

---

## Naming and binding

---

- 
- My 17" PowerBook
  - The rightmost computer on my desk
  - Paul's big aluminum laptop
  - talbot
  - talbot.pk.org
  - 192.168.60.155
  - 00:0a:95:a0:c8:9c
- 

Paul Krzyzanowski • Distributed Systems

---

## Naming things

- User names
    - Login, email
  - Machine names
    - rlogin, email, web
  - Files
  - Devices
  - Variables in programs
  - Network services
- 

Paul Krzyzanowski • Distributed Systems

---

## Naming Service

Allows you to look up names

Might be implemented as

- Search through file
  - Client-server program
  - Database query
  - ...
- 

Paul Krzyzanowski • Distributed Systems

---

## What's a name?

**Name:** identifies what you want

- Naming service often returns an *address* when we query it

**Address:** identifies where it is

**Route:** identifies how to get there

**Binding:** associates a name with an address

- "choose a lower-level-implementation for a higher-level semantic construct"

*RFC 1498: Inter-network Naming, addresses, routing*

---

Paul Krzyzanowski • Distributed Systems

---

## Names

Need names for:

- **Services:** e.g., time of day
- **Nodes:** computer that can run services
- **Paths:** route
- **Objects within service:** e.g. files on a file server

Naming convention can take any format

- Hopefully one that will suit application and user
  - E.g., human readable names for humans, binary identifiers for machines
- 

Paul Krzyzanowski • Distributed Systems

## Uniqueness of names

---

Easy on a small scale

Problematic on a large scale

Hierarchy allows uniqueness to be maintained

**compound name**: set of atomic names connected with a name separator

---

Paul Krzyzanowski • Distributed Systems

## Terms: Naming convention

---

Naming system determines syntax for a name

- Unix file names:  
Parse components from left to right separated by /  
`/home/paul/src/gps/gui.c`
- Internet domain names:  
Ordered right to left and delimited by .  
`www.cs.rutgers.edu`
- LDAP names  
Attribute/value pairs ordered right to left, delimited by commas:  
`cn=Paul Krzyzanowski, o=Rutgers, c=US`

---

Paul Krzyzanowski • Distributed Systems

## Terms: Context

---

- A particular set of **name** → **object** bindings
- Each context has an associated naming convention
- A name is *always* interpreted relative to some context
  - E.g., directory `/usr` in **UNIX file system**

---

Paul Krzyzanowski • Distributed Systems

## Terms: Naming System

---

Connected set of contexts of the same type (same naming convention) and provides a common set of operations

For example:

- System that implements DNS
- System that implements LDAP

---

Paul Krzyzanowski • Distributed Systems

## Terms: Name space

---

Set of names in the naming system

For example,

- Names of all files and directories in a UNIX file system

---

Paul Krzyzanowski • Distributed Systems

## Terms: Resolution

---

Name lookup

return the underlying representation

For example,

- `www.rutgers.edu` → `128.6.4.5`

---

Paul Krzyzanowski • Distributed Systems

## Directory Service

---

Naming services often extended with **directory service**

- Associates names with objects
- Allows objects to have attributes
- Can search based on attributes

For example,

- Netscape directory: general-purpose directory service based on LDAP
- Directory can be object store:
  - Look up printer object and send data stream to it

Paul Krzyzanowski • Distributed Systems

## Name resolution

---

To send data to a service:

1. Find a node on which the service resides (service name resolution)
2. Find an address (or network attachment point) for that node (node name location)
3. Find a path from this location to the service (routing service)

Paul Krzyzanowski • Distributed Systems

## Name resolution

---

E.g., access "paul's service":

### **File lookup:**

"paul's service" → cs.rutgers.edu: 1234

### **DNS lookup:**

cs.rutgers.edu → 128.6.4.2

### **ARP resolution:**

128.6.4.2 → 08:00:20:90:9c:23

### **IP routing:**

route: remus → lcsr-gw → aramis

Paul Krzyzanowski • Distributed Systems

## Binding

---

### **Static binding**

- Hard-coded

### **Early binding**

- Look up binding before use
- Cache previously used binding

### **Late binding**

- Look up just before use

Paul Krzyzanowski • Distributed Systems

## Example: DNS

---

Internet Domain Name Service

- Maps machine names ([www.rutgers.edu](http://www.rutgers.edu)) to IP addresses (128.6.4.5)

In the past:

- Search /etc/hosts for machine name
- File periodically downloaded from Network Information Center (NIC) at the Stanford Research Institute (SRI)

Paul Krzyzanowski • Distributed Systems

## Internet Domain Name Space

---

Tree structure

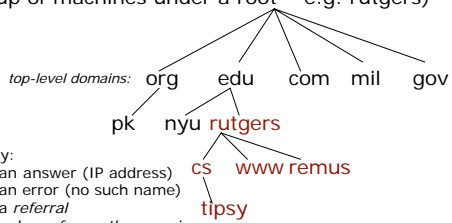
- Each node has resource information associated with it
- **owner**: domain name whose resource record is found
- **type of resource**:
  - Host address (A)
  - Alias name (C)
  - Name server for domain (NS)
  - Mail server (MX)
- **TTL** (time to live) → for **caching**
- **Relevant data** (e.g., address)

Paul Krzyzanowski • Distributed Systems

## Domain Name Server

### Essential task

Answer queries about data in its *zone*  
(group of machines under a root – e.g. rutgers)



A query may:

1. Return an answer (IP address)
2. Return an error (no such name)
3. Return a *referral*
4. *Recursively perform other queries*

Paul Krzyzanowski • Distributed Systems

## Sample Query

- Rutgers registers rutgers.edu with domain registry
  - educause.net for .edu domain
  - See [internic.net](http://internic.net) for ICANN-accredited list of registrars for top-level domains
- Name and associated name server info loaded to **root name servers**
  - 13 computers: replicated information
  - Contain addresses for all registries of top-level domains

Paul Krzyzanowski • Distributed Systems

## Sample Query

Submit query to a local *DNS resolver*:

1. query(cs.rutgers.edu) → root name server
2. *referral* to *edu* name server (e.g. A\_ROOT\_SERVERS\_NET)
3. query(cs.rutgers.edu) → edu name server (198.41.0.4)
4. *referral* to *rutgers.edu* name server:
  - DNS1.rutgers.edu
  - DNS2.rutgers.edu
5. query(cs.rutgers.edu) → rutgers name server (165.230.144.131) ♂ often located at ISP
6. rutgers name server returns 128.6.4.2

Paul Krzyzanowski • Distributed Systems

## DNS

### BIND

- Implementation of DNS provided by the Internet Software Consortium ([www.isc.org](http://www.isc.org))

Programs to perform queries:

- dnsquery, nslookup, dig, host

Paul Krzyzanowski • Distributed Systems

## Naming: files

File system maps file pathname

/home/paul/src/map.c

↓  
*namei* in kernel

major=3, minor=6, inode=6160

Paul Krzyzanowski • Distributed Systems