frame-relay end-to-end keepalive error-threshold

To modify the keepalive error threshold value, use the **frame-relay end-to-end keepalive error-threshold** map-class configuration command. To reset the error threshold value to its default, use the **no** form of this command.

frame-relay end-to-end keepalive error-threshold {**send** | **receive**} *count*

no frame-relay end-to-end keepalive error-threshold {**send** | **receive**}

Syntax Description

send	Number of send-side errors in the event window before keepalive status goes from up to down.
receive	Number of receive-side errors in the event window before keepalive status goes from up to down.
count	Number of errors required. The maximum value is 32.

Defaults

The default value for both the send and receive error threshold is 2.

Command Modes

Map-class configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

The send-side value can only be configured in bidirectional and request modes. The receive-side value can only be configured in bidirectional and reply modes. See the **frame-relay end-to-end keepalive mode** command. When you configure the error threshold, you will also want to configure the event window. See the **frame-relay end-to-end keepalive event-window** command.

Examples

The following example shows increasing the receive-side error threshold to 4 and changing the event window to 7:

```
map-class frame-relay olga
  frame-relay end-to-end keepalive reply
  frame-relay end-to-end keepalive error-threshold receive 4
  frame-relay end-to-end keepalive event-window receive 7
```

Related Commands

Command	Description
frame-relay end-to-end keepalive event-window	Modifies the keepalive event window value.
frame-relay end-to-end keepalive mode	Enables Frame Relay end-to-end keepalives.
frame-relay end-to-end keepalive success-events	Modifies the keepalive success events value.
frame-relay end-to-end keepalive timer	Modifies the keepalive timer.

Command	Description
map-class frame-relay	Specifies a map class to define QoS values for an SVC.
show frame-relay end-to-end keepalive	Displays statistics about Frame Relay end-to-end keepalive.

frame-relay end-to-end keepalive event-window

To modify the keepalive event window value, use the **frame-relay end-to-end keepalive event-window** map-class configuration command. To reset the default event window size, use the **no** form of this command.

frame-relay end-to-end keepalive event-window {**send** | **receive**} *size*

no frame-relay end-to-end keepalive event-window {**send** | **receive**}

Syntax Description

send	The size of the send-side event window.
receive	The size of the receive-side event window.
size	Number of events in the event window. The maximum value is 32.

Defaults

The default value for both the send and receive event windows is 3.

Command Modes

Map-class configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

The send-side value can only be configured in bidirectional and request modes. The receive-side value can only be configured in bidirectional and reply modes. See the **frame-relay end-to-end keepalive mode** command. When you configure the event window, you will also want to configure the error-threshold. See the **frame-relay end-to-end keepalive error-threshold** command.

Examples

The following example shows increasing the receive-side error threshold to 4 and changing the event window to 7:

```
map-class frame-relay olga
  frame-relay end-to-end keepalive reply
  frame-relay end-to-end keepalive error-threshold receive 4
  frame-relay end-to-end keepalive event-window receive 7
```

Related Commands

Command	Description
frame-relay end-to-end keepalive error-threshold	Modifies the keepalive error threshold value.
frame-relay end-to-end keepalive mode	Enables Frame Relay end-to-end keepalives.
frame-relay end-to-end keepalive success-events	Modifies the keepalive success events value.
frame-relay end-to-end keepalive timer	Modifies the keepalive timer.

Command	Description
map-class frame-relay	Specifies a map class to define QoS values for an SVC.
show frame-relay end-to-end keepalive	Displays statistics about Frame Relay end-to-end keepalive.

frame-relay end-to-end keepalive mode

To enable Frame Relay end-to-end keepalives, use the **frame-relay end-to-end keepalive mode** map-class configuration command. To disable Frame Relay end-to-end keepalives, use the **no** form of this command.

frame-relay end-to-end keepalive mode {**bidirectional** | **request** | **reply** | **passive-reply**}

no frame-relay end-to-end keepalive mode

Syntax Description

bidirectional	Enables bidirectional mode.
request	Enables request mode.
reply	Enables reply mode.
passive-reply	Enables passive reply mode.

Defaults

When a Frame Relay end-to-end keepalive mode is enabled, default values depend on which mode is selected. For the meaning of the parameters, see the **frame-relay end-to-end keepalive timer**, **frame-relay end-to-end keepalive event-window**, **frame-relay end-to-end keepalive error-threshold**, and **frame-relay end-to-end keepalive success-events** commands.

Command Modes

Map-class configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

To enable Frame Relay end-to-end keepalives, Frame Relay must be configured. In addition, a map-class must be associated and a DLCI assigned to an interface, subinterface, VC or PVC. For more information on associating a frame-relay class with an interface, subinterface, VC or PVC, see the **frame-relay class** command. For more information on assigning a DLCI to an interface, subinterface, VC or PVC, see the **frame-relay interface-dlci** command.

In bidirectional mode, both ends of a virtual circuit (VC) send keepalive requests and respond to keepalive requests. If one end of the VC is configured in the bidirectional mode, the other end must also be configured in the bidirectional mode.

In request mode, the router sends keepalive requests and expects replies from the other end of the VC. If one end of a VC is configured in the request mode, the other end must be configured in the reply or passive-reply mode.

In reply mode, the router does not send keepalive requests, but waits for keepalive requests from the other end of the VC and replies to them. If no keepalive request has arrived within the timer interval, the router times out and increments the error counter by 1. If one end of a VC is configured in the reply mode, the other end must be configured in the request mode.

In passive-reply mode, the router does not send keepalive requests, but waits for keepalive requests from the other end of the VC and replies to them. No timer is set when in this mode, and the error counter is not incremented. If one end of a VC is configured in the passive-reply mode, the other end must be configured in the request mode.

Table 23 displays parameter values for send- and receive-sides in bidirectional mode

Table 23 Bidirectional Mode

Parameter	Send-Side	Receive-Side
Timer	10 seconds	15 seconds
Event window	3	3
Error threshold	2	2
Success events	2	2

Table 24 displays parameter values for send- and receive-sides in request mode.

Table 24 Request Mode

Parameter	Send-Side	Receive-Side
Timer	10 seconds	no value set
Event window	3	no value set
Error threshold	2	no value set
Success events	2	no value set

Table 25 displays parameter values for send- and receive-sides in reply mode.

Table 25 Reply Mode

Parameter	Send-Side	Receive-Side
Timer	no value set	15 seconds
Event window	no value set	3
Error threshold	no value set	2
Success events	no value set	2

Passive-Reply Mode

In passive-reply mode, no values are set.

Examples

The following example configures one end of a VC so that a DLCI is assigned to a Frame Relay serial interface, a map class is associated with the interface, and Frame Relay end-to-end keepalive is configured in bidirectional mode using default values:

```
router1(config) interface serial 0/0.1 point-to-point
router1(config-if) ip address 10.1.1.1 255.255.255.0
router1(config-if) frame-relay interface-dlci 16
router1(config-if) frame-relay class vcgrp1
router1(config-if) exit
!
```

```
router1(config)# map-class frame-relay vcgrp1
router1(config-map-class)# frame-relay end-to-end keepalive mode bidirectional
```

The following example configures one end of a VC to reply to keepalive requests and to increment its error counter if no keepalive requests are received 30 seconds after the latest request:

```
router1(config)# map-class frame-relay oro34
router1(config-map-class)# frame-relay end-to-end keepalive reply
router1(config-map-class)# frame-relay end-to-end keepalive timer receive 30
```

Related Commands

Command	Description
frame-relay end-to-end keepalive error-threshold	Modifies the keepalive error threshold value.
frame-relay end-to-end keepalive event-window	Modifies the keepalive event window value.
frame-relay end-to-end keepalive success-events	Modifies the keepalive success events value.
frame-relay end-to-end keepalive timer	Modifies the keepalive timer.
map-class frame-relay	Specifies a map class to define QoS values for an SVC.
show frame-relay end-to-end keepalive	Displays statistics about Frame Relay end-to-end keepalive.

frame-relay end-to-end keepalive success-events

To modify the keepalive success events value, use the **frame-relay end-to-end keepalive success-events** map-class configuration command. To reset the success events value to its default, use the **no** form of this command.

frame-relay end-to-end keepalive success-events {**send** | **receive**} *count*

no frame-relay end-to-end keepalive success-events {**send** | **receive**}

Syntax Description

send	The number of consecutive send-side success events required to change the keepalive state from down to up.
receive	The number of consecutive receive-side success events required to change the keepalive state from down to up.
count	Number of consecutive success events required. The maximum value is 32.

Defaults

The default value for both the send and receive success events is 2.

Command Modes

Map-class configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

The send-side value can only be configured in bidirectional and request modes. The receive-side value can only be configured in the bidirectional and reply modes. See the **frame-relay end-to-end keepalive mode** command.

If the success events value is set to a low value at the same time that a low value is set for the error threshold value of the **frame-relay end-to-end keepalive error-threshold** command, the keepalive state of the VC may flap from state to state.

Examples

The following example shows how to increase the success events value:

```
map-class frame-relay vcgrp4
 frame-relay end-to-end keepalive request
 frame-relay end-to-end keepalive success-events send 4
```

Related Commands

Command	Description
frame-relay end-to-end keepalive error-threshold	Modifies the keepalive error threshold value.
frame-relay end-to-end keepalive event-window	Modifies the keepalive event window value.
frame-relay end-to-end keepalive mode	Enables Frame Relay end-to-end keepalives.

Command	Description
frame-relay end-to-end keepalive timer	Modifies the keepalive timer.
map-class frame-relay	Specifies a map class to define QoS values for an SVC.
show frame-relay end-to-end keepalive	Displays statistics about Frame Relay end-to-end keepalive.

frame-relay end-to-end keepalive timer

To modify the keepalive timer value, use the **frame-relay end-to-end keepalive timer** map-class configuration command. To reset the timer value to its default, use the **no** form of this command.

frame-relay end-to-end keepalive timer {**send** | **receive**} *interval*

no frame-relay end-to-end keepalive timer {**send** | **receive**}

Syntax Description

send	How frequently to send a keepalive request.
receive	How long before the receive-side error counter is incremented if no request is received.
interval	Time in seconds for the timer to expire.

Defaults

The default value for the send timer is 10 seconds. The default value for the receive timer is 15 seconds.

Command Modes

Map-class configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

The send-side value can only be configured in bidirectional and request modes. The receive-side value can only be configured in the bidirectional and reply modes. See the **frame-relay end-to-end keepalive mode** command.

The send-side timer expires if a reply has not been received *interval* seconds after a request is sent. The receive-side timer expires if a request has not been received *interval* seconds after the previous request.

Examples

The following example shows how to set up one end of a virtual circuit (VC) to send a keepalive request every 15 seconds and increment the error counter if more than 22 seconds elapse between receiving keepalive responses:

```
map-class frame-relay vcgrp1
  frame-relay end-to-end keepalive bidirectional
  frame-relay end-to-end keepalive timer send 15
  frame-relay end-to-end keepalive timer receive 22
```

Related Commands

Command	Description
frame-relay end-to-end keepalive error-threshold	Modifies the keepalive error threshold value.
frame-relay end-to-end keepalive event-window	Modifies the keepalive event window value.
frame-relay end-to-end keepalive mode	Enables Frame Relay end-to-end keepalives.
frame-relay end-to-end keepalive success-events	Modifies the keepalive success events value.

Command	Description
map-class frame-relay	Specifies a map class to define QoS values for an SVC.
show frame-relay end-to-end keepalive	Displays statistics about Frame Relay end-to-end keepalive.

frame-relay fair-queue

To enable weighted fair queueing for one or more Frame Relay permanent virtual circuits (PVCs), use the **frame-relay fair-queue** map-class configuration command in conjunction with the **map-class frame-relay** command. To disable weighted fair queueing for a Frame Relay map class, use the **no** form of this command.

frame-relay fair-queue [*congestive_discard_threshold* [*number_dynamic_conversation_queues* [*number_reservable_conversation_queues* [*max_buffer_size_for_fair_queues*]]]]

no frame-relay fair-queue [*congestive_discard_threshold* [*number_dynamic_conversation_queues* [*number_reservable_conversation_queues* [*max_buffer_size_for_fair_queues*]]]]

Syntax Description

congestive_discard_threshold	(Optional) Specifies the number of messages allowed in each queue. The range is from 1 to 4096 messages; the default is 64.
number_dynamic_conversation_ queues	(Optional) Specifies the number of dynamic queues to be used for best-effort conversations—normal conversations not requiring any special network services. Valid values are 16, 32, 64, 128, 256, 512, 1024, 2048, and 4096; the default is 16.
number_reservable_conversation_queues	(Optional) Specifies the number of reserved queues to be used for carrying voice traffic. The range is from 0 to 100; the default is 0. (The command-line interface (CLI) will not allow a value of less than 2 if fragmentation is configured for the Frame Relay map-class.)
max_buffer_size_for_fair_queues	(Optional) Specifies the maximum buffer size in bytes for all of the fair queues. The range is from 0 to 4096 bytes; the default is 600.

Defaults

Disabled

Command Modes

Map-class configuration

Command History

Release	Modification
12.0(3)XG	This command was introduced.
12.0(4)T	This command was implemented in Cisco IOS Release 12.0 T.

Usage Guidelines

To use this command, you must first associate a Frame Relay map class with a specific data-link connection identifier (DLCI), and then enter map-class configuration mode and enable or disable weighted fair queueing for that map class.

When Frame Relay fragmentation is enabled, weighted fair queueing is the only queueing strategy allowed.

If this command is entered without any accompanying numbers, the default values for each of the four parameters will be set. If you desire to alter only the value of the first parameter (*congestive_discard_threshold*), you only need to enter the desired value for that parameter. If you desire to alter only the value of the second, third, or fourth parameters, you must enter values for the preceding parameters as well as for the parameter you wish to change.

Examples

The following example shows how to enable weighted fair queueing and set the default parameter values for the "vofr" Frame Relay map class on a Cisco 2600 series, 3600 series, or 7200 series router or on a Cisco MC3810:

```
interface serial 1/1
 frame-relay interface-dlci 100
  class vofr
  exit
map-class frame-relay vofr
 frame-relay fair-queue
```

The following example shows how to enable weighted fair queueing and set the *congestive_discard_threshold* parameter to a value other than the default value for the "vofr" Frame Relay map class on a Cisco 2600 series, 3600 series, or 7200 series router or on an MC3810 concentrator:

```
interface serial 1/1
 frame-relay interface-dlci 100
  class vofr
  exit
map-class frame-relay vofr
 frame-relay fair-queue 255
```

The following example shows how to enable weighted fair queueing and set the*number_reservable_conversation_queues* to a value of 25 for the "vofr" Frame Relay map class on a Cisco 2600 series, 3600 series, or 7200 series router or on a Cisco MC3810:

```
interface serial 1/1
 frame-relay interface-dlci 100
  class vofr
  exit
map-class frame-relay vofr
 frame-relay fair-queue 64 256 25
```

Related Commands

Command	Description
class (virtual circuit)	Associates a map class with a specified DLCI.
frame-relay fragment	Enables fragmentation for a Frame Relay map class.
frame-relay interface-dlci	Assigns a DLCI to a specified Frame Relay subinterface on the router or access server.
map-class frame-relay	Specifies a map class to define QoS values for an SVC.

frame-relay fragment

To enable fragmentation of Frame Relay frames for a Frame Relay map class, use the **frame-relay fragment** map-class configuration command. To disable Frame Relay fragmentation, use the **no** form of this command.

frame-relay fragment *fragment_size* [**switched**]

no frame-relay fragment

Syntax Description

fragment_size	Specifies the number of payload bytes from the original Frame Relay frame that will go into each fragment. This number excludes the Frame Relay header of the original frame. All the fragments of a Frame Relay frame except the last will have a payload size equal to *fragment_size*; the last fragment will have a payload less than or equal to *fragment_size*. Valid values are from 16 to 1600 bytes; the default is 53.
switched	(Optional) Specifies that fragmentation will be enabled on a switched permanent virtual circuit (PVC).

Defaults

Fragmentation is disabled.

Command Modes

Map-class configuration

Command History

Release	Modification
12.0(3)XG	This command was introduced.
12.0(4)T	This command was implemented in Cisco IOS Release 12.0 T.
12.1(2)T	This command was modified to extend end-to-end FRF.12 fragmentation support to additional platforms and to switched Frame Relay PVCs.
12.1(2)E	This command was introduced for Cisco 7500 series routers with a Versatile Interface Processor.
12.1(5)T	This command was introduced for Cisco 7500 series routers with a Versatile Interface Processor running Cisco IOS Release 12.1(5)T.

Usage Guidelines

You should enable fragmentation for low-speed links (meaning those operating at less than 768 kbps).

Frame Relay fragmentation is enabled on a per-PVC basis. Before enabling Frame Relay fragmentation, you must first associate a Frame Relay map class with a specific data-link connection identifier (DLCI), and then enter map-class configuration mode and enable or disable fragmentation for that map class. In addition, you must enable Frame Relay traffic shaping on the interface in order for fragmentation to work.

Selecting a Fragmentation Format

Frame Relay frames are fragmented using one of the following formats, depending on how the PVC is configured:

- Pure end-to-end FRF.12 format
- FRF.11 Annex C format
- Cisco proprietary format

Only pure end-to-end FRF.12 fragmentation can be configured on switched PVCs.

Cisco recommends pure end-to-end FRF.12 fragmentation on PVCs that are carrying VoIP packets and on PVCs that are sharing the link with other PVCs carrying Voice over Frame Relay (VoFR) traffic.

In pure end-to-end FRF.12 fragmentation, Frame Relay frames having a payload less than the fragment size configured for that PVC are transmitted without the fragmentation header.

FRF.11 Annex C and Cisco proprietary fragmentation are used when VoFR frames are transmitted on a PVC. When fragmentation is enabled on a PVC, FRF.11 Annex C format is implemented when **vofr** is configured on that PVC; Cisco proprietary format is implemented when **vofr cisco** is configured.

In FRF.11 Annex C and Cisco proprietary fragmentation, VoFR frames are never fragmented, and all data packets (including VoIP packets) contain the fragmentation header regardless of the payload size.

Selecting a Fragment Size

You should set the fragment size based on the lowest port speed between the routers. For example, for a hub-and-spoke Frame Relay topology where the hub has a T1 speed and the remote routers have 64 kbps port speeds, the fragmentation size must be set for the 64 kbps speed on both routers. Any other PVCs that share the same physical interface must use the same fragmentation size used by the voice PVC.

With pure end-to-end FRF.12 fragmentation, you should select a fragment size that is larger than the voice packet size.

Table 26 shows the recommended fragmentation sizes for a serialization delay of 10 ms.

Table 26 Recommended Fragment Size for 10 ms Serialization Delay

Lowest Link Speed in Path	Recommended Fragment Size
56 kbps	70 bytes
64 kbps	80 bytes
128 kbps	160 bytes
256 kbps	320 bytes
512 kbps	640 bytes
768 kbps	1000 bytes
1536 kbps	1600 bytes

Examples

FRF.12 Fragmentation on a Switched PVC Example

The following example shows how to configure pure end-to-end FRF.12 fragmentation in the map class "data." The map class is associated with switched PVC 20 on serial interface 3/3.

```
Router(config)# frame-relay switching
!
Router(config)# interface Serial3/2
Router(config-if)# encapsulation frame-relay
```

```
Router(config-if)# frame-relay intf-type dce
!
Router(config)# interface Serial3/3
Router(config-if)# encapsulation frame-relay
Router(config-if)# frame-relay traffic-shaping
Router(config-if)# frame-relay interface-dlci 20 switched
Router(config-fr-dlci)# class data
Router(config-if)# frame-relay intf-type dce
!
Router(config)# map-class frame-relay data
Router(config-map-class)# frame-relay fragment 80 switched
Router(config-map-class)# frame-relay cir 64000
Router(config-map-class)# frame-relay bc 640
!
Router(config)# connect data Serial3/2 16 Serial3/3 20
```

End-to-End FRF.12 Fragmentation Examples

The following example shows how to enable pure end-to-end FRF.12 fragmentation for the "frag" map class. The fragment payload size is set to 160 bytes. Frame Relay traffic shaping is required on the PVC; the only queueing type supported on the PVC when fragmentation is configured is weighted fair queueing (WFQ).

```
Router(config)# interface serial 1/0/0
Router(config-if)# frame-relay traffic-shaping
Router(config-if)# frame-relay interface-dlci 100
Router(config-fr-dlci)# class frag
Router(config-fr-dlci)# exit

Router(config)# map-class frame-relay frag
Router(config-map-class)# frame-relay cir 128000
Router(config-map-class)# frame-relay bc 1280
Router(config-map-class)# frame-relay fragment 160
Router(config-map-class)# frame-relay fair-queue
```

The following example is for the same configuration on a VIP-enabled Cisco 7500 series router:

```
Router(config)# class-map frf
Router(config-cmap)# match protocol vofr
Router(config-cmap)# exit
Router(config)# policy-map llq
Router(config-pmap)# class frf
Router(config-pmap-c)# priority 2000
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# policy-map llq-shape
Router(config-pmap)# class class-default
Router(config-pmap-c)# shape average 1000 128000
Router(config-pmap-c)# service-policy llq
Router(config-pmap-c)# exit
Router(config-pmap)# exit

Router(config)# interface serial 1/0/0.1
Router(config-if)# frame-relay interface-dlci 100
Router(config-fr-dlci)# class frag
Router(config-fr-dlci)# exit

Router(config)# map-class frame-relay frag
Router(config-map-class)# frame-relay fragment 40
Router(config-map-class)# service-policy llq-shape
Router(config-map-class)#
```

FRF.11 Annex C Fragmentation Configuration Examples

The following example shows how to enable FRF.11 Annex C fragmentation for data on a Cisco MC3810 PVC configured for VoFR. Note that fragmentation must be configured if a VoFR PVC is to carry data. The fragment payload size is set to 160 bytes. Frame Relay traffic shaping is required on the PVC; the only queueing type supported on the PVC when fragmentation is configured is weighted fair queueing (WFQ).

```
Router(config)# interface serial 1/1
Router(config-if)# frame-relay traffic-shaping
Router(config-if)# frame-relay interface-dlci 101
Router(config-fr-dlci)# vofr
Router(config-fr-dlci)# class frag
Router(config-fr-dlci)# exit

Router(config)# map-class frame-relay frag
Router(config-map-class)# frame-relay cir 128000
Router(config-map-class)# frame-relay bc 1280
Router(config-map-class)# frame-relay fragment 160
Router(config-map-class)# frame-relay fair-queue
Router(config-map-class)#
```

The following example is for the same configuration on a VIP-enabled Cisco 7500 series router:

```
Router(config)# class-map frf
Router(config-cmap)# match protocol vofr
Router(config-cmap)# exit
Router(config)# policy-map llq
Router(config-pmap)# class frf
Router(config-pmap-c)# priority 2000
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# policy-map llq-shape
Router(config-pmap)# class class-default
Router(config-pmap-c)# shape average 1000 128000
Router(config-pmap-c)# service-policy llq
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# interface serial 1/1/0.1
Router(config-if)# frame-relay interface-dlci 101
Router(config-fr-dlci)# class frag
Router(config-fr-dlci)# exit

Router(config)# map-class frame-relay frag
Router(config-map-class)# frame-relay fragment 40
Router(config-map-class)# service-policy llq-shape
Router(config-map-class)#
```

Cisco-Proprietary Fragmentation Examples

The following example shows how to enable Cisco-proprietary Frame Relay fragmentation for the "frag" Frame Relay map class on a Cisco 2600 series, 3600 series, or 7200 series router, starting from global configuration mode. The fragment payload size is set to 160 bytes. Frame Relay traffic shaping is required on the PVC; the only queueing type supported on the PVC when fragmentation is configured is weighted fair queueing (WFQ).

```
Router(config)# interface serial 2/0/0
Router(config-if)# frame-relay traffic-shaping
Router(config-if)# frame-relay interface-dlci 102
Router(config-fr-dlci)# vofr cisco
Router(config-fr-dlci)# class frag
Router(config-fr-dlci)# exit
```

```
Router(config)# map-class frame-relay frag
Router(config-map-class)# frame-relay cir 128000
Router(config-map-class)# frame-relay bc 1280
Router(config-map-class)# frame-relay fragment 160
Router(config-map-class)# frame-relay fair-queue
```

The following example is for the same configuration on a VIP-enabled Cisco 7500 series router:

```
Router(config)# class-map frf
Router(config-cmap)# match protocol vofr
Router(config-cmap)# exit
Router(config)# policy-map llq
Router(config-pmap)# class frf
Router(config-pmap-c)# priority 2000
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# policy-map llq-shape
Router(config-pmap)# class class-default
Router(config-pmap-c)# shape average 1000 128000
Router(config-pmap-c)# service-policy llq
Router(config-pmap-c)# exit
Router(config-pmap)# exit

Router(config)# interface serial 2/0/0.1
Router(config-if)# frame-relay interface-dlci 102
Router(config-fr-dlci)# class frag
Router(config-fr-dlci)# exit

Router(config)# map-class frame-relay frag
Router(config-map-class)# frame-relay fragment 40
Router(config-map-class)# service-policy llq-shape
```

Related Commands

Command	Description
class (virtual circuit)	Associates a map class with a specified DLCI.
debug frame-relay fragment	Displays information related to Frame Relay fragmentation on a PVC.
frame-relay fair-queue	Enables weighted fair queueing for one or more Frame Relay PVCs.
frame-relay interface-dlci	Assigns a DLCI to a specified Frame Relay subinterface on the router or access server.
frame-relay traffic-shaping	Enables traffic shaping and per-virtual circuit queueing for all PVCs and SVCs on a Frame Relay interface.
map-class frame-relay	Specifies a map class to define QoS values for an SVC.

frame-relay holdq

To configure the maximum size of a traffic-shaping queue on a switched PVC, use the **frame-relay holdq** map-class configuration command. To reconfigure the size of the queue, use the **no** form of this command.

frame-relay holdq *queue-size*

no frame-relay holdq *queue-size*

Syntax Description

queue-size	Size of the traffic-shaping queue, as specified in maximum number of packets. The range is from 1 to 512.

Defaults

40 packets

Command Modes

Map-class configuration

Command History

Release	Modification
12.1(2)T	This command was introduced.

Usage Guidelines

You must enable Frame Relay traffic shaping, using the **frame-relay traffic-shaping** interface command, before **frame-relay holdq** and other traffic-shaping map-class commands will be effective.

You must enable Frame Relay switching, using the **frame-relay switching** global command, before the **frame-relay holdq** command will be effective on switched PVCs.

The **frame-relay holdq** command can be applied to switched PVCs that use FIFO default queueing.

Examples

The following example illustrates the configuration of the maximum size of the traffic-shaping queue on a switched PVC. The queue size is configured in a map class called perpvc_congestion:

```
map-class frame-relay perpvc_congestion
 frame-relay holdq 100
```

Related Commands

Command	Description
frame-relay switching	Enables PVC switching on a Frame Relay DCE or NNI.
frame-relay traffic-shaping	Enables both traffic shaping and per-PVC queueing for all PVCs and SVCs on a Frame Relay interface.

frame-relay idle-timer

To specify the idle timeout interval for a switched virtual circuit (SVC), use the **frame-relay idle-timer** map-class configuration command. To reset the idle timer to its default interval, use the **no** form of this command.

frame-relay idle-timer [**in** | **out**] *seconds*

no frame-relay idle-timer *seconds*

Syntax Description

in	(Optional) timeout interval applies to inbound packet activity.
out	(Optional) timeout interval applies to outbound packet activity.
seconds	Time interval, in seconds, with no frames exchanged on a switched virtual circuit, after which the SVC is released.

Defaults

120 seconds

Command Modes

Map-class configuration

Command History

Release	Modification
11.2	This command was introduced.
11.3	The following keywords were added: • **in** • **out**

Usage Guidelines

The **frame-relay idle-timer** command applies to switched virtual circuits that are associated with the map class where the idle-timer is defined.

The idle timer must be tuned for each application. Routing protocols such as Routing Information Protocol (RIP) might keep the SVC up indefinitely because updates go out every 10 seconds.

Beginning in Release 11.3, if **in** and **out** are not specified in the command, the timeout interval applies to both timers. In Release 11.2, the timeout interval applies to the outbound timer.

Examples

The following example defines the traffic rate and idle timer for the fast_vcs map class and applies those values to DLCI 100, which is associated with that map class:

```
interface serial 0
 frame-relay interface-dlci 100
  class fast_vc

map-class frame-relay fast_vcs
 frame-relay traffic-rate 56000 128000
 frame-relay idle-timer 30
```

Related Commands

Command	Description
map-class frame-relay	Specifies a map class to define QoS values for an SVC.

frame-relay interface-dlci

To assign a data-link connection identifier (DLCI) to a specified Frame Relay subinterface on the router or access server, or to assign a specific permanent virtual circuit (PVC) to a DLCI, or to apply a virtual template configuration for a PPP session, use the **frame-relay interface-dlci** interface configuration command. To remove this assignment, use the **no** form of this command.

frame-relay interface-dlci *dlci* [**ietf** | **cisco**] [**voice-cir** *cir*] [**ppp** *virtual-template-name*]

no frame-relay interface-dlci *dlci* [**ietf** | **cisco**] [**voice-cir** *cir*] [**ppp** *virtual-template-name*]

BOOTP server only

frame-relay interface-dlci *dlci* [**protocol ip** *ip-address*]

Syntax Description

dlci	DLCI number to be used on the specified subinterface.
ietf | **cisco**	(Optional) Encapsulation type: Internet Engineering Task Force (IETF) Frame Relay encapsulation or Cisco Frame Relay encapsulation.
voice-cir *cir*	(Optional; supported on the Cisco MC3810 only.) Specifies the upper limit on the voice bandwidth that may be reserved for this DLCI. The default is the committed information rate (CIR) configured for the Frame Relay map class. For more information, see the "Usage Guidelines" section.
ppp	(Optional) Enables the circuit to use the PPP in Frame Relay encapsulation.
virtual-template-name	(Optional) Specifies which virtual template interface to apply the PPP connection to.
protocol ip *ip-address*	(Optional) Indicates the IP address of the main interface of a new router or access server onto which a router configuration file is to be automatically installed over a Frame Relay network. Use this option only when this device will act as the BOOTP server for automatic installation over Frame Relay.

Defaults

No DLCI is assigned.

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.
11.3(1)MA	The **voice-encap** option was added for the Cisco MC3810.
12.0(1)T	The **ppp** keyword and *virtual-template-name* argument were introduced.
12.0(2)T	The **voice-cir** option was added for the Cisco MC3810.
12.0(3)T	The keyword **x25 profile** was introduced.

Release	Modification
12.0(4)T	Usage guidelines for the Cisco MC3810 were added.
12.0(7)XK	The **voice-encap** keyword for the Cisco MC3810 was removed. This keyword is no longer supported.
12.1(2)T	The **voice-encap** keyword for the Cisco MC3810 was removed. This keyword is no longer supported.

Usage Guidelines

This command is typically used for subinterfaces; however, it can also be used on main interfaces. Using the **frame-relay interface-dlci** command on main interfaces will enable the use of routing protocols on interfaces that use Inverse ARP. The **frame-relay interface-dlci** command on a main interface is also valuable for assigning a specific class to a single PVC where special characteristics are desired. Subinterfaces are logical interfaces associated with a physical interface. You must specify the interface and subinterface before you can use this command to assign any DLCIs and any encapsulation or broadcast options. See the "Examples" section for the sequence of commands.

This command is required for all point-to-point subinterfaces; it is also required for multipoint subinterfaces for which dynamic address resolution is enabled. It is not required for multipoint subinterfaces configured with static address mappings.

Use the **protocol ip** *ip-address* option only when this router or access server will act as the BOOTP server for autoinstallation over Frame Relay.

By issuing the **frame-relay interface-dlci** interface configuration command, you enter Frame Relay DLCI interface configuration mode (see the first example below). This gives you the following command options, which must be used with the relevant class or X.25-profile names you previously assigned:

- **class** *name*—Assigns a mapclass to a DLCI.
- **default**—Sets a command to its defaults.
- **no** {**class** *name* | **x25-profile** *name*}—Cancels the relevant class or X.25 profile.
- **x25-profile** *name*—Assigns an X.25 profile to a DLCI. (Annex G).

A Frame Relay DLCI configured for Annex G can be thought of as a single logical X.25/LAPB interface. Therefore, any number of X.25 routes may be configured to route X.25 calls to that logical interface.

The **voice-cir** option on the Cisco MC3810 provides call admission control; it does not provide traffic shaping. A call setup will be refused if the unallocated bandwidth available at the time of the request is not at least equal to the value of the **voice-cir** option.

When configuring the **voice-cir** option on the Cisco MC3810 for Voice over Frame Relay, do not set the value of this option to be higher than the physical link speed. If Frame Relay traffic shaping is enabled for a PVC sharing voice and data, do not configure the **voice-cir** option to be higher than the value set with the **frame-relay mincir** command.

Note On the Cisco MC3810 only, the **voice-cir** option performs the same function as the **frame-relay voice bandwidth** map-class configuration command introduced in Cisco IOS Release 12.0(3)XG.

For more information about automatically installing router configuration files over a Frame Relay network, see the "Loading and Maintaining System Images" chapter in the *Cisco IOS Configuration Fundamentals Configuration Guide*.

Examples

The following example assigns DLCI 100 to serial subinterface 5.17:

```
! Enter interface configuration and begin assignments on interface serial 5
interface serial 5
! Enter subinterface configuration by assigning subinterface 17
interface serial 5.17
! Now assign a DLCI number to subinterface 5.17
frame-relay interface-dlci 100
```

The following example specifies DLCI 26 over subinterface serial 1.1 and assigns the characteristics under virtual-template 2 to this PPP connection:

```
Router(config)# interface serial1.1 point-to-point
Router(config-if)# frame-relay interface-dlci 26 ppp virtual-template2
```

The following example shows an Annex G connection being created by assigning the X.25 profile "NetworkNodeA" to the Frame Relay DLCI interface 20 on interface serial 1 (having enabled Frame Relay encapsulation on that interface):

```
Router(config)# interface serial1
Router(config-if)# encapsulation frame-relay
Router(config-if)# frame-relay interface-dlci 20
Router(config-fr-dlci)# x25-profile NetworkNodeA
```

The following example assigns DLCI 100 to serial subinterface 5.17:

```
Router(config)# interface serial 5
Router(config-if)# interface serial 5.17
Router(config-if)# frame-relay interface-dlci 100
```

The following example assigns DLCI 100 to a serial interface, starting from global configuration mode:

```
router(config)# interface serial 1/1
router(config-if)# frame-relay interface-dlci 100
router(config-fr-dlci)#
```

Related Commands

Command	Description
frame-relay class	Associates a map class with an interface or subinterface.
show frame-relay pvc	Displays statistics about PVCs for Frame Relay interfaces.
show interface	Displays P1024B/C information.
vofr	Configures subchannels and enables Voice over Frame Relay for a specific DLCI.

frame-relay interface-dlci switched

To indicate that a Frame Relay data-link connection identifier (DLCI) is switched, use the **frame-relay interface-dlci switched** interface configuration command. To remove this assignment, use the **no** form of this command.

frame-relay interface-dlci *dlci* **switched**

no frame-relay interface-dlci *dlci* **switched**

Syntax Description

dlci	DLCI number to be used on the specified interface or subinterface.

Defaults

No DLCI is assigned.

The default PVC type is terminated.

Command Modes

Interface configuration

Command History

Release	Modification
12.1(2)T	This command was introduced.

Usage Guidelines

Use the **frame-relay interface-dlci switched** command to allow a map class to be associated with a switched permanent virtual circuit (PVC).

You cannot change an existing PVC from terminated to switched or vice versa. You must delete the PVC and recreate it in order to change the type.

Use the **frame-relay interface-dlci switched** command to create switched PVCs for configuring Frame Relay-ATM network interworking (FRF.5) and Frame Relay-ATM service interworking (FRF.8).

By issuing the **frame-relay interface-dlci switched** interface configuration command, you enter Frame Relay DLCI interface configuration mode (see the example below).

Examples

In the following example, DLCI 16 on serial interface 0 is identified as a switched PVC and is associated with a map class called “shape256K.”

```
Router(config) # interface serial0
Router(config-if) # encapsulation frame-relay
Router(config-if) # frame-relay interface-dlci 16 switched
Router(config-fr-dlci) #  class shape256K
```

Related Commands

Command	Description
connect (Frame Relay)	Defines connections between Frame Relay PVCs.
frame-relay class	Associates a map class with an interface or subinterface.
frame-relay switching	Enables PVC switching on a Frame Relay DCE or NNI.
show frame-relay pvc	Displays statistics about PVCs for Frame Relay interfaces.

frame-relay intf-type

To configure a Frame Relay switch type, use the **frame-relay intf-type** interface configuration command. To disable the switch, use the **no** form of this command.

frame-relay intf-type [**dce** | **dte** | **nni**]

no frame-relay intf-type [**dce** | **dte** | **nni**]

Syntax Description

dce	(Optional) Router or access server functions as a switch connected to a router.
dte	(Optional) Router or access server is connected to a Frame Relay network.
nni	(Optional) Router or access server functions as a switch connected to a switch—supports Network-to-Network Interface (NNI) connections.

Defaults

dte

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Usage Guidelines

This command can be used only if Frame Relay switching has previously been enabled globally by means of the **frame-relay switching** command.

Examples

The following example configures a DTE switch type:

```
frame-relay switching
!
interface serial 2
 frame-relay intf-type dte
```

frame-relay inverse-arp

To reenable Inverse Address Resolution Protocol (Inverse ARP) on a specified interface or subinterface if the Inverse ARP was previously disabled on a router or access server configured for Frame Relay, use the **frame-relay inverse-arp** interface configuration command. To disable this feature, use the **no** form of this command.

frame-relay inverse-arp [*protocol*] [*dlci*]

no frame-relay inverse-arp [*protocol*] [*dlci*]

Syntax Description

protocol	(Optional) Supported protocols: **appletalk**, **decnet**, **ip**, **ipx**, **vines**, and **xns**.
dlci	(Optional) One of the DLCI numbers used on the interface. Acceptable numbers are integers in the range from 16 through 1007.

Defaults

Enabled

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Usage Guidelines

To enable Inverse ARP for all protocols that were enabled before the prior **no frame-relay inverse-arp** command was issued, use the **frame-relay inverse-arp** command without arguments. To disable Inverse ARP for all protocols of an interface, use the **no frame-relay inverse-arp** command without arguments.

To enable or disable Inverse ARP for a specific protocol and DLCI pair, use both the *protocol* and *dlci* arguments. To enable or disable Inverse ARP for all protocols on a DLCI, use only the *dlci* argument. To enable or disable Inverse ARP for a protocol for all DLCIs on the specified interface or subinterface, use only the *protocol* argument.

This implementation of Inverse ARP is based on RFC 1293. It allows a router or access server running Frame Relay to discover the protocol address of a device associated with the virtual circuit.

In Frame Relay, permanent virtual circuits (PVCs) are identified by a DLCI, which is the equivalent of a hardware address. By exchanging signaling messages, a network announces a new virtual circuit, and with Inverse ARP, the protocol address at the other side of the circuit can be discovered.

The **show frame-relay map** command displays the word “dynamic” to flag virtual circuits that are created dynamically by Inverse ARP.

Examples

The following example sets Inverse ARP on an interface running AppleTalk:

```
interface serial 0
 frame-relay inverse-arp appletalk 100
```

Related Commands

Command	Description
clear frame-relay-inarp	Clears dynamically created Frame Relay maps, which are created by the use of Inverse ARP.
show frame-relay map	Displays the current map entries and information about the connections.

frame-relay ip tcp compression-connections

To specify the maximum number of TCP header compression connections that can exist on a Frame Relay interface, use the **frame-relay ip tcp compression-connections** interface configuration command. To restore the default, use the **no** form of this command.

frame-relay ip tcp compression-connections *number*

no frame-relay ip tcp compression-connections

Syntax Description

number	Maximum number of TCP header compression connections. The range is from 3 to 256.

Defaults

No default behavior or values.

Command Modes

Interface configuration

Command History

Release	Modification
12.1(2)T	This command was introduced.

Usage Guidelines

Before you can configure the maximum number of connections, TCP header compression must be configured on the interface using the **frame-relay ip tcp header-compression** command.

The number of TCP header compression connections must be set to the same value at each end of the connection.

Examples

The following example shows the configuration of a maximum of 150 TCP header compression connections on serial interface 0:

```
interface serial 0
 encapsulation frame-relay
 frame-relay ip tcp header-compression
 frame-relay ip tcp compression-connections 150
```

Related Commands

Command	Description
frame-relay ip tcp header-compression	Enables TCP header compression for all Frame Relay maps on a physical interface.
frame-relay map ip compress	Enables both RTP and TCP header compression on a link.
frame-relay map ip tcp header-compression	Assigns header compression characteristics to an IP map that differ from the compression characteristics of the interface with which the IP map is associated.
show frame-relay ip tcp header-compression	Displays statistics and TCP/IP header compression information for the interface.

frame-relay ip tcp header-compression

To configure an interface to ensure that the associated permanent virtual circuit (PVC) will always carry outgoing TCP/IP headers in compressed form, use the **frame-relay ip tcp header-compression** interface configuration command. To disable compression of TCP/IP packet headers on the interface, use the **no** form of this command.

frame-relay ip tcp header-compression [**passive**]

no frame-relay ip tcp header-compression

Syntax Description

passive	(Optional) Compresses the outgoing TCP/IP packet header only if an incoming packet had a compressed header.

Defaults

Active TCP/IP header compression; all outgoing TCP/IP packets are subjected to header compression.

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Usage Guidelines

This command applies to interfaces that support Frame Relay encapsulation, specifically serial ports and High-Speed Serial Interface (HSSI).

Frame Relay must be configured on the interface before this command can be used.

TCP/IP header compression and Internet Engineering Task Force (IETF) encapsulation are mutually exclusive. If an interface is changed to IETF encapsulation, all encapsulation and compression characteristics are lost.

When you use this command to enable TCP/IP header compression, every IP map inherits the compression characteristics of the interface, unless header compression is explicitly rejected or modified by use of the **frame-relay map ip tcp header compression** command.

We recommend that you shut down the interface prior to changing encapsulation types. Although this is not required, shutting down the interface ensures the interface is reset for the new type.

Examples

The following example configures serial interface 1 to use the default encapsulation (cisco) and passive TCP header compression:

```
interface serial 1
 encapsulation frame-relay
 frame-relay ip tcp header-compression passive
```

Related Commands

Command	Description
frame-relay map ip tcp header-compression	Assigns header compression characteristics to an IP map different from the compression characteristics of the interface with which the IP map is associated.

frame-relay lapf frmr

To resume the default setting of sending the Frame Reject (FRMR) frame at the Link Access Procedure for Frame Relay (LAPF) Frame Reject procedure after having set the option of not sending the frame, use the **frame-relay lapf frmr** command. To set the option of *not* sending the Frame Reject (FRMR) frame at the LAPF Frame Reject procedure, use the **no** form of this command.

frame-relay lapf frmr

no frame-relay lapf frmr

Syntax Description

This command has no arguments or keywords.

Defaults

Send FRMR during the Frame Reject procedure.

Command Modes

Interface configuration

Command History

Release	Modification
11.2	This command was introduced.

Usage Guidelines

If the Frame Relay switch does not support FRMR, use the **no** form of this command to suppress the transmission of FRMR frames.

Examples

The following example suppresses the transmission of FRMR frames:

```
no frame-relay lapf frmr
```

frame-relay lapf k

To set the Link Access Procedure for Frame Relay (LAPF) window size *k*, use the **frame-relay lapf k** interface configuration command. To reset the maximum window size *k* to the default value, use the **no** form of this command.

frame-relay lapf k *number*

no frame-relay lapf k [*number*]

Syntax Description

number	Maximum number of Information frames that either are outstanding for transmission or are transmitted but unacknowledged, in the range from 1 through 127.

Defaults

7 frames

Command Modes

Interface configuration

Command History

Release	Modification
11.2	This command was introduced.

Usage Guidelines

This command is used to tune Layer 2 system parameters to work well with the Frame Relay switch. Normally, you do not need to change the default setting.

Manipulation of Layer 2 parameters is not recommended if you do not know well the resulting functional change. For more information, refer to the ITU-T Q.922 specification for LAPF.

Examples

The following example resets the LAPF window size *k* to the default value:

```
no frame-relay lapf k
```

Related Commands

Command	Description
frame-relay lapf t203	Sets the LAPF link idle timer value T203 of DLCI 0.

frame-relay lapf n200

To set the Link Access Procedure for Frame Relay (LAPF) maximum retransmission count *N200*, use the **frame-relay lapf n200** interface configuration command. To reset the maximum retransmission count to the default of 3, use the **no** form of this command.

frame-relay lapf n200 *retries*

no frame-relay lapf n200 [*retries*]

Syntax Description

retries	Maximum number of retransmissions of a frame.

Defaults

3 retransmissions

Command Modes

Interface configuration

Command History

Release	Modification
11.2	This command was introduced.

Usage Guidelines

This command is used to tune Layer 2 system parameters to work well with the Frame Relay switch. Normally, you do not need to change the default setting.

Manipulation of Layer 2 parameters is not recommended if you do not know well the resulting functional change. For more information, refer to the ITU-T Q.922 specification for LAPF.

Examples

The following example resets the N200 maximum retransmission count to the default value:

```
no frame-relay lapf n200
```

frame-relay lapf n201

To set the Link Access Procedure for Frame Relay (LAPF) N201 value (the maximum length of the Information field of the LAPF I frame), use the **frame-relay lapf n201** interface configuration command. To reset the maximum length of the Information field to the default of 260 bytes (octets), use the **no** form of this command.

frame-relay lapf n201 *bytes*

no frame-relay lapf n201 [*bytes*]

Syntax Description

bytes	Maximum number of bytes in the Information field of the LAPF I frame, between 1 and 16384.

Defaults

260 bytes

Command Modes

Interface configuration

Command History

Release	Modification
11.2	This command was introduced.

Usage Guidelines

This command is used to tune Layer 2 system parameters to work well with the Frame Relay switch. Normally, you do not need to change the default setting.

Manipulation of Layer 2 parameters is not recommended if you do not know well the resulting functional change. For more information, refer to the ITU-T Q.922 specification for LAPF.

Examples

The following example resets the N201 maximum information field length to the default value:

```
no frame-relay lapf n201
```

frame-relay lapf t200

To set the Link Access Procedure for Frame Relay (LAPF) retransmission timer value T200, use the **frame-relay lapf t200** interface configuration command. To reset the T200 timer to the default value of 15, use the **no** form of this command.

frame-relay lapf t200 *tenths-of-a-second*

no frame-relay lapf t200

Syntax Description

tenths-of-a-second	Time, in tenths of a second, in the range from 1 through 100.

Defaults

15 tenths of a second (1.5 seconds)

Command Modes

Interface configuration

Command History

Release	Modification
11.2	This command was introduced.

Usage Guidelines

The retransmission timer value T200 should be less than the link idle timer value T203 (using the same time unit).

This command is used to tune Layer 2 system parameters to work well with the Frame Relay switch. Normally, you do not need to change the default setting.

Manipulation of Layer 2 parameters is not recommended if you do not know well the resulting functional change. For more information, refer to the ITU-T Q.922 specification for LAPF.

Examples

The following example resets the T200 timer to the default value:

```
no frame-relay lapf t200
```

Related Commands

Command	Description
frame-relay lapf t203	Sets the LAPF link idle timer value T203 of DLCI 0.

frame-relay lapf t203

To set the Link Access Procedure for Frame Relay (LAPF) link idle timer value T203 of data-link connection identifier (DLCI) 0, use the **frame-relay lapf t203** interface configuration command. To reset the link idle timer to the default value, use the **no** form of this command.

frame-relay lapf t203 *seconds*

no frame-relay lapf t203

Syntax Description

seconds	Maximum time allowed with no frames exchanged, in the range from 1 through 65535 seconds.

Defaults

30 seconds

Command Modes

Interface configuration

Command History

Release	Modification
11.2	This command was introduced.

Usage Guidelines

The **frame-relay lapf t203** command applies to the link; that is, it applies to DLCI 0. Circuits other than DLCI 0 are not affected.

The link idle timer value T203 should be greater than the retransmission timer value T200 (using the same time unit).

This command is used to tune Layer 2 system parameters to work well with the Frame Relay switch. Normally, you do not need to change the default setting.

Manipulation of Layer 2 parameters is not recommended if you do not know well the resulting functional change. For more information, refer to the ITU-T Q.922 specification for LAPF.

Examples

The following example resets the T203 idle link timer to the default value:

```
no frame-relay lapf t203
```