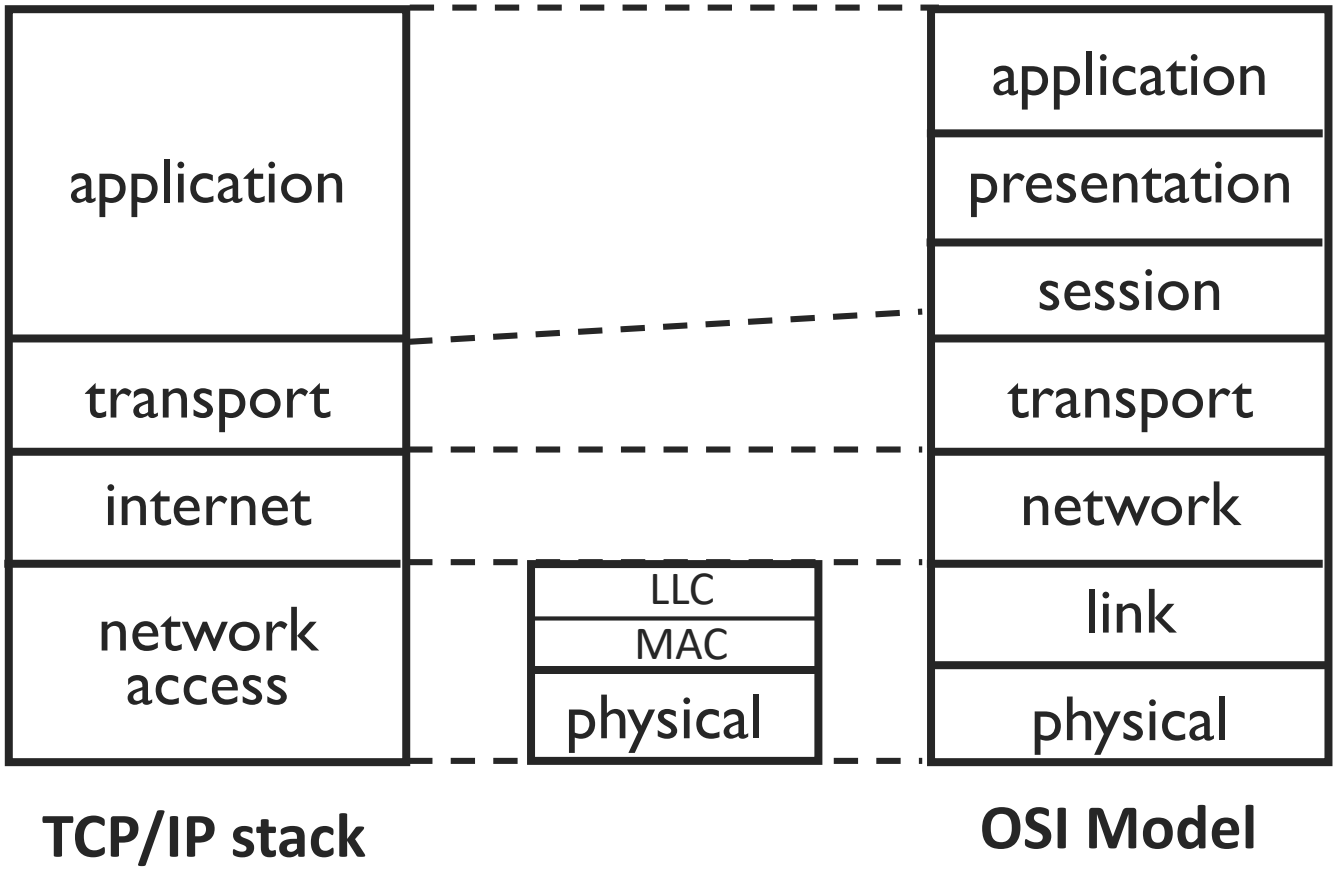
Institute of Telecommunications
Warsaw University of Technology
2019

Internet Technologies and Standards

- Piotr Gajowniczek
- Andrzej Bąk

TCP/IP Stack: Network Access Layer (Link/Phy)

TCP/IP Stack vs OSI Reference Model



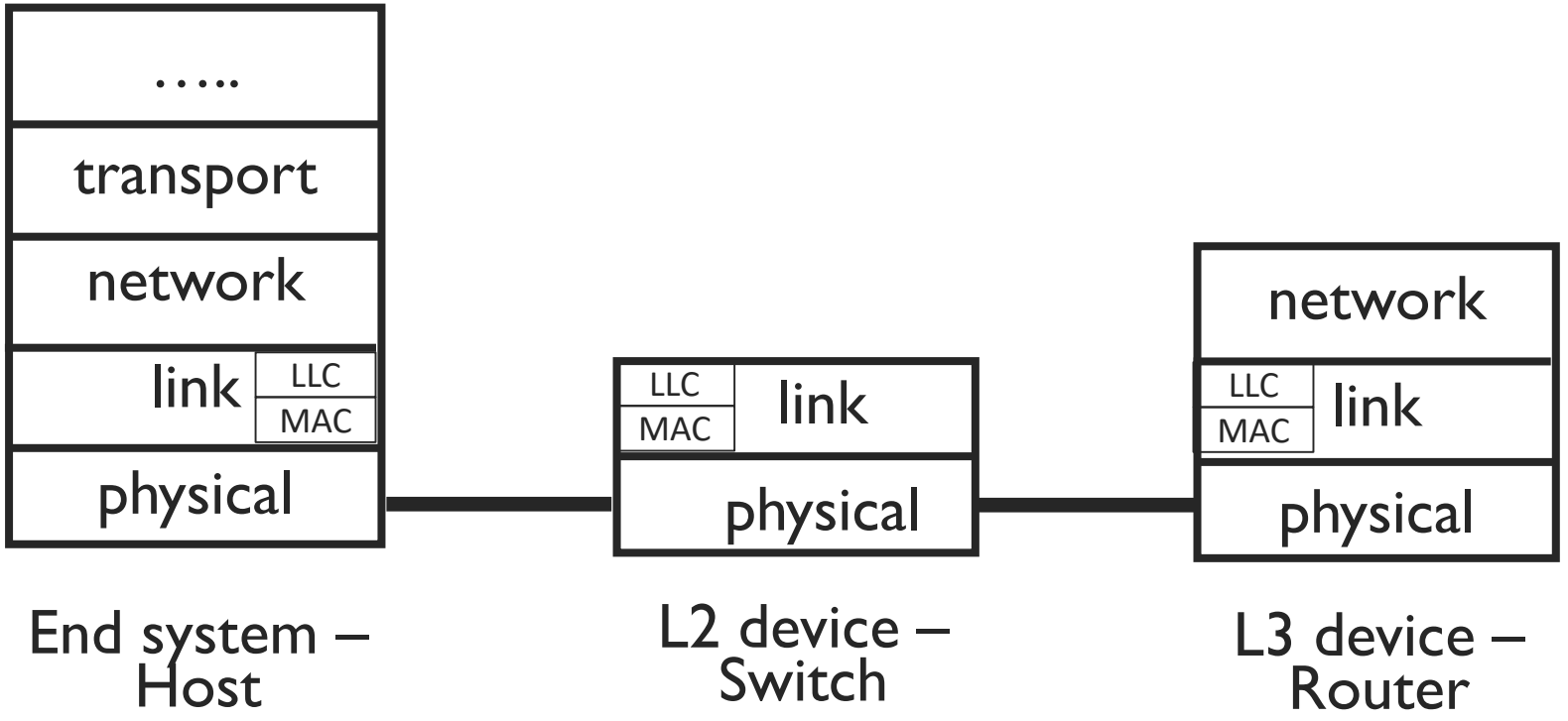
Network Access Layer Services

- *Framing (encapsulation)*
  - encapsulate datagrams into frames, adding header, trailer
  - multiplexing/demultiplexing of higher level protocol packets
- *Establishing connection between adjacent nodes (Data-link protocol handshaking)*
  - establishment, configuration, and testing the data-link connection
  - negotiation of protocol parameters: half-duplex/full-duplex, speed etc.
- *Link access (Medium Access Control)*
  - channel access if shared medium (e.g. half duplex transmission on point-to-point links or bus)
- *Flow control*
- *Error Detection/ARQ (retransmission)*
  - frame check sum calculation
  - Stop&Wait, Go back-N, Selective Repeat
- *Authentication*
- *Data compression*

TCP/IP Network Access/Link Layer Protocols

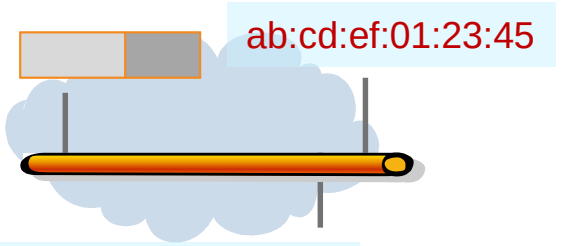
- *Types of IP "links" :*
  - point-to-point
    - PPP
    - PPPoE
    - PPPoA
    - POS (Packet over SDH)
  - Non-broadcast multiple access network (NBMA)
    - ATM
    - Frame Relay
    - X.25
  - Broadcast networks (shared medium)
    - Ethernet 802.3 (IEEE and version 2)
    - 802.11 wireless LAN
    - 802.5 Token Ring
    - 802.16 WiMax

Network Devices in TCP/IP Networks

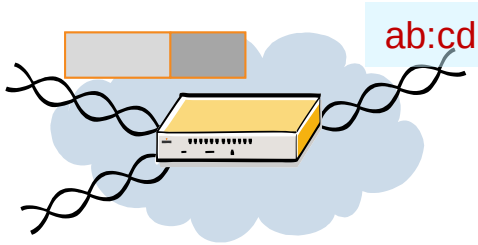


Ethernet Standard

- *Ethernet Genesis (introduced in 1983)*
  - Problem – how to economically connect a limited number of computers in a limited area with high-speed links
  - Solution – IEEE 802.x standards, especially Ethernet 802.3 and WiFi 802.11
- *Capacity from 10 Mbps (old technology) up to 400 Gbps (newest standard)*
- *Media: coaxial cables, cooper (twisted pair) cables, fibres*
  - Initially shared medium over coaxial cables with bus topology
  - now star/mesh topology with point-to-point fullduplex links (twisted pair or fibres)



shared medium



L2 switch (full-duplex, MAC learning)

Ethernet in TCP/IP Stack vs OSI Model

- *Ethernet implements Network Access Layer in TCP/IP stack*
  - Data Link and Physical layer in OSI model
- *Differences in relation to the OSI model*
  - best effort data transfer (unreliable, no retransmission)
  - datagram transmission (no connection establishment)
  - no flow control mechanism
- *Ethernet is more then just „framing protocol over wire"*
  - Ethernet „networking" devices (switches) – complicated topologies - not only bus or point-to-point wires
  - node addressing: e.g. on Ethernet "MAC" addresses used in frame headers to identify source, destination
    - different from IP addresses!
  - spanning tree protocol for routing

Ethernet MAC Sublayer

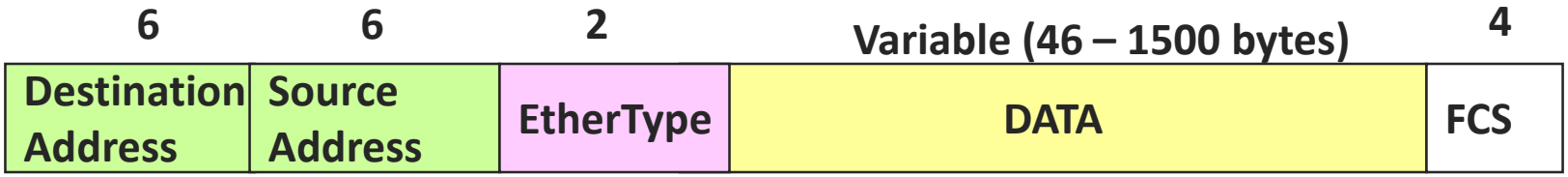
- *Medium Access Control (MAC) layer (used over shared medium – half-duplex transmission)*
  - CSMA/CD (Collision Detection)
  - CSMA/CA (Collision Avoidance), used in 802.11 (wireless LANs)
- *Source/destination station addressing by MAC address*
  - 48 bit identifier is assigned to each end station (hardcoded in the NIC card)
  - Flat address space – addresses are assign to the NIC cards during production process, no restriction or requirements is imposed on where given address can be used (in which network)

Ethernet Frame Formats

- *Version 2 (actually endorsed under 802.3 umbrella) but the name sticks…*
- *802.3/802.2 LLC Standard*
- *802.3/802.2 LLC/SNAP Standard*

- *Note: a single Ethernet segment may simultaneously carry all three of these!!!*
  - Version 2: ARP, IPv4 bearer traffic
  - 802.2 LLC: Spanning Tree protocol (BDPUs), NetBIOS, SNA, CDP/VTP
  - 802.2 LLC/SNAP: Many Cisco control protocols use this encapsulation, AppleTalkv2

Ethernet II Frame (aka DIX frame)



- *FCS: 4 bytes frame check sequence (CRC).*

Note: Frames are shown without preamble and IFG.

- *DATA: User data – 46 to 1500 bytes:*
  - *64 bytes minimum frame (requirements of collision detection algorithm)*
  - *18 bytes DA, SA, ET, FCS*
  - *46 bytes minimum data field*

- *EtherType (what's in that user data?) > 1536 in case of Ethernet II:*
  - *0x0800 is IP version 4 (= 2048 in decimal)*
  - *0x86dd is IP version 6*
  - *0x0806 is ARP (Address Resolution Protocol)*

- *Destination & Source Addresses are MAC addresses: 6 bytes each*

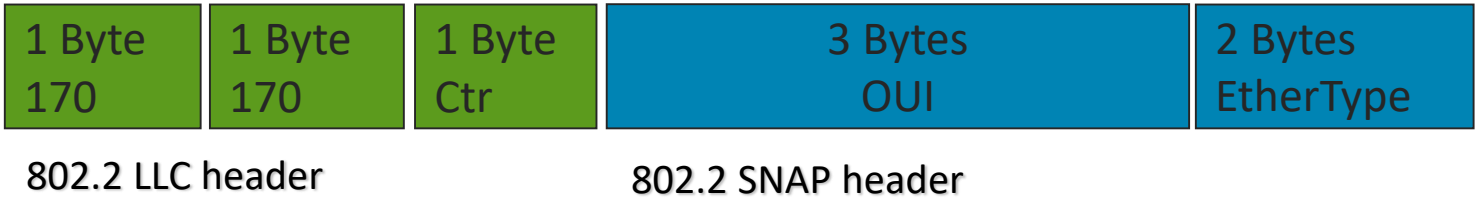
Ethernet IEEE 802.3 LLC&SNAP Frame

- *EtherType filed interpreted as frame length*
- *IEEE 802.2 LLC (link sublayer)*
  - Multiplexing/demultiplexing
  - Flow control
  - Retransmission
  - Connection oriented or connectionless mode
  - Unicast, multicast and broadcast transmission
  - LLC types
    - LLC1 – connectionless without ACK (no retransmission), suport for unicast, multicast and broadcast – can be used by TCP/IP with DSAP=SSAP=6 (IP protocol)
    - LLC2 – connection oriented with for flow control, frames sequencing and error recovery – used by NetBIOS or SNA
    - LLC3 – connectionless with ACK (practically not used)

802.2 LLC header

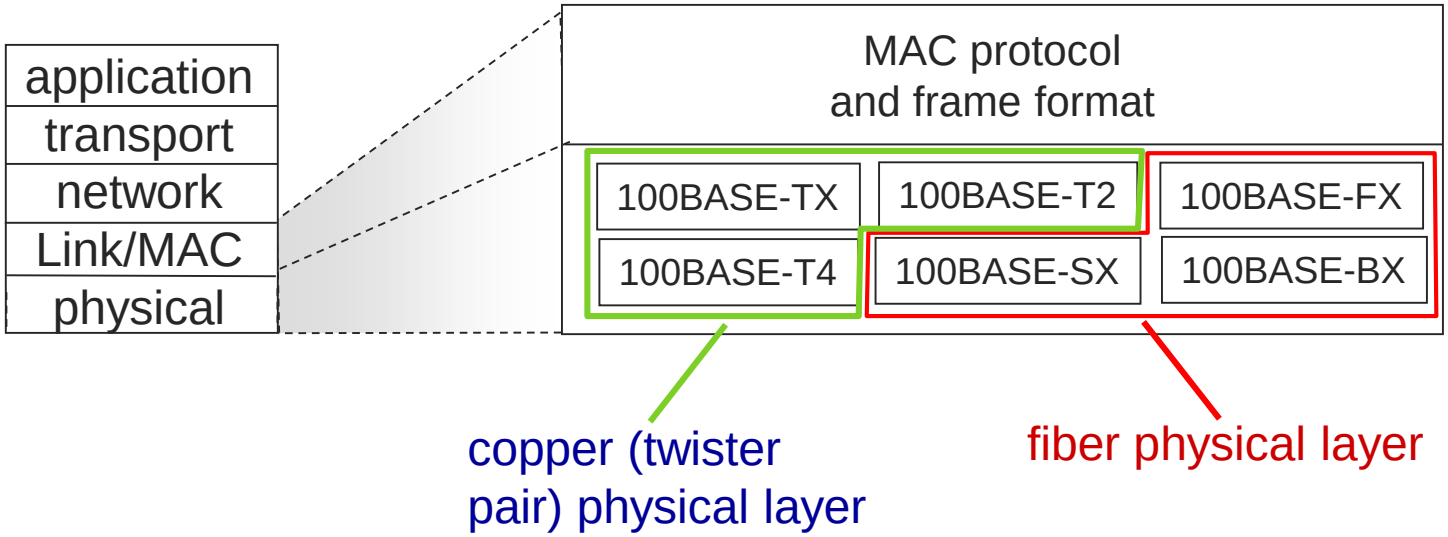
Ethernet IEEE 802.3 LLC&SNAP Frame

- *IEEE 802.2 LLC/SNAP (link sublayer)*
  - Extension of 802.2 LLC (DSAP=SSAP=170)
    - Extends the protocol identification as compared to LLC (1 byte vs 2 bytes identifier)
      - EtherType – identification of higher layer protocol
    - Allows independent organizations to define own EtherType values
      - OUI=0 means EtherType values like in Ethernet II
      - Otherwise OUI contains organizationn identifier that defines the EtherType values
  - Can be used by TCP/IP (e.g. over FDDI)



802.3 Ethernet Physical Layers

- *many different Ethernet standards*
  - common MAC protocol and frame format
  - different speeds: 2 Mbps, 10 Mbps, 100 Mbps, 1Gbps, 10Gbps, 100Gbps, 400Gbps
  - different physical layer media: fiber, cable
  - Baseband transmission (no modulation)



VLANs

- *Switch ports can be grouped (by switch management software) so that single physical switch ……*

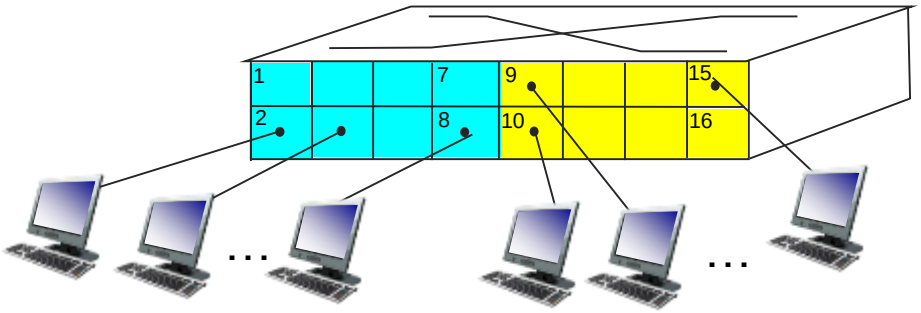
Virtual LAN

switch(es) supporting VLAN capabilities can be configured to define multiple ***virtual*** LANs over single physical infrastructure

Each VLAN works as independent Ethernet network

VLANs are identified by unique VLAN ID assigned to Ethernet frames

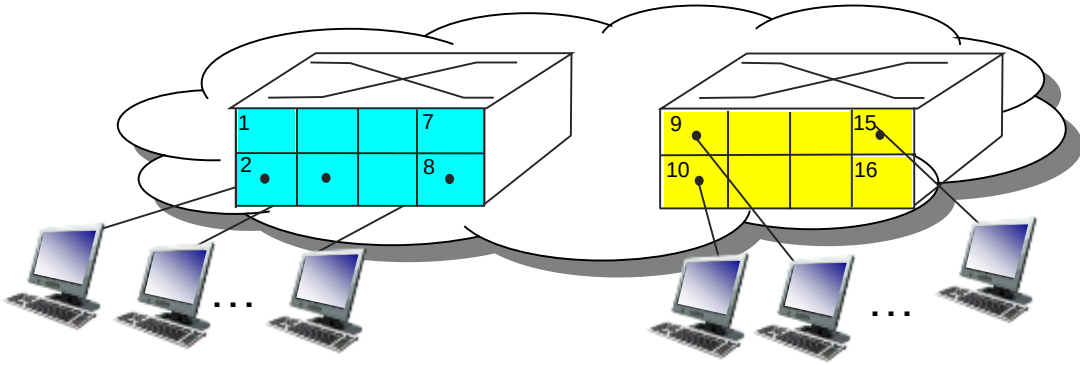
To send traffic from one VLAN to another we need router



Electrical Engineering
(VLAN ports 1-8)

Computer Science
(VLAN ports 9-15)

… operates as *multiple* virtual switches



Electrical Engineering
(VLAN ports 1-8)

Computer Science
(VLAN ports 9-16)

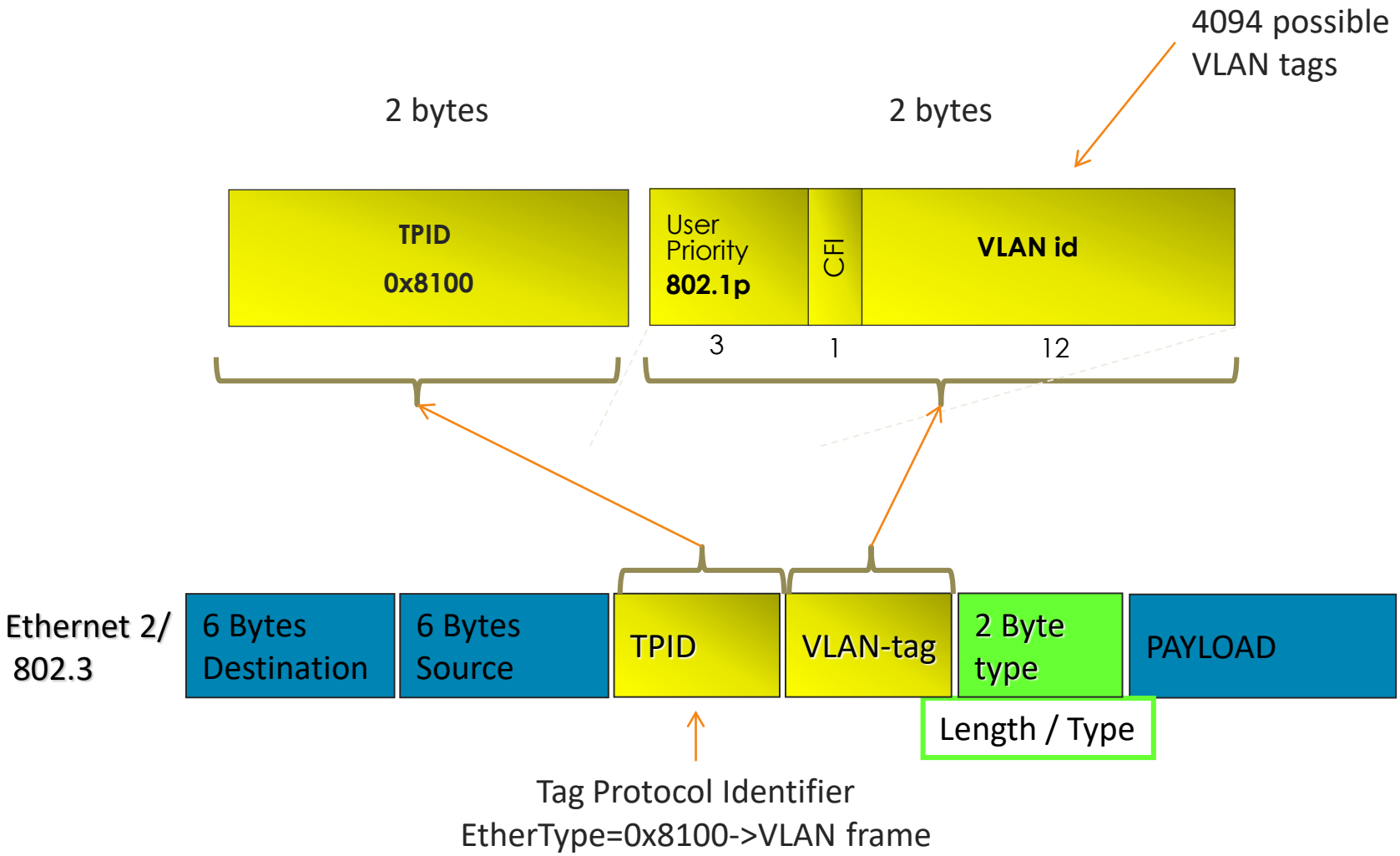
VLAN Tagged/Untagged Ports

- *tagged or trunk VLAN ports*
- 802.1Q framing
- *tagged port is used by multiple VLANs*

- *untagged VLAN ports*
- standard Ethernet framing
- „whole" port is assigned to one VLAN

- untagged ports not assigned to any VLAN are considered to belong to default VLAN

VLAN 802.1Q Frame Format



MAC Address vs. IP Address

- *IP address:*
  - network-layer address for interface
  - used for layer 3 (network layer) forwarding
  - 32 bit address, software settable
- *MAC (or LAN or physical or Ethernet) address:*
  - function: used 'locally" to get frame from one interface to another physically-connected interface (same network, in IP-addressing sense)
  - 48 bit MAC address (for most LANs) burned in NIC ROM, also sometimes software settable
  - e.g.: 1A-2F-BB-76-09-AD

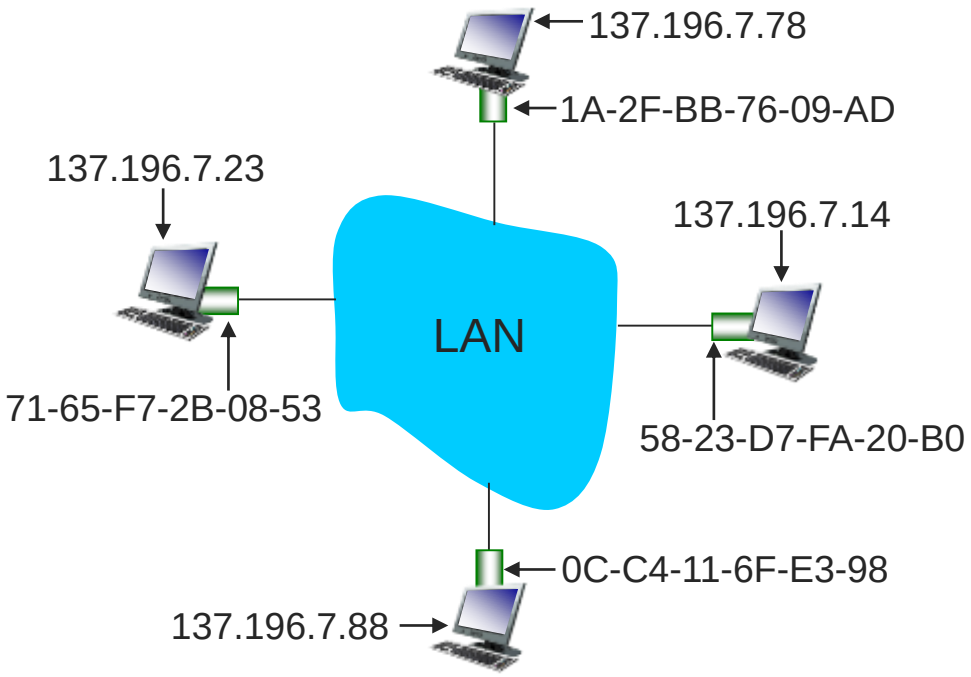
hexadecimal (base 16) notation
(each "number" represents 4 bits)

MAC Address vs. IP Address (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 hierarchical address not portable*
  - address depends on IP subnet to which node is attached

Address Resolution Protocol (ARP)

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



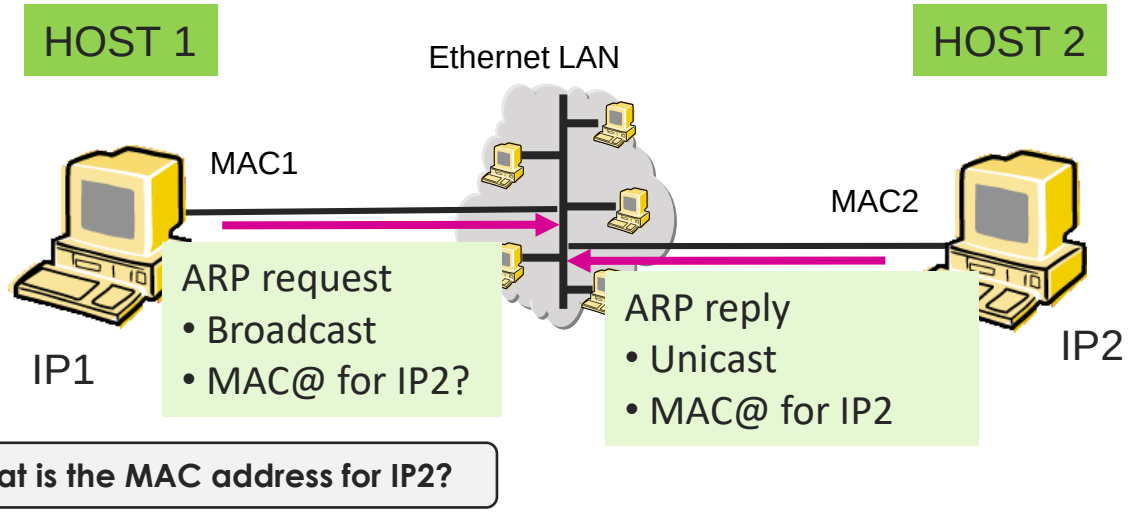
- *ARP table: each IP node (host, router) on LAN has table*
  - IP/MAC address mappings for some LAN nodes:

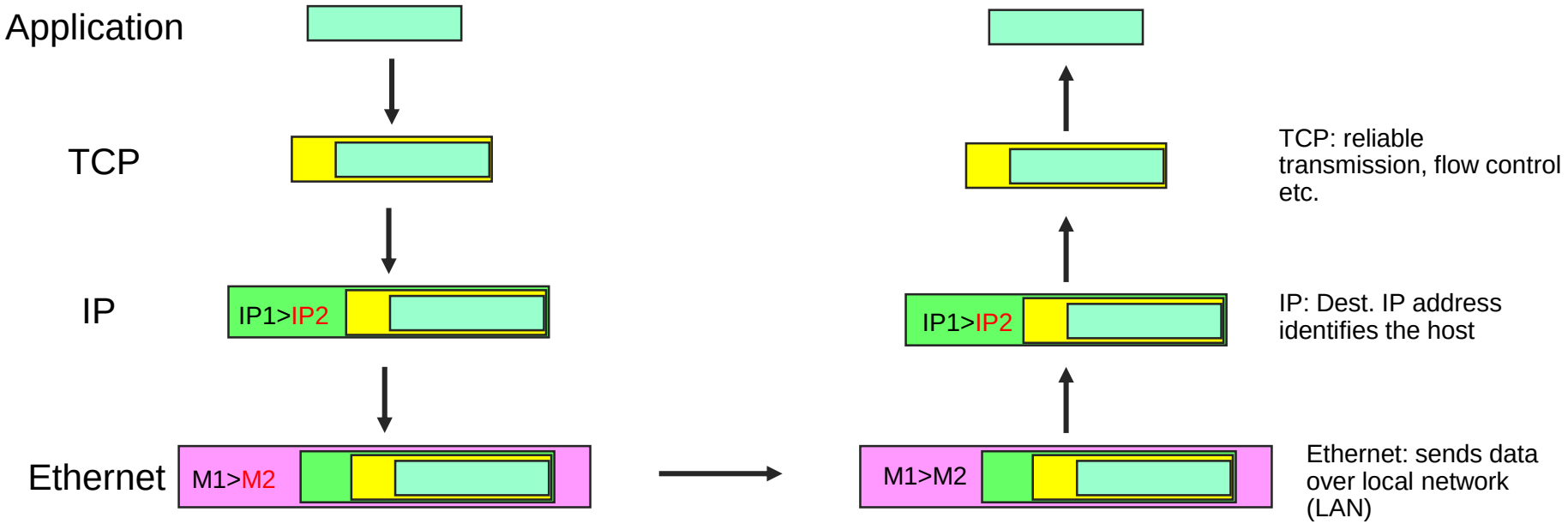
  < IP address; MAC address; TTL>
  - TTL (Time To Live): time after which address mapping will be forgotten (typically 20 min)

ARP protocol: same LAN

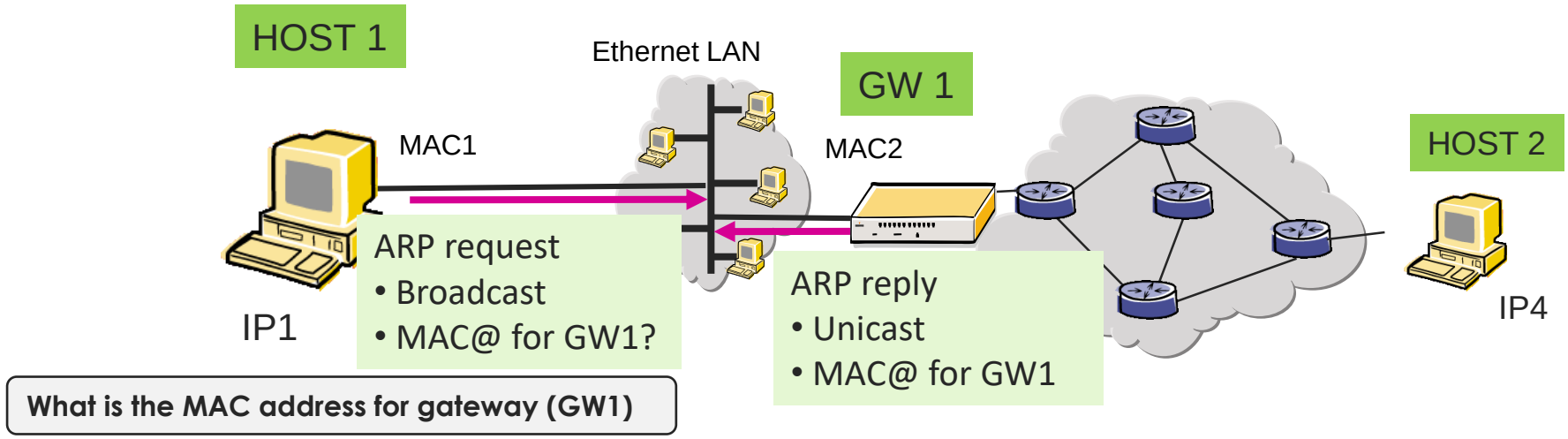
- *A wants to send datagram to B*
  - B's MAC address not in A's ARP table.
- *A broadcasts ARP query packet, containing B's IP address*
  - dest MAC address = FF-FF-FF-FF-FF-FF
  - all nodes on LAN receive ARP query
- *B receives ARP packet, replies to A with its (B's) MAC address*
  - frame sent to A's MAC address (unicast)
- *A caches (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

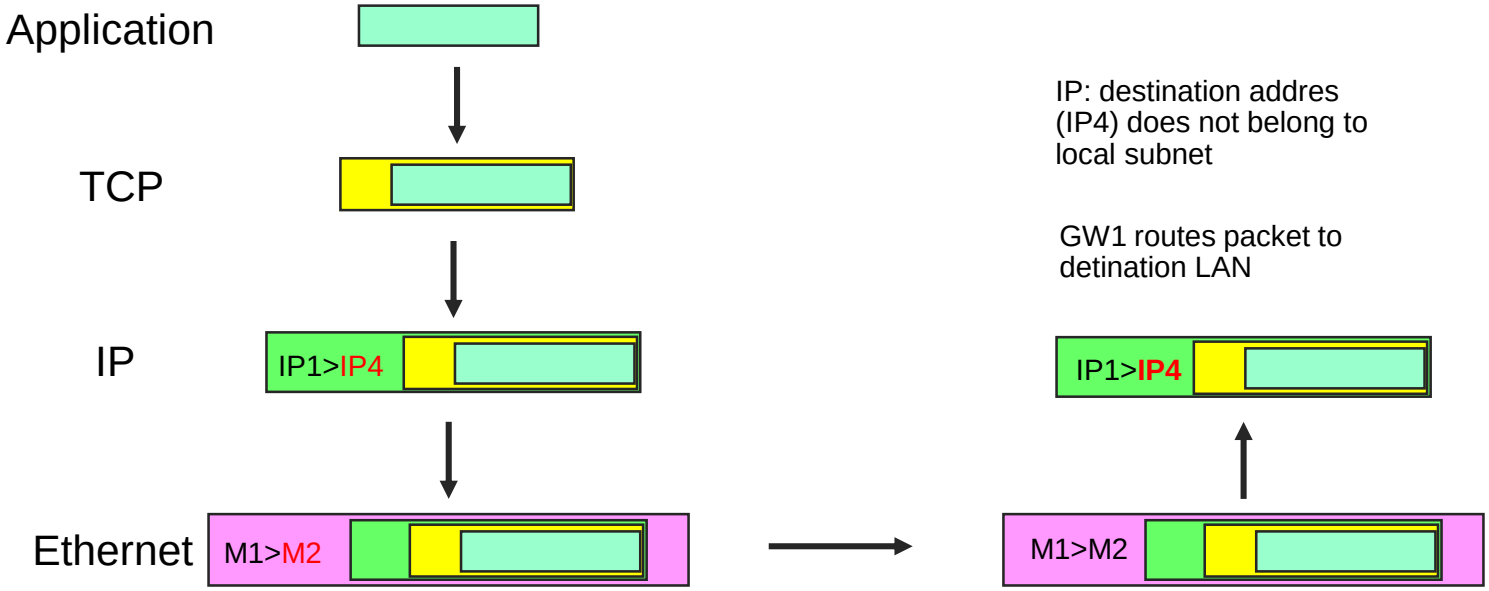
ARP protocol: same LAN





addressing: routing to another LAN





addressing: routing to another LAN