COMPSCI 314 S1 C

Data Communications Fundamentals

COMPSCI 314 S1 C 2004
Data Communications Fundamentals

Lecturer

- Assoc Prof Peter Fenwick – Room 585, `p.fenwick@auckland.ac.nz`

Tutor

Li Jihong – Rm 596, `jli018@ec.auckland.ac.nz`

Test Date

Monday 3 May, evening

Assignments due

(dates subject to revision)

Thu	11 March,
Thu	1 April,
Tue	27 Apr,
Tue	18 May

Other matters

- Class representative.

- Assignment extensions. I will consider extensions for the assignment due date only for —
    1. Illness or other unforeseeable emergency.
    2. Conflicts with other assignments, but only if the request is made within 1 week of the assignment being distributed.
       I will not be sympathetic if told "The 314 assignment is due tomorrow and I have 3 other assignments also due then; can I please have an extension?" The dates have been published weeks in advance; you must plan your work better or arrange earlier for an extension.
- Your first contact for questions should be the tutor, not the lecturer.
- Email must include the course number (314) and your student ID.
- Cheating and plagiarism. Departmental policy is that all "similar" assignments get zero marks – affected students may appeal by letter to Head of Department (NOT lecturer).
- Each missed (or zeroed!) assignment drops result by one grade.

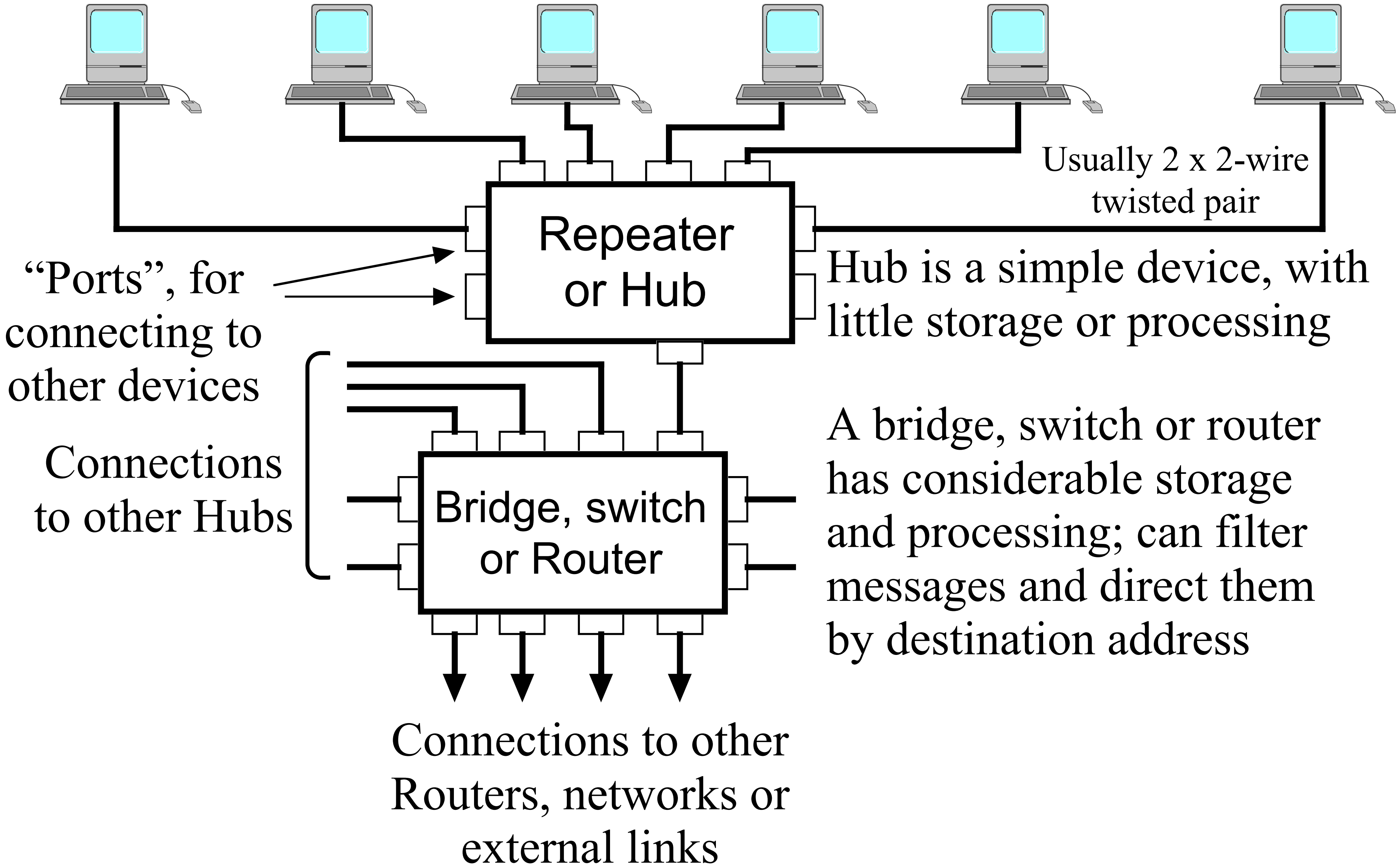
Approach to material

- The usual approach is to work bottom-up through the following order, paralleling the OSI model which we will see later.
  - **Using messages for some user function (more MSIS)**
  - **Getting reliable end-to-end messages through networks**
  - **Directing messages through a network**
  - **Assembling bits into a message**
  - **Sending bits over wire, fibre or wireless**
- We will start more or less in the middle, looking at
  1. the physical components as seen by the user
  2. the components and protocols of a message
  3. sending the message through a larger network
  4. how to send a message over copper, fibre or wireless

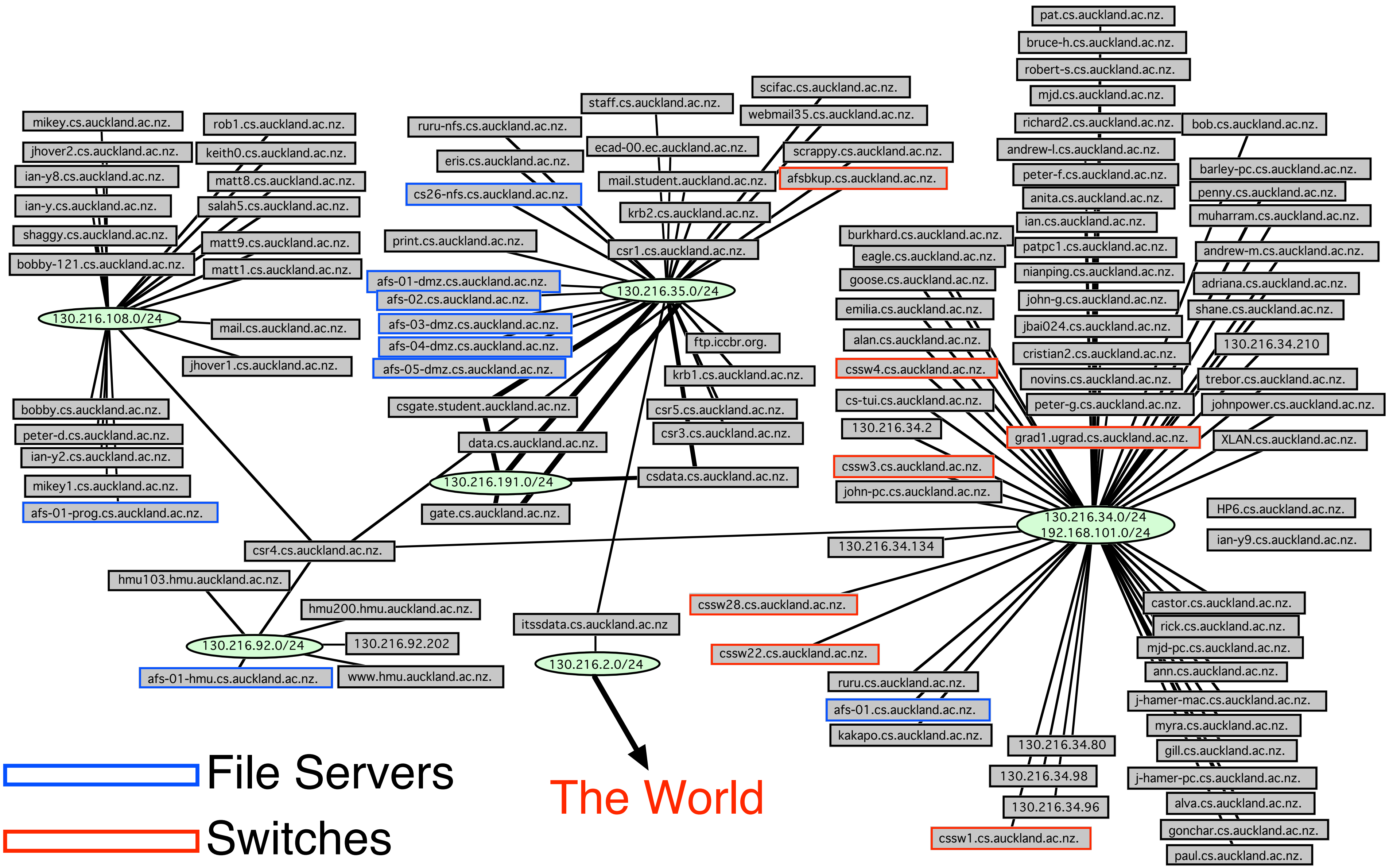
Approximate plan of course

Monday	Tuesday	Thursday	Friday	Ass etc (approx)
1 Mar 2004	1 Introduction	2 Introduction	3 Introduction	
8 Mar 2004	4 Ethernet	5 Ethernet	6 Ethernet	**Ass 1 due Thu Mar 11**
15 Mar 2004	7 Comm Basics	8 Comm Basics	9 Comm Basics	
22 Mar 2004	10 Protocols	11 Protocols	12 Protocols	
29 Mar 2004	13 Protocols	14 Parity & CRC	15 Parity & CRC	**Ass 2 due Thu Apr 1**
5 Apr 2004	16 Parity & CRC	17 IEEE 802.2	18 LAN connect	
12 Apr 2004	——— **Mid Semester Break** ———			
19 Apr 2004	——— **Mid Semester Break** ———			
26 Apr 2004	19 Routing, VCs	20 Routing, VCs	21 Routing, VCs	**Ass 3 due Tue Apr 27**
3 May 2004	22 Routing, VCs	23 IPv4, IPv6	24 IPv4, IPv6	**Test – Mon 3rd May**
10 May 2004	25 IP addressing	26 TCP	27 TCP	
17 May 2004	28 Physical	29 Physical	30 Physical	**Ass 4 due Tue May 18**
24 May 2004	31 Physical	32 Spread Spectrum	33 MIB & SNMP	
31 May 2004	34 MIB & SNMP	35 ATM	36 ATM	
7 Jun 2004	*No lectures — just lots of time to study*		**314 Exam**	

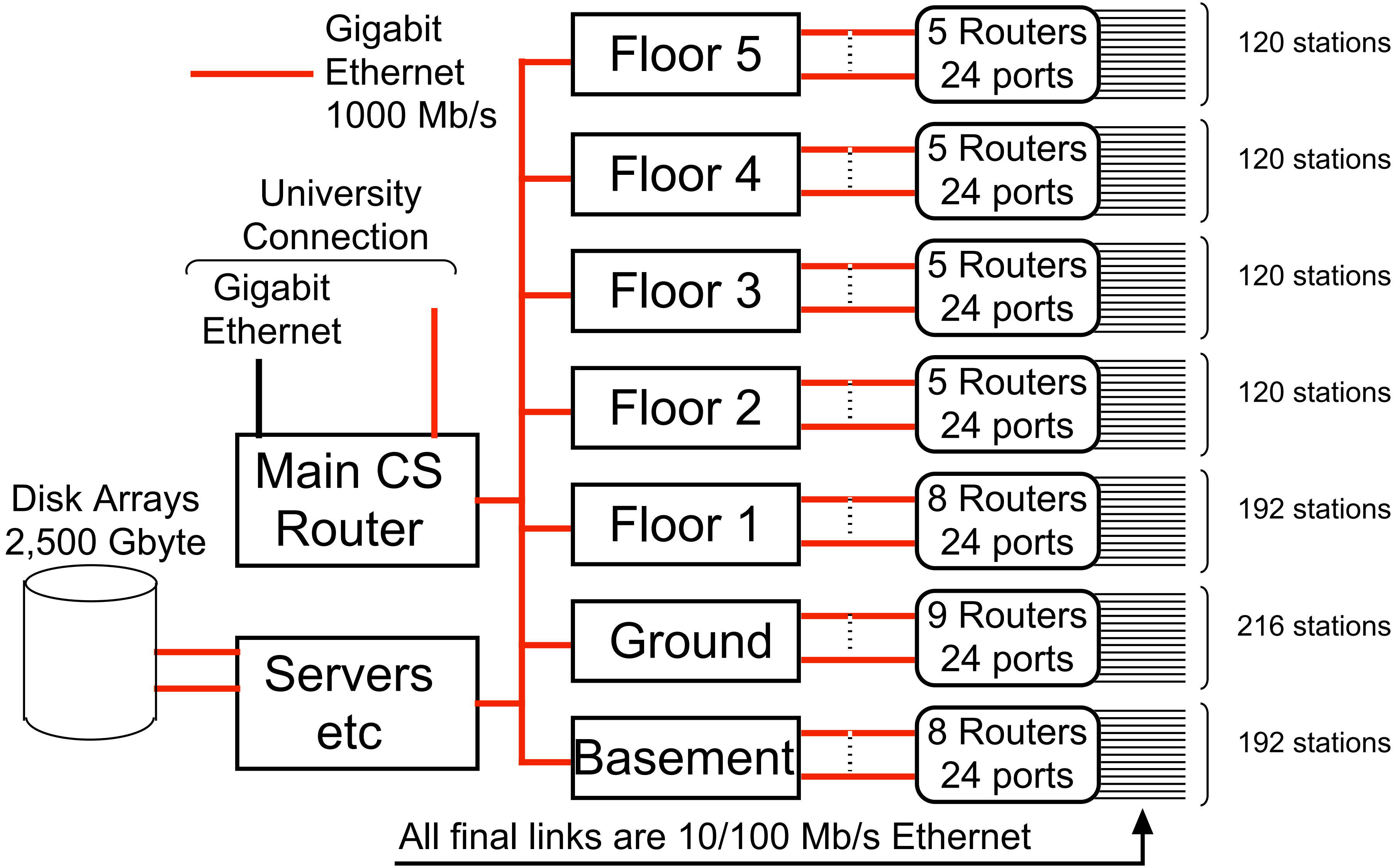
A user connection into a network



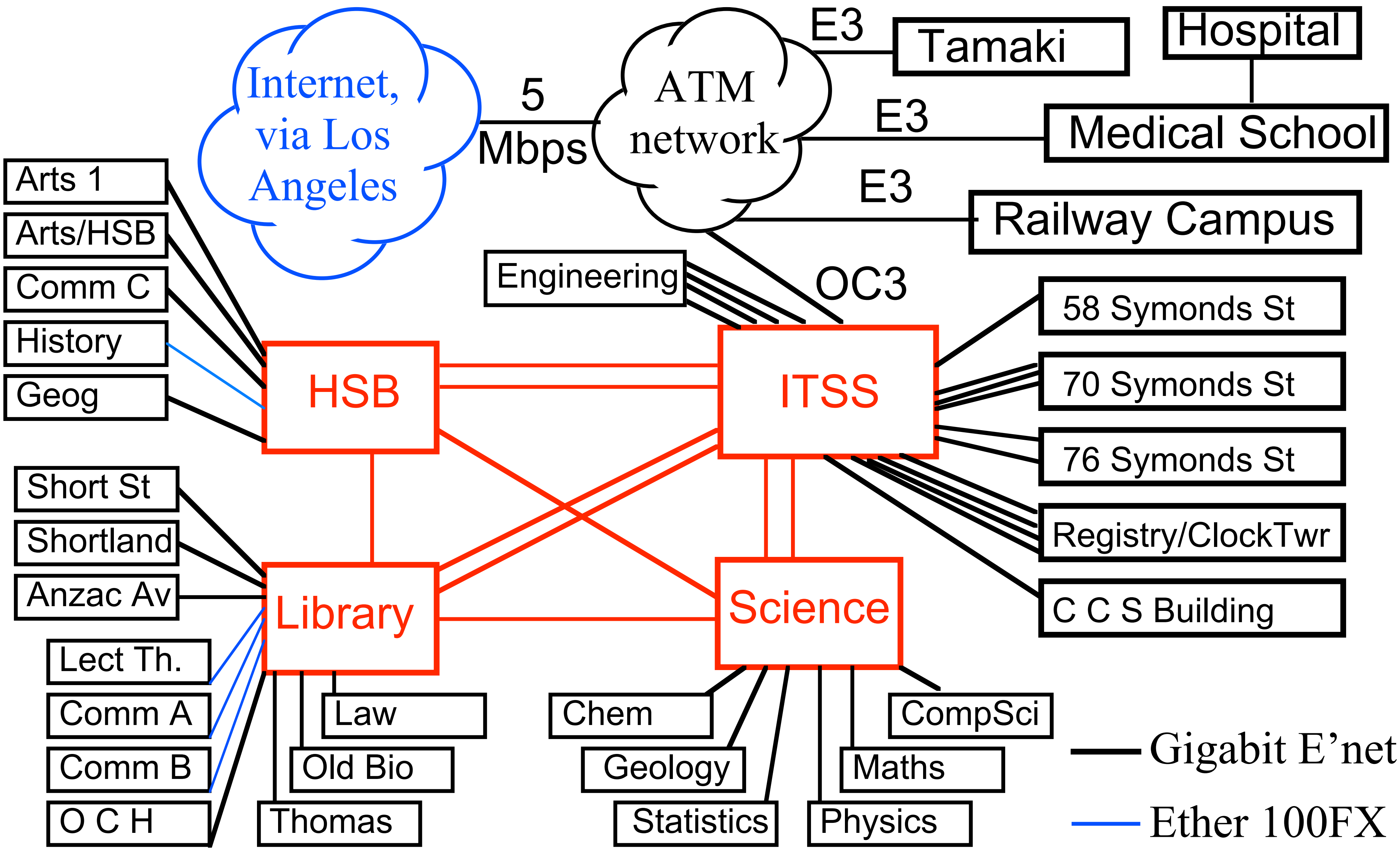
Computer Science Network 2002 (omit Student stations)



Simple view of Computer Science Network, 2004



The University of Auckland Network 2002



Important information on transmission of bits

Bits, as electrical signals, always travel at a "propagation speed" of —

- 300,000 km/s in "free space" (radio, satellites, etc) (30cm per nanosecond)
- 200,000 km/s on copper or fibre-optic cables (20cm per nanosecond)

A "faster" link has the bits arriving *more often* (say 1000 per microsecond, rather than 100 per microsecond), but they *never travel any faster*.

- The circumference of the Earth is 40,000 km (by the definition of the metre)
- The distance from New Zealand to North America, South America, Japan or Singapore is close to 10,000 km.
- The delay or "latency" from New Zealand to almost anywhere except Australia is at least 1/20 second (50 ms). This delay cannot be reduced.

Distances between bits, on optical fibre.

Assume propagation speed of 200,000 km/s in glass fibre

(These distances are nearly correct if the page is printed on A4 paper).

OC-3 (155 Mbps) 1.33 m

1.33 m

OC-12 (622 Mbps) 32 cm

32 cm

20 cm

1 Gbps 20 cm

8.3 cm

OC-48 (2.4 Gbps) 8.3 cm

2.1 cm

OC-192 (9.6 Gbps) 2.1 cm

Message Formats

- The user sees a message as a collection, block or packet of 8-bit bytes. (Communications often prefers the term *octets*, to emphasise 8 bits.)
- The communication system sees a stream of bits, transmitted *serially*, or one after another. (There are a very few exceptions.)
- So, assume that the sequence of bytes or octets can be converted into a stream of bits, with the bits of each byte in order and the most-significant bit of one byte adjoining the least significant bit of an adjacent byte or octet.
- The bit ordering is usually least-significant bit first, bytes in order —

	MSB Bit number, transmission order LSB							
1st byte	8	7	6	5	4	3	2	1
2nd byte	16	15	14	13	12	11	10	9
3rd byte	24	23	22	21	20	19	18	17
4th byte	32	31	30	29	28	27	26	25
5th byte	40	39	38	37	36	35	34	33

Parts of a message

Communications messages usually have —

1. a beginning (the header)
2. a middle (the user data, perhaps with other things as well)
3. an end (the trailer)

A header may contain some of the fields

1. a Start Delimiter to signal the start of the message
2. a Source Address (who sent the message)
3. a Destination Address (who will receive the message)
4. the message type (user data, system control, etc)
5. the message length
6. sometimes other information such as priorities, security control

A message trailer may contain the fields

1. a checksum, parity, or Cyclic Redundancy Check, to ensure data validity.
2. a continuation to say “more messages in this sequence”
3. an End Delimiter to mark the end of the message

But these are only guidelines. Some messages omit some of the fields and some even have the length at the end!

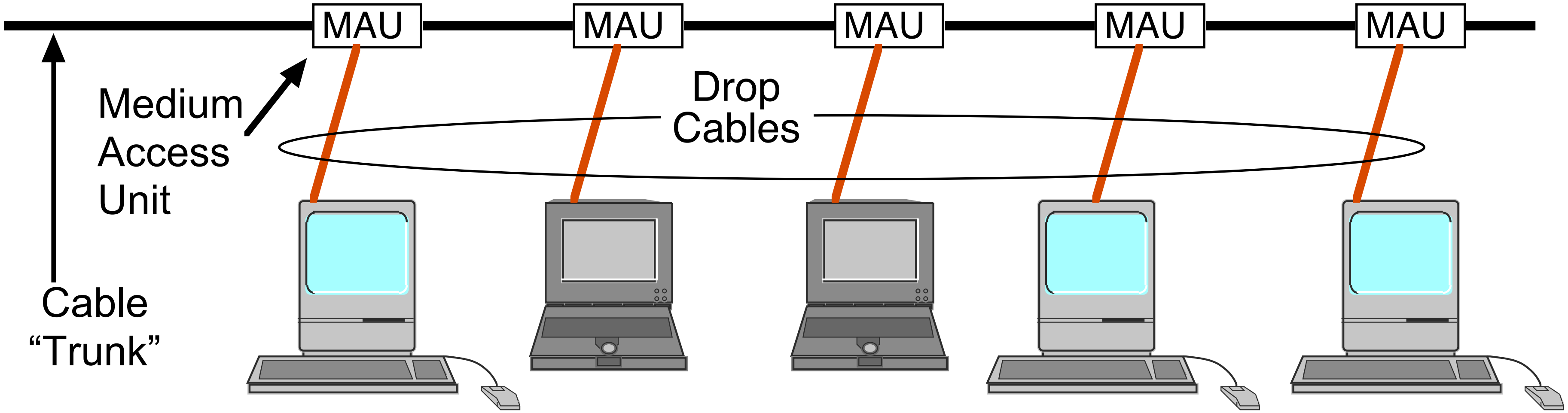
Alternative names. All the following mean pretty much the same thing, but sometimes one term is preferred and sometimes another.

- Message   more as seen by the user
- Frame   more as viewed by the physical transmission
- Packet   more as seen by the network operation

Example of an actual message format

Most of the internal communication within the Computer Science network uses "Ethernet" or IEEE 802.3 messages, at 10, 100, or 1000 Mbit/s.

- Traditional Ethernet/802.3 uses a "multidropped" cable trunk. All stations can transmit on to the one cable and all listen to all the traffic.
- Roughly speaking, if a station wants to send it listens and waits if the "medium" is busy, "deferring" until there is no traffic – gory details later.



Modern Ethernet is really quite different, but the appearance to the user remains the same.

Ethernet / IEEE 802.3 message format

56 bits	Preamble	1010101010…
8 bits	Start Delimiter	10101011
48 bits	Destination Address	
48 bits	Source Address	
16 bits	length (< 0x600) or type (> 0x600)	
8*N bits	Payload, or data	46 ≤ N ≤ 1500
32 bits	Frame Check Sequence	

or an alternative picture

1010... (56 bits)	10101011	48 bits	48 bits	16 bits	8*N bits	32 bits
Preamble	Stt delim	Dest Adr	Src Adr	Length	Data	FCS

- The Preamble allows the receiver to “lock on” to the bit pattern
- The Start Delimiter continues the preamble; a final “1” starts the message
- The Destination Address is the address to which the message is sent
- The Source Address is the address of the station sending the message
- The length is number of actual data bytes (or octets) in the message
- The data is the user data. Very short messages are “padded” to 46 bytes.
- The FCS (Frame Check Sequence) is a parity check on the message.

There are two cases for the length/type field –

- Value < 1536 (0x600 – old IEEE 802.3) indicates the number of valid bytes. A special LLC/SNAP header is needed to indicate the message type).
- Value > 1536 (old Ethernet) indicates the type of message.

Addresses

- Ethernet addresses are always 48 bits (we never use the alternative 16-bits)
- The address is further structured (~~remember least-significant bit first~~) —

I/G	U/L	46 bit address

I/G	0	Individual address
I/G	1	Group address
U/L	0	Globally administered address
U/L	1	Locally administered address.

- The addresses are unique; part of the 48 bits (U/L=0) is a manufacturer code and part is a unique device number allocated by the manufacturer.
  No two legal 802.3 devices ever have the same address.
- Some systems derive local addresses from other addresses; setting U/L=1 avoids any confusion here (but this is rare).
- See later section on bit ordering

Group addressing

- Most stations have I/G=0 for an individual address.
- If a known group of stations is to receive identical traffic (eg disk loads to student computers or video data to multiple screens) the stations can be put into a multicast group and all allocated the same group address (I/G=1).
- Then one station can send to that group address and all stations in the group will recognise the address and accept the traffic.
- The special all-1s “broadcast address” is defined to mean “all stations on this network”.
  It is useful for enquiries such as “Is there any station with this name ?”.
- Some networks (not 802.3) define the all-0s address as a null address which goes to nobody but is useful for testing.

Ethernet bit ordering

- Bytes or octets are transmitted *most-significant* first – the transmission order corresponds to ordinary left-to-right reading of the digits (in hexadecimal).
- Bits within an octet are transmitted *least-significant* bit first, as in normal serial transmission.
- As the I/G and U/L bits of the 48-bit address are the *first bits transmitted*, they are the least significant (right-most) bits of the first address octet.
- This diagram gives the bits in computer order, and then the order in which they are transmitted.
  Assuming that the bit stream is the start of an address, the diagram also shows the position of the I/G bit and U/L bit.

Octet boundaries	**Octet 1**								**Octet 2**								**Octet 3**								**Octet 4**								**Octet 5**							
Bits, "computer order"	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
Bits, as transmitted	8	7	6	5	4	3	2	1	16	15	14	13	12	11	10	9	24	23	22	21	20	19	18	17	32	31	30	29	28	27	26	25	40	39	38	37	36	35	34	33

8 I/G bit
7 U/L bit

The user data or payload

- When we look later at the full Ethernet protocol we will see that it is desirable to have both a minimum and a maximum message length.
- As the minimum frame length is 64 octets (including header and FCS, but excluding preamble) very short messages may need “padding” or filling out so that the data is at least 46 octets.
- A user message of 16 octets would be followed in the Ethernet frame by 30 octets of “fill” or “pad” (usually zeros) and have a length field of 16.

Carrying other data – encapsulation or tunnelling

- Often a network such as Ethernet or IEEE 802.3 must carry some other sort of data, such as IP or AppleTalk.
- With "Ethernet style" (type/length > 0x600) we put a "well known" value in the type field to identify the embedded protocol and then put the "foreign" packet in the data.
- With "802.3 style" (type/length < 1500) we must first "encapsulate" the foreign packet, giving it a special header to say what it is and then sending the combination of header and user data as data in the 802.3 frame.
- Similar techniques may be used to carry newer types of traffic over older systems (or vice versa).
  For example most Internet traffic uses Internet Protocol version 4 (IPv4), but there is a new standard called IPv6.
  If only an IPv4 link is available the IPv6 message may be encapsulated and "tunnelled" through the older system, or *vice versa*.

Sub-Network Access Point (SNAP) headers

- When carrying "strange" data over IEEE 802.3 with *length indication* the strange data is preceded by an 8-octet header

  1. The first 3 octets are an LLC (Logical Link Control) header, with the (hexadecimal) values `AA-AA-03`. The two `AA` codes specify that a "SNAP" header follows and the `03` says "this is data" (not control).

  2. The following 5-byte SNAP Header has a 3 byte (24 bit) code giving an "administering authority" and 16 bits for the protocol (08-00 for IP).

- The full prefix is then

```
0x AA-AA-03             LLC Header
0x 00-00-00 08-00       SNAP Header
```

- When transferring AppleTalk the SNAP header value is

  `0x 08-00-07 80-9B` and the full header is

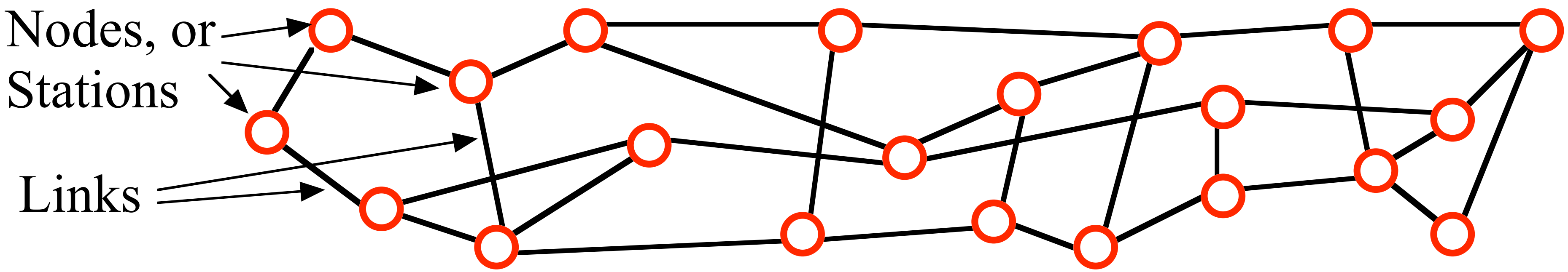
```
0x AA-AA-03             LLC Header
0x 08-00-07 80-9B       SNAP Header
```

LANs and WANs (and MANs and DANs)

- Local Area Networks (LANs) are privately administered, serving a single campus, single building or even single office.
- They mostly use Ethernet, (10, 100 or 1000 Mbit/s), but sometimes token rings, ATM or some other technology.
- Traditionally faster than Wide Area Networks (WANs), but not always.
- They generally use some form of multidrop or broadcast network for easy communication among many stations (hundreds in Computer Science);
- Any station can connect directly to any other (but bridging, routers, virtual LANs may affect this).
- LANs are usually structured as a tree, or sometimes a few parallel trees with well-chosen cross-links for redundancy and failure protection.

LANs and WANs (and MANs and DANs)

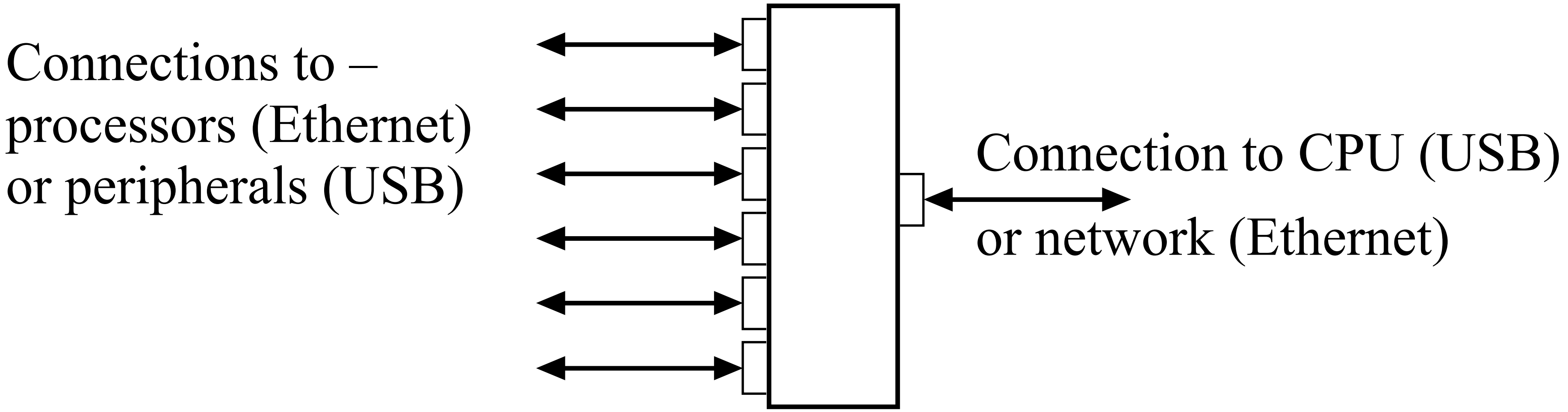
- Wide Area Networks (WANs) are long distance (say > 5 km), usually provided by or leased from some public carrier (eg Telecom, Clear).
- Internally the network is usually a mesh of point-to point links; a connection between two nodes or stations involves finding a path through the network. Stations usually have *logical* addresses, different from physical addresses.



- Metropolitan Area Networks (MANs) resemble LANs, but cover 10’s of kilometres rather than kilometres to extend over a city.
- Desk Area Networks (DANs) are more of a cute term describing the connection of many peripherals to a computer (seldom used).

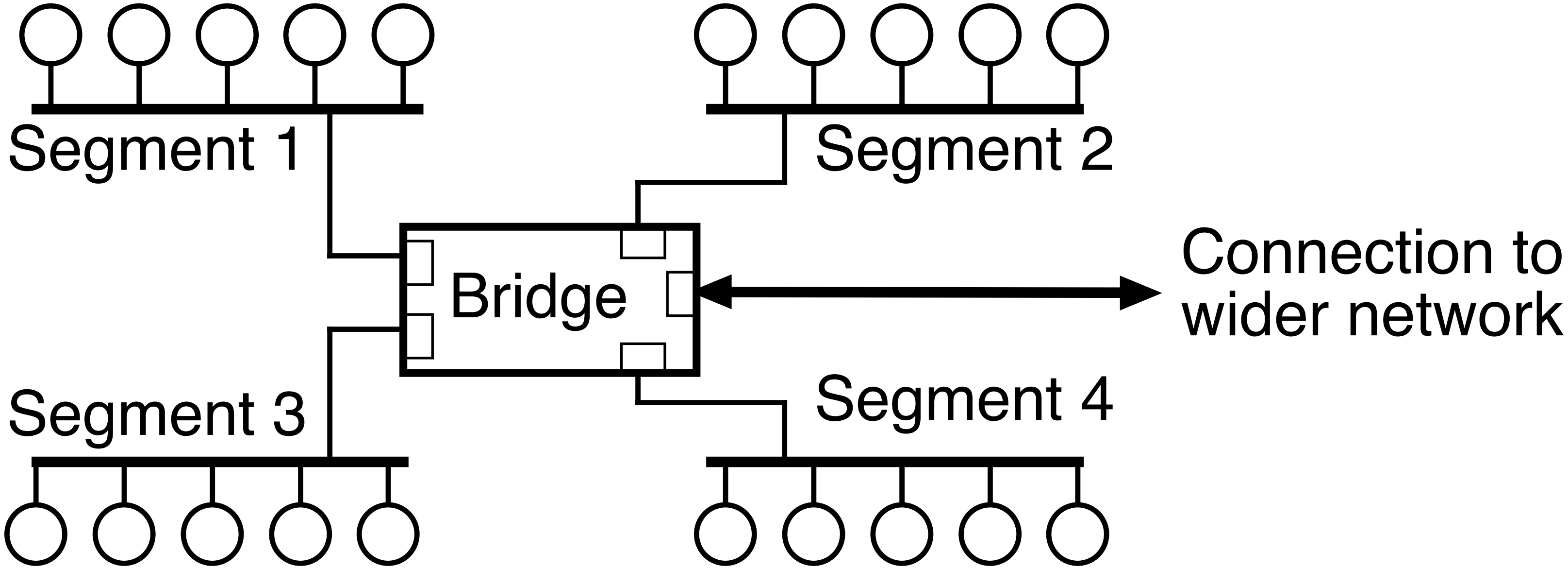
Connection boxes

- Networks are full of boxes which connect many things to one (usually one nearer the centre of the network), or sometimes many things to many things.
- Repeaters or hubs operate at the bit level, sometimes enabling longer connections, but often allowing many devices to use one connection. The connection is generally time-multiplexed among the devices; only one can work at a time.



- Remember that all the “boxes” connect many things to one or a few things and look similar; hubs are physically small, switches and routers larger.

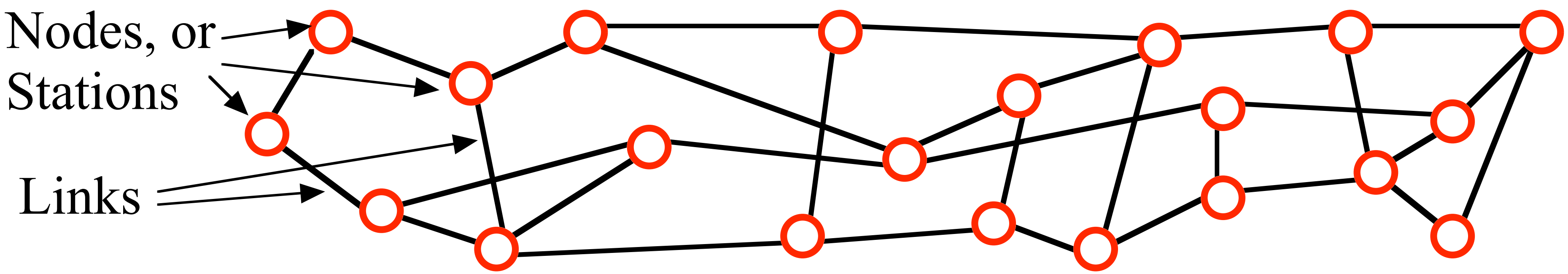
- Bridges (and some switches) operate within a network, dividing it into "segments".



- Bridges recognise frames and process *physical* addresses —
  - traffic within a segment is not repeated to other segments,
  - traffic between segments appears only on those two segments.
- Groups of addresses, scattered among segments, can be configured into "virtual LANs", separating stations on the same physical network.
- Nowadays the term "switch" often includes bridges and routers.

Routers recognise protocols (eg AppleTalk and IP) and *logical* addresses.

- Routers can filter messages by address and protocol, such as keeping all AppleTalk within the department and not releasing it into the University network, or separating traffic between laboratories and staff.
- Wide Area Networks (WANs) are usually point-to-point connections between routers. (A WAN router may be an entry to a complete LAN.)
- A router receiving a message on one *input* port must somehow decide the best *output* port to get the message closer to its destination.



- Routing messages is a major task of WANs — not just any path, but the shortest path, the cheapest path, the path with least congestion, etc, etc.

Things to do in a network

1. Transmit bits from one place to another
2. Assemble bits into bytes and messages, check for reliable transmission
3. Send messages between end-nodes in a mesh-type network
4. In a mesh network, handle lost packets, broken links etc
5. Handle extended connections between endpoints, LANs, etc
6. Resolve differences between data representation in different computers
7. Do something useful (User application)

These are the seven layers of the "Open Systems Interconnection" (OSI) communications model.

We discuss only levels 1–4 (leave levels 5–7 to MSIS!)