CS 43: Computer Networks

06: Distributed Systems and DNS

September 21, 2026

SWARTHMORE COLLEGE

Slides adapted from Kurose & Ross, Vasanta Chaganti, Kevin Webb, Laurent Vanbever

Where we are

Application: the application (DNS)

Transport: end-to-end connections, reliability

Network: routing

Link (data-link): framing, error detection

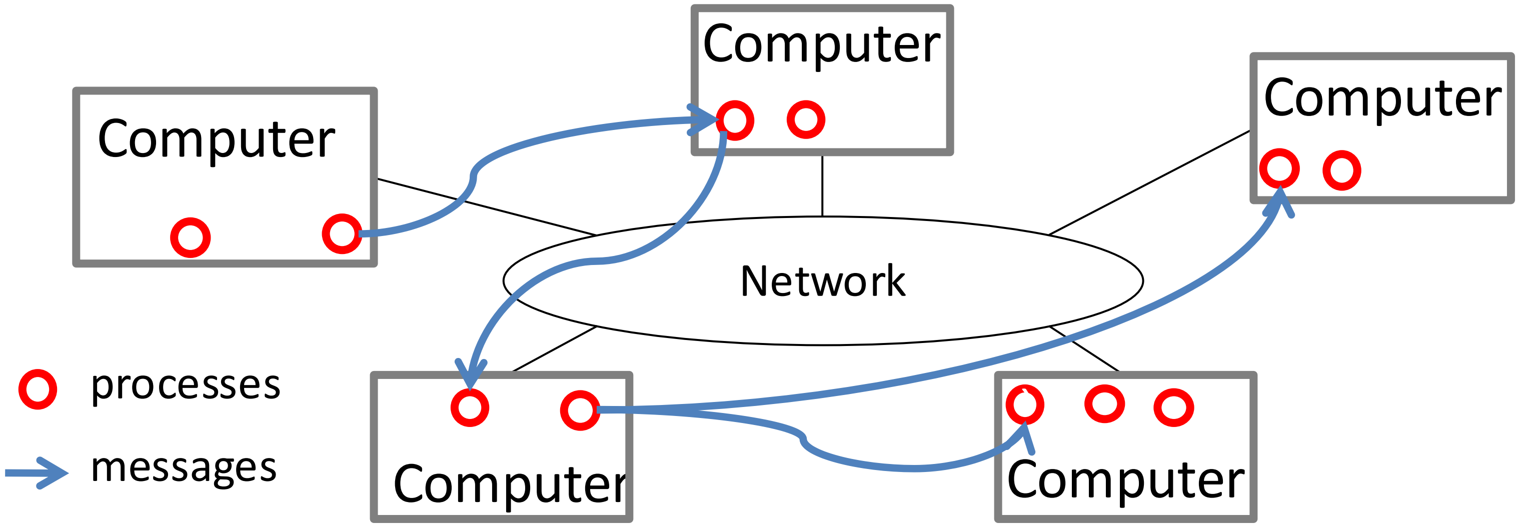
Physical: 1’s and 0’s/bits across a medium
(copper, the air, fiber)

This week's question:

How does DNS achieve consistency, availability, and high performance? Which of these does it prioritize?

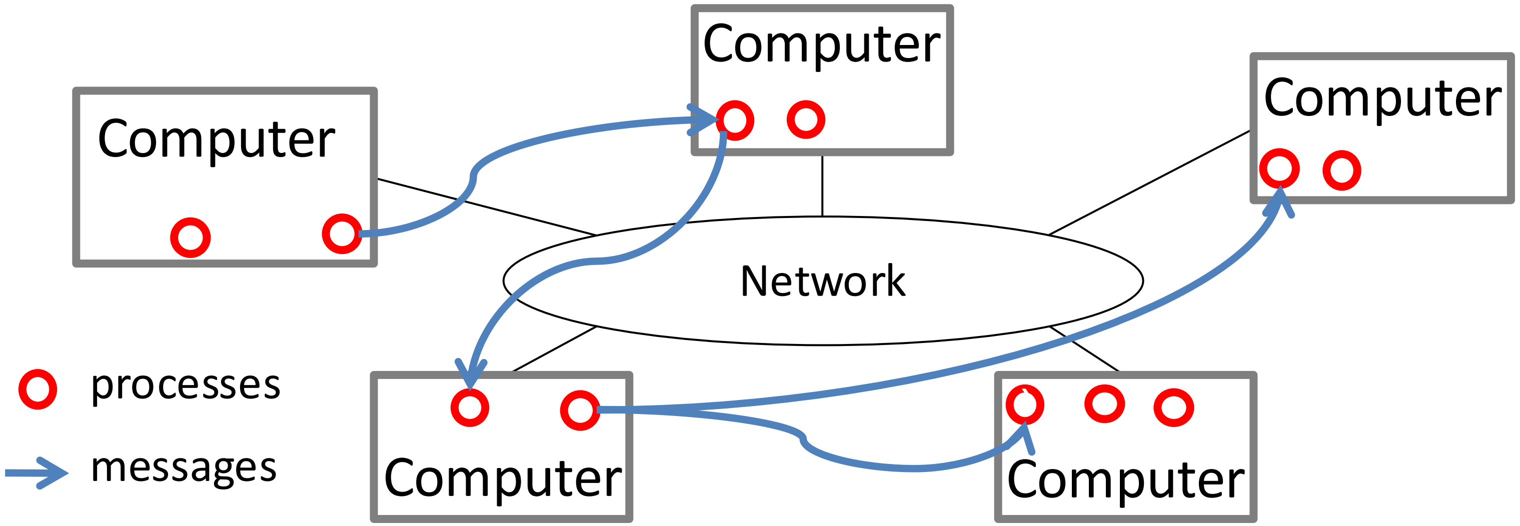
Distributed Network Applications

What is a distributed application?



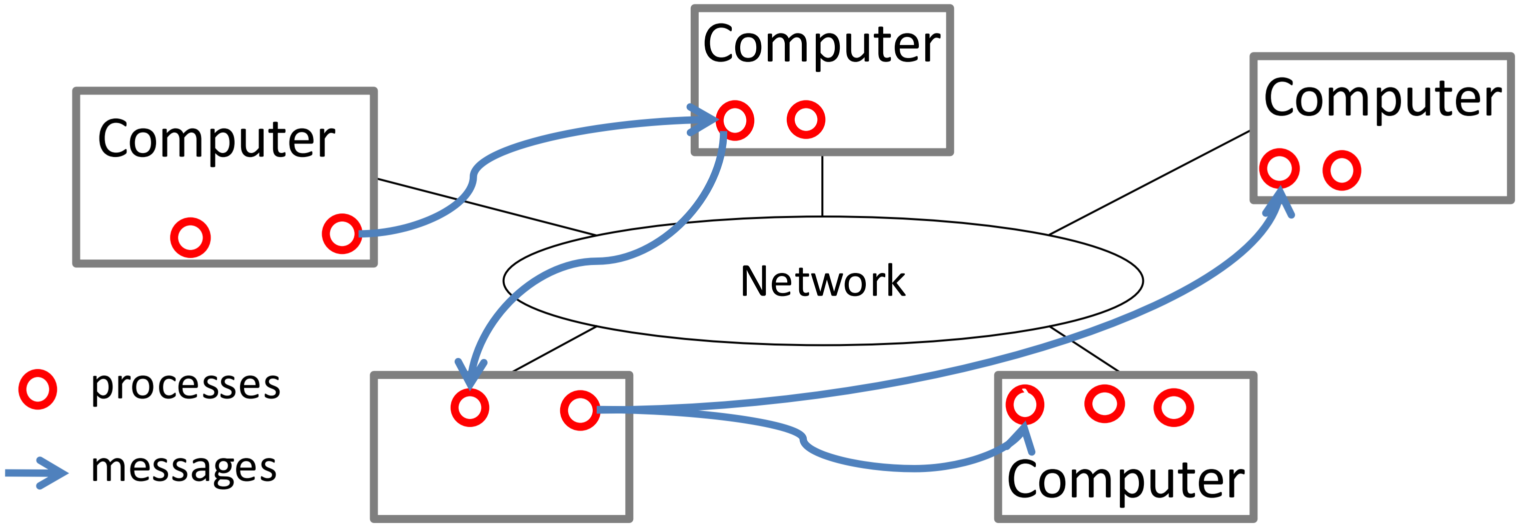
- Cooperating processes in a computer network
- Varying degrees of integration
  - Loose: email, web browsing
  - Medium: chat, Skype, remote execution, remote file systems
  - Tight: process migration, distributed file systems

Distributed Systems: Advantages



- Speed: parallelism, less contention
- Reliability: redundancy, fault tolerance
- Scalability: incremental growth, economy of scale
- Geographic distribution: low latency, reliability

Distributed Systems: Disadvantages



- Fundamental problems of decentralized control
  - State uncertainty: no shared memory or clock
  - Action uncertainty: mutually conflicting decisions
- Distributed algorithms are complex

On a single system…

- You have a number of components
  - CPU
  - Memory
  - Disk
  - Power supply
- If any of these go wrong, you’re (usually) toast.

On multiple systems…

- New classes of failures (**partial failures**).
  - A link might fail
  - One (of many) processes might fail
  - The network might be partitioned

On multiple systems…

- New classes of failures (**partial failures**).
  - A link might fail
  - One (of many) processes might fail
  - The network might be partitioned

Introduces major complexity!

If a process sends a message, can it tell the difference between a slow link and a delivery failure?

A. Yes

B. No

What should we do to handle a partial failure? Under what circumstances, or what types of distributed applications?

A. If one process fails or becomes unreachable, switch to a spare.
B. Pause or shut down the application until all connectivity and processes are available.
C. Allow the application to keep running, even if not all processes can communicate.
D. Handle the failure in some other way.

Desirable Properties

- Consistency
  - Nodes agree on the distributed system's state
- Availability
  - The system is able and willing to process requests
- Partition tolerance
  - The system is robust to network (dis)connectivity

The CAP Theorem

- **C**onsistency
  – Nodes agree on the distributed system's state
- **A**vailability
  – The system is able and willing to process requests
- **P**artition tolerance
  – The system is robust to network (dis)connectivity
- Choose Two
- "CAP prohibits only a tiny part of the design space: perfect availability and consistency in the presence of partitions, which are rare."*

* Brewer, Eric. "CAP twelve years later: How the" rules" have changed." Computer 45.2 (2012): 23-29.

Consistency: Event Ordering

- It’s very useful if all nodes can agree on the order of events in a distributed system

- For example: Two users trying to update a shared file across two replicas

If two events occur (digitally or in the “real world”), can we always tell which happened first?

A. Yes

B. No

Consistency: Event Ordering

- It’s very useful if all nodes can agree on the order of events in a distributed system
- For example: Two users trying to update a shared file across two replicas
- “Time, Clocks, and the Ordering of Events in a Distributed System” by Leslie Lamport (1978)
  - Establishes causal orderings
  - Cited > 8000 times

Summary

- Distributed systems are hard to build!
  - Partial failures
  - Ordering of events

- Take CS 87 for more details!

DNS: Domain Name System

People: many identifiers:

– name, swat ID, SSN, passport #

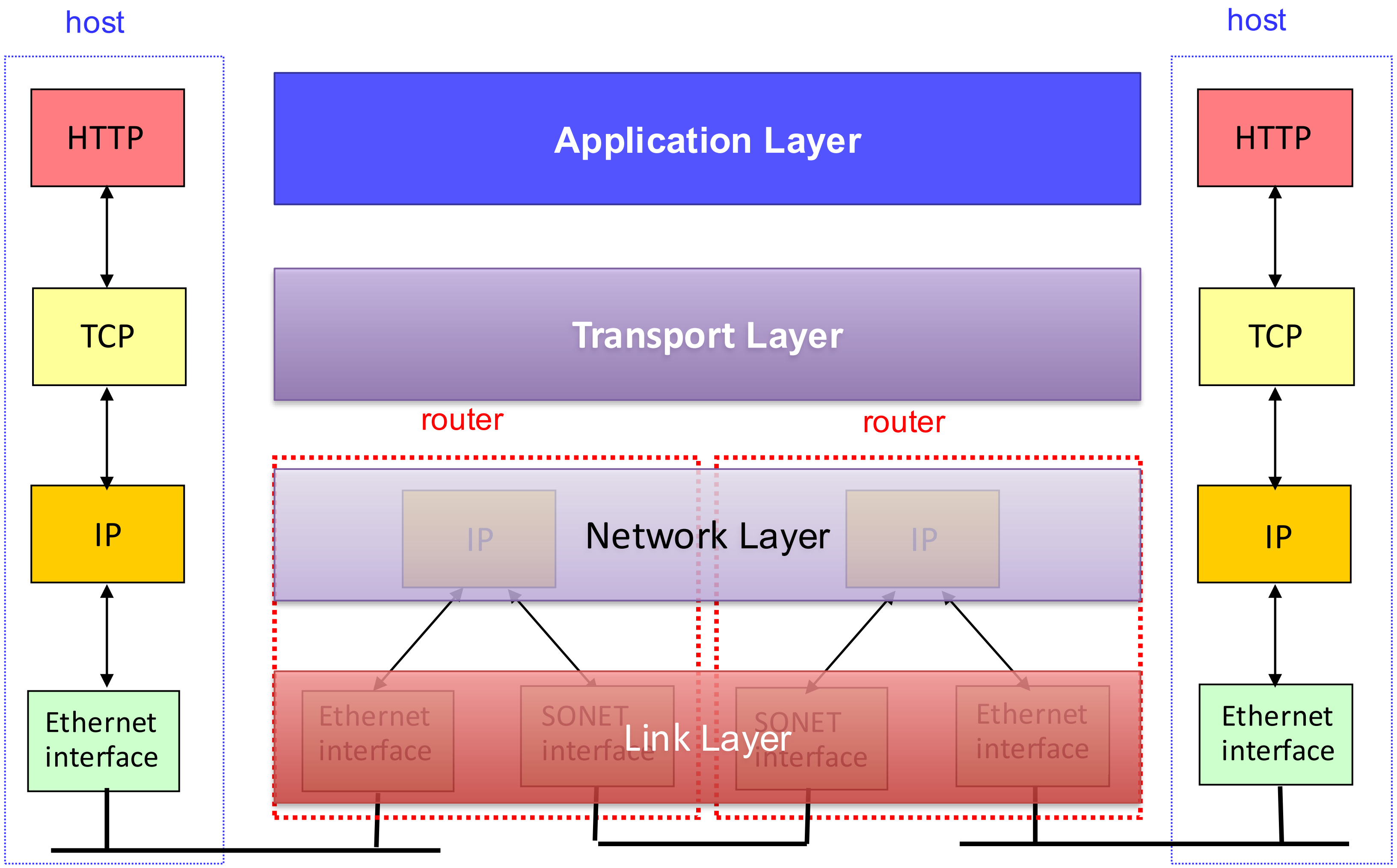
Internet hosts (endpoints), routers (devices inside a n/w):

– “name”, e.g., www.google.com - used by humans

– IP address (32 bit) - used for addressing packets

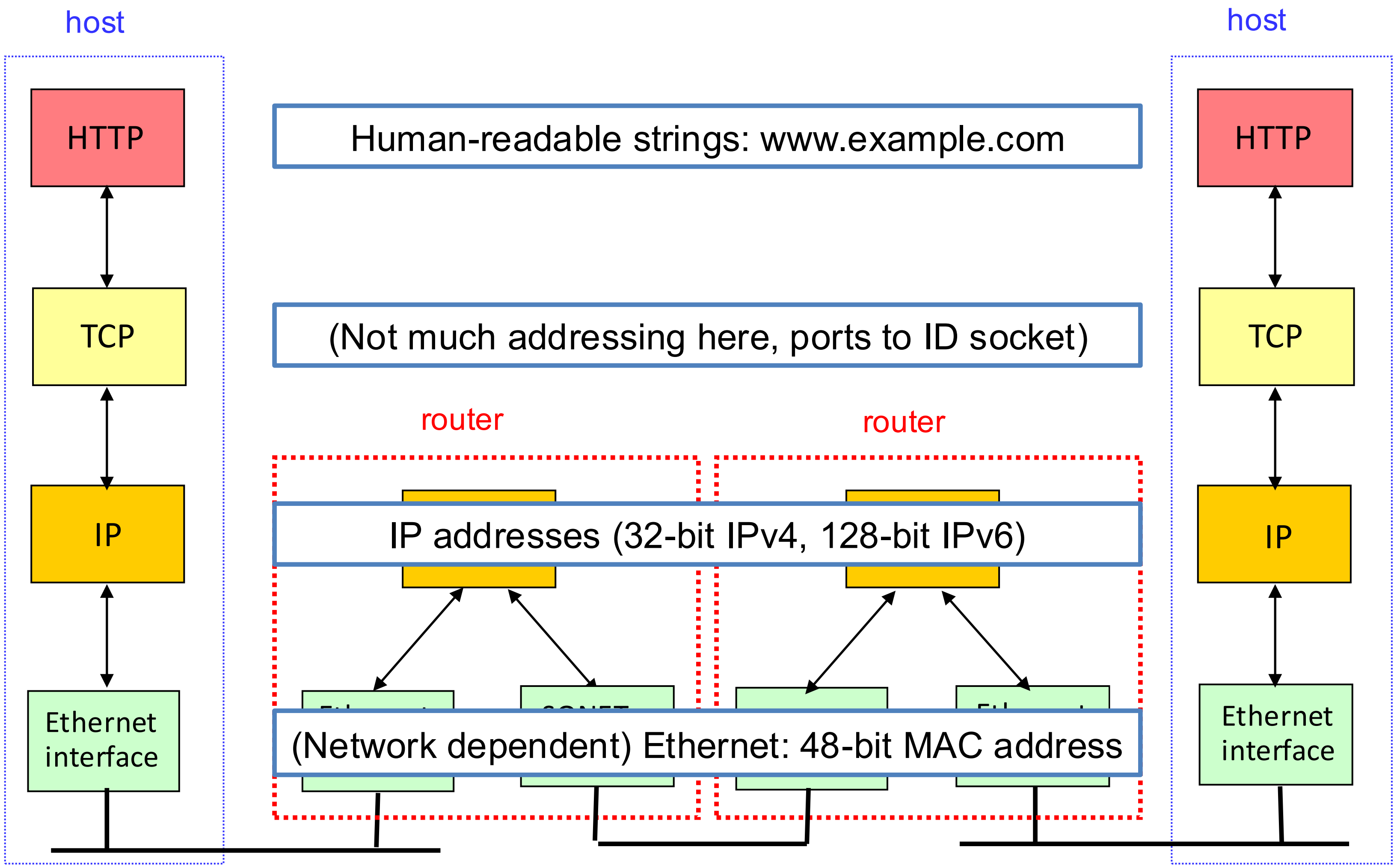
How do we map between IP address and name, and vice versa ?

Where



host
HTTP
TCP
IP
Ethernet interface
Application Layer
Transport Layer
router
router
Network Layer
IP
IP
Ethernet interface
SONET interface
Link Layer
SONET interface
Ethernet interface
host
HTTP
TCP
IP
Ethernet interface

Recall: TCP/IP Protocol Stack



Why do we need to map names to IP addresses? Why not route on names at the network layer?

A. Domain names are hierarchical, so we can route on domain names too.
B. Domain names are variable length, vs IP are fixed length, some changes will be required to switch.
C. With domain names we wouldn't know where to route to geographically.
D. Some other reason.

Identifiers

- Host name (e.g., www.swarthmore.edu)
  - Used by humans to specify host of interest
  - Unique, selected by host administrator
  - Hierarchical, variable-length string of alphanumeric characters
- IP address (e.g., 130.58.68.164)
  - Used by routers to forward packets
  - Unique, topologically meaningful locator
  - Hierarchical namespace of 32 bits

Mapping Between Identifiers

- Domain Name System (DNS)
  - Given a host name, provide the IP address
  - Given an IP address, provide the host name

Is a hostname to IP address mapping always permanent?

A. Yes
B. No
C. I don't know

X

What's the biggest challenge for DNS?

A. It's old.

B. The fact that the Internet is global.

C. The fact that DNS is now critical infrastructure.

D. The sheer number of name lookups happening at any given time.

E. How and when the name -> IP address mapping should change.

In the old days…

- Pre-1982, everyone downloads a "hosts.txt" file from SRI
- Pre-1998, Jon Postel, researcher at USC, runs the Internet Assigned Numbers Authority (IANA)
  - RFCs 882 & 883 in 1983
  - RFCs 1034 & 1035 in 1987

Emailed 8/12 root DNS servers, asked change to his authority. They did.

http://www.wired.com/wiredenterprise/2012/10/joe-postel/

Who should control DNS?

A. US government

B. UN / International government

C. Private corporation

D. Someone else

Since 1998…

- Control of Internet Assigned Numbers Authority (IANA) transferred to Internet Corporation for Assigned Names and Numbers (ICANN)
  – ICANN is a private non-profit (formerly) blessed by US DOC
  – Global advisory committee for dealing with international issues
  – 2000’s: Many efforts for UN control, US resisted
  – 2016: ICANN no longer partnered with DOC

DNS Root Server and Politics

ars TECHNICA

BIZ & IT TECH SCIENCE POLICY CARS GAMING & CULTURE

REVOKE .RU DOMAIN, UKRAINE SAYS —

Ukraine asks ICANN to revoke Russian domains and shut down DNS root servers

Expert: Cutting DNS links would harm Russian people but have little impact on gov't.

JON BRODKIN - 3/2/2022, 2:33 PM

ars TECHNICA

BIZ & IT TECH SCIENCE POLICY CARS GAMING & CULTURE

THE INTERNET MUST KEEP WORKING —

ICANN won't revoke Russian Internet domains, says effect would be "devastating"

ICANN's mission: Make sure the Internet works "regardless of the provocations."

JON BRODKIN - 3/4/2022, 1:13 PM

Uses of DNS

Hostname to IP address translation

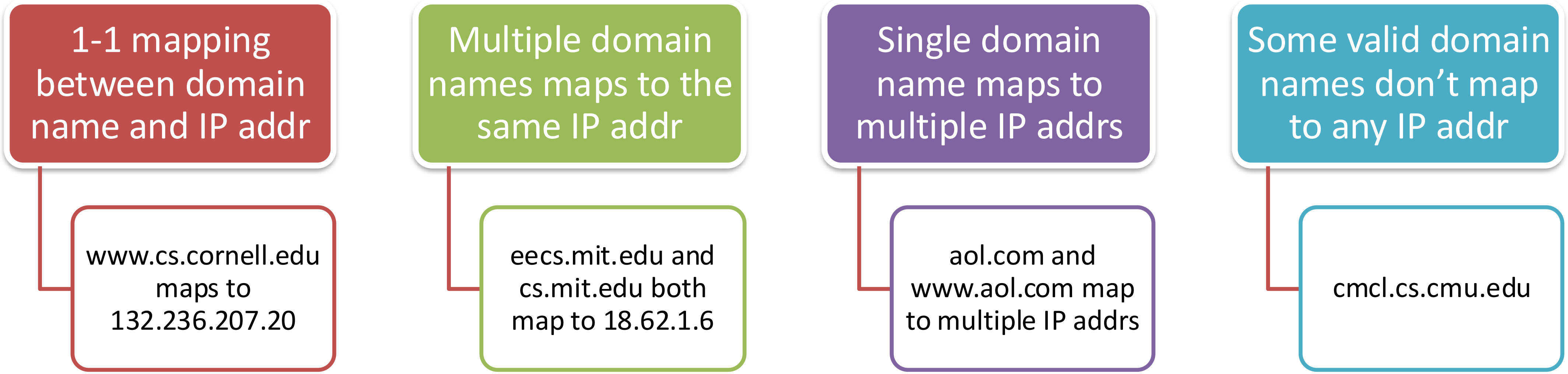
- Reverse lookup: IP address to hostname translation

Host name aliasing: other DNS names for a host

- Alias hostnames point to canonical hostname

Email: look up domain’s mail server by domain name

Different DNS Mappings



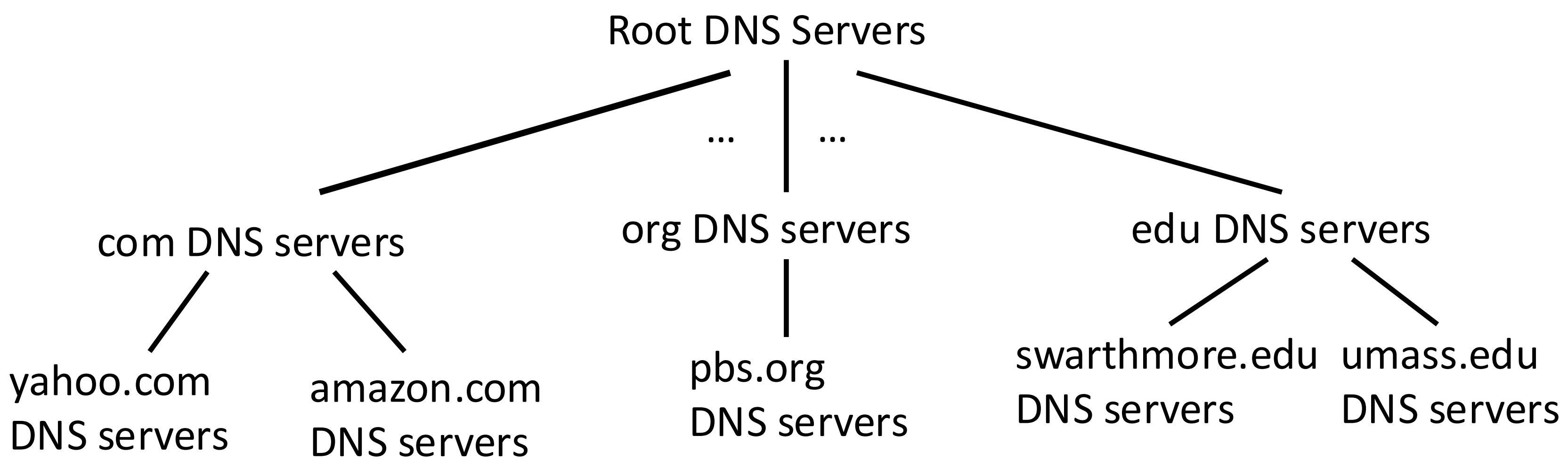
Domain Name System (DNS)

- Distributed administrative control
  - Hierarchical name space divided into zones
  - Distributed over a collection of DNS servers

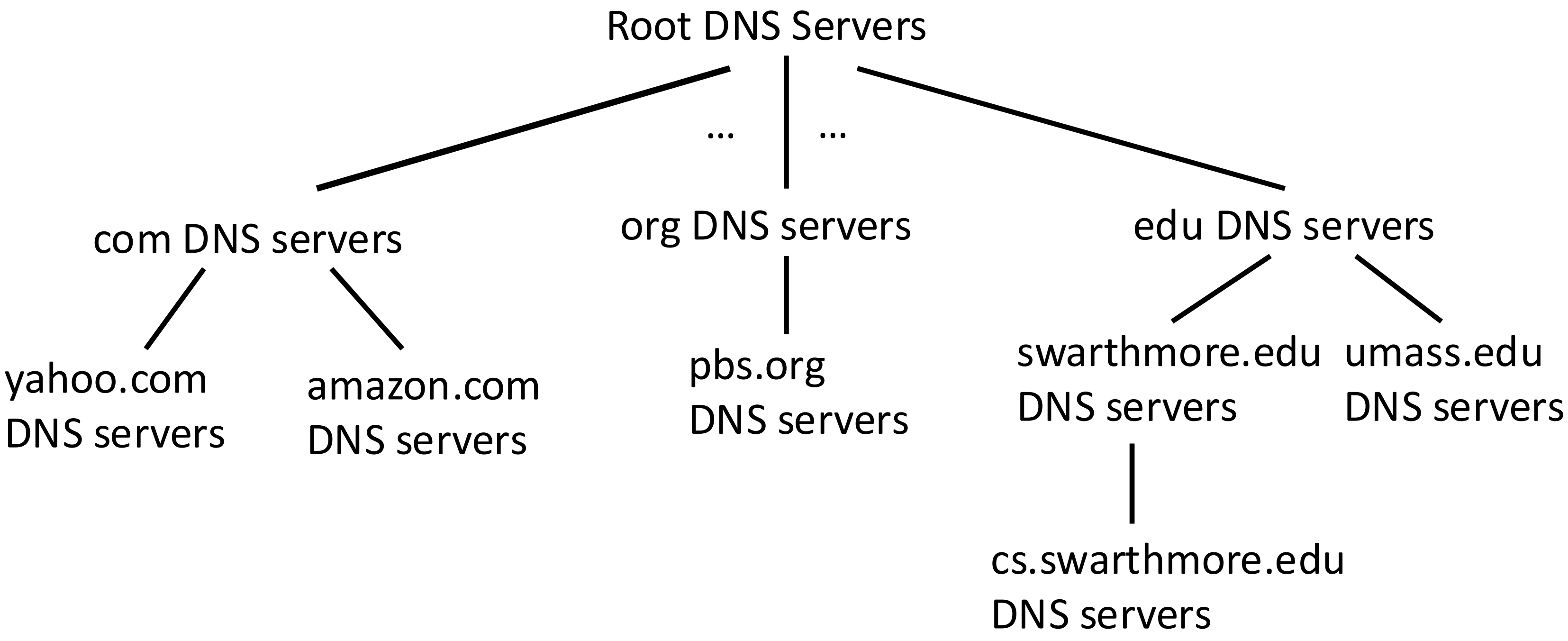
- Hierarchy of DNS servers
  - Root servers
  - Top-level domain (TLD) servers
  - Authoritative DNS servers

- Performing the translations
  - Local DNS servers
  - Resolver software

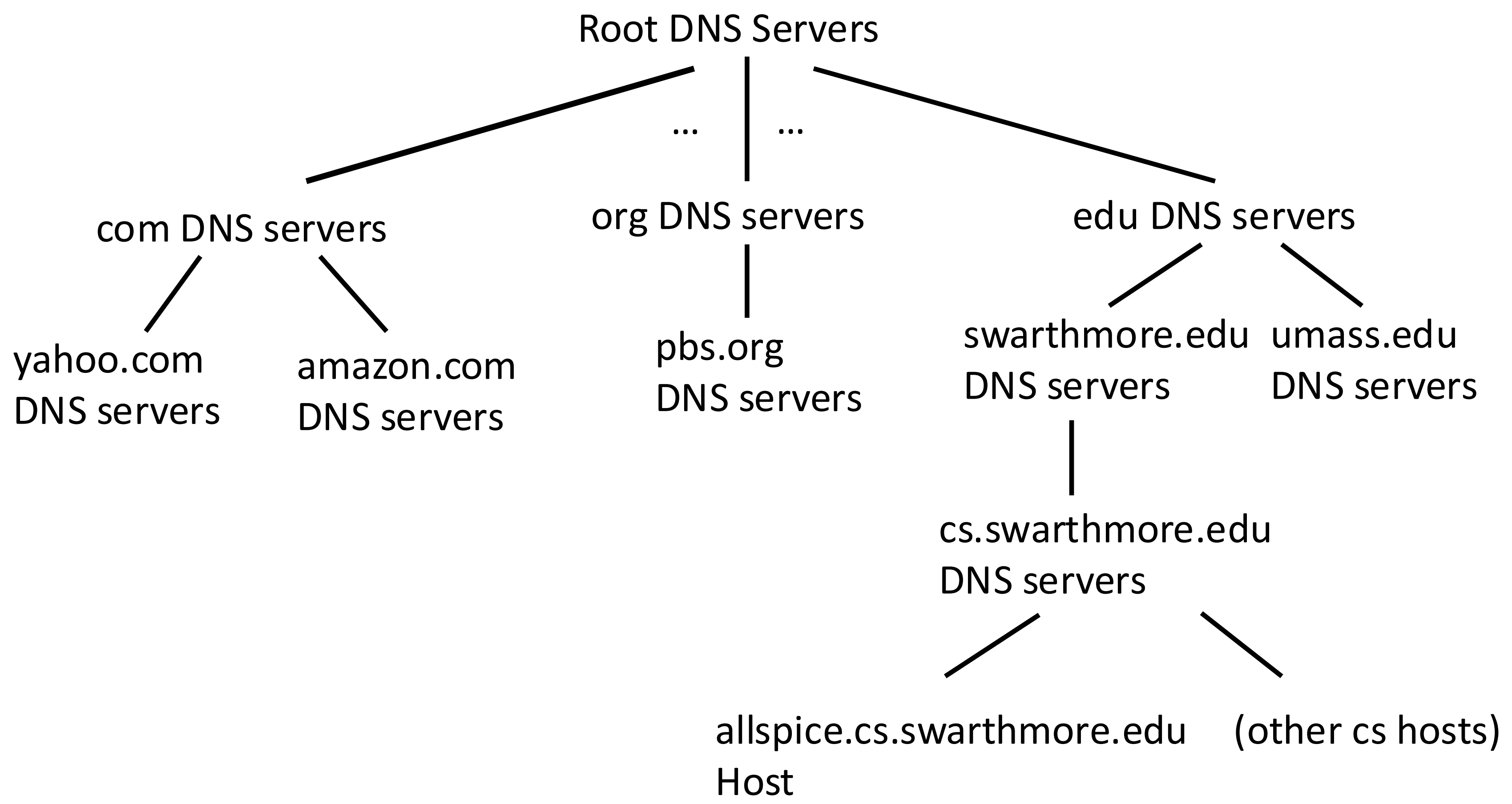
DNS: a distributed, hierarchical database



DNS: a distributed, hierarchical database



DNS: a distributed, hierarchical database



- allspice.cs.swarthmore.edu.

Nameless root,
Usually implied.

Why do we structure DNS like this? Which of these helps the most? Drawbacks?

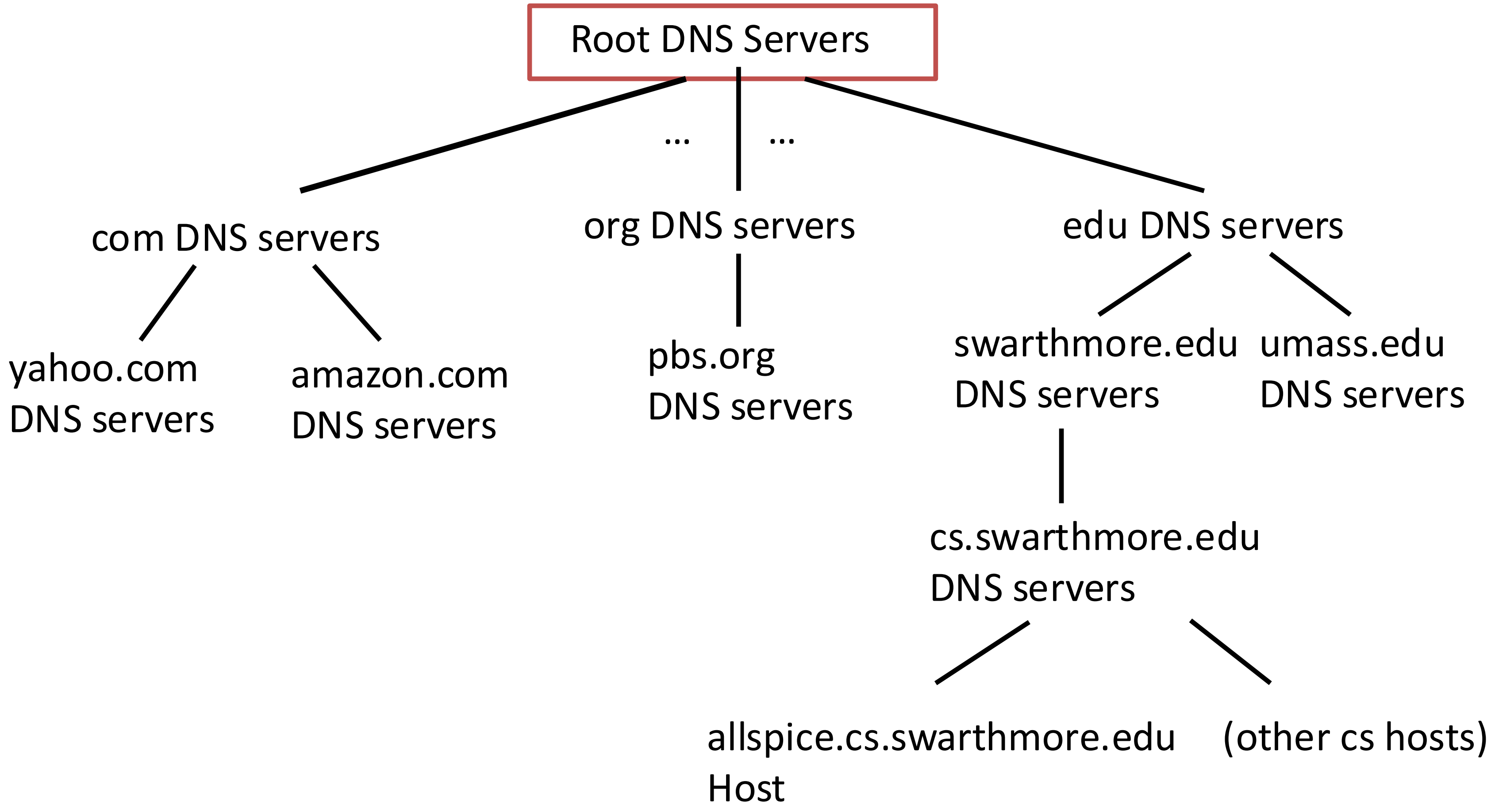
A. It divides up responsibility among parties.

B. It improves performance of the system.

C. It reduces the size of the state that a server needs to store.

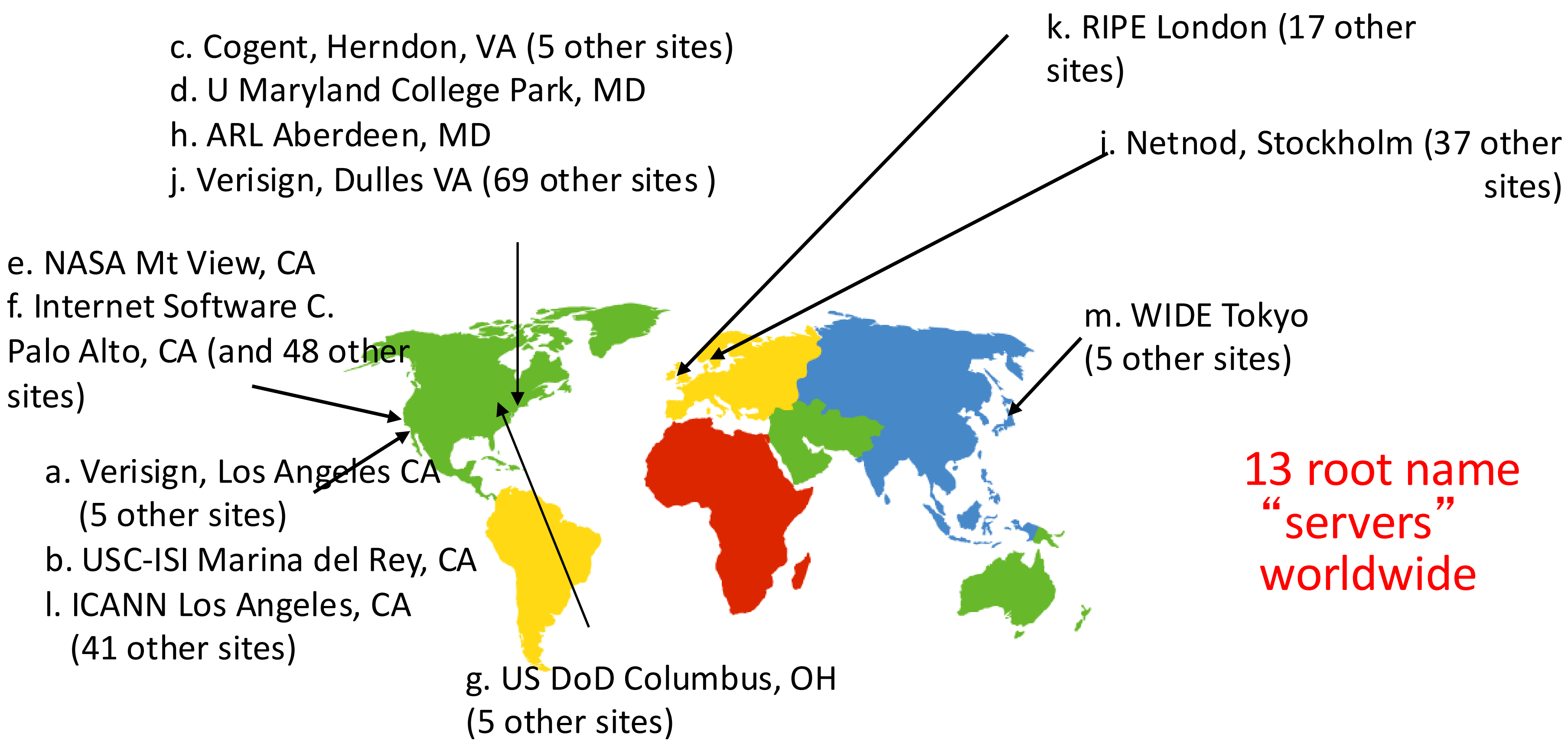
D. Some other reason.

DNS: a distributed, hierarchical database



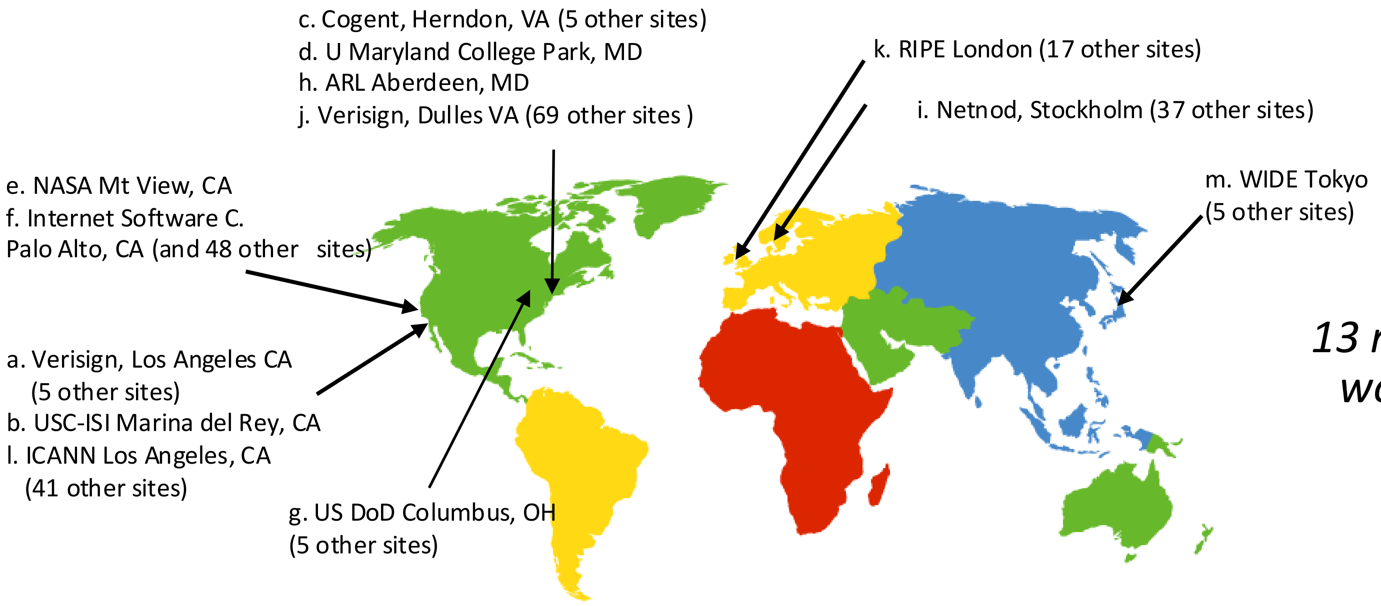
DNS: Root Name Servers

- Root name server:
  - Knows how to find top-level domains (.com, .edu, .gov, etc.)
  - How often does the location of a TLD change?



DNS: Root Name Servers

- Root name server:
  - Knows how to find top-level domains (.com, .edu, .gov, etc.)
  - How often does the location of a TLD change?
  - approx. 1900 total root servers (as of Dec 2025)
  - Significant amount of traffic is not legitimate



13 root name "servers" worldwide

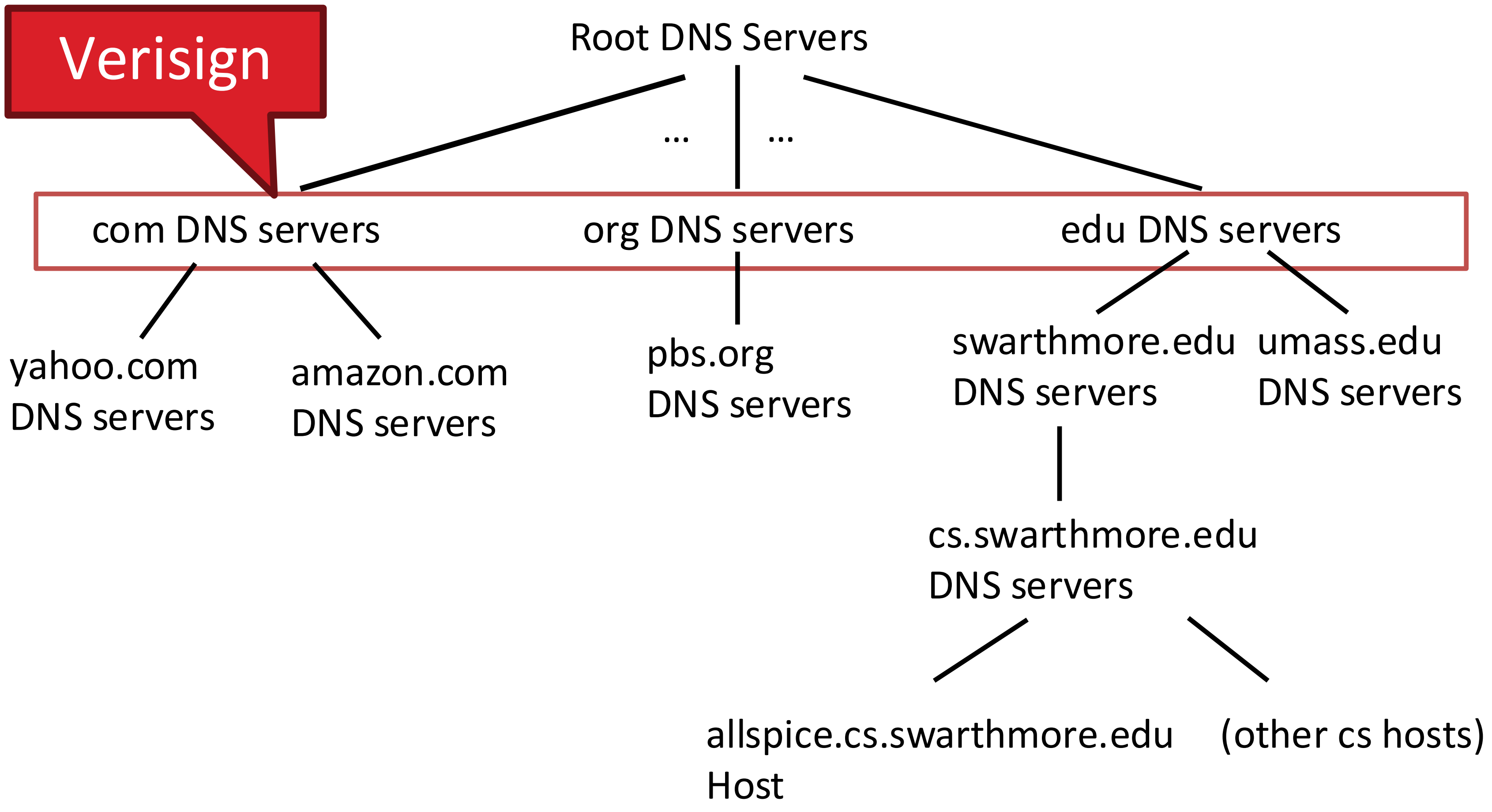
Root Name Servers

- Responsible for the Root Zone File

```
com.        172800 IN   NS  a.gtld-servers.net.
com.        172800 IN   NS  b.gtld-servers.net.
com.        172800 IN   NS  c.gtld-servers.net.
```

  - In practice, most systems cache this information
  - Lists the TLDs and who controls them
  - ~272KB in size

Top Level Domain (TLD) servers

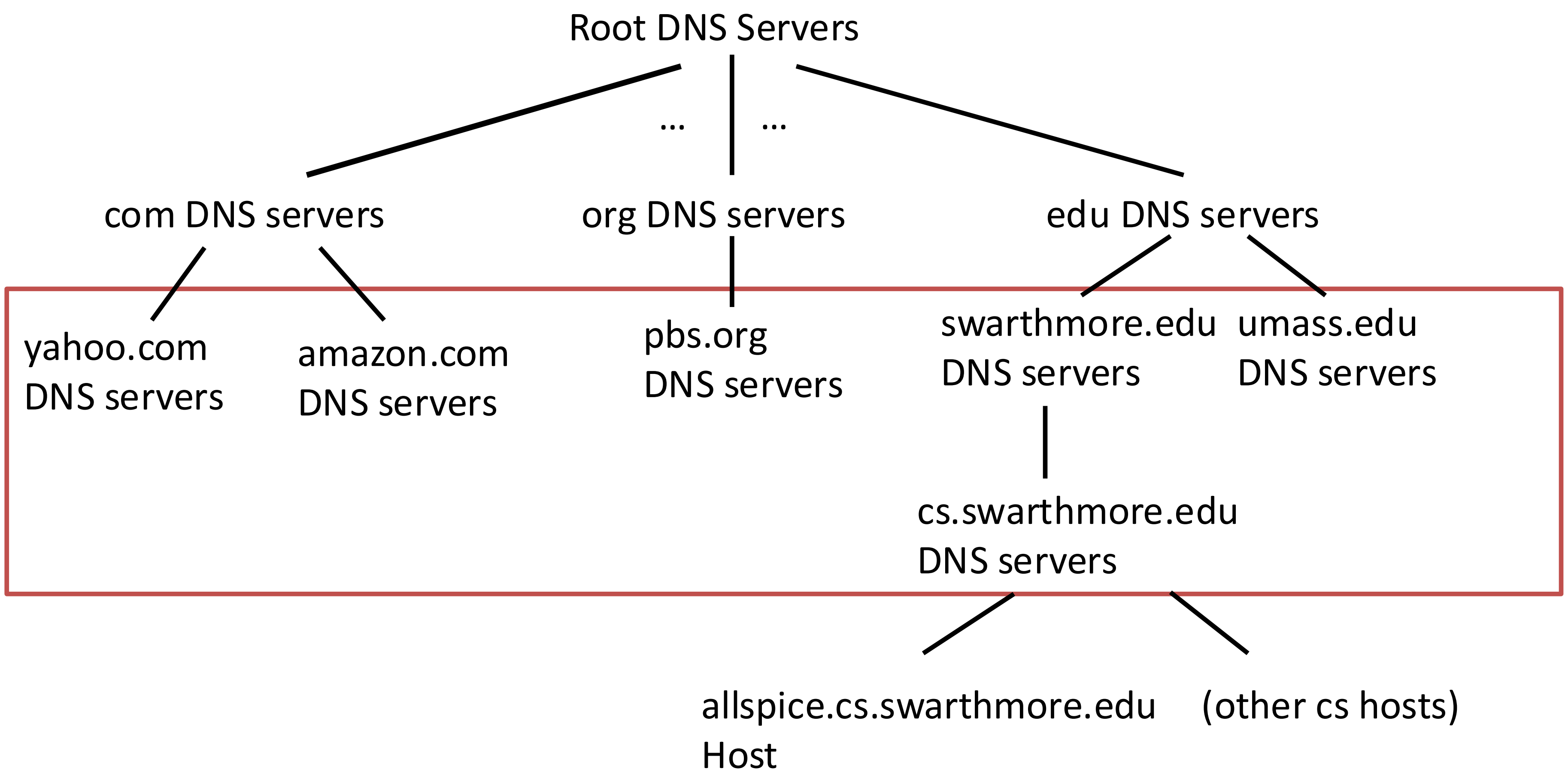


Top Level Domain (TLD) servers

- who maintains the servers?:
  - Verisign: .com, .net
  - Educause: .edu (Verisign backend)
  - local governments or companies

- Responsible for:
  - com, org, net, edu, gov, aero, jobs, museums,
  - all top-level country domains, e.g.: uk, fr, de, ca, jp, etc

Authoritative Servers



Authoritative Servers

Authoritative DNS servers:

- Organization’s own DNS server(s), providing authoritative hostname to IP mappings for organization’s named hosts
- Can be maintained by organization or service provider, easily changing entries
- Often, but not always, acts as organization’s local name server (for responding to look-ups)

Resolution Process

- End host wants to look up a name, who should it contact?
  - It could traverse the hierarchy, starting at a root
  - More efficient for ISP to provide a local server

- ISP’s local server for handling queries not necessarily a part of the pictured hierarchy

Local DNS Name Server

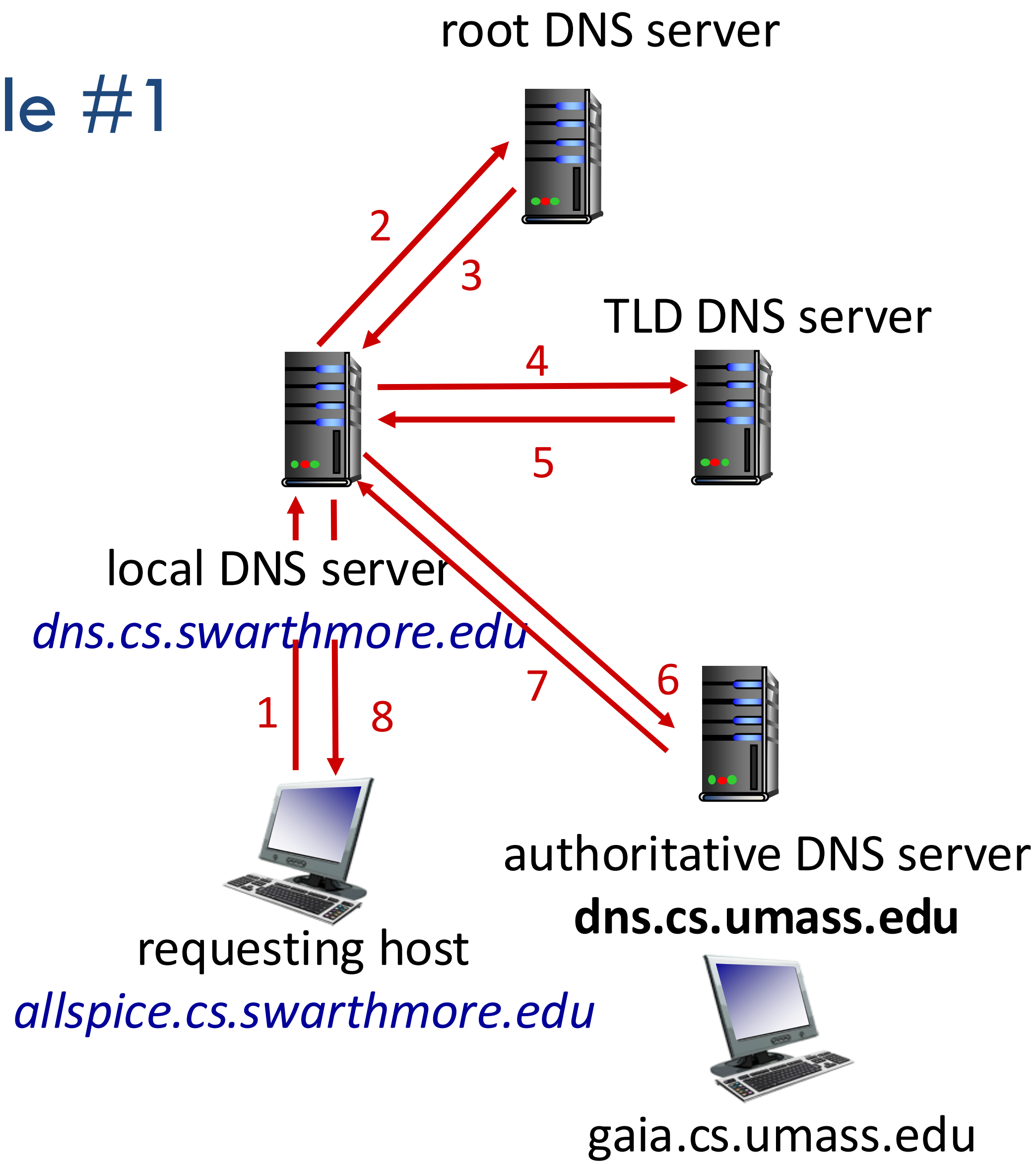
- Each ISP (residential ISP, company, university) has (at least) one
  - also called “default name server”
- When host makes DNS query, query is sent to its local DNS server
  - has local cache of recent name-to-address translation pairs (but may be out of date!)
  - acts as proxy, forwards query into hierarchy

DNS name resolution example #1

- allspice wants IP address for gaia.cs.umass.edu

iterative query:

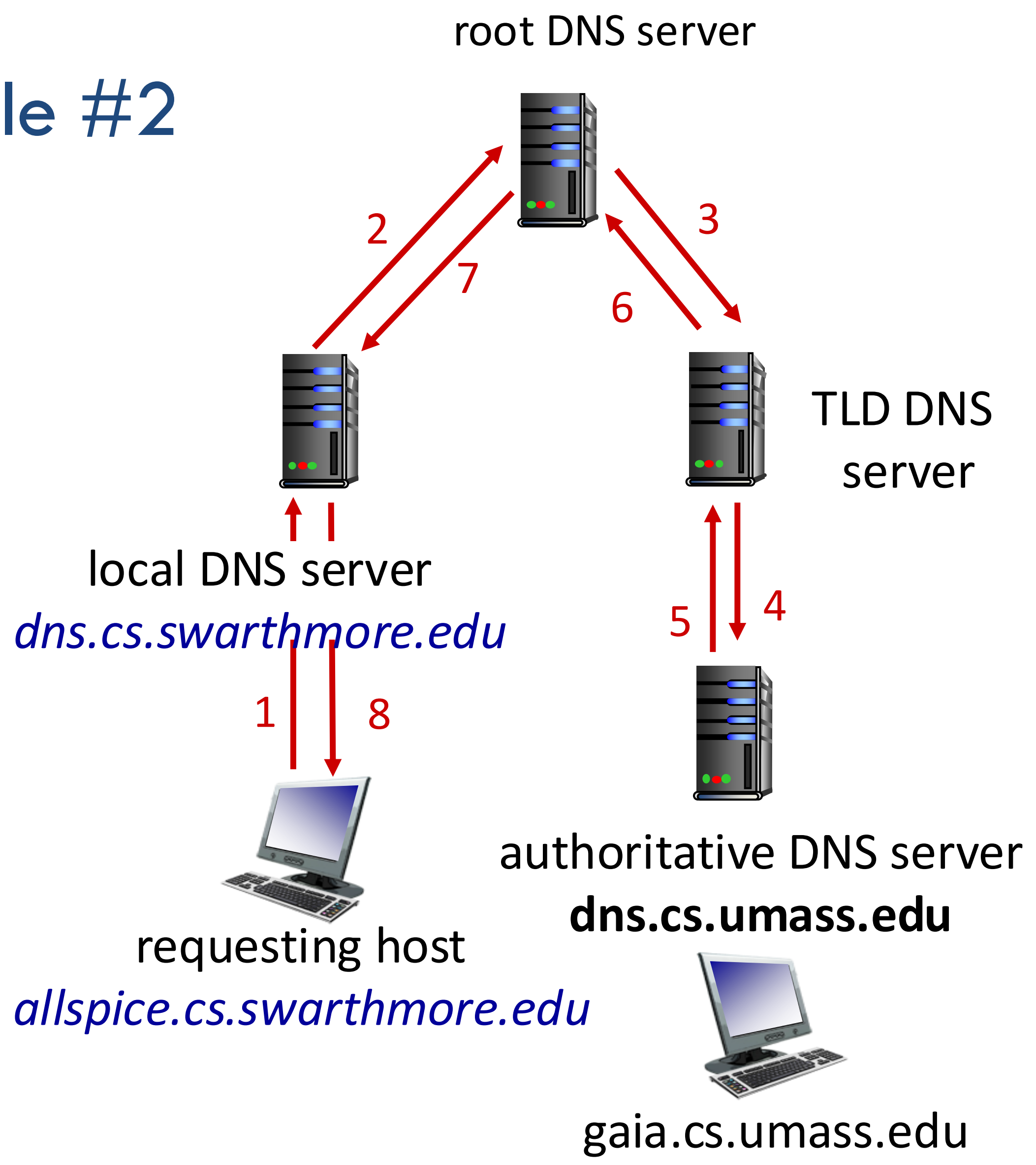
- contacted server replies with name of server to contact
- "I don't know this name, but ask this server"



DNS name resolution example #2

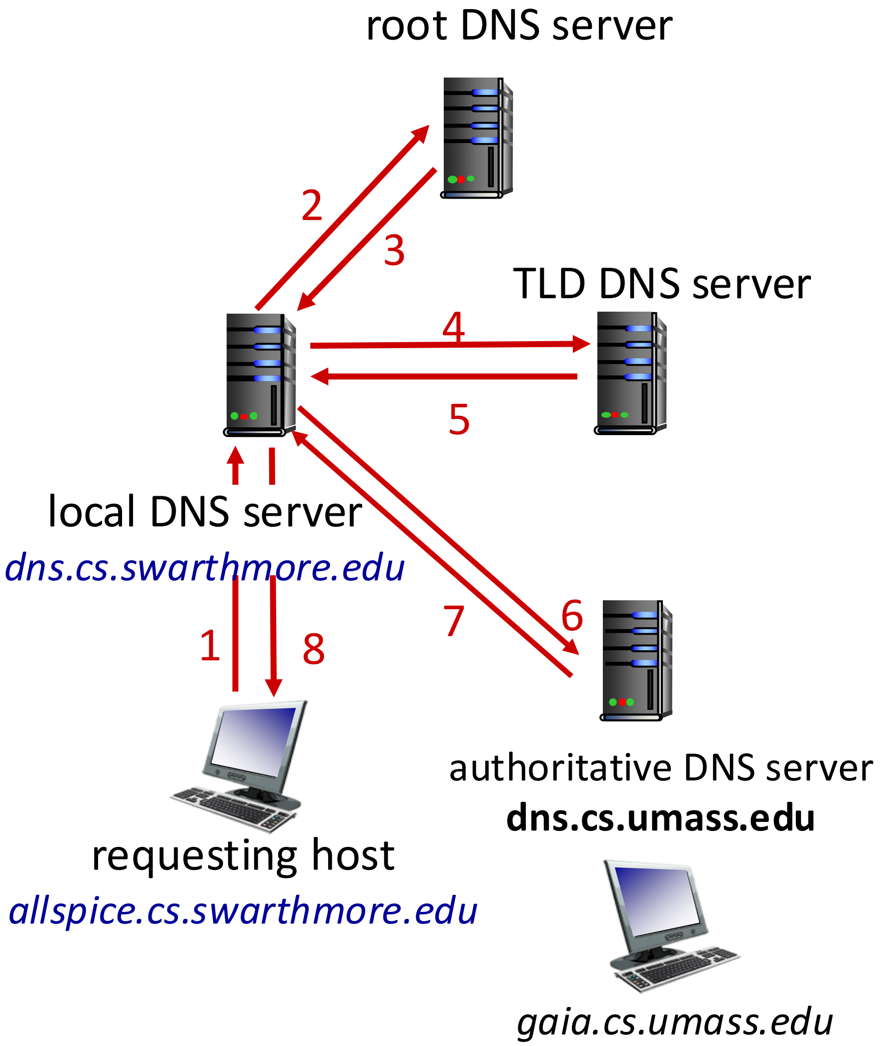
recursive query:

- each server asks the next one, in a chain

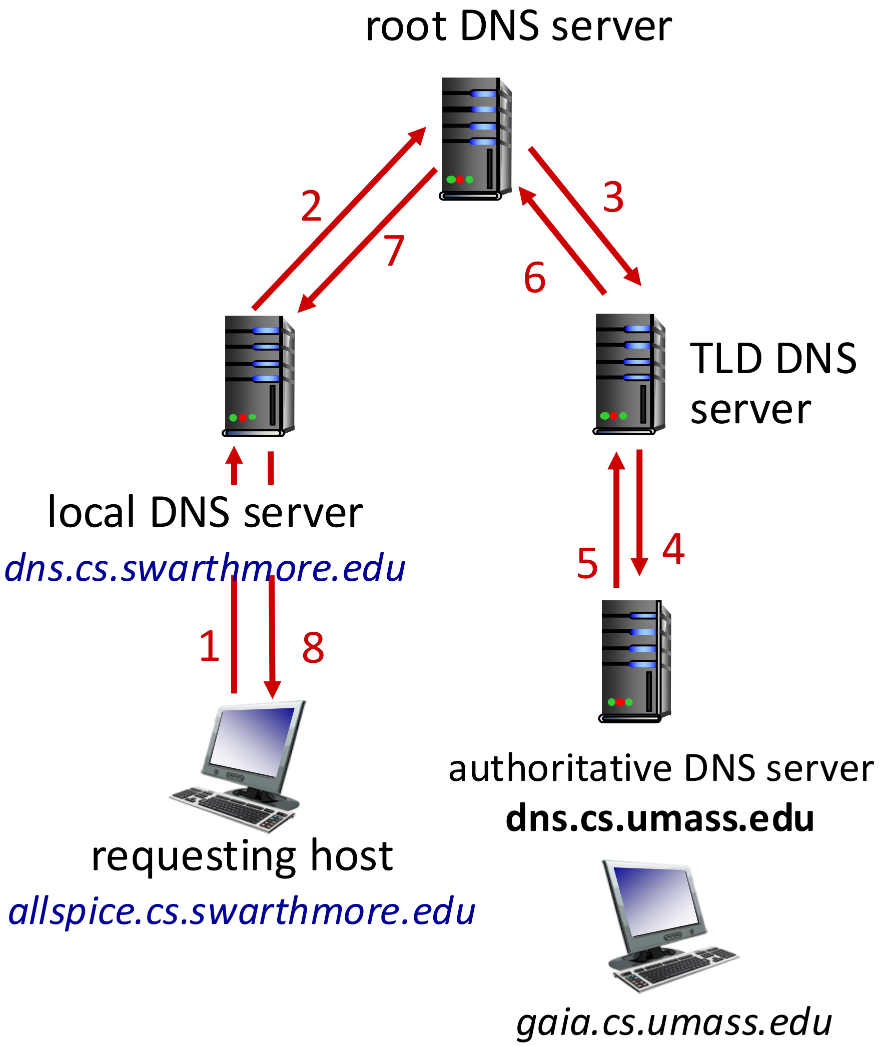


Which would you use? Why?

A. Iterative



B. Recursive



Example: iterative query using dig()

dig . ns

dig +norec demo.cs.swarthmore.edu @a.root-servers.net

dig +norec demo.cs.swarthmore.edu @a.edu-servers.net

dig +norec demo.cs.swarthmore.edu @ibext.its.swarthmore.edu

demo.cs.swarthmore.edu. 259200 IN A 130.58.120.152