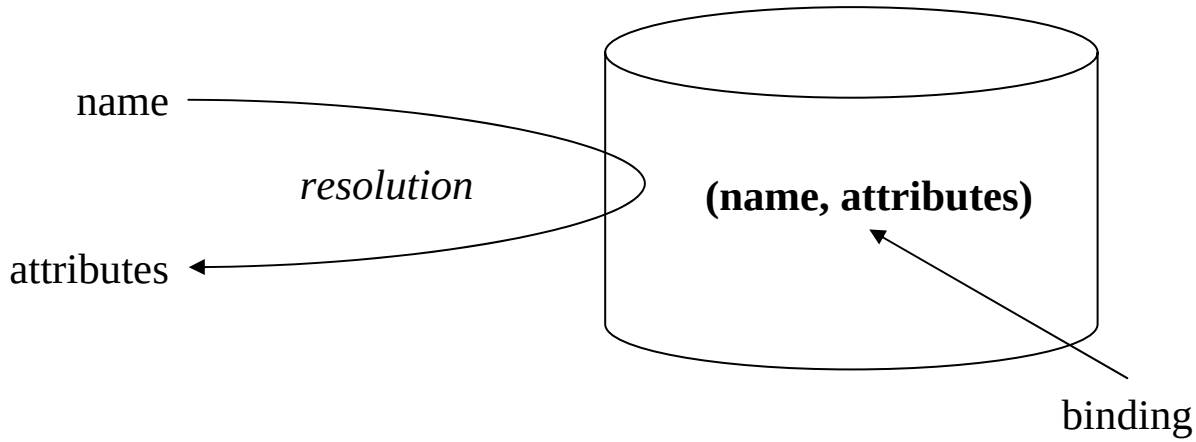
Naming and Directory Services

1. General Concepts
2. Examples
   - Domain Name Service (DNS)
   - X.500/LDAP
   - Corba Naming Service

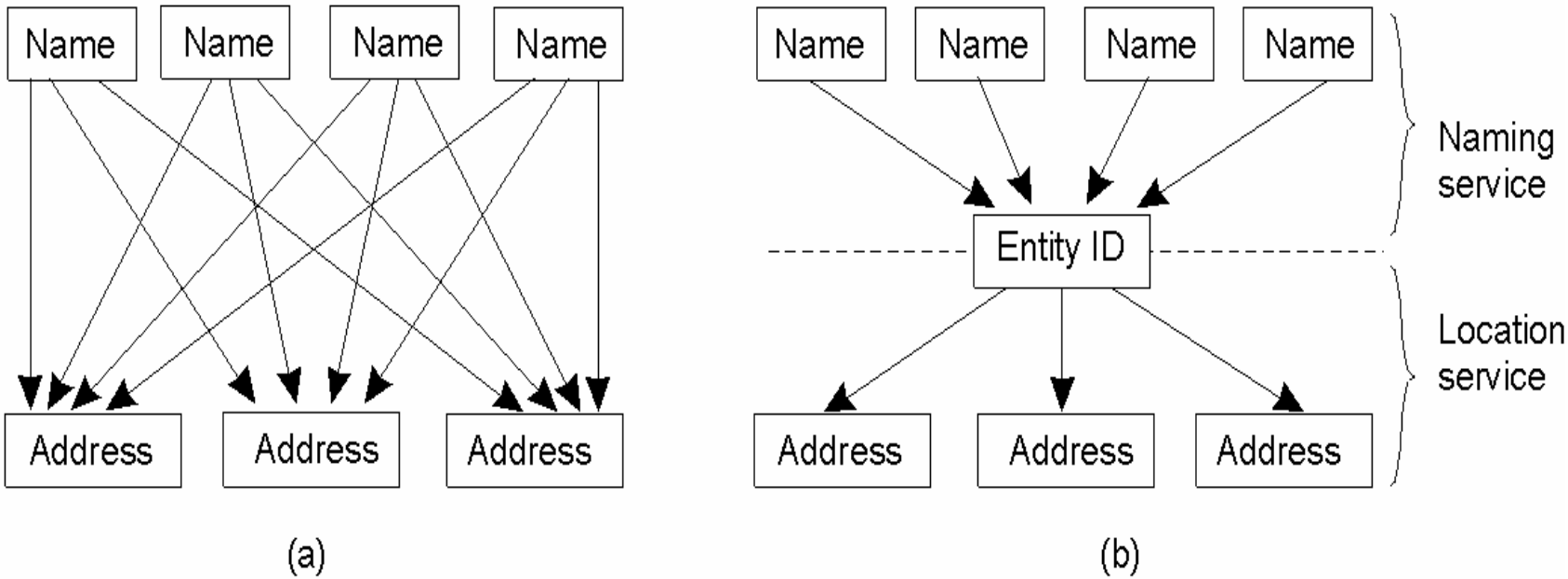
1. General Concepts

Structure:



- name/attributes may refer to a person, device, object, service, etc.
- types of resolution:
    - search by name (white pages)
    - search by attributes (yellow pages)

Naming versus Locating Entities



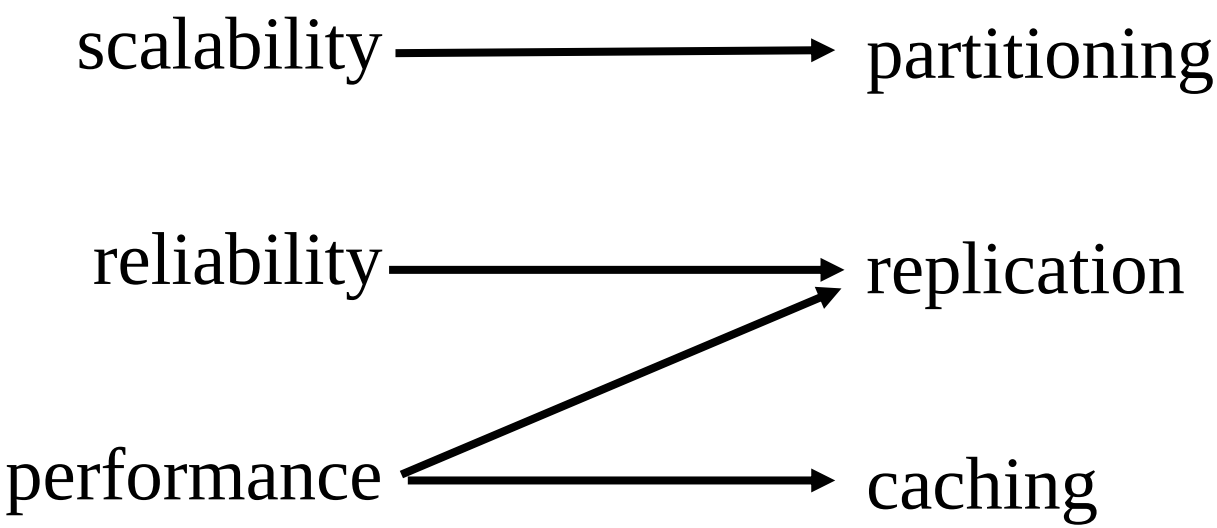
a) Direct, single level mapping between names and addresses.

b) Two-level mapping using identities.

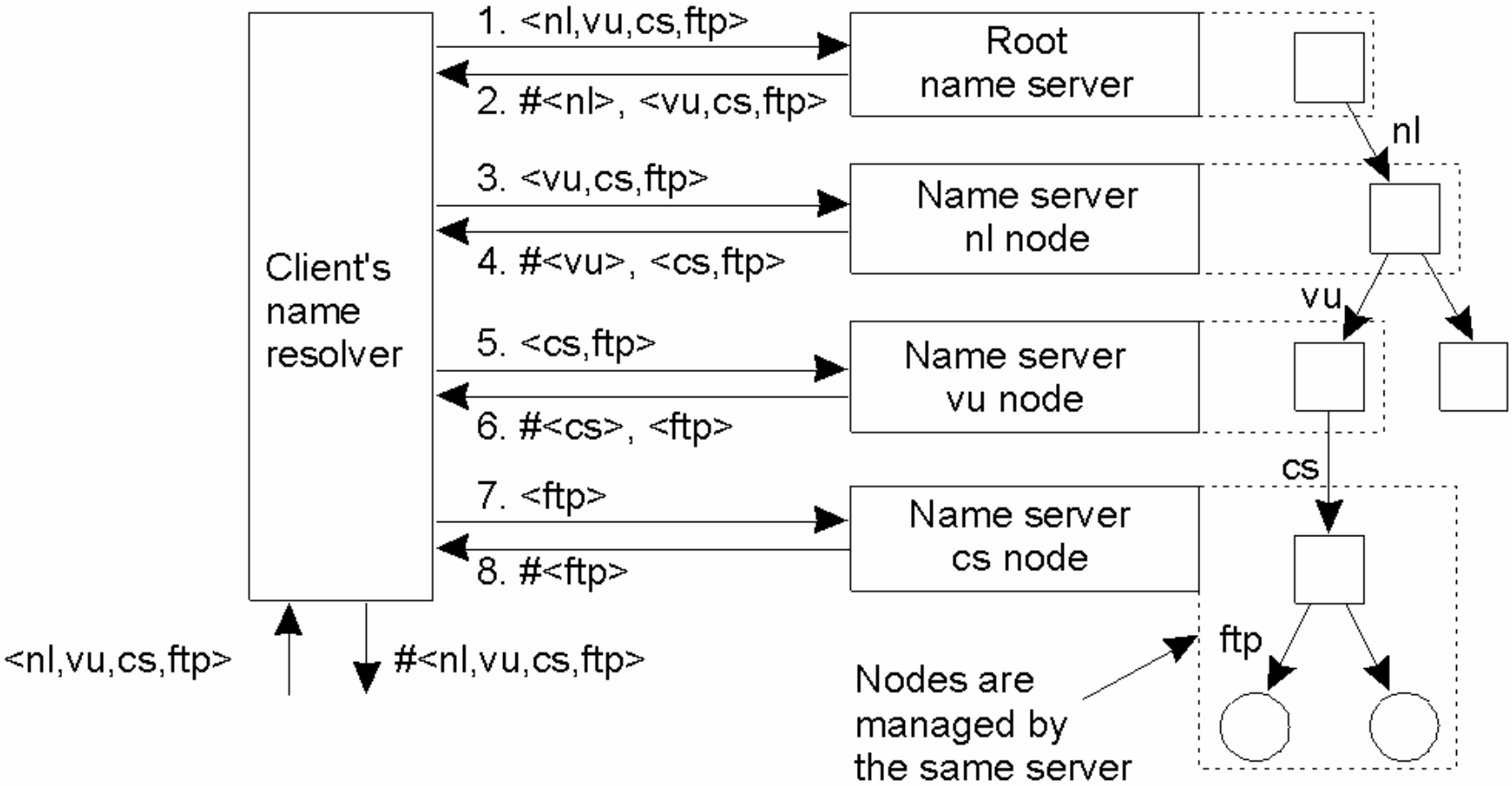
Design Issues

- Binding information is relatively static
  - Optimized for high-volume reading
- Principal operations are query/change
  - Need not support complex database operations
  - May provide a simple query API
- Information tailored for a specific purpose
  - Predefined/fixed schema

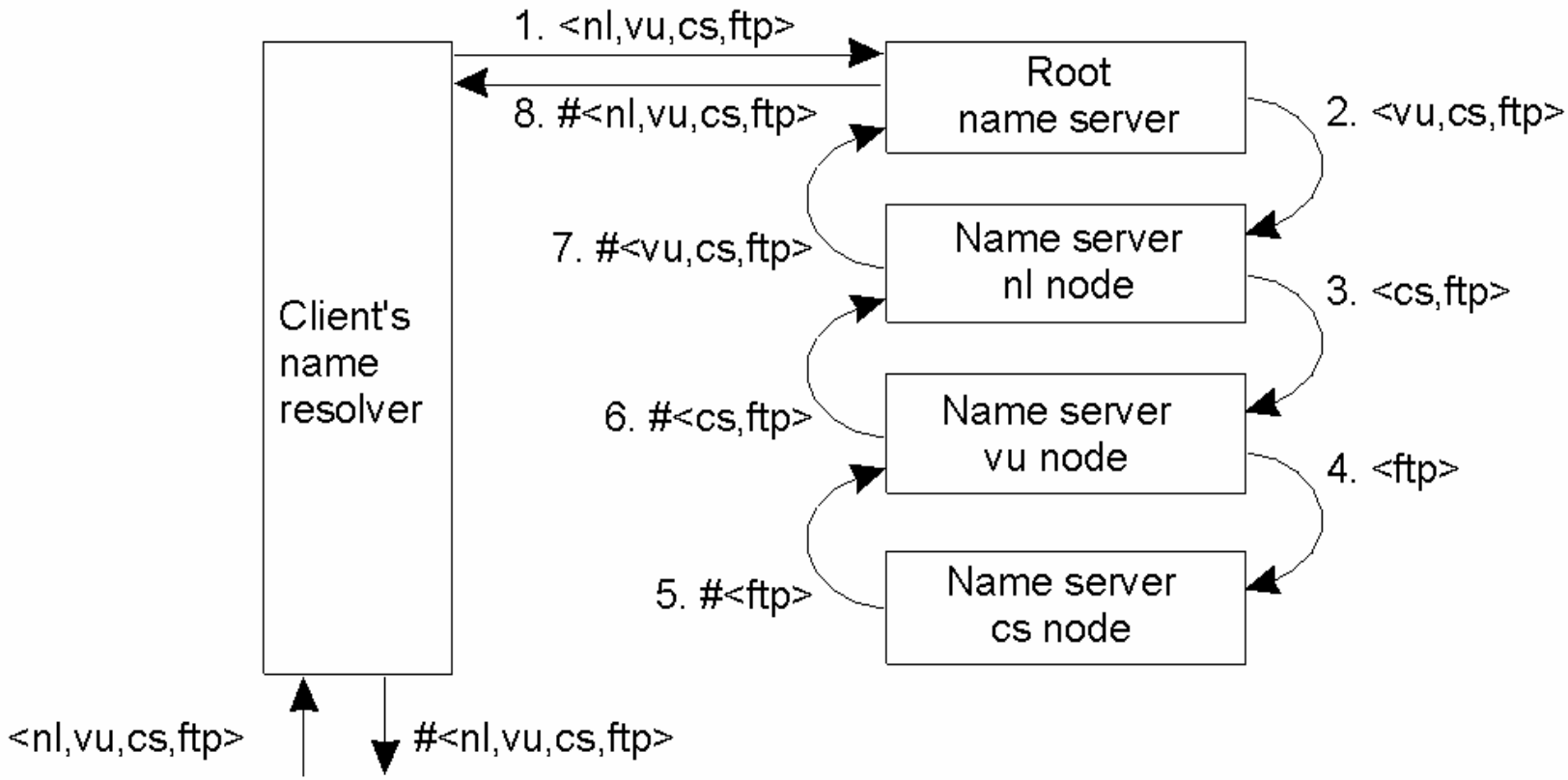
Design Issues



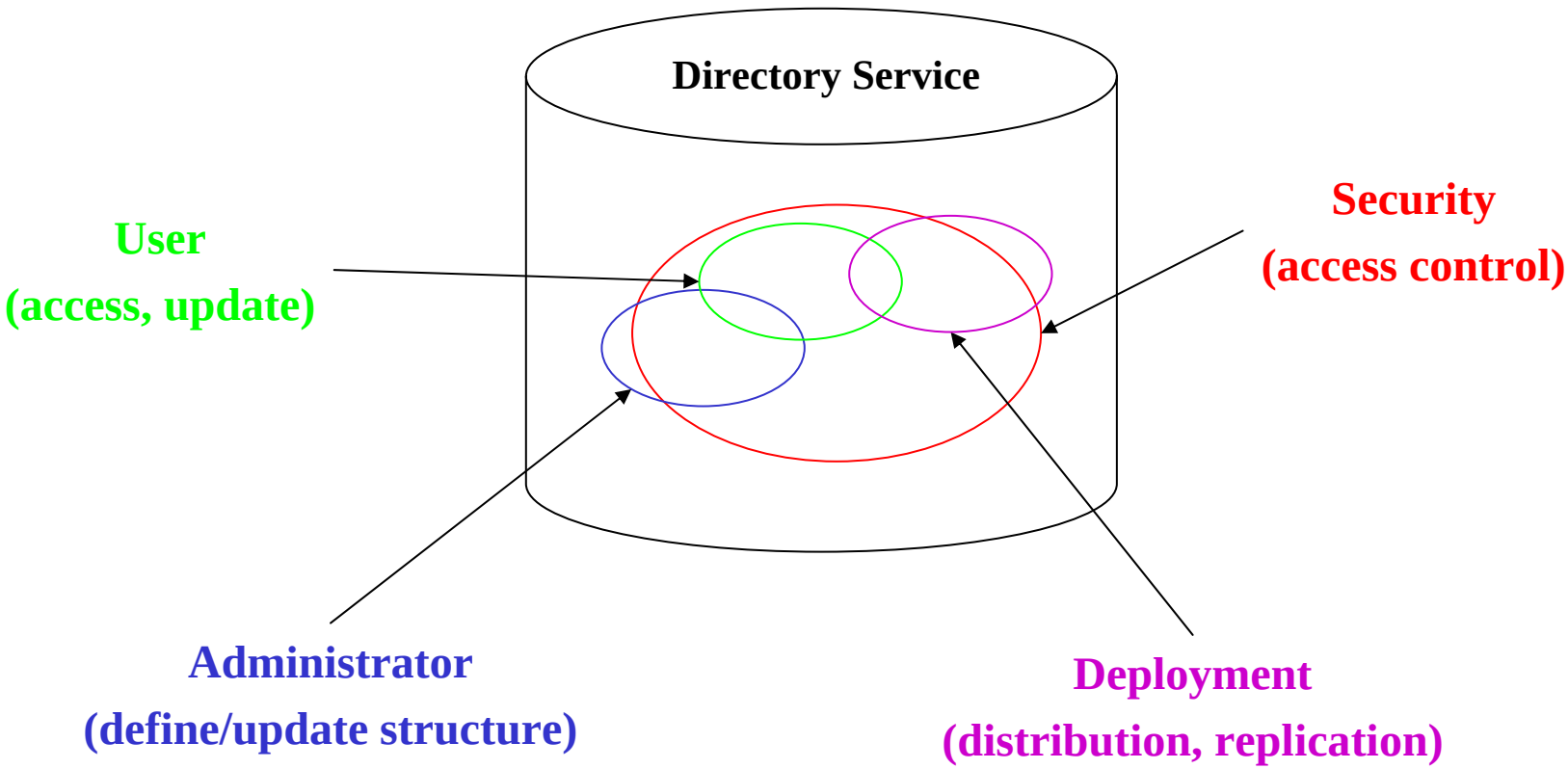
Iterative Name Resolution



Recursive Name Resolution



Information Views



2. Examples

- Domain Name Service (DNS)

  www.cs.vt.edu ⟶ IP address

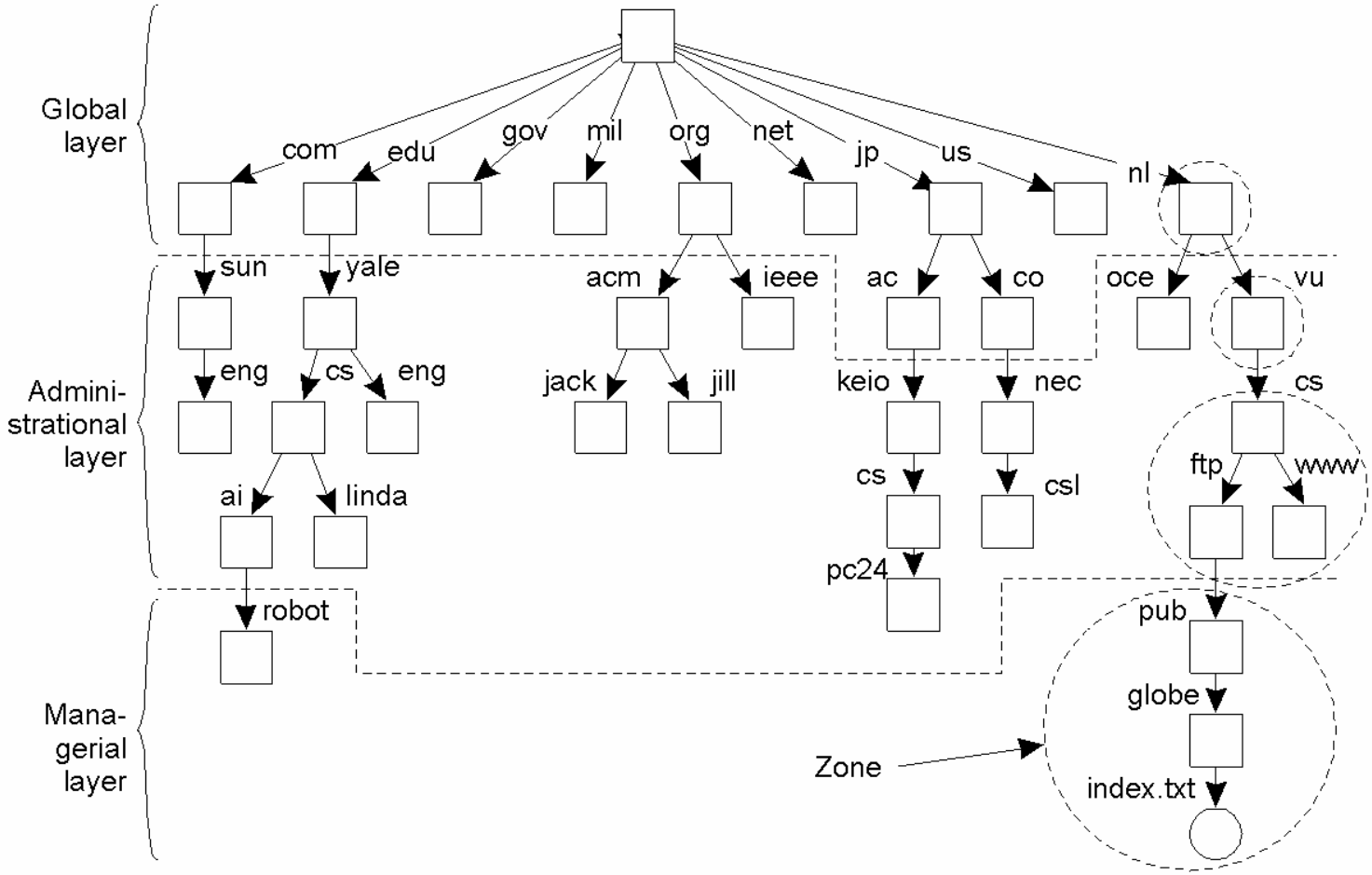
- X.500/Lightweight Directory Access Protocol (LDAP)

  C=US, OU=Virginia Tech, CN=Dennis Kafura ⟶ email address

- Corba Naming Service

  name1.kind1/name2.kind2/name3.kind3 ⟶ object

DNS Organization



- An example partitioning of the DNS name space, including Internet-accessible files, into three layers.

DNS Organization

Item	Global	Administrational	Managerial
Geographical scale of network	Worldwide	Organization	Department
Total number of nodes	Few	Many	Vast numbers
Responsiveness to lookups	Seconds	Milliseconds	Immediate
Update propagation	Lazy	Immediate	Immediate
Number of replicas	Many	None or few	None
Is client-side caching applied?	Yes	Yes	Sometimes

- A comparison between name servers for implementing nodes from a large-scale name space partitioned into a global layer, as an administrational layer, and a managerial layer.

The DNS Name Space - Records

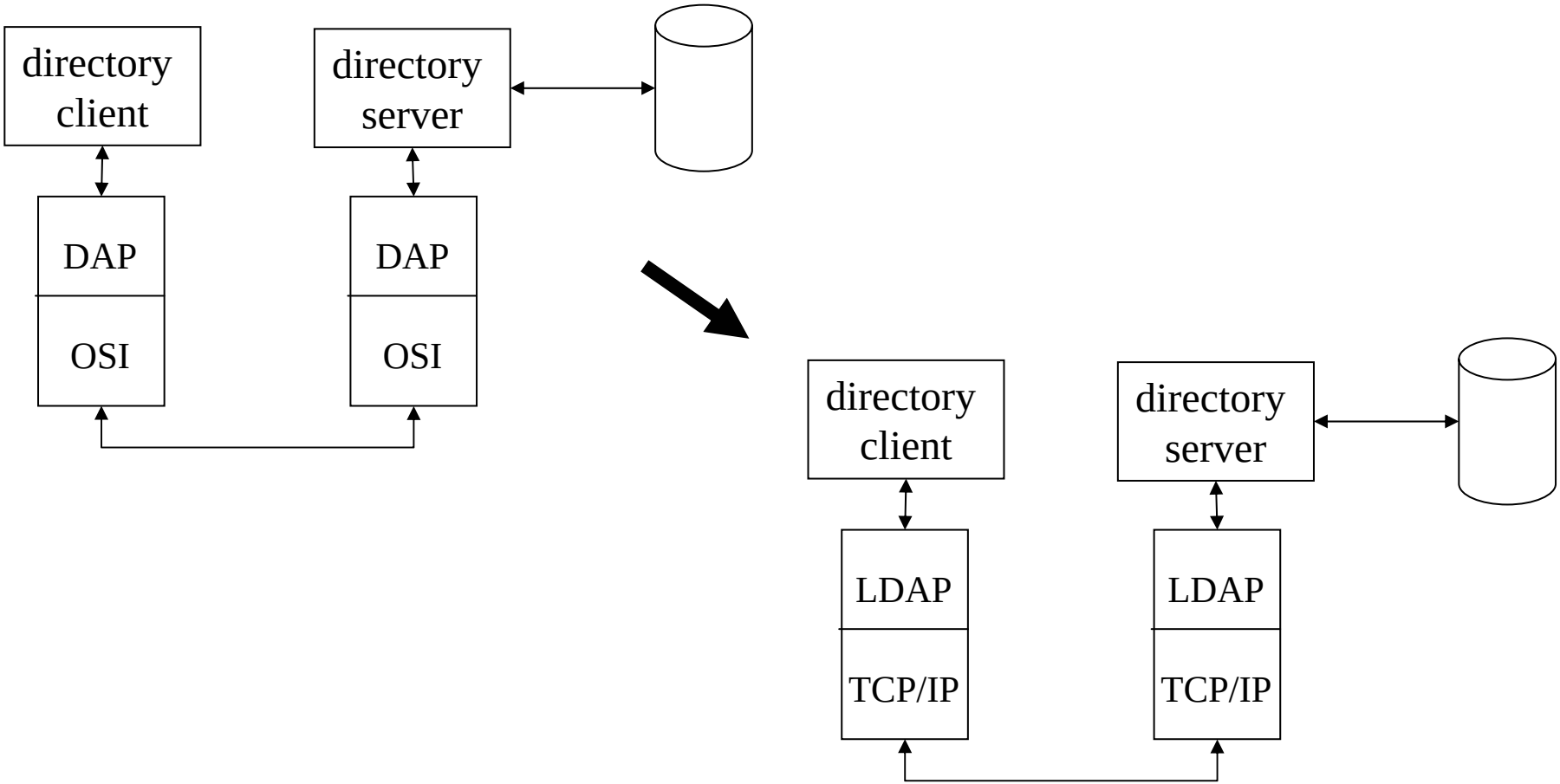
Type of record	Associated entity	Description
SOA	Zone	Holds information on the represented zone
A	Host	Contains an IP address of the host this node represents
MX	Domain	Refers to a mail server to handle mail addressed to this node
SRV	Domain	Refers to a server handling a specific service
NS	Zone	Refers to a name server that implements the represented zone
CNAME	Node	Symbolic link with the primary name of the represented node
PTR	Host	Contains the canonical name of a host
HINFO	Host	Holds information on the host this node represents
TXT	Any kind	Contains any entity-specific information considered useful

DNS Example

- An excerpt from the DNS database for the zone *cs.vu.nl*.

Name	Record type	Record value
cs.vu.nl	SOA	star (1999121502,7200,3600,2419200,86400)
cs.vu.nl	NS	star.cs.vu.nl
cs.vu.nl	NS	top.cs.vu.nl
cs.vu.nl	NS	solo.cs.vu.nl
cs.vu.nl	TXT	"Vrije Universiteit - Math. & Comp. Sc."
cs.vu.nl	MX	1 zephyr.cs.vu.nl
cs.vu.nl	MX	2 tornado.cs.vu.nl
cs.vu.nl	MX	3 star.cs.vu.nl
star.cs.vu.nl	HINFO	Sun Unix
star.cs.vu.nl	MX	1 star.cs.vu.nl
star.cs.vu.nl	MX	10 zephyr.cs.vu.nl
star.cs.vu.nl	A	130.37.24.6
star.cs.vu.nl	A	192.31.231.42
zephyr.cs.vu.nl	HINFO	Sun Unix
zephyr.cs.vu.nl	MX	1 zephyr.cs.vu.nl
zephyr.cs.vu.nl	MX	2 tornado.cs.vu.nl
zephyr.cs.vu.nl	A	192.31.231.66
www.cs.vu.nl	CNAME	soling.cs.vu.nl
ftp.cs.vu.nl	CNAME	soling.cs.vu.nl
soling.cs.vu.nl	HINFO	Sun Unix
soling.cs.vu.nl	MX	1 soling.cs.vu.nl
soling.cs.vu.nl	MX	10 zephyr.cs.vu.nl
soling.cs.vu.nl	A	130.37.24.11
laser.cs.vu.nl	HINFO	PC MS-DOS
laser.cs.vu.nl	A	130.37.30.32
vucs-das.cs.vu.nl	PTR	0.26.37.130.in-addr.arpa
vucs-das.cs.vu.nl	A	130.37.26.0

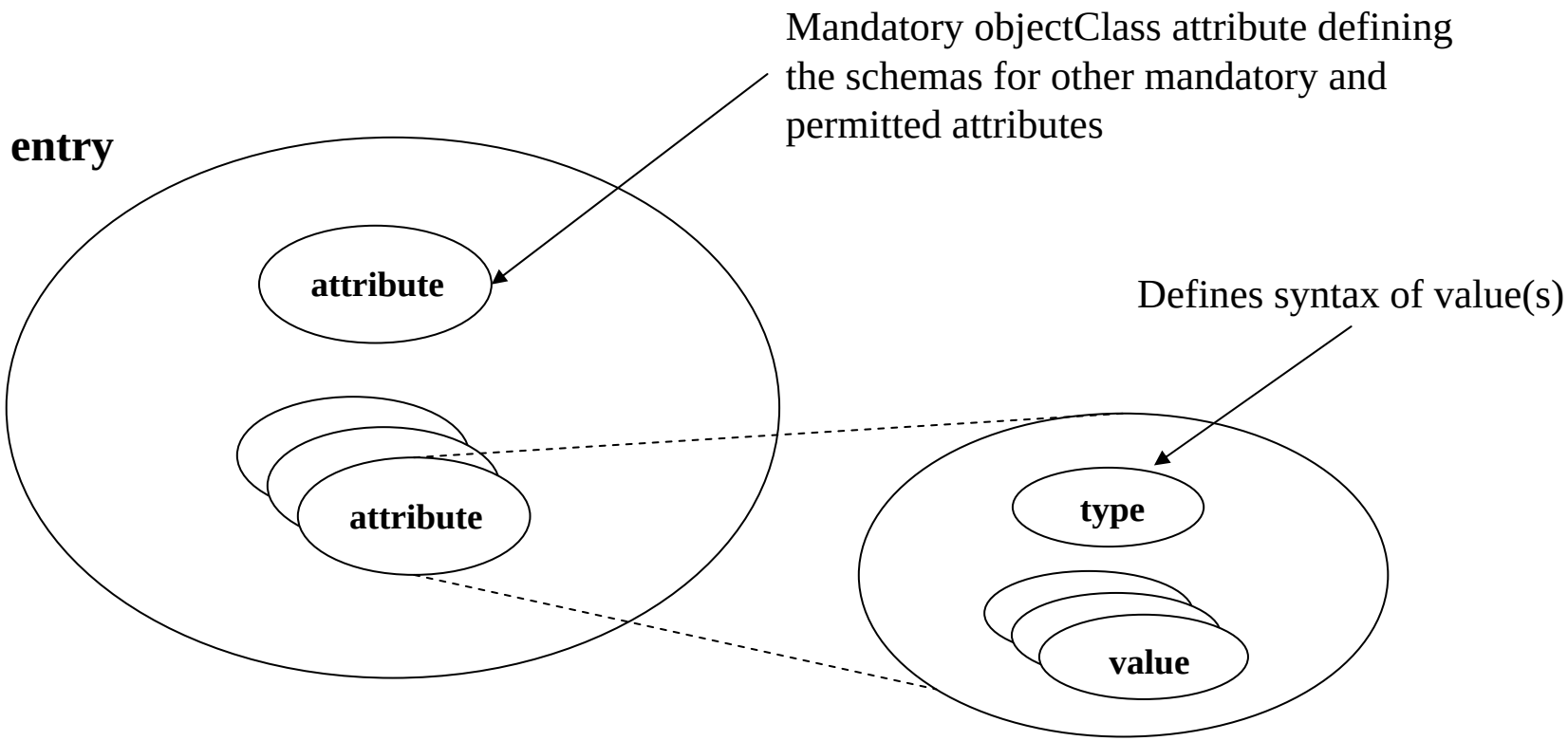
LDAP: Evolution



LDAP/X.500 - Models

- Information (what is stored in an entry)
- Naming (how are entries identified/organized)
- Functional (what operations are provided)
- Security (how is authentication and authorization provided)

LDAP Information Model

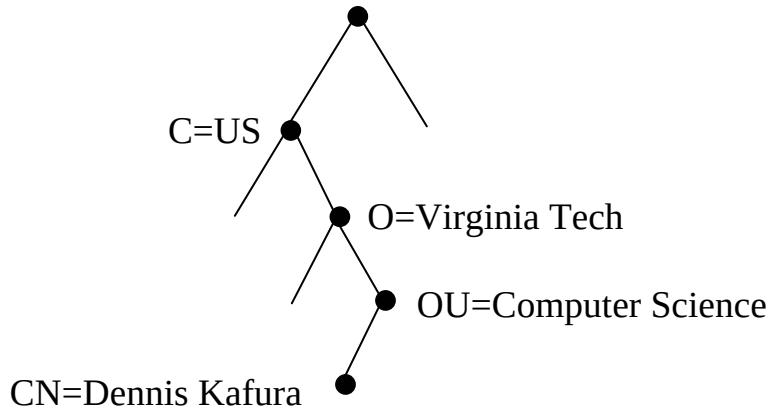


Examples:

attribute	syntax
commonName (CN)	cis: case ignore string
telephoneNumber	tel: text, ignoring blanks and dashes

LDAP Naming Model

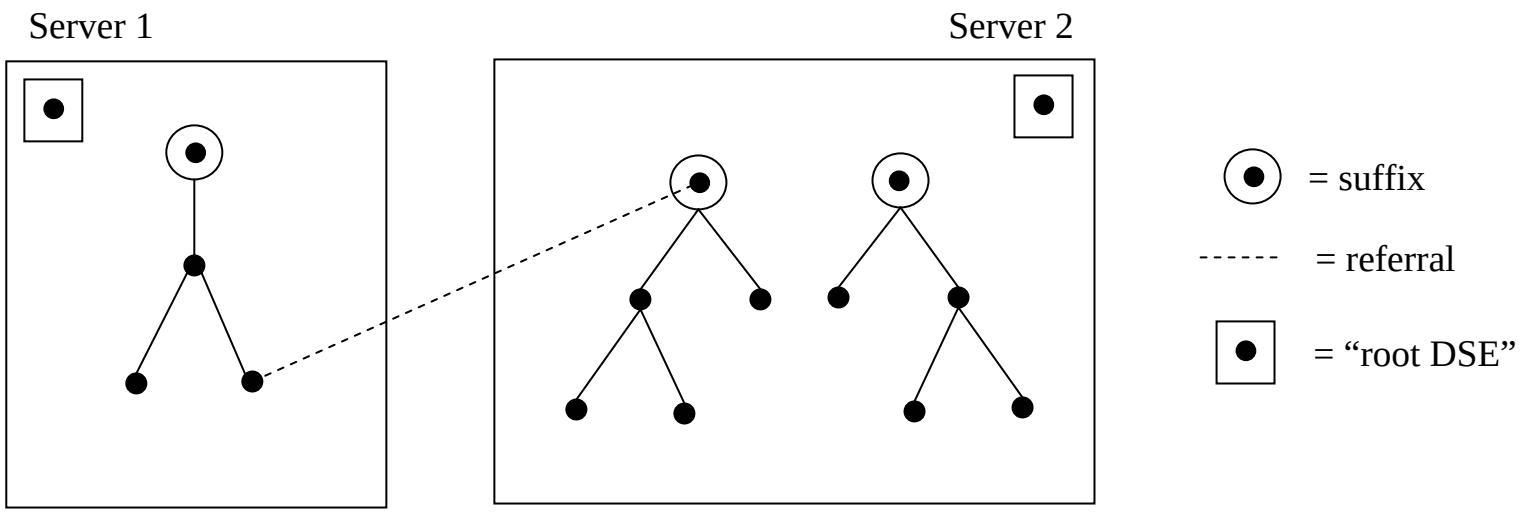
Directory Information Tree (DIT)



- relative distinguished name (RDN) represented as a *name = value* pair (e.g. OU = Computer Science)

- distinguished name (DN) represented as a comma-separated sequence of RDNs (e.g., CN=Dennis Kafura, OU=Computer Science, O=Virginia Tech, C= US)

LDAP Naming Model - Distribution



The “root DSE” stored at a server is an empty (zero-length) distinguished name that is used to store as its attributes:

- suffixes provided by this server
- object classes and attribute schema

LDAP Functional Model

- Operation Categories
  - Query (search, compare)
  - Update (add, delete, modify)
  - Authentication (bind, unbind, abandon)
- Search parameters
  - base (DN of where to begin search
  - scope (extent of subtree examined)
  - filter (criteria for an entry to be matched)
  - attributes (list of attributes returned from each matched entry)
  - alias (whether aliases are followed)
  - limits (in size/time)
- Filters
  - boolean combination of attribute-value comparisons
  - example: (| (SN = Smith) (& (OU=Autstin) (SN=Miller))

LDAP Security Model

- Authentication
  - None (anonymous session)
  - Basic (DN (i.e., username) and password)
  - Simple Authentication and Security Layer (SASL)
    - GSSAPI
    - Kerberos
    - External (e.g., SSL/TLS)

- Access control
  - vendor specific
  - access control list is typical

Corba Naming Service - Names

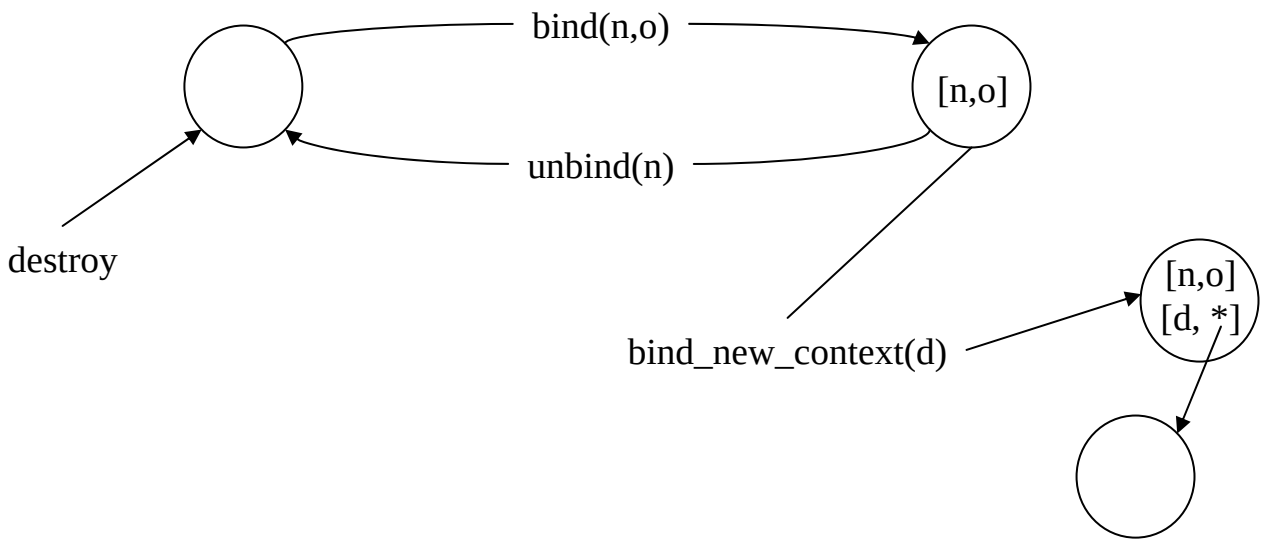
- Implemented as a CORBA service; operations described as an IDL interface.
- Names: each name component can refer to
  - naming context (an object for resolving names)
  - application object
- Naming contexts can be structured to create naming graphs
- Compound names locate an object in a naming graph (a sequence of names defining a path in the naming graph)

```
struct NameComponent {
  String id;
  String kind;
};
typedef sequence<NameComponent> Name;
```

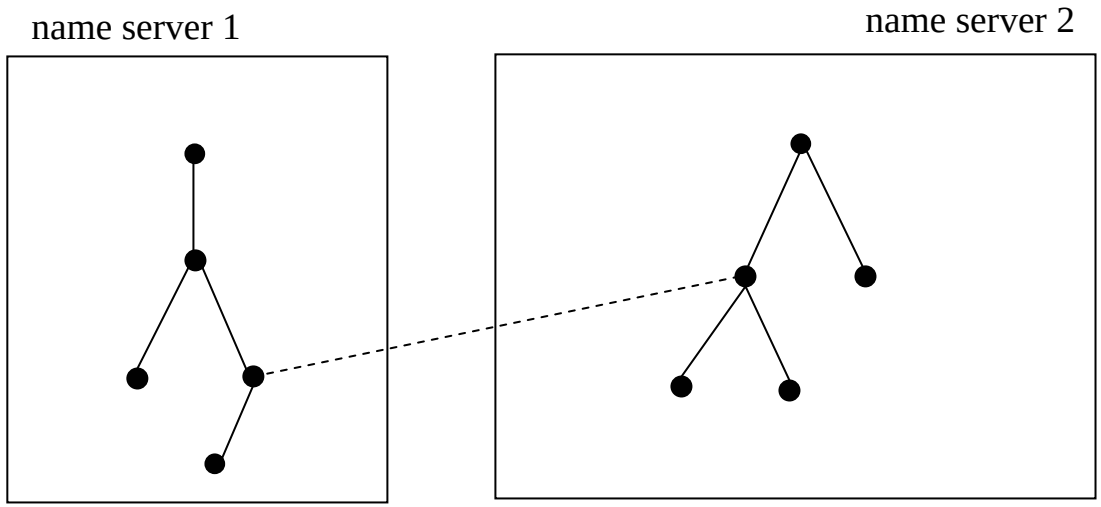
Corba Naming Service - Operations

Operations:

```
void bind (in Name n,m in Object o);
Object resolve (in Name n);
void undind (Name n);
NamingContext bind_new_context (in Name n);
void destroy ();
```



Corba Naming Service - Federation



A federated naming graph can span multiple naming services.