3 Foundations of Distributed Systems

Aim: How do we send messages between computers across a network?

Physical Concepts: Bandwidth, Latency

Packet Concepts: Data, Headers

Routing Concepts: Shared Media, Switches and Routing Tables, Routing Protocols

For more detailed notes, see Vladimiro Sassone's *Multimedia Communications Systems* notes in http://www.informatics.sussex.ac.uk/courses/mct/.

3.1 Physical Concepts

What is a message?

- A piece of information which needs to move from one process to each other eg a request to open a file, an email message, the results of a remote print request.

For the network, this is a just a sequence of bits in memory.

Need to communicate this sequence of bits across communications system to other host.

Signal to other side whether each bit in sequence is 0 or 1.

To communicate, need each end to each access a common substrate, and then for the sender to change the value of a physical characteristic of the substrate.

Examples - voltage level of a piece of wire, light level in optical fibre, frequency of radio wave in air

3.1.1 Signal characteristics

If physical characteristic has two levels then signal is binary.

If it has more than one level, can encode a number of bits to each level.

If 4 levels, then level 0 = 00, level 1 = 01, level 2 = 10, level 3 = 11

If 8 levels, how many bits can be encoded?

3.1.2 Bandwidth

Adjusting the level at one end, and recognising the level has changed at the other takes finite time.

The finite time limits the speed at which bits can be signalled.

The rate of signalling is the bandwidth, measured in bits per second.

If it takes 20 microseconds to raise the voltage on a wire, and 10 milliseconds to recognise the new level, what is the possible bandwidth if there are two levels? Eight levels?

3.1.3 Noise and errors

Can we get infinite bandwidth by increasing the number of levels?

No, since noise makes it impossible to correctly distinguish level.

Noise is random changes in the physical characteristic to which all physical phenomena are prone.

Always some probability that level will be misinterpreted.

Always some probability of error in message.

Goal of communications engineers is to make this probabilty as low as necessary.

3.1.4 Latency

Does signal propagate along wire infinitely fast?

No, limit to speed of propagation of light. Hence described as *propagation delay*.

Latency is time taken for signal to travel from one end of communication system to destination.

Since communication system may reconstruct message at intermediate points, time taken to reconstruct message is also part of latency. Known as *switching delay*

Your Task: Describe the bandwidth, propagation delay and switching delay in a game of chinese whispers.

3.2 Packets

If there is an error in a message, how is it detected and rectified?

1. Compute a *checksum* over the message
2. send the checksum with the message
3. Calculate a new checksum over the received message
4. Compare the checksums - if different, then message is in error
5. Ask for the message to be resent

Probability of error rises with length of message,

Thus message sent in separate lumps with maximum number of bits per lump, known as *packets*.

If the message fits in one packet, good. Otherwise message is in many packets.

3.2.1 Addressing

How do we direct the packet to correct recipient(s)?

- Put header bits on front of packet, analogous to address and other information on envelope in postal system. Add source of packet to allow returns
- Destination || Source || Packet body
- If destination and source share the same physical medium, then destination can listen for its own address (ethernet, wireless, single point to point link)
- If they don't share the same LAN, we use *routing*

3.2.2 Names, addresses and routes

Name Identifier of object eg Ian Wakeman, Mum

Address Location of object eg Rm 4C6, COGS

Route How to get there from here - "turn left, first right and up stairs, first door on left"

Internet: Name is a Domain Name, such as www.cogs.susx.ac.uk. More later.

To get packet through network, turn name into address by asking Domain Name Service (DNS).

Place address in packet header and hand to nearest router.

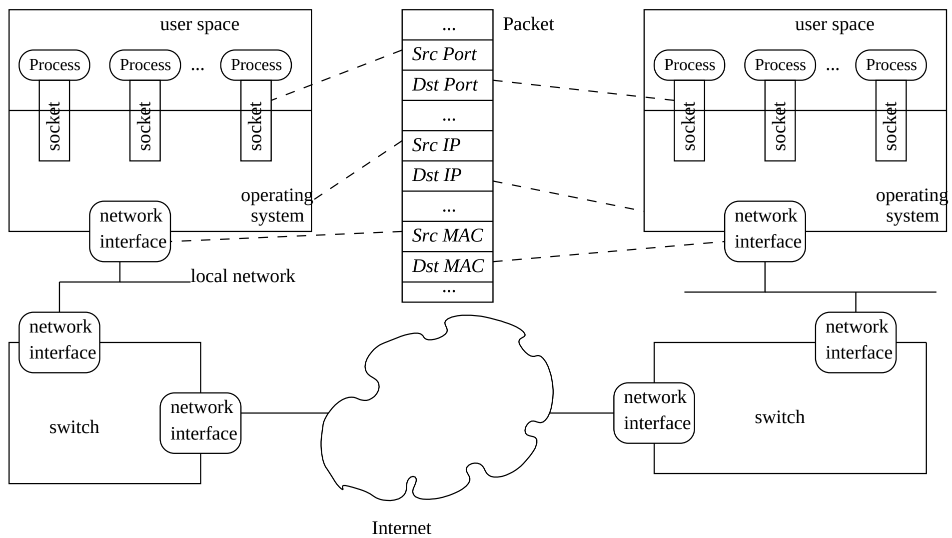
Router locates MAC address corresponding to IP address, if they're on the same LAN.

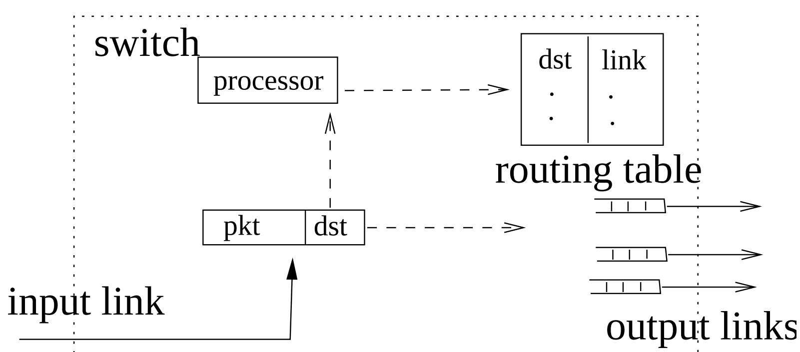
3.2.3 Indirect Addressing

3.2.4 Architecture of a switch

A switch is a specialised computer. Can turn a PC into a router, using free software.

1. A packet arrives on a link
2. The switch gets the destination of the packet from the packet header
3. The switch looks up the destination in a routing table in memory and discovers the output link
4. The switch queues the packet to be sent on the output link


user space
Process
Process
...
Process
socket
socket
socket
operating
system
network
interface
local network
...
Packet
Src Port
Dst Port
...
Src IP
Dst IP
...
Src MAC
Dst MAC
...
user space
Process
Process
...
Process
socket
socket
socket
operating
system
network
interface
network
interface
network
interface
switch
Internet
network
interface
network
interface
switch


switch
processor
dst
link
routing table
pkt
dst
input link
output links

3.2.5 Distributed Routing Table Maintenance

The situation:

- People are network administrators (and end users)
- Communication systems are links (possibly multipoint)
- Machines are switches
- Messages may be lost

The problem: Given an address, how does a router know which output link to send the packet on?

Choices:

1. Packet could contain list of all output links - source routing. Requires source to lookup and determine route. May be good thing.
2. Router could look up address in local routing table, and send out of corresponding link. If not in table, send out default link.

How do we construct tables? Could install entries by hand, known as *static routing*.

But

- limited to entries people know about.
- Unable to ensure consistency and absence of loops
- Unable to respond to changing topology, sharks gnawing through undersea cables etc.
- Internet has no centralised authority

So we use distributed routing algorithm

3.2.6 Distance Vector Routing

1. Each switch knows its own address
2. Each link has"cost", such as a value of 1 per link, or measure of delay.
3. Each switch starts out with distance vector, consisting of 0 for itself, and infinity for everyone else
4. Switches exchange distance vectors with neighbour switches, and whenever info changes
5. Switch saves most recent vector from neighbours
6. Switch calculates own distance vector by examining cost from neighbour and adding cost of link to neighbour
7. Use link with minimum cost to destination as link to route out.

Examples include RIP and BGP.

3.2.7 Link State Routing

1. Each switch knows addresses that are direct neighbours
2. Switch constructs packets saying who are neighbours - *link state packets*.
3. Link state packets flooded to all other switches
4. Switch constructs complete graph using most recent link state packets from all other switches
5. Use Dijkstra shortest path to figure out routing table.

Examples include OSPF.

3.3 Conclusion: Network Properties

Packet Switched Networks present certain fundamental problems to the distributed systems programmer:

- Whilst switches converge on a consistent set of routes, packets will bounce around in the network, and suffer delays or get dropped.
- Switches shared with other packets, therefore get queues.
- Total latency is therefore variable.
- Loss rate is variable (noise, available switch buffers).
- Queue size management (aka congestion control) changes the bandwidth available to machines. Therefore bandwidth is variable.