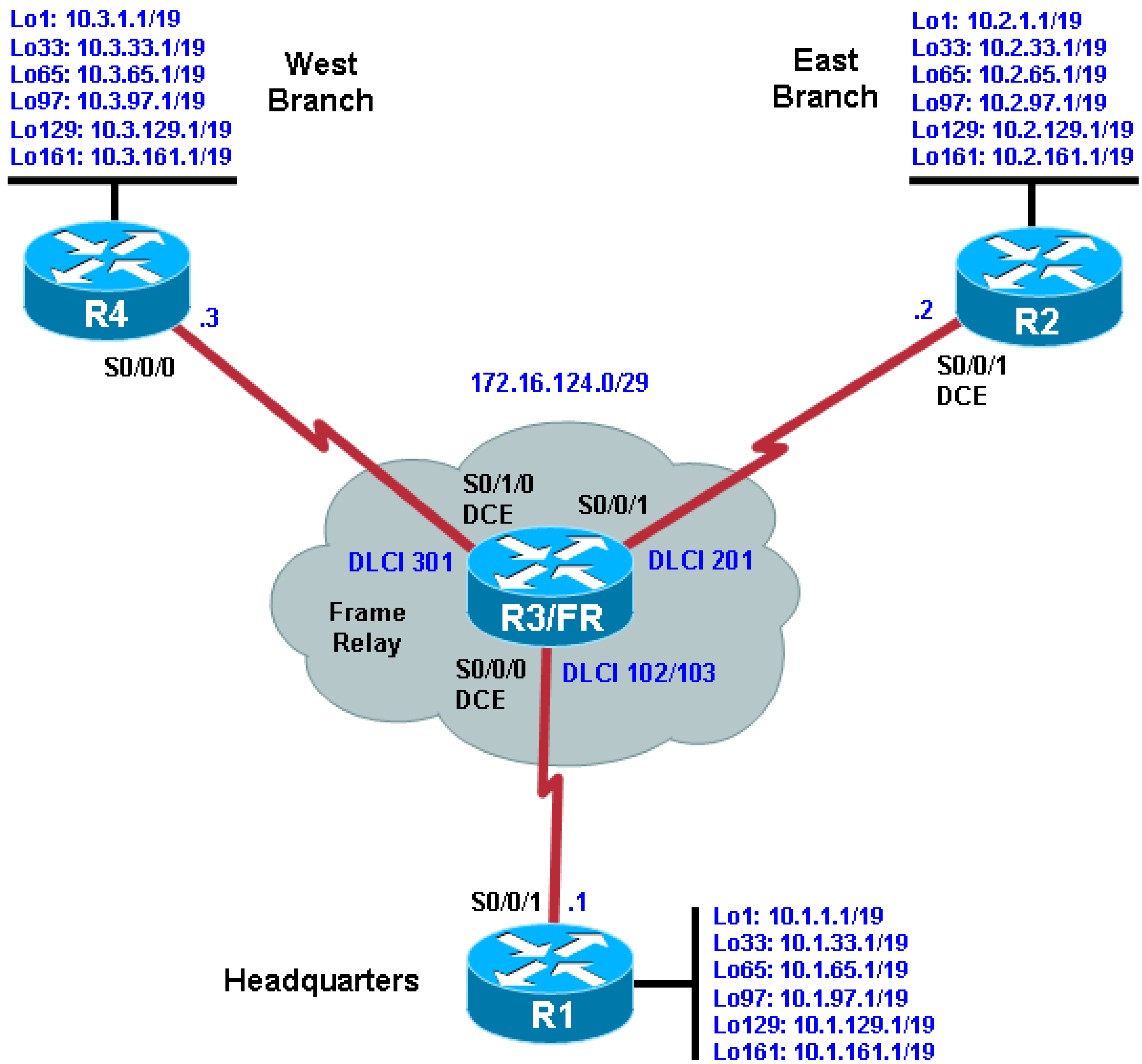
Chapter 2 Lab 2-4, EIGRP Frame Relay Hub-and-Spoke: Router Used as a Frame Relay Switch

Topology



Objectives

- Review a basic configuration of EIGRP on a serial interface.
- Configure EIGRP over Frame Relay hub-and-spoke.
- Configure a router as a Frame Relay switch.
- Configure the **ip bandwidth-percent** command.
- Disable split horizon.

- Use EIGRP in non-broadcast mode.
- Enable EIGRP manual summarization in topologies with discontiguous major networks.

Background

You are responsible for configuring and testing the new network that connects your company's headquarters (HQ) and EAST and WEST branches. The three locations are connected over hub-and-spoke Frame Relay, using the company headquarters as the hub. In this lab, you model each branch office's network with multiple loopback interfaces on each router and configure EIGRP to allow full connectivity between all departments.

To simulate the Frame Relay WAN connections, use a router with three serial ports to act as a Frame Relay switch. The configuration of the router as a Frame Relay switch is described in Step 2.

When accessing a Frame Relay service, a modem or a CSU/DSU is used at the customer premises to connect the router to the local loop and over the local loop to the Frame Relay switch. The modem or CSU/DSU provides the Layer 1 DCE (clocking) functions to the router. However, in this lab, R2 interface serial 0/0/1 is configured as the DCE for compatibility with other labs. If you are uncertain which side of the connection is the DCE, use the **show controllers serial** *interface-number* command:

```
FRS# show controllers serial0/0/0
Interface Serial0/0/0
Hardware is GT96K
DCE V.35, clock rate 64000
```

Note: In this lab, Router R3 acts as the Frame Relay switch and requires two serial interface cards. If you are using an Adtran as a Frame Relay switch, see Appendix A for the Adtran configuration. When using the Adtran, the clock (DCE) is provided for each serial link.

Note: This lab uses Cisco 1841 routers with Cisco IOS Release 12.4(24)T1 and the Advanced IP Services image c1841-advipservicesk9-mz.124-24.T1.bin. You can use other routers (such as 2801 or 2811) and Cisco IOS Software versions if they have comparable capabilities and features. Depending on the router and Cisco IOS Software version, the commands available and output produced might vary from what is shown in this lab.

Required Resources

- 3 routers (Cisco 1841 with Cisco IOS Release 12.4(24)T1 Advanced IP Services or comparable)
- 1 router acting as a Frame Relay switch (Cisco 1841 with Cisco IOS Release 12.4(24)T1 Advanced IP Services or comparable)
- Serial and console cables

Step 1: Configure loopback addressing.

Using the addressing scheme in the diagram, apply IP addresses to the loopback interfaces on HQ, EAST, and WEST. You can paste the following configurations into your routers to begin.

Router R1 (HQ)

```
hostname HQ
!
interface Loopback1
 ip address 10.1.1.1 255.255.224.0
interface Loopback33
 ip address 10.1.33.1 255.255.224.0
interface Loopback65
 ip address 10.1.65.1 255.255.224.0
interface Loopback97
```



```
 ip address 10.1.97.1 255.255.224.0
interface Loopback129
 ip address 10.1.129.1 255.255.224.0
interface Loopback161
 ip address 10.1.161.1 255.255.224.0
!
```

Router R2 (EAST)

```
hostname EAST
!
interface Loopback1
 ip address 10.2.1.1 255.255.224.0
interface Loopback33
 ip address 10.2.33.1 255.255.224.0
interface Loopback65
 ip address 10.2.65.1 255.255.224.0
interface Loopback97
 ip address 10.2.97.1 255.255.224.0
interface Loopback129
 ip address 10.2.129.1 255.255.224.0
interface Loopback161
 ip address 10.2.161.1 255.255.224.0
!
end
```

Router R4 (WEST)

```
hostname WEST
!
interface Loopback1
 ip address 10.3.1.1 255.255.224.0
interface Loopback33
 ip address 10.3.33.1 255.255.224.0
interface Loopback65
 ip address 10.3.65.1 255.255.224.0
interface Loopback97
 ip address 10.3.97.1 255.255.224.0
interface Loopback129
 ip address 10.3.129.1 255.255.224.0
interface Loopback161
 ip address 10.3.161.1 255.255.224.0
!
end
```

Step 2: Configure the Frame Relay switch.

Use a fourth Cisco router with three serial interfaces as a Frame Relay switch, and cable the routers according to the diagram. Paste the following configuration into the router that is simulating the Frame Relay switch.

Note: If you are using an Adtran as a Frame Relay switch, see Appendix A for the Adtran configuration and which cables to use between the routers and the Adtran.

Router R3 (FRS)

```
hostname FRS
!
frame-relay switching
!
```

```
interface Serial0/0/0
 description FR to HQ
 no ip address
 encapsulation frame-relay ietf
 clock rate 128000
 frame-relay lmi-type cisco
 frame-relay intf-type dce
 frame-relay route 102 interface Serial0/0/1 201
 frame-relay route 103 interface Serial0/1/0 301
 no shutdown
!
interface Serial0/0/1
 description FR to EAST
 no ip address
 encapsulation frame-relay ietf
 frame-relay lmi-type cisco
 frame-relay intf-type dce
 frame-relay route 201 interface Serial0/0/0 102
 no shutdown
!
interface Serial0/1/0
 description FR to WEST
 no ip address
 encapsulation frame-relay ietf
 clock rate 64000
 frame-relay lmi-type cisco
 frame-relay intf-type dce
 frame-relay route 301 interface Serial0/0/0 103
 no shutdown
!
End
```

Note: You do not need to configure the LMI type as **cisco** because it is the default. In addition, the HQ, EAST, and WEST routers are able to automatically determine the LMI type. However, you could configure the Frame Relay switch with a different LMI type on each of its interfaces to demonstrate support for the existing types (cisco, ansi, q933a) and to show that they interoperate cleanly without requiring any particular configuration on the Frame Relay switch or end routers. Additionally, the **ietf** keyword is meaningful only on Frame Relay end devices, not on Frame Relay switches. It is configured on R3 here for clarity.

Step 3: Configure the Frame Relay endpoints.

You will be configuring HQ to be the Frame Relay hub, with EAST and WEST as the spokes. Check the topology diagram for the data-link connection identifiers (DLCIs) to use in the Frame Relay maps. You will also be turning off Frame Relay Inverse Address Resolution Protocol (InARP) for all interfaces. Inverse ARP allows a Frame Relay network to discover the IP address associated with the virtual circuit. This is sometimes a desirable trait in a production network. However, in the lab, Inverse ARP is turned off because you are using static maps.

a. Assign the Frame Relay interface of each router an IP address in the Frame Relay subnet 172.16.124.0 /29, as indicated in the topology diagram.

b. Enable Frame Relay encapsulation, disable Frame Relay Inverse ARP, and map the other IPs in the subnet to DLCIs using the **frame-relay map ip** *address dlci* **broadcast** command. The **broadcast** keyword is important because without it, no broadcast or multicast packets including EIGRP messages are sent through the Frame Relay cloud. Bring up the interfaces with the **no shutdown** command.

Note: It is unnecessary and actually undesirable to specify the mappings from EAST to WEST and from WEST to EAST with the **broadcast** option. Specifying this option duplicates all broadcasts and multicasts sent from EAST or WEST routers. Because of the hub-and-spoke topology, HQ will also receive all broadcasts and multicasts twice. Specify the **broadcast** option if a direct PVC is going toward the mapped IP, as is the case between EAST and HQ and between WEST and HQ.

Note: It is good practice to specify the IETF encapsulation format, which is an open format. Using the IETF frame format helps ensure interoperability between different vendors.

```
HQ# conf t
HQ(config)# interface serial 0/0/1
HQ(config-if)# ip address 172.16.124.1 255.255.255.248
HQ(config-if)# encapsulation frame-relay ietf
HQ(config-if)# no frame-relay inverse-arp
HQ(config-if)# frame-relay map ip 172.16.124.2 102 broadcast
HQ(config-if)# frame-relay map ip 172.16.124.3 103 broadcast
HQ(config-if)# no shutdown

EAST# conf t
EAST(config)# interface serial 0/0/1
EAST(config-if)# ip address 172.16.124.2 255.255.255.248
EAST(config-if)# clock rate 64000
EAST(config-if)# encapsulation frame-relay ietf
EAST(config-if)# no frame-relay inverse-arp
EAST(config-if)# frame-relay map ip 172.16.124.1 201 broadcast
EAST(config-if)# frame-relay map ip 172.16.124.3 201
EAST(config-if)# no shutdown

WEST# conf t
WEST(config)# interface serial 0/0/0
WEST(config-if)# ip address 172.16.124.3 255.255.255.248
WEST(config-if)# encapsulation frame-relay ietf
WEST(config-if)# no frame-relay inverse-arp
WEST(config-if)# frame-relay map ip 172.16.124.1 301 broadcast
WEST(config-if)# frame-relay map ip 172.16.124.2 301
WEST(config-if)# no shutdown
```

You will configure the bandwidth for the serial links in Step 4.

c. Verify that you have connectivity across the Frame Relay network by pinging the remote routers from each of the Frame Relay endpoints.

```
HQ# ping 172.16.124.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.124.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

The only interface that the Frame Relay interface is unable to communicate with is itself. This is not a significant problem in Frame Relay networks. To allow the Frame Relay interface to ping itself, you can map the local IP address to be forwarded out a PVC. The remote router at the other end of the PVC can then forward the IP address back based on its Frame Relay map statements. This solution is so that the Tcl scripts used for testing return successful echo replies under all circumstances. You do not need the **broadcast** keyword on this DLCI, because it is not important to forward broadcast and multicast packets (such as EIGRP hellos) to your own interface.

d. Configure the local mappings.

```
HQ(config)# interface serial 0/0/1
```



```
HQ(config-if)# frame-relay map ip 172.16.124.1 102

EAST(config)# interface serial 0/0/1
EAST(config-if)# frame-relay map ip 172.16.124.2 201

WEST(config)# interface serial 0/0/0
WEST(config-if)# frame-relay map ip 172.16.124.3 301
```

HQ now forwards packets destined for its own serial interface (172.16.124.1) to the EAST serial interface (172.16.124.2) and then back. This allows HQ to ping its own serial interface in the lab Frame Relay network. In a production network in which a company is billed based on per-PVC usage, this is not a preferred configuration. However, in this lab network, it helps ensure full ICMP connectivity in the Tcl scripts.

Step 4: Set interface-level bandwidth.

a. On the three routers, set the Frame Relay serial interface bandwidth with the **bandwidth** *bandwidth* command in interface configuration mode. Specify the bandwidth in kilobits per second (kb/s). For HQ, use 128 kb/s. On EAST and WEST, use 64 kb/s.

Recall from Lab 2-1 that, by default, EIGRP limits its bandwidth usage to 50 percent of the value specified by the **bandwidth** parameter. The default bandwidth for a serial interface is 1544 kb/s. This means that each neighbor for which this is an outbound interface has a traffic limit of a fraction of that 50 percent, represented by 1/*N*, where *N* is the number of neighbors out that interface.

```
HQ(config)# interface serial 0/0/1
HQ(config-if)# bandwidth 128

EAST(config)# interface serial 0/0/1
EAST(config-if)# bandwidth 64

WEST(config)# interface serial 0/0/0
WEST(config-if)# bandwidth 64
```

The HQ serial interface divides its total EIGRP bandwidth into fractional amounts according to the number of neighbors out that interface.

How much bandwidth on HQ serial 0/0/1 is reserved for EIGRP traffic to EAST?

b. You can control both the bandwidth and the EIGRP bandwidth percentage on a per-interface basis. On HQ, limit the bandwidth used by EIGRP to 40 percent without changing the **bandwidth** parameter on the interface using the interface-level command **ip bandwidth-percent eigrp** *as_number percent*.

```
HQ(config-if)# ip bandwidth-percent eigrp 1 40
```

Step 5: Configure EIGRP routing.

a. Configure EIGRP AS 1 on HQ, EAST, and WEST.

The network represented in the diagram is a discontiguous network (10.0.0.0/8) configured on all three routers. If you enabled auto-summarization, HQ sends and receives summaries for 10.0.0.0/8 with both EAST and WEST. Auto-summarization causes considerable routing disruption in this network, because HQ does not know which of the two spokes is the correct destination for subnets of 10.0.0.0/8. For this reason, turn off auto-summarization on each router.

b. Add your network statements to EIGRP on all three routers. The two major networks being used are 10.0.0.0 for the loopbacks and 172.16.0.0 for the Frame Relay cloud.

```
HQ(config)# router eigrp 1
HQ(config-router)# network 10.0.0.0
HQ(config-router)# network 172.16.0.0
HQ(config-router)# no auto-summary

EAST(config)# router eigrp 1
EAST(config-router)# network 10.0.0.0
EAST(config-router)# network 172.16.0.0
EAST(config-router)# no auto-summary

WEST(config)# router eigrp 1
WEST(config-router)# network 10.0.0.0
WEST(config-router)# network 172.16.0.0
WEST(config-router)# no auto-summary
```

c. Issue the **show ip eigrp topology** command on EAST.

```
EAST# show ip eigrp topology
IP-EIGRP Topology Table for AS(1)/ID(172.16.124.2)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 10.2.0.0/19, 1 successors, FD is 128256
        via Connected, Loopback1
P 10.1.0.0/19, 1 successors, FD is 40640000
        via 172.16.124.1 (40640000/128256), Serial0/0/1
P 10.2.32.0/19, 1 successors, FD is 128256
        via Connected, Loopback33
P 10.1.32.0/19, 1 successors, FD is 40640000
        via 172.16.124.1 (40640000/128256), Serial0/0/1
P 10.2.64.0/19, 1 successors, FD is 128256
        via Connected, Loopback65
P 10.1.64.0/19, 1 successors, FD is 40640000
        via 172.16.124.1 (40640000/128256), Serial0/0/1
P 10.2.96.0/19, 1 successors, FD is 128256
        via Connected, Loopback97
P 10.1.96.0/19, 1 successors, FD is 40640000
        via 172.16.124.1 (40640000/128256), Serial0/0/1
P 10.2.128.0/19, 1 successors, FD is 128256
        via Connected, Loopback129
P 10.1.128.0/19, 1 successors, FD is 40640000
        via 172.16.124.1 (40640000/128256), Serial0/0/1
P 10.2.160.0/19, 1 successors, FD is 128256
        via Connected, Loopback161
P 10.1.160.0/19, 1 successors, FD is 40640000
        via 172.16.124.1 (40640000/128256), Serial0/0/1
P 172.16.124.0/29, 1 successors, FD is 40512000
        via Connected, Serial0/0/1
EAST#
```


Which networks are missing from the topology database?

What do you think is responsible for this problem?

d. HQ needs the **no ip split-horizon eigrp** *as_number* command on its serial Frame Relay interface. This command disables split horizon for an EIGRP autonomous system. If split horizon is enabled (the default), route advertisements from EAST to HQ do not travel to WEST and vice versa, as shown in the above output.

```
HQ(config)# interface serial 0/0/1
HQ(config-if)# no ip split-horizon eigrp 1
```

e. Verify that you see the correct EIGRP adjacencies with the **show ip eigrp neighbors** command.

```
HQ# show ip eigrp neighbors
IP-EIGRP neighbors for process 1
H   Address             Interface       Hold Uptime   SRTT   RTO  Q  Seq
                                        (sec)         (ms)       Cnt Num
1   172.16.124.2        Se0/0/1          176 00:00:05 1588  5000  0  6
0   172.16.124.3        Se0/0/1          176 00:00:05   23  1140  0  6

EAST# show ip eigrp neighbors
IP-EIGRP neighbors for process 1
H   Address             Interface       Hold Uptime   SRTT   RTO  Q  Seq
                                        (sec)         (ms)       Cnt Num
0   172.16.124.1        Se0/0/1          129 00:00:52   20  2280  0  20

WEST# show ip eigrp neighbors
IP-EIGRP neighbors for process 1
H   Address             Interface       Hold Uptime   SRTT   RTO  Q  Seq
                                        (sec)         (ms)       Cnt Num
0   172.16.124.1        Se0/0/0          176 00:00:55   20  2280  0  13
```

f. Verify that you have IP routes on all three routers for the entire topology with the **show ip route** command.

```
HQ# show ip route
<output omitted>

    172.16.0.0/29 is subnetted, 1 subnets
C      172.16.124.0 is directly connected, Serial0/0/1
    10.0.0.0/19 is subnetted, 18 subnets
D      10.2.0.0 [90/20640000] via 172.16.124.2, 00:04:36, Serial0/0/1
D      10.3.0.0 [90/20640000] via 172.16.124.3, 00:04:20, Serial0/0/1
C      10.1.0.0 is directly connected, Loopback1
D      10.2.32.0 [90/20640000] via 172.16.124.2, 00:04:36, Serial0/0/1
D      10.3.32.0 [90/20640000] via 172.16.124.3, 00:04:20, Serial0/0/1
C      10.1.32.0 is directly connected, Loopback33
D      10.2.64.0 [90/20640000] via 172.16.124.2, 00:04:37, Serial0/0/1
```



```
D       10.3.64.0 [90/20640000] via 172.16.124.3, 00:04:21, Serial0/0/1
C       10.1.64.0 is directly connected, Loopback65
D       10.2.96.0 [90/20640000] via 172.16.124.2, 00:04:37, Serial0/0/1
D       10.3.96.0 [90/20640000] via 172.16.124.3, 00:04:21, Serial0/0/1
C       10.1.96.0 is directly connected, Loopback97
D       10.2.128.0 [90/20640000] via 172.16.124.2, 00:04:37, Serial0/0/1
D       10.3.128.0 [90/20640000] via 172.16.124.3, 00:04:21, Serial0/0/1
C       10.1.128.0 is directly connected, Loopback129
D       10.2.160.0 [90/20640000] via 172.16.124.2, 00:04:37, Serial0/0/1
D       10.3.160.0 [90/20640000] via 172.16.124.3, 00:04:21, Serial0/0/1
C       10.1.160.0 is directly connected, Loopback161

EAST# show ip route
<output omitted>

    172.16.0.0/29 is subnetted, 1 subnets
C       172.16.124.0 is directly connected, Serial0/0/1
    10.0.0.0/19 is subnetted, 18 subnets
C       10.2.0.0 is directly connected, Loopback1
D       10.3.0.0 [90/41152000] via 172.16.124.1, 00:01:31, Serial0/0/1
D       10.1.0.0 [90/40640000] via 172.16.124.1, 00:07:12, Serial0/0/1
C       10.2.32.0 is directly connected, Loopback33
D       10.3.32.0 [90/41152000] via 172.16.124.1, 00:01:31, Serial0/0/1
D       10.1.32.0 [90/40640000] via 172.16.124.1, 00:07:13, Serial0/0/1
C       10.2.64.0 is directly connected, Loopback65
D       10.3.64.0 [90/41152000] via 172.16.124.1, 00:01:32, Serial0/0/1
D       10.1.64.0 [90/40640000] via 172.16.124.1, 00:07:13, Serial0/0/1
C       10.2.96.0 is directly connected, Loopback97
D       10.3.96.0 [90/41152000] via 172.16.124.1, 00:01:32, Serial0/0/1
D       10.1.96.0 [90/40640000] via 172.16.124.1, 00:07:13, Serial0/0/1
C       10.2.128.0 is directly connected, Loopback129
D       10.3.128.0 [90/41152000] via 172.16.124.1, 00:01:32, Serial0/0/1
D       10.1.128.0 [90/40640000] via 172.16.124.1, 00:07:13, Serial0/0/1
C       10.2.160.0 is directly connected, Loopback161
D       10.3.160.0 [90/41152000] via 172.16.124.1, 00:01:32, Serial0/0/1
D       10.1.160.0 [90/40640000] via 172.16.124.1, 00:07:13, Serial0/0/1

WEST# show ip route
<output omitted>

    172.16.0.0/29 is subnetted, 1 subnets
C       172.16.124.0 is directly connected, Serial0/0/0
    10.0.0.0/19 is subnetted, 18 subnets
D       10.2.0.0 [90/41152000] via 172.16.124.1, 00:02:00, Serial0/0/0
C       10.3.0.0 is directly connected, Loopback1
D       10.1.0.0 [90/40640000] via 172.16.124.1, 00:07:41, Serial0/0/0
D       10.2.32.0 [90/41152000] via 172.16.124.1, 00:02:00, Serial0/0/0
C       10.3.32.0 is directly connected, Loopback33
D       10.1.32.0 [90/40640000] via 172.16.124.1, 00:07:43, Serial0/0/0
D       10.2.64.0 [90/41152000] via 172.16.124.1, 00:02:01, Serial0/0/0
C       10.3.64.0 is directly connected, Loopback65
D       10.1.64.0 [90/40640000] via 172.16.124.1, 00:07:43, Serial0/0/0
D       10.2.96.0 [90/41152000] via 172.16.124.1, 00:02:01, Serial0/0/0
C       10.3.96.0 is directly connected, Loopback97
D       10.1.96.0 [90/40640000] via 172.16.124.1, 00:07:43, Serial0/0/0
D       10.2.128.0 [90/41152000] via 172.16.124.1, 00:02:01, Serial0/0/0
C       10.3.128.0 is directly connected, Loopback129
D       10.1.128.0 [90/40640000] via 172.16.124.1, 00:07:43, Serial0/0/0
```



```
D       10.2.160.0 [90/41152000] via 172.16.124.1, 00:02:01, Serial0/0/0
C       10.3.160.0 is directly connected, Loopback161
D       10.1.160.0 [90/40640000] via 172.16.124.1, 00:07:43, Serial0/0/0
```

g. Run the following Tcl script on all routers to verify full connectivity.

```
HQ# tclsh

foreach address {
10.1.1.1
10.1.33.1
10.1.65.1
10.1.97.1
10.1.129.1
10.1.161.1
172.16.124.1
10.2.1.1
10.2.33.1
10.2.65.1
10.2.97.1
10.2.129.1
10.2.161.1
172.16.124.2
10.3.1.1
10.3.33.1
10.3.65.1
10.3.97.1
10.3.129.1
10.3.161.1
172.16.124.3
} { ping $address }
```

You should get ICMP echo replies for every address pinged.

Step 6: Configure non-broadcast EIGRP mode.

Currently, you are using EIGRP in its default mode, which multicasts packets to the link-local address 224.0.0.10. However, not all Frame Relay configurations support multicast. EIGRP supports unicasts to remote destinations using non-broadcast mode on a per-interface basis. This mode is analogous to configuring RIPv2 with a passive interface and statically configuring neighbors out that interface.

a. Implement packet unicasts to neighbors.

```
HQ(config)# router eigrp 1
HQ(config-router)# neighbor 172.16.124.2 serial 0/0/1
HQ(config-router)# neighbor 172.16.124.3 serial 0/0/1

EAST(config)# router eigrp 1
EAST(config-router)# neighbor 172.16.124.1 serial 0/0/1

WEST(config)# router eigrp 1
WEST(config-router)# neighbor 172.16.124.1 serial 0/0/0
```

b. HQ now has two neighbor statements, and the other two routers have one. After you configure neighbor statements for an interface, EIGRP automatically stops multicasting packets out that interface and starts unicasting packets instead. Use the **show ip eigrp neighbors** command to verify the changes.

```
HQ# show ip eigrp neighbors
IP-EIGRP neighbors for process 1
H   Address            Interface       Hold Uptime   SRTT   RTO  Q  Seq
```



```
                                           (sec)         (ms)       Cnt Num
1   172.16.124.2       Se0/0/1              153 00:00:28   65   390  0  9
0   172.16.124.3       Se0/0/1              158 00:00:28 1295  5000  0  9

EAST# show ip eigrp neighbors
IP-EIGRP neighbors for process 1
H   Address            Interface       Hold Uptime    SRTT   RTO  Q  Seq
                                       (sec)          (ms)       Cnt Num
0   172.16.124.1       Se0/0/1          146 00:02:19   93   558  0  15

WEST# show ip eigrp neighbors
IP-EIGRP neighbors for process 1
H   Address            Interface       Hold Uptime    SRTT   RTO  Q  Seq
                                       (sec)          (ms)       Cnt Num
0   172.16.124.1       Se0/0/0          160 00:03:00   59   354  0  15
```

Step 7: Implement EIGRP manual summarization.

a. Implement EIGRP manual summarization on each router. Each router should advertise only one network summarizing all of its loopbacks. Using the commands you learned in EIGRP Lab 2-3, configure the summary address on the serial interfaces.

What is the length of the network mask that is used to summarize all the loopbacks on each router?

b. Look at the simplified EIGRP topology table on each router using the **show ip eigrp topology** command.

```
HQ# show ip eigrp topology
IP-EIGRP Topology Table for AS(1)/ID(10.1.161.1)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 10.2.0.0/16, 1 successors, FD is 20640000
        via 172.16.124.2 (20640000/128256), Serial0/0/1
P 10.3.0.0/16, 1 successors, FD is 20640000
        via 172.16.124.3 (20640000/128256), Serial0/0/1
P 10.1.0.0/16, 1 successors, FD is 128256
        via Summary (128256/0), Null0
P 10.1.0.0/19, 1 successors, FD is 128256
        via Connected, Loopback1
P 10.1.32.0/19, 1 successors, FD is 128256
        via Connected, Loopback33
P 10.1.64.0/19, 1 successors, FD is 128256
        via Connected, Loopback65
P 10.1.96.0/19, 1 successors, FD is 128256
        via Connected, Loopback97
P 10.1.128.0/19, 1 successors, FD is 128256
        via Connected, Loopback129
P 10.1.160.0/19, 1 successors, FD is 128256
        via Connected, Loopback161

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 172.16.124.0/29, 1 successors, FD is 20512000
```



```
        via Connected, Serial0/0/1

EAST# show ip eigrp topology
IP-EIGRP Topology Table for AS(1)/ID(10.2.161.1)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 10.2.0.0/16, 1 successors, FD is 128256
        via Summary (128256/0), Null0
P 10.2.0.0/19, 1 successors, FD is 128256
        via Connected, Loopback1
P 10.3.0.0/16, 1 successors, FD is 41152000
        via 172.16.124.1 (41152000/20640000), Serial0/0/1
P 10.1.0.0/16, 1 successors, FD is 40640000
        via 172.16.124.1 (40640000/128256), Serial0/0/1
P 10.2.32.0/19, 1 successors, FD is 128256
        via Connected, Loopback33
P 10.2.64.0/19, 1 successors, FD is 128256
        via Connected, Loopback65
P 10.2.96.0/19, 1 successors, FD is 128256
        via Connected, Loopback97
P 10.2.128.0/19, 1 successors, FD is 128256
        via Connected, Loopback129
P 10.2.160.0/19, 1 successors, FD is 128256
        via Connected, Loopback161
P 172.16.124.0/29, 1 successors, FD is 40512000
        via Connected, Serial0/0/1
        via 172.16.124.1 (41024000/20512000), Serial0/0/1

WEST# show ip eigrp topology
IP-EIGRP Topology Table for AS(1)/ID(172.16.124.3)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 10.2.0.0/16, 1 successors, FD is 41152000
        via 172.16.124.1 (41152000/20640000), Serial0/0/0
P 10.3.0.0/16, 1 successors, FD is 128256
        via Summary (128256/0), Null0
P 10.3.0.0/19, 1 successors, FD is 128256
        via Connected, Loopback1
P 10.1.0.0/16, 1 successors, FD is 40640000
        via 172.16.124.1 (40640000/128256), Serial0/0/0
P 10.3.32.0/19, 1 successors, FD is 128256
        via Connected, Loopback33
P 10.3.64.0/19, 1 successors, FD is 128256
        via Connected, Loopback65
P 10.3.96.0/19, 1 successors, FD is 128256
        via Connected, Loopback97
P 10.3.128.0/19, 1 successors, FD is 128256
        via Connected, Loopback129
P 10.3.160.0/19, 1 successors, FD is 128256
        via Connected, Loopback161
P 172.16.124.0/29, 1 successors, FD is 40512000
        via Connected, Serial0/0/0
        via 172.16.124.1 (41024000/20512000), Serial0/0/0
```


Router Interface Summary Table

Router Interface Summary				
Router Model	Ethernet Interface #1	Ethernet Interface #2	Serial Interface #1	Serial Interface #2
1700	Fast Ethernet 0 (FA0)	Fast Ethernet 1 (FA1)	Serial 0 (S0)	Serial 1 (S1)
1800	Fast Ethernet 0/0 (FA0/0)	Fast Ethernet 0/1 (FA0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2600	Fast Ethernet 0/0 (FA0/0)	Fast Ethernet 0/1 (FA0/1)	Serial 0/0 (S0/0)	Serial 0/1 (S0/1)
2800	Fast Ethernet 0/0 (FA0/0)	Fast Ethernet 0/1 (FA0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)

Note: To find out how the router is configured, look at the interfaces to identify the type of router and how many interfaces the router has. Rather than list all combinations of configurations for each router class, this table includes identifiers for the possible combinations of Ethernet and serial interfaces in the device. The table does not include any other type of interface, even though a specific router might contain one. For example, for an ISDN BRI interface, the string in parenthesis is the legal abbreviation that can be used in Cisco IOS commands to represent the interface.

Appendix A: Adtran Frame Relay Switch Configuration

If an Adtran Atlas 550 is used for the Frame Relay switch, connect the serial cable from each router interface in the topology diagram to the Adtran interface indicated in the table below. The Adtran is preconfigured to simulate a Frame Relay service that provides the following PVCs.

Connected Router	Router Interface	Adtran Interface	Ingress DLCI	Egress DLCI	Egress Router
HQ	S0/0/1 DTE	port 1/1	102	201	East
HQ	S0/0/1 DTE	port 1/1	103	301	West
East	S0/0/1 DTE	port 1/2	201	102	HQ
West	S0/0/0 DTE	port 2/1	301	103	HQ

Frame Relay Switching Configuration

The Adtran Frame Relay switch interfaces all provide the DCE clock. Be sure to use the appropriate cable between each router and the Adtran. All the router interfaces are DTE, and the cable to the Adtran interface should be serial to V.35 DCE. Use the **show controllers** command to verify which cable type is connected to a given router interface.

```
HQ# show controllers s0/0/1
Interface Serial0/0/1
Hardware is GT96K
DTE V.35 TX and RX clocks detected.

<output omitted>
```