Chapter 14

Frame Relay

Introduction 14-3
Encapsulation 14-4
Data Link Connections 14-6
The Local Management Interface (LMI) 14-7
Logical Interfaces 14-8
Disabling or Resetting a Frame Relay Interface 14-8
Disabling or Resetting a Frame Relay DLC 14-9
Slow-Start Mechanism 14-9
Congestion Detection 14-10
    Detection using CLLM messages 14-10
    Detection using BECN bits 14-11
Congestion Control Scheme 14-12
Frame Relay on the Router 14-14
Encryption and Compression over Frame Relay 14-15
Troubleshooting Frame Relay Networks 14-15
Configuring Frame Relay 14-16
Configuration Examples 14-19
    Frame Relay without an LMI 14-19
    Frame Relay with an LMI 14-23
    Using Frame Relay Logical Interfaces 14-25
Command Reference 14-26
    add framerelay dlc 14-27
    add framerelay li 14-28
    create framerelay 14-29
    delete framerelay dlc 14-32
    delete framerelay li 14-32
    destroy framerelay 14-33
    disable framerelay 14-33
    disable framerelay congestioncontrol 14-34
    disable framerelay debug 14-34
    disable framerelay dlc 14-35
    disable framerelay dlc debug 14-36
    disable framerelay li debug 14-37
    disable framerelay slowstart 14-38
    enable framerelay 14-38
    enable framerelay congestioncontrol 14-39
    enable framerelay debug 14-40
    enable framerelay dlc 14-41
    enable framerelay dlc debug 14-42
    enable framerelay li debug 14-44
    enable framerelay slowstart 14-44

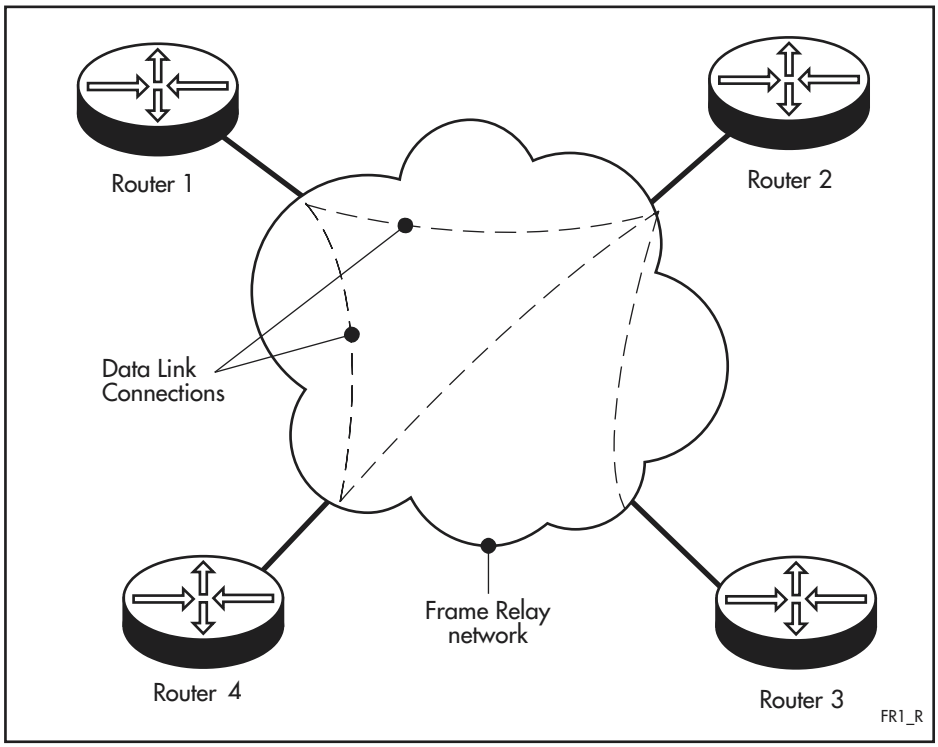
reset framerelay 14-45
reset framerelay dlc 14-46
set framerelay 14-47
set framerelay dlc 14-50
show framerelay 14-52
show framerelay dlc 14-62
show framerelay li 14-70

Introduction

This chapter describes the Frame Relay service provided by the router, and how to set up and use Frame Relay on the router. Frame Relay is available on routers with PIC bays only.

Frame Relay is a network service, defined by ITU-T (formerly CCITT), ANSI and vendor standards, to which routers may connect in order to communicate with one another and exchange data (Figure 14-1). Connections can be made via synchronous lines, ISDN calls or G.703 TDM (Time Division Multiplexing) links.

Figure 14-1: Model of a basic Frame Relay network



A Frame Relay network provides Data Link Connections (DLCs) between the routers connected to the network. These DLCs are set up by the Frame Relay network administration. One DLC is reserved for the communication of management information between the routers and the Frame Relay network, in a dialogue called the Local Management Interface (LMI).

Frame Relay itself exists purely as a way for frames to get from one router to another in an efficient manner. Frames sent to the network must contain the Data Link Connection Identifier (DLCI) of the DLC to use to deliver the frame. Except for LMI frames, the rest of the content of the frame is determined by the router-to-router communication and is not used by the network. In order for routers to transport multiple protocols across a single DLC the data being transmitted must be encapsulated to allow the remote router to identify the type of network protocol packet contained in the frame. A common standard for carrying multiple protocols over Frame Relay is specified by the IETF in RFC 1294. This standard is compatible with the ISO Frame Relay standards.

In many respects Frame Relay is similar to X.25, in that one physical link to the network carries many different streams of data. However, Frame Relay operates entirely in the lowest sublayer of the OSI data link layer, while X.25

operates at both the data link and network layers. Frame Relay assumes that the underlying physical medium is reliable and that errors (such as lost or corrupt packets) are handled by higher layer protocols such as IP. In comparison, X.25 assumes the underlying physical medium is not reliable and performs error checking and correction at each protocol layer and sublayer. Frame Relay is thus much simpler and more efficient than X.25, and can operate at much higher speeds.

The original CCITT Frame Relay standards were modified by a vendors' consortium in order to develop a working implementation. The standards were also altered in the process of becoming ANSI standards. These alterations mean that some aspects of the protocol, such as the DLCI used for the LMI, vary from implementation to implementation. Other features of Frame Relay that remain undefined or implementation dependent include the use of switched connections, the availability of broadcast or multicast DLCIs and the number of octets used to encode the DLCI in the frame.

Encapsulation

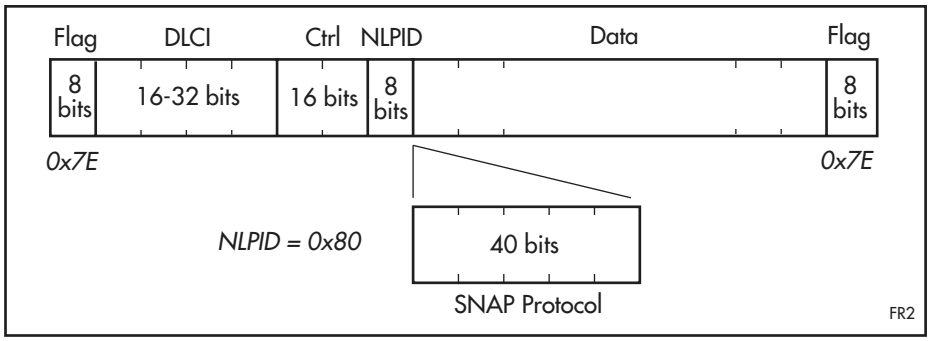
At the lowest level, Frame Relay is an example of an HDLC protocol but with the following features:

- Data comes in frames, delimited by special characters called flags.
- When a frame is not being sent, the sender transmits flags continually. This means that there is constant activity on any synchronous line that is running properly.
- The first bytes of data in a frame are interpreted as an address and a control field. The address may or may not have any special meaning, depending on the encapsulation, but the control field usually has some meaning.
- The data that follows the address and control fields is interpreted by the device receiving the frame depending on the encapsulation type.

The router supports two encapsulations: the IETF standard defined in RFC 1490, and a proprietary encapsulation defined by Cisco Systems.

The Frame Relay HDLC frame begins with a flag, an HDLC address field and an HDLC control field. The address field can be 2 to 4 octets long. The control field can be 1 or 2 octets long. RFC 1490 defines a 1-octet field following the address and control fields, known as the Network Layer Protocol ID (NLPID), which takes different values for different protocols (Figure 14-2).

Figure 14-2: Format of the Frame Relay HDLC frame with RFC 1490 encapsulation



The list of values and protocols assigned to them is administered by ISO and CCITT. Table 14-1 lists some examples of NLPIDs and Table 14-2 on page 14-5 lists the encapsulations supported by the router.

Table 14-1: Examples of assigned NLPID values for Frame Relay

NLPID	Meaning
0x80	Next 5 octets are protocol type (SNAP format)
0x81	ISO 8473
0x82	ISO ES-IS
0x83	ISO IS-IS
0xCE	Next 2 octets are Ethernet type field
0xCC	IP protocol

Table 14-2: Frame Relay encapsulations and NLPID values supported by the router

Protocol	NLPID	Next Octets (hexadecimal)
IP	0xcc	
Data Compression/Encryption	0xB0	
ARP	0x80	00-00-00-08-06
DECnet	0x80	00-00-00-60-03
IPX	0x80	00-00-00-81-37
Bridging	0x80	00-80-c2-00-01
Spanning tree	0x80	00-80-c2-00-0e
AppleTalk	0x80	08-00-07-80-9b

The router also supports a proprietary encapsulation defined by Cisco Systems. This encapsulation uses an Ethernet type field to identify the encapsulated protocol (Figure 14-3 and Table 14-3).

Figure 14-3: Format of the HDLC frame with Cisco proprietary encapsulation

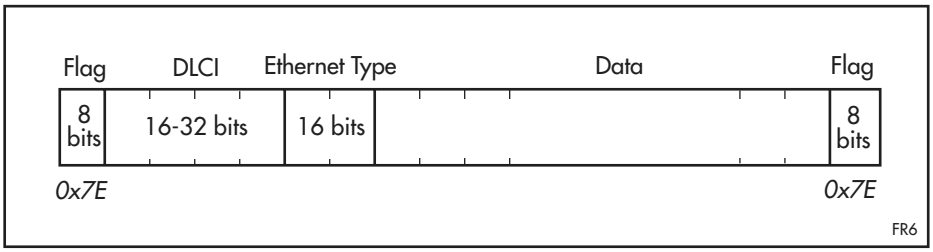


Table 14-3: Ethernet types supported for the Cisco proprietary encapsulation

Protocol	Ethernet Type (hexadecimal)
IP	0x0800
ARP	0x0806
DECnet	0x6003
IPX	0x8137

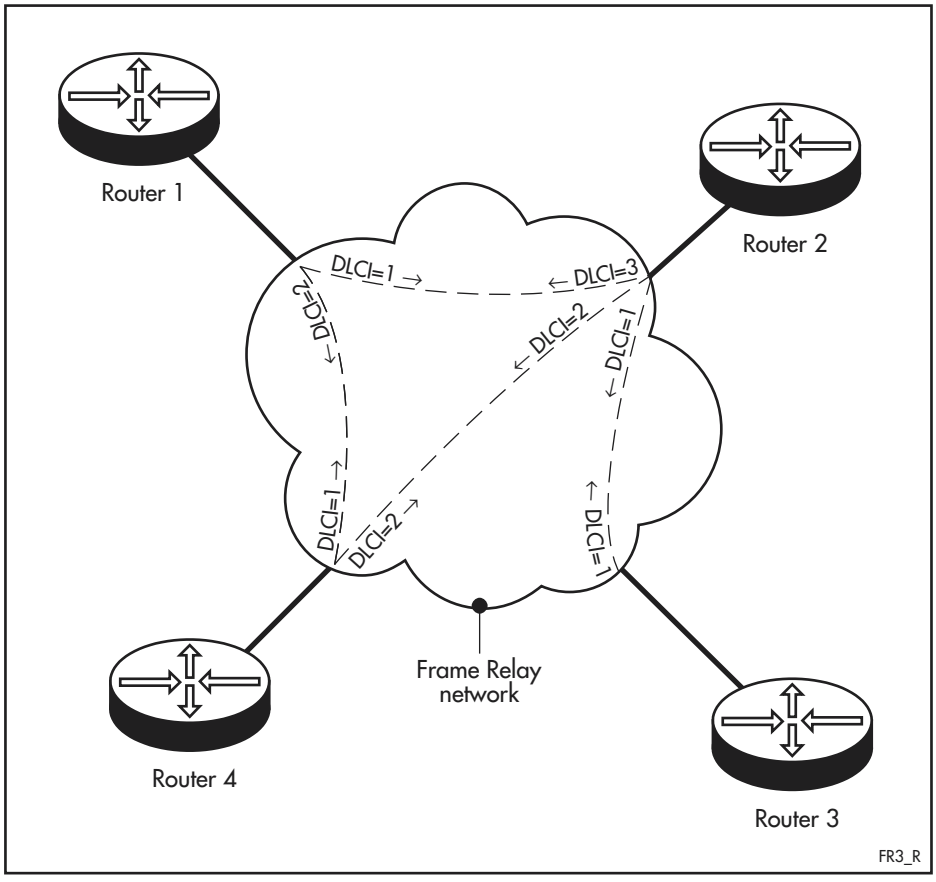
Data Link Connections

Each router connected to a Frame Relay network has a number of Data Link Connections (DLCs) to other routers. To send data to a particular router, a frame of data is sent to the network, with the Data Link Connection Identifier (DLCI) field set to the DLCI for the DLC on which the data is being transmitted. Each DLC from a given router has a different DLCI. The DLCIs for a given router have local significance only. Part of the work of the Frame Relay network is to change the DLCI in a frame of data so that the router receiving the frame sees the DLCI as the DLCI for the DLC from its point of view.

The following examples show some ways to assign DLCIs to DLCs.

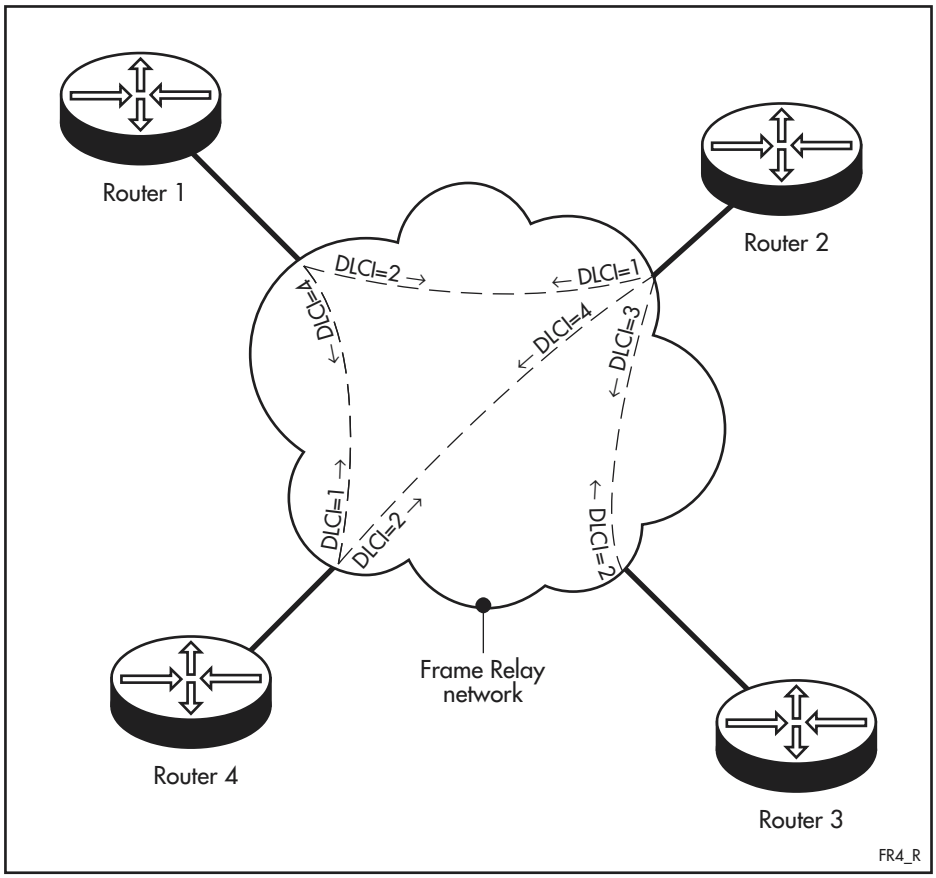
In Figure 14-4 the DLCIs have been chosen at random with the proviso that the DLCIs are locally unique for any given router. In this example, router 4 sends frames via DLC 2 to router 2, and router 2 receives them via DLC 2. However, router 1 sends frames via DLC 1 to router 2, but router 2 receives them via DLC 3.

Figure 14-4: Frame Relay network with randomly selected but unique DLCIs



In Figure 14-5 on page 14-7, the DLCs have been assigned DLCIs as if the DLCIs referred to the router rather than the actual DLC. In this example, routers send frames to router 2 via DLC 2, but router 2 receives the frames on DLCs 1, 3, and 4.

Figure 14-5: Frame Relay network with DLCIs as router identifiers.



The Local Management Interface (LMI)

A feature of Frame Relay is the dialogue that the network maintains with the routers connected to it. This dialogue is known as the Local Management Interface (LMI).

A special DLC is reserved for messages concerned with the LMI. Frames sent to the LMI DLC do not go to a remote router; they go to the network itself. The actual DLC that is used varies with the type of LMI supported by the Frame Relay routers in the network. The LMI defined by the vendors' consortium uses DLC 1023. The LMIs defined by the ANSI and Q.933 standards use DLC 0.

The router polls the network at regular intervals, and the network responds by informing the router about active DLCs. This ensures that the router and the Frame Relay network are always aware of each other's operational status. Frame Relay DLCs are set up on application to the network administration, and regular polling informs the router about new or relinquished DLCs.

However, an LMI is not provided by all Frame Relay networks. For networks that do not run an LMI, the switch can be configured to let static DLCs access the Frame Relay network.

Logical Interfaces

A Frame Relay interface exists over a physical interface in a one-to-one relationship. The Frame Relay interface provides a way for higher layers to attach to and use the DLCs provided by the Frame Relay network on that physical connection. In some cases it is useful for higher layers to be able to attach to a single DLC or to a subset of the DLCs provided by the Frame Relay network. This introduces the idea of Frame Relay Logical Interfaces (FRLIs).

An FRLI can be configured to contain a subset of the total DLCs on a physical interface. A number of FRLIs can be configured for each physical Frame Relay interface. By default a FRLI 0 is created for each physical Frame Relay interface that contains all the DLCs provided by the network. FRLIs must be created as Point-To-Point (PTP) interfaces when they are to contain only one DLC, or as Non-Broadcast Multiple Access (NBMA) interfaces when they are to contain multiple DLCs.

Disabling or Resetting a Frame Relay Interface

It is possible to disable or reset one of the router's Frame Relay interfaces. The disable action disables the entire interface. No routing modules can use the interface to receive or transmit data. The LMI dialogue is also halted if it was operational. If you halt the LMI dialogue, the Frame Relay network reports a fault for the router on which the interface was disabled.

The Frame Relay interface should be disabled infrequently, for example for a serious network problem and related debugging. To disable a Frame Relay interface, use the **disable framerelay** command on page 14-33.

To enable the interface again, use the **enable framerelay** command on page 14-38.

A Frame Relay interface can also be reset, which re-initialising it. When the LMI dialogue is active, default DLCs are removed from the list of DLCs and routing modules are informed of the loss. Information about the DLCs is restored when the next full status message in the LMI dialogue is received, and the DLCs are re-established as active. If the LMI dialogue is turned off, routing modules are informed of the loss of the DLCs, and are immediately informed when the DLCs are active again.

The interface should be reset infrequently, for example to change the LMI dialogue scheme. Resetting is sometimes necessary to activate various parameters. To reset a Frame Relay interface, use the **reset framerelay** command on page 14-45.

Disabling or Resetting a Frame Relay DLC

All DLCs in the Frame Relay network are under the administration of the controllers of the network. This is true whether an LMI dialogue is running or not. If the LMI dialogue is not running, DLCs are set up by adding or removing them as described above. In this case it is not necessary to disable a DLC; it can simply be removed.

If the LMI dialogue is running, the network informs the router about DLCs that are active. It may be desirable at times to explicitly disable a DLC that would otherwise be active. This means not sending any traffic to a particular DLC and throwing away traffic from that DLC. The network informs the router that the DLC in question is active but it is not used.

The routing modules using the Frame Relay interface are informed that the disabled DLC is no longer available and they do not try to send traffic down it.

To disable a DLC for a Frame Relay interface, use the **disable framerelay dlc** command on page 14-35.

To enable a previously disabled DLC, use the **enable framerelay dlc** command on page 14-41.

The Frame Relay network must have a DLC enabled for traffic to be sent and received. Enabling a DLC that the Frame Relay network does not know about does not make the DLC active. When the LMI dialogue is running, all DLCs that the Frame Relay network informs the router about are enabled by default. A DLC is typically disabled only for network problems and debugging.

To reset a DLC on a Frame Relay interface, use the **reset framerelay dlc** command on page 14-46.

This command resets MIB counters for the DLC. If encryption is enabled on the DLC, this command resets the ECP. If compression is enabled on the DLC without encryption, this command resets the CCP. The routing modules are informed of the loss of the DLC, and immediately informed of the appearance of the DLC again.

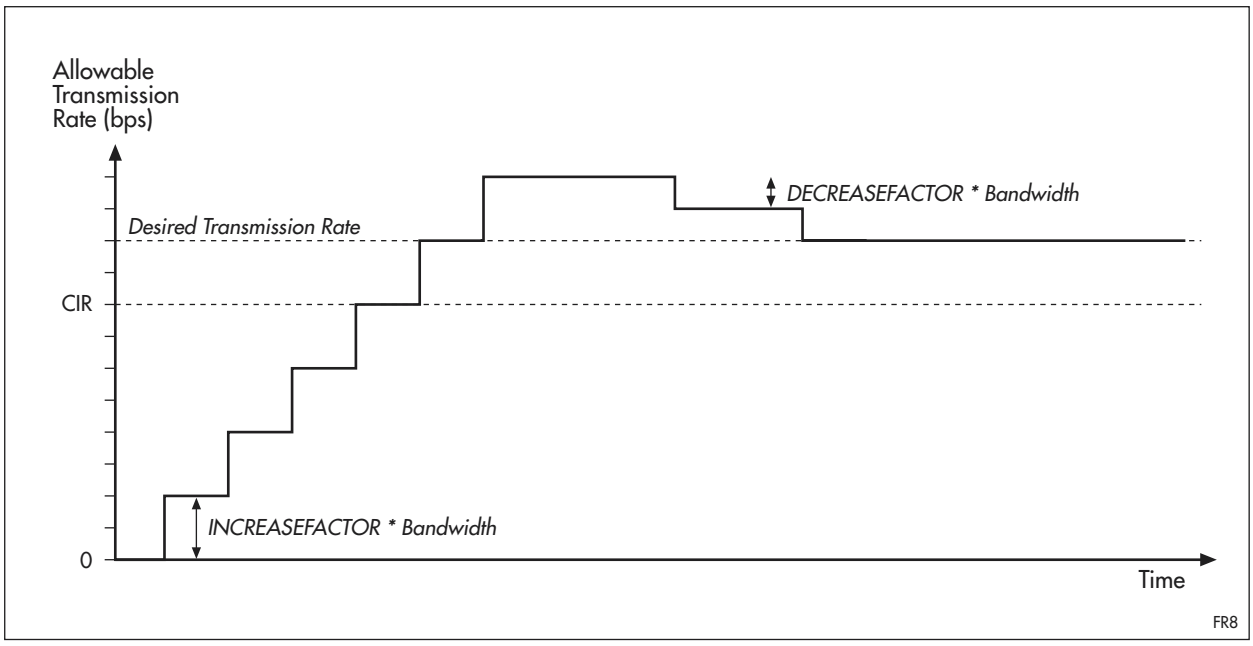
In all commands that affect a DLC, the DLC is specified by its DLCI. A feature of Frame Relay is that the DLCI for a given DLC is not necessarily the same for both routers on the DLC. Therefore, disabling or resetting a DLC from different ends may require different DLCIs.

Slow-Start Mechanism

When a router begins to transmit data into a Frame Relay network, the sudden increase in traffic may congest the network. To minimise the impact of increased network traffic and reduce the severity of congestion, a slow-start mechanism can be used. A slow-start mechanism works by limiting the rate at which data is transmitted into the network. When a DLC first becomes active the transmission rate for that DLC is gradually increased from a small value until it equals the rate at which data is being passed to the Frame Relay module by higher layers, or the bandwidth of the physical interface is reached, or congestion occurs.

Figure 14-6 on page 14-10 illustrates how the allowable transmission rate is increased by steps of *INCREASEFACTOR*bandwidth* and then drops by steps of *DECREASEFACTOR*bandwith* until it settles on a desired rate at which packets are passed to Frame Relay by the higher layers for transmission. The allowable transmission rate increases above the desired rate to clear packets that queued while the allowable transmission rate was below the desired rate.

Figure 14-6: The effect of the slow-start mechanism on transmission rates



Congestion Detection

When a Frame Relay network becomes congested, the network informs its routers about the problem so they can reduce traffic they send to the network. The network uses the following mechanisms to inform connected routers:

- **Detection using CLLM messages**
- **Detection using BECN bits**

Detection using CLLM messages

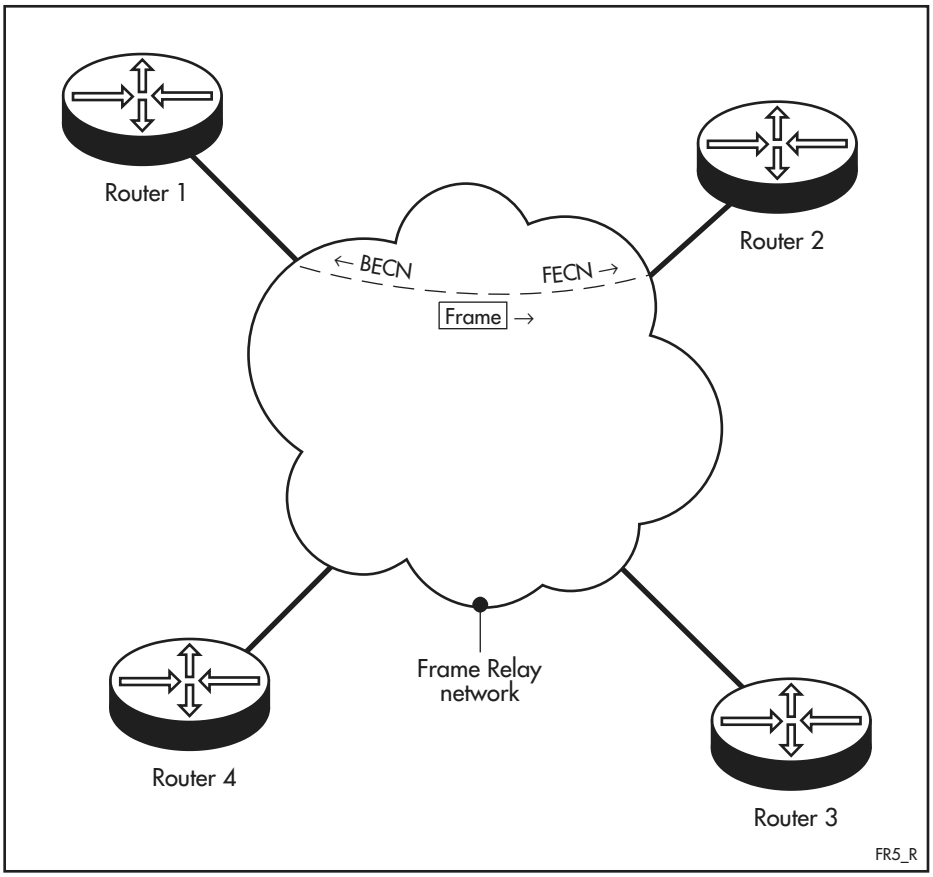
A Frame Relay network that supports the Q.933 standard can send the router CLLM messages on a special DLC with a DLCI value of 1007. These messages contain an indication of congestion in the Frame Relay network. The cause of the congestion is specified along with a list of DLCs that are congested due to the specified cause. The router must extract the congestion cause and then set the congestion state of each DLC specified in the CLLM message accordingly. While the network remains congested, a CLLM message is received every 10 seconds. If a CLLM message is not seen for 11 seconds, the network is assumed to be uncongested and the congestion state of the affected DLCs is set accordingly.

Detection using BECN bits

When a network becomes congested, it sets the BECN bit in frames passing through the congested parts of the network (Figure 14-7 on page 14-11). The router examines the BECN bit in each packet from the network, and for each DLC it counts the number of continuous packets received with the BECN bit set.

The first BECN bit detected in a sequence indicates *mild congestion* and the congestion state of the DLC is set accordingly. If the sequence of continuous BECN bits reaches a user-defined limit, set by the **becnlimit** parameter, the network is deemed to be *severely congested* and the congestion state of the DLC is set accordingly. For both mild and severe congestion, the network is deemed to be uncongested after a continuous sequence of **becnlimit**/2 packets is received with the BECN bit cleared or when the BECN timer expires This timer is started when the first packet with the BECN bit cleared is received.

Figure 14-7: Congestion notification in a Frame Relay network



Congestion Control Scheme

Routers detect congestion through a BECN bit set in data frames or through CLLM packets. When a router detects that its DLCs are congested, it limits traffic to let congestion clear. The router recognises the following causes from CLLM packets:

- mild congestion
- severe congestion
- fixed fault notification
- all frames discard notification

When mild congestion is detected, the allowable transmission rate of the DLC is limited to the DLC's Committed Information Rate (CIR). When severe congestion is detected, the allowable transmission rate is reduced to 0.675*CIR after an interval set with the CTICK parameter in the **create framerelay** command on page 14-29.

If severe congestion continues for another interval, the transmission rate is reduced to 0.5*CIR; and then after another interval, it is finally reduced to 0.25*CIR.

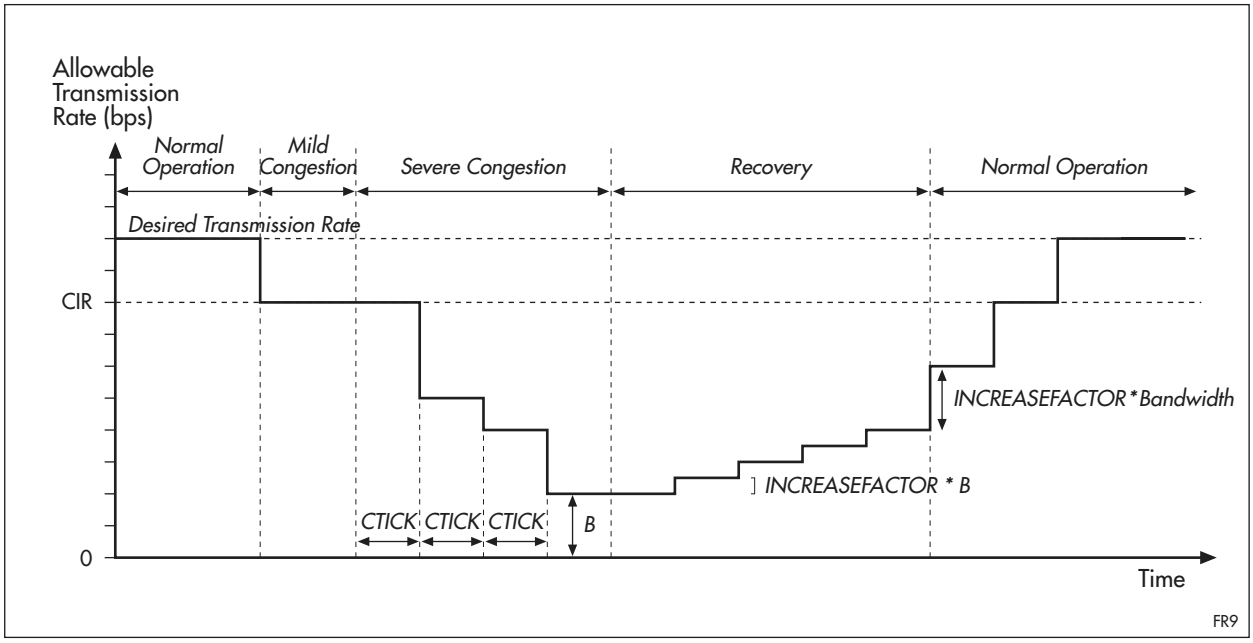
When a Fixed Fault notification or All Frames Discard notification is received, the router stops sending frames to the network entirely. Packets that higher layers want to pass to Frame Relay for transmission are first queued, and then discarded if the queue reaches its limit.

To permanently limit and control the rate at which data is transmitted onto a DLC, use the CIR Limiting function. Using CIR Limiting avoids both the overhead of detecting congestion and also the delay while routers back-off their transmission. To set CIR Limiting, use the **set framerelay dlc** command on page 14-50. Set **cirlimited** to **on** to set the maximum transmission rate of the DLC to the CIR of the DLC, regardless of congestion state of the network. Setting it to **off** means that the transmission rate is unlimited unless there is congestion on the network and congestion control is **on**. The default is **off**.

When the DLC becomes uncongested, the allowable transmission rate is increased at a controlled rate using the slow-start mechanism, if enabled. If slow-start is not enabled, the transmission rate becomes unrestricted immediately when the DLC becomes uncongested.

Figure 14-8 shows how the allowable transmission rate changes when congestion is detected with both congestion control and the slow-start mechanism enabled. It also illustrates how the configuration parameters relate to changes in the allowable transmission rate.

Figure 14-8: The effect of congestion control on the transmission rate into a congested Frame Relay network



Excess Information Rate (EIR) complements the CIR functionality. EIR allows the data transmission rate of a DLC to be set below the maximum link when the network is uncongested. The data transmission rate of a DLC is currently set below the link rate (to a value determined by CIR) if the DLC becomes congested, or if the **cirlimited** parameter is on (see the **set framerelay dlc** command on page 14-50). Having EIR allows the uncongested transmission rate of each DLC to be controlled.

Table 14-4 shows the transmission of a Frame Relay circuit depending on the parameters specified in the **set framerelay dlc** command on page 14-50.

Table 14-4: Transmission of a Frame Relay circuit depending on **set framerelay** parameters

EIR	CIR	Maximum Transmission Rate
NONE	NONE	MaxB Always
NONE	0 <= Bc<= MaxB-Be	C: Bc; NC: MaxB
0 <= Be <= MaxB-Bc	NONE	Be Always
0 <= Be <= MaxB-Bc	0 <= Bc<= MaxB-Be	C: Bc; NC: Bc+Be

Key
Bc = Committed Bit Rate
Be = Excess Bit Rate
MaxB = Maximum Bit Rate for the interface
C = Congested
NC = Not Congested

Frame Relay on the Router

The router supports Frame Relay as Data Terminal Equipment (DTE). The router can communicate with other routers through the Frame Relay network carrying multiple protocols (for example IP, IPX and DECnet Phase IV). The router implements the IETF specification for the transmission of protocols over Frame Relay (RFC 1490). The router supports both the IETF encapsulation and Cisco Systems proprietary encapsulation.

Frame Relay can be configured to use any synchronous interface on the router, although an X.21, V.35 interface is required for access speeds of greater than 64000 bits per second. Frame Relay can also be configured to use an ISDN call or a G.703 TDM group to access the Frame Relay network. See Chapter 11, Integrated Services Digital Network (ISDN) for details of how to configure ISDN and defined ISDN calls, and Chapter 12, Time Division Multiplexing (TDM) for information about configuring TDM groups. Any combination of synchronous interfaces, ISDN calls and TDM groups can be configured for Frame Relay, allowing multiple connections into one Frame Relay network, or connections into more than one Frame Relay network.

The router supports three variations of the LMI, the ANSI, and Q.933 standards, which both use DLC 0 for the LMI, and the vendor standard, which uses DLC 1023. The LMI can also be turned off entirely, requiring static DLCs to be set up.

The router provides a slow-start mechanism that gradually increases the rate that it transmits data to the network (see "Slow-Start Mechanism" on page 14-9). The slow-start mechanism is disabled by default. When disabled, packets are transmitted by the Frame Relay module at the same rate that they are received from higher layers. The slow-start mechanism can be enabled with the **enable framerelay slowstart** command on page 14-44.

The router provides support for congestion control (see "Congestion Control Scheme" on page 14-12). Congestion control is disabled by default; CLLM messages are still received and BECN bits are still counted however no action is taken to control the rate at which data is transmitted. Congestion control can be enabled with **enable framerelay congestioncontrol** command on page 14-39. The router does not set the DE bit for any frames being sent to the network.

The LMI parameters can be configured to control the frequency of the LMI poll and the criteria that the router uses to determine whether the Frame Relay network is operational or not. Frame Relay interfaces can be disabled if required for debugging purposes. Individual DLCs can also be disabled. DLCs may be statically configured if the router is required to operate with a Frame Relay network not running the LMI.

Any combination of routing protocols can attach to an entire Frame Relay interface but only IP can attach to a specific FRLI and thus a subset of the DLCs.

Full status information, by interface and by DLC is available via a manager port. SNMP support is added in a future release. The Frame Relay MIB as defined in RFC 1315 is supported. The information available includes counts of frames and octets for each DLC, time of last status change of a DLC, and the last LMI frame seen that contained an error.

Encryption and Compression over Frame Relay

The router can encrypt and/or compress Frame Relay data packets on a per-DLC basis using a protocol based on the Frame Relay Forum's "*Data Compression Over Frame Relay Implementation Agreement*", FRF.9, extended to include encryption support. DLCs with encryption and/or compression enabled must negotiate the use of encryption and/or compression with the destination router before the DLC becomes usable. This negotiation is carried out using the CCP and ECP as defined for PPP in RFCs 1962 and 1968 respectively. If encryption cannot be supported by the destination router, the DLC never becomes usable.

An encryption-capable MAC card must be installed in the router for encryption to be used on a DLC. Compression is available regardless of whether a compression-capable MAC card is installed.

Troubleshooting Frame Relay Networks

Debugging modes and counters are available to help analyse problems when they occur. Debugging modes can be enabled on an interface, logical interface, or DLC. Debugging modes for interfaces consist of LMI packet, LMI state, CLLM packet, and utilisation debugging. Only user debugging is available for logical interfaces. Debugging modes available for DLCs are:

- compression packet
- CCP packet
- encryption packet
- ECP packet
- IP packet
- IPX packet
- DECnet packet
- AppleTalk packet
- Bridge packet
- CCP state
- ECP state
- ENCO state
- PLLC state
- DLC state
- congestion state debugging

Enabling packet debug displays all packets transmitted and received for the type specified. Enabling state debugging displays state transitions for the state machine specified. Enabling utilisation debugging displays the utilisation of each DLC every five seconds.

To display counters for an entire Frame Relay interface, set the **counter** parameter in the **show framerelay** command on page 14-52.

To display counters for a single logical interface, use the command:

```
show framerelay li=n counter
```

To display counters for a single DLC, including counters for encryption and compression, use the command:

```
show framerelay dlc=n counter
```

Configuring Frame Relay

The following steps are necessary to configure Frame Relay:

1. Create a Frame Relay interface.
2. Define parameters that affect the LMI dialogue.
3. Define defaults for encryption and compression.
4. Add static DLCs as necessary.
5. Add logical interfaces as necessary.
6. Enable routing modules to use the interface.

Create a Frame Relay interface

To create the interface and associate it with a synchronous interface, or an ISDN call, use the command:

```
create framerelay=n over=physical-interface
```

In this command *n* is the number of the Frame Relay interface and *physical-interface* is a synchronous interface such as syn0 or an ISDN call such as isdn-Head Office.

To display each Frame Relay interface, the physical interface it uses, and the logical interfaces it provides, use the **show framerelay** command on page 14-52.

Define parameters that affect the LMI dialogue

Set the **lmischeme** parameter when you create the FR interface with the **create framerelay** command on page 14-29.

Parameters that affect the LMI dialogue, their values, and defaults are defined in the Frame Relay standards. Defaults are used when no LMI parameters are specified. We recommend that you consult your Frame Relay network provider before changing parameters that affect the LMI dialogue.

After creating the Frame Relay interface, you can change LMI parameters with the **set framerelay** command on page 14-47. Note that changes take effect immediately for the **defencapsulation**, **nt1**, **nn1**, **nn2**, and **nn3** parameters. When you change the other parameters, they automatically reset the FR interface before taking effect.

To display current values, use the **config** parameter in the **show framerelay** command on page 14-52.

Define defaults for encryption and compression

Set the **defencryption** and **defcompression** parameters to define defaults when you are creating the FR interface with the **create framerelay** command. These values are used by all DLCs on the interface unless you specifically override a particular one.

Add static DLCs as necessary

When the LMI dialogue is turned off, the router is not informed about active DLCs. In this case, set up static DLCs with the **add framerelay dlc** command on page 14-27. Static DLCs and the LMI dialogue are mutually exclusive.

Set encryption and compression for static DLCs when you add them with the **add framerelay dlc** command on page 14-27. If no values are specified, the interface defaults set with **defencryption** and **defcompression** are used for the DLC.

You can set encryption and compression parameters for an individual DLC with the **set framerelay dlc** command on page 14-50. When parameters of a DLC that the router has not been informed of via the LMI are changed from their defaults, a DLC is added to the interface. The DLC goes into a AWAIT_LMI state until the network informs the router via the LMI that the DLC is active.

The actual values to use for DLCs must be obtained from the administrators of the Frame Relay network where the router is connected. Communication across the Frame Relay network occurs for the DLCs that are statically configured.

Add logical interfaces as necessary

Frame Relay Logical Interfaces (FRLI) provide a mechanism for organising DLCs into groups. Each FRLI is a group of DLCs that you can assign its own IP address thereby splitting the Frame Relay network into subnets. A default FRLI 0 is always created when a Frame Relay interface is created and all DLCs are associated with it.

To create additional FRLIs, use the **add framerelay li** command on page 14-28.

To associate DLCs with other FRLIs, use the **li** parameter in the **set framerelay dlc** command on page 14-50.

Enable routing modules to use the interface

After a Frame Relay interface has been defined and configured, routing modules can be configured to use it. Configuration procedures are described in individual chapters for the particular routing module.

In general, commands that contain the INTERFACE= parameter can refer to a Frame Relay interface by name. The form of the name is "fr*n*" where *n* is the instance for the Frame Relay module. Examples of commands that can refer to a Frame Relay interface include:

```
add ip interface=frn...
add ipx circuit=circuit interface=frn...
set dnt add=interface interface=frn...
```

One important point concerning the use of Frame Relay interfaces by the IP routing module is the way that the IP routing module maps IP addresses to a Frame Relay DLCI and vice versa. This mapping is an example of Address Resolution Protocol (ARP). Two methods are supported on the router for Frame Relay interfaces – Inverse ARP and static ARP.

The router supports the Inverse ARP, a protocol specially developed for Frame Relay that involves the exchange of packets between routers connected by a DLC in order to map an IP address to a DLCI. Inverse ARP is described in RFC 1293.

To enable the router to communicate with DTEs that do not support Inverse ARP, add static ARP entries with the command:

```
add ip arp=ipadd interface=FRn dlci=dlci
```

For more information about adding static ARP information, see Chapter 22, Internet Protocol (IP).

Using static DLCs and static ARP information should not normally be required for interoperation of the router with other vendors' equipment. These facilities are provided for interoperation with equipment that does not fully support the Frame Relay standards. Networks that consist purely of routers that support Inverse ARP do not need static ARPs.

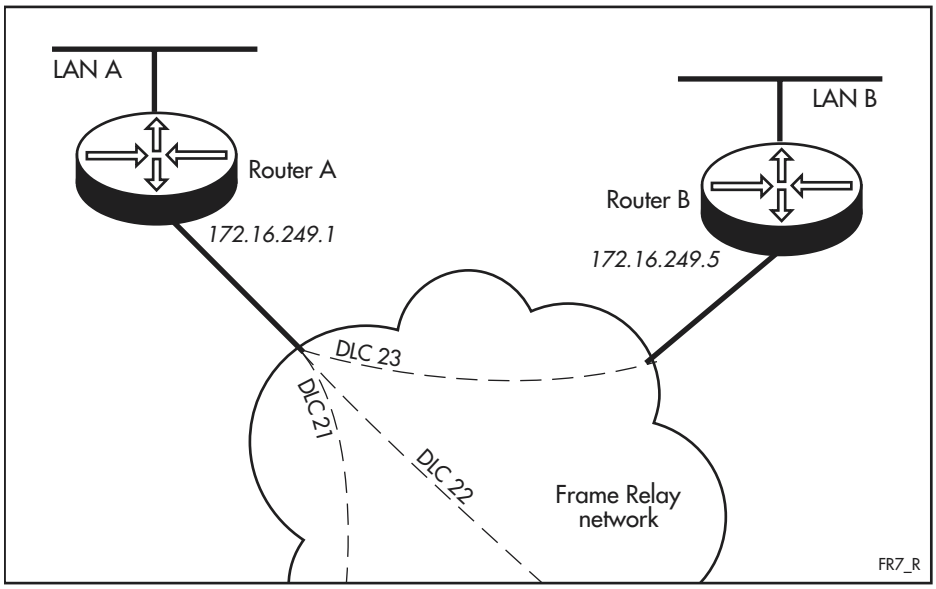
Configuration Examples

The following examples show how to configure a Frame Relay interface, starting with no Frame Relay interface set up, and continuing until it is configured with the IP routing module running over the interface. Finally, a change to the LMI scheme is made. The use of FRLIs is also shown.

Frame Relay without an LMI

This example shows how to configure a Frame Relay network that does not support an LMI dialogue; therefore, static DLCs must be configured. One of the DLCs connects to a router that does not support Inverse ARP so a static ARP entry must be created. Only the steps for Router A are described. Router B is assumed to be a router from another vendor and under the control of a different network administrator (Figure 14-9 on page 14-19).

Figure 14-9: Example configuration for a Frame Relay network without an LMI



To configure Frame Relay without an LMI

1. **Create the Frame Relay interface.**

   To create an interface over synchronous port 0, use the command:

   ```
   create framerelay=0 over=syn0
   ```

   To use this interface in higher layer commands, use “fr0”, and you may also need to specify the DLCI.

   To check the configuration, use the command:

   ```
   show framerelay config
   ```

   The output is shown in Figure 14-10 on page 14-20. All LMI parameters have taken their defaults. The default LMI scheme is LMI Revision 1, so DLC 1023 has been created for the LMI dialogue.

Figure 14-10: Example output from the **show framerelay config** command for a Frame Relay interface created with default parameters

```
Interface
  Parameter                                   Value
-------------------------------------------------------
fr0
  Over ................................... syn0
  Enabled ................................ Yes
  ifIndex ................................ 4
  Logical Interfaces ..................... 0
  LMI Scheme ............................. LMI revision 1
  Polling Interval (NT1) (secs) ......... 10
  Full Enquiry Interval (NT1*NN1) (secs)  60
  Error Threshold (NN2) .................. 2
  Monitored Events (NN3) ................. 4
  Default encapsulation .................. IETF
  Default encryption ..................... Off
  Default compression .................... Off
  Slow Start ............................. Disabled
  Increase Factor ........................ 0.05
  Decrease Factor ........................ 0.025
  Dropoff Threshold (%) .................. 70
  Maximum Transmission Queue Length ..... 100
  Congestion Control ..................... Disabled
  Congestion Detection Method ........... BECN and CLLM
  Continuous BECN Limit .................. 100
  BECN Timeout (secs) .................... 10
  Congestion Tick (secs) ................. 10
  Congestion Recovery Time (secs) ....... 60
-------------------------------------------------------
```

2. Configure a LMI.

The Frame Relay interface runs without an LMI dialogue because the vendor's hardware does not support LMI. Disable the LMI dialogue with the commands:

```
set framerelay=0 lmi=none
reset framerelay=0
```

3. Create static DLCs.

Create three DLCs numbered 21, 22, and 23, and check the configuration:

```
add framerelay=0 dlc=21
add framerelay=0 dlc=22
add framerelay=0 dlc=23
show framerelay config
show framerelay dlc
```

Figure 14-11: Example output from the **show framerelay config** and **show framerelay dlc** commands for a Frame Relay interface configured with three DLCIs and no LMI

```
Interface
  Parameter                                    Value
-------------------------------------------------------
fr0
  Over ..................................... syn0
  Enabled .................................. Yes
  ifIndex .................................. 4
  Logical Interfaces ....................... 0
  LMI Scheme ............................... None
  Polling Interval (NT1) (secs) ............ 10
  Full Enquiry Interval (NT1*NN1) (secs) . 60
  Error Threshold (NN2) .................... 2
  Monitored Events (NN3) ................... 4
  Default encapsulation .................... IETF
  Default encryption ....................... Off
  Default compression ...................... Off
  Slow Start ............................... Disabled
  Increase Factor .......................... 0.05
  Decrease Factor .......................... 0.025
  Dropoff Threshold (%) .................... 70
  Maximum Transmission Queue Length ....... 100
  Congestion Control ....................... Disabled
  Congestion Detection Method .............. BECN and CLLM
  Continuous BECN Limit .................... 100
  BECN Timeout (secs) ...................... 10
  Congestion Tick (secs) ................... 10
  Congestion Recovery Time (secs) ......... 60
-------------------------------------------------------

Interface  DLC   LI   State           ECP       CCP
-----------------------------------------------------------
fr0        0021  0    Active          -         -
           0022  0    Active          -         -
           0023  0    Active          -         -
-----------------------------------------------------------
```

4. Configure the IP routing module to use the Frame Relay interface.

IP is to be run over the Frame Relay interface. The interface requires one IP address, which is to be 172.16.249.1, with a subnet mask of 255.255.255.0. The routers at the other end of the DLCs with DLCI 21 and 22 are routers that support Inverse ARP, so Inverse ARP runs and sets the IP addresses for those routers. The router accessed via DLC 23 does not support Inverse ARP, so a static ARP is configured. The IP address of this router is 172.16.249.5. The commands are:

```
enable ip

add ip INT=fr0 ip=172.16.249.1 mask=255.255.255.0

add ip arp=172.16.249.5 int=fr0 dlci=23
```

To see that the interface is set up correctly, use the **show ip interface** command on page 22-202 of Chapter 22, Internet Protocol (IP).

Figure 14-12: Example output from the **show ip interface** command for a Frame Relay interface

```
Interface     Type     IP Address       Bc Fr PArp  Filt RIP Met.   SAMode  IPSc
Pri. Filt     Pol.Filt Network Mask     MTU   VJC   GRE  OSPF Met.  DBcast  Mul.
-------------------------------------------------------------------------------
Local         ---      Not set          1  n  -     ---  --         Pass    --
---           ---      Not set          1500  -     ---  --         ---     ---
fr0           Static   172.16.249.1     1  n  -     ---  01         Pass    --
---           ---      255.255.255.0    1500  -     ---  0000000001 No      Rec
-------------------------------------------------------------------------------
```

Use the **show framerelay li** command on page 14-70 to check that the Frame Relay interface is being used by the IP interface (Figure 14-13). The Frame Relay interface should have IP and ARP as user modules.

Figure 14-13: Example output from the **show framerelay li** command for a Frame Relay interface with IP and ARP user modules

```
Interface   LI    IfIndex   Type   Number of DLCs   User Modules
-----------------------------------------------------------------
fr0         All                                   IP
                                                  ARP
            0     05        NBMA   3                None
-----------------------------------------------------------------
```

The static ARP can be checked with the command:

```
show ip arp
```

Other ARP information comes from normal Ethernet traffic (Figure 14-14 on page 14-22). The router is now ready to use the Frame Relay network.

Figure 14-14: Example output from the **show ip arp** command for a Frame Relay interface configured with a static ARP

```
 Interface    IP Address       Physical Address      ARP Type    Status
-------------------------------------------------------------------------
 eth0         172.16.8.2       AA-00-04-00-28-08     Dynamic     Active
 eth0         172.16.8.34      00-00-0C-02-5A-0A     Dynamic     Active
 eth0         172.16.9.198     00-00-E8-01-1F-B0     Dynamic     Active
 fr0          172.16.249.5     0023                  Static      Active
-------------------------------------------------------------------------
```

Frame Relay with an LMI

This example shows how to configure Frame Relay for connection to a Frame Relay network that supports an LMI dialogue, using ANSI standard T1.617, Annex D.

To configure Frame Relay with an LMI

1. **Create the Frame Relay interface.**

   The Frame Relay interface uses synchronous port 0 of the router. The interface is the first Frame Relay interface, so it is named "fr0":

   ```
   create fr=0 over=syn0
   ```

   The configuration can be checked with the command:

   ```
   show fr config
   ```

2. **Configure the LMI.**

   The network administration want to see a STATUS ENQUIRY message from the router every 20 seconds. Updates of active DLCs are required as often as possible given the 20 second poll time. The alarm condition values do not need to be changed. The commands are:

   ```
   set fr=0 lmi=annexd nt1=20 NN1=1
   reset fr=0
   show fr config
   show fr dlc
   ```

   The LMI scheme is shown as Annex D and DLC 0, the DLCI reserved for the LMI dialogue, is enabled (Figure 14-15 on page 14-24).

Figure 14-15: Example output from the **show framerelay config** and **show framerelay dlc** commands for a Frame Relay interface configured to use Q.933 LMI

```
Interface
  Parameter                                          Value
-------------------------------------------------------------
fr0
  Over ..................................... syn0
  Enabled ..................................... Yes
  ifIndex ..................................... 4
  Logical Interfaces .......................... 0
  LMI Scheme .................................. Annex D
  Polling Interval (NT1) (secs) ............... 20
  Full Enquiry Interval (NT1*NN1) (secs) ...... 20
  Error Threshold (NN2) ....................... 2
  Monitored Events (NN3) ...................... 4
  Default encapsulation ....................... IETF
  Default encryption .......................... Off
  Default compression ......................... Off
  Slow Start .................................. Disabled
  Increase Factor ............................. 0.05
  Decrease Factor ............................. 0.025
  Dropoff Threshold (%) ....................... 70
  Maximum Transmission Queue Length ........... 100
  Congestion Control .......................... Disabled
  Congestion Detection Method ................. BECN and CLLM
  Continuous BECN Limit ....................... 100
  BECN Timeout (secs) ......................... 10
  Congestion Tick (secs) ...................... 10
  Congestion Recovery Time (secs) ............. 60
-------------------------------------------------------------

Interface  DLC   LI   State               ECP       CCP
-------------------------------------------------------------
fr0        0000  0    Active              -         -
```

3. Configure the IP routing module to use the Frame Relay interface.

IP is to be run over the Frame Relay interface. The interface requires one IP address, which is to be 172.16.249.1, with a subnet mask of 255.255.255.0. The routers at the other end of the DLCs with DLCI 21 and 22 are routers that support Inverse ARP, so Inverse ARP runs and sets the IP addresses for those routers. The commands are:

```
enable ip
add ip int=fr0 ip=172.16.249.1 mask=255.255.255.0
```

Check that the interface has been set up correctly, using the command:

```
show ip interface
```

Check that the Frame Relay interface is being used by the IP interface. The Frame Relay interface should have IP and ARP as user modules:

```
show fr li
```

Using Frame Relay Logical Interfaces

The FRLI feature allows Frame Relay DLCs to be separated into groups belonging to different FRLIs. These FRLIs can then be assigned different IP addresses making it possible to split the Frame Relay network into separate IP subnets. A typical example would be a Frame Relay network provider who is providing a value-added service by supporting routers at a customer's sites. Management of the routers is carried out by providing a separate DLC from the service provider's site to each customer site. The DLC from the service provider to the customer needs to have a separate IP address so it forms an entirely distinct IP subnet.

To configure two logical interfaces

1. **Create the physical Frame Relay interface.**

   Create a physical Frame Relay interface 0 over synchronous port 0:

   ```
   create fr=0 over=syn0
   ```

2. **Add the second logical interface.**

   A default logical interface (FRLI 0) is always created when a Frame Relay interface is created. Another FRLI must be added to the Frame Relay interface to provide the two FRLIs over which to create the IP interfaces.

   ```
   add fr=0 li=1
   ```

3. **Associate DLCs with the logical interfaces.**

   By default all the DLCs for the interface are associated with FRLI 0. In this example there are five DLCs – 101, 102, 103, 104, and 105. DLC 105 is to be associated with FRLI 0. The other DLCs are to be associated with FRLI 1. To associate these DLCs with FRLI 1, use the commands:

   ```
   set fr=0 dlc=101 li=1
   set fr=0 dlc=102 li=1
   set fr=0 dlc=103 li=1
   set fr=0 dlc=104 li=1
   ```

4. **Create IP interfaces to use the Frame Relay logical interfaces.**

   Two IP interfaces are to be created with separate IP addresses, to use the two FRLIs:

   ```
   add ip int=fr0.0 ip=192.168.10.1
   add ip int=fr0.1 ip=192.168.50.1
   ```

Command Reference

This section describes the commands available on the router to configure and manage Frame Relay. Frame Relay can be used on synchronous, ISDN and G.703 TDM interfaces. See Chapter 9, Interfaces for details of the commands required to configure synchronous and interfaces. See Chapter 11, Integrated Services Digital Network (ISDN) for details of the commands required to configure ISDN interfaces. See Chapter 12, Time Division Multiplexing (TDM) for details of the commands required to configure TDM interfaces.

The shortest valid command is denoted by capital letters in the Syntax section. See "Conventions" on page lxv of About this Software Reference in the front of this manual for details of the conventions used to describe command syntax. See Appendix A, Messages for a complete list of messages and their meanings.

add framerelay dlc

Syntax

```
ADD FRamerelay=fr-interface DLC=dlci
   [COMPression={DEFault|ON|OFF}] [ENCapsulation={DEFault|
   IETF|CISCO}] [ENCryption={DEFault|ON|OFF}]
```

where:

- *fr-interface* is the number of the Frame Relay interface.
- *dlci* is the identification number of a Frame Relay Data Link Connection (DLC) from 0 to 1023.

Description

This command adds a static DLC (also known as a circuit) for use by the Frame Relay interface. The DLC must not already exist. Static DLCs must be defined if the LMI is disabled. If the LMI is enabled any static DLCs are ignored. The DLC is automatically enabled to receive or transmit data, and routing modules are informed that the DLC is active.

The **encapsulation** parameter specifies the type of encapsulation to use for transmission of packets over the DLC. If **ietf** is specified, the IETF encapsulation defined in RFC 1490 is used. If **cisco** is specified, the Cisco Systems proprietary encapsulation is used. If **default** is specified, the interface encapsulation default is used for the DLC.

The **encryption** parameter enables or disables encryption on the DLC. This parameter requires a user with Security Officer privilege when the router is in security mode. The **compression** parameter enables or disables compression on the DLC. If **default** is specified for either of the parameters, the interface default is used.

Examples

To add a static DLC with a DLCI of 23 to Frame Relay interface 0, and explicitly enable both encryption and compression on the DLC, use the command:

```
add fr=0 dlc=23 enc=on comp=on
```

Related Commands

add framerelay li
create framerelay
delete framerelay dlc
disable framerelay dlc
disable framerelay dlc debug
enable framerelay dlc
enable framerelay dlc debug
reset framerelay dlc
set framerelay dlc
show framerelay

add framerelay li

Syntax

```
ADD FRamerelay=fr-interface LI=logical-interface
   [TYPe={NBMA|PTP}]
```

where:

- *fr-interface* is the number of the Frame Relay interface.
- *logical-interface* is the number of the logical interface from 1 to 1023.

Description

This command adds a logical interface to a Frame Relay interface.

The **type** parameter specifies whether the FRLI is **ptp** or **nbma**. If **nbma** is specified, then any number of DLCs can be associated with the FRLI. If **ptp** is specified, then only one DLC can be associated with the FRLI. The default is **nbma**.

Examples

To add a new LI with an identifier of 3 to Frame Relay interface 0 use the command:

```
add fr=0 li=3
```

Related Commands

add framerelay dlc
create framerelay
delete framerelay li
disable framerelay li debug
enable framerelay li debug
show framerelay li

create framerelay

Syntax

```
CREate FRamerelay=fr-interface OVER=physical-interface
   [DEFATencap={Appleoui|Ietfoui}]
   [BECNLimit=2..4294967295] [BECNTimeout=1..4294967295]
   [CDETection={BECN|CLLM|BOTH}] [CTICK=1..60]
   [DECreasefactor={0.025|0.05|0.1|0.2}]
   [DEFCcompression={ON|OFF}] [DEFENCApsulation={IETF|
   CISCO}] [DEFEncryption={ON|OFF}]
   [DROPOFFthreshold=1..100] [INCreasefactor={0.05|0.1|
   0.2}] [LMIscheme={LMIREV1|Q933A|ANNEXA|T1617B|ANNEXB|
   T1617D|ANNEXD|NONE}] [MAXTXQlength=1..4294967295]
   [NN1=nn1] [NN2=nn2] [NN3=nn3] [NT1=nt1]
   [RECoverytime=1..4294967295]
```

where:

- *fr-interface* is the number of the Frame Relay interface.
- *physical-interface* is the physical interface to use, SYN*n*, TDM-*groupname* or ISDN-*callname*.
- *nn1* is the value of the NN1 parameter.
- *nn2* is the value of the NN2 parameter.
- *nn3* is the value of the NN3 parameter.
- *nt1* is the value of the NT1 parameter.

Description

This command creates a Frame Relay interface running over a synchronous port, a TDM group or an ISDN call (referred to as the physical interface), and optionally sets operational parameters for the interface. The interface must not already exist. If the physical interface is a TDM group or an ISDN call, the group or call must have been defined previously with the **create tdm** command on page 12-9 of Chapter 12, Time Division Multiplexing (TDM) or the **add isdn call** command on page 11-64 of Chapter 11, Integrated Services Digital Network (ISDN). Operational parameters can be set or changed with the **set framerelay** command on page 14-47 after the interface is created.

This command requires a user with security officer privilege when the router is in security mode. Any or all of the operational parameters may be specified. These parameters and their values are defined in the Frame Relay standards. Defaults for the LMI parameters are also defined in the standards but can be altered to reflect requirements of the Frame Relay network.

The **defatencap** parameter specifies the default AppleTalk packet encapsulation (OUI SNAP encapsulation) that is applied to the packet before it is sent over the Frame Relay interface. **appleoui** specifies SNAP encapsulation with Apple OUI and **ietfoui** specifies SNAP encapsulation with IETF OUI. The default is **appleoui**.

The **becnlimit** parameter specifies the number of continuous frames that can be received with the BECN bit set before the DLC is deemed to be severely congested. The default is **100**.

The **becntimeout** parameter specifies the maximum time, in seconds, that a DLC remains congested after a packet with the BECN bit cleared is received, if congestion is being detected using BECN bits. The default is **10**.

The **cdetection** parameter specifies the method used to detect congestion on the Frame Relay interface. If **becn** is specified, congestion is detected using only BECN bits in received packets. If **cllm** is specified, congestion is detected using only CLLM messages. If **both** is specified, both CLLM messages and BECN bits are used. The default is **both**.

The **ctick** parameter specifies the period of the congestion timer. The congestion timer is started when a DLC becomes severely congested. When it expires the first time, the allowable transmission rate of the DLC is reduced to 0.625*CIR. When it expires the second time, throughput is reduced to 0.5*CIR. After the third expiry, throughput drops to 0.25*CIR and the timer is not started again. The default is 10 seconds.

The **decreasefactor** parameter specifies the fractional amount by which to decrease the allowable transmission rate each second if the actual transmission rate in the last second was below the **dropoffthreshold** percentage of the allowable transmission rate. The default is **0.025**.

The **defencryption** and **defcompression** parameters set the default encryption and compression values respectively for all DLCs in the interface. The default is **off** for both. This parameter requires a user with Security Officer privilege when the router is in security mode.

The **defencapsulation** parameter specifies the default protocol encapsulation used for all DLCs in the Frame Relay interface. If **ietf** is specified, the IETF encapsulation defined in RFC 1490 is used. If **cisco** is specified, the Cisco Systems proprietary encapsulation is used. The default is **ietf**.

The **dropoffthreshold** parameter specifies the percentage of the allowable transmission rate that the actual transmission rate must stay above for the allowable transmission rate to not be reduced. If the actual transmission rate drops below the **dropoffthreshold** percentage of the allowable transmission rate, then the allowable transmission rate is reduced. The default is 70%.

The **increasefactor** parameter specifies the fractional amount by which to increase the allowable transmission rate each second if there is data to send. The default is **0.05**.

The **lmischeme** parameter specifies the scheme used for the LMI, the dialogue between the router and the Frame Relay network. **lmischeme** can be one of **lmirev1, q933a, annexa, t1617b, annexb, t1617d, annexd** or **none. annexa, annexb** and **annexd** are synonyms for **q933a**, **t1617b** and **t1617d** respectively. **lmirev1** is the initial standard created by the vendors' consortium that first produced Frame Relay products. **annexa** refers to Annex A of the Q.933 ITU Recommendation. **annexb** and **annexd** refer to the Annexes of the ANSI standard for Frame Relay. **none** is used to indicate that the LMI is turned off altogether. The default is **lmirev1**.

The **maxtxqlength** parameter specifies the maximum length for the transmission queue of the Frame Relay interface. The default is **100**.

The **nn1** parameter specifies the ratio of normal status enquiry messages to full status enquiry messages. The router uses full status enquiry messages to request a full status message from the network. The full status message is the message that contains the status of all DLCs known to the network. Since the status of DLCs is not expected to change frequently and the full status message is longer than the normal status message, it is usual to set **nn1** to a number greater than 1 to ensure that excess bandwidth is not devoted to LMI messages. The value *nn1* must be a number between 1 and 255 inclusive, and defaults to **6**.

The **nn2** parameter specifies the number of error events in the last **nn3** events that flag an alarm condition. An event in this context is the reception of a reply to the status enquiry message sent by the router. An error event is a timeout before this message is received, or a reply being received with an incorrect format. The value *nn2* must be a number between 1 and 10 inclusive, and must be less than or equal to the value of NN3. The default for **nn2** is **2**.

The **nn3** parameter specifies the number of events used to determine alarm conditions and the negation of the alarm condition. If **nn2** out of the last **nn3** events are in error, an alarm is set. The alarm is cleared again when NN3 events are received without error. The value *nn3* must be a number between 1 and 10 inclusive, and must be greater than or equal to the value of **nn2**. The default for **nn3** is **4**.

nn2 and **nn3** are dependent on each other. Any value may be specified for **nn2** or **nn3** as long as the value of **nn2** is less than or equal to the value of **nn3** when at the end of processing the command line.

The **nt1** parameter specifies the seconds in the interval between status enquiry messages that the router transmits to the Frame Relay network. The status enquiry message is the basic poll message of the LMI dialogue. The value *nt1* must be a multiple of 5 and must fall between 5 and 30 inclusive. The default for **nt1** is 10. The change to the value of NT1 takes place immediately, although the effect only takes place after the next status enquiry message is sent to the network.

The values of **nt1** and **nn1** combine to determine how often the router sends full status enquiry messages to the network. The minimum value for the period between full status enquiry messages is 5 seconds and the maximum value is 7650 seconds (2 hours, 7 minutes, 30 seconds).

The **recoverytime** parameter specifies the time over which the DLC is deemed to be recovering from severe congestion. The allowable transmission rate is increased during this time by the **increasefactor** value * the lowest allowable transmission rate reached during severe congestion rather than by the **increasefactor** value * bandwidth. After the recovery time passes, the rate is increased by **increasefactor** value * bandwidth again. The default is 20 seconds.

Examples

To create Frame Relay interface 0 over the ISDN called HeadOffice, and enable both encryption and compression by default on all DLCs, use the command:

```
cre fr=0 over=isdn-headoffice defe=on defc=on
```

Related Commands

add framerelay dlc
add framerelay li
destroy framerelay
disable framerelay
enable framerelay
reset framerelay
set framerelay
show framerelay

delete framerelay dlc

Syntax

```
DELete FRamerelay=fr-interface DLC=dlci
```

where:

- *fr-interface* is the number of the Frame Relay interface.
- *dlci* is the identification number of a Frame Relay Data Link Connection (DLC) from 0 to 1023.

Description

This command permanently removes a static DLC (also known as a circuit) from use by the Frame Relay interface. The DLC must already exist. Static DLCs must be defined if the LMI is disabled. If the LMI is enabled any static DLCs are ignored.

Examples

To delete the static DLC with a DLCI of 23 from Frame Relay interface 0, use the command:

```
del fr=0 dlc=23
```

Related Commands

add framerelay dlc
disable framerelay dlc
disable framerelay dlc debug
enable framerelay dlc
enable framerelay dlc debug
reset framerelay dlc
set framerelay dlc
show framerelay dlc

delete framerelay li

Syntax

```
DELete FRamerelay=fr-interface LI=logical-interface
```

where:

- *fr-interface* is the number of the Frame Relay interface.
- *logical-interface* is the number of the logical interface from 1 to 1023.

Description

This command deletes a logical interface from a Frame Relay interface. A logical interface can only be deleted if there are no routing modules currently attached.

Examples

To delete LI 3 from Frame Relay interface 0, use the command:

```
del fr=0 li=3
```

Related Commands

add framerelay li
disable framerelay li debug
enable framerelay li debug
show framerelay li

destroy framerelay

Syntax `DESTroy FRamerelay=`*`fr-interface`*

where *fr-interface* is the number of the Frame Relay interface

Description This command destroys a Frame Relay interface. The interface must already exist. The interface is no longer available for use by routing modules.

Examples To destroy Frame Relay interface 0, use the command:

```
dest fr=0
```

Related Commands
create framerelay
disable framerelay
enable framerelay
reset framerelay
set framerelay
show framerelay

disable framerelay

Syntax `DISable FRamerelay=`*`fr-interface`*

where *fr-interface* is the number of the Frame Relay interface

Description This command disables a Frame Relay interface. No data is transmitted or received via the interface. If the interface is already disabled this command has no effect.

Examples To disable Frame Relay interface 2, use the command:

```
dis fr=2
```

Related Commands
create framerelay
destroy framerelay
enable framerelay
reset framerelay
set framerelay
show framerelay

disable framerelay congestioncontrol

Syntax

```
DISable FRamerelay=fr-interface CONgestioncontrol
```

where *fr-interface* is the number of the Frame Relay interface

Description

This command disables Frame Relay congestion control on the specified interface so that congestion is not detected and data transmission is not limited.

Examples

To disable congestion control on Frame Relay interface 0, use the command:

```
dis fr=0 con
```

Related Commands

disable framerelay debug
enable framerelay congestioncontrol
show framerelay

disable framerelay debug

Syntax

```
DISable FRamerelay=fr-interface DEBug={ALL|CLLMPkt|
   LMIState|LMIPkt|UTILisation}
```

where *fr-interface* is the number of the Frame Relay interface

Description

This command disables Frame Relay interface debug modes.

The **debug** parameter specifies the type of debug mode to disable. If **all** is specified, all currently enabled debug modes are disabled on the interface.

cllmpkt debug mode decodes and displays CLLM packets that are received by the router.

lmistate debug mode displays the state changes in the LMI state machine that occur when LMI packets are received.

lmipkt debug mode decodes and displays LMI packets that are received and transmitted by the router.

utilisation debug mode displays the utilisation of each DLC and the total utilisation of the interface once per second.

Examples

To disable **lmipkt** debug mode on Frame Relay interface 0, use the command:

```
dis fr=0 deb=lmip
```

Related Commands

disable framerelay
disable framerelay dlc debug
disable framerelay li debug
enable framerelay
enable framerelay debug
enable framerelay dlc debug
enable framerelay li debug
show framerelay

disable framerelay dlc

Syntax `DISable FRamerelay=`*`fr-interface`* `DLC=`*`dlci`*

where:

- *fr-interface* is the number of the Frame Relay interface.
- *dlci* is the identification number of a Frame Relay Data Link Connection (DLC) from 0 to 1023.

Description This command disables a DLC; an enabled one must already exist. A disabled DLC does not receive or transmit data. Routing modules are not informed that a disabled DLC is active. Once a DLC has been disabled, it can be enabled again.

It is not possible to disable the LMI DLC because it must remain active for the LMI dialogue to take place. The LMI DLC is different from other DLCs in that it does not necessarily appear in the full status messages that the network sends to the router.

Examples To disable the DLC with a DLCI of 8 on Frame Relay interface 1, use the command:

```
dis fr=1 dlc=8
```

Related Commands

disable framerelay
disable framerelay dlc debug
enable framerelay
enable framerelay dlc
enable framerelay dlc debug
show framerelay dlc

disable framerelay dlc debug

Syntax

```
DISable FRamerelay=fr-interface DLC={dlci|ALL} DEBug={ALL|
   ATKPkt|BRGPkt|CCPPkt|CCPState|CNGSTstate|COMPPkt|
   DLCState|DNTPkt|ECPPkt|ECPState|ENCO|ENCPkt|IPPkt|
   IPXPkt|PKT|PLLCstate}
```

where:

- *fr-interface* is the number of the Frame Relay interface.
- *dlci* is the identification number of a Frame Relay Data Link Connection (DLC) from 0 to 1023.

Description

This command disables all debug modes or a specific one for all DLCs or a specific one.

The **debug** parameter specifies the debug mode to disable. If **all** is specified, all currently enabled debug modes are disabled on the DLC.

atkpkt debug mode displays AppleTalk packets that are received and transmitted by the router.

brgpkt debug mode displays bridged packets that are received and transmitted by the router.

ccppkt debug mode displays CCP (Compression Control Protocol) packets that are received and transmitted by the router.

ccpstate debug mode displays the state changes in the CCP (Compression Control Protocol) state machine that occur when CCP packets are received.

cngststate debug mode displays the state changes in the congestion state machine that occur when the network becomes congested.

comppkt debug mode displays packets containing compressed data that are received and transmitted by the router.

dlcstate debug mode displays the state changes in the DLC state machine that occur when LMI packets are received.

dntpkt debug mode displays DECnet packets that are received and transmitted by the router.

ecppkt debug mode displays ECP (Encryption Control Protocol) packets that are received and transmitted by the router.

ecpstate debug mode displays the state changes in the ECP (Encryption Control Protocol) state machine that occur when ECP packets are received.

enco debug mode displays the state changes in the ENCO interface state machine when ENCO interface events occur.

encpkt debug mode displays packets containing encrypted data that are received and transmitted by the router.

ippkt debug mode displays IP packets that are received and transmitted by the router.

ipxpkt debug mode displays Novell IPX packets that are received and transmitted by the router.

pkt debug mode displays all network protocol packets that are received and transmitted by the router.

pllcstate debug mode displays the state changes in the PLLC state machine that occur when LMI STATUS packets are received.

Examples To disable the **ippkt** debug mode on DLC 23 on Frame Relay interface 0, use the command:

```
dis fr=0 dlc=23 deb=ipp
```

Related Commands
disable framerelay
disable framerelay dlc
enable framerelay
enable framerelay dlc
enable framerelay dlc debug
show framerelay dlc

disable framerelay li debug

Syntax

```
DISable FRamerelay=fr-interface LI=logical-interface
   DEBug=USER
```

where:

- *fr-interface* is the number of the Frame Relay interface.
- *logical-interface* is the number of the logical interface from 1 to 1023.

Description This command disables debug modes for a specific logical interface.

The **debug** parameter specifies the type of debug mode to disable. USER debug mode displays debugging information about the network protocols using the logical interface.

Examples To disable **user** debug mode on LI 0 of Frame Relay interface 0, use the command:

```
dis fr=0 li=0 deb=user
```

Related Commands
disable framerelay debug
disable framerelay dlc debug
enable framerelay debug
enable framerelay dlc debug
enable framerelay li debug
show framerelay li

disable framerelay slowstart

Syntax `DISable FRamerelay=`*`fr-interface`* `SLOWstart`

where *fr-interface* is the number of the Frame Relay interface

Description This command disables the slow-start mechanism for a Frame Relay interface.

Examples To disable the slow-start mechanism on Frame Relay interface 2, use the command:

```
dis fr=2 slow
```

Related Commands
disable framerelay congestioncontrol
enable framerelay congestioncontrol
enable framerelay slowstart
show framerelay

enable framerelay

Syntax `ENAble FRamerelay=`*`fr-interface`*

where *fr-interface* is the number of the Frame Relay interface

Description This command enables a previously disabled Frame Relay interface. If the interface is already enabled, this command has no effect. Data can be transmitted or received via the interface.

Examples To enable Frame Relay interface 2, use the command:

```
ena fr=2
```

Related Commands
create framerelay
destroy framerelay
disable framerelay
reset framerelay
set framerelay
show framerelay

enable framerelay congestioncontrol

Syntax `ENAble FRamerelay=`*`fr-interface`* `CONgestioncontrol`

where *fr-interface* is the number of the Frame Relay interface

Description This command enables congestion control on an interface to detect congestion and limit data transmission as necessary.

Examples To enable congestion control on Frame Relay interface 3, use the command:

```
ena fr=3 con
```

Related Commands

disable framerelay congestioncontrol
disable framerelay debug
show framerelay

enable framerelay debug

Syntax

```
ENAble FRamerelay=fr-interface DEBug={ALL|CLLMPkt|
   LMIState|LMIPkt|UTILisation} [ASYn=port-number]
```

where:

- *fr-interface* is the number of the Frame Relay interface.
- *port-number* is the number of an asynchronous port on the router.

Description

This command enables Frame Relay interface debug modes.

The **debug** parameter specifies the type of debug mode to enable. If **all** is specified, all debug modes are enabled on the interface. **cllmpkt** debug mode decodes and displays CLLM packets that are received by the router. **lmistate** debug mode displays the state changes in the LMI state machine that occur when LMI packets are received. **lmipkt** debug mode decodes and displays LMI packets that are received and transmitted by the router. **utilisation** debug mode displays the utilisation of each DLC and the total utilisation of the interface once per second.

The **asyn** parameter specifies the asynchronous port where the debug output is to be sent. This permits debugging to be enabled in a script. The default is to send output to the terminal or Telnet session where the command was executed.

Examples

To enable **lmistate** debug mode on Frame Relay interface 1, use the command:

```
ena fr=1 deb=lims
```

Related Commands

disable framerelay
disable framerelay debug
disable framerelay dlc debug
disable framerelay li debug
enable framerelay
enable framerelay dlc debug
enable framerelay li debug
show framerelay

enable framerelay dlc

Syntax `ENAble FRamerelay=`*`fr-interface`* `DLC=`*`dlci`*

where:

- *fr-interface* is the number of the Frame Relay interface.
- *dlci* is the identification number of a Frame Relay Data Link Connection (DLC) from 0 to 1023.

Description This command enables a DLC for use by the Frame Relay interface; a disabled DLC must already exist. A DLC that has been enabled can receive and transmit data, and routing modules are informed that a DLC is active.

It is not possible to enable an LMI DLC because it must remain active for the LMI dialogue to take place. The LMI DLC is different from other DLCs in that it does not necessarily appear in the full status messages that the network sends to the router.

Examples To enable the DLC with a DLCI of 8 on Frame Relay interface 1, use the command:

```
ena fr=1 dlc=8
```

Related Commands

add framerelay dlc
delete framerelay dlc
disable framerelay
disable framerelay dlc
enable framerelay
reset framerelay
reset framerelay dlc
show framerelay

enable framerelay dlc debug

Syntax

```
ENAble FRamerelay=fr-interface DLC=dlci DEBug={ALL|ATKPkt|
   BRGPkt|CCPPkt|CCPState|CNGSTstate|COMPPkt|DLCState|
   DNTPkt|ECPPkt|ECPState|ENCO|ENCPkt|IPPkt|IPXPkt|PKT|
   PLLCstate}} [ASYn=port-number]
```

where:

- *fr-interface* is the number of the Frame Relay interface.
- *dlci* is the identification number of a Frame Relay Data Link Connection (DLC) from 0 to 1023.
- *port-number* is the number of an asynchronous port on the router.

Description

This command enables debug modes for a specific DLC.

The **debug** parameter specifies the type of debug mode to disable. If **all** is specified, all enabled debug modes are disabled on the DLC.

atkpkt debug mode displays AppleTalk packets that are received and transmitted by the router.

brgpkt debug mode displays bridged packets that are received and transmitted by the router.

ccppkt debug mode displays CCP (Compression Control Protocol) packets that are received and transmitted by the router.

ccpstate debug mode displays the state changes in the CCP (Compression Control Protocol) state machine that occur when CCP packets are received.

cngststate debug mode displays the state changes in the congestion state machine that occur when the network becomes congested.

comppkt debug mode displays packets containing compressed data that are received and transmitted by the router.

dlcstate debug mode displays the state changes in the DLC state machine that occur when LMI packets are received.

dntpkt debug mode displays DECnet packets that are received and transmitted by the router.

ecppkt debug mode displays ECP (Encryption Control Protocol) packets that are received and transmitted by the router.

ecpstate debug mode displays the state changes in the ECP (Encryption Control Protocol) state machine that occur when ECP packets are received.

enco debug mode displays the state changes in the ENCO interface state machine when ENCO interface events occur.

encpkt debug mode displays packets containing encrypted data that are received and transmitted by the router.

ippkt debug mode displays IP packets that are received and transmitted by the router.

ipxpkt debug mode displays Novell IPX packets that are received and transmitted by the router.

pkt debug mode displays all network protocol packets that are received and transmitted by the router.

pllcstate debug mode displays the state changes in the PLLC state machine that occur when LMI STATUS packets are received.

The **asyn** parameter specifies the asynchronous port to which to send the debug output. This permits debugging to be enabled in a script. The default is to send the output to the terminal or Telnet session where the command was executed.

Examples

To enable the **ippkt** debug mode on DLC 23 on Frame Relay interface 0 and send the output to a terminal attached to asynchronous port 3, use the command:

```
ena fr=0 dlc=23 deb=ipp asy=3
```

Related Commands

disable framerelay
disable framerelay dlc
disable framerelay dlc debug
enable framerelay
enable framerelay dlc
show framerelay dlc

enable framerelay li debug

Syntax

```
ENAble FRamerelay=fr-interface LI=logical-interface
   DEBug=USER [ASYn=port-number]
```

where:

- *fr-interface* is the number of the Frame Relay interface.
- *logical-interface* is the number of the logical interface from 1 to 1023.
- *port-number* is the number of an asynchronous port on the router.

Description

This command enables debug modes for a specific logical interface.

The **debug** parameter specifies the type of debug mode to enable. **user** debug mode displays debugging information about network protocols that use the logical interface.

The **asyn** parameter specifies the asynchronous port where the debug output is to be sent. This permits debugging to be enabled in a script. The default is to send output to the terminal or Telnet session where the command was executed.

Examples

To enable **user** debug mode on LI 0 of Frame Relay interface 0, use the command:

```
ena fr=0 LI=0 deb=user
```

Related Commands

disable framerelay debug
disable framerelay dlc debug
disable framerelay li debug
enable framerelay debug
enable framerelay dlc debug
show framerelay li

enable framerelay slowstart

Syntax

```
ENAble FRamerelay=fr-interface SLOWstart
```

where *fr-interface* is the number of the Frame Relay interface

Description

This command enables the slow-start mechanism for a Frame Relay interface.

Examples

To enable the slow-start mechanism on Frame Relay interface 1, use the command:

```
ena fr=1 slow
```

Related Commands

disable framerelay congestioncontrol
disable framerelay slowstart
enable framerelay congestioncontrol
show framerelay

reset framerelay

Syntax `RESET FRamerelay=`*`fr-interface`*

where *fr-interface* is the number of the Frame Relay interface

Description This command resets a Frame Relay interface. All frames currently being transmitted or received are discarded. All active DLCs are removed and all counters are reset to zero. The LMI dialogue is restarted and the DLCs are re-established over time.

Examples To reset Frame Relay interface 1, use the command:

```
reset fr=1
```

Related Commands

create framerelay
destroy framerelay
disable framerelay
enable framerelay
set framerelay
show framerelay

reset framerelay dlc

Syntax `RESET FRamerelay=`*`fr-interface`* `DLC=`*`dlci`*

where:

- *fr-interface* is the number of the Frame Relay interface.
- *dlci* is the identification number of a Frame Relay Data Link Connection (DLC) from 0 to 1023.

Description This command resets an existing DLC being used by the Frame Relay interface. The DLC is removed from the list of active DLCs until the LMI dialogue brings the DLC up again. All status counters for the DLC are reset to zero.

It is not possible to reset the LMI DLC because this DLC must remain active for the LMI dialogue to take place. The LMI DLC is different from other DLCs in that it does not necessarily appear in the full status messages that the network sends to the router.

Examples To reset the DLC with a DLCI of 8 on Frame Relay interface 1, use the command:

```
reset fr=1 dlc=8
```

Related Commands

add framerelay dlc
delete framerelay dlc
disable framerelay
disable framerelay dlc
enable framerelay
enable framerelay dlc
set framerelay
show framerelay

set framerelay

Syntax

```
SET FRamerelay=fr-interface [DEFATencap={Appleoui|
   Ietfoui}] [BECNLimit=2..4294967295]
   [BECNTimeout=1..4294967295] [CDETection={BECN|CLLM|
   BOTH}] [CTICK=1..60] [DECreasefactor={0.025|0.05|0.1|
   0.2}] [DEFCcompression={ON|OFF}]
   [DEFENCApsulation={IETF|CISCO}] [DEFEncryption={ON|
   OFF}] [DROPOFFthreshold=1..100] [INCreasefactor={0.05|
   0.1|0.2}] [LMIscheme={LMIREV1|Q933A|ANNEXA|T1617B|
   ANNEXB|T1617D|ANNEXD|NONE}]
   [MAXTXQlength=1..4294967295] [NN1=nn1] [NN2=nn2]
   [NN3=nn3] [NT1=nt1] [RECoverytime=1..4294967295]
```

where:

- *fr-interface* is the number of the Frame Relay interface.
- *nn1* is the value of the NN1 parameter.
- *nn2* is the value of the NN2 parameter.
- *nn3* is the value of the NN3 parameter.
- *nt1* is the value of the NT1 parameter.

Description

This command changes the operational parameters of a Frame Relay interface. The interface must have been created with the **create framerelay** command on page 14-29. All of these parameters may also be set with the **create framerelay** command when the interface is created.

Any or all of the parameters may be specified. These parameters and their values are defined in the Frame Relay standards. Defaults for LMI parameters are defined in the standards but can be altered to reflect the requirements of the Frame Relay network.

Note that changes take effect immediately for the **defencapsulation**, **nt1**, **nn1**, **nn2**, and **nn3** parameters. However, when you change other parameters, they automatically reset the FR interface before taking effect.

The **defencapsulation** parameter specifies the default AppleTalk packet encapsulation (OUI SNAP encapsulation) that is applied to the packet before it is sent over the Frame Relay interface. The **appleoui** option specifies SNAP encapsulation with Apple OUI and **ietfoui** specifies SNAP encapsulation with IETF OUI. The default is **appleoui**.

The **becnlimit** parameter specifies the number of continuous frames that can be received with the BECN bit set before the DLC is deemed to be severely congested. The default is **100**.

The **becntimeout** parameter specifies the maximum number of seconds that a DLC remains congested after a packet with the BECN bit cleared is received, when congestion is being detected using BECN bits. The default is **10**.

The **cdetection** parameter specifies the method used to detect congestion on the Frame Relay interface. If b**ecn** is specified, congestion is detected using only the BECN bits in received packets. If **cllm** is specified, congestion is detected using only CLLM messages. If **both** is specified, both CLLM messages and BECN bits are used. The default is **both**.

The **ctick** parameter specifies the period of the congestion timer. The congestion timer is started when a DLC becomes severely congested. When it expires the first time the allowable transmission rate of the DLC is reduced to 0.625*CIR. When it expires the second time the allowable throughput is reduced to 0.5*CIR. After the third expiry the allowable throughput is set to 0.25*CIR and the timer is not started again. The default is 10 seconds.

The **decreasefactor** parameter specifies the fractional amount by which to decrease the allowable transmission rate each second if the actual transmission rate in the last second was below the **dropoffthreshold** percentage of the allowable transmission rate. The default is 0.025.

The **defencryption** and **defcompression** parameters set the default encryption and compression values respectively for all DLCs in the interface. The default is **off** for both. The **defencryption** parameter requires a user with security officer privilege when the router is in security mode.

The **defencapsulation** parameter specifies the default protocol encapsulation used for all DLCs in the Frame Relay interface. If **ietf** is specified, the IETF encapsulation defined in RFC 1490 is used. If **cisco** is specified, the Cisco Systems proprietary encapsulation is used. The default is **ietf**.

The **dropoffthreshold** parameter specifies the percentage of the allowable transmission rate that the actual transmission rate must stay above for the allowable transmission rate to not be reduced. If the actual transmission rate drops below the **dropoffthreshold** percentage of the allowable transmission rate then the allowable transmission rate is reduced. The default is 70%.

The **increasefactor** parameter specifies the fractional amount by which to increase the allowable transmission rate each second if there is data to send. The default is **0.05**.

The **lmischeme** parameter specifies the scheme used for the LMI, the dialogue between the router and the Frame Relay network. ANNEXA, ANNEXB and ANNEXD are synonyms for Q933A, T1617B and T1617D respectively. LMIREV1 is the initial standard created by the vendors' consortium that first produced Frame Relay products. ANNEXA refers to Annex A of the Q.933 ITU Recommendation. ANNEXB and ANNEXD refer to the Annexes of the ANSI standard for Frame Relay. To turn off LMI, specify **none**. The default is **lmirev1**.

The **maxtxqlength** parameter specifies the maximum length for the transmission queue of the Frame Relay interface. The default is **100**.

The **nn1** parameter specifies the ratio of normal status enquiry messages to full status enquiry messages. The full status enquiry message is used by the router to request a full status message from the network. The full status message is the message that contains the status of all DLCs known to the network. Since the status of DLCs is not expected to change frequently and the full status message is longer than the normal status message, it is usual to set NN1 to a number greater than 1 to ensure that excess bandwidth is not devoted to LMI messages. The value *nn1* must be a number between 1 and 255 inclusive and the default is **6**.

The **nn2** parameter specifies the number of error events in the last **nn3** events that flag an alarm condition. An event in this context is the reception of a reply to the status enquiry message sent by the router. An error event is a timeout before this message is received, or a reply being received with an incorrect format. The value *nn2* must be a number between 1 and 10 inclusive, and must be less than or equal to the value of **nn3**. The default is **2**.

The **nn3** parameter specifies the number of events used to determine alarm conditions and the negation of the alarm condition. If **nn2** out of the last **nn3** events are in error, an alarm is set. The alarm is cleared again when **nn3** events are received without error. The value *nn3* must be a number between 1 and 10 inclusive, and must be greater than or equal to the value of **nn2**. The default for **nn3** is **4**.

Note the dependence of **nn2** and **nn3** on each other. Any value may be specified for **nn2** and **nn3**, as long as at the end of processing the command line, the value of **nn2** is less than or equal to the value of **nn3**.

The **nt1** parameter specifies the interval between status enquiry messages transmitted by the router to the Frame Relay network. The status enquiry message is the basic poll message of the LMI dialogue. The value *nt1* must be a multiple of 5 and must fall between 5 and 30 inclusive. The default is 10 seconds. The change to **nt1** takes place immediately, although the effect takes place after the next status enquiry message is sent to the network.

The values of **nt1** and **nn1** combine to determine how often full status enquiry messages are sent to the network. The minimum value for the period between full status enquiry messages is 5 seconds and the maximum value is 7650 seconds (2 hours, 7 minutes, 30 seconds).

The **recoverytime** parameter specifies the time over which the DLC is deemed to be recovering from severe congestion. During this time the allowable transmission rate is increased by the **increasefactor** value * the lowest allowable transmission rate reached during severe congestion rather than by the **increasefactor** value * bandwidth. Once the recovery time has passed the rate is increased by **increasefactor** value * bandwidth again. The default is 20 seconds.

Examples

To disable the LMI on Frame Relay interface 0, use the command:

```
set fr=0 lmi=none
```

Related Commands

create framerelay
destroy framerelay
disable framerelay
enable framerelay
reset framerelay
show framerelay

set framerelay dlc

Syntax

```
SET FRamerelay=fr-interface DLC=dlci [CIR={cir|NONE}]
   [CIRLIMited=ON|OFF|YES|NO]
   [COMPression={DEFault|ON|OFF}] [EIR={eir|NONE}]
   [ENCAPsulation={DEFault|IETF|CISCO}]
   [ENCryption={DEFault|ON|OFF}] [LI=logical-interface]
```

where:

- *fr-interface* is the number of the Frame Relay interface.
- *dlci* is the identification number of a Frame Relay Data Link Connection (DLC) from 0 to 1023.
- *cir* is the Committed Information Rate in bits per second.
- *eir* is the Excess Information Rate in bits per second.
- *logical-interface* is the number of the logical interface from 1 to 1023.

Description

This command sets the parameters of a DLC. At least one of **cir**, **eir**, **encryption**, **compression**, or **li** must be specified.

The **cir** parameter specifies the Committed Information Rate that the DLC provides. This value sets the rate at which data is transmitted when the network is congested, and when first sending data on the DLC after it has been idle. The **cir** must be less than the bandwidth of the link. If **none** is specified and **none** is also set for the **eir** parameter, the transmission rate is unrestricted. Otherwise, the transmission rate is always set to the EIR value. The default is half the bandwidth of the link.

The **cirlimited** parameter is used to permanently throttle the transmission rate back to the CIR. Doing this avoids both the overhead of detecting congestion and the delay while routers back-off their transmission. If **on** or **yes** is specified, the maximum transmission rate of the DLC to the CIR of that DLC is set, regardless of congestion state of the network. If **off** or **no** is specified, the transmission rate is unlimited unless there is congestion on the network and congestion control is on. The default is **off/no**.

The **eir** parameter specifies the Excess Information Rate that the DLC provides. The transmission rate is the sum of **eir** and **cir** when the network is not congested. The **eir** must be less than the bandwidth of the link, and the sum of the **eir** and **cir** must be less than the bandwidth of the link. If **none** is specified and **none** is also set for the **cir**, the transmission rate is unrestricted. If **eir** is set to **none** and **cir** is **not** set to **none**, then the transmission rate is unrestricted while the network is not congested, and set by the CIR when the network is congested. These options are shown in Table 14-4 on page 14-13. The default is **none**.

The **encapsulation** parameter specifies the type of encapsulation to use for transmission of packets over the DLC. If **ietf** is specified, the **ietf** encapsulation defined in RFC 1490 is used. If **cisco** is specified, the Cisco Systems proprietary encapsulation is used. If **default** is specified, the interface encapsulation default is used for the DLC.

The **encryption** parameter enables or disables encryption on the DLC, and requires a user with security officer privilege when the router is in security mode.

The **compression** parameter enables or disables compression on the DLC. If **default** is specified for either of the parameters the interface default is used.

If a parameter is set to a non-default for a DLC that the router has not been informed about by the LMI, then a DLC is created to record the information. The DLC goes into a AWAIT_LMI state until the network informs the router via the LMI that the DLC is active.

The **li** parameter specifies the logical interface to associate with the DLC.

Examples

To disable compression and encryption and set a CIR of 64Kbits/sec on the DLC with a DLCI of 8 on Frame Relay interface 1, use the command:

```
set fr=1 dlc=8 enc=off comp=off cir=64000
```

Related Commands

add framerelay dlc
create framerelay
delete framerelay dlc
disable framerelay dlc
disable framerelay dlc debug
enable framerelay dlc
enable framerelay dlc debug
set framerelay
show framerelay
show framerelay dlc

show framerelay

Syntax

```
SHow FRamerelay[=fr-interface] [{CIRUTILisation|CONFig|
   COUnter|DEBug|LMI|UTILisation}]
```

where *fr-interface* is the number of the Frame Relay interface

Description

This command displays information about the Frame Relay interfaces on the router. If an interface is specified, information is displayed for the specified Frame Relay interface. If an interface is not specified, information is displayed for all Frame Relay interfaces. If a parameter is not specified, a summary of the Frame Relay interfaces is displayed (Figure 14-16 on page 14-52, Table 14-5 on page 14-52).

The **cirutilisation** parameter displays the percentage of CIR being used on each DLC in the Frame Relay interface (Figure 14-17 on page 14-53, Table 14-6 on page 14-53).

The **config** parameter displays configuration parameters for the Frame Relay interface (Figure 14-18 on page 14-54, Table 14-7 on page 14-54).

The **counter** parameter displays general configuration counters and counters from the interface MIB and the Frame Relay MIB for the Frame Relay interface and each DLC (Figure 14-19 on page 14-56, Table 14-8 on page 14-57).

The **debug** parameter displays details of the debug modes that have been enabled for the Frame Relay interface, all the DLCs in the Frame Relay interface and all the logical interfaces of the Frame Relay interface (Figure 14-20 on page 14-60, Table 14-9 on page 14-60).

The **lmi** parameter displays information about the management dialogue between the router and the Frame Relay network (Figure 14-21 on page 14-60, Table 14-10 on page 14-60).

The **utilisation** parameter displays information about the percentage of the total link bandwidth used by each DLC in the Frame Relay interface (Figure 14-22 on page 14-61, Table 14-11 on page 14-61).

Figure 14-16: Example output from the **show framerelay** command

```
Interface Enabled IfIndex Over    Active Logical Interfaces
-----------------------------------------------------------
fr0         YES     05    syn0    YES    0,1,2,3,7
fr1         YES     06    syn1    YES    0,2,3
fr2         YES     07    syn2    YES    0
-----------------------------------------------------------
```

Table 14-5: Parameters in the output of the **show framerelay** command

Parameter	Meaning
Interface	Interface name.
Enabled	Whether the interface is enabled.
IfIndex	Value of IfIndex for the Frame Relay interface.

Table 14-5: Parameters in the output of the **show framerelay** command (Continued)

Parameter	Meaning
Over	Physical interface used by the Frame Relay interface; either syn*n*, TDM-*groupname,* or ISDN-*callnam*e.
Logical Interfaces	Logical interfaces defined for the Frame Relay interface.

Figure 14-17: Example output from the **show framerelay cirutilisation** command

```
Interface
 Dlc     CIR (bps)  CIR Utilisation (%)  Bandwidth: 48000
-------------------------------------------------------------
fr0
 0020     36400       89
 0021      9600      120
-------------------------------------------------------------
```

Table 14-6: Parameters in the output of the **show framerelay cirutilisation** command

Parameter	Meaning
Interface	Interface name.
Dlc	DLC identifier.
CIR	Committed Information Rate (CIR) defined for the DLC.
CIR Utilisation	Percentage of the CIR that is being used. This value is determined from outgoing traffic only.
Bandwidth	Bandwidth of the Frame Relay interface.

Figure 14-18: Example output from the **show framerelay config** command

```
Interface
  Parameter                                      Value
-------------------------------------------------------------
fr0
  Over ................................... syn0
  Enabled ................................ Yes
  ifIndex ................................ 4
  Logical Interfaces ..................... 0
  LMI Scheme ............................. None
  Polling Interval (NT1) (secs) .......... 10
  Full Enquiry Interval (NT1*NN1) (secs) . 60
  Error Threshold (NN2) .................. 2
  Monitored Events (NN3) ................. 4
  Default encapsulation .................. IETF
  Default encryption ..................... Off
  Default compression .................... Off
  Slow Start ............................. Disabled
  Increase Factor ........................ 0.05
  Decrease Factor ........................ 0.025
  Dropoff Threshold (%) .................. 70
  Maximum Transmission Queue Length ...... 100
  Congestion Control ..................... Disabled
  Congestion Detection Method ............ BECN and CLLM
  Continuous BECN Limit .................. 100
  BECN Timeout (secs) .................... 10
  Congestion Tick (secs) ................. 10
  Congestion Recovery Time (secs) ........ 60
-------------------------------------------------------------
```

Table 14-7: Parameters in the output of the **show framerelay config** command

Parameter	Meaning
Interface	Interface name.
Over	Physical interface used by the Frame Relay interface; either syn*n*, ISDN-*callname,* or TDM-*groupname*.
Enabled	Whether the interface is enabled.
IfIndex	Value of IfIndex for the Frame Relay interface.
Logical Interfaces	Logical interfaces defined for the Frame Relay interface.
LMI Scheme	LMI scheme used for the interface: LMI revision 1
	Annex A
	Annex B
	Annex D
	None
	Undefined
Polling Interval (NT1) (secs)	Interval in seconds at which the router transmits status enquiry messages to the Frame Relay network.
Full Enquiry Interval (NT1*NN1) (secs)	Interval in seconds at which the router transmits full status enquiry messages.
Error Threshold (NN2)	Number of error events in the last NN3 events that causes an alarm condition to be flagged.

Table 14-7: Parameters in the output of the **show framerelay config** command (Continued)

Parameter	Meaning
Monitored Events (NN3)	Number of events that determine alarm conditions.
Default encapsulation	Whether the default encapsulation on the interface is IETF or Cisco.
Default encryption	Whether the default encryption is used by DLCs in the interface.
Default compression	Whether the default compression is used by DLCs in the interface.
Slow Start	Whether the slow-start mechanism is enabled on the interface.
Increase Factor	Fraction of CIR to increase the allowable transmission rate by for the slow-start mechanism: 0.05
	0.1
	0.2
Decrease Factor	Fraction of CIR to decrease the allowable transmission rate by when data is being transmitted at a rate below the dropoff threshold percentage of the allowable transmission rate: 0.025
	0.05
	0.1
	0.2
Dropoff Threshold (%)	Percentage of allowable transmission rate that the actual transmission rate must remain above to prevent reducing the allowable transmission rate.
Maximum Transmission Queue Length	Maximum length of the transmission queue.
Congestion Control	Whether congestion control is enabled on the interface.
Congestion Detection Method	Whether the method that detects congestion on the interface is BECN, CLLM, or BECN and CLLM. This field is valid when the Congestion Control field is enabled.
Continuous BECN Limit	Number of continuous frames that can be received with the BECN bit set before the DLC is deemed to be severely congested.
BECN Timeout (secs)	Maximum number of seconds that a DLC remains in a congested state after a packet with the BECN bit cleared has been received.
Congestion Tick (secs)	Congestion timer interval in seconds.
Congestion Recovery Time (secs)	Interval in seconds during which the DLC recovers from severe congestion.

Figure 14-19: Example output from the **show framerelay counter** command

```
Interface Counter:

fr0            401  seconds     Last change at:  10 seconds
Interface Counter
  ifInOctets             935     ifOutOctets              550
  ifInUcastPkts            0     ifOutUcastPkts            41
  ifInNUcastPkts           0     ifOutNUcastPkts            0
  ifInDiscards             0     ifOutDiscards              0
  ifInErrors               0     ifOutErrors                0
  ifInUnknownProtos        0     ifOutQLen                  0

  ifInNullCircuits         0     ifOutNullCircuits          0
  ifNoHLReceiveFunction    0     ifBadPackets               0
  ifGoodConfiguration      2     cllmMessagesReceived      19
  shortFrames              0     lostTxReadyInds            0
  inLMIStatus             33     outLMIStatusEnq           33
  inLMIFullStatus          7     outLMIFullStatusEnq        7

DLCI: 0
  State                Active
  Created                  6     Last Changed               6
  statusActive             0     statusInactive             0
  inOctets               555     outOctets                520
  inFrames                40     outFrames                 40
  BECNs Received           0     FECNs Received             0
  notEnabled               0     ECPNotOpen                 0
  congestionDiscards       0
  cllmCongestionMild       0     cllmCongestionSevere       0
  cllmDiscarding           0     cllmNetworkTrouble         0
```

```
DLCI: 21
  State              Inactive
  Created                1000     Last Changed           37000
  statusActive              6     statusInactive             1
  inOctets                  0     outOctets                 30
  inFrames                  0     outFrames                  1
  BECNs Received            0     FECNs Received             0
  notEnabled                0     ECPNotOpen                 0
  congestionDiscards        0
  cllmCongestionMild        4     cllmCongestionSevere       4
  cllmDiscarding           11     cllmNetworkTrouble         0

TOTAL
  inOctets                555     outOctets                550
  inFrames                 40     outFrames                 41
  BECNs Received            0     FECNs Received

General Counter:
  configBadInstance         0     configBadProtocol          0
  configUnusedInstance      0     dataReqNullFrame           0
  dataReqBadInstance        0     compEventBadUserId         0
  encryptEventBadUserId     0     encrStarEventBadUserId     0
  idleBadInstance           0     idleUnusedInterface        0
  interRxCompBadInst        0
```

Table 14-8: Parameters in the output of the **show framerelay counter** command

Parameter	Meaning
fr0	Interface name.
seconds	Seconds since the interface was last re-initialised.
Last change at	Seconds since the interface entered its current operational state.
ifInOctets	Number of octets received on the interface, including framing characters.
ifOutOctets	Number of octets transmitted out of the interface, including framing characters.
ifInUcastPkts	Number of subnetwork-unicast frames delivered to a higher-layer protocol.
ifOutUcastPkts	Number of frames that higher-layer protocols requested be transmitted to a subnetwork-unicast address, including those that were discarded or not sent.
ifInNUcastPkts	Number of non-unicast frames delivered to a higher-layer protocol.
ifOutNUcastPkts	Number of frames that higher-layer protocols requested be transmitted to a non-unicast address, including those that were discarded or not sent.
ifInDiscards	Number of inbound frames to be discarded even though no errors were detected to prevent their being deliverable to a higher-layer protocol. One possible reason for discarding such a frame could be to free up buffer space.
ifOutDiscards	Number of outbound frames to be discarded even though no errors were detected to prevent their being transmitted. One possible reason for discarding such a frame could be to free up buffer space.
ifInErrors	Number of inbound frames that contained errors preventing them from being deliverable to a higher-layer protocol.
ifOutErrors	Number of outbound frames that contained errors preventing them from being transmitted.
ifInUnknownProtos	Number of frames received via the interface were discarded because of an unknown or unsupported protocol.
ifOutQLen	Length of the output frame queue.
ifInNullCircuits	Number of inbound frames addressed to a DLC that did not exist.
ifOutNullCircuits	Number of outbound frames addressed to a DLC that did not exist.
ifNoHLReceiveFunction	Number of inbound frames addressed to higher layers with no receive function specified.
ifBadPackets	Number of frames received with a badly formed Frame Relay header.
ifGoodConfiguration	Number of good configure requests received from user modules.
cllmMessagesReceived	Number of CLLM messages received by a Frame Relay interface.

Table 14-8: Parameters in the output of the **show framerelay counter** command (Continued)

Parameter	Meaning
shortFrames	Number of frames received that do not contain enough data to be processed.
lostTxReadyInds	Number of Transmit Ready indications from the Layer 1 driver that were expected but not received.
inLMIStatus	Number of LMI keep alive status messages received by a Frame Relay interface.
outLMIStatusEnq	Number of LMI keepalive status enquiries sent by a Frame Relay interface.
TinLMIFullStatus	Number of LMI full status messages received by a Frame Relay interface.
outLMIFullStatusEnq	Number of LMI full status enquiries sent by a Frame Relay interface.
DLCI	**DLCI number for a DLC**
State	Whether the state of the DLC is active, inactive, invalid, or undefined.
Created	Time the DLC was created.
Last change	Time of the last change to the status of the DLC.
statusActive	Number of messages received indicating that a circuit is currently active.
statusInactive	Number of messages received indicating that a circuit is currently inactive.
In Octets	Number of octets received from the Frame Relay network over the DLC.
OutOctets	Number of octets transmitted to the Frame Relay network over the DLC.
In Frames	Number of frames received from the Frame Relay network over the DLC.
OutFrames	Number of frames transmitted to the Frame Relay network over the DLC.
FECNs Received	Number of frames received from the Frame Relay network over the DLC with the FECN bit set.
BECNs Received	Number of frames received from the Frame Relay network over the DLC with the BECN bit set.
notEnabled	Number of frames received from the Frame Relay network over the DLC while the DLC was locally disabled.
ECPNotOpen	Number of frames received from the Frame Relay network over the DLC while the DLC's ECP was not in the opened state.
congestionDiscards	Number of frames passed to Frame Relay by a higher layer that were discarded because the Frame Relay network was congested.
cllmCongestionMild	Number of CLLMs received by a circuit indicating mild congestion.
cllmCongestionSevere	Number of CLLMs received by a circuit indicating severe congestion.

Table 14-8: Parameters in the output of the **show framerelay counter** command (Continued)

Parameter	Meaning
cllmDiscarding	Number of CLLMs received by a circuit indicating discarding.
cllmNetworkTrouble	Number of CLLMs received by a circuit indicating network trouble.
TOTAL	**Totals for the Frame Relay interface**
In Octets	Total octets received from the Frame Relay network over all DLCs.
OutOctets	Total octets transmitted to the Frame Relay network over all DLCs.
In Frames	Total frames received from the Frame Relay network over all DLCs.
OutFrames	Total frames transmitted to the Frame Relay network over all DLCs.
configBadInstance	Number of configure requests received from user modules for an invalid interface.
configBadProtocol	Number of configure requests received from user modules with a bad protocol.
configUnusedInstance	Number of configure requests received from user modules for an unused interface.
dataReqNullFrame	Number of data requests received from user modules with no data frame.
dataReqBadInstance	Number of data requests received from user modules for an unused or invalid interface.
compEventBadUserId	Number of events received from the compression module with a bad user id.
encryptEventBadUserId	Number of events received from the encryption module with a bad user id.
encrStarEventBadUserId	Number of star events received from the encryption module with a bad user id.
idleBadInstance	Number of input packets received by the idle loop for an invalid interface.
idleUnusedInterface	Number of input packets received by the idle loop for an unused interface.
interRxCompBadInst	Number of input packets received for compression for an invalid interface.

Figure 14-20: Example output from the **show framerelay debug** command

```
Interface
  Debug       Device  Enabled Modes
------------------------------------------
fr0
  Interface   16      UTILISATION

  Dlc 20      16      IPXPKT, DLCSTATE

  Li 0        16      USER
------------------------------------------
```

Table 14-9: Parameters in the output of the **show framerelay debug** command

Error	Meaning
Interface	Interface name.
Debug	Whether the type of object for which debugging is enabled is Interface, DLC, or LI. When DLC or LI, the DLC or LI identifier is also specified.
Device	Device on which the debug output is displayed.
Enabled Modes	Debug modes enabled for each debug type and instance.

Figure 14-21: Example output from the **show framerelay lmi** command

```
Interface
  Parameter                     Value
-----------------------------------------------------
fr0
  LMI State ................... Idle
  Polling Interval (secs) ..... 10
  Alarm Condition ............. False
  Alarm Condition count ....... 0
  Good Events ................. 4
  Error Events ................ 0
  Current Sequence ............ 12
  Receive Sequence ............ 11
-----------------------------------------------------
```

Table 14-10: Parameters in the output of the **show framerelay lmi** command

Error	Meaning
Interface	Interface name.
LMI State	Whether the state of the LMI is down, idle, or status_wait.
Polling Interval (secs)	The time interval between transmissions of status request messages to the network.
Alarm Condition	Whether the network is responding to status request messages.
Alarm Condition count	Number of times the interface entered the alarm condition.
Good events	Number of times the network responded to a status request message.

Table 14-10: Parameters in the output of the **show framerelay lmi** command (Continued)

Error	Meaning
Error Events	Number of times the network failed to respond to a status request message.
Current Sequence	Sequence number of the last status request message sent to the network.
Receive Sequence	Sequence number of the last status message received from the network.

Figure 14-22: Example output from the **show framerelay utilisation** command

```
Utilisation: In %  Out %  Bytes: In    Out  Bandwidth:48000
-------------------------------------------------------------
fr0
  Dlc 0020   15     10           900    00
  Dlc 0021   24     13          1440   780
  Overall    39     23
-------------------------------------------------------------
```

Table 14-11: Parameters in the output of the **show framerelay utilisation** command

Error	Meaning
Interface	Interface name.
Dlc	DLC identifier.
Utilisation: In %	Utilisation percentage for incoming packets on the DLC.
Utilisation: Out %	Utilisation percentage for outgoing packets on the DLC, which can exceed 100% as an indication that packets are discarded from the transmission queue.
Bytes: In	Number of bytes received on the DLC.
Bytes: Out	Number of bytes transmitted on the DLC (not number of bytes over the physical layer).
Bandwidth	Total available bandwidth in bytes on the DLC.
Overall	Percentage utilisation for the entire interface.

Examples To display summary information about all Frame Relay interfaces, use the command:

```
sh fr
```

To display the configuration of Frame Relay interface 0, use the command:

```
sh fr=0 conf
```

To display the counters for Frame Relay interface 2, use the command:

```
sh fr=2 cou
```

Related Commands

create framerelay
destroy framerelay
disable framerelay
enable framerelay
reset framerelay
set framerelay
show framerelay dlc
show framerelay li

show framerelay dlc

Syntax `SHow FRamerelay[=`*`fr-interface`*`] DLC[={ALL|`*`dlci`*`}] [COUnter]`

where:

- *fr-interface* is the number of the Frame Relay interface.
- *dlci* is the identification number of a Frame Relay Data Link Connection (DLC) from 0 to 1023.

Description This command displays information about Data Link Connections for a Frame Relay interface. If an interface is specified, its details are displayed. If an interface is not specified, interface 0 (fr0) is assumed.

If **dlc** is not specified, summary information about all DLCs on the router or interface is displayed (Figure 14-23 on page 14-62, Table 14-12 on page 14-62).

If a **all** or **dlci** is specified for the **dlc** parameter, information about the specific DLC or all DLCs is displayed (Figure 14-24 on page 14-63, Table 14-13 on page 14-64).

The **counter** parameter displays counter information for a specific DLC and *dlci* must be specified (Figure 14-25 on page 14-65, Table 14-14 on page 14-66).

Figure 14-23: Example output from the **show framerelay dlc** command

```
Interface  DLC   LI   State                ECP        CCP
-----------------------------------------------------------------
fr0        0013  0    Active               -          OPENED
           0021  0    Active               OPENED     -
           1023  0    Active               -          -
-----------------------------------------------------------------
```

Table 14-12: Parameters in the output of the **show framerelay dlc** command

Parameter	Meaning	
Interface	Interface name.	
DLC	DLC identifier.	
LI	Logical interface to which the DLC is assigned.	
State	State of the DLC: Inactive	
	Await_LMI	
	Disabled	
	Enc_Starting	
	Active	
	Congested	Mild
	Congested	Severe
	Discarding	
	Await_LMI	
ECP	State of the DLC's Encryption Control Protocol: Inactive	

Table 14-12: Parameters in the output of the **show framerelay dlc** command (Continued)

Parameter	Meaning
	Starting
	Closed
	Stopped
	Closing
	Stopping
	Req
	Sent
	ACK Rcvd
	ACK Sent
	Opened
	Does not exist (-)
CCP	State of the DLC's Compression Control Protocol: Inactive
	Starting
	Closed
	Stopped
	Closing
	Stopping
	Req
	Sent
	ACK Rcvd
	ACK Sent
	Opened
	Does not exist (-)

Figure 14-24: Example output from the **show framerelay dlc** command for a specific DLC

```
Interface
  Parameter                          Value
-----------------------------------------------
fr0
  Dlc Index ....................... 20
  State ........................... Active
  Type ............................ Lmi
  Logical Interface ............... 1
  Excess Information Rate (bps) ..... 9600
  Committed Information Rate (bps)... 9600
  Encapsulation ..................... Def_IETF
  Encryption ....................... On
  Compression ...................... Def_On
-----------------------------------------------
```

Table 14-13: Parameters in the output of the **show framerelay dlc** command for a specific DLC

Parameter	Meaning
Interface	Interface name.
DLC Index	DLC identifier.
State	State of the DLC: Inactive
	Await_LMI
	Disabled
	Enc_Starting
	Active
	Congested Mild
	Congested Severe
	Discarding
Type	Whether the type of DLC is Lmi or Static.
Logical Interface	Logical interface to which the DLC is assigned.
Excess Information Rate	Data rate limit above the CIR at which packets can be transmitted over the DLC.
Committed Information Rate	Rate at which packets may be transmitted over the DLC.
Encapsulation	Encapsulation used on the interface: Def_IETF
	Def_CISCO
	IETF
	CISCO
Encryption	DLC encryption:
	Def_On with the interface default **on**
	Def_Off with the interface default **off**
	On
	Off
Compression	DLC compression:
	Def_On with the interface default **on**
	Def_Off with the interface default **off**
	On
	Off

Figure 14-25: Example output from the **show framerelay dlc counter** command

```
Interface: fr0
DLCI: 13
  Created                        0     Last Changed                1000

  inOctets                    7886     outOctets                   4405
  inFrames                     188     outFrames                    149
  BECNs Received                 0     FECNs Received                 0

  notEnabled                     0     ECPNotOpen                     0
  congestionDiscards             0

CCP Counter:
  inOctets                    6768     outOctets                   3584
  inUserPkts                    74     outUserPkts                   76
  inDataDiscard                 40     outDataDiscard                 0
  inDataError                   23     outDataError                   0
  inControlPkts                 50     outControlPkts                73
  inControlError                 0     outControlError                0
  inConfigureRequest             5     outConfigureRequest            6
  inConfigureAcknowledge         4     outConfigureAcknowledge        4
  inConfigureNAK                 1     outConfigureNAK                0
  inConfigureReject              0     outConfigureReject             0
  inTerminateRequest             0     outTerminateRequest            0
  inTerminateAcknowledge         0     outTerminateAcknowledge        0
  inCodeReject                   0     outCodeReject                  0
  inResetRequests                0     outResetRequests              63
  inResetAcks                   40     outResetAcks                   0
  eventIgnored                   0     illegalAction                  1
  goodTLU                        4     goodTLD                        3
  goodTLS                        1     goodTLF                        0
  goodAttatch                    2     badAttatch                     0
  attatchFail                    0     badUserID                      0
  eChanDetach                    0     dChanDetach                    0
  compGo                         0     compHalt                       0
  decompGo                      24     decompHalt                    23
  compReset                      0     sendRR                        23
  sendRA                         0     badRAIn                       20
  resetNotOpen                   0     compPktDiscarded               0
  encodeSuccesses               74     encodeFailures                 0
  decodeSuccesses               34     decodeFailures                 0
```

Table 14-14: Parameters in the output of the **show framerelay dlc counter** command

Parameter	Meaning
Interface	Interface name.
DLCI	DLC identifier.
Created	Time the DLC was created.
Last changed	Time of the last change to the status of the DLC.
In Octets	Number of octets received from the Frame Relay network over the DLC.
OutOctets	Number of octets transmitted to the Frame Relay network over the DLC.
In Frames	Number of frames received from the Frame Relay network over the DLC.
OutFrames	Number of frames transmitted to the Frame Relay network over the DLC.
BECNs Received	Number of frames received from the Frame Relay network over the DLC with the BECN bit set.
FECNs Received	Number of frames received from the Frame Relay network over the DLC with the FECN bit set.
notEnabled	Number of frames received from the Frame Relay network over the DLC while the DLC was locally disabled.
ECPNotOpen	Number of frames received from the Frame Relay network over the DLC while the DLC's ECP was not in the OPENED state.
congestionDiscards	Number of frames passed to Frame Relay by a higher layer that were discarded because the Frame Relay network was congested.
inOctets	Number of octets received by the control protocol.
outOctets	Number of octets transmitted by the control protocol.
inUserPkts	Number of user packets received by the control protocol.
outUserPkts	Number of user packets transmitted by the control protocol.
inDataDiscard	Number of inbound user packets discarded by the control protocol.
outDataDiscard	Number of outbound user packets discarded by the control protocol.
inDataError	Number of inbound user packets that contained errors preventing them from being delivered to a higher-layer protocol.
outDataError	Number of outbound user packets that contained errors preventing them from being transmitted.
inControlPkts	Number of inbound control protocol packets discarded by the control protocol.
outControlPkts	Number of outbound control protocol packets discarded by the control protocol.
inControlError	Number of inbound control protocol packets that contained errors preventing their use.

Table 14-14: Parameters in the output of the **show framerelay dlc counter** command (Continued)

Parameter	Meaning
outControlError	Number of outbound control protocol packets that contained errors preventing them from being transmitted.
inConfigureRequest	Number of Configure Request packets received by the control protocol.
outConfigureRequest	Number of Configure Request packets transmitted by the control protocol.
inConfigureAcknowledge	Number of Configure Acknowledge packets received by the control protocol.
outConfigureAcknowledge	Number of Configure Acknowledge packets transmitted by the control protocol.
inConfigureNAK	Number of Configure NAK packets received by the control protocol.
outConfigureNAK	Number of Configure NAK packets transmitted by the control protocol.
inConfigureReject	Number of Configure Reject packets received by the control protocol.
outConfigureReject	Number of Configure Reject packets transmitted by the control protocol.
inTerminateRequest	Number of Terminate Request packets received by the control protocol.
outTerminateRequest	Number of Terminate Request packets transmitted by the control protocol.
inTerminateAcknowledge	Number of terminate Acknowledge packets received by the control protocol.
outTerminateAcknowledge	Number of terminate Acknowledge packets transmitted by the control protocol.
inCodeReject	Number of Code Reject packets received by the control protocol.
outCodeReject	Number of Code Reject packets transmitted by the control protocol.
inResetRequests	Number of Reset Request packets received by the control protocol.
outResetRequests	Number of Reset Request packets transmitted by the control protocol.
inResetAcks	Number of Reset Acknowledge packets received by the control protocol.
outResetAcks	Number of Reset Acknowledge packets transmitted by the control protocol.
eventIgnored	Number of control protocol state machine events ignored because the DLC was not enabled.
illegalAction	Number of control protocol state machine actions ignored because they were illegal.
goodTLU	Number of times the control protocol successfully entered the opened state.
goodTLD	Number of times the control protocol successfully left the opened state.

Table 14-14: Parameters in the output of the **show framerelay dlc counter** command (Continued)

Parameter	Meaning
goodTLS	Number of times the control protocol successfully entered the starting state.
goodTLF	Number of times the control protocol successfully entered the closed or stopped state.
goodAttach	Number of times the control protocol successfully attached to the encryption/compression facility.
badAttach	Number of times the control protocol failed to attach to a decoding channel of the encryption/compression facility.
attachFail	Number of times the control protocol failed to attach to an encoding channel of the encryption/compression facility.
badUserID	Number of times an error was found with the control protocol's user ID.
eChanDetach	Number of times the control protocol detached from an encoding channel of the encryption/compression facility.
dChanDetach	Number of times the control protocol detached from a decoding channel of the encryption/compression facility.
compGo	Number of "compression go" events received by the control protocol from the encryption/compression facility.
compHalt	Number of "compression halt" events received by the control protocol from the encryption/compression facility.
decompGo	Number of "decompression go" events received by the control protocol from the encryption/compression facility.
decompHalt	Number of "decompression halt" events received by the control protocol from the encryption/compression facility.
compReset	Number of "reset compression" events received by the control protocol from the encryption/compression facility.
sendRR	Number of Reset Request packets sent by the control protocol in response to a request from the encryption/ compression facility.
sendRA	Number of Reset Acknowledge packets sent by the control protocol in response to a request from the encryption/ compression facility.
badRAIn	Number of Reset Acknowledge packets received that did not match the currently outstanding Reset Request packet.
resetNotOpen	Number of reset packets received while the control protocol was not in the OPENED state.
compPktDiscarded	Number of packets discarded because they were marked as both encrypted and compressed.
encodeSuccesses	Number of packets successfully encoded.
decodeSuccesses	Number of packets successfully decoded.
encodeFailures	Number of packets that failed to encode.
decodeFailures	Number of packets that failed to decode.

Examples

To display summary information about all DLCs on the router, use the command:

```
sh fr dlc
```

To display summary information about all DLCs on Frame Relay interface 0, use the command:

```
sh fr=0 dlc
```

To display detailed information about DLC 8 on Frame Relay interface 1, use the command:

```
sh fr=1 dlc=8
```

To display the counters for DLC 8 on Frame Relay interface 1, use the command:

```
sh fr=1 dlc=8 cou
```

Related Commands

add framerelay dlc
delete framerelay dlc
disable framerelay dlc
enable framerelay dlc
reset framerelay dlc
set framerelay dlc
show framerelay

show framerelay li

Syntax

```
SHow FRamerelay[=fr-interface] LI[={ALL|
   logical-interface}] [COUnter]
```

where:

- *fr-interface* is the number of the Frame Relay interface.
- *logical-interface* is the number of the logical interface from 1 to 1023.

Description

This command displays information about Frame Relay logical interfaces. If a Frame Relay interface is specified, then only information about the specified interface is displayed. If a Frame Relay interface is not specified, information about all interfaces is displayed (Figure 14-26 on page 14-70, Table 14-15 on page 14-71).

The **li** parameter specifies the logical interface about which information is to be displayed. If a logical interface is not specified, summary information about all logical interfaces defined for the Frame Relay interface is displayed. If a logical interface or **all** is specified detailed information about the specified logical interface or all logical interfaces is displayed (Figure 14-27 on page 14-71, Table 14-16 on page 14-71).

The **counter** parameter specifies that counter information about a logical interface is to be displayed (Figure 14-28 on page 14-72, Table 14-17 on page 14-72). A logical interface (other than **all**) must be specified with the **counter** parameter.

Figure 14-26: Example output from the **show framerelay li** command

```
Interface  LI   IfIndex   Type   Number of DLCs   User Modules
-------------------------------------------------------------
fr0        All                                   IPX
           0       06     NBMA   4                IP
                                                  ARP
           1       07     PTP    1                IP
                                                  ARP
           2       08     PTP    1                IP
                                                  ARP
           3       10     PTP    1                IP
                                                  ARP
           7       11     NBMA   3                IP
                                                  ARP

fr1        0       09     NBMA   3                IP
                                                  ARP
           2       12     PTP    1                IP
                                                  ARP
           3       13     NBMA   1                IP
                                                  ARP
-------------------------------------------------------------
```

Table 14-15: Parameters in the output of the **show framerelay li** command

Parameter	Meaning
Interface	Interface name.
LI	An LI number or "all".
IfIndex	Value of ifIndex for the LI.
Type	Whether the type is PTP or NBMA.
Number of DLCs	Number of DLCs associated with the LI.
User Modules	Routing modules configured to use the Frame Relay interface.

Figure 14-27: Example output from the **show framerelay li** command for a specific LI

```
Interface  LI   IfIndex   Type   DLCs    User Modules
-----------------------------------------------------------
fr0        0    06        NBMA   0021    IP
                                 0022    ARP
                                 0023
-----------------------------------------------------------
```

Table 14-16: Parameters in the output of the **show framerelay li** command for a specific LI

Parameter	Meaning
Interface	Interface name.
LI	Specific LI number or All.
IfIndex	The value of ifIndex for the LI.
Type	Whether the type is PTP or NBMA.
DLCs	List of DLCs associated with the LI.
User Modules	Routing modules configured to use the Frame Relay interface.

Figure 14-28: Example output from the **show framerelay li counter** command

```
fr0.0          74378 seconds     Last change at:    0 seconds
Li Counter
  ifInOctets              549     ifOutOctets              469
  ifInUcastPkts            17     ifOutUcastPkts            20
  ifInNUcastPkts            0     ifOutNUcastPkts            0
  ifInDiscards              0     ifOutDiscards              0
  ifInErrors                0     ifOutErrors                0
  ifInUnknownProtos         0     ifOutQLen                  0
```

Table 14-17: Parameters in the output of the **show framerelay li counter** command

Parameter	Meaning
ifInOctets	Number of octets received on the logical interface, including framing characters.
ifOutOctets	Number of octets transmitted out of the logical interface, including framing characters.
ifInUcastPkts	Number of subnetwork-unicast frames delivered by the logical interface to a higher-layer protocol.
ifOutUcastPkts	Number of frames that higher-layer protocols requested be transmitted to a subnetwork-unicast address by the logical interface, including those that were discarded or not sent.
ifInNUcastPkts	Number of non-unicast frames delivered by the logical interface to a higher-layer protocol.
ifOutNUcastPkts	Number of frames that higher-layer protocols requested be transmitted to a non-unicast address by the logical interface, including those that were discarded or not sent.
ifInDiscards	Number of inbound frames determined to be associated with the logical interface that are to be discarded even though no errors were detected to prevent their being deliverable to a higher-layer protocol. One possible reason for discarding such a frame could be to free up buffer space.
ifOutDiscards	Number of outbound frames determined to be associated with the logical interface that are to be discarded even though no errors were detected to prevent their being transmitted. One possible reason for discarding such a frame could be to free up buffer space.
ifInErrors	Number of inbound frames determined to be associated with the logical interface that contained errors preventing them from being deliverable to a higher-layer protocol.
ifOutErrors	Number of outbound frames determined to be associated with the logical interface that contained errors preventing them from being transmitted.
ifInUnknownProtos	Number of frames received via the logical interface that were discarded because of an unknown or unsupported protocol.
ifOutQLen	Length of the output frame queue.

Examples To display summary information about all LIs on the router, use the command:

```
sh fr li
```

To display summary information about all LIs on Frame Relay interface 0, use the command:

```
sh fr=0 li
```

To display detailed information about LI 8 on Frame Relay interface 1, use the command:

```
sh fr=1 li=8
```

To display the counters for LI 8 on Frame Relay interface 1, use the command:

```
sh fr=1 li=8 cou
```

Related Commands

add framerelay li
create framerelay
delete framerelay li
disable framerelay li debug
enable framerelay li debug