Computer Networks: A programmer's perspective

What you should already know about

- OS processes
  - ∗ In Unix: fork, exec, waitpid.
- File input and output
  - ∗ In Unix: open, read, write, close. File descriptors.

Reading

Chapter 11 of Computer Systems: A Programmer's Perspective.

Outline of this section

- Communication between processes
- The OSI model
- The Ethernet/internet model
- Ethernet
- Internet
- TCP
- The Sockets API
- Writing a client with sockets
- Writing a server with sockets
- Designing distributed applications

Communication between processes

[Language note: The fancy words for "communication between processes" are "interprocess communication" sometimes abbreviated "IPC".]

You may have seen some communication between processes on the same host.

For example Unix's pipes allow information written by one process to be read by another

```
ps aux | grep root | more
```

How can processes on different hosts communicate with each other?

For this we need networking.

The same networking methods can also be useful for processes on the same host to communicate.

The OSI model

A common way of looking at networks is the OSI 7 layer reference model.

Typically the operating system provides the first 4 layers.

1	Physical	Move bits across some physical medium (wire, optical fiber, radio) Ethernet physical layer, IEEE 802.11
2	Data link	Move data frames between 2 physically connected nodes Ethernet, IEEE 802.2
3a	Network	Move data frames around a local area network (LAN) Ethernet
3b	Internetwork	Move packets between hosts on different LANs IPv4, IPv6
4	Transport	Move streams or datagrams between ports TCP, UDP

Applications can implement higher layers above the transport layer.

We look at both sides of the application/OS interface:

- How do OSes provide services like TCP and UDP to applications?
- How can applications be written to use TCP or UDP to

implement network applications.

The Ethernet/internet model

Internet technology allows computers (hosts) on different networks to communicate.

- For example a computer on a university Ethernet network
- and a computer on a home WiFi network.

In the next few slides I'll give a quick overview of

- How Ethernet implements local area networks.
- How IP4 allows computers on different local area networks to communicate.

Note that the Ethernet/internet model and the OSI models do not quite line up with each other.

- This is why I split OSI layer 3 into two sublayers.

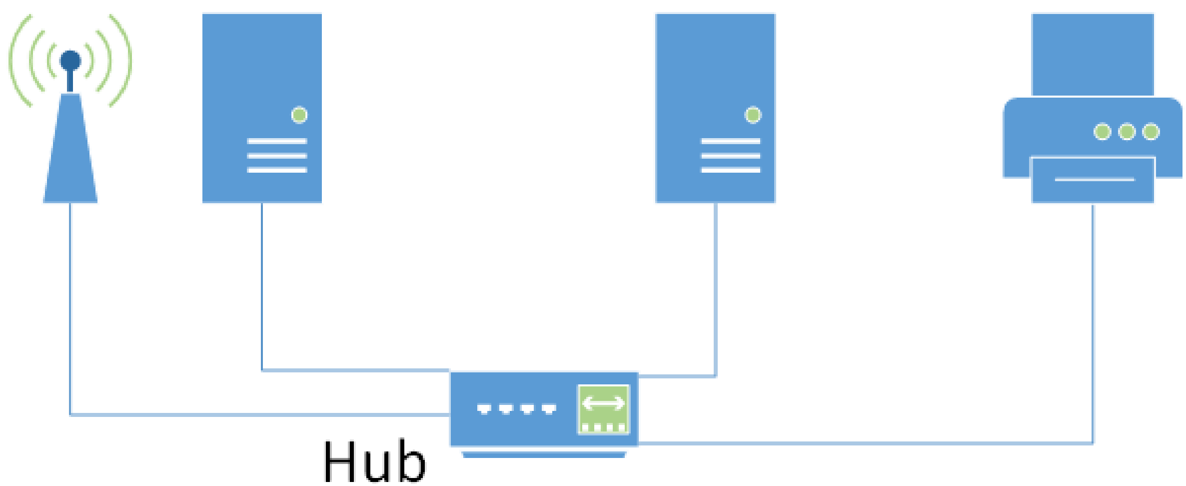
Ethernet

Ethernet implements layers

- 1 (Physical)
- 2 (Data link)
- 3 (the network model within a single Local Area Network (LAN).

Ethernet physical layer

Ethernet uses twisted pair cables (e.g., CAT 6) to connect the Ethernet adapter (i.e. network card) on a host to a hub.



The set of hosts connected to the same hub is called an <u>ethernet segment</u>

All bits sent by any host to the hub are seen by all hosts in the same segment.

The hosts cooperate (e.g. by backing off when two try to send at the same time) to share the bandwidth of the segment.

Ethernet datalink layer

Each ethernet adapter has a globally unique address —it's MAC— 6 octets.

The unit of transmission in a ethernet network is an <u>ethernet data frame</u>: a sequence of octets (8-bit bytes) consisting of (among other things)

- The destination MAC
- The source MAC
- A length integer
- A payload of up to about 1500 octets.
- A check sum (CRC)

ethernet header	payload	CRC

When a network adapter sees a segment with a destination MAC address

- equal to its MAC address, it stores the frame and interrupts the OS.
- with a different MAC address, it ignores the frame.

So, within a segment a set of computers can exchange data frames.

This implements layer OSI 2.

[See animation slides.]

For physical reasons, each segment has a limited size ($<$ 100m from host to hub).

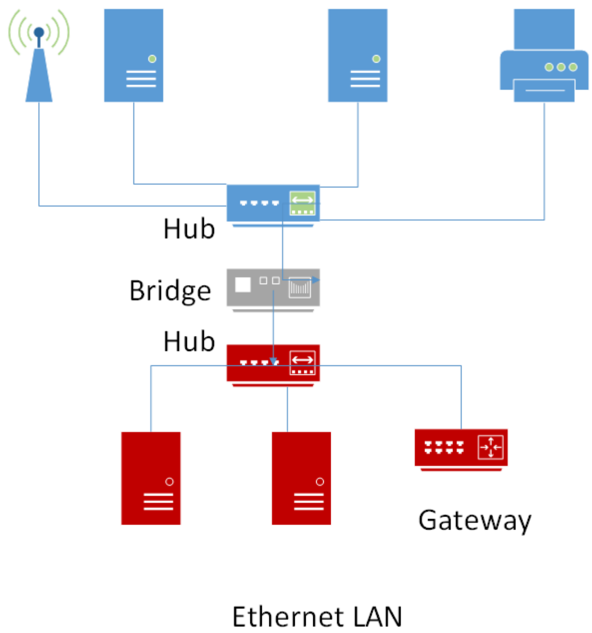
To reduce collisions, the number of hosts per segment should be limited.

[Discussion: Ethernet limits the payload size to about 1500 octets. What are the advantages of a larger limit? What are the advantages of a smaller limit?]

Bridges

Multiple segments can be connected by bridges

- A bridge has a network adaptor on at least 2 segments.
- A bridge looks at all data frames on a segment.
- When a data frame's destination MAC is for an adaptor on the other segment, it stores the frame
- and forwards the frame to another bridge or segment.



[Exercise: Draw a picture or series of pictures illustrating how a data frame travels from a host on one segment to a host on another.]

[See animation.]

Bridges can do some simple routing so that if host A sends to host B on a different segment, the data frame is only seen on their two segments.

So bridges extend the datalink (layer 2) beyond physically connected computers and they implement simple networking (layer 3).

Bridges allow you to unplug from one segment and into another. So they must learn which MAC addresses are on which side of the bridge.

[Discussion question: Consider a graph in which segments and bridges are nodes and a segment being connected to a bridge is an edge. What properties must the graph have if each MAC address is unambiguously on one side or another of each bridge.]

LANs

All together, the set of hosts, hubs, and bridges form a Local Area Network (LAN).

Any host on an ethernet LAN can communicate with any other host on the same LAN by sending it a data frame.

Within an ethernet LAN, the MAC address identifies the host.

Connecting LANs (the internet protocol)

Connecting just 2 LANs

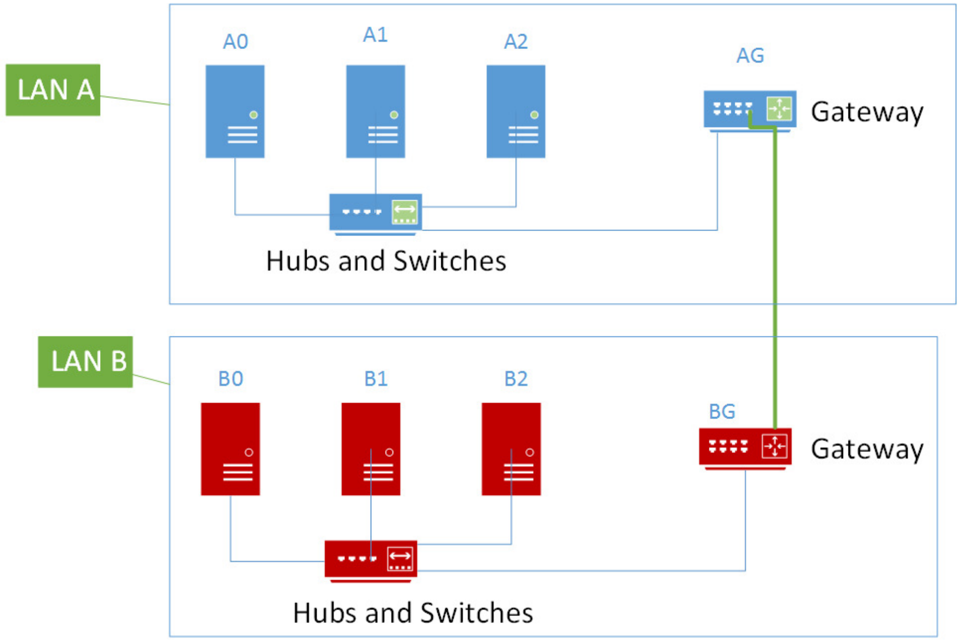
Suppose you wanted to connect computers on **two** LAN networks.

These LANs might use different implementations of layer 2.

- E.g. one could be ethernet and the other WiFi

We could put a special device on each LAN that detects when data frames contain data destined for computers on the other LAN.

Such a device is a gateway.



This picture looks a bit like the one with a bridge.

But there are important differences:

- Each LAN has its own gateway.
- The two LANs might use completely different implementations
- Even if both are ethernet:
  - ∗ Addresses may conflict between networks.
  - ∗ The connection between the gateways may not be ethernet.

Typical scenario:

- Host A0 sends (by LAN) a data frame to AG
- AG sends the payload to BG (somehow)
- BG makes a new data frame and sends it (by LAN) to B2

[Exercise: Draw a picture or a series of pictures illustrating this scenario.]

But.

- How can a computer on network A address a packet to a computer on network B?
  - ∗ MAC addresses are fine for ethernet, but the other LAN might not be using ethernet.
- The data frames on the networks might have different sizes.

Internet protocol to the rescue

Under the internet protocol version 4 (IPv4), each host has a 32 bit address.

- Addresses are written as 4 decimal numbers from 0 through 255. E.g. 42.139.17.9

[Language note: “inter-” means “between”. “net” is short for “network”.]

Computers communicate via packets of length up to 65kB.

IP header (20 bytes)	payload

For example

From: A0 To: B2 Len: 30 Id: 123...	To turn from darkness to light
IP Header	IP Payload

Each fragment is the payload of a LAN data frame: E.g. on ethernet we have

Ethernet header	IPv4 header	payload	CRC
	Ethernet payload		

It is a riddle wrapped in a mystery inside an enigma — Winston Churchill

How do the following pictures relate to LANs and the internet?

Splitting packets

A single packet may not fit in a LAN data frame payload.

It can be split into multiple fragments.

From: A0	To: B2	Len: 10	Id: 123	Offset: 0	More: 1	To turn fr
From: A0	To: B2	Len: 10	Id: 123	Offset: 10	More: 1	om darknes
From: A0	To: B2	Len: 10	Id: 123	Offset: 20	More: 0	s to light

Typical scenario

- A0 makes a packet and addresses it to B2
- A0 breaks its packet into fragments
- A0 sends each fragment to AG inside a data frame
- AG sends each fragment to BG, fragmenting more if needed.
- ~~BG breaks the packet into fragments~~
- BG sends each fragment to B2,, fragmenting more if needed.

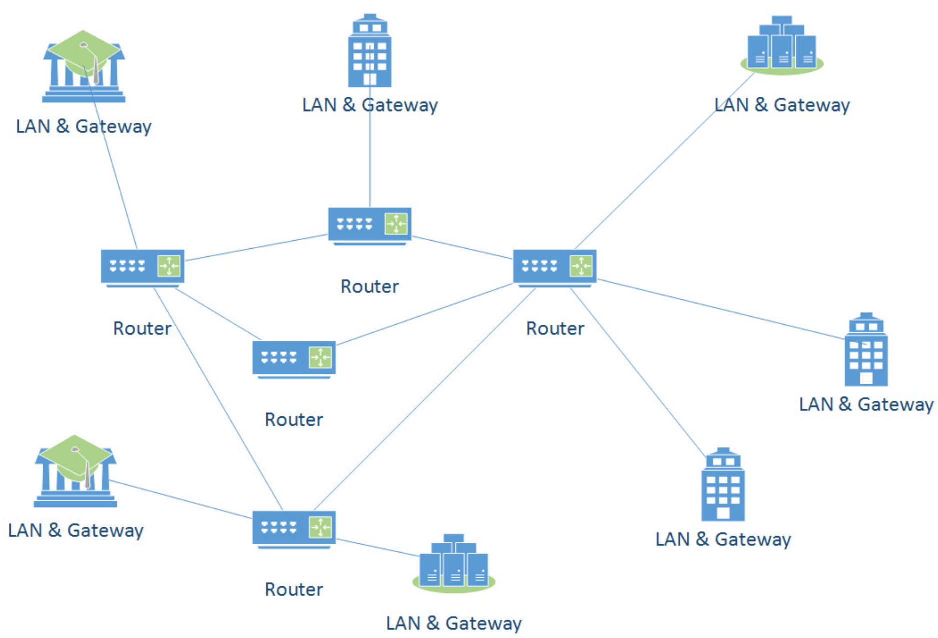
- B2 collects the fragments and assembles the packet

[Exercise: Draw a picture or series of pictures illustrating this scenario. Suppose the packet size is 1000 bytes. Network A has data frame payloads of 800 bytes. The connecting link has data frame payloads of 500 bytes and network B has data frame payloads of size 400.]

Connecting many LANs

Once we can connect two LANs, we can connect many LANs through networks of routers and long distance connections — typically optical fiber.

Most LANs in the world are currently connected to form The Internet.



How do routers know where to route a given packet?

- A MAC address is (usually) built into the device and does not change.

- An IP address, on the other hand, is assigned when the host joins the network.
- Part of the IPv4 address identifies the LAN (or subnet).
- The remainder identifies the host within the LAN.

$$\underbrace{42.139.17}_{\text{Address of LAN}} . \underbrace{9}_{\text{Host}}$$

What if a router of link fails?

- The internet protocol is designed to be resilient against partial failure.
- If packets can not be delivered, the routers attempt to find alternative routes.

[Discussion question: Why is there only one global network? Would it make sense to try to create a better, but incompatible, network?]

[Exercise: Suppose every person on Earth has one device that needs one IP address. Could we use IPv4?]

Internet Protocol version 6 (IPv6)

IPv6 performs the same functions as IPv4, but allows 16 byte IP addresses.

China in particular has promoted the growth of IPv6 infrastructure.

https://en.wikipedia.org/wiki/China_Next_Generation_Internet for more.

From here on I'll use IP to mean either IPv4 or IPv6.

[Exercise: Suppose that in year 2100 the Earth's population is 20,000,000,000. Approximately how may IPv6 addresses would there be per person?]

[Discussion question: An 'open' system has properties of interoperability, portability, and open standards. The internet is sometimes said to be an 'open system'. What technical properties of IPv4 contribute to it being an open system?]

[Exercise: The IP standards allows a diversity of applications to run on a diversity of platforms. The UNIX interface(s) play a similar role. Can you draw a picture that illustrates this similarity between IP and UNIX? Are there other systems that follow the same pattern?]

[Question: What other examples do you know of in which a fixed length field has lead to problems?]

End of networks-00