Naming and name resolution mechanisms

- Names, Identifiers, and Addresses
- Flat Naming
- Structured Naming
- Attribute-Based Naming
Naming

- Naming system itself may be distributed – distribution criteria affects the efficiency and scalability of naming systems
- We will address various naming mechanisms and name resolution mechanisms
Names

- A string of bits or characters that is used to identify an entity (hosts, printers, ...)
- An entity can be operated on through an access point
- A location independent entity name is often much easier and more flexible to use
Names

- Types of names – addresses, identifiers, human friendly names

- Identifiers
  - An identifier refers to at most one entity
  - Each entity is referred to by at most one identifier
  - An identifier always refers to the same entity

- Human friendly names
  - Targeted to be used by humans represented as character strings
  - Not suited for highly mobile entities
Addresses

- The name of an access point of an entity is called an address of that access point or entity.
- An address is treated as a special kind of name:
  - An entity can offer more than one access points (a person having more than one tel. no.) - which address to use as a reference?
  - An entity may easily change an access point (dynamically generated IP addresses for laptops), or an access point may be reassigned to a different entity.
Naming versus Locating Entities

Naming services usually maintain a direct mapping between human friendly names and the addresses of the entities – any address change requires new mapping.

a) Direct (single level) mapping between names and addresses
b) T-level mapping using identities (separating naming from locating entities)
Naming Types

- **Flat Naming**
  - Systems need to resolve an identifier to the address of its associated entity
  - Identifiers are usually random bit strings – no information to locate the access point of its associated entity

- **Structured Naming**
  - Organized in a name space (naming graph)
  - A node represents a named entity and the label on an edge represents the name under which that entity is known

- **Attribute-Based Naming**
  - Entities are described by a collection of (attribute, value) pairs
Flat Naming

- How flat names can be resolved – locating an entity only from its identifier
  - Broadcasting
  - Multicasting
  - Forwarding Pointers
  - Home-Based Approaches
  - Hierarchical Approaches – next lecture
Broadcasting

- A message containing the identifier is broadcasted to each m/c and each m/c is requested to check whether it has that entity (who the owner is of a given IP address)
- Only the machines that can offer the (Ethernet) address of that access point (Internet Address Resolution Protocol) reply
- Broadcasting becomes inefficient when the network grows – the bandwidth wasted by requests and too many hosts may be interrupted by requests they cannot respond
Multicasting

- Only a restricted set of hosts receive the request
- Other usage of multicasting
  - Locate entities in a point-to-point network
  - General location service for multiple entities
  - Locate the nearest replica
Forwarding Pointers

- When an entity moves from A to B, it leaves behind a reference to its new location.
- Forwarding pointers using proxy-skeleton pairs – skeleton for entry and proxy for exit.

![Diagram showing forwarding pointers and proxy-skeleton pairs](image-url)
Forwarding Pointers – contd.

- Short-cut a chain of proxy-skeleton pairs
- Redirecting a forwarding pointer, by storing a shortcut in a proxy

(a) Invocation request is sent to object
Skeleton at object's current process returns the current location

(b) Skeleton is no longer referenced by any proxy
Client proxy sets a shortcut
Forwarding Pointers – contd.

**Advantages**
- Simple – as soon as an entity has been located, a client can look up the current address by following the chain of forwarding pointers.

**Disadvantages**
- If no special measures are taken, a chain can become so long that locating an entity is expensive.
- All intermediate locations in a chain will have to maintain their part of the chain of forwarding pointers as long as needed.
- Vulnerability to broken links – as soon as a forwarding pointer is lost the entity cannot be reached any more.
Home-Based Approach

- A home location keeps track of the current location of an entity – acts as a *care-of address* of the moved objects
- The principle of Mobile IP
Disadvantages

- Increased communication latency – a client has to communicate with the home first which may be located in a different location
- Home has to be located in a fixed location so that its address is always known to the clients
- The root may be required to handle so many look up and update requests that it will become a bottleneck – alternative?
Scalability Issue

- Alternative - uniformly place subnodes of a partitioned root node across the network covered by a location service
Summary

**Naming**
- Names, Identifiers, and Addresses
- Flat Naming
  - Broadcasting
  - Multicasting
  - Forwarding Pointers
  - Home-Based Approaches
  - Hierarchical Approaches – will be discussed later
- Structured Naming
- Attribute-Based Naming