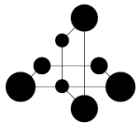


INFORMATION CODING
Linköping University

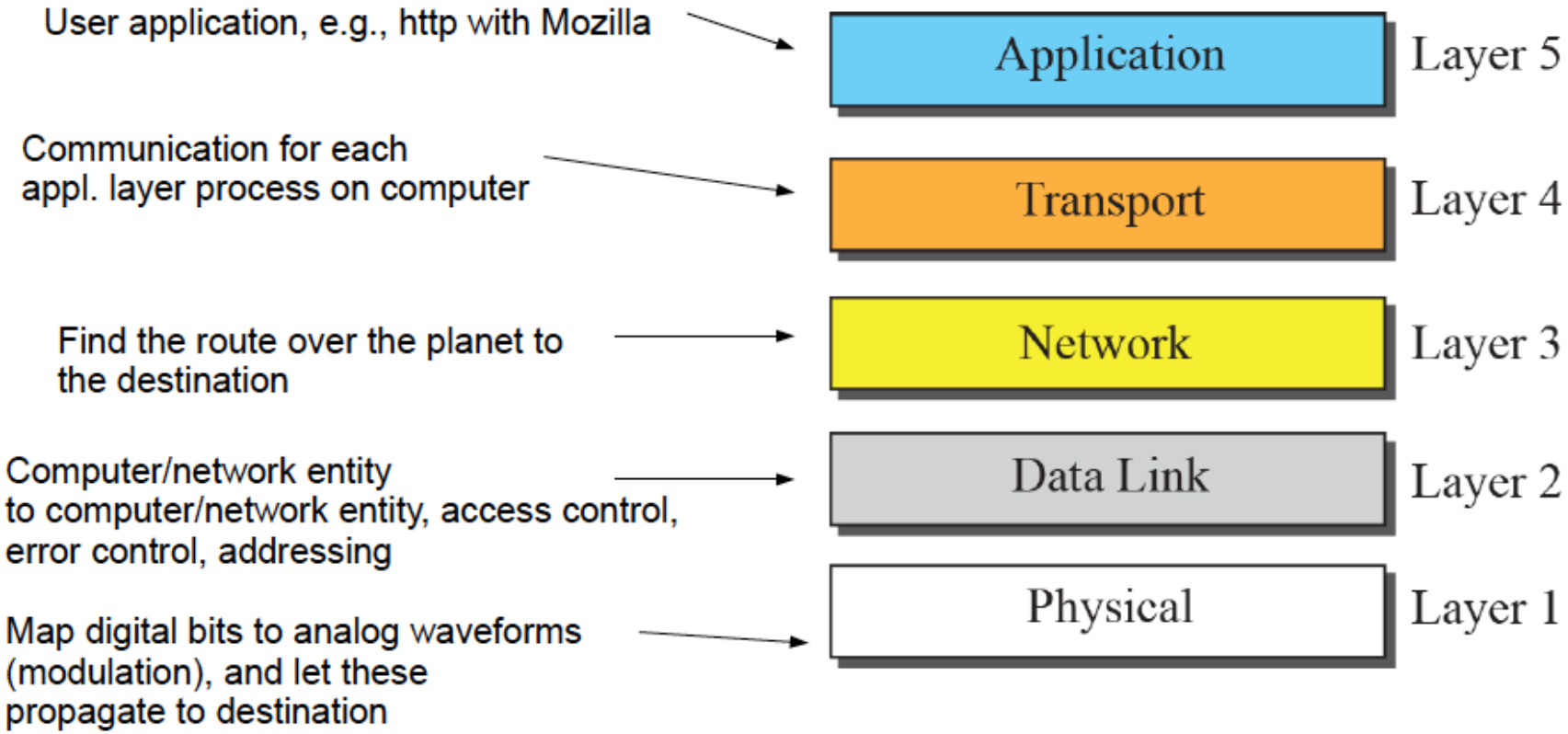
TSIN02 Internetworking

Lecture 2 – Network of networks

Outline

- The physical layer
- The link layer
- Small networks - Ethernet
- Big networks – Internet
- The network layer
- Routing
- Network management

TCP/IP protocol suite



Layer 1: Physical layer

How to put digital bits onto an analog signal waveform (*modulation*).

Example:

- On-Off switching (*wire, fiber*)
- Amplitude/frequency/phase modulation of a sine wave (*radio, wire*)

Modulation

The process of sending a message signal (e.g. a sequence of digital bits) with an analog signal that can be physically transmitted.

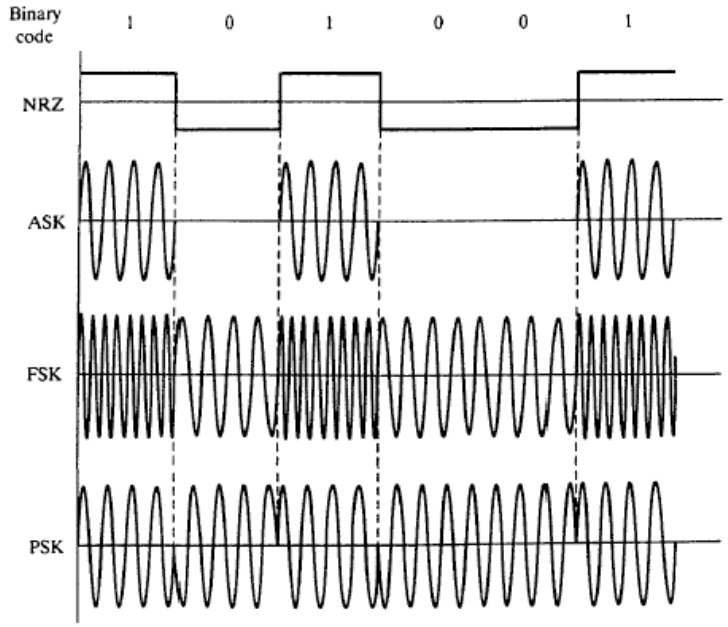
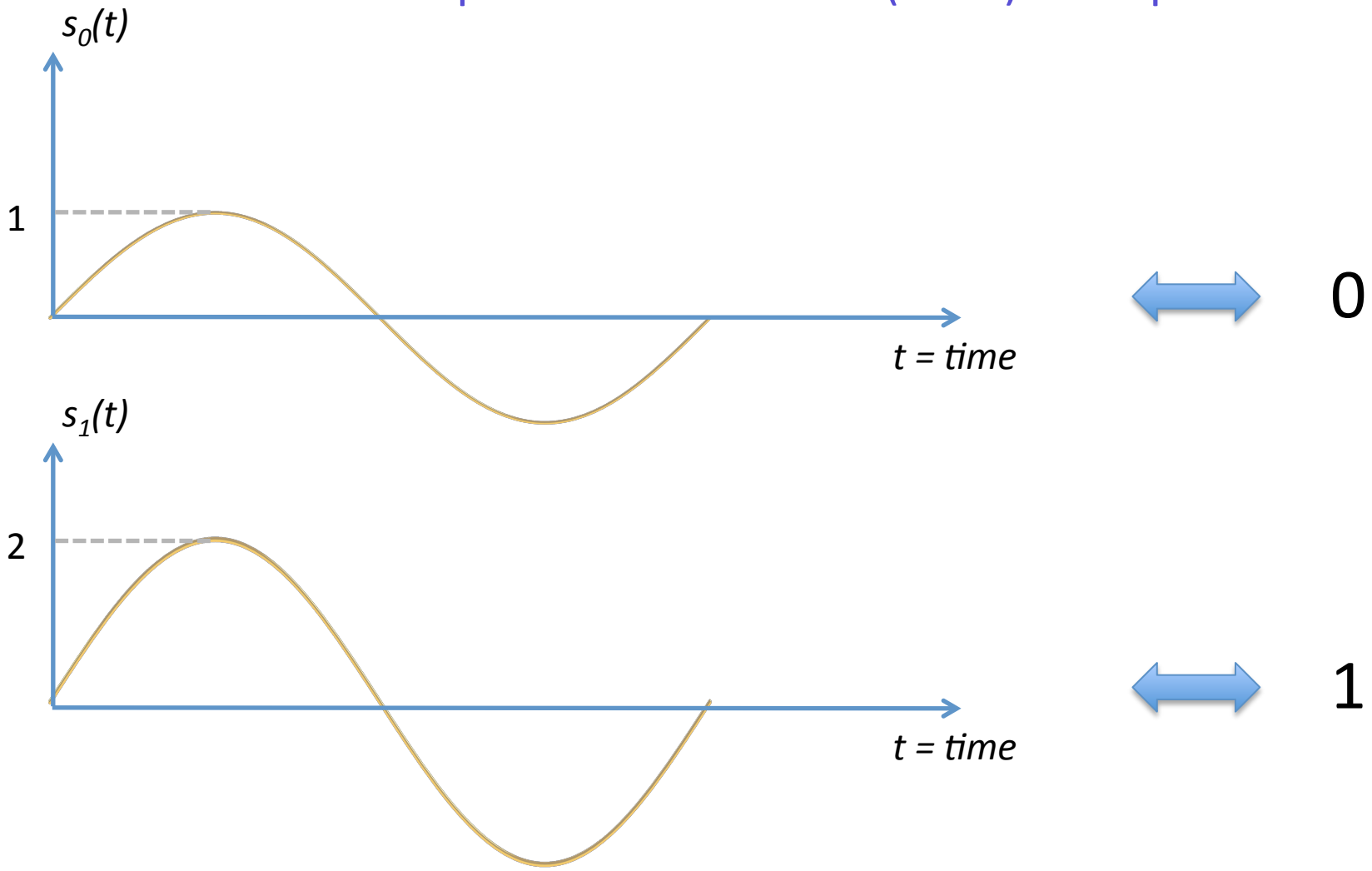


Fig. 5-16 Digital carrier modulation

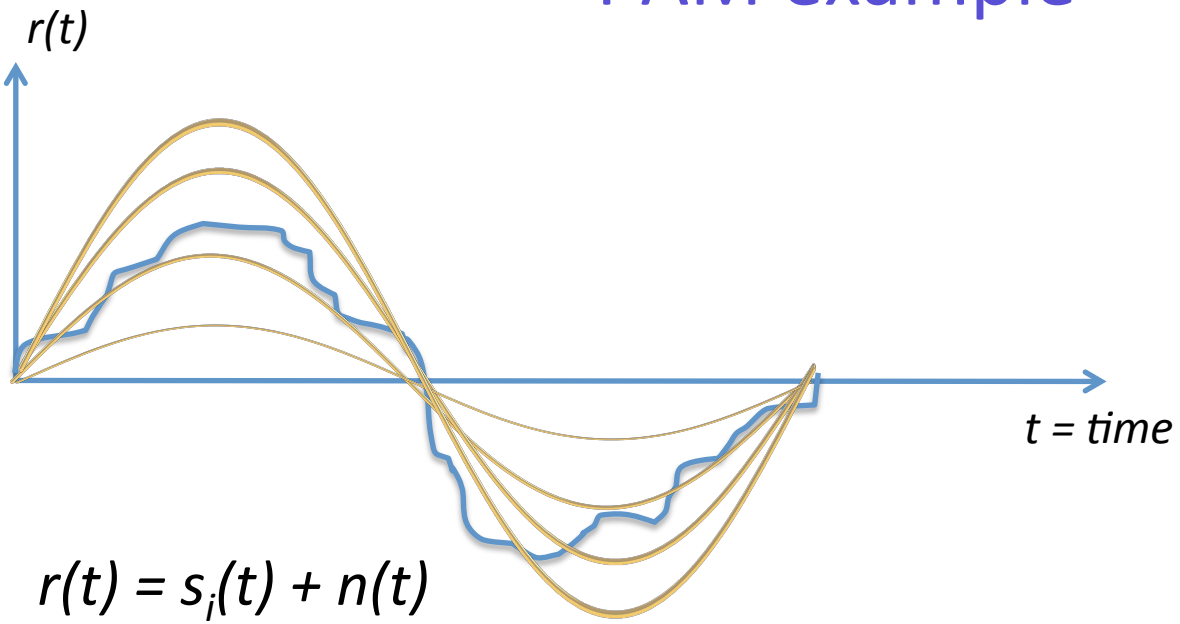
Transmitter

Pulse-amplitude modulation (PAM) example



Receiver

PAM example



0/1 ?

$r(t) = s_i(t) + n(t)$

n(t): noise from channel

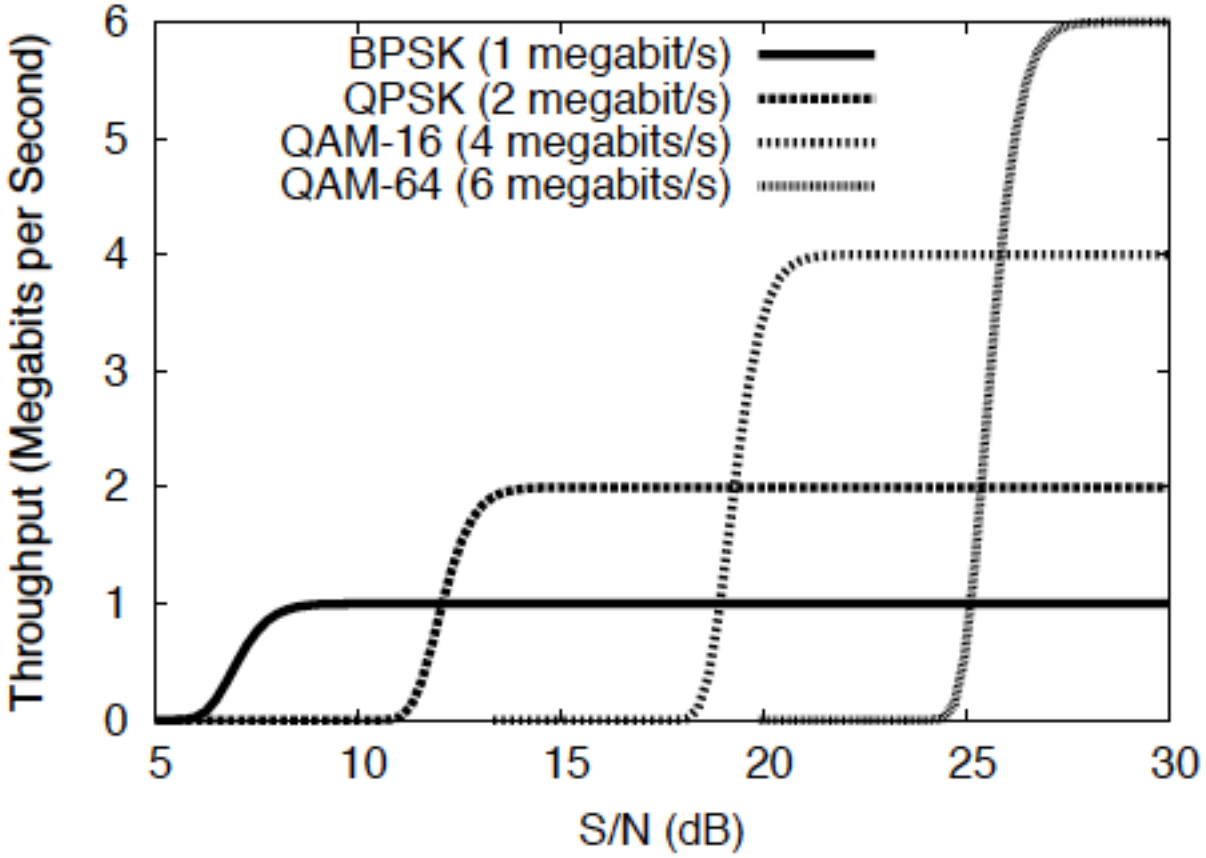
If the noise is strong relative to the signal is is difficult to see if s_0 or s_1 was transmitted. This leads to errors which need to be taken care of by the link layer error control.

Larger signals (more power) reduce errors and also allow for more bits in every pulse.

E.g. double data rate with four signals: $s_0 \Leftrightarrow 00$, $s_1 \Leftrightarrow 01$, $s_2 \Leftrightarrow 10$, $s_3 \Leftrightarrow 11$

Theoretical performance

(assuming 1 μs pulses)



BPSK sends 1 bit/pulse

…

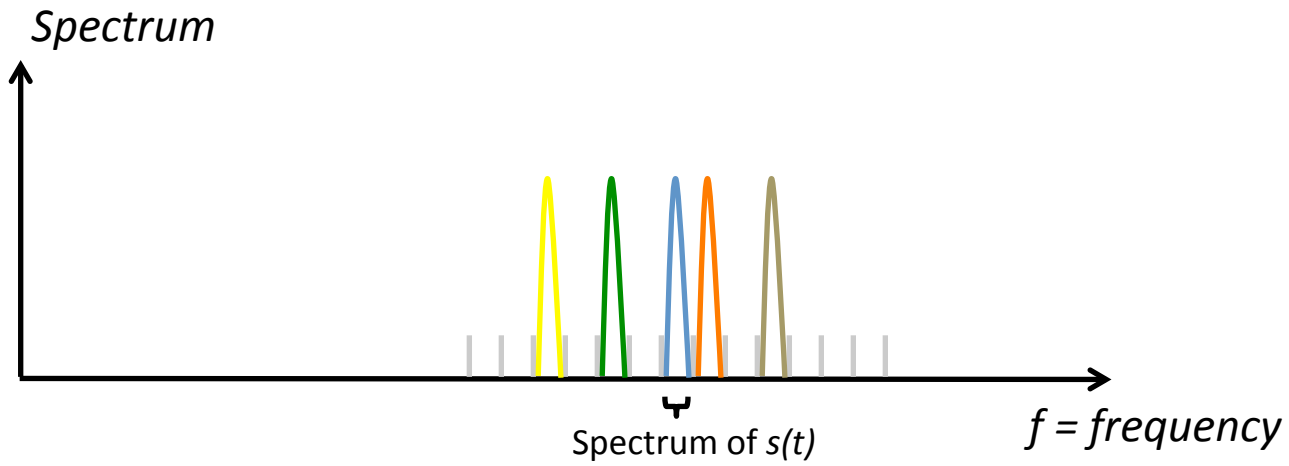
QAM-64 sends 6 bits/pulse

S/N: signal-to-noise ratio

dB: logarithmic scale (decibel)

From John. Bicket, *Bit-rate Selection in Wireless Networks*

The spectrum view

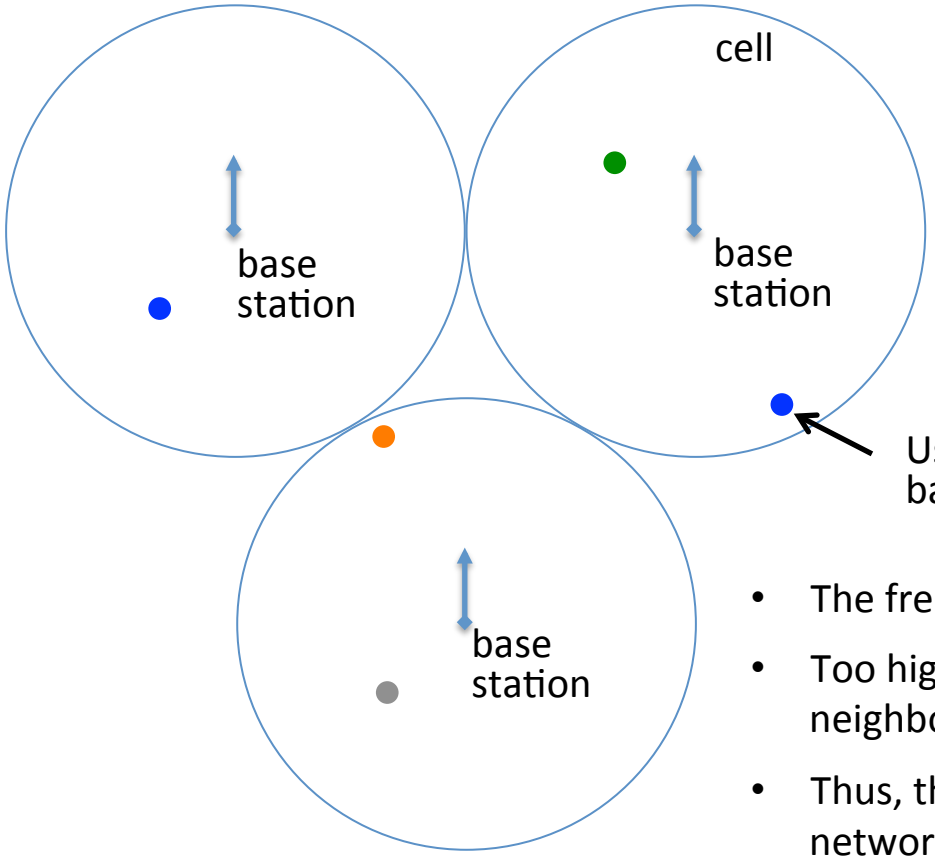


More bandwidth => more sinusoids can be used (of different frequencies)

⇒ More bits can be transmitted

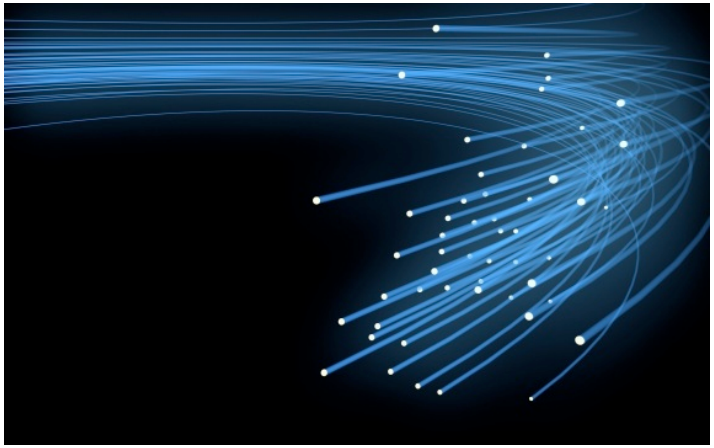
Signals with different frequency bands are easily separated at the receiver. This is used for channel partitioning (FDM).

Wireless access networks



- The frequency bands are re-used in the cells.
- Too high signal power in one cell disturbs neighboring cells on the same band.
- Thus, throughput is limited in wireless access networks.
- More throughput is achieved by using denser cells.
- More on wireless access in a later lecture.

Fiber-optic transmission

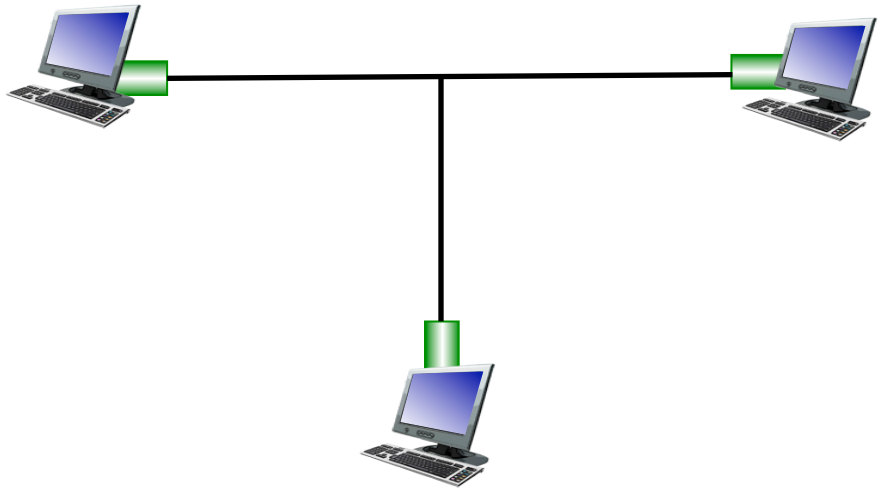


- The signals are confined within the cables and do not disturb each other.
- Throughput is limited by end equipment and ultimately by the optical bandwidth (principally >100 Tbit/s per fiber).
- Throughput is increased by faster end equipment and the use of parallel cables.
- More on fiber-optics in later lectures.

Layer 2: Link layer

Sending data (a frame) between two network interfaces 

- Single-hop addressing
- Handling multiple access
  - FDMA, TDMA, CSMA
- Handling errors
  - FEC, ARQ



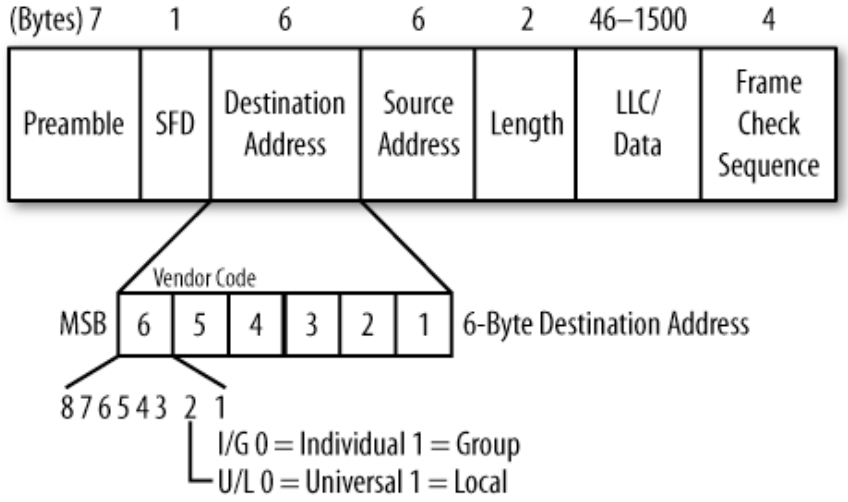
Single hop addressing

Physical layer addresses: Each frame has a source and a destination physical layer address

Common example: The standard (IEEE 802) format uses Media access control (MAC) addresses: 48-bit (6 bytes = 12 hexa-decimal digits). Example: 07-01-02-01-2C-4B.

MAC addresses are unique for every physical network interface produced!

The link layer frame



Handling multiple access

Or,

How to split the channel between users

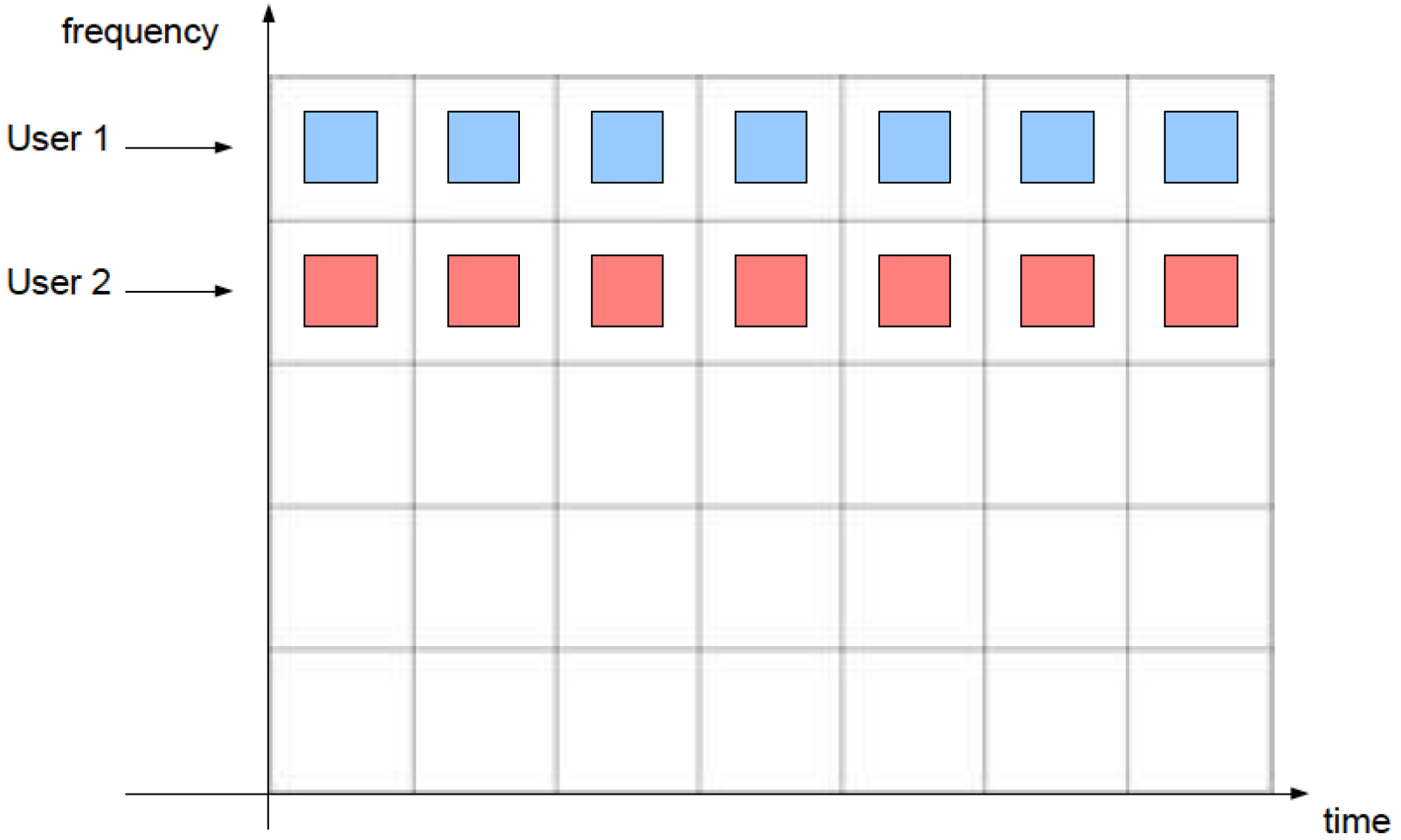
Access control: two classes

- **Channel partitioning** (centrally controlled): Time division modulation (TDM), Frequency division modulation (FDM).

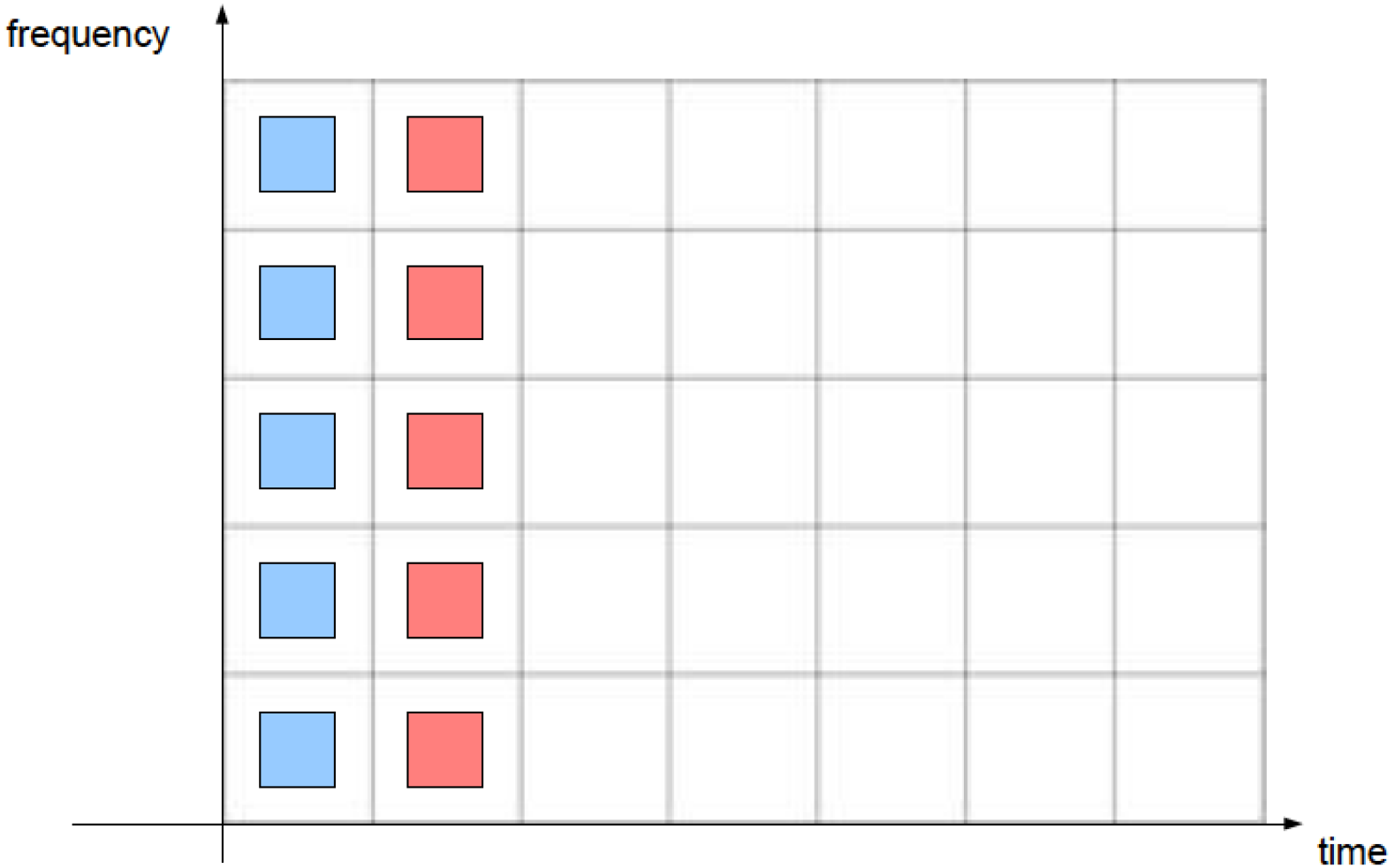
  When applied to a distributed (e.g. radio) channel with multiple users, the same methods are referred to as TDMA (Time Division Multiple Access) and FDMA (Frequency Division Multiple Access).

- **Random access** (locally controlled): CSMA-CD, CSMA-CA, each user transmits using the whole channel when it thinks that the channel is free. This may lead to collisions.

Frequency division multiplexing (FDM)



Time division multiplexing (TDM)



Random access methods

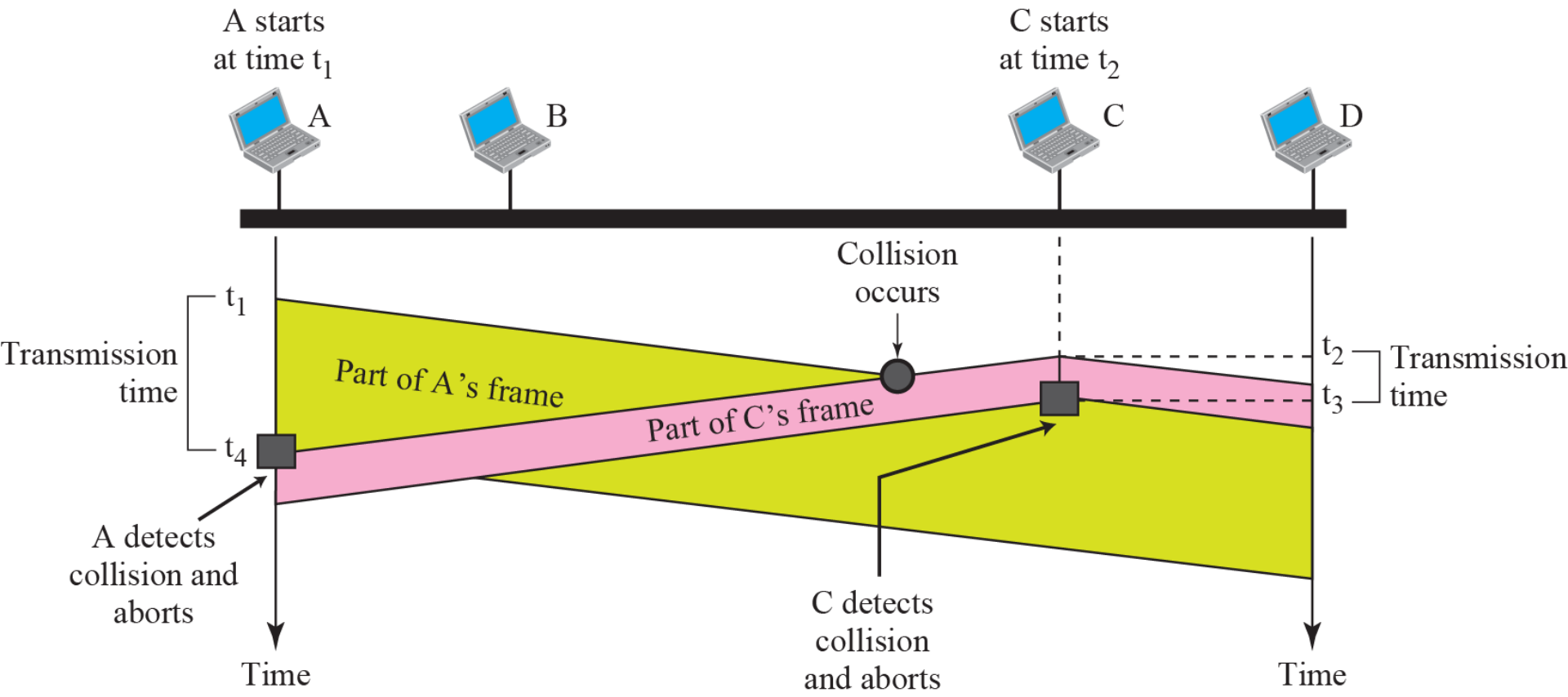
Carrier Sense Multiple Access (CSMA) with collision detection: CSMA-CD

Send, and simultaneously, listen for collision. If no collision, ok!

Otherwise, wait a random time and try again a few times

(works fine with cable transmission, not with radio)

CSMA-CD


A starts at time t_1
C starts at time t_2
A
B
C
D
Collision occurs
t_1
Transmission time
t_4
Part of A's frame
Part of C's frame
t_2
t_3
Transmission time
A detects collision and aborts
C detects collision and aborts
Time
Time

Wireless CSMA

- For CSMA-CD, the hosts need to send and receive at the same time. This is judged too costly for wireless systems.

- We need a different solution!

CSMA with collision avoidance (CSMA-CA) method 1 !

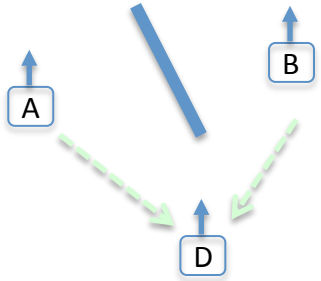
- Listen if someone else is sending. Wait a random time interval if someone else is sending. Go on like this and send when the channel is considered free.

- Wait for acknowledgement packet (ACK). This is a packet sent from the receiver as a confirmation.

- If no ACK, wait a random time before trying to resend. Repeat a few times or until ACK is received.

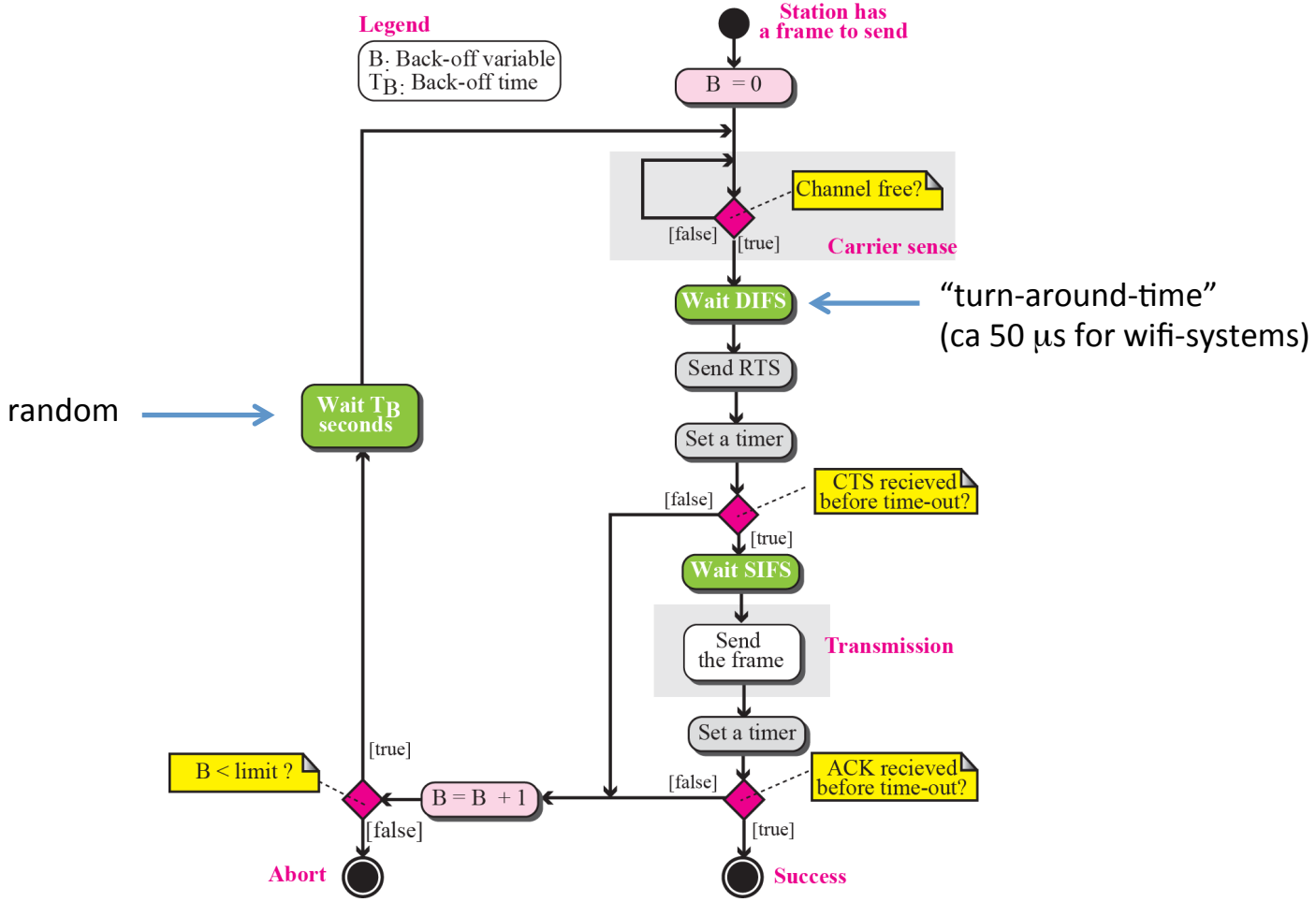
Hidden terminal problem

Sending hosts A and B may reach their destination D but without hearing each other. This can result in both hosts contacting D at the same time, thus creating a collision – the "hidden terminal problem"



- Method 2: Use request-to-send packet (RTS) and wait for clear-to-send (CTS) before starting transmitting the message.

CSMA-CA, method 2, flow diagram



Error control

- **Error control** = Error detection or correction
- Error correction can be either *forward error correction* (FEC) or *automatic repeat request* (ARQ)

Error detection - main principle

- Add extra bits ("parity bits") to the message on the sender side, e.g. repetition code:

  0: 00

- On the receiving side, the receiver message is checked to see if something has gone wrong.

  00: ok!

  10: something wrong!

- In practice, more advanced codes are used such as checksums or *CRC* (Cyclic Redundancy Check) with good error detection properties.

FER - main principle

- Add sufficient parity bits to the message on the sender side, to allow correction, e.g. repetition code: 0 corresponds to codeword “000”, 1 corresponds to codeword “111”.

- On the receiving side, the received message is checked to see if an error occurred (FEC). E.g. 010 -> 000. Only binary messages can be corrected with this particular code. In practice, more advanced codes are used (Hamming, Reed-Solomon, etc) to protect multi-string messages.

- “Rule-of-thumb”: the amount of parity bits must at least be sufficient to address the position of the error(s). E.g. messages of 4 bits require at least 3 additional bits to correct single errors (e.g Hamming(7,4) code).

ARQ

- The receiver sends an acknowledgement (ACK) packet to the sender if the received packet is ok. If the sender has not received an ACK after a while, the sender resends.
- Bit error example: Add a checksum bit, e.g even number of ones: 1001 is transmitted, 1101 is received. In this case, an ACK is not transmitted.
- Loss example: The whole packet was lost. The receiver does not know that the packet was transmitted, does not send ACK, and after a while, the sender resends.
- If the sender does not receive ACK, it can thus be because of bit errors or because the receiver did not receive the packet, or because the ACK was lost
- Sometimes ARQ is referred to as “backward error correction”
- ARQ causes delay!

The local area network
– an example of a link layer network

The local area network

Consider a small network connecting a few hosts (computers, printers et cetera). Such a network is called a local area network (“LAN”, “access network”, “subnet”[1])

shared wire (e.g., cabled Ethernet)

shared RF (e.g., 802.11 WiFi)

[1] ”subnet” if the nodes share a common IP subnet address (to be explained later)

Local area networks

- Small number of devices (typically 10-100), geographically close
- Often sharing a common communication medium, such as a common cable or radio frequency.
- Access method is typically CSMA-CD if cable (Ethernet) or CSMA-CA if radio (WiFi).
- Easy to forward data to any device as they all share the same medium. Device identifier (physical address, MAC address) can be stored in all devices on the network.
- Implements essentially only the Link level (and a trivial part of the Network level, devices may use IP addresses to identify themselves).
- Eventhough the routing functions of the Network level is not available, hosts may still apply the TCP/IP protocols to run “Internet applications” on the local area network.

LAN addresses

each adapter on LAN has unique MAC address

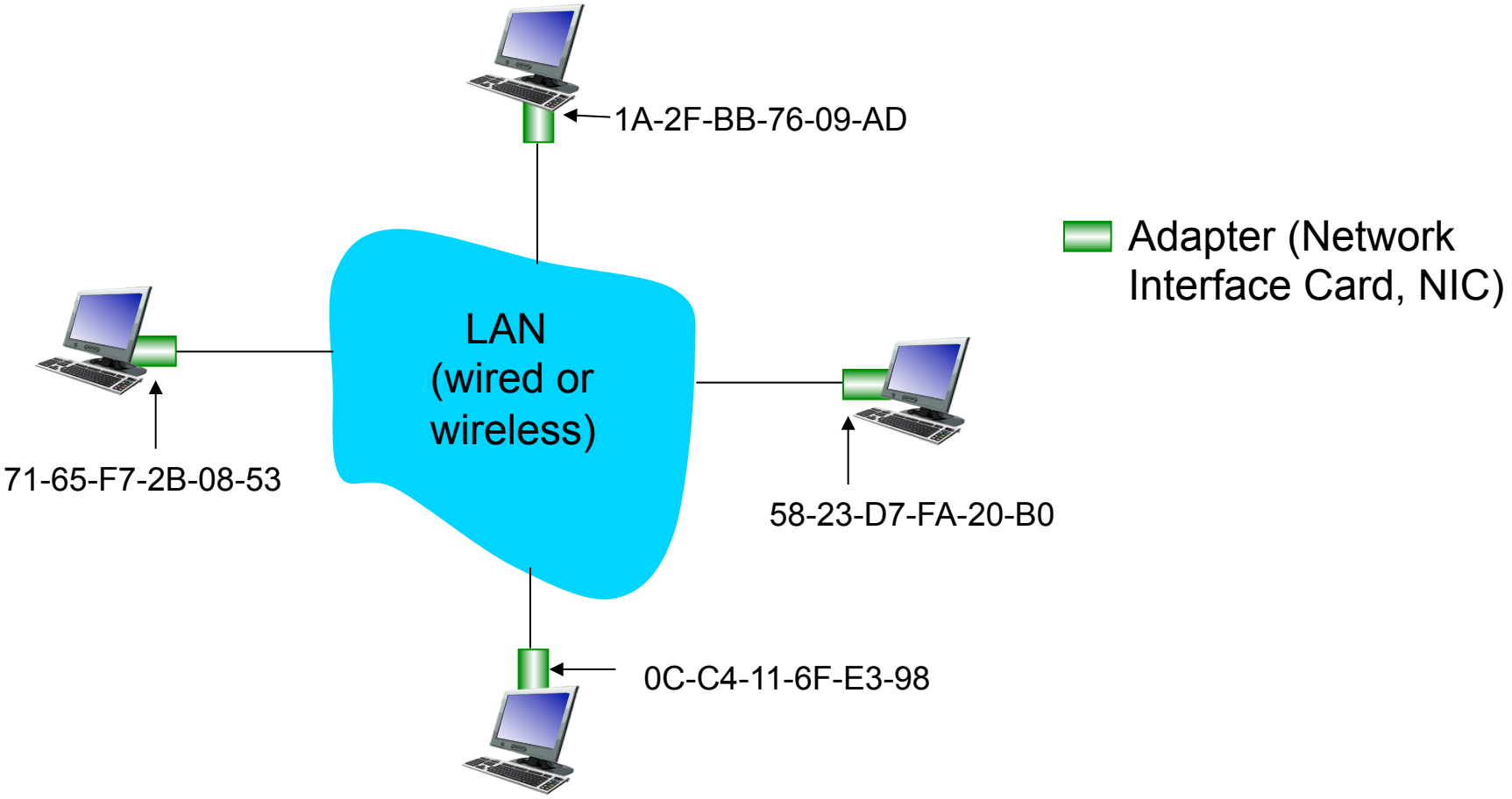


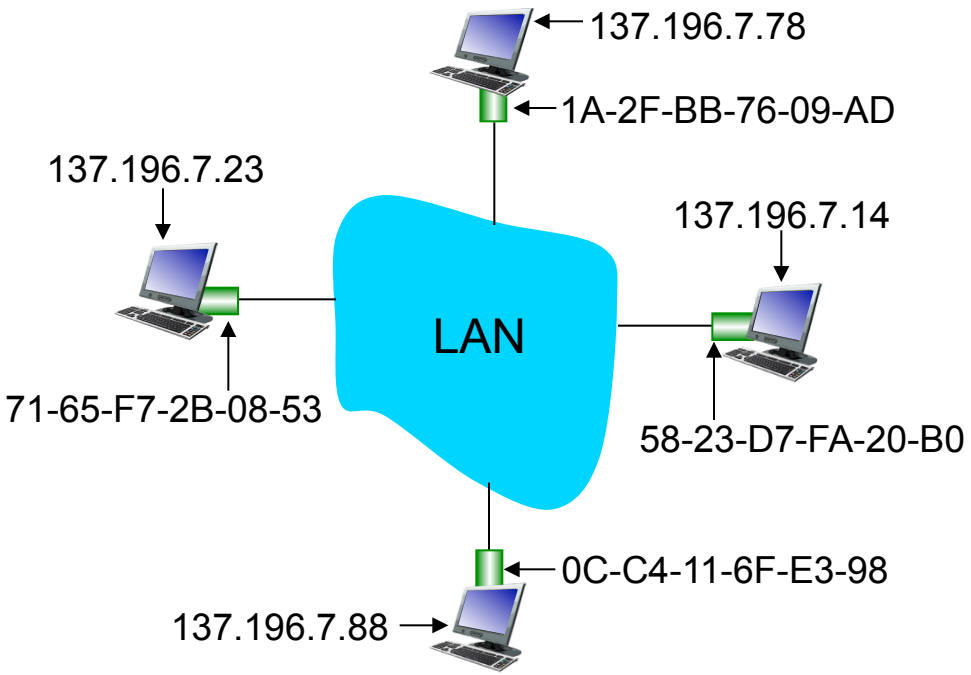
Figure from: J.F Kurose and K.W. Ross,
Computer Networking: A Top-Down Approach

LAN addresses (more)

- MAC address allocation administered by IEEE
- manufacturer buys portion of MAC address space (to assure uniqueness)
- analogy:
  - MAC address: like Social Security Number
  - IP address: like postal address
- MAC flat address ➜ portability
  - can move LAN card from one LAN to another
- IP address *not* portable (as it is hierarchical)
  - address depends on IP subnet to which node is attached
- IP address is part of layer 3 not layer 2 but is commonly used to identify hosts also on a layer 2 LAN.

ARP: address resolution protocol

Question: how to determine interface's MAC address, given its IP address?



ARP table: each node on the LAN keeps a table:

- IP/MAC address mappings for some LAN nodes:

  < IP address; MAC address; TTL >

- TTL (Time To Live): time after which address mapping will be forgotten (typically 20 min)

Slide from: J.F Kurose and K.W. Ross, Computer Networking: A Top-Down Approach

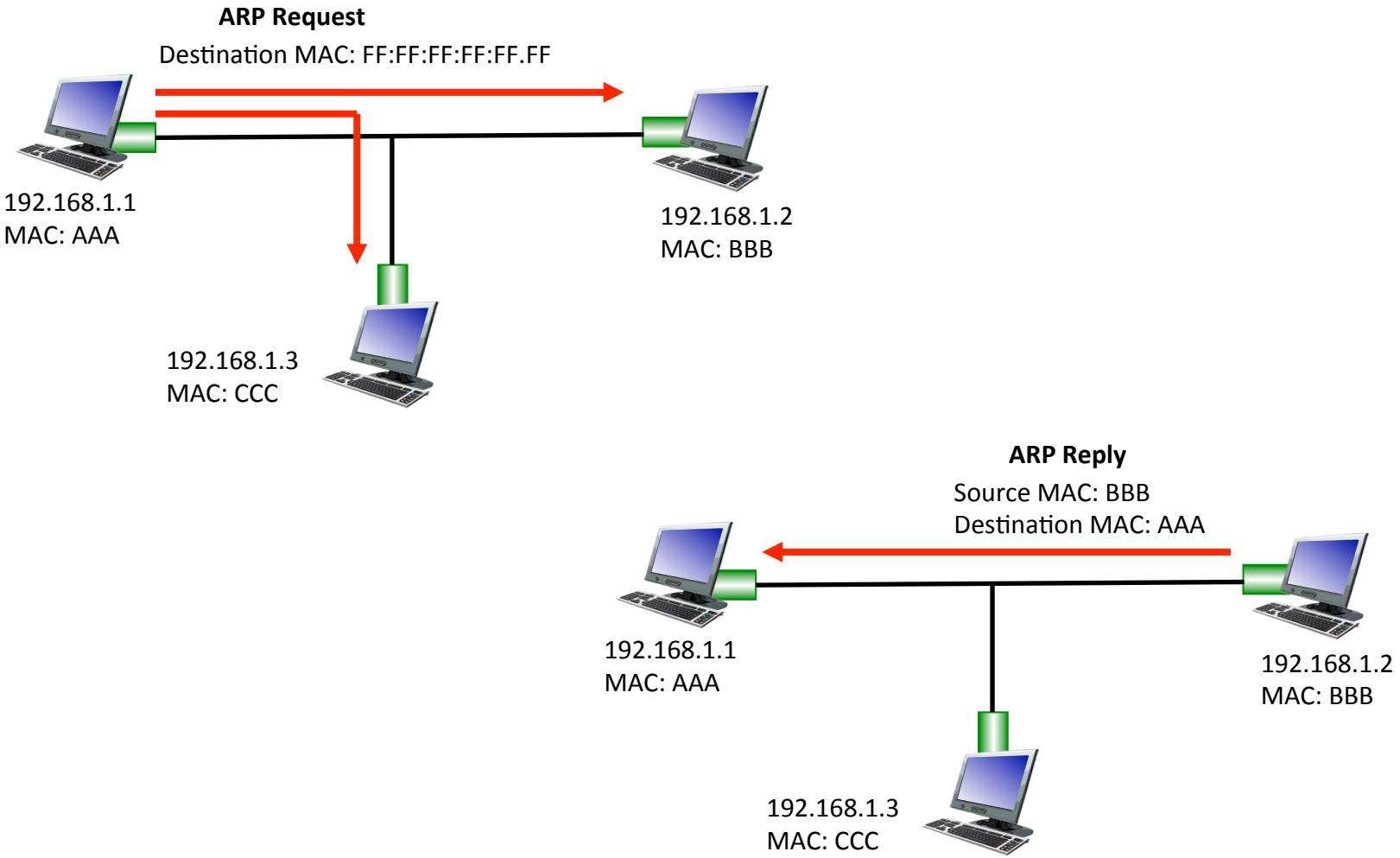
ARP protocol: same LAN

- A wants to send a packet to B
  - B's MAC address is not in A's ARP table.
- A broadcasts ARP query packet, containing B's IP address

  dest MAC address = FF-FF-FF-FF-FF-FF

  all nodes on LAN receive ARP query
- B receives ARP packet, replies to A with its (B's) MAC address
  - frame sent to A's MAC address (unicast)
- A saves IP-to-MAC address pair in its ARP table until information becomes old (times out)
  - soft state: information that times out (goes away) unless refreshed
- ARP is "plug-and-play":
  - nodes create their ARP tables *without intervention from net administrator*

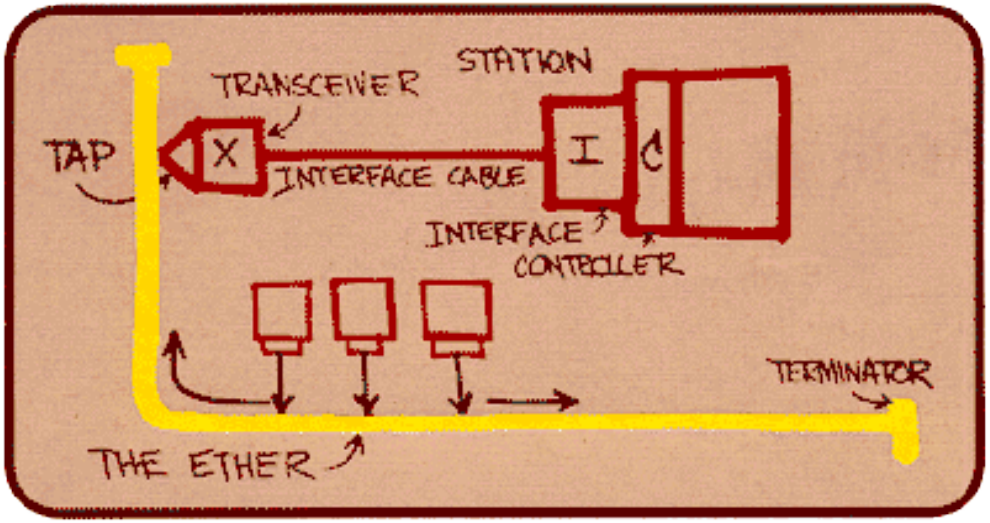
ARP protocol: Illustrated



Ethernet

"dominant" wired LAN technology:

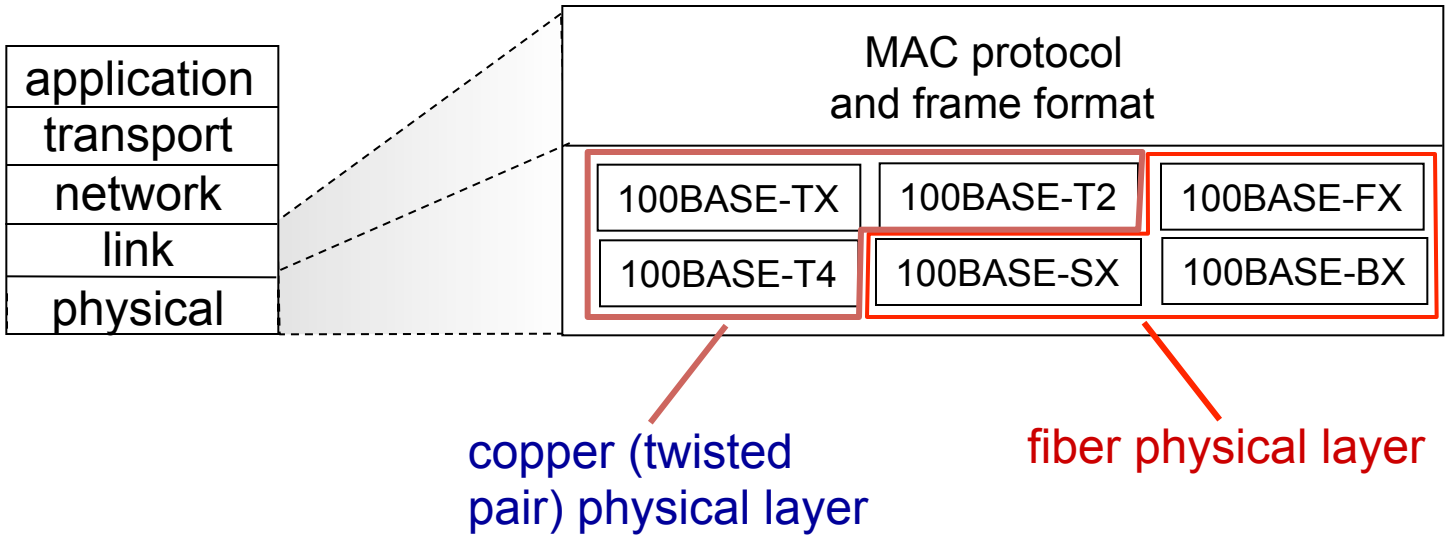
- cheap $20 for NIC
- first widely used LAN technology
- simpler, cheaper than token LANs and ATM
- kept up with speed race: 10 Mbps – 10 Gbps



Metcalfe's Ethernet sketch

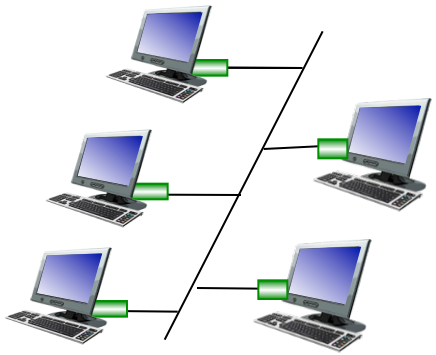
802.3 Ethernet standards: link & physical layers

- *many* different Ethernet standards
  - common MAC protocol and frame format
  - different speeds: 2 Mbps, 10 Mbps, 100 Mbps, 1Gbps, 10G bps
  - different physical layer media: fiber, cable

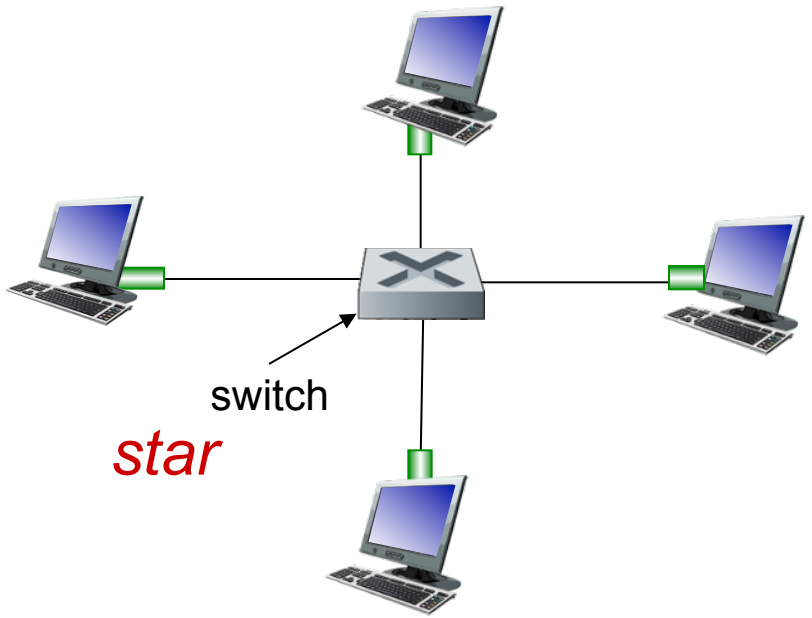


Ethernet: physical topology

- *bus:* popular through mid 90s
  - all nodes in same collision domain (can collide with each other)
- *star:* prevails today
  - active *switch* in center
  - each "spoke" runs a (separate) Ethernet protocol (nodes do not collide with each other)



bus: coaxial cable or hub



star

Ethernet frame structure

sending adapter encapsulates IP datagram (or other network layer protocol packet) in Ethernet frame

preamble:

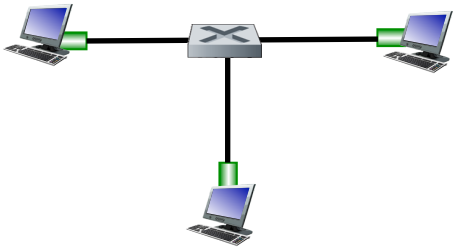
- 7 bytes with pattern 10101010 followed by one byte with pattern 10101011
- used to synchronize receiver, sender clock rates

Ethernet frame structure (cont'd)

- *addresses:* 6 byte source, destination MAC addresses
  - if adapter receives frame with matching destination address, or with broadcast address (e.g. ARP packet), it passes data in frame to network layer protocol
  - otherwise, adapter discards frame
- *type:* indicates higher layer protocol (mostly IP but others possible, e.g., Novell IPX, AppleTalk)
- *CRC:* cyclic redundancy check at receiver
  - error detected: frame is dropped

Ethernet: unreliable, connectionless

- *connectionless:* no handshaking between sending and receiving NICs
- *unreliable:* receiving NIC doesn't send acks or nacks to sending NIC
- Ethernet's MAC protocol: unslotted *CSMA/CD with binary backoff* (average waiting time before retransmission is doubled for every try)
- Collision is avoided by using a *switch*

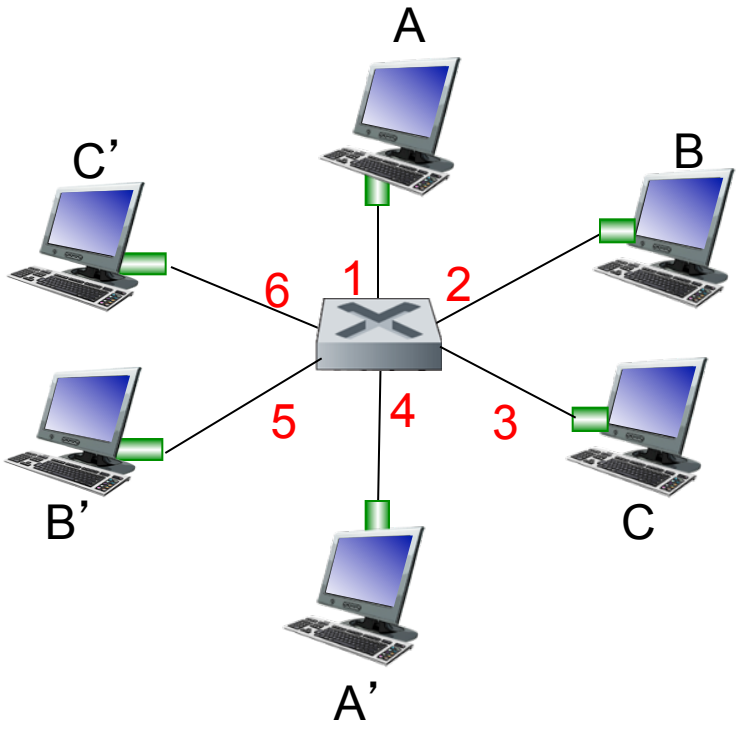


The Ethernet switch

- link-layer device: takes an *active* role
  - store, forward Ethernet frames
  - examine incoming frame's MAC address, selectively forward frame to one-or-more outgoing links when frame is to be forwarded on segment, uses CSMA/CD to access segment
- *transparent*
  - hosts are unaware of presence of switches
- *plug-and-play, self-learning*
  - switches do not need to be configured

Switch: *multiple* simultaneous transmissions

- hosts have dedicated, direct connection to switch
- the switch buffer packets
- Ethernet protocol used on *each* incoming link, but no collisions; full duplex
  - each link is its own collision domain
- *switching:* A-to-A' and B-to-B' can transmit simultaneously, without collisions



switch with six interfaces (1,2,3,4,5,6)

Figure from: J.F Kurose and K.W. Ross, Computer Networking: A Top-Down Approach

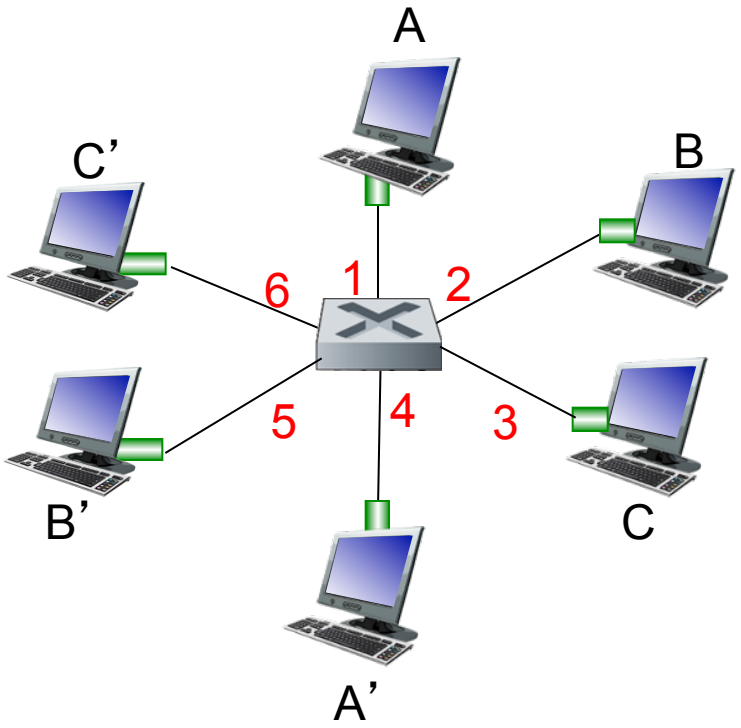
Switch forwarding table

Q: how does switch know A' reachable via interface 4, B' reachable via interface 5?

A: the switch has a switch table, with entries:

- (MAC address of host, interface to reach host, time stamp)

MAC addr	interface	TTL



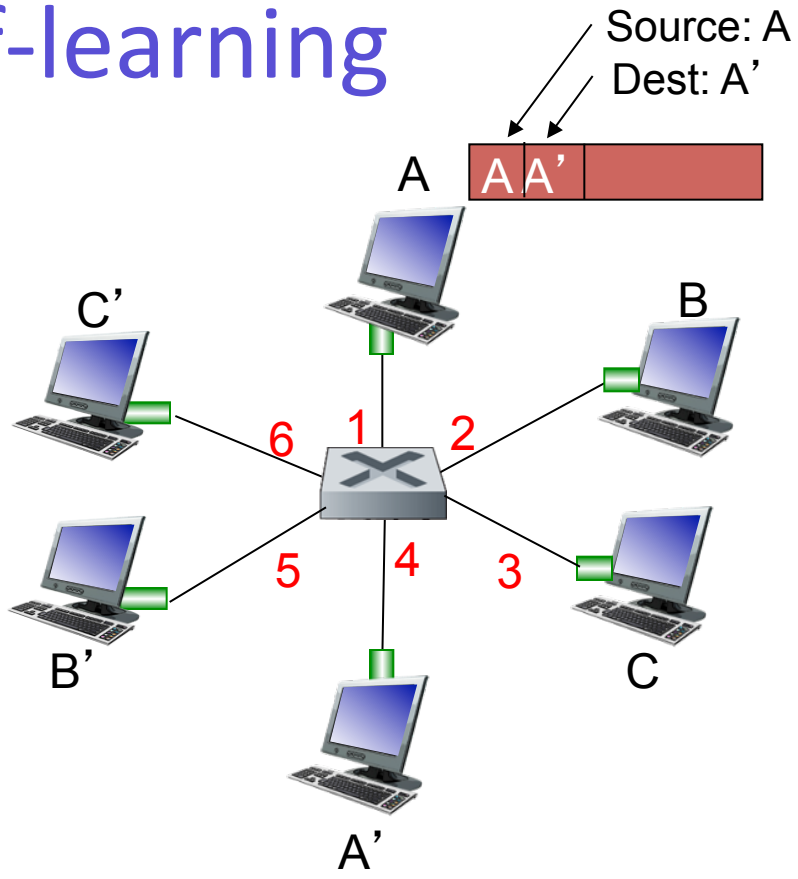
switch with six interfaces (1,2,3,4,5,6)

Q: how are entries created, maintained in switch table?

Figure from: J.F Kurose and K.W. Ross, Computer Networking: A Top-Down Approach

Switch: self-learning

- switch *learns* which hosts can be reached through which interfaces
  - when frame received, switch "learns" location of sender: incoming LAN segment
  - records sender/ location pair in switch table



MAC addr	interface	TTL
A	*1*	*60*

Switch table (initially empty)

Figure from: J.F Kurose and K.W. Ross, Computer Networking: A Top-Down Approach

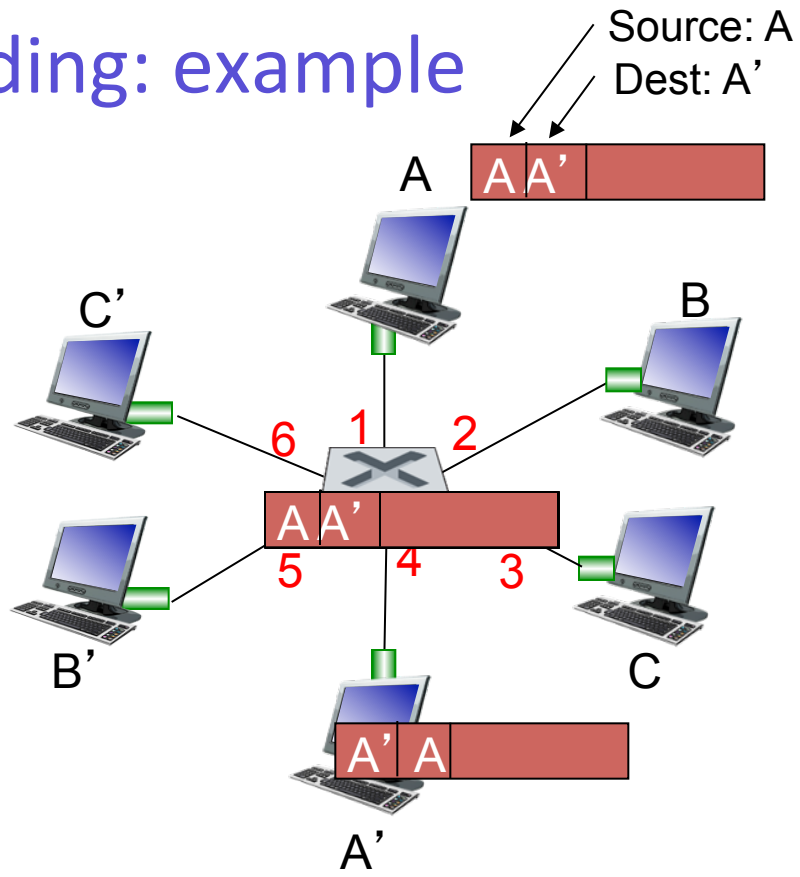
Self-learning, forwarding: example

- frame destination, A', locaton unknown:

  flood

- destination A location known:

  selectively send on just one link

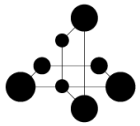


If the switch does not know on which interface a physical destination address is located, it broadcasts, except on the interface where it received the frame

MAC addr	interface	TTL
A	*1*	*60*
A'	*4*	*60*

switch table (initially empty)

Figure from: J.F Kurose and K.W. Ross, Computer Networking: A Top-Down Approach



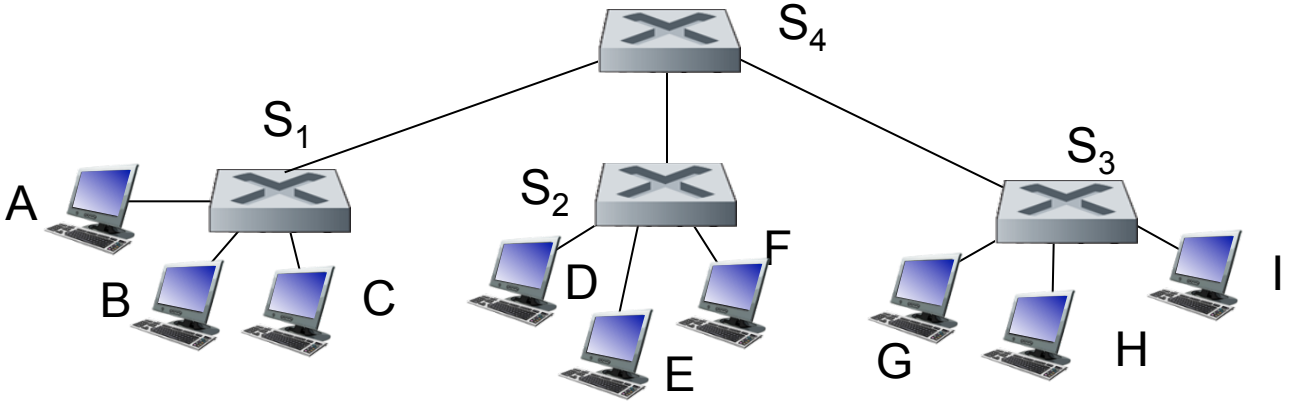
Switch: frame filtering/forwarding

when frame received at switch:

1. record incoming link, MAC address of sending host
2. index switch table using MAC destination address
3. if entry found for destination
   then {
   if destination on segment from which frame arrived
   then drop frame
   else forward frame on interface indicated by entry
   }
   else flood /* forward on all interfaces except arriving
   interface */

Interconnecting switches

switches can be connected together



Q: sending from A to G - how does S_1 know to forward frame destined to F via S_4 and S_3?

A: self learning! (works exactly the same as in single-switch case!)

Big networks - Internet

How can we extend the LAN concept into a world-wide network (“internet-working”)?

All nodes cannot keep track of the physical addresses of all other nodes -> routing problem!

- To be solved by *routers*

Can not guarantee that packets are delivered in correct sequence, in time or at all -> “best effort”!

- To be handled by the *applications*

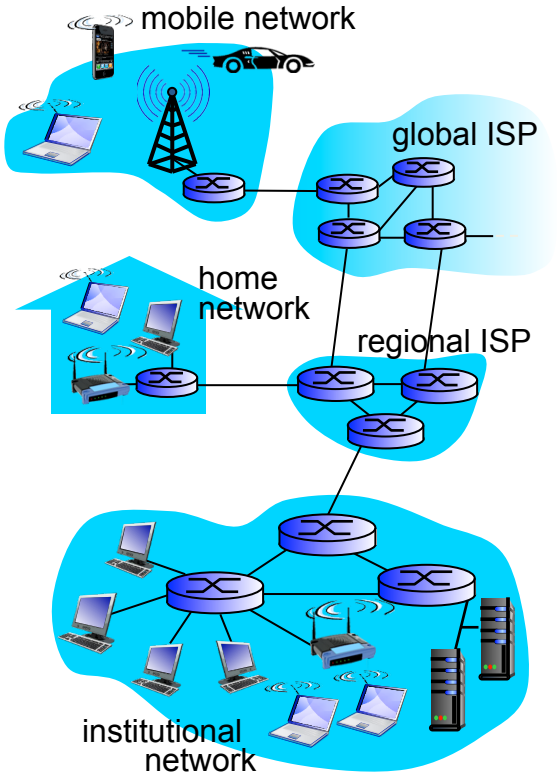
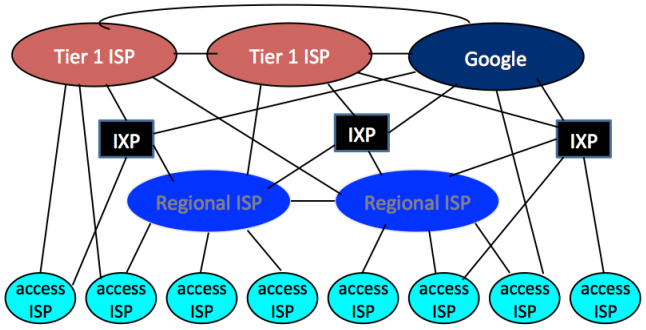


Figure from: J.F Kurose and K.W. Ross, Computer Networking: A Top-Down Approach

Layer 3: Network layer

- How can we extend the LAN concept into a world-wide network?
- Method 1: connect LANs directly with each other (mesh network)
- Method 2: connect LANs to a common "Core structure" (hierarchical network)
- Internet - a combination: Hierarchical but the core has a mesh structure.
- Both solutions require
  - a globally accepted addressing scheme,
  - device to connect networks => "router"
  - global routing of packets,
  - network management (technical and administrative)

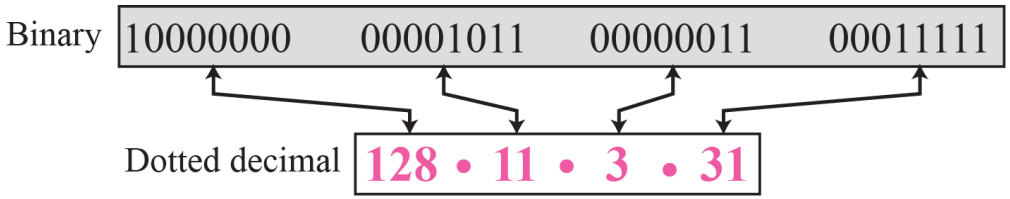


- a globally accepted addressing scheme,
- device to connect networks => “router”
- global routing of packets,
- network management

A globally accepted addressing scheme

- Sufficient address size to accommodate all computers on the network (billions)
- Small enough to be carried in each packet
- Have some structure (geographical, hierarchical or other) to enable fast routing of packets
- Manually or automatically definable (not hard-coded)

IPv4 addressing: Classless interdomain routing (CIDR) addressing



- Each host is assigned a 32 bit IP address
- $2^{32} \approx 4.3$ billion possible addresses/hosts
- Essentially hierarchical (Tier 1 -> Tier 2 -> …-> LAN -> computer)
- Dotted decimal notation, e.g. 128.11.3.31
- Some organizations have been assigned too many addresses. We were running out of IPv4 address space already in the 1990:s!

  (we will return to this issue later)

CIDR subnets

Common for all computers in subnet

Differentiate between computers in subnet

Format: *a.b.c.d/x*

First address in subnet usually

Subnet mask = prefix length in bits, decides how big the subnet is
(suffix length = 32-x bits)

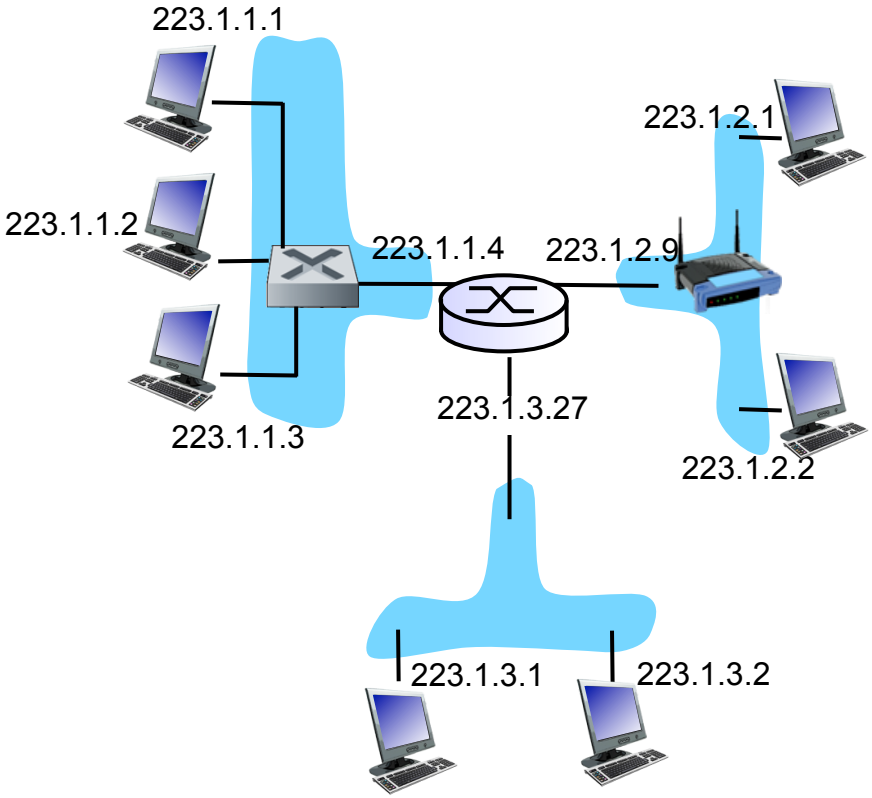
Format example: *1.2.3.0/24*

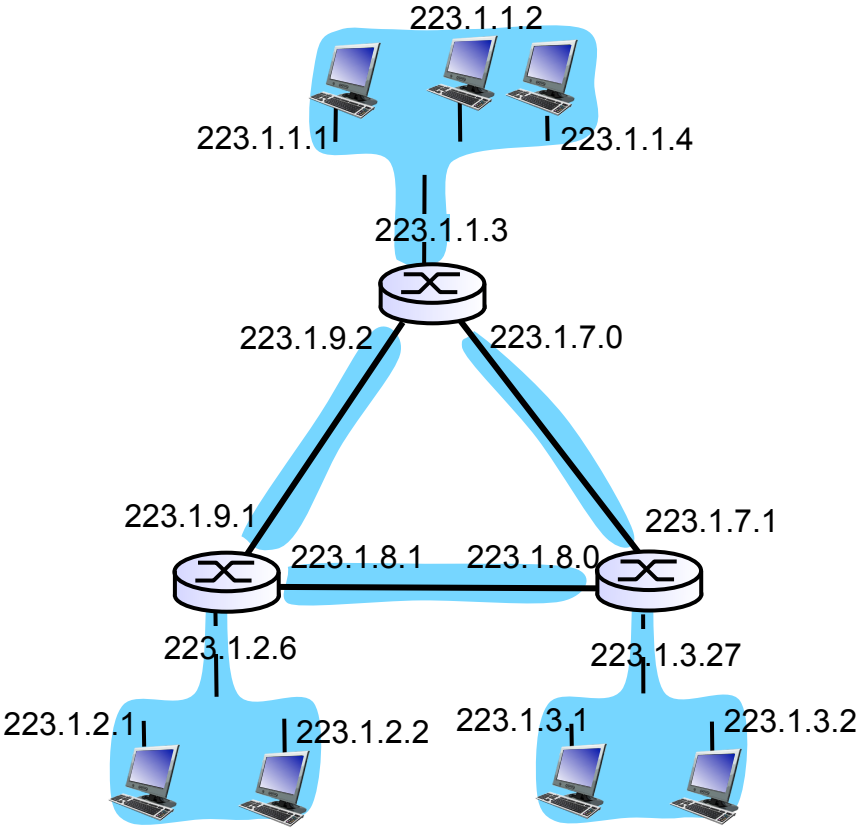
1.2.3 are fix for the subnet. The last entry can vary between 0 and 255.

Suffix "000…0" is used as the (sub)network address while "111…1" is a broadcasting address. Thus, these addresses cannot be assigned to hosts!

The Router

- a device that connects subnets*

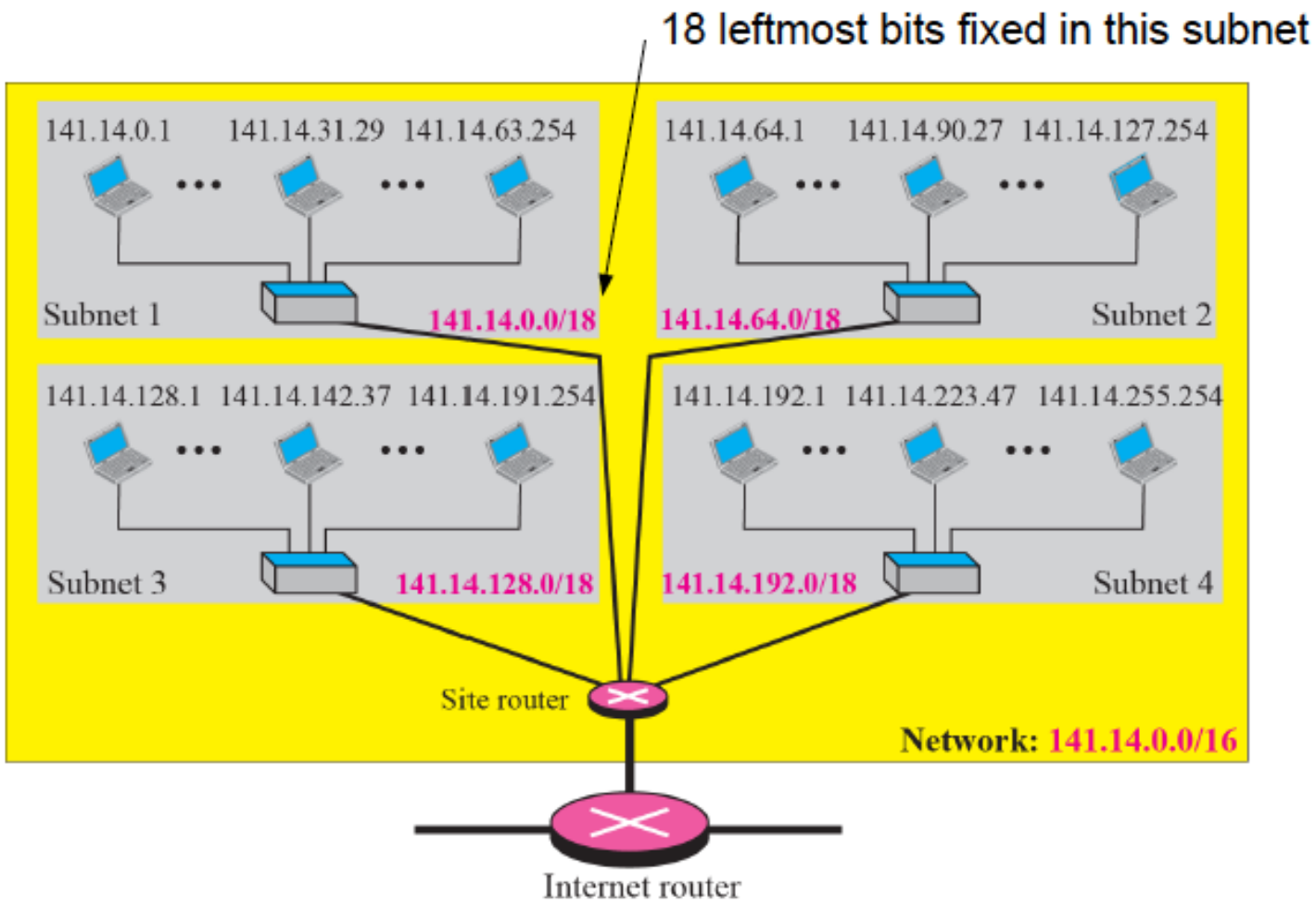




*remember: “subnets” are local networks where the nodes share the same network address.

Q: How many subnets are there in the two networks above (assuming 8 bit suffix length)?

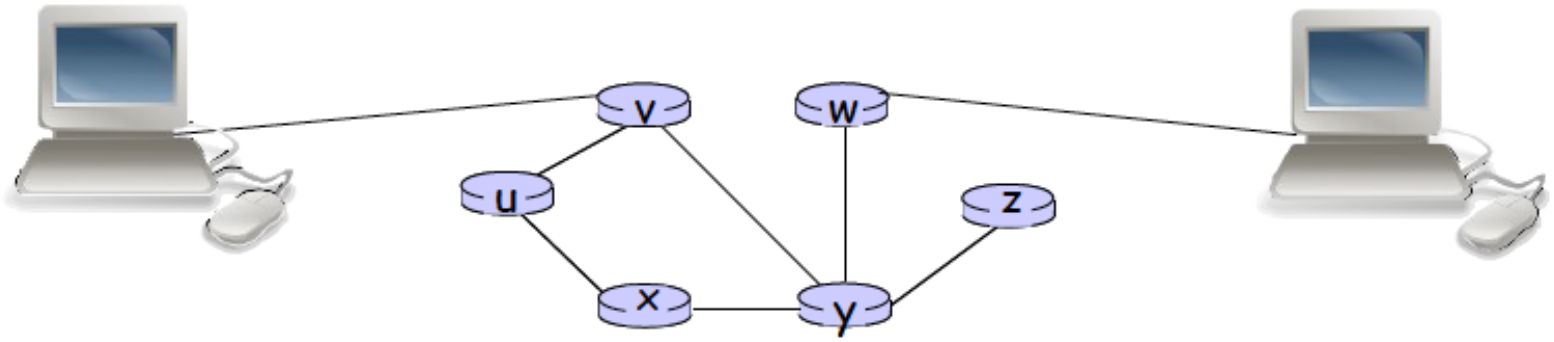
Subnetting



An example of a subnet (141.14.0.0) which is divided into four subnets.

Global routing of packets

Which way should packets travel through the network?



The ARP broadcast messages that are used on the link level are not spread between the different interfaces of routers, which is typically how the router is configured – broadcast floods are avoided.

Router services

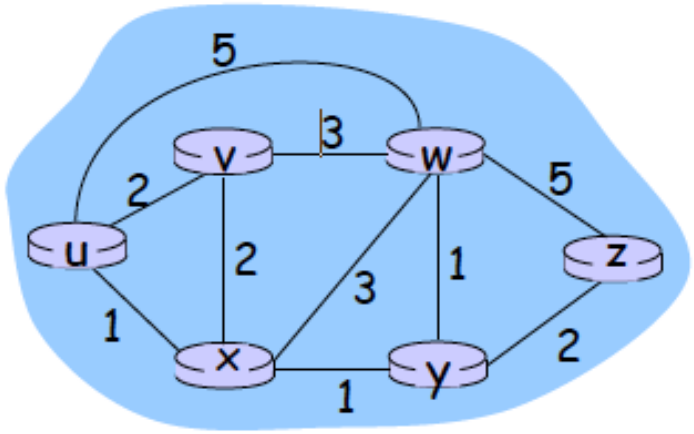
- **Forwarding**: Delivery of datagram from a router input link to the right output link
  - *corresponds to table look-up in a link-level switch*
- **Routing**: Find the best paths to destinations and fill out the forwarding tables
  - *corresponds to learning in a link-level switch*
- **Fragmentation**: Split frames into several IP packets, if needed.
  - reassembly is done at the receiving host

The source/destination physical addresses change with every router link, but the IP source/destination addresses stay the same during transmission over the network!

A switch never modifies the frame addresses.

Routing: how to obtain the forwarding tables

There is not a single best protocols for how to obtain shortest/least cost paths in networks, i.e. to answer what is the best path from x to the other nodes? Link cost may include bandwidth, delay, traffic et cetera.

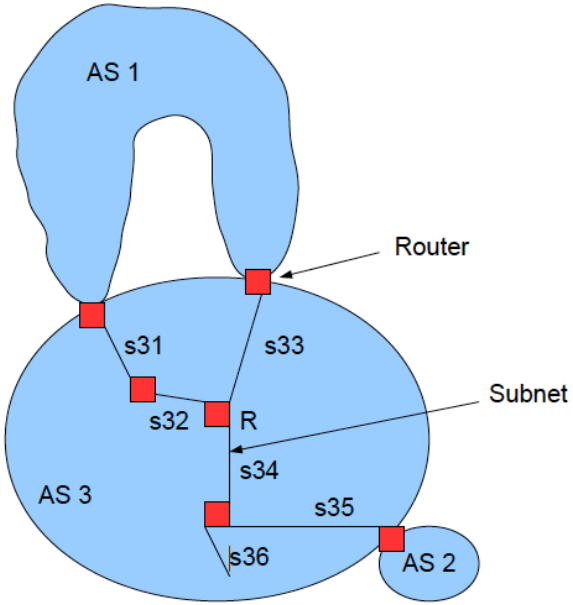


- The “link state routing” (LSR) uses Dijkstra’s algorithm. Each node has to know all link costs in the whole network. Basic implementation complexity is $\mathcal{O}(N^2)$ and N*E messages have to be transmitted where N is the number of nodes and E is the number of links.
- The “distance vector routing” (DVR) uses Bellman-Ford algorithm. Complexity varies. Affected by instability if link costs change.
- In the simplest case, total cost is the number of traversed subnets (links) between routers.

Routing problems and solutions

- Around 1 billion hosts in the Internet. Impossible for the routers to keep track of all!
- Solution:
  - The network is divided into 55000 *Autonomous Systems* (AS).
  - Routers first find the way to the AS (obtain connecting router as well), then to the subnet and then within the subnet.
    - Inter-AS: Routing to AS (cost is traversed ASs)
    - Intra-AS: Routing to subnets in the AS (cost is traversed subnets)
    - Within the subnets: no routing, simple procedure to be specified.
- All routers run both an inter-AS and an intra-AS protocol
- Assume that there is only one host in every subnet. There will then roughly be $\approx 10^5$ AS and $\approx 10^4$ subnets in each AS leading to LSR complexity = $O((10^5)^2) + O((10^4)^2)$ instead of $O((10^9)^2)$.
- Good complexity reduction!

Inter-AS and Intra-AS routing



Forwarding table at the router R :
AS 1: via s33 (hot potato)
AS 2: via s34
s31: via s32
s32: direct
s33: direct
s34: direct
s35: via s34
s36: via s34

Observe that:
1) the forwarding table spans the whole network.
2) The 6 last entries are given by intra routing only.
-The 2 first entries are given by inter and intra routing.
-AS 1 is at equal distance from two routers. Then we find the best path by hot-potato routing, i.e., by only considering intra-costs
3) Intra-routing is performed by a possibly different protocol in each AS
4) Inter-routing is performed by the same protocol all over the network
5) It is easy to translate tables above to IPv4 address ranges.

Forwarding: Storage and search of forwarding tables in routers

- The IP address are ordered in a hierarchical manner with:
  - geographically closely located computers belonging to the same subnet
  - at a router, an interface has a subnet connected to it

- This means that we can hold a network specific table with small storage space and which is easily searched for forwarding in routers

0.0.0.0-127.255.255.255	send on interface A
128.0.0.0-191.255.255.255	send on interface B
192.0.0.0-255.255.255.255	send on interface C

- Small tables are also achieved by:
  - Next-hop table; Only the next hop is stored

Network management

The Internet Control Message Protocol (ICMP)

- ICMP is used to report errors.
- Common errors are *Destination Unreachable*, *TTL Expired*, *Options Handling Issue*, and *Fragmentation Needed But Not Permitted*.
- ICMP messages can also be used to provide information such as reporting a timestamp or the local link's subnet mask.
- ICMP is the mechanism behind the *echo* request and response functionality referred to as the *ping* command.
- ICMP messages are contained within the standard IP datagrams (layer 3).
- For deeper analysis and management tasks, one can use SNMP (Simple Network Management Protocol). This is an application layer protocol.

traceroute

- a command that uses ICMP

- Uses the parameter Time To Live (TTL) in the IP header. Each router subtracts 1 from TTL. When TTL=0, the router does not forward the packet, but sends an ICMP error message back to the destination.

- traceroute first sends a datagram with TTL=1, to get a message from the first router. Then it sends a datagram with TTL=2, to get a message from the second router. It continues like this until the final destination is reached, and traceroute can finally list the whole route with propagation delays to each router. The last stop is the destination. Here we use a wrong user datagram protocol (UDP) port to provoke an ICMP error message from the host- more about UDP in the next lecture!

```
traceroute to google.com (209.85.149.106), 30 hops max, 60 byte packets
1 isylogon.isy.liu.se (130.236.58.1) 0.210 ms
2 isy-gw.isy.liu.se (130.236.48.1) 1.964 ms
3 130.236.6.21 (130.236.6.21) 0.826 ms
4 green-a.net.liu.se (130.236.9.6) 32.037 ms
5 liu-br2.sunet.se (193.11.0.21) 0.795 ms
6 m1tug-xe-7-3-3.sunet.se (130.242.85.173) 22.578 ms
7 t1tug-ae1-v1.sunet.se (130.242.83.41) 49.556 ms
8 se-tug.nordu.net (109.105.102.17) 3.669 ms
9 se-tug2.nordu.net (109.105.97.18) 3.757 ms
10 google-gw.nordu.net (109.105.98.6) 4.969 ms
11 216.239.43.122 (216.239.43.122) 4.967 ms
12 209.85.254.153 (209.85.254.153) 22.490 ms
13 209.85.254.21 (209.85.254.21) 31.918 ms
14 ber01s02-in-f106.1e100.net (209.85.149.106) 23.976 ms
```

Try several network diagnostics tools online at http://network-tools.com