Chapter 6
The Internet/Network Layer: IP

The Internet/Network Layer

- The basis the Internet is built upon
- Very simple, but allows more complex stuff to be layered on top
- Shall describe IP version 4, IPv4
- Talk about Ipv6 later

The Internet/Network Layer

- A best-effort, connectionless, unreliable, packet based protocol
  - Note: “unreliable” means “not guaranteed reliable” -- unreliable networks are often very reliable these days

The Internet/Network Layer

- A best-effort, connectionless, unreliable, packet based protocol
- Represents the lowest common denominator of network properties
- We don't rely on any particular property of a link layer, so can run on top of almost any link layer

The Internet/Network Layer

- IP is a cooperative system: for a packet to get from source to destination it is handed from one network to the next, hop by hop
- No single machine anywhere has any idea what the entirety of the Internet looks like

The Internet/Network Layer

The nodes in the network have various roles:

- Host. A machine you actually use to do some work
- Gateway. Connects two networks together
- Router. A machine whose primary function is to determine where a packet goes next

These are not mutually exclusive: gateways and routers can be hosts; gateways do trivial routing

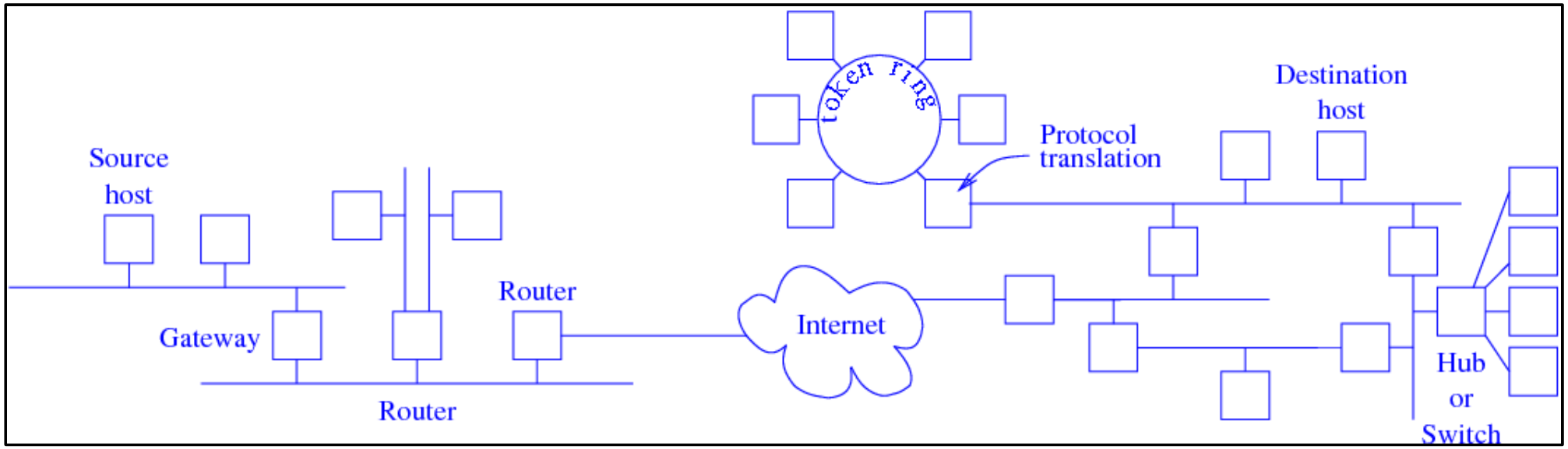
The Internet/Network Layer

The basic idea is that a packet does not know how to get from source to destination: this is the routers' job (and it can be quite complex: see later)

The IP layer breaks the data stream into packets, called *datagrams* in this context, and prepends a header

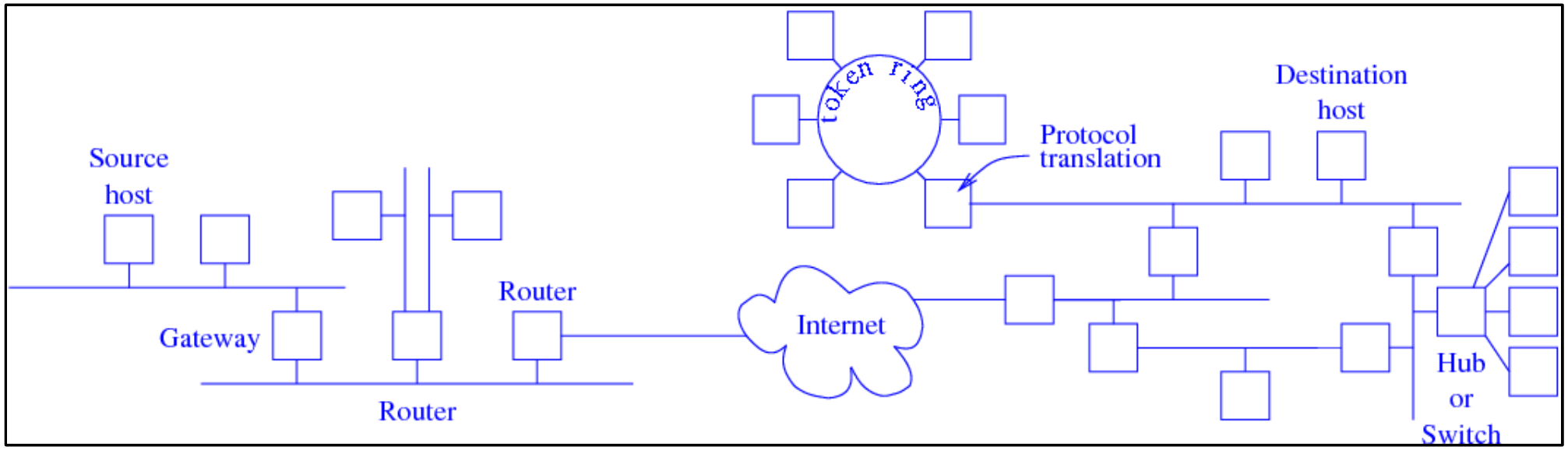
Datagrams can be up to 64KB in size, but are usually no larger than 1500 bytes (Ethernet, again)

The Internet/Network Layer



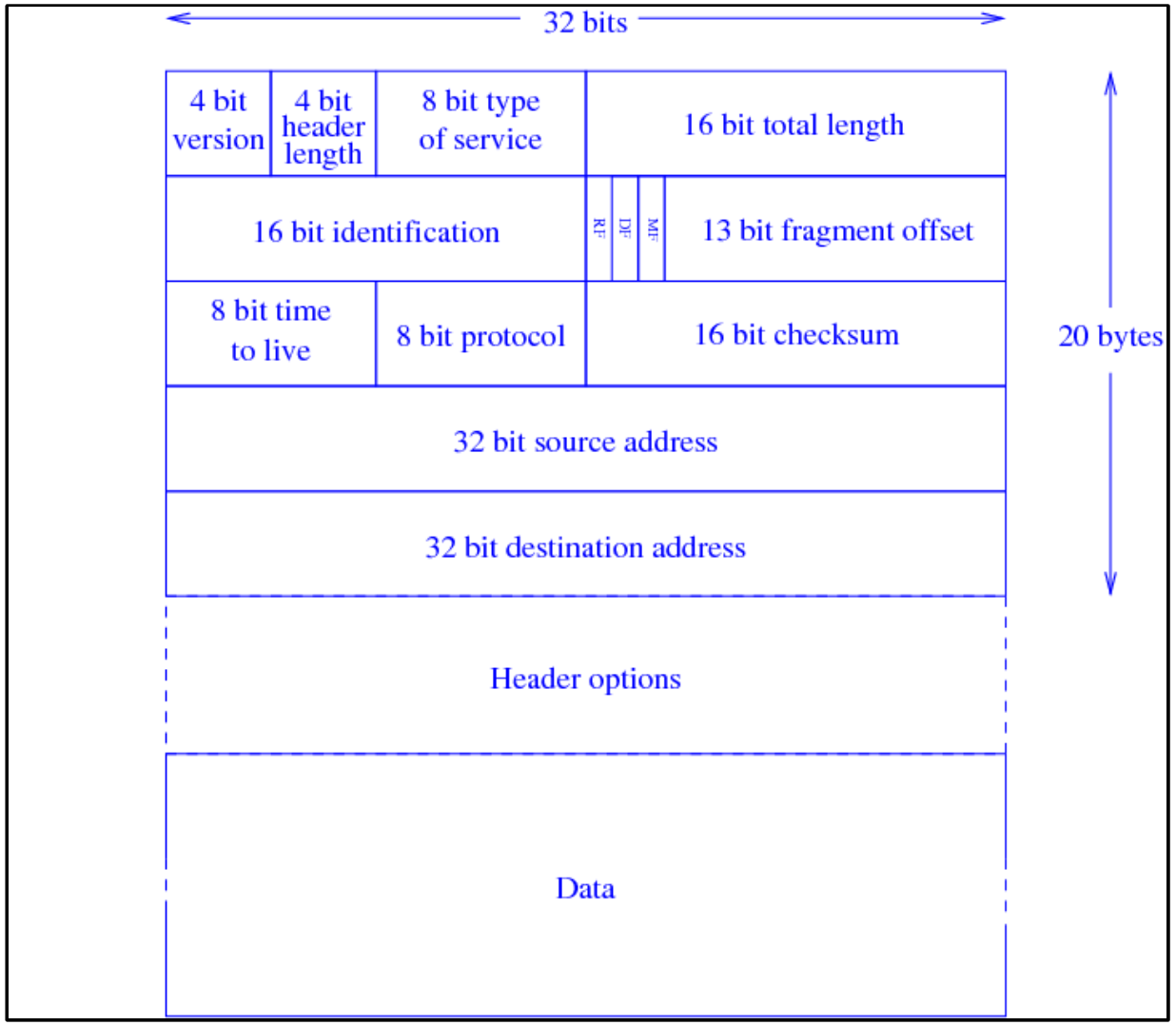
As IP runs over many different kinds of hardware, it must face the problem of differing link layer properties, in particular maximum packet size

The Internet/Network Layer



IP deals with this by *fragmentation*: a packet is subdivided by a router into several smaller packets; it is the the destination's problem to glue them back together

The Internet/Network Layer



IP datagram header

The Internet/Network Layer

IP datagram header

- Version. Four bit field containing the value 4. A later version of IP (Ipv6) contains 6
- Header length. The header can vary in size, so this is needed to distinguish the end of the header. Given as a number of 4 byte words. Four bits, maximum value 15, so maximum header length of 60 bytes

The Internet/Network Layer

IP datagram header

- Type of service. Eight bits allocated (not all used). To indicate to a router how this datagram should be treated in terms of cost, speed and reliability (if possible)

  e.g., for audio it is better to get data through quickly rather than 100% reliability as the human ear is more sensitive to gaps than occasional errors

The Internet/Network Layer

IP datagram header

- Type of service. Four bits:
  1. Minimise delay. Do not hold onto this datagram longer than necessary, and perhaps prioritise it over others
  2. Maximise throughput. Not quite as minimising, as collecting together several small datagrams and sending them off together may be more bandwidth efficient

The Internet/Network Layer

IP datagram header

- Type of service. Four bits:
  3. Maximise reliability. Try not to drop this datagram if the router is becoming overloaded; drop another datagram first
  4. Minimise cost. Cost is more important than reliability or speed. This packet can be delayed if it makes transmission cheaper

The Internet/Network Layer

IP datagram header

- Type of service. Routers can ignore the TOS field, but controlling the quality of service is becoming very important

The Internet/Network Layer

IP datagram header

Application	TOS	value (hex)
Telnet	Minimise delay	1000
FTP control	Minimise delay	1000
FTP data	Maximise throughput	0100
SMTP control	Minimise delay	1000
SMTP data	Maximise throughput	0100
ICMP	None	0000
NNTP (news)	Minimise cost	0001

Suggested values for the TOS for some applications

The Internet/Network Layer

IP datagram header

- Type of service. The TOS was only ever weakly supported in real implementations, so this has been updated to be the *Differentiated Services Field*, with expanded quality of service responsibilities. See RFC2474

The Internet/Network Layer

IP datagram header

- Total Length. Of the entire datagram, including header, in bytes. 16 bits, so giving a maximum size of 65535 bytes. Much larger than domestic networks need, but too small for future high-speed networks.

The Internet/Network Layer

IP datagram header

- Total Length. As usual, larger sizes mean lower overheads:
  - Time overhead in splitting data into packets, adding headers, then removing headers and reassembling
  - Bandwidth overhead as each header is 20 or more bytes that is not data

The Internet/Network Layer

IP datagram header

- Identification. 16 bits. A value that is unique to each datagram, often incrementing by 1 for each successive datagram sent

Used in fragmentation to reassemble the fragments of a single datagram. All the fragments get their own IP header, but share the same identification

The Internet/Network Layer

IP datagram header

- Flags. Three bits: two used and one reserved
  1. RF. Reserved for later use

The Internet/Network Layer

IP datagram header

- Flags. Three bits: two used and one reserved

  2. DF. *Don't fragment*. If the destination can't (or doesn't want to) reassemble fragments this bit is set to inform the routers on the path to the destination. A router might choose an alternative non-fragmenting route, or simply drop the packet and send an error message back to the source which can then send smaller packets

  All hosts are required to be able to accept datagrams of 576 bytes

The Internet/Network Layer

IP datagram header

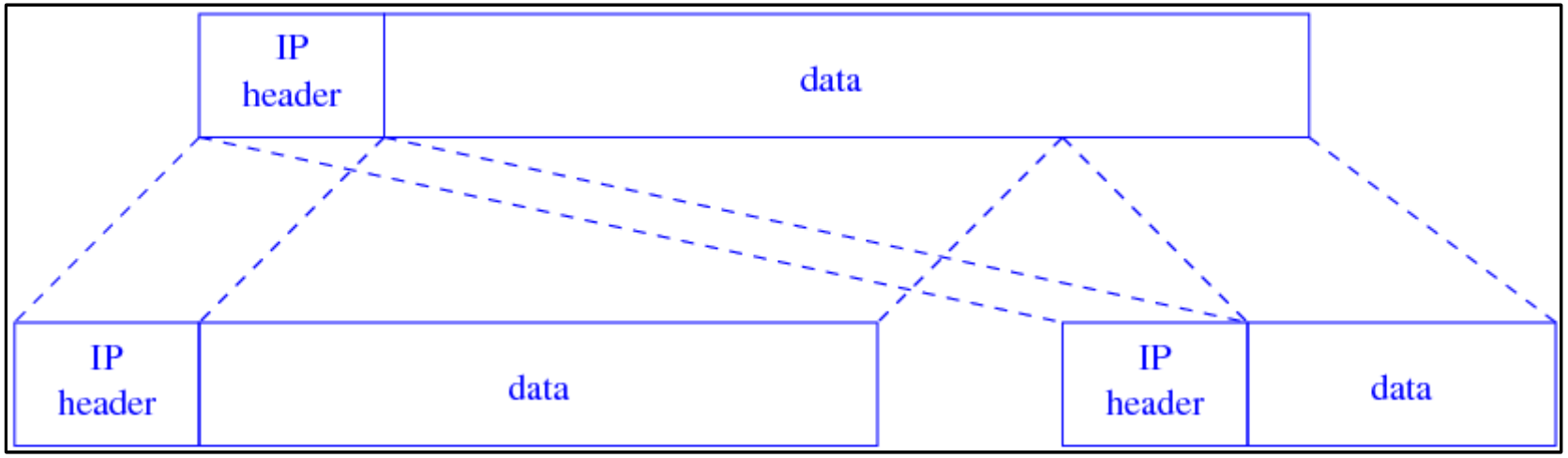
- Flags. Three bits: two used and one reserved

  3. MF. *More fragments*. All fragments except the last have this set

The Internet/Network Layer

IP datagram header

- Fragment Offset. Where this fragment came from in the original datagram



The Internet/Network Layer

IP datagram header

- Fragment Offset. 13 bits, giving the offset divided by 8. E.g., value of 20 means an offset of 160
  - So 13 bits is enough to cover the 16 range of sizes
  - And every fragment (apart from the last) must be a multiple of 8 bytes long

The Internet/Network Layer

IP datagram header

- Fragment Offset. Every fragment has a copy of the original IP header, but with the various fragmentation and length fields set appropriately
- When the fragment with MF not set is received, its fragment offset and length will give the length of the original

The Internet/Network Layer

IP datagram header

- Fragmentation is costly and should be avoided
  - Performing fragmentation in a router takes time
  - More overhead as more fragments for a given amount of data
  - More overhead as more datagrams are traversing the network
  - More datagrams means a greater probability one will be lost of corrupted

The Internet/Network Layer

IP datagram header

- Fragmentation is costly and should be avoided
  - If a fragment is lost, the *entire* original datagram must be retransmitted
- Setting DF prohibits fragmentation; if a router cannot avoid fragmenting it drops the packet and returns a “fragmentation needed but DF set” error message back. The sender can then send smaller packets

The Internet/Network Layer

IP datagram header

- DF allows *MTU Discovery*. The MTU is the largest packet a host of network can transmit. The *path MTU* is the smallest MTU for the entire path from source to destination
- A packet not larger than the path MTU will not get fragmented
- Send variously sized packets with DF set, and monitor the errors returned

The Internet/Network Layer

IP datagram header

- When a packet reaches the destination with no fragmentation error we have found a lower bound for the path MTU
- This is approximate as the network is dynamic and paths may change

The Internet/Network Layer

IP datagram header

- Better is the approach of IPv6: a datagram is never fragmented en route, but simply drops a large packet and returns an error message
- This is simpler, and so faster

The Internet/Network Layer

IP datagram header

- Time To Live. An eight bit counter used to limit the lifetime of a packet
- Poorly configured routers might bounce packets back and forth or in circles indefinitely, thus clogging the network with lost packets
- The TTL starts at 64, or 32, say, and is reduced by one as it passes through each router

The Internet/Network Layer

IP datagram header

- Time To Live. If a TTL ever reaches 0, that packet is discarded, and an error message is sent back to the source
- This limits errant packets: eventually the TTL reaches 0 and the packet is dropped

The Internet/Network Layer

IP datagram header

- Time To Live. Eight bits means a maximum path of length 255, but this seems enough for the current Internet: no valid paths as long as this are known

The *width* of the Internet is the length of the longest path: this is uncertain but definitely over 32

The Internet/Network Layer

IP datagram header

- Time To Live. Originally the TTL was to be a measure of *time*, reducing by one for each second in a router. In practice no implementations did this, but just decremented by one regardless. This is now the expected behaviour.

The Internet/Network Layer

IP datagram header

- Protocol. This eight bit field connects the IP layer to the transport layer. This is a value indicating which transport layer to pass the packet to. For example, UDP is 17 and TCP is 6

The Internet/Network Layer

IP datagram header

- Header checksum. As for the Ethernet header, this is a simple function of the bytes in the IP header. If the checksum is bad, the packet is silently dropped. A higher layer must detect this and perform whatever action it needs. (Recall that the IP layer is not guaranteed reliable)

The checksum includes the TTL, so it must be recomputed for the packet by each router

The Internet/Network Layer

IP datagram header

- Source and Destination Address. 32 bit numbers that uniquely determine the source and destination machines on the Internet
- So there is at most 4,294,967,296 hosts on the Internet
  - Not enough, particularly as many addresses are reserved for special purposes
  - But "uniquely determine" is not really true: see later

The Internet/Network Layer

IP datagram header

- Optional Fields. A variable length list of (usually absent) optional bits and pieces to allow for extensions to the IP

Also allows for rarely used stuff, so that the header is not cluttered with mostly unused fields (overhead, again)

The Internet/Network Layer

IP datagram header

- Optional Fields. Including:
  - Security and authentication
  - Record Route. Each router records its address in the header as the packet passes by
  - Timestamp. Each router records its address and the current time in the header as the packet passes by

The Internet/Network Layer

IP datagram header

- Optional Fields. Including:
  - Strict Source Routing. A list of addresses that give the entire path from source to destination
  - Loose Source Routing. A list of addresses that must be included in the path from source to destination

The Internet/Network Layer

IP datagram header

- Optional Fields. Most options are for debugging or profiling behaviour. Mobile IP uses Source Routing (see later)

The Internet/Network Layer

IP datagram header

- Optional Fields. Record route is limited: as options are restricted to 40 bytes, this means at most nine routers can record their addresses (4 bytes per address, plus a couple of bytes for the option header). Routes can easily be over 30 hops long so other techniques are used to map paths (see traceroute later)

The Internet/Network Layer

IP Addresses and Routing Tables

- Roughly speaking, every machine on the Internet has a unique address
- These are not random, but allocated in such a way to make routing between hosts much easier
- If there were no structure on the addresses every router everywhere would have to know where every other router in the world was

The Internet/Network Layer

IP Addresses and Routing Tables

- The Internet is a collection of networks
- The addresses is split into two parts:
  - A network number
  - A host number on that network
- The host number defines the host uniquely on a network
- The network number defines a network uniquely on the Internet

The Internet/Network Layer

IP Addresses and Routing Tables

- To an end host routing is trivial
  - If the destination is on the same network, simply put the packet out on the network
  - If not, send the packet to a gateway, and let it deal with the problem

The Internet/Network Layer

IP Addresses and Routing Tables

- To a gateway or router the problem is to send the packet on towards the destination network
- Which one? This is the difficult bit
- But there are very many fewer networks than hosts, so this is already a great simplification

The Internet/Network Layer

IP Addresses and Routing Tables

- A router contains a table of IP addresses, with gateways associated with those addresses

The Internet/Network Layer

IP Addresses and Routing Tables

- Each row in the table contains
  - A destination address. This can be the address of a single host, or a network address
  - The address of the *next hop* router, i.e., the address of where to send the packet next. This is the address of a router that is directly connected to the current one
  - Which *interface* to send the packet out on to get to that router. A router has many interfaces and this describes which one to use

The Internet/Network Layer

IP Addresses and Routing Tables

- When a packet arrives at a router it checks the table
  - If the packet destination matches a host address, send the packet to the indicated gateway on the indicated interface
  - Else if the packet destination matches a network address, send the packet to the indicated gateway on the indicated interface
  - Else find an entry in the table marked *default*, and send the packet to the indicated gateway on the indicated interface

The Internet/Network Layer

IP Addresses and Routing Tables

- If there is no default route, drop the packet and return a error message “network unreachable” to the source

- For now, regard routers as machines with tables that tell them where to send packets. We will see how the tables are created later