Naming

470410-1
Spring 2017
3/30/2017
Kyoung Shin Park
Multimedia Engineering
Dankook University

Outline

- Naming
- Approaches to naming
- Other naming issues
- Alternative references on naming

Name

- Name
  - Identifies what you want
- Address
  - Identifies where it is
- Route
  - Identifies how to get there
- Binding
  - The association of a name with the object
  - "choose a lower-level implementation for a higher-level semantic construct"

Naming

- Naming
  - Process of mapping a name to an object
- Examples
  - User name: used for system login, email, chat
  - Machine name: used for ssh, email, web
  - Filename/Pathname
  - Device name
  - Objects, functions, variables in programs
  - Network services

Naming

- Name Resolution
  - "The process of mapping a name to an object, or in the case of replication, multiple objects"
- Name Space
  - "A collection of names which may or may not share an identical resolution mechanism"
- Location Independent
  - A file can be moved without changing the filename
- Location Transparent
  - Filename does not tell where the file is located

Location Transparency

- Must be provided via global naming
- Dependent on a name being location independent, (i.e., a universal name)
- E.g. phone number (+82-42-550-3469)
  - Routing +82-(42)-550-3469
    - 82 = country code
    - 042 = area code
    - 550 = exchange
- E.g. home street address (Multimedia Engineering, Dankook University, 119, Dandae-ro, Dongnam-gu, Cheonan-si, Chungnam, 31116, S. Korea)

File Naming

- On a PC, the filename consists of a drive letter followed by a pathname
  - G:₩MM₩courses₩DistributedMultimediaService2016₩lecture4.pdf
- In Unix, the filename does not contain a drive letter, but the mount table enables the OS to discover on what drive the file is located
  - /home/park/test.txt
  - File pathname changes if you move the object

Process Naming

- Processes that want to communicate must have a way to refer to each other.
- Direct
  - send (P, message) send a message to process P
  - receive (Q, message) receive a message from process Q
  - receive (id, message) receive a message from any process
- Indirect
  - send (A, message) send a message to mailbox A
  - receive (A, message) receive a message from mailbox A

System Naming

- The first component of network communication is the naming of the systems in the network.
- Connected set of context of the same type (same naming convention) along with a common set of operations
- E.g. system that implements **DNS (Domain Name System)**
- E.g. system that implements **LDAP (Lightweight Directory Access Protocol)**

Naming Convention

- Naming system determines syntax for names
- Naming convention can take any format
  - Ideally one that will suit the application and user
  - UNIX file names:
    - Parse components from left to right separated by **/**
    - /home/park/test.txt
  - Internet domain names
    - Ordered right to left and delimited by **.**
    - dis.dankook.ac.kr
  - LDAP names
    - Attribute/value pairs ordered right to left, delimited by **,**
    - cn=Kyoung Park, o=dankook, c=kr

Namespace

- A container for a set of names in the naming system
- A namespace has a scope
  - Scope = region where the name exists & refers to the object
  - E.g. Names of all files in a directory
  - E.g. All domain names within dankook.ac.kr
  - E.g. Java package, local variables
- A namespace may be tree structured (hierarchical)
  - Fully-qualified or hierarchical names may be used to identify names outside the local namespace
  - Global namespace = root of the hierarchical tree

Name Resolution

- Resolution = name lookup
  - Return the underlying representation of the name
  - Look up the **binding** of the name to its object
- E.g. dis.dankook.ac.kr -> 203.237.220.128

```
~>nslookup dis.dankook.ac.kr
Server:    164.124.101.2
Address:  164.124.101.2#53

Name:    dis.dankook.ac.kr
Address:  203.237.220.128
```

Name Service

- The service that performs name resolution
- Allows you to resolve names
  - Looking up a name gives the corresponding address as a response
- Can be implemented as
  - Search through file
  - Database query
  - Client-server program (name server) – may be distributed
  - …

Directory Service

- Extension of name service
  - Associates names with objects
  - Allows objects to have attributes
  - Can search based on attributes
- E.g. LDAP (Lightweight Directory Access Protocol)
  - Directory can be an object store

Global Naming Considerations

- A global name space requires
  - Name resolution
  - Location resolution
- Name resolution
  - Maps symbolic filenames to computer filenames
- Location resolution
  - Involves mapping global names to a location
- This can be difficult if name transparency and location transparency are both supported

Naming Approaches

- Add hostname to names of files on that host
  - Provides unique names
  - Loses network transparency
  - Loses location transparency
  - Moving file to a different host causes change of filename
    - Possible changes to applications using that file
  - Easy to find a file

Naming Approaches

- Mount remote directories onto local directories
  - To do the mount, need to know host
  - Once mounted, references are location transparent
  - Can resolve filenames easily
  - However, a difficult approach to do
    - Not fault tolerant
    - File migration requires lots of updates

Naming Approaches

- Use a single global directory
  - Does not have disadvantages of previous approaches
  - Variations found in Sprite and Apollo
  - Need a single computing facility or a few with lots of cooperation
    - Need system-wide unique filenames
  - Not good on a heterogeneous system
  - Not good on a wide geographic system

Other Naming Issues

- Contexts
  - Used to partition a namespace
    - To avoid problems with system-wide unique names
    - Geographical, organizational, etc.
  - A name space in which to resolve a name
  - A filename has two parts
    - Context
    - Local filename
  - Almost like another level of directory
  - Used in X-Kernel logical file system

Other Naming Issues

- Name Server
  - Maps names to files and directories
  - Centralized
    - Easy to use
    - A bottleneck
    - Not fault tolerant
  - Distributed
    - Servers deal with different domains
    - Several servers may be needed to deal with all the components in a filename

Removing Unreferenced Entities

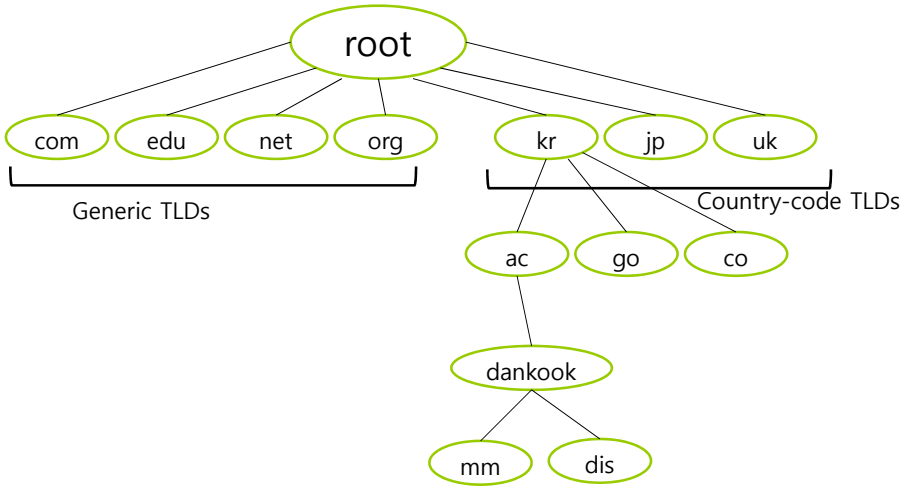
- Problem of unreferenced objects
- Reference counting
- Reference listing
- Identifying unreachable entities

Case Study: DNS (Domain Name System)

- IP addresses are distributed hierarchically
- Internet Assigned Numbers Authority (IANA) at the top
  - IANA is currently run by ICANN (Internet Cooperation for Assigned Names and Numbers)
- RIR (Regional Internet Registry)
  - Manages the allocation and registration of Internet number resources within a particular region of the world

RIR (Region Internet Registry) Map

Domain Name Hierarchy



Top Level Domains (TLDs)

- There are currently 1097 top-level domains
- Each top-level domain has an administrator assigned to it
- Assignment is delegated to various organizations by the Internet Assigned Numbers Authority (IANA)
- IANA keeps track of the root servers
- http://www.iana.org/domains/root/db
- .kr -> Korea Internet & Security Agency (KISA)

Domain Name System (DNS)

- Distributed database: a hierarchy of name servers
- DNS is an application-layer protocol
  - Name-address resolution is handled at the edge
  - The network core is unaware of hostnames
  - There is no special relationship between names and addresses

Domain Name System (DNS)

- DNS provides
  - Name to IP address translation
  - Aliasing of names (called canonical names)
  - Identification of name servers
  - Mail server names
  - Load distribution
    - Multiple name servers may handle a query for a domain
    - Caching: store past lookups
    - Ability to provide a set of IP address for a name

Authoritative DNS Server

- An authoritative name server is responsible for answering queries about its zone
  - Provides real answers vs. cached answers
  - Configured by the administrator
- Zone
  - Group of machines under a node in the tree
  - E.g. dankook.ac.kr

Authoritative DNS Server

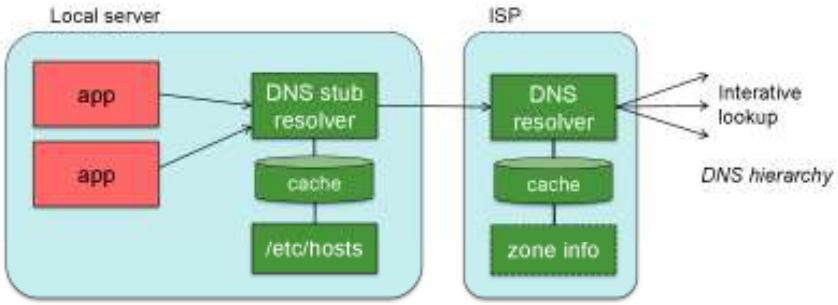
- DNS server returns answers to queries
- Key data that a DNS server maintains (partial list)

Information	Abbreviation	Description
Host	A	Host address (name to address) including name, IP address, time-to-live (TTL)
Canonical name	CNAME	Name for an alias
Mail exchanger	MX	Host that handles email for the domain
Name server	NS	Identifies the name server for the zone: tell other servers that yours is the authority for info within the domain
Start of Zone Authority	SOA	Specifies authoritative server for the zone. Identifies the zone, time-to-live, and primary name server for the zone

DNS Resolver

- The client side of DNS is called a DNS resolver.
  - Not really a part of DNS hierarchy
  - Acts as an intermediary between programs that need to resolve names and the name servers
  - A resolver is responsible for performing the full resolution of the query
- Where are DNS resolvers?
  - Each local system has one: that's what applications contact
    - Local cache; may be a process or a library
    - On Linux & Windows, these are limited DNS servers (called stub resolvers)
  - ISPs (and organizations) run them on behalf of their customers
- Resolvers cache past lookups – they are not responsible for zones

DNS Resolver



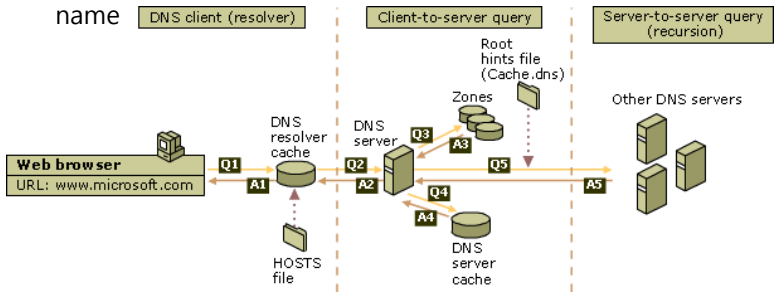
Local stub resolver:
- check local cache
- check local hosts file
- send request to external resolver

E.g., on Linux: resolver is configured via the /etc/resolv.conf file

External resolver
- DNS server that accepts recursion
- Running at ISP, Google Public DNS, OpenDNS, etc.

DNS Query

- When a DNS client needs to look up a name used in a program, it queries DNS servers to resolve the name.
- DNS query process
  1. Local resolver: A name query begins at a client computer and is passed to a resolver (DNS client service) for resolution
  2. Query a DNS server: When the query cannot be resolved locally, DNS servers can be queried as needed to resolve the name



References

- www.cs.colostate.edu/~cs551/CourseNotes/Naming/NamingTOC.html
- https://www.cs.rutgers.edu/~pxk/417/notes/naming.html