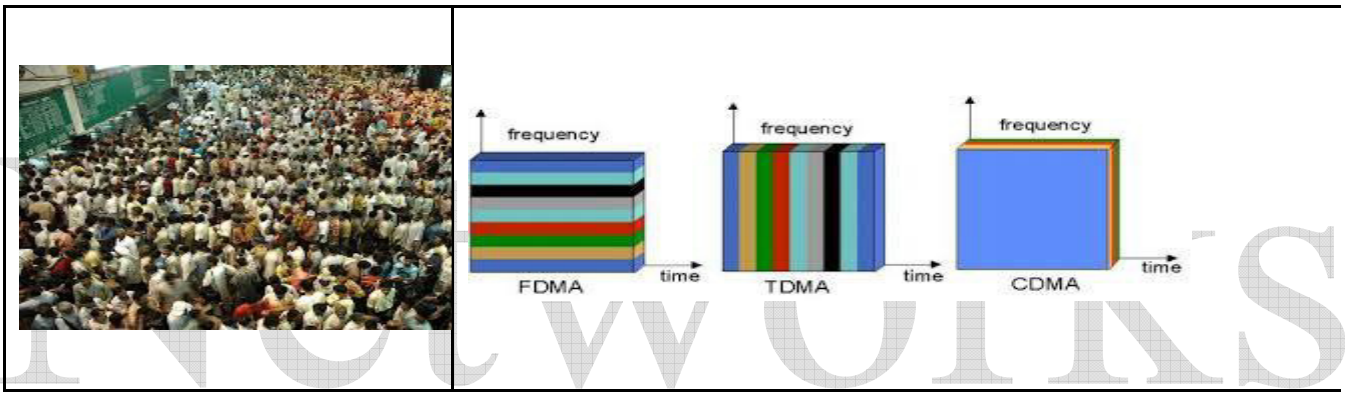
UNIT II
MEDIA ACCESS & INTERNETWORKING

Medium access control:

In telecommunications and computer networks, a **channel access method** or **multiple access method** allows several terminals connected to the same multi-point transmission medium to transmit over it and to share its capacity. Examples of shared physical media are wireless networks, bus networks, ring networks and half-duplex point-to-point links.

A channel-access scheme is based on a multiplexing method, that allows several data streams or signals to share the same communication channel or physical medium. Multiplexing is in this context provided by the physical layer. Note that multiplexing also may be used in full-duplex point-to-point communication between nodes in a switched network, which should not be considered as multiple accesses.



A channel-access scheme is also based on a multiple access protocol and control mechanism, also known as media access control (MAC). This protocol deals with issues such as addressing, assigning multiplex channels to different users, and avoiding collisions. The MAC-layer is a sub-layer in Layer 2 (Data Link Layer) of the OSI model and a component of the Link Layer of the TCP/IP model.

Frequency Division Multiple Access (FDMA)

The frequency-division multiple access (FDMA) channel-access scheme is based on the frequency-division multiplexing (FDM) scheme, which provides different frequency bands to different data-streams. In the FDMA case, the data streams are allocated to different nodes or devices. An example of FDMA systems were the first-generation (1G) cell-phone systems, where each phone call was assigned to a specific uplink frequency channel, and another downlink frequency channel. Each message signal (each phone call) is modulated on a specific carrier frequency.

A related technique is wavelength division multiple access (WDMA), based on wavelength-division multiplexing (WDM), where different datastreams get different colors in fiber-optical communications. In the WDMA case, different network nodes in a bus or hub network get a different color.

An advanced form of FDMA is the orthogonal frequency-division multiple access (OFDMA) scheme, for example used in 4G cellular communication systems. In OFDMA, each node may use several sub-carriers, making it possible to provide different quality of service (different data rates) to different users. The assignment of sub-carriers to users may be changed dynamically, based on the current radio channel conditions and traffic load.

Time division multiple access (TDMA)

The time division multiple access (TDMA) channel access scheme is based on the time-division multiplexing (TDM) scheme, which provides different time-slots to different data-streams (in the TDMA case to different transmitters) in a cyclically repetitive frame structure. For example, node 1 may use time slot 1, node 2 time slot 2, etc. until the last transmitter. Then it starts all over again, in a repetitive pattern, until a connection is ended and that slot becomes free or assigned to another node. An advanced form is Dynamic TDMA (DTDMA), where a scheduling may give different time sometimes but some times node 1 may use time slot 1 in first frame and use another time slot in next frame.

As an example, 2G cellular systems are based on a combination of TDMA and FDMA. Each frequency channel is divided into eight timeslots, of which seven are used for seven phone calls, and one for signaling data.

Code division multiple access (CDMA)/Spread spectrum multiple access (SSMA)

The code division multiple access (CDMA) scheme is based on spread spectrum, meaning that a wider radio spectrum in Hertz is used than the data rate of each of the transferred bit streams, and several message signals are transferred simultaneously over the same carrier frequency, utilizing different spreading codes.

The wide bandwidth makes it possible to send with a very poor signal-to-noise ratio of much less than 1 (less than 0 dB) according to the Shannon-Heartly formula, meaning that the transmission power can be reduced to a level below the level of the noise and co-channel interference (cross talk) from other message signals sharing the same frequency.

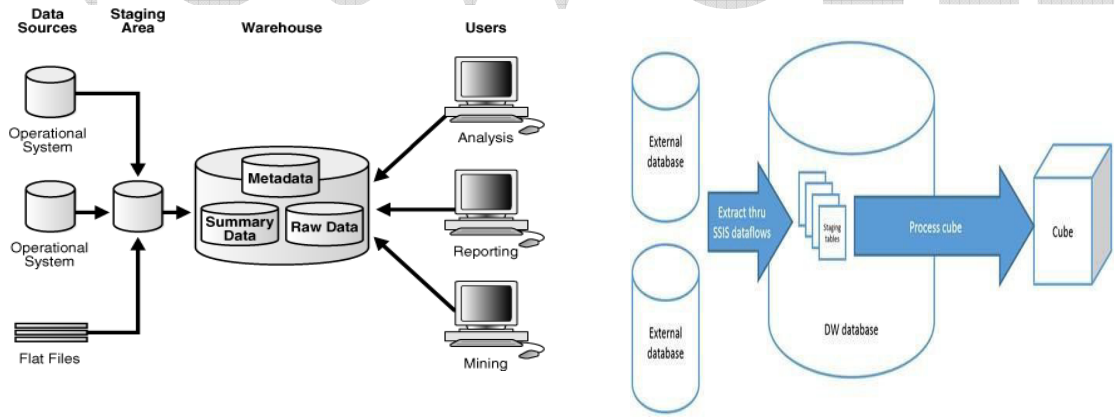
One form is direct sequence spread spectrum (DS-CDMA), used for example in 3G cell phone systems. Each information bit (or each symbol) is represented by a long code sequence of several pulses, called chips. The sequence is the spreading code, and each message signal (for example each phone call) use different spreading code.

Another form is frequency-hopping (FH-CDMA), where the channel frequency is changing very rapidly according to a sequence that constitutes the spreading code. As an example, the Bluetooth communication system is based on a combination of frequency-hopping and either CSMA/CA packet mode communication (for data communication applications) or TDMA (for audio transmission). All nodes belonging to the same user (to the same virtual private area network or picante) use the same frequency hopping sequences synchronously, meaning that they send on the same frequency channel, but CDMA/CA or TDMA is used to avoid collisions within the VPAN. Frequency-hopping is used to reduce the cross-talk and collision probability between nodes in different VPAN's.

Ethernet:

Ethernet is a family of computer networking technologies for local area networks (LANs). Ethernet was commercially introduced in 1980 and standardized in 1983 as IEEE 802.3.[1] Ethernet has largely replaced competing wired LAN technologies such as token ring, FDDI, and ARCNET.

The Ethernet standards comprise several wiring and signaling variants of the OSI physical layer in use with Ethernet. The original 10BASE5 Ethernet used coaxial cable as a shared medium. Later the coaxial cables were replaced with twisted pair and fiber optic links in conjunction with hubs or switches. Data rates were periodically increased from the original 10 megabits per second to 100 gigabits per second.



Systems communicating over Ethernet divide a stream of data into shorter pieces called frames. Each frame contains source and destination addresses and error-checking data so that damaged data can be detected and re-transmitted. As per the OSI model, Ethernet provides services up to and including the data link layer.

Since its commercial release, Ethernet has retained a good degree of compatibility. Features such as the 48-bit MAC address and Ethernet frame format have influenced other networking protocols.

Developed in the mid-1970s by researchers at the Xerox Palo Alto Research Center (PARC), the Ethernet eventually became the dominant local area networking technology, emerging from a pack of competing technologies. Today, it competes mainly with 802.11 wireless networks but remains extremely popular in campus networks and data centers. The more general name for the technology behind the Ethernet is Carrier Sense, Multiple Access with Collision Detect (CSMA/CD). As indicated by the CSMA name, the Ethernet is a multiple-access network, meaning that a set of nodes sends and receives frames over a shared link. You can, therefore, think of an Ethernet as being like a bus that has multiple stations plugged into it. The "carrier sense" in CSMA/CD means that all the nodes can distinguish between an idle and a busy link, and "collision detect" means that a node listens as it transmits and can therefore detect when a frame it is transmitting has interfered (collided) with a frame transmitted by another node.

The Ethernet has its roots in an early packet radio network, called Aloha, developed at the University of Hawaii to support computer communication across the Hawaiian Islands. Like the Aloha network, the fundamental problem faced by the Ethernet is how to mediate access to a shared medium fairly and efficiently (in Aloha, the medium was the atmosphere, while in the Ethernet the medium is a coax cable). The core idea in both Aloha and the Ethernet is an algorithm that controls when each node can transmit. Interestingly, modern Ethernet links are now largely point to point; that is, they connect one host to an Ethernet switch, or they interconnect switches. Hence, "multiple access" techniques are not used much in today's Ethernets. At the same time, wireless networks have become enormously popular, so the multiple access technologies that started in Aloha are today again mostly used in wireless networks such as 802.11 (Wi-Fi) networks.

We will discuss Ethernet switches in the next chapter. For now, we'll focus on how a single Ethernet link works. And even though multi-access Ethernet is becoming a bit of a historical curiosity, the principles of multi-access networks continue to be important enough to warrant some further discussion, which we provide below. Digital Equipment Corporation and Intel Corporation joined Xerox to define a 10-Mbps Ethernet standard in 1978. This standard then formed the basis for IEEE standard 802.3, which additionally defines a much wider collection of physical media over which an Ethernet can operate, including 100-Mbps, 1-Gbps, and 10-Gbps versions.

Physical Properties:

Ethernet segments were originally implemented using coaxial cable of length up to 500 m. (Modern Ethernets use twisted copper pairs, usually a particular type known as "Category 5," or optical fibers, and in some cases can be quite a lot longer than 500 m.) This cable was similar to the type used for cable TV. Hosts connected to an Ethernet segment by tapping into it. A *transceiver*, a small device directly attached to the tap, detected when the line was idle and drove the signal when

the host was transmitting. It also received incoming signals. The transceiver, in turn, connected to an Ethernet adaptor, which was plugged into the host.

Multiple Ethernet segments can be joined together by *repeaters*. A repeater is a device that forwards digital signals, much like an amplifier Transceiver Ethernet cable Adaptor Host.

Wireless LAN's:

Wireless technologies differ from wired links in some important ways, while at the same time sharing many common properties. Like wired links, issues of bit errors are of great concern—typically even more so due to the predictable noise environment of most wireless links. Framing and reliability also have to be addressed. Unlike wired links, power is a big issue for wireless, especially because wireless links are often used by small mobile devices (like phones and sensors) that have limited access to power (e.g., a small battery). Furthermore, you can't go blasting away at arbitrarily high power with a radio transmitter—there are concerns about interference with other devices and usually regulations about how much power a device may emit at any given frequency. Wireless media are also inherently multi-access; it's difficult to direct your radio transmission to just a single receiver or to avoid receiving radio signals from any transmitter with enough power in your neighborhood. Hence, media access control is a central issue for wireless links. And, because it's hard to control who receives your signal when you transmit over the air, issues of eavesdropping may also have to be addressed.

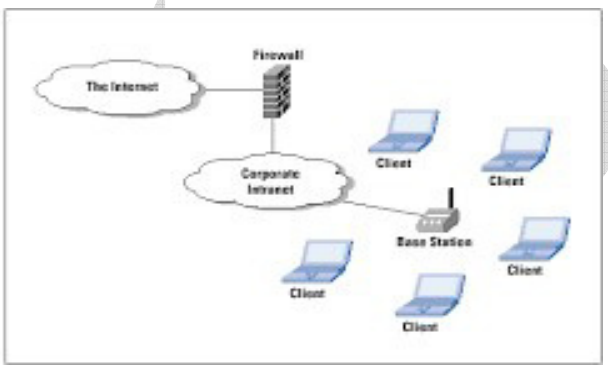


Figure 1: Typical IEEE 802.11 Configuration in Infrastructure Mode

Wi-Fi (more formally known as 802.11), Bluetooth, and the third-generation or "3G" family of cellular wireless standards. Frequency band in hertz and sometimes the data rate of a link. Because both these concepts come up in discussions of wireless networks, we're going to use *bandwidth* here in its stricter sense—width of a frequency band—and use the term *data rate* to describe the number of bits per second that can be sent over the link. Because wireless links all share the same medium, the challenge is to share that medium efficiently, without unduly interfering with each other. Most of this sharing is accomplished by dividing it up along the dimensions of frequency and space. Exclusive use of a particular frequency in a particular geographic area may be allocated to an individual entity such as a corporation. It is feasible to limit the area covered by an electromagnetic signal because such signals weaken, or *attenuate*, with the distance from their origin. To reduce the area covered by your signal, reduce the power of your transmitter. These allocations are typically determined by government agencies, such as the Federal Communications Commission (FCC) in the United States. Specific bands (frequency ranges) are allocated to certain uses. Some bands are reserved for

government use. Other bands are reserved for uses such as AM radio, FM radio, television, satellite communication, and cellular phones.

A second spread spectrum technique, called *direct sequence*, adds redundancy for greater tolerance of interference. Each bit of data is represented by multiple bits in the transmitted signal so that, if some of the transmitted bits are damaged by interference, there is usually enough redundancy to recover the original bit. For each bit the sender wants to transmit, it actually sends the exclusive-OR of that bit and n random bits. As with frequency hopping, the sequence of random bits is generated by a pseudorandom number generator known to both the sender and the receiver. The transmitted values, known as an n-bit *chipping code*, spread the signal across a frequency band that is n times wider than the frame would have otherwise required.

802.11/Wi-Fi

Most readers will have used a wireless network based on the IEEE 802.11standards, often referred to as *Wi-Fi*.9 Wi-Fi is technically a trademark, owned by a trade group called the Wi-Fi Alliance, which certifies product compliance with 802.11. Like Ethernet, 802.11 is designed for use in a limited geographical area (homes, office buildings, campuses), and its primary challenge is to mediate access to a shared communication medium—in this case, signals propagating through space. Physical Properties 802.11 defines a number of different physical layers that operate in various frequency bands and provide a range of different data rates. At the time of writing, 802.11n provides the highest maximum data rate, topping out at 600 Mbps. The original 802.11 standard defined two radio-based physical layers standards, one using frequency hopping (over 79 1-MHz-wide frequency bandwidths) and the other using direct sequence spread spectrum (with an 11-bit chipping sequence). Both provided data rates in the 2 Mbps range. The physical layer standard 802.11b was added subsequently. Using a variant of direct sequence, 802.11b provides up to 11 Mbps. These three standards all operated in the license-exempt 2.4-GHz frequency band of the electromagnetic spectrum. Then came 802.11a, which delivers up to 54 Mbps using a variant of FDM called *orthogonal frequency division multiplexing (OFDM)*; 802.11a runs in the license-exempt 5-GHz band. On one hand, this band is less used, so there is less interference. On the other hand, there is more absorption of the signal and it is limited to almost line of sight. 802.11g followed; 802.11g also uses OFDM, delivers up to 54 Mbps, and is backward compatible with 802.11b (and returns to the 2.4-GHz band). Most recently 802.11n has appeared on the scene, with a standard that was approved in 2009 (although pre-standard products also existed). 802.11n achieves considerable advances in maximum possible data rate using multiple antennas and allowing greater wireless channel bandwidths. The use of multiple antennas is often called *MIMO* for multiple input, multiple-output.

The systems try to pick an optimal bit rate based on the noise environment in which they find themselves; the algorithms for bit rate selection can be quite complex (see the Further Reading section for an example). Interestingly, the 802.11 standards do not specify a particular approach but leave the algorithms to the various vendors. The basic approach to picking a bit rate is to estimate the bit error rate either by directly measuring the signal-to-noise ratio (SNR) at the physical layer or by estimating the SNR by measuring how often packets are successfully transmitted and acknowledged. In some approaches, a sender will occasionally probe a higher bit rate by sending one or more packets at that rate to see if it succeeds.

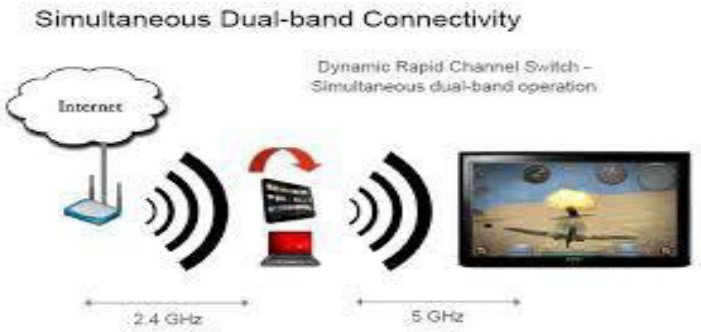
Distribution System:

As described so far, 802.11 would be suitable for a network with a mesh (*ad hoc*) topology, and development of a 802.11s standard for mesh networks is nearing completion. At the current time, however, nearly all 802.11 networks use a base-station-oriented topology. Instead of all nodes being created equal, some nodes are allowed to roam (e.g., your laptop) and some are connected to a wired network infrastructure. 802.11 calls these base stations *access points* (APs), and they are connected to each other by a so-called *distribution system*.

A distribution system that connects three access points, each of which services the nodes in some region. Each access point operates on some channel in the appropriate frequency range, and each AP will typically be on a different channel than its neighbors. The details of the distribution system are not important to this discussion—it could be an Ethernet, for example. The only important point is that the distribution network operates at the link layer, the same protocol layer as the wireless links. In other words, it does not depend on any higher-level protocols (such as the network layer).

Bluetooth:

Bluetooth fills the niche of very short range communication between mobile phones, PDAs, notebook computers, and other personal or peripheral devices. For example, Bluetooth can be used to connect a mobile phone to a headset or a notebook computer to a keyboard. Roughly speaking, Bluetooth is a more convenient alternative to connecting two devices with a wire. In such applications, it is not necessary to provide much range or bandwidth. This means that Bluetooth radios can use quite low power transmission, since transmission power is one of the main factors affecting bandwidth and range of wireless links. This matches the target applications for Bluetooth-enabled devices—most of them are battery powered (such as the ubiquitous phone headset) and hence it is important that they not consume much power.

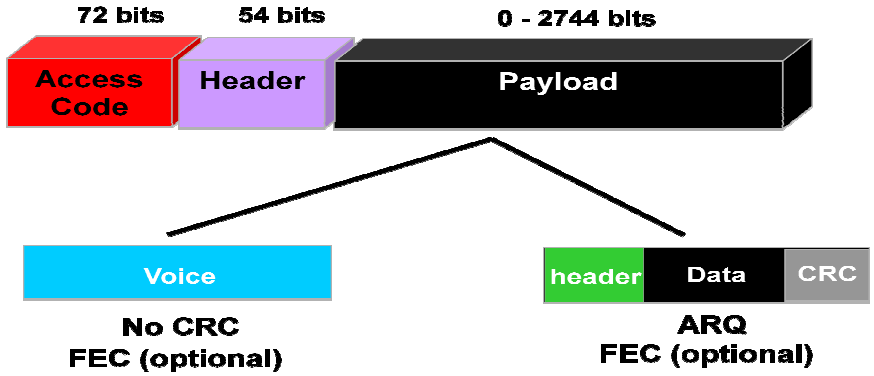


Bluetooth operates in the license-exempt band at 2.45 GHz. Bluetooth links have typical bandwidths around 1 to 3 Mbps and a range of about 10 m. For this reason, and because the communicating devices typically belong to one individual or group, Bluetooth is sometimes categorized as a Personal Area Network (PAN).

Bluetooth is specified by an industry consortium called the *Bluetooth Special Interest Group*. It specifies an entire suite of protocols, going beyond the link layer to define application protocols, which it calls *profiles*, for a range of applications. For example, there is a profile for synchronizing a PDA with a personal computer. Another profile gives a mobile computer access to a wired LAN in the manner of 802.11, although this was not Bluetooth's original goal. The IEEE 802.15.1 standard is based on Bluetooth but excludes the application protocols.

The basic Bluetooth network configuration, called a *picante*, consists of a master device and up to seven slave devices. Any communication is between the master and a slave; the slaves do not communicate directly with each other. Because slaves have a simpler role, their Bluetooth hardware and software can be simpler and cheaper. Since Bluetooth operates in an license-exempt band, it is required to use a spread spectrum technique. To deal with possible interference in the band. It uses frequency-hopping with 79 *channels* (frequencies), using each for 625 μs at a time. This provides a natural time slot for Bluetooth to use for synchronous time division multiplexing. A frame takes up 1, 3, or 5 consecutive time slots. Only the master can start to transmit in odd-numbered slots. A slave can start to transmit in an even-numbered slot—but only in response to a request from the master during the previous slot, thereby preventing any contention between the slave devices. A slave device can be *parked*; that is, it is set to an inactive, low-power state. A parked device cannot communicate on the piconte; it can only be reactivated by the master. A picante can have up to 255 parked devices in addition to its active slave devices.

Packet structure



Classification

Typical Bluetooth Scenario

Bluetooth will support wireless point-to-point and point-to-multipoint (broadcast) between devices in a piconet.

Point to Point Link

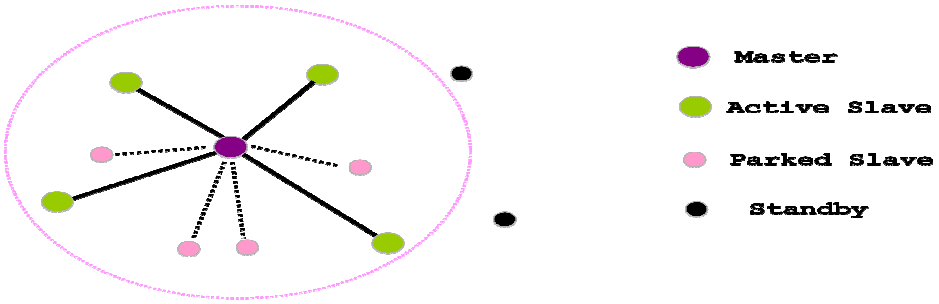
Master - slave relationship

Bluetooth devices can function as masters or slaves

Piconet

It is the network formed by a Master and one or more slaves (max 7)
Each piconet is defined by a different hopping channel to which users synchronize to
Each piconet has max capacity (1 Mbps)

Piconet Structure



- **All devices in piconet hop together.**
- **Master's ID and master's clock determines frequency hopping sequence & phase.**

Wireless peripherals:

- Headsets
- Cameras

Security

Security Measures

Link Level Encryption & Authentication.
Personal Identification Numbers (PIN) for device access.
Long encryption keys are used (128 bit keys).
These keys are not transmitted over wireless. Other parameters are transmitted over wireless which in combination with certain information known to the device, can generate the keys.
Further encryption can be done at the application layer.

Future of Bluetooth

Success of Bluetooth depends on how well it is integrated into consumer products

Consumers are more interested in applications than the technology
Bluetooth must be successfully integrated into consumer products
Must provide benefits for consumer
Must not destroy current product benefits

Switching And Bridging:

In the simplest terms, a switch is a mechanism that allows us to interconnect links to form a larger network. A switch is a multi-input, multi-output device that transfers packets from an input to one or more outputs. Thus, a switch adds the star topology to the point-to-point link, bus (Ethernet), and ring topologies established in the last chapter.

A star topology has several attractive properties: Even though a switch has a fixed number of inputs and outputs, which limits the number of hosts that can be connected to a single switch, large networks can be built by interconnecting a number of switches.

We can connect switches to each other and to hosts using point-to-point links, which

typically means that we can build networks of large geographic scope. n Adding a new host to the network by connecting it to a switch does not necessarily reduce the performance of the network for other hosts already connected. This last claim cannot be made for the shared-media networks.

Internetworking

- Switching and Bridging
- Basic Internetworking (IP)

- Switch
  - A mechanism that allows us to interconnect links to form a large network
  - A multi-input, multi-output device which transfers packets from an input to one or more outputs

A switch is connected to a set of links and for each of these links, runs the appropriate data link protocol to communicate with that node
A switch's primary job is to receive incoming packets on one of its links and to transmit them on some other link
- This function is referred as *switching or forwarding*
- According to OSI architecture this is the main function of the network layer

Switching and Forwarding

Characteristics of Connectionless (Datagram) Network

- A host can send a packet anywhere at any time, since any packet that turns up at the switch can be immediately forwarded using the **forwarding table**
- When a host sends a packet, **it does NOT know** if the network is capable of delivering it or if the destination host is even up and running
- Each packet is **forwarded independently** of previous packets that might have been sent to the same destination.
  - Thus two successive packets from host A to host B may follow completely different paths
- A switch or link failure **might not have any serious effect** on communication if it is possible to find an alternate route around the failure and update the forwarding table accordingly

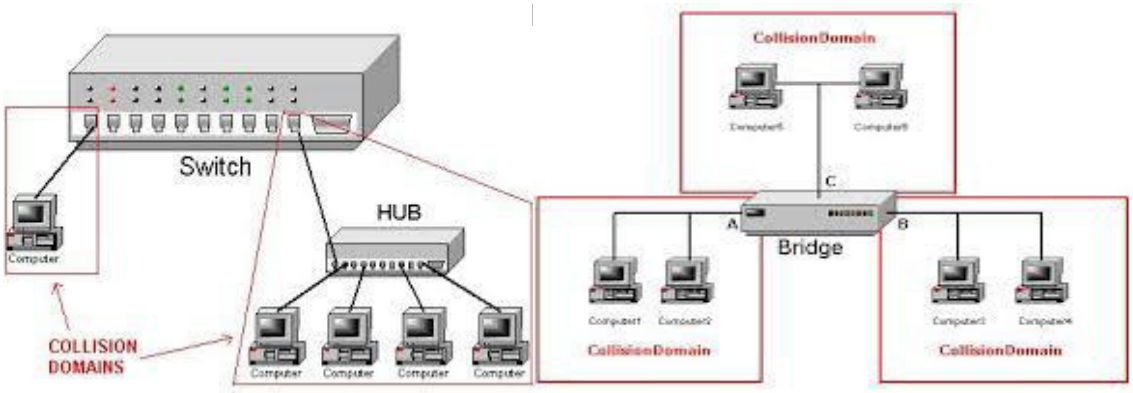
Bridges and LAN Switches

Bridges and LAN Switches

Class of switches that is used to forward packets between shared-media LANs such as Ethernets

Known as LAN switches, Referred to as Bridges Suppose you have a pair of Ethernets that you want to interconnect.One approach is put a repeater in between them It might exceed the physical limitation of the Ethernet No more than four repeaters between any pair of hosts No more than a total of 2500 m in length is allowed
An alternative would be to put a node between the two Ethernets and have the node forward frames from one Ethernet to the other This node is called a **Bridge** A collection of LANs connected by one or more bridges is usually said to form an **Extended LAN**

.



For example, it is impossible for two hosts on the same 10-Mbps Ethernet segment to transmit continuously at 10 Mbps because they share the same transmission medium. Every host on a switched network has its own link to the switch, so it may be entirely possible for many hosts to transmit at the full link speed (bandwidth), provided that the switch is designed with enough aggregate capacity. Providing high aggregate throughput is one of the design goals for a switch; we return to this topic later. In general, switched networks are considered more *scalable* (i.e., more capable of growing to large numbers of nodes) than shared-media networks because of this ability to support many hosts at full speed. A switch is connected to a set of links and, for each of these links, runs the appropriate data link protocol to communicate with the node at the other end of the link. A switch's primary job is to receive incoming packets on one of its links and to transmit them on some other link.

This function is sometimes referred to as either *switching* or *forwarding,* and in terms of the Open Systems Interconnection (OSI) architecture, it is the main function of the network layer. The first is the *datagram* or *connectionless* approach. The second is the *virtual circuit* or *connection-oriented* approach. A third approach, *source routing*, is less common than these other two, but it does have some useful applications. One thing that is common to all networks is that we need to have a way to identify the end nodes. Such identifiers are usually called *addresses*.

We have already seen examples of addresses in the previous chapter, such as the 48-bit address used for Ethernet. The only requirement for Ethernet addresses is that no two nodes on a network have the same address. This is accomplished by making sure that all Ethernet cards are assigned a *globally unique* identifier. For the following discussions, we assume that each host has a globally unique address. Later on, we consider other useful properties that an address might have, but global uniqueness is adequate to get us started. Another assumption that we need to make is that there is some way to identify the input and output ports of each switch. There are at least two sensible ways to identify ports: One is to number each port, and the other is to identify the port by the name of the

node (switch or host) to which it leads. For now, we use numbering of the ports.

Datagrams:

The idea behind datagrams is incredibly simple: You just include in every packet enough information to enable any switch to decide how to get it to its destination. That is, every packet contains the complete destination address. Consider the example network in which the hosts have addresses A, B, C, and so on. To decide how to forward a packet, a switch consults a *forwarding table* (sometimes called a *routing table*). This particular table shows the forwarding information that switch 2 needs to forward datagrams in the example network. It is pretty easy to figure out such a table when you have a complete map of a simple network like that depicted here; we could imagine a network operator configuring Datagram networks have the following characteristics: n A host can send a packet anywhere at any time, since any packet that turns up at a switch can be immediately forwarded (assuming a correctly populated forwarding table). For this reason, datagram networks are often called *connectionless*; this contrasts with the *connection-oriented* networks described below, in which some *connection state* needs to be established before the first data packet is sent. n When a host sends a packet, it has no way of knowing if the network is capable of delivering it or if the destination host is even up and running. n Each packet is forwarded independently of previous packets that might have been sent to the same destination. Thus, two successive packets from host A to host B may follow completely different paths (perhaps because of a change in the forwarding table at some switch in the network).

A switch or link failure might not have any serious effect on communication if it is possible to find an alternate route around the failure and to update the forwarding table accordingly. This last fact is particularly important to the history of datagram networks. One of the important design goals of the Internet is robustness to failures.

Virtual Circuit Switching:

A second technique for packet switching, which differs significantly from the datagram model, uses the concept of a *virtual circuit* (VC). A *virtual circuit identifier* (VCI) hat uniquely identifies the connection at this switch and which will be carried inside the header of the packets that belong to this connection n An incoming interface on which packets for this VC arrive at the switch n An outgoing interface in which packets for this VC leave the switch. A potentially different VCI that will be used for outgoing packets The semantics of one such entry is as follows: If a packet arrives on the designated incoming interface and that packet contains the designated VCI value in its header, then that packet should be sent out the specified outgoing interface with the specified outgoing VCI value having been first placed in its header.

Basic Internetworking (IP, CIDR, ARP, DHCP,ICMP):

BASIC INTERNETWORKING (IP):

In the previous section, we saw that it was possible to build reasonably large LANs using bridges and LAN switches, but that such approaches were limited in their ability to scale and to handle heterogeneity. In this section, we explore some ways to go beyond the limitations of bridged networks, enabling us to build large, highly heterogeneous networks with reasonably efficient routing. We refer to such networks as *internetworks.* We'll continue the discussion of

how to build a truly global internetworking the next chapter, but for now we'll explore the basics. We start by considering more carefully what the word *internetwork* means.

What Is an Internetwork?

We use the term *internetwork*, or sometimes just *internet* with a lowercase *i*, to refer to an arbitrary collection of networks interconnected to provide some sort of host-to-host packet delivery service. For example, a corporation with many sites might construct a private internetwork by interconnecting the LANs at their different sites with point-to-point links leased from the phone company. When we are talking about the widely used global internetwork to which a large percentage of networks are now connected, we call it the *Internet* with a capital *I*. In keeping with the first principles approach of this book, we mainly want you to learn about the principles of "lowercase *i*" internetworking, but we illustrate these ideas with real-world examples from the "big *I*" Internet. Another piece of terminology that can be confusing is the difference between networks, sub-networks, and internetworks. We are going to avoid sub-networks (or subnets) altogether. For now, we use *network* to mean either a directly connected or a switched network.

An *inter-network* is an interconnected collection of such networks. Sometimes, to avoid ambiguity, we refer to the underlying networks that we are interconnecting as *physical* networks. An internet is a *logical* network built out of a collection of physical networks. In this context, a collection of Ethernets connected by bridges or switches would still be viewed as a single network. An internetwork is often referred to as a "network of networks" because it is made up of lots of smaller networks. In this figure, we see Ethernets, a wireless network, The *Internet Protocol* is the key tool used today to build scalable, heterogeneous internetworks. It was originally known as the Kahn-Cerf protocol after its inventors.6 One way to think of IP is that it runs on all the nodes (both hosts and routers) in a collection of networks and defines the infrastructure that allows these nodes and networks to function as a single logical internetwork. Such as TCP and UDP, typically run on top of IP on the hosts.

Service Model

A good place to start when you build an internetwork is to define its *service model*, that is, the

host-to-host services you want to provide. The main concern in defining a service model for an internetwork is that we can provide a host-to-host service only if this service can somehow be provided over each of the underlying physical networks. For example, it would be no good deciding that our internetwork service model was going to provide guaranteed delivery of every packet in 1 ms or less if there were underlying network technologies that could arbitrarily delay packets. The philosophy used in defining the IP service model, therefore, was to make it undemanding enough that just about any network technology that might turn up in an internetwork would be able to provide the necessary service. The IP service model can be thought of as having two parts: an addressing scheme, which provides a way to identify all hosts in the internetwork, and a datagram (connectionless) model of data delivery. This service model is sometimes called *best effort* because, although IP makes every effort to deliver datagrams, it makes no guarantees. We postpone a discussion of the addressing scheme for now and look first at the data delivery model.

Datagram Delivery

The IP datagram is fundamental to the Internet Protocol. A datagram is a type of packet that happens to be sent in a connectionless manner over a network. Every datagram carries enough information to let the network forward the packet to its correct destination; there is no need for any advance setup mechanism to tell the network what to do when the packet arrives. You just send it, and the network makes its best effort to get it to the desired destination. The "best-effort" part means that if something goes wrong and the packet gets lost, corrupted, miss delivered, or in any way fails to reach its intended destination, the network does nothing—it made its best effort, and that is all it has to do. It does not make any attempt to recover from the failure. This is sometimes called an *unreliable* service.

Clearly, a key part of the IP service model is the type of packets that can be carried. The IP datagram, like most packets, consists of a header followed by a number of bytes of data. A different style of representing packets than the one we used in previous chapters. This is because packet formats at the internetworking layer and above, where we will be focusing our attention.

CIDR:

CIDR, therefore, tries to balance the desire to minimize the number of routes that a router needs to know against the need to hand out addresses efficiently. To do this, CIDR helps us to *aggregate* routes. That is, it lets us use a single entry in a forwarding table to tell us how to reach a lot of different networks. As noted above it does this by breaking the rigid boundaries between address classes. To understand how this works, consider our hypothetical organization with 16 class C network numbers.

Instead of handing out 16 addresses at random, we can hand out a block of *contiguous* class C addresses. Suppose we assign the class C network numbers from 192.4.16 through 192.4.31. Observe that the top 20 bits of all the addresses in this range are the same (11000000 00000100 0001). Thus, what we have effectively created is a 20-bit network number—something that is between a class B network number and a class C number in terms of the number of hosts that it can support. In other words, we get both the high address efficiency of handing out addresses in chunks smaller than a class B network, and a single network prefix that can be used in forwarding tables. Observe that, for this scheme to work, we need to hand out blocks of class C addresses that share a

common prefix, which means that each block must contain a number of class C networks that is a power of two.

CIDR requires a new type of notation to represent network numbers, or *prefixes* as they are known, because the prefixes can be of any length. The convention is to place a /X after the prefix, where X is the prefix length in bits. So, for the example above, the 20-bit prefix for all the networks 192.4.16 through 192.4.31 is represented as 192.4.16/20. By contrast, if we wanted to represent a single class C network number, which is 24 bits long, we would write it 192.4.16/24. Today, with CIDR being the norm, it is more common to hear people talk about "slash 24" prefixes than class C networks. Note that representing a network address in this way is similar to the h mask, value i approach used in sub netting, as long as masks consist of contiguous bits starting from the most significant bit (which in practice is almost always the case).

Address Translation (ARP):

In the previous section we talked about how to get IP datagrams to the right physical network but glossed over the issue of how to get a datagram to a particular host or router on that network. The main issue is that IP datagrams contain IP addresses, but the physical interface hardware on the host or router to which you want to send the datagram only understands the addressing scheme of that particular network. Thus, we need to translate the IP address to a link-level address that makes sense on this network (e.g., a 48-bit Ethernet address). We can then encapsulate the IP datagram inside a frame that contains that link-level address and send it either to the ultimate destination or to a router that promises to forward the datagram toward the ultimate destination. One simple way to map an IP address into a physical network address is to encode a host's physical address in the host part of its IP address.

For example, a host with physical address 00100001 01001001 (which has the decimal value 33 in the upper byte and 81 in the lower byte) might be given the IP address 128.96.33.81. While this solution has been used on some networks, it is limited in that the network's physical addresses can be no more than 16 bits long in this example; they can be only 8 bits long on a class C network. This clearly will not work for 48-bit Ethernet addresses.

Host Configuration (DHCP):

Ethernet addresses are configured into the network adaptor by the manufacturer, and this process is managed in such a way to ensure that these addresses are globally unique. This is clearly a sufficient condition to ensure that any collection of hosts connected to a single Ethernet (including an extended LAN) will have unique addresses. Furthermore, uniqueness is all we ask of Ethernet addresses. IP addresses, by contrast, not only must be unique on a given internetwork but also must reflect the structure of the internetwork. As noted above, they contain a network part and a host part, and the network part must be the same for all hosts on the same network.

Thus, it is not possible for the IP address to be configured once into a host when it is manufactured, since that would imply that the manufacturer knew which hosts were going to end up on which networks, and it would mean that a host, once connected to one network, could never move to another. For this reason, IP addresses need to be reconfigurable. In addition to an IP address, there

are some other pieces of information a host needs to have before it can start sending packets. The most notable of these is the address of a default router—the place to which it can send packets whose destination address is not on the same network as the sending host.

Error Reporting (ICMP):

The next issue is how the Internet treats errors. While IP is perfectly willing to drop datagrams when the going gets tough—for example, when a router does not know how to forward the datagram or when one fragment of a datagram fails to arrive at the destination—it does not necessarily fail silently. IP is always configured with a companion protocol, known as the *Internet Control Message Protocol* (ICMP), that defines a collection of error messages that are sent back to the source host whenever a router or host is unable to process an IP datagram successfully.

For example, ICMP defines error messages indicating that the destination host is unreachable (perhaps due to a link failure), that the reassembly process failed, that the TTL had reached 0 and that the IP header checksum failed, and so on. ICMP also defines a handful of control messages that a router can send back to a source host. One of the most useful control messages, called an *ICMP-Redirect*, tells the source host that there is a better route to the destination.

ICMP-Redirects are used in the following situation. Suppose a host is connected to a network that has two routers attached to it, called *R1* and *R2*, where the host uses R1 as its default router. Should R1 ever receive a datagram from the host, where based on its forwarding table it knows that R2 would have been a better choice for a particular destination address, it sends an ICMP-Redirect back to the host, instructing it to use R2 for all future datagrams addressed to that destination. The host then adds this new route to its forwarding table.