Imperial College London

Computer Networks and Distributed Systems

Part 1 - Introduction

Course 527 – Spring Term 2015-2016

Emil C Lupu & Daniele Sgandurra

e.c.lupu@imperial.ac.uk, d.sgandurra@imperial.ac.uk

Course Structure

1st half: Computer Networks - covers basic principles of networking through examples of real technology

2nd half: Distributed Systems – covers basic distributed systems architectures, remote (object) interactions, remote procedure calls, security

Whilst networks are concerned with communicating data from one endpoint to another (or several) distributed systems help us design systems whose components are hosted on different computers.

Course Structure

2 lectures + 1 tutorial each week

1 coursework

Electronic handouts available from CATE

Please ask questions!

Acknowledgements:

- Computer Neworks based on material by Peter Pietzuch, Dan Chalmers and Ian Harries
- Distributed Systems based on material by Morris Sloman

Course Attendance

B.Eng & M.Eng Electronic and Information Engineering
2nd year Required

B.Eng & M.Eng Mathematics and Computer Science
3rd Year Selective

M.Sc Computing Science Selective

Recommended Books

"Computer Networks", Andrew S. Tanenbaum, Prentice Hall, 2005 (5th Edition)

- Main reference and worth reading

"Distributed Systems: Concepts and Design", George Coulouris, Jean Dollimore, Tim Kindberg, Addison-Wesley, 2005 (5th Edition)

IEEE, IETF, ITU, OSI and W3C standards form basis of much of the material, but not designed as tutorials

Exam Questions

Not about low-level details

- Q: "What's the 50th bit in the IP packet header?"
- A: "It's the 'Don't Fragment' Flag"

But rather about principles and design trade-offs

- Q: "You need to design a transport layer for a network with the following characteristics… How would you do this?"
- A: "I'd use a reliable transport service, similar to TCP, because…"

Always explain your reasoning!

Some (simple) maths involved

Part 1: Computer Networks Introduction and Overview

Computer Networks Help us Answer Some Questions

How do I get bits down a wire?

How many computers can be connected to an Ethernet LAN?

How do we provide network connectivity to a laptop that moves about?

Why does it matter whether I use `java.net.Socket` or `java.net.DatagramSocket` when programming?

How do Windows PCs, Linux boxes and Macs communicate on the Internet?

Why is the Internet sometimes so slow?

Syllabus Overview

Introduce networking concepts and terminology
- Introduce OSI and TCP/IP engineering models
- Course loosely follows OSI Reference Model

Describe basic network standards and protocols
- Learn how design choices affect network behaviour

Describe how networks inter-connect

Illustrate how networks interact with applications

Part 1 – Contents

Basic terminology

Network types

Network topologies

Network protocol standards
- OSI Seven Layer Network Model
- TCP/IP Internet Model

Information and Data

Information
- Stimuli that have meaning in some context for receiver

Data
- Information translated into form more convenient to move or process by computer

Channel
- Path through which signals can flow

Network
- Graph of devices interconnected by channels

Node
- Device on network graph
- May refer to end-point (e.g. computer) or communications device (e.g. router)

Network Metrics

Bandwidth
- Data transferred per unit time (usually bits/second)

Delay or **Latency**
- Time a bit takes to get from source to destination (seconds)

Jitter
- Variation in delay (usually % of delay or value +/- seconds)

Loss
- Rate of loss of units of transfer (percentage, unit depends on what is being lost)

Bandwidth

Careful! Bandwidth also technical (EE) term

- Measure of frequency range of analogue channel

(Informally) used for **channel capacity**

- How much data can be sent through a channel?
- Refers to **transmission rate** (throughput)
- "This is a high bandwidth connection."

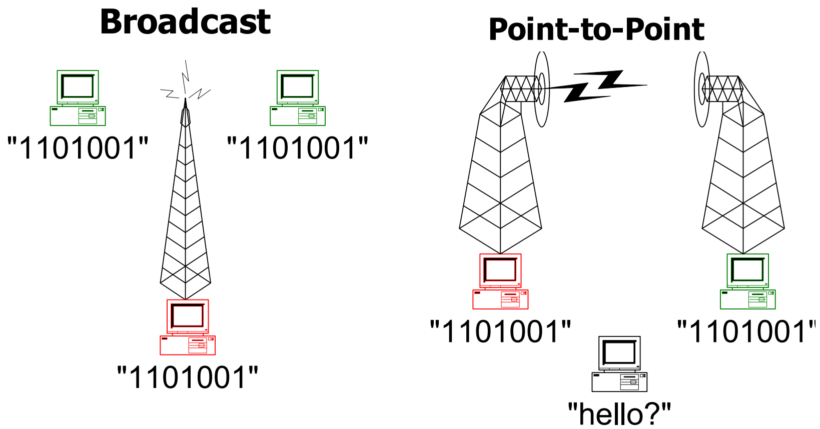
Classes of Communication

Many ways to describe a network

- Wires (or media) that form channels
- Behaviour of channels
- Range in physical and organisational terms
- Needs and capabilities of nodes

We need models to describe diverse networks

Types of Connections



From Connections to Networks

Most networks have >2 devices that connect dynamically

Individual wires between each pair of computers

- Simple but clearly not scalable

Shared wires between computers

- Only listen to messages addressed to you

Larger networks by having switches make dynamic connections over shared pool of channels

We'll come back to this later…

Types of Networks

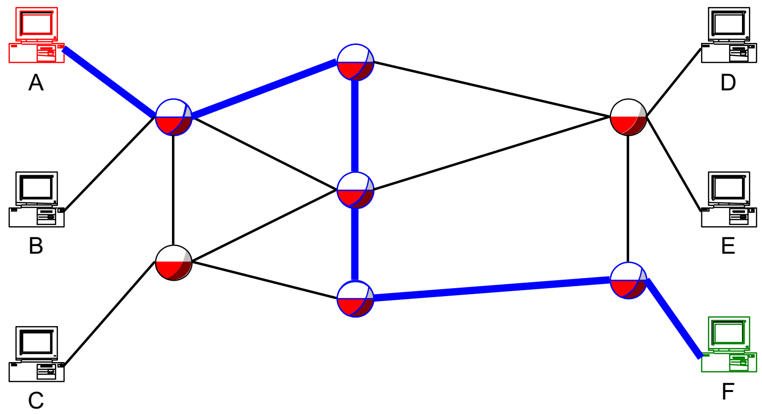
We now examine:

- Two forms of switch operation for networks
- Two types of service that networks can provide

Each valid but offer different behaviour

- Compare telephone network vs. computer network

Circuit Switching (CS)



Circuit Switching Features

One maintained path (**circuit**) (e.g. telephone call)

Three phases:

1. Circuit establishment
2. Data transfer
3. Circuit disconnection

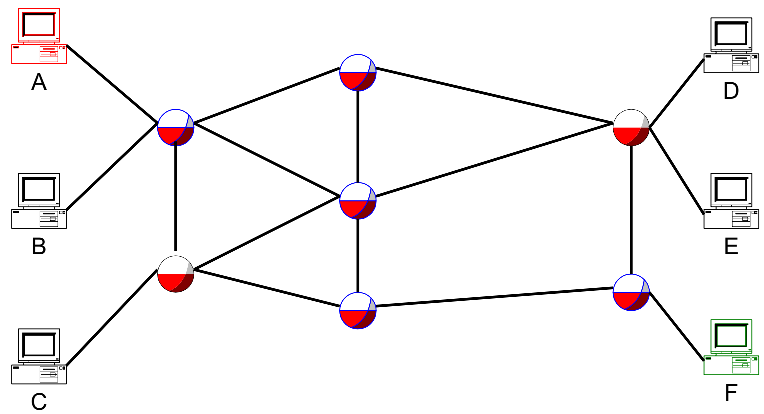
Overhead for call set-up, no overhead for use

Provides guaranteed resources

Connection breaks if any link or switch on route fails

Charging typically by time

Packet Switching (PS)



Packet Switching Features

Route calculated for each **packet** (e.g. postal service)
- Packets may arrive out of order
- Switches may store and forward packets

All data has addressing and control overhead
- But no initial overhead

Usually no guaranteed resources

Failures accommodated transparently
- Different routes may have different properties
- Packets may be lost/retransmitted due to failure

Charging typically by packet

Circuit Switching vs. Packet Switching

Circuit Switching
- Fixed bandwidth
- Unused bandwidth wasted
- Call set-up required
- Congestion may occur at call set-up (arrival rate = transmission rate)
- Overhead on call setup only
- In-order delivery
- Circuit fails if any link or switch fails

Packet Switching
- Variable bandwidth
- Uses only bandwidth required
- No call set-up
- Congestion may occur on any packet (causing delay and reordering)
- Overhead on every packet
- Out-of-order delivery
- New route found if any link or switch fails (some data may be lost)

Types of Connection Service

Network provides **connection service** to programs
- May be **connectionless** or **connection-oriented**

Uses underlying network to achieve this
- Network may be PS or CS
- Network doesn't determine service type provided
  - Software can add behaviour

Connectionless Service (CL)

No conceptual connection or maintained route

Unit of connection is **datagram** (packet)

No guarantee of order

Packet switched networks provide pure CL service
- Packets addressed by destination and routed accordingly
- Each packet handled separately
- No state at switches or set-up/tear-down calls

Connection-Oriented Service (CO)

Connection maintained between end-points

Unit of connection is the **circuit**

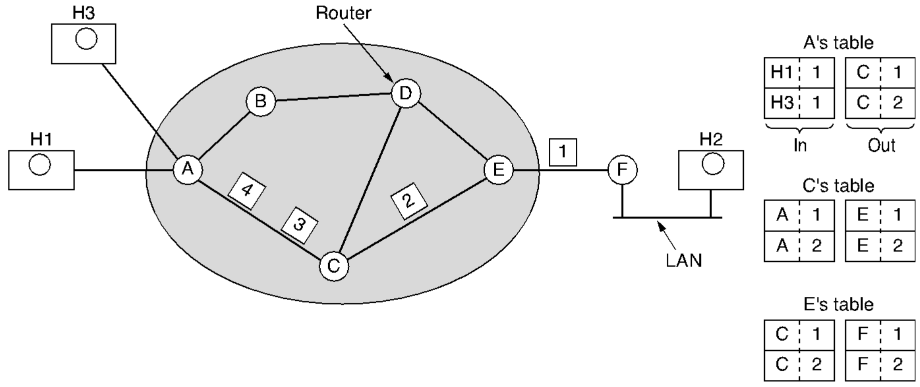
Order is preserved

Circuit switched networks provide pure CO service

– Circuit defines destination and route

Packet switched networks can provide CO service by using **virtual circuits**

Virtual Circuits I



Establish route using connection set-up packet

Soft connection using circuit

– Route packets by **circuit identifier**

– Each packet includes circuit identifier in short header

Virtual Circuits II

Less routing overhead per packet than PS

– Need to maintain circuit info at switches

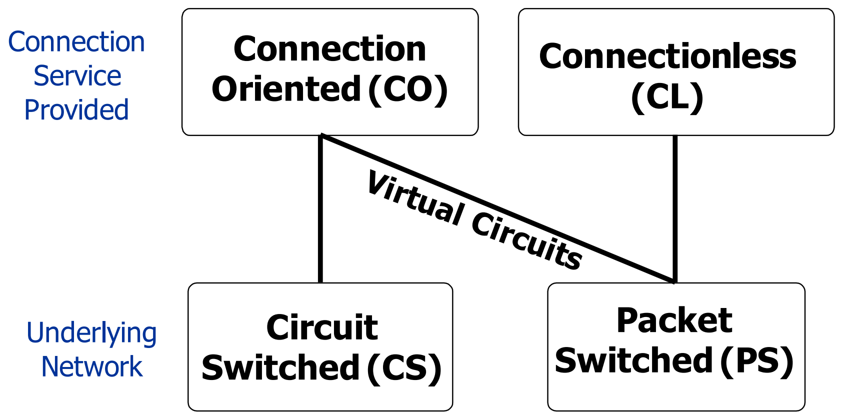
– Set-up / tear-down overhead

– No dedicated resources but reservations may be possible

Order may be maintained (unlike CL)

Asynchronous Transfer Mode (ATM)

Summary: Classes of Network Connection



Scale of Networks

Interprocessor distance	Processors located in same	Example
1 m	Square meter	Personal area network
10 m	Room	
100 m	Building	Local area network
1 km	Campus	
10 km	City	Metropolitan area network
100 km	Country	Wide area network
1000 km	Continent	
10,000 km	Planet	The Internet

Local Area Networks (LANs)

Transmission through buildings
- Typically 80% of communications are local

Many and varied devices
- Different message sizes and rates
- Nodes may connect and disconnect, or fail
- Systems may compete or co-operate

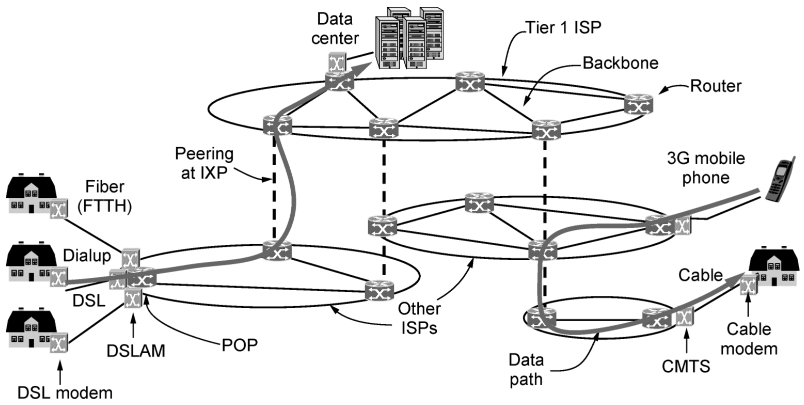
Typically under single administrative domain

Metropolitan, Wide-Area, Inter-nets

Formed from interconnected LANs
- Longer distances
- Costs of long cables, satellite links
- Delay and bandwidth restrictions due to distance

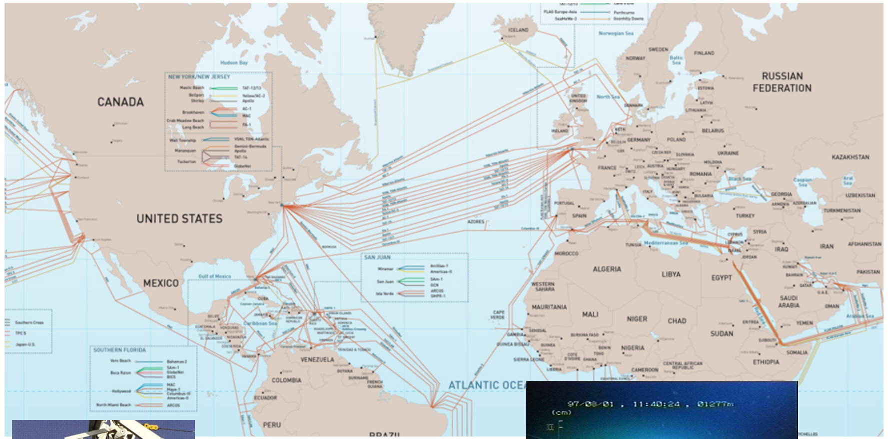
Politics of shared ownership and international connections

Overview of Internet Architecture

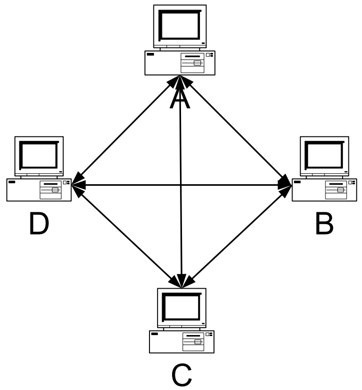


From: Tanenbaum and Wetherall, Pearson Ed. 2011

Internet Submarine Cables



LAN Topologies: Mesh



Fully connected graph

- Requires n(n-1)/2 links
- Doesn't scale to many nodes

No routing, as all nodes connected to each other

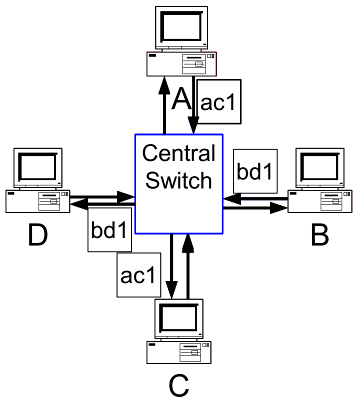
LAN Topologies: Star

Central switch

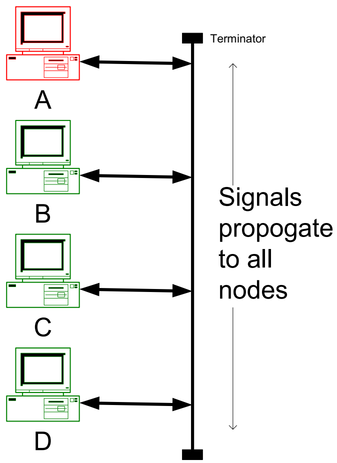
- Used by all nodes
- Links have simplicity of single communicating pair
- Switch must store/forward packets

Full bandwidth of link available to each node

- Assuming switch can keep up



LAN Topologies: Bus

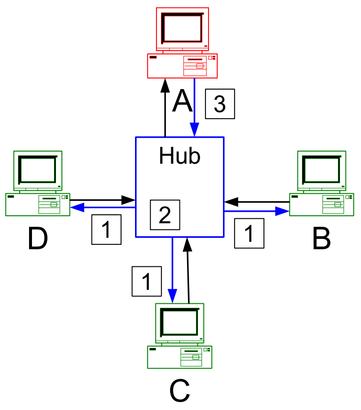


Bus is shared medium

- Divide data into <u>frames</u> (packets) to share link fairly
- Must <u>address</u> frames
- Must avoid and/or cope with frames <u>colliding</u>

Need **medium access control**

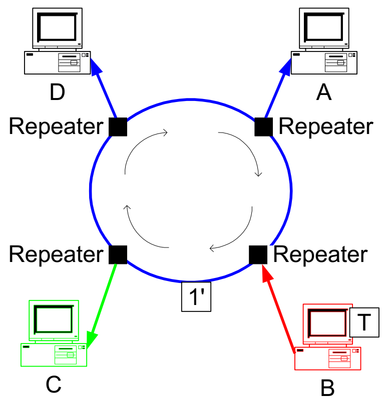
LAN Topologies: Bus with Hub



Central hub
- Used by all nodes
- Forwards all frames onto all outgoing links

Like bus with central connection
- Hubs can be formed into tree to extend bus size

LAN Topologies: Ring



Data circulates on ring in one direction
- Divide data into frames with addresses
- Permission to send given by permission **token T**
- Source node removes frame on return

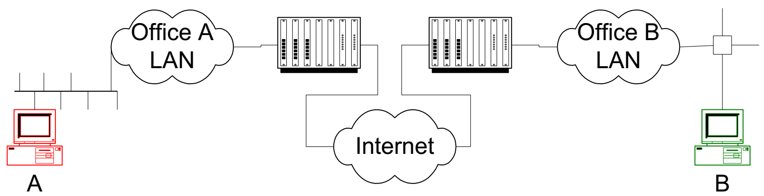
Comparing LAN Topologies

Switch
- Dedicates connections to communicating node
- Not as cheap as some alternatives

Ring
- Ring with tokens has simple equal use system

Bus/Hub
- Link bandwidth shared between all nodes
- Good for multicast/broadcast
- Require medium access control
- Require switched division to scale

Typical Wide-Area Connection Path



Subnets are smaller networks that form connection
- Subnets segment traffic
- Limit propagation of signals

Support for underlying networks
- With different physical technologies
- With different administrative ownerships

Network Abstractions

Applications view network as black box service

- Hide the details of the network
- Many parameters are orthogonal

☛ How do we describe a complete network architecture?

General-purpose networks are complex

- Different networking technologies
- Equipment provided by multiple manufacturers
- Managed by different people

☛ How do we define intended behaviour?

Standards

Standardised ways of connecting systems

- Hardware and software (protocol) standards
- Freeze technology and require backwards compatibility
- Do not prescribe implementation

Many standard bodies exist

- e.g. ISO, ITU, IEEE, IETF, W3C, …

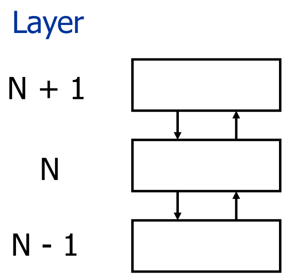
Different types of standards

- Open (published, free) vs proprietary standards
- e.g. industry provides (de-facto) standards

Network Stack Model

Model network as **layered stack**

- Layer N provides well-defined service to Layer N+1
- Layer N uses Layer N–1 for communication



Layering provides modularity

- Layers do not process data from higher layers
- May replace implementation of layers

But too many layers lead to inefficiency

Design issues at a layer

Error Correction

Multiplexing

Addressing

Routing

Congestion

Flow Control

Quality of Service

Real Time

Security

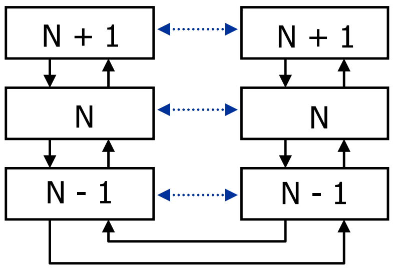
… layers may deal with only some of these issues

Protocols

Protocol: "an agreement between parties on how communication is to proceed"

- Defines msg formats, relationships between msgs, …
- Reuse protocol implementations across apps

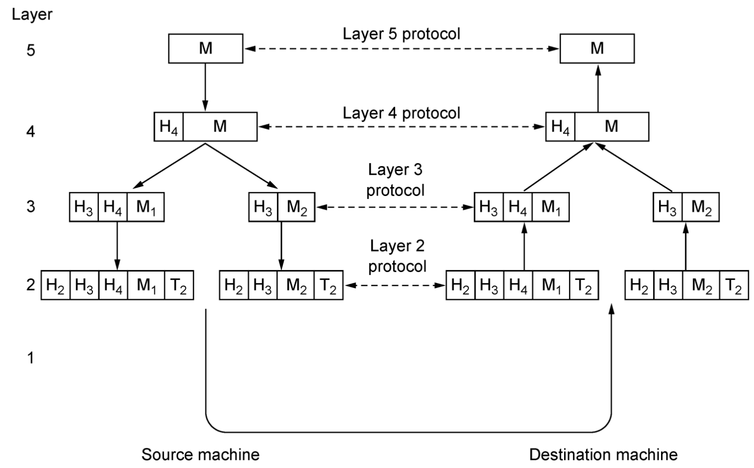
PDUs



Entity at one host exchanges **protocol data units** (**PDU**) with peer entity at another host

- Actual connection only at lowest layer

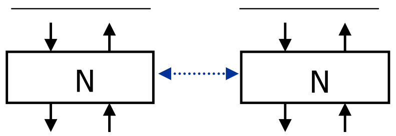
Protocols and Data Encapsulation

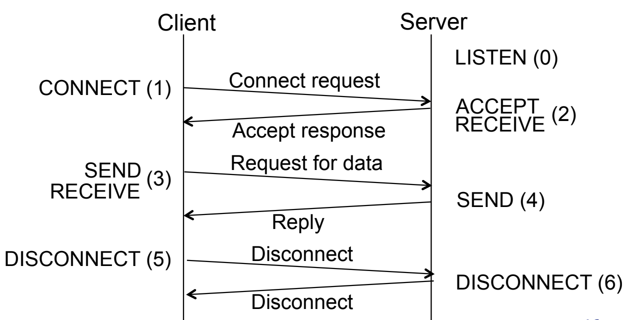


Example information flow supporting communication in layer 5.

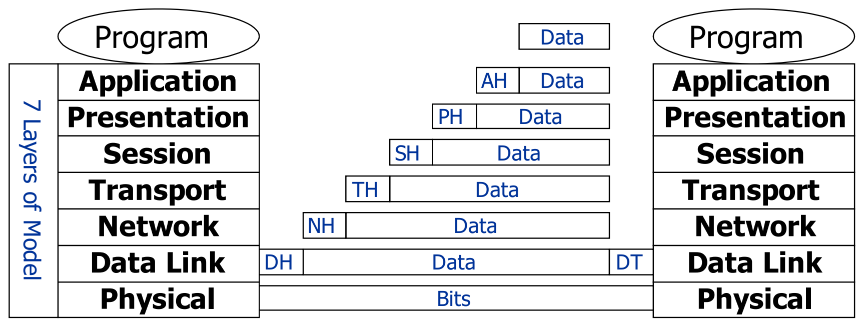
Protocols and Service Primitives

Primitive	Meaning
LISTEN	Block waiting for an incoming connection
CONNECT	Establish a connection with a waiting peer
ACCEPT	Accept an incoming connection from a peer
RECEIVE	Block waiting for an incoming message
SEND	Send a message to the peer
DISCONNECT	Terminate a connection



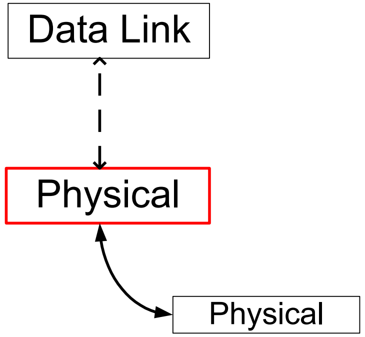


OSI Reference Model



PDUs

OSI – Physical Layer

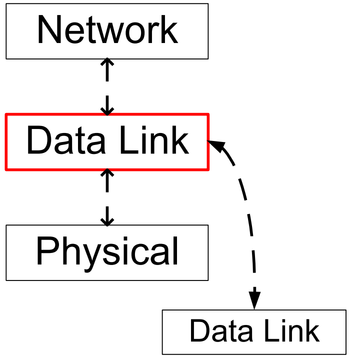


Transmission of bit-stream over medium

Encodes data according to signalling standards

Connectors and cables defined

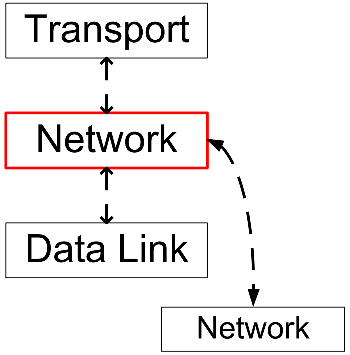
OSI – Data Link Layer



Arranges data into bit stream for sending over physical link
- Data encoded in transmission frames
- Low-level flow and error control for single hop

Possible services to network layer
- Unacknowledged CL
- Acknowledged CL
- Acknowledged CO

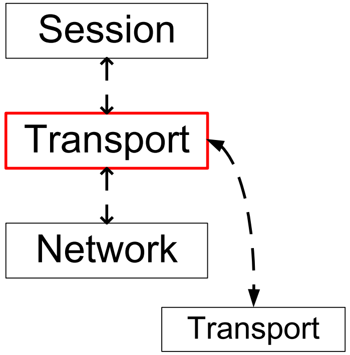
OSI – Network Layer



Provides end-to-end transmission of data
- Set-up and termination of connections (CO)
- Global addressing and routing (CL)
- Hides differences in underlying networks

Uses data link layer to provide transmission over single hops

OSI – Transport Layer



Provides transparent transfer service
- End-to-end flow control and error recovery
- Can be more reliable than underlying network

OSI – Session and Presentation Layers

Session Layer
- Enhances transport for sessions with special services
- e.g. dialogue synchronisation, exception handling

Presentation Layer
- Manages syntax and semantics of data exchanged
- e.g. data encryption, authentication, and compression
- e.g. data marshalling, byte ordering, …

☛ We don't look at session and presentation layers much in this course.

OSI – Application Layer

Provides interface to application
- But does not include the application!
- Network functionality specific to given application
- Most users only have contact with app layer

Protocols for common application interactions
- e.g. file transfer, e-mail, web

TCP/IP Model

OSI	TCP/IP
Application	Application
Presentation	Not present
Session	Not present
Transport	Transport
Network	Internet
Data Link	Host-to-host network
Physical	Host-to-host network

Developed by DoD for ARPANET
- Still used in Internet
- Designed to be resilient to failures

Presentation and session functions not seen as necessary

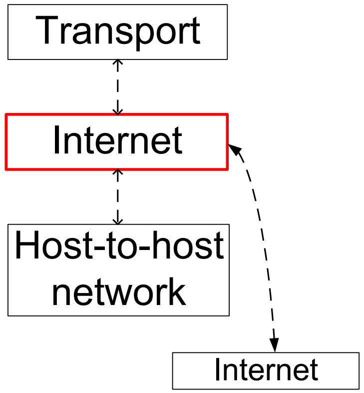
Host-to-host network largely undefined

End-to-End Principle (Saltzer, Reed, Clark)

"Communications protocol operations should take place at ends of protocol connection"
- Data link control happens at ends of wires
- Network control happens at ends of subnets
- Transport control happens at ends of connections

Results in efficiency and transparency
- Each layer doesn't do unnecessary work
- Intermediate nodes don't process higher layers
  - Could result in unexpected behaviour

Internet Layer



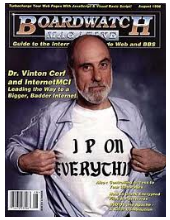
Packet-switched (PS), connectionless (CL), inter-networking layer

Delivery to destination

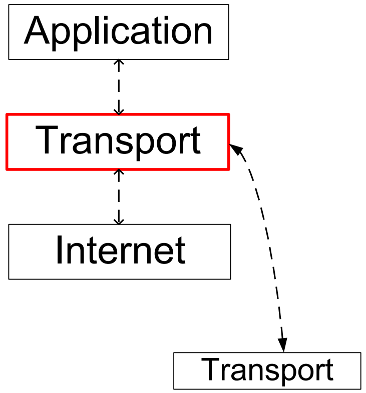
- Routing, congestion control
- Hides different physical networks

IP protocol realises layer

- Defines packet format



Transport Layer

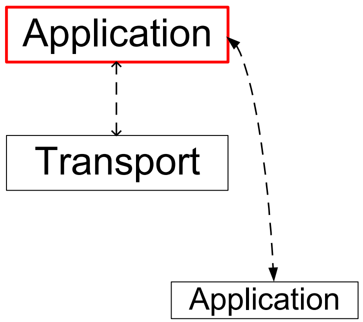


End-to-end connections

- Flow control
- Error recovery

TCP and UDP realise layer

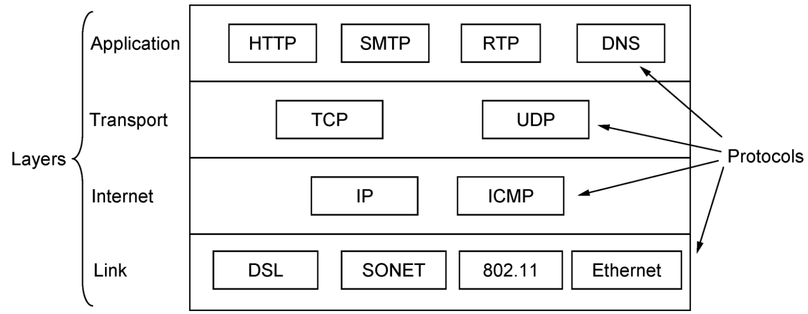
Application Layer



Protocols for application interaction

- HTTP (web)
- SMTP (e-mail)
- DNS (host naming)
- FTP (file transfer)
- NNTP (usenet news)

Example Protocols



Comparing Reference Models

OSI Model	TCP/IP Model
The standard model	Concepts lack generality
Can be complex, not all layers always used	Host-network layer poorly defined
OSI protocols unpopular and poor implementation	TCP/IP protocol most widely used

- Computing (and this course) tends to use OSI model but Internet protocols