

# Domain Name System

- Ashwini Sridar Athreya

# What is DNS?

- ❖ Hosts have numbered IPv4 or IPv6 addresses that are complex and hard to remember.
- ❖ Humans cannot remember the address of every site they want to visit and prefer to use names rather than numbers.
- ❖ There was a need for a directory system that arose as the internet grew bigger and bigger.
- ❖ This directory system would, in theory, be able to map names to addresses.
- ❖ This gave rise to the Domain Name System, or DNS.

# Name Space

- ❖ Naming each host or address must be done very carefully to avoid duplication.
- ❖ There are two main ways in which a name can be assigned: Flat and Hierarchical.
- ❖ In Flat method of assignment, there is no naming system, and all the names are given one address. Any common element is purely coincidental and holds no meaning.
- ❖ Hierarchical: In this method of assignment, each name is split into multiple parts, and each part has a specific meaning.

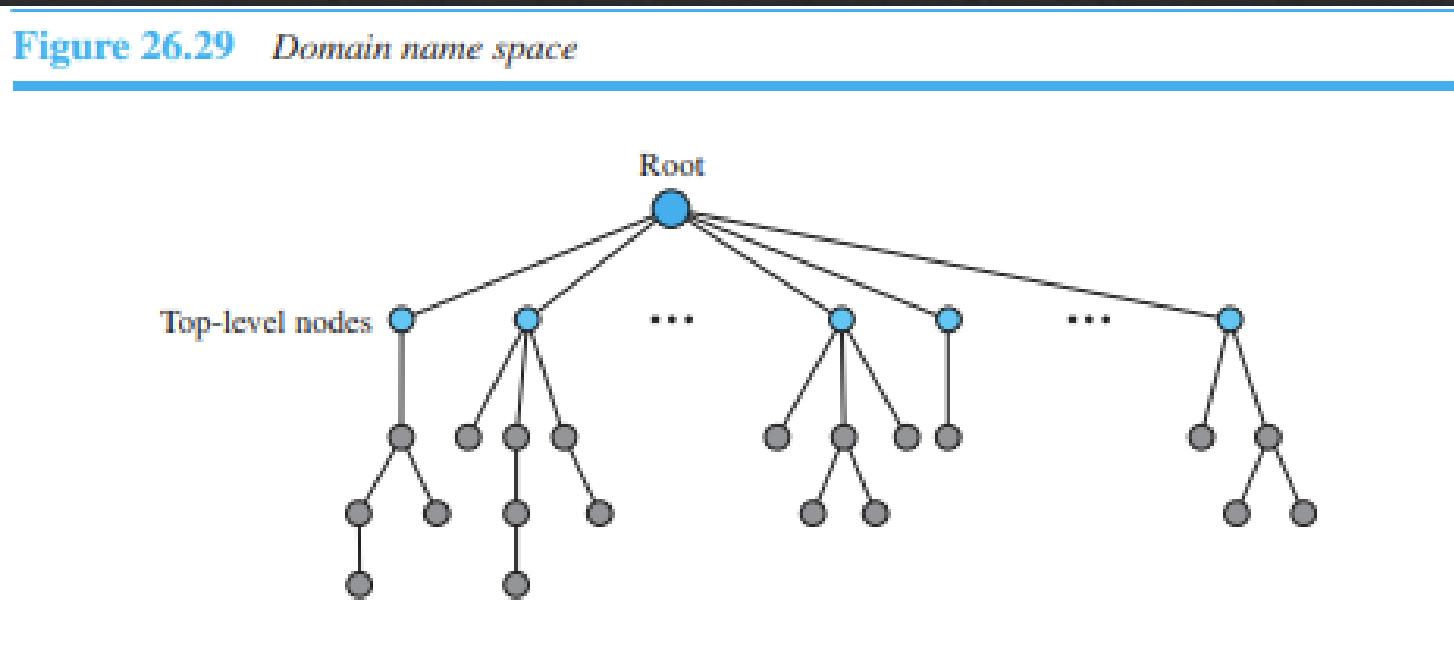
# Hierarchical Naming

- ❖ The first part of every address defines the nature of the organization.
- ❖ The second part defines the name of the organization, and the third part defines particular departments in the said organization and so on and so forth.
- ❖ Authority to assign and control the name spaces themselves can be decentralized.
- ❖ A central authority decides and assigns the part that defines the name and nature of the organization.
- ❖ The organizations then can add prefixes or suffixes to define different hosts, resources and such.

# Domain Name Space

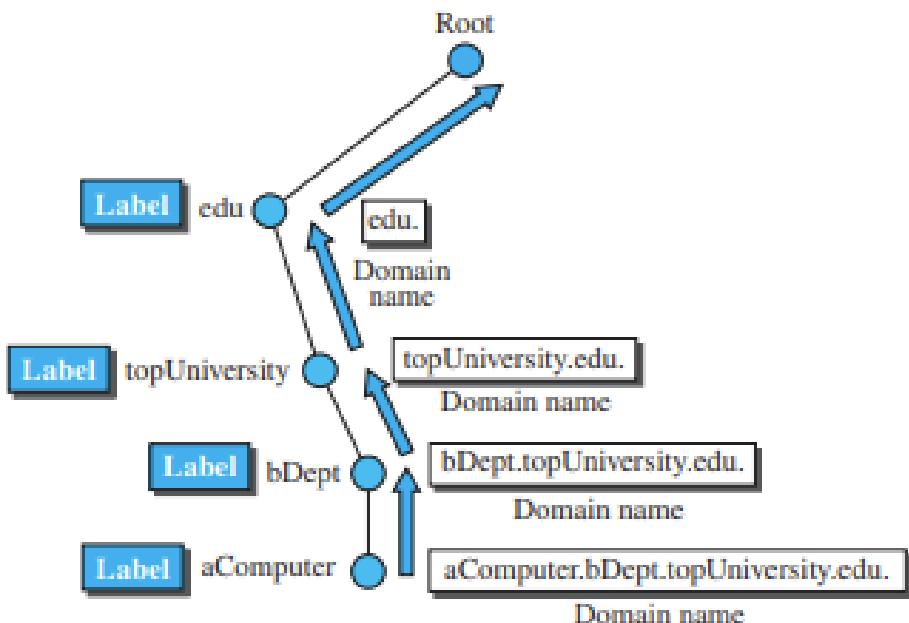
- ❖ The Domain Name Space takes the form an inverted tree, with 127 levels.
- ❖ Level zero is the root and has only a null string. It always ends with a full stop.
- ❖ Each node in the tree has a label of maximum 63 characters.
- ❖ Each child of the same node must be unique.
- ❖ If the name, or label ends with a full stop, then it is a fully qualified domain name.
- ❖ Other names will not reach the root and are called a partially qualified domain name.

# Domain Name Space



# Domain Names And Labels

Figure 26.30 Domain names and labels

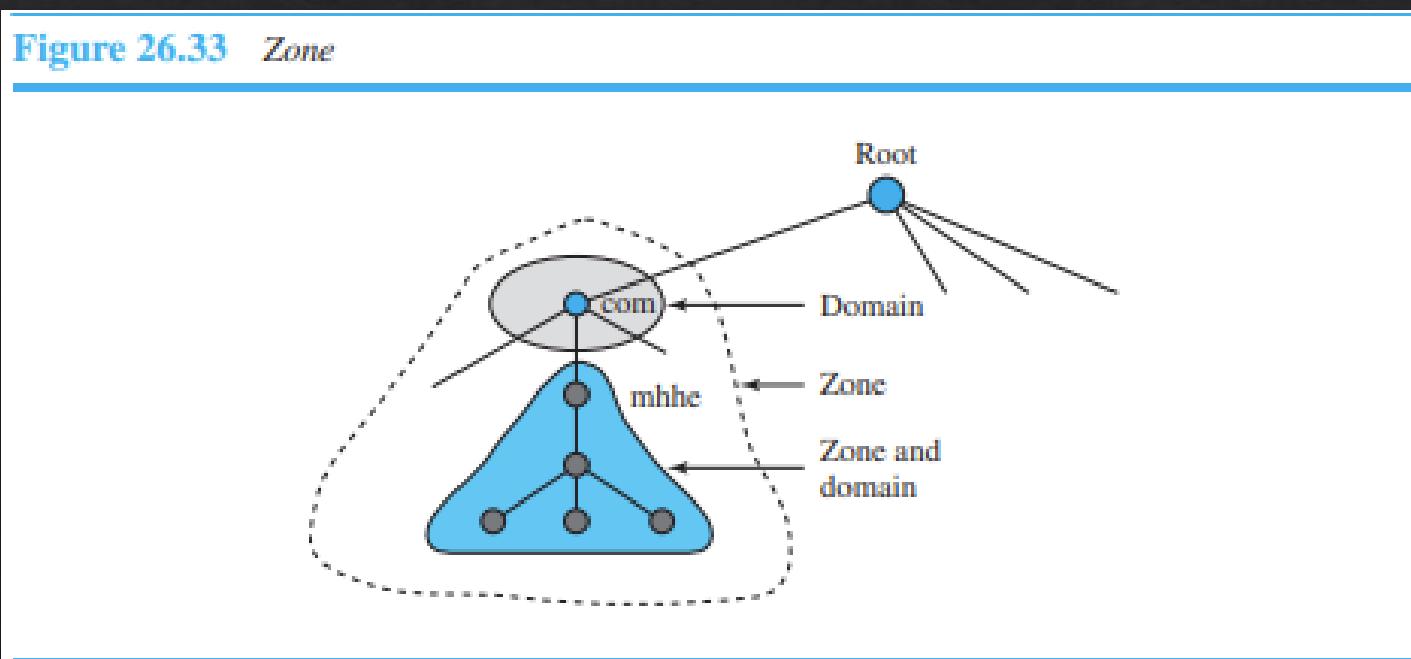


# Domains

- ❖ A node at the top of a subtree is a domain. The branch below it falls under the name of the subtree.
- ❖ If all the addresses on the internet are stored on one system, then the system may crash or lag when accessing or retrieving data.
- ❖ Separate servers are designed and implemented to store different types of addresses depending on the name of the subtree.
- ❖ Such PCs are called DNS servers and are used to store the data efficiently.
- ❖ The root server oversees all the sub servers, and all servers under a particular label are overseen by the parent.

# Zones

- ❖ A zone is what a server has authority over.
- ❖ It is a contiguous part of the tree.
- ❖ If the domain does not further subdivide then a zone and a domain are the same.



# Zone Files

- ❖ A zone file is a file that stores all the data relevant to the nodes underneath a parent.
- ❖ If there is a difference between the zone and the domain, the top servers will have references to the data.
- ❖ The actual data is stored in the lower-level servers.

# Root

- ❖ A root server has the whole tree as it's zone. It usually does not store any information about the domains underneath it.
- ❖ All relevant information is delegated to lower-level servers, and it simply maintains references.
- ❖ There are many root servers maintained at key locations over the world.

# Primary and Secondary Servers

- ❖ A primary server stores a file about the zone which it has authority over.
- ❖ Primary servers are responsible for creating, maintaining and updating the zone file.
- ❖ This file is stored on a local disk.
- ❖ A secondary server, on the other hand, neither updates nor maintains the zone file.
- ❖ It simply transfers the zone file from a primary or secondary server and stores it locally.
- ❖ In case of the primary server failing, the secondary server can pick up and continue the services being rendered.
- ❖ A server can be primary for one server and secondary to another.

# DNS and the Internet

- ❖ In the beginning, there were three main domain name spaces:
  - ❖ Generic
  - ❖ Country
  - ❖ Inverse (Deprecated)

# Generic Domains

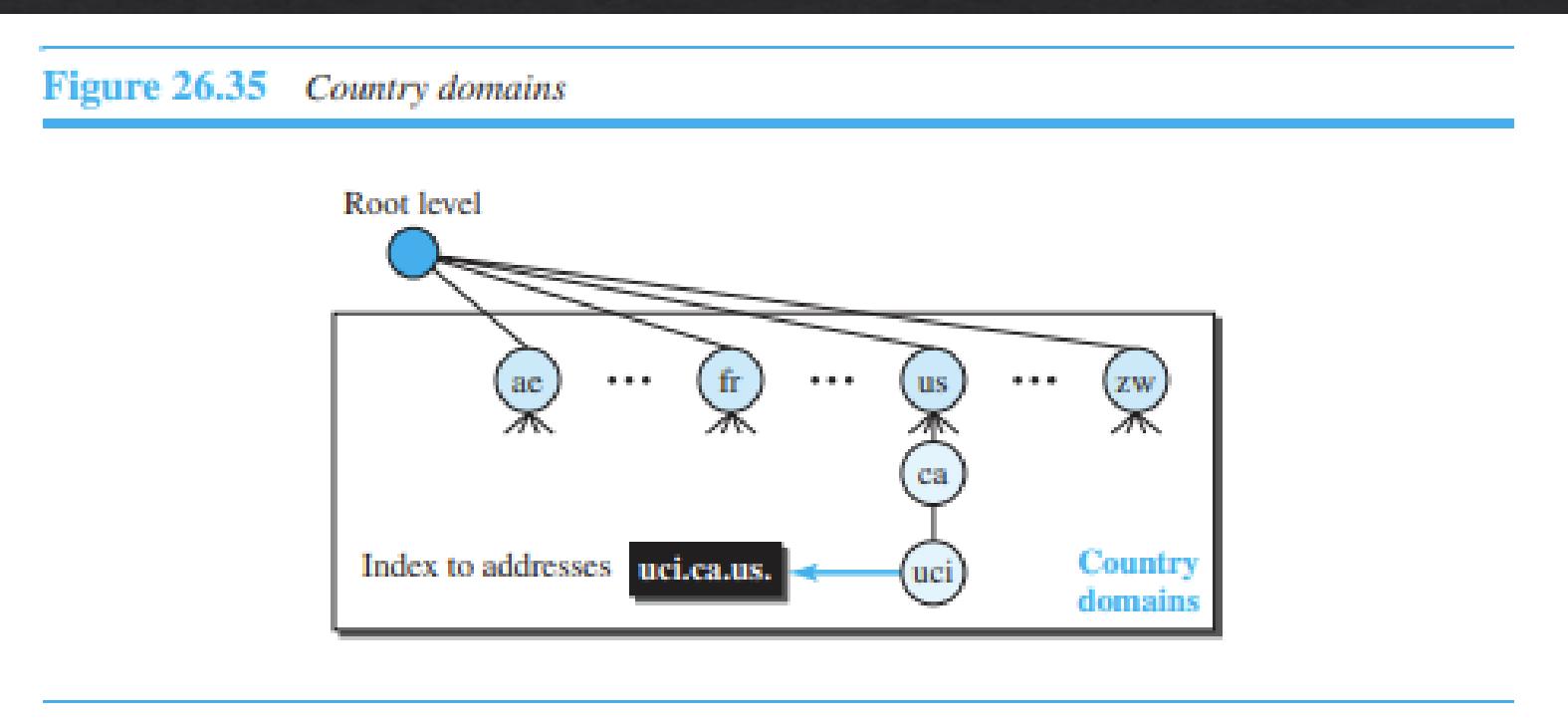
- ❖ Generic domains sorts hosts according to the behavior they exhibit, or the service they render.
- ❖ There are 14 possible labels that can be given to various organizations, as shown below.

**Table 26.12** *Generic domain labels*

<i>Label</i>	<i>Description</i>	<i>Label</i>	<i>Description</i>
aero	Airlines and aerospace	int	International organizations
biz	Businesses or firms	mil	Military groups
com	Commercial organizations	museum	Museums
coop	Cooperative organizations	name	Personal names (individuals)
edu	Educational institutions	net	Network support centers
gov	Government institutions	org	Nonprofit organizations
info	Information service providers	pro	Professional organizations

# Country Domains

- ❖ Uses two-character country domains.
- ❖ Second labels can be organizational.



# Resolution and Mapping

- ❖ Mapping a name to an address or vice versa is called Name-Address resolution.
- ❖ DNS is designed as a client-server application.
- ❖ A host that needs to map a name to its address or vice versa calls a DNS client called the resolver.
- ❖ This resolver will access the closest DNS server with a mapping request.
- ❖ If the server has the required information, then it satisfies the resolver.
- ❖ If not, it refers the resolver to other servers that may have the data it requires.
- ❖ The DNS server could also contact other servers and ask them to retrieve the necessary information.

## (contd)

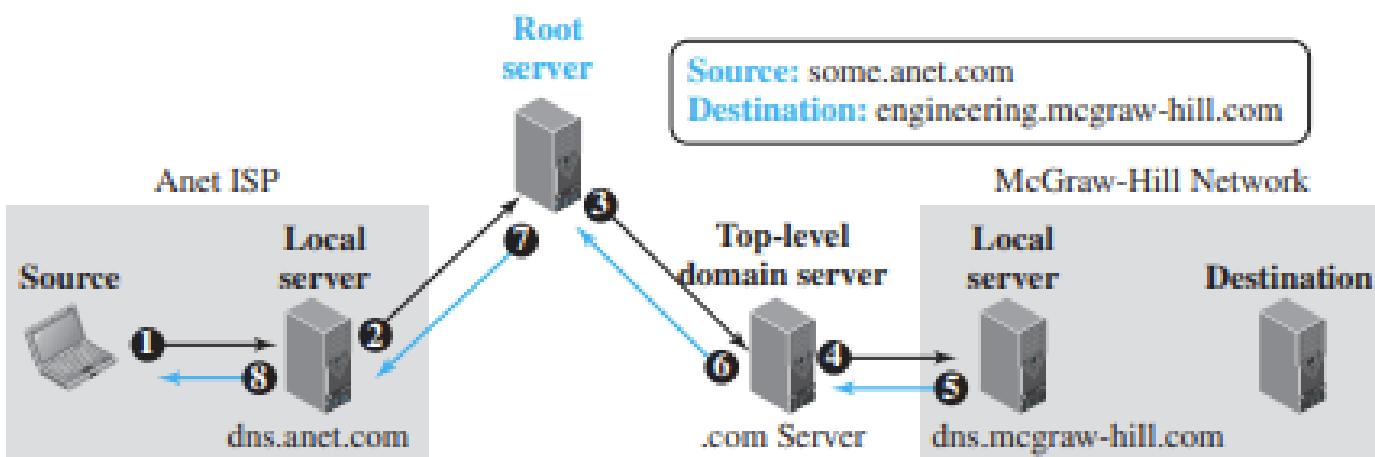
- ❖ After the resolver receives the mapping, it resolves it to check whether the information provided is legitimate and up to date. Once the check is done, the resolver will let the process that requested the information.
- ❖ There are two forms of resolution:
  - ❖ Recursive
  - ❖ Iterative

# Recursive Resolution

- ❖ In this mode of resolution, the source host calls the DNS resolver to find the IP address of the destination host.
- ❖ The resolver sends the query to the local DNS server, which may or may not contain the required IP address.
- ❖ If it doesn't, the DNS server passes on the query to a root DNS server, whose IP address is known by this local DNS server.
- ❖ The root server passes on this request to a top-level domain server, the name of which is contained in the address itself.
- ❖ The top-level server will have information about the next level of the address, and finally the IP of the destination host is received. Once received, the path the query took is reversed and reaches the initial querying process.

# Recursive Resolution

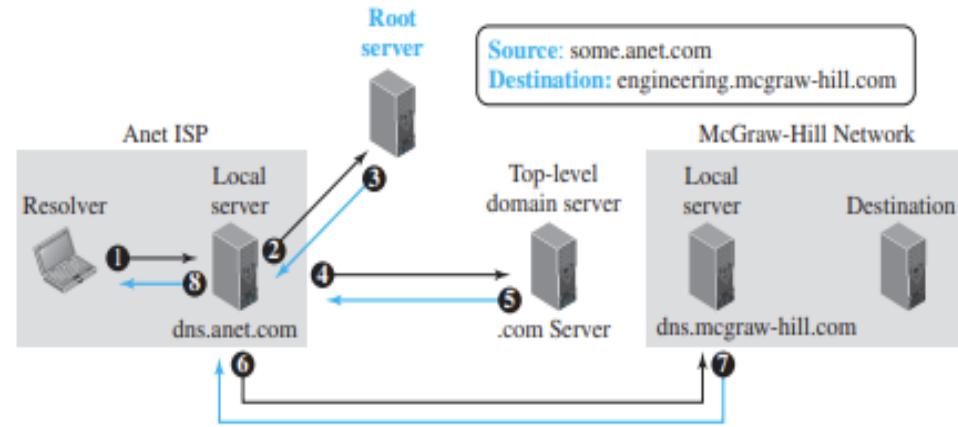
Figure 26.36 *Recursive resolution*



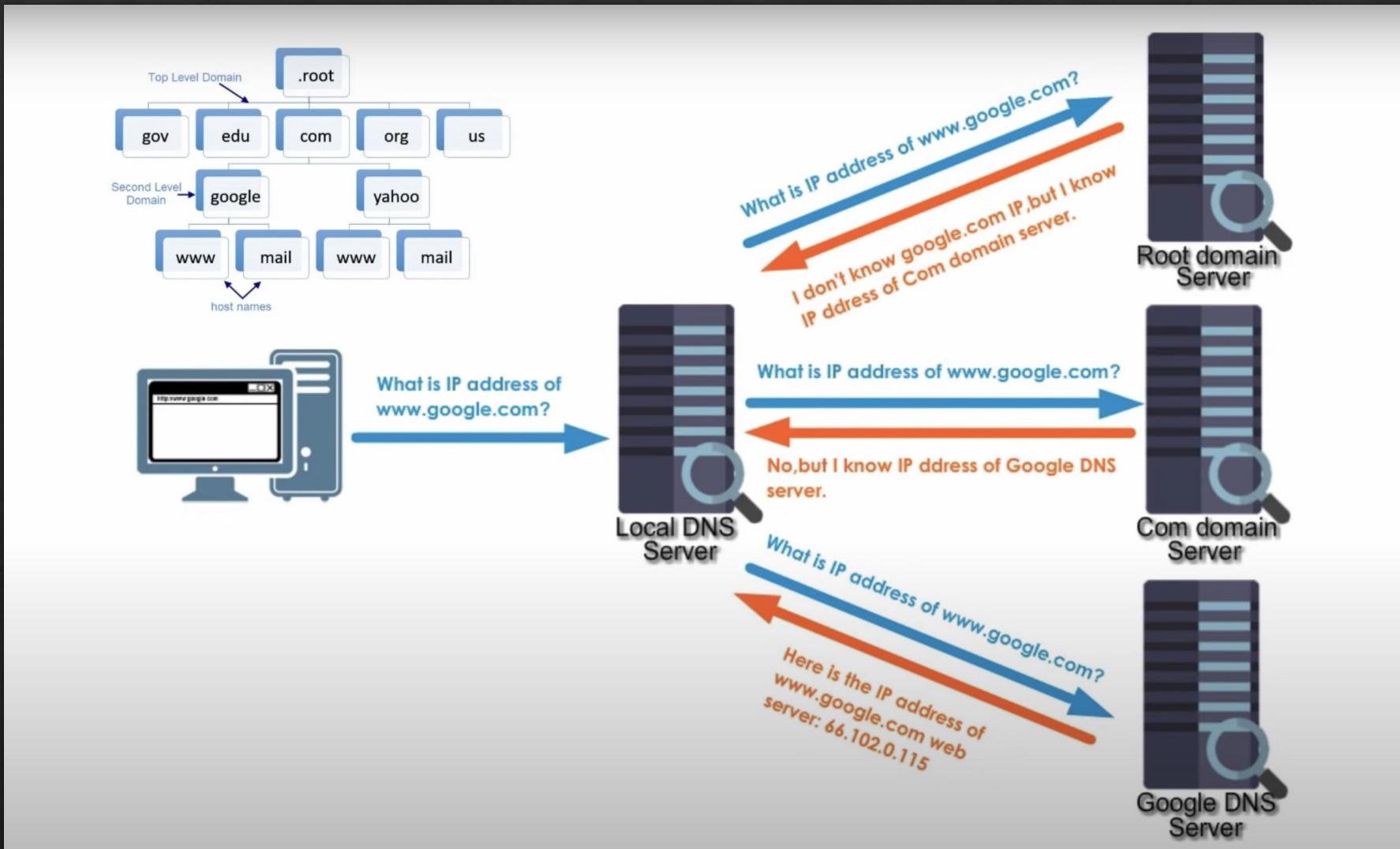
# Iterative Resolution

- ❖ In Iterative resolution, the server that initially receives the request cannot fulfill the request.
- ❖ It sends back the IP address of the next server that can be contacted for information.
- ❖ This keeps happening till the destination host is reached.

Figure 26.37 Iterative resolution



# Resolution Example



# Resource Records

- ❖ The zone information associated with each server is implemented as a set of resource records.
- ❖ A name server stores a database of resource records, which are five tuple structures consisting of (Domain Name, Type, Class, TTL, Value)
  - ❖ The domain name field is what identifies the resource record.
  - ❖ The type defines how the value should be interpreted
  - ❖ The class defines the type of network; we are only interested in the class IN (Internet)
  - ❖ The TTL defines the number of seconds for which the information is valid.
  - ❖ The value defines the information kept about the domain name.
  - ❖ There are various types, as shown in the next slide.

# Types of Interpretation

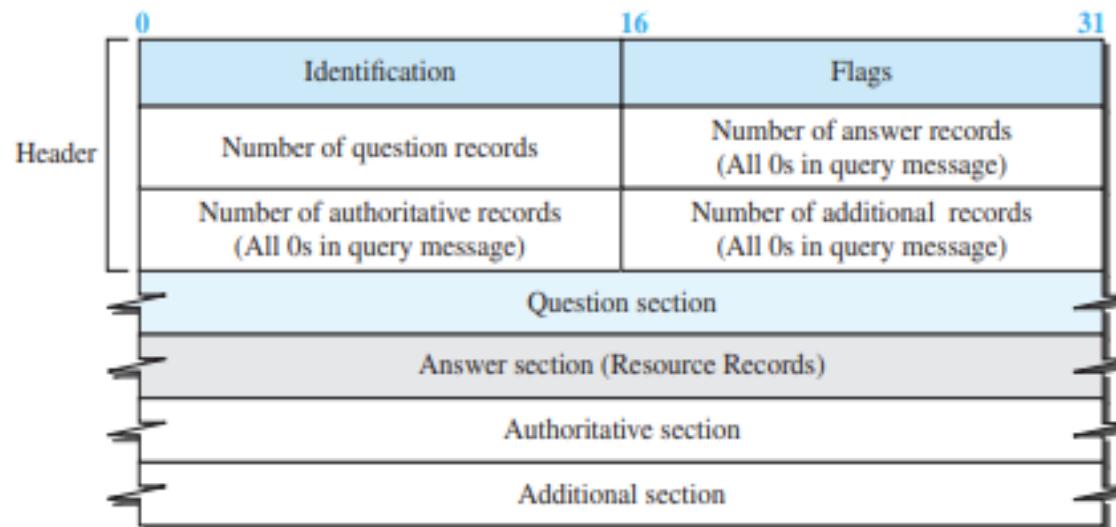
**Table 26.13** *Types*

<i>Type</i>	<i>Interpretation of value</i>
A	A 32-bit IPv4 address (see Chapter 18)
NS	Identifies the authoritative servers for a zone
CNAME	Defines an alias for the official name of a host
SOA	Marks the beginning of a zone
MX	Redirects mail to a mail server
AAAA	An IPv6 address (see Chapter 22)

# DNS Messages

- ◊ DNS uses two main types of messages to retrieve information about hosts, queries and responses.
- ◊ Both have the same format, as shown in the following figure.

**Figure 26.38** *DNS message*



**Note:**

The query message contains only the question section.  
The response message includes the question section, the answer section, and possibly two other sections.

# Encapsulation of the message

- ❖ DNS can use either UDP or TCP.
- ❖ There are certain use cases for both.
- ❖ UDP is used when the size of the response is unknown. If the size exceeds 512 bytes, it is truncated. The server then turns on the TC bit, to which the resolver responds to by opening a TCP connection. The request is repeated, and the response is obtained in its entirety.
- ❖ TCP is used when the size is a known value. If the size of the response is greater than 512 bytes, and the host has prior knowledge about it, then TCP is used over UDP.

# Registrars

- ❖ New domains are added to the DNS servers through registrars.
- ❖ A registrar is a commercial entity accredited by ICANN.
- ❖ The registrar first verifies that the requested domain name is unique, and then enters it into the DNS database.
- ❖ There is a fee charged for this process.

# DDNS

- ❖ Dynamic DNS is a mechanism by which the name server in the DNS is automatically updated with the custom domain name and the ever-changing IP addresses.
- ❖ The DDNS method is helpful in the case of dynamic IP addresses, where the IP address is mapped to a custom domain that changes frequently.
- ❖ However, in the case of a static IP address mapped to a custom domain, DDNS is not required.

# Summary

- ❖ Name Space
- ❖ Domains
- ❖ Zones and Zone Files
- ❖ Root Servers
- ❖ Primary And Secondary Servers
- ❖ DNS and the internet
- ❖ Resolution and Mapping
- ❖ Resource Records
- ❖ DNS Messages and Encapsulation