Application Layer

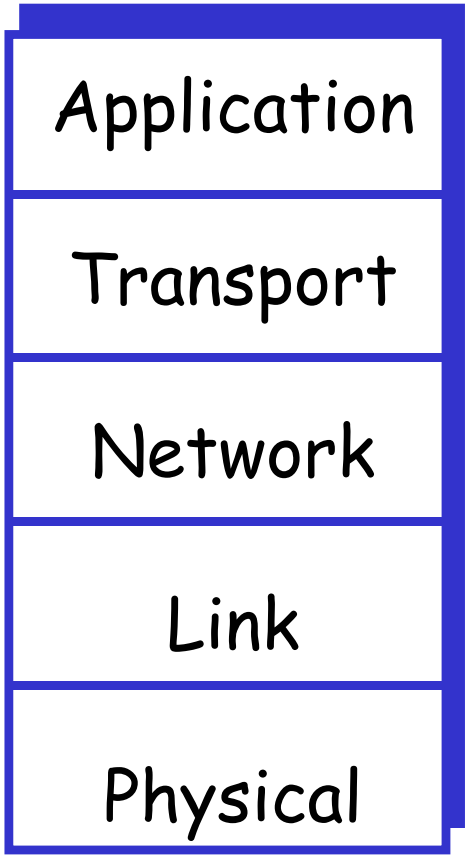
ELEC1200

- Application-layer concepts
- http, DNS

** The slides are adapted from ppt slides (in substantially unaltered form) available from "Computer Networking: A Top-Down Approach," 4th edition, by Jim Kurose and Keith Ross, Addison-Wesley, July 2007. Part of the materials are also adapted from ELEC315 and MIT 6.02 course notes.*

Internet protocol stack

- application: supporting network applications
  - HTTP, SMTP, FTP, DNS
- transport: process-process data transfer
  - TCP, UDP
- network: routing of datagrams from source to destination
  - IP, routing protocols
- link: data transfer between neighboring network elements
  - 802.11, Ethernet
- physical: bits "on the wire"



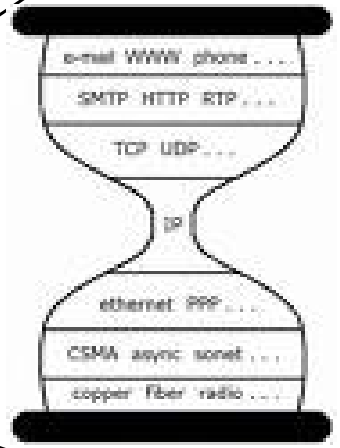
The Internet "Hourglass"

Applications

Web FTP Mail News Video Audio ping napster

TCP SCTP UDP ICMP

IP



Ethernet 802.11 Power lines ATM Optical Satellite Bluetooth

Link technologies

- Many applications, transports, and link protocols
- All use IP at the network layer: universal network layer

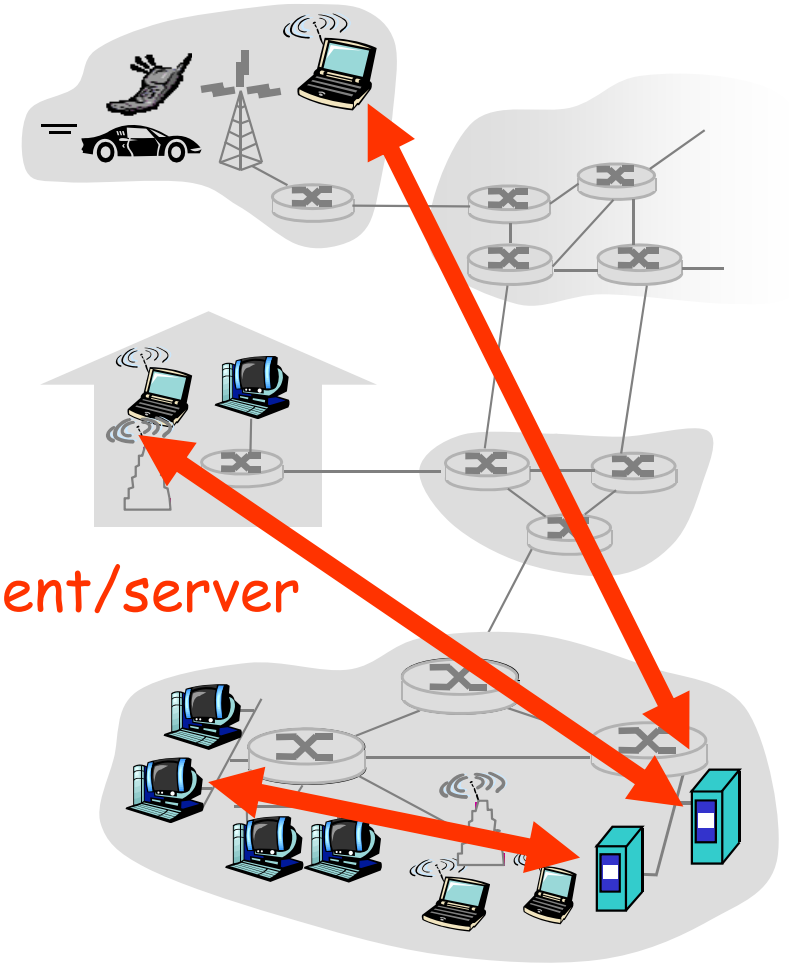
Some network apps

- e-mail
- web
- instant messaging
- remote login
- P2P file sharing
- multi-user network games
- streaming stored video clips
- voice over IP
- real-time video conferencing
- grid computing

Application architectures

- Client-server
- Peer-to-peer (P2P)
- Hybrid of client-server and P2P

Client-server architecture



server:

- always-on host
- permanent IP address
- server farms for scaling

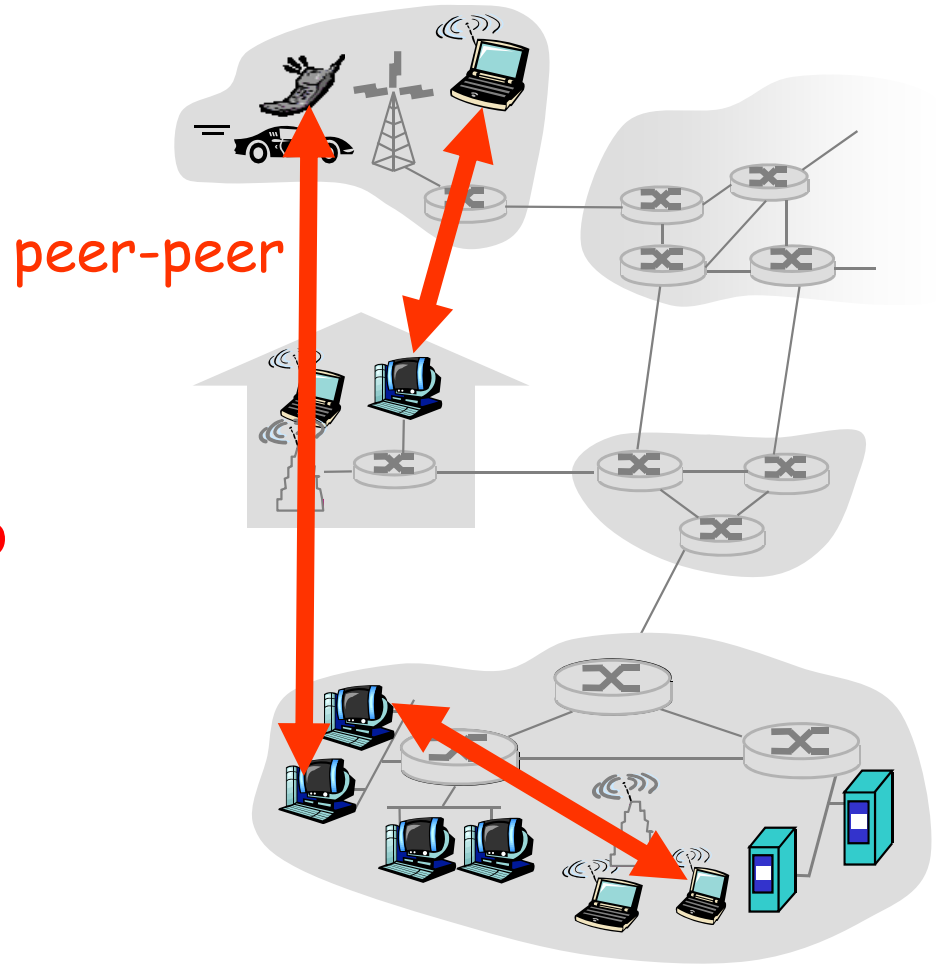
clients:

- communicate with server
- may be intermittently connected
- may have dynamic IP addresses
- do not communicate directly with each other

Pure P2P architecture

- *no* always-on server
- arbitrary end systems directly communicate
- peers are intermittently connected and change IP addresses

Highly scalable but difficult to manage



Hybrid of client-server and P2P

Skype

- voice-over-IP P2P application
- centralized server: finding address of remote party:
- client-client connection: direct (not through server)

Instant messaging

- chatting between two users is P2P
- centralized service: client presence detection/location
  - user registers its IP address with central server when it comes online
  - user contacts central server to find IP addresses of buddies

Server examples

- SMTP simple mail transfer protocol. Electronic mail delivery protocol
- Telnet Terminal emulation. Enables login and interactive session on a remote system
- FTP File transfer protocol. Reliable file transfer. Allows ASCII and binary files to be transferred interactively
- DNS Domain Name system. Name service used by the Internet. Can use both TCP and UDP transport-layer protocols
- HTTP

Web and HTTP

First some jargon

- Web page consists of objects
- Object can be HTML file, JPEG image, Java applet, audio file,…
- Web page consists of base HTML-file which includes several referenced objects
- Each object is addressable by a URL
- Example URL:

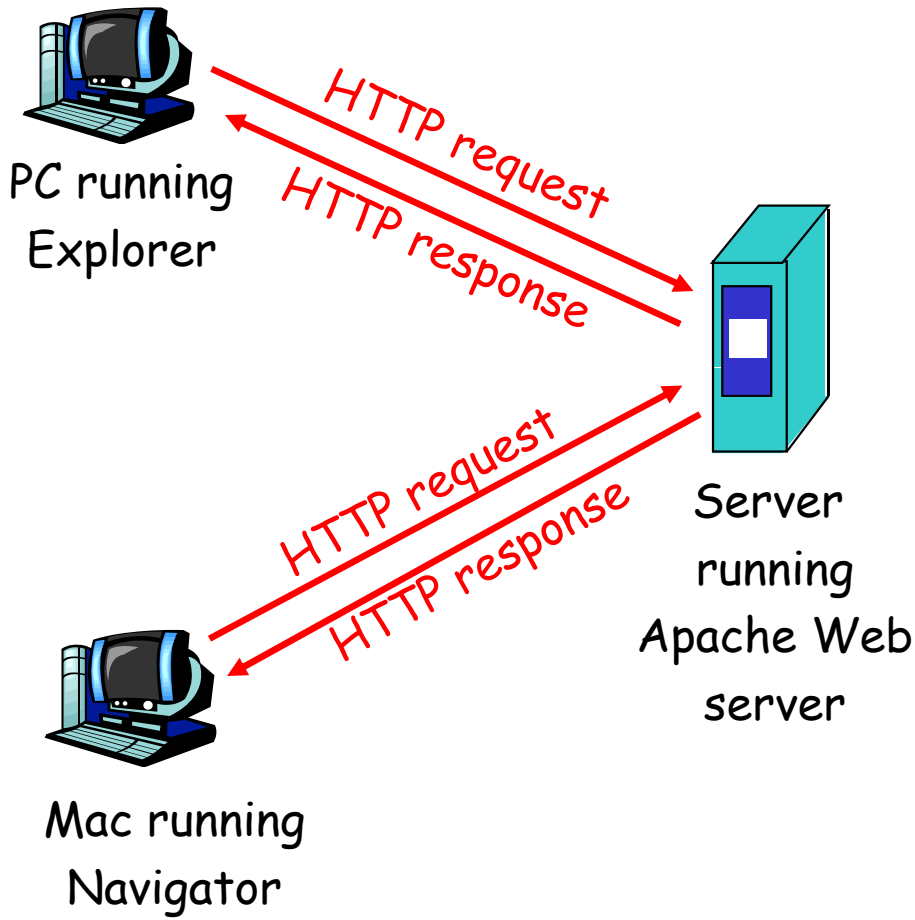
`www.someschool.edu/someDept/pic.gif`

host name | path name

HTTP overview

HTTP: hypertext transfer protocol

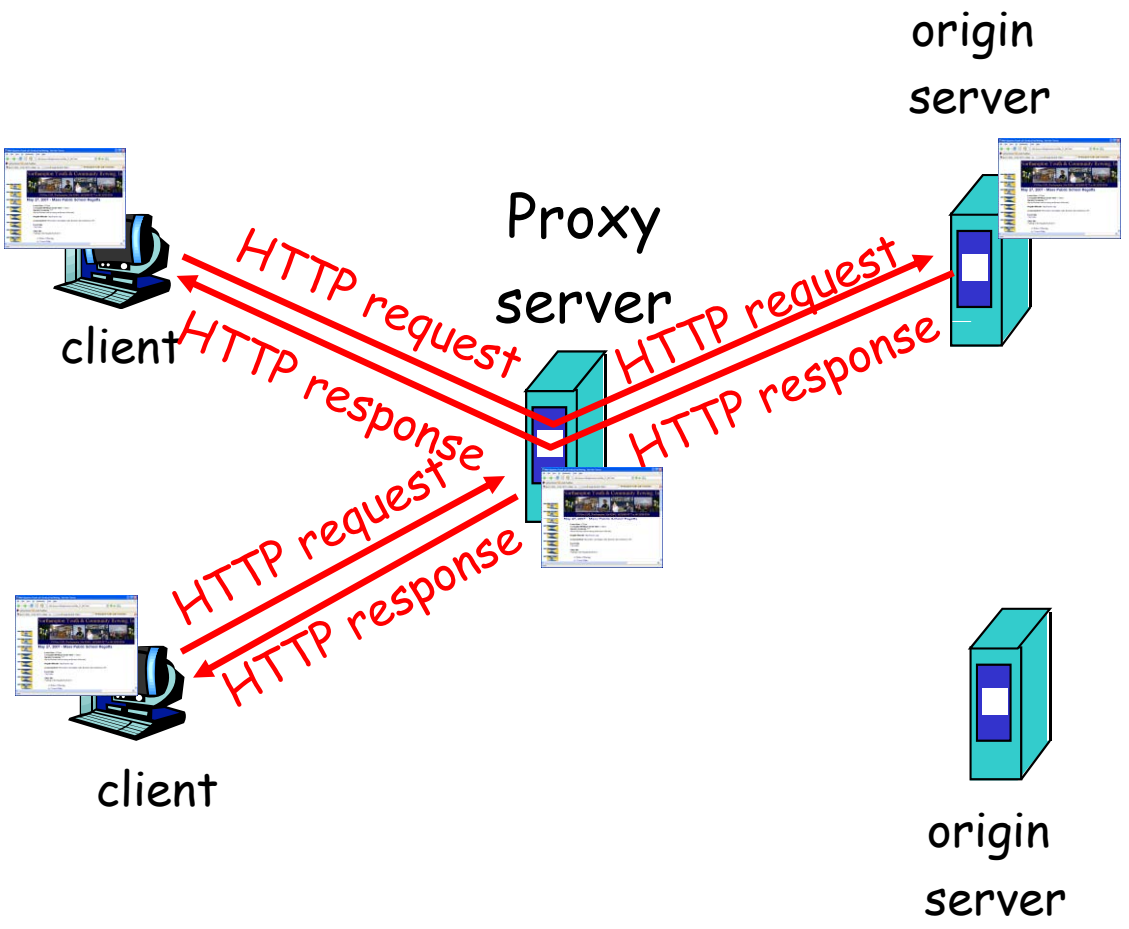
- Web's application layer protocol over TCP
- Server port: 80
- client/server model
  - *client:* browser that requests, receives, "displays" Web objects
  - *server:* Web server sends objects in response to requests



Web caches (proxy server)

Goal: satisfy client request without involving origin server

- user sets browser: Web accesses via cache
- browser sends all HTTP requests to cache
  - object in cache: cache returns object
  - else cache requests object from origin server, then returns object to client



Trying out HTTP (client side) for yourself

1. Telnet to your favorite Web server:

```
telnet www.ee.ust.hk 80
```

Opens TCP connection to port 80 (default HTTP server port) at www.ee.ust.hk. Anything typed in sent to port 80 at www.ee.ust.hk

2. Type in a GET HTTP request:

```
GET /~eetsang/ HTTP/1.1
Host: www.ee.ust.hk
```

By typing this in (hit carriage return twice), you send this minimal (but complete) GET request to HTTP server

3. Look at response message sent by HTTP server

DNS – the Internet's Directory Service

People: many identifiers:

- ID card #, name, passport #

Internet hosts, routers:

- IP address (32 bit) - used for addressing datagrams, e.g. 209.131.36.158
- "hostname", e.g., www.yahoo.com - used by humans

Q: map between IP addresses and name ?

Domain Name System:

- *an Internet-wide service* that provides mappings between IP addresses and hostnames
- *distributed database* that resides in multiple machines in the Internet, implemented in *hierarchy of many name servers*
- *application-layer protocol* host, routers, name servers to communicate to *resolve* names (address/name translation)
  - note: an essential Internet function, implemented as application-layer protocol
- *usually involves short messages*, and so primarily uses services provided by UDP

DNS name servers

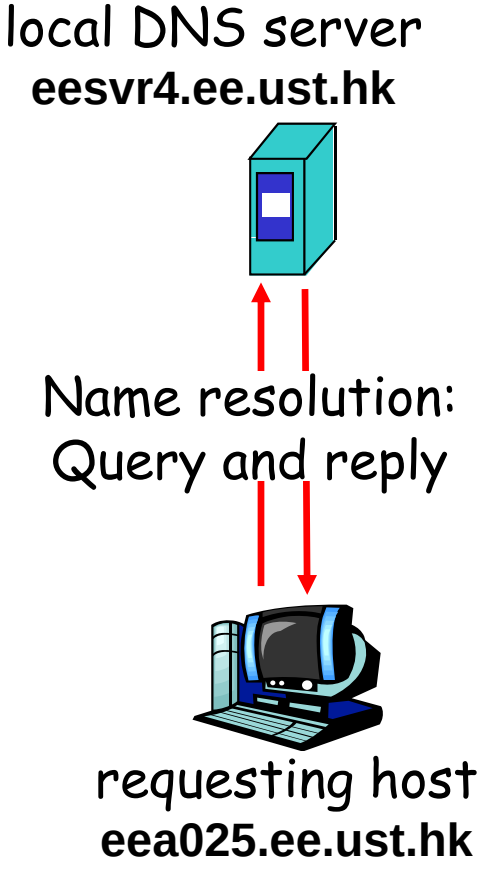
- Why not centralize DNS?
  - a server process on a big, well connected supercomputer?

- Centralized systems do not scale!
  - poor reliability: single point of failure
  - poor performance: remote access for most users
  - difficult to manage – all traffic goes to one location

- A centralized system is not politically feasible in a global network

DNS – a distributed service

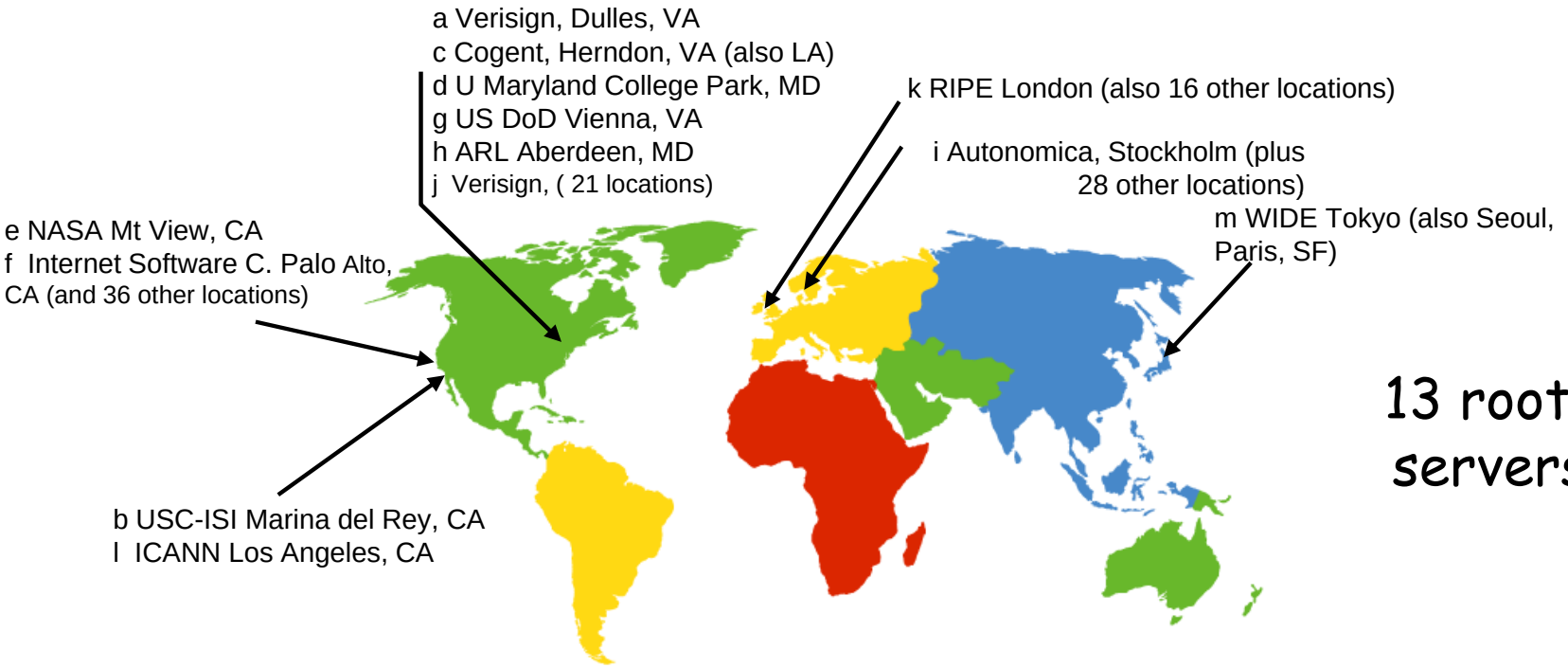
- No server has all name-to-IP address mappings
- Local name servers
  - each ISP, company has local (default) name server
  - host DNS query first goes to local name server
- Authoritative name server
  - every host is registered with at least two authoritative servers which store that host's IP address, name
  - Can perform name/address translation for that host



What if the name is not a local host?
(e.g. www.yahoo.com)

DNS: Root name servers

- contacted by local name server that cannot resolve name
- root name server:
  - contacts authoritative name server if name mapping not known
  - gets mapping
  - returns mapping to local name server

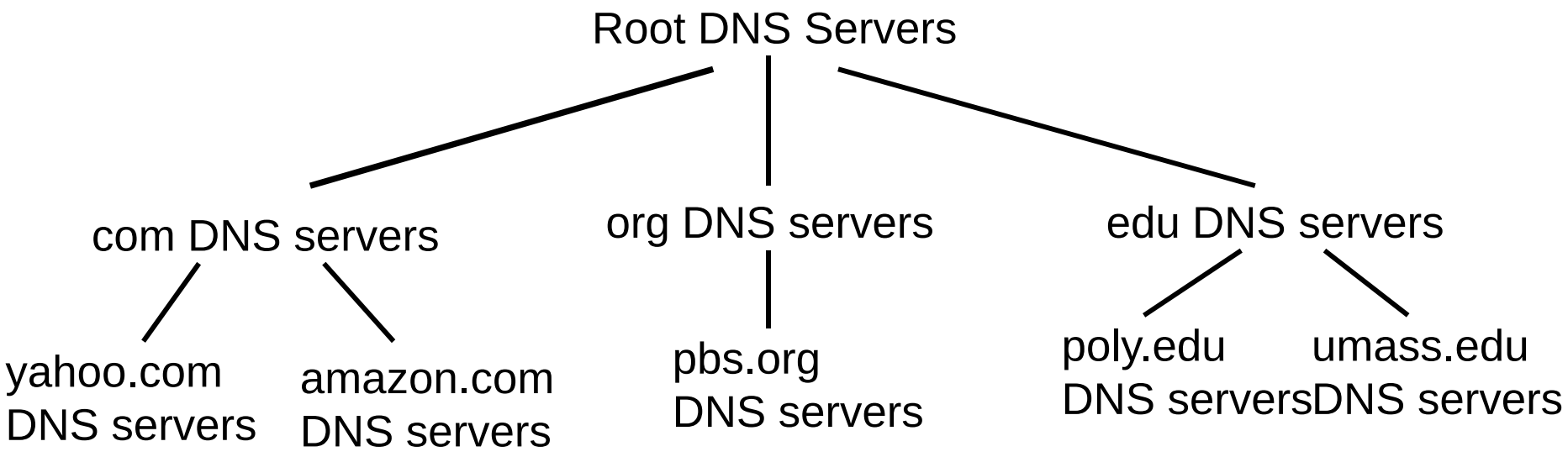


13 root name servers worldwide

TLD and Authoritative Servers

- Top-level domain (TLD) servers:
  - responsible for com, org, net, edu, etc, and all top-level country domains uk, fr, ca, jp.
  - Network Solutions maintains servers for com TLD
  - Educause for edu TLD
- Authoritative DNS servers:
  - organization's DNS servers, providing authoritative hostname to IP mappings for organization's servers (e.g., Web, mail).
  - can be maintained by organization or service provider

Distributed, Hierarchical Database



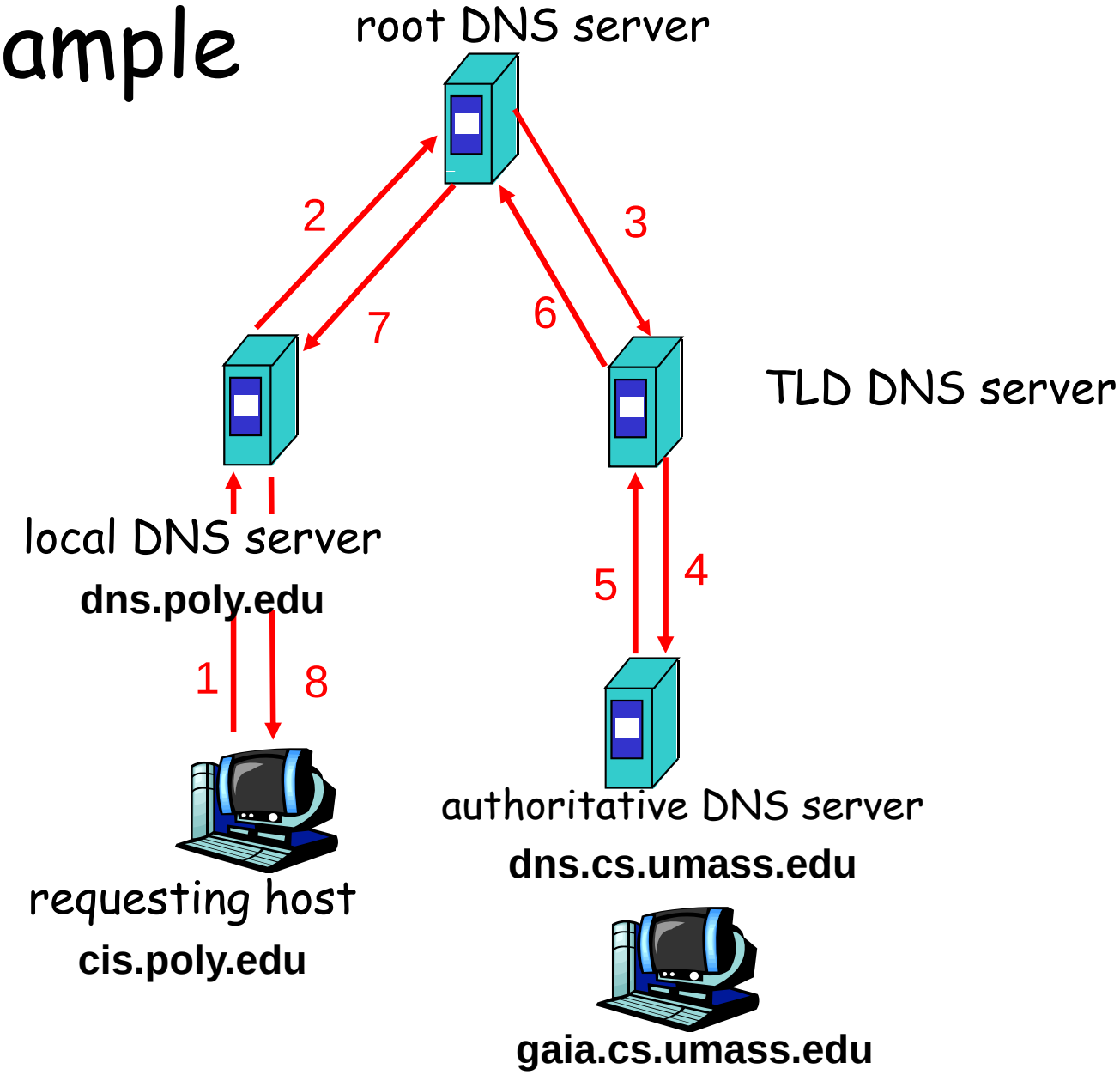
Client wants IP for www.yahoo.com; 1st approx:

- client queries a root server to find com DNS server
- client queries com DNS server (a TLD name server) to get amazon.com DNS server (authoritative name server)
- client queries yahoo.com DNS server to get IP address for www.yahoo.com

DNS name resolution example

recursive query:

- puts burden of name resolution on contacted name server
- heavy load?



DNS: caching and updating records

- once (any) name server learns mapping, it *caches* mapping
  - cache entries timeout (disappear) after some time
  - TLD servers typically cached in local name servers
    - Thus root name servers not often visited
- update/notify mechanisms under design by IETF
  - RFC 2136
  - http://www.ietf.org/html.charters/dnsind-charter.html

Summary

- Many applications
- Use the communication system with different protocols
- Applications include http and DNS