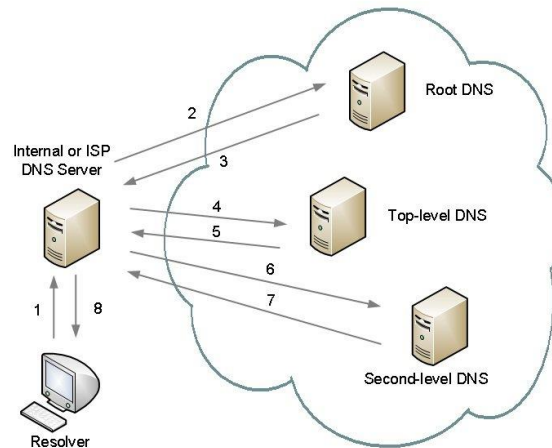


CS348: Computer Networks

DNS



Dr. Manas Khatua

Assistant Professor

Dept. of CSE, IIT Guwahati

E-mail: manaskhatua@iitg.ac.in

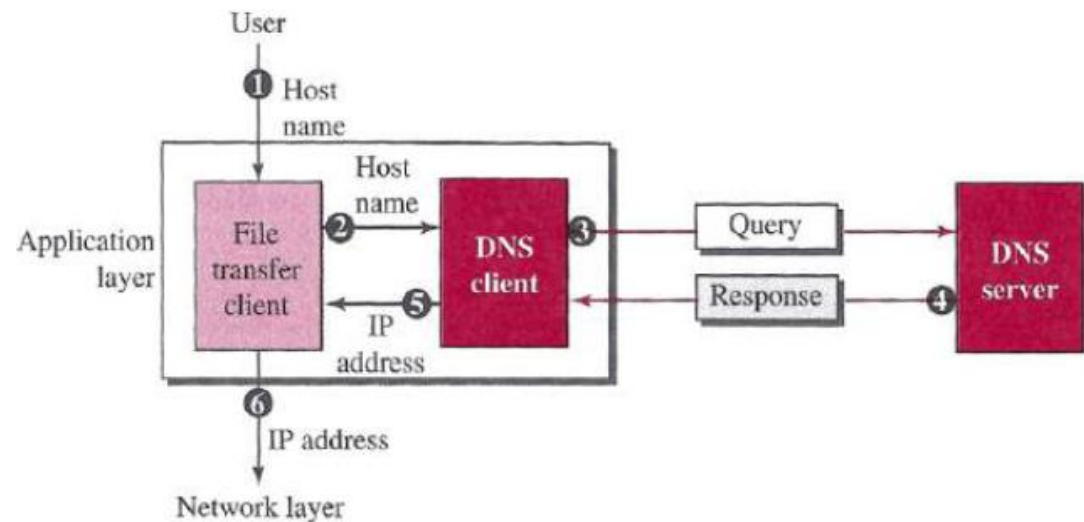
DNS - Internet's Directory Service




- Just as humans can be identified in many ways, so too can **Internet hosts**.
- Two ways:
 - **Hostname** (e.g., gmail.co.in, iitg.ac.in)
 - these are **mnemonic**, user friendly for **Humans**
 - **IP Address** (e.g., 121.7.106.83, 172.17.0.10)
 - these are structured **numeric** digits, user friendly for **Routers**
- The **Internet** needs to have **a directory system that can map a name to an address.**
- The Internet is so huge today
 - a **central directory system** cannot hold all the mapping.
- **Better solution:**
 - distribute the directory information **among many computers** in the world.
 - This method is used by the **Domain Name System (DNS)**.

Cont...

- The DNS is a combination of :
 - a **distributed database** -- implemented in a hierarchy of **DNS servers**, and
 - an **application-layer protocol** -- that **allows hosts to query** the distributed database
- Let the purpose of accessing the Internet is to make a connection between the **file transfer client** and **server**. But before **this can happen, another connection needs to be made between the DNS client and DNS server**
- **DNS protocol** runs over **UDP (/TCP)** and uses **port 53**.
- The **DNS servers** are often UNIX machines running the Berkeley Internet Name Domain (BIND) software
- In UNIX and Windows, the **nslookup** utility can be used to retrieve address/name mapping.

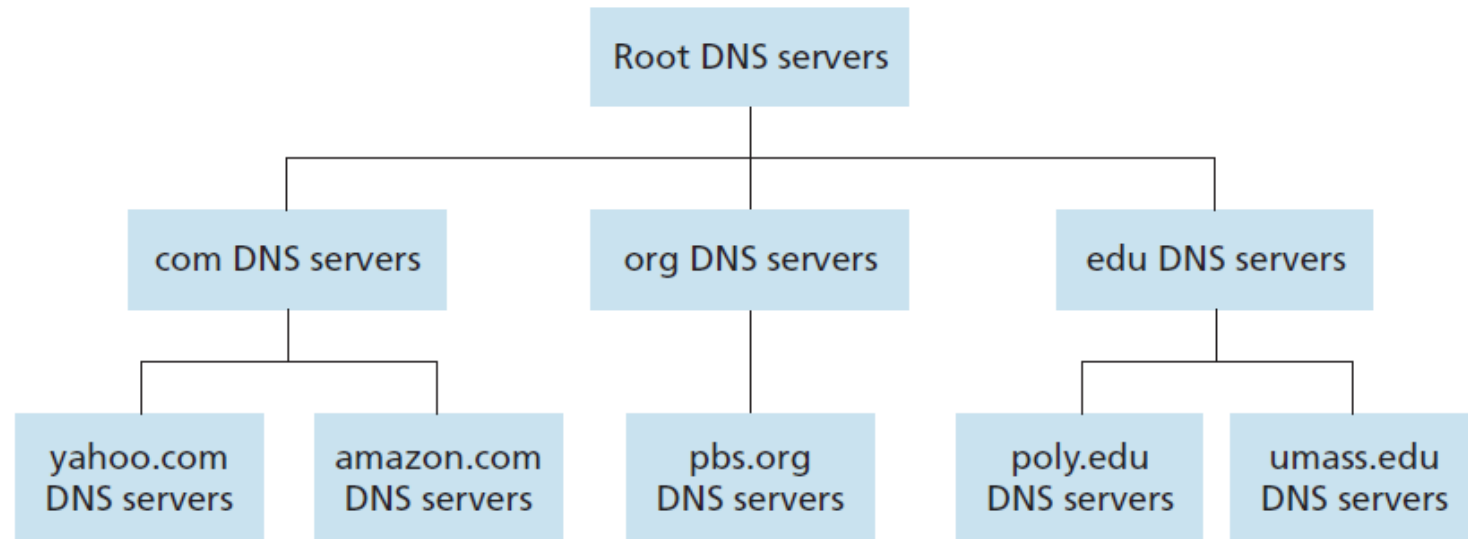


Design for DNS

- Design for DNS:
 - Centralized / Distributed
 - The problems with a **centralized design** include:
 - **A single point of failure**: DNS server crashes, so does the entire Internet!
- 
- **Traffic volume**: A single DNS server would have to handle all DNS queries generated from hundreds of millions of hosts
 - **Distant database**: A single DNS server cannot be “close to” all the querying clients.
 - **Maintenance**: The single DNS server would have to keep records for all Internet hosts. Management of it becomes very difficult!

- **Fundamental service** : directory service (translates hostnames to IP addresses).
- provides a few **other** important **services** :
 - **Host aliasing**: [relay1.west-coast.enterprise.com](#) could have, say, two aliases such as [enterprise.com](#) and [www.enterprise.com](#)
 - **Mail server aliasing**: the canonical hostname of the Hotmail server might be something like [relay1.west-coast.hotmail.com](#) but the mail server is simply [hotmail.com](#)
 - **Load distribution**: used to perform **load distribution** among replicated servers. For replicated servers, a set of IP addresses is thus associated with one canonical hostname.

Hierarchy of DNS servers



- the mappings for all the hosts in Internet are distributed across the DNS servers
- three classes of DNS servers
 - **root** DNS servers (until 2012, Internet has 13 root DNS servers)
 - top-level domain (**TLD**) DNS servers
 - **authoritative** DNS servers (large university /organization may have it)
 - **local** DNS server : Each ISP has one or more local DNS

Cont...

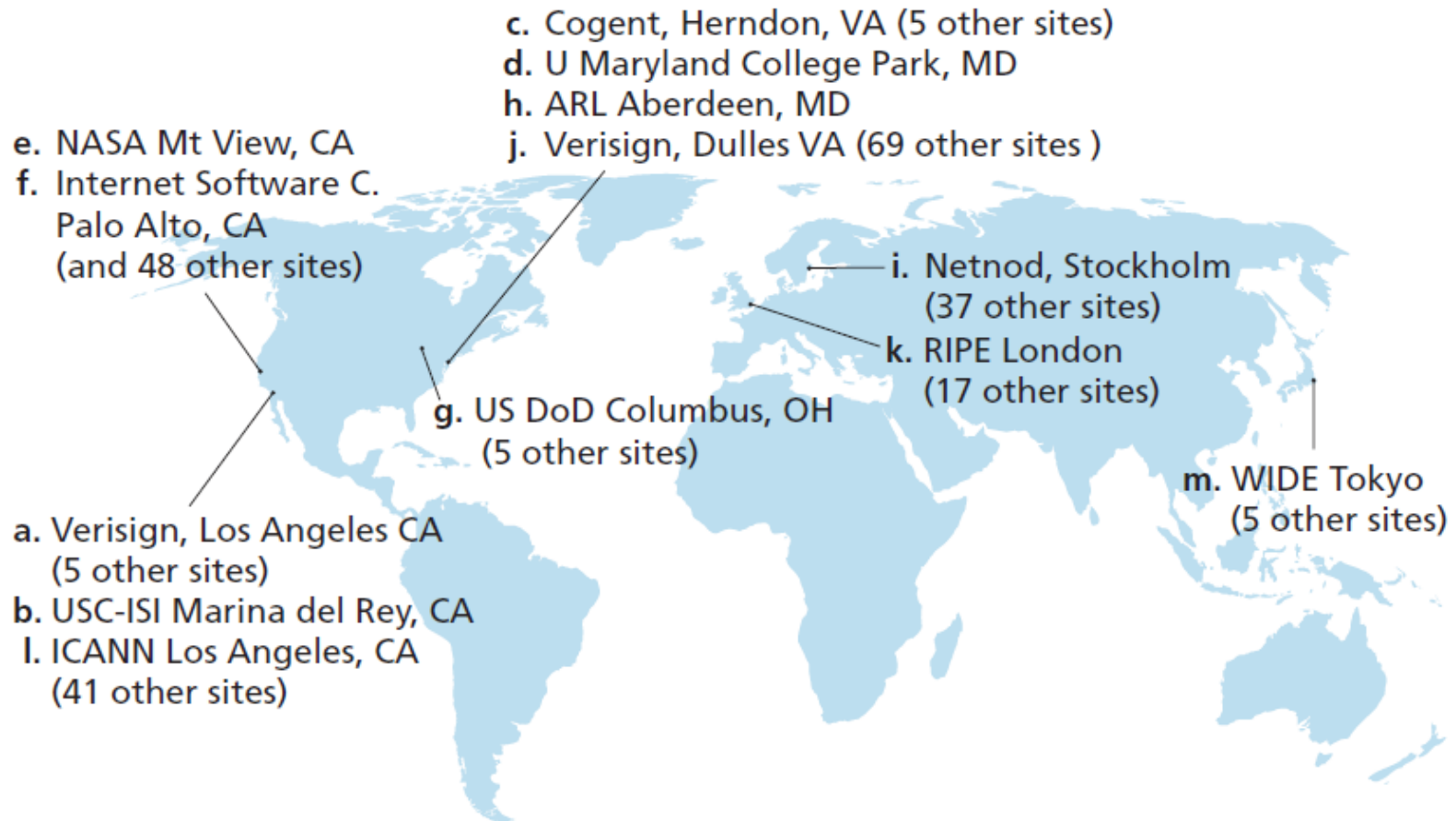
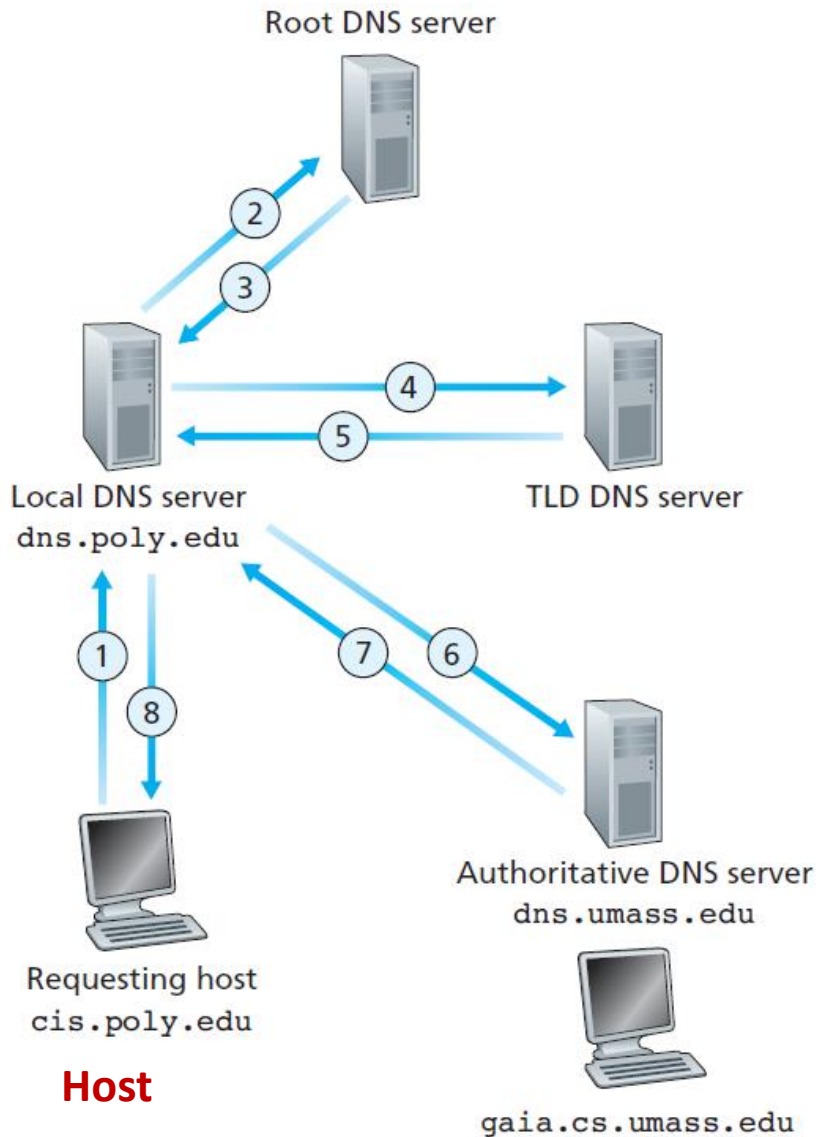


Figure 2.20 ♦ DNS root servers in 2012 (name, organization, location)

Interaction among DNS servers



- Let the **host** `cis.poly.edu` desires the IP address of `gaia.cs.umass.edu`.
- Let the Polytechnic's **local** DNS server is called `dns.poly.edu`
- Let an **authoritative** DNS server for `gaia.cs.umass.edu` is called `dns.umass.edu`.

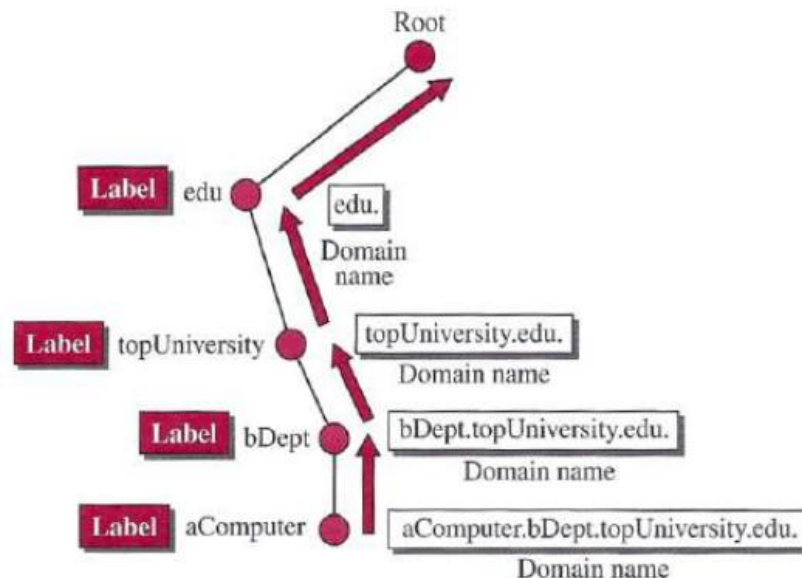
- The **host** first sends a DNS query message to its **local** DNS server.
- The **local** DNS server forwards the query message to a **root** DNS server.
- The **root** DNS server takes note of the `edu` suffix and **returns** a list of IP addresses for **TLD** servers responsible for `edu`.
- The **local** DNS server then resends the query to one of these **TLD** servers.
- The **TLD** server responds with the IP address of the **authoritative** DNS server
- Finally, the **local** DNS server resends the query message directly to the **authoritative** DNS server

Name Space

- the **names must be unique** because the addresses are unique.
- A **name space** that **maps each address to a unique name** can be organized in two ways:
 - flat
 - hierarchical
- **flat name space**
 - a name is assigned to an address
 - a name is a **sequence of characters without structure**
 - The names may or may not have a common section
 - **Disadvantage**: it cannot be used in a large system such as the Internet because **it must be centrally controlled** to avoid ambiguity and duplication

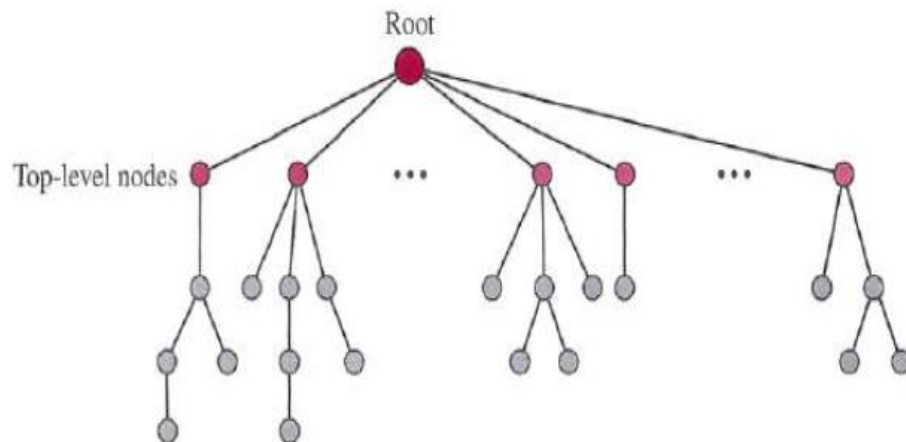
Cont...

- **Hierarchical name space:** each name is made of **several parts**
 - the **first part** can define the **nature** of the organization
 - the **second part** can define the **name** of an organization
 - the **third part** can define **departments** in the organization
- **Advantages**
 - the authority to assign and control the name spaces can be decentralized.
 - A central authority can assign the part of the name. E.g, name & nature of the organization
Rest of the name can be assigned by the organization itself



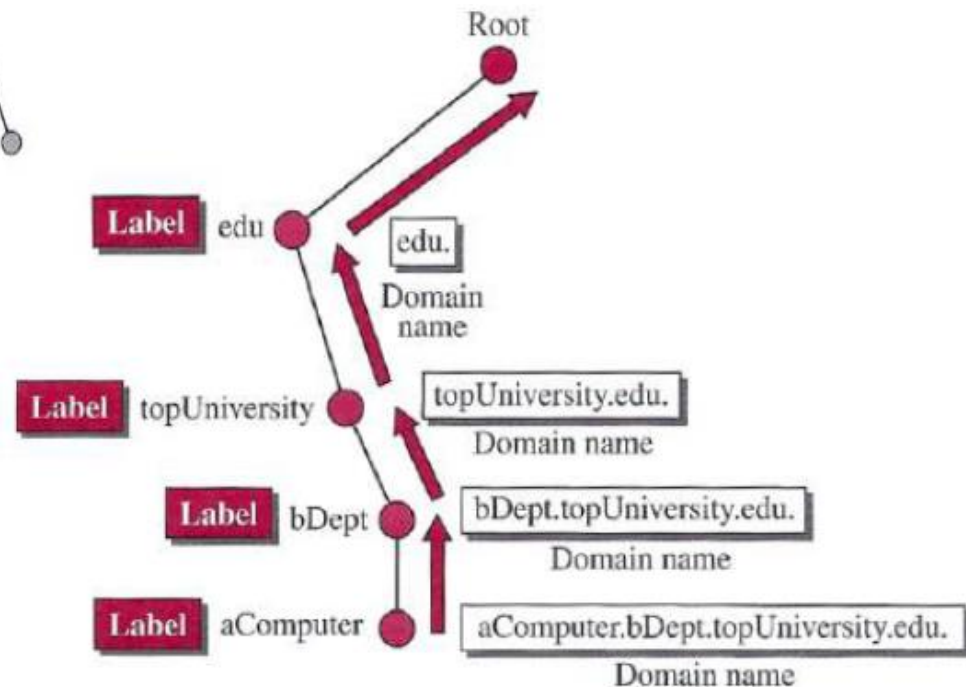
Domain Name Structure

- names are defined in an **inverted-tree structure** with the **root** at the top.




Label:

- Each node in the tree has a label, which is a string with a maximum of 63 characters.
- The root label is a null string (empty string).



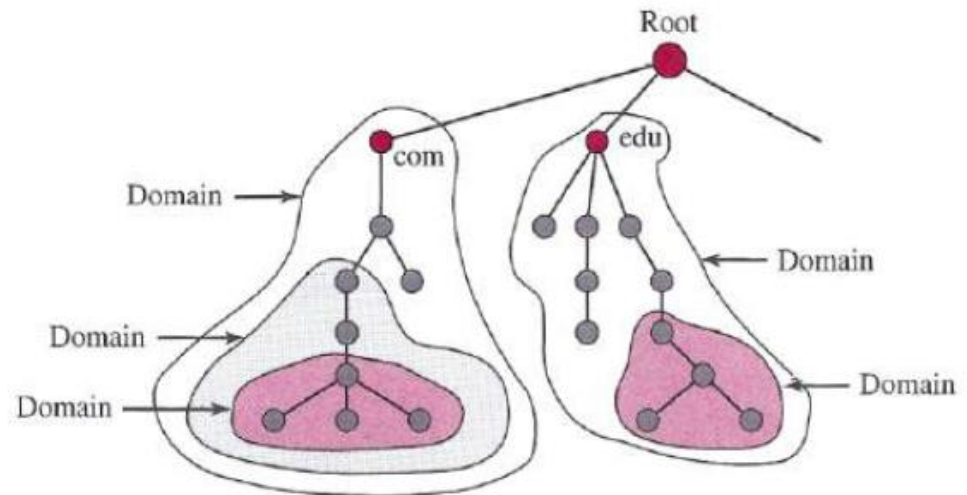
Cont...

- *Domain Name:*

- Each node in the tree has a domain name. 
- A full domain name is a sequence of labels separated by dots (.)
- The domain names are always read from the node up to the root.
- The last label is the label of the root (null).

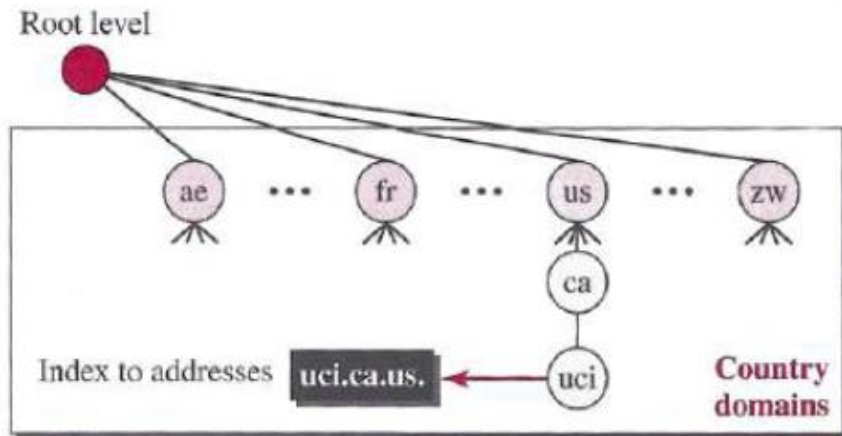
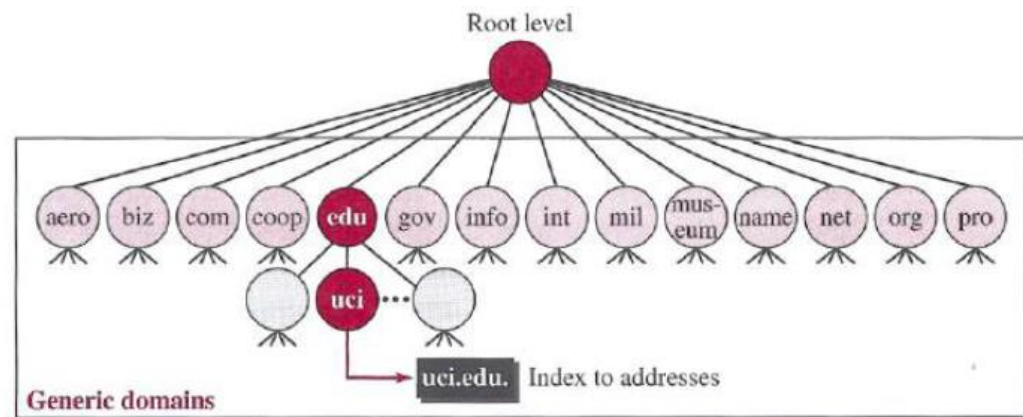
- Fully qualified domain name (FQDN):

- If a label is terminated by a null string.
- Else, it is partially qualified domain name (PQDN)



DNS in the Internet

- DNS is a protocol that can be used in different platforms.
- the **domain name space (tree)** is designed by many different ways:
 - **generic** domains
 - **country** domains

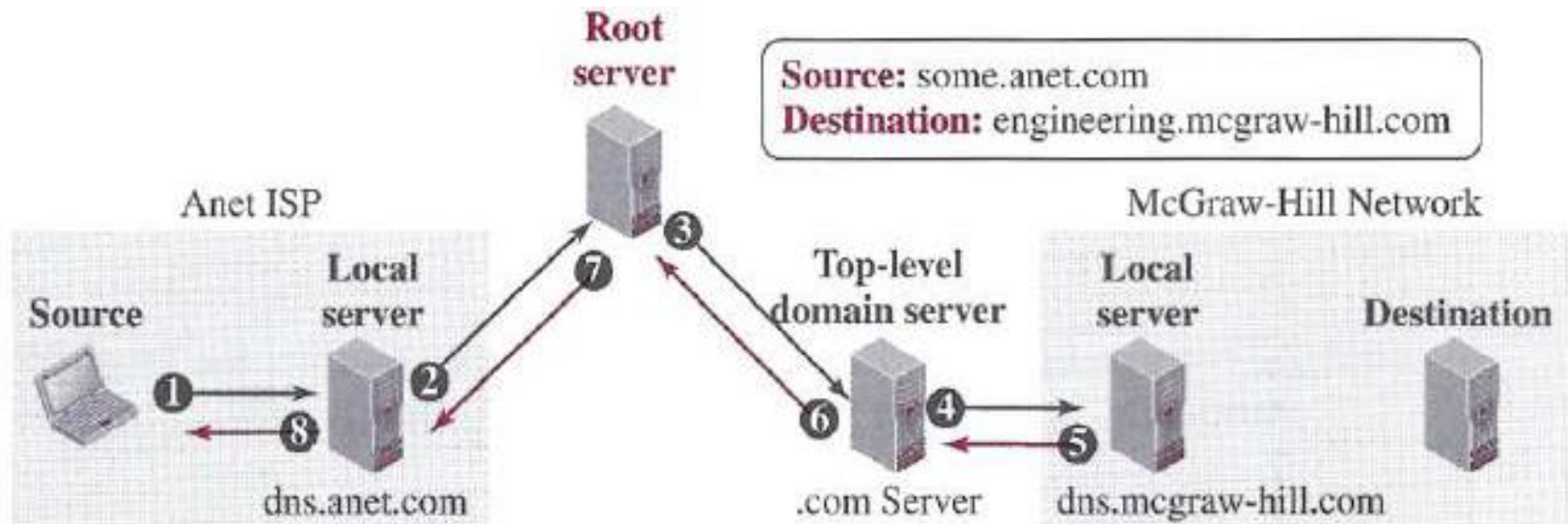


- E.g, The address [uci.ca.us](#) can be translated to **University of California, Irvine**, in the state of **California** in the **United States**.

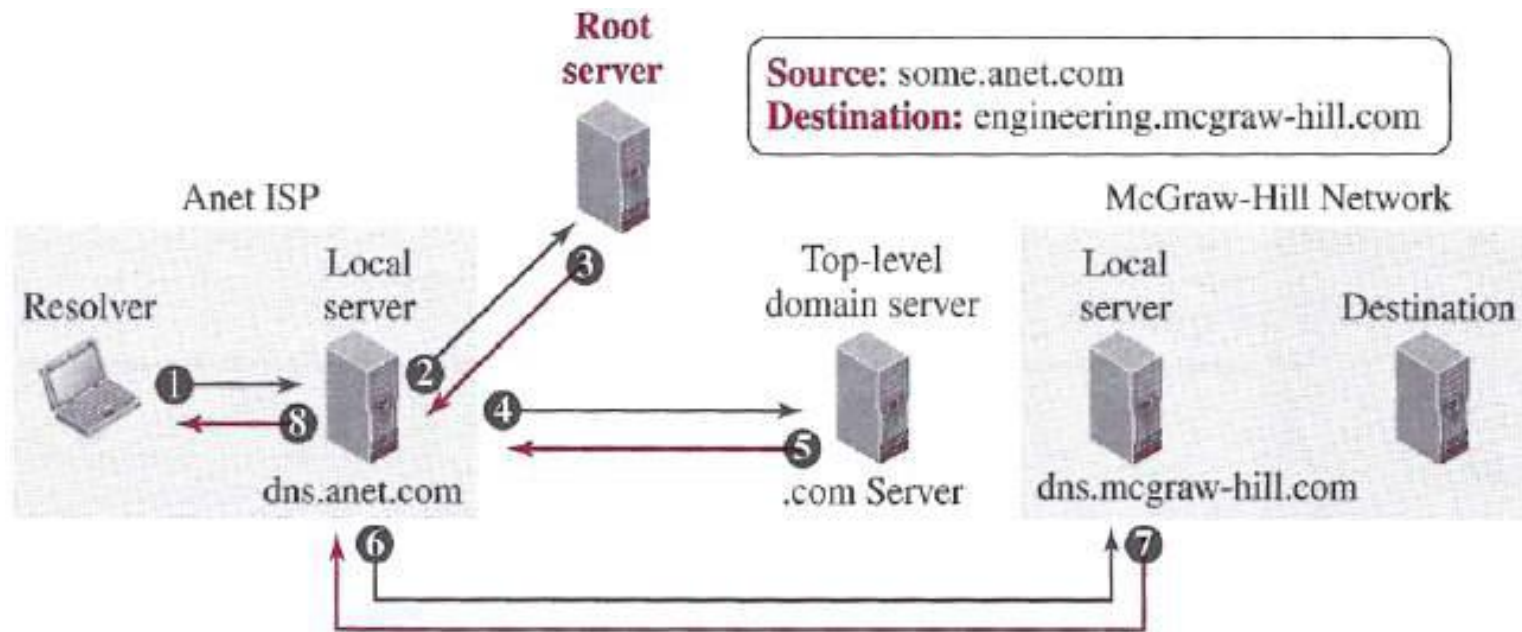
Name-Address Resolution

- Mapping a name to an address is called *name-address resolution*
- DNS is designed as a **client-server application**.
- Resolution process can be:
 - Recursive
 - Iterative

Recursive resolution



Iterative resolution



DNS Caching

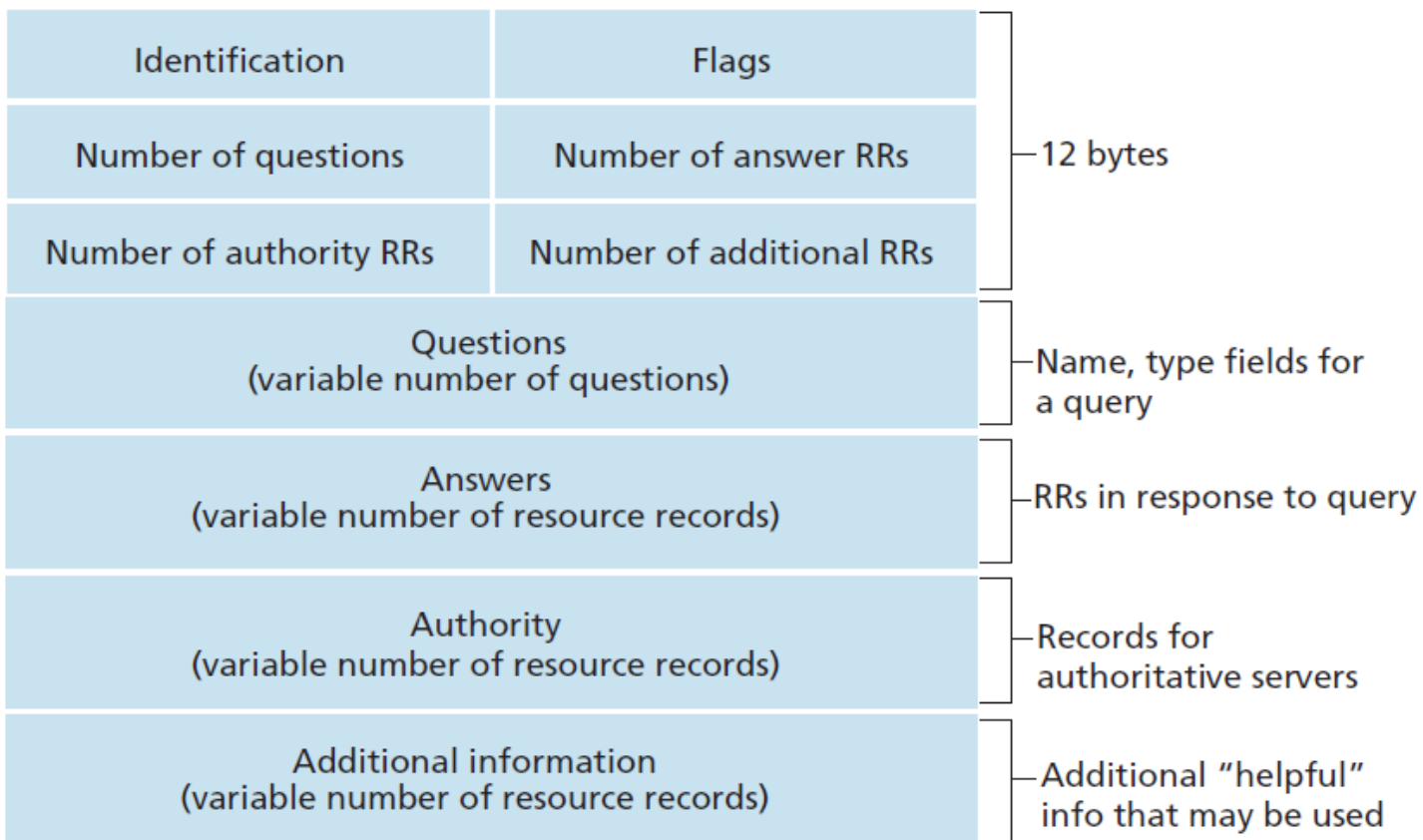


- Each time a server receives a query for a name that is not in its domain, it needs to search its database for a server IP address.
- Reduction of this search time would increase efficiency.
- DNS server handles this with a mechanism called *caching*
- Caching speeds up resolution, but it can also be problematic by sending outdated mapping.
- To counter this, TTL (*time-to-live*) based technique is used.



DNS Messages

- The **identification field** is used by the client to match the response with the query.
- The **flag field** defines whether the message is a query or response.



Thanks!