

Introduction

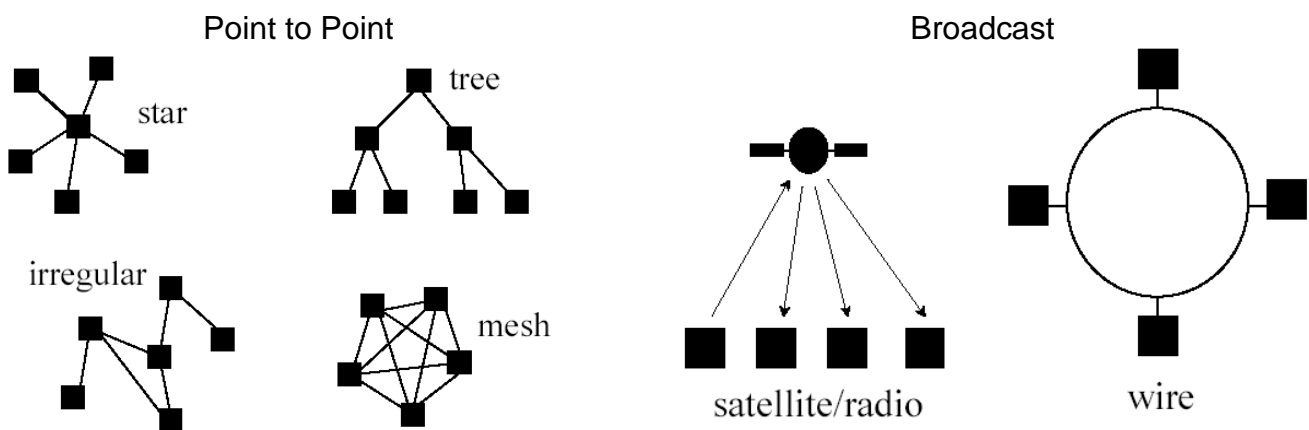
A computer network is 'an interconnection of *autonomous* computers'. Network applications include access to remote resources, human communication, mobile computing and increased power through parallelism.

Networks can be classified by size (LAN, WAN, etc.), connectivity (Point to Point, Broadcast, etc.), communication medium and mobility (fixed or mobile).

Differences between LANs and WANs

- Speed (bandwidth, latency (time taken for a packet to be sent))
- Management
- Security
- Reliability
- Billing
- Heterogeneity (similarity of components?) and Standards

Connectivity



Differences between Communication Media

- Speed (bandwidth, latency)
- Range
- Sharing
- Topology
- Installation & Maintenance costs
- Reliability

Issues arising in Mobile Networks

- Location & Tracking
- Semi-persistent connections
- Complex administration and billing

Common Issues in Networking

- Addressing
- Routing
- Framing & Encoding
- Error detection and correction
- Flow and congestion

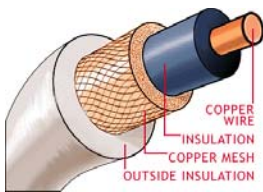
PART I – DATA TRANSMISSION

Transmission Media

Copper Wire

- Inexpensive, easy to install
- Copper has a low resistance
- Problem – interference (worst in close parallel wire)

To reduce crosstalk or electromagnetic induction between pairs of wires, two insulated copper wires are twisted around each other – **twisted pair**. One wire to carry the signal, one as ground. As any interference would effect both wires, the signal – ground = original signal.



Another alternative is **coaxial cable**, where the insulated signal carrying wire is surrounded by a metal shield that also acts as a ground. This medium requires an amplifier every 10km or so.

Glass Fibre (or Optical Fibre)

- Thin glass tube that reflects light internally.
- Transmitter uses LED or LASER and receiver uses a light sensitive transistor.
- Advantages: No electrical interference, carries more information much further than copper, only needs one fibre – not a pair like copper wire, only requires an amplifier about every 100km.
- Disadvantages: Installation requires specialist equipment, breaks in fibre are hard to detect and repair.

Radio

- Broadcast of EM waves through the air at radio frequency (RF)
- Each computer uses an antenna (size \equiv range) – no direct connection required.

Microwave

- Higher frequency EM waves
- Advantages: directional (privacy implications), higher bandwidth than RF
- Disadvantages: cannot penetrate metal structures

Satellites

- Radio and Microwave do not bend around the earth, so satellites are required for long distances.
- Satellites are expensive and so each will deal with many customers on different frequencies.
- Three satellites in geo-synchronous orbit (about 27,000km altitude) can cover the entire earth.

Infrared

- Higher frequency than radio or microwave. Initially used for remote controls
- Advantages: portable, inexpensive
- Disadvantages: short range, sensitive to trans/receiver orientation

Revision Notes

Laser

- Uses light to carry information through the air.
- Stays focussed over long distances, but requires a starlight line of sight and can easily be blocked.

Comparison of Media

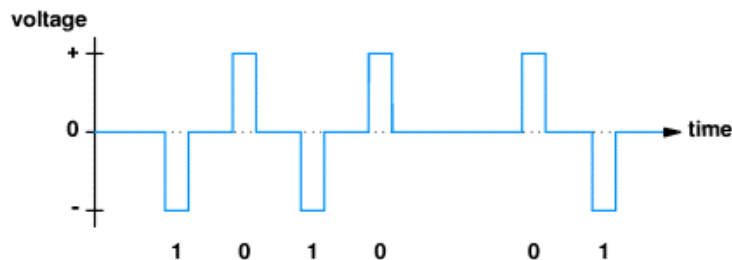
Medium	Cost	Speed	Atten	Interfere	Security
UTP	Low	1-100M	High	High	Low
STP	Medium	1-150M	High	Medium	Low
Coax	Medium	1M-1G	Medium	Medium	Low
Fibre	High	10M-2G	Low	Low	High
Radio	Medium	1-10M	Varies	High	Low
Microwav	High	1M-10G	Varies	High	Medium
Satellite	High	1 M-10G	Varies	High	Medium
Cellular	High	9.6-19.2K	Low	Medium	Low

Note: *Attenuance* is a measure of how much of the signal strength is lost during transmission, i.e. Strength of Signal Sent ÷ Strength of Signal Received.

Local Asynchronous Communication and the RS-232 Standard

- Receiver does not know when the Sender will transmit.
 - o Transmit when data is ready
 - o Variable delays between transmissions
 - o No sender-receiver coordination before transmission.
- E.g., a keyboard connected to a computer

Using Electric Current to Send Bits



Communication Standards

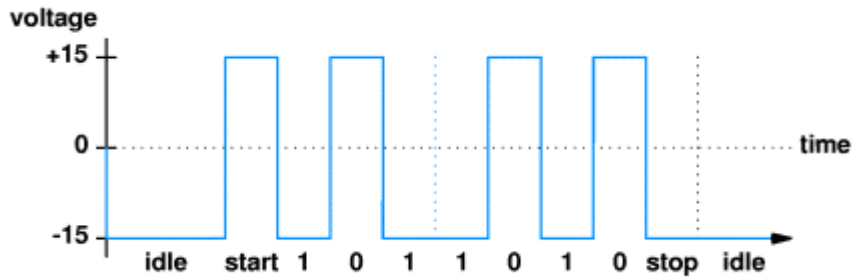
- Standard ensure that hardware from different manufactures can interoperate, by agreeing on standard voltages, timings, etc.
- Standards are published by standard organisations such as ITU, ISO, etc.

RS-232

- Used to connect keyboards, terminals, etc. to computers over copper wire.
- Transmits data in 7-bit characters
- Serial communication
- An idle line is the same as a 1 bit.
- A 0 bit is sent to mark the start of transmission

Revision Notes

- After the 7-bit character is sent, a 0 bit marks the end of transmission
- Must be at least a one bit gap between characters.



Framing Errors

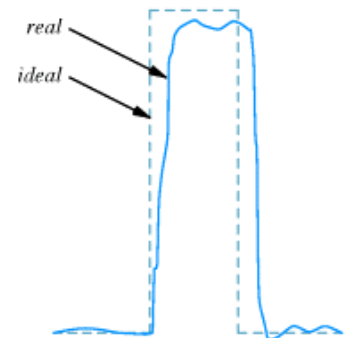
- Start and stop bits represent a framing of each character
- If transmitter and receiver are using different speeds, stop bit will not be received at the expected time. This problem is called a framing error.
- RS-232 devices may send an intentional framing error called a BREAK to abort a transmission.

Full Duplex Communication

- Two wires required to send information in one direction (one signal, one ground)
- Full duplex (2-way) communication requires 3 wires (two signal, one ground)
- RS-232 uses a 25-pin connector (extra pins used for control functions)
- Computer transmits on pin 3 and receives on pin 2 (opposite to MODEM)

Limitations

- Voltage cannot change instantly so RS-232 must define a tolerance to imperfect signals and a maximum rate at which signals can change – **bandwidth**¹.
- Measured in cycles per second – Hz.



Sampling Theorems

- Theoretical limit on the maximum speed at which data can be sent.
- Nyquist defined an equation for an error-free (noiseless) medium:

$$D = 2B \cdot \log_2 K$$

- Shannon extended this to work for noisy media:

$$D = B \cdot \log_2 (1 + S/N)$$

The Signal to Noise Ratio (SNR) is expressed in dB and given by:

$$10 \cdot \log_{10} (S/N)$$

where *D* is data rate

B is bandwidth

K is the levels of signal (RS-232 uses two).

S is signal power

N is the noise power

¹ *Bit rate, Baud rate and Bandwidth*

- Bit rate – number of bits sent per second \equiv number of *people* travelling down a road
- Baud rate – number of signal changes per second (for RS-232, bit rate and baud rate are the same) \equiv number of *cars* travelling down the road
- Bandwidth – maximum number of signal changes that the medium can accommodate \equiv number of cars that the road can take

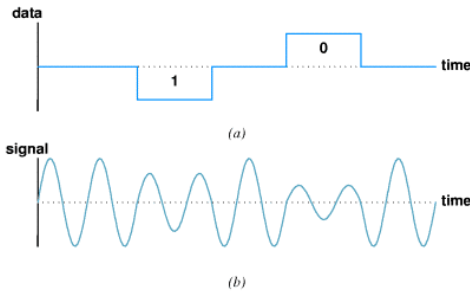
Long Distance Communications, Carriers and Modems

Resistance in wires leads to signal loss and so the current cannot be propagated over long distances. However, a continuous oscillating signal will propagate further than other types of signal.

To transmit binary data, therefore, an oscillating *carrier wave* is modulated to carry the data. There are several modulation techniques:

Amplitude Modulation (AM)

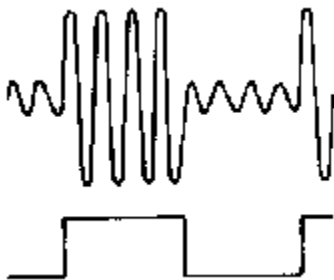
- Change the amplitude of the carrier:



AM is not very safe as the modulations are small and easily disrupted. It is used with Fibre Optics, which has a low attenuation.

Frequency Modulation (FM)

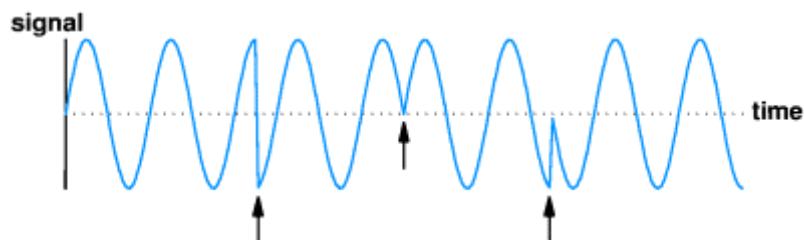
- Slightly change frequency of the carrier:



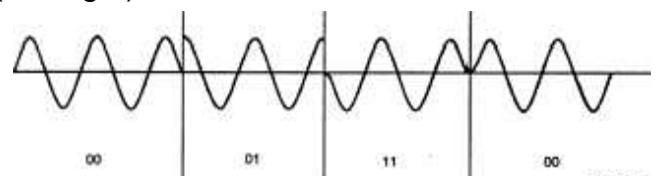
Frequency modulation requires a wider bandwidth than amplitude modulation by an equivalent modulating signal, but this also makes the signal more robust against interference. Frequency modulation is also more robust against simple signal amplitude fading phenomena.

Phase Shift Modulation

- FM and AM require at least one wave cycle for each bit. Phase Shift modulation changes the timing of the carrier and can send several bits per cycle. When the phase shifts, this represents a change from low-high or high-low.



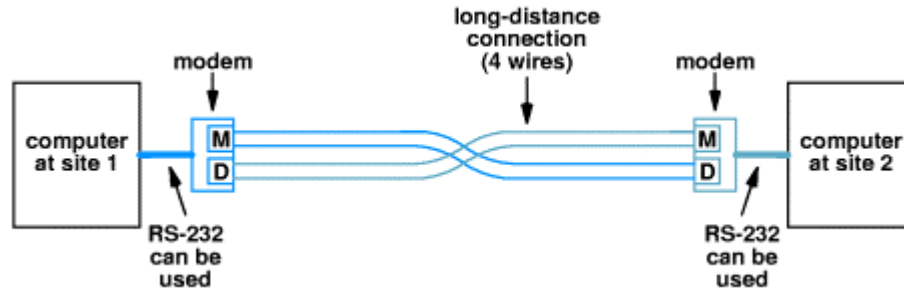
- Multiple bits can determine the phase shift (see right).
- In general, K bits per shift and B shifts per second gives a Baud rate of KB bps.
- A *Modulator* modulates the carrier wave.
- A *Demodulator* interprets the signal.
- Full duplex communication requires a modulator and demodulator at each end – a MoDem.



Revision Notes

Leased Serial Data Circuits

- Long distance four-wires circuits (example below) can be leased from a phone company. This is often called a serial line or serial data circuit.

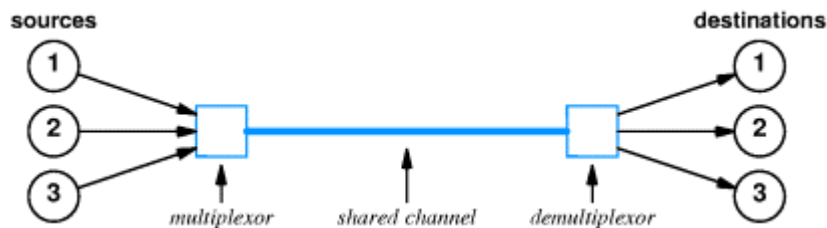


Modems

- Used with optical fibre, radio and conventional analogue telephone systems.
- Dial-up modems work with the existing phone system to essentially 'mimic' a telephone (including circuitry to represent picking up, waiting for dialtone, hanging up, etc.). The carrier is an audible tone. It uses a single voice channel (2-wire circuit) and co-ordinates the send/receive times to achieve full duplex communication.

Multiplexing

- Carrying multiple signals on one medium is called multiplexing



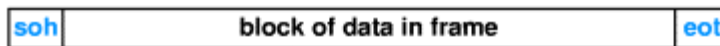
- *Frequency Division Multiplexing (FDM)* achieves multiplexing by using different carrier frequencies. The Receiver can "tune" to specific frequency and extract modulation for that one channel. The frequencies must be separated to avoid interference. Only useful in media that can carry multiple signals with different frequencies - high-bandwidth required.
- *Time Division Multiplexing (TDM)* is an alternative method where the sources share the medium:
 - o *Synchronous TDM* – pre-assigned timeslots
 - o *Statistical TDM* – optimizes time slots according to need, i.e. busier computers get more time.

PART II – PACKET TRANSMISSION

Packets and Frames

- Most networks transmit data in small blocks called *packets*. This helps in detecting errors and gives fair access for a shared medium. Such networks are called *packet networks* or *packet switching networks*.
- Each type of hardware frames packets in a specific way.

An Example of Framing

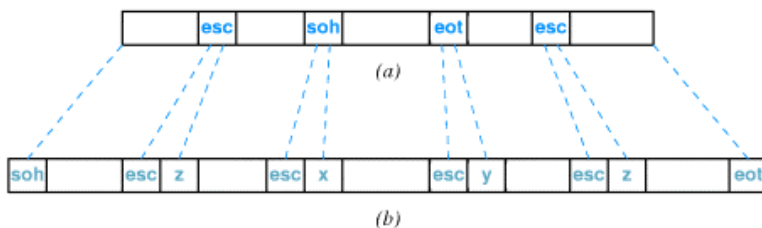


- The frame puts the data between control data marking it's start and end.
- Sending computer sends `soh` first, then data, finally `eot`. Receiving computer interprets and discards `soh`, stores data in buffer and interprets and discards `eot`.
- This technique has advantages in error detection in that if the sender crashes, `eot` will not arrive and if the receiver crashes, `soh` marks the next valid frame.
- However, this technique requires two extra characters each frame, and cannot carry arbitrary values (i.e. the `soh` and `eot` characters cannot appear in the data block).

Byte Stuffing

Since the characters `soh` and `eot` are used to delimit the frame, the two characters cannot be used again in the data. This problem can be resolved by *byte stuffing*. This process reserves

a third control character to mark occurrences of control characters in the data block. In the example, left, the ASCII `esc` character is used. `soh` is replaced by `esc x`, `eot` by `esc y` and `esc` by `esc z`.



Errors

- Much of the complexity of networks arises from susceptibility to interference than can cause either transmitted data to be lost or changed or for random data to appear.
- Errors can be *single-bit errors* or *burst errors*.

Basic error checking methods entail the sender including some extra information that summarizes the actual data block. The receiver checks that the data it received complies to this summary. Then, if an error has occurred, the block can either be corrected or retransmitted.

Parity Bits and Parity Checking

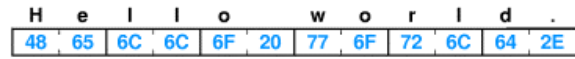
- An extra bit is added on the end of the data to make the number of '1's either even or odd (depending on whether the system is using even or odd parity).
- E.g.: Even parity: 1011001 → parity bit of 0; Odd parity: 1011001 → parity of 1.
- Introduces additional costs
- Only detect limited errors, i.e. only detects an error if an odd number of bits are changed. Therefore only normally used to detect single-bit errors.

Revision Notes

Checksums

- Interprets the data as a sequence of integers and then adds them (plus any carry bits) together to compute a *checksum*. This is then appended to the frame. 16- and 32-bit checksums are common and are usually computed for a whole packet.

E.g.:



$$4865 + 6C6C + 6F20 + 776F + 726C + 642E + \text{carry} = 71FC$$

(In this case, the characters are grouped in pairs and then added)

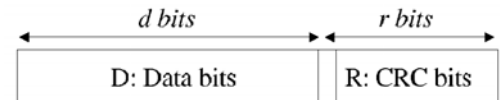
- May fail if, for example, one bit in each of four data items is reversed:

Data Item In Binary	Checksum Value	Data Item In Binary	Checksum Value
0001	1	0011	3
0010	2	0000	0
0011	3	0001	1
0001	1	0011	3
totals		7	

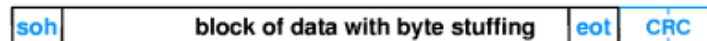
Cyclic Redundancy Checks (CRC)

Detects more errors than checksums and only requires simple hardware. Based on binary division rather than addition.

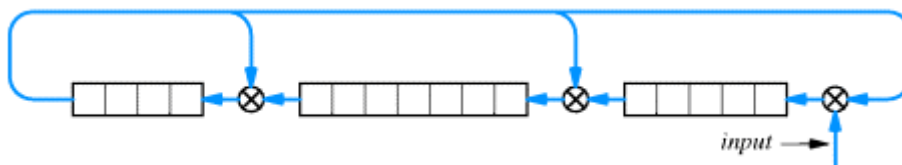
- The data, D is d bits long.
- The system uses a generator pattern, G, which is r+1 bits long.
- The sender appends a bit pattern R (length r) to D such that the entire sequence is exactly divisible (using binary (modulo 2) arithmetic) by G.
- The receiver divides the received bit pattern by G and checks the remainder.
- CRC can detect burst errors of less than r+1 bits and odd numbers of bit errors.
- CRC can detect burst errors of greater than r+1 bit with probability $1-0.5^r$.



Revised frame format:



Generation of CRC code at a hardware level:



- Combines 3 shift registers and 3 xor units.
- Feed in each bit of the message. The final state of the system is R.

Network Topologies

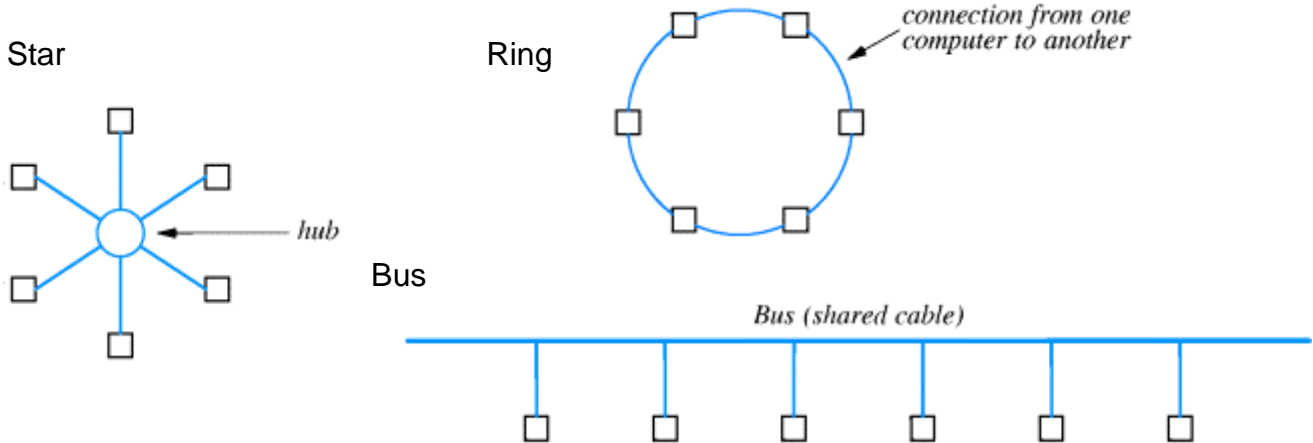
- Early local networks used dedicated links between each computer to create a *mesh* network. These had their advantages, in that hardware and frame details can be tailored for each link and it is easy to enforce security. However, as the number of machines increase, the links grows very fast indeed! Also, if the network was split across buildings, many links would have to travel between the two sites.

Revision Notes

- Shared Communication Channels rely on computers sharing a single medium and to coordinate their access. This is low cost, but not suitable for wide areas as communication delays inhibit coordination.

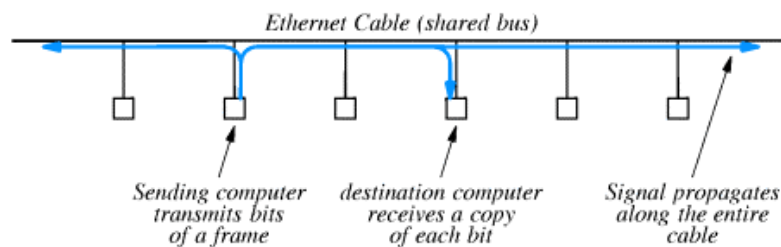
The Locality of Reference Principal states that computers in close proximity are more likely to communicate with each other, and a computer is likely to communicate with the same computers repeatedly.

LAN Topologies



- Star is more robust (severed link only affects one machine), but hub may be a bottleneck.
- Ring enables easy coordination but is sensitive to a severed cable.
- Bus requires less wiring but is also sensitive to a severed cable.

An example Ethernet Bus Network



- Single coaxial cable (terminators at each end to prevent signal reflection)
- IEEE specifies details such as data rates, maximum length, minimum separation, frame formats, electric and physical details, etc.
- Signal propagates along entire cable - destination computer receives copy.
- The computers can detect when a signal is on the Ether - *carrier sense*.
- Any machine can transmit when the Ether is free - *carrier sense with multiple access*.

Collisions on a CSMA Network

- Occur if two computers sense an idle ether and attempt transmission simultaneously.
- Each machine also recognizes garbled transmissions.
- This adds *Collision Detection* to the CSMA - *CSMD/CD*.
- This is an example of a Medium Access Control (MAC) protocol.

Collision Recovery

- After a collision occurs, the computers must wait before retrying.
- A sending machine chooses a random delay up to a specified maximum. For each subsequent collision, the delay is doubled. This is called *Exponential Backoff*.
- More offered traffic = more congestion = more backoff = reduced throughput.

What to do if the Ether is busy

- Non-persistent (Deferntial) CSMA:
 - o If medium is idle, transmit
 - o If medium is busy, wait a random time and then try again
- 1-persistent (Selfish) CSMA:
 - o If idle, transmit
 - o If busy, listen until idle, then transmit
- p-persistent (Compromise) CSMA:
 - o If idle, transmits with probability p or waits (conversely, with probability 1-p)

Another Example - LocalTalk

- LAN technology for Apple computers
- MAC protocol is CSMA/CA - collision *avoidance*.
- Each computer sends a message to reserve the bus before sending.
- Despite these attractive properties, *LocalTalk* has only 2% of the bandwidth of Ethernet - 230.4Kbs

Wireless LANs and CSMA/CA

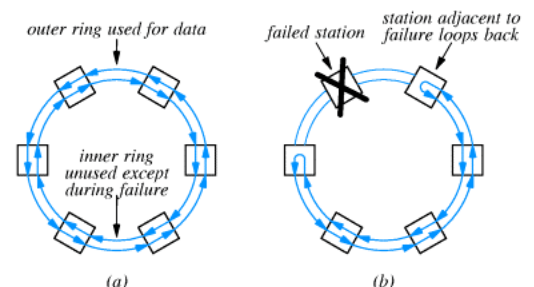
- Collision detection would not work - not all computers may 'hear' the announcement of a collision.
- Therefore, Wireless LANs use Collision Avoidance:
 - o Sender sends a small message to the receiver
 - o Receiver responds with a 'clear to send' message, received by all adjacent machines.
- 2Mbs

The IBM Token Ring Network

- MAC protocol based on token passing
- Computer must wait for permission (by receiving a *token*) before transmitting
- Once a computer receives a token, it takes it, sends a frame which then flows right round the ring. The receiver makes a copy. When the frame arrives back at the sender, it checks it for errors, then gives back the token.
- 16Mps

The FDDI Network

- A ring network fails if the cable is severed. FDDI solves this by using two counter-rotating cables. This is very expensive.
- 100Mbs (fibre optic)



The ATM Star Network

- Asynchronous Transfer Mode
- Uses pairs of optical fibres to connect computers to the central hub/switch.
- The switch forms a point-to-point connection.
- Over 100Mbs

Hardware Addressing and Frame Type Identification

- Addressing is done at a hardware level to prevent unnecessary messages being seen by a computer.
- Each station has its own unique *hardware* or *physical address*.
- The network hardware in each station decodes each frame and uses the destination address to determine whether to accept it.

Static, dynamic and configurable hardware addresses

- **Static:** the manufacturer assigns the hardware address which never changes
Easier for customer, addresses never change and computers can easily move around the network.
- **Dynamic:** assign the address whenever the station boots up
Easier for vendor, addresses can be smaller, but conflicts can arise
- **Configurable** - the network administrator can set the address (usually on installation)
Allows permanent and smaller addresses

Broadcasting

- Some applications require that a sender transmits a frame to all stations on the network.
- This is accomplished by using a special reserved broadcast address (in practice, this consists of all '1' bits).
- All stations configured to accept packets for their own address *and* the broadcast address.

Multicasting

- Some applications require that a sender transmits a frame to a group of stations on the network, e.g., teleconferencing.
- In this case, the group is allocated a single address, which stations in the group only are configured to recognize.

Promiscuous mode

- Network hardware can often be programmed to accept all packets. This is useful for network analysers (sniffers) but also has some serious security implications!

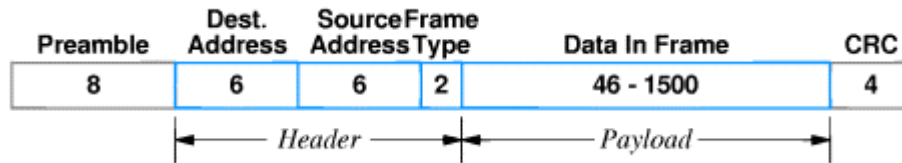
Identifying packet contents

- Computer may need to know what type a packet is in order to process it.
- **Explicit frame type** - network hardware designers specify how type information is included in the frame and may define types
- **Implicit frame type** - the sending and receiving computers must agree how to specify types in software

Frame format

- Frames typically include a frame header (of fixed size) and a frame data area (of a size between a standard minimum and maximum).
- **Ethernet Frame format:**
Ethernet uses a 48-bit static addressing scheme and a 16-bit frame type:

Revision Notes



The Frame Type codes are defined by standard organizations, of which there are many. Therefore the code must be in two parts: the first part identifies which Standards Organization is being followed, the second component is frame type as defined by that organization.

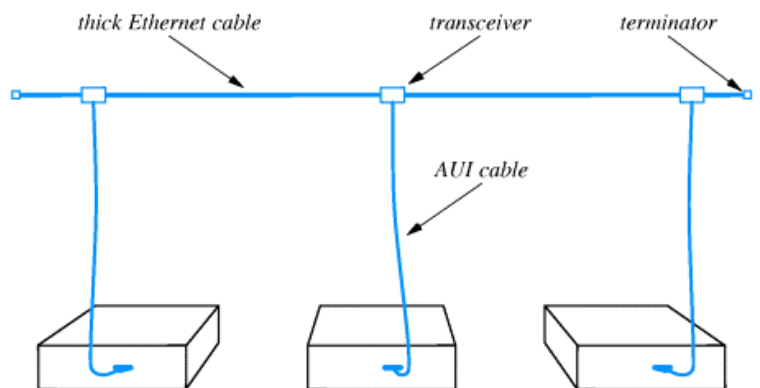
LAN wiring and physical topology

Network Interface Card (NIC)

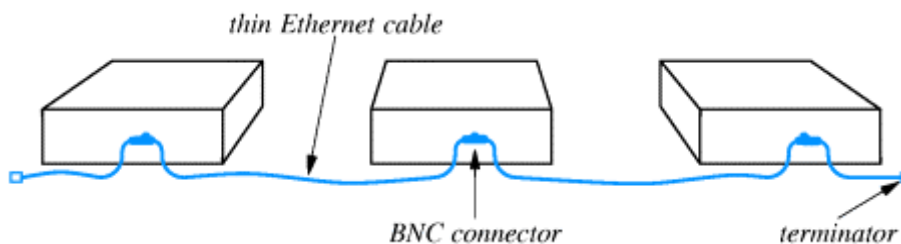
- Handles details of transmission (assembling frame, checking ranges, CRC, buffering, etc.) independently of the CPU. This is because the network speed is effectively faster than the CPU. E.g., a 100Mbps network sending 100M bits every second. Although an 800MHz processor can perform 800M operations a second, each bit received by the network will need several operations performed on it.
- Hardware specific – e.g., a Token Ring NIC cannot be used with Ethernet.

Thick Ethernet

- Uses thick coax cable
- AUI cable (or transceiver or drop cable) connects from NIC to transceiver. AUI cable carries digital signal from NIC to transceiver.
- Transceiver generates analog signal on coax.
- Wires in AUI cable carry digital signals, power and other control signals.
- Can be put through a Multiplexor (enables multiple machine to share one transceiver (and performs error checking)) to reduce number of transceivers.
- Can change computers without disruption
- Accessing remote transceivers can be difficult



Thin Ethernet



- Thinner coax cable - no transceivers, rather simple BNC connectors on each machine.
- Easier access
- Greater possibility of disconnection

Twisted-pair Ethernet (10Base-T)

- Twisted pair cable connects all computers to a central hub.
- Physically, it is a *Star* topology, but logically, it is a *Bus*, because the hub simulates bus networking, in effect, acting as a single cable.
- Cheap, cable/connection failure only affects one machine.

All Ethernet wirings use the specifications, frame format, CSMA/CD algorithms etc. Different wirings can be mixed on a single network. Some NICs provide access points for all three types.

Extending LANs

- MAC protocols such as CSMA/CD require the time to be proportional to the length of the cable.
- Electrical signal weakens with distance

Fiber Optics Extensions

- Extend connection between computer and transceiver. No signal loss.

Repeaters

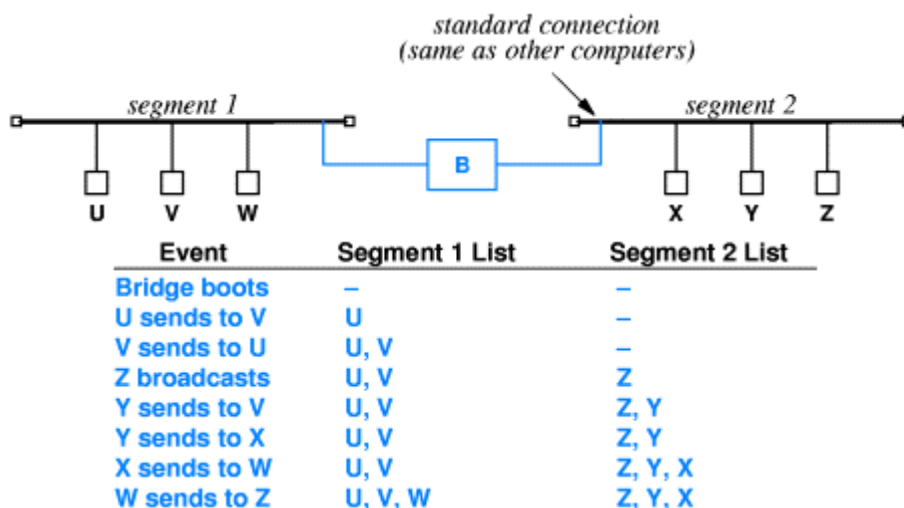
- Join Ethernet cables together. It has no knowledge of frames and simply amplifies whatever signals it receives.
- Multiple repeaters are allowed, but Ethernet standards dictate that there should be no more than four between any two machines.
- Fiber modems can be used between repeaters for long distance extensions.

Bridges

- Connect two segments, working at the frame level. Uses promiscuous mode, i.e. forwards all frames, but does not forward erroneous frames.

Frame Filtering

- Only forward a frame if i) destination is on the other segment, ii) broadcast address is used.
- The bridge learns which segment is on when that computer sends a frame.



- Propagation principle: a bridge only forwards as far as is necessary

Dijkstra's Algorithm – finds the shortest distance from the switch to every other node.

- Put all the nodes in a priority queue. Initialize all nodes to have an infinite distance, except the source node whose distance is initialized to 0.

```

while the queue is not empty {
    reorder the queue;
    dequeue a node, u;
    add u to path(u);

    for each neighbour of u, v {
        if (queue contains v) {
            if (distance(v) > distance(u) + weight(u,v)) {
                distance(v) = distance(u) + weight(u,v);
                path(v) = path(u);
            }
        }
    }
}

```

Distributed Route Computation

- Dijkstra's algorithm requires each switch to hold a complete description of the network.
- With Distributed Route Computation, each switch periodically computes a new local table and sends it to its neighbours.

Method (given a local routing table, a weight for each link and an incoming message):

```

loop forever {
    Wait for message. Let the sender be switch N
    for each entry in the message {
        V = the destination;
        D = the distance from here to V;
        C = D plus the cost of link over which the message arrived;
        // update routing table:
        if (no route exists from here to V) {
            add row to local routing table specifying:
            

| Destination | Next Hop | Distance |
|-------------|----------|----------|
| V           | N        | C        |


        }
        else if (a route exists with next hop N) replace existing distance with C;
        else if (route exists with distance > C ) change next hop to N and distance to C;
    }
}

```

Link State Routing

- Each switch periodically broadcasts the state of specific links to other switches
- Switches collect these messages and apply Dijkstra's algorithm to their version of the network.

Ownership, Service Paradigm & Performance

Ownership

- Private network: owned by the people that use it. Usually LANs combined with private switches and leased lines.
- Public network: owned and operated by an independent service provider. Any subscriber can potentially communicate with any other subscriber.

Service paradigm

- Connection oriented:
 - o Establish a connection
 - o Form a data path - a connection stream
 - o Either computer can close the connection
- Ease of accounting
- Error reporting

- Connectionless:
 - o Wraps data in a frame
 - o Supplies a destination address
 - o Passes it to network to send when possible
- No delay before sending

Permanent and switched connections

- Permanent connection: Established manually, Held in non-volatile memory.
- Switched connection: Established when needed

Technology	Connection-Oriented	Connectionless	used for LAN	used for WAN
Ethernet		•	•	
Token Ring		•	•	
FDDI		•	•	
Frame Relay	•			•
SMDS		•		•
ATM	•		•	•
LocalTalk		•	•	

Addresses and connection identifiers

- An address is a complete, unique identifier. Connectionless delivery requires address on each packet, but connection-oriented delivery can use a shorthand that identifies the *connection* rather than the destination.
- E.g., ATM:
 - o 428 bit packets
 - o 160-bit address
 - o 28-bit connection identifier:
 - 12-bit virtual path identifier (VPI) – the route used
 - 16-bit virtual circuit identifier (VCI) – the switch used
- The connection identifier is local to each computer
- May be different in different parts of the ATM switch

Revision Notes

Performance Characteristics

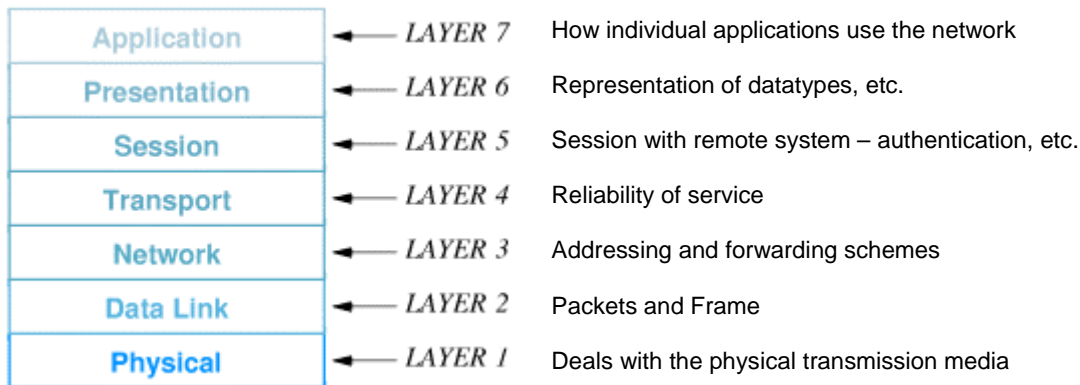
- Delay – the time taken for one bit to travel across the network.
 - o May vary with location on the network
 - o Propagation delays (time for signal to travel across medium), Switching delays (time taken for switch to forward frames), Access delays (time to get control of medium – CSMA/CA etc.) and Queuing delays (queues in switches).
- Throughput – rate at which data gets sent (bps). Compare to bandwidth – maximum possible data rate.
- Current delay = Delay when idle / 1 – utilization
- Delay Throughput product – measure of data that can be present on the network.

Protocols and Layering

- A *Network Protocol* is a set of rules for exchanging messages. They give a high-level interface to programmers and enables communication over different hardware.
- There is not normally one enormous protocol, but rather a protocol suite, making design, testing, extension etc. much easier and enabling selection and combination.
- Most commonly, protocols are divided into *layers*, each dealing with a different level of abstraction.

The ISO OSI (Open Systems Interconnect) 7 Layer Model

- ISO model for standardizing networks
- Very out-of-date, but a good example. Most networks now use the TCP/IP protocol (see later).



- Each layer only communicates with the layer directly above or below. Conceptually, a message would flow down the stack on transmission, through the Physical layer, and then up the stack on receipt by the destination machine.
- Each layer places information in a header before passing it to a lower level, or removes and processing the corresponding header before passing it to a higher level.
- Layer n software on the destination computer must receive exactly the same message as sent by layer n on the sending computer.

Common Networking Issues

- Sequencing for out-of-order delivery
 - o Connectionless network with dynamic routing may deliver packets out of order.
 - o Solution: each packet given a sequence number and passed up to the next layer in the correct order.
- Sequencing to eliminate duplicate packets
 - o Malfunctioning hardware could lead to duplicate packets being sent
 - o The Sequence number system above allow duplicate to be identified and discarded.
- Re-transmitting lost packets
 - o Destination computer sends an acknowledgement on receipt. If the Sender has not received the acknowledgement after a certain time, it resends.
 - o Limit the number of attempts before giving up.
- Avoiding replay caused by excessive delay
 - o A duplicate packet may turn up in a later session.
 - o Solution: Include a Session identifier, as well as a packet sequence number.
- Flow control to prevent data overrun
 - o Data overrun occurs when the sender sends quicker than the receiver can receive.
 - o Simple solution: wait for acknowledgement before sending next packet. This is very time consuming.
 - o Better solution – ‘Sliding Window Protocol’:
 - Sender and receiver agree a window size (number of packets)
 - Each window is sent and each packet acknowledged.
- Mechanisms to avoid network congestion
 - o Congestion can occur at bottlenecks in the network and because switches have a limited storage capacity.
 - o Detection:
 - Switches can inform senders
 - Packet loss can be used as a measure of congestion
 - o Solution: Rate control

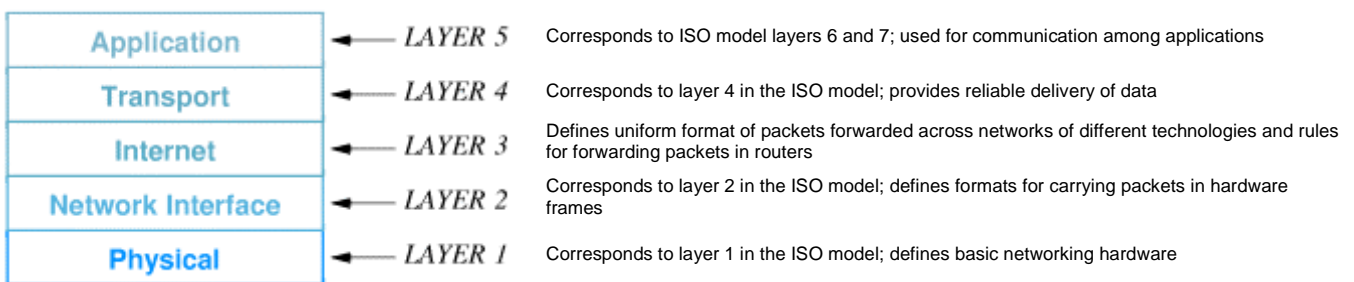
Very small changes in the design of network protocols can have drastic effects!

PART III – Internetworking

A large organization will use several networking technologies and will want any two computers to be able to communicate. However, two network technologies cannot just be wired together. Internetworking connects heterogeneous networks together through use of Routers and Universal Protocols to create a single virtual network.

The internet uses the TCP/IP protocols first introduced in the 1970s. Emerging into the public field in the 1990s, the Internet is governed by the Internet Engineering Task Force (IETF). In TCP/IP terms, a *host* is any computer that connects to the internet.

TCP contains five levels, instead of the OSI model's seven:



The Internet Protocol Addressing Scheme

- Addressing in TCP/IP is specified by the Internet Protocol (IP). Each host is assigned a 32-bit number called the IP address or Internet address. It is unique across entire Internet.
- Each IP address is divided into a prefix and a suffix. The prefix identifies the *network* to which computer is attached; the suffix identifies the *computer* within that network.

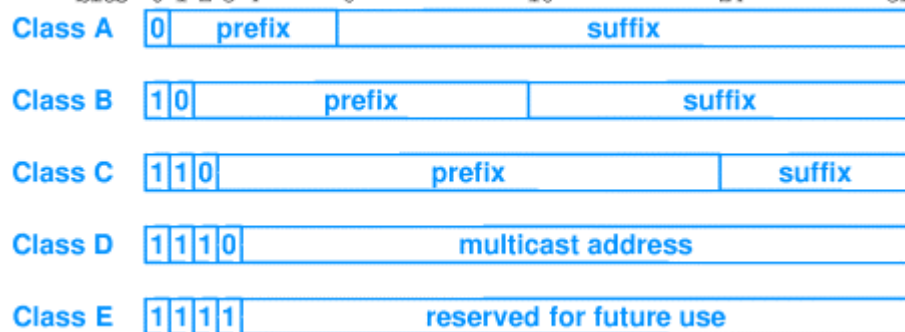
This

address format makes routing efficient.

- Assignment of network numbers must be coordinated globally; assignment of host addresses can be managed locally.
- A large prefix and small suffix gives many networks but few hosts per network. A small prefix and large suffix gives few networks but many hosts per network. The designers

chose a compromise in multiple address formats that allow both large and small prefixes.

Each format is called an address class. The class of an address is identified by first bits.



Revision Notes

- Dotted decimal notation represents the 32-bit number as four decimal numbers separated by dots, i.e. 10000001 00110100 00000110 00000000 → 129.52.6.0

- Makes representation easier, but class recognition harder. The table on the right shows which classes are represented by which number (the first part of the dotted decimal number).

Class	Range of Values
A	0 through 127
B	128 through 191
C	192 through 223
D	224 through 239
E	240 through 255

Address Class	Bits In Prefix	Maximum Number of Networks	Bits In Suffix	Maximum Number Of Hosts Per Network
A	7	128	24	16777216
B	14	16384	16	65536
C	21	2097152	8	256

Prefix	Suffix	Type Of Address	Purpose
all-0s	all-0s	this computer	used during bootstrap
network	all-0s	network	identifies a network
network	all-1s	directed broadcast	broadcast on specified net
all-1s	all-1s	limited broadcast	broadcast on local net
127	any	loopback	testing

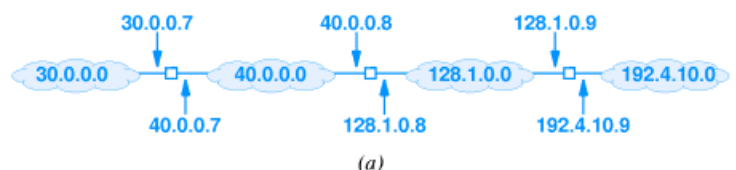
- Note: Routers will have two or more IP addresses (one for each different network connected to that router).
- An Internet packet passes through a series of routers. Each hop takes it over a particular network, either to a specific computer on that network or to the next router. In either case, the sending router has to map between the protocol (IP) address and a hardware address. This is called address resolution.

IP Datagrams

TCP/IP supports both connectionless and connection-oriented services. The fundamental delivery service is connectionless, on the Internet layer. An optional connection-oriented service is carried above this, on the transport layer.

Packets are sent across multiple physical networks via routers, so we can't just format the frame as the addresses are different in each network. IP defines a universal, virtual packet – the IP datagram. The maximum size (including header and data) is 64K octets.

Each router forwards a virtual packet by using a local routing table. This table matches the destination address against a mask (to show which part of the IP address is the prefix) and the next hop (which could be the IP address of another router, or direct delivery). Then it resolves the addresses.

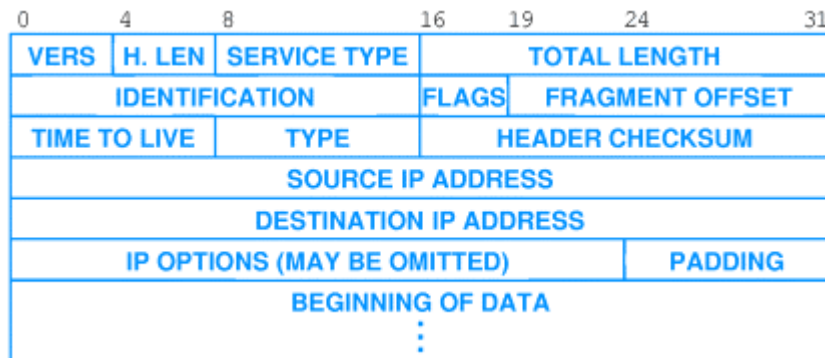


Destination	Mask	Next Hop
30.0.0.0	255.0.0.0	40.0.0.7
40.0.0.0	255.0.0.0	deliver direct
128.1.0.0	255.255.0.0	deliver direct
192.4.10.0	255.255.255.0	128.1.0.9

Revision Notes

IP does not provide guaranteed service. Errors such as duplicate, out-of-sequence, lost or delayed packets and data corruption may occur. The network layer – IP – can detect and report these errors, but fixing them is up to the Transport Layer.

The IP Header



Vers: Version of IP

H. Len: Length of Header

Service Type: Shortest Path or Highest Bandwidth

Flags/Fragment Offset – see Fragmentation, below.

Time to Live: Number of hops before self-destruct (to prevent loops)

Type: Type of data

IP datagrams are encapsulated in the data section of which ever type of frame the current network is using, with the frame type being set to 'IP'.

Fragmentation

Each network has an Maximum Transmission Unit (MTU), which specifies the largest possible packet allowed on that particular network. If a datagram exceeds this size, it is fragmented into a sequence of smaller datagrams (each with their own header). The Flag declares whether it is an original datagram, or a fragment of one. The Fragment Offset specifies where in the sequence this fragment should appear.

Reassembly is done at the final destination host. This allows fragments to take different routes and means the routers need not have the extra state information. If a single fragment is lost, the whole datagram is lost.

The Future of IP (version 6)

IP has been extremely successful at coping with the enormous growth of the internet over the past few decades. However, the limited (32-bit) address space will soon run out. Also, new applications, such as real-time audio/video require a guaranteed service, and some collaborative technologies require methods to send packets to a group of hosts.

The current version of IP is version 4. The new version (originally to be called IP – The Next Generation, is version 6. (Version 5 was an experimental release.)

IP 6 is connectionless, like IP 4 and has a 128-bit address space. There are different addressing modes for unicast, multicast and cluster. Extension headers can be used and there is support for audio and video.

Multicast:

- A single address corresponds to a group of computers. Membership can change at any time.
- One copy of the datagram is delivered to each machine, but only one copy passes over intervening networks.

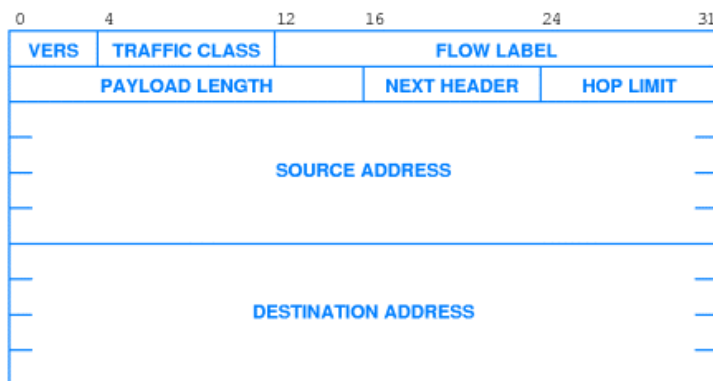
Cluster:

- Address corresponds to a group of computers whose addresses have a common prefix.
- Datagram sent to one of these.
- Used for replicating a service.

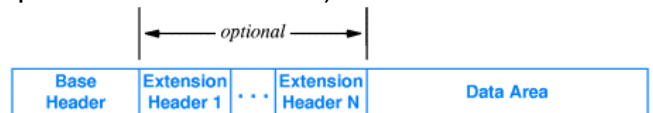
Address Representation

Dotted decimal is not really practical (an 128-bit address would have 16 parts). This can be reduced to 8 parts through using hexadecimal notation (parts delimited by colons). Further reduction is achieved by zero compression. For example, FF0C:0:0:0:0:0:0:B1 becomes FF0C::B1. This is especially useful as IP v.6 addresses beginning with 96 zeros corresponds to an IP v.4 address.

Datagram Format



IP version 6 headers are much smaller than those in version 4. Any extensions are added as extension headers (each header points to the next one).



This also means that the protocol can be updated without having to be redesigned.

The Traffic class specifies the type of data. This has implications for the acceptable delay levels, paths used, etc. The Flow Label identifies which path to use.

Paths

Applications can set up network paths in advance. These can be associated with different traffic classes that provide different Quality of Service (for service such as real-time audio and video).

Transmission Control Protocol (TCP)

IP has no guarantees as to quality of service. TCP establishes reliable end-to-end communication service on top of the IP layer. TCP is:

- Connection oriented
- Point to point
- Reliable – delivered as sent
- Full duplex
- Stream interface
- Reliable connection startup
- Graceful connection shutdown

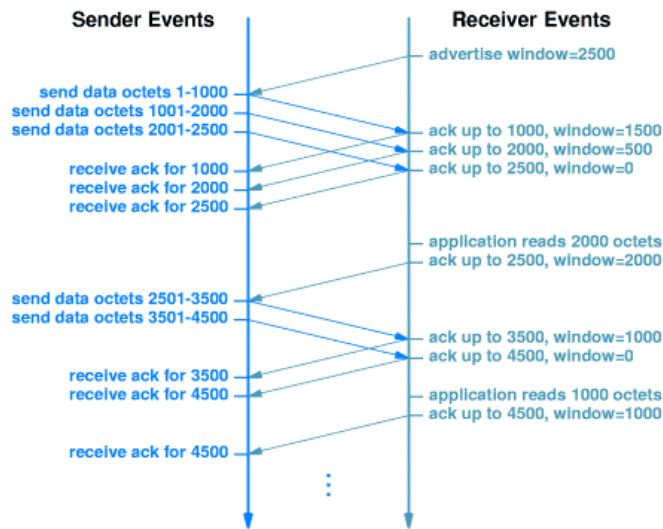
TCP has no knowledge of the underlying structure of the Internet, but sees only the sender and receiver. Problems such as non-delivery by the IP layer and destination machine crashes can be dealt with by TCP.

Packet Loss and Retransmission

- Sender sets a timeout.
- Receiver sends an acknowledgement
- Timeout results in retransmission

- Timeout values depend on the network. TCP sets the timer by noting the time taken to receive acknowledgements and computing weighted averages and variances over many transmissions.

Flow Control



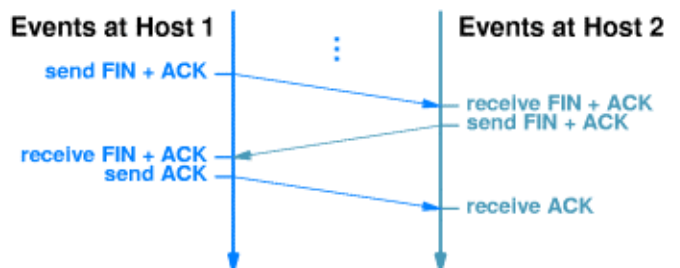
As in Ethernet, TCP using a window mechanism. Each end of the connection allocates a buffer and notifies the other of its size. The receiver includes its available window size in each acknowledgement (a *window advertisement*), and also when an application consumes some data (thus increasing the available window size). A zero window advertisement tells the sender to stop transmitting.

Opening and Closing Connections

TCP uses a three-way handshake to open and close connections. This consists of synchronization (SYN) and finish (FIN) messages and acknowledgements that all data has been received at both ends. FIGURE

Congestion Control

- On loss of a message, TCP backs off and sends just one small message.
- If this is not lost, the data size is doubled, and two messages are sent.
- This exponential growth continues until half the receiver's window size is reached and then slows the data size increase rate to linear.



Revision Notes

Applications delivers arbitrarily large chunks of data to TCP as a stream. TCP breaks this data into *segments*, each of which fits into an IP datagram. The original stream is numbered by bytes. The segment contains sequence number of data bytes. The Code Bits specify the nature of the data.

0	4	10	16	24	31
SOURCE PORT			DESTINATION PORT		
SEQUENCE NUMBER					
ACKNOWLEDGEMENT NUMBER					
HLEN	NOT USED	CODE BITS	WINDOW		
CHECKSUM			URGENT POINTER		
OPTIONS (if any)					
BEGINNING OF DATA					
⋮					

PART IV – Network Applications

The Transport Layer gives reliable transfer of data between applications. The Application layer supports:

- initiating connections
- an API
- data encoding
- user friendly namings
- definition of specific applications

The Client-Server Paradigm

This is a very widely used form of communication. The server is invoked on boot up and then waits passively for contact from clients to which it provides a specific service. When a client application initiates contact with the server, information can flow in both directions. Normally, many clients interact with each server.

A *client* is a general application that performs local operations, but that can become a client when remote access is needed. It is invoked by the user and executes for one session. It runs on the user's local machine and actively initiates contact with the server. Although multiple servers can be accessed, only one server at a time can be contacted.

A *server* is a special purpose program that provides a particular service. It can handle multiple clients at once and is invoked when the system boots. It runs through many sessions on a relatively powerful computer.

Clients and servers communicate using the Transport Protocol, unaware of underlying layers. A single server class can run multiple servers, as long as multithreading is supported by the system.

Transport Protocols assign a unique identifier to each service. The server registers its ID on boot and the client specifies the desired ID when making a connection. TCP calls these identifiers *protocol port numbers*.

A Server creates a separate thread for each new client (N.B. n clients would give rise to n+1 threads, as the server needs a thread for itself). TCP uses a combination of the destination and source ports to identify threads. A server can itself become a client of another server (beware of circular dependencies!)

Distributed Programming extends the client-server paradigm providing greater transparency for programmers (remote procedure calls (RPC), distributed objects and components, etc) and by providing standard services for locating and manufacturing other services through uses of traders and factories: A service exports a service offer to the *trader*. The trader then adds this service to its database. The client requests a particular service from the trader, which connects the client to the relevant service. If a client request a service which does not exist, the factory creates the service.

The Socket Interface

An application requests the OS to create a socket. The system returns a small integer descriptor. The application then specifies details such as the Transport Protocol, addresses, etc. The application can use the descriptor as an argument to functions that read and write data.

Server: Socket → Bind → Listen → Accept → Recv/Send → Closes
Client: Socket → Connect → Send/Recv → Close

Each new thread initially inherits all existing sockets from its parents. A thread typically closes sockets it doesn't need. The system maintains a reference count for each socket (how many threads are using it) and terminates the socket when this reaches zero.

Creating and Using a Socket in C

Server:

- **Creating a socket**
`descriptor = socket(protocol_family, connection_type, protocol);`
The descriptor is an integer value. `protocol_family` specifies what type of protocol to use. IP version 4 is specified by the integer constant `PF_INET`. A connection-oriented connection is given by using `SOCK_STREAM` as a connection type. Most protocol families have only one protocol associated with them, so the protocol variable is normally 0, but can specify different protocols.
- **Binding a socket to an address**
`bind(socket_descriptor, local_address, address_length);`
The `local_address` is a structure containing a local address.
- **Listening**
`listen(socket, queuesize);`
The `queuesize` specifies the length of the request queue.
- **Creating new threads**
`newsock = accept(socket, client_address, client_addresslen);`

Client:

- **Connecting to a server**
`connect (socket, server_address, server_addresslen);`

Sending and Receiving Data:

```
send (socket, data, length, flags);  
recv(socket, buffer, length, flags);  
sendto, sendmsg, recvfrom and recvmsg are used with connectionless  
communication.
```

Other functions:

- A server can use `getpeername` to get the complete address of a client
- A server or client can use `gethostbyname` to get information about the host on which it is running.
- `gethostbyaddr` maps an IP to a host name.

Revision Notes

Creating and Using a Socket in Java

Server:

- Creating a socket
`ServerSocket Ssock = new ServerSocket();`
- Binding a socket to an address
`Ssock.bind(new InetSocketAddress (hostStr, portInt));`
- Creating new threads
`Socket connection = Ssock.accept();`

Client:

- Creating a socket
`Socket sock = new Socket();`
- Connecting to a server
`sock.connect(new InetSocketAddress (hostStr, portInt));`

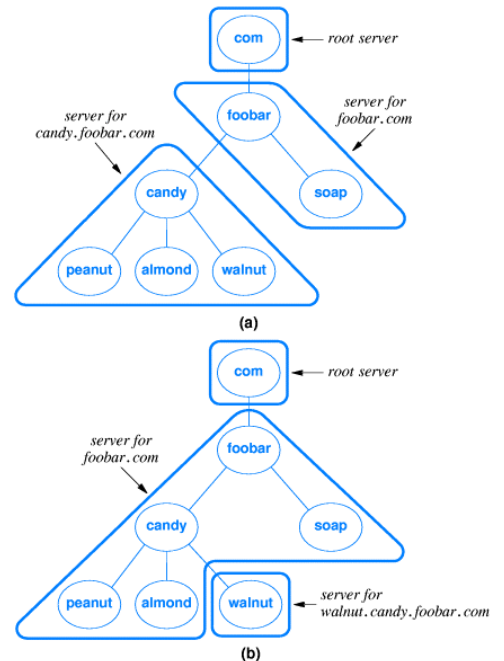
Sending and Receiving Data:

```
BufferedReader br = new BufferedReader( new
    InputStream(connection.getInputStream()));
PrintStream ps = new PrintStream ( new
    OutputStream(sock.getOutputStream()));
```

The Domain Name System

Users can give symbolic alphanumeric names for computers instead of IP addresses. The structure of these addresses is hierarchical with the most significant part on the right. An organization can register with a top level domain and then manage its own subdomains. Some common top level domains are com, edu, net, gov, org.

Applications use the *Domain Name System* to map hostnames to IP addresses. These applications thus become clients of the DNS. The DNS database is distributed across multiple servers. A local server contacts others if necessary. The root servers are the authorities for the top level domains. Each organization can decide how to partition authority among servers (see right). The DNS is not limited to a single level of the hierarchy.



A Domain Name Server which provides for iterative lookup performs resolution using the information within it's own lookup tables. It does not query other name servers for information. When a client request is sent to the server, it searches it's local database, and if it has an answer, it will reply. If it does not have an answer, it will respond with a 'host not found' message. A recursive lookup, however, queries another server if a result is not found. Caching can improve efficiency by exploiting the temporal locality of reference.