Chapter 3:

Implementing Inter-VLAN Routing

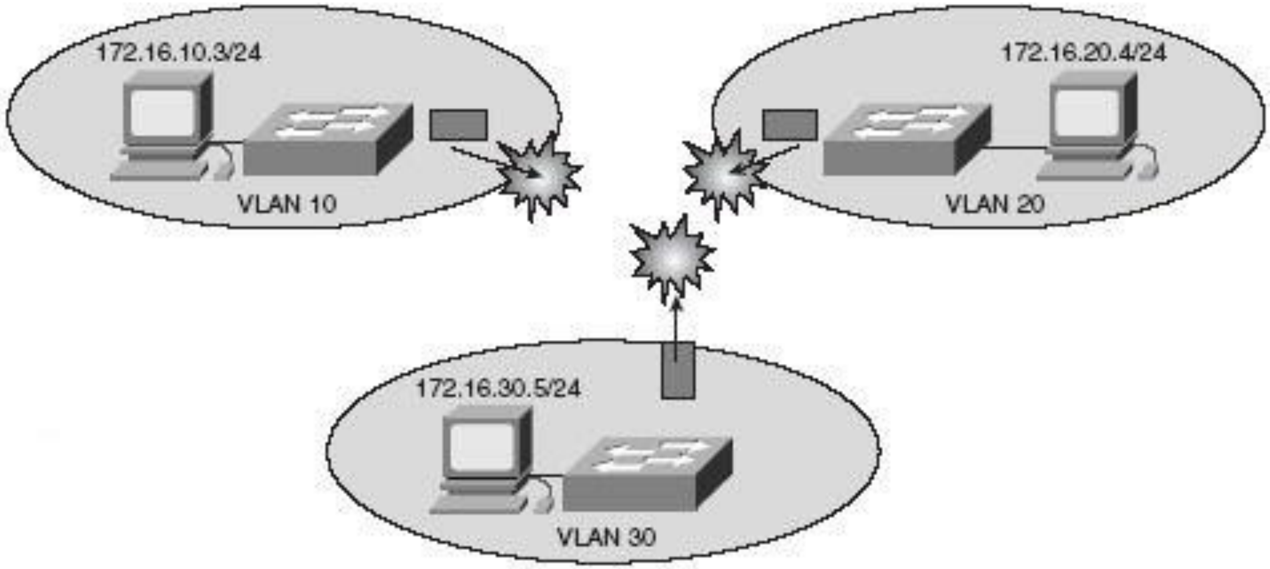
- CCNP-RS SWITCH

Chapter 3 Objectives

- Explain methods of inter-VLAN routing.
- Configure and verify inter-VLAN routing in a Layer 2 topology using multilayer switching.
- Explain DHCP operation and configure DHCP.
- Configure and verify inter-VLAN routing in a Layer 2 topology using CEF-based multilayer switching.

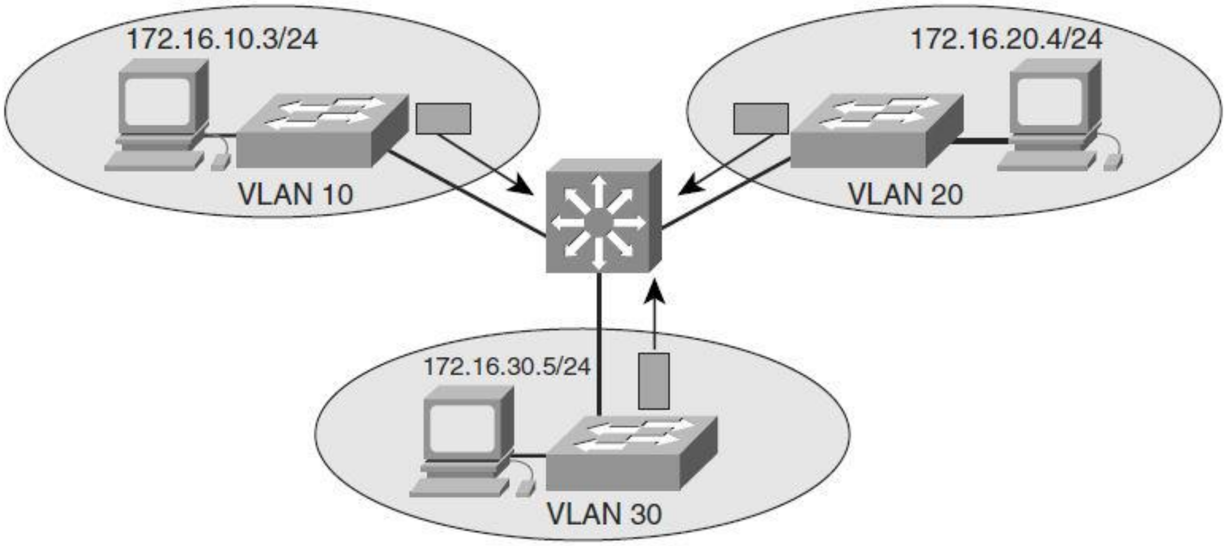
Describing Inter-VLAN Routing

Introduction to Inter-VLAN Routing



- VLANs isolate traffic by design.
- Inter-VLAN router of some sort required.
- Inter-VLAN routing should occur in the distribution layer.
- Multilayer switch is recommended to terminate VLANs.

Inter-VLAN Routing Options



- External router with a separate interface for each VLAN.
- External router trunked to Layer 2 switch (router-on-a-stick).
- Multilayer switch (pictured).

Catalyst Switch Layer 3 Interfaces

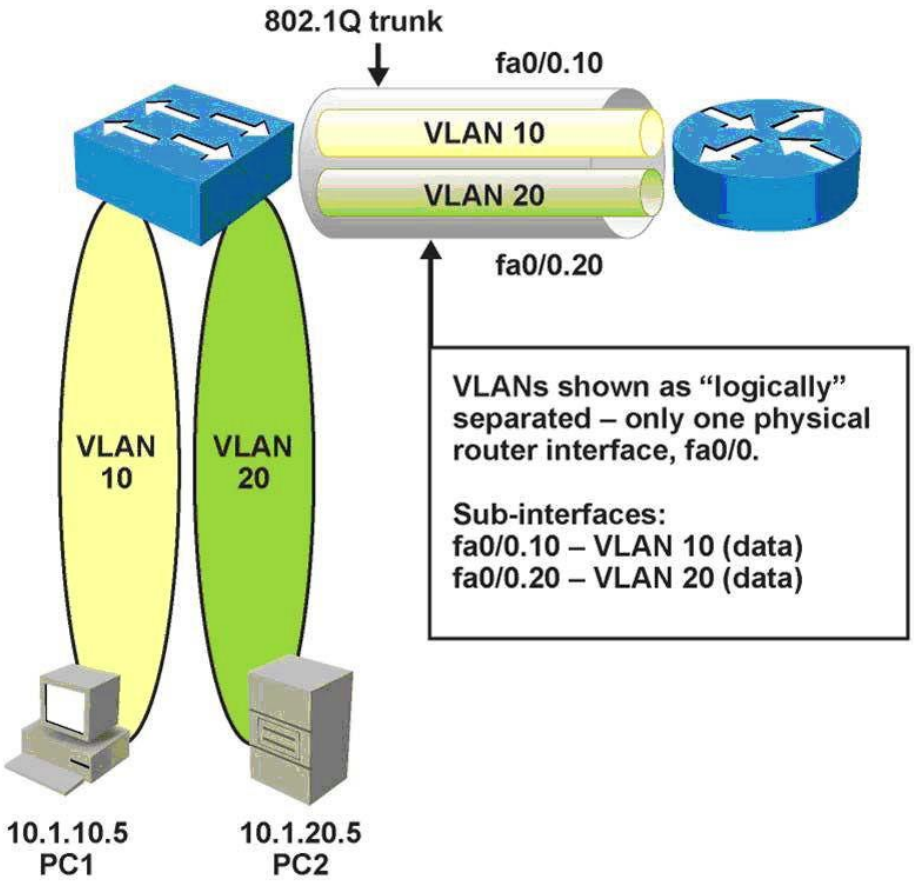
- **Routed port:** A pure Layer 3 interface similar to a routed port on a Cisco IOS router.
- **Switch virtual interface (SVI):** A virtual VLAN interface for inter-VLAN routing. In other words, SVIs are virtual routed VLAN interfaces.
- **Bridge virtual interface (BVI):** A Layer 3 virtual bridging interface. Used in some DSL applications, but not used much any more since bridging protocols across interfaces is no longer necessary.

Catalyst Switch Layer 3 Interfaces

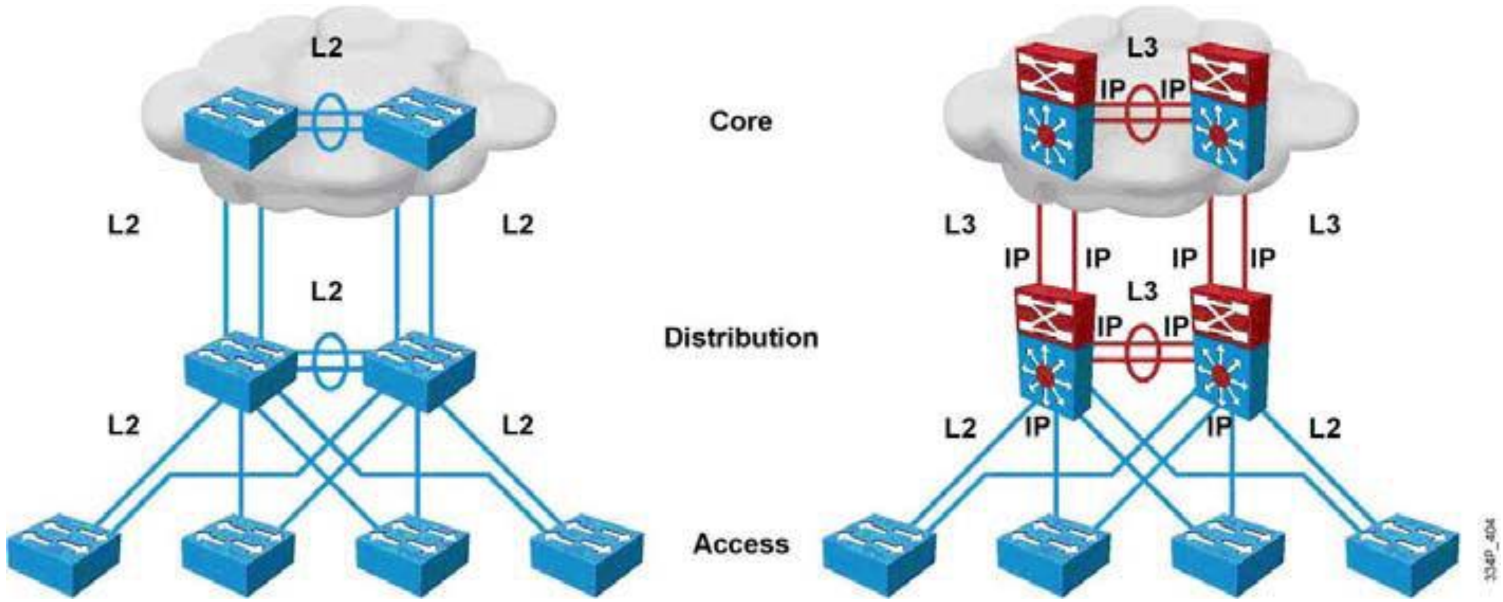
Type of Switch	Inter-VLAN Routing Capability	Inter-VLAN Routing Solution
Catalyst 2940/2950/2955/ 2960/2970	No	–
Catalyst 3560/3750/3760	Yes	Catalyst 4000 running Cisco CatOS with Supervisor I or II, using the Layer 3 module, WS-X4232-L3
Catalyst 4000/4500/4948	Yes	Catalyst 4000 with a Supervisor II+, III, IV, or V running Cisco IOS using integrated routing
Catalyst 6500	Yes	Catalyst 6500 with an MSFC, MSFC II, or MSFC III daughter card running Cisco CatOS on the supervisors and Cisco IOS on the MSFC Catalyst 6500 with MSFC, MSFC II, or MSFC III running Cisco Native IOS Catalyst 6500 using a legacy MSM

Router-on-a-Stick

- Layer 2 switch linked to router via trunk (in lieu of using a multilayer switch).
- Router interface, typically Fast Ethernet, subdivided into logical subinterfaces, one per VLAN.

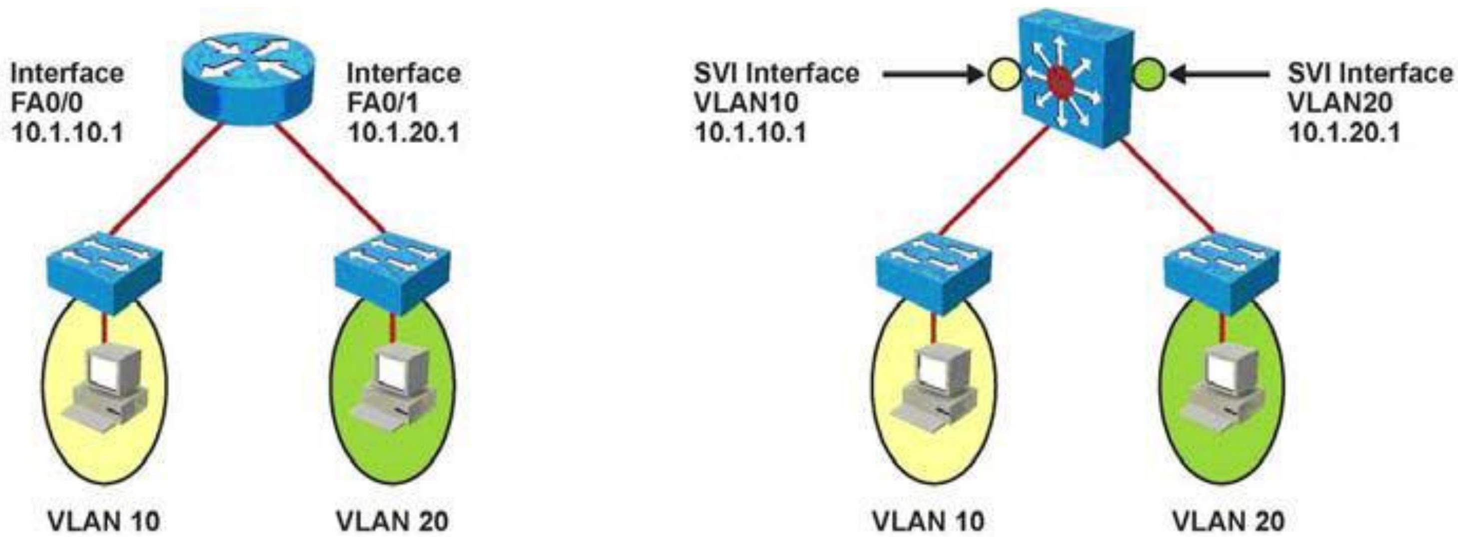


Routed/L3-Switched vs. L2 Switched Design



- Routing can now be performed at L2 switching speeds by switching frames/packets using specialized hardware circuits.
- L3 switches serve as default gateways, terminating VLANs (one IP subnet per VLAN).

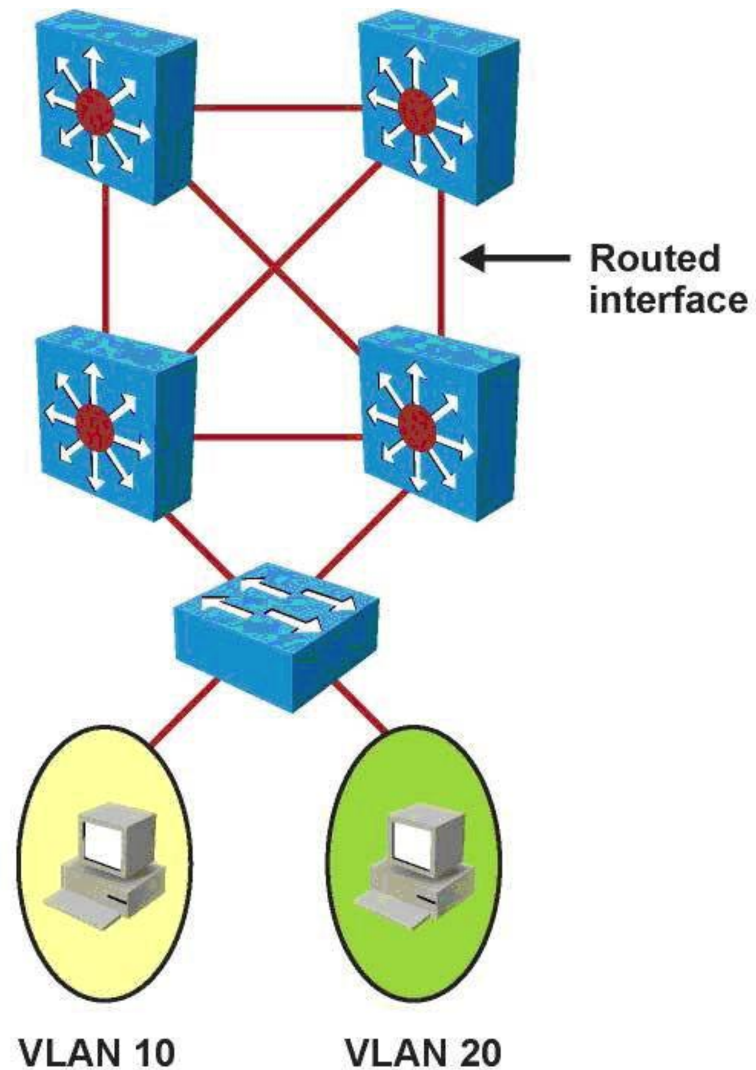
Switch Virtual Interfaces (SVI's)



- Configured on multilayer switches, one per VLAN.
- The management interface on an L2 switch is an SVI, but an L2 switch is limited to one active SVI.
- An SVI associates with an L2 VLAN – a switch must have an active L2 instance of a VLAN in order for an (L3) SVI to function.

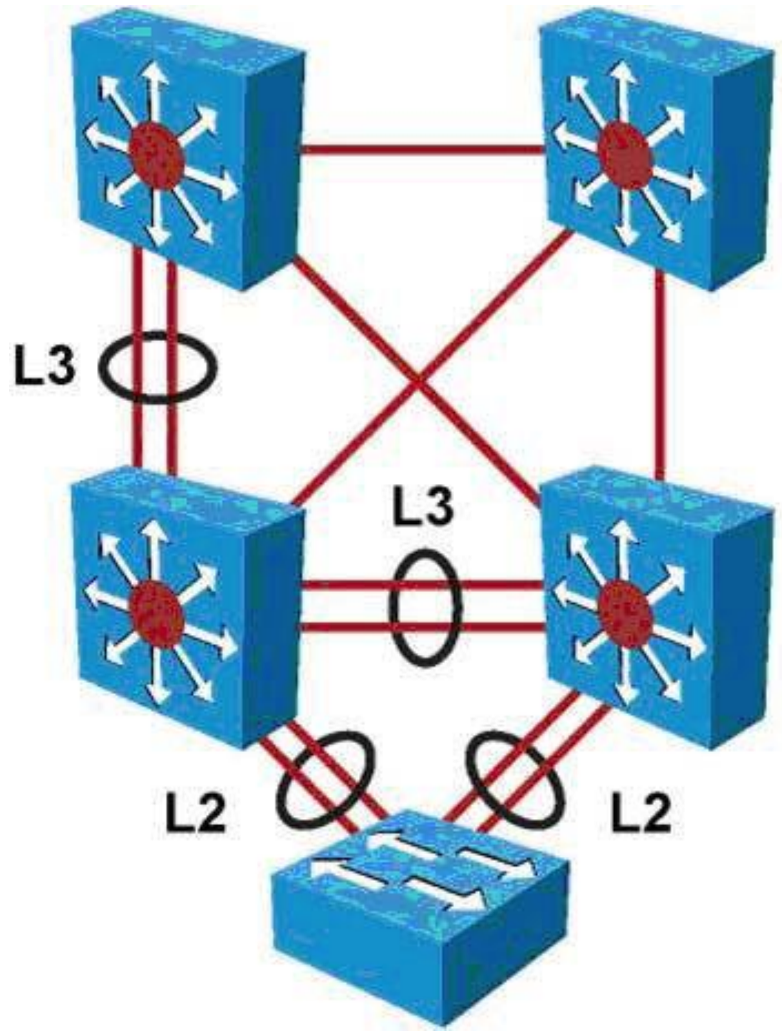
Routed Ports

- Use the **`no switchport`** command to configure a physical switch port as a routed port.
- Routed ports are used in conjunction with SVI's.
- Routed ports connect point-to-point (L3) links between distribution layer and core layer switches.
- A 48-port L3 switch can be configured as a 48-port router.



L3 EtherChannels

- Just as with physical interfaces on multilayer switches, bundles of interfaces (port channels) can be configured as routed ports.
- Port channels configured as routed ports are called L3 EtherChannels.
- L2 EtherChannels are normally used only when connecting from an access layer switch.



Configuring Inter-VLAN Routing

Configuring Router-on-a-Stick

- **Step 1.** Enable trunking on the switch port.

Switch(config-if)# switchport trunk encapsulation dot1q

Switch(config-if)# switchport mode trunk

Switch(config-if)# switchport trunk native *vlan #*

- **Step 2.** Enable the router interface.

Router(config-if)# no shutdown

- **Step 3.** Create the subinterfaces for each VLAN that requires inter-VLAN routing.

Router(config)# interface *interface_id slot/port.subinterface*

- **Step 4.** Configure the trunking encapsulation and IP address on the subinterfaces corresponding to the VLANs.

Router(config-subif)# encapsulation [**dot1q** | **isl**] *vlan-id* {**native**}

Router(config-subif)# ip address *ip_address subnet_mask*

Router-on-a-Stick Example

Here, VLAN 100 is used as native VLAN. It is a security best practice to use a dummy/unused VLAN for the native VLAN.

```
Router(config)# interface FastEthernet0/0
Router(config-if)#no shutdown
Router(config-if)# interface FastEthernet 0/0.1
Router(config-subif) description VLAN 1
Router(config-subif)# encapsulation dot1q 1 native
Router(config-subif)# ip address 10.1.1.1 255.255.255.0
Router(config-subif)# exit
Router(config)# interface FastEthernet 0/0.2
Router(config-subif)# description VLAN 2
Router(config-subif)# encapsulation dot1q 2
Router(config-subif)# ip address 10.2.2.1 255.255.255.0
Router(config-subif)# exit
Router(config)# end
#####Cisco IOS switch Trunking Configuration Connected to Interface
FastEthernet0/0
Switch(config)# interface FastEthernet 4/2
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# end
```

Configuring Inter-VLAN Routing with SVI's

- **Step 1.** Specify an SVI by using a VLAN interface command:

  Switch(config)# **interface vlan** *vlan-id*

- **Step 2.** Assign an IP address to the VLAN:

  Switch(config-if)# **ip address** *ip_address subnetmask*

- **Step 3.** Enable the interface:

  Switch(config-if)# **no shutdown**

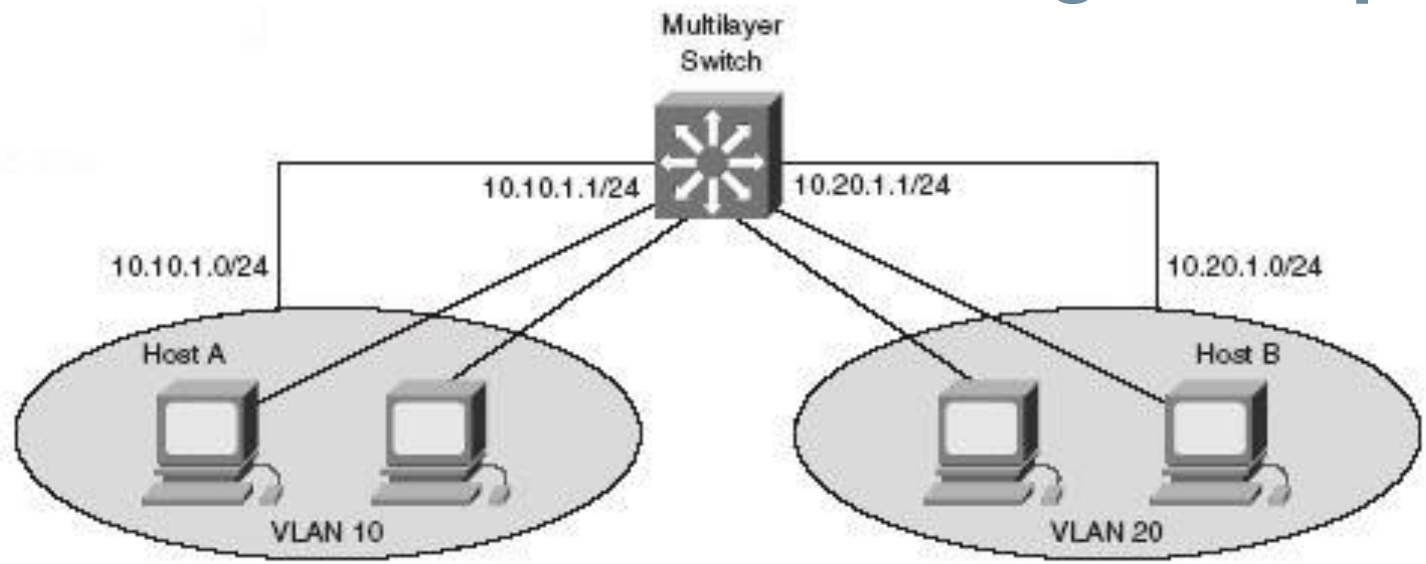
- **Step 4.** (Optional.) Enable IP routing on the router:

  Switch(config)# **ip routing**

- **Step 5.** (Optional.) Specify an IP routing protocol or use static routes:

  Switch(config)# **router** *ip_routing_protocol options*

SVI-Based Inter-VLAN Routing Example



- Switch(config)# **ip routing**
- Switch(config)# **router rip**
- Switch(config-router)# **network 10.0.0.0**
- Switch(config)# **interface vlan 10**
- Switch(config-if)# **ip address 10.10.1.1 255.0.0.0**
- Switch(config-if)# **no shutdown**
- Switch(config-if)# **interface vlan 20**
- Switch(config-if)# **ip address 10.20.1.1 255.255.255.0**
- Switch(config-if)# **no shutdown**

Configuring Routed Ports

- **Step 1**. Select the interface for configuration.

  `Switch(config)#` **`interface`** *`interface-id`*

- **Step 2**. Convert this port from a physical Layer 2 port to a physical Layer 3 interface.

  `Switch(config-if)#` **`no switchport`**

- **Step 3**. Configure the IP address and IP subnet mask. This address will be used by hosts on the segment connected to this interface for communication to the switch on this interface, or as the default gateway to other networks.

  `Switch(config-if)#` **`ip address`** *`ip_address subnet_mask`*

- **Step 4**. (Optional.) Enable IP routing on the router.

  `Switch(config)#` **`ip routing`**

- **Step 5**. (Optional.) Specify an IP routing protocol or use static routes:

  `Switch(config)#` **`router`** *`ip_routing_protocol options`*

Routed Port Example

```
Switch(config)# interface GigabitEthernet 1/1
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.10.1.1 255.255.255.252
Switch(config-if)# exit
Switch(config)# interface GigabitEthernet 1/2
Switch(config-if)# ip address 10.20.1.254 255.255.255.252
% IP addresses may not be configured on L2 links.
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.20.1.254 255.255.255.252
```

Inter-VLAN Routing Verification (1)

Verify the status of an SVI.

```
Switch# show interfaces vlan 20
Vlan20 is up, line protocol is up
Hardware is Ethernet SVI, address is 00D.588F.B604 (bia 00D.588F.B604)
Internet address is 10.1.20.1/24
MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, loopback not set
ARP type: ARPA, ARP Timeout 04:00:00
Last input never, output never, output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
0 packets input, 0 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
0 packets output, 0 bytes, 0 underruns
0 output errors, 0 interface resets
0 output buffer failures, 0 output buffers swapped out
```

Inter-VLAN Routing Verification (2)

Display the interface configuration of a routed port.

```
Switch# show running-config interface FastEthernet 2/8
Building configuration...
!
interface FastEthernet2/8
no switchport
ip address 172.16.22.2 255.255.255.252
<output omitted>
```

Inter-VLAN Routing Verification (3)

Display the IP properties on a routed port.

```
Switch# show ip interface fastethernet0/24
FastEthernet0/24 is up, line protocol is up
Internet address is 10.1.10.1/24
Broadcast address is 255.255.255.255
Address determined by setup command
MTU is 1500 bytes
Helper address is not set
Directed broadcast forwarding is disabled
Multicast reserved groups joined: 224.0.0.10
Outgoing access list is not set
Inbound access list is not set
Proxy ARP is enabled
Local Proxy ARP is disabled
Security level is default
Split horizon is enabled
ICMP redirects are always sent
ICMP unreachables are always sent
ICMP mask replies are never sent
IP fast switching is enabled
IP CEF switching is enabled
```

Common Inter-VLAN Routing Problems

Problem	Possible Cause
Missing VLAN	VLAN might not be defined across all the switches. VLAN might not be enabled on the trunk ports. Ports might not be in the right VLANs.
Layer 3 interface configuration	Virtual interface might have the wrong IP address or subnet mask. Virtual interface might not be up. Virtual interface number might not be match with the VLAN number. Routing has to be enabled to route frames between VLAN. Routing might not be enabled.
Routing protocol misconfiguration	Every interface or network needs to be added in the routing protocol. The new interface might not be added to the routing protocol. Routing protocol configuration is needed only if VLAN subnets needs to communicate to the other routers, as previously mentioned in this chapter.
Host misconfiguration	Host might not have the right IP or subnetmask. Each host has to have the default gateway that is the SVI or Layer 3 interface to communicate the other networks and VLAN. Host might not be configured with the default gateway.

Configuring Layer 3 EtherChannels

- **Step 1.** Create a virtual Layer 2 interface.

  Switch(config)# **interface port-channel 1**

- **Step 2.** Convert to a Layer 3 interface to enable IP configuration.

  Switch(config-if)# **no switchport**

- **Step 3.** Assign an IP address to the port-channel interface:

  Switch(config-if)# **ip address** *ip_address subnet_mask*

- **Step 4.** Navigate to the interfaces that are to be associated with the EtherChannel bundle:

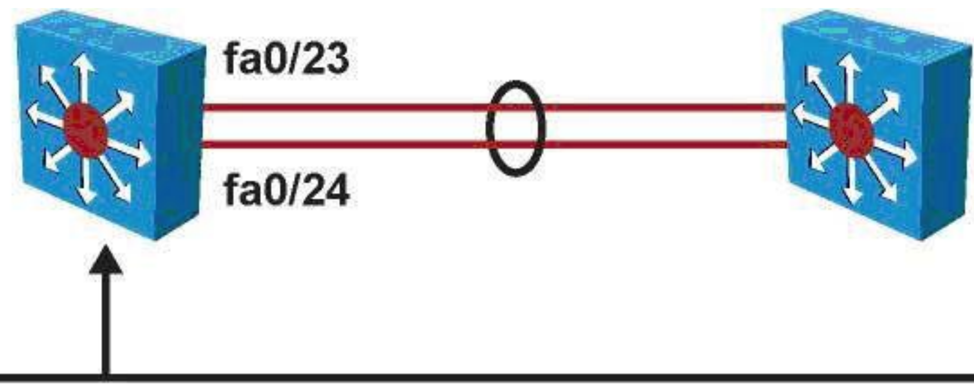
  Switch(config)# **interface range** *interface_id portnumber_range*

- **Step 5.** For a Layer 3 EtherChannel to form, the associated physical ports must be configured as Layer 3 ports. Assign the interfaces to the EtherChannel group:

  Switch(config-if-range# **no switchport**

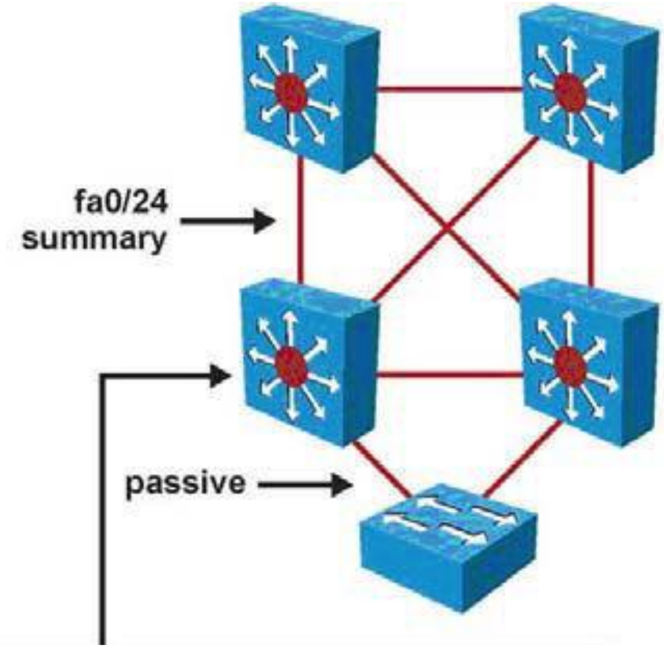
  Switch(config-if-range)# **channel-group** *channel-group-number* **mode** {**auto** [**non-silent**] | **desirable** [**non-silent**] | **on**} | {**active** | **passive**}

Layer 3 EtherChannel Example




```
switch(config)# interface fastethernet 0/23
switch(config-if)# no switchport
switch(config-if)# channel-group 1 mode on
switch(config)# interface fastethernet 0/24
switch(config-if)# no switchport
switch(config-if)# channel-group 1 mode on
switch(config)# interface port-channel 1
switch(config-if)# no switchport
switch(config-if)# ip address 10.1.20.1 255.255.255.0
```

Routing Protocol Configuration



- Switch(config)# **ip routing**
- Switch(config)# **router eigrp 100**
- Switch(config-router)# **no auto-summary**
- Switch(config-router)# **network 10.0.0.0**
- Switch(config-router)# **passive-interface default**
- Switch(config-router)# **no passive-interface fa0/24**
- Switch(config)# **interface fa0/24**
- Switch(config-if)# **description Uplink**
- Switch(config-if)# **ip summary-address eigrp 100 10.1.0.0 255.255.240.0**

Verifying Routing (1)

```
Switch# show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF,
       IA - OSPF inter area
       N1 - OSPF NSSA external type 1,
       N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1,
       L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default,
       U - per-user static route
       o - ODR, P - periodic downloaded static route
Gateway of last resort is not set
      10.0.0.0/8 is variably subnetted, 13 subnets, 2 masks
D        10.1.3.0/24 [90/28416] via 10.1.10.10, 08:09:49, Vlan10
D        10.1.2.0/24 [90/28416] via 10.1.10.10, 08:09:49, Vlan10
C        10.1.10.0/24 is directly connected, Vlan10
```

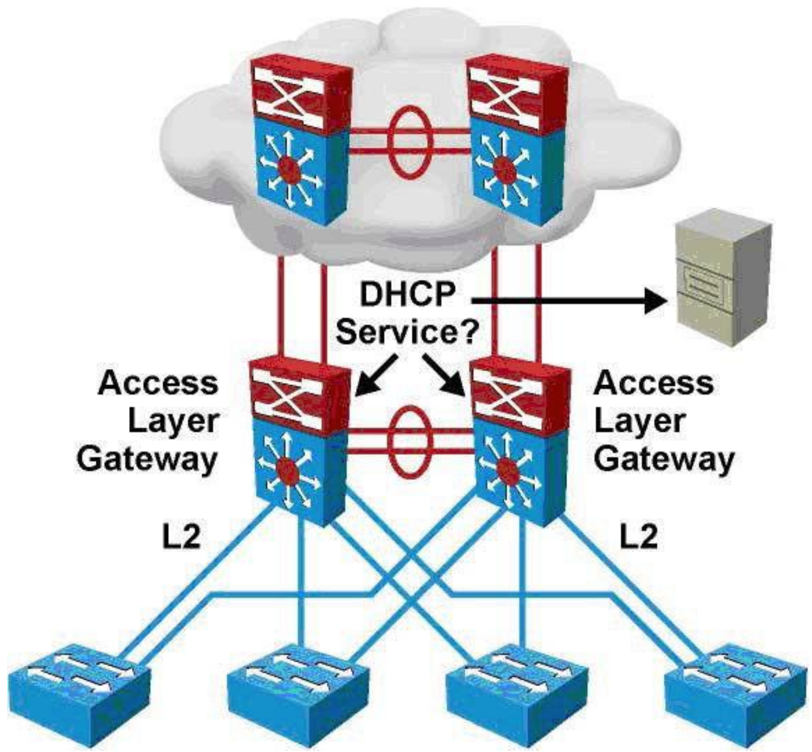
Verifying Routing (2)

```
Switch# show ip protocol
Routing Protocol is "eigrp 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  EIGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
  EIGRP maximum hopcount 100
  EIGRP maximum metric variance 1
  Redistributing: eigrp 1
  Automatic network summarization is in effect
  Maximum path: 4
  Routing for Networks:
    10.0.0.0
  Passive Interface(s):
    Vlan1
    Vlan11
  Routing Information Sources:
    Gateway              Distance       Last Update
    10.100.117.202             90       20:25:10
    10.100.113.201             90       20:25:10
  Distance: internal 90 external 170
```

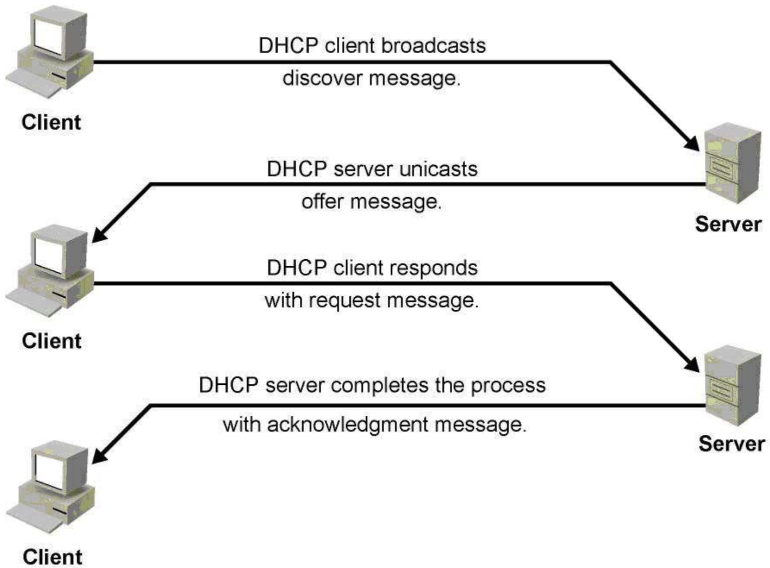
Implementing Dynamic Host Configuration in a Multilayer Switched Environment

DHCP Overview

- Distribution multilayer switches often act as Layer 3 gateways for clients connecting to the access switches on various VLANs. Therefore, the DHCP service can be provided directly by the distribution switches.
- Alternatively, DHCP services can be concentrated in an external, dedicated DHCP server. In that case, distribution switches need to redirect the incoming clients DHCP requests to the external DHCP server.



DHCP Operation



- **Step 1.** The client sends a DHCPDISCOVER broadcast message to locate a Cisco IOS DHCP server.
- **Step 2.** A DHCP server offers configuration parameters (such as an IP address, a MAC address, a domain name, and a lease for the IP address) to the client in a DHCPOFFER unicast message. A DHCP client might receive offers from multiple DHCP servers and can accept any one of the offers; however, the client usually accepts the first offer it receives. Additionally, the offer from the DHCP server is not a guarantee that the IP address will be allocated to the client; however, the server usually reserves the address until the client has had a chance to formally request the address.
- **Step 3.** The client returns a formal request for the offered IP address to the DHCP server in a DHCPREQUEST broadcast message.
- **Step 4.** The DHCP server confirms that the IP address has been allocated to the client by returning a DHCPACK unicast message to the client.

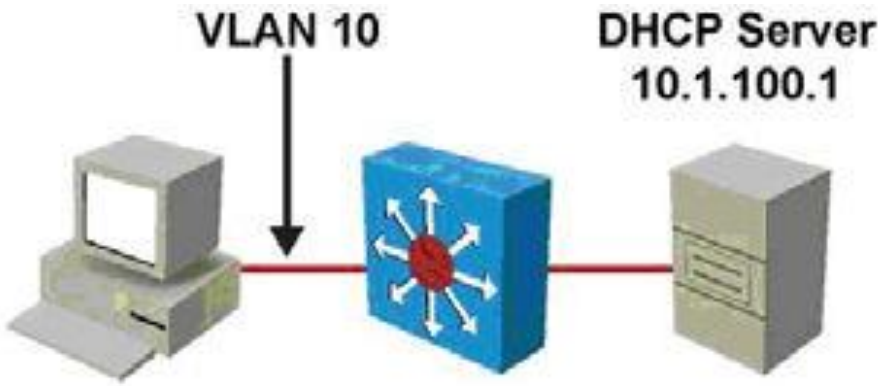
Configuring DHCP

- Step 1. Create a pool with the **`ip dhcp pool`** command.
- Step 2. Within the dhcp pool configuration submode, configure the network value, which indicates in which subnet addresses are offered. Also, configure items such as the default-gateway, lease duration, subnetmask, and DNS server IP addresses, among others.
- Step 3. By default, the switch offers addresses taken from the whole range. To exclude some addresses, in global configuration mode, use the **`ip dhcp excluded-address`** command followed by the range of addresses to exclude from the DHCP offers. For a discontinuous address range, configure excluded addresses for each DHCP scope.

```
Switch(config)# ip dhcp excluded-address 10.1.10.1 10.1.10.20
Switch(config)# ip dhcp pool XYZ10
Switch(config-dhcp)# network 10.1.10.0 255.255.255.0
Switch(config-dhcp)# default-router 10.1.10.1
Switch(config-dhcp)# option 150 10.1.1.50
Switch(config-dhcp)# lease 0 8 0
Switch(config-dhcp)# ! 0 days 8 hours 0 minutes
Switch(config)# interface vlan10
Switch(config-if)# ip address 10.1.10.1 255.255.255.0
```

DHCP Relay

- Use the **`ip helper-address`** command on the interface which connects to the subnet containing devices which request IP addresses from the DHCP server.
- On a multilayer switch, the interface "connecting" to the relevant subnet is typically an SVI.



- Switch(config)# **interface vlan10**
- Switch(config-if)# **ip address 10.1.10.1 255.255.255.0**
- Switch(config-if)# **ip helper-address 10.1.100.1**

Verifying and Troubleshooting DHCP

```
Switch# show ip dhcp binding
Bindings from all pools not associated with VRF:
IP address      Client-ID/              Lease expiration              Type
                Hardware address/
                User name
10.1.10.21      0100.1bd5.132a.d2       Jun 25 2009 06:09 AM   Automatic
10.1.10.22      0100.4096.a46a.90       Jun 25 2009 09:40 AM   Automatic
10.1.10.23      0100.4096.aa98.95       Jun 25 2009 11:28 AM   Automatic

Switch# debug ip dhcp server packet
DHCPD: DHCPDISCOVER received from client 0100.1bd5.132a.d2 on interface Vlan6.
DHCPD: Sending DHCPOFFER to client 0100.1bd5.132a.d2 (10.1.10.21).
DHCPD: broadcasting BOOTREPLY to client 001b.d513.2ad2.
DHCPD: DHCPREQUEST received from client 0100.1bd5.132a.d2.
DHCPD: Sending DHCPACK to client 0100.1bd5.132a.d2 (10.1.10.21).
DHCPD: broadcasting BOOTREPLY to client 001b.d513.2ad2.
```

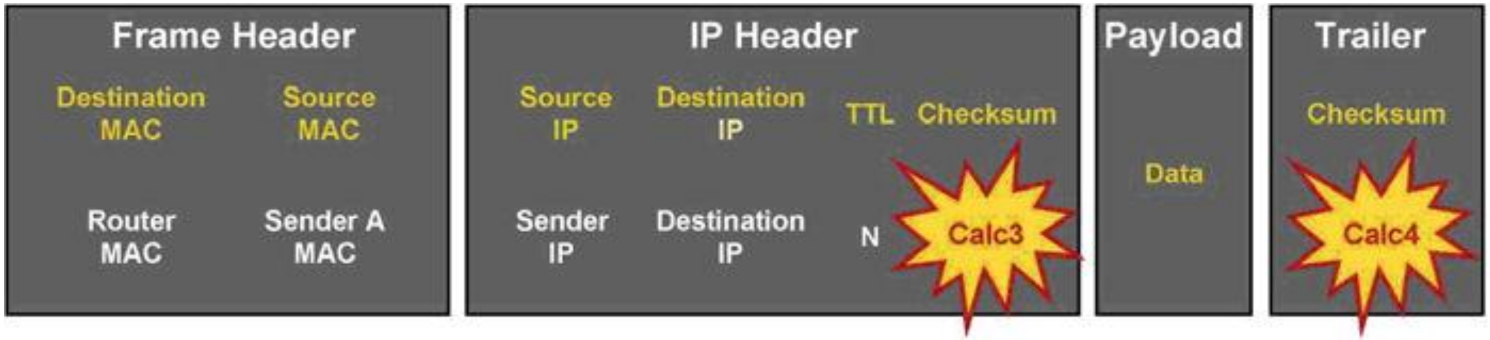
Deploying CEF-Based Multilayer Switching

Multilayer Switch Processing

- Combines functionality of switch and router
- Offloads software-based routing process (packet rewrite) to specialized ASIC hardware
- Provides wire-speed Ethernet routing and switching services
- Optimized for campus LAN
- Performs three major functions:
  - Packet switching
  - Route processing
  - Intelligent network services

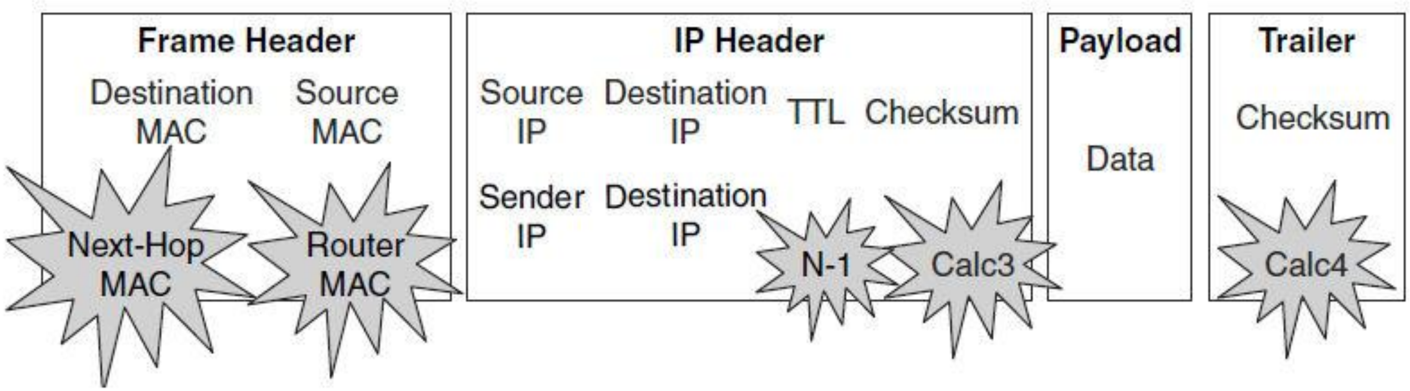
Frame Rewrite

- The incoming frame checksum is verified to ensure that no frame corruption or alteration occurs during transit.
- The incoming IP header checksum is verified to ensure that no packet corruption or alteration occurs during transit.



IP Unicast Packet Rewrite on Output Interface

- The source MAC address changes from the sender MAC address to the outgoing router MAC address.
- The destination MAC address changes from the MAC address of the router’s incoming interface to the MAC address of the next-hop router’s receiving interface.
- The TTL is decremented by one, and as a result, the IP header checksum is recalculated.
- The frame checksum is recalculated.



High-Speed Memory Tables

- Multilayer switches build routing, bridging, QoS, and ACL tables for centralized or distributed switching.
- Switches perform lookups in these tables to make decisions, such as to determine whether a packet with a specific destination IP address is supposed to be dropped according to an ACL.
- These tables support high-performance lookups and search algorithms to maintain line-rate performance.
- Multilayer switches deploy these memory tables using specialized memory architectures, referred to as content addressable memory (CAM), and ternary content addressable memory (TCAM).

CAM Table

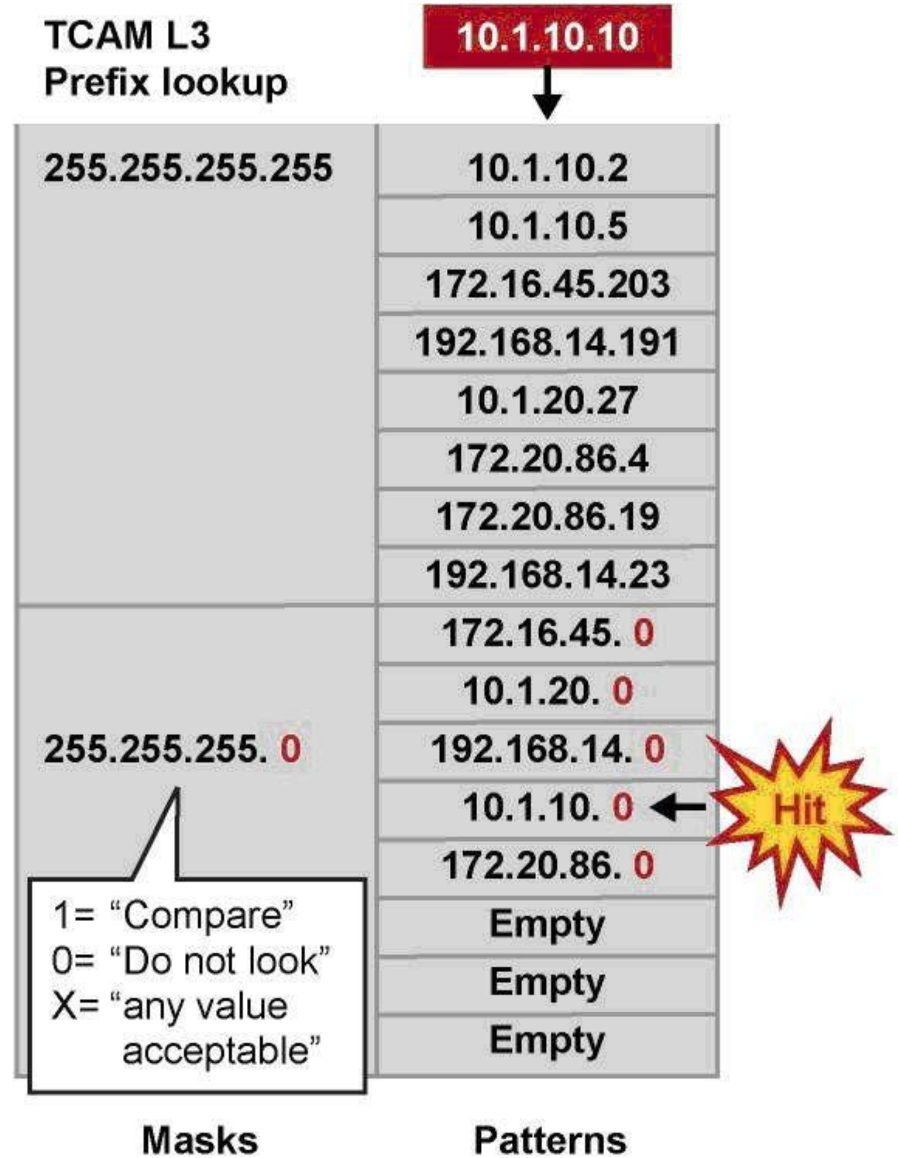
- Matches based on two values: 0 (true) or 1 (false).
- Useful for building tables that search on exact matches such as MAC address tables.
- Primary table used to make Layer 2 forwarding decisions.
- Built by recording the source MAC address and inbound port of all incoming frames. When a frame arrives at the switch with a destination MAC address of an entry in the CAM table, the frame is forwarded out through only the port that is associated with that specific MAC address.

TCAM Table

- Matches based on three values: 0, 1, or x (where x is either number).
- TCAM is most useful for building tables for searching on the longest match, such as IP routing tables organized by IP prefixes.
- The memory structure is broken into a series of patterns and associated masks.
- Stores ACL, QoS, and other information generally associated with Layer 3 and higher processing.

TCAM Lookup

- TCAM table used here for a CEF prefix lookup to make a routing decision.
- Entry, 10.1.10.10, does not appear as a pattern in the TCAM table when all bits are compared.
- Search continues for the longest match, resulting in a 24-bit hit.



TCAM Match Region Types

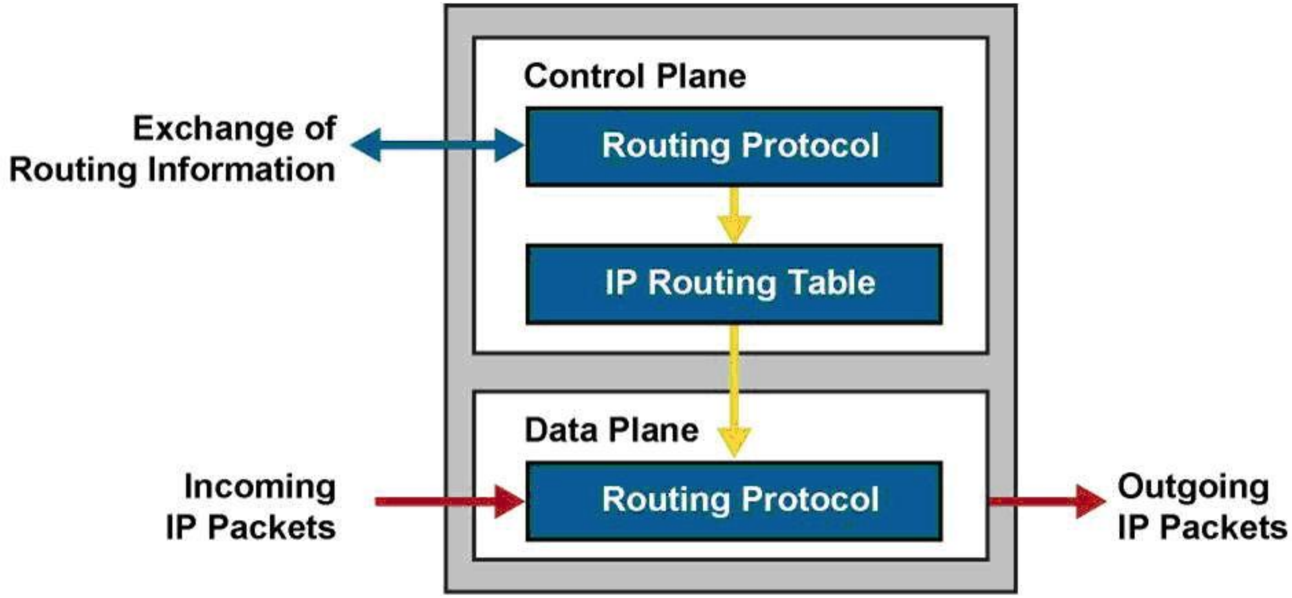
- **Exact-match region**: Layer 3 entries for IP adjacencies; IP adjacencies are next hop information, such as MAC addresses, associated with IP addresses. Other examples of exact-match regions are Layer 2 switching tables and UDP flooding tables.
- **Longest-match region**: multiple “buckets” or groups of Layer 3 address entries organized in decreasing order by mask length. All entries within a bucket share the same mask value and key size. The buckets change their size dynamically by borrowing address entries from neighboring buckets. Although the size of the whole protocol region is fixed, several platforms support configuration of the region size. For most platforms, the reconfigured size of the protocol region is effective only after the next system reboot.
- **First-match region**: Stops lookups after the first match of the entry. For example, a first-match region is used for ACL entries.

TCAM Protocol Regions

Region Name	Cisco IOS Region Name	Lookup Type	Key Size	Sample Result
IP adjacency	ip-adjacency	Exact-match	32 bits	MAC address rewrite information
IP prefix	ip-prefix	Longest-match	32 bits	Next-hop routing information
IP multicast	ip-mcast	Longest-match	64 bits	Next-hop routing information
Layer 2 switching	l2-switching	Exact-match	64 bits	Destination interface and VLAN
UDP flooding	udp-flooding	Exact-match	64 bits	Next-hop routing or MAC address rewrite information
Access Lists	access-list	First-match	128 bits	Permit, deny, or wildcard

Distributed Hardware Forwarding

- Layer 3 switching software employs a distributed architecture in which the control path and data path are relatively independent.
- The control path code, such as routing protocols, runs on the route processor. (For example the Routing Engine in the Control Plane and the Forwarding Engine in the Data Plane use routing protocols to perform the routing process.)
- Each interface module includes a microcoded processor that handles all packet forwarding. The Ethernet interface module and the switching fabric forward most of the data packets.



Cisco Switching Methods

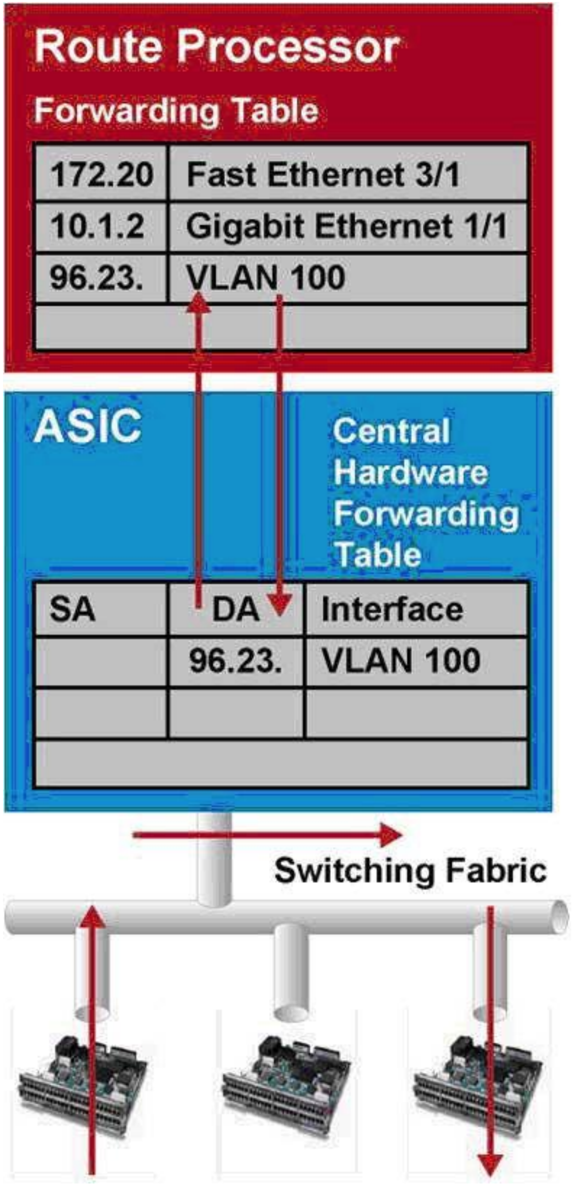
- **Process Switching**: Router strips off the Layer 2 header for each incoming frame, looks up the Layer 3 destination network address in the routing table for each packet, and then sends the frame with rewritten Layer 2 header, including computed cyclic redundancy check (CRC), to the outgoing interface. All these operations are done by software running on the CPU for each individual frame. Process switching is the most CPU-intensive method available in Cisco routers. It can greatly degrade performance and is generally used only as a last resort or during troubleshooting.
- **Fast Switching:** After the lookup of the first packet destined for a particular IP network, the router initializes the fast-switching cache used by the fast switching mode. When subsequent frames arrive, the destination is found in this fast-switching cache. The frame is rewritten with corresponding link addresses and is sent over the outgoing interface.
- **Cisco Express Forwarding (CEF):** The default-switching mode. CEF is less CPU-intensive than fast switching or process switching. A router with CEF enabled uses information from tables built by the CPU, such as the routing table and ARP table, to build hardware-based tables known as the Forwarding Information Base (FIB) and adjacency tables. These tables are then used to make hardware-based forwarding decisions for all frames in a data flow, even the first. Although CEF is the fastest switching mode, there are limitations, such as other features that are not compatible with CEF or rare instances in which CEF functions can actually degrade performance, such as CEF polarization in a topology using load-balanced Layer 3 paths.

Cisco Forwarding Decision Methods

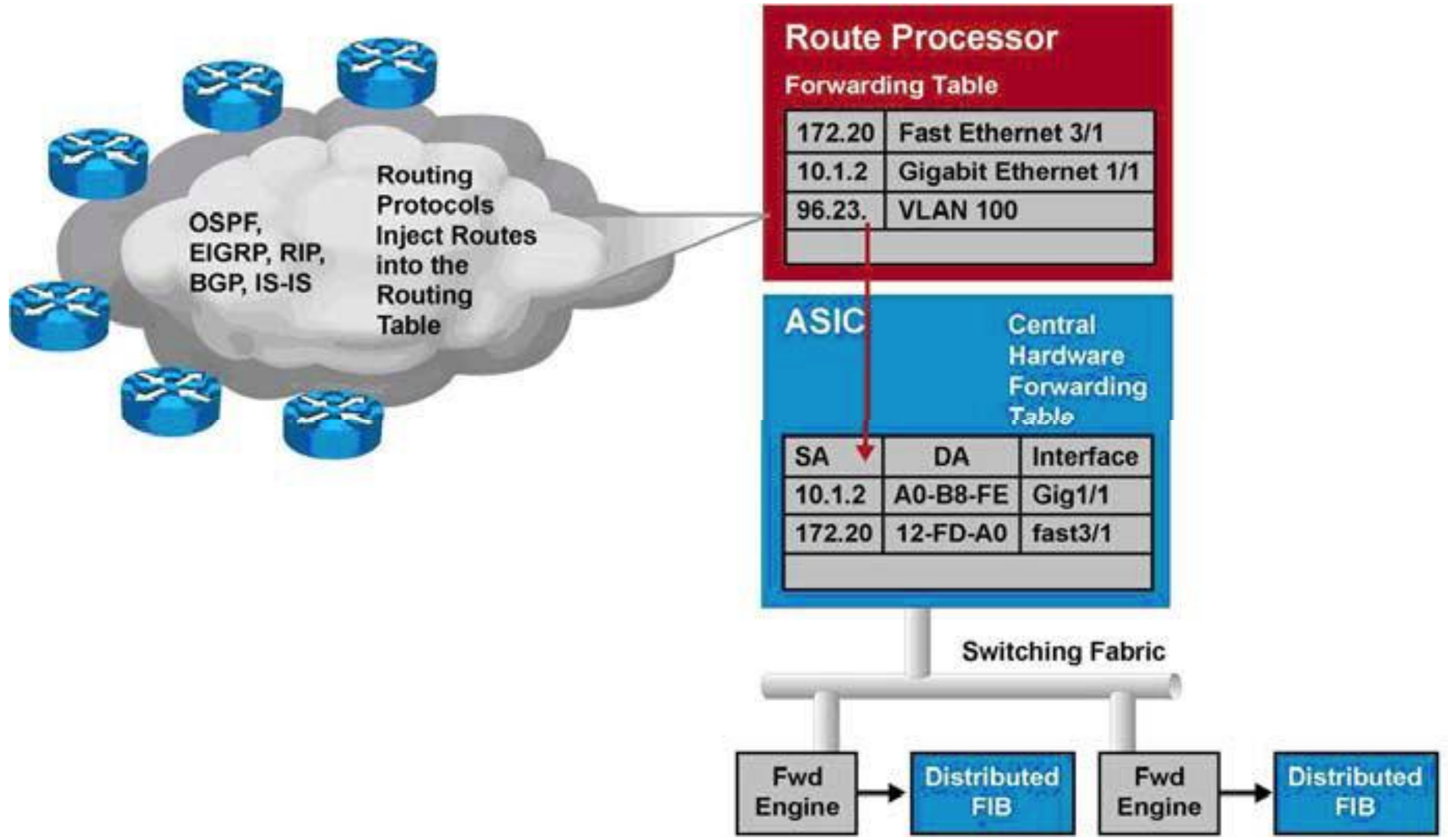
- **Route caching:** Also known as flow-based or demand-based switching, a Layer 3 route cache is built within hardware functions as the switch sees traffic flow into the switch. This is functionally equivalent to Fast Switching in the Cisco router IOS.
- **Topology-based switching:** Information from the routing table is used to populate the route cache, regardless of traffic flow. The populated route cache is called the FIB. CEF is the facility that builds the FIB. This is functionally equivalent to CEF in the Cisco router IOS.

Route Caching

- First packet in a stream is switched in software by the route processor.
- Information is stored in cache table as a flow.
- All subsequent packets are switched in hardware.



Topology-Based Switching



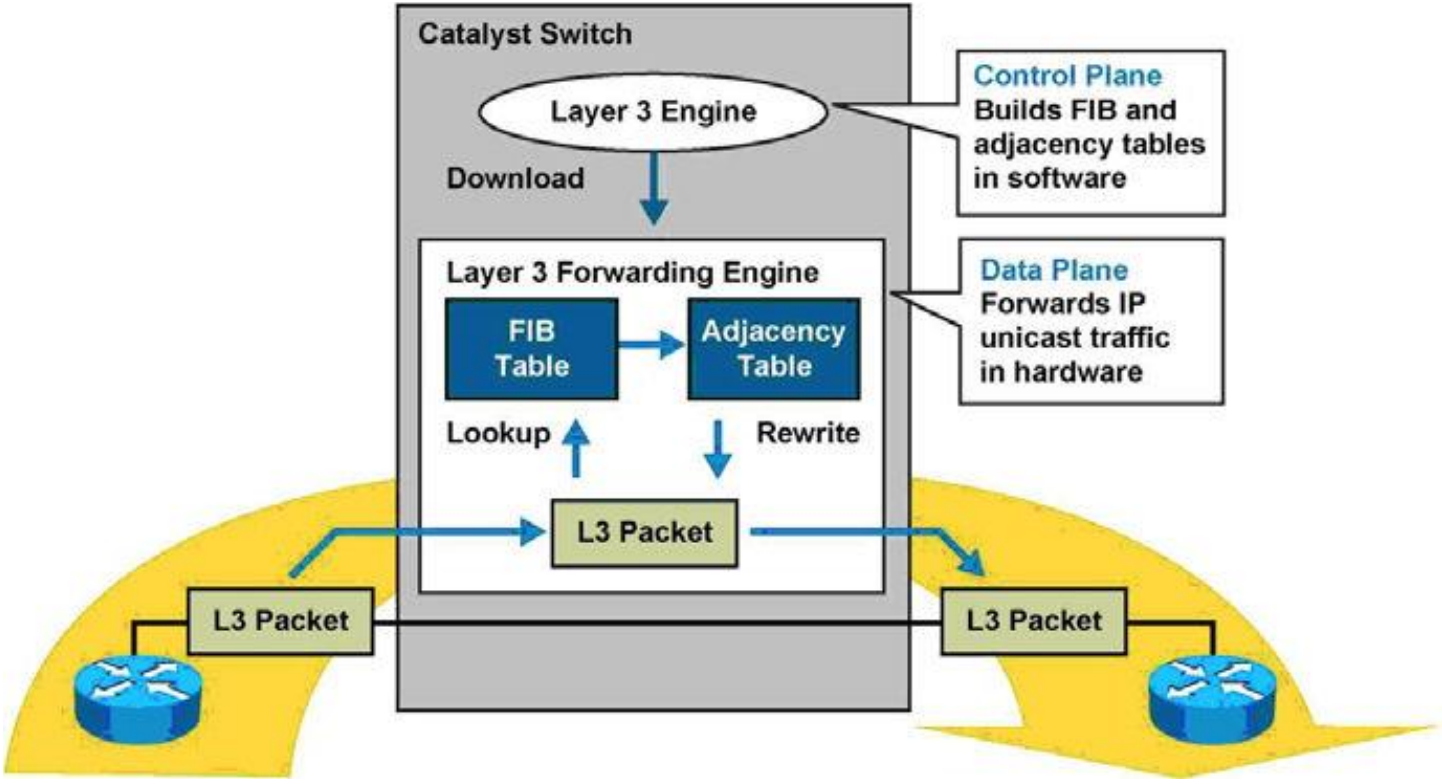
- Faster than route caching. Even first packet forwarded by hardware.
- CEF populates FIB with information from routing table.

CEF Switching Locations

- **Centralized switching:** Carries out forwarding decisions on a specialized ASIC that is central to all interfaces of a Layer 3 switch.
- **Distributed switching (dCEF):** Interfaces or line modules on Layer 3 switches handle forwarding decisions independently. With distributed switching, a centralized switching engine synchronizes Layer 3 forwarding, routing, and rewrite tables to local tables on distributed switching–capable modules. As a result, individual line cards or ports make forwarding decisions without the aid of the centralized switching engine; frames pass between ports directly across the fabric. In other words, switches using distributed switching place additional copies of the CEF FIB and adjacency table on line modules or interfaces for routing and switching of frames.

CEF Processing

- CEF uses special strategies to switch data packets to their destinations expediently. It caches the information generated by the Layer 3 routing engine even before the switch encounters any data flows.
- CEF caches routing information in one table (FIB) and caches Layer 2 next-hop addresses and frame header rewrite information for all FIB entries in another table, called the adjacency table (AT).



Forwarding Information Base (FIB)

- Derived from the IP routing table.
- Arranged for maximum lookup throughput.
- IP destination prefixes stored in TCAM, from most-specific to least-specific entry.
- FIB lookup based on Layer 3 destination address prefix (longest match) – matches structure of CEF entries within the TCAM.
- When TCAM full, wildcard entry redirects frames to the Layer 3 engine.
- Updated after each network change but only once. Each change in the IP routing table triggers a similar change in the FIB.
- Contains all known routes. Contains all next-hop addresses associated with all destination networks.

Adjacency Table (AT)

- Derived from ARP table and contains Layer 2 header rewrite (MAC) information for each next hop contained in the FIB. Nodes in network are said to be adjacent if they are within a single hop from each other.
- Maintains Layer 2 next-hop addresses and link-layer header information for all FIB entries.
- Populated as adjacencies are discovered.
- Each time adjacency entry created (such as via ARP), a Layer 2 header for that adjacent node is pre-computed and stored in the adjacency table.
- When the adjacency table is full, a CEF TCAM entry points to the Layer 3 engine to redirect the adjacency.

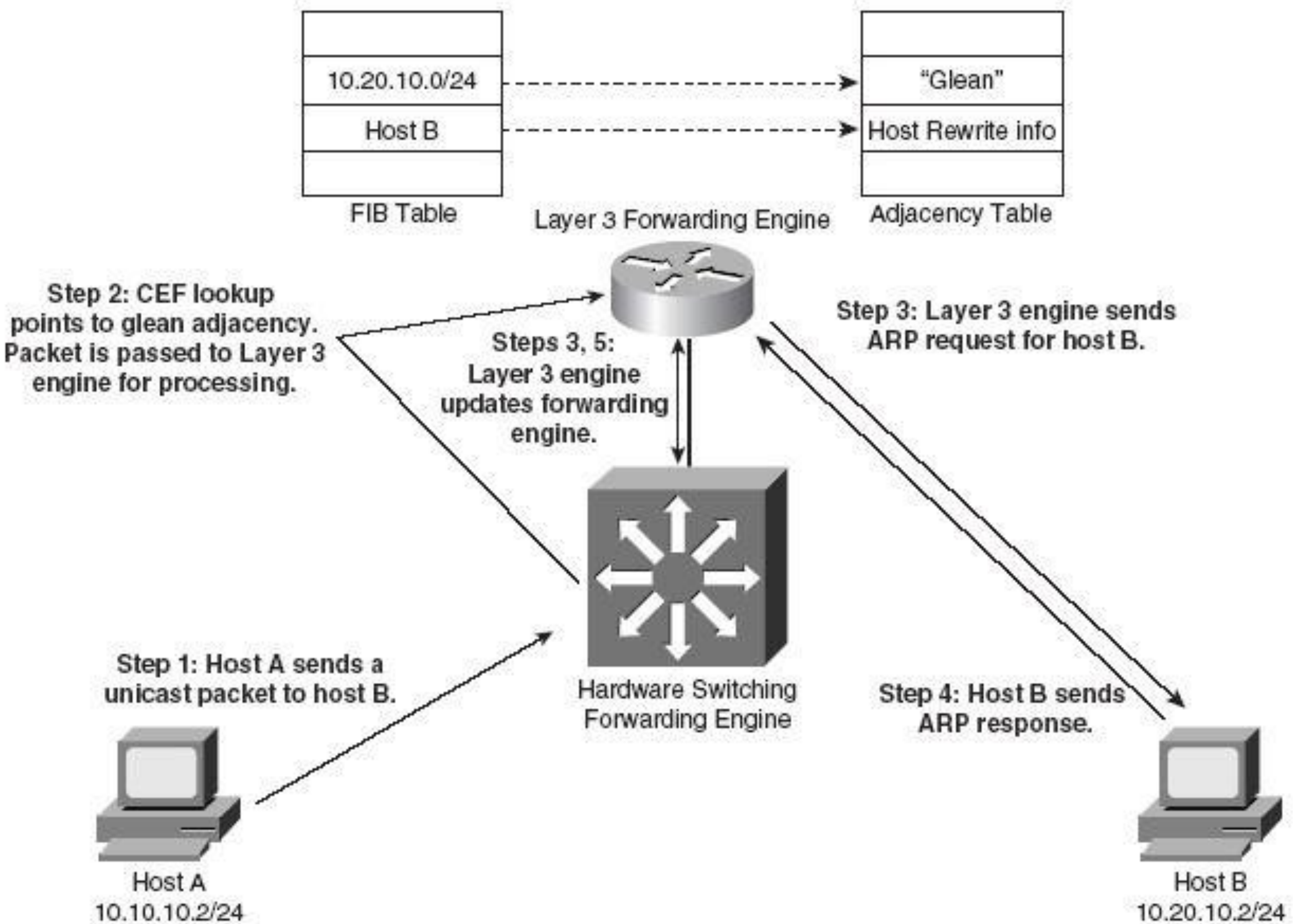
Types of Adjacencies

- **Punt adjacency:** Used for packets that require special handling by the Layer 3 engine or for features that are not yet supported by hardware switching.
- **Drop or discard adjacency:** Used to drop ingress packets.
- **Null adjacency:** Used to drop packets destined for a Null0 interface. The use of a Null0 interface is for access filtering of specific source IP packets.

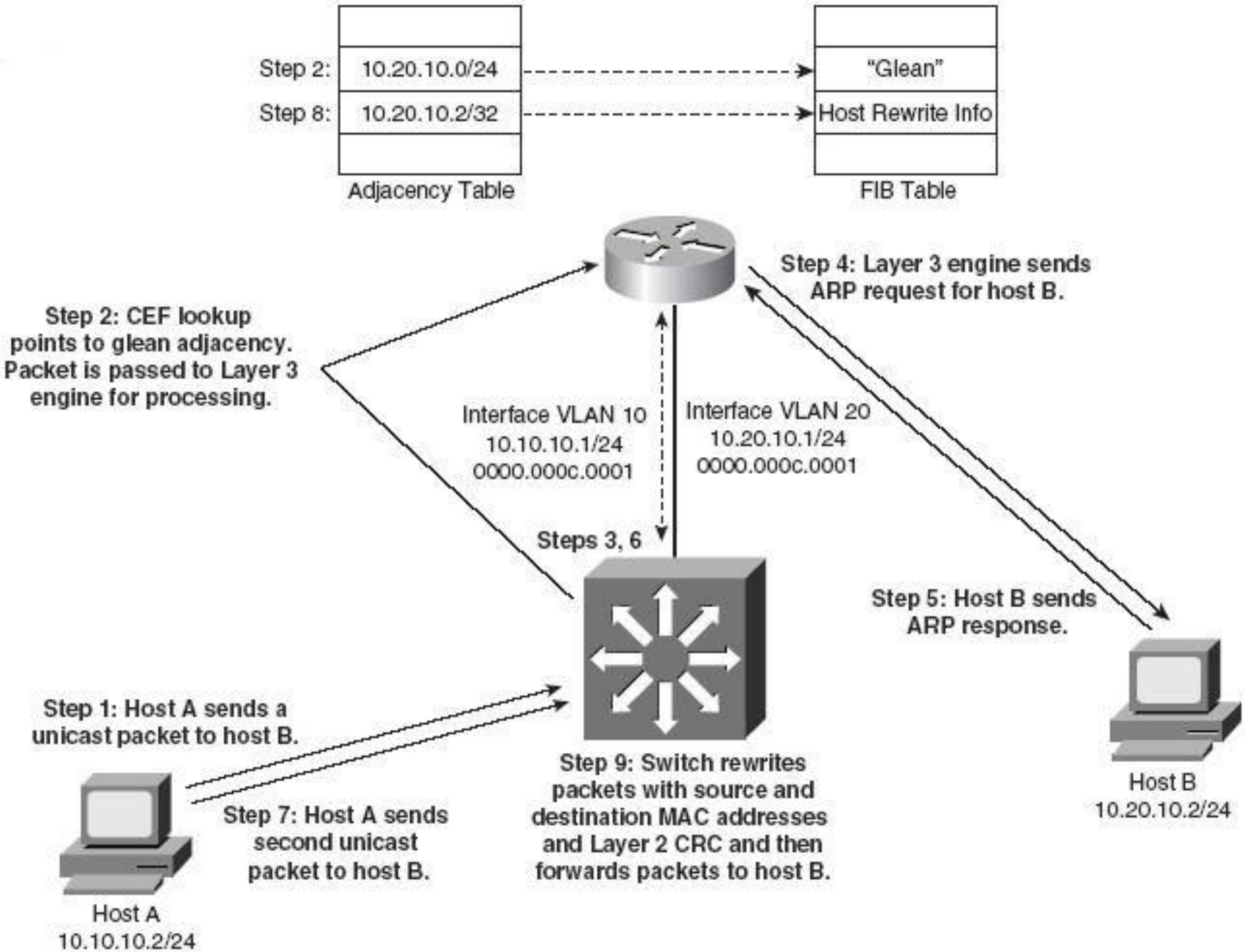
Packet Types Forcing Software Processing

- Use of IP header options (packets that use TCP header options are switched in hardware because they do not affect the forwarding decision).
- Have an expiring IP TTL counter
- Forwarded to a tunnel interface
- Arrive with non-supported encapsulation types
- Routed to interface with non-supported encapsulation type
- Exceed the maximum transmission unit (MTU) of an output interface and must be fragmented
- Network Address Translation (NAT)

ARP Throttling



CEF Operation



CEF Load Sharing

- Up to 6 adjacencies for a single FIB entry on a Catalyst 6500 – for load sharing per destination.
- CEF selects a particular adjacency based on the hash of the following packet characteristics (default varies with Catalyst switch families):
  - Source IP address
  - Destination IP address
  - Source and destination IP Layer 4 ports
- Because CEF by default would always select the same path for a given host pair, CEF “polarizes” the traffic. CEF polarization decreases as the number of host-pairs increase. In smaller networks, CEF tuning may be needed.

Configuring CEF

- Cisco Catalyst switches that use the CEF-based MLS architecture use CEF by default.
- For Catalyst switches that support CEF-based MLS, CEF and per-destination load balancing with CEF are enabled by default. As a result, no configuration is required for CEF-based MLS.
- Network engineers should not disable CEF on Catalyst switches for any reason except under the supervision of a Cisco TAC engineer for the specific purpose of troubleshooting.
- Disabling CEF on Cisco Catalyst switches yields low switching performance and can result in undesirable behavior.

Verifying CEF

To verify CEF information, use the following commands to help verify any issues:

- View statistics for hardware switching Layer 3 packets.

  show interface *type number*

- Verify the FIB.

  show ip cef

- Verify detailed information about a particular vlan or interface.

  show ip cef [*type mod/port* | *vlan_interface*] [**detail**]

- Verify adjacency table.

  show adjacency *type mod/port* | **port-channel** *number*} | **detail** | **internal** | **summary** –

CEF Verification Example (1)

Display L3 Switching Statistics on Cisco IOS-Based Catalyst 6500.

```
Router# show interface port-channel 9
Port-channel9 is up, line protocol is up (connected)
Hardware is EtherChannel, address is 00d0.039b.e80a (bia 00d0.039b.e800)
Description: POINT-TO-POINT TO CORE-4
! Output omitted for brevity
Output queue: 0/40 (size/max)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
L2 Switched: ucast: 205744 pkt, 34282823 bytes - mcast: 216245 pkt, 66357101 bytes
L3 in Switched: ucast: 367825 pkt, 361204150 bytes - mcast: 0 pkt, 0 bytes mcast
L3 out Switched: ucast: 248325 pkt, 243855150 bytes 682964 packets input, 431530341 bytes,
0 no buffer
Received 311465 broadcasts (50899 IP multicast)
0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
0 watchdog, 0 multicast, 0 pause input
0 input packets with dribble condition detected
554167 packets output, 309721969 bytes, 0 underruns
0 output errors, 0 collisions, 8 interface resets
0 babbles, 0 late collision, 0 deferred
0 lost carrier, 0 no carrier, 0 PAUSE output
0 output buffer failures, 0 output buffers swapped out
```

CEF Verification Example (2)

Display CEF FIB on a multilayer switch.

```
Switch# show ip cef
Prefix Next Hop Interface
0.0.0.0/32 receive
1.0.0.0/24 attached GigabitEthernet0/2
1.0.0.0/32 receive
1.0.0.1/32 receive
1.0.0.55/32 1.0.0.55 GigabitEthernet0/2
```

CEF Verification Example (3)

Display CEF FIB details for a VLAN.

```
Switch# show ip cef vlan 10 detail
IP CEF with switching (Table Version 11), flags=0x0
10 routes, 0 reresolve, 0 unresolved (0 old, 0 new), peak 0
13 leaves, 12 nodes, 14248 bytes, 14 inserts, 1 invalidations
0 load sharing elements, 0 bytes, 0 references
universal per-destination load sharing algorithm, id 4B936A24
2(0) CEF resets, 0 revisions of existing leaves
Resolution Timer: Exponential (currently 1s, peak 1s)
0 in-place/0 aborted modifications
refcounts: 1061 leaf, 1052 node

Table epoch: 0 (13 entries at this epoch)

10.1.10.0/24, version 6, epoch 0, attached, connected
0 packets, 0 bytes
via Vlan10, 0 dependencies
valid glean adjacency
```

CEF Verification Example (4)

Display CEF adjacency table information.

```
Switch# show adjacency
Protocol        Interface               Address
IP              GigabitEthernet0/3      2.0.0.55(5)
IP              GigabitEthernet0/2      1.0.0.55(5)



Switch# show adjacency gigabitethernet 1/5 detail
Protocol        Interface               Address
IP              GigabitEthernet1/5      172.20.53.206(11)
                                        504 packets, 6110 bytes
                                        00605C865B82       <-- Next-Hop Mac
                                        000164F83FA50800   <-- Local MAC+Ethertype
                                        ARP     03:49:31
```

Troubleshooting CEF

- **Step 1.** Verify that the IP routing information on the Layer 3 engine is correct. Use the **show ip route** or **show ip route** *destination-network* command to verify that the destination network routing entry exists and is associated with a valid next-hop address. If the route does not exist or the next-hop address is incorrect, troubleshooting of routing protocol, next-hop interfaces, or route configuration is required.
- **Step 2.** Verify that the next-hop address has a valid next-hop MAC address by using the **show ip arp** *ip-address* command. If the entry is incomplete, troubleshooting of the ARP process is required.
- **Step 3.** Verify that the IP route entry in the FIB on the Layer 3 engine contains the same next-hop address as in Step 1 by using the **show ip cef** *destination-network* command.
- **Step 4.** Verify that the CEF adjacency table contains the same rewrite information as the ARP table from Step 2 by using the **show adjacency detail | begin** *next_hop_IP_address* command.
- **Step 5.** When all other troubleshooting steps have been exhausted and the CEF-based MLS switch is still experiencing unicast routing issues, verify the population of the FIB and adjacency table in TCAM under the supervision of a TAC engineer.

Chapter 3 Summary (1)

- This chapter discussed in detail Layer 3 routing and its implementation, including coverage of inter-VLAN routing and router-on-a-stick, DHCP services, and the forwarding path of multilayer switching using CEF.
- Inter-VLAN routing provides communication between the devices in different VLANs. Devices in different VLANs cannot communicate beyond VLAN boundaries without a Layer 3 device. Multilayer switches support two types of Layer 3 interfaces: routed ports and SVIs (VLAN interfaces).
- Routed ports are point-to-point connections such as those that interconnect the building distribution submodules and the campus backbone submodules.
- SVIs are VLAN interfaces that route traffic between VLANs. In multilayer switched networks with Layer 3 in the distribution layer and Layer 2 in the access layer, SVIs route traffic from VLANs on the access-layer switches.

Chapter 3 Summary (2)

- Using router-on-a -stick is an alternative and legacy method of implementing inter-VLAN routing for low-throughput and latency-tolerant applications.
- On multilayer switches, Layer 3 links can be aggregated using Layer 3 EtherChannels. When a Layer 3 interface is configured, routing can be enabled.
- DHCP functions can be configured on the switches.
- Multilayer switches can forward traffic based on either Layer 2 or Layer 3 header information. Multilayer switches rewrite frame and packet headers using information from tables cached in hardware. Multilayer switching is high-performance packet switching in hardware. Multilayer switching can use centralized or distributed switching, and route caching or topology-based switching. Multilayer switching functionality can be implemented using CEF, which utilizes two tables in hardware to forward packets: a Forwarding Information Base (FIB) and an Adjacency Table (AT).

Chapter 3 Labs

- **SW-LAB-2.1**
  - **Inter Vlan Routing**

- **SW-LAB-2.2**
  - **DHCP**

Q&A