Multilink Frame Relay (FRF.16.1)

First Published: May 14, 2001
Last Updated: May 28, 2008

The Multilink Frame Relay (FRF.16.1) feature introduces functionality based on the Frame Relay Forum Multilink Frame Relay UNI/NNI Implementation Agreement (FRF.16.1). This feature provides a cost-effective way to increase bandwidth for particular applications by enabling multiple serial links to be aggregated into a single bundle of bandwidth. Multilink Frame Relay (MFR) is supported on User-to-Network Interfaces (UNI) and Network-to-Network Interfaces (NNI) in Frame Relay networks.

History for the Multilink Frame Relay (FRF.16.1) Feature

Release	Modification
12.0(17)S	This feature was introduced on the Cisco 12000 series.
12.2(8)T	This feature was integrated into Cisco IOS Release 12.2(8)T.
12.2(14)S	This feature was integrated into Cisco IOS Release 12.2(14)S.
12.3(9)	Frame Relay fragmentation (FRF.12) support was integrated into Cisco IOS Release 12.3(9).
12.3(11)T	Frame Relay fragmentation (FRF.12) support was integrated into Cisco IOS Release 12.3(11)T.
12.0(30)S	Variable bandwidth class support was integrated into Cisco IOS Release 12.0(30)S.
12.4(2)T	Variable bandwidth class support was integrated into Cisco IOS Release 12.4(2)T.
12.2(30)S	Frame Relay fragmentation (FRF.12) support was integrated into Cisco IOS Release 12.2(30)S.
12.2(28)SB	This feature was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This feature was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(31)SB2	Support was added for the Cisco 10000 Series Router.
12.2(33)SXH	This feature was integrated into Cisco IOS Release 12.2(33)SXH.
12.2(33)SB	Support was added for the multilink Frame Relay MIB (RFC 3020) for the Cisco 10000 series router.

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Finding Support Information for Platforms and Cisco IOS and Catalyst OS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst OS software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

Prerequisites for Multilink Frame Relay (FRF.16.1)

- Multilink Frame Relay must be configured on the peer device.

Restrictions for Multilink Frame Relay (FRF.16.1)

- ISDN interfaces and any type of virtual interface cannot be a bundle link.
- Frame Relay fragmentation (FRF.12) is not supported in Cisco IOS releases 12.0(17)S, 12.2(8)T, and 12.2(14)S.
- Cisco IOS Release 12.2(33)SB adds support for the multilink Frame Relay MIB (RFC 3020). All releases prior to Cisco 12.2(33)SB do not support the multilink Frame Relay MIB.
- FRF.9 hardware compression over multilink Frame Relay is not supported.

Information About Multilink Frame Relay (FRF.16.1)

To enable multilink Frame Relay (FRF.16.1) variable bandwidth class support, you should understand the following concepts:

- Benefits of Multilink Frame Relay (FRF.16.1), page 3
- Link Integrity Protocol Control Messages, page 3
- Variable Bandwidth Class Support, page 3
- Load Balancing with Multilink Frame Relay (FRF.16.1), page 4

Benefits of Multilink Frame Relay (FRF.16.1)

Flexible Pool of Bandwidth

By combining multiple physical interfaces into a bundle, you can design a Frame Relay interface that has more bandwidth than is available from any single physical interface. For example, many new network applications require more bandwidth than is available on a T1 line. One option is to invest in a T3 line; however, T3 lines can be expensive and are not available in some locations. Multilink Frame Relay provides a cost-effective solution to this problem by allowing multiple T1 lines to be aggregated into a single bundle of bandwidth.

Greater Service Resilience When Links Fail

Greater service resilience is provided when multiple physical interfaces are provisioned as a single bundle. When a link fails, the bundle continues to support the Frame Relay service by transmitting across the remaining bundle links.

Link Integrity Protocol Control Messages

For link management, each end of a bundle link follows the MFR Link Integrity Protocol and exchanges link-control messages with its peer (the other end of the bundle link). For a bundle link to be brought up, each end of the link must complete an exchange of ADD_LINK and ADD_LINK_ACK messages. To maintain the link, both ends periodically initiate the exchange of HELLO and HELLO_ACK messages. This exchange of hello messages and acknowledgments serves as a keepalive mechanism for the link. If a router is sending hello messages but not receiving acknowledgments, it will resend the hello message up to a configured maximum number of times. If the router exhausts the maximum number of retries, the bundle link line protocol is considered down (nonoperational).

The bundle link interface’s line protocol status is considered up (operational) when the peer device acknowledges that it will use the same link for the bundle. The line protocol remains up when the peer device acknowledges the hello messages from the local router.

The bundle interface’s line protocol status is considered up when the Frame Relay data-link layer at the local router and peer device is synchronized using the Local Management Interface (LMI), when LMI is enabled. The bundle line protocol remains up as long as the LMI keepalives are successful.

Variable Bandwidth Class Support

Multilink Frame Relay (FRF.16.1) variable bandwidth class support allows you to specify the criterion used to activate or deactivate a Frame Relay bundle. Consistent with the Frame Relay Forum Multilink Frame Relay UNI/NNI Implementation Agreement (FRF.16.1), bandwidth classes A (single link), B (all links), and C (threshold) are supported.

Class A (Single Link)

The Frame Relay bundle is provisioned when one or more bundle links indicate by issuing a BL_ACTIVATE message that operational bandwidth is available. When this occurs, the bundle emulates a physical link by issuing a PH_ACTIVATE message to the data-link layer.

When the operational bandwidth of a bundle link fails to meet operational requirements (for instance, if it is in rollback mode), the bundle link issues a BL_DEACTIVATE message. When all bundle links are down in a class A bundle, a PH_DEACTIVATE message is sent to the data-link layer, indicating that the Frame Relay bundle cannot accept frames.

Class B (All Links)

The Frame Relay bundle is provisioned when all bundle links indicate by issuing a BL_ACTIVATE message that operational bandwidth is available. When this occurs, the bundle emulates a physical link by issuing a PH_ACTIVATE message to the data-link layer.

When the operational bandwidth of a bundle link fails to meet operational requirements (for instance, if it is in loopback mode), the bundle link issues a BL_DEACTIVATE message. When any bundle link is down in a class B bundle, a PH_DEACTIVATE message is sent to the data-link layer, indicating that the Frame Relay bundle cannot accept frames.

Class C (Threshold)

The Frame Relay bundle is provisioned when the minimum number of links in the configured bundle issue a BL_ACTIVATE message. When this occurs, the bundle emulates a physical link by issuing a PH_ACTIVATE message to the data-link layer.

When the number of bundle links that are issuing a BL_ACTIVATE message falls below the configured threshold value, a PH_DEACTIVATE message is sent to the data-link layer, indicating that the Frame Relay bundle cannot accept frames.

Load Balancing with Multilink Frame Relay (FRF.16.1)

Multilink Frame Relay provides load balancing across the bundle links within a bundle. If a bundle link chosen for transmission happens to be busy transmitting a long packet, the load-balancing mechanism can try another link, thus solving the problems seen when delay-sensitive packets have to wait.

How to Enable Multilink Frame Relay (FRF.16.1)

This section contains the following procedures:

- Configuring a Multilink Frame Relay Bundle, page 4
- Configuring a Multilink Frame Relay Bundle Link, page 8
- Monitoring and Maintaining Multilink Frame Relay (FRF.16.1), page 10

Configuring a Multilink Frame Relay Bundle

To configure the bundle interface for multilink Frame Relay, perform the steps in this section.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface mfr** *interface-number*
4. **frame-relay multilink bandwidth-class** [**a** | **b** | **c** [*threshold*]]
5. **frame-relay intf-type dce**
6. **frame-relay multilink bid** *name*
7. **frame-relay multilink output-threshold** *bytes*

8. **interface mfr** *interface-number***.***subinterface-number* **point-to-point**
9. **ip address** *ip-address mask*

10. **frame-relay interface-dlci** *dlci*
11. **end**
12. **show frame-relay multilink**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		- Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`interface mfr`** *`interface-number`*
	Example:
	`Router(config)# interface mfr mfr1`
		Configures a multilink Frame Relay bundle interface.

Step 4	**`frame-relay multilink bandwidth-class`** `[`**`a`** `|` **`b`** `|` **`c`** `[`*`threshold`*`]]`
	Example:
	`Router(config-if)# frame-relay multilink bandwidth-class a`
	or
	`Router(config-if)# frame-relay multilink bandwidth-class b`
	or
	`Router(config-if)# frame-relay multilink bandwidth-class c 3`
		(Optional) Specifies the bandwidth class criterion used to activate or deactivate a Frame Relay bundle.
		- Class A (single link)—The bundle will activate when any bundle link is up and will deactivate when all bundle links are down (default).
		- Class B (all links)—The bundle will activate when all bundle links are up and will deactivate when any bundle link is down.
		- Class C (threshold)—The bundle will activate when the minimum configured number of bundle links is up (the threshold) and will deactivate when the minimum number of configured bundle links fails to meet the threshold.
		Note If no bandwidth class criterion is specified by using the **frame-relay multilink bandwidth-class** command, the Frame Relay bundle will default to class A (single link).

	Command or Action	Purpose
Step 5	**frame-relay intf-type dce** **Example:** `Router(config-if)# frame-relay intf-type dce`	Configures a device to function as the data circuit-terminating equipment (DCE). • Only one end of a link should be configured as the DCE. The other end will function as the data terminal equipment (DTE), which is the default setting. • This command can be used only if Frame Relay switching has been enabled by entering the **frame-relay switching** command in global configuration mode.
Step 6	**frame-relay multilink bid** *name* **Example:** `Router(config-if)# frame-relay multilink bid router1`	(Optional) Assigns a bundle identification name to a multilink Frame Relay bundle. • The bundle identification (BID) will not go into effect until the interface has gone from the "down" state to the "up" state. One way to bring the interface down and back up again is by using the **shutdown** and **no shutdown** commands in interface configuration mode.
Step 7	**frame-relay multilink output-threshold** *bytes* **Example:** `Router(config-if)# frame-relay multilink output-threshold 500`	(Optional) Configures the number of bytes that a bundle link will transmit before the load-balancing mechanism causes transmission to roll over to the next available link. • When configured on the bundle interface, this command applies to all bundle links in the bundle.
Step 8	**interface mfr** *interface-number.subinterface-number* **point-to-point** **Example:** `Router(config-if)# interface mfr1.1 point-to-point`	Configures a point-to-point multilink Frame Relay subinterface.
Step 9	**ip address** *ip-address mask* **Example:** `Router(config-subif)# ip address 10.0.1.1 255.255.255.0`	Configures the IP address for the subinterface.
Step 10	**frame-relay interface-dlci** *dlci* **Example:** `Router(config-subif)# frame-relay interface-dlci 100`	Assigns a data-link connection identifier (DLCI) to a Frame Relay subinterface.

	Command or Action	Purpose

Step 11	**end**
	Example:
	`Router(config-subif)# end`
		Ends the configuration session and returns to privileged EXEC mode.

Step 12	**show frame-relay multilink** **Example:** `Router# show frame-relay multilink`	(Optional) Displays the current Frame Relay multilink configuration.

Configuring a Multilink Frame Relay Bundle Link

To configure a bundle link interface for multilink Frame Relay, perform the steps in this section.

Tip To minimize latency that results from the arrival order of packets, we recommend bundling physical links of the same line speed in one bundle.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface serial** *number*
4. **encapsulation frame-relay mfr** *number* [*name*]
5. **frame-relay multilink output-threshold** *bytes*
6. **frame-relay multilink lid** *name*
7. **frame-relay multilink hello** *seconds*
8. **frame-relay multilink ack** *seconds*
9. **frame-relay multilink retry** *number*
10. **end**
11. **show frame-relay multilink**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**interface serial** *number* **Example:** `Router(config)# interface serial 5/0`	Selects a physical interface and enters interface configuration mode.
Step 4	**encapsulation frame-relay mfr** *number* [*name*] **Example:** `Router(config-if)# encapsulation frame-relay mfr1`	Creates a multilink Frame Relay bundle link and associates the link with a bundle.
Step 5	**frame-relay multilink output-threshold** *bytes* **Example:** `Router(config-if)# frame-relay multilink output-threshold 500`	(Optional) Configures the number of bytes that a bundle link will transmit before the load-balancing mechanism causes transmission to roll over to the next available link.
Step 6	**frame-relay multilink lid** *name* **Example:** `Router(config-if)# frame-relay multilink lid first-link`	(Optional) Assigns a bundle link identification name with a multilink Frame Relay bundle link. • The bundle link identification (LID) will not go into effect until the interface has gone from the “down” state to the “up” state. One way to bring the interface down and back up again is by using the **shutdown** and **no shutdown** commands in interface configuration mode.
Step 7	**frame-relay multilink hello** *seconds* **Example:** `Router(config-if)# frame-relay multilink hello 9`	(Optional) Configures the interval at which a bundle link will send out hello messages. • The default value is 10 seconds.
Step 8	**frame-relay multilink ack** *seconds* **Example:** `Router(config-if)# frame-relay multilink ack 6`	(Optional) Configures the number of seconds that a bundle link will wait for a hello message acknowledgment before resending the hello message. • The default value is 4 seconds.
Step 9	**frame-relay multilink retry** *number* **Example:** `Router(config-if)# frame-relay multilink retry 3`	(Optional) Configures the maximum number of times that a bundle link will resend a hello message while waiting for an acknowledgment. • The default value is 2 tries.

Step 10	**end**
	Example:
	`Router(config-if)# end`
		Ends the configuration session and returns to privileged EXEC mode.

Step 11	**show frame-relay multilink** **Example:** `Router# show frame-relay multilink`	(Optional) Displays the current Frame Relay multilink configuration.

Monitoring and Maintaining Multilink Frame Relay (FRF.16.1)

To monitor and maintain multilink Frame Relay, perform the steps in this section.

SUMMARY STEPS

1. **enable**
2. **debug frame-relay multilink** [**control** [**mfr** *number* | **serial** *number*]]
3. **show frame-relay multilink** [**mfr** *number* | **serial** *number*] [**detailed**]
4. **show interfaces mfr** *number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**debug frame-relay multilink** [**control** [**mfr** *number* | **serial** *number*]] **Example:** `Router# debug frame-relay multilink control mfr1`	(Optional) Displays debug messages for multilink Frame Relay bundles and bundle links.
Step 3	**show frame-relay multilink** [**mfr** *number* | **serial** *number*] [**detailed**] **Example:** `Router# show frame-relay multilink mfr1 detailed`	(Optional) Displays configuration information and statistics about multilink Frame Relay bundles and bundle links.
Step 4	**show interfaces mfr** *number* **Example:** `Router# show interfaces mfr1`	(Optional) Displays information and packet statistics for the bundle interface.

Examples

The following example shows output for the **show frame-relay multilink** command. Because a particular bundle or bundle link is not specified, information for all bundles and bundle links is displayed:

```
Router# show frame-relay multilink

Bundle: MFR0, state up, class A, no fragmentation
 ID: Bundle-Dallas
 Serial5/1, state up/up, ID: BL-Dallas-1
 Serial5/3, state up/add-sent, ID: BL-Dallas-3

Bundle: MFR1, state down, class B, fragmentation
 ID: Bundle-NewYork#1
 Serial3/0, state up/up, ID: BL-NewYork-1
 Serial3/2, state admin-down/idle, ID: BL-NewYork-2
```

The following example shows output for the **show frame-relay multilink** command when a Frame Relay bundle is configured as bandwidth class C (threshold):

```
Router# show frame-relay multilink

Bundle: MFR0, state down, class C (threshold 3), no fragmentation
 ID: Bundle-Dallas
 Serial5/1, state up/up, ID: BL-Dallas-1
 Serial5/3, state up/add-sent, ID: BL-Dallas-3
```

The following example shows output for the **show frame-relay multilink** command when the **serial** *number* keyword and argument are specified. It displays information about the specified bundle link:

```
Router# show frame-relay multilink serial 3/2

 Bundle links :
 Serial3/2, HW state :down, Protocol state :Down_idle, LID :Serial3/2
 Bundle interface = MFR0,  BID = MFR0
```

The following examples show output for the **show frame-relay multilink** command when the **serial** *number* keyword and argument and the **detailed** option are specified. Detailed information about the specified bundle links is displayed. The first example shows a bundle link in the “idle” state. The second example shows a bundle link in the “up” state:

```
Router# show frame-relay multilink serial 3 detail

Bundle links:

  Serial3, HW state = up, link state = Idle, LID = Serial3
  Bundle interface = MFR0,  BID = MFR0
    Cause code = none, Ack timer = 4, Hello timer = 10,
    Max retry count = 2, Current count = 0,
    Peer LID = Serial5/3, RTT = 0 ms
    Statistics:
    Add_link sent = 0, Add_link rcv'd = 10,
    Add_link ack sent = 0, Add_link ack rcv'd = 0,
    Add_link rej sent = 10, Add_link rej rcv'd = 0,
    Remove_link sent = 0, Remove_link rcv'd = 0,
    Remove_link_ack sent = 0, Remove_link_ack rcv'd = 0,
    Hello sent = 0, Hello rcv'd = 0,
    Hello_ack sent = 0, Hello_ack rcv'd = 0,
    outgoing pak dropped = 0, incoming pak dropped = 0

Router# show frame-relay multilink serial 3 detail

Bundle links:

  Serial3, HW state = up, link state = Up, LID = Serial3
  Bundle interface = MFR0,  BID = MFR0
    Cause code = none, Ack timer = 4, Hello timer = 10,
    Max retry count = 2, Current count = 0,
    Peer LID = Serial5/3, RTT = 4 ms
    Statistics:
    Add_link sent = 1, Add_link rcv'd = 20,
    Add_link ack sent = 1, Add_link ack rcv'd = 1,
    Add_link rej sent = 19, Add_link rej rcv'd = 0,
    Remove_link sent = 0, Remove_link rcv'd = 0,
    Remove_link_ack sent = 0, Remove_link_ack rcv'd = 0,
    Hello sent = 0, Hello rcv'd = 1,
    Hello_ack sent = 1, Hello_ack rcv'd = 0,
    outgoing pak dropped = 0, incoming pak dropped = 0
```

Configuration Examples for Multilink Frame Relay (FRF.16.1)

This section provides the following configuration examples:

- Configuring Multilink Frame Relay: Example, page 12
- Configuring Variable Bandwidth Class Support: Example, page 12

Configuring Multilink Frame Relay: Example

The following example shows the configuration of bundle "MFR1." Serial interfaces 5/0 and 6/0 are configured as bundle links:

```
interface MFR1
 no ip address
 mls qos trust dscp
 frame-relay intf-type dce
 frame-relay multilink bid router1
!
interface MFR1.1 point-to-point
 ip address 10.0.1.1 255.255.255.0
 ip pim sparse-mode
 mls qos trust dscp
 frame-relay interface-dlci 100

interface Serial5/0
 encapsulation frame-relay MFR1
 frame-relay multilink lid first-link
 frame-relay multilink hello 9
 frame-relay multilink retry 3

interface Serial6/0
 encapsulation frame-relay MFR1
 frame-relay multilink ack 4
```

Configuring Variable Bandwidth Class Support: Example

The following example configures the Frame Relay bundle "MFR1" to use the class B (all links) criterion to be activated or deactivated:

```
interface MFR1
 ip address 10.1.1.1 255.255.255.0
 frame-relay interface-dlci 100
 frame-relay multilink bandwidth-class b
```

Additional References

The following sections provide references related to multilink Frame Relay (FRF.16.1).

Related Documents

Related Topic	Document Title
Frame Relay configuration	*Cisco IOS Wide-Area Networking Configuration Guide,* Release 12.4T
Frame Relay commands	*Cisco IOS Wide-Area Networking Command Reference,* Release 12.4T

Standards

Standard	Title
FRF.16.1	*Multilink Frame Relay UNI/NNI Implementation Agreement,* May 2002

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
None	—

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

Command Reference

This section documents only commands that are new or modified.

- **debug frame-relay multilink**
- **encapsulation frame-relay mfr**
- **frame-relay multilink ack**
- **frame-relay multilink bandwidth-class**
- **frame-relay multilink bid**
- **frame-relay multilink hello**
- **frame-relay multilink lid**
- **frame-relay multilink output-threshold**
- **frame-relay multilink retry**
- **interface mfr**
- **show frame-relay multilink**

debug frame-relay multilink

To display debug messages for multilink Frame Relay bundles and bundle links, use the **debug frame-relay multilink** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug frame-relay multilink [**control** [**mfr** *number* | **serial** *number*]]

no debug frame-relay multilink

Syntax Description

control	(Optional) Displays incoming and outgoing bundle link control messages and bundle link status changes.
mfr *number*	(Optional) Displays information for a specific bundle interface.
serial *number*	(Optional) Displays information for a specific bundle link interface.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.0(17)S	This command was introduced.
12.0(24)S	This command was introduced on VIP-enabled Cisco 7500 series routers.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(17b)SXA	This command was integrated into Cisco IOS Release 12.2(17b)SXA.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

Caution Using the **debug frame-relay multilink** command without the **control** keyword could severely impact router performance and is not recommended.

Using the **debug frame-relay multilink** command without the **mfr** or **serial** keywords displays error conditions that occur at the bundle layer.

Examples

The following example shows output from the **debug frame-relay multilink** command for bundle “MFR0,” which has three bundle links:

```
Router# debug frame-relay multilink control MFR0

00:42:54:Serial5/3(o):msg=Add_link, Link=Serial5/3, Bundle=MFR0, Link id=Serial5/3,
BL state=Idle
E1 00 01 01 07 4D 46 52 30 00
```

```
00:42:54:Serial5/2(o):msg=Add_link, Link=Serial5/2, Bundle=MFR0, Link id=Serial5/2,
BL state=Idle
E1 00 01 01 07 4D 46 52 30 00
00:42:54:Serial5/1(o):msg=Add_link, Link=Serial5/1, Bundle=MFR0, Link id=Serial5/1,
BL state=Idle
E1 00 01 01 07 4D 46 52 30 00
00:42:54:%LINK-3-UPDOWN:Interface MFR0, changed state to down
00:42:54:Serial5/3(i):msg=Add_link_ack, Link=Serial5/3, Bundle=MFR0, Link id=Serial5/3,
BL state=Add_sent
E1 00 02 01 07 4D 46 52 30 00
00:42:54:Serial5/2(i):msg=Add_link_ack, Link=Serial5/2, Bundle=MFR0, Link id=Serial5/2,
BL state=Add_sent
E1 00 02 01 07 4D 46 52 30 00
00:42:54:Serial5/1(i):msg=Add_link_ack, Link=Serial5/1, Bundle=MFR0, Link id=Serial5/1,
BL state=Add_sent
E1 00 02 01 07 4D 46 52 30 00
00:42:54:%SYS-5-CONFIG_I:Configured from console by console
00:43:00:Serial5/1(i):msg=Add_link, Link=Serial5/1, Bundle=MFR0, Link id=Serial5/1,
BL state=Ack_rx
E1 00 01 01 07 4D 46 52 30 00
00:43:00:Serial5/1(o):msg=Add_link_ack, Link=Serial5/1, Bundle=MFR0, Link id=Serial5/1,
BL state=Ack_rx
E1 00 02 01 07 4D 46 52 30 00
00:43:00:%LINK-3-UPDOWN:Interface MFR0, changed state to up
00:43:00:Serial5/1(i):msg=Hello, Link=Serial5/1, Bundle=MFR0, Linkid=Serial5/1, BL
state=Up
E1 00 04 03 06 30 A7 E0 54 00
00:43:00:Serial5/1(o):msg=Hello_ack, Link=Serial5/1, Bundle=MFR0, Link id=Serial5/1, BL
state=Up
E1 00 05 03 06 90 E7 0F C2 06
00:43:01:Serial5/2(i):msg=Add_link, Link=Serial5/2, Bundle=MFR0, Link id=Serial5/2,
BL state=Ack_rx
E1 00 01 01 07 4D 46 52 30 00
00:43:01:Serial5/2(o):msg=Add_link_ack, Link=Serial5/2, Bundle=MFR0, Link id=Serial5/2,
BL state=Ack_rx
E1 00 02 01 07 4D 46 52 30 00
00:43:01:Serial5/2(i):msg=Hello, Link=Serial5/2, Bundle=MFR0, Linkid=Serial5/2, BL
state=Up
E1 00 04 03 06 30 A7 E0 54 00
00:43:01:Serial5/2(o):msg=Hello_ack, Link=Serial5/2, Bundle=MFR0, Link id=Serial5/2,
BL state=Up
E1 00 05 03 06 90 E7 0F C2 06
00:43:01:%LINEPROTO-5-UPDOWN:Line protocol on Interface Serial5/1, changed state to up
00:43:01:Serial5/3(i):msg=Add_link, Link=Serial5/3, Bundle=MFR0, Link id=Serial5/3,
BL state=Ack_rx
E1 00 01 01 07 4D 46 52 30 00
00:43:01:Serial5/3(o):msg=Add_link_ack, Link=Serial5/3, Bundle=MFR0, Link id=Serial5/3,
BL state=Ack_rx
E1 00 02 01 07 4D 46 52 30 00
00:43:01:Serial5/3(i):msg=Hello, Link=Serial5/3, Bundle=MFR0, Linkid=Serial5/3, BL
state=Up
E1 00 04 03 06 30 A7 E0 54 00
00:43:01:Serial5/3(o):msg=Hello_ack, Link=Serial5/3, Bundle=MFR0, Link id=Serial5/3,
BL state=Up
E1 00 05 03 06 90 E7 0F C2 06
00:43:02:%LINEPROTO-5-UPDOWN:Line protocol on Interface Serial5/2 , changed state to up
00:43:02:%LINEPROTO-5-UPDOWN:Line protocol on Interface Serial5/3 , changed state to up
```

Table 1 describes the significant fields shown in the display.

Table 1 debug frame-relay multilink Field Descriptions

Field	Description
msg	Type of bundle link control message that was sent or received.
Link	Interface number of the bundle link.
Bundle	Bundle with which the link is associated.
Link id	Bundle link identification name.
BL state	Operational state of the bundle link.

Related Commands

Command	Description
show frame-relay multilink	Displays configuration information and statistics about multilink Frame Relay bundles and bundle links.

encapsulation frame-relay mfr

To create a multilink Frame Relay bundle link and to associate the link with a bundle, use the **encapsulation frame-relay mfr** command in interface configuration mode. To remove the bundle link from the bundle, use the **no** form of this command.

encapsulation frame-relay mfr *number* [*name*]

no encapsulation frame-relay mfr *number* [*name*]

Syntax Description

number	Interface number of the multilink Frame Relay bundle with which this bundle link will be associated.
name	(Optional) Bundle link identification (LID) name. The name can be up to 49 characters long. The default is the name of the physical interface.

Command Default

Frame Relay encapsulation is not enabled.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.0(17)S	This command was introduced on the Cisco 12000 series routers.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.0(24)S	This command was implemented on VIP-enabled Cisco 7500 series routers.
12.3(4)T	Support for this command on VIP-enabled Cisco 7500 series routers was integrated into Cisco IOS Release 12.3(4)T.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

Use the *name* argument to assign a LID name to a bundle link. This name will be used to identify the bundle link to peer devices and to enable the devices to determine which bundle links are associated with which bundles. The LID name can also be assigned or changed by using the **frame-relay multilink lid** command on the bundle link interface. If the LID name is not assigned, the default name is the name of the physical interface.

Tips To minimize latency that results from the arrival order of packets, we recommend bundling physical links of the same line speed in one bundle.

To remove a bundle link from a bundle, use the **no encapsulation frame-relay mfr** command or configure a new type of encapsulation on the interface by using the **encapsulation** command.

Examples

The following example shows serial interface 0 being associated as a bundle link with bundle interface "mfr0." The bundle link identification name is "BL1."

```
interface mfr0
!
interface serial 0
 encapsulation frame-relay mfr0 BL1
```

Related Commands

Command	Description
debug frame-relay multilink	Displays debug messages for multilink Frame Relay bundles and bundle links.
encapsulation	Sets the encapsulation method used by the interface.
frame-relay multilink lid	Assigns a LID name to a multilink Frame Relay bundle link.
show frame-relay multilink	Displays configuration information and statistics about multilink Frame Relay bundles and bundle links.

frame-relay multilink ack

To configure the number of seconds for which a bundle link will wait for a hello message acknowledgment before resending the hello message, use the **frame-relay multilink ack** command in interface configuration mode. To reset this parameter to the default setting, use the **no** form of this command.

frame-relay multilink ack *seconds*

no frame-relay multilink ack

Syntax Description

seconds	Number of seconds for which a bundle link will wait for a hello message acknowledgment before resending the hello message. Range: 1 to 10. Default: 4.

Command Default

The acknowledgment interval is 4 seconds.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.0(17)S	This command was introduced.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.0(24)S	This command was implemented on VIP-enabled Cisco 7500 series routers.
12.3(4)T	Support for this command on VIP-enabled Cisco 7500 series routers was integrated into Cisco IOS Release 12.3(4)T.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

The **frame-relay multilink ack** command can be configured only on bundle link interfaces that have been associated with a bundle using the **encapsulation frame-relay mfr** command.

Both ends of a bundle link send out hello messages at regular intervals. When a peer device receives a hello message, it responds by sending an acknowledgment. This exchange of hello messages and acknowledgments serves as a keepalive mechanism for the link. If the bundle link sends a hello message but does not receive an acknowledgment, it will resend the hello message up to a configured maximum number of times. If the bundle link exhausts the maximum number of retries, the bundle link line protocol is considered down (nonoperational).

The **frame-relay multilink ack** command setting on the local router is independent of the setting on the peer device.

Examples

The following example shows how to configure the bundle link to wait 6 seconds before resending hello messages:

```
interface serial0
 encapsulation frame-relay mfr0
 frame-relay multilink ack 6
```

Related Commands

Command	Description
encapsulation frame-relay mfr	Creates a multilink Frame Relay bundle link and associates the link with a bundle.
frame-relay multilink bandwidth-class	Specifies the bandwidth class used to trigger activation or deactivation of the Frame Relay bundle.
frame-relay multilink hello	Configures the interval at which a bundle link will send out hello messages.
frame-relay multilink retry	Configures the maximum number of times that a bundle link will resend a hello message while waiting for an acknowledgment.

frame-relay multilink bandwidth-class

To specify the criterion used to activate or deactivate a Frame Relay bundle, use the **frame-relay multilink bandwidth-class** command in interface configuration mode. To reset the bandwidth class to the default, use the **no** form of this command.

frame-relay multilink bandwidth-class [**a** | **b** | **c** [*threshold*]]

no frame-relay multilink bandwidth-class

Syntax Description

a	(Optional) Bandwidth class A (single link) criterion will be used to activate or deactivate the Frame Relay bundle. This is the default. • Criterion for activation—One or more bundle links indicate (by issuing a BL_ACTIVATE message) that operational bandwidth is available. When this occurs, the bundle emulates a physical link by issuing a PH_ACTIVATE message to the data-link layer. • Criterion for deactivation—All bundle links are down and issue a BL_DEACTIVATE message, which triggers a PH_DEACTIVATE message to be sent to the data-link layer, indicating that the Frame Relay bundle cannot accept frames.
b	(Optional) Bandwidth class B (all links) criterion will be used to activate or deactivate the Frame Relay bundle. • Criterion for activation—All bundle links indicate (by issuing a BL_ACTIVATE message) that operational bandwidth is available. When this occurs, the bundle emulates a physical link by issuing a PH_ACTIVATE message to the data-link layer. • Criterion for deactivation—Any bundle link is down and issues a BL_DEACTIVATE message, which triggers a PH_DEACTIVATE message to be sent to the data-link layer, indicating that the Frame Relay bundle cannot accept frames.
c	(Optional) Bandwidth class C (threshold) criterion will be used to activate or deactivate the Frame Relay bundle. • Criterion for activation—The minimum number of links in the configured bundle issue a BL_ACTIVATE message. When this occurs, the bundle emulates a physical link by issuing a PH_ACTIVATE message to the data-link layer. • Criterion for deactivation—The number of bundle links issuing a BL_ACTIVATE message falls below the configured *threshold* value. When this occurs, a PH_DEACTIVATE message is sent to the data-link layer, which indicates that the Frame Relay bundle cannot accept frames.
threshold	(Optional) Number of bundle links. The range is from 1 to 65535. If the *threshold* argument is not specified, the default value is 1.

Command Default

Frame Relay bundles use bandwidth class A (single link).

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.0(30)S	This command was introduced.
12.4(2)T	This command was integrated into Cisco IOS Release 12.4(2)T.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

The **frame-relay multilink bandwidth-class** command can be configured only on a bundle’s main interface. If no bandwidth class is specified by using the **frame-relay multilink bandwidth-class** command, the Frame Relay bundle uses the class A (single link) criterion.

Examples

The following example shows how to specify the class B (all links) bandwidth class to trigger activation or deactivation of the Frame Relay bundle on MFR interface 0:

```
interface mfr0
 frame-relay multilink bandwidth-class b
```

The following example shows how to specify the class C (threshold) bandwidth class to trigger activation or deactivation of the Frame Relay bundle on MFR interface 0, where the minimum threshold of links indicating BL_ACTIVATE is 3:

```
interface mfr0
 frame-relay multilink bandwidth-class c 3
```

Related Commands

Command	Description
interface mfr	Configures a multilink Frame Relay bundle interface.
show frame-relay multilink	Displays configuration information and statistics about multilink Frame Relay bundles and bundle links.

frame-relay multilink bid

To assign a bundle identification (BID) name to a multilink Frame Relay bundle, use the **frame-relay multilink bid** command in interface configuration mode. To reset the name to the default, use the **no** form of this command.

frame-relay multilink bid *name*

no frame-relay multilink bid

Syntax Description

name	Bundle identification (BID) name. The name can be up to 49 characters long. The default is "mfr" followed by the number assigned to the bundle using the **interface mfr** command; for example, "mfr0."

Command Default

The BID name is assigned automatically as "mfr" followed by the number assigned to the bundle.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.0(17)S	This command was introduced.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.0(24)S	This command was implemented on VIP-enabled Cisco 7500 series routers.
12.3(4)T	Support for this command on VIP-enabled Cisco 7500 series routers was integrated into Cisco IOS Release 12.3(4)T.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

This command can be entered only on the multilink Frame Relay bundle interface.

Note You can enter the **frame-relay multilink bid** command at any time without affecting the current state of the interface; however, the BID will not go into effect until the interface has gone from the down state to the up state. One way to bring the interface down and back up again is by using the **shutdown** and **no shutdown** commands in interface configuration mode.

Only one BID is allowed per bundle. A later entry of the **frame-relay multilink bid** command supersedes prior entries.

The local and peer BIDs do not have to be unique.

Examples

The following example shows how to assign a BID of “bundle1” to the multilink Frame Relay bundle. The previous BID for the bundle was “mfr0.”

```
interface mfr0
 frame-relay multilink bid bundle1
```

Related Commands

Command	Description
frame-relay multilink lid	Assigns a LID name to a multilink Frame Relay bundle link.
interface mfr	Configures a multilink Frame Relay bundle interface.
show frame-relay multilink	Displays configuration information and statistics about multilink Frame Relay bundles and bundle links.
shutdown (interface)	Disables an interface.

frame-relay multilink hello

To configure the interval at which a bundle link will send out hello messages, use the **frame-relay multilink hello** command in interface configuration mode. To reset this value to the default setting, use the **no** form of this command.

frame-relay multilink hello *seconds*

no frame-relay multilink hello

Syntax Description

seconds	Interval, in seconds, at which a bundle link will send out hello messages. Range: 1 to 180. Default: 10.

Command Default

The interval is set at 10 seconds.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.0(17)S	This command was introduced.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.0(24)S	This command was implemented on VIP-enabled Cisco 7500 series routers.
12.3(4)T	Support for this command on VIP-enabled Cisco 7500 series routers was integrated into Cisco IOS Release 12.3(4)T.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

The **frame-relay multilink hello** command can be configured only on bundle link interfaces that have been associated with a bundle using the **encapsulation frame-relay mfr** command.

Both ends of a bundle link send out hello messages at regular intervals. When a peer device receives a hello message, it responds by sending an acknowledgment. This exchange of hello messages and acknowledgments serves as a keepalive mechanism for the link. If the bundle link sends a hello message but does not receive an acknowledgment, it will resend the hello message up to a configured maximum number of times. If the bundle link exhausts the maximum number of retries, the bundle link line protocol is considered down (nonoperational).

The setting of the hello message interval on the local router is independent of the setting on the peer device.

Examples

The following example shows how to configure a bundle link to send hello messages every 15 seconds:

```
interface serial0
 encapsulation frame-relay mfr0
 frame-relay multilink hello 15
```

Related Commands

Command	Description
encapsulation frame-relay mfr	Creates a multilink Frame Relay bundle link and associates the link with a bundle.
frame-relay multilink ack	Configures the number of seconds that a bundle link will wait for a hello message acknowledgment before resending the hello message.
frame-relay multilink retry	Configures the maximum number of times that a bundle link will resend a hello message while waiting for an acknowledgment.

frame-relay multilink lid

To assign a bundle link identification (LID) name to a multilink Frame Relay bundle link, use the **frame-relay multilink lid** command in interface configuration mode. To reset the name to the default, use the **no** form of this command.

frame-relay multilink lid *name*

no frame-relay multilink lid

Syntax Description

name	Bundle link identification (LID) name. The name can be up to 49 characters long. The default is the name of the physical interface.

Command Default

The name of the physical interface is used as the LID.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.0(17)S	This command was introduced.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.0(24)S	This command was implemented on VIP-enabled Cisco 7500 series routers.
12.3(4)T	Support for this command on VIP-enabled Cisco 7500 series routers was integrated into Cisco IOS Release 12.3(4)T.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

The **frame-relay multilink lid** command can be configured only on bundle link interfaces that have been associated with a bundle using the **encapsulation frame-relay mfr** command.

Note You can enter the **frame-relay multilink lid** command at any time without affecting the current state of the interface; however, the LID will not go into effect until the interface has gone from the down state to the up state. One way to bring the interface down and back up again is by using the **shutdown** and **no shutdown** commands in interface configuration mode.

The LID will be used to identify the bundle link to peer devices and to enable the devices to identify which bundle links are associated with which bundles. The LID can also be assigned when the bundle link is created by using the **encapsulation frame-relay mfr** command with the *name* argument. If the LID is not assigned, the default LID is the name of the physical interface.

The local and peer LIDs do not have to be unique.

Examples

The following example shows the LID “BL1” assigned to serial interface 0:

```
interface serial 0
 encapsulation frame-relay mfr0
 frame-relay multilink lid BL1
```

Related Commands

Command	Description
encapsulation frame-relay mfr	Creates a multilink Frame Relay bundle link and associates the link with a bundle.
frame-relay multilink bid	Assigns a BID name to a multilink Frame Relay bundle.
show frame-relay multilink	Displays configuration information and statistics about multilink Frame Relay bundles and bundle links.
shutdown (interface)	Disables an interface.

frame-relay multilink output-threshold

To configure the number of bytes that a bundle link will transmit before the load-balancing mechanism causes transmission to roll over to the next available link, use the **frame-relay multilink output-threshold** command in interface configuration mode. To reset this value to the default setting, use the **no** form of this command.

frame-relay multilink output-threshold *bytes*

no frame-relay multilink output-threshold

Syntax Description

bytes	Number of bytes that a bundle link will transmit before the load-balancing mechanism causes transmission to roll over to the next link. Range: 20 to 2147483647. Default: 300.

Command Default

The number of bytes transmitted is set at 300.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.2(8)T	This command was introduced.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.0(30)S	This command was integrated into Cisco IOS Release 12.0(30)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

Multilink Frame Relay enables load balancing across bundle links that are in the same bundle. When a bundle link has reached its output threshold, transmission rolls over to the next available bundle link in the bundle.

The output threshold mechanism applies only when the bundle interface is using FIFO output queueing. When the bundle interface is not using FIFO output queuing, the algorithm for choosing a bundle link interface for output selects the bundle link that has the empty or shortest output queue.

The default output threshold is 300 bytes. This default value will work effectively if all the bundle links in the bundle have the same speed. To efficiently use bundle links with different speeds, use the **frame-relay multilink output-threshold** command to adjust the output threshold of the links as appropriate.

The **frame-relay multilink output-threshold** command can be used on the bundle interface and the bundle links. If the command is used on the bundle interface, the configured output threshold will apply to all bundle links in the bundle. If the command is used on a specific bundle link, the output threshold will overwrite the current setting for that bundle link.

Examples

The following example shows how to configure the bundle link output threshold at 600 bytes. When the bundle link reaches the threshold, transmission will roll over to the next link.

```
interface serial0
 encapsulation frame-relay mfr0
 frame-relay multilink output-threshold 600
```

Related Commands

Command	Description
encapsulation frame-relay mfr	Creates a multilink Frame Relay bundle link and associates the link with a bundle.
frame-relay multilink bandwidth-class	Specifies the bandwidth class used to trigger activation or deactivation of the Frame Relay bundle.

frame-relay multilink retry

To configure the maximum number of times that a bundle link will resend a hello message while waiting for an acknowledgment, use the **frame-relay multilink retry** command in interface configuration mode. To reset this value to the default setting, use the **no** form of this command.

frame-relay multilink retry *number*

no frame-relay multilink retry

Syntax Description

number	Maximum number of times that a bundle link will resend a hello message while waiting for an acknowledgment. Range: 1 through 5. Default: 2.

Command Default

The number of retries is set at 2.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.0(17)S	This command was introduced.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.0(24)S	This command was implemented on VIP-enabled Cisco 7500 series routers.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.3(4)T	Support for this command on VIP-enabled Cisco 7500 series routers was integrated into Cisco IOS Release 12.3(4)T.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

The **frame-relay multilink retry** command can be configured only on bundle link interfaces that have been associated with a bundle using the **encapsulation frame-relay mfr** command.

If the bundle link sends the maximum number of hello messages without receiving an acknowledgment, the bundle link line protocol is considered down (nonoperational).

The maximum number of retries configured on the local router is independent of the maximum number configured on the peer device.

Examples

The following example shows how to configure a bundle link to send a hello message a maximum of 3 times while waiting for an acknowledgment:

```
interface serial0
 encapsulation frame-relay mfr0
 frame-relay multilink retry 3
```

Related Commands

Command	Description
encapsulation frame-relay mfr	Creates a multilink Frame Relay bundle link and associates the link with a bundle.
frame-relay multilink ack	Configures the number of seconds that a bundle link will wait for a hello message acknowledgment before resending the hello message.
frame-relay multilink hello	Configures the interval at which a bundle link will send out hello messages.

interface mfr

To configure a multilink Frame Relay bundle interface, use the **interface mfr** command in global configuration mode. To remove the bundle interface, use the **no** form of this command.

interface mfr *number*

no interface mfr *number*

Syntax Description

number	Number that will uniquely identify this bundle interface. Range: 0 to 2147483647.

Command Default

A Frame Relay bundle interface is not configured.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.0(17)S	This command was introduced.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.0(24)S	This command was introduced on VIP-enabled Cisco 7500 series routers.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.3(4)T	Support for this command on VIP-enabled Cisco 7500 series routers was integrated into Cisco IOS Release 12.3(4)T.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

Frame Relay encapsulation is the default encapsulation type for multilink Frame Relay bundle interfaces.

A bundle interface is a virtual interface that serves as the Frame Relay data link and performs the same functions as a physical interface. The bundle is made up of physical serial links, called bundle links. The bundle links within a bundle function as one physical link and one pool of bandwidth. Functionality that you want to apply to the bundle links must be configured on the bundle interface.

The **no interface mfr** command will work only if all bundle links have been removed from the bundle by using the **no encapsulation frame-relay mfr** command.

Examples

The following example shows the configuration of a bundle interface called “mfr0.” The bundle identification (BID) name “BUNDLE-A” is assigned to the bundle. Serial interfaces 0 and 1 are assigned to the bundle as bundle links.

```
interface mfr0
 frame-relay multilink bid BUNDLE-A
!
interface serial0
 encapsulation frame-relay mfr0
!
interface serial1
 encapsulation frame-relay mfr0
```

Related Commands

Command	Description
debug frame-relay multilink	Displays debug messages for multilink Frame Relay bundles and bundle links.
encapsulation frame-relay mfr	Creates a multilink Frame Relay bundle link and associates the link with a bundle.
frame-relay multilink bandwidth-class	Specifies the bandwidth class used to trigger activation or deactivation of the Frame Relay bundle.
frame-relay multilink bid	Assigns a BID name to a multilink Frame Relay bundle.
show frame-relay multilink	Displays configuration information and statistics about multilink Frame Relay bundles and bundle links.

show frame-relay multilink

To display configuration information and statistics about multilink Frame Relay bundles and bundle links, use the **show frame-relay multilink** command in privileged EXEC mode.

show frame-relay multilink [**mfr** *number* | **serial** *number*] [**detailed**]

Syntax Description

mfr *number*	(Optional) Specific bundle interface for which information will be displayed.
serial *number*	(Optional) Specific bundle link interface for which information will be displayed.
detailed	(Optional) Display replace information, including counters for the control messages sent to and from the peer device and the status of the bundle links.

Command Default

Information for all bundles and bundle links is displayed.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.0(17)S	This command was introduced.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.0(24)S	This command was implemented on VIP-enabled Cisco 7500 series routers.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.3(4)T	This command was implemented on VIP-enabled Cisco 7500 series routers.
12.0(30)S	This command was updated to display Multilink Frame Relay variable bandwidth class status.
12.4(2)T	This command was updated to display Multilink Frame Relay variable bandwidth class status.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

If you do not specify a bundle or bundle link, information for all bundles and bundle links is displayed.

Examples

All Bundles and Bundle Links: Example

The following example shows output for the **show frame-relay multilink** command (see Table 2 for descriptions of the fields). Because a specific bundle or bundle link is not specified, information for all bundles and bundle links is displayed:

```
Router# show frame-relay multilink
```

```
Bundle:MFR0, State = up, class = A, fragmentation disabled
 BID = MFR0
 Bundle links :
 Serial2/1:3, HW state :up, Protocol state :Idle, LID :Serial2/1:3
 Serial2/1:2, HW state :up, Protocol state :Idle, LID :Serial2/1:2
 Serial2/1:1, HW state :up, Protocol state :Idle, LID :Serial2/1:1
```

The following example shows output for the **show frame-relay multilink** command when a Frame Relay bundle is configured as bandwidth class C (threshold) (see Table 2 for descriptions of the fields):

```
Router# show frame-relay multilink

Bundle: MFR0, state down, class C (threshold 2), no fragmentation
 ID: Bundle-Dallas
 Serial5/1, state up/up, ID: BL-Dallas-1
 Serial5/3, state up/add-sent, ID: BL-Dallas-3
```

Bundle Link: Example

The following example shows output for the **show frame-relay multilink** command when it is issued with the **serial** *number* option (see Table 2 for descriptions of the fields). It displays information about the specified bundle link:

```
Router# show frame-relay multilink serial 3/2

 Bundle links :
 Serial3/2, HW state : down, Protocol state :Down_idle, LID :Serial3/2
 Bundle interface = MFR0,  BID = MFR0
```

Detailed Bundle Links: Examples

The following example shows output for the **show frame-relay multilink** command when it is issued with the **serial** *number* and **detailed** options (see Table 2 for descriptions of the fields). The example shows a bundle link in the "idle" state:

```
Router# show frame-relay multilink serial 3 detailed

 Bundle links:

  Serial3, HW state = up, link state = Idle, LID = Serial3
  Bundle interface = MFR0,  BID = MFR0
    Cause code = none, Ack timer = 4, Hello timer = 10,
    Max retry count = 2, Current count = 0,
    Peer LID = Serial5/3, RTT = 0 ms
    Statistics:
    Add_link sent = 0, Add_link rcv'd = 10,
    Add_link ack sent = 0, Add_link ack rcv'd = 0,
    Add_link rej sent = 10, Add_link rej rcv'd = 0,
    Remove_link sent = 0, Remove_link rcv'd = 0,
    Remove_link_ack sent = 0, Remove_link_ack rcv'd = 0,
    Hello sent = 0, Hello rcv'd = 0,
    Hello_ack sent = 0, Hello_ack rcv'd = 0,
    outgoing pak dropped = 0, incoming pak dropped = 0
```

The following example shows output for the **show frame-relay multilink** command when it is issued with the **serial** *number* and **detailed** options (see Table 2 for descriptions of the fields). The example shows a bundle link in the "up" state:

```
Router# show frame-relay multilink serial 3 detailed

 Bundle links:

  Serial3, HW state = up, link state = Up, LID = Serial3
  Bundle interface = MFR0,  BID = MFR0
```

```
    Cause code = none, Ack timer = 4, Hello timer = 10,
    Max retry count = 2, Current count = 0,
    Peer LID = Serial5/3, RTT = 4 ms
    Statistics:
    Add_link sent = 1, Add_link rcv'd = 20,
    Add_link ack sent = 1, Add_link ack rcv'd = 1,
    Add_link rej sent = 19, Add_link rej rcv'd = 0,
    Remove_link sent = 0, Remove_link rcv'd = 0,
    Remove_link_ack sent = 0, Remove_link_ack rcv'd = 0,
    Hello sent = 0, Hello rcv'd = 1,
    Hello_ack sent = 1, Hello_ack rcv'd = 0,
    outgoing pak dropped = 0, incoming pak dropped = 0
```

Table 2 describes significant fields shown in the displays.

Table 2 show frame-relay multilink Field Descriptions

Field	Description
Bundle	Bundle interface.
State	Operational state of the bundle interface.
class	The bandwidth class criterion used to activate or deactivate a Frame Relay bundle. • Class A (single link)—The bundle activates when any bundle link is up and deactivates when all bundle links are down (default). • Class B (all links)—The bundle activates when all bundle links are up and deactivates when any bundle link is down. • Class C (threshold)—The bundle activates when the minimum configured number of bundle links is up (the threshold) and deactivates when the minimum number of configured bundle links fails to meet the threshold.
BID	Bundle identification.
Bundle links	Bundle links for which information will be displayed.
HW state	Operational state of the physical link.
Protocol state	Operational state of the bundle link line protocol.
link state	Operational state of the bundle link.
LID	Bundle link identification.
Bundle interface	Bundle interface with which the bundle link is associated.

Table 2 show frame-relay multilink Field Descriptions (continued)

Field	Description
Cause code	May be one of the following values: • ack timer expiry—Add link synchronization process is exhausted. • bundle link idle—Peer's bundle link is idle. This usually occurs when the peer's bundle interface is shut down. • inconsistent bundle—Peer already has this bundle associated with another bundle. • loopback detected—Local bundle link's physical line is looped back. • none—ADD_LINK and ADD_LINK_ACK messages were properly exchanged, and no cause code was recorded. • other—Indicates one of the following: a link identifier (LID) mismatch, an ID from the peer that is too long, or a failure to allocate ID memory. • unexpected Add_link—ADD_LINK message is received when the bundle link is already in the "up" state. This code might appear when the line protocol is being set up, but will disappear once the connection is stabilized.
Ack timer	Number of seconds for which the bundle link waits for a hello acknowledgment before resending a hello message or resending an ADD_LINK message used for initial synchronization.
Hello timer	Interval at which a bundle link sends out hello messages.
Max retry count	Maximum number of times that a bundle link will resend a hello message before receiving an acknowledgment or resending an ADD_LINK message.
Current count	Number of retries that have been attempted.
Peer LID	Bundle link identification name of the peer end of the link.
RTT	Round-trip time as measured by utilizing the Timestamp Information Element in the HELLO and HELLO_ACK messages.
Statistics	Displays statistics for each bundle link.
Add_link sent	Number of Add_link messages sent. Add_link messages notify the peer endpoint that the local endpoint is ready to process frames.
Add_link rcv'd	Number of Add_link messages received.
Add_link ack sent	Number of Add_link acknowledgments sent. Add_link acknowledgments notify the peer endpoint that an Add_link message was received.
Add_link ack rcv'd	Number of Add_link acknowledgments received.
Add_link rej sent	Number of Add_link_reject messages sent.
Add_link rej rcv'd	Number of Add_link_reject messages received.

Table 2 show frame-relay multilink Field Descriptions (continued)

Field	Description
Remove_link sent	Number of Remove_link messages sent. Remove_link messages notify the peer that on the local end, a bundle link is being removed from the bundle.
Remove_link rcv'd	Number of Remove_link messages received.
Remove_link_ack sent	Number of Remove_link acknowledgments sent. Remove_link acknowledgments notify the peer that a Remove_link message has been received.
Remove_link_ack rcv'd	Number of Remove_link acknowledgments received.
Hello sent	Number of hello messages sent. Hello messages notify the peer endpoint that the local endpoint remains in the "up" state.
Hello rcv'd	Number of hello messages received.
Hello_ack sent	Number of hello acknowledgments sent. Hello acknowledgments notify the peer that hello messages have been received.
Hello_ack rcv'd	Number of hello acknowledgments received.
outgoing pak dropped	Number of outgoing packets dropped.
incoming pak dropped	Number of incoming packets dropped.

Related Commands

Command	Description
debug frame-relay multilink	Displays debug messages for multilink Frame Relay bundles and bundle links.

Glossary

BID—Bundle identification. The BID is the name used to identify the bundle. The BID can be assigned, or the default can be used.

BL_ACTIVATE—A message that controls the addition of a bundle link to a Frame Relay bundle.

BL_DEACTIVATE—A message that controls the removal a bundle link from a Frame Relay bundle.

bundle—A logical grouping of one or more physical interfaces using the formats and procedures of multilink Frame Relay. A bundle emulates a physical interface to the Frame Relay data-link layer. The bundle is also referred to as the *MFR interface*.

bundle link—An individual physical interface that is a member of a bundle.

DLCI—data-link connection identifier. A value that identifies a permanent virtual circuit (PVC) in a Frame Relay network.

HELLO message—A message that notifies a peer endpoint that the local endpoint is in the operational state (up).

HELLO_ACK—A message that notifies a peer endpoint that a hello message has been received.

LID—link identification. The LID is the name used to identify a bundle link. The LID can be assigned, or the default can be used.

LMI—Local Management Interface. A set of enhancements to the basic Frame Relay specification. LMI includes support for a keepalive mechanism, which verifies that data is flowing; a multicast mechanism, which provides the network server with its local DLCI and the multicast DLCI; global addressing, which gives DLCIs global rather than local significance in Frame Relay networks; and a status mechanism, which provides an ongoing status report on the DLCIs known to the switch.

NNI—Network-to-Network Interface. The interface between two Frame Relay devices that are both located in a private network or both located in a public network.

PH_ACTIVATE—A message that indicates that the Frame Relay bundle is up.

PH_DEACTIVATE—A message that indicates that the Frame Relay bundle is down.

UNI—User-to-Network Interface. The interface between a Frame Relay device in a public network and a Frame Relay device in a private network.

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.