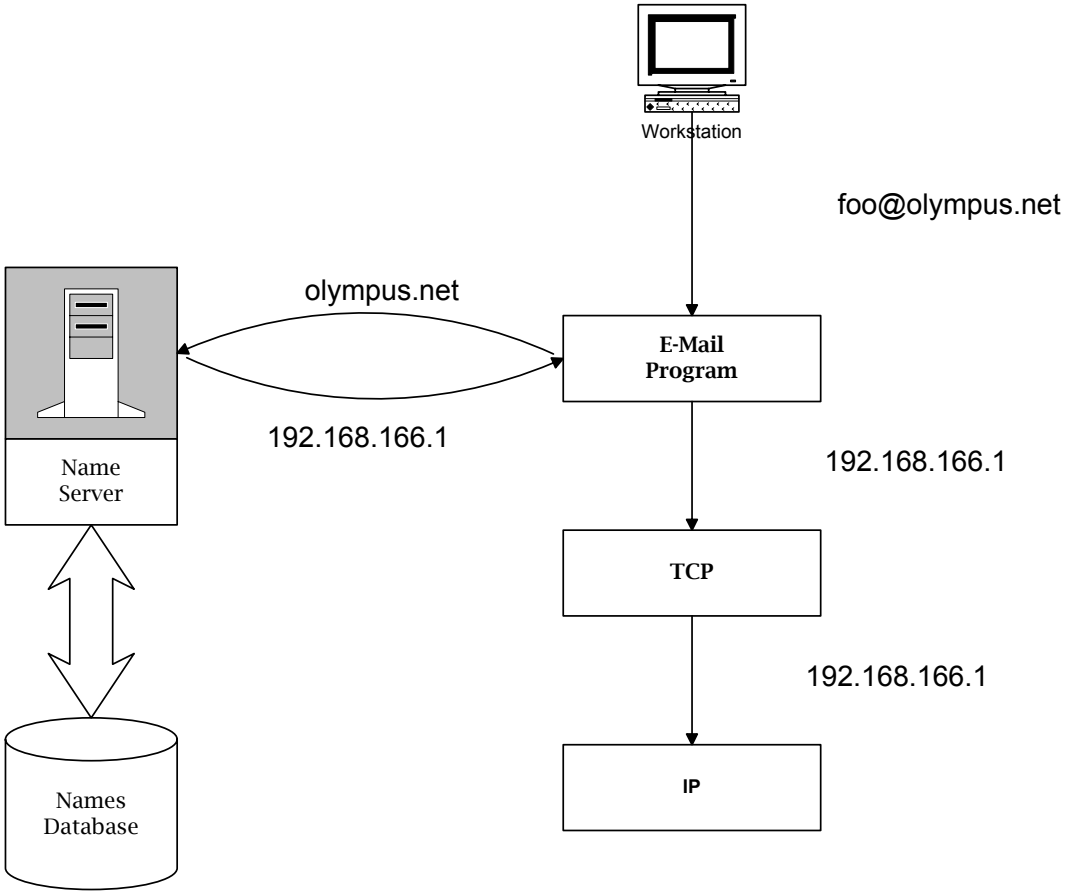
The Domain Name System (DNS)

- Each host on a network is assigned a unique name to identify it.
- For most humans host names are easier to remember than IP addresses.
- We will now discuss a scheme for assigning high-level host names to host in a network and the mechanism for mapping between the host name and its IP address.
- A **name space** defines a set of possible host names.
- The **name space** can be one of two forms:
    1. **Flat**: Names are not divisible into individual components (each name is a sequence of characters with no further structure).
    2. **Hierarchical**: Names can be broken down into individual components. UNIX file names are an example of this.
- The **naming system** also contains a mechanism for **binding host names to IP addresses**.
- A **resolution mechanism** is a function or procedure which will return an **IP address** when invoked with a corresponding **host name**.
- The resolution mechanism is implemented within a network on a **name server** which can be queried by users and programs.
- All the above functions are embodied in the **Domain Name System** (**DNS**), which has two independent aspects.
- First, the DNS specifies the name syntax and rules for naming hosts.
- Second, it specifies the implementation of a distributed computing system that efficiently maps host names to IP addresses.
- **DNS** uses a **hierarchical name space** which is implemented as a table of bindings.
- The table of bindings is broken down into **subtables** which are made available to name servers over the network.

- The diagram given illustrates the sequence of events in the host name to IP address translation.



TCP/IP Internet Domain Names

- **DNS** uses a **hierarchical name space** for Internet objects known as **domain names**.
- DNS names are processed from left to right with the naming components (labels) separated by periods.
- Thus, the domain name:

  milliways.bcit.ca

  contains **three labels**: milliways, bcit, ca.
- Any suffix of a label in a domain name is also known as a domain. The **lowest level** domain in the above is **milliways.bcit.ca** (the domain name for the Data Comm web server).
- The **second level** domain is **bcit.ca** (the domain name for BCIT).
- The **top level** is **ca** (the country code for Canada).
- Most users of the domain technology adhere to the hierarchical labels established by the official Internet domain system.
- This is a comprehensive and flexible scheme which can accommodate a wide variety of organizations, and allows each group to select between geographical or organizational naming hierarchies.

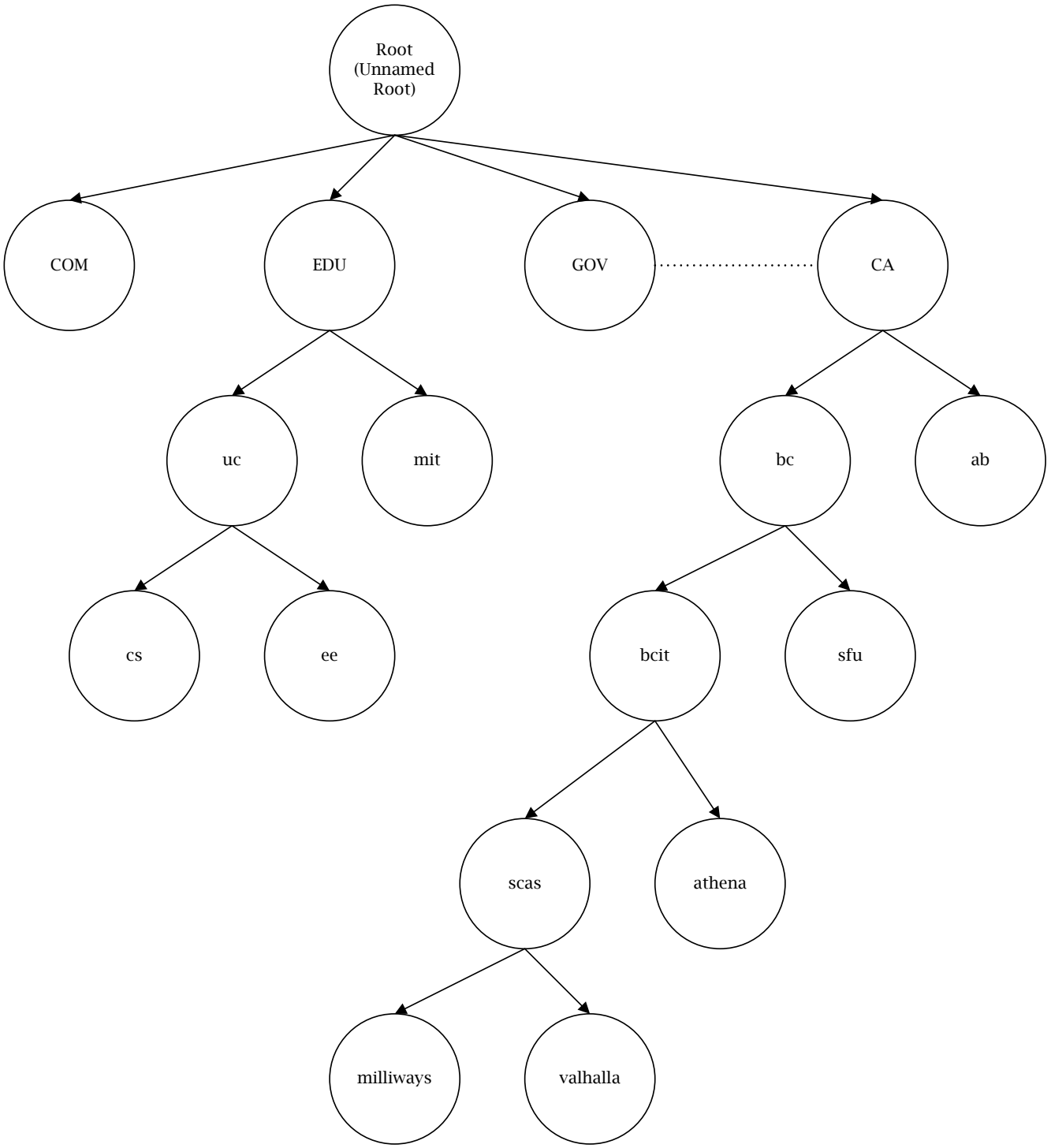
- The Internet authority has established a scheme to partition its top level into domains listed in the table below.

Top Level Domain name	Meaning
COM	Commercial organizations
EDU	Educational Institutions
GOV	Government Institutions
MIL	Military agencies
NET	Major network support centers
ORG	Organizations other than those listed
ARPA	Major research arm of the US defense dept. (obsolete)
INT	International organizations
Country Code	Each country (**ca** for **Canada**)

- The hierarchy is partitioned into **subtrees** (subtables) called **zones**.

- Each **zone** corresponds to an **administrative authority** that is responsible for that portion of the hierarchy.

- The diagram shown illustrates this hierarchical tree structure.


Root
(Unnamed
Root)
COM
EDU
GOV
CA
uc
mit
bc
ab
cs
ee
bcit
sfu
scas
athena
milliways
valhalla

- The host name **milliways** in the School of Computing has the official **domain name**:

  milliways.scas.bcit.bc.ca

- This host name was approved and registered by the **local network manager** in the School of Computing.

- The **local network manager** had previously obtained permission from **the BCIT network manager** for the **subdomain**:

  scas.bcit.bc.ca

- The **BCIT network manager** retains authority for the **bcit.bc.ca domain**.

- The **Internet authority** retains control of the **ca domain**.

- Thus, the School of Computing manages its own **department-level zone**.

- The relevance of the **zone** is that it corresponds to the fundamental **implementation unit** in the DNS **name server**.

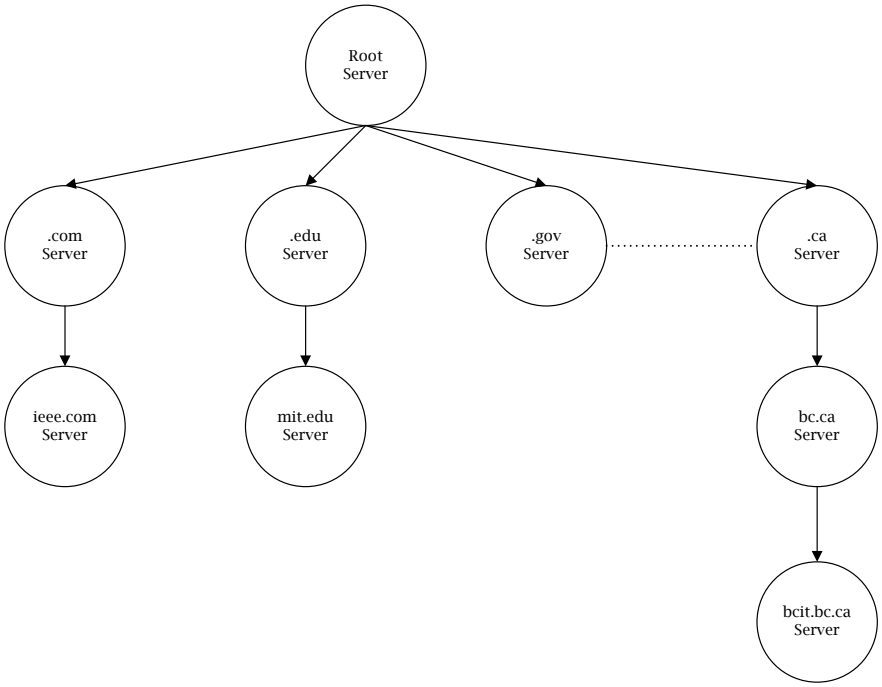
- The **information** contained in each **zone** is implemented in one or more **name servers**.

- Each **name server** is a **program** which can be accessed from other hosts on the Internet.

- **Clients** from each host can send **queries** to the **name servers**, who in turn respond with the requested information.

- The response could be the **final answer** (the information requested by the client) or a **pointer** to **another server** that the client should query next.

- In this way, the DNS is implemented as a hierarchy of name servers rather than a hierarchy of domain names. This is illustrated in the diagram below.



- Each zone is usually implemented in two or more name servers for the purposes of **redundancy**.

- Each name server implements the zone information as a large set **of resource records** which are **name-to-value** (IP) bindings.

- A resource record is a **5-tuple** configured as:

  {Name, Value, Type, Class, TTL}

- **Name** is the **Domain Name** to which the resource record refers.

- **Value** is a fie**ld which can provide the IP address or some other information based on the** Type field.

- **Type** specifies the how the **Value** field should be interpreted. **Type = A** means that **Value** is an **IP address**.

- Thus, **Type A** provides the typical **host name** to **IP address** mapping.

- Other Type parameters are:

NS: Value provides the domain name for a host that is running a name server that knows how to resolve names within the specified domain.

CNAME: Value provides an alias for a particular host.

MX: Value provides the domain name for a host that is running a mail server that accepts messages for a specified domain.

- The **Class** field specifies the data's class. The only widely used class is the one used by the **Internet**, denoted by **IN**.

- The **TTL** (Time To Live) field is an integer that specifies the number of seconds information in this resource can be **cached**.

- It is used by clients who have requested a name binding and want to cache the results. When the TTL expires, the server must evict the record from its cache.

Domain Name Resolution

- **Domain name resolution** proceeds **top-down**, starting with the root name server and proceeding to servers located at the leaves of the tree.

- Clients can either contact name servers one at a time (**iterative resolution**), or ask the name server system to perform the complete translation (**recursive resolution**).

- In either case, the client software forms a **domain name query** that contains the name to be resolved, the class of the name, the type of answer desired, and a code that specifies whether the name server should translate the name completely.

- When a domain name server receives the query, it checks to see if the name lies in the subdomain for which it is an authority.

- If so, it translates the name to an address according to its database, and appends an answer to the query before sending it back to the client.

- If the name server cannot resolve the name completely, it checks to see what type of resolution the client requested.

- If the client requested **recursive resolution** (complete translation) the server contacts a domain name server that can resolve the name and returns the answer to the client.

- If the client requested **iterative resolution** (non-recursive translation), the name server cannot supply an answer and generates a reply that specifies the next name server that the client should contact to resolve the name.

- A client must know how to contact at least one name server. To ensure that a domain name server can reach others, the domain system requires that each server know the address of at least one root server.

- Domain name servers use a well-known port for all communication, so clients know how to communicate with the server once they know the IP address of the machine on which the server executes.

- The example diagram shown illustrates the steps in a typical name resolution.