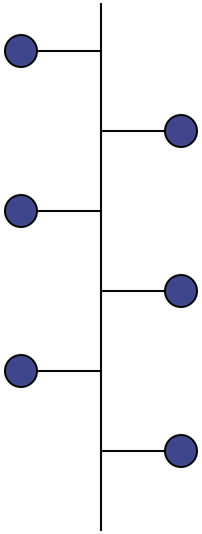
Rechnernetze und Verteilte Systeme

Introduction to Communication Networks and Distributed Systems



Unit 4a: From WWW to Web Services

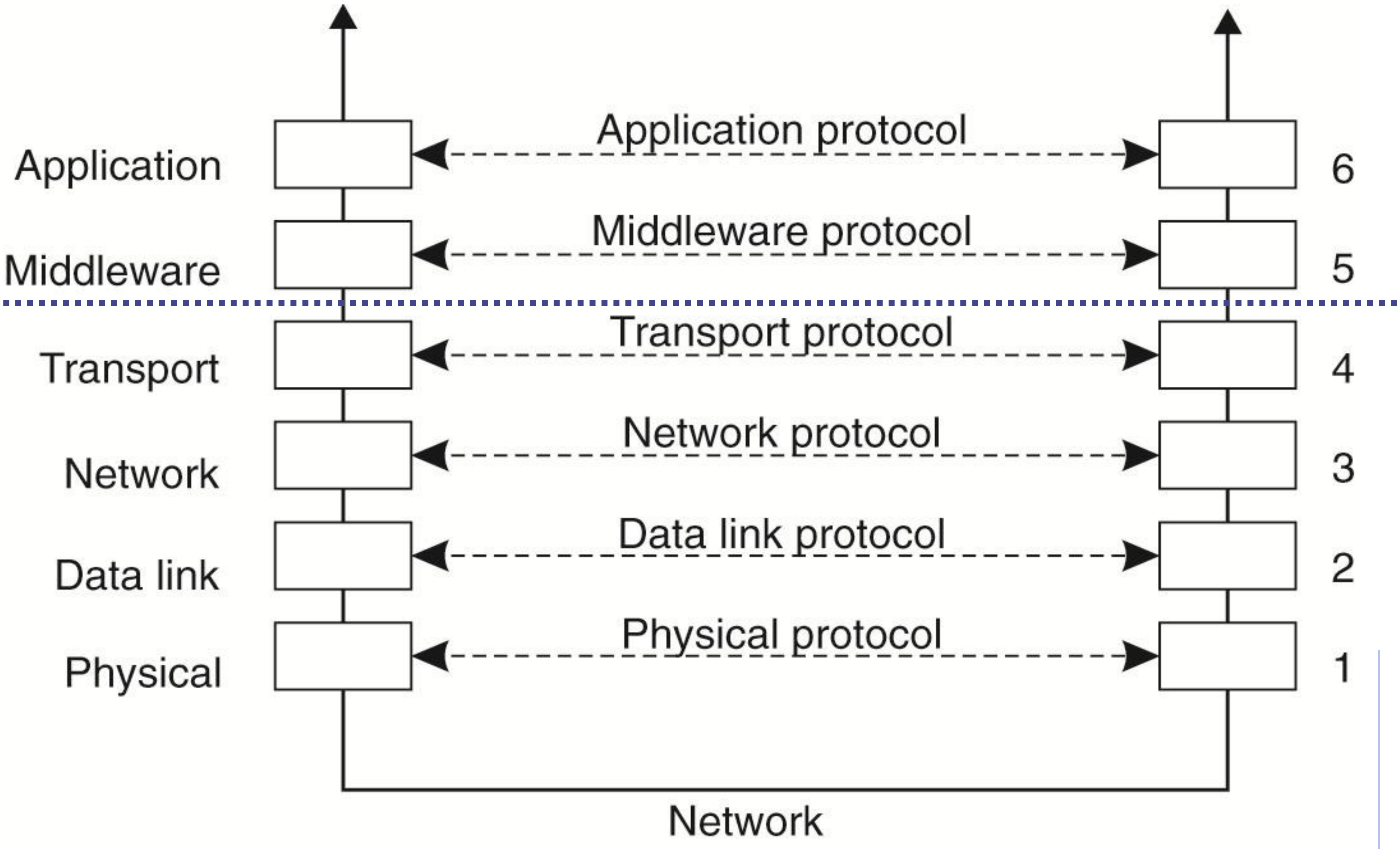
Prof. Dr.-Ing. Adam Wolisz

TKN **Telecommunication Networks Group**

Acknowledgements:

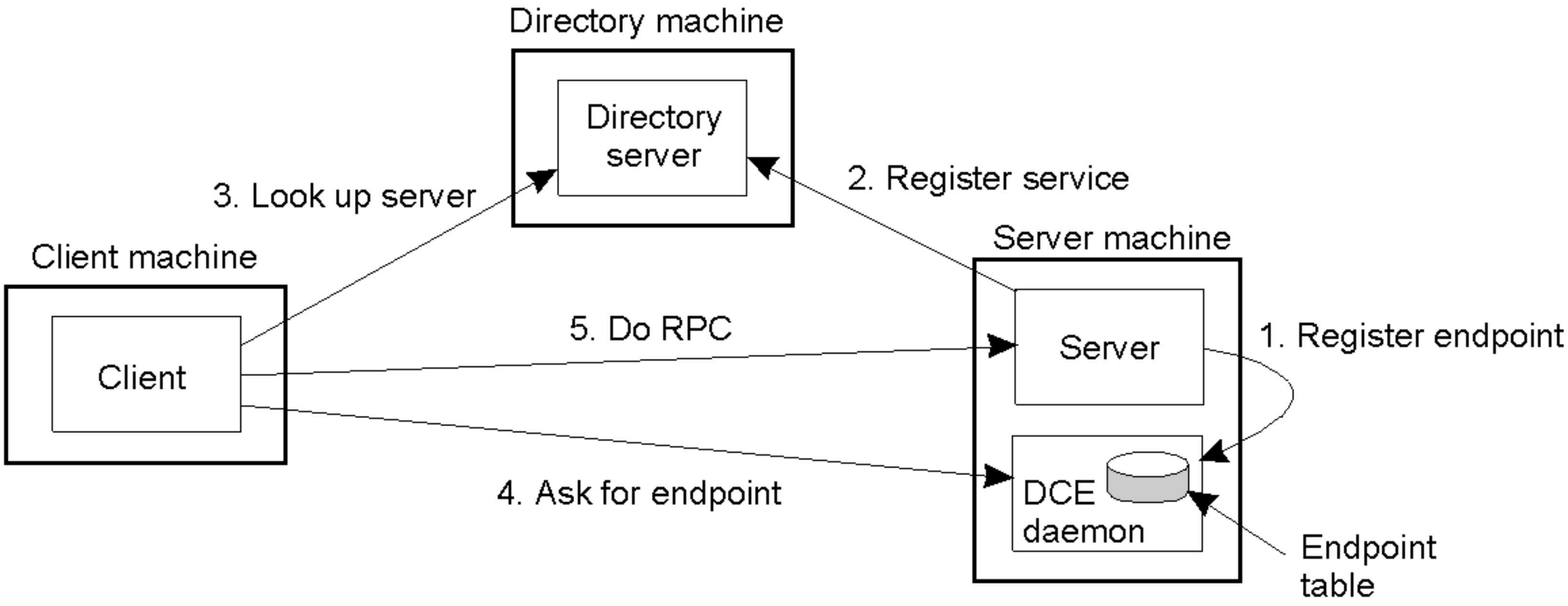
- We acknowledge the use of slides from: Prof. Holger Karl, Paderborn; Prof. Ion Stoica, Berkeley, Prof. Ashay Parekh; Berkeley; Prof Lauer WPI; Prof. Baker, ACET, as well as slides form books by Tannenbaum, Kurose and Ross, Colouris at al.

An adapted communication model [Tanenbaum]



Binding a Client to a Server

- Client-to-server binding in Distributed Computing Environment



Notions

- Names
  - Data types that refer to specific entities in a system

    Example: ***Jonathan M. Smith***
- Addresses
  - Identifiers of places to find the named entities

    Example: ***123 Park St., Andover, MA***
- Routes (also called *paths*)
  - steps to follow to get to named entities

    Example:

    - ***From Andover center, go west 1.2 miles***
    - ***Turn right, then take 3rd left***
    - ***He lives at the 2nd house on the righ***

Name Service

- Name space: set of possible names and their relationship

  (Names are usually chosen so as to be meaningful/easy to memorize)

- Bindings: the mapping between names and addresses
- Name resolution: procedure that, when invoked with a name, returns the corresponding value
- Name server (*directory*): specific implementation of a resolution mechanism that is available on the network and that can be queried by sending messages

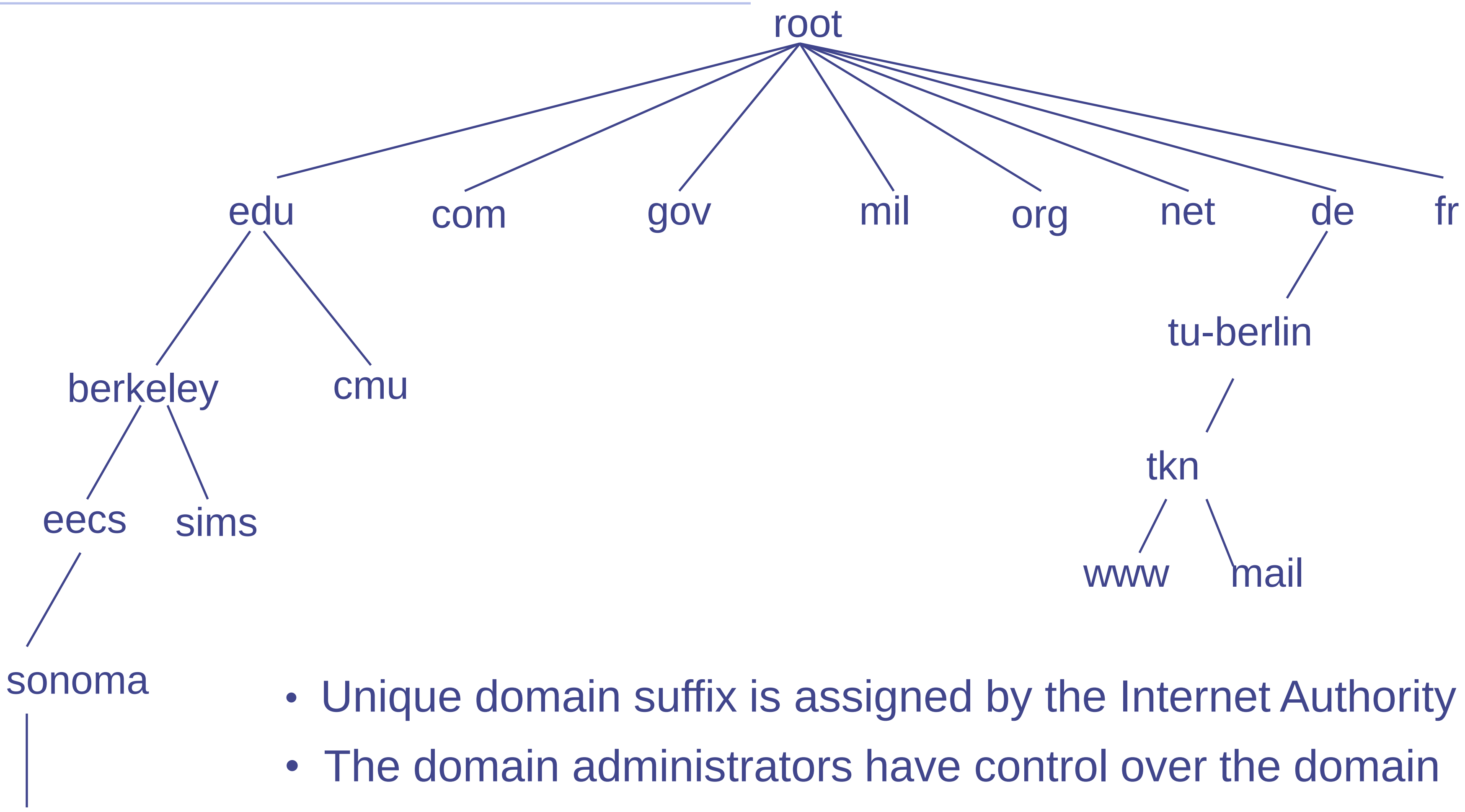
➔ Uses a distributed data base !!!

Internet → DNS *(Features)*

- Hierarchical Namespace
- Distributed architecture for storing names
  - Name servers assigned zones of the hierarchical namespace
  - Backup servers available for redundancy
- Administration divided along the same hierarchy

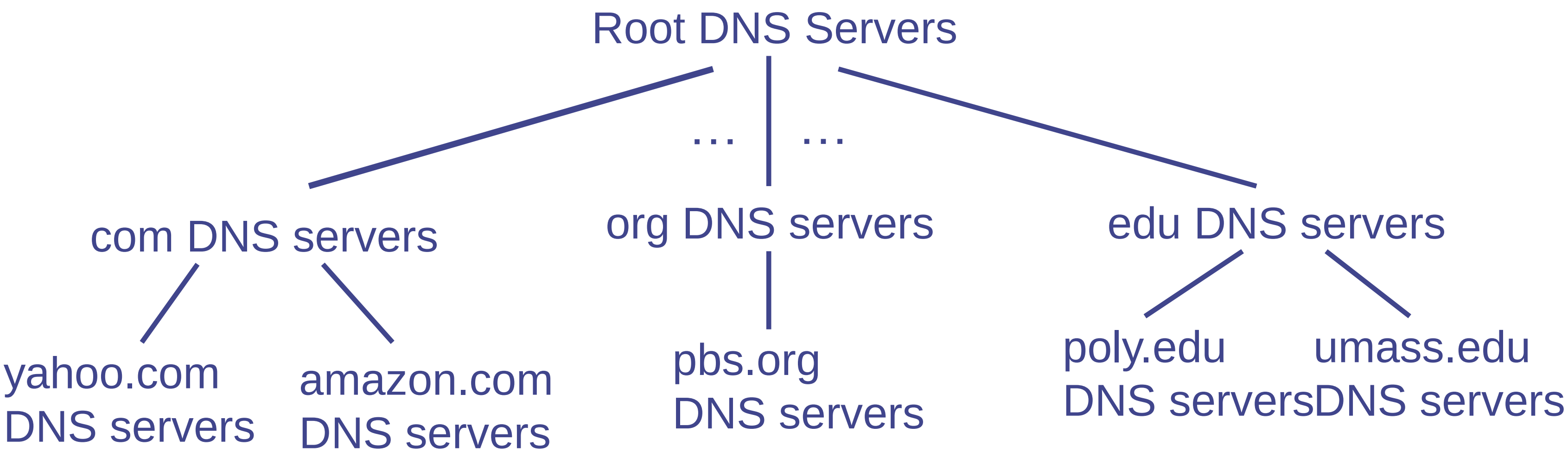
- Client server interaction on Connectionless (UDP) Port 53

Domain Name System (DNS) Hierarchy



- Unique domain suffix is assigned by the Internet Authority
- The domain administrators have control over the domain
- No limit on the number of subdomains or number of levels
- Name space is not related with the physical interconnection
- ´ A name could be a domain or an individual object

DNS: a distributed, hierarchical database



Root DNS Servers

…

…

com DNS servers

org DNS servers

edu DNS servers

yahoo.com
DNS servers

amazon.com
DNS servers

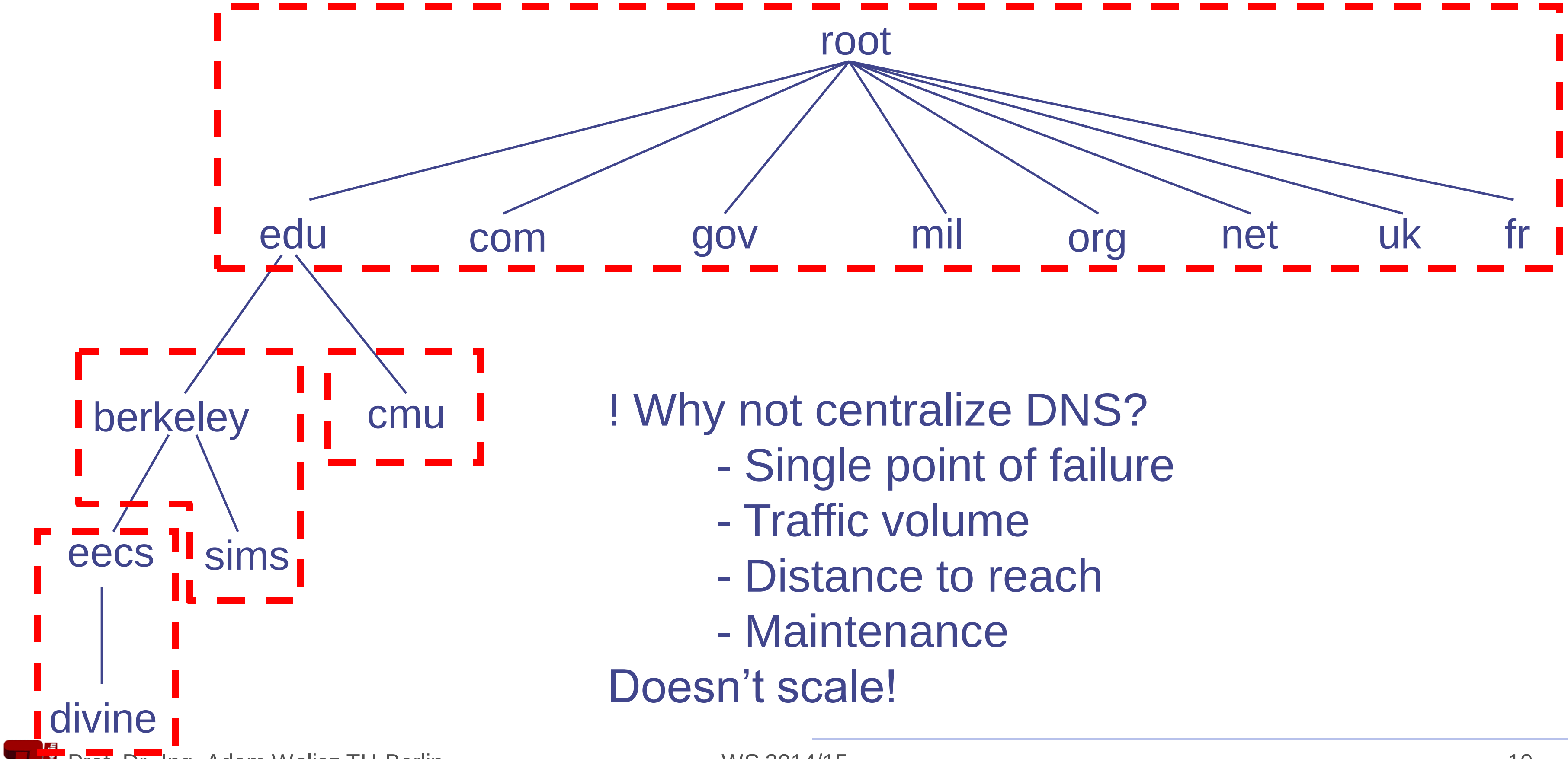
pbs.org
DNS servers

poly.edu
DNS servers

umass.edu
DNS servers

Server Hierarchy: Zones

- A zone corresponds to an administrative authority that is responsible for that portion of the hierarchy



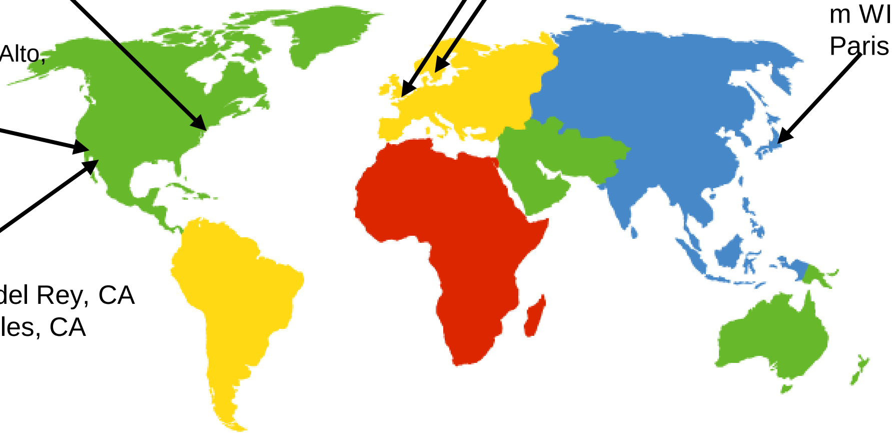
! Why not centralize DNS?

- Single point of failure
- Traffic volume
- Distance to reach
- Maintenance

Doesn’t scale!

DNS: Root name servers

- contacted by local name server that can not resolve name
- Is a critical part of the Internet!!
- Therefore replicated - 13 root name servers worldwide
- Have to carry consistent content…



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)

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

m WIDE Tokyo (also Seoul, Paris, SF)

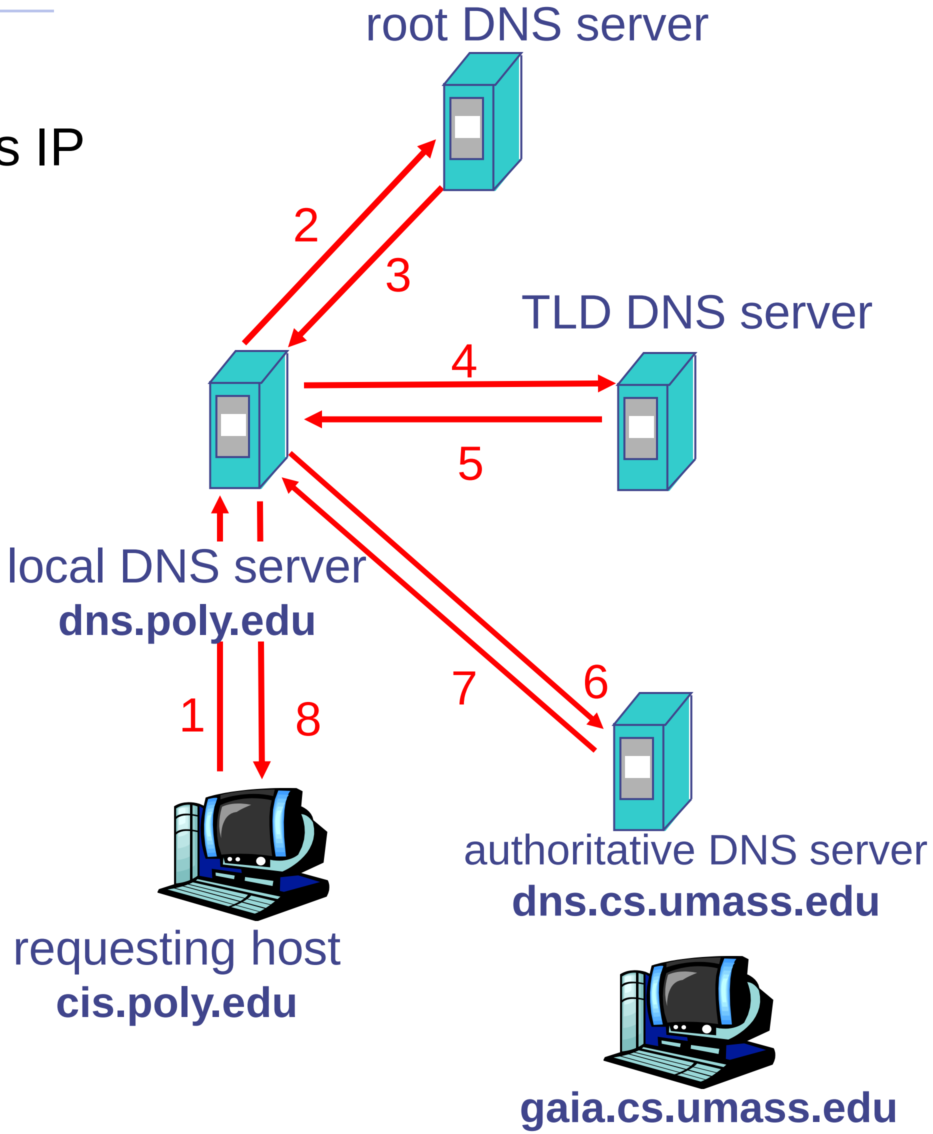
b USC-ISI Marina del Rey, CA
l ICANN Los Angeles, CA

DNS name resolution

➔host at cis.poly.edu wants IP address for gaia.cs.umass.edu

iterated query:

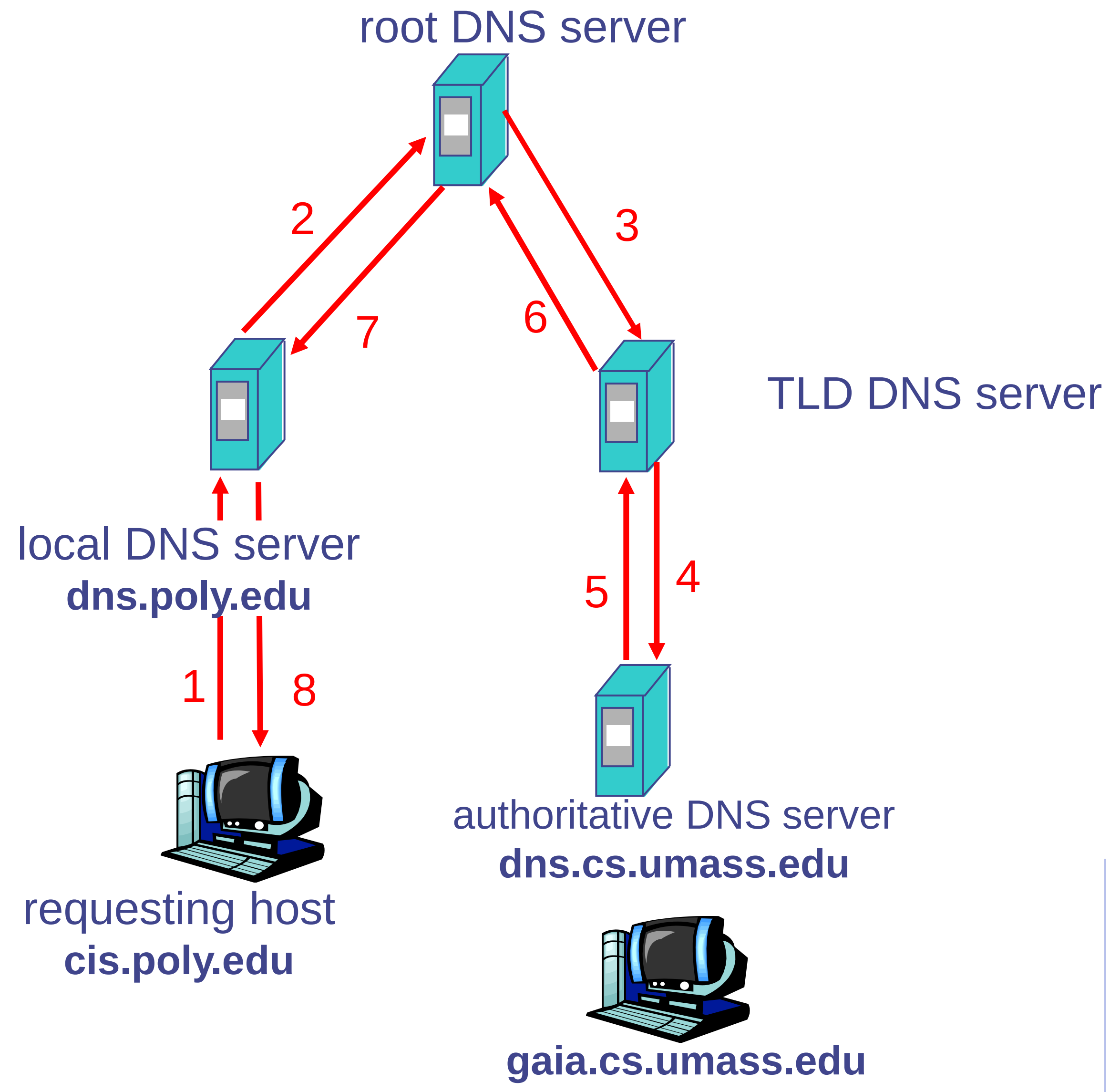
- contacted server replies with name of server to contact
- "I don't know this name, but ask this server"



Alternatively: Recursive Query Resolution

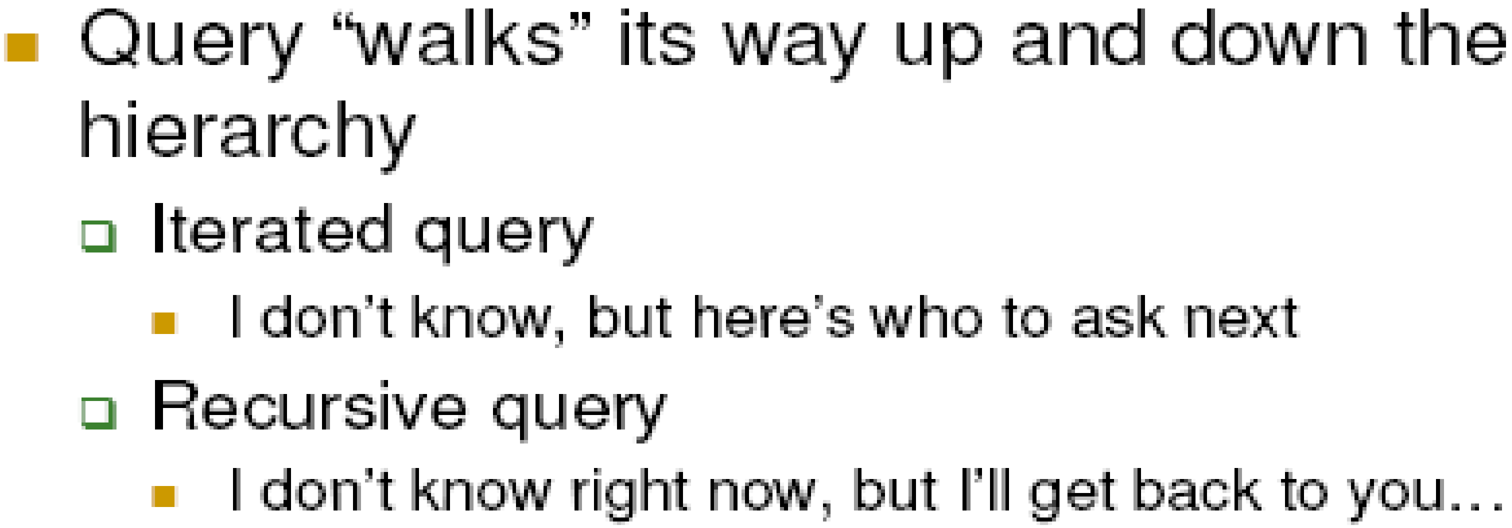
recursive query:

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



Autoritative/Local Name server

- When a host makes a DNS Query, query is sent to its LOCAL DNS server.
- If this server is not able to resolve, query starts walking up the hierarchy, until somebody can point out the Authoritative server for the queried domain…
- Afterwards the query „walks down“ until the proper sub-domain is found



- Query “walks” its way up and down the hierarchy
  - □ Iterated query
    - I don’t know, but here’s who to ask next
  - □ Recursive query
    - I don’t know right now, but I’ll get back to you…

DNS Caching

- Performing all these queries takes time
  - And all this before actual communication takes place
  - E.g., 1-second latency before starting Web download
- Caching can greatly reduce overhead
  - The top-level servers very rarely change
  - Popular sites (e.g., www.cnn.com) visited often
  - Local DNS server often has the information cached
- How DNS caching works
  - DNS servers cache responses to queries
  - Responses include a “time to live” (TTL) field
  - Server deletes cached entry after TTL expires

DNS and Virtual IP addresses

DNS records don’t have to store the real IP address of the host.

- Multiple names can map onto the same address
  - Example: www.berkeley.edu and arachne.berkeley.edu maps to the same machine (i.e., the same IP address)
  - Example: All hosts in the acme.com may have the same IP address/port – in reality a gatekeeper
    - The Gatekeeper decides whether to “admit” a transport level connection to the host x.acme.com (the ***firewall*** functionality)
    - The Gatekeeper decides to forward the connection to one of several identical servers (A ***load balancer*** functionality)

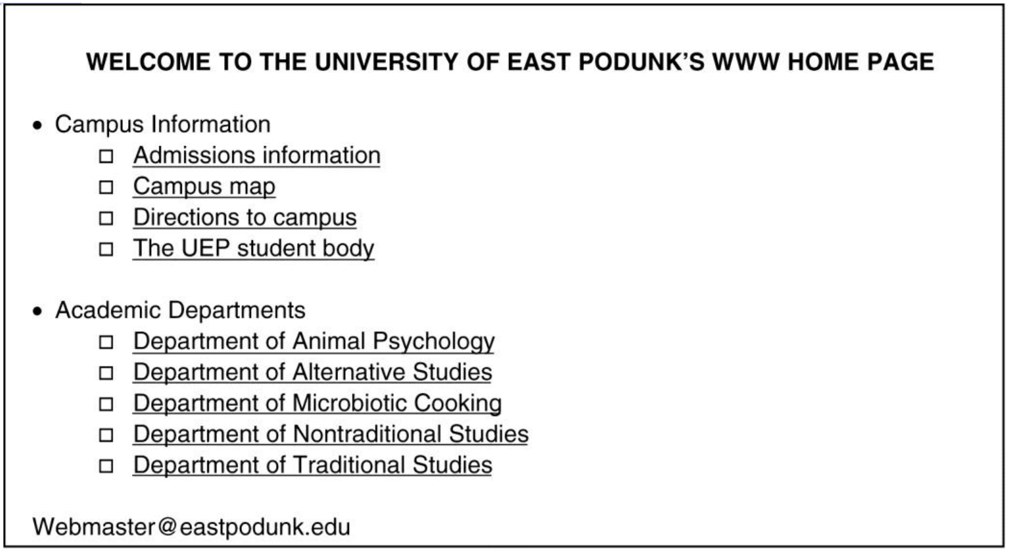
WWW

WWW

- we use it,
- do we understand it?

The notion of a hyperlink is – in fact – very old...

The technical implementation: information system instead of books- makes the difference!

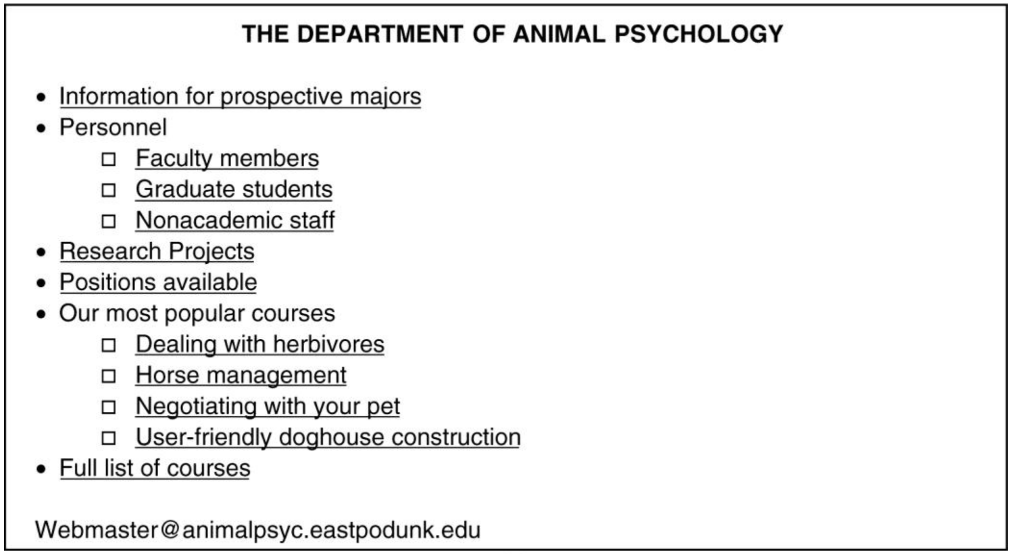


WELCOME TO THE UNIVERSITY OF EAST PODUNK'S WWW HOME PAGE

- Campus Information
  - □ Admissions information
  - □ Campus map
  - □ Directions to campus
  - □ The UEP student body
- Academic Departments
  - □ Department of Animal Psychology
  - □ Department of Alternative Studies
  - □ Department of Microbiotic Cooking
  - □ Department of Nontraditional Studies
  - □ Department of Traditional Studies

Webmaster@eastpodunk.edu

(a)



THE DEPARTMENT OF ANIMAL PSYCHOLOGY

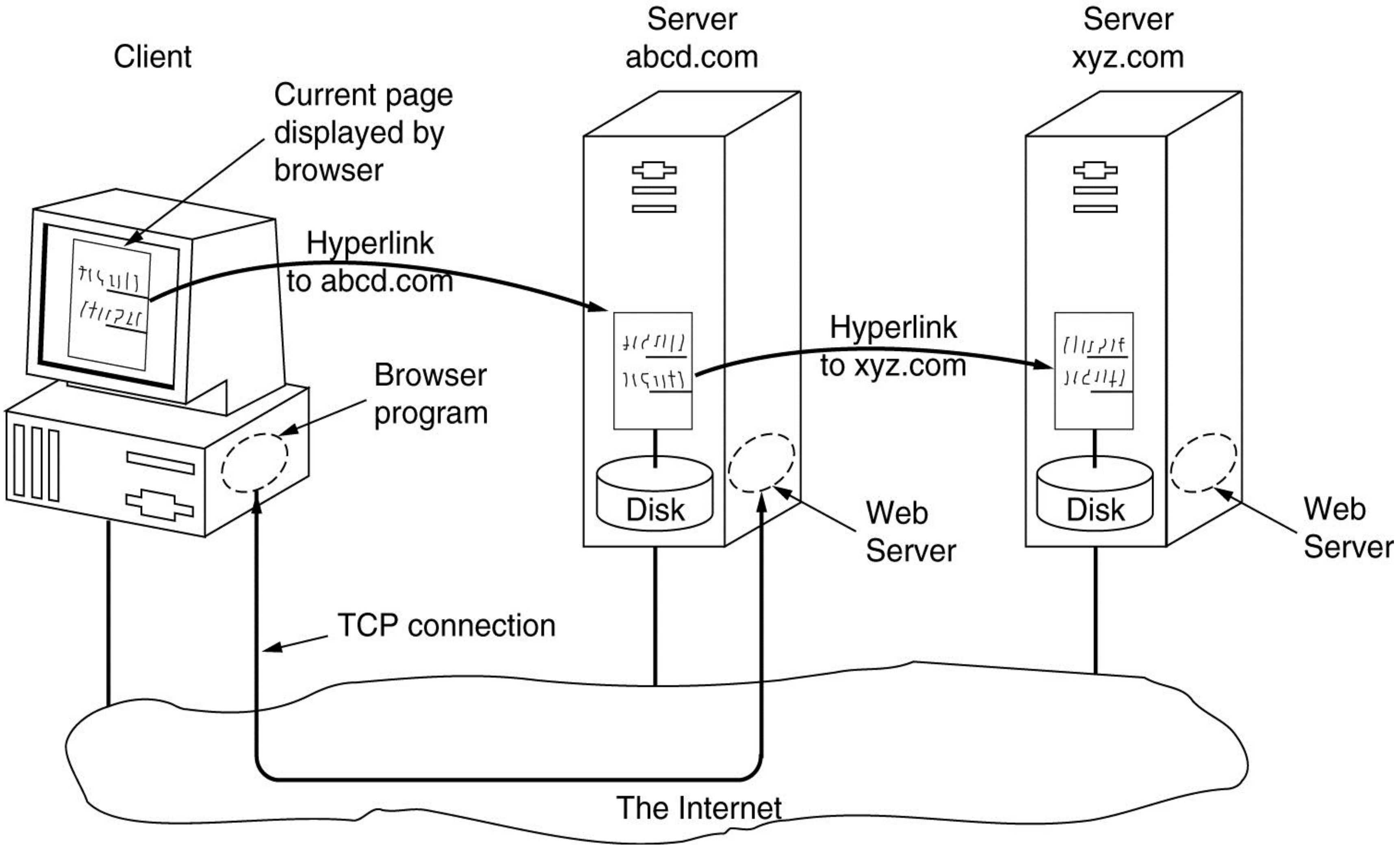
- Information for prospective majors
- Personnel
  - □ Faculty members
  - □ Graduate students
  - □ Nonacademic staff
- Research Projects
- Positions available
- Our most popular courses
  - □ Dealing with herbivores
  - □ Horse management
  - □ Negotiating with your pet
  - □ User-friendly doghouse construction
- Full list of courses

Webmaster@animalpsyc.eastpodunk.edu

(b)

Architectural Overview

[Tanenbaum]



The WWW

- A distributed database of URLs (*Uniform Resource Locators*)
- Core components:
  - Servers which store “web pages” and execute remote commands
  - Browsers retrieve and display “pages” of content linked by hypertext
    - Each link is a URL

➔ Can build arbitrarily complex applications, all of which share a uniform client!

- Need a language to define the objects and the layout
  - HTML, XML
- Need a protocol to transfer information between clients and servers
  - HTTP

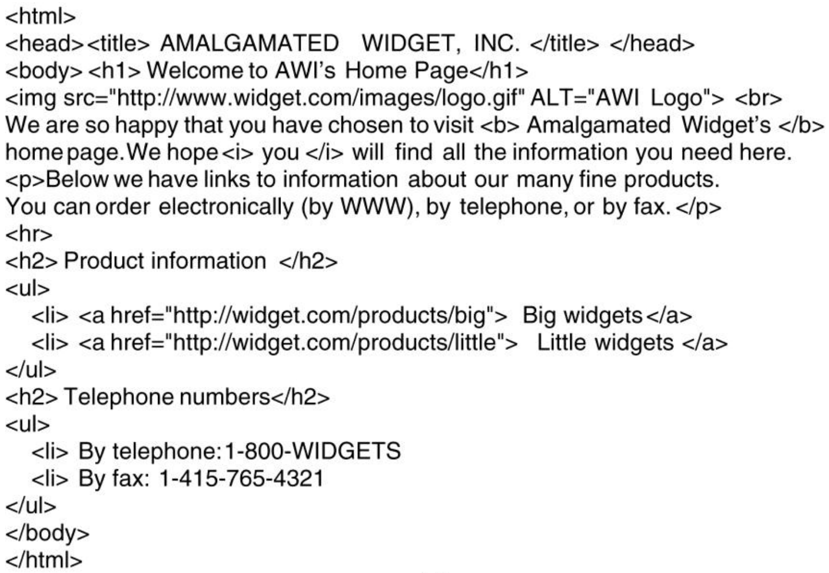
Uniform Record Locator

- Defines *(roughly)* the Access Method and the Path….
- E.g. protocol://host-name:port/directory-path/resource
- http**://www.eecs.berkeley.edu/122/Lecture6/presentation.ppt**

HTML – HyperText Markup Language

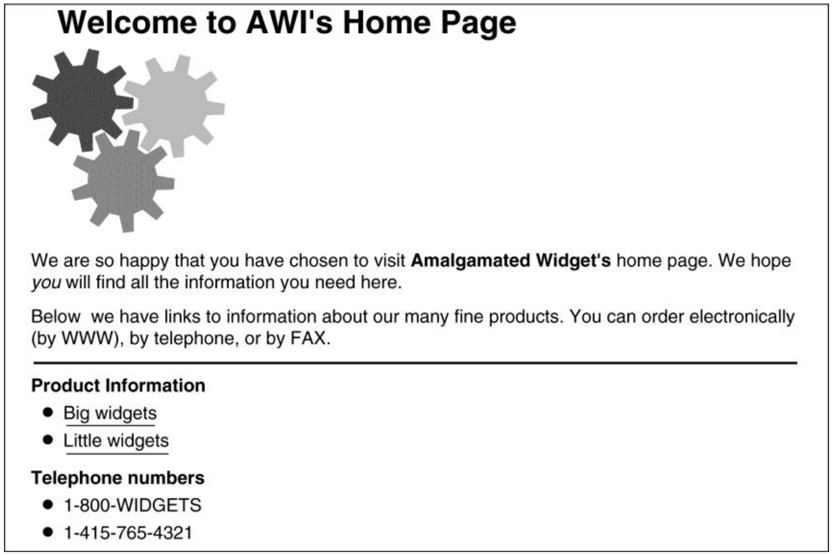
(a) The HTML for a sample Web page.

(b) The formatted page.



```
<html>
<head><title> AMALGAMATED  WIDGET, INC. </title> </head>
<body> <h1> Welcome to AWI's Home Page</h1>
<img src="http://www.widget.com/images/logo.gif" ALT="AWI Logo"> <br>
We are so happy that you have chosen to visit <b> Amalgamated Widget's </b>
home page. We hope <i> you </i> will find all the information you need here.
<p>Below we have links to information about our many fine products.
You can order electronically (by WWW), by telephone, or by fax. </p>
<hr>
<h2> Product information </h2>
<ul>
  <li> <a href="http://widget.com/products/big"> Big widgets </a>
  <li> <a href="http://widget.com/products/little"> Little widgets </a>
</ul>
<h2> Telephone numbers</h2>
<ul>
  <li> By telephone: 1-800-WIDGETS
  <li> By fax: 1-415-765-4321
</ul>
</body>
</html>
```

(a)



Welcome to AWI's Home Page

We are so happy that you have chosen to visit **Amalgamated Widget's** home page. We hope *you* will find all the information you need here.

Below we have links to information about our many fine products. You can order electronically (by WWW), by telephone, or by FAX.

Product Information

- Big widgets
- Little widgets

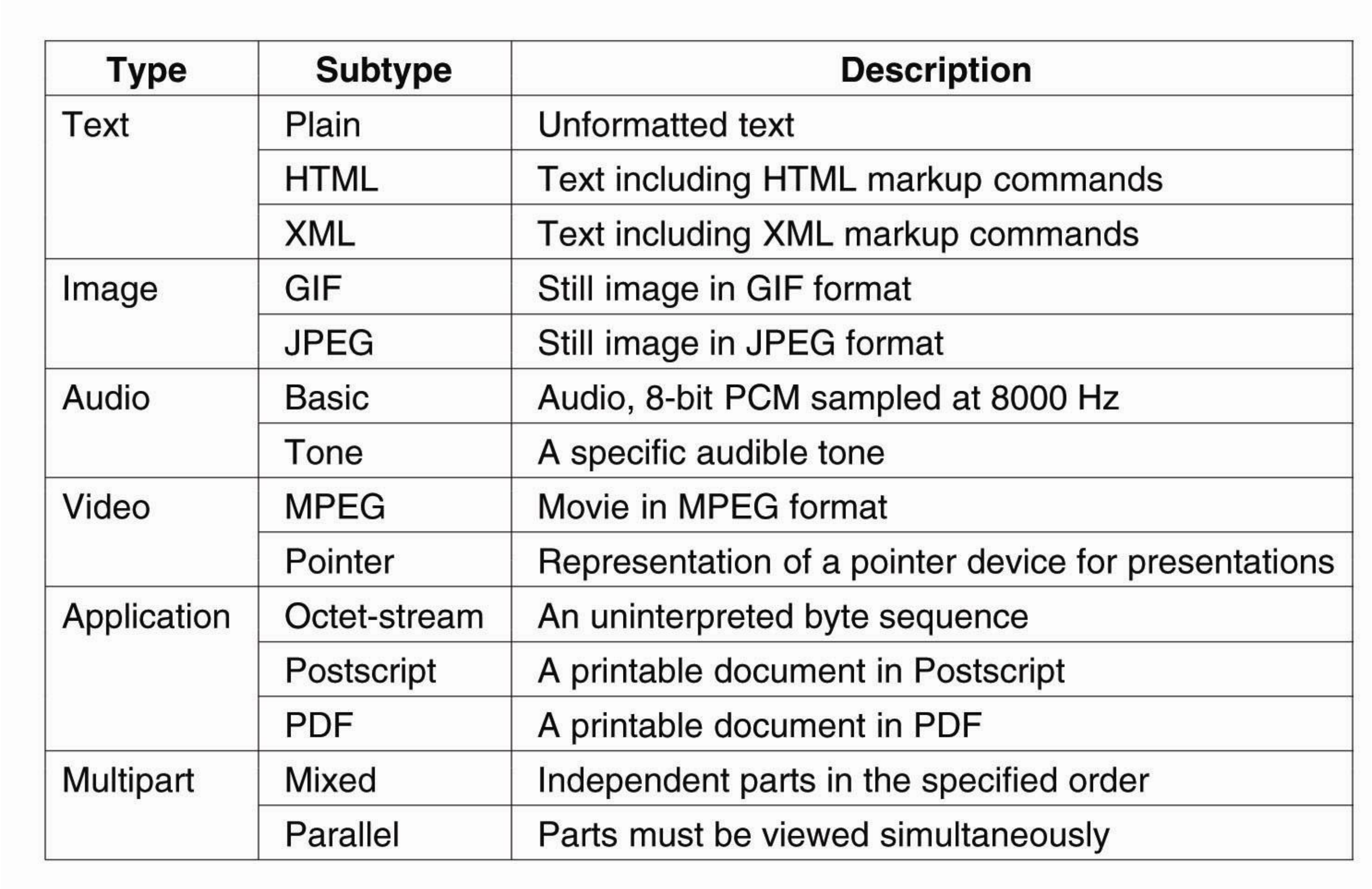
Telephone numbers

- 1-800-WIDGETS
- 1-415-765-4321

(b)

Web Documents

[Tanenbaum]



Type	Subtype	Description
Text	Plain	Unformatted text
	HTML	Text including HTML markup commands
	XML	Text including XML markup commands
Image	GIF	Still image in GIF format
	JPEG	Still image in JPEG format
Audio	Basic	Audio, 8-bit PCM sampled at 8000 Hz
	Tone	A specific audible tone
Video	MPEG	Movie in MPEG format
	Pointer	Representation of a pointer device for presentations
Application	Octet-stream	An uninterpreted byte sequence
	Postscript	A printable document in Postscript
	PDF	A printable document in PDF
Multipart	Mixed	Independent parts in the specified order
	Parallel	Parts must be viewed simultaneously

Six top-level MIME types and some common subtypes.

XML …

- **XML is a text-based markup language that is fast becoming the standard for data interchange on the Web.**
- **XML has syntax analogous to HTML.**
- **Unlike HTML, XML tags tell you what the data *means*, rather than how to display it.**
- **Example:**

```
 <message>
          <to>you@yourAddress.com</to> <from>me@myAddress.com</from>
          <subject>XML Is Really Cool</subject>
          <text> How many ways is XML cool? Let me count the ways... </text>
  </message>
```

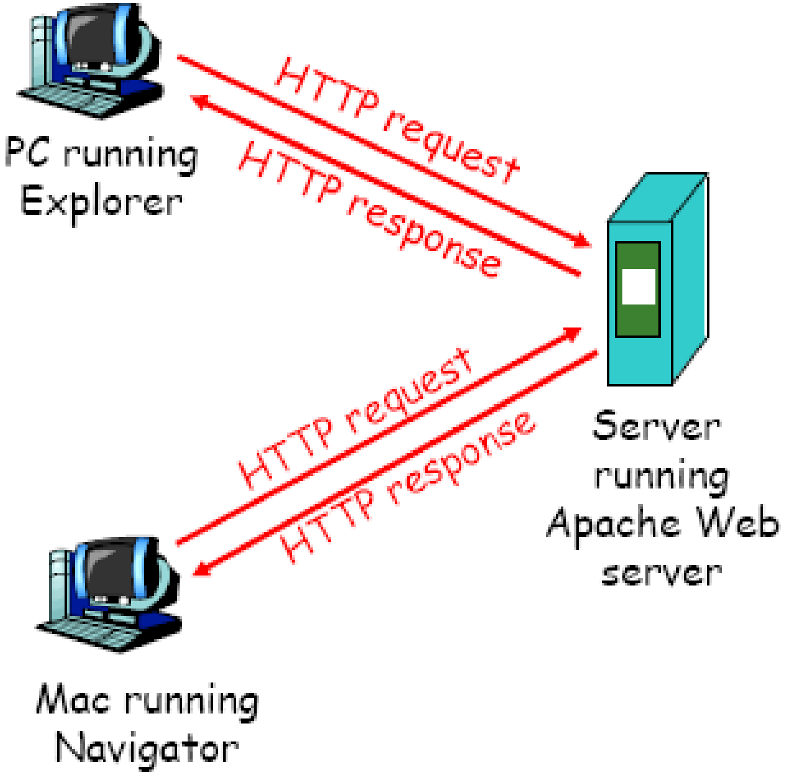
Taken from: http://java.sun.com/xml/docs/tutorial/overview/1_xml.html#intro

HTTP Overview

[this and following EECS122, Abhay Parekh]

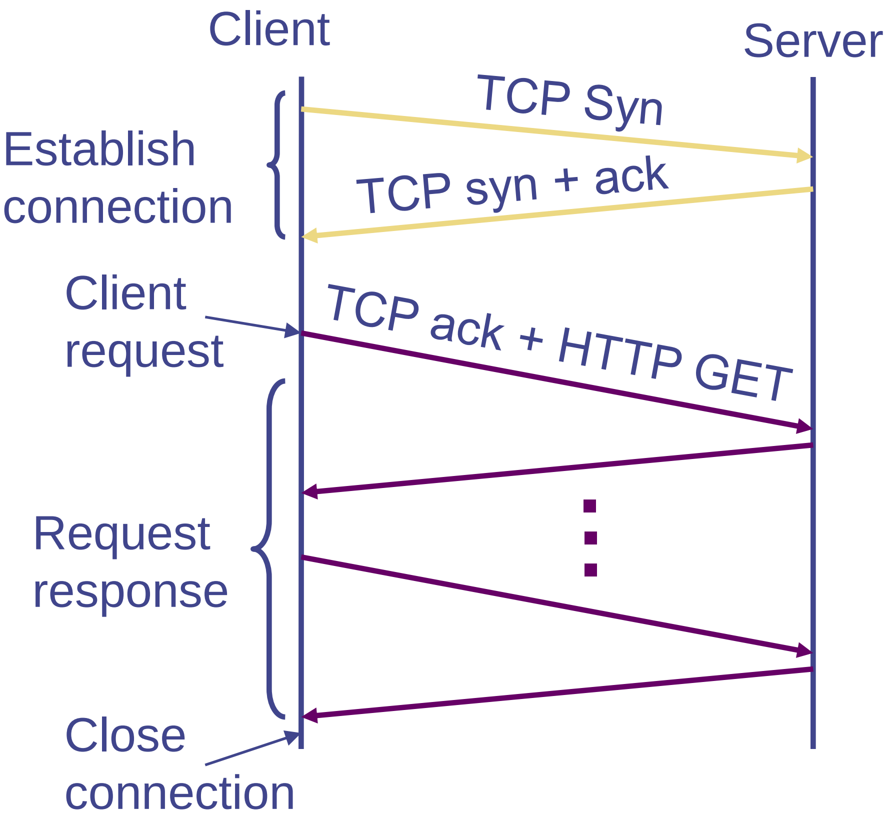
HTTP: hypertext transfer protocol

- Web's application layer protocol
- client/server model
  - *client:* browser that requests, receives, "displays" Web objects
  - *server:* Web server sends objects in response to requests
- HTTP 1.0: RFC 1945
- HTTP 1.1: RFC 2068

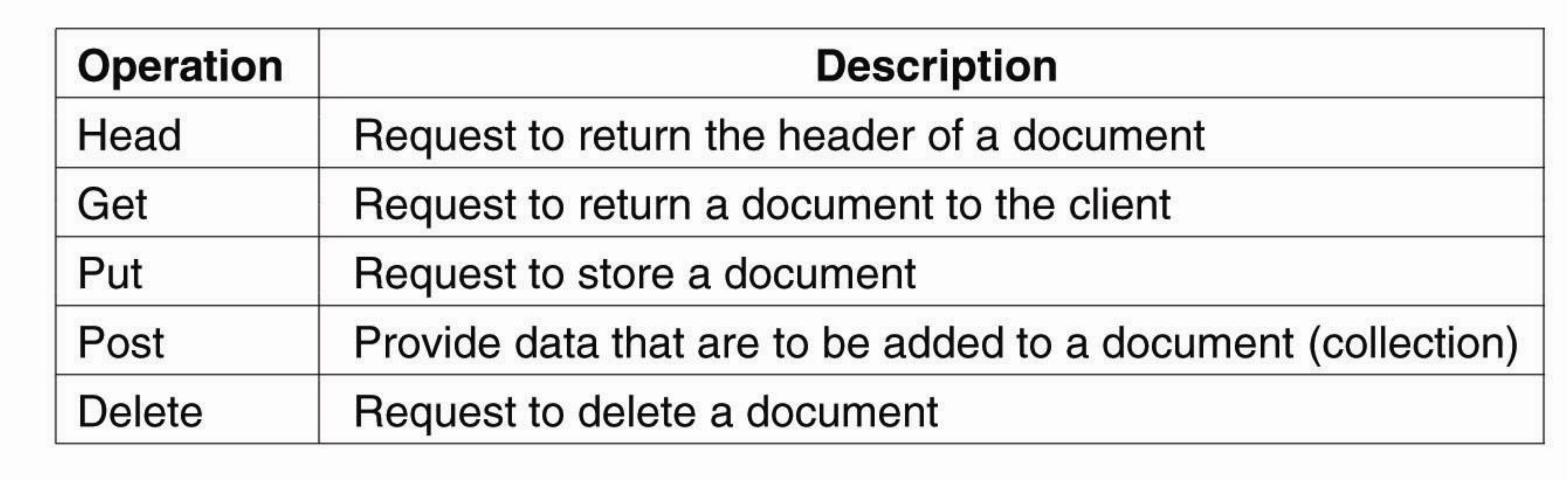


Note: The server is stateless, i.e. each request is self contained!

Big Picture - HTTP



Most important HTTP Methods



Operation	Description
Head	Request to return the header of a document
Get	Request to return a document to the client
Put	Request to store a document
Post	Provide data that are to be added to a document (collection)
Delete	Request to delete a document

How does it work – Example

[Kurose and Ross]

- http://www.mylife.org/mypictures.htm
- After finding out the IP address of the host…(DNS!)

1. http client initiates a TCP connection on :80
2. Client sends the get request via socket established in 1
3. Server sends the html file, which is encapsulated in its response
4. http server tells tcp to terminate connection
5. http client receives the file and the browser parses it…contains ten jpeg images
6. Client repeats steps 1-4

Persistency of connection usage

- A web page typically contains many objects
  - E.g. Images
  - Each object must be requested with a separate http “Get” command
  - Non Persistent Connection:
    - Different __TCP__ connection for each object request.
    - HTTP 1.0
  - Persistent Connection
    - Reuse the same TCP connection for each object request
    - HTTP 1.1

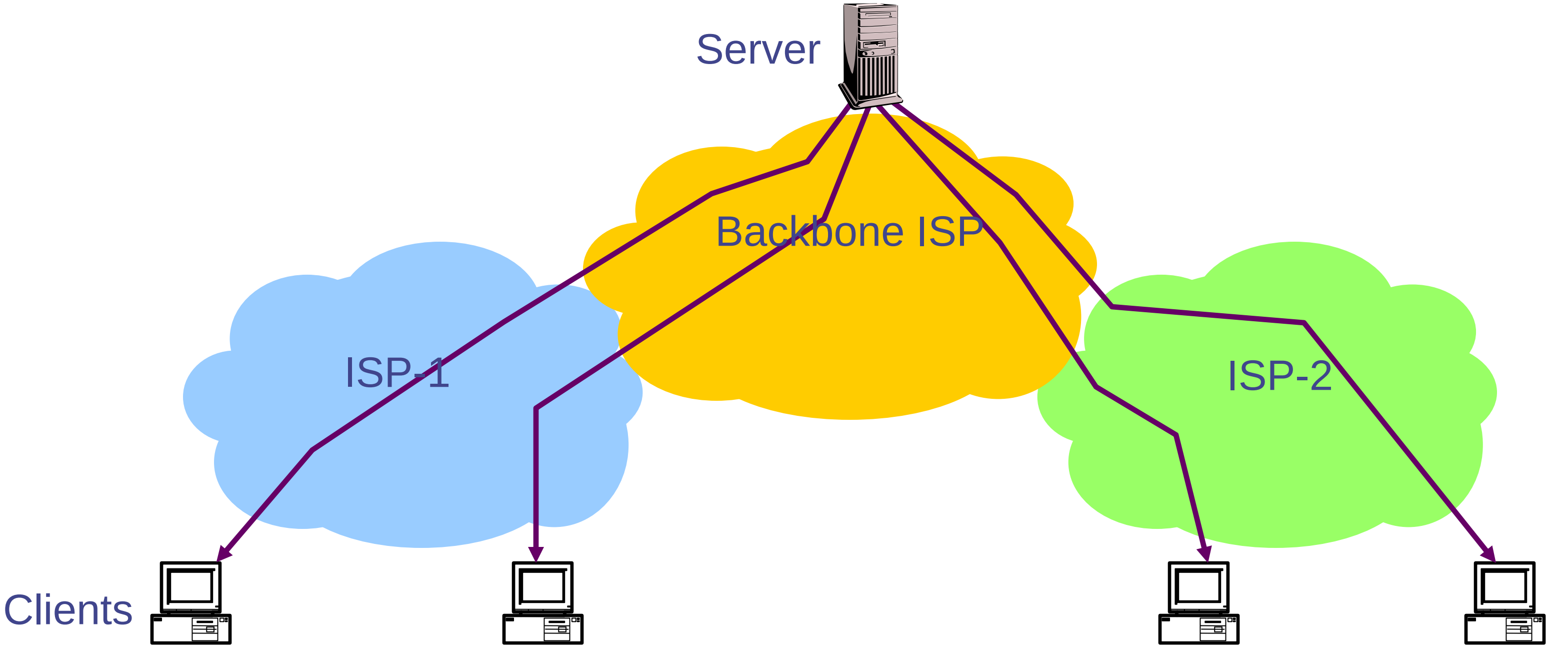
Performance and Reliability...

- Problem: You are a web content provider
  - How do you handle millions of web clients?
  - How do you ensure that all clients experience good performance?
  - How do you maintain availability in the presence of server and network failures?

- Solutions:
  - Add more servers at different locations → If you are CNN this might work!
  - Caching
  - Content Distribution Networks (Replication)

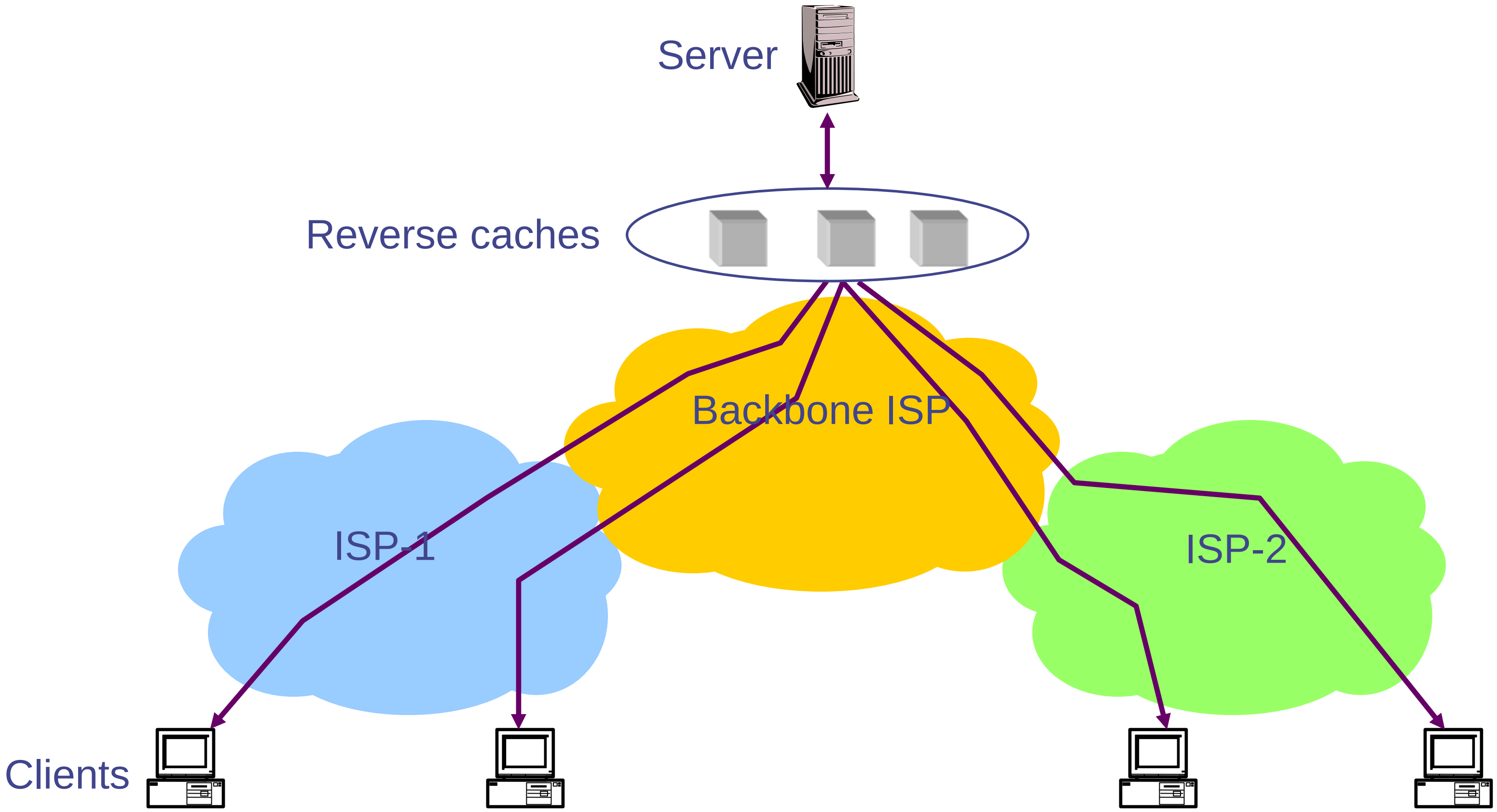
“Base-line”

- Many clients transfer same information
  - Generate unnecessary server and network load
  - Clients experience unnecessary latency



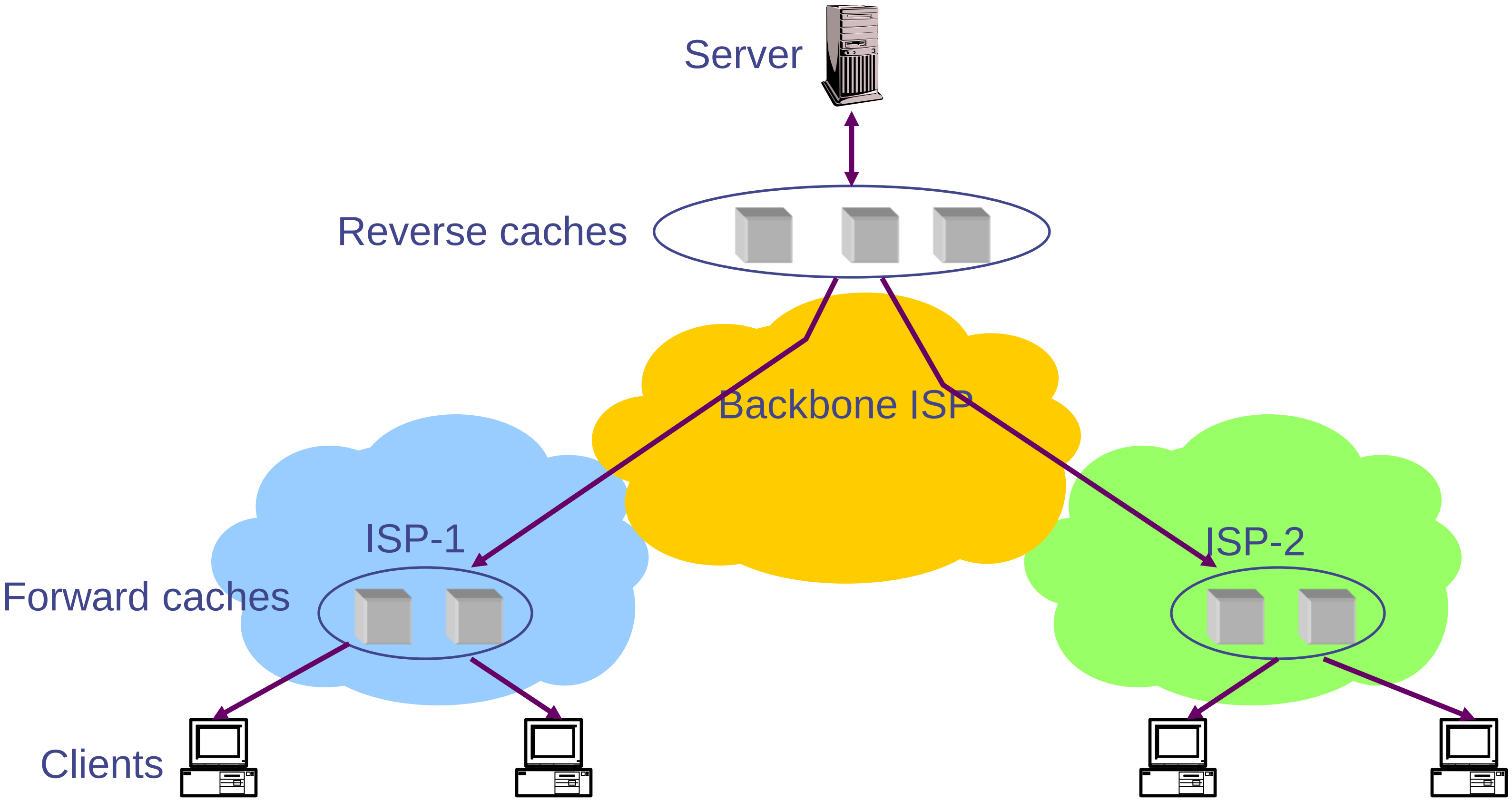
Reverse Caches

- Cache documents close to server → decrease server load
- Typically done by content providers



Forward Proxies

- Cache documents close to clients → reduce network traffic and decrease latency
- Typically done by ISPs or corporate LANs



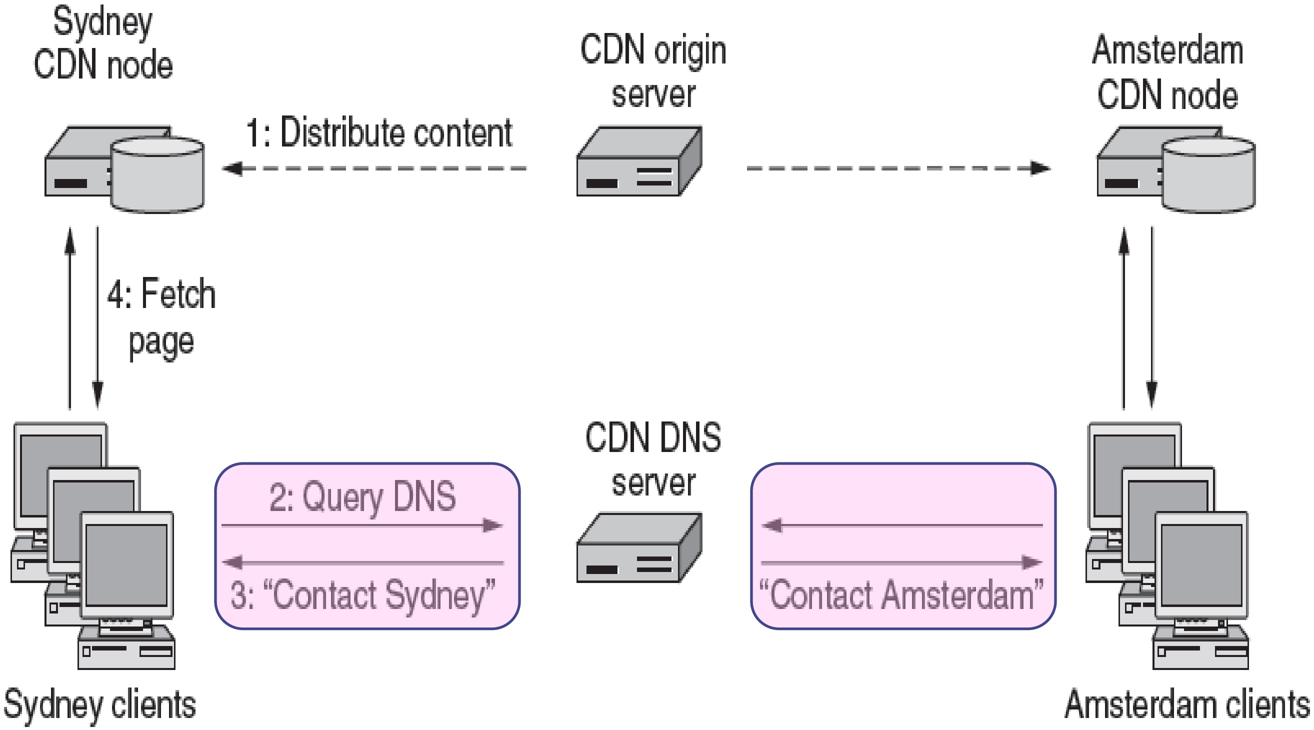
Client Side - HTTP/1.0 Caching Support

- Exploit locality of reference
- A modifier to the GET request:
  - If-modified-since – return a “not modified” response if resource was not modified since specified time
- A response header:
  - Expires – specify to the client for how long it is safe to cache the resource
- A request directive:
  - No-cache – ignore all caches and get resource directly from server
- These features can be best taken advantage of with HTTP proxies
  - Locality of reference increases if many clients share a proxy

Content Delivery Network

[Tanenbaum, Wetheral]

- DNS resolution of site gives different answers to clients
  - Tell each client the site is the nearest replica (map client IP)



Example: www.akamai.com

DNS records don’t have to store the real IP address of the host.

- One name can map onto multiple addresses
  - Example: www.yahoo.com can be mapped to multiple machines
  - Example: www.akamai.com

    From Berkeley 64.164.108.148

    From the NY Area 63.240.15.146

    From the UK *194.82.174.224*

  What do we gain: shorter delay, load distribution