W12 (2)

Computer Networks II

Application Layer

Basic Architectures, DNS

Amitangshu Pal
Computer Science and Engineering
IIT Kanpur

Creating a Network App

- Application programs run on end systems
  - Network-core devices do not run user applications
  - Communicate over network
    - Web server software communicates with browser software

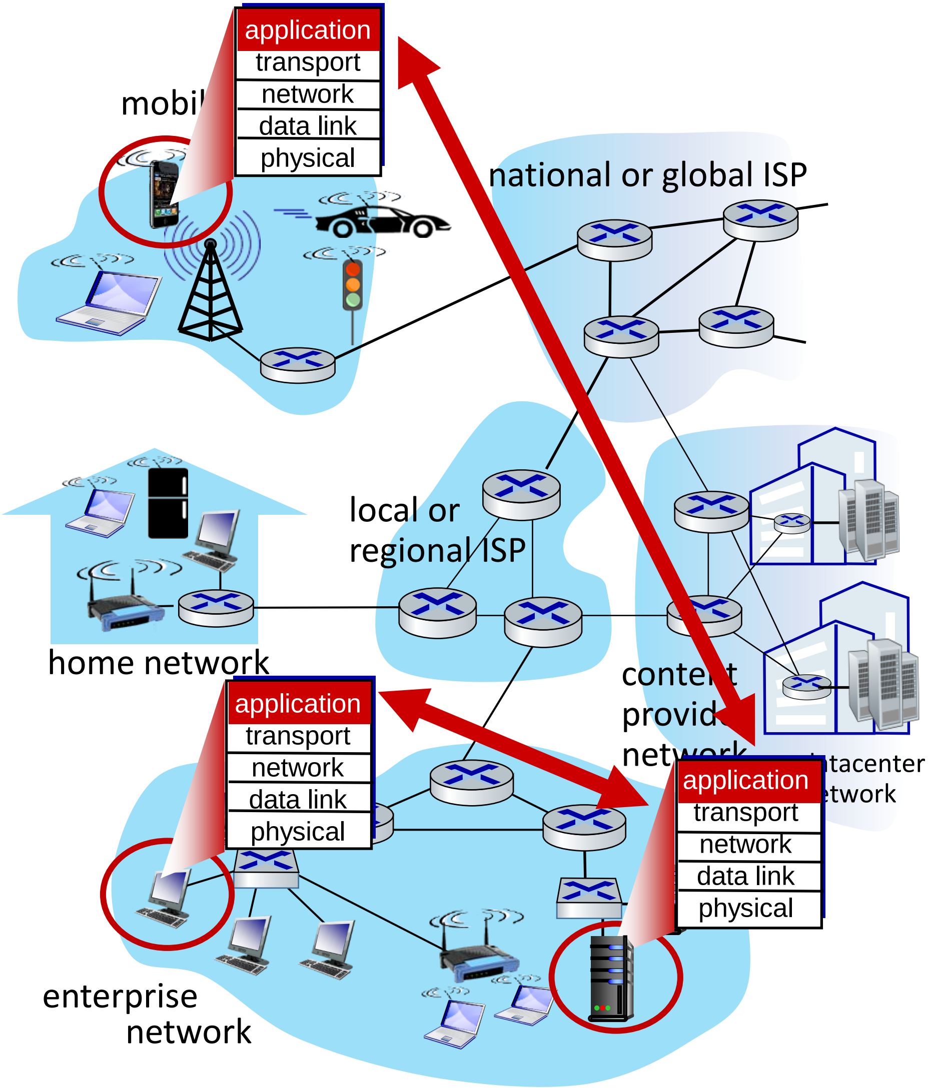
User Processes

Operating Systems

Application

Transport

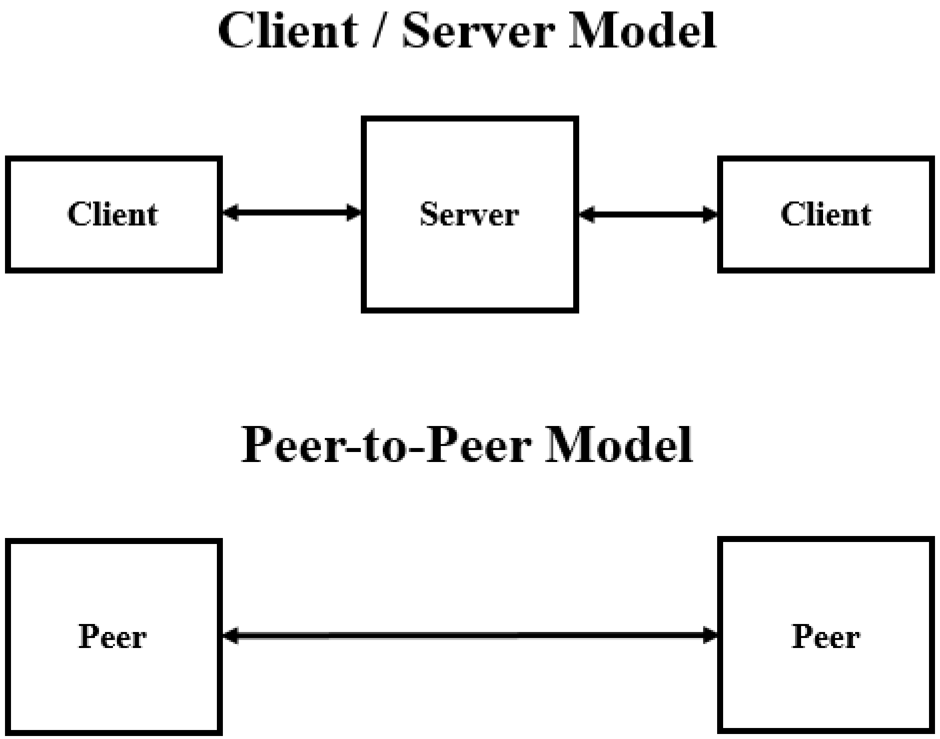
Network



Application architectures

Possible structure of applications:

- Client-server
- Peer-to-peer (P2P)



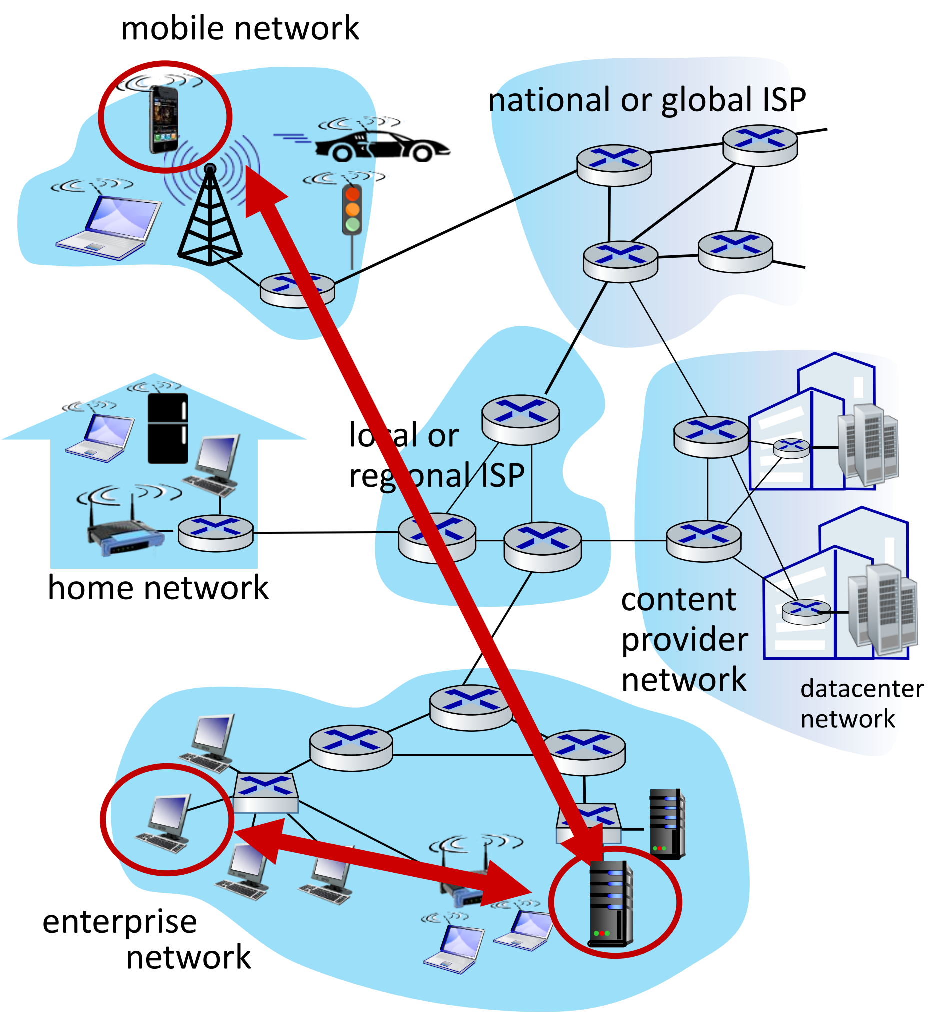
Src: https://commons.wikimedia.org/wiki/File:Client-server_Vs_peer-to-peer_-_en.png

Client-server architecture

Server:
- Always-on host
- Permanent IP address
- Data centers for scaling

Clients:
- Communicate with server
- May be intermittently connected
- May have dynamic IP addresses
- Do not communicate directly with each other

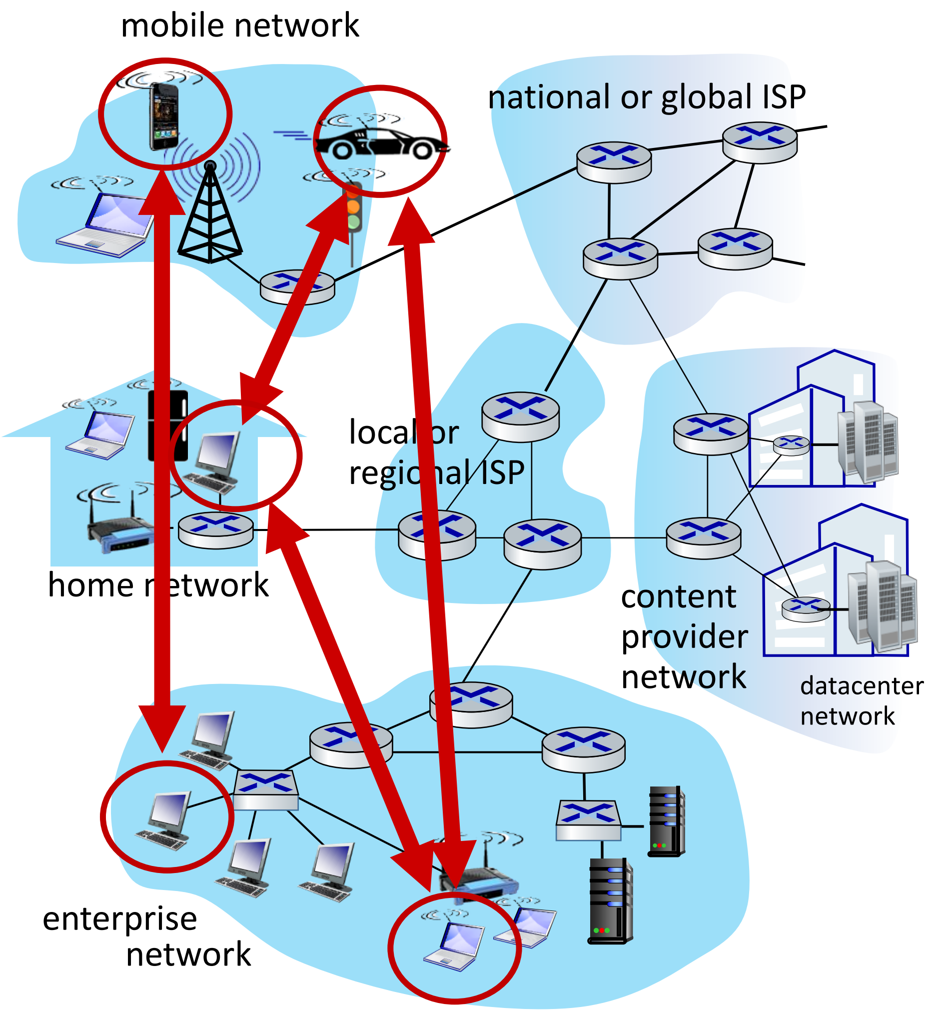
Examples: HTTP, FTP, IMAP



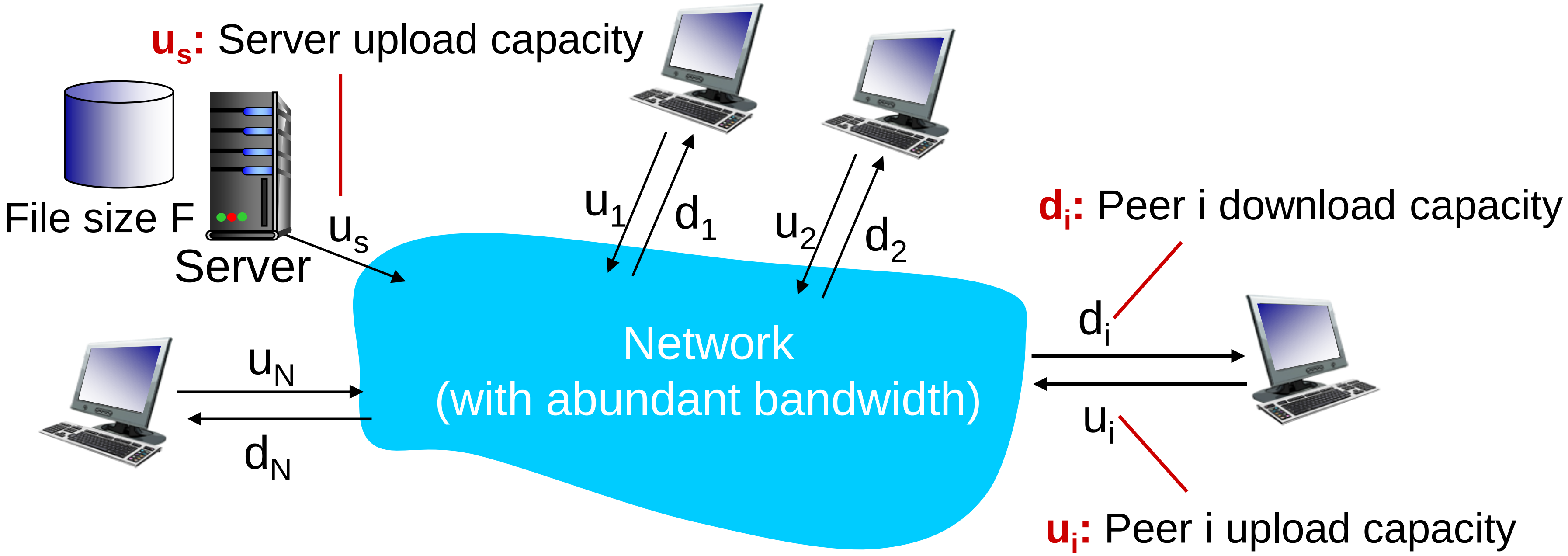
P2P architecture

- No always-on server
- Arbitrary end systems directly communicate
- Peers request service from other peers, provide service in return to other peers
    - Self scalability – new peers bring new service capacity, as well as new service demands
- Peers are intermittently connected and change IP addresses
    - Complex management

- Examples: P2P file sharing (BitTorrent)

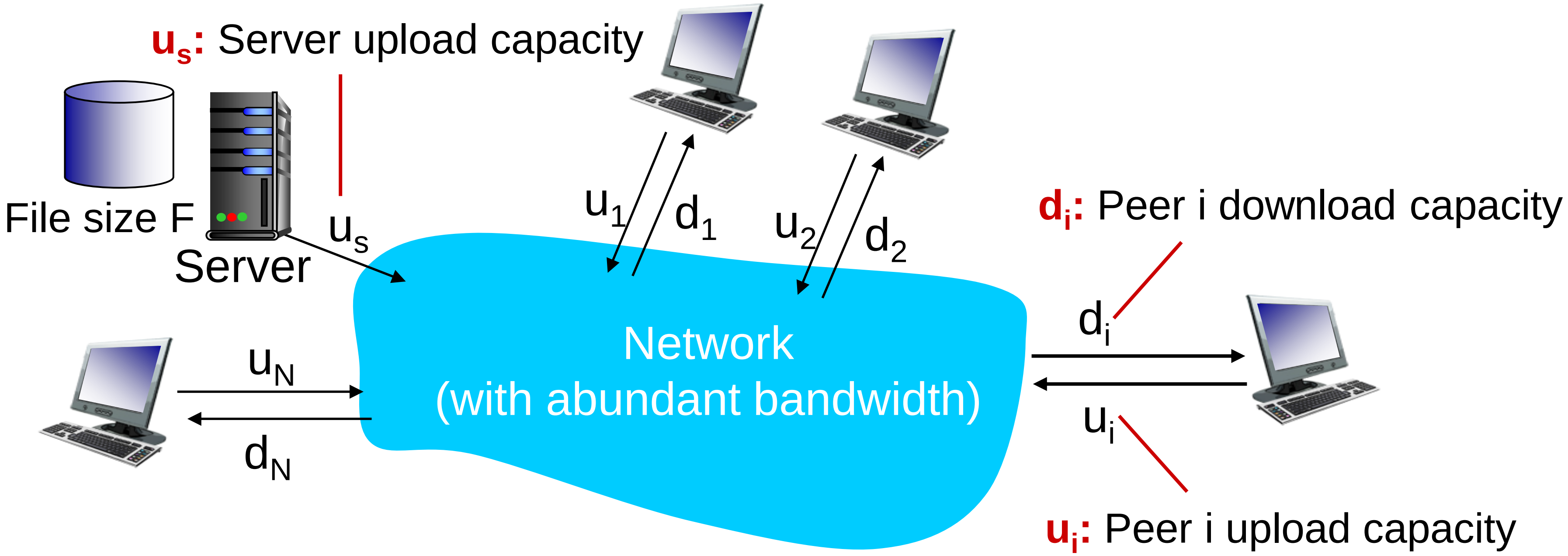


File Distribution Time: Client-server



$$\boxed{D_{C-S} \geq \max\{NF/u_s\ , F/d_{min}\}}$$

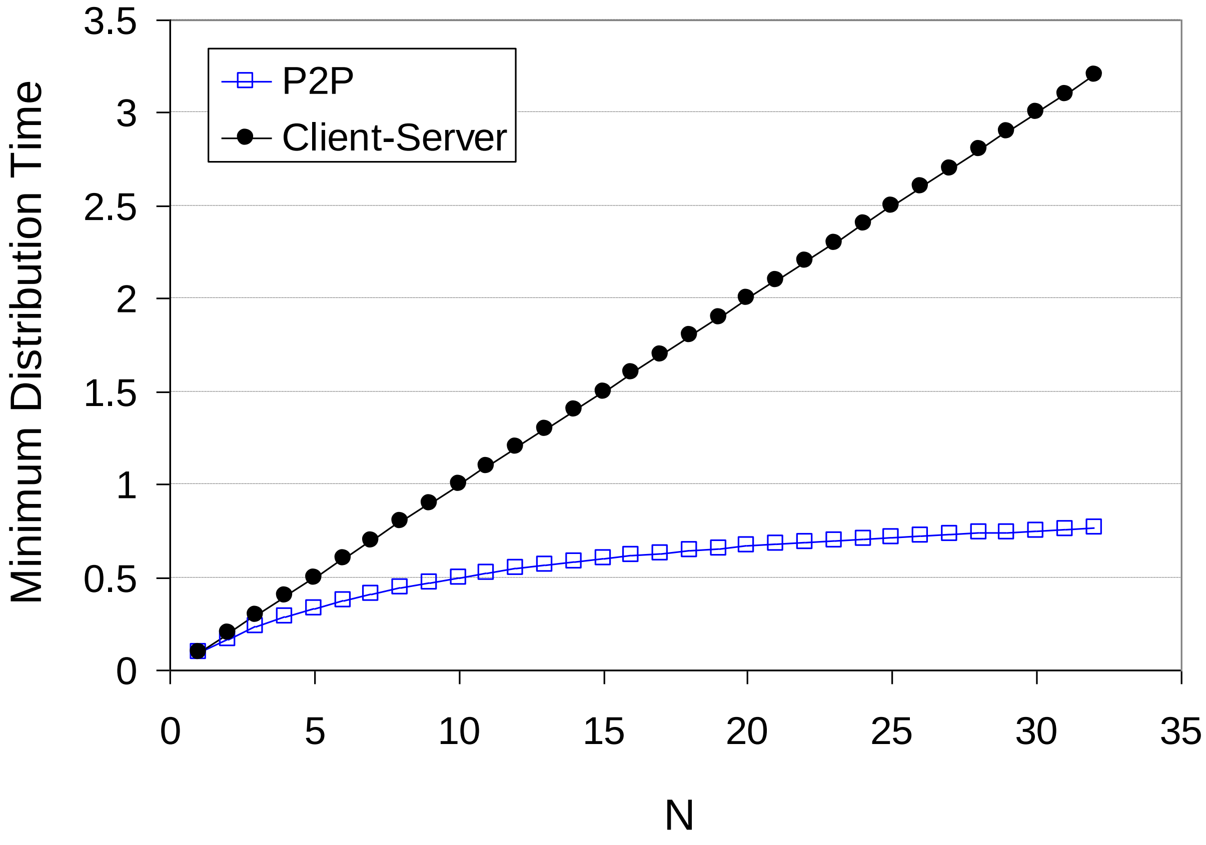
File Distribution Time: P2P



$$D_{P2P} \geq \max\{F/u_s\ , F/d_{min}, NF/(u_s + \sum u_i\ )\}$$

Client-server vs. P2P

Client upload rate = u
F/u = 1 hour
$u_s = 10u$
$d_{min} \geq u_s$



What Transport Service Does an App Need?

Data integrity

- Some apps (e.g., file transfer, web transactions) require 100% reliable data transfer
- Other apps (e.g., audio) can tolerate some loss

Timing

- Some apps (e.g., Internet telephony, interactive games) require low delay to be "effective"

Throughput

- Some apps (e.g., multimedia) require minimum amount of throughput to be "effective"
- Other apps ("elastic apps") make use of whatever throughput they get

Why throughput is different than timing constraint?

Internet Apps: Types and Requirements

Application	Data loss	Throughput	Time sensitive
File transfer	no loss	elastic	no
e-mail	no loss	elastic	no
Web documents	no loss	elastic	no
Real-time audio/video	loss-tolerant	audio: 5kbps-1Mbps video:10kbps-5Mbps	yes, 10’s msec
Stored audio/video	loss-tolerant	same as above	yes, few secs
Interactive games	loss-tolerant	few kbps up	yes, 10’s msec

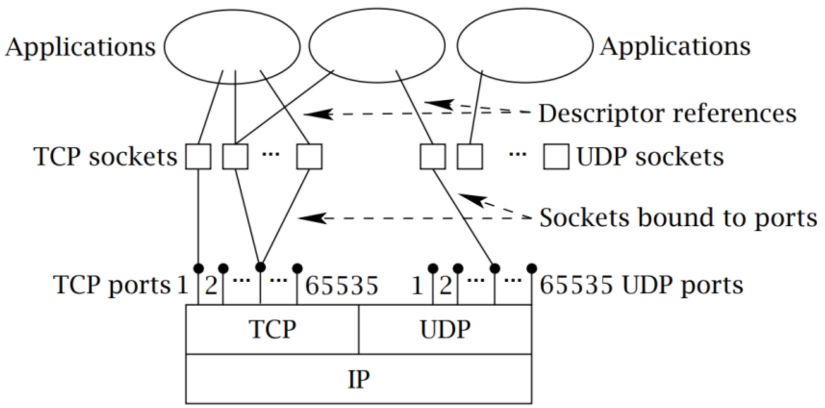
Internet Apps: Application, Transport protocols

Application	Application layer protocol	Underlying transport protocol
e-mail	SMTP [RFC 2821]	TCP
Remote terminal access	Telnet [RFC 854]	TCP
Web	HTTP [RFC 2616]	TCP
File transfer	FTP [RFC 959]	TCP
Streaming multimedia	HTTP (e.g., YouTube), RTP [RFC 1889]	TCP or UDP
Internet telephony	SIP, RTP, proprietary (e.g., Skype)	TCP or UDP

Internet Apps: Common Port Numbers

- Application processes are identified by IP addresses and port numbers
  - Popular apps have well known port numbers

Port Number	Protocol	Application
20	TCP	FTP data
21	TCP	FTP control
22	TCP	SSH
23	TCP	Telnet
25	TCP	SMTP
53	UDP, TCP[1]	DNS
67	UDP	DHCP Server
68	UDP	DHCP Client
69	UDP	TFTP
80	TCP	HTTP (WWW)
110	TCP	POP3
161	UDP	SNMP



Src: https://linuxwheel.com/chapter-5-fundamentals-of-tcp-ip-transport-and-application/

Domain Name System (DNS)

DNS: Domain Name System

- Domain name: Identifies Internet resources, such as computers, networks, and services, with a text-based label

- Human prefers hostnames/domain names
  - www.google.com - used by humans
- Internet hosts, routers use IP addresses

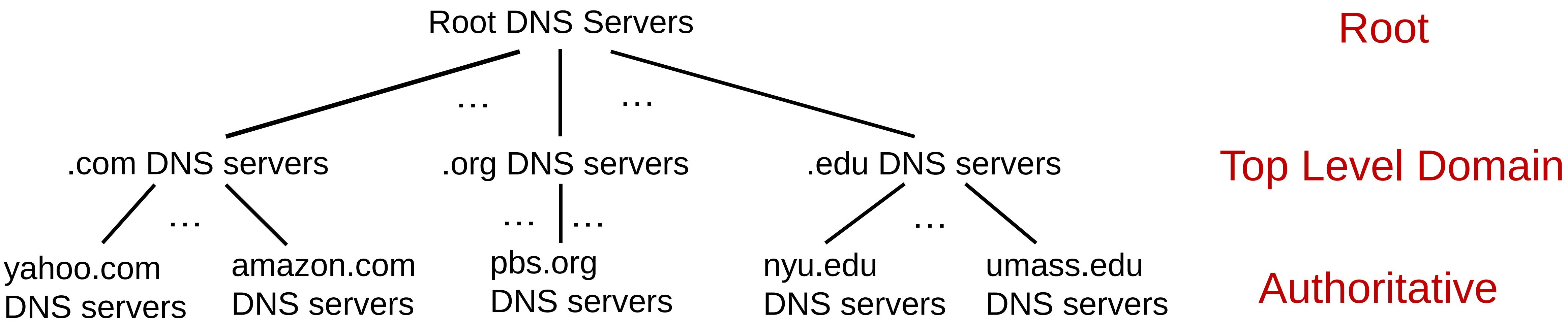
- DNS services: Hostname to IP address translation
  - Runs on UDP and uses port number 53

THE 100 OLDEST CURRENTLY-REGISTERED .COM DOMAINS

Rank	Create date	Domain name
1.	15-Mar-1985	SYMBOLICS.COM
2.	24-Apr-1985	BBN.COM
3.	24-May-1985	THINK.COM
4.	11-Jul-1985	MCC.COM
5.	30-Sep-1985	DEC.COM
6.	07-Nov-1985	NORTHROP.COM
7.	09-Jan-1986	XEROX.COM
8.	17-Jan-1986	SRI.COM
9.	03-Mar-1986	HP.COM
10.	05-Mar-1986	BELLCORE.COM
11=	19-Mar-1986	IBM.COM
11=	19-Mar-1986	SUN.COM
13=	25-Mar-1986	INTEL.COM
13=	25-Mar-1986	TI.COM
15.	25-Apr-1986	ATT.COM
16=	08-May-1986	GMR.COM
16=	08-May-1986	TEK.COM
18=	10-Jul-1986	FMC.COM
18=	10-Jul-1986	UB.COM
20=	05-Aug-1986	BELL-ATL.COM

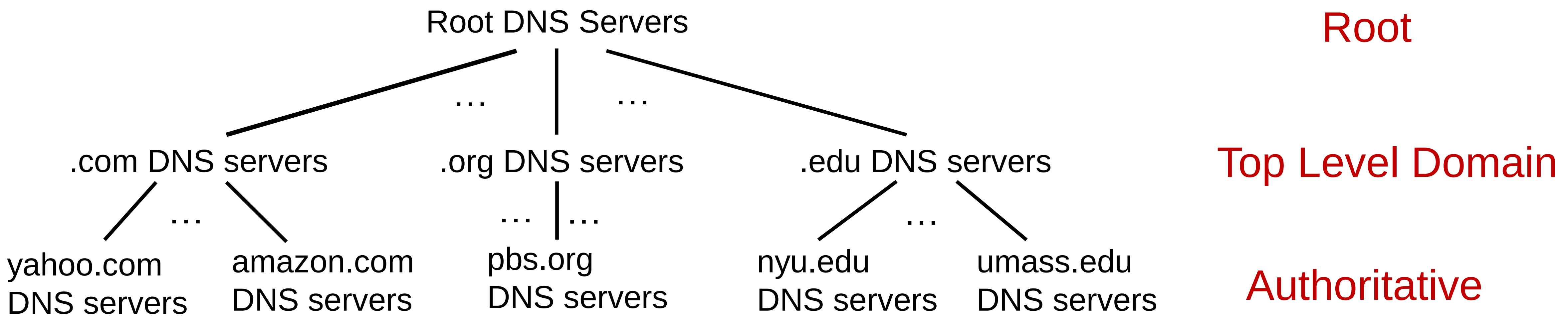
Src: https://www.flickr.com/photos/simonbarratt/106385627

DNS: A Distributed, Hierarchical Database



- Distributed database implemented in hierarchy of many name servers
- Application-layer protocol: hosts, name servers communicate to resolve names (name → address translation)

DNS: A Distributed, Hierarchical Database

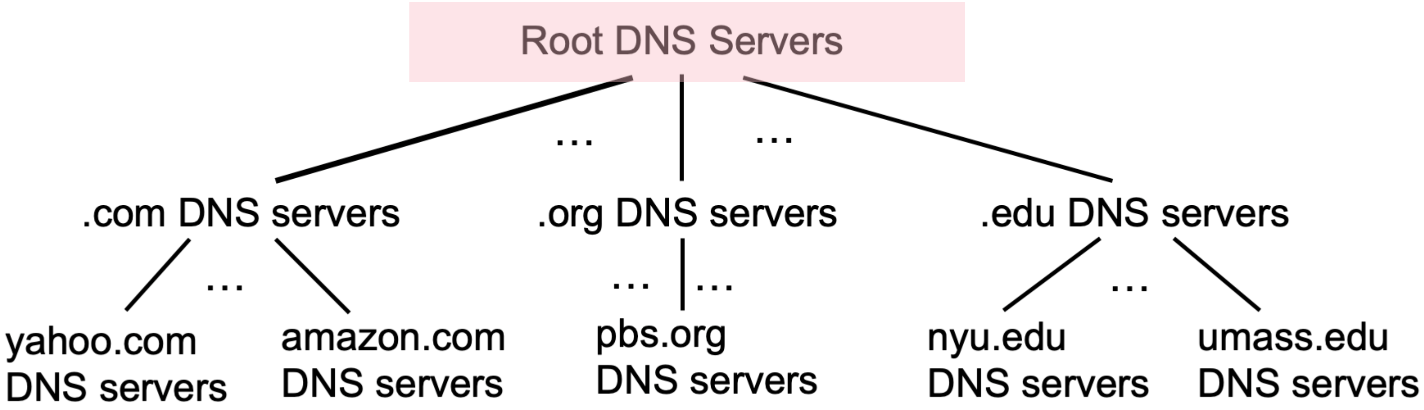


Client wants IP for www.amazon.com:

- Client queries root server to find .com DNS server
- Client queries .com DNS server to get amazon.com DNS server
- Client queries amazon.com DNS server to get IP address for www.amazon.com

DNS: Root Name Servers

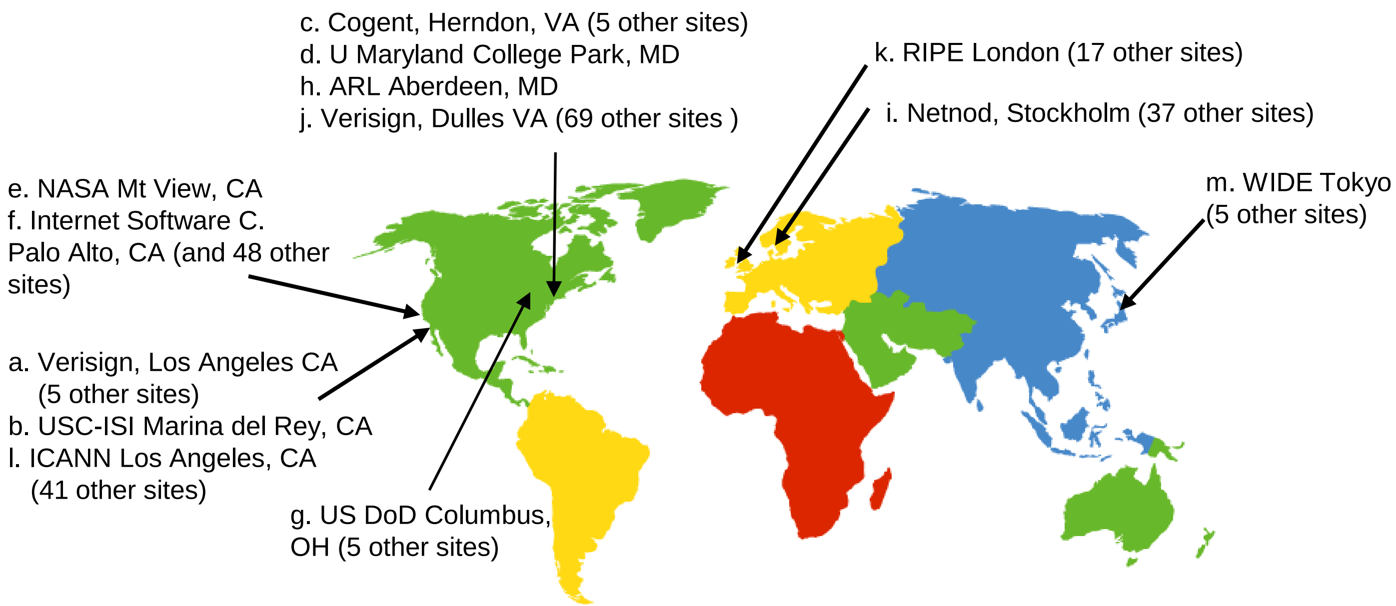
- Contacted by local name server that can not resolve name
- Root name server:
  - Provides IP addresses of the TLD servers
  - ICANN (Internet Corporation for Assigned Names and Numbers) manages root DNS domain



DNS: Root Name Servers

13 logical root name “servers” worldwide

- Each “server” replicated many times



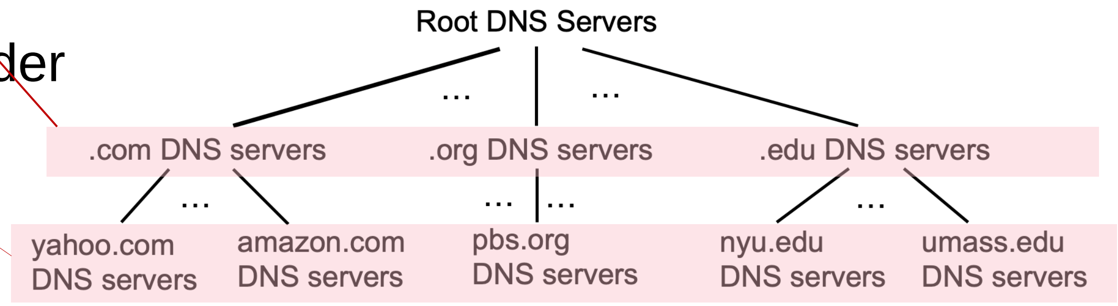
TLD, Authoritative Servers

Top-level Domain (TLD) Servers:

- Responsible for com, org, net, edu, and all top-level country domains, e.g.: in, uk, fr, ca, jp
  - Verisign Global Registry Services maintains the TLD servers for the .com domain
  - Educause maintains the TLD servers for the .edu domain

Authoritative DNS servers:

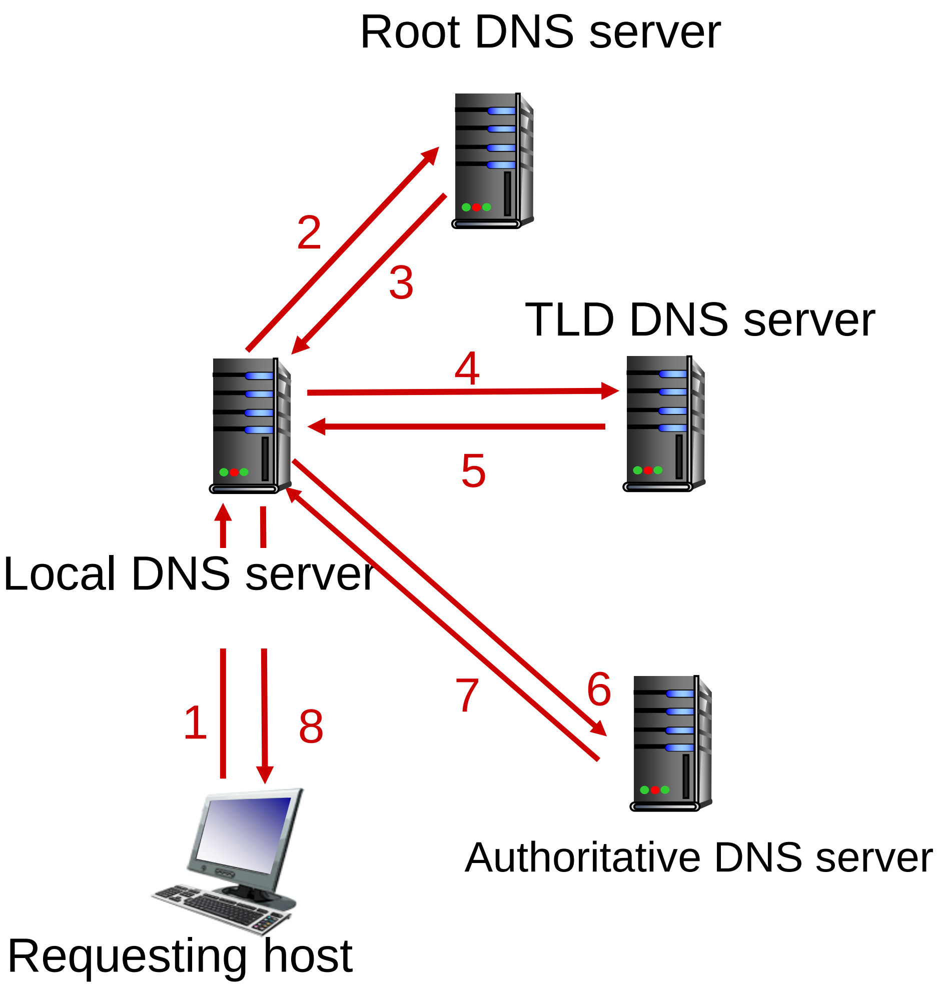
- Organization's own DNS server(s), providing authoritative hostname to IP mappings for organization's named hosts
- Can be maintained by organization or service provider



Local DNS Name Server

- Does not strictly belong to hierarchy
- Each ISP (residential ISP, company, university) has one or more local DNS servers
  - 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
  - Acts as proxy, forwards query into hierarchy

DNS Name Resolution



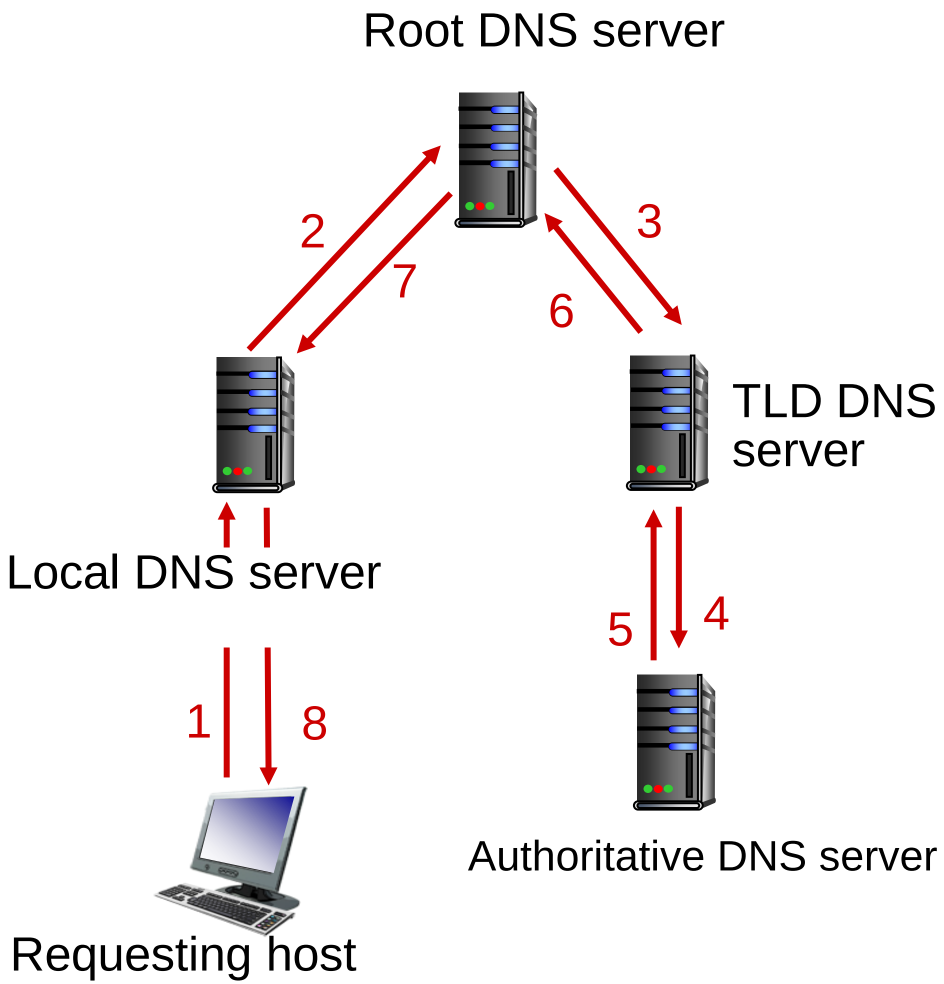
Iterated query:

- Contacted server replies with name of server to contact
- “I don’t know this name, but ask this server”

DNS Name Resolution

Recursive query:

- Puts burden of name resolution on contacted name server
- Heavy load at upper levels of hierarchy



DNS: Caching, Updating Records

- Once (any) name server learns mapping, it caches mapping
  - Cache entries timeout (disappear) after some time (TTL)
  - TLD servers typically cached in local name servers
    - Thus root name servers not often visited

DNS Services

DNS services:

- Hostname-to-IP-address translation
- Host aliasing
  - Canonical, alias names
  - Many names may map to same IP address
- Mail server aliasing
  - hotmail.com → relay1.west-coast.hotmail.com
- Load distribution
  - Replicated Web servers: many IP addresses correspond to one name

DNS: Distributed Database 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 hostname of authoritative name server for this domain

; Authoritative data for cs.vu.nl

cs.vu.nl.	86400	IN	MX	1 zephyr
cs.vu.nl.	86400	IN	MX	2 top
cs.vu.nl.	86400	IN	NS	star
star	86400	IN	A	130.37.56.205
zephyr	86400	IN	A	130.37.20.10
top	86400	IN	A	130.37.20.11
www	86400	IN	CNAME	star.cs.vu.nl
ftp	86400	IN	CNAME	zephyr.cs.vu.nl

DNS: Distributed Database Storing Resource Records (RR)

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

type=CNAME

- `name` is alias name for some "canonical" (the real) name
- `value` is canonical name

type=MX

- `value` is name of SMTP mail server associated with `name`

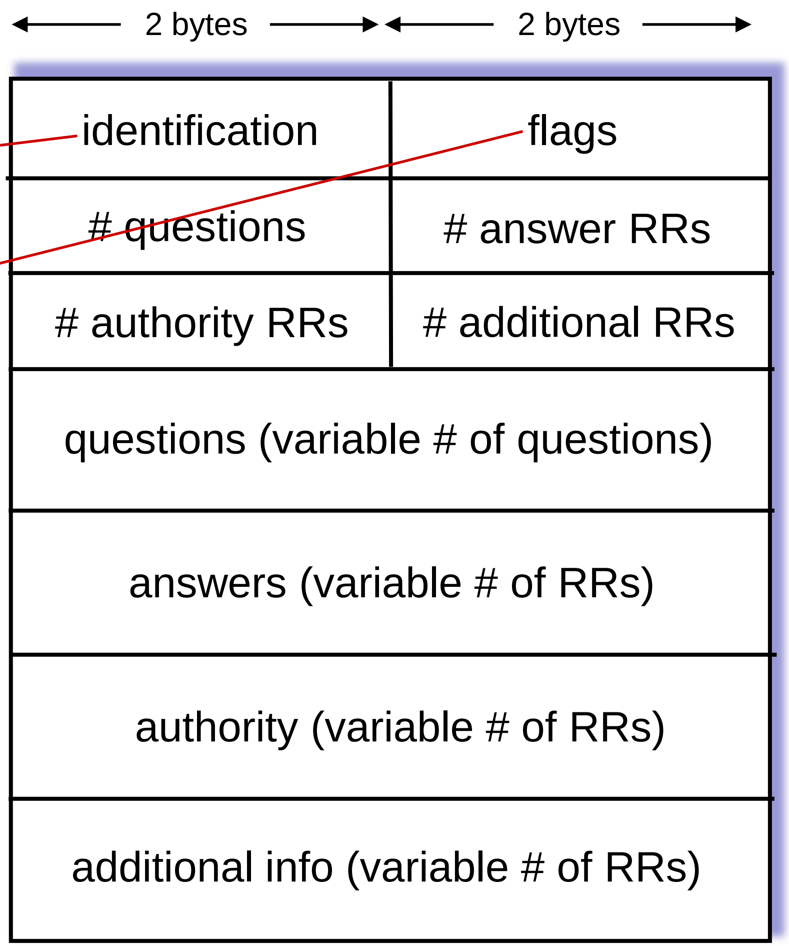
; Authoritative data for cs.vu.nl

cs.vu.nl.	86400	IN	MX	1 zephyr
cs.vu.nl.	86400	IN	MX	2 top
cs.vu.nl.	86400	IN	NS	star
star	86400	IN	A	130.37.56.205
zephyr	86400	IN	A	130.37.20.10
top	86400	IN	A	130.37.20.11
www	86400	IN	CNAME	star.cs.vu.nl
ftp	86400	IN	CNAME	zephyr.cs.vu.nl

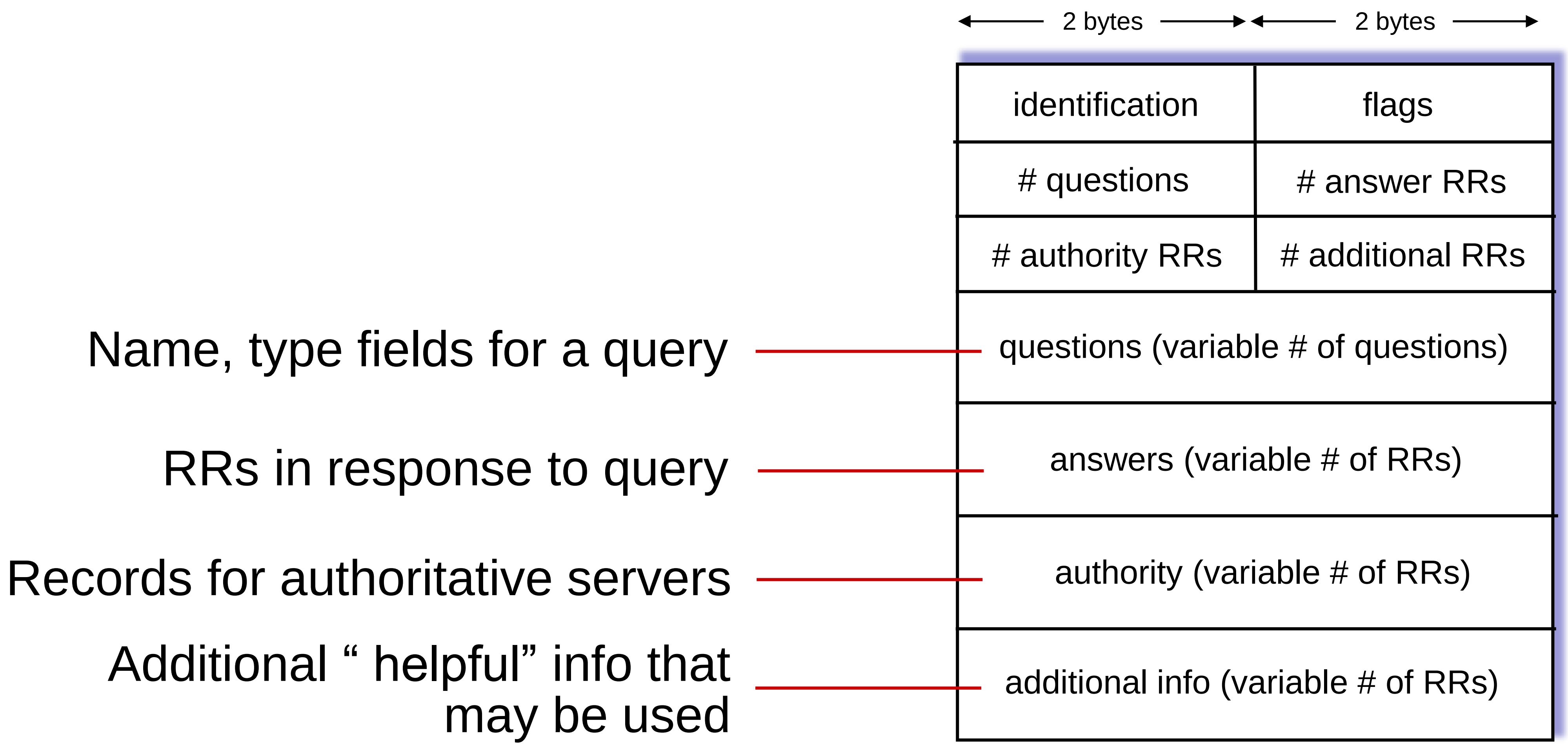
DNS Query and Reply Messages

Message header:

- Identification: 16 bit # for query, reply to query uses same #
- Flags:
  - Query or reply
  - Recursion desired
  - Recursion available
  - Reply is authoritative



DNS Query and Reply Messages



Getting Your Info into the DNS

Example: new startup “Network Utopia”

- Register name networkuptopia.com at DNS registrar
  - Provide names, IP addresses of authoritative name server (primary and secondary)
  - Registrar inserts NS, A RRs into .com TLD server:

    (networkutopia.com, dns1.networkutopia.com, NS)

    (dns1.networkutopia.com, 212.212.212.1, A)

- Create authoritative server locally with IP address 212.212.212.1
  - Type A and MX records for www.networkuptopia.com

Summary

- Application architectures:
  - Client-server model
  - Peer-to-peer model

- DNS:
  - Hostname to IP address mapping