Δίκτυα Υπολογιστών

Πρόγραμμα Μεταπτυχιακών Σπουδών:
Τεχνο – Οικονομικά Συστήματα

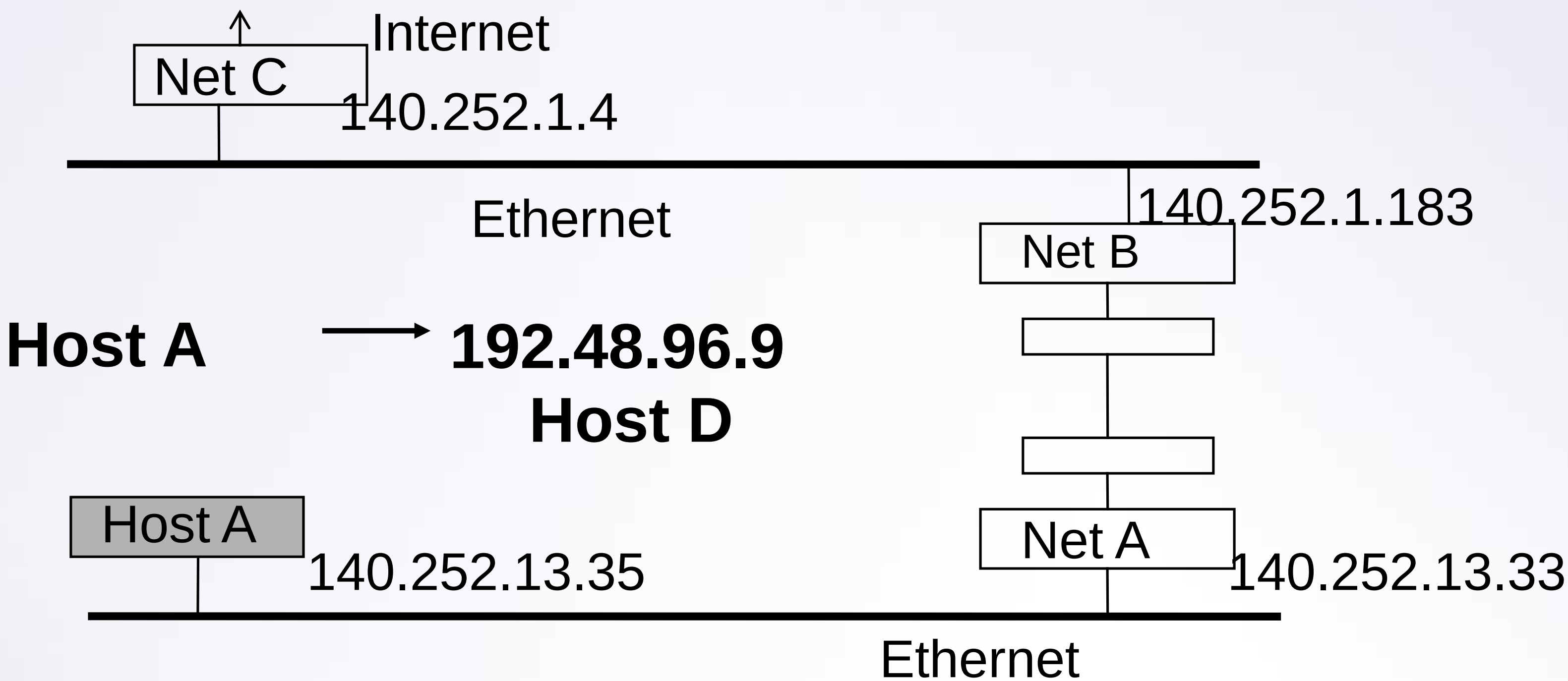
Καθηγητής Συμεών Παπαβασιλείου

Εθνικό Μετσόβιο Πολυτεχνείο
Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Επικοινωνιών, Ηλεκτρονικής & Συστημάτων Πληροφορικής

(E-mail: papavass@mail.ntua.gr   Τηλ: 210 772-2550
Γραφείο: Β.3.15 Νέο Κτίριο Ηλεκτρολόγων)

Βιβλίο Αναφοράς: TCP/IP Illustrated, Vol. I, by W.R. Stevens (Addison Wesley)

Complexity of Networking

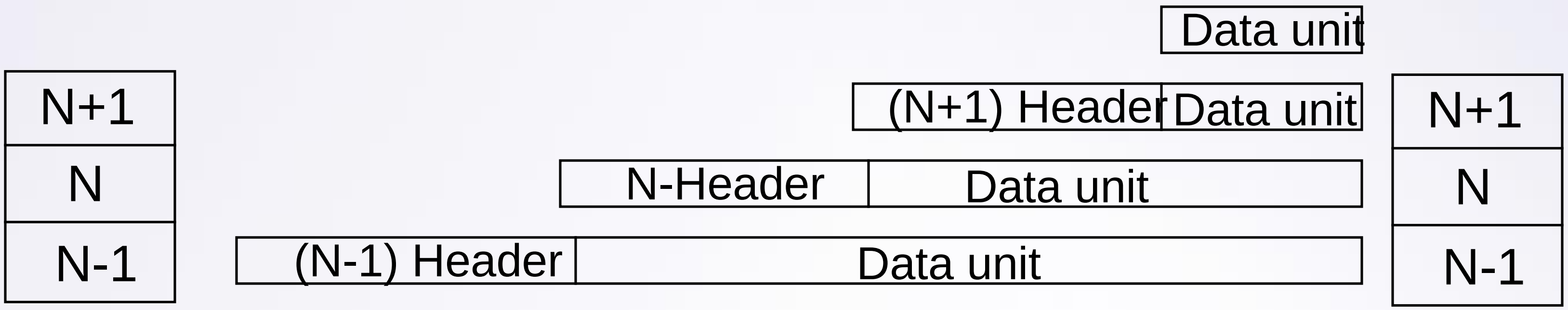


Some Steps:
access database DNS for name resolution
TCP request
TCP--->IP
Directly connected? Different network?
Translate IP address to Hardware address
Perform routing

Peer-to-Peer Communication

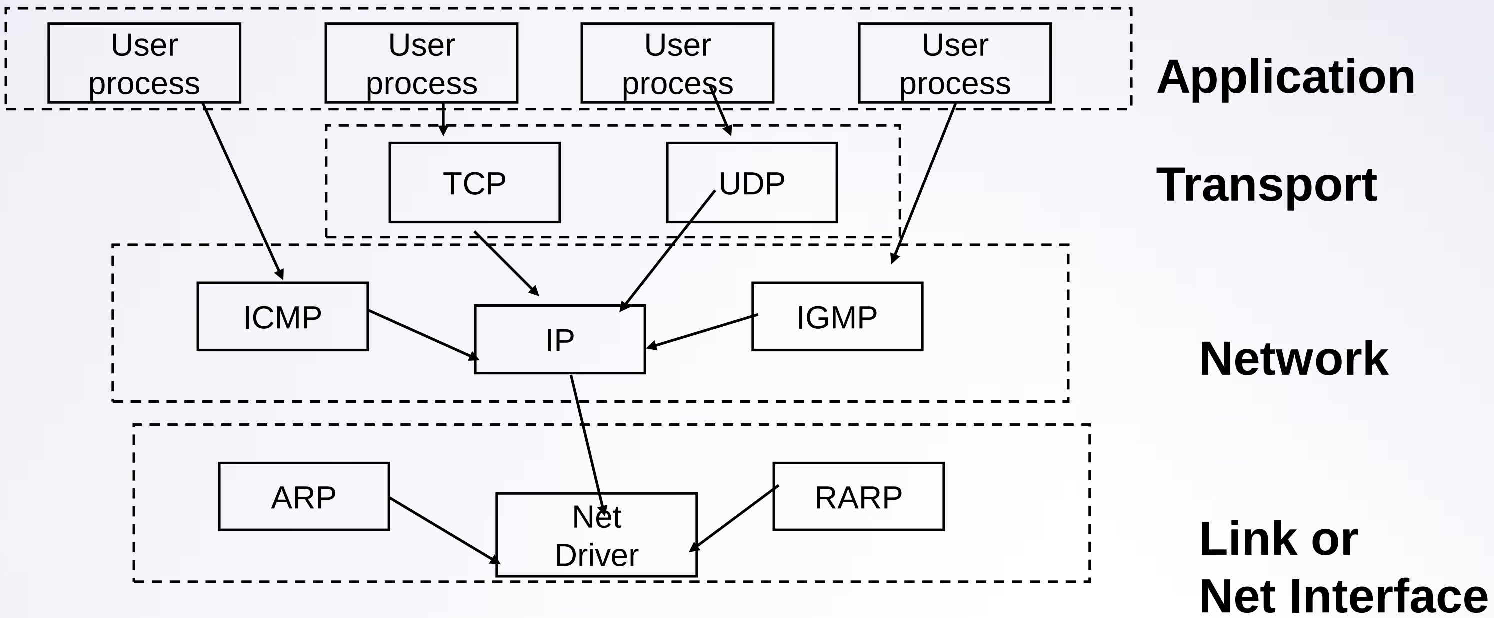
- Each layer uses its own layer protocol to communicate with its peer layer in the other system. Each layer's protocol exchanges information called Protocol Data Unit (PDU) between peer layers.
- This peer-layer communication is achieved by using the services of the layers below it. The layer (N-1) below any current layer (N) provides its services to the current layer (N).
- Ultimately these PDUs must become bits as the data is finally transmitted by the physical-layer protocol using hardware.

Data Encapsulation Process



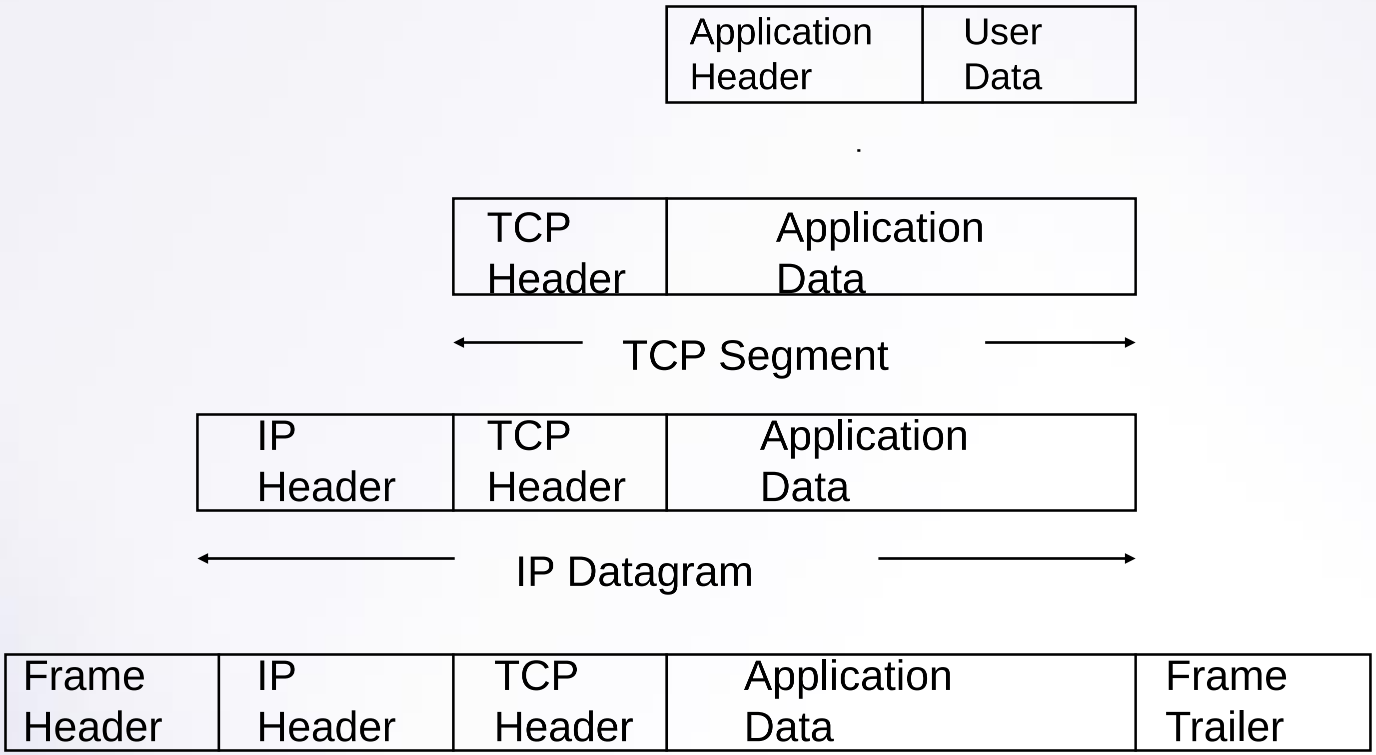
Each layer depends on the service function of the layer below. To provide this service the lower layer uses encapsulation to put the PDU from the upper layer into its data field. Then it can add whatever headers (and trailers) the layer uses to perform its functions.

TCP/IP Layering and Protocols



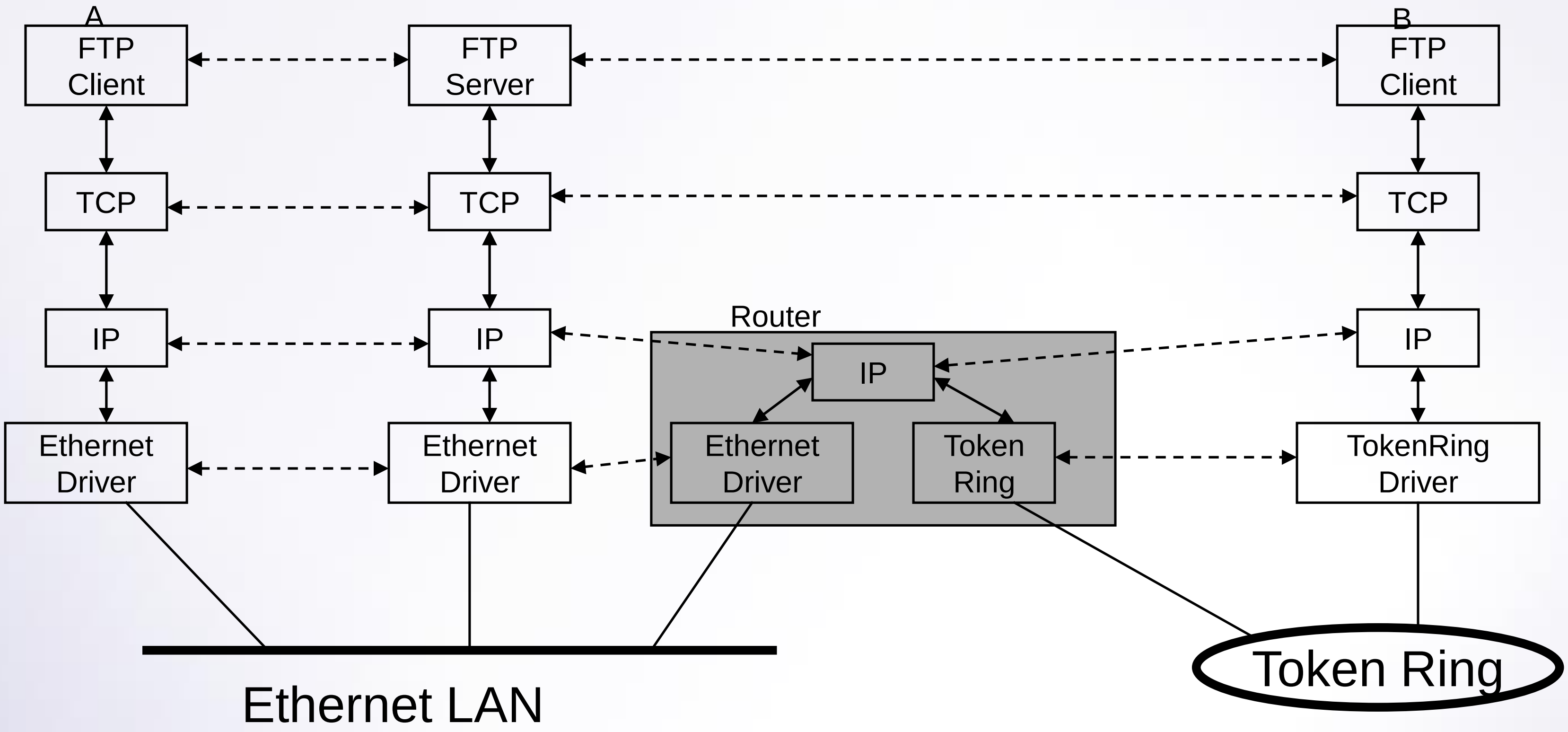
ICMP: Internet Control Message Protocol
IGMP: Internet Group Management Protocol
IP : Internet Protocol
ARP : Address Resolution Protocol
RARP: Reverse Address Resolution Protocol
TCP : Transmission Control Protocol
UDP : User Datagram Protocol

TCP/IP Encapsulation



Examples

A) Two hosts on same LAN (FTP server and FTP client A)
B) Two networks connected by router (FTP server and FTP client B)



Δίκτυα Υπολογιστών

Πρόγραμμα Μεταπτυχιακών Σπουδών:
Τεχνο – Οικονομικά Συστήματα

Logical and Physical Addresses

Physical and Logical Addresses

- Two important types of addresses are link-layer addresses and network layer addresses
- Link-layer addresses (called also physical or hardware addresses) are usually resident in the interface circuitry. Computers with one physical network connection have only a single link-layer address. Link-layer addresses exist at Layer 2 of the OSI reference model and usually exist within a flat addressing space
- Network-layer addresses (also called virtual or logical addresses) exist at Layer 3 of the OSI reference model. They usually have hierarchical structure. Example: mail addresses that describe a person’s location by providing a country, a state, a zip code, a city, a street address, and finally a name.

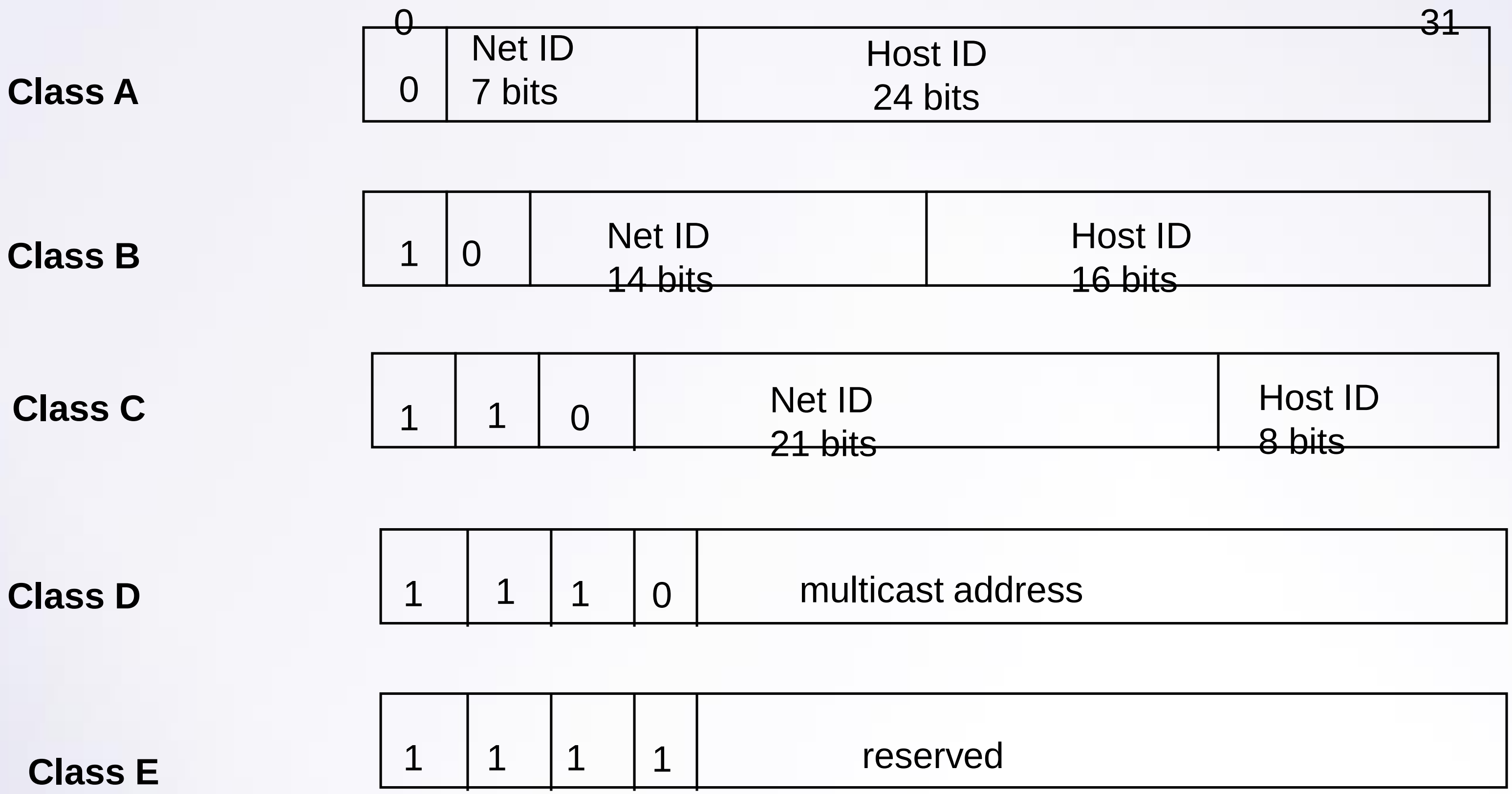
MAC Addresses

- For multiple stations to share the same medium and still uniquely identify each other, the MAC sublayer defines a hardware or data-link layer address called MAC address. The MAC address is unique for each LAN interface.
- MAC address is a 48-bit address usually expressed as 12 hexadecimal digits. The first 6 digits of a MAC address contain the manufacturer identification (vendor code) and the last 6 digits are administered by each vendor and often represent the interface serial number.
- Example: 00000C 123456

IP Addressing

- Every Interface on network has an Internet Address (32-bit number)
- Multi-homed hosts require multiple addresses
- Each address is a pair: one number identifies the network (net ID), and the other gives the location of the host (Host ID) on that particular network
- The high-order address bits determine the address format

IP Address Formats



Addresses normally written in decimal,
one byte at a time

11000000	00001011	00100010	00001001

192.11.34.9

IP Subnet-Addressing

- **Why subnetting?**
  - Too many hosts for a single network (e.g. a single class B address allows for (2^{16}-2) Host IDs).
  - Reduces the size of the Internet's routing tables
- **What is subnetting?**
  - Divide Host ID such that:
  - Host ID=subnet ID + Host ID
- **How subnetting is done?**
  - Using **Mask :** 32-bit value that contains ones for net ID and subnet ID and zeros for Host ID
  - The high-order address bits determine the address format (class)
  - Subnet mask determines the subnet ID and Host ID

IP Addressing Examples

Example 1

Net ID 16 bits	Subnet ID 8 bits	Host ID 8 bits

Subnet Mask: 11111111 11111111 11111111 00000000
OR 255.255.255.0 OR $FFFFFF00_{(16)}$

Example 2

Net ID 16 bits	Subnet ID 10 bits	Host ID 6 bits

Subnet Mask: 11111111 11111111 11111111 11000000
OR 255.255.255.102 OR $FFFFFFC0_{(16)}$

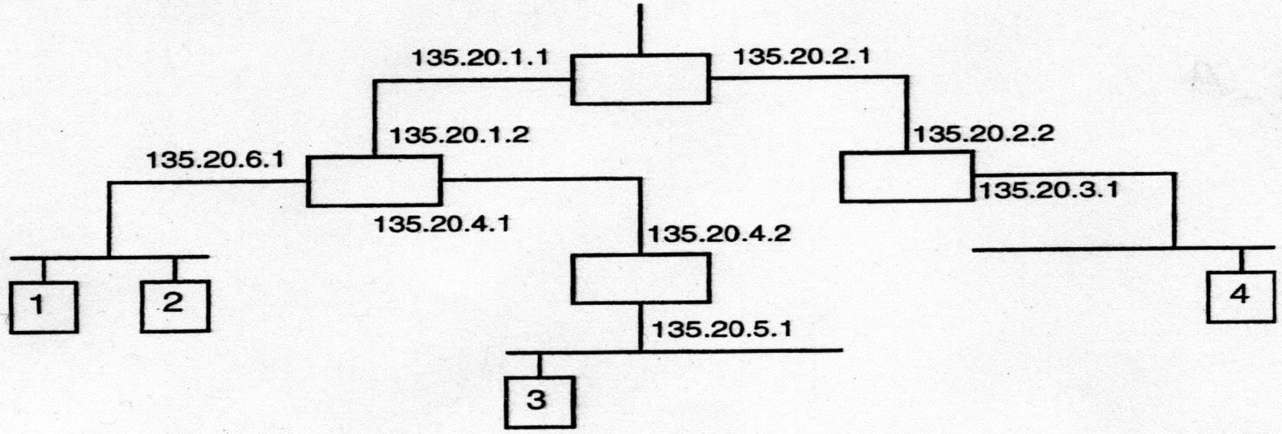
Example 3

IP Address: 140.252.1.22 and Subnet Mask: 255.255.255.0

10001100	11111100	00000001	00010110

Then: Class B address, 8 bits subnet ID
and 8 bits Host ID

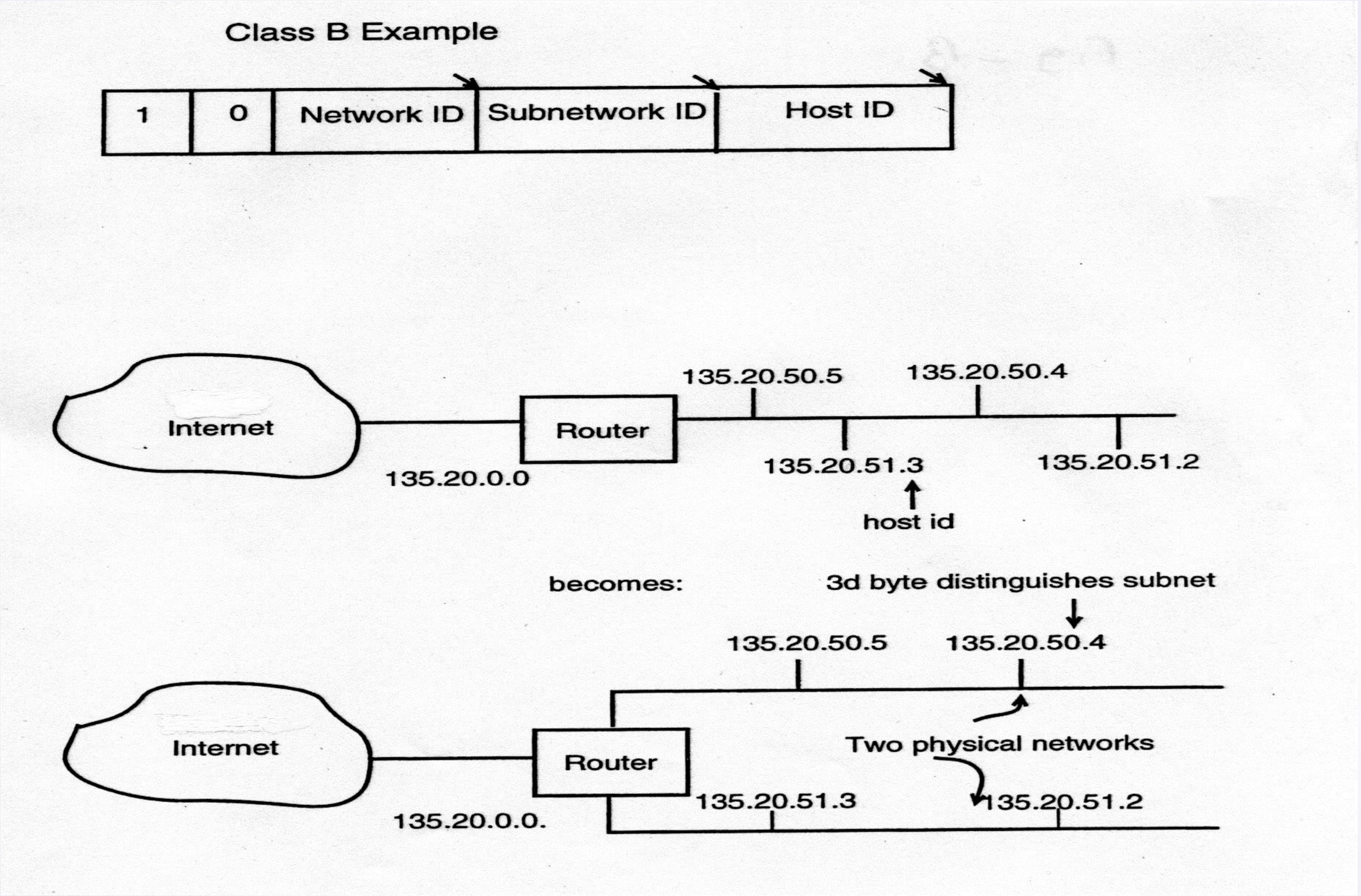
Subnets: Case study



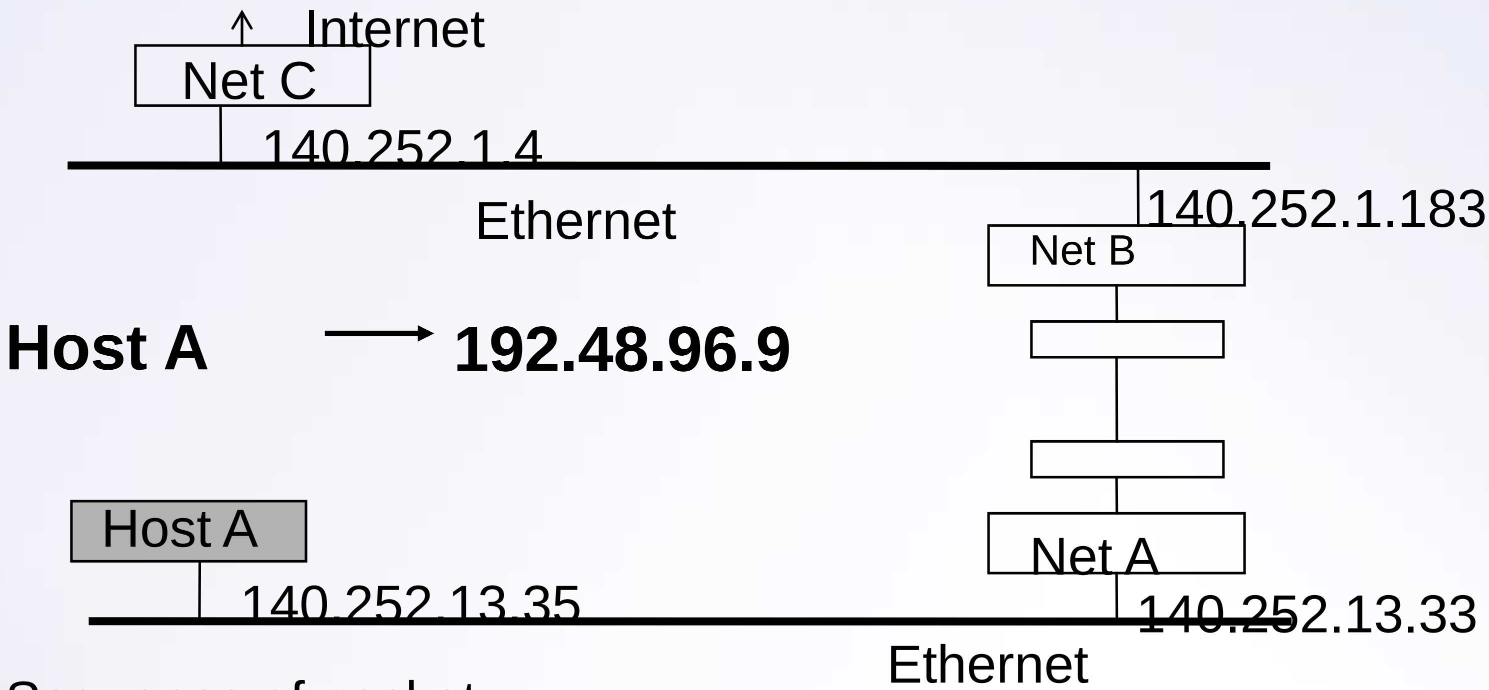
Host	Address	
1	135.20.6.2	
2	135.20.6.3	
3	Possible Address =	________
4	Possible Address =	________

- Class C MASK = 255.255.255.0 = 0xFFFF0000
- Logical AND with IP address
  - 135.20.6.2 MASK (C) = 135.20.6.0
  - network # = 135.20.6

Internet Subnet-Addressing



IP Routing Example



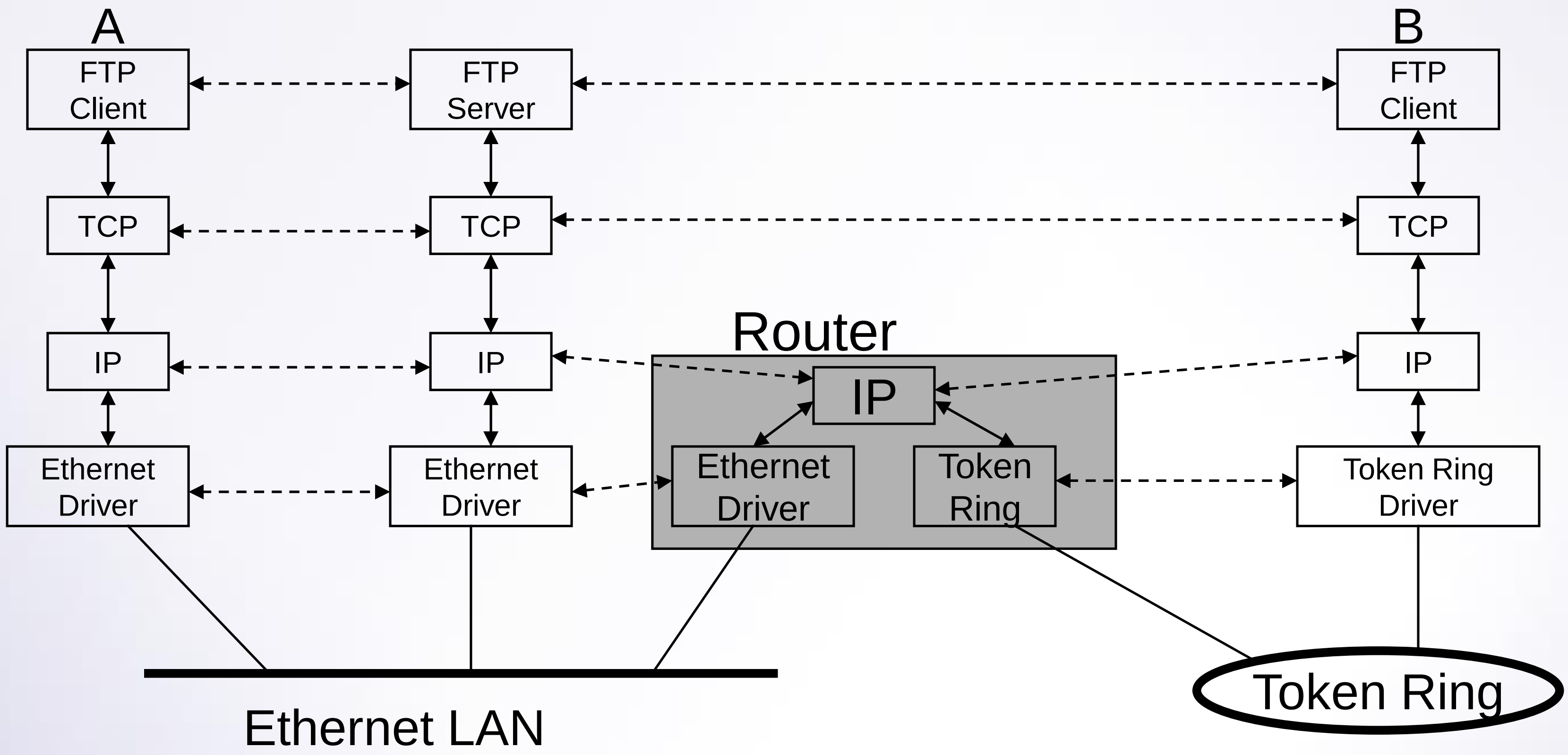
Sequence of packets:

HostA ---> NetA	Dest_IP_addr=192.48.96.9 Dest_Eth_addr=Eth_addr(NetA)
NetA ---> NetB	Dest_IP_addr=192.48.96.9
NetB ---> NetC	Dest_IP_addr=192.48.96.9 Dest_Eth_addr=Eth_addr(NetC)

Examples

A) Two hosts on same LAN (FTP server and FTP client A)

B) Two networks connected by router (FTP server and FTP client B)



Address Resolution Protocol (ARP)

- IP addresses make sense only to TCP/IP
- Data Link Layer has its own addressing scheme (i.e. 48-bit Ethernet address for Ethernet Link layer)
- Therefore it is required:
  - Mapping between the two different forms of addresses
- ARP is used to map from 32-bit IP address to Data Link Layer Address

How ARP Works?

- Step1: ARP sends a frame **“ARP request”** to every host on the network (**broadcast**) containing the IP address of the destination and requesting from the host with that IP address to reply with its hardware address
- Step2: The destination host’s ARP layer receives this broadcast, recognizes that the sender is asking for its IP address and replies (**unicast**) with **“ARP reply”** containing the IP address and the corresponding hardware address
- Step3:The ARP reply is received and the IP datagram that forced the ARP request-reply can be sent

ARP Packet Format

<table>
<tr><td colspan="2">0</td><td colspan="2">31</td></tr>
<tr><td colspan="2">Hardware Type (octets 0-1)</td><td colspan="2">Protocol Type (octets 0-1)</td></tr>
<tr><td>Hard Size</td><td>Prot Size</td><td colspan="2">Operation (octets 0-1)</td></tr>
<tr><td colspan="4">Sender Hard Address (HA) (octets 0-3)</td></tr>
<tr><td colspan="2">Sender HA (octets 4-5)</td><td colspan="2">Sender IP Address (octets 0-1)</td></tr>
<tr><td colspan="2">Sender IP Address (octets 2-3)</td><td colspan="2">Target HA (octets 0-1)</td></tr>
<tr><td colspan="4">Target HA (octets 2-5)</td></tr>
<tr><td colspan="4">Target IP Address (octets 0-3)</td></tr>
</table>

Operation field: 1 ARP request
2 ARP Reply
3 RARP Request
4 RARP Reply

ARP

- **ARP Cache:** ARP maintains a cache on each host that contains recent mapping from Internet Addresses to Hardware Addresses for some pre-specified time (usually 20 minutes)
- **Send out ARP request looking for your own IP address:**
  - If a reply comes this means that another host is configured with same IP address (usually you receive the message “IP address duplicate”)
  - If a host sending this ARP request has just changed its hardware address, then this packet causes any other host on the network that has an entry in its cache for the old hardware address to update its ARP cache accordingly

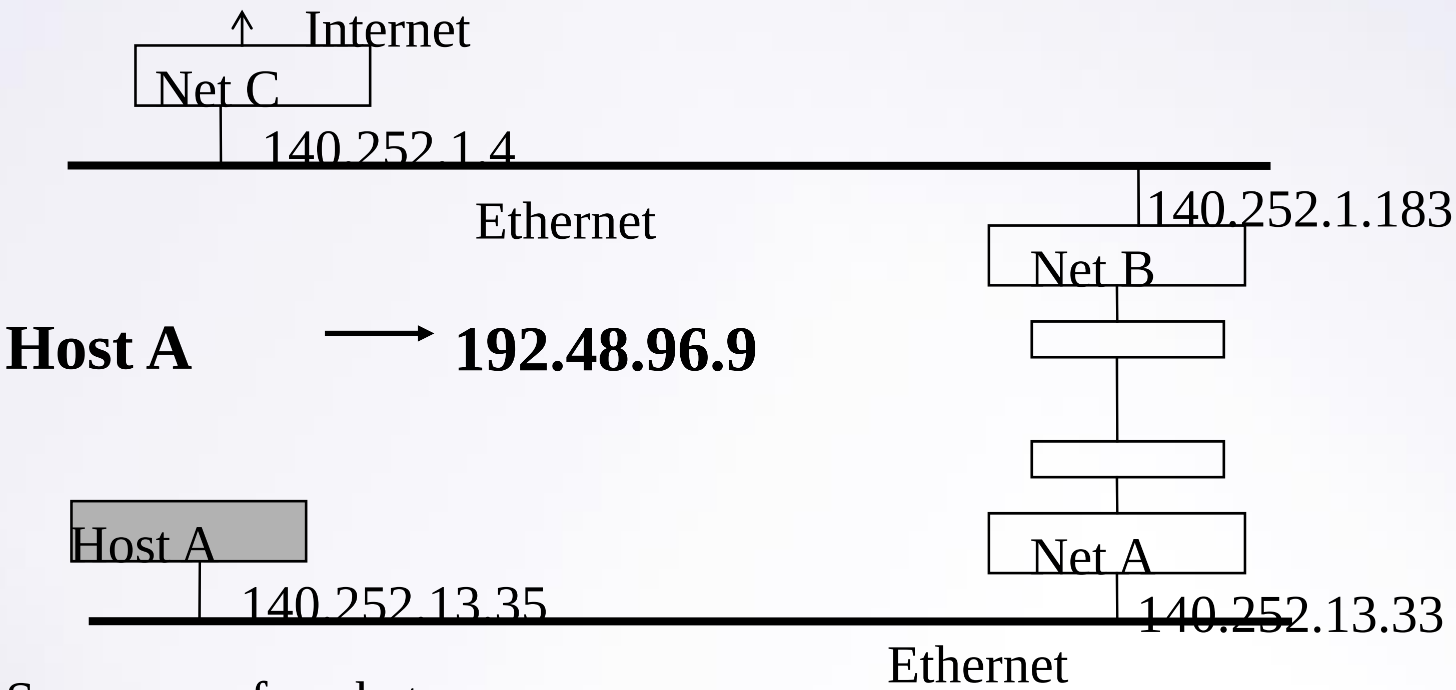
ARP with Duplicated IP Addresses

- Scenario 1: **NodeA_IP_addr=NodeB_IP_addr**
  - Host: IP_addr=Y1 MAC_addr=X1
  - Node A: IP_addr=Y2 MAC_addr=X2
  - Node B: IP_addr=Y2 MAC_addr=X3
  - If Node A connects to Host then Host's ARP cache contains: Y2<-->X2
  - If Node B connects to Host then Host's ARP cache is updated to: Y2<-->X3
  - Eventually traffic from Host destined to Node A goes to Node B that discards it and the traffic is lost

ARP with Duplicated IP Addresses (cont.)

- Scenario 2: **Host_IP_addr=NodeB_IP_addr**
  - Host: IP_addr=Y1 MAC_addr=X1
  - Node A: IP_addr=Y2 MAC_addr=X2
  - Node B: IP_addr=Y1 MAC_addr=X3
  - If Node A tries to connect to Host then both Host and Node B may reply to Node A's ARP request. Depending on timing and implementation Node A keeps in its ARP cache either the MAC address of Host or Node B.
    - If Host's MAC then everything is ok.
    - If Node B's MAC then there is problem
    - If Node B is off then everything looks ok.
  - The problem may go up and down.

ARP Exchange example



Sequence of packets:

HostA ---> NetA	Dest_IP_addr=192.48.96.9 Dest_Eth_addr=Eth_addr(NetA)
NetA ---> NetB	Dest_IP_addr=192.48.96.9
NetB ---> NetC	Dest_IP_addr=192.48.96.9 Dest_Eth_addr=Eth_addr(NetC)