15 January 2025 09:46

# Advanced Computer Networks:-

## The Internet: a nuts and bolts view

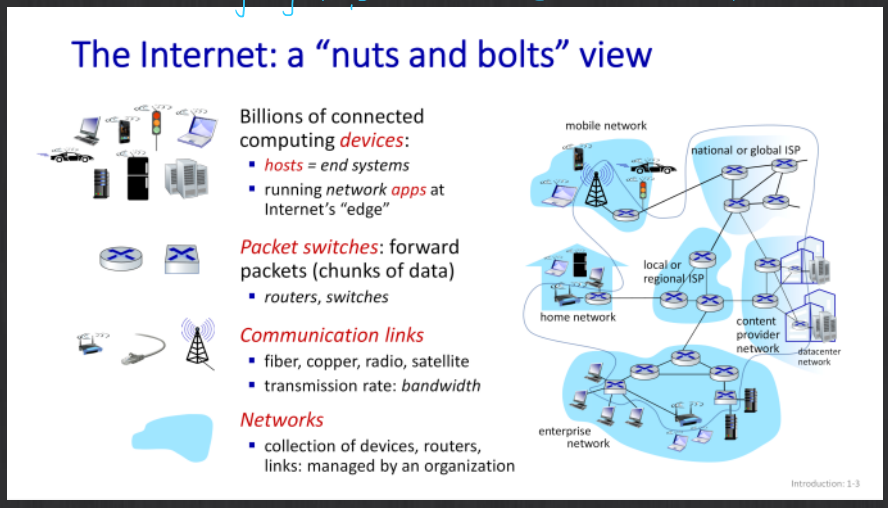
01. The Internet is a computer network that interconnects billions of computing devices throughout the world.

The Internet is a computer network that interconnects billions of computing devices throughout the world.
Not too long ago, these computing devices were primarily traditional desktop computers, Linux workstations, and so-called servers that store and transmit information such as Web pages and e-mail messages.

03. But however today users connect to the internet with smartphones and tablets.

Furthermore, nontraditional Internet "things" such as TVs, gaming consoles, thermostats, home security systems, home appliances, watches, eye glasses, cars, traffic control systems, and more are being connected to the Internet.

05. In Internet jargon, all the devices are called hosts or end systems



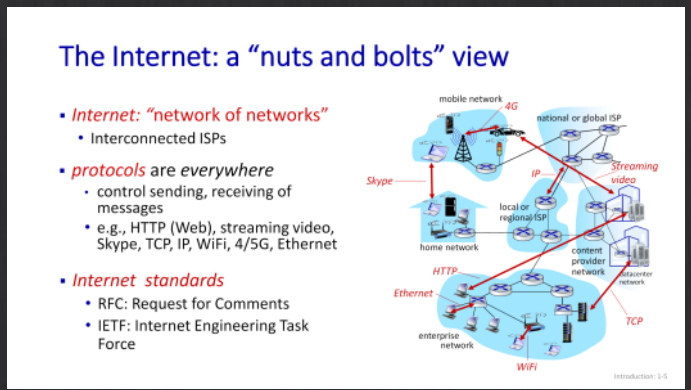

06. End systems are connected together by a n/w of communication link and packet switches

07. There are many types of communication links which are made up of different types of physical media including co-axial cable, copper wire, optical fibre and radio spectrum.

08. Different links can transmit data at different rates, with the transmission rate of a link measured in bits/sec

When one end system has data to send to another end system, the sending end system segments the data and adds header bytes to each segment.
The resulting packages of information, known as **packets** in the jargon of computer networks, are then sent through the network to the destination end system, where they are reassembled into the original data.

End systems access the Internet through **Internet Service Providers** (ISPs), including residential ISPs such as local cable or telephone companies; corporate ISPs; university ISPs; ISPs that provide WiFi access in airports, hotels, coffee shops, and other public places; and cellular data ISPs, providing mobile access to our smartphones and other devices.
Each ISP is in itself a network of packet switches and communication links. ISPs provide a variety of types of network access to the end systems, including residential broadband access such as cable modem or DSL, high-speed local area network access, and mobile wireless access. ISPs also provide Internet access to content providers, connecting servers directly to the Internet.

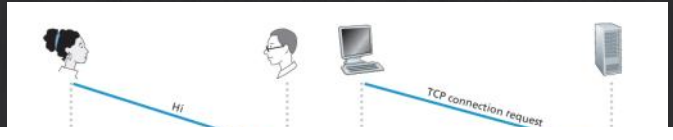



(Internet = Network of Networks)

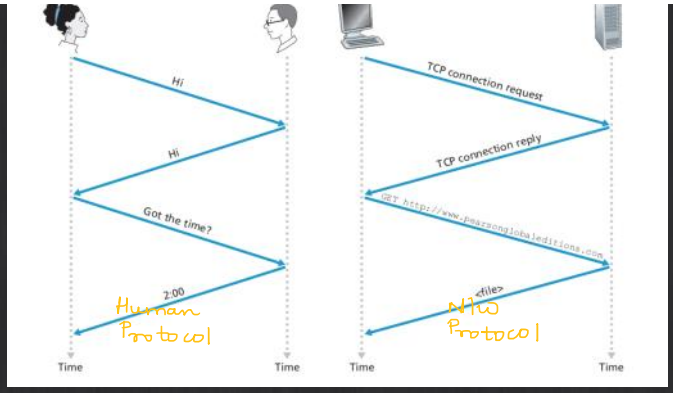
Given the importance of protocols to the Internet, it's important that everyone agree on what each and every protocol does, so that people can create systems and products that interoperate. This is where standards come into play. Internet standards are developed by the Internet Engineering Task Force (IETF) [IETF 2020]. The IETF standards documents are called requests for comments (RFCs). RFCs started out as general requests for comments (hence the name) to resolve network and protocol design problems that faced the precursor to the Internet [Allman 2011]. RFCs tend to be quite technical and detailed. They define protocols such as TCP, IP, HTTP (for the Web), and SMTP (for e-mail). There are currently nearly 9000 RFCs.

## Protocols ⇒

All the activity in the internet that involves two or more communicating devices is governed by protocols




A *protocol* defines the format and the order of messages exchanged between two or more communicating entities, as well as the actions taken on the transmission and/or receipt of a message or other event




Human Protocol

HTTP Protocol

> *A **protocol** defines the format and the order of messages exchanged between two or more communicating entities, as well as the actions taken on the transmission and/or receipt of a message or other event.*

## What's a protocol?

*Human protocols:*
- "what's the time?"
- "I have a question"
- introductions

Rules for:
... specific messages sent
... specific actions taken when message received, or other events

*Network protocols:*
- computers (devices) rather than humans
- all communication activity in Internet governed by protocols

*Protocols define the format, order of messages sent and received among network entities, and actions taken on message transmission, receipt*

# A closer look at internet Structure

Network Edge :-

## A closer look at Internet structure

Network edge:
- hosts: clients and servers
- servers often in data centers

Host = End systems.

Hosts
- Clients
- Servers

Clients = smartphones, desktops

data centre = large server

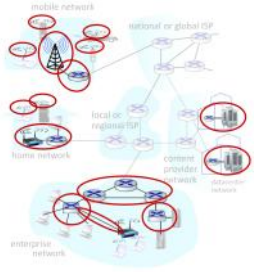
Servers = powerful machines that store and distribute web pages, stream video, relay e-mail and so on

Google → 19 data centres → 4 continents → containing millions of servers.

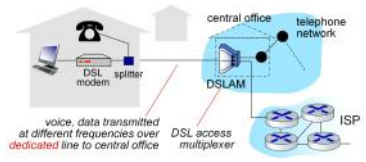
## 1.2.1 Access Networks

## Access networks and physical media

*Q: How to connect end systems to edge router?*
- residential access nets
- institutional access networks (school, company)
- mobile access networks (WiFi, 4G/5G)



## Access networks: digital subscriber line (DSL)




- use *existing* telephone line to central office DSLAM
  - data over DSL phone line goes to Internet
  - voice over DSL phone line goes to telephone net
- 24-52 Mbps dedicated downstream transmission rate
- 3.5-16 Mbps dedicated upstream transmission rate

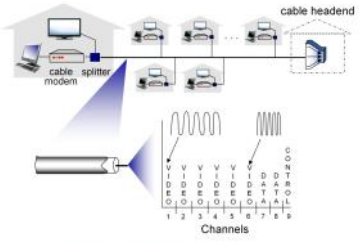
- Today, the two most prevalent types of broadband residential access are **digital subscriber line (DSL)** and **cable**.
- A residence typically obtains DSL Internet access from the same local telephone company (telco) that provides its wired local phone access. Thus, when DSL is used, a customer's telco is also its ISP.
- each customer's DSL modem uses the existing telephone line exchange data with a digital subscriber line access multiplexer (DSLAM) located in the telco's local central office (CO).
- The home's DSL modem takes digital data and translates it to high- frequency tones for transmission over telephone wires to the CO; the analog signals from many such houses are translated back into digital format at the DSLAM.

The residential telephone line carries both data and traditional telephone signals simultaneously, which are encoded at different frequencies:
- A high-speed downstream channel, in the 50 kHz to 1 MHz band
- A medium-speed upstream channel, in the 4 kHz to 50 kHz band
- An ordinary two-way telephone channel, in the 0 to 4 kHz band

- The DSL standards define multiple transmission rates, including downstream transmission rates of 24 Mbs and 52 Mbs, and upstream rates of 3.5 Mbps and 16 Mbps; the newest standard provides for aggregate upstream plus downstream rates of 1 Gbps [ITU 2014].
- Because the downstream and upstream rates are dif ferent, the access is said to be asymmetric.

## CABLE based

## Access networks: cable-based access




*frequency division multiplexing (FDM):* different channels transmitted in different frequency bands

- While DSL makes use of the telco's existing local telephone infrastructure, **cable Internet access** makes use of the cable television company's existing cable television infrastructure.
- A residence obtains cable Internet access from the same company that provides its cable television.

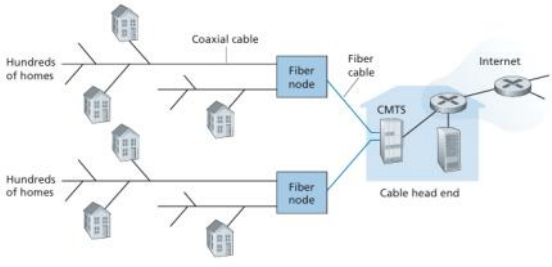



Figure 1.6 ♦ A hybrid fiber-coaxial access network

- **fiber optics** connect the cable head end to neighborhood-level junctions, from which tradi tional **coaxial cable** is then used to reach individual houses and apartments.
- Each neighborhood junction typically supports 500 to 5,000 homes.
- Because both **fiber** and **coaxial cable** are employed in this system, it is often referred to as hybrid fiber coax (HFC).

## Access in the Enterprise (and the Home): Ethernet and WiFi

On corporate and university campuses, and increasingly in home settings, a local area network (LAN) is used to connect an end system to the edge router. Although there are many types of LAN technologies, Ethernet is by far the most prevalent access technology in corporate, univ ersity, and home networks.

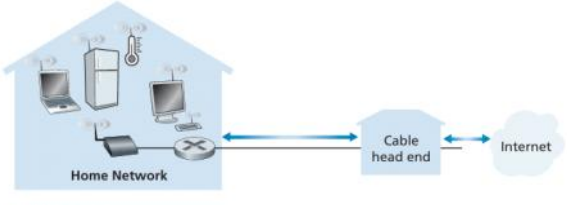



Figure 1.9 ♦ A typical home network

- Even though Ethernet and WiFi access networks were initially deployed in enterprise (corporate, university) settings, they ar e also common components of home networks.
- Many homes combine broadband residential access (that is, cable modems or DSL) with these inexpensive wireless LAN technologi es to create pow erful home networks Figure 1.9 shows a typical home network.
- This home network consists of a roaming laptop, multiple Internet-connected home appliances, as well as a wired PC; a base station (the wireless access

- This home network consists of a roaming laptop, multiple Internet-connected home appliances, as well as a wired PC; a base station (the wireless access point), which communicates with the wireless PC and other wireless devices in the home; and a home router that con nects the wireless access point, and any other wired home devices, to the Internet.
- This network allows household members to have broadband access to the Internet with one member roaming from the kitchen to th e backyard to the bedrooms.

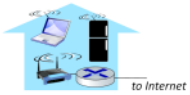
## Wireless access networks

Shared *wireless* access network connects end system to router

- via base station aka "access point"

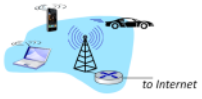
**Wireless local area networks** (WLANs)

- typically within or around building (~100 ft)
- 802.11b/g/n (WiFi): 11, 54, 450 Mbps transmission rate

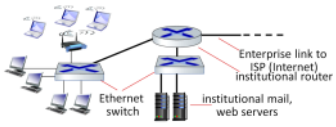



**Wide-area cellular access networks**

- provided by mobile, cellular network operator (10's km)
- 10's Mbps
- 4G/5G cellular networks




## Access networks: enterprise networks

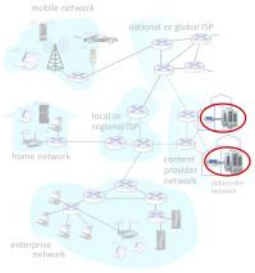



- companies, universities, etc.
- mix of wired, wireless link technologies, connecting a mix of switches and routers (we'll cover differences shortly)
  - Ethernet: wired access at 100Mbps, 1Gbps, 10Gbps
  - WiFi: wireless access points at 11, 54, 450 Mbps

## Access networks: data center networks

- high-bandwidth links (10s to 100s Gbps) connect hundreds to thousands of servers together, and to Internet

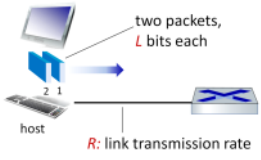
Courtesy: Massachusetts Green High Performance Computing Center (mghpcc.org)



## Host: sends *packets* of data

host sending function:

- takes application message
- breaks into smaller chunks, known as *packets*, of length *L* bits
- transmits packet into access network at *transmission rate R*
  - link transmission rate, aka link *capacity, aka link bandwidth*




$$\text{packet transmission delay} = \text{time needed to transmit } L\text{-bit packet into link} = \frac{L \text{ (bits)}}{R \text{ (bits/sec)}}$$

## 1.2.2 Physical Media

## Links: physical media

- bit: propagates between transmitter/receiver pairs
- physical link: what lies between transmitter & receiver
- guided media:
  - signals propagate in solid media: copper, fiber, coax
- unguided media:
  - signals propagate freely, e.g., radio

**Twisted pair (TP)**

- two insulated copper wires
  - Category 5: 100 Mbps, 1 Gbps Ethernet
  - Category 6: 10Gbps Ethernet

- we said that HFC uses a combination of fiber cable and coaxial cable.
- We said that DSL and Ethernet use copper wire. And we said that mobile access networks use the radio spectrum.

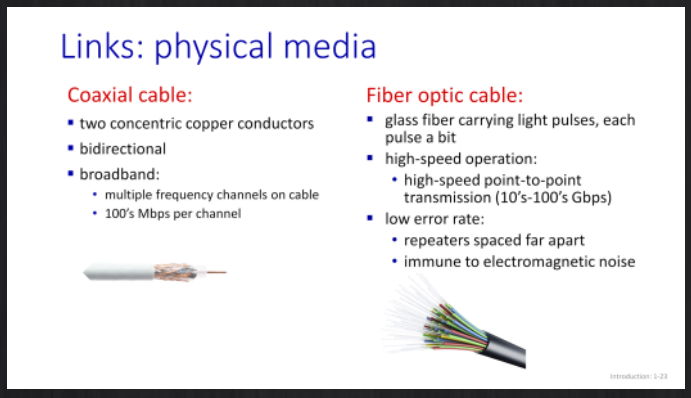
- For each transmitter-receiver pair, the bit is sent by propagating electromagnetic waves or optical pulses across a **physical medium.**
- The physical medium can take many shapes and forms and does not have to be of the same type for each transmitter-receiver pair along the path.
- Examples of physical media include
  - twisted-pair copper wire,
  - coaxial cable,
  - multimode fiber-optic cable,
  - terrestrial radio spectrum,
  - and satellite radio spectrum.
- Physical media fall into two categories:
  - **guided media and unguided media.**
  - With guided media, the waves are guided along a solid medium, such as a fiber-optic cable, a twisted-pair copper wire, or a coaxial cable.
  - With unguided media, the waves propagate in the atmosphere and in outer space, such as in a wireless LAN or a digital satellite channel.

### Twisted-Pair Copper Wire

- The least expensive and most commonly used guided transmission medium is twisted-pair copper wire.
- Unshielded twisted pair (UTP) is com monly used for computer networks within a building, that is, for LANs. Data rates for LANs using twisted pair

today range from 10 Mbps to 10 Gbps.

- Modern twisted-pair technology, such as category 6a cable, can achieve data rates of 10 Gbps for distances up to a hundred meters. In the end, twisted pair has emerged as the dominant solution for high-speed LAN networking.

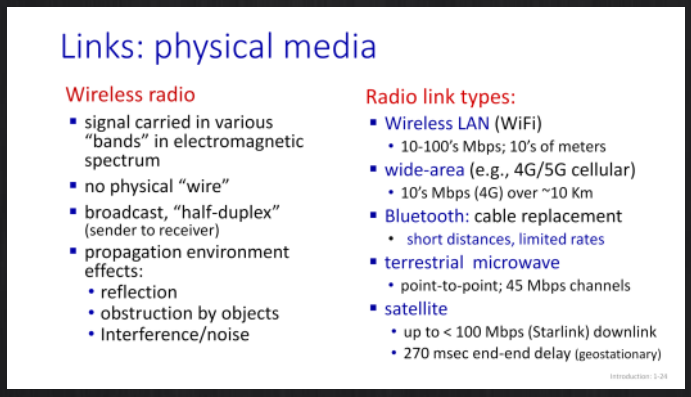



## Coaxial Cable

- Like twisted pair, coaxial cable consists of two copper conductors, but the two con ductors are concentric rather than parallel. With this construction and special insula tion and shielding, coaxial cable can achieve high data transmission rates.
- Coaxial cable can be used as a guided shared medium.

## Fiber Optics

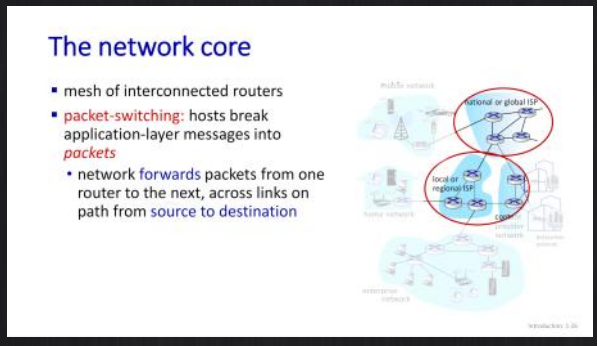
- An optical fiber is a thin, flexible medium that conducts pulses of light, with each pulse representing a bit. A single optical fiber can support tremendous bit rates, up to tens or even hundreds of gigabits per second.
- They are immune to electromagnetic interference, have very low signal attenuation up to 100 kilometers, and are very hard to tap.




- Radio channels carry signals in the electromagnetic spectrum.
- They are an attractive medium because they require no physical wire to be installed, can penetrate walls, provide connectivity to a mobile user, and can potentially carry a signal for long distances.
- Environmental considerations determine path loss and shadow fading (which decrease the signal strength as the signal travels over a distance and around/through obstructing objects), multipath fading (due to signal reflection off of interfering objects), and interference (due to other transmissions and electro magnetic signals).

Terrestrial radio channels can be broadly classified into three groups:
**those that operate over very short distance (e.g., with one or two meters);**
**those that operate in local areas, typically spanning from ten to a few hundred meters; those that oper ate in the wide area, spanning tens of kilometers.**
Personal devices such as wireless headsets, keyboards, and medical devices operate over short distances; the wireless LAN technologies described in Section 1.2.1 use local -area radio channels; the cel lular access technologies use wide -area radio channels.

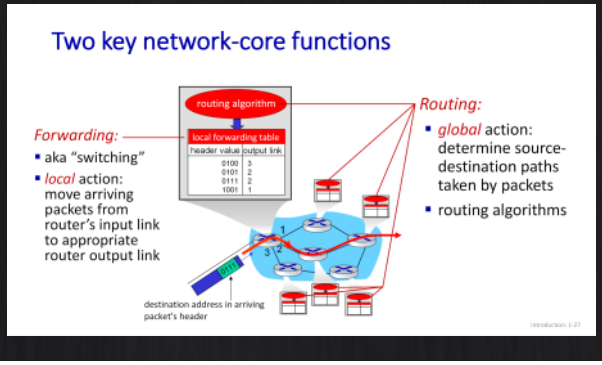
# 1.3 The Network Core:




let us now delve more deeply inside the network core—the mesh of packet switches and links that interconnects the Internet's end systems.

- In a network application, end systems exchange messages with each other.
- To send a message from a source end system to a destination end system, the source breaks long messages into smaller chunks of data known as **packets**.
- Between source and destination, each packet travels through communication links and packet switches (for which there are two predominant types, **routers** and **link-layer switches**).

## Store-and-Forward Transmission




- Most packet switches use store-and-forward transmission at the inputs to the links. Store-and-forward transmission means that the packet switch must receive the entire packet before it can begin to transmit the first bit of the packet onto the outbound link.

the router has the rather simple task of transferring a packet from one (input) link to the only other attached link.

In this example, the source has three packets, each consisting of L bits, to send to the destination. At the snapshot of time shown in Figure 1.11, the source has transmitted some of packet 1, and the front of packet 1 has already arrived at the router. Because the router employs store-and-forwarding, at this instant of time, the router cannot transmit the bits it has received; instead it must first buffer (i.e., "store") the packet's bits. Only after the router has received all of the packet's bits can it begin to transmit (i.e., "forward") the packet onto the outbound link. To gain some insight into store-and-forward transmission, let's now calculate the amount of time that elapses from when the source begins to send the packet until the destination has received the entire packet. (Here we will ignore propagation delay—the time it takes for the bits to travel across the wire at near the speed of light—which will be discussed in Section 1.4.) The source begins to transmit at time 0; at time L/R seconds, the source has transmitted the entire packet, and the entire packet has been received and stored at the router (since there is no propagation delay). At time L/R seconds, since the router has just received the entire packet, it can begin to transmit the packet onto the outbound link towards the des tination; at time 2L/R, the router has transmitted the entire packet, and the entire packet has been received by the destination. Thus, the total delay is 2L/R.

Now let's calculate the amount of time that elapses from when the source begins to send the first packet until the destination has received all three packets. As before, at time L/R, the router begins to forward the first packet. But also at time L/R the source will begin to send the second packet, since it has just finished sending the entire first packet. Thus, at time 2L/R, the destination has received the first packet and the router has received the second packet. Similarly, at time 3L/R, the destina tion has received the first two packets and the router has received the third packet. Finally, at time 4L/R the destination has received all three packets!

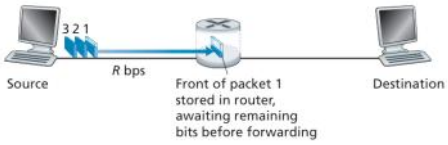


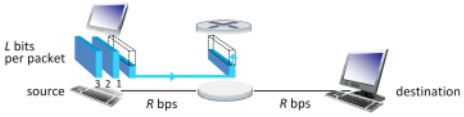

**Figure 1.11** ♦ Store-and-forward packet switching

Let's now consider the general case of sending one packet from source to destination over a path consisting of $N$ links each of rate $R$ (thus, there are $N$-1 routers between source and destination). Applying the same logic as above, we see that the end-to-end delay is:

$$d_{\text{end-to-end}} = N\frac{L}{R} \qquad (1.1)$$

You may now want to try to determine what the delay would be for $P$ packets sent over a series of $N$ links.

## Packet-switching: store-and-forward

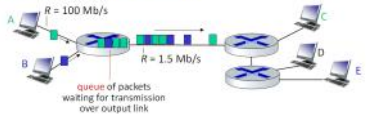



- packet transmission delay: takes $L/R$ seconds to transmit (push out) $L$-bit packet into link at $R$ bps
- *store and forward: entire* packet must arrive at router before it can be transmitted on next link

*One-hop numerical example:*
- $L$ = 10 Kbits
- $R$ = 100 Mbps
- one-hop transmission delay = 0.1 msec

## Queuing Delays and Packet Loss:

### Packet-switching: queueing




*Packet queuing and loss:* if arrival rate (in bps) to link exceeds transmission rate (bps) of link for some period of time:
- packets will queue, waiting to be transmitted on output link
- packets can be dropped (lost) if memory (buffer) in router fills up

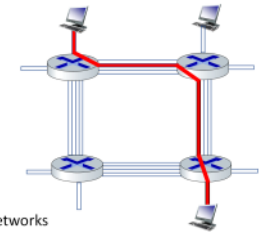
- Each packet switch has multiple links attached to it. For each attached link, the packet switch has an **output buffer** (also called an **output queue**), which stores packets that the router is about to send into that link.
- If an arriving packet needs to be transmitted onto a link but finds the link busy with the transmission of another packet, the arriving packet must wait in the output buffer.
- Thus, in addition to the store-and-forward delays, packets suffer output buffer queuing delays.
- Since the amount of buffer space is finite, an arriving packet may find that the buffer is completely full with other packets waiting for transmission. In this case, **packet loss** will occur—either the arriving packet or one of the already-queued packets will be dropped.

## 1.3.2 Circuit Switching

## Alternative to packet switching: circuit switching

end-end resources allocated to, reserved for "call" between source and destination

- in diagram, each link has four circuits.
  - call gets 2nd circuit in top link and 1st circuit in right link.
- dedicated resources: no sharing
  - circuit-like (guaranteed) performance
- circuit segment idle if not used by call (no sharing)
- commonly used in traditional telephone networks



* Check out the online interactive exercises for more examples: http://gaia.cs.umass.edu/kurose_/ross/interactive

Introduction: 1-33

- There are two fundamental approaches to moving data through a network of links and switches: **circuit switching** and **packet switching.**
- In circuit-switched networks, the resources needed along a path (buffers, link transmission rate) to provide for communication between the end systems are reserved for the duration of the communication session between the end systems.
- In packet-switched networks, these resources are not reserved; a session's messages use the resources on demand and, as a consequence, may have to wait (that is, queue) for access to a communication link.

As a simple **analogy**, consider two restaurants, one that requires reservations and another that neither requires reservations nor accepts them. For the restaurant that requires reservations, we have to go through the hassle of calling before we leave home. But when we arrive at the restaurant we can, in principle, immediately be seated and order our meal. For the restaurant that does not require reservations, we don't need to bother to reserve a table. But when we arrive at the restaurant, we may have to wait for a table before we can be seated.

Traditional telephone networks are examples of circuit-switched networks.

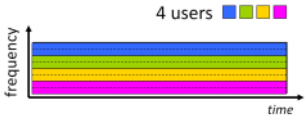
- Traditional telephone networks are examples of circuit-switched networks. Consider what happens when one person wants to send information (voice or facsimile) to another over a telephone network.
- Before the sender can send the information, the network must establish a connection between the sender and the receiver.
- This is a bona fide connection for which the switches on the path between the sender and receiver maintain connection state for that connection.
- In the jargon of telephony, this connection is called a **circuit**.
- When the network establishes the circuit, it also reserves a constant transmission rate in the network's links (representing a fraction of each link's transmission capacity) for the duration of the connection. Since a given transmission rate has been reserved for this sender-to-receiver connection, the sender can transfer the data to the receiver at the guaranteed constant rate.

## Multiplexing in Circuit-Switched Networks

## Circuit switching: FDM and TDM

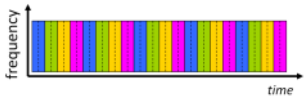
Frequency Division Multiplexing (FDM)

- optical, electromagnetic frequencies divided into (narrow) frequency bands
- each call allocated its own band, can transmit at max rate of that narrow band




Time Division Multiplexing (TDM)

- time divided into slots
- each call allocated periodic slot(s), can transmit at maximum rate of (wider) frequency band (only) during its time slot(s)




Introduction: 1-34

- A circuit in a link is implemented with either **frequency-division multiplexing** (FDM) or **time-division multiplexing (TDM).**

**FDM:**

- With FDM, the frequency spectrum of a link is divided up among the connections established across the link.
- Specifi cally, the link dedicates a frequency band to each connection for the duration of the connection. In telephone networks, this frequency band typically has a width of 4 kHz (that is, 4,000 hertz or 4,000 cycles per second). The width of the band is called, not surprisingly, the bandwidth. FM radio stations also use FDM to share the frequency spectrum between 88 MHz and 108 MHz, with each station being allocated a specific frequency band.

**TDM:**

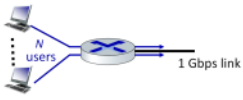
- For a TDM link, time is divided into frames of fixed duration, and each frame is divided into a fixed number of time slots.
- When the network establishes a connection across a link, the network dedicates one time slot in every frame to this connection.
- These slots are dedicated for the sole use of that connection, with one time slot avail able for use (in every frame) to transmit the connection's data.

## Packet Switching Versus Circuit Switching:

## Packet switching versus circuit switching

example:

- 1 Gb/s link
- each user:
  - 100 Mb/s when "active"
  - active 10% of time




*Q:* how many users can use this network under circuit-switching and packet switching?

- *circuit-switching:* 10 users
- *packet switching:* with 35 users, probability > 10 active at same time is less than .0004 *

*Q:* how did we get value 0.0004?
*A:* HW problem (for those with course in probability only)

* Check out the online interactive exercises for more examples: http://gaia.cs.umass.edu/kurose_ross/interactive

Introduction: 1-35

Proponents of packet switching argue that
(1) it offers better sharing of transmission capacity than circuit switching and
(2) it is simpler, more efficient, and less costly to implement than circuit switching.
Generally speaking, people who do not like to hassle with restaurant reservations prefer packet switching to circuit switching.

### Why is packet switching more efficient?

Suppose users share a 1 Mbps link. Also suppose that each user alternates between peri ods of activity, when a user generates data at a constant rate of 100 kbps, and periods of inactivity, when a user generates no data. Suppose further that a user is active only 10 percent of the time (and is idly drinking coffee during the remaining 90 percent of the time). With circuit switching, 100 kbps must be reserved for each user at all times. For example, with circuit-switched TDM, if a one-second frame is divided into 10 time slots of 100 ms each, then each user would be allocated one time slot per frame. Thus, the circuit-switched link can support only 10 (= 1 Mbps/100 kbps) simul taneous users. With packet switching, the probability that a specific user is active is 0.1 (that is, 10 percent). If there are 35 users, the probability that there are 11 or more simultaneously active users is approximately 0.0004. (Homework Problem P8 outlines how this probability is obtained.) When there are 10 or fewer simultane ously active users (which happens with probability 0.9996), the aggregate arrival rate of data is less than or equal to 1 Mbps, the output rate of the link. Thus, when there are 10 or fewer active users, users' packets flow through the link essentially

**Let's now consider a second simple example.**

Suppose there are 10 users and that one user suddenly generates one thousand 1,000-bit packets, while other users remain quiescent and do not generate packets. Under TDM circuit switching with 10 slots per frame and each slot consisting of 1,000 bits, the active user can only use its one time slot per frame to transmit data, while the remaining nine time slots in each frame remain idle. It will be 10 seconds before all of the active user's one million bits of data has been transmitted. In the case of packet switching, the active user can continuously send its packets at the full link rate of 1 Mbps, since there are no other users generating packets that need to be multiplexed with the active user's packets. In this case, all of the active user's data will be transmitted within 1 second.

## Packet switching versus circuit switching

Is packet switching a "slam dunk winner"?

- great for "bursty" data – sometimes has data to send, but at other times not
  - resource sharing
  - simpler, no call setup
- excessive congestion possible: packet delay and loss due to buffer overflow
  - protocols needed for reliable data transfer, congestion control
- *Q:* How to provide circuit-like behavior with packet-switching?
  - "It's complicated." We'll study various techniques that try to make packet switching as "circuit-like" as possible.

*Q:* human analogies of reserved resources (circuit switching) versus on-demand allocation (packet switching)?

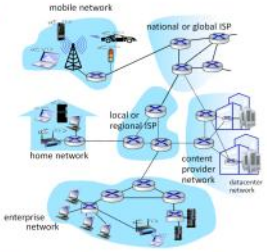


---

# 1.3.3 A Network of Networks

## Internet structure: a "network of networks"

- hosts connect to Internet via access Internet Service Providers (ISPs)
- access ISPs in turn must be interconnected
  - so that *any* two hosts *(anywhere!)* can send packets to each other
- resulting network of networks is very complex
  - evolution driven by economics, national policies

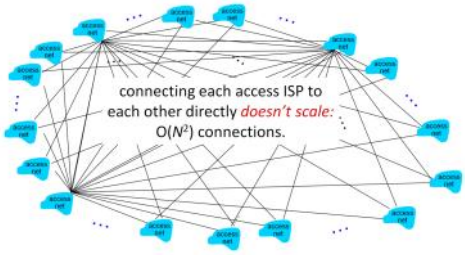



*Let's take a stepwise approach to describe current Internet structure*

- We saw earlier that end systems (PCs, smartphones, Web servers, mail servers, and so on) connect into the Internet via an access ISP.
- The access ISP can provide either wired or wireless connectivity, using an array of access technologies including DSL, cable, FTTH, Wi-Fi, and cellular.
- Note that the access ISP does not have to be a telco or a cable company; instead it can be, for example, a university (providing Internet access to students, staff, and faculty), or a company (providing access for its employees).
- But connecting end users and content providers into an access ISP is only a small piece of solving the puzzle of connecting the billions of end systems that make up the Internet.
- To complete this puzzle, the access ISPs themselves must be interconnected. This is done by creating a network of networks—understanding this phrase is the key to understanding the Internet.

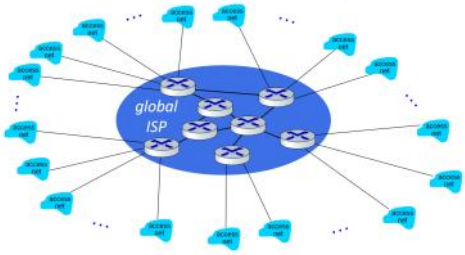
## Internet structure: a "network of networks"

*Question:* given *millions* of access ISPs, how to connect them together?



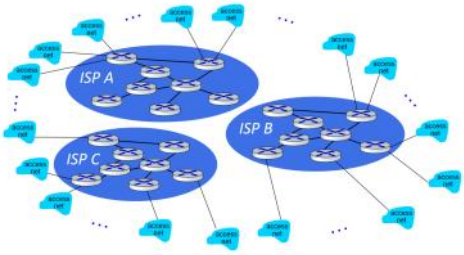

## Internet structure: a "network of networks"

*Option: connect each access ISP to one global transit ISP?*
*Customer and provider ISPs have economic agreement.*



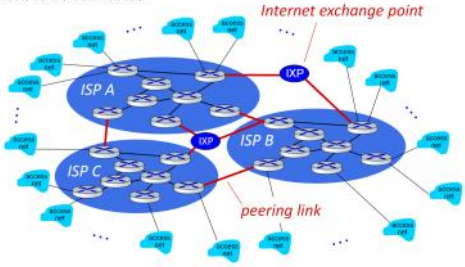

## Internet structure: a "network of networks"

But if one global ISP is viable business, there will be competitors ....




## Internet structure: a "network of networks"

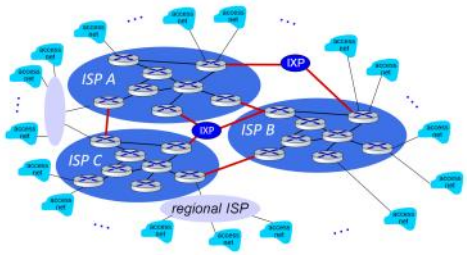
But if one global ISP is viable business, there will be competitors .... who will want to be connected




Along these same lines, a third-party company can create an **Internet Exchange Point (IXP)**, which is a meeting point where multiple ISPs can peer together. An IXP is typically in a stand-alone building with its own switches [Ager 2012]. There are over 600 IXPs in the Internet today [PeeringDB 2020].

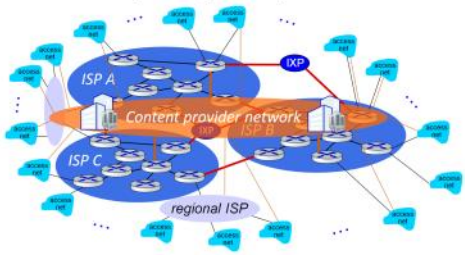
## Internet structure: a "network of networks"

… and regional networks may arise to connect access nets to ISPs

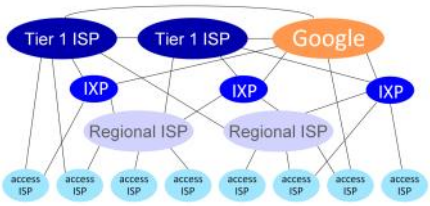



## Internet structure: a "network of networks"

… and content provider networks (e.g., Google, Microsoft, Akamai) may run their own network, to bring services, content close to end users




## Internet structure: a "network of networks"




At "center": small # of well-connected large networks

- "tier-1" commercial ISPs (e.g., Level 3, Sprint, AT&T, NTT), national & international coverage
- content provider networks (e.g., Google, Facebook): private network that connects its data centers to Internet, often bypassing tier-1, regional ISPs

---

# 1.4 Delay, Loss, and Throughput in Packet-Switched Networks:

- Ideally, we would like Internet services to be able to move as much data as we want between any two end systems, instantaneously, without any loss of data.
- computer networks necessarily constrain throughput (the amount of data per second that can be transferred) between end sys tems, introduce delays between end systems, and can actually lose packets.

## 1.4.1 Overview of Delay in Packet-Switched Network:

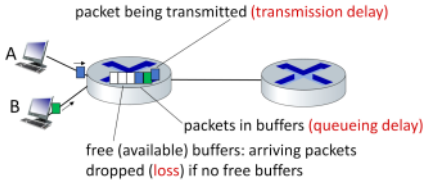
1. As a packet travels from one node (host or router) to the subsequent node (host or router) along this path, the packet suffer s from several types of delays at each node along the path.
2. The most important of these delays are the **nodal processing delay**, **queuing delay**, **transmission delay**, and **propagation delay**; together, these delays accumulate to give a **total nodal delay.**

### Types of Delay:

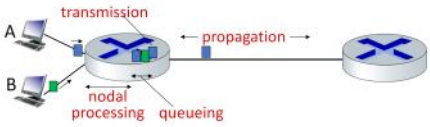
1. **Processing Delay:**
   a. The time required to examine the packet's header and determine where to direct the packet is part of the processing delay.
   b. The processing delay can also include other factors, such as the time needed to check for bit-level errors in the packet that occurred in transmitting the packet's bits from the upstream node to router A.
   c. Processing delays in high-speed routers are typically on the order of microseconds or less. After this nodal processing, the router directs the packet to the queue that precedes the link to router B.
2. **Queuing Delay:**
   a. At the queue, the packet experiences a queuing delay as it waits to be transmitted onto the link.
   b. The length of the queuing delay of a specific packet will depend on the number of earlier-arriving packets that are queued and waiting for transmission onto the link.
   c. If the queue is empty and no other packet is currently being transmitted, then our packet's queuing delay will be zero. On the other hand, if the traffic is heavy and many other packets are also waiting to be transmitted, the queuing delay will be long.
   d. Queuing delays can be on the order of microseconds to milliseconds in practice.
3. **Transmission Delay:**
   a. Assuming that packets are transmitted in a first-come-first-served manner, as is com mon in packet-switched networks, our packet can be transmitted only after all the packets that have arrived before it have been transmitted.
   b. Denote the length of the packet by L bits, and denote the transmission rate of the link from router A to router B by R bits/sec.
      i. For example, for a 10 Mbps Ethernet link, the rate is R = 10 Mbps; for a 100 Mbps Ethernet link, the rate is R = 100 Mbps. The transmission delay is L/R. This is the amount of time required to push (that is, transmit) all of the packet's bits into the link.
   c. **Transmission delays are typically on the order of microseconds to milliseconds in practice.**
4. **Propagation delay:**
   a. Once a bit is pushed into the link, it needs to propagate to router B. The time required to propagate from the beginning of the link to router B is the propagation delay.
   b. The bit propagates at the propagation speed of the link. The propagation speed depends on the physical medium of the link (th at is, fiber optics, twisted-pair copper wire, and so on) and is in the range of
   c. $2 \cdot 10^8$ meters/sec to $3 \cdot 10^8$ meters/sec
      which is equal to, or a little less than, the speed of light.
   d. The propagation delay is the distance between two routers divided by the propagation speed. That is, the propaga tion delay is d/s, where d is the distance between router A and router B and s is the propagation speed of the link.
   e. In wide-area networks, propagation delays are on the order of **milliseconds.**

## How do packet delay and loss occur?

- packets *queue* in router buffers, waiting for turn for transmission
  - queue length grows when arrival rate to link (temporarily) exceeds output link capacity
- packet *loss* occurs when memory to hold queued packets fills up




## Packet delay: four sources

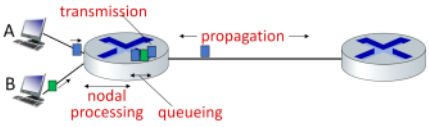



$d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$

$d_{proc}$: nodal processing
- check bit errors
- determine output link
- typically < microsecs

$d_{queue}$: queueing delay
- time waiting at output link for transmission
- depends on congestion level of router

## Packet delay: four sources




$d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$

$d_{trans}$: transmission delay:
- $L$: packet length (bits)
- $R$: link transmission rate (bps)
- $d_{trans} = L/R$

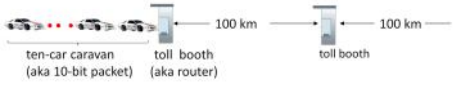
$d_{prop}$: propagation delay:
- $d$: length of physical link
- $s$: propagation speed (~$2 \times 10^8$ m/sec)
- $d_{prop} = d/s$

$d_{trans}$ and $d_{prop}$ *very* different

## Caravan analogy


- car ~ bit; caravan ~ packet; toll service ~ link transmission
- toll booth takes 12 sec to service car (bit transmission time)
- "propagate" at 100 km/hr
- *Q:* How long until caravan is lined up before 2nd toll booth?
- time to "push" entire caravan through toll booth onto highway = 12*10 = 120 sec
- time for last car to propagate from 1st to 2nd toll both: 100km/(100km/hr) = 1 hr
- *A:* 62 minutes

## Caravan analogy




- suppose cars now "propagate" at 1000 km/hr
- and suppose toll booth now takes one min to service a car
- *Q:* Will cars arrive to 2nd booth before all cars serviced at first booth?

  *A: Yes!* after 7 min, first car arrives at second booth; three cars still at first booth

**Comparing Transmission and Propagation Delay:**

- **The transmission delay** is the amount of time required for the router to push out the packet; it is a function of the packet's length and the transmission rate of the link, but has nothing to do with the distance between the two routers.
- **The propagation delay**, on the other hand, is the time it takes a bit to propagate from one router to the next; it is a function of the distance between the two routers, but has nothing to do with the packet's length or the transmission rate of the link.

$$d_{\text{nodal}} = d_{\text{proc}} + d_{\text{queue}} + d_{\text{trans}} + d_{\text{prop}}$$

.

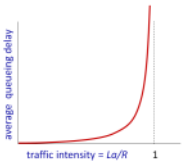
- The contribution of these delay components can vary significantly.
  - For example, **dprop** can be negligible (for example, a couple of microseconds) for a link connecting two routers on the same university campus; however, dprop is hundreds of milliseconds for two routers interconnected by a geostationary satellite link, and can be the dominant term in dnodal.
- Similarly, **dtrans** can range from negligible to significant. Its contribution is typically negligible for transmission rates of 10 Mbps and higher (for example, for LANs); however, it can be hundreds of milliseconds for large Internet packets sent over low-speed dial-up modem links.
- The processing delay, **dproc**, is often negligible; however, it strongly influences a router's maximum throughput, which is the maximum rate at which a router can forward packets.

## 1.4.2 Queuing Delay and Packet Loss

## Packet queueing delay (revisited)

- *a:* average packet arrival rate
- *L:* packet length (bits)
- *R:* link bandwidth (bit transmission rate)

$\frac{L \cdot a}{R}$ : $\frac{\text{arrival rate of bits}}{\text{service rate of bits}}$ *"traffic intensity"*




- *La/R* ~ 0: avg. queueing delay small
- *La/R* -> 1: avg. queueing delay large
- *La/R* > 1: more "work" arriving is more than can be serviced - average delay infinite!

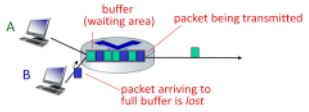

**When is the queuing delay large and when is it insignificant?**

- The answer to this question depends on the rate at which traffic arrives at the queue, the transmission rate of the link, and the nature of the arriving traffic, that is, whether the traffic arrives periodically or arrives in bursts.

- The most complicated and interesting component of nodal delay is the **queuing delay,**
- the queuing delay can vary from packet to packet.
  - For example, if 10 packets arrive at an empty queue at the same time, the first packet transmitted will suffer no queuing delay, while the last packet transmitted will suffer a relatively large queuing delay (while it waits for the other nine packets to be transmitted).
- Therefore, when characterizing queuing delay, one typically uses statistical measures, such as average queuing delay, variance of queuing delay, and the probability that the queuing delay exceeds some specified value.

## Packet loss

- queue (aka buffer) preceding link in buffer has finite capacity
- packet arriving to full queue dropped (aka lost)
- lost packet may be retransmitted by previous node, by source end system, or not at all




* Check out the Java applet for an interactive animation (on publisher's website) of queuing and loss

**Packet Loss:**

- In reality a queue preceding a link has finite capacity, although the queuing capacity greatly depends on the router design and cost.
- Because the queue capacity is finite, packet delays do not really approach infinity as the traffic intensity approaches 1.
- Instead, a packet can arrive to find a full queue. With no place to store such a packet, a router will **drop** that packet; that is, the packet will be lost.
- This overflow at a queue can again be seen in the interactive animation when the traffic intensity is greater than 1.
- The fraction of lost packets increases as the **traffic intensity increases**. Therefore, performance at a node is often measured not only in terms of delay, but also in terms of the probability of packet loss.

## 1.4.3 End-to-End Delay

Our discussion up to this point has focused on the nodal delay, that is, the delay at a single router. Let's now consider the total delay from source to destination. To get a handle on this concept, suppose there are $N - 1$ routers between the source host and the destination host. Let's also suppose for the moment that the network is uncongested (so that queuing delays are negligible), the processing delay at each router and at the source host is $d_{\text{proc}}$, the transmission rate out of each router and out of the source host is $R$ bits/sec, and the propagation on each link is $d_{\text{prop}}$. The nodal delays accumulate and give an end-to-end delay,

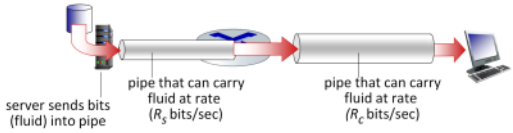
$$d_{\text{end-end}} = N(d_{\text{proc}} + d_{\text{trans}} + d_{\text{prop}}) \qquad (1.2)$$

where, once again, $d_{\text{trans}} = L/R$, where $L$ is the packet size. Note that Equation 1.2 is a generalization of Equation 1.1, which did not take into account processing and propagation delays. We leave it to you to generalize Equation 1.2 to the case of heterogeneous delays at the nodes and to the presence of an average queuing delay at each node.

## 1.4.4 Throughput in Computer Networks

### Throughput

- *throughput:* rate (bits/time unit) at which bits are being sent from sender to receiver
  - *instantaneous:* rate at given point in time
  - *average:* rate over longer period of time




- To define throughput, consider transferring a large file from Host A to Host B across a computer network. This transfer might be, for example, a large video clip from one computer to another.
- The instantaneous throughput at any instant of time is the rate (in bits/sec) at which Host B is receiving the file.
- If the file con sists of F bits and the transfer takes T seconds for Host B to receive all F bits, then the **average throughput** of the file transfer is F/T bits/sec.

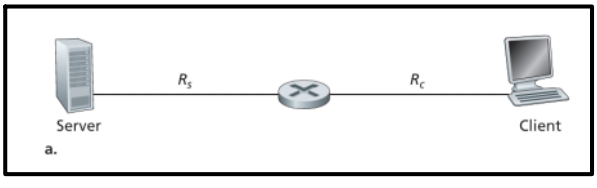
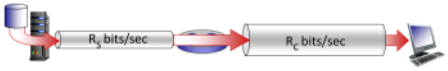


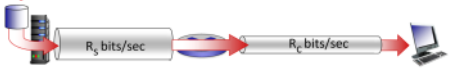

Figure 1.19(a) shows two end systems, a server and a client, con nected by two communication links and a router. Consider the throughput for a file transfer from the server to the client. Let Rs denote the rate of the link between the server and the router; and Rc denote the rate of the link between the router and the client. Suppose that the only bits being sent in the entire network are those from the server to the client. We now ask, in this ideal scenario, what is the server - to-client throughput? To answer this question, we may think of bits as fluid and com munication links as pipes. Clearly, the server cannot pump bits through its link at a rate faster than Rs bps; and the router cannot forward bits at a rate faster than Rc bps. If Rs 6 Rc, then the bits pumped by the server will "flow" right through the router and arrive at the client at a rate of Rs bps, giving a throughput of Rs bps. If, on the other hand, Rc 6 Rs, then the router will not be able to forward bits as quickly as it receives them. In this case, bits will only leave the router at rate Rc, giving an end to-end throughput of Rc.

## Throughput

$R_s < R_c$ What is average end-end throughput?




$R_s > R_c$ What is average end-end throughput?




*bottleneck link*

link on end-end path that constrains end-end throughput

Introduction: 1-57

Thus, for this simple two-link network, the throughput is **min{Rc, Rs},** that is, it is the transmission rate of the **bottleneck link.**

For a **specific example**, suppose that you are downloading an MP3 file of F = 32 million bits, the server has a transmission rate of Rs = 2 Mbps, andyou have an access link of Rc = 1 Mbps. The time needed to transfer the file is then **32 seconds**. Of course, these expressions for throughput and transfer time are only approximations, as they do not account for storeand-forward and processing delays as well as protocol issues.

# 1.5 Protocol Layers and Their Service Models

## 1.5.1 Layered Architecture

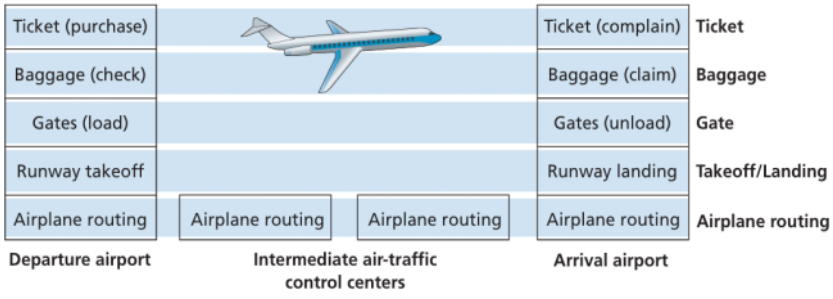



**Figure 1.22** ♦ Horizontal layering of airline functionality

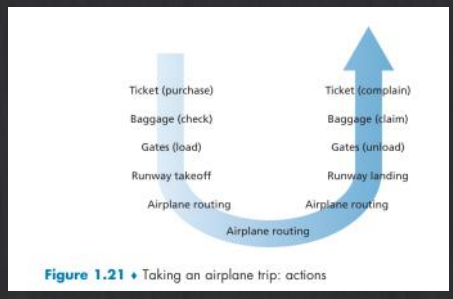



**Figure 1.21** ♦ Taking an airplane trip: actions

- A layered architecture allows us to discuss a well-defined, specific part of a large and complex system.
- This simplification itself is of considerable value by providing modularity, making it much easier to change the implementation of the service provided by the layer. As long as the layer provides the same service to the layer above it, and uses the same services from the layer below it, the remainder of the system remains unchanged when a layer's implementation is changed.

### Protocol "layers" and reference models

Networks are complex, with many "pieces":
- hosts
- routers
- links of various media
- applications
- protocols
- hardware, software

*Question:* is there any hope of *organizing* structure of network?
- and/or our *discussion* of networks?

Introduction: 1-67

### Why layering?

Approach to designing/discussing complex systems:
- explicit structure allows identification, relationship of system's pieces
  - layered *reference model* for discussion
- modularization eases maintenance, updating of system
  - change in layer's service *implementation*: transparent to rest of system
  - e.g., change in gate procedure doesn't affect rest of system

Introduction: 1-70

### Layered Internet protocol stack

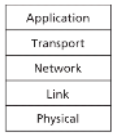
- *application:* supporting network applications
  - HTTP, IMAP, SMTP, DNS
- *transport:* process-process data transfer
  - TCP, UDP
- *network:* routing of datagrams from source to destination
  - IP, routing protocols
- *link:* data transfer between neighboring network elements
  - Ethernet, 802.11 (WiFi), PPP
- *physical:* bits "on the wire"


Introduction: 1-71

- *physical:* bits "on the wire"

# Protocol Layering

- To provide structure to the design of network protocols, network designers organize protocols—and the network hardware and software that implement the protocols— in layers.
- We are again interested in the services that a layer offers to the layer above—the so-called service model of a layer. Just as in the case of our airline example, each layer provides its service by
  - (1) performing certain actions within that layer and by
  - (2) using the services of the layer directly below it.
    - For example, the services provided by layer n may include reliable delivery of messages from one edge of the network to the other. This might be implemented by using an unreliable edge-to-edge message delivery service of layer n- 1, and adding layer n functionality to detect and retransmit lost messages.
- A protocol layer can be implemented in software, in hardware, or in a combina tion of the two. **Application-layer protocols**—such as **HTTP** and **SMTP**—are almost always implemented in software in the end systems; so are transport-layer protocols.
- Because the **physical layer** and **data link layers** are responsible for handling commu nication over a specific link, they are typically implemented in a network interface card (for example, Ethernet or WiFi interface cards) associated with a given link.
- The **network layer** is often a mixed implementation of hardware and software.



Five-layer Internet protocol stack

When taken together, the protocols of the various layers are called the protocol stack. The Internet protocol stack consists of five layers:
the physical,
link,
network,
transport, and
application layers,

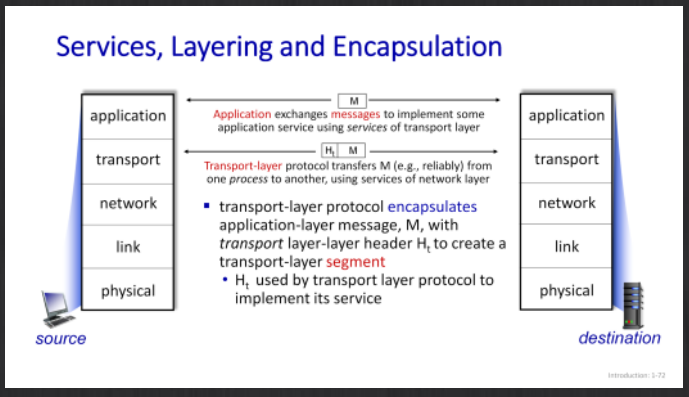
# Application Layer

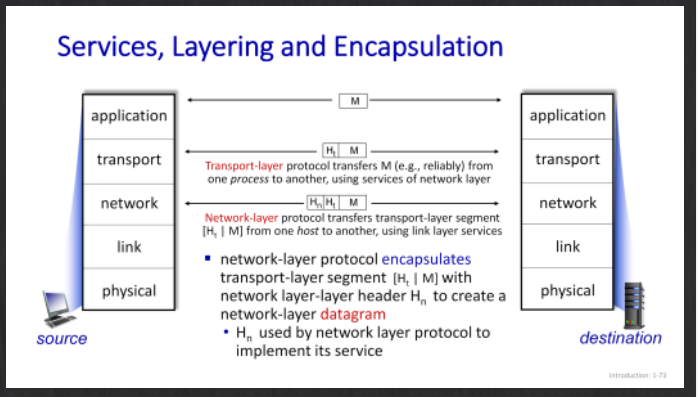
- The application layer is where network applications and their application-layer pro tocols reside.
- The Internet's application layer includes many protocols, such as the
  - **HTTP** protocol (which provides for Web document request and transfer),
  - **SMTP** (which provides for the transfer of e-mail messages), and
  - **FTP** (which provides for the transfer of files between two end systems).
- We'll see that certain network functions, such as the translation of human-friendly names for Internet end systems like www.ietf.org to a 32-bit network address, are also done with the help of a specific application-layer protocol, namely, **the domain name system (DNS).**
- An application-layer protocol is distributed over multiple end systems, with the application in one end system using the protocol to exchange packets of information with the application in another end system. We'll refer to this packet of information at the application layer as a **message**.

# Transport Layer

- The Internet's transport layer transports application-layer messages between application endpoints.
- In the Internet, there are two transport protocols, **TCP** and **UDP**, either of which can transport application-layer messages.

- TCP
  - TCP provides a connection-oriented service to its applications.
  - This service includes guaranteed delivery of application-layer
  - TCP also breaks long messages into shorter segments and provides a congestion-control mechanism, so that a source throttles its transmission rate when the network is con gested.
- **UDP**
  - The UDP protocol provides a **connectionless** service to its applications.
  - This is a no-frills service that provides no reliability, no flow control, and no congestion control.
  - In this book, we'll refer to a transport-layer packet as a **segment**.




# Network Layer

- The Internet's network layer is responsible for moving network-layer packets known as datagrams from one host to another.
- The Internet transport-layer protocol (TCP or UDP) in a source host passes a transport-layer segment and a destination address to the network layer, just as you would give the postal service a letter with a destination address.
- The network layer then provides the service of delivering the segment to the transport layer in the destination host.
- The Internet's network layer includes the celebrated **IP protocol**, which defines the fields in the datagram as well as how the end systems and routers act on these

fields.

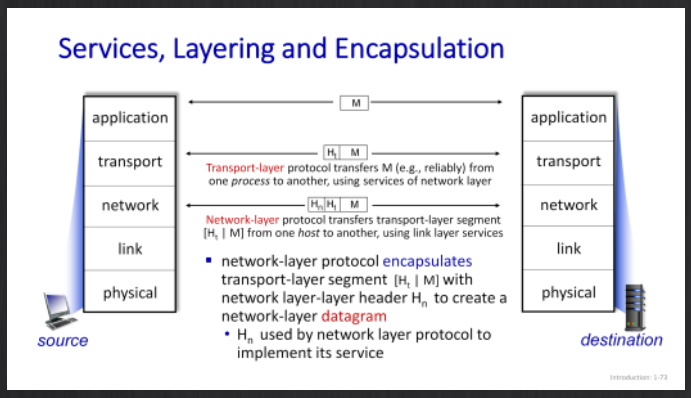
- There is **only one IP protocol**, and all Internet components that have a network layer must run the IP protocol.
- The Internet's network layer also contains routing protocols that determine the routes that datagrams take between sources and destina tions. The Internet has many routing protocols.

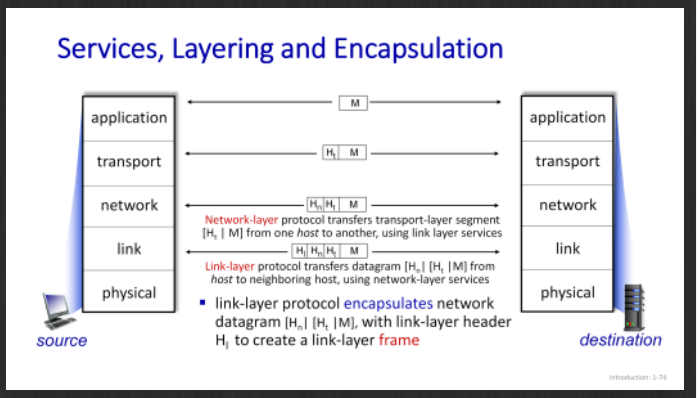
## Link Layer

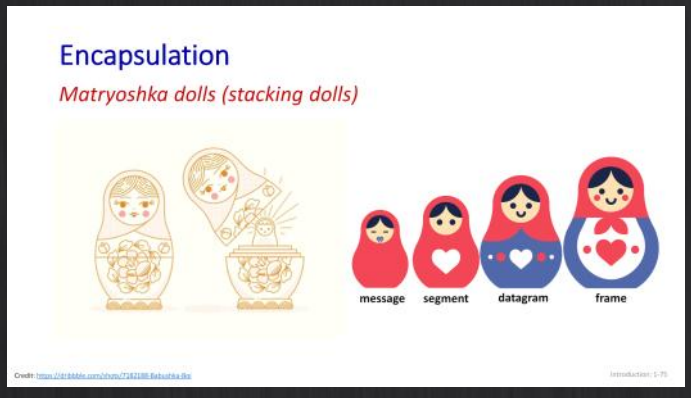
- The Internet's network layer routes a datagram through a series of routers between the source and destination.
- To move a packet from one node (host or router) to the next node in the route, the network layer relies on the services of the link layer. In particular, at each node, the network layer passes the datagram down to the link layer, which delivers the datagram to the next node along the route. At this next node, the link layer passes the datagram up to the network layer.
  - Examples of link-layer pro tocols include Ethernet, WiFi, and the cable access network's DOCSIS protocol.
- we'll refer to the link-layer packets as **frames.**

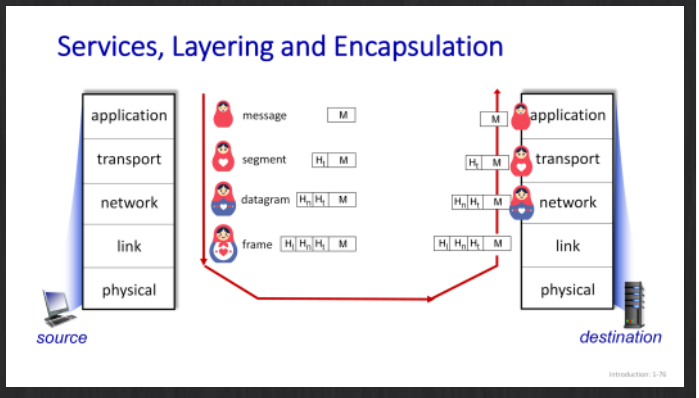
## Physical Layer

- While the job of the link layer is to move entire frames from one network element to an adjacent network element, the job of the physical layer is to move the individual bits within the frame from one node to the next.
- The protocols in this layer are again link dependent and further depend on the actual transmission medium of the link (for example, twisted-pair copper wire, single-mode fiber optics).
  - For example, Ether net has many physical-layer protocols: one for twisted-pair copper wire, another for coaxial cable, another for fiber, and so on. In each case, a bit is moved across the link in a different way.

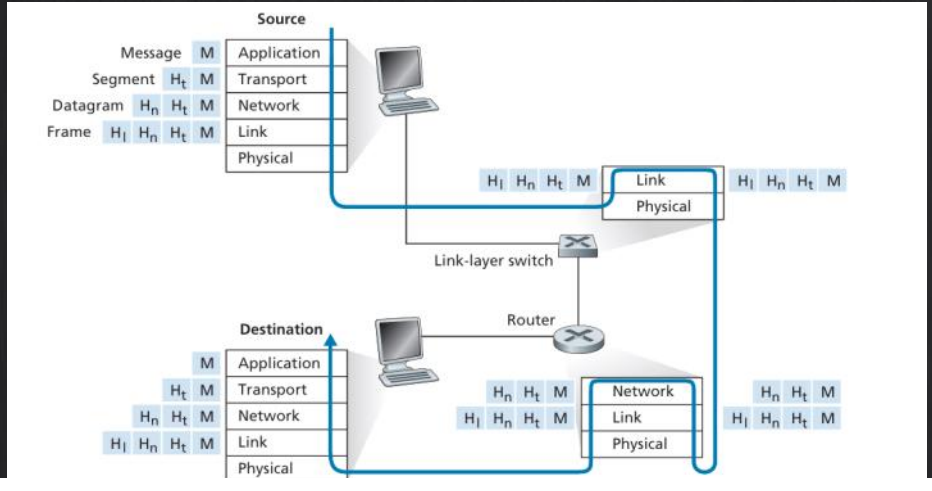



## 1.5.2 Encapsulation




### Router and Switches ke baare mai thosisi jaankari

- routers and link-layer switches are both packet switches.
- Similar to end systems, routers and link-layer switches organize their network ing **hardware and software into layers.**
- But routers and link-layer switches do not implement all of the layers in the protocol stack; they typically implement only the **bottom layers**.

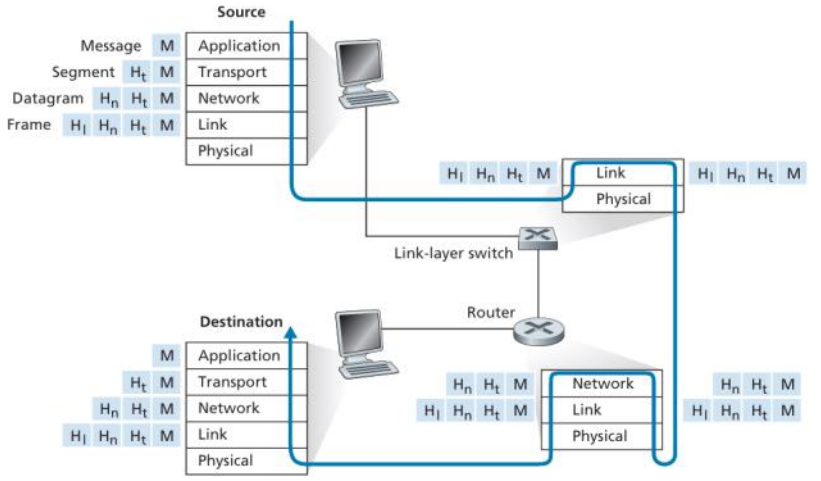



**Figure 1.24** ♦ Hosts, routers, and link-layer switches; each contains a different set of layers, reflecting their differences in functionality

### Router and Switches ke baare mai thosisi jaankari

- routers and link-layer switches are both packet switches.
- Similar to end systems, routers and link-layer switches organize their network ing **hardware and software into layers.**
- But routers and link-layer switches do not implement all of the layers in the protocol stack; they typically implement only the **bottom layers**.

- Figure 1.24 also illustrates the important concept of encapsulation.
- At the sending host, an application-layer message (M in Figure 1.24) is passed to the transport layer.
- In the simplest case, the transport layer takes the message and appends additional information (so-called transport-layer header information, Ht in Figure 1.24) that will be used by the receiver-side transport layer. The **application-layer message** and the **transport-layer header information** together constitute the **transport-layer segment**. The transport-layer segment thus encapsulates the application-layer message. The added information might include information allowing the receiver-side transport layer to deliver the message up to the **appro priate application**, and **error-detection** bits that allow the receiver to determine whether bits in the message have been changed in route.
- The transport layer then passes the segment to the **network layer**, which adds network-layer header information (Hn in Figure 1.24) such as **source and destination end system addresses,** creating a network-layer **datagram**.
- The datagram is then passed to the link layer, which (of course!) will add its own link-layer header information and cre ate a link-layer frame. Thus, we see that at each layer, a packet has two types of fields: **header fields and a payload field**. The payload is typically a packet from the layer above.

# 1.6 Networks Under Attack

## Network security

- Internet not originally designed with (much) security in mind
  - *original vision:* "a group of mutually trusting users attached to a transparent network" ☺
  - Internet protocol designers playing "catch-up"
  - security considerations in all layers!
- We now need to think about:
  - how bad guys can attack computer networks
  - how we can defend networks against attacks
  - how to design architectures that are immune to attacks

## The Bad Guys Can Put Malware into Your Host Via the Internet

- We attach devices to the Internet because we want to receive/send data from/to the Internet.
- This includes all kinds of good stuff, including Instagram posts, Internet search results, streaming music, video conference calls, streaming movies, and so on. But, unfortunately, along with all that good stuff comes malicious stuff— collectively known as malware—that can also enter and infect our devices.
- Once malware infects our device it can do all kinds of devious things, including delet ing our files and installing spyware that collects our private information, such as social security numbers, passwords, and keystrokes, and then sends this (over the Internet, of course!) back to the bad guys.
- Much of the malware out there today is **self-replicating**: once it infects one host, from that host it seeks entry into other hosts over the Internet, and from the newly infected hosts, it seeks entry into yet more hosts. In this manner, self- replicating malware can spread exponentially fast.

## The Bad Guys Can Attack Servers and Network Infrastructure

### Bad guys: denial of service

*Denial of Service (DoS):* attackers make resources (server, bandwidth) unavailable to legitimate traffic by overwhelming resource with bogus traffic

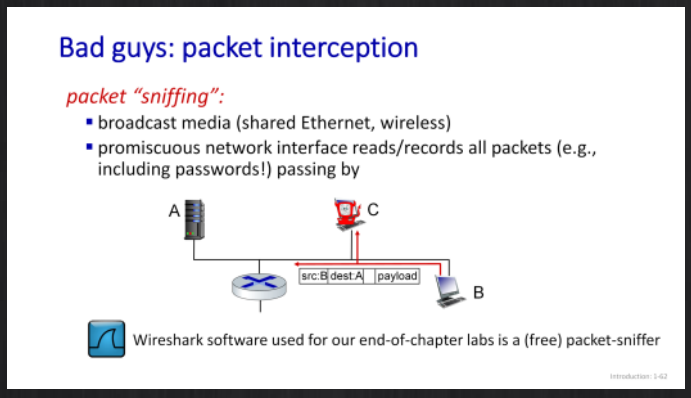
1. select target
2. break into hosts around the network (see botnet)
3. send packets to target from compromised hosts


- Another broad class of security threats are known as **denial-of-service (DoS) attacks.**
- As the name suggests, a DoS attack renders a network, host, or other piece of infrastructure unusable by legitimate users.
- Web servers, e-mail servers, DNS servers (discussed in Chapter 2), and institutional networks can all be subject to DoS attacks.
- **Vulnerability attack:** This involves sending a few well-crafted messages to a vulnerable application or operating system running on a targeted host. If the right sequence of packets is sent to a vulnerable application or operating system, the service can stop or, worse, the host can crash.
- **Bandwidth flooding:** The attacker sends a deluge of packets to the targeted host—so many packets that the target's access link becomes clogged, preventing legitimate packets from reaching the server.
- **Connection flooding:** The attacker establishes a large number of half-open or fully open TCP connections at the target host. The host can become so bogged down with these bogus connections that it stops accepting

**In a distributed DoS (DDoS) attack,** the attacker controls multiple sources and has each source blast traffic at the target. With this approach, the aggregate traffic rate across all the controlled sources needs to be approximately R to cripple the service.

bogged down with these bogus connections that it stops accepting legitimate connections.

## The Bad Guys Can Sniff Packets




- These packets can contain all kinds of sensitive information, including passwords, social security numbers, trade secrets, and private personal messages. A passive receiver that records a copy of every packet that flies by is called a packet sniffer.