ECE/COMPSCI 356 Computer Network Architecture

Lecture 2: Design Requirements

Neil Gong

neil.gong@duke.edu

Slides credit: Xiaowei Yang, PD

Overview

- Design requirements of a computer network
  - different applications and different communities place requirements on a computer network

Supporting diverse applications

- Most people know about the Internet (a computer network) through applications
  - Short videos
  - World Wide Web
  - Email
  - Online Social Network
  - Streaming Audio/Video
  - File Sharing
  - Instant Messaging
  - …

Different communities

- Application Programmer
  - List the services that his application needs: delay bounded delivery of data
- Network Designer
  - Design a cost-effective network with sharable resources
- Network Provider
  - List the characteristics of a system that is easy to manage

Design requirements

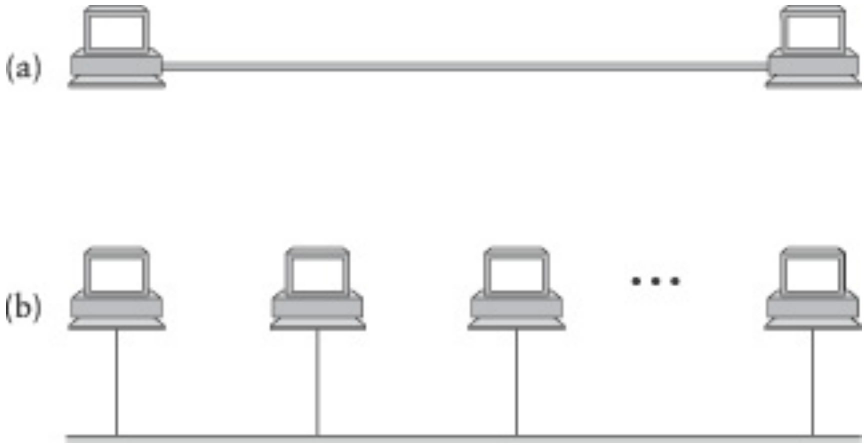
1. Scalable connectivity
2. Cost-effective resource sharing
3. Support for common services
4. Manageability
5. Security

1. Scalable connectivity

- Scale: A system is designed to grow to an arbitrary large size is said to scale
  - How to connect an arbitrary large number of computers on a network?

- How to connect two computers
  - Direct link
  - Indirect link

Direct links



Point-to-Point

Multiple-Access

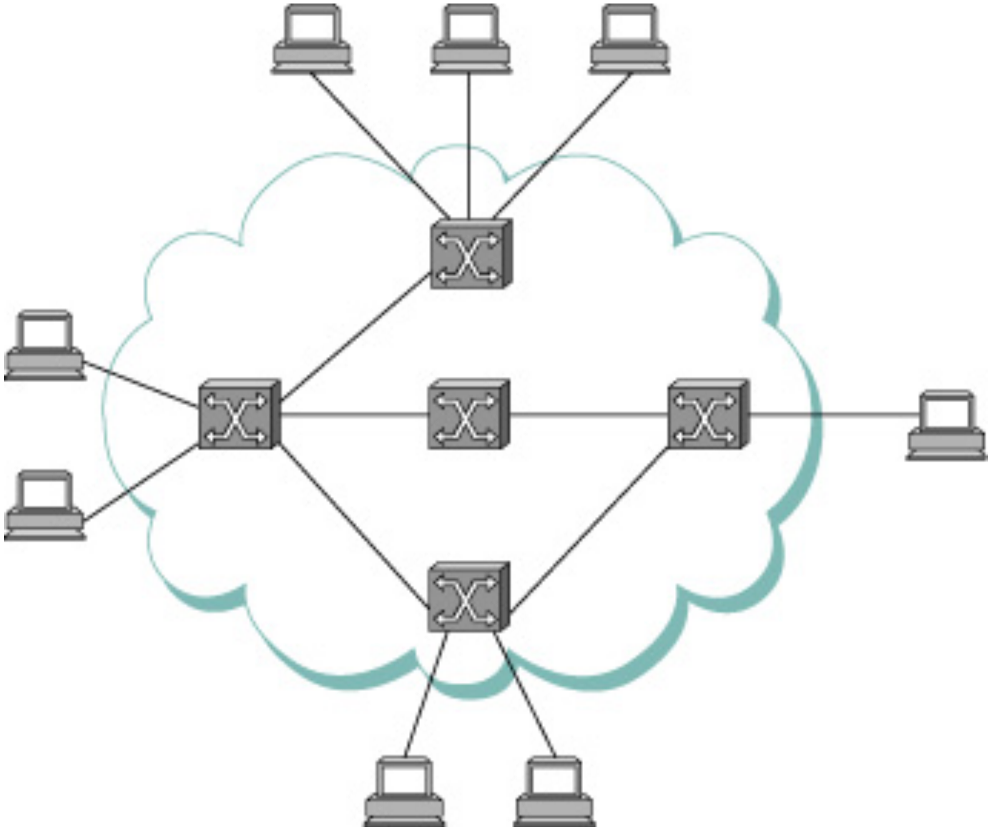
- Connect two or more computers via a physical medium or electromagnetic waves
- Computers are referred to as nodes
- The physical medium is referred to as a link

Indirect links

- Switched networks
- Interconnection of networks

Switched networks

- Achieving indirect connectivity
- Concepts
  - Switches
    - Nodes that are attached to at least two links forward data from one link to another link
  - Hosts
  - Cloud

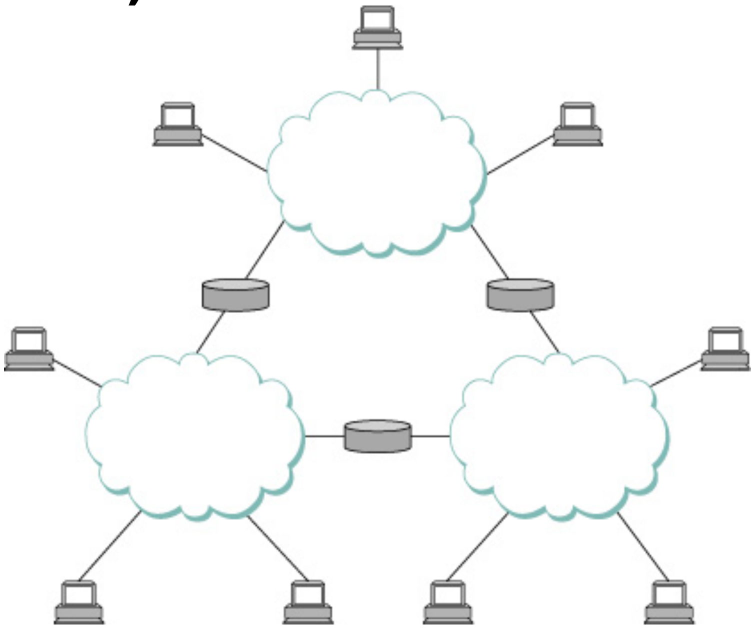


Switched networks

- Circuit switching
  - Sets up a circuit before nodes can communicate
  - Switches connect circuits on different links
- Packet switching
  - Data are split into blocks of data called packets
    - Data: message
    - Block: packet
  - Store and forward
  - Nodes send packets and switches forward them

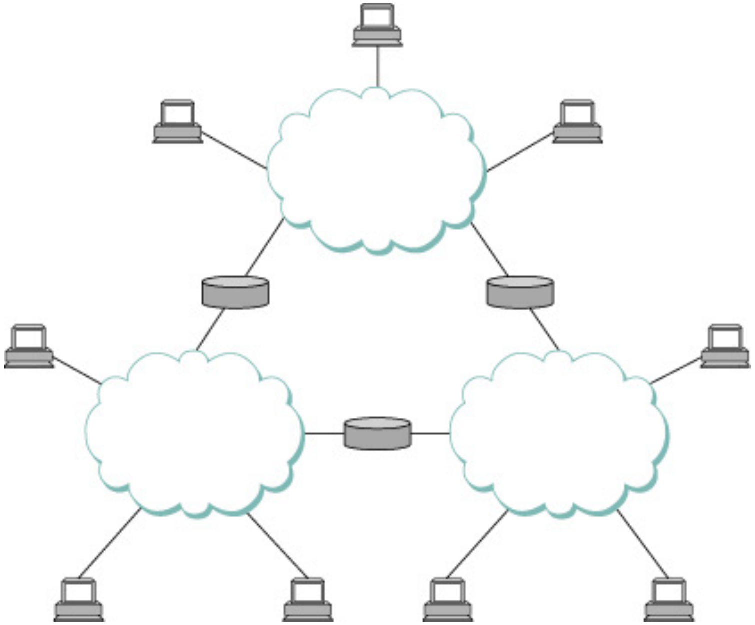
Interconnection of networks (internetwork)

- Achieving indirect connectivity
  - Each cloud is a network
- Concepts
  - internetwork
    - Also called internet
    - Different from the Internet
  - Router/gateway
    - A node that is connected to two or more networks
    - Use different protocols from switches
- Recursively build larger clouds by connecting smaller ones



Addressing and routing in internetwork

- Host-to-host connectivity
  - Indirect
- Nodes are assigned addresses
- Routing
  - Routers forward messages
- Different destinations
  - Unicast
    - One to one
  - Broadcast
    - One to all
  - Multicast
    - One to many

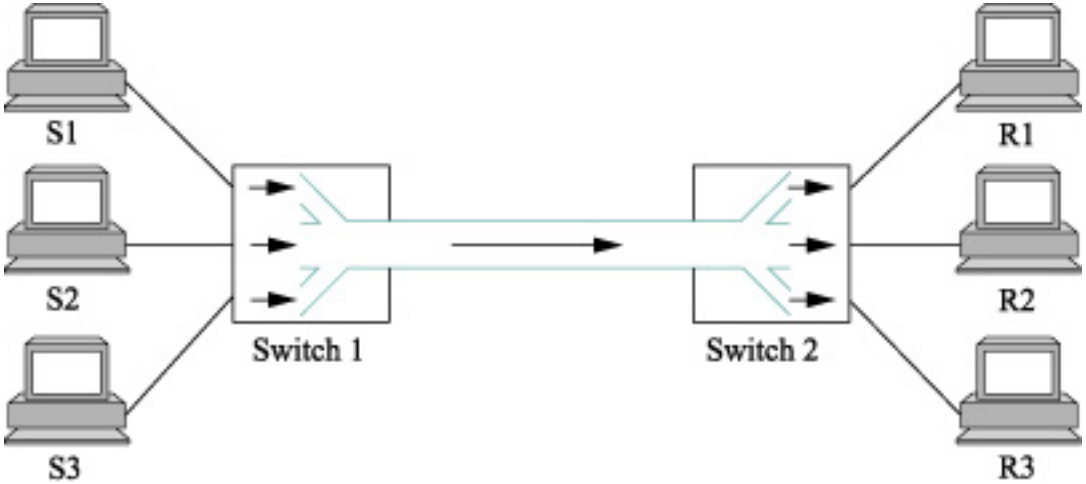


2. Cost-effective resource sharing

- Question: how do all the hosts share the network when they want to communicate with each other?
  - Use at the same time
  - Fair

- Multiplexing: a system resource is shared among multiple users
  - Analogy: CPU sharing

- Mechanisms to multiplexing
  - Time-division multiplexing (TDM)
  - Frequency-division multiplexing (FDM)
  - Statistical multiplexing



Multiplex

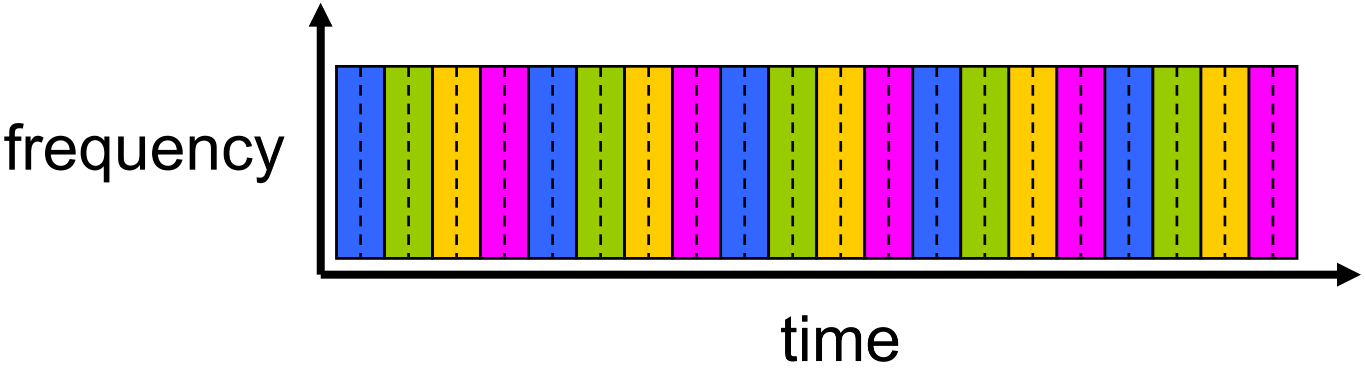
Demultiplex

TDM and FDM

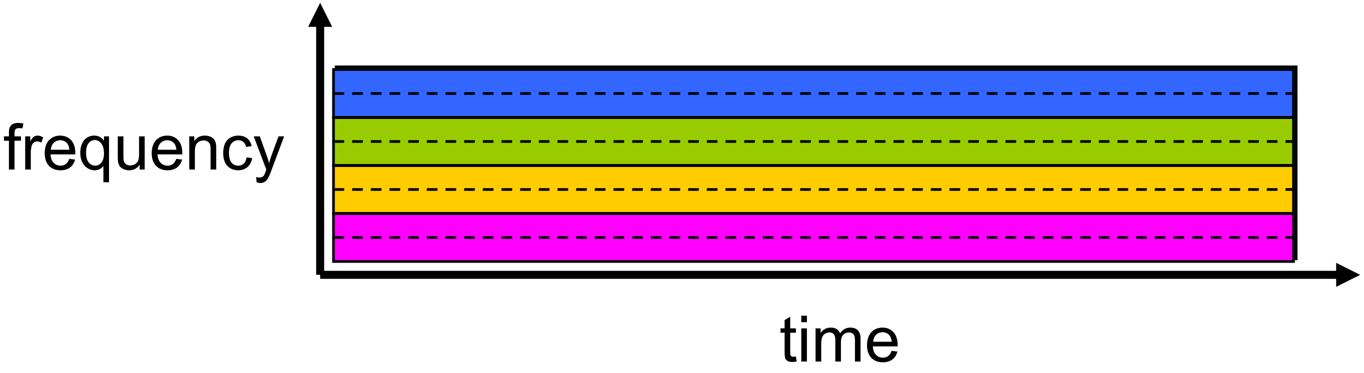
Example:

4 users 🟦🟩🟨🟪

TDM



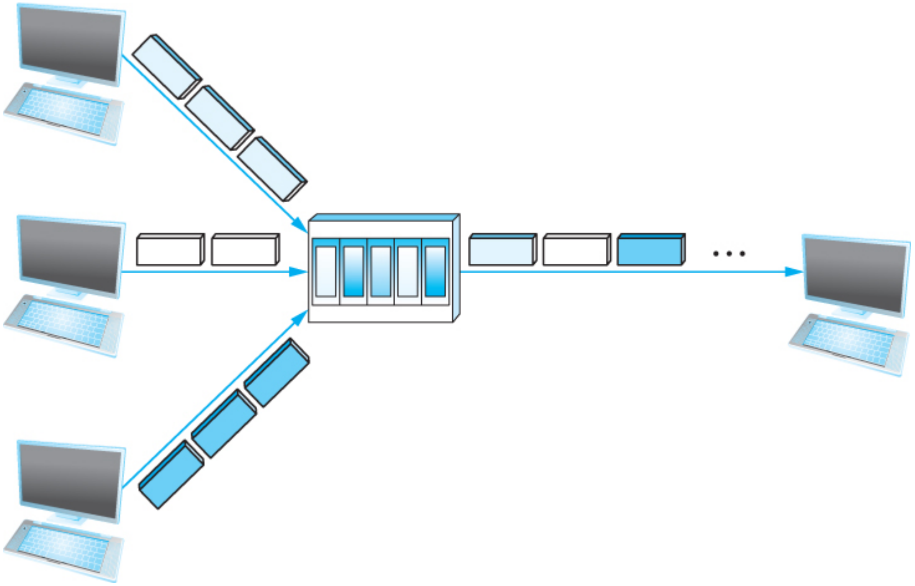
FDM



Problems with FDM and TDM

- What if a user does not have data to send all the time (Over-provision)?
  - Consider web browsing
  - → Inefficient use of resources
- Max # of flows is fixed and known ahead of time (Under-provision)
  - Not practical to change the size of quantum or add additional quanta for TDM
  - Nor add more frequencies in FDM
- Often used in circuit switching networks
  - Telephone network, cellular network

Statistical Multiplexing



- The physical link is shared over time (like TDM)
- But no fixed pattern
  - No predetermined slots
  - Packets are sent on demand

Pros and Cons

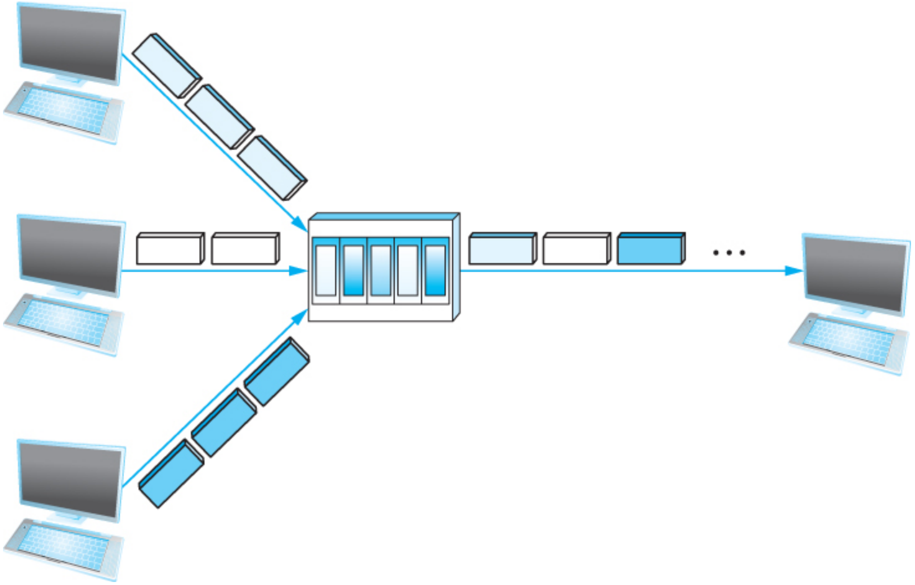
- Assumption: traffic is largely bursty
- Pros: Resources are not wasted when hosts are idle
- Cons: No guarantee hosts would have their turns to transmit
- Some possible fixes:
  - Divide message to packets
  - Limit maximum packet size
  - Fair scheduling of packets for transmission

Packets

- Divide an application message into blocks of data → packets

- Limit maximum packet size
  - packets sent on demand
  - Must give each application/host its turn to send

Packet scheduling



- Scheduling: which packet to send
  - First come first serve (FIFQ)
  - Analogy to processing scheduling in operating systems

3. Support for common services

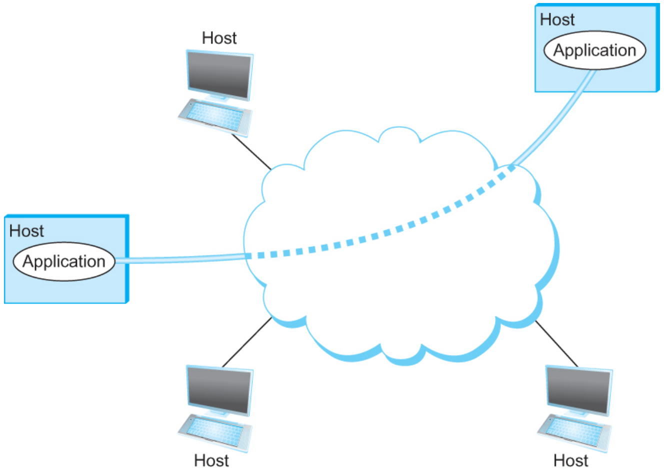
- Application developers want a network to provide services that make application programs communicate with each other, not just sending packets
  - E.g. reliably delivering an email message from a sender to a receiver

- Many complicated things need to happen
  - Can you name a few?

- Design choices
  - Application developers build all functions they need
  - Network provides common services → a layered network architecture
    - Build it once, and shared many times

3. Support for Common Services

- Provide services that make application programs communicate with each other, not just send packets
- Logical Channels
  – Application-to-Application communication path or a pipe



Process communicating over an abstract channel

Reliability

- Network should hide the errors
- Bits are lost
  - Bit errors (1 to a 0, and vice versa)
  - Burst errors – several consecutive errors
- Packets are lost
- Links and node failures
- Messages are delayed
- Messages are delivered out-of-order

4. Manageability

- Manage the network as it grows
- When things go wrong
  - Easily locate and isolate the faults
- An open research challenge
  - Datacenter networks
  - Software defined networking

5. Security

- Security is a big concern
  - Denial of service attacks
  - Data breaches
  - Attacks to HTTP protocols
  - …

- Security requirement was largely ignored when designing computer networks

Design requirements

1. Scalable connectivity
2. Cost-effective resource sharing
3. Support for common services
4. Manageability
5. Security