

# Link Layer

---

6.1 introduction, services

6.2 error detection, correction

6.3 Multiple Access Protocols

- channel partitioning
  - TDMA, FDMA, CSMA
- random access
  - Aloha: pure, slotted
  - CSMA, CSMA/CD

6.4 LANs

- addressing, ARP
- Ethernet
- switches
- VLANs

6.5 link virtualization:  
MPLS

6.6 data center  
networking

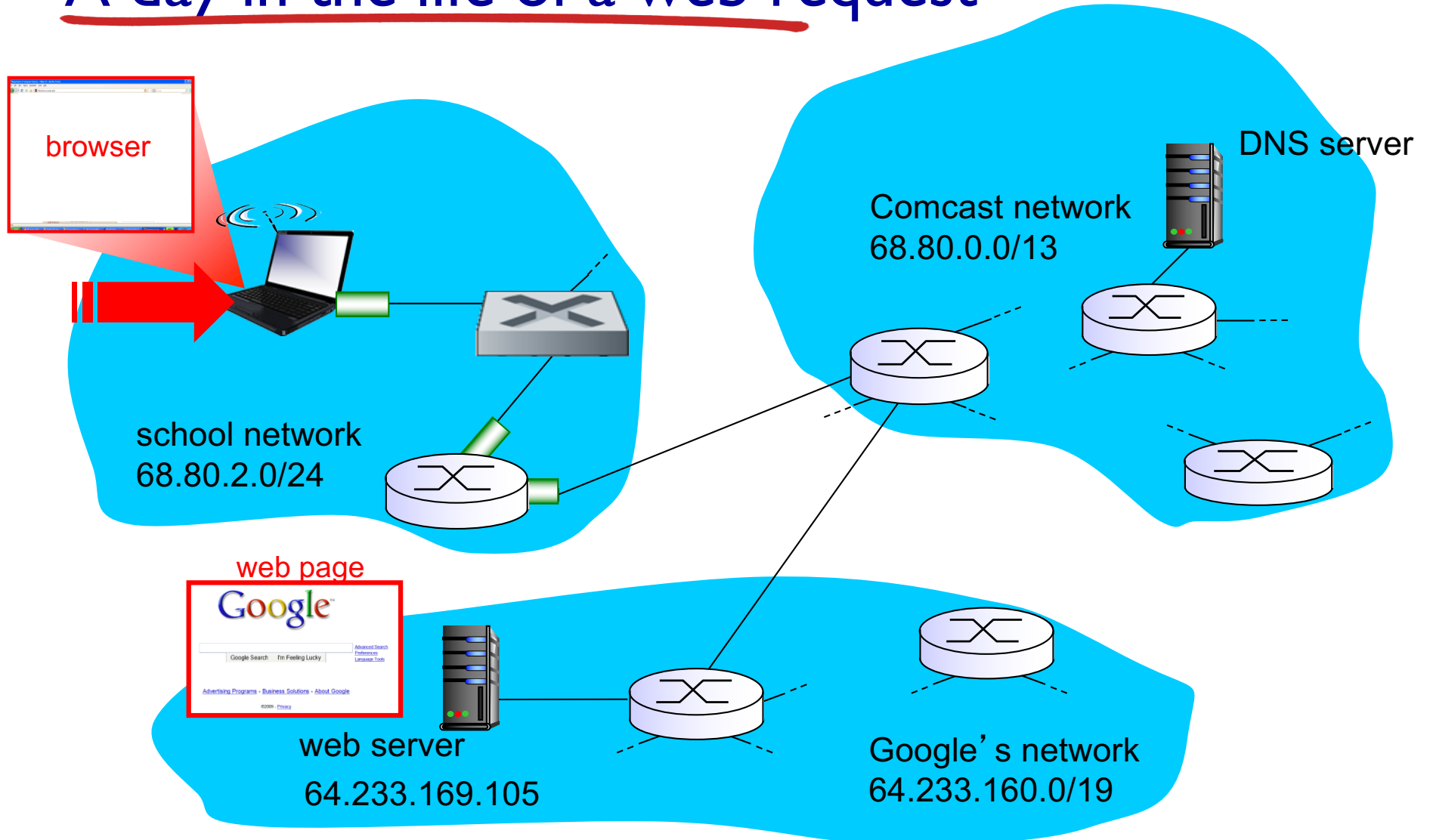
7.3: Detour WiFi

6.7 a day in the life of a  
web request

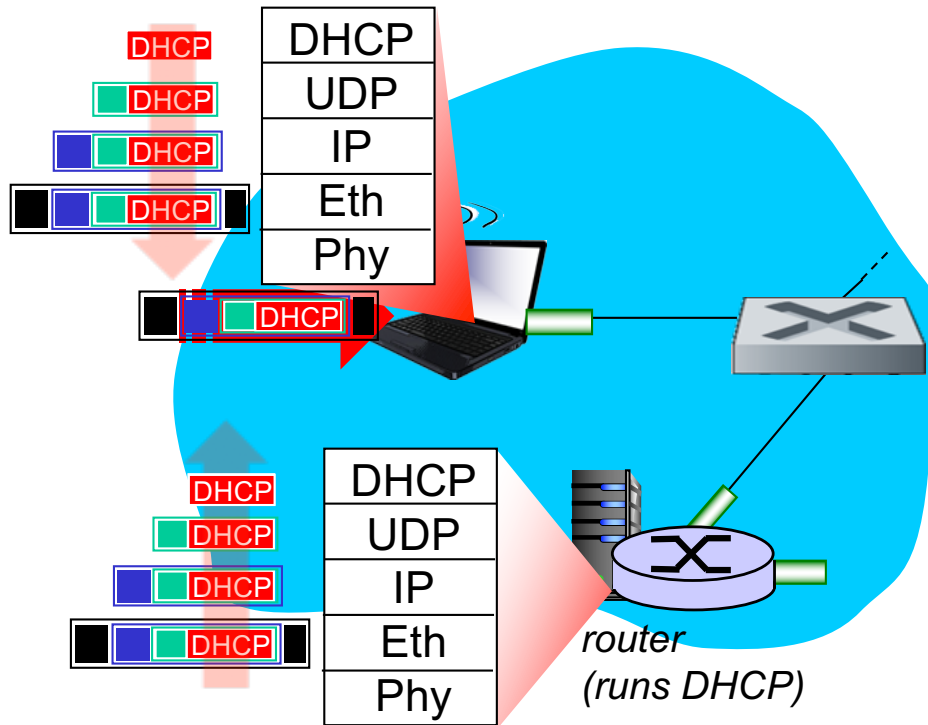
## Synthesis: a day in the life of a web request

- journey down protocol stack complete!
  - application, transport, network, link
- putting-it-all-together: synthesis!
  - *goal*: identify, review, understand protocols (at all layers) involved in seemingly simple scenario: requesting www page
  - *scenario*: student attaches laptop to campus network, requests/receives www.google.com

# A day in the life of a web request

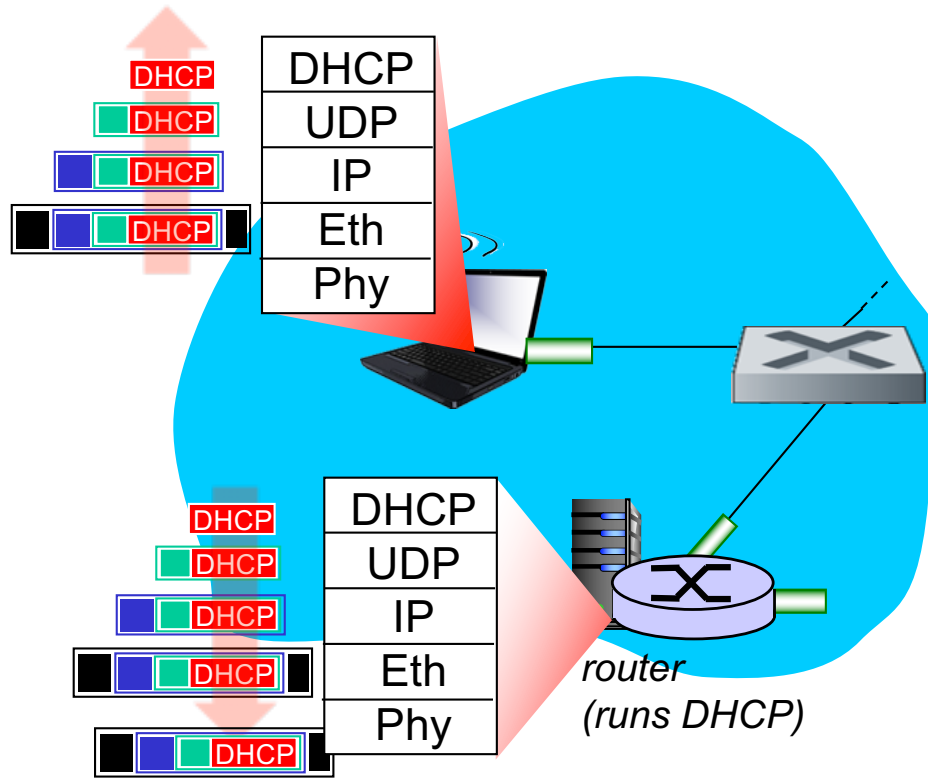


# A day in the life... connecting to the Internet



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

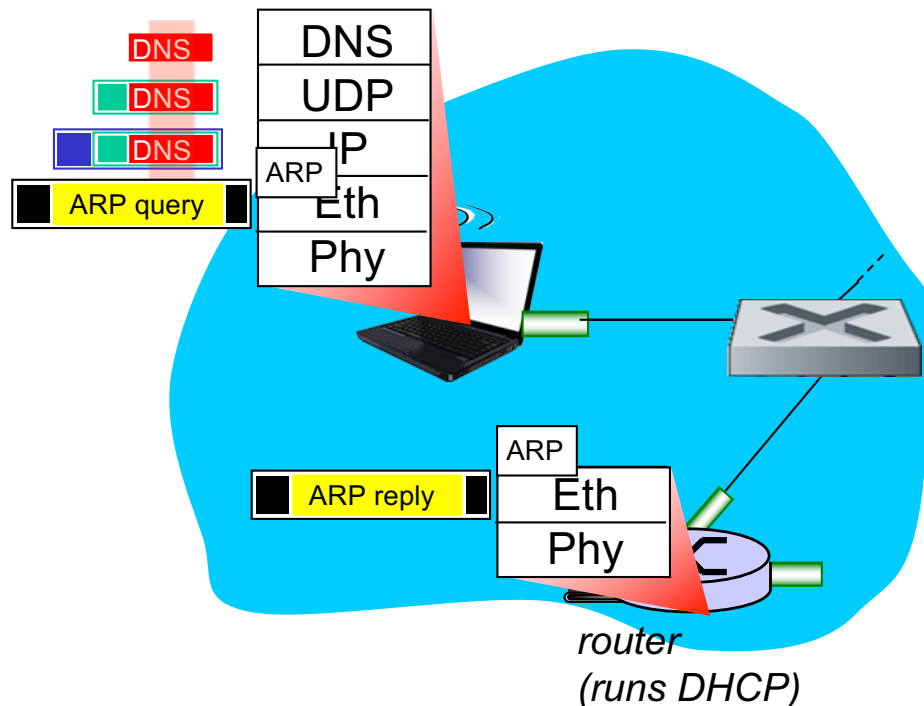
# A day in the life... connecting to the Internet



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

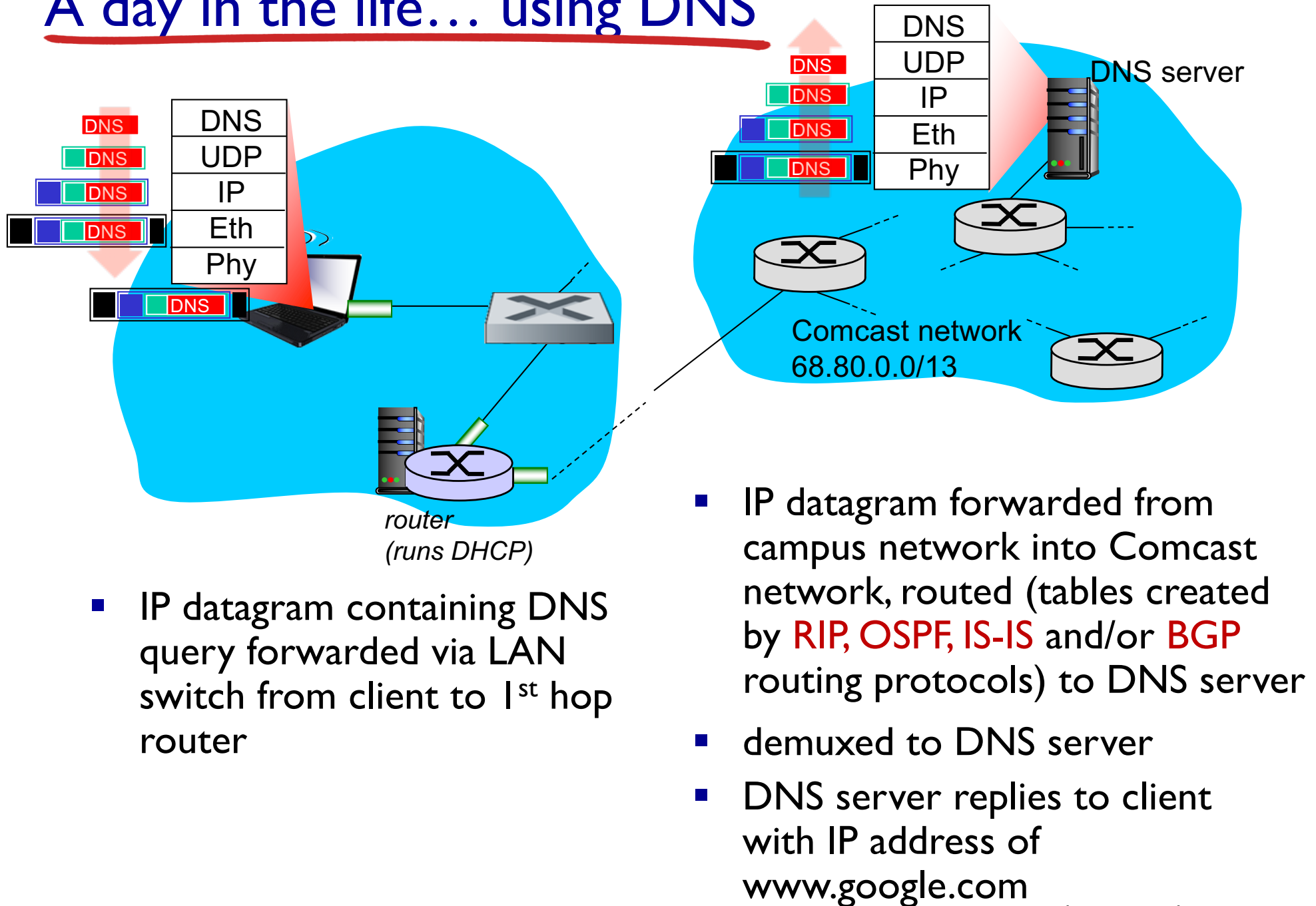
*Client now 1) has IP address 2) knows name & addr of DNS server 3) IP address of its first-hop router*

# A day in the life... ARP (before DNS, before HTTP)

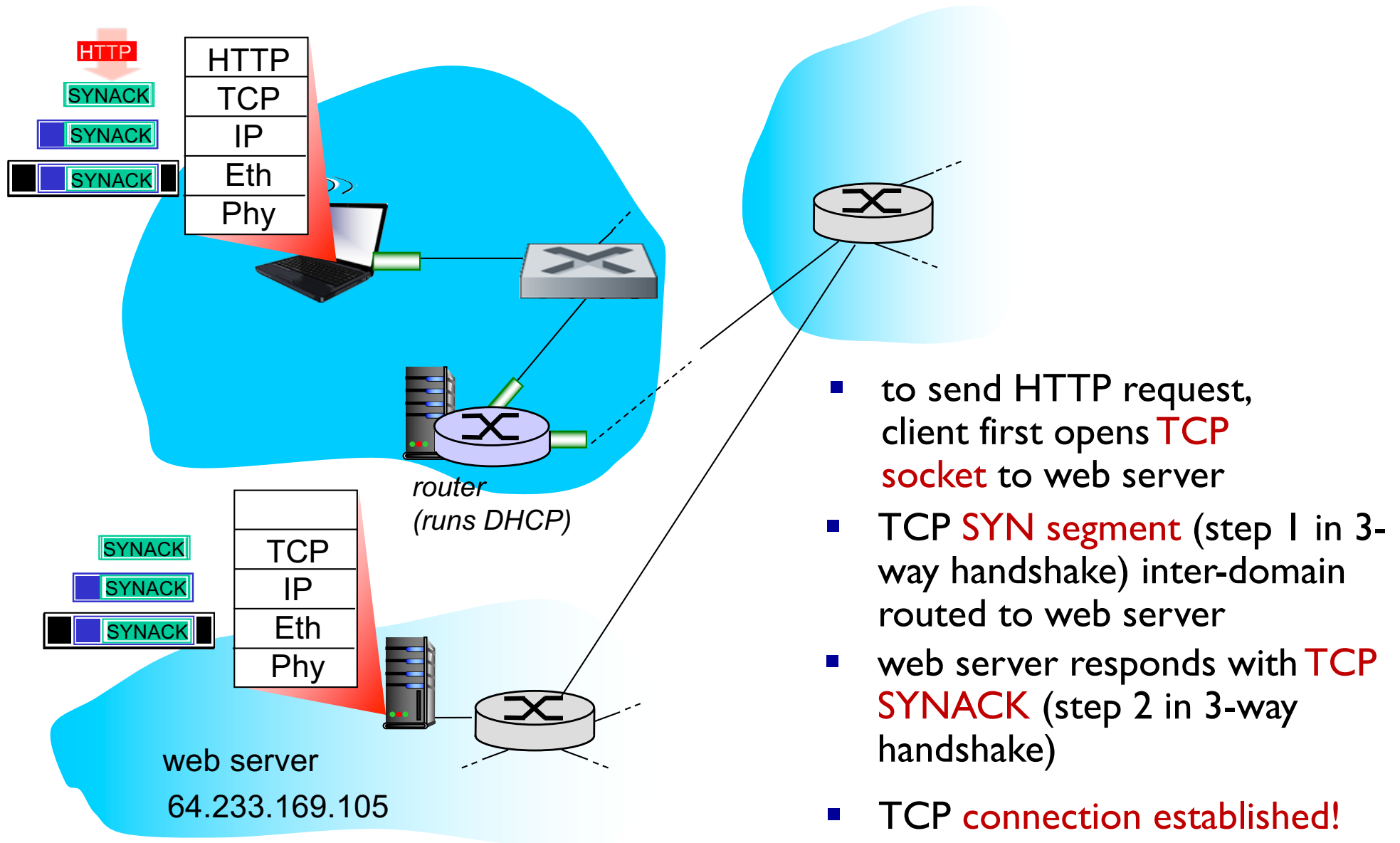


- before sending *HTTP* request, need IP address of `www.google.com`:  
*DNS*
- DNS query created, encapsulated in UDP, encapsulated in IP, encapsulated in Eth. To send frame to router, need MAC address of router interface: *ARP*
- *ARP query* broadcast, received by router, which replies with *ARP reply* giving MAC address of router interface
- client now knows MAC address of first hop router, so can now send frame containing DNS query

# A day in the life... using DNS

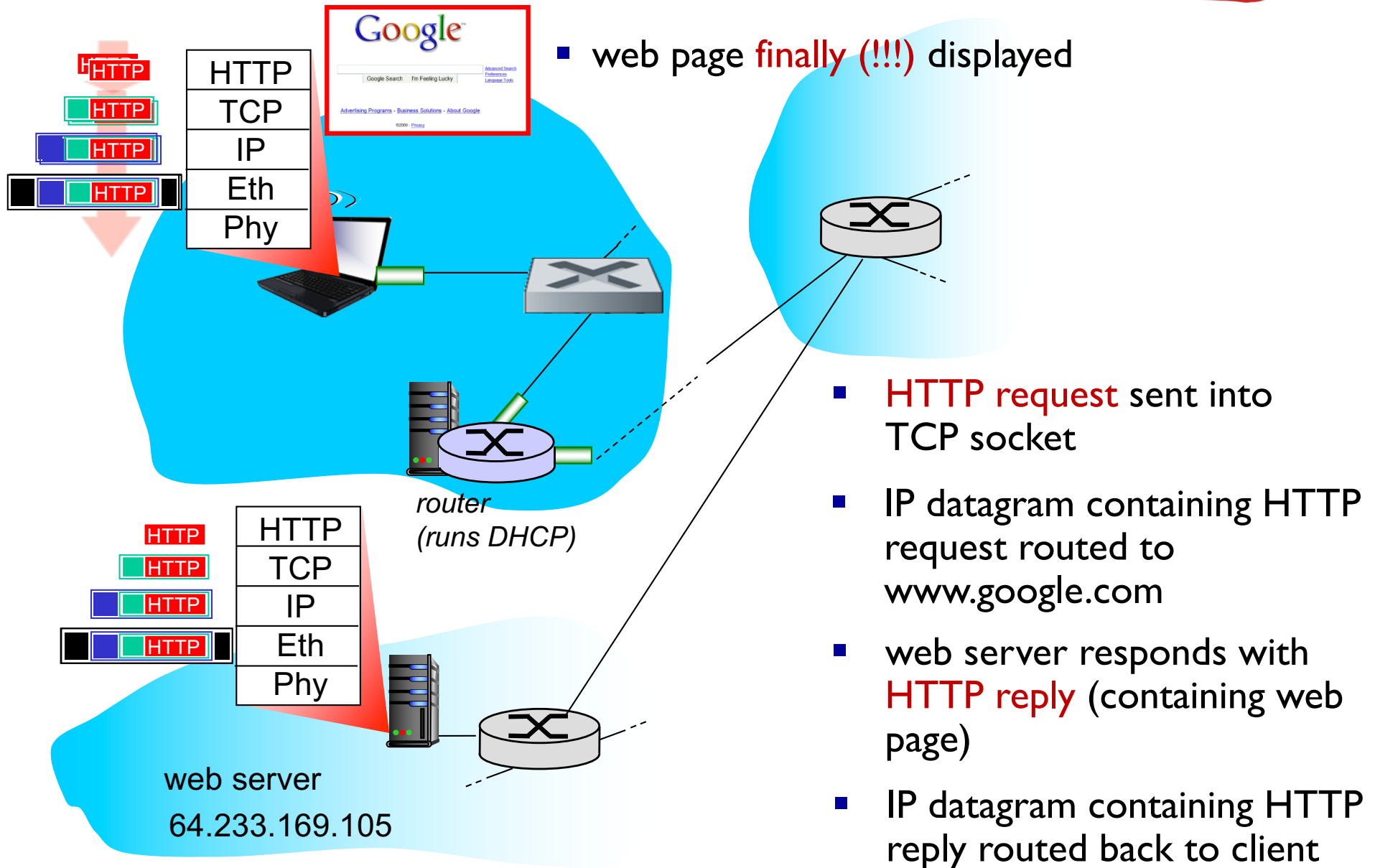


# A day in the life...TCP connection carrying HTTP





# A day in the life... HTTP request/reply



# Chapter 6: Summary

- principles behind data link layer services:
  - error detection, correction
  - sharing a broadcast channel: multiple access
  - link layer addressing
- instantiation and implementation of various link layer technologies
  - Ethernet
  - switched LANS, VLANs
  - virtualized networks as a link layer: MPLS
- synthesis: a day in the life of a web request

# Link layer, LANs: outline

6.1 introduction, services

6.2 error detection,  
correction

6.3 multiple access  
protocols

6.4 LANs

- addressing, ARP
- Ethernet
- switches
- VLANs

6.5 link virtualization:  
MPLS

6.6 data center  
networking

6.7 a day in the life of a  
web request

# Conclusion

---

- What this class is about
  - journey down protocol stack *complete* (except PHY)
  - understanding of networking principles, protocols
- What next....
  - *lots* of interesting topics!
    - wireless, mobile
    - datacenters
    - multimedia, games
    - security
  - Classes@UCI?
    - cs133, cs134; <http://networkedsystems.uci.edu/>