Distributed Systems

a.y. 2023/2024

Distributed Systems:

Naming

- Names:
  - Resource sharing
  - Unique resource identification
  - Reference of locations
- Name resolution
- Naming systems (distributed across…)
- Efficiency and scalability issues

A name is …

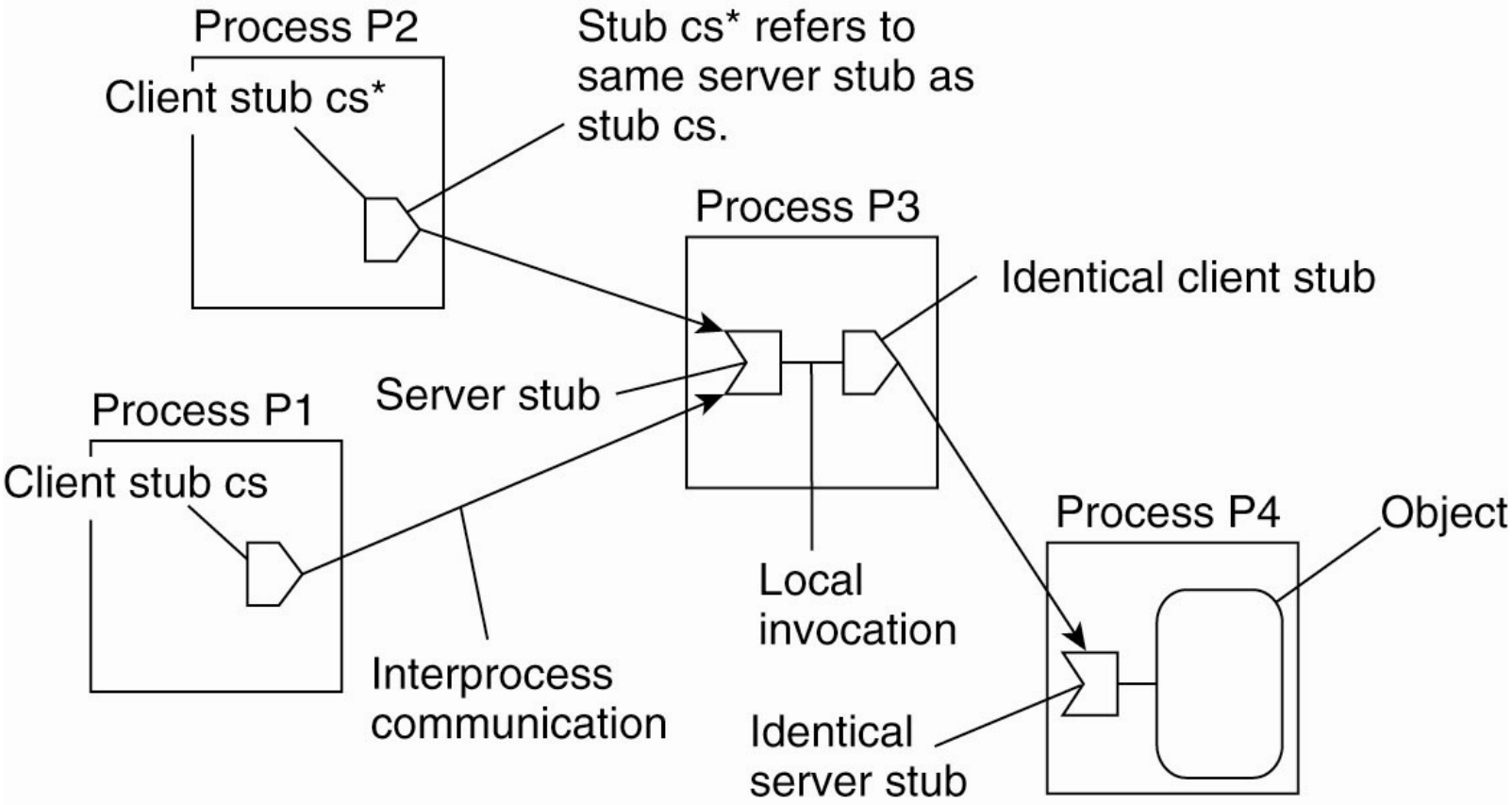
- …string of bit (characters) to refer to an entity

- To operate on an entity…access point
- The name of an access point is called address
- Can the name of the access point of an entity be the name of the entity ?
- Location independent names

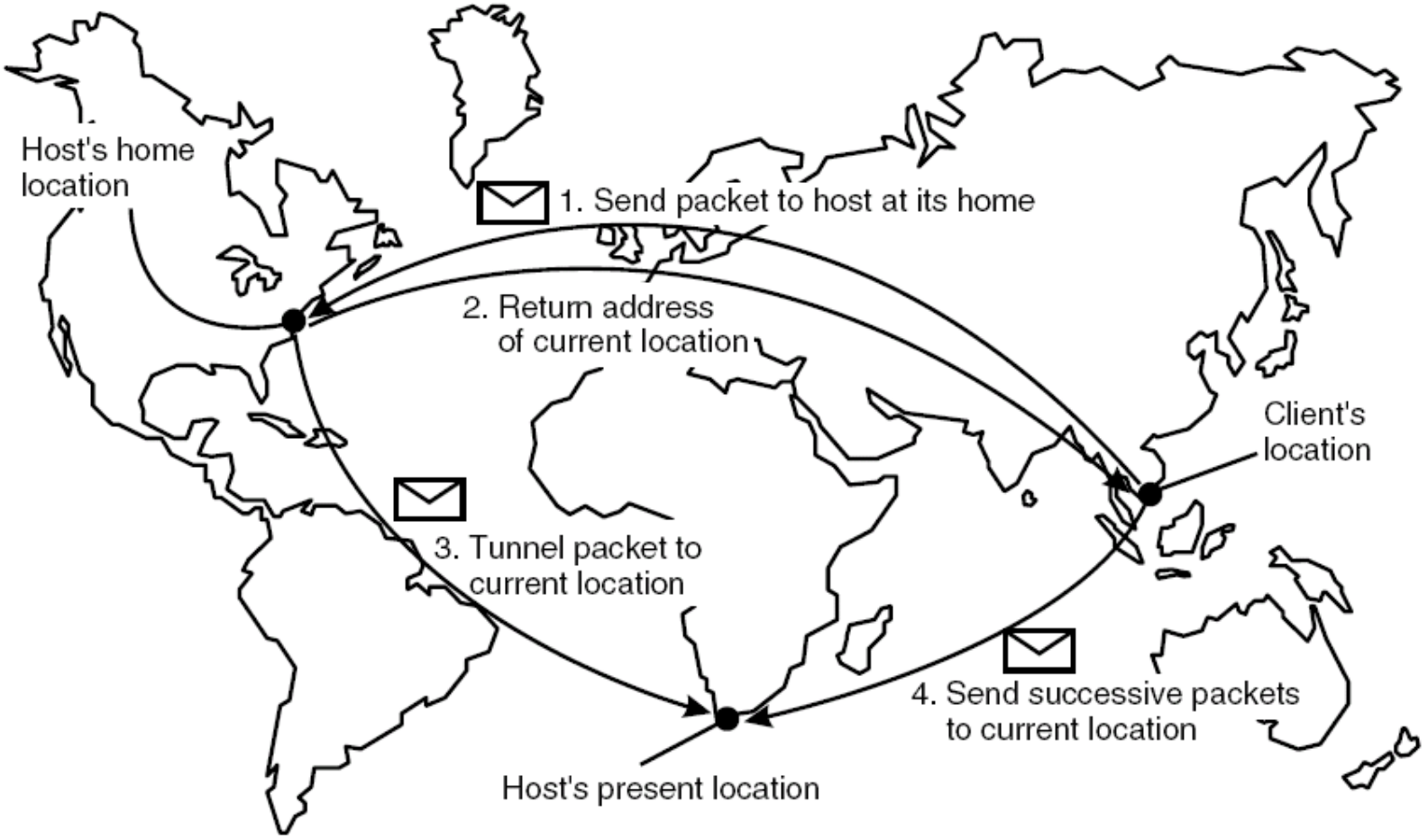
A true identifiers

- 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

Forwarding Pointers using (client stub, server stub) pairs



Home-Based Approaches

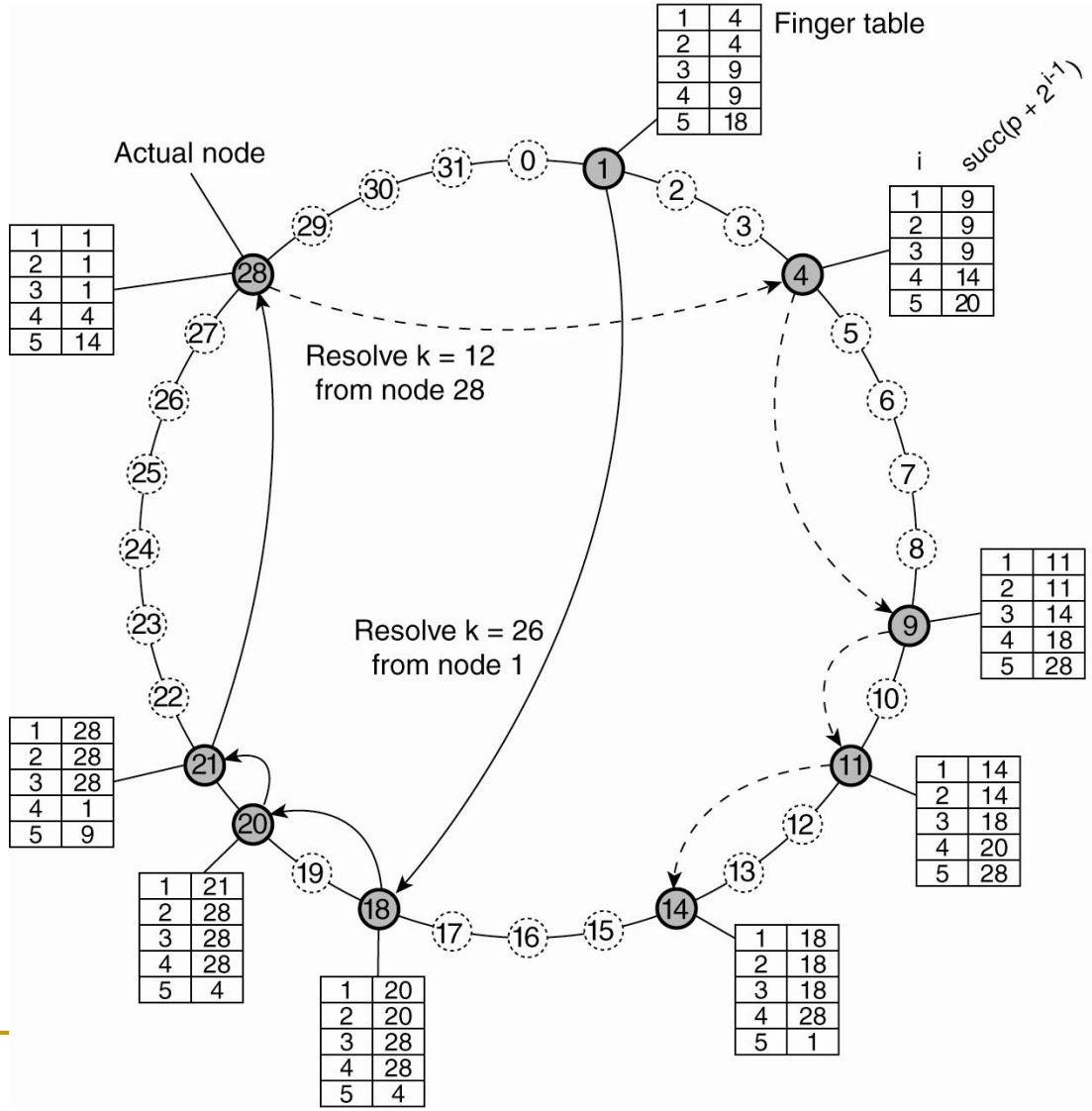


Names to address binding

- Human-friendly names
- A (name,address) table over the network…
- Solving names is related to message routing

Distributed Hash Tables

General Mechanism



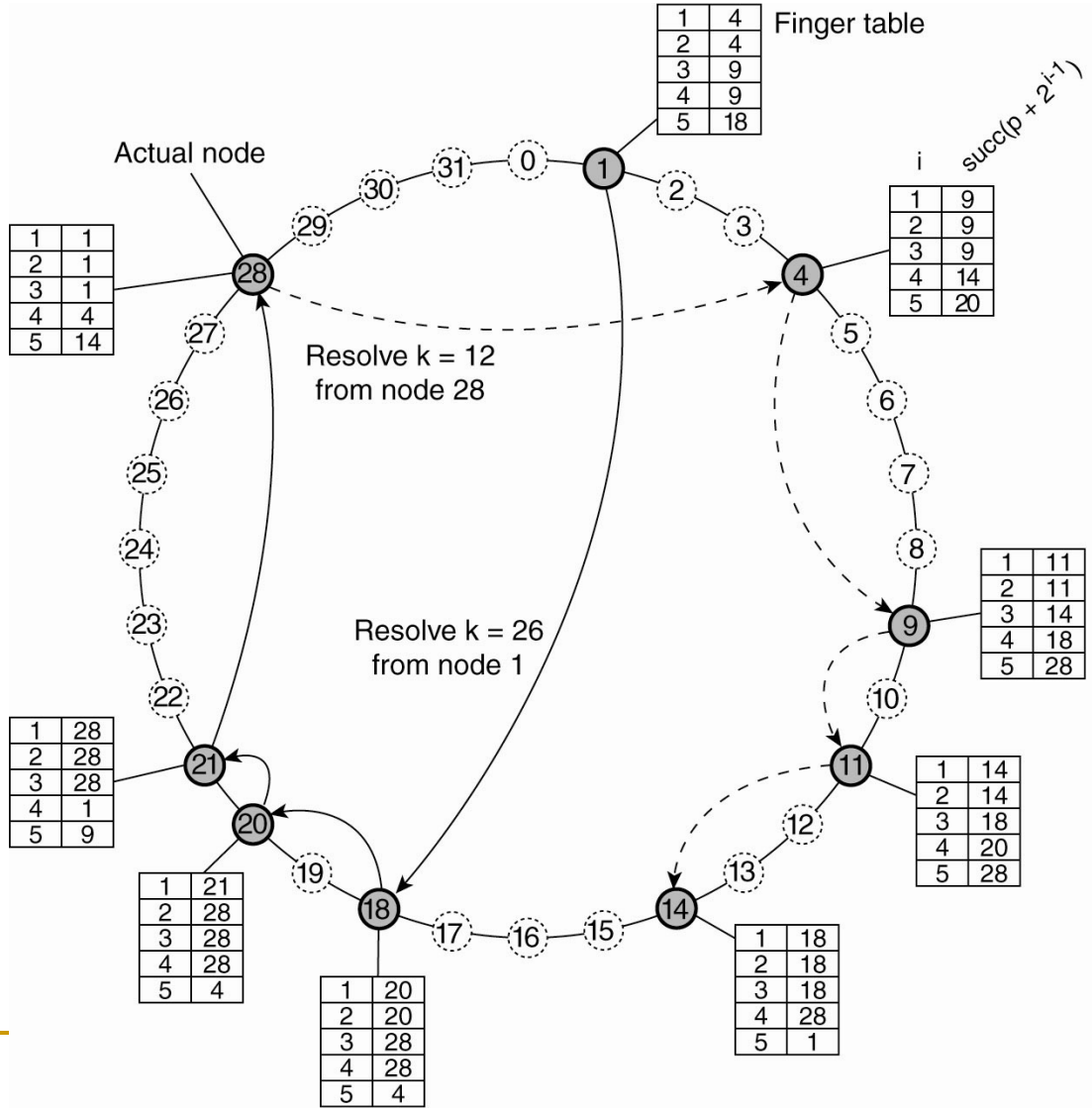
- Resolve a key k to the address of succ(k)

- A linear approach:
  - p
  - succ(p+1)
  - prec(p)

- Each node maintains a table with m entries
- $FT_p[i] = succ(p+2^{i-1})$
- $q = FT_p[i] < k < FT_p[i+1]$

Distributed Hash Tables

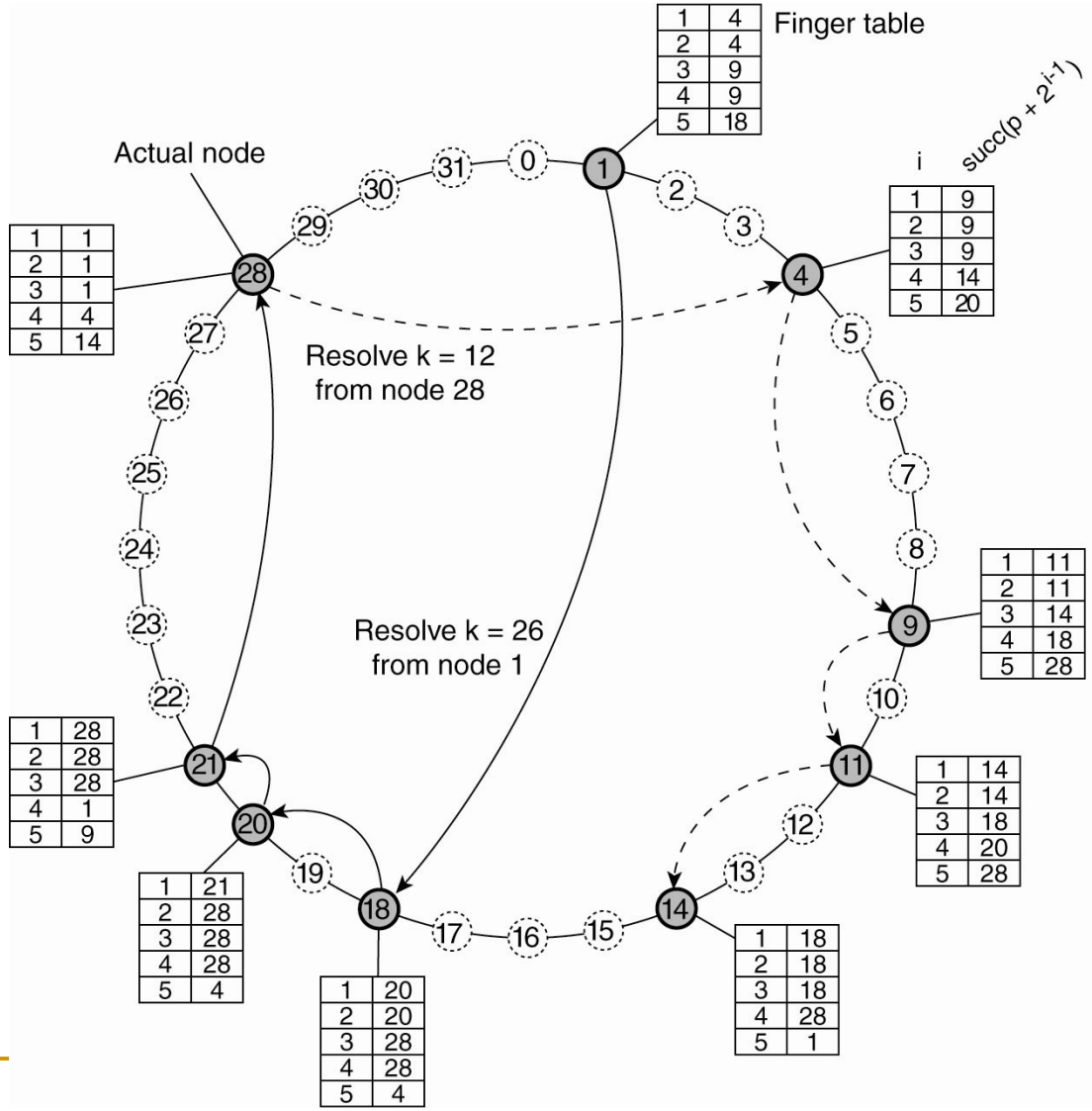
General Mechanism



Distributed Hash Tables

General Mechanism

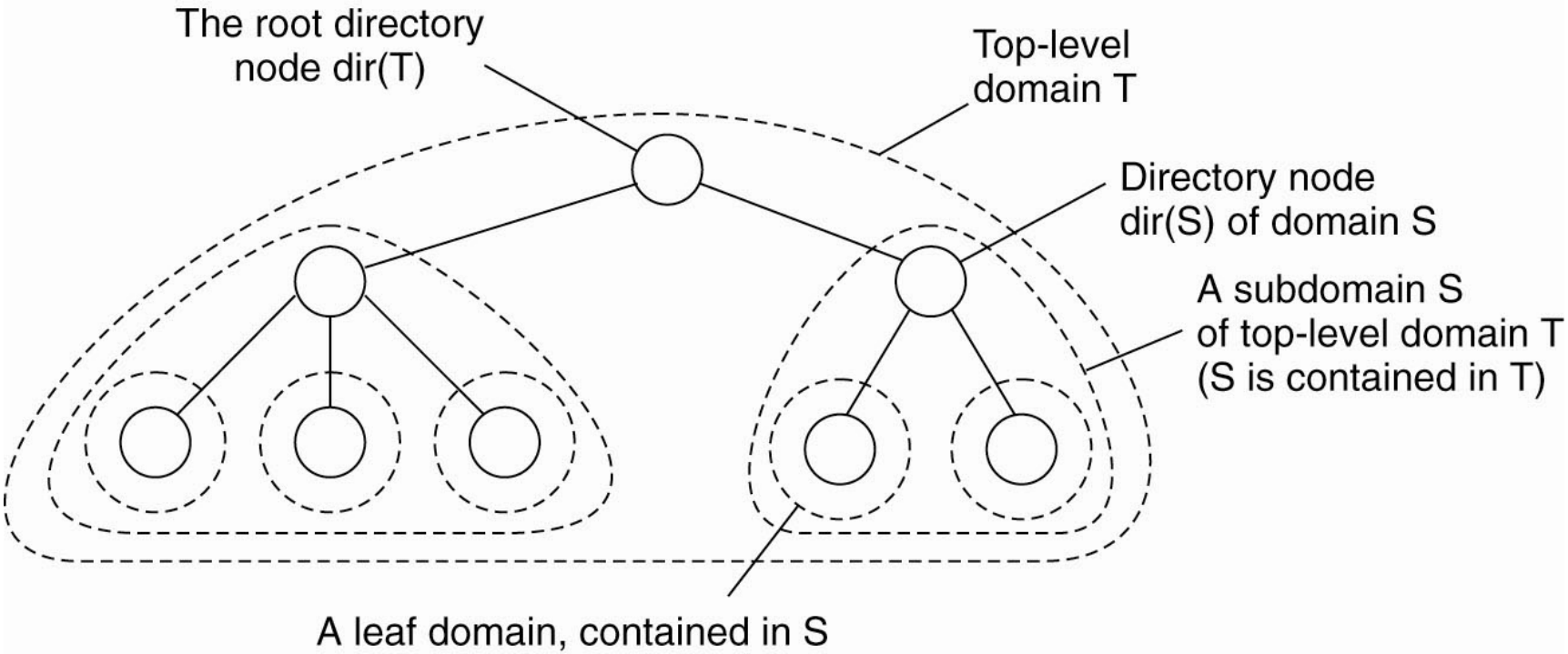
- Resolving key 26 from node 1 and key 12 from node 28



- Entering the ring....
  - Lookup for succ(p+1)
- Leaving the ring...

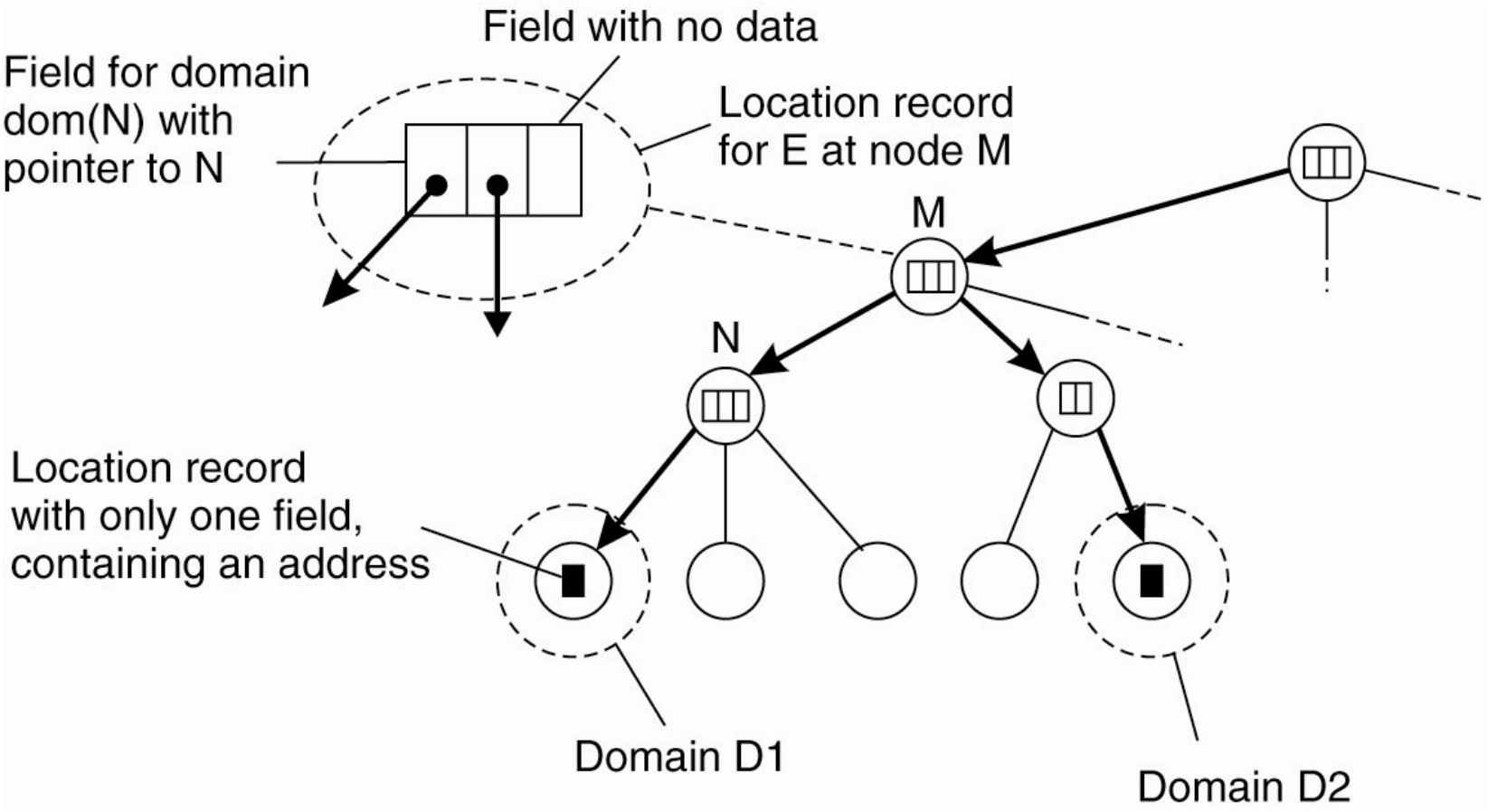
- Keeping the table up-to-date
  - q = pred(succ(q+1)) ... For $FT_q[1]$
  - k = q + 2^{i+1} for each entry …

Hierarchical Approaches



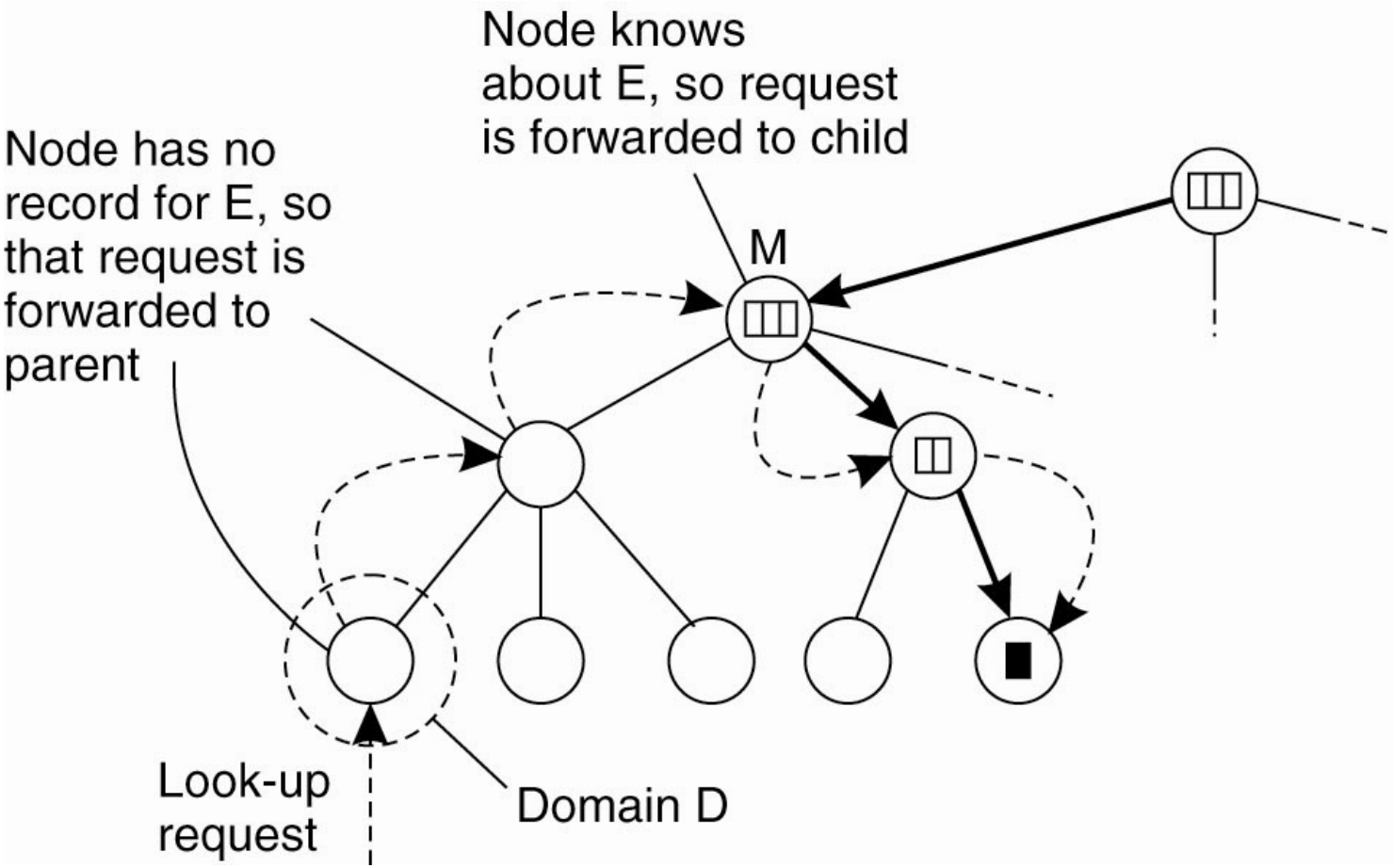
- Hierarchical organization of a location service into domains, each having an associated directory node.

Hierarchical Approaches



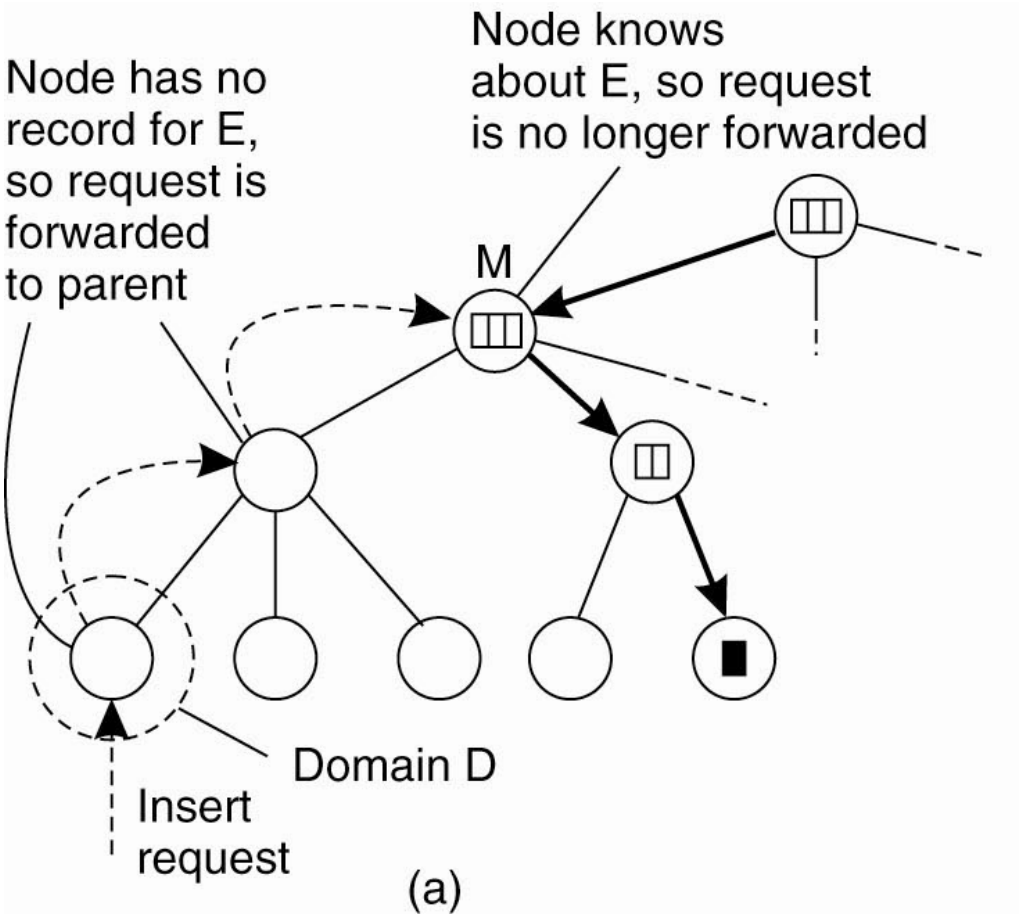
- Storing information of an entity having two addresses in different leaf domains.

Hierarchical Approaches



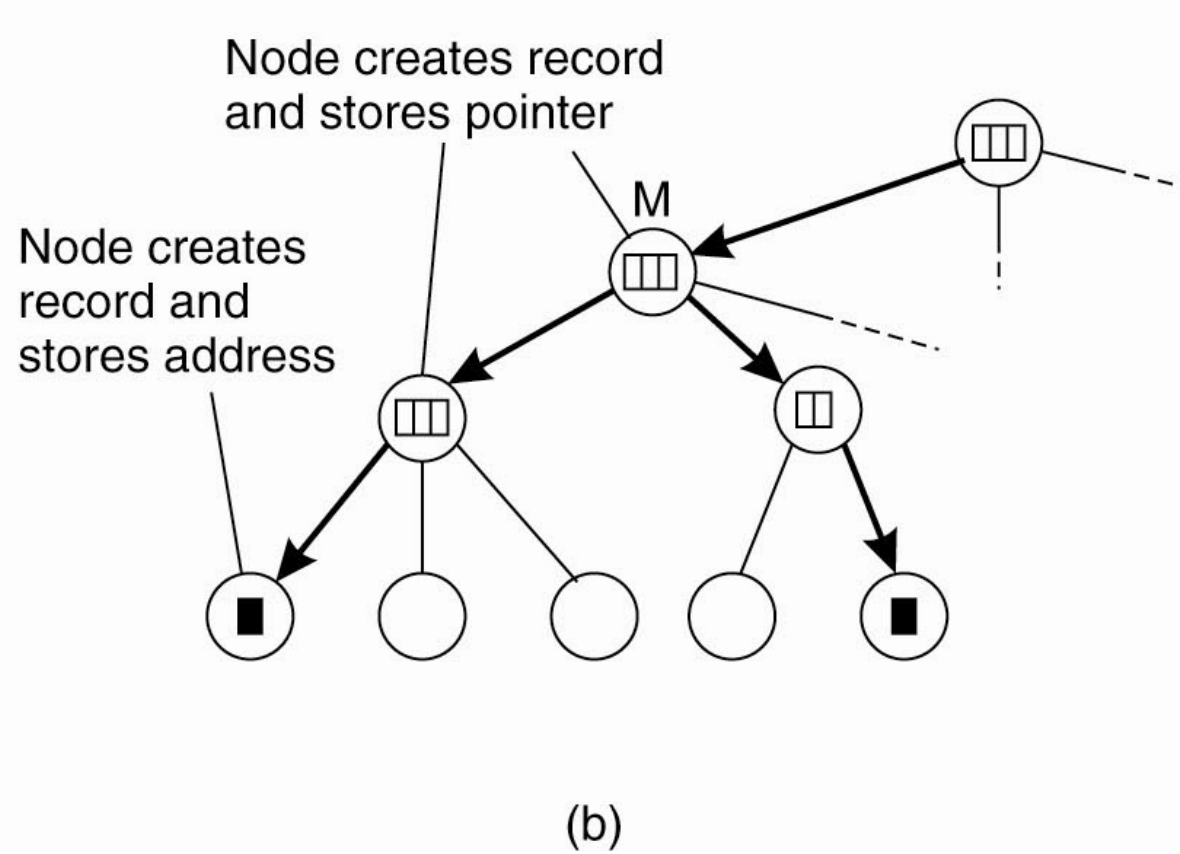
- Looking up a location in a hierarchically organized location service.

Hierarchical Approaches



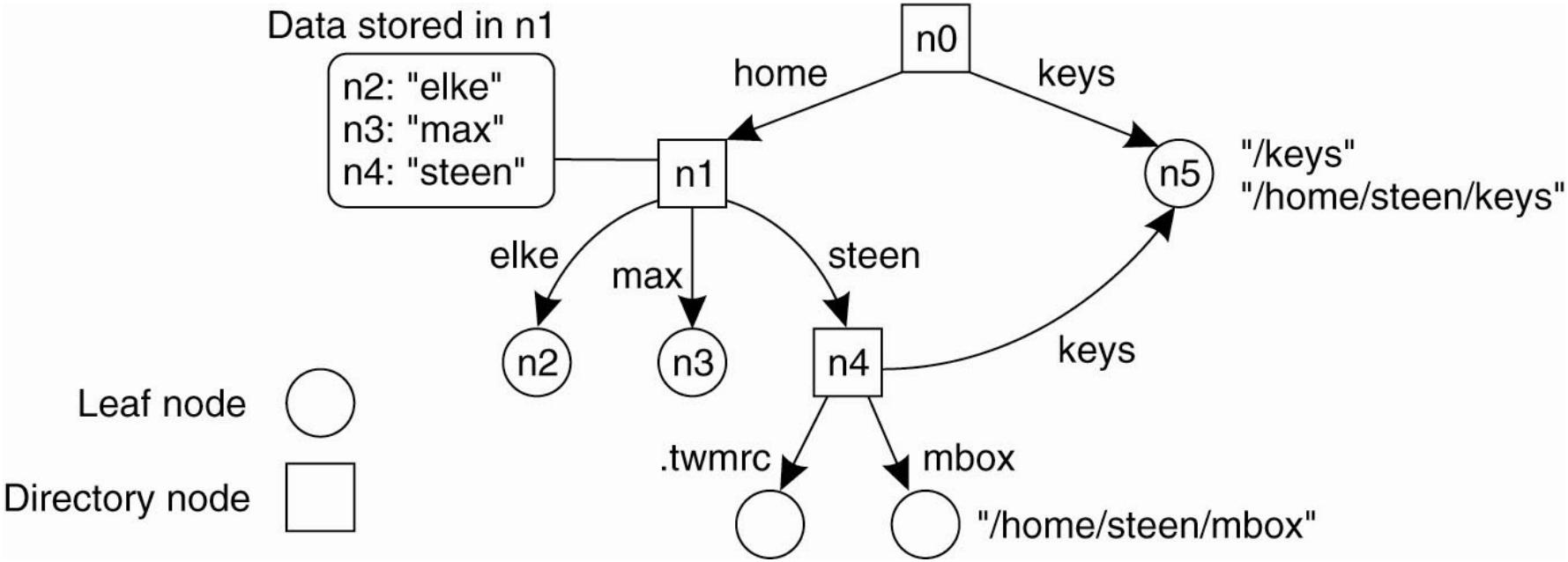
- (a) An insert request is forwarded to the first node that knows about entity E.

Hierarchical Approaches



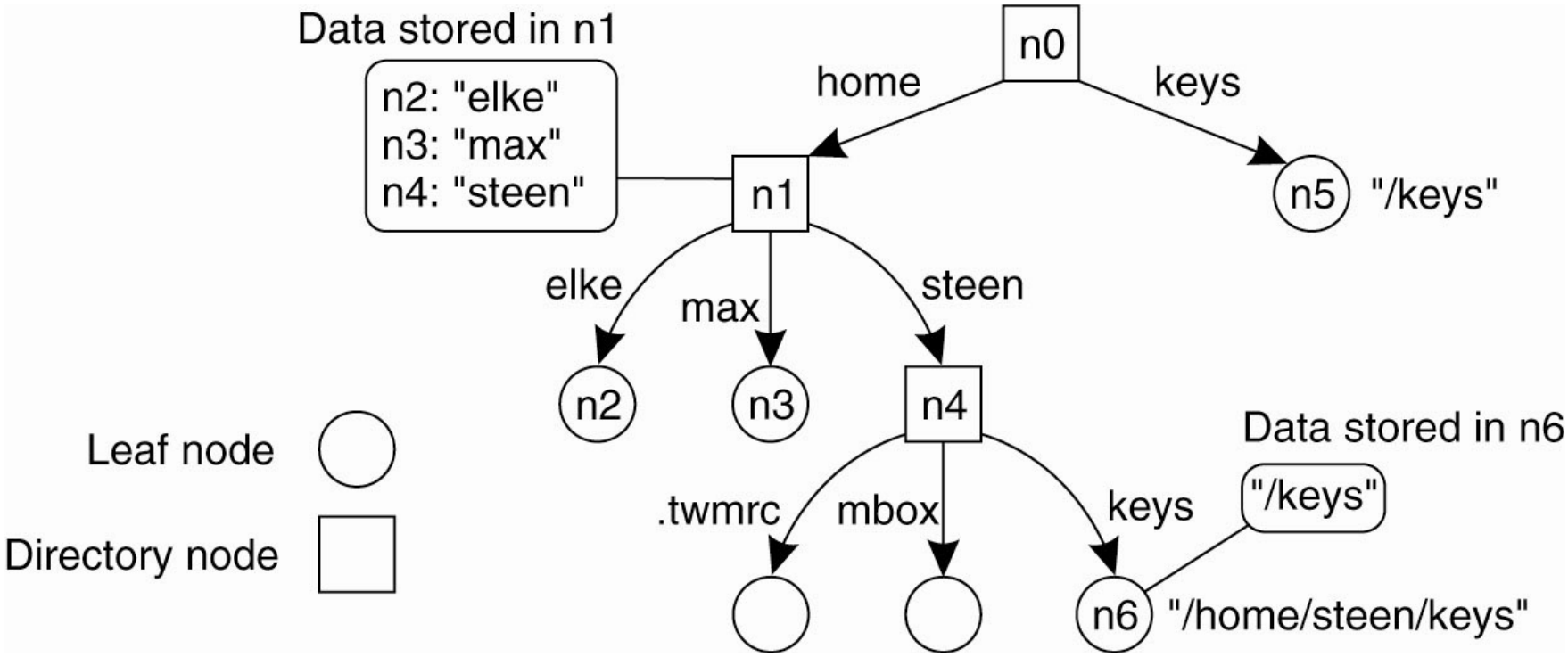
- (b) A chain of forwarding pointers to the leaf node is created.

Name Spaces



- A general naming graph with a single root node.

Linking and Mounting (1)

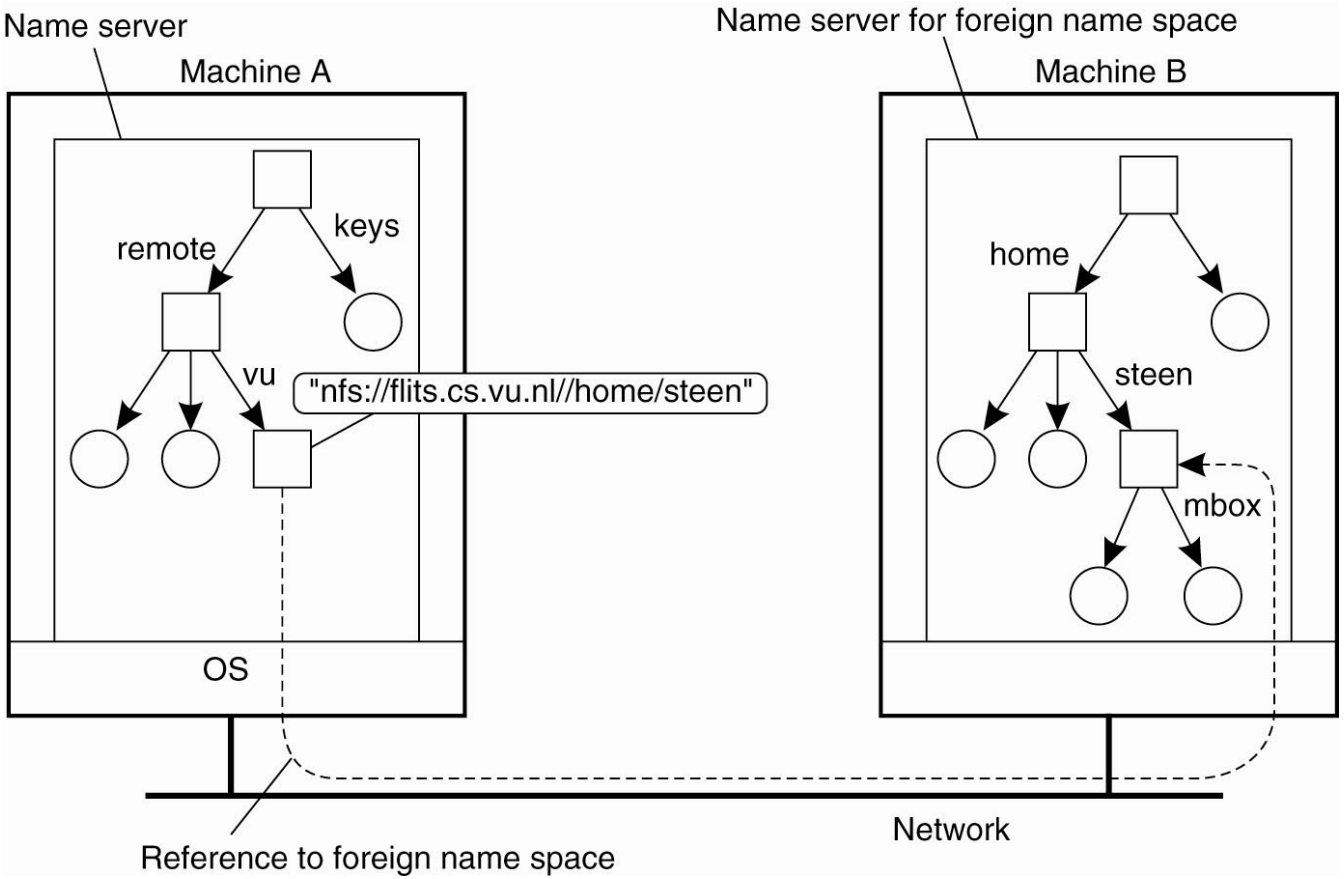


- Figure 5-11. The concept of a symbolic link explained in a naming graph.

Linking and Mounting (2)

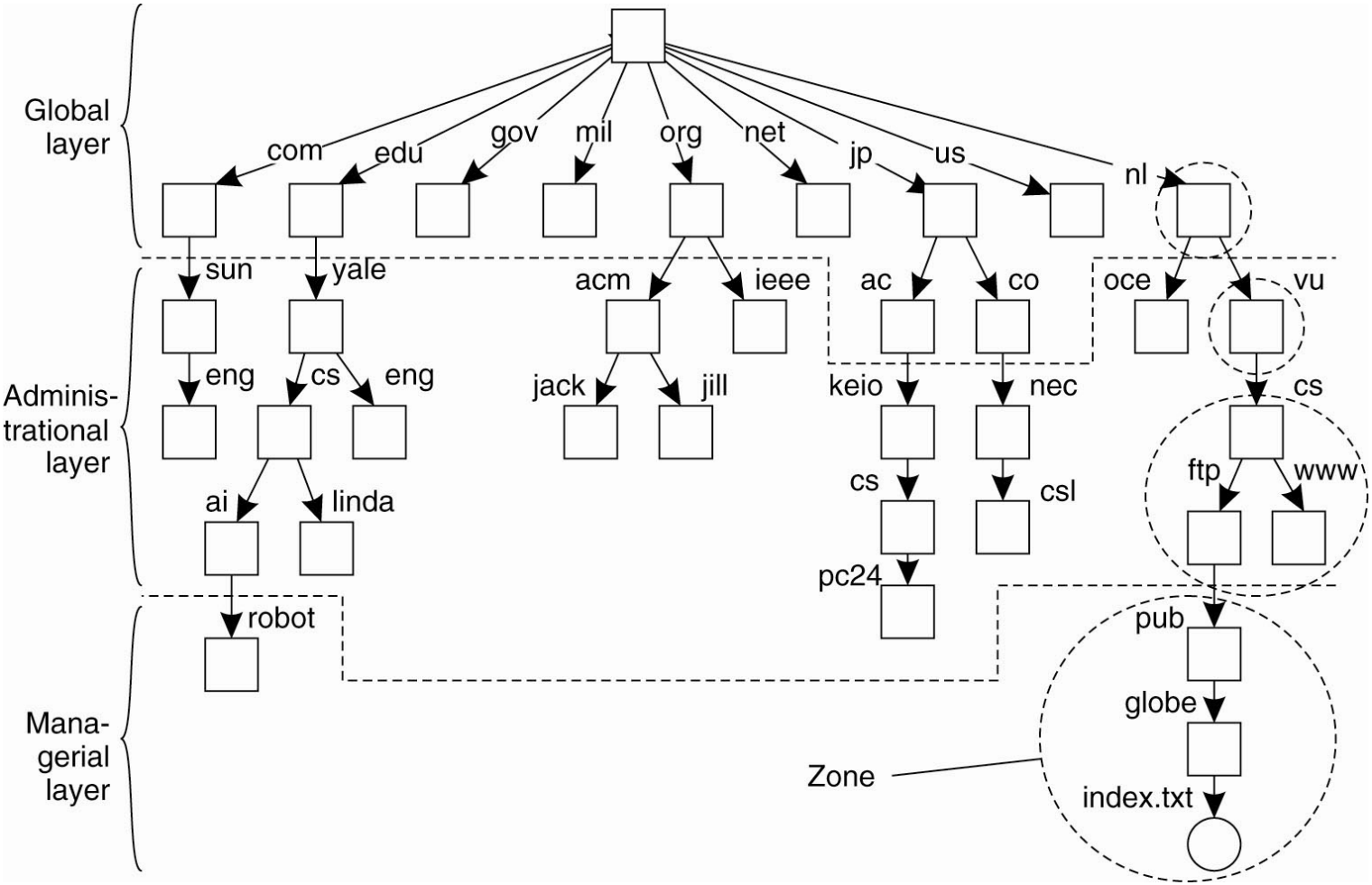
- Information required to mount a foreign name space in a distributed system
- The name of an access protocol.
- The name of the server.
- The name of the mounting point in the foreign name space.

Linking and Mounting (3)



- Mounting remote name spaces through a specific access protocol.

Name Space Distribution

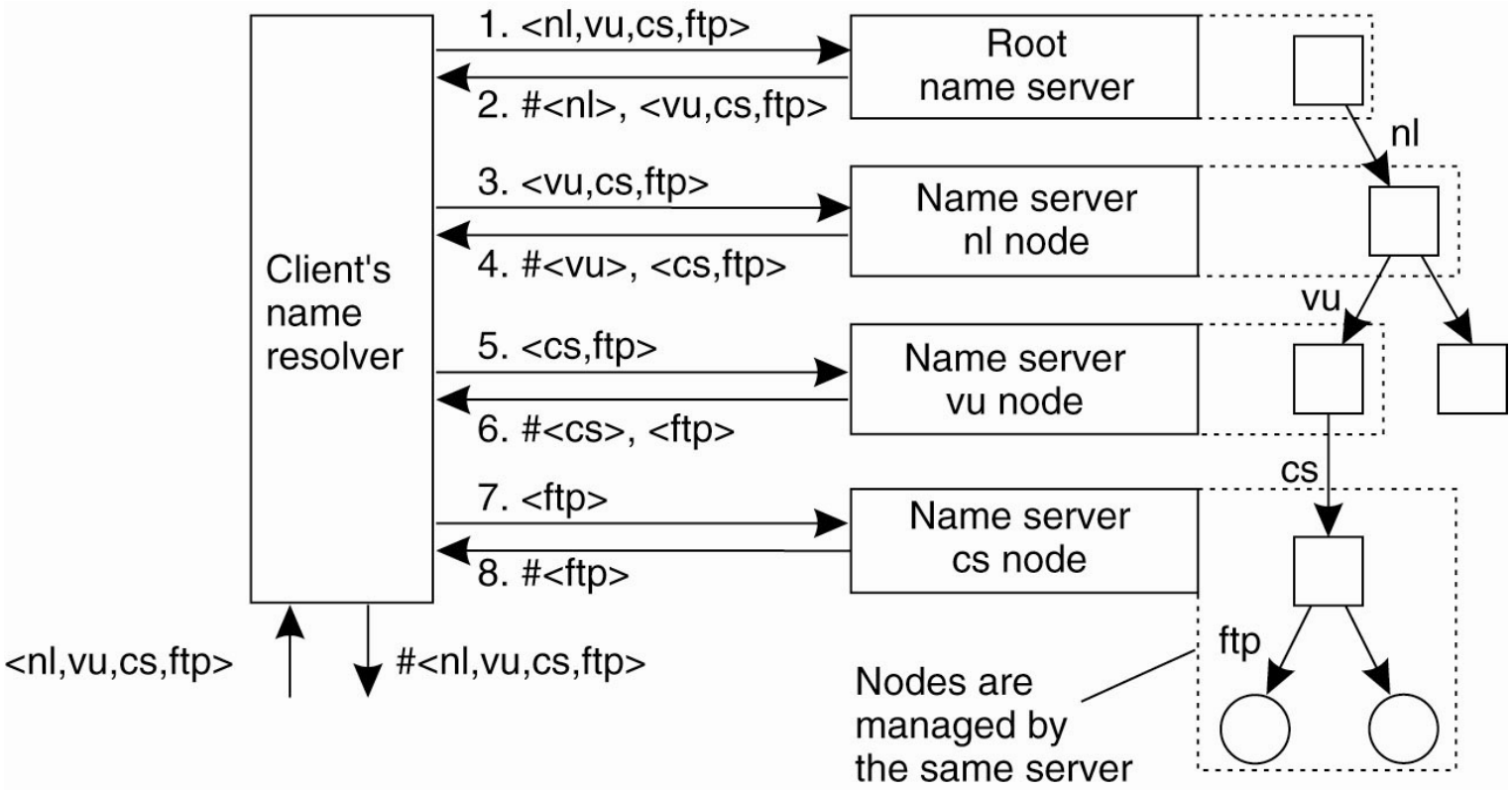


Name Space Distribution (2)

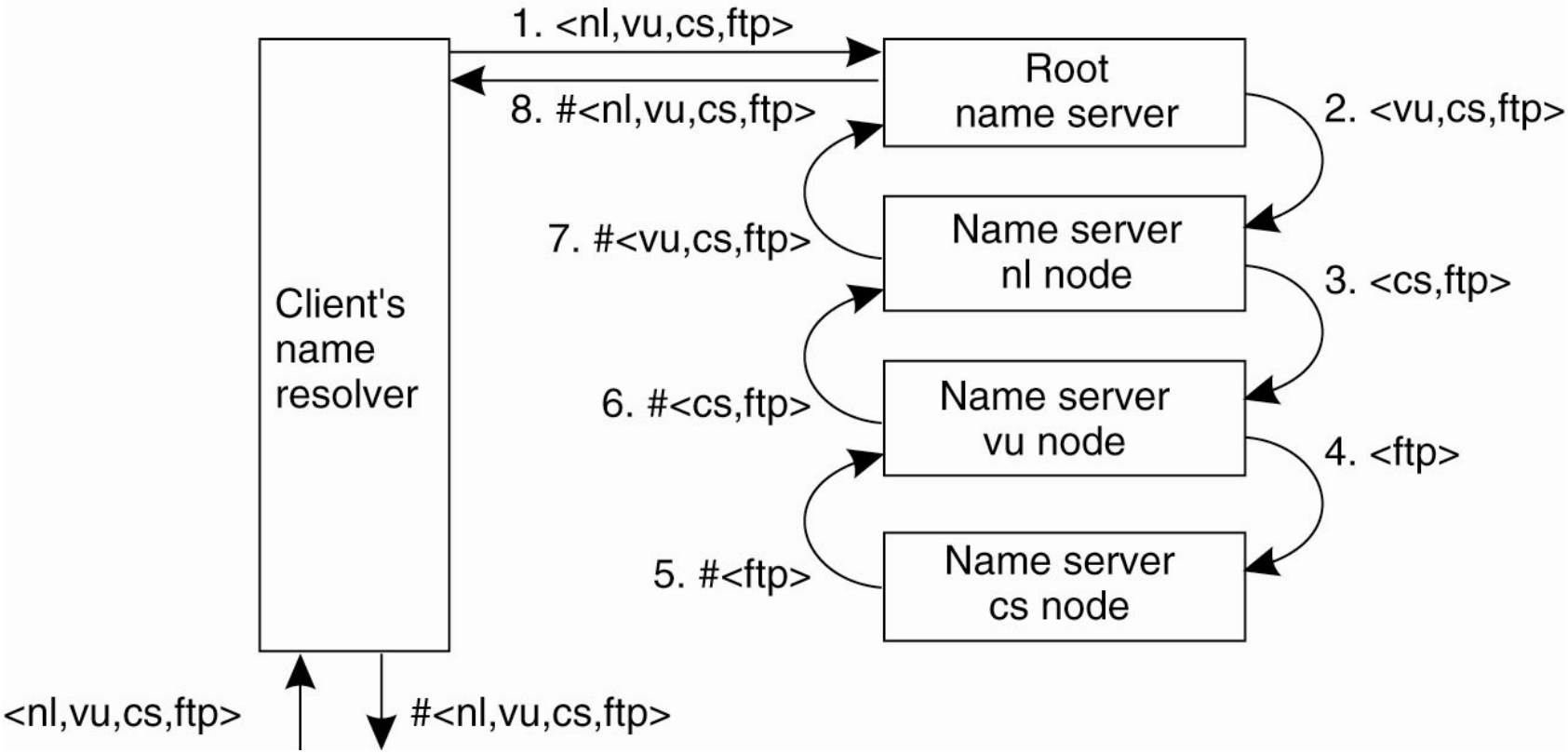
Item	Global	Administrational	Managerial
Geographical scale of network	Worldwide	Organization	Department
Total number of nodes	Few	Many	Vast numbers
Responsiveness to lookups	Seconds	Milliseconds	Immediate
Update propagation	Lazy	Immediate	Immediate
Number of replicas	Many	None or few	None
Is client-side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

- Figure 5-14. A comparison between name servers for implementing nodes from a large-scale name space partitioned into a global layer, an administrational layer, and a managerial layer.

Iterative name resolution



Recursive name resolution



Implementation of Name Resolution (3)

Server for node	Should resolve	Looks up	Passes to child	Receives and caches	Returns to requester
cs	<ftp>	#<ftp>	—	—	#<ftp>
vu	<cs,ftp>	#<cs>	<ftp>	#<ftp>	#<cs> #<cs, ftp>
nl	<vu,cs,ftp>	#<vu>	<cs,ftp>	#<cs> #<cs,ftp>	#<vu> #<vu,cs> #<vu,cs,ftp>
root	<nl,vu,cs,ftp>	#<nl>	<vu,cs,ftp>	#<vu> #<vu,cs> #<vu,cs,ftp>	#<nl> #<nl,vu> #<nl,vu,cs> #<nl,vu,cs,ftp>

- Recursive name resolution of <*nl, vu, cs, ftp*>. Name servers cache intermediate results for subsequent lookups.

Attribute-based naming

- *(attribute, value)* list
- Directory services
- RDF…resource descriptor framework

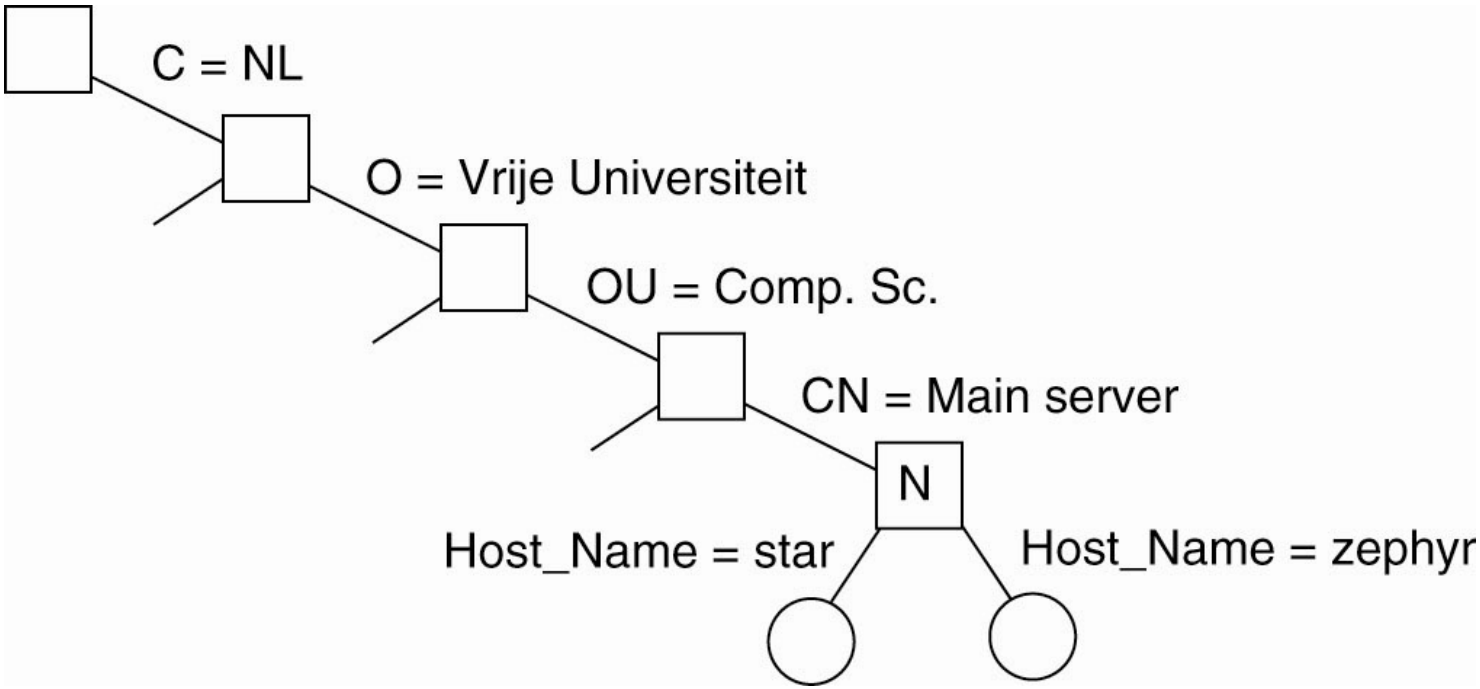
LDAP

- Lightweight directory access protocol

an LDAP directory entry…

Attribute	Abbr.	Value
Country	C	NL
Locality	L	Amsterdam
Organization	O	Vrije Universiteit
OrganizationalUnit	OU	Comp. Sc.
CommonName	CN	Main server
Mail_Servers	—	137.37.20.3, 130.37.24.6, 137.37.20.10
FTP_Server	—	130.37.20.20
WWW_Server	—	130.37.20.20

Hierarchical Implementations: LDAP



- Part of a directory information tree.

Hierarchical Implementations: LDAP (3)

Attribute	Value
Country	NL
Locality	Amsterdam
Organization	Vrije Universiteit
OrganizationalUnit	Comp. Sc.
CommonName	Main server
Host_Name	star
Host_Address	192.31.231.42

Attribute	Value
Country	NL
Locality	Amsterdam
Organization	Vrije Universiteit
OrganizationalUnit	Comp. Sc.
CommonName	Main server
Host_Name	zephyr
Host_Address	137.37.20.10

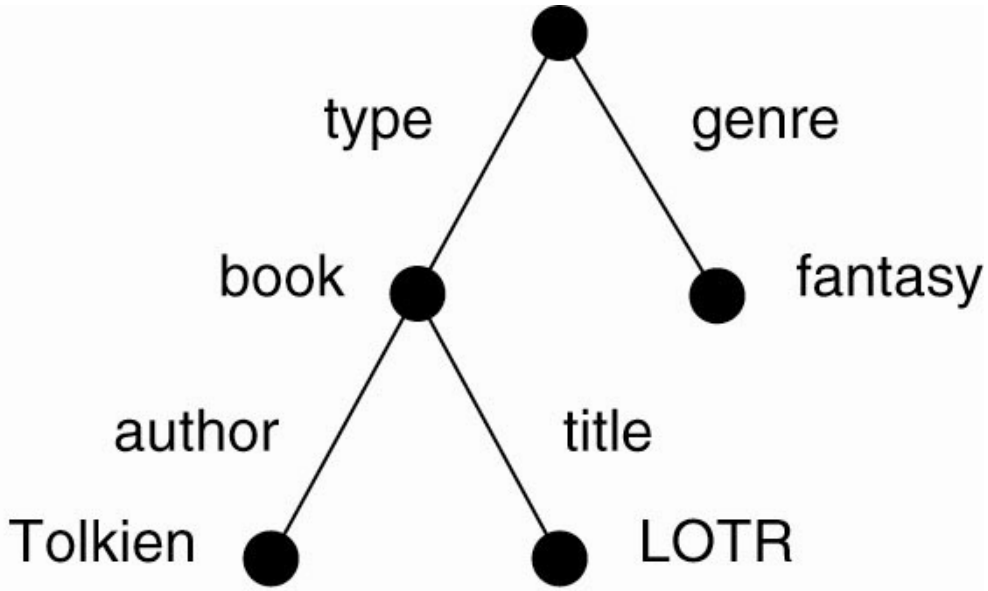
(b)

- Two directory entries having *Host_Name* as RDN.

Mapping to Distributed Hash Tables (1)

```
description {
    type = book
    description {
        author = Tolkien
        title = LOTR
    }
    genre = fantasy
}
```

(a)



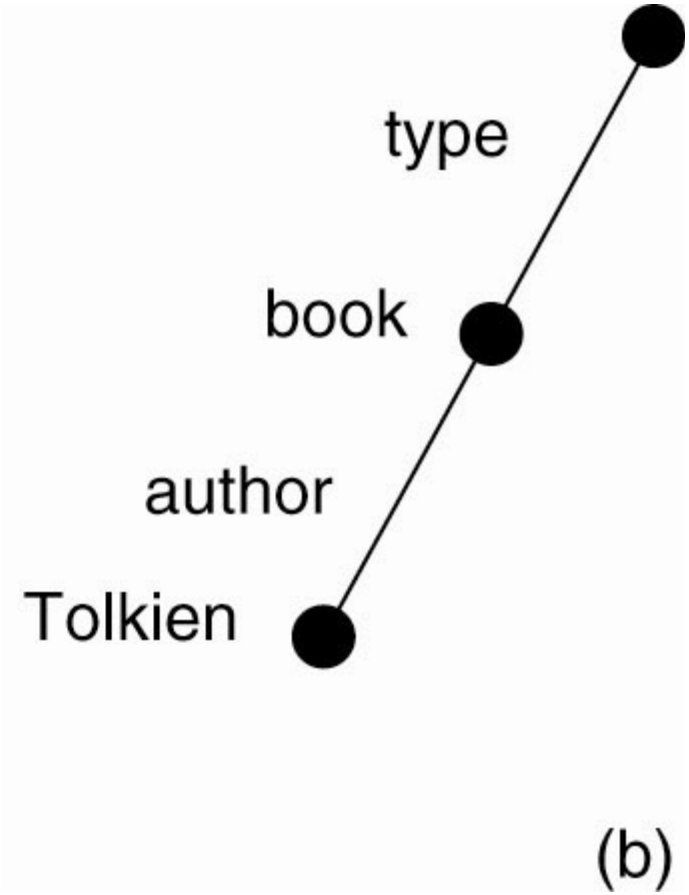
(b)

- (a) A general description of a resource.
  (b) Its representation as an AVTree.

Mapping to Distributed Hash Tables (2)

```
description {
    type = book
    description {
        author = Tolkien
        title = *
    }
    genre = *
}
```

(a)



(b)

- (a) The resource description of a query. (b) Its representation as an AVTree.

…Distributed Systems…

End of lecture