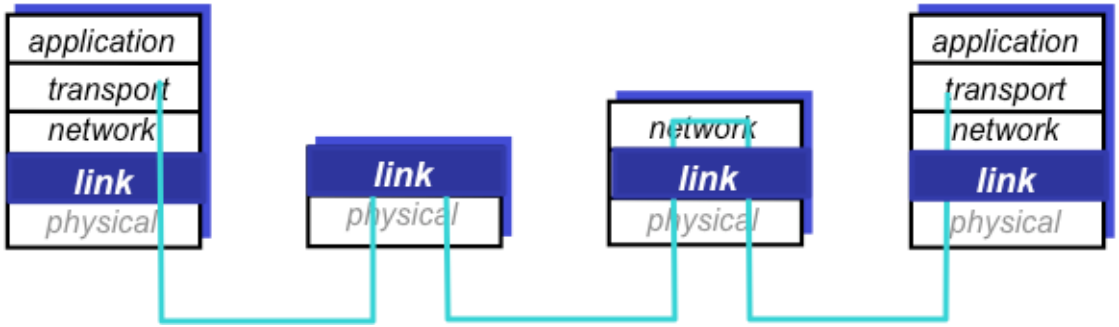
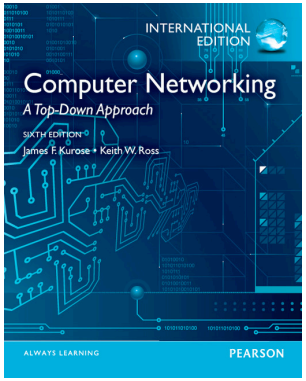
Link layer: Links, access networks and LANs

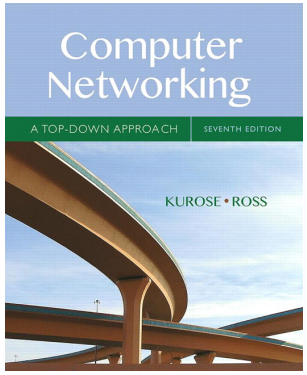
Chapter 5 (6ed) / 6 (7ed)

Kjersti Moldeklev, Prof II
Information Security and Communication Technology

kjersti.moldeklev@ntnu.no







The *content* of some of these slides are based on slides available from the web-site of the book by J.F Kurose and K.W. Ross.

Link layer and Local Area Network (LAN) February 23 – 24, 2016

7-8	See "it's learning" for when assistants are present.	Programming Lab 1: *HTTP Web Server* Programming Lab 2: *UDP Pinger* Programming Lab 3: *SMTP Mail client*	P15 - Rall	Assistants/ Ida/Bank	One must deliver and pass at least 2 of the 3 programming labs.
8	Wednesday 18:15 – 19:00 Thursday 12:15 – 14:00 Thursday 14:15 – 15:00	"Help Lecture" for project Link Layer & LAN Theory Assignment 4: *Network Layer* Wireshark Lab 3: *IP (optional but highly recommended!)*	F1 R1 R1	Bank/ Magnus Kjersti Assistants/ Ida/Norvald	One must fulfil the project requirement. Chapter 5 One must deliver and pass at least 5 of the 8 theory assignments.
8	Friday 09:15 – 11:00	Link Layer & LAN (cont)	R1	Kjersti	Chapter 5
8	**Sunday 24:00**	**Deadline** for **ALL** programming labs - Delivery in "It's learning" **NB! NB! NB!**			

NTNU

Chapter 5

The data link layer

Goals

- understand principles behind data link layer services
  - **error detection**, **error correction**
  - sharing a broadcast channel: **multiple access**
  - **link layer addressing**
  - reliable data transfer, flow control: done!

implementation of **various link layer technologies**

NTNU

Abbreviations in this slide deck...

- LAN Local Area Network
- NIC Network interface card
- CPU Central Processing Unit
- MAC Medium Access Control
- TDMA Time Division Multiple Access
- FDMA Frequency Division Multiple Access
- WLAN Wireless LAN
- WiFi Wireless Fidelity
- CSMA Carrier Sense Multiple Access
- CSMA/CD CSMA/Collision Detection
- CSMA/CA CSMA/Collision Avoidance
- EDC Error Detection and Correction
- CRC Cyclic Redundancy Check

- DOCSIS Data Over Cable Service Interface Specification
- AIS Automatic Identification System
- SAR Search And Rescue
- GNSS Global Navigation Satellite System
- VHF Very High Frequency
- ARP Address Resolution Protocol
- IP Internet Protocol
- UDP User Datagram Protocol
- Transmission Control Protocol
- DHCP Dynamic Host Configuration Protocol
- DNS Domain Name System

Roadmap

5.1 Introduction and services

5.2 Error detection and correction

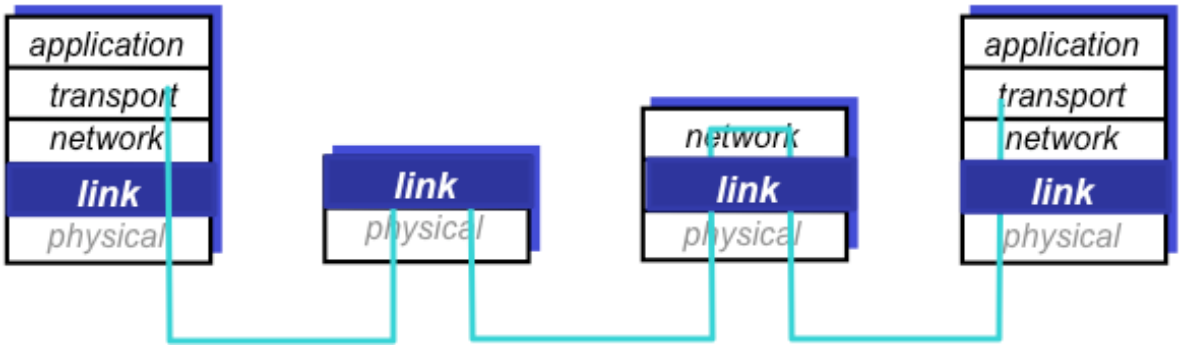
5.3 Multiple access protocols

5.4 Local Area Networks
- addressing, ARP
- Ethernet
- switches
- ~~Virtual LANS (VLANs)~~

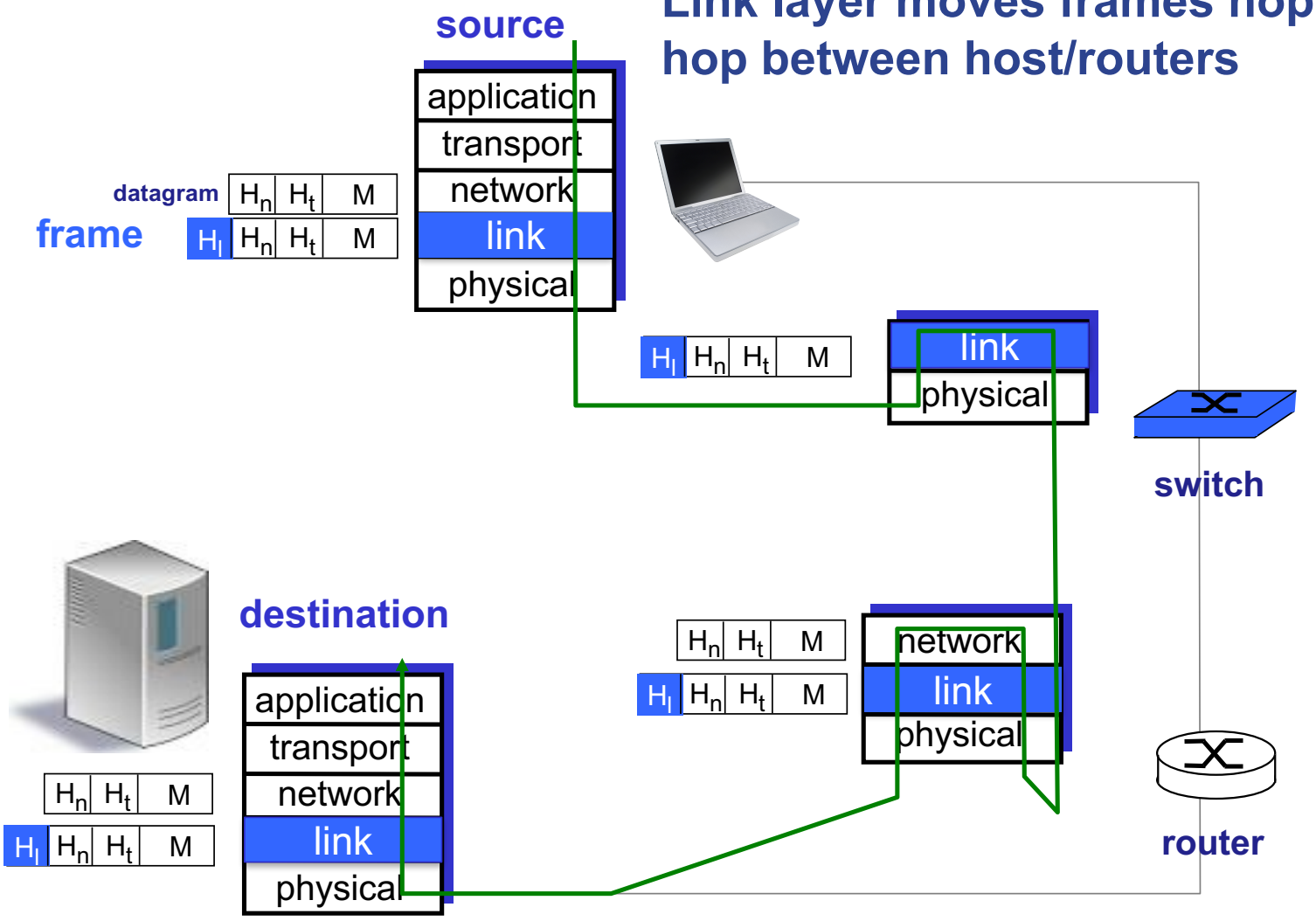
~~5.5 Link virtualization~~

~~5.6 Data center networking~~

5.7 A day in the life of a web request

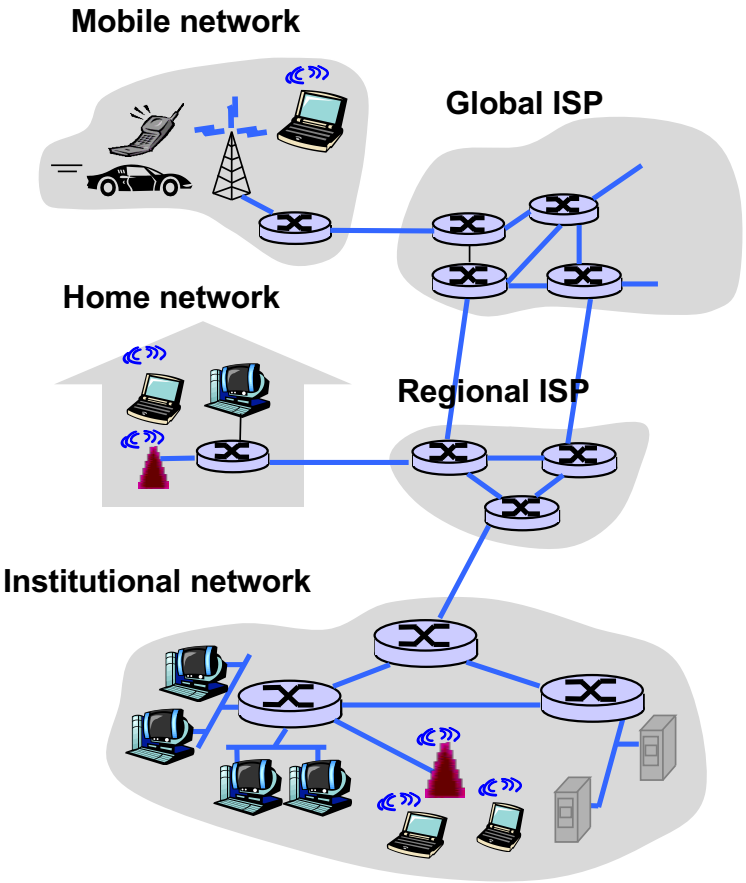


Link layer moves frames hop by hop between host/routers



Link layer directly connects host, routers and switches

- Communication channels that connect adjacent nodes along the communication path are **links**
  - wired links
  - wireless links
  - LANs
- Layer-2 **frame** encapsulates typically an IP datagram

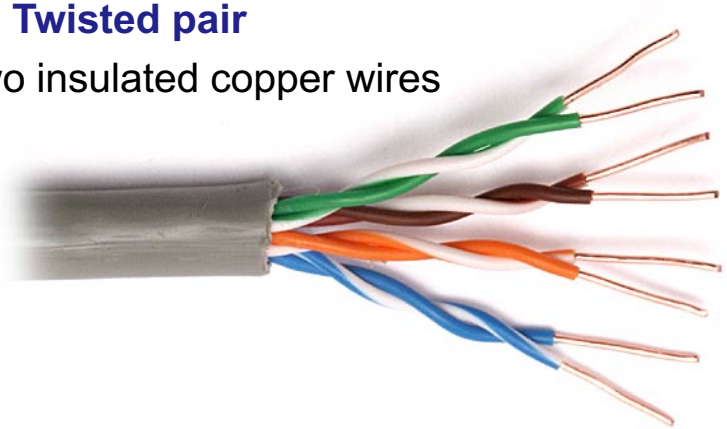


Data link layer has the responsibility of transferring datagrams over a link from one node to the adjacent node

Link layer transmits frames over a physical medium

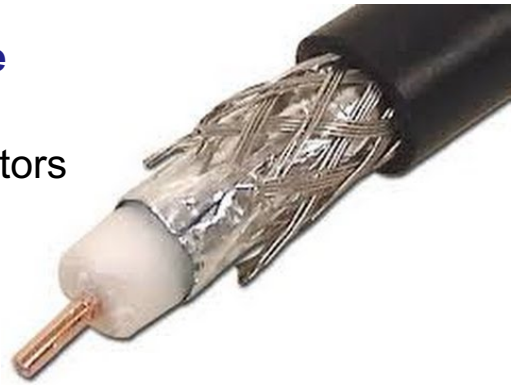
1. Twisted pair

- two insulated copper wires



2. Coaxial cable

- two concentric copper conductors



3. Fiber optic cable

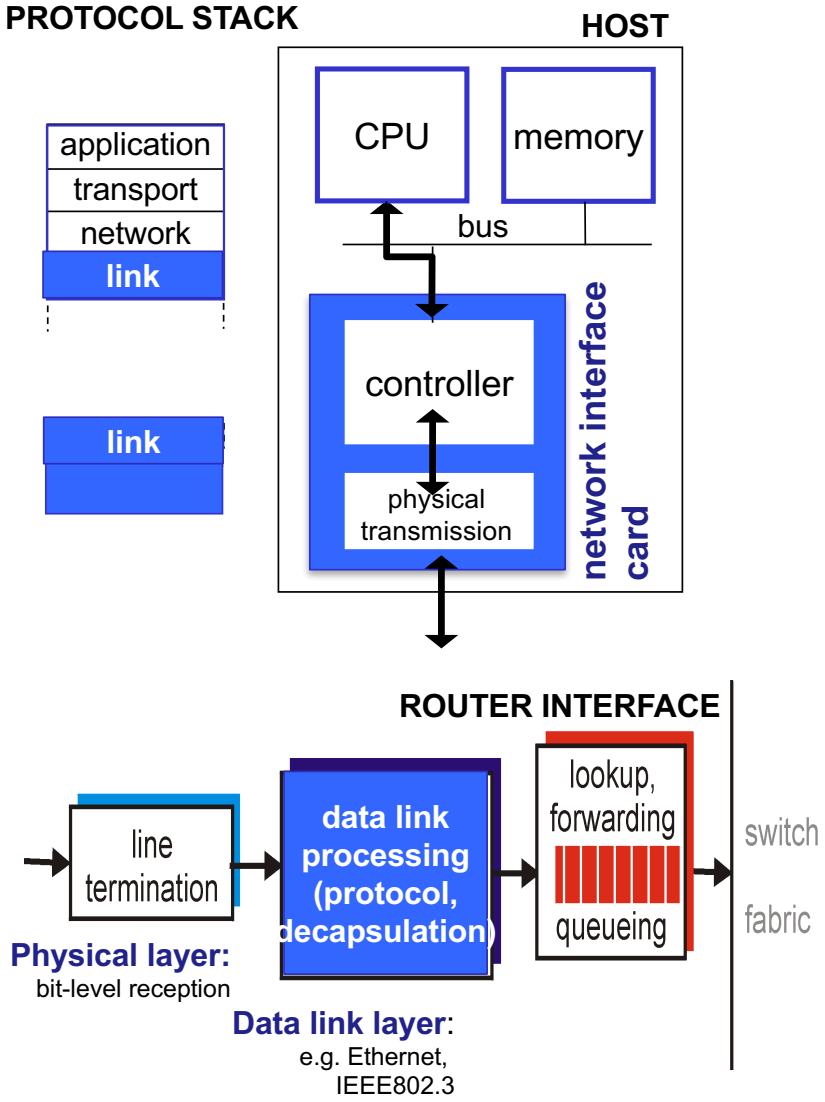
- glass fiber carrying light pulses, each pulse a bit
- high-speed operation
- low error rate

4. Radio link

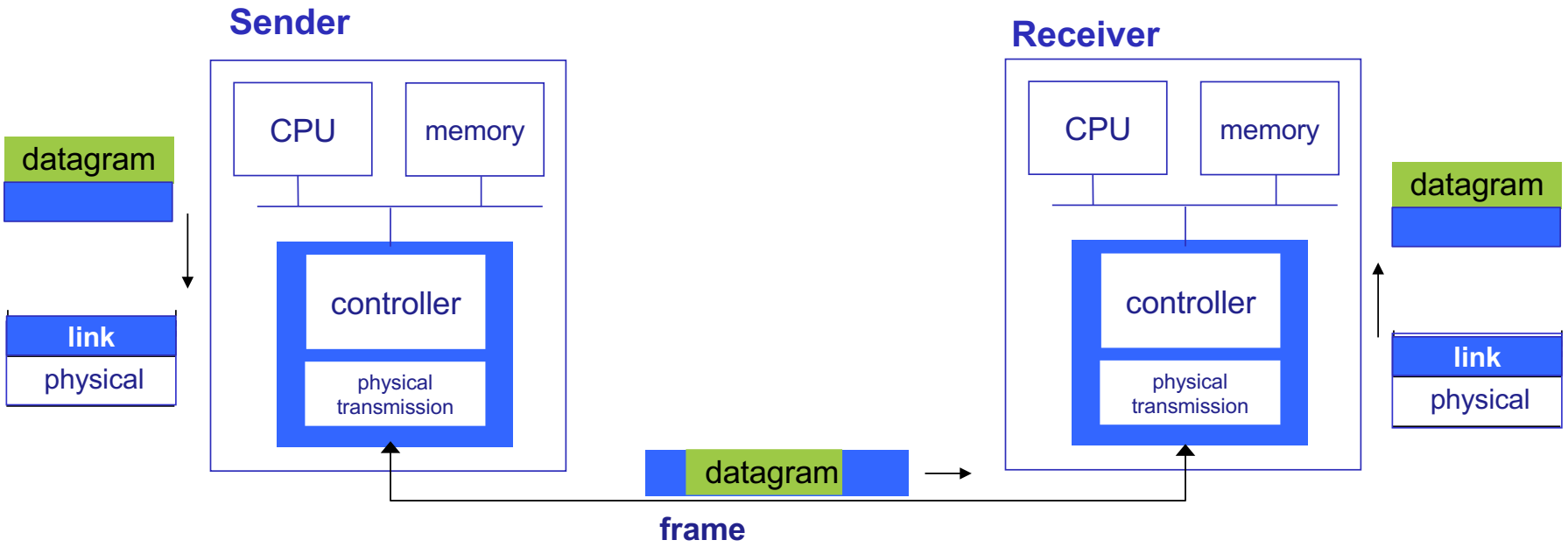
- terrestrial microwave
- wireless LAN
- wide-area (e.g. cellular)
- satellite

The link layer is implemented in each and every end system and network node

- **Network interface card** (NIC) implements link + physical layer
  - Ethernet card 802.3, WLAN card 802.11
- Combination of **hardware**, **software**, **firmware**



Network interfaces communicate by exchanging frames

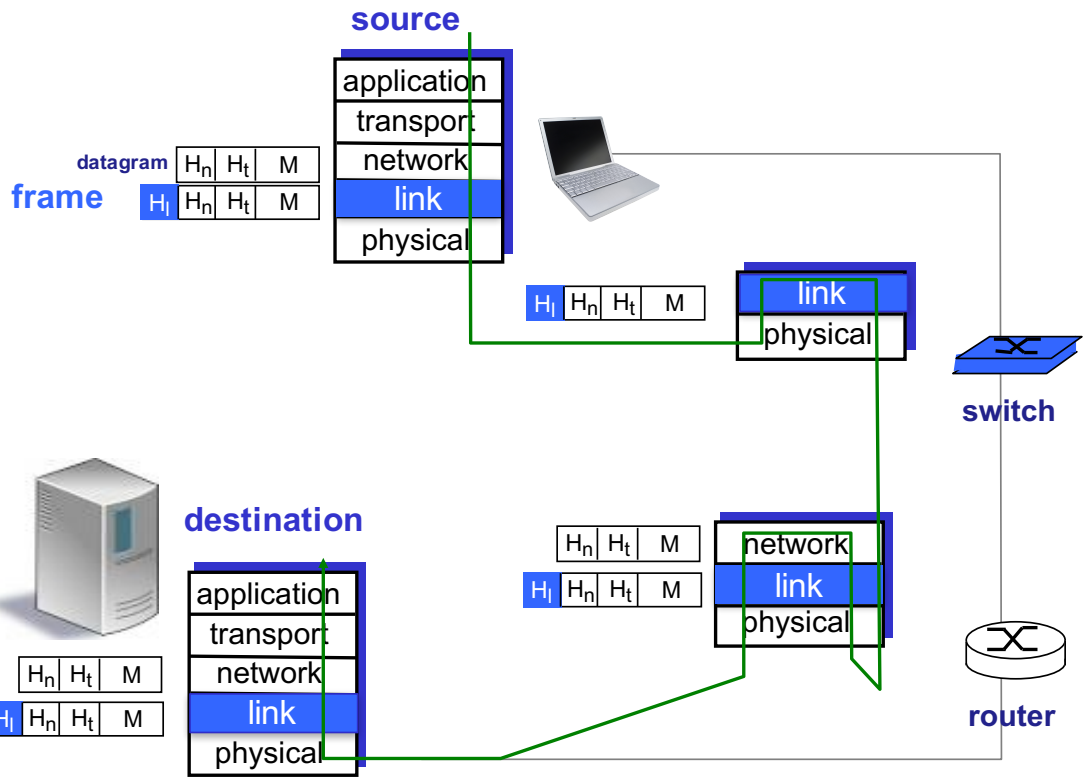


- encapsulates datagrams in frames
- adds error checking bits, reliable data transfer, flow control, etc.

- looks for errors, reliable data transfer, flow control, etc.
- extracts datagram, passes to upper layer at receiving side

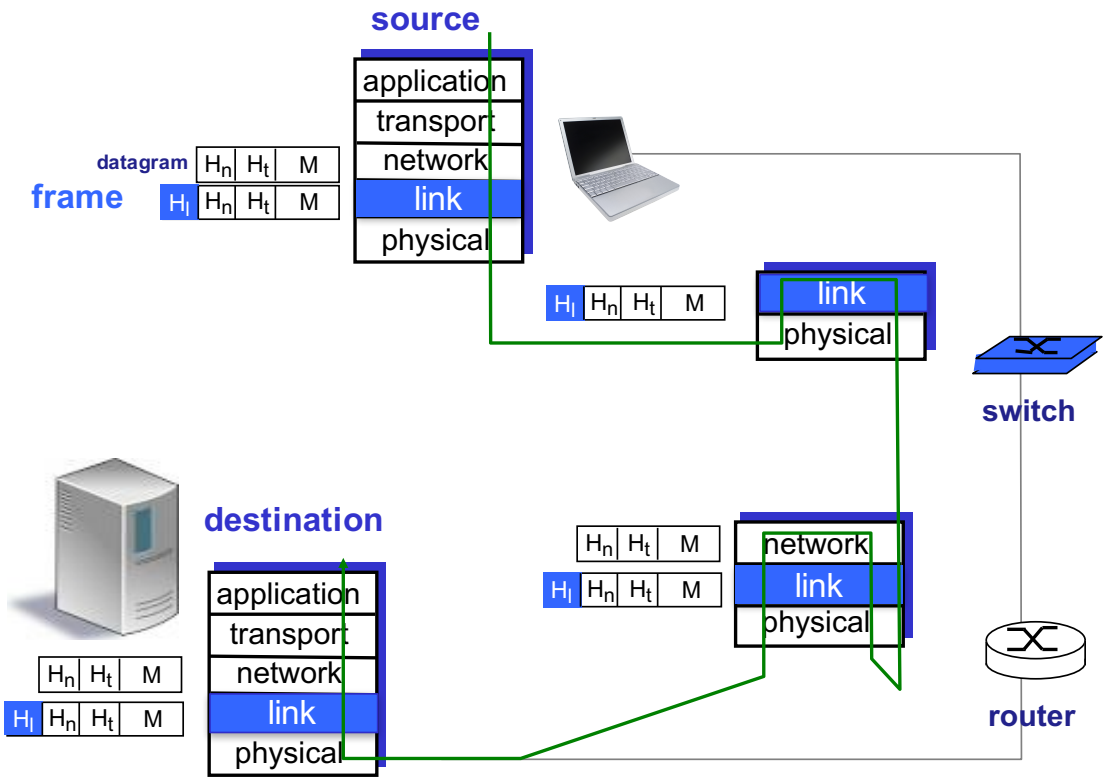
Link layer service depends on the specific link-layer network technology

- **Framing**
  - Encapsulate datagram into frame, adding header/trailer
- **Link access**
  - Channel if shared medium
  - "MAC" (**Medium Access Control**) addresses in frame headers to identify source and destination
    - different from IP address!
- **Half-** or **full-duplex**
  - With half duplex nodes at both ends of link can transmit, but not at the same time



Link layer services (more)

- **Error detection**
  - Errors caused by signal attenuation, noise
  - Receiver detects errors
    - signals sender or drops frame
- **Error correction**
  - Receiver identifies and corrects bit error(s) without retransmission
- **Reliable delivery** between adjacent nodes
  - Seldom used on low bit-error links
  - Wireless links: higher error rates
- **Flow control**
  - Pacing between adjacent sending and receiving nodes



Link layer and local area networks: Roadmap

5.1 Introduction and services

5.2 Error detection and correction

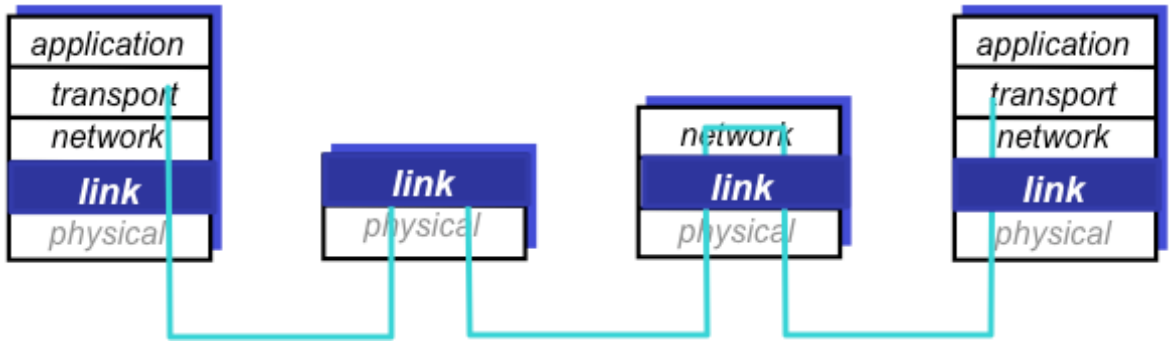
5.3 Multiple access protocols

5.4 LANs
- addressing, ARP
- Ethernet
- switches
- ~~VLANS~~

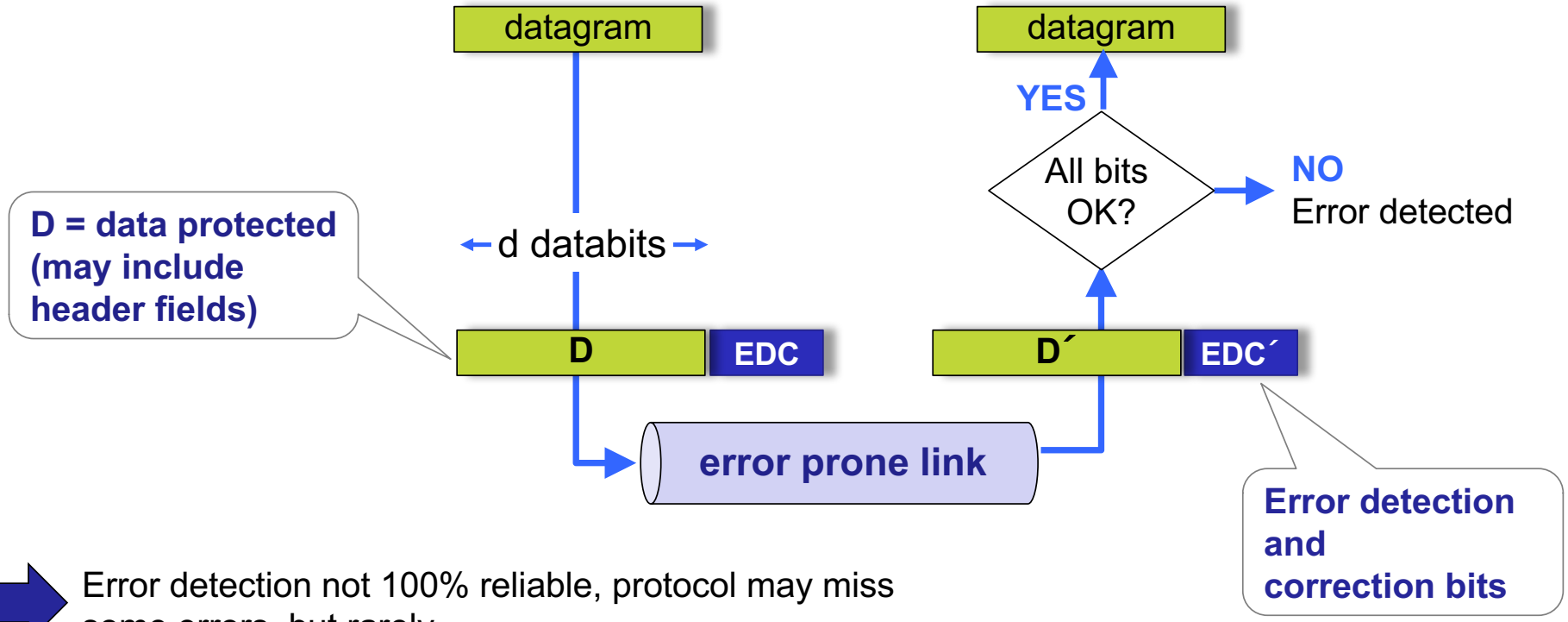
~~5.5 Link virtualization: MPLS~~

~~5.6 Data center networking~~

5.7 A day in the life of a web request



Error detection through redundancy bits

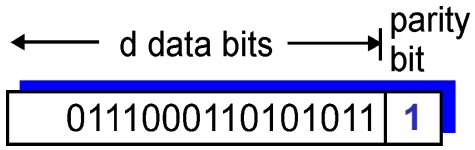


- Error detection not 100% reliable, protocol may miss some errors, but rarely
  - larger EDC field yields better detection and correction

Parity checking – specific parity bits

Single bit parity:

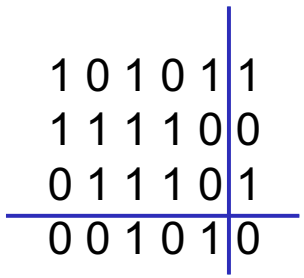
Detects single bit errors



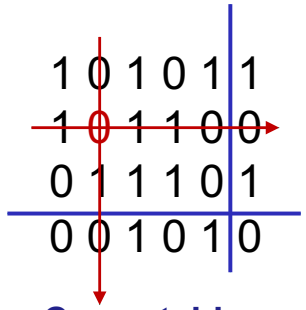
Even/Odd parity: 1-bit parity bit chosen so that number of ones in the d+1 bits is even/odd

Two dimensional bit parity:

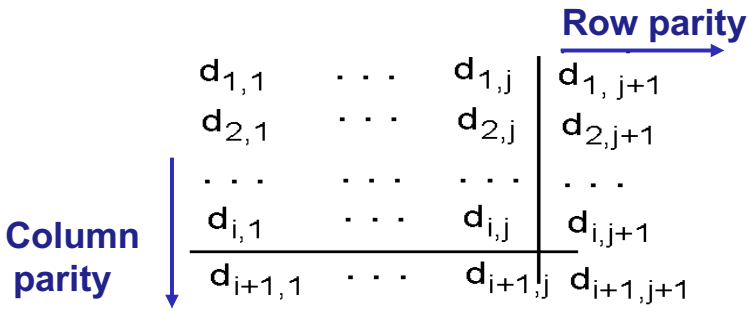
Detects and corrects single bit errors



No errors



Correctable single bit error



NTNU

Exercise: 2-dimensional even parity

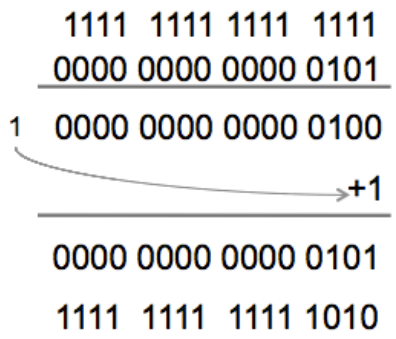
Content of a packet is the 16-bit pattern
1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 1

What is the value of a minimum checksum
using a two-dimensional even parity scheme?

Internet checksum is a simple algorithm used by IP, UDP and TCP to detect “errors” (e.g. flipped bits)

Sender

- Segment content treated as sequence of 16-bit integers
- Checksum: addition (**1’s complement sum**) of segment contents with overflow wrapped around



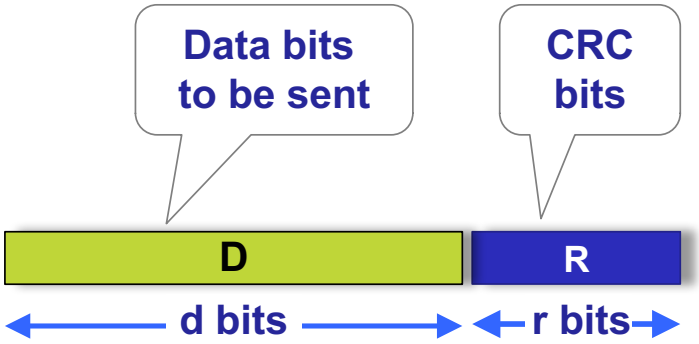
Receiver

- Compute checksum of received segment
- Check if computed checksum equals checksum field value
  - NO - error detected
  - YES - no error detected

2^{-16} chance of not detecting error
Detects all burst error <=16 bits except
0000000000000000 <-> 1111111111111111

Cyclic Redundancy Check (CRC) – more complex, more robust

- View **data bits**, **D**, as a binary number
- Choose a **generator G** = **r+1** bits
  - first and last bit = 1
- Find r **CRC-bits**, **R**, such that
  - <**D,R**> exactly divisible by **G** (modulo 2)
    - ⇔ < D,R > = $D*2^r$ XOR R = nG
    - ⇔ $D*2^r$ = nG XOR R
    - ⇔ **R = remainder[$\frac{D*2^r}{G}$]**

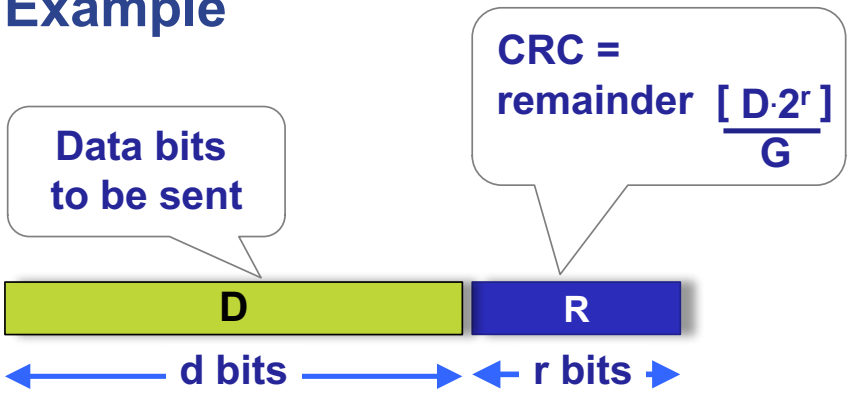


Send < D,R > = $D*2^r$ XOR R

- Receiver knows **G**, divides <**D,R**> by **G**
- If non-zero remainder: error detected!

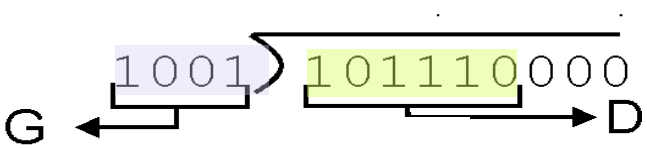
- CRC of r bits can detect
  - all burst errors < r+1 bits
  - burst error > r+1 detected with probability $1^{-0.5r}$
  - all odd number of bit errors
- Widely used in practice (Ethernet, 802.11 WiFi)

CRC Example



$D*2^r = 101110000$

n =



Data D=101110, d = 6

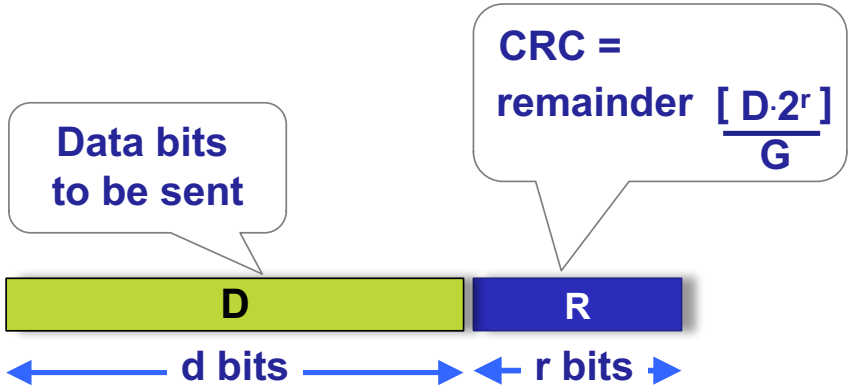
Generator G =1001 (x^3+1) , r = 3

< D,R > = 101110011

Exercise CRC-1

Data D=10101, d=5

Generator G=11 (x+1), r = 1

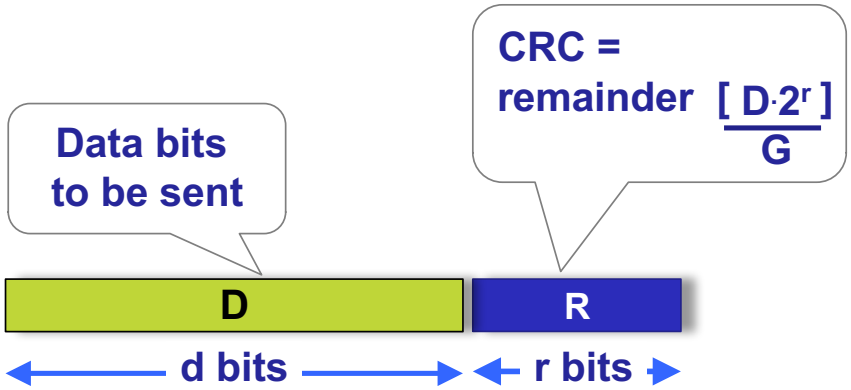


What is the transmitted frame <D , R>?

Transmitted frame:
<D , R> = 10101 1

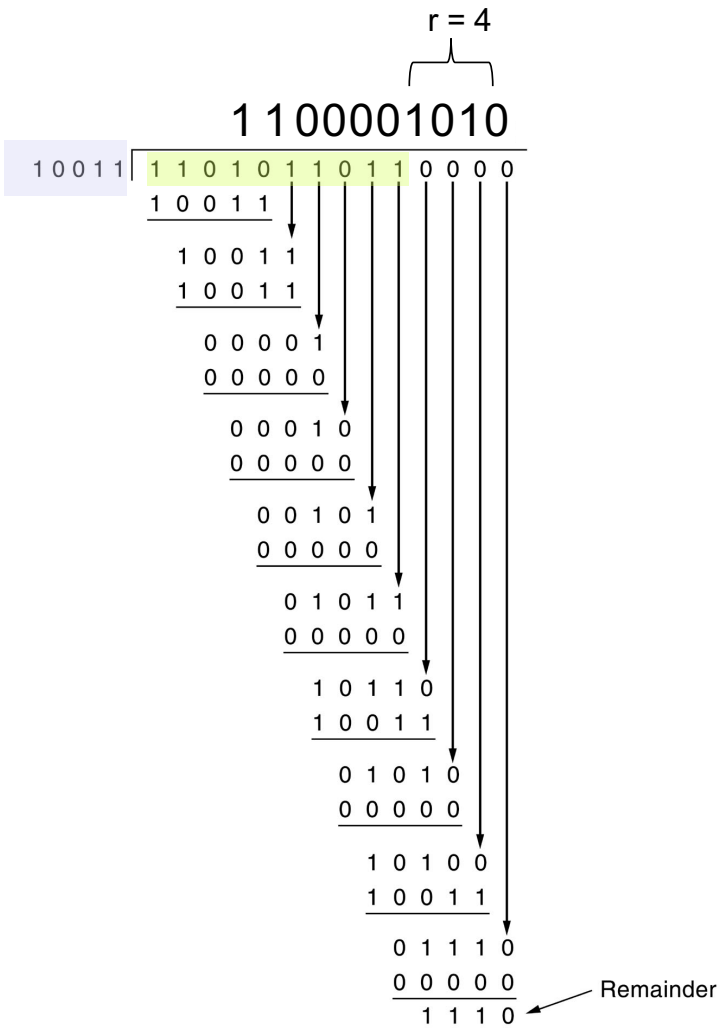
CRC-1 equals even parity

Exercise: CRC-4 find <D,R>



- Data D=1101011011, d = 10
- Generator G=10011, r = 4

Transmitted frame:
<D , R> = 1101011011 1110



R = 1110

Link layer and local area networks: Roadmap

5.1 Introduction and services

5.2 Error detection and correction

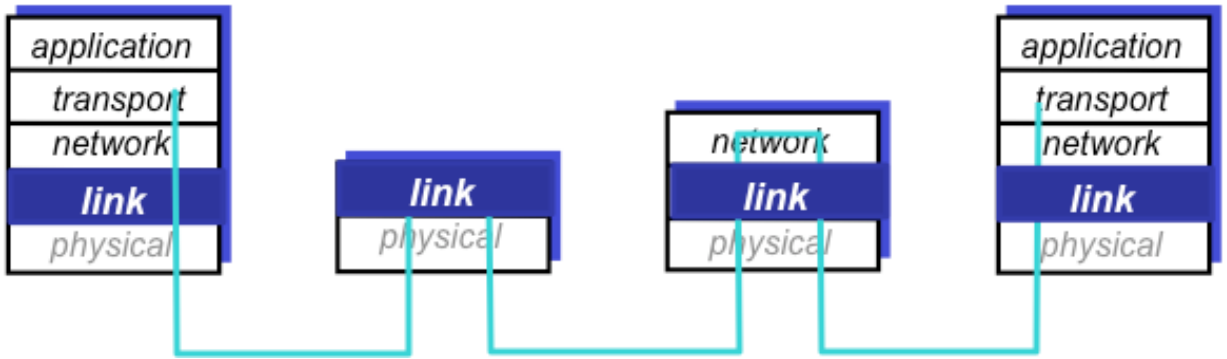
5.3 Multiple access protocols

5.4 Local Area Networks
- addressing, ARP
- Ethernet
- switches
- ~~Virtual LANS~~

~~5.5 Link virtualization~~

~~5.6 Data center networking~~

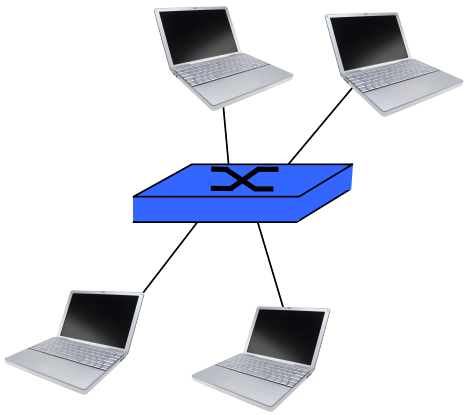
5.7 A day in the life of a web request



Two types of links

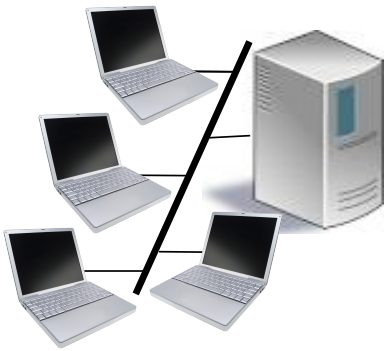
Point-to-point

- single sender, single receiver
  - **PPP** (point-to-point protocol) for dial-up access
  - **switched ethernet** point-to-point link between Ethernet switch and host

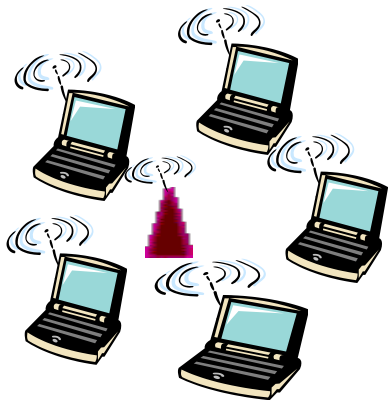


Broadcast

- senders and receivers on shared medium
- multiple access, e.g.
  - old-fashioned **Ethernet**
  - upstream Hybrid Fiber-Coax (**HFC**) in cable-TV networks
  - **802.11** wireless LAN



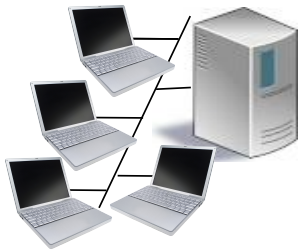
shared wire
(e.g. cabled Ethernet)

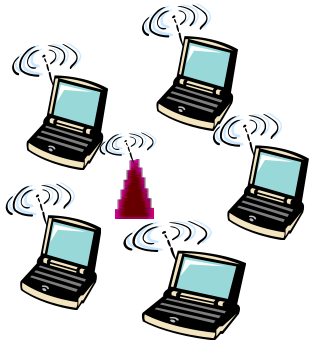


shared radio frequency
(e.g. 802.11 WiFi)

Transmission into a shared broadcast channel must be controlled

- Two or more simultaneous transmissions by nodes: **interference**
- **Collision** if node receives two or more signals at the same time
- Communication about channel sharing must use channel itself!
  - **no out-of-band channel** for coordination





An ideal medium access protocol
on an R b/s link:

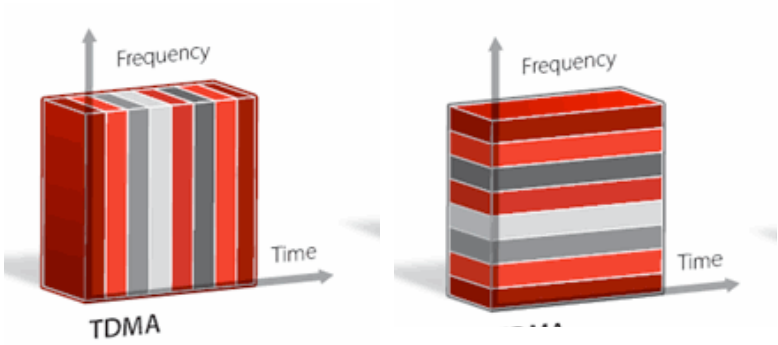
- sends at rate R when **one node** wants to transmit
- sends at **average rate R/M** when **M nodes** want to transmit
- is **fully decentralized**
  - no special node to coordinate transmissions
  - no synchronization of clocks, slots
- is **simple**

Transmission is regulated by a medium access protocol

Three MAC (multiple access control) protocol classes

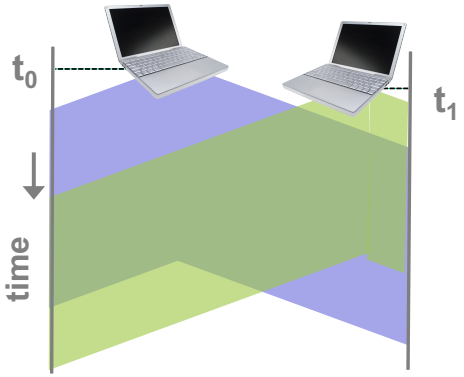
1. Channel partitioning

- divide channel into smaller “pieces” (**time** slots, **frequency**, **code**)
- allocate piece to node for exclusive use



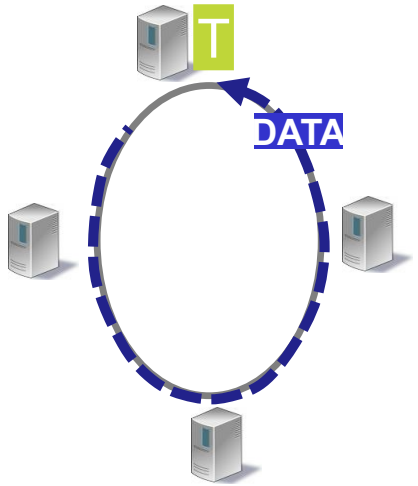
2. Random access

- channel not divided, allow collisions/interference
- “recover” from collisions



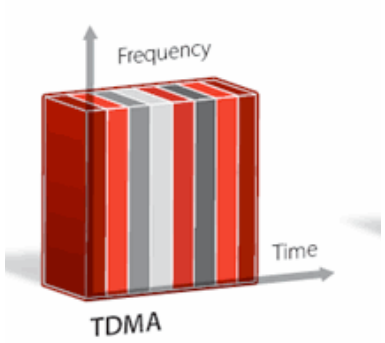
3. Taking turns

- nodes take turns, and nodes with more to send can take longer turns



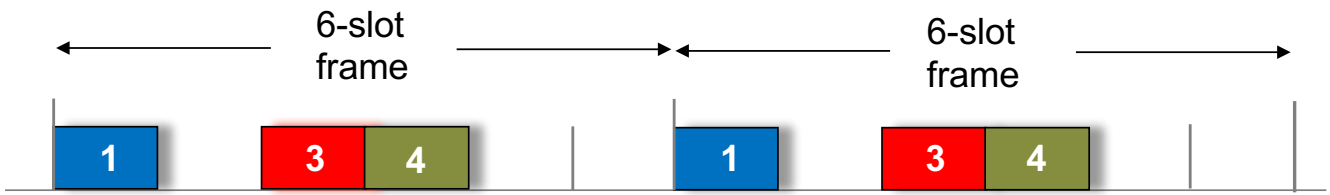
1. Channel partitioning MAC protocols

TDMA (time division multiple access)



- Access to channel in frames
- Each station gets fixed length slot (length = packet transmission time) in each frame
- Unused slots go idle

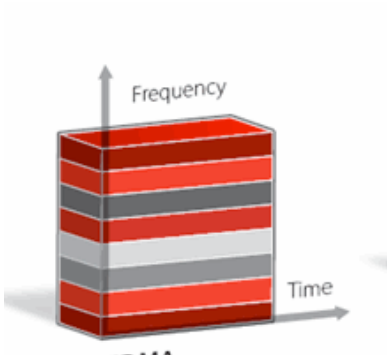
- Example: 6-stations
  - 1,3,4 have packet
  - slots 2,5,6 idle

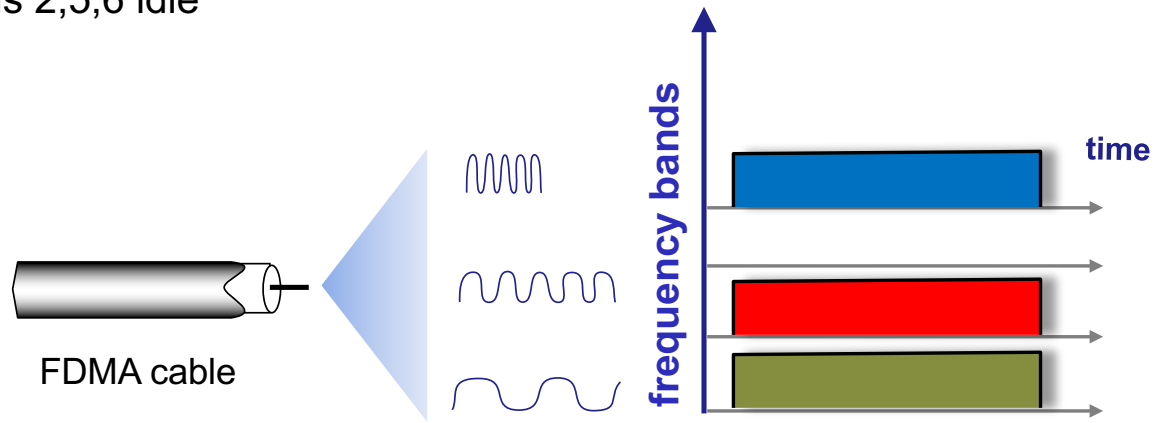


1. Channel partitioning MAC protocols

FDMA (frequency division multiple access)

- Channel spectrum divided into frequency bands
- Each station assigned fixed frequency band
- Unused transmission time in frequency bands go idle
- Example: 6-station LAN
  - 1,3,4 have packet
  - frequency bands 2,5,6 idle

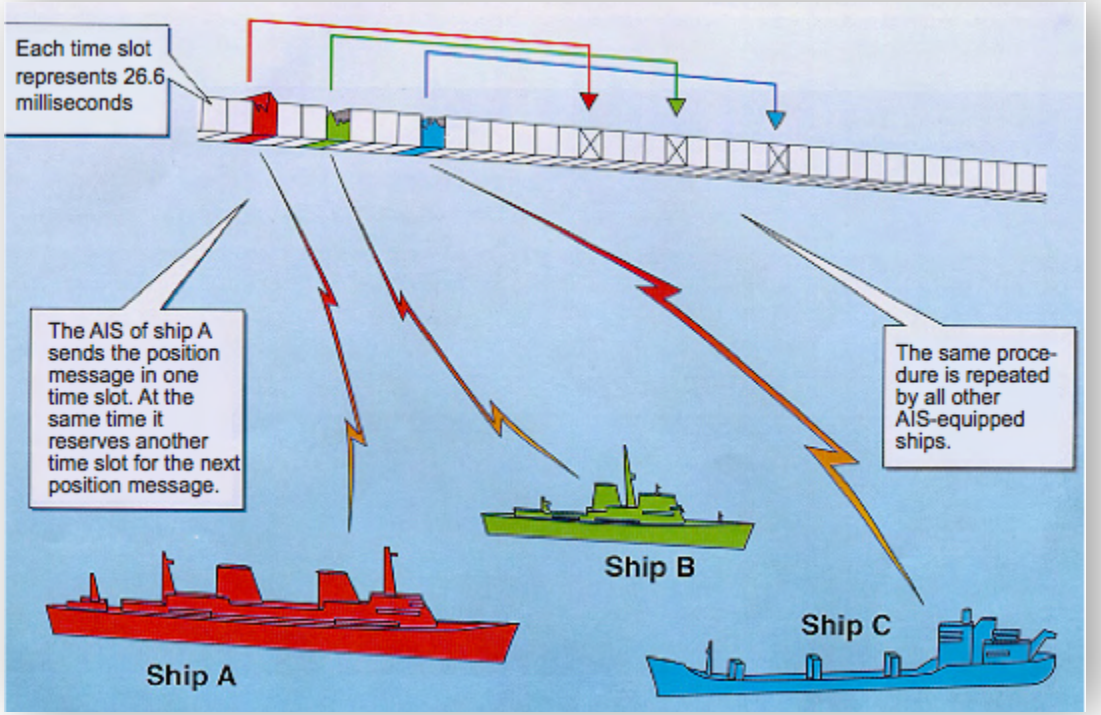




TDMA & FDMA example

Automatic Identification System - AIS

- AIS for situational awareness,
  collision avoidance,
  aid to navigation (buoys and lights),
  search and rescue (SAR) transmitter,
  SAR aircraft



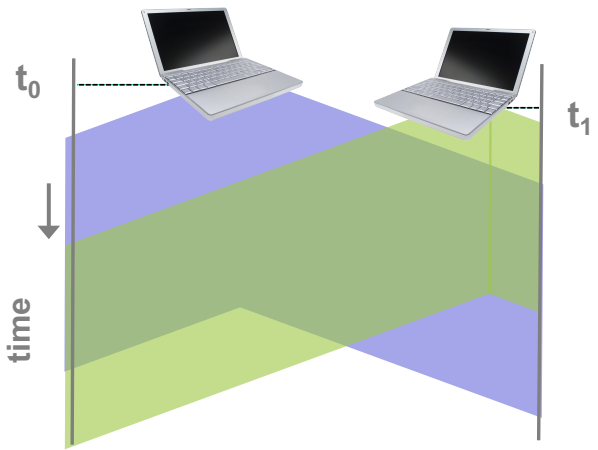
- Very High Frequency (VHF)
  radio broadcasting on
  two AIS frequencies
- 2250 **time slots** per frequency
  repeated every 60s,
  TDMA (self-organising or carrier sense)
- Timing and positional information from a
  GNSS (global navigation satellite system – such as GPS)

MarineTraffic

KYSTVERKET

2. Random access MAC (medium access control) protocols

- No a priori coordination among nodes
- When node has packet to send
  - transmit at full channel data rate R
- Two or more transmitting nodes
  ➜ **collision**

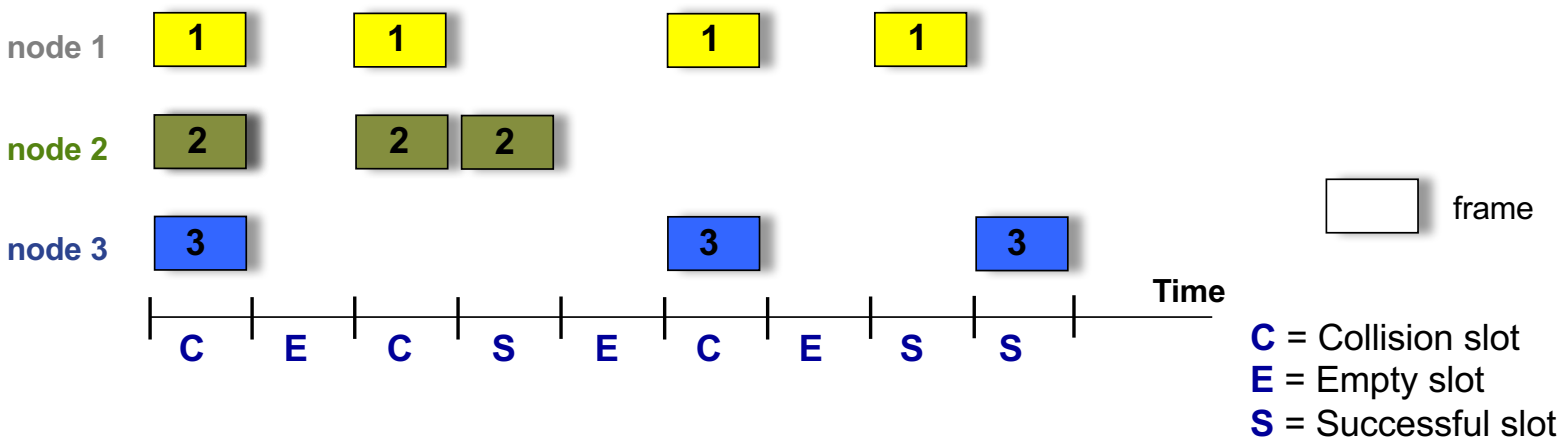


- Random access MAC protocol specifies
  - how to detect collisions
  - how to recover from collisions (e.g. via delayed retransmissions)

- Examples of random access MAC protocols
  - **slotted ALOHA**
  - **pure (unslotted) ALOHA**
  - **CSMA (carrier sense multiple access)**
  - **CSMA/CD (collision detection)**
  - **CSMA/CA (collision avoidance)**

2. Random access MAC (medium access control) protocols

Slotted ALOHA



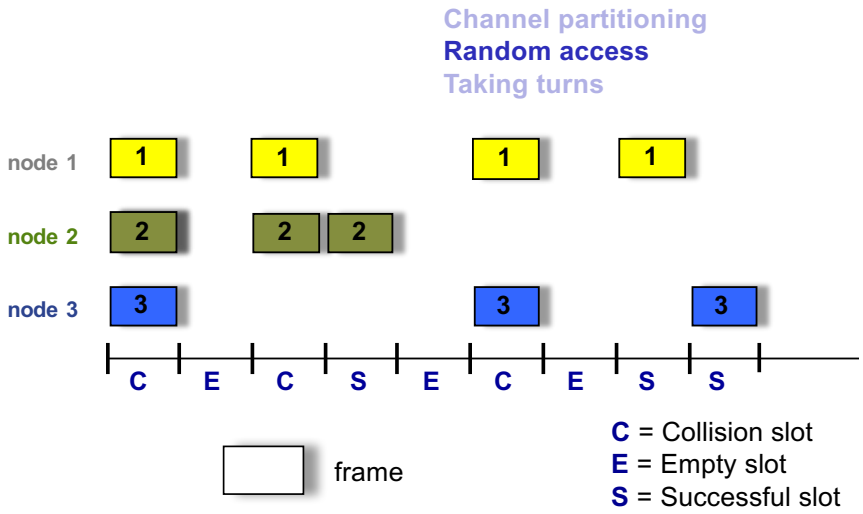
- Nodes are synchronized in time
- Time divided into **equal size slots** (time to transmit 1 frame)
- All **frames same size**
- Nodes **start** to **transmit** only at **slot beginning**
- If 2 or more nodes transmit in slot, all nodes detect collision

Slotted ALOHA operations

- When node obtains fresh frame, transmits in next slot
  - if **no collision**: node can send new frame in next slot
  - if **collision**: node retransmits frame in each subsequent slot with probability p until success

Pros

- Single active node can continuously transmit at full rate of channel
- Highly decentralized
- Extremely simple



Cons

- Collisions: wasting slots
- Empty slots
- Clock synchronization – slots need to be in sync
- Nodes must be able to detect collision in less time than time to transmit packet

Slotted ALOHA efficiency

Efficiency: long-run fraction of successful slots

- Suppose: **N** nodes with many frames to send, each transmits in slot with probability **p**
- Probability that given node has success in a slot = $\mathbf{p(1-p)^{N-1}}$
- Probability that any node has a success = $\mathbf{Np(1-p)^{N-1}}$

- Max efficiency: find p* that maximizes $Np(1-p)^{N-1}$
- For many nodes, take limit of $Np^*(1-p^*)^{N-1}$ as N goes to ∞, gives max efficiency = 1/e = .37

At best: channel used for useful transmissions 37% of time!

Slotted ALOHA – max efficiency

- **Max efficiency** find p^* that maximizes $Np(1-p)^{N-1}$
- $E'(p^*) = 0$
- $E'(p^*) = N(1-p^*)^{N-1} - Np^*(N-1)(1-p^*)^{N-2}$

  $\Rightarrow N(1-p^*)^{N-2} * [(1-p^*)-p^*(N-1)]$

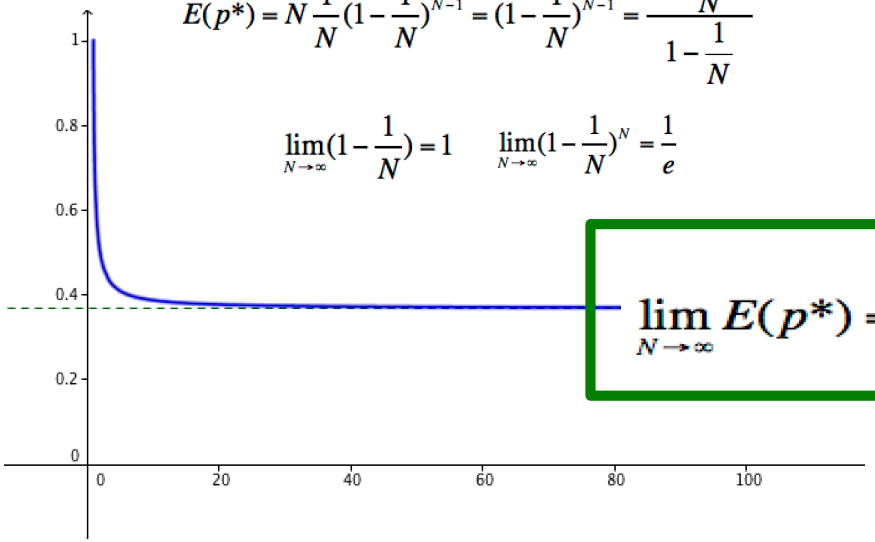
  $= 0$

-> $p^* = 1/N$

- Max efficiency is **$E(p^*) = Np^*(1-p^*)^{N-1}$, $p^* = 1/N$**

$$E(p^*) = N\frac{1}{N}(1-\frac{1}{N})^{N-1} = (1-\frac{1}{N})^{N-1} = \frac{(1-\frac{1}{N})^N}{1-\frac{1}{N}}$$

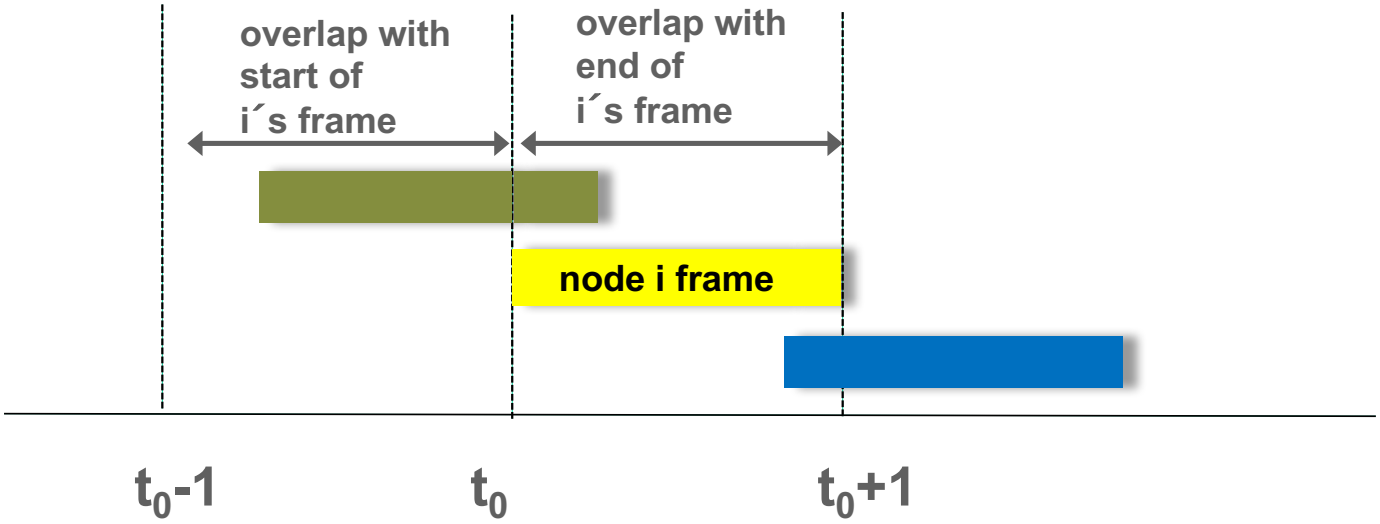
$$\lim_{N\to\infty}(1-\frac{1}{N}) = 1 \qquad \lim_{N\to\infty}(1-\frac{1}{N})^N = \frac{1}{e}$$



$$\lim_{N\to\infty} E(p^*) = \frac{1}{e}$$

2. Random access MAC protocols

Pure unslotted ALOHA



- Simpler, no synchronization of slots
- When frame first arrives, transmit immediately
- Collision probability increases
  - frame sent at t_0 collides with other frames sent in $[t_0-1,t_0+1]$

Pure unslotted ALOHA efficiency

P(success by given node) = P(node transmits) *

P(no other node transmits in $[t_0-1,t_0]$ *

P(no other node transmits in $[t_0, t_0+1]$

= p *
$(1-p)^{N-1}$ *
$(1-p)^{N-1}$ = p * $(1-p)^{2(N-1)}$

… choosing optimum p and then letting N -> ∞

= 1/(2e) = **.18**

even worse than slotted Aloha!

2. Random access MAC protocols

CSMA (Carrier Sense Multiple Access)

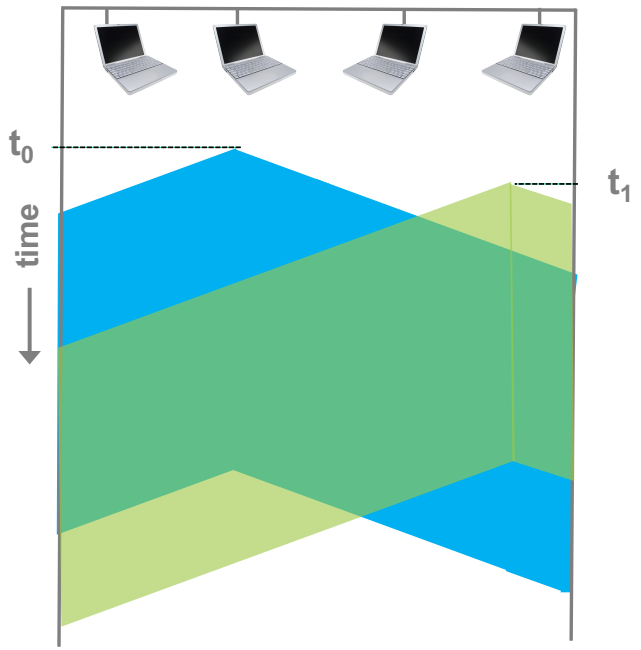
Listen before transmit

- If channel sensed idle, transmit entire frame
- If channel sensed busy, defer transmission

Collisions can still occur

- Due to propagation delay two nodes may not hear each other's transmission
- Entire packet transmission time wasted

- Distance & propagation delay play a role in determining collision probability

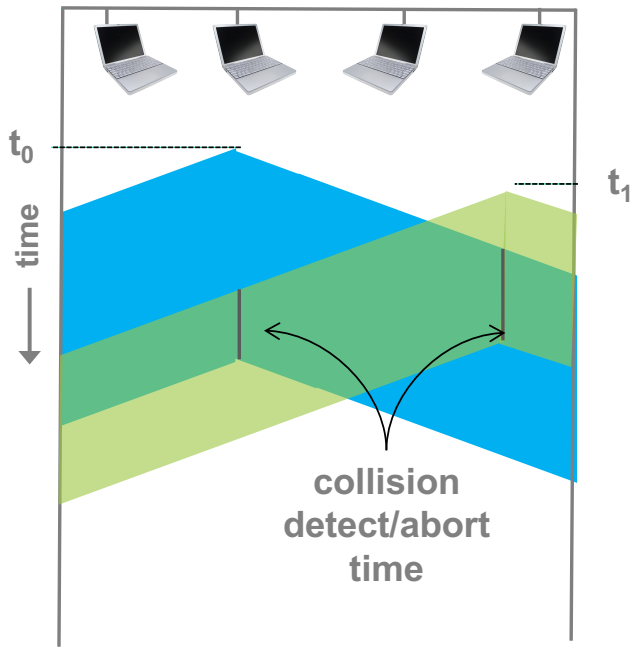


NTNU

2. Random access MAC protocols

CSMA/CD (Collision Detection)

- **CSMA/CD: carrier sense multiple access**
- **CSMA/CD: collision detection**
  - colliding transmissions aborted, reducing channel wastage
  - Wired LANs easy: measure signal strengths, compare transmitted and received signals
  - Wireless LANs difficult: received signal strength overwhelmed by local transmission strength

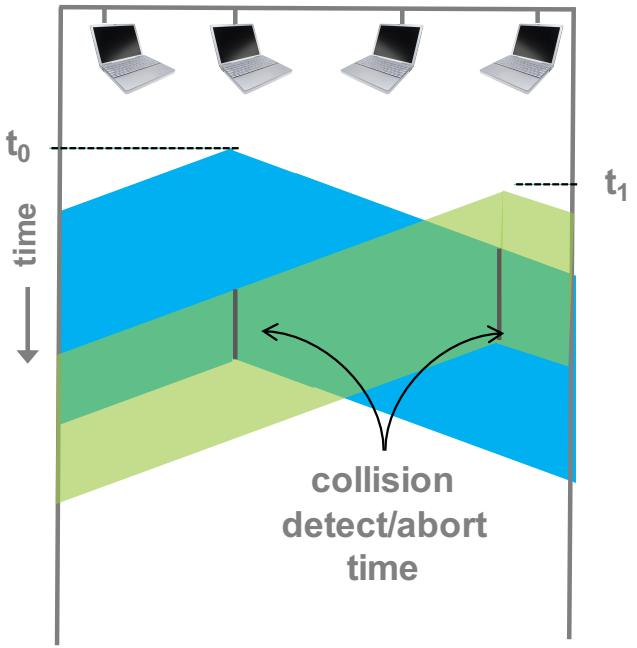


This is the ethernet medium access control protocol

Ethernet CSMA/CD algorithm is run for each new frame to be transmitted

1. Prepare an ethernet frame from the network layer datagram
2. If NIC (Network Interface Card) senses
   - **channel idle**, start frame **transmission**
   - **channel busy**, **wait** until channel idle, then transmit
3. If NIC transmits entire frame without detecting another transmission, NIC is done with frame!
4. If NIC **detects collision**
   - **abort** and send **jam signal**
5. After aborting, NIC enters **exponential backoff**

Repeat.



http://www.youtube.com/watch?v=yGJGIUa-xWE&feature=related

Ethernet CSMA/CD operation details

- **Detecting collision:** detects signal energy from other adapters
- **Jam signal**: make sure all other transmitters are aware of collision; 48 bits
- **Bit time**
  - 100 nanosec for 10 Mbps
  - 10 nanosec for 100 Mbps
  - 1 nanosec for 1 Gbps
- **Exponential backoff**
  - Goal: adapt retransmission attempts to estimated current load
  - heavy load: random wait will be longer
  - **first collision**: choose K from {0,1}; delay is K*512 bit transmission times
  - after **second collision**: choose K from {0,1,2,3}…
  - after **ten collisions**, choose K from {0,1,2,3,4,…,1023}
  - for K=1023, wait time is about
    - 50 msec for 10 Mbps
    - 5 msec for 100 Mbps
    - ,5 msec for 1 Gbps

CSMA/CD efficiency depends on network length and max frame size

- t_{prop} = max propagation delay between 2 nodes in LAN
- t_{trans} = time to transmit max-size frame

$$efficiency = \frac{1}{1 + 5t_{prop}/t_{trans}}$$

- Efficiency goes to 1
  - as t_{prop} goes to 0
  - as t_{trans} becomes very large

- **Better performance than ALOHA**: and simple, cheap, decentralized!

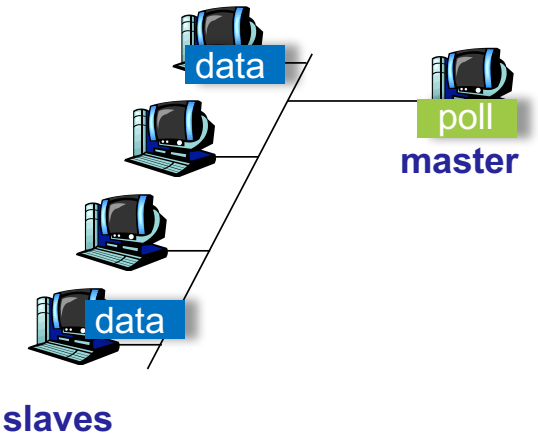
NTNU

3. Taking turns MAC protocols

Polling protocol

Polling

- **Master** node “invites” slave nodes to transmit in turn
- Typically used with “dumb” **slave** devices
- Concerns
  - polling overhead
  - latency
  - single point of failure (master)

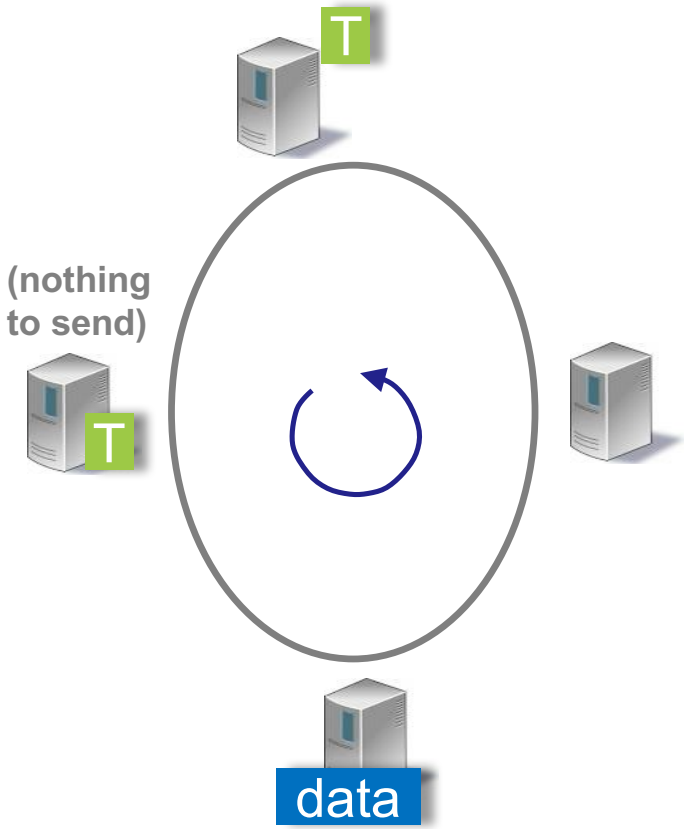


NTNU

3. Taking turns MAC protocols

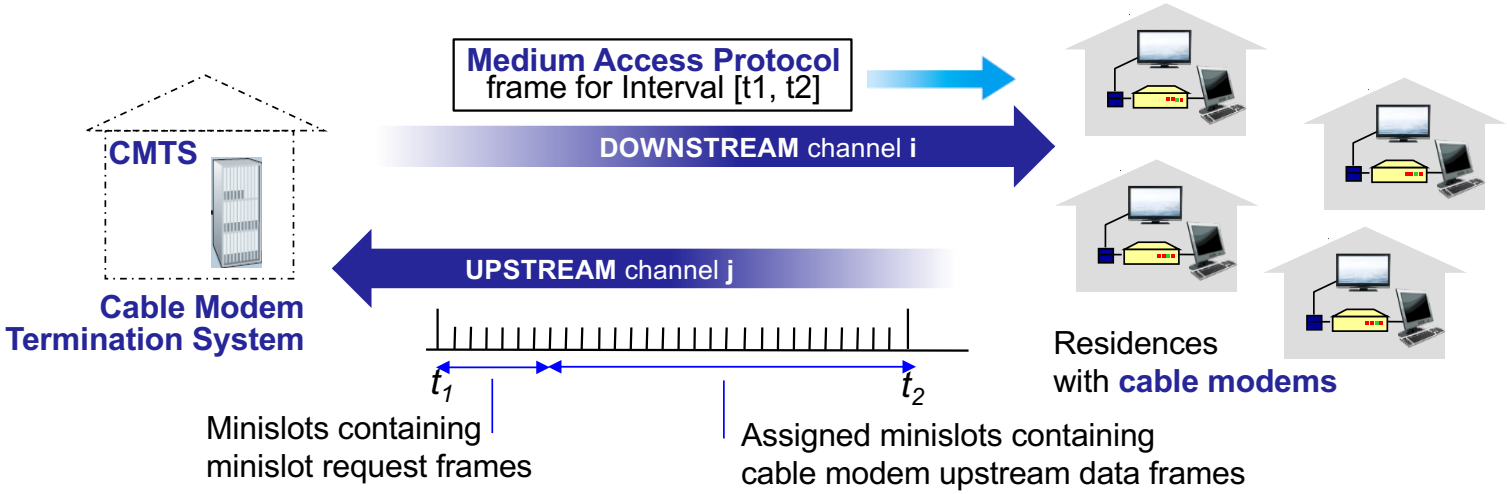
Token passing protocol

- Control token passed from one node to next sequentially
  - token message
- No master node
- Concerns
  - token overhead
  - latency
  - single point of failure (token)



Channel partitioning, random access, taking turn example

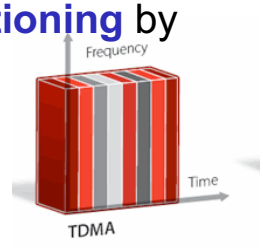
Cable access network EuroDOCSIS: European Data Over Cable Service Interface Specification



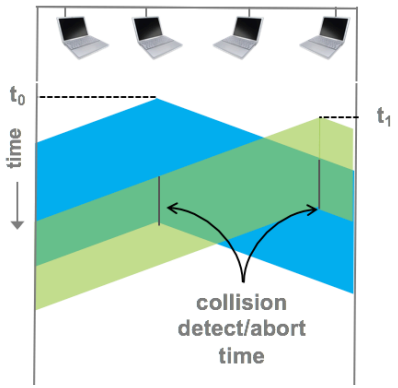
- **FDMA** upstream, downstream frequency channels
- Multiple **downstream** (broadcast) channels (50 Mb/s)
  - downstream **MAP (medium access protocol)** frame: **assigns upstream slots (taking turn)**
- Multiple **upstream** (TDM) channels (27 Mbps)
  - some slots centrally assigned (taking turn)
  - some have contention request for upstream slots (and data) transmitted by **random access** (binary exponential backoff) in selected slots

Summary of MAC (medium access control) protocols

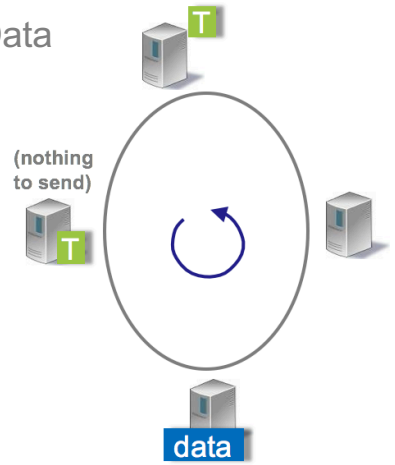
- **Channel partitioning** by time, frequency or code



- **Random access** (dynamic)
  - ALOHA, slotted-ALOHA
  - Carrier sensing: easy in some technologies (wire), hard in others (wireless)
  - CSMA/CD used in Ethernet



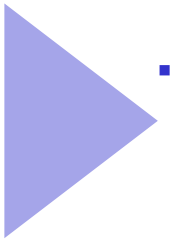
- **Taking turns**
  - Polling from central site
    - Bluetooth
    - 802.15 - low-rate wireless personal area networks (PANs)
  - Token passing
    - FDDI (Fiber Distributed Data Interface)
    - IBM Token Ring



NTNU

Performance of the three MAC protocol classes

- **Channel partitioning MAC** protocols
  - high load: share channel fairly and efficiently
  - low load: inefficient, delay in channel access, 1/N bandwidth allocated even if only 1 active node!
- **Random access MAC** protocols
  - high load: collision overhead
  - low load: efficient, single node can fully utilize channel



- **Taking turns** protocols
  - look for best of both worlds!

NTNU

Links, access networks and LANs
Roadmap

5.1 Introduction and services

5.2 Error detection and correction

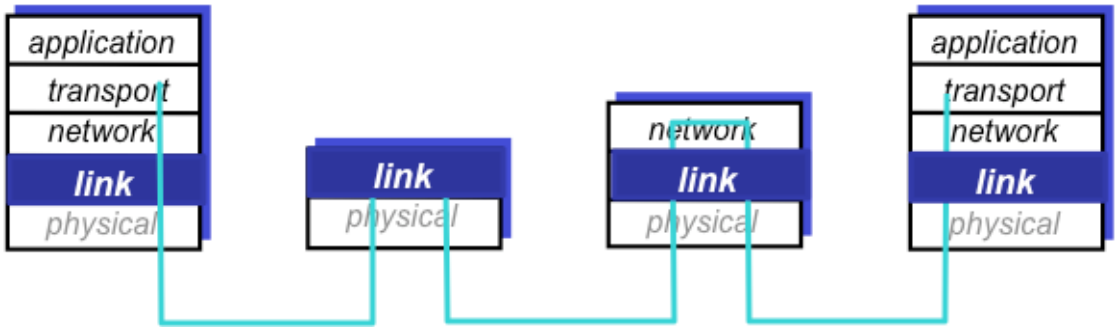
5.3 Multiple access protocols

5.4 Local Area Networks
- addressing, ARP
- Ethernet
- switches
- ~~Virtual LANS~~

~~5.5 Link virtualization: MPLS~~

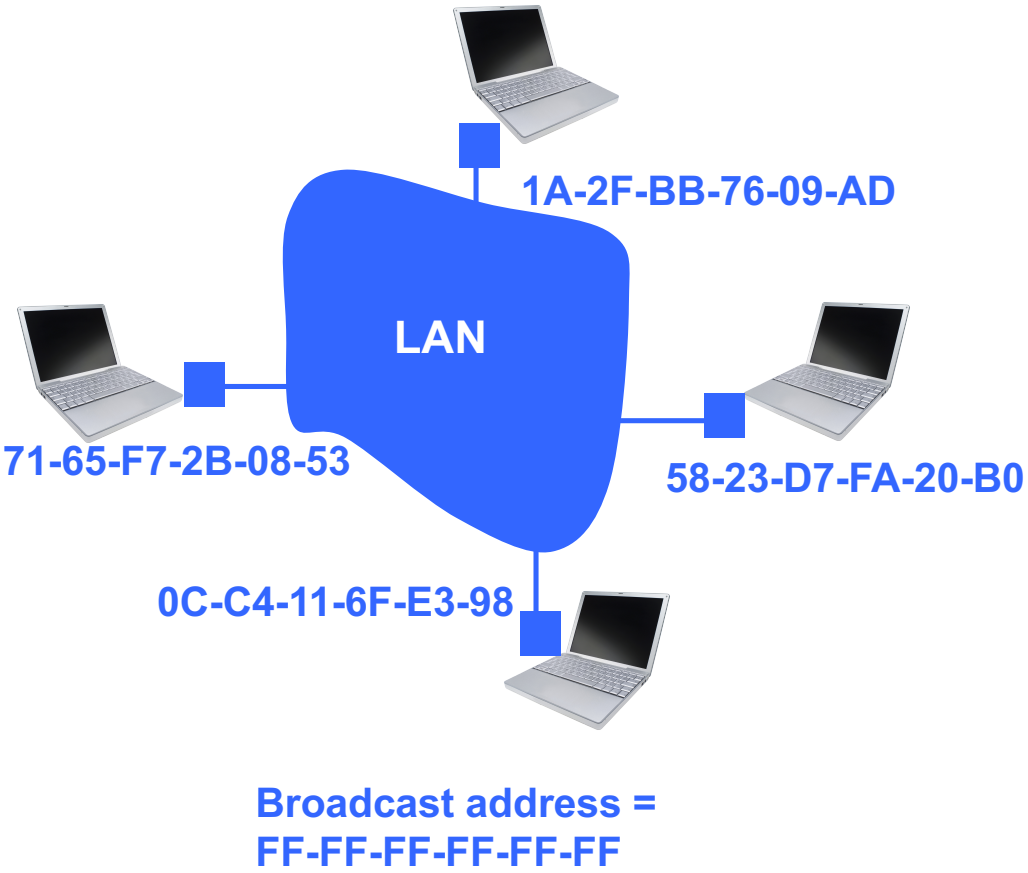
~~5.6 Data center networking~~

5.7 A day in the life of a web request



Each network interface has a unique MAC addresses = LAN address

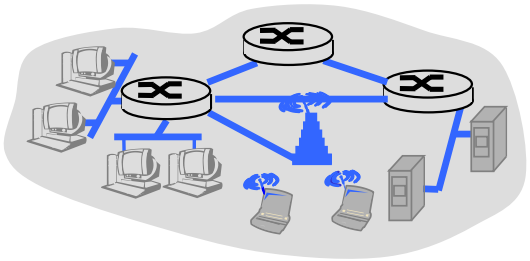
- **MAC** (LAN or physical) **address**
  - 48-bit
    - flat address structure
    - burned in network interface card, sometimes software settable
- MAC address allocation administered by IEEE
  - manufacturer buys portion of MAC address space (to assure uniqueness)
    http://standards.ieee.org/regauth/oui/oui.txt



LAN (MAC) address vs IP address

- 48-bit **MAC address**
  - get frame **from one interface to another**
    - like social security number
- MAC address is flat ➜ portability
  - can move LAN card from one LAN to another
- **72:00:01:0c:33:71**

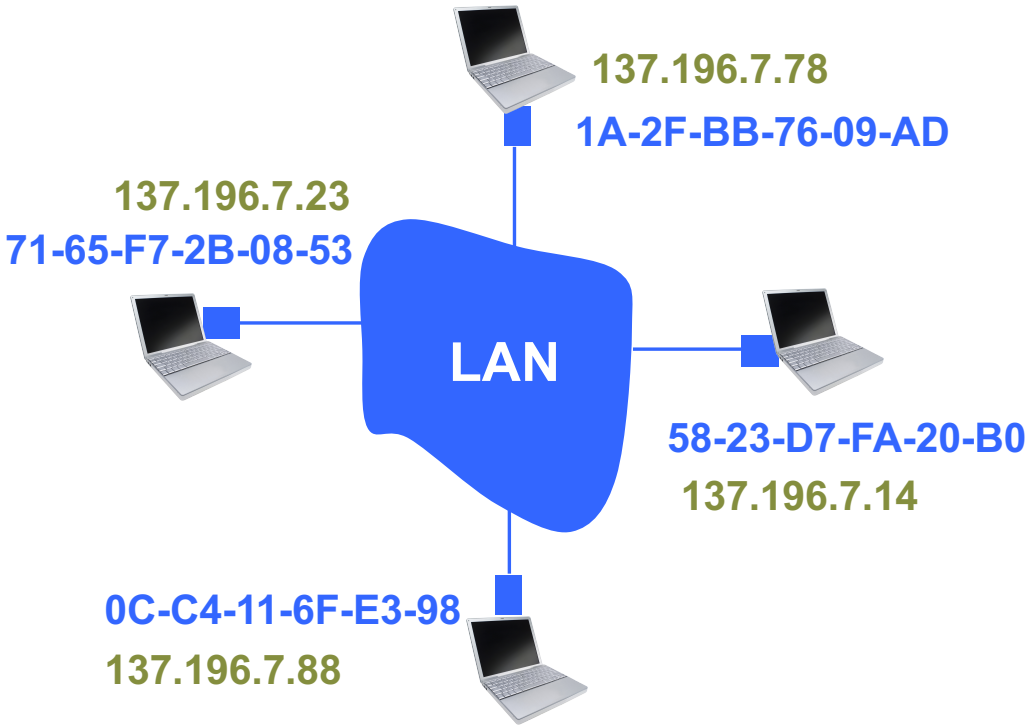
- 32-bit **IPv4 address**
  - used to get datagram to **destination IP subnet**
    - like postal address
- IP address is hierarchical ➜ NOT portable
  - address depends on IP subnet to which node is attached
- **129.241.67.244**



The Address Resolution Protocol (ARP) finds mapping between IP and MAC address within a LAN

- Each IP node on LAN has ARP table
- **ARP table**: IP/MAC address mappings for LAN nodes
  - [**IP address**; MAC address; TTL]

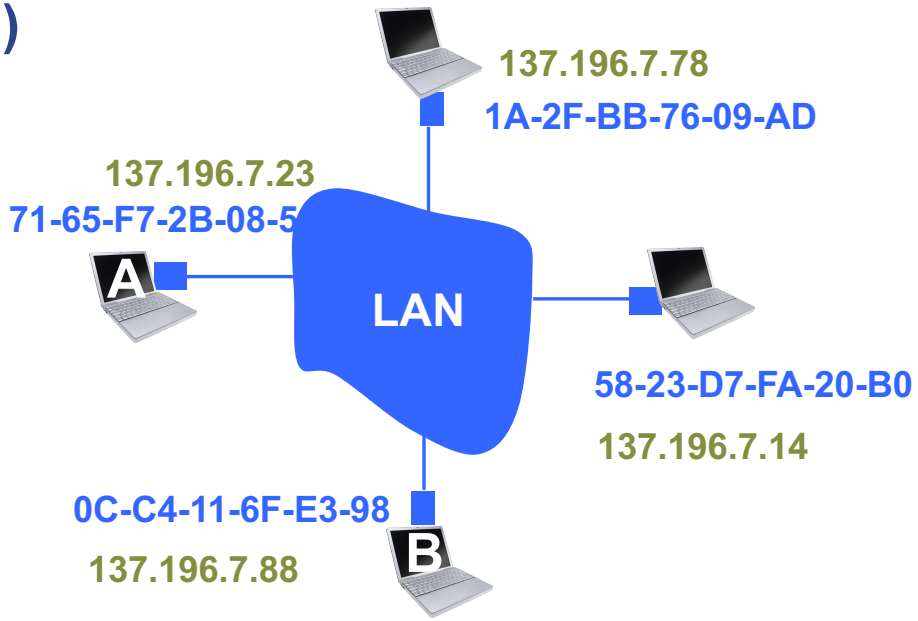
  TTL (Time To Live): time after which address mapping will be forgotten (typically 20 min)



The Address Resolution Protocol (ARP) is plug-and play

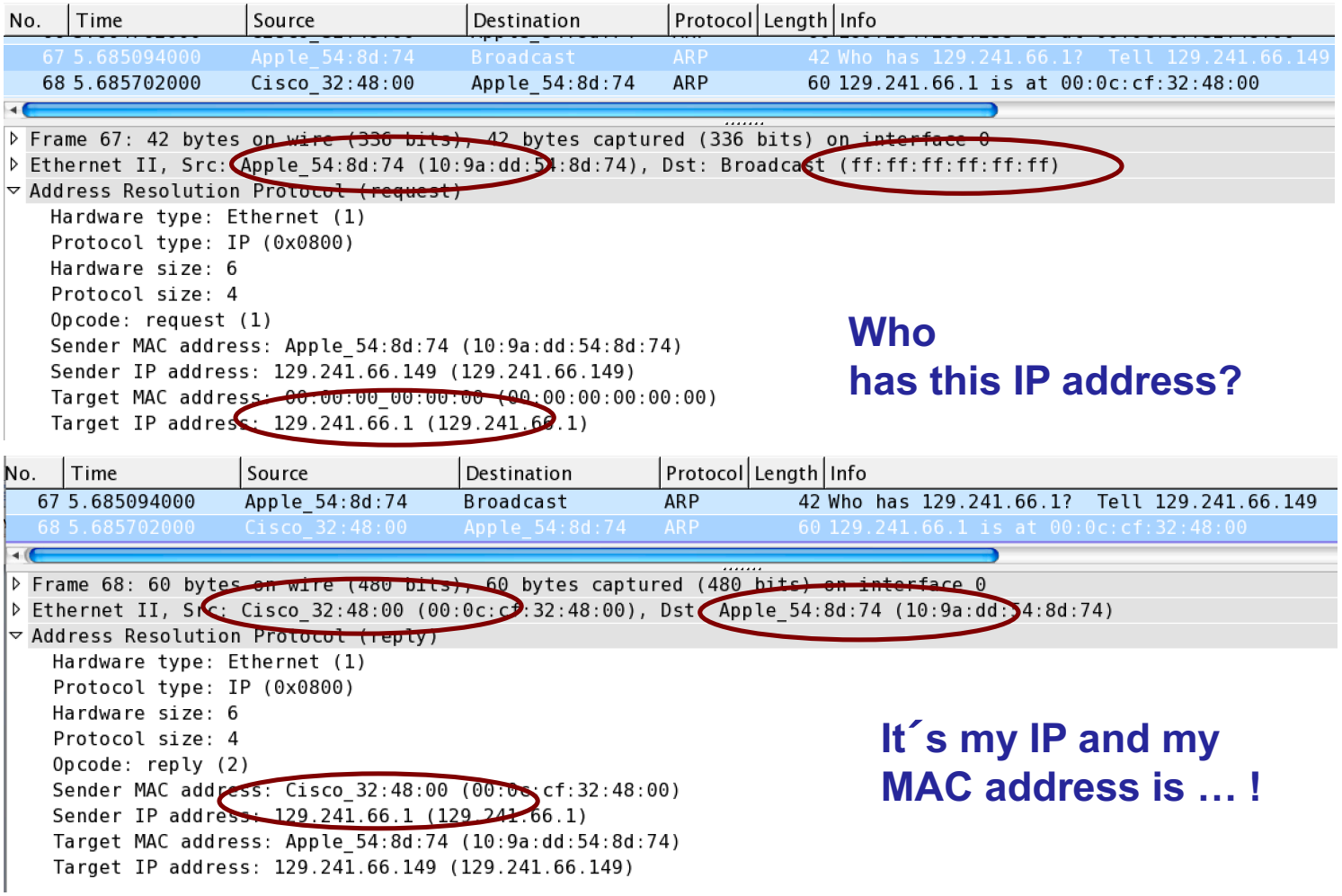
- ARP is “plug-and-play”
  - nodes **create** their **ARP tables without intervention** from net administrator
- **A** wants to send datagram to **B**, **B**’s MAC address not in **A**’s ARP table
- **A broadcasts ARP query** packet, containing **B**'s IP address

  all machines on LAN receive ARP query
  - dest MAC address = FF-FF-FF-FF-FF-FF
- **B** receives ARP packet, replies to **A** with its (**B**'s) MAC address
  - **reply** sent to **A**’s MAC address (**unicast**)



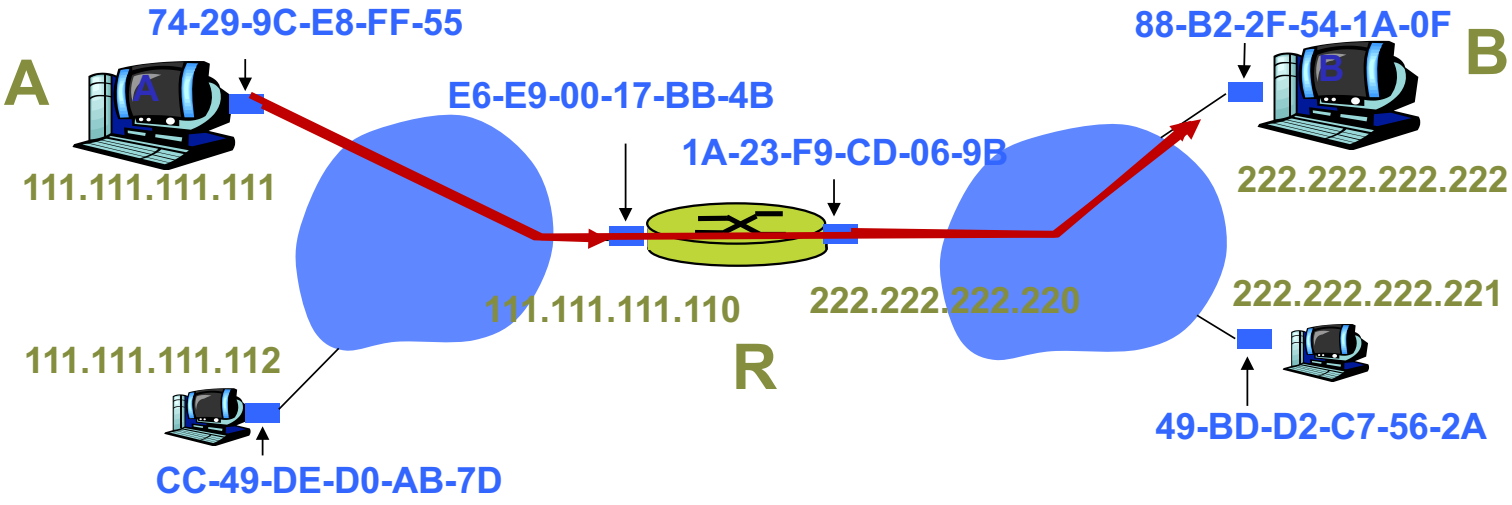
- **A** caches (saves) IP-to-MAC address pair in its **ARP table** until information becomes old (times out)
  - **soft state**: information that times out (goes away) unless refreshed

ARP request – response in Wireshark



ARP is run within each separate LAN

- Send datagram from **A** to **B** via **R**
  - Routing table in **A**: Destination **B** -> **R** 111.111.111.110 next hop
  - Routing table in **R**: Destination **B** -> **B** 222.222.222.222 next hop
- Two ARP tables in router R, one for each IP subnetwork (LAN)



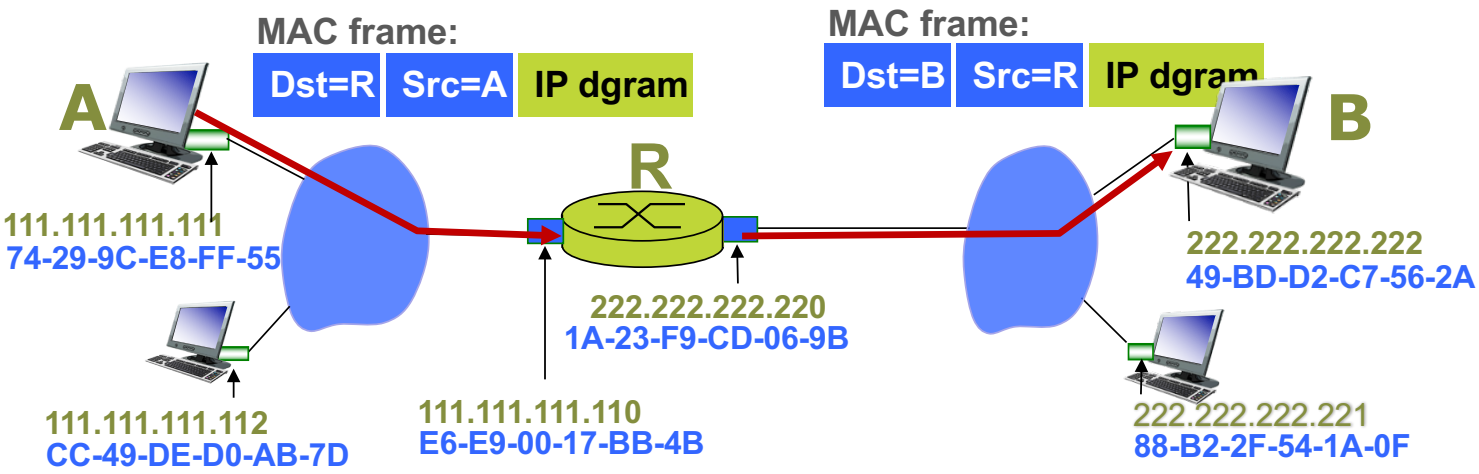
Packet to other subnet is forwarded hop-by-hop via link layer

IP datagram :

Src=A	Dst=B	

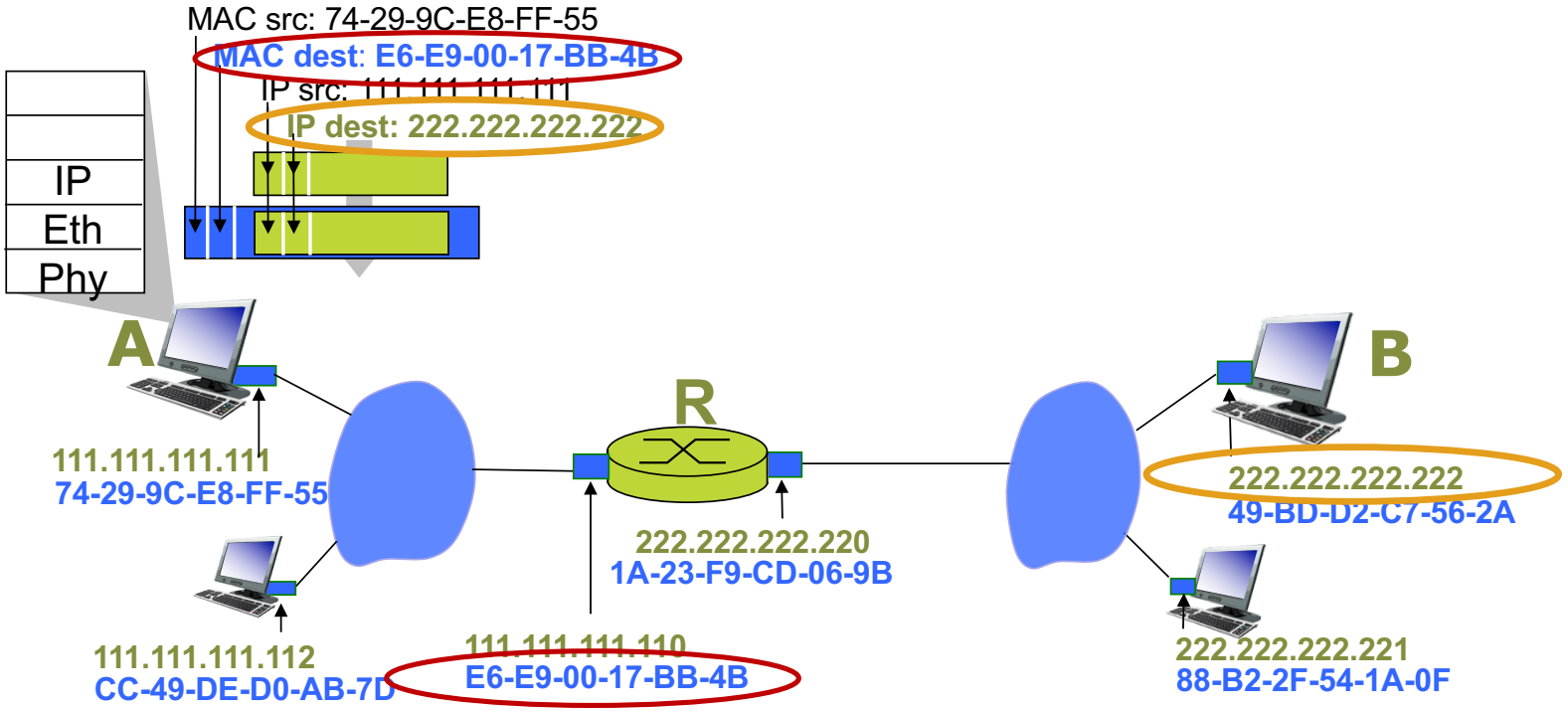
- Forwarding table gives next hop 111.111.111.110
- ARP in **A** gets MAC address **R** for next hop IP-address

- **R**'s network interface card receives frame removes ethernet header
- Datagram destined to **B**; ARP gets 's **B** MAC address



From source host to first hop router

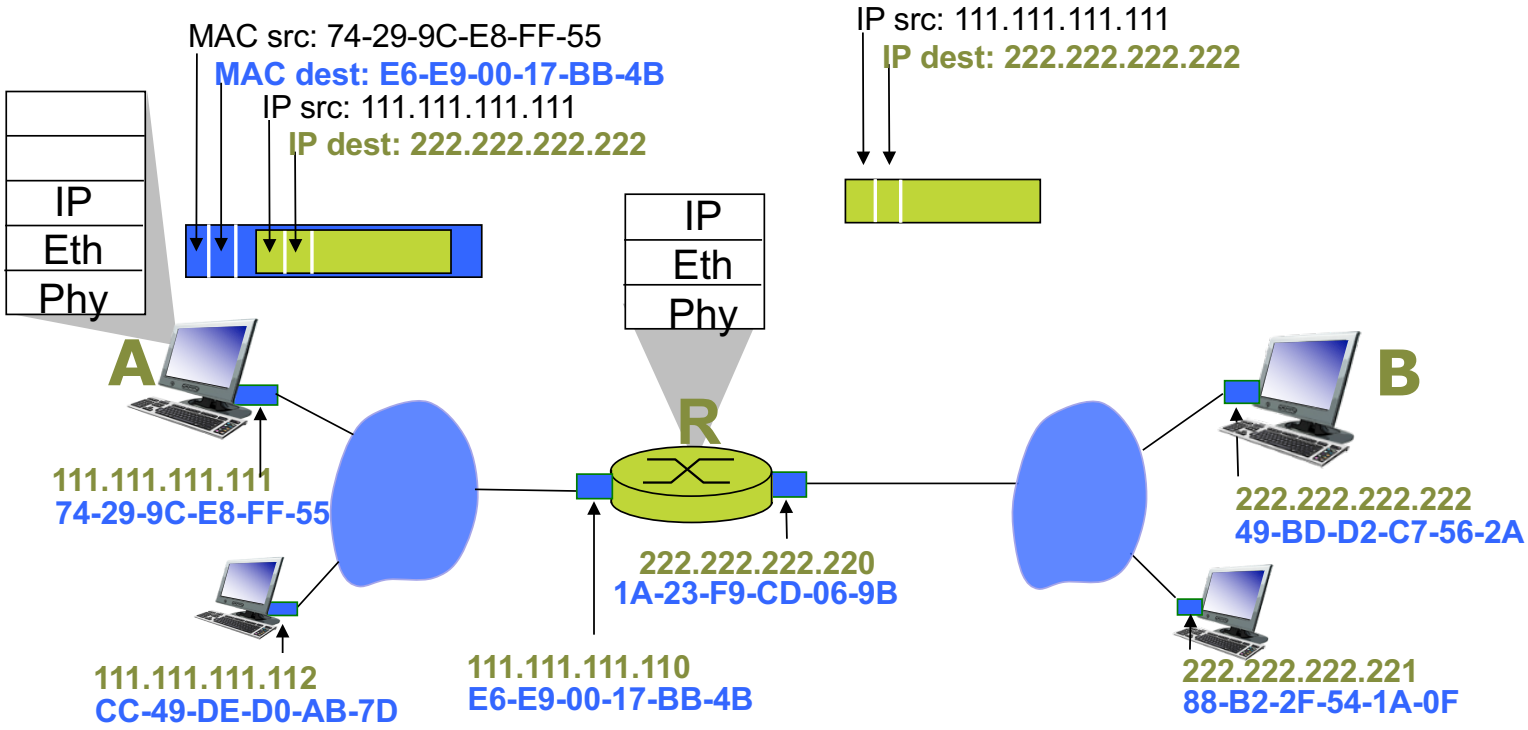
- A creates IP datagram with IP source A, destination B
- A creates link-layer frame with R's MAC address as destination, frame contains A-to-B IP datagram



Addressing: routing to another LAN

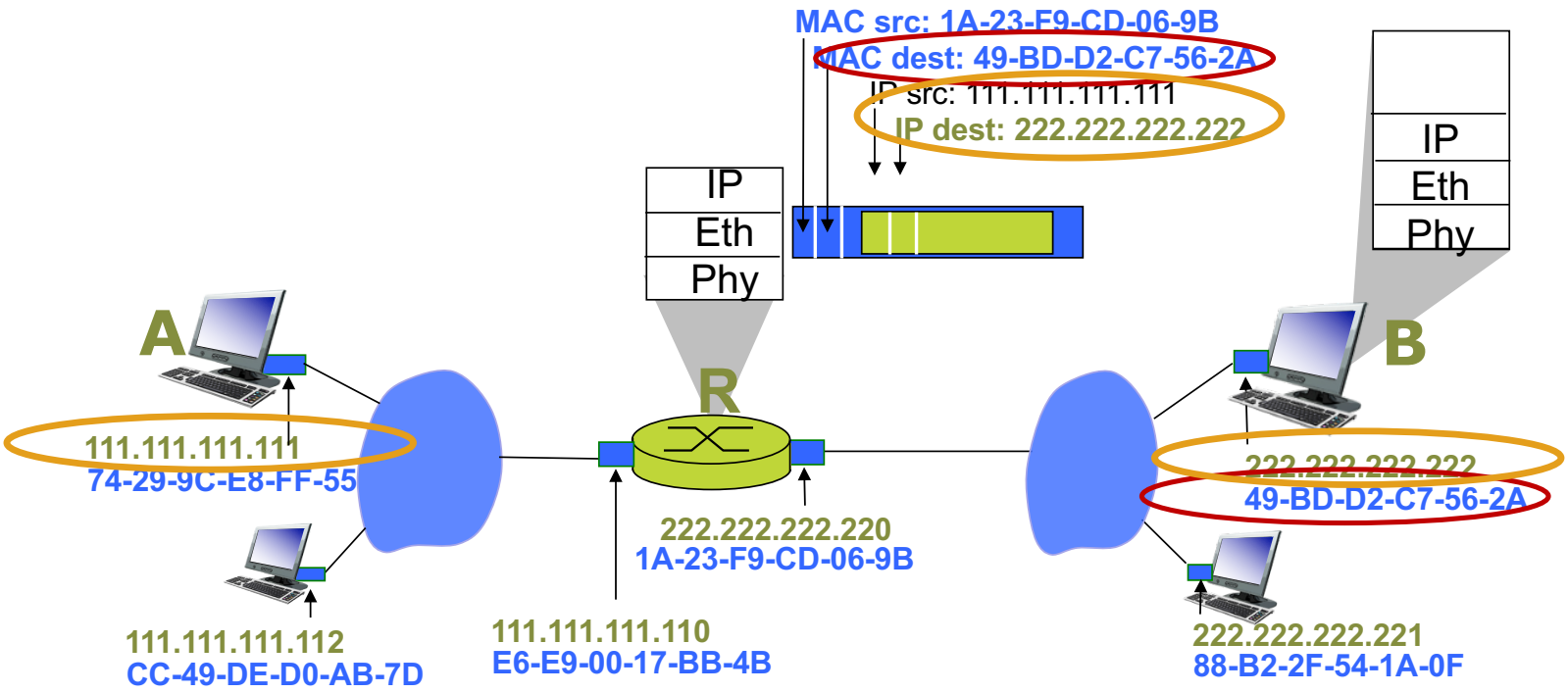
From source host to first hop router

- Source **A**´s link layer sends frame to **R**
- **R** receives link-layer frame, removes header, and passes datagram to IP



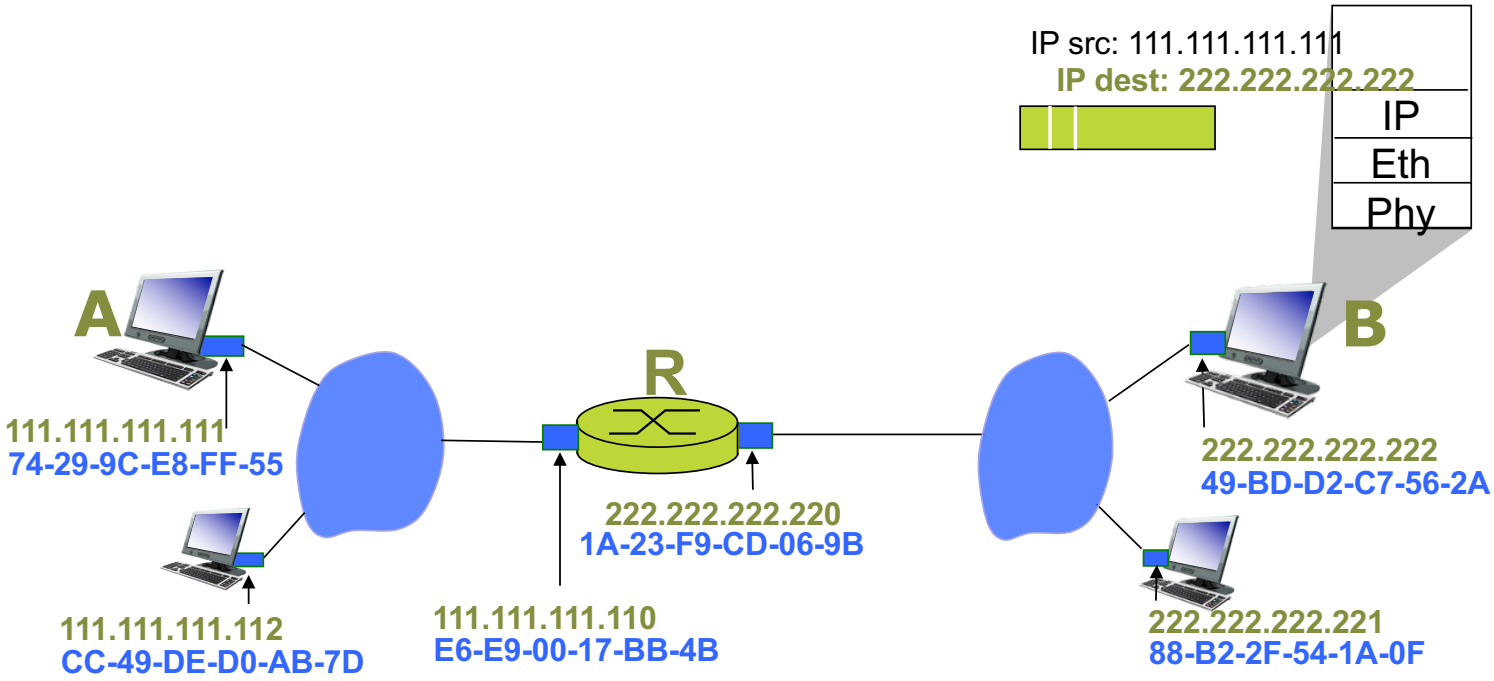
Addressing: routing from last hop router to destination host

- R forwards datagram with IP source A, destination B
- R creates new link-layer frame with B's MAC address as destination, frame contains A-to-B IP datagram



Addressing: routing from last hop router to destination host

- Destination **B** receives link-layer frame, removes headers and passes datagram up to IP



NTNU

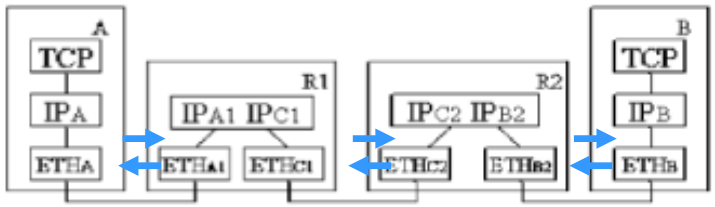
Exercise

Address resolution

Preamble	Dest. address	Source address	Type	Data, eg IP packet	CRC

Version	IHL	Type of Service	Total Length	
Identification			Flags	Fragment Offset
Time to Live	Protocol	Header Checksum		
Source Address				
Destination Address				
Options				Padding

A TCP connection is set-up between A and B through two routers R1 and R2. List the order in which the relevant address translations are taking place.



IPC2 is translated to ETHC2		**IPB is translated to ETHA1**	
IPA is translated to ETHC2		**IPB is translated to ETH**	
IPA is translated to ETHA1		**IPC1 is translated to ETHC1**	
IPB2 is translated to ETHB2		**IPB is translated to ETHB**	
IPB is translated to ETHC2		**IPB is translated to ETHC1**	
IPA is translated to ETHC1		**IPA is translated to ETHA**	
IPA1 is translated to ETHA1		**IPA is translated to ETHB2**	

Links, access networks and LANs
Roadmap

5.1 Introduction and services

5.2 Error detection and correction

5.3 Multiple access protocols

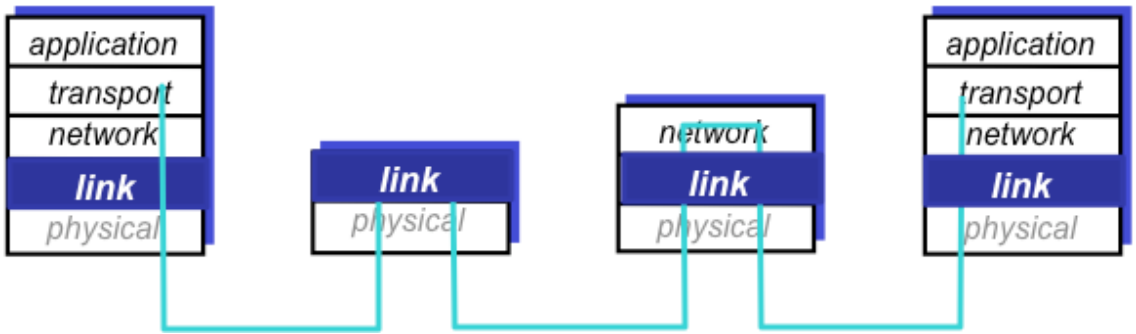
5.4 LANs

- addressing, ARP
- **Ethernet**
- switches
- ~~VLANS~~

~~5.5 Link virtualization: MPLS~~

~~5.6 Data center networking~~

5.7 A day in the life of a web request

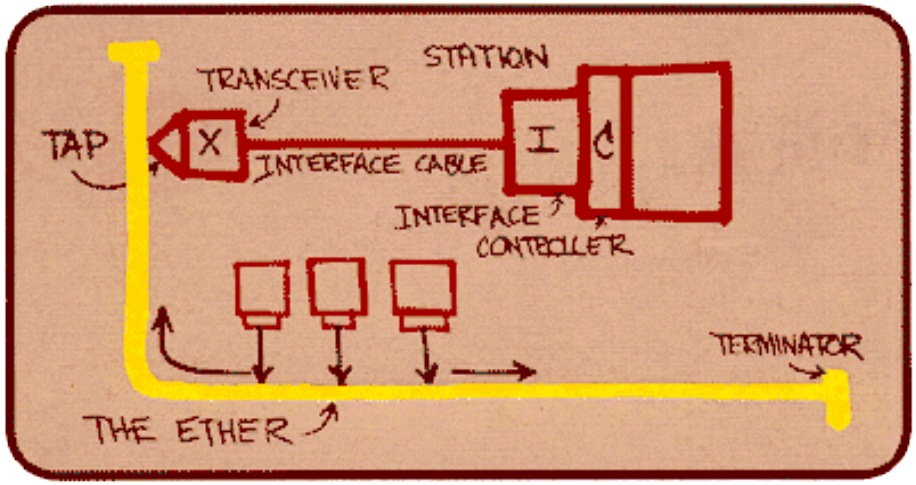


Ethernet – a good old technology

- **First** and dominant wired used LAN technology
- **Cheap** $20 for network interface card
- Kept up with **speed** race: 10 Mbps – 10 Gbps

- **Connectionless**: No handshaking between sending and receiving network interface cards (NICs)
- **Unreliable**: receiving NIC (network interface card) doesn't send ACKs or NACKs to sending NIC
  - Stream of datagrams passed to network layer can have gaps (missing datagrams)
- Ethernet's medium access control protocol: unslotted CSMA/CD

Metcalfe's original Ethernet sketch

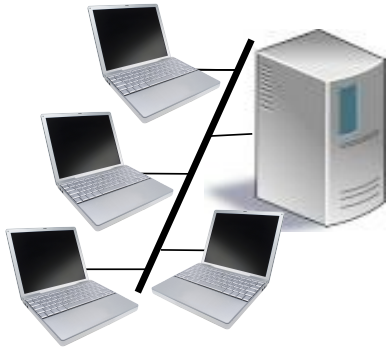


http://www.ieee802.org/3/

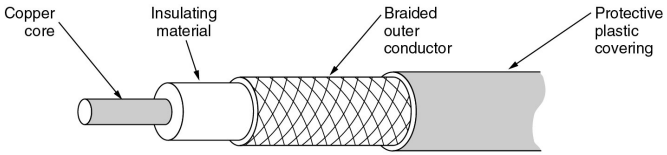
How is this handled?

Ethernet physical topology: from bus to star with switch in center

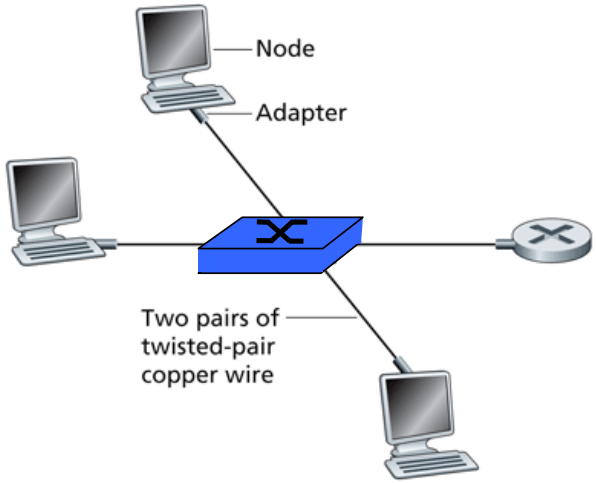
- **Bus topology** popular through mid 90s
  - All nodes in same collision domain (can collide with each other)



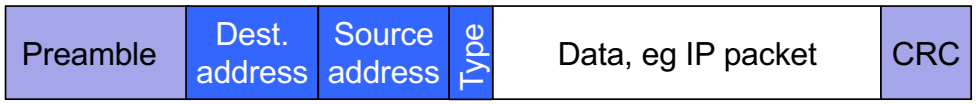
Bus: coaxial cable



- Today: **star topology** prevails
  - Active switch in center
  - Each “spoke” runs a (separate) Ethernet protocol
  - Nodes do not collide with each other



Ethernet frame structure



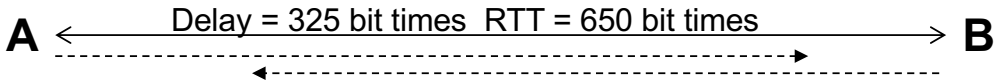
- **Preamble**: 8 bytes
  7*10101010 + 10101011
  - used to synchronize receiver-sender clock rates
- **Destination and source addresses**: 6 bytes each
  - received frame matching destination address, pass data in frame to network layer protocol
  - otherwise, discard frame
- **Type**: 2 bytes, indicates higher layer protocol (mostly IP but others possible, e.g. Novell IPX, AppleTalk)
- **Data field:** 46-1500 bytes
- **CRC**: 4 bytes, CRC-32 checked at receiver: if error is detected, frame is dropped

Where is the preamble and CRC?

```
▼ Ethernet II, Src: Apple_98:a8:e0 (60:03:08:98:a8:e0), Dst: Cisco_9f:f0:c8)
  ▶ Destination: Cisco_9f:f0:c8 (00:00:0c:9f:f0:c8)
  ▶ Source: Apple_98:a8:e0 (60:03:08:98:a8:e0)
    Type: ARP (0x0806)
▶ Address Resolution Protocol (request)
```

There is a maximum ethernet segment length to be sure to detect collisions

- Suppose A and B on the same ethernet.
  Propagation delay between the two nodes:



- A begins transmitting a frame, and before it finishes, B starts transmitting a frame

- A begins transmission at t=0. Worst case A transmits a minimum sized frame of

(8+ (14 +46) +4) bytes = 72 bytes = 576 bits.

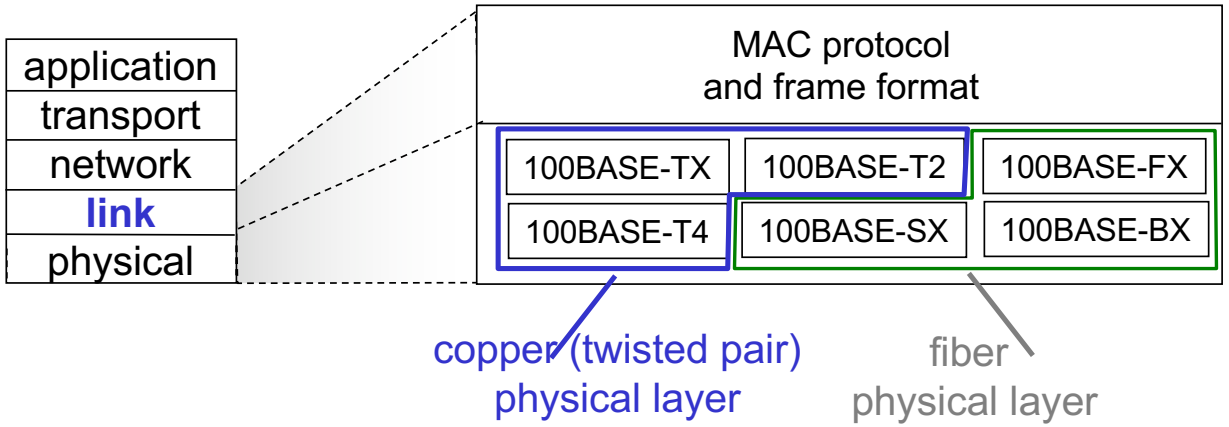
- Worst case B starts transmitting at t=324.
  At time T=324+325 = 649 Bs first bit arrives at A. T= 649 > 576
- **A finishes before it detects that B has transmitted**

The maximum length depends on network speed/bit time delay

IEEE 802.3 ethernet – one common link layer and different physical layers

- Common MAC protocol and frame format

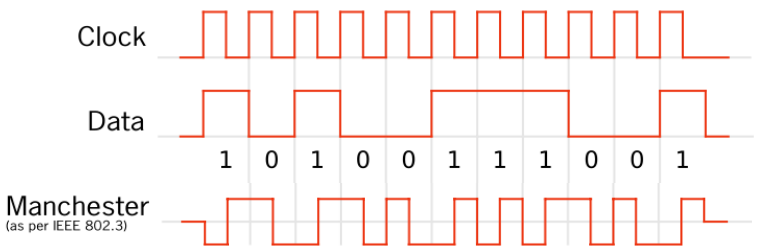
- Different speeds: 2 Mbps, 10 Mbps, 100 Mbps, 1 Gbps, 10 Gbps
- Different physical layer media: copper wire, fiber



- **Baseband transmission**: digital signal directly into broadcast channel

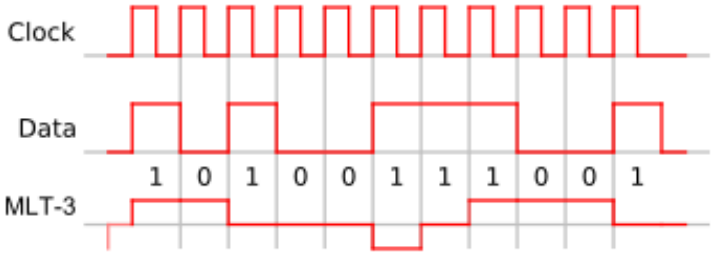
Baseband transmission with line encoding depending on physical layer

Manchester encoding



- 10BaseT
- Each bit has a transition
  - Allows clocks in sending and receiving nodes to synchronize to each other: No need for a centralized, global clock among nodes!

4B5B encoding



- 100BaseT
- 4B5B encoding at least two transitions per block of bits

4 bit value nibble	5 bit value symbol	4 bit value nibble	5 bit value symbol
0000	11110	1000	10010
0001	01001	1001	10011
0010	10100	1010	10110
0011	10101	1011	10111
0100	01010	1100	11010
0101	01011	1101	11011
0110	01110	1110	11100
0111	01111	1111	11101

- MLT-3 encoded: Cycle through -1,0,+1,0 and moves to next stage to transmit a 1

MLT = Multi Level Transmission

A variety of Ethernet media standards, media and speed affect segment distance

10/100/1000 on Copper

Standard	Media	distance	notes
10BASE-T	Cat-3 (2 pair)	100m	
100BASE-T	Cat-5 (2 pair)	100m	
1000BASE-T	Cat-5 (4 pair)	100m	Cat-5e recommended

100 Mbit Fiber

Standard	Media	distance	wavelength
100BASE-FX	62.5µm MMF	2km	1300nm

10 Gigabit Copper

Standard	Media	distance	notes
10GBASE-CX4	Cat-5e	15m	802.3ap
10GBASE-T	Cat-6 Unshielded	55m	802.3an
10GBASE-T	Cat-6 Shielded	100m	802.3an
10GBASE-T	Cat-6a	100m	802.3an
10GBASE-T	Cat-7	100m	802.3an
10GBASE-TSR10	?	< 10m	Low power autonegotiation. Pre-standard
10GBASE-TSR30	?	< 30m	Low power autonegotiation. Pre-standard
SFP+ Direct Attach	twin-ax	1-10m	

Source: Wikipedia

Gigabit Fiber

Standard	Media	distance	wavelength	notes
1000BASE-SX	62.5µm MMF	220m	850nm	
1000BASE-SX	50µm MMF	500m	850nm	
1000BASE-LX	62.5µm MMF	550m	1310nm	requires launch conditioning
1000BASE-LX	50µm MMF	550m	1310nm	requires launch conditioning
1000BASE-LX/LH	SMF	10km	1310nm	typical distance
1000BASE-ZX	SMF	70km	1550nm	not standardized

10 Gigabit Fiber

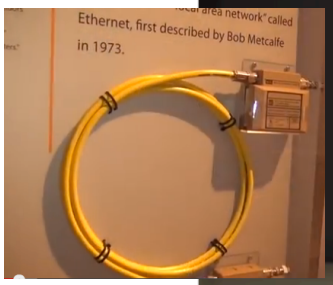
Standard	Media	distance	wavelength	notes
10GBASE-SR	62.5µm MMF	26m-82m	850nm	
10GBASE-LRM	62.5µm MMF	220m	1310nm	
10GBASE-LX4	62.5µm MMF	300m	1300nm CWDM	
10GBASE-SR	50µm OM3	300m	850nm	
10GBASE-LRM	50µm OM3	260m	1310nm	
10GBASE-LX4	50µm OM3	300m	1300nm CWDM	
10GBASE-LR	SMF	10km	1310nm	known to do ~15km
10GBASE-LX4	SMF	10km	1300nm CWDM	
10GBASE-ER	SMF	40km	1550nm	
10GBASE-ZR	SMF	80-120km	1550nm	not standardized, be careful about chromatic dispersion penalty

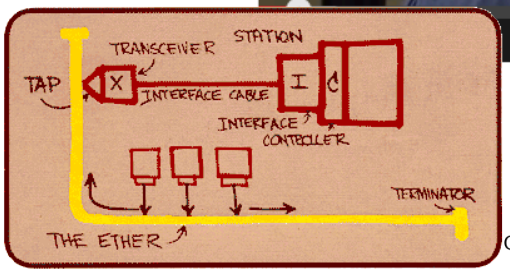
40 Gigabit

40GBASE-CR4	Cat-7?	10m	802.3ba
40GBASE-SR4	OM3 MMF	100m	850nm, 4 strands
40GBASE-SR4	OM4 MMF	125m	850nm, 4 strands
40GBASE-LR4	SMF	10km	(4) 10G ~1310nm waves
40GBASE-FR	SMF	10km	(1) 40G 1550nm wave

Physical layer	40 Gigabit Ethernet	100 Gigabit Ethernet
Backplane		100GBASE-KP4
Improved Backplane	40GBASE-KR4	100GBASE-KR4
7 m over twinax copper cable	40GBASE-CR4	100GBASE-CR10 100GBASE-CR4
30 m over "Cat.8" twisted pair	40GBASE-T	
100 m over OM3 MMF	40GBASE-SR4	100GBASE-SR10 100GBASE-SR4
125 m over OM4 MMF[17]		
2 km over SMF, serial	40GBASE-FR	100GBASE-CWDM4[20]
10 km over SMF	40GBASE-LR4	100GBASE-LR4
40 km over SMF	40GBASE-ER4	100GBASE-ER4

Bob Metcalfe on first Ethernet LAN



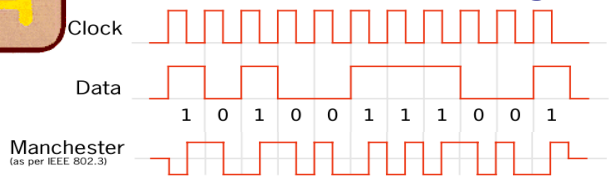


application
transport
network
LINK
physical

Ethernet is going UP (speed), through (WAN), over (wireless), down (embedded microcontrollers being networked), across (metro ethernet bridging LAN and WAN)

"Efficiency depends on the diameter of the network (in bits) , and as you go faster and faster the efficiency goes down"

Manchester encoding



"Diameter of network in bits" t_{prop}

A ⟷ **B**

$$efficiency = \frac{1}{1 + 5t_{prop}/t_{trans}}$$

Links, access networks and LANs
Roadmap

5.1 Introduction and services

5.2 Error detection and correction

5.3 Multiple access protocols

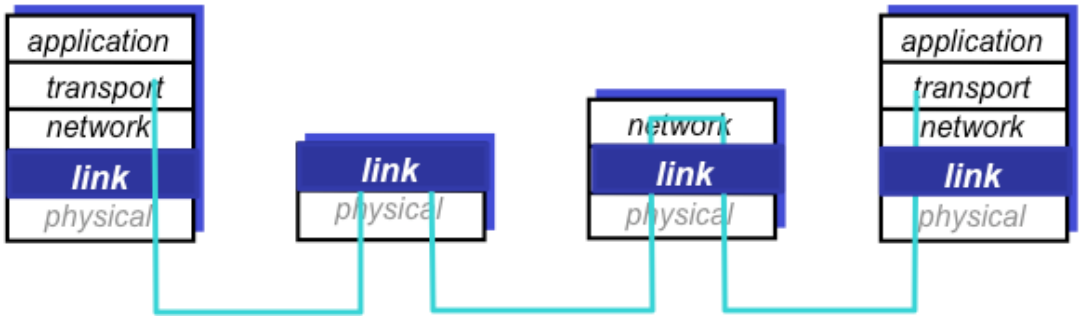
5.4 LANs

- addressing, ARP
- Ethernet
- switches
- ~~VLANS~~

~~5.5 Link virtualization: MPLS~~

~~5.6 Data center networking~~

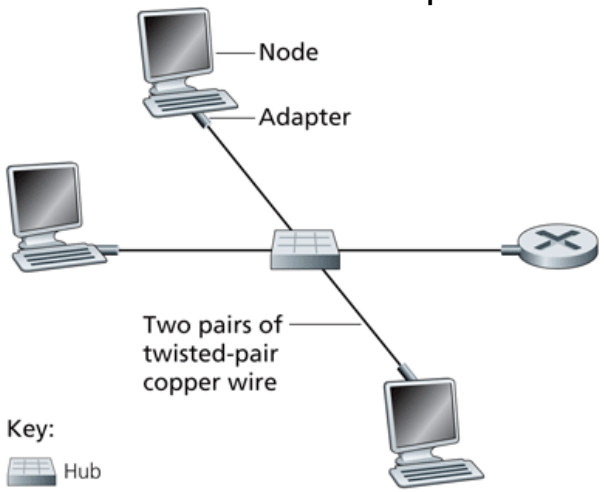
5.7 A day in the life of a web request



Hubs are physical-layer ("dumb") repeaters, link-layer switches are multi-port bridges

Hub

- Bits in on one link out on all other links at same rate
- All nodes connected to hub can collide with one another
- No frame buffering
- No CSMA/CD at hub: host adapters detect collis



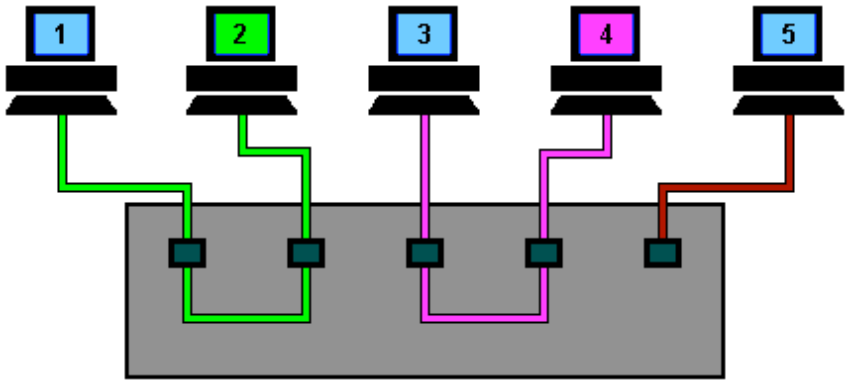
Switch

- **Store and forward** Ethernet frames
- **Examine** incoming **frame's MAC address**
  - selectively forward frame on 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

NTNU

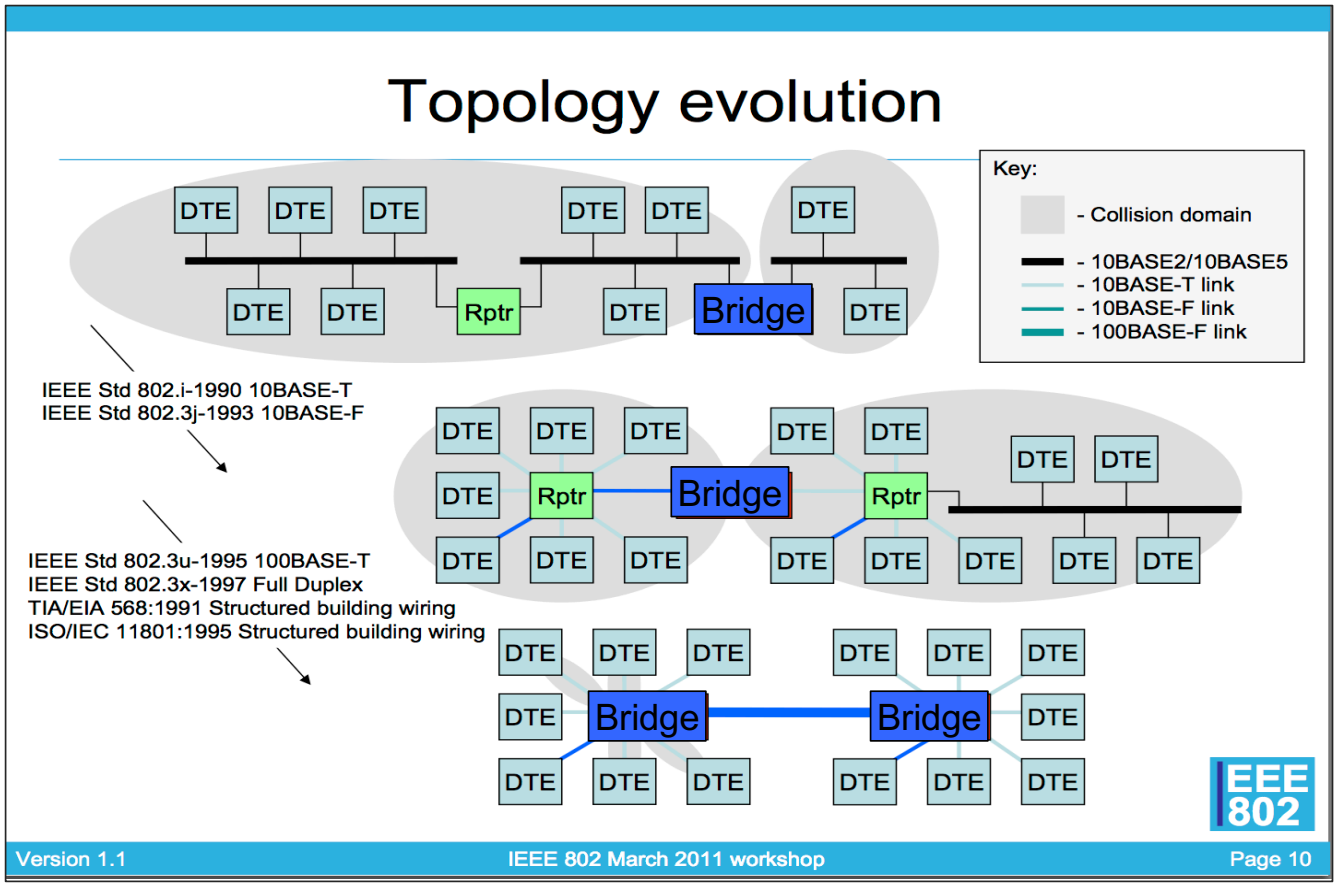
Switch: allows multiple simultaneous transmissions

- Hosts have dedicated, **direct connection** to switch
- Switch **buffers packets**
- Ethernet protocol used on each link, no collisions; full duplex
  - **each link** is its own **collision domain**
- Switching: 1-to-2 and 3-to-4 simultaneously, without collisions
  - not possible with dumb hub



http://www.datacottage.com/nch/eoperation.htm

Ethernet collision domains are getting fewer



Data Terminal Equipment - DTE: These devices are either the source or destination of the data being sent. Devices such as PCs, file servers, print servers and the like fall into this category.

Self-learning, forwarding by flooding or selective send

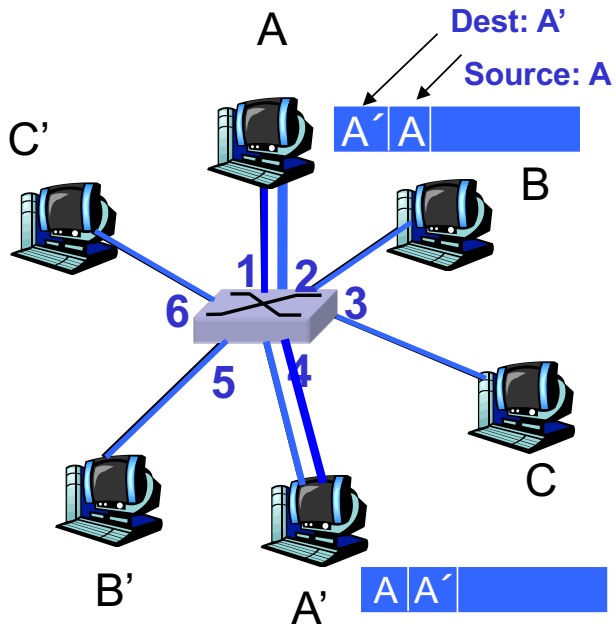
- Switch learns which hosts can be reached through which interfaces

  (MAC address of host,
  interface to reach host, time stamp)
- When frame received, switch records sender/location in switch table
- Frame destination unknown: **flood**
- Destination known: **selective send** on one link

Switch table

MAC addr	interface	TTL
A	1	60
A’	4	60

Time To Live

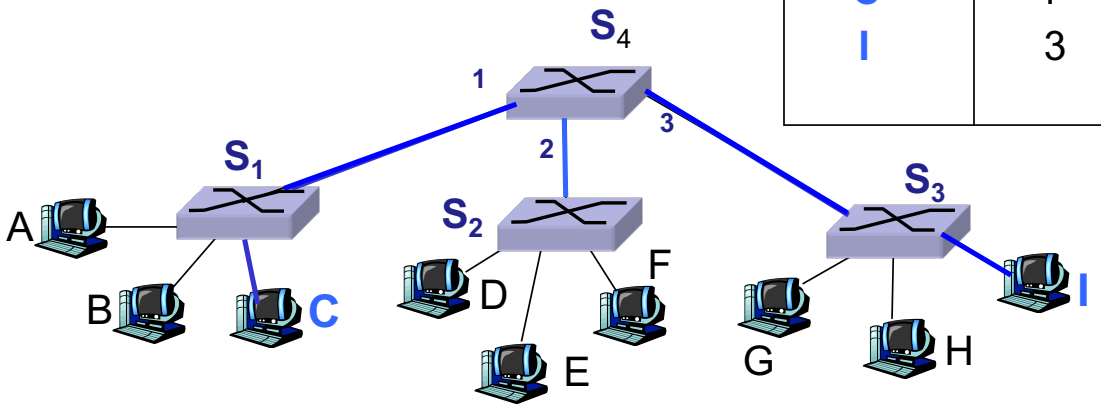


Preamble	Dest. address	Source address	Type	Data, eg IP packet	CRC

Interconnecting switches: plug-and-play through self-learning

Preamble	Dest. address	Source address	Type	Data, eg IP packet	CRC

- C sends frame to I, I responds to C



S_4 MAC	interface	TTL
C	1	60
I	3	60

S_1 MAC	if	TTL
C	c	60
I	1	60

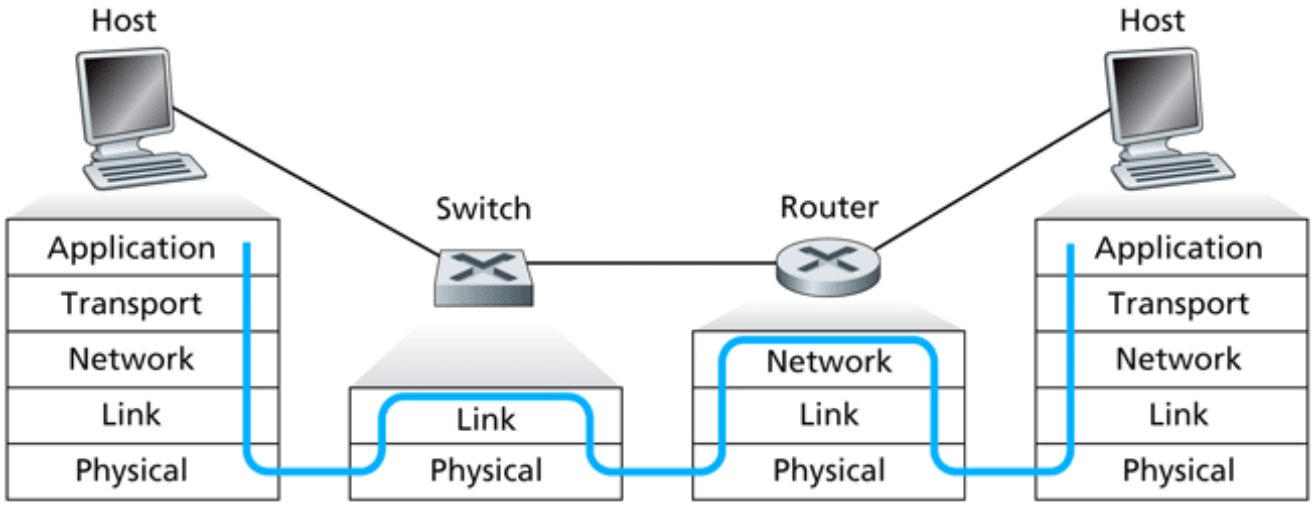
S_2 MAC	if	TTL
C	2	60

S_3 MAC	if	TTL
C	3	60
I	i	60

Switches and routers are both store-and-forward devices

- **Switches:** link layer devices examine **link layer** headers
  - MAC address
- Maintain switch tables, implement filtering, learning algorithms

- **Routers**: network layer devices examine **network layer** headers
  - IP address
- Maintain routing tables, implement routing algorithms



NTNU

Links, access networks and LANs
Roadmap

5.1 Introduction and services

5.2 Error detection and correction

5.3 Multiple access protocols

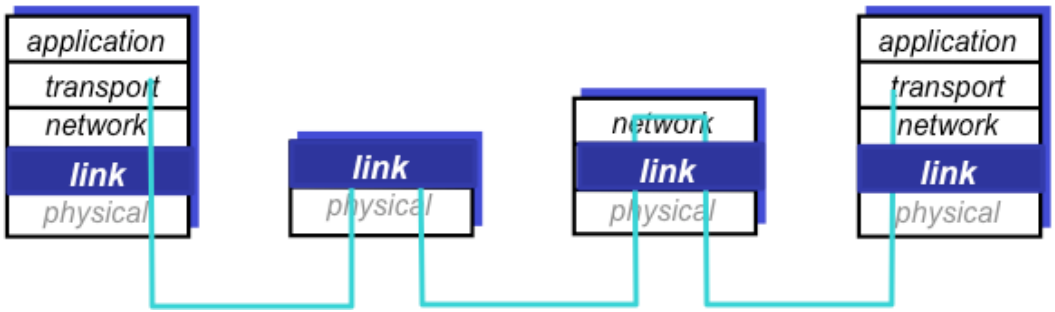
5.4 LANs

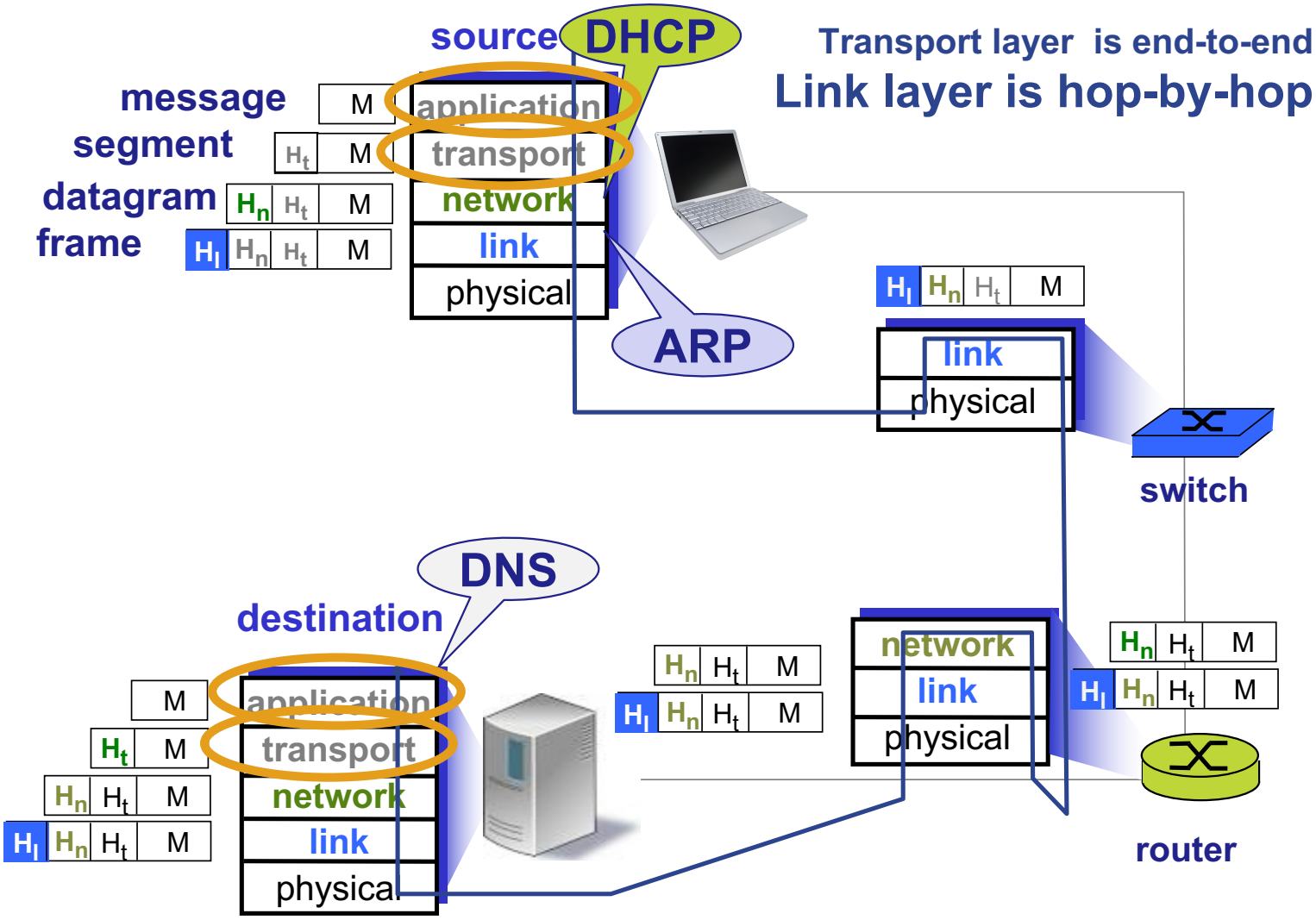
- addressing, ARP
- Ethernet
- switches
- ~~VLANS~~

~~5.5 Link virtualization: MPLS~~

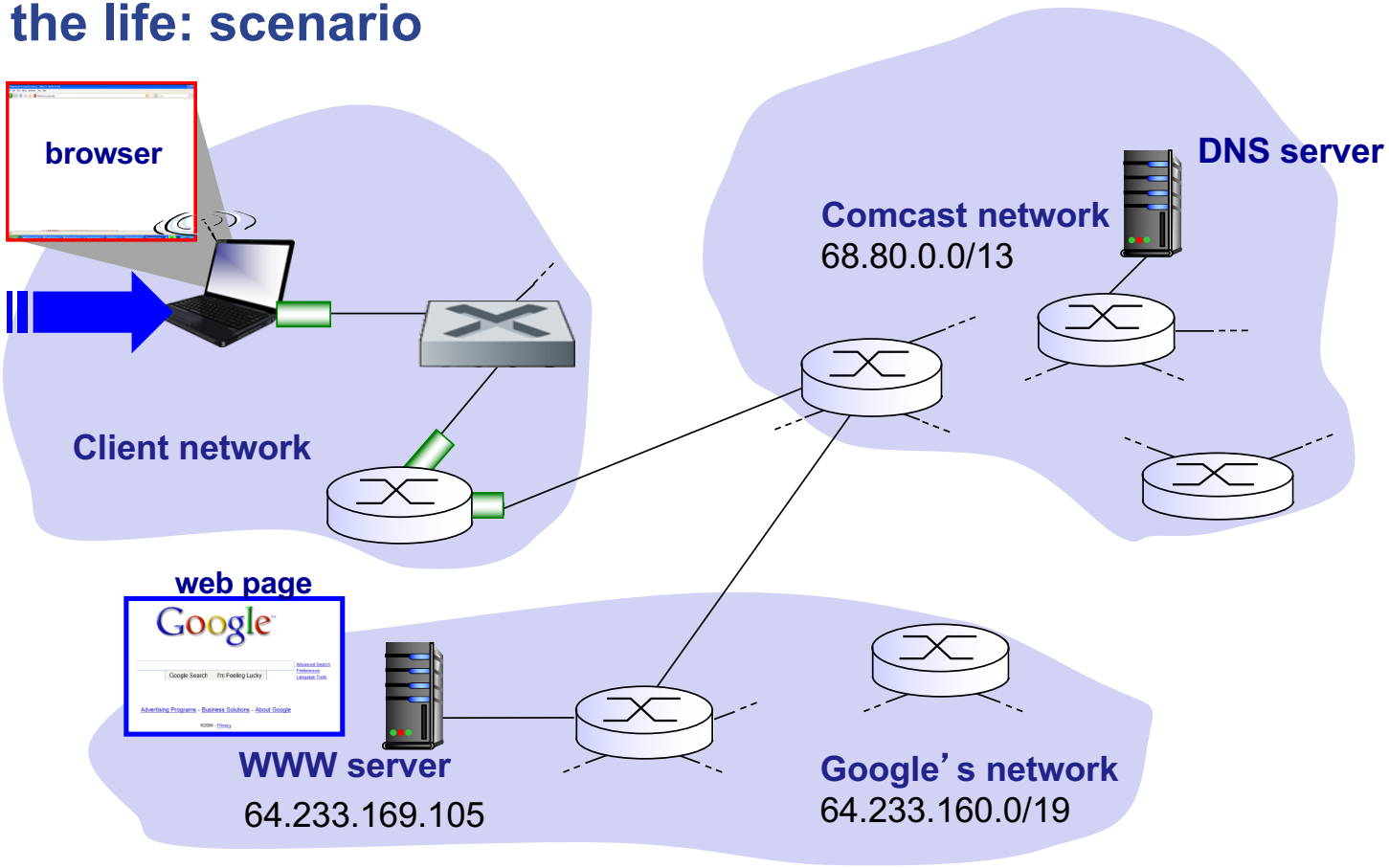
~~5.6 Data center networking~~

5.7 A day in the life of a web request

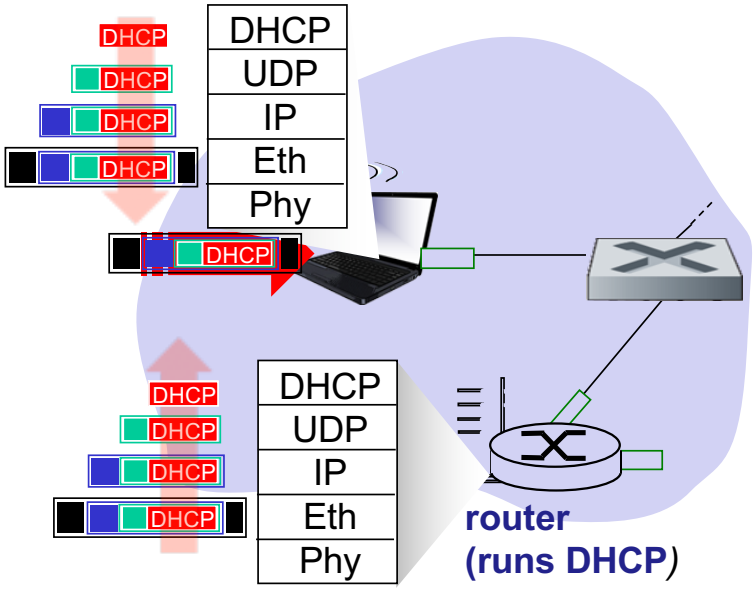



source
DHCP
Transport layer is end-to-end
Link layer is hop-by-hop
message
segment
datagram
frame
M
Ht M
Hn Ht M
Hl Hn Ht M
application
transport
network
link
physical
ARP
Hl Hn Ht M
link
physical
switch
DNS
destination
M
Ht M
Hn Ht M
Hl Hn Ht M
application
transport
network
link
physical
Hn Ht M
Hl Hn Ht M
network
link
physical
Hn Ht M
Hl Hn Ht M
router

A day in the life: scenario

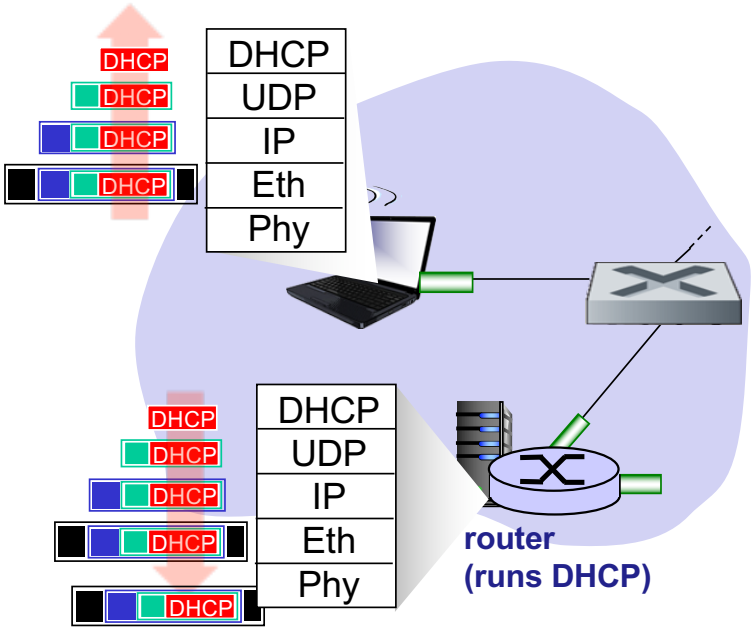


A day in the life … connecting to the Internet



- Connecting laptop needs to get its own IP address, address of first-hop router, address of DNS server: use DHCP
- DHCP request encapsulated in **UDP**, encapsulated in **IP**, encapsulated in **802.3** Ethernet
- Ethernet frame **broadcast** (dest: FFFFFFFFFFFF) on LAN, received at router running **DHCP** server
- Ethernet to IP to UDP to DHCP

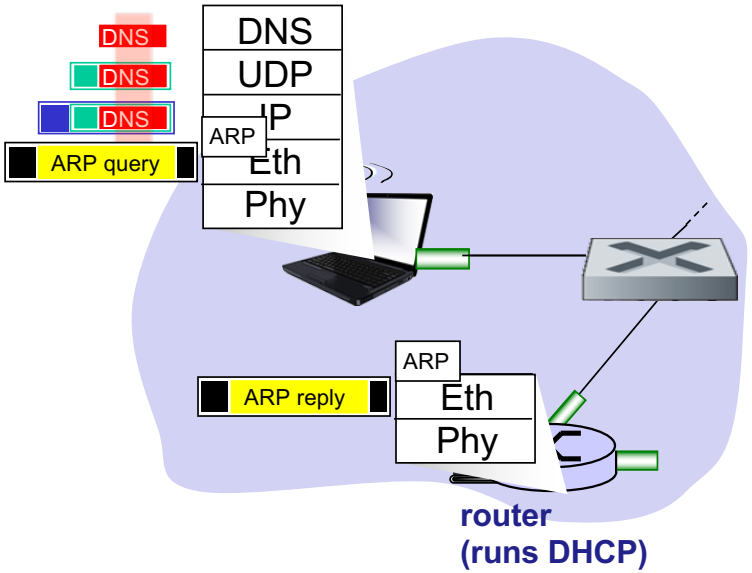
A day in the life … connecting to the Internet



- DHCP server formulates DHCP ACK containing
  - client's IP address
  - IP address of first-hop router for client
  - name & IP address of DNS server
- Encapsulation at DHCP server, frame forwarded (**switch learning**) through LAN, demultiplexing at client
- DHCP client receives **DHCP ACK reply**

Client now has IP address, knows name & address of DNS server and IP address of its first-hop router

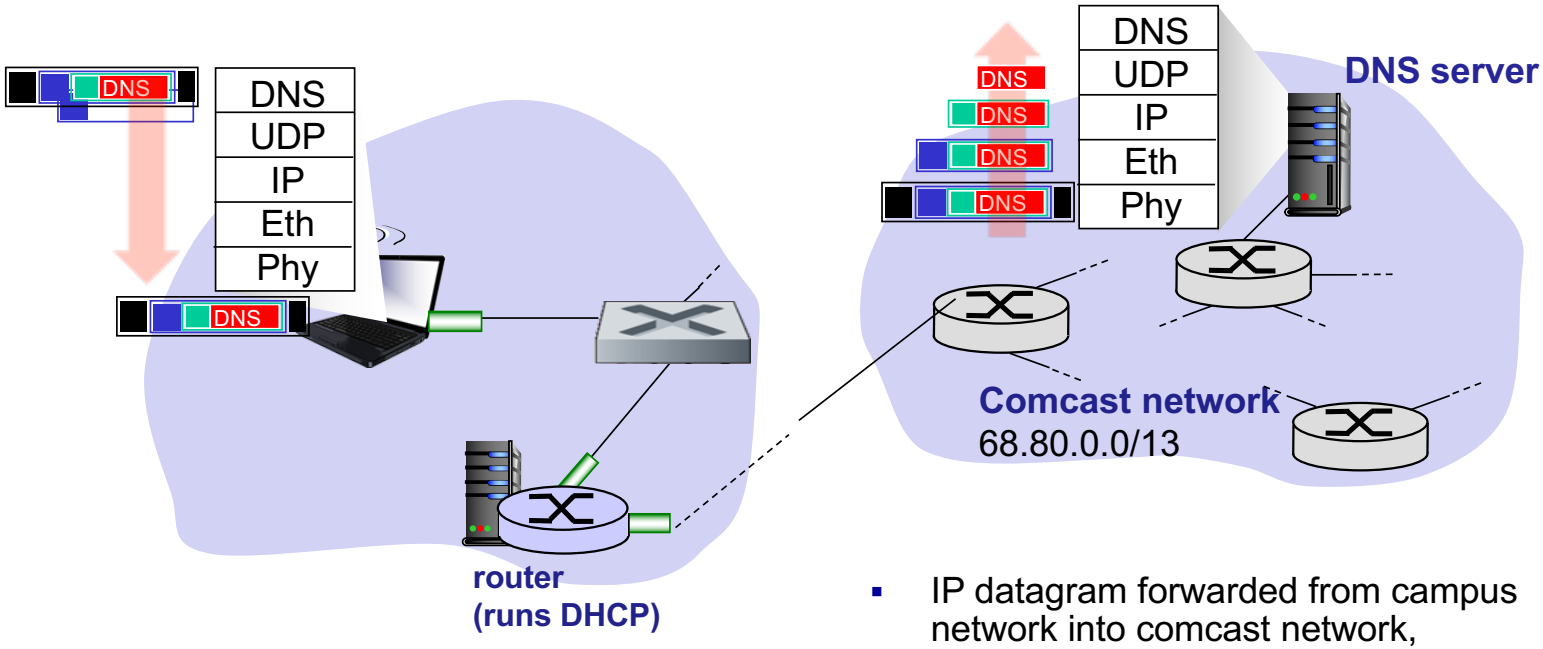
A day in the life … ARP, before DNS and HTTP



- Before sending HTTP request, need IP address of www.google.com: **DNS**
  - DNS query created
- To send DNS request in ethernet frame to first-hop router, MAC address of router interface needed: **ARP**
- **ARP query** broadcast received by router, which replies with **ARP reply** giving MAC address of router interface

When client knows MAC address of first hop router, it can send the frame containing the DNS query

A day in the life … using DNS

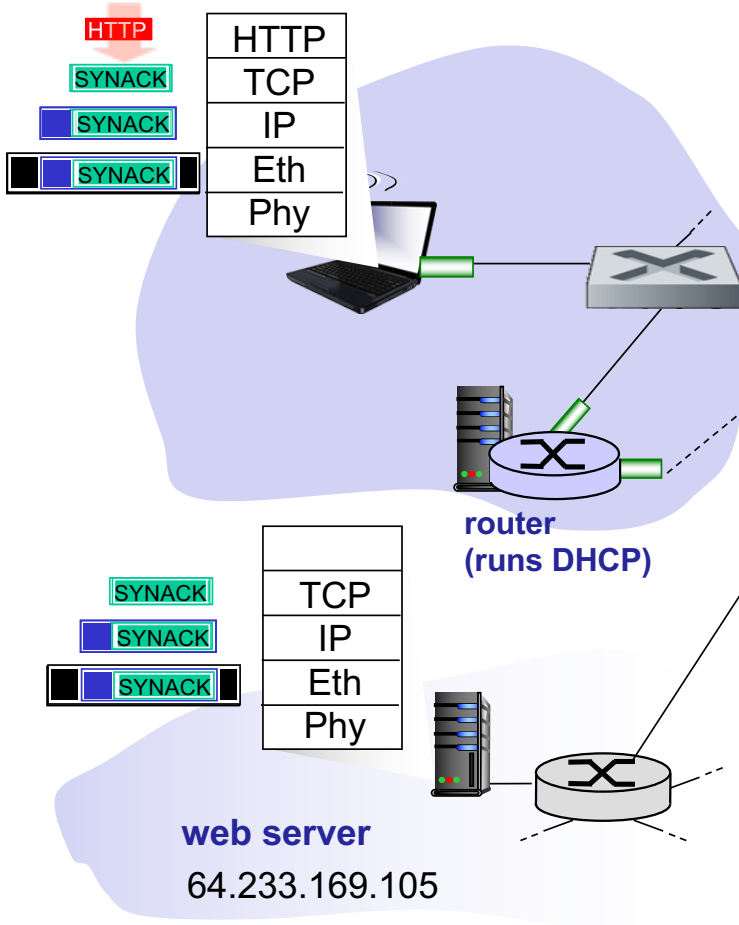


- IP datagram containing DNS query forwarded via LAN switch from client to 1st hop router

- IP datagram forwarded from campus network into comcast network, routed to DNS server (routing tables created by routing protocols)

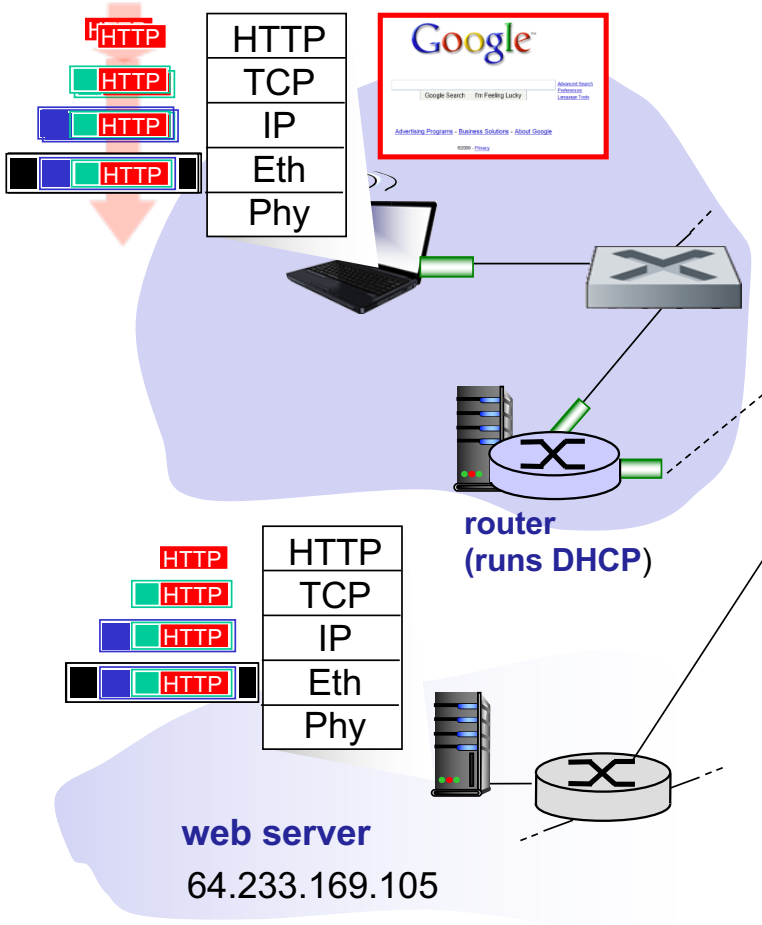
- DNS server replies to client with IP address of www.google.com

A day in the life … TCP connection carrying HTTP



- TCP connection needed to send HTTP request
- TCP **SYN segment** (step 1 in 3-way handshake) inter-domain routed to web server
- Web server responds with **TCP SYNACK** (step 2 in 3-way handshake)
- Client sends ACK
  TCP **connection established!**

A day in the life … HTTP request/reply

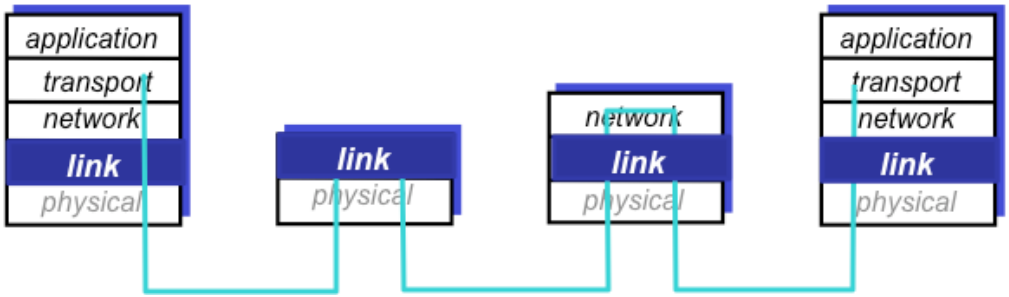


➡ web page **finally (!!!)** displayed

- **HTTP request** sent into TCP socket
- IP datagram containing HTTP request routed to www.google.com
- Web server responds with **HTTP reply** (containing web page)
- IP datagram containing HTTP reply routed back to client

Summary link layer and LAN

- Principles behind data link layer services:
  - error detection, correction
  - sharing a broadcast channel: multiple access
  - link layer addressing
  - **ARP** (address resolution protocol)
- Various link layer technologies
  - Ethernet
  - switched LANS



- Synthesis: a day in the life of a web request

Journey down protocol stack complete