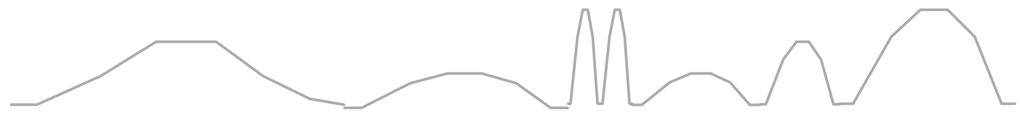
Datalink – Framing, Switching

From Signals to Packets

Analog Signal

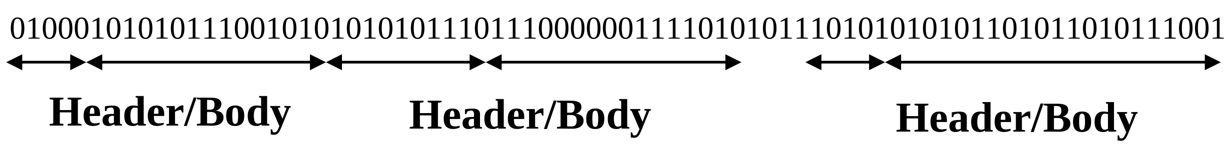


"Digital" Signal

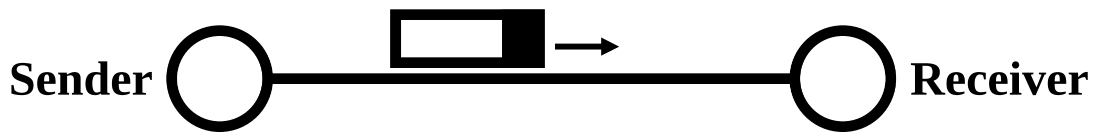
Bit Stream

0 0 1 0 1 1 1 0 0 0 1

Packets



Packet Transmission

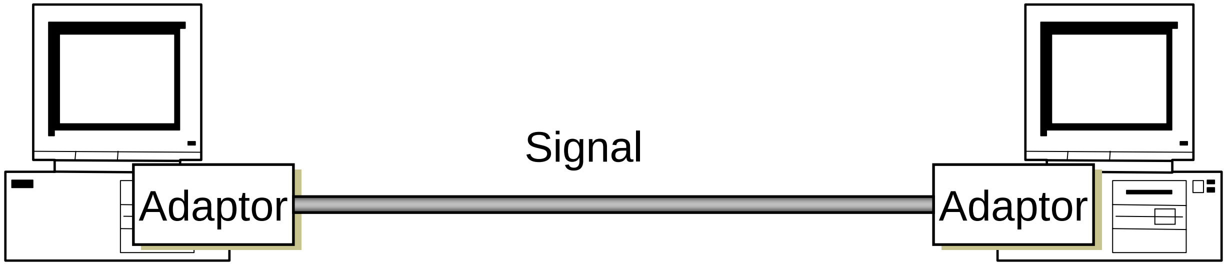


Datalink Functions

- Framing: encapsulating a network layer datagram into a bit stream.
  - Add header, mark and detect frame boundaries
- Media access: controlling which frame should be sent over the link next.
- Error control: error detection and correction to deal with bit errors.
  - May also include other reliability support, e.g. retransmission
- Flow control: avoid that the sender outruns the receiver
- Hubbing, bridging: extend the size of the network

Encoding

Mapping bits into signal



Adaptor: convert bits into physical signal and physical signal back into bits

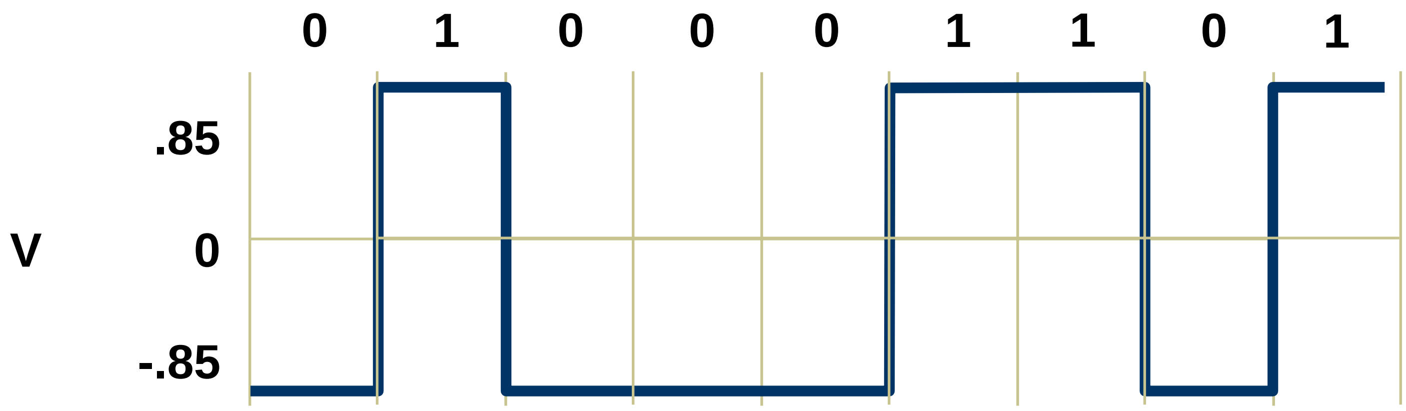
Why Do We Need Encoding?

- Meet certain electrical constraints.
  - Receiver needs enough “transitions” to keep track of the transmit clock
  - Avoid receiver saturation
- Create control symbols, besides regular data symbols.
  - E.g. start or end of frame, escape, ...
- Error detection or error corrections.
  - Some codes are illegal so receiver can detect certain classes of errors
  - Minor errors can be corrected by having multiple adjacent signals mapped to the same data symbol
- Encoding can be very complex, e.g. wireless.

Encoding

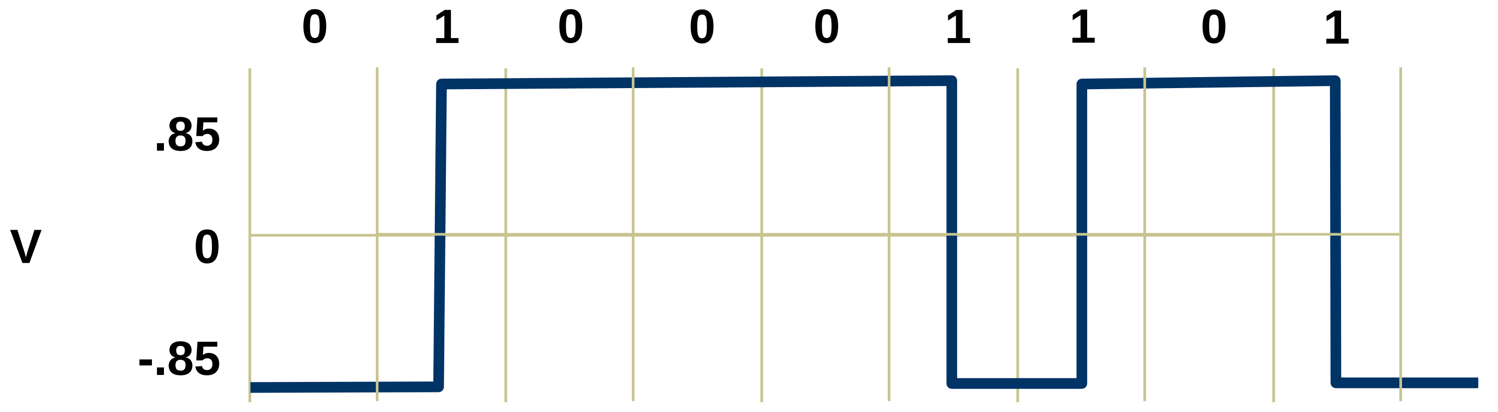
- We use two discrete signals, high and low, to encode 0 and 1
- The transmission is synchronous, i.e., there is a clock used to sample the signal
  - In general, the duration of one bit is equal to one or two clock ticks

Non-Return to Zero (NRZ)



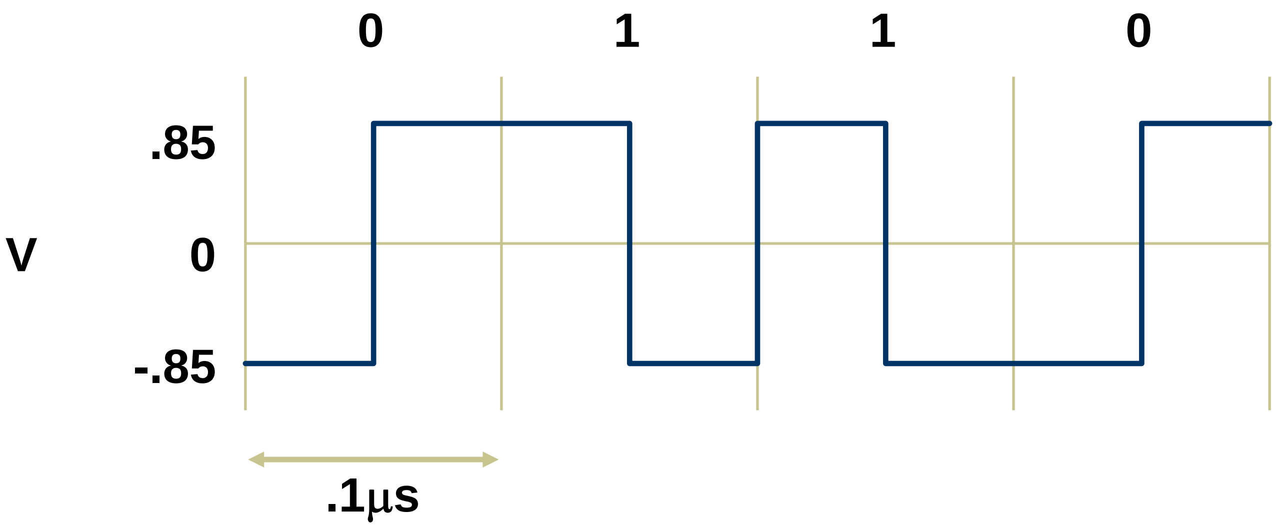
- 1 -> high signal; 0 -> low signal
- Long sequences of 1's or 0's can cause problems:
  - Sensitive to clock skew, i.e. hard to recover clock
  - Difficult to interpret 0's and 1's

Non-Return to Zero Inverted (NRZI)



- 1 -> make transition; 0 -> signal stays the same
- Solves the problem for long sequences of 1's, but not for 0's.

Ethernet Manchester Encoding



- Positive transition for 0, negative for 1
- Transition every cycle communicates clock (but need 2 transition times per bit)
- DC balance has good electrical properties

4B/5B Encoding

- Data coded as *symbols* of 5 line bits => 4 data bits, so 100 Mbps uses 125 MHz.
  - Uses less frequency space than Manchester encoding
- Uses NRI to encode the 5 code bits
- Each valid symbol has at least two 1s: get dense transitions.
- 16 data symbols, 8 control symbols
  - Data symbols: 4 data bits
  - Control symbols: idle, begin frame, etc.
- Example: FDDI.

4B/5B Encoding

Data	Code
0000	11110
0001	01001
0010	10100
0011	10101
0100	01010
0101	01011
0110	01110
0111	01111

Data	Code
1000	10010
1001	10011
1010	10110
1011	10111
1100	11010
1101	11011
1110	11100
1111	11101

Other Encodings

- 8B/10B: Fiber Channel and Gigabit Ethernet
  - DC balance
- 64B/66B: 10 Gbit Ethernet
- B8ZS: T1 signaling (bit stuffing)

Error Coding

- Transmission process may introduce errors into a message.
  - Single bit errors versus burst errors
- Detection:
  - Requires a convention that some messages are invalid
  - Hence requires extra bits
  - An (n,k) code has codewords of n bits with k data bits and r = (n-k) redundant check bits
- Correction
  - Forward error correction: many related code words map to the same data word
  - Detect errors and retry transmission

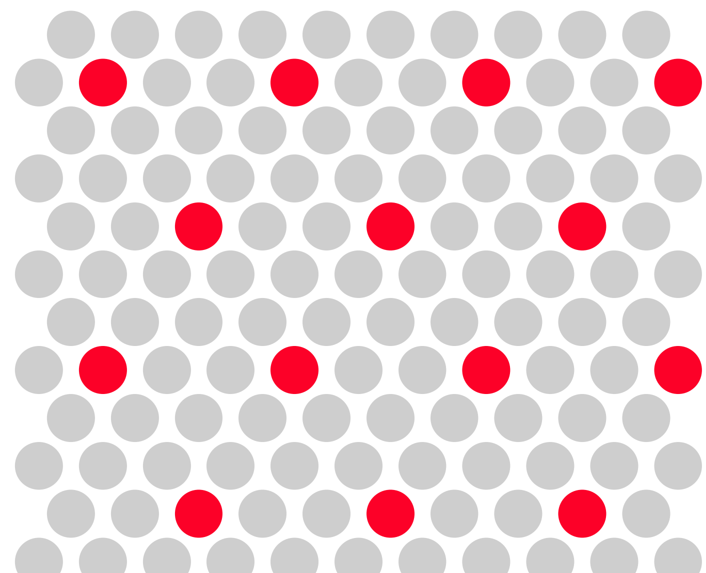
Basic Concept: Hamming Distance

- *Hamming distance* of two bit strings = number of bit positions in which they differ.
- If the valid words of a code have minimum Hamming distance D, then D-1 bit errors can be detected.
- If the valid words of a code have minimum Hamming distance D, then [(D-1)/2] bit errors

1	0	1	1	0
1	1	0	1	0

HD=2

HD=3



Cyclic Redundancy Codes (CRC)

- Commonly used codes that have good error detection properties.
  - Can catch many error combinations with a small number or redundant bits
- Based on division of polynomials.
  - Errors can be viewed as adding terms to the polynomial
  - Should be unlikely that the division will still work
- Can be implemented very efficiently in hardware.
- Examples:
  - CRC-32: Ethernet
  - CRC-8, CRC-10, CRC-32: ATM

Framing

- A link layer function, defining which bits have which function.
- Minimal functionality: mark the beginning and end of packets (or frames).
- Some techniques:
  - out of band delimiters (e.g. FDDI 4B/5B control symbols)
  - frame delimiter characters with character stuffing
  - frame delimiter codes with bit stuffing
  - synchronous transmission (e.g. SONET)

Character and Bit Stuffing

- Mark frames with special character.
  - What happens when the user sends this character?
  - Use escape character when controls appear in data:

    *abc*def -> *abc*def
  - Very common on serial lines, in editors, etc.
- Mark frames with special bit sequence
  - must ensure data containing this sequence can be transmitted
  - example: suppose 11111111 is a special sequence.
  - transmitter inserts a 0 when this appears in the data:
  - 11111111 -> 111111101
  - must stuff a zero any time seven 1s appear:
  - 11111110 -> 111111100
  - receiver unstuffs.

Example: Ethernet Framing

- Preamble is 7 bytes of 10101010 (5 MHz square wave) followed by one byte of 10101011
- Allows receivers to recognize start of transmission after idle channel

Baud Rate, Bandwidth, Clock Rate, Bit Rate

- Nyquist: maximum baud rate given a fixed bandwidth (frequency range)
- Many practical issues that may result in lower bit rate
  - Encoding overhead to deal with physical layer issues
  - Encoding overhead to handle errors
  - Bit/byte stuffing
- Application throughput is lower than physical bit rate, why?

Other Issues Impacting Performance

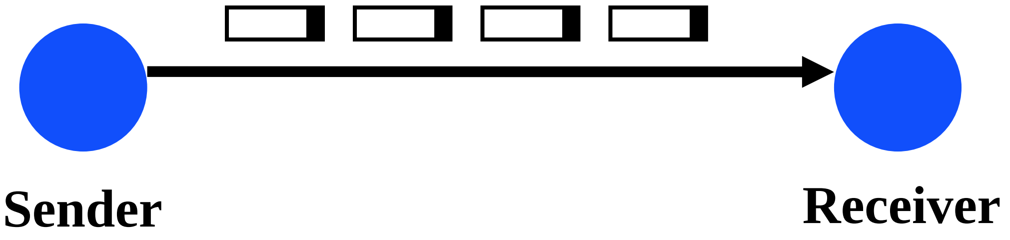
- Contention resolution (last lecture)
- Reliability control
- Congestion control
- Flow control

Link Flow Control and Error Control

- Naïve protocol.
- Dealing with receiver overflow: flow control.
- Dealing with packet loss and corruption: error control.
- Meta-comment: these issues are relevant at many layers.
  - Link layer: sender and receiver attached to the same “wire”
  - End-to-end: transmission control protocol (TCP) - sender and receiver are the end points of a connection
- How can we implement flow control?
  - “You may send” (windows, stop-and-wait, etc.)
  - “Please shut up” (source quench, 802.3x pause frames, etc.)
  - Where are each of these appropriate?

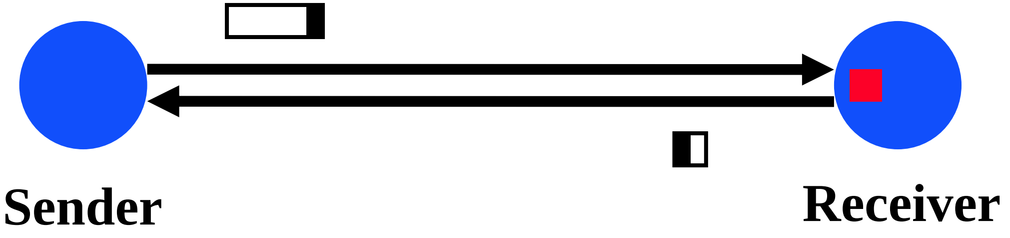
A Naïve Protocol

- Sender simply sends to the receiver whenever it has packets.
- Potential problem: sender can outrun the receiver.
  - Receiver too slow, buffer overflow, ..
- Not always a problem: receiver might be fast enough.



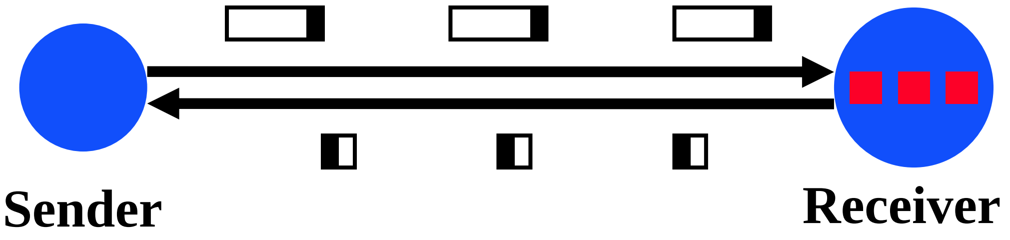
Adding Flow Control

- Stop and wait flow control: sender waits to send the next packet until the previous packet has been acknowledged by the receiver.
  - Receiver can pace the receiver
- Drawbacks: adds overheads, slowdown for long links.



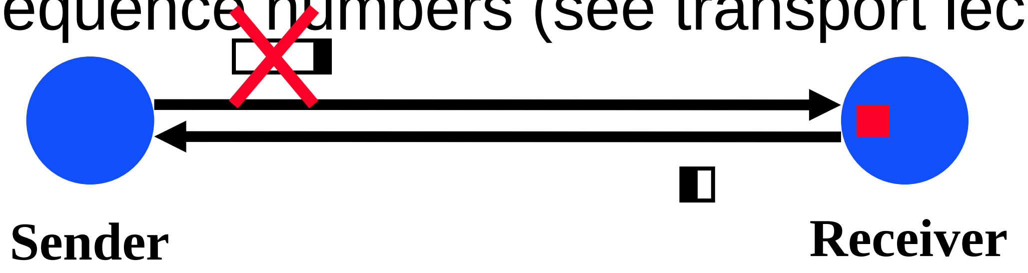
Window Flow Control

- Stop and wait flow control results in poor throughput for long-delay paths: packet size/ roundtrip-time.
- Solution: receiver provides sender with a window that it can fill with packets.
  - The window is backed up by buffer space on receiver
  - Receiver acknowledges the a packet every time a packet is consumed and a buffer is freed



Dealing with Errors
Stop and Wait Case

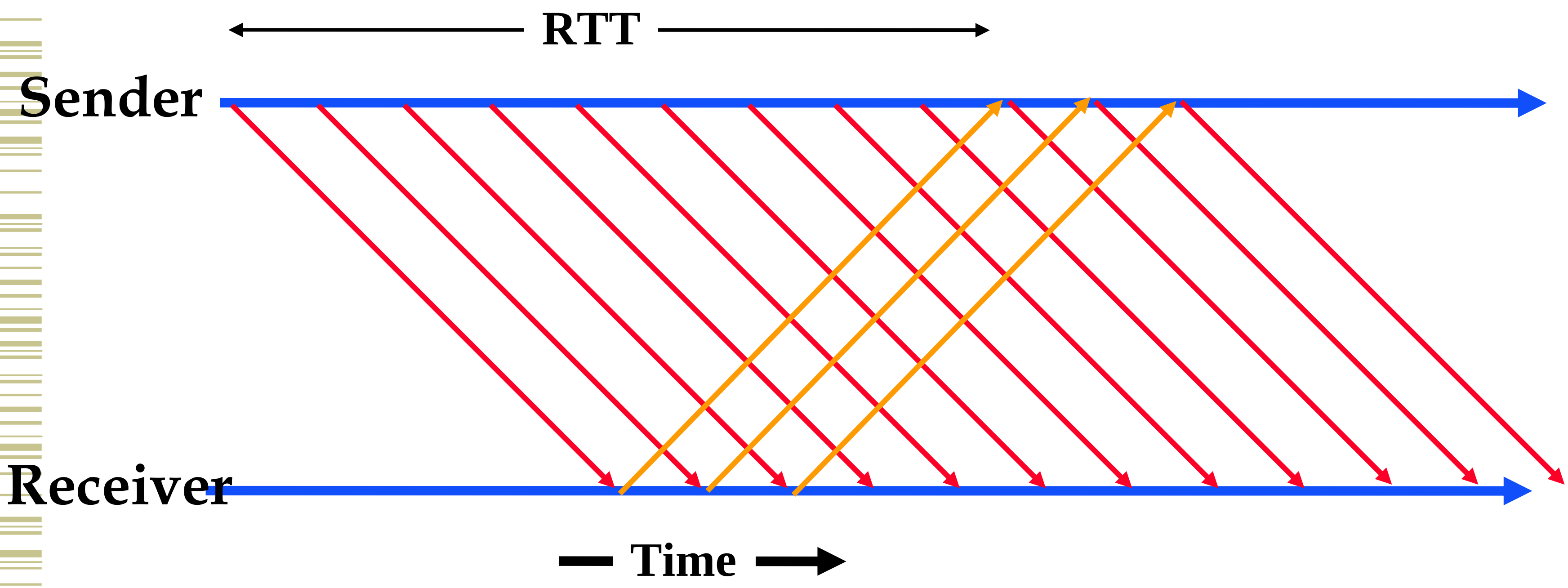
- Packets can get lost, corrupted, or duplicated.
  - Error detection or correction turns corrupted packet in lost or correct packet
- Duplicate packet: use sequence numbers.
- Lost packet: time outs and acknowledgements.
  - Positive versus negative acknowledgements
  - Sender side versus receiver side timeouts
- Window based flow control: more aggressive use of sequence numbers (see transport lectures).



Issues with Window-based Protocol

- Receiver window size: # of out-of-sequence packets that the receiver can receive
- Sender window size: # of total outstanding packets that sender can send without acknowledged
- How to deal with sequence number wrap around?

Bandwidth-Delay Product

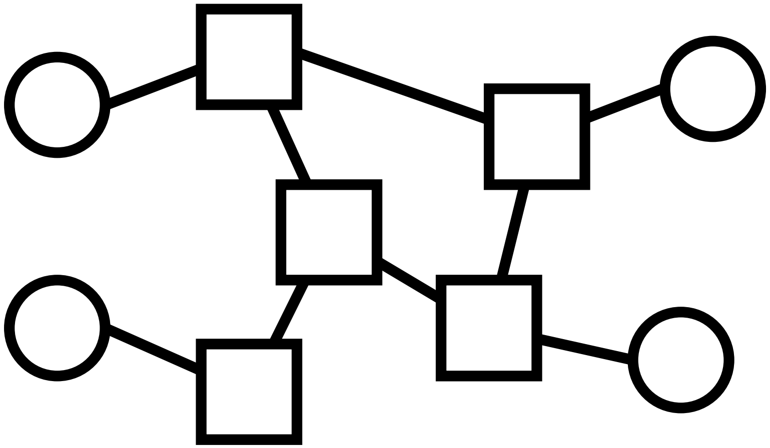


$$\text{Max Throughput} = \frac{\text{Window Size}}{\text{Roundtrip Time}}$$

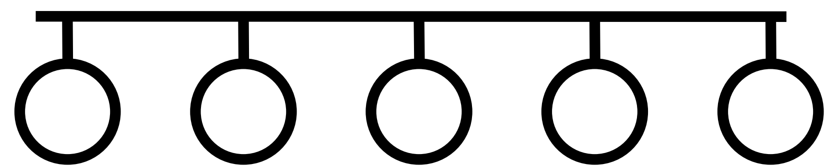
Physical and Data Link

- Medium
  - Unshielded Twisted Pair (UTP)
  - coaxial cable: baseband, broadband
  - fiber: multi-mode, single mode
  - radio, infrared
- LAN technologies
  - Ethernet: CSMA-CD protocol
  - Fast Ethernet, Gigabit Ethernet
  - FDDI, Token Ring
  - ATM
- WAN technologies
  - analog transmission: modem
  - digital transmission: T-1, T-3, Sonet, OC-3, OC-12
  - ATM, frame relay

Datalink Architectures



- Packet forwarding.
- Error and flow control.



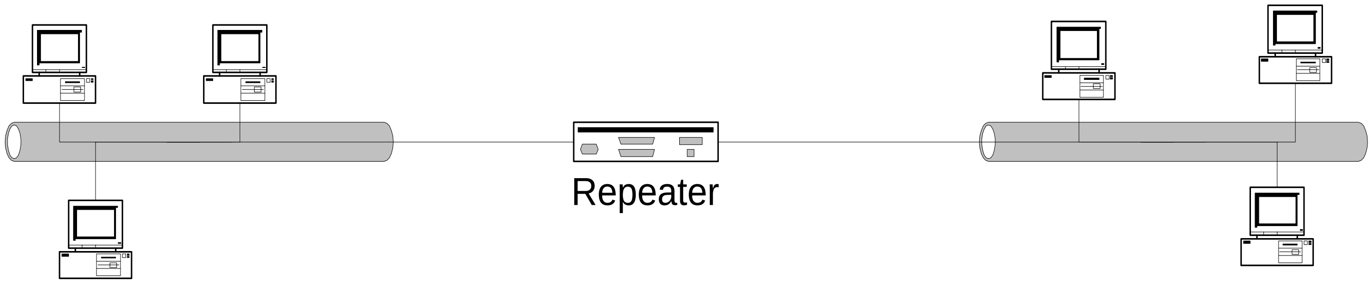
- Media access control.
- Scalability.

Media Access Control

- How do we transfer packets between two hosts connected to the same network?
- Switches connected by point-to-point links -- store-and-forward.
  - Used in WAN, LAN, and for home connections
  - Conceptually similar to "routing"
    - But at the datalink layer instead of the network layer
  - Today
- Multiple access networks -- contention based.
  - Multiple hosts are sharing the same transmission medium
  - Used in LANs and wireless
  - Need to control access to the medium

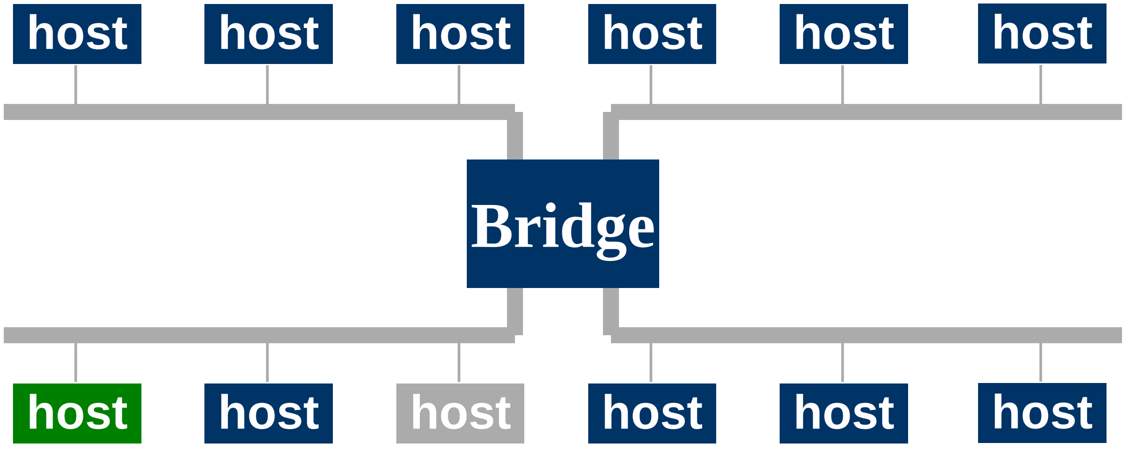
Repeaters

- Used to interconnect multiple Ethernet segments
- Merely extends the baseband cable
- Amplifies all signals including collisions



Building Larger LANs: Bridges

- Bridges connect multiple IEEE 802 LANs at layer 2.
  - Only forward packets to the right port
  - Reduce collision domain compared with single LAN
- In contrast, hubs rebroadcast packets.



Transparent Bridges

- Overall design goal: **Complete transparency**
  - "Plug-and-play"
  - Self-configuring without hardware or software changes
  - Bridges should not impact operation of existing LANs

- Three parts to transparent bridges:

  (1) Forwarding of Frames

  (2) Learning of Addresses

  (3) Spanning Tree Algorithm

Frame Forwarding

- Each bridge maintains a **forwarding database** with entries

  < MAC address, port, age>

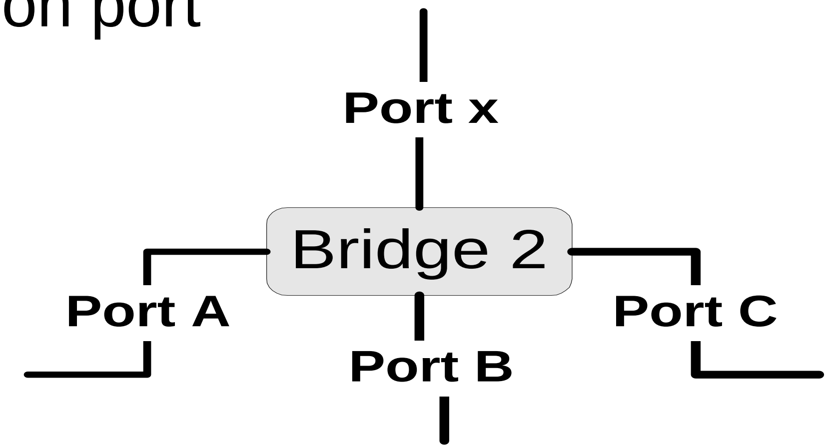
MAC address:	host name or group address
port:	port number of bridge
age:	aging time of entry

with interpretation:

- a machine with **MAC address** lies in direction of the **port** number from the bridge. The entry is **age** time units old.

Frame Forwarding 2

- Assume a MAC frame arrives on port x.



Search if MAC address of destination is listed for ports A, B, or C.

Found?

Forward the frame on the appropriate port

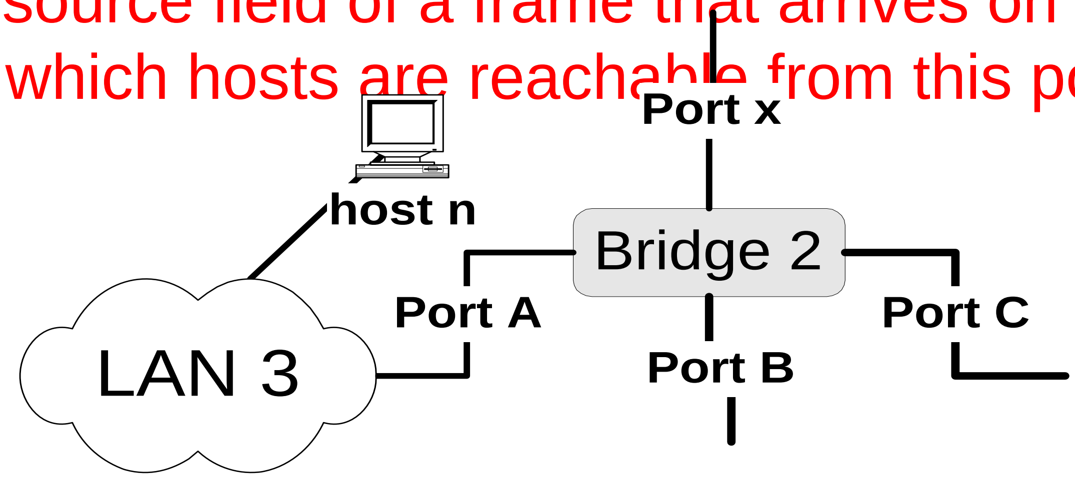
Not found ?

Flood the frame, i.e., send the frame on all ports except port x.

Address Learning

- In principle, the forwarding database could be set statically (=static routing)
- In the 802.1 bridge, the process is made automatic with a simple heuristic:

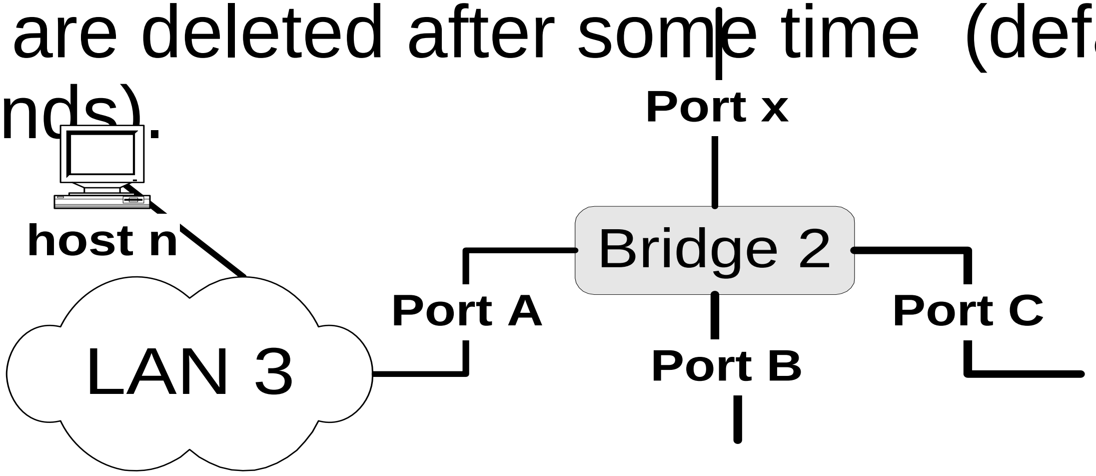
The source field of a frame that arrives on a port tells which hosts are reachable from this port.



Address Learning 2

Algorithm:

- For each frame received, the source stores the source field in the forwarding database together with the port where the frame was received.
- All entries are deleted after some time (default is 15 seconds).

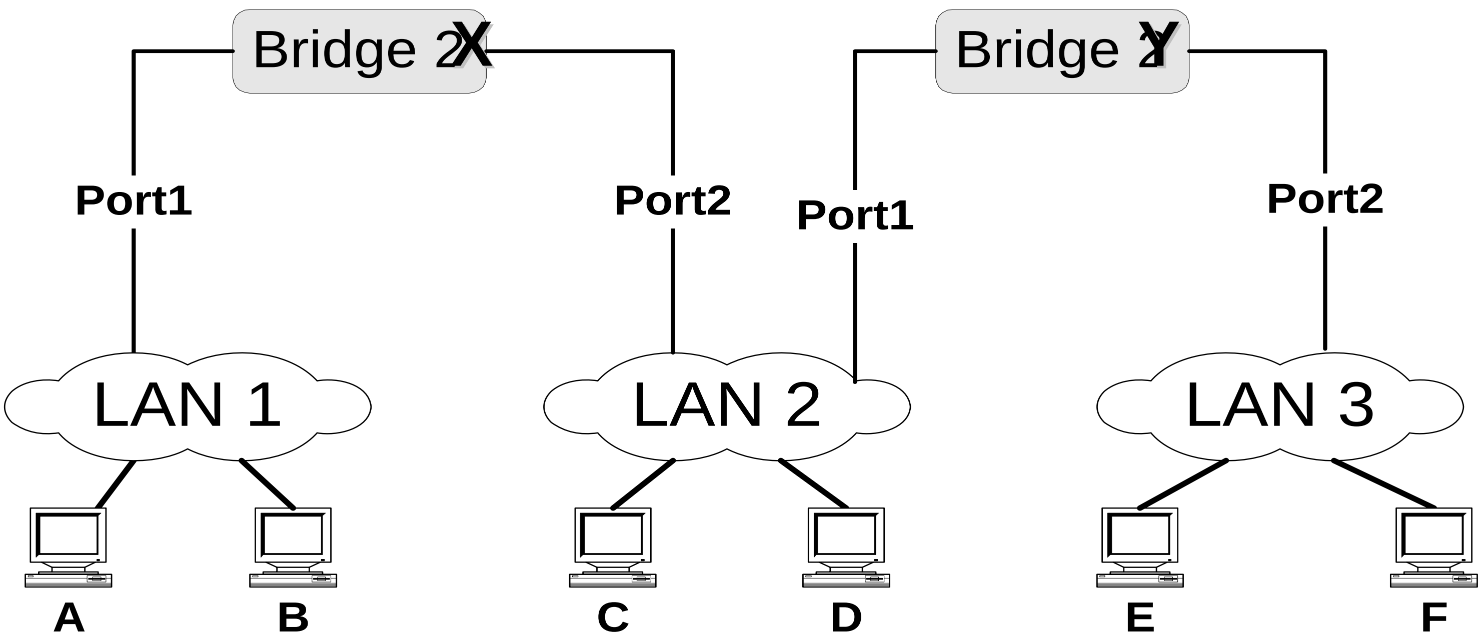


Example

- Consider the following packets:

<Src=A, Dest=F>, <Src=C, Dest=A>, <Src=E, Dest=C>

- What have the bridges learned?



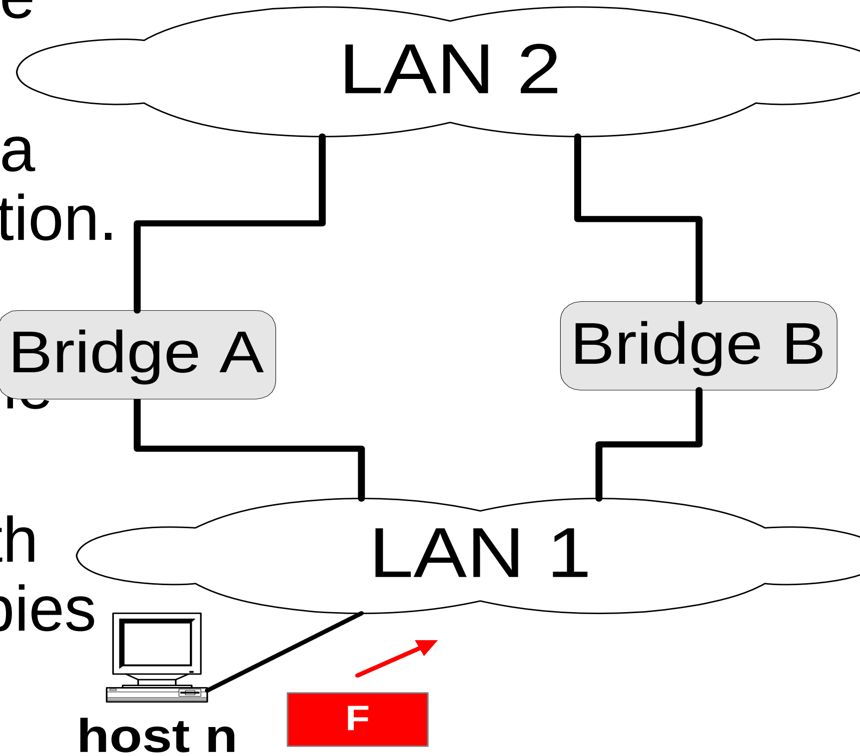
Danger of Loops

- Consider the two LANs that are connected by two bridges.
- Assume *host n* is transmitting a frame F with unknown destination.

What is happening?

- Bridges A and B flood the frame to LAN 2.
- Bridge B sees F on LAN 2 (with unknown destination), and copies the frame back to LAN 1
- Bridge A does the same.
- The copying continues

Where's the problem? What's the solution ?



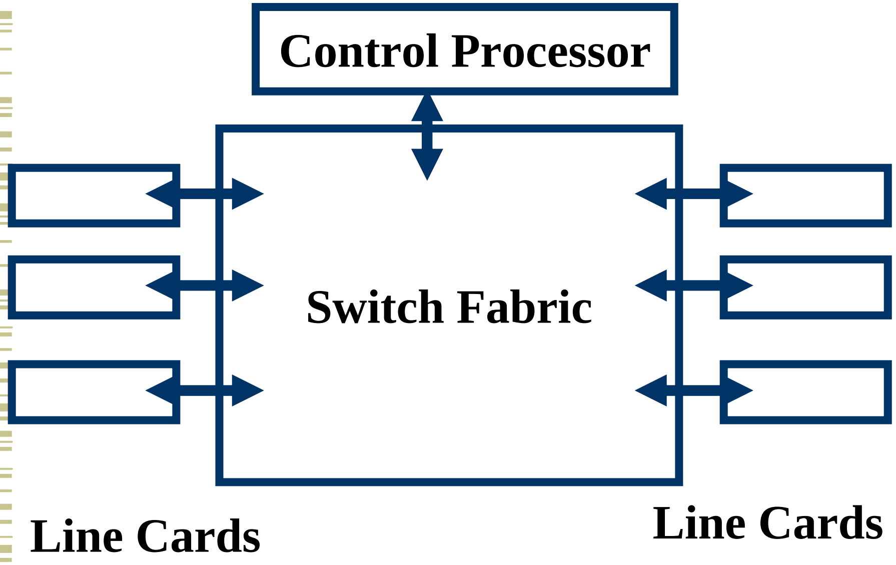
Spanning Trees

- The solution to the loop problem is to not have loops in the topology
- IEEE 802.1 has an algorithm that builds and maintains a spanning tree in a dynamic environment.
- Bridges exchange messages to configure the bridge (Configuration Bridge Protocol Data Unit, Configuration BPDUs) to build the tree.

Ethernet Switches

- Bridges make it possible to increase LAN capacity.
  - Packets are no longer broadcasted - they are only forwarded on selected links
  - Adds a switching flavor to the broadcast LAN
- Ethernet switch is a special case of a bridge: each bridge port is connected to a single host.
  - Can make the link full duplex (really simple protocol!)
  - Simplifies the protocol and hardware used (only two stations on the link) – no longer full CSMA/CD
  - Can have different port speeds on the same switch
    - Unlike in a hub, packets can be stored
    - An alternative is to use cut through switching

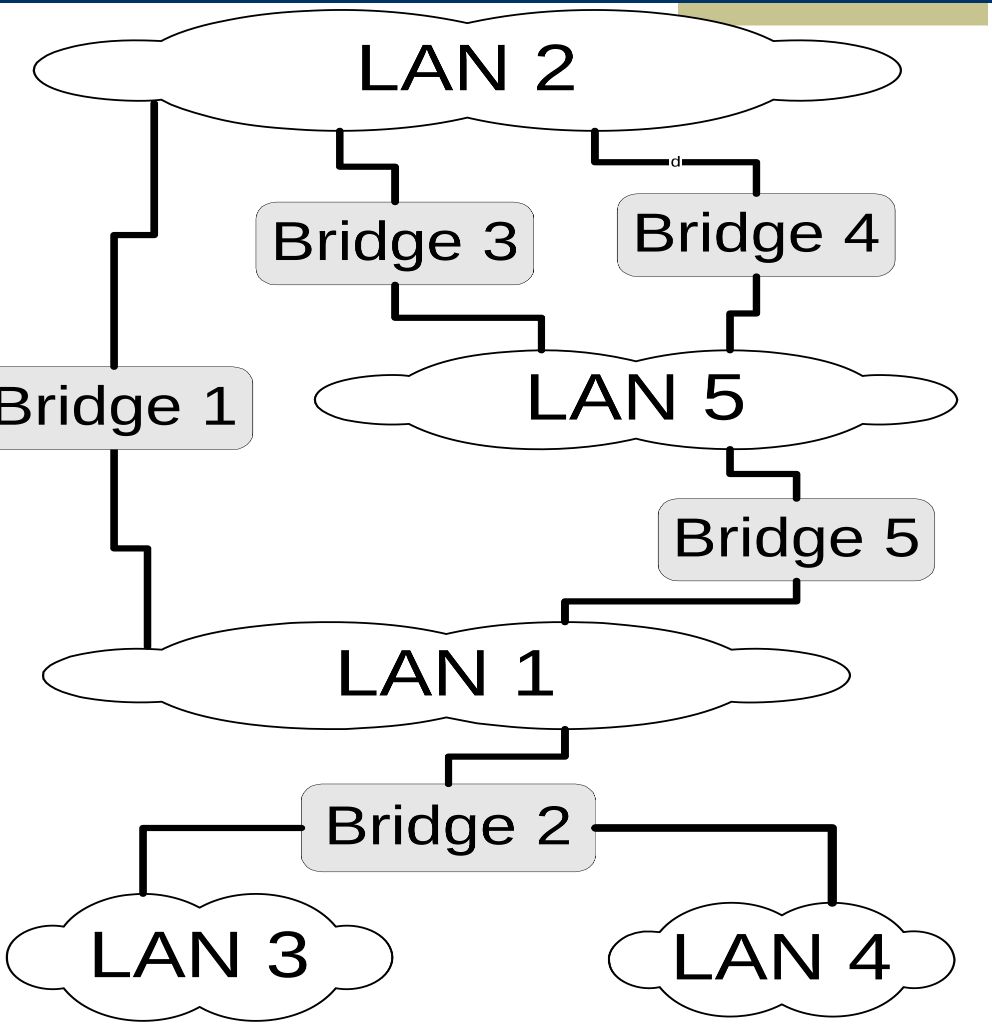
Structure of A Generic Communication Switch



- Switch fabric
  - high capacity interconnect
- Line card
  - address lookup in the data path (forwarding)
- Control Processor
  - load the forwarding table (routing or signaling)

- Switches
  - circuit switch
  - Ethernet switch
  - ATM switch
  - IP router

What Are the Issues of Bridging?



Long Distance Transmission

- For historical reasons, long-haul links, standards are determined by telephone networks
- Bandwidth of telephone channel is under 4KHz, so when digitizing:

  8000 samples/sec * 8 bits = 64Kbits/second

- Common data rates supported by telcos in North America:
  - Modem: rate improved over the years
  - T1/DS1: 24 voice channels plus 1 bit per sample

    (24 * 8 + 1) * 8000 = 1.544 Mbits/second

  - T3/DS3: 28 T1 channels:

    7 * 4 * 1.544 = 44.736 Mbits/second

Synchronous Data Transfer

- Optical transmission standard adopted by telephone companies
- Sender and receiver are always synchronized.
  - Frame boundaries are recognized based on the clock
  - No need to continuously look for special bit sequences
- SONET frames contain room for control and data.
  - Data frame multiplexes bytes from many users
  - Control provides information on data, management, …

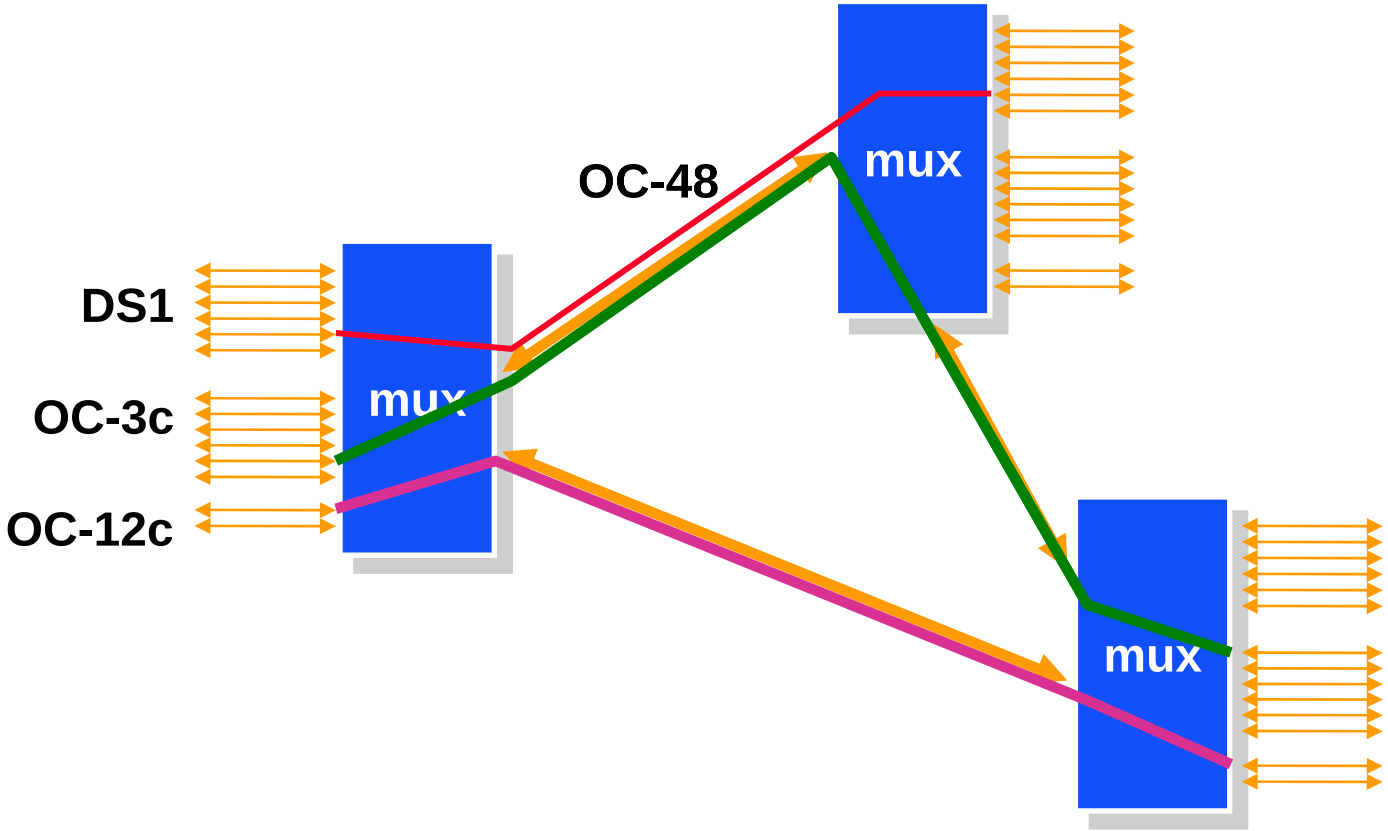
The SONET Signal Hierarchy

STS-1 carries one DS-3 plus overhead

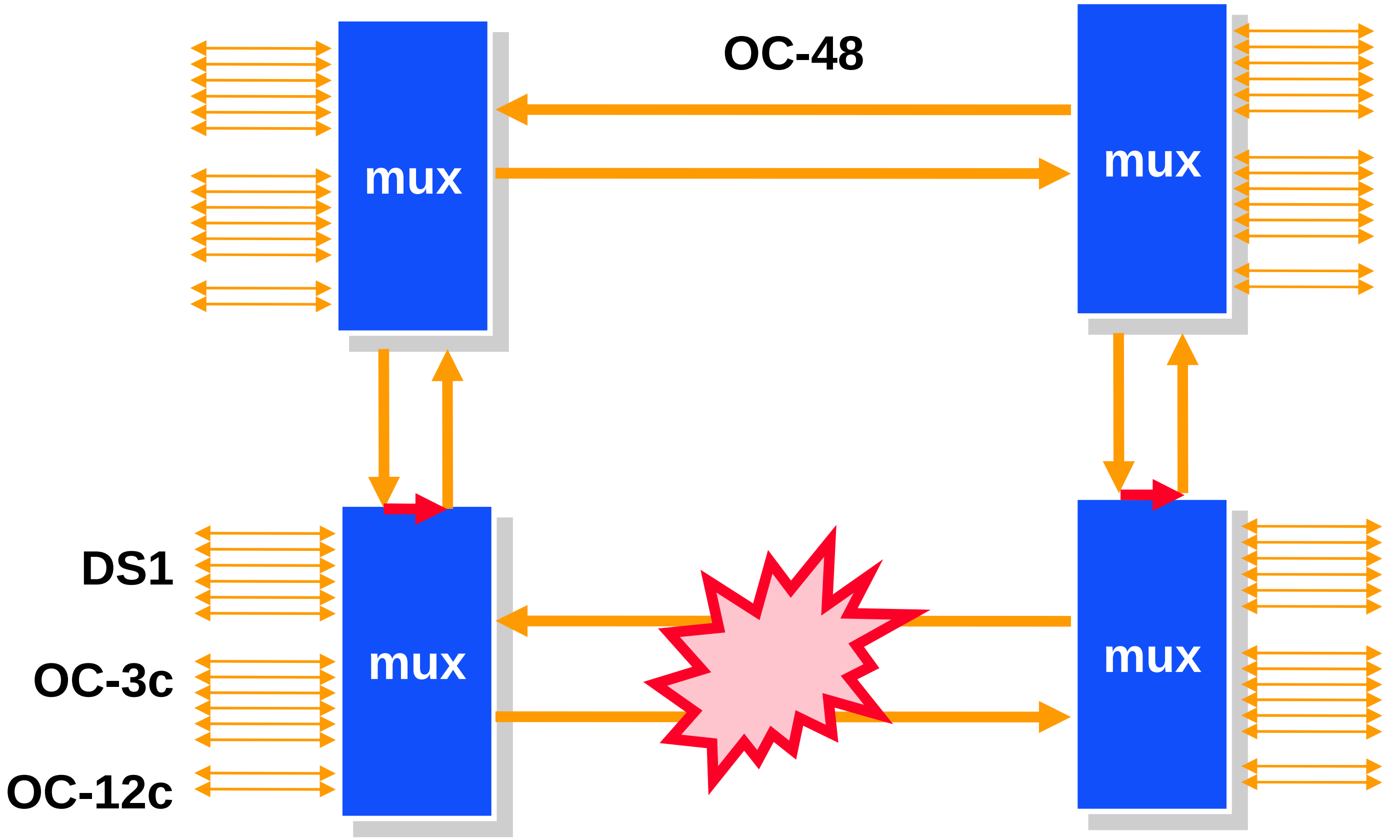
Signal Type	line rate	# of DS0
DS0 (POTS)	64 Kbs	1
DS1	1.544 Mbs	24
DS3	44.736 Mbs	672
OC-1	51.84 Mbs	672
OC-3	155 Mbs	2,016
OC-12	622 Mbs	8,064
STS-48	2.49 Gbs	32,256
STS-192	9.95 Gbs	129,024
STS-768	39.8 Gbs	516,096

SONET Can Be A Network

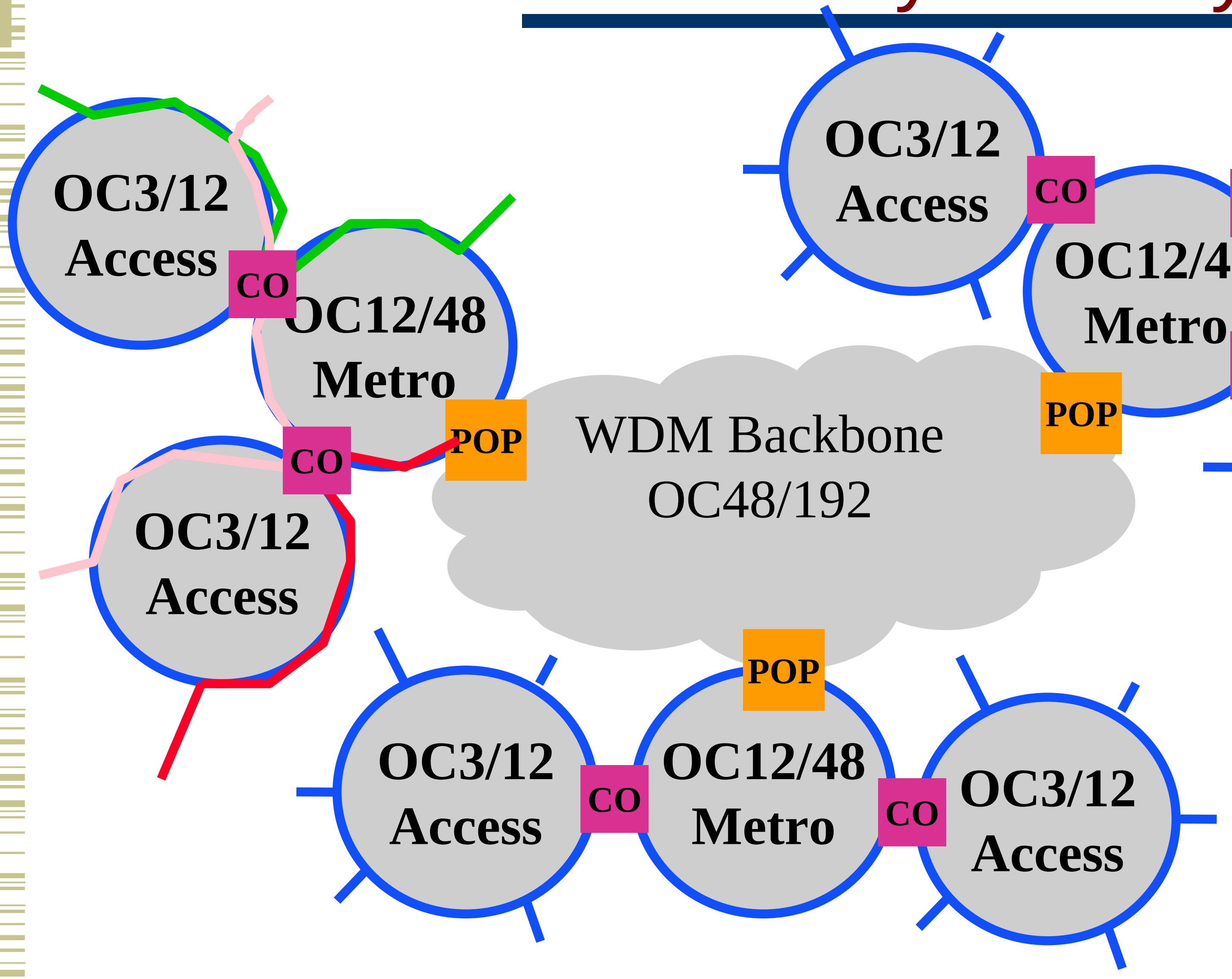
Add-drop capability allows soft configuration of networks, usually managed manually.



Self-Healing SONET Rings



SONET Network as Physical Layer



Addressing and Look-up

- Flat address
  - Ethernet: 48 bit MAC address
  - ATM: 28 bit VPI/VCI
  - DS-0: timeslot location
- Limited scalability
- High speed lookup

- Hierarchical address
  - IP <network>.<subnet>.<host>
  - Telephone: country.area.home
- Scalable
- Easy lookup if boundary is fixed
  - telephony
- Difficult lookup if boundary is flexible
  - longest prefix match for IP