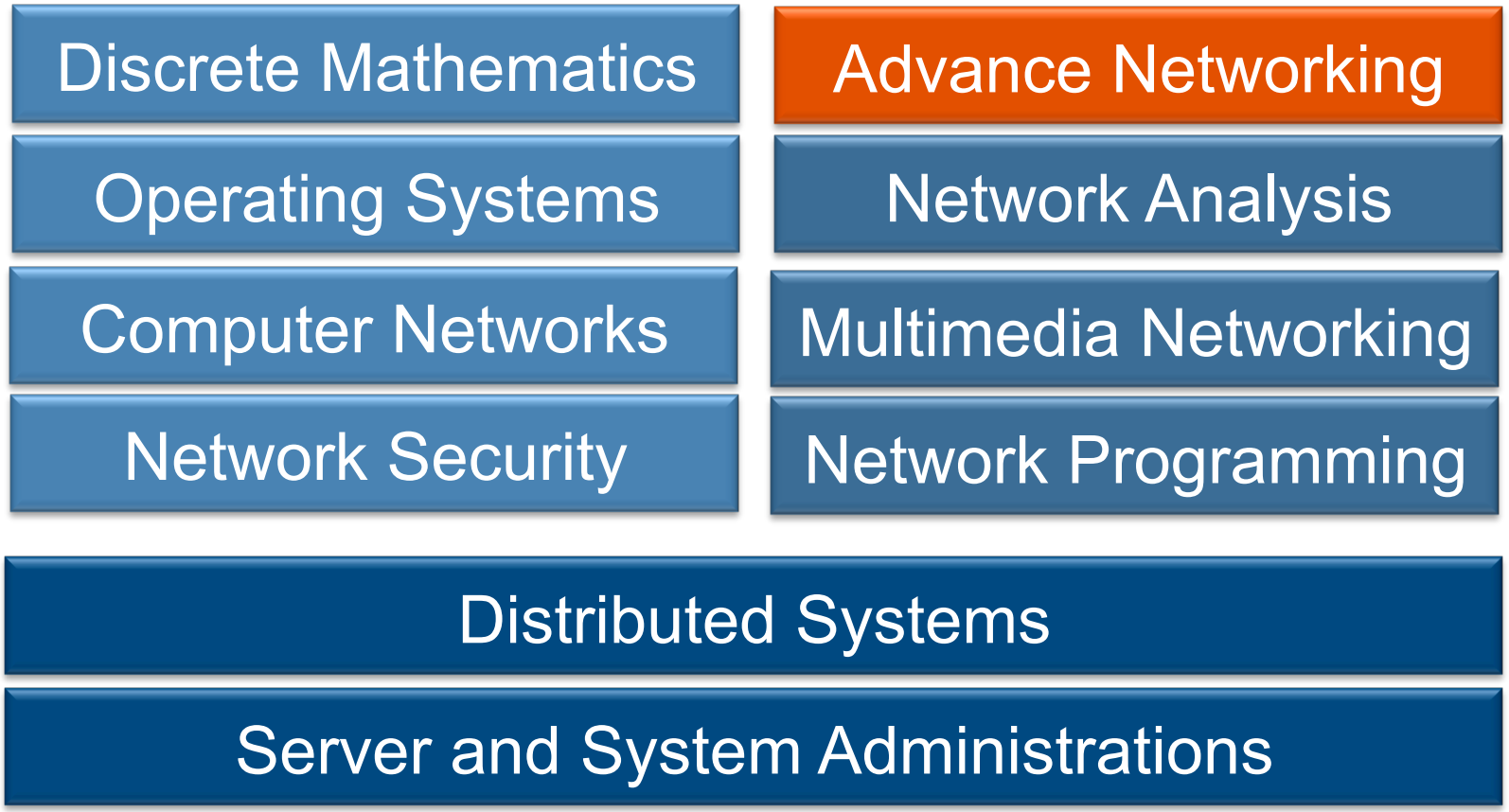
Advance Computer Networks

#1 Review of Computer Networks

Semester Ganjil 2012

PTIIK – Universitas Brawijaya

Networking Courses … at a glance



Today's Lecture

- This Course
  - Goals
  - Outline/Schedule
  - Grading Policy

- Introduction to Multimedia Networking

References

- Kurose & Ross, “Computer Networking : Top down Approach”, 6th Ed., Pearson/Addison, 2012.
- Tanenbaum, “Computer Networks”, 5th Ed., Prentice, 2010.
- Hofmann & Beaumont, “Content Networking”, Morgan, 2005.
- Rosenberg, “A Primer of Multicast Routing”, Springer, 2012.
- O’Driscoll, “Next Generation IPTV Service and Technologies”, Wiley, 2008.
- **Feamster, Proactive Techniques for Correct & Predictable Internet Routing,** Massachusetts Institute of Technology

Goals

- Who can take this class?
  - Almost anyone
- Learn the complexity of networks and protocols, content delivery network, advance network issues
- Prerequisites:
  - Minimum: Have taken “Computer Networks” class

Outline of this Course

- #1 Reviews of Computer Networks
- #2 Routing Algorithms
- #3 Routing in the Internet
- #4 Routing in the Internet (cont'd)
- #5 Broadcast & Multicast Routing
- #6 Overlay Networks
- #7 P2P
- #8 CDN: Introduction
- #9 CDN: Transport
- #10 CDN: Web Caching
- #11 Network Management
- #12 Network Performance Issues
- #13 DTN
- #14 IP NGN

Grading Policy

- Classes
  - Two (2) Credits
- Exercises (assistant required)
  - One (1) Credit
- Evaluation
  - Homework 15%
  - Paper Reading and Presentation 30%
  - MidTerm Test 20%
  - Final Exam 35%

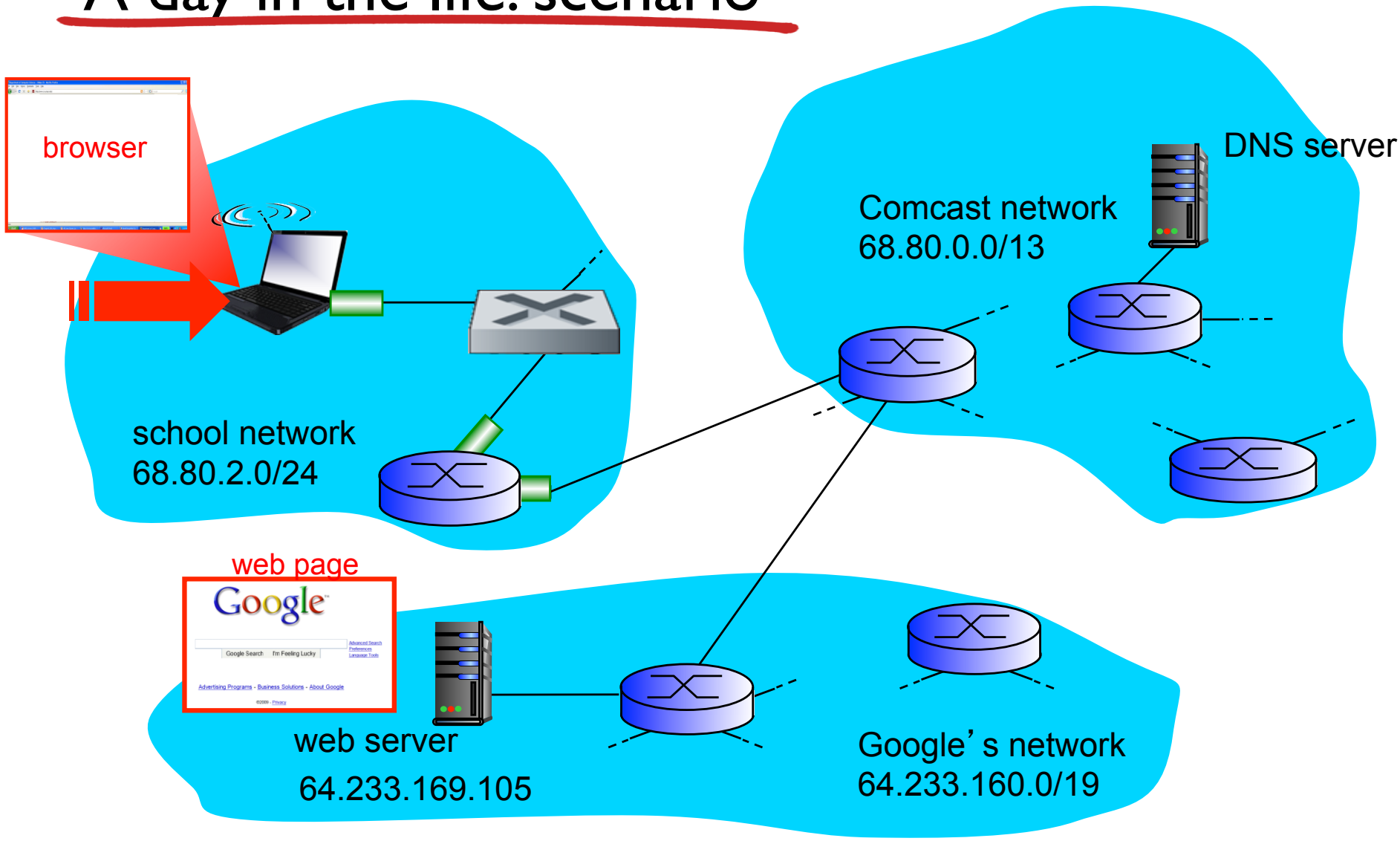
#1 - Reviews of Computer Networks

a day in the life of a web request

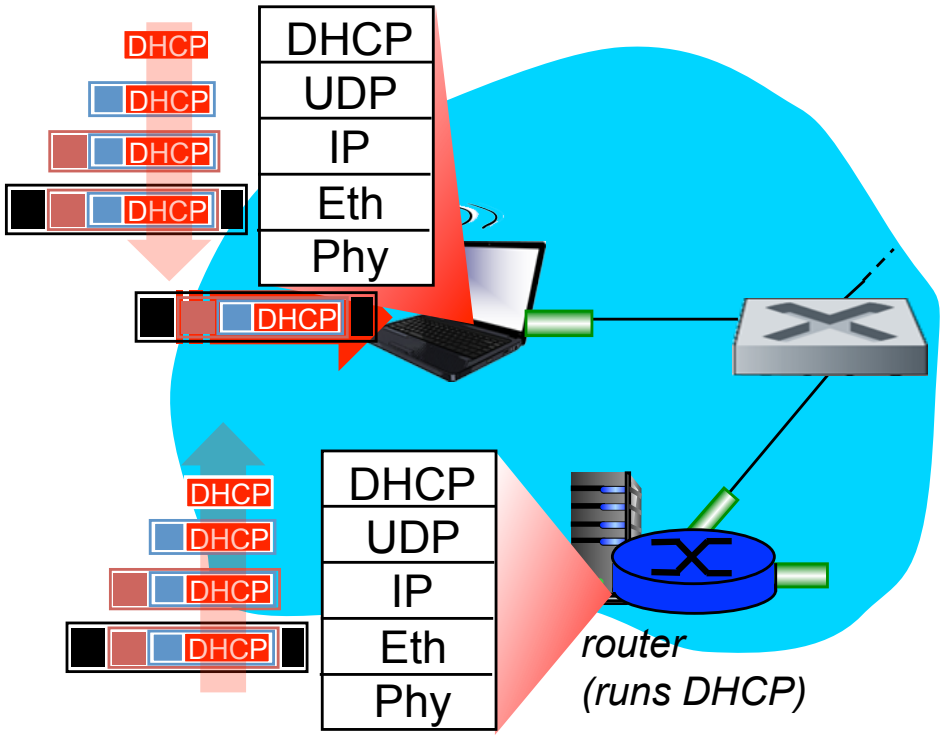
Synthesis: a day in the life of a web request

- 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: scenario



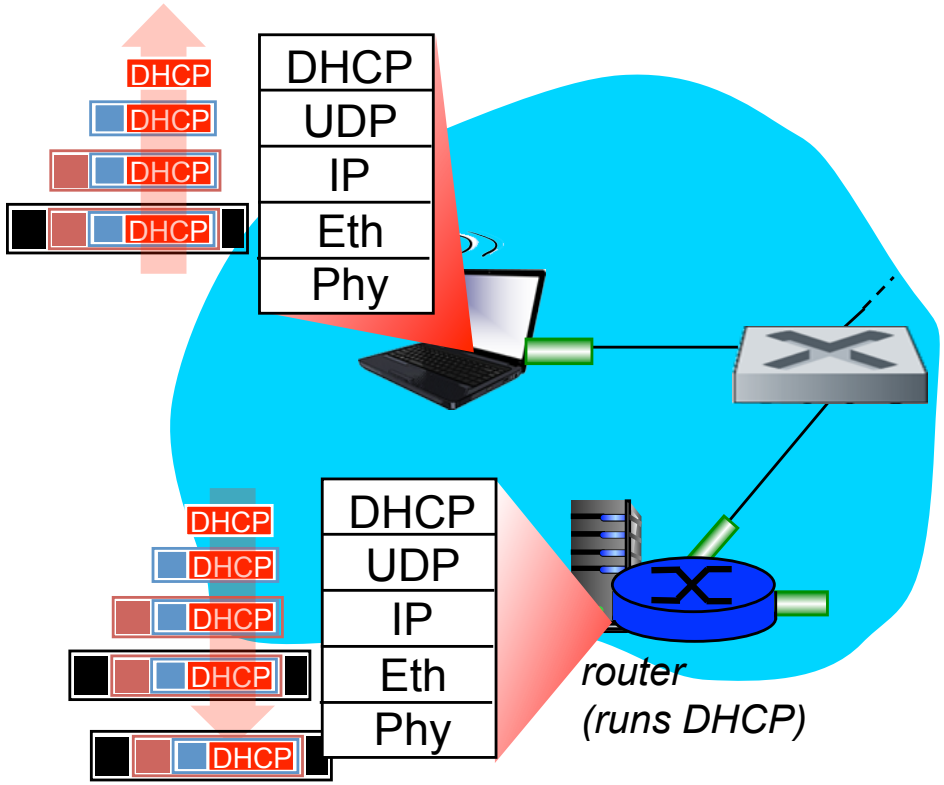
A day in the life… connecting to the Internet



- 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: FFFFFFFFFFFF) on LAN, received at router running DHCP server
- Ethernet demuxed to IP demuxed, UDP demuxed to DHCP

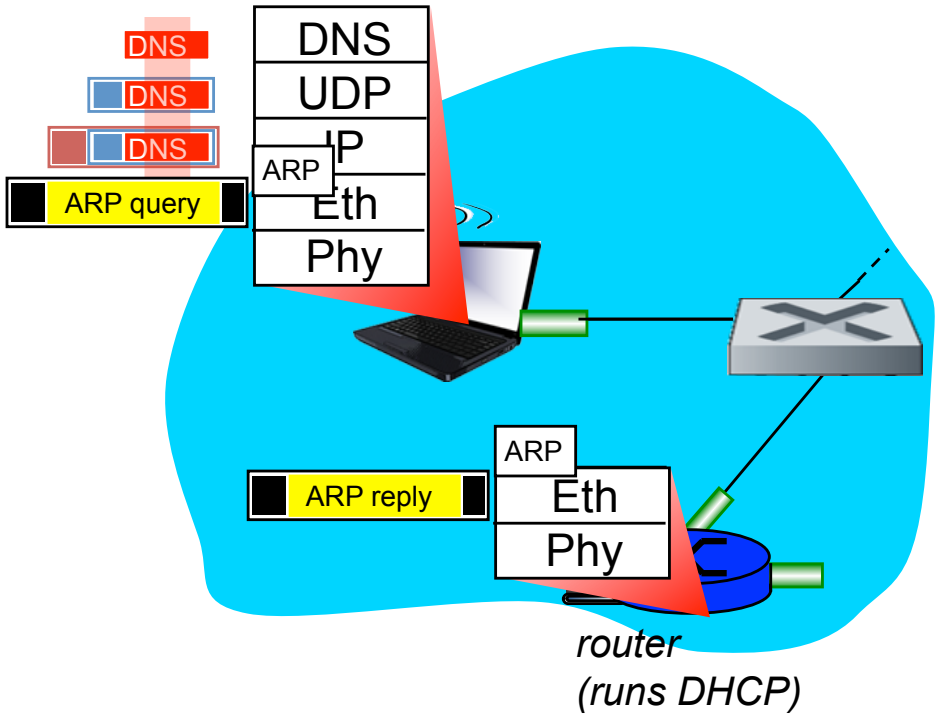
A day in the life… connecting to the Internet



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

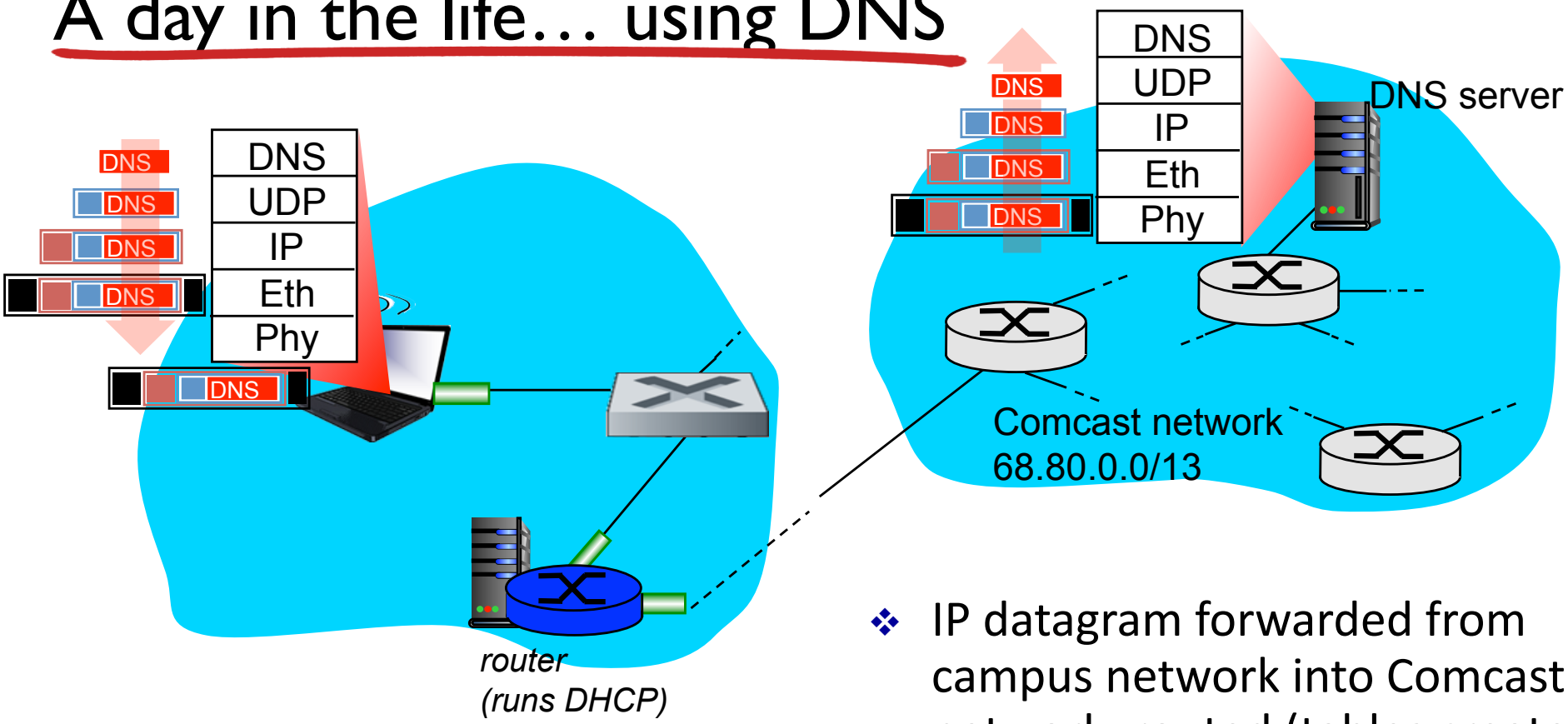
Client now has IP address, knows name & addr of DNS server, IP address of its first-hop router

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



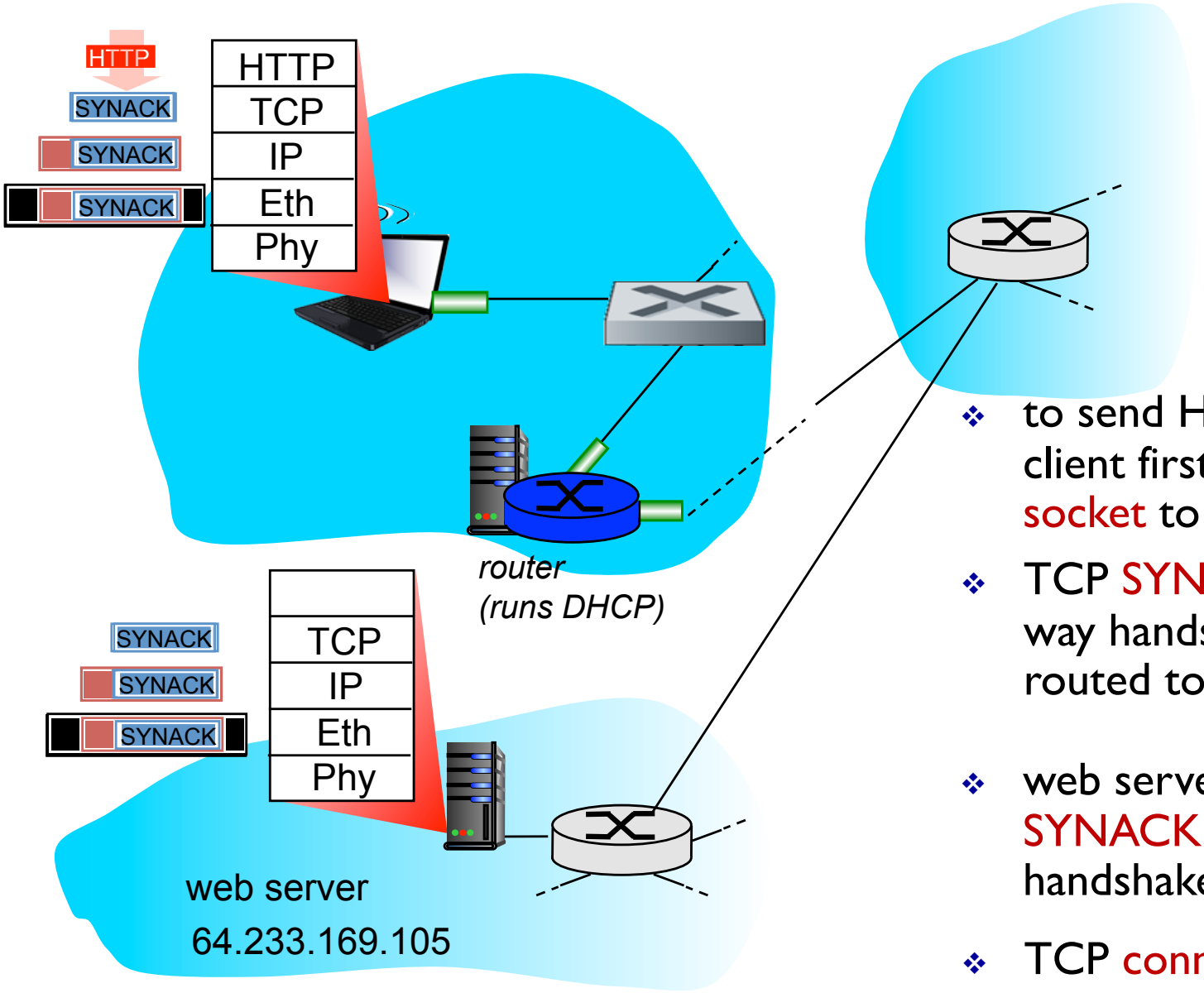
- before sending *HTTP* request, need IP address of www.google.com: *DNS*
- DNS query created, encapsulated in UDP, encapsulated in IP, encapsulated in Eth. To send frame to router, need MAC address of router interface: ARP
  - ARP query broadcast, received by router, which replies with ARP reply giving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



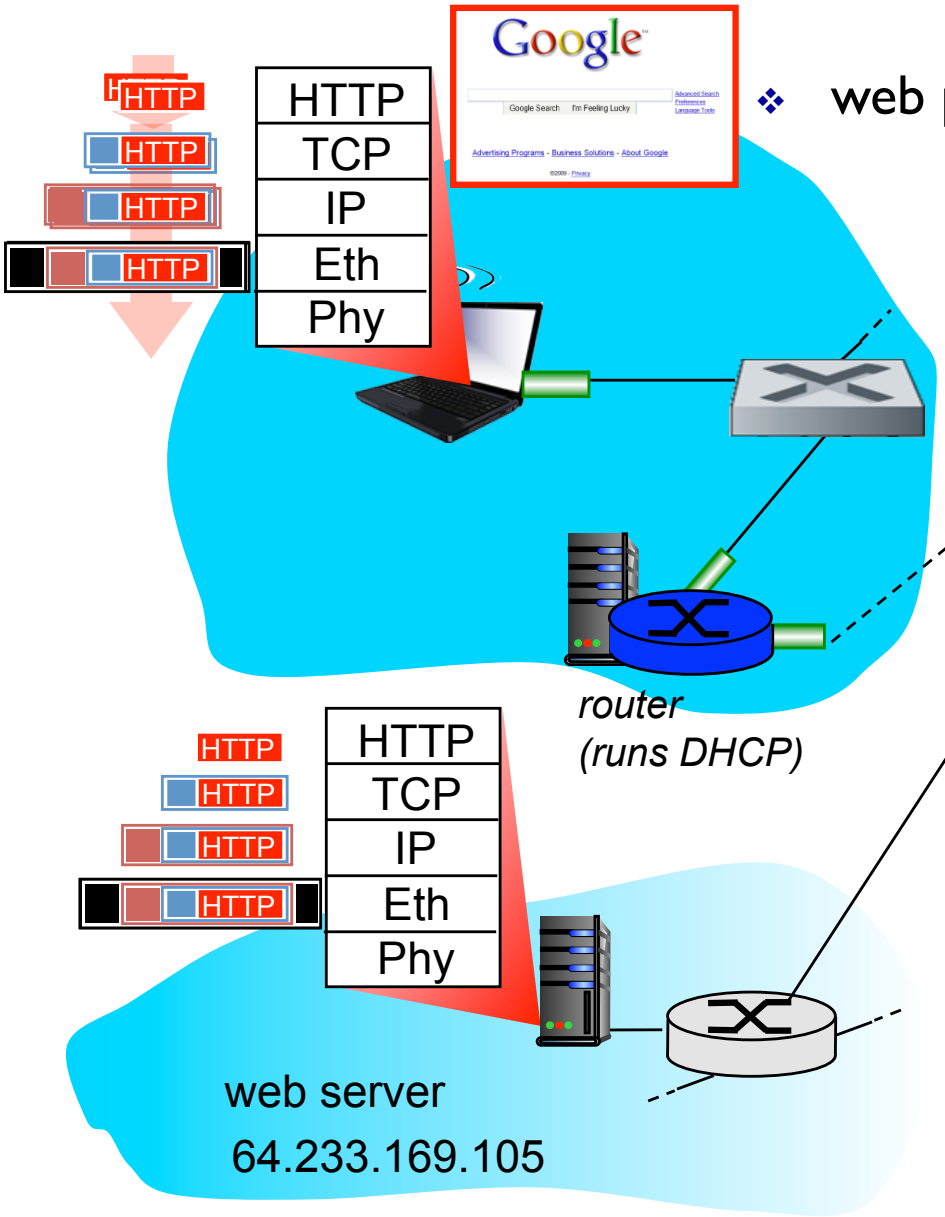
- IP datagram containing DNS query forwarded via LAN switch from client to 1st hop router
- IP datagram forwarded from campus network into Comcast network, routed (tables created by RIP, OSPF, IS-IS and/or BGP routing protocols) to DNS server
- demux'ed to DNS server
- DNS server replies to client with IP address of www.google.com

A day in the life…TCP connection carrying HTTP



- to send HTTP request, client first opens TCP socket to web server
- 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)
- 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