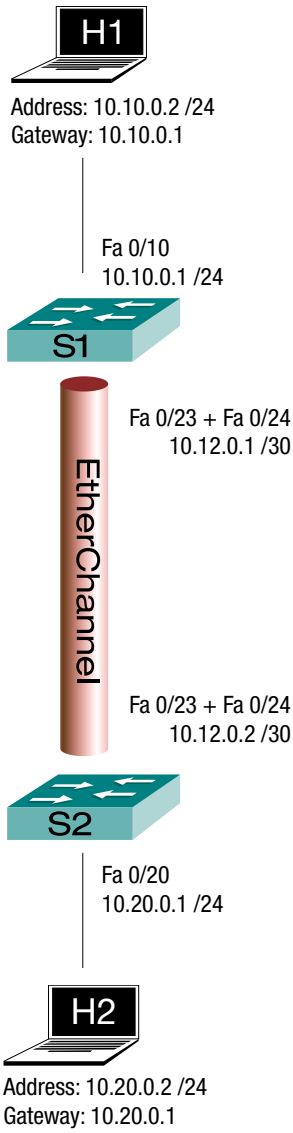
Lab 19c. L3 EtherChannels

GOAL

Configure end-to-end connectivity between H1 and H2 using L3 routed ports on S1 and S2 to connect computers and an L3 EtherChannel between the switches. If necessary, use EIGRP.



Topology

If you already have the topology of the previous labs, 19a, 19(b)1, and 19(b)2, then you can keep the hosts configured as they are and completely erase the switch S1 (including vlan.dat) and reload. You'll also need a second L3 switch (S2).

For this, we're L3 end-to-end with three subnets. The L3 switches really aren't acting like switches at all; they're more like routers.

CONFIGURATION STEPS

Ensure that your hosts are configured according to the diagram.

Configure the switch interfaces that lead to the hosts as routed L3 ports with IP addresses according to the diagram.

Change the switchports connecting S1 and S2 to L3 routed ports.

Manually create an L3 EtherChannel between S1 and S2, with addressing from the diagram.

Enable EIGRP on the switches for end-to-end routing from H1 to H2.

Ping from one host to the other to verify successful completion.

CONFIGURATION WALKTHROUGH

Ensure That Your Hosts Are Configured According to the Diagram

If you happen to be using routers for hosts (see earlier labs in this chapter for the technique), your configuration can be verified in two commands. The term "Default gateway" instead of "Gateway of last resort" [Line 7] tells us that routing is turned off ("no ip routing").

H1

```
1  H1# show ip interface brief
2  Interface                  IP-Address      OK? Method Status                Protocol
3  FastEthernet0/0            10.10.0.2       YES NVRAM  up                    up
4  FastEthernet0/1            unassigned      YES NVRAM  administratively down down
5
6  H1# show ip route
7  Default gateway is 10.10.0.1
8  Host               Gateway           Last Use    Total Uses  Interface
9  ICMP redirect cache is empty
```

Configure Ports Leading to Hosts as L3 Routed Ports

After some boilerplate config on our freshly wiped L3 switches, we can add the following to their respective configs:

S1	S2

`interface FastEthernet0/10`
`  no switchport`
`  ip address 10.10.0.1 255.255.255.0`
	`interface FastEthernet0/20`
	`  no switchport`
	`  ip address 10.20.0.1 255.255.255.0`

If you're configuring the switches from their console ports, you'll see "updown" logging messages when you turn off L2 switching; that's normal. Before leaving each switch, double check its configuration and ability to reach its host.

S1

```
S1# show ip interf brief fa0/10
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/10       10.10.0.1       YES manual up                    up

S1# ping 10.10.0.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.0.2, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/2/4 ms
```

Turn Off L2 Switching on the Ports That Connect S1 and S2 and Create an EtherChannel

Killing switching on the ports before creating an EtherChannel will save a lot of grief later. If you wait and try to modify an L2 EtherChannel to an L3 EtherChannel later, the conversion will kick the switchports out of the EtherChannel without warning, leaving you to figure out what happened and put them back in. You'll quite literally have a PortChannel with no ports.

S1

```
1  S1# show cdp neighbors
2  Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
3                    S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone
4
5  Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
6  S2               Fas 0/24          145           S I      WS-C3550- Fas 0/24
7  S2               Fas 0/23          145           S I      WS-C3550- Fas 0/23
8  H1               Fas 0/10          134           S I      2621XM    Fas 0/0
9  S1# conf t
10 S1(config)# interface range fa0/23 - 24
11 S1(config-if-range)# no switchport
12 S1(config-if-range)# channel-group 1 mode on
```

CDP reminds us which interfaces lead to the other switch [Lines 6-7] so we can change them to L3 routed ports [Line 11] and put them in an EtherChannel group [Line 12]. The channel group number "1" is arbitrary and doesn't need to match at the other end. Once you do this, you may see the other switch complain that the same MAC address is appearing on two of its switchports. That will take care of itself when you configure the other end of the EtherChannel on S2. Do it.

To verify the EtherChannel, we really don't want to know much, just that it's up.

S1

```
S1# show ip interface brief port-channel 1
Interface              IP-Address      OK? Method Status                Protocol
Port-channel1          unassigned      YES unset  up                    up
```

Change the EtherChannel into an L3 Routed Port

Now that we've combined our two ports into one port-channel port (at each end), we can change our port-channel to an L3 routed port by giving it an IP address.

S1

```
S1(config)# interface port-channel 1
S1(config-if)# ip address 10.12.0.1 255.255.255.252
```

S2

```
S2(config)# interf port-channel 2
S2(config-if)# ip address 10.12.0.2 255.255.255.252
S2(config-if)# do ping 10.12.0.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.12.0.1, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/2/4 ms
```

After doing the same on the other end, our pings show that the EtherChannel is working and we can double check that it's fully utilizing the ports we gave it with one of the "show etherchannel" commands.

S2

```
S2# show etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port


Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
2      Po2(RU)          -        Fa0/23(P)   Fa0/24(P)
```

We can see [Line 19] that our port-channel is operating at layer 3 (R) and is in use (U). We can also see that both of the individual interfaces are successfully bundled into the port-channel (P).

Tell Both L3 Switches to Route IPv4 Packets at L3

```
S1(config)# ip routing
S2(config)# ip routing
```

Configure a Routing Protocol for End-to-end Routing

We know from the EIGRP chapter how to elegantly and carefully tailor our network statements to include only the ports we desire. With this lab almost over, we'll instead adopt a quick and dirty approach, including every interface on the diagram. Our only nod to subtlety will be passive interfaces leading to our hosts.

S1

```
S1(config)# router eigrp 10
S1(config-router)# network 10.0.0.0
S1(config-router)# passive-interface default
S1(config-router)# no passive-interface port-channel 1
```

S2

```
S2(config)# router eigrp 10
S2(config-router)# network 10.0.0.0
S2(config-router)# passive-interface default
S2(config-router)# no passive-interface port-channel 2
```

RECAP AND VERIFICATION

The most important thing to remember is that interfaces must be compatible with a port-channel in order to be bundled into it (or remain in it). Change your interfaces to non-switchports before creating the port-channel from them. Then, your EtherChannel will already have non-switchport endpoints ready for their IP addresses.

With everything done, a simple end-to end ping will verify our L3 EtherChannels, our EIGRP route-sharing, and our end-to-end connectivity.

H1

```
H1> ping 10.20.0.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.20.0.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```