University of Southern California

Viterbi School of Engineering

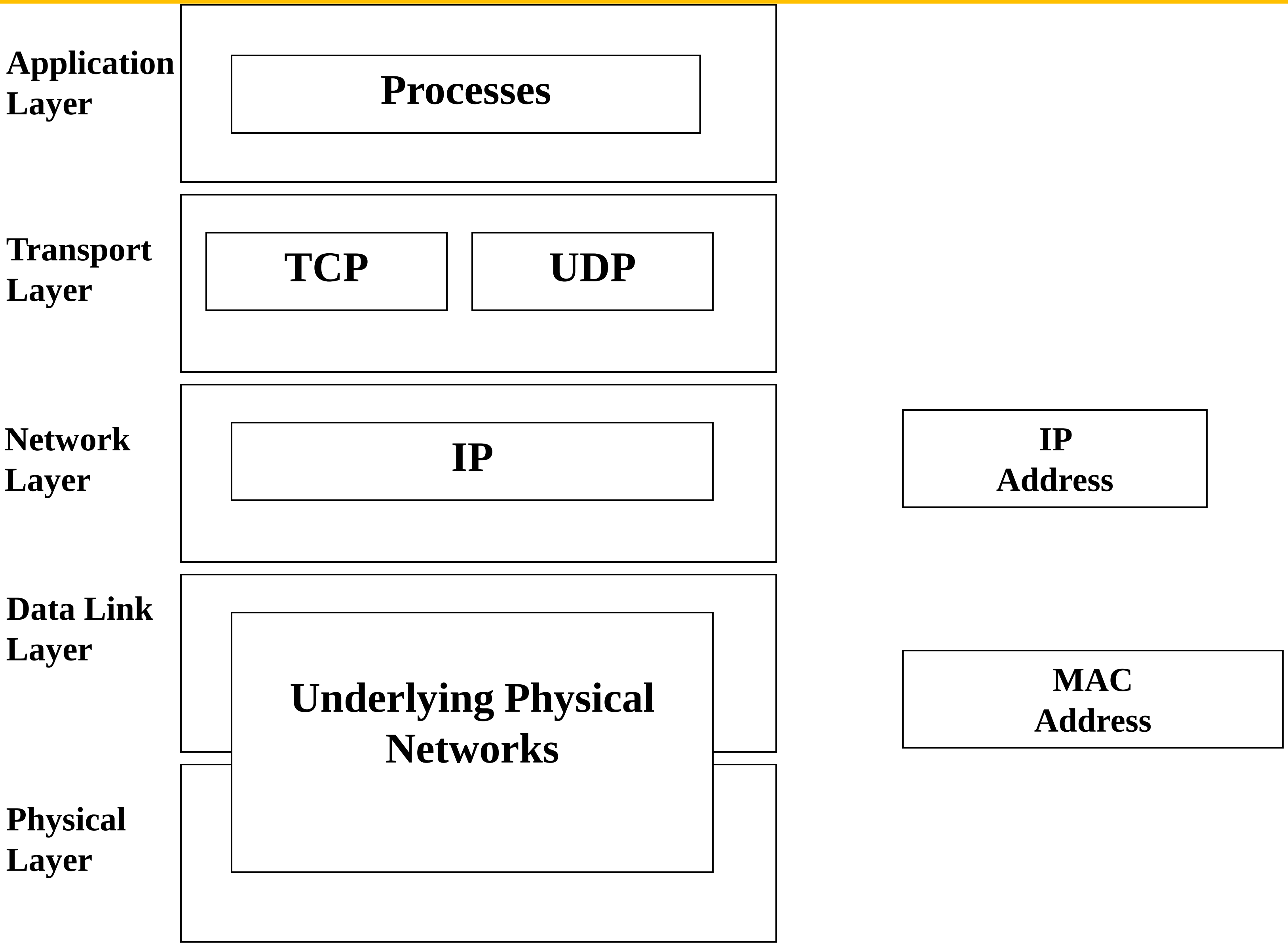
EE450
Computer Networks

Network Addressing
ARP
DNS, and DHCP

Shahin Nazarian

Spring 2013

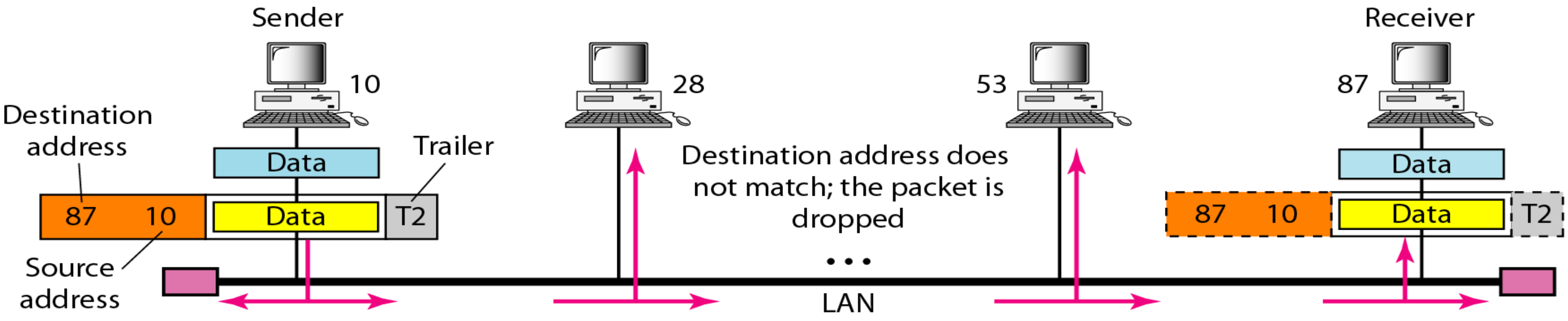
TCP/IP Addressing



TCP/IP Addressing – Physical Address

- 3 levels of addressing are required. The addresses go into the headers

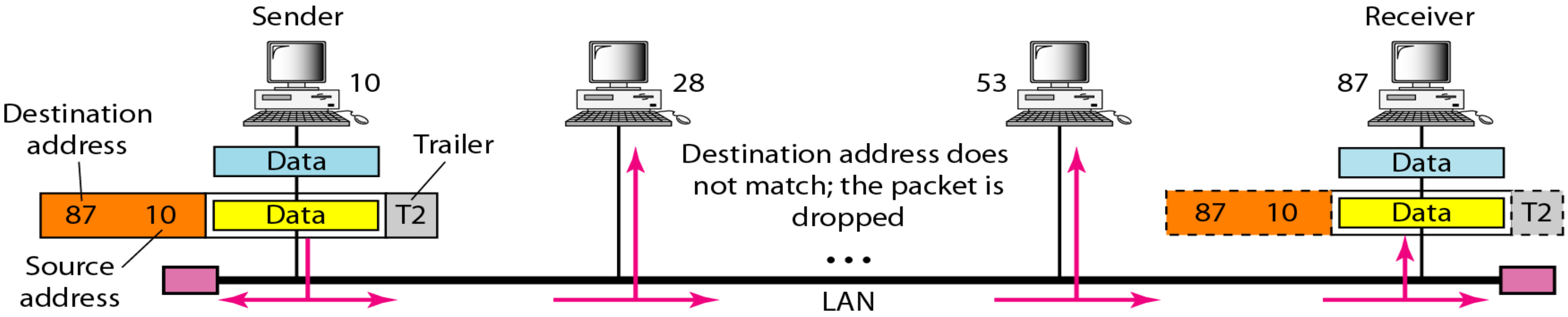
- Physical address (aka DLC address or MAC address)
  - DLC stands for Data Link Control, MAC stands for Medium Access Control



Physical Address (Cont.)

- MAC address is the physical address of NIC

- Example: A node (sender) with physical address 10 sends a frame to a node (receiver) with physical address 87. The two nodes are connected by a link (bus topology LAN)

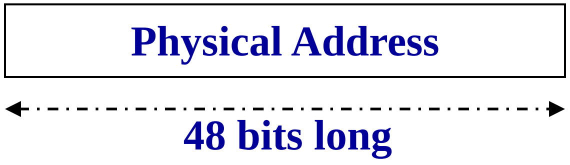


Physical Address (Cont.)

- Reminder: If a computer that is connected to a network needs to be connected to a different network (technology), that computer needs a new NIC [which has a different MAC address], e.g., wireless NIC and Ethernet NICs are different
- If a computer is connected to an Ethernet and needs to be connected to another Ethernet, there is no need to change the NIC, as the NIC can be recognized by the new Ethernet via NIC's global address
- If the user moves his/her computer from one network to another, the IP address of that computer needs to change, because networks have different logical (IP) addresses, however the NIC's MAC address would not change

Physical Address (Cont.)

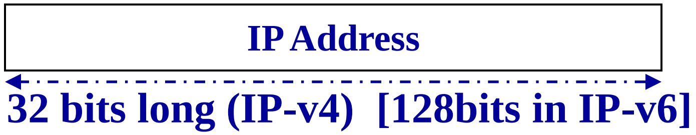
- Note that the physical address or MAC address is in layer 2 or Data Link layer (aka MAC layer); not in the Physical layer
- Question: What header, MAC address goes into? Frame's
- The 48 bits of physical address are divided into six 8-bits (or 6 bytes, or 6 chunks) each of which represented by a two digit hexadecimal number, e.g., 25:A2:17:CF:29:7D
- Incomplete delivery – Example: You mail your friend a letter and it gets to the apartment management office. This is an incomplete delivery. Delivery will be complete when the mail is finally delivered by the office to your friend


Physical Address

48 bits long

TCP/IP Addressing – IP Address

- IP address (aka layer 3 address) goes into the network header (nh) (i.e., packet header) and is for the purpose of packet delivery (which is an incomplete delivery) to the Network layer. IP address is divided into two parts: The network address which is globally unique and the address of the host that is attached to the network
- Netid and hostid
- IP address is configurable (software based,) meaning moving from one network to another, the IP address needs to be changed
- 32 bits in IP are divided into four 8-bits (or 4 bytes or 4 chunks,) e.g., 128.125.30.4 (http://www.ip-db.com/128)



IP Address (Cont.)

- Question: Is the delivery of the packet to the network layer a complete delivery or an incomplete one? It is an incomplete delivery; the message has to be delivered all the way to the host (Application)
- Question: Do all hosts on the same network have to have an identical network address? Yes, but can they have the same host address? No, not at the same time
- The maximum number of IP addresses in IP-v4 is about 4.3 billion, which means we may run out of IP addresses soon
- http://www.computerworld.com/s/article/9191518/Final_IPv4_addresses_to_be_issued_within_months_NRO_warns
- http://www.bgpexpert.com/ianaglobalpool.php

Optional: IPv4 Updates

- IPv4 address exhaustion
  - http://en.wikipedia.org/wiki/IPv4_address_exhaustion
- IPv4 space registry
  - http://www.bgpexpert.com/ianaglobalpool.php
- The status of IPv4 vs IPv6:
  - http://arstechnica.com/business/2013/01/ipv6-takes-one-step-forward-ipv4-two-steps-back-in-2012/

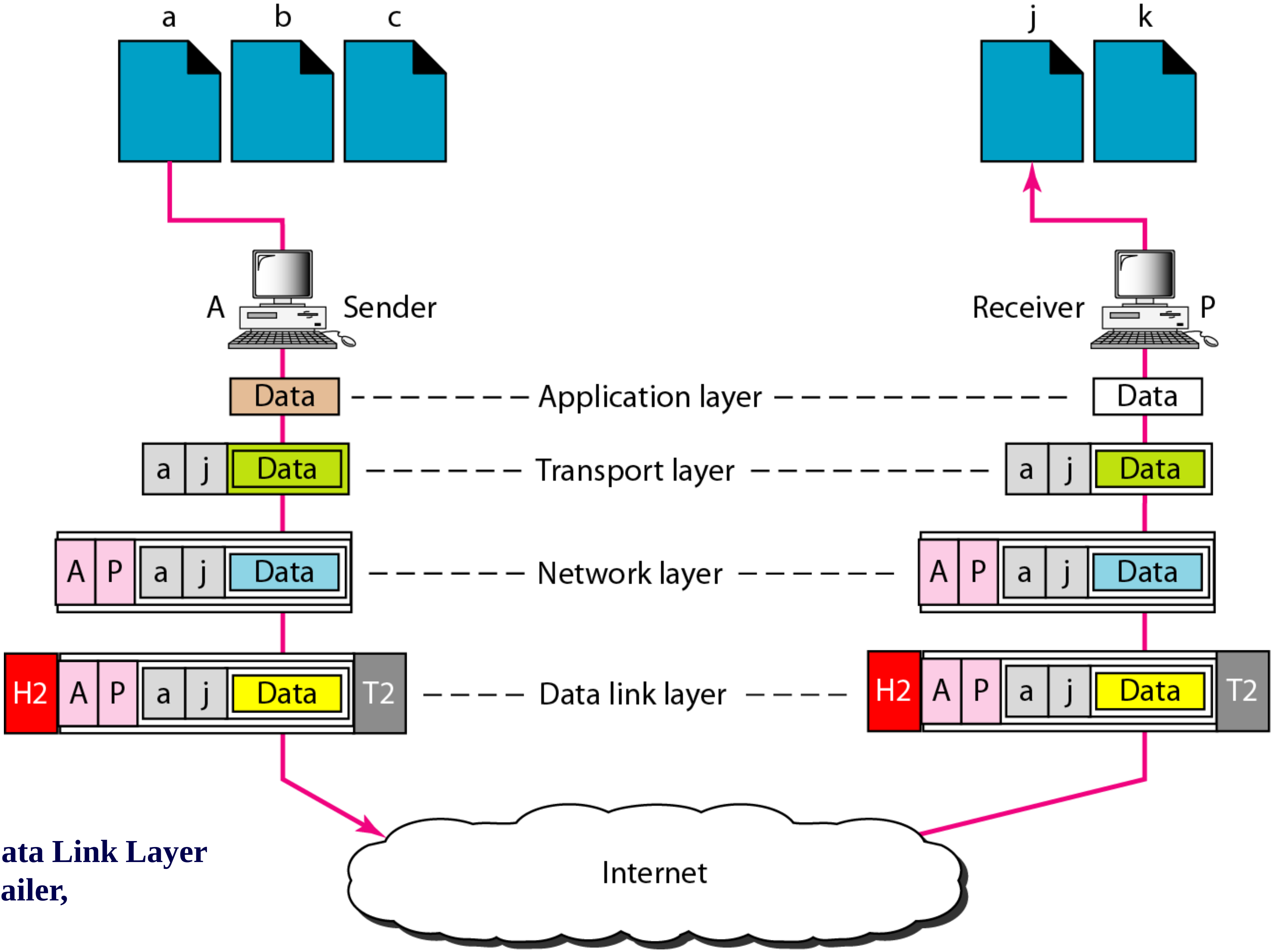
IP Address (Cont.)

- The network address is assigned by Internet Corporation for Assigned Names and Numbers
- In reality ICANN assigns the network address to the ISP, then ISP distributes them among the customers
- The host address is assigned locally by the network administrator
- IP address is hence a globally unique address. This true for a public IP address, so if two IP addresses are found to be the same, the administrators must have made a mistake
- Another class of IP addresses which is called private IP addresses, is not allowed to be used outside the private network

TCP/IP Addressing – Port Address

- It is possible that a host is running multiple applications at the same time. The network layer receives the packets from the sender and needs to deliver the application data to the application layer. How would the network layer know which application to forward the application data to?
- Remember that IP delivery is incomplete, it's just to deliver the packet to the host, but the data needs to be delivered to the right application in the host
- Port address goes into the Transport header (th) [remember that TCP/IP does not have a Session or Presentation layer, so Application layer is right on top of the Transport layer]

Port Address Example



H2 and T2: Data Link Layer header and trailer, respectively

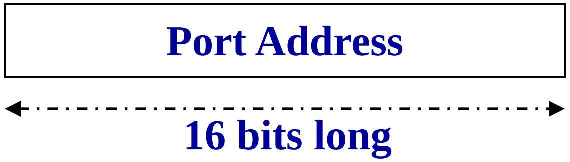
Port Address Example (Cont.)

- In this example, the sending computer is running three processes at this time with port addresses a, b, and c
- The receiving computer is running two processes at this time with port addresses j and k
- Process a in the sending computer needs to communicate with process j in the receiving computer

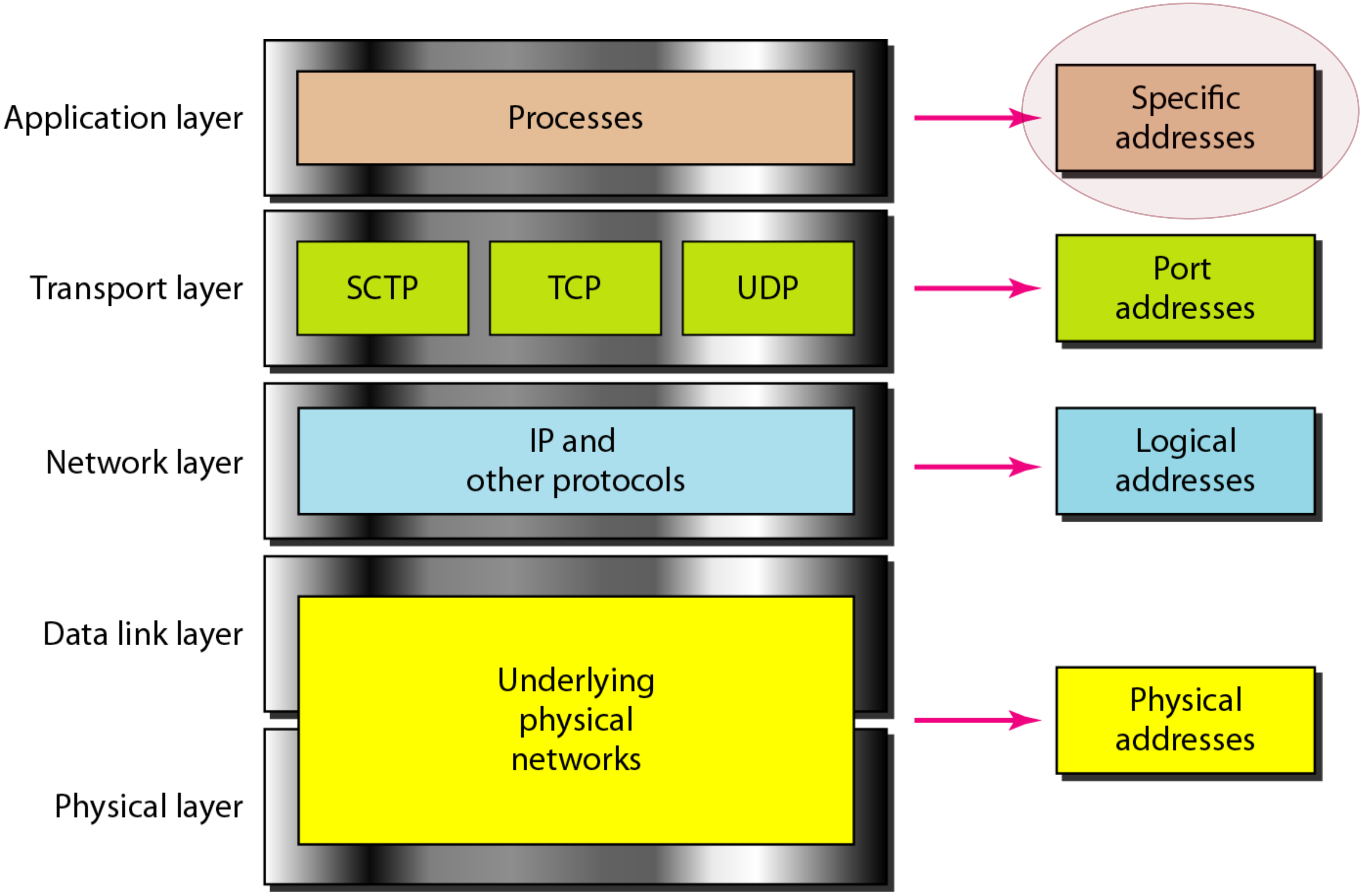
- Note that although physical addresses change from hop to hop, logical and port addresses remain the same from the source to destination

Port Address (Cont.)

- Keep in mind that the port address resides in the header of either the segment or datagram depending on which Transport layer protocol (TCP or UDP) is used for the corresponding application
- The maximum number of port addresses is 2^{16}, which is currently enough for the number of applications, however it is not guaranteed that in the future it would be enough
  - Note that we need one port address for the client application and another for the server application, to be able to indentify the right application on the client machine and as well on the server machine



More General Classification



Specific Address

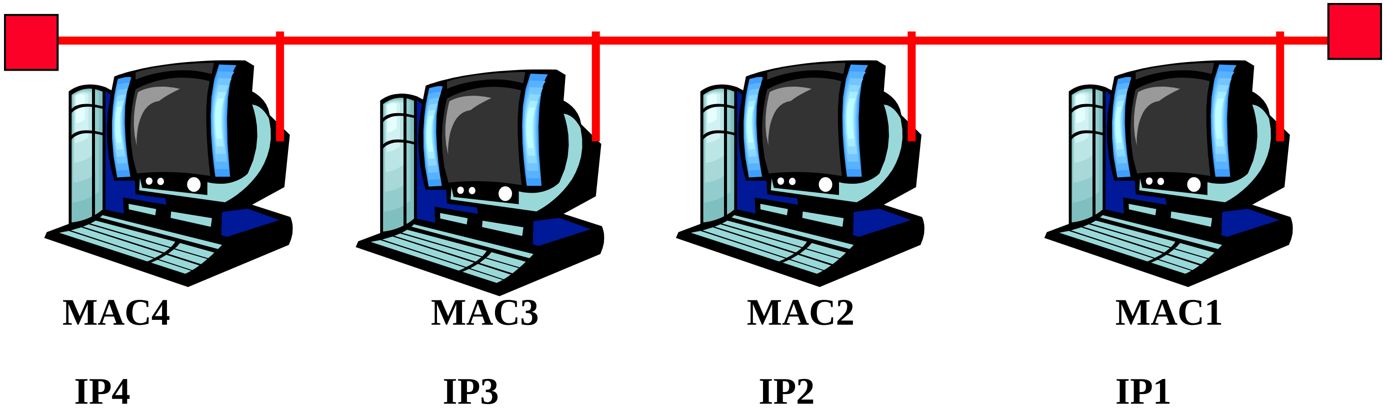
- Some applications have user-friendly addresses, referred to as the specific addresses
- However a specific address will eventually be converted to the corresponding port and IP address by the sending computer
- Examples of specific addresses:
  - Email addresses such as trojan@usc.edu which define the recipient of an email
  - URL (Universal Resource Locator) such as www.usc.edu which is used to find a document on the WWW (World Wide Web)

TCP/IP Addressing (Cont.)

- The port number for the application running in the server machine is well known as the port number (hence referred to as the well-known port number) The client port number is typically chosen randomly by the network operating system
- Note that using the port number, the application data can be delivered to the right application, this is referred to as a complete delivery
- The IP address of the source and that of the destination both are put in the network header (nh) , however as far as the delivery is concerned, the network does not care about the source IP address

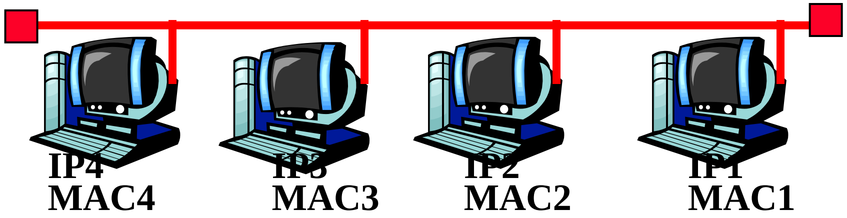
Network Communication

- In this diagram MAC1 to MAC4 are the 48 bit (6 byte) MAC addresses of 4 end systems E1 to E4 connected to each other by a bus (their IP addresses are IP1 to IP4)
- There is no router, therefore the whole thing is considered as only one network (Like a LAN with Ethernet)



Network Communication (Cont.)

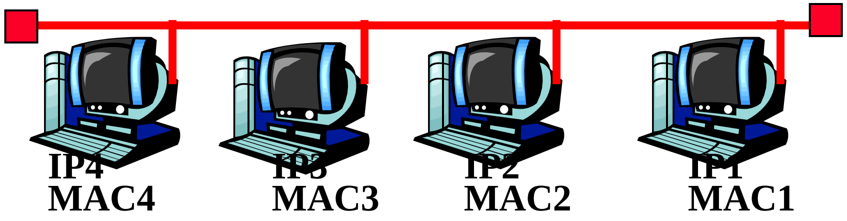
- Assume host E1 with MAC address MAC1 (source) sends a message to host E3 (with MAC address MAC3). At the Data Link layer the frame header will have those MAC addresses
- Question: What is the first piece of information the source E1 needs to know about the destination E3? E3's MAC address or it's IP address? And what is the first task the source needs to perform before sending the message? It needs the destination's IP address, IP3 to find out if the destination is on the same network or on a different one



dt	Payload	______ ______

Network Communication (Cont.)

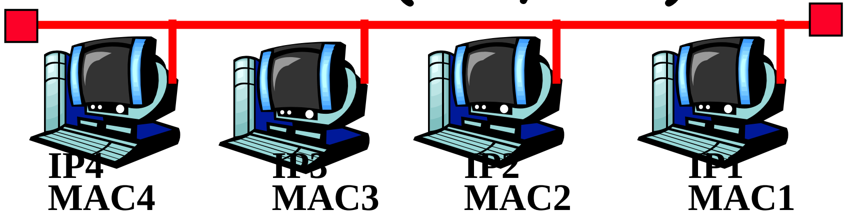
- Looking at the destination's IP address and the part that has the network address, it compares it with its own network address to find out if they are on the same network or not
- To get the IP address of the destination, the source needs to know something about the destination? That is the host name for the destination host, e.g., www.usc.edu
- However routers understand only IP addresses and not host names, so a mapping is required that maps host names to their IP addresses



dt	Payload	MAC1 MAC3

Network Communication (Cont.)

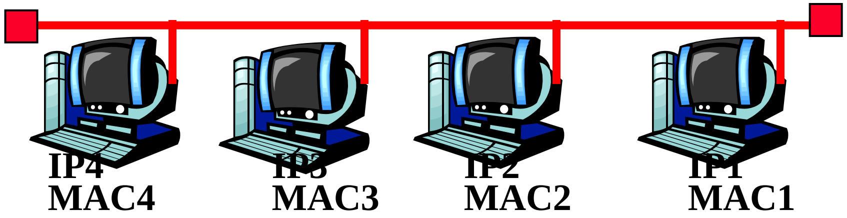
- Question: Routers do not recognize host names. They are designed to understand IP addresses. Wouldn't it be more efficient if the source could memorize the IP address of the destination and then "host name to IP mapping" would not be required? Not really, Host names are easy to memorize, but IP addresses are not
- When the source E1 enters the host name in the browser, the DNS (Domain Name System) of source as the client will contact the DNS server machine and requests the IP address of the destination. DNS server machine has a directory of host names mapped to their IP addresses
- DNS server responds by sending the IP address of the destination (i.e., IP3) to the client machine



dt	Payload	MAC1 MAC3

Network Communication (Cont.)

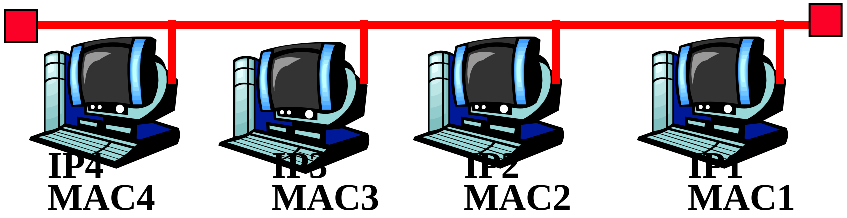
- Subnet masking is done next. Here, the source realizes they are on the same network, so it does not need the help of a router. Remember that this is happening in the 3rd layer
- Now source creates the packet: the source and destination IP addresses (IP1, and IP3, respectively) are put in the packet then passed to the 2nd layer to create the frame
- The 2nd layer (of the source) needs the MAC address of the destination to put it in the header of the frame
- Question: why is the source IP address included in the nh, when we know the network does not need it?
  - The reason is that the destination needs the source IP address, for example to reply



dt	Payload	_____ _____

Network Communication (Cont.)

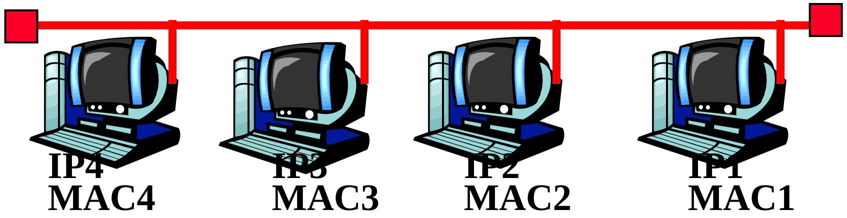
- Question: How does the source know the MAC address of the destination?
  - Through a protocol called Address Resolution Protocol. It's a protocol that maps the IP address to a MAC address
- ARP module has a table, with two columns: the IP address and the MAC address
- So in the header of the frame it puts MAC1 of itself and MAC3 found through ARP
- Remember that the IP addresses are in Data Link Layer's Data part, but here they do not play a role as far as the delivery is concerned



dt	Payload	MAC1 MAC3

Network Communication (Cont.)

- This is a shared network, the bus terminator (or bus end or cable end) will absorb the frame and will not reflect the message back to the bus
- Every NIC card (not the computer or the end user) will receive this frame. For example the E2's NIC (i.e., the one with MAC address MAC2) will receive it: first its physical layer will receive the frame, not knowing what the bits mean; it then gives them to its 2nd layer to create the frame; 2nd layer looks at the header of the frame and sees MAC3, compares it with its own MAC2, and they don't match, so the frame is not destined to it. The NIC card disregards the frame. This means it won't process it any further, meaning E2's upper layers won't receive any thing

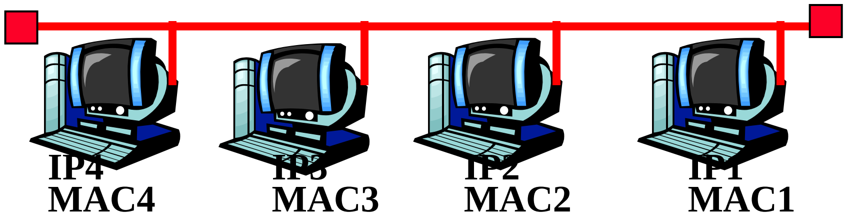


dt	Payload	MAC1 MAC3

Network Communication (Cont.)

- E3 sees in its 2nd layer that the destination MAC address matches its MAC address. The 2nd layer removes the header and trailer after it has performed the functionality and gives the payload (the packet) to its 3rd layer
- At this point E3 knows which end system sent the message. If it wants to reply, the packet of the reply message has a change as follows: the source IP address, IP1 and the destination's IP address, IP3 are swapped. Also in the frame, MAC1 and MAC3 are swapped

Frame (message) from E1 to E3	dt	Payload	MAC1 MAC3
Frame (reply) from E3 to E1	dt	Payload'	_____ _____

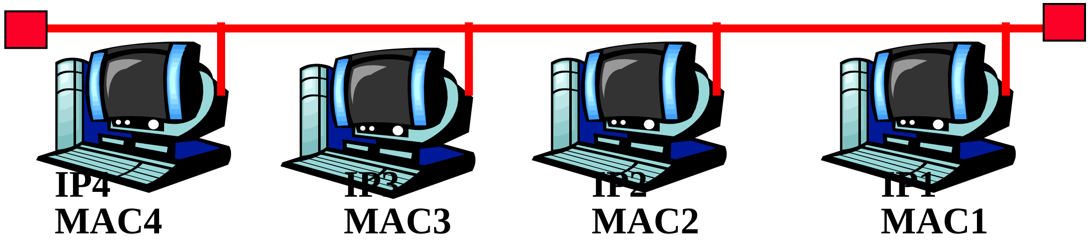


Question: are Payload and Payload' identical?

Network Communication (Cont.)

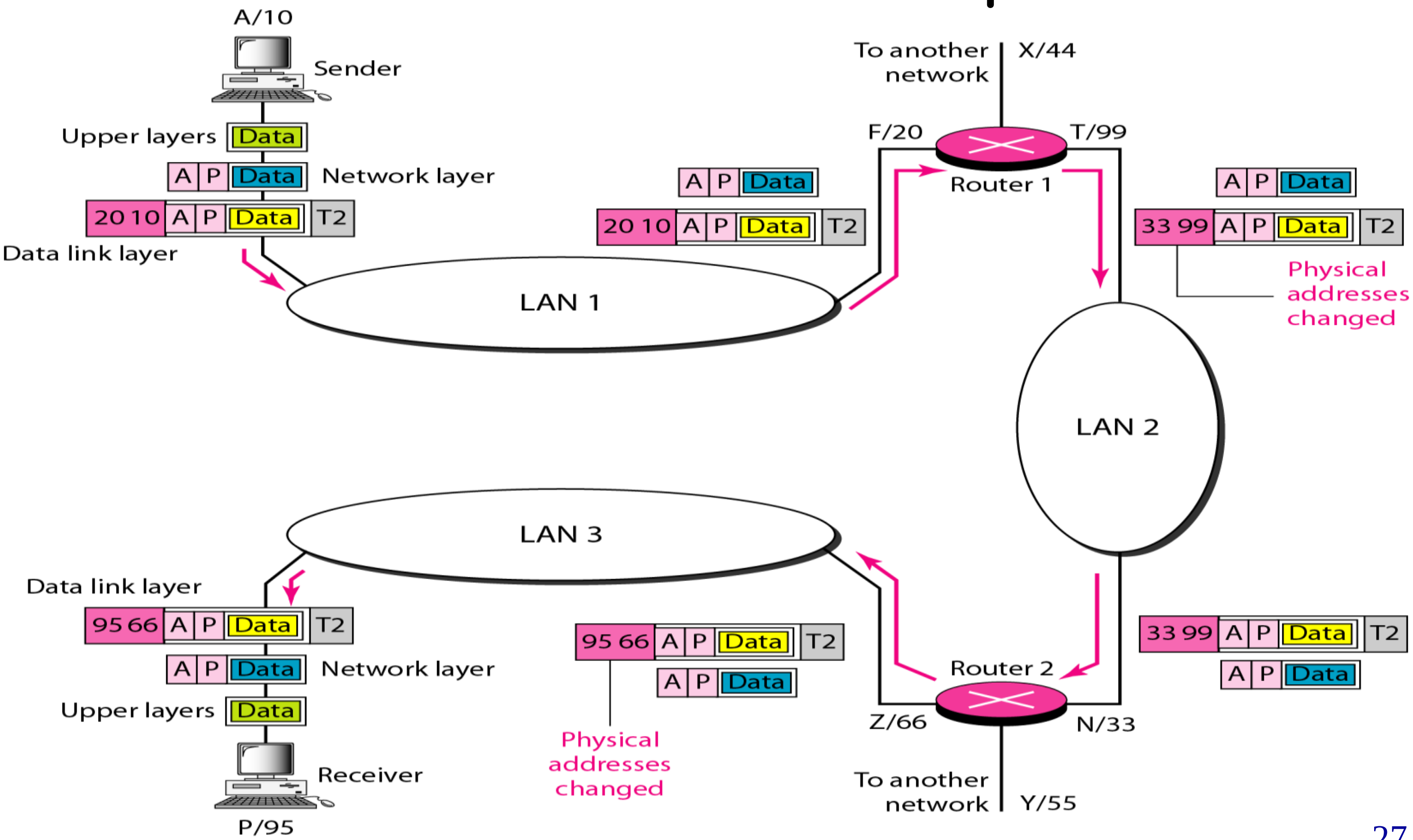
- The role of IP address in the delivery of the frame
- IP is only included in the packet so that the destination knows who sent the message and uses it to reply if needed. It is the MAC address that plays the role in the delivery process

Frame (message)	dt	Payload	MAC1 MAC3
Frame (reply)	dt	Payload’	MAC3 MAC1



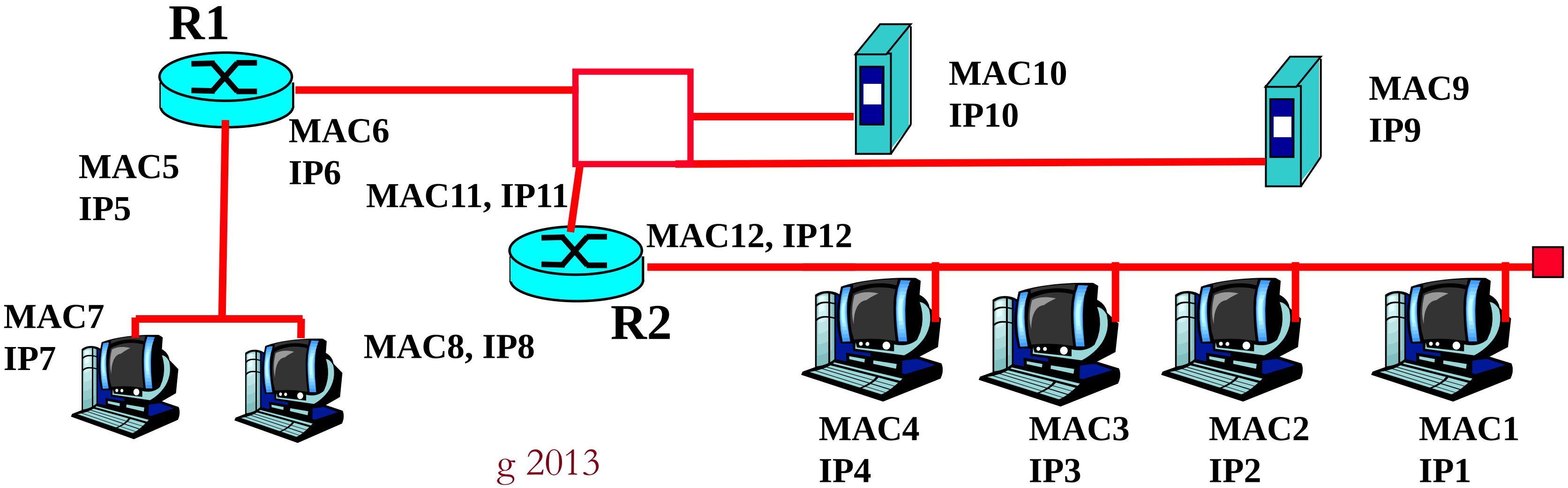
Internetwork Communication – Example

- A part of an internet with two routers connecting 3 LANs. Each device (computer or router) should have a pair of addresses (logical and physical) for each connection. Each computer is connected to only one link, and so has one address pair, whereas each router is connected to 3 connections each of which has a separate address pair



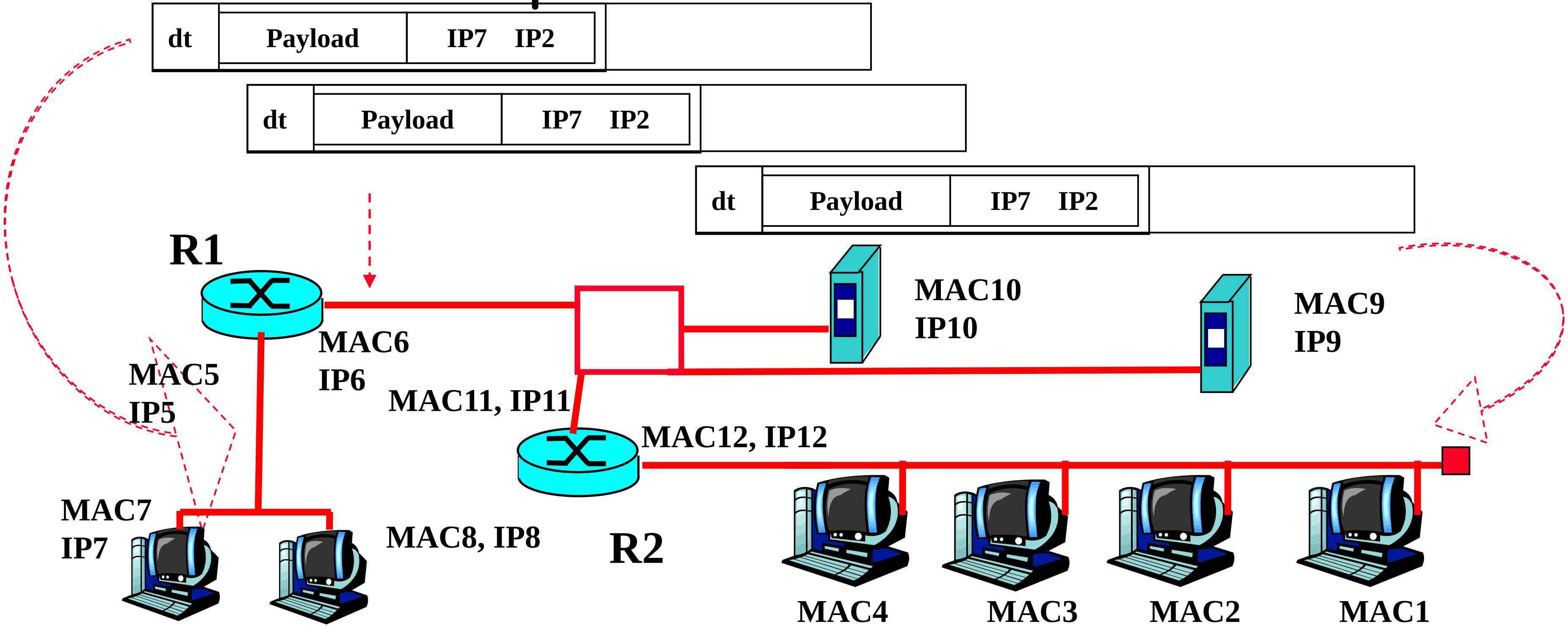
Internetwork Communication

- There are 3 networks in this diagram. Remember that routers have not only MAC addresses but also IP addresses because they have 3 layers [Question: How about layer 2 switches? Or hubs?]
- Consider a router here. For each network (link) it is connected to, it needs to have a different address pair, because the NIC card is different (and network address part is different)
- Suppose the end system, E7 with addresses MAC7 and IP7 wants to send a message to the end system, E2 with MAC2 and IP2



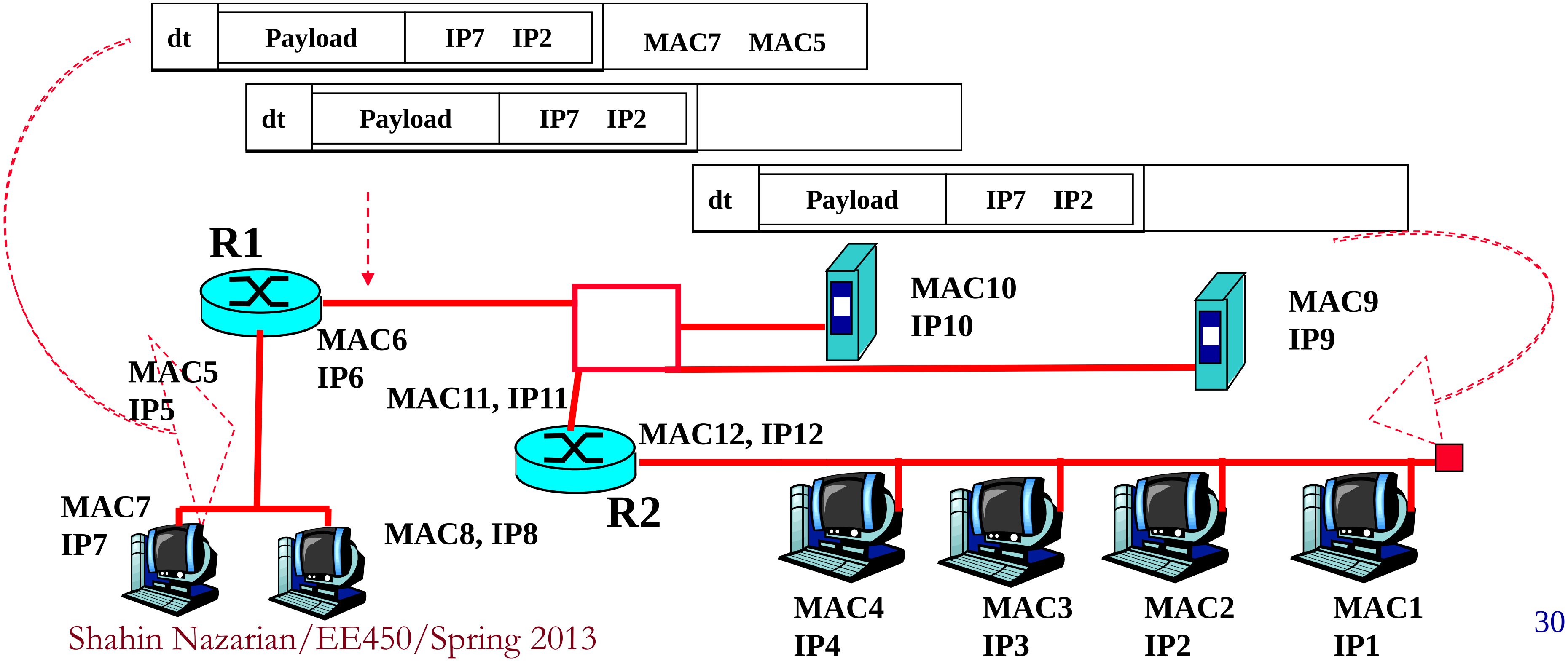
Internetwork Communication (Cont.)

- E7 needs E2's IP address (IP2) to determine whether E2 is located on the same network. It will then compare the network address part of its IP address with that of E2. Here E7 finds out networks are different, so E7 needs the help of a router
- E7 then creates a packet with its IP address and E2's



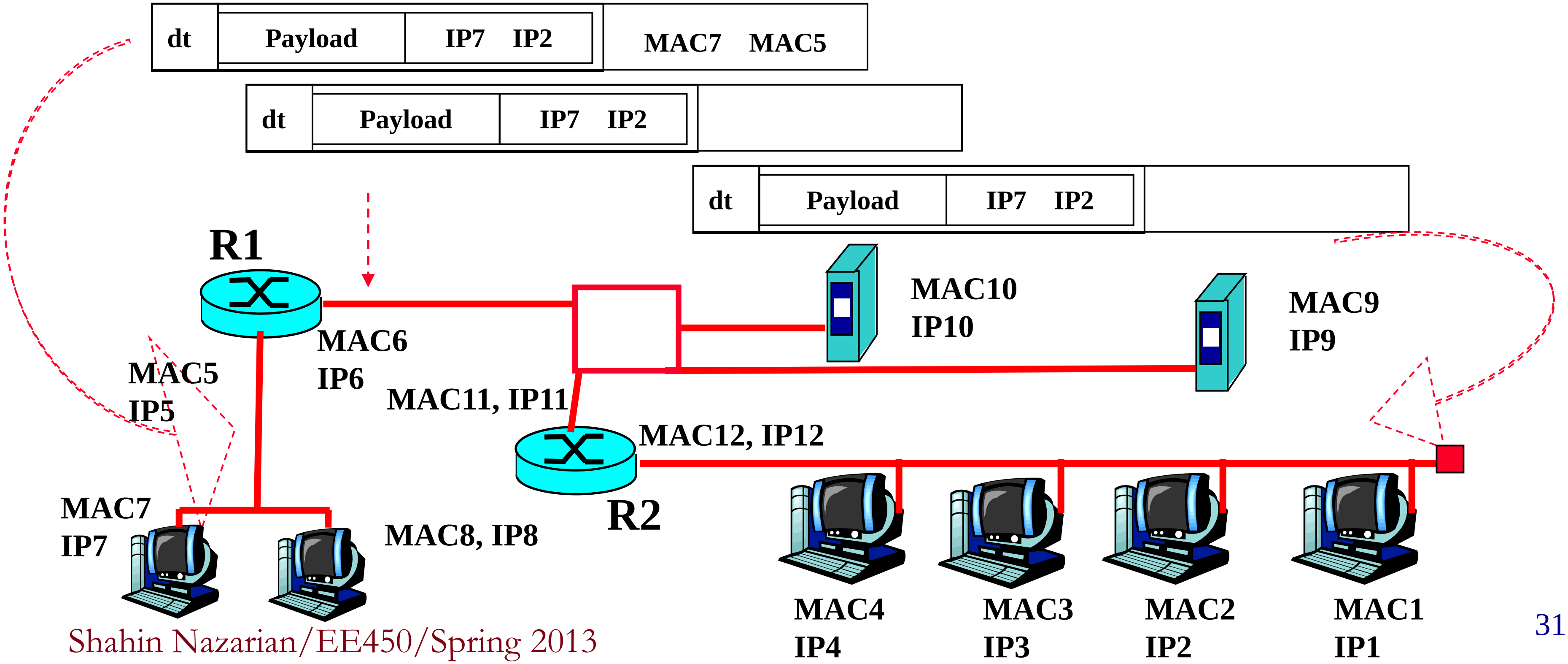
Internetwork Communication (Cont.)

- E7 needs the MAC address of the router not E2. In general if the destination cannot be reached directly, source does not need destination's MAC address. The router has two MAC addresses, MAC5 and MAC6, it needs MAC5. So the frame has the address to the next node, whereas the packet has the address to the final destination. In a shared network (containing both source and destination) the next node is the final destination



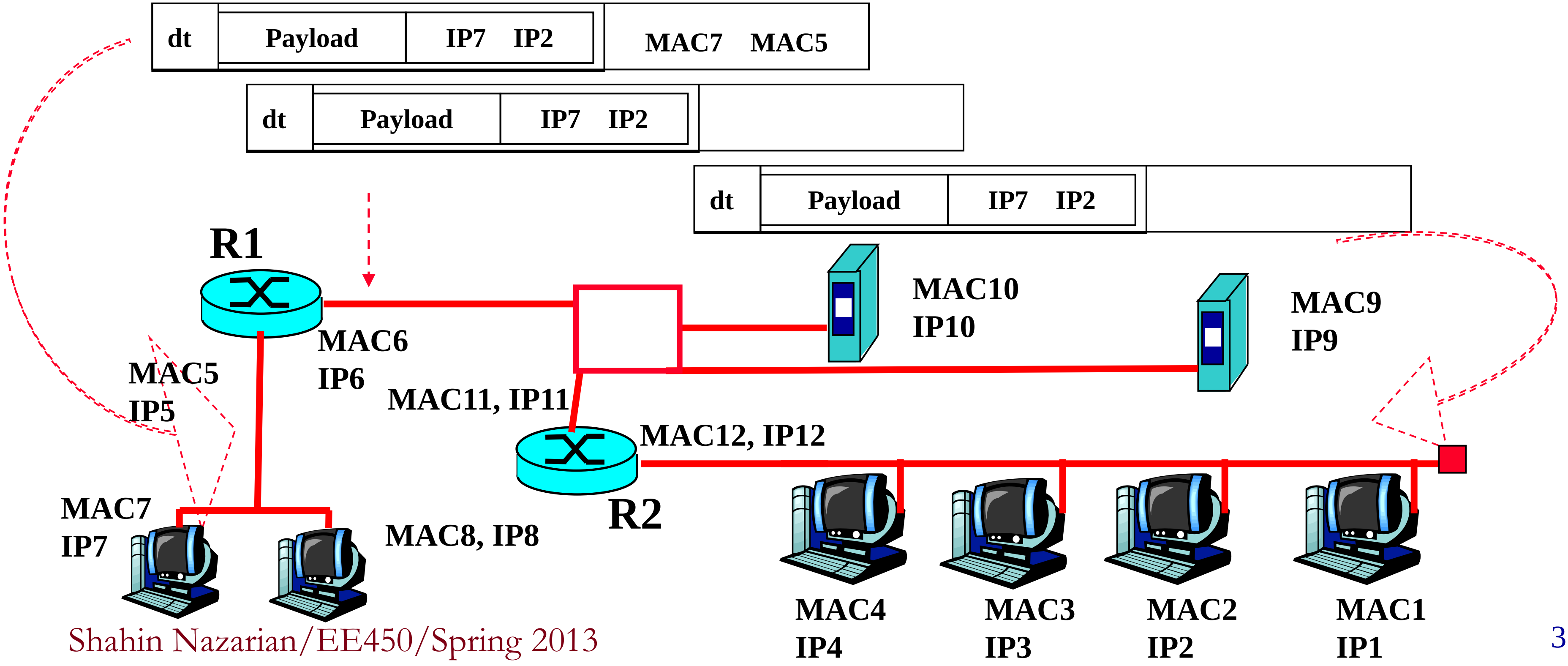
Internetwork Communication (Cont.)

- E7 needs router's MAC address. It gets it through ARP service, it needs the IP address of the router, but router's IP address cannot be received through DNS because DNS is only for the hosts not for the routers
- Question: How does E7 know the router's IP address?



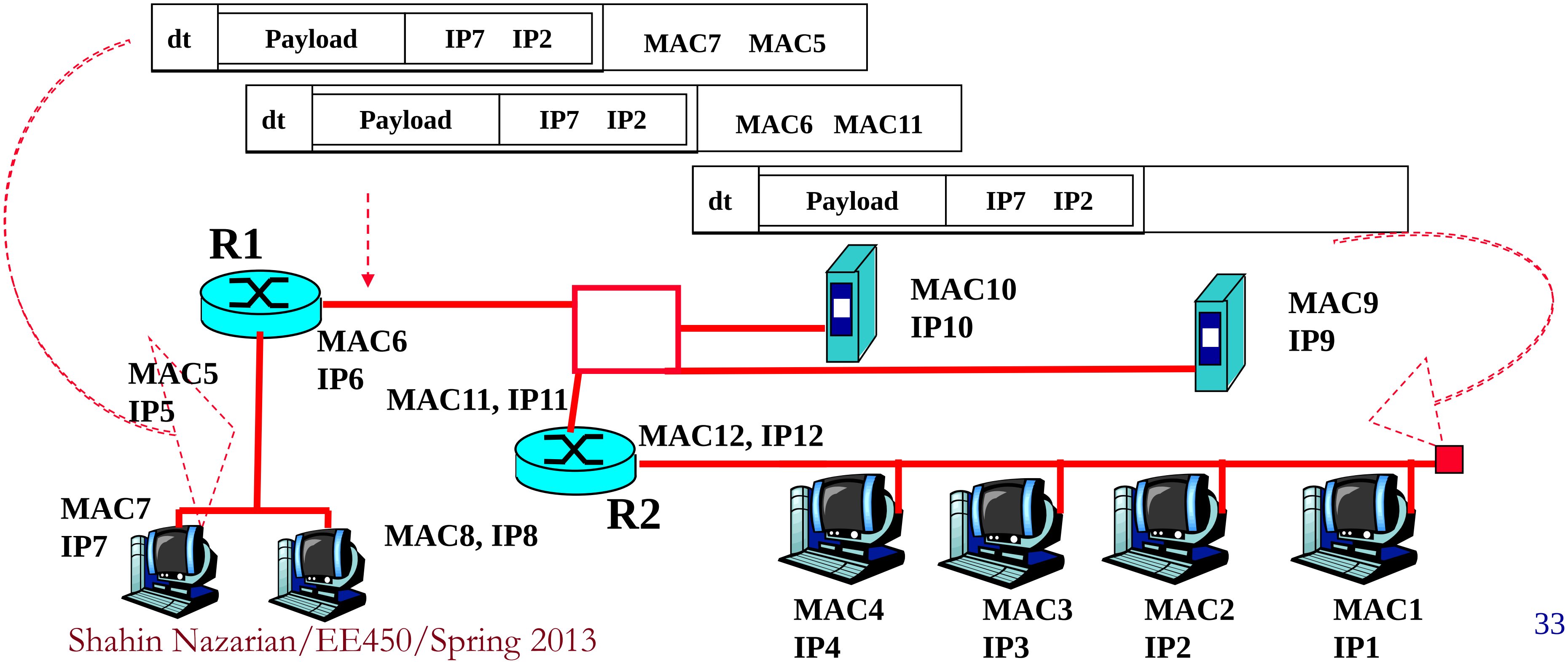
Internetwork Communication (Cont.)

- So E7 is not only configured [either manually by the network admin. or dynamically through the service of Dynamic Host Configuration Protocol with IP7, but also with the IP address of the default router (in this case, R1's, i.e., IP5)
- Note that E8's NIC will disregard the frame by comparing its MAC address, MAC8 with destination's MAC address, MAC5



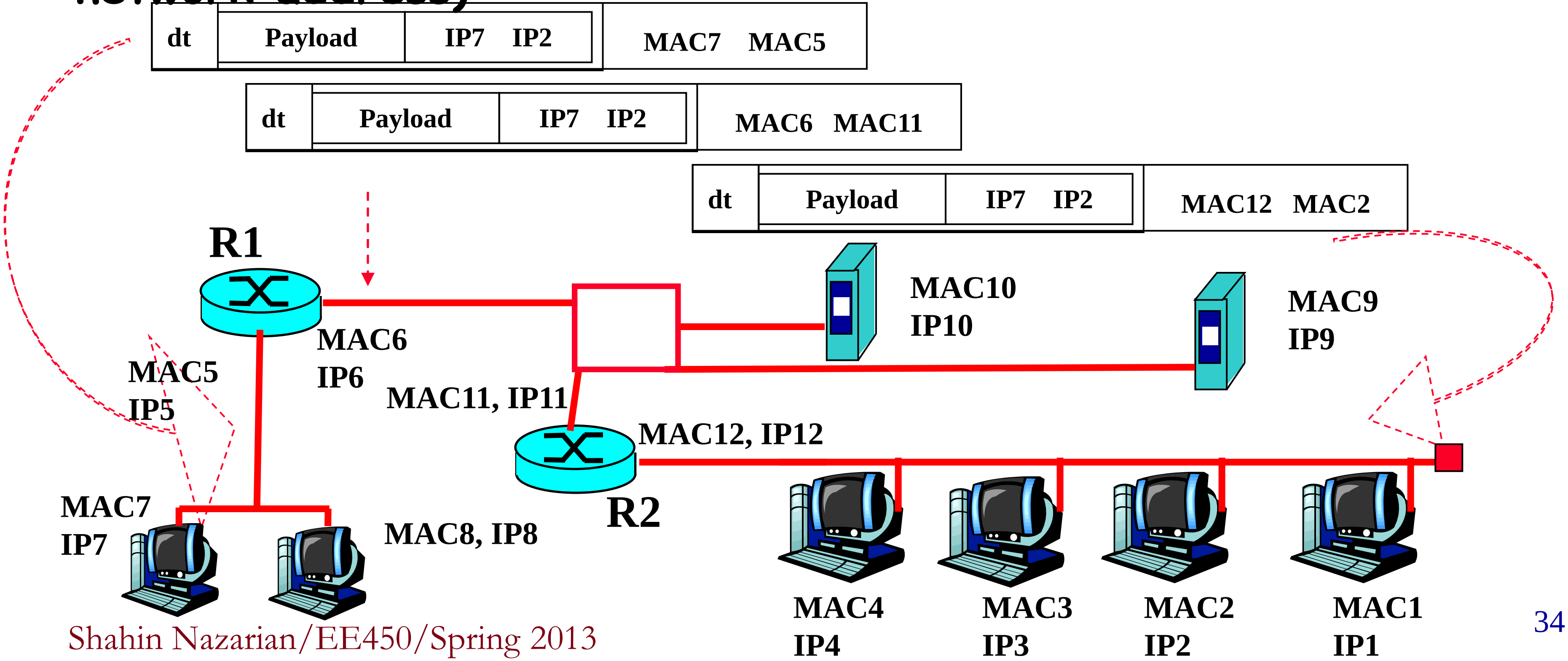
Internetwork Communication (Cont.)

- The router, R1, gives the packet to its layer 3. Layer 3 looks at the packet header and checks the routing table and finds out that it cannot reach IP2 directly by itself, so it needs to get help of another router, R2. E7 does not know R1 cannot deliver the frame directly to E2, but it does not need to know
- Note that the two routers, R1 and R2 are on the same network



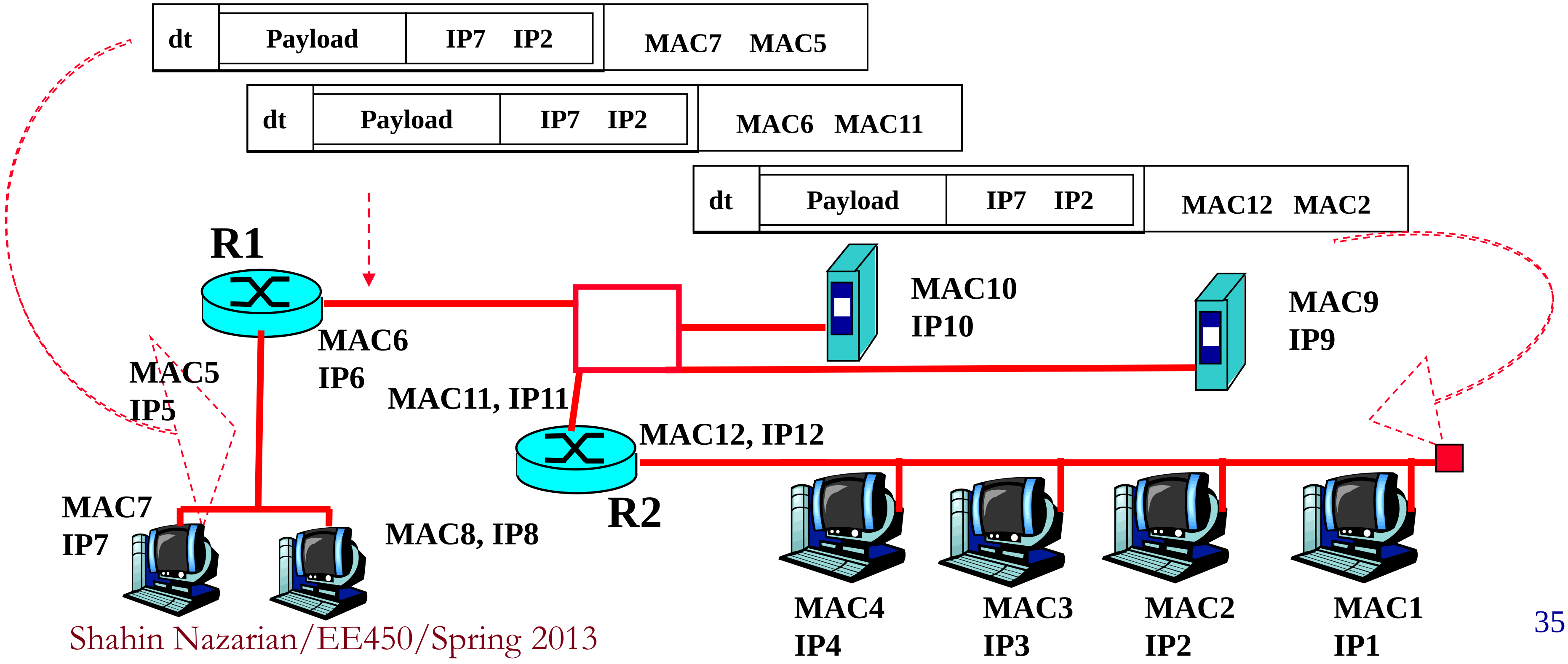
Internetwork Communication (Cont.)

- E10 and E9 would receive the frame created by the router (R1), but their 2nd layer will disregard the frame by comparing MAC11 and their MAC address. But R2 will process it further to 3rd layer. R2 then checks the routing table and finds out that it can reach E2 directly without help of any router (by comparing the network address of the packet with its own network address)



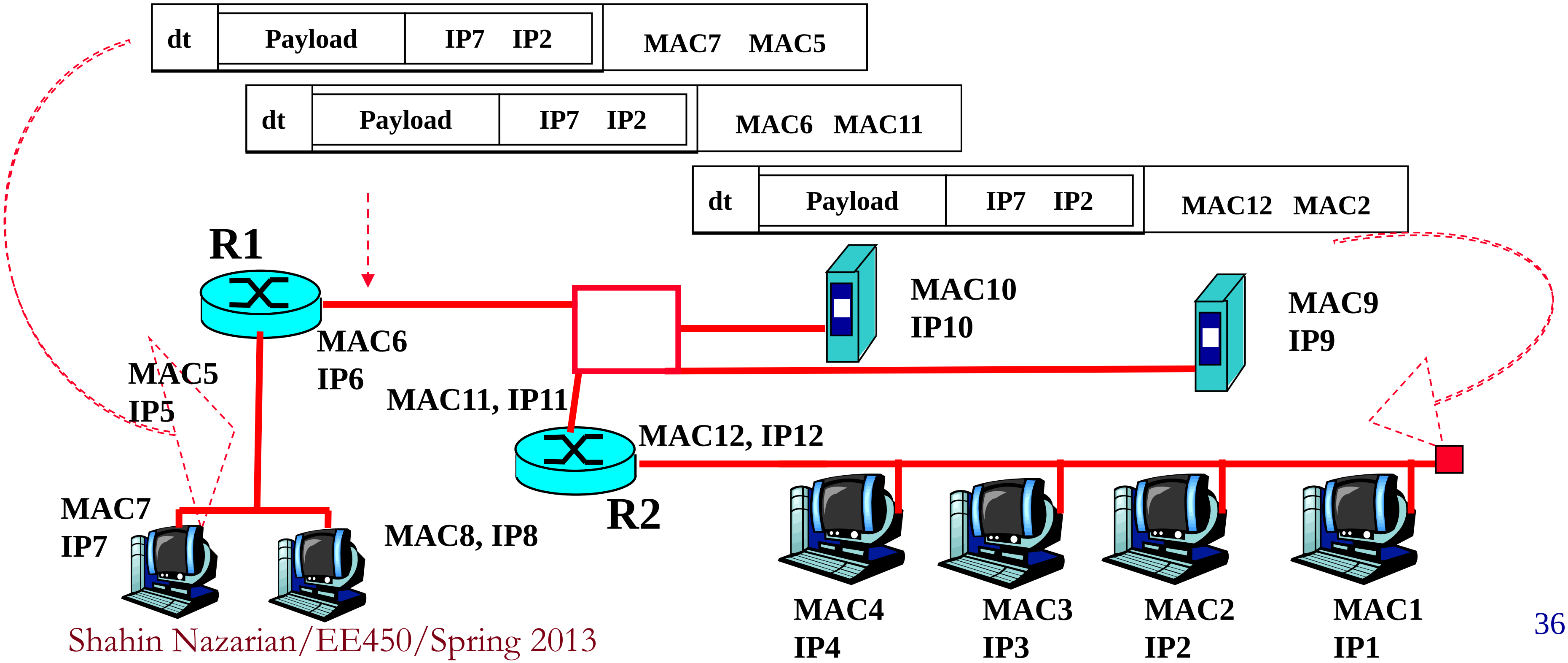
Internetwork Communication (Cont.)

- R2 does not change the packet addresses, but changes the frame by putting MAC2 as the destination (and MAC12 as the source.) It knows MAC2 through ARP
- E2 receives the frame in its 2nd layer, but does not know yet who sent the packet, however it knows R2 sent the frame
- The role of IP address in delivery



Internetwork Communication (Cont.)

- E2's layer2 does the required layer2 functions, then removes the header and trailer and gives the payload (to be the packet) to the 3rd layer. In 3rd layer it realizes that E7 sent the packet
- Note that the delivery is still incomplete, but the packet data has the port number which eventually makes the delivery complete in upper layers



Internetwork Communication (Cont.)

- IP address plays a major role in delivery in this case, but MAC address does not. Compare this with our first example that MAC played the role in delivery, but IP address did not
- Same concept explained here for internetworking can be extended to the whole Internet
- In Internet communication, IP addresses do not change while the frames (and packets) are traversing the networks, but the MAC addresses do change
- This concept is a bit different from the postage system, as here seems like there are two envelops and the top layer envelop needs to change from hop to hop!

Internetwork Communication (Cont.)

- Question: R1 realizes the MAC of R2, through ARP, but it needs its IP first to do ARP
- These two routers communicate between themselves this information, meaning that R2 informs R1 that it can reach certain networks and R1 will keep that information in mind in its routing table and will know which router would be useful for which network (discussed later as part of routing protocols)

ARP

- A router and an end system, both use ARP (Address Resolution Protocol) to get the MAC address of another end system or router on the same Local Area Network (LAN)
  - Therefore ARP is a local protocol and it is domain or effectiveness is within the network

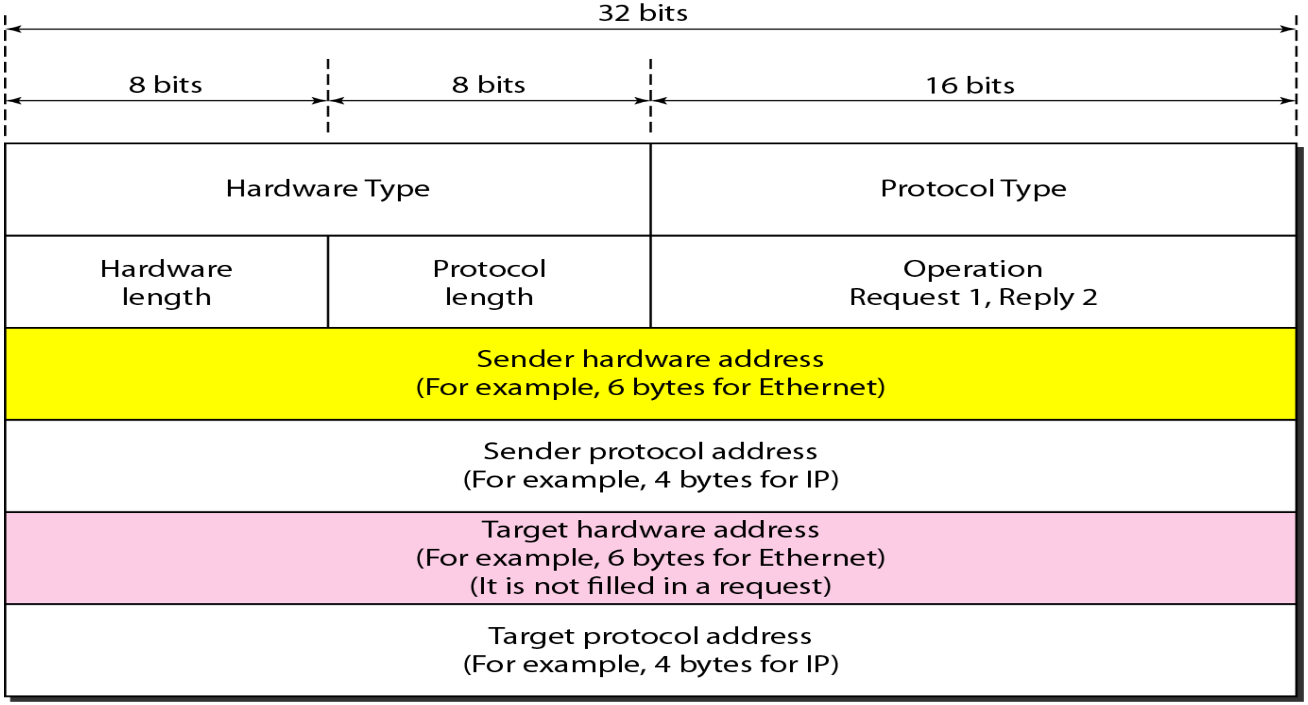
- Be reminded that if the destination node is not on the same LAN, the source node does not need to know the destination node's MAC address; instead it needs the MAC address of its default router

ARP (Cont.)

- ARP is a simple, layer 3 protocol and provides IP with its service. ARP resides underneath the IP (both on the Network layer)
- Note that each node, host or router in the network has the ARP module
- The IP protocol wants to create a packet. IP gives the IP address of a host or router to the ARP protocol and asks for the MAC address of that host or router (that host or router resides in the same network)
- ARP will reply to a service request by an ARP packet, but that packet is not the IP packet. The actual IP packets is buffered and is waiting for the response from the ARP

ARP Packet

- The packet components below are following each other serially
- Hardware type is a code used to show the technology type, e.g., Ethernet. Protocol type is IP. Hardware length is the length of the MAC address and protocol length is the length of the IP address



32 bits

8 bits

8 bits

16 bits

Hardware Type

Protocol Type

Hardware length

Protocol length

Operation Request 1, Reply 2

Sender hardware address (For example, 6 bytes for Ethernet)

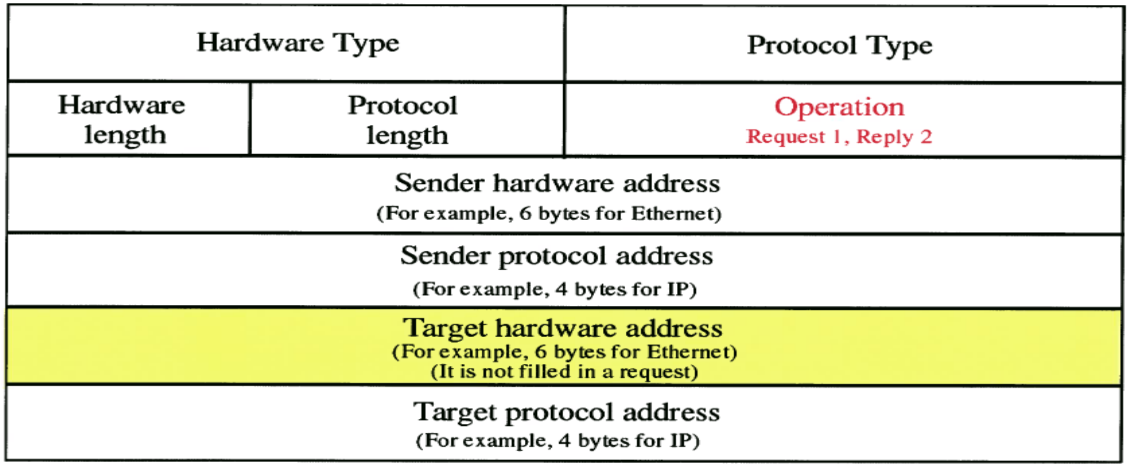
Sender protocol address (For example, 4 bytes for IP)

Target hardware address (For example, 6 bytes for Ethernet) (It is not filled in a request)

Target protocol address (For example, 4 bytes for IP)

ARP Packet (Cont.)

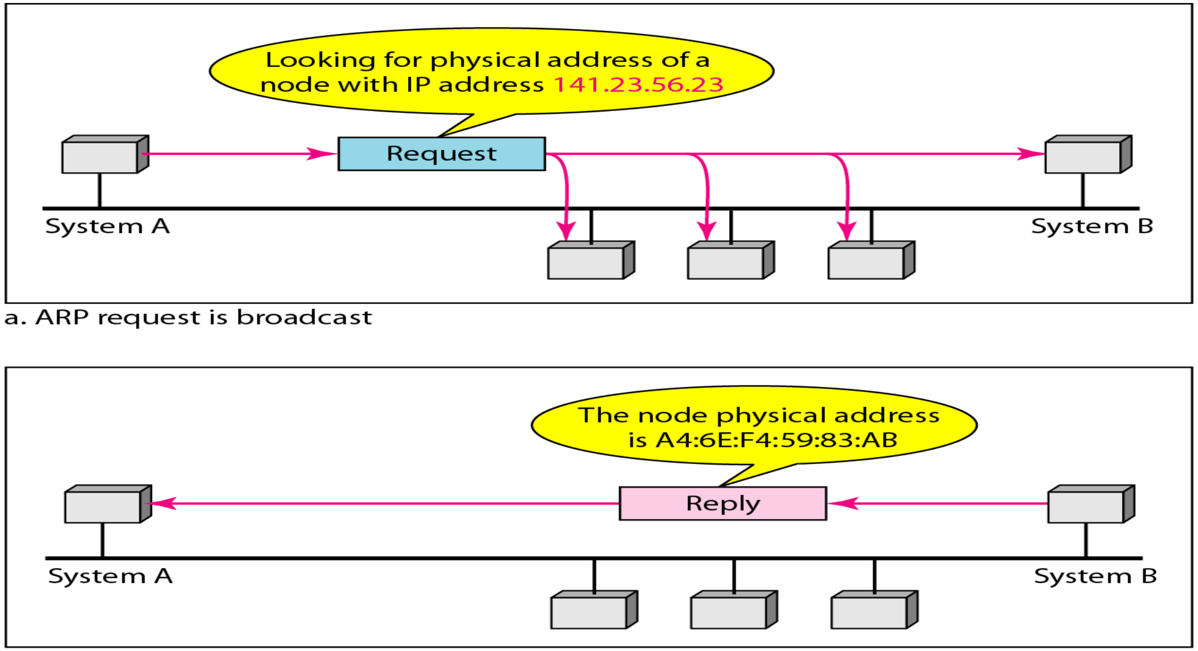
- Sender hardware address and sender protocol address are the MAC and IP addresses of the sender respectively
- The Target hardware address (i.e., the destination MAC address) is what we are trying to get, so it'll be empty (filled with 0s) in the request. Target protocol address is the target IP address
- ARP gives its ARP packet to the 2nd layer
- The frame is going to be broadcasted, so all NICs in the network are required to process this ARP request



Hardware Type		Protocol Type
Hardware length	Protocol length	Operation Request 1, Reply 2
Sender hardware address (For example, 6 bytes for Ethernet)		
Sender protocol address (For example, 4 bytes for IP)		
Target hardware address (For example, 6 bytes for Ethernet) (It is not filled in a request)		
Target protocol address (For example, 4 bytes for IP)		

ARP Operation

- For all NICs, the 2nd layer, processes the frame by removing the header and trailer and passes the rest (i.e., the ARP packet) to the NIC's 3rd layer. If the target IP address is different from its own IP, the NIC ignores the packet and does not process it any further
- Note that when a node receives the frame, it does not know yet whether this is an ARP request. The node will know about it after processing the ARP request

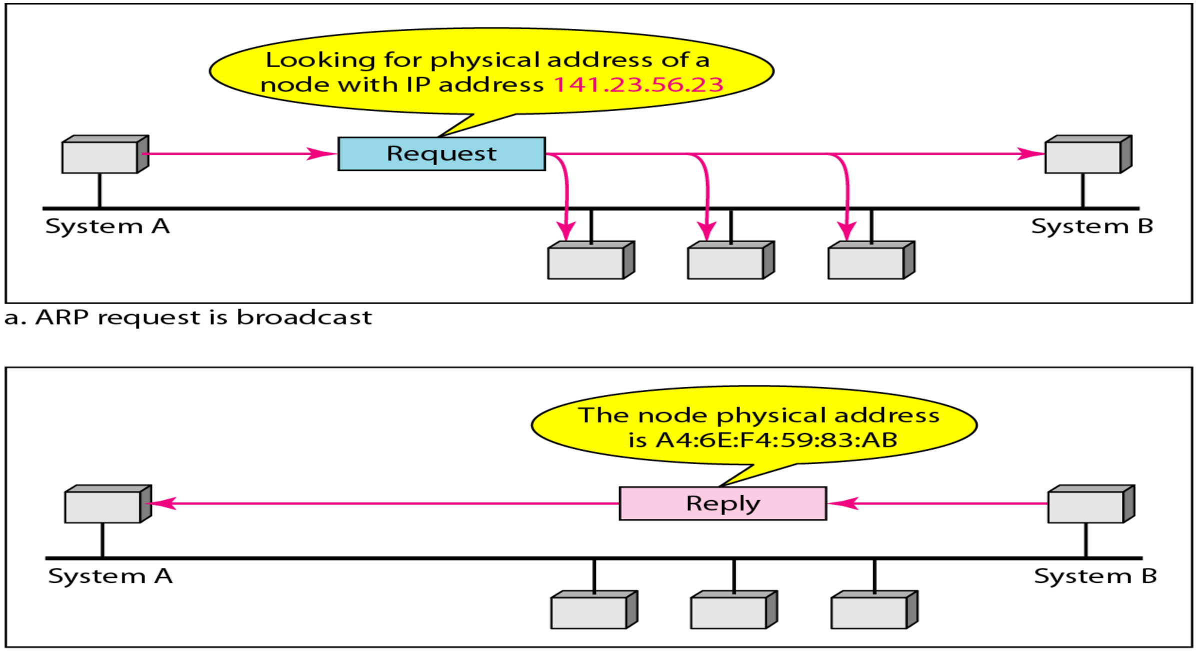

Looking for physical address of a node with IP address 141.23.56.23
Request
System A
System B
The node physical address is A4:6E:F4:59:83:AB
Reply
System A
System B

a. ARP request is broadcast

b. ARP reply is unicast

ARP Operation (Cont.)

- Among all the NICs, the target NIC (B's NIC) recognizes that the target IP address is its IP address; B's ARP module then creates an ARP Reply packet and gives it to its 2nd layer and finally unicasts it to A, the sender of the ARP broadcast
- When A receives the message, it does not know yet that it is an ARP reply. It knows it after processing the ARP packet

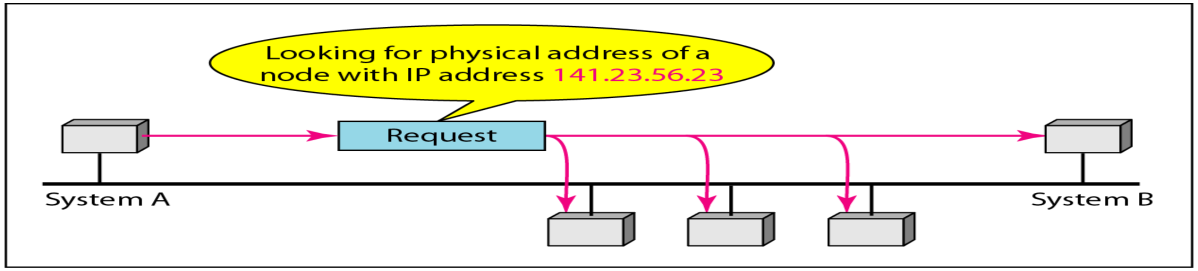

Looking for physical address of a node with IP address 141.23.56.23
Request
System A
System B
The node physical address is A4:6E:F4:59:83:AB
Reply
System A
System B

a. ARP request is broadcast

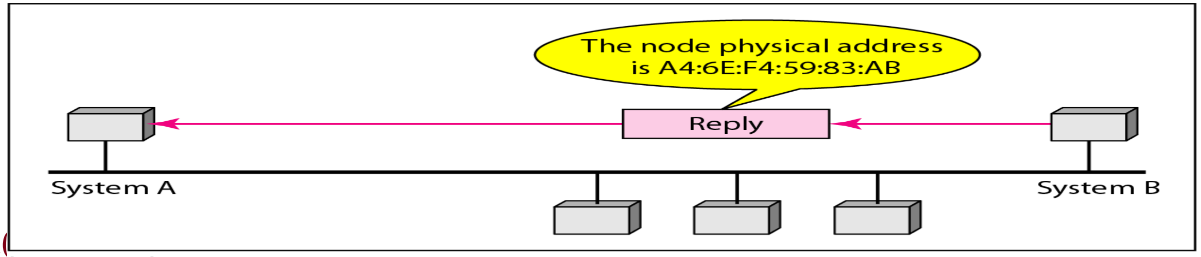
b. ARP reply is unicast

ARP Operation (Cont.)

- ARP has a cache table that stores the IP to MAC address mappings for some of the nodes in the network
- Nodes cache the MAC address and IP mappings (e.g. in A it caches the IP and MAC addresses of B) to avoid repeating ARP next time. Remember that ARP involves a broadcast ARP request which is not good
- Cache life time is typically less than 30 or 20 minutes, suppose B's IP and MAC get erased from A's cache, then later on if A again needs B's MAC address it needs to repeat the ARP request


Looking for physical address of a node with IP address 141.23.56.23
Request
System A
System B

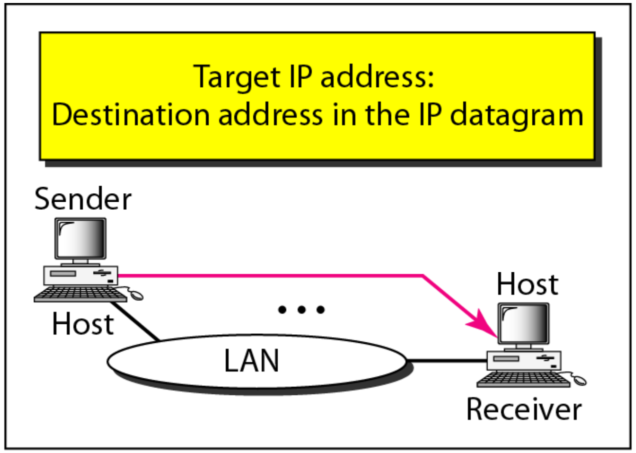
a. ARP request is broadcast



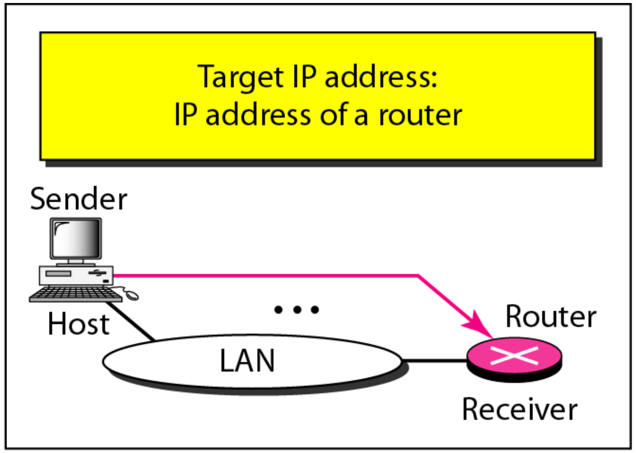
b. ARP reply is unicast

Four Cases Using ARP

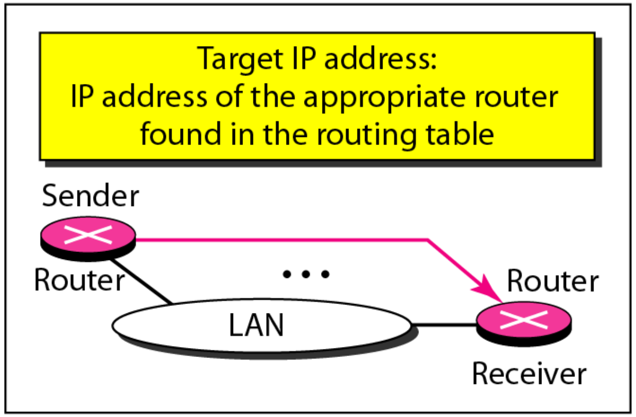
- Although simple, ARP protocol is tiresome, because it involves broadcast messages


Target IP address:
Destination address in the IP datagram

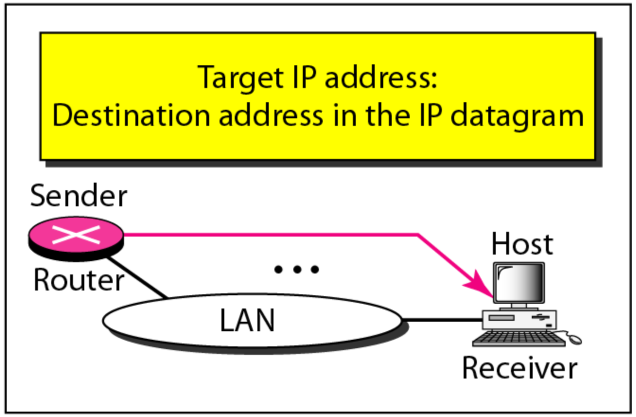
Case 1. A host has a packet to send to another host on the same network.


Target IP address:
IP address of a router

Case 2. A host wants to send a packet to another host on another network.
It must first be delivered to a router.


Target IP address:
IP address of the appropriate router
found in the routing table

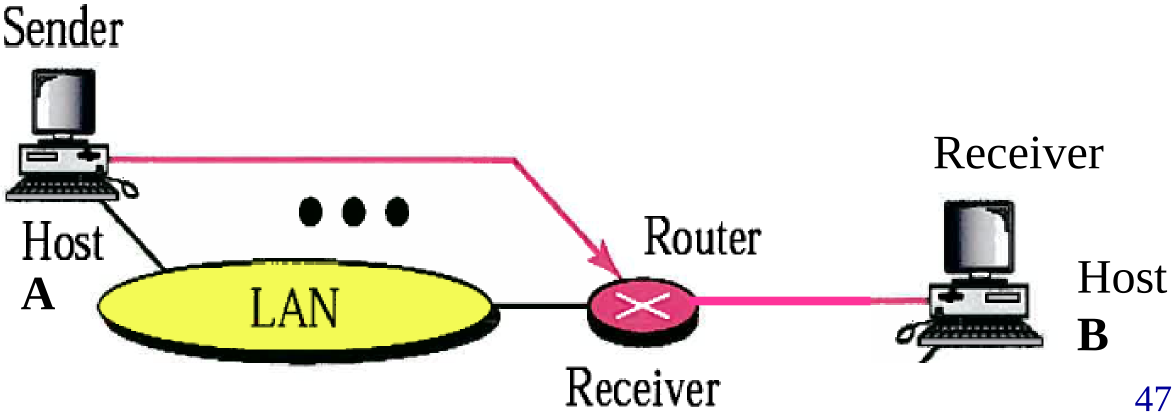
Case 3. A router receives a packet to be sent to a host on another network. It must first be delivered to the appropriate router.


Target IP address:
Destination address in the IP datagram

Case 4. A router receives a packet to be sent to a host on the same network.

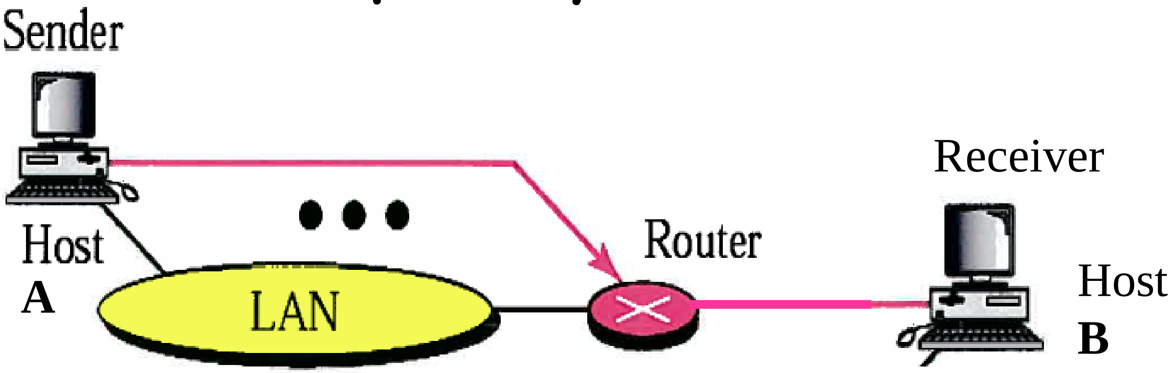
ARP Operation – Internetwork

- A needs IP address of B and gets it through DNS, then needs to find out whether or not B is located on the same network. In this case, it's not. A then needs the MAC address of the router. It gets it through ARP
- A needs to execute ARP by sending a broadcast ARP request (in ARP packet, the Sender MAC and IP addresses are A's MAC and IP addresses, Target MAC address is empty (filled with 0s) and Target IP address is router's IP which is configured statically or dynamically in A) Note that the ARP frame is broadcasted
- A caches the MAC address of Router
- ARP response (which in this case is created by R) is unicast



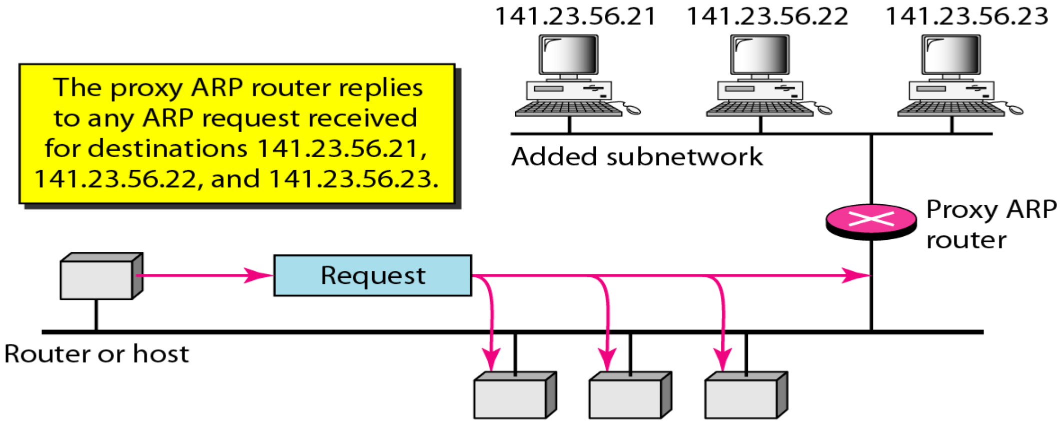
ARP Operation – Internetwork (Cont.)

- After getting the MAC address of Router, A prepares the IP packet in layer 3 and then frame in layer 2 and sends the frame to the router through its physical layer
- Router then sends the frame to B: It first checks its cache to see if it can find the MAC address of B in there and if so, it would not need to send an ARP request, but if not, then the ARP packet has the sender MAC and IP addresses as Router's; in ARP packet, the Target physical address is empty, the Target IP address is B's IP address. In ARP frame, MAC address is a broadcast one. B then responds to R's broadcast ARP request by a unicast ARP reply



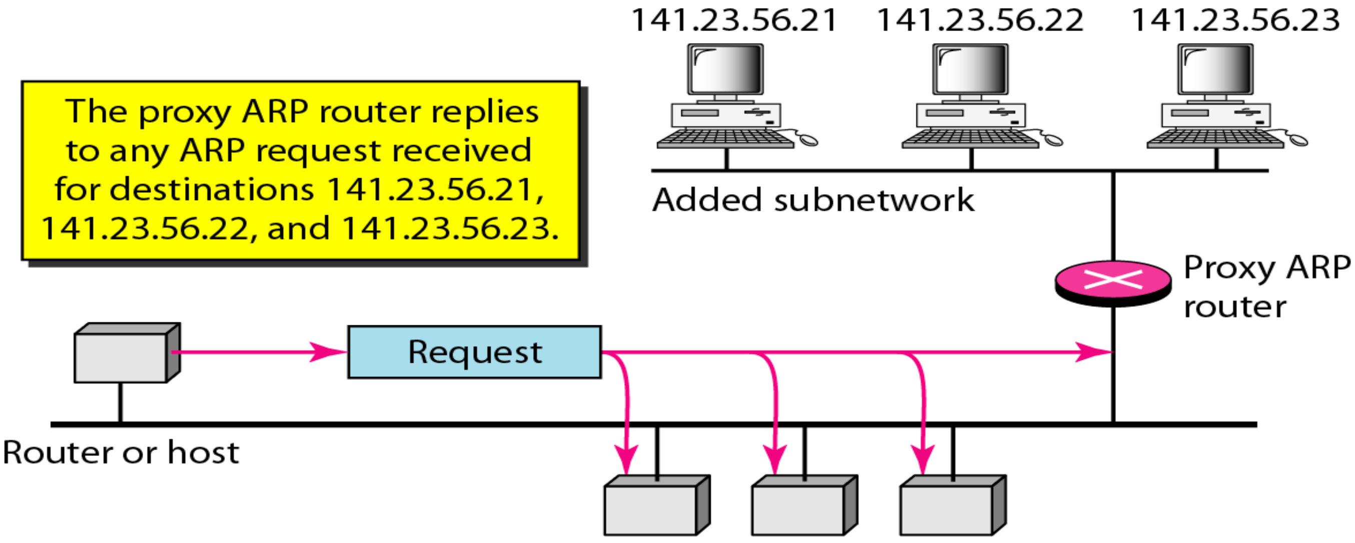
Proxy ARP

- Proxy implies somebody or something that would substitute the other(s)
- In case of proxy ARP, the end systems are all logically on the same network, but some (the ones in the added subnetwork) are physically on different cables; the end systems' network address is the same, however the network administrator puts some end systems in the back, so that nobody can reach them from outside the LAN due to security reasons


141.23.56.21
141.23.56.22
141.23.56.23
The proxy ARP router replies to any ARP request received for destinations 141.23.56.21, 141.23.56.22, and 141.23.56.23.
Added subnetwork
Proxy ARP router
Request
Router or host

Proxy ARP (Cont.)

- The admin uses a router called the Proxy ARP router
- Example: if the Target IP address is 141.23.56.22 (and its MAC address is denoted by MAC_d,) the proxy router responds by giving its own MAC address instead of MAC_d

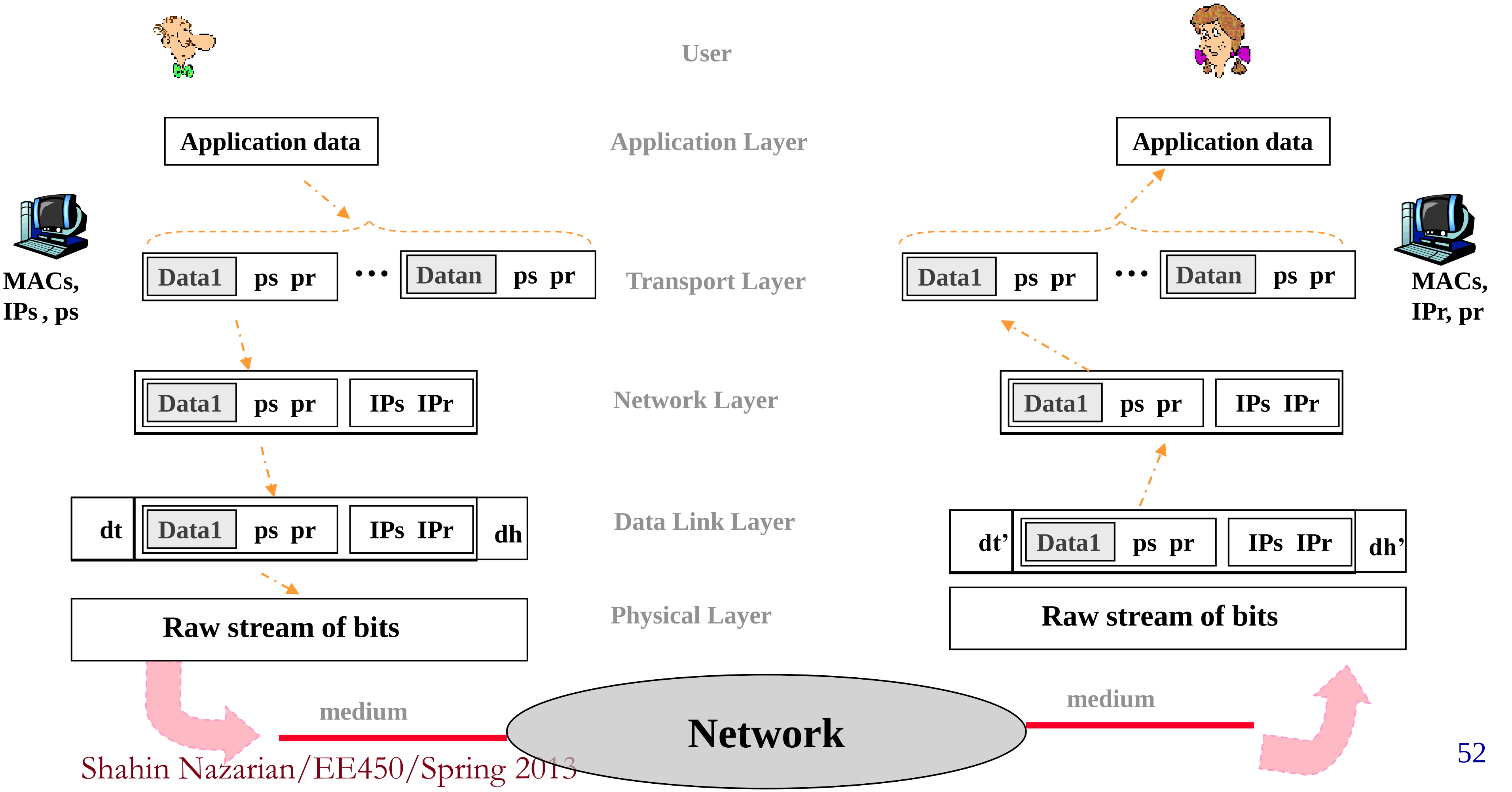

141.23.56.21
141.23.56.22
141.23.56.23
The proxy ARP router replies to any ARP request received for destinations 141.23.56.21, 141.23.56.22, and 141.23.56.23.
Added subnetwork
Proxy ARP router
Request
Router or host

Host Configuration

- Reminder: a network administrator assigns a host an IP address (and lets the host know.) The IP address is either static IP address, or dynamic IP which is on demand basis (i.e., the idea of assigning it to you when you need it)
- A static IP needs to be globally unique, so the network administrator removes the IP address from the list of available IPs when it is statically assigned
- A network administrator provides the host also with the subnet mask, local DNS server and default router

Port Address

- A port address has 16 bits and is represented by a decimal number, e.g., 993 (http://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers)



Port Address (Cont.)

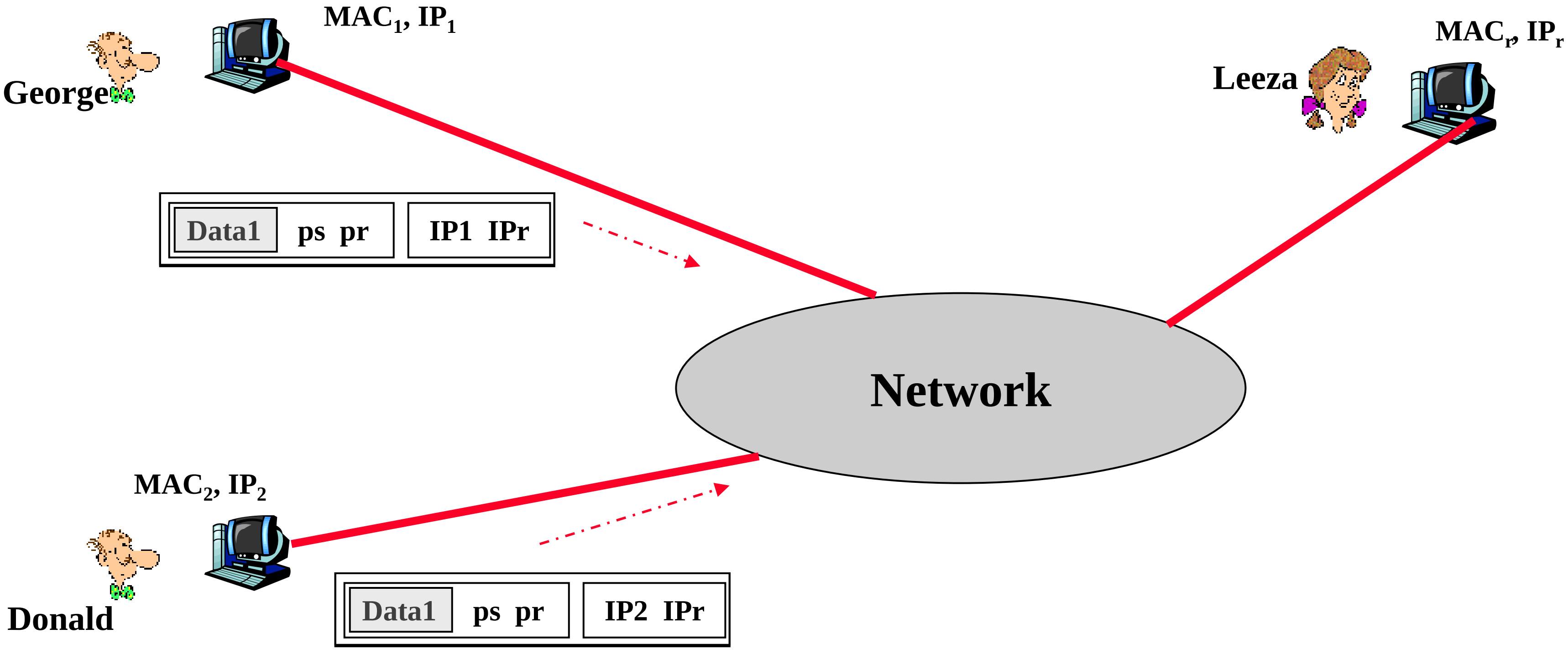
- The server port number is called the well-known port number
- If the server is the sender, the source or sender port number, ps, is the well-known port number, but if the client is the sender, the destination or receiver port number, pr, is the well-known one
- Data is split into n parts each will have both ps and pr. Each part will also be converted into a packet and then a frame

Questions:

- What MAC addresses go into dh? Answer the same question for the case of dh'
- Can dh and dh' be the same?
- Can the host port number be the well-known port number?

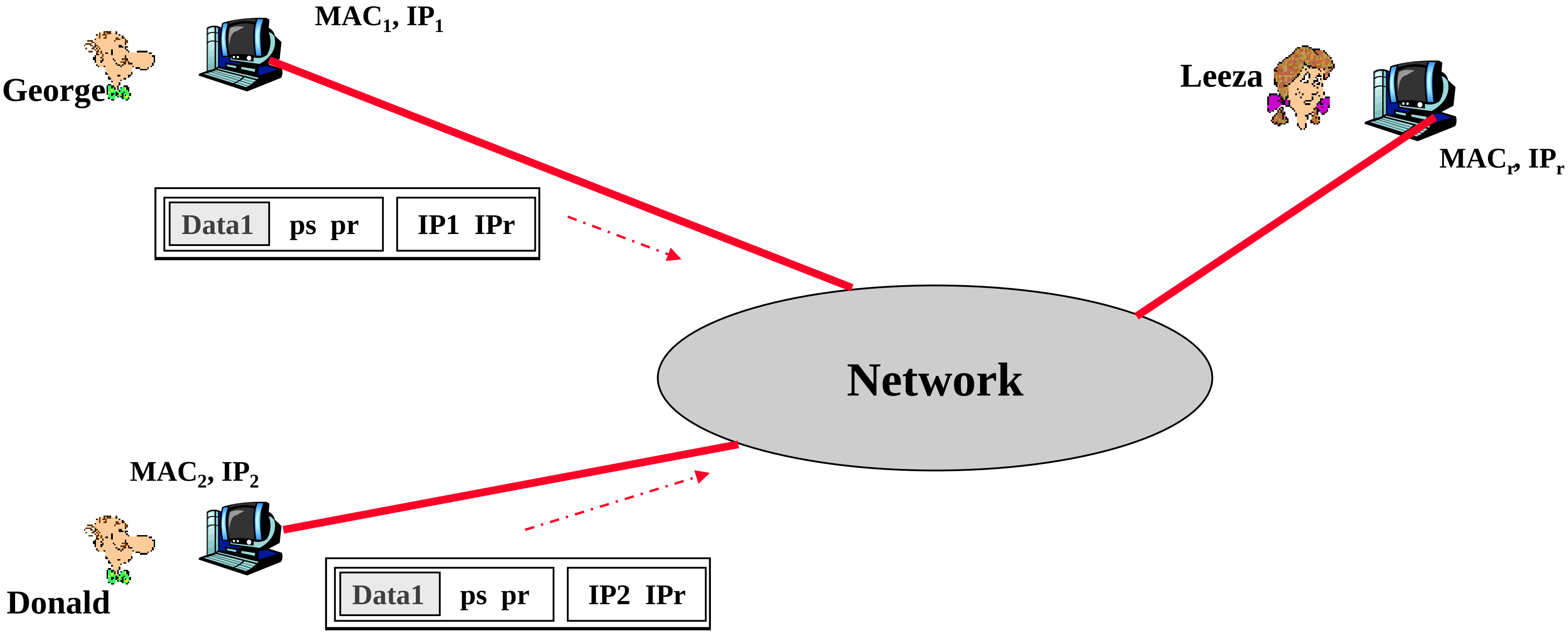
Port Address (Cont.)

- Reminder: the Network can be as big as the Internet
- Question: Can the port numbers be identical for the machines George and Donald are using, while they are using the same application, trying to communicate with an application in the machine Leeza is using?



Port Address (Cont.)

- Yes! Destination port number, pr is identical for both. If Leeza's machine is the server, port numbers for the senders are chosen randomly by their NOS. Therefore it may be the case that ps numbers that are randomly chosen turn out to be the same
- Question: How would the receiver machine (Leeza's) distinguish between them? By their IP addresses



DHCP

- A host gets it's own dynamic IP address through DHCP (Dynamic Host Configuration Protocol.) Note that in dynamic IP assignment, the idea is that the IP is assigned when it is needed
- Once an organization has obtained a block of addresses, it can assign individual IP addresses to the host and router interfaces in its organization. A system admin. will typically manually configure the IP addresses into the router with a network management tool. Host addresses can also be configured manually but more often this task is done using the DHCP. DHCP allows a host to obtain (be allocated) an IP address automatically

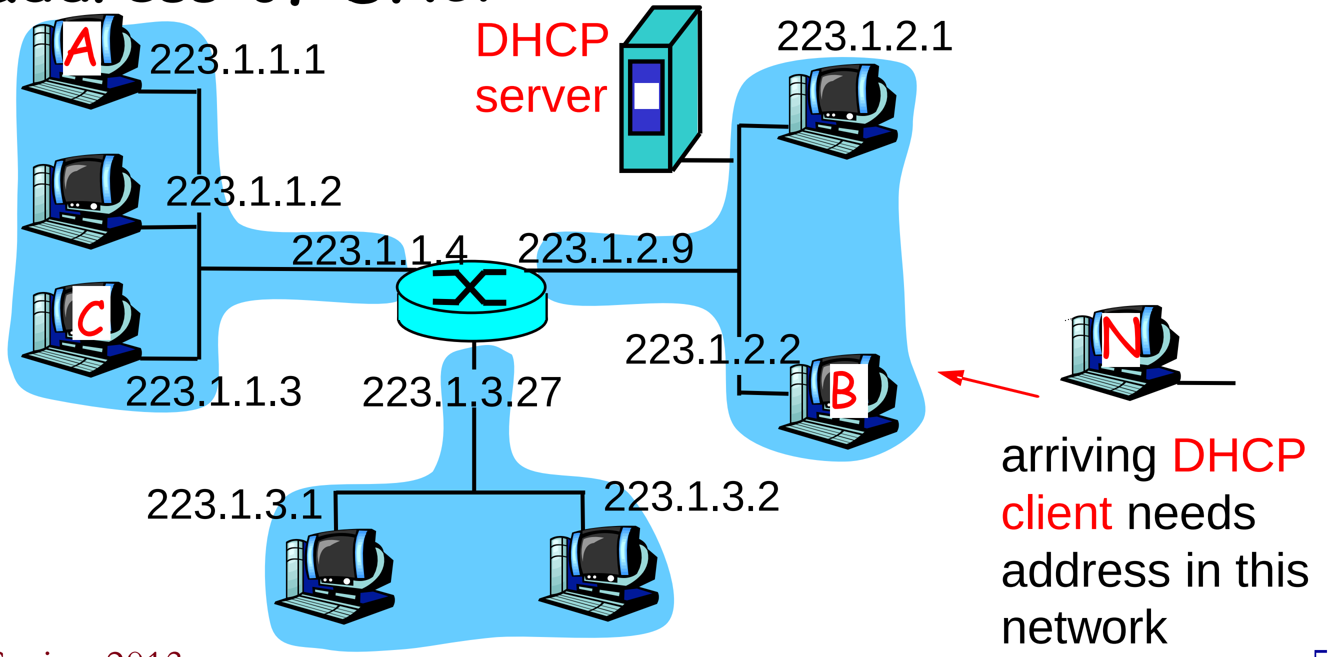
DHCP (Cont.)

- Network admin. can configure DHCP such that a given host receives the same IP address each time it gets connected to the network or it may be assigned a temporary IP address that will be different each time the host connects to the network
- In addition to host IP address management, DHCP also allows a host to learn the subnet mask, the address of its first-hop router (aka the default gateway or router) and the address of its local DNS server
- Because of DHCP's ability to automate the network-related aspects of connecting a host into a network it is often referred to as a plug-and-play protocol
- Optional: study UPnP (Universal Plug and Play), you may start at:

  http://en.wikipedia.org/wiki/Universal_Plug_and_Play

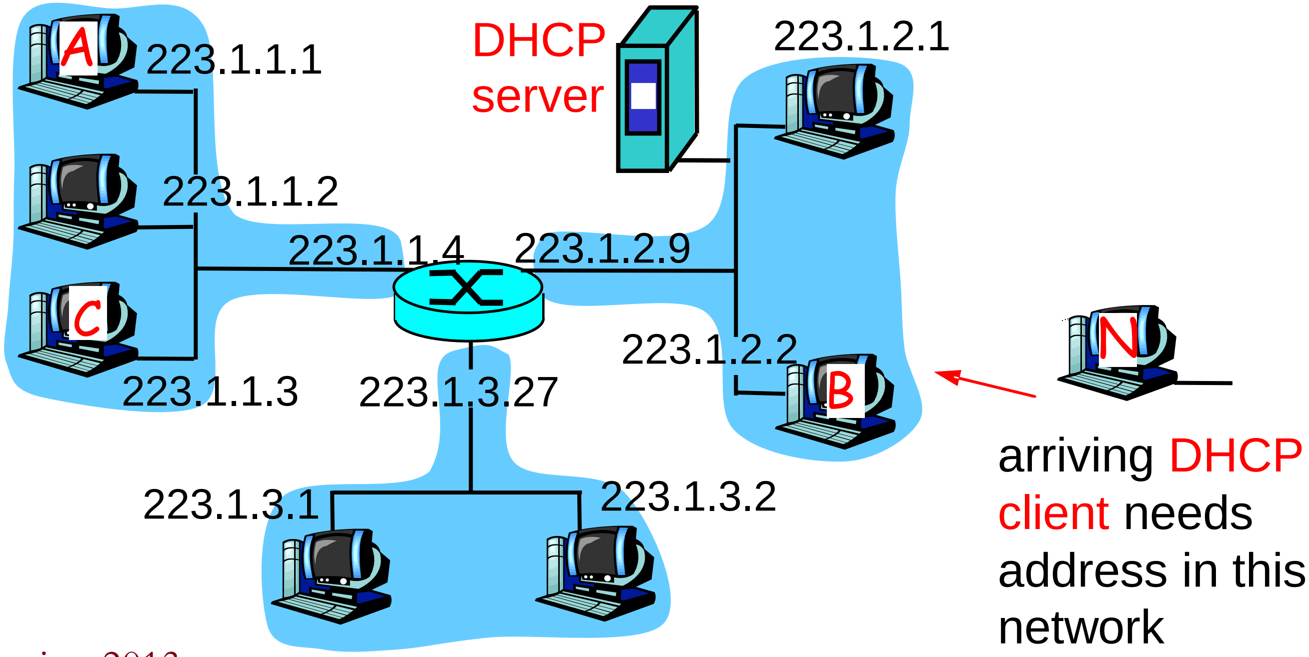
DHCP (Cont.)

- DHCP is a client-server protocol, i.e., one runs on client machine and another on server (with 67 as the well-known port)
- N, the new client to the network, does not have an IP address and knows nothing about the network. It is going to get it from the DHCP server, but N does not know the IP address of DHCP



DHCP (Cont.)

- New client, N will go to the discovery phase (initiated by N) to discover the DHCP server's IP address. There can be multiple DHCP servers in the company. There is a pool of IP addresses inside each DHCP server; some IPs are permanent and some temporary
- In the following figure, the client can be connected to the network by cable, DSL, etc. In any case, the DHCP server may be located in the same network, or can be on some other network



DHCP – Discovery Phase

- DHCP uses a client-server architecture and the services of UPD, so no connection setup is required
- In some applications, even client uses a specific port number, e.g., DHCP client always uses 68 & DHCP server always 67 (port numbers are assigned by IANA)
- DHCP packet has to be broadcast. Why?
- How about DHCP frame?
- Compare the DHCP and ARP broadcasts

Optional: IANA

- The IANA (Internet Assigned Numbers Authority) is the entity that oversees global IP address allocation, root zone management for the DNS, media types, and other Internet Protocol related assignments
- IANA used to be administrated by Jon Postel at the ISI (Information Sciences Institute) at USC (University of Southern California,) under a contract USC/ISI had with the United States Department of Defense
- However IANA is currently operated by ICANN (Internet Corporation for Assigned Names and Numbers) which was created under United States Department of Commerce contract to operate IANA

DHCP – Offer Phase

- Question: Will other nodes, e.g. C process the DHCP frame? How about the DHCP packet? C will process the frame and packet both, because they are broadcast ones. C processes the frame and then the packet and finds out the message is "if you are the DHCP server, offer an IP address to me", so C will drop it at its layer 3

Offer Phase:

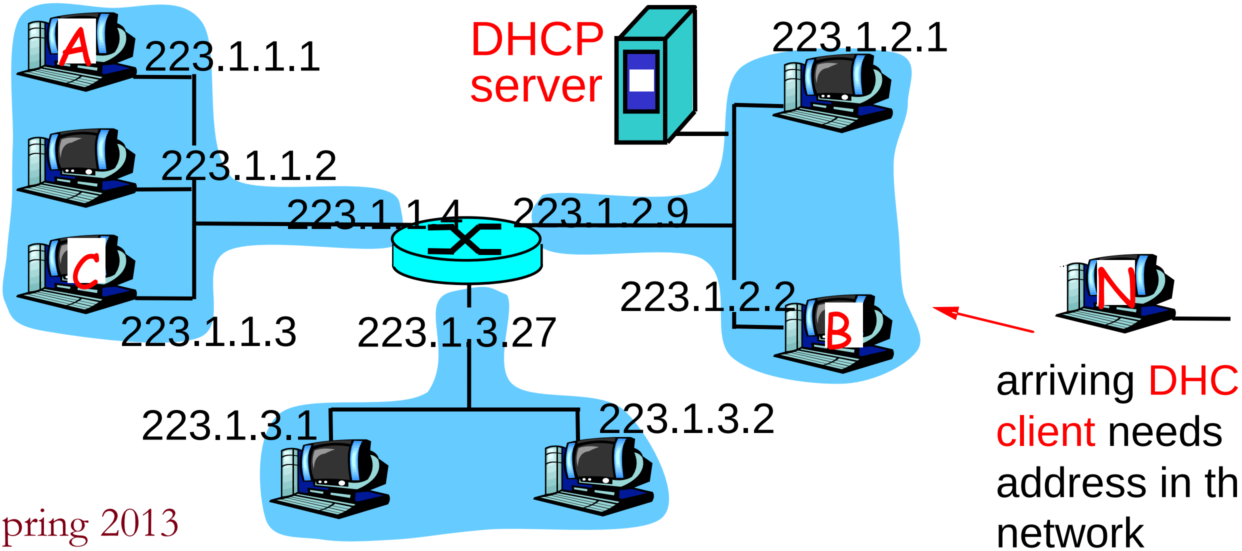
- DHCP server will send (offer) an IP address to the host which requested it (here N) in the offer phase
- Offer phase packet by DHCP server: source IP address is DHCP's IP, and destination's IP address is still 255.255.255.255
- However inside the offer message: DHCP server offers N the IP address x.y.z.w and a lease time, e.g., 24 hours
- This is just an offer, now if N wants it, it needs to request it
- Note that multiple DHCP servers may respond to N, so N will have multiple offers with possibly different lease times

DHCP – Request Phase

- Request phase: The new client (N) sends a packet (broadcasts the request) in case N received multiple DHCP offers; however if there was only one offer, then destination IP address is the DHCP server's IP address
- In case of multiple DHCP offers, N needs to broadcast the request to let all DHCP servers know which offer N is choosing, so that other DHCP servers whose offers, N did not select, can take the IP address they offered to N, back to the pool of the available IP addresses
- The frame is broadcasted
- The request message that N sends implies that N likes the offer (IP address: x.y.z.w with the offered lease time,) however the IP is not N's yet. DHCP needs to send an acknowledgement to N, and then that IP is finally N's

DHCP - Acknowledge Phase

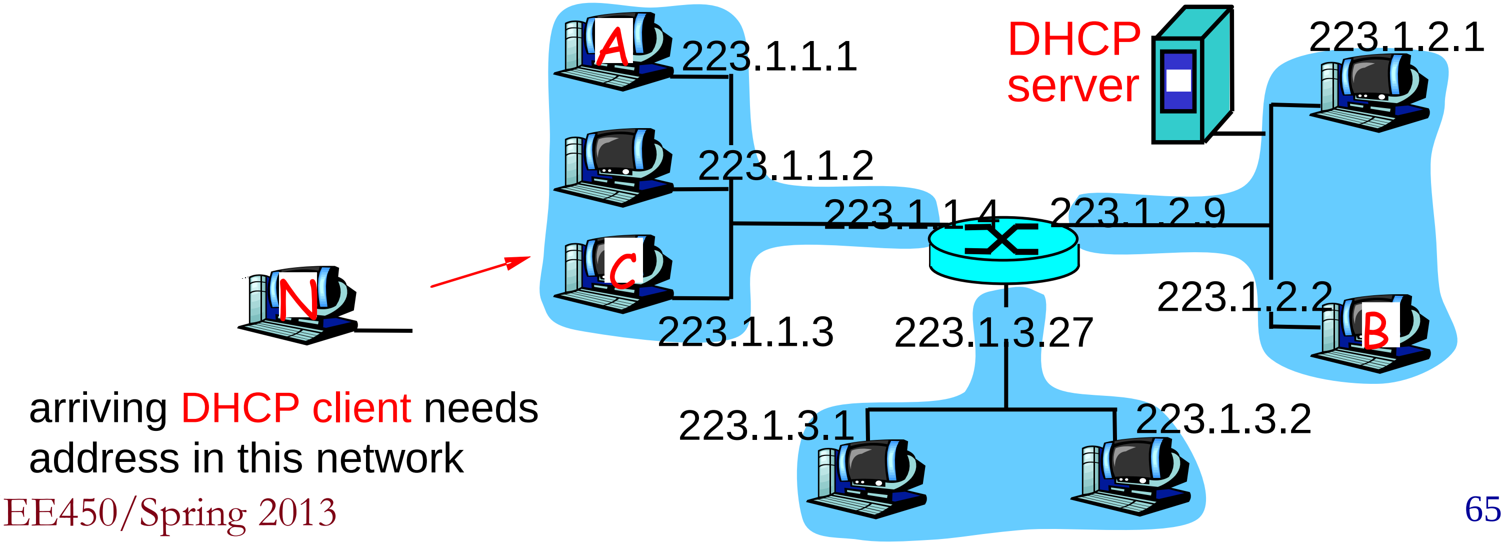
- ACK-phase is by the DHCP server. The packet has source IP address of DHCP server and destination is still a broadcast address. The message implies to N that the IP address that was offered and N requested is given to N. N can then start using that certain IP
- Question: What happens if the new client, N, is not in DHCP's network? (the typical case for new residential customers)



arriving DHCP client needs address in this network

DHCP Sever and New Client on Different Networks

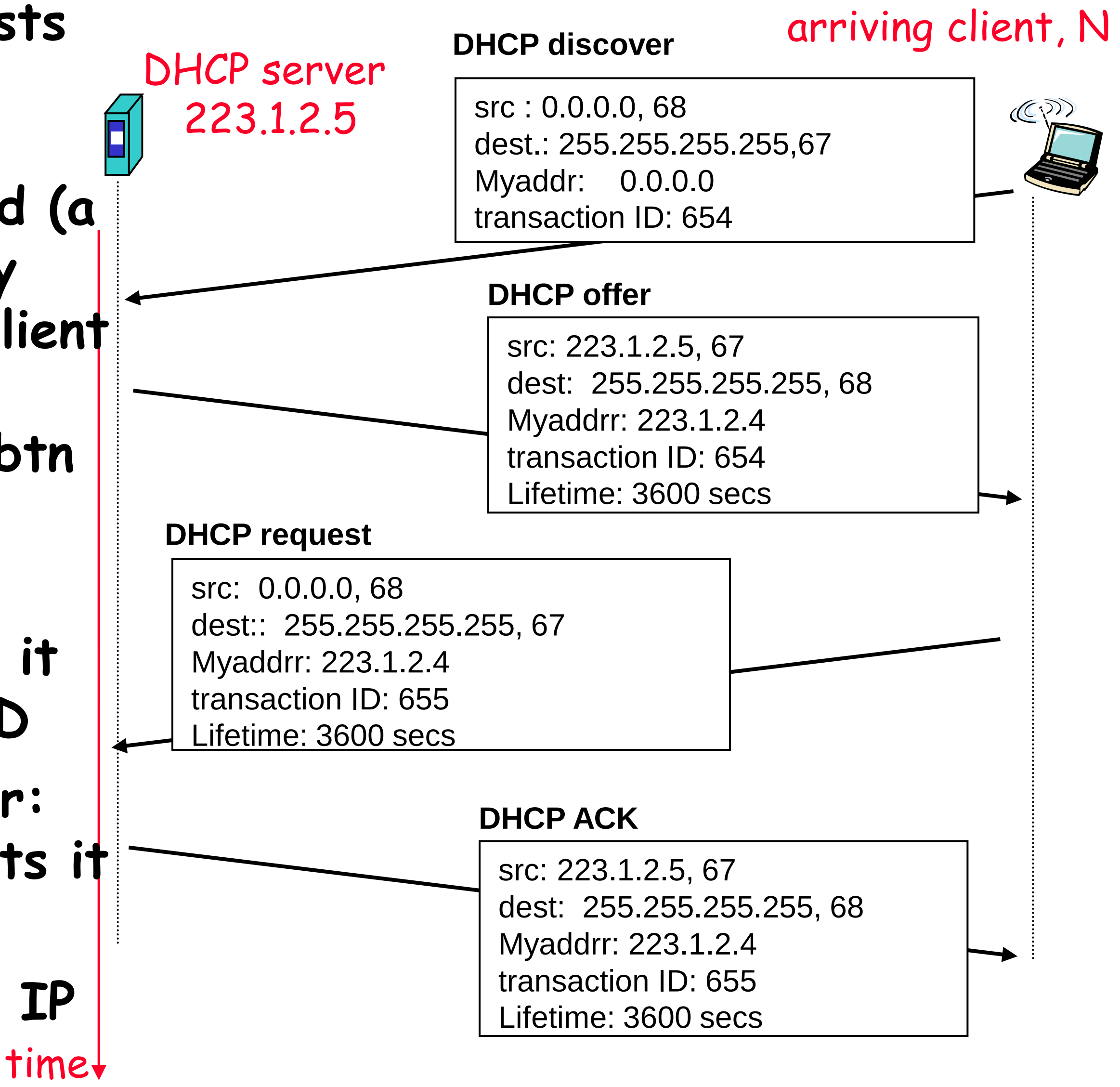
- The new client's discovery packet (that was broadcasted) won't be received by the DHCP server, but the router receives the message that N is trying to discover DHCP server's IP
- The router then relays the message but unicasts the message to the DHCP server, because the router has the IP address of the DHCP server that is configured in it
- The DHCP replies, then the router relays that to N, so it's as if the router is a proxy router standing in btn N and DHCP server
- Most of the times the DHCP service is implemented in the router itself, meaning the router will assign the hosts, the IP addresses



arriving DHCP client needs address in this network

DHCP Client-Server Scenario

- It is possible multiple hosts request IP addresses simultaneously, so a transaction ID is required (a random number chosen by the client, used by the client and server to associate messages and responses btn them)
- DHCP offer is then broadcasted as well, and it echoes the transaction ID
- In DHCP request: Myaddr: 223.1.2.4, means N wants it
- In DHCP ACK: Myaddr: 223.1.2.4 means it's N's IP from now on



DHCP discover

src : 0.0.0.0, 68
dest.: 255.255.255.255,67
Myaddr: 0.0.0.0
transaction ID: 654

DHCP offer

src: 223.1.2.5, 67
dest: 255.255.255.255, 68
Myaddrr: 223.1.2.4
transaction ID: 654
Lifetime: 3600 secs

DHCP request

src: 0.0.0.0, 68
dest:: 255.255.255.255, 67
Myaddrr: 223.1.2.4
transaction ID: 655
Lifetime: 3600 secs

DHCP ACK

src: 223.1.2.5, 67
dest: 255.255.255.255, 68
Myaddrr: 223.1.2.4
transaction ID: 655
Lifetime: 3600 secs

DNS (Domain Name System)

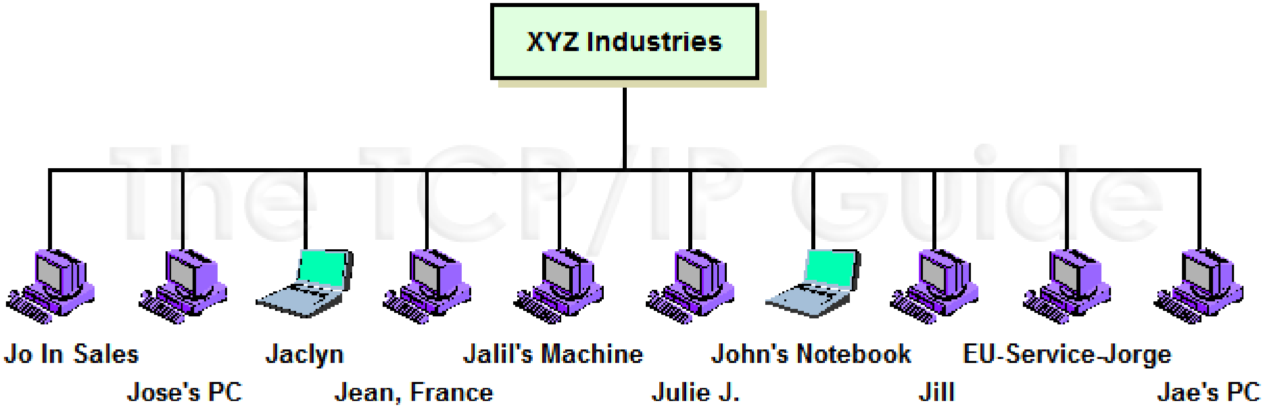
- A host (end-system) obtains the IP address of another host using DNS, a TCP/IP client-server type of service that mainly runs over UDP with well-known port number of 53. For DNS transactions that are UDP based, connection is not required
- Note that if the host knows the IP address of the destination host, DNS is not required, however we prefer to memorize host names rather than their IP addresses, so we need DNS directory
- Example: When you as the end user type http://www.cnn.com/travel.html in the web browser, you are using WWW application and by "http" you imply the application protocol which runs over TCP, i.e., connection oriented; www.cnn.com is the end system name and travel.html is just a file name. The first thing to find is the IP address of the end system www.cnn.com, so travel.html is not the focus yet, but where travel.html is located (the end system) is the focus and it's IP should be found. The browser asks the DNS server for the IP address of www.cnn.com (the DNS client communicates with the DNS server)

DNS (Cont.)

- In order to communicate with the DNS server, the end system client needs the IP address of the DNS server, which is already configured in it, statically or dynamically
- DNS gives service to applications. DNS interacts with application protocols, such as http, and not the end user, therefore the user can enter HTTP, FTP, SMTP, Telnet but not the DNS. If the DNS server has the answer, it replies to the client with the IP address of www.cnn.com
- In that case, the DNS server will not be involved any longer and the next step is to setup the TCP connection with the HTTP server (HTTP well-known port number is 80)
- Only after TCP connection is set up user can use the http
- Question: What if the DNS server does not have the IP address of www.cnn.com? DNS uses a distributed hierarchical (tree-like) structure (similar to Internet itself)
- DNS defines several domains, each domain is a collection of related sites. DNS is distributed across the Internet

Name Space – Flat

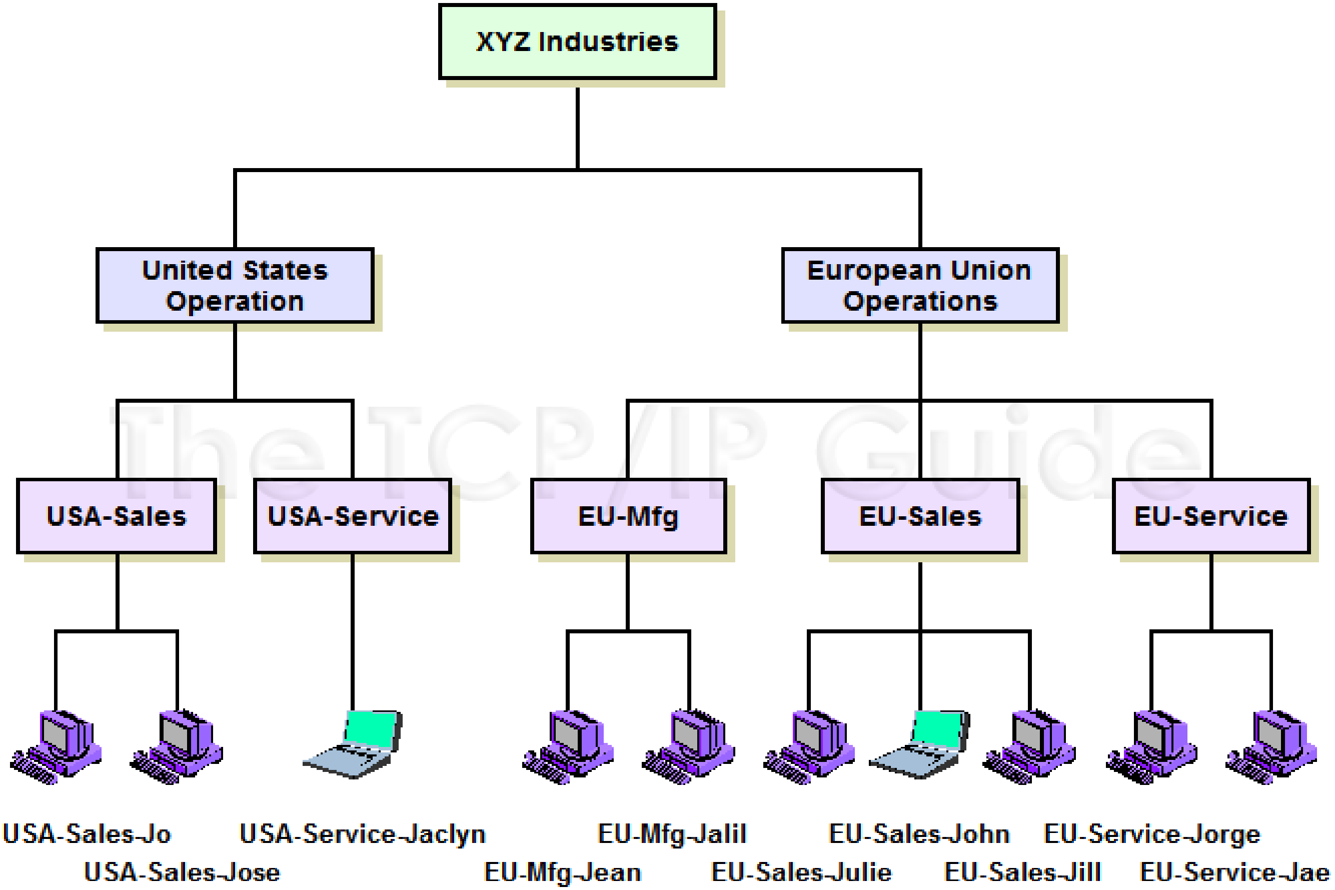
- To be unambiguous, the names assigned to machines (the name to the IP address mapping) must be unique
- A name space that maps each IP address to a unique name can be organized in two ways flat and hierarchical
- In a flat name space, a name is assigned to an address. A name in this space is a sequence of characters without structure. The names may or may not have a common section: if they do, it has no meaning. The main disadvantage of a flat name space is that it cannot be used in a large system such as the Internet because it must be centrally controlled to avoid ambiguity and duplication



An example of a flat name architecture. There is no structure that organizes the names or dictates how they must be constructed. Logically, each device is a peer of each of the others

Name Space – Hierarchical (Structured)

- In this type, each name is made of several parts: 1st part defines the nature of the organization, 2nd the name of the organization, 3rd the department in the organization, and so on



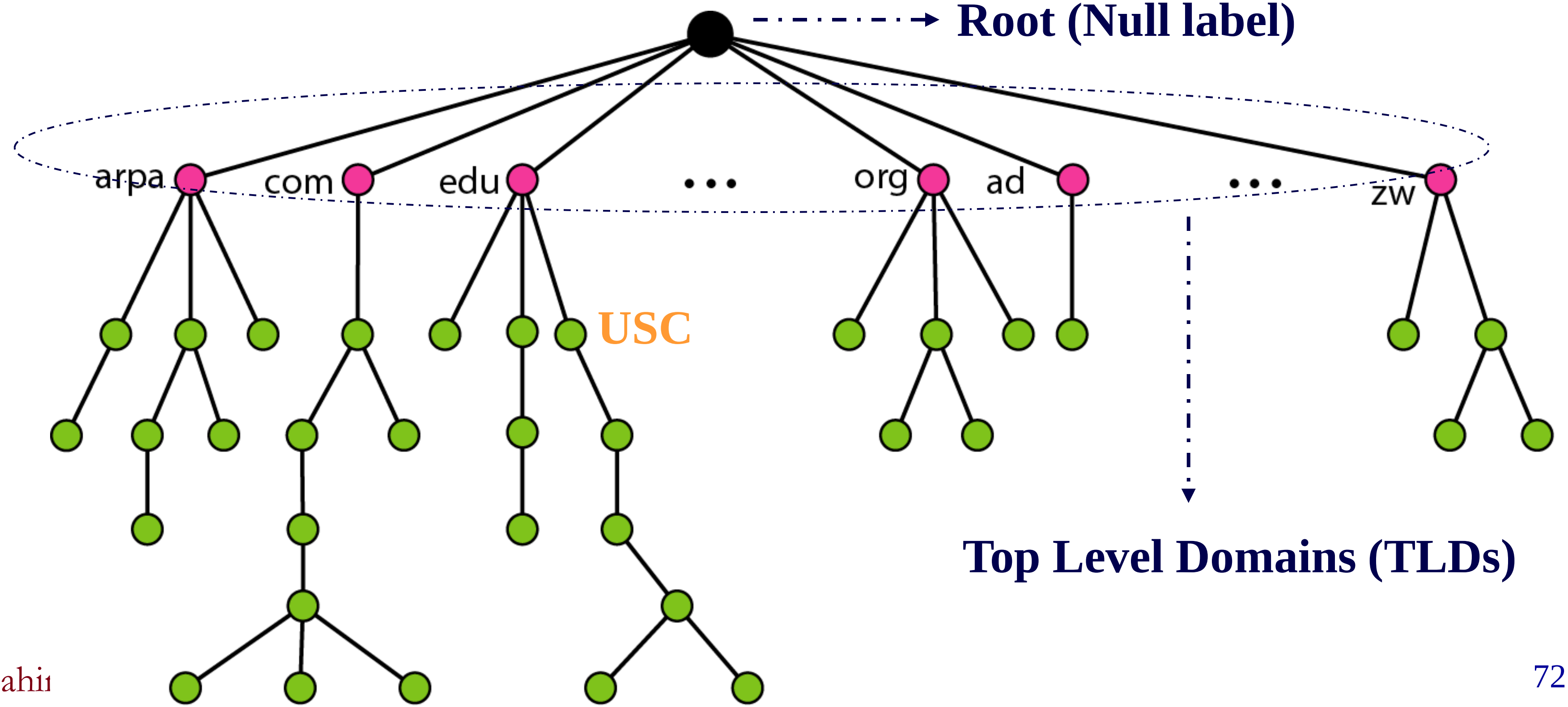
The TCP/IP Guide

Hierarchical Name Space (Cont.)

- The authority to assign and control the namespaces can be decentralized, yet a central authority can assign the parts that define the nature and name of the organization
- The responsibility of the rest of the name can be given to the organization itself
- The organization can add **prefix or suffix** to the name to define its host or resources without worrying that the suffix or prefix chosen for host might be taken by another organization
- To have a hierarchical name space, a **domain name space** was designed

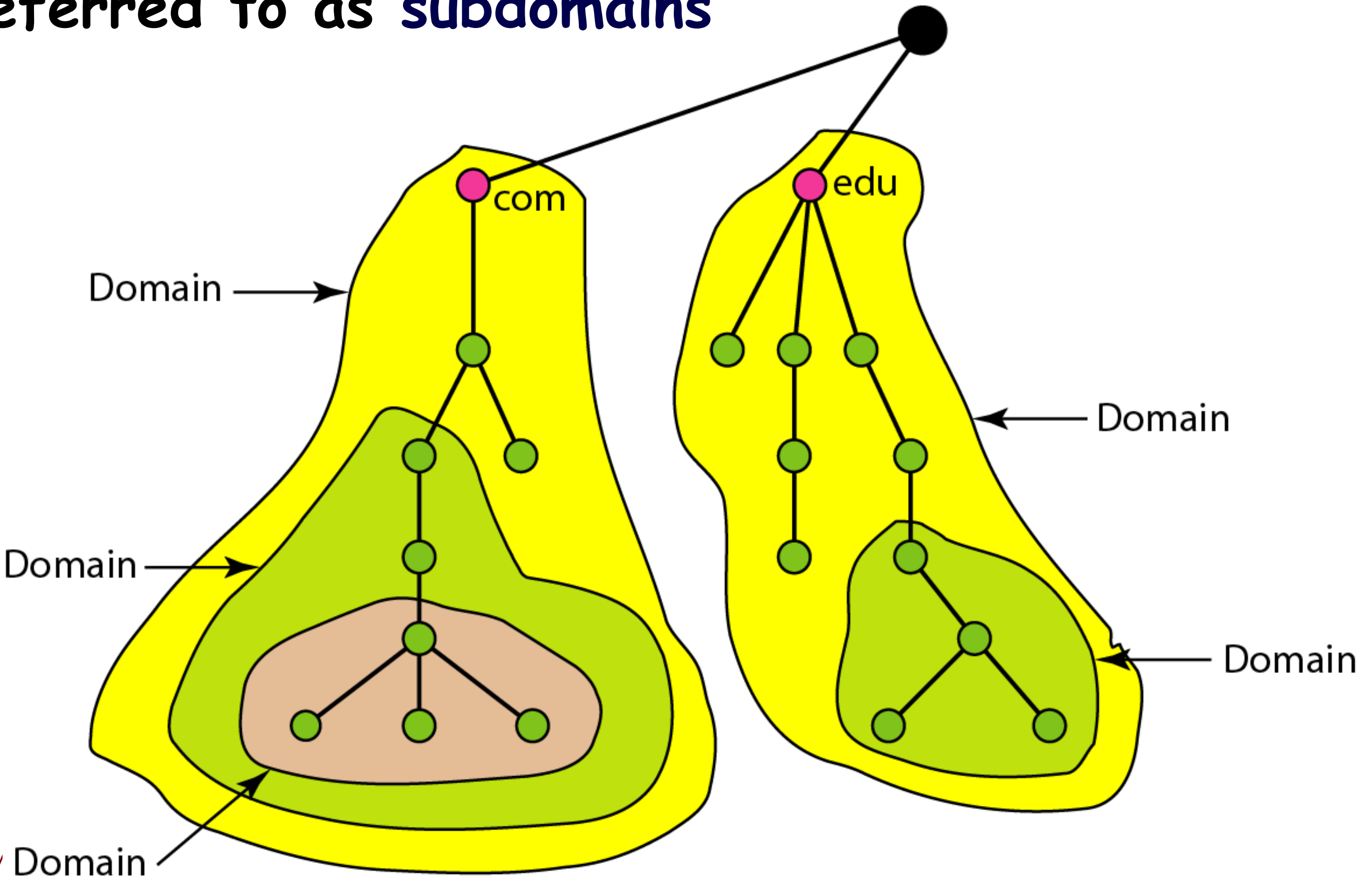
Domain Name Space

- Each node in the tree has a label (a string of max 63 chars)
- The root label is a null string. To guarantee name uniqueness, DNS requires the children of a certain node have different labels
- Each node in the tree has a domain name. A full domain name is a sequence of labels separated by dots (.) from node up to the root
- Therefore a full domain name always ends in a null label

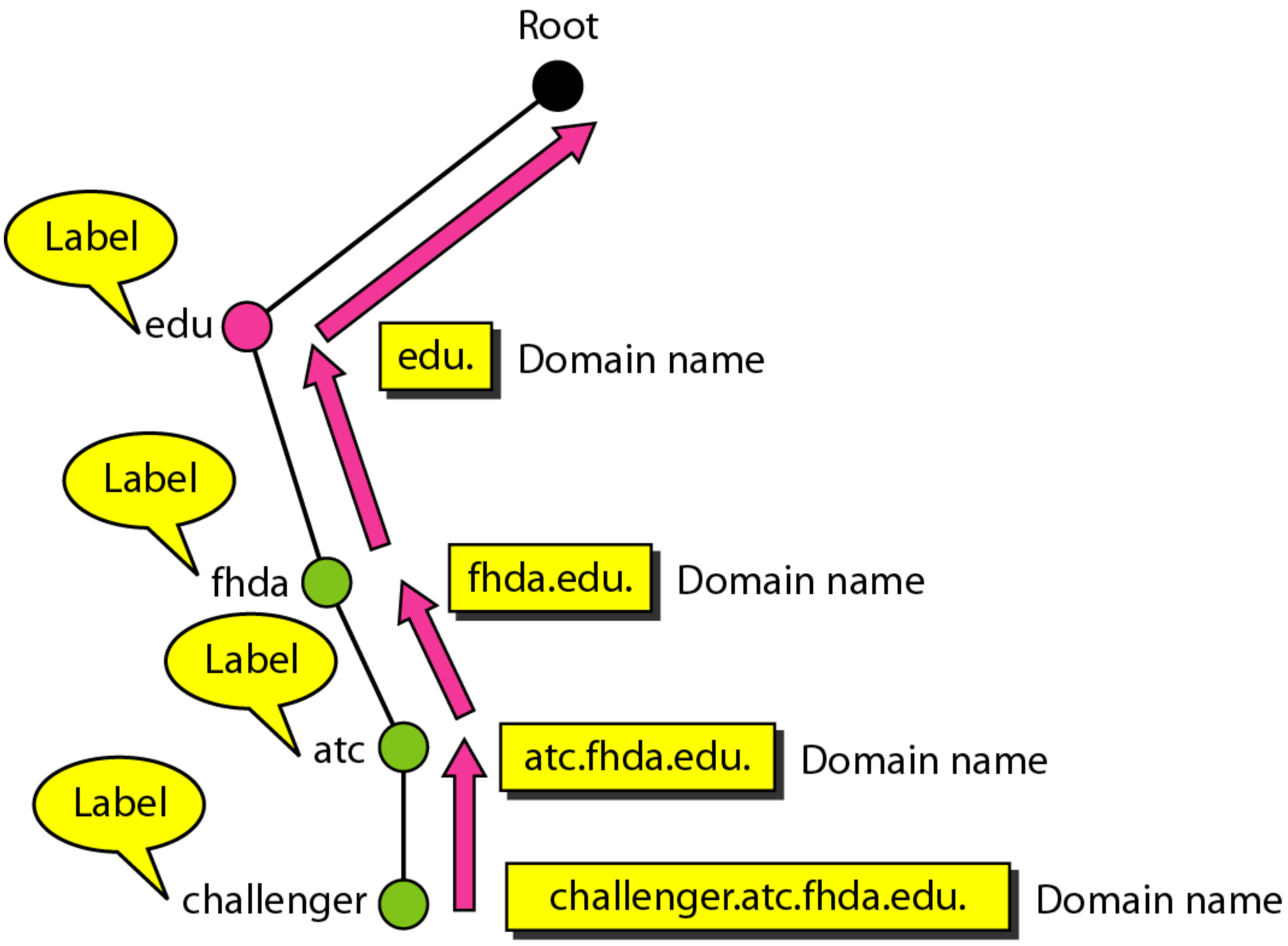


DNS – Domains

- A domain is a subtree of the domain name space
- The name of the domain is the domain name of the node at the top of the subtree
- A domain can hence be a superset of some domains, which may be referred to as subdomains

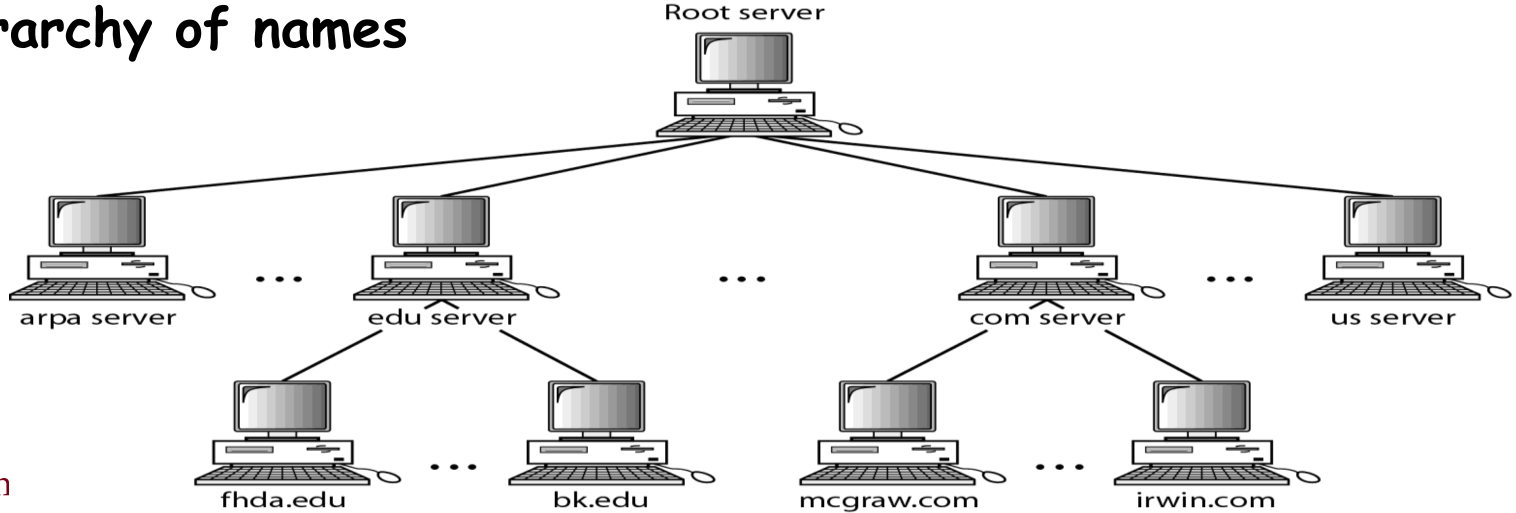


Domain Names and Labels – Example



Hierarchy of Name Servers

- It is inefficient to have only one computer store the whole domain name space as DNS requests from all over the world place a heavy load on the system. It is also unreliable as any system failure makes the service inaccessible. Therefore the information is distributed among many computers called DNS servers
- One way is to let the root stand alone and create as many domains (subtrees) as there are 1st-level nodes. A domain could be very large, therefore DNS allows domains to be divided further into smaller domains (i.e., subdomains). Each server can be responsible (authoritative) for either a large or a small domain. In other words there is a hierarchy of servers in the same way that as there is a hierarchy of names



Zone

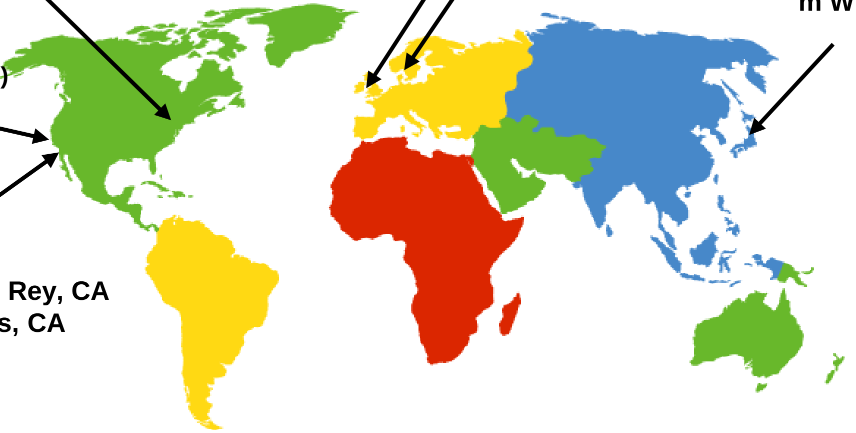
- What a server is responsible for or has authority over is called a **zone**
- If a server accepts the responsibility for a domain and does not divide the domain into smaller domains, the domain and the zone refer to the same thing, otherwise they are different
- Each server makes a database called a zone file and keeps all the info for every node under that domain
- The information about the nodes in the subdomain is stored in the servers at lower levels with the original server keeping some sort of reference to these lower level servers
- A root server is a server whose zone consists of the whole tree
- A root tree typically does not store any info about domains, but delegates its authority to other servers, keeping references to those servers
- There are several root servers (distributed around the world) each covering the whole domain name space

Name Servers

- Root (name) servers: If the local name server does not have the IP address queried by a client, it acts as a DNS client and queries one of the root servers. There are 13 root name servers which are mostly in the US
- Top-level domain (TLD) servers: are responsible for com, org, net, edu, etc, and all top-level country domains uk, fr, ca, jp. [e.g., the company Network Solutions maintains TLD servers for com and the company Educause for edu]
- Authoritative name server: It is where the end system registers its name and IP address. It's the organization's DNS servers, providing authoritative hostname to IP mappings for organization's servers (e.g., Web, mail) and is maintained by organization or service provider
- Local name servers: this is the default name server in client's network, or the network which is closest to the client and the first name server the client interacts with for DNS queries. The IP address of the local name server is configured in the client machine, either statically or dynamically. For residential customers the local name server is typically at the local office of the service provider

DNS – Root Name Servers

- Contacted by local name server that can not resolve name
- Root name server:
  - Contacts authoritative name server if name mapping not known
  - Gets mapping
  - Returns mapping to local name server



a Verisign, Dulles, VA
c Cogent, Herndon, VA (also LA)
d U Maryland College Park, MD
g US DoD Vienna, VA
h ARL Aberdeen, MD
j Verisign, (21 locations)

k RIPE London (also 16 other locations)

i Autonomica, Stockholm (plus 28 other locations)

m WIDE Tokyo (also Seoul, Paris, SF)

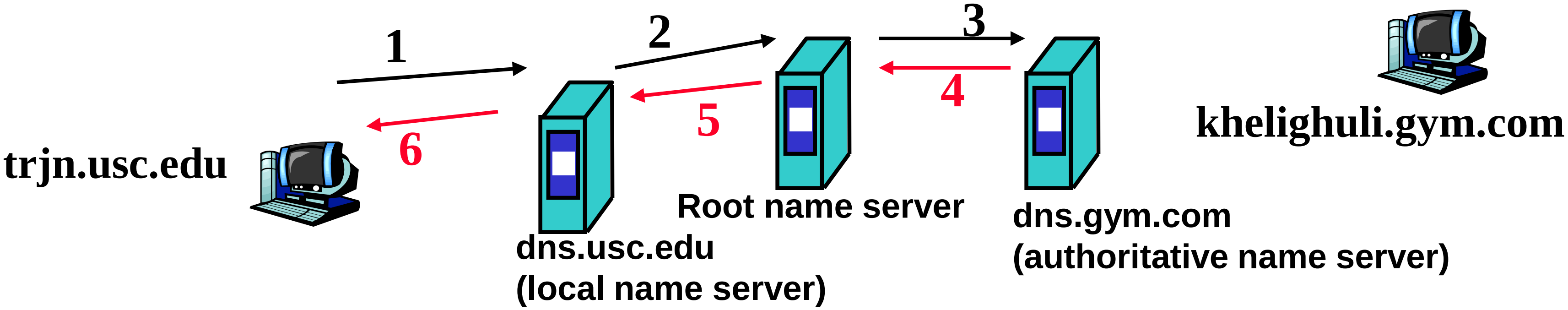
e NASA Mt View, CA
f Internet Software C. Palo Alto, CA (and 36 other locations)

b USC-ISI Marina del Rey, CA
l ICANN Los Angeles, CA

13 root name servers worldwide

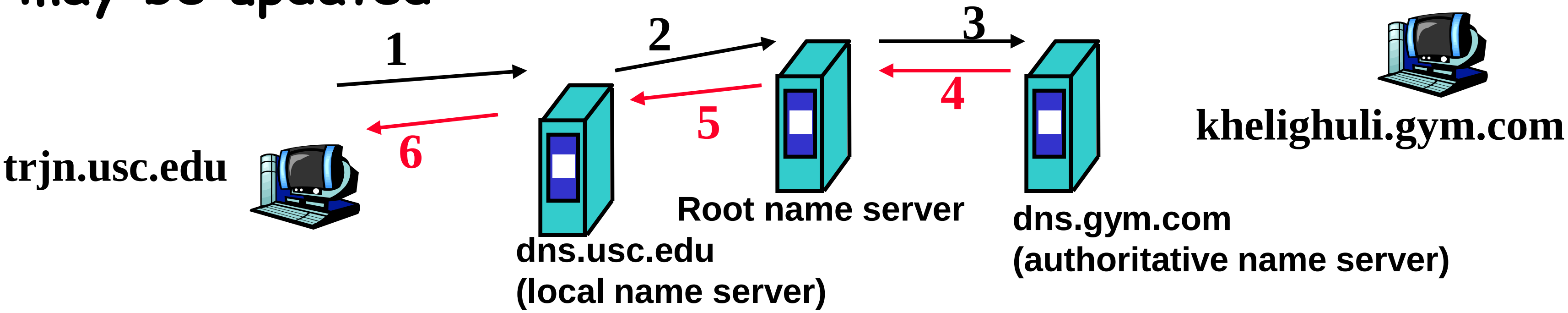
Recursive DNS – Example I

- trjn.usc.edu wants to contact khelighuli.gym.com
- DNS client sends a request to the local name server, dns.usc.edu for khelighuli.gym.com's IP address [1]
- dns.usc.edu does not know the address, so becomes client, queries to the root server server [2]
- The root server doesn't know either so it becomes client, queries the authoritative server, dns.gym.com [3]. khelighuli.gym.com is registered in it

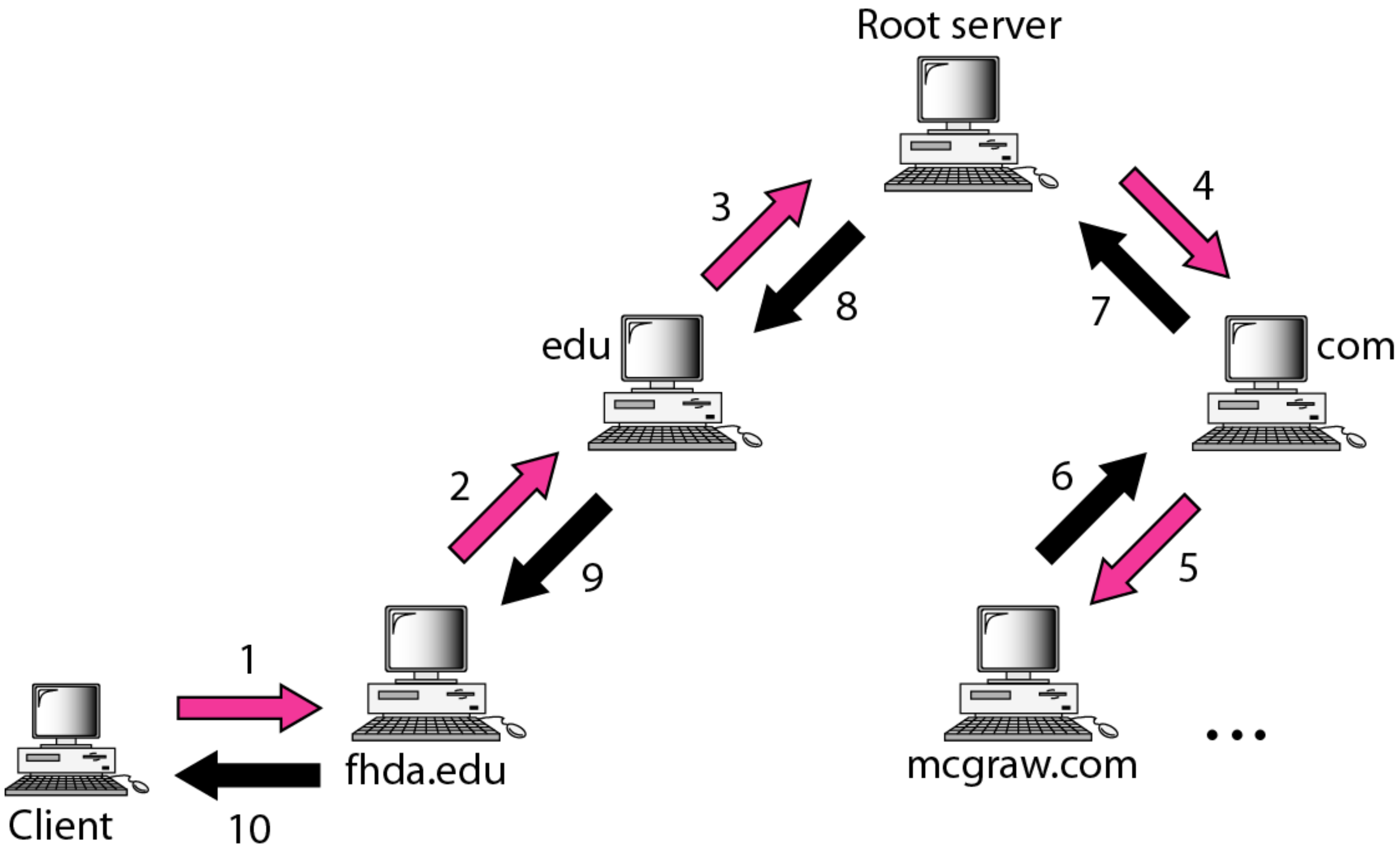


Recursive DNS – Example I (Cont.)

- dns.gym.com responds through the root name server [4], then local name server [5] and finally to client trjn.usc.edu [6]
- When the IP of khelighuli.gym.com gets to the local name server, the local name server can cache it, so in case trjn tries to communicate with khelighuli again later, it won't repeat DNS steps
- The idea of caching also works when you access a website, then some accessed documents are cached in your local server; however the issue is that the original documents may be updated



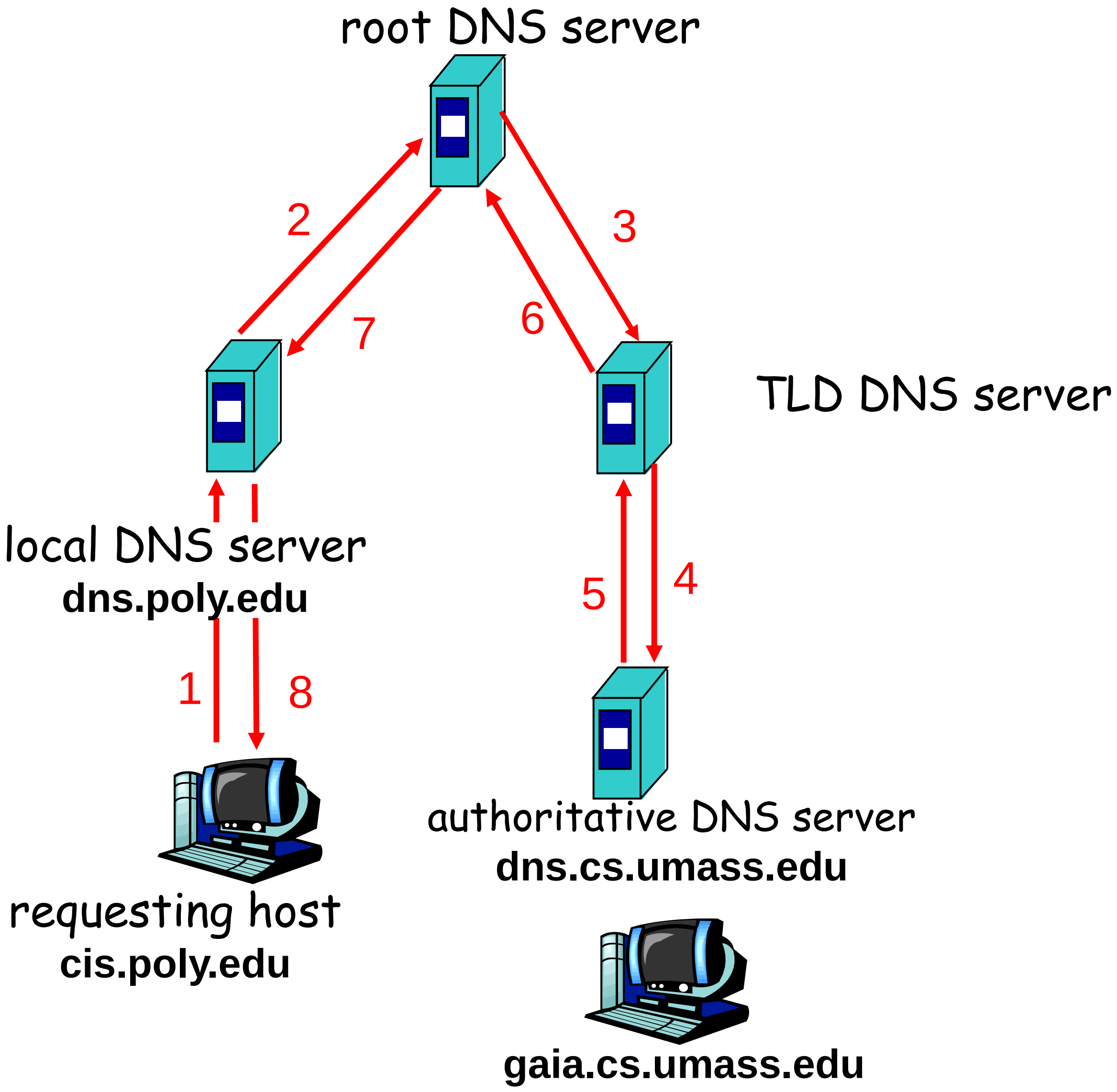
Recursive DNS – Example II



Recursive DNS - Example III

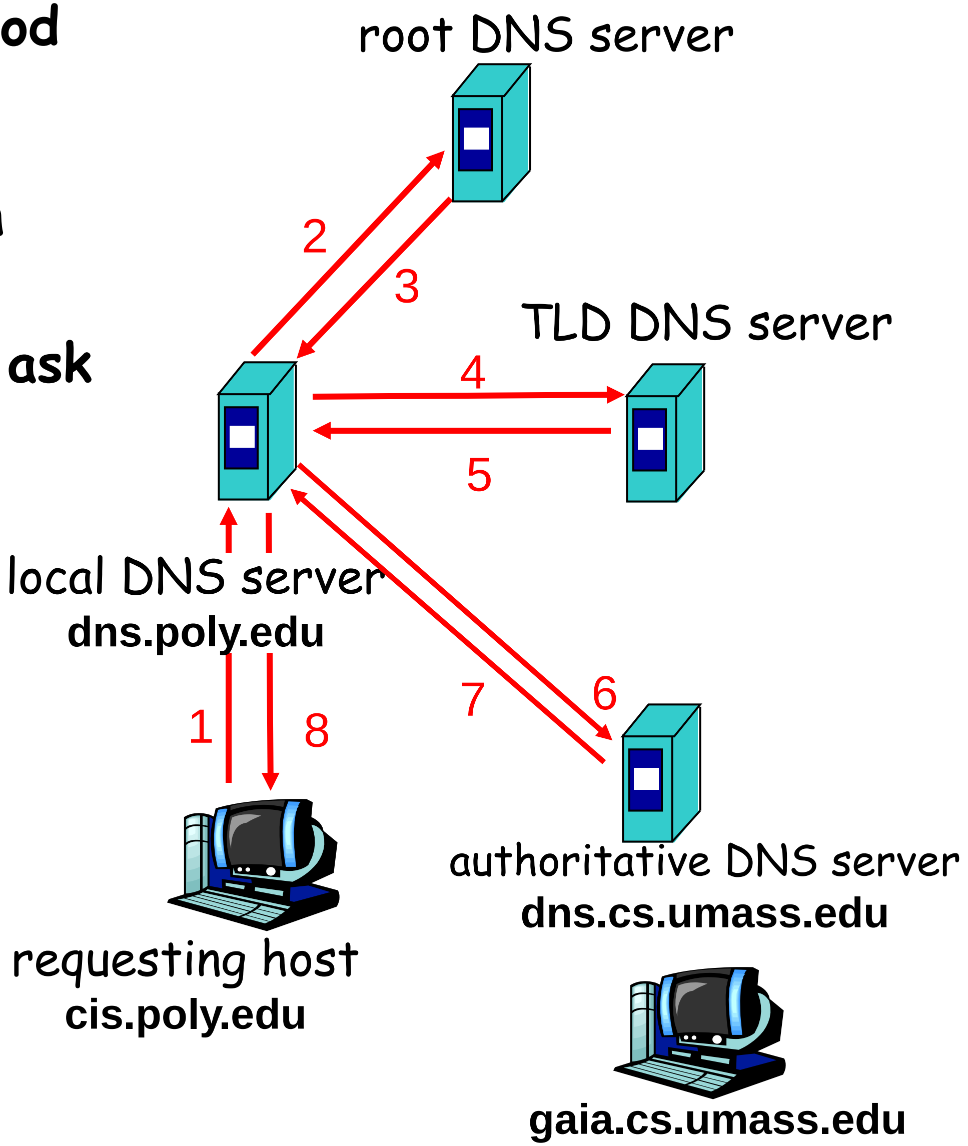
Recursive Query:

- Puts burden of name resolution on contacted name server
- Heavy load?



Iterative DNS

- One issue with recursive DNS is that root name server may have to act as a client which is not good



Iterated query:

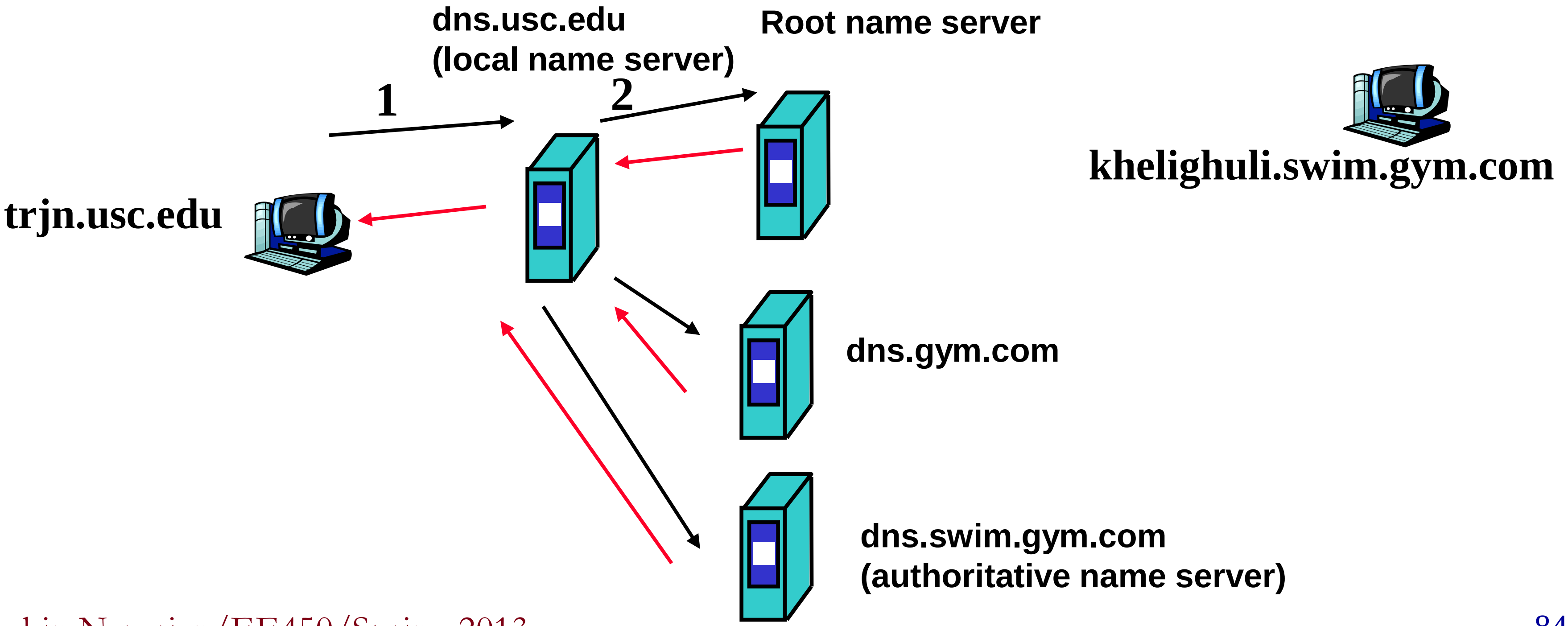
- Contacted server replies with name of server to contact
- "I don't know this name, but ask this server"

- Example: Host at cis.poly.edu wants the IP address of gaia.cs.umass.edu

Iterative DNS (Cont.)

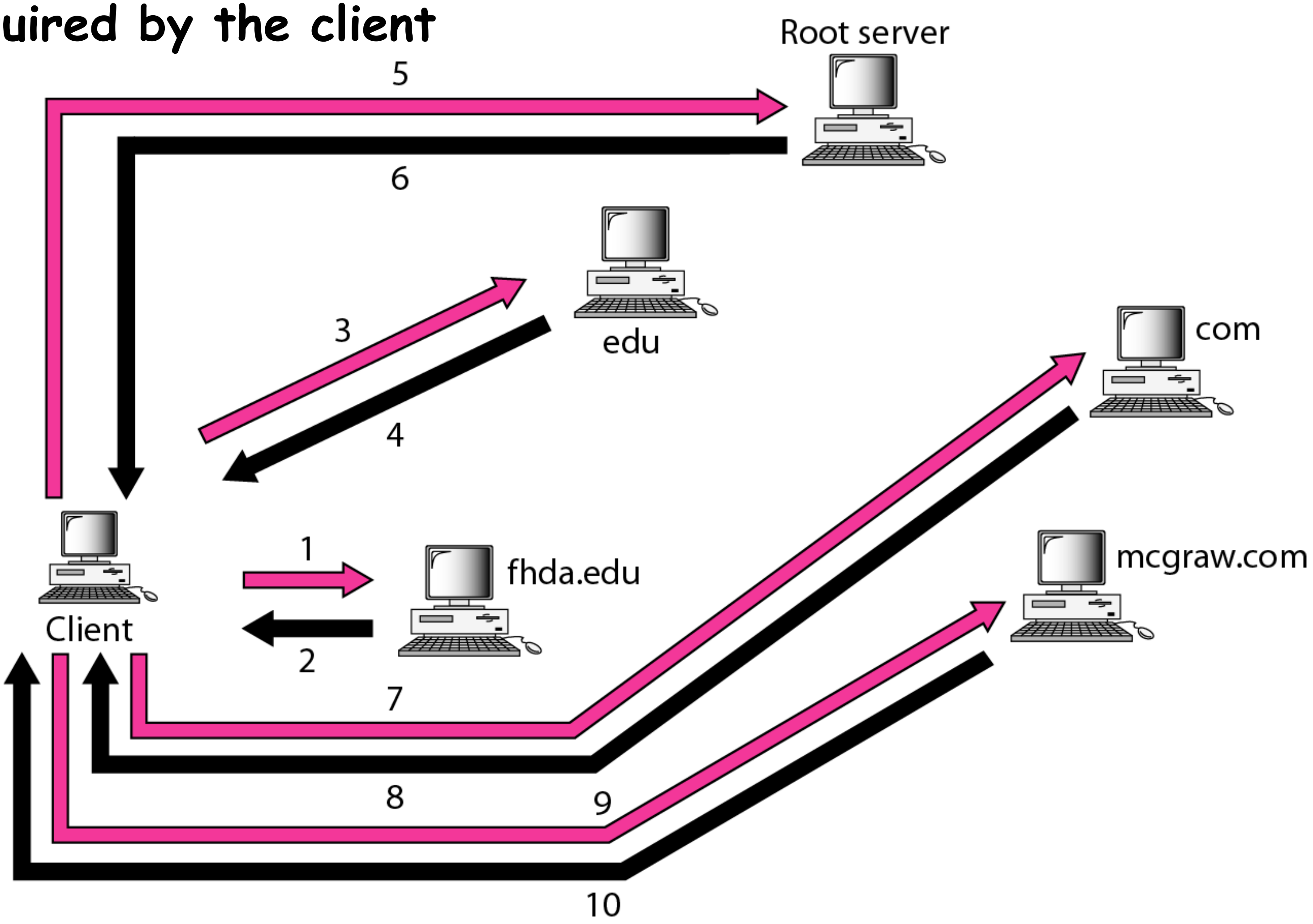
- Note that in iterative DNS, local DNS server is still the interface to be able to cache some of the IP addresses. Compare with pure iterative [next page])

- Exercise: Fill in the iterative DNS step numbers in the following figure



Pure Iterative DNS

- What is the problem with pure iterative approach?
  - The local DNS (fhda.edu) cannot cache the IP that was inquired by the client



DNS: Caching and Updating Records

- Once (any) name server learns mapping, it *caches* mapping
  - Cache entries timeout (disappear) after some time
  - TLD servers are typically cached in local name servers
    - Thus root name servers not often visited

- Optional: Update/notify mechanisms under design by IETF
  - RFC 2136
  - http://www.ietf.org/html.charters/dnsind-charter.html