

Data Link Connection Identifier

(DLCI)

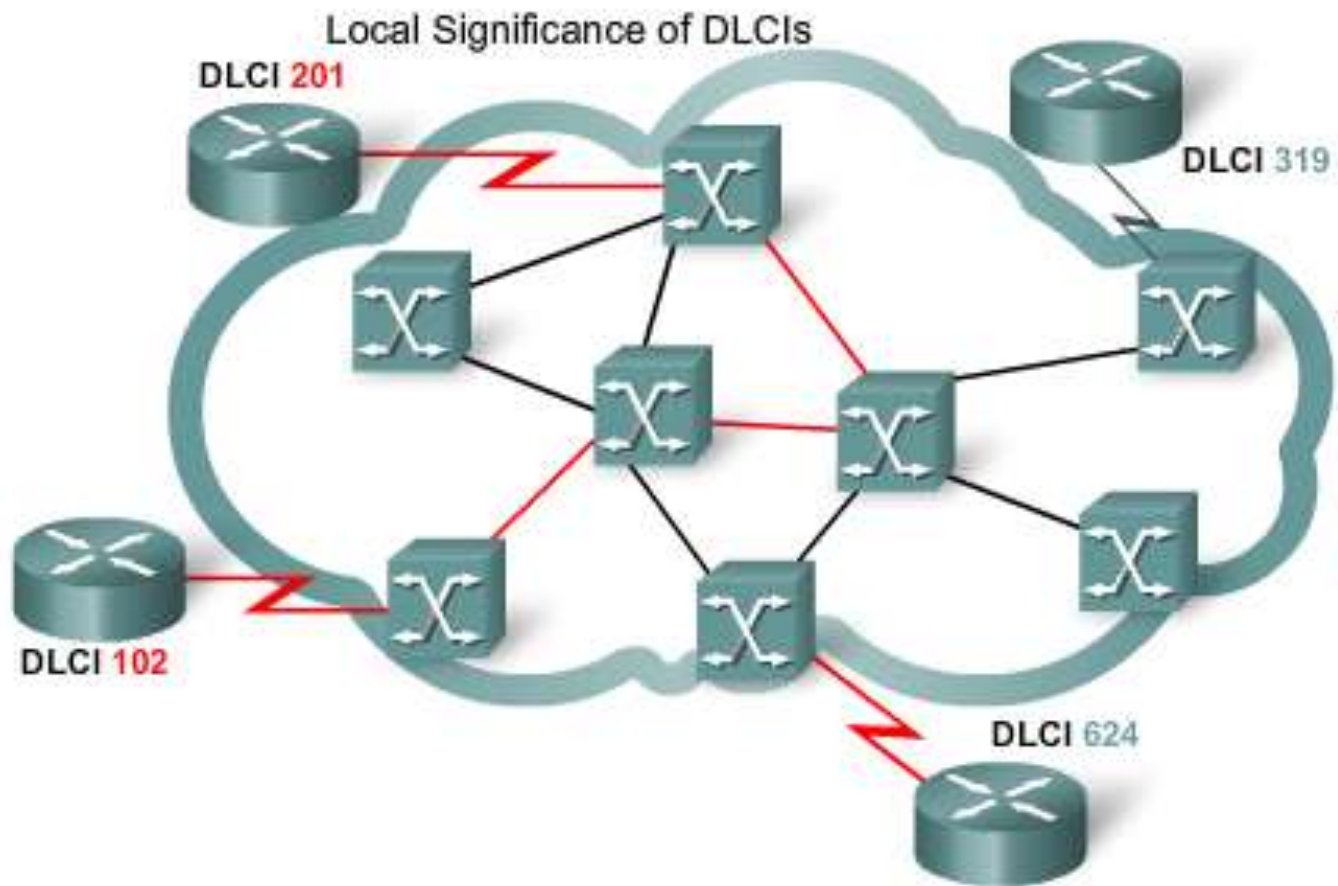
DLCI

- In Frame Relay, the end of each connection has a number to identify it called a Data Link Connection Identifier (DLCI).
- Any station can connect with any other simply by stating the address of that station and DLCI number of the line it needs to use.

- In a later section, you will learn that when Frame Relay is configured, all the data from all the configured DLCIs flows through the same port of the router.

- VCs provide a bidirectional communication path from one device to another. VCs are identified by DLCIs. DLCI values typically are assigned by the Frame Relay service provider (for example, the telephone company). Frame Relay DLCIs have local significance, which means that the values themselves are not unique in the Frame Relay WAN. A DLCI identifies a VC to the equipment at an endpoint. A DLCI has no significance beyond the single link. Two devices connected by a VC may use a different DLCI value to refer to the same connection.

- Locally significant DLCIs have become the primary method of addressing, because the same address can be used in several different locations while still referring to different connections. Local addressing prevents a customer from running out of DLCIs as the network grows.



DLCI values have local significance, which means that they are unique only to the physical channel on which they reside. Therefore, devices at opposite ends of a connection can use the same DLCI values to refer to different virtual circuits.

Identifying VCs

- When the frame moves across the network, Frame Relay labels each VC with a DLCI. The DLCI is stored in the address field of every frame transmitted to tell the network how the frame should be routed. The Frame Relay service provider assigns DLCI numbers.
- Usually, DLCIs 0 to 15 and 1008 to 1023 are reserved for special purposes. Therefore, service providers typically assign DLCIs in the range of 16 to 1007.

- In this example, the frame uses DLCI 102. It leaves the router (R1) using Port 0 and VC 102. At switch A, the frame exits Port 1 using VC 432. This process of VC-port mapping continues through the WAN until the frame reaches its destination at DLCI 201, as shown in the [Video](#). The DLCI is stored in the address field of every frame transmitted.

Explanation:

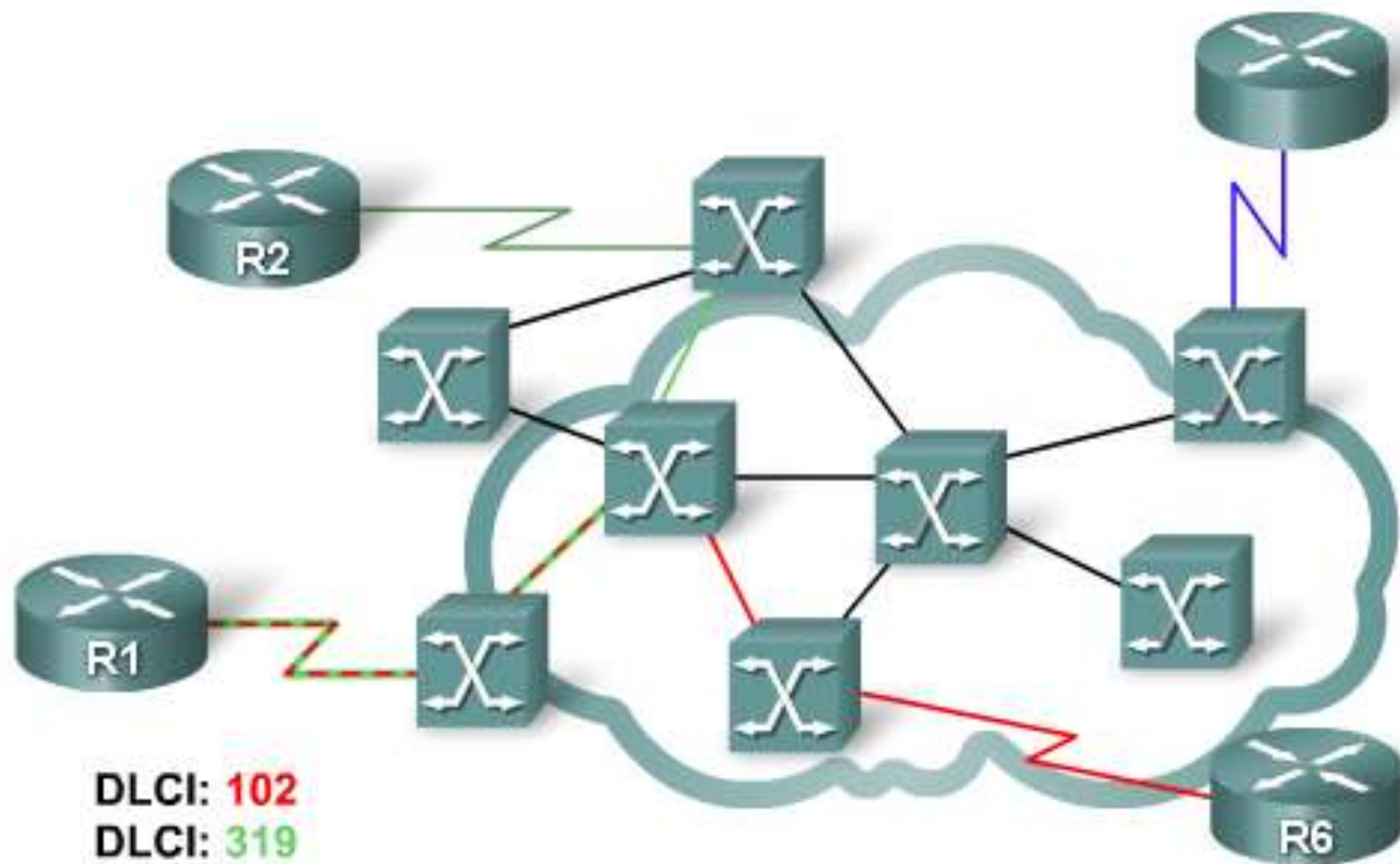
- *DLCI stands for Data Link Connection Identifier. DLCI values are used on Frame Relay interfaces to distinguish between different virtual circuits. DLCIs have local significance because the identifier references the point between the local router and the local Frame Relay switch to which the DLCI is connected.*

Multiple VCs

- Frame Relay is statistically multiplexed, meaning that it transmits only one frame at a time, but that many logical connections can co-exist on a single physical line. The Frame Relay Access Device (FRAD) or router connected to the Frame Relay network may have multiple VCs connecting it to various endpoints. Multiple VCs on a single physical line are distinguished because each VC has its own DLCI. Remember that the DLCI has only local significance and may be different at each end of a VC.

- The figure shows an example of two VCs on a single access line, each with its own DLCI, attaching to a router (R1).

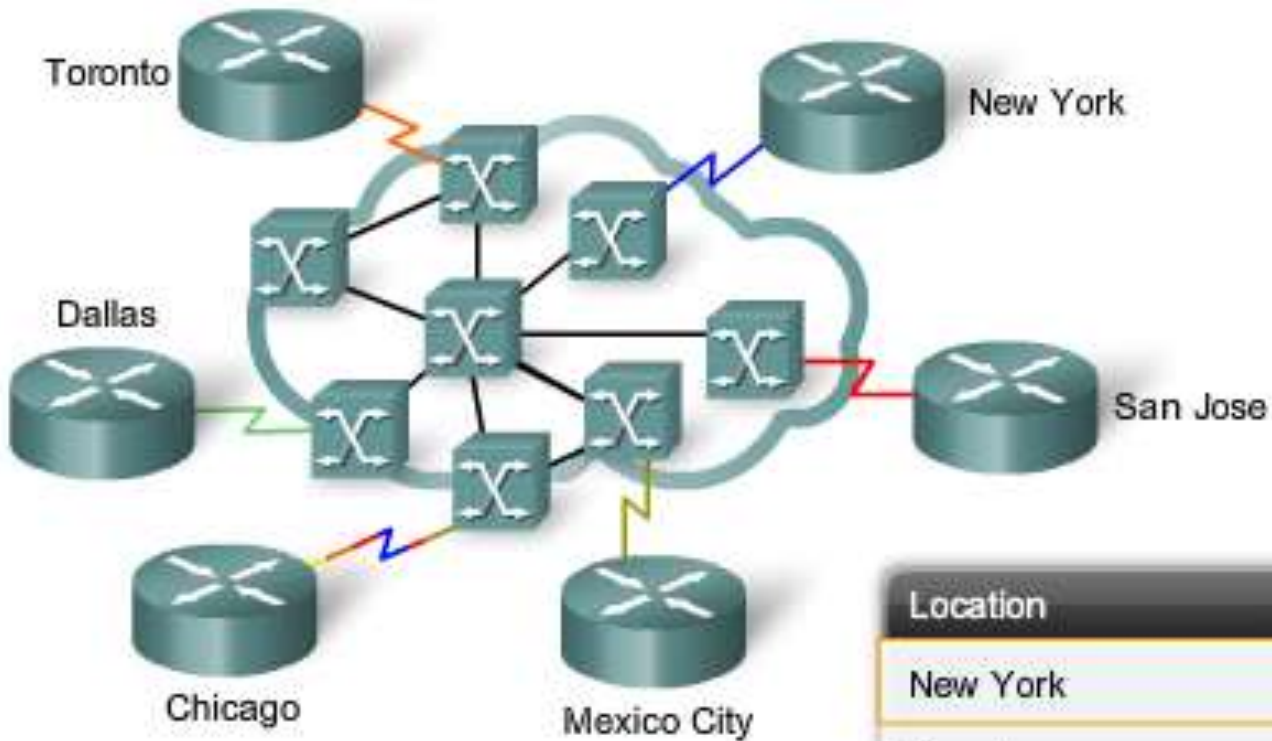
Multiple VCs on A Single Access Line



Multiple VCs on the same access link are distinguishable by the DLCI.

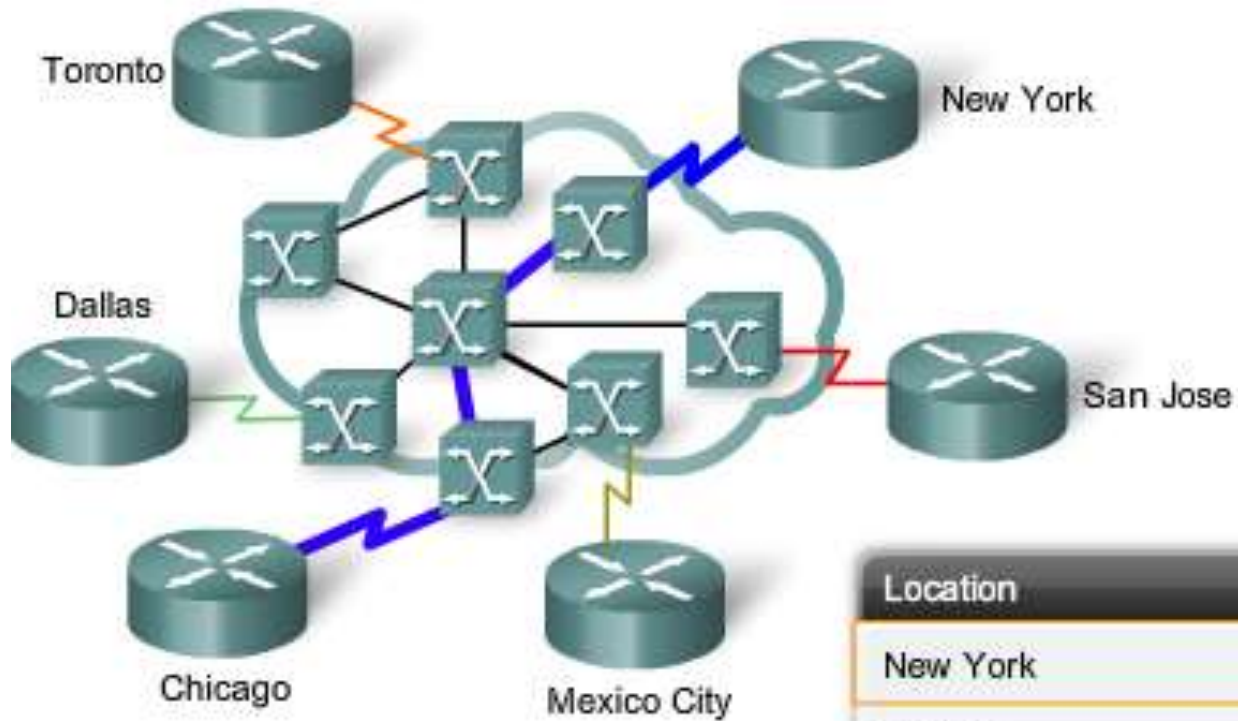
- Span Engineering has five locations, with its headquarters in Chicago. Chicago is connected to the network using five VCs and each VC is given a DLCI. To see Chicago's respective DLCI mappings, click on the location in the table.

Span Engineering DLCIs from Chicago



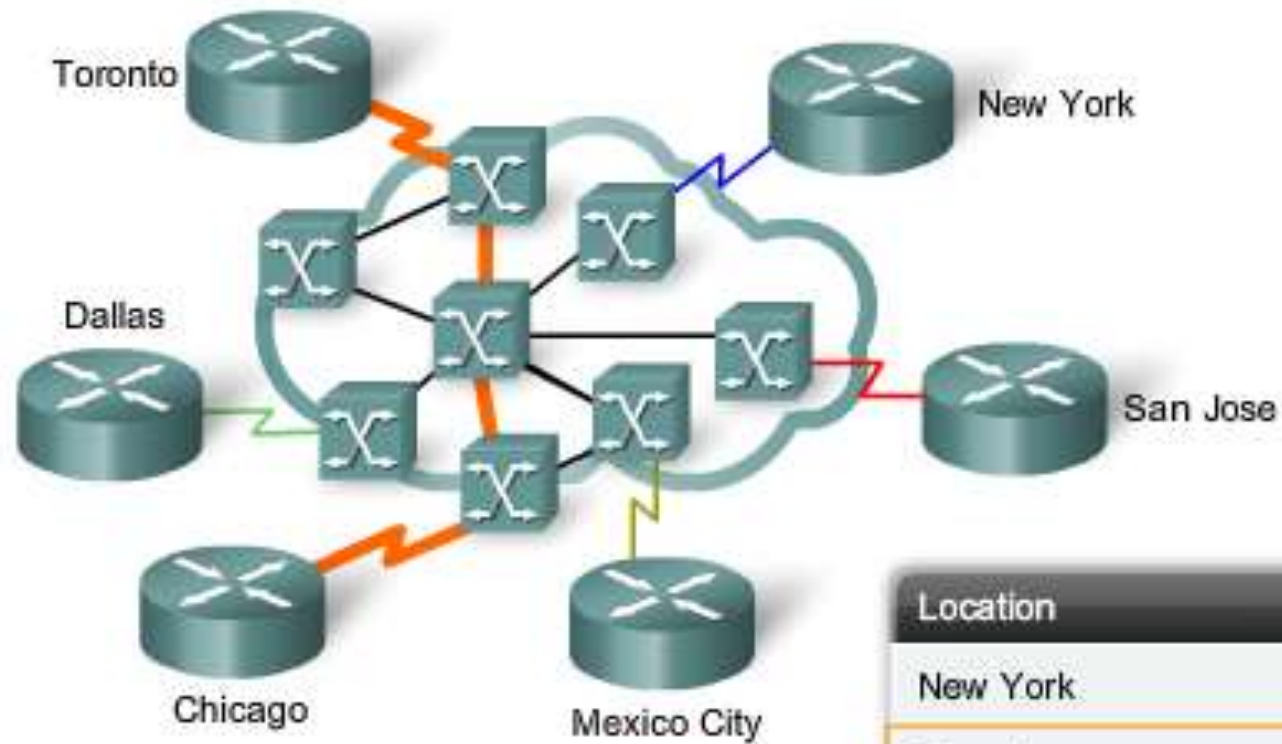
Location	DLCI
New York	17
Toronto	18
Dallas	19
Mexico City	20
San Jose	21

Span Engineering DLCIs from Chicago



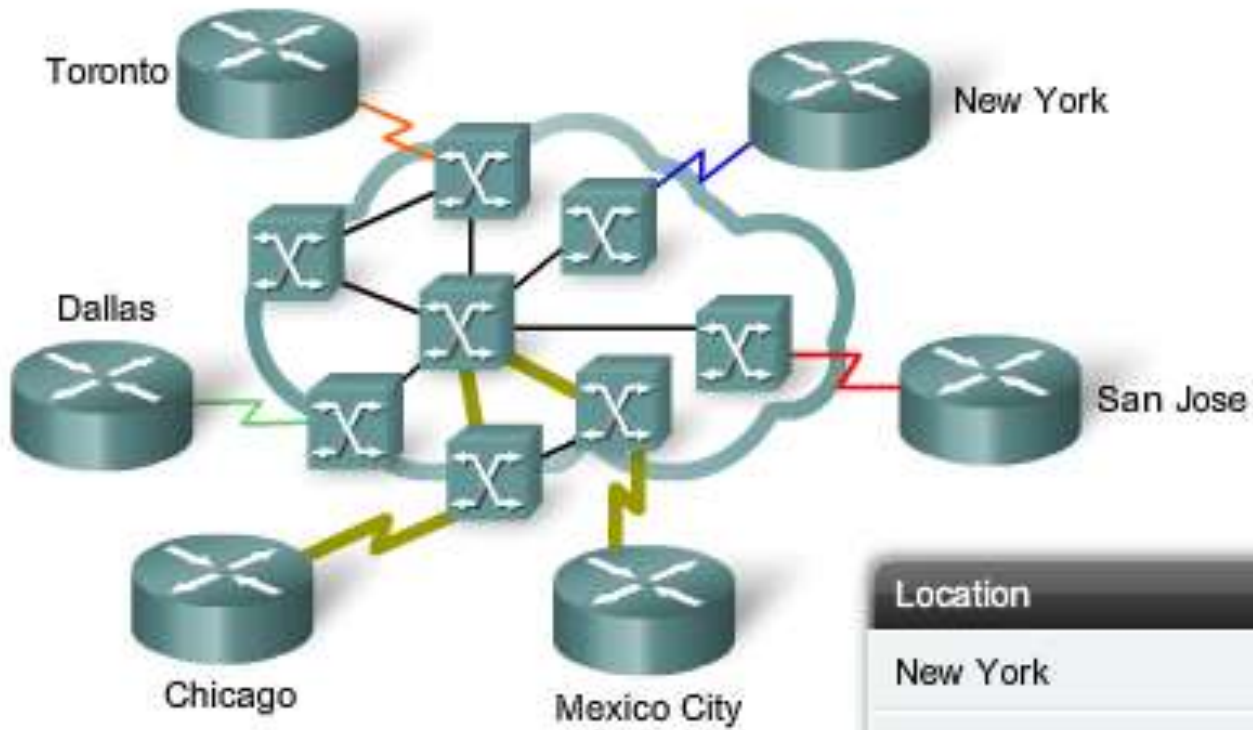
Location	DLCI
New York	17
Toronto	18
Dallas	19
Mexico City	20
San Jose	21

Span Engineering DLCIs from Chicago



Location	DLCI
New York	17
Toronto	18
Dallas	19
Mexico City	20
San Jose	21

Span Engineering DLCIs from Chicago



Location	DLCI
New York	17
Toronto	18
Dallas	19
Mexico City	20
San Jose	21

FR Addressing Mapping

- Before a router is able to transmit data over Frame Relay, it needs to know which local DLCI maps to the Layer 3 address of the remote destination. Routers support all Network layer protocols over Frame Relay, such as IP, IPX, and AppleTalk. This address-to-DLCI mapping can be accomplished either by static or dynamic mapping.

Inverse ARP

- The Inverse Address Resolution Protocol, obtains Layer 3 addresses of other stations from Layer 2 addresses, such as the DLCI in Frame Relay networks. It is primarily used in Frame Relay networks, where Layer 2 addresses of VCs are sometimes obtained from Layer 2 signaling, and the corresponding Layer 3 addresses must be available before these VCs can be used. Whereas ARP resolves Layer 3 addresses to Layer 2 addresses, Inverse ARP does the opposite.

Dynamic Mapping

- Dynamic address mapping relies on Inverse ARP to resolve a next hop network protocol address to a local DLCI value. The Frame Relay router sends out Inverse ARP requests on its PVC to discover the protocol address of the remote device connected to the Frame Relay network. The router uses the responses to populate an address-to-DLCI mapping table on the Frame Relay router or access server. The router builds and maintains this mapping table, which contains all resolved Inverse ARP requests, including both dynamic and static mapping entries.

The figure shows the frame-relay map.

```
R1# show frame-relay map
Serial0/0/0 (up): ip 10.1.1.2 dlci 102(0x66,0x1860), static,
                broadcast,
                CISCO, status defined, active
R1#
```

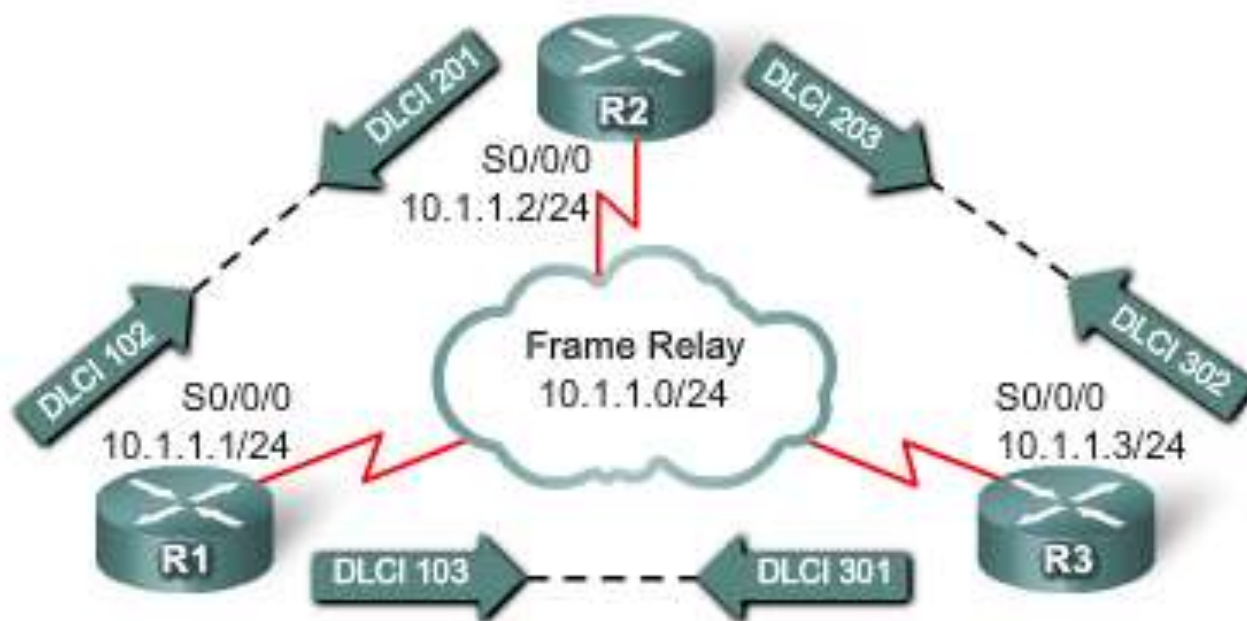
- You can see that the interface is up and that the destination IP address is 10.1.1.2. The DLCI identifies the logical connection being used to reach this interface.

- On Cisco routers, Inverse ARP is enabled by default for all protocols enabled on the physical interface. Inverse ARP packets are not sent out for protocols that are not enabled on the interface.

Static Mapping

- The user can choose to override dynamic Inverse ARP mapping by supplying a manual static mapping for the next hop protocol address to a local DLCI. A static map works similarly to dynamic Inverse ARP by associating a specified next hop protocol address to a local Frame Relay DLCI. You cannot use Inverse ARP and a map statement for the same DLCI and protocol.

Static FR Address Mapping



```
R1(config)# interface s0/0/0
R1(config-if)# ip address 10.1.1.1 255.255.255.0
R1(config-if)# encapsulation frame-relay
R1(config-if)# no frame-relay inverse-arp
R1(config-if)# frame-relay map ip 10.1.1.2 102 broadcast cisco
R1(config-if)# no shut
R1(config-if)#
```

```
+-----+
| 10.1.1.1 10.1.1.2 10.1.1.3 |
+-----+
```


- An example of using static address mapping is a situation in which the router at the other side of the Frame Relay network does not support dynamic Inverse ARP for a specific network protocol. To provide accessibility, a static mapping is required to complete the remote Network layer address to local DLCI resolution.

- Another example is on a hub-and-spoke Frame Relay network. Use static address mapping on the spoke routers to provide spoke-to-spoke reachability. Because the spoke routers do not have direct connectivity with each other, dynamic Inverse ARP would not work between them. Dynamic Inverse ARP relies on the presence of a direct point-to-point connection between two ends. In this case, dynamic Inverse ARP only works between hub and spoke, and the spokes require static mapping to provide reachability to each other.

Local Management Interface (LMI)

- Basically, the LMI is a keepalive mechanism that provides status information about Frame Relay connections between the router (DTE) and the Frame Relay switch (DCE). Every 10 seconds or so, the end device polls the network, either requesting a dumb sequenced response or channel status information. If the network does not respond with the requested information, the user device may consider the connection to be down. When the network responds with a FULL STATUS response, it includes status information about DLCIs that are allocated to that line. The end device can use this information to determine whether the logical connections are able to pass data.

- It is easy to confuse the LMI and encapsulation. The LMI is a definition of the messages used between the DTE (R1) and the DCE (the Frame Relay switch owned by the service provider). Encapsulation defines the headers used by a DTE to communicate information to the DTE at the other end of a VC. The switch and its connected router care about using the same LMI. The switch does not care about the encapsulation. The endpoint routers (DTEs) do care about the encapsulation.

Using LMI and Inverse ARP to Map Addresses

- 1- LMI status messages combined with Inverse ARP messages allow a router to associate Network layer and Data Link layer addresses.

- In this example, when R1 connects to the Frame Relay network, it sends an LMI status inquiry message to the network. The network replies with an LMI status message containing details of every VC configured on the access link.

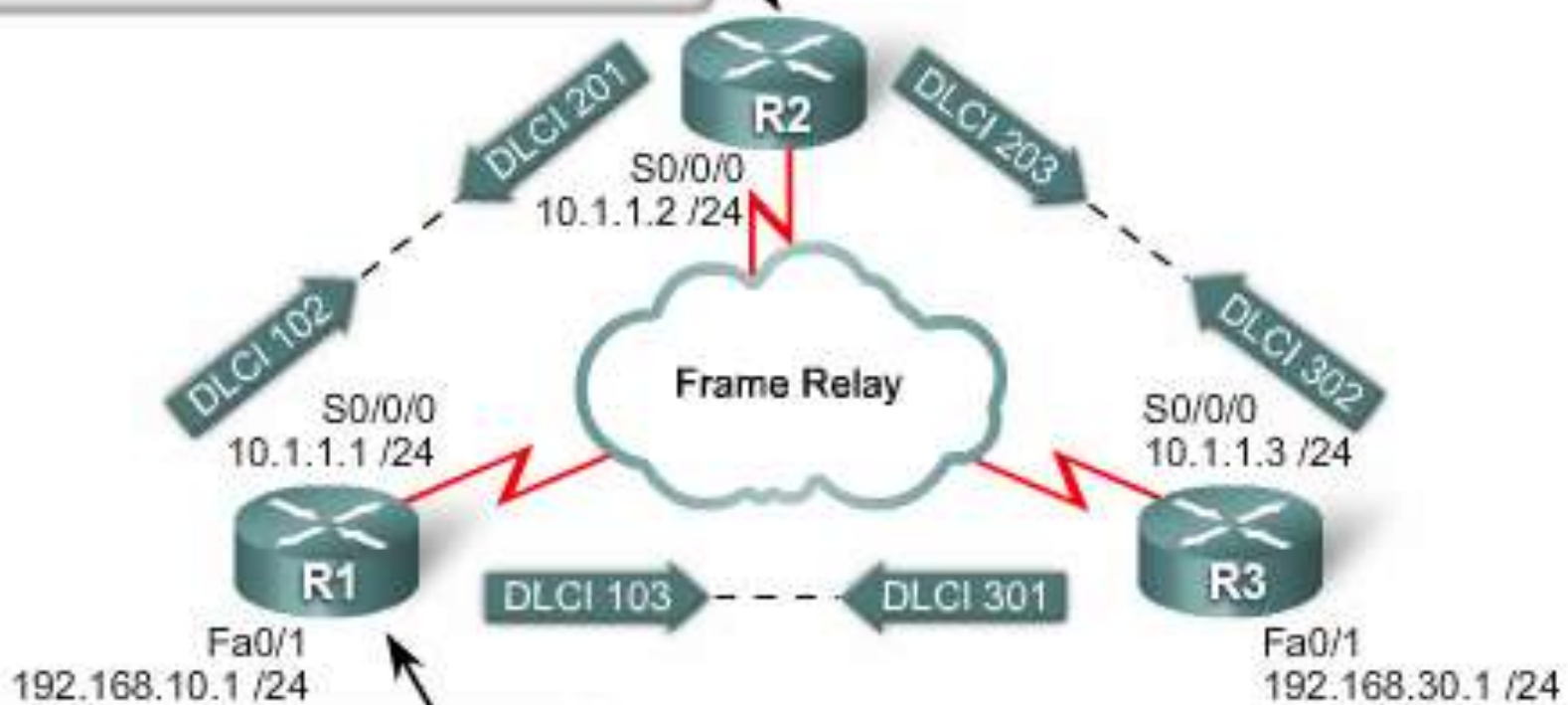
2- If the router needs to map the VCs to Network layer addresses, it sends an Inverse ARP message on each VC. The Inverse ARP message includes the Network layer address of the router, so the remote DTE, or router, can also perform the mapping. The Inverse ARP reply allows the router to make the necessary mapping entries in its address-to-DLCI map table. If several Network layer protocols are supported on the link, Inverse ARP messages are sent for each one.

Configuring Basic Frame Relay

- This first figure, displays how Frame Relay has been configured on the serial interfaces. This involves assigning an IP address, setting the encapsulation type, and allocating bandwidth. The figure shows routers at each end of the Frame Relay link with the configuration scripts for routers R1 and R2.

Configuring Frame Relay

```
interface s0/0/0  
ip address 10.1.1.2 255.255.255.0  
encapsulation frame-relay  
bandwidth 64
```



```
interface s0/0/0  
ip address 10.1.1.1 255.255.255.0  
encapsulation frame-relay  
bandwidth 64
```

Step 1. Setting the IP Address on the Interface

- On a Cisco router, Frame Relay is most commonly supported on synchronous serial interfaces. Use the **ip address** command to set the IP address of the interface. You can see that R1 has been assigned 10.1.1.1/24, and R2 has been assigned IP address 10.1.1.2/24.

Step 2. Configuring Encapsulation

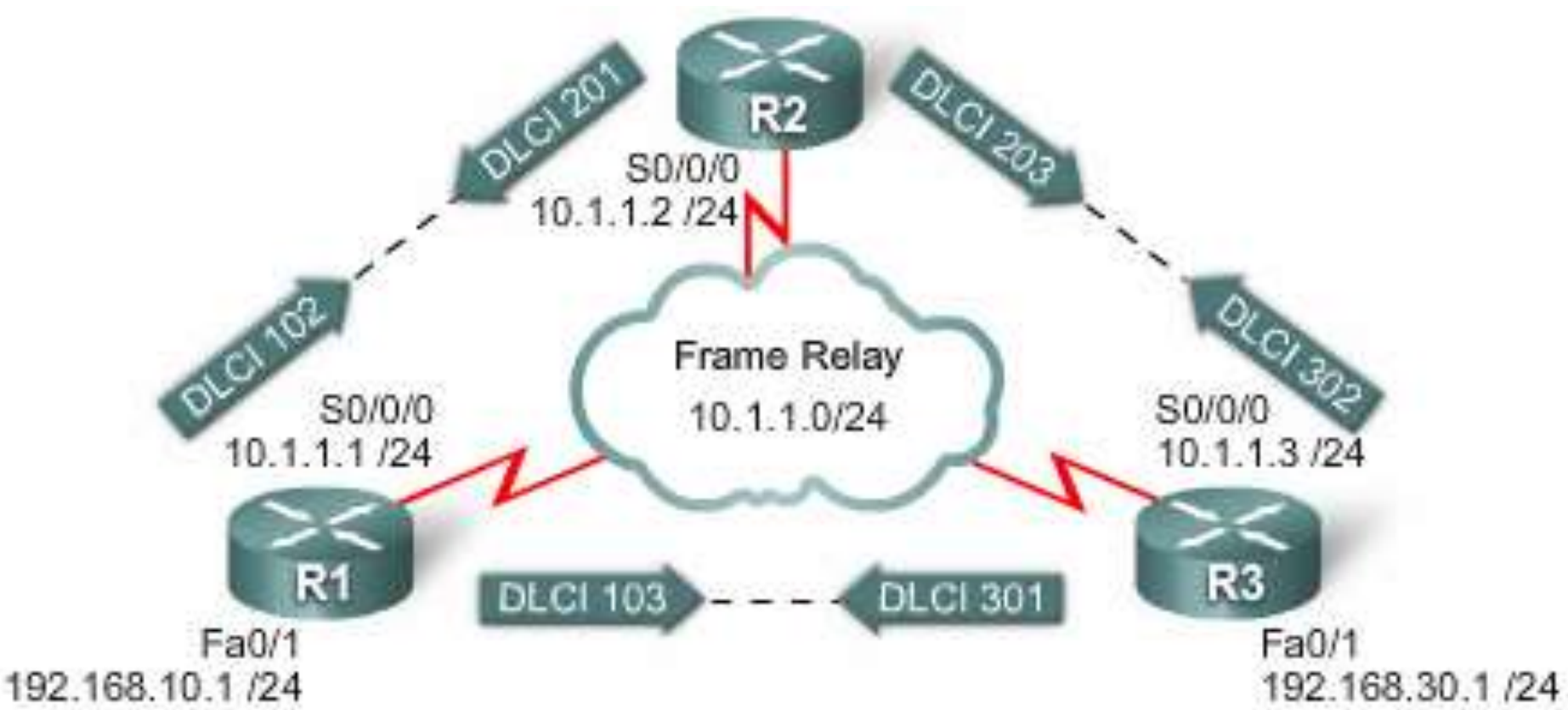
- The **encapsulation frame-relay** interface configuration command enables Frame Relay encapsulation and allows Frame Relay processing on the supported interface.

Step 3. Setting the Bandwidth

- Use the *bandwidth* command to set the bandwidth of the serial interface. Specify bandwidth in kb/s. This command notifies the routing protocol that bandwidth is statically configured on the link. The EIGRP and OSPF routing protocols use the bandwidth value to calculate and determine the metric of the link.

Verifying Configuration

- *show interfaces serial* command verifies the configuration.



Configuring a Static Frame Relay Map

- Cisco routers support all Network layer protocols over Frame Relay, such as IP, IPX, and AppleTalk, and the address-to-DLCI mapping can be accomplished either by dynamic or static address mapping.

- Dynamic mapping is performed by the Inverse ARP feature. Because Inverse ARP is enabled by default, no additional command is required to configure dynamic mapping on an interface.

- Static mapping is manually configured on a router. Establishing static mapping depends on your network needs. To map between a next hop protocol address and a DLCI destination address, use the **frame-relay** **map** protocol protocol-address dlci command.

Configuration for R1

```
interface s0/0/1
ip address 10.1.1.1 255.255.255.252
encapsulation frame-relay
bandwidth 64
frame-relay map ip 10.1.1.2 102 broadcast
```

- To verify the Frame Relay mapping, use the ***show frame-relay map*** command.

```
R1#show frame-relay map
Serial0/0/1 (up): ip 10.1.1.2 dlci 102(0x66,0x1860), static,
broadcast,
CISCO, status defined, active
```

```
R2#show frame-relay map
Serial0/0/1 (up): ip 10.1.1.1 dlci 201(0xC9,0x3090), static,
broadcast,
CISCO, status defined, active
```