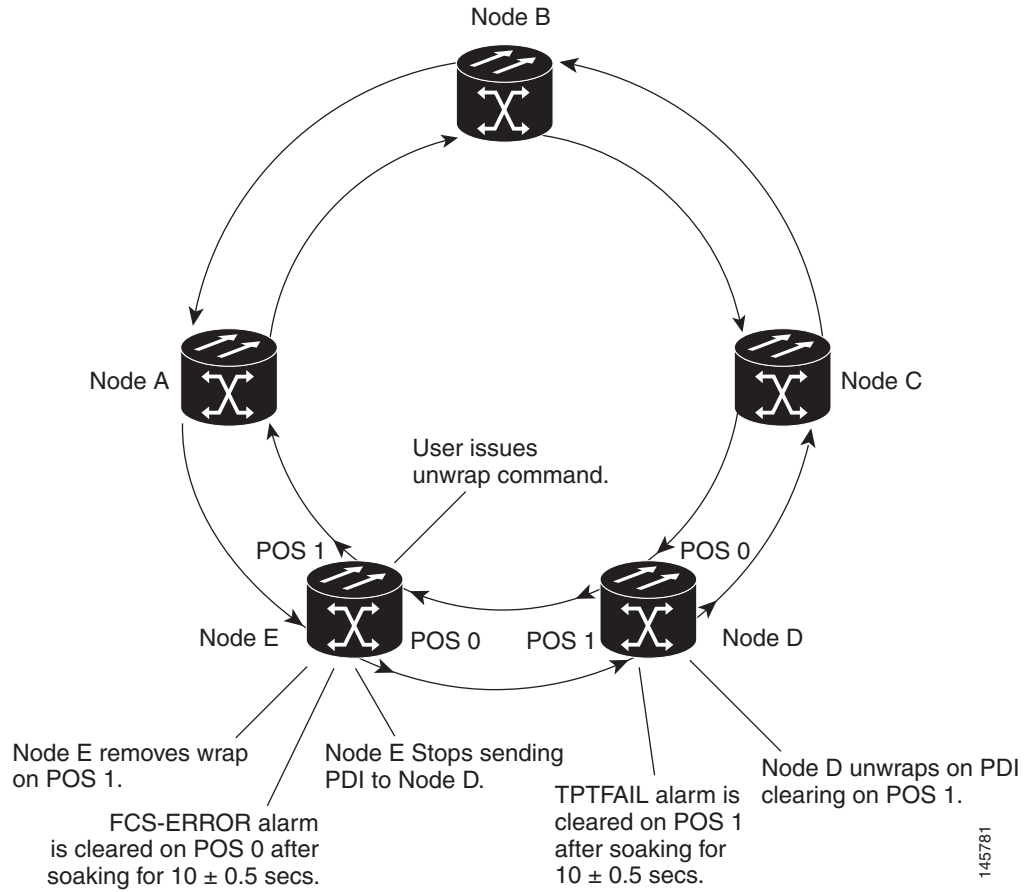
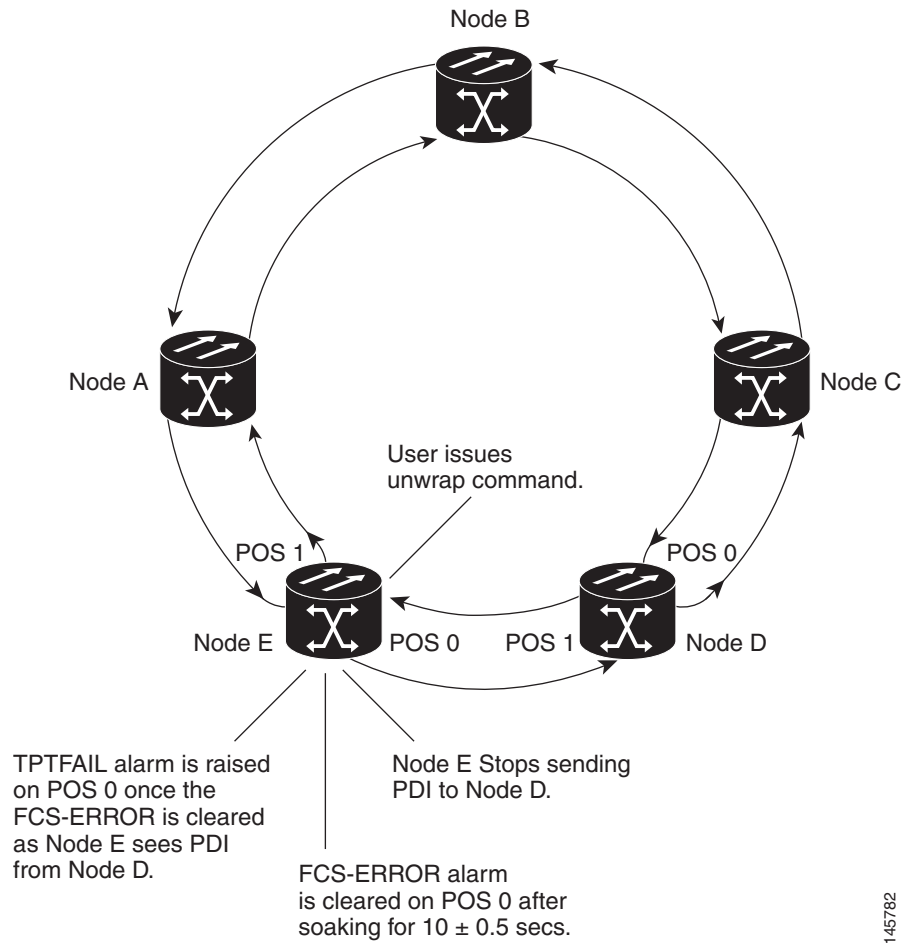


Figure 19-5 illustrates the first part of the unwrap sequence for Figure 19-6. This occurs after the unwrap command is configured on node E. For unwrap in this bidirectional scenario, the user must configure the command on the POS ports at both ends of the link.

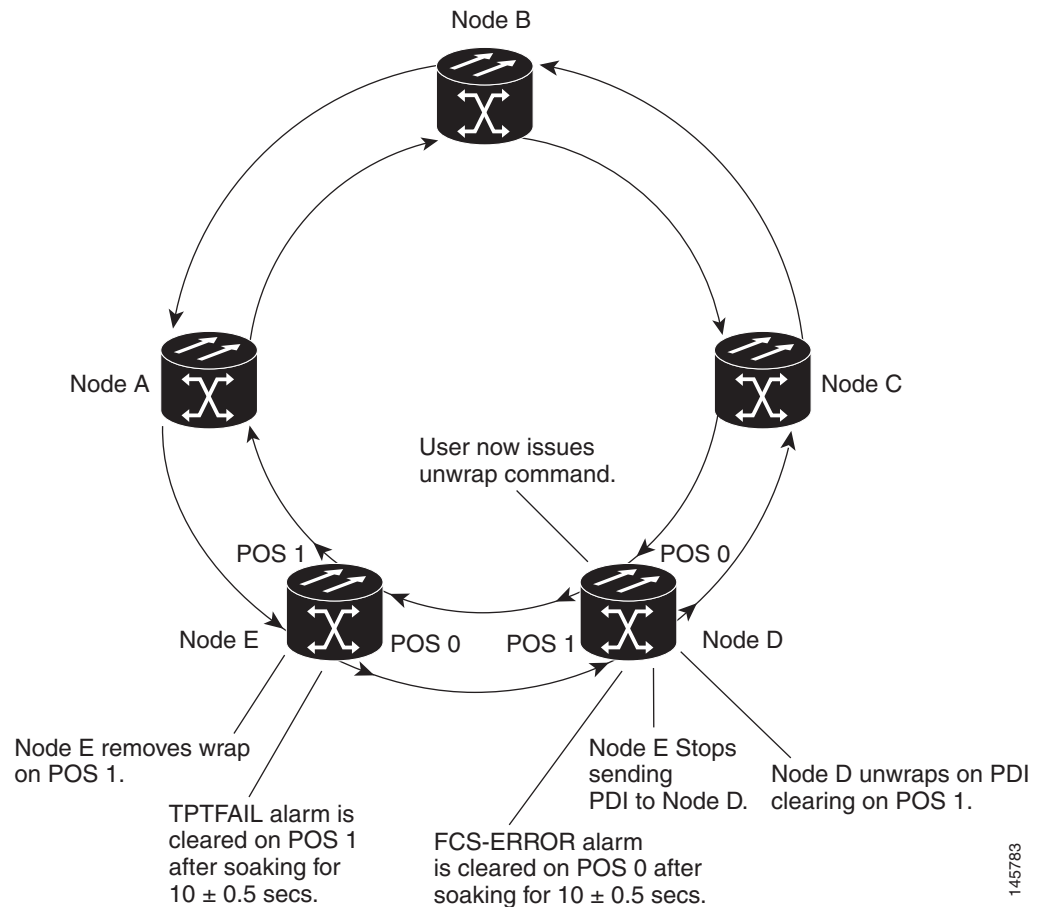
Figure 19-5 First Stage of Unwrapped RPR with Bidirectional Excessive CRC Errors



Node E has not unwrapped POS port 1 after the first CRC-ALARM clear command. Since node D continues to send PDI to node E, node E will raise the TPTFAIL alarm once the CRC-ALARM is cleared. At this point, the RPR is in a state similar to the unidirectional failure. The unwrap completes after the user issues the second unwrap command, as illustrated by [Figure 19-6](#).

145782

Figure 19-6 Second Stage of Unwrapped RPR with Bidirectional Excessive CRC Errors



Configuring the ML-Series Card CRC Error Threshold

By default, the CRC-ALARM is disabled. When the alarm is configured, the link down and wrap actions are still disabled by default.

Beginning in privileged EXEC mode, follow these steps to configure the ML-Series card CRC error threshold:

	Command	Purpose
Step 1	ML_Series# configure terminal	Enter global configuration mode.
Step 2	ML_Series (config)# interface interface-type interface-number	Enters interface configuration mode.

	Command	Purpose
Step 3	ML_Series (config-if)# [no] trigger crc threshold <i>[threshold-value]</i>	Sets an fcs error level as a percentage of bandwidth to trip the SONET/SDH CRC-ALARM. Valid values are: <ul style="list-style-type: none"> • 2—10e-2 or 1% traffic (1 CRC error in 100 packets) • 3—10e-3 or 0.1% traffic (1 CRC error in 1000 packets) (default) • 4—10e-4 or 0.01% traffic (1 CRC error in 10000 packets) The no form of the command sets the level back to the default threshold of 3 .
Step 4	ML_Series (config-if)# [no] trigger crc action	(Optional) Sets the CRC-ALARM to trigger a link down for the reporting port. Set on an RPR POS port, this also wraps the RPR. The no form of the command sets the trigger back to the default off.
Step 5	ML_Series(config-if)# [no] trigger crc delay soak-time	(Optional) Sets the minutes of soak time for excessive CRC error detection. The valid value is from 3 minutes to 10 minutes. The no form of the command sets the delay back to the default of one minute.
Step 6	ML_Series# end	Return to privileged EXEC mode.
Step 7	ML_Series# show running-config	Verify your entries.
Step 8	ML_Series# copy running-config startup-config	(Optional) Save your entries in the configuration file.

Clearing the CRC-ALARM Wrap with the Clear CRC Error Command

The Cisco IOS CLI **clear crc alarm interface** *interface-type interface-number* command is intended to clear the RPR wrap when it occurs due to FCS errors without corresponding SONET/SDH errors. It is not intended to unwrap wraps due to other causes, such as SONET/SDH defects or keep alive (KA) failures. If SONET/SDH or KA defects are present without FCS errors, the software rejects the command with an error message. When FCS errors are present and SONET/SDH or KA defects are present, the command is accepted by the software but the node unwraps only after all the failures have been fixed. In this case, the user does not need to reissue the command after the SONET/SDH or KA defect has cleared.



Note

The unwrap does not occur immediately, but after conditions are met.

Beginning in privileged EXEC mode, follow these steps to clear the ML-Series card CRC-ALARM:

	Command	Purpose
Step 1	ML_Series # clear crc alarm interface <i>interface-type</i> <i>interface-number</i>	Clears the SONET/SDH CRC-ALARM and allows the RPR to unwrap when conditions are met.

Configuring ML-Series Card RMON for CRC Errors

The ML-Series card supports using an NMS for SNMP performance monitoring (PM), including monitoring cyclic redundancy check (CRC) errors. If the NMS supports periodic polling and programmed threshold values to monitor interface index errors (ifInErrors) for all the ML-Series card interfaces, you can manage and monitor CRC errors by relying on the NMS.

If the NMS does not support polling or if the desired polling frequency uses too much bandwidth, you can configure SNMP traps on the ML-Series card through the Cisco IOS CLI. This method is only for ML-Series cards on the ONS 15454 SONET/SDH. RMON capabilities for ML-Series cards on the ONS 15310-CL and ONS 15310-MA are best managed through Cisco Transport Controller (CTC), Transaction Language One (TL1), or Cisco Transport Manager (CTM) in the standard manner for the node.

Configuration Guidelines for CRC Thresholds on the ML-Series Card

These are the guidelines for determining the interface CRC errors (ifInErrors) threshold values for generating an NMS PM alert:

- SONET/SDH bit errors also create POS CRC errors. There is no alarm suppression hierarchy between the SONET/SDH errors and POS errors, so each set of errors creates separate alerts.
- The actual packet rate of an interface is unpredictable. A high bandwidth interface might forward only a few packets per minute in a particular time period of low data traffic, which means a relatively low number of CRC errors would represent a 100 percent loss. A lower bandwidth interface might forward a high packet count (millions) per minute during a particular time period, and so a relatively few CRC errors would represent an error rate of 10^{-9} . This situation prevents the straightforward determination of a maximum bit error rate (BER), which is often used for non-packet-based PM.
- You can set up the monitoring of ML-Series card CRC errors for either signs of minor trouble or signs of major trouble. For minor trouble monitoring, set a relatively quick and sensitive error rate trigger, such as 10 errors in a 60 second period. This method will likely generate an NMS alert every time an interface goes up or down, a fiber error occurs, or a SONET/SDH protection event occurs (even though protection might occur within 50 ms). To monitor only major trouble and to reduce the number of alerts, set a relatively high threshold, such as 1000 errors in a 300 second period.

Accessing CRC Errors Through SNMP

CRC errors for each interface are reported in the IF-MIB object ifInErrors (OID 1.3.6.1.2.1.2.1.14). Users can check the current value of ifInErrors through SNMP get requests. Each ML-Series card runs a separate instance of SNMP. SNMP requests are relayed to the individual ML-Series card based on the community string. The community string uses the following format:

```
com_str_configured_from_CTC@m1_slot_number
```

ifIndex Operation in the ML-Series Card

ifIndex mapping is saved to the TCC2/TCC2P database so a replacement card also receives the same mapping. However, the ifIndex mapping is only saved with the IOS CLI command **copy running-config startup-config**.

The ifIndex mapping is cleared when an external startup-config file is uploaded for the ML-Series card slot using a non-Cisco IOS management interface, including the startup-configuration file upload feature of CTC, CTM, and TL1. ML-Series card resets with no stored ifIndex mapping may re-assign ifIndex values. This is important when deployment configurations are created offline and uploaded to the node through the startup-config upload feature. The ifindex mapping may also be lost during upgrades or downgrades of the system software. SNMP requires that all interface ifindex values be persistent (do not change) once established, even after configuration changes and ML-Series card resets.

**Caution**

ML-Series card resets may re-assign ifIndex values if configurations are created offline and uploaded to the node. The Cisco IOS CLI global commands **snmp-server ifindex persist** and **copy running-config startup-config** prevent this.

To force persistent ifindex values in Cisco IOS, the mapping from interface to ifindex must be saved whenever the configuration is saved, and restored after a reset. This is not done by default. To enable persistent snmp ifindex values, the following Cisco IOS CLI command must be entered and then saved to the startup configuration.

Global configuration command: **[no] snmp-server ifindex persist**

Example:

```
Router(config)# snmp-server ifindex persist
```

```
Router(config)# copy running-config startup-config
```

When this command is enabled and the new configuration is saved, this will also store the interface to ifindex mapping to the TCC2/TCC2P cards.

Configuring an SNMP Trap for the CRC Error Threshold Using Cisco IOS

The ML-Series card supports RMON trap functionality in Cisco IOS. You must use the Cisco IOS CLI to configure RMON to monitor ifInErrors and generate a trap to an NMS when a threshold is crossed. The ML-Series card on the ONS 15454 SONET/SDH does not support the configuration of RMON traps through an SNMP set request, which typically initiates an action on a network device.

Beginning in privileged EXEC mode, follow these steps to configure RMON to monitor ifInErrors and generate a trap for an NMS when a threshold is crossed:

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	rmon event <i>number</i> [log] [trap <i>community</i>] [description <i>string</i>] [owner <i>string</i>]	<p>Add an event in the RMON event table that is associated with an RMON event number.</p> <ul style="list-style-type: none"> • For <i>number</i>, assign an event number. The range is 1 to 65535. • (Optional) Use the log keyword to generate an RMON log entry when the event is triggered. • (Optional) For trap <i>community</i>, enter the SNMP community string used for this trap. • (Optional) For description <i>string</i>, specify a description of the event. • (Optional) For owner <i>string</i>, specify the owner of this event.
Step 3	rmon alarm <i>number</i> ifInErrors . <i>ifIndex-number</i> <i>interval</i> { absolute delta } rising-threshold <i>value</i> [<i>event-number</i>] falling-threshold <i>value</i> [<i>event-number</i>] [owner <i>string</i>]	<p>Set an alarm on the MIB object.</p> <ul style="list-style-type: none"> • For <i>number</i>, specify the alarm number. The range is 1 to 65535. • The <i>ifIndex-number</i> variable is the ifIndex number of an ML-Series card interface in decimal form. (For information about determining this number, see “Determining the ifIndex Number for an ML-Series Card” section on page 19-19.) • For <i>interval</i>, specify the time in seconds the alarm monitors the MIB variable. The range is 1 to 4294967295 seconds. • Specify the absolute keyword to test each MIB variable directly. Specify the delta keyword to test the change between samples of a MIB variable. • For <i>value</i>, specify a number at which the alarm is triggered and a number at which the alarm is reset. The range for the rising threshold and falling threshold values is -2147483648 to 2147483647. • (Optional) For <i>event-number</i>, specify the event number to trigger when the rising or falling threshold exceeds its limit. • (Optional) For owner <i>string</i>, specify the owner of the alarm.
Step 4	end	Return to privileged EXEC mode.

	Command	Purpose
Step 5	<code>show running-config</code>	Verify your entries.
Step 6	<code>copy running-config startup-config</code>	(Optional) Save your entries in the configuration file.

Below is an example of configuring an SNMP trap for the CRC error threshold.

```
ML_Series # configure terminal
ML_Series(config)# rmon event 10 log trap slot15 owner config
ML_Series(config)# rmon alarm 9 ifInErrors.983043 300 delta rising-threshold 1000 10
falling-threshold 1000 10 owner config
ML_Series(config)# end
ML_Series # show running-config
ML_Series # copy running-config startup-config
```

The ifIndex number of an ML-Series card interface in decimal form used for the **rmon alarm** command in the example is **ifInErrors.983043**. This variable is the MIB object to monitor combined with the ifIndex number of an ML-Series card interface. For information on determining the ifIndex number for an ML-Series card, see [“Determining the ifIndex Number for an ML-Series Card” section on page 19-19](#).

Below is an example of a rising-threshold trap generated by 1002 ifInErrors crossing a threshold of 1000 in a 5-minute period.

```
2005-03-22 16:25:38 ptlm9-454e56-97.cisco.com [10.92.56.97]:
SNMPv2-MIB:sysUpTime.0 = Wrong Type (should be Timeticks): 43026500
SNMPv2-MIB:snmpTrapOID.0 = OID: RMON-MIB:risingAlarm
RFC1271-MIB:alarmIndex.9 = 9
RFC1271-MIB:alarmVariable.9 = OID: IF-MIB:ifInErrors.983043
RFC1271-MIB:alarmSampleType.9 = deltaValue(2)
RFC1271-MIB:alarmValue.9 = 1002
RFC1271-MIB:alarmRisingThreshold.9 = 1000
SNMPv2-SMI:snmpModules.18.1.3.0 = IPAddress: 10.92.56.97
```

Determining the ifIndex Number for an ML-Series Card

When an NMS polls an ML-Series card for performance data, the NMS uses ifIndex numbers internally to consolidate interface data from multiple MIBs and associate this data with an interface name. The user can rely on the interface name and does not need to know the actual ifIndex number.

When you use the Cisco IOS CLI to configure the ML-Series card to generate traps directly, you do not have this associated name to use. You must use the actual ifIndex number for each interface being configured with a trap. To determine the actual ifIndex number, you can use an NMS to retrieve the ifIndex number of each ML-Series card interface and VLAN subinterface, or you can calculate the ifIndex number for the interface.

The user can also use a MIB browser (SNMP MIB definition lookup service) to examine the ifDescr for the appropriate ifIndex number. The ifIndex number from the ifDescr must be the ifIndex number for the desired port.

On an ML-Series card, the ifIndex number of Ethernet and POS interfaces is compiled from two pieces of information:

- The chassis slot number of the card—The slot number is the number of the physical space in the shelf that the ML-Series card resides in. It ranges from Slot 1 to Slot 6 or Slot 12 to Slot 17 on an ONS 15454 SONET/SDH shelf. You can find this information in many ways, including through the graphical representation of the shelf slots on CTC, or by looking at the front of the physical shelf.

- A local port number within the card—Port numbers of the ML-Series cards for the ONS 15454 SONET/SDH match the interface numbers for Fast Ethernet and Gigabit Ethernet interfaces. POS port numbers do not match the interface numbers and do not consecutively follow the Ethernet port numbering. A consecutive value is skipped between the last Ethernet port number and the first POS number (POS Port 0). Port numbers for the interfaces are listed in [Table 19-1](#):

Table 19-1 Port Numbers for the Interfaces of ML-Series Cards

ML100T-12 FastEthernet Interfaces	ML100T-12 POS Interfaces	ML100X-8 FastEthernet Interfaces	ML100X-8 POS Interfaces	ML1000-2 Gigabit Ethernet Interfaces	ML1000-2 POS Interfaces
FE 0 = Port 0	POS 0 = Port 13	FE 0 = Port 0	POS 0 = Port 9	GE 0 = Port 0	POS 0 = Port 3
FE 1 = Port 1	POS 1 = Port 14	FE 1 = Port 1	POS 1 = Port 10	GE 1 = Port 1	POS 1 = Port 4
FE 2 = Port 2		FE 2 = Port 2			
FE 3 = Port 3		FE 3 = Port 3			
FE 4 = Port 4		FE 4 = Port 4			
FE 5 = Port 5		FE 5 = Port 5			
FE 6 = Port 6		FE 6 = Port 6			
FE 7 = Port 7		FE 7 = Port 7			
FE 8 = Port 8					
FE 9 = Port 9					
FE 10 = Port 10					
FE 11 = Port 11					

The slot and port are combined to form the ifIndex using the following formula:

$$\text{ifIndex} = (\text{slot} * 10000\text{h}) + (\text{port})$$

10000h is the hexadecimal equivalent number of 65536. The resulting ifIndex is a meaningful two-part number in hexadecimal, but seems confusing and arbitrary in decimal. For example, ifIndex E0002h is Slot 14, Port 2. This same number in decimal notation is 917506. The **rmon alarm** command requires the ifindex number in decimal form.

As an additional reference for calculating the correct ifindex value to use with the **rmon alarm** command, [Table 19-1](#) lists the base ifindex number for Slots 1 to 17. The desired port number can be added to the slot base number to quickly determine the correct ifIndex number.

Table 19-2 Port Numbers for the Interfaces of ML-Series Cards

Slot Number for the ML-Series Card	Base ifIndex Number in Hexidecimal Format	Base ifIndex Number in Decimal Format
1	10000h	65536
2	20000h	131072
3	30000h	196608
4	40000h	262144
5	50000h	327680

Table 19-2 Port Numbers for the Interfaces of ML-Series Cards (continued)

Slot Number for the ML-Series Card	Base ifIndex Number in Hexidecimal Format	Base ifIndex Number in Decimal Format
6	60000h	393216
12	C0000h	786432
13	D0000h	851968
14	E0000h	917504
15	F0000h	983040
16	100000h	1048576
17	110000h	1114112

Manually Checking CRC Errors on the ML-Series Card

Users can also check the current count of ML-Series card CRC errors on an interface by using the **show interface** command. The example shows six total input errors, which are all CRC errors, in the last line of the output.

```
ML_Series(config)# show interface pos 0

POS0 is up, line protocol is up
Hardware is Packet/Ethernet over Sonet, address is 0005.9a39.713e (bia 0005.9a39.713e)
MTU 1500 bytes, BW 48384 Kbit, DLY 100 usec,
reliability 255/255, txload 1/255, rxload 182/255
Encapsulation: Cisco-EoS-LEX, crc 32, loopback not set
Keepalive set (10 sec)
Scramble enabled
ARP type: ARPA, ARP Timeout 04:00:00
Last input never, output never, output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 34621000 bits/sec, 60083 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
311190527 packets input, 931220183 bytes
Received 0 broadcasts (0 IP multicast)
6 runts, 0 giants, 0 throttles
0 parity
6 input errors, 6 CRC, 0 frame, 0 overrun, 0 ignored
```

Displaying RMON Status



Note

RMON status commands do not work for POS interfaces.

To display the RMON status, use one or more of the privileged EXEC commands in [Table 19-3](#).

Table 19-3 Commands for Displaying RMON Status

Command	Purpose
show rmon	Displays general RMON statistics.
show rmon alarms	Displays the RMON alarm table.
show rmon events	Displays the RMON event table.
show rmon history	Displays the RMON history table.
show rmon statistics	Displays the RMON statistics table.

Example 19-1 shows examples of the commands in Table 19-3.

Example 19-1 CRC Errors Displayed with show rmon Commands

```
ML_Series# show rmon alarms

Alarm 9 is active, owned by config
Monitors ifInErrors.983043 every 300 second(s)
Taking delta samples, last value was 0
Rising threshold is 1000, assigned to event 10
Falling threshold is 1000, assigned to event 10
On startup enable rising or falling alarm

ML_Series# show rmon events
Event 10 is active, owned by config
Description is
Event firing causes log and trap to community slot15,
last event fired at 0y3w2d,00:32:39,
Current uptime      0y3w6d,03:03:12
Current log entries:
index  uptime          description
1      0y3w2d,00:32:39
```




Configuring SNMP

This chapter describes how to configure the ML-Series card for operating with Simple Network Management Protocol (SNMP).



Note

For complete syntax and usage information for the commands used in this chapter, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.

This chapter consists of these sections:

- [Understanding SNMP, page 20-1](#)
- [Configuring SNMP, page 20-5](#)
- [Displaying SNMP Status, page 20-14](#)

Understanding SNMP

SNMP is an application-layer protocol that provides a message format for communication between managers and agents. The SNMP system consists of an SNMP manager, an SNMP agent, and a MIB. The SNMP manager can be part of a network management system (NMS) such as CiscoWorks. To configure SNMP, you define the relationship between the manager and the agent.

The SNMP agent contains MIB variables whose values the SNMP manager can request or change. A manager can get a value from an agent or store a value in an agent. The agent gathers data from the MIB, the repository for information about device parameters and network data. The agent can also respond to a manager's requests to get or set data.

An agent can send unsolicited traps to the manager. Traps are messages that alert the SNMP manager to a condition on the network. Traps can mean improper user authentication, restarts, link status (up or down), MAC address tracking, closing of a Transmission Control Protocol (TCP) connection, loss of connection to a neighbor, or other significant events.

This section includes information about these topics:

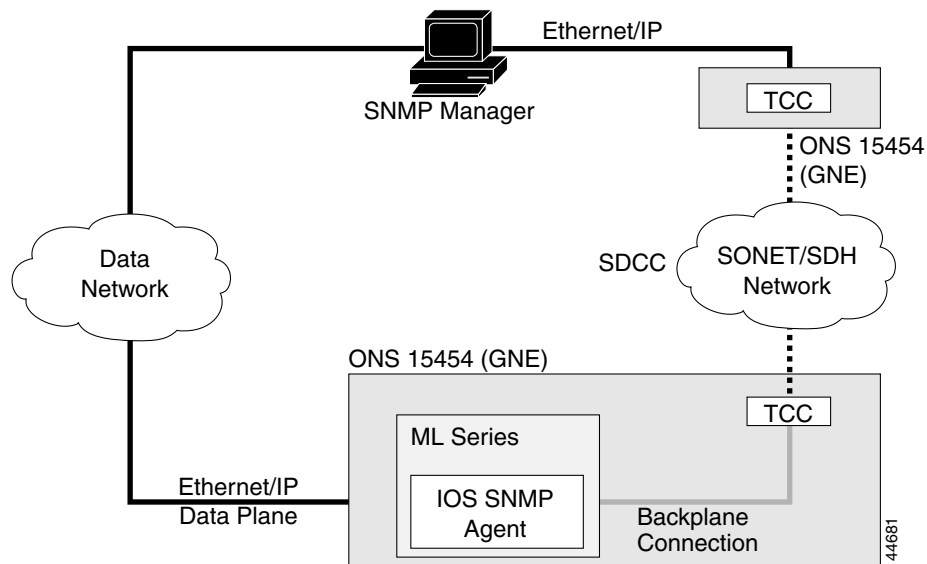
- [SNMP on the ML-Series Card, page 20-2](#)
- [SNMP Versions, page 20-3](#)
- [SNMP Manager Functions, page 20-3](#)
- [SNMP Agent Functions, page 20-4](#)
- [SNMP Community Strings, page 20-4](#)

- [Using SNMP to Access MIB Variables, page 20-4](#)
- [Supported MIBs, page 20-5](#)
- [SNMP Notifications, page 20-5](#)

SNMP on the ML-Series Card

SNMP operates in two different ways on the ONS 15454 SONET/SDH ML-Series card. One way is to communicate directly. This is also how SNMP operates on a small Catalyst switch, using direct communication, Cisco IOS, and the data plane. An SNMP agent interacting with an ML-Series card can also communicate through the ONS 15454 SONET/SDH and the SONET network. Both ways are shown in [Figure 20-1](#).

Figure 20-1 SNMP on the ML-Series Card Example



When the ONS 15454 SONET/SDH node relays the ML-Series card SNMP communication, the node uses a proxy agent to accept, validate, and forward get, getNext, and set requests to the ML-Series card. These ML-Series card requests contain the slot identification of the ML-Series card cards to distinguish the request from a general SNMP request for the ONS 15454 SONET/SDH node. The responses from the ML-Series card are then relayed by the ONS 15454 SONET/SDH node to the requesting SNMP agents.

SNMP access is useful for collecting Cisco IOS data plane events, alarms, and statistics for the ML-Series card. All SNMP events and traps defined on the ML-Series card are reported to the TCC2/TCC2P card SNMP agent by default. If the TCC2/TCC2P card SNMP agent is active, these events are sent to the defined SNMP server.

SNMP Versions

Both the ML-Series card and the ONS 15454 SONET/SDH nodes support SNMP Version 1 (SNMPv1) and SNMP Version 2c (SNMPv2c), defined as:

- **SNMPv1**—The Simple Network Management Protocol, a full Internet standard, defined in RFC 1157.
- **SNMPv2c** replaces the party-based administrative and security framework of SNMPv2 classic with the community-string-based administrative framework of SNMPv2c while retaining the bulk retrieval and improved error handling of SNMPv2classic. The bulk retrieval mechanism retrieves tables and large quantities of information, minimizing the number of round-trips required. The SNMPv2c improved error-handling includes expanded error codes that distinguish different kinds of error conditions; these conditions are reported through a single error code in SNMPv1. Error return codes in SNMPv2c report the error type.

SNMPv1 and SNMPv2c have the same security models and levels:

- **Level**—noAuthNoPriv
- **Authentication**—community string
- **Encryption**—none
- **Result**—Uses a community string match for authentication.

You must configure the SNMP agent to use the SNMP version supported by the management station. Because an agent can communicate with multiple managers, you can configure the software to support communications using SNMPv1, and SNMPv2c protocols.

SNMP Manager Functions

The SNMP manager uses information in the MIB to perform the operations described in [Table 20-1](#).

Table 20-1 *SNMP Operations*

Operation	Description
get-request	Retrieves a value from a specific variable.
get-next-request	Retrieves a value from a variable within a table. ¹
get-bulk-request ²	Retrieves large blocks of data, such as multiple rows in a table, that would otherwise require the transmission of many small blocks of data.
get-response	Replies to a get-request, get-next-request, or set-request sent by an NMS.
set-request	Stores a value in a specific variable.
trap	An unsolicited message sent by an SNMP agent to an SNMP manager when some event has occurred.

1. With this operation, an SNMP manager does not need to know the exact variable name. A sequential search is performed to find the needed variable from within a table.
2. The **get-bulk-request** command only works with SNMPv2 or later.

SNMP Agent Functions

The SNMP agent responds to SNMP manager requests as follows:

- Get a MIB variable—The SNMP agent begins this function in response to a request from the NMS. The agent retrieves the value of the requested MIB variable and responds to the NMS with that value.
- Set a MIB variable—The SNMP agent begins this function in response to a message from the NMS. The SNMP agent changes the value of the MIB variable to the value requested by the NMS.

The SNMP agent also sends unsolicited trap messages to notify an NMS that a significant event has occurred on the agent. Examples of trap conditions include, but are not limited to, when a port or module goes up or down, when spanning-tree topology changes occur, and when authentication failures occur.

SNMP Community Strings

SNMP community strings authenticate access to MIB objects and function as embedded passwords. In order for the NMS to access the ML-Series card, the community string definitions on the NMS must match at least one of the three community string definitions on the ML-Series card.

A community string can have one of these attributes:

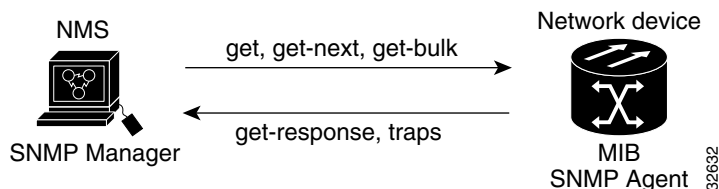
- Read-only (RO)—Gives read access to authorized management stations to all objects in the MIB except the community strings, but does not allow write access
- Read-write (RW)—Gives read and write access to authorized management stations to all objects in the MIB, but does not allow access to the community strings
- Read-write-all—Gives read and write access to authorized management stations to all objects in the MIB, including the community strings

Using SNMP to Access MIB Variables

An example of an NMS is the CiscoWorks network management software. CiscoWorks software uses the ML-Series card MIB variables to set device variables and to poll devices on the network for specific information. The results of a poll can be displayed as a graph and analyzed to troubleshoot problems, increase network performance, verify the configuration of devices, monitor traffic loads, and more.

As shown in [Figure 20-2](#), the SNMP agent gathers data from the MIB. The agent can send traps, or notification of certain events, to the SNMP manager, which receives and processes the traps. Traps alert the SNMP manager to a condition on the network such as improper user authentication, restarts, link status (up or down), MAC address tracking, and so forth. The SNMP agent also responds to MIB-related queries sent by the SNMP manager in get-request, get-next-request, and set-request format.

Figure 20-2 SNMP Network



Supported MIBs

The complete list of supported MIBs for the ML-Series card is found in the MIBsREADME.txt file on the ONS Software CD for your release. This software CD also includes the needed MIB modules and information on loading MIBs.

Some of the important MIBs supported include:

- Spanning Tree Protocol (STP) traps from Bridge-MIB (RFC 1493)
- Authentication traps from RFC 1157
- Link-up and link-down traps for Ethernet ports from IF-MIB (RFC 1573)
- Export of quality of service (QoS) statistics through the CISCO-PORT-QOS-MIB extension



Note

The ML-Series card CISCO-PORT-QOS-MIB extension includes support for QoS indexing based on cost of service (CoS). It does not support configuration objects.

SNMP Notifications

SNMP allows the ML-Series card to send notifications to SNMP managers when particular events occur. SNMP notifications can be sent as traps or as inform requests. In command syntax, unless there is an option in the command to select either traps or inform requests, the keyword *traps* refers to either traps or inform requests, or both. Use the **snmp-server host** command to specify whether to send SNMP notifications as traps or inform requests.



Note

SNMPv1 does not support inform requests.

Traps are unreliable because the receiver does not send an acknowledgment when it receives a trap, so the sender cannot determine if the trap was received. When an SNMP manager receives an inform request, it acknowledges the message with an SNMP response protocol data unit (PDU). If the sender does not receive a response, the inform request can be sent again. Because they can be re-sent, inform requests are more likely than traps to reach their intended destination.

The characteristics that make informs more reliable than traps also consume more resources in the ML-Series card and in the network. Unlike a trap, which is discarded as soon as it is sent, an inform request is held in memory until a response is received or the request times out. Traps are sent only once, but an inform might be re-sent or retried several times. The retries increase traffic and contribute to a higher overhead on the network. Therefore, traps and informs require a trade-off between reliability and resources. If it is important that the SNMP manager receive every notification, use inform requests. If traffic on the network or memory in the ML-Series card is a concern and notification is not required, use traps.

Configuring SNMP

This section describes how to configure SNMP on your ML-Series card. It contains this configuration information:

- [Default SNMP Configuration, page 20-6](#)
- [SNMP Configuration Guidelines, page 20-6](#)

- [Disabling the SNMP Agent, page 20-7](#)
- [Configuring Community Strings, page 20-7](#)
- [Configuring SNMP Groups and Users, page 20-8](#)
- [Configuring SNMP Notifications, page 20-10](#)
- [Setting the Agent Contact and Location Information, page 20-12](#)
- [Limiting TFTP Servers Used Through SNMP, page 20-13](#)
- [SNMP Examples, page 20-13](#)

Default SNMP Configuration

Table 20-2 shows the default SNMP configuration.

Table 20-2 **Default SNMP Configuration**

Feature	Default Setting
SNMP agent	Enabled
SNMP community strings	Read-Only: Public Read-Write: Private Read-Write-all: Secret
SNMP trap receiver	None configured
SNMP traps	None enabled except the trap for TCP connections (tty)
SNMP version	If no version keyword is present, the default is Version 1.
SNMP notification type	If no type is specified, all notifications are sent.

SNMP Configuration Guidelines

When configuring SNMP, follow these guidelines:

- When configuring an SNMP group, do not specify a notify view. The **snmp-server host** global configuration command autogenerates a notify view for the user and then adds it to the group associated with that user. Modifying the group's notify view affects all users associated with that group. For information about when you should configure notify views, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.
- An SNMP *group* is a table that maps SNMP users to SNMP views.
- An SNMP *user* is a member of an SNMP group.
- An SNMP *host* is the recipient of an SNMP trap operation.
- An SNMP *engine ID* is a name for the local or remote SNMP engine.

Disabling the SNMP Agent

Beginning in privileged EXEC mode, follow these steps to disable the SNMP agent:

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	no snmp-server	Disable the SNMP agent operation.
Step 3	end	Return to privileged EXEC mode.
Step 4	show running-config	Verify your entries.
Step 5	copy running-config startup-config	(Optional) Save your entries in the configuration file.

The **no snmp-server** global configuration command disables all running versions on the device. No specific Cisco IOS command exists to enable SNMP. The first **snmp-server** global configuration command that you enter enables all versions of SNMP.

Configuring Community Strings

You use the SNMP community string to define the relationship between the SNMP manager and the agent. The community string acts like a password to permit access to the agent on the ML-Series card. Optionally, you can specify one or more of these characteristics associated with the string:

- An access list of IP addresses of the SNMP managers that are permitted to use the community string to gain access to the agent
- A MIB view, which defines the subset of all MIB objects accessible to the given community
- Read and write or read-only permission for the MIB objects accessible to the community

Beginning in privileged EXEC mode, follow these steps to configure a community string on the ML-Series card:

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	snmp-server community <i>string</i> [view <i>view-name</i>] [ro rw] [<i>access-list-number</i>]	Configure the community string. <ul style="list-style-type: none"> • For <i>string</i>, specify a string that acts like a password and permits access to the SNMP protocol. You can configure one or more community strings of any length. • (Optional) For view <i>view-name</i>, specify the view record accessible to the community. • (Optional) Specify either read-only (ro) if you want authorized management stations to retrieve MIB objects, or specify read-write (rw) if you want authorized management stations to retrieve and modify MIB objects. By default, the community string permits read-only access to all objects. • (Optional) For <i>access-list-number</i>, enter an IP standard access list numbered from 1 to 99 and 1300 to 1999.

	Command	Purpose
Step 3	<code>access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>]</code>	<p>(Optional) If you specified an IP standard access list number in Step 2, then create the list, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> For <i>access-list-number</i>, enter the access list number specified in Step 2. The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. For <i>source</i>, enter the IP address of the SNMP manager that are permitted to use the community string to gain access to the agent. (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>Recall that the access list is always terminated by an implicit deny statement for everything.</p>
Step 4	<code>end</code>	Return to privileged EXEC mode.
Step 5	<code>show running-config</code>	Verify your entries.
Step 6	<code>copy running-config startup-config</code>	(Optional) Save your entries in the configuration file.



Note

To disable access for an SNMP community, set the community string for that community to the null string (do not enter a value for the community string).

To remove a specific community string, use the **no snmp-server community *string*** global configuration command.

This example shows how to assign the string comaccess to SNMP, to allow read-only access, and to specify that IP access list 4 can use the community string to gain access to the ML-Series card SNMP agent:

```
ML_Series(config)# snmp-server community comaccess ro 4
```

Configuring SNMP Groups and Users

You can specify an identification name (engine ID) for the local or remote SNMP server engine on the ML-Series card. You can configure an SNMP server group that maps SNMP users to SNMP views, and you can add new users to the SNMP group.

Beginning in privileged EXEC mode, follow these steps to configure SNMP on the ML-Series card:

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	snmp-server engineID { local <i>engineid-string</i> remote <i>ip-address</i> [udp-port <i>port-number</i>]}	Configure a name for either the local or remote copy of SNMP. <ul style="list-style-type: none"> The <i>engineid-string</i> is a 24-character ID string with the name of the copy of SNMP. If you select remote, specify the <i>ip-address</i> of the device that contains the remote copy of SNMP and the optional UDP port on the remote device. The UDP port default is 162.
Step 3	snmp-server group <i>groupname</i> { v1 v2c [auth noauth priv]} [read <i>readview</i>] [write <i>writeview</i>] [notify <i>notifyview</i>] [access <i>access-list</i>]	Configure a new SNMP group on the remote device. <ul style="list-style-type: none"> For <i>groupname</i>, specify the name of the group. Specify a security model: <ul style="list-style-type: none"> v1 is the less secure model. v2c is the more secure model. It allows transmission of inform requests and integers that are twice the normal width. <p>Note The priv keyword is available only when the crypto software image is installed.</p> <ul style="list-style-type: none"> (Optional) Enter read <i>readview</i> with a string (not to exceed 64 characters) that is the name of the view in which you can only view the contents of the agent. (Optional) Enter write <i>writeview</i> with a string (not to exceed 64 characters) that is the name of the view in which you can enter data and configure the contents of the agent. (Optional) Enter notify <i>notifyview</i> with a string (not to exceed 64 characters) that is the name of the view in which you can specify a notify, inform request, or trap. (Optional) Enter access <i>access-list</i> with a string (not to exceed 64 characters) that is the name of the access list.
Step 4	snmp-server user <i>username</i> <i>groupname</i> [remote <i>host</i> [udp-port <i>port</i>]] { v1 v2c [access <i>access-list</i>]}	Configure a new user to an SNMP group. <ul style="list-style-type: none"> The <i>username</i> is the name of the user on the host that connects to the agent. The <i>groupname</i> is the name of the group with which the user is associated. (Optional) Enter remote to specify a remote SNMP entity to which the user belongs and the hostname or IP address of that entity along with the optional UDP port number. The default UDP port number is 162. Enter the SNMP version number (v1 or v2c). (Optional) Enter access <i>access-list</i> with a string (not to exceed 64 characters) that is the name of the access list.
Step 5	end	Return to privileged EXEC mode.

	Command	Purpose
Step 6	<code>show running-config</code>	Verify your entries.
Step 7	<code>copy running-config startup-config</code>	(Optional) Save your entries in the configuration file.

Configuring SNMP Notifications

A trap manager is a management station that receives and processes notification types (traps). Traps are system alerts that the ML-Series card generates when certain events occur. By default, no trap manager is defined, and no traps are sent. To enable all traps, configure the `snmp-server enable traps` command with no notification type keywords specified.

Table 20-3 describes some of the more commonly used traps supported by the ML-Series card. You can enable any or all of these traps and configure a trap manager to receive them.

Table 20-3 ML-Series Card Notification Types

Notification Type Keyword	Description
<code>bridge</code>	Generates Spanning Tree Protocol (STP) bridge MIB traps.
<code>config</code>	Generates a trap for SNMP configuration changes.
<code>config-copy</code>	Generates a trap for SNMP copy configuration changes.
<code>entity</code>	Generates SNMP entity traps.
<code>rsvp</code>	Generates RSVP flow change traps.
<code>rtr</code>	Generates a trap for the SNMP Response Time Reporter (RTR).

You can send the `snmp-server host` global configuration command to a specific host to receive the notification types listed in Table 20-3.

Beginning in privileged EXEC mode, follow these steps to configure the ML-Series card to send traps or inform requests to a host:

	Command	Purpose
Step 1	<code>configure terminal</code>	Enter global configuration mode.
Step 2	<code>snmp-server engineID remote ip-address engineid-string</code>	Specify the IP address and engine ID for the remote host.

	Command	Purpose
Step 3	snmp-server user <i>username</i> <i>groupname</i> remote <i>host</i> [udp-port <i>port</i>] { v1 v2c }[access <i>access-list</i>]	Configure an SNMP user to be associated with the remote host created in Step 2. <ul style="list-style-type: none"> The <i>username</i> is the name of the user on the host that connects to the agent. The <i>groupname</i> is the name of the group with which the user is associated. (Optional) Enter remote to specify a remote SNMP entity to which the user belongs and the hostname or IP address of that entity along with the optional UDP port number. The default UDP port number is 162. Enter the SNMP version number (v1 or v2c). (Optional) Enter access <i>access-list</i> with a string (not to exceed 64 characters) that is the name of the access list. <p>Note You cannot configure a remote user for an address without first configuring the engine ID for the remote host. If you try to configure the user before configuring the remote engine ID, you receive an error message, and the command is not executed.</p>
Step 4	snmp-server host <i>host-addr</i> [traps informs] [version { 1 2c }] <i>community-string</i> [udp-port <i>port</i>] [<i>notification-type</i>]	Specify the recipient of an SNMP trap operation. <ul style="list-style-type: none"> For <i>host-addr</i>, specify the name or Internet address of the host (the targeted recipient). (Optional) Enter traps (the default) to send SNMP traps to the host. (Optional) Enter informs to send SNMP inform requests to the host. (Optional) Specify the SNMP version (1 or 2c). SNMPv1 does not support inform requests. For <i>community-string</i>, enter the password-like community string sent with the notification operation. (Optional) For udp-port <i>port</i>, enter the remote device UDP port. (Optional) For <i>notification-type</i>, use the keywords listed in Table 20-3 on page 20-10. If no type is specified, all notifications are sent.
Step 5	snmp-server enable traps <i>notification-types</i>	Enable the ML-Series card to send traps or inform requests and specify the type of notifications to be sent. For a list of notification types, enter: <p>snmp-server enable traps ?</p> To enable multiple types of traps, you must enter a separate snmp-server enable traps command for each trap type.
Step 6	snmp-server trap-source <i>interface-id</i>	(Optional) Specify the source interface, which provides the IP address for the trap message. This command also sets the source IP address for inform requests.
Step 7	snmp-server queue-length <i>length</i>	(Optional) Establish how many trap messages each trap host can hold (message queue length.) The range is 1 to 1000; the default is 10.
Step 8	snmp-server trap-timeout <i>seconds</i>	(Optional) Define how often to resend trap messages. The range is 1 to 1000; the default is 30 seconds.
Step 9	end	Return to privileged EXEC mode.

	Command	Purpose
Step 10	<code>show running-config</code>	Verify your entries.
Step 11	<code>copy running-config startup-config</code>	(Optional) Save your entries in the configuration file.

The `snmp-server host` command specifies which hosts receive the notifications. The `snmp-server enable trap` command globally enables the mechanism for the specified notification (for traps and inform requests). To enable a host to receive an inform request, you must configure an `snmp-server host informs` command for the host and globally enable inform requests by using the `snmp-server enable traps` command.

To remove the specified host from receiving traps, use the `no snmp-server host host` global configuration command. The `no snmp-server host` command with no keywords disables traps, but not inform requests, to the host. To disable inform requests, use the `no snmp-server host informs` global configuration command. To disable a specific trap type, use the `no snmp-server enable traps notification-types` global configuration command.

Setting the Agent Contact and Location Information

Beginning in privileged EXEC mode, follow these steps to set the system contact and location of the SNMP agent so that these descriptions can be accessed through the configuration file:

	Command	Purpose
Step 1	<code>configure terminal</code>	Enter global configuration mode.
Step 2	<code>snmp-server contact text</code>	Set the system contact string. For example: <code>snmp-server contact Dial System Operator at beeper 21555.</code>
Step 3	<code>snmp-server location text</code>	Set the system location string. For example: <code>snmp-server location Building 3/Room 222</code>
Step 4	<code>end</code>	Return to privileged EXEC mode.
Step 5	<code>show running-config</code>	Verify your entries.
Step 6	<code>copy running-config startup-config</code>	(Optional) Save your entries in the configuration file.

Limiting TFTP Servers Used Through SNMP

Beginning in privileged EXEC mode, follow these steps to limit the TFTP servers used for saving and loading configuration files through SNMP to the servers specified in an access list:

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	snmp-server tftp-server-list <i>access-list-number</i>	Limit TFTP servers used for configuration file copies through SNMP to the servers in the access list. For <i>access-list-number</i> , enter an IP standard access list numbered from 1 to 99 and 1300 to 1999.
Step 3	access-list <i>access-list-number</i> { deny permit } <i>source</i> [<i>source-wildcard</i>]	Create a standard access list, repeating the command as many times as necessary. <ul style="list-style-type: none"> For <i>access-list-number</i>, enter the access list number specified in Step 2. The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. For <i>source</i>, enter the IP address of the TFTP servers that can access the ML-Series card. (Optional) For <i>source-wildcard</i>, enter the wildcard bits, in dotted decimal notation, to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
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.

SNMP Examples

This example shows how to enable all versions of SNMP. The configuration permits any SNMP manager to access all objects with read-only permissions using the community string “public.” This configuration does not cause the ML-Series card to send any traps.

```
ML-Series(config)# snmp-server community public
```

This example shows how to permit any SNMP manager to access all objects with read-only permission using the community string “public.” The ML-Series card also sends VTP traps to the hosts 192.180.1.111 and 192.180.1.33 using SNMPv1 and to the host 192.180.1.27 using SNMPv2c. The community string “public” is sent with the traps.

```
ML-Series(config)# snmp-server community public
ML-Series(config)# snmp-server host 192.180.1.27 version 2c public
ML-Series(config)# snmp-server host 192.180.1.111 version 1 public
ML-Series(config)# snmp-server host 192.180.1.33 public
```

This example shows how to allow read-only access for all objects to members of access list 4 that use the comaccess community string. No other SNMP managers have access to any objects. SNMP authentication failure traps are sent by SNMPv2c to the host cisco.com using the community string “public.”

```
ML_Series(config)# snmp-server community comaccess ro 4
ML_Series(config)# snmp-server enable traps snmp authentication
ML_Series(config)# snmp-server host cisco.com version 2c public
```

This example shows how to send Entity MIB traps to the host cisco.com. The community string is restricted. The second line specifies the destination of these traps and overwrites any previous **snmp-server host** commands for the host cisco.com.

```
ML_Series(config)# snmp-server enable traps
ML_Series(config)# snmp-server host cisco.com restricted
```

This example shows how to enable the ML-Series card to send all traps to the host myhost.cisco.com using the community string “public.”

```
ML_Series(config)# snmp-server enable traps
ML_Series(config)# snmp-server host myhost.cisco.com public
```

Displaying SNMP Status

To display SNMP input and output statistics, including the number of illegal community string entries, errors, and requested variables, use the **show snmp** privileged EXEC command. You can also use the other privileged EXEC commands in [Table 20-4](#) to display SNMP information. For information about the fields in the output displays, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.

Table 20-4 Commands for Displaying SNMP Information

Feature	Default Setting
show snmp	Displays SNMP statistics.
show snmp group	Displays information about each SNMP group on the network.
show snmp pending	Displays information about pending SNMP requests.
show snmp sessions	Displays information about the current SNMP sessions.
show snmp user	Displays information about each SNMP user name in the SNMP users table.



POS on ONS Ethernet Cards

This chapter describes packet-over-SONET/SDH (POS) and its implementation on ONS Ethernet cards.

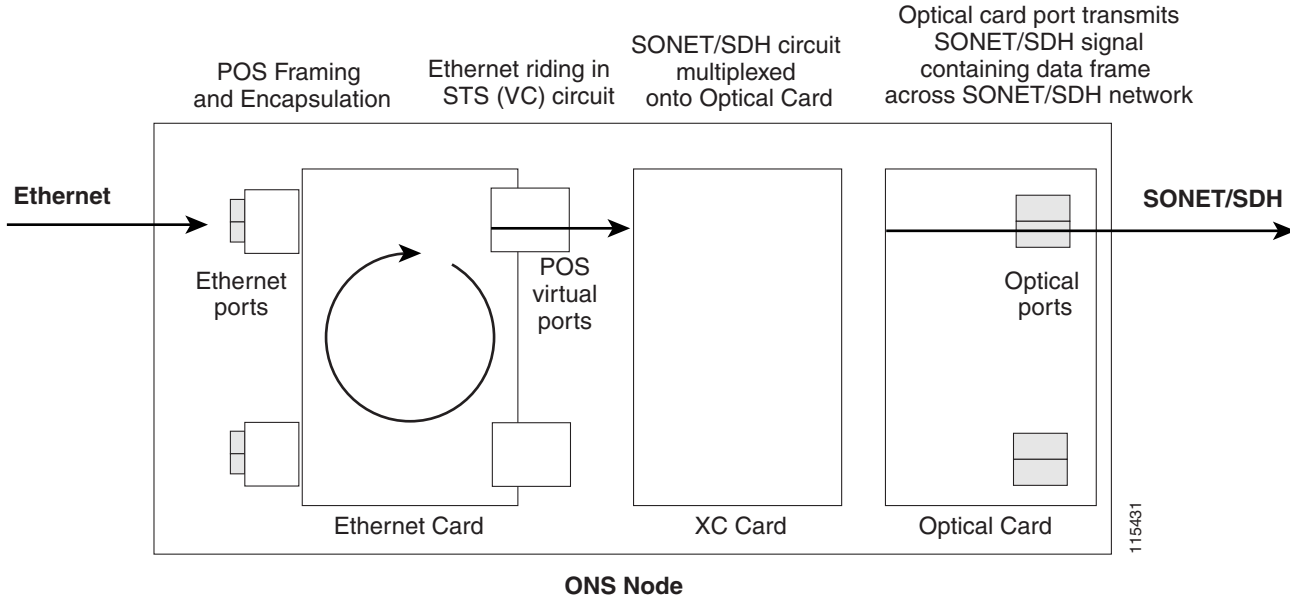
This chapter contains the following major sections:

- [POS Overview, page 21-1](#)
- [POS Interoperability, page 21-2](#)
- [POS Encapsulation Types, page 21-4](#)
- [POS Framing Modes, page 21-7](#)
- [POS Characteristics of Specific ONS Ethernet Cards, page 21-7](#)
- [Ethernet Clocking Versus SONET/SDH Clocking, page 21-11](#)

POS Overview

Unlike Asynchronous Transfer Mode (ATM) and Frame Relay, Ethernet was not originally designed for interfacing with SONET/SDH. Ethernet data packets need to be framed and encapsulated into a SONET/SDH frame for transport across the SONET/SDH network. This framing and encapsulation process is known as POS.

Figure 21-1 Ethernet to POS Process on ONS Node



ONS Ethernet cards all use POS. The Ethernet frame comes into the card on a standard Fast Ethernet or Gigabit Ethernet port and is processed through the ONS Ethernet card’s framing mechanism and encapsulated into a POS frame. When the POS frame exits, the ONS Ethernet card in a POS circuit, this circuit is treated as any other SONET circuit (STS) or SDH circuit (VC) in the ONS node. It is cross-connected and rides the SONET/SDH signal out the port of an optical card and across the SONET/SDH network.

The destination of the POS circuit is an ONS Ethernet card or other device that supports a POS interface. The POS frames received by the destination card have the data packets stripped out and processed into Ethernet frames. The Ethernet frames are then sent to a standard Ethernet port of the ONS Ethernet card and transmitted onto an Ethernet network.

The G-Series, CE-Series, and E-Series (configured in port-mapper mode) ONS Ethernet cards map this SONET/SDH or POS circuit directly to one of the card’s Ethernet ports. The ML-Series and E-Series (configured in EtherSwitch mode) cards include the POS port as a switchport in a switching fabric that includes the standard Ethernet ports on the card.

POS Interoperability

In addition to POS circuits between Ethernet cards of the same family, POS circuits between some Ethernet cards of different families are possible. The Cisco Transport Controller (CTC) circuit creation wizard shows available interoperable Ethernet cards under the destination card options, when a specific Ethernet card type is chosen as the circuit creation source card. You cannot mix circuits from an SDH node with circuits from a SONET node. POS circuits can be created between the mapper-type cards and the switch-type ONS Ethernet cards.

For Ethernet card POS interoperability, three main POS port characteristics must match:

- POS encapsulation
- CRC size
- Framing Mode

The CRC size option does not need to match on the two endpoints when using GFP-F framing mode.

All Ethernet cards do not interoperate or support all the POS port characteristic options. The following two tables list the interoperable Ethernet cards and characteristics. [Table 21-1](#) lists this information for cards supporting and configured with high-level data link control (HDLC) framing mode.

[Table 21-2](#) lists this information for cards supporting and configured with frame-mapped generic framing procedure (GFP-F) framing mode. With [Table 21-2](#) and GFP-F framing, the word LEX is used to represent standard mapped Ethernet over GFP-F according to ITU-T G.7041. Under GFP-F framing, the Cisco IOS CLI also uses this lex keyword to represent standard mapped Ethernet over GFP-F according to ITU-T G.7041.

Table 21-1 ONS SONET/SDH Ethernet Card Interoperability under HDLC Framing with Encapsulation Type and CRC

	Port-mapped E-Series (ONS 15327)¹	Port-mapped E-Series (ONS 15454 SONET/SDH)¹	G-Series (All Platforms)	ML-Series (ONS 15454 SONET/SDH)	ML-Series (ONS 15310-CL/ONS 15310-MA)	CE-Series (All Platforms)
Port-mapped E-Series (ONS 15327)	Proprietary LEX (CRC 16)	Proprietary	Not compatible	LEX (CRC 16) ²	Not compatible	Not compatible
Port-mapped E-Series (ONS 15454 SONET/SDH)	Proprietary	Proprietary	Not compatible	Not compatible	Not compatible	Not compatible
G-Series (All Platforms)	Not compatible	Not compatible	LEX (CRC 16) LEX (CRC 32)	LEX (CRC 16) LEX (CRC 32)	LEX (CRC 32)	LEX (CRC 32)
ML-Series (ONS 15454 SONET/SDH)	LEX (CRC 16) ²	Not compatible	LEX (CRC 16) LEX (CRC 32)	LEX (CRC 16) LEX (CRC 32) Cisco HDLC PPP/BCP	LEX (CRC 32)	LEX (CRC 32)
ML-Series (ONS 15310-CL/ONS 15310-MA)	Not compatible	Not compatible	LEX (CRC 32)	LEX (CRC 32)	LEX (CRC 32)	LEX (CRC 32)
CE-Series (All Platforms)	Not compatible	Not compatible	LEX (CRC 32)	LEX (CRC 32)	LEX (CRC 32)	LEX (CRC 32)

1. E-Series cards in EtherSwitch modes do not interoperate with other ONS Ethernet card types.
2. The ML-Series card needs special Inter Packet Gap (IPG) configuration to interoperate with the ONS 15327 E-Series card in port-mapped mode.

Table 21-2 ONS SONET/SDH Ethernet Card Interoperability under GFP-F Framing with Encapsulation Type

	ML-Series (ONS 15454)	ML-Series (ONS 15310)	CE-Series (All Platforms)
ML-Series (ONS 15454 SONET/SDH)	LEX (CRC 32) Cisco HDLC (CRC 32) PPP/BCP (CRC 32) IEEE 802.17b	LEX (CRC 32) Cisco HDLC (CRC 32) PPP/BCP (CRC 32)	LEX (CRC 32)
ML-Series (ONS 15310-CL/ ONS 5310-MA)	LEX (CRC 32) Cisco HDLC (CRC 32) PPP/BCP (CRC 32)	LEX (CRC 32 or None) Cisco HDLC (CRC 32 or None) PPP/BCP (CRC 32 or None)	LEX (CRC 32 or None)
CE-Series (All Platforms)	LEX (CRC 32)	LEX (CRC 32 or None)	LEX (CRC 32 or None)

**Note**

Cisco proprietary RPR requires LEX encapsulation on all ML-Series cards. IEEE 802.17 RPR is not configurable and uses IEEE 802.17b encapsulation.

**Note**

When over GFP-F, it is standard Mapped Ethernet over GFP-F according to ITU-T G.7041.

GFP-F framing is only supported on nodes running Software Release 5.0 and later. The ML100T-12 and ML1000-2 cards also require field programmable gate array (FPGA) version 4.0 or later for GFP-F framing.

POS Encapsulation Types

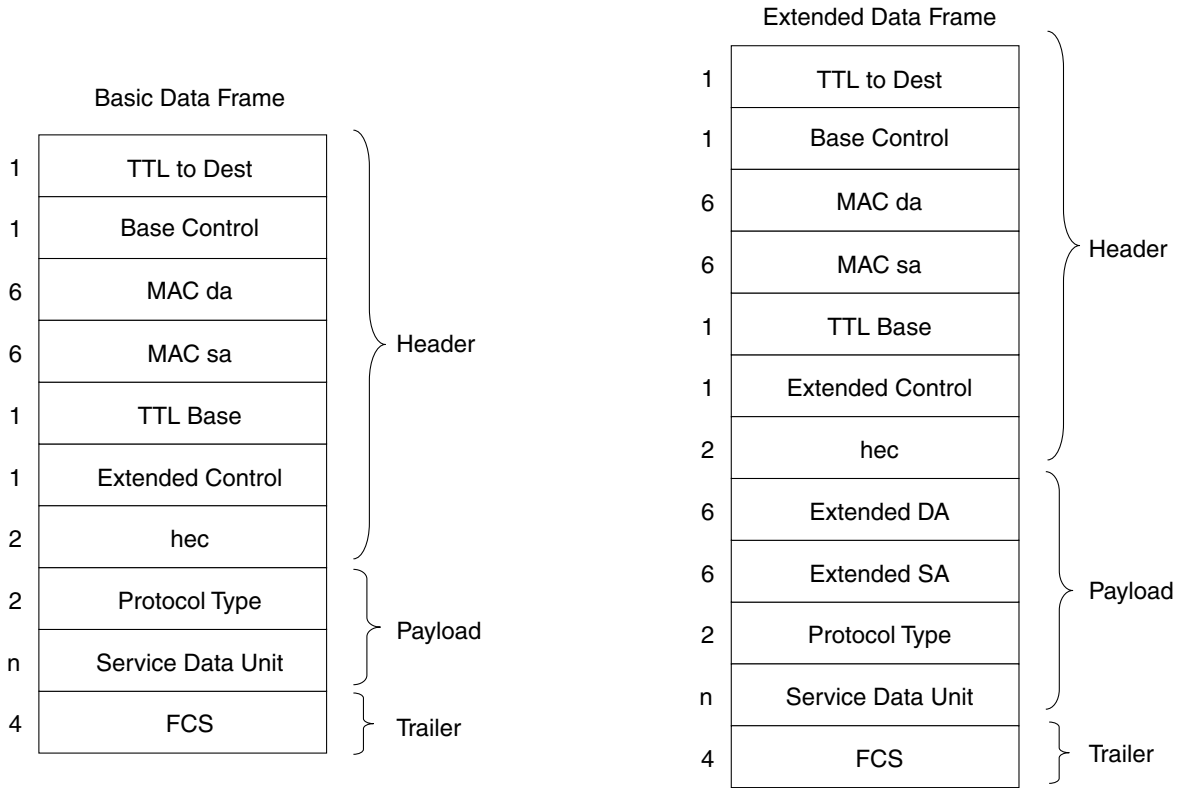
The ONS Ethernet cards support five POS encapsulation methods: Cisco Ethernet-over-SONET LEX (LEX), Cisco HDLC, Point-to-Point Protocol/Bridging Control Protocol (PPP/BCP), IEEE 802.17b, and E-Series proprietary. The ONS Ethernet source card and destination card must be configured with the same POS encapsulation to interoperate. All ONS Ethernet cards do not interoperate or support all types of encapsulation.

IEEE 802.17b

IEEE 802.17b encapsulation is the set encapsulation when the ML-Series card mode is 802.17. It is only supported on the ONS 15454 and ONS 15454 SDH ML-Series cards in Release 7.2 and later.

[Figure 21-2](#) illustrates the IEEE 802.17b extended data frame used by the ML-Series card. It is used with bridging. For comparison, the IEEE 802.17 basic data frame for IP only networks is also shown. The extended data frame adds an extended destination address and extended source address to the basic data frame.

Figure 21-2 RPR Data Frames



151965

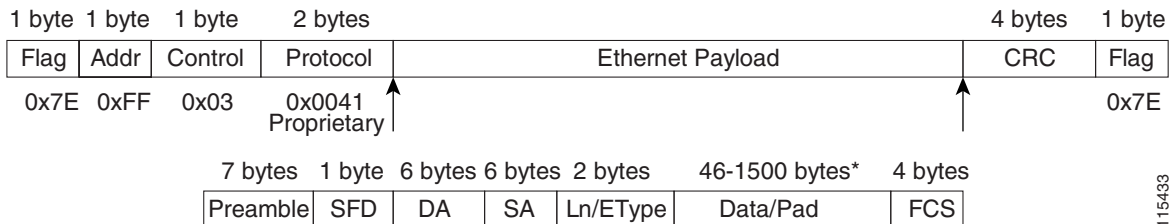
LEX

The Cisco EoS LEX is the primary encapsulation of ONS Ethernet cards. This encapsulation is used under HDLC framing, and the protocol field is set to the values specified in Internet Engineering Task Force (IETF) Request For Comments (RFC) 1841. Under GFP-F framing, the Cisco IOS CLI also uses the keyword lex. With GFP-F framing, the lex keyword is used to represent standard mapped Ethernet over GFP-F according to ITU-T G.7041.

Figure 21-3 illustrates EoS LEX under HDLC framing.

LEX is supported by all the ONS Ethernet cards, except the ONS 15454 and ONS 15454 SDH E-Series cards.

Figure 21-3 LEX Under HDLC Framing

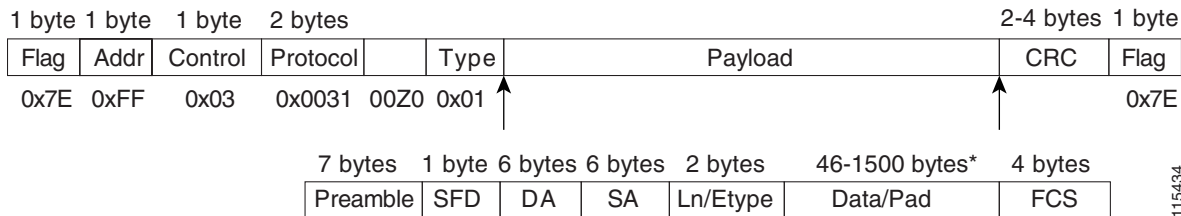


115433

PPP/BCP

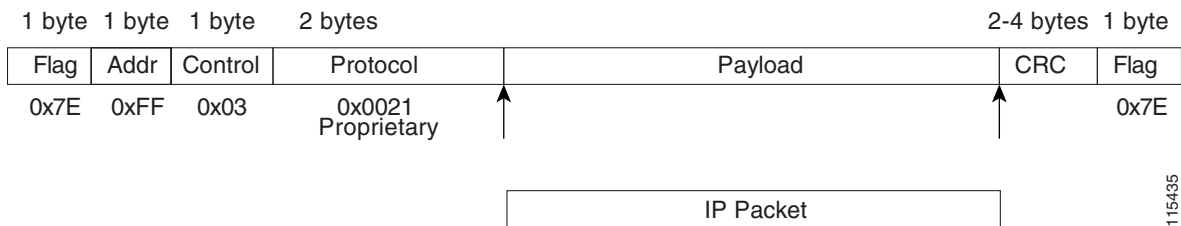
The PPP encapsulation is a standard implementation of RFC 2615 (PPP-over-SONET/SDH), and provides a standard implementation of RFC 3518 (BCP) to provide the transmission of 802.1Q tagged and untagged Ethernet frames over SONET. [Figure 21-4](#) illustrates BCP.

Figure 21-4 BCP Under HDLC Framing



In some framing modes, the ONS 15454/ONS 15454 SDH ML-Series card supports routing functions. When this card POS port is configured to support routing with the PPP encapsulation, the IP packets are mapped into the HDLC frames that use the standard 0x0021 protocol code point. [Figure 21-5](#) illustrates PPP.

Figure 21-5 PPP Frame Under HDLC Framing

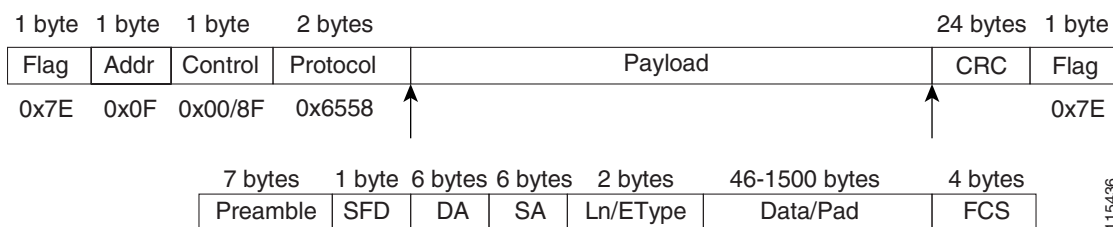


Cisco HDLC

Cisco HDLC is a Cisco-standard mapping of packets into a serial interface. This encapsulation can be used to connect the interface on an ML-Series card to a POS interface on Cisco HDLC-compliant routers and switches.

When used to carry IP packets, the same HDLC frame structure is used, however the protocol field is set to 0x0800, and the payload contains the IP packet. [Figure 21-6](#) illustrates Cisco HDLC.

Figure 21-6 Cisco HDLC Under HDLC Framing



E-Series Proprietary

The E-Series uses a proprietary HDLC-like encapsulation that is incompatible with LEX, Cisco HDLC, or PPP/BCP. This proprietary encapsulation prevents the E-Series from interoperating with other ONS Ethernet cards.

In Release 5.0 and later, the ONS 15327 E-Series card, E10/100-4, supports LEX encapsulation with a 16-bit CRC as well as the original proprietary E-Series encapsulation.

POS Framing Modes

The framing mode is the type of framing mechanism employed by the ONS Ethernet card to frame and encapsulate data packets into a POS signal. These data packets were originally encapsulated in Ethernet frames that entered the standard Fast Ethernet or Gigabit Ethernet interfaces of the ONS Ethernet card. All ONS Ethernet cards support HDLC framing. ML-Series and CE-Series cards also offer GFP-F framing mode.

HDLC Framing

HDLC is one of the most popular Layer 2 protocols. The framing mechanism used by the HDLC protocol, HDLC framing, is employed by a variety of other protocols, including POS on the ONS Ethernet cards. The HDLC framing mechanism is detailed in the IETF's RFC 1662, "PPP in HDLC-like Framing."

The HDLC frame uses the zero insertion/deletion process (commonly known as bit stuffing) to ensure that the bit pattern of the delimiter flag does not occur in the fields between flags. The HDLC frame is synchronous and therefore relies on the physical layer to provide a method of clocking and synchronizing the transmission and reception of frames.

GFP-F Framing

GFP defines a standard-based mapping of different types of services onto SONET/SDH. The ML-Series and CE-Series support frame-mapped GFP (GFP-F), which is the PDU-oriented client signal adaptation mode for GFP. GFP-F maps one variable length data packet onto one GFP packet.

GFP is composed of common functions and payload specific functions. Common functions are those shared by all payloads. Payload-specific functions are different depending on the payload type. GFP is detailed in the ITU recommendation G.7041.

POS Characteristics of Specific ONS Ethernet Cards

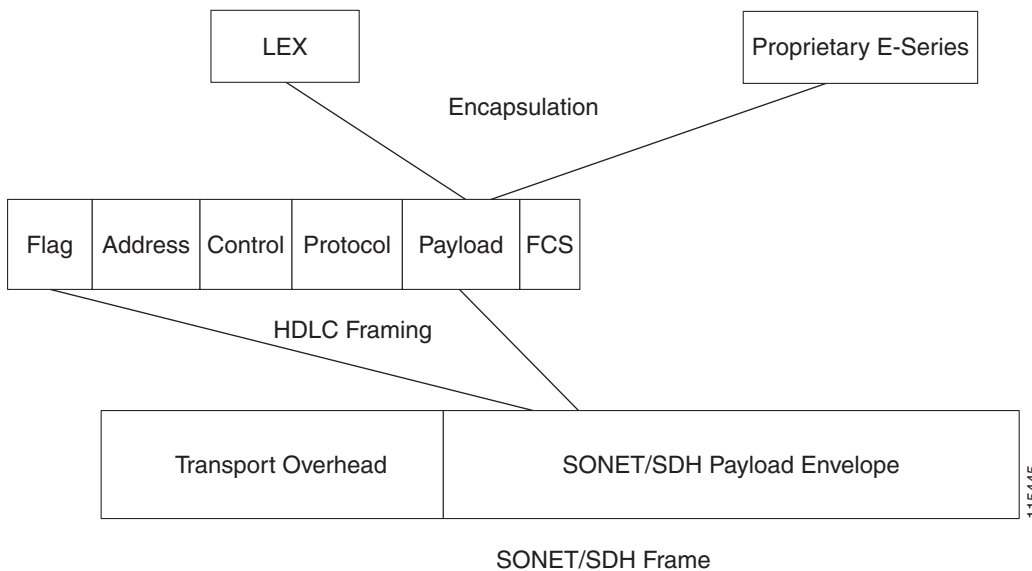
The following sections list and illustrate the various framing and encapsulation options supported by specific ONS Ethernet cards.

ONS 15327 E-10/100-4 Framing and Encapsulation Options

For Software Release 5.0 and later, the E-10/100-4 card on the ONS 15327, configured in port-mapped mode, offers the choice of configuring LEX or the original proprietary E-Series encapsulation. When configured for LEX encapsulation, the ONS 15327 E-Series card interoperates with ML-Series cards. When the E-10/100-4 is configured in EtherSwitch mode, it is restricted to the original proprietary E-Series encapsulation. The E-Series card on the ONS 15327 is restricted to a 16-bit CRC. [Figure 21-7](#) illustrates ONS 15327 E-Series framing and encapsulation.

Refer to the *ONS 15327 Procedure Guide* for port provisioning procedures.

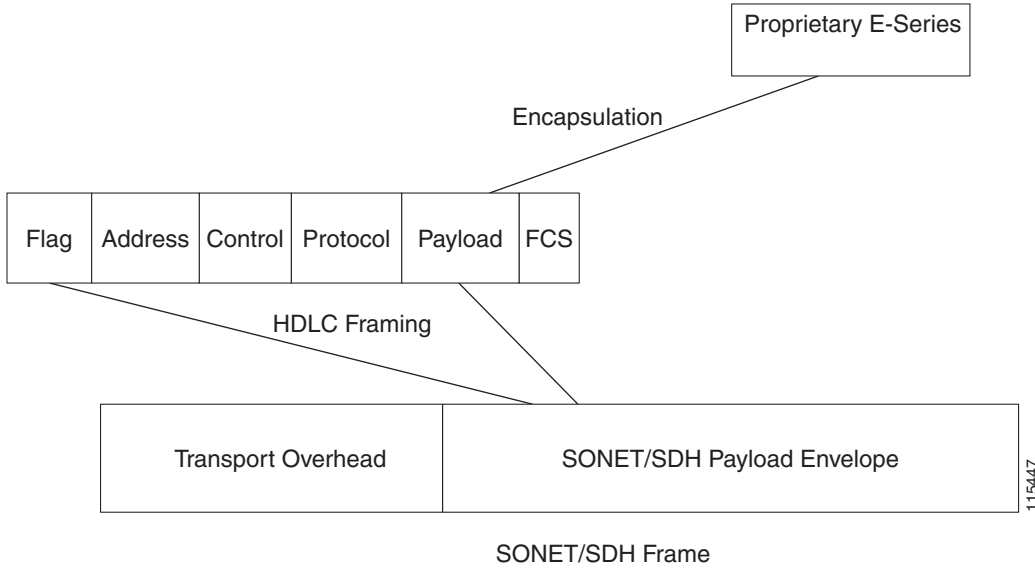
Figure 21-7 ONS 15327 E-Series Encapsulation and Framing Options



ONS 15454 and ONS 15454 SDH E-Series Framing and Encapsulation Options

LEX is not available on the ONS 15454 or ONS 15454 SDH E-Series cards. These cards are limited to the original proprietary E-Series encapsulation, which does not allow POS interoperability with non E-Series cards. [Figure 21-8](#) illustrates ONS 15454 and ONS 15454 SDH E-Series framing and encapsulation.

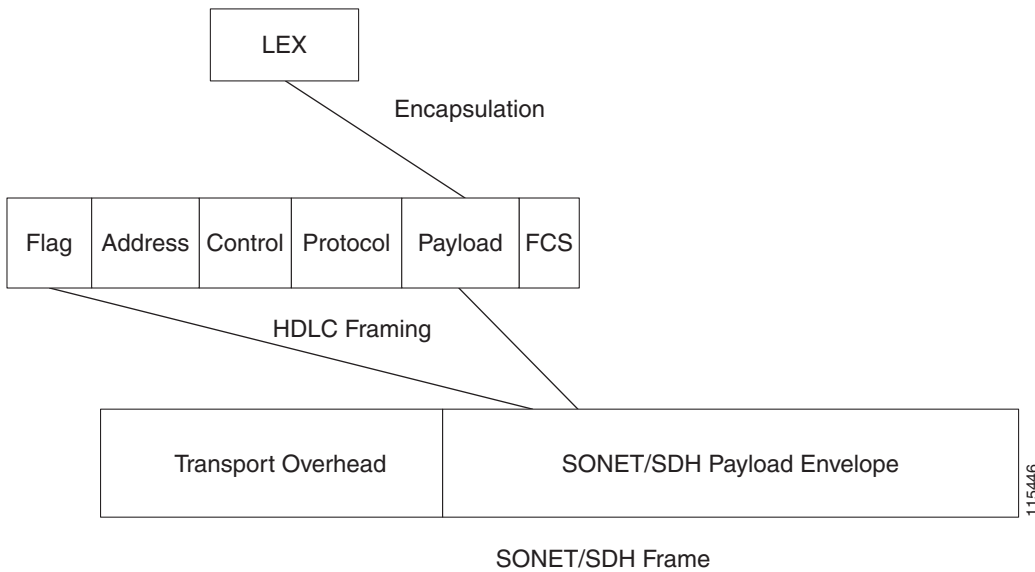
Figure 21-8 ONS 15454 and ONS 15454 SDH E-Series Encapsulation and Framing Options



G-Series Encapsulation and Framing

The G-Series cards are supported on the ONS 15454, ONS 15454 SDH, and ONS 15327 platforms. They support LEX encapsulation and HDLC framing. There are no other POS framing modes or encapsulation options on this card. [Figure 21-9](#) illustrates G-Series encapsulation and framing.

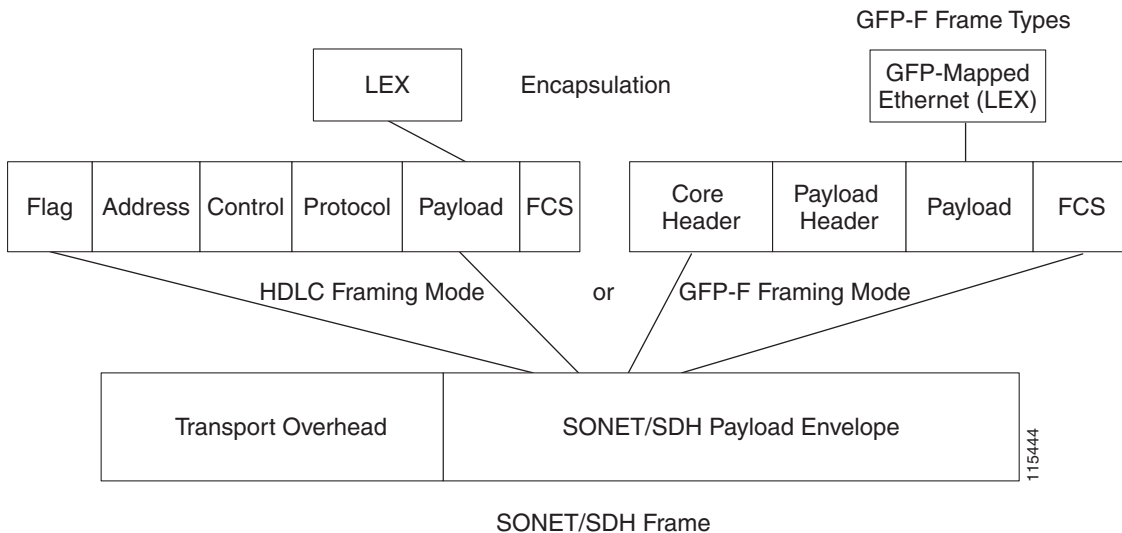
Figure 21-9 ONS G-Series Encapsulation and Framing Options



ONS 15454, ONS 15454 SDH, ONS 15310-CL, and and ONS 15310-MA CE-Series Cards Encapsulation and Framing

CE-100T-8 cards are available for the ONS 15454, ONS 15454 SDH, ONS 15310-CL, and ONS 15310-MA platforms. CE-1000-4 cards are available for the ONS 15454 and ONS 15454 SDH platforms. They support HDLC Framing and GFP-F framing. Under the GFP-F or HDLC framing mode, only LEX encapsulation is supported. [Figure 21-10](#) illustrates CE-Series framing and encapsulation.

Figure 21-10 ONS CE-100T-8 and ONS CE-1000-4 Encapsulation and Framing Options



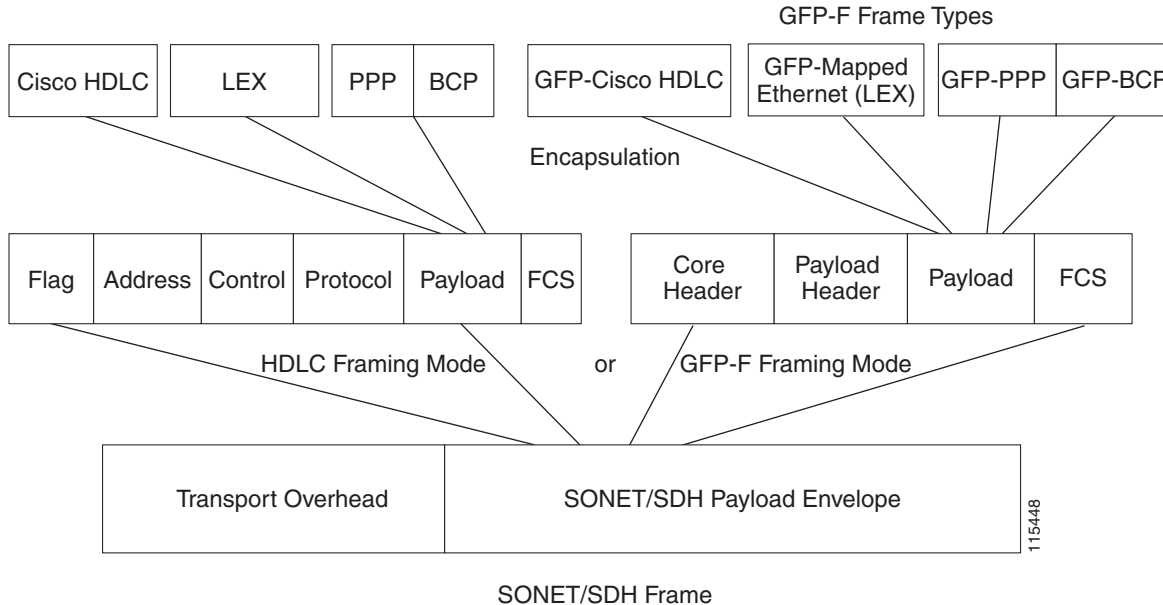
ONS 15310 ML-100T-8 Encapsulation and Framing

The ML-100T-8 card on the ONS 15310 supports HDLC framing and GFP-F framing. Under the HDLC framing mode, LEX is supported. Under the GFP-F framing mode, LEX, Cisco HDLC, and PPP/BCP encapsulation are supported. LEX encapsulation is also the encapsulation for Cisco proprietary RPR on the ML-Series card. Cisco proprietary RPR requires LEX encapsulation in either framing mode.

ONS 15454 and ONS 15454 SDH ML-Series Protocol Encapsulation and Framing

The ML-Series card on the ONS 15454 and ONS 15454 SDH supports HDLC framing and GFP-F framing. Under both the HDLC framing mode and the GFP-F framing mode, LEX, Cisco HDLC, and PPP/BCP encapsulation is supported. LEX encapsulation is also the encapsulation for Cisco proprietary RPR on the ML-Series card. Cisco proprietary RPR requires LEX encapsulation in either framing mode. 802.17b encapsulation is the set encapsulation in IEEE 802.17b compliant RPR, which is only supported in GFP-F framing. [Figure 21-11](#) illustrates the ONS 15454 and ONS 15454 SDH framing and encapsulation options.

Figure 21-11 ML-Series Card Framing and Encapsulation Options



Ethernet Clocking Versus SONET/SDH Clocking

Ethernet clocking is asynchronous. IEEE 802.3 clock tolerance allows some links in a network to be as much as 200 ppm (parts or bits per million) slower than other links (0.02%). A traffic stream sourced at line rate on one link may traverse other links which are 0.02% slower. A fast source clock, or slow intermediate clocks, may limit the end-to-end throughput to only 99.98% of the source link rate.

Traditionally, Ethernet is a shared media that is under utilized except for brief bursts which may combine from multiple devices to exceed line-rate at an aggregation point. Due to this utilization model, the asynchronous clocking of Ethernet has been acceptable. Some Service Providers accustomed to loss-less TDM transport may find the 99.98% throughput guarantee of Ethernet surprising.

Clocking enhancements on ONS Ethernet cards, excluding the E-Series cards, ensure Ethernet transmit rates that are at worst 50 ppm slower than the fastest compliant source clock, ensuring a worst-case clocking loss of 50 ppm - a 99.995% throughput guarantee. In many cases, the card's clock will be faster than the source traffic clock, and line-rate traffic transport will have zero loss. Actual results will depend on clock variation of the traffic source transmitter.



E-Series and G-Series Ethernet Operation



Note

The terms “Unidirectional Path Switched Ring” and “UPSR” may appear in Cisco Literature. These terms do not refer to using Cisco ONS 15xxx products in a unidirectional path switched ring configuration. Rather, these terms, as well as “Path Protected Mesh Network” and “PPMN,” refer generally to Cisco’s path protection feature, which may be used in any topological network configuration. Cisco does not recommend using its path protection feature in any particular topological network configuration.

This chapter covers the operation of the E-Series and G-Series Ethernet cards. E-Series and G-Series cards are supported on the ONS 15454, ONS 15454 SDH, and ONS 15327. Provisioning is done through Cisco Transport Controller (CTC) or Transaction Language One (TL1). Cisco IOS is not supported on the E-Series or G-Series cards.

For Ethernet card specifications, refer to the *Cisco ONS 15454 Reference Manual*, *Cisco ONS 15454 SDH Reference Manual*, or *Cisco ONS 15327 Reference Manual*. For step-by-step Ethernet card circuit configuration procedures, refer to the *Cisco ONS 15454 Procedure Guide*, *Cisco ONS 15454 SDH Procedure Guide*, or the *Cisco ONS 15327 Procedure Guide*. Refer to the *Cisco ONS SONET TL1 Command Guide* or *Cisco ONS 15454 SDH TL1 Command Guide* for TL1 provisioning commands.

Chapter topics include:

- [G-Series Application, page 22-1](#)
- [G-Series Circuit Configurations, page 22-6](#)
- [G-Series Gigabit Ethernet Transponder Mode, page 22-8](#)
- [E-Series Application, page 22-13](#)
- [E-Series Circuit Configurations, page 22-23](#)
- [Remote Monitoring Specification Alarm Thresholds, page 22-27](#)

G-Series Application

The G-Series cards reliably transport Ethernet and IP data across a SONET/SDH backbone. The G-Series cards on the ONS 15454 and ONS 15454 SDH map up to four Gigabit Ethernet ports onto a SONET/SDH transport network and provide scalable and provisionable transport bandwidth at signal levels up to STS-48c/VC4-16 per card. The G-Series card on the ONS 15327 maps two Gigabit Ethernet ports. The G-Series cards provide line rate forwarding for all Ethernet frames (unicast, multicast, and broadcast) and can be configured to support Jumbo frames (defined as a maximum of 10,000 bytes). The G-Series cards incorporate features optimized for carrier-class applications such as:

- High availability (HA), including hitless (< 50 ms) performance with software upgrades and all types of SONET/SDH equipment protection switches
- Hitless reprovisioning
- Support of Gigabit Ethernet traffic at full line rate
- Full TL1-based provisioning capability
- Serviceability options including enhanced port states, terminal and facility loopback, and J1 path trace
- SONET/SDH-style alarm support
- Ethernet performance monitoring (PM) and remote monitoring (RMON) functions

The G-Series cards allow you to provision and manage an Ethernet private line service like a traditional SONET or SDH line. G-Series card applications include providing carrier-grade transparent LAN services (TLS), 100-Mbps Ethernet private line services (when combined with an external 100-Mb Ethernet switch with Gigabit uplinks), and high-availability transport.

On the ONS 15454 or ONS 15327, the card maps a single Ethernet port to a single STS circuit. You can independently map the four ports on a G-Series card to any combination of STS-1, STS-3c, STS-6c, STS-9c, STS-12c, STS-24c, and STS-48c circuit sizes, provided that the sum of the circuit sizes that terminate on a card do not exceed STS-48c.

On the ONS 15454 SDH, the cards map a single Ethernet port to a single STM circuit. You can independently map the four ports on the G-Series card to any combination of VC4, VC4-2c, VC4-3c, VC4-4c, VC4-8c, and VC4-16c circuit sizes, provided the sum of the circuit sizes that terminate on a card do not exceed VC4-16c.

To support a Gigabit Ethernet port at full line rate, an STS/VC4 circuit with a capacity greater or equal to 1 Gbps (bidirectional 2 Gbps) is needed. An STS-24c/VC4-8c is the minimum circuit size that can support a Gigabit Ethernet port at full line rate. A G-Series card supports a maximum of two ports at full line rate.

The G-Series transmits and monitors the J1 Path Trace byte in the same manner as OC-N/STM-N cards. For more information, see the appropriate platform reference book, either *ONS 15454 Reference Manual*, *ONS 15454 SDH Reference Manual*, or *ONS 15327 Reference Manual*.

**Note**

The G-Series uses LEX encapsulation. LEX is standard high-level data link control (HDLC) framing over SONET/SDH as described in RFC 1622 and RFC 2615, with the Point-to-Point Protocol (PPP) field set to the value specified in RFC 1841. For more information on LEX, see [Chapter 21, “POS on ONS Ethernet Cards.”](#)

G1K-4 and G1000-4 Comparison

The G1K-4 and the G1000-4 cards comprise the ONS 15454/ONS 15454 SDH G-Series. The G1K-4 is the hardware equivalent of the earlier G1000-4.

When installed in ONS 15454s running Software Release 3.4 and earlier, both cards require the XC10G card to operate. However, when installed on an ONS 15454 running Software R4.0 and later, the G1K-4 card is not limited to installation in ONS 15454s with XC10G cards but can also be installed in ONS 15454s with XC and XCVT cards. When used with XC and XCVT cards on an ONS 15454 running Software R4.0 and later, the G1K-4 is limited to Slots 5, 6, 12, and 13.

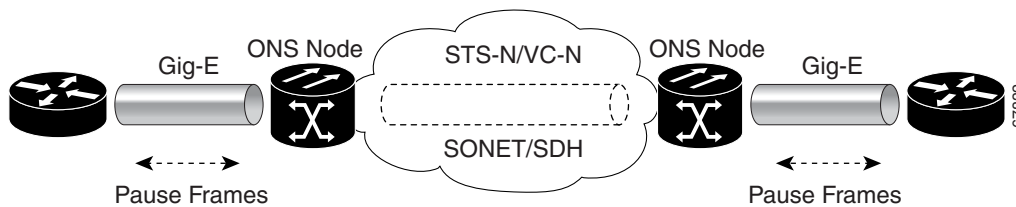
These constraints do not apply to a G-Series card configured for Gigabit Ethernet Transponder Mode; see the [“G-Series Gigabit Ethernet Transponder Mode”](#) section on page 22-8 for more information.

Software R4.0 and later identifies G1K-4 cards at physical installation. Software R3.4 and earlier identifies both G1000-4 and G1K-4 cards as G1000-4 cards at physical installation.

G-Series Example

Figure 22-1 shows a G-Series application. In this example, data traffic from the Gigabit Ethernet port of a high-end router travels across the ONS node's point-to-point circuit to the Gigabit Ethernet port of another high-end router.

Figure 22-1 Data Traffic on a G-Series Point-to-Point Circuit



The G-Series cards carry any Layer 3 protocol that can be encapsulated and transported over Gigabit Ethernet, such as IP or IPX. The data is transmitted on the Gigabit Ethernet fiber into the standard Cisco Gigabit Interface Converter (GBIC) on an ONS 15454 or ONS 15454 SDH G-Series card or into the standard Small Form-factor Pluggable (SFP) modules on an ONS 15327 G-Series card. The G-Series card transparently maps Ethernet frames into the SONET/SDH payload by multiplexing the payload onto an OC-N/STM-N card. When the payload reaches the destination node, the process is reversed and the data is transmitted from the standard Cisco GBIC or SFP in the destination G-Series card onto the Gigabit Ethernet fiber.

The G-Series cards discard certain types of erroneous Ethernet frames rather than transport them over SONET/SDH. Erroneous Ethernet frames include corrupted frames with cycle redundancy check (CRC) errors and under-sized frames that do not conform to the minimum 64-byte length Ethernet standard. The G-Series cards forward valid frames unmodified over the SONET/SDH network. Information in the headers is not affected by the encapsulation and transport. For example, packets with formats that include IEEE 802.1Q information will travel through the process unaffected.

IEEE 802.3z Flow Control and Frame Buffering

The G-Series supports IEEE 802.3z flow control and frame buffering to reduce data traffic congestion. To prevent over-subscription, 512 KB of buffer memory is available for the receive and transmit channels on each port. When the buffer memory on the Ethernet port nears capacity, the G-Series uses IEEE 802.3z flow control to transmit a pause frame to the source at the opposite end of the Gigabit Ethernet connection.

The pause frame instructs the source to stop sending packets for a specific period of time. The sending station waits the requested amount of time before sending more data. Figure 22-1 illustrates pause frames being sent and received by G-Series cards and attached switches.

The G-Series cards have symmetric flow control. Symmetric flow control allows the G-Series cards to respond to pause frames sent from external devices and to send pause frames to external devices. Prior to Software R4.0, flow control on the G-Series cards was asymmetric, meaning that the cards sent pause frames and discarded received pause frames.

Software Release 5.0 and later features separate CTC provisioning of autonegotiation and flow control. A failed autonegotiation results in a link down.

When both autonegotiation and flow control are enabled, the G-Series card proposes symmetrical flow control to the attached Ethernet device. Flow control may be used or not depending on the result of the autonegotiation.

If autonegotiation is enabled but flow control is disabled, then the G-Series proposes no flow control during the autonegotiation. This negotiation succeeds only if the attached device agrees to no flow control.

If autonegotiation is disabled, then the attached device's provisioning is ignored. The G-Series card's flow control is enabled or disabled based solely on the G-Series card's provisioning.

This flow-control mechanism matches the sending and receiving device throughput to that of the bandwidth of the STS/VC circuit. For example, a router might transmit to the Gigabit Ethernet port on the G-Series card. This particular data rate might occasionally exceed 622 Mbps, but the SONET circuit assigned to the G-Series port might be only STS-12c (622 Mbps). In this example, the ONS 15454 sends out a pause frame and requests that the router delay its transmission for a certain period of time. With flow control and a substantial per-port buffering capability, a private line service provisioned at less than full line rate capacity (STS-24c) is efficient because frame loss can be controlled to a large extent. The same concept applies to the ONS 15454 SDH or ONS 15327.

The G-Series cards have flow control threshold provisioning, which allows a user to select one of three watermark (buffer size) settings: default, low latency, or custom. Default is the best setting for general use and was the only setting available prior to Software R4.1. Low latency is good for sub-rate applications, such as voice-over-IP (VoIP) over an STS-1. For attached devices with insufficient buffering, best effort traffic, or long access line lengths, set the G-Series to a higher latency.

The custom setting allows you to specify the buffer size of Flow Ctrl Lo and Flow Ctrl Hi thresholds. The range is 1 to 511 units, where 1 unit is equal to 192 bytes. Make sure that the value of Flow Ctrl Lo is lesser than Flow Ctrl Hi with a difference of at least 160 units between the two values to ensure packets are not dropped. The flow control high setting is the watermark for sending the Pause On frame to the attached Ethernet device; this frame signals the device to temporarily stop transmitting. The flow control low setting is the watermark for sending the Pause Off frame, which signals the device to resume transmitting. With a G-Series card, you can only enable flow control on a port if autonegotiation is enabled on the device attached to that port.

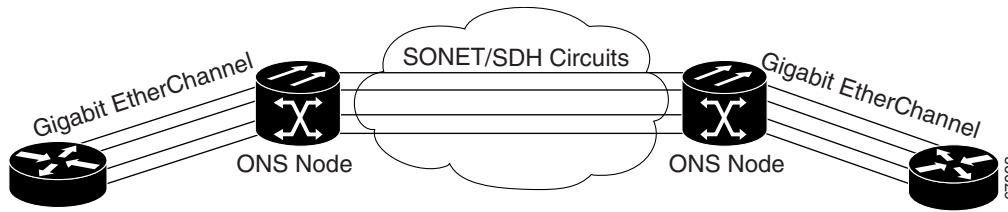

Note

External Ethernet devices with autonegotiation configured to interoperate with G-Series cards running releases prior to Software R4.0 do not need to change autonegotiation settings when interoperating with G-Series cards running Software R4.0 and later.

Gigabit EtherChannel/IEEE 802.3ad Link Aggregation

The G-Series supports all forms of link aggregation technologies including GEC, which is a Cisco proprietary standard, and the IEEE 802.3ad standard. The end-to-end link integrity feature of the G-Series allows a circuit to emulate an Ethernet link. This allows all flavors of Layer 2 and Layer 3 rerouting to work correctly with the G-Series. [Figure 22-2](#) illustrates G-Series GEC support.

Figure 22-2 G-Series Gigabit EtherChannel (GEC) Support



Although the G-Series cards do not actively run GEC, they support the end-to-end GEC functionality of attached Ethernet devices. If two Ethernet devices running GEC connect through G-Series cards to an ONS network, the ONS SONET/SDH side network is transparent to the EtherChannel devices. The EtherChannel devices operate as if they are directly connected to each other. Any combination of G-Series parallel circuit sizes can be used to support GEC throughput.

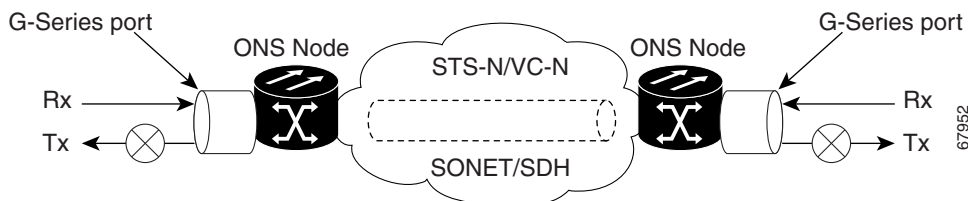
GEC provides line-level active redundancy and protection (1:1) for attached Ethernet equipment. It can also bundle parallel G-Series data links together to provide more aggregated bandwidth. Spanning Tree Protocol (STP) operates as if the bundled links are one link and permits GEC to utilize these multiple parallel paths. Without GEC, STP permits only a single nonblocked path. GEC can also provide G-Series card-level protection or redundancy because it can support a group of ports on different cards (or different nodes) so that if one port or card has a failure, traffic is rerouted over the other port or card.

The end-to-end Ethernet link integrity feature can be used in combination with Gigabit EtherChannel (GEC) capability on attached devices. The combination provides an Ethernet traffic restoration scheme that has a faster response time than alternate techniques such as spanning tree rerouting, yet is more bandwidth efficient because spare bandwidth does not need to be reserved.

Ethernet Link Integrity Support

The G-Series supports end-to-end Ethernet link integrity (Figure 22-3). This capability is integral to providing an Ethernet private line service and correct operation of Layer 2 and Layer 3 protocols on the attached Ethernet devices. End-to-end Ethernet link integrity essentially means that if any part of the end-to-end path fails, the entire path fails. Failure of the entire path is ensured by turning off the transmit lasers at each end of the path. The attached Ethernet devices recognize the disabled transmit laser as a loss of carrier and consequently an inactive link.

Figure 22-3 End-to-End Ethernet Link Integrity Support



Note

Some network devices can be configured to ignore a loss of carrier condition. If a device configured to ignore a loss of carrier condition attaches to a G-Series card at one end, alternative techniques (such as use of Layer 2 or Layer 3 keep-alive messages) are required to route traffic around failures. The response time of such alternate techniques is typically much longer than techniques that use link state as indications of an error condition.

As shown in [Figure 22-3](#), a failure at any point of the path causes the G-Series card at each end to disable its Tx transmit laser, which causes the devices at both ends to detect a link down. If one of the Ethernet ports is administratively disabled or set in loopback mode, the port is considered a “failure” for the purposes of end-to-end link integrity because the end-to-end Ethernet path is unavailable. The port “failure” also disables both ends of the path.

Enhanced State Model for Gigabit Ethernet Ports

For Release 5.0 and later, the G-Series supports the Enhanced State Model (ESM) for the Gigabit Ethernet ports, as well as for the SONET/SDH circuit. For more information on ESM, see the “Enhanced State Model” appendix in the *ONS 15454 SONET Reference Manual*, *ONS 15327 SONET Reference Manual* or *ONS 15454 SDH Reference Manual*.

The Gigabit Ethernet ports can be set to the ESM service states including the automatic in-service administrative state (IS, AINS). IS, AINS initially puts the port in the out of service automatic, automatic in-service (OOS-AU, AINS) state. In this service state, alarm reporting is suppressed, but traffic is carried and loopbacks are allowed. After the soak period passes, the port changes to in-service, not reported (IS-NR). Raised fault conditions, whether their alarms are reported or not, can be retrieved on the CTC Conditions tab or by using the TL1 RTRV-COND command.

Two Ethernet port alarms/conditions, CARLOSS and TPTFAIL, can prevent the port from going into service. This occurs even though alarms are suppressed when a G-Series circuit is provisioned with the Gigabit Ethernet ports set to IS, AINS state. Because the G-Series link integrity function is active and ensures that the Tx transmit lasers at either end are not enabled until all SONET and Ethernet errors along the path are cleared. As long as the link integrity function keeps the end-to-end path down both ports will have at least one of the two conditions needed to suppress the AINS to IS transition so the ports will remain in the AINS state with alarms suppressed.

ESM also applies to the SONET/SDH circuits of the G-Series card. If the SONET/SDH circuit had been setup in IS, AINS state and the Ethernet error occurs before the circuit transitions to IS, then link integrity will also prevent the circuit transition to the IS state until the Ethernet port errors are cleared at both ends. Service state will be OOS-AU,AINS as long as the admin state is IS,AINS. Once there are no Ethernet or SONET errors link integrity enables the Gigabit Ethernet TX transmit lasers at each end. Simultaneously, the AINS countdown begins as normal. If no additional conditions occur during the time period each port transitions to the IS, NR state. During the AINS countdown the soak time remaining is available in CTC and TL1. The AINS soaking logic restarts from the beginning if a condition re-appears during the soak period.

A SONET/SDH circuit provisioned in the IS, AINS state remains in the initial OOS state until the Gigabit Ethernet ports on either end of the circuit transition to the IS, NR state. The SONET/SDH circuit transports Ethernet traffic and count statistics when link integrity turns on the Gigabit Ethernet port Tx transmit lasers, regardless of whether this AINS to IS transition is complete.

G-Series Circuit Configurations

This section explains G-Series point-to-point circuits and manual cross-connects. Ethernet manual cross-connects allow you to bridge non-ONS SONET/SDH network segments.

G-Series Point-to-Point Ethernet Circuits

G-Series cards support point-to-point circuit configurations (Figure 22-4). Circuits are configured through CTC in the same manner as SONET or SDH line cards. G-Series cards support ESM for circuit service states.

On the ONS 15454 and ONS 15327, provisionable SONET circuit sizes are STS 1, STS 3c, STS 6c, STS 9c, STS 12c, STS 24c, and STS 48c. On the ONS 15454 SDH, provisionable SDH circuits are VC4, VC4-2c, VC4-3c, VC4-4c, VC4-8c, and VC4-16c. Each Ethernet port maps to a unique STS/VC circuit on the G-Series card.

Figure 22-4 G-Series Point-to-Point Circuit



The G-Series supports any combination of up to four circuits from the list of valid circuit sizes; however, the circuit sizes can add up to no more than 48 STSs or 16 VC4s.

Due to hardware constraints, the card imposes an additional restriction on the combinations of circuits that can be dropped onto a G-Series card. These restrictions are transparently enforced by the node, and you do not need to keep track of restricted circuit combinations.

When a single STS-24c/VC4-8c terminates on a card, the remaining circuits on that card can be another single STS-24c/VC4-8c or any combination of circuits of STS-12c/VC4-4c size or less that adds up to no more than 12 STSs or 4 VC4s (that is, a total of 36 STSs or 12 VC4s on the card).

If STS-24c/VC4-8c circuits are not being dropped on the card, the full bandwidth can be used with no restrictions (for example, using either a single STS-48c/VC4-16c or four STS-12c/VC4-4c circuits).

Because the STS-24c/VC4-8c restriction applies only when a single STS-24c/VC4-8c circuit is dropped; this restriction's impact can be minimized. Group the STS-24c/VC4-8c circuits together on a card separate from circuits of other sizes. The grouped circuits can be dropped on other G-Series cards.



Note

The G-Series uses STS/VC cross-connects only. No VT level cross-connects are used.



Caution

G-Series cards do not connect with ONS 15454 E-Series cards. For more information on interoperability, see [Chapter 21, "POS on ONS Ethernet Cards."](#)

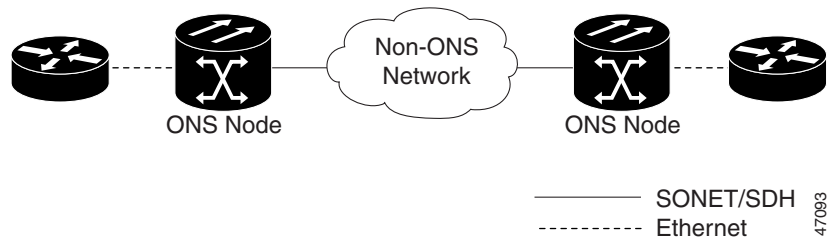
G-Series Manual Cross-Connects

ONS nodes require end-to-end CTC visibility between nodes for normal provisioning of Ethernet circuits. When other vendors' equipment sits between ONS nodes, Simple Network Management Protocol/Target Identifier Address Resolution Protocol (OSI/TARP)-based equipment does not allow tunneling of the ONS node TCP/IP-based data communications channel (DCC). To circumvent inconsistent DCCs, the Ethernet circuit must be manually cross connected to an STS/VC channel using the non-ONS network. Manual cross-connects allow an Ethernet circuit to run from ONS node to ONS node while utilizing the non-ONS network (Figure 22-5).

**Note**

In this section, “cross-connect” and “circuit” have the following meanings: Cross-connect refers to the connections that occur within a single ONS node to allow a circuit to enter and exit an ONS node. Circuit refers to the series of connections from a traffic source (where traffic enters the ONS node network) to the drop or destination (where traffic exits an ONS node network).

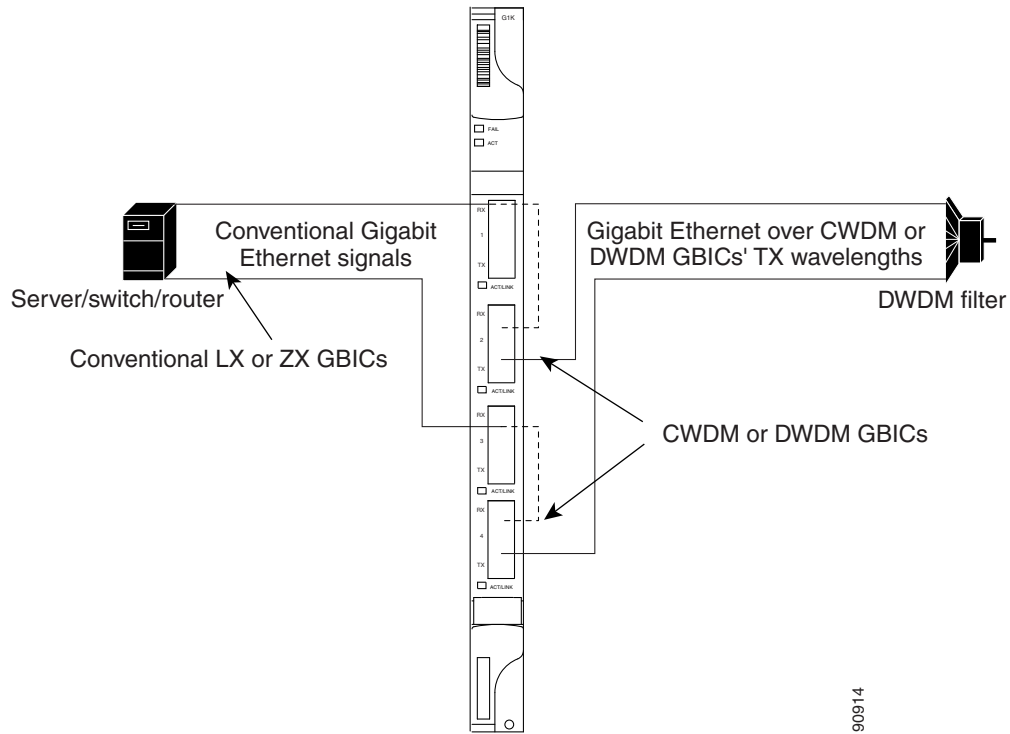
Figure 22-5 G-Series Manual Cross-Connects



G-Series Gigabit Ethernet Transponder Mode

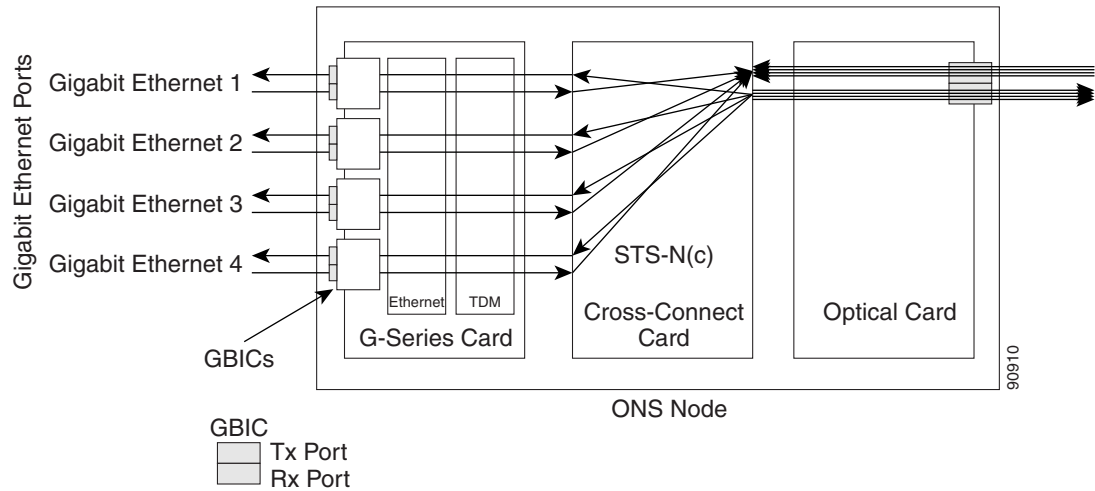
The ONS 15454 and ONS 15454 SDH G-Series cards can be configured as transponders. ONS 15327 G-Series cards cannot be configured as transponders. Transponder mode can be used with any G-Series-supported GBIC (SX, LX, ZX, coarse wavelength division multiplexing [CWDM], or dense wavelength division multiplexing [DWDM]). [Figure 22-6](#) shows a card level overview of a transponder mode application.

Figure 22-6 Card Level Overview of G-Series One-Port Transponder Mode Application



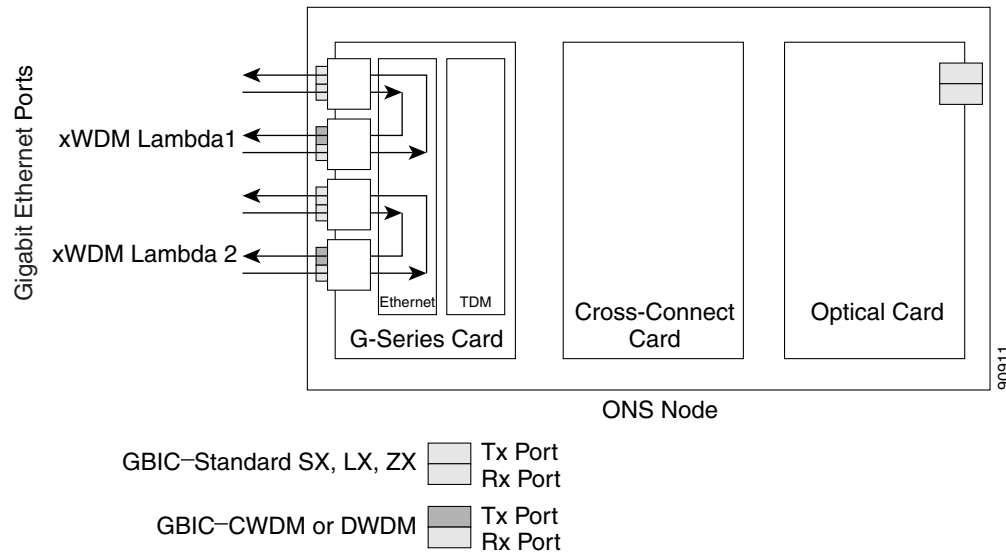
A G-Series card configured as a transponder operates quite differently than a G-Series card configured for SONET/SDH. In SONET/SDH configurations, the G-Series card receives and transmits Gigabit Ethernet traffic out the Ethernet ports and GBICs on the front of the card. This Ethernet traffic is multiplexed on and off the SONET/SDH network through the cross-connect card and the optical card (Figure 22-7).

Figure 22-7 G-Series in Default SONET/SDH Mode



In transponder mode, the G-Series Ethernet traffic never comes into contact with the cross-connect card or the SONET/SDH network, but stays internal to the G-Series card and is routed back to a GBIC on that card (Figure 22-8).

Figure 22-8 G-Series Card in Transponder Mode (Two-Port Bidirectional)



A G-Series card can be configured either for transponder mode or as the SONET/SDH default. When any port is provisioned in transponder mode, the card is in transponder mode and no SONET/SDH circuits can be configured until every port on the card goes back to SONET/SDH mode. To provision G-Series ports for transponder mode, refer to the *Cisco ONS 15454 Procedure Guide* or the *Cisco ONS 15454 SDH Procedure Guide*.

All SONET/SDH circuits must be deleted before a G-Series card can be configured in transponder mode. An ONS 15454 or ONS 15454 SDH can host the G-Series card configured in transponder mode in any or all of the 12 traffic slots and supports a maximum of 24 bidirectional or 48 unidirectional lambdas.

A G-Series card configured as a transponder can be in one of three modes:

- Two-port bidirectional transponder mode
- One-port bidirectional transponder mode
- Two-port unidirectional transponder mode

Two-Port Bidirectional Transponder Mode

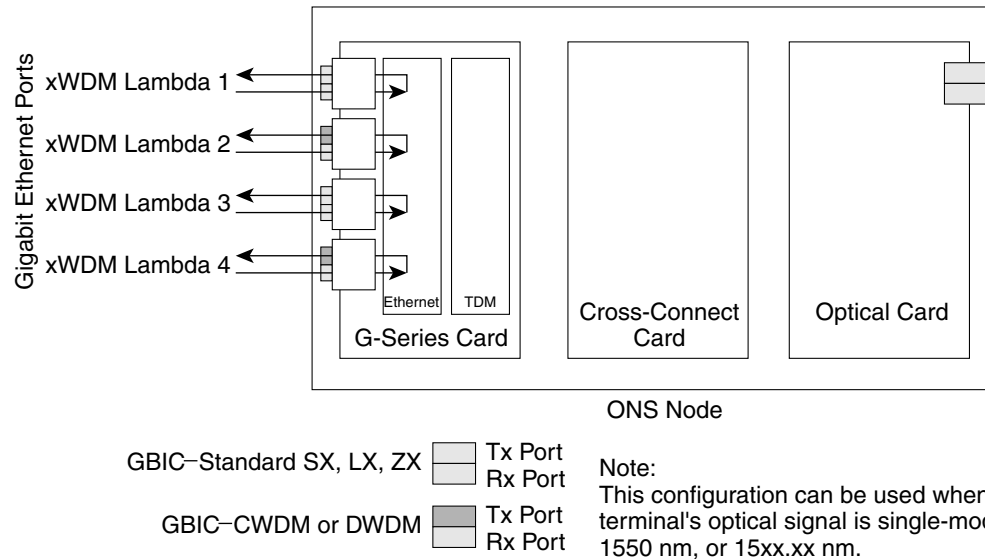
Two-port bidirectional transponder mode maps the transmitted and received Ethernet frames of one G-Series card port into the transmitted and received Ethernet frames of another port (Figure 22-8). Transponder bidirectional port mapping can be done from any port to any other port on the same card.

One-Port Bidirectional Transponder Mode

One-port bidirectional transponder mode maps the Ethernet frames received at a port out the transmitter of the same port (Figure 22-9). This mode is similar to two-port bidirectional transponder mode except that a port is mapped only to itself instead of to another port. Although the data path of the one-port bidirectional transponder mode is identical to that of a facility loopback, the transponder mode is not a maintenance mode and does not suppress non-SONET/SDH alarms, such as loss of carrier (CARLOSS).

This mode can be used for intermediate DWDM signal regeneration and to take advantage of the wide band capability of the CWDM and DWDM GBICs. This allows the node to receive on multiple wavelengths but transmit on a fixed wavelength.

Figure 22-9 One-Port Bidirectional Transponder Mode

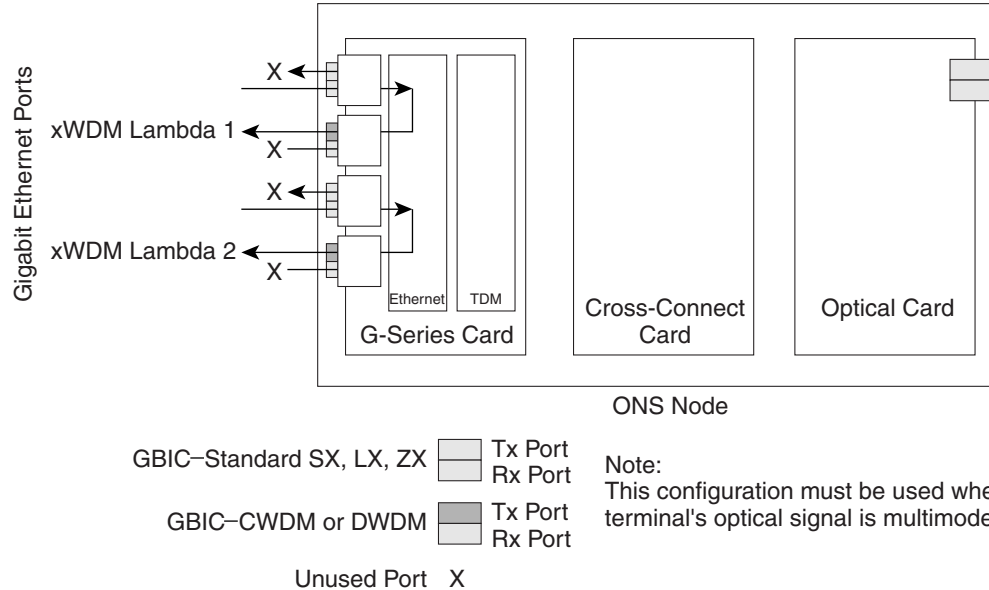


90913

Two-Port Unidirectional Transponder Mode

Ethernet frames received at one port's receiver will be transmitted out the transmitter of another port. This mode is similar to two-port bidirectional transponder mode except only one direction is used (Figure 22-10). One port has to be provisioned as unidirectional transmit only and the other port as unidirectional receive. The port configured as unidirectional transmit ignores any lack of signal on the receive port, so the receive port fiber does not need to be connected. The port configured as unidirectional receive does not turn on the transmit laser, so the transmit port fiber does not need to be connected.

This mode can be used when only one direction needs to be transmitted over CWDM/DWDM, for example, certain video on demand (VoD) applications.

Figure 22-10 Two-Port Unidirectional Transponder

Note:
 This configuration must be used when the client terminal's optical signal is multimode, 850 nm.

G-Series Transponder Mode Characteristics

The operation of a G-Series card in transponder mode differs from a G-Series card in SONET/SDH mode in several ways:

- A G-Series card set to transponder mode will not show up in the CTC list of provisionable cards when the user is provisioning a SONET/SDH circuit.
- G-Series cards set to transponder mode do not require cross-connect cards (for example, XC10G), but do require TCC2/TCC2P cards.
- G-Series ports configured as transponders do not respond to flow control pause frames and pass the pause frames transparently through the card. In SONET/SDH mode, ports can respond to pause frames and do not pass the pause frames through the card.
- There is no TL1 provisioning support for configuring transponder mode. However, transponder mode and port information can be retrieved in the output for the TL1 command RTRV-G1000.
- All SONET/SDH-related alarms are suppressed when a card is in transponder mode.
- There are no slot number or cross-connect restrictions for G1000-4 or G1K-4 cards in transponder mode.
- Facility and terminal loopbacks are not fully supported in unidirectional transponder mode, but are supported in both bidirectional transponder modes.
- Ethernet autonegotiation is not supported and cannot be provisioned in unidirectional transponder mode. Autonegotiation is supported in both bidirectional transponder modes.
- No end-to-end link integrity function is available in transponder mode.

**Note**

In normal SONET/SDH mode, the G-Series cards supports an end-to-end link integrity function. This function causes an Ethernet or SONET/SDH failure to disable and turn the transmitting laser off in the corresponding mapped Ethernet port. In transponder mode, the loss of signal on an Ethernet port has no impact on the transmit signal of the corresponding mapped port.

The operation of a G-Series card in transponder mode is also similar to the operation of a G-Series card in SONET/SDH mode:

- G-Series Ethernet statistics are available for ports in both modes.
- Ethernet port level alarms and conditions are available for ports in both modes.
- Jumbo frame and non-Jumbo frame operation is the same in both modes.
- Collection, reporting, and threshold crossing conditions for all existing counters and PM parameters are the same in both modes.
- Simple Network Management Protocol (SNMP) and RMON support is the same in both modes.

E-Series Application

The ONS 15454, ONS 15454 SDH and ONS 15327 all support E-Series cards. E-Series cards include the E100T-12/E100T-G and the E1000-2/E1000-2-G on the ONS 15454 and ONS 15454 SDH. The E100T-G is the functional equivalent of the earlier E100T-12. The E1000-2-G is the functional equivalent of the earlier E1000-2. An ONS 15454 using XC10G cards requires the G versions (E100T-G or E1000-2-G) of the E-Series Ethernet cards. An ONS 15454 or ONS 15454 SDH supports a maximum of ten E-Series cards. You can insert E-Series Ethernet cards in any multipurpose slot.

The ONS 15327 E-Series card is the E10/100-4. This is the only E-Series card that supports LEX encapsulation configuration, which allows interoperability with ML-Series cards, for more information see [Chapter 21, “POS on ONS Ethernet Cards.”](#)

**Note**

The ONS 15454 and ONS 15454 SDH E-Series cards do not support LEX encapsulation.

E-Series Modes

An E-Series card operates in one of three modes: multcard EtherSwitch group, single-card EtherSwitch, or port-mapped. E-Series cards in multcard EtherSwitch group or single-card EtherSwitch mode support Layer 2 features, including virtual local area networks (VLANs), IEEE 802.1Q, STP, and IEEE 802.1D. Port-mapped mode configures the E-Series to operate as a straight mapper card and does not support these Layer 2 features. Within a node containing multiple E-Series cards, each E-Series card can operate in any of the three separate modes. At the Ethernet card view in CTC, click the **Provisioning > Ether Card** tabs to reveal the card modes.

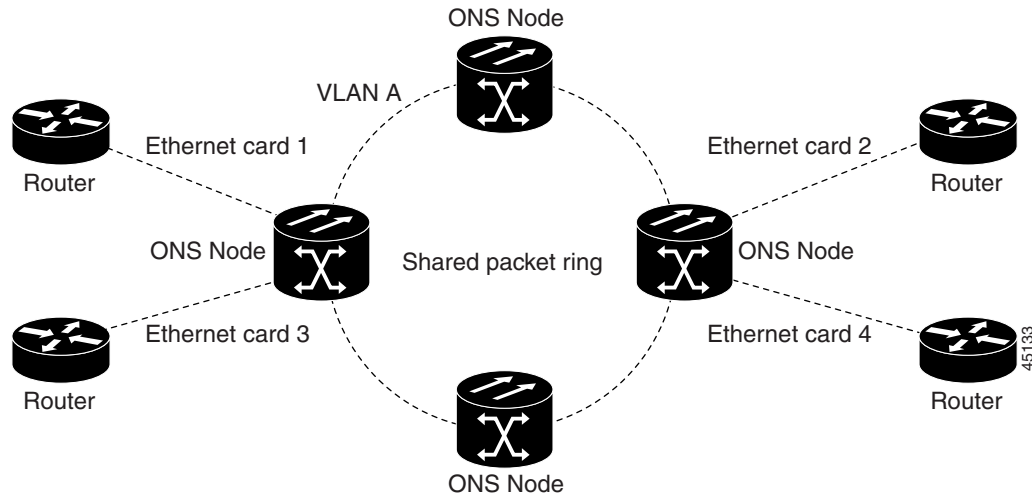
**Note**

Port-mapped mode eliminates issues inherent in other E-Series modes and is detailed in the field notice, “E-Series Ethernet Line Card Packet Forwarding Limitations.”

E-Series Multicard EtherSwitch Group

Multicard EtherSwitch group provisions two or more Ethernet cards to act as a single Layer 2 switch. [Figure 22-11](#) illustrates a multicard EtherSwitch configuration. Multicard EtherSwitch limits bandwidth to STS-6c of bandwidth between two Ethernet circuit points for the ONS 15454, VC4-2c for the ONS 15454 SDH and STS-3c of bandwidth between ONS 15327 E-Series cards, but allows you to add nodes and cards and make a shared packet ring.

Figure 22-11 Multicard EtherSwitch Configuration



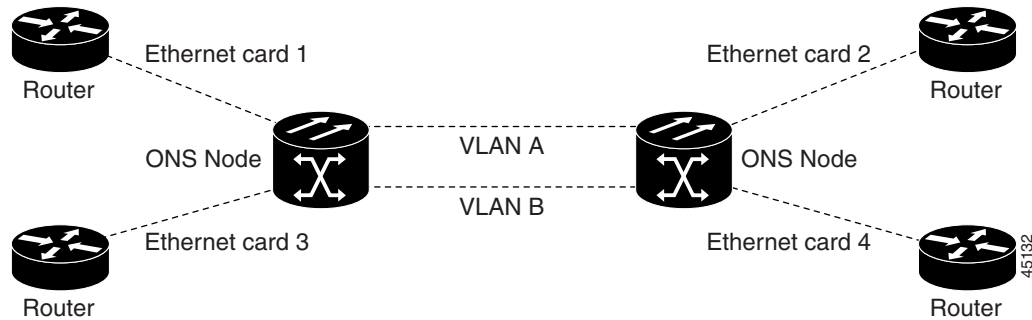
Caution

If you terminate two STS-3c/VC4-2c multicard EtherSwitch circuits on an Ethernet card and later delete the first circuit, also delete the remaining STS-3c/VC4-2c circuit before you provision an STS-1/VC4 circuit to the card. If you attempt to create an STS-1/VC4 circuit after only deleting the first STS-3c/VC4-2c circuit, the STS-1/VC4 circuit will not work and no alarms will indicate this condition. To avoid this situation, delete the second STS-3c/VC4-2c before creating an STS-1/VC4 circuit.

E-Series Single-Card EtherSwitch

On all E-Series cards, Single-card EtherSwitch allows each Ethernet card to remain a single switching entity within the ONS node. [Figure 22-12](#) illustrates a single-card EtherSwitch configuration.

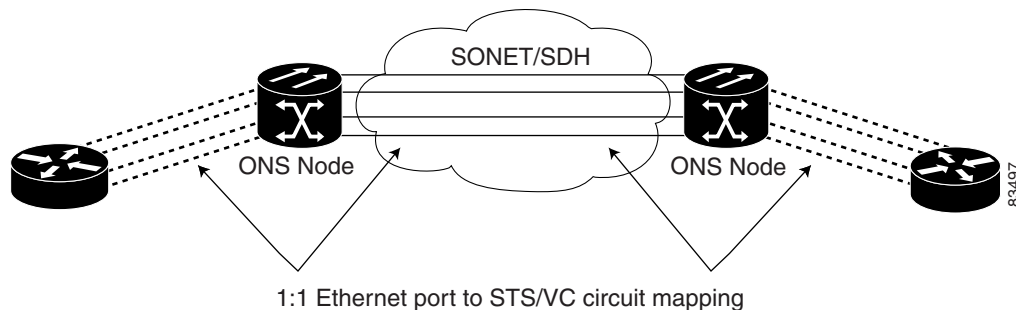
Figure 22-12 Single-Card EtherSwitch Configuration



Port-Mapped (Linear Mapper)

Port-mapped mode, also referred to as linear mapper, configures the E-Series card to map a specific E-Series Ethernet port to one of the card's specific STS/VC circuits (Figure 22-13). Port-mapped mode ensures that Layer 1 transport has low latency for unicast, multicast, and mixed traffic. Ethernet and Fast Ethernet on the E100T-G or E10/100-4 card operate at line-rate speed. Gigabit Ethernet transport is limited to a maximum of 600 Mbps because the E1000-2-G card has a maximum bandwidth of STS-12c/VC4-4c. Ethernet frame sizes up to 1522 bytes are also supported, which allow transport of IEEE 802.1Q tagged frames. The larger maximum frame size of Q-in-Q frames (IEEE 802.1Q in IEEE 802.1Q wrapped frames) is not supported.

Figure 22-13 E-Series Mapping Ethernet Ports to STS/VC Circuits



Port-mapped mode disables Layer 2 functions supported by the E-Series in single-card and multcard mode, including STP, VLANs, and MAC address learning. It significantly reduces the service-affecting time for cross-connect and TCC2/TCC2P card switches.

Port-mapped mode does not support VLANs in the same manner as multcard and single-card mode. The ports of E-Series cards in multcard and single-card mode can join specific VLANs. E-Series cards in port-mapped mode do not have this Layer 2 capability and only transparently transport external VLANs over the mapped connection between ports. An E-Series card in port-mapped mode does not inspect the tag of the transported VLAN, so a VLAN range of 1 through 4096 can be transported in port-mapped mode.

Port-mapped mode does not perform any inspection or validation of the Ethernet frame header. The Ethernet CRC is validated, and any frame with an invalid Ethernet CRC is discarded.

Port-mapped mode also allows the creation of STS/VC circuits between any two E-Series cards, including the E100T-G, E1000-2-G, and the E10/100-4 (the ONS 15327 E-Series card). Port-mapped mode does not allow ONS 15454 E-Series cards to connect to the ML-Series or G-Series cards, but does allow an ONS 15327 E10/100-4 card provisioned with LEX encapsulation to connect to the ML-Series or G-Series cards.

Available Circuit Sizes For E-Series Modes

Table 22-1 shows the circuit sizes available for E-Series modes on the ONS 15454, ONS 15454 SDH, and ONS 15327.

Table 22-1 ONS 15454 and ONS 15327 E-Series Ethernet Circuit Sizes

ONS 15327 E-Series Port-Mapped and Single-Card EtherSwitch	ONS 15327 E-Series Multicard EtherSwitch	ONS 15454 E-Series Port-Mapped and Single-Card EtherSwitch	ONS 15454 E-Series Multicard EtherSwitch	ONS 15454 SDH E-Series Port-Mapped and Single-Card EtherSwitch	ONS 15454 SDH E-Series Multicard EtherSwitch
STS-1	STS-1	STS-1	STS-1	VC4	VC4
STS-3c	STS-3c	STS-3c	STS-3c	VC4-2c	VC4-2c
STS-6c	—	STS-6c	STS-6c	VC4-4c	—
STS-12c	—	STS-12c	—	—	—

Available Total Bandwidth For E-Series Modes

Table 22-2 shows the total bandwidth available for E-Series modes on the ONS 15454, ONS 15454 SDH, and ONS 15327.

Table 22-2 ONS 15454 and ONS 15327 E-Series Total Bandwidth Available

ONS 15327 E-Series Port-Mapped and Single-Card EtherSwitch	ONS 15327 E-Series Multicard EtherSwitch	ONS 15454 E-Series Port-Mapped and Single-Card EtherSwitch	ONS 15454 E-Series Multicard EtherSwitch	ONS 15454 SDH E-Series Port-Mapped and Single-Card EtherSwitch	ONS 15454 SDH E-Series Multicard EtherSwitch
Combined total of STS-12c	Combined total of STS-3c	Combined total of STS-12c	Combined total of STS-6c	Combined total of VC4-4c	Combined total of VC4-2c

E-Series IEEE 802.3z Flow Control

The E100T-G or E10/100-4 (operating in any mode) and the E1000-2-G (operating port-mapped mode) support IEEE 802.3z symmetrical flow control and propose symmetric flow control when autonegotiating with attached Ethernet devices. For flow control to operate, both the E-Series port and the attached Ethernet device must be set to autonegotiation (AUTO) mode. The attached Ethernet device might also need to have flow control enabled. The flow-control mechanism allows the E-Series to respond to pause frames sent from external devices and send pause frames to external devices.

For the E100T-G or E10/100-4 (operating in any mode) and the E1000-2-G (operating port-mapped mode), flow control matches the sending and receiving device throughput to that of the bandwidth of the STS circuit. This same concept applies to the ONS 15454, ONS 15454 SDH and ONS 15327. For example, a router might transmit to the Gigabit Ethernet port on the E-Series in port-mapped mode. The data rate transmitted by the router might occasionally exceed 622 Mbps, but the ONS 15454 circuit assigned to the E-Series port in port-mapped mode is a maximum of STS-12c (622.08 Mbps). In this scenario, the ONS 15454 sends out a pause frame and requests that the router delay its transmission for a certain period of time.

**Note**

To enable flow control between an E-Series in port-mapped mode and a SmartBits test set, manually set Bit 5 of the MII register to 0 on the SmartBits test set. To enable flow control between an E-Series in port-mapped mode and an Ixia test set, select Enable the Flow Control in the Properties menu of the attached Ixia port.

E-Series VLAN Support

You can provision up to 509 VLANs per network with the CTC software. Specific sets of ports define the broadcast domain for the ONS node. The definition of VLAN ports includes all Ethernet and packet-switched SONET/SDH port types. All VLAN IP address discovery, flooding, and forwarding is limited to these ports.

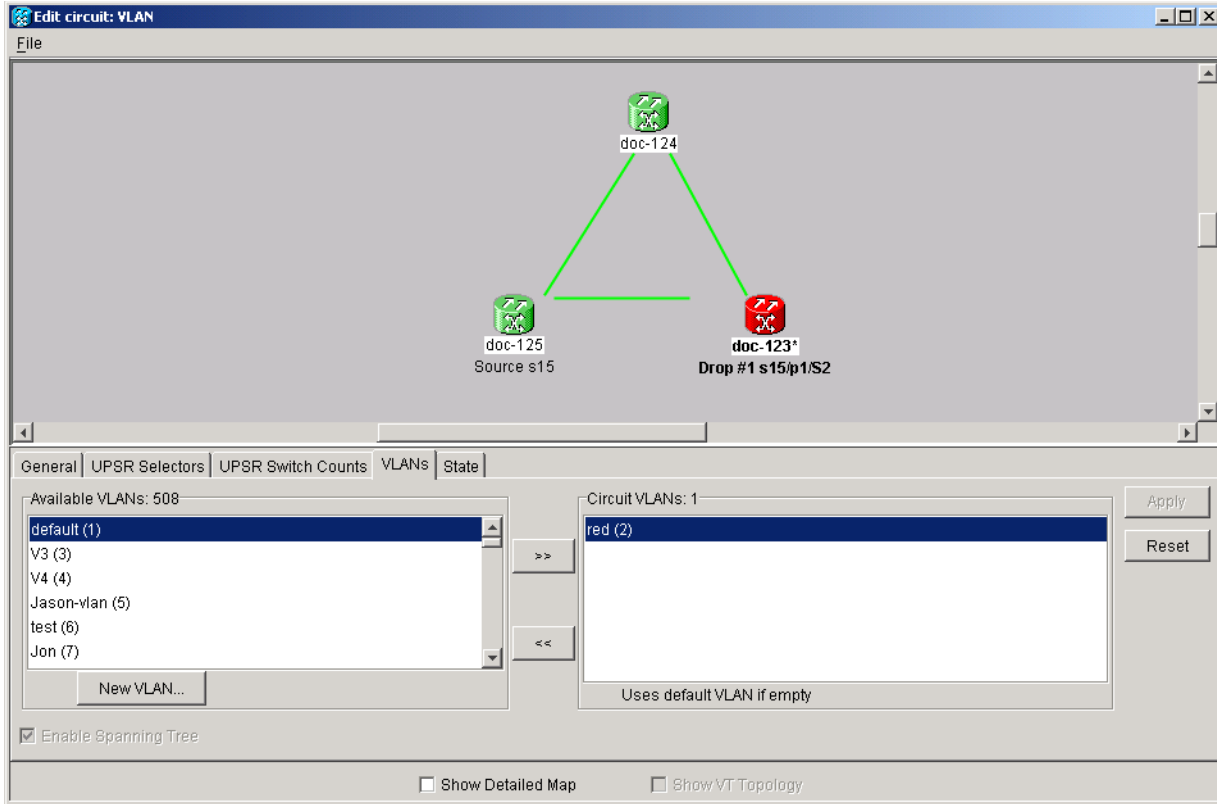
The IEEE 802.1Q-based VLAN mechanism provides logical isolation of subscriber LAN traffic over a common SONET/SDH transport infrastructure. Each subscriber has an Ethernet port at each site, and each subscriber is assigned to a VLAN. Although the subscriber's VLAN data flows over shared circuits, the service appears to the subscriber as a private data transport.

**Note**

Port-mapped mode does not support VLANs.

The number of VLANs used by circuits and the total number of VLANs available for use appears in CTC on the VLAN counter ([Figure 22-14](#)).

Figure 22-14 Edit Circuit Dialog Box Featuring Available VLANs



78632

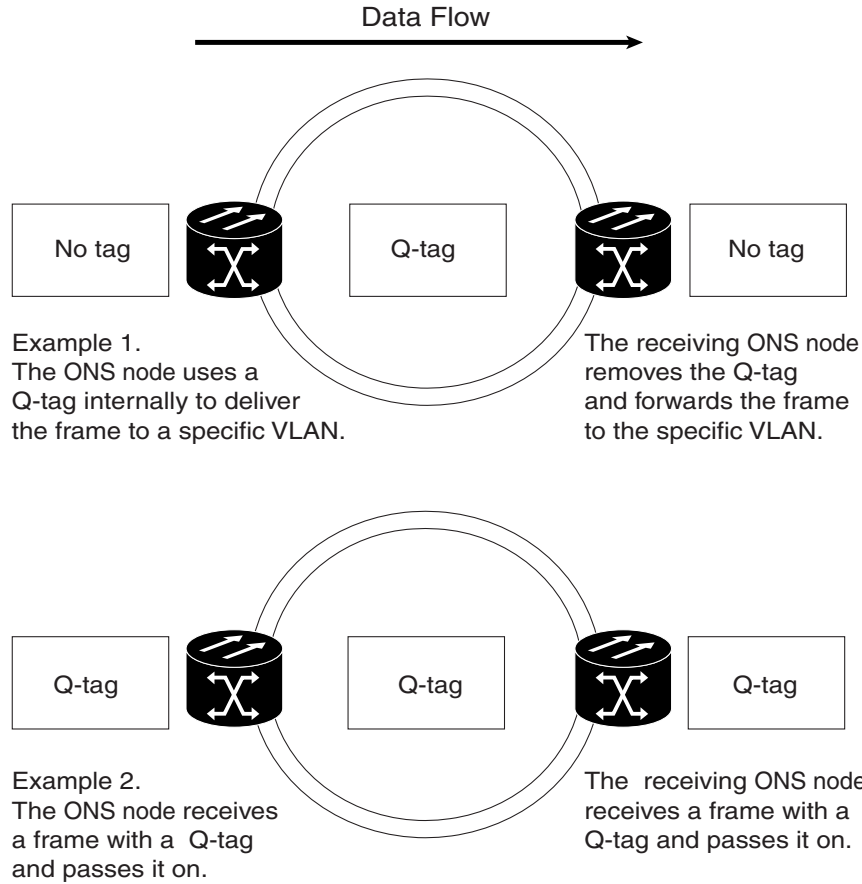
E-Series Q-Tagging (IEEE 802.1Q)

E-Series cards in single-card and multicard mode support IEEE 802.1Q. IEEE 802.1Q allows the same physical port to host multiple IEEE 802.1Q VLANs. Each IEEE 802.1Q VLAN represents a different logical network. E-Series cards in port-mapped mode transport IEEE 802.1Q tags (Q-tags), but do not remove or add these tags.

The ONS node works with Ethernet devices that support IEEE 802.1Q and those that do not support IEEE 802.1Q. If a device attached to an E-Series Ethernet port does not support IEEE 802.1Q, the ONS node uses Q-tags internally only. The ONS node associates these Q-tags with specific ports.

With Ethernet devices that do not support IEEE 802.1Q, the ONS node takes non-tagged Ethernet frames that enter the ONS network and uses a Q-tag to assign the packet to the VLAN associated with the ONS network's ingress port. The receiving ONS node removes the Q-tag when the frame leaves the ONS network (to prevent older Ethernet equipment from incorrectly identifying the IEEE 802.1Q packet as an illegal frame). The ingress and egress ports on the ONS network must be set to Untag for the removal to occur. Untag is the default setting for ONS ports. Example 1 in [Figure 22-15](#) illustrates Q-tag use only within an ONS network.

Figure 22-15 Q-tag Moving Through VLAN



The ONS node uses the Q-tag attached by the external Ethernet devices that support IEEE 802.1Q. Packets enter the ONS network with an existing Q-tag; the ONS node uses this same Q-tag to forward the packet within the ONS network and leaves the Q-tag attached when the packet leaves the ONS network. The entry and egress ports on the ONS network must be set to Tagged for this process to occur. Example 2 in Figure 22-15 illustrates the handling of packets that both enter and exit the ONS network with a Q-tag.

For more information about setting ports to Tagged and Untag, refer to the *Cisco ONS 15454 Procedure Guide*, the *Cisco ONS 15454 SDH Procedure Guide*, or the *Cisco ONS 15327 Procedure Guide*.

Caution

ONS nodes propagate VLANs whenever a node appears on the network view of another node, regardless of whether the nodes are in the same SONET/SDH network or connect through DCC. For example, if two ONS nodes without DCC connectivity belong to the same login node group, VLANs propagate between the two ONS nodes. VLAN propagation happens even though the ONS nodes do not belong to the same SONET/SDH ring.

E-Series Priority Queuing (IEEE 802.1Q)

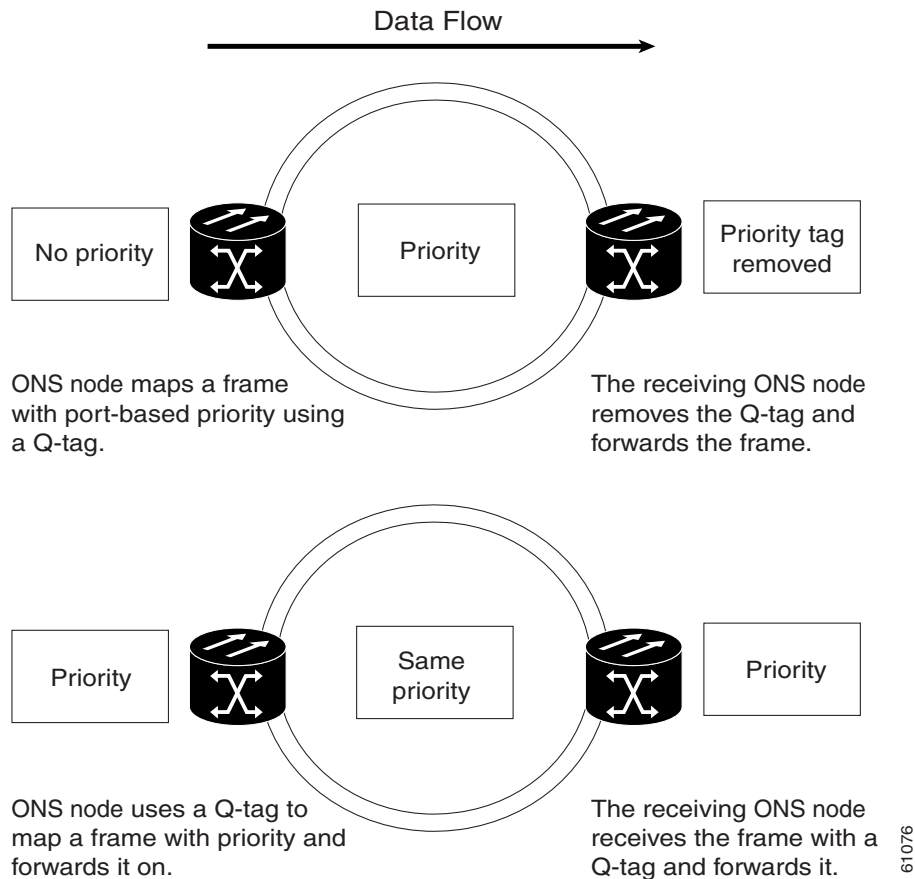
Networks without priority queuing handle all packets on a first-in-first-out (FIFO) basis. Priority queuing reduces the impact of network congestion by mapping Ethernet traffic to different priority levels. The E-Series card supports priority queuing. The E-Series card maps the eight priorities specified in IEEE 802.1Q to two queues, low priority and high priority (Table 22-3).

Table 22-3 Priority Queuing

User Priority	Queue	Allocated Bandwidth
0,1,2,3	Low	30%
4,5,6,7	High	70%

Q-tags carry priority queuing information through the network (Figure 22-16).

Figure 22-16 Priority Queuing Process



The ONS node uses a “leaky bucket” algorithm to establish a weighted priority. A weighted priority, as opposed to a strict priority, gives high-priority packets greater access to bandwidth, but does not totally preempt low-priority packets. During periods of network congestion, about 70 percent of bandwidth goes to the high-priority queue and the remaining 30 percent goes to the low-priority queue. A network that is too congested will drop packets.



Note IEEE 802.1Q was formerly known as IEEE 802.1P.



Note E-Series cards in port-mapped mode and G-Series cards do not support priority queuing (IEEE 802.1Q).

E-Series Spanning Tree (IEEE 802.1D)

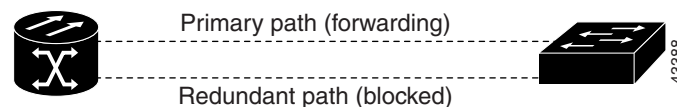
The E-Series operates IEEE 802.1D Spanning Tree Protocol (STP). The E-Series card supports common STPs on a per-circuit basis up to a total of eight STP instances. It does not support per-VLAN STP. In single-card mode, STP can be disabled or enabled on a per-circuit basis during circuit creation. Disabling STP will preserve the number of available STP instances.

STP operates over all packet-switched ports including Ethernet and OC-N/STM-N ports. On Ethernet ports, STP is enabled by default but can be disabled. A user can also disable or enable STP on a circuit-by-circuit basis on Ethernet cards configured as single-card EtherSwitch (unstitched) in a point-to-point configuration. However, turning off STP protection on a circuit-by-circuit basis means that the SONET/SDH system is not protecting the Ethernet traffic on this circuit, and the Ethernet traffic must be protected by another mechanism in the Ethernet network. On OC-N/STM-N interface ports, the ONS node activates STP by default, and STP cannot be disabled.

The Ethernet card can enable STP on the Ethernet ports to create redundant paths to the attached Ethernet equipment. STP connects cards so that both equipment and facilities are protected against failure.

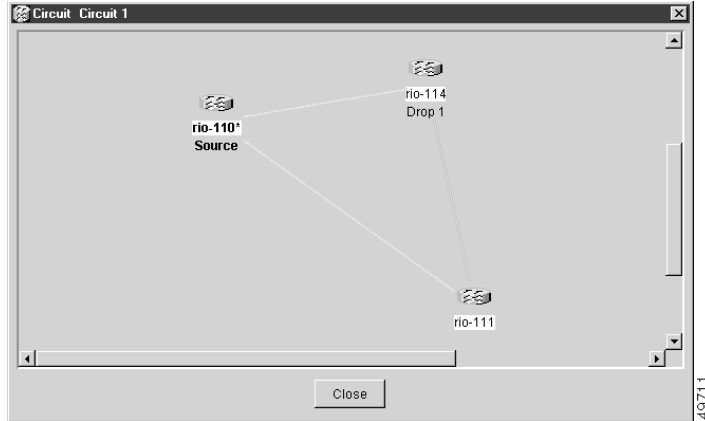
STP detects and eliminates network loops. When STP detects multiple paths between any two network hosts, STP blocks ports until only one path exists between any two network hosts (Figure 22-17). The single path eliminates possible bridge loops. This is crucial for shared packet rings, which naturally include a loop.

Figure 22-17 STP Blocked Path



To remove loops, STP defines a tree that spans all the switches in an extended network. STP forces certain redundant data paths into a standby (blocked) state. If one network segment in the STP becomes unreachable, the STP algorithm reconfigures the STP topology and reactivates the blocked path to reestablish the link. STP operation is transparent to end stations, which do not discriminate between connections to a single LAN segment or to a switched LAN with multiple segments. The ONS node supports one STP instance per circuit and a maximum of eight STP instances per ONS node.

The Circuit window shows forwarding spans and blocked spans on the spanning tree map (Figure 22-18).

Figure 22-18 Spanning Tree Map on Circuit Window**Note**

Green represents forwarding spans and purple represents blocked (protect) spans. If you have a packet ring configuration, at least one span should be purple.

**Caution**

Multiple circuits with STP protection enabled will incur blocking if the circuits traverse a common card and use the same VLAN.

**Note**

E-Series port-mapped mode does not support STP (IEEE 802.1D).

E-Series Multi-Instance Spanning Tree and VLANs

The ONS node can operate multiple instances of STP to support VLANs in a looped topology. You can dedicate separate circuits across the SONET/SDH ring for different VLAN groups. Each circuit runs its own STP to maintain VLAN connectivity in a multi-ring environment.

Spanning Tree on a Circuit-by-Circuit Basis

You can also disable or enable STP on a circuit-by-circuit basis on single-card EtherSwitch E-Series cards in a point-to-point configuration. This feature allows customers to mix spanning tree protected circuits with unprotected circuits on the same card. It also allows two single-card EtherSwitch E-Series cards on the same node to form an intranode circuit.

E-Series Spanning Tree Parameters

Default STP parameters are appropriate for most situations ([Table 22-4](#)). Contact the Cisco Technical Assistance Center (Cisco TAC) before you change the default STP parameters.

Table 22-4 Spanning Tree Parameters

Parameter	Description
BridgeID	ONS node unique identifier that transmits the configuration bridge protocol data unit (BPDU); the bridge ID is a combination of the bridge priority and the ONS node MAC address.
TopoAge	Amount of time in seconds since the last topology change.
TopoChanges	Number of times the STP topology has been changed since the node booted up.
DesignatedRoot	Identifies the STP's designated root for a particular STP instance.
RootCost	Identifies the total path cost to the designated root.
RootPort	Port used to reach the root.
MaxAge	Maximum time that received-protocol information is retained before it is discarded.
HelloTime	Time interval, in seconds, between the transmission of configuration BPDUs by a bridge that is the spanning tree root or is attempting to become the spanning tree root.
HoldTime	Minimum time period, in seconds, that elapses during the transmission of configuration information on a given port.
ForwardDelay	Time spent by a port in the listening state and the learning state.

E-Series Spanning Tree Configuration

To view the spanning tree configuration, at the node view click the **Provisioning > Etherbridge > Spanning Trees** tabs (Table 22-5).

Table 22-5 Spanning Tree Configuration

Column	Default Value	Value Range
Priority	32768	0–65535
Bridge max age	20 seconds	6–40 seconds
Bridge Hello Time	2 seconds	1–10 seconds
Bridge Forward Delay	15 seconds	4–30 seconds

E-Series Circuit Configurations

E-Series Ethernet circuits can link ONS nodes through point-to-point (straight), shared packet ring, or hub-and-spoke configurations. Two nodes usually connect with a point-to-point configuration. More than two nodes usually connect with a shared packet ring configuration or a hub-and-spoke configuration. Ethernet manual cross-connects allow you to cross connect individual Ethernet circuits to an STS/VC channel on the ONS node optical interface and also to bridge non-ONS SONET/SDH network segments. To configure E-Series circuits, refer to the *Cisco ONS 15454 Procedure Guide*, the *Cisco ONS 15454 SDH Procedure Guide*, or the *Cisco ONS 15327 Procedure Guide*.

E-Series Circuit Protection

Different combinations of E-Series circuit configurations and SONET/SDH network topologies offer different levels of E-Series circuit protection. Table 22-6 details the available protection.

Table 22-6 Protection for E-Series Circuit Configurations

Configuration	Path Protection (SNCP)	BLSR (MS-SPRing)	1 + 1
Point-to-point multcard EtherSwitch	None	SONET/SDH	SONET/SDH
Point-to-point single-card EtherSwitch	SONET/SDH	SONET/SDH	SONET/SDH
Point-to-point port-mapped mode	SONET/SDH	SONET/SDH	SONET/SDH
Shared packet ring multcard EtherSwitch	STP	SONET/SDH	SONET/SDH
Common control card switch	STP	STP	STP



Note

Before making Ethernet connections, choose an STS/STM circuit size.



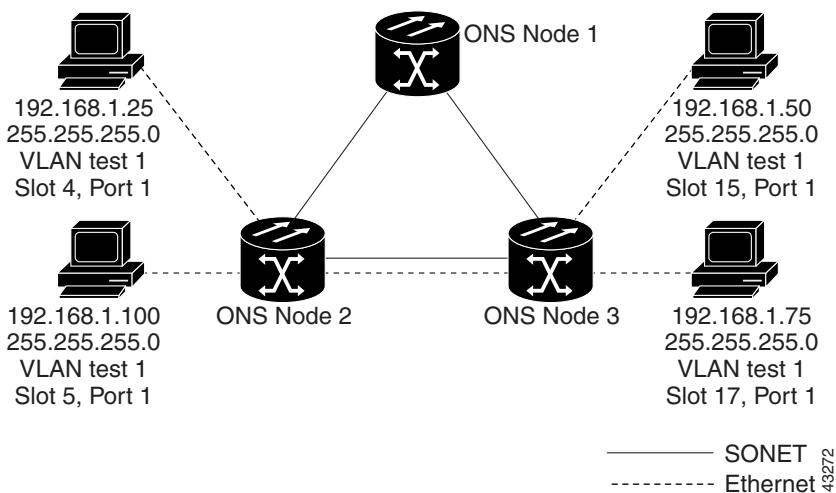
Note

To make an STS-12c/VC4-4c Ethernet circuit, Ethernet cards must be configured in single-card EtherSwitch or port-mapped mode. Multicard mode does not support STS-12c/VC4-4c Ethernet circuits.

E-Series Point-to-Point Ethernet Circuits

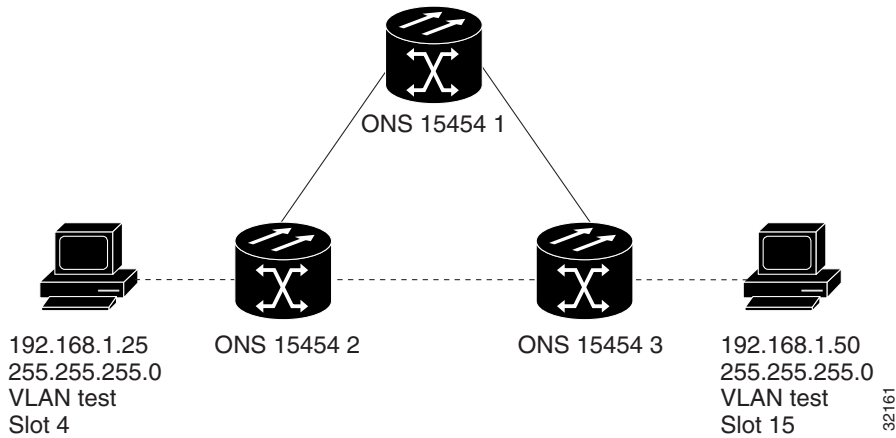
The ONS nodes can set up a point-to-point (straight) Ethernet circuit as single-card, port-mapped, or multcard circuit (Figure 22-19).

Figure 22-19 Multicard EtherSwitch Point-to-Point Circuit



Single-card EtherSwitch and port-mapped modes provide a full STS-12c of bandwidth between two Ethernet circuit endpoints (Figure 22-20).

Figure 22-20 Single-Card EtherSwitch or Port-Mapped Point-to-Point Circuit



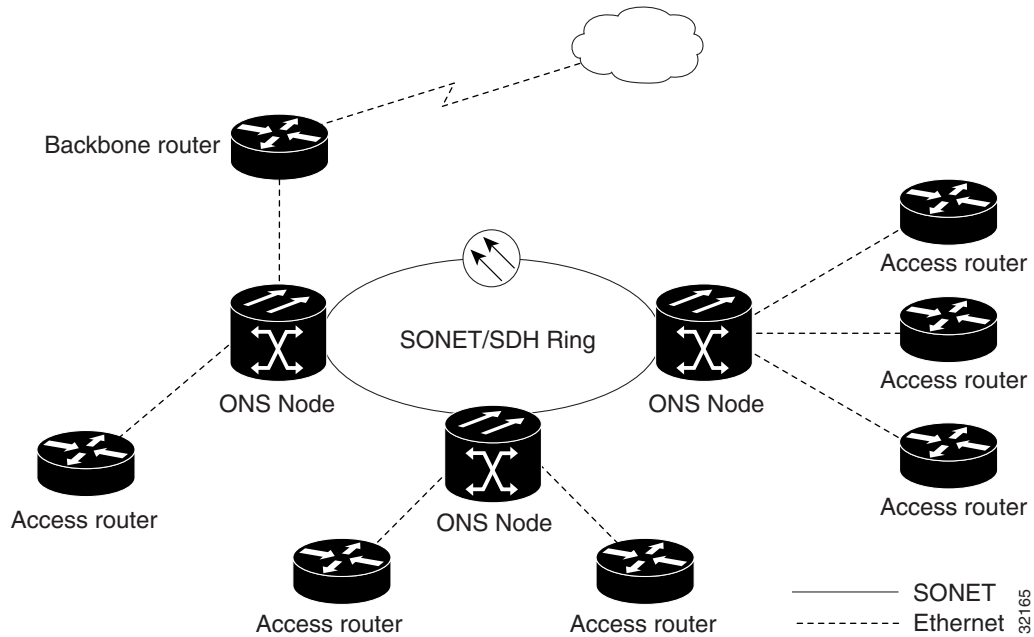
Note

A port-mapped, point-to-point circuit cannot join an E-Series port-based VLAN, but can transport external VLANs.

E-Series Shared Packet Ring Ethernet Circuits

A shared packet ring allows additional nodes (besides the source and destination nodes) access to an Ethernet STS circuit. The E-Series card ports on the additional nodes can share the circuit's VLAN and bandwidth. Figure 22-21 illustrates a shared packet ring. Your network architecture might differ from the example.

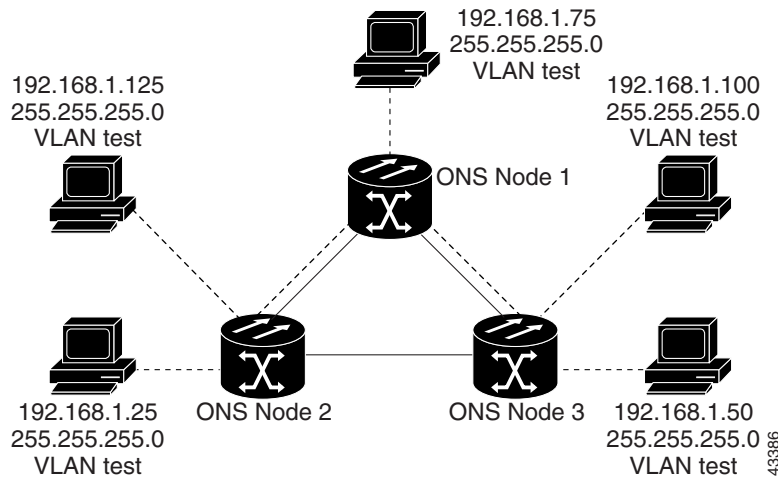
Figure 22-21 Shared Packet Ring Ethernet Circuit



E-Series Hub-and-Spoke Ethernet Circuit Provisioning

The hub-and-spoke configuration connects point-to-point circuits (the spokes) to an aggregation point (the hub). In many cases, the hub links to a high-speed connection and the spokes are Ethernet cards. [Figure 22-22](#) illustrates a hub-and-spoke ring. Your network architecture might differ from the example.

Figure 22-22 Hub-and-Spoke Ethernet Circuit



E-Series Ethernet Manual Cross-Connects

ONS nodes require end-to-end CTC visibility between nodes for normal provisioning of Ethernet circuits. When other vendors' equipment sits between ONS nodes, OSI/TARP-based equipment does not allow tunneling of the ONS node TCP/IP-based DCC. To circumvent this inconsistent DCC, the Ethernet circuit must be manually cross connected to an STS channel using the non-ONS network. The manual cross-connect allows an Ethernet circuit to run from ONS node to ONS node utilizing the non-ONS network.

**Note**

In this section, “cross-connect” and “circuit” have the following meanings: cross-connect refers to the connections that occur within a single ONS node to allow a circuit to enter and exit an ONS 15454. Circuit refers to the series of connections from a traffic source (where traffic enters the ONS 15454 network) to the drop or destination (where traffic exits an ONS 15454 network).

Remote Monitoring Specification Alarm Thresholds

The ONS nodes features remote monitoring (RMON) that allows network operators to monitor the health of the network with a network management system (NMS).

One of the ONS node's RMON MIBs is the Alarm group, which consists of the alarmTable. An NMS uses the alarmTable to find the alarm-causing thresholds for network performance. The thresholds apply to the current 15-minute interval and the current 24-hour interval. RMON monitors several variables, such as Ethernet collisions, and triggers an event when the variable crosses a threshold during that time interval. For example, if a threshold is set at 1000 collisions and 1001 collisions occur during the 15-minute interval, an event triggers. CTC allows you to provision these thresholds for Ethernet statistics.

For Ethernet RMON alarm threshold procedures, refer to the *Cisco ONS 15454 Troubleshooting Guide*, *Cisco ONS 15454 Troubleshooting Guide* or *Cisco ONS 15327 Troubleshooting Guide*.



CE-100T-8 Ethernet Operation

This chapter covers the operation of the CE-100T-8 (Carrier Ethernet) card supported on the ONS 15454 (15454-CE-100T-8). A CE-100T-8 card is also supported on the ONS 15310 (15310-CE-100T-8). Provisioning is done through Cisco Transport Controller (CTC) or Transaction Language One (TL1). Cisco IOS is not supported on the CE-100T-8 card.

For Ethernet card specifications, refer to the *Cisco ONS 15454 Reference Manual*. For step-by-step Ethernet card circuit configuration procedures, refer to the *Cisco ONS 15454 Procedure Guide*. Refer to the *Cisco ONS SONET TL1 Command Guide* for TL1 provisioning commands.

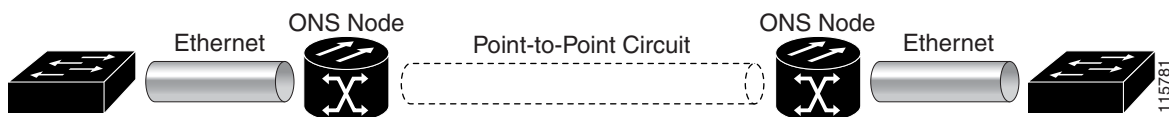
Chapter topics include:

- [CE-100T-8 Overview, page 23-1](#)
- [CE-100T-8 Ethernet Features, page 23-2](#)
- [CE-100T-8 SONET Circuits and Features, page 23-6](#)

CE-100T-8 Overview

The CE-100T-8 is a Layer 1 mapper card with eight 10/100 Ethernet ports. It maps each port to a unique SONET circuit in a point-to-point configuration. [Figure 23-1](#) illustrates a sample CE-100T-8 application. In this example, data traffic from the Fast Ethernet port of a switch travels across the point-to-point circuit to the Fast Ethernet port of another switch.

Figure 23-1 CE-100T-8 Point-to-Point Circuit



The CE-100T-8 cards allow you to provision and manage an Ethernet private line service like a traditional SONET line. CE-100T-8 card applications include providing carrier-grade Ethernet private line services and high-availability transport.

The CE-100T-8 card carries any Layer 3 protocol that can be encapsulated and transported over Ethernet, such as IP or IPX. The Ethernet frame from the data network is transmitted on the Ethernet cable into the standard RJ-45 port on a CE-100T-8 card. The CE-100T-8 card transparently maps Ethernet frames into the SONET payload using packet-over-SONET (POS) encapsulation. The POS circuit with its encapsulated Ethernet inside is then multiplexed onto an OC-N card like any other SONET STS. When

the payload reaches the destination node, the process is reversed and the data is transmitted from the standard RJ-45 port in the destination CE-100T-8 card onto the Ethernet cable and data network. The POS process is covered in detail in [Chapter 21, “POS on ONS Ethernet Cards.”](#)

The CE-100T-8 card supports ITU-T G.707 and Telcordia GR-253 based standards for SONET. It offers a carrier-class level of features and reliability. This includes errorless (0-msec impact on traffic) reprovisioning. When circuit or port provisioning takes place, this operation does not affect the performance of other ports and circuit configurations that are already established on the card.

Software upgrades are errorless. However when the CE-100T-8 firmware is upgraded, the upgrade has an effect on traffic similar to the effect of a hard reset on the CE-100T-8. A software upgrade or a firmware upgrade does not affect the existing provisioning of the ports and circuits on the CE-100T-8 card.

Span upgrades are hitless. Protection and maintenance switches are also hitless.

The CE-100T-8 offers full TL1-based provisioning capability. Refer to the *Cisco ONS SONET TL1 Command Guide* for CE-100T-8 TL1 provisioning commands.

CE-100T-8 Ethernet Features

The CE-100T-8 card has eight front-end Ethernet ports which use standard RJ-45 connectors for 10BASE-T Ethernet/100BASE-TX Ethernet media. Ethernet Ports 1 through 8 each map to a POS port with a corresponding number. The console port on the CE-100T-8 card is not functional.

The CE-100T-8 cards forward valid Ethernet frames unmodified over the SONET network. Information in the headers is not affected by the encapsulation and transport. For example, included IEEE 802.1Q information will travel through the process unaffected.

The ONS 15454 CE-100T-8 and the ONS 15310 CE-100T-8 support maximum Ethernet frame sizes of 1600 bytes including the CRC. The MTU size is not configurable and is set at a 1500 byte maximum (standard Ethernet MTU). Baby giant frames in which the standard Ethernet frame is augmented by 802.1 Q tags or MPLS tags are also supported. Full Jumbo frames are not supported.

The CE-100T-8 cards discard certain types of erroneous Ethernet frames rather than transport them over SONET. Erroneous Ethernet frames include corrupted frames with cyclic redundancy check (CRC) errors and undersized frames that do not conform to the minimum 64-byte length Ethernet standard.



Note

Many Ethernet attributes are also available through the network element default feature. For more information on NE defaults, refer to the "Network Element Defaults" appendix in the Cisco ONS 15454 Reference Manual.

Autonegotiation, Flow Control, and Frame Buffering

On the CE-100T-8, Ethernet link autonegotiation is on by default. The user can also set the link speed, duplex, and flow control manually under the card-level Provisioning tab of CTC.

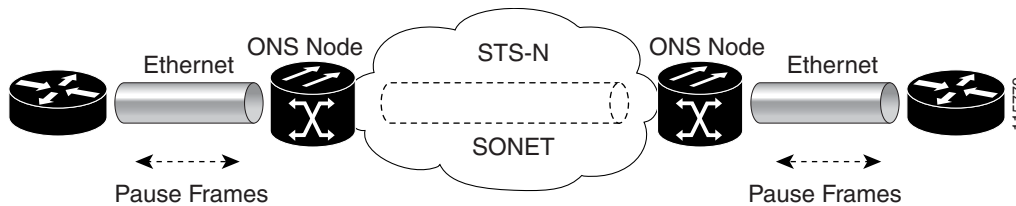
The CE-100T-8 supports IEEE 802.3x flow control and frame buffering to reduce data traffic congestion. Flow control is on by default.

To prevent over-subscription, buffer memory is available for each port. When the buffer memory on the Ethernet port nears capacity, the CE-100T-8 uses IEEE 802.3x flow control to transmit a pause frame to the attached Ethernet device. Flow control and autonegotiation frames are local to the Fast Ethernet interfaces and the attached Ethernet devices. These frames do not continue through the POS ports.

The CE-100T-8 card has symmetric flow control and proposes symmetric flow control when autonegotiating flow control with attached Ethernet devices. Symmetric flow control allows the CE-100T-8 cards to respond to pause frames sent from external devices and to send pause frames to external devices.

The pause frame instructs the source to stop sending packets for a specific period of time. The sending station waits the requested amount of time before sending more data. [Figure 23-2](#) illustrates pause frames being sent and received by CE-100T-8 cards and attached switches.

Figure 23-2 Flow Control



This flow-control mechanism matches the sending and receiving device throughput to that of the bandwidth of the STS circuit. For example, a router might transmit to the Ethernet port on the CE-100T-8 card. This particular data rate might occasionally exceed 51.84 Mbps, but the SONET circuit assigned to the CE-100T-8 port might be only STS-1 (51.84 Mbps). In this example, the CE-100T-8 sends out a pause frame and requests that the router delay its transmission for a certain period of time. With flow control and a substantial per-port buffering capability, a private line service provisioned at less than full line rate capacity (STS-1) is efficient because frame loss can be controlled to a large extent.

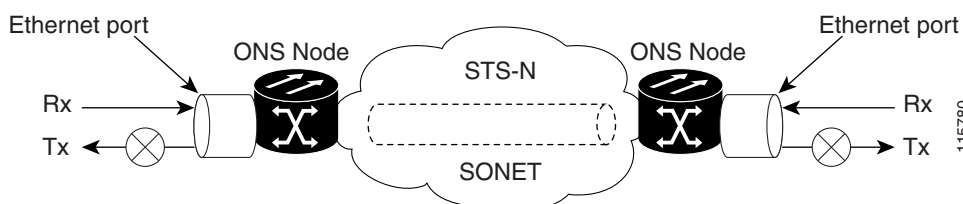
Ethernet Link Integrity Support

The CE-100T-8 supports end-to-end Ethernet link integrity ([Figure 23-3](#)). This capability is integral to providing an Ethernet private line service and correct operation of Layer 2 and Layer 3 protocols on the attached Ethernet devices.

End-to-end Ethernet link integrity means that if any part of the end-to-end path fails, the entire path fails. It disables the Ethernet port on the CE-100T-8 card if the remote Ethernet port is unable to transmit over the SONET network or if the remote Ethernet port is disabled.

Failure of the entire path is ensured by turning off the transmit pair at each end of the path. The attached Ethernet devices recognize the disabled transmit pair as a loss of carrier and consequently an inactive link or link fail.

Figure 23-3 End-to-End Ethernet Link Integrity Support



**Note**

Some network devices can be configured to ignore a loss of carrier condition. If a device configured to ignore a loss of carrier condition attaches to a CE-100T-8 card at one end, alternative techniques (such as use of Layer 2 or Layer 3 keep-alive messages) are required to route traffic around failures. The response time of such alternate techniques is typically much longer than techniques that use link state as indications of an error condition.

Enhanced State Model for Ethernet and SONET/SDH Ports

The CE-100T-8 supports the Enhanced State Model (ESM) for the Ethernet ports, as well as for the SONET/SDH circuit. For more information about the ESM, see the “Enhanced State Model” appendix in the *ONS 15454 SONET Reference Manual* or the *ONS 15454 SDH Reference Manual*.

The Ethernet ports can be set to the ESM service states including the In-Service, Automatic In-Service (IS,AINS) administrative state. IS,AINS initially puts the port in the Out-of-Service and Autonomous, Automatic In-Service (OOS-AU,AINS) state. In this service state, alarm reporting is suppressed, but traffic is carried and loopbacks are allowed. After the soak period passes, the port changes to In-Service and Normal (IS-NR). Raised fault conditions, whether their alarms are reported or not, can be retrieved on the CTC Conditions tab or by using the TL1 RTRV-COND command.

Two Ethernet port alarms/conditions, CARLOSS and TPTFAIL, can prevent the port from going into service. This occurs even though alarms are suppressed when a CE-100T-8 circuit is provisioned with the Ethernet ports set to the IS,AINS state, because the CE-100T-8 link integrity function is active and ensures that the links at both ends are not enabled until all SONET and Ethernet errors along the path are cleared. As long as the link integrity function keeps the end-to-end path down, both ports will have at least one of the two conditions needed to suppress the AINS-to-IS transition. Therefore, the ports will remain in the AINS state with alarms suppressed.

ESM also applies to the SONET/SDH circuits of the CE-100T-8 card. If the SONET/SDH circuit is set up in IS,AINS state and the Ethernet error occurs before the circuit transitions to IS, then link integrity will also prevent the circuit transition to the IS state until the Ethernet port errors are cleared at both ends. The service state will be OOS-AU,AINS as long as the administrative state is IS,AINS. When there are no Ethernet or SONET errors, link integrity enables the Ethernet port at each end. Simultaneously, the AINS countdown begins as normal. If no additional conditions occur during the time period, each port transitions to the IS-NR state. During the AINS countdown, the soak time remaining is available in CTC and TL1. The AINS soaking logic restarts from the beginning if a condition appears again during the soak period.

A SONET/SDH circuit provisioned in the IS,AINS state remains in the initial Out-of-Service (OOS) state until the Ethernet ports on each end of the circuit transition to the IS-NR state. The SONET/SDH circuit transports Ethernet traffic and counts statistics when link integrity turns on the Ethernet port, regardless of whether this AINS-to-IS transition is complete.

IEEE 802.1Q CoS and IP ToS Queuing

The CE-100T-8 references IEEE 802.1Q class of service (CoS) thresholds and IP type of service (ToS) (IP Differentiated Services Code Point [DSCP]) thresholds for priority queueing. CoS and ToS thresholds for the CE-100T-8 are provisioned on a per port level. This allows the user to provide priority treatment based on open standard quality of service (QoS) schemes already existing in the data network attached to the CE-100T-8. The QoS treatment is applied to both Ethernet and POS ports.

Any packet or frame with a priority greater than the set threshold is treated as priority traffic. This priority traffic is sent to the priority queue instead of the normal queue. When buffering occurs, packets on the priority queue preempt packets on the normal queue. This results in lower latency for the priority traffic, which is often latency-sensitive traffic, such as VoIP.

Because these priorities are placed on separate queues, the priority queuing feature should not be used to separate rate-based CIR/EIR marked traffic (sometimes done at a Metro Ethernet service provider edge). This could result in out-of-order packet delivery for packets of the same application, which would cause performance issues with some applications.

For an IP ToS-tagged packet, the CE-100T-8 can map any of the 256 priorities specified in IP ToS to priority or best effort. The user can configure a different ToS on CTC at the card-level view under the Provisioning > Ether Ports tabs. Any ToS class higher than the class specified in CTC is mapped to the priority queue, which is the queue geared towards low latency. By default, the ToS is set to 255, which is the highest ToS value. This results in all traffic being treated with equal priority by default.

Table 23-3 shows which values are mapped to the priority queue for sample IP ToS settings. (ToS settings span the full 0 to 255 range, but only selected settings are shown.)

Table 23-1 IP ToS Priority Queue Mappings

ToS Setting in CTC	ToS Values Sent to Priority Queue
255 (default)	None
250	251–255
150	151–255
100	101–255
50	51–255
0	1–255

For a CoS-tagged frame, the CE-100T-8 can map the eight priorities specified in CoS to priority or best effort. The user can configure a different CoS on CTC at the card-level view under the Provisioning > Ether Ports tabs. Any CoS class higher than the class specified in CTC is mapped to the priority queue, which is the queue geared towards low latency. By default, the CoS is set to 7, which is the highest CoS value. This results in all traffic being treated with equal priority by default.

Table 23-3 shows which values are mapped to the priority queue for CoS settings.

Table 23-2 CoS Priority Queue Mappings

CoS Setting in CTC	CoS Values Sent to Priority Queue
7 (default)	none
6	7
5	6, 7
4	5, 6, 7
3	4, 5, 6, 7
2	3, 4, 5, 6, 7
1	2, 3, 4, 5, 6, 7
0	1, 2, 3, 4, 5, 6, 7

Ethernet frames without VLAN tagging use ToS-based priority queuing if both ToS and CoS priority queuing is active on the card. The CE-100T-8 card's ToS setting must be lower than 255 (default) and the CoS setting lower than 7 (default) for CoS and ToS priority queuing to be active. A ToS setting of 255 (default) disables ToS priority queuing, so in this case the CoS setting would be used.

Ethernet frames with VLAN tagging use CoS-based priority queuing if both ToS and CoS are active on the card. The ToS setting is ignored. CoS based priority queuing is disabled if the CoS setting is the 7 (default), so in this case the ToS setting would be used.

If the CE-100T-8 card's ToS setting is 255 (default) and the CoS setting is 7 (default), priority queuing is not active on the card, and data gets sent to the default normal traffic queue. Also if data is not tagged with a ToS value or a CoS value before it enters the CE-100T-8 card, it gets sent to the default normal traffic queue.

**Note**

Priority queuing has no effect when flow control is enabled (default) on the CE-100T-8. Under flow control a 6 kilobyte single-priority first in first out (FIFO) buffer fills, then a PAUSE frame is sent. This results in the packet ordering priority becoming the responsibility of the external device, which is buffering as a result of receiving the PAUSE flow-control frames.

**Note**

Priority queuing has no effect when the CE-100T-8 is provisioned with STS-3C circuits. The STS-3c circuit has more data capacity than Fast Ethernet, so CE-100T-8 buffering is not needed. Priority queuing only takes effect during buffering.

RMON and SNMP Support

The CE-100T-8 card features remote monitoring (RMON) that allows network operators to monitor the health of the network with a network management system (NMS). The CE-100T-8 uses the ONG RMON. The ONG RMON contains the statistics, history, alarms, and events MIB groups from the standard RMON MIB. A user can access RMON threshold provisioning through TL1 or CTC. For RMON threshold provisioning with CTC, see the Cisco ONS 15454 Procedure Guide (NTP-A279) and the Cisco ONS 15454 Troubleshooting Guide. For TL1 information, see the Cisco ONS SONET TL1 Command Guide.

Statistics and Counters

The CE-100T-8 has a full range of Ethernet and POS statistics under **Performance > Ether Ports** or **Performance > POS Ports**.

CE-100T-8 SONET Circuits and Features

The CE-100T-8 has eight POS ports, numbered one through eight, which are exposed to management with CTC or TL1. Each POS port is statically mapped to a matching Ethernet port. By clicking the card-level Provisioning tab > POS Ports tab, the user can configure the Administrative State, Framing Type, and Encapsulation Type. By clicking the card-level Performance tab > POS Ports tab, the user can view the statistics, utilization, and history for the POS ports.

Available Circuit Sizes and Combinations

Each POS port terminates an independent contiguous SONET concatenation (CCAT) or virtual SONET concatenation (VCAT). The SONET circuit is created for these ports through CTC or TL1 in the same manner as a SONET circuit for a non-Ethernet line card. [Table 23-3](#) shows the circuit sizes available for the CE-100T-8 on the ONS 15454.

Table 23-3 CE-100T-8 Supported Circuit Sizes

CCAT High Order	VCAT High Order	VCAT Low Order
STS-1	STS-1-1v	VT1.5- <i>nV</i> (<i>n</i> = 1 to 64)
STS-3c	STS-1-2v	
	STS-1-3v	

A single circuit provides a maximum of 100 Mbps of throughput, even when an STS-3c circuit, which has a bandwidth equivalent of 155 Mbps, is provisioned. This is due to the hardware restriction of the Fast Ethernet port. A VCAT circuit is also restricted in this manner. [Table 23-3](#) shows the minimum SONET circuit sizes required for 10 Mbps and 100 Mbps wire speed service.

Table 23-4 SONET Circuit Sizes and Ethernet Services

Ethernet Wire Speed	CCAT High Order	VCAT High Order	VCAT Low Order
Line Rate 100BaseT	STS-3c	STS-1-3v, STS-1-2v*	Not applicable
Sub Rate 100BaseT	STS-1	STS-1-1v	VT1.5- <i>xv</i> (<i>x</i> =1-64)
Line Rate 10BaseT	STS-1	Not applicable	VT1.5-7v
Sub Rate 10BaseT	Not applicable	Not applicable	VT1.5- <i>xv</i> (<i>x</i> =1-6)

*STS-1-2v provides a total transport capacity of 98 Mbps.

The number of available circuits and total combined bandwidth for the CE-100T-8 depends on the combination of circuit sizes configured. [Table 23-5](#) shows the circuit size combinations available for the CE-100T-8 on the ONS 15454.

Table 23-5 CCAT High Order Circuit Size Combinations

Number of STS-3c Circuits	Maximum Number of STS-1 Circuits
None	8
1	7
2	6
3	3
4	None

Table 23-6 VCAT High Order Circuit Combinations for STS-1-3v and STS-1-2v

Number of STS-1-3v Circuits	Maximum Number of STS-1-2v Circuits
None	4
1	3
2	2
3	1
4	None

The CE-100T-8 supports up to eight low order VCAT circuits. The available circuit sizes are VT1.5-Xv, where X is the range from 1 to 64. A maximum of four circuits are available at the largest low order VCAT circuit size, VT1.5-64v.

A user can combine CCAT high order, VCAT high order and VCAT low order circuits. The following table details the maximum density service combinations.

Table 23-7 CE-100T-8 Maximum Service Densities

Service Combination	STS-3c or STS-1-3v	STS-1-2v	STS-1	VT1.5-xV (x=1-7)	Number of Active Service
1	4	0	0	0	4
2	3	1	1	0	5
3	3	0	3	0	6
4	3	0	0	4 (x=1-21)*	7*
5	2	2	2	0	6
6	2	1	4	0	7
7	2	1	1	4 (x=1-21)*	8*
8	2	0	6	0	8
9	2	0	3	3 (x=1-28)	8
10	2	0	0	6 (x=1-28)	8
11	1	3	3	0	7
12	1	2	5	0	8
13	1	2	2	3 (x=1-28)	8
14	1	1	1	5 (x=1-28)	8
15	1	0	7	0	8
16	1	0	3	4 (x=1-42)	8
17	1	0	0	7 (x=1-28)	8
18	0	4	4	0	8
19	0	3	3	2 (x=1-42)	8
20	0	0	8	0	8
21	0	0	4	4 (x=1-42)	8
22	0	0	0	8 (x=1-42)	8

*This low order VCAT circuit combination is achievable if one of the first two circuits created on the card is an low order VCAT circuit. If the first two circuits created on the card are high order VCAT or CCAT, then a maximum of three low order VCAT circuits can be created on the card.

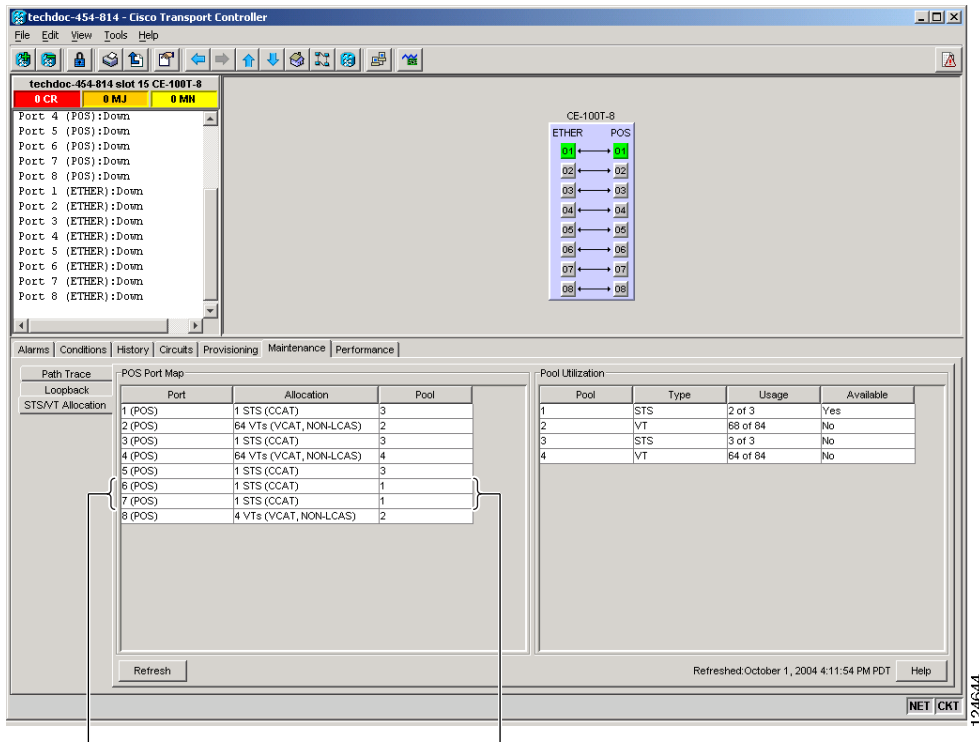
CE-100T-8 Pools and STS/VT Allocation Tab

The CE-100T-8 has four pools, each with a maximum capacity of three STSs. All VCAT circuit members must be from the same pool.

One of the four memory pools is reserved for the low order VCAT circuits, if sufficient bandwidth exists to support the high order circuits on the remaining three pools. The high order CCAT circuits use all the available capacity from a single memory pool before beginning to use the capacity of a new pool. The memory pools will be allocated alternatively for the first three high order VCAT circuits if the pools have the sufficient bandwidth to support the requested circuit size. To help prevent stranding bandwidth, provision your high order VCAT circuits first to distribute them evenly.

At the CTC card-level view under the Maintenance tab, the STS/VT Allocation tab displays how the provisioned circuits populate the four pools. This information can be useful in freeing up the bandwidth required for provisioning a circuit, if there is not enough existing capacity on any one pool for provisioning the desired circuit. The user can look at the distribution of the existing circuits among the four pools and decide which circuits to delete in order to free up space for the desired circuit.

Figure 23-4 CE-100T-8 STS/VT Allocation Tab



Both Port 6 and Port 7 belong to Pool 1

For example if a user needs to provision an STS-3c or STS-1-3v on the CE-100T-8 card shown in Figure 23-4, an STS-3c or STS-1-3v worth of bandwidth is not available from any of the four pools. The user needs to delete circuits from the same pool to free up bandwidth. If the bandwidth is available but scattered among the pools, the circuit cannot be provisioned. Looking at the POS Port Map table, the user can determine which circuits belong to which pools. The Pool and Port columns in Figure 23-4 show that port 6 and port 7 are both drawn from Pool 1, and no other circuits are drawn from Pool 1. Deleting these two STS-1 circuits will free up an STS-3c or STS-1-3v worth of bandwidth from a single pool.

If the user did not determine what circuits to delete from the table information, he might delete the STS-1 circuits on port 3, port 5 and port 6. This frees up an STS-3c or STS-1-3v worth of bandwidth but the required bandwidth is not available from a single pool, and the STS-3c or STS-1-3v circuit is not provisionable.

The POS Port table has a row for each port with three columns (Figure 23-4). They show the port number, the circuit size and type, and the pool it is drawn from. The Pool Utilization table has four columns and shows the pool number, the type of circuits on that pool, how much of the pool's capacity is being used, and whether additional capacity is available.

CE-100T-8 VCAT Characteristics

The ML-100T-8 card and the CE-100T-8 card (both the ONS 15310-CL version and the ONS 15454 SONET/SDH version) have hardware-based support for the ITU-T G.7042 standard link capacity adjustment scheme (LCAS). This allows the user to dynamically resize a high order or low order VCAT circuit through CTC or TL1 without affecting other members of the VCG (errorless). ML-100T-8 LCAS support is high order only and is limited to a two member VCG.

The ONS 15454 SONET/SDH ML-Series card has a software-based LCAS (SW-LCAS) scheme. This scheme is also supported by both the ML-100T-8 card and both versions of the CE-100T-8, but only for circuits terminating on a ONS 15454 SONET/SDH ML-Series card.

The CE-100T-8 card allows independent routing and protection preferences for each member of a VCAT circuit. The user can also control the amount of VCAT circuit capacity that is fully protected, unprotected or if the circuit is on a bidirectional line switched ring (BLSR), uses protection channel access (PCA). Alarms are supported on a per-member as well as per virtual concatenation group (VCG) basis.



Note

The maximum tolerable VCAT differential delay for the CE-100T-8 is 48 milliseconds. The VCAT differential delay is the relative arrival time measurement between members of a virtual concatenation group (VCG).

CE-100T-8 POS Encapsulation, Framing, and CRC

The CE-100T-8 uses Cisco EoS LEX (LEX). LEX is the primary encapsulation of ONS Ethernet cards. In this encapsulation the protocol field is set to the values specified in Internet Engineering Task Force (IETF) Request For Comments (RFC) 1841. The user can provision GPF-F framing (default) or high-level data link control (HDLC) framing. With GFP-F framing, the user can also configure a 32-bit CRC (the default) or no CRC (none). When LEX is used over GFP-F it is standard Mapped Ethernet over GFP-F according to ITU-T G.7041. HDLC framing provides a set 32-bit CRC. From CTC go to CE card view and click the Provisioning >pos ports tab, to see the various parameters that can be configured on the POS ports, see “[ML-Series POS and Ethernet Statistics Fields and Buttons](#)”. Various parameters like, admin state, service state, framing type, CRC, MTU and soak time for a port can be configured here. For more details on the interoperability of ONS Ethernet cards, including information on encapsulation, framing, and CRC, see the “[POS on ONS Ethernet Cards](#)” chapter.

The CE-100T-8 card supports GFP-F null mode. GFP-F CMFs are counted and discarded.

CE-100T-8 Loopback, J1 Path Trace, and SONET Alarms

The CE-100T-8 card supports terminal and facility loopbacks. It also reports SONET alarms and transmits and monitors the J1 Path Trace byte in the same manner as OC-N cards. Support for path termination functions includes:

- H1 and H2 concatenation indication
- C2 signal label
- Bit interleaved parity 3 (BIP-3) generation
- G1 path status indication
- C2 path signal label read/write
- Path level alarms and conditions, including loss of pointer, unequipped, payload mismatch, alarm indication signal (AIS) detection, and remote defect indication (RDI)
- J1 path trace for high order paths
- J2 path trace for low order paths
- J2 path trace for low order VCAT circuits at the member level
- Extended signal label for the low order paths



Command Reference



Note

The terms “Unidirectional Path Switched Ring” and “UPSR” may appear in Cisco literature. These terms do not refer to using Cisco ONS 15xxx products in a unidirectional path switched ring configuration. Rather, these terms, as well as “Path Protected Mesh Network” and “PPMN,” refer generally to Cisco’s path protection feature, which may be used in any topological network configuration. Cisco does not recommend using its path protection feature in any particular topological network configuration.

This appendix provides a command reference for those Cisco IOS commands or those aspects of Cisco IOS commands unique to ML-Series cards. For information about the standard Cisco IOS Release 12.2 commands, refer to the Cisco IOS documentation set available at <http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/>.

[no] bridge *bridge-group-number* protocol {drpri-rstp | ieee | rstp}

To define the protocol employed by a bridge-group, use the **bridge protocol** global configuration command. If no protocol will be employed by the bridge-group, this command is not needed. To remove a protocol from the bridge group, use the no form of this command with the appropriate keywords and arguments.

Syntax Description	Parameter	Description
	drpri-rstp	The protocol that enables the Dual RPR Interconnect (DRPRI) feature of the ML-Series cards.
	ieee	IEEE 802.1D Spanning Tree Protocol.
	rstp	IEEE 802.1W Rapid Spanning Tree Protocol.
	<i>bridge-group-number</i>	The identifying number of the bridge group being assigned a protocol.

Defaults N/A

Command Modes Global configuration

Usage Guidelines The protocol DRPRI-RSTP is only employed when configuring ML-Series cards as part of a DRPRI. A bridge group with DRPRI is limited to one protocol, so the bridge group cannot also implement rapid spanning tree protocol (RSTP) or spanning tree protocol (STP).

Examples The following example assigns the DRPRI protocol to the bridge group with the bridge group number of 100.

```
Router(config)# bridge 100 protocol drpri-rstp
```

[no] clock auto

Use the **clock auto** command to determine whether the system clock parameters are configured automatically from the TCC2/TCC2P card. When enabled, both summertime and timezone are automatically configured, and the system clock is periodically synchronized to the TCC2/TCC2P card. Use the no form of the command to disable this feature.

Syntax Description This command has no arguments or keywords.

Defaults The default setting is clock auto.

Command Modes Global configuration

Usage Guidelines The no form of the command is required before any manual configuration of summertime, timezone, or clock. The no form of the command is required if Network Time Protocol (NTP) is configured in Cisco IOS. The ONS 15454 SONET/SDH is also configured through Cisco Transport Controller (CTC) to use a NTP or Simple Network Time Protocol (SNTP) server to set the date and time of the node.

Examples Router(config)# **no clock auto**

Related Commands

- clock summertime
- clock timezone
- clock set

interface spr 1

Use this command to create a shared packet ring (spr) interface on an ML-Series card for a resilient packet ring (RPR). If the interface has already been created, this command enters spr interface configuration mode. The only valid spr interface number is 1.

Defaults

N/A

Command Modes

Global configuration

Usage Guidelines

The command allows the user to create a virtual interface for the RPR/SPR. Commands such as **spr wrap** or **spr station-id** can then be applied to the RPR through SPR configuration command mode.

Examples

The following example creates the shared packet ring interface:

```
Router(config)# interface spr 1
```

Related Commands

spr drpri-id
spr-intf-id
spr station-id
spr wrap

[no] pos pdi holdoff *time*

Use this command to specify the time, in milliseconds, to holdoff sending the path defect indication (PDI) to the far-end when a VCAT member circuit is added to the virtual concatenation group (VCG). Use the no form of the command to use the default value.

Syntax Description

Parameter	Description
<i>time</i>	delay time in milliseconds, 100 to 1000

Defaults

The default value is 100 milliseconds.

Command Modes

Interface configuration mode (POS only)

Usage Guidelines

This value is normally configured to match the setting on the peer PTE. The time granularity for this command is 1 milliseconds.

Examples

```
Gateway(config)# int pos0
Gateway(config-if)# pos pdi holdoff 500
```

Related Commands

pos trigger defects

[no] pos report *alarm*

Use this command to specify which alarms/signals are logged to the console. This command has no effect on whether alarms are reported to the TCC2/TCC2P and CTC. These conditions are soaked and cleared per Telcordia GR-253. Use the no form of the command to disable reporting of a specific alarm/signal.

Syntax Description	Parameter	Description
	<i>alarm</i>	The SONET/SDH alarm that is logged to the console. The alarms are as follows: all —All link down alarm failures ber_sd_b3 —PBIP BER in excess of SD threshold failure ber_sf_b3 —PBIP BER in excess of SD threshold failure encap —Path Signal Label Encapsulation Mismatch failure pais —Path Alarm Indication Signal failure plop —Path Loss of Pointer failure ppdi —Path Payload Defect Indication failure pplm —Payload label mismatch path prdi —Path Remote Defect Indication failure ptim —Path Trace Indicator Mismatch failure puneq —Path Label Equivalent to Zero failure

Defaults The default is to report all alarms.

Command Modes Interface configuration mode (POS only)

Usage Guidelines This value is normally configured to match the setting on the peer PTE.

Examples

```
Gateway(config)# int pos0
Gateway(config-if)# pos report all
Gateway(config-if)# pos flag c2 1
03:16:51: %SONET-4-ALARM: POS0: PPLM
Gateway(config-if)# pos flag c2 0x16
03:17:34: %SONET-4-ALARM: POS0: PPLM cleared
```

Related Commands pos trigger defects

[non] pos trigger defects *condition*

Use this command to specify which conditions cause the associated POS link state to change. These conditions are soaked/cleared using the delay specified in the **pos trigger delay** command. Use the no form of the command to disable triggering on a specific condition.

Syntax Description	Parameter	Description
	<i>condition</i>	<p>The SONET/SDH condition that causes the link state change. The conditions are as follows:</p> <ul style="list-style-type: none"> all—All link down alarm failures ber_sd_b3—PBIP BER in excess of SD threshold failure ber_sf_b3—PBIP BER in excess of SD threshold failure (default) encap—Path Signal Label Encapsulation Mismatch failure (default) pais—Path Alarm Indication Signal failure (default) plop—Path Loss of Pointer failure (default) ppdi—Path Payload Defect Indication failure (default) pplm—Payload label mismatch path (default) prdi—Path Remote Defect Indication failure (default) ptim—Path Trace Indicator Mismatch failure (default) puneq—Path Label Equivalent to Zero failure (default)
Defaults		See list in above description.
Command Modes		Interface configuration mode (POS only)
Usage Guidelines		This value is normally configured to match the setting on the peer PTE.
Examples		<pre>Gateway(config)# int pos0 Gateway(config-if)# pos trigger defects all</pre>
Related Commands		pos trigger delay

[no] pos trigger delay *time*

Use this command to specify which conditions cause the associated POS link state to change. The conditions specified in the **pos trigger defects** command are soaked/cleared using this delay. Use the no form of the command to use the default value.

Syntax Description	Parameter	Description
	<i>time</i>	delay time in milliseconds, 200 to 2000

Defaults The default value is 200 milliseconds.

Command Modes Interface configuration mode (POS only)

Usage Guidelines This value is normally configured to match the setting on the peer PTE. The time granularity for this command is 50 milliseconds.

Examples

```
Gateway(config)# int pos0
Gateway(config-if)# pos trigger delay 500
```

Related Commands pos trigger defects

[no] pos scramble-spe

Use this command to enable scrambling.

Syntax Description This command has no arguments or keywords.

Defaults The default value depends on the encapsulation.

encap	scrambling
LEX	pos scramble-spe
PPP/HDLC	no pos scramble-spe

Command Modes Interface configuration mode (POS only)

Usage Guidelines This value is normally configured to match the setting on the peer PTE. This command might change the pos flag c2 configuration.

Examples

```
Gateway(config)# int pos0
Gateway(config-if)# pos scramble-spe
```

Related Commands pos flag c2

[no] pos vcat defect {immediate | delayed}

Sets the VCAT defect processing mode to either handle a defects state change the instant it is detected or wait for the time specified by **pos trigger delay**. Use the no form of the command to use the default value.

Syntax Description

Parameter	Description
immediate	Handles a defect state change the instant it is detected.
delayed	Handles the defect after the time specified by the command pos trigger delay . If delay is configured and the circuit is on RPR, then the RPR defect processing will also be delayed by the delay time.

Defaults

The default setting is immediate.

Command Modes

POS interface configuration

Usage Guidelines

Immediate should be used if the VCAT circuit uses unprotected SONET circuits. Delayed should be run if the VCAT circuit uses SONET protected circuits (bidirectional line switch ring [BLSR] or path protection).

Examples

The following example sets an ML-Series card to delayed:

```
Router(config)# interface pos 1
Router(config-if)# pos vcat defect delayed
```

Related Commands

```
interface spr 1
spr wrap
interface pos 1
pos trigger delay
```

[no] pos vcat resequence {enable | disable}

Enables or disables the SW-LCAS H4 byte sequence number re-sequencing feature. If an ML-Series card running Software Release 4.6.2 or later is interoperating with an ML-Series card running Software Release 4.6.0 or 4.6.1, then the **pos vcat resequence disable** command must be added to the configuration of the ML-Series card running R4.6.2 or later.

Syntax Description	Parameter	Description
	Enable	Enables the re-sequencing of the H4 byte sequence numbers when a member is added to the VCAT group or removed from the VCAT group. If both members are up, then member #0 will have sequence number of zero (0) and member #1 will have sequence number one (1). If only one member is up, then the sequence number of that member will be zero (0).
	Disables	Disables the re-sequencing of the H4 byte sequence numbers when a member is added to the VCAT group or removed from the VCAT group. Member #0 will always have a sequence number of zero (0) and member #1 will always have a sequence number of one (1).

Defaults The default setting is Enable.

Command Modes Per POS port configuration

Usage Guidelines The no form of the command will set the mode to the default.

Examples The following example disables the re-sequencing of the H4 byte sequence numbers for POS port 0:

```
Router(config)# int pos 0
Router(config)# pos vcat resequence disable
```

Related Commands None

show controller pos *interface-number* [details]

Use this command to display the status of the POS controller. Use the details argument to obtain certain additional information.

Syntax Description	Parameter	Description
	<i>interface-number</i>	Number of the POS interface (0–1)

Defaults N/A

Command Modes Privileged EXEC

Usage Guidelines This command can be used to help diagnose and isolate POS or SONET problems.

Examples

Continuous Concatenation Circuit (CCAT) Show Controller Output Example

```
Router# show controller pos 0
Interface POS0
Hardware is Packet/Ethernet over Sonet
Concatenation: CCAT
Circuit state: IS
PATH
  PAIS      = 0          PLOP      = 0          PRDI      = 0          PTIM      = 0
  PPLM      = 0          PUNEQ     = 0          PPDI      = 0          PTIU      = 0
  BER_SF_B3 = 0          BER_SD_B3 = 0          BIP(B3)   = 20         REI       = 2
  NEWPTR    = 0          PSE       = 0          NSE       = 0

Active Alarms : None
Demoted Alarms: None
Active Defects: None
Alarms reportable to CLI: PAIS PLOP PUNEQ PTIM PPLM PRDI PPDI BER_SF_B3 BER_SD_B3
VCAT_OOU_TPT LOM SQM
Link state change defects: PAIS PLOP PUNEQ PTIM PPLM PRDI PPDI BER_SF_B3
Link state change time   : 200 (msec)

DOS FPGA channel number : 0
Starting STS (0 based)  : 0
VT ID (if any) (0 based): 255
Circuit size            : VC4
RDI Mode                : 1 bit
C2 (tx / rx)           : 0x01 / 0x01
Framing                 : SDH

Path Trace
Mode                    : off
Transmit String         :
Expected String         :
Received String         :
```

```

Buffer          : Stable
Remote hostname :
Remote interface:
Remote IP addr  :

B3 BER thresholds:
SFBER = 1e-4,   SDBER = 1e-7

5 total input packets, 73842 post-HDLC bytes
0 input short packets, 73842 pre-HDLC bytes
0 input long packets , 0 input runt packets
67 input CRCerror packets , 0 input drop packets
0 input abort packets
0 input packets dropped by ucode

0 total output packets, 0 output pre-HDLC bytes
0 output post-HDLC bytes

Carrier delay is 200 msec

```

VCAT Show Controller Output Example

```

Router# show controller POS 1
Interface POS1
Hardware is Packet/Ethernet over Sonet
Concatenation: VCAT
VCG State: VCG_NORMAL
LCAS Type:NO LCAS
Defect Processing Mode: IMMEDIATE
PDI Holdoff Time: 100 (msec)
Active Alarms : None
Demoted Alarms: None

***** Member 1 *****
ESM State: IS
VCG Member State: VCG_MEMBER_NORMAL
    PAIS      = 0          PLOP      = 0          PRDI      = 0          PTIM      = 0
    PPLM      = 0          PUNEQ     = 0          PPDI      = 0          PTIU      = 0
    BER_SF_B3 = 0          BER_SD_B3 = 0          BIP(B3)   = 16         REI       = 17
    NEWPTR    = 0          PSE       = 0          NSE       = 0

Active Alarms : None
Demoted Alarms: None
Active Defects: None
Alarms reportable to CLI: PAIS PLOP PUNEQ PTIM PPLM PRDI PPDI BER_SF_B3 BER_SD_B3
VCAT_OOU_TPT LOM SQM
Link state change defects: PAIS PLOP PUNEQ PTIM PPLM PRDI PPDI BER_SF_B3
Link state change time : 200 (msec)

DOS FPGA channel number : 2
Starting STS (0 based)  : 3
VT ID (if any) (0 based): 255
Circuit size            : VC4
RDI Mode                 : 1 bit
C2 (tx / rx)            : 0x01 / 0x01
Framing                  : SDH

Path Trace
Mode                     : off

```

show controller pos interface-number [details]

```

Transmit String :
Expected String :
Received String :
Buffer          : Stable
Remote hostname :
Remote interface:
Remote IP addr  :

```

```

B3 BER thresholds:
SFBER = 1e-4,   SDBER = 1e-7

```

```

***** Member 2 *****

```

```

ESM State: IS

```

```

VCG Member State: VCG_MEMBER_NORMAL

```

```

  PAIS      = 0          PLOP      = 0          PRDI      = 0          PTIM      = 0
  PPLM      = 0          PUNEQ     = 0          PPDI      = 0          PTIU      = 0
  BER_SF_B3 = 0          BER_SD_B3 = 0          BIP(B3)   = 15         REI       = 35
  NEWPTR    = 0          PSE       = 0          NSE       = 0

```

```

Active Alarms : None

```

```

Demoted Alarms: None

```

```

Active Defects: None

```

```

Alarms reportable to CLI: PAIS PLOP PUNEQ PTIM PPLM PRDI PPDI BER_SF_B3 BER_SD_B3

```

```

VCAT_OOU_TPT LOM SQM

```

```

Link state change defects: PAIS PLOP PUNEQ PTIM PPLM PRDI PPDI BER_SF_B3

```

```

Link state change time   : 200 (msec)

```

```

DOS FPGA channel number : 3
Starting STS (0 based)  : 24
VT ID (if any) (0 based): 255
Circuit size            : VC4
RDI Mode                 : 1 bit
C2 (tx / rx)            : 0x01 / 0x01
Framing                  : SDH

```

```

Path Trace

```

```

Mode          : off
Transmit String :
Expected String :
Received String :
Buffer        : Stable
Remote hostname :
Remote interface:
Remote IP addr  :

```

```

B3 BER thresholds:
SFBER = 1e-4,   SDBER = 1e-7

```

```

13 total input packets, 5031 post-HDLC bytes
0 input short packets, 5031 pre-HDLC bytes
0 input long packets , 0 input runt packets
0 input CRCerror packets , 0 input drop packets
0 input abort packets
0 input packets dropped by ucode

```

```

13 total output packets, 5031 output pre-HDLC bytes
5031 output post-HDLC bytes

```

```

Carrier delay is 200 msec

```


Related Commands

show interface pos
clear counters

show interface pos *interface-number*

Use this command to display the status of the POS.

Syntax Description	Parameter	Description
	<i>interface-number</i>	Number of the POS interface (0-1)

Defaults N/A

Command Modes Privileged EXEC

Usage Guidelines This command can be used to help diagnose and isolate POS or SONET/SDH problems.

Examples

```

Gateway# show interfaces pos0
POS0 is up, line protocol is up
  Hardware is Packet/Ethernet over Sonet
  Description: foo bar
  MTU 4470 bytes, BW 155520 Kbit, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, crc 32, loopback not set
  Keepalive set (10 sec)
  Scramble enabled
  Last input 00:00:09, output never, output hang never
  Last clearing of "show interface" counters 05:17:30
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue :0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec

    2215 total input packets, 223743 post-HDLC bytes
    0 input short packets, 223951 pre-HDLC bytes
    0 input long packets , 0 input runt packets
    0 input CRCerror packets , 0 input drop packets
    0 input abort packets
    0 input packets dropped by ucode

    0 packets input, 0 bytes
  Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 parity
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort

  2216 total output packets, 223807 output pre-HDLC bytes
  224003 output post-HDLC bytes

    0 packets output, 0 bytes, 0 underruns
    0 output errors, 0 applique, 8 interface resets
    0 output buffer failures, 0 output buffers swapped out
    0 carrier transitions

```

Related Commands

show controller pos
clear counters

show ons alarm

Use this command to display all the active alarms on the card.

Syntax Description This command has no arguments or keywords.

Defaults N/A

Command Modes Privileged EXEC

Usage Guidelines This command can be used to help diagnose and isolate card problems.

Examples

```
router# show ons alarm
Equipment Alarms
Active: CONTBUS-IO-A CTNEQPT-PBWORK
```

```
Port Alarms
  POS0 Active: None
  POS1 Active: None
  FastEthernet0 Active: None
  FastEthernet1 Active: None
  FastEthernet2 Active: None
  FastEthernet3 Active: None
  FastEthernet4 Active: None
  FastEthernet5 Active: None
  FastEthernet6 Active: None
  FastEthernet7 Active: None
  FastEthernet8 Active: None
  FastEthernet9 Active: None
  FastEthernet10 Active: None
  FastEthernet11 Active: None
```

```
POS0
```

```
Active Alarms : None
Demoted Alarms: None
```

```
POS1 VCG State: VCG_NORMAL
VCAT Group
Active Alarms : None
Demoted Alarms: None
```

```
Member 0
Active Alarms : None
Demoted Alarms: None
```

```
Member 1
Active Alarms : None
Demoted Alarms: None
```

Related Commands

- show controller pos
- show ons alarm defects
- show ons alarm failures

show ons alarm defect eqpt

This command displays the equipment layer defects.

Syntax Description This command has no arguments or keywords.

Defaults N/A

Command Modes Privileged EXEC

Usage Guidelines This command displays the set of active defects for the equipment layer and the possible set of defects that can be set.

Examples

```
router# show ons alarm defect eqpt
Equipment Defects
Active: CONTBUS-IO-B
Reportable to TCC/CLI: CONTBUS-IO-A CONTBUS-IO-B CTNEQPT-PBWORK CTNEQPT-PBPROT EQPT
RUNCFG-SAVENEED ERROR-CONFIG
```

Related Commands show ons alarm failures

show ons alarm defect port

This commands displays the port layer defects.

Syntax Description This command has no arguments or keywords.

Defaults N/A

Command Modes Privileged EXEC

Usage Guidelines This commands displays the set of active defects for the link layer and the possible set of defects that can be set. Note that the TPTFAIL defect can only occur on the POS ports and the CARLOSS defect can only occur on the Ethernet ports.

Examples

```
router# show ons alarm defect port
Port Defects
  POS0
  Active: TPTFAIL
  Reportable to TCC: CARLOSS TPTFAIL
  POS1
  Active: TPTFAIL
  Reportable to TCC: CARLOSS TPTFAIL
  GigabitEthernet0
  Active: None
  Reportable to TCC: CARLOSS TPTFAIL
  GigabitEthernet1
  Active: None
  Reportable to TCC: CARLOSS TPTFAIL
```

Related Commands show interface
show ons alarm failures

show ons alarm defect pos *interface-number*

This commands displays the link layer defects.

Syntax Description	Parameter	Description
	<i>interface-number</i>	Number of the interface (0–1)
Defaults	N/A	
Command Modes	Privileged EXEC	
Usage Guidelines	This commands displays the set of active defects for the POS layer and the possible set of defects that can be set.	
Examples	<pre>router# show ons alarm defect pos0 POS0 Active Defects: None Alarms reportable to TCC/CLI: PAIS PRDI PLOP PUNEQ PPLM PTIM PPDI BER_SF_B3 BER_SD_B3</pre>	
Related Commands	show controller pos show ons alarm failures	

show ons alarm failure eqpt

This command displays the equipment layer failures.

Syntax Description This command has no arguments or keywords.

Defaults N/A

Command Modes Privileged EXEC

Usage Guidelines This command displays the set of active failures for the equipment layer. If an EQPT alarm is present, the Board Fail defect that was the source of the alarm is displayed.

Examples

```
router# show ons alarm failure eqpt
Equipment
Active Alarms: None
```

Related Commands show ons alarm defect

show ons alarm failure port

This command displays the port layer failures.

Syntax Description This command has no arguments or keywords.

Defaults N/A

Command Modes Privileged EXEC

Usage Guidelines This command displays the set of active failures for the link layer.

Examples

```
router# show ons alarm failure port
Port Alarms
  POS0 Active: TPTFAIL
  POS1 Active: TPTFAIL
  GigabitEthernet0 Active: None
  GigabitEthernet1 Active: None
```

Related Commands

- show interface
- show ons alarm defect

show ons alarm failure pos *interface-number*

This commands displays the link layer failures.

Syntax Description	Parameter	Description
	<i>interface-number</i>	Number of the interface (0-1)

Defaults N/A

Command Modes Privileged EXEC

Usage Guidelines This commands displays the set of active failures for a specific interface at the POS layer. The display also specifies if an alarm has been demoted, as defined in Telcordia GR-253.

Examples

```
router# show ons alarm failure pos 0
POS0
Active Alarms : None
Demoted Alarms: None
```

Related Commands

- show controller pos
- show ons alarm defect

spr drpri-id { 0 | 1 }

Creates a DRPRI identification number of 0 or 1 to differentiate between the ML-Series cards paired for the DRPRI protection feature.

Defaults

N/A

Command Modes

Shared packet ring (SPR) interface configuration

Usage Guidelines

DRPRI paired sets share the same SPR station ID, so the DRPRI identification number helps identify a particular card in a DRPRI pair.

Examples

The following example assigns a DRPRI identification number of zero to the SPR interface on an ML-Series card:

```
Router(config)# interface spr 1  
Router(config-if)# spr drpri-id 0
```

Related Commands

interface spr 1
spr-intf-id
spr station-id
spr wrap

spr-intf-id *shared-packet-ring-number*

Assigns the POS interface to the SPR interface.

Syntax Description	Parameter	Description
	shared-packet-ring-number	The only valid shared-packet-ring-number (SPR number) is 1.

Defaults N/A

Command Modes POS interface configuration

Usage Guidelines

- The SPR number must be 1, which is the same SPR number assigned to the SPR interface.
- The members of the SPR interface must be POS interfaces.
- An SPR interface is configured similarly to a EtherChannel (port-channel) interface. Instead of using the **channel-group** command to define the members, you use the **spr-intf-ID** command. Like port-channel, you then configure the SPR interfaces instead of the POS interface.

Examples The following example assigns an ML-Series card POS interface to an SPR interface with a shared-packet-ring-number of 1:

```
Router(config)# interface pos 0
Router(config-if)# spr-intf-id 1
```

Related Commands

```
interface spr 1
spr drpri-id
spr station-id
spr wrap
```

[no] spr load-balance { auto | port-based }

Specifies the RPR load-balancing scheme for Unicast packets.

Syntax Description	Parameter	Description
	auto	The default <i>auto</i> option balances the load based on the MAC addresses or source and destination addresses of the IP packet.
	port-based	The <i>port-based</i> load balancing option maps even ports to the POS 0 interface and odd ports to the POS 1 interface.

Defaults The default setting is auto.

Command Modes SPR interface configuration

Examples The following example configures an SPR interface to use port-based load balancing:

```
Router(config)# interface spr 1
Router(config-if)# spr load-balance port-based
```

Related Commands interface spr 1

spr station-id *station-id-number*

Configures a station ID.

Syntax Description	Parameter	Description
	station-id-number	The user must configure a different number for each SPR interface that attaches to the RPR. Valid station ID numbers range from 1 to 254.

Defaults N/A

Command Modes SPR interface configuration

Usage Guidelines The different ML-Series cards attached to the RPR all have the same interface type and number, spr1. The station ID helps to differentiate the SPR interfaces.

Examples The following example sets an ML-Series card SPR station ID to 100:

```
Router(config)# interface spr 1
Router(config-if)# spr station-id 100
```

Related Commands

- interface spr 1
- spr drpri-id
- spr-intf-id
- spr wrap

sbr wrap immediate | delayed

Sets the RPR wrap mode to either wrap traffic the instant it detects a link state change or to wrap traffic after the carrier delay, which gives the SONET protection time to register the defect and declare the link down.

Syntax Description	Parameter	Description
	immediate	Wraps RPR traffic the instant it detects a link state change.
	delayed	Wraps RPR traffic after the carrier delay time expires.

Defaults The default setting is immediate.

Command Modes SPR interface configuration

Usage Guidelines Immediate should be used if RPR is running over unprotected SONET circuits. Delayed should be run for SONET protected circuits (BLSR or path protection).

Examples The following example sets an ML-Series card to delayed:

```
Router(config)# interface sbr 1
Router(config-if)# sbr wrap delayed
```

Related Commands

- interface sbr 1
- sbr drpri-id
- sbr-intf-id
- sbr station-id



Unsupported CLI Commands

This appendix lists some of the command-line interface (CLI) commands that are not supported in this release, either because they are not tested, or because of hardware limitations. These unsupported commands are displayed when you enter the question mark (?) at the CLI prompt. This is not a complete list. Unsupported commands are listed by command mode.

Unsupported Privileged Exec Commands

```
clear ip accounting
show ip accounting
show ip cache
show ip tcp header-compression
show ip mcache
show ip mpacket
show controller pos pm
show controller pos [variable] pm
```

Unsupported Global Configuration Commands

```
access-list aaa <1100-1199>
access-list aaa <200-299>
access-list aaa <700-799>
async-bootp
boot
bridge <num> acquire
bridge <num> address
bridge cmf
bridge <num> bitswap-layer3-addresses
bridge <num> circuit-group
```

bridge <num> domain
bridge <num> lat-service-filtering
bridge <num> protocol dec
bridge <num> protocol ibm
bridge <num> protocol vlan-bridge
chat-script
class-map match access-group
class-map match class-map
class-map match destination-address
class-map match mpls
class-map match protocol
class-map match qos-group
class-map match source-address
clns
define
dialer
dialer-list
downward-compatible-config
file
ip access-list log-update
ip access-list logging
ip address-pool
ip alias
ip bootp
ip gdp
ip local
ip reflexive-list
ip security
ip source-route
ip tcp
ipc
map-class
map-list
multilink
netbios
partition
policy-map class queue-limit
priority-list

queue-list
router iso-igrp
router mobile
service compress-config
service disable-ip-fast-frag
service exec-callback
service nagle
service old-slip-prompts
service pad
service slave-log
subscriber-policy

Unsupported POS Interface Configuration Commands

access-expression
autodetect
bridge-group x circuit-group
bridge-group x input-
bridge-group x lat-compression
bridge-group x output-
bridge-group x subscriber-loop-control
clock
clns
custom-queue-list
down-when-looped
fair-queue
flowcontrol
full-duplex
half-duplex
hold-queue
ip accounting
ip broadcast-address
ip load-sharing per-packet
ip route-cache
ip security
ip tcp
ip verify
iso-igrp

loopback
multilink-group
netbios
pos flag c2
pos mode gfp
priority-group
pulse-time
random-detect
rate-limit
rmon
serial
service-policy history
source
timeout
transmit-interface
tx-ring-limit

Unsupported FastEthernet or GigabitEthernet Interface Configuration Commands

access-expression
clns
custom-queue-list
fair-queue
hold-queue
ip accounting
ip broadcast-address
ip load-sharing per-packet
ip route-cache
ip security
ip tcp
ip verify
iso-igrp
keepalive
loopback
max-reserved-bandwidth
multilink-group

netbios
priority-group
random-detect
rate-limit
service-policy history
timeout
transmit-interface
tx-ring-limit

Unsupported Port-Channel Interface Configuration Commands

access-expression
carrier-delay
cdp
clns
custom-queue-list
duplex
down-when-looped
encapsulation
fair-queue
flowcontrol
full-duplex
half-duplex
hold-queue
iso-igrp
keepalive
max-reserved-bandwidth
multilink-group
negotiation
netbios
ppp
priority-group
rate-limit
random-detect
timeout
tx-ring-limit

Unsupported BVI Interface Configuration Commands

access-expression
carrier-delay
cdp
clns
flowcontrol
hold-queue
iso-igrp
keepalive
l2protocol-tunnel
load-interval
max-reserved-bandwidth
mode
multilink-group
netbios
ntp
mtu
rate-limit
timeout
transmit-interface
tx-ring-limit



Using Technical Support

This appendix describes how to resolve problems with your ML-Series card.

The appendix contains the following sections:

- [Gathering Information About Your Internetwork, page C-1](#)
- [Getting the Data from Your ML-Series Card, page C-2](#)
- [Providing Data to Your Technical Support Representative, page C-3](#)

To help resolve these problems, use the “[Gathering Information About Your Internetwork](#)” section on [page C-1](#) as a guideline for gathering relevant information about your network prior to calling.



Note

When you have a problem that you cannot resolve, contact the Cisco Technical Assistance Center (Cisco TAC).

Gathering Information About Your Internetwork

Before gathering any specific data, compile a list of all symptoms that users have reported on the internetwork (such as connections dropping or slow host response).

The next step is to gather specific information. Typical information needed to troubleshoot internetworking problems falls into two general categories: information required for any situation; and information specific to the topology, technology, or protocol.

Information that is always required by technical support engineers includes the following:

- Network topology map for the data network and the SONET/SDH topology and provisioning.
- List of hosts and servers: Include the host and server type, number on network, and a description of the host operating systems that are implemented.
- Configuration listing of all switch routers and switches involved.
- Complete specifications of all switch routers and switches involved.
- Version numbers of software (obtained with the **show version** command) and Flash code (obtained with the **show controllers** command) on all relevant switch routers and switches.
- List of network layer protocols, versions, and vendors.
- List of alarms and conditions on all nodes in the SONET/SDH topology.
- Node equipment and configuration; including type of cross-connect cards, ML-Series cards' slot numbers, OC-N cards, and TCC2/TCC2P cards.

To assist you in gathering this required data, the **show tech-support** EXEC command has been added in Cisco IOS Release 11.1(4) and later. This command provides general information about the switch router that you can provide to your technical support representative when you are reporting a problem.

The **show tech-support** command outputs the equivalent of the **show version**, **show running-config**, **show controllers**, **show stacks**, **show interfaces**, **show buffers**, **show process memory**, and **show process** EXEC commands.

The specific information requirements that might be needed by technical support vary depending on the situation. They include the following:

- Output from the following general **show** commands:
 - show interfaces**
 - show controllers**
 - show processes {cpu | mem}**
 - show buffer**
 - show mem summary**
- Output from the following protocol-specific **show** commands:
 - show protocol route**
 - show protocol traffic**
 - show protocol interfaces**
 - show protocol arp**
- Output from provisioning show commands
- Output from relevant **debug** privileged EXEC commands
- Output from protocol-specific **ping** and **trace** diagnostic tests, as appropriate
- Network analyzer traces, as appropriate
- Core dumps obtained using the **exception dump** command, or using the **write core** command if the system is operational, as appropriate

Getting the Data from Your ML-Series Card

When obtaining the information from your ML-Series card, you must tailor your method to the system that you are using to retrieve the information. Following are some hints for different platforms:

- PC and Macintosh—Connect a PC or Macintosh to the console port of the ML-Series card and log all output to a disk file (using a terminal emulation program). The exact procedure varies depending on the communication package used with the system.
- Terminal connected to the console port or remote terminal—The only way to get information with a terminal connected to the console port or with a remote terminal is to attach a printer to the AUX port on the terminal (if one exists) and to force all screen output to go to the printer. Using a terminal is undesirable because there is no way to capture the data to a file.
- UNIX workstation—At the UNIX prompt, enter the command **script filename**, then use Telnet to connect to the ML-Series card. The UNIX **script** command captures all screen output to the specified filename. To stop capturing output and close the file, enter the end-of-file character (typically **Ctrl-D**) for your UNIX system.

**Note**

To get your system to automatically log specific error messages or operational information to a UNIX syslog server, enter the **logging** *internet-address* command. For more information about using the **logging** command and setting up a syslog server, refer to the Cisco IOS configuration guides and command references.

Providing Data to Your Technical Support Representative

When submitting information to your technical support representative, electronic data is preferred. Electronic data significantly eases the transfer of information between technical support personnel and development staff. Common electronic formats include data sent through electronic mail and files sent using FTP.

If you are submitting data to your technical support representative, use the following list (in order of most to least favorable) to determine the preferred method for submission:

- The preferred method of information submission is through FTP service over the Internet. If your environment supports FTP, you can place your file in the incoming directory on the host Cisco.com.
- The next best method is to send data by e-mail. Before using this method, be sure to contact your technical support representative, especially when transferring binary core dumps or other large files.
- Transfer through a PC-based communications protocol, such as Kermit, to upload files to Cisco.com. Again, be sure to contact your technical support representative before attempting any transfer.
- Transfer by disk or tape.
- The least favorable method is hard-copy transfer by fax or physical mail.

**Note**

If you use e-mail, do not use encoding methods such as binhex or zip. Only MIME-compliant mail should be used.



Numerics

802.1D. *See* STP
802.1Q. *See* IEEE 802.1Q

A

abbreviating commands [3-18](#)
ABRs [11-9](#)
access control lists. *See* ACL
access-list command [20-8](#)
ACL
 about [16-1](#)
 applying ACLs [16-4](#)
 creating
 extended IP ACLs [16-3](#)
 IP ACLs [16-3](#)
 named extended IP ACLs [16-4](#)
 named IP ACLs [16-3](#)
 named standard IP ACLs [16-4](#)
 numbered standard IP ACLs [16-3](#)
 implementation guidelines IP ACL [16-2](#)
 named IP ACL [16-2](#)
adapter cable [3-5](#)
addresses
 dynamic
 accelerated aging [7-9](#)
 default aging [7-9](#)
 multicast, STP address management [7-8](#)
administrative distances
 OSPF [11-17](#)
 routing protocol defaults [11-32](#)
advertisements RIP [11-5](#)

aging time, accelerated for STP [7-9, 7-20](#)
alarms [5-6](#)
alarms, RMON [19-3](#)
area border routers. *See* ABRs
ASBRs [11-9](#)
autonomous system boundary routers. *See* ASBRs

B

bandwidth command traffic classes [14-13, 18-4](#)
BGP, about [11-27](#)
Border Gateway Protocol. *See* BGP
BPDU RSTP format [7-13](#)
bridge-group command [4-4, 4-5, 4-10, 6-2](#)
bridge groups, routing [12-1](#)
bridge-group virtual interface. *See* BVIs
bridge irb command [12-3](#)
bridge priority command [6-2](#)
bridge protocol command [6-2, 18-9, 18-10](#)
bridging
 configuring [6-3](#)
 feature list [1-2](#)
 monitoring and verifying [6-3](#)
bvi command [12-3](#)
BVIs
 configuring [12-3](#)
 description [12-1](#)
 displaying information about [12-5](#)
 routing enabled on [12-2](#)

C

cable, RJ-11 to RJ-45 adapter [3-5](#)

- card description [1-1](#)
- CDP, Layer 2 protocol tunneling [9-9](#)
- CE-100T-8
 - capacity restrictions [23-9](#)
 - Enhanced State Model (ESM) [23-4](#)
 - Ethernet features [23-1](#)
 - frame buffering [23-2](#)
 - IEEE 802.1Q [23-4](#)
 - IS, AINS [23-4](#)
 - link integrity [23-3](#)
 - overview [23-1](#)
 - pools [23-9](#)
 - priority queuing (ToS and CoS) [23-4](#)
 - statistics and counters [23-6](#)
- CE100T-8
 - MTU [23-2](#)
- CE-100T-8 card
 - maximizing bandwidth [23-9](#)
 - STS/VT allocation tab [23-9](#)
- channel-group command [10-3, 10-5](#)
- circuits
 - definition [22-8](#)
- Cisco HDLC [21-6](#)
- Cisco IOS
 - backing out one level [3-18](#)
 - command modes [3-17 to 3-19](#)
 - console configuration mode [3-18](#)
 - global configuration mode [3-17](#)
 - interface configuration mode [3-18](#)
 - listing commands [3-19](#)
 - privileged EXEC mode [3-17](#)
 - software basics [3-17](#)
 - startup configuration file [3-9](#)
 - upgrading image [1-5](#)
 - user EXEC mode [3-17](#)
- Cisco IOS software image [3-2](#)
- CiscoWorks 2000 [20-4](#)
- clear bridge command [6-4](#)
- clear vlan command [8-5](#)
- clear vlan statistics command [6-4](#)
- clocking tolerances [21-11](#)
- commands
 - access-list [20-8](#)
 - bridge-group [4-4, 4-5, 4-10, 6-2](#)
 - bridge irb [12-3](#)
 - bridge priority [6-2](#)
 - bridge protocol [6-2, 18-9, 18-10](#)
 - bridge protocol drpri-rstp [24-2](#)
 - channel-group [10-3, 10-5](#)
 - clear bridge [6-4](#)
 - clear vlan [8-5](#)
 - clear vlan statistics [6-4](#)
 - debug vlan packet [8-5](#)
 - hostname [3-9](#)
 - interface bvi [12-3](#)
 - interface spr 1 [24-4](#)
 - ip multicast-routing [11-34](#)
 - ip pim [11-34](#)
 - line vty [3-9](#)
 - listing [3-19](#)
 - network area [11-3](#)
 - reference chapter [24-1](#)
 - rmon alarm [19-3](#)
 - rmon collection history [19-5](#)
 - rmon collection stats [19-6](#)
 - rmon event [19-3](#)
 - router bgp [11-3](#)
 - router eigrp [11-2](#)
 - sdm size [15-3](#)
 - show bridge [6-4](#)
 - show bridge group [6-4](#)
 - show interfaces bvi [12-5](#)
 - show interfaces irb [12-5](#)
 - show interfaces port-channel [10-9](#)
 - show ip mroute [11-35](#)
 - show rmon [19-22](#)
 - show rmon alarms [19-22](#)
 - show rmon events [19-22](#)

- show rmon history [19-22](#)
- show rmon statistics [19-22](#)
- show sdm size [15-3](#)
- show snmp [20-14](#)
- show snmp group [20-14](#)
- show snmp pending [20-14](#)
- show snmp sessions [20-14](#)
- show snmp user [20-14](#)
- show tech-support [26-2](#)
- show vlan [8-5](#)
- snmp-server community [20-7](#)
- snmp-server contact [20-12](#)
- snmp-server enable traps [20-11](#)
- snmp-server engineID [20-9](#)
- snmp-server group [20-9](#)
- snmp-server host [20-11](#)
- snmp-server location [20-12](#)
- snmp-server queue-length [20-11](#)
- snmp-server tftp-server-list [20-13](#)
- snmp-server trap-source [20-11](#)
- snmp-server trap-timeout [20-11](#)
- snmp-server user [20-9](#)
- spri drpri-id [24-26](#)
- spri-intf-id [24-27](#)
- spr station-id [24-29](#)
- spr wrap [24-30](#)
- community strings
 - configuring [20-7](#)
 - overview [20-4](#)
- configuration examples
 - RPR [17-8, 17-15](#)
 - SNMP [20-13](#)
- configuration files
 - limiting TFTP server access [20-13](#)
 - system contact and location information [20-12](#)
- configuration guidelines
 - SNMP [20-6](#)
- configuration mode
 - console [3-18](#)
 - global [3-17](#)
- configuring
 - BVIs [12-3](#)
 - EtherChannel encapsulation [10-7](#)
 - Fast EtherChannel [10-1](#)
 - host name [3-9](#)
 - integrated routing and bridging. *See* IRB
 - interface, overview [4-1](#)
 - IP [11-1](#)
 - IP multicast [11-33](#)
 - ISL over FEC [10-7](#)
 - management port [3-8](#)
 - VLANs [8-1](#)
- connecting to console port [3-5](#)
- connection procedures [3-5 to 3-6](#)
- console port, connecting to [3-5](#)
- CoS-based Packet Statistics [14-28](#)
- CoS-based QoS [14-16](#)
- cos commit command [14-16](#)
- CRC errors
 - accessing through SNMP [19-16](#)
 - checking manually [19-21](#)
 - configuring SNMP traps [19-17](#)
 - monitoring [19-16](#)
 - threshold configuration guidelines [19-16](#)
- CTC
 - Cisco IOS on CTC [3-2](#)
 - Ethernet port provisioning information [2-2](#)
 - POS port provisioning information [2-4](#)
 - POS statistics [2-1](#)
 - SONET alarms [2-6](#)
 - SONET circuit provisioning [2-8](#)

D

- Database restore [3-11](#)
- debug vlan packet command [8-5](#)
- default configuration
 - EIGRP [11-21](#)

- Layer 2 protocol tunneling [9-10](#)
 - OSPF [11-10](#)
 - RIP [11-5](#)
 - RMON [19-2](#)
 - SNMP [20-6](#)
 - STP [7-16](#)
 - Default Multicast QoS [14-23](#)
 - dense mode, PIM [11-34](#)
 - Diffusing Update Algorithm (DUAL) [11-20](#)
 - double-tagged packets
 - IEEE 802.1Q tunneling [9-2](#)
 - Layer 2 protocol tunneling [9-10](#)
 - drop
 - definition [22-8](#)
 - DRPRI
 - characteristics [17-36](#)
 - configuring [17-37](#)
 - example [17-39](#)
 - monitoring and verifying [17-43](#)
 - overview [1-5](#)
 - understanding [17-36](#)
 - DUAL finite state machine, EIGRP [11-20](#)
 - dynamic addresses. *See* addresses
-
- E**
- Egress priority marking [14-8](#)
 - EIGRP
 - authentication [11-25](#)
 - components [11-20](#)
 - configuring [11-22](#)
 - default configuration [11-21](#)
 - definition [11-20](#)
 - interface parameters, configuring [11-23](#)
 - monitoring [11-26](#)
 - e-mail, technical support [26-3](#)
 - enable mode [3-17](#)
 - enable passwords [3-8](#)
 - enable secret passwords [3-8](#)
 - encapsulation
 - configuring EtherChannels [10-7](#)
 - configuring IEEE 802.1Q VLANs [8-2](#)
 - Enhanced IGRP. *See* EIGRP
 - Enhanced performance monitoring [14-28](#)
 - Enhanced State Model (ESM) [22-6, 23-4](#)
 - error messages, logging [26-3](#)
 - E-Series card
 - applications [22-13](#)
 - circuit protection [22-24](#)
 - EtherSwitch
 - multicard [22-14](#)
 - single-card [22-14](#)
 - flow control [22-16](#)
 - hub-and-spoke Ethernet circuit [22-26](#)
 - IEEE 802.1Q [22-18](#)
 - IEEE 802.3z flow control [22-16](#)
 - Layer 2 switching [22-14](#)
 - linear mapper [22-15](#)
 - manual cross-connect [22-27](#)
 - multicard EtherSwitch [22-14](#)
 - point-to-point circuit [22-24](#)
 - port-mapped [22-15](#)
 - priority queuing [22-20](#)
 - proprietary encapsulation [21-7](#)
 - Q-tagging [22-18](#)
 - RMON alarm thresholds [22-27](#)
 - shared packed ring [22-25](#)
 - single-card EtherSwitch [22-14](#)
 - spanning tree (STP) [22-21](#)
 - VLAN counter [22-17](#)
 - VLAN support [22-17](#)
 - EtherChannel
 - configuring encapsulation [10-7](#)
 - ISL VLANs [10-1](#)
 - port channels supported [10-1](#)
 - Ethernet
 - applications [23-1](#)
 - clocking [21-11](#)

- flow control on CE-100T-8 [23-2](#)
- frame buffering [23-2](#)
- Ethernet configuration tasks [4-4](#)
- Ethernet Wire Service (EWS) [9-7](#)
- events, RMON [19-3](#)
- extended system ID, STP [7-4](#)

F

- Fast EtherChannel. *See* FEC
- Fast Ethernet
 - configuring autonegotiation [4-4](#)
 - configuring interfaces [4-4](#)
 - Fast EtherChannel [10-1](#)
- feature list [1-2](#)
- FEC
 - cautions [10-2](#), [10-5](#), [13-3](#)
 - configuring [10-2](#), [10-4](#), [13-2](#)
 - configuring encapsulation [10-7](#)
 - configuring ISL [10-7](#)
 - port channels supported [10-1](#)
- flow control [23-2](#)
- FPGA [2-7](#)

G

- GEC
 - bandwidth scalability [10-2](#)
 - configuring [10-2](#), [10-4](#), [13-2](#)
 - configuring encapsulation [10-7](#)
- get-bulk-request operation [20-3](#)
- get-next-request operation [20-3](#), [20-4](#)
- get-request operation [20-3](#), [20-4](#)
- get-response operation [20-3](#)
- GFP-F framing [1-5](#), [21-7](#)
- Gigabit EtherChannel. *See* GEC
- Gigabit Ethernet
 - configuring autonegotiation [4-5](#), [4-10](#)

- configuring interfaces [4-5](#), [4-10](#)
- global configuration mode [3-17](#)
- G-Series card
 - application [22-1](#)
 - autonegotiation [22-4](#)
 - circuit restrictions [22-7](#)
 - circuits [22-6](#)
 - Enhanced State Model (ESM) [22-6](#)
 - flow control watermark provisioning [22-4](#)
 - frame buffering [22-3](#)
 - Gigabit EtherChannel (GEC) [22-4](#)
 - IS, AINS [22-6](#)
 - link integrity [22-5](#)
 - manual cross-connect [22-7](#)
 - point-to-point Ethernet circuit [22-7](#)
 - separate autonegotiation and flow control [22-4](#)
 - STS-24c/VC4-8c restrictions [22-7](#)
 - transponder mode [22-8](#)

H

- hard reset on ML-Series [3-2](#)
- hostname command [3-9](#)
- HSRP, EtherChannel compatibility with [10-1](#)

I

- IEEE [9-4](#)
- IEEE 802.1D. *See* STP
- IEEE 802.1Q tunneling
 - compatibility with other features [9-4](#)
 - defaults [9-4](#)
 - described [9-1](#)
- ifIndex [19-16](#)
- IGMP [11-33](#)
- IGP [11-9](#)
- Ingress priority marking [14-8](#)
- integrated routing and bridging. *See* IRB

interface configuration mode [3-18](#)

interface parameters, configuring

- EtherChannel [10-2, 10-5, 13-2](#)
- general [4-3](#)
- overview [4-1](#)

interface port IDs [4-2](#)

Interior Gateway Protocol. *See* IGP

Internet Group Membership Protocol. *See* IGMP

Internet protocol multicast. *See* IP multicast routing

Inter-Switch Link protocol. *See* ISL

IOS. *See* Cisco IOS

IOS commands [24-1](#)

IP access control list. *See* ACL

IP multicast routing

- description [11-33](#)
- IGMP [11-33](#)
- PIM [11-33](#)

ip multicast-routing command [11-34](#)

ip pim command [11-34](#)

IP routes, monitoring [11-33](#)

IP routing protocols, configuration tasks [11-1](#)

IP unicast routing

- administrative distances [11-32](#)
- configuring static routes [11-31](#)
- IGP [11-9](#)

IPX [23-1](#)

IRB

- BVIs [12-1](#)
- configuration considerations [12-1](#)
- configuring [12-2](#)
- description [12-1](#)
- displaying information about [12-5](#)
- monitoring and verifying [12-4](#)

IS, AINS [22-6, 23-4](#)

ISL [10-1](#)

J

J1 bytes [2-8](#)

K

keepalive command [5-6](#)

Kermit protocol [26-3](#)

L

Layer 2 feature list [1-2](#)

Layer 2 protocol tunneling [9-10](#)

- configuring [9-10](#)
- default configuration [9-10](#)
- defined [9-10](#)
- guidelines [9-11](#)

Layer 3 feature list [1-4](#)

LCAS

- CE-100T-8 [23-10](#)

LEX encapsulation [21-5](#)

line vty command [3-9](#)

link integrity [23-3](#)

link state advertisements (LSAs) [11-14](#)

logging command [26-3](#)

logging router output [26-2](#)

M

MAC addresses [4-2](#)

management options

- SNMP [20-1](#)

management ports

- See also* console ports
- configuring [3-8](#)

match any command [14-11](#)

match cos command [14-11](#)

match ip dscp command [14-11](#)

match ip precedence command [14-11](#)

Media Access Control addresses. *See* MAC addresses

message logging [26-3](#)

metro tags [9-2](#)

MIBs

- overview [20-1](#)
- SNMP interaction with [20-4](#)
- microcode images [3-11](#)
- Modular QoS Command-Line Interface
 - configuration (example) [14-17](#)
 - configuration, verifying [14-16](#)
 - configuring [14-10](#)
- monitoring
 - EIGRP [11-26](#)
 - IEEE 802.1Q tunneling [9-12](#)
 - IP routes [11-33](#)
 - Layer 2 protocol tunneling [9-12](#)
 - OSPF [11-19, 11-32](#)
 - traffic flow [19-1](#)
 - tunneling [9-12](#)
- MSTP, interoperability with IEEE 802.1D [7-15](#)
- MST protocol tunneling [9-10](#)
- multicast, IP. *See* IP multicast routing
- Multicast priority queuing [14-23](#)
- Multicast QoS [14-23](#)

N

- neighbor discovery/recovery, EIGRP [11-20](#)
- network element default [23-2](#)
- networking protocols, IP multicast routing [11-33 to 11-34](#)
- network management
 - RMON [19-1](#)
 - SNMP [20-1](#)
- not-so-stubby areas. *See* NSSA
- NSSA, OSPF [11-14](#)

O

- OSPF
 - area parameters, configuring [11-14](#)
 - configuring [11-3, 11-11](#)
 - default configuration

- metrics [11-17](#)
- route [11-16](#)
- settings [11-10](#)
- described [11-9](#)
- interface parameters, configuring [11-13](#)
- LSA group pacing [11-18](#)
- monitoring [11-19, 11-32](#)
- network area command [11-3](#)
- process ID [11-3](#)
- router IDs [11-19](#)
- route summarization [11-16](#)
- virtual links [11-16](#)

P

- passive interface OSPF [11-17](#)
- passwords [3-8](#)
- path cost for STP [7-18](#)
- PC, connecting to switch [3-5](#)
- per-VLAN Spanning Tree+ [7-8](#)
- PIM
 - configuring [11-34](#)
 - modes [11-33 to 11-34](#)
 - rendezvous point [11-34](#)
- pin mappings for RJ-11 to RJ-45 [3-5](#)
- port-channel command [10-1](#)
- port channels [10-1](#)
- port IDs [4-2](#)
- port priority, STP [7-17](#)
- POS
 - common ML-Series configurations [5-11](#)
 - configuring interfaces [5-4](#)
 - description [5-1](#)
 - encapsulation types [21-4](#)
 - framing [21-7](#)
 - GFP-F framing [1-5, 21-7](#)
 - interoperability [21-2](#)
 - LEX [21-5](#)
 - overview [21-1](#)

SONET alarms [5-7, 5-8](#)
 pos delay triggers command [5-8](#)
 pos report command [5-7](#)
 pos scramble-atm command [5-9](#)
 PPP/BCP [21-6](#)
 Priority Multicast QoS [14-23](#)
 priority queuing [23-4](#)
 privileged EXEC mode [3-17](#)
 procedures, connection [3-5 to 3-6](#)
 protocol-dependent modules, EIGRP [11-21](#)
 Protocol Independent Multicast. *See* PIM
 PVST+. *See* per-VLAN Spanning Tree+

Q

QinQ [9-1](#)
 QoS policers [14-14](#)
 queuing [23-4](#)

R

reliable transport protocol, EIGRP [11-20](#)
 Remote Network Monitoring. *See* RMON
 remote terminals, logging router output [26-2](#)
 rendezvous points [11-34](#)
 RFC
 1058, RIP [11-4](#)
 1157, SNMPv1 [20-3](#)
 1253, OSPF [11-9](#)
 1493, Bridge MIB [20-5](#)
 1573, IF-MIB [20-5](#)
 1587, NSSAs [11-9](#)
 RFI [4-6](#)
 RIP
 advertisements [11-5](#)
 authentication [11-7](#)
 configuring [11-5](#)
 default configuration [11-5](#)

 described [11-5](#)
 hop counts [11-5](#)
 split horizon [11-8](#)
 summary addresses [11-8](#)
 RJ-11 to RJ-45 console cable adapter [3-5](#)
 RJ-45 connector, console port [3-6](#)
 RMON
 configuring alarms and events [19-3](#)
 configuring traps [19-18](#)
 default configuration [19-2](#)
 displaying status [19-21](#)
 ML-Series [1-6](#)
 Monitoring CRC errors [19-16](#)
 overview [19-1](#)
 statistics
 collecting group Ethernet [19-6](#)
 collecting group history [19-5](#)
 rmon alarm command [19-3](#)
 rmon collection history command [19-5](#)
 rmon collection stats command [19-6](#)
 rmon event command [19-3](#)
 route calculation timers, OSPF [11-17](#)
 router bgp command [11-3](#)
 router eigrp command [11-2](#)
 router ID, OSPF [11-19](#)
 router isis command [11-30](#)
 route summarization, OSPF [11-16](#)
 routing protocol administrative distances [11-32](#)
 RPF [11-34](#)
 RPR
 configuring [17-7](#)
 CoS-based QoS [14-16](#)
 Dual RPR Interconnect. *See* DRPRI
 example [17-8, 17-15](#)
 framing process [17-5](#)
 keep alive, keep alive on RPR [17-31](#)
 Link Fault Propagation (LFP)
 configuring [17-29](#)
 example [17-28](#)

- monitoring and verifying [17-30](#)
 - understanding [17-27](#)
 - MAC address and VLAN support [17-6](#)
 - monitoring and verifying [17-17](#)
 - overview [1-6](#)
 - packet handling operations [17-2](#)
 - QoS [14-10, 17-6](#)
 - ring wrapping [17-3](#)
 - understanding [17-1](#)
- RPRW** [18-20](#)
- RSTP**
- overview [7-9](#)
 - active topology, determining [7-10](#)
 - BPDU
 - format [7-13](#)
 - processing [7-14](#)
 - designated port, defined [7-10](#)
 - designated switch, defined [7-10](#)
 - interoperability with IEEE 802.1D
 - described [7-15](#)
 - topology changes [7-14](#)
 - port roles
 - described [7-9](#)
 - synchronized [7-12](#)
 - proposal-agreement handshake process [7-11](#)
 - rapid convergence
 - point-to-point links [7-11](#)
 - root ports [7-11](#)
 - root port, defined [7-10](#)
-
- S**
- script command [26-2](#)
 - SDH alarms [5-6](#)
 - SDM
 - See also* TCAM
 - configuring
 - autolearn [15-2](#)
 - size [15-2](#)
 - regions [15-1](#)
 - sdm access-list command [15-3](#)
 - sdm size command [15-3](#)
 - service-policy command, traffic policies [14-15](#)
 - service-policy input command [14-16](#)
 - service-policy output command [14-16](#)
 - service-provider networks
 - and customer VLANs [9-2](#)
 - and IEEE 802.1Q tunneling [9-1](#)
 - Layer 2 protocols across [9-10](#)
 - set qos-group command [14-15](#)
 - set-request operation [20-4](#)
 - show bridge command [6-4](#)
 - show bridge group command [6-4](#)
 - show interfaces bvi command [12-5](#)
 - show interfaces irb command [12-5](#)
 - show interfaces port-channel command [10-9](#)
 - show ip mroute command [11-35](#)
 - show policy-map command [14-17](#)
 - show rmon alarms command [19-22](#)
 - show rmon command [19-22](#)
 - show rmon events command [19-22](#)
 - show rmon history command [19-22](#)
 - show rmon statistics command [19-22](#)
 - show sdm size command [15-3](#)
 - show snmp command [20-14](#)
 - show snmp group command [20-14](#)
 - show snmp pending command [20-14](#)
 - show snmp sessions command [20-14](#)
 - show snmp user command [20-14](#)
 - show tech-support command [26-2](#)
 - show vlan command [8-5](#)
 - SNMP [1-5](#)
 - accessing MIB variables with [20-4](#)
 - agent
 - described [20-4](#)
 - disabling [20-7](#)
 - community strings
 - configuring [20-7](#)

- overview 20-4
- configuration examples 20-13
- configuration guidelines 20-6
- default configuration 20-6
- groups 20-6, 20-8
- hosts 20-6
- informs
 - and trap keyword 20-10
 - described 20-5
 - differences from traps 20-5
 - enabling 20-12
- limiting access by TFTP servers 20-13
- manager functions 20-3
- notifications 20-5
- overview 20-1, 20-4
- status, displaying 20-14
- system contact and location 20-12
- trap manager, configuring 20-10
- traps
 - configuring 19-17
 - described 20-1, 20-5
 - differences from informs 20-5
 - enabling 20-10
 - ifIndex number, determining 19-19
 - overview 20-2, 20-4
 - types of 20-10
- users 20-6, 20-8
- versions supported 20-3
- snmp-server community command 20-7
- snmp-server contact command 20-12
- snmp-server enable traps command 20-11
- snmp-server engineID command 20-9
- snmp-server group command 20-9
- snmp-server host command 20-11
- snmp-server ifindex persist 19-17
- snmp-server location command 20-12
- snmp-server queue-length command 20-11
- snmp-server tftp-server-list command 20-13
- snmp-server trap-source command 20-11
- snmp-server trap-timeout command 20-11
- snmp-server user command 20-9
- SNMPv2C 20-3
- soft reset on ML-Series 3-2
- SONET alarms 5-6
- source 22-8
- sparse mode, PIM 11-34
- startup configuration file 3-9
- Startup configurationfile restoration 3-11
- static routes, configuring 11-31
- statistics
 - RMON group Ethernet 19-6
 - RMON group history 19-5
 - SNMP input and output 20-14
- statistics, OSPF 11-19, 11-32
- STP
 - BPDU message exchange 7-2
 - configuring
 - forward-delay time 7-20
 - hello time 7-19
 - path cost 7-18
 - port priority 7-17
 - root switch 7-17
 - switch priority 7-19
 - default configuration 7-16
 - designated port, defined 7-3
 - designated switch, defined 7-3
 - disabling 7-16
 - displaying status 7-20
 - extended system ID
 - overview 7-4
 - unexpected behavior 7-17
 - forward-delay time 7-6
 - inferior BPDU 7-3
 - interface states
 - blocking 7-6
 - disabled 7-7
 - forwarding 7-6, 7-7
 - learning 7-7

- listening [7-7](#)
- overview [7-5](#)
- Layer 2 protocol tunneling [9-9](#)
- limitations with IEEE 802.1Q trunks [7-8](#)
- multicast addresses, affect of [7-8](#)
- overview [7-2](#)
- redundant connectivity [7-8](#)
- root port, defined [7-3](#)
- root switch
 - effects of extended system ID [7-4](#)
 - election [7-3](#)
 - unexpected behavior [7-17](#)
- superior BPDU [7-3](#)
- supported number of spanning-tree instances [7-2, 7-9](#)
- timers, described [7-4](#)
- stub areas, OSPF [11-14](#)
- support, technical. *See* technical support
- SW-LCAS [5-3](#)
- syslog server [26-3](#)
- system MTU
 - IEEE 802.1Q tunneling [9-4](#)
 - maximums [9-4](#)

T

- tagged packets, Layer 2 protocol [9-9](#)
- TCAM
 - See also* SDM
 - entries [15-2](#)
 - Layer 3 switching information [15-1](#)
 - protocol regions [15-1](#)
 - space [15-1](#)
- technical support
 - FTP service [26-3](#)
 - gathering data [26-1](#)
 - logging router output [26-2](#)
 - providing data [26-3](#)
 - show tech-support command [26-2](#)
- terminals

- connecting to switch [3-5](#)
- terminal-emulation software [3-5](#)
- terminals, logging router output [26-2](#)
- ternary content addressable memory. *See* TCAM
- TFTP
 - limiting access by servers [20-13](#)
- traffic classes [14-10](#)
- traffic engineering
 - interface configuration to support RSVP-based tunnel signaling and IGP flooding [18-15](#)
 - MPLS traffic engineering tunnel, configuring [18-16](#)
 - OSPF for MPLS traffic engineering, configuring [18-16](#)
 - tasks [18-15](#)
 - tunnel support, configuring [18-15](#)
- traffic policies
 - creating [14-11](#)
 - interfaces, attaching [14-15](#)
- Transponder mode for G-Series [22-8](#)
- traps
 - configuring managers [20-10](#)
 - defined [20-3](#)
 - enabling [20-10](#)
 - notification types [20-10](#)
 - overview [20-2, 20-4](#)
- trunk ports [8-1](#)
- tunneling
 - defined [9-1](#)
 - IEEE 802.1Q [9-1](#)
 - Layer 2 protocol [9-10](#)
- tunnel ports
 - described [9-1](#)
 - IEEE 802.1Q, configuring [9-4, 9-11, 9-12](#)
 - incompatibilities with other features [9-4](#)

U

- user EXEC mode [3-17](#)

V

verifying

 IP multicast operation [11-35](#)

 VLAN operation [8-5](#)

version up software upgrade [3-14](#)

virtual LANs. *See* VLANs

VLANs

 aging dynamic addresses [7-9](#)

 configuring IEEE 802.1Q [8-2](#)

 customer numbering in service-provider
 networks [9-3](#)

 number per system [8-1](#)

 STP and IEEE 802.1Q trunks [7-8](#)

 trunk ports [8-1](#)

VLAN-specific services [9-6](#)

VRF Lite

 configuring [13-2](#)

 example [13-3](#)

 monitoring and verifying [13-7](#)

 understanding [13-1](#)

VTP Layer 2 protocol tunneling [9-10](#)

vty [3-4](#)