CSCI 4621 Intro to CyberSecurity

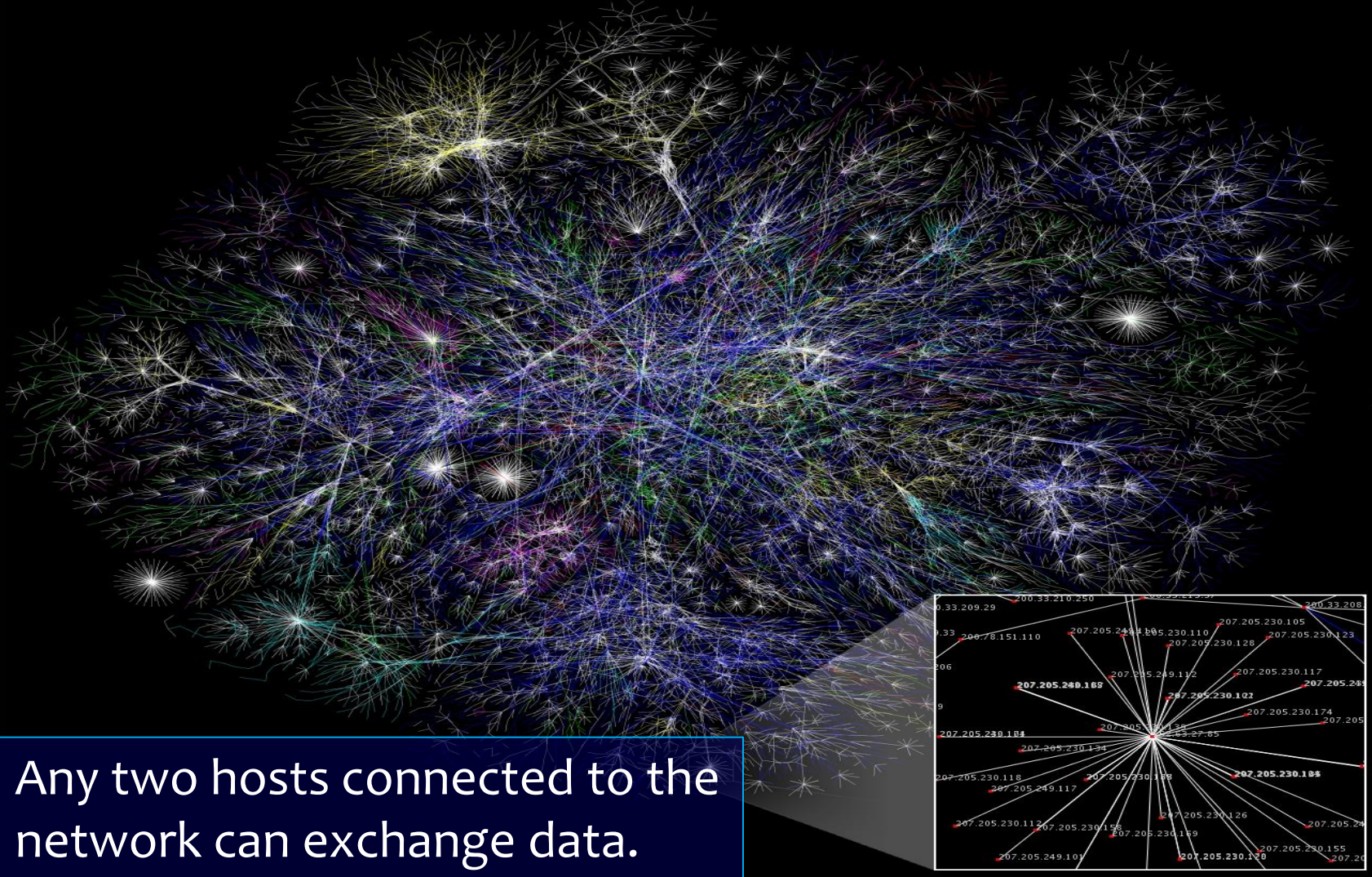
Review – Networking

Vassil Roussev

vassil@cs.uno.edu

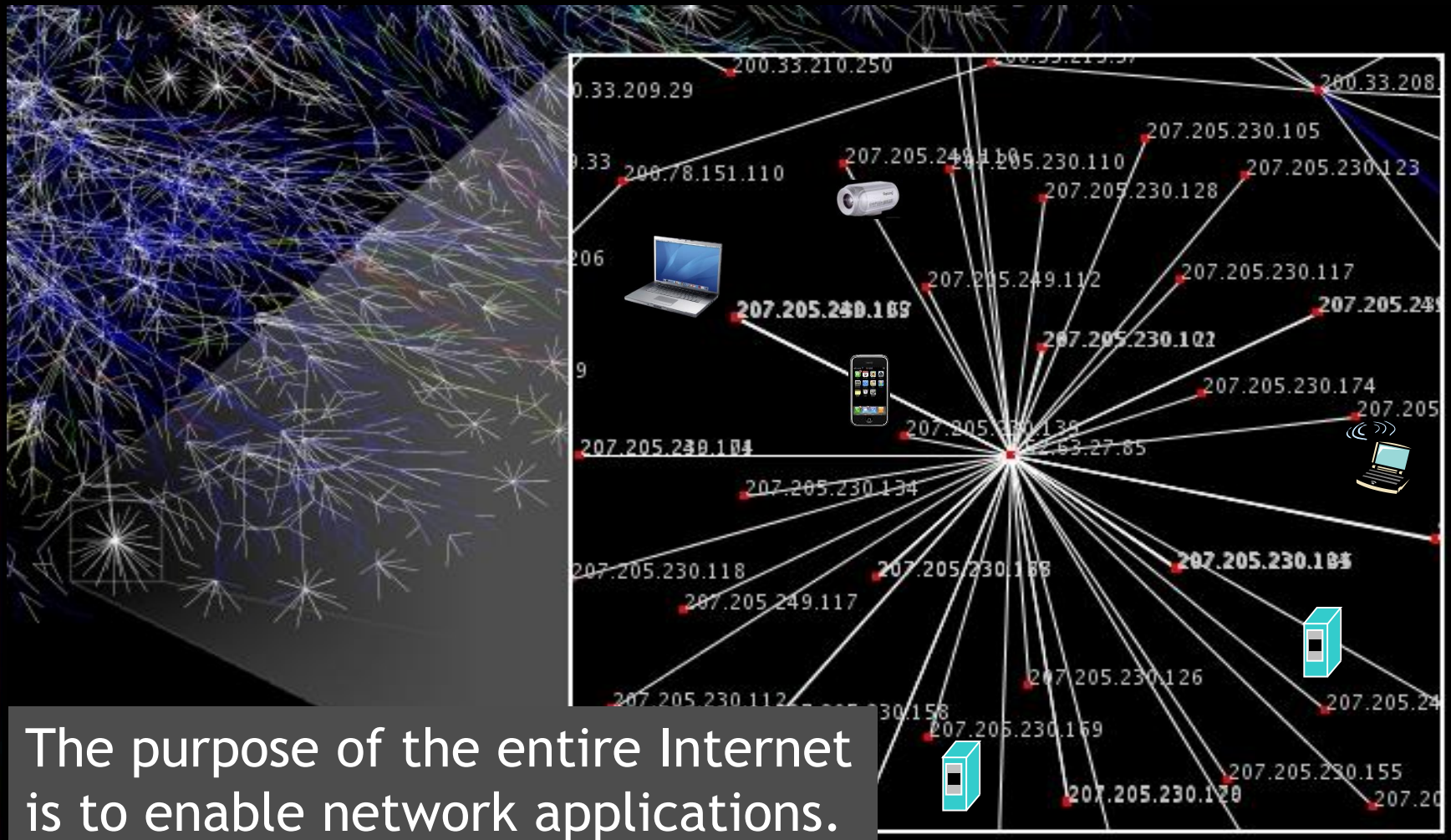
the Internet

The Internet is a global data communication infrastructure (network).



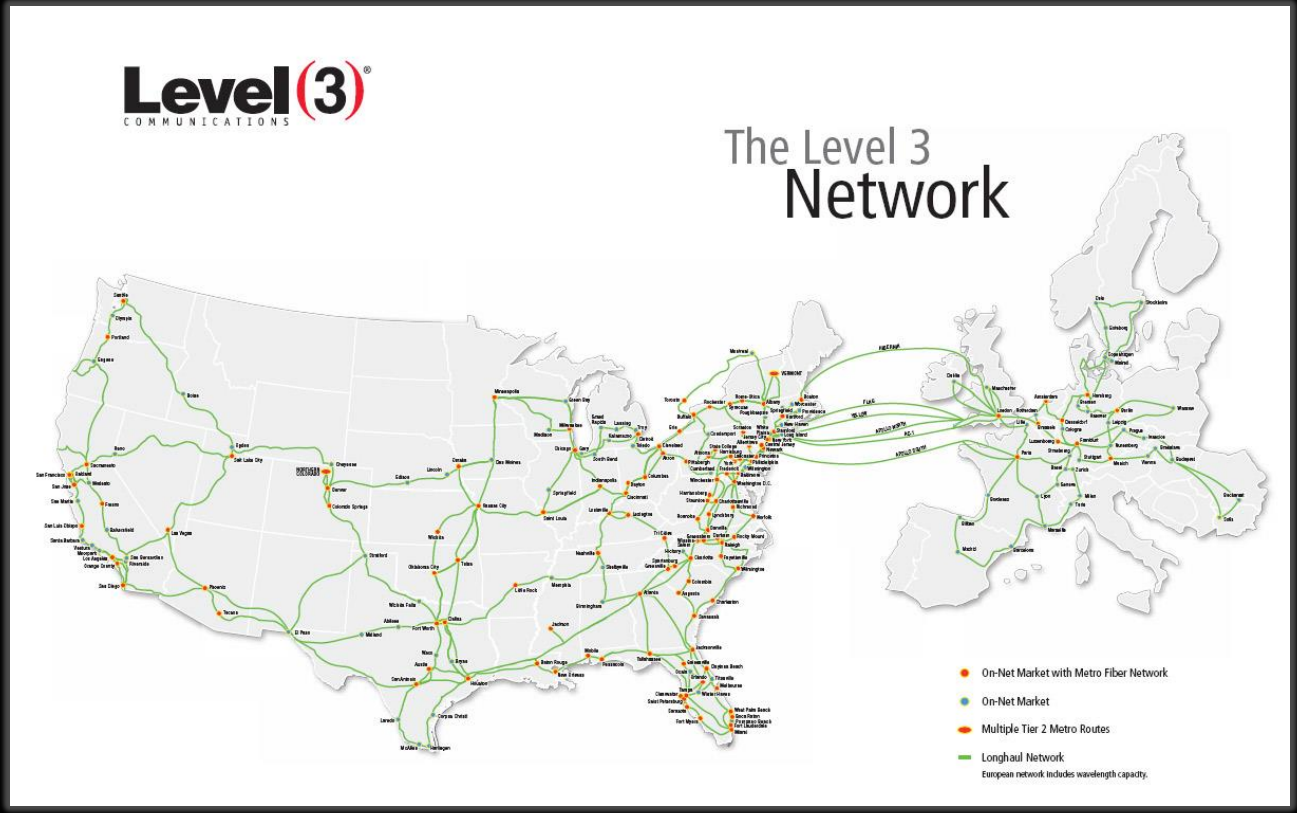
Any two hosts connected to the network can exchange data.

The network edge consists of all the hosts that serve end users via network applications.



The purpose of the entire Internet is to enable network applications.

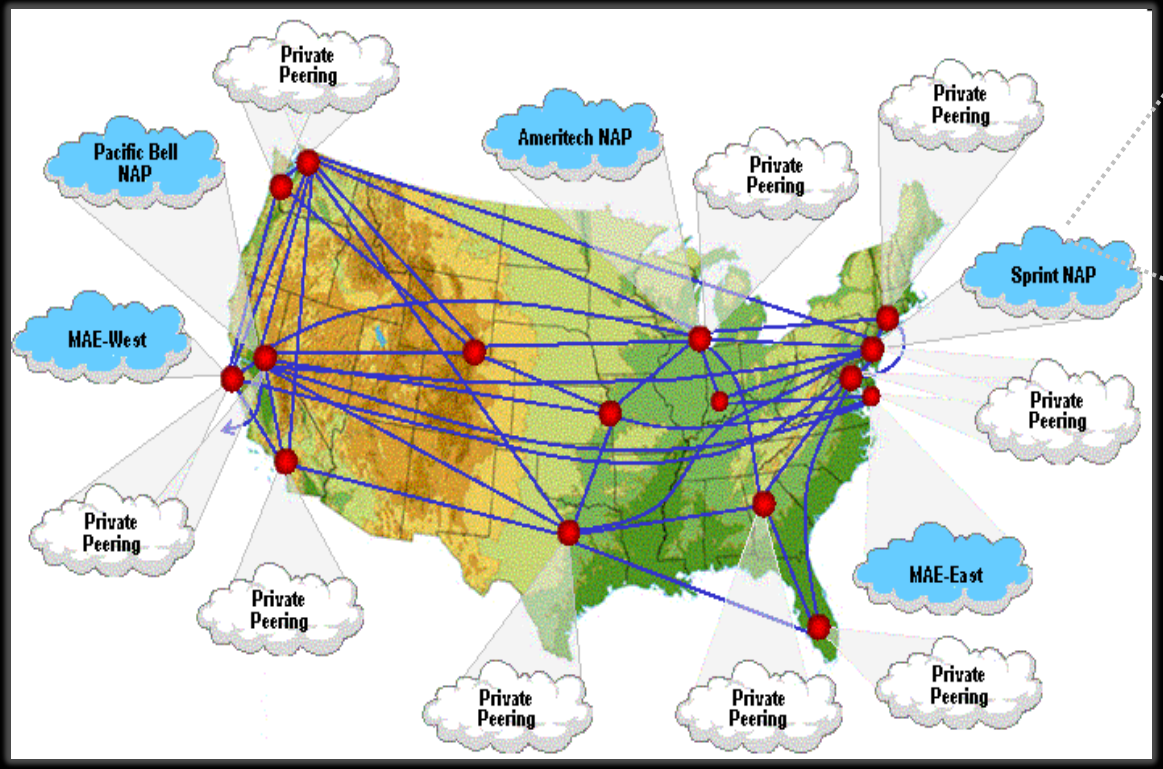
The network core consists of routers and communication links that connect all hosts.



Routers are specialized computers that manage network traffic.

Internet communication links are organized in a loose hierarchy similar to transport networks.

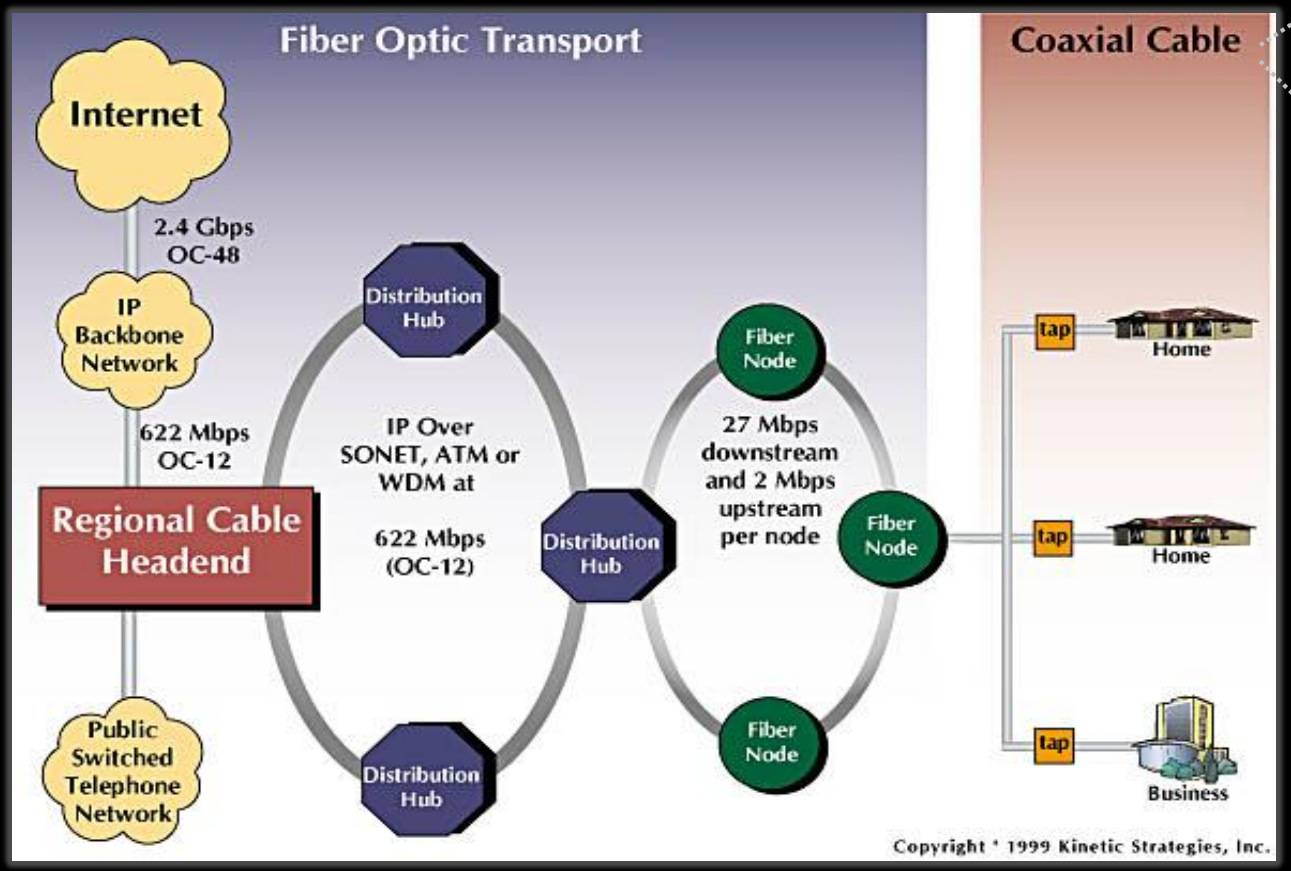
The 'Net has decentralized structure with different parts owned/managed autonomously.



LINX (London)

Backbone Network Service Providers (NSPs) own long-haul networks and exchange data at *Internet eXchange Points* (IXPs), a.k.a. *Network Access Points* (*NAP*) in the US.

Regional/local ISPs deliver local traffic & aggregate traffic for long-haul delivery.



New installations use fiber-optic.

Long-haul fiber links are expensive; traffic aggregation ensures proper utilization.

Regional ISPs increasingly use long-haul links for internal traffic delivery.

Beyond a certain (traffic) threshold, it is cheaper to build out the network then to pay other ISPs for delivery.

Observing routes: `traceroute`

```
Tracing route to internet-service.ucc.uno.edu [137.30.1.92]
over a maximum of 30 hops:

  1     9 ms     9 ms     8 ms  10.128.32.1
  2    11 ms    10 ms    10 ms  68.11.12.25
  3    11 ms     8 ms    12 ms  mctydsrc01-gew0304.rd.no.cox.net [68.11.14.9]
  4    13 ms     9 ms    13 ms  mctybbrc01-pos0101.rd.no.cox.net [68.1.0.64]
  5    10 ms    10 ms     9 ms  mctybbrc02-pos0100.rd.no.cox.net [68.1.0.63]
  6    21 ms    19 ms    20 ms  lkhnbbrc02-pos0102.rd.at.cox.net [68.1.0.10]
  7    22 ms    21 ms    20 ms  lkhnbbrc01-pos0100.rd.at.cox.net [68.1.0.2]
  8    29 ms    21 ms    21 ms  so-1-2-0-0.gar2.Atlanta1.Level3.net [65.59.222.5]
  9    21 ms    21 ms    21 ms  so-0-3-0.bbr2.Atlanta1.Level3.net [209.247.11.225]
 10    35 ms    36 ms    34 ms  so-0-0-0.bbr1.Washington1.Level3.net [64.159.1.2]
 11    35 ms    37 ms    34 ms  so-6-0-0.edge1.Washington1.Level3.net [209.244.11
 12    36 ms    35 ms    36 ms  qwest-level3-oc48.Washington1.Level3.net [209.244
 13    35 ms    35 ms    36 ms  205.171.251.33
 14    35 ms    35 ms    37 ms  dca-core-02.inet.qwest.net [205.171.8.221]
 15    36 ms    37 ms    36 ms  atl-core-02.inet.qwest.net [205.171.8.153]
 16    37 ms    46 ms    37 ms  atl-edge-05.inet.qwest.net [205.171.21.54]
 17    51 ms    52 ms    51 ms  65.112.33.250
 18    56 ms    56 ms    60 ms  nor2-isp2.atm-vcc.La.Net [162.75.7.78]
 19     *        *        *     Request timed out.
```

← Harahan, LA 70123

New Orleans

Atlanta

Washington, DC

Atlanta

← Louisiana

← UNO campus, 70148

Q: Why do we connect to the Internet?

A: To use different network applications

twitter

BitTorrent

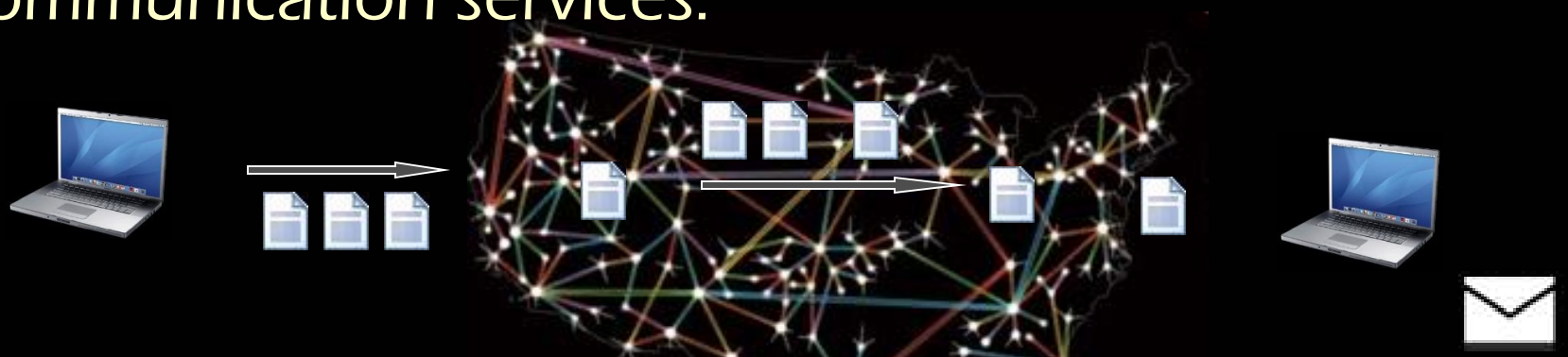
facebook

PANDORA

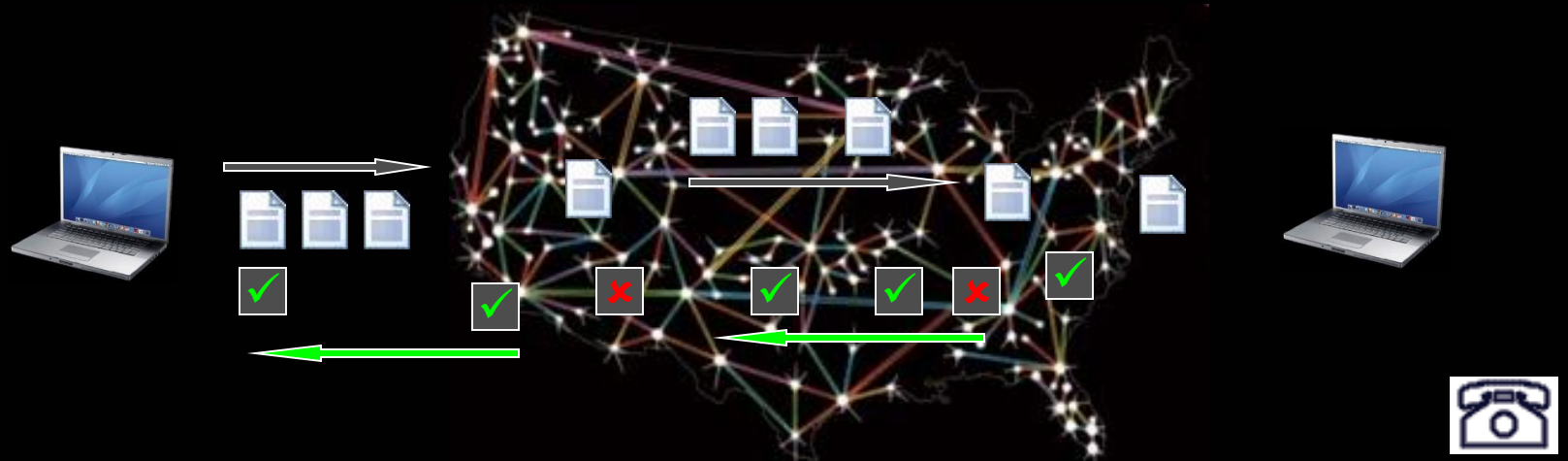
skype

All network applications rely on standard 'Net communication services.

The 'Net infrastructure offers two standard communication services:



Connectionless service (**UDP**) provides no feedback/delivery guarantees.

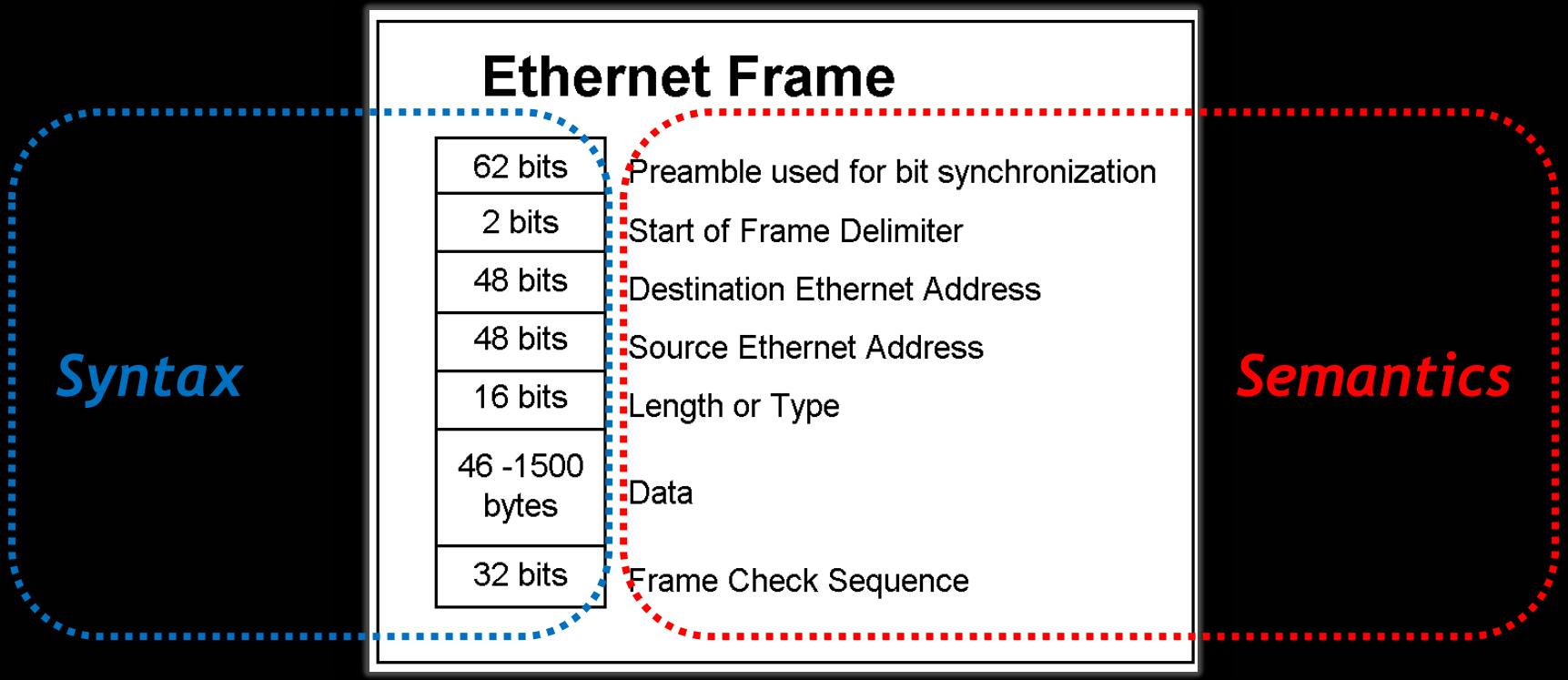


Connection-oriented service (***TCP***) provides acknowledgements and guarantees eventual delivery.

A **communication protocol** *defines the* **syntax**, **semantics**, *order, and timing of data transmissions.*

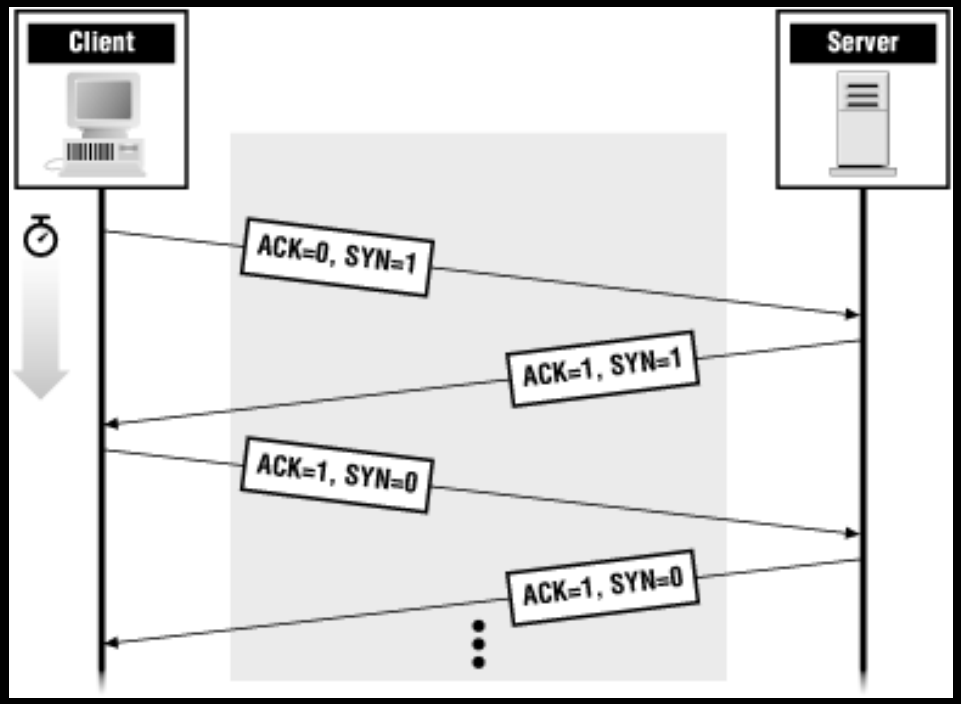
Syntax: the ***format*** of data transmitted.

Semantics: the ***meaning*** of the data transmitted.



A **communication protocol** defines the syntax, semantics, **order**, and timing of data transmissions.

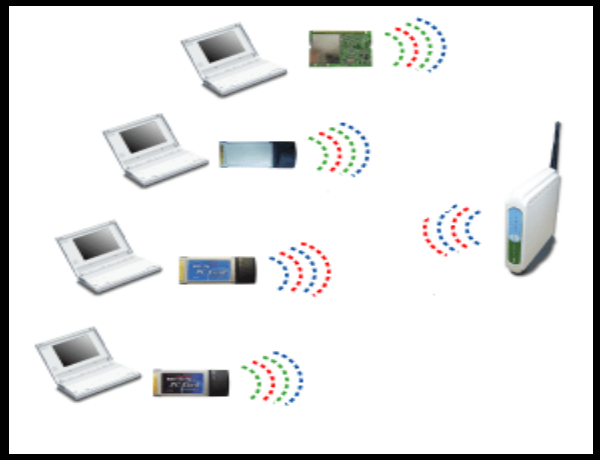
Order: the ***conversation rules*** of the transmissions.



TCP handshake

A **communication protocol** defines the syntax, semantics, order, and **timing** of data transmissions.

Timing: the ***beginning*** and ***rate*** of the transmissions.

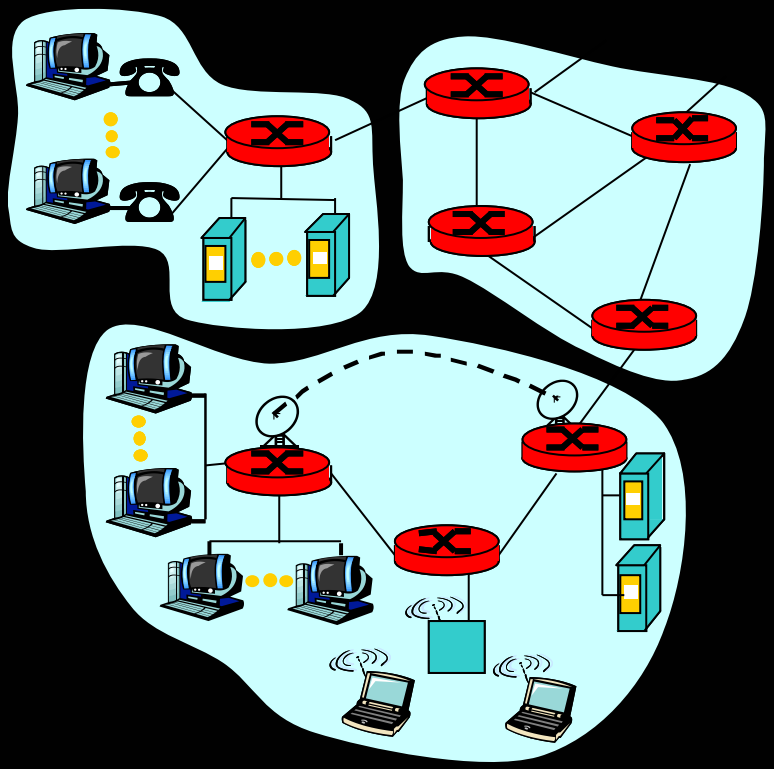


MAC protocols coordinate transmissions; necessary to avoid interference.

Ethernet ports negotiate transmission rates (10,100,1000M).

The Internet Protocols Food Chain

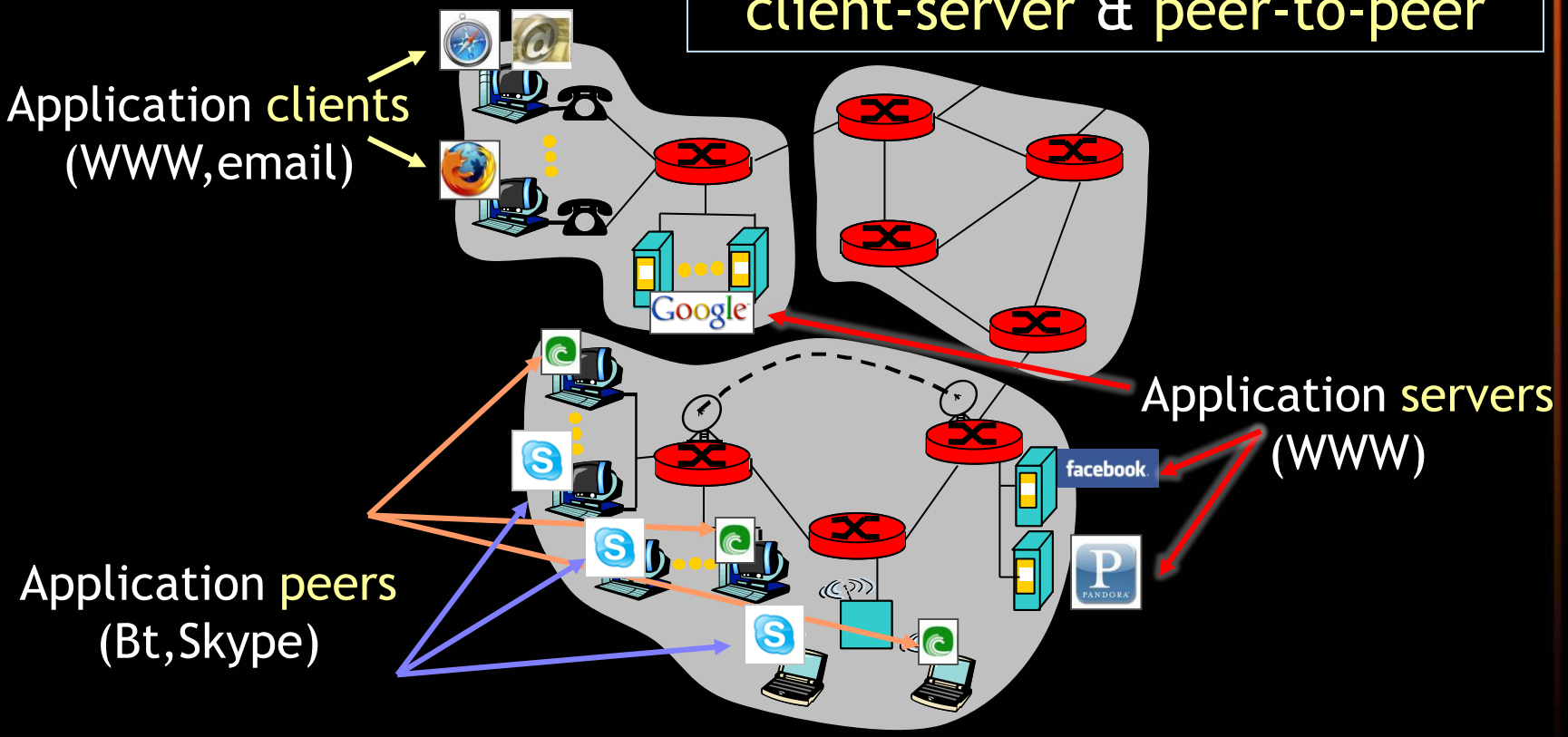
- Network interface-to-transmission media
  - *Media access protocols*
- Router-to-router
  - *Routing protocol*
- Operating system-to-server
  - *Name resolution protocols*
- Transport protocol-to-TP
  - *Reliable transmission*
  - *Congestion & flow control*
- Application-level
  - *File transfer, email, …*



Application Layer

Network applications consist of two, or more, processes communicating over the network.

Two application architectures:
client-server & peer-to-peer



Each application defines its own protocol for exchanging messages; multiple implementations are possible.

Try this at a shell prompt:

```
>telnet google.com 80 > google.html↵
GET / HTTP/1.0↵
↵
Connection closed by foreign host.
>_
```

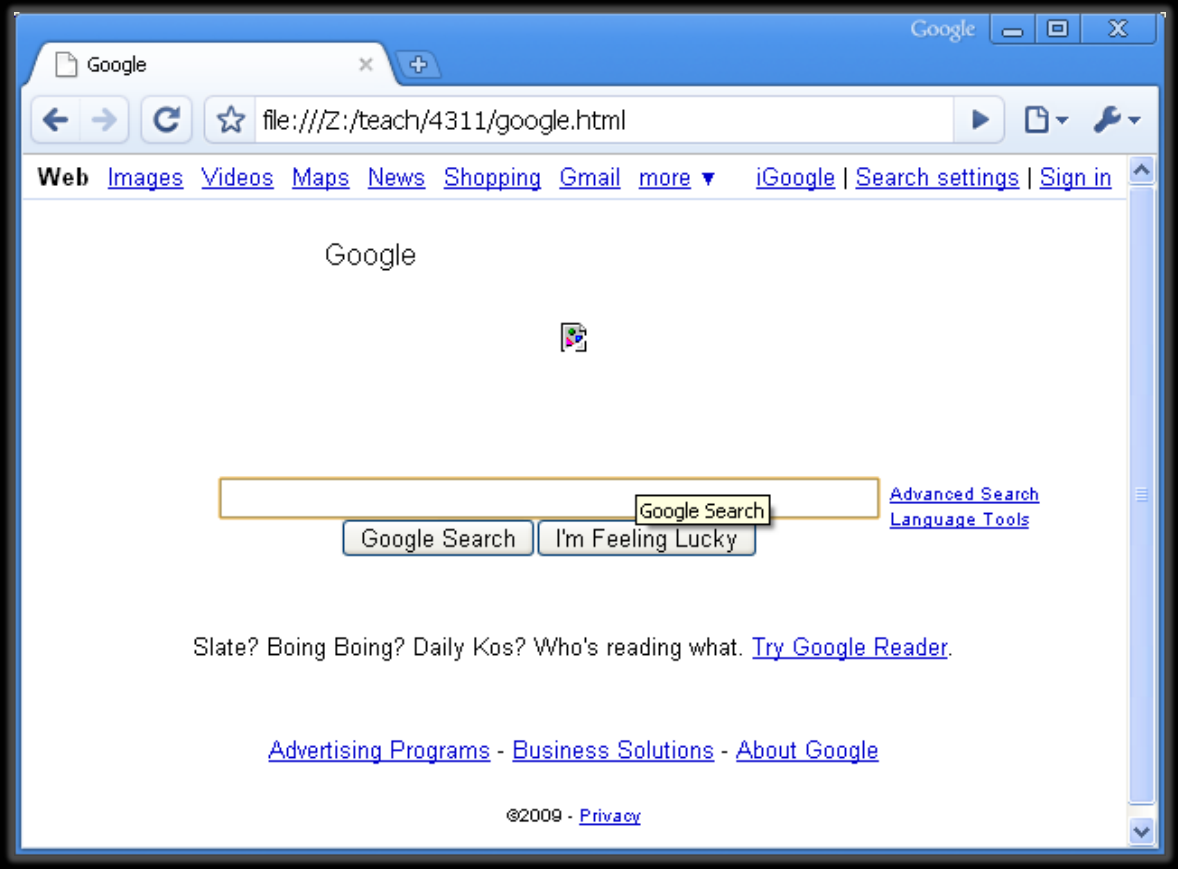
google.html should look like this:

```
Trying 74.125.127.100...
Connected to google.com.
Escape character is '^]'.
HTTP/1.0 200 OK
Date: Wed, 02 Sep 2009 21:39:00 GMT
Expires: -1
Cache-Control: private, max-age=0
Content-Type: text/html; charset=ISO-8859-1
Set-Cookie: PREF=ID=28e4088ee4e9eb4b:TM=1251927540:LM=12
MT; path=/; domain=.google.com
Set-Cookie: NID=26=gkijYBJQ-o_lcBItp1xNqy4enRCW9k4aNIA7d
H12-IRASV81x3YLylObeCsjkDli; expires=Thu, 04-Mar-2010 21
Server: gws

<!doctype html><html><head><meta http-equiv="content-typ
<script>window.google={kEI:"9OWeSt2NL4LStgPp6cjmDQ",kEXP
WeSt2NL4LStgPp6cjmDQ"},kHL:"en"};
```

Remove these lines, save, and open in browser

Browser view



Congratulations—you just 'spoke' HTTP!

The World Wide Web (WWW)

Communication protocol

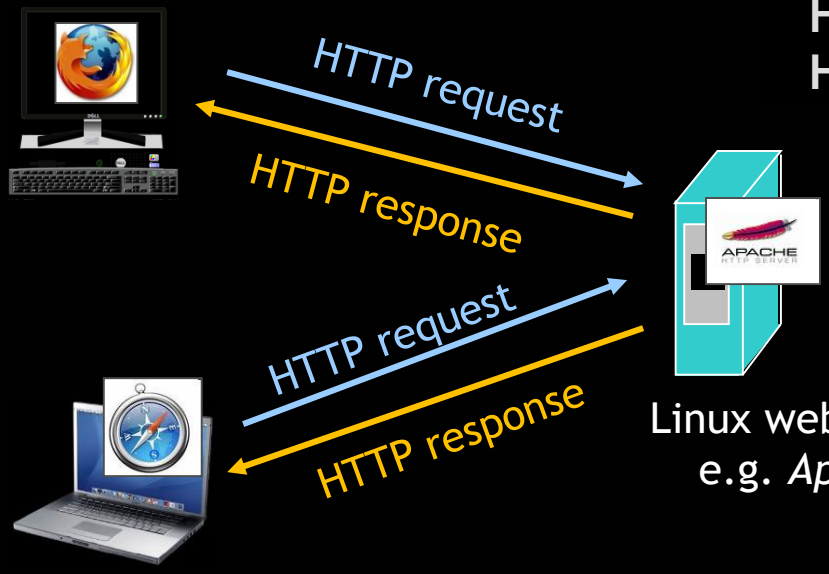
Network application

Document naming

Document format

URL + HTTP + HTML = WWW

HTTP 1.0: RFC 1945
HTTP 1.1: RFC 2068



Uniform Resource Locator (URL)

Protocol:
`http,ftp,…`

Server port
(80 for WWW)

`http://www.cs.uno.edu:80/index.html`

Server domain name/
IP address

Local object name
(opaque)

Interpretation:

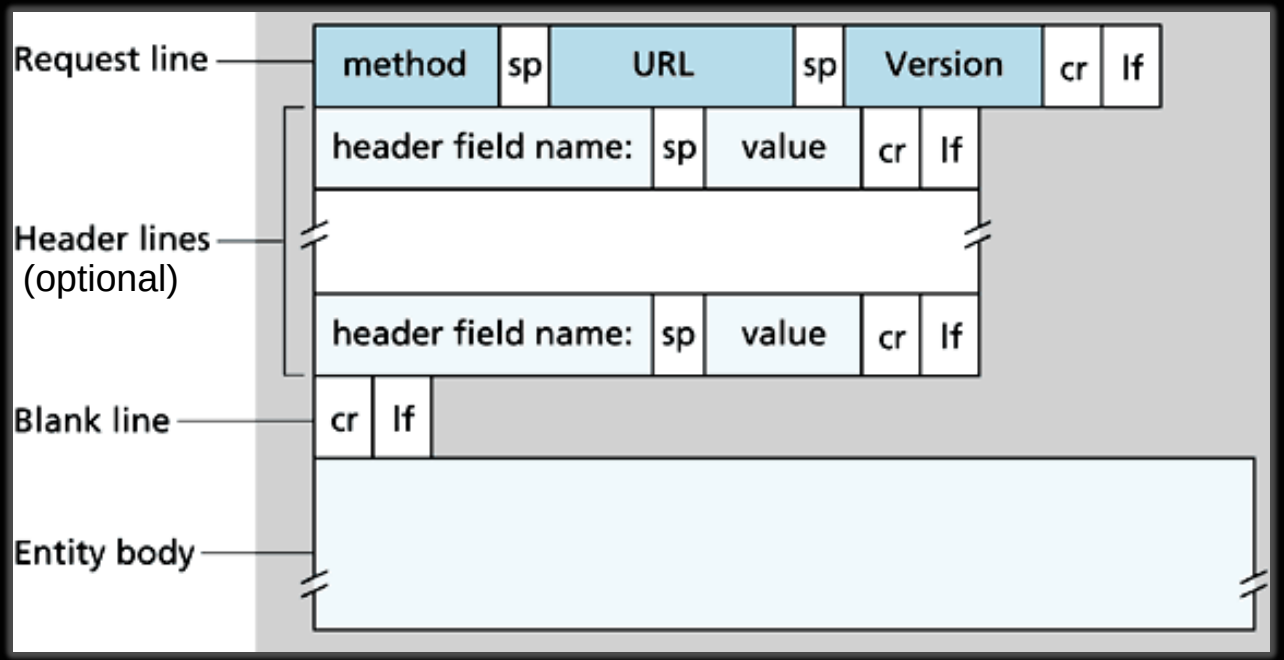
- Open TCP socket to `www.cs.uno.edu:80`
- Request `/index.html` using the `http` protocol

HTTP request format

```
GET, POST, HEAD  (1.0)
PUT, DELETE      (1.1)
```

```
/index.html
```

```
HTTP/1.0 or HTTP/1.1
```

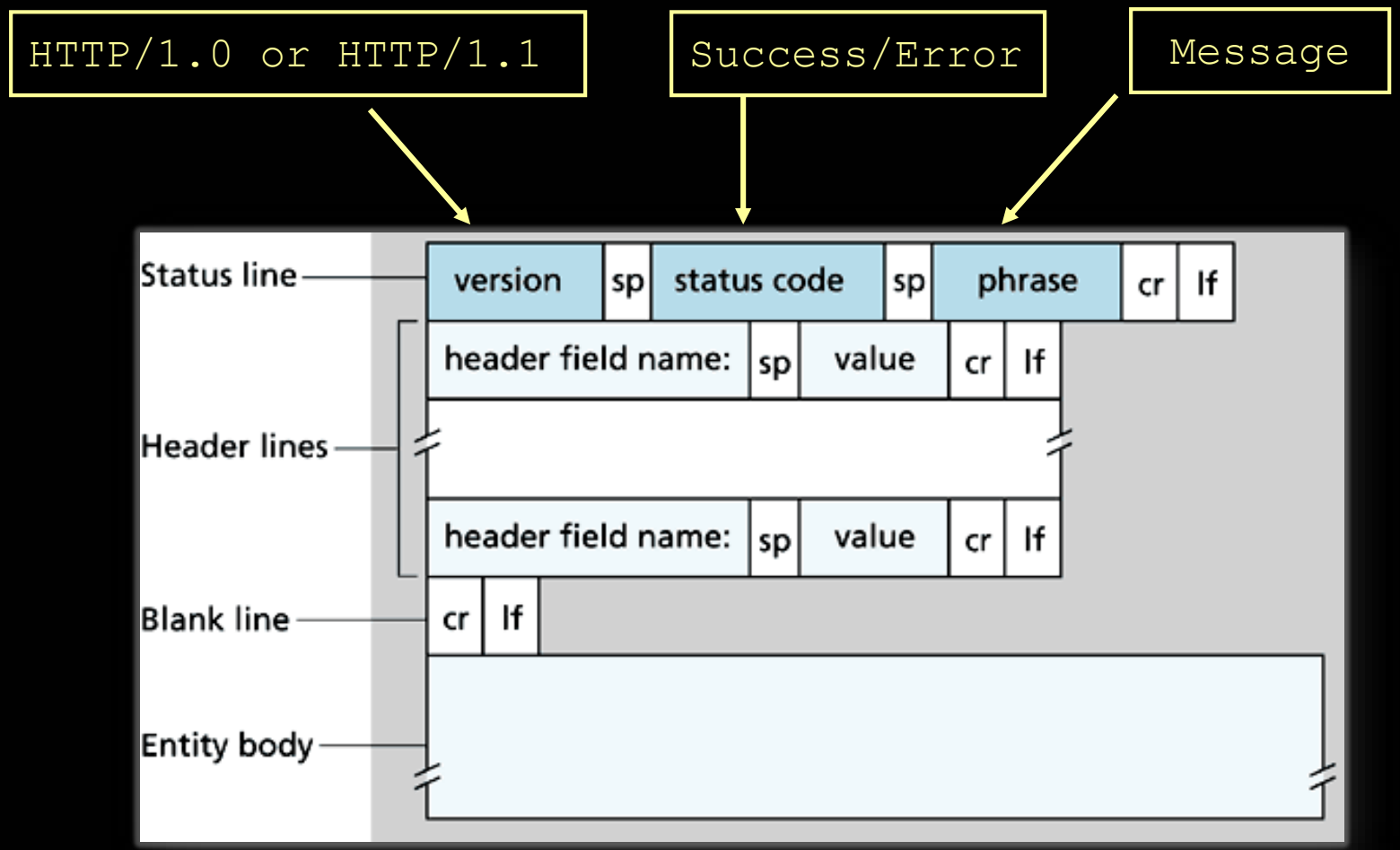


HTTP request examples

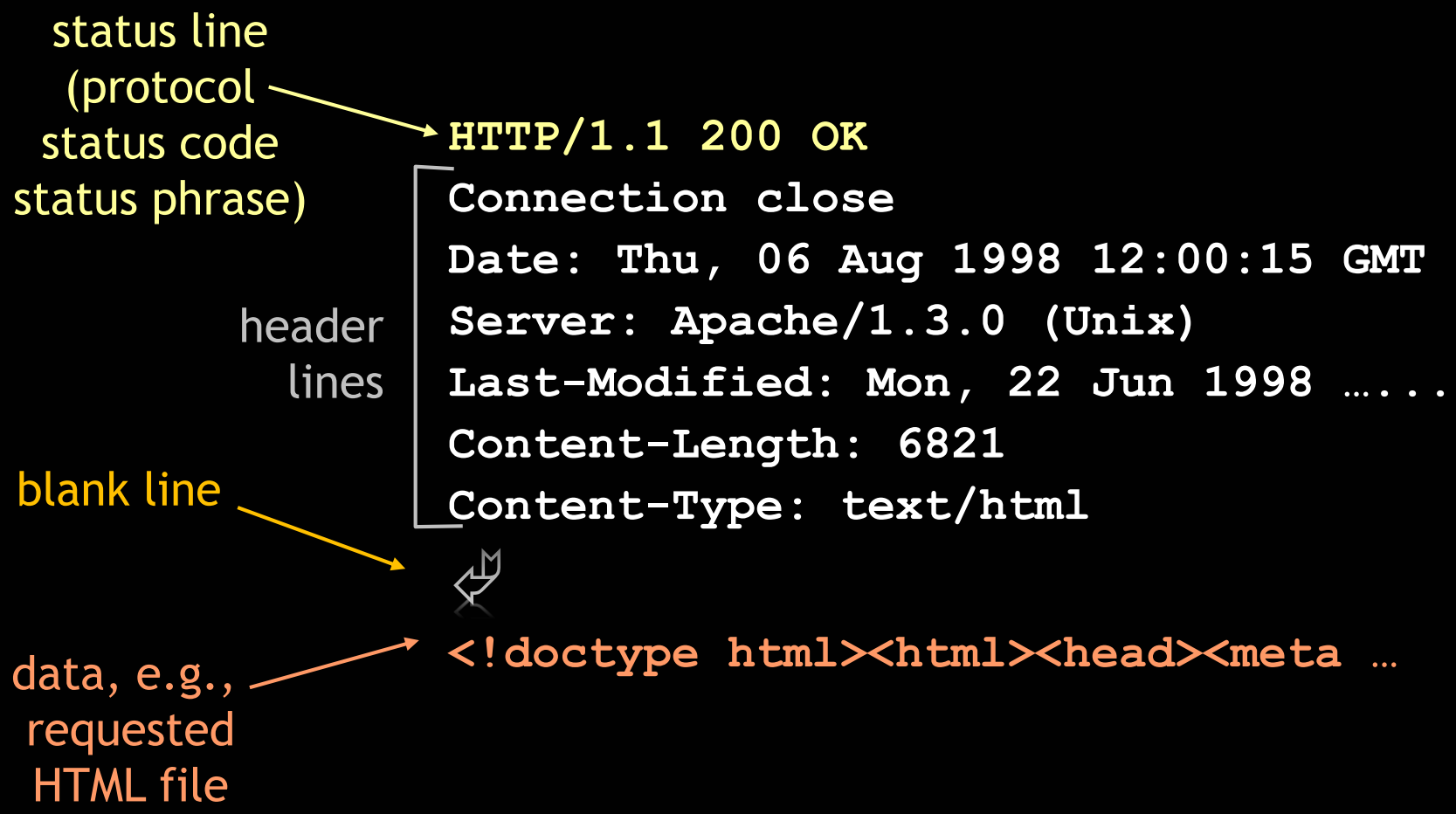
```
GET /~vassil HTTP/1.0
Connection: Keep-Alive
User-Agent: Mozilla/4.74 [en] (WinNT; U)
Host: dilbert.cs.uno.edu:8080
Accept: image/gif, image/x-xbitmap, image/jpeg,
        image/pjpeg, image/png, */*
Accept-Encoding: gzip
Accept-Language: en
Accept-Charset: iso-8859-1,*,utf-8
Cookie: SITESERVER=ID=8a064b7855a043146e45991174a3d970
```

```
GET /~vassil HTTP/1.1
Accept: image/gif, image/x-xbitmap, image/jpeg,
        image/pjpeg, application/msword,
        application/vnd.ms-excel,
        application/vnd.ms-powerpoint, */*
Accept-Language: en-us
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 5.5; Windows NT 4.0)
Host: dilbert.cs.uno.edu:8080
Connection: Keep-Alive
```

HTTP response format



HTTP response example



HTTP response types (RFC 2616)

2xx Success

request succeeded, requested object later in this message

3xx Redirection

requested object moved, new location specified later in this message

4xx Bad client request

request message not understood by server

5xx Server error

an error occurred while processing the request

Common response status codes

200 OK

- request succeeded, requested object later in this message

301 Moved Permanently

- requested object moved, new location specified later in this message (Location:)

400 Bad Request

- request message not understood by server

404 Not Found

- requested document not found on this server

505 HTTP Version Not Supported

DNS: Domain Name System

DNS: Domain Name System

People identifiers:

- Name/SSN/Passport #/…
- Why so many?

Internet hosts, routers:

- IP address (32 bit) - used for addressing datagrams
- Issues:
  - Host IP address may change
  - IP address not suitable for human consumption
- Solution:
  - Human-readable "name": e.g., `www.cs.uno.edu`

New problem:

Internet-wide mapping b/w names and IP addresses!

Solution: *DNS*

Distributed database
implemented in a hierarchy of many *name servers*

Application-layer protocol

Host, routers, name servers to communicate to *resolve* names (i.e. perform name ➔ address translation)

Essential Internet function, implemented as an application-layer protocol

DNS: Root Name Servers

http://root-servers.org

DNS Recursive Query

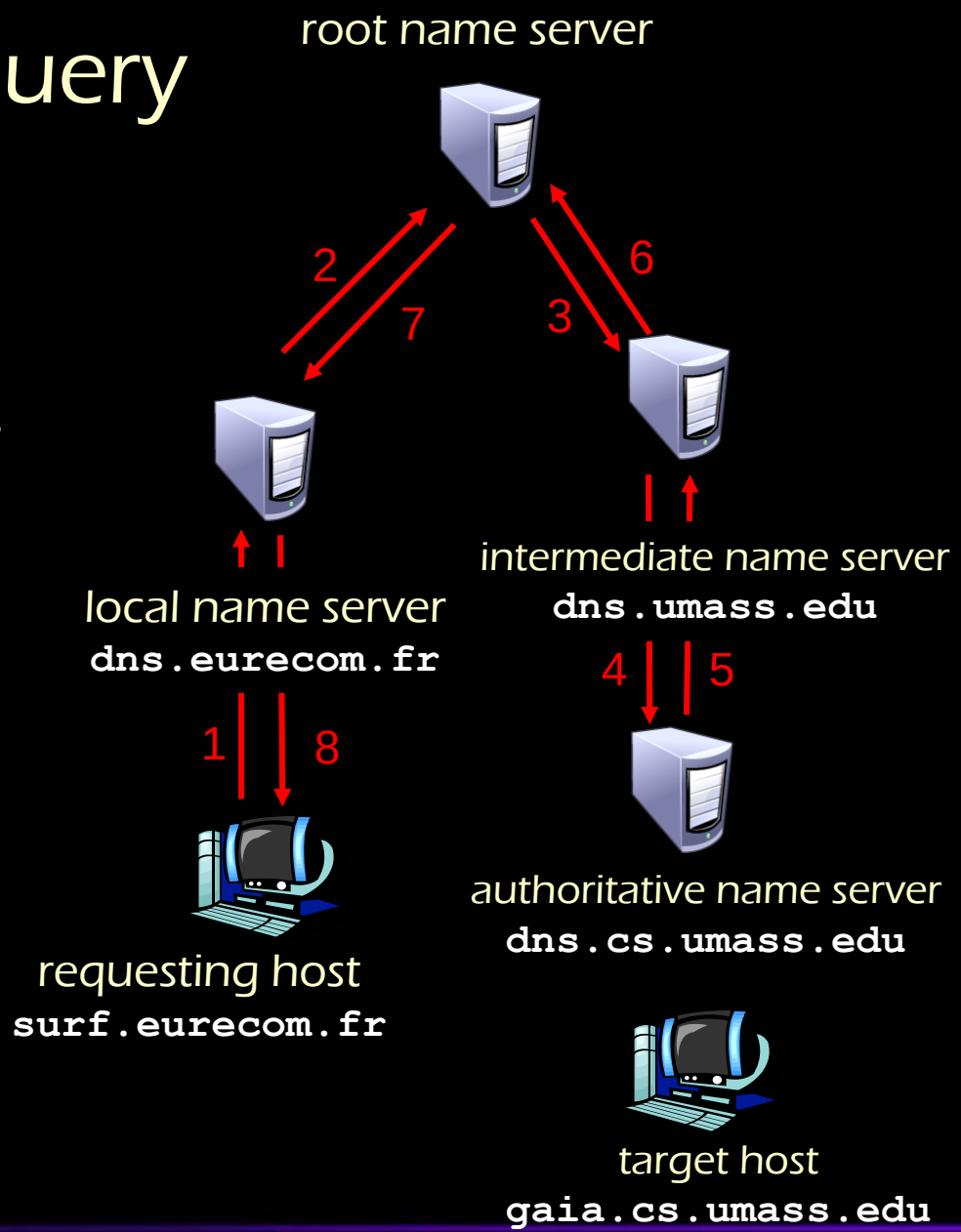
Root name server:

May not know authoritative name server

But does know *intermediate name server*

In a recursive query, each serve in the chain performs inquiries on behalf of requestors.

NB: Root servers never perform recursive queries on behalf of clients.



DNS Iterated Query

Recursive query:

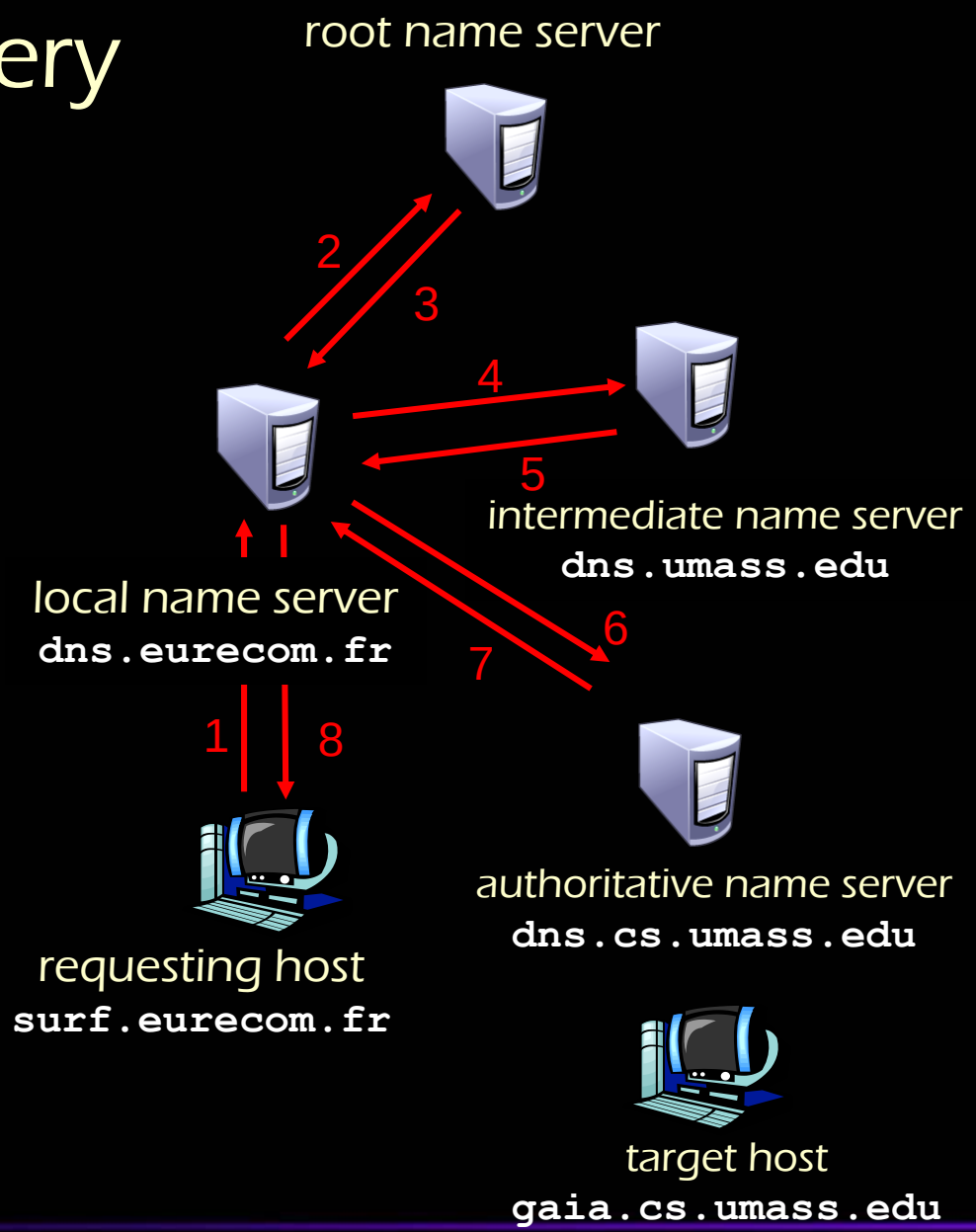
Puts burden of name resolution on contacted name server

Heavy load?

Iterated query:

Contacted server replies with name of server to contact

"I don't know this name, but ask this server"



DNS: Caching and Updating Records

Once (any) name server learns mapping, it *caches* mapping:

- cache entries timeout (disappear) after some time
- a.k.a. **soft state**

Update/notify mechanisms under design by IETF

- RFC 2136
- http://www.ietf.org/html.charters/dnsind-charter.html

DNS Records

DNS: Distributed DB storing resource records (RR)

RR format: **(name, value, type, ttl)**

Type=A

- **name** is hostname
- **value** is IP address

Type=NS

- **name** *is domain (e.g. foo.com)*
- **value** *is IP address of authoritative name server for this domain*

Type=CNAME

- **name** is alias name for some "cannonical" (the real) name

  `www.ibm.com` is really

  `servereast.backup2.ibm.com`
- **value** is canonical name

Type=MX

- **value** is name of mailserver associated with **name**