



Junos[®] OS

Channelized Interfaces Configuration Guide

Release
11.2



Published: 2011-05-20

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 Channelized Interfaces Configuration Guide

Release 11.2

Copyright © 2011, Juniper Networks, Inc.

All rights reserved.

Revision History

May 2011—R1 JUNOS 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)).

Abbreviated Table of Contents

	About This Guide	xxi
Part 1	Channelized Interfaces Configuration Statements Overview	
Chapter 1	Channelized Interfaces Configuration Statements and Hierarchy	3
Part 2	Configuring Channelized Interfaces	
Chapter 2	Channelized Interfaces	27
Chapter 3	Configuring Channelized OC48/STM16 IQE Interfaces	47
Chapter 4	Configuring Channelized OC12/STM4 Interfaces	65
Chapter 5	Configuring Channelized OC3 IQ and IQE Interfaces	97
Chapter 6	Configuring Channelized STM1 Interfaces	105
Chapter 7	Configuring Channelized T3 Interfaces	123
Chapter 8	Configuring Channelized T1 Interfaces	137
Chapter 9	Configuring Channelized E1 Interfaces	143
Chapter 10	Configuring Channelized E1 PRI and T1 PRI Interfaces	151
Part 3	Configuring E1, E3, T1, and T3 Interfaces	
Chapter 11	Configuring E1 Interfaces	161
Chapter 12	Configuring E3 Interfaces	169
Chapter 13	Configuring T1 Interfaces	177
Chapter 14	Configuring T3 Interfaces	187
Part 4	Configuring Frame Relay	
Chapter 15	Configuring Frame Relay	203
Part 5	Channelized Interface Configuration Statements	
Chapter 16	Summary of Channelized Interface Configuration Statements	217
Part 6	Index	
	Index	297
	Index of Statements and Commands	303

Table of Contents

	About This Guide	xxi
	JUNOS Documentation and Release Notes	xxi
	Objectives	xxii
	Audience	xxii
	Supported Routing Platforms	xxii
	Using the Indexes	xxiii
	Using the Examples in This Manual	xxiii
	Merging a Full Example	xxiii
	Merging a Snippet	xxiv
	Documentation Conventions	xxiv
	Documentation Feedback	xxvi
	Requesting Technical Support	xxvi
	Self-Help Online Tools and Resources	xxvii
	Opening a Case with JTAC	xxvii
Part 1	Channelized Interfaces Configuration Statements Overview	
Chapter 1	Channelized Interfaces Configuration Statements and Hierarchy	3
	[edit chassis] Hierarchy Level	3
	[edit interfaces] Hierarchy Level	4
	[edit logical-systems] Hierarchy Level	20
Part 2	Configuring Channelized Interfaces	
Chapter 2	Channelized Interfaces	27
	Channelized Interfaces Overview	27
	Channelized Interface Capabilities	28
	Data-Link Connection Identifiers on Channelized Interfaces	30
	Clock Sources on Channelized Interfaces	32
	Channelized E1 and T1 PIM Properties	35
	Channelized IQ and IQE Interfaces Properties	36
	Structure of Channelized IQ and Channelized IQE PICs	38
Chapter 3	Configuring Channelized OC48/STM16 IQE Interfaces	47
	Channelized OC48/STM16 IQE Interfaces Overview	47
	Configuring Channelized OC48/STM16 IQE Interfaces in SONET Mode	49
	Configuring OC12 Interfaces	49
	Example: Configuring OC12 Interfaces	50
	Configuring OC3 Interfaces	50
	Example: Configuring OC3 Interfaces	51

Configuring T3 Interfaces	51
Example: Configuring T3 Interfaces	52
Configuring T1 Interfaces	52
Example: Configuring T1 Interfaces	53
Configuring Fractional T1 Interfaces	54
Example: Configuring Fractional T1 Interfaces	54
Configuring NxDS0 Interfaces	54
Example: Configuring NxDS0 Interfaces	56
Configuring Channelized OC48/STM16 IQE Interfaces (SDH Mode)	56
Configuring a Channelized OC48/STM16 IQE PIC for SDH Mode	57
Configuring Clear Channel STM1 and STM4 Interfaces	57
Configuring Channelized AU-4 Interfaces	58
Example: Configuring Channelized AU-4 Interfaces	58
Configuring E3 Interfaces	58
Example: Configuring E3 Interfaces	59
Configuring E1 or Channelized E1 Interfaces	59
Example: Configuring E1 and Channelized E1 Interfaces	60
Configuring NxDS0 IQE Interfaces	60
Example: Configuring NxDS0 IQE Interfaces	60
Configuring T3 or Channelized T3 Interfaces	60
Example: Configuring T3 or Channelized T3 Interfaces	61
Configuring T1 or Channelized T1 Interfaces	61
Example: Configuring T1 or Channelized T1 Interfaces	61
Configuring Link PIC Failover on Channelized OC48/STM16 IQE Interfaces	62
Example: Configuring Channelized OC48 Interfaces with Partitioned Channels	62
Chapter 4	
Configuring Channelized OC12/STM4 Interfaces	65
Channelized OC12/STM4 IQ and IQE Interfaces Overview	65
Channelization of OC12/STM4 IQ and Channelized OC12/STM4 IQE PICs (SONET Mode)	65
Channelization of OC12/STM4 IQE PIC (SDH Mode)	66
Channelization of OC12/STM4 IQ PIC (SDH Mode)	67
Channelization of OC12 PIC (SONET Mode)	68
Configuring Channelized OC12/STM4 IQ and IQE Interfaces (SONET Mode)	69
Configuring an OC12/STM4 Interface	69
Configuring T3 Interfaces	69
Example: Configuring T3 Interfaces	70
Configuring OC3 Interfaces	71
Example: Configuring OC3 Interfaces	71
Configuring T1 Interfaces on Channelized OC12 IQ and IQE Interfaces	71
Example: Configuring T1 Interfaces	72
Configuring NxDS0 Interfaces	73
Example: Configuring NxDS0 Interfaces	74

Configuring Fractional T1 Interfaces	75
Example: Configuring Fractional T1 Interfaces	75
Configuring Channelized OC12/STM4 IQE Interfaces (SDH Mode)	76
Configuring Channelized OC12/STM4 IQE PICs for SDH Mode	76
Configuring an Unpartitioned SDH (VC-4-4C) Interface on a Channelized OC12/STM4 IQE PIC	77
Example: Configuring an Unpartitioned SDH (VC-4-4C) Interface	77
Configuring SDH (VC-4) Interfaces on Channelized OC12/STM4 IQE PICs	77
Example: Configuring SDH (VC-4) Interfaces	78
Configuring Channelized AU-4 Interfaces	78
Example: Configuring Channelized AU-4 Interfaces	79
Configuring E3 Interfaces	79
Example: Configuring E3 Interfaces	80
Configuring E1 or Channelized E1 Interfaces	80
Example: Configuring E1 or Channelized CE1 Interfaces	81
Configuring NxDS0 Interfaces on Channelized OC12/STM4 IQE PICs	81
Example: Configuring NxDS0 Interfaces	81
Configuring Channelized OC12/STM4 IQ Interfaces (SDH Mode)	81
Configuring Channelized OC12/STM4 IQ PICs for SDH Mode	82
Configuring an Unpartitioned SDH (VC-4-4C) Interface on a Channelized OC12/STM4 IQ PIC	83
Example: Configuring an Unpartitioned SDH (VC-4-4C) Interface	83
Configuring SDH (VC-4) Interfaces on Channelized OC12/STM4 IQ PICs	83
Example: Configuring SDH (VC-4) Interfaces	84
Configuring Channelized AU-4 Interfaces	84
Example: Configuring Channelized AU-4 Interfaces	84
Configuring T3 or Channelized T3 Interfaces Under Channelized AU-4 Interfaces	85
Example: Configuring T3 or Channelized T3 Interfaces	85
Configuring T1 or Channelized T1 Interfaces Under Channelized AU-4 Interfaces	85
Example: Configuring T1 or Channelized T1 Interfaces Under Channelized AU-4 Interfaces	86
Configuring T1 or Channelized T1 Interfaces Under Channelized T3 Interfaces	86
Example: Configuring T1 or Channelized T1 Interfaces Under Channelized T3 Interfaces	86
Configuring NxDS0 Interfaces on Channelized OC12/STM4 IQ PICs	87
Example: Configuring NxDS0 Interfaces	87
Configuring Channelized OC12 Interfaces	87
Example: Configuring Channelized OC12 Interfaces	88
Configuring Link PIC Failover on Channelized OC12/STM4 IQ and IQE Interfaces	90
Example: Configuring a Channelized OC12 IQ Interface as an Unpartitioned Clear Channel	90
Example: Configuring Channelized OC12 Interfaces with Partitioned Channels	94

Chapter 5	Configuring Channelized OC3 IQ and IQE Interfaces	97
	Channelized OC3 IQ and IQE Overview	97
	Partitions, OC Slices, Interface Types, and Time Slots	98
	Configuring a Clear Channel on Channelized OC3 IQ and IQE PICs	99
	Configuring T3 Interfaces on IQ and IQE Interfaces	99
	Example: Configuring T3 Interfaces	100
	Configuring T1 and NxDS0 Interfaces	100
	Example: Configuring T1 and NxDS0 Interfaces	102
	Example: Setting Remote Loopback and Running BERT Tests on NxDS0 Interfaces	103
	Configuring Fractional T1 IQ Interfaces	103
	Example: Configuring Fractional T1 IQ Interfaces	104
	Configuring Link PIC Failover on Channelized OC3 IQ and IQE Interfaces	104
Chapter 6	Configuring Channelized STM1 Interfaces	105
	Channelized STM1 Interfaces Overview	105
	Configuring Channelized STM1 IQ and IQE Interfaces	105
	Configuring an STM1 IQ or STM1 IQE Interface	105
	Configuring E1 IQ and IQE Interfaces	106
	Example: Configuring E1 IQ and IQE Interfaces	106
	Configuring Fractional E1 IQ and IQE Interfaces	107
	Example: Configuring Fractional E1 Interfaces	107
	Configuring an NxDS0 IQ Interface	108
	Example: Configuring an NxDS0 IQ Interface	109
	Example: Configuring Channelized STM1 IQ and IQE Interfaces	109
	Configuring Channelized STM1 Interfaces	110
	Configuring Channelized STM1 Interface Properties	111
	Configuring Virtual Tributary Mapping of Channelized STM1 Interfaces	112
	Configuring Link PIC Failover on Channelized STM1 Interfaces	119
	Example: Configuring Channelized STM1 Interfaces	119
Chapter 7	Configuring Channelized T3 Interfaces	123
	Configuring Channelized T3 IQ Interfaces	123
	Configuring T3 IQ Interfaces	123
	Configuring T1 IQ Interfaces	123
	Example: Configuring T1 IQ and IQE Interfaces	124
	Configuring Fractional T1 IQ and IQE Interfaces	124
	Example: Configuring Fractional T1 IQ Interfaces	125
	Configuring an NxDS0 IQ Interface	125
	Example: Configuring an NxDS0 IQ Interface	126
	Configuring Channelized DS3-to-DS0 Interfaces	126
	Configuring Channelized DS3-to-DS1 Interfaces	129
	Example: Configuring Channelized T3 IQ Interfaces	130
	Examples: Configuring Channelized DS3-to-DS0 Interfaces	131
	Examples: Configuring Channelized DS3-to-DS1 Interfaces	134

Chapter 8	Configuring Channelized T1 Interfaces	137
	Channelized T1 IQ and IQE Interfaces Overview	137
	Configuring Channelized T1 IQ and IQE Interfaces	137
	Configuring T1 IQ and IQE Interfaces	137
	Configuring Fractional T1 IQ and IQE Interfaces	138
	Example: Configuring Fractional T1 IQ and IQE Interfaces	138
	Configuring NxDS0 IQ and IQE Interfaces	139
	Example: Configuring an NxDS0 IQ or IQE Interface	139
	Configuring Payload Loopback	139
	Configuring Channelized T1 Interface Properties	140
	Example: Configuring Channelized T1 IQ and IQE Interfaces	141
Chapter 9	Configuring Channelized E1 Interfaces	143
	Channelized E1 IQ and IQE Interfaces Overview	143
	Configuring Channelized E1 IQ and IQE Interfaces	143
	Configuring E1 IQ and IQE Interfaces	143
	Configuring Fractional E1 IQ and IQE Interfaces	144
	Example: Configuring Fractional E1 IQ and IQE Interfaces	144
	Configuring NxDS0 IQ and IQE Interfaces	144
	Example: Configuring an NxDS0 IQ or IQE Interface	145
	Configuring Channelized E1 Interfaces	145
	Configuring Channelized E1 Interface Properties	146
	Example: Configuring Channelized E1 IQ or IQE Interfaces	147
	Example: Configuring Channelized E1 Interfaces	148
Chapter 10	Configuring Channelized E1 PRI and T1 PRI Interfaces	151
	Channelized E1 PRI and T1 PRI Overview	151
	Configuring a Clear Channel on a Dual-Port Channelized T1-E1 PIM	152
	Configuring a Channelized T1/E1 Interface to Drop and Insert Time Slots	152
	Configuring Primary Rate Interfaces	153
	Allocating B-Channels for Dialout	155
	Configuring PRI Interfaces	155
	Example: Configuring a Channelized T1 Interface as Primary Rate Interface	156
Part 3	Configuring E1, E3, T1, and T3 Interfaces	
Chapter 11	Configuring E1 Interfaces	161
	E1 Interfaces Overview	161
	Configuring E1 Physical Interface Properties	162
	Configuring E1 BERT Properties	162
	Configuring the E1 Frame Checksum	163
	Configuring E1 Framing	163
	Configuring the E1 Idle Cycle Flag	164
	Configuring E1 Data Inversion	164
	Configuring E1 Loopback Capability	164
	Example: Configuring E1 Loopback Capability	165
	Configuring E1 Start and End Flags	166
	Configuring Fractional E1 Time Slots	166
	Example: Configuring Fractional E1 Time Slots	167

Chapter 12	Configuring E3 Interfaces	169
	E3 Interfaces Overview	169
	Configuring E3 Physical Interface Properties	170
	Configuring E3 BERT Properties	170
	Configuring the E3 CSU Compatibility Mode	171
	Configuring the E3 Frame Checksum	172
	Configuring the E3 Idle Cycle Flag	173
	Configuring E3 Data Inversion	173
	Configuring E3 Loopback Capability	173
	Example: Configuring E3 Loopback Capability	174
	Configuring E3 HDLC Payload Scrambling	175
	Configuring the E3 Start and End Flags	175
	Configuring E3 IQ and IQE Unframed Mode	175
Chapter 13	Configuring T1 Interfaces	177
	T1 Interfaces Overview	177
	Configuring T1 Physical Interface Properties	178
	Configuring T1 BERT Properties	178
	Configuring the T1 Buildout	179
	Configuring T1 Byte Encoding	179
	Configuring T1 CRC Error Major Alarm Thresholds	180
	Configuring T1 CRC Error Minor Alarm Thresholds	180
	Configuring T1 Data Inversion	181
	Configuring the T1 Frame Checksum	181
	Configuring the T1 Remote Loopback Response	181
	Configuring T1 Framing	182
	Configuring T1 Line Encoding	182
	Configuring T1 Loopback Capability	182
	Configuring the T1 Idle Cycle Flag	184
	Configuring T1 Start and End Flags	184
	Configuring Fractional T1 Time Slots	185
	Example: Configuring Fractional T1 Time Slots	185
Chapter 14	Configuring T3 Interfaces	187
	T3 Interfaces Overview	187
	Configuring T3 Physical Interface Properties	188
	Configuring T3 BERT Properties	188
	Disabling T3 C-Bit Parity Mode	189
	Configuring the T3 CSU Compatibility Mode	190
	Configuring the T3 Frame Checksum	192
	Configuring the T3 FEAC Response	193
	Configuring the T3 Idle Cycle Flag	193
	Configuring the T3 Line Buildout	194
	Configuring T3 Loopback Capability	194
	Configuring T3 HDLC Payload Scrambling	196
	Configuring T3 Start and End Flags	196
	Examples: Configuring T3 Interfaces	197

Part 4	Configuring Frame Relay	
Chapter 15	Configuring Frame Relay	203
	Frame Relay Overview	203
	Configuring Frame Relay Interface Encapsulation	204
	Configuring the Frame Relay Encapsulation on a Physical Interface	204
	Example: Configuring the Encapsulation on a Physical Interface	207
	Configuring the Frame Relay Encapsulation on a Logical Interface	207
	Configuring Frame Relay Control Bit Translation	208
	Configuring the Media MTU on Frame Relay Interfaces	209
	Setting the Protocol MTU with Frame Relay Encapsulation	209
	Configuring Frame Relay Keepalives	210
	Configuring Tunable Keepalives for Frame Relay LMI	210
	Configuring Inverse Frame Relay ARP	211
	Configuring the Router as a DCE with Frame Relay Encapsulation	212
	Configuring Frame Relay DLCIs	212
	Configuring a Point-to-Point Frame Relay Connection	212
	Configuring a Point-to-Multipoint Frame Relay Connection	213
	Configuring a Multicast-Capable Frame Relay Connection	214
Part 5	Channelized Interface Configuration Statements	
Chapter 16	Summary of Channelized Interface Configuration Statements	217
	address	218
	advertise-interval	219
	aps	220
	authentication-key	221
	bchannel-allocation	221
	bert-algorithm	222
	bert-error-rate	224
	bert-period	225
	buildout (T1 Interfaces)	226
	byte-encoding	227
	bytes	228
	cbit-parity	229
	clocking	230
	compatibility-mode	231
	crc-minor-alarm-threshold	232
	data-input	233
	dce	234
	dlci	235
	ds0-options	236
	e1-options	237
	e3-options	238
	encapsulation	239
	encapsulation (Logical Interface)	240
	encapsulation (Physical Interface)	243
	family	247
	fcs	251

feac-loop-respond	252
force	253
framing (E1, E3, and T1 Interfaces)	254
hold-time (APS)	255
idle-cycle-flag	256
interface-type	257
inverse-arp	258
invert-data	259
isdn-options	260
line-encoding	261
lmi (Frame Relay)	262
lmi-type	263
lockout	264
long-buildout	264
loop-timing	265
loopback (ADSL, DS0, E1/E3, SONET/SDH, SHDSL, and T1/T3)	266
loopback-clear-timer	267
mtu	268
multicast-dlci	269
multipoint-destination	270
neighbor	271
no-keepalives	272
no-partition	273
no-termination-request	274
oc-slice	275
paired-group	275
partition	276
payload-scrambler	277
protect-circuit	278
remote-loopback-respond	278
request	279
revert-time	279
sonet-options	280
start-end-flag	282
switching-mode	283
t1-options	284
t3-options	285
timeslots	286
translate-discard-eligible	287
unframed	287
unit	288
vtmapping	294
working-circuit	294

Part 6

Index

Index	297
Index of Statements and Commands	303

List of Figures

Part 2	Configuring Channelized Interfaces	
Chapter 2	Channelized Interfaces	27
	Figure 1: Channelized OC48/STM16 IQE PIC (in SONET Mode)	38
	Figure 2: Channelized OC48/STM16 IQE PIC (in SDH Mode)	39
	Figure 3: Channelized OC12 IQ PIC and Channelized OC12/STM4 IQE PIC (in SONET Mode)	39
	Figure 4: Channelized OC12/STM4 IQE PIC (in SDH Mode)	40
	Figure 5: Channelized OC12/STM4 IQ PIC (in SDH Mode)	40
	Figure 6: Channelized OC3 Ports (in SONET Mode) on Channelized OC3 IQ and Channelized OC3/STM1 IQE PICs	41
	Figure 7: Channelized CSTM1 Ports (in SDH Mode) on Channelized OC3/STM1 IQE PIC	41
	Figure 8: Channelized STM1 IQ PIC	42
	Figure 9: Channelized CDS3/E3 IQE PIC (in DS3 Mode)	42
	Figure 10: Channelized CDS3/E3 IQE PIC (in E3 Mode)	42
	Figure 11: Channelized DS3 IQ PIC	43
	Figure 12: Channelized T1 IQ and IQE PIC	43
	Figure 13: Channelized E1 IQ and IQE PIC	43
Chapter 3	Configuring Channelized OC48/STM16 IQE Interfaces	47
	Figure 14: Sample Channelization of OC48/STM16 IQE PIC (SONET Mode)	47
	Figure 15: Sample Channelization of OC48/STM16 IQE PIC (SDH Mode)	48
	Figure 16: Sample Channelization of OC48/STM16 IQE PIC to E3 Channels	49
	Figure 17: T1 Interfaces on a Channelized OC48 PIC	53
	Figure 18: Sample Channelization of OC48 IQE PIC	56
Chapter 4	Configuring Channelized OC12/STM4 Interfaces	65
	Figure 19: Sample Channelization of OC12/STM4 IQ or IQE PIC (SONET Mode)	66
	Figure 20: Sample Channelization of OC12/STM4 IQE PIC (SDH Mode)	67
	Figure 21: Sample Channelization of OC12/STM4 IQ PIC (SDH Mode)	68
	Figure 22: Sample Channelization of OC12 PIC (non IQ and IQE)	68
	Figure 23: T1 Interfaces on a Channelized OC12 PIC	72
	Figure 24: Sample Channelization of OC12 IQE PIC	74
Chapter 5	Configuring Channelized OC3 IQ and IQE Interfaces	97
	Figure 25: Channelized OC3 IQ Interface Example for Show Interfaces Controller	98
	Figure 26: T1 Interfaces on a Channelized OC3 PIC	101
	Figure 27: Sample Channelization of OC3 IQ or IQE PIC	102
Chapter 7	Configuring Channelized T3 Interfaces	123

	Figure 28: Sample Channelization of DS3 IQ or IQE PIC	130
Part 3	Configuring E1, E3, T1, and T3 Interfaces	
Chapter 11	Configuring E1 Interfaces	161
	Figure 29: Remote and Local E1 Loopback	165
Chapter 12	Configuring E3 Interfaces	169
	Figure 30: Remote and Local E3 Loopback	174
Chapter 13	Configuring T1 Interfaces	177
	Figure 31: Remote and Local T1 Loopback	183
Chapter 14	Configuring T3 Interfaces	187
	Figure 32: Remote and Local T3 Loopback	194

List of Tables

	About This Guide	xxi
	Table 1: Notice Icons	xxv
	Table 2: Text and Syntax Conventions	xxv
Part 2	Configuring Channelized Interfaces	
Chapter 2	Channelized Interfaces	27
	Table 3: Frame Relay DLCI Limitations for Channelized Interfaces	30
	Table 4: Per Unit Scheduler DLCI Limitations for Channelized Interfaces	31
	Table 5: Protocol Family Combinations	32
	Table 6: Clocking Capabilities by Channelized PIC Type	33
	Table 7: Structural Differences: Channelized IQE PICs	44
	Table 8: Structural Differences: Channelized IQ PICs	45
	Table 9: Structural Differences: Channelized PICs	46
Chapter 4	Configuring Channelized OC12/STM4 Interfaces	65
	Table 10: OC12-to-DS3 Numbering Scheme	88
Chapter 6	Configuring Channelized STM1 Interfaces	105
	Table 11: Channelized STM1-to-E1 Channel Mapping	112
	Table 12: Channelized STM1-to-T1 Channel Mapping	115
Chapter 7	Configuring Channelized T3 Interfaces	123
	Table 13: Ranges for Channelized DS3-to-DS0 Configuration	127
Chapter 8	Configuring Channelized T1 Interfaces	137
	Table 14: Ranges for Channelized T1 IQ Configuration	140
Chapter 9	Configuring Channelized E1 Interfaces	143
	Table 15: Ranges for Channelized E1 Configuration	146
Part 3	Configuring E1, E3, T1, and T3 Interfaces	
Chapter 12	Configuring E3 Interfaces	169
	Table 16: Subrate Values for E3 Digital Link Compatibility Mode	172
Chapter 14	Configuring T3 Interfaces	187
	Table 17: Subrate Values for T3 Digital Link Compatibility Mode	191
Part 4	Configuring Frame Relay	
Chapter 15	Configuring Frame Relay	203
	Table 18: PIC Support for Enhanced Frame Relay Encapsulation Types	206

About This Guide

This preface provides the following guidelines for using the *Junos[®] OS Channelized Interfaces Configuration Guide*:

- JUNOS Documentation and Release Notes on page xxi
- Objectives on page xxii
- Audience on page xxii
- Supported Routing Platforms on page xxii
- Using the Indexes on page xxiii
- Using the Examples in This Manual on page xxiii
- Documentation Conventions on page xxiv
- Documentation Feedback on page xxvi
- Requesting Technical Support on page xxvi

JUNOS Documentation and Release Notes

For a list of related JUNOS documentation, see
<http://www.juniper.net/techpubs/software/junos/>.

If the information in the latest release notes differs from the information in the documentation, follow the *JUNOS Release Notes*.

To obtain the most current version of all Juniper Networks[®] technical documentation, see the product documentation page on the Juniper Networks website at
<http://www.juniper.net/techpubs/>.

Juniper Networks supports a technical book program to publish books by Juniper Networks engineers and subject matter experts with book publishers around the world. These books go beyond the technical documentation to explore the nuances of network architecture, deployment, and administration using the Junos operating system (Junos OS) and Juniper Networks devices. In addition, the Juniper Networks Technical Library, published in conjunction with O'Reilly Media, explores improving network security, reliability, and availability using Junos OS configuration techniques. All the books are for sale at technical bookstores and book outlets around the world. The current list can be viewed at <http://www.juniper.net/books>.

Objectives

This guide provides an overview of the network interfaces features of the JUNOS Software and describes how to configure these properties on the routing platform.



NOTE: For additional information about the Junos OS—either corrections to or information that might have been omitted from this guide—see the software release notes at <http://www.juniper.net/>.

Audience

This guide is designed for network administrators who are configuring and monitoring a Juniper Networks M Series, MX Series, T Series, EX Series, or J Series router or switch.

To use this guide, you need a broad understanding of networks in general, the Internet in particular, networking principles, and network configuration. You must also be familiar with one or more of the following Internet routing protocols:

- Border Gateway Protocol (BGP)
- Distance Vector Multicast Routing Protocol (DVMRP)
- Intermediate System-to-Intermediate System (IS-IS)
- Internet Control Message Protocol (ICMP) router discovery
- Internet Group Management Protocol (IGMP)
- Multiprotocol Label Switching (MPLS)
- Open Shortest Path First (OSPF)
- Protocol-Independent Multicast (PIM)
- Resource Reservation Protocol (RSVP)
- Routing Information Protocol (RIP)
- Simple Network Management Protocol (SNMP)

Personnel operating the equipment must be trained and competent; must not conduct themselves in a careless, willfully negligent, or hostile manner; and must abide by the instructions provided by the documentation.

Supported Routing Platforms

For the features described in this manual, the JUNOS Software currently supports the following routing platforms:

- J Series
- M Series

- MX Series
- T Series

Using the Indexes

This reference contains two indexes: a complete index that includes topic entries, and an index of statements and commands only.

In the index of statements and commands, an entry refers to a statement summary section only. In the complete index, the entry for a configuration statement or command contains at least two parts:

- The primary entry refers to the statement summary section.
- The secondary entry, *usage guidelines*, refers to the section in a configuration guidelines chapter that describes how to use the statement or command.

Using the Examples in This Manual

If you want to use the examples in this manual, you can use the **load merge** or the **load merge relative** command. These commands cause the software to merge the incoming configuration into the current candidate configuration. If the example configuration contains the top level of the hierarchy (or multiple hierarchies), the example is a *full example*. In this case, use the **load merge** command.

If the example configuration does not start at the top level of the hierarchy, the example is a *snippet*. In this case, use the **load merge relative** command. These procedures are described in the following sections.

Merging a Full Example

To merge a full example, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration example into a text file, save the file with a name, and copy the file to a directory on your routing platform.

For example, copy the following configuration to a file and name the file **ex-script.conf**. Copy the **ex-script.conf** file to the **/var/tmp** directory on your routing platform.

```
system {
  scripts {
    commit {
      file ex-script.xml;
    }
  }
}
interfaces {
  fxp0 {
    disable;
    unit 0 {
      family inet {
```

```
        address 10.0.0.1/24;
    }
}
}
```

2. Merge the contents of the file into your routing platform configuration by issuing the **load merge** configuration mode command:

```
[edit]
user@host# load merge /var/tmp/ex-script.conf
load complete
```

Merging a Snippet

To merge a snippet, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration snippet into a text file, save the file with a name, and copy the file to a directory on your routing platform.

For example, copy the following snippet to a file and name the file **ex-script-snippet.conf**. Copy the **ex-script-snippet.conf** file to the **/var/tmp** directory on your routing platform.

```
commit {
  file ex-script-snippet.xml; }
```

2. Move to the hierarchy level that is relevant for this snippet by issuing the following configuration mode command:

```
[edit]
user@host# edit system scripts
[edit system scripts]
```

3. Merge the contents of the file into your routing platform configuration by issuing the **load merge relative** configuration mode command:

```
[edit system scripts]
user@host# load merge relative /var/tmp/ex-script-snippet.conf
load complete
```

For more information about the **load** command, see the [Junos OS CLI User Guide](#).

Documentation Conventions

Table 1 on page xxv defines notice icons used in this guide.

Table 1: Notice Icons




Icon	Meaning	Description
	Informational note	Indicates important features or instructions.
	Caution	Indicates a situation that might result in loss of data or hardware damage.
	Warning	Alerts you to the risk of personal injury or death.
	Laser warning	Alerts you to the risk of personal injury from a laser.

Table 2 on page xxv defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

Convention	Description	Examples
Bold text like this	Represents text that you type.	To enter configuration mode, type the configure command: <code>user@host> configure</code>
Fixed-width text like this	Represents output that appears on the terminal screen.	<code>user@host> show chassis alarms</code> <code>No alarms currently active</code>
<i>Italic text like this</i>	<ul style="list-style-type: none"> Introduces important new terms. Identifies book names. Identifies RFC and Internet draft titles. 	<ul style="list-style-type: none"> A policy <i>term</i> is a named structure that defines match conditions and actions. <i>Junos OS System Basics Configuration Guide</i> RFC 1997, <i>BGP Communities Attribute</i>
<i>Italic text like this</i>	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name: [edit] root@# set system domain-name <i>domain-name</i>
Text like this	Represents names of configuration statements, commands, files, and directories; interface names; configuration hierarchy levels; or labels on routing platform components.	<ul style="list-style-type: none"> To configure a stub area, include the stub statement at the [edit protocols ospf area area-id] hierarchy level. The console port is labeled CONSOLE.
< > (angle brackets)	Enclose optional keywords or variables.	<code>stub <default-metric metric>;</code>

Table 2: Text and Syntax Conventions (*continued*)

Convention	Description	Examples
(pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	broadcast multicast <i>(string1 string2 string3)</i>
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	rsvp { # Required for dynamic MPLS only
[] (square brackets)	Enclose a variable for which you can substitute one or more values.	community name members [community-ids]
Indentation and braces ({ })	Identify a level in the configuration hierarchy.	[edit] routing-options { static { route default { nexthop address; retain; } } }
;(semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
J-Web GUI Conventions		
Bold text like this	Represents J-Web graphical user interface (GUI) items you click or select.	<ul style="list-style-type: none"> In the Logical Interfaces box, select All Interfaces. To cancel the configuration, click Cancel.
> (bold right angle bracket)	Separates levels in a hierarchy of J-Web selections.	In the configuration editor hierarchy, select Protocols>Ospf .

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can send your comments to techpubs-comments@juniper.net, or fill out the documentation feedback form at <https://www.juniper.net/cgi-bin/docbugreport/>. If you are using e-mail, be sure to include the following information with your comments:

- Document or topic name
- URL or page number
- Software release version (if applicable)

Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active J-Care or JNASC support contract,

or are covered under warranty, and need postsales technical support, you can access our tools and resources online or open a case with JTAC.

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the JTAC User Guide located at <http://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf> .
- Product warranties—For product warranty information, visit <http://www.juniper.net/support/warranty/> .
- JTAC Hours of Operation —The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

Self-Help Online Tools and Resources

For quick and easy problem resolution, Juniper Networks has designed an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings: <http://www.juniper.net/customers/support/>
- Find product documentation: <http://www.juniper.net/techpubs/>
- Find solutions and answer questions using our Knowledge Base: <http://kb.juniper.net/>
- Download the latest versions of software and review release notes: <http://www.juniper.net/customers/csc/software/>
- Search technical bulletins for relevant hardware and software notifications: <https://www.juniper.net/alerts/>
- Join and participate in the Juniper Networks Community Forum: <http://www.juniper.net/company/communities/>
- Open a case online in the CSC Case Management tool: <http://www.juniper.net/cm/>

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: <https://tools.juniper.net/SerialNumberEntitlementSearch/>

Opening a Case with JTAC

You can open a case with JTAC on the Web or by telephone.

- Use the Case Management tool in the CSC at <http://www.juniper.net/cm/> .
- Call 1-888-314-JTAC (1-888-314-5822 toll-free in the USA, Canada, and Mexico).

For international or direct-dial options in countries without toll-free numbers, visit us at <http://www.juniper.net/support/requesting-support.html>

PART 1

Channelized Interfaces Configuration Statements Overview

- Channelized Interfaces Configuration Statements and Hierarchy on page 3

CHAPTER 1

Channelized Interfaces Configuration Statements and Hierarchy

The following network interfaces hierarchy listings show the complete configuration statement hierarchy for the indicated hierarchy levels, listing all possible configuration statements within the indicated hierarchy levels, and showing their level in the configuration hierarchy. When you are configuring the Junos OS, your current hierarchy level is shown in the banner on the line preceding the **user@host#** prompt.

This section contains the following topics:

- [edit chassis] Hierarchy Level on page 3
- [edit interfaces] Hierarchy Level on page 4
- [edit logical-systems] Hierarchy Level on page 20

[edit chassis] Hierarchy Level

```
chassis {
  aggregated-devices {
    ethernet {
      device-count number;
    }
    sonet {
      device-count number;
    }
  }
  channel-group number {
    ethernet {
      device-count number;
    }
    fpc slot-number {
      pic pic-number {
        adaptive-services {
          service-package (layer-2 | layer-3);
        }
        aggregate-ports;
        atm-cell-relay-accumulation;
        atm-l2circuit-mode (aal5 | cell | trunk trunk);
        cel {
          el link-number {
```

```

        channel-group group-number;
        timeslots time-slot-range;
    }
}
ct1 {
    t1 link-number {
        channel-group group-number;
        timeslots time-slot-range;
    }
}
ct3 {
    port port-number {
        t1 link-number {
            channel-group group-number;
            timeslots time-slot-range;
        }
    }
    framing sdh;
}
max-queues-per-interface number;
mlfr-uni-nni-bundles num-intf;
no-concatenate;
shdsl {
    pic-mode (1-port-atm | 2-port-atm);
}
vtmapping (klm | itu-t);
}
}
fpc slot-number{
pic pic-number{
    egress-policer-overhead bytes;
    ingress-policer-overhead bytes;
}
}
}
}

```

[edit interfaces] Hierarchy Level

The statements at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level can also be configured at the [edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*] hierarchy level.



NOTE: The accounting-profile statement is an exception to this rule. The accounting-profile statement can be configured at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level, but it cannot be configured at the [edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*] hierarchy level.

```

interfaces {
    traceoptions {
        file filename <files number> <match regular-expression> <size size> <world-readable |
        no-world-readable> ;
    }
}

```



```

    flag flag <disable>;
}
interface-name {
    accounting-profile name;
    aggregated-ether-options {
        (flow-control | no-flow-control);
        lacp {
            (active | passive);
            link-protection {
                disable;
                (revertive | non-revertive);
                periodic interval;
                system-priority priority;
            }
        }
        link-protection;
        link-speed speed;
        (loopback | no-loopback);
        mc-ae {
            chassis-id chassis-id;
            mc-ae-id mc-ae-id;
            mode (active-active | active-standby);
            redundancy-group group-id;
            status-control (active | standby);
        }
        minimum-links number;
        source-address-filter {
            mac-address;
        }
        (source-filtering | no-source-filtering);
    }
    aggregated-sonet-options {
        link-speed speed | mixed;
        minimum-links number;
    }
    atm-options {
        cell-bundle-size cells;
        ilmi;
        linear-red-profiles profile-name {
            high-plp-max-threshold percent;
            low-plp-max-threshold percent;
            queue-depth cells high-plp-threshold percent low-plp-threshold percent;
        }
        mpls {
            pop-all-labels {
                required-depth number;
            }
        }
        pic-type (atm1 | atm2);
        plp-to-clp;
        promiscuous-mode {
            vpi vpi-identifier;
        }
        scheduler-maps map-name {
            forwarding-class class-name {
                epd-threshold cells plp1 cells;
                linear-red-profile profile-name;
            }
        }
    }
}

```

```
        priority (high | low);
        transmit-weight (cells number | percent number);
    }
    vc-cos-mode (alternate | strict);
}
use-null-cw;
vpi vpi-identifier {
    maximum-vcs maximum-vcs;
    oam-liveness {
        down-count cells;
        up-count cells;
    }
    oam-period (seconds | disable);
    shaping {
        (cbr rate | rtvbr peak rate sustained rate burst length | vbr peak rate sustained rate
        burst length);
        queue-length number;
    }
}
}
clocking clock-source;
data-input (system | interface interface-name);
dce;
serial-options {
    clock-rate rate;
    clocking-mode (dce | internal | loop);
    control-polarity (negative | positive);
    cts-polarity (negative | positive);
    dcd-polarity (negative | positive);
    dce-options {
        control-signal (assert | de-assert | normal);
        cts (ignore | normal | require);
        dcd (ignore | normal | require);
        dsr (ignore | normal | require);
        dtr signal-handling-option;
        ignore-all;
        indication (ignore | normal | require);
        rts (assert | de-assert | normal);
        tm (ignore | normal | require);
    }
    dsr-polarity (negative | positive);
    dte-options {
        control-signal (assert | de-assert | normal);
        cts (ignore | normal | require);
        dcd (ignore | normal | require);
        dsr (ignore | normal | require);
        dtr signal-handling-option;
        ignore-all;
        indication (ignore | normal | require);
        rts (assert | de-assert | normal);
        tm (ignore | normal | require);
    }
}
dtr-circuit (balanced | unbalanced);
dtr-polarity (negative | positive);
encoding (nrz | nrzi);
indication-polarity (negative | positive);
```

```

line-protocol protocol;
loopback mode;
rts-polarity (negative | positive);
tm-polarity (negative | positive);
transmit-clock invert;
}
description text;
dialer-options {
    pool pool-name <priority priority>;
}
disable;
ds0-options {
    bert-algorithm algorithm;
    bert-error-rate rate;
    bert-period seconds;
    byte-encoding (nx56 | nx64);
    fcs (16 | 32);
    idle-cycle-flag (flags | ones);
    invert-data;
    loopback payload;
    start-end-flag (filler | shared);
}
e1-options {
    bert-error-rate rate;
    bert-period seconds;
    fcs (16 | 32);
    framing (g704 | g704-no-crc4 | unframed);
    idle-cycle-flag (flags | ones);
    invert-data;
    loopback (local | remote);
    start-end-flag (filler | shared);
    timeslots time-slot-range;
}
e3-options {
    atm-encapsulation (direct | plcp);
    bert-algorithm algorithm;
    bert-error-rate rate;
    bert-period seconds;
    framing feat;
    compatibility-mode (digital-link | kentrox | larscom) <subrate value>;
    fcs (16 | 32);
    framing (g.751 | g.832);
    idle-cycle-flag (filler | shared);
    invert-data;
    loopback (local | remote);
    (payload-scrambler | no-payload-scrambler);
    start-end-flag (filler | shared);
    (unframed | no-unframed);
}
encapsulation type;
es-options {
    backup-interface es-fpc/pic/port;
}
fastether-options {
    802.3ad aex;
    (flow-control | no-flow-control);
}

```

```
ignore-l3-incompletes;
ingress-rate-limit rate;
(loopback | no-loopback);
mpls {
  pop-all-labels {
    required-depth number;
  }
}
source-address-filter {
  mac-address;
}
(source-filtering | no-source-filtering);
}
flexible-vlan-tagging;
gigether-options {
  802.3ad aex;
  (asynchronous-notification | no-asynchronous-notification);
  (auto-negotiation | no-auto-negotiation) remote-fault <local-interface-online |
    local-interface-offline>;
  auto-reconnect seconds;
  (flow-control | no-flow-control);
  ignore-l3-incompletes;
  (loopback | no-loopback);
  mpls {
    pop-all-labels {
      required-depth number;
    }
  }
}
no-auto-mdix;
source-address-filter {
  mac-address;
}
(source-filtering | no-source-filtering);
ethernet-switch-profile {
  (mac-learn-enable | no-mac-learn-enable);
  tag-protocol-id [ tpids ];
  ethernet-policer-profile {
    input-priority-map {
      ieee802.1p premium [ values ];
    }
    output-priority-map {
      classifier {
        premium {
          forwarding-class class-name {
            loss-priority (high | low);
          }
        }
      }
    }
  }
}
policer cos-policer-name {
  aggregate {
    bandwidth-limit bps;
    burst-size-limit bytes;
  }
  premium {
    bandwidth-limit bps;
  }
}
```

9

```
n393dce number;  
n393dte number;  
t391dte seconds;  
t392dce seconds;  
}  
lsq-failure-options {  
  no-termination-request;  
  [ trigger-link-failure interface-name ];  
}  
mac mac-address;  
mlfr-uni-nni-bundle-options {  
  acknowledge-retries number;  
  acknowledge-timer milliseconds;  
  action-red-differential-delay (disable-tx | remove-link);  
  drop-timeout milliseconds;  
  fragment-threshold bytes;  
  cisco-interoperability send-lip-remove-link-for-link-reject;  
  hello-timer milliseconds;  
  link-layer-overhead percent;  
  lmi-type (ansi | itu);  
  minimum-links number;  
  mrru bytes;  
  n391 number;  
  n392 number;  
  n393 number;  
  red-differential-delay milliseconds;  
  t391 seconds;  
  t392 seconds;  
  yellow-differential-delay milliseconds;  
}  
modem-options {  
  dialin (console | routable);  
  init-command-string initialization-command-string;  
}  
mtu bytes;  
multiservice-options {  
  (core-dump | no-core-dump);  
  (syslog | no-syslog);  
}  
native-vlan-id number;  
no-gratuitous-arp-request;  
no-keepalives;  
no-partition {  
  interface-type type;  
}  
otn-options {  
  fec (efec | gfec | none);  
  (laser-enable | no-laser-enable);  
  (line-loopback | no-line-loopback);  
  pass-thru;  
  rate (fixed-stuff-bytes | no-fixed-stuff-bytes | pass-thru);  
  transmit-payload-type number;  
  trigger (oc-lof | oc-lom | oc-los | oc-wavelength-lock | odu-ais | odu-bbe-th | odu-bdi  
    | odu-es-th | odu-lck | odu-oci | odu-sd | odu-ses-th | odu-ttim | odu-uas-th |  
    opu-ptm | otu-ais | otu-bbe-th | otu-bdi | otu-es-th | otu-fec-deg | otu-fec-exe |  
    otu-iae | otu-sd | otu-ses-th | otu-ttim | otu-uas-th);
```

```

tti;
}
optics-options {
    wavelength nm;
    alarm alarm-name {
        (syslog | link-down);
    }
    warning warning-name {
        (syslog | link-down);
    }
}
partition partition-number oc-slice oc-slice-range interface-type type;
timeslots time-slot-range;
passive-monitor-mode;
per-unit-scheduler;
ppp-options {
    chap {
        access-profile name;
        default-chap-secret name;
        local-name name;
        passive;
    }
    compression {
        acfc;
        pfc;
    }
    dynamic-profile profile-name;
    no-termination-request;
    pap {
        access-profile name;
        local-name name;
        local-password password;
        compression;
    }
}
receive-bucket {
    overflow (discard | tag);
    rate percentage;
    threshold bytes;
}
redundancy-options {
    priority sp-fpc/pic/port;
    secondary sp-fpc/pic/port;
    hot-standby;
}
satop-options {
    payload-size n;
}
schedulers number;
serial-options {
    clock-rate rate;
    clocking-mode (dce | internal | loop);
    control-polarity (negative | positive);
    cts-polarity (negative | positive);
    dcd-polarity (negative | positive);
    dce-options {

```

```
control-signal (assert | de-assert | normal);
cts (ignore | normal | require);
dcd (ignore | normal | require);
dsr (ignore | normal | require);
dtr signal-handling-option;
ignore-all;
indication (ignore | normal | require);
rts (assert | de-assert | normal);
tm (ignore | normal | require);
}
dsr-polarity (negative | positive);
dte-options {
    control-signal (assert | de-assert | normal);
    cts (ignore | normal | require);
    dcd (ignore | normal | require);
    dsr (ignore | normal | require);
    dtr signal-handling-option;
    ignore-all;
    indication (ignore | normal | require);
    rts (assert | de-assert | normal);
    tm (ignore | normal | require);
}
dtr-circuit (balanced | unbalanced);
dtr-polarity (negative | positive);
encoding (nrz | nrzi);
indication-polarity (negative | positive);
line-protocol protocol;
loopback mode;
rts-polarity (negative | positive);
tm-polarity (negative | positive);
transmit-clock invert;
}
services-options {
    inactivity-timeout seconds;
    open-timeout seconds;
    session-limit {
        maximum number;
        rate new-sessions-per-second;
    }
    syslog {
        host hostname {
            facility-override facility-name;
            log-prefix prefix-number;
            services priority-level;
        }
    }
}
shdsl-options {
    annex (annex-a | annex-b);
    line-rate line-rate;
    loopback (local | remote);
    snr-margin {
        current margin;
        snext margin;
    }
}
```



```

sonet-options {
  aggregate asx;
  aps {
    advertise-interval milliseconds;
    annex-b;
    authentication-key key;
    force;
    hold-time milliseconds;
    lockout;
    neighbor address;
    paired-group group-name;
    preserve-interface;
    protect-circuit group-name;
    request;
    revert-time seconds;
    switching-mode (bidirectional | unidirectional);
    working-circuit group-name;
  }
  bytes {
    c2 value;
    e1-quiet value;
    f1 value;
    f2 value;
    s1 value;
    z3 value;
    z4 value;
  }
  fcs (16 | 32);
  loopback (local | remote);
  mpls {
    pop-all-labels {
      required-depth number;
    }
  }
  path-trace trace-string;
  (payload-scrambler | no-payload-scrambler);
  rfc-2615;
  trigger {
    defect ignore;
    hold-time up milliseconds down milliseconds;
  }
  vtmapping (itu-t | klm);
  (z0-increment | no-z0-increment);
}
speed (10m | 100m | 1g | oc3 | oc12 | oc48);
stacked-vlan-tagging;
switch-options {
  switch-port port-number {
    (auto-negotiation | no-auto-negotiation);
    speed (10m | 100m | 1g);
    link-mode (full-duplex | half-duplex);
  }
}
t1-options {
  bert-algorithm algorithm;
  bert-error-rate rate;
}

```

```
bert-period seconds;  
buildout value;  
byte-encoding (nx56 | nx64);  
crc-major-alarm-threshold (1e-3 | 5e-4 | 1e-4 | 5e-5 | 1e-5);  
crc-minor-alarm-threshold (1e-3 | 5e-4 | 1e-4 | 5e-5 | 1e-5 | 5e-6 | 1e-6);  
fcs (16 | 32);  
framing (esf | sf);  
idle-cycle-flag (flags | ones);  
invert-data;  
line-encoding (ami | b8zs);  
loopback (local | payload | remote);  
remote-loopback-respond;  
start-end-flag (filler | shared);  
timeslots time-slot-range;  
}  
t3-options {  
  atm-encapsulation (direct | plcp);  
  bert-algorithm algorithm;  
  bert-error-rate rate;  
  bert-period seconds;  
  buildout feet;  
  (cbit-parity | no-cbit-parity);  
  compatibility-mode (adtran | digital-link | kentrox | larscom | verilink) <subrate  
    value>;  
  fcs (16 | 32);  
  (feac-loop-respond | no-feac-loop-respond);  
  idle-cycle-flag value;  
  (long-buildout | no-long-buildout);  
  (loop-timing | no-loop-timing);  
  loopback (local | payload | remote);  
  (mac | no-mac);  
  (payload-scrambler | no-payload-scrambler);  
  start-end-flag (filler | shared);  
}  
traceoptions {  
  flag flag <flag-modifier> <disable>;  
}  
transmit-bucket {  
  overflow discard;  
  rate percentage;  
  threshold bytes;  
}  
(traps | no-traps);  
unidirectional;  
vlan-tagging;  
vlan-vci-tagging;  
unit logical-unit-number {  
  accept-source-mac {  
    mac-address mac-address {  
      policer {  
        input cos-policer-name;  
        output cos-policer-name;  
      }  
    }  
  }  
}  
}  
accounting-profile name;
```

```

allow-any-vci;
atm-scheduler-map (map-name | default);
backup-options {
    interface interface-name;
}
bandwidth rate;
cell-bundle-size cells;
clear-dont-fragment-bit;
compression {
    rtp {
        f-max-period number;
        maximum-contexts number <force>;
        queues [ queue-numbers ];
        port {
            minimum port-number;
            maximum port-number;
        }
    }
}
compression-device interface-name;
copy-tos-to-outer-ip-header;
demux-destination family;
demux-source family;
demux-options {
    underlying-interface interface-name;
}
description text;
dial-options {
    l2tp-interface-id name;
    (dedicated | shared);
}
dialer-options {
    activation-delay seconds;
    callback;
    callback-wait-period time;
    deactivation-delay seconds;
    dial-string [ dial-string-numbers ];
    idle-timeout seconds;
    incoming-map {
        caller (caller-id | accept-all);
        initial-route-check seconds;
        load-interval seconds;
        load-threshold percent;
        pool pool-name;
        redial-delay time;
        watch-list {
            [ routes ];
        }
    }
}
disable;
disable-mlppp-inner-ppp-pfc;
dlci dlci-identifier;
drop-timeout milliseconds;
dynamic-call-admission-control {
    activation-priority priority;

```

```
    bearer-bandwidth-limit kilobits-per-second;
  }
  encapsulation type;
  epd-threshold cells plp1 cells;
  fragment-threshold bytes;
  inner-vlan-id-range start start-id end end-id;
  input-vlan-map {
    (pop | pop-pop | pop-swap | push | push-push | swap | swap-push | swap-swap);
    inner-tag-protocol-id tpid;
    inner-vlan-id number;
    tag-protocol-id tpid;
    vlan-id number;
  }
  interleave-fragments;
  inverse-arp;
  layer2-policer {
    input-policer policer-name;
    input-three-color policer-name;
    output-policer policer-name;
    output-three-color policer-name;
  }
  link-layer-overhead percent;
  minimum-links number;
  mrru bytes;
  multicast-dlci dlci-identifier;
  multicast-vci vpi-identifier.vci-identifier;
  multilink-max-classes number;
  multipoint;
  oam-liveness {
    down-count cells;
    up-count cells;
  }
  oam-period (seconds | disable);
  output-vlan-map {
    (pop | pop-pop | pop-swap | push | push-push | swap | swap-push | swap-swap);
    inner-tag-protocol-id tpid;
    inner-vlan-id number;
    tag-protocol-id tpid;
    vlan-id number;
  }
  passive-monitor-mode;
  peer-unit unit-number;
  plp-to-clp;
  point-to-point;
  ppp-options {
    chap {
      access-profile name;
      default-chap-secret name;
      local-name name;
      passive;
    }
    compression {
      acfc;
      pfc;
      pap;
      default-pap-password password;
    }
  }
}
```

```

        local-name name;
        local-password password;
        passive;
    }
    dynamic-profile profile-name;
    lcp-max-conf-req number;
    lcp-restart-timer milliseconds;
    loopback-clear-timer seconds;
    ncp-max-conf-req number;
    ncp-restart-timer milliseconds;
}
pppoe-options {
    access-concentrator name;
    auto-reconnect seconds;
    (client | server);
    service-name name;
    underlying-interface interface-name;
}
proxy-arp;
service-domain (inside | outside);
shaping {
    (cbr rate | rtvbr peak rate sustained rate burst length | vbr peak rate sustained rate
    burst length);
    queue-length number;
}
short-sequence;
transmit-weight number;
(traps | no-traps);
trunk-bandwidth rate;
trunk-id number;
tunnel {
    backup-destination address;
    destination address;
    key number;
    routing-instance {
        destination routing-instance-name;
    }
    source source-address;
    ttl number;
}
vci vpi-identifier.vci-identifier;
vci-range start start-vci end end-vci;
vpi vpi-identifier;
vlan-id number;
vlan-id-list [vlan-id vlan-id-vlan-id];
vlan-id-range number-number;
vlan-tags inner tpid.vlan-id outer tpid.vlan-id;
vlan-tags-outer tpid.vlan-id inner-list [vlan-id vlan-id-vlan-id];
family family {
    accounting {
        destination-class-usage;
        source-class-usage {
            direction;
        }
    }
}
access-concentrator name;

```

```
address address {
  destination address;
}
bundle ml-fpc/pic/port | ls-fpc/pic/port);
duplicate-protection;
dynamic-profile profile-name;
filter {
  group filter-group-number;
  input filter-name;
  input-list {
    [ filter-names ];
    output filter-name;
  }
  output-list {
    [ filter-names ];
  }
}
ipsec-sa sa-name;
keep-address-and-control;
max-sessions number;
mtu bytes;
multicast-only;
negotiate-address;
no-redirects;
policer {
  arp policer-template-name;
  input policer-template-name;
  output policer-template-name;
}
primary;
proxy inet-address address;
receive-options-packets;
receive-ttl-exceeded;
remote (inet-address address | mac-address address);
rpf-check {
  fail-filter filter-name;
  mode loose;
}
sampling {
  direction;
}
service {
  input {
    service-set service-set-name <service-filter filter-name>;
    post-service-filter filter-name;
  }
  output {
    service-set service-set-names <service-filter filter-name>;
  }
}
service-name-table table-name
targeted-broadcast {
  forward-and-send-to-re;
  forward-only;
}
(translate-discard-eligible | no-translate-discard-eligible);
```

19

[edit logical-systems] Hierarchy Level

The following lists the statements that can be configured at the **[edit logical-systems]** hierarchy level that are also documented in this manual. For more information about logical systems, see the *Junos OS Routing Protocols Configuration Guide*.

```
logical-systems logical-system-name {  
  interfaces interface-name {  
    unit logical-unit-number {  
      accept-source-mac {  
        mac-address mac-address {  
          policer {  
            input cos-policer-name;  
            output cos-policer-name;  
          }  
        }  
      }  
    }  
    allow-any-vci;  
    atm-scheduler-map (map-name | default);  
    bandwidth rate;  
    backup-options {  
      interface interface-name;  
    }  
    cell-bundle-size cells;  
    clear-dont-fragment-bit;  
    compression {  
      rtp {  
        f-max-period number;  
        port {  
          minimum port-number;  
          maximum port-number;  
        }  
      }  
      queues [ queue-numbers ];  
    }  
  }  
  compression-device interface-name;  
  description text;  
  dial-options {  
    l2tp-interface-id name;  
    (dedicated | shared);  
  }  
  dialer-options {  
    activation-delay seconds;  
    deactivation-delay seconds;  
    dial-string [ dial-string-numbers ];  
    idle-timeout seconds;  
    initial-route-check seconds;  
    load-threshold number;  
    pool pool;  
    remote-name remote-callers;  
    watch-list {  
      [ routes ];  
    }  
  }  
}
```



```

disable;
dlci dlci-identifier;
drop-timeout milliseconds;
dynamic-call-admission-control {
    activation-priority priority;
    bearer-bandwidth-limit kilobits-per-second;
}
encapsulation type;
epd-threshold cells plp1 cells;
fragment-threshold bytes;
input-vlan-map {
    inner-tag-protocol-id;
    inner-vlan-id;
    (pop | pop-pop | pop-swap | push | push-push | swap | swap-push | swap-swap);
    tag-protocol-id tpid;
    vlan-id number;
}
interleave-fragments;
inverse-arp;
layer2-policer {
    input-policer policer-name;
    input-three-color policer-name;
    output-policer policer-name;
    output-three-color policer-name;
}
link-layer-overhead percent;
minimum-links number;
mrru bytes;
multicast-dlci dlci-identifier;
multicast-vci vpi-identifier.vci-identifier;
multilink-max-classes number;
multipoint;
oam-liveness {
    up-count cells;
    down-count cells;
}
oam-period (seconds | disable);
output-vlan-map {
    inner-tag-protocol-id;
    inner-vlan-id;
    (pop | pop-pop | pop-swap | push | push-push | swap | swap-swap);
    tag-protocol-id tpid;
    vlan-id number;
}
passive-monitor-mode;
peer-unit unit-number;
plp-to-clp;
point-to-point;
ppp-options {
    chap {
        access-profile name;
        default-chap-secret name;
        local-name name;
        passive;
    }
    compression {

```

```
    acfc;
    pfc;
  }
}
dynamic-profile profile-name;
pap {
  default-pap-password password;
  local-name name;
  local-password password;
  passive;
}
}
proxy-arp;
service-domain (inside | outside);
shaping {
  (cbr rate | rtvbr peak rate sustained rate burst length | vbr peak rate sustained rate
  burst length);
  queue-length number;
}
short-sequence;
transmit-weight number;
(traps | no-traps);
trunk-bandwidth rate;
trunk-id number;
tunnel {
  backup-destination address;
  destination address;
  key number;
  routing-instance {
    destination routing-instance-name;
  }
  source source-address;
  ttl number;
}
vci vpi-identifier.vci-identifier;
vlan-id number;
vlan-id-list [vlan-id vlan-id-vlan-id]
vlan-tags inner tpid.vlan-id outer tpid.vlan-id;
vlan-tags outer tpid.vlan-id inner-list [vlan-id vlan-id-vlan-id]
vpi vpi-identifier;
family family {
  accounting {
    destination-class-usage;
    source-class-usage {
      direction;
    }
  }
}
bundle interface-name;
filter {
  group filter-group-number;
  input filter-name;
  input-list {
    [ filter-names ];
  }
  output filter-name;
  output-list {
```

```

    [ filter-names ];
  }
}
ipsec-sa sa-name;
keep-address-and-control;
mtu bytes;
multicast-only;
no-redirects;
policer {
  arp policer-template-name;
  input policer-template-name;
  output policer-template-name;
}
primary;
proxy inet-address address;
receive-options-packets;
receive-ttl-exceeded;
remote (inet-address address | mac-address address);
rpf-check <fail-filter filter-name> {
  <mode loose>;
}
sampling {
  direction;
}
service {
  input {
    service-set service-set-name <service-filter filter-name>;
    post-service-filter filter-name;
  }
  output {
    service-set service-set-name <service-filter filter-name>;
  }
}
(translate-discard-eligible | no-translate-discard-eligible);
(translate-fecn-and-becn | no-translate-fecn-and-becn);
unnumbered-address interface-name destination address destination-profile
  profile-name;
address address {
  arp ip-address (mac | multicast-mac) mac-address <publish>;
  broadcast address;
  destination address;
  destination-profile name;
  eui-64;
  multipoint-destination address (dlci dlci-identifier | vci vci-identifier);
  multipoint-destination address {
    epd-threshold cells plp1 cells;
    inverse-arp;
    oam-liveness {
      up-count cells;
      down-count cells;
    }
    oam-period (seconds | disable);
    shaping {
      (cbr rate | rtvbr peak rate sustained rate burst length | vbr peak rate sustained
        rate burst length);
      queue-length number;
    }
  }
}

```

```
    }
    vci vpi-identifier.vci-identifier;
  }
  preferred;
  primary;
  (vrrp-group | vrrp-inet6-group) group-number {
    (accept-data | no-accept-data);
    advertise-interval seconds;
    authentication-type authentication;
    authentication-key key;
    fast-interval milliseconds;
    (preempt | no-preempt) {
      hold-time seconds;
    }
    priority-number number;
    track {
      priority-cost seconds;
      priority-hold-time interface-name {
        interface priority;
        bandwidth-threshold bits-per-second {
          priority;
        }
      }
    }
    route ip-address/mask routing-instance instance-name priority-cost cost;
  }
}
virtual-address [ addresses ];
}
}
}
```

PART 2

Configuring Channelized Interfaces

- Channelized Interfaces on page 27
- Configuring Channelized OC48/STM16 IQE Interfaces on page 47
- Configuring Channelized OC12/STM4 Interfaces on page 65
- Configuring Channelized OC3 IQ and IQE Interfaces on page 97
- Configuring Channelized STM1 Interfaces on page 105
- Configuring Channelized T3 Interfaces on page 123
- Configuring Channelized T1 Interfaces on page 137
- Configuring Channelized E1 Interfaces on page 143
- Configuring Channelized E1 PRI and T1 PRI Interfaces on page 151

CHAPTER 2

Channelized Interfaces

This section provides a high-level overview of channelized interfaces, focusing mainly on the capabilities, properties, and structure of channelized IQ and IQE interfaces, and includes the following topics:

- Channelized Interfaces Overview on page 27
- Channelized Interface Capabilities on page 28
- Data-Link Connection Identifiers on Channelized Interfaces on page 30
- Clock Sources on Channelized Interfaces on page 32
- Channelized E1 and T1 PIM Properties on page 35
- Channelized IQ and IQE Interfaces Properties on page 36
- Structure of Channelized IQ and Channelized IQE PICs on page 38

Channelized Interfaces Overview

Channelized interfaces enable you to configure a number of individual channels that subdivide the bandwidth of a larger interface and minimize the number of Physical Interface Cards (PICs) that an installation requires.



NOTE: Channelized intelligent queuing (IQ) and channelized enhanced intelligent queuing (IQE) interfaces require M Series Enhanced Flexible PIC Concentrators (FPCs) and MX Series Enhanced Flexible PIC Concentrators (FPCs).

Wherever Junos configuration guides refer to channelized interfaces and PICs without the “intelligent queuing IQ or IQE” descriptor, they are referring to the original channelized interfaces and PICs.

On M40e routers, all supported interface types support a maximum number of 784 traffic-bearing interfaces that can be created per interface port and includes ports on channelized PICs.

MX Series routers support two Type 2 Channelized IQ PICs: OC12/STM4 IQE PIC with SFP and OC48/STM16 IQE PIC with SFP. Each channelized OC12/STM4 PIC supports 4 ports, and the channelized OC48/STM16 PIC supports one port.

T640 and TX Matrix routers support Type 3 Channelized IQE PICs: 4xCOC12 IQE PIC with SFP.

Channelized 4xCOC12 IQE PICs support deep-channelization of up to six OC slices (STS1) per port. For example, only six OC slices can be channelized to CT1/T1 or CE1/E1.

Channelized COC48 IQE PICs support deep-channelization of up to six OC slices (STS1) in a block of 12 contiguous OC slices. For example, only six OC slices out of OC slice 1-12 can be channelized to CT1/T1 or CE1/E1. The PIC supports deep-channelization of maximum 24 OC slices in this way.

Channelized OC48 IQE PICs do not support STS-48 clear-channel mode.

IQ and IQE PICs do not support aggregated SONET (link bonding).

For channelized IQ and IQE logical interfaces, you can configure class of service (CoS). For more information, see the [Junos OS Class of Service Configuration Guide](#).

Channelized Interface Capabilities

You can configure each port of a channelized IQ PIC or channelized IQE PIC as a single interface that uses the entire available bandwidth, or partition each port into smaller data channels. In either case, you start with a channelized interface (designated by a **c** in the interface name, as in **coc12**). From the channelized interfaces, you configure data channels. Following are the channelized interface names and data-channel interface names associated with channelized IQ and IQE PICs.

Channelized Interface Names

This section lists the channelized interface names.

- **coc48-fpc/pic/port**—Channelized OC48 IQE interface. Configure on a Channelized OC48 IQE PIC.
- **coc12-fpc/pic/port**—Channelized OC12 interface. Configure on Channelized OC12 IQ or IQE PICs.
- **coc3-fpc/pic/port:channel**—Channelized OC3 interface. Configure on Channelized OC3 IQ or IQE, Channelized OC12 IQ or IQE PICs.
- **coc1-fpc/pic/port:channel**—Channelized OC1 interface. Configure on Channelized OC3 IQ or IQE, Channelized OC12 IQ or IQE, or Channelized OC48 IQE PICs.
- **ct3-fpc/pic/port:channel**—Channelized T3 interface. Configure on Channelized OC3 IQ or IQE, Channelized OC12 IQ or IQE, Channelized OC48 IQE, or Channelized DS3 IQ or IQE PICs.
- **cstm16-fpc/pic/port**—Channelized STM16 interface. Configure on a Channelized OC48 IQE PIC in SDH mode.
- **cstm4-fpc/pic/port**—Channelized STM4 interface. Configure on a Channelized OC12 IQ or IQE PIC in SDH mode.
- **cstm1-fpc/pic/port**—Channelized STM1 interface. Configure on a Channelized STM1 IQ or IQE PIC.
- **cau4-fpc/pic/port:channel**—Channelized AU-4 IQ interface. Configure on Channelized STM1 IQ or IQE, Channelized OC48 IQE, or Channelized OC12 IQE PICs.
- **ct1-fpc/pic/port:channel**—Channelized T1 interface. Configure on Channelized OC3 IQ or IQE, Channelized OC12 IQ or IQE, Channelized T1 IQ or IQE, Channelized OC48 IQE, or Channelized DS3 IQ or IQE PICs.
- **ce1-fpc/pic/port:channel**—Channelized E1 interface. Configure on Channelized E1 IQ or IQE, Channelized STM1 IQ or IQE, Channelized OC48/STM16 IQE, or Channelized OC12/STM4 IQE PICs.

Data-Channel Interface Names

This section lists the data-channel interface names.

- **e1-fpc/pic/port:channel**—E1 channel. Configure on Channelized E1 IQ or IQE, Channelized STM1 IQ or IQE, Channelized OC12/STM4 IQE, or Channelized OC48/STM16 IQE PICs.
- **e3-fpc/pic/port:channel**—E3 channel. Configure on Channelized OC3/STM1 IQE, or Channelized OC12/STM4 IQE, Channelized OC48 IQE, or Channelized/Clear channel DS3E3 IQE or E3 IQ PICs.
- **ds-fpc/pic/port:channel**—NxDS0 channel. Configure on Channelized OC3 IQ or IQE, Channelized OC12 IQ or IQE, Channelized OC48/STM16 IQE, Channelized STM1 IQ or IQE, Channelized DS3 IQ or IQE, Channelized T1 IQ, or Channelized E1 IQ or IQE PICs.
- **so-fpc/pic/port:channel**—SONET/SDH channel. Configure one OC3 channel on a Channelized OC3 IQ or IQE, four OC3 channels on a Channelized OC12 IQ or IQE, one OC12 channel on a Channelized OC12 IQ or IQE, four OC12 channels on Channelized OC48 IQE, or one STM1 channel on a Channelized STM1 IQ or IQE PICs.

- **t1-fpc/pic/port:channel**—T1 channel. Configure on Channelized T1 IQ or IQE, Channelized OC3 IQ or IQE, Channelized OC12 IQ or IQE, Channelized OC48 IQE, or Channelized DS3 IQ or IQE PICs.
- **t3-fpc/pic/port:channel**—T3 channel. Configure on Channelized OC3 IQ or IQE, Channelized OC12 IQ or IQE, Channelized OC48 IQE, Clear Channel DS3E3 IQE, or Channelized DS3 IQ or IQE PICs.

Data-Link Connection Identifiers on Channelized Interfaces

If you use Frame Relay encapsulation on a channelized interface, see Table 3 on page 30 for the maximum number of data-link connection identifiers (DLCIs) per channel that you can configure at each channel level for various channelized PICs.

If you use a per-unit-scheduler configuration on a channelized interface, see Table 4 on page 31 for the maximum number of data-link connection identifiers (DLCIs) per channel that you can configure at each channel level for various channelized PICs.



NOTE: The actual number of DLCIs you can configure for each channel is determined by the capabilities of your system, such as the number and types of PICs installed. If the number of DLCIs in the configuration exceeds the capabilities of your system, the router might not be able to support the maximum DLCI values shown in Table 3 on page 30. To determine the capabilities of your system, please contact Juniper Networks customer support.

Table 3: Frame Relay DLCI Limitations for Channelized Interfaces

PIC Types	Number of DLCIs per Level	Range
Original Channelized PICs		
DS0 level channels	3 for sparse mode	1–1022 for sparse mode (0 is reserved for the Local Management Interface [LMI])
T3 and T1 level channels	63 for regular mode	1–63 for regular mode
	3 for sparse mode	1–1022 for sparse mode (0 is reserved for the LMI)
Channelized IQ and IQE PICs		
DS0 level channels (Channelized DS3 IQ or IQE, Channelized STM1 IQ or IQE, Channelized E1 IQ or IQE, Channelized OC3 IQ or IQE, or Channelized OC12 IQ or IQE PICs)	16	1–1022 (0 is reserved for the LMI)
E1 level channels (Channelized E1 IQ or IQE PIC)	64	1–1022 (0 is reserved for the LMI)

Table 3: Frame Relay DLCI Limitations for Channelized Interfaces (*continued*)

PIC Types	Number of DLCIs per Level	Range
E1 level channels (Channelized STM1 IQ or IQE PIC)	64	1–1022 (0 is reserved for the LMI)
OC3 level channels (Channelized OC3 IQ or IQE, or Channelized OC12 IQ or IQE PIC)	1022	1–1022 (0 is reserved for the LMI)
OC12 level channels (Channelized OC12 IQ or IQE, Channelized OC48/STM16 IQE PICs, and (per port on) OC12 ports on 4xOC12/STM4 IQE PICs)	1022	1–1022 (0 is reserved for the LMI)
STM1 level channel (Channelized STM1 IQ or IQE PIC)	1022	1–1022 (0 is reserved for the LMI)
T1 level channels (Channelized DS3 IQ or IQE PIC)	64	1–1022 (0 is reserved for the LMI)
T1 level channels (Channelized OC3 IQ or IQE, or Channelized OC12 IQ or IQE PIC)	64	1–1022 (0 is reserved for the LMI)
T3 level channel (Channelized DS3 IQ or IQE, Channelized OC3 IQ or IQE, or Channelized OC12 IQ or IQE PIC)	1022	1–1022 (0 is reserved for the LMI)

Table 4: Per Unit Scheduler DLCI Limitations for Channelized Interfaces

PIC Types	Number of DLCIs per Level			
	Non M40e Platforms		M40e Platform Only	
	With Per-Unit-Scheduler	Without Per-Unit-Scheduler	With Per-Unit-Scheduler	Without Per-Unit-Scheduler
DS0 level channels	64	64	16	16
T1/E1 level channels	64	64	64	64
DS3/E3 level channels	975	† Protocol family combinations apply	256	256
SONET	975	† Protocol family combinations apply	975	† Protocol family combinations apply

† In these router, PIC, and scheduler configurations, combining multiple protocol families per PIC changes the number of Frame Relay DLCIs as shown in Table 5 on page 32.

Table 5: Protocol Family Combinations

Protocol Family Combinations	Number of DLCIs per PIC
inet	3600
inet6	3600
mpls	3000
inet, inet6	2400
inet, mpls	2000
inet6, mpls	2000
inet, inet6, mpls	1550

Clock Sources on Channelized Interfaces

Channelized interfaces and channelized IQ and IQE interfaces have different clocking capabilities. For channelized IQ and IQE interfaces, you can configure clocking on each interface independently by including the **clocking (internal | external)** statement at the **[edit interfaces interface-name]** hierarchy level.

For channelized IQ and IQE interfaces, clocking is provided as follows:

- For all channelized IQ and IQE PICs, the **clocking** statement is supported on all channels. To configure clocking on individual interfaces, include the **clocking** statement at the **[edit interfaces type-fpc/pic/port:channel]** hierarchy level. If you do not include the **clocking** statement, the individual interfaces use internal clocking by default.
- SONET/SDH-level clocking is provided at the root controller interface at the **[edit interfaces type-fpc/pic/port]** hierarchy level.
- Configure T3-level clocking by including the **clocking** statement at the **[edit interfaces ct3-fpc/pic/port]** hierarchy level.
- Configure T1-level clocking by including the **clocking** statement at the **[edit interfaces t1-fpc/pic/port:channel]** hierarchy level.
- Configure E1-level clocking by including the **clocking** statement at the **[edit interfaces ce1-fpc/pic/port]** hierarchy level.
- Configure clocking for all NxDSO channels by including the **clocking** statement at the **[edit interfaces ct1-fpc/pic/port:channel]** or **[edit interfaces ce1-fpc/pic/port]** hierarchy level.

- The **clocking** statement is ignored if you include it at the **[edit interfaces coc1-fpc/pic/port:channel]** or **[edit interfaces cau4-fpc/pic/port:channel]** hierarchy level.
- SONET/SDH level clocking is applicable only at the controller interfaces for channelized IQ and IQE PICs. Clocking configuration is not effective at the **so-fpc/pic/port** or **so-fpc/pic/port:channel** for channelized IQ and IQE PICs.

For non-IQ and non-IQE channelized interfaces, clocking at each channel level is provided as follows:

- For Channelized OC12, DS3, and E1 PICs, the **clocking** statement is supported only for channel 0; it is ignored if included in the configuration of other channels. The clock source configured for channel 0 applies to all channels on these channelized interfaces.
- For the Channelized STM1 PIC, the **clocking** statement is supported on channels 0 through 62. To configure clocking on the STM1 interface, include the **loop-timing** statement at the **[edit interfaces e1-fpc/pic/port:0 sonet-options]** hierarchy level. To configure clocking on individual E1 interfaces, include the **clocking** statement at the **[edit interfaces e1-fpc/pic/port:channel]** hierarchy level. The channel number can be 0 through 62. If you do not include the **clocking** statement, the individual E1 interfaces use internal clocking by default.
- For channelized STM1 interfaces, you should configure the clock source at one side of the connection to be internal and configure the other side of the connection to be external.
- When you configure the clock source for a channelized interface—**t3-fpc/pic/port:0**, for example—you must also include the **channel-group** statement at the **[edit chassis]** hierarchy level, and specify channel group 0.

Table 6 on page 33 lists the clocking capabilities for each channelized PIC.

Table 6: Clocking Capabilities by Channelized PIC Type

PIC Type	SONET/SDH Level	DS3 Level	DS1/E1 Level
Channelized PICs			
Channelized DS3 and Multichannel DS3	Not applicable.	The loop-timing statement is supported at the [edit interfaces t1-fpc/pic/port:0 t3-options] or [edit interfaces fpc/pic/port:0:0 t3-options] hierarchy level.	The clocking statement is supported at the [edit interfaces t1-fpc/pic/port:0] or [edit interfaces ds-fpc/pic/port:0:0] hierarchy level.
Channelized E1	Not applicable.	Not applicable.	The clocking statement is supported at the [edit interfaces e1-fpc/pic/port:0] or [edit interfaces ds-fpc/pic/port:0] hierarchy level.
Channelized OC12	Not configurable.	The clocking statement is supported at the [edit interfaces t3-fpc/pic/port:0] hierarchy level.	Not applicable.

Table 6: Clocking Capabilities by Channelized PIC Type (*continued*)

PIC Type	SONET/SDH Level	DS3 Level	DS1/E1 Level
Channelized STM1	Not configurable.	Not applicable.	The clocking statement is supported at the [edit interfaces e1-fpc/pic/port:[0-62]] hierarchy level.
Channelized IQ and IQE PICs			
Channelized DS3 IQ or IQE	Not applicable.	<p>The clocking statement is supported at the [edit interfaces ct3-fpc/pic/port] hierarchy level.</p> <p>The clocking statement is ignored if you include it at the [edit interfaces t3-fpc/pic/port] hierarchy level.</p>	<p>For T1 channels, the clocking statement is supported at the [edit interfaces t1-fpc/pic/port:[1-28]] hierarchy level.</p> <p>For NxDS0 channels, the clocking statement is supported at the [edit interfaces ct1-fpc/pic/port:[1-28]] hierarchy level.</p>
Channelized E1 IQ	Not applicable.	Not applicable.	<p>For E1 and NxDS0 channels, the clocking statement is supported at the [edit interfaces ce1-fpc/pic/port] hierarchy level.</p> <p>The clocking statement is ignored if you include it at the [edit interfaces e1-fpc/pic/port] hierarchy level.</p>
Channelized OC3 IQ or IQE	<p>The clocking statement is supported at the [edit interfaces coc3-fpc/pic/port] hierarchy level.</p> <p>The clocking statement is ignored if you include it at the [edit interfaces so-fpc/pic/port] hierarchy level.</p>	<p>The clocking statement is supported at the [edit interfaces t3-fpc/pic/port:[1-12]] hierarchy level.</p> <p>The clocking statement is ignored if you include it at the [edit interfaces coc1-fpc/pic/port:channel] hierarchy level.</p>	<p>The clocking statement is supported at the [edit interfaces ct1-fpc/pic/port:[1-12]:[1-28]] and [edit interfaces t1-fpc/pic/port:[1-12]:[1-28]] hierarchy levels.</p>
Channelized OC12 IQ or IQE	<p>The clocking statement is supported at the [edit interfaces coc12-fpc/pic/port] hierarchy level.</p> <p>The clocking statement is ignored if you include it at the [edit interfaces so-fpc/pic/port] hierarchy level.</p>	<p>The clocking statement is supported at the [edit interfaces t3-fpc/pic/port:[1-12]] hierarchy level.</p> <p>The clocking statement is ignored if you include it at the [edit interfaces coc1-fpc/pic/port:channel] hierarchy level.</p>	<p>The clocking statement is supported at the [edit interfaces ct1-fpc/pic/port:[1-12]:[1-28]] and [edit interfaces t1-fpc/pic/port:[1-12]:[1-28]] hierarchy levels.</p>

Table 6: Clocking Capabilities by Channelized PIC Type (*continued*)

PIC Type	SONET/SDH Level	DS3 Level	DS1/E1 Level
Channelized OC48 IQE	<p>The clocking statement is supported at the [edit interfaces coc48-fpc/pic/port] hierarchy level.</p> <p>The clocking statement is ignored if you include it at the [edit interfaces so-fpc/pic/port] hierarchy level.</p>	<p>The clocking statement is supported at the [edit interfaces t3-fpc/pic/port:[1-48]] hierarchy level.</p> <p>The clocking statement is ignored if you include it at the [edit interfaces coc1-fpc/pic/port:channel] hierarchy level.</p>	<p>The clocking statement is supported at the [edit interfaces ct1fpc/pic/port:[1-48]:[1-28]] and [edit interfaces t1-fpc/pic/port:[1-48]:[1-28]] hierarchy levels.</p>
Channelized STM1 IQ or IQE	<p>The clocking statement is supported at the [edit interfaces cstm1-fpc/pic/port] hierarchy level.</p> <p>The clocking statement is ignored if you include it at the [edit interfaces cau4-fpc/pic/port:channel] or [edit interfaces so-fpc/pic/port] hierarchy level.</p>	Not applicable.	<p>For E1 and NxDS0 channels, the clocking statement is supported at the [edit interfaces ce1-fpc/pic/port[1-63]] hierarchy level.</p> <p>The clocking statement is ignored if you include it at the [edit interfaces e1-fpc/pic/port] hierarchy level.</p>
Channelized STM4 IQ or IQE	<p>The clocking statement is supported at the [edit interfaces cstm4-fpc/pic/port] hierarchy level.</p> <p>The clocking statement is ignored if you include it at the [edit interfaces cau4-fpc/pic/port:channel] or [edit interfaces so-fpc/pic/port] hierarchy level.</p>	Not applicable.	<p>For E1 and NxDS0 channels, the clocking statement is supported at the [edit interfaces ce1-fpc/pic/port[1-4]:[1-63]] hierarchy level.</p> <p>The clocking statement is ignored if you include it at the [edit interfaces e1-fpc/pic/port] hierarchy level.</p>
Channelized STM16 IQE	<p>The clocking statement is supported at the [edit interfaces cstm16-fpc/pic/port] hierarchy level.</p> <p>The clocking statement is ignored if you include it at the [edit interfaces cau4-fpc/pic/port:channel] or [edit interfaces so-fpc/pic/port] hierarchy level.</p>	Not applicable.	<p>For E1 and NxDS0 channels, the clocking statement is supported at the [edit interfaces ce1-fpc/pic/port[1-16]:[1-63]] hierarchy level.</p> <p>The clocking statement is ignored if you include it at the [edit interfaces e1-fpc/pic/port] hierarchy level.</p>

Channelized E1 and T1 PIM Properties

Channelized E1 and T1 PIMs on J Series routers provide support for ISDN Primary Rate Interface (PRI) connectivity for dial-in and callback and for use as primary or backup network connections. You can configure up to 30 channelized E1 time slots (**ce1-pim/0/port**) or 23 channelized T1 time slots (**ct1-pim/0/port**) as an ISDN PRI group,

with the 16th E1 time slot or the 24th T1 time slot operating as the D-channel to control the group of time slots as B-channels. These B-channels can operate unconfigured. The encapsulation type **multilink-ppp**, **cisco-hdlc**, or **ppp** is configured under the dialer interface.

For more information about configuring the dialer interface, see *Configuring ISDN Logical Interface Properties*.

E1 and T1 time slots unused by ISDN PRI can operate normally as DS0 interfaces. PRI B-channels run at 64 Kbps, but do not support the 56-Kbps line rate.

For more information about Channelized E1 PIMs, ISDN PRI connectivity, and the ISDN features they support, see the *Junos OS Interfaces and Routing Configuration Guide*.

Channelized IQ and IQE Interfaces Properties

On channelized IQ and IQE interfaces, you can specify options that are globally applied to all interface types associated with channelized IQ and IQE interfaces. For example, **e1-options** statements that you include at the **[edit interfaces ce1-fpc/pic/port]** hierarchy level apply globally to all E1 and NxDS0 interfaces that you create by partitioning **ce1-fpc/pic/port**. Likewise, **t3-options** statements that you include at the **[edit interfaces ct3-fpc/pic/port]** hierarchy level apply globally to all T1 and NxDS0 interfaces that you create by partitioning **ct3-fpc/pic/port**.

You can also apply interface options at the channel level. For example, you can include **t1-options** statements at the **[edit interfaces t1-fpc/pic/port <:channel>]** hierarchy level, and **ds0-options** statements at the **[edit interfaces ds-0/1/1<:channel>]** hierarchy level.

Only a subset of the interface options is valid on each type of channelized IQ interface. You configure all HDLC information at the end-data channel level, not at the parent level. For example, configure HDLC information at the **[edit interfaces ds-fpc/pic/port<:channel>]** hierarchy level, not at the **[edit interfaces ct1-fpc/pic/port<:channel>]** hierarchy level.

Automatic Protection Switching (APS) is supported on channelized OC3, OC12, STM1, and STM4 IQ interfaces. To configure APS, include the **aps** statement with options at the **[edit interfaces interface-name sonet-options]** hierarchy level. For information about configuring APS, see *Configuring SONET/SDH Physical Interface Properties*.

In interchassis and intrachassis redundant LSQ configurations that use MLPPP and SONET APS, you can inhibit a router from sending PPP termination-request messages to the remote host if the link PIC fails. To inhibit the router from sending PPP termination-request messages to the remote host if the link PIC fails, include the **no-termination-request** statement at the **[edit interfaces interface-name ppp-options]** hierarchy level.

The **no-termination-request** statement is supported only with MLPPP and SONET APS configurations and works with PPP, PPP over Frame Relay, and MLPPP interfaces only. The supported PIC types are as follows:

- Channelized OC48/STM16 IQE PICs
- Channelized OC12/STM4 IQ and IQE PICs

- Channelized OC3 IQ and IQE PICs
- Channelized STM1 IQ and IQE PICs

Channelized IQ and IQE interfaces do not support receive buckets or transmit buckets.

For channelized IQ and IQE interfaces, there are some limitations on where you place certain statements in the configuration. When you configure clocking, bit error rate testing (BERT), C-bit parity, and loopback statements on T3, T1, or DS0 channels, you must follow these guidelines:

- For T3 IQ interfaces, you can include the **loopback payload** statement at the **[edit interfaces ct3-fpc/pic/port]** and **[edit interfaces t3-fpc/pic/port:channel]** hierarchy levels. For T1 interfaces, you can include the **loopback payload** statement at the **[edit interfaces t1-fpc/pic/port:channel]** hierarchy level; it is ignored if included at the **[edit interfaces ct1-fpc/pic/port]** hierarchy level. For NxDS0 interfaces, payload and remote loopback are the same. If you configure one, the other is ignored. NxDS0 IQ interfaces do not support local loopback.
- If you include clocking, BERT, and C-bit parity configurations at both the **[edit interfaces ct3-fpc/pic/port<:channel> t3-options]** and **[edit interfaces t3-fpc/pic/port<:channel> t3-options]** hierarchy levels, the channelized T3-level statements are valid, and the T3-level statements are ignored.
- If you include clocking, BERT, and C-bit parity configurations at both the **[edit interfaces ct3-fpc/pic/port<:channel> t3-options]** and **[edit interfaces t1-fpc/pic/port<:channel> t1-options]** hierarchy levels, the channelized T3-level statements are operational for the T3 connections and the T1-level statements are operational for the T1 connections.
- Because DS0 channels do not have clocking capability, you must configure clocking at the **[edit interfaces ct1-fpc/pic/port<:channel> t1-options]** or **[edit interfaces ce1-fpc/pic/port<:channel> e1-options]** hierarchy level for channelized NxDS0 IQ interfaces.
- You can set BERT at the **[edit interfaces t3-fpc/pic/port<:channel> t3-options]** hierarchy level or on any partitioned channel of the channelized T3 interface. There are 12 BERT patterns available for NxDS0 channels and 28 BERT patterns for T1, channelized T1, T3, and channelized T3 interfaces within channelized IQ interfaces.
- For channelized IQ and IQE PICs, SONET/SDH level, use the **sonet-options loopback** statement **local** and **remote** options at the controller interface (coc48, cstm16, coc12, cstm4, coc3, cstm1). It is ignored for path-level interfaces **so-fpc/pic/port** or **so-fpc/pic/port:channel**.
- For channelized interfaces that use Frame Relay encapsulation, the number of configurable DLCIs varies by channelized interface type.
- For channelized interfaces, you can configure class of service (CoS) on channels, but not at the controller level.
- For original Channelized OC12 PICs, limited CoS functionality is supported. For more information, contact Juniper Networks customer support.
- CoS is not configurable on controller interfaces.

Structure of Channelized IQ and Channelized IQE PICs

Figure 1 on page 38 through Figure 13 on page 43 show the structural organization of the channelized PICs, channelized IQ PICs, and channelized IQE PICs. Table 7 on page 44 through Table 9 on page 46 show the structure of channelized IQE PICs, channelized IQ PICs, and channelized PICs.

Figure 1: Channelized OC48/STM16 IQE PIC (in SONET Mode)

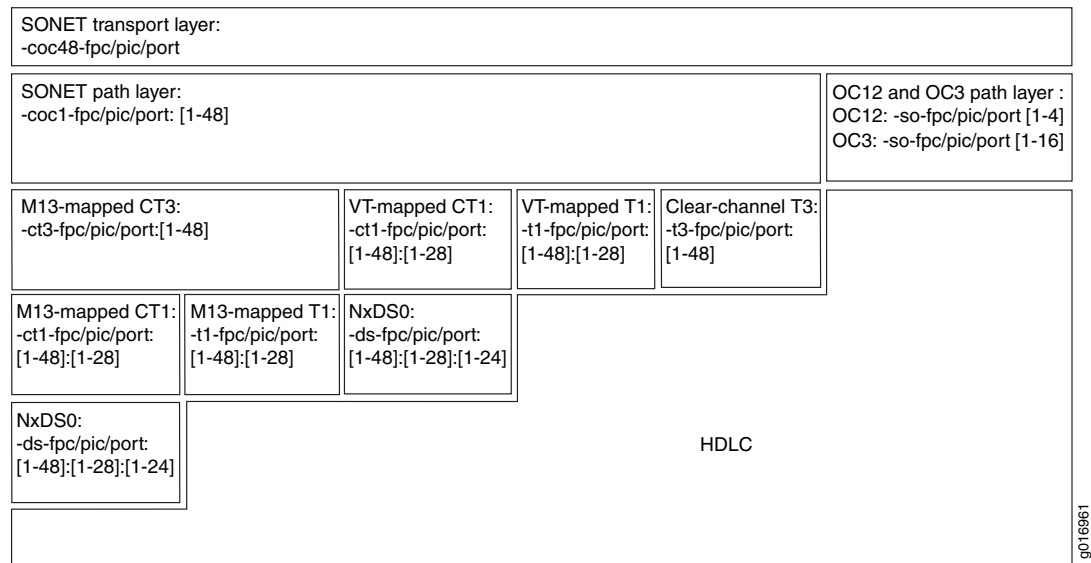


Figure 2: Channelized OC48/STM16 IQE PIC (in SDH Mode)

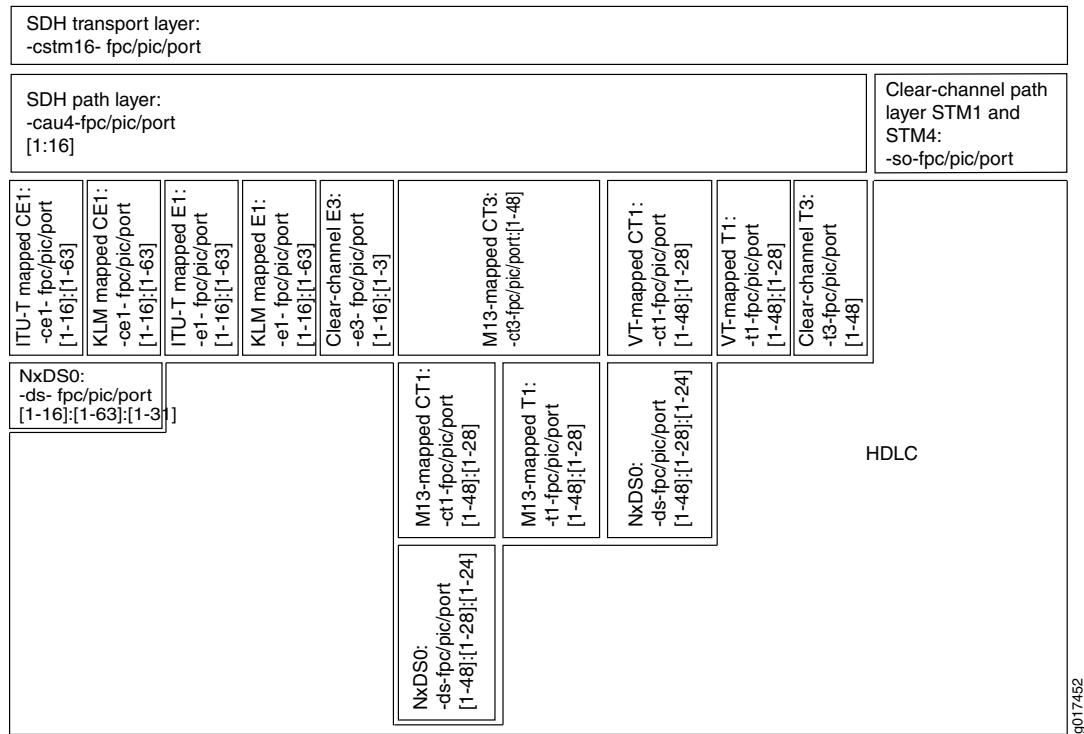


Figure 3: Channelized OC12 IQ PIC and Channelized OC12/STM4 IQE PIC (in SONET Mode)

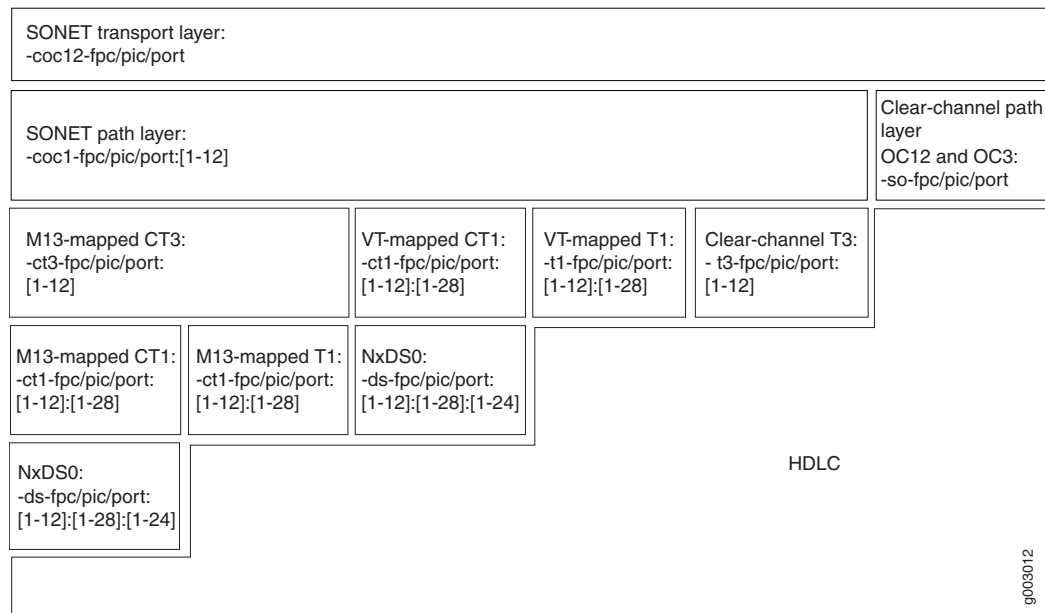


Figure 4: Channelized OC12/STM4 IQE PIC (in SDH Mode)

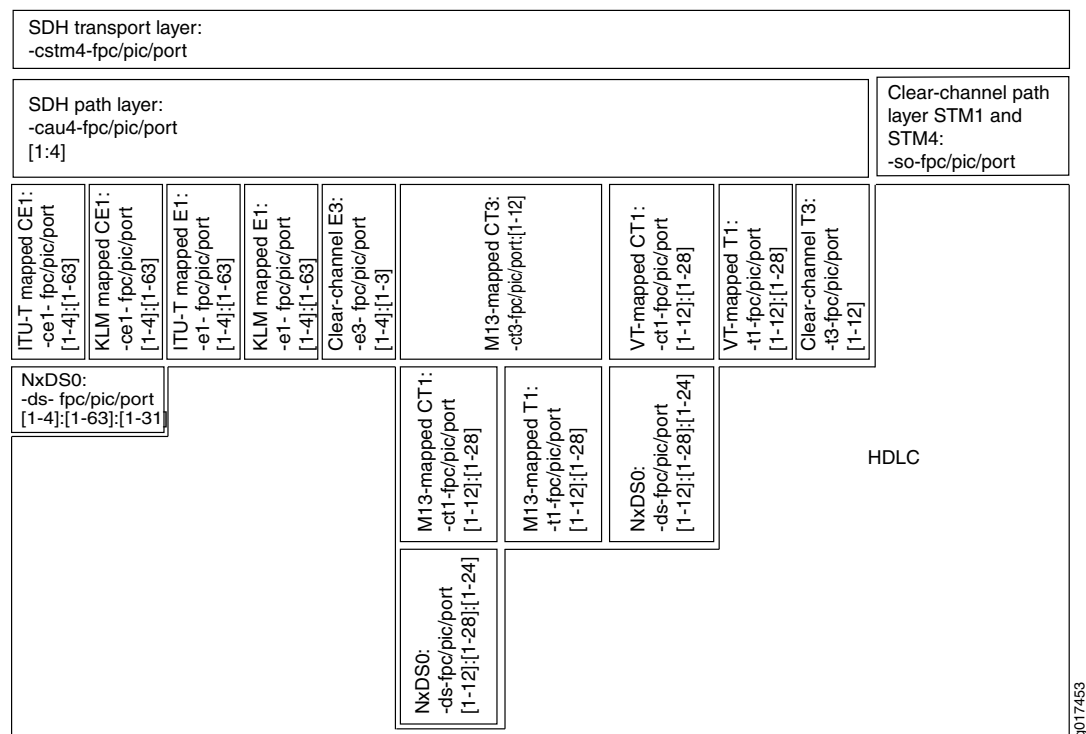


Figure 5: Channelized OC12/STM4 IQ PIC (in SDH Mode)

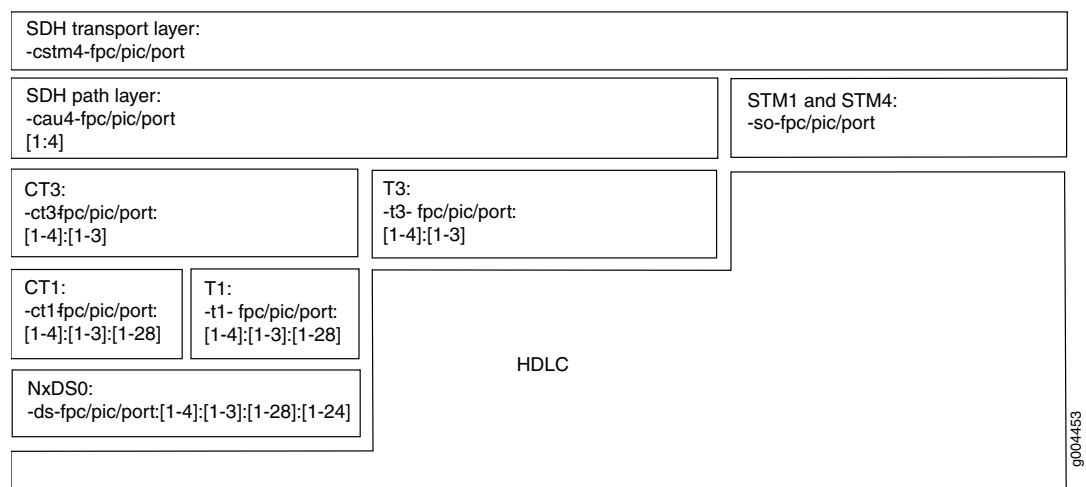


Figure 6: Channelized OC3 Ports (in SONET Mode) on Channelized OC3 IQ and Channelized OC3/STM1 IQE PICs

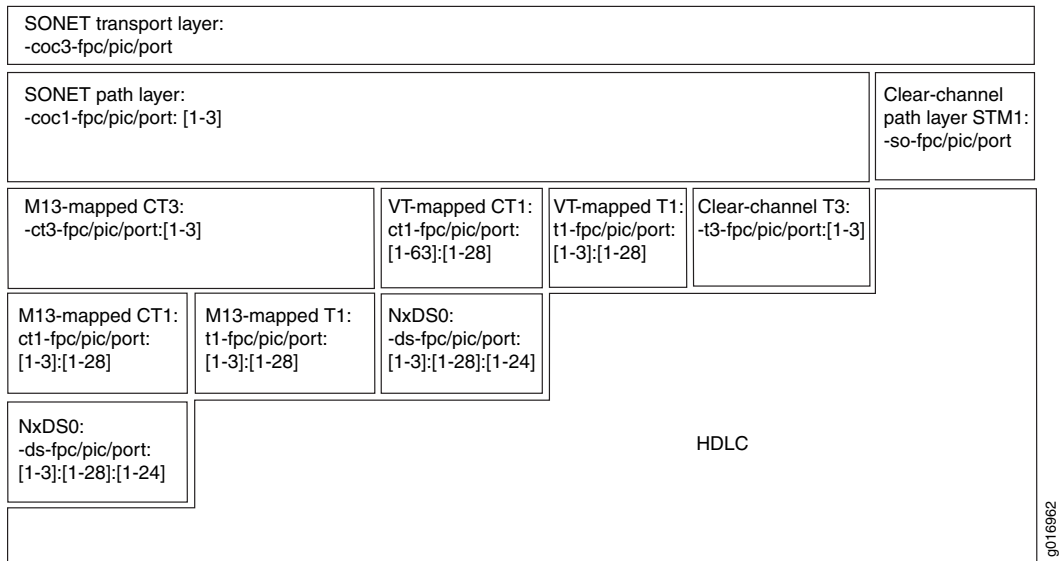


Figure 7: Channelized CSTM1 Ports (in SDH Mode) on Channelized OC3/STM1 IQE PIC

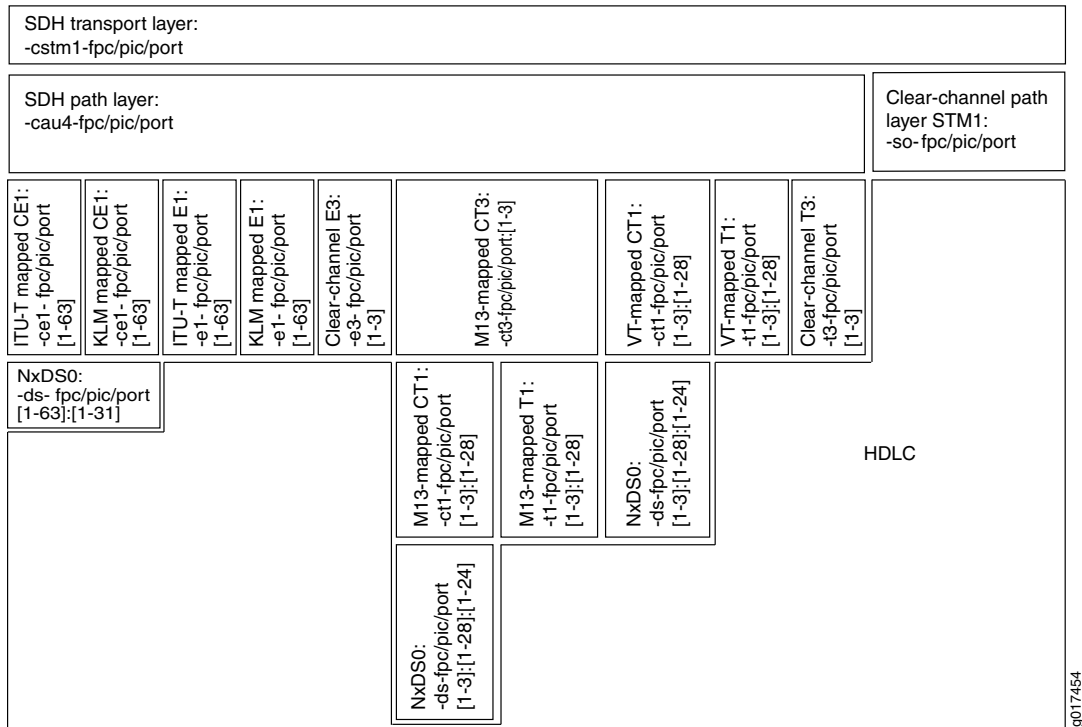


Figure 8: Channelized STM1 IQ PIC

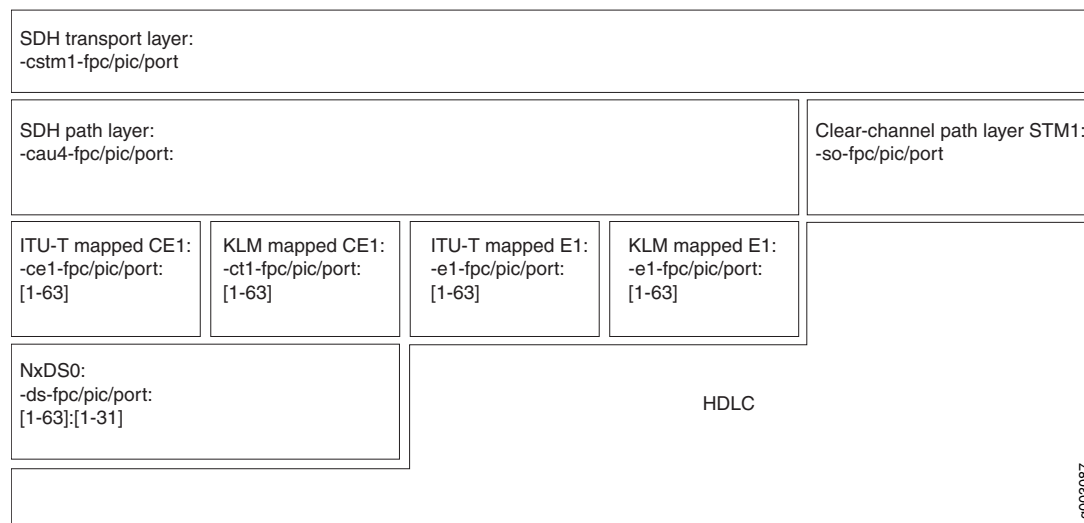


Figure 9: Channelized CDS3/E3 IQE PIC (in DS3 Mode)

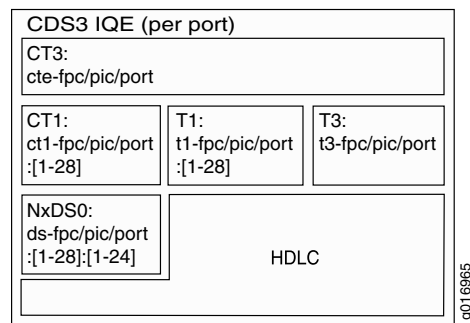


Figure 10: Channelized CDS3/E3 IQE PIC (in E3 Mode)

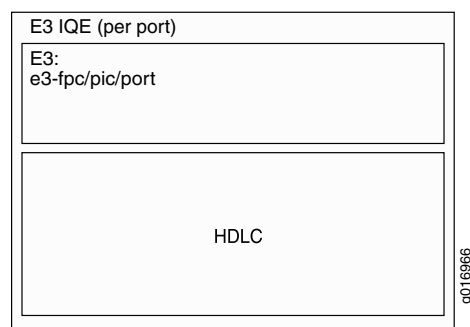


Figure 11: Channelized DS3 IQ PIC

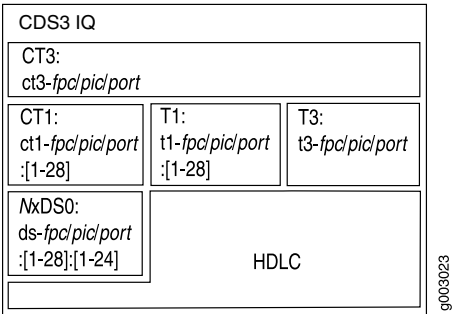


Figure 12: Channelized T1 IQ and IQE PIC

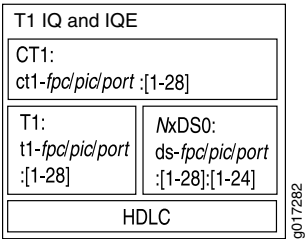


Figure 13: Channelized E1 IQ and IQE PIC

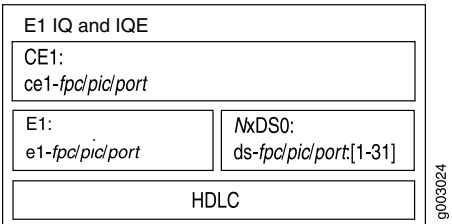


Table 7: Structural Differences: Channelized IQE PICs

PIC Type	Transport	Path	DS3	DS1/E1	E3
Channelized IQE PICs					
Channelized OC48/STM16 IQE (SONET Mode)	<i>coc48-fpc/pic/port</i>	<i>coc1-fpc/pic/port</i> :[1-48]	<i>ct3-fpc/pic/port</i> :[1-48]	<i>ct1-fpc/pic/port</i> :[1-48]:[1-28]	Not applicable.
		<i>so-fpc/pic/port</i>	<i>t3-fpc/pic/port</i> :[1-48]	<i>t1-fpc/pic/port</i> :[1-48]:[1-28]	
Channelized OC48/STM16 IQE (SDH Mode)	<i>cstm16-fpc/pic/port</i>	<i>cau4-fpc/pic/port</i> :[1-16]	<i>ct3-fpc/pic/port</i> :[1:16]:[1:3]	<i>ce1-fpc/pic/port</i> :[1-16]:[1-63]	<i>e3-fpc/pic/port</i> :[1-16]:[1-3]
		<i>so-fpc/pic/port</i>	<i>t3-fpc/pic/port</i> :[1:16]:[1:3]	<i>e1-fpc/pic/port</i> :[1-16]:[1-63]	
			<i>ct1-fpc/pic/port</i> :[1:16]:[1-84]	<i>ct1-fpc/pic/port</i> :[1:16]:[1-3]:[1-28]	
			<i>t1-fpc/pic/port</i> :[1:16]:[1-84]	<i>t1-fpc/pic/port</i> :[1:16]:[1-3]:[1-28]	
Channelized OC12 IQE (SONET Mode)	<i>coc12-fpc/pic/port</i>	<i>coc1-fpc/pic/port</i> :[1-12]	<i>ct3-fpc/pic/port</i> :[1-12]	<i>ct1-fpc/pic/port</i> :[1-12]:[1-28]	Not applicable.
		<i>so-fpc/pic/port</i>	<i>t3-fpc/pic/port</i> :[1-12]	<i>t1-fpc/pic/port</i> :[1-12]:[1-28]	
Channelized STM4 IQE (SDH Mode)	<i>cstm4-fpc/pic/port</i>	<i>cau4-fpc/pic/port</i> : [1-4]	<i>ct3-fpc/pic/port</i> :[1-4]:[1-3]	<i>ce1-fpc/pic/port</i> :[1-4]:[1-63]	<i>e3-fpc/pic/port</i> :[1-4]:[1-3]
		<i>so-fpc/pic/port</i>	<i>t3-fpc/pic/port</i> :[1-4]:[1-3]	<i>e1-fpc/pic/port</i> :[1-4]:[1-63]	
			<i>ct1-fpc/pic/port</i> :[1-4]:[1-84]	<i>ct1-fpc/pic/port</i> :[1-4]:[1-3]:[1-28]	
			<i>t1-fpc/pic/port</i> :[1-4]:[1-84]	<i>t1-fpc/pic/port</i> :[1-4]:[1-3]:[1-28]	
Channelized OC3 IQE (SONET)	<i>coc3-fpc/pic/port</i>	<i>coc1-fpc/pic/port</i> :[1-3]	<i>ct3-fpc/pic/port</i> :[1-3]	<i>ct1-fpc/pic/port</i> :[1-3]:[1-28]	Not applicable.
		<i>so-fpc/pic/port</i>	<i>t3-fpc/pic/port</i> :[1-3]	<i>t1-fpc/pic/port</i> :[1-3]:[1-28]	
Channelized STM1 IQE	<i>cstm1-fpc/pic/port</i>	<i>cau4-fpc/pic/port</i>	<i>ct3-fpc/pic/port</i> :[1-3]	<i>ce1-fpc/pic/port</i> :[1-63]	<i>e3-fpc/pic/port</i> :[1:3]]
		<i>so-fpc/pic/port</i>	<i>t3-fpc/pic/port</i> :[1-3]	<i>e1-fpc/pic/port</i> :[1-63]	
			<i>ct1-fpc/pic/port</i> :[1-84]	<i>ct1-fpc/pic/port</i> :[1-3]:[1-28]	
			<i>t1-fpc/pic/port</i> :[1-84]	<i>t1-fpc/pic/port</i> :[1-3]:[1-28]	

Table 7: Structural Differences: Channelized IQE PICs (*continued*)

PIC Type	Transport	Path	DS3	DS1/E1	E3
Channelized DS3 IQE	Not applicable.	Not applicable.	<i>ct3-fpc/pic/port</i> <i>t3-fpc/pic/port</i>	<i>ct1-fpc/pic/port</i> :[1-28] <i>t1-fpc/pic/port</i> :[1-28]	Not applicable.
Channelized E3 IQE	Not applicable.	Not applicable.	Not applicable.	Not applicable.	<i>e3-fpc/pic/port</i> :[1:4]
Channelized T1 IQE	Not applicable.	Not applicable.	Not applicable.	<i>ct1-fpc/pic/port</i> <i>t1-fpc/pic/port</i>	Not applicable.
Channelized E1 IQE	Not applicable.	Not applicable.	Not applicable.	<i>ce1-fpc/pic/port</i> <i>e1-fpc/pic/port</i>	Not applicable.

Table 8: Structural Differences: Channelized IQ PICs

PIC Type	Transport	Path	DS3	DS1/E1	E3
Channelized IQ PICs					
Channelized OC12/STM4 IQ (SONET Mode)	<i>coc12-fpc/pic/port</i>	<i>coc1-fpc/pic/port</i> :[1-12] <i>so-fpc/pic/port</i>	<i>ct3-fpc/pic/port</i> :[1-4]:[1-3] <i>t3-fpc/pic/port</i> :[1-4]:[1-3]	<i>ct1-fpc/pic/port</i> :[1-3]:[1-28] <i>t1-fpc/pic/port</i> :[1-4]:[1-3]:[1-28]	Not applicable.
Channelized OC12/STM4 IQ (SDH Mode)	<i>cstm4-fpc/pic/port</i>	<i>cau4-fpc/pic/port</i> : [1-4] <i>so-fpc/pic/port</i>	<i>ct3-fpc/pic/port</i> :[1-4]:[1-3] <i>t3-fpc/pic/port</i> :[1-4]:[1-3]	<i>ct1-fpc/pic/port</i> :[1-3]:[1-28] <i>t1-fpc/pic/port</i> :[1-4]:[1-3]:[1-28]	Not applicable.
Channelized OC3 IQ (SONET)	<i>coc3-fpc/pic/port</i>	<i>coc1-fpc/pic/port</i> :[1-3] <i>so-fpc/pic/port</i>	<i>ct3-fpc/pic/port</i> :[1-3] <i>t3-fpc/pic/port</i> :[1-3]	<i>ct1-fpc/pic/port</i> :[1-3]:[1-28] <i>t1-fpc/pic/port</i> :[1-3]:[1-28]	Not applicable.
Channelized STM1 IQ (SDH)	Not applicable.	<i>cau4-fpc/pic/port</i> <i>so-fpc/pic/port</i>	Not applicable.	<i>ce1-fpc/pic/port</i> :[1-63] <i>e1-fpc/pic/port</i> :[1-63]	Not applicable.
Channelized DS3 IQ	Not applicable.	Not applicable.	<i>ct3-fpc/pic/port</i> <i>t3-fpc/pic/port</i>	<i>ct1-fpc/pic/port</i> :[1-28] <i>t1-fpc/pic/port</i> :[1-28]	Not applicable.

Table 8: Structural Differences: Channelized IQ PICs (*continued*)

PIC Type	Transport	Path	DS3	DS1/E1	E3
Channelized E1 IQ	Not applicable.	Not applicable.	Not applicable.	<i>ce1-fpc/pic/port</i> <i>e1-fpc/pic/port</i>	Not applicable.

Table 9: Structural Differences: Channelized PICs

PIC Type	Transport	Path	DS3	DS1/E1	E3
Channelized PICs					
Channelized OC12	<i>t3-fpc/pic/port</i> :0	<i>t3-fpc/pic/port</i> :[0-11]	<i>t3-fpc/pic/port</i> :[0-11]	Not applicable.	Not applicable.
Channelized STM1	<i>e1-fpc/pic/port</i> :0	<i>e1-fpc/pic/port</i> :0	Not applicable.	<i>e1-fpc/pic/port</i> :[0-63]	Not applicable.
Channelized T3 and Multichannel T3	Not applicable.	Not applicable.	<i>t1-fpc/pic/port</i> :0	<i>t1-fpc/pic/port</i> :[0-27]	Not applicable.
Channelized E1	Not applicable.	Not applicable.	Not applicable.	<i>e1-fpc/pic/port</i> <i>ds-fpc/pic/port</i> :0	Not applicable.

CHAPTER 3

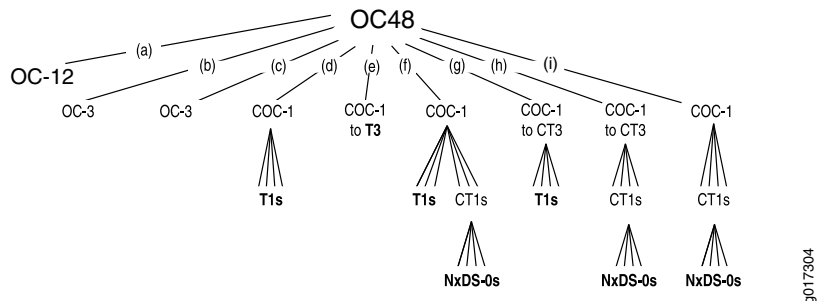
Configuring Channelized OC48/STM16 IQE Interfaces

- Channelized OC48/STM16 IQE Interfaces Overview on page 47
- Configuring Channelized OC48/STM16 IQE Interfaces in SONET Mode on page 49
- Configuring Channelized OC48/STM16 IQE Interfaces (SDH Mode) on page 56
- Configuring Link PIC Failover on Channelized OC48/STM16 IQE Interfaces on page 62
- Example: Configuring Channelized OC48 Interfaces with Partitioned Channels on page 62

Channelized OC48/STM16 IQE Interfaces Overview

Channelized enhanced intelligent queuing (IQE) interfaces allow arbitrary and dynamic channelization of serial links, allowing greater flexibility than the channelized interfaces. Figure 14 on page 47, Figure 15 on page 48, and Figure 16 on page 49 illustrate the Channelized OC48/STM16 IQE Physical Interface Cards (PICs) in several examples of many possible configurations.

Figure 14: Sample Channelization of OC48/STM16 IQE PIC (SONET Mode)



Bold entries correspond to actual packet channels.

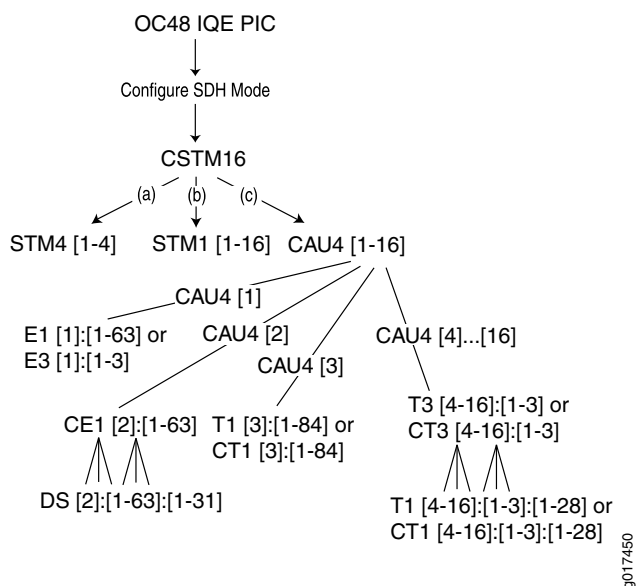
In the example in Figure 14 on page 47, a Channelized OC48/STM16 IQE PIC operating in SONET mode is partitioned into the following OC slices:

- a. A clear channel OC12 interface.
- b. An OC3 interface.
- c. A channelized COC1 partitioned into T1 interfaces.

- d. A channelized COC1 partitioned into a T3 interface.
- e. A channelized COC1 partitioned into CT3, partitioned into T1 interfaces, and CT1s partitioned into NxDS0 interfaces.
- f. A channelized COC1 partitioned into CT3, partitioned into T1 interfaces.
- g. A channelized COC1 partitioned into CT3, partitioned into CT1s, partitioned into NxDS0 interfaces.
- h. A channelized COC1 partitioned into CT1s, partitioned into NxDS0 interfaces.

This is one of thousands of ways to configure a Channelized OC48/STM16 IQE PIC. To configure the interfaces shown in Figure 15 on page 48, see “Configuring Channelized OC48/STM16 IQE Interfaces (SDH Mode)” on page 56.

Figure 15: Sample Channelization of OC48/STM16 IQE PIC (SDH Mode)



In Figure 15 on page 48, a Channelized OC48/STM16 IQE PIC operating in SDH mode results in a channelized STM16 interface, which can be partitioned as the following:

- a. Up to 4 STM4s.
- b. Up to 16 STM1s.
- c. Up to 16 CAU4s that can each be partitioned into up to 63 E1s, up to 3 E3s, or up to 63 CE1s. Up to 16 CAU4s that can each be partitioned into up to 84 T1s, 84 CT1s, 63 E1s, 63 CE1s, 3 E3s, 3 T3s or 3 CT3s. Each CE1 can be partitioned into up to 31 NxDS0s. Each CT1 can be partitioned into up to 24 NxDS0s. Each CT3 can be partitioned into up to 28 T1s or 28 CT1s.

This is one of thousands of ways to configure a Channelized OC48/STM16 IQE PIC.

Figure 16: Sample Channelization of OC48/STM16 IQE PIC to E3 Channels

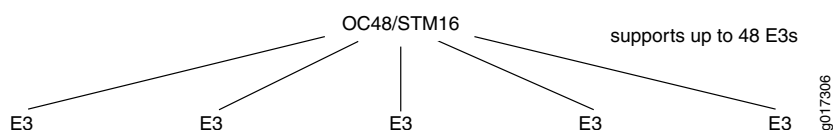


Figure 16 on page 49 shows five E3 channels configured on the Channelized OC48/STM16 IQE PIC. You can configure 43 additional E3 channels. For more information about configuring E3 channels on Channelized OC48/STM16 IQE PICs, see “Configuring E3 Interfaces” on page 58.

For a full configuration example, see the *Junos OS Feature Guides*.

Configuring Channelized OC48/STM16 IQE Interfaces in SONET Mode

- Configuring OC12 Interfaces on page 49
- Configuring OC3 Interfaces on page 50
- Configuring T3 Interfaces on page 51
- Configuring T1 Interfaces on page 52
- Configuring Fractional T1 Interfaces on page 54
- Configuring NxDS0 Interfaces on page 54

Configuring OC12 Interfaces

You can configure up to four OC12 interfaces on a 1-port Channelized OC48/STM16 IQE PIC. To configure an OC12 interface, include the **partition**, **oc-slice**, and **interface-type** statements at the `[edit interfaces coc48-fpc/pic/port]` hierarchy level, specifying the **so** interface type:

```
[edit interfaces coc48-fpc/pic/port]
partition partition-number oc-slice oc-slice-range interface-type so;
```

The partition number is the sublevel interface partition index. For SONET/SDH interfaces, the partition number does not correlate with bandwidth size. For OC12 interfaces, the partition number can be from 1 through 4.



NOTE: For channelized OC48 IQE interfaces, channel numbering begins with 1 (:1).

The OC-slice range is the range of SONET/SDH slices. For SONET/SDH interfaces, the OC-slice range specifies the bandwidth size required for the interface type you are configuring. OC12 interfaces must occupy 12 consecutive OC slices per interface, in one of the following forms:

- 1–12
- 13–24

- 25–26
- 37–48

By contrast, the T3 and OC1 interfaces each occupy one OC slice per interface and OC3 interfaces occupy three slices per interface.

The interface type is the channelized interface type or data channel you are creating. For channelized OC48 IQE interfaces, the interface type can be **so**.

Example: Configuring OC12 Interfaces

Configure an OC12 interface, using partition 1 and OC slices 1 through 12. This configuration creates interface **so-1/1/0:1**.

```
[edit interfaces coc48-1/1/0]
partition 1 oc-slice 1-12 interface-type so;
```

For a full configuration example, see the *Junos OS Feature Guides*.

Configuring OC3 Interfaces

To configure an OC3 interface, include the **partition**, **oc-slice**, and **interface-type** statements at the **[edit interfaces coc48-fpc/pic/port]** hierarchy level, specifying the **so** interface type:

```
[edit interfaces coc48-fpc/pic/port]
partition partition-number oc-slice oc-slice-range interface-type so;
```

The partition number is the sublevel interface partition index. For SONET/SDH interfaces, the partition number does not correlate with bandwidth size. For OC3 interfaces, the partition number can be from 1 through 16.



NOTE: For channelized OC48 IQE interfaces, channel numbering begins with 1 (:1).

The OC-slice range is the range of SONET/SDH slices. For SONET/SDH interfaces, the OC-slice range specifies the bandwidth size required for the interface type you are configuring. OC3 interfaces must occupy three consecutive OC slices per interface, in one of the following forms:

- 1–3
- 4–6
- 7–9
- 10–12
- and so on (in groups of 3), up to 48

By contrast, the T3 and OC1 interfaces each occupy one OC slice per interface.

The interface type is the channelized interface type or data channel you are creating. For channelized OC48 IQE interfaces, the interface type can be **so**.

Example: Configuring OC3 Interfaces

Configure an OC3 interface, using partition 1 and OC slices 4 through 6. This configuration creates interface **so-1/1/0:1**.

```
[edit interfaces coc48-1/1/0]
partition 1 oc-slice 4-6 interface-type so;
```

For a full configuration example, see the *Junos OS Feature Guides*.

Configuring T3 Interfaces

To configure a T3 interface on an OC48/STM16 IQE PIC, include the **partition**, **oc-slice**, and **interface-type** statements at the **[edit interfaces coc48-fpc/pic/port]** hierarchy level, specifying the **coc1** interface type:

```
[edit interfaces coc48-fpc/pic/port]
partition partition-number oc-slice oc-slice-range interface-type coc1;
```

This configuration creates interface **coc1-fpc/pic/port:channel**.

Then, include the **no-partition interface-type** statement at the **[edit interfaces coc1-fpc/pic/port:channel]** hierarchy level, specifying the **t3** interface type:

```
[edit interfaces coc1-fpc/pic/port:channel]
no-partition interface-type t3;
```

This configuration creates interface **t3-fpc/pic/port:channel**.

The partition number is the sublevel interface partition index and is correlated with the channel number. For channelized OC1 interfaces, the partition number can be from 1 through 48. For channelized OC48/STM16 IQE interfaces, channel numbering begins with 1 (:1).

The OC-slice range is the range of SONET/SDH slices. For SONET/SDH interfaces, the OC-slice range specifies the bandwidth size required for the interface type you are configuring. For channelized OC1 interfaces, the OC slice can be from 1 through 12. You can configure only one OC slice per channelized OC1 interface.

The interface type is the channelized interface type or clear channel you are creating. For channelized OC48 interfaces, **type** can be **so** or **coc1**.



NOTE: Channelized OC48/STM16 IQE interfaces in M Series, MX Series, and T Series routers reserve channels 0 through 3 of each OC12 space for STS3C SONET channels.

When you configure E3 or T3 channels in OC12 spaces on the described PICs, the Junos OS allocates them starting from channel 4 because channels 0 through 3 are reserved for four STS3c SONET channels. Channel numbers are allocated sequentially in the following order: 4, 5, 6, 7, 8, 9, 11, 0, 1, 2, 3.

Only after channels 4 through 11 of the OC12 space are exhausted (all 4 through 11 configured) for E3 or T3 channels will the Junos OS then allocate channel 0 through 3 space for further E3 or T3 channels; thereby using up the 0 through 3 space previously reserved for four STS3c SONET channels.

If a subsequent reconfiguration of this OC12 space occurs, where you try to replace channels 4 through 6 or 7 through 9 with an OC3 SONET channel; the configuration fails because the channel 0 through 3 space is already occupied by the last E3 or T3 channels configured. This causes a failure in channel allocation and the Device Control Daemon (DCD) keeps retrying forever to configure the channel allocation on the interface. The only resolution is to reconfigure the last configured E3/T3 channels with OC3 channels, to free channels 0 through 3.

Example: Configuring T3 Interfaces

Configure a T3 interface using partition 3 and OC slice 3. This configuration creates interface **t3-1/1/0:3**:

```
[edit interfaces coc48-1/1/0]
partition 3 oc-slice 3 interface-type coc1;
[edit interfaces coc1-1/1/0:3]
no-partition interface-type t3;
```

For a full configuration example, see the *Junos OS Feature Guides*.

Configuring T1 Interfaces

To configure T1 interfaces on a Channelized OC48 IQE PIC, perform the following tasks:

1. Partition the channelized OC48 IQE interface into channelized OC1 interfaces by including the **partition**, **oc-slice**, and **interface-type** statements at the **[edit interfaces coc48-fpc/pic/port]** hierarchy level, specifying the **coc1** interface type:

```
[edit interfaces coc48-fpc/pic/port]
partition partition-number oc-slice oc-slice-range interface-type coc1;
```

2. If your network equipment uses VT mapping, partition the channelized OC1 interface into T1 interfaces by including the **partition** and **interface-type** statements at the **[edit interfaces coc1-fpc/pic/port]** hierarchy level, specifying the **t1** interface type:

```
[edit interfaces coc1-fpc/pic/port:channel]
partition partition-number interface-type t1;
```


3. If your network equipment uses M13 or C-bit parity, convert the channelized OC1 interface into a channelized T3 interface by including the **no-partition** and **interface-type** statements at the `[edit interfaces coc1-fpc/pic/port:channel]` hierarchy level, specifying the **ct3** interface type. Note that because the **no-partition** statement is included, this configuration does not create another level of channelization, as denoted by the number of colons in the resulting interface.

```
[edit interfaces coc1-fpc/pic/port:channel]
no-partition partition-number interface-type ct3;
```

4. Partition the channelized T3 interface into T1 interfaces by including the **partition** and **interface-type** statements at the `[edit interfaces ct3-fpc/pic/port:channel]` hierarchy level, specifying the **t1** interface type:

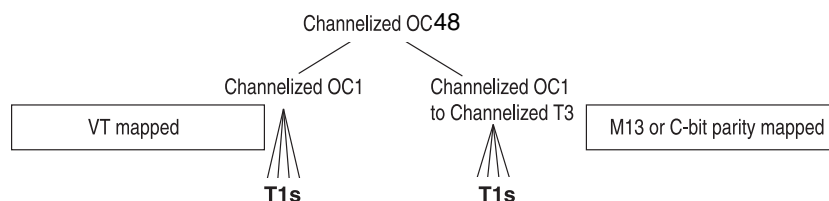
```
[edit interfaces ct3-fpc/pic/port:channel]
partition partition-number interface-type t1;
```



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQE interfaces. You can only apply CoS rules to the aggregate bit streams.

Figure 17 on page 53 shows VT-mapped and M13 or C-bit parity-mapped configurations of T1 interfaces.

Figure 17: T1 Interfaces on a Channelized OC48 PIC



g017307

Bold entries correspond to actual packet channels.

Example: Configuring T1 Interfaces

Configure the following T1 interfaces:

```
t1-0/0/0:1:1
t1-0/0/0:1:2
t1-0/0/0:1:3
t1-0/0/0:1:4
t1-0/0/0:1:5
```

VT-Mapped Configuration

```
[edit interfaces coc48-0/0/0]
partition 1 oc-slice 1 interface-type coc1;
```

```
[edit interfaces coc1-0/0/0:1]
partition 1-5 interface-type t1;
```

**M13 or C-bit
Parity-Mapped
Configuration**

```
[edit interfaces coc48-0/0/0]
partition 1 oc-slice 1 interface-type coc1;

[edit interfaces coc1-0/0/0:1]
no-partition interface-type ct3;

[edit interfaces ct3-0/0/0:1]
partition 1-5 interface-type t1;
```

For a full configuration example, see the *Junos OS Feature Guides*.

Configuring Fractional T1 Interfaces

By default, all the time slots on a channelized T1 interface are used. To configure a fractional T1 interface on a Channelized OC48 IQE PIC, perform the following tasks:

1. Configure a T1 interface. For more information, see “Configuring T1 Interfaces” on page 52.
2. Configure the number of time slots allocated to the T1 interface by including the **timeslots** statement at the **[edit interfaces t1-fpc/pic/port<:channel> t1-options]** hierarchy level:

```
[edit interfaces t1-fpc/pic/port<:channel> t1-options]
timeslots time-slot-range;
```

For channelized T1 interfaces, the time-slot range is from 1 through 24. You can designate any combination of time slots. To configure ranges, use hyphens. To configure discontinuous time slots, use commas. Do not include spaces. For more information, see “Configuring Fractional T1 Time Slots” on page 185.

Example: Configuring Fractional T1 Interfaces

Configure a fractional T1 interface that uses time slots 1 through 5 and 10:

```
[edit interfaces coc48-0/0/0]
partition 1 oc-slice 1 interface-type coc1;
[edit interfaces coc1-0/0/0:1]
partition 1 interface-type t1;
[edit interfaces t1-0/0/0:1 t1-options]
timeslots 1-5,10;
```

For a full configuration example, see the *Junos OS Feature Guides*.

Configuring NxDSO Interfaces

To configure NxDSO interfaces on a Channelized OC48 IQE PIC, perform the following tasks:

1. Partition the channelized OC48 IQE interface into channelized OC1 interfaces by including the **partition**, **oc-slice**, and **interface-type** statements at the **[edit interfaces coc48-fpc/pic/port:channel]** hierarchy level, specifying the **coc1** interface type:

```
[edit interfaces coc48-fpc/pic/port:channel]
partition partition-number oc-slice oc-slice-range interface-type coc1;
```

2. If your network equipment uses VT mapping, partition the channelized OC1 interface into channelized T1 interfaces by including the **partition** and **interface-type** statements at the `[edit interfaces coc1-fpc/pic/port:channel]` hierarchy level, specifying the **ct1** interface type:

```
[edit interfaces coc1-fpc/pic/port:channel]
partition partition-number interface-type ct1;
```



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQE interfaces. You can only apply CoS rules to the aggregate bit streams.

If your network equipment uses M13 or C-bit parity, convert the channelized OC1 interface into a channelized T3 interface by including the **no-partition** and **interface-type** statements at the `[edit interfaces coc1-fpc/pic/port]` hierarchy level, specifying the **ct3** interface type:

```
[edit interfaces coc1-fpc/pic/port:channel]
no-partition partition-number interface-type ct3;
```



NOTE: Because the **no-partition** statement is included, this configuration task does not create another level of channelization, as denoted by the number of colons in the resulting interface.

3. Partition the channelized T3 interface into channelized T1 interfaces by including the **partition** and **interface-type** statements at the `[edit interfaces ct3-fpc/pic/port:channel]` hierarchy level, specifying the **ct1** interface type:

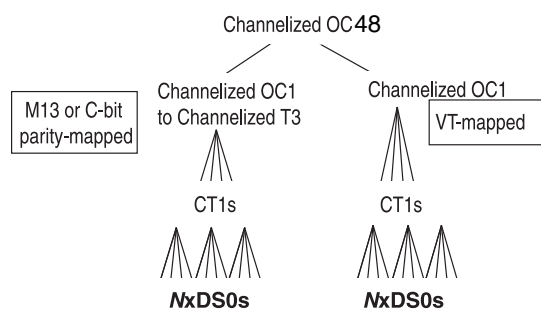
```
[edit interfaces ct3-fpc/pic/port:channel]
partition partition-number interface-type ct1;
```

4. Configure channelized NxDS0 interfaces on the channelized T1 interface by including the **partition**, **timeslots**, and **interface-type** statements at the `[edit interfaces ct1-fpc/pic/port:channel:channel]` hierarchy level, specifying the **ds** interface type:

```
[edit interfaces ct1-fpc/pic/port:channel:channel]
partition partition-number timeslots time-slot-range interface-type ds;
```

Figure 18 on page 56 shows VT-mapped and M13 or C-bit parity-mapped configurations of NxDS0 interfaces.

Figure 18: Sample Channelization of OC48 IQE PIC



g017308

Bold entries correspond to actual packet channels.

Example: Configuring NxDS0 Interfaces

Configure the following two NxDS0 interfaces with 10 time slots and 4 time slots, respectively:

```
ds-0/0/0:1:2:1
ds-0/0/0:1:2:2
```

VT-Mapped Configuration

```
[edit interfaces coc48-0/0/0]
partition 1 oc-slice 1 interface-type coc1;

[edit interfaces coc1-0/0/0:1]
partition 2 interface-type ct1;

[edit interfaces ct1-0/0/0:1:2]
partition 1 timeslots 1-10 interface-type ds;
partition 2 timeslots 12-15 interface-type ds;
```

M13 or C-bit Parity-Mapped Configuration

```
[edit interfaces coc48-0/0/0]
partition 1 oc-slice 1 interface-type coc1;

[edit interfaces coc1-0/0/0:1]
no-partition interface-type ct3;

[edit interfaces ct3-0/0/0:1]
partition 2 interface-type ct1;

[edit interfaces ct1-0/0/0:1:2]
partition 1 timeslots 1-10 interface-type ds;
partition 2 timeslots 12-15 interface-type ds;
```

For a full configuration example, see the *Junos OS Feature Guides*.

Configuring Channelized OC48/STM16 IQE Interfaces (SDH Mode)

The Channelized OC48 IQE PIC configured for SDH mode creates a single channelized STM16 interface. You can configure the STM16 interface using the **partition** statement

at the `[edit interfaces cstm16-fpc/pic/port]` hierarchy level to partition it into the following OC slices:

- 16 channelized AU-4 interfaces or a path layer with 4 STM4 or 16 STM1 interfaces.
- 16 channelized AU-4 interfaces, each partitioned to 3 clear channel E3 interfaces or 63 CE1 or E1 (ITU-T or KLM) interfaces. Combination of E1, CE1 and E3 are not supported in a single cau4.
- 16 channelized AU-4 interfaces, each partitioned to 63 CE1 (ITU-T or KLM) interfaces each partitioned to 31 NxDS0 interfaces

This section describes how to configure the following channelized OC48 IQE interfaces on a Channelized OC48 IQE PIC configured in SDH mode:

- Configuring a Channelized OC48/STM16 IQE PIC for SDH Mode on page 57
- Configuring Clear Channel STM1 and STM4 Interfaces on page 57
- Configuring Channelized AU-4 Interfaces on page 58
- Configuring E3 Interfaces on page 58
- Configuring E1 or Channelized E1 Interfaces on page 59
- Configuring NxDS0 IQE Interfaces on page 60
- Configuring T3 or Channelized T3 Interfaces on page 60
- Configuring T1 or Channelized T1 Interfaces on page 61

Configuring a Channelized OC48/STM16 IQE PIC for SDH Mode

To configure a Channelized OC48/STM16 IQE PIC to operate in SDH mode, include the **framing sdh** statement at the `[edit chassis fpc fpc/pic/port]` hierarchy level:

```
[edit chassis ]
  fpc 0 {
    pic 2 {
      framing sdh;
    }
  }
}
```

This configuration creates interface **cstm16-0/2/0**.

For more information, see the *Junos OS System Basics Configuration Guide*.

Configuring Clear Channel STM1 and STM4 Interfaces

On a Channelized OC48/STM16 IQE PIC, you can partition the CSTM16 transport layer into 4 clear channel STM4 interfaces or 16 clear channel STM1 interfaces. Combinations of STM4 and STM1 are also permitted, but you must observe the OC-slice parameters.

To configure an STM4 interface, include the **partition** and **interface-type** statements at the `[edit interfaces cstm16-fpc/pic/port]` hierarchy level:

```
[edit interfaces cstm16-fpc/pic/port]
  partition partition-number oc-slice oc-slice-range interface-type so;
```

This configuration creates interface **so-fpc/pic/port.channel**.

To configure an STM1 interface, include the **partition** and **interface-type** statements at the **[edit interfaces cstm16-fpc/pic/port]** hierarchy level:

```
[edit interfaces cstm16-fpc/pic/port]
partition partition-number oc-slice oc-slice-range interface-type so;
```

This configuration creates interface **so-fpc/pic/port.channel**.

Configuring Channelized AU-4 Interfaces

To configure a channelized AU-4 interface, include the **partition**, **oc-slice**, and **interface-type** statements at the **[edit interfaces cstm16-fpc/pic/port:channel]** hierarchy level, specifying the **cau4** interface type:

```
[edit interfaces cstm16-fpc/pic/port]
partition partition-number oc-slice oc-slice-range interface-type cau4;
```

This configuration creates interface **cau4-fpc/pic/port:channel**.

The partition number is the sublevel interface partition index. For SDH interfaces, the partition number is not correlated with bandwidth size. For channelized OC48/STM16 IQE interfaces, channelized STM16 interface can have from 1 through 16 partition numbers and channel numbering begins with 1 (:1).

The OC-slice range is the range of SONET/SDH slices. For SDH interfaces, the OC-slice range specifies the bandwidth size required for the interface type you are configuring. The interface type is the channelized interface type or data channel you are creating.

Example: Configuring Channelized AU-4 Interfaces

Configure channelized AU-4 interfaces:

```
[edit interfaces cstm16-0/2/0]
partition 1 oc-slice 1-3 interface-type cau4;
```

Configuring E3 Interfaces

To configure E3 interfaces, include the **partition** and **interface-type** statements at the **[edit interfaces cau4-fpc/pic/port]** hierarchy level, specifying the **e3** interface type:

```
[edit interfaces]
cau4-fpc/pic/port {
  partition partition-number interface-type e3;
}
```

This configuration creates the interfaces **e3-fpc/pic/port:channel**.



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQE interfaces. You can only apply CoS rules to the aggregate bit streams.



NOTE: Channelized OC48/STM16 IQE interfaces in M Series, MX Series, and T Series routers reserve channels 0-3 of each OC12 space for STS3C SONET channels.

When you configure E3 or T3 channels in OC12 spaces on the described PICs, Junos OS allocates them starting from channel 4 because channels 0-3 are reserved for four STS3c SONET channels. Channel numbers are allocated sequentially in the following order: 4, 5, 6, 7, 8, 9, 11, 0, 1, 2, 3.

Only after channels 4 through 11 of the OC12 space are exhausted (all 4 through 11 configured) for E3 or T3 channels will Junos OS then allocate channel 0-3 space for further E3 or T3 channels; thereby using up the 0-3 space previously reserved for four STS3c SONET channels.

If a subsequent reconfiguration of this OC12 space occurs, where you try to replace channels 4-6 or 7-9 with an OC3 SONET channel; it fails because the channel 0-3 space is already occupied by the last E3 or T3 channels configured. This causes a failure in channel allocation and the Device Control Daemon (DCD) keeps retrying forever to configure the channel allocation on the interface. The only resolution is to reconfigure the last configured E3/T3 channels with OC3 channels, to free channels 0-3.

Example: Configuring E3 Interfaces

Configure E3 interfaces, using partition 1:

```
[edit interfaces]
cau4-0/2/0:1 {
  partition 1 interface-type e3;
}
e3-0/2/0:1:1;
```

Configuring E1 or Channelized E1 Interfaces

To configure E1 or channelized E1 interfaces, include the **partition** and **interface-type** statements at the **[edit interfaces cau4-fpc/pic/port]** hierarchy level, specifying the **e1** or **ce1** interface type:

```
[edit interfaces]
cau4-fpc/pic/port {
  partition partition-number interface-type e1;
}
cau4-fpc/pic/port {
  partition partition-number interface-type ce1;
}
```

This configuration creates the interfaces **e1-fpc/pic/port:channel** and **ce1-fpc/pic/port:channel**.



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQE interfaces. You can only apply CoS rules to the aggregate bit streams.

Example: Configuring E1 and Channelized E1 Interfaces

Configure E1 or channelized E1 interfaces, using partition 3 and partition 4:

```
[edit interfaces]
cau4-0/2/0:1 {
  partition 3 interface-type e1;
}
cau4-0/2/0:1 {
  partition 4 interface-type ce1;
}
```

This configuration creates interfaces **e1-0/2/0:1:3** and **ce1-0/2/0:1:4**.

Configuring NxDS0 IQE Interfaces

Configure channelized NxDS0 IQE interfaces on the channelized E1 IQE interface by including the **partition**, **timeslots**, and **interface-type** statements at the **[edit interfaces ce1-fpc/pic/port:channel]** hierarchy level, specifying the **ds** interface type:

```
[edit interfaces ce1-fpc/pic/port:channel:channel]
partition partition-number timeslots time-slot-range interface-type ds;
```

This configuration creates the interface **ds-fpc/pic/port:channel**.

The time-slot range is from 1 through 31. You can designate any combination of time slots. To configure ranges, use hyphens. To configure discontinuous time slots, use commas. You can use a combination of ranges and discontinuous time slots, for example:

1,9-18,21

Example: Configuring NxDS0 IQE Interfaces

Configure channelized NxDS0 interfaces, using partition 4 and time slots 1 through 10:

```
[edit interfaces]
ce1-0/2/0:1:2:3 {
  partition 4 interface-type ds0 timeslots 1-10;
}
```

This configuration creates interface **ds0-0/2/0:1:2:3:4**.

Configuring T3 or Channelized T3 Interfaces

To configure T3 or channelized T3 interfaces, include the **partition** and **interface-type** statements at the **[edit interfaces cau4-fpc/pic/port]** hierarchy level, specifying the **t3** or **ct3** interface type:

```
[edit interfaces]
cau4-fpc/pic/port {
  partition partition-number interface-type t3;
```



```

}
cau4-fpc/pic/port {
  partition partition-number interface-type ct3;
}

```

This configuration creates the interfaces **t3-fpc/pic/port:channel** and **t3-fpc/pic/port:channel**.



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQ interfaces. You can only apply CoS rules to the aggregate bit streams.

Example: Configuring T3 or Channelized T3 Interfaces

Configure T3 and channelized T3 interfaces, using partition 1 and partition 2:

```

[edit interfaces]
cau4-0/2/0:1 {
  partition 1 interface-type t3;
}
cau4-0/2/0:1 {
  partition 2 interface-type ct3;
}
t3-0/2/0:1:1 ct3-0/2/0:1:2;

```

Configuring T1 or Channelized T1 Interfaces

To configure T1 or channelized T1 interfaces, include the **partition** and **interface-type** statements at the **[edit interfaces cau4-fpc/pic/port]** hierarchy level, specifying the **t1** or **ct1** interface type:

```

[edit interfaces]
cau4-fpc/pic/port {
  partition partition-number interface-type t1;
}
cau4-fpc/pic/port {
  partition partition-number interface-type ct1;
}

```

This configuration creates the interfaces **t1-fpc/pic/port:channel** and **t1-fpc/pic/port:channel**.



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQ interfaces. You can only apply CoS rules to the aggregate bit streams.

Example: Configuring T1 or Channelized T1 Interfaces

Configure T1 and channelized T1 interfaces, using partition 1 and partition 2:

```

[edit interfaces]
cau4-0/2/0:1 {

```

```
    partition 1 interface-type t1;
  }
  cau4-0/2/0:1 {
    partition 2 interface-type ct1;
  }
  t1-0/2/0:1:1 ct1-0/2/0:1:2;
```

Configuring Link PIC Failover on Channelized OC48/STM16 IQE Interfaces

For Channelized OC48 IQE PICs used as linking PICs in redundant LSQ configurations, you can inhibit the router from sending PPP termination-request messages to the remote host if the link PIC fails. To do this, include the **no-termination-request** statement at the **[edit interfaces *interface-name* ppp-options]** hierarchy level:

```
no-termination-request;
```

The **no-termination-request** statement is supported only with MLPPP and SONET APS configurations and works with PPP, PPP over Frame Relay, and MLPPP interfaces only.

For information about interchassis and intrachassis LSQ failover, see the [Junos OS Services Interfaces Configuration Guide](#).

Example: Configuring Channelized OC48 Interfaces with Partitioned Channels

The following configuration is sufficient to get the channelized OC48 interface up and running. The OC48 interface can be divided into up to 4 OC12 channels, up to 16 OC3 channels, or up to 48 OC1 channels and combinations are permitted; for example, 1 OC12, 4 OC3s, and 24 OC1s. There are 48 OC1 slices available on the OC48 IQE interface. An OC48 configuration uses all 48 slices, each OC12 uses 12 slices, each OC1 uses 1 slice. Permissible combinations must fit within the 48 available OC1 slices. DS1 channels can use the following encapsulation types:

- PPP, PPP CCC, and PPP TCC
- Frame Relay, Frame Relay CCC, and Frame Relay TCC
- Cisco HDLC, Cisco HDLC CCC, and Cisco HDLC TCC

The channels can also have logical interfaces.

```
[edit interfaces]
t3-fpc/pic/port:0 {
  encapsulation cisco-hdlc;
  t3-options {
    compatibility-mode larscom;
    payload-scrambler;
  }
  unit 0 {
    family inet {
      address 10.11.30.1/30;
    }
    family iso;
  }
}
```

```
t3-fpc/pic/port:1 {
  encapsulation ppp;
  t3-options {
    compatibility-mode larscom;
    payload-scrambler;
  }
  unit 0 {
    family inet {
      address 10.11.30.5/30;
    }
    family iso;
  }
}
t3-fpc/pic/port:2 {
  encapsulation frame-relay;
  t3-options {
    compatibility-mode larscom;
    payload-scrambler;
  }
  unit 0 {
    dlci 100;
    family inet {
      address 10.11.30.9/30;
    }
    family iso;
  }
  unit 1 {
    dlci 101;
    family inet {
      address 10.11.31.9/30;
    }
    family iso;
  }
}
t3-lfpc/pic/port:3 {
  encapsulation cisco-hdlc-ccc;
  t3-options {
    compatibility-mode larscom;
    payload-scrambler;
  }
  unit 0;
}
t3-fpc/pic/port:4 {
  encapsulation ppp-ccc;
  t3-options {
    compatibility-mode larscom;
    payload-scrambler;
  }
  unit 0;
}
t3-fpc/pic/port:5 {
  dce;
  encapsulation frame-relay-ccc;
  t3-options {
    compatibility-mode larscom;
    payload-scrambler;
  }
}
```

```
    }  
    unit 0 {  
        encapsulation frame-relay-ccc;  
        dlci 1000;  
    }  
    unit 1 {  
        encapsulation frame-relay-ccc;  
        dlci 1001;  
    }  
}
```

CHAPTER 4

Configuring Channelized OC12/STM4 Interfaces

- Channelized OC12/STM4 IQ and IQE Interfaces Overview on page 65
- Channelization of OC12/STM4 IQ and Channelized OC12/STM4 IQE PICs (SONET Mode) on page 65
- Channelization of OC12/STM4 IQE PIC (SDH Mode) on page 66
- Channelization of OC12/STM4 IQ PIC (SDH Mode) on page 67
- Channelization of OC12 PIC (SONET Mode) on page 68
- Configuring Channelized OC12/STM4 IQ and IQE Interfaces (SONET Mode) on page 69
- Configuring Channelized OC12/STM4 IQE Interfaces (SDH Mode) on page 76
- Configuring Channelized OC12/STM4 IQ Interfaces (SDH Mode) on page 81
- Configuring Channelized OC12 Interfaces on page 87
- Configuring Link PIC Failover on Channelized OC12/STM4 IQ and IQE Interfaces on page 90
- Example: Configuring a Channelized OC12 IQ Interface as an Unpartitioned Clear Channel on page 90
- Example: Configuring Channelized OC12 Interfaces with Partitioned Channels on page 94

Channelized OC12/STM4 IQ and IQE Interfaces Overview

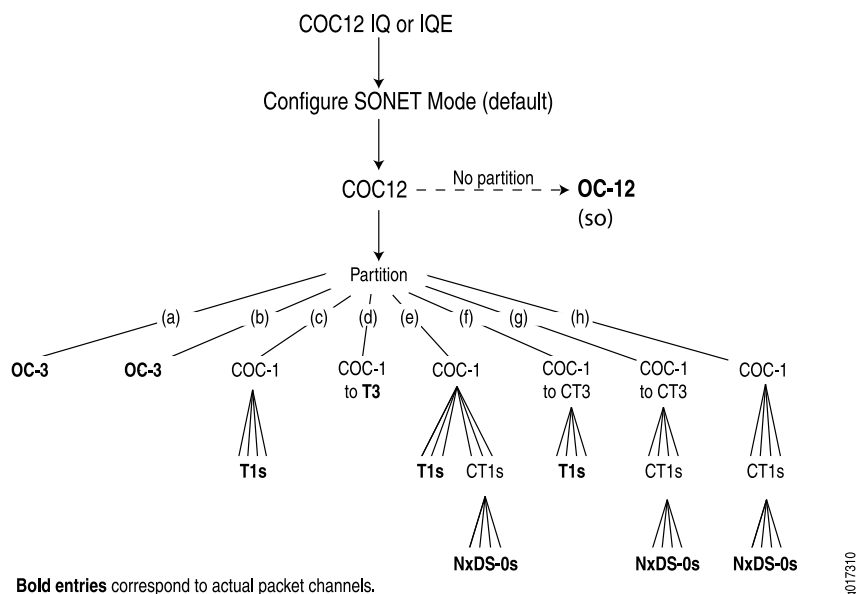
Channelized IQ and channelized IQE interfaces allow arbitrary and dynamic channelization of serial links, allowing greater flexibility than the channelized interfaces. Channelized OC12/STM4 IQ and IQE Physical Interface Cards (PICs) can be configured to operate in SONET or SDH mode. Each physical port on a multiple-port IQE PIC can be configured to operate in either SONET or SDH mode for increased granularity. The following sections describe the different modes of operation and channelization possibilities.

Channelization of OC12/STM4 IQ and Channelized OC12/STM4 IQE PICs (SONET Mode)

Channelized OC12/STM4 IQ PICs and Channelized OC12/STM4 IQE PICs can be configured to operate in SONET or SDH mode and partitioned into various partitions. Figure 19 on

page 66 illustrates one possible channelization configuration for Channelized OC12/STM4 IQ and IQE PICs operating in SONET mode.

Figure 19: Sample Channelization of OC12/STM4 IQ or IQE PIC (SONET Mode)



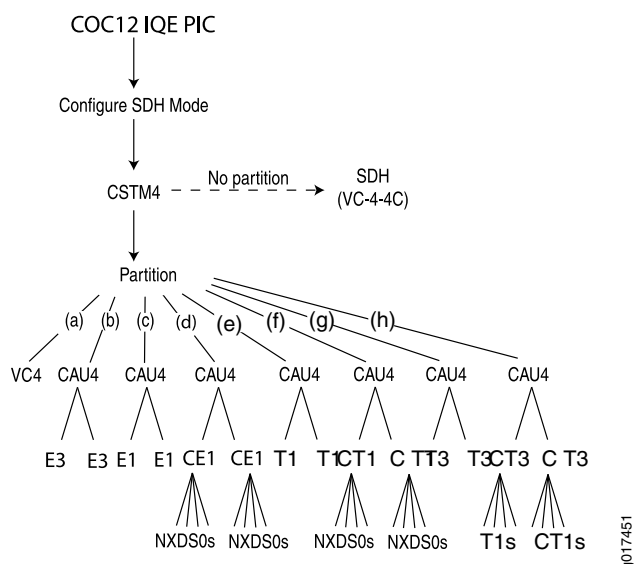
In the example in Figure 19 on page 66, a Channelized OC12/STM4 IQ or IQE PIC operating in SONET mode is partitioned into the following OC slices:

- a. An OC3 interface.
- b. Another OC3 interface.
- c. A channelized OC1 partitioned into T1 interfaces.
- d. A channelized OC1 converted into a T3 interface.
- e. A channelized OC1 partitioned into T1 interfaces and channelized T1s, which are partitioned into NxDS0 interfaces.
- f. A channelized OC1 converted into a channelized T3, which is partitioned into T1 interfaces.
- g. A channelized OC1 converted into a channelized T3, which is partitioned into T1 interfaces and a channelized T1, which is partitioned into NxDS0 interfaces.
- h. A channelized OC1 partitioned into channelized T1s, which are partitioned into NxDS0 interfaces.

Channelization of OC12/STM4 IQE PIC (SDH Mode)

Channelized OC12/STM4 IQE PICs can be configured to operate in SONET or SDH mode and partitioned to various smaller partitions. Figure 20 on page 67 illustrates one possible channelization configuration for Channelized OC12/STM4 IQE PICs operating in SDH mode.

Figure 20: Sample Channelization of OC12/STM4 IQE PIC (SDH Mode)



In Figure 20 on page 67, a Channelized OC12/STM4 IQE PIC operating in SDH mode results in a channelized STM4 interface, which can be nonpartitioned into one SDH VC-4-VC interface or partitioned into the following OC slices:

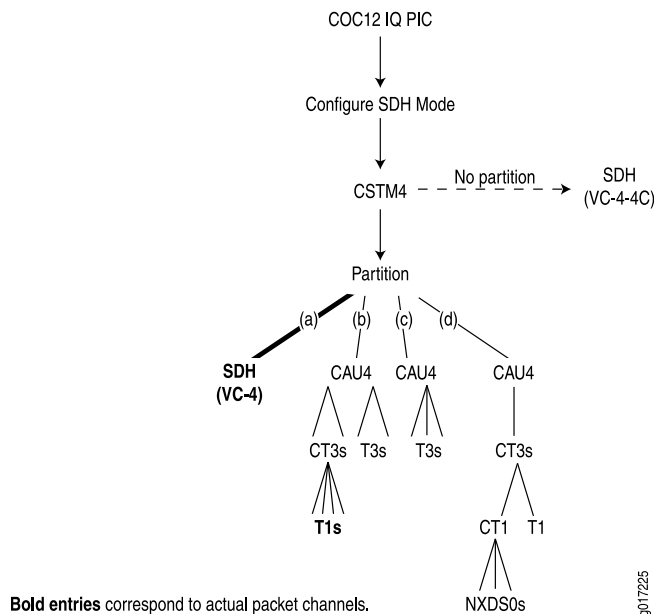
- a. An SDH VC-4 interface.
- b. A channelized AU-4 partitioned into E3 interfaces.
- c. A channelized AU-4 interface partitioned into E1 interfaces.
- d. A channelized AU-4 interface partitioned into CE1 interfaces partitioned into NxDS0 interfaces.
- e. A channelized AU-4 interface partitioned into T1 interfaces.
- f. A channelized AU-4 interface partitioned into CT1 interfaces.
- g. A channelized AU-4 interface partitioned into T3 interfaces partitioned into T1 interfaces.
- h. A channelized AU-4 interface partitioned into CT3 interfaces partitioned into CT1 interfaces.

This is one of thousands of ways to configure a Channelized OC12/STM4 IQE PIC.

Channelization of OC12/STM4 IQ PIC (SDH Mode)

Channelized OC12/STM4 IQ PICs can be configured to operate in SONET or SDH mode and partitioned into various smaller partitions. Figure 21 on page 68 illustrates one possible channelization configuration for Channelized OC12/STM4 IQ PICs operating in SDH mode.

Figure 21: Sample Channelization of OC12/STM4 IQ PIC (SDH Mode)



In Figure 21 on page 68, a Channelized OC12/STM4 IQ PIC operating in SDH mode results in a channelized STM4 interface, which can be nonpartitioned into one SDH VC-4-VC interface or partitioned into the following OC slices:

- An SDH VC-4 interface.
- A channelized AU-4 partitioned into channelized T3 interfaces and T3 interfaces.
- Another channelized AU-4 interface converted into T3 interfaces.
- Another channelized AU-4 interface converted into a channelized T3 interface, which is partitioned further into a channelized T1 and a T1 interface. The channelized T1 interface is further partitioned into NxDS0 interfaces.

This is one of thousands of ways to configure a Channelized OC12/STM4 IQ PIC.

Channelization of OC12 PIC (SONET Mode)

OC12 PICs can be configured to various smaller partitions, such as T3s.

Figure 22: Sample Channelization of OC12 PIC (non IQ and IQE)

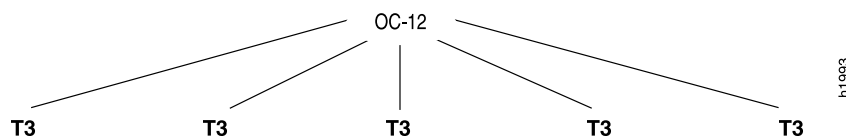


Figure 22 on page 68 shows five T3 channels configured on the Channelized OC12 PIC. You can configure seven additional T3 channels. For more information about configuring Channelized OC12 PICs, see “Configuring Channelized OC12 Interfaces” on page 87.

Configuring Channelized OC12/STM4 IQ and IQE Interfaces (SONET Mode)

This section describes how to configure channelized OC12/STM4 intelligent queuing (IQ) and enhanced intelligent queuing (IQE) interfaces, discussing the following topics:

- Configuring an OC12/STM4 Interface on page 69
- Configuring T3 Interfaces on page 69
- Configuring OC3 Interfaces on page 71
- Configuring T1 Interfaces on Channelized OC12 IQ and IQE Interfaces on page 71
- Configuring NxDS0 Interfaces on page 73
- Configuring Fractional T1 Interfaces on page 75

Configuring an OC12/STM4 Interface

You can configure one OC12 interface on a 1-port Channelized OC12/STM4 IQ or IQE PIC. On a 4-port OC12/STM4 IQ or IQE PIC, you can configure one OC12 interface per port. To configure an OC12 interface, include the **no-partition** and **interface-type** statements at the **[edit interfaces coc12-fpc/pic/port]** hierarchy level:

```
[edit interfaces coc12-fpc/pic/port]
no-partition interface-type so;
```

This configuration creates interface **so-fpc/pic/port**.



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQ and IQE interfaces. You can apply CoS rules only to the aggregate bit streams.



NOTE: If you configure the **per-unit-scheduler** statement on the physical interface of a 4-port Channelized OC-12 IQ PIC and configure 975 logical interfaces or DLCIs, some of the logical interfaces or data link connection identifiers (DLCIs) will drop all packets intermittently.

Configuring T3 Interfaces

To configure a T3 interface on an OC12 PIC, include the **partition**, **oc-slice**, and **interface-type** statements at the **[edit interfaces coc12-fpc/pic/port]** hierarchy level, specifying the **coc1** interface type:

```
[edit interfaces coc12-fpc/pic/port]
partition partition-number oc-slice oc-slice-range interface-type coc1;
```

This configuration creates interface **coc1-fpc/pic/port:channel**.

Then, include the **no-partition interface-type** statement at the **[edit interfaces coc1-fpc/pic/port:channel]** hierarchy level, specifying the **t3** interface type:

```
[edit interfaces coc1-fpc/pic/port:channel]  
no-partition interface-type t3;
```

This configuration creates interface **t3-fpc/pic/port:channel**.

The partition number is the sublevel interface partition index and is correlated with the channel number. For channelized OC1 interfaces, the partition number can be from 1 through 12.



NOTE: For channelized OC12 interfaces, channel numbering begins with 0 (:0). For channelized OC12/STM4 IQ and IQE interfaces, channel numbering begins with 1 (:1).

The OC-slice range is the range of SONET/SDH slices. For SONET/SDH interfaces, the OC-slice range specifies the bandwidth size required for the interface type you are configuring. For channelized OC1 interfaces, the OC slice can be from 1 through 12. You can configure only one OC slice per channelized OC1 interface.

The interface type is the channelized interface type or clear channel you are creating. For channelized OC12 interfaces, **type** can be **so** or **coc1**.



NOTE: Channelized OC12/STM4 IQ and IQE interfaces in M Series, MX Series, and T Series routers reserve channels 0 through 3 of each OC12 space for STS3c SONET channels.

When you configure E3 or T3 channels in OC12 spaces on the described PICs, Junos OS allocates them starting from channel 4 because channels 0 through 3 are reserved for four STS3c SONET channels. Channel numbers are allocated sequentially in the following order: 4, 5, 6, 7, 8, 9, 11, 0, 1, 2, 3.

Only after channels 4 through 11 of the OC12 space are exhausted (that is, channels 4 through 11 are configured) for E3 or T3 channels will Junos OS then allocate the channel 0–3 space for further E3 or T3 channels; thereby using up the 0–3 space previously reserved for four STS3c SONET channels.

If a subsequent reconfiguration of this OC12 space occurs, where you try to replace channels 4–6 or 7–9 with an OC3 SONET channel; it fails because the channel 0–3 space is already occupied by the last E3 or T3 channels configured. This causes a failure in channel allocation and the device control daemon (dcd) keeps retrying forever to configure the channel allocation on the interface. The only resolution is to reconfigure the last configured E3 or T3 channels with OC3 channels, to free channels 0 through 3.

Example: Configuring T3 Interfaces

Configure a T3 interface using partition 3 and OC slice 3. This configuration creates interface **t3-1/1/0:3**:

```
[edit interfaces coc12-1/1/0]
```

```
partition 3 oc-slice 3 interface-type coc1;
[edit interfaces coc1-1/1/0:3]
no-partition interface-type t3;
```

For a full configuration example, see the *Junos OS Feature Guides*.

Configuring OC3 Interfaces

To configure an OC3 interface, include the **partition**, **oc-slice**, and **interface-type** statements at the **[edit interfaces coc12-fpc/pic/port]** hierarchy level, specifying the **so** interface type:

```
[edit interfaces coc12-fpc/pic/port]
partition partition-number oc-slice oc-slice-range interface-type so;
```

The partition number is the sublevel interface partition index. For SONET/SDH interfaces, the partition number does not correlate with bandwidth size. For OC3 interfaces, the partition number can be from 1 through 4.



NOTE: For channelized OC12 interfaces, channel numbering begins with 0 (:0). For channelized OC12 IQ and IQE interfaces, channel numbering begins with 1 (:1).

The OC-slice range is the range of SONET/SDH slices. For SONET/SDH interfaces, the OC-slice range specifies the bandwidth size required for the interface type you are configuring. OC3 interfaces must occupy three consecutive OC slices per interface, in one of the following forms:

- 1–3
- 4–6
- 7–9
- 10–12

By contrast, the T3 and OC1 IQ interfaces each occupy one OC slice per interface.

The interface type is the channelized interface type or data channel you are creating. For channelized OC12 interfaces, the interface type can be **coc1** or **so**.

Example: Configuring OC3 Interfaces

Configure an OC3 interface, using partition 1 and OC slices 4 through 6. This configuration creates interface **so-1/1/0:1**:

```
[edit interfaces coc12-1/1/0]
partition 1 oc-slice 4-6 interface-type so;
```

For a full configuration example, see the *Junos OS Feature Guides*.

Configuring T1 Interfaces on Channelized OC12 IQ and IQE Interfaces

To configure T1 interfaces on a Channelized OC12 IQ or IQE PIC, perform the following tasks:

1. Partition the channelized OC12 interface into channelized OC1 interfaces by including the **partition**, **oc-slice**, and **interface-type** statements at the **[edit interfaces coc12-fpc/pic/port]** hierarchy level, specifying the **coc1** interface type:

```
[edit interfaces coc12-fpc/pic/port]
partition partition-number oc-slice oc-slice-range interface-type coc1;
```

2. If your network equipment uses virtual tributary (VT) mapping, partition the channelized OC1 interface into T1 interfaces by including the **partition** and **interface-type** statements at the **[edit interfaces coc1-fpc/pic/port]** hierarchy level, specifying the **t1** interface type:

```
[edit interfaces coc1-fpc/pic/port]
partition partition-number interface-type t1;
```

3. If your network equipment uses M13 or C-bit parity, convert the channelized OC1 interface into a channelized T3 interface by including the **no-partition** and **interface-type** statements at the **[edit interfaces coc1-fpc/pic/port:channel]** hierarchy level, specifying the **ct3** interface type. Note that because the **no-partition** statement is included, this configuration does not create another level of channelization, as denoted by the number of colons in the resulting interface.

```
[edit interfaces coc1-fpc/pic/port]
no-partition partition-number interface-type ct3;
```

4. Partition the channelized T3 interface into T1 interfaces by including the **partition** and **interface-type** statements at the **[edit interfaces ct3-fpc/pic/port]** hierarchy level, specifying the **t1** interface type:

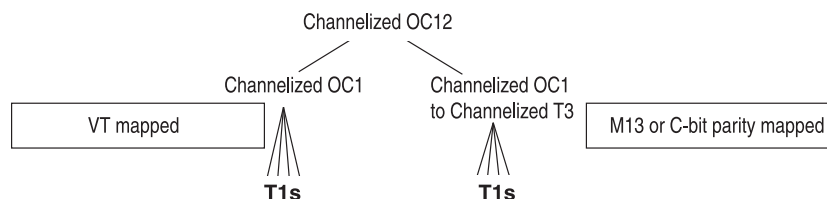
```
[edit interfaces ct3-fpc/pic/port]
partition partition-number interface-type t1;
```



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQ interfaces. You can apply CoS rules only to the aggregate bit streams.

Figure 23 on page 72 shows VT-mapped and M13 or C-bit parity-mapped configurations of T1 interfaces.

Figure 23: T1 Interfaces on a Channelized OC12 PIC



Bold entries correspond to actual packet channels.

Example: Configuring T1 Interfaces

Configure the following T1 interfaces:

```

t1-0/0/0:1:1
t1-0/0/0:1:2
t1-0/0/0:1:3
t1-0/0/0:1:4
t1-0/0/0:1:5

VT-Mapped
Configuration [edit interfaces coc12-0/0/0]
partition 1 oc-slice 1 interface-type coc1;

```

```

[edit interfaces coc1-0/0/0:1]
partition 1-5 interface-type t1;

```

```

M13 or C-bit
Parity-Mapped
Configuration [edit interfaces coc12-0/0/0]
partition 1 oc-slice 1 interface-type coc1;

```

```

[edit interfaces coc1-0/0/0:1]
no-partition interface-type ct3;

```

```

[edit interfaces ct3-0/0/0:1]
partition 1-5 interface-type t1;

```

For a full configuration example, see the [Junos OS Feature Guides](#).

Configuring NxDS0 Interfaces

To configure NxDS0 interfaces on a Channelized OC12 IQE PIC, perform the following tasks:

1. Partition the channelized OC12 IQE interface into channelized OC1 interfaces by including the **partition**, **oc-slice**, and **interface-type** statements at the **[edit interfaces coc12-fpc/pic/port]** hierarchy level, specifying the **coc1** interface type:

```

[edit interfaces coc12-fpc/pic/port]
partition partition-number oc-slice oc-slice-range interface-type coc1;

```

2. If your network equipment uses VT mapping, partition the channelized OC1 interface into channelized T1 interfaces by including the **partition** and **interface-type** statements at the **[edit interfaces coc1-fpc/pic/port]** hierarchy level, specifying the **ct1** interface type:

```

[edit interfaces coc1-fpc/pic/port]
partition partition-number interface-type ct1;

```



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQ interfaces. You can apply CoS rules only to the aggregate bit streams.

3. If your network equipment uses M13 or C-bit parity, convert the channelized OC1 interface into a channelized T3 interface by including the **no-partition** and **interface-type** statements at the **[edit interfaces coc1-fpc/pic/port]** hierarchy level, specifying the **ct3** interface type:

```

[edit interfaces coc1-fpc/pic/port]

```

```
no-partition partition-number interface-type ct3;
```



NOTE: Because the **no-partition** statement is included, this configuration task does not create another level of channelization, as denoted by the number of colons in the resulting interface.

- Partition the channelized T3 interface into channelized T1 interfaces by including the **partition** and **interface-type** statements at the **[edit interfaces ct3-fpc/pic/port]** hierarchy level, specifying the **ct1** interface type:

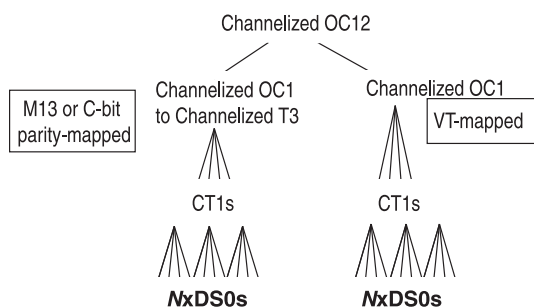
```
[edit interfaces ct3-fpc/pic/port]
partition partition-number interface-type ct1;
```

- Configure channelized NxDSO IQ interfaces on the channelized T1 IQ interface by including the **partition**, **timeslots**, and **interface-type** statements at the **[edit interfaces ct1-fpc/pic/port]** hierarchy level, specifying the **ds** interface type:

```
[edit interfaces ct1-fpc/pic/port:channel:channel]
partition partition-number timeslots time-slot-range interface-type ds;
```

Figure 24 on page 74 shows VT-mapped and M13 or C-bit parity-mapped configurations of NxDSO IQ interfaces.

Figure 24: Sample Channelization of OC12 IQE PIC



Bold entries correspond to actual packet channels.

g003014

Example: Configuring NxDSO Interfaces

Configure the following two NxDSO interfaces with 10 time slots and 4 time slots, respectively:

```
ds-0/0/0:1:2:1
ds-0/0/0:1:2:2
```

**VT-Mapped
Configuration**

```
[edit interfaces coc12-0/0/0]
partition 1 oc-slice 1 interface-type coc1;
```

```
[edit interfaces coc1-0/0/0:1]
partition 2 interface-type ct1;
```

**M13 or C-bit
Parity-Mapped
Configuration**

```
[edit interfaces ct1-0/0/0:1:2]
partition 1 timeslots 1-10 interface-type ds;
partition 2 timeslots 12-16 interface-type ds;
```

```
[edit interfaces coc12-0/0/0]
partition 1 oc-slice 1 interface-type coc1;
```

```
[edit interfaces coc1-0/0/0:1]
no-partition interface-type ct3;
```

```
[edit interfaces ct3-0/0/0:1]
partition 2 interface-type ct1;
```

```
[edit interfaces ct1-0/0/0:1:2]
partition 1 timeslots 1-10 interface-type ds;
partition 2 timeslots 12-16 interface-type ds;
```

For a full configuration example, see the [Junos OS Feature Guides](#).

Configuring Fractional T1 Interfaces

By default, all the time slots on a channelized T1 interface are used. To configure a fractional T1 interface on a Channelized OC12 IQE PIC, perform the following tasks:

1. Configure a T1 interface. For more information, see “Configuring T1 Interfaces” on page 52.
2. Configure the number of time slots allocated to the T1 interface by including the **timeslots** statement at the **[edit interfaces t1-fpc/pic/port<:channel> t1-options]** hierarchy level:

```
[edit interfaces t1-fpc/pic/port<:channel> t1-options]
timeslots time-slot-range;
```

For channelized T1 interfaces, the time-slot range is from 1 through 24. You can designate any combination of time slots. To configure ranges, use hyphens. To configure discontinuous time slots, use commas. Do not include spaces. For more information, see “Configuring Fractional T1 Time Slots” on page 185.

Example: Configuring Fractional T1 Interfaces

Configure a fractional T1 interface that uses time slots 1 through 5 and 10:

```
[edit interfaces coc12-0/0/0]
partition 1 oc-slice 1 interface-type coc1;
[edit interfaces coc1-0/0/0:1]
partition 1 interface-type t1;
[edit interfaces t1-0/0/0:1:1 t1-options]
timeslots 1-5,10;
```

For a full configuration example, see the [Junos OS Feature Guides](#).

Configuring Channelized OC12/STM4 IQE Interfaces (SDH Mode)

The Channelized OC12 IQE PIC configured for SDH mode creates a single channelized STM4 interface. You can configure this interface as unpartitioned using the **no-partition** statement at the **[edit interfaces cstm4-fpc/pic/port]** hierarchy level to create a single SDH VC-4-4C interface, or you can partition it into the following OC slices:

- SDH virtual concatenation 4 (VC-4) and channelized AU-4 interfaces (4 interfaces, any combination)
- E3 interfaces from a channelized AU-4 interface (3 interfaces, any combination)
- Channelized E1 or E1 interfaces from a channelized AU-4 interface (63 interfaces, any combination)
- NxDS0 interfaces from a channelized E1 interface

This section describes how to configure the following channelized OC12 IQE interfaces on a Channelized OC12 IQE PIC configured in SDH mode:

- Configuring Channelized OC12/STM4 IQE PICs for SDH Mode on page 76
- Configuring an Unpartitioned SDH (VC-4-4C) Interface on a Channelized OC12/STM4 IQE PIC on page 77
- Configuring SDH (VC-4) Interfaces on Channelized OC12/STM4 IQE PICs on page 77
- Configuring Channelized AU-4 Interfaces on page 78
- Configuring E3 Interfaces on page 79
- Configuring E1 or Channelized E1 Interfaces on page 80
- Configuring NxDS0 Interfaces on Channelized OC12/STM4 IQE PICs on page 81

Configuring Channelized OC12/STM4 IQE PICs for SDH Mode

The 4-port Channelized OC12 IQE PIC allows SONET/SDH configuration on a per port basis, permitting combinations of SONET and SDH ports on the same PIC. The 1-port Channelized OC12 IQE PIC operates in either SONET or SDH mode only.

To configure a 1-port Channelized OC12 IQE PIC to operate in SDH mode, include the **framing sdh** statement at the **[edit chassis fpc fpc/pic/port]** hierarchy level:

```
[edit chassis]
fpc 0 {
  pic 2 {
    framing sdh;
  }
}
```

This configuration creates interface **cstm4-0/2/0**.

You can also use the above configuration example to configure all 4 ports of a 4-port Channelized OC12 IQE PIC for SDH mode. To configure individual ports to operate in SDH mode, include the **framing sdh** statement at the **[edit chassis fpc fpc/pic/port]** hierarchy level. The following example configures port 2 for SDH mode:


```
[edit chassis]
fpc 0 {
  pic 2 {
    port 2 {
      framing sdh;
    }
  }
}
```

This configuration creates interface **cstm4-0/2/2**.

For more information, see the *Junos OS System Basics Configuration Guide*.

Configuring an Unpartitioned SDH (VC-4-4C) Interface on a Channelized OC12/STM4 IQE PIC

On a Channelized OC12 IQE PIC, you can configure one SDH (VC-4-4C) interface. To configure an SDH (VC-4-4C) interface, include the **no-partition** and **interface-type** statements at the **[edit interfaces cstm4-fpc/pic/port]** hierarchy level:

```
[edit interfaces cstm4-fpc/pic/port]
no-partition interface-type so;
```

This configuration creates interface **so-fpc/pic/port**.

Example: Configuring an Unpartitioned SDH (VC-4-4C) Interface

Configure an unpartitioned SDH (VC-4-4C) interface, using partition 1 and OC slices 4 through 6:

```
[edit interfaces cstm4-0/2/0]
no-partition interface-type so;
```

This configuration creates the interface **so-0/2/0**.

Configuring SDH (VC-4) Interfaces on Channelized OC12/STM4 IQE PICs

To configure an SDH (VC-4) interface on a Channelized OC12 IQE PIC, include the **partition**, **oc-slice**, and **interface-type** statements at the **[edit interfaces cstm4-fpc/pic/port]** hierarchy level, specifying the **so** interface type:

```
[edit interfaces cstm4-fpc/pic/port]
partition partition-number oc-slice oc-slice-range interface-type so;
```

This configuration creates interface **so-fpc/pic/port:channel**.

The partition number is the sublevel interface partition index and is correlated with the channel number. For Channelized OC12 IQE PICs, the OC-slice range can be from 1 through 12.



NOTE: For channelized OC12 IQE interfaces, channel numbering begins with 1 (:1).

The OC-slice range is the range of SONET/SDH slices. For SDH interfaces, the OC-slice range specifies the bandwidth size required for the interface type you are configuring.

SDH (VC-4) interfaces must occupy three consecutive OC slices per interface, in one of the following forms:

- 1–3
- 4–6
- 7–9
- 10–12

The interface type is the channelized interface type or data channel you are creating.

Example: Configuring SDH (VC-4) Interfaces

Configure SDH (VC-4) interfaces:

```
[edit interfaces cstm4-0/2/0]
partition 1 oc-slice 1-3 interface-type so;
partition 2 oc-slice 4-6 interface-type so;
partition 3 oc-slice 7-9 interface-type so;
partition 4 oc-slice 10-12 interface-type so;
```

This configuration creates the interfaces **so-0/2/0:1** through **so-0/2/0:4**.

Configuring Channelized AU-4 Interfaces

To configure a channelized AU-4 interface, include the **partition**, **oc-slice**, and **interface-type** statements at the **[edit interfaces cstm4-fpc/pic/port]** hierarchy level, specifying the **cau4** interface type:

```
[edit interfaces cstm4-fpc/pic/port]
partition partition-number oc-slice oc-slice-range interface-type cau4;
```

This configuration creates interface **cau4-fpc/pic/port:channel**.

The partition number is the sublevel interface partition index. For SDH interfaces, the partition number is not correlated with bandwidth size. A channelized STM-4 interface can have from 1 through 4 partition numbers.



NOTE: For channelized OC12 interfaces, channel numbering begins with 0 (:0). For channelized OC12 interfaces (both IQ and IQE), channel numbering begins with 1 (:1).

The OC-slice range is the range of SONET/SDH slices. For SDH interfaces, the OC-slice range specifies the bandwidth size required for the interface type you are configuring. Channelized AU-4 IQ interfaces must occupy three consecutive OC slices per interface, in one of the following forms:

- 1–3
- 4–6
- 7–9
- 10–12

The interface type is the channelized interface type or data channel you are creating.

Example: Configuring Channelized AU-4 Interfaces

Configure channelized AU-4 interfaces, using partitions 1 through 4:

```
[edit interfaces cstm4-0/2/0]
partition 1 oc-slice 1-3 interface-type cau4;
partition 2 oc-slice 4-6 interface-type cau4;
partition 3 oc-slice 7-9 interface-type cau4;
partition 4 oc-slice 10-12 interface-type cau4;
```

This configuration creates the interfaces **cau4-0/2/0:1** through **cau4-0/2/0:4**.

Configuring E3 Interfaces

To configure E3 interfaces, include the **partition** and **interface-type** statements at the **[edit interfaces cau4-fpc/pic/port]** hierarchy level, specifying the **e3** interface type:

```
[edit interfaces]
cau4-fpc/pic/port {
  partition partition-number interface-type e3;
}
```

This configuration creates the interfaces **e3-fpc/pic/port:channel** and **e3-fpc/pic/port:channel**.



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQE interfaces. You can only apply CoS rules to the aggregate bit streams.



NOTE: Channelized OC12/STM4 IQ and IQE interfaces in M Series, MX Series, and T Series routers reserve channels 0-3 of each OC12 space for STS3C SONET channels.

When you configure E3 or T3 channels in OC12 spaces on the described PICs, Junos OS allocates them starting from channel 4 because channels 0-3 are reserved for four STS3c SONET channels. Channel numbers are allocated sequentially in the following order: 4, 5, 6, 7, 8, 9, 11, 0, 1, 2, 3.

Only after channels 4 through 11 of the OC12 space are exhausted (all 4 through 11 configured) for E3 or T3 channels will Junos OS then allocate channel 0-3 space for further E3 or T3 channels; thereby using up the 0-3 space previously reserved for four STS3c SONET channels.

If a subsequent reconfiguration of this OC12 space occurs, where you try to replace channels 4-6 or 7-9 with an OC3 SONET channel; it fails because the channel 0-3 space is already occupied by the last E3 or T3 channels configured. This causes a failure in channel allocation and the Device Control Daemon (DCD) keeps retrying forever to configure the channel allocation on the interface. The only resolution is to reconfigure the last configured E3/T3 channels with OC3 channels, to free channels 0-3.

Example: Configuring E3 Interfaces

Configure E3 interfaces, using partition 1:

```
[edit interfaces]
cau4-0/2/0:1 {
  partition 1 interface-type e3;
}
e3-0/2/0:1:1;
```

Configuring E1 or Channelized E1 Interfaces

To configure E1 or channelized E1 interfaces, include the **partition** and **interface-type** statements at the **[edit interfaces cau4-fpc/pic/port]** hierarchy level, specifying the **e1** or **ce1** interface type:

```
[edit interfaces]
cau4-fpc/pic/port {
  partition partition-number interface-type e1;
}
cau4-fpc/pic/port {
  partition partition-number interface-type ce1;
}
```

This configuration creates the interfaces **e1-fpc/pic/port:channel** and **ce1-fpc/pic/port:channel**.



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQE interfaces. You can only apply CoS rules to the aggregate bit streams.

Example: Configuring E1 or Channelized CE1 Interfaces

Configure E1 or channelized CE1 interfaces, using partition 3 and partition 4:

```
[edit interfaces]
cau4-0/2/0:1 {
  partition 3 interface-type e1;
}
cau4-0/2/0:1 {
  partition 4 interface-type ce1;
}
```

This configuration creates interfaces **e1-0/2/0:1:3** and **ce1-0/2/0:1:4**.

Configuring NxDS0 Interfaces on Channelized OC12/STM4 IQE PICs

Configure channelized NxDS0 interfaces on the channelized E1 interface by including the **partition**, **timeslots**, and **interface-type** statements at the **[edit interfaces ce1-fpc/pic/port:channel]** hierarchy level, specifying the **ds** interface type:

```
[edit interfaces ce1-fpc/pic/port:channel:channel]
partition partition-number timeslots time-slot-range interface-type ds;
```

This configuration creates the interface **ds-fpc/pic/port:channel**.

The time-slot range is from 1 through 32. You can designate any combination of time slots. To configure ranges, use hyphens. To configure discontinuous time slots, use commas. You can use a combination of ranges and discontinuous time slots, for example:

```
1,9-18,21
```

Example: Configuring NxDS0 Interfaces

Configure channelized NxDS0 interfaces, using partition 4 and time slots 1 through 10:

```
[edit interfaces]
ce1-0/2/0:1:2:3 {
  partition 4 interface-type ds0 timeslots 1-10;
}
```

This configuration creates interface **ds-0/2/0:1:2:4**.

Configuring Channelized OC12/STM4 IQ Interfaces (SDH Mode)

The Channelized OC12 IQ PIC configured for SDH mode creates a single channelized STM4 interface. You can configure this interface as unpartitioned using the **no-partition** statement at the **[edit interfaces cstm4-fpc/pic/port]** hierarchy level to create a single SDH VC-4-4C interface, or you can partition it into the following OC slices:

- SDH virtual concatenation 4 (VC-4) and channelized AU-4 interfaces (4 interfaces, any combination)
- Channelized T3 or T3 interfaces from a channelized AU-4 interface (3 interfaces, any combination)
- Channelized T1 or T1 interfaces from a channelized T3 interface (28 interfaces, any combination)
- NxDS0 interfaces from a channelized T1 interface



NOTE: If you configure the `per-unit-scheduler` statement on the physical interface of a 4-port channelized OC-12 IQ PIC and configure 975 logical interfaces or data link connection identifiers (DLCIs), some of the logical interfaces or DLCIs will drop all packets intermittently.

This section describes how to configure the following channelized OC12 IQ interfaces on a Channelized OC12 IQ PIC configured in SDH mode:

- Configuring Channelized OC12/STM4 IQ PICs for SDH Mode on page 82
- Configuring an Unpartitioned SDH (VC-4-4C) Interface on a Channelized OC12/STM4 IQ PIC on page 83
- Configuring SDH (VC-4) Interfaces on Channelized OC12/STM4 IQ PICs on page 83
- Configuring Channelized AU-4 Interfaces on page 84
- Configuring T3 or Channelized T3 Interfaces Under Channelized AU-4 Interfaces on page 85
- Configuring T1 or Channelized T1 Interfaces Under Channelized AU-4 Interfaces on page 85
- Configuring T1 or Channelized T1 Interfaces Under Channelized T3 Interfaces on page 86
- Configuring NxDS0 Interfaces on Channelized OC12/STM4 IQ PICs on page 87

Configuring Channelized OC12/STM4 IQ PICs for SDH Mode

To configure a Channelized OC12 IQ PIC to operate in SDH mode, include the **framing sdh** statement at the **[edit chassis fpc fpc/pic/port]** hierarchy level:

```
[edit chassis]
fpc 0 {
  pic 2 {
    framing sdh;
  }
}
```

This configuration creates interface **cstm4-0/2/0**.

For more information, see the *Junos OS System Basics Configuration Guide*.

Configuring an Unpartitioned SDH (VC-4-4C) Interface on a Channelized OC12/STM4 IQ PIC

On a Channelized OC12 IQ PIC, you can configure one SDH (VC-4-4C) interface. To configure an SDH (VC-4-4C) interface, include the **no-partition** and **interface-type** statements at the **[edit interfaces cstm4-fpc/pic/port]** hierarchy level:

```
[edit interfaces cstm4-fpc/pic/port]
no-partition interface-type so;
```

This configuration creates interface **so-fpc/pic/port**.

Example: Configuring an Unpartitioned SDH (VC-4-4C) Interface

Configure an unpartitioned SDH (VC-4-4C) interface, using partition 1 and OC slices 4 through 6:

```
[edit interfaces cstm4-0/2/0]
no-partition interface-type so;
```

This configuration creates the interface **so-0/2/0**.

Configuring SDH (VC-4) Interfaces on Channelized OC12/STM4 IQ PICs

To configure an SDH (VC-4) interface on a Channelized OC12 IQ PIC, include the **partition**, **oc-slice**, and **interface-type** statements at the **[edit interfaces cstm4-fpc/pic/port]** hierarchy level, specifying the **so** interface type:

```
[edit interfaces cstm4-fpc/pic/port]
partition partition-number oc-slice oc-slice-range interface-type so;
```

This configuration creates interface **so-fpc/pic/port:channel**.

The partition number is the sublevel interface partition index and is correlated with the channel number. For Channelized OC12 IQ PICs, the OC-slice range can be from 1 through 12.



NOTE: For channelized OC12 IQ interfaces, channel numbering begins with 1 (:1).

The OC-slice range is the range of SONET/SDH slices. For SDH interfaces, the OC-slice range specifies the bandwidth size required for the interface type you are configuring. SDH (VC-4) interfaces must occupy three consecutive OC slices per interface, in one of the following forms:

- 1–3
- 4–6
- 7–9
- 10–12

The interface type is the channelized interface type or data channel you are creating.

Example: Configuring SDH (VC-4) Interfaces

Configure SDH (VC-4) interfaces:

```
[edit interfaces cstm4-0/2/0]
partition 1 oc-slice 1-3 interface-type so;
partition 2 oc-slice 4-6 interface-type so;
partition 3 oc-slice 7-9 interface-type so;
partition 4 oc-slice 10-12 interface-type so;
```

This configuration creates the interfaces **so-0/2/0:1** through **so-0/2/0:4**.

Configuring Channelized AU-4 Interfaces

To configure a channelized AU-4 interface, include the **partition**, **oc-slice**, and **interface-type** statements at the **[edit interfaces cstm4-fpc/pic/port]** hierarchy level, specifying the **cau4** interface type:

```
[edit interfaces cstm4-fpc/pic/port]
partition partition-number oc-slice oc-slice-range interface-type cau4;
```

This configuration creates interface **cau4-fpc/pic/port:channel**.

The partition number is the sublevel interface partition index. For SDH interfaces, the partition number is not correlated with bandwidth size. A channelized STM-4 interface can have from 1 through 4 partition numbers.



NOTE: For channelized OC12 interfaces, channel numbering begins with 0 (:0). For channelized OC12 interfaces (both IQ and IQE), channel numbering begins with 1 (:1).

The OC-slice range is the range of SONET/SDH slices. For SDH interfaces, the OC-slice range specifies the bandwidth size required for the interface type you are configuring. Channelized AU-4 IQ interfaces must occupy three consecutive OC slices per interface, in one of the following forms:

- 1–3
- 4–6
- 7–9
- 10–12

The interface type is the channelized interface type or data channel you are creating.

Example: Configuring Channelized AU-4 Interfaces

Configure channelized AU-4 interfaces, using partitions 1 through 4:

```
[edit interfaces cstm4-0/2/0]
partition 1 oc-slice 1-3 interface-type cau4;
partition 2 oc-slice 4-6 interface-type cau4;
partition 3 oc-slice 7-9 interface-type cau4;
partition 4 oc-slice 10-12 interface-type cau4;
```


This configuration creates the interfaces **cau4-0/2/0:1** through **cau4-0/2/0:4**.

Configuring T3 or Channelized T3 Interfaces Under Channelized AU-4 Interfaces

To configure T3 or channelized T3 interfaces, include the **partition** and **interface-type** statements at the **[edit interfaces cau4-fpc/pic/port]** hierarchy level, specifying the **t3** or **ct3** interface type:

```
[edit interfaces]
cau4-fpc/pic/port {
  partition partition-number interface-type t3;
}
cau4-fpc/pic/port {
  partition partition-number interface-type ct3;
}
```

This configuration creates the interfaces **t3-fpc/pic/port:channel** and **ct3-fpc/pic/port:channel**.



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQ interfaces. You can apply CoS rules only to the aggregate bit streams.

Example: Configuring T3 or Channelized T3 Interfaces

Configure T3 and channelized T3 interfaces, using partition 1 and partition 2:

```
[edit interfaces]
cau4-0/2/0:1 {
  partition 1 interface-type t3;
}
cau4-0/2/0:1 {
  partition 2 interface-type ct3;
}
t3-0/2/0:1:1 ct3-0/2/0:1:2;
```

Configuring T1 or Channelized T1 Interfaces Under Channelized AU-4 Interfaces

To configure T1 or channelized T1 interfaces under channelized AU-4 interfaces, include the **partition** and **interface-type** statements at the **[edit interfaces cau4-fpc/pic/port]** hierarchy level, specifying the **t1** or **ct1** interface type:

```
[edit interfaces]
cau4-fpc/pic/port {
  partition partition-number interface-type t1;
}
cau4-fpc/pic/port {
  partition partition-number interface-type ct1;
}
```

This configuration creates the interfaces **t1-fpc/pic/port:channel** and **ct1-fpc/pic/port:channel**.



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQ interfaces. You can apply CoS rules only to the aggregate bit streams.

Example: Configuring T1 or Channelized T1 Interfaces Under Channelized AU-4 Interfaces

Configure T1 and channelized T1 interfaces, using partition 1 and partition 2:

```
[edit interfaces]
cau4-0/2/0:1 {
  partition 1 interface-type t1;
}
cau4-0/2/0:1 {
  partition 2 interface-type ct1;
}
t1-0/2/0:1:1 ct1-0/2/0:1:2;
```

Configuring T1 or Channelized T1 Interfaces Under Channelized T3 Interfaces

To configure T1 or channelized T1 interfaces under channelized T3 interfaces, include the **partition** and **interface-type** statements at the **[edit interfaces ct3-fpc/pic/port]** hierarchy level, specifying the **t1** or **ct1** interface type:

```
[edit interfaces]
ct3-fpc/pic/port {
  partition partition-number interface-type t1;
}
ct3-fpc/pic/port {
  partition partition-number interface-type ct1;
}
```

This configuration creates the interfaces **t1-fpc/pic/port:channel** and **ct1-fpc/pic/port:channel**.



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQ interfaces. You can apply CoS rules only to the aggregate bit streams.

Example: Configuring T1 or Channelized T1 Interfaces Under Channelized T3 Interfaces

Configure T1 or channelized T1 interfaces, using partition 3 and partition 4:

```
[edit interfaces]
ct3-0/2/0:1:2 {
  partition 3 interface-type t1;
}
ct3-0/2/0:1:2 {
  partition 4 interface-type ct1;
}
```

This configuration creates interfaces **t1-0/2/0:1:2:3** and **ct1-0/2/0:1:2:4**.

Configuring NxDS0 Interfaces on Channelized OC12/STM4 IQ PICs

Configure channelized NxDS0 IQ interfaces on the channelized T1 IQ interface by including the **partition**, **timeslots**, and **interface-type** statements at the **[edit interfaces ct1-fpc/pic/port:channel]** hierarchy level, specifying the **ds** interface type:

```
[edit interfaces ct1-fpc/pic/port:channel:channel]
partition partition-number timeslots time-slot-range interface-type ds;
```

This configuration creates the interface **ds-fpc/pic/port:channel**.

The time-slot range is from 1 through 24. You can designate any combination of time slots. To configure ranges, use hyphens. To configure discontinuous time slots, use commas. You can use a combination of ranges and discontinuous time slots:

```
1,9-18,21
```

Example: Configuring NxDS0 Interfaces

Configure channelized NxDS0 interfaces, using partition 4 and time slots 1 through 10:

```
[edit interfaces]
ct1-0/2/0:1:2:3 {
  partition 4 interface-type ds0 timeslots 1-10;
}
```

This configuration creates interface **ds-0/2/0:1:2:3:4**.

Configuring Channelized OC12 Interfaces

On Channelized OC12 PICs, you can configure 12 T3 channels per port. To configure channelized OC12 interface properties, you can include the **sonet-options** and **t3-options** statements at the **[edit interfaces interface-name]** hierarchy level. Some SONET/SDH options are ignored, and some can only be configured for channel 0, though they apply equally to all channels. The **long-buildout** statement under **t3-options** is also ignored.

For T3 channels on a channelized OC12 interface, the **clocking** statement is supported only for channel 0; it is ignored if included in the configuration of channels 1 through 11. The clock source configured for channel 0 applies to all channels on the channelized OC12 interface. The individual T3 channels use a gapped 45-MHz clock as the transmit clock. When you configure the clock source for a channelized interface—**ds-fpc/pic/port :0**, for example—you must also include the **channel-group** statement at the **[edit chassis]** hierarchy level and specify channel group 0. For more information, see “Clock Sources on Channelized Interfaces” on page 32.

For more information, see SONET/SDH Interfaces Overview and “T3 Interfaces Overview” on page 187. For a configuration example, see Configuring Aggregated SONET/SDH Interfaces.

Table 10 on page 88 summarizes the OC12-to-DS3 numbering scheme.

Table 10: OC12-to-DS3 Numbering Scheme

Two-Level STS-1 Number (STS-3,STS-1)	One-Level STS Number	OC12-to-DS3 PIC DS3 Number
1,1	1	0
1,2	2	1
1,3	3	2
2,1	4	3
2,2	5	4
2,3	6	5
3,1	7	6
3,2	8	7
3,3	9	8
4,1	10	9
4,2	11	10
4,3	12	11

Example: Configuring Channelized OC12 Interfaces

The following configuration is sufficient to get the channelized OC12 interface up and running. The OC12 interface can be divided into 12 channels. DS3 channels can use the following encapsulation types:

- PPP, PPP CCC, and PPP TCC
- Frame Relay, Frame Relay CCC, and Frame Relay TCC
- Cisco HDLC, Cisco HDLC CCC, and Cisco HDLC TCC

The channels can also have logical interfaces.

```
[edit interfaces]
t3-fpc/pic/port:0 {
  encapsulation cisco-hdlc;
  t3-options {
    compatibility-mode larscom;
    payload-scrambler;
  }
  unit 0 {
    family inet {
      address 10.11.30.1/30;
    }
  }
}
```

```

    }
    family iso;
  }
}
t3-fpc/pic/port:1 {
  encapsulation ppp;
  t3-options {
    compatibility-mode larscom;
    payload-scrambler;
  }
  unit 0 {
    family inet {
      address 10.11.30.5/30;
    }
    family iso;
  }
}
t3-fpc/pic/port:2 {
  encapsulation frame-relay;
  t3-options {
    compatibility-mode larscom;
    payload-scrambler;
  }
  unit 0 {
    dlci 100;
    family inet {
      address 10.11.30.9/30;
    }
    family iso;
  }
  unit 1 {
    dlci 101;
    family inet {
      address 10.11.31.9/30;
    }
    family iso;
  }
}
t3-lfpc/pic/port:3 {
  encapsulation cisco-hdlc-ccc;
  t3-options {
    compatibility-mode larscom;
    payload-scrambler;
  }
  unit 0;
}
t3-fpc/pic/port:4 {
  encapsulation ppp-ccc;
  t3-options {
    compatibility-mode larscom;
    payload-scrambler;
  }
  unit 0;
}
t3-fpc/pic/port:5 {
  dce;
}

```

```
encapsulation frame-relay-ccc;
t3-options {
    compatibility-mode larscom;
    payload-scrambler;
}
unit 0 {
    encapsulation frame-relay-ccc;
    dlci 1000;
}
unit 1 {
    encapsulation frame-relay-ccc;
    dlci 1001;
}
}
```

Configuring Link PIC Failover on Channelized OC12/STM4 IQ and IQE Interfaces

For Channelized OC12 IQ or IQE PICs used as linking PICs in redundant LSQ configurations, you can inhibit the router from sending PPP termination-request messages to the remote host if the link PIC fails. To do this, include the **no-termination-request** statement at the **[edit interfaces *interface-name* ppp-options]** hierarchy level:

```
no-termination-request;
```

The **no-termination-request** statement is supported only with MLPPP and SONET APS configurations and works with PPP, PPP over Frame Relay, and MLPPP interfaces only.

For information about interchassis and intrachassis LSQ failover, see the [Junos OS Services Interfaces Configuration Guide](#).

Example: Configuring a Channelized OC12 IQ Interface as an Unpartitioned Clear Channel

Configuring a SONET/SDH Interface

Configure a channelized OC12 interface as an unpartitioned, clear channel:

```
[edit interfaces]
coc12-5/0/0 {
    no-partition interface-type so; # so-5/0/0
}
```

Configuring Multiple Interface Types

Configure the following interfaces on a Channelized OC12 IQ or IQE PIC:

- An OC3 interface
- Another OC3 interface
- A channelized OC1 partitioned into T1 interfaces
- A channelized OC1 converted into a T3 interface
- A channelized OC1 partitioned into T1 interfaces and channelized T1s, which are partitioned into NxDS0 interfaces
- A channelized OC1 converted into a channelized T3, which is partitioned into T1 interfaces

- g. A channelized OC1 converted into a channelized T3, which is partitioned into T1 interfaces and a channelized T1, which is partitioned into NxDS0 interfaces
- h. A channelized OC1 partitioned into channelized T1s, which are partitioned into NxDS0 interfaces

Configuring the Interface Partitions

```
[edit interfaces]
coc12-1/1/0 {
  sonet-options {
    sonet-options-statements;
  }
  partition 1 oc-slice 1-3 interface-type so; # (a) so-1/1/0:1
  partition 2 oc-slice 4-6 interface-type so; # (b) so-1/1/0:2
  partition 3 oc-slice 7 interface-type coc1; # (c) coc1-1/1/0:3
  partition 4 oc-slice 8 interface-type coc1; # (d) coc1-1/1/0:5
  partition 5 oc-slice 9 interface-type coc1; # (e) coc1-1/1/0:5
  partition 6 oc-slice 10 interface-type coc1; # (f) coc1-1/1/0:6
  partition 7 oc-slice 11 interface-type coc1; # (g) coc1-1/1/0:7
  partition 8 oc-slice 12 interface-type coc1; # (h) coc1-1/1/0:8
}

(a) so-1/1/0:1 {
  description "(a) OC-slice 1-3 of coc12-1/1/0. COC12 > OC3;";
  sonet-options {
    sonet-options-statements;
  }
}

(b) so-1/1/0:2 {
  description "(b) OC-slice 4-6 of coc12-1/1/0. COC12 > OC3;";
  sonet-options {
    sonet-options-statements;
  }
}

(c) coc1-1/1/0:3 {
  description "(c) OC-slice 7 of coc12-1/1/0. COC12 to COC1 VT-mapped to T1s.";
  sonet-options {
    sonet-options-statements;
  }
  partition 1 - 10 interface-type t1; # t1-1/1/0:[1-10]
}
t1-1/1/0:3:1 {
  description "(c) OC-slice 7 of coc12-1/1/0. T1 interface configuration.";
  t1-options {
    t1-options-statements;
  }
}
...

(d) coc1-1/1/0:4 {
  description "(d) OC-slice 8 of coc12-1/1/0. COC12 to COC1 converted to a T3.";
  sonet-options {
    sonet-options-statements;
  }
}
```

```
no-partition interface-type t3; # t3-1/1/0:4
}
t3-1/1/0:4 {
  description "(d) OC-slice 8 of coc12-1/1/0. T3 interface configuration.";
}

(e) coc1-1/1/0:5 {
  description "(e) OC-slice 9 of coc12-1/1/0. COC12 to COC1 VT-mapped to T1s.";
  sonet-options {
    sonet-options-statements;
  }
  partition 1 - 3 interface-type t1; # t1-1/1/0:5:[1-3]
  partition 4 interface-type ct1; # ct1-1/1/0:5:4
}
t1-1/1/0:5:1 {
  description "(e) OC-slice 9 of coc12-1/1/0. T1 interface configuration.";
  t1-options {
    t1-options-statements;
  }
}
...
ct1-1/1/0:5:4 {
  description "(e) OC-slice 9 of coc12-1/1/0. CT1 to NxDSOs.;
  t1-options {
    t1-options-statements;
  }
  partition 1 timeslots 0 - 10 interface-type ds0; # ds-1/1/0:5:4:1
  partition 2 timeslots 11- 23 interface-type ds0; # ds-1/1/0:5:4:2
  ...
}

(f) coc1-1/1/0:6 {
  description "(f) OC-slice 10 of coc12-1/1/0. COC12 to COC1 converted to a CT3 to T1s.";
  sonet-options {
    sonet-options-statements;
  }
  no-partition interface-type ct3; # ct3-1/1/0:6
}
ct3-1/1/0:6 {
  description "(f) COC12 to CT3 M-13 and C-bit parity-mapped to T1s.;
  sonet-options {
    sonet-options-statements;
  }
  partition 1 - 10 interface-type t1; # t1-1/1/0:6:[1-10]
}
t1-1/1/0:6:1 {
  description "(f) T1 interface configuration.";
  t1-options {
    t1-options-statements;
  }
}
...

(g) coc1-1/1/0:7 {
```



```

description "(g) OC-slice 11 of coc12-1/1/0. COC12 to COC1 converted to a CT3 to T1s and
  CT1 to NxDSOs.";
sonet-options {
  sonet-options-statements;
}
no-partition interface-type ct3; # ct3-1/1/0:7
}
ct3-1/1/0:7 {
  description "(g) COC12 to CT3 M-13 and C-bit parity-mapped to T1s and CT1.";
  sonet-options {
    sonet-options-statements;
  }
  partition 1 - 10 interface-type t1; # t1-1/1/0:7:[1-10]
  partition 2 interface-type ct1; # ct1-1/1/0:7:11
}
t1-1/1/0:7:1 {
  description "(g) T1 interface configuration.";
  t1-options {
    t1-options-statements;
  }
}
...
ct1-1/1/0:7:11 {
  description "(g) CT1 to NxDSOs.";
  t1-options {
    t1-options-statements;
  }
  partition 1 timeslots 0 - 10 interface-type ds0; # ds-1/1/0:7:11:1
  partition 2 timeslots 11- 23 interface-type ds0; # ds-1/1/0:7:11:2
  ...
}

(h) coc1-1/1/0:8 {
  description "(h) OC-slice 12 of coc12-1/1/0. COC12 to COC1 VT-mapped to CT1 to NxDSOs.";
  sonet-options {
    sonet-options-statements;
  }
  partition 1 interface-type t1; # ct1-1/1/0:8:1
}
ct1-1/1/0:8:1 {
  description "(h) CT1 to NxDSOs.";
  t1-options {
    t1-options-statements;
  }
  partition 1 timeslots 0 - 10 interface-type ds0; # ds-1/1/0:8:1:1
  partition 2 timeslots 11- 23 interface-type ds0; # ds-1/1/0:8:1:2
  ...
}

```

For a full configuration example, see the *Junos OS Feature Guides*.

Example: Configuring Channelized OC12 Interfaces with Partitioned Channels

The following configuration is sufficient to get the channelized OC12 interface up and running. The OC12 interface can be divided into 12 channels. DS3 channels can use the following encapsulation types:

- PPP, PPP CCC, and PPP TCC
- Frame Relay, Frame Relay CCC, and Frame Relay TCC
- Cisco HDLC, Cisco HDLC CCC, and Cisco HDLC TCC

The channels can also have logical interfaces.

```
[edit interfaces]
t3-fpc/pic/port:0 {
  encapsulation cisco-hdlc;
  t3-options {
    compatibility-mode larscom;
    payload-scrambler;
  }
  unit 0 {
    family inet {
      address 10.11.30.1/30;
    }
    family iso;
  }
}
t3-fpc/pic/port:1 {
  encapsulation ppp;
  t3-options {
    compatibility-mode larscom;
    payload-scrambler;
  }
  unit 0 {
    family inet {
      address 10.11.30.5/30;
    }
    family iso;
  }
}
t3-fpc/pic/port:2 {
  encapsulation frame-relay;
  t3-options {
    compatibility-mode larscom;
    payload-scrambler;
  }
  unit 0 {
    dlci 100;
    family inet {
      address 10.11.30.9/30;
    }
    family iso;
  }
  unit 1 {
```

```
        dlci 101;
        family inet {
            address 10.11.31.9/30;
        }
        family iso;
    }
}
t3-1/fpc/pic/port:3 {
    encapsulation cisco-hdlc-ccc;
    t3-options {
        compatibility-mode larscom;
        payload-scrambler;
    }
    unit 0;
}
t3-fpc/pic/port:4 {
    encapsulation ppp-ccc;
    t3-options {
        compatibility-mode larscom;
        payload-scrambler;
    }
    unit 0;
}
t3-fpc/pic/port:5 {
    dce;
    encapsulation frame-relay-ccc;
    t3-options {
        compatibility-mode larscom;
        payload-scrambler;
    }
    unit 0 {
        encapsulation frame-relay-ccc;
        dlci 1000;
    }
    unit 1 {
        encapsulation frame-relay-ccc;
        dlci 1001;
    }
}
```


CHAPTER 5

Configuring Channelized OC3 IQ and IQE Interfaces

- Channelized OC3 IQ and IQE Overview on page 97
- Partitions, OC Slices, Interface Types, and Time Slots on page 98
- Configuring a Clear Channel on Channelized OC3 IQ and IQE PICs on page 99
- Configuring T3 Interfaces on IQ and IQE Interfaces on page 99
- Configuring T1 and NxDS0 Interfaces on page 100
- Configuring Fractional T1 IQ Interfaces on page 103
- Configuring Link PIC Failover on Channelized OC3 IQ and IQE Interfaces on page 104

Channelized OC3 IQ and IQE Overview

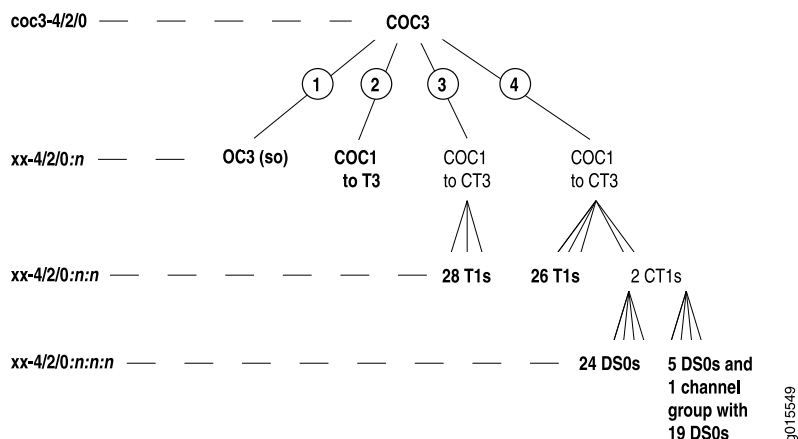
Channelized intelligent queuing (IQ) and channelized enhanced intelligent queuing (IQE) interfaces allow arbitrary and dynamic channelization of serial links, allowing greater flexibility than regular channelized interfaces.

On each port of a Channelized OC3 IQ or a Channelized OC3 IQE interface, you can configure the following interface types:

- One OC3 SONET interface
- Up to three T3 interfaces
- Up to 84 T1 interfaces
- Up to three E3 interfaces (COC3 IQE PICs in SDH mode)
- Up to 63 E1 interfaces (COC3 IQE PICs in SDH mode)
- Up to 336 NxDS0 interfaces on an M Series router
- Up to 768 NxDS0 interfaces on a T Series router

Figure 25 on page 98 shows an example of how a Channelized OC3 PIC might be partitioned. In the figure, the OC3 SONET interface would be a standalone interface because it would use the entire bandwidth of the PIC. The same applies to each port of the 2-port Channelized OC3 Enhanced IQ (IQE) PIC.

Figure 25: Channelized OC3 IQ Interface Example for Show Interfaces Controller



You can configure the following encapsulation types:

- PPP
- Frame Relay
- Cisco HDLC
- CCC
- TCC
- MPLS—On IQE interfaces.

For more information about interface encapsulation, see [Configuring Interface Encapsulation on Physical Interfaces](#) and [Configuring Interface Encapsulation on Logical Interfaces](#).

To configure channelized interfaces, include the following statements at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]
no-partition interface-type type;
partition partition-number oc-slice oc-slice-range interface-type type;
partition partition-number timeslots time-slot-range interface-type type;
```

Partitions, OC Slices, Interface Types, and Time Slots

The partition number is the sublevel interface partition index and is correlated with the channel number. For channelized OC3 interfaces, you can configure up to three OC1 interfaces, so the partition number can be 1, 2, or 3. For channelized T3 interfaces (**ct3**), you can configure multiple interfaces at once by including a partition range, such as 1-3. This creates three T1 interfaces with channel numbers 1 through 3.



NOTE: For channelized IQ and IQE interfaces, channel numbering begins with 1 (:1). For regular channelized interfaces, channel numbering begins with 0 (:0).

You configure the OC-slice range for SONET/SDH interfaces only. The OC-slice range is correlated with the bandwidth size required for the interface type you are configuring. For example, a channelized OC3 interface (**coc3**) can be divided into three OC1 interfaces, each containing one OC slice. Therefore the OC-slice value must be **1, 2, or 3**.

The configurable interface types are dependent on the hierarchy level at which you include the **interface-type** and **partition** or **no-partition** statements. For example, when you include the **no-partition** statement at the **[edit interfaces coc3-fpc/pic/port]** hierarchy level, the only configurable interface type is **so**, because the **no-partition** statement signals that you are creating a clear-channel SONET/SDH interface. When you include the **partition** statement at the **[edit interfaces coc1-fpc/pic/port]** hierarchy level, the configurable interface types are **ct1** or **t1**. If you want to create a T1 interface, include the **t1** option. If you want to further channelize down to the NxDS0 level, include the **ct1** option as an intermediate step before dividing the channelized T1 interface (**ct1**) into NxDS0 interfaces.

You configure time slots for fractional T1 interfaces and NxDS0 interfaces. You can configure ranges by using hyphens. You can configure discontinuous time slots by using commas. Do not include spaces.

Configuring a Clear Channel on Channelized OC3 IQ and IQE PICs

A *clear channel* is an interface that uses the entire bandwidth of the PIC. To configure a clear channel, include the **no-partition** and **interface-type** statements in the configuration.

On Channelized OC3 IQ and IQE PICs, you can configure one OC3 clear-channel interface per port. To configure an OC3 interface, include the **no-partition** and **interface-type** statements at the **[edit interfaces coc3-fpc/pic/port]** hierarchy level:

```
[edit interfaces coc3-fpc/pic/port]
no-partition interface-type so;
```

This configuration creates interface **so-fpc/pic/port**. When you include the **no-partition** statement at the **[edit interfaces coc3-fpc/pic/port]** hierarchy level, the only configurable interface type is **so**, because the **no-partition** statement signals that you are creating a clear-channel SONET/SDH interface.

On a 2-port or 4-port Channelized OC3 IQE PIC, you can configure two to four separate OC3 clear-channel interfaces by additionally specifying the port numbers. Configuration is otherwise the same as previously described on a (1-port) Channelized OC3 IQ PIC.

Configuring T3 Interfaces on IQ and IQE Interfaces

To configure a T3 interface on an OC3 PIC, include the **partition**, **oc-slice**, and **interface-type** statements at the **[edit interfaces coc3-fpc/pic/port]** hierarchy level, specifying the **coc1** interface type:

```
[edit interfaces coc3-fpc/pic/port]
partition partition-number oc-slice oc-slice-range interface-type coc1;
```

When you include the **partition** statement at the **[edit interfaces coc3-fpc/pic/port]** hierarchy level, the only configurable interface type is **coc1**. This configuration creates interface **coc1-fpc/pic/port:channel**.



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQ interfaces. You can only apply CoS rules to the aggregate bit streams.

Then, include the **no-partition interface-type** statement at the **[edit interfaces coc1-fpc/pic/port:channel]** hierarchy level, specifying the **t3** interface type:

```
[edit interfaces coc1-fpc/pic/port:channel]
no-partition interface-type t3;
```

This configuration creates interface **t3-fpc/pic/port:channel**.

Example: Configuring T3 Interfaces

Configure a T3 interface using partition 3 and OC slice 3. This configuration creates interface **t3-1/1/0:3**.

```
[edit interfaces coc3-1/1/0]
partition 3 oc-slice 3 interface-type coc1;
[edit interfaces coc1-1/1/0:3]
no-partition interface-type t3;
```

For a full configuration example, see the [Junos OS Feature Guides](#).

Configuring T1 and NxDS0 Interfaces

To configure T1 interfaces on a Channelized OC3 IQ or IQE PIC, perform the following tasks:

1. Partition the channelized OC3 interface into channelized OC1 interfaces by including the **partition**, **oc-slice**, and **interface-type** statements at the **[edit interfaces coc3-fpc/pic/port]** hierarchy level, specifying the **coc1** interface type:

```
[edit interfaces coc3-fpc/pic/port]
partition partition-number oc-slice oc-slice-range interface-type coc1;
```

2. If your network equipment uses VT mapping, partition the channelized OC1 interface into T1 interfaces by including the **partition** and **interface-type** statements at the **[edit interfaces coc1-fpc/pic/port:channel]** hierarchy level, specifying the **t1** interface type:

```
[edit interfaces coc1-fpc/pic/port:channel]
partition partition-number interface-type t1;
```

3. If your network equipment uses M13 or C-bit parity, convert the channelized OC1 interface into a channelized T3 interface by including the **no-partition** and **interface-type** statements at the **[edit interfaces coc1-fpc/pic/port:channel]** hierarchy level, specifying the **ct3** interface type:


```
[edit interfaces coc1-fpc/pic/port:channel]
no-partition partition-number interface-type ct3;
```



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQ interfaces. You can only apply CoS rules to the aggregate bit streams.

Note that because the **no-partition** statement is included, this configuration does not create another level of channelization, as denoted by the number of colons in the resulting interface.

4. To configure T1 interfaces, partition the channelized T3 interface into T1 interfaces by including the **partition** and **interface-type** statements at the **[edit interfaces ct3-fpc/pic/port:channel]** hierarchy level, specifying the **t1** interface type:

```
[edit interfaces ct3-fpc/pic/port:channel]
partition partition-number interface-type t1;
```

5. To configure NxDS0 interfaces, partition the channelized T3 interface into channelized T1 interfaces by including the **partition** and **interface-type** statements at the **[edit interfaces ct3-fpc/pic/port:channel]** hierarchy level and specifying the **ct1** interface type:

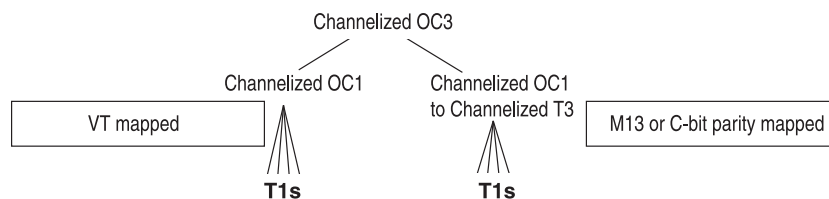
```
[edit interfaces ct3-fpc/pic/port:channel]
partition partition-number interface-type ct1;
```



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQ interfaces. You can only apply CoS rules to the aggregate bit streams.

Figure 26 on page 101 shows VT-mapped and M13 or C-bit parity-mapped configurations of T1 IQ interfaces.

Figure 26: T1 Interfaces on a Channelized OC3 PIC



Bold entries correspond to actual packet channels.

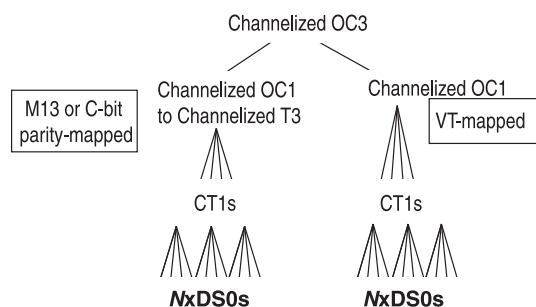
9015503

6. Configure channelized NxDS0 IQ interfaces on the channelized T1 IQ interface by including the **partition**, **timeslots**, and **interface-type** statements at the **[edit interfaces ct1-fpc/pic/port:channel]** hierarchy level, specifying the **ds** interface type:

```
[edit interfaces ct1-fpc/pic/port:channel]
partition partition-number timeslots time-slot-range interface-type ds;
```

Figure 27 on page 102 shows VT-mapped and M13 or C-bit parity-mapped configurations of NxDS0 IQ interfaces.

Figure 27: Sample Channelization of OC3 IQ or IQE PIC



Bold entries correspond to actual packet channels.

g015504

Example: Configuring T1 and NxDS0 Interfaces

Configure the following T1 interfaces:

```

t1-0/0/0:1:1
t1-0/0/0:1:2
t1-0/0/0:1:3
t1-0/0/0:1:4
t1-0/0/0:1:5

```

VT-Mapped Configuration

```

[edit interfaces coc3-0/0/0]
partition 1 oc-slice 1 interface-type coc1;
[edit interfaces coc1-0/0/0:1]
partition 1-5 interface-type t1;

```

M13 or C-bit Parity-Mapped Configuration

```

[edit interfaces coc3-0/0/0]
partition 1 oc-slice 1 interface-type coc1;
[edit interfaces coc1-0/0/0:1]
no-partition interface-type ct3;
[edit interfaces ct3-0/0/0:1]
partition 1-5 interface-type t1;

```

Configure the following two NxDS0 interfaces with 10 time slots and 4 time slots, respectively:

```

ds-0/0/0:1:2:1
ds-0/0/0:1:2:2

```

VT-Mapped Configuration

```

[edit interfaces coc3-0/0/0]
partition 1 oc-slice 1 interface-type coc1;
[edit interfaces coc1-0/0/0:1]
partition 2 interface-type ct1;
[edit interfaces ct1-0/0/0:1:2]
partition 1 timeslots 1-10 interface-type ds;
partition 2 timeslots 12-16 interface-type ds;

```

M13 or C-bit Parity-Mapped Configuration	<pre>[edit interfaces coc3-0/0/0] partition 1 oc-slice 1 interface-type coc1; [edit interfaces coc1-0/0/0:1] no-partition interface-type ct3; [edit interfaces ct3-0/0/0:1] partition 2 interface-type ct1; [edit interfaces ct1-0/0/0:1:2] partition 1 timeslots 1-10 interface-type ds; partition 2 timeslots 12-16 interface-type ds;</pre>
---	--

For a full configuration example, see the *Junos OS Feature Guides*.

Example: Setting Remote Loopback and Running BERT Tests on NxDSO Interfaces

For Channelized OC3 IQ and IQE PICs, if you need remote loopback on a far-end NxDSO interface, and you are running a BERT test from the local NxDSO interface, you must set remote loopback on the far-end router's associated channelized T1 interface (**ct1**). To do this, include the **loopback remote** statement at the **[edit interfaces ct1-fpc/pic/port t1-options]** hierarchy level. For example:

Local router:

```
[edit interfaces]
ct1-0/0/0:2:2 {
  partition 1 timeslots 1-10 interface-type ds;
  ds-0/0/0:2:2:1 {
    ds0-options {
      bert-period 30;
    }
  }
}
```

Remote router:

```
[edit interfaces]
ct1-0/0/0:2:2 {
  partition 1 timeslots 1-10 interface-type ds;
  t1-options {
    loopback remote;
  }
}
```

Configuring Fractional T1 IQ Interfaces

By default, all the time slots on a channelized T1 interface are used. To configure a fractional T1 interface on a Channelized OC3 IQ or IQE PIC, you must perform the following tasks:

1. Configure a T1 interface on the Channelized OC3 IQ or IQE PIC. For more information, see "Configuring T1 and NxDSO Interfaces" on page 100.
2. Configure the number of time slots allocated to the T1 IQ interface by including the **timeslots** statement at the **[edit interfaces t1-fpc/pic/port<:channel> t1-options]** hierarchy level:

```
[edit interfaces t1-fpc/pic/port<:channel> t1-options]
```

timeslots *time-slot-range*;

For channelized T1 IQ interfaces, the time-slot range is from 1 through 24. You can designate any combination of time slots. To configure ranges, use hyphens. To configure discontinuous time slots, use commas. Do not include spaces. For more information, see “Configuring Fractional T1 Time Slots” on page 185.

Example: Configuring Fractional T1 IQ Interfaces

Configure a fractional T1 interface that uses time slots 1 through 5 and 10:

```
[edit interfaces coc3-0/0/0]
partition 1 oc-slice 1 interface-type coc1;
[edit interfaces coc1-0/0/0:1]
partition 1 interface-type t1;
[edit interfaces t1-0/0/0:1 t1-options]
timeslots 1-5,10;
```

For a full configuration example, see the *Junos OS Feature Guides*.

Configuring Link PIC Failover on Channelized OC3 IQ and IQE Interfaces

For Channelized OC3 IQ or IQE PICs used as linking PICs in redundant LSQ configurations, you can inhibit the router from sending PPP termination-request messages to the remote host if the link PIC fails. To do this, include the **no-termination-request** statement at the **[edit interfaces *interface-name* ppp-options]** hierarchy level:

no-termination-request;

The **no-termination-request** statement is supported only with MLPPP and SONET APS configurations and works with PPP, PPP over Frame Relay, and MLPPP interfaces only.

For information about interchassis and intrachassis LSQ failover, see the *Junos OS Services Interfaces Configuration Guide*.

CHAPTER 6

Configuring Channelized STM1 Interfaces

- Channelized STM1 Interfaces Overview on page 105
- Configuring Channelized STM1 IQ and IQE Interfaces on page 105
- Configuring Channelized STM1 Interfaces on page 110
- Configuring Link PIC Failover on Channelized STM1 Interfaces on page 119
- Example: Configuring Channelized STM1 Interfaces on page 119

Channelized STM1 Interfaces Overview

Each Channelized STM1 PIC and Channelized STM1 Intelligent Queuing (IQ) PIC has one STM1 port.

For the Channelized STM1 IQ or IQE PIC, you can channelize the single port to the *NxDS0* level. Each E1 interface has 32 time slots (DS0), in which time slot 0 is reserved.

You can combine one or more of these DS0 time slots (channels) to create a channel group (*NxDS0*).

Configuring Channelized STM1 IQ and IQE Interfaces

This section includes the following topics:

- Configuring an STM1 IQ or STM1 IQE Interface on page 105
- Configuring E1 IQ and IQE Interfaces on page 106
- Configuring Fractional E1 IQ and IQE Interfaces on page 107
- Configuring an *NxDS0* IQ Interface on page 108
- Example: Configuring Channelized STM1 IQ and IQE Interfaces on page 109

Configuring an STM1 IQ or STM1 IQE Interface

On a one-port Channelized STM1 IQ PIC, or each individual port of the 4-port Channelized STM1 IQE PIC, you can configure one SDH STM1 interface. To configure an SDH STM1 interface, include the **no-partition interface-type** statement at the **[edit interfaces cstm1-*fpc/pic/port*]** hierarchy level, specifying the **so** interface type:

```
[edit interfaces cstm1-fpc/pic/port]  
no-partition interface-type so;
```

This configuration creates interface **so-fpc/pic/port**.



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQ and IQE interfaces. You can only apply CoS rules to the aggregate bit streams.

Configuring E1 IQ and IQE Interfaces

To configure an E1 interface on a Channelized STM1 IQ or IQE PIC, perform the following tasks:

1. Include the **no-partition** and **interface-type** statements at the **[edit interfaces cstm1-fpc/pic/port]** hierarchy level, specifying the **cau4** interface type. This converts the channelized STM1 interface into a channelized AU-4 interface. The resulting interface name is **cau4-fpc/pic/port**:

```
[edit interfaces cstm1-fpc/pic/port]
no-partition interface-type cau4;
```

2. Partition the channelized AU-4 interface into E1 interfaces by including the **partition** and **interface-type** statements at the **[edit interfaces cau4-fpc/pic/port]** hierarchy level, specifying the **e1** interface type. This configuration creates interface **e1-fpc/pic/port:channel**. The partition number is the sublevel interface partition index and is correlated with the channel number. For channelized E1 interfaces, the partition number can be from 1 through 63. The interface type is the channelized interface type or clear channel you are creating. For channelized AU-4 interfaces, **type** can be **ce1** or **e1**.

```
[edit interfaces cau4-fpc/pic/port]
partition partition-number interface-type e1;
```



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQ or IQE interfaces. You can only apply CoS rules to the aggregate bit streams.



NOTE: For channelized STM1 interfaces, channel numbering begins with 0 (:0). For channelized STM1 IQ and IQE interfaces, channel numbering begins with 1 (:1).

Example: Configuring E1 IQ and IQE Interfaces

Configure the following five E1 interfaces:

```
e1-0/0/0:1
e1-0/0/0:2
e1-0/0/0:3
e1-0/0/0:4
e1-0/0/0:5
```

```
[edit interfaces cstm1-0/0/0]
no-partition interface-type cau4;
[edit interfaces cau4-0/0/0]
partition 1-5 interface-type e1;
```

For a full configuration example, see the *Junos OS Feature Guides*.

Configuring Fractional E1 IQ and IQE Interfaces

By default, all the time slots on a channelized E1 interface are used. To configure a fractional E1 interface on a Channelized STM1 IQ or IQE PIC, perform the following tasks:

1. Include the **no-partition** and **interface-type** statements at the **[edit interfaces cstm1-fpc/pic/port]** hierarchy level, specifying the **cau4** interface type. This converts the channelized STM1 interface into a channelized AU-4 interface. The resulting interface name is **cau4-fpc/pic/port**:

```
[edit interfaces cstm1-fpc/pic/port]
no-partition interface-type cau4;
```

2. Partition the channelized AU-4 interface into E1 interfaces by including the **partition** and **interface-type** statements at the **[edit interfaces cau4-fpc/pic/port]** hierarchy level, specifying the **e1** interface type. The partition number is the sublevel interface partition index and is correlated with the channel number. For channelized E1 interfaces, the partition number can be from 1 through 63. The interface type is the channelized interface type or clear channel you are creating. For channelized AU-4 interfaces, **type** can be **ce1** or **e1**. This configuration creates interface **e1-fpc/pic/port:channel**:

```
[edit interfaces cau4-fpc/pic/port]
partition partition-number interface-type e1;
```

3. Configure the number of time slots allocated to the E1 IQ or IQE interface by including the **timeslots** statement at the **[edit interfaces e1-fpc/pic/port:channel e1-options]** hierarchy level. NxDS0 time slots configured on either a channelized STM1 IQ or IQE interface or channelized E1 IQ or IQE interface are numbered from 1 to 31 (0 is reserved), while fractional E1 time slots range from 2 to 32 (1 is reserved). To configure ranges, use hyphens. To configure discontinuous time slots, use commas. Do not include spaces.

```
[edit interfaces e1-fpc/pic/port:channel e1-options]
timeslots time-slot-range;
```



NOTE: For channelized STM1 interfaces, channel numbering begins with 0 (:0). For channelized STM1 IQ or IQE interfaces, channel numbering begins with 1 (:1).

For more information about E1 time slots, see “Configuring Fractional E1 Time Slots” on page 166.

Example: Configuring Fractional E1 Interfaces

Configure a fractional E1 interface that uses time slots 2 through 10:

```
[edit interfaces cstm1-0/0/0]
```

```
no-partition cau4;  
[edit interfaces cau4-0/0/0]  
partition 1 interface-type e1;  
[edit interfaces e1-0/0/0 e1-options]  
timeslots 2-10;
```

For a full configuration example, see the *Junos OS Feature Guides*.

Configuring an NxDSO IQ Interface

By default, all the time slots on a channelized STM1 interface are used. To configure an NxDSO IQ interface on a Channelized STM1 IQ or IQE PIC, perform the following tasks:

1. Include the **no-partition** and **interface-type** statements at the **[edit interfaces cstm1-fpc/pic/port]** hierarchy level, specifying the **cau4** interface type. This converts the channelized STM1 interface into a channelized AU-4 interface. The resulting interface name is **cau4-fpc/pic/port**:

```
[edit interfaces cstm1-fpc/pic/port]  
no-partition interface-type cau4;
```

2. Partition the channelized AU-4 interface into E1 interfaces by including the **partition** and **interface-type** statements at the **[edit interfaces cau4-fpc/pic/port]** hierarchy level, specifying the **ce1** interface type. This configuration creates interface **ce1-fpc/pic/port:channel**. The partition number is the sublevel interface partition index and is correlated with the channel number. For channelized E1 interfaces, the partition number can be from 1 through 63. The interface type is the channelized interface type or clear channel you are creating. For channelized AU-4 interfaces, **type** can be **ce1** or **e1**:

```
[edit interfaces cau4-fpc/pic/port]  
partition partition-number interface-type ce1;
```

3. Configure the number of time slots allocated to the NxDSO IQ interface by including the **partition**, **timeslots**, and **interface-type** statements at the **[edit interfaces e1-fpc/pic/port:channel]** hierarchy level, specifying the **ds** interface type. For channelized E1 IQ interfaces, the partition number range is from 1 through 31. For E1 IQ interfaces (**e1-fpc/pic/port**), the time-slot range is from 2 through 31. For channelized E1 IQ interfaces (**ce1-fpc/pic/port**), the time-slot range is from 1 through 31. You can designate any combination of time slots. To configure ranges, use hyphens. To configure discontinuous time slots, use commas. Do not include spaces:

```
[edit interfaces ce1-fpc/pic/port:channel]  
partition partition-number timeslots time-slot-range interface-type ds;
```



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQ and IQE interfaces. You can only apply CoS rules to the aggregate bit streams.



NOTE: For channelized STM1 interfaces, channel numbering begins with 0 (:0). For channelized STM1 IQ and IQE interfaces, channel numbering begins with 1 (:1).

For more information about E1 time slots, see “Configuring Fractional E1 Time Slots” on page 166.

Example: Configuring an NxDS0 IQ Interface

Configure an NxDS0 interface that uses time slots 1 through 10. This configuration creates the **ds-0/0/0:1:1** interface.

```
[edit interfaces cstm1-0/0/0]
no-partition interface-type cau4;
[edit interfaces cau4-0/0/0]
partition 1 interface-type ce1;
[edit interfaces ce1-0/0/0:1]
partition 1 timeslots 1-10 interface-type ds;
```

For a full configuration example, see the *Junos OS Feature Guides*.

Example: Configuring Channelized STM1 IQ and IQE Interfaces

Configure STM1, E1, fractional E1, and NxDS0 interfaces:

STM1 Interface	<pre>[edit interfaces] cstm1-0/0/0 { no-partition interface-type so; } so-0/0/0 { unit 0 { family inet { address 10.10.12.1/30; } } }</pre>
E1 Interface	<pre>[edit interfaces] cstm1-1/1/0 { no-partition interface-type cau4; } [edit interfaces] cau4-1/1/0 { partition 1-63 interface-type e1; } [edit interfaces] e1-1/1/0:1 { unit 0 { family inet { address 10.10.10.1/30; } } } ...</pre>

Fractional E1 Interface	<pre>[edit interfaces] cstm1-1/0/0 { no-partition interface-type cau4; } [edit interfaces] cau4-1/0/0 { partition 1-63 interface-type e1; } [edit interfaces] e1-1/1/0:1 { e1-options { timeslots 2-10; } unit 0 { family inet { address 10.10.10.1/30; } } } ...</pre>
DS0 Interface	<pre>[edit interfaces] cstm1-2/0/0 { no-partition interface-type cau4; } [edit interfaces] cau4-2/0/0 { partition 1-10 interface-type ce1; } [edit interfaces] ce1-2/0/0:1 { partition 1 interface-type ds timeslots 2-10; [edit interfaces] ds-2/0/0:1:1 { unit 0 { family inet { address 10.12.12.1/30; } } } } ...</pre>

For a full configuration example, see the [Junos OS Feature Guides](#).

Configuring Channelized STM1 Interfaces

To specify the channel number, include it after the colon (:) in the interface name. For example, a Channelized STM1-to-E1 PIC in FPC 1 and slot 1 will have the following physical interface, depending on the media type:

e1-1/1/0:x

The E1 channel number can be from 0 through 62.

This section contains the following topics:

- Configuring Channelized STM1 Interface Properties on page 111
- Configuring Virtual Tributary Mapping of Channelized STM1 Interfaces on page 112

Configuring Channelized STM1 Interface Properties

To configure the interface properties for Channelized STM1-to-E1 PICs, include the **e1-options** and **sonet-options** statements for both sides of the connection. The following configurations list all the valid statements.

To specify options for each of the E1 channels on the Channelized STM1-to-E1 PIC, include the **e1-options** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]
e1-options {
  bert-error-rate;
  bert-period;
  fcs (16 | 32);
  framing (g704 | g704-no-crc4 | unframed);
  idle-cycle-flag (flags | ones);
  loopback (local | remote);
  start-end-flag (filler | shared);
  timeslots time-slot-number;
}
```



NOTE: When a channelized STM1 interface experiences a line transition, the E1 channels configured in unframed mode log a large number of drops (around 24,000) as the channelized STM1 interface clocks resynchronize. This does not occur on framed channels, because the framing resynchronizes clocks very quickly.

To specify options for the SONET/SDH side of the connection, include the **sonet-options** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]
sonet-options {
  aps {
    advertise-interval milliseconds;
    authentication-key key;
    force;
    hold-time milliseconds;
    lockout;
    neighbor address;
    paired-group group-name;
    protect-circuit group-name;
    request;
    revert-time seconds;
    switching-mode (bidirectional | unidirectional);
    working-circuit group-name;
  }
  bytes {
    e1-quiet value;
  }
}
```

```

    f1 value;
    f2 value;
    s1 value;
    z3 value;
    z4 value;
  }
  loopback (local | remote);
}

```



NOTE: On channelized STM1 interfaces, you should configure the clock source on one side of the connection to be internal (the default Junos configuration) and on the other side of the connection to be external.

For information about Frame Relay DLCI limitations for channelized interfaces, see “Data-Link Connection Identifiers on Channelized Interfaces” on page 30. For more information about Frame Relay DLCIs, see “Configuring a Point-to-Point Frame Relay Connection” on page 212. For information about DLCI sparse mode, see the *Junos OS System Basics Configuration Guide*.

For more information about specific statements, see “E1 Interfaces Overview” on page 161, SONET/SDH Interfaces Overview, and “T1 Interfaces Overview” on page 177. For a configuration example, see “Example: Configuring Channelized STM1 Interfaces” on page 119.

Configuring Virtual Tributary Mapping of Channelized STM1 Interfaces

You can configure virtual tributary mapping to use KLM mode or ITU-T mode. To configure virtual tributary mapping, include the **vtmapping** statement at the **[edit chassis fpc slot-number pic pic-number]** hierarchy level:

```

[edit chassis fpc slot-number pic pic-number]
  vtmapping (klm | itu-t);

```

By default, virtual tributary mapping uses KLM mode. For more information, see the *Junos OS System Basics Configuration Guide*.

For the Channelized STM1 IQ and IQE PICs, you can configure virtual tributary mapping by including the **vtmapping** statement at the **[edit interfaces cau4-fpc/pic/port sonet-options]** hierarchy level:

```

[edit interfaces cau4-fpc/pic/port sonet-options]
  vtmapping (klm | itu-t);

```

Table 11 on page 112 lists the KLM mappings used by the channelized STM1-to-E1 PIC interfaces. The PIC defaults to KLM numbering with an offset of –1; for example, KLM 1= STM1 PIC 0.

Table 11: Channelized STM1-to-E1 Channel Mapping

Channel Number	KLM Number	Tributary Unit Group 3	Tributary Unit Group 2	Virtual Tributary	ITU-T Number
0	1	1	1	1	1

Table 11: Channelized STM1-to-E1 Channel Mapping (*continued*)

Channel Number	KLM Number	Tributary Unit Group 3	Tributary Unit Group 2	Virtual Tributary	ITU-T Number
1	2	1	1	2	22
2	3	1	1	3	43
3	4	1	2	1	4
4	5	1	2	2	25
5	6	1	2	3	46
6	7	1	3	1	7
7	8	1	3	2	28
8	9	1	3	3	49
9	10	1	4	1	10
10	11	1	4	2	31
11	12	1	4	3	52
12	13	1	5	1	13
13	14	1	5	2	34
14	15	1	5	3	55
15	16	1	6	1	16
16	17	1	6	2	37
17	18	1	6	3	58
18	19	1	7	1	19
19	20	1	7	2	40
20	21	1	7	3	61
21	22	2	1	1	2
22	23	2	1	2	23
23	24	2	1	3	44

Table 11: Channelized STM1-to-E1 Channel Mapping (*continued*)

Channel Number	KLM Number	Tributary Unit Group 3	Tributary Unit Group 2	Virtual Tributary	ITU-T Number
24	25	2	2	1	5
25	26	2	2	2	26
26	27	2	2	3	47
27	28	2	3	1	8
28	29	2	3	2	29
29	30	2	3	3	50
30	31	2	4	1	11
31	32	2	4	2	32
32	33	2	4	3	53
33	34	2	5	1	14
34	35	2	5	2	35
35	36	2	5	3	56
36	37	2	6	1	17
37	38	2	6	2	38
38	39	2	6	3	59
39	40	2	7	1	20
40	41	2	7	2	41
41	42	2	7	3	62
42	43	3	1	1	3
43	44	3	1	2	24
44	45	3	1	3	45
45	46	3	2	1	6
46	47	3	2	2	27

Table 11: Channelized STM1-to-E1 Channel Mapping (*continued*)

Channel Number	KLM Number	Tributary Unit Group 3	Tributary Unit Group 2	Virtual Tributary	ITU-T Number
47	48	3	2	3	48
48	49	3	3	1	9
49	50	3	3	2	30
50	51	3	3	3	51
51	52	3	4	1	12
52	53	3	4	2	33
53	54	3	4	3	54
54	55	3	5	1	15
55	56	3	5	2	36
56	57	3	5	3	57
57	58	3	6	1	18
58	59	3	6	2	39
59	60	3	6	3	60
60	61	3	7	1	21
61	62	3	7	2	42
62	63	3	7	3	63

Table 12 on page 115 lists the KLM mappings used by the channelized STM1-to-T1 PIC interfaces. The PIC defaults to KLM numbering with an offset of -1; for example, KLM 1= STM1 PIC 0.

Table 12: Channelized STM1-to-T1 Channel Mapping

Channel Number	KLM Number	Tributary Unit Group 3	Tributary Unit Group 2	Virtual Tributary	ITU-T Number
0	1	1	1	1	1
1	2	1	1	2	22
2	3	1	1	3	43

Table 12: Channelized STM1-to-T1 Channel Mapping (*continued*)

Channel Number	KLM Number	Tributary Unit Group 3	Tributary Unit Group 2	Virtual Tributary	ITU-T Number
3	4	1	2	1	4
4	5	1	2	2	25
5	6	1	2	3	46
6	7	1	3	1	7
7	8	1	3	2	28
8	9	1	3	3	49
9	10	1	4	1	10
10	11	1	4	2	31
11	12	1	4	3	52
12	13	1	5	1	13
13	14	1	5	2	34
14	15	1	5	3	55
15	16	1	6	1	16
16	17	1	6	2	37
17	18	1	6	3	58
18	19	1	7	1	19
19	20	1	7	2	40
20	21	1	7	3	61
21	22	2	1	1	2
22	23	2	1	2	23
23	24	2	1	3	44
24	25	2	2	1	5
25	26	2	2	2	26

Table 12: Channelized STM1-to-T1 Channel Mapping (*continued*)

Channel Number	KLM Number	Tributary Unit Group 3	Tributary Unit Group 2	Virtual Tributary	ITU-T Number
26	27	2	2	3	47
27	28	2	3	1	8
28	29	2	3	2	29
29	30	2	3	3	50
30	31	2	4	1	11
31	32	2	4	2	32
32	33	2	4	3	53
33	34	2	5	1	14
34	35	2	5	2	35
35	36	2	5	3	56
36	37	2	6	1	17
37	38	2	6	2	38
38	39	2	6	3	59
39	40	2	7	1	20
40	41	2	7	2	41
41	42	2	7	3	62
42	43	3	1	1	3
43	44	3	1	2	24
44	45	3	1	3	45
45	46	3	2	1	6
46	47	3	2	2	27
47	48	3	2	3	48
48	49	3	3	1	9

Table 12: Channelized STM1-to-T1 Channel Mapping (*continued*)

Channel Number	KLM Number	Tributary Unit Group 3	Tributary Unit Group 2	Virtual Tributary	ITU-T Number
49	50	3	3	2	30
50	51	3	3	3	51
51	52	3	4	1	12
52	53	3	4	2	33
53	54	3	4	3	54
54	55	3	5	1	15
55	56	3	5	2	36
56	57	3	5	3	57
57	58	3	6	1	18
58	59	3	6	2	39
59	60	3	6	3	60
60	61	3	7	1	21
61	62	3	7	2	42
62	63	3	7	3	63
63	64	4	1	1	24
64	65	4	1	2	45
65	66	4	1	3	66
66	67	4	2	1	27
67	68	4	2	2	48
68	69	4	2	3	69
69	70	4	3	1	30
70	71	4	3	2	51
71	72	4	3	3	72

Table 12: Channelized STM1-to-T1 Channel Mapping (*continued*)

Channel Number	KLM Number	Tributary Unit Group 3	Tributary Unit Group 2	Virtual Tributary	ITU-T Number
72	73	4	4	1	33
73	74	4	4	2	54
74	75	4	4	3	75
75	76	4	5	1	36
76	77	4	5	2	57
77	78	4	5	3	78
78	79	4	6	1	39
79	80	4	6	2	60
80	81	4	6	3	81
81	82	4	7	1	42
82	83	4	7	2	63
83	84	4	7	3	84

Configuring Link PIC Failover on Channelized STM1 Interfaces

For Channelized STM1 IQ and IQE PICs used as linking PICs in redundant LSQ configurations, you can inhibit the router from sending PPP termination-request messages to the remote host if the link PIC fails. To do this, include the **no-termination-request** statement at the **[edit interfaces *interface-name* ppp-options]** hierarchy level:

```
no-termination-request;
```

The **no-termination-request** statement is supported only with MLPPP and SONET APS configurations and works with PPP, PPP over Frame Relay, and MLPPP interfaces only.

For information about interchassis and intrachassis LSQ failover, see the [Junos OS Services Interfaces Configuration Guide](#).

Example: Configuring Channelized STM1 Interfaces

The following configuration is sufficient to get the Channelized STM1-to-E1 PIC interface up and running. The channelized STM1-to-E1 interface is an STM1 that is divided into 63 E1 interfaces. E1 interfaces can use the following encapsulation types:

- PPP, PPP CCC, and PPP TCC

- Frame Relay, Frame Relay CCC, and Frame Relay TCC
- Cisco HDLC, Cisco HDLC CCC, and Cisco HDLC TCC

The channels can also have logical interfaces. For information about Frame Relay DLCI limitations for channelized interfaces, see “Data-Link Connection Identifiers on Channelized Interfaces” on page 30. For more information about Frame Relay DLCIs, see “Configuring a Point-to-Point Frame Relay Connection” on page 212. For more information about DLCI sparse mode, see the *Junos OS System Basics Configuration Guide*.

You apply all STM1 interface SONET/SDH options to the first E1 interface in the configuration by including the **sonet-options** statement at the **[edit interfaces e1-fpc/pic/port:channel]** hierarchy level:

```
[edit]
interfaces {
  e1-0/0/1:0 {
    encapsulation cisco-hdlc;
    sonet-options {
      no-z0-increment;
    }
    e1-options {
      framing g704;
    }
    unit 0 {
      family inet {
        address 10.11.30.1/30;
      }
    }
  }
  e1-0/0/1:1 {
    encapsulation frame-relay;
    e1-options {
      framing g704;
    }
    unit 1 {
      dlci 16;
      family inet {
        address 10.11.31.9/30;
      }
    }
  }
  e1-0/0/1:2 {
    encapsulation ppp;
    no-keepalives;
    unit 0 {
      family inet {
        address 10.11.31.47/30;
      }
    }
  }
}
[edit]
chassis {
  fpc 2 {
```

```
pic 0 {  
    vtmapping klm;  
}  
}
```


CHAPTER 7

Configuring Channelized T3 Interfaces

- Configuring Channelized T3 IQ Interfaces on page 123
- Configuring Channelized DS3-to-DS0 Interfaces on page 126
- Configuring Channelized DS3-to-DS1 Interfaces on page 129
- Example: Configuring Channelized T3 IQ Interfaces on page 130
- Examples: Configuring Channelized DS3-to-DS0 Interfaces on page 131
- Examples: Configuring Channelized DS3-to-DS1 Interfaces on page 134

Configuring Channelized T3 IQ Interfaces



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQ interfaces. You can only apply CoS rules to the aggregate bit streams.

This section describes how to configure channelized T3 intelligent queuing (IQ) interfaces, discussing the following topics:

- Configuring T3 IQ Interfaces on page 123
- Configuring T1 IQ Interfaces on page 123
- Configuring Fractional T1 IQ and IQE Interfaces on page 124
- Configuring an NxDS0 IQ Interface on page 125

Configuring T3 IQ Interfaces

To configure a T3 interface, include the **no-partition** and **interface-type** statements at the **[edit interfaces ct3-fpc/pic/port]** hierarchy level:

```
[edit interfaces ct3-fpc/pic/port]
no-partition interface-type t3;
```

This configuration creates interface **t3-fpc/pic/port**.

Configuring T1 IQ Interfaces

On a Channelized DS3 IQ or IQE Physical Interface Card (PIC), you can create up to 112 T1 interfaces. To configure a T1 interface on a Channelized DS3 IQ or IQE PIC, include the

partition and **interface-type** statements at the **[edit interfaces ct3-*fpc/pic/port*]** hierarchy level, specifying the **t1** interface type:

```
[edit interfaces ct3-fpc/pic/port]  
partition partition-number interface-type t1;
```

This configuration creates interface **t1-*fpc/pic/port:channel***.

The partition number is the sublevel interface partition index and is correlated with the channel number. For channelized T3 interfaces, the partition number can be from 1 through 28.



NOTE: For channelized T3 interfaces, channel numbering begins with 0 (:0). For channelized T3 IQ and IQE interfaces, channel numbering begins with 1 (:1).

The interface type is the channelized interface type or clear channel you are creating. For channelized T3 interfaces, **type** can be **ct1** or **t1**.

Example: Configuring T1 IQ and IQE Interfaces

Configure the following five T1 interfaces:

```
t1-0/0/0:1  
t1-0/0/0:2  
t1-0/0/0:3  
t1-0/0/0:4  
t1-0/0/0:5
```

```
[edit interfaces ct3-0/0/0]  
partition 1-5 interface-type t1;
```

For a full configuration example, see the *Junos OS Feature Guides*.

Configuring Fractional T1 IQ and IQE Interfaces

By default, all the time slots on a channelized T1 interface are used. To configure a fractional T1 interface on a Channelized DS3 IQ or IQE PIC, perform the following tasks:

1. Configure a T1 IQ interface. For more information, see “Configuring T1 IQ Interfaces” on page 123.

This configuration creates interface **t1-*fpc/pic/port:channel***.

2. Configure the number of time slots allocated to the T1 IQ interface by including the **timeslots** statement at the **[edit interfaces t1-*fpc/pic/port:channel* t1-options]** hierarchy level:

```
[edit interfaces t1-fpc/pic/port t1-options]  
timeslots time-slot-range;
```

For channelized T1 IQ interfaces, the time-slot range is from 1 through 24. You can designate any combination of time slots. To configure ranges, use hyphens. To configure

discontinuous time slots, use commas. Do not include spaces. For more information about T1 time slots, see “Configuring Fractional T1 Time Slots” on page 185.

Example: Configuring Fractional T1 IQ Interfaces

Configure a fractional T1 interface that uses time slots 1 through 10:

```
[edit interfaces ct3-0/0/0:1]
partition 1 interface-type t1;
[edit interfaces t1-0/0/0:1:1 t1-options]
timeslots 1-10;
```

For a full configuration example, see the *Junos OS Feature Guides*.

Configuring an NxDS0 IQ Interface

By default, all the time slots on a channelized T3 interface are used. To configure an NxDS0 IQ interface on a Channelized DS3 IQ or IQE PIC, perform the following tasks:

1. Partition the channelized T3 interface into channelized T1 interfaces by including the **partition** and **interface-type** statements at the **[edit interfaces ct3-fpc/pic/port]** hierarchy level, specifying the **ct1** interface type:

```
[edit interfaces ct3-fpc/pic/port]
partition partition-number interface-type ct1;
```

This configuration creates interface **ct1-fpc/pic/port:channel**.

The partition number is the sublevel interface partition index and is correlated with the channel number. For channelized T1 interfaces, the partition number can be from 1 through 28.

The interface type is the channelized interface type or clear channel you are creating. For channelized T3 interfaces, **type** can be **ct1** or **t1**.



NOTE: For channelized T3 interfaces, channel numbering begins with 0 (:0). For channelized T3 IQ interfaces, channel numbering begins with 1 (:1).

2. Configure the number of time slots allocated to the NxDS0 IQ interface by including the **partition**, **timeslots**, and **interface-type** statements at the **[edit interfaces ct1-fpc/pic/port:channel]** hierarchy level, specifying the **ds** interface type:

```
[edit interfaces ct1-fpc/pic/port:channel]
partition partition-number timeslots time-slot-range interface-type ds;
```

For channelized T1 IQ interfaces, the partition number range is from 1 through 28; the time-slot range is from 1 through 24. You can designate any combination of time slots. To configure ranges, use hyphens. To configure discontinuous time slots, use commas. Do not include spaces. For more information about T1 time slots, see “Configuring Fractional T1 Time Slots” on page 185.

Example: Configuring an NxDS0 IQ Interface

Configure the following two NxDS0 interfaces with 10 time slots and 4 time slots, respectively:

```
ds-0/0/0:1:1
```

```
ds-0/0/0:1:2
```

```
[edit interfaces ct3-0/0/0]
partition 1 interface-type ct1;
[edit interfaces ct1-0/0/0:1]
partition 1 timeslots 1-10 interface-type ds;
partition 2 timeslots 12-16 interface-type ds;
```

For a full configuration example, see the *Junos OS Feature Guides*.

Configuring Channelized DS3-to-DS0 Interfaces

For channelized interfaces, you can configure 28 T1 channels per T3 interface. Each T1 link can have up to eight DS0 channel groups, and each channel group can hold any combination of DS0 time slots. To specify the T1 link and DS0 channel group number in the interface name, use colons (:) as separators. For example, a Multichannel DS3 PIC might have the following physical and virtual interfaces:

```
ds-0/0/0:x:y
```

where *x* is a T1 link ranging from 0 through 27 and *y* is a DS0 channel group from 0 through 7. For more information about ranges, see Table 13 on page 127.

You can use any of the values within the range available for *x* and *y*, and you do not have to configure the links sequentially. In addition, the Junos OS applies the interface options you configure according to the following rules:

- To configure the T1 options, you must set channel group *y* to 0; the T1 link *x* can be any value:

```
ds-0/0/0:x:0
```

- To configure the T3 options, you must set the T1 link *x* to 0 and channel group *y* to 0:

```
ds-0/0/0:0:0
```

- There are no restrictions on configuring the DS0 options.
- If you delete a configuration you previously committed for channel group 0, the options return to default values.

By default, all the time slots are used. To configure the channel groups and time slots for a channelized DS3-to-DS0 interface, include the **channel-group** and **timeslots** statements at the **[edit chassis fpc slot-number pic pic-number ct3 port port-number t1 link-number]** hierarchy level:

```
[edit chassis fpc slot-number pic pic-number ct3 port port-number t1 link-number ]
channel-group group-number;
timeslots time-slot-range;
```



NOTE: If you commit the interface name but do not include the `[edit chassis]` configuration, the channelized DS3-to-DS0 interface behaves like a channelized DS3-to-DS1 interface: none of the DS0 functionality is accessible.

Table 13 on page 127 shows the ranges you can specify for each of the elements in the preceding configuration.

Table 13: Ranges for Channelized DS3-to-DS0 Configuration

Item	Option	Range
FPC slot	<code>slot-number</code>	0 through 7 (see note below)
PIC slot	<code>pic-number</code>	0 through 3
Port	<code>port-number</code>	0 through 1
T1 link	<code>link-number</code>	0 through 27
DS0 channel group	<code>group-number</code>	0 through 7
Time slot	<code>time-slot-range</code>	1 through 24



NOTE: The FPC slot range depends on the router. For a routing matrix, the range is from 0 through 31. For M40, M40e, M160, M320, M120, and other T Series routers, the range is from 0 through 7. For M20 routers, the range is from 0 through 3. For M10 and M10i routers the range is from 0 through 1. For M5 and M7i routers, the only applicable value is 0.

Bandwidth limitations restrict the interface to a maximum of 128 channel groups per T3 port, rather than the theoretical maximum of $8 * 28 = 224$.

There are 24 time slots on a T1 interface. You can designate any combination of time slots. To configure ranges, use hyphens. To configure discontinuous time slots, use commas. Do not include spaces. You can use each time slot number on only one channel group within the same T1 link.

To configure channelized DS3-to-DS0 interface properties, you can include the **t3-options**, **t1-options**, and **ds0-options** statements. Only a subset of the T3 options are valid for this configuration, and the **buildout**, **invert-data**, and **line-encoding** statements at the `[edit interfaces interface-name t1-options]` hierarchy level are ignored. Likewise, only a subset of the DS0 options are valid for this configuration, and the **bert-algorithm**, **bert-error-rate**, **bert-period**, and **loopback payload** statements at the `[edit interfaces interface-name ds0-options]` hierarchy level are ignored. The following configurations list all the valid parameters.



NOTE: The set of options the Junos OS applies to the interface depends on how you specify the interface name. For more information, see “Examples: Configuring Channelized DS3-to-DS0 Interfaces” on page 131.

To specify options for the T3 side of the connection, include the **t3-options** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]  
t3-options {  
  bert-algorithm algorithm;  
  bert-error-rate rate;  
  bert-period seconds;  
  (cbit-parity | no-cbit-parity);  
  (long-buildout | no-long-buildout);  
  loopback (local | payload | remote);  
}
```

The statements at the **t3-options** hierarchy are supported only for channel 0; they are ignored if configured on other channels. To specify options for each of the T1 channels, include the **t1-options** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]  
t1-options {  
  byte-encoding (nx56 | nx64);  
  fcs (16 | 32);  
  framing (esf | lf);  
  idle-cycle-flag (flags | ones);  
  invert-data;  
  loopback (local | payload | remote);  
  start-end-flag (filler | shared);  
  timeslots time-slot-number;  
}
```

To specify options for each of the DS0 channels, include the **ds0-options** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]  
ds0-options {  
  bert-algorithm algorithm;  
  bert-error-rate rate;  
  bert-period seconds;  
  byte-encoding (nx56 | nx64);  
  fcs (16 | 32);  
  idle-cycle-flag (flags | ones);  
  invert-data;  
  loopback payload;  
  start-end-flag (filler | shared);  
}
```

For more information about specific parameters, see “E1 Interfaces Overview” on page 161, “E3 Interfaces Overview” on page 169, “T1 Interfaces Overview” on page 177, and “T3 Interfaces Overview” on page 187. For a configuration example, see “Examples: Configuring Channelized DS3-to-DS0 Interfaces” on page 131.

For information about Frame Relay DLCI limitations for channelized interfaces, see “Data-Link Connection Identifiers on Channelized Interfaces” on page 30. For more information about Frame Relay DLCIs, see “Configuring a Point-to-Point Frame Relay Connection” on page 212. For more information about DLCI sparse mode, see the *Junos OS System Basics Configuration Guide*.

Each T1 link can have up to eight DS0 channel groups, and each channel group can hold any combination of DS0 time slots.

Configuring Channelized DS3-to-DS1 Interfaces

You can configure 28 T1 channels per T3 interface, and each interface can have logical interfaces. To specify the channel number, include it after the colon (:) in the interface name. For example, a 4-port T3 PIC in FPC 1 and slot 1 will have the following physical interfaces, depending on the media type:

```
t1-1/1/0:x
t1-1/1/1:x
t1-1/1/2:x
t1-1/1/3:x
```

where *x* is a channel number ranging from 0 through 27.

To configure channelized DS3-to-DS1 interface properties, you can include both the **t1-options** and **t3-options** statements. Only a subset of the T3 options is valid for this configuration, and the **buildout**, **invert-data**, and **line-encoding** statements at the **[edit interfaces *interface-name* t1-options]** hierarchy level are ignored. Likewise, only a subset of the DS0 options are valid for this configuration, and the **bert-algorithm**, **bert-error-rate**, **bert-period**, and **loopback payload** statements at the **[edit interfaces *interface-name* ds0-options]** hierarchy level are ignored. The following configuration lists all the valid parameters.

To specify options for the T3 side of the connection, include the **t3-options** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]
t3-options {
  bert-algorithm algorithm;
  bert-error-rate rate;
  bert-period seconds;
  (cbit-parity | no-cbit-parity);
  (feac-loop-respond | no-feac-loop-respond);
  loopback (local | payload | remote);
}
```

The statements in the **t3-options** hierarchy are supported only for channel 0; they are ignored if configured on other channels.

To specify options for each of the T1 channels, include the **t1-options** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]
t1-options {
  byte-encoding (nx56 | nx64);
```

```

fcs (16 | 32);
framing (sf | esf);
idle-cycle-flag (flags | ones);
loopback (local | payload | remote);
start-end-flag (filler | shared);
timeslots time-slot-number;
}

```

For T1 channels on a channelized T3 interface, the **clocking** statement is supported only for channel 0; it is ignored if included in the configuration of channels 1 through 11. The clock source configured for channel 0 applies to all channels on the channelized T3 interface. The individual T1 channels use a gapped 45-MHz clock as the transmit clock. When you configure the clock source for a channelized interface—**ds-fpc/pic/port :0**, for example—you must also include the **channel-group** statement at the **[edit chassis]** hierarchy level, and specify channel group 0. For more information, see “Clock Sources on Channelized Interfaces” on page 32.

For information about Frame Relay DLCI limitations for channelized interfaces, see “Data-Link Connection Identifiers on Channelized Interfaces” on page 30. For more information about Frame Relay DLCIs, see “Configuring a Point-to-Point Frame Relay Connection” on page 212. For more information about DLCI sparse mode, see the [Junos OS System Basics Configuration Guide](#).

For more information about specific parameters, see “T1 Interfaces Overview” on page 177 and “T3 Interfaces Overview” on page 187. For a configuration example, see “Examples: Configuring Channelized DS3-to-DS1 Interfaces” on page 134.

Example: Configuring Channelized T3 IQ Interfaces

Configure a channelized T3 interface as an unpartitioned, clear channel.

Configuring a T3 Interface

```

[edit interfaces]
ct3-5/0/0 {
  no-partition interface-type t3;
}

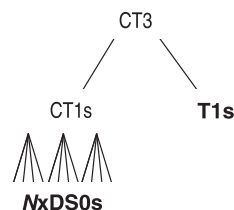
```

Configuring NxDS0 and T1 Interfaces

Figure 28 on page 130 shows the following interfaces on a Channelized DS3 IQ or IQE PIC:

- A channelized T1, which is partitioned into NxDS0 interfaces
- T1 interfaces

Figure 28: Sample Channelization of DS3 IQ or IQE PIC



Bold entries correspond to actual packet channels.

9003015

```

[edit interfaces]
ct3-1/1/0 {
  description "CT3 to CT1 and CT3 to T1.";
  t3-options {
    loopback remote;
    looptiming;
  }
  partition 1 interface-type ct1; # ct1-1/1/0:1.
  partition 2-28 interface-type t1; # t1-1/1/0:[2-28]
}
ct1-1/1/0:1 {
  description "case (a) CT1s to NxDSOs.";
  t1-options {
    bert-algorithm all-ones-repeating;
    framing sf;
    line-encoding ami;
  }
  partition 1 timeslots 2 - 10 interface-type ds0; # ds-1/1/0:1:1, channel group with 10 DS0s
  partition 2 timeslots 11- 23 interface-type ds0; # ds-1/1/0:1:2, channel group with 13 DS0s
  ...
}

```

Examples: Configuring Channelized DS3-to-DS0 Interfaces

The following configuration is sufficient to get the channelized DS3-to-DS0 interface up and running. The T3 interface can be divided into 28 channels, each at T1 line rate. DS3 channels can use the following encapsulation types for their logical interfaces:

- PPP, PPP CCC, and PPP TCC
- Frame Relay, Frame Relay CCC, and Frame Relay TCC
- Cisco HDLC, Cisco HDLC CCC, and Cisco HDLC TCC

For more information, see “Configuring a Point-to-Point Frame Relay Connection” on page 212.



NOTE: All these configuration examples specify channel group 0 in the interface address, which is required for configuring the **t3-options** and **t1-options** statements.

Configuring Cisco HDLC Encapsulation on a Channelized DS3-to-DS0 Interface

```

[edit interfaces]
ds-2/0/1:20:0 {
  encapsulation cisco-hdlc;
  unit 0 {
    family inet {
      address 10.0.4.40/32 {
        destination 10.0.4.41;
      }
    }
  }
}
[edit chassis]

```

```
fpc 2 {  
  pic 0 {  
    ct3 {  
      port 1 {  
        t1 20 {  
          channel-group 0 timeslots 1-5;  
        }  
      }  
    }  
  }  
}
```

**Configuring PPP
Encapsulation on a
Channelized
DS3-to-DS0 Interface**

```
[edit interfaces]  
ds-2/0/1:20:0 {  
  encapsulation ppp;  
  unit 0 {  
    family inet {  
      address 10.0.4.40/32 {  
        destination 10.0.4.41;  
      }  
    }  
  }  
}  
[edit chassis]  
fpc 2 {  
  pic 0 {  
    ct3 {  
      port 1 {  
        t1 20 {  
          channel-group 0 timeslots 1-5;  
        }  
      }  
    }  
  }  
}
```

**Configuring Three
Frame Relay DLCIs on
a Channelized DS3
Interface**

```
[edit interfaces]  
t1-5/1/3:0 {  
  mtu 9192;  
  encapsulation frame-relay;  
  unit 1 {  
    dlci 101;  
    family inet {  
      mtu 9000;  
      address 10.123.1.2/32 {  
        destination 10.123.1.1;  
      }  
    }  
    family iso {  
      mtu 9000;  
    }  
    family mpls {  
      mtu 9000;  
    }  
  }  
}
```



```

unit 2 {
  dlc1 102;
  family inet {
    mtu 9000;
    address 10.123.1.4/32 {
      destination 10.123.1.3;
    }
  }
  family iso {
    mtu 9000;
  }
  family mpls {
    mtu 9000;
  }
}
unit 3 {
  dlc1 103;
  family inet {
    mtu 9000;
    address 10.123.1.6/32 {
      destination 10.123.1.5;
    }
  }
  family iso {
    mtu 9000;
  }
  family mpls {
    mtu 9000;
  }
}
}

```

**Configuring Cisco
HDLC Encapsulation
with Byte-Encoding**

```

[edit interfaces ds-0/1/0:5:0]
no-keepalives;
encapsulation cisco-hdlc;
ds0-options {
  byte-encoding nx56;
}
unit 0 {
  family inet {
    address 10.221.2.8/24;
  }
}

```

**Configuring Cisco
HDLC Encapsulation
with Byte-Encoding
and Framing**

```

[edit interfaces ds-0/1/0:5:0]
no-keepalives;
encapsulation cisco-hdlc;
t1-options {
  byte-encoding nx56;
  framing sf;
}
unit 0 {
  family inet {
    address 10.221.2.8/24;
  }
}

```

	}
Use Time Slots 1 Through 10	[edit chassis fpc <i>slot-number</i> pic <i>pic-number</i> ct3 port <i>port-number</i> t1 <i>link-number</i>] channel-group <i>group-number</i> ; timeslots 1-10;
Use Time Slots 1 Through 5, 10, and 24	[edit chassis fpc <i>slot-number</i> pic <i>pic-number</i> ct3 port <i>port-number</i> t1 <i>link-number</i>] channel-group <i>group-number</i> ; timeslots 1-5,10,24;

Examples: Configuring Channelized DS3-to-DS1 Interfaces

The following configuration is sufficient to get the channelized DS3-to-DS1 interface up and running. The T3 interface can be divided into 28 channels, each at T1 line rate. DS3 channels can use the following encapsulation types for their logical interfaces:

- PPP, PPP CCC, and PPP TCC
- Frame Relay, Frame Relay CCC, and Frame Relay TCC
- Cisco HDLC, Cisco HDLC CCC, and Cisco HDLC TCC

For more information, see “Configuring a Point-to-Point Frame Relay Connection” on page 212.

Configuring Cisco HDLC Encapsulation on a Channelized DS3 Interface	[edit interfaces] t1-2/0/1:20 { encapsulation cisco-hdlc; unit 0 { family inet { address 10.0.4.40/32 { destination 10.0.4.41; } } } }
--	--

Configuring PPP Encapsulation on a Channelized DS3 Interface	[edit interfaces] t1-2/0/1:20 { encapsulation ppp; unit 0 { family inet { address 10.0.4.40/32 { destination 10.0.4.41; } } } }
---	---

Configuring Five Frame Relay DLCIs on a Channelized DS3 Interface	[edit interfaces] t1-5/1/3:0 { mtu 9192; encapsulation frame-relay; unit 1 { dlci 101; } }
--	---

```
family inet {
    mtu 9000;
    address 10.123.1.2/32 {
        destination 10.123.1.1;
    }
}
family iso {
    mtu 9000;
}
family mpls {
    mtu 9000;
}
}
unit 2 {
    dlci 102;
    family inet {
        mtu 9000;
        address 10.123.1.4/32 {
            destination 10.123.1.3;
        }
    }
    family iso {
        mtu 9000;
    }
    family mpls {
        mtu 9000;
    }
}
unit 3 {
    dlci 103;
    family inet {
        mtu 9000;
        address 10.123.1.6/32 {
            destination 10.123.1.5;
        }
    }
    family iso {
        mtu 9000;
    }
    family mpls {
        mtu 9000;
    }
}
unit 4 {
    dlci 104;
    family inet {
        mtu 9000;
        address 10.123.1.8/32 {
            destination 10.123.1.7;
        }
    }
    family iso {
        mtu 9000;
    }
    family mpls {
        mtu 9000;
    }
}
```

```
    }  
  }  
  unit 5 {  
    dlci 105;  
    family inet {  
      mtu 9000;  
      address 10.123.1.10/32 {  
        destination 10.123.1.9;  
      }  
    }  
    family iso {  
      mtu 9000;  
    }  
    family mpls {  
      mtu 9000;  
    }  
  }  
}
```

**Configuring Cisco
HDLC Encapsulation
with Byte-Encoding**

```
[edit interfaces t1-1/1/0:1]  
no-keepalives;  
encapsulation cisco-hdlc;  
t1-options {  
  byte-encoding nx56;  
}  
unit 0 {  
  family inet {  
    address 10.221.2.8/24;  
  }  
}
```

**Configuring Cisco
HDLC Encapsulation
with Byte-Encoding
and Framing**

```
[edit interfaces t1-1/1/0:1]  
no-keepalives;  
encapsulation cisco-hdlc;  
t1-options {  
  byte-encoding nx56;  
  framing sf;  
}  
unit 0 {  
  family inet {  
    address 10.221.2.8/24;  
  }  
}
```

CHAPTER 8

Configuring Channelized T1 Interfaces

- Channelized T1 IQ and IQE Interfaces Overview on page 137
- Configuring Channelized T1 IQ and IQE Interfaces on page 137
- Example: Configuring Channelized T1 IQ and IQE Interfaces on page 141

Channelized T1 IQ and IQE Interfaces Overview

The Channelized T1 intelligent queuing (IQ) and enhanced intelligent queuing (IQE) PICs have 10 T1 ports that you can channelize to the DS0 level. Each T1 interface has 24 DS0 time slots. You can combine DS0 time slots (channels) to create a channel group ($N \times \text{DS0}$).

The Channelized T1 IQ and IQE PICs are supported on the M7i, M10i, M20, M40e, M120, and M320 routers.

Configuring Channelized T1 IQ and IQE Interfaces

- Configuring T1 IQ and IQE Interfaces on page 137
- Configuring Fractional T1 IQ and IQE Interfaces on page 138
- Configuring $N \times \text{DS0}$ IQ and IQE Interfaces on page 139
- Configuring Payload Loopback on page 139
- Configuring Channelized T1 Interface Properties on page 140

Configuring T1 IQ and IQE Interfaces

To configure a T1 interface, include the **no-partition** and **interface-type** statements at the **[edit interfaces ct1-fpc/pic/port]** hierarchy level:

```
[edit interfaces ct1-fpc/pic/port]  
no-partition interface-type t1;
```

This configuration creates the interface **t1-fpc/pic/port**.



NOTE: For a T1 (t1-) interface configured on channelized T1 (ct1-) interface on a Channelized T1 IQ or IQE PIC, you can configure the following T1 options, but these options do not take effect for the T1 interface:

- bert-algorithm
- bert-error-rate
- bert-period
- buildout
- framing
- line-encoding
- loopback
- remote-loopback-respond

The T1 interface inherits these option settings from the parent channelized T1 interface.

Configuring Fractional T1 IQ and IQE Interfaces

By default, all the time slots on a channelized T1 interface are used. To configure a fractional T1 interface on a Channelized T1 IQ or IQE PIC, perform the following tasks:

1. Include the **no-partition** statement at the **[edit interfaces ct1-fpc/pic/port]** hierarchy level. This configuration creates the interface **t1-fpc/pic/port**.

```
[edit interfaces ct1-fpc/pic/port]
no-partition interface-type t1;
```

2. Configure the number of time slots allocated to the T1 IQ or IQE interface by including the **timeslots** statement at the **[edit interfaces t1-fpc/pic/port t1-options]** hierarchy level. DSO time slots configured on the channelized T1 IQ or IQE interface are numbered from 1 to 24. To configure ranges, use hyphens. To configure discontinuous time slots, use commas. Do not include spaces.

```
[edit interfaces t1-fpc/pic/port t1-options]
timeslots time-slot-range;
```

For more information about T1 time slots, see “Configuring Fractional T1 Time Slots” on page 185.

Example: Configuring Fractional T1 IQ and IQE Interfaces

Configure a fractional T1 interface that uses time slots 2 through 10:

```
[edit interfaces t1-0/0/0]
no-partition interface-type t1;
[edit interfaces t1-0/0/0 t1-options]
timeslots 1-10;
```

For a full configuration example, see the *Junos OS Feature Guides*.

Configuring NxDS0 IQ and IQE Interfaces

By default, all the time slots on a channelized T1 interface are used. To configure an NxDS0 IQ or IQE interface on a Channelized T1 IQ or IQE PIC, you must configure the number of time slots allocated to the NxDS0 IQ or IQE interface by including the **partition**, **timeslots**, and **interface-type** statements at the **[edit interfaces t1-fpc/pic/port]** hierarchy level, specifying the **ds** interface type:

```
[edit interfaces t1-fpc/pic/port]
  partition partition-number timeslots time-slot-range interface-type ds;
```

For channelized T1 IQ or IQE interfaces, the partition number range is from 1 through 24.

For channelized T1 IQ or IQE interfaces (**t1-fpc/pic/port**), the time-slot range is from 1 through 24. You can designate any combination of time slots. To configure ranges, use hyphens. To configure discontinuous time slots, use commas. Do not include spaces. For more information about T1 time slots, see “Configuring Fractional T1 Time Slots” on page 185.

Example: Configuring an NxDS0 IQ or IQE Interface

Configure an NxDS0 interface that uses time slots 2 through 10. This configuration creates the **ds-0/0/0:1** interface.

```
[edit interfaces t1-0/0/0:1]
  partition 1 timeslots 1-10 interface-type ds;
```

For a full configuration example, see the *Junos OS Feature Guides*.

Configuring Payload Loopback

Clocking and loopback options are configured at the controller level for all IQ-based and IQE-based interfaces. However, for the channelized T1 IQ or IQE interfaces, configure the payload loopback on the T1 interfaces instead of the channelized T1 IQ or IQE interface. To configure the payload option, include the **loopback payload** statement at the **[edit interfaces t1-fpc/pic/port t1-options loopback]** hierarchy level.

By default, all the time slots on a channelized T1 IQ or IQE interface are used. There can be a maximum of 24 channel groups per channelized T1 IQ or IQE interface. Thus, you can configure a maximum of 240 channel groups per PIC.

To specify the DS0 channel group number in the interface name, include a colon (:) as a separator. For example, a Channelized T1 IQ or IQE PIC might have the following physical and virtual interfaces:

```
ds-0/0/0:x
```

x is a DS0 channel group from 1 through 24 (for more information about ranges, see Table 14 on page 140).

You can use any of the values within the range available for **x**; you do not have to configure the links sequentially. In addition, the Junos OS applies the interface options you configure according to the following rules:

- To configure the **t1-options** statement, you must set channel group **x** to **0**:
`ds-0/0/0:0`
- There are no restrictions on configuring the **ds0-options** statement.
- If you delete a configuration you previously committed for channel group 0, the options return to default values.

To configure the channel groups and time slots for a channelized T1 IQ or IQE interface, include the following statements at the **[edit chassis]** hierarchy level:

```
[edit chassis]
fpc slot-number {
  pic pic-number {
    ct1 {
      t1 link-number {
        channel-group group-number;
        timeslots time-slot-range;
      }
    }
  }
}
```

There are 24 time slots on a T1 interface. You can designate any combination of time slots. To configure ranges, use hyphens. To configure discontinuous time slots, use commas. Do not include spaces.

Table 14 on page 140 shows the ranges you can specify.

Table 14: Ranges for Channelized T1 IQ Configuration

Item	Option	Range
FPC slot	<i>slot-number</i>	0 through 7
PIC slot	<i>pic-number</i>	0 through 3
T1 port	<i>port-number</i>	0 through 9
DS0 channel group	<i>partition</i>	1 through 24
Time slot	<i>time-slot-range</i>	1 through 24

The theoretical maximum number of channel groups possible per PIC is $10 * 24 = 240$. This is within the maximum bandwidth available.

Configuring Channelized T1 Interface Properties

To configure channelized T1 IQ or IQE interface properties, include the **t1-options** statement at the **[edit interfaces interface-name]** hierarchy level:

```
[edit interfaces interface-name]
t1-options {
  byte-encoding (nx56 | nx64)
```



```

fcs (16 | 32);
framing (esf | sf);
idle-cycle-flag (flags | ones);
invert-data;
line-encoding (ami | b8zs);
loopback (local | payload | remote);
start-end-flag (filler | shared);
}

```



NOTE: If you configure the line-encoding statement with the ami option and the byte-encoding statement with the nx64 option, excessive zeros in the payload area may bring the interface down. To prevent this, configure the byte-encoding statement with the nx56 option or include the invert-data statement.

To specify options for each of the DS0 channels, include the **ds0-options** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```

[edit interfaces interface-name]
ds0-options {
  byte-encoding (nx56 | nx64);
  fcs (16 | 32);
  idle-cycle-flag (flags | ones);
  loopback payload;
  start-end-flag (filler | shared);
}

```

Only a subset of the T1 options is valid for the channelized configuration; you specify the time slots using the **[edit chassis]** configuration described in Examples: Interface Naming. For more information about the T1 and DS0 options, see “T1 Interfaces Overview” on page 177.

Each T1 interface has 24 time slots (DSOs). You can combine one or more of these DS0 time slots (channels) to create a channel group (*N*xDSO). There can be a maximum of 24 channel groups per T1 interface.

Example: Configuring Channelized T1 IQ and IQE Interfaces

Configure a channelized T1 interface as an unpartitioned, clear channel.

Configuring a T1 Interface

```

[edit interfaces]
ct1-2/0/0 {
  no-partition interface-type t1; # t1-2/0/0
}

```

Configure a partitioned channel group.

Configuring a Channel Group

```

[edit interfaces]
ct1-0/0/1 {
  partition 1 interface-type ds0 timeslots 1-10;
  partition 2 interface-type ds0 timeslots 11-20;
}

```

The following configuration is sufficient to get the channelized T1 IQ or IQE interface up and running:

**Configuring Multiple
Interface Types**

```
[edit]
interfaces {
  ct1-1/2/3 {
    partition 1 timeslots 10 interface-type ds; # ds-1/2/3:1
    partition 2 timeslots 1-9 interface-type ds; # ds-1/2/3:2
  }
  ds-1/2/3:1 {
    unit 0 {
      family inet {
        address 10.25.1.2/24;
      }
    }
  }
  ds-1/2/3:2 {
    unit 0 {
      family inet {
        address 10.25.2.2/24;
      }
    }
  }
}
[edit]
interfaces {
  ct1-1/2/6 {
    no-partition interface-type t1; # t1-1/2/6
  }
  t1-1/2/6 {
    t1-options {
      timeslots 1-2;
    }
    unit 0 {
      family inet {
        address 10.255.126.2/24;
      }
    }
  }
}
```

CHAPTER 9

Configuring Channelized E1 Interfaces

- Channelized E1 IQ and IQE Interfaces Overview on page 143
- Configuring Channelized E1 IQ and IQE Interfaces on page 143
- Configuring Channelized E1 Interfaces on page 145
- Example: Configuring Channelized E1 IQ or IQE Interfaces on page 147
- Example: Configuring Channelized E1 Interfaces on page 148

Channelized E1 IQ and IQE Interfaces Overview

Each Channelized E1 PIC, Channelized E1 Intelligent Queuing (IQ) PIC and Channelized E1 Enhanced Intelligent Queuing (IQE) PIC has 10 E1 ports that you can channelize to the **NxDS0** level. Each E1 interface has 32 time slots (DS0), in which time slot 0 is reserved. You can combine one or more of these DS0 time slots (channels) to create a channel group **NxDS0**.

Configuring Channelized E1 IQ and IQE Interfaces

- Configuring E1 IQ and IQE Interfaces on page 143
- Configuring Fractional E1 IQ and IQE Interfaces on page 144
- Configuring NxDS0 IQ and IQE Interfaces on page 144



NOTE: Class-of-service (CoS) rules cannot be applied to an individual channel configured on channelized IQ and IQE interfaces. You can only apply CoS rules to the aggregate bit streams.

Configuring E1 IQ and IQE Interfaces

To configure an E1 interface, include the **no-partition** and **interface-type** statements at the **[edit interfaces ce1-fpc/pic/port]** hierarchy level:

```
[edit interfaces ce1-fpc/pic/port]  
no-partition interface-type e1;
```

This configuration creates interface **e1-fpc/pic/port**.

Configuring Fractional E1 IQ and IQE Interfaces

By default, all the time slots on a channelized E1 interface are used. To configure a fractional E1 interface on a Channelized E1 IQ PIC, perform the following tasks:

1. Include the **no-partition** statement at the **[edit interfaces ce1-fpc/pic/port]** hierarchy level:

```
[edit interfaces ce1-fpc/pic/port]
no-partition interface-type e1;
```

This configuration creates interface **e1-fpc/pic/port**.

2. Configure the number of time slots allocated to the E1 IQ or IQE interface by including the **timeslots** statement at the **[edit interfaces e1-fpc/pic/port e1-options]** hierarchy level:

```
[edit interfaces e1-fpc/pic/port e1-options]
timeslots time-slot-range;
```

NxDS0 time slots configured on either a channelized STM1 IQ or IQE interface or a channelized E1 IQ or IQE interface are numbered from 1 to 31 (0 is reserved), while fractional E1 time slots are numbered from 2 to 32 (1 is reserved).

To configure ranges, use hyphens. To configure discontinuous time slots, use commas. Do not include spaces.

For more information about E1 time slots, see “Configuring Fractional E1 Time Slots” on page 166.

Example: Configuring Fractional E1 IQ and IQE Interfaces

Configure a fractional E1 interface that uses time slots 2 through 10:

```
[edit interfaces ce1-0/0/0]
no-partition interface-type e1;
[edit interfaces e1-0/0/0 e1-options]
timeslots 2-10;
```

For a full configuration example, see the [Junos OS Feature Guides](#).

Configuring NxDS0 IQ and IQE Interfaces

By default, all the time slots on a channelized E1 interface are used. To configure an NxDS0 IQ interface on a Channelized E1 IQ or IQE PIC, you must configure the number of time slots allocated to the NxDS0 IQ or IQE interface by including the **partition**, **timeslots**, and **interface-type** statements at the **[edit interfaces ce1-fpc/pic/port]** hierarchy level, specifying the **ds** interface type:

```
[edit interfaces ce1-fpc/pic/port]
partition partition-number timeslots time-slot-range interface-type ds;
```

For channelized E1 IQ and IQE interfaces, the partition number range is from 1 through 31.

For E1 IQ and IQE interfaces (**e1-fpc/pic/port**), the time-slot range is from 2 through 31. For channelized E1 IQ and IQE interfaces (**ce1-fpc/pic/port**), the time-slot range is from

1 through 31. You can designate any combination of time slots. To configure ranges, use hyphens. To configure discontinuous time slots, use commas. Do not include spaces. For more information about E1 time slots, see “Configuring Fractional E1 Time Slots” on page 166.

Example: Configuring an NxDS0 IQ or IQE Interface

Configure an NxDS0 interface that uses time slots 2 through 10. This configuration creates the **ds-0/0/0:1** interface.

```
[edit interfaces ce1-0/0/0:1]
partition 1 timeslots 2-10 interface-type ds;
```

For a full configuration example, see the *Junos OS Feature Guides*.

Configuring Channelized E1 Interfaces

By default, all the time slots on a channelized E1 interface are used. There can be a maximum of 24 channel groups per channelized E1 interface. Thus, you can configure a maximum of 240 channel groups per PIC.

To specify the DS0 channel group number in the interface name, include a colon (:) as a separator. For example, a Channelized E1 PIC might have the following physical and virtual interfaces:

```
ds-0/0/0:x
```

where *x* is a DS0 channel group from 0 through 23 (for more information about ranges, see Table 15 on page 146).

You can use any of the values within the range available for *x*; you do not have to configure the links sequentially. In addition, the Junos OS applies the interface options you configure according to the following rules:

- To configure the **e1-options** statement, you must set channel group *x* to 0:
`ds-0/0/0:0`
- There are no restrictions on configuring the **ds0-options** statement.
- If you delete a configuration you previously committed for channel group 0, the options return to default values.

To configure the channel groups and time slots for a channelized E1 interface, include the following statements at the **[edit chassis]** hierarchy level:

```
[edit chassis]
fpc slot-number {
  pic pic-number {
    ce1 {
      e1 link-number {
        channel-group group-number;
        timeslots time-slot-range;
      }
    }
  }
}
```

}



NOTE: If you commit the interface name but do not include the `[edit chassis]` configuration, the Channelized E1 PIC behaves like a standard E1 PIC, and none of the DSO functionality is accessible.

There are 32 time slots on an E1 interface; however, time slot 0 is reserved. You can designate any combination of time slots. To configure ranges, use hyphens. To configure discontinuous time slots, use commas. Do not include spaces.

Table 15 on page 146 shows the ranges you can specify.

Table 15: Ranges for Channelized E1 Configuration

Item	Option	Range
FPC slot	<i>slot-number</i>	0 through 7 (see note below)
PIC slot	<i>pic-number</i>	0 through 3
E1 link	<i>link-number</i>	0 through 9
DSO channel group	<i>group-number</i>	0 through 23
Time slot	<i>time-slot-range</i>	0 through 31 (with time slot 0 reserved) (see note below)

The theoretical maximum number of channel groups possible per PIC is $10 * 24 = 240$. This is within the maximum bandwidth available.



NOTE: $N \times$ DSO time slots configured on either a channelized STM1 IQ or IQE interface or channelized E1 IQ or IQE interface are numbered from 1 to 31 (0 is reserved), while fractional E1 time slots range from 2 to 32 (1 is reserved).

The FPC slot range depends on the router. For a routing matrix, the range is from 0 through 31. For M40, M40e, M160, M320, M120, and other T Series routers, the range is from 0 through 7. For M20 routers, the range is from 0 through 3. For M10 and M10i routers, the range is from 0 through 1. For M5 and M7i routers, the only applicable value is 0.

Configuring Channelized E1 Interface Properties

To configure channelized E1 interface properties, include the **e1-options** statement at the `[edit interfaces interface-name]` hierarchy level:

```
[edit interfaces interface-name]
e1-options {
  fcs (16 | 32);
```

```

framing (g704 | g704-no-crc4 | unframed);
idle-cycle-flag (flags | ones);
loopback (local | remote);
start-end-flag (filler | shared);
}

```

To specify options for each of the DS0 channels, include the **ds0-options** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```

[edit interfaces interface-name]
ds0-options {
  byte-encoding (nx56 | nx64);
  fcs (16 | 32);
  idle-cycle-flag (flags | ones);
  loopback payload;
  start-end-flag (filler | shared);
}

```

For DS0 channels on a channelized E1 interface, the **clocking** statement is supported only for channel 0; it is ignored if included in the configuration of channels 1 through 11. The clock source configured for channel 0 applies to all channels on the channelized E1 interface. The individual DS0 channels use a gapped 45-MHz clock as the transmit clock. When you configure the clock source for a channelized interface—**ds-fpc/pic/port:0**, for example—you must also include the **channel-group** statement at the **[edit chassis]** hierarchy level, and specify channel group 0. For more information, see “Clock Sources on Channelized Interfaces” on page 32.

Only a subset of the E1 options is valid for the channelized configuration; you specify the time slots using the **[edit chassis]** configuration described in Examples: Interface Naming. For more information about the E1 and DS0 options, see “E1 Interfaces Overview” on page 161 “E1 Interfaces Overview” on page 161 and “T1 Interfaces Overview” on page 177.

Each E1 interface has 32 time slots (DS0s), in which time slot 0 is reserved. You can combine one or more of these DS0 time slots (channels) to create a channel group (NxDS0). There can be a maximum of 24 channel groups per E1 interface.

Example: Configuring Channelized E1 IQ or IQE Interfaces

Configure a channelized E1 interface as an unpartitioned, clear channel:

```

Configuring an E1 Interface [edit interfaces]
                             ce1-2/0/0 {
                               no-partition interface-type e1; # e1-2/0/0
                             }

```

The following configuration is sufficient to get the channelized E1 IQ or IQE interface up and running:

```

Configuring Multiple Interface Types [edit]
                                     interfaces {
                                       ce1-1/2/3 {
                                         partition 1 timeslots 10 interface-type ds; # ds-1/2/3:1
                                         partition 2 timeslots 1-9 interface-type ds; # ds-1/2/3:2
                                       }
                                       ds-1/2/3:1 {

```

```
    unit 0 {
      family inet {
        address 10.25.1.2/24;
      }
    }
  }
  ds-1/2/3:2 {
    unit 0 {
      family inet {
        address 10.25.2.2/24;
      }
    }
  }
}
[edit]
interfaces {
  ce1-1/2/6 {
    no-partition interface-type e1; # e1-1/2/6
  }
  e1-1/2/6 {
    e1-options {
      timeslots 1-2;
    }
    unit 0 {
      family inet {
        address 10.255.126.2/24;
      }
    }
  }
}
```

Example: Configuring Channelized E1 Interfaces

The following configuration is sufficient to get the channelized E1 interface up and running:

Configuring an E1 Interface, E1 Options, and DS0 Options

```
[edit chassis]
fpc 0 {
  pic 1 {
    ce1 {
      e1 0 {
        channel-group 0 timeslots 1;
        channel-group 1 timeslots 2;
        channel-group 5 timeslots 5-7;
      }
      e1 4 {
        channel-group 10 timeslots 11,17,28-31;
      }
    }
  }
}
[edit interfaces ds-0/1/0:0]
e1-options {
  fcs 32;
  framing g704-non-grc;
  loopback remote;
```



```

}
[edit interfaces ds-0/1/4:10]
ds0-options {
  byte-encoding nx56;
  start-end-flag filler;
}

```

The above configuration results in the following interfaces:

```

ds-0/1/0:1, with time slot 1 allocated
ds-0/1/0:5, with time slots 5 through 7 allocated
ds-0/1/4:10, with time slots 11, 17, and 28 through 31 allocated

```

The remaining ports (other than 0 and 4) remain as regular E1 interfaces (and follow the **e1-0/1/x** naming convention).

```

[edit chassis]
fpc 0 {
  pic 1 {
    ce1 {
      e1 0 {
        channel-group 1 timeslots 1;
        channel-group 5 timeslots 5-7;
      }
      e1 4 {
        channel-group 10 timeslots 11,17, 28-31;
      }
    }
  }
}

```

Use Time Slots 1 Through 10	<pre>[edit chassis fpc <i>slot-number</i> pic <i>pic-number</i> ce1 e1 <i>link-number</i>] channel-group <i>group-number</i>; timeslots 1-10;</pre>
--	---

Use Time Slots 1 Through 5, 10, and 24	<pre>[edit chassis fpc <i>slot-number</i> pic <i>pic-number</i> ce1 e1 <i>link-number</i>] channel-group <i>group-number</i>; timeslots 1-5,10,24;</pre>
---	--

CHAPTER 10

Configuring Channelized E1 PRI and T1 PRI Interfaces

- Channelized E1 PRI and T1 PRI Overview on page 151
- Configuring a Clear Channel on a Dual-Port Channelized T1/E1 PIM on page 152
- Configuring a Channelized T1/E1 Interface to Drop and Insert Time Slots on page 152
- Configuring Primary Rate Interfaces on page 153
- Allocating B-Channels for Dialout on page 155
- Configuring PRI Interfaces on page 155
- Example: Configuring a Channelized T1 Interface as Primary Rate Interface on page 156

Channelized E1 PRI and T1 PRI Overview

J Series Services Routers equipped with a Dual-Port Channelized T1/E1 PIM support Integrated Services Digital Network (ISDN) Primary Rate Interfaces (PRIs). ISDN PRI, referred to as S2M in Europe, is the “primary” extended ISDN network interface. It offers a larger capacity of digital channels utilizing a variety of improved mediums, and is used by large organizations with intensive communication needs. In contrast, the ISDN Basic Rate Interface (BRI), known as SO in Europe, provides a limited number of channels, transmitting over copper wire, and is used by smaller organizations or individuals with less intensive communication needs. For more information about configuring ISDN BRI interfaces, see *Configuring ISDN Physical Interface Properties*.

Unlike channelized PICs on the M Series and T Series routers, the interface type on the Dual-Port Channelized T1/E1 PIM is configurable. A single interface can operate as either a channelized T1 or channelized E1 interface (or clear channel) or as an ISDN PRI. The ISDN PRI channels can operate on the same interface as T1 or E1 channels. The PIM also supports a “drop-and-insert” feature, allowing you to insert channels from one port on the PIM into the other port on the PIM.

These ISDN channels are delivered to the user in one of two predefined configurations:

- ISDN BRI is configured by specifying properties for a physical (**br-**) interface and a logical (**dlr**) interface.
- For ISDN PRI, you configure:

1. Either a channelized E1 (**ce1-pim/0/port**) or channelized T1 (**ct1-pim/0/port**) interface.
2. Time slots within a **ce1-pim/0/port** interface or **ct1-pim/0/port** interface.
3. A bearer (B) channel **bc-pim/0/port:channel** interface for each time slot that you want to function as an ISDN PRI B-channel. The B-channel is used for data, video, voice, and multimedia. You can create up to 30 B-channels on a channelized E1 interface, and 23 B-channels on a channelized T1 interface.
4. One delta (D) channel, used between switching equipment in the ISDN network and the ISDN equipment at your site for signaling. For channelized E1, the D-channel must be time slot 16. For channelized T1, the D-channel must be time slot 24.



NOTE: Time slots can also be shared with **ds-pim/0/port** time slots within the same channelized interface.

Channelized E1 and T1 PIMs on J Series routers provide support for ISDN PRI connectivity for dial-in and callback, and for use as primary or backup network connections.

Configuring a Clear Channel on a Dual-Port Channelized T1-E1 PIM

A *clear channel* is an interface that uses the entire bandwidth of the port on a PIM. To configure a clear channel, include the **no-partition** and **interface-type** statements in the configuration. On a Dual-Port Channelized T1-E1 PIM, you can configure two clear-channel interfaces.

To configure an E1 interface, include the **no-partition** and **interface-type** statements at the **[edit interfaces ce1-pim/0/port]** hierarchy level:

```
[edit interfaces ce1-pim/0/port]
no-partition interface-type e1;
```

This configuration creates interface **e1-pim/0/port**.

To configure a T1 interface, include the **no-partition** and **interface-type** statements at the **[edit interfaces ct1-pim/0/port]** hierarchy level:

```
[edit interfaces ct1-pim/0/port]
no-partition interface-type t1;
```

This configuration creates interface **t1-pim/0/port**.

Configuring a Channelized T1/E1 Interface to Drop and Insert Time Slots

On channelized T1/E1 interfaces configured for channelized operation, you can insert channels (time slots) from one port (for example, channels carrying voice) directly into the other port on the PIM, to replace channels coming through the Routing Engine. This feature, known as drop and insert, allows you to integrate voice and data on a single T1 or E1 link by removing the DS0 time slots of one T1 or E1 port and replacing them by inserting the time slots of another T1 or E1 port. It is not necessary to use the same time

slots on both interfaces, but the time slots count must be same. The channels that are not configured for the drop-and-insert feature are used for normal traffic.

You can configure:

- 30 channelized E1 time slots, with the 16th time slot operating as the signaling channel
- 23 channelized T1 time slots, with the 24th time slot operating as the signaling channel

The signaling channel, or D-channel, must be part of the channels that are being switched through the drop-and-insert functionality. The Junos OS does not support switching of voice and data between ports by default.

Both ports involved in the drop-and-insert configuration must use the same clock source—either the router's internal clock or an external clock.

The following clock source settings are valid:

- When port 0 is set to use the internal clock, port 1 must also be set to use it, and vice versa.
- When port 0 is set to use its external clock, port 1 must be set to run on the same clock—the external clock for port 0.
- When port 1 is set to use its external clock, port 0 must be set to run on the same clock—the external clock for port 1.

For more details about valid clock combinations, see the *Junos OS Interfaces and Routing Configuration Guide*.

To configure drop-and-insert time slots on a channelized T1 interface, include the **partition** statement at the **[edit interfaces ct1-pim/0/port]** hierarchy level with the **timeslots** statement and **interface-type** statements specified:

```
[edit interfaces]
ct1-pim/0/port {
  partition 1 timeslots 1-10 interface-type ds;
  partition 2 timeslots 11-14 interface-type ds;
  partition 3 timeslots 15-32 interface-type ds;
}
```

This configuration creates interfaces **ds-pim/0/port:1**, **ds-pim/0/port:2**, and **ds-pim/0/port:3**.

Use the same configuration to create drop-and insert time slots on a channelized E1 interface by including the **partition** statement and options at the **[edit interfaces ce1-pim/0/port]** hierarchy level.

Configuring Primary Rate Interfaces

Primary rate interfaces are a combination of B-channels with one controlling D-channel for the group. Configure B-channel interfaces for each time slot that you want to function as an ISDN PRI interface. The B-channel is used for data, video, voice, and multimedia.

You can create:

- 23 B-channels on a channelized T1 interface
- 30 B-channels on a channelized E1 interface

To configure B-channels on a channelized T1 interface, include the **partition** statement at the **[edit interfaces ct1-pim/O/port]** hierarchy level with the **timeslots** statement and **interface-type bc** specified:

```
[edit interfaces]
ct1-pim/O/port {
  partition 1-23 timeslots 1-23 interface-type bc;
}
```

This configuration creates interfaces **bc-pim/O/port:1** through **bc-pim/O/port:1**, and **ds-pim/O/port:3**.

Use the same configuration to create B-channels on a channelized E1 interface by including the **partition** statement and options at the **[edit interfaces ce1-pim/O/port]** hierarchy level.

One D-channel is used between switching equipment in the ISDN network and the ISDN equipment at your site for signaling. For channelized E1, the D-channel must be time slot 16. For channelized T1, the D-channel must be time slot 24.

To configure a D-channel on a channelized T1 interface, include the **partition** statement at the **[edit interfaces ct1-pim/O/port]** hierarchy level with the **timeslots** statement and **interface-type dc** specified:

```
[edit interfaces]
ct1-pim/O/port {
  partition 24 timeslots 24 interface-type dc;
}
```

This configuration creates interfaces **dc-pim/O/port**.

Use the same configuration to create B-channels on a channelized E1 interface by including the **partition** statement and options at the **[edit interfaces ce1-pim/O/port]** hierarchy level.

```
[edit interfaces]
ce1-pim/O/port {
  partition 16 timeslots 16 interface-type dc;
}
```

To view PRI or ISDN options information about interface, use the following operational mode commands supporting BRI interfaces:

- **show interfaces *interface-name* detail**
- **show interface *dln***
- **show isdn calls**
- **show isdn history**
- **show isdn q921 statistics**
- **show isdn q931 statistics**
- **show isdn status**



NOTE: You must configure a D-channel and B-channels to complete your ISDN PRI line configuration.



NOTE: You can configure `dso-options` on the B-channel, but you cannot configure parameters for a D-channel. However, when interface statistics are displayed, both B-channel and D-channel interfaces have statistical values.

Allocating B-Channels for Dialout

You can configure the system to allocate B-channels for dialout from lowest or highest numbered B-channel (ascending or descending order). By configuring this feature, you reduce chances of “glare” on PRI lines carrying a mix of incoming and outgoing calls.

To configure the B-channel allocation, include the `idsn-options` and `bchannel-allocation` statements at the `[edit interfaces ct1-pim/O/port | ce1-pim/O/port]` hierarchy level:

```
[edit interfaces]
(ct1-pim/O/port | ce1-pim/O/port) {
  isdn-options {
    (bchannel-allocation (ascending | descending);
  }
}
```

Configuring PRI Interfaces

When you create a PRI from a channelized E1 or channelized T1 interface, you can select all the slots for the PRI, or just a few of them, leaving the rest as `ds-` interfaces.

To configure a PRI from a channelized T1 interface, include the `partition` statement at the `[edit interfaces ct1-pim/O/port]` hierarchy level with the `timeslots` statement and `interface-type bc` specified:

```
[edit interfaces]
ct1-pim/O/port {
  partition 1 timeslots 1-10 interface-type ds;
  partition 2 timeslots 11-24 interface-type pr;
}
```

This configuration creates interfaces `ds-pim/O/port:1` through `pr-pim/O/port:2`.

Use the same configuration to create interfaces on a channelized E1 interface by including the `partition` statement and options at the `[edit interfaces ce1-pim/O/port]` hierarchy level.

To configure channelized E1 interface properties, include the `e1-options` statement at the `[edit interfaces interface-name]` hierarchy level:

```
[edit interfaces interface-name]
e1-options {
```

```
fcs (16 | 32);
framing (g704 | g704-no-crc4 | unframed);
idle-cycle-flag (flags | ones);
loopback (local | remote);
start-end-flag (filler | shared);
}
```

To specify options for each of the DS0 channels, include the **ds0-options** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]
ds0-options {
  byte-encoding (nx56 | nx64);
  fcs (16 | 32);
  idle-cycle-flag (flags | ones);
  loopback payload;
  start-end-flag (filler | shared);
}
```

Example: Configuring a Channelized T1 Interface as Primary Rate Interface

Configure a channelized T1 interface to operate fully as a PRI:

```
[edit interfaces]
ct1-2/0/0 {
  partition 1-23 timeslots 1-23 interface-type bc;
  partition 24 timeslots 24 interface-type dc;
  t1-options {
    line-encoding b8zs;
    framing esf;
  }
  traceoptions {
    flag q931;
    flag q921;
    file {
      pri_trace_log;
    }
  }
  dialer-options {
    pool 1 priority 25;
  }
  isdn-options {
    switch-type att5e;
    bchannel-allocation descending;
    incoming-called-number 384101;
    incoming-called-number 384102;
    incoming-called-number 384103;
  }
}

[edit interfaces]
d10 {
  unit 0 {
    dialer-options {
      pool 1;
      dial-string 384010;
    }
  }
}
```



```
        incoming-map {
            accept-all;
        }
    }
    family inet {
        filter {
            dialer int-packet;
        }
        address 13.1.1.2/24;
    }
}
[edit firewall]
family inet {
    dialer-filter int-packet {
        term term1 {
            from {
                destination address {
                    13.1.1.1/24;
                }
                protocol icmp;
                then note;
            }
        }
        term term2 {
            then ignore;
        }
    }
}
```


PART 3

Configuring E1, E3, T1, and T3 Interfaces

- [Configuring E1 Interfaces on page 161](#)
- [Configuring E3 Interfaces on page 169](#)
- [Configuring T1 Interfaces on page 177](#)
- [Configuring T3 Interfaces on page 187](#)

CHAPTER 11

Configuring E1 Interfaces

- E1 Interfaces Overview on page 161
- Configuring E1 Physical Interface Properties on page 162
- Configuring E1 BERT Properties on page 162
- Configuring the E1 Frame Checksum on page 163
- Configuring E1 Framing on page 163
- Configuring the E1 Idle Cycle Flag on page 164
- Configuring E1 Data Inversion on page 164
- Configuring E1 Loopback Capability on page 164
- Configuring E1 Start and End Flags on page 166
- Configuring Fractional E1 Time Slots on page 166

E1 Interfaces Overview

E1 is a standard WAN digital communication format designed to operate over copper facilities at a rate of 2.048 Mbps. Widely used outside North America, it is a basic time-division multiplexing scheme used to carry digital circuits. The following standards apply to E1 interfaces:

- ITU-T Recommendation G.703, *Physical/electrical characteristics of hierarchical digital interfaces*, describes data rates and multiplexing schemes for the E Series.
- ITU-T Recommendation G.751, *General Aspects of Digital Transmission Systems: Terminal Equipment*, describes framing methods.
- ITU-T Recommendation G.775, *Loss of Signal (LOS) and Alarm Indication Signal (AIS) Defect Detection and Clearance Criteria*, describes alarm reporting methods.



NOTE: The Juniper Networks E1 Physical Interface Card (PIC) does not support Channel Associated Signaling (CAS).

Configuring E1 Physical Interface Properties

To configure E1-specific physical interface properties, include the **e1-options** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]  
e1-options {  
  bert-error-rate rate;  
  bert-period seconds;  
  fcs (16 | 32);  
  framing (g704 | g704-no-crc4 | unframed);  
  idle-cycle-flag (flags | ones);  
  invert-data;  
  loopback (local | remote);  
  start-end-flag (filler | shared);  
  timeslots time-slot-range;  
}
```

Configuring E1 BERT Properties

This section discusses BERT properties for the E1 interface specifically. For general information about the Junos implementation of the BERT procedure, see Interface Diagnostics.

You can configure an E1 interface or a CE1 or E1 partition on a channelized PIC to execute a bit error rate test (BERT) when the interface receives a request to run this test. You specify the duration of the test and the error rate to include in the bit stream by including the **bert-period** and **bert-error-rate** statements at the **[edit interfaces *interface-name* e1-options]** hierarchy level:

```
[edit interfaces interface-name e1-options]  
bert-error-rate rate;  
bert-period seconds;
```

By default, the BERT period is 10 seconds. You can configure the BERT period to last from 1 through 239 seconds on some PICs and from 1 through 240 seconds on other PICs. Standard CE1, standard E1, E1 IQ, and E1 IQE interfaces, and PICs partitioned to CE1 and E1 channels, support an extended BERT period range, up to 86,400 seconds (24 hours), and have a default BERT period value of 240 seconds.



NOTE: When configuring E1 and CE1 interfaces on 10-port Channelized E1/T1 IQE PICs, the **bert-period** statement must be included at the **[edit interfaces *ce1-fpc/pic/port*]** hierarchy level.

rate is the bit error rate. This can be an integer from 0 through 7, which corresponds to a bit error rate from 10^{-0} (0, which corresponds to no errors) to 10^{-7} (1 error per 10 million bits). The default is 0.

Individual concatenated E1 interfaces do not support the **bert-algorithm** configuration statement. For individual concatenated E1 interfaces, the **bert-algorithm** statement at

the `[edit interfaces interface-name e1-options]` hierarchy level is ignored. The algorithm for the E1 BERT procedure is **pseudo-2e15-o151** (pattern is $2^{15}-1$, as defined in the CCITT/ITU O.151 standard).

For channelized E1 intelligent queuing (IQ and IQE) interfaces, you can configure the BERT algorithm by including the **bert-algorithm** statement at the `[edit interfaces ce1-fpc/pic/port e1-options]` or `[edit interfaces e1-fpc/pic/port e1-options]` hierarchy level:

```
[edit interfaces ce1-fpc/pic/port e1-options]
bert-algorithm algorithm;
[edit interfaces e1-fpc/pic/port e1-options]
bert-algorithm algorithm;
```

For a list of supported algorithms, enter a **?** after the **bert-algorithm** statement; for example:

```
[edit interfaces ce1-0/0/0 e1-options]
user@host# set bert-algorithm ?
Possible completions:
pseudo-2e11-o152 Pattern is 2^11 -1 (per O.152 standard)
pseudo-2e15-o151 Pattern is 2^15 - 1 (per O.152 standard)
pseudo-2e20-o151 Pattern is 2^20 - 1 (per O.151 standard)
pseudo-2e20-o153 Pattern is 2^20 - 1 (per O.153 standard)
```

Configuring the E1 Frame Checksum

By default, the E1 interface supports a 16-bit checksum. You can configure a 32-bit checksum, which provides more reliable packet verification. However, some older equipment might not support 32-bit checksums.

To configure a 32-bit checksum, include the **fcs 32** statement at the `[edit interfaces interface-name e1-options]` hierarchy level:

```
[edit interfaces interface-name e1-options]
fcs 32;
```

To return to the default 16-bit frame checksum, delete the **fcs 32** statement from the configuration:

```
[edit]
user@host# delete interfaces e1-fpc/pic/port e1-options fcs 32
```

To explicitly configure a 16-bit checksum, include the **fcs 16** statement at the `[edit interfaces interface-name e1-options]` hierarchy level:

```
[edit interfaces interface-name e1-options]
fcs 16;
```

Configuring E1 Framing

By default, E1 interfaces use the G704 framing mode. You can configure the alternative unframed mode if needed.

To have the interface use the unframed mode, include the **framing** statement at the `[edit interfaces interface-name e1-options]` hierarchy level, specifying the **unframed** option:

```
[edit interfaces interface-name e1-options]  
framing unframed;
```

To explicitly configure G704 framing, include the **framing** statement at the **[edit interfaces *interface-name* e1-options]** hierarchy level, specifying the **g704** option:

```
[edit interfaces interface-name e1-options]  
framing g704;
```

By default, G704 framing uses CRC4. To explicitly configure an interface's G704 framing to not use CRC4, include the **framing** statement at the **[edit interfaces *interface-name* e1-options]** hierarchy level, specifying the **g704-no-crc4** option:

```
[edit interfaces interface-name e1-options]  
framing g704-no-crc4;
```

Configuring the E1 Idle Cycle Flag

By default, an E1 interface transmits the value 0x7E in the idle cycles. To have the interface transmit the value 0xFF (all ones) instead, include the **idle-cycle-flag** statement at the **[edit interfaces *interface-name* e1-options]** hierarchy level, specifying the **ones** option:

```
[edit interfaces interface-name e1-options]  
idle-cycle-flag ones;
```

To explicitly configure the default value of 0x7E, include the **idle-cycle-flag** statement with the **flags** option:

```
[edit interfaces interface-name e1-options]  
idle-cycle-flag flags;
```

Configuring E1 Data Inversion

By default, data inversion is disabled. To enable data inversion at the HDLC level, include the **invert-data** statement at the **[edit interfaces *interface-name* e1-options]** hierarchy level:

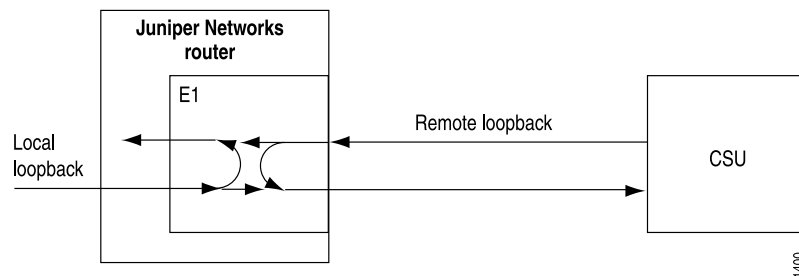
```
[edit interfaces interface-name e1-options]  
invert-data;
```

When you enable data inversion, all data bits in the data stream are transmitted inverted; that is, zeroes are transmitted as ones and ones as zeroes. Data inversion is normally used only in AMI mode to guarantee ones density in the transmitted stream.

Configuring E1 Loopback Capability

You can configure loopback capability between the local E1 interface and the remote channel service unit (CSU), as shown in Figure 29 on page 165. You can configure the loopback to be local or remote. With local loopback, the E1 interface can transmit packets to the CSU, but receives its own transmission back again and ignores data from the CSU. With remote loopback, packets sent from the CSU are received by the E1 interface, forwarded if there is a valid route, and immediately retransmitted to the CSU.

Figure 29: Remote and Local E1 Loopback



To configure loopback capability on an E1 interface, include the **loopback** statement at the **[edit interfaces *interface-name* e1-options]** hierarchy level:

```
[edit interfaces interface-name e1-options]
loopback (local | remote);
```

Packets can be looped on either the local router or the remote CSU.

To exchange BERT patterns between a local router and a remote router, include the **loopback remote** statement in the interface configuration at the remote end of the link. From the local router, you issue the **test interface** command.

For more information about configuring BERT, see Interface Diagnostics. For more information about using operational mode commands to test interfaces, see the [Junos OS System Basics and Services Command Reference](#).

To turn off the loopback capability, remove the **loopback** statement from the configuration:

```
[edit]
user@host# delete interfaces e1-fpc/pic/port e1-options loopback
```

You can determine whether there is an internal problem or an external problem by checking the error counters in the output of the **show interface *interface-name* extensive** command:

```
user@host> show interfaces interface-name extensive
```

Example: Configuring E1 Loopback Capability

To determine whether a problem is internal or external, loop packets on both the local and the remote router. To do this, include the **no-keepalives** and **encapsulation cisco-hdlc** statements at the **[edit interfaces *interface-name*]** hierarchy level and the **loopback local** statement at the **[edit interfaces *interface-name* e1-options]** hierarchy level.

With this configuration, the link stays up, so you can loop ping packets to a remote router. The **loopback local** statement causes the interface to loop within the PIC just before the data reaches the transceiver.

```
[edit interfaces]
e1-1/0/0 {
  no-keepalives;
  encapsulation cisco-hdlc;
  e1-options {
    loopback local;
```

```
    }  
    unit 0 {  
        family inet {  
            address 10.100.100.1/24;  
        }  
    }  
}
```

Configuring E1 Start and End Flags

By default, start and end flags are shared.

To configure an E1 interface to wait two idle cycles between the start and end flags, include the **start-end-flag** statement with the **filler** option at the **[edit interfaces *interface-name* e1-options]** hierarchy level:

```
[edit interfaces interface-name e1-options]  
start-end-flag filler;
```

To revert to the default behavior, sharing the transmission of start and end flags, include the **start-end-flag** statement with the **shared** option at the **[edit interfaces *interface-name* e1-options]** hierarchy level:

```
[edit interfaces interface-name e1-options]  
start-end-flag shared;
```

Configuring Fractional E1 Time Slots

By default, all the time slots on an E1 interface are used. To configure the number of time slots allocated to a fractional E1 interface, include the **timeslots** statement at the **[edit interfaces *interface-name* e1-options]** hierarchy level:

```
[edit interfaces interface-name e1-options]  
timeslots time-slot-range;
```

There are 32 time slots on an E1 interface. Time slot 0 is always reserved for framing and cannot be used to configure a fractional E1 interface.

Time slot numbering constraints vary for different E1 PICs, as follows:

- For 4-port E1 PICs, the configurable time slot range is 1 through 31 (time slot 0 is reserved for framing).
- For 10-port Channelized E1 and 10-port Channelized E1 Intelligent Queuing (IQ) PICs, the configurable time slot range is 2 through 32 (time slots 0 and 1 are reserved for framing).
- For Enhanced Intelligent Queuing (IQE) PICs, the configurable time slot range is 2 through 32.

- NxDS0 time slots configured on either a channelized STM1 IQ interface or a channelized E1IQ interface are numbered from 1 to 31 (0 is reserved), while fractional E1 time slots are numbered from 2 to 32 (0 and 1 are reserved).
- For fractional E1 interfaces only, if you connect a 4-port E1 PIC to a device that uses time slot numbering from 2 through 32, you must subtract 1 from the configured number of time slots. To do this, include the **timeslots** statement at the **[edit interfaces interface-name e1-options]** hierarchy level, and offset 1 from the specified slot number.



NOTE: When configuring fractional E1 time slots, you also must include the framing **g704** statement at the **[edit interfaces e1-fpc/port e1-options]** hierarchy level.

To configure ranges, use hyphens. To configure discontinuous time slots, use commas. Do not include spaces.

Example: Configuring Fractional E1 Time Slots

In this example, time slots are offset by 1 to compensate for the fractional E1 interface being connected to a device that uses time slot numbering from 0 through 31.

Use Time Slots 4 Through 6, 11, and 25	[edit interfaces interface-name e1-options] # Fractional E1 interface timeslots 4-6,11,25;
Use Time Slots 1 Through 10	[edit interfaces interface-name e1-options] timeslots 1-10;
Use Time Slots 1 Through 5, 10, and 24	[edit interfaces interface-name e1-options] timeslots 1-5,10,24;

Configuring E3 Interfaces

- E3 Interfaces Overview on page 169
- Configuring E3 Physical Interface Properties on page 170
- Configuring E3 BERT Properties on page 170
- Configuring the E3 CSU Compatibility Mode on page 171
- Configuring the E3 Frame Checksum on page 172
- Configuring the E3 Idle Cycle Flag on page 173
- Configuring E3 Data Inversion on page 173
- Configuring E3 Loopback Capability on page 173
- Configuring E3 HDLC Payload Scrambling on page 175
- Configuring the E3 Start and End Flags on page 175
- Configuring E3 IQ and IQE Unframed Mode on page 175

E3 Interfaces Overview

E3 is a high-speed WAN digital communication technique designed to operate over copper facilities at a rate of 34.368 Mbps. Widely used outside North America, it is the time-division multiplexing scheme used to carry 16 E1 circuits. The following standards apply to E3 interfaces:

- ITU-T Recommendation G.703, *Physical/electrical characteristics of hierarchical digital interfaces*, describes data rates and multiplexing schemes for the E Series.
- ITU-T Recommendation G.751, *General Aspects of Digital Transmission Systems: Terminal Equipment*, describes framing methods.
- ITU-T Recommendation G.775, *Loss of Signal (LOS) and Alarm Indication Signal (AIS) Defect Detection and Clearance Criteria*, describes alarm reporting methods.

The Junos OS supports the E3 Physical Interface Card (PIC) and the E3 Intelligent Queuing (IQ and IQE) PICs. The E3 IQ and E3 IQE PICs supports transmission scheduling on logical interfaces. For more information, see the [Junos OS Class of Service Configuration Guide](#).



NOTE: In unframed mode, the E3 IQ and E3 IQE PICs do not detect yellow or loss-of-frame alarms.

Configuring E3 Physical Interface Properties

To configure E3-specific physical interface properties, include the **e3-options** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]  
e3-options {  
  bert-algorithm algorithm;  
  bert-error-rate rate;  
  bert-period seconds;  
  compatibility-mode (digital-link | kentrox | larscom) <subrate value>;  
  fcs (16 | 32);  
  idle-cycle-flag value;  
  invert-data;  
  loopback (local | remote);  
  (payload-scrambler | no-payload-scrambler);  
  start-end-flag value;  
  (unframed | no-unframed);  
}
```

Configuring E3 BERT Properties

This section discusses BERT properties for the E3 interface specifically. For general information about the Junos implementation of the BERT procedure, see Interface Diagnostics.

You can configure an E3 interface to execute a bit error rate test (BERT) when the interface receives a request to run this test. You specify the duration of the test, the pattern to send in the bit stream, and the error rate to include in the bit stream by including the **bert-period**, **bert-algorithm**, and **bert-error-rate** statements at the **[edit interfaces *interface-name* e3-options]** hierarchy level:

```
[edit interfaces interface-name e3-options]  
bert-algorithm algorithm;  
bert-error-rate rate;  
bert-period seconds;
```

By default, the BERT period is 10 seconds. You can configure the BERT period to last from 1 through 239 seconds on some PICs and from 1 through 240 seconds on other PICs.

rate is the bit error rate. This can be an integer from 0 through 7, which corresponds to a bit error rate from 10^{-0} (0, which corresponds to no errors) to 10^{-7} (1 error per 10 million bits).

algorithm is the pattern to send in the bit stream. On E3 interfaces, you can also select the pattern to send in the bit stream by including the **bert-algorithm** statement at the **[edit interfaces *interface-name* *interface-options*]** hierarchy level:

```
[edit interfaces interface-name interface-options]  
bert-algorithm algorithm;
```

For a list of supported algorithms, enter a ? after the **bert-algorithm** statement; for example:

```
[edit interfaces e3-0/0/0 e3-options]
user@host# set bert-algorithm ?
Possible completions:
pseudo-2e11-o152 Pattern is 2^11 - 1 (per O.152 standard)
pseudo-2e15-o151 Pattern is 2^15 - 1 (per O.152 standard)
pseudo-2e20-o151 Pattern is 2^20 - 1 (per O.151 standard)
pseudo-2e20-o153 Pattern is 2^20 - 1 (per O.153 standard)
```

For specific hierarchy information, see individual interface types. For information about running the BERT procedure, see the *Junos OS System Basics and Services Command Reference*.

Configuring the E3 CSU Compatibility Mode

Subrating an E3 interface reduces the maximum allowable peak rate by limiting the High-level Data Link Control (HDLC)-encapsulated payload. Subrate modes configure the PIC to connect with channel service units (CSUs) that use proprietary methods of multiplexing.

On M Series and T Series routers, you can configure E3 interfaces to be compatible with a Digital Link, Kentrox, or Larscom CSU. On J Series Services Routers, you can configure E3 interfaces to be compatible with a Digital Link or Kentrox CSU.



NOTE: To subrate an E3 interface to be compatible with a Kentrox CSU, you must have an IQ-based PIC. Non-IQ PICs allow a commit of the configuration, but the interfaces remain at the full E3 rate for the Kentrox compatibility mode.

For E3 interfaces on IQE PICs, subrate is not supported and the `E3-options compatibility-mode` and `payload-scrambler` are invalid. Although Junos OS CLI allows a commit of this configuration, the interfaces remain at the full E3 rate and implicitly default to only Kentrox compatibility mode.

To configure an E3 interface so that it is compatible with the CSU at the remote end of the line, include the `compatibility-mode` statement at the `[edit interfaces interface-name e3-options]` hierarchy level:

```
[edit interfaces interface-name e3-options]
compatibility-mode (digital-link | kentrox | larscom) <subrate value>;
```

The subrate of an E3 interface must exactly match that of the remote CSU. To specify the subrate, include the `subrate` statement in the configuration:

- For Kentrox CSUs, specify the subrate as a number from 1 through 48 that exactly matches the value configured on the CSU. Each increment of the subrate value corresponds to a rate increment of about 0.5 Mbps.
- For Digital Link CSUs, you can specify the subrate value to match the data rate configured on the CSU in the format `xkb` or `x.xMb`. You can configure the subrate values shown in Table 16 on page 172.
- Larscom CSUs do not support the E3 subrate.

Table 16: Subrate Values for E3 Digital Link Compatibility Mode

358 Kbps	7.2 Mbps	14.0 Mbps	20.8 Mbps	27.6 Mbps
716 Kbps	7.5 Mbps	14.3 Mbps	21.1 Mbps	27.9 Mbps
1.1 Mbps	7.9 Mbps	14.7 Mbps	21.5 Mbps	28.3 Mbps
1.4 Mbps	8.2 Mbps	15.0 Mbps	21.8 Mbps	28.6 Mbps
1.8 Mbps	8.6 Mbps	15.4 Mbps	22.2 Mbps	29.0 Mbps
2.1 Mbps	9.0 Mbps	15.8 Mbps	22.6 Mbps	29.4 Mbps
2.5 Mbps	9.3 Mbps	16.1 Mbps	22.9 Mbps	29.7 Mbps
2.9 Mbps	9.7 Mbps	16.5 Mbps	23.3 Mbps	30.1 Mbps
3.2 Mbps	10.0 Mbps	16.8 Mbps	23.6 Mbps	30.4 Mbps
3.6 Mbps	10.4 Mbps	17.2 Mbps	24.0 Mbps	30.8 Mbps
3.9 Mbps	10.7 Mbps	17.5 Mbps	24.3 Mbps	31.1 Mbps
4.3 Mbps	11.1 Mbps	17.9 Mbps	24.7 Mbps	31.5 Mbps
4.7 Mbps	11.5 Mbps	18.3 Mbps	25.1 Mbps	31.9 Mbps
5.0 Mbps	11.8 Mbps	18.6 Mbps	25.4 Mbps	32.2 Mbps
5.4 Mbps	12.2 Mbps	19.0 Mbps	25.8 Mbps	32.6 Mbps
5.7 Mbps	12.5 Mbps	19.3 Mbps	26.1 Mbps	32.9 Mbps
6.1 Mbps	12.9 Mbps	19.7 Mbps	26.5 Mbps	33.3 Mbps
6.4 Mbps	13.2 Mbps	20.0 Mbps	26.9 Mbps	33.7 Mbps
6.8 Mbps	13.6 Mbps	20.4 Mbps	27.2 Mbps	

For information about subrating a T3 interface, see “Configuring the T3 CSU Compatibility Mode” on page 190.

Configuring the E3 Frame Checksum

You can configure a 32-bit checksum, which provides more reliable packet verification. However, some older equipment might not support 32-bit checksums.

On a channelized OC12 interface, the **fcs** statement is not supported. To configure FCS on each E3 channel, you must include the **e3-options fcs** statement in the configuration for each channel.

To configure a 32-bit checksum, include the **fcs** statement at the **[edit interfaces *interface-name* e3-options]** hierarchy level:

```
[edit interfaces interface-name e3-options]  
fcs 32;
```

To return to the default 16-bit frame checksum, delete the **fcs 32** statement from the configuration:

```
[edit]  
user@host# delete interfaces e3-fpc/pic/port e3-options fcs 32
```

To explicitly configure a 16-bit checksum, include the **fcs** statement at the **[edit interfaces *interface-name* e3-options]** hierarchy level:

```
[edit interfaces interface-name e3-options]  
fcs 16;
```

Configuring the E3 Idle Cycle Flag

By default, an E3 interface transmits the value 0x7E in the idle cycles. To have the interface transmit the value 0xFF (all ones) instead, include the **idle-cycle-flag** statement at the **[edit interfaces *interface-name* e3-options]** hierarchy level, specifying the **ones** option:

```
[edit interfaces interface-name e3-options]  
idle-cycle-flag ones;
```

To explicitly configure the default value of 0x7E, include the **idle-cycle-flag** statement with the **flags** option:

```
[edit interfaces interface-name e3-options]  
idle-cycle-flag flags;
```

Configuring E3 Data Inversion

By default, data inversion is disabled. To enable data inversion at the HDLC level, include the **invert-data** statement at the **[edit interfaces *interface-name* e3-options]** hierarchy level:

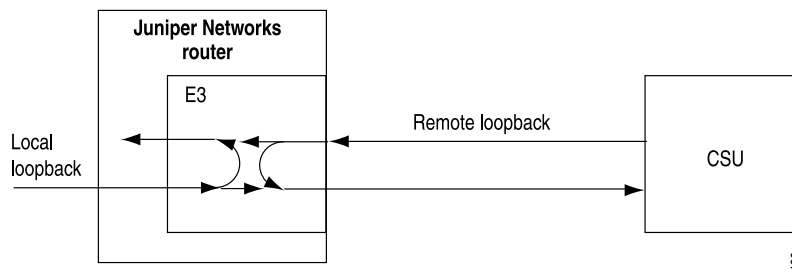
```
[edit interfaces interface-name e3-options]  
invert-data;
```

When you enable data inversion, unused data bits in the data stream are transmitted inverted; that is, zeroes are transmitted as ones and ones as zeroes. Enable inversion to be compatible with another vendor's E3 interface.

Configuring E3 Loopback Capability

You can configure loopback capability between the local E3 interface and the remote CSU. You can configure the loopback to be local or remote. With local loopback, the E3 interface can transmit packets to the CSU, but receives its own transmission back again and ignores data from the CSU. With remote loopback, packets sent from the CSU are received by the E3 interface, forwarded if there is a valid route, and immediately retransmitted to the CSU (see Figure 30 on page 174).

Figure 30: Remote and Local E3 Loopback



To configure loopback capability on an E3 interface, include the **loopback** statement at the **[edit interfaces *interface-name* e3-options]** hierarchy level:

```
[edit interfaces interface-name e3-options]
loopback (local | remote);
```

Packets can be looped on either the local router or the remote CSU.

To exchange BERT patterns between a local router and a remote router, include the **loopback remote** statement in the interface configuration at the remote end of the link. From the local router, you issue the **test interface** command.

For more information about configuring BERT, see [Interface Diagnostics](#). For more information about using operational mode commands to test interfaces, see the [Junos OS System Basics and Services Command Reference](#).

To turn off the loopback capability, remove the **loopback** statement from the configuration:

```
[edit]
user@host# delete interfaces e3-fpc/pic/port e3-options loopback
```

You can determine whether there is an internal problem or an external problem by checking the error counters in the output of the **show interface *interface-name* extensive** command:

```
user@host> show interfaces interface-name extensive
```

Example: Configuring E3 Loopback Capability

To determine whether a problem is internal or external, loop packets on both the local and the remote router. To do this, include the **no-keepalives** and **encapsulation cisco-hdlc** statements at the **[edit interfaces *interface-name*]** hierarchy level and the **loopback local** statement at the **[edit interfaces *interface-name* e3-options]** hierarchy level. With this configuration, the link stays up, so you can loop ping packets to a remote router. The **loopback local** statement causes the interface to loop within the PIC just before the data reaches the transceiver.

```
[edit interfaces]
e3-1/0/0 {
  no-keepalives;
  encapsulation cisco-hdlc;
  e3-options {
    loopback local;
  }
}
```

```

unit 0 {
  family inet {
    address 10.100.100.1/24;
  }
}

```

Configuring E3 HDLC Payload Scrambling

E3 HDLC payload scrambling, which is disabled by default, provides better link stability. Both sides of a connection must either use or not use scrambling.

To configure scrambling on the interface, you can include the **payload-scrambler** statement at the **[edit interfaces *interface-name* e3-options]** hierarchy level:

```

[edit interfaces interface-name e3-options]
payload-scrambler;

```

To explicitly disable HDLC payload scrambling, include the **no-payload-scrambler** statement at the **[edit interfaces *interface-name* e3-options]** hierarchy level:

```

[edit interfaces interface-name e3-options]
no-payload-scrambler;

```

To disable payload scrambling again (return to the default), delete the **payload-scrambler** statement from the configuration:

```

[edit]
user@host# delete interfaces e3-fpc/pic/port e3-options payload-scrambler

```

Configuring the E3 Start and End Flags

By default, an E3 interface shares the transmission of the start and end flags

To configure an E3 interface to wait two idle cycles between the start and end flags, include the **start-end-flag** statement with the **filler** option at the **[edit interfaces *interface-name* e3-options]** hierarchy level:

```

[edit interfaces interface-name e3-options]
start-end-flag filler;

```

To revert to the default behavior, sharing the transmission of start and end flags, include the **start-end-flag** statement with the **shared** option at the **[edit interfaces *interface-name* e3-options]** hierarchy level:

```

[edit interfaces interface-name e3-options]
start-end-flag shared;

```

Configuring E3 IQ and IQE Unframed Mode

For E3 IQ and IQE interfaces only, you can enable or disable unframed mode. In unframed mode, the E3 IQ and IQE interfaces do not detect yellow (**ylw**) or loss-of-frame (**lof**) alarms.

By default, unframed mode is disabled. To enable unframed mode, include the **unframed** statement at the **[edit interfaces *interface-name* e3-options]** hierarchy level:

```
[edit interfaces interface-name e3-options]  
unframed;
```

To explicitly configure the default of framed mode, include the **no-unframed** statement:

```
[edit interfaces interface-name e3-options]  
no-unframed;
```

CHAPTER 13

Configuring T1 Interfaces

- T1 Interfaces Overview on page 177
- Configuring T1 Physical Interface Properties on page 178
- Configuring T1 BERT Properties on page 178
- Configuring the T1 Buildout on page 179
- Configuring T1 Byte Encoding on page 179
- Configuring T1 CRC Error Major Alarm Thresholds on page 180
- Configuring T1 CRC Error Minor Alarm Thresholds on page 180
- Configuring T1 Data Inversion on page 181
- Configuring the T1 Frame Checksum on page 181
- Configuring the T1 Remote Loopback Response on page 181
- Configuring T1 Framing on page 182
- Configuring T1 Line Encoding on page 182
- Configuring T1 Loopback Capability on page 182
- Configuring the T1 Idle Cycle Flag on page 184
- Configuring T1 Start and End Flags on page 184
- Configuring Fractional T1 Time Slots on page 185

T1 Interfaces Overview

T1 is the basic physical layer protocol used by the Digital Signal level 1 (DS1) multiplexing method in North America. A T1 interface operates at a bit rate of 1.544 Mbps and can support 24 DS0 channels. Supported DS1 standards include:

- ANSI T1.107, T1.102
- GR 499-core, GR 253-core
- AT&T Pub 54014
- ITU G.751, G.703

Configuring T1 Physical Interface Properties

To configure T1-specific physical interface properties, include the **t1-options** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]  
t1-options {  
  bert-algorithm algorithm;  
  bert-error-rate rate;  
  bert-period seconds;  
  buildout value;  
  byte-encoding (nx56 | nx64);  
  crc-major-alarm-threshold (1e-3 | 5e-4 | 1e-4 | 5e-5 | 1e-5);  
  crc-minor-alarm-threshold (1e-3 | 5e-4 | 1e-4 | 5e-5 | 1e-5 | 5e-6 | 1e-6);  
  fcs (16 | 32);  
  framing (esf | sf);  
  idle-cycle-flag (flags | ones);  
  invert-data;  
  line-encoding (ami | b8zs);  
  loopback (local | payload | remote);  
  remote-loopback-respond;  
  start-end-flag (filler | shared);  
  timeslots time-slot-range;  
}
```

Configuring T1 BERT Properties

This section discusses BERT properties for the T1 interface specifically. For general information about the Junos implementation of the BERT procedure, see Interface Diagnostics.

You can configure a T1 interface or partitioned CT1 or T1 channel to execute a bit error rate test (BERT) when the interface receives a request to run this test. You specify the duration of the test and the error rate to include in the bit stream by including the **bert-period** and **bert-error-rate** statements at the **[edit interfaces *interface-name* t1-options]** hierarchy level:

```
[edit interfaces interface-name t1-options]  
bert-algorithm algorithm;  
bert-error-rate rate;  
bert-period seconds;
```

seconds is the duration of the BERT procedure. The test can last from 1 through 239 seconds; the default is 10 seconds. Standard CT1, standard T1, T1 IQ, and T1 IQE interfaces, and PICs partitioned to CT1 and T1 channels, support an extended BERT period range, up to 86,400 seconds (24 hours), and have a default BERT period value of 240 seconds.



NOTE: When configuring T1 and CT1 interfaces on 10-port Channelized E1/T1 IQE PICs, the **bert-period** statement must be included at the **[edit interfaces *ct1-fpc/pic/port*]** hierarchy level.

rate is the bit error rate. This can be an integer from 0 through 7, which corresponds to a bit error rate from 10^{-0} (1 error per bit) to 10^{-7} (1 error per 10 million bits).

algorithm is the pattern to send in the bit stream. On T1 interfaces, you can also select the pattern to send in the bit stream by including the **bert-algorithm** statement at the **[edit interfaces interface-name interface-options]** hierarchy level:

```
[edit interfaces interface-name interface-options]
bert-algorithm algorithm;
```

For a list of supported algorithms, enter a **?** after the **bert-algorithm** statement; for example:

```
[edit interfaces t1-0/0/0 t1-options]
user@host# set bert-algorithm ?
Possible completions:
pseudo-2e11-o152 Pattern is 2^11 - 1 (per O.152 standard)
pseudo-2e15-o151 Pattern is 2^15 - 1 (per O.152 standard)
pseudo-2e20-o151 Pattern is 2^20 - 1 (per O.151 standard)
pseudo-2e20-o153 Pattern is 2^20 - 1 (per O.153 standard)
```

For specific hierarchy information, see individual interface types. For information about running the BERT procedure, see the [Junos OS System Basics and Services Command Reference](#).

Configuring the T1 Buildout

A T1 interface has five possible setting ranges for the T1 line buildout: **0-132**, **133-265**, **266-398**, **399-531**, or **532-655** feet. By default, the T1 interface uses the shortest setting (0-132).

To have the interface drive a line at one of the longer distance ranges, include the **buildout** statement with the appropriate value at the **[edit interfaces interface-name t1-options]** hierarchy level:

```
[edit interfaces interface-name t1-options]
buildout value;
```

Configuring T1 Byte Encoding

By default, T1 interfaces use a byte encoding of 8 bits per byte (nx64). You can configure an alternative byte encoding of 7 bits per byte (nx56).

To have the interface use 7 bits per byte encoding, include the **byte-encoding** statement at the **[edit interfaces interface-name t1-options]** hierarchy level, specifying the **nx56** option:

```
[edit interfaces interface-name t1-options]
byte-encoding nx56;
```

To explicitly configure nx64 byte encoding, include the **byte-encoding** statement at the **[edit interfaces interface-name t1-options]** hierarchy level, specifying the **nx64** option:

```
[edit interfaces interface-name t1-options]
```

byte-encoding nx64;

Configuring T1 CRC Error Major Alarm Thresholds

Junos OS collects CRC errors from PICs every second. On Channelized OC3 IQ and IQE PICs, Channelized OC12 IQ and IQE PICs, and Channelized T3 IQ PICs, you can configure major error thresholds for T1 CRC errors.

When the threshold is exceeded for 1 second, a defect condition is declared. If the defect condition continues for the monitoring period, an alarm condition is declared. You can display the CRC error threshold configuration, CRC errors count, and the alarm condition using the **show interfaces extensive** command.

To configure a CRC major error threshold, include the **crc-major-alarm-threshold** statement at the **[edit interfaces *interface-name* t1-options]** hierarchy level, specifying the errors per bits as **1e-3**, **5e-4**, **1e-4**, **5e-5** or **1e-5**:

```
[edit interfaces interface-name t1-options]
  crc-major-alarm-threshold (1e-3 | 5e-4 | 1e-4 | 5e-5 | 1e-5);
```

To configure a T1 CRC error major alarm for five errors in 10^{-4} bits, include the **crc-major-alarm-threshold** statement at the **[edit interfaces *interface-name* t1-options]** hierarchy level, specifying the **5e-4** option:

```
[edit interfaces interface-name t1-options]
  crc-major-alarm-threshold 5e-4;
```

All settings except **1e-5** use a 10-second monitoring period. The **1e-5** value uses a 50-second monitoring period.

Configuring T1 CRC Error Minor Alarm Thresholds

Junos OS collects CRC errors from PICs every second. On Channelized OC3 IQ and IQE PICs, Channelized OC12 IQ and IQE PICs, and Channelized T3 IQ PICs, you can configure minor error thresholds for T1 CRC errors.

When the threshold is exceeded for 1 second, a defect condition is declared. If the defect condition continues for the monitoring period, an alarm condition is declared. You can display the CRC error threshold configuration, CRC errors count, and the alarm condition using the **show interfaces extensive** command.

To configure a CRC minor error threshold, include the **crc-minor-alarm-threshold** statement at the **[edit interfaces *interface-name* t1-options]** hierarchy level, specifying the errors per bits as **1e-3**, **5e-4**, **1e-4**, **5e-5**, **1e-5**, **5e-6**, or **1e-6**:

```
[edit interfaces interface-name t1-options]
  crc-minor-alarm-threshold (1e-3 | 5e-4 | 1e-4 | 5e-5 | 1e-5 | 5e-6 | 1e-6);
```

To configure a T1 CRC error minor alarm for five errors in 10^{-4} bits, include the **crc-minor-alarm-threshold** statement at the **[edit interfaces *interface-name* t1-options]** hierarchy level, specifying the **5e-4** option:

```
[edit interfaces interface-name t1-options]
  crc-minor-alarm-threshold 5e-4;
```


The 10-second monitoring period is used for values **1e-3**, **5e-4**, **1e-4**, and **5e-5**. The **1e-5** value uses a 50-second monitoring period. The **5e-6** value uses a 100-second monitoring period. The **1e-6** value uses a 500-second monitoring period.

Configuring T1 Data Inversion

By default, data inversion is disabled. To enable data inversion at the HDLC level, include the **invert-data** statement at the **[edit interfaces *interface-name* t1-options]** hierarchy level:

```
[edit interfaces interface-name t1-options]
invert-data;
```

When you enable data inversion, all data bits in the data stream are transmitted inverted; that is, zeroes are transmitted as ones and ones as zeroes. Data inversion is normally used only in AMI mode to guarantee ones density in the transmitted stream.

Configuring the T1 Frame Checksum

By default, T1 interfaces use a 16-bit frame checksum. You can configure a 32-bit checksum, which provides more reliable packet verification. However, some older equipment might not support 32-bit checksums.

To configure a 32-bit checksum, include the **fcs 32** statement at the **[edit interfaces *interface-name* t1-options]** hierarchy level:

```
[edit interfaces interface-name t1-options]
fcs 32;
```

To return to the default 16-bit frame checksum, delete the **fcs 32** statement from the configuration:

```
[edit]
user@host# delete interfaces t1-fpc/pic/port t1-options fcs 32
```

To explicitly configure a 16-bit checksum, include the **fcs 16** statement at the **[edit interfaces *interface-name* t1-options]** hierarchy level:

```
[edit interfaces interface-name t1-options]
fcs 16;
```

Configuring the T1 Remote Loopback Response

The T1 facilities data-link loop request signal is used to communicate various network information in the form of in-service monitoring and diagnostics. Extended superframe, through the facilities data link (FDL), supports nonintrusive signaling and control, thereby offering clear-channel communication. Remote loopback requests can be over the FDL or inband. To configure the router to respond to remote loopback requests, include the **remote-loopback-respond** statement at the **[edit interfaces *interface-name* t1-options]** hierarchy level:

```
[edit interfaces interface-name t1-options]
remote-loopback-respond;
```

By default, the router does not respond to remote loopback requests.

Configuring T1 Framing

By default, T1 interfaces use extended superframe framing format. You can configure SF (superframe) as an alternative.

To have the interface use the SF framing format, include the **framing** statement at the **[edit interfaces *interface-name* t1-options]** hierarchy level, specifying the **sf** option:

```
[edit interfaces interface-name t1-options]  
framing sf;
```

To explicitly configure ESF framing, include the **framing** statement at the **[edit interfaces *interface-name* t1-options]** hierarchy level, specifying the **esf** option:

```
[edit interfaces interface-name t1-options]  
framing esf;
```

Configuring T1 Line Encoding

By default, T1 interfaces use B8ZS line encoding. You can configure AMI line encoding if necessary.

To have the interface use AMI line encoding, include the **line-encoding** statement at the **[edit interfaces *interface-name* t1-options]** hierarchy level, specifying the **ami** option:

```
[edit interfaces interface-name t1-options]  
line-encoding ami;
```

To explicitly configure B8ZS line encoding, include the **line-encoding** statement at the **[edit interfaces *interface-name* t1-options]** hierarchy level, specifying the **b8zs** option:

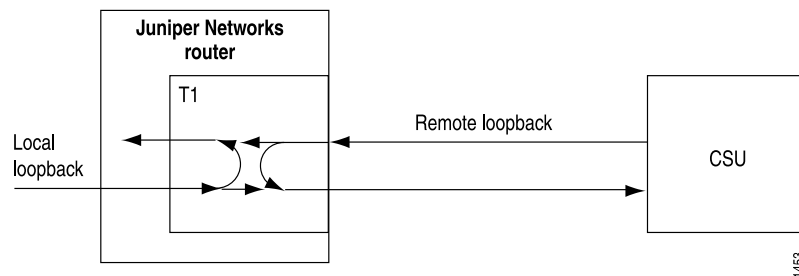
```
[edit interfaces interface-name t1-options]  
line-encoding b8zs;
```

For M Series and T Series routers, you must set the line encoding parameter for paired ports to the same value. Ports 0 and 1 must share the same value, and likewise ports 2 and 3 must share the same value, but ports 0 and 1 can have a different value from that of ports 2 and 3.

Configuring T1 Loopback Capability

You can configure loopback capability between the local T1 interface and the remote channel service unit (CSU), as shown in Figure 31 on page 183. You can configure the loopback to be local or remote. With local loopback, the T1 interface can transmit packets to the CSU, but receives its own transmission back again and ignores data from the CSU. With remote loopback, packets sent from the CSU are received by the T1 interface, forwarded if there is a valid route, and immediately retransmitted to the CSU.

Figure 31: Remote and Local T1 Loopback



To configure loopback capability on a T1 interface, include the **loopback** statement at the **[edit interfaces *interface-name* t1-options]** hierarchy level:

```
[edit interfaces interface-name t1-options]
loopback (local | payload | remote);
```

Packets can be looped on either the local router or the remote CSU. Local and remote loopback loop back both data and clocking information.

To exchange BERT patterns between a local router and a remote router, include the **loopback remote** statement in the interface configuration at the remote end of the link. From the local router, issue the **test interface** command.

For more information about configuring BERT, see *Interface Diagnostics*. For more information about using operational mode commands to test interfaces, see the [Junos OS System Basics and Services Command Reference](#).

For channelized T3, T1, and NxDS0 intelligent queuing (IQ) interfaces only, you can include the **loopback payload** statement in the configuration to loop back data only (without clocking information) on the remote router's PIC. In payload loopback, overhead is recalculated. For T3 IQ interfaces, you can include the **loopback payload** statement at the **[edit interfaces *ct3-fpc/pic/port*]** and **[edit interfaces *t3-fpc/pic/port:channel*]** hierarchy levels. For T1 interfaces, you can include the **loopback payload** statement in the configuration at the **[edit interfaces *t1-fpc/pic/port:channel*]** hierarchy level; it is ignored if included at the **[edit interfaces *ct1-fpc/pic/port*]** hierarchy level. For NxDS0 interfaces, payload and remote loopback are the same. If you configure one, the other is ignored. NxDS0 IQ interfaces do not support local loopback.

To determine whether a problem is internal or external, you can loop packets on both the local and the remote router. To do this, include the **no-keepalives** and **encapsulation cisco-hdlc** statements at the **[edit interfaces *interface-name*]** hierarchy level and the **loopback local** statement at the **[edit interfaces *interface-name* t1-options]** hierarchy level, as shown in the following example:

```
[edit interfaces]
t1-1/0/0 {
  no-keepalives;
  encapsulation cisco-hdlc;
  t1-options {
    loopback local;
  }
  unit 0 {
```

```
family inet {  
    address 10.100.100.1/24;  
}  
}
```

With this configuration, the link stays up, so you can loop ping packets to a remote router. The **loopback local** statement causes the interface to loop within the PIC just before the data reaches the transceiver.

To turn off the loopback capability, remove the **loopback** statement from the configuration:

```
[edit]  
user@host# delete interfaces t1-fpc/pic/port t1-options loopback
```

You can determine whether there is an internal problem or an external problem by checking the error counters in the output of the **show interface *interface-name* extensive** command, for example:

```
user@host> show interfaces t1-fpc/pic/port extensive
```

Configuring the T1 Idle Cycle Flag

By default, a T1 interface transmits the value 0x7E in the idle cycles. To have the interface transmit the value 0xFF (all ones) instead, include the **idle-cycle-flag** statement at the **[edit interfaces *interface-name* t1-options]** hierarchy level, specifying the **ones** option:

```
[edit interfaces interface-name t1-options]  
idle-cycle-flag ones;
```

To explicitly configure the default value of 0x7E, include the **idle-cycle-flag** statement with the **flags** option:

```
[edit interfaces interface-name t1-options]  
idle-cycle-flag flags;
```

Configuring T1 Start and End Flags

By default, a T1 interface shares the transmission of the start and end flags.

To configure a T1 interface to wait two idle cycles between the start and end flags, include the **start-end-flag** statement with the **filler** option at the **[edit interfaces *interface-name* t1-options]** hierarchy level:

```
[edit interfaces interface-name t1-options]  
start-end-flag filler;
```

To revert to the default behavior, sharing the transmission of start and end flags, include the **start-end-flag** statement with the **shared** option at the **[edit interfaces *interface-name* t1-options]** hierarchy level:

```
[edit interfaces interface-name t1-options]  
start-end-flag shared;
```

Configuring Fractional T1 Time Slots

By default, all the time slots on a T1 interface are used. To configure the number of time slots allocated to a fractional T1 interface, include the **timeslots** statement at the **[edit interfaces *interface-name* t1-options]** hierarchy level:

```
[edit interfaces interface-name t1-options]
timeslots time-slot-range;
```

For T1 interfaces, the time-slot range is from 1 through 24. There are 24 time slots on a T1 interface. You can designate any combination of time slots. To configure ranges, use hyphens. To configure discontinuous time slots, use commas. Do not include spaces.

Example: Configuring Fractional T1 Time Slots

Use Time Slots 1 Through 10	[edit interfaces <i>interface-name</i> t1-options] timeslots 1-10;
Use Time Slots 1 Through 5, 10, and 24	[edit interfaces <i>interface-name</i> t1-options] timeslots 1-5,10,24;
Use the First Four Odd-Numbered Time Slots	[edit interfaces <i>interface-name</i> t1-options] timeslots 1,3,5,7;

Configuring T3 Interfaces

- T3 Interfaces Overview on page 187
- Configuring T3 Physical Interface Properties on page 188
- Configuring T3 BERT Properties on page 188
- Disabling T3 C-Bit Parity Mode on page 189
- Configuring the T3 CSU Compatibility Mode on page 190
- Configuring the T3 Frame Checksum on page 192
- Configuring the T3 FEAC Response on page 193
- Configuring the T3 Idle Cycle Flag on page 193
- Configuring the T3 Line Buildout on page 194
- Configuring T3 Loopback Capability on page 194
- Configuring T3 HDLC Payload Scrambling on page 196
- Configuring T3 Start and End Flags on page 196
- Examples: Configuring T3 Interfaces on page 197

T3 Interfaces Overview

T3 is the physical layer protocol used by the Digital Signal level 3 (DS3) multiplexing method in North America. A T3 interface operates at a bit rate of 44.736 Mbps. The Junos OS supports payload scrambling and subrate operation on each physical T3 interface. One encapsulation format—Point-to-Point Protocol (PPP), Frame Relay, or High-level Data Link Control (HDLC)—must be configured for the interface. DS3 standards supported include:

- ANSI T1.107, T1.102
- GR 499-core, GR 253-core
- Bellcore TR-TSY-000009
- AT&T Pub 5404
- ITU G.751, G.703, G823

Configuring T3 Physical Interface Properties

To configure T3-specific physical interface properties, include the **t3-options** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]  
t3-options {  
  bert-algorithm algorithm;  
  bert-error-rate rate;  
  bert-period seconds;  
  (cbit-parity | no-cbit-parity);  
  compatibility-mode (adtran | digital-link | kentrox | larscom | verilink) <subrate value>;  
  fcs (16 | 32);  
  (feac-loop-respond | no-feac-loop-respond);  
  idle-cycle-flag value;  
  (long-buildout | no-long-buildout);  
  (loop-timing | no-loop-timing);  
  loopback (local | payload | remote);  
  (payload-scrambler | no-payload-scrambler);  
  start-end-flag value;  
}
```

Configuring T3 BERT Properties

This section discusses BERT properties for the T3 interface specifically. For general information about the Junos implementation of the BERT procedure, see Interface Diagnostics.

You can configure a T3 interface to execute a bit error rate test (BERT) when the interface receives a request to run this test. You specify the duration of the test, the pattern to send in the bit stream, and the error rate to include in the bit stream by including the **bert-period**, **bert-algorithm**, and **bert-error-rate** statements at the **[edit interfaces *interface-name* t3-options]** hierarchy level:

```
[edit interfaces interface-name t3-options]  
bert-algorithm algorithm;  
bert-error-rate rate;  
bert-period seconds;
```

By default, the BERT period is 10 seconds. You can configure the BERT period to last from 1 through 239 seconds on some PICs and from 1 through 240 seconds on other PICs.

rate is the bit error rate. This can be an integer from 0 through 7, which corresponds to a bit error rate from 10^{-0} (1 error per bit) to 10^{-7} (1 error per 10 million bits).

algorithm is the pattern to send in the bit stream. The default algorithm for the DS3 BERT procedure is **pseudo-2e15-o151** (pattern is $2^{15}-1$, as defined in the CCITT/ITU O.151 standard).

On T3 interfaces, you can also select the pattern to send in the bit stream by including the **bert-algorithm** statement at the **[edit interfaces *interface-name* *interface-options*]** hierarchy level:


```
[edit interfaces interface-name interface-options]
bert-algorithm algorithm;
```

For a list of supported algorithms, enter a ? after the **bert-algorithm** statement; for example:

```
[edit interfaces t3-0/0/0 t3-options]
user@host# set bert-algorithm ?
Possible completions:
all-ones-repeating Repeating one bits
all-zeros-repeating Repeating zero bits
alternating-double-ones-zeros Alternating pairs of ones and zeros
alternating-ones-zeros Alternating ones and zeros
pseudo-2e10 Pattern is 2^10 - 1
...
```

For specific hierarchy information, see individual interface types. For information about running the BERT procedure, see the *Junos OS System Basics and Services Command Reference*.

Disabling T3 C-Bit Parity Mode

C-bit parity mode controls the type of framing that is present on the transmitted T3 signal. When C-bit parity mode is enabled, the C-bit positions are used for the FEBE, FEAC, terminal data link, path parity, and mode indicator bits, as defined in ANSI T1.107a-1989. When C-bit parity mode is disabled, the basic T3 framing mode (M23) is used.

By default, C-bit parity mode is enabled. To disable C-bit parity mode and use M23 framing for your T3 link, include the **no-cbit-parity** statement at the **[edit interfaces *interface-name* t3-options]** hierarchy level:

```
[edit interfaces interface-name t3-options]
no-cbit-parity;
```



NOTE: For ATM, ATM2 IQ2, IQ2-E, and T3 interfaces, M23 framing is used when the **no-cbit-parity** statement is included. For all other interfaces, M13 framing is used when the **no-cbit-parity** statement is included.

To return to the default, enabling C-bit parity mode, delete the **no-cbit-parity** statement from the configuration:

```
[edit]
user@host# delete interfaces t3-fpc/pic/port t3-options no-cbit-parity
```

To explicitly enable C-bit parity mode, include the **cbit-parity** statement at the **[edit interfaces *interface-name* t3-options]** hierarchy level:

```
[edit interfaces interface-name t3-options]
cbit-parity;
```

Configuring the T3 CSU Compatibility Mode

Subrating a T3 interface reduces the maximum allowable peak rate by limiting the HDLC-encapsulated payload. Subrate modes configure the PIC to connect with channel service units (CSUs) that use proprietary methods of multiplexing.

You can configure T3 interfaces to be compatible with a Digital Link, Kentrox, or Larscom CSUs. For T3 intelligent queuing (IQ) channels only, you can also configure Adtran or Verilink CSU compatibility.



NOTE: To subrate an E3 interface to be compatible with a Kentrox CSU, you must have an IQ or IQE based PIC. Non-IQ or IQE PICs allow a commit of the configuration, but the interfaces remain at the full E3 rate for the Kentrox compatibility mode.

4-port and 2-port channelized DS3(T3) IQ PICs do not support Adtran and Verilink compatibility modes. If configured, the default mode is applied on both the interfaces.

To configure a T3 interface so that it is compatible with the CSU at the remote end of the line, include the **compatibility** statement at the **[edit interfaces *interface-name* t3-options]** hierarchy level:

```
[edit interfaces interface-name t3-options]
  compatibility-mode (adtran | digital-link | kentrox | larscom | verilink) <subrate value>;
```

The subrate of a T3 interface must exactly match that of the remote CSU. To specify the subrate, include the **subrate** statement in the configuration:

- For Adtran CSUs, specify the subrate as a number from 1 through 588 that exactly matches the value configured on the CSU. A subrate value of 588 corresponds to 44.2 Mbps, or 100 percent of the HDLC-encapsulated payload. A subrate value of 1 corresponds to $44.2 / 588$, which is 75.17 Kbps, or 0.17 percent of the HDLC-encapsulated payload.
- For Digital Link CSUs, specify the subrate as the data rate you configured on the CSU in the format xKb or x.xMb. For Digital Link CSUs, you can specify the subrate value to match the data rate configured on the CSU in the format **xkb** or **x.xMb**. You can configure the subrate values shown in Table 17 on page 191.
- For Kentrox CSUs, specify the subrate as a number from 1 through 69 that exactly matches the value configured on the CSU. A subrate value of 69 corresponds to 34.995097 Mbps, or 79.17 percent of the HDLC-encapsulated payload (44.2 Mbps). A subrate value of 1 corresponds to 999.958 Kbps, which is 2.26 percent of the HDLC-encapsulated payload. Each increment of the subrate value corresponds to a rate increment of about 0.5 Mbps.
- For Larscom CSUs, specify the subrate as a number from 1 through 14 that exactly matches the value configured on the CSU. A subrate value of 14 corresponds to 44.2 Mbps, or 100 percent of the HDLC-encapsulated payload. A subrate value of 1

corresponds to $44.2 / 14$, which is 3.16 Mbps, 7.15 percent of the HDLC-encapsulated payload.

- For Verilink CSUs, specify the subrate as a number from 1 through 28 that exactly matches the value configured on the CSU. To calculate the maximum allowable peak rate, multiply the configured subrate by 1.578 Mbps. For example, a subrate value of 28 corresponds to 28×1.578 Mbps, which is 44.2 Mbps, 100 percent of the HDLC-encapsulated payload. A subrate value of 1 corresponds to 1.578 Mbps, 3.57 percent of the HDLC-encapsulated payload. A subrate value of 20 corresponds to 20×1.578 Mbps, which is 31.56 Mbps, 71.42 percent of the HDLC-encapsulated payload.



NOTE: Verilink configuration is not functional if an IQ interface is paired with an IQE interface.

Verilink configuration on an IQE PIC is also not functional when the PIC is connected to any other vendor equipment that operates in Verilink Port B mode. The Verilink configuration on an IQE PIC works only when it is paired with another IQE PIC or any other vendor equipment that operates in Port A mode.

Table 17: Subrate Values for T3 Digital Link Compatibility Mode

301 Kbps	9.3 Mbps	18.3 Mbps	27.4 Mbps	36.4 Mbps
601 Kbps	9.6 Mbps	18.6 Mbps	27.7 Mbps	36.7 Mbps
902 Kbps	9.9 Mbps	18.9 Mbps	28.0 Mbps	37.0 Mbps
1.2 Mbps	10.2 Mbps	19.2 Mbps	28.3 Mbps	37.3 Mbps
1.5 Mbps	10.5 Mbps	19.5 Mbps	28.6 Mbps	37.6 Mbps
1.8 Mbps	10.8 Mbps	19.8 Mbps	28.9 Mbps	37.9 Mbps
2.1 Mbps	11.1 Mbps	20.1 Mbps	29.2 Mbps	38.2 Mbps
2.4 Mbps	11.4 Mbps	20.5 Mbps	29.5 Mbps	38.5 Mbps
2.7 Mbps	11.7 Mbps	20.8 Mbps	29.8 Mbps	38.8 Mbps
3.0 Mbps	12.0 Mbps	21.1 Mbps	30.1 Mbps	39.1 Mbps
3.3 Mbps	12.3 Mbps	21.4 Mbps	30.4 Mbps	39.4 Mbps
3.6 Mbps	12.6 Mbps	21.7 Mbps	30.7 Mbps	39.7 Mbps
3.9 Mbps	12.9 Mbps	22.0 Mbps	31.0 Mbps	40.0 Mbps
4.2 Mbps	13.2 Mbps	22.3 Mbps	31.3 Mbps	40.3 Mbps

Table 17: Subrate Values for T3 Digital Link Compatibility Mode (*continued*)

4.5 Mbps	13.5 Mbps	22.6 Mbps	31.6 Mbps	40.6 Mbps
4.8 Mbps	13.8 Mbps	22.9 Mbps	31.9 Mbps	40.9 Mbps
5.1 Mbps	14.1 Mbps	23.2 Mbps	32.2 Mbps	41.2 Mbps
5.4 Mbps	14.4 Mbps	23.5 Mbps	32.5 Mbps	41.5 Mbps
5.7 Mbps	14.7 Mbps	23.8 Mbps	32.8 Mbps	41.8 Mbps
6.0 Mbps	15.0 Mbps	24.1 Mbps	33.1 Mbps	42.1 Mbps
6.3 Mbps	15.3 Mbps	24.4 Mbps	33.4 Mbps	42.4 Mbps
6.6 Mbps	15.6 Mbps	24.7 Mbps	33.7 Mbps	42.7 Mbps
6.9 Mbps	15.9 Mbps	25.0 Mbps	34.0 Mbps	43.0 Mbps
7.2 Mbps	16.2 Mbps	25.3 Mbps	34.3 Mbps	43.3 Mbps
7.5 Mbps	16.5 Mbps	25.6 Mbps	34.6 Mbps	43.6 Mbps
7.8 Mbps	16.8 Mbps	25.9 Mbps	34.9 Mbps	43.9 Mbps
8.1 Mbps	17.1 Mbps	26.2 Mbps	35.2 Mbps	44.2 Mbps
8.4 Mbps	17.4 Mbps	26.5 Mbps	35.5 Mbps	
8.7 Mbps	17.7 Mbps	26.8 Mbps	35.8 Mbps	
9.0 Mbps	18.0 Mbps	27.1 Mbps	36.1 Mbps	

For information about subrating an E3 interface, see “Configuring the E3 CSU Compatibility Mode” on page 171.

Configuring the T3 Frame Checksum

By default, T3 interfaces use a 16-bit frame checksum. You can configure a 32-bit checksum, which provides more reliable packet verification. However, some older equipment might not support 32-bit checksums.

On a channelized OC12 interface, the **fcs** statement is not supported. To configure FCS on each DS3 channel, you must include the **t3-options fcs** statement in the configuration for each channel.

To configure a 32-bit checksum, include the **fcs** statement at the **[edit interfaces interface-name t3-options]** hierarchy level:

```
[edit interfaces interface-name t3-options]
fcs 32;
```

To return to the default 16-bit frame checksum, delete the **fcs 32** statement from the configuration:

```
[edit]
user@host# delete interfaces t3-fpc/pic/port t3-options fcs 32
```

To explicitly configure a 16-bit checksum, include the **fcs** statement at the **[edit interfaces *interface-name* t3-options]** hierarchy level:

```
[edit interfaces interface-name t3-options]
fcs 16;
```

Configuring the T3 FEAC Response

The T3 far-end alarm and control (FEAC) signal is used to send alarm or status information from the far-end terminal back to the near-end terminal and to initiate T3 loopbacks at the far-end terminal from the near-end terminal.

By default, the router does not respond to FEAC requests. To allow the remote CSU to place the local router into loopback, you must configure the router to respond to the CSU's FEAC request by including the **feac-loop-respond** statement at the **[edit interfaces *interface-name* t3-options]** hierarchy level:

```
[edit interfaces interface-name t3-options]
feac-loop-respond;
```

If you configure remote or local loopback with the T3 **loopback** statement, the router does not respond to FEAC requests from the CSU even if you include the **feac-loop-respond** statement in the configuration. For the router to respond, you must delete the **loopback** statement from the configuration.

To explicitly configure the router not to respond to FEAC requests, include the **no-feac-loop** statement at the **[edit interfaces *interface-name* t3-options]** hierarchy level:

```
[edit interfaces interface-name t3-options]
no-feac-loop-respond;
```

Configuring the T3 Idle Cycle Flag

By default, a T3 interface transmits the value 0x7E in the idle cycles. To have the interface transmit the value 0xFF (all ones) instead, include the **idle-cycle-flag** statement at the **[edit interfaces *interface-name* t3-options]** hierarchy level, specifying the **ones** option:

```
[edit interfaces interface-name t3-options]
idle-cycle-flag ones;
```

To explicitly configure the default value of 0x7E, include the **idle-cycle-flag** statement with the **flags** option:

```
[edit interfaces interface-name t3-options]
idle-cycle-flag flags;
```

Configuring the T3 Line Buildout

A T3 interface has two settings for the T3 line buildout: a short setting, which is less than 255 feet (about 68 meters), and a long setting, which is greater than 255 feet and less than 450 feet (about 137 meters). By default, the interface uses the short setting.

The **long-buildout** and **no-long-buildout** statements apply only to copper-cable-based T3 interfaces. You cannot configure a line buildout for a DS3 channel on a channelized OC12 interface, which runs over fiber-optic cable. If you configure this statement on a channelized OC12 interface, it is ignored.

To have the interface drive a line that is longer than 255 feet and shorter than 450 feet, include the **long-buildout** statement at the **[edit interfaces *interface-name* t3-options]** hierarchy level:

```
[edit interfaces interface-name t3-options]
long-buildout;
```

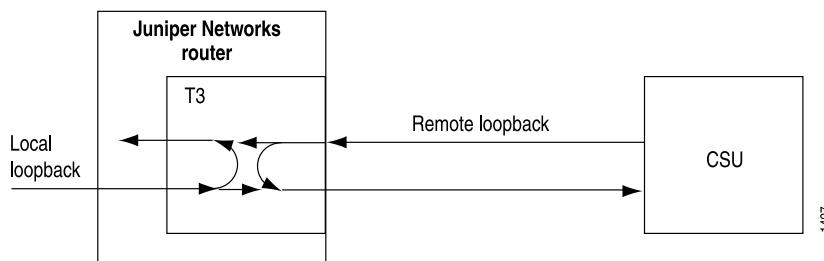
To explicitly configure the default short line buildout, include the **no-long-buildout** statement at the **[edit interfaces *interface-name* t3-options]** hierarchy level:

```
[edit interfaces interface-name t3-options]
no-long-buildout;
```

Configuring T3 Loopback Capability

You can configure loopback capability between the local T3 interface and the remote CSU, as shown in Figure 32 on page 194. You can configure the loopback to be local or remote. With local loopback, the T3 interface can transmit packets to the CSU, but receives its own transmission back again and ignores data from the CSU. With remote loopback, packets sent from the CSU are received by the T3 interface, forwarded if there is a valid route, and immediately retransmitted to the CSU.

Figure 32: Remote and Local T3 Loopback



To configure loopback capability on a T3 interface, include the **loopback** statement at the **[edit interfaces *interface-name* t3-options]** hierarchy level:

```
[edit interfaces interface-name t3-options]
loopback (local | payload | remote);
```

Packets can be looped on either the local router or the remote CSU. Local and remote loopback loop back both data and clocking information.

To exchange BERT patterns between a local router and a remote router, include the **loopback remote** statement in the interface configuration at the remote end of the link. From the local router, you issue the **test interface** command.

For more information about configuring BERT, see Interface Diagnostics. For more information about using operational mode commands to test interfaces, see the *Junos OS System Basics and Services Command Reference*.

For channelized T3, T1, and NxDS0 IQ interfaces only, you can include the **loopback payload** statement in the configuration to loop back data only (without clocking information) on the remote router's PIC. In payload loopback, overhead is recalculated. For T3 IQ interfaces, you can include the **loopback payload** statement at the **[edit interfaces ct3-fpc/pic/port]** and **[edit interfaces t3-fpc/pic/port:channel]** hierarchy levels. For T1 interfaces, you can include the **loopback payload** statement in the configuration at the **[edit interfaces t1-fpc/pic/port:channel]** hierarchy level; it is ignored if included at the **[edit interfaces ct1-fpc/pic/port]** hierarchy level. For NxDS0 interfaces, payload and remote loopback are the same. If you configure one, the other is ignored. NxDS0 IQ interfaces do not support local loopback.

To determine whether a problem is internal or external, you can loop packets on both the local and the remote router. To do this, include the **no-keepalives** and **encapsulation cisco-hdlc** statements at the **[edit interfaces interface-name]** hierarchy level and the **loopback local** statement at the **[edit interfaces interface-name t3-options]** hierarchy level, as shown in the following example:

```
[edit interfaces]
t3-1/0/0 {
  no-keepalives;
  encapsulation cisco-hdlc;
  t3-options {
    loopback local;
  }
  unit 0 {
    family inet {
      address 10.100.100.1/24;
    }
  }
}
```

With this configuration, the link stays up, so you can loop ping packets to a remote router. The **loopback local** statement causes the interface to loop within the PIC just before the data reaches the transceiver.

To turn off the loopback capability, remove the **loopback** statement from the configuration:

```
[edit]
user@host# delete interfaces t3-fpc/pic/port t3-options loopback
```

You can determine whether there is an internal problem or an external problem by checking the error counters in the output of the **show interface interface-name extensive** command, for example:

```
user@host> show interfaces t3-fpc/pic/port extensive
```

For channel 0 on channelized interfaces only, you can include the **loopback** statement at the **[edit interfaces *interface-name* *interface-type-options*]** hierarchy level. The loopback setting configured for channel 0 applies to all channels on the channelized interface. The **loopback** statement is ignored if you include it at this hierarchy level in the configuration of other channels. To configure loopbacks on individual channels, you must include the ***channel-type-options* loopback** statement in the configuration for each channel. This allows each channel to be put in loopback mode independently.

For example, for DS3 channels on a channelized OC12 interface, the **sonet-options loopback** statement is supported only for channel 0; it is ignored if included in the configuration for channels 1 through 11. The SONET loopback configured for channel 0 applies to all 12 channels equally. To configure loopbacks on the individual DS3 channels, you must include the **t3-options loopback** statement in the configuration for each channel. This allows each DS3 channel can be put in loopback mode independently.

Configuring T3 HDLC Payload Scrambling

T3 HDLC payload scrambling, which is disabled by default, provides better link stability. Both sides of a connection must either use or not use scrambling.

On a channelized OC12 interface, the SONET **payload-scrambler** statement is ignored. To configure scrambling on the DS3 channels on the interface, you can include the **t3-options payload-scrambler** statement at the **[edit interfaces *interface-name* t3-options]** hierarchy level for each DS3 channel.

If you enable HDLC payload scrambling on a T3 interface, you must also configure the interface to be compatible with the channel service unit (CSU) at the remote end of the line before you commit the interface configuration. For information about subrating a T3 interface, see “Configuring the T3 CSU Compatibility Mode” on page 190.

```
[edit interfaces interface-name t3-options]
  compatibility-mode (adtran | digital-link | kentrox | larscom | verilink) <subrate value>;
  payload-scrambler;
```

To explicitly disable HDLC payload scrambling, include the **no-payload-scrambler** statement at the **[edit interfaces *interface-name* t3-options]** hierarchy level:

```
[edit interfaces interface-name t3-options]
  no-payload-scrambler;
```

To disable payload scrambling again (return to the default), delete the **payload-scrambler** statement from the configuration:

```
[edit]
user@host# delete interfaces t3-fpc/port t3-options payload-scrambler
```

Configuring T3 Start and End Flags

By default, a T3 interface shares the transmission of the start and end flags.

To configure a T3 interface to wait two idle cycles between the start and end flags, include the **start-end-flag** statement with the **filler** option at the **[edit interfaces *interface-name* t3-options]** hierarchy level:


```
[edit interfaces interface-name t3-options]
start-end-flag filler;
```

To revert to the default behavior, sharing the transmission of start and end flags, include the **start-end-flag** statement with the **shared** option at the **[edit interfaces *interface-name* t3-options]** hierarchy level:

```
[edit interfaces interface-name t3-options]
start-end-flag shared;
```

Examples: Configuring T3 Interfaces

T3 interfaces can use PPP, Cisco HDLC, or Frame Relay encapsulation.

PPP Encapsulation on a DS3 PIC

```
[edit]
interfaces {
  t3-0/0/0 {
    encapsulation ppp;
    t3-options {
      no-long-buildout;
      compatibility-mode larscom;
      payload-scrambler;
    }
    unit 0 {
      family inet {
        address 10.0.0.1/32 {
          destination 10.0.0.2;
        }
      }
      family iso;
    }
  }
}
```

Cisco HDLC Encapsulation on a DS3 PIC

```
[edit]
interfaces {
  t3-0/0/1 {
    encapsulation cisco-hdlc;
    t3-options {
      no-long-buildout;
      compatibility-mode larscom;
      payload-scrambler;
    }
    unit 0 {
      family inet {
        address 10.0.0.1/32 {
          destination 10.0.0.2;
        }
      }
      family iso;
    }
  }
}
```

Configure Frame Relay encapsulation on two routers, where one router is a DTE device and the other is a DCE device:

On DTE Router

```
[edit]
interfaces {
  t3-1/0/1 {
    encapsulation frame-relay;
    t3-options {
      no-long-buildout;
      compatibility-mode larscom;
      payload-scrambler;
    }
    unit 1 {
      dlci 1;
      family inet {
        address 10.0.0.1/32 {
          destination 10.0.0.2;
        }
      }
      family iso;
    }
    unit 2 {
      dlci 2;
      family inet {
        address 10.0.0.3/32 {
          destination 10.0.0.4;
        }
      }
      family iso;
    }
  }
}
```

On DCE Router

```
[edit]
interfaces {
  t3-1/1/1 {
    dce;
    encapsulation frame-relay;
    t3-options {
      no-long-buildout;
      compatibility-mode larscom;
      payload-scrambler;
    }
    unit 1 {
      dlci 1;
      family inet {
        address 10.0.0.2/32 {
          destination 10.0.0.1;
        }
      }
      family iso;
    }
    unit 2 {
      dlci 2;
      family inet {
        address 10.0.0.4/32 {
          destination 10.0.0.3;
        }
      }
    }
  }
}
```

```
    }  
    family iso;  
  }  
}
```


PART 4

Configuring Frame Relay

- [Configuring Frame Relay on page 203](#)

Configuring Frame Relay

- Frame Relay Overview on page 203
- Configuring Frame Relay Interface Encapsulation on page 204
- Configuring Frame Relay Control Bit Translation on page 208
- Configuring the Media MTU on Frame Relay Interfaces on page 209
- Setting the Protocol MTU with Frame Relay Encapsulation on page 209
- Configuring Frame Relay Keepalives on page 210
- Configuring Inverse Frame Relay ARP on page 211
- Configuring the Router as a DCE with Frame Relay Encapsulation on page 212
- Configuring Frame Relay DLCIs on page 212

Frame Relay Overview

The Frame Relay protocol allows network designers to reduce costs by using shared facilities that are managed by a Frame Relay service provider. Users pay fixed charges for the local connections from each site in the Frame Relay network to the first point of presence (POP) in which the provider maintains a Frame Relay switch. The portion of the network between the endpoint switches is shared by all the customers of the service provider, and individual data-link connection identifiers (DLCIs) are assigned to ensure that each customer receives only their own traffic.

Users contract with their providers for a specific minimum portion of the shared bandwidth committed information rate (CIR) and for a maximum allowable peak rate, burst information rate (BIR). Depending on the terms of the contract, traffic exceeding the CIR can be marked as eligible for discard, in the event of network congestion, or a best-effort term can apply up to the BIR rate.

Frame Relay does not require private and permanently connected wide area network facilities, unlike some older WAN protocols.

Frame Relay was developed as a replacement for the older and much slower X.25 protocol. It scales to much higher data rates because it does not require explicit acknowledgment of each frame of data.

You can configure the Frame Relay protocol on SONET/SDH, E1/E3, and T1/T3 physical router interfaces, and on the channelized DS3, channelized OC12, channelized T3 intelligent queuing (IQ), channelized OC12 IQ, and channelized E1 IQ interfaces.

Starting with Junos OS Release 11.2, multiple DLCIs are supported on a Frame Relay interface for End-to-End Multilink Frame Relay Implementation Agreement (FRF.15) bundles. Each DLCI should be part of a unique bundle and it is not possible to have more than one DLCI from the same Frame Relay interface in the same bundle. This feature enables you to have FRF.12 functionality over multiple DLCIs per Frame Relay interface.



NOTE: This capability is available on all M Series and MX Series routers supporting Layer 2 services.

All Multiservices PICs and Multiservices Dense Port Concentrators support this feature. But Adaptive Services PICs (AS PICs) do not support the feature. DLCIs having different bandwidths that are part of the same bundle are also not supported.

Configuring Frame Relay Interface Encapsulation

Point-to-Point Protocol (PPP) encapsulation is the default encapsulation type for physical interfaces. You need not configure encapsulation for any physical interfaces that support PPP encapsulation. If you do not configure encapsulation, PPP is used by default. For physical interfaces that do not support PPP encapsulation, you must configure an encapsulation to use for packets transmitted on the interface. You can optionally configure an encapsulation on a logical interface, which is the encapsulation used within certain packet types.

For more information, see the following sections:

- Configuring the Frame Relay Encapsulation on a Physical Interface on page 204
- Configuring the Frame Relay Encapsulation on a Logical Interface on page 207

Configuring the Frame Relay Encapsulation on a Physical Interface

For Frame Relay interfaces, configure Frame Relay encapsulation on the physical interface. This encapsulation is defined in RFC 1490, *Multiprotocol Interconnect over Frame Relay*. SONET/SDH and T3 interfaces can use Frame Relay encapsulation.

To configure Frame Relay encapsulation on a physical interface, include the **encapsulation** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]  
  encapsulation type;
```

When you configure a multipoint encapsulation (such as Frame Relay), the physical interface can have multiple logical units, and the units can be either point-to-point or multipoint.

The encapsulation type can be one of the following:

- Flexible Frame Relay (**flexible-frame-relay**)—IQ interfaces can use flexible Frame Relay encapsulation. You use flexible Frame Relay encapsulation when you want to configure multiple per-unit Frame Relay encapsulations. This encapsulation type allows you to configure any combination of TCC, CCC, and standard Frame Relay encapsulations on a single physical port. Also, each logical interface can have any DLCI value from 1 through 1022.
- Frame Relay (**frame-relay**)—Defined in RFC 1490. E1, E3, link services, SONET/SDH, T1, T3, and voice services interfaces can use Frame Relay encapsulation. Five related versions are supported:
 - Circuit cross-connect (CCC) version (**frame-relay-ccc**)—The same as standard Frame Relay for DLCIs 0 through 511. DLCIs 512 through 1022 are dedicated to CCC. The logical interface must also have **frame-relay-ccc** encapsulation. When you use this encapsulation type, you can configure the **ccc** family only.
 - Translational cross-connect (TCC) version (**frame-relay-tcc**)—Similar to Frame Relay CCC and has the same configuration restrictions, but used for circuits with different media on either side of the connection.
 - Extended CCC version (**extended-frame-relay-ccc**)—This encapsulation type allows you to dedicate DLCIs 1 through 1022 to CCC. The logical interface must have **frame-relay-ccc** encapsulation. When you use this encapsulation type, you can configure the **ccc** family only.
 - Extended TCC version (**extended-frame-relay-tcc**)—Similar to extended Frame Relay CCC, this encapsulation type allows you to dedicate DLCIs 1 through 1022 to TCC, which is used for circuits with different media on either side of the connection.
 - Port CCC version (**frame-relay-port-ccc**)—Defined in the Internet Engineering Task Force (IETF) document, *Frame Relay Encapsulation over Pseudo-Wires* (expired December 2002). This encapsulation type allows you to transparently carry all the DLCIs between two customer edge (CE) routers without explicitly configuring each DLCI on the two provider edge (PE) routers with Frame Relay transport. The connection between the two CE routers can be either user-to-network interface (UNI) or network-to-network interface (NNI); this is completely transparent to the PE routers. The logical interface does not require an encapsulation statement. When you use this encapsulation type, you can configure the **ccc** family only.
- Frame Relay Ether Type (**frame-relay-ether-type**)—Physical interfaces can use Frame Relay ether type encapsulation for compatibility with Cisco Frame Relay. IETF frame relay encapsulation identifies the payload format using NLPID and SNAP formats. Cisco-compatible Frame Relay encapsulation uses the Ethernet type to identify the type of payload. Two related versions are supported:
 - TCC version (**frame-relay-ether-type-tcc**)—Cisco-compatible Frame Relay for DLCIs 0 through 511. DLCIs 512 through 1022 are dedicated to TCC. This encapsulation is used for circuits with different media on either side of the connection.
 - Extended TCC version (**extended-frame-relay-ether-type-tcc**)—This encapsulation allows you to dedicate Cisco-compatible Frame Relay TCC for DLCIs 1 through 1022. This encapsulation is used for circuits with different media on either side of the

connection. Extended Frame Relay ether type TCC encapsulation is supported on the same PICs as extended Frame Relay TCC encapsulation.



NOTE: When the encapsulation type is set to Cisco-compatible Frame Relay encapsulation, ensure that the LMI type is set to ANSI or Q933-A.

Support for extended Frame Relay and flexible Frame Relay differs by PIC type, as shown in Table 18 on page 206.

Table 18: PIC Support for Enhanced Frame Relay Encapsulation Types

PIC Type	Extended Frame Relay CCC	Extended Frame Relay TCC	Flexible Frame Relay
Intelligent Queuing			
1-port Channelized CHOC12 IQ	Yes	Yes	Yes
4-port Channelized DS3 IQ	Yes	Yes	Yes
10-port Channelized E1 IQ	Yes	Yes	Yes
4-port E3 IQ	Yes	Yes	Yes
1-port Channelized STM1 IQ	Yes	Yes	Yes
4-port OC48/STM16 Type 3 PIC	Yes	Yes	Yes
SONET/SDH			
1-port OC12	Yes	Yes	No
2-port OC3	Yes	Yes	No
1-port OC48	Yes	Yes	No
1-port OC192	Yes	Yes	No
1-port STM16 SDH, SMSR	Yes	Yes	No
Others			
4-port E1	No	No	No
4-port T1	No	No	No
4-port T3	No	No	No
10-port Channelized E1	No	No	No

Table 18: PIC Support for Enhanced Frame Relay Encapsulation Types (*continued*)

PIC Type	Extended Frame Relay CCC	Extended Frame Relay TCC	Flexible Frame Relay
2-port Channelized DS3	No	No	No
1-port Channelized OC12, SMIR	No	No	No
4-port Channelized DS3	No	No	No
1-port Channelized STM1, SMIR	No	No	No
2-port Serial	No	No	No

Example: Configuring the Encapsulation on a Physical Interface

Configure Frame Relay encapsulation on a SONET/SDH interface. The second and third **family** statements allow Intermediate System-to-Intermediate System (IS-IS) and Multiprotocol Label Switching (MPLS) to run on the interface.

```
[edit interfaces]
so-7/0/0 {
  encapsulation frame-relay;
  unit 0 {
    point-to-point;
    family inet {
      address 192.168.1.113/32 {
        destination 192.168.1.114;
      }
    }
    family iso;
    family mpls;
  }
}
```

Configuring the Frame Relay Encapsulation on a Logical Interface

Generally, you configure an interface's encapsulation at the **[edit interfaces *interface-name*]** hierarchy level. However, for Frame Relay encapsulation, you can also configure the encapsulation type that is used inside the Frame Relay packet itself. To do this, include the **encapsulation** statement, specifying the **frame-relay-ccc**, **frame-relay-ppp**, **frame-relay-tcc**, **frame-relay-ether-type**, or **frame-relay-ether-type-tcc** option:

```
encapsulation (frame-relay-ccc | frame-relay-ppp | frame-relay-tcc |
  frame-relay-ether-type | frame-relay-ether-type-tcc);
```

You can include this statement at the following hierarchy levels:

- **[edit interfaces *interface-name* unit *logical-unit-number*]**
- **[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*]**

Configuring Frame Relay Control Bit Translation

On interfaces with Frame Relay CCC encapsulation, you can configure Frame Relay control bit translation, as defined in the IETF documents:

- Internet draft draft-martini-frame-encap-mpls-00.txt, *Frame Relay Encapsulation over Pseudo-Wires* (expired December 2002)
- Internet draft draft-martini-l2circuit-encap-mpls-07.txt, *Encapsulation Methods for Transport of Layer 2 Frames Over IP and MPLS Networks* (expired December 2004)

To support Frame Relay services over IP and MPLS backbones using Layer 2 VPNs and Layer 2 circuits, you can configure translation of the Frame Relay control bits. When you configure translation of Frame Relay control bits, the bits are mapped into the Layer 2 circuit control word and preserved across the IP or MPLS backbone.

The Junos OS allows you to translate the following Frame Relay control bits:

- Discard eligibility (DE)—A header bit used to identify lower priority traffic that can be dropped during periods of congestion.
- Forward explicit congestion notification (FECN)—A header bit transmitted by the source router requesting that the destination router slow down its requests for data.
- Backward explicit congestion notification (BECN)—A header bit transmitted by the destination router requesting that the source router send data more slowly.

By default, translation of Frame Relay control bits is disabled. If you enable Frame Relay control bit translation, the bits are translated in both directions (CE to PE and PE to CE):

- From CE to PE—At ingress, the DE, FECN, and BECN header bits from the incoming Frame Relay header are mapped to the control word.
- From PE to CE—At egress, the DE, FECN, and BECN header bits from the control word are mapped to the outgoing Frame Relay header.

The Frame Relay control bits do not map to MPLS EXP labels, and do not affect class-of-service (CoS) behavior inside the provider network.

You enable or explicitly disable translation of Frame Relay control bits by including the **translate-discard-eligible** and **translate-fecn-and-becn** statements:

```
(translate-discard-eligible | no-translate-discard-eligible);  
(translate-fecn-and-becn | no-translate-fecn-and-becn);
```

You can include these statements at the following hierarchy levels:

- [edit interfaces *interface-name* unit *logical-unit-number* family ccc]
- [edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* family ccc]

If you enable or disable Frame Relay control bit translation on one CE-facing interface, you must configure the same Frame Relay control bit translation settings on the other CE-facing interface.

If you change the Frame Relay control bit translation settings, the circuit goes down and comes back up, which might result in traffic loss for a few seconds.

If you enable Frame Relay control bit translation, the number of supportable Layer 2 virtual private networks (VPNs) and Layer 2 circuits is reduced to one eighth of what the router can support without Frame Relay control bit translation enabled.

For ATM2 IQ interfaces, the control word contains a field to carry ATM cell loss priority (CLP) information by default. For more information, see *Configuring Layer 2 Circuit Transport Mode*.

For more information about Layer 2 circuits, see the *Junos OS VPNs Configuration Guide* and the *Junos OS Routing Protocols Configuration Guide*. For a comprehensive example, see the *Junos OS Feature Guides*.

Configuring the Media MTU on Frame Relay Interfaces

For Frame Relay interfaces, the default media maximum transmission unit (MTU) is 4482 bytes. (For a complete list of MTU values, see *Configuring the Media MTU*.)

To modify the default media MTU size for a physical interface, include the **mtu** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]  
mtu bytes;
```

If you change the size of the media MTU, you must ensure that the size is equal to or greater than the sum of the protocol MTU and the encapsulation overhead. You can include the **mtu** statement at the following hierarchy levels:

- **[edit interfaces *interface-name* unit *logical-unit-number* family *family*]**
- **[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* family *family*]**

For more information, see “Setting the Protocol MTU with Frame Relay Encapsulation” on page 209.

Setting the Protocol MTU with Frame Relay Encapsulation

For each interface, you can configure an interface-specific MTU by including the **mtu** statement at the **[edit interfaces *interface-name*]** hierarchy level. If you need to modify this MTU for a particular protocol family, include the **mtu** statement:

```
mtu mtu;
```

You can include this statement at the following hierarchy levels:

- **[edit interfaces *interface-name* unit *logical-unit-number* family *family*]**

- **[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* family *family*]**

For Frame Relay encapsulation, the default protocol MTU is 4470 bytes.

If you increase the size of the protocol MTU, you must ensure that the size of the media MTU is equal to or greater than the sum of the protocol MTU and the encapsulation overhead. (You configure the media MTU by including the **mtu** statement at the **[edit interfaces *interface-name*]** hierarchy level, as discussed in “Configuring the Media MTU on Frame Relay Interfaces” on page 209.)

When the family is **mpls**, the default protocol MTU is 1488 bytes. MPLS packets are 1500 bytes and have 4 to 12 bytes of overhead.

Configuring Frame Relay Keepalives

By default, physical interfaces configured with Cisco High-level Data Link Control (HDLC) or Point-to-Point Protocol (PPP) encapsulation send keepalive packets at 10-second intervals. The Frame Relay term for keepalives is Local Management Interface (LMI) packets; note that the Junos OS supports both ANSI T1.617 Annex D LMIs, ITU Q933 Annex A LMIs and Consortium LMI.

Consortium LMI is supported only on M320 Routers with Enhanced III FPCs and specific Enhanced Intelligent Queuing PICs.

To disable the sending of keepalives on a physical interface, include the **no-keepalives** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]  
no-keepalives;
```

For back-to-back Frame Relay connections, either disable the sending of keepalives on both sides of the connection, or configure one side of the connection as a data terminal equipment (DTE) (the default Junos configuration) and the other as a data circuit-terminating equipment (DCE).

If keepalives are enabled, the number of possible DLCI configurations on a multipoint or multicast connection is limited by the MTU size selected for the interface. To calculate the available DLCIs, use the formula $(MTU - 12) / 5$. To increase the number of possible DLCIs, disable keepalives.

Configuring Tunable Keepalives for Frame Relay LMI

On interfaces configured with Frame Relay connections, you can tune the keepalive settings by using the **lmi** statement. A Frame Relay interface can be either DCE or DTE (the default Junos configuration). DTE acts as a master, requesting status from the DCE part of the link.

By default, the Junos OS uses ANSI T1.617 Annex D LMIs. To change to ITU Q933 Annex A LMIs, include the **lmi-type itu** statement at the **[edit interfaces *interface-name* lmi]** hierarchy level:

```
[edit interfaces interface-name lmi]
```

```
lmi-type itu;
```

To configure Frame Relay keepalive parameters, include the **lmi** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]
lmi {
  lmi-type (ansi | itu);
  n391dte number;
  n392dce number;
  n392dte number;
  n393dce number;
  n393dte number;
  t391dte seconds;
  t392dce seconds;
}
```

You can include the following statements:

- **n391dte**—DTE full status polling interval. The DTE sends a status inquiry to the DCE at the interval specified by **t391dte**. **n391dte** specifies the frequency at which these inquiries expect a full status report; for example, a **n391dte** value of 10 would specify a full status report in response to every tenth inquiry. The intermediate inquiries ask for a keepalive exchange only. The range is from 1 through 255, with a default value of 6.
- **n392dce**—DCE error threshold. The number of errors required to bring down the link, within the event-count specified by **n393dce**. The range is from 1 through 10, with a default value of 3.
- **n392dte**—DTE error threshold. The number of errors required to bring down the link, within the event-count specified by **n393dte**. The range is from 1 through 10, with a default value of 3.
- **n393dce**—DCE monitored event-count. The range is from 1 through 10, with a default value of 4.
- **n393dte**—DTE monitored event-count. The range is from 1 through 10, with a default value of 4.
- **t391dte**—DTE keepalive timer. Period at which the DTE sends out a keepalive response request to the DCE and updates status depending on the DTE error threshold value. The range is from 5 through 30 seconds, with a default value of 10 seconds.
- **t392dce**—DCE keepalive timer. Period at which the DCE checks for keepalive responses from the DTE and updates status depending on the DCE error threshold value. The range is from 5 through 30 seconds, with a default value of 15 seconds.

Configuring Inverse Frame Relay ARP

Frame Relay interfaces support inverse Frame Relay ARP, as described in RFC 2390, *Inverse Address Resolution Protocol*. When inverse Frame Relay ARP is enabled, the router responds to received inverse Frame Relay ARP requests by providing IP address information to the requesting router on the other end of the Frame permanent virtual circuit (PVC).

The router does not initiate inverse Frame Relay ARP requests.

By default, inverse Frame Relay ARP is disabled. To configure a router to respond to inverse Frame Relay ARP requests, include the **inverse-arp** statement:

```
inverse-arp;
```

For a list of hierarchy levels at which you can include this statement, see **inverse-arp**.

You must configure Frame Relay encapsulation on the logical interface to support inverse ARP. For more information, see “Configuring Frame Relay Interface Encapsulation” on page 204.

Configuring the Router as a DCE with Frame Relay Encapsulation

By default, when you configure an interface with Frame Relay encapsulation, the routing platform is assumed to be DTE. That is, the routing platform is assumed to be at a terminal point on the network. To configure the routing platform to be DCE, include the **dce** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]  
dce;
```

When you configure the router to be a DCE, keepalives are disabled by default.

For back-to-back Frame Relay connections, either disable the sending of keepalives on both sides of the connection, or configure one side of the connection as a DCE and the other as a DTE by removing the **dce** statement from the configuration (the default Junos configuration).

Configuring Frame Relay DLCIs

When you are using Frame Relay encapsulation on an interface, each logical interface corresponds to one or more permanent virtual circuits (PVCs) or switched virtual circuits (SVCs). For each PVC or SVC, you must configure one data-link connection identifier (DLCI).

A Frame Relay interface can be a point-to-point interface or a point-to-multipoint (also called a multipoint non-broadcast multiaccess [NBMA]) connection.

To configure Frame Relay DLCIs, you can do the following:

- Configuring a Point-to-Point Frame Relay Connection on page 212
- Configuring a Point-to-Multipoint Frame Relay Connection on page 213
- Configuring a Multicast-Capable Frame Relay Connection on page 214

Configuring a Point-to-Point Frame Relay Connection

To configure a point-to-point Frame Relay connection, include the **dlci** statement:

```
dlci dlci-identifier;
```

You can include this statement at the following hierarchy levels:

- [edit interfaces *interface-name* unit *logical-unit-number*]
- [edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*]

The DLCI identifier is a value from 16 through 1022. Numbers 1 through 15 are reserved for future use. A point-to-point interface can have one DLCI.



NOTE: For information about Frame Relay DLCI limitations for channelized interfaces, see “Data-Link Connection Identifiers on Channelized Interfaces” on page 30.

You configure the router to use DLCI sparse mode by including the `sparse-dlcis` statement at the [edit chassis fpc *slot-number* pic *pic-number*] hierarchy level. For more information about DLCI sparse mode, see the [Junos OS System Basics Configuration Guide](#).

For more information about Frame Relay DLCIs, see “Configuring a Point-to-Point Frame Relay Connection” on page 212.

When you are configuring point-to-point connections, the MTU sizes on both sides of the connection must be the same.

Configuring a Point-to-Multipoint Frame Relay Connection

To configure a point-to-multipoint Frame Relay connection (also called a multipoint NBMA connection), include the `multipoint-destination` statement:

```
multipoint-destination address dlci dlci-identifier;
```

You can include this statement at the following hierarchy levels:

- [edit interfaces *interface-name* unit *logical-unit-number* family *family* address *address*]
- [edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* family *family* address *address*]

For each destination, include one `multipoint-destination` statement. **address** is the address of the remote side of the connection, and **dlci-identifier** is the DLCI identifier for the connection.

When you are configuring point-to-multipoint connections, all interfaces in the subnet must use the same MTU size.

If keepalives are enabled, causing the interface to send LMI messages during idle times, the number of possible DLCI configurations is limited by the MTU selected for the interface. For more information, see “Configuring Frame Relay Keepalives” on page 210.

Configuring a Multicast-Capable Frame Relay Connection

By default, Frame Relay connections assume unicast traffic. If your Frame Relay switch performs multicast replication, you can configure the connection to support multicast traffic by including the **multicast-dlci** statement:

```
multicast-dlci dlci-identifier;
```

You can include this statement at the following hierarchy levels:

- [edit interfaces *interface-name* unit *logical-unit-number*]
- [edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*]

The DLCI identifier is a value from 16 through 1022 that defines the Frame Relay DLCI over which the switch expects to receive multicast packets for replication.

You can configure multicast support only on point-to-multipoint Frame Relay connections.

If keepalives are enabled, causing the interface to send LMI messages during idle times, the number of possible DLCI configurations is limited by the MTU selected for the interface. For more information, see “Configuring Frame Relay Keepalives” on page 210.

PART 5

Channelized Interface Configuration Statements

- Summary of Channelized Interface Configuration Statements on page 217

CHAPTER 16

Summary of Channelized Interface Configuration Statements

The following descriptions explain each of the interface configuration statements. The statements are organized alphabetically.

address

```

Syntax  address address {
        arp ip-address (mac | multicast-mac) mac-address <publish>;
        broadcast address;
        destination address;
        destination-profile name;
        eui-64;
        master-only;
        multipoint-destination address dlcid dlcid-identifier;
        multipoint-destination address {
            epd-threshold cells;
            inverse-arp;
            oam-liveness {
                up-count cells;
                down-count cells;
            }
            oam-period (disable | seconds);
            shaping {
                (cbr rate | rtvbr peak rate sustained rate burst length | vbr peak rate sustained rate burst
                 length);
                queue-length number;
            }
            vci vpi-identifier.vci-identifier;
        }
        primary;
        preferred;
        (vrrp-group | vrrp-inet6-group) group-number {
            (accept-data | no-accept-data);
            advertise-interval seconds;
            authentication-type authentication;
            authentication-key key;
            fast-interval milliseconds;
            (preempt | no-preempt) {
                hold-time seconds;
            }
            priority-number number;
            track {
                priority-cost seconds;
                priority-hold-time interface-name {
                    interface priority;
                    bandwidth-threshold bits-per-second {
                        priority;
                    }
                }
            }
            route ip-address/mask routing-instance instance-name priority-cost cost;
        }
        virtual-address [ addresses ];
    }
}

```

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number* family *family*],
 [edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*
 family *family*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for QFX Series switches.

Description Configure the interface address.



NOTE: The vrrp High Availability functionality is not available for the QFX Series switches

Options *address*—Address of the interface.

The remaining statements are explained separately.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring the Protocol Family
- negotiate-address
- unnumbered-address (Ethernet)
- *Junos OS System Basics Configuration Guide*

advertise-interval

Syntax *advertise-interval milliseconds;*

Hierarchy Level [edit interfaces *interface-name* sonet-options *aps*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Modify the Automatic Protection Switching (APS) interval at which the protect and working routers send packets to their neighbors to advertise that they are operational. A router considers its neighbor to be operational for a period, called the hold time, that is, by default, three times the advertisement interval.

Options *milliseconds*—Interval between advertisement packets.
Range: 1 through 65,534 milliseconds
Default: 1000 milliseconds

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring APS Timers

aps

Syntax `aps {
 advertise-interval milliseconds;
 annex-b
 authentication-key key;
 force;
 hold-time milliseconds;
 lockout;
 neighbor address;
 paired-group group-name;
 preserve-interface;
 protect-circuit group-name;
 request;
 revert-time seconds;
 switching-mode (bidirectional | unidirectional);
 working-circuit group-name;
 }`

Hierarchy Level [edit interfaces *interface-name* sonet-options]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure Automatic Protection Switching (APS) on the router.

For DS3 channels on a channelized OC12 interface, configure APS on channel 0 only. If you configure APS on channels 1 through 11, it is ignored.

The statements are explained separately.

Required Privilege Level interface—To view this statement in the configuration.
 interface-control—To add this statement to the configuration.

Related Documentation • Configuring APS and MSP

authentication-key

Syntax	<code>authentication-key key;</code>
Hierarchy Level	<code>[edit interfaces <i>interface-name</i> sonet-options aps]</code>
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Configure the Automatic Protection Switching (APS) authentication key (password).
Options	key —Authentication password. It can be 1 through 8 characters long. Configure the same key for both the working and protect routers.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> Configuring Basic APS Support For information about the authentication-key statement at the <code>[edit interfaces <i>interface-name</i> unit <i>unit-number</i> family inet address <i>address</i> (vrrp-group vrrp-inet6-group) <i>group-number</i>]</code> or <code>[edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>unit-number</i> family (inet inet6) address <i>address</i> (vrrp-group vrrp-inet6-group) <i>group-number</i>]</code> hierarchy level, see the Junos OS High Availability Configuration Guide.

bchannel-allocation

Syntax	<code>bchannel-allocation (ascending descending);</code>
Hierarchy Level	<code>[edit interfaces <i>interface-name</i> isdn-options]</code>
Release Information	Statement introduced in Junos OS Release 8.3.
Description	(J Series Services Routers equipped with a Dual-Port Channelized T1/E1 PIM) For Integrated Services Digital Network Primary Rate Interfaces (ISDN PRI), allocate PRI dialout B-channels in ascending or descending order.
Options	(ascending descending) —Allocate the B-channels in ascending (from low to high) or descending (from high to low) order. Default: Descending order
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> Allocating B-Channels for Dialout on page 155 Junos OS Interfaces and Routing Configuration Guide

bert-algorithm

Syntax `bert-algorithm algorithm;`

Hierarchy Level `[edit interfaces ce1-fpc/pic/port],`
`[edit interfaces ct1-fpc/pic/port],`
`[edit interfaces interface-name ds0-options],`
`[edit interfaces interface-name e1-options],`
`[edit interfaces interface-name e3-options],`
`[edit interfaces interface-name t1-options],`
`[edit interfaces interface-name t3-options]`

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure the pattern to send in the bit stream during a bit error rate test (BERT). Applies to T1, E3, T3, and multichannel DS3 interfaces, the channelized interfaces (DS3, OC12, STM1), and channelized IQ and IQE interfaces (E1, E3 and DS3).



NOTE: When configuring CE1 or CT1 interfaces on 10-port Channelized E1/T1 IQE PICs, the `bert-algorithm` statement must be included at the `[edit interfaces ce1-fpc/pic/port]` or `[edit interfaces ct1-fpc/pic/port]` hierarchy level as appropriate.

Options `algorithm`—Pattern to send in the bit stream. There are two categories of test patterns: pseudorandom and repetitive. Both patterns conform to CCITT/ITU O.151, O.152, O.153, and O.161 standards. The algorithm can be one of the following patterns:

- **all-ones-repeating**—Pattern is all ones.
- **all-zeros-repeating**—Pattern is all zeros.
- **alternating-double-ones-zeros**—Pattern is alternating pairs of ones and zeros.
- **alternating-ones-zeros**—Pattern is alternating ones and zeros.
- **pseudo-2e3**—Pattern is $2^3 - 1$.
- **pseudo-2e4**—Pattern is $2^4 - 1$.
- **pseudo-2e5**—Pattern is $2^5 - 1$.
- **pseudo-2e6**—Pattern is $2^6 - 1$.
- **pseudo-2e7**—Pattern is $2^7 - 1$.
- **pseudo-2e9-o153**—Pattern is $2^9 - 1$, as defined in the O153 standard.
- **pseudo-2e10**—Pattern is $2^{10} - 1$.
- **pseudo-2e11-o152**—Pattern is $2^{11} - 1$, as defined in the O152 standard.
- **pseudo-2e15-o151**—Pattern is $2^{15} - 1$, as defined in the O151 standard.

- **pseudo-2e17**—Pattern is $2^{17} - 1$.
- **pseudo-2e18**—Pattern is $2^{18} - 1$.
- **pseudo-2e20-o151**—Pattern is $2^{20} - 1$, as defined in the O151 standard.
- **pseudo-2e20-o153**—Pattern is $2^{20} - 1$, as defined in the O153 standard.
- **pseudo-2e21**—Pattern is $2^{21} - 1$.
- **pseudo-2e22**—Pattern is $2^{22} - 1$.
- **pseudo-2e23-o151**—Pattern is $2^{23} - 1$, as defined in the O151 standard.
- **pseudo-2e25**—Pattern is $2^{25} - 1$.
- **pseudo-2e28**—Pattern is $2^{28} - 1$.
- **pseudo-2e29**—Pattern is $2^{29} - 1$.
- **pseudo-2e31**—Pattern is $2^{31} - 1$.
- **pseudo-2e32**—Pattern is $2^{32} - 1$.
- **repeating-1-in-4**—One bit in four is set to 1; the others are set to 0.
- **repeating-1-in-8**—One bit in eight is set to 1; the others are set to 0.
- **repeating-3-in-24**—Three bits in twenty four are set to 1; the others are set to 0.

Default: pseudo-2e3

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Interface Diagnostics
- Configuring E1 BERT Properties on page 162
- Configuring E3 BERT Properties on page 170
- Configuring T1 BERT Properties on page 178
- Configuring T3 BERT Properties on page 188
- Examples: Configuring T3 Interfaces on page 197
- **bert-error-rate on page 224**
- **bert-period on page 225**

bert-error-rate


Syntax	<code>bert-error-rate rate;</code>
Hierarchy Level	<code>[edit interfaces ce1-fpc/pic/port],</code> <code>[edit interfaces ct1-fpc/pic/port],</code> <code>[edit interfaces interface-name ds0-options],</code> <code>[edit interfaces interface-name e1-options],</code> <code>[edit interfaces interface-name e3-options],</code> <code>[edit interfaces interface-name t1-options],</code> <code>[edit interfaces interface-name t3-options]</code>
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Configure the bit error rate to use in a BERT procedure. Applies to E1, E3, T1, or T3 interfaces, and to the channelized interfaces (DS3, OC3, OC12, and STM1).



NOTE: When configuring CE1 or CT1 interfaces on 10-port Channelized E1/T1 IQE PICs, the `bert-error-rate` statement must be included at the `[edit interfaces ce1-fpc/pic/port]` or `[edit interfaces ct1-fpc/pic/port]` hierarchy level as appropriate.


Options	rate —Bit error rate. Range: 0 through 7, which corresponds to 10^{-1} (1 error per bit) to 10^{-7} (1 error per 10 million bits) Default: 0
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> Interface Diagnostics Configuring E1 BERT Properties on page 162 Configuring E3 BERT Properties on page 170 Configuring T1 BERT Properties on page 178 Configuring T3 BERT Properties on page 188 Examples: Configuring T3 Interfaces on page 197 bert-algorithm on page 222 bert-period on page 225

bert-period

Syntax	<code>bert-period <i>seconds</i>;</code>
Hierarchy Level	<code>[edit interfaces <i>ce1-fpc/pic/port</i>],</code> <code>[edit interfaces <i>ct1-fpc/pic/port</i>],</code> <code>[edit interfaces <i>interface-name ds0-options</i>],</code> <code>[edit interfaces <i>interface-name e1-options</i>],</code> <code>[edit interfaces <i>interface-name e3-options</i>],</code> <code>[edit interfaces <i>interface-name t1-options</i>],</code> <code>[edit interfaces <i>interface-name t3-options</i>]</code>
Release Information	Statement introduced before Junos OS Release 7.4.
Description	<p>Configure the duration of a BERT test. Applies to E1, E3, T1, and T3 interfaces, and to E1, E3, T1, and T3 partitions on the channelized interfaces (CE1, CT1, DS3, OC3, OC12, OC48, STM1, STM4, and STM16).</p> <p>E1 and T1 IQ, IQE, and standard interfaces support an extended BERT period range, up to 86,400 seconds (24 hours).</p>
	<div>  <p>NOTE: When configuring CE1 or CT1 interfaces on 10-port Channelized E1/T1 IQE PICs, the <code>bert-period</code> statement must be included at the <code>[edit interfaces <i>ce1-fpc/pic/port</i>]</code> or <code>[edit interfaces <i>ct1-fpc/pic/port</i>]</code> hierarchy level as appropriate.</p> </div>
Options	<p><i>seconds</i>—Test duration. Range and default values vary by interface type.</p> <p>Range:</p> <ul style="list-style-type: none"> PIC-dependent—Normal BERT period: either 1 through 239 seconds or 1 through 240 seconds PIC-dependent—Extended BERT period: from 1 through 86,400 seconds <p>Default:</p> <ul style="list-style-type: none"> Normal BERT period: 10 seconds Extended BERT period (on supported E1 interfaces): 10 seconds Extended BERT period (on supported T1 interfaces): 240 seconds
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> Interface Diagnostics Configuring E1 BERT Properties on page 162 Configuring E3 BERT Properties on page 170

- Configuring T1 BERT Properties on page 178
- Configuring T3 BERT Properties on page 188
- **bert-algorithm** on page 222
- **bert-error-rate** on page 224

buildout (T1 Interfaces)

Syntax	<code>buildout value;</code>
Hierarchy Level	<code>[edit interfaces ct1-fpc/pic/port]</code> <code>[edit interfaces interface-name t1-options]</code>
Release Information	Statement introduced before Junos OS Release 7.4.
Description	For T1 interfaces, set the buildout value.
	<div>NOTE: When configuring CT1 interfaces on 10-port Channelized E1/T1 IQE PICs, the buildout statement must be included at the hierarchy level.</div>
Default	The default buildout value is 0 through 132 feet.
Options	You can set the buildout value to one of the following: <ul style="list-style-type: none">• 0-132—0 through 132 feet (0 through 40 meters)• 133-265—133 through 265 feet (40 through 81 meters)• 266-398—266 through 398 feet (81 through 121 meters)• 399-531—399 through 531 feet (121 through 162 meters)• 532-655—532 through 655 feet (162 through 200 meters)• long-0db—For J Series routers only, long buildout with 0 decibel (dB) transmit attenuation• long-7.5db—For J Series routers only, long buildout with 7.5 dB transmit attenuation• long-15db—For J Series routers only, long buildout with 15 dB transmit attenuation• long-22.5db—For J Series routers only, long buildout with 22.5 dB transmit attenuation
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring the T1 Buildout on page 179• <i>Junos OS Interfaces and Routing Configuration Guide</i>

byte-encoding

Syntax	byte-encoding (nx56 nx64);
Hierarchy Level	[edit interfaces t1- <i>fpc/pic/port</i>], [edit interfaces <i>interface-name</i> ds0-options], [edit interfaces <i>interface-name</i> t1-options]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Set the byte encoding on a DS0 or T1 interface to use 7 bits per byte or 8 bits per byte.



NOTE: When configuring T1 interfaces on the 10-port Channelized E1/T1 IQE PIC, the `byte-encoding` statement must be included at the [edit interfaces t1-*fpc/pic/port*] hierarchy level.

Default	The default byte encoding is 8 bits per byte (nx64).
Options	nx56—Use 7 bits per byte. nx64—Use 8 bits per byte.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> Configuring T1 Byte Encoding on page 179

bytes

Syntax	<pre>bytes { c2 <i>value</i>; e1-quiet <i>value</i>; f1 <i>value</i>; f2 <i>value</i>; s1 <i>value</i>; z3 <i>value</i>; z4 <i>value</i>; }</pre>
Hierarchy Level	[edit interfaces <i>interface-name</i> sonet-options]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Set values in some SONET/SDH header bytes.
Options	<p>c2 <i>value</i>—Path signal label SONET/SDH overhead byte. SONET/SDH frames use the C2 byte to indicate the contents of the payload inside the frame. SONET/SDH interfaces use the C2 byte to indicate whether the payload is scrambled.</p> <p>Range: 0 through 255</p> <p>Default: 0xCF</p> <p>e1-quiet <i>value</i>—Default idle byte sent on the orderwire SONET/SDH overhead bytes. The router does not support the orderwire channel, and hence sends this byte continuously.</p> <p>Range: 0 through 255</p> <p>Default: 0x7F</p> <p>f1 <i>value</i>, f2 <i>value</i>, z3 <i>value</i>, z4 <i>value</i>—SONET/SDH overhead bytes.</p> <p>Range: 0 through 255</p> <p>Default: 0x00</p> <p>s1 <i>value</i>—Synchronization message SONET overhead byte. This byte is normally controlled as a side effect of the system reference clock configuration and the state of the external clock coming from an interface if the system reference clocks have been configured to use an external reference.</p> <p>Range: 0 through 255</p> <p>Default: 0xCC</p>
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">Configuring SONET/SDH Header Byte Valuesno-concatenate

cbit-parity

Syntax	(cbit-parity no-cbit-parity);
Hierarchy Level	[edit interfaces <i>interface-name</i> t3-options]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	For T3 interfaces only, enable or disable C-bit parity mode, which controls the type of framing that is present on the transmitted T3 signal. When C-bit parity mode is enabled, the C-bit positions are used for the far-end block error (FEBE), far-end alarm and control (FEAC), terminal data link, path parity, and mode indicator bits, as defined in ANSI T1.107a-1989. For ATM and ATM2 IQ2 and IQ2-E interfaces, M23 framing is used when the no-cbit-parity statement is included. For all other interfaces, M13 framing is used when the no-cbit-parity statement is included.
Default	C-bit parity mode is enabled.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring E3 and T3 Parameters on ATM Interfaces• Disabling T3 C-Bit Parity Mode on page 189

clocking

Syntax	clocking (external [interface <i>interface-name</i>] internal);
Hierarchy Level	[edit interfaces <i>interface-name</i>]
Release Information	Statement introduced before Junos OS Release 7.4. interface option added in Junos OS Release 8.2.
Description	For interfaces that can use various clock sources, configure the source of the transmit clock on each interface.
Options	<p>external—The clock source is provided by the data communication equipment (DCE).</p> <p>interface <i>interface-name</i>—For interfaces operating on T1/E1 PIMs for J Series Services Routers only, configure clocking for the drop-and insert feature. When configuring this feature, both ports must use the same clock source: either the router's internal clock or an external clock on one of the interfaces. If an external clock source is required, one interface must specify clocking external and the other must specify the same clock.</p> <p>internal—Use the internal stratum 3 clock as the reference clock.</p> <p>Default: internal</p>
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring the Clock Source• Configuring the Clock Source on SONET/SDH Interfaces• Clock Sources on Channelized Interfaces on page 32• Configuring a Channelized T1/E1 Interface to Drop and Insert Time Slots on page 152• loop-timing on page 265

compatibility-mode

Syntax	<code>compatibility-mode (adtran digital-link kentrox larscom verilink) <subrate value>;</code>
Hierarchy Level	[edit interfaces <i>interface-name</i> e3-options], [edit interfaces <i>interface-name</i> t3-options]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Configure the E3 or T3 interface to be compatible with the channel service unit (CSU) at the remote end of the line.



NOTE: The `compatibility-mode` statement at the [edit interfaces *interface-name* e3-options] hierarchy level is not valid for IQE PICs.

Default	If you omit this option, the full E3 or T3 rate is used.
Options	<p>adtran—For T3 IQ interfaces only, configure compatibility with Adtran CSUs.</p> <p>digital-link—Configure compatibility with Digital Link CSUs. If you include this option on an E3 interface, you must also disable payload scrambling.</p> <p>kentrox—Configure compatibility with Kentrox CSUs. Kentrox subrate is valid for E3 IQ and T3 IQ interfaces only.</p> <p>larscom—For T3 and T3 IQ interfaces only, configure compatibility with Larscom CSUs.</p> <p>verilink—For T3 IQ and T3 IQE interfaces only, configure compatibility with Verilink CSUs.</p>



NOTE: Verilink configuration is not functional if an IQ interface is paired with an IQE interface.

subrate value—Subrate of the E3 or T3 line.

Range: For Kentrox CSUs on E3 IQ interfaces and T3 IQ interfaces the subrate value must match the value configured on the CSU. Each increment of the subrate value corresponds to a rate increment of about 0.5 Mbps.

Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> Configuring the E3 CSU Compatibility Mode on page 171 Configuring the T3 CSU Compatibility Mode on page 190 <code>payload-scrambler</code> on page 277

crc-minor-alarm-threshold

Syntax	crc-minor-alarm-threshold (1e-3 5e-4 1e-4 5e-5 1e-5 5e-6 1e-6);
Hierarchy Level	[edit interfaces <i>interface-name</i> t1-options]
Release Information	Statement introduced in Junos OS Release 8.5.
Description	Minor alarm error thresholds for T1 CRC errors. When the threshold is exceeded for one second, a defect condition is declared. If the defect condition continues for the monitoring period, an alarm condition is declared.
Default	10-second monitoring period for values 1e-3, 5e-4, 1e-4, and 5e-5. The 1e-5 value uses a 50-second monitoring period. The 5e-6 value uses a 100-second monitoring period. The 1e-6 value uses a 500-second monitoring period.
Options	rate —Error rate expressed as the number of errors per number of bits. The value 1e-3 is one error in 10 ⁻³ bits and 5e-4 is five errors in 10 ⁻⁴ bits. Default: 5e-6
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">Configuring T1 CRC Error Minor Alarm Thresholds on page 180

data-input

Syntax	<code>data-input (system interface <i>interface-name</i>);</code>
Hierarchy Level	<code>[edit interfaces <i>ds-pim</i>/0/<i>port:channel</i>]</code>
Release Information	Statement introduced in Junos OS Release 8.2.
Description	<p>For interfaces operating on T1/E1 PIMs for J Series Services Routers only, configure whether an interface should send and receive data from the Routing Engine or from a given interface name. On channelized T1/E1 interfaces partitioned into channels, you can insert time slots from one port directly into the other port on the same PIM, to replace time slots coming through the Routing Engine.</p> <p>To avoid slips, both ports must use the same clock source: either the router's internal clock or an external clock on one of the interfaces. If an external clock source is required, one interface must specify <code>clocking external</code> and the other must specify the same clock by including the <code>clocking external interface <i>interface-name</i></code> statement at the <code>[edit interfaces <i>interface-name</i>]</code> hierarchy level.</p>
Options	<p>system—Interface sends and receives data from the Routing Engine.</p> <p>interface <i>interface-name</i>—Interface sends and receives data from a specific interface.</p> <p>Default: Data is sent and received from the Routing Engine (system).</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> Configuring a Channelized T1/E1 Interface to Drop and Insert Time Slots on page 152 <i>Junos OS Interfaces and Routing Configuration Guide</i> <code>clocking</code> on page 230

dce

Syntax	dce;
Hierarchy Level	[edit interfaces <i>interface-name</i>], [edit interfaces <i>interface-name</i> serial-options clocking-mode]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	For Frame Relay only, respond to status enquiry message keepalives. When you configure the router to be a DCE, keepalives are disabled by default.
Default	The router operates in DTE mode.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring the Router as a DCE with Frame Relay Encapsulation on page 212

dlci

Syntax	<code>dlci <i>dlci-identifier</i>;</code>
Hierarchy Level	[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i>], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i>]
Release Information	Statement introduced before Junos OS Release 7.4. Statement introduced in Junos OS Release 11.1 for the QFX Series.
Description	<p>For Frame Relay and Multilink Frame Relay (MLFR) user-to-network interface (UNI) network-to-network interface (NNI) encapsulation only, and for link services, voice services and point-to-point interfaces only, configure the data-link connection identifier (DLCI) for a permanent virtual circuit (PVC) or an switched virtual circuit (SVC).</p> <p>To configure a DLCI for a point-to-multipoint interface, use the multipoint-destination statement to specify the DLCI.</p>
Options	<p><i>dlci-identifier</i>—Data-link connection identifier.</p> <p>Range: 16 through 1022.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • Data-Link Connection Identifiers on Channelized Interfaces on page 30 • Configuring Frame Relay DLCIs on page 212 • Junos OS Services Interfaces Configuration Guide • encapsulation (Logical Interface) on page 240 • multicast-dlci on page 269 • multipoint-destination on page 270

ds0-options

Syntax ds0-options {
 bert-algorithm *algorithm*;
 bert-error-rate *rate*;
 bert-period *seconds*;
 byte-encoding (nx56 | nx64);
 fcs (16 | 32);
 idle-cycle-flag (flags | ones);
 invert-data;
 loopback payload;
 start-end-flag (filler | shared);
 }

Hierarchy Level [edit interfaces *interface-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure DS0-specific physical interface properties.

 The statements are explained separately.

Required Privilege Level interface—To view this statement in the configuration.
 interface-control—To add this statement to the configuration.

Related Documentation • Configuring Channelized DS3-to-DS0 Interfaces on page 126

e1-options

Syntax	<pre>e1-options { bert-algorithm <i>algorithm</i>; bert-error-rate <i>rate</i>; bert-period <i>seconds</i>; fcs (16 32); framing (g704 g704-no-crc4 unframed); idle-cycle-flag (flags ones); invert-data; loopback (local remote); start-end-flag (filler shared); timeslots <i>time-slot-range</i>; }</pre>
Hierarchy Level	[edit interfaces <i>interface-name</i>]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	<p>Configure E1-specific physical interface properties.</p> <p>The statements are explained separately.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • Channelized E1 IQ and IQE Interfaces Overview on page 143 • Channelized STM1 Interfaces Overview on page 105 • E1 Interfaces Overview on page 161 • T1 Interfaces Overview on page 177

e3-options

Syntax e3-options {
 atm-encapsulation (direct | plcp);
 bert-algorithm *algorithm*;
 bert-error-rate *rate*;
 bert-period *seconds*;
 buildout *feet*;
 compatibility-mode (digital-link | kentrox | larscom) <subrate *value*>;
 fcs (16 | 32);
 framing (g.751 | g.832);
 idle-cycle-flag *value*;
 invert-data;
 loopback (local | remote);
 (payload-scrambler | no-payload-scrambler);
 start-end-flag *value*;
 (unframed | no-unframed);
 }

Hierarchy Level [edit interfaces *interface-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure E3-specific physical interface properties.

For ATM1 interfaces, you can configure a subset of E3 options statements.

The statements are explained separately.

Required Privilege Level interface—To view this statement in the configuration.
 interface-control—To add this statement to the configuration.

Related Documentation

- E3 Interfaces Overview on page 169
- T3 Interfaces Overview on page 187
- atm-options

encapsulation

See the following sections:

- **encapsulation (Logical Interface) on page 240**
- **encapsulation (Physical Interface) on page 243**

encapsulation (Logical Interface)

Syntax	encapsulation (atm-ccc-cell-relay atm-ccc-vc-mux atm-cisco-nlpid atm-mlppp-llc atm-nlpid atm-ppp-llc atm-ppp-vc-mux atm-snap atm-tcc-snap atm-tcc-vc-mux atm-vc-mux ether-over-atm-llc ether-vpls-over-atm-llc ether-vpls-over-fr ether-vpls-over-ppp ethernet frame-relay-ccc frame-relay-ether-type frame-relay-ether-type-tcc frame-relay-ppp frame-relay-tcc multilink-frame-relay-end-to-end multilink-ppp ppp-over-ether ppp-over-ether-over-atm-llc vlan-bridge vlan-ccc vlan-vci-ccc vlan-tcc vlan-vpls);
Hierarchy Level	[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i>], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i>] [edit interfaces <i>rlsnumber</i> unit <i>logical-unit-number</i>]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Logical link-layer encapsulation type.
Options	<p>atm-ccc-cell-relay—Use ATM cell-relay encapsulation.</p> <p>atm-ccc-vc-mux—Use ATM virtual circuit (VC) multiplex encapsulation on CCC circuits. When you use this encapsulation type, you can configure the ccc family only.</p> <p>atm-cisco-nlpid—Use Cisco ATM network layer protocol ID (NLPID) encapsulation. When you use this encapsulation type, you can configure the inet family only.</p> <p>atm-mlppp-llc—For ATM2 IQ interfaces only, use Multilink PPP (MLPPP) over AAL5 LLC. For this encapsulation type, your router must be equipped with a Link Services or Voice Services PIC. MLPPP over ATM encapsulation is not supported on ATM2 IQ OC48 interfaces.</p> <p>atm-nlpid—Use ATM NLPID encapsulation. When you use this encapsulation type, you can configure the inet family only.</p> <p>atm-ppp-llc—For ATM2 IQ interfaces only, use PPP over AAL5 LLC encapsulation.</p> <p>atm-ppp-vc-mux—For ATM2 IQ interfaces only, use PPP over ATM AAL5 multiplex encapsulation.</p> <p>atm-snap—Use ATM subnetwork attachment point (SNAP) encapsulation.</p> <p>atm-tcc-snap—Use ATM SNAP encapsulation on translational cross-connect (TCC) circuits.</p> <p>atm-tcc-vc-mux—Use ATM VC multiplex encapsulation on TCC circuits. When you use this encapsulation type, you can configure the tcc family only.</p> <p>atm-vc-mux—Use ATM VC multiplex encapsulation. When you use this encapsulation type, you can configure the inet family only.</p> <p>ether-over-atm-llc—For interfaces that carry IPv4 traffic, use Ethernet over ATM LLC encapsulation. When you use this encapsulation type, you cannot configure multipoint interfaces.</p>

ether-vpls-over-atm-llc—For ATM2 IQ interfaces only, use the Ethernet virtual private LAN service (VPLS) over ATM LLC encapsulation to bridge Ethernet interfaces and ATM interfaces over a VPLS routing instance (as described in RFC 2684, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*). Packets from the ATM interfaces are converted to standard ENET2/802.3 encapsulated Ethernet frames with the frame check sequence (FCS) field removed.

ether-vpls-over-fr—For E1, T1, E3, T3, and SONET interfaces only, use the Ethernet virtual private LAN service (VPLS) over Frame Relay encapsulation to support Bridged Ethernet over Frame Relay encapsulated TDM interfaces for VPLS applications, as per RFC 2427 (1490).

ether-vpls-over-ppp—For E1, T1, E3, T3 and SONET interfaces only, use the Ethernet virtual private LAN service (VPLS) over point-to-point-protocol (PPP) encapsulation to support Bridged Ethernet over PPP encapsulated TDM interfaces for VPLS applications.

ethernet—Use Ethernet II encapsulation (as described in RFC 894, *A Standard for the Transmission of IP Datagrams over Ethernet Networks*).

ethernet-vpls—Use Ethernet VPLS encapsulation on Ethernet interfaces that have VPLS enabled and that must accept packets carrying standard Tag Protocol ID (TPID) values.



NOTE: The built-in Gigabit Ethernet PIC on an M7i router does not support extended VLAN VPLS encapsulation.

frame-relay-ccc—Use Frame Relay encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only.

frame-relay-ppp—Use PPP over Frame Relay circuits. When you use this encapsulation type, you can configure the **ppp** family only. J Series Routers do not support frame-relay-ppp encapsulation.

frame-relay-tcc—Use Frame Relay encapsulation on TCC circuits for connecting unlike media. When you use this encapsulation type, you can configure the **tcc** family only.

frame-relay-ether-type—Use Frame Relay ether type encapsulation for compatibility with Cisco Frame Relay. The physical interface must be configured with flexible-frame-relay encapsulation.

frame-relay-ether-type-tcc—Use Frame Relay ether type TCC for Cisco-compatible Frame Relay on TCC circuits to connect unlike media. The physical interface must be configured with flexible-frame-relay encapsulation.

multilink-frame-relay-end-to-end—Use MLFR FRF.15 encapsulation. This encapsulation is used only on multilink, link services, voice services interfaces and their constituent T1 or E1 interfaces, and is supported on LSQ and redundant LSQ interfaces.

multilink-ppp—Use MLPPP encapsulation. This encapsulation is used only on multilink, link services, and voice services interfaces and their constituent T1 or E1 interfaces.

ppp-over-ether—For underlying Ethernet interfaces on J Series Services Routers, use PPP over Ethernet encapsulation. When you use this encapsulation type, you cannot configure the interface address. Instead, configure the interface address on the PPP interface. You also use PPP over Ethernet encapsulation to configure an underlying Ethernet interface for a dynamic PPPoE logical interface on M120 and M320 Series routers with Intelligent Queuing 2 (IQ2) PICs, and on MX Series routers with Trio MPC/MIC interfaces.

ppp-over-ether-over-atm-llc—For underlying ATM interfaces on J Series Services Routers only, use PPP over Ethernet over ATM LLC encapsulation. When you use this encapsulation type, you cannot configure the interface address. Instead, configure the interface address on the PPP interface.

vlan-bridge—Use Ethernet VLAN bridge encapsulation on Ethernet interfaces that have IEEE 802.1Q tagging, flexible-ethernet-services, and bridging enabled and that must accept packets carrying TPID 0x8100 or a user-defined TPID.

vlan-ccc—Use Ethernet virtual LAN (VLAN) encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only.

vlan-vci-ccc—Use ATM-to-Ethernet interworking encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only.

vlan-tcc—Use Ethernet VLAN encapsulation on TCC circuits. When you use this encapsulation type, you can configure the **tcc** family only.

vlan-vpls—Use Ethernet VLAN encapsulation on VPLS circuits.

Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
---------------------------------	---

Related Documentation	<ul style="list-style-type: none">• Configuring Interface Encapsulation on Logical Interfaces• Circuit and Translational Cross-Connects Overview• Identifying the Access Concentrator• Configuring ATM Interface Encapsulation• Configuring VLAN Encapsulation• Configuring Extended VLAN Encapsulation• Configuring ISDN Logical Interface Properties• Configuring ATM-to-Ethernet Interworking• Junos OS Services Interfaces Configuration Guide
------------------------------	--

encapsulation (Physical Interface)

Syntax	encapsulation (atm-ccc-cell-relay atm-pvc cisco-hdlc cisco-hdlc-ccc cisco-hdlc-tcc ethernet-bridge ethernet-ccc ethernet-over-atm ethernet-tcc ethernet-vpls extended-frame-relay-ccc extended-frame-relay-ether-type-tcc extended-frame-relay-tcc extended-vlan-bridge extended-vlan-ccc extended-vlan-tcc extended-vlan-vpls flexible-ethernet-services flexible-frame-relay frame-relay frame-relay-ccc frame-relay-ether-type frame-relay-ether-type-tcc frame-relay-port-ccc frame-relay-tcc multilink-frame-relay-uni-nni ppp ppp-ccc ppp-tcc vlan-ccc vlan-vci-ccc vlan-vpls);
Hierarchy Level	[edit interfaces <i>interface-name</i>], [edit interfaces <i>rlsnumber:number</i>]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Physical link-layer encapsulation type.
Default	PPP encapsulation.
Options	<p>atm-ccc-cell-relay—Use ATM cell-relay encapsulation.</p> <p>atm-pvc—Use ATM PVC encapsulation.</p> <p>cisco-hdlc—Use Cisco-compatible High-Level Data Link Control (HDLC) framing.</p> <p>cisco-hdlc-ccc—Use Cisco-compatible HDLC framing on CCC circuits.</p> <p>cisco-hdlc-tcc—Use Cisco-compatible HDLC framing on TCC circuits for connecting unlike media.</p> <p>ethernet-bridge—Use Ethernet bridge encapsulation on Ethernet interfaces that have bridging enabled and that must accept all packets.</p> <p>ethernet-ccc—Use Ethernet CCC encapsulation on Ethernet interfaces that must accept packets carrying standard Tag Protocol ID (TPID) values. For 8-port, 12-port, and 48-port Fast Ethernet PICs, CCC is not supported</p> <p>ethernet-over-atm—For interfaces that carry IPv4 traffic, use Ethernet over ATM encapsulation. When you use this encapsulation type, you cannot configure multipoint interfaces. As defined in RFC 1483, <i>Multiprotocol Encapsulation over ATM Adaptation Layer 5</i>, this encapsulation type allows ATM interfaces to connect to devices that support only bridged protocol data units (BPDUs). The Junos OS does not completely support bridging, but accepts BPDU packets as a default gateway. If you use the router as an edge device, then the router acts as a default gateway. It accepts Ethernet LLC/SNAP frames with IP or ARP in the payload, and drops the rest. For packets destined to the Ethernet LAN, a route lookup is done using the destination IP address. If the route lookup yields a full address match, the packet is encapsulated with an LLC/SNAP and MAC header, and the packet is forwarded to the ATM interface.</p>

ethernet-tcc—For interfaces that carry IPv4 traffic, use Ethernet TCC encapsulation on interfaces that must accept packets carrying standard TPID values. For 8-port, 12-port, and 48-port Fast Ethernet PICs, TCC is not supported.

ethernet-vpls—Use Ethernet VPLS encapsulation on Ethernet interfaces that have VPLS enabled and that must accept packets carrying standard TPID values.

extended-frame-relay-ccc—Use Frame Relay encapsulation on CCC circuits. This encapsulation type allows you to dedicate DLCIs 1 through 1022 to CCC.

extended-frame-relay-tcc—Use Frame Relay encapsulation on TCC circuits to connect unlike media. This encapsulation type allows you to dedicate DLCIs 1 through 1022 to TCC.

extended-vlan-bridge—Use extended VLAN bridge encapsulation on Ethernet interfaces that have IEEE 802.1Q VLAN tagging and bridging enabled and that must accept packets carrying TPID 0x8100 or a user-defined TPID.

extended-vlan-ccc—Use extended VLAN encapsulation on CCC circuits with Gigabit Ethernet and 4-port Fast Ethernet interfaces that must accept packets carrying 802.1Q values. For 8-port, 12-port, and 48-port Fast Ethernet PICs, extended VLAN CCC is not supported. For 4-port Gigabit Ethernet PICs, extended VLAN CCC is not supported.

extended-vlan-tcc—For interfaces that carry IPv4 traffic, use extended VLAN encapsulation on TCC circuits with Gigabit Ethernet interfaces on which you want to use 802.1Q tagging. For 4-port Gigabit Ethernet PICs, extended VLAN TCC is not supported.

extended-vlan-vpls—Use extended VLAN VPLS encapsulation on Ethernet interfaces that have VLAN 802.1Q tagging and VPLS enabled and that must accept packets carrying TPIDs 0x8100, 0x9100, and 0x9901.



NOTE: The built-in Gigabit Ethernet PIC on an M7i router does not support extended VLAN VPLS encapsulation.

flexible-ethernet-services—For Gigabit Ethernet IQ interfaces and Gigabit Ethernet PICs with small form-factor pluggable transceivers (SFPs) (except the 10-port Gigabit Ethernet PIC and the built-in Gigabit Ethernet port on the M7i router), use flexible Ethernet services encapsulation when you want to configure multiple per-unit Ethernet encapsulations. Aggregated Ethernet bundles can use this encapsulation type. This encapsulation type allows you to configure any combination of route, TCC, CCC, Layer 2 virtual private networks (VPNs), and VPLS encapsulations on a single physical port. If you configure flexible Ethernet services encapsulation on the physical interface, VLAN IDs from 1 through 511 are no longer reserved for normal VLANs.

flexible-frame-relay—For IQ interfaces only, use flexible Frame Relay encapsulation when you want to configure multiple per-unit Frame Relay encapsulations. This encapsulation type allows you to configure any combination of TCC, CCC, and standard Frame Relay encapsulations on a single physical port. Also, each logical interface can have any DLCI value from 1 through 1022.

frame-relay—Use Frame Relay encapsulation.

frame-relay-ccc—Use Frame Relay encapsulation on CCC circuits.

frame-relay-port-ccc—Use Frame Relay port CCC encapsulation to transparently carry all the DLCIs between two customer edge (CE) routers without explicitly configuring each DLCI on the two provider edge (PE) routers with Frame Relay transport. When you use this encapsulation type, you can configure the **ccc** family only.

frame-relay-tcc—Use Frame Relay encapsulation on TCC circuits to connect unlike media.

frame-relay-ether-type—Use Frame Relay ether type encapsulation for compatibility with Cisco Frame Relay.

frame-relay-ether-type-tcc—Use Frame Relay ether type TCC for Cisco-compatible Frame Relay on TCC circuits to connect unlike media.

extended-frame-relay-ether-type-tcc—Use extended Frame Relay ether type TCC for Cisco-compatible Frame Relay for DLCIs 1 through 1022. This encapsulation is used for circuits with different media on either side of the connection.

multilink-frame-relay-uni-nni—Use MLFR UNI NNI encapsulation. This encapsulation is used on link services, voice services interfaces functioning as FRF.16 bundles and their constituent T1 or E1 interfaces, and is supported on LSQ and redundant LSQ interfaces.

ppp—Use serial PPP encapsulation.

ppp-ccc—Use serial PPP encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only.

ppp-tcc—Use serial PPP encapsulation on TCC circuits for connecting unlike media. When you use this encapsulation type, you can configure the **tcc** family only.

vlan-ccc—Use Ethernet VLAN encapsulation on CCC circuits.

vlan-vci-ccc—Use ATM-to-Ethernet interworking encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only. All logical interfaces configured on the Ethernet interface must also have the encapsulation type set to **vlan-vci-ccc**.

vlan-vpls—Use VLAN VPLS encapsulation on Ethernet interfaces with VLAN tagging and VPLS enabled. Interfaces with VLAN VPLS encapsulation accept packets carrying standard TPID values only.

Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Interface Encapsulation on Physical Interfaces• Defining the Encapsulation for Switching Cross-Connects• Configuring ATM Interface Encapsulation• Configuring VLAN Encapsulation• Configuring ATM-to-Ethernet Interworking• Configuring Extended VLAN Encapsulation• Configuring Encapsulation for Layer 2 Wholesale VLAN Interfaces

family

```

Syntax  family family {
            accounting {
                destination-class-usage;
                source-class-usage {
                    (input | output | input output);
                }
            }
            access-concentrator name;
            address address {
                ... the address subhierarchy appears after the main [edit interfaces interface-name unit
                    logical-unit-number family family-name] hierarchy ...
            }
            bridge-domain-type (bvlan | svlan);
            bundle interface-name;
            core-facing;
            demux-destination {
                destination-prefix;
            }
            demux-source {
                source-prefix;
            }
            duplicate-protection;
            dynamic-profile profile-name;
            filter {
                group filter-group-number;
                input filter-name;
                input-list [ filter-names ];
                output filter-name;
                output-list [ filter-names ];
            }
            interface-mode (access | trunk);
            ipsec-sa sa-name;
            isid-list all-service-groups;
            keep-address-and-control;
            mac-validate (loose | strict);
            max-sessions number;
            mtu bytes;
            multicast-only;
            negotiate-address;
            no-redirects;
            policer {
                arp policer-template-name;
                input policer-template-name;
                output policer-template-name;
            }
            primary;
            protocols [inet iso mpls];
            proxy inet-address address;
            receive-options-packets;
            receive-ttl-exceeded;
            remote (inet-address address | mac-address address);
            rpf-check {

```

```
fail-filter filter-name
mode loose;
}
sampling {
input;
output;
}
service {
input {
post-service-filter filter-name;
service-set service-set-name <service-filter filter-name>;
}
output {
service-set service-set-name <service-filter filter-name>;
}
}
service-name-table table-name
(translate-discard-eligible | no-translate-discard-eligible);
(translate-fecn-and-becn | no-translate-fecn-and-becn);
unnumbered-address interface-name destination address destination-profile profile-name;
vlan-id number;
vlan-id-list [number number-number];
address address {
arp ip-address (mac | multicast-mac) mac-address <publish>;
broadcast address;
destination address;
destination-profile name;
eui-64;
master-only;
multipoint-destination address dlci dlci-identifier;
multipoint-destination address {
epd-threshold cells;
inverse-arp;
oam-liveness {
up-count cells;
down-count cells;
}
oam-period (disable | seconds);
shaping {
(cbr rate | rtvbr burst length peak rate sustained rate | vbr burst length peak rate
sustained rate);
queue-length number;
}
vci vpi-identifier.vci-identifier;
}
preferred;
primary;
(vrrp-group | vrrp-inet6-group) group-number {
(accept-data | no-accept-data);
advertise-interval seconds;
authentication-type authentication;
authentication-key key;
fast-interval milliseconds;
(preempt | no-preempt) {
hold-time seconds;
}
}
```

```

priority number;
track {
    interface interface-name {
        bandwidth-threshold bits-per-second priority-cost number;
    }
    priority-hold-time seconds;
    route ip-address/prefix-length routing-instance instance-name priority-cost cost;
}
virtual-address [ addresses ];
virtual-link-local-address ipv6-address;
vrrp-inherit-from {
    active-interface interface-name;
    active-group group-number;
}
}
}

```

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number*],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure protocol family information for the logical interface.



NOTE: Not all subordinate stanzas are available to every protocol family. See the *Junos OS Configuration Statements and Commands* for details about each protocol family.

Options *family*—Protocol family:

- **any**—Protocol-independent family used for Layer 2 packet filtering
- **bridge**—(M Series and T Series routers only) Configure only when the physical interface is configured with **ethernet-bridge** type encapsulation or when the logical interface is configured with **vlan-bridge** type encapsulation
- **ccc**—Circuit cross-connect protocol suite
- **inet**—Internet Protocol version 4 suite
- **inet6**—Internet Protocol version 6 suite
- **iso**—International Organization for Standardization Open Systems Interconnection (ISO OSI) protocol suite
- **mlfr-end-to-end**—Multilink Frame Relay FRF.15
- **mlfr-uni-nni**—Multilink Frame Relay FRF.16
- **multilink-ppp**—Multilink Point-to-Point Protocol
- **mpls**—Multiprotocol Label Switching (MPLS)
- **pppoe**—Point-to-Point Protocol over Ethernet
- **tcc**—Translational cross-connect protocol suite
- **tnp**—Trivial Network Protocol
- **vpls**—(M Series and T Series routers only) Virtual private LAN service


The remaining statements are explained separately.

Required Privilege Level interface—To view this statement in the configuration.
 interface-control—To add this statement to the configuration.

Related Documentation

- Configuring the Protocol Family
- Example: Configuring E-LINE and E-LAN Services for a PBB Network on MX Series Routers
- [Junos OS Services Interfaces Configuration Guide](#)

fcs

Syntax	fcs (16 32);
Hierarchy Level	[edit interfaces e1- <i>fpc/pic/port</i>], [edit interfaces t1- <i>fpc/pic/port</i>], [edit interfaces <i>interface-name</i> ds0-options], [edit interfaces <i>interface-name</i> e1-options], [edit interfaces <i>interface-name</i> e3-options], [edit interfaces <i>interface-name</i> sonet-options], [edit interfaces <i>interface-name</i> t1-options], [edit interfaces <i>interface-name</i> t3-options]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	<p>For E1/E3, SONET/SDH, and T1/T3 interfaces, configure the frame checksum (FCS) on the interface. The checksum must be the same on both ends of the interface.</p> <p>On a channelized OC12 interface, the SONET/SDH fcs statement is not supported. To configure FCS on each DS3 channel, you must include the t3-options fcs statement in the configuration for each channel. For SONET/SDH, the channelized OC12 interface supports DS3 to STS-1 to OC12. For SDH, the channelized OC12 interface supports NxDS3 to NxVC3 to AU3 to STM.</p>
	<div>  <p>NOTE: When configuring E1 or T1 interfaces on 10-port Channelized E1/T1 IQE PICs, the fcs statement must be included at the [edit interfaces e1-<i>fpc/pic/port</i>] or [edit interfaces t1-<i>fpc/pic/port</i>] hierarchy level as appropriate.</p> </div>
Options	<p>16—Use a 16-bit frame checksum on the interface.</p> <p>32—Use a 32-bit frame checksum on the interface. Using a 32-bit checksum provides more reliable packet verification, but some older equipment might not support 32-bit checksums.</p> <p>Default: 16</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> Configuring the E1 Frame Checksum on page 163 Configuring the E3 Frame Checksum on page 172 Configuring the SONET/SDH Frame Checksum Configuring the T1 Frame Checksum on page 181 Configuring the T3 Frame Checksum on page 192


feac-loop-respond

Syntax	(feac-loop-respond no-feac-loop-respond);
Hierarchy Level	[edit interfaces <i>interface-name</i> t3-options]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	<p>For T3 interfaces only, configure the router so a remote CSU can place the local router into loopback.</p> <p>If you configure remote or local loopback with the T3 loopback statement, the router does not respond to FEAC requests from the CSU even if you include the feac-loop-respond statement in the configuration. For the router to respond, you must delete the loopback statement from the configuration.</p> <p>You must rollback the setting done on the remote CSU prior to deactivating the feac-loop-respond statement. If the remote CSU cannot comply, clear the remote loop through local configuration to achieve the cleanup. For example, configure remote loopback on the interface and then delete the remote loopback.</p>
Default	The router does not respond to FEAC requests.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring the T3 FEAC Response on page 193• loopback (ADSL, DS0, E1/E3, SONET/SDH, SHDSL, and T1/T3) on page 266• remote-loopback-respond on page 278

force

Syntax	<code>force (protect working);</code>
Hierarchy Level	[edit interfaces <i>interface-name</i> sonet-options aps]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Perform a forced switch between the protect and working circuits. This statement is honored only if there are no higher-priority reasons to switch. It can be overridden by a signal failure on the protect circuit, thus causing a switch to the working circuit.
Options	protect —Request the circuit to become the protect circuit. working —Request the circuit to become the working circuit.
Required Privilege Level	interface —To view this statement in the configuration. interface-control —To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">Configuring Switching Between the Working and Protect Circuitsrequest on page 279

framing (E1, E3, and T1 Interfaces)

Syntax	framing (g704 g704-no-crc4 g.751 g.832 unframed sf esf);
Hierarchy Level	[edit interfaces ce1- <i>fpc/pic/port</i>], [edit interfaces ct1- <i>fpc/pic/port</i>], [edit interfaces at- <i>fpc/pic/port</i> e3-options], [edit interfaces e1- <i>fpc/pic/port</i> e1-options], [edit interfaces t1- <i>fpc/pic/port</i> t1-options]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Configure the framing format.
	<div>  <p>NOTE: When configuring CE1 or CT1 interfaces on 10-port Channelized E1/T1 IQE PICs, the framing statement must be included at the [edit interfaces ce1-<i>fpc/pic/port</i>] or [edit interfaces ct1-<i>fpc/pic/port</i>] hierarchy level as appropriate.</p> </div>
Default	esf for T1 interfaces; g704 for E1 interfaces. There is no default value for E3 over ATM interfaces.
Options	<p>esf—Extended superframe (ESF) mode for T1 interfaces.</p> <p>g704—G.704 framing format for E1 interfaces.</p> <p>g704-no-crc4—G.704 framing with no cyclic redundancy check 4 (CRC4) for E1 interfaces.</p> <p>g.751—G.751 framing format for E3 over ATM interfaces.</p> <p>g.832—G.832 framing format for E3 over ATM interfaces.</p> <p>sf—Superframe (SF) mode for T1 interfaces.</p> <p>unframed—Unframed mode for E1 interfaces.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> Configuring E1 Framing on page 163 Configuring E3 and T3 Parameters on ATM Interfaces Configuring T1 Framing on page 182

hold-time (APS)

Syntax	hold-time <i>milliseconds</i> ;
Hierarchy Level	[edit interfaces <i>interface-name</i> sonet-options aps]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Hold-time value to use to determine whether a neighbor APS router is operational.
Options	<i>milliseconds</i> —Hold-time value. Range: 1 through 65,534 milliseconds Default: 3000 milliseconds (3 times the advertisement interval)
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">Configuring APS Timersadvertise-interval on page 219

idle-cycle-flag

Syntax	<code>idle-cycle-flag <i>value</i>;</code>
Hierarchy Level	<code>[edit interfaces <i>e1-fpc/pic/port</i>],</code> <code>[edit interfaces <i>t1-fpc/pic/port</i>],</code> <code>[edit interfaces <i>interface-name</i> <i>ds0-options</i>],</code> <code>[edit interfaces <i>interface-name</i> <i>e1-options</i>],</code> <code>[edit interfaces <i>interface-name</i> <i>e3-options</i>],</code> <code>[edit interfaces <i>interface-name</i> <i>serial-options</i>],</code> <code>[edit interfaces <i>interface-name</i> <i>t1-options</i>],</code> <code>[edit interfaces <i>interface-name</i> <i>t3-options</i>]</code>
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Configure the value that the DS0, E1, E3, T1, or T3 interface transmits during idle cycles.



NOTE: When configuring E1 or T1 interfaces on 10-port Channelized E1/T1 IQE PICs, the `idle-cycle-flag` statement must be included at the `[edit interfaces e1-fpc/pic/port]` or `[edit interfaces t1-fpc/pic/port]` hierarchy level as appropriate.

Options	<i>value</i> —Value to transmit in the idle cycles: <ul style="list-style-type: none">• flags—Transmit the value 0x7E.• ones—Transmit the value 0xFF (all ones).
	Default: <code>Flags</code>
Required Privilege Level	<code>interface</code> —To view this statement in the configuration. <code>interface-control</code> —To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring the E1 Idle Cycle Flag on page 164• Configuring the E3 Idle Cycle Flag on page 173• Configuring the T1 Idle Cycle Flag on page 184• Configuring the T3 Idle Cycle Flag on page 193

interface-type

Syntax	<code>interface-type (bc coc1 ct1 ct3 dc ds so t1 t3);</code>
Hierarchy Level	<code>[edit interfaces <i>interface-name</i> no-partition],</code> <code>[edit interfaces <i>interface-name</i> partition <i>partition-number</i>],</code> <code>[edit interfaces <i>interface-name</i> partition <i>partition-number</i> oc-slice <i>oc-slice-range</i>],</code> <code>[edit interfaces <i>interface-name</i> partition <i>partition-number</i> timeslot <i>timeslot-range</i>]</code>
Release Information	Statement introduced before Junos OS Release 7.4.
Description	For IQ and IQE interfaces only, configure the sublevel interface type.
Options	<p>bc—Dual—Port Channelized E1 and T1 ISDN PRI interface type. You can specify this interface type at the <code>[edit interfaces <i>interface-name</i> partition <i>partition-number</i> timeslot <i>timeslot-range</i>]</code> hierarchy level to create a bearer (B) channel <code>bc-pim/0/port:channel</code> interface for each time you want to function as an ISDN PRI B-channel.</p> <p>coc1—Channelized OC1 interface type. You can specify this interface type at the <code>[edit interfaces <i>interface-name</i> partition <i>partition-number</i> oc-slice <i>oc-slice-range</i> interface-type coc12-fpc/pic/port]</code> hierarchy level.</p> <p>ct1—Channelized T1 interface type. You can specify this interface type at the <code>[edit interfaces <i>interface-name</i> partition <i>partition-number</i> interface-type ct3-fpc/pic/port<:channel>]</code> hierarchy level.</p> <p>ct3—Channelized T3 interface type. You can specify this interface type at the <code>[edit interfaces <i>interface-name</i> partition <i>partition-number</i> oc-slice <i>oc-slice-range</i> interface-type coc1-fpc/pic/port:channel no-partition]</code> hierarchy level.</p> <p>dc—Dual-Port Channelized E1 and T1 ISDN PRI interface type. You can specify this interface type at the <code>[edit interfaces <i>interface-name</i> partition <i>partition-number</i> timeslot <i>timeslot-range</i>]</code> hierarchy level to create a (D) channel <code>dc-pim/0/port</code> to control the B-channels.</p> <p>ds—DS0 interface type. You can specify this interface type at the <code>[edit interfaces <i>interface-name</i> partition <i>partition-number</i> interface-type (ce1-fpc/pic/port ct1-fpc/pic/port<:channel>)]</code> hierarchy level.</p> <p>so—SONET/SDH interface type. You can specify this interface type at the <code>[edit interfaces <i>interface-name</i> partition <i>partition-number</i> oc-slice <i>oc-slice-range</i> interface-type coc12-fpc/pic/port]</code> hierarchy level.</p> <p>t1—T1 interface type. You can specify this interface type at the <code>[edit interfaces <i>interface-name</i> partition <i>partition-number</i> oc-slice <i>oc-slice-range</i> interface-type (coc12-fpc/pic/port coc1-fpc/pic/port)]</code> hierarchy level.</p> <p>t3—T3 interface type. You can specify this interface type at the <code>[edit interfaces <i>interface-name</i> partition <i>partition-number</i> oc-slice <i>oc-slice-range</i> interface-type (coc12-fpc/pic/port coc1-fpc/pic/port:channel no-partition)]</code> hierarchy level.</p>

Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Channelized E1 IQ and IQE Interfaces Overview on page 143• Channelized OC12/STM4 IQ and IQE Interfaces Overview on page 65• Configuring Channelized T3 IQ Interfaces on page 123

inverse-arp

Syntax	inverse-arp;
Hierarchy Level	[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i>], [edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> family inet address <i>address</i> multipoint-destination <i>destination</i>], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i>], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i> family inet address <i>address</i> multipoint-destination <i>destination</i>]
Release Information	Statement introduced before Junos OS Release 7.4. Statement introduced in Junos OS Release 11.1 for the QFX Series.
Description	For ATM encapsulation, enable responses to receive inverse ATM ARP requests. For Frame Relay encapsulation, enable responses to receive inverse Frame Relay ARP requests.
Default	Inverse ARP is disabled on all ATM and Frame Relay interfaces.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Inverse ATM1 or ATM2 ARP• Configuring Inverse Frame Relay ARP on page 211

invert-data

Syntax	invert-data;
Hierarchy Level	[edit interfaces e1- <i>fpc/pic/port</i>], [edit interfaces t1- <i>fpc/pic/port</i>], [edit interfaces <i>interface-name</i> ds0-options], [edit interfaces <i>interface-name</i> e1-options], [edit interfaces <i>interface-name</i> t1-options], [edit interfaces <i>interface-name</i> e3-options]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Invert the transmission of unused data bits on the DS0, E1, E3, and T1 interface.




NOTE: When configuring E1 or T1 interfaces on 10-port Channelized E1/T1 IQE PICs, the invert-data statement must be included at the [edit interfaces e1-*fpc/pic/port*] or [edit interfaces t1-*fpc/pic/port*] hierarchy level as appropriate.

Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> Configuring E1 Data Inversion on page 164 Configuring E3 Data Inversion on page 173 Configuring T1 Data Inversion on page 181

isdn-options

Syntax	<pre>isdn-options { bchannel-allocation (ascending descending); calling-number <i>number</i>; incoming-called-number <i>number</i> <reject>; spid1 <i>spid-string</i>; spid2 <i>spid-string</i>; static-tei-val <i>value</i>; switch-type (att5e etsi nil ntdms100 ntt); t310 <i>seconds</i>; tei-option (first-call power-up); }</pre>
Hierarchy Level	<pre>[edit interfaces br-<i>pim</i>/0/<i>port</i>], [edit interfaces ct1-<i>pim</i>/0/<i>port</i>], [edit interfaces ce1-<i>pim</i>/0/<i>port</i>]</pre>
Release Information	Statement introduced before Junos OS Release 7.4. bchannel-allocation option added in Junos OS Release 8.3.
Description	<p>For J Series Services Routers only. Specify the ISDN options for configuring ISDN interfaces for group and user sessions.</p> <p>The statements are explained separately.</p>
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">Configuring ISDN Physical Interface PropertiesAllocating B-Channels for Dialout on page 155<i>Junos OS Interfaces and Routing Configuration Guide</i>

line-encoding

Syntax	line-encoding (ami b8zs);
Hierarchy Level	[edit interfaces ct1- <i>fpc/pic/port</i>], [edit interfaces <i>interface-name</i> t1-options]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Set the line encoding format on the T1 interface.
	<div>  <p>NOTE: When configuring CT1 interfaces on the 10-port Channelized E1/T1 IQE PIC, the line-encoding statement must be included at the [edit interfaces ct1-<i>fpc/pic/port</i>] hierarchy level.</p> </div>
Default	The default line encoding is B8ZS.
Options	ami —Use Alternate Mark Inversion (AMI) line encoding. b8zs —Use bipolar with 8-zeros substitution (B8ZS) line encoding.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> Configuring T1 Line Encoding on page 182

lmi (Frame Relay)

Syntax lmi {
 lmi-type (ansi | itu);
 n391dte *number*;
 n392dce *seconds*;
 n392dte *number*;
 n393dce *number*;
 n393dte *number*;
 t391dte *number*;
 t392dce *seconds*;
 }

Hierarchy Level [edit interfaces *interface-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Set Frame Relay keepalive parameters.

Options n391dte—DTE full status polling interval.
 Range: 1 through 255
 Default: 6

 n392dce—DCE error threshold, in number of errors.
 Range: 1 through 10
 Default: 3

 n392dte—DTE error threshold, in number of errors.
 Range: 1 through 10
 Default: 3

 n393dce—DCE monitored event-count.
 Range: 1 through 10
 Default: 4

 n393dte—DTE monitored event-count.
 Range: 1 through 10
 Default: 4


 t391dte—DTE polling timer.
 Range: 5 through 30 seconds
 Default: 10 seconds

 t392dce—DCE polling timer.
 Range: 5 through 30 seconds
 Default: 15 seconds

 The remaining statements are explained separately.

Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> Configuring Tunable Keepalives for Frame Relay LMI on page 210 lmi-type on page 263

lmi-type

Syntax	lmi-type (ansi itu c-lmi);
Hierarchy Level	[edit interfaces <i>interface-name</i> lmi], [edit interfaces <i>interface-name</i> mlfr-uni-nni-bundle-options]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Set Frame Relay Local Management Interface (LMI) type.
	<div>  <p>NOTE: Consortium LMI is supported only on M320 routers with Enhanced III FPCs and specific IQE PICs.</p> </div>
Options	<p>ansi—Use ANSI T1.167 Annex D LMIs.</p> <p>itu—Use ITU Q933 Annex A LMIs.</p> <p>c-lmi—Use Consortium LMI.</p> <p>Default: ansi</p>
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> Configuring Tunable Keepalives for Frame Relay LMI on page 210 Junos OS Services Interfaces Configuration Guide

lockout

Syntax	lockout;
Hierarchy Level	[edit interfaces <i>interface-name</i> sonet-options aps]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Configure a lockout of protection, forcing the use of the working circuit and locking out the protect circuit regardless of anything else.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">Configuring Switching Between the Working and Protect Circuits

long-buildout

Syntax	(long-buildout no-long-buildout);
Hierarchy Level	[edit interfaces <i>interface-name</i> t3-options]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	<p>Configure the T3 line buildout. A T3 interface has two settings for the T3 line buildout: a short setting, which is less than 255 feet (68 meters), and a long setting, which is greater than 255 feet and shorter than 450 feet (137 meters).</p> <p>This statement applies to copper-cable-based T3 interfaces only. You cannot configure a line buildout for a DS3 channel on a channelized OC12 interface, which runs over fiber-optic cable.</p>
Default	A T3 interface uses the short line buildout setting (no-long-buildout) for wires shorter than 255 feet (68 meters).
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">Configuring the T3 Line Buildout on page 194

loop-timing

Syntax	(loop-timing no-loop-timing);
Hierarchy Level	[edit interfaces ct3- <i>fpc/pic/port</i> t3-options], [edit interfaces e1- <i>fpc/pic/port:0</i> sonet-options], [edit interfaces stm1- <i>fpc/pic/port</i> sonet-options]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	For channelized IQ interfaces and non-IQ channelized STM1 interfaces only, configure the SONET/SDH or DS3-level clocking source.



NOTE: On M Series, MX Series, and T Series routers, under E1 channels, loop timing can be configured only at channel 0. When you configure on channel 0, it is applicable on all channels as internal by default.

Options	loop-timing —Configure loop timing (external) clocking. no-loop-timing —Configure line timing (internal) clocking. Default: no-loop-timing
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> Configuring Channelized IQ and IQE SONET/SDH Loop Timing Configuring the Channelized T3 Loop Timing clocking on page 230

loopback (ADSL, DS0, E1/E3, SONET/SDH, SHDSL, and T1/T3)

Syntax	<code>loopback (local payload remote);</code>
Hierarchy Level	<code>[edit interfaces ce1-fpc/pic/port],</code> <code>[edit interfaces ct1-fpc/pic/port],</code> <code>[edit interfaces t1-fpc/pic/port],</code> <code>[edit interfaces interface-name ds0-options],</code> <code>[edit interfaces interface-name dsl-options],</code> <code>[edit interfaces interface-name e1-options],</code> <code>[edit interfaces interface-name e3-options],</code> <code>[edit interfaces interface-name shdsl-options],</code> <code>[edit interfaces interface-name sonet-options],</code> <code>[edit interfaces interface-name t1-options],</code> <code>[edit interfaces interface-name t3-options]</code>
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Configure a loopback connection. To turn off the loopback capability, remove the loopback statement from the configuration.



NOTE: When configuring CE1 or CT1 interfaces on 10-port Channelized E1/T1 IQE PICs, the **loopback** statement must be included with the **local** or **remote** option at the `[edit interfaces ce1-fpc/pic/port]` or `[edit interfaces ct1-fpc/pic/port]` hierarchy level as appropriate.

When configuring T1 interfaces on 10-port Channelized E1/T1 IQE PICs, the **loopback** statement must be included with the **payload** option at the `[edit interfaces t1-fpc/pic/port]` hierarchy level.

To configure loopback on channelized IQ and IQE PICs, SONET/SDH level, use the **sonet-options loopback** statement **local** and **remote** options at the controller interface (`coc48`, `cstm16`, `coc12`, `cstm4`, `coc3`, `cstm1`). It is ignored for path-level interfaces **so-fpc/pic/port** or **so-fpc/pic/port:channel**.



Options	<p>local—Loop packets, including both data and timing information, back on the local router's PIC. NxDS0 IQ interfaces do not support local loopback.</p> <p>payload—For channelized T3, T1, and NxDS0 IQ interfaces only, loop back data only (without clocking information) on the remote router's PIC. With payload loopback, overhead is recalculated. Neither ATM-over-asymmetrical digital subscriber line (ADSL) interfaces nor ATM-over-SHDSL interfaces support payload loopback.</p> <p>remote—Loop packets, including both data and timing information, back on the remote router's interface card. NxDS0 IQ interfaces do not support remote loopback.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>

- Related Documentation**
- Configuring E3 and T3 Parameters on ATM Interfaces
 - Configuring E1 Loopback Capability on page 164
 - Configuring E3 Loopback Capability on page 173
 - Configuring Channelized IQ and IQE SONET/SDH Loop Timing
 - Configuring SHDSL Operating Mode on an ATM Physical Interface
 - Configuring T1 Loopback Capability on page 182
 - Configuring T3 Loopback Capability on page 194
 - **feac-loop-respond** on page 252

loopback-clear-timer

- Syntax** `loopback-clear-timer seconds;`
- Hierarchy Level** [edit interfaces *interface-name* unit *logical-unit-number* ppp-options],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* ppp-options]
- Release Information** Statement introduced in Junos OS Release 8.5.
- Description** For interfaces with PPP, PPP TCC, PPP over Ethernet, PPP over ATM, and PPP over Frame Relay encapsulations, configure a loop detection clear timer for the Link Control Protocol (LCP) component of a PPP session.
- Options** *seconds*—The time in seconds to wait before the loop detection flag is cleared if it is not cleared by the protocol.
Range: 1 through 60 seconds
Default: 9 seconds
- Required Privilege Level** interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.
- Related Documentation**
- Configuring the PPP Clear Loop Detected Timer

mtu

Syntax	<code>mtu bytes;</code>
Hierarchy Level	<code>[edit interfaces <i>interface-name</i>],</code> <code>[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> family <i>family</i>],</code> <code>[edit interfaces interface-range <i>name</i>],</code> <code>[edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i> family <i>family</i>]</code>
Release Information	<p>Statement introduced before Junos OS Release 7.4.</p> <p>Statement introduced in Junos OS Release 9.0 for EX Series switches.</p>
Description	<p>Specify the maximum transmission unit (MTU) size for the media or protocol. The default MTU size depends on the device type. Changing the media MTU or protocol MTU causes an interface to be deleted and added again.</p> <p>On EX Series switches, keep the following points in mind if you are configuring MTU size for jumbo frames on these special types of interfaces:</p> <ul style="list-style-type: none"> • For LAG interfaces—Configuring the jumbo MTU size on a link aggregation group (LAG) interface (aex) automatically configures the jumbo MTU size on the member links. • For RVIs—Jumbo frames of up to 9216 bytes are supported on the routed VLAN interface (RVI), which is named vlan. The RVI functions as a logical router. To route jumbo data packets on the RVI, you must configure the jumbo MTU size on the member physical interfaces of the RVI and not on the RVI itself (the vlan interface). However, for jumbo control packets—for example, to ping the RVI with a packet size of 6000 bytes or more—you must explicitly configure the jumbo MTU size on the interface named vlan (the RVI).
	<div>  <p>CAUTION: For EX Series switches, setting or deleting the jumbo MTU size on the RVI (the vlan interface) while the switch is transmitting packets might result in dropped packets.</p> </div>
	<div>  <p>NOTE: Not all devices allow you to set an MTU value, and some devices have restrictions on the range of allowable MTU values. You cannot configure an MTU for management Ethernet (fxp0, or em0, or me0) interfaces or for loopback, multilink, and multicast tunnel devices.</p> </div>
Options	<p>bytes—MTU size.</p> <p>Range: 256 through 9192 bytes</p>

For more information on configuring MTU for specific interfaces and router or switch combinations, see Configuring the Media MTU.

Default: 1500 bytes (INET, INET6, and ISO families), 1448 bytes (MPLS), 1514 bytes (EX Series interfaces)

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring the Media MTU
- Configuring Gigabit Ethernet Interfaces (CLI Procedure)
- Configuring Routed VLAN Interfaces (CLI Procedure)
- Setting the Protocol MTU

multicast-dlci

Syntax `multicast-dlci dlci-identifier;`

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number*],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*]

Release Information Statement introduced before Junos OS Release 7.4.

Description For point-to-multipoint Frame Relay, link services, and voice services interfaces only, enable multicast support on the interface. You can configure multicast support on the interface if the Frame Relay switch performs multicast replication.

Options *dlci-identifier*—DLCI identifier, a number from 16 through 1022 that defines the Frame Relay DLCI over which the switch expects to receive multicast packets for replication.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring a Multicast-Capable Frame Relay Connection on page 214
- **dlci on page 235**
- **multipoint-destination on page 270**
- [Junos OS Services Interfaces Configuration Guide](#)

multipoint-destination

Syntax	<pre>multipoint-destination <i>address</i> dlc <i>dlci-identifier</i>; multipoint-destination <i>address</i> { epd-threshold <i>cells</i>; inverse-arp; oam-liveness { down-count <i>cells</i>; up-count <i>cells</i>; } oam-period (disable <i>seconds</i>); shaping { (cbr <i>rate</i> rtvbr peak <i>rate</i> sustained <i>rate</i> burst <i>length</i> vbr peak <i>rate</i> sustained <i>rate</i> burst <i>length</i>); queue-length <i>number</i>; } vci <i>vpi-identifier.vci-identifier</i>; }</pre>
Hierarchy Level	[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> family <i>family</i> address <i>address</i>], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i> family <i>family</i> address <i>address</i>]
Release Information	Statement introduced before Junos OS Release 7.4. Statement introduced in Junos OS Release 11.1 for the QFX Series.
Description	For point-to-multipoint Frame Relay or ATM interfaces only, enable the support of multicast on the interface. You can configure multicast support on the interface if the Frame Relay or ATM switch performs multicast replication.
Options	<p><i>address</i>—Address of the remote side of the point-to-multipoint connection.</p> <p><i>dlci-identifier</i>—For Frame Relay interfaces, the data-link connection identifier. Range: 0 through 0xFFFFF (24 bits)</p> <p><i>vci-identifier</i>—For ATM interfaces, the virtual circuit identifier. Range: 0 through 16,384</p> <p><i>vpi-identifier</i>—For ATM interfaces, the virtual path identifier. Range: 0 through 255 Default: 0</p> <p>The remaining statements are explained separately.</p>
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">Configuring a Point-to-Point ATM1 or ATM2 IQ ConnectionConfiguring a Point-to-Multipoint Frame Relay Connection on page 213dlci on page 235

- [encapsulation \(Logical Interface\)](#) on page 240

neighbor

Syntax	<code>neighbor <i>address</i>;</code>
Hierarchy Level	[edit interfaces <i>interface-name</i> sonet-options aps]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	<p>If you are configuring one router to be the working router and a second to be the protect router, configure the address of the remote interface. You configure this on one or both of the interfaces.</p> <p>The address you specify for the neighbor must never be routed through the interface on which APS is configured, or instability will result. We strongly recommend that you directly connect the working and protect routers and that you configure the interface address of this shared network as the neighbor address.</p>
Options	<i>address</i> —Neighbor's address.
Required Privilege Level	<code>interface</code> —To view this statement in the configuration. <code>interface-control</code> —To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Basic APS Support

no-keepalives

Syntax	no-keepalives;
Hierarchy Level	[edit interfaces <i>interface-name</i>], [edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i>], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i>]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	<p>Disable the sending of keepalives on a physical interface configured with PPP, Frame Relay, or Cisco HDLC encapsulation. The default keepalive interval is 10 seconds.</p> <p>For ATM2 IQ interfaces only, you can disable keepalives on a logical interface unit if the logical interface is configured with one of the following PPP over ATM encapsulation types:</p> <ul style="list-style-type: none">• atm-ppp-llc—PPP over AAL5 LLC encapsulation.• atm-ppp-vc-mux—PPP over AAL5 multiplex encapsulation.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Keepalives• Disabling the Sending of PPPoE Keepalive Messages• Configuring Frame Relay Keepalives on page 210

no-partition

Syntax	no-partition interface-type (e1 (cau4 so) (ct3 t3) so t3);
Hierarchy Level	<pre>[edit interfaces ce1-fpc/pic/port], [edit interfaces coc1-fpc/pic/port:channel], [edit interfaces coc12-fpc/pic/port], [edit interfaces cstm1-fpc/pic/port], [edit interfaces ct3-fpc/pic/port]</pre>
Release Information	Statement introduced before Junos OS Release 7.4.
Description	<p>For Channelized E1 IQ PICs only, configure the channelized E1 interface as an unpartitioned, clear channel.</p> <p>For Channelized OC12 PIC only, convert the channelized OC1 IQ interface into a channelized T3 interface or a T3 interface. You perform this configuration task for C-bit parity and M13-mapped configurations.</p> <p>For Channelized OC12 IQ PICs only, configure the channelized OC12 interface as an unpartitioned, clear channel.</p> <p>For Channelized STM1 PIC only, convert the channelized STM1 IQ interface into a channelized Administrative Unit 4 (AU-4) interface or a SONET/SDH STM1 interface.</p> <p>For Channelized DS3 PIC only, configure the channelized T3 interface as an unpartitioned, clear channel.</p>
Default	If you do not include either this statement or the partition statement, the Channelized IQ PIC is not partitioned, and no data channels are configured.
Options	<p>The option used must correspond to the physical interface type:</p> <p>e1—E1 interface type.</p> <p>coc12 so—Channelized OC12 interface type, in SONET mode.</p> <p>cau4—Channelized AU-4 interface type.</p> <p>cstm1—SONET/SDH STM1 interface type, in SDH mode.</p> <p>ct3—Channelized T3 interface type.</p> <p>t3—T3 interface type.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> Channelized E1 IQ and IQE Interfaces Overview on page 143 Channelized OC12/STM4 IQ and IQE Interfaces Overview on page 65 Configuring an OC12/STM4 Interface on page 69

- Configuring Channelized STM1 IQ and IQE Interfaces on page 105
- Configuring T3 IQ Interfaces on page 123
- **partition on page 276**

no-termination-request

Syntax	no-termination-request;
Hierarchy Level	[edit interfaces <i>interface-name</i> ppp-options], [edit interfaces lsq- <i>fpc/pic/port</i> lsq-failure-options]
Release Information	Statement introduced in Junos OS Release 7.4. Support at the [edit interfaces <i>interface-name</i> ppp-options] hierarchy level added in Junos OS Release 8.3.
Description	For LSQ PICs or link PICs in redundant LSQ configurations, you can inhibit the router from sending PPP termination-request messages to the remote host if the PIC fails.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Link PIC Failover on Channelized OC3 IQ and IQE Interfaces on page 104• Configuring Link PIC Failover on Channelized OC12/STM4 IQ and IQE Interfaces on page 90• Configuring Link PIC Failover on Channelized STM1 Interfaces on page 119• Junos OS Services Interfaces Configuration Guide

oc-slice

Syntax	<code>oc-slice <i>oc-slice-range</i>;</code>
Hierarchy Level	[edit interfaces <i>interface-name</i> partition <i>partition-number</i>]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	For channelized OC12 IQ interfaces only, configure the range of SONET/SDH slices.
Default	If you do not include either this statement or the no-partition statement, the Channelized OC12 IQ PICs not partitioned, and no data channels are configured.
Options	<p><i>oc-slice-range</i>—Range of SONET/SDH slices. OC3 interfaces must occupy three consecutive OC slices per interface, in the form 1–3, 4–6, 7–9, or 10–12. The T3, T1, and DS0 interface types each occupy one OC slice per interface.</p> <p>Range: For OC3 interfaces, 1–3, 4–6, 7–9, or 10–12; for SONET/SDH and T3 interfaces, 1–12</p> <p>The remaining statement is explained separately.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> Channelized OC12/STM4 IQ and IQE Interfaces Overview on page 65


paired-group

Syntax	<code>paired-group <i>group-name</i>;</code>
Hierarchy Level	[edit interfaces <i>interface-name</i> sonet-options aps]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Configure load sharing between two working protect circuit pairs.
Options	<i>group-name</i> —Circuit's group name, as configured with the protect-circuit or working-circuit statement.
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> Configuring APS Load Sharing Between Circuit Pairs working-circuit on page 294

partition

Syntax	<code>partition <i>partition-number</i> oc-slice <i>oc-slice-range</i> interface-type <i>type</i> timeslots <i>time-slot-range</i>;</code>
Hierarchy Level	[edit interfaces <i>interface-name</i>]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	For IQ interfaces and J Series interfaces on the Dual-Port Channelized E1 and T1PIM, configure the channelized interface partition. The partition number is correlated with the channel number. Partition and channel numbering on IQ interfaces begins with :1, not :0.
Default	If you omit this statement, the channelized PIC or PIM is not partitioned, and no data channels are configured.
Options	<p><i>partition-number</i>—Sublevel interface partition index.</p> <p>Range:</p> <ul style="list-style-type: none">• 1 through 4 for an OC3 interface on a channelized OC12 IQ interface.• 1 through 12 for a T3 interface on a channelized OC12 IQ interface.• 1 through 4 for a T3 interface on a channelized T3 IQ interface.• 1 through 28 for a T1 IQ interface on a channelized OC12 IQ or channelized T3 IQ interface.• 1 through 10 for an E1 interface on a channelized E1 IQ interface.• 1 through 30 on a channelized E1 interface.• 1 through 23 on a channelized T1 interface.• 1 through 24 for NxDS0 interfaces on either channelized OC12 IQ or channelized DS3 IQ interfaces.• 0 through 31 (with 0 reserved for framing) for NxDS0 interfaces on channelized E1 IQ interfaces. <p>The remaining statements are explained separately.</p>
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Channelized E1 IQ and IQE Interfaces Overview on page 143• Channelized OC12/STM4 IQ and IQE Interfaces Overview on page 65• Configuring Channelized T3 IQ Interfaces on page 123• no-partition on page 273

payload-scrambler

Syntax	(payload-scrambler no-payload-scrambler);
Hierarchy Level	[edit interfaces <i>interface-name</i> e3-options], [edit interfaces <i>interface-name</i> sonet-options], [edit interfaces <i>interface-name</i> t3-options]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	<p>Enable or disable HDLC scrambling on an E3, a SONET/SDH, or a T3 interface. This type of scrambling provides better link stability. Both sides of a connection must either use or not use scrambling.</p> <p>If you commit a T3 interface configuration that has HDLC payload scrambling enabled, the interface must also be configured to be compatible with the channel service unit (CSU) at the remote end of the line.</p> <p>Disable payload scrambling on an E3 interface if Digital Link compatibility mode is used.</p> <p>On a channelized OC12 interface, the sonet payload-scrambler statement is ignored. To configure scrambling on the DS3 channels on the interface, you can include the t3-options payload-scrambler statement in the configuration for each DS3 channel.</p>
	<div>  <p>NOTE: The payload-scrambler statement at the [edit interfaces <i>interface-name</i> e3-options] hierarchy level is not valid for IQE PICs.</p> </div>
Default	Payload scrambling is disabled on all E3 and T3 interfaces; it is enabled by default on E3/T3 over ATM interfaces and on SONET/SDH interfaces.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> Configuring E3 and T3 Parameters on ATM Interfaces Configuring E3 HDLC Payload Scrambling on page 175 Configuring SONET/SDH HDLC Payload Scrambling Configuring T3 HDLC Payload Scrambling on page 196 Examples: Configuring T3 Interfaces on page 197 compatibility-mode on page 231

protect-circuit

Syntax	<code>protect-circuit <i>group-name</i>;</code>
Hierarchy Level	[edit interfaces <i>interface-name</i> sonet-options aps]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Configure the protect router in an APS circuit pair. When the working interface fails, APS brings up the protection circuit and the traffic is moved to the protection circuit.
Options	<i>group-name</i> —Circuit's group name.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">Configuring Basic APS Supportworking-circuit on page 294

remote-loopback-respond

Syntax	<code>remote-loopback-respond;</code>
Hierarchy Level	[edit interfaces ct1- <i>fpc/pic/port</i>], [edit interfaces <i>interface-name</i> t1-options]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	For T1 interfaces only, configure the router to respond to remote loopback requests. Remote loopback requests can be from the facilities data link or inband.



NOTE: When configuring CT1 interfaces on the 10-port Channelized E1/T1 IQE PIC, the `remote-loopback-respond` statement must be included at the [edit interfaces ct1-*fpc/pic/port*] hierarchy level.

Default	The router does not respond to remote loop requests.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">Configuring the T1 Remote Loopback Response on page 181feac-loop-respond on page 252loopback (ADSL, DS0, E1/E3, SONET/SDH, SHDSL, and T1/T3) on page 266

request

Syntax	<code>request (protect working);</code>
Hierarchy Level	[edit interfaces <i>interface-name</i> sonet-options aps]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Perform a manual switch between the protect and working circuits. This statement is honored only if there are no higher-priority reasons to switch.
Options	<p>protect—Request that the circuit become the protect circuit.</p> <p>working—Request that the circuit become the working circuit.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> Configuring Switching Between the Working and Protect Circuits force on page 253

revert-time

Syntax	<code>revert-time <i>seconds</i>;</code>
Hierarchy Level	[edit interfaces <i>interface-name</i> sonet-options aps]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Configure APS revertive mode.
Default	APS operates in nonrevertive mode.
Options	<p>seconds—Amount of time to wait after the working circuit has again become functional before making the working circuit active again.</p> <p>Range: 1 through 65,535 seconds</p> <p>Default: None (APS operates in nonrevertive mode)</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> Configuring Revertive Mode

sonet-options

```
Syntax  sonet-options {
        aps {
            advertise-interval milliseconds;
            annex-b
            authentication-key key;
            force;
            hold-time milliseconds;
            lockout;
            neighbor address;
            paired-group group-name;
            protect-circuit group-name;
            request;
            revert-time seconds;
            switching-mode (bidirectional | unidirectional);
            working-circuit group-name;
        }
        bytes {
            c2 value;
            e1-quiet value;
            f1 value;
            f2 value;
            s1 value;
            z3 value;
            z4 value;
        }
        fcs (16 | 32);
        loopback (local | remote);
        mpls {
            pop-all-labels {
                required-depth number;
            }
        }
        path-trace trace-string;
        (payload-scrambler | no-payload-scrambler);
        rfc-2615;
        trigger {
            defect ignore;
            defect hold-time up milliseconds down milliseconds;
        }
    }
    vtmapping (itu-t | klm);
    (z0-increment | no-z0-increment);
```

Hierarchy Level [edit interfaces *interface-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure SONET/SDH-specific interface properties.

On SONET/SDH OC48 interfaces that you configure for channelized (multiplexed) mode (by including the **no-concatenate** statement at the [edit chassis fpc *slot-number* pic *pic-number*] hierarchy level), the **bytes e1-quiet** and **bytes f1** options have no effect. The

bytes f2, **bytes z3**, **bytes z4**, and **path-trace** options work correctly on channel 0 and work in the transmit direction only on channels 1, 2, and 3.

On a channelized OC12 interface, the **bytes e1-quiet**, **bytes f1**, **bytes f2**, **bytes z3**, and **bytes z4** options are not supported. The **fcs** and **payload-scrambler** statements are also not supported; you must configure these for each DS3 channel using the **t3-options fcs** and **t3-options payload-scrambler** statements. The **aps** and **loopback** statements are supported only on channel 0 and are ignored if included in the configurations for channels 1 through 11. You can configure loopbacks for each DS3 channel with the **t3-options loopback** statement. The **path-trace** statement can be included in the configuration for each DS3 channel, thereby configuring a unique path trace for each channel.


To configure loopback on channelized IQ and IQE PICs, SONET/SDH level, use the **loopback** statement **local** and **remote** options at the controller interface (coc48, cstm16, coc12, cstm4, coc3, and cstm1). It is ignored for path-level interfaces *so-fpc/pic/port* or *so-fpc/pic/port:channel*.

If you are running Intermediate System-to-Intermediate System (IS-IS) over SONET/SDH interfaces, use PPP if you are running Cisco IOS Release 12.0 or later. If you need to run HDLC, configure an ISO family MTU of 4469 on the router.

The statements are explained separately.

Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring SONET/SDH Parameters on ATM Interfaces • Channelized OC12/STM4 IQ and IQE Interfaces Overview on page 65 • Channelized STM1 Interfaces Overview on page 105 • Configuring SONET/SDH Physical Interface Properties • no-concatenate

start-end-flag

Syntax	start-end-flag (filler shared);
Hierarchy Level	[edit interfaces e1- <i>fpc/pic/port</i>], [edit interfaces t1- <i>fpc/pic/port</i>], [edit interfaces <i>interface-name</i> ds0-options], [edit interfaces <i>interface-name</i> e1-options], [edit interfaces <i>interface-name</i> e3-options], [edit interfaces <i>interface-name</i> t1-options], [edit interfaces <i>interface-name</i> t3-options]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	For DS0, E1, E3, T1, and T3 interfaces, configure the interface to share the transmission of start and end flags.
	<div> NOTE: When configuring E1 or T1 interfaces on the 10-port Channelized E1/T1 IQE PIC, the start-end-flag statement must be included at the [edit interfaces e1-<i>fpc/pic/port</i>] or [edit interfaces t1-<i>fpc/pic/port</i>] hierarchy level as appropriate.</div>
Options	filler —Wait two idle cycles between the start and end flags. shared —Share the transmission of the start and end flags. This is the default.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring E1 Start and End Flags on page 166• Configuring the E3 Start and End Flags on page 175• Configuring T1 Start and End Flags on page 184• Configuring T3 Start and End Flags on page 196

switching-mode

Syntax	switching-mode (bidirectional unidirectional);
Hierarchy Level	[edit interfaces <i>interface-name</i> sonet-options aps]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	For unchannelized OC3, OC12, and OC48 SONET/SDH interfaces on T Series routers only, configure the interface to interoperate with SONET/SDH line-terminating equipment (LTE) that is provisioned for unidirectional linear APS in 1+1 architecture.
Default	If the switching-mode statement is not configured, the mode is bidirectional, and the interface does not interoperate with a unidirectional SONET/SDH LTE.
Options	bidirectional —Support bidirectional mode only. unidirectional —Interoperate with a SONET/SDH LTE provisioned for unidirectional mode.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">Configuring Unidirectional Switching Mode Support

t1-options

Syntax t1-options {
 bert-algorithm *algorithm*;
 bert-error-rate *rate*;
 bert-period *seconds*;
 buildout *value*;
 byte-encoding (nx56 | nx64);
 crc-major-alarm-threshold (1e-3 | 5e-4 | 1e-4 | 5e-5 | 1e-5);
 crc-minor-alarm-threshold (1e-3 | 5e-4 | 1e-4 | 5e-5 | 1e-5 | 5e-6 | 1e-6);
 fcs (16 | 32);
 framing (esf | sf);
 idle-cycle-flag (flags | ones);
 invert-data;
 line-encoding (ami | b8zs);
 loopback (local | payload | remote);
 remote-loopback-respond;
 start-end-flag (filler | shared);
 timeslots *time-slot-range*;
 }

Hierarchy Level [edit interfaces *interface-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure T1-specific physical interface properties.

 The statements are explained separately.

Required Privilege Level interface—To view this statement in the configuration.
 interface-control—To add this statement to the configuration.

Related Documentation • T1 Interfaces Overview on page 177

t3-options

Syntax	<pre> t3-options { atm-encapsulation (direct plcp); bert-algorithm <i>algorithm</i>; bert-error-rate <i>rate</i>; bert-period <i>seconds</i>; (cbit-parity no-cbit-parity); compatibility-mode (digital-link kentrox larscom) <subrate <i>value</i>>; fcs (16 32); (feac-loop-respond no-feac-loop-respond); idle-cycle-flag <i>value</i>; (long-buildout no-long-buildout); (loop-timing no-loop-timing); loopback (local payload remote); start-end-flag <i>value</i>; } </pre>
Hierarchy Level	[edit interfaces <i>interface-name</i>]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	<p>Configure T3-specific physical interface properties, including the properties of DS3 channels on a channelized OC12 interface. The long-buildout statement is not supported for DS3 channels on a channelized OC12 interface.</p> <p>On T3 interfaces, the default encapsulation is PPP.</p> <p>For ATM1 interfaces, you can configure a subset of E3 options statements.</p> <p>The statements are explained separately.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> T3 Interfaces Overview on page 187

timeslots

Syntax	<code>timeslots <i>time-slot-range</i>;</code>
Hierarchy Level	[edit interfaces <i>e1-fpc/pic/port</i>], [edit interfaces <i>t1-fpc/pic/port</i>], [edit interfaces <i>interface-name</i> e1-options], [edit interfaces <i>interface-name</i> partition <i>partition-number</i>], [edit interfaces <i>interface-name</i> t1-options]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	For E1 and T1 interfaces, allocate the specific time slots by number.



NOTE: When configuring E1 or T1 interfaces on the 10-port Channelized E1/T1 IQE PIC, the `timeslots` statement must be included at the [edit interfaces *e1-fpc/pic/port*] or [edit interfaces *t1-fpc/pic/port*] hierarchy level as appropriate.

Options	<p><i>time-slot-range</i>—Actual time slot numbers allocated:</p> <p>Range: Ranges vary by interface type and configuration option as follows:</p> <ul style="list-style-type: none"> • 1 through 24 for T1 interfaces (0 is reserved) • 1 through 31 for 4-port E1 PICs (0 is reserved) • 1 through 31 for NxDS0 interfaces (0 is reserved) • 2 through 32 for 10-port Channelized E1 and 10-port Channelized E1 IQ PICs (1 is reserved) • 2 through 32 for the setting under e1-options with IQE PICs (1 is reserved) (when creating fractional E1) • 1 through 31 for the setting under partition with IQE PICs (0 is reserved) (when creating NxDS0)
----------------	--



NOTE: When creating fractional E1 interfaces only, if you connect a 4-port E1 PIC interface to a device that uses time slot numbering from 2 through 32, you must subtract 1 from the configured number of time slots.

Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • Configuring Fractional E1 IQ and IQE Interfaces on page 144 • Configuring Fractional T1 IQ and IQE Interfaces on page 124

- Configuring Fractional E1 Time Slots on page 166
- Configuring Fractional T1 Time Slots on page 185
- Configuring a Channelized T1/E1 Interface to Drop and Insert Time Slots on page 152

translate-discard-eligible

Syntax	(translate-discard-eligible no-translate-discard-eligible);
Hierarchy Level	[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> family <i>ccc</i>], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i> family <i>ccc</i>]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	For interfaces with encapsulation type Frame Relay CCC, enable or disable translation of Frame Relay discard eligible (DE) control bits.
Default	DE bit translation is disabled.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring Frame Relay Control Bit Translation on page 208

unframed

Syntax	(unframed no-unframed);
Hierarchy Level	[edit interfaces <i>interface-name</i> e3-options]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	For E3 IQ interfaces only, enable or disable unframed mode. In unframed mode, the E3 IQ interface do not detect yellow (ylw) or loss-of-frame (lof) alarms.
Default	Unframed mode is disabled.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring E3 IQ and IQE Unframed Mode on page 175

unit

Syntax `unit logical-unit-number {`
 `accept-source-mac {`
 `mac-address mac-address {`
 `policer {`
 `input cos-policer-name;`
 `output cos-policer-name;`
 `}`
 `}`
 `}`
 `accounting-profile name;`
 `allow-any-vci;`
 `atm-scheduler-map (map-name | default);`
 `backup-options {`
 `interface interface-name;`
 `}`
 `bandwidth rate;`
 `cell-bundle-size cells;`
 `clear-dont-fragment-bit;`
 `compression {`
 `rtp {`
 `maximum-contexts number <force>;`
 `f-max-period number;`
 `queues [queue-numbers];`
 `port {`
 `minimum port-number;`
 `maximum port-number;`
 `}`
 `}`
 `}`
 `compression-device interface-name;`
 `copy-tos-to-outer-ip-header;`
 `demux-destination family;`
 `demux-source family;`
 `demux-options {`
 `underlying-interface interface-name;`
 `}`
 `description text;`
 `dial-options {`
 `l2tp-interface-id name;`
 `(dedicated | shared);`
 `}`
 `dialer-options {`
 `activation-delay seconds;`
 `callback;`
 `callback-wait-period time;`
 `deactivation-delay seconds;`
 `dial-string [dial-string-numbers];`
 `idle-timeout seconds;`
 `incoming-map {`
 `caller caller-id) | accept-all;`
 `initial-route-check seconds;`
 `load-interval seconds;`
 `}`
 `}`

```

    load-threshold percent;
    pool pool-name;
    redial-delay time;
    watch-list {
        [ routes ];
    }
}
disable;
disable-mlppp-inner-ppp-pfc;
dlci dlci-identifier;
drop-timeout milliseconds;
dynamic-call-admission-control {
    activation-priority priority;
    bearer-bandwidth-limit kilobits-per-second;
}
encapsulation type;
epd-threshold cells plp1 cells;
family family-name {
    ... the family subhierarchy appears after the main [edit interfaces interface-name unit
        logical-unit-number] hierarchy ...
}
fragment-threshold bytes;
inner-vlan-id-range start start-id end end-id;
input-vlan-map {
    (pop | pop-pop | pop-swap | push | push-push | swap |
    swap-push | swap-swap);
    inner-tag-protocol-id tpid;
    inner-vlan-id number;
    tag-protocol-id tpid;
    vlan-id number;
}
interleave-fragments;
inverse-arp;
layer2-policer {
    input-policer policer-name;
    input-three-color policer-name;
    output-policer policer-name;
    output-three-color policer-name;
}
link-layer-overhead percent;
minimum-links number;
mrru bytes;
multicast-dlci dlci-identifier;
multicast-vci vpi-identifier.vci-identifier;
multilink-max-classes number;
multipoint;
oam-liveness {
    up-count cells;
    down-count cells;
}
oam-period (disable | seconds);
output-vlan-map {
    (pop | pop-pop | pop-swap | push | push-push | swap |
    swap-push | swap-swap);
    inner-tag-protocol-id tpid;

```

```
    inner-vlan-id number;  
    tag-protocol-id tpid;  
    vlan-id number;  
  }  
  passive-monitor-mode;  
  peer-unit unit-number;  
  plp-to-clp;  
  point-to-point;  
  ppp-options {  
    chap {  
      access-profile name;  
      default-chap-secret name;  
      local-name name;  
      passive;  
    }  
    compression {  
      acfc;  
      pfc;  
    }  
    dynamic-profile profile-name;  
    lcp-restart-timer milliseconds;  
    loopback-clear-timer seconds;  
    ncp-restart-timer milliseconds;  
    pap {  
      access-profile name;  
      default-pap-password password;  
      local-name name;  
      local-password password;  
      passive;  
    }  
  }  
  pppoe-options {  
    access-concentrator name;  
    auto-reconnect seconds;  
    (client | server);  
    service-name name;  
    underlying-interface interface-name;  
  }  
  pppoe-underlying-options {  
    access-concentrator name;  
    dynamic-profile profile-name;  
    max-sessions number;  
  }  
  proxy-arp;  
  service-domain (inside | outside);  
  shaping {  
    (cbr rate | rtvbr peak rate sustained rate burst length | vbr peak rate sustained rate burst  
      length);  
    queue-length number;  
  }  
  short-sequence;  
  targeted-distribution;  
  transmit-weight number;  
  (traps | no-traps);  
  trunk-bandwidth rate;  
  trunk-id number;
```

```

tunnel {
    backup-destination address;
    destination address;
    key number;
    routing-instance {
        destination routing-instance-name;
    }
    source source-address;
    ttl number;
}
vci vpi-identifier.vci-identifier;
vci-range start start-vci end end-vci;
vpi vpi-identifier;
vlan-id number;
vlan-id-range number-number;
vlan-tags inner tpid.vlan-id outer tpid.vlan-id;
family family {
    accounting {
        destination-class-usage;
        source-class-usage {
            (input | output | input output);
        }
    }
    access-concentrator name;
    address address {
        ... the address subhierarchy appears after the main [edit interfaces interface-name unit
            logical-unit-number family family-name] hierarchy ...
    }
    bridge-domain-type (bvlan | svlan);
    bundle interface-name;
    core-facing;
    demux-destination {
        destination-prefix;
    }
    demux-source {
        source-prefix;
    }
    duplicate-protection;
    dynamic-profile profile-name;
    filter {
        group filter-group-number;
        input filter-name;
        input-list [ filter-names ];
        output filter-name;
        output-list [ filter-names ];
    }
    interface-mode (access | trunk);
    ipsec-sa sa-name;
    isid-list all-service-groups;
    keep-address-and-control;
    mac-validate (loose | strict);
    max-sessions number;
    mtu bytes;
    multicast-only;
    no-redirects;
    policer {

```

```
    arp policer-template-name;  
    input policer-template-name;  
    output policer-template-name;  
  }  
  primary;  
  protocols [inet iso mpls];  
  proxy inet-address address;  
  receive-options-packets;  
  receive-ttl-exceeded;  
  remote (inet-address address | mac-address address);  
  rpf-check {  
    fail-filter filter-name  
    mode loose;  
  }  
  sampling {  
    input;  
    output;  
  }  
  service {  
    input {  
      post-service-filter filter-name;  
      service-set service-set-name <service-filter filter-name>;  
    }  
    output {  
      service-set service-set-name <service-filter filter-name>;  
    }  
  }  
  service-name-table table-name  
  (translate-discard-eligible | no-translate-discard-eligible);  
  (translate-fecn-and-becn | no-translate-fecn-and-becn);  
  unnumbered-address interface-name destination address destination-profile profile-name;  
  vlan-id number;  
  vlan-id-list [number number-number];  
  address address {  
    arp ip-address (mac | multicast-mac) mac-address <publish>;  
    broadcast address;  
    destination address;  
    destination-profile name;  
    eui-64;  
    master-only;  
    multipoint-destination address {  
      dlci dlci-identifier;  
      epd-threshold cells <plp1 cells>;  
      inverse-arp;  
      oam-liveness {  
        up-count cells;  
        down-count cells;  
      }  
      oam-period (disable | seconds);  
      shaping {  
        (cbr rate | rtvbr burst length peak rate sustained rate | vbr burst length peak rate  
          sustained rate);  
        queue-length number;  
      }  
      vci vpi-identifier.vci-identifier;  
    }  
  }  
}
```



```

preferred;
primary;
(vrrp-group | vrrp-inet6-group) group-number {
  (accept-data | no-accept-data);
  advertise-interval seconds;
  authentication-type authentication;
  authentication-key key;
  fast-interval milliseconds;
  (preempt | no-preempt) {
    hold-time seconds;
  }
  priority number;
  track {
    interface interface-name {
      bandwidth-threshold bits-per-second priority-cost number;
    }
    priority-hold-time seconds;
    route ip-address/prefix-length routing-instance instance-name priority-cost cost;
  }
  virtual-address [ addresses ];
  virtual-link-local-address ipv6-address;
  vrrp-inherit-from {
    active-interface interface-name;
    active-group group-number;
  }
}
}
}

```

Hierarchy Level	[edit interfaces <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i>], [edit interfaces interface-set <i>interface-set-name</i> interface <i>interface-name</i>]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Configure a logical interface on the physical device. You must configure a logical interface to be able to use the physical device.
Options	<p>logical-unit-number—Number of the logical unit.</p> <p>Range: 0 through 1,073,741,823 for demux and PPPoE static interfaces only. 0 through 16,385 for all other static interface types.</p> <p>The remaining statements are explained separately.</p>
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> Configuring Logical Interface Properties Example: Configuring E-LINE and E-LAN Services for a PBB Network on MX Series Routers Junos OS Services Interfaces Configuration Guide

vtmapping

Syntax	vtmapping (itu-t klm);
Hierarchy Level	[edit interfaces <i>interface-name</i> sonet-options]; [edit chassis <i>fpc number</i> pic <i>number</i>]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	<p>For the Channelized STM1 IQ PIC or Channelized STM1 PIC, configure virtual tributary mapping.</p> <p>For the Channelized STM1 PIC, you configure virtual tributary mapping at the [edit chassis <i>fpc number</i> pic <i>number</i>] hierarchy level.</p>
Options	<p>itu-t—International Telephony Union standard.</p> <p>klm—KLM standard.</p> <p>Default: klm</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none">Configuring Virtual Tributary Mapping of Channelized STM1 Interfaces on page 112Junos OS System Basics Configuration Guide

working-circuit

Syntax	working-circuit <i>group-name</i> ;
Hierarchy Level	[edit interfaces <i>interface-name</i> sonet-options aps]
Release Information	Statement introduced before Junos OS Release 7.4.
Description	Configure the working router in an APS circuit pair.
Options	<i>group-name</i> —Circuit's group name.
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none">Configuring Basic APS Supportprotect-circuit on page 278

PART 6

Index

- Index on page 297
- Index of Statements and Commands on page 303

Index

Symbols

#, comments in configuration statements.....	xxvi
(), in syntax descriptions.....	xxvi
< >, in syntax descriptions.....	xxv
[], in configuration statements.....	xxvi
{ }, in configuration statements.....	xxvi
(pipe), in syntax descriptions.....	xxvi

A

address statement.....	218
advertise-interval statement.....	219
aps statement.....	220
authentication-key statement.....	221

B

bchannel-allocation statement.....	221
bert-algorithm statement.....	222
usage guidelines.....	170, 178, 188
bert-error-rate statement.....	224
bert-period statement.....	225
braces, in configuration statements.....	xxvi
brackets	
angle, in syntax descriptions.....	xxv
square, in configuration statements.....	xxvi
buildout statement	
T1 interfaces.....	226
byte encoding.....	227
byte-encoding statement.....	227
bytes statement.....	228

C

C-bit parity mode.....	229
cbit-parity statement.....	229
channelized AU-4 interfaces.....	58, 78, 84
example configuration.....	58, 79, 84
channelized E1 interfaces	
example configuration.....	60, 81, 110, 145, 148
interface naming.....	110, 145
channelized E1 IQ and IQE interfaces	
time slots.....	107

channelized E1 IQ interfaces	
example configuration.....	147
time slots.....	144
channelized E1 IQE interfaces	
example configuration.....	147
time slots.....	144
channelized E3 (COC48/STM16) interfaces	
example configuration.....	59
channelized interfaces	
clock sources.....	32
channelized IQ interfaces	
supported options.....	36
channelized NxDS0 IQ interfaces	
example configuration.....	61, 139, 145
channelized NxDS0 IQE interfaces	
example configuration.....	145
channelized OC12 (COC48/STM16 IQE)	
interfaces.....	49
example configuration.....	50
channelized OC12 interfaces	
example configuration.....	87, 88, 94
channelized OC12 IQ interfaces	
example configuration.....	90
channelized OC3 (COC48/STM16 IQE)	
interfaces.....	50
example configuration.....	51
channelized OC3 interfaces.....	71
example configuration.....	71
channelized OC48 IQE interfaces	
example configuration.....	62
channelized STM1 interfaces	
example configuration.....	119
time slots.....	108, 144
channelized STM1 IQ interfaces	
example configuration.....	109
channelized T1 (COC48/STM16 IQE) interfaces	
VT mapping.....	53
channelized T1 interfaces	
example configuration.....	86, 156
VT mapping.....	72, 81

channelized T1 IQ and IQE interfaces	
example configuration.....	86
channelized T1 IQ interfaces	
VT mapping.....	87, 102
channelized T1 IQE interfaces	
VT mapping.....	60
channelized T1 or CT1 (COC48/STM16, SDH mode)	
IQE interfaces	
example configuration.....	61
channelized T3 (COC48/STM16 IQE) interfaces	
example configuration.....	52
channelized T3 interfaces	
example configuration.....	70
channelized T3 IQ and IQE interfaces	
example configuration.....	85
channelized T3 IQ interfaces	
example configuration.....	100
channelized T3 IQE interfaces	
example configuration.....	100
Cisco HDLC encapsulation	
example configuration.....	197
clock sources.....	230
channelized interfaces.....	32
clocking statement.....	230
comments, in configuration statements.....	xxvi
compatibility-mode statement.....	231
conventions	
text and syntax.....	xxv
crc-minor-alarm-threshold statement.....	232
curly braces, in configuration statements.....	xxvi
customer support.....	xxvi
contacting JTAC.....	xxvi

D

data circuit-terminating equipment	See DCE
data terminal equipment	See DTE
data-input statement.....	233
DCE.....	197, 212, 234
dce statement.....	234
usage guidelines.....	212
dlci statement.....	235
documentation	
comments on.....	xxvi
ds0-options statement.....	236
usage guidelines.....	110, 145
DTE.....	197

E

E1 interfaces	
configuration statements.....	161
example configuration.....	110, 145
ITU-T standards.....	161
physical interface properties.....	161
time slots	
example configuration.....	167
E1 IQ interfaces	
example configuration.....	106
E1 IQE interfaces	
example configuration.....	106
e1-options statement.....	237
usage guidelines.....	110, 145, 161
E3 interfaces	
configuration statements.....	169
ITU-T standards.....	169
e3-options statement.....	238
usage guidelines.....	169
encapsulation	
media MTU size and.....	209
encapsulation statement.....	239
logical interface.....	240
physical interface.....	243
encoding	
byte.....	227
Enhanced IQ (IQE) interfaces	
channelized OC3.....	97
external clock sources.....	230

F

family statement.....	247
fcs statement.....	251
feac-loop-respond statement.....	252
font conventions.....	xxv
force statement.....	253
fractional E1 IQ and IQE interfaces	
time slots.....	107
fractional E1 IQ interfaces	
example configuration.....	107, 144
time slots.....	144
fractional E1 IQE interfaces	
example configuration.....	144
fractional T1 (COC48/STM16 IQE) interfaces	
example configuration.....	54
time slots.....	54
fractional T1 interfaces	
example configuration.....	75
time slots.....	75

fractional T1 IQ interfaces	
example configuration.....	104, 125, 138
Frame Relay encapsulation	
DCE.....	212, 234
example configuration.....	197, 207
Frame Relay protocol.....	203
Frame Relay Ether Type encapsulation.....	207
framing statement	
channelized OC12 IQ interfaces	
usage guidelines.....	82
channelized OC12 IQE interfaces	
usage guidelines.....	76
channelized OC48 IQE interfaces	
usage guidelines.....	57
E1, E3, and T1 interfaces.....	254

H

hold-time statement	
APS.....	255

I

icons defined, notice.....	xxiv
idle-cycle-flag statement.....	256
interface-type statement.....	257
channelized (COC48/STM16 IQE) interfaces	
usage guidelines.....	49, 50
channelized E1 (COC48/STM16) IQE interfaces	
usage guidelines.....	59
channelized E1 IQE interfaces	
usage guidelines.....	80
channelized E3 (COC48/STM16) IQE interfaces	
usage guidelines.....	58
channelized E3 interfaces	
usage guidelines.....	79
channelized OC3 interfaces	
usage guidelines.....	78
channelized OC3 IQ interfaces	
usage guidelines.....	71, 84
channelized OC48 IQE interfaces	
usage guidelines.....	58
channelized T1 (COC48/STM16 IQE) interfaces	
usage guidelines.....	52
channelized T1 (COC48/STM16, SDH mode)	
IQE interfaces	
usage guidelines.....	61
channelized T1 interfaces	
usage guidelines.....	71
channelized T1 IQ interfaces	
usage guidelines.....	86, 106

channelized T1 IQE interfaces	
usage guidelines.....	85
channelized T3 (COC48/STM16, SDH mode)	
IQE interfaces	
usage guidelines.....	60
channelized T3 IQ and IQE interfaces	
usage guidelines.....	85
E1 (COC48/STM16) IQE interfaces	
usage guidelines.....	59
NxDS0 (COC48/STM16 IQE) interfaces	
usage guidelines.....	54
NxDS0 interfaces	
usage guidelines.....	73
T1 (COC48/STM16, SDH mode) IQE interfaces	
usage guidelines.....	61
T3 (COC48/STM16, SDH mode) IQE interfaces	
usage guidelines.....	60
T3 IQ and IQE interfaces	
usage guidelines.....	85
interfaces	
channelized E1 PRI.....	151
channelized T1 PRI.....	151
clock sources.....	230
configuration statements.....	4, 217
internal clock sources.....	230
inverse-arp statement.....	258
invert-data statement.....	259
IQ interfaces.....	27
channelized.....	27
channelized E1.....	143
channelized OC12.....	65
channelized OC3.....	97
channelized STM4.....	65
channelized T1.....	137
channelized T3.....	123
IQE interfaces.....	27
channelized.....	27
channelized E1.....	143
channelized OC12.....	65
channelized OC48.....	47
channelized OC48/STM16.....	47
channelized STM16.....	47
channelized STM4.....	65
channelized T1.....	137
isdn-options statement.....	260
ITU-T standards	
E1 interfaces.....	161
E3 interfaces.....	169

L

line-encoding statement.....	261
link PIC failover	
channelized OC12 IQ interfaces.....	90
channelized OC48 IQE interfaces.....	62
channelized STM1 IQ and IQE PICs.....	119
lmi statement	
Frame Relay keepalives.....	262
lmi-type statement.....	263
lockout statement.....	264
logical systems	
configuration statements.....	20
long-buildout statement.....	264
loop-timing statement.....	265
loopback capability	
E1 interfaces	
example configuration.....	165
E3 interfaces	
example configuration.....	174
loopback statement	
ADSL, DS0, E1/E3, SONET/SDH, SHDSL, and	
T1/T3.....	266
loopback-clear-timer statement.....	267

M

manuals	
comments on.....	xxvi
mtu statement.....	268
multicast-dlci statement.....	269
multipoint-destination statement.....	270

N

neighbor statement.....	271
no-cbit-parity statement.....	229
no-feac-loop-respond statement.....	252
no-keepalives statement.....	272
no-long-buildout statement.....	264
no-loop-timing statement.....	265
no-partition statement.....	273
channelized E1 IQ interfaces	
usage guidelines.....	143
channelized E1 IQE interfaces	
usage guidelines.....	143
channelized OC12 IQ interfaces	
usage guidelines.....	77, 83
channelized OC12/STM4 interfaces	
usage guidelines.....	69
channelized OC3 IQ interfaces	
usage guidelines.....	99, 152

channelized T3 (COC48/STM16 IQE) interfaces	
usage guidelines.....	51
channelized T3 interfaces	
usage guidelines.....	69
channelized T3 IQ interfaces	
usage guidelines.....	105
no-payload-scrambler statement.....	277
no-termination-request statement.....	274
no-translate-discard-eligible statement.....	287
no-unframed statement.....	287
notice icons defined.....	xxiv
NxDS0 (COC48/STM16) IQE interfaces	
usage guidelines.....	60
NxDS0 interfaces	
usage guidelines.....	81
NxDS0 IQ interfaces	
example configuration.....	109, 126
usage guidelines.....	87

O

oc-slice statement.....	275
usage guidelines.....	51, 69, 77, 83
OC12 interfaces	
example configuration.....	87

P

paired-group statement.....	275
parentheses, in syntax descriptions.....	xxvi
partition statement.....	276
channelized AU-4 (on OC48 IQE) interfaces	
usage guidelines.....	58
channelized AU-4 interfaces	
usage guidelines.....	78, 84
channelized E1 (COC48/STM16) IQE interfaces	
usage guidelines.....	59
channelized E1 IQE interfaces	
usage guidelines.....	80
channelized E3 (COC48/STM16) IQE interfaces	
usage guidelines.....	58
channelized E3 interfaces	
usage guidelines.....	79
channelized OC12 (COC48/STM16 IQE)	
interfaces	
usage guidelines.....	49
channelized OC3 (COC48/STM16 IQE)	
interfaces	
usage guidelines.....	50
channelized OC3 interfaces	
usage guidelines.....	71

channelized T1 (COC48/STM16 IQE) interfaces	
usage guidelines.....	52
channelized T1 (COC48/STM16, SDH mode)	
IQE interfaces	
usage guidelines.....	61
channelized T1 interfaces	
usage guidelines.....	71
channelized T1 IQ interfaces	
usage guidelines.....	86, 106
channelized T1 IQE interfaces	
usage guidelines.....	85
channelized T3 (COC48/STM16, SDH mode)	
IQE interfaces	
usage guidelines.....	60
channelized T3 IQ and IQE interfaces	
usage guidelines.....	85
clear channel STM1, STM4, and STM16	
interface	
usage guide.....	57
E1 (COC48/STM16) IQE interfaces	
usage guidelines.....	59
NxDS0 (COC48/STM16 IQE) interfaces	
usage guidelines.....	54
NxDS0 (COC48/STM16) IQE interfaces	
usage guidelines.....	60
NxDS0 interfaces	
usage guidelines.....	73, 81
NxDS0 IQ interfaces	
usage guidelines.....	87
SDH (VC-4) interfaces	
usage guidelines.....	77, 83
T1 (COC48/STM16, SDH mode) IQE interfaces	
usage guidelines.....	61
T3 (COC48/STM16, SDH mode) IQE interfaces	
usage guidelines.....	60
T3 IQ and IQE interfaces	
usage guidelines.....	85
payload-scrambler statement.....	277
physical interfaces	
byte encoding.....	227
C-bit parity mode.....	229
clock sources.....	230
DCE.....	212, 234
time slots.....	108
fractional E1 IQ and IQE interfaces.....	107
fractional E1 IQ interfaces.....	144
fractional T1 interfaces.....	75
fractional T1 IQE interfaces.....	54
NxDS0 IQ interfaces.....	144
protect-circuit statement.....	278
R	
remote-loopback-respond statement.....	278
request statement.....	279
revert-time statement.....	279
S	
SDH (VC-4) interfaces	
example configuration.....	78, 84
SDH (VC-4-4C) interface	
example configuration.....	77, 83
SONET parameters	
on channelized OC12 interfaces.....	87
sonet-options statement.....	280
usage guidelines.....	87
start-end-flag statement.....	282
support, technical See technical support	
switching-mode statement.....	283
syntax conventions.....	xxv
T	
T1 interfaces	
byte encoding.....	227
configuration statements.....	177
overview.....	177
time slots	
example configuration.....	185
T1 IQ and IQE interfaces	
example configuration.....	86
T1 IQ interfaces	
example configuration.....	124
t1-options statement.....	284
usage guidelines.....	177
T3 interfaces	
C-bit parity mode.....	229
configuration statements.....	188
overview.....	187
T3 IQ and IQE interfaces	
example configuration.....	85
t3-options statement.....	285
usage guidelines.....	188
technical support	
contacting JTAC.....	xxvi
time slots	
channelized E1 IQ and IQE interfaces.....	107
channelized E1 IQE interfaces.....	144
channelized NxDS0 IQ interfaces.....	108, 144

E1 interfaces	
example configuration.....	167
fractional E1 IQ and IQE interfaces.....	107
fractional E1 IQ interfaces.....	144
fractional T1 (COC48/STM16 IQE)	
interfaces.....	54
fractional T1 interfaces.....	75
T1 interfaces	
example configuration.....	185
timeslots statement.....	286
channelized E1 IQ and IQE interfaces	
usage guidelines.....	107
channelized E1 IQ interfaces	
usage guidelines.....	144
channelized E1 IQE interfaces	
usage guidelines.....	144
channelized NxDS0 IQ interfaces	
usage guidelines.....	108, 144
fractional E1 IQ and IQE interfaces	
usage guidelines.....	107
fractional E1 IQ interfaces	
usage guidelines.....	144
fractional E1 IQE interfaces	
usage guidelines.....	144
fractional T1 (COC48/STM16 IQE) interfaces	
usage guidelines.....	54
fractional T1 interfaces	
usage guidelines.....	75
NxDS0 (COC48/STM16 IQE) interfaces	
usage guidelines.....	54
NxDS0 interfaces	
usage guidelines.....	73
translate-discard-eligible statement.....	287

U

unframed statement.....	287
unit statement.....	288

V

VT mapping	
channelized T1 (COC48/STM16 IQE)	
interfaces.....	53
channelized T1 interfaces.....	72, 81
channelized T1 IQ interfaces.....	87, 102
channelized T1 IQE interfaces.....	60
vtmapping statement.....	294
usage guidelines.....	112

W

working-circuit statement.....	294
--------------------------------	-----

Index of Statements and Commands

A

address statement.....	218
advertise-interval statement.....	219
aps statement.....	220
authentication-key statement.....	221

B

bchannel-allocation statement.....	221
bert-algorithm statement.....	222
bert-error-rate statement.....	224
bert-period statement.....	225
buildout statement	
T1 interfaces.....	226
byte-encoding statement.....	227
bytes statement.....	228

C

cbit-parity statement.....	229
clocking statement.....	230
compatibility-mode statement.....	231
crc-minor-alarm-threshold statement.....	232

D

data-input statement.....	233
dce statement.....	234
dlci statement.....	235
ds0-options statement.....	236

E

e1-options statement.....	237
e3-options statement.....	238
encapsulation statement.....	239
logical interface.....	240
physical interface.....	243

F

family statement.....	247
fcs statement.....	251
feac-loop-respond statement.....	252
force statement.....	253

framing statement

E1, E3, and T1 interfaces.....	254
--------------------------------	-----

H

hold-time statement	
APS.....	255

I

idle-cycle-flag statement.....	256
interface-type statement.....	257
inverse-arp statement.....	258
invert-data statement.....	259
isdn-options statement.....	260

L

line-encoding statement.....	261
lmi statement	
Frame Relay keepalives.....	262
lmi-type statement.....	263
lockout statement.....	264
long-buildout statement.....	264
loop-timing statement.....	265
loopback statement	
ADSL, DS0, E1/E3, SONET/SDH, SHDSL, and	
T1/T3.....	266
loopback-clear-timer statement.....	267

M

mtu statement.....	268
multicast-dlci statement.....	269
multipoint-destination statement.....	270

N

neighbor statement.....	271
no-cbit-parity statement.....	229
no-feac-loop-respond statement.....	252
no-keepalives statement.....	272
no-long-buildout statement.....	264
no-loop-timing statement.....	265
no-partition statement.....	273
no-payload-scrambler statement.....	277

no-termination-request statement.....	274
no-translate-discard-eligible statement.....	287
no-unframed statement.....	287

O

oc-slice statement.....	275
-------------------------	-----

P

paired-group statement.....	275
partition statement.....	276
payload-scrambler statement.....	277
protect-circuit statement.....	278

R

remote-loopback-respond statement.....	278
request statement.....	279
revert-time statement.....	279

S

sonet-options statement.....	280
start-end-flag statement.....	282
switching-mode statement.....	283

T

t1-options statement.....	284
t3-options statement.....	285
timeslots statement.....	286
translate-discard-eligible statement.....	287

U

unframed statement.....	287
unit statement.....	288

V

vtmapping statement.....	294
--------------------------	-----

W

working-circuit statement.....	294
--------------------------------	-----