UIC UNIVERSITY OF ILLINOIS AT CHICAGO Department of Computer Science
COLLEGE OF ENGINEERING

CS450 – Introduction to Networking
Lecture 8 – DNS

Phu Phung

January 30, 2015

Domain Name System (DNS) provides translation services of

A. Domain name to IP address

B. Domain name aliases

C. Mail server alias for a domain name

D. A, B, C

E. A, B

DNS: domain name system

people: many identifiers:

- SSN, name, passport #

Internet hosts, routers:

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

Q: how to map between IP address and name, and vice versa ?

Domain Name System:

- *distributed database* implemented in hierarchy of many *name servers*
- *application-layer protocol:* hosts, name servers communicate to *resolve* names (address/name translation)
  - note: core Internet function, implemented as application-layer protocol
  - complexity at network's "edge"

DNS: services, structure

DNS services

- hostname to IP address translation
- host aliasing
  - canonical, alias names
- mail server aliasing
- load distribution
  - replicated Web servers: many IP addresses correspond to one name

why not centralize DNS?

- single point of failure
- traffic volume
- distant centralized database
- maintenance

A: doesn't scale!

Select a wrong statement

A. A name in DNS might be mapped with many IP addresses
B. Multiple domain name might be pointed to one IP address
C. DNS service can provide a reverse lookup from an IP address to a domain name
D. DNS only uses TCP transport service

DNS reverse lookup

$nslookup 8.8.8.8
Server: 192.168.0.1
Address: 192.168.0.1#53

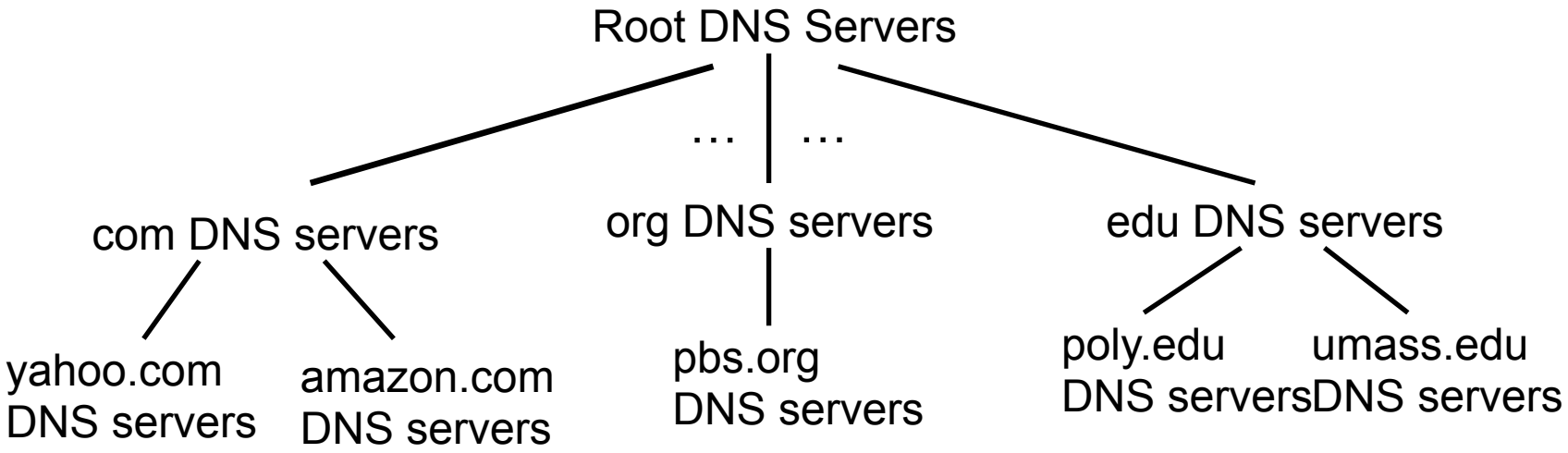
Non-authoritative answer:
8.8.8.8.in-addr.arpa name = google-public-dns-a.google.com.

$ nslookup 8.8.8.8 garcon.eecs.uic.edu
Server: garcon.eecs.uic.edu
Address: 131.193.32.254#53

Non-authoritative answer:
8.8.8.8.in-addr.arpa name = google-public-dns-a.google.com.

Authoritative answers can be found from:
8.in-addr.arpa nameserver = ns2.level3.net.
8.in-addr.arpa nameserver = ns1.level3.net.

DNS: a distributed, hierarchical database

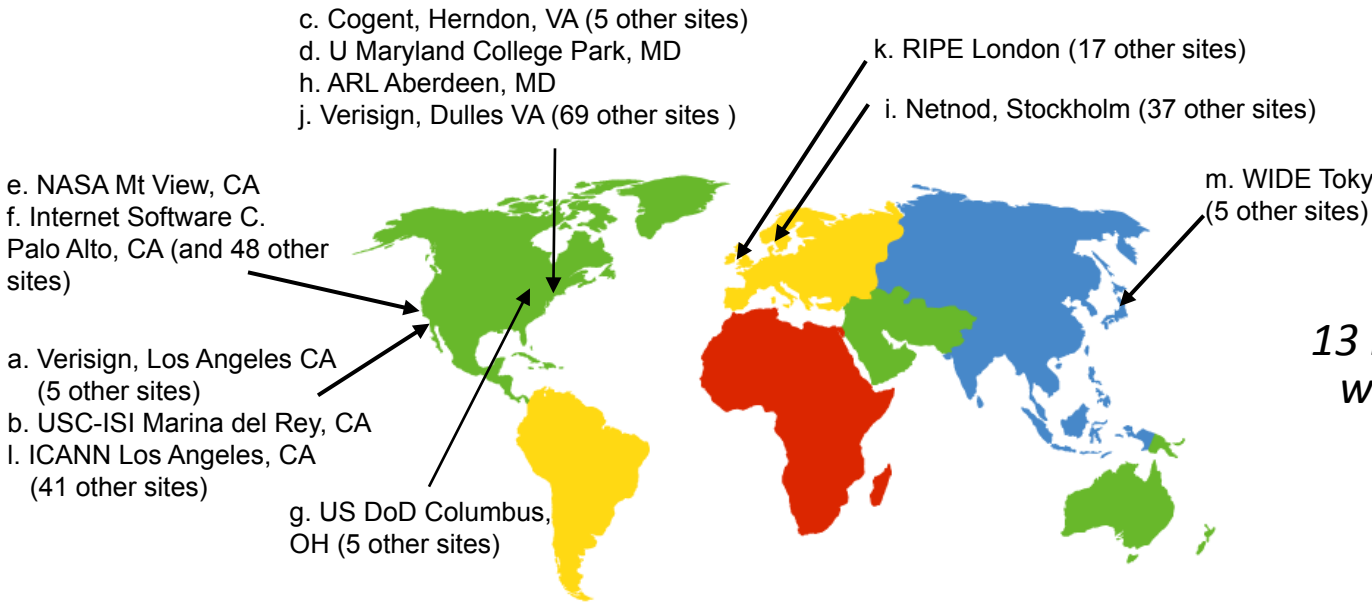


client wants IP for www.amazon.com; 1st approx:

- 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:
  - contacts authoritative name server if name mapping not known
  - gets mapping
  - returns mapping to local name server



13 root name "servers" worldwide

```
$ dig
; <<>> DiG 9.8.3-P1 <<>>
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 38364
;; flags: qr rd ra; QUERY: 1, ANSWER: 13, AUTHORITY: 0, ADDITIONAL: 13

;; QUESTION SECTION:
;.                      IN    NS

;; ANSWER SECTION:
.                  100262   IN    NS    j.root-servers.net.
.                  100262   IN    NS    e.root-servers.net.
.                  100262   IN    NS    c.root-servers.net.
.                  100262   IN    NS    k.root-servers.net.
.                  100262   IN    NS    d.root-servers.net.
.                  100262   IN    NS    b.root-servers.net.
.                  100262   IN    NS    l.root-servers.net.
.                  100262   IN    NS    i.root-servers.net.
.                  100262   IN    NS    h.root-servers.net.
.                  100262   IN    NS    f.root-servers.net.
.                  100262   IN    NS    m.root-servers.net.
.                  100262   IN    NS    g.root-servers.net.
.                  100262   IN    NS    a.root-servers.net.
```

TLD, authoritative servers

top-level domain (TLD) servers:

- responsible for com, org, net, edu, aero, jobs, museums, and all top-level country domains, e.g.: uk, fr, ca, jp
- Network Solutions maintains servers for .com TLD
- Educause for .edu TLD

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

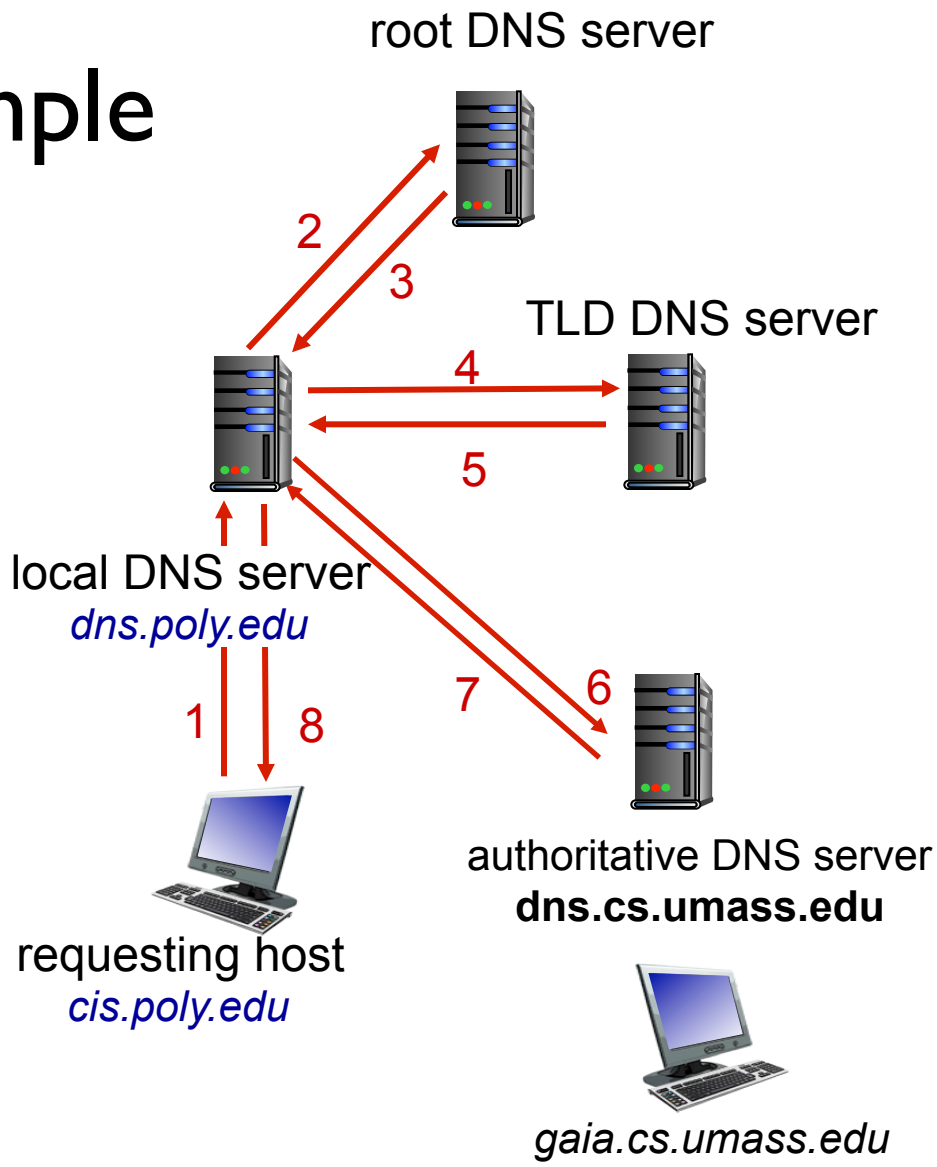
- does not strictly belong to hierarchy
- each ISP (residential ISP, company, university) has one
  - also called "default name server"
- when host makes DNS query, query is sent to its local DNS server
  - has local cache of recent name-to-address translation pairs (but may be out of date!)
  - acts as proxy, forwards query into hierarchy

DNS name resolution example

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

iterated query:

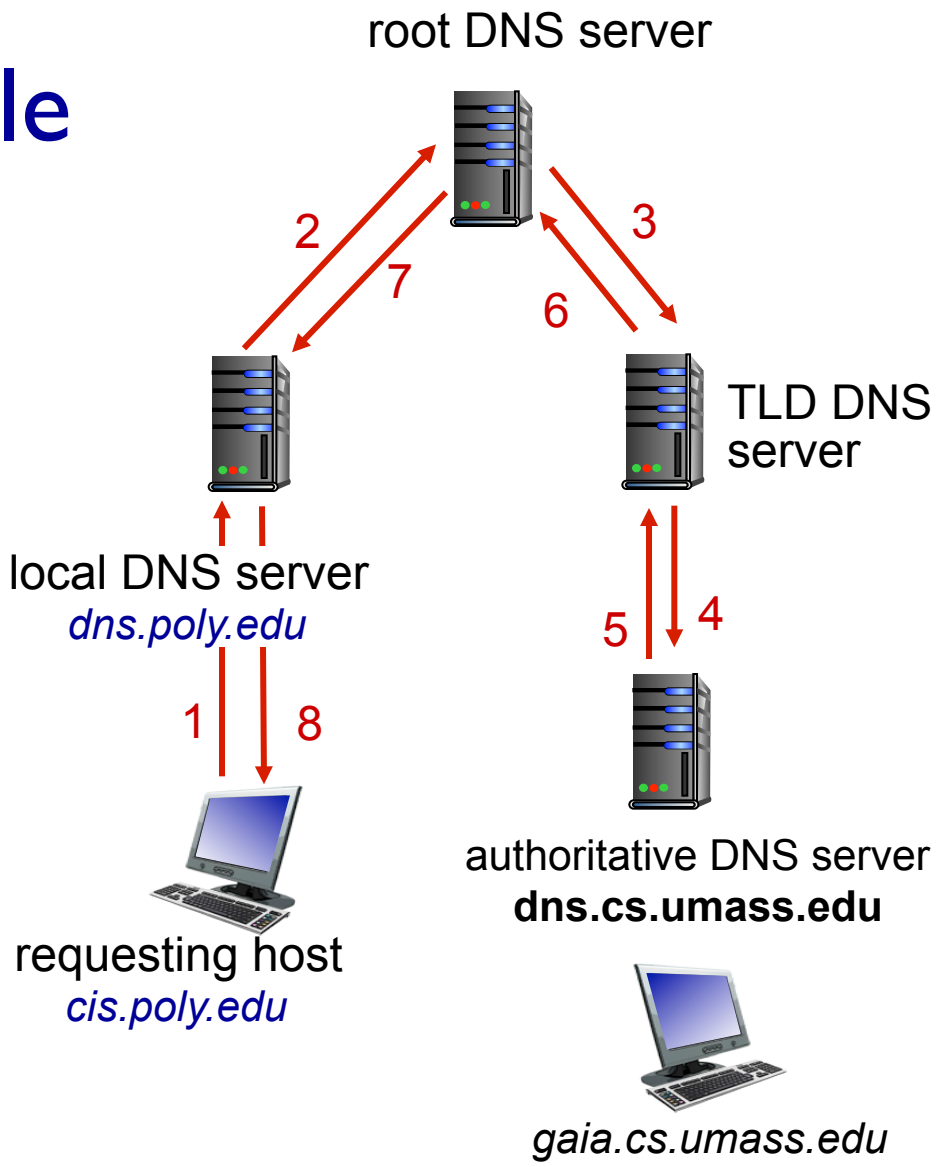
- contacted server replies with name of server to contact
- "I don't know this name, but ask this server"



DNS name resolution example

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
- cached entries may be *out-of-date* (best effort name-to-address translation!)
  - if name host changes IP address, may not be known Internet-wide until all TTLs expire
- update/notify mechanisms proposed IETF standard
  - RFC 2136

hosts local database

```
##
# Host Database
#
# localhost is used to configure the
loopback interface
# when the system is booting.  Do not
change this entry.
##
127.0.0.1       localhost
255.255.255.255 broadcasthost
::1             localhost
fe80::1%lo0     localhost
172.252.120.6   facebook.com
```

- Unix-like (+Mac)

  /etc/hosts

- Windows

  %SystemRoot%
  \system32\drivers\etc
  \hosts

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 hostname of authoritative name server for this domain

type=CNAME

- **`name`** is alias name for some "canonical" (the real) name
- **`www.ibm.com`** is really **`servereast.backup2.ibm.com`**
- **`value`** is canonical name

type=MX

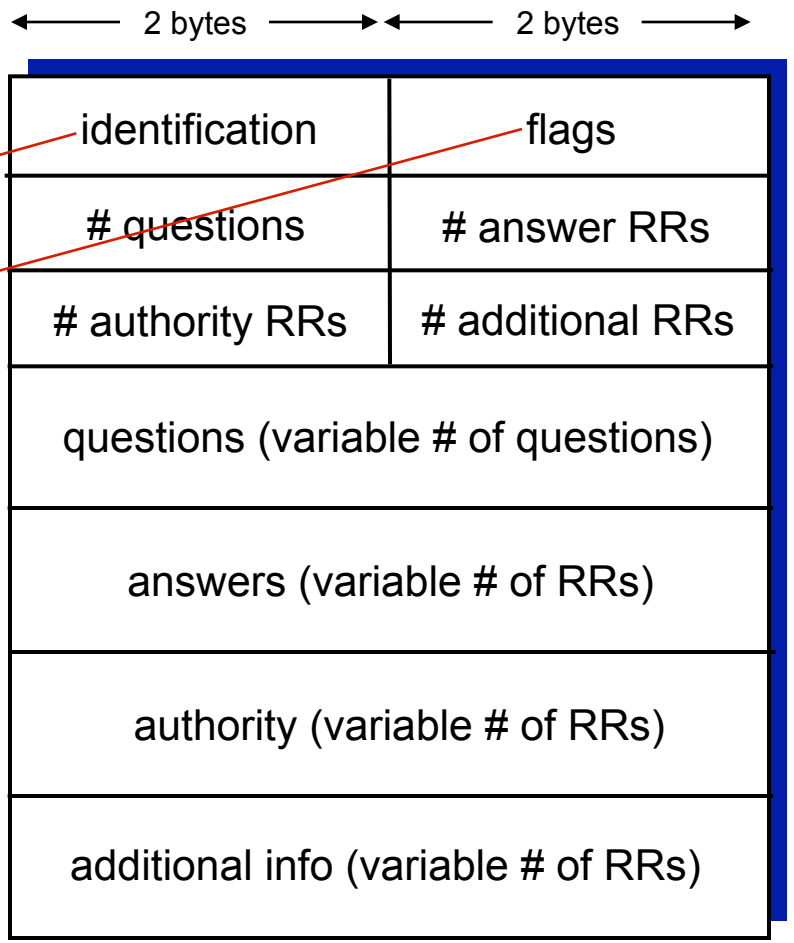
- **`value`** is name of mailserver associated with **`name`**

DNS protocol, messages

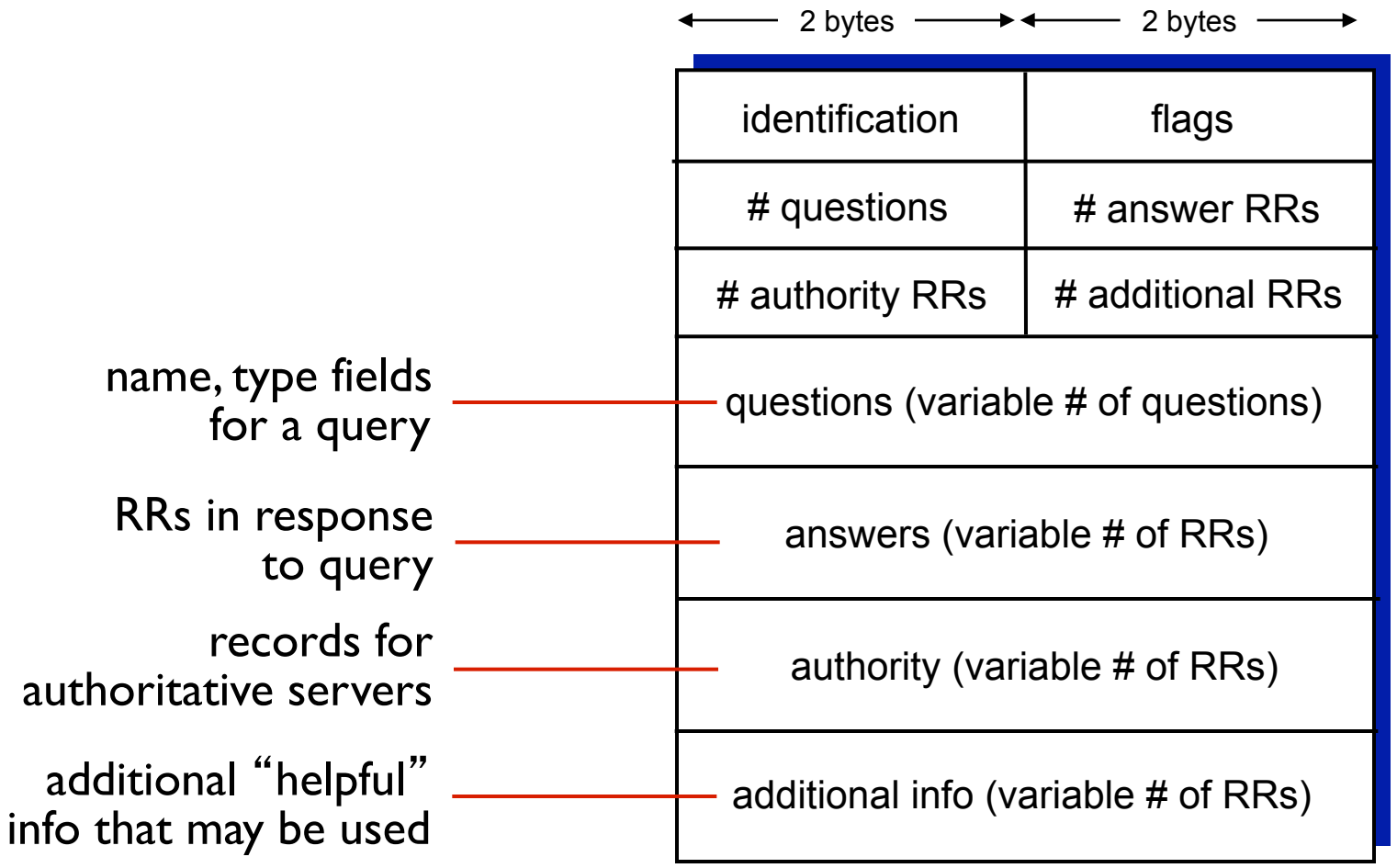
- *query* and *reply* messages, both with same *message format*

msg header

- identification: 16 bit # for query, reply to query uses same #
- flags:
  - query or reply
  - recursion desired
  - recursion available
  - reply is authoritative



DNS protocol, messages



DNS query sample

```
Domain Name System (query)
   [Response In: 20]
   Transaction ID: 0x0003
▽  Flags: 0x0100 Standard query
     0... .... .... .... = Response: Message is a query
     .000 0... .... .... = Opcode: Standard query (0)
     .... ..0. .... .... = Truncated: Message is not truncated
     .... ...1 .... .... = Recursion desired: Do query recursively
     .... .... .0.. .... = Z: reserved (0)
     .... .... ...0 .... = Non-authenticated data: Unacceptable
   Questions: 1
   Answer RRs: 0
   Authority RRs: 0
   Additional RRs: 0
▽  Queries
   ▷ www.mit.edu: type A, class IN
```

DNS response sample

```
Domain Name System (response)
   [Request In: 19]
   [Time: 0.016757000 seconds]
   Transaction ID: 0x0003
▷  Flags: 0x8580 Standard query response, No error
   Questions: 1
   Answer RRs: 1
   Authority RRs: 3
   Additional RRs: 3
▽  Queries
   ▷ www.mit.edu: type A, class IN
▽  Answers
   ▷ www.mit.edu: type A, class IN, addr 18.7.22.83
▽  Authoritative nameservers
   ▷ mit.edu: type NS, class IN, ns BITSY.mit.edu
   ▷ mit.edu: type NS, class IN, ns STRAWB.mit.edu
   ▷ mit.edu: type NS, class IN, ns W20NS.mit.edu
▽  Additional records
   ▷ BITSY.mit.edu: type A, class IN, addr 18.72.0.3
   ▷ STRAWB.mit.edu: type A, class IN, addr 18.71.0.151
   ▷ W20NS.mit.edu: type A, class IN, addr 18.70.0.160
```

Inserting records into DNS

- example: new startup "Network Utopia"
- register name networkuptopia.com at *DNS registrar* (e.g., Network Solutions)
  - provide names, IP addresses of authoritative name server (primary and secondary)
  - registrar inserts two RRs into .com TLD server:
    `(networkutopia.com, dns1.networkutopia.com, NS)`
    `(dns1.networkutopia.com, 212.212.212.1, A)`
- create authoritative server type A record for www.networkuptopia.com; type MX record for networkutopia.com

When lookup a domain name, your machine first sends a request to

A. The local name server of your network

B. Top-level name server

C. Authoritative name server

D. Root name server

E. A name server set in your machine

Homework (optional)

- $dig
- $nslookup
- $wireshark

Next lecture

- P2P
  - Readings 2.6