



Junos[®] OS

GMPLS Feature Guide

Release
11.2



Published: 2011-05-17

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

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

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

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

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

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

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

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

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

Junos® OS GMPLS Feature Guide

Release 11.2

Copyright © 2011, Juniper Networks, Inc.

All rights reserved.

Revision History

April 2011—R1 Junos OS 11.2

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

YEAR 2000 NOTICE

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

END USER LICENSE AGREEMENT

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Table of Contents

Part 1	GMPLS	
Chapter 1	GMPLS Concepts and Reference Materials	3
	GMPLS Overview	3
	GMPLS Operation	5
	GMPLS Phase 2 Implementation	6
	GMPLS System Requirements	7
	GMPLS Terms and Acronyms	8
Chapter 2	GMPLS Configuration	9
	Configuring Link Management Protocol Traffic Engineering Links	9
	Configuring Link Management Protocol Peers	10
	Configuring Peer Interfaces in OSPF and RSVP	11
	Establishing GMPLS LSP Path Information	11
	Defining GMPLS Label-Switched Paths	12
	Displaying Local Identifiers and Configuring Remote Identifiers	13
	Option: Tearing Down GMPLS LSPs Gracefully	13
	Option: Allowing Nonpacket GMPLS LSPs to Establish a Path Through Junos OS-based Routers	14
	Option: Selecting the Peer Model for GMPLS	14
	Option: Selecting the Overlay Model for GMPLS	15
	Option: GMPLS Graceful Restart	15
	Option: Configuring an LMP Control Channel	16
	Option: Configuring GMPLS Support for Unnumbered Links	17
Chapter 3	GMPLS Configuration Examples	19
	Example: GMPLS Configuration	19
	Verifying Your Work	24
	Router A Status	24
	Router C Status	29
	Example: Configuring Traffic Engineering Link and Interface Identifiers	30
	Example: LMP Control Channel Configuration	31
	Verifying Your Work	37
	Router 1 Status	37
	Router 4 Status	38
	For More Information	39
Part 2	Index	
	Index	43

List of Figures

Part 1	GMPLS	
Chapter 1	GMPLS Concepts and Reference Materials	3
	Figure 1: GMPLS LSP Hierarchy	4
Chapter 3	GMPLS Configuration Examples	19
	Figure 2: GMPLS Topology Diagram	19
	Figure 3: Traffic Engineering Link and Interface ID Example	30
	Figure 4: LMP Control Channel Topology Diagram	31

List of Tables

Part 1

GMPLS

Chapter 2

GMPLS Configuration 9

Table 1: Default Values for LMP Protocol Fields 16

PART 1

GMPLS

- GMPLS Concepts and Reference Materials on page 3
- GMPLS Configuration on page 9
- GMPLS Configuration Examples on page 19

CHAPTER 1

GMPLS Concepts and Reference Materials

This section contains the following topics:

- GMPLS Overview on page 3
- GMPLS Operation on page 5
- GMPLS Phase 2 Implementation on page 6
- GMPLS System Requirements on page 7
- GMPLS Terms and Acronyms on page 8

GMPLS Overview

Generalized Multiprotocol Label Switching (GMPLS) is the next-generation implementation of Multiprotocol Label Switching (MPLS). GMPLS extends the functionality of MPLS to include a wider range of label-switched path (LSP) options for a variety of network devices.

You should have a general understanding of MPLS, label switching concepts, and GMPLS Phase 1. For more information about MPLS, see the *Junos MPLS Applications Configuration Guide*. For more information about GMPLS Phase 1, see the *Junos 5.5 Feature Guide* at: <http://www.juniper.net/techpubs/software/junos/junos55/feature-guide55/feature-guide-55.pdf>.

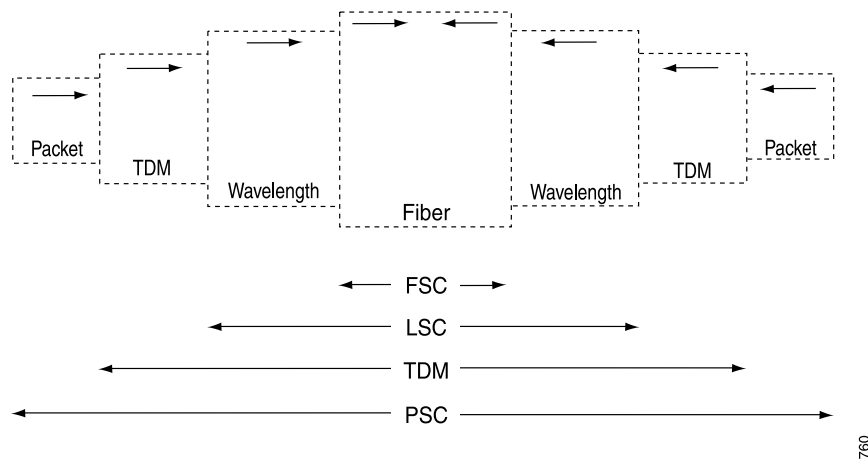
Traditional MPLS is designed to carry Layer 3 IP traffic by establishing IP-based paths and associating these paths with arbitrarily assigned labels. These labels can either be configured explicitly by a network administrator or dynamically assigned by a protocol such as the Label Distribution Protocol (LDP) or Resource Reservation Protocol (RSVP).

In contrast, GMPLS can carry various types of Layer 1 through Layer 3 traffic. GMPLS labels and LSPs can be processed at four levels, as depicted in Figure 1 on page 4. The levels are Fiber-Switched Capable (FSC), Lambda-Switched Capable (LSC), Time-Division Multiplexing Capable (TDM), and Packet-Switched Capable (PSC).

LSPs must start and end on links with the same switching capability. To send an LSP, a label-switched device must communicate with another device at the same layer of the Open System Interconnection (OSI) model. Thus, routers can set up PSC LSPs with other routers at Layer 3, and SONET/SDH add/drop multiplexers (ADMs) can establish TDM LSPs with other ADMs at Layer 1. As seen in Figure 1 on page 4, a router PSC LSP can be

carried over a TDM LSP, a TDM LSP can be carried over a wavelength LSC LSP, and so on.

Figure 1: GMPLS LSP Hierarchy



This extension of the MPLS protocol expands the number of devices that can participate in label switching. Lower layer devices, such as optical cross-connects (OXC)s and SONET/SDH ADMs, can now participate in GMPLS signaling and set up paths to transfer data. Additionally, routers can participate in signaling optical paths across a transport network.

GMPLS labeling is also more flexible than MPLS. A GMPLS label can represent a TDM time slot, a Dense Wavelength Division Multiplexing (DWDM) wavelength (also known as a lambda), or a physical port number. The labels can be derived from physical components of the network devices, such as interfaces.

There are two service models for GMPLS. Each model determines how much visibility a client node, such as a router, has into the optical core or transport network. The first model is a user-to-network interface (UNI), which is often referred to as the overlay model. The second is known as the peer model. Juniper Networks supports both models.

To enable multilayer LSPs, GMPLS uses the following mechanisms:

- Separation of the control channel from the data channel—A new protocol called Link Management Protocol (LMP) is used to define and manage both control channels and data channels between GMPLS peers. Messages for GMPLS LSP setup are sent from one device to a peer device over an out-of-band control channel. Once the LSP setup is complete and the path is provisioned, the data channel is established and can be used to carry traffic. In GMPLS, the control channel is always separate from the data channel.
- RSVP-TE extensions for GMPLS—RSVP-TE was designed to signal the setup of packet LSPs only. The protocol has been extended to request path setup for nonpacket LSPs that use wavelengths, time slots, and fibers as potential labels.

- OSPF extensions for GMPLS—OSPF was designed to route packets to physical and logical interfaces related to a PIC. This protocol has been extended to route packets to virtual peer interfaces defined in an LMP configuration.
- Bidirectional LSPs—Unlike unidirectional LSP paths found in the standard, packet-based version of MPLS, data can travel both ways between GMPLS devices over a single LSP path. Nonpacket LSPs in GMPLS are bidirectional by default.

GMPLS is intended to bridge the gap between the traditional transport infrastructure and the IP layer. Since this protocol is supported by several network industry organizations and standards bodies, GMPLS is designed to enable multivendor interoperability and multilayer functionality. In the near future, routers will be able to make dynamic requests for extra bandwidth on demand from the optical network. Consequently, service providers envision GMPLS as a means to set up optical circuits and services dynamically instead of manually. Many industry professionals are cautiously optimistic regarding the advent of GMPLS, and Juniper Networks is pleased to continue its support for this protocol.

GMPLS Operation

GMPLS requires close interaction between LMP, RSVP, and OSPF. The following sequence of events describes how GMPLS works:

1. LMP notifies RSVP and OSPF of the control peer, the control adjacency, and resources for the traffic engineering link.
2. GMPLS extracts the LSP attributes from the configuration and requests that RSVP signal one or more specific paths, specified by the traffic engineering link addresses.
3. RSVP determines the local traffic engineering link, corresponding control adjacency and active control channel, and transmission parameters (such as IP destination). It requests that LMP allocate a resource from the traffic engineering link with the specified attributes. If LMP successfully finds a resource matching the attributes, label allocation succeeds. RSVP sends a **PathMsg** hop by hop until it reaches the target router.
4. The target router, on receiving the RSVP **PathMsg**, requests that LMP allocate a resource based on the signaled parameters. If label allocation succeeds, it sends back a **ResvMsg**.
5. If the signaling is successful, an optical path is provisioned.

GMPLS Phase 2 Implementation

The major changes between GMPLS Phase 1 and GMPLS Phase 2 are as follows:

- You must configure one or more control channels between peers when you configure LMP (in addition to the existing statements for LMP peers and traffic engineering links). The control channels must travel across a point-to-point link or tunnel. To configure a static control channel, include the **control-channel** statement at the **[edit protocols link-management peer *peer-name*]** hierarchy level. To configure a control channel that uses control channel management and link property correlation, include the **lmp-control-channel** statement at the **[edit protocols link-management peer *peer-name*]** hierarchy level.



NOTE: You can configure either the **control-channel** statement or the **lmp-control-channel** statement at the **[edit protocols link-management peer *peer-name*]** hierarchy level, but not both statements simultaneously.

- OSPF and RSVP have been extended to allow control adjacencies between peers using virtual peer interfaces. The peer interfaces are derived from LMP and can be used for the control adjacency between peers instead of the physical interfaces. To configure OSPF, include the **peer-interface** statement at the **[edit protocols ospf area *area-number*]** hierarchy level. To configure RSVP, include the **peer-interface** statement at the **[edit protocols rsvp]** hierarchy level. However, when you enable peer interfaces, you must disable RSVP and OSPF on all physical control channel interfaces. Alternatively, you can omit the physical control channel interfaces when configuring these protocols.
- The Constrained Shortest Path First (CSPF) algorithm has been extended to permit use with nonpacket LSPs. In GMPLS Phase 2, the **no-cspf** statement can be omitted from the LSP configuration because it is no longer mandatory. When this statement is omitted, you must configure the signal type attribute for the LSP. For CSPF to work correctly, OSPF extensions for GMPLS need to be implemented on all devices in the GMPLS network.
- LSP paths now can be strict, loose, or dynamic for GMPLS LSPs because traffic engineering link information is now exchanged by OSPF. (GMPLS Phase 1 required strict LSP paths.)

Junos OS supports the following GMPLS functionality:

- Out-of-band signaling controls the setup of LSP paths, enabling a control plane that is separate from the data plane.
- RSVP-TE extensions support additional objects beyond Layer 3 packets, such as ports, time slots, and wavelengths.
- Link Management Protocol (LMP) creates and maintains a database of traffic engineering links, control channels, and peer information. Only the static version of this protocol is supported.
- Bidirectional LSPs are required between nonpacket GMPLS devices.

- Several GMPLS label types are defined in RFC 3471, *Generalized MPLS - Signaling Functional Description*. The MPLS, Generalized, SONET/SDH, Suggested, and Upstream label types are supported.
- Generalized labels do not contain a type field because the nodes are expected to know from the context of their connection what type of label to expect. For example, an encoding type, such as Ethernet or SONET/SDH, is determined by the resources in a traffic engineering link.
- Traffic parameters facilitate GMPLS bandwidth encoding and SONET/SDH formatting.
- Interface Identification/Errored Interface Identification, UNI-style signaling, and Secondary LSP paths are supported.
- Original channelized interfaces (such as channelized OC12 to DS3, channelized OC3 to T1, and channelized STM1 to E1) support GMPLS signaling.
- GMPLS graceful restart for RSVP LSP paths is supported.
- RSVP-TE over unnumbered links is supported.

The following functionality is *not* supported:

- Notify messages
- GMPLS routing extensions for IS-IS
- GMPLS link bundling
- Dynamic LMP



NOTE: There is not necessarily a one-to-one correspondence between a physical interface and a GMPLS interface. If a GMPLS connection uses a nonchannelized physical connector, the GMPLS label can use the physical port ID. However, the label for channelized interfaces often is based on a channel or time slot. Consequently, it is best not to refer to GMPLS labels as “interfaces.” To avoid confusion, refer to them as traffic engineering links and refer to the physical interfaces as resources.

GMPLS System Requirements

To implement GMPLS Phase 2, your system must meet these minimum requirements:

- Junos OS Release 8.5 or later for GMPLS support of unnumbered links.
- Junos OS Release 8.1 or later for LMP control channels
- Junos OS Release 7.0 or later for graceful teardown of GMPLS LSPs and graceful restart of GMPLS neighbors
- Junos OS Release 5.6 or later for GMPLS Phase 2
- Two Juniper Networks M Series or T Series routers, and two optical cross-connects (OXCs) that support GMPLS

GMPLS Terms and Acronyms

C

control adjacency	A signaling path between peer devices in a GMPLS network that typically travels across virtual peer interfaces. Protocols are enabled on the control adjacency, which can have one or more associated control channels.
control channel	The actual interfaces where protocol packets are sent and received by GMPLS peers. If more than one control channel is configured, LMP selects which control channel is active.

F

Fiber-Switched Capable (FSC)	LSPs are switched between two fiber-based devices, such as optical cross-connects (OXC), that operate at the level of individual fibers.
forwarding adjacency	A forwarding path for sending data between peer devices in a GMPLS network.

G

Generalized Multiprotocol Label Switching (GMPLS)	An extension to MPLS that allows data from multiple layers to be switched over label-switched paths (LSPs). GMPLS LSPs are possible between equivalent Layer 1, Layer 2, and Layer 3 devices. For more information about GMPLS and MPLS, see the <i>Junos MPLS Applications Configuration Guide</i> .
GMPLS label	A fiber port, TDM time slot, DWDM wavelength, or data packet identifier of a GMPLS-enabled device used as a next-hop identifier.

L

Lambda-Switched Capable (LSC)	LSPs are switched between two DWDM devices, such as such as OXC, that operate at the level of individual wavelengths.
Link Management Protocol (LMP)	A GMPLS-related protocol defined in RFC 4204 that is used to define control adjacencies and forwarding adjacencies between peers and to maintain and allocate resources on traffic engineering links.

P

Packet-Switched Capable (PSC)	LSPs are switched between two packet-based devices, such as routers or ATM switches.
--------------------------------------	--

T

TDM-Switched Capable (TDM)	LSPs are switched between two TDM devices, such as SONET/SDH ADMs.
traffic engineering link	A logical connection between GMPLS-enabled devices. traffic engineering links can have addresses or IDs and are associated with certain resources or interfaces. They also have certain inherent attributes, such as encoding type, switching capability, and bandwidth. Each traffic engineering link represents a forwarding adjacency between a pair of devices.

CHAPTER 2

GMPLS Configuration

To implement GMPLS, you must configure traffic link management protocol traffic engineering links and protocol peers and OSPF and RSVP peer interfaces, establish GMPLS LSP path information, define GMPLS paths, discover local identifiers and configure remote identifiers. This section contains these configuration procedures plus some optional configuration procedures:

- Configuring Link Management Protocol Traffic Engineering Links on page 9
- Configuring Link Management Protocol Peers on page 10
- Configuring Peer Interfaces in OSPF and RSVP on page 11
- Establishing GMPLS LSP Path Information on page 11
- Defining GMPLS Label-Switched Paths on page 12
- Displaying Local Identifiers and Configuring Remote Identifiers on page 13
- Option: Tearing Down GMPLS LSPs Gracefully on page 13
- Option: Allowing Nonpacket GMPLS LSPs to Establish a Path Through Junos OS-based Routers on page 14
- Option: Selecting the Peer Model for GMPLS on page 14
- Option: Selecting the Overlay Model for GMPLS on page 15
- Option: GMPLS Graceful Restart on page 15
- Option: Configuring an LMP Control Channel on page 16
- Option: Configuring GMPLS Support for Unnumbered Links on page 17

Configuring Link Management Protocol Traffic Engineering Links

To begin your GMPLS configuration, enable LMP to define the data channel interconnection between devices at the **[edit protocols link-management]** hierarchy level.

To configure data channels in LMP, include the **te-link *te-link-name*** statement at the **[edit protocols link-management]** hierarchy level. Define all traffic engineering link options shown. (You will configure **remote-id** statements at the **te-link** and **interface** levels later.) We recommend that you use a different IP address and mask on your traffic engineering link addresses from the ones configured on your physical interfaces. This way, you can identify which addresses are physical and which ones belong to the traffic engineering link (TE link).

```
[edit]
protocols {
  link-management {
    te-link te-link-name { # Collection of physical ports or time slots.
      local-address ip-address; # Local IP address associated with the TE link.
      remote-address ip-address; # Remote IP address mapped to the TE link.
      interface interface-name { # Interface used for data transfer.
        local-address ip-address; # Local IP address for the TE link.
        remote-address ip-address; # Remote IP address for the TE link.
      }
    }
  }
}
```

Configuring Link Management Protocol Peers

After you set up traffic engineering links, configure LMP network peers with the **peer** statement at the **[edit protocols link-management]** hierarchy level. A peer is the network device that your router communicates with when setting up the control and data channels. Often, the peer is an OXC. Designate a peer name, configure the peer's router ID as the address (often a loopback address), specify the interface that will be used as a control channel, and apply the traffic engineering link to be associated with this peer.

You can configure one or more control channels for a peer. The control channels must have point-to-point connectivity with the peer (for example, you can use a point-to-point link or a tunnel). You can also configure the generic routing encapsulation (GRE) tunnel interface of the Routing Engine as a control channel. To configure a static control channel, include the **control-channel** statement at the **[edit protocols link-management peer peer-name]** hierarchy level. To configure a control channel that uses control channel management and link property correlation, see "Option: Configuring an LMP Control Channel" on page 16. Without a control channel, the commit operation fails.

```
[edit]
protocols {
  link-management {
    peer peer-name { # Configure the name of your network peer.
      address ip-address; # Include the router ID of the peer.
      control-channel interface; # Specify the interface for the control channel.
      te-link te-link-name; # Assign a TE link to this peer.
    }
  }
}
```



NOTE: Although you can configure the **gre-** tunnel interface on a Routing Engine as a control channel, this interface is not supported, nor is it configurable for other applications.



NOTE: You can configure either the **control-channel** statement or the **lmp-control-channel** statement at the **[edit protocols link-management peer peer-name]** hierarchy level, but not both statements simultaneously.

Configuring Peer Interfaces in OSPF and RSVP

After you establish LMP peers, add peer interfaces to OSPF and RSVP. A peer interface is a virtual interface used to support a control adjacency between two peers. OSPF and RSVP form adjacencies between peers by using the peer interfaces instead of the physical interfaces.

Because actual protocol packets are sent and received by peer interfaces, the peer interfaces can be signaled and advertised to peers like any other interface enabled for RSVP and OSPF. The peer interface name must match the peer name configured in LMP. To configure RSVP signaling for LMP peers, include the **peer-interface** statement at the **[edit protocols rsvp]** hierarchy level. To configure OSPF routing for LMP peers, include the **peer-interface** statement at the **[edit protocols ospf area area-number]** hierarchy level.

```
[edit]
protocols {
  rsvp {
    peer-interface peer-name { # Configure the name of your LMP peer.
    }
  }
  ospf {
    area area-number {
      peer-interface peer-name { # Configure the name of your LMP peer.
      }
    }
  }
}
```



NOTE: When adding the virtual peer interfaces to RSVP and OSPF, do not configure the corresponding physical control channel interface in either protocol. If the interface all option is used, you must disable the protocols manually on the control channel interface.

- To disable OSPF, use the **disable** statement at the **[edit protocols ospf area area-number interface interface-name]** hierarchy level.
- To disable RSVP, use the **disable** statement at the **[edit protocols rsvp interface interface-name]** hierarchy level.

Establishing GMPLS LSP Path Information

When you configure LSP paths for GMPLS, you must use the traffic engineering link remote address as your next-hop address. When CSPF is supported, you can use any path option you wish. However, when CSPF is disabled with the **no-cspf** statement at the **[edit protocols mpls label-switched-path lsp-name]** hierarchy level, you must use strict paths.

```
[edit]
protocols {
  mpls {
```

```

    path path-name {
        next-hop-address (strict | loose);
    }
}

```

Defining GMPLS Label-Switched Paths

Next, define LSP attributes at the **[edit protocols mpls label-switched-path]** hierarchy level. To enable the proper GMPLS switching parameters, configure the attributes appropriate for your network connection. The default values, which are also appropriate for standard MPLS, are **ipv4** for **gp-id**, **none** for **signal-bandwidth**, and **psc-1** for **switching-type**.



NOTE: In Junos OS Release 5.6 and later, the **signal-bandwidth** statement replaces the **signal-type** statement. Also, virtual tributary (VT) 1.5 and 2.0 SONET/SDH bandwidth options are available at the **[edit protocols mpls label-switched-path lsp-name lsp-attributes signal-bandwidth]** hierarchy level.

```

[edit]
protocols {
  mpls {
    label-switched-path lsp-name {
      from ip-address;
      to ip-address;
      primary path-name;
      secondary path-name;
      no-cspf; # This statement to disable CSPF is optional.
      lsp-attributes { # Attributes determine the selection of an LSP.
        gp-id type; # Payload type, such as Ethernet or PPP.
        signal-bandwidth type; # Bandwidth encoding, such as DS3 or STM1.
        switching-type type; # Switching method, such as psc-1 or lambda.
      }
    }
  }
}

```



NOTE: Because MPLS and GMPLS use the same configuration hierarchy for LSPs, it is helpful to know which LSP attributes control LSP functionality. Standard MPLS packet-switched LSPs are unidirectional. GMPLS nonpacket LSPs are bidirectional.

If you use the default packet switching type of **psc-1**, your LSP becomes unidirectional. To enable a GMPLS bidirectional LSP, you must select a nonpacket switching type option, such as **lambda** or **fiber**, at the **[edit mpls label-switched-path *lsp-name* lsp-attributes]** hierarchy level.

Displaying Local Identifiers and Configuring Remote Identifiers

Once LMP is enabled on a router, the router automatically assigns two local IDs: one at the **te-link** level, the other at the **interface** level. You must configure these port-to-label mappings manually for LMP on both the router and its peer. To configure the mapping, set the local IDs of one device (such as the router) as remote IDs on the peer device (such as an OXC) with the **remote-id** statement at the **[edit protocols link-management te-link *te-link-name*]** and **[edit protocols link-management te-link *te-link-name* interface *interface-name*]** hierarchy levels.

You can view the traffic engineering link and interface local IDs by using the **show link-management te-link** command. Once you have learned these IDs, configure them as **remote-id** statements at the corresponding **te-link** and **interface** levels on the peer.

Because peers vary, check with your OXC vendor for the configuration statements and location of the local ID information for your specific optical peer device. If you do not manage the peer device, ask the peer's administrator to enable LMP and generate the IDs for you. GMPLS will not work unless these local IDs from both the router and the peer are configured as remote IDs on the opposite device.

To disable an entire traffic engineering link for administrative purposes, include the **disable** statement at the **[edit protocols link-management te-link *te-link-name*]** hierarchy level. To disable an interface within a traffic engineering link, include the **disable** statement at the **[edit protocols link-management te-link *te-link-name* interface *interface-name*]** hierarchy level.

```
[edit]
protocols {
  link-management {
    te-link te-link-name {
      disable; # Disable the entire TE link.
      remote-id id-number; # TE link ID number of the peer device.
      interface interface-name { # Name of the interface used for data transfer.
        disable; # Disable an interface in the TE link.
        remote-id id-number; # ID number of the remote device.
      }
    }
  }
}
```

Option: Tearing Down GMPLS LSPs Gracefully

You can tear down a nonpacket GMPLS LSP in a two-step process that gracefully withdraws the RSVP session used by the LSP. For all neighbors that support graceful teardown, a request for the teardown is sent by the routing platform to the destination endpoint for the LSP and all RSVP neighbors in the path. The request is included within the **ADMIN_STATUS** field of the RSVP packet. When neighbors receive the request, they prepare for the RSVP session to be withdrawn. A second message is sent by the routing platform to complete the teardown of the RSVP session. If a neighbor does not support graceful teardown, the request is handled as a standard session teardown rather than a graceful one.

To perform a graceful teardown of a GMPLS LSP RSVP session, issue the **clear rsvp session gracefully** command. Optionally, you can specify the source and destination address of the RSVP session, the LSP identifier of the RSVP sender, and the tunnel identifier of the RSVP session. To use these qualifiers, include the **connection-source**, **connection-destination**, **lsp-id**, and **tunnel-id** options when you issue the **clear rsvp session gracefully** command.

You can also configure the amount of time that the routing platform waits for neighbors to receive the graceful teardown request before initiating the actual teardown. To configure the timeout, include the **graceful-deletion-timeout** statement at the **[edit protocols rsvp]** hierarchy level. The default graceful deletion timeout value is 30 seconds, with a minimum value of 1 second and a maximum value of 300 seconds. To view the current value configured for graceful deletion timeout, issue the **show rsvp version** operational command.

Option: Allowing Nonpacket GMPLS LSPs to Establish a Path Through Junos OS-based Routers

When you configure a nonpacket LSP as administratively down, an external device (not a router running Junos OS) can either perform a Layer 1 path setup test or help bring up an optical cross-connect through a router running Junos OS.

To configure a nonpacket LSP as administratively down, you must set the A-bit in the ADMIN_STATUS field of a RSVP packet. This feature does not affect control path setup or data forwarding for packet LSPs.

To configure the ADMIN_STATUS field for a GMPLS LSP RSVP packet, issue the **admin-down** statement at the **[edit protocols mpls label-switched-path *lsp-name*]** hierarchy level or the **[edit logical-systems *logical-system-name* protocols mpls label-switched-path *lsp-name*]** hierarchy level.

Option: Selecting the Peer Model for GMPLS

To implement the peer model, perform the following configuration tasks:

- Use the default CSPF calculation that is inherent in MPLS LSPs.
- Include a traffic engineering link and a control channel within the **peer** statement at the **[edit protocols link-management]** hierarchy level.
- Reference the **peer-interface** in both OSPF and RSVP.
- Configure OSPF and OSPF traffic engineering to connect with the OXC.
- Optionally, define a GMPLS LSP path.

Option: Selecting the Overlay Model for GMPLS

To implement the overlay model, perform the following configuration tasks:

- Disable CSPF by including the **no-cspf** statement at the **[edit protocols mpls label-switched-path *lsp-name*]** hierarchy level.
- Define a strict GMPLS LSP data path to the remote OXC peer.
- Include a traffic engineering link and a control channel within the **peer** statement at the **[edit protocols link-management]** hierarchy level.
- Reference the **peer-interface** in RSVP.

Option: GMPLS Graceful Restart

GMPLS supports graceful restart, a mechanism that allows a restarting router to continue forwarding packets to neighbors without interruption. The restarting router relies on its forwarding table temporarily while it receives updates from “helper” neighbors that assist the restarting router in rebuilding its routing table.

To enable graceful restart for all routing protocols including GMPLS, include the **graceful-restart** statement at the **[edit routing-options]** hierarchy level. To disable graceful restart for GMPLS and RSVP, include the **disable** statement at the **[edit protocols rsvp graceful-restart]** hierarchy level. To disable GMPLS and RSVP graceful restart helper capability, include the **helper-disable** statement at the **[edit protocols rsvp graceful-restart]** hierarchy level.

To configure the maximum amount of time the routing platform retains information for restarting neighbors, include the **maximum-helper-recovery-time** statement at the **[edit protocols rsvp graceful-restart]** hierarchy level. The default helper recovery time is 180 seconds, with a minimum value of 1 second and a maximum value of 3600 seconds. To view the current value configured for the helper recovery time, issue the **show rsvp version** operational mode command.

To configure the maximum amount of time the routing platform waits until a neighbor is declared dead, include the **maximum-helper-restart-time** statement at the **[edit protocols rsvp graceful-restart]** hierarchy level. The default helper restart time is 20 seconds, with a minimum value of 1 second and a maximum value of 1800 seconds. To view the current value configured for the helper restart time, issue the **show rsvp version** operational command.

```
[edit]
protocols {
  rsvp {
    graceful-restart {
      disable;
      helper-disable;
      maximum-helper-recovery-time seconds;
      maximum-helper-restart-time seconds;
    }
  }
}
```

```
}
}
```

For more information about graceful restart, see the *Junos OS High Availability Configuration Guide*.

Option: Configuring an LMP Control Channel

In contrast with statically configured control channels that provide basic connectivity, LMP control channels provide control channel management and link property correlation as defined in RFC 4204, *Link Management Protocol (LMP)*. To manage the state of the control channel, LMP uses the following sequence of events:

1. The LMP peer with the highest router ID sends configuration messages to peers.
2. When a peer sends an acknowledgement of a **Config** message back to the originator, the control channel enters the **active** state.
3. When a peer sends LMP hello messages and receives them from a neighbor, the control channel transitions to the **up** state.
4. Once the control channel is up, link summary messages and acknowledgements are sent between LMP peers to share traffic engineering link information.

To configure an LMP control channel, include the **lmp-control-channel** statement at the **[edit protocols link-management peer *peer-name*]** hierarchy level and specify the physical IP address of the peer as the remote address. You can also specify a hello interval, a hello dead interval, a retransmission interval, and a retry limit. Default values for these statements are shown in Table 1 on page 16.

Table 1: Default Values for LMP Protocol Fields

LMP Protocol Field	Value
Hello interval	150 milliseconds
Hello dead interval	500 milliseconds
Retransmission interval	500 milliseconds
Retry limit	3 retries

If you plan to use these default values, you do not need to configure them. However, if you choose to manually configure any of these values, you must include all four statements (**hello-interval**, **hello-dead-interval**, **retransmission-interval**, and **retry-limit**) at the **[edit protocols link-management peer *peer-name* lmp-protocol]** hierarchy level. Also, the hello dead interval must be at least three times larger than the hello interval.

If you do not want the routing platform to initiate LMP negotiations, include the **passive** statement at the **[edit protocols link-management peer *peer-name* lmp-protocol]** hierarchy level. To log hello packets, other LMP messages, and state transitions of the control

channels and traffic engineering links, include the **hello-packets**, **packets**, and **state** traceoptions flags at the **[edit protocols link-management traceoptions]** hierarchy level.

```
[edit]
protocols {
  link-management {
    peer peer-name { # Configure the name of your network peer.
      address ip-address; # Include the router ID of the peer.
      lmp-control-channel interface { # Specify the control channel interface.
        remote-address ip-address; # Configure the peer's physical IP address.
      }
      lmp-protocol { # Manually configure LMP protocol values.
        hello-dead-interval milliseconds;
        hello-interval milliseconds;
        passive; # Respond to LMP peers, but do not initiate LMP negotiations.
        retransmission-interval milliseconds;
        retry-limit number;
      }
      te-link te-link-name; # Assign a TE link to this peer.
    }
    traceoptions {
      flag (hello-packets | packets | state);
    }
  }
}
```

For a full example of an LMP protocol control channel configuration, see “Example: LMP Control Channel Configuration” on page 31.



NOTE: You can configure either the **control-channel** statement or the **lmp-control-channel** statement at the **[edit protocols link-management peer *peer-name*]** hierarchy level, but not both statements simultaneously.

Option: Configuring GMPLS Support for Unnumbered Links

The Junos OS supports RSVP traffic engineering over unnumbered interfaces. With this feature, you do not have to configure IP addresses for each interface participating in the RSVP-signaled network. Traffic engineering information about unnumbered links is carried in the IGP traffic engineering extensions for OSPF and IS-IS, as described in RFC 4203, *OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)*. Unnumbered links can also be specified in the MPLS traffic engineering signaling, as described in RFC 3477, *Signaling Unnumbered Links in Resource ReSerVation Protocol - Traffic Engineering (RSVP-TE)*.

To configure RSVP for unnumbered interfaces, you must configure a router ID using the **router-id** statement specified at the **[edit routing-options]** hierarchy level. The router ID must be routable (you can typically use the loopback address). The RSVP control messages for the unnumbered links are sent using the router ID address (rather than a randomly selected address). To configure link protection and fast reroute on a router with unnumbered interfaces enabled, you must configure at least two addresses. In

addition to the router ID, we recommend that you configure a secondary interface on the loopback:

```
[edit]
routing-options {
  router-id address;
}
```

CHAPTER 3

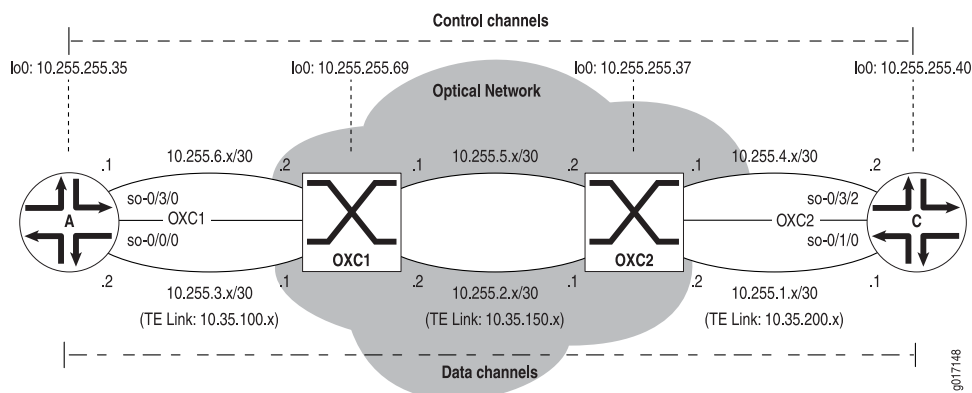
GMPLS Configuration Examples

This section contains configuration examples and commands you can issue to verify a GMPLS configuration:

- Example: GMPLS Configuration on page 19
- Example: Configuring Traffic Engineering Link and Interface Identifiers on page 30
- Example: LMP Control Channel Configuration on page 31
- For More Information on page 39

Example: GMPLS Configuration

Figure 2: GMPLS Topology Diagram



In Figure 2 on page 19, a control channel is established between Router A and OXC1, OXC1 and OXC2, and OXC2 and Router C. A data channel is enabled on a second connection between each pair of devices. The optical network cloud can contain OXCs, ADMs, or other lower-layer devices. In this example, OXC1 and OXC2 are in the direct data path between Routers A and C and the two OXCs have point-to-point connectivity with each other and the directly connected peer routers.

Starting with Router A, configure LMP traffic engineering links and peers to create a data channel and a control channel to connect with OXC1. To differentiate the logical traffic engineering link from the physical network, the local and remote addresses in the traffic engineering link are not related to the IP addresses assigned to the physical interfaces.

When you enable LMP peering on both Router A and OXC1, include the control channel interface as one of the peer statements. Use the name of the peer (in this case, **oxc1**) as the peer interface name when you add the **peer-interface** statement to RSVP at the **[edit protocols rsvp]** hierarchy level and OSPF at the **[edit protocols ospf area area-number]** hierarchy level.

The **peer-interface** statement adds the remote address and local address from your LMP configuration into the routing and signaling processes activated between Router A and OXC1. Make sure the physical control channel is a point-to-point link and has some form of IP reachability through static routes, an interior gateway protocol (IGP), or BGP (this example uses OSPF). Another way to achieve point-to-point links, especially if there are multiple hops between peers, is to use a generic routing encapsulation (GRE) tunnel for the control channel.

Next, configure an MPLS LSP on Router A to reach Router C. For this example, assume your data plane connection uses STM1 and Point-to-Point Protocol (PPP) over a fiber-switched network. Configure these LSP attributes in the LSP. Because this LSP does not use packet switching, a bidirectional LSP is enabled by default. As a result, you do not need to configure a return path LSP on Router C.

Finally, remember to discover the local IDs and configure them on OXC1 with the **remote-id** statement at the **[edit protocols link-management te-link te-link-name]** and **[edit protocols link-management te-link te-link-name interface]** hierarchy levels. For Router A, use the command **show link-management te-link** to find Router A's two local IDs (**te-link** and **interface**); then configure these IDs as remote IDs on OXC1 at the equivalent hierarchy levels.

```
Router A [edit]
interfaces {
  so-0/0/0 {
    description "Data channel to OXC1";
    encapsulation ppp;
    unit 0 {
      family inet {
        address 10.255.3.2/30 {
          destination 10.255.3.1;
        }
      }
      family mpls;
    }
  }
  so-0/3/0 {
    description "Control channel to OXC1";
    encapsulation ppp;
    unit 0 {
      family inet {
        address 10.255.6.1/30 {
          destination 10.255.6.2;
        }
      }
      family mpls;
    }
  }
}
```



```

lo0 {
  unit 0 {
    family inet {
      address 10.255.255.35/32;
    }
  }
}
protocols
  rsvp {
    interface all;
    interface so-0/3/0.0 {
      disable;
    }
    peer-interface oxc1;
  }
  mpls {
    label-switched-path gmpls-lsp1 {
      to 10.255.255.40;
      lsp-attributes {
        signal-bandwidth stm-1;
        switching-type fiber;
        gpip ppp;
      }
      primary path-lsp1;
    }
    path path-lsp1 {
      10.35.100.1 strict; # This example does not disable CSPF,
      10.35.150.1 strict; # so this step is optional.
      10.35.200.1 strict;
    }
    interface all;
  }
  ospf {
    area 0.0.0.0 {
      interface lo0.0;
      interface fxp0.0 {
        disable;
      }
      peer-interface oxc1;
    }
  }
  link-management {
    te-link te-oxc1 {
      local-address 10.35.100.2;
      remote-address 10.35.100.1;
      remote-id 8256;
      interface t3-3/3/0:0 {
        local-address 10.35.100.2;
        remote-address 10.35.100.1;
        remote-id 65536;
      }
    }
  }
  peer oxc1 {
    address 10.255.255.69;
    control-channel so-0/3/0.0;
  }
}

```

```

    te-link te-oxc1;
  }
}

```

On OXC1, complete your configuration of the control channel and the traffic engineering link data channel to Router A. Refer to your OXC vendor's instructions to configure a traffic engineering link on your specific device. Enable LMP peering, configure Router A's local IDs as remote IDs on OXC1, and discover OXC1's local IDs. Finally, configure OXC1's local IDs as remote IDs on Router A.

In the optical network between your OXCs, configure a traffic engineering link and a control channel between OXC1 and OXC2. Refer to the OXC vendor's instructions to configure this link. For the example shown in Figure 2 on page 19, you can assume a traffic engineering link with an address space of **10.255.150.x/30** has been enabled over a physical network with IP addresses **10.255.2.x/30**. Also, a control channel has been created over the **10.255.4.x/30** link.

On OXC2, configure a traffic engineering link to Router A. Refer to your OXC vendor's instructions to configure this traffic engineering link on your device. Enable LMP peering, configure Router C's local IDs as remote IDs on OXC2, and discover OXC2's local IDs. Finally, configure OXC2's local IDs as remote IDs on Router C.

Now you are ready to complete this GMPLS example. On Router C, set up your traffic engineering link, LMP peer, and control channel statements to connect to OXC2. As with Router A, the local and remote addresses in the traffic engineering link on Router C are not related to the IP addresses assigned to the physical interface.

Next, configure RSVP, MPLS, and OSPF to match the control channel protocols you configured on Router A. You do not need to set up an LSP on Router C because Router A's nonpacket LSP is bidirectional by default. Also, because RSVP is enabled for all interfaces and you are using a peer interface, you must disable RSVP on the physical control channel interface **so-0/3/2**.

After you enable LMP on both Router C and OXC2, discover the local IDs and configure them as remote IDs on OXC2. For Router C, use the command **show link-management te-link** to discover Router C's two local IDs (**te-link** and **interface**); then configure these IDs as remote IDs on OXC2 at the equivalent hierarchy levels.

```

Router C [edit]
interfaces {
  so-0/3/2 {
    description "Control channel to OXC2";
    unit 0 {
      family inet {
        address 10.255.4.2/30 {
          destination 10.255.4.1;
        }
      }
      family mpls;
    }
  }
  so-0/1/0 {
    description "Data channel to OXC2";

```

```

encapsulation ppp;
unit 0 {
    family inet {
        address 10.255.1.1/30 {
            destination 10.255.1.2;
        }
    }
    family mpls;
}
}
lo0 {
    unit 0 {
        family inet {
            address 10.255.255.40/32;
        }
    }
}
}
protocols
rsvp {
    interface all;
    interface so-0/3/2.0 {
        disabled;
    }
    peer-interface oxc2;
}
mpls {
    interface all;
}
ospf {
    area 0.0.0.0 {
        interface fxp0.0 {
            disable;
        }
        interface lo0.0;
        peer-interface oxc2;
    }
}
link-management {
    te-link te-oxc2 {
        local-address 10.35.200.1;
        remote-address 10.35.200.2;
        remote-id 41060;
        interface so-0/1/0 {
            local-address 10.35.200.1;
            remote-address 10.35.200.2;
            remote-id 22278;
        }
    }
    peer oxc2 {
        address 10.255.255.37;
        control-channel so-0/3/2.0;
        te-link te-oxc2;
    }
}
}

```

Verifying Your Work

To verify proper operation of GMPLS, you can use the following commands:

- `show link-management (te-link | peer)`
- `show link-management routing (te-link | peer)`
- `show mpls lsp (bidirectional | unidirectional)`
- `show mpls lsp (detail | extensive)`
- `show ospf interface`
- `show ospf neighbor`
- `show rsvp interface link-management`
- `show rsvp session (bidirectional | unidirectional)`
- `show rsvp session te-link`
- `show rsvp session detail`
- `show rsvp neighbor detail`
- `show ted database extensive`
- `traceroute` (using the `lsp` flag with RSVP protocol-level trace options)

The following sections show the output of these commands used with the configuration example:

- Router A Status on page 24
- Router C Status on page 29

Router A Status

After you enter the **local-address**, **remote-address**, and **interface** parameters in traffic engineering link **te-oxc1** and commit the changes, the router automatically creates a local ID at the **te-link** and **interface** levels of the **[edit protocols link-management]** hierarchy. To view these IDs, issue the **show link-management te-link** command.

```
user@RouterA> show link-management te-link
TE link name: te-oxc1 , State: Up
  Local identifier: 8255, Remote identifier: 0 , Local address: 10.35.100.2,
Remote address: 10.35.100.1, Encoding: SDH/SONET,
  Minimum bandwidth: 155.52Mbps, Maximum bandwidth: 155.52Mbps, Total bandwidth:
155.52Mbps, Available bandwidth: 0bps
  Name           Local ID Remote ID   Bandwidth In use   LSP
  so-0/0/0       65535    0         155.52Mbps No
```

Once you find these values on Router A, configure them as remote IDs at the same hierarchy levels on OXC1. In this example, 8255 is Router A's local traffic engineering link ID (configure this as the traffic engineering link remote-ID on OXC1) and 65535 is Router A's local interface ID (configure this as the interface remote-ID on OXC1).

After you configure both remote IDs on both peers, the GMPLS traffic engineering links should work. Using the same command as before, you can verify whether the link is functional, with both remote and local IDs in place:

```
user@RouterA> show link-management te-link
TE link name: te-oxc1, State: Up
  Local identifier: 8255, Remote identifier: 8256, Local address: 10.35.100.2, Remote
address: 10.35.100.1, Encoding: SDH/SONET,
  Minimum bandwidth: 155.52Mbps, Maximum bandwidth: 155.52Mbps, Total bandwidth:
155.52Mbps, Available bandwidth: 0bps
  Name      Local ID Remote ID  Bandwidth In use  LSP
so-0/0/0    65535   65536   155.52Mbps Yes    gmpls-lsp1
```

To further verify proper operation, use the following commands:

```
user@RouterA> show link-management routing peer
Peer name: oxc1, System identifier: 13892
State: Up, Control address: 10.255.255.69
Control-channel      State
so-0/3/0.0           Active
```

```
user@RouterA> show link-management routing te-link
TE link name: te-oxc1, State: Up
  Local identifier: 8255, Remote identifier: 8256, Local address: 10.35.100.2,
Remote address: 10.35.100.1, Encoding: SDH/SONET,
  Minimum bandwidth: 155.52Mbps, Maximum bandwidth: 155.52Mbps, Total bandwidth:
155.52Mbps, Available bandwidth: 0bps
```

```
user@RouterA> show link-management peer
Peer name: oxc1, System identifier: 13892
State: Up, Control address: 10.255.255.69
Control-channel      State
so-0/3/0.0           Active
TE links:
te-oxc1
```

```
user@RouterA> show mpls lsp bidirectional
Ingress LSP: 1 sessions
To          From          State Rt ActivePath      P      LSPname
10.255.255.40 10.255.255.35 Up    0 path-lsp1      *      gmpls-lsp1 Bidir
Total 1 displayed, Up 1, Down 0
```

```
Egress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0
```

```
Transit LSP: 0 sessions
Total 0 displayed, Up 0, Down 0
```

```
user@RouterA> show mpls lsp bidirectional extensive
Ingress LSP: 1 sessions
```

```
10.255.255.40
  From: 10.255.255.35, State: Up, ActiveRoute: 0, LSPname: gmpls-lsp1
Bidirectional
ActivePath: path-lsp1 (primary)
LoadBalance: Random
Signal type: STM-1
Encoding type: SDH/SONET, Switching type: Fiber, GPID: PPP
*Primary path-lsp1      State: Up
  Bandwidth: 155.52Mbps
  Computed ERO (S [L] denotes strict [loose] hops): (CSPF metric: 2)
```

```

10.35.100.1 S 10.35.150.1 S 10.35.200.1 S
Received RR0:
10.35.100.1 10.35.150.1 10.35.200.1
7 Nov 7 15:47:11 Selected as active path
6 Nov 7 15:47:11 Record Route: 10.35.100.1 10.35.150.1 10.35.200.1
5 Nov 7 15:47:11 Up
4 Nov 7 15:47:11 Update LSP Encoding Type
3 Nov 7 15:47:11 Originate Call
2 Nov 7 15:47:11 CSPF: computation result accepted
1 Nov 7 15:46:41 CSPF failed: no route toward 10.255.255.40
Created: Thu Nov 7 15:46:38 2002
Total 1 displayed, Up 1, Down 0

```

```

Egress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

```

```

Transit LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

```

If you configure an LMP peer interface in OSPF, you can see that this virtual interface is treated as a point-to-point link. To view this, use the **show ospf interface** command.

```

user@RouterA> show ospf interface

```

Interface	State	Area	DR ID	BDR ID	Nbrs
lo0.0	DR	0.0.0.0	10.255.255.35	0.0.0.0	0
oxc1	PtToPt	0.0.0.0	0.0.0.0	0.0.0.0	1

The next command is useful because it indicates whether RSVP is disabled on the control channel. It also shows the state of the reservations on the traffic engineering links.

```

user@RouterA> show rsvp interface link-management
RSVP interface: 1 active
oxc1 State Up
Active control channel: so-0/3/0.0 RSVP disabled
TElink: te-oxc1, Local identifier: 8255
ActiveResv 1, PreemptionCnt 0
StaticBW: 155.52Mbps, ReservedBW: 155.52Mbps, AvailableBW: 0bps

```

```

user@RouterA> show rsvp session detail
Ingress RSVP: 1 sessions

```

```

10.255.255.40
From: 10.255.255.35, LSPstate: Up, ActiveRoute: 0
LSPname: gmpls-lsp1, LSPpath: Primary
Bidirectional, Upstream label in: 27676, Upstream label out: -
Suggested label received: -, Suggested label sent: 27676
Recovery label received: -, Recovery label sent: 60444
Resv style: 1 FF, Label in: -, Label out: 60444
Time left: -, Since: Thu Nov 7 15:47:11 2002
Tspec: rate 0bps size 0bps peak 1.544Mbps m 20 M 1500
Port number: sender 1 receiver 17 protocol 0
PATH rcvfrom: localclient
PATH sentto: 10.255.255.40 (oxc1) 157 pkts
RESV rcvfrom: 10.255.255.40 (oxc1) 71 pkts
Explct route: 10.35.100.1 10.35.150.1 10.35.200.1
Record route: <self> 10.35.100.1 10.35.150.1 10.35.200.1
Total 1 displayed, Up 1, Down 0

```

```

Egress RSVP: 0 sessions
Total 0 displayed, Up 0, Down 0

```

```
Transit RSVP: 0 sessions
Total 0 displayed, Up 0, Down 0
```

```
user@RouterA> show rsvp session bidirectional
Ingress RSVP: 1 sessions
To          From          State Rt Style LabelIn LabelOut LSPName
10.255.255.40 10.255.255.35 Up    0  1 FF      -    60444 gmpls-lsp1 Bidir
Total 1 displayed, Up 1, Down 0
```

```
Egress RSVP: 0 sessions
Total 0 displayed, Up 0, Down 0
```

```
Transit RSVP: 0 sessions
Total 0 displayed, Up 0, Down 0
```

```
user@RouterA> show rsvp session te-link te-oxc1
Ingress RSVP: 1 sessions
To          From          State Rt Style LabelIn LabelOut LSPName
10.255.255.40 10.255.255.35 Up    0  1 FF      -    60444 gmpls-lsp1 Bidir
Total 1 displayed, Up 1, Down 0
```

```
Egress RSVP: 0 sessions
Total 0 displayed, Up 0, Down 0
```

```
Transit RSVP: 0 sessions
Total 0 displayed, Up 0, Down 0
```

```
user@RouterA> show ted database extensive
TED database: 0 ISIS nodes 4 INET nodes
NodeID: 10.255.255.35
  Type: Rtr, Age: 2178 secs, LinkIn: 4, LinkOut: 5
  Protocol: OSPF(0.0.0.0)
    To: 10.255.255.69, Local: 10.35.100.2, Remote: 10.35.100.1
    Metric: 1
    Static BW: 155.52Mbps
    Reservable BW: 155.52Mbps
    Available BW [priority] bps:
      [0] 0bps      [1] 0bps      [2] 0bps      [3] 0bps
      [4] 0bps      [5] 0bps      [6] 0bps      [7] 0bps
    Interface Switching Capability Descriptor(1):
      Switching type: Fiber
      Encoding type: SDH/SONET
      Maximum LSP BW [priority] bps:
        [0] 155.52Mbps [1] 155.52Mbps [2] 155.52Mbps [3] 155.52Mbps
        [4] 155.52Mbps [5] 155.52Mbps [6] 155.52Mbps [7] 155.52Mbps
      Minimum LSP BW: 155.52Mbps
      Interface MTU: 2595
NodeID: 10.255.255.37
  Type: Rtr, Age: 2852 secs, LinkIn: 5, LinkOut: 5
  Protocol: OSPF(0.0.0.0)
    To: 10.255.255.69, Local: 10.35.150.1, Remote: 10.35.150.2
    Metric: 1
    Static BW: 622.08Mbps
    Reservable BW: 622.08Mbps
    Available BW [priority] bps:
      [0] 622.08Mbps [1] 622.08Mbps [2] 622.08Mbps [3] 622.08Mbps
      [4] 622.08Mbps [5] 622.08Mbps [6] 622.08Mbps [7] 622.08Mbps
    Interface Switching Capability Descriptor(1):
      Switching type: Fiber
      Encoding type: SDH/SONET
      Maximum LSP BW [priority] bps:
```

```

        [0] 622.08Mbps [1] 622.08Mbps [2] 622.08Mbps [3] 622.08Mbps
        [4] 622.08Mbps [5] 622.08Mbps [6] 622.08Mbps [7] 622.08Mbps
        Minimum LSP BW: 622.08Mbps
        Interface MTU: 2597
    To: 10.255.255.40, Local: 10.35.200.2, Remote: 10.35.200.1
    Metric: 1
    Static BW: 155.52Mbps
    Reservable BW: 155.52Mbps
    Available BW [priority] bps:
        [0] 0bps [1] 0bps [2] 0bps [3] 0bps
        [4] 0bps [5] 0bps [6] 0bps [7] 0bps
    Interface Switching Capability Descriptor(1):
        Switching type: Fiber
        Encoding type: SDH/SONET
        Maximum LSP BW [priority] bps:
            [0] 155.52Mbps [1] 155.52Mbps [2] 155.52Mbps [3] 155.52Mbps
            [4] 155.52Mbps [5] 155.52Mbps [6] 155.52Mbps [7] 155.52Mbps
        Minimum LSP BW: 155.52Mbps
        Interface MTU: 2600
NodeID: 10.255.255.40
Type: Rtr, Age: 2854 secs, LinkIn: 2, LinkOut: 2
Protocol: OSPF(0.0.0.0)
    To: 10.255.255.37, Local: 10.35.200.1, Remote: 10.35.200.2
    Metric: 1
    Static BW: 155.52Mbps
    Reservable BW: 155.52Mbps
    Available BW [priority] bps:
        [0] 0bps [1] 0bps [2] 0bps [3] 0bps
        [4] 0bps [5] 0bps [6] 0bps [7] 0bps
    Interface Switching Capability Descriptor(1):
        Switching type: Fiber
        Encoding type: SDH/SONET
        Maximum LSP BW [priority] bps:
            [0] 155.52Mbps [1] 155.52Mbps [2] 155.52Mbps [3] 155.52Mbps
            [4] 155.52Mbps [5] 155.52Mbps [6] 155.52Mbps [7] 155.52Mbps
        Minimum LSP BW: 155.52Mbps
        Interface MTU: 2600
NodeID: 10.255.255.69
Type: Rtr, Age: 2832 secs, LinkIn: 8, LinkOut: 7
Protocol: OSPF(0.0.0.0)
    To: 10.255.255.35, Local: 10.35.100.1, Remote: 10.35.100.2
    Metric: 1
    Static BW: 155.52Mbps
    Reservable BW: 155.52Mbps
    Available BW [priority] bps:
        [0] 0bps [1] 0bps [2] 0bps [3] 0bps
        [4] 0bps [5] 0bps [6] 0bps [7] 0bps
    Interface Switching Capability Descriptor(1):
        Switching type: Fiber
        Encoding type: SDH/SONET
        Maximum LSP BW [priority] bps:
            [0] 155.52Mbps [1] 155.52Mbps [2] 155.52Mbps [3] 155.52Mbps
            [4] 155.52Mbps [5] 155.52Mbps [6] 155.52Mbps [7] 155.52Mbps
        Minimum LSP BW: 155.52Mbps
        Interface MTU: 2595
    To: 10.255.255.37, Local: 10.35.150.2, Remote: 10.35.150.1
    Metric: 1
    Static BW: 622.08Mbps
    Reservable BW: 622.08Mbps
    Available BW [priority] bps:
        [0] 622.08Mbps [1] 622.08Mbps [2] 622.08Mbps [3] 622.08Mbps

```



```

[4] 622.08Mbps [5] 622.08Mbps [6] 622.08Mbps [7] 622.08Mbps
Interface Switching Capability Descriptor(1):
  Switching type: Fiber
  Encoding type: SDH/SONET
  Maximum LSP BW [priority] bps:
    [0] 622.08Mbps [1] 622.08Mbps [2] 622.08Mbps [3] 622.08Mbps
    [4] 622.08Mbps [5] 622.08Mbps [6] 622.08Mbps [7] 622.08Mbps
  Minimum LSP BW: 622.08Mbps
  Interface MTU: 2597

```

```

user@RouterA> show rsvp neighbor detail
RSVP neighbor: 1 learned
Address: 10.255.255.40 via: oxc1 status: Up
  Last changed time: 50:52, Idle: 0 sec, Up cnt: 1, Down cnt: 0
  Message received: 145
  Hello: sent 338, received: 338, interval: 9 sec
  Remote instance: 0x643087c7, Local instance: 0x3271e0a4
  Refresh reduction: not operational
  Link protection: disabled
  Bypass LSP: does not exist, Backup routes: 0, Backup LSPs: 0

```

Router C Status

After you enter the **local-address**, **remote-address**, and **interface** parameters in traffic engineering link **te-oxc2** and commit the changes, the router automatically creates a local ID at the **te-link** and **interface** levels of the **[edit protocols link-management]** hierarchy. To view these IDs, issue the **show link-management te-link** command.

```

user@RouterC> show link-management te-link
  TE link name: te-oxc2, State: Up
    Local identifier: 41059, Remote identifier: 0, Local address: 10.35.200.1, Remote
address: 10.35.200.2, Encoding: SDH/SONET,
    Minimum bandwidth: 155.52Mbps, Maximum bandwidth: 155.52Mbps, Total bandwidth:
155.52Mbps, Available bandwidth: 0bps
    Name Local ID Remote ID Bandwidth In use LSP
so-0/1/0 22277 0 155.52Mbps No

```

Once you see what these values are, configure them as remote IDs at the same hierarchy levels on OXC2 where you found them on Router C. In this example, 41059 is Router C's local traffic engineering link ID (configure this as the traffic engineering link **remote-ID** on OXC2) and 22277 is Router C's local interface ID (configure this as the interface **remote-ID** on OXC2).

After you configure both remote IDs on both peers, the GMPLS traffic engineering links should work. Using the same command as before, you can determine whether the link is functional, with both remote and local IDs in place:

```

user@RouterC> show link-management te-link
  TE link name: te-oxc2, State: Up
    Local identifier: 41059, Remote identifier: 41060, Local address: 10.35.200.1, Remote
address: 10.35.200.2, Encoding: SDH/SONET,
    Minimum bandwidth: 155.52Mbps, Maximum bandwidth: 155.52Mbps, Total bandwidth:
155.52Mbps, Available bandwidth: 0bps
    Name Local ID Remote ID Bandwidth In use LSP
so-0/1/0 22277 22278 155.52Mbps Yes gmpls-lsp1

```

The other **show** commands operate like those in "Router A Status" on page 24.

Example: Configuring Traffic Engineering Link and Interface Identifiers

Figure 3: Traffic Engineering Link and Interface ID Example

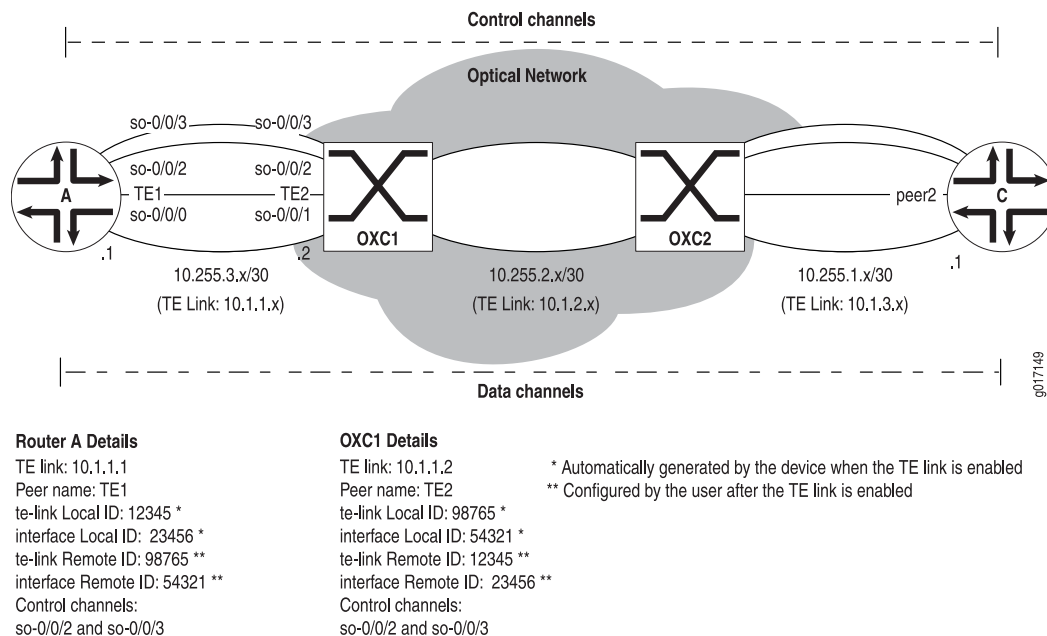


Figure 3 on page 30 shows where the IDs come from and where you must assign them. This example highlights the connections between Router A and OXC1, but the same configuration concepts apply to all pairs of peers.

First, you configure a traffic engineering link named TE1 on Router A, which contains the local address 10.1.1.1, remote address 10.1.1.2, data channel interface **so-0/0/0**, and control channel interfaces **so-0/0/2** and **so-0/0/3**. You also configure a traffic engineering link named TE2 on OXC1, which contains the local address 10.1.1.2, remote address 10.1.1.1, data channel interface **so-0/0/1**, and control channel interfaces **so-0/0/2** and **so-0/0/3**. When the traffic engineering links are enabled on Router A and OXC1, these two peer devices each generate two local IDs: one for the traffic engineering link itself and one for the logical interface.

If Router A has a local ID of 12345 for its traffic engineering link and a local ID of 23456 for its interface, you must configure 12345 as the traffic engineering link **remote-ID** and 23456 as the interface **remote-ID** on the TE2 traffic engineering link of OXC1. Similarly, if OXC1 has local IDs of 98765 for its traffic engineering link and 54321 for its interface, you configure Router A's TE1 traffic engineering link with 98765 as the traffic engineering link **remote-ID** and 54321 as the interface **remote-ID**.

To complete the full data path, you need to enable LMP on each link in the path. This means you must configure remote-ID and local-ID pairs between linked devices.

The diagram illustrates a four-node network topology. Four nodes, represented by circles with internal switch symbols and numbered 1, 2, 3, and 4, are arranged horizontally. Above the nodes is a dashed line labeled "Control channel", and below is a dashed line labeled "Data channel".

Control Channel Connections:

- Node 1 to Node 2: Interface .1 (10.35.100.0/30) to .2 (10.255.71.242). Label: so-1/1/0, so-5/0/0.
- Node 2 to Node 3: Interface .5 (10.35.100.4/30) to .6 (10.255.71.243). Label: so-1/1/0, so-0/1/1.
- Node 3 to Node 4: Interface .21 (10.35.100.20/30) to .22 (10.255.71.244). Label: so-1/1/0, so-6/0/0.

Data Channel Connections:

- Node 1 to Node 2: Label: sonet (TE Link: 10.35.1.1).
- Node 2 to Node 3: Label: sonet.
- Node 3 to Node 4: Label: sonet (TE Link: 10.35.1.2).
- Node 1 to Node 4: A curved connection labeled .9 (10.35.100.8/30) and .10 (so-3/0/2).

Node Details:

- Node 1: Switch symbol with up, down, left, and right arrows.
- Node 2: Switch symbol with up, down, left, and right arrows.
- Node 3: Switch symbol with up, down, left, and right arrows.
- Node 4: Switch symbol with up, down, left, and right arrows.

IP Address Summary:

- Node 1: .1 (10.35.100.0/30)
- Node 2: .2 (10.255.71.242), .5 (10.35.100.4/30)
- Node 3: .6 (10.255.71.243), .21 (10.35.100.20/30)
- Node 4: .22 (10.255.71.244), .10 (so-3/0/2)

Configure a bidirectional GMPLS LSP to reach Router 4. Use the loopback address of 10.255.71.244 as the destination for the LSP, disable CSPF, and configure a strict path to the remote address of the traffic engineering link.

31

```
        address 10.35.100.1/30;
    }
    family iso;
    family mpls;
}
}
lo0 {
    unit 0 {
        family inet {
            address 10.255.71.241/32;
        }
        family iso {
            address 47.0005.0000.0000.0000.0000.0102.5507.1241.00;
        }
    }
}
}
routing-options {
    router-id 10.255.71.241;
}
protocols {
    rsvp {
        interface all;
    }
    mpls {
        label-switched-path gmpls-router1-router4 {
            to 10.255.71.244;
            lsp-attributes {
                switching-type fiber;
            }
            no-cspf;
            primary path1;
        }
        path path1 {
            10.35.1.2 strict;
        }
        interface so-1/1/0.0;
    }
    isis {
        interface so-1/1/0.0;
        interface fxp0.0 {
            disable;
        }
        interface lo0.0 {
            passive;
        }
    }
}
link-management {
    te-link sonet {
        local-address 10.35.1.1;
        remote-address 10.35.1.2;
        remote-id 8070;
        interface so-0/1/2 {
            remote-id 21303;
        }
    }
}
```

```

peer router4 {
  address 10.255.71.244;
  lmp-control-channel so-1/1/0.0 {
    remote-address 10.35.100.22;
  }
  te-link sonet;
}
traceoptions {
  file lmp.logs size 5m files 10 world-readable;
  flag hello-packets;
  flag packets;
  flag state;
}
}
}

```

On Router 2, configure IS-IS, MPLS, and RSVP to provide backbone connectivity for GMPLS and LMP between Routers 1 and 3.

```

Router 2 [edit]
interfaces {
  so-1/1/0 {
    description "Connection to Router 3";
    unit 0 {
      family inet {
        address 10.35.100.5/30;
      }
      family iso;
      family mpls;
    }
  }
  so-5/0/0 {
    description "Connection to Router 1";
    unit 0 {
      family inet {
        address 10.35.100.2/30;
      }
      family iso;
      family mpls;
    }
  }
  lo0 {
    unit 0 {
      family inet {
        address 10.255.71.242/32;
      }
      family iso {
        address 47.0005.0000.0000.0000.0000.0000.0102.5507.1242.00;
      }
    }
  }
}
routing-options {
  router-id 10.255.71.242;
}
protocols {

```

```
    rsvp {
      interface all;
    }
    mpls {
      interface all;
    }
    isis {
      interface so-1/1/0.0;
      interface so-5/0/0.0;
      interface fxp0.0 {
        disable;
      }
      interface lo0.0 {
        passive;
      }
    }
  }
  link-management {
    traceoptions {
      file lmp.logs size 5m files 10 world-readable;
      flag hello-packets;
      flag packets;
      flag state;
    }
  }
}
```

The configuration of Router 3 is very similar to the configuration on Router 2. Configure IS-IS, MPLS, and RSVP to provide backbone connectivity for GMPLS and LMP between Routers 2 and 4.

```
Router 3  [edit]
            interfaces {
              so-0/1/1 {
                description "Connection to Router 2";
                unit 0 {
                  family inet {
                    address 10.35.100.6/30;
                  }
                  family iso;
                  family mpls;
                }
              }
              so-1/1/0 {
                description "Connection to Router 4";
                unit 0 {
                  family inet {
                    address 10.35.100.21/30;
                  }
                  family iso;
                  family mpls;
                }
              }
            }
            lo0 {
              unit 0 {
                family inet {
                  address 10.255.71.243/32;
                }
              }
            }
          }
```

```

    }
    family iso {
        address 47.0005.0000.0000.0000.0000.0000.0102.5507.1243.00;
    }
}
}
routing-options {
    router-id 10.255.71.243;
}
protocols {
    rsvp {
        interface all;
    }
    mpls {
        interface all;
    }
    isis {
        interface so-0/1/1.0;
        interface so-1/1/0.0;
        interface fxp0.0 {
            disable;
        }
        interface lo0.0 {
            passive;
        }
    }
}
link-management {
    traceoptions {
        file lmp.logs size 5m files 10 world-readable;
        flag hello-packets;
        flag packets;
        flag state;
    }
}
}

```

On Router 4, complete the example by configuring IS-IS, MPLS, and RSVP to support the LMP control channel. For the control channel, use the **so-6/0/0** interface as the origin and specify **10.35.100.1** (the **so-1/1/0** interface on Router 1) as the remote address. For the traffic engineering link, use the **so-3/0/2** interface as the origin, swap the local address and remote address pair configured on Router 1 and add them here, and configure the remote IDs to match the local IDs generated by the remote peer.

Because GMPLS LSPs are bidirectional by default, you do not need to configure a return path to Router 1.

```

Router 4 [edit]
interfaces {
    so-3/0/2 {
        description "Data channel to Router 1";
        unit 0 {
            family inet {
                address 10.35.100.10/30;
            }
        }
    }
}

```

```
        family iso;
        family mpls;
    }
}
so-6/0/0 {
    description "Control channel to Router 1";
    unit 0 {
        family inet {
            address 10.35.100.22/30;
        }
        family iso;
        family mpls;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 10.255.71.244/32;
        }
        family iso {
            address 47.0005.0000.0000.0000.0000.0000.0102.5507.1244.00;
        }
    }
}
}
routing-options {
    router-id 10.255.71.244;
}
protocols {
    rsvp {
        interface all;
    }
    mpls {
        interface all;
    }
    isis {
        interface so-6/0/0.0;
        interface fxp0.0 {
            disable;
        }
        interface lo0.0 {
            passive;
        }
    }
}
link-management {
    te-link sonet {
        local-address 10.35.1.2;
        remote-address 10.35.1.1;
        remote-id 8070;
        interface so-3/0/2 {
            remote-id 22279;
        }
    }
}
peer router1 {
    address 10.255.71.241;
    lmp-control-channel so-6/0/0.0 {
```



```

        remote-address 10.35.100.1;
    }
    te-link sonet;
}
traceoptions {
    file lmp.logs size 5m files 10 world-readable;
    flag hello-packets;
    flag packets;
    flag state;
}
}
}

```

Verifying Your Work

To verify proper operation of LMP control channels, use the following commands:

- **show link-management peer**
- **show link-management statistics**
- **show link-management te-link**

The following sections show the output of these commands used with the configuration example:

- Router 1 Status on page 37
- Router 4 Status on page 38

Router 1 Status

On Router 1, issue the **show link-management** commands to verify that the control channel, traffic engineering links, and LMP negotiations are working as expected. The **show link-management peer** command indicates peer addresses, names, and identifiers, as well as control channel identifiers. The **show link-management statistics** command shows the number of LMP hellos and messages that have been exchanged. The **show link-management te-link** command displays names and identifiers configured for the traffic engineering link.

```

user@router1> show link-management peer
Peer name: router4, System identifier: 37495
State: Up, Control address: 10.255.70.103
  CC local ID CC remote ID State      TxSeqNum  RcvSeqNum  Flags
    22515      38882 Up          1692      1691
TE links:
sonet

```

```

user@router1> show link-management statistics
Statistics for peer router4
Received packets
  ConfigAck: 1
  Hello: 748
  LinkSummary: 13
  LinkSummaryAck: 1
Small packets: 0
Wrong protocol version: 0
Messages for unknown peer: 0

```

```

Messages for bad state: 0
Stale acknowledgements: 0
Stale negative acknowledgements: 0
Sent packets
  Config: 24
  Hello: 748
  LinkSummary: 13
  LinkSummaryAck: 13
Retransmitted packets
  Config: 18
  LinkSummary: 9
Dropped packets
  Config: 5
  LinkSummary: 3

```

```

user@router1> show link-management te-link
TE link name: sonet, State: Up
Local identifier: 8070, Remote identifier: 8070, Local address: 10.35.1.1,
Remote address: 10.35.1.2, Encoding: SDH/SONET, Switching: PSC-1,
Minimum bandwidth: 622.08Mbps, Maximum bandwidth: 622.08Mbps,
Total bandwidth: 622.08Mbps, Available bandwidth: 622.08Mbps
  Name      State Local ID Remote ID   Bandwidth Used  LSP-name
so-0/1/2   Up      22279   21303   622.08Mbps    No

```

Router 4 Status

On Router 4, issue the **show link-management** commands to verify that the control channel, traffic engineering links, and LMP negotiations are being reciprocated by Router 1.

```

user@router4> show link-management peer
Peer name: router1, System identifier: 56483
State: Up, Control address: 10.255.71.242
  CC local ID CC remote ID State      TxSeqNum RcvSeqNum Flags
      38882      22515 Up          1451      1450
TE links:
sonet

```

```

user@router4> show link-management statistics
Statistics for peer router1
Received packets
  Config: 1
  Hello: 255
  LinkSummary: 1
  LinkSummaryAck: 1
Small packets: 0
Wrong protocol version: 0
Messages for unknown peer: 0
Messages for bad state: 0
Stale acknowledgements: 0
Stale negative acknowledgements: 0
Sent packets
  Config: 31
  ConfigAck: 1
  Hello: 255
  LinkSummary: 13
  LinkSummaryAck: 1
Retransmitted packets
  Config: 23
  LinkSummary: 9
Dropped packets
  Config: 7

```

LinkSummary: 3

```

user@router4> show link-management te-link
TE link name: sonet, State: Up
Local identifier: 8070, Remote identifier: 8070, Local address: 10.35.1.2,
Remote address: 10.35.1.1, Encoding: SDH/SONET, Switching: PSC-1,
Minimum bandwidth: 622.08Mbps, Maximum bandwidth: 622.08Mbps,
Total bandwidth: 622.08Mbps, Available bandwidth: 622.08Mbps
  Name          State Local ID Remote ID    Bandwidth Used LSP-name
  so-3/0/2      Up          21303    22279    622.08Mbps    No

```

For More Information

For additional information about implementing GMPLS, see the following:

- *Junos MPLS Applications Configuration Guide*
- RFC 2205, *Resource ReSerVation Protocol (RSVP)*
- RFC 3471, *Generalized Multi-Protocol Label Switching (GMPLS)—Signaling Functional Description*
- RFC 3473, *Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions*
- RFC 4204, *Link Management Protocol (LMP)* (The Junos OS supports sections 3 and 4)
- Internet draft draft-ietf-ccamp-gmpls-rsvp-te-ason-02.txt, *Generalized MPLS (GMPLS) RSVP-TE Signalling in support of Automatically Switched Optical Network (ASON)* (expires January 2005)
- Internet draft draft-ietf-ccamp-gmpls-sonet-sdh-08.txt, *GMPLS Extensions for SONET and SDH Control* (expires August 2003)
- Internet draft draft-ietf-ccamp-ospf-gmpls-extensions-12.txt, *OSPF Extensions in Support of Generalized MPLS* (expires April 2004)
- Internet draft draft-ietf-mpls-bundle-04.txt, *Link Bundling in MPLS Traffic Engineering* (expires January 2003)
- Internet draft draft-ietf-mpls-lsp-hierarchy-08.txt, *LSP Hierarchy with Generalized MPLS TE* (expires March 2003)
- Internet draft draft-ietf-ccamp-gmpls-routing-09.txt, *Routing Extensions in Support of Generalized MPLS* (expires April 2004)

PART 2

Index

- Index on page 43

Index

G

GMPLS

configuration procedure.....	9
example configuration	
static peers.....	19
operational mode commands.....	24
options	
administratively down nonpacket	
LSPs.....	14
graceful restart.....	15
graceful teardown.....	13
LMP control channel.....	16
overlay model.....	15
peer model.....	14
overview.....	3
sample configuration	
LMP control channel.....	31
system requirements.....	7

L

Link Management Protocol See LMP

LMP.....	16
control channel.....	16
<i>See also</i> GMPLS	

S

system requirements	
GMPLS.....	7

