



Multilink Frame Relay (FRF.16.1)

First Published: May 14, 2001

Last Updated: August 08, 2007

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.

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.



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

© 2007 Cisco Systems, Inc. All rights reserved.

Contents

- [Prerequisites for Multilink Frame Relay \(FRF.16.1\), page 2](#)
- [Restrictions for Multilink Frame Relay \(FRF.16.1\), page 2](#)
- [Information About Multilink Frame Relay \(FRF.16.1\), page 2](#)
- [How to Enable Multilink Frame Relay \(FRF.16.1\), page 4](#)
- [Configuration Examples for Multilink Frame Relay \(FRF.16.1\), page 11](#)
- [Additional References, page 12](#)
- [Command Reference, page 13](#)
- [Glossary, page 14](#)

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.
- The multilink Frame Relay MIB (RFC 3020) is not supported.
- 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 2](#)
- [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 7](#)
- [Monitoring and Maintaining Multilink Frame Relay \(FRF.16.1\), page 9](#)

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 dlc1`
11. `end`
12. `show frame-relay multilink`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example: Router> enable</p>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example: Router# configure terminal</p>	<p>Enters global configuration mode.</p>
Step 3	<p><code>interface mfr interface-number</code></p> <p>Example: Router(config)# interface mfr mfr1</p>	<p>Configures a multilink Frame Relay bundle interface.</p>
Step 4	<p><code>frame-relay multilink bandwidth-class [a b c [threshold]]</code></p> <p>Example: Router(config-if)# frame-relay multilink bandwidth-class a</p> <p>or</p> <p>Router(config-if)# frame-relay multilink bandwidth-class b</p> <p>or</p> <p>Router(config-if)# frame-relay multilink bandwidth-class c 3</p>	<p>(Optional) Specifies the bandwidth class criterion used to activate or deactivate a Frame Relay bundle.</p> <ul style="list-style-type: none"> 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. <p> 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).</p>

	Command or Action	Purpose
Step 5	<pre>frame-relay intf-type dce</pre> <p>Example: Router(config-if)# frame-relay intf-type dce</p>	<p>Configures a device to function as the data circuit-terminating equipment (DCE).</p> <ul style="list-style-type: none"> 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	<pre>frame-relay multilink bid name</pre> <p>Example: Router(config-if)# frame-relay multilink bid router1</p>	<p>(Optional) Assigns a bundle identification name to a multilink Frame Relay bundle.</p> <ul style="list-style-type: none"> 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	<pre>frame-relay multilink output-threshold bytes</pre> <p>Example: Router(config-if)# frame-relay multilink output-threshold 500</p>	<p>(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.</p> <ul style="list-style-type: none"> When configured on the bundle interface, this command applies to all bundle links in the bundle.
Step 8	<pre>interface mfr interface-number.subinterface-number point-to-point</pre> <p>Example: Router(config-if)# interface mfr1.1 point-to-point</p>	<p>Configures a point-to-point multilink Frame Relay subinterface.</p>
Step 9	<pre>ip address ip-address mask</pre> <p>Example: Router(config-subif)# ip address 10.0.1.1 255.255.255.0</p>	<p>Configures the IP address for the subinterface.</p>
Step 10	<pre>frame-relay interface-dlci dlci</pre> <p>Example: Router(config-subif)# frame-relay interface-dlci 100</p>	<p>Assigns a data-link connection identifier (DLCI) to a Frame Relay subinterface.</p>

	Command or Action	Purpose
Step 11	<code>end</code> Example: <code>Router(config-subif)# end</code>	Ends the configuration session and returns to privileged EXEC mode.
Step 12	<code>show frame-relay multilink</code> Example: <code>Router# show frame-relay multilink</code>	(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	<code>enable</code> Example: <code>Router> enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<code>configure terminal</code> Example: <code>Router# configure terminal</code>	Enters global configuration mode.

	Command or Action	Purpose
Step 3	<code>interface serial number</code> Example: Router(config)# interface serial 5/0	Selects a physical interface and enters interface configuration mode.
Step 4	<code>encapsulation frame-relay mfr number [name]</code> Example: Router(config-if)# encapsulation frame-relay mfr1	Creates a multilink Frame Relay bundle link and associates the link with a bundle.
Step 5	<code>frame-relay multilink output-threshold bytes</code> 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	<code>frame-relay multilink lid name</code> Example: Router(config-if)# frame-relay multilink lid first-link	(Optional) Assigns a bundle link identification name with a multilink Frame Relay bundle link. <ul style="list-style-type: none"> 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	<code>frame-relay multilink hello seconds</code> Example: Router(config-if)# frame-relay multilink hello 9	(Optional) Configures the interval at which a bundle link will send out hello messages. <ul style="list-style-type: none"> The default value is 10 seconds.
Step 8	<code>frame-relay multilink ack seconds</code> 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. <ul style="list-style-type: none"> The default value is 4 seconds.
Step 9	<code>frame-relay multilink retry number</code> 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. <ul style="list-style-type: none"> The default value is 2 tries.
Step 10	<code>end</code> Example: Router(config-if)# end	Ends the configuration session and returns to privileged EXEC mode.
Step 11	<code>show frame-relay multilink</code> 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. <ul style="list-style-type: none"> • 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 11](#)
- [Configuring Variable Bandwidth Class Support: Example, page 11](#)

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	<i>Multilink Frame Relay UNI/NNI Implementation Agreement</i> , 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
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>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.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/techsupport</p>

Command Reference

The following commands are introduced or modified in the feature or features documented in this module. For information about these commands, see the *Cisco IOS Wide-Area Networking Command Reference* at http://www.cisco.com/en/US/docs/ios/wan/command/reference/wan_book.html. For information about all Cisco IOS commands, go to the Command Lookup Tool at <http://tools.cisco.com/Support/CLILookup> or to the *Cisco IOS Master Commands List*.

- **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**

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.

CCVP, the Cisco logo, and Welcome to the Human Network are trademarks of Cisco Systems, Inc.; Changing the Way We Work, Live, Play, and Learn is a service mark of Cisco Systems, Inc.; and Access Registrar, Aironet, Catalyst, CCDA, CCDP, CCIE, CCIP, CCNA, CCNP, CCSP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Cisco Unity, Enterprise/Solver, EtherChannel, EtherFast, EtherSwitch, Fast Step, Follow Me Browsing, FormShare, GigaDrive, HomeLink, Internet Quotient, IOS, iPhone, IP/TV, iQ Expertise, the iQ logo, iQ Net Readiness Scorecard, iQuick Study, LightStream, Linksys, MeetingPlace, MGX, Networkers, Networking Academy, Network Registrar, PIX, ProConnect, ScriptShare, SMARTnet, StackWise, The Fastest Way to Increase Your Internet Quotient, and TransPath are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

All other trademarks mentioned in this document or Website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0711R)

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.

© 2007 Cisco Systems, Inc. All rights reserved.