# Introduction

Krishna Gummadi, Peter Druschel
MPI for Software Systems

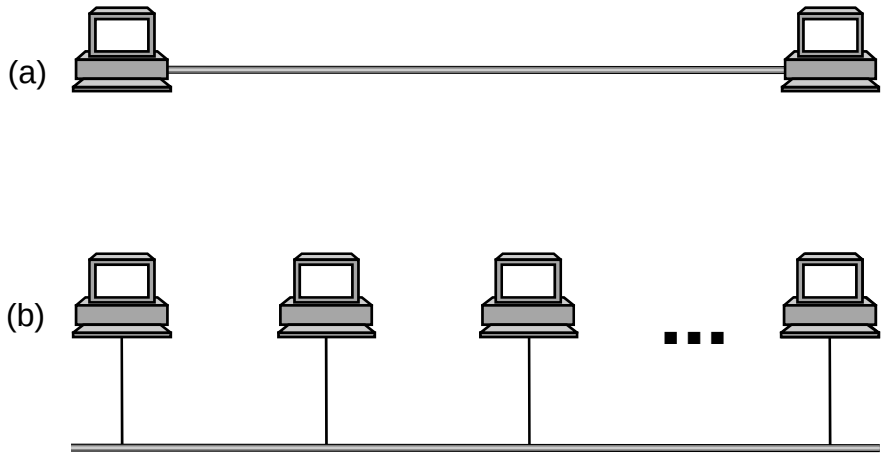
Thanks to Larry Peterson (Princeton), Eugene Ng (Rice)

# Questions

- What is a network / inter-network?
- What is the purpose of a network?
- How do we identify who we want to talk to?
- How do we get data through the network?
- How do we make it efficient?
- What can go wrong?
- How do we manage the complexity?
- How do we make it secure?
- How do we make it usable?
- How do we measure performance?

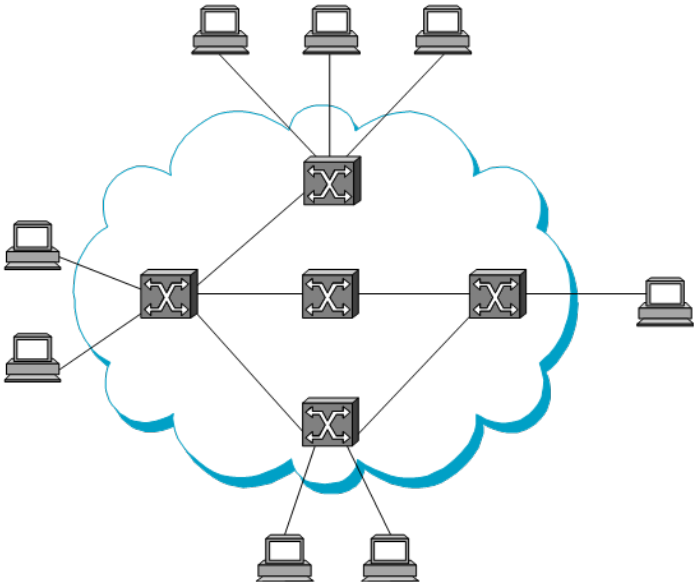
# Network building blocks

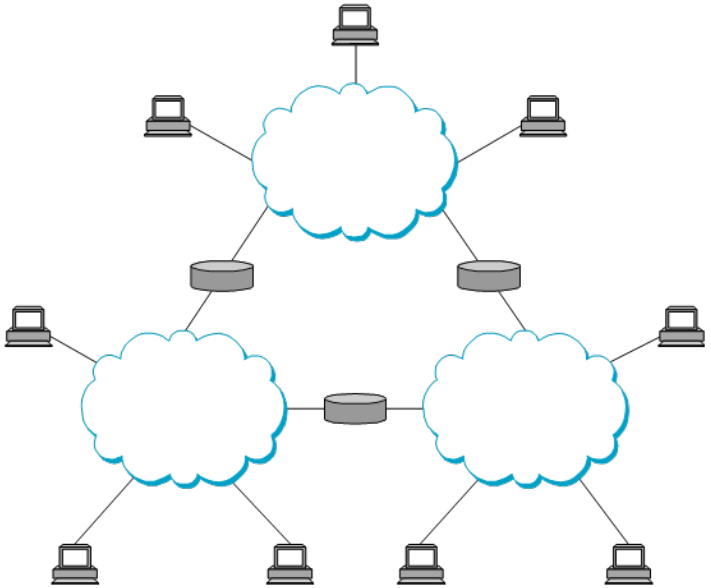
- Nodes: PC, devices, special-purpose routers…
  - hosts
  - switches
- Links: twisted pair, optical fiber, wireless radio…
  - point-to-point
  - multiple access



# Switched networks

- A network can be defined recursively as...
  - two or more nodes connected by a link, or
  - two or more networks connected by a node





# Questions

- What is a network / inter-network?
- What is the purpose of a network?
- How do we identify who we want to talk to?
- How do we get data through the network?
- How do we make it efficient?
- What can go wrong?
- How do we manage the complexity?
- How do we make it secure?
- How do we make it usable?
- How do we measure performance?

# Purpose

- Communication
- Coordination
- Sharing of resources and information
- Content distribution
- Data collection

# Questions

- What is a network / inter-network?
- What is the purpose of a network?
- How do we identify who we want to talk to?
- How do we get data through the network?
- How do we make it efficient?
- What can go wrong?
- How do we manage the complexity?
- How do we make it secure?
- How do we make it usable?
- How do we measure performance?

# Addressing

- Address: byte-string that identifies a node
  - usually unique
- Types of addresses
  - unicast: node-specific
  - anycast: some node in the network
  - broadcast: all nodes on the network
  - multicast: some subset of nodes on the network
- Structure
  - Hierarchical: address indicates location/organization
  - Flat: address does not reveal any info other than identity

# Naming

Human-readable identifier

- http://www.mpi-sws.org (WWW)
- peter.druschel (Skype)
- druschel@mpi-sws.mpg.de (Email)

# Questions

- What is a network / inter-network?
- What is the purpose of a network?
- How do we identify who we want to talk to?
- How do we get data through the network?
- How do we make it efficient?
- What can go wrong?
- How do we manage the complexity?
- How do we make it secure?
- How do we make it usable?
- How do we measure performance?

# Switching and routing

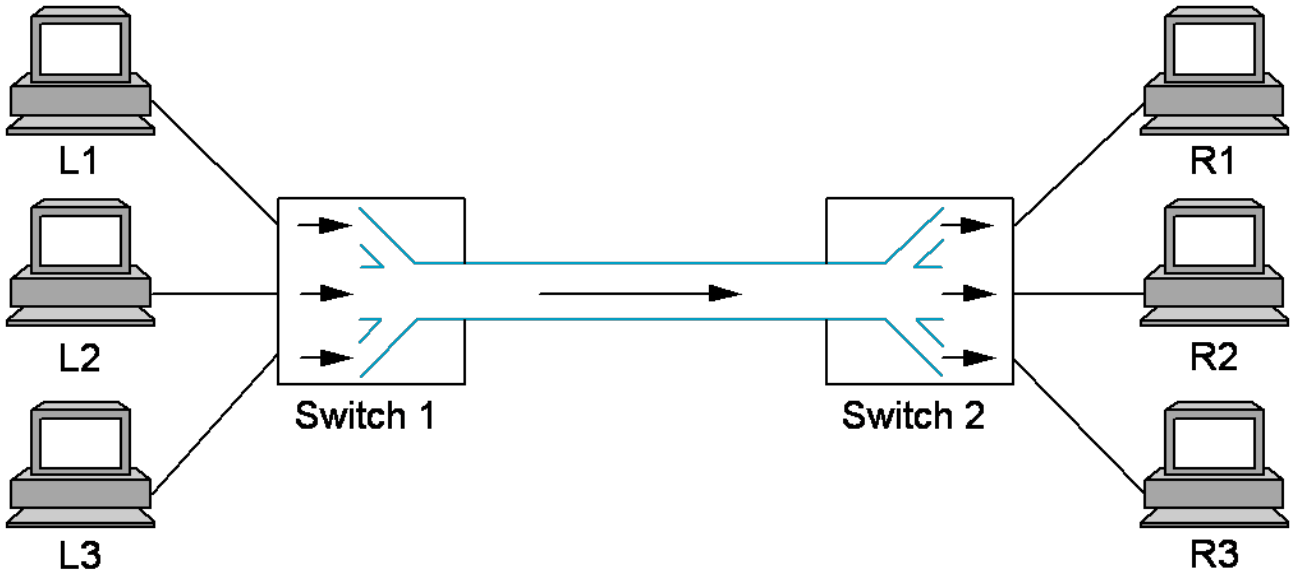
- Circuit switching: carry bit streams
  - telephone network
- Packet switching: store-and-forward messages
  - Internet
- Virtual circuit switching: packets along circuits
  - ATM
- Routing: deciding how to forward data toward the destination node(s) based on an address

# Questions

- What is a network / inter-network?
- What is the purpose of a network?
- How do we identify who we want to talk to?
- How do we get data through the network?
- How do we make it efficient?
- What can go wrong?
- How do we manage the complexity?
- How do we make it secure?
- How do we make it usable?
- How do we measure performance?

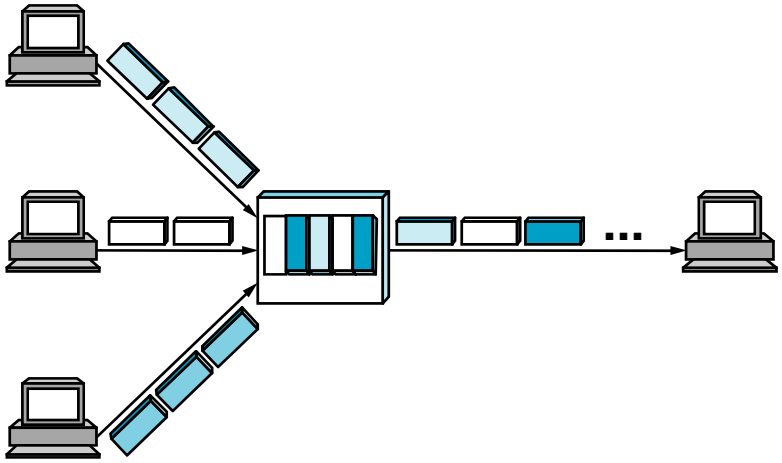
# Multiplexing

- Time-Division Multiplexing (TDM)
- Frequency-Division Multiplexing (FDM)
- Code-Division Multiplexing (CDM)



# Statistical Multiplexing

- On-demand time-division
- Schedule link on a per-*packet* basis
- Packets from different sources interleaved on link
- Buffer packets that are *contending* for the link
- Buffer (queue) overflow is called *congestion*



# Questions

- What is a network / inter-network?
- What is the purpose of a network?
- How do we identify who we want to talk to?
- How do we get data through the network?
- How do we make it efficient?
- What can go wrong?
- How do we manage the complexity?
- How do we make it secure?
- How do we make it usable?
- How do we measure performance?

# What Goes Wrong in the Network?

- Bit-level errors (e.g. electrical interference)
- Packet-level errors (congestion)
- Link and node failures

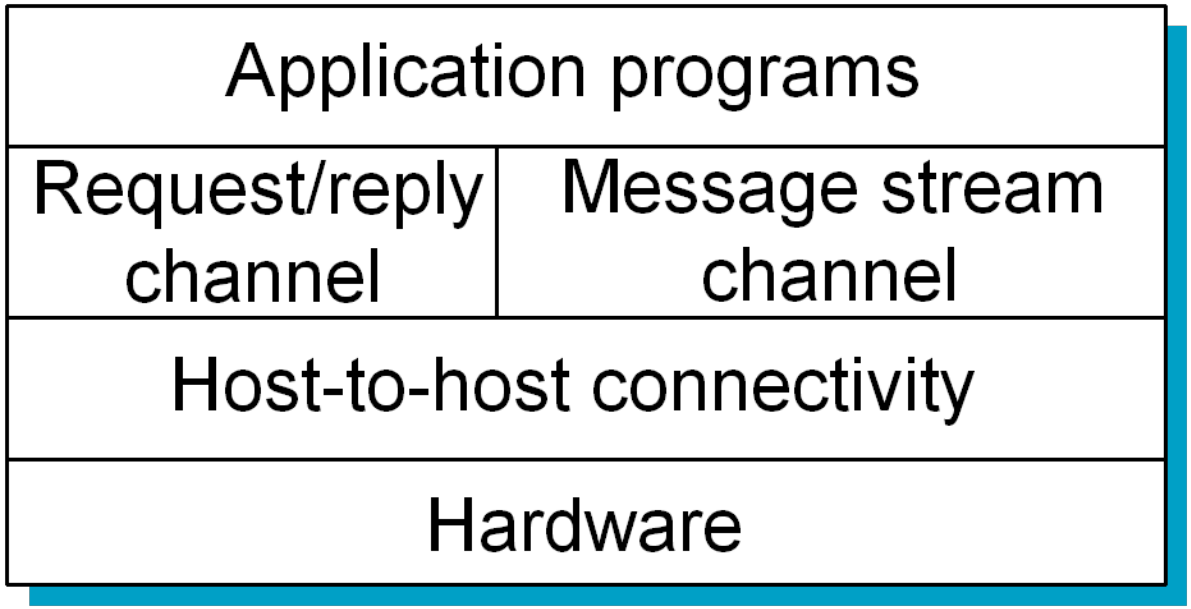
- Packets are delayed
- Packets are delivered out-of-order
- Third parties eavesdrop, manipulate, fabricate

# Questions

- What is a network / inter-network?
- What is the purpose of a network?
- How do we identify who we want to talk to?
- How do we get data through the network?
- How do we make it efficient?
- What can go wrong?
- How do we manage the complexity?
- How do we make it secure?
- How do we make it usable?
- How do we measure performance?

# Layering

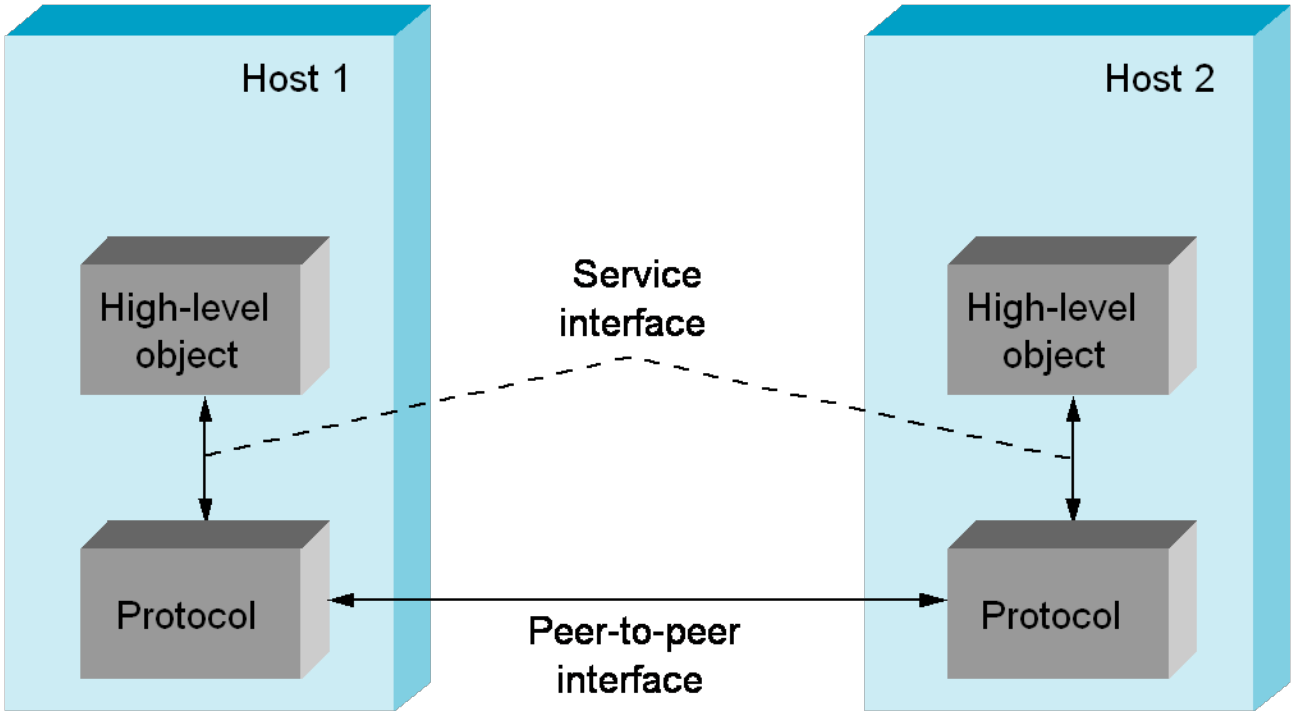
- Use abstractions to hide complexity
- Abstraction naturally leads to layering
- Alternative abstractions at each layer



# Protocols

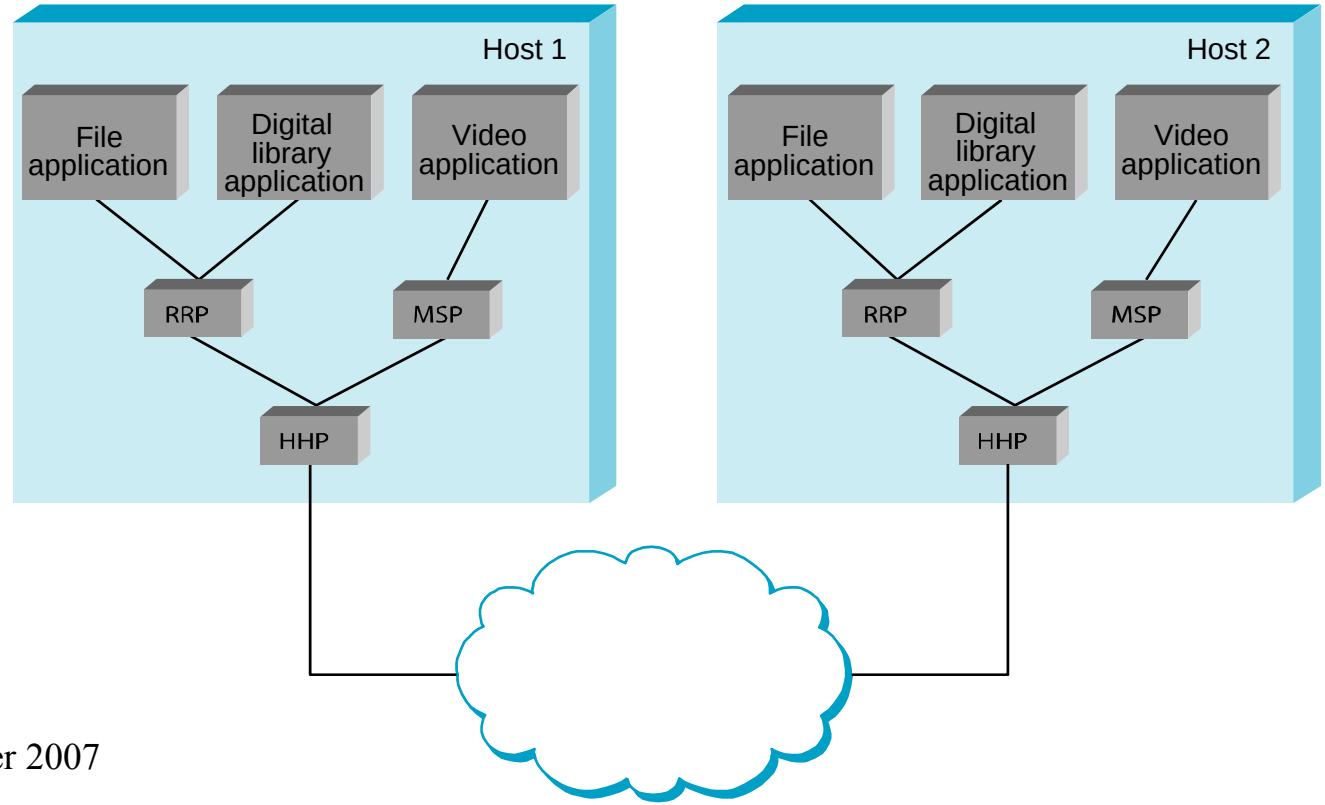
- Building blocks of a network architecture
- Each protocol object has two different interfaces
  - *service interface*: operations on this protocol
  - *peer-to-peer interface*: messages exchanged with peer
- Term "protocol" is overloaded
  - specification of peer-to-peer interface
  - module that implements this interface

# Interfaces


Host 1
Host 2
High-level object
High-level object
Service interface
Protocol
Protocol
Peer-to-peer interface

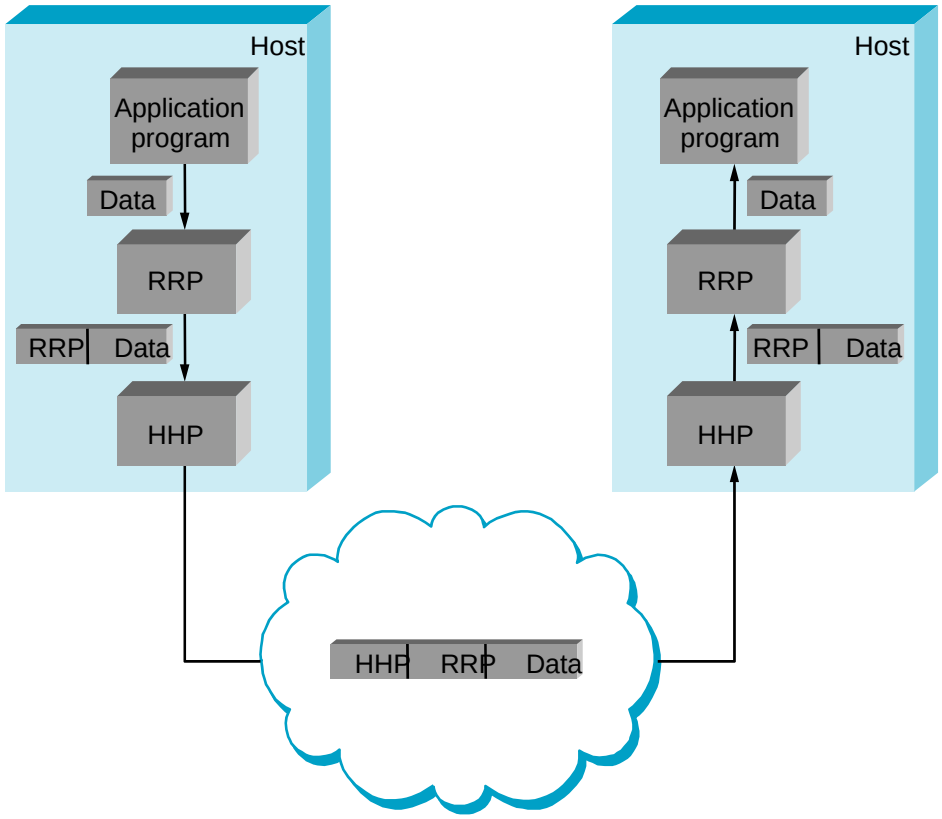
# Protocol Machinery

- Protocol Graph
  - most peer-to-peer communication is indirect
  - peer-to-peer is direct only at hardware level



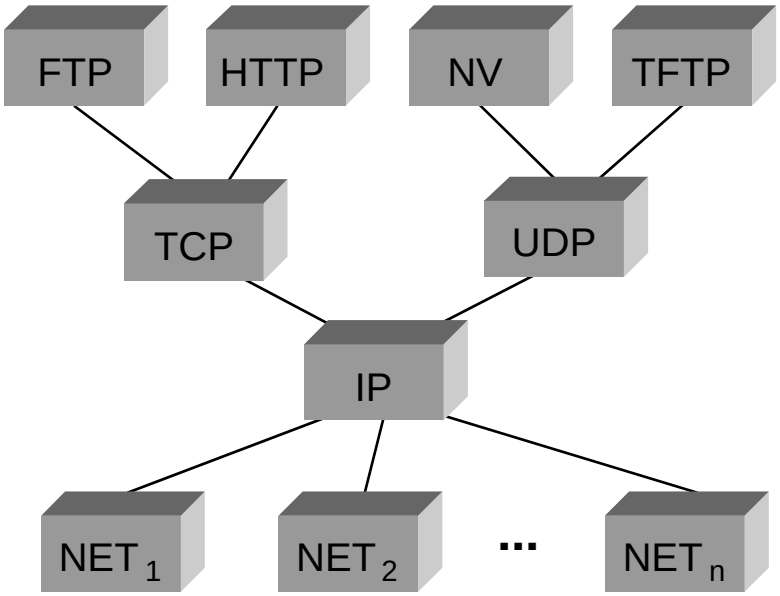
# Machinery (cont)

- Multiplexing and Demultiplexing (demux key)
- Encapsulation (header/body)

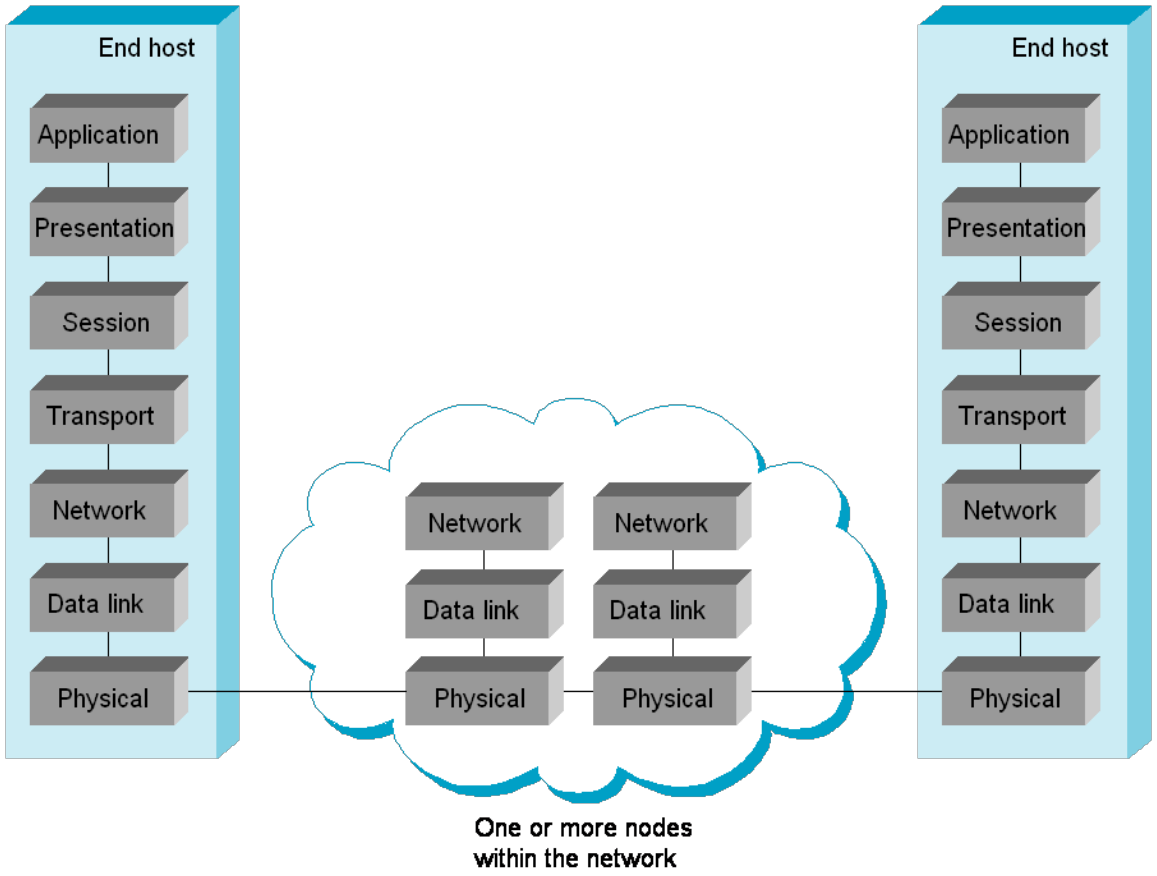


# Internet Architecture

- Defined by Internet Engineering Task Force (IETF)
- Hourglass Design
- Application vs Application Protocol (FTP, HTTP)



# ISO Architecture



# Questions

- What is a network / inter-network?
- What is the purpose of a network?
- How do we identify who we want to talk to?
- How do we get data through the network?
- How do we make it efficient?
- What can go wrong?
- How do we manage the complexity?
- How do we make it secure?
- How do we make it usable?
- How do we measure performance?

# Network security

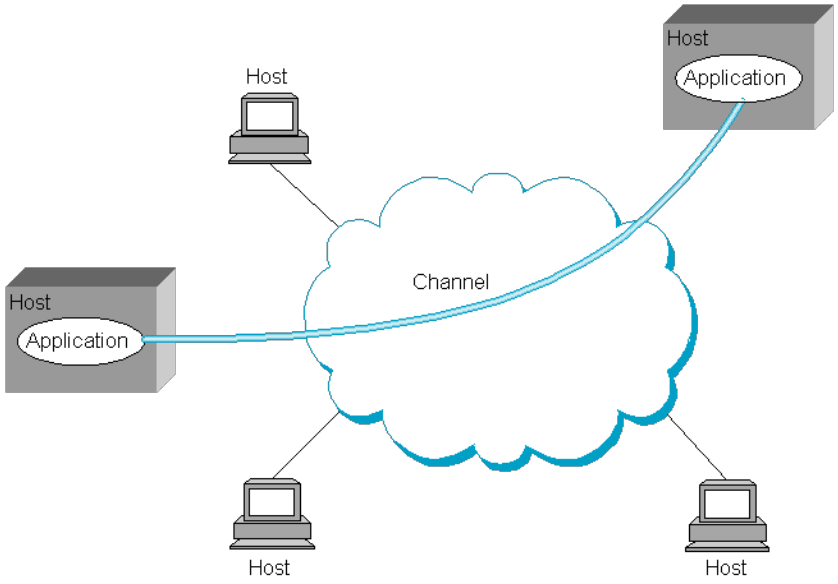
- Authentication: who am I talking to?
- Integrity: do I get the bits the other party sent?
- Privacy: can anyone else see what we say?
- Availability: can we communicate?
- Anonymity: can I hide the fact that I am the sender/receiver?
- Non-repudiation: Can someone deny that he said something he actually said?

# Questions

- What is a network / inter-network?
- What is the purpose of a network?
- How do we identify who we want to talk to?
- How do we get data through the network?
- How do we make it efficient?
- What can go wrong?
- How do we manage the complexity?
- How do we make it secure?
- How do we make it usable?
- How do we measure performance?

# Inter-Process Communication

- Turn host-to-host connectivity into process-to-process communication.
- Fill  gap between what applications expect and what the underlying technology provides.



# IPC Abstractions

- Request/Reply
  - distributed file systems
  - digital libraries (web)
- Stream-Based
  - video: sequence of frames
    - 1/4 NTSC = 352x240 pixels
    - (352 x 240 x 24)/8=247.5KB
    - 30 fps = 7500KBps = 60Mbps
  - video applications
    - on-demand video
    - video conferencing

# Questions

- What is a network / inter-network?
- What is the purpose of a network?
- How do we identify who we want to talk to?
- How do we get data through the network?
- How do we make it efficient?
- What can go wrong?
- How do we manage the complexity?
- How do we make it secure?
- How do we make it usable?
- How do we measure performance?

# Performance Metrics

- Bandwidth (throughput)
  - data transmitted per time unit
  - link versus end-to-end
  - notation
    - KB = $2^{10}$ bytes
    - Mbps = $10^6$ bits per second
- Latency (delay)
  - time to send message from point A to point B
  - one-way versus round-trip time (RTT)
  - components

    Latency = Propagation + Transmit + Queue

    Propagation = Distance / c

    Transmit = Size / Bandwidth

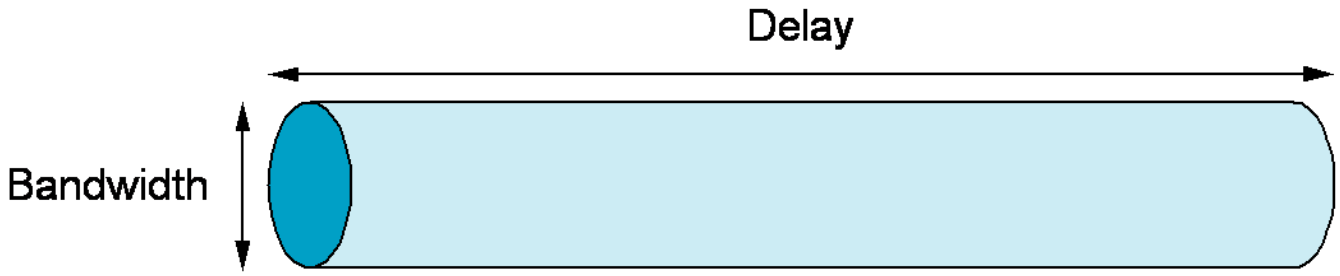
# Bandwidth versus Latency

- Relative importance
  - 1-byte: 1ms vs 100ms dominates 1Mbps vs 100Mbps
  - 25MB: 1Mbps vs 100Mbps dominates 1ms vs 100ms

- Infinite bandwidth
  - RTT dominates
    - Throughput = TransferSize / TransferTime
    - TransferTime = RTT + 1/Bandwidth x TransferSize
  - 1-MB *file* to 1-Gbps link as 1-KB *packet* to 1-Mbps link

# Delay x Bandwidth Product

- Amount of data “in flight” or “in the pipe”
- Usually relative to RTT
- Example: 100ms x 45Mbps = 560KB



# Expectations

Course IS about…

- Principles and concepts
- General-purpose networks
- Internet
- Software
- Design

Course IS NOT about…

- Survey of standards
- Specialized networks
- OSI reference model
- Hardware
- Modeling