Introduction

Srinidhi Varadarajan

What is a network?

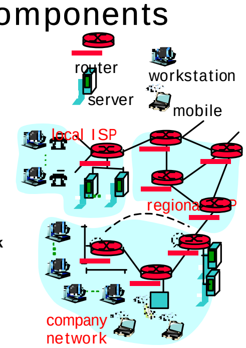
- Carrier of information between connected entities
- What does a network consist of?
  - End hosts connected to the network
  - Routers/switches that move data through the network
  - Physical links that carry information
    - E.g. Ethernet, FDDI, ATM, Token Ring
  - Applications that communicate with each other to provide services
    - E-Mail, File Transfer, Web Browser

What is an internetwork?

- A set of connected networks is called an internetwork.
- The *Internet* is a specific example of an internetwork.
  - The Internet is characterized by the use of a common network protocol (IP).
- An internetwork need NOT be connected to the *Internet*

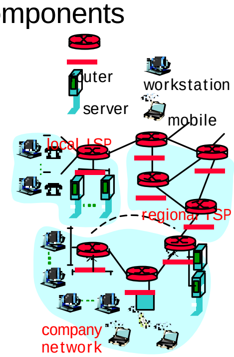
Internet: Components

- ***Network edge:***
  - millions of connected computing devices running ***network apps***
  - pc's workstations, servers
  - PDA's phones, toasters
- ***Network core:***
  - ***routers:*** forward packets (chunks) of data thru network
- ***Media:***
  - ***communication links:*** fiber, copper, radio, satellite



Internet: Components

- ***protocols*****:** control sending, receiving of msgs
  - e.g., TCP, IP, HTTP, FTP, PPP
- ***Internet:*** **"network of networks"**
  - loosely hierarchical
  - public Internet versus private intranet
- Internet standards
  - RFC: Request for comments
  - IETF: Internet Engineering Task Force



Internet: Services

- communication *infrastructure* enables distributed applications:
  - WWW, email, games, e-commerce, database., voting,
  - more?
- communication services provided:
  - connectionless
  - connection-oriented
- cyberspace [Gibson]:
  "a consensual hallucination experienced daily by billions of operators, in every nation, ..."

What's a protocol?

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

… specific msgs sent

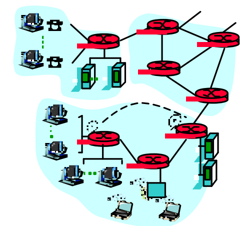
… specific actions taken when msgs received, or other events

network protocols:
- machines rather than humans
- all communication activity in Internet governed by protocols

protocols define format, order of msgs sent and received among network entities, and actions taken on msg transmission, receipt

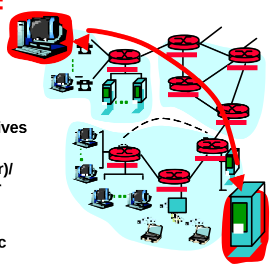
A closer look at network structure

- **network edge:** applications and hosts
- **network core:**
  - routers
  - network of networks
- **access networks, physical media:** communication links



The Network Edge

- **end systems (hosts):**
  - run application programs
  - e.g., WWW, email
  - at "edge of network"
- **client/server model**
  - client host requests, receives service from server
  - e.g., WWW client (browser)/ server; email client/server
- **peer-peer model:**
  - host interaction symmetric
  - e.g.: teleconferencing



Network edge: connection-oriented service

Goal: data transfer between end sys.
- *handshaking:* setup (prepare for) data transfer ahead of time
  - *set up "state"* in two communicating hosts
- TCP - Transmission Control Protocol
  - Internet's connection-oriented service

TCP service [RFC 793]
- *reliable, in-order* byte-stream data transfer
  - loss: acknowledgements and retransmissions
- *flow control:*
  - sender won't overwhelm receiver
- *congestion control:*
  - senders "slow down sending rate" when network congested

Network edge: connectionless service

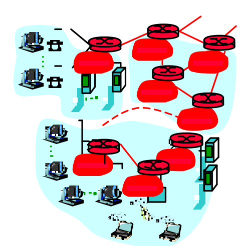
Goal: data transfer between end systems
- same as before!
- UDP - User Datagram Protocol [RFC 768]: Internet's connectionless service
  - unreliable data transfer
  - no flow control
  - no congestion control

App's using TCP:
- HTTP (WWW), FTP (file transfer), Telnet (remote login), SMTP (email)

App's using UDP:
- streaming media, teleconferencing, Internet telephony

The Network Core

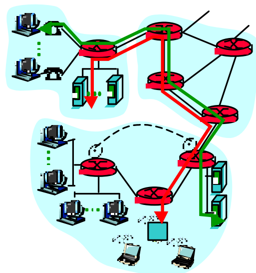
- mesh of interconnected routers
- *the* **fundamental question:** how is data transferred through net?
  - **circuit switching:** dedicated circuit per call: telephone net
  - **packet-switching:** data sent thru net in discrete "chunks"



Network Core: Circuit Switching

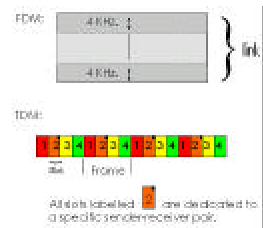
End-end resources reserved for "call"

- link bandwidth, switch capacity
- dedicated resources: no sharing
- circuit-like (guaranteed) performance
- call setup required



Network Core: Circuit Switching

- network resources (e.g., bandwidth) **divided into "pieces"**
- pieces allocated to calls
- resource piece *idle* if not used by owning call *(no sharing)*
- dividing link bandwidth into "pieces"
  - frequency division
  - time division



Network Core: Packet Switching

each end-end data stream divided into *packets*

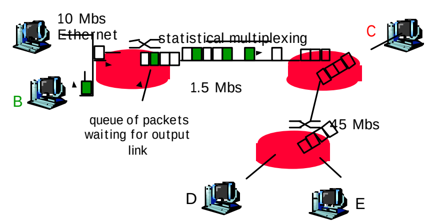
- user A, B packets *share* network resources
- each packet uses full link bandwidth
- resources used *as needed*,

Bandwidth division into "pieces"
Dedicated allocation
Resource reservation

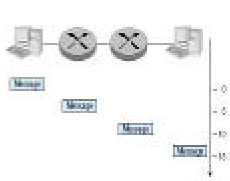
resource contention:

- aggregate resource demand can exceed amount available
- congestion: packets queue, wait for link use
- store and forward: packets move one hop at a time
  - transmit over link
  - wait turn at next link

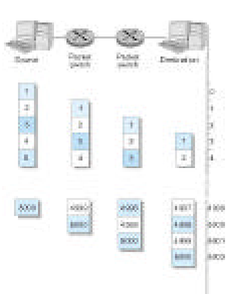
Network Core: Packet Switching



Network Core: Packet Switching



Message Switching

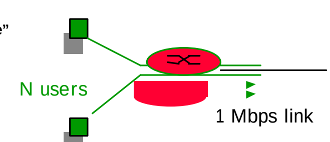


Segmented Packet Switching

Packet switching versus circuit switching

Packet switching allows more users to use network!

- 1 Mbit link
- each user:
  - 100Kbps when "active"
  - active 10% of time
- circuit-switching:
  - 10 users
- packet switching:
  - with 35 users, probability > 10 active less that .004



Packet switching versus circuit switching

Is packet switching the "ultimate solution"

- Great for bursty data
  - resource sharing
  - no call setup
- **Excessive congestion:** packet delay and loss
  - protocols needed for reliable data transfer, congestion control
- **Q: How to provide circuit-like behavior?**
  - bandwidth guarantees needed for audio/video apps

still an unsolved problem

Packet-switched networks: routing

- ***Goal:*** move packets among routers from source to destination
- **datagram network:**
  - *destination address* determines next hop
  - routes may change during session
  - analogy: driving, asking directions
- **virtual circuit network:**
  - each packet carries tag (virtual circuit ID), tag determines next hop
  - fixed path determined at *call setup time*, remains fixed thru call
  - routers maintain per-call state

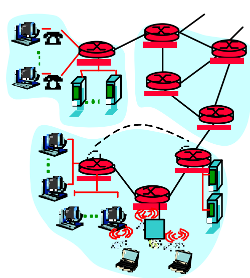
Access networks and physical media

Q: How to end systems connect to an edge router?

- residential access nets
- institutional access networks (school, company)
- mobile access networks

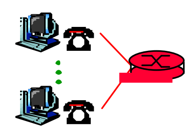
Keep in mind:

- bandwidth (bits per second) of access network?
- shared or dedicated?



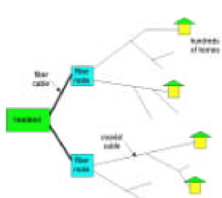
Residential access: point to point access

- **Dialup via modem**
  - up to 56Kbps direct access to router (conceptually)
- **ISDN:** integrated services digital network: 128Kbps all-digital connect to router
- **ADSL:** asymmetric digital subscriber line
  - up to 1 Mbps home-to-router
  - up to 8 Mbps router-to-home



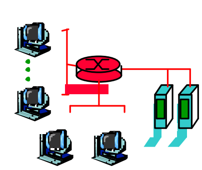
Residential access: cable modems

- **HFC: hybrid fiber coax**
  - asymmetric: up to 10Mbps upstream, 1 Mbps downstream
- **network** of cable and fiber attaches homes to ISP router
  - shared access to router among home
  - issues: congestion, dimensioning
- deployment: available via cable companies, e.g., MediaOne



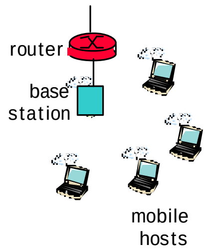
Institutional access: local area networks

- company/univ **local area network** (LAN) connects end system to edge router
- **Ethernet:**
  - shared or dedicated cable connects end system and router
  - 10 Mbs, 100Mbps, Gigabit Ethernet
- **deployment:** institutions, home LANs soon



Wireless access networks

- shared ***wireless*** **access network connects end system to router**
- **wireless LANs:**
  - radio spectrum replaces wire
  - e.g., Lucent Wavelan 10 Mbps
- **wider-area wireless access**
  - CDPD: wireless access to ISP router via cellular network



Physical Media

- **physical link:** transmitted data bit propagates across link
- **guided media:**
  - signals propagate in solid media: copper, fiber
- **unguided media:**
  - signals propagate freelye.g., radio

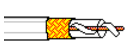
Twisted Pair (TP)

- **two insulated copper wires**
  - Category 3: traditional phone wires, 10 Mbps ethernet
  - Category 5 TP: 100Mbps ethernet

Physical Media: coax, fiber

Coaxial cable:

- wire (signal carrier) within a wire (shield)
  - baseband: single channel on cable
  - broadband: multiple channels on cable
- bidirectional
- common use in 10Mbs Ethernet



Fiber optic cable:

- glass fiber carrying light pulses
- high-speed operation:
  - 100Mbps Ethernet
  - high-speed point-to-point transmission (e.g., 5 Gbps)
- low error rate

Physical media: radio

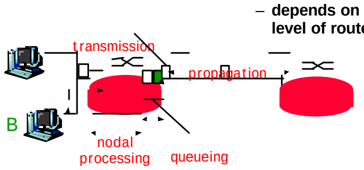
- signal carried in electromagnetic spectrum
- no physical "wire"
- bidirectional
- propagation environment effects:
  - reflection
  - obstruction by objects
  - interference

Radio link types:

- **microwave**
  - e.g. up to 45 Mbps channels
- **LAN** (e.g., waveLAN)
  - 2Mbps, 11Mbps
- **wide-area** (e.g., cellular)
  - e.g. CDPD, 10's Kbps
- **satellite**
  - up to 50Mbps channel (or multiple smaller channels)
  - 270 Msec end-end delay
  - geosynchronous versus LEOS

Delay in packet-switched networks

- packets experience **delay** on end-to-end path
- **four** sources of delay at each hop
- nodal processing:
  - check bit errors
  - determine output link
- queueing
  - time waiting at output link for transmission
  - depends on congestion level of router



Delay in packet-switched networks

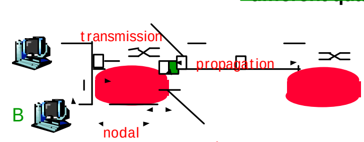
Transmission delay:

- R=link bandwidth (bps)
- L=packet length (bits)
- time to send bits into link = L/R

Propagation delay:

- d = length of physical link
- s = propagation speed in medium (~2×10^8 m/sec)
- propagation delay = d/s

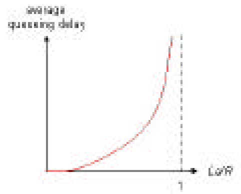
Note: s and R are *very* different quantitites!



Queuing delay (revisited)

- R=link bandwidth (bps)
- L=packet length (bits)
- a=average packet arrival rate

traffic intensity = La/R



- La/R ~ 0: average queuing delay small
- La/R <= 1: delays become large
- La/R > 1: more "work" arriving than can be serviced, average delay infinite!

Protocol "Layers"

Networks are complex!

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

Question:
Is there any hope of *organizing* structure of network?

Organization of air travel

ticket (purchase) ticket (complain)
baggage (check) baggage (claim)
gates (load) gates (unload)
runway takeoff runway landing
airplane routing airplane routing
airplane routing

- a series of steps

Organization of air travel: a different view

ticket (purchase) ticket (complain)
baggage (check) baggage (claim)
gates (load) gates (unload)
runway takeoff runway landing
airplane routing airplane routing
airplane routing

Layers: each layer implements a service
- via its own internal-layer actions
- relying on services provided by layer below

Layered air travel: services

Counter-to-counter delivery of person+bags
baggage-claim-to-baggage-claim delivery
people transfer: loading gate to arrival gate
runway-to-runway delivery of plane
airplane routing from source to destination

Distributed implementation of layer functionality

Departing airport
ticket (purchase)
baggage (check)
gates (load)
runway takeoff
airplane routing

arriving airport
ticket (complain)
baggage (claim)
gates (unload)
runway landing
airplane routing

intermediate air traffic sites
airplane routing
airplane routing
airplane routing

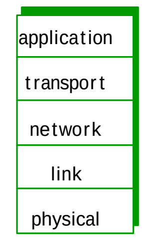
Why layering?

Dealing with complex systems:

- **explicit structure allows identification, relationship of complex system's pieces**
  - **layered reference model for discussion**
- **modularization eases maintenance, updating of system**
  - **change of implementation of layer's service transparent to rest of system**
  - **e.g., change in gate procedure doesn't affect rest of system**
- **layering considered harmful?**

Internet protocol stack

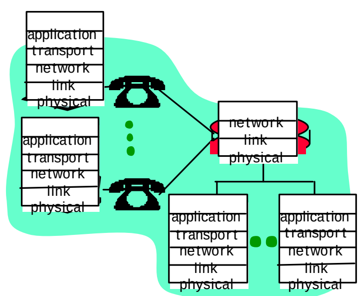
- **application: supporting network applications**
  - **ftp, smtp, http**
- **transport: process-process data transfer**
  - **tcp, udp**
- **network: routing of datagrams from source to destination**
  - **ip, routing protocols**
- **link: data transfer between neighboring network elements**
  - **ppp, ethernet**
- **physical: bits "on the wire"**



Layering: logical communication

Each layer:

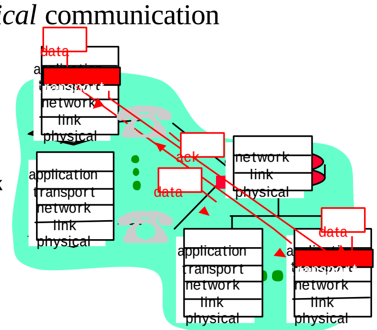
- **distributed**
- **"entities" implement layer functions at each node**
- **entities perform actions, exchange messages with peers**



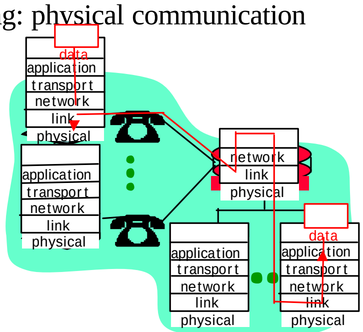
Layering: *logical* communication

E.g.: transport

- **take data from app**
- **add addressing, reliability check info to form "datagram"**
- **send datagram to peer**
- **wait for peer to ack receipt**
- **analogy: post office delivery to person**



Layering: physical communication



Protocol layering and data

Each layer takes data from above

- **adds header information to create new data unit**
- **passes new data unit to layer below**