University of Science and Technology

Faculty of Computer Science and Information Technology

Computer Science Department

Computer Networking /Semester (8)

Lecture (6)

Ethernet

5.4.2 Ethernet Frame Structure

We can learn a lot about Ethernet by examining the Ethernet frame, which is shown in Figure 5.20. To give this discussion about Ethernet frames a tangible context, let's consider sending an IP datagram from one host to another host, with both hosts on the same Ethernet LAN (for example, the Ethernet LAN in Figure 5.17.)

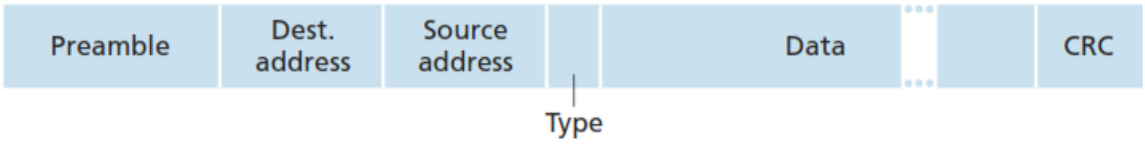

Figure 5.20 ♦ Ethernet frame structure

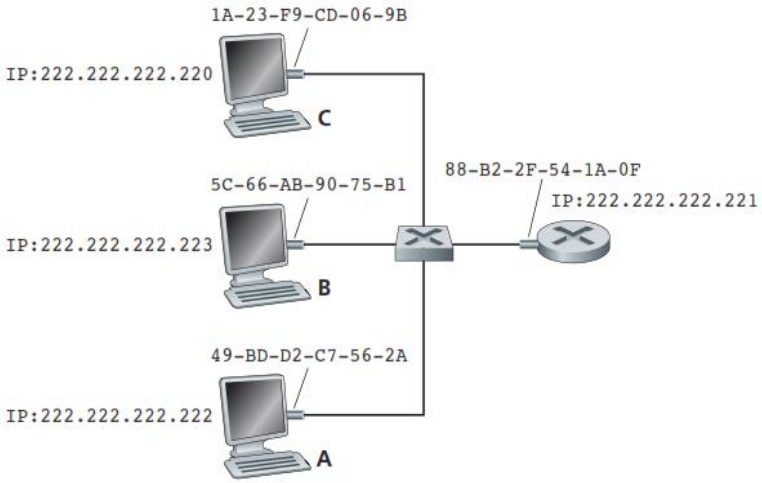

Figure 5.17 ♦ Each interface on a LAN has an IP address and a MAC address

Let the sending adapter, adapter A, have the MAC address AA-AA-AA-AA-AA-AA and the receiving adapter, adapter B, have the MAC address BB-BB-BB-BB-BB-BB. The sending adapter encapsulates the IP datagram within an Ethernet frame and passes the frame to the physical layer.

The receiving adapter receives the frame from the physical layer, extracts the IP datagram, and passes the IP datagram to the network layer.

Let's now examine the six fields of the Ethernet frame, as shown in Figure 5.20.

- ***Data field (46 to 1,500 bytes).***

This field carries the IP datagram. The maximum **transmission unit (MTU) of Ethernet is 1,500 bytes.** This means that if the IP datagram exceeds 1,500 bytes, then the host has to **fragment the datagram**. **The minimum size of the data field is 46 bytes.** This means that if the IP datagram is less than 46 bytes, the data field has to be "stuffed" to fill it out to 46 bytes. When stuffing is used, the data passed to the network layer contains the stuffing as well as an IP datagram. **The network layer uses the length field in the IP datagram header to remove the stuffing.**

- *Destination MAC address (6 bytes).*

This field contains the MAC address of the destination adapter, BB-BB-BB-BB-BB-BB. When adapter B receives an Ethernet frame whose destination address is either BB-BB-BB-BB-BB-BB or the MAC broadcast address, it passes the contents of the frame's data field to the network layer; if it receives a frame with any other MAC address, it discards the frame.

- ***Source MAC address (6 bytes).*** This field contains the MAC address of the adapter that transmits the frame onto the LAN, in this example, AA-AA-AA-AA-AA-AA.

- ***Type field (2 bytes).***

The type field permits **Ethernet to multiplex network-layer protocols.** To understand this, we need to keep in mind that hosts can use other network-layer protocols besides IP. In fact, a given host may support multiple network-layer protocols using different protocols for different applications. For this reason, when the Ethernet frame arrives at adapter B, adapter B needs to know to which network-layer protocol it should pass (that is, demultiplex) the contents of the data field. IP and other network-layer protocols (for example, Novell IPX or AppleTalk) each has their own, standardized type number.

Furthermore, the ARP protocol (discussed in the previous section) has its own type number, and if the arriving frame contains an ARP packet (i.e., has a type field of 0806 hexadecimal), the ARP packet will be demultiplexed up to the ARP protocol.

• Cyclic redundancy checks (CRC) (4 bytes).

As discussed in Section 5.2.3, the purpose of the CRC field is to allow the receiving adapter, adapter B, to detect bit errors in the frame.

• Preamble (8 bytes).

The Ethernet frame begins with an 8-byte preamble field. Each of the first 7 bytes of the preamble has a value of 10101010; the last byte is 10101011. The first 7 bytes of the preamble serve to "wake up" the receiving adapters and to synchronize their clocks to that of the sender's clock.

Why should the clocks be out of synchronization?

Keep in mind that adapter A aims to transmit the frame at 10 Mbps, 100 Mbps, or 1 Gbps, depending on the type of Ethernet LAN. However, because nothing is absolutely perfect, adapter A will not transmit the frame at exactly the target rate; there will always be some ***drift*** from the target rate, a drift which is not known *a priori* by the other adapters on the LAN. A receiving adapter can lock onto adapter A's clock simply by locking onto **the bits in the first 7 bytes of the preamble**. The last 2 bits of the eighth byte of the preamble (the first

two consecutive 1s) alert adapter B that the "important stuff" is about to come.

Ethernet is Connectionless

All of the Ethernet technologies provide connectionless service to the network layer. That is, when adapter A wants to send a datagram to adapter B, adapter A encapsulates the datagram in an Ethernet frame and sends the frame into the LAN, without first handshaking with adapter B. This layer-2 connectionless service is analogous to IP's layer-3 datagram service and UDP's layer-4 connectionless service.

5.4.3 Reliability

Ethernet technologies provide an unreliable service to the network layer. Specifically, when adapter B receives a frame from adapter A, it runs the frame through a CRC check, **but neither sends an acknowledgment when a frame passes the CRC check nor sends a negative acknowledgment when a frame fails the CRC check**. When a frame fails the CRC check, adapter B simply discards the frame.

Thus, adapter A has no idea whether its transmitted frame reached adapter B and passed the CRC check. This lack of reliable transport (at the link layer) helps to make Ethernet simple and cheap. But it also means that the stream of datagrams passed to the network layer can have gaps.

5.4.4 Virtual Local Area Networks (VLANs)

In our earlier discussion of Figure 5.15, we noted that modern institutional LANs are often configured hierarchically; with each workgroup (department) having its own switched LAN connected to the switched LANs of other groups via a switch hierarchy. *While such a configuration works well in an ideal world, the real world is often far from ideal.*

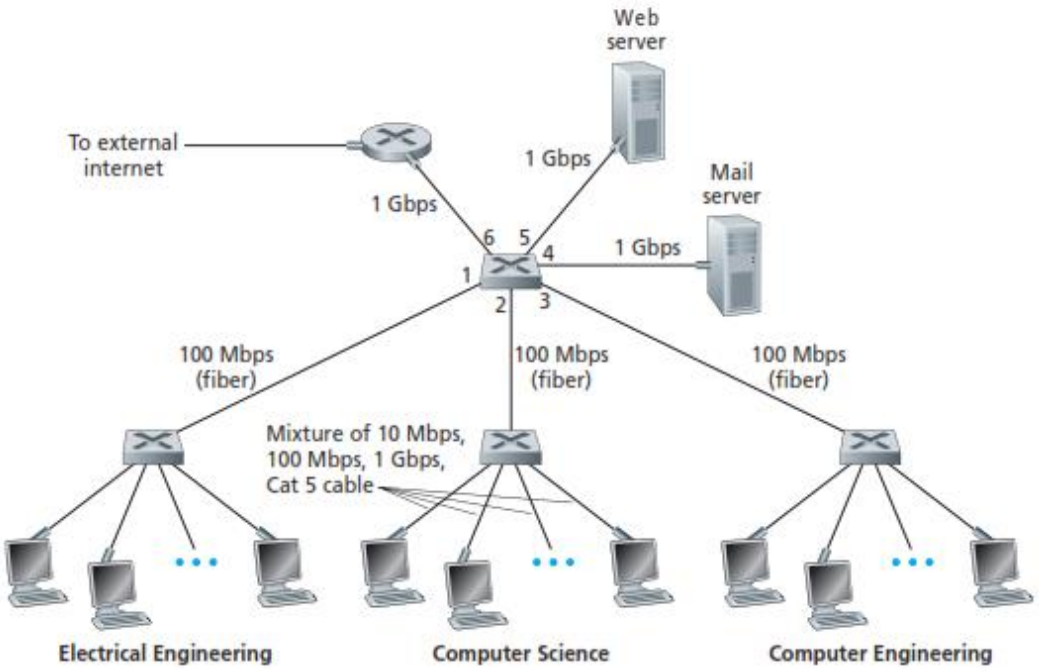

Figure 5.15 ♦ An institutional network connected together by four switches

Three drawbacks can be identified in the configuration in Figure 5.15

What are the drawbacks of hierarchical Configuration of LANs?

1• Lack of traffic isolation.

Although the hierarchy localizes group traffic to within a single switch, broadcast must still traverse the entire institutional network. Limiting the scope of such broadcast traffic would improve LAN performance.

Perhaps more importantly, it also may be desirable to limit LAN broadcast **traffic for security/privacy** reasons.

2• Inefficient use of switches.

If instead of three groups, the institution had 10 groups, then 10 first-level switches would be required. If each group were small, say less than 10 people, then a single 96-port switch

would likely be large enough to accommodate everyone, **but this single switch would not provide traffic isolation.**

• Managing users Problems.

If an employee moves between groups, **the physical cabling must be changed to connect the employee to a different switch** in Figure 5.15. Employees belonging to two groups make the problem even harder.

Fortunately, each of these difficulties can be handled by **a switch that supports virtual local area networks (VLANs).**

In a port-based VLAN, the **switch's ports (interfaces) are divided into groups** by the network manager. **Each group constitutes a VLAN**, with **the ports in each VLAN forming a broadcast domain** (i.e., broadcast traffic from one port can only reach other ports in the group).

Figure 5.25 shows a single switch with 16 ports. Ports 2 to 8 belong to the EE VLAN, while ports 9 to 15 belong to the CS VLAN (ports 1 and 16 are unassigned).

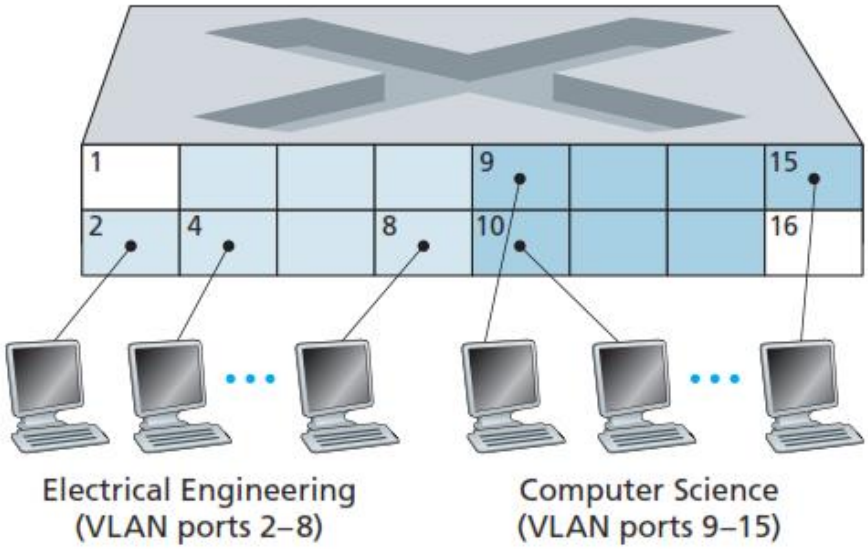

Figure 5.25 ♦ A single switch with two configured VLANs

This VLAN solves all of the difficulties noted above

- EE and CS VLAN frames are isolated from each other,
- the two switches in Figure 5.15 have been replaced by a single switch, and
- if the user at switch port 8 joins the CS **Department, the network operator simply reconfigures the VLAN software so that port 8 is now associated with the CS VLAN**.

But by completely isolating the two VLANs, we have introduced a new difficulty!

How can traffic from the EE Department be sent to the CS Department?

One way to handle this would be **to connect a VLAN switch port (e.g., port 1 in Figure 5.25) to an external router and configure that port to belong to both the EE and CS VLANs**.

In this case, even though the EE and CS departments share the same physical switch, **the logical configuration would look as if the EE and CS departments had separate switches connected via a router**.

An IP datagram going from the EE to the CS department would first cross the EE VLAN to reach the **router and then be forwarded by the router back over the CS VLAN to the CS host**.

A Better Solution is fortunately, switch vendors make such configurations easy for the network manager by building a single device that contains both a VLAN switch *and* a router, so a separate external router is not needed.

The Question now is: How to connect EE and CS staff that is housed in a separate building to the same VLAN?

One Solution is: to use a second switch, where the switch ports have been defined as belonging to the EE or the CS VLAN, as needed. The two switches are interconnected by **defining a port belonging to the CS VLAN on each switch (similarly for the EE VLAN), and to connect these ports to each other, as shown in Figure 5.26(a).**

This solution ***doesn't scale***, however, since *N* VLANS would require *N* ports on each switch simply to interconnect the two switches.

A more scalable approach to interconnecting VLAN switches are known as **VLAN Trunking**. In the **VLAN Trunking** approach shown in Figure 5.26(b), **a special port on each switch (port 16 on the left switch and port 1 on the right switch) is configured as a trunk port to interconnect the two VLAN switches**. The trunk port belongs to all VLANs, and frames sent to any VLAN are forwarded over the trunk link to the other switch.

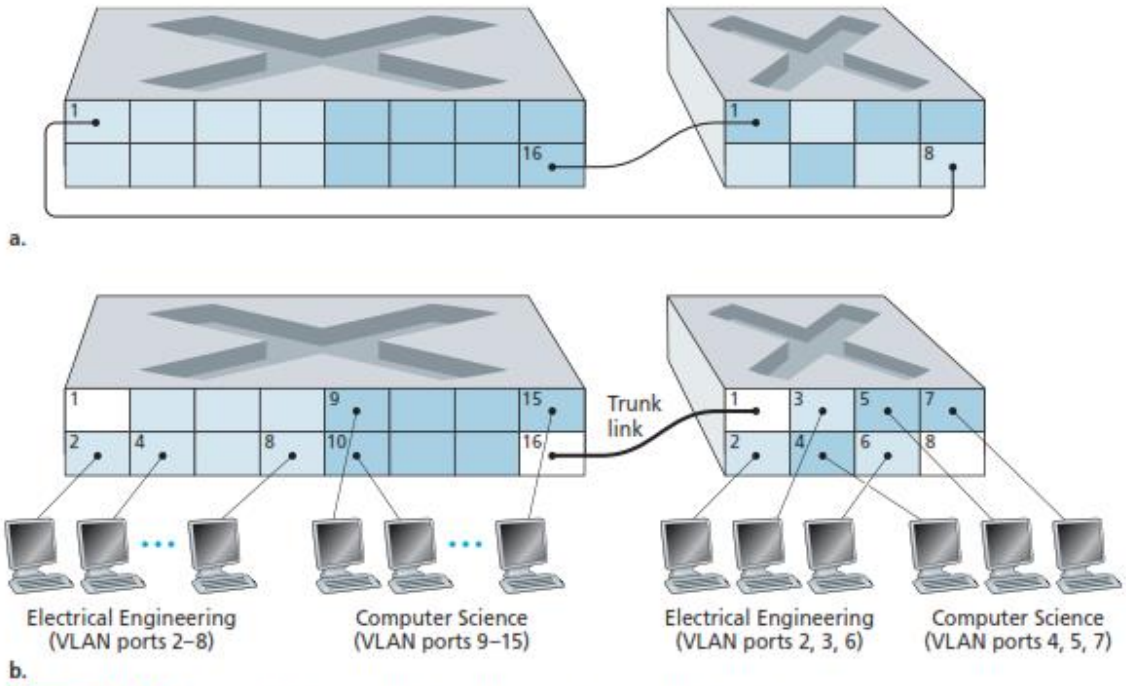

Figure 5.26 ♦ Connecting two VLAN switches with two VLANs: (a) two cables (b) trunked