Computer Networks I

application
transport
network
link
physical

Data link control

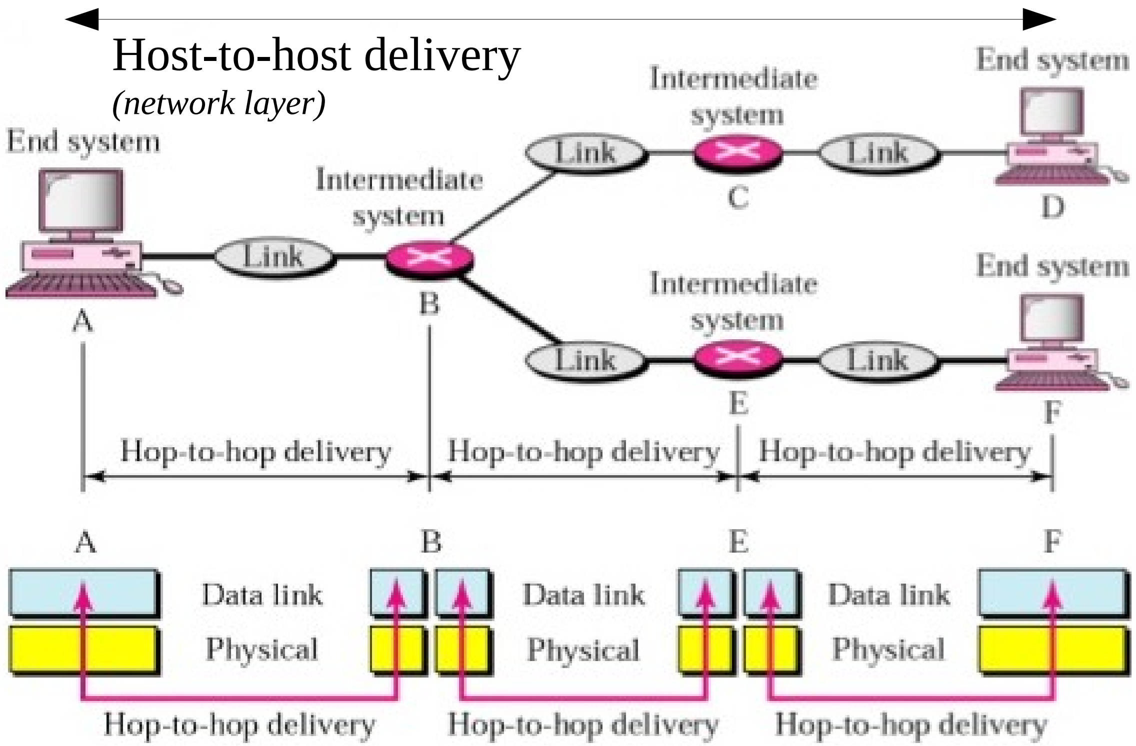
Fernando.rincon@uclm.es

Outline

- Introduction
- Data link control
- Link protocols
- Ethernet / IEEE standards
- ARP protocol
- Colision domains
    - Bridges and switches

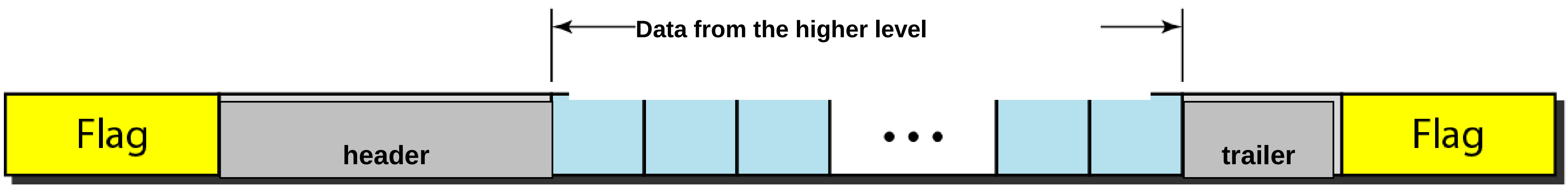
Data link control

- Deals with node-to-node (hop-to-hop) communication.
  - Built on top of the physical layer
  - Provides interconnection service to the network layer
  - Direct communication (no mediators)
- Main problems:
  - Data transmission errors
  - Propagation delay
  - Framing



Frames

- In the link layer, data (bits) are packed into frames
- A frame encloses:
  - A header (i.e. Destination address, size, etc)
  - Payload (comming from network layer)
  - Trailer (i.e. CRC error)
  - Flags to delimiter start and end of frame (Flag:01111110)



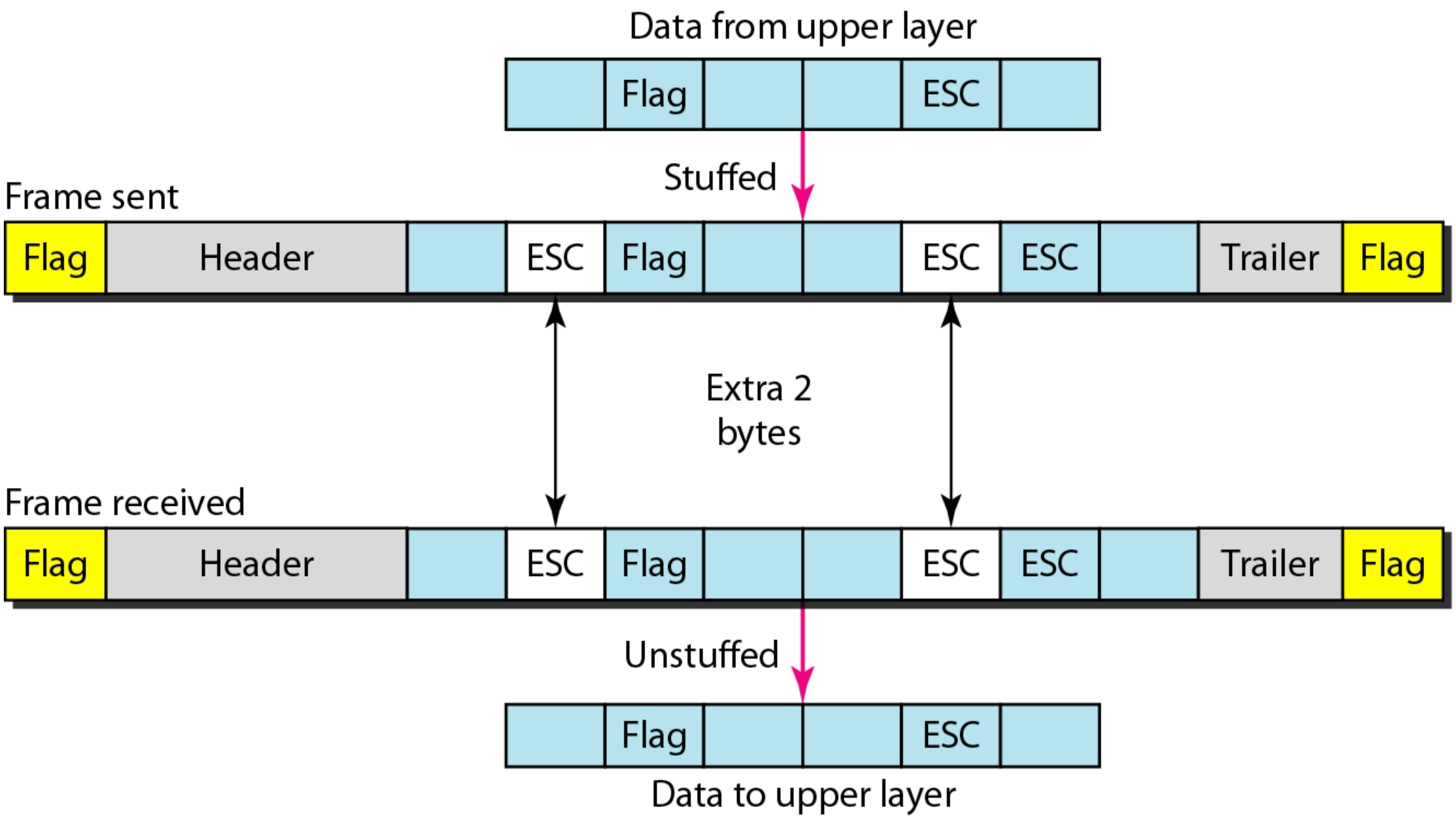
Framing

- Framing is the encapsulation process
  - Division bits stream intro frames (fixed or variable size)
- Frames usually include error detection or correction codes (i.e. Checksum)
- Different types of frame delimiters (variable size):
  - Character oriented (byte stuffing)
  - Bit oriented (bit stuffing)

Character oriented protocol

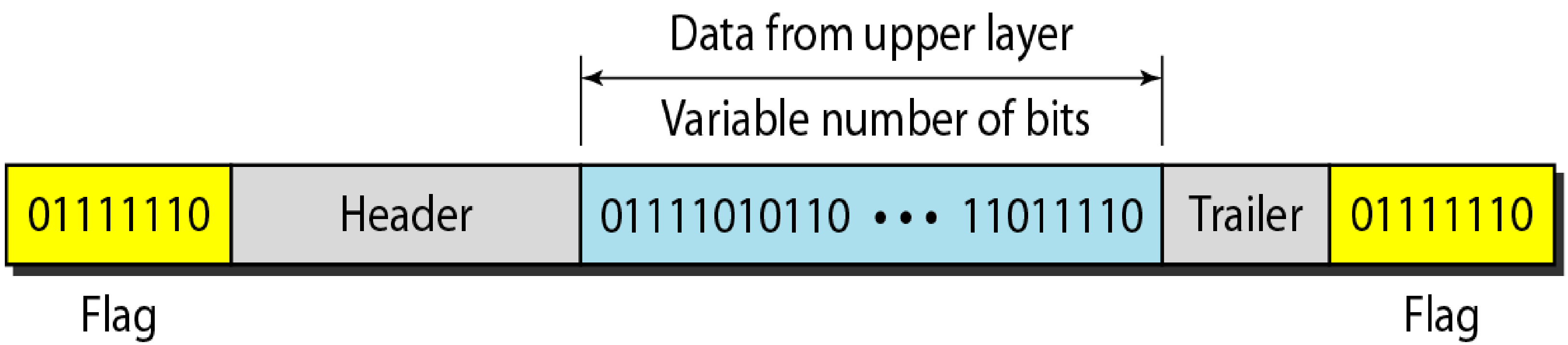
- A special byte (FLAG: 01111110) delimiters start and end of the frame
- Problems when data contains a value identical to the FLAG: can be confused with a real FLAG
- Solution: byte stuffing
  - Add a special prefix before the FLAG
  - Prefix is the ESC character
  - ESC can also prefix ESC when data value = ESC and avoid the same confusion

Byte stuffing Example



Bit oriented protocol

- Same idea but working at the level of bit
  - Flag is byte 01111110
  - Si there’s data with value “011111”, and extra ‘0’ is added after the last ‘1’ to avoid six ones in a row
  - This way it’s not confuse with the flag
  - When the receiver sees the value “0111110”, it will translate it into “011111”

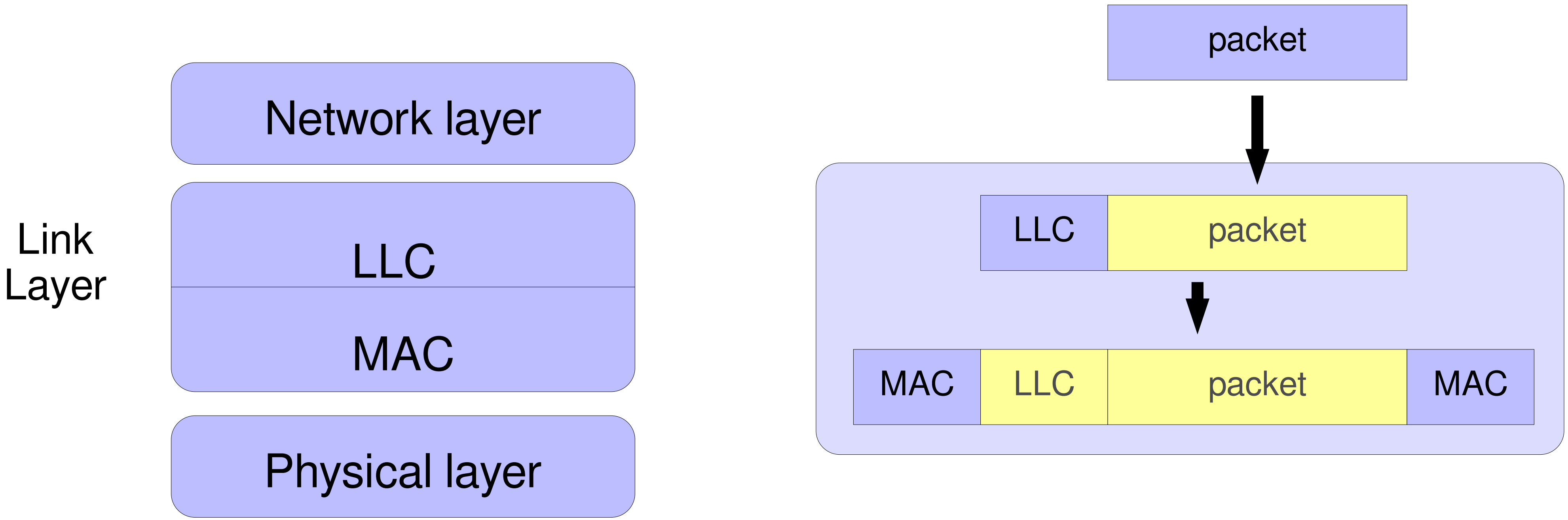


Ethernet

- First LAN technology in history
- Designed by Xerox in 1976. First published by Digital, Xerox and Intel in 1980 (open)
- Standard IEEE in 1985 (802.3)
- Backwards compatibility in following evolutions
- We have reached the fourth generation
  - Standard Eth (10Mbps), Fast Eth (100Mbps), Gigabit Eth (1Gbps), Ten-Gigabit Eth (10Gpbs)

Link layer in LAN

- 2 sublayers
  - LLC (Logical Link Control)
  - MAC (Media Access Control)

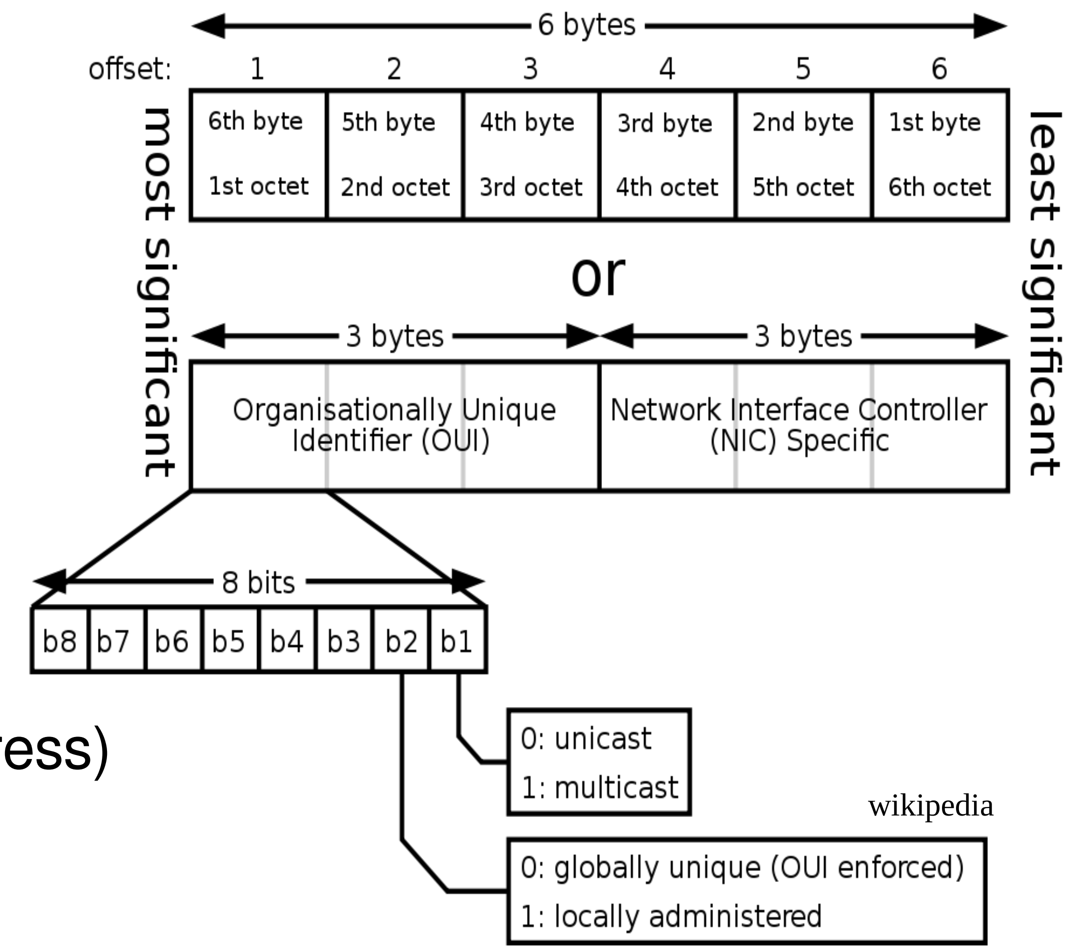


MAC addresses

- Each host in the ethernet network has a NIC (Network Interface Card)
  - The NIC must have a unique address (in ROM)
- The Ethernet address has 48 bits (6 bytes)
  - Written as 12 hex digits
  - Bytes separated by colons
  - Ex. 00:00:0C:23:A5:7D

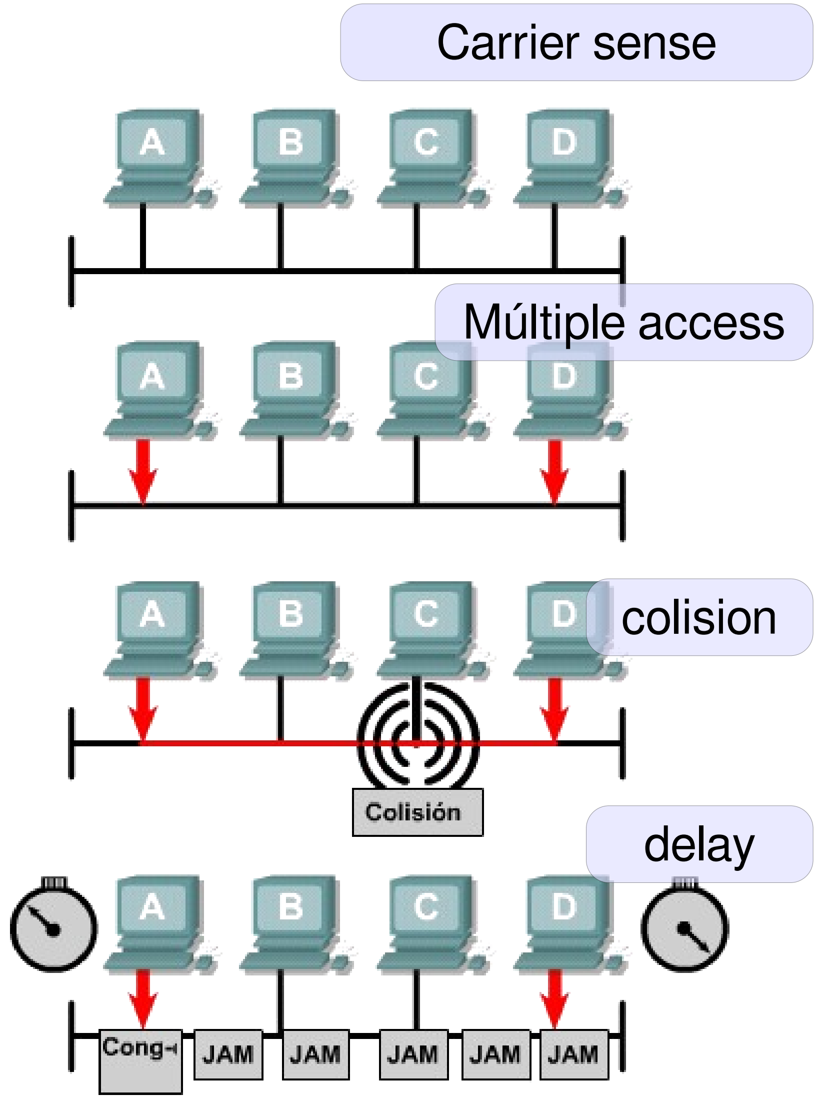
MAC address types

- Globally unique
  - The first 2 bits of the MSB octet are 00
  - ???? ??00:--:--:--:--:--
- Locally unique
  - The first 2 bits of the MSB octet are 10
  - ???? ??10:--:--:--:--:--
- Broadcast (not valid as source address)
  - FF:FF:FF:FF:FF:FF
- Multicast groups (not valid as source address)
  - First bit of the LSB is 1
  - ???? ???1:--:--:--:--:--
    –



Ethernet MAC layer

- Diffusion link with no controller to assign medium access
  - Shared media, called *collision domain*
  - *Collisions happen because nodes try to communicate at the same time*
- CSMA/CD (*Carrier Sense Multiple Access / Collision Detect*)
  - A way to avoid collisions
  - First listen channel
  - If not used, transmit and check you only here yourself (otherwise collision)
  - If channel busy wait for a certain time:
    - Immediately (IEEE 802.3)
    - Random
  - Collisions degrade performance



Ethernet frame format

Preamble	Target address	Source address	type	data	Checksum FCS
8	6	6	2	46-1500	4

- Frame total size ranges 64 to 1518 bytes
  - Preamble is not considered
  - When data field too short (<46 bytes) => '0' padding
- *Type* field values:
  - 0-45: not valid
  - 46-1500: Reserved for 802.3 + 802.2
  - 0x0800: IP v4
  - 0x0806: ARP

Note

See the list of "ethernet numbers" in http://www.iana.org/assignments/ethernet-numbers

Ethernet frame format

8	6	6	2	46-1500	4
Preamble	Target Address	Source Address	type	data	Checksum

type	data	
0x0800	IP v4 datatgram	
0x0806	ARP Request/Reply	PAD
0x0835	RARP Request/Reply	PAD
0x86dd	IP v6 datagram	
0x8137	IPX datagram	
xxx	xxxxx	

Ethernet frame evolution

- Original ethernet frame
  - Type > 1500

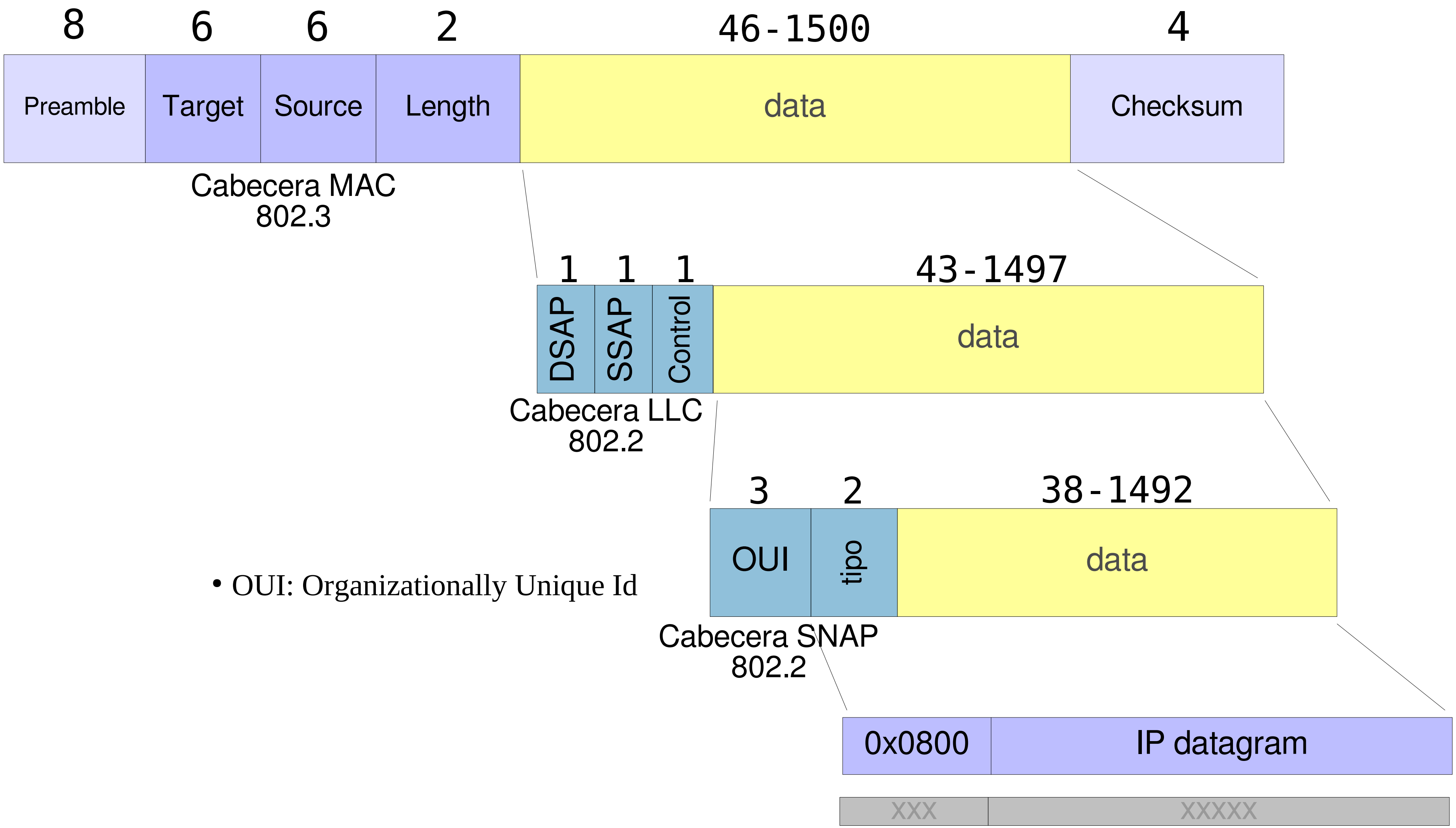
Target address	Source address	type	data
6	6	2	46-1500

- 802.3 standard
  - Extension to encapsulate 802.2 frames

Target address	Source addresss	Length	data
6	6	2	46-1500

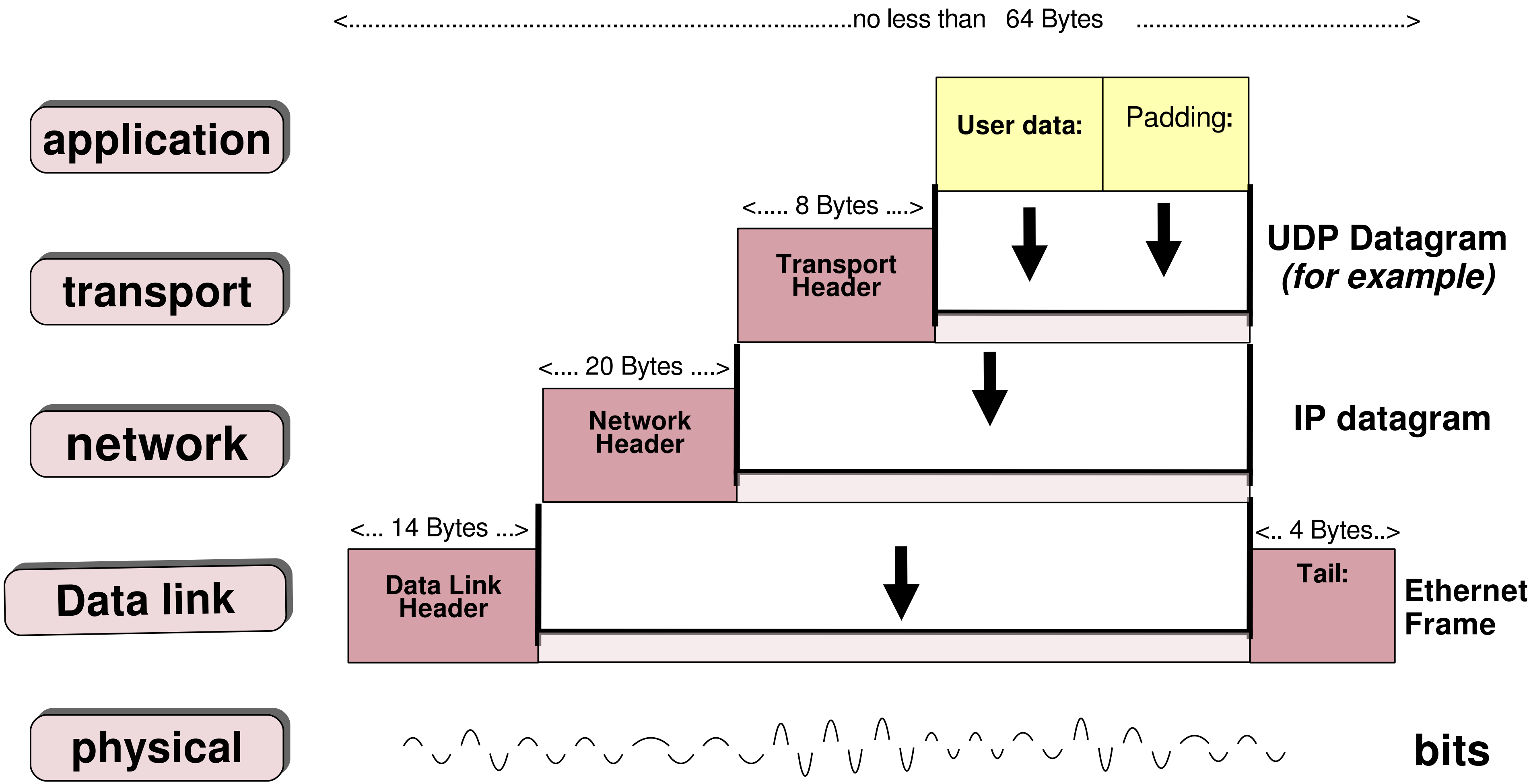
  - Length > 1500 equals to original eth. Type
  - 46 <= Length <= 1500 802.2 encapsulation

802.2 encapsulation



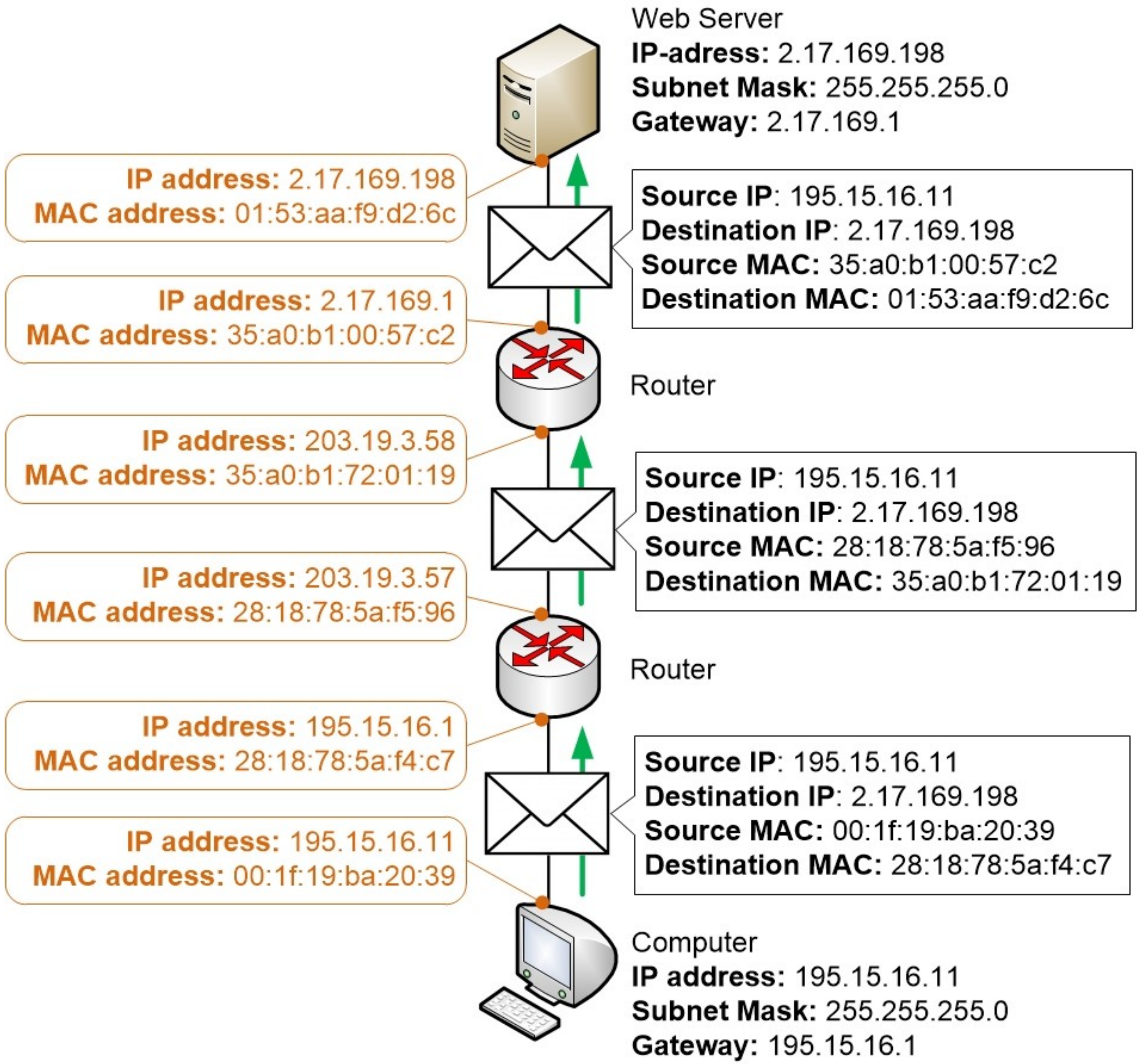
- OUI: Organizationally Unique Id

Layer encapsulation revisited



Physical vs Logical addresses

- Routing is based on IPs, but local delivery inside the LAN uses MAC addresses
- How do we know which MAC corresponds to which IP?



ARP
Address Resolution Protocol

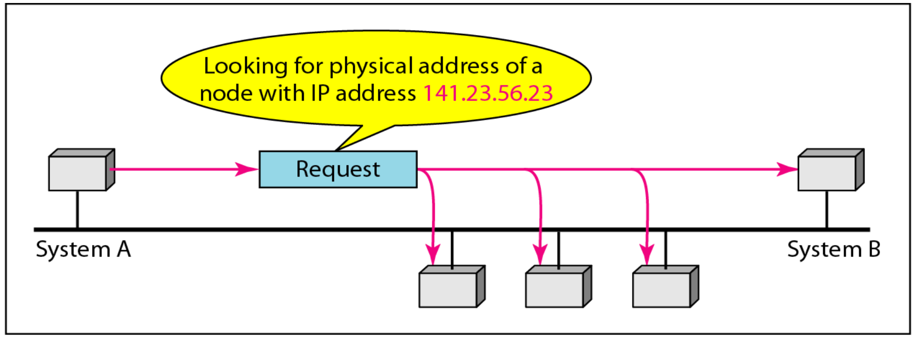
RFC 826

- ARP is a neighbor discovery protocol.
  - Can ask for the MAC address of a certain IP host
- It abstracts the physical addressing from the logical addressing, that depends on the underlying network technology.
- Each host holds an ARP cache. It contains pairs (physical address, logical address).

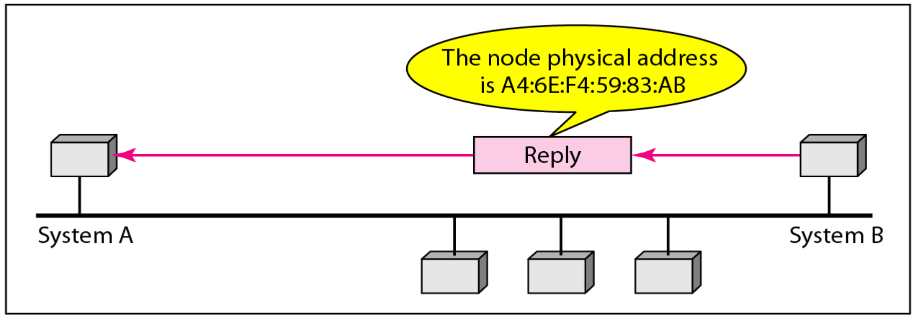
ARP
Address Resolution Protocol

When a host requires a neighbor physical address:

- It sends an **ARP request** (broadcast) including the destination host IP address.
- That host answers with an **ARP reply** (unicast) including its physical address.



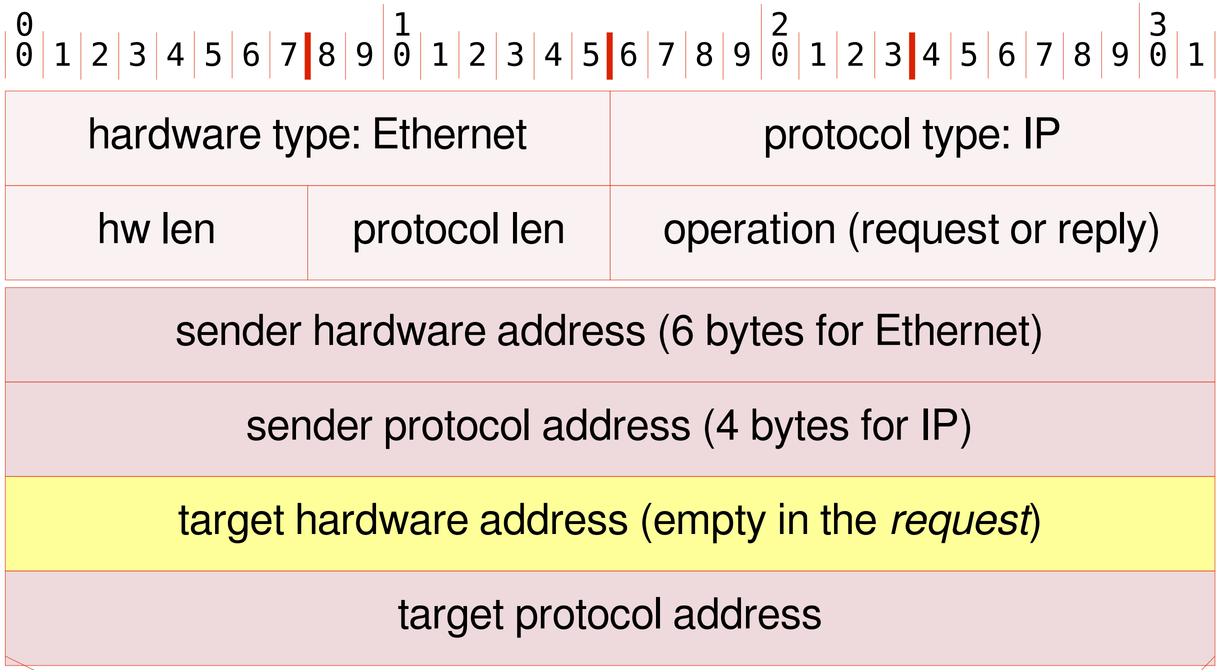
a. ARP request is broadcast



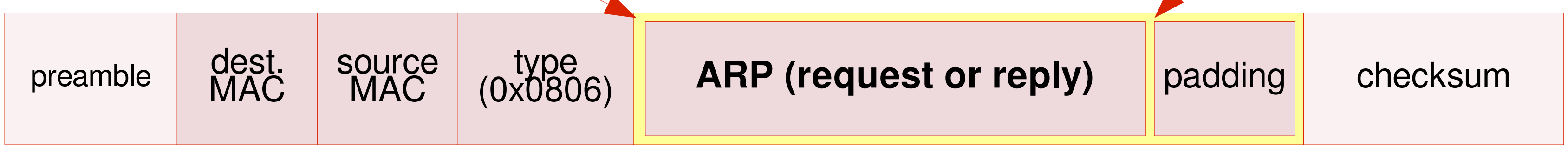
b. ARP reply is unicast

ARP
Message format

- Message format



- Ethernet encapsulation



ARP
Sample

ARP request (from A to all) →

CRC	ARP packet	tipo (0x0806)	**B1:34:56:23:AD:1E**	**FF:FF:FF:FF:FF:FF**	preamble

0x0002		0x0800	
0x06	0x04	0x0001	
B1:34:56:23:AD:1E			
161.67.38.12			
00:00:00:00:00:00			
161.67.38.95			

ARP reply (from B to A) ←

preamble	**B1:34:56:23:AD:1E**	**46:57:92:AF:FC:21**	tipo (0x0806)	ARP packet	CRC

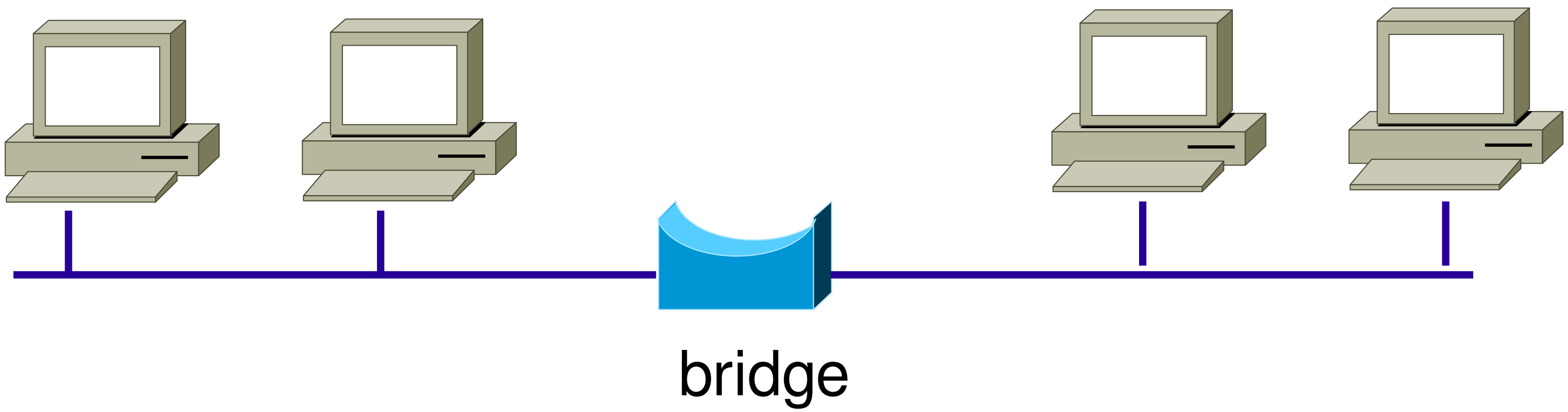
0x0002		0x0800
0x06	0x04	0x0002
46:57:92:AF:FC:21		
161.67.38.95		
B1:34:56:23:AD:1E		
161.67.38.12		

Bridging and switching

- The original ethernet also has evolved to provide higher data rates and eficiency:
  - Using bridges in a first step to deal with collision domains
  - Extending the idea to switches to turn the shared medium into a point-to-point physical connection

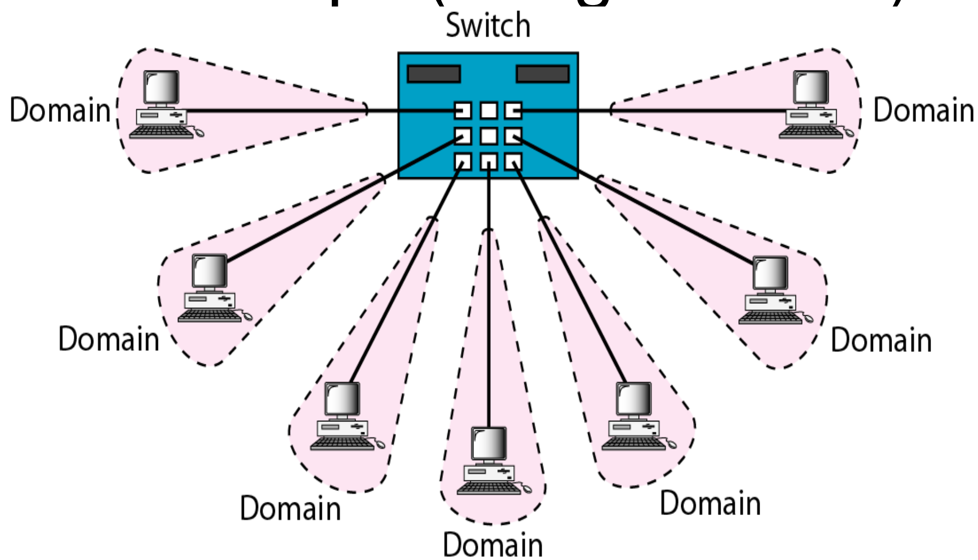
Bridges

- Network subdivision
  - But not bandwidth division. It’s kept in each subnetwork
- Separation of collision domains
- Bridges forward frames belonging to other collision domains
- MAC addresses not changed



Bridges

- Be a 10 Mbps network with 6 hosts and a single domain. The bandwidth for each host will be 10/6 Mbps < 2 Mbps (due to collisions)
- Be the same network with a bridge with 2 ports:
  - 2 collision domains with 3 hosts each
  - Bandwith = 10 / (3+1) Mbps = 2.5 Mbps (bridge counts)
- The switch provides 1 host per collision domain
  - Also called level 2 switch (because link layer is level 2)
  - Maximum bandwith (10 Mbps)

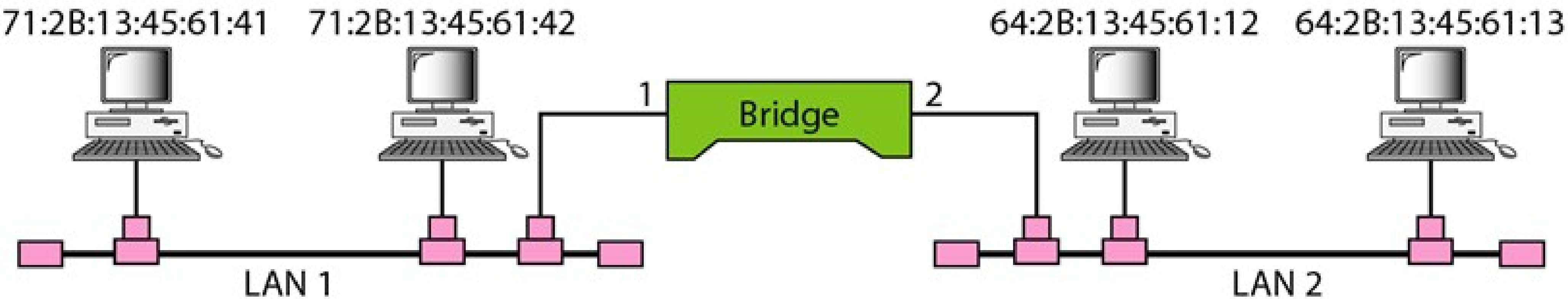


Bridges: domain link

- How does the bridge when messages should be forwarded to other domains?
- It includes a table linking MACs and output ports

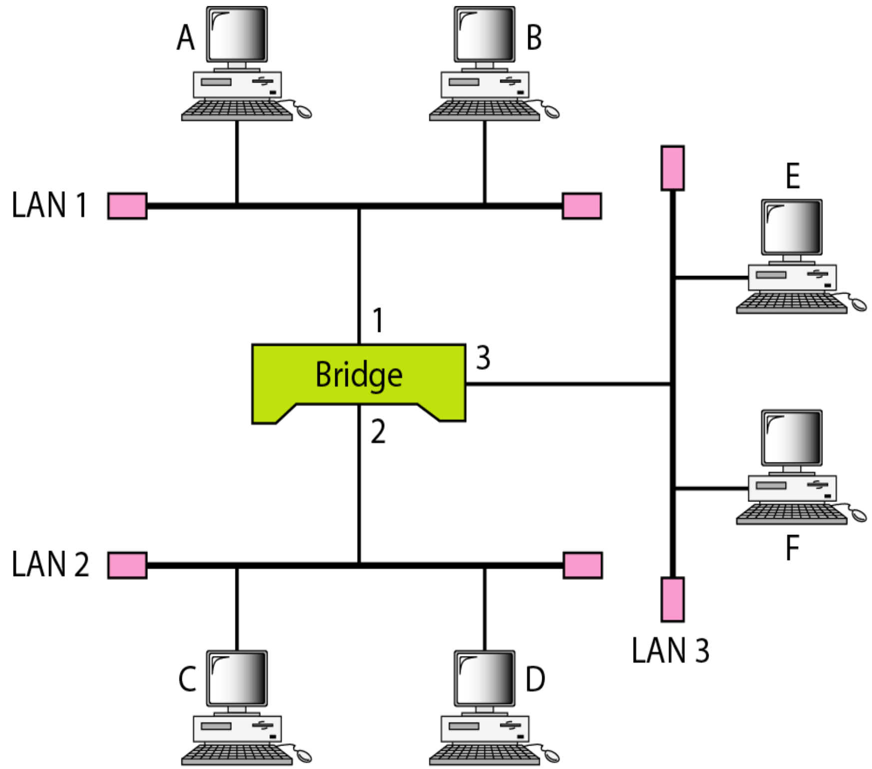
Address	Port
71:2B:13:45:61:41	1
71:2B:13:45:61:42	1
64:2B:13:45:61:12	2
64:2B:13:45:61:13	2

Bridge Table



Bridges: learning algorithm

- Bridges learn the location of the hosts
- When turned on the routing table is empty
- When a frame arrives
  - Writes down the source address in the receiving interface
  - If the interface is known the frame is redirected. Otherwise flood through all ports
- Entries in the table are flushed if not used after a certain time



Address	Port

a. Original

Address	Port
A	1

b. After A sends a frame to D

Address	Port
A	1
E	3

c. After E sends a frame to A

Address	Port
A	1
E	3
B	1

d. After B sends a frame to C

Switches

- Multiport bridges
- Full-duplex communication though dedicated pairs
- No need for CSMA/CD