This article focuses on the implementation of packet-switching services and addresses internetwork design in terms of the following packet-switching service topics:

- Hierarchical internetwork design
- Topology design
- Broadcast issues
- Performance issues

Information provided in this article is organized around these central topics. An introductory discussion outlines the general issues; subsequent discussions focus on considerations for the specific packet-switching technologies.

Note: This article focuses on general packet-switching considerations and Frame Relay internetworks. Frame Relay was selected as the focus for this a because it presents a comprehensive illustration of design considerations for interconnection to packet-switching services.

Guide Contents
Internetworking Design Basics
Designing various internetworks
Network Enhancements
IP Routing Concepts
UDP Broadcast Flooding
Large-Scale H.323 Network Design for Service Providers
LAN Switching
Subnetting an IP Address Space
IBM Serial Link Implementation Notes
References and Recommended Reading

Contents

- 1 Understanding Packet-Switched Internetwork Design
  - 1.1 Hierarchical Design
    - 1.1.1 Figure: Hierarchical packet-switched interconnection
    - 1.1.2 Scalability of Hierarchical Internetworks
    - 1.1.3 Manageability of Hierarchical Internetworks
    - 1.1.4 Optimization of Broadcast and Multicast Control Traffic
  - 1.2 Topology Design
    - 1.2.1 Star Topologies
      - 1.2.1.1 Figure: Star topology for a regional internetwork
    - 1.2.2 Fully Meshed Topologies
      - 1.2.2.1 Figure: Fully meshed topology
    - 1.2.3 Partially Meshed Topologies
      - 1.2.3.1 Figure: Partially meshed topology
  - 1.3 Broadcast Issues
    - 1.3.1 Table: Broadcast Traffic Levels of Protocols in Large-Scale Internetworks
  - 1.4 Performance Issues
- 2 Frame Relay Internetwork Design
  - 2.1 Hierarchical Design for Frame Relay Internetworks

- ◊ 2.1.1 Hierarchical Meshed Frame Relay Internetworks
  - · 2.1.1.1 Figure: Fully meshed hierarchical Frame Relay environment
- ♦ 2.2 Hybrid Meshed Frame Relay Internetworks
  - ◊ 2.2.1 Figure: Hybrid hierarchical Frame Relay internetwork
- ♦ 2.3 Regional Topologies for Frame Relay Internetworks
  - ◊ 2.3.1 Star Topologies
  - ◊ 2.3.2 Fully Meshed Topologies
    - · 2.3.2.1 Figure: Fully meshed Frame Relay
  - ◊ 2.3.3 Partially Meshed Topologies
    - · 2.3.3.1 Figure: Twin-star, partially meshed Frame Relay internetwork
    - · 2.3.3.2 Figure: Partially meshed Frame Relay internetwork
- ♦ 2.4 Broadcast Issues for Frame Relay Internetworks
  - ◊ 2.4.1 Figure: SAP replication in Frame Relay virtual interface environment
- ♦ 2.5 Creating a Broadcast Queue for an Interface
- ♦ 2.6 Performance Issues for Frame Relay Internetworks
  - ◊ 2.6.1 Packet-Switched Service Provider Tariff Metrics
    - · 2.6.1.1 Figure: Example of a CIR and CBR traffic limiting situation
  - ◊ 2.6.2 Multiprotocol Traffic Management Requirements
    - · 2.6.2.1 Figure: Virtual interfaces assigned specific protocols
    - · 2.6.2.2 Figure: Virtual interface configuration example

- 3 Configuring Frame Relay Traffic Shaping
- 4 Summary

Understanding Packet-Switched Internetwork Design

The chief trade-off in linking local-area networks (LANs) and private wide-area networks (WANs) into packet-switching data network (PSDN) services is between cost and performance. An ideal design optimizes packet-services. Service optimization does not necessarily translate into picking the service mix that represents the lowest possible tariffs. Successful packet-service implementations result from adhering to two basic rules:

- When implementing a packet-switching solution, be sure to balance cost savings derived by instituting PSDN interconnections with your computing community's performance requirements.
- Build an environment that is manageable and that can scale up as more WAN links are required.

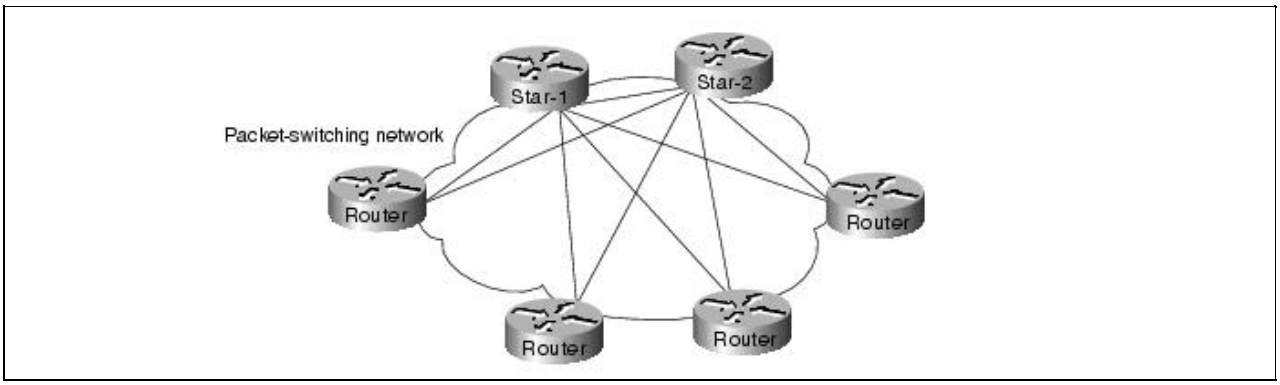
These rules recur as underlying themes in the discussions that follow. The introductory sections outline the overall issues that influence the ways in which packet-switched internetworks are designed.

Hierarchical Design

The objective of a hierarchical internetwork design is to modularize the elements of a large internetwork into layers of internetworking. The general model of this hierarchy is described in Internetworking Design Basics. The key functional layers in this model are the access, distribution, and backbone (or core) routing layers. In essence, a hierarchical approach strives to split networks into subnetworks so that traffic and nodes can be

more easily managed. Hierarchical designs also facilitate scaling of internetworks because new subnetwork modules and internetworking technologies can be integrated into the overall scheme without disrupting the existing backbone. Figure: Hierarchical packet-switched interconnection illustrates the basic approach to hierarchical design.

Figure: Hierarchical packet-switched interconnection



Three basic advantages tilt the design decision in favor of a hierarchical approach:

- Scalability of Hierarchical Internetworks
- Manageability of Hierarchical Internetworks
- Optimization of Broadcast and Multicast Control Traffic

Scalability of Hierarchical Internetworks

Scalability is a primary advantage that supports using a hierarchical approach to packet-service connections. Hierarchical internetworks are more scalable because they allow you to grow your internetwork in incremental modules without running into the limitations that are quickly encountered with a flat, nonhierarchical structure.

However, hierarchical internetworks raise certain issues that require careful planning. These issues include the costs of virtual circuits, the complexity inherent in a hierarchical design (particularly when integrated with a meshed topology), and the need for additional router interfaces to separate layers in your hierarchy.

To take advantage of a hierarchical design, you must match your hierarchy of internetworks with a complementary approach in your regional topologies. Design specifics depend on the packet services you implement, as well as your requirements for fault tolerance, cost, and overall performance.

Manageability of Hierarchical Internetworks

Hierarchical designs offer several management advantages:

- Internetwork simplicity-Adopting a hierarchical design reduces the overall complexity of an internetwork by partitioning elements into smaller units. This partitioning of elements makes troubleshooting easier, while providing inherent protection against the propagation of broadcast storms, routing loops, or other potential problems.
- Design flexibility-Hierarchical internetwork designs provide greater flexibility in the use of WAN packet services. Most internetworks benefit from using a hybrid approach to the overall internetwork structure. In many cases, leased lines can be implemented in the backbone, with packet-switching services used in the distribution and access internetworks.
- Router management-With the use of a layered, hierarchical approach to router implementation, the

complexity of individual router configurations is substantially reduced because each router has fewer neighbors or peers with which to communicate.

Optimization of Broadcast and Multicast Control Traffic

The effect of broadcasting in packet-service networks (discussed in "Broadcast Issues" later in this article) require you to implement smaller groups of routers. Typical examples of broadcast traffic are the routing updates and Novell Service Advertisement Protocol (SAP) updates that are broadcast between routers on a PSDN. An excessively high population of routers in any area or layer of the overall internetwork might result in traffic bottlenecks brought on by broadcast replication. A hierarchical scheme allows you to limit the level of broadcasting between regions and into your backbone.

Topology Design

After you have established your overall internetwork scheme, you must settle on an approach for handling interconnections among sites within the same administrative region or area. In designing any regional WAN, whether it is based on packet-switching services or point-to-point interconnections, there are three basic design approaches that you can adopt:

- Star Topologies
- Fully Meshed Topologies
- Partially Meshed Topologies

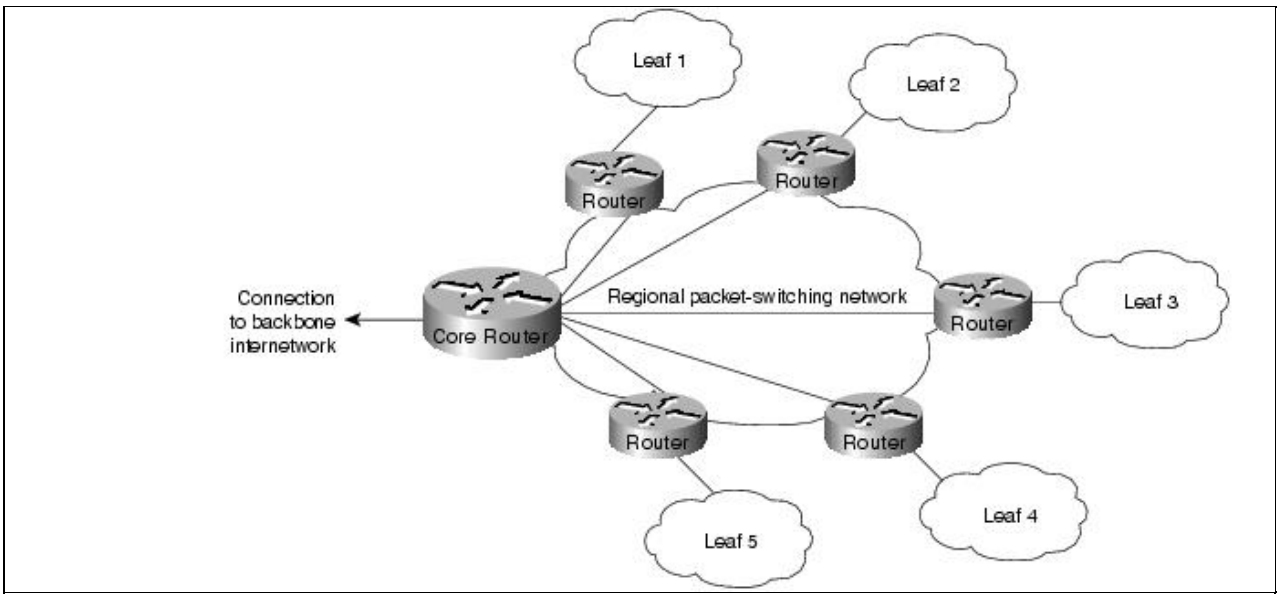
The following discussions introduce these topologies. Technology-specific discussions presented in this article address the applicability of these topologies for the specific packet-switching services.

Note: Illustrations in this article use lines to show the interconnections of specific routers on the PSDN network. These interconnections are virtual connections, facilitated by mapping features within the routers. Actual physical connections generally are made to switches within the PSDN. Unless otherwise specified, the connecting lines represent these virtual connections in the PSDN.

Star Topologies

A star topology features a single internetworking hub providing access from leaf internetworks into the backbone and access to each other only through the core router. Figure: Star topology for a regional internetwork illustrates a packet-switched star topology for a regional internetwork.

Figure: Star topology for a regional internetwork

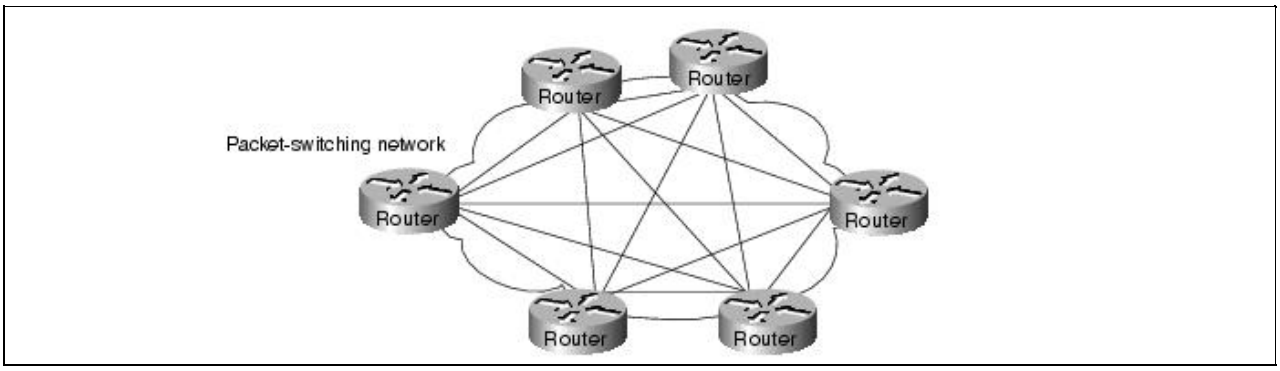


The advantages of a star approach are simplified management and minimized tariff costs. However, the disadvantages are significant. First, the core router represents a single point of failure. Second, the core router limits overall performance for access to backbone resources because it is a single pipe through which all traffic intended for the backbone (or for the other regional routers) must pass. Third, this topology is not scalable.

Fully Meshed Topologies

A fully meshed topology means that each routing node on the periphery of a given packet-switching network has a direct path to every other node on the cloud. Figure: Fully meshed topology illustrates this kind of arrangement.

Figure: Fully meshed topology



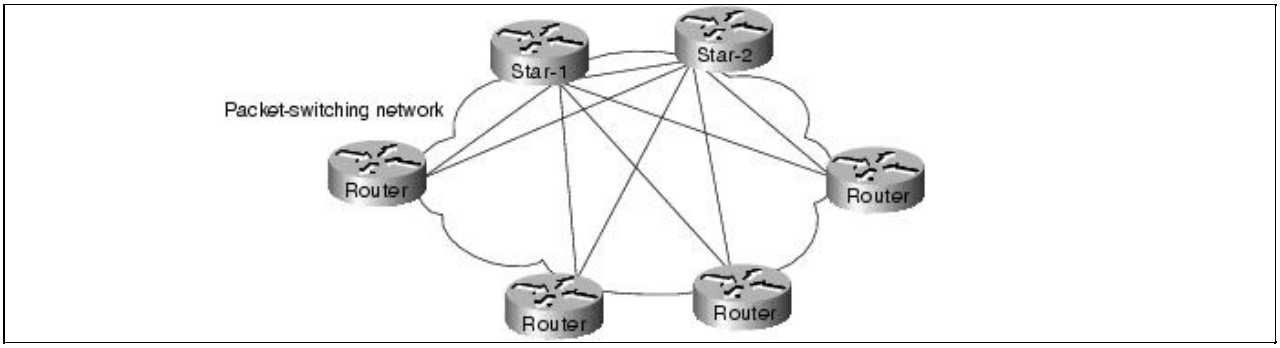
The key rationale for creating a fully meshed environment is to provide a high level of redundancy. Although a fully meshed topology facilitates support of all network protocols, it is not tenable in large packet-switched internetworks. Key issues are the large number of virtual circuits required (one for every connection between routers), problems associated with the large number of packet/broadcast replications required, and the configuration complexity for routers in the absence of multicast support in nonbroadcast environments.

By combining fully meshed and star approaches into a partially meshed environment, you can improve fault tolerance without encountering the performance and management problems associated with a fully meshed approach. The next section discusses the partially meshed approach.

Partially Meshed Topologies

A partially meshed topology reduces the number of routers within a region that have direct connections to all other nodes in the region. All nodes are not connected to all other nodes. For a nonmeshed node to communicate with another nonmeshed node, it must send traffic through one of the collection point routers. Figure: Partially meshed topology illustrates such a situation.

Figure: Partially meshed topology



There are many forms of partially meshed topologies. In general, partially meshed approaches are considered to provide the best balance for regional topologies in terms of the number of virtual circuits, redundancy, and performance.

Broadcast Issues

The existence of broadcast traffic can present problems when introduced into packet-service internetworks. Broadcasts are necessary for a station to reach multiple stations with a single packet when the specific address of each intended recipient is not known by the sending node. Table: Broadcast Traffic Levels of Protocols in Large-Scale Internetworks lists common networking protocols and the general level of broadcast traffic associated with each, assuming a large-scale internetwork with many routing nodes.

Table: Broadcast Traffic Levels of Protocols in Large-Scale Internetworks

Network Protocol	Routing Protocol	Relative Broadcast Traffic Level
AppleTalk	Routing Table Maintenance Protocol (RTMP)	High
	Enhanced Interior Gateway Routing Protocol (Enhanced IGRP)	Low
Novell Internetwork Packet Exchange (IPX)	Routing Information Protocol (RIP)	High
	Service Advertisement Protocol (SAP)	High
	Enhanced IGRP	Low
Internet Protocol (IP)	RIP	High

	Interior Gateway Routing Protocol (IGRP)	High
	Open Shortest Path First (OSPF)	Low
	Intermediate System-to-Intermediate System (IS-IS)	Low
	Enhanced IGRP	Low
	Border Gateway Protocol (BGP)	None
	Exterior Gateway Protocol (EGP)	None
DECnet Phase IV	DECnet Routing	High
DECnet Phase V	IS-IS	Low
International Organization for Standardization (ISO) Connectionless Network Service (CLNS)	IS-IS	Low
	ISO-IGRP	High
Xerox Network Systems (XNS)	RIP	High
Banyan Virtual Integrated Network Service (VINES)	Routing Table Protocol (RTP)	High
	Sequenced RTP	Low

The relative values high and low in Table: Broadcast Traffic Levels of Protocols in Large-Scale Internetworks provide a general range for these protocols. Your situation and implementation will determine the magnitude of broadcast traffic. For example, the level of broadcast traffic generated in an AppleTalk Enhanced IGRP environment depends on the setting of the Enhanced IGRP hello-timer interval. Another issue relates to the size of the internetwork. In a small-scale internetwork, the amount of broadcast traffic generated by Enhanced IGRP nodes might be higher than with a comparable RTMP-based internetwork. However, for large-scale internetworks, Enhanced IGRP nodes generate substantially less broadcast traffic than RTMP-based nodes.

Managing packet replication is an important design consideration when integrating broadcast-type LANs (such as Ethernet) with nonbroadcast packet services (such as X.25). With the multiple virtual circuits that are characteristic of connections to packet-switched environments, routers must replicate broadcasts for each virtual circuit on a given physical line.

With highly meshed environments, replicating broadcasts can be expensive in terms of increased required bandwidth and number of CPU cycles. Despite the advantages that meshed topologies offer, they are generally impractical for large packet-switching internetworks. Nonetheless, some level of circuit meshing is essential to ensure fault tolerance. The key is to balance the trade-off in performance with requirements for circuit redundancy.

Performance Issues

When designing a WAN around a specific packet service type, you must consider the individual characteristics of the virtual circuit. For example, performance under certain conditions will depend on a given virtual circuit's capability to accommodate mixed protocol traffic. Depending on how the multiprotocol traffic is queued and streamed from one node to the next, certain protocols may require special handling. One solution might be to assign specific virtual circuits to specific protocol types. Performance concerns for specific packet-switching services include committed information rates (CIR) in Frame Relay internetworks and window size limitations in X.25 internetworks. (The CIR corresponds to the maximum average rate per connection [PVC] for a period of time.)

Frame Relay Internetwork Design

One of the chief concerns when designing a Frame Relay implementation is scalability. As your requirements for remote interconnections grow, your internetwork must be able to grow to accommodate changes. The internetwork must also provide an acceptable level of performance, while minimizing maintenance and management requirements. Meeting all these objectives simultaneously can be quite a balancing act. The discussions that follow focus on several important factors for Frame Relay internetworks:

- Hierarchical design
- Regional topologies
- Broadcast issues
- Performance issues

The guidelines and suggestions that follow are intended to provide a foundation for constructing scalable Frame Relay internetworks that balance performance, fault tolerance, and cost.

Hierarchical Design for Frame Relay Internetworks

In general, the arguments supporting hierarchical design for packet-switching networks discussed in the section "Hierarchical Design" earlier in this article apply to hierarchical design for Frame Relay internetworks. To review, the three factors driving the recommendation for implementing a hierarchical design are the following:

- Scalability of hierarchical internetworks
- Manageability of hierarchical internetworks
- Optimization of broadcast and multicast control traffic

The method by which many Frame Relay vendors tariff services is by Data Link Connection Identifier (DLCI), which identifies a Frame Relay permanent virtual connection. A Frame Relay permanent virtual connection is equivalent to an X.25 permanent virtual circuit, which, in X.25 terminology, is identified by a logical channel number (LCN). The DLCI defines the interconnection between Frame Relay elements. For any given internetwork implementation, the number of Frame Relay permanent virtual connections is highly dependent on the protocols in use and actual traffic patterns.

How many DLCIs can be configured per serial port? It varies depending on the traffic level. You can use all of them (about 1,000), but in common use, 200-300 is a typical maximum. If you broadcast on the DLCIs, 30-50 is more realistic due to CPU overhead in generating broadcasts. Specific guidelines are difficult because overhead varies by configuration. However, on low-end boxes (4,500 and below), the architecture is bound by the available I/O memory. The specific number depends on several factors that should be considered together:

- Protocols being routed-Any broadcast-intensive protocol constrains the number of assignable DLCIs. For example, AppleTalk is a protocol that is characterized by high levels of broadcast overhead. Another example is Novell IPX, which sends both routing and service updates resulting in higher broadcast bandwidth overhead. In contrast, IGRP is less broadcast intensive because it sends routing updates less often (by default, every 90 seconds). However, IGRP can become broadcast intensive if its IGRP timers are modified so that updates are sent more frequently.
- Broadcast traffic-Broadcasts, such as routing updates, are the single most important consideration in determining the number of DLCIs that can be defined. The amount and type of broadcast traffic will guide your ability to assign DLCIs within this general recommended range. Refer to Table: Broadcast Traffic Levels of Protocols in Large-Scale Internetworks earlier in this article for a list of the relative level of broadcast traffic associated with common protocols.
- Speed of lines-If broadcast traffic levels are expected to be high, you should consider faster lines and DLCIs with higher CIR and excess burst (Be) limits. You should also implement fewer DLCIs.
- Static routes-If static routing is implemented, you can use a larger number of DLCIs per line, because a larger number of DLCIs reduces the level of broadcasting.
- Size of routing protocol and SAP updates-The larger the internetwork, the larger the size of these updates. The larger the updates, the fewer the number of DLCIs that you can assign.

Two forms of hierarchical design can be implemented:

- Hierarchical Meshed Frame Relay Internetworks
- Hybrid Meshed Frame Relay Internetworks

Both designs have advantages and disadvantages. The brief discussions that follow contrast these two approaches.

Hierarchical Meshed Frame Relay Internetworks

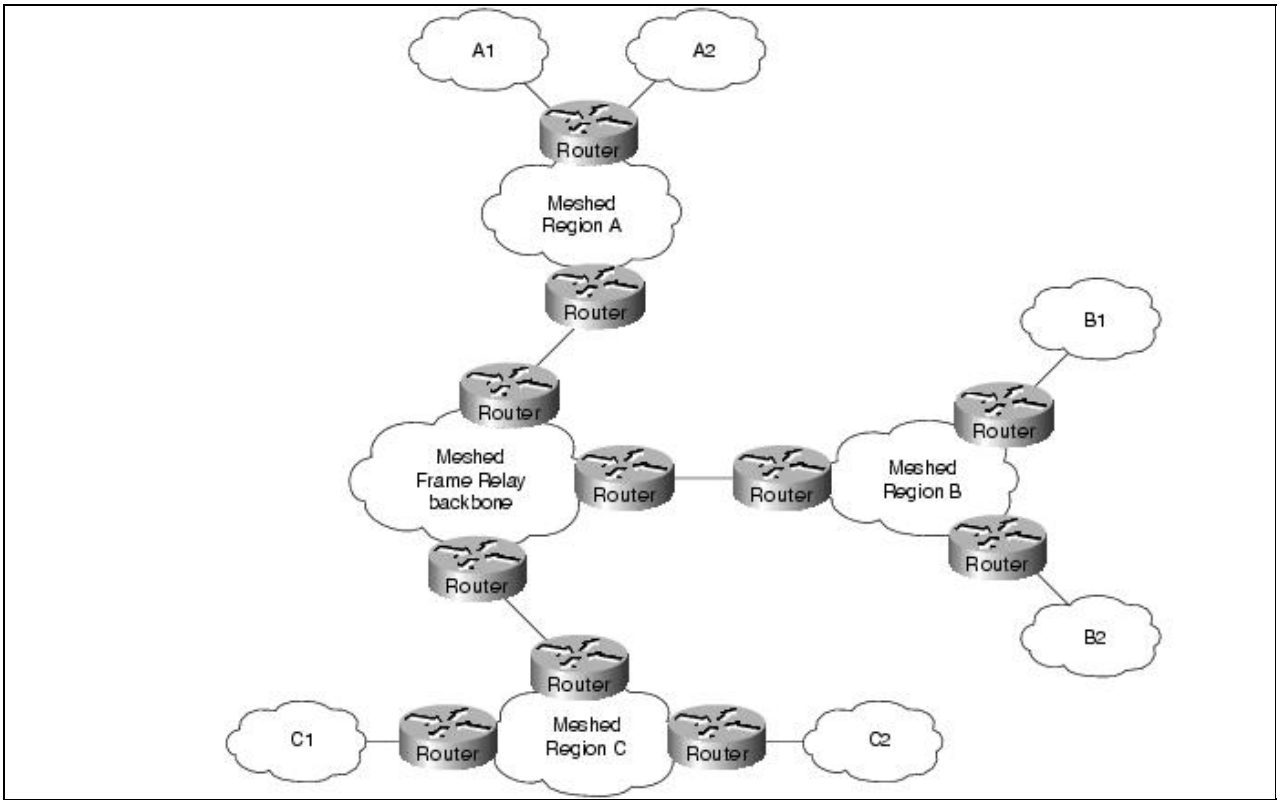
The objectives of implementing a hierarchical mesh for Frame Relay environments are to avoid implementing excessively large numbers of DLCIs and to provide a manageable, segmented environment. The hierarchical meshed environment features full meshing within the core PSDN and full meshing throughout the peripheral internetworks. The hierarchy is created by strategically locating routers between internetwork elements in the hierarchy.

[[Internetwork Design Guide -- Designing Packet Service Internetworks#Figure 5 illustrates a simple hierarchical mesh. The internetwork illustrated in Figure: Fully meshed hierarchical Frame Relay environment illustrates a fully meshed backbone, with meshed regional internetworks and broadcast networks at the outer periphery.

The key advantages of the hierarchical mesh are that it scales well and localizes traffic. By placing routers between fully meshed portions of the internetwork, you limit the number of DLCIs per physical interface, segment your internetwork, and make the internetwork more manageable. However, consider the following two issues when implementing a hierarchical mesh:

- Broadcast and packet replication-In an environment that has a large number of multiple DLCIs per router interface, excessive broadcast and packet replication can impair overall performance. With a high level of meshing throughout a hierarchical mesh, excessive broadcast and packet replication is a significant concern. In the backbone, where traffic throughput requirements are typically high, preventing bandwidth loss due to broadcast traffic and packet replication is particularly important.
- Increased costs associated with additional router interfaces-Compared with a fully meshed topology, additional routers are needed to separate the meshed backbone from the meshed peripheral internetworks. However, by using these routers, you can create much larger internetworks that scale almost indefinitely in comparison to a fully meshed internetwork.

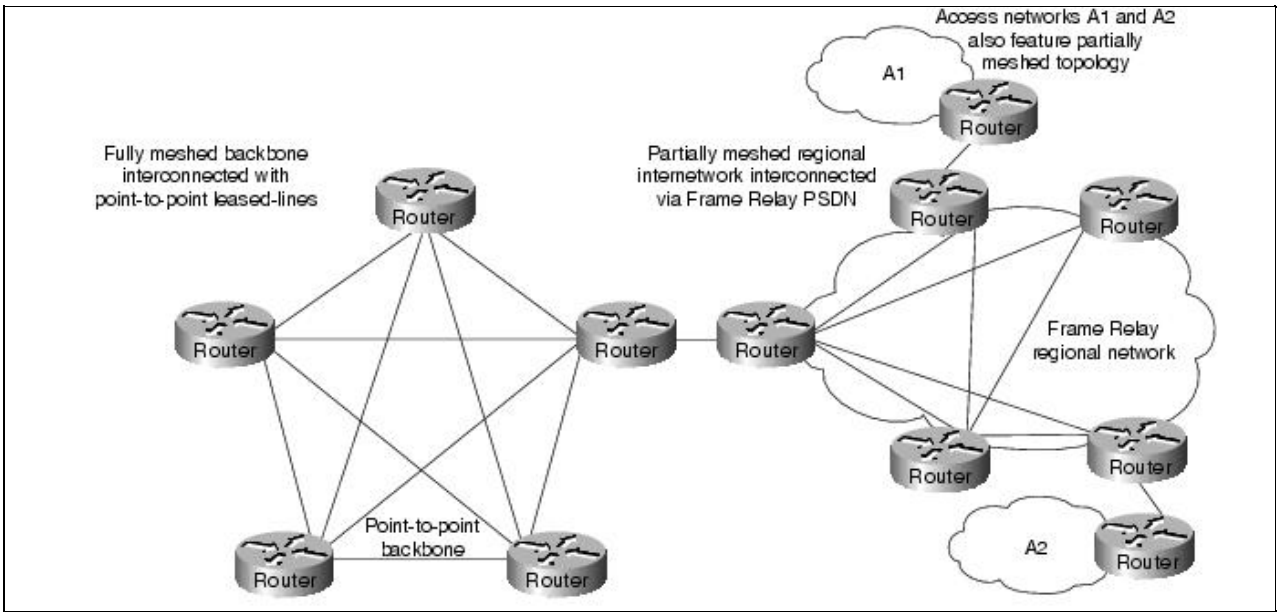
Figure: Fully meshed hierarchical Frame Relay environment



Hybrid Meshed Frame Relay Internetworks

The economic and strategic importance of backbone environments often force internetwork designers to implement a hybrid meshed approach to WAN internetworks. Hybrid meshed internetworks feature redundant, meshed leased lines in the WAN backbone and partially (or fully) meshed Frame Relay PSDNs in the periphery. Routers separate the two elements. Figure: Hybrid hierarchical Frame Relay internetwork illustrates such a hybrid arrangement.

Figure: Hybrid hierarchical Frame Relay internetwork



Hybrid hierarchical meshes have the advantages of providing higher performance on the backbone, localizing traffic, and simplifying scaling of the internetwork. In addition, hybrid meshed internetworks for Frame Relay are attractive because they can provide better traffic control in the backbone and they allow the backbone to be made of dedicated links, resulting in greater stability.

The disadvantages of hybrid hierarchical meshes include high costs associated with the leased lines as well as broadcast and packet replication that can be significant in access internetworks.

Regional Topologies for Frame Relay Internetworks

You can adopt one of three basic design approaches for a Frame Relay-based packet service regional internetwork:

- Star Topologies
- Fully Meshed Topologies
- Partially Meshed Topologies

Each of these is discussed in the following sections. In general, emphasis is placed on partially meshed topologies integrated into a hierarchical environment. Star and fully meshed topologies are discussed for structural context.

Star Topologies

The general form of the star topology is addressed in the section "Topology Design" earlier in this article. Stars are attractive because they minimize the number of DLCIs required and result in a low-cost solution. However, a star topology presents some inherent bandwidth limitations. Consider an environment where a backbone router is attached to a Frame Relay cloud at 256 Kbps, while the remote sites are attached at 56 Kbps. Such a topology will throttle traffic coming off the backbone intended for the remote sites.

As suggested in the general discussion, a strict star topology does not offer the fault tolerance needed for many internetworking situations. If the link from the hub router to a specific leaf router is lost, all connectivity to the leaf router is lost.

Fully Meshed Topologies

A fully meshed topology mandates that every routing node connected to a Frame Relay internetwork is logically linked via an assigned DLCI to every other node on the cloud. This topology is not tenable for larger Frame Relay internetworks for several reasons:

- Large, fully meshed Frame Relay internetworks require many DLCIs. One is required for each logical link between nodes. As shown in Figure: Fully meshed Frame Relay, a fully connected topology requires the assignment of **[n(n-1)]/2** DLCIs, where **n** is the number of routers to be directly connected.

Figure: Fully meshed Frame Relay

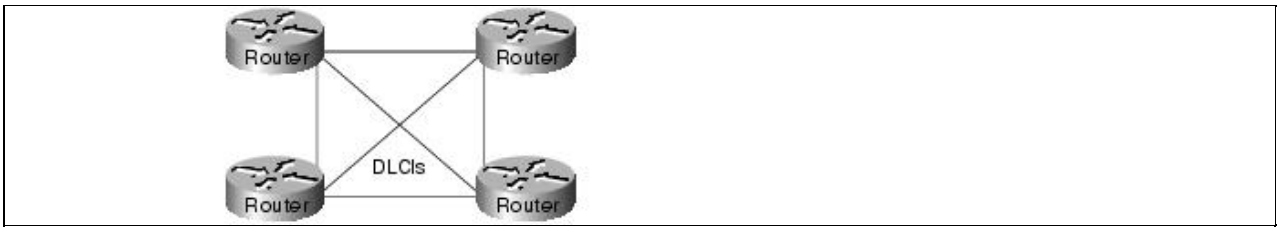


Figure: Hybrid hierarchical Frame Relay internetwork

- Broadcast replication will choke internetworks in large, meshed Frame Relay topologies. Routers inherently treat Frame Relay as a broadcast medium. Each time a router sends a multicast frame (such as a routing update, spanning tree update, or SAP update), the router must copy the frame to each DLCI for that Frame Relay interface.

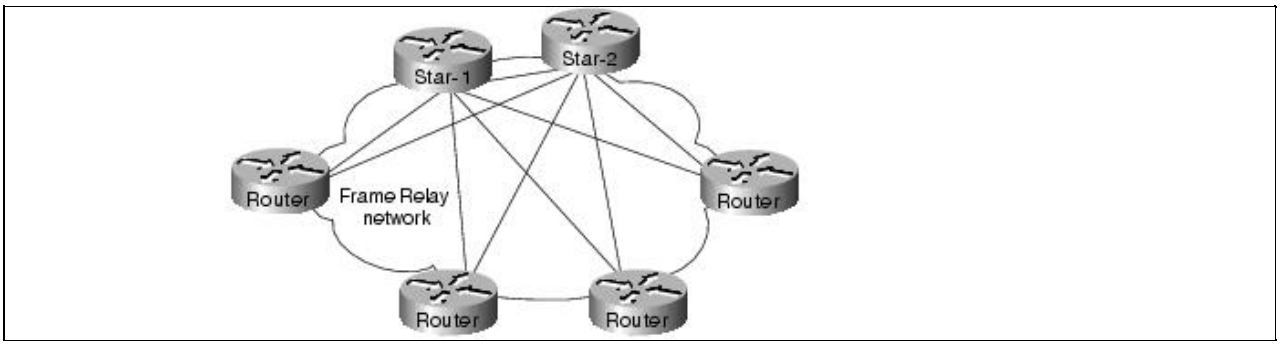
These problems combine to make fully meshed topologies unworkable and unscalable for all but relatively small Frame Relay implementations.

Partially Meshed Topologies

Combining the concepts of the star topology and the fully meshed topology results in the partially meshed topology. Partially meshed topologies are generally recommended for Frame Relay regional environments because they offer superior fault tolerance (through redundant stars) and are less expensive than a fully meshed environment. In general, you should implement the minimum meshing to eliminate single point-of-failure risk.

Figure: Twin-star, partially meshed Frame Relay internetwork illustrates a twin-star, partially meshed approach. This arrangement is supported in Frame Relay internetworks running IP, ISO CLNS, DECnet, Novell IPX, AppleTalk, and bridging.

Figure: Twin-star, partially meshed Frame Relay internetwork

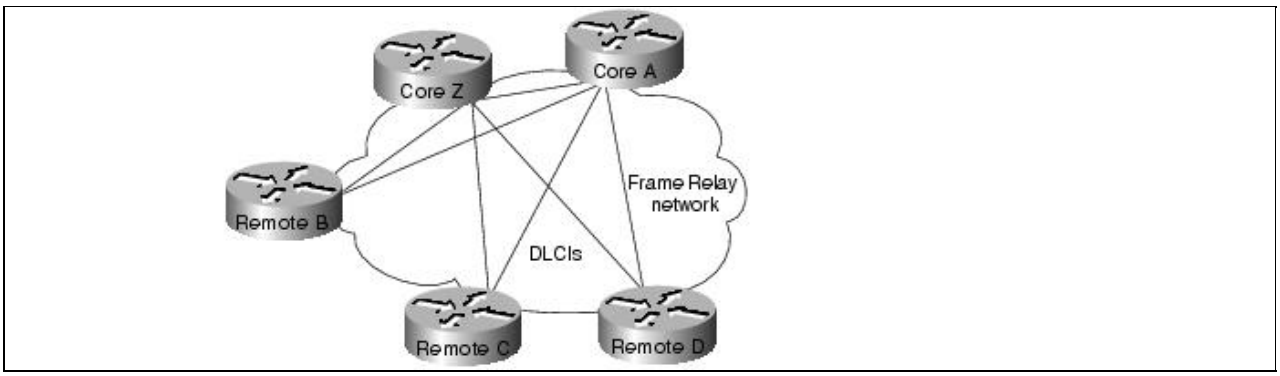


A feature called virtual interfaces (introduced with Software Release 9.21) allows you to create internetworks using partially meshed Frame Relay designs, as shown in Figure: Twin-star, partially meshed Frame Relay internetwork.

To create this type of internetwork, individual physical interfaces are split into multiple virtual (logical) interfaces. The implication for Frame Relay is that DLCIs can be grouped or separated to maximize utility. For example, small fully meshed clouds of Frame Relay-connected routers can travel over a group of four DLCIs clustered on a single virtual interface, whereas a fifth DLCI on a separate virtual interface provides connectivity to a completely separate internetwork. All of this connectivity occurs over a single physical interface connected to the Frame Relay service.

Prior to Software Release 9.21, virtual interfaces were not available and partially meshed topologies posed potential problems, depending on the internetwork protocols used. Consider the topology illustrated in Figure: Partially meshed Frame Relay internetwork.

Figure: Partially meshed Frame Relay internetwork



Given a standard router configuration and router software predating Software Release 9.21, the connectivity available in the internetwork shown in Figure: Partially meshed Frame Relay internetwork can be characterized as follows:

- Core A and Core Z can reach all the remote routers.
- Remote B, Remote C, and Remote D cannot reach each other.

For Frame Relay implementations running software prior to Software Release 9.21, the only way to permit connectivity among all these routers is by using a distance vector routing protocol that can disable split horizon, such as RIP or IGRP for IP. Any other internetwork protocol, such as AppleTalk or ISO CLNS, does not work. The following configuration listing illustrates an IGRP configuration to support a partially meshed arrangement.

```
router igrp 20
network 45.0.0.0
!
interface serial 3
encapsulation frame-relay
ip address 45.1.2.3 255.255.255.0
no ip split-horizon
```

This topology only works with distance vector routing protocols, assuming you want to establish connectivity from Remote B, C, or D to Core A or Core Z, but not across paths. This topology does not work with link state routing protocols because the router cannot verify complete adjacencies. Note that you will see routes and services of the leaf nodes that cannot be reached.

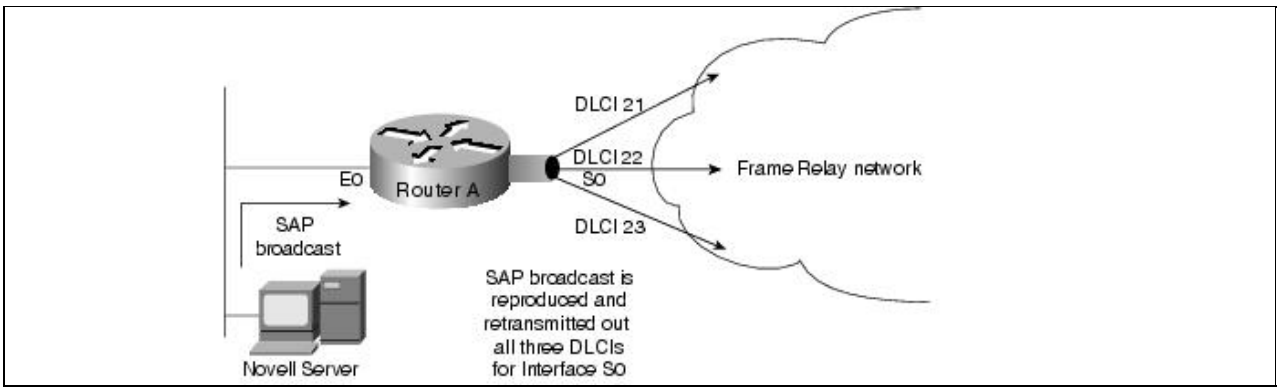
Broadcast Issues for Frame Relay Internetworks

Routers treat Frame Relay as a broadcast media, which means that each time the router sends a multicast frame (such as a routing update, spanning tree update, or SAP update), the router must replicate the frame to each DLCI for the Frame Relay interface. Frame replication results in substantial overhead for the router and for the physical interface.

Consider a Novell IPX environment with multiple DLCIs configured for a single physical serial interface. Every time a SAP update is detected, which occurs every 60 seconds, the router must replicate it and send it down the virtual interface associated with each DLCI. Each SAP frame contains up to seven service entries, and each update is 64 bytes. Figure: SAP replication in Frame Relay virtual interface environment illustrates this situation.

Note: One way to reduce broadcasts is to implement more efficient routing protocols, such as Enhanced IGRP, and to adjust timers on lower speed Frame Relay services.

Figure: SAP replication in Frame Relay virtual interface environment



Creating a Broadcast Queue for an Interface

Very large Frame Relay networks might have performance problems when many DLCIs terminate in a single router or access server that must replicate routing updates and service advertising updates on each DLCI. The updates can consume access-link bandwidth and cause significant latency variations in user traffic; the updates can also consume interface buffers and lead to higher packet rate loss for both user data and routing updates.

To avoid such problems, you can create a special broadcast queue for an interface. The broadcast queue is managed independently of the normal interface queue, has its own buffers, and has a configurable size and service rate.

A broadcast queue is given a maximum transmission rate (throughput) limit measured in both bytes per second and packets per second. The queue is serviced to ensure that no more than this maximum is provided. The broadcast queue has priority when transmitting at a rate below the configured maximum, and hence has a guaranteed minimum bandwidth allocation. The two transmission rate limits are intended to avoid flooding the interface with broadcasts. The actual transmission rate limit in any second is the first of the two rate limits that is reached.

Performance Issues for Frame Relay Internetworks

Two important performance concerns must be addressed when you are implementing a Frame Relay internetwork:

- Packet-Switched Service Provider Tariff Metrics
- Multiprotocol Traffic Management Requirements

Each of these must be considered during the internetwork planning process. The following sections briefly discuss the impact that tariff metrics and multiprotocol traffic management can have on overall Frame Relay performance.

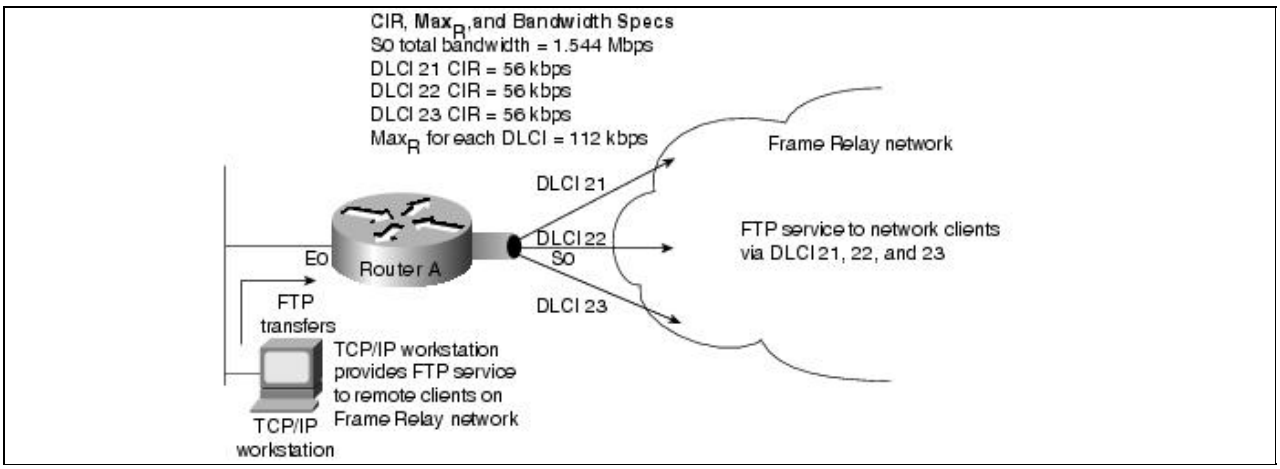
Packet-Switched Service Provider Tariff Metrics

When you contract with Frame Relay packet-switched service providers for specific capabilities, CIR, measured in bits per second, is one of the key negotiated tariff metrics. CIR is the maximum permitted traffic level that the carrier will allow on a specific DLCI into the packet-switching environment. CIR can be anything up to the capacity of the physical limitation of the connecting line.

Other key metrics are committed burst (Bc) and excess burst (Be). Bc is the number of bits that the Frame Relay internetwork is committed to accept and transmit at the CIR. Be sets the absolute limit for a DLCI in bits. This is the number of bits that the Frame Relay internetwork will attempt to transmit after Bc is accommodated. Be determines a peak or maximum Frame Relay data rate (MaxR), where MaxR = (Bc + Be)/Bc * CIR, measured in bits per second.

Consider the situation illustrated in Figure: Example of a CIR and CBR traffic limiting situation. In this environment, DLCIs 21, 22, and 23 are assigned CIRs of 56 Kbps. Assume the MaxR for each line is 112 Kbps (double the CIR). The serial line to which Router A is connected is a T1 line capable of 1.544 Mbps total throughput. Given that the type of traffic being sent into the Frame Relay internetwork consists of FTP file transfers, the potential is high that the router will attempt to transmit at a rate in excess of MaxR. If this occurs, traffic might be dropped without notification if the Be buffers (allocated at the Frame Relay switch) overflow.

Figure: Example of a CIR and CBR traffic limiting situation



Unfortunately, there are relatively few ways to automatically prevent traffic on a line from exceeding the MaxR. Although Frame Relay itself uses the Forward Explicit Congestion Notification (FECN) and Backward Explicit Congestion Notification (BECN) protocols to control traffic in the Frame Relay internetwork, there is no formally standardized mapping between the Frame Relay (link) level and most upper-layer protocols. At this time, an FECN bit detected by a router is mapped to the congestion notification byte for DECnet Phase IV or ISO CLNS. No other protocols are supported.

The actual effect of exceeding specified CIR and derived MaxR settings depends on the types of application running on the internetwork. For instance, TCP/IP's backoff algorithm will see dropped packets as a congestion indication and sending hosts might reduce output. However, NFS has no backoff algorithm, and dropped packets will result in lost connections. When determining the CIR, Bc, and Be for Frame Relay connection, you should consider the actual line speed and applications to be supported.

Most Frame Relay carriers provide an appropriate level of buffering to handle instances when traffic exceeds the CIR for a given DLCI. These buffers allow excess packets to be spooled at the CIR and reduce packet loss, given a robust transport protocol such as TCP. Nonetheless, overflows can happen. Remember that although routers can prioritize traffic, Frame Relay switches cannot. You can specify which Frame Relay packets have low priority or low time sensitivity and will be the first to be dropped when a Frame Relay switch is congested. The mechanism that allows a Frame Relay switch to identify such packets is the discard eligibility (DE) bit.

This feature requires that the Frame Relay network be able to interpret the DE bit. Some networks take no action when the DE bit is set. Other networks use the DE bit to determine which packets to discard. The most

desirable interpretation is to use the DE bit to determine which packets should be dropped first and also which packets have lower time sensitivity. You can define DE lists that identify the characteristics of packets to be eligible for discarding, and you can also specify DE groups to identify the DLCI that is affected.

You can specify DE lists based on the protocol or the interface, and on characteristics such as fragmentation of the packet, a specific TCP or User Datagram Protocol (UDP) port, an access list number, or a packet size.

Note: To avoid packet loss, implement unacknowledged application protocols (such as packetized video) carefully. With these protocols, there is a greater potential for buffer overflow.

Multiprotocol Traffic Management Requirements

With multiple protocols being transmitted into a Frame Relay internetwork through a single physical interface, you might find it useful to separate traffic among different DLCIs based on protocol type. To split traffic in this way, you must assign specific protocols to specific DLCIs. This can be done by specifying static mapping on a per virtual interface basis or by defining only specific types of encapsulations for specific virtual interfaces.

Figure: Virtual interfaces assigned specific protocols illustrates the use of virtual interfaces (assigned using subinterface configuration commands) to allocate traffic to specific DLCIs. In this case, traffic of each configured protocol is sent down a specific DLCI and segregated on a per-circuit basis. In addition, each protocol can be assigned a separate CIR and a separate level of buffering by the Frame Relay service provider.

Figure: Virtual interfaces assigned specific protocols

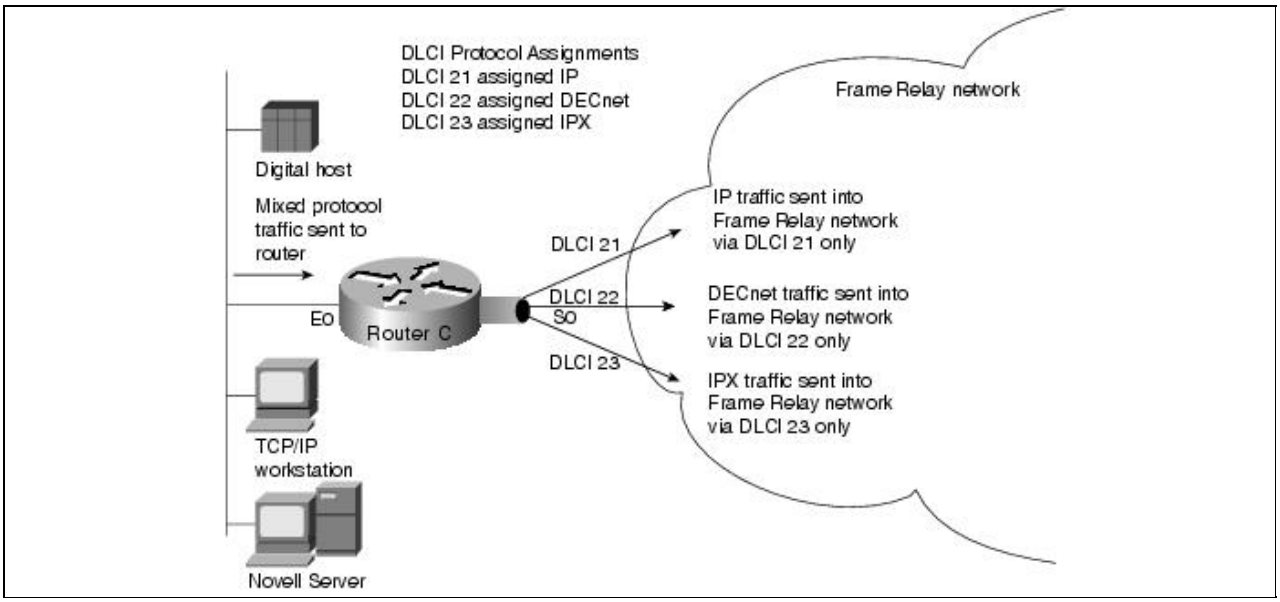


Figure: Virtual interface configuration example provides a listing of the subinterface configuration commands needed to support the configuration illustrated in Figure: Virtual interfaces assigned specific protocols. The command listing in Figure: Virtual interface configuration example illustrates the enabling of the relevant protocols and the assignment of the protocols to the specific subinterfaces and associated Frame Relay DLCIs. Software Release 9.1 and later uses Frame Relay Inverse Address Resolution Protocol (IARP)

to map protocol addresses to Frame Relay DLCIs dynamically. For that reason, Figure: Virtual interface configuration example does not show Frame Relay mappings.

Figure: Virtual interface configuration example

```
interface Ethernet 0
ip address 192.198.78-9 255.255.255.0
ipx network AC
denet cost 4
no mcp enabled
!
interface Serial0
no ip address
encapsulation frame-realy
!
interface Serial0.1 point-to-point
ip address 131.108.3.12 255.255.255.0
frame-relay interface-dlci 21 broadcast
no frame-relay inverse-arp IP 21
no frame-relay inverse-arp NOVELL 21
no frame-relay inverse-arp APPLETALK 21
no frame-relay inverse-arp INS 21
!
interface Serial0.2 point-to-point
no ip address
decnet cost 10
frame-relay interface-dlci 22 broadcast
no frame-relay inverse-arp IP 22
no frame-relay inverse-arp NOVELL 22
no frame-relay inverse-arp APPLETALK 22
no frame-relay inverse-arp INS 22
!
interface Serial0.3 point-to-point
no ip address
ipx network A3
frame-relay interface-dlci 23 broadcast
no frame-relay inverse-arp IP 23
no frame-relay inverse-arp NOVELL 23
no frame-relay inverse-arp APPLETALK 23
no frame-relay inverse-arp INS 23
!
router igrp 109
network 192.198.78.0
!
ip name-server 255.255.255.255
!
snmp-server community
!
line con 0
line aux 0
line vty 0 4
end
```

Subinterface command configuration defining Frame Relay DLCIs and assigning protocols to specific DLCIs

You can use the following commands in Software Release 9.1 to achieve a configuration that is similar to the configuration shown in Figure: Virtual interface configuration example:

You can use the following commands in Software Release 9.1 to achieve a configuration that is similar to the configuration shown in Figure: Virtual interface configuration example:

```
 Version 9.1
interface serial 0
ip address 131.108.3.12 255.255.255.0
decnet cost 10
novell network A3
frame-relay map IP 131.108.3.62 21 broadcast
frame-relay map DECNET 10.3 22 broadcast
frame-relay map NOVELL C09845 23 broadcast
```

Configuring Frame Relay Traffic Shaping

Beginning with Release 11.2, Cisco IOS supports Frame Relay traffic shaping, which provides the following features:

- Rate enforcement on a per-virtual circuit basis-The peak rate for outbound traffic can be set to the CIR or some other user-configurable rate.
- Dynamic traffic throttling on a per-virtual circuit basis-When BECN packets indicate congestion on the network, the outbound traffic rate is automatically stepped down; when congestion eases, the outbound traffic rate is stepped up again. This feature is enabled by default.
- Enhanced queuing support on a per-virtual circuit basis-Either custom queuing or priority queuing can be configured for individual virtual circuits.

By defining separate virtual circuits for different types of traffic and specifying queuing and an outbound traffic rate for each virtual circuit, you can provide guaranteed bandwidth for each type of traffic. By specifying different traffic rates for different virtual circuits over the same time, you can perform virtual time division multiplexing. By throttling outbound traffic from high-speed lines in central offices to low-speed lines in remote locations, you can ease congestion and data loss in the network; enhanced queuing also prevents congestion-caused data loss. Traffic shaping applies to both PVCs and SVCs.

Summary

This article has focused on the implementation of packet-switching services and addresses internetwork design in terms of the packet-switching service topics including hierarchical internetwork design, topology design, broadcast issues, and performance issues.