Advanced Operating Systems Lecture notes

Dr. Dongho Kim
Dr. Tatyana Ryutov
University of Southern California
Information Sciences Institute

CSci555: Advanced Operating Systems
Lecture 4 – September 16 2005
Naming and Binding

Dr. Dongho Kim
Dr. Tatyana Ryutov
University of Southern California
Information Sciences Institute
(lecture slides written by Dr. Clifford Neuman)

Naming Concepts

- **Name**
  - **What you call something**
- **Address**
  - **Where it is located**
- **Route**
  - **How one gets to it**
    What is http://www.isi.edu/~dongho ?

- **But it is not that clear anymore, it depends on perspective. A name from one perspective may be an address from another.**
  - **Perspective means layer of abstraction**

What are the things we name

- **Users**
  - **To direct, and to identify**
- **Hosts (computers)**
  - **High level and low level**
- **Services**
  - **Service and instance**
- **Files and other "objects"**
  - **Content and repository**
- **Groups**
  - **Of any of the above**

How we name things

- **Host-Based Naming**
  - **Host-name is required part of object name**
- **Global Naming**
  - **Must look-up name in global database to find address**
  - **Name transparency**
- **User/Object Centered Naming**
  - **Namespace is centered around user or object**
- **Attribute-Based Naming**
  - **Object identified by unique characteristics**
  - **Related to resource discovery / search / indexes**

Namespace

- **A name space maps:**

$$\Sigma^* \rightarrow X_\epsilon O$$

  At a particular point in time.

- **The rest of the definition, and even some of the above, is open to discussion/debate.**
- **What is a "flat namespace"**
  - **Implementation issue**

Case Studies

- **Host Table**
  - **Flat namespace (?)**
  - **Global namespace (?)**
- **Grapevine**
  - **Two-level, iterative lookup**
  - **Clearinghouse 3 level**
- **Domain name system**
  - **Arbitrary depth**
  - **Iterative or recursive(chained) lookup**
  - **Multi-level caching**

```
GetHostByName(usc.arpa){
    scan(host file);
    return(matching entry);
}
```

Case Studies

- **Host Table**
  - **Flat namespace (?)**
  - **Global namespace (?)**
- **Grapevine**
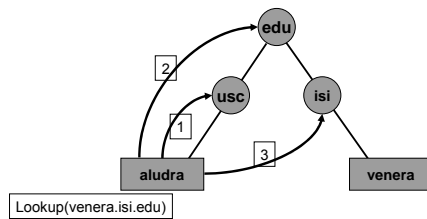 - **Two-level, iterative lookup**
  - **Clearinghouse 3 level**
- **Domain name system**
  - **Arbitrary depth**
  - **Iterative or recursive(chained) lookup**
  - **Multi-level caching**

GetHostByName(host.sc)

gv es   sc
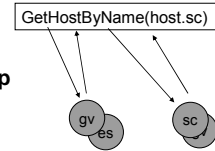
Domain Name System

Iterative query

Lookup(venera.isi.edu)

Caching in the Domain Name System

Chained query
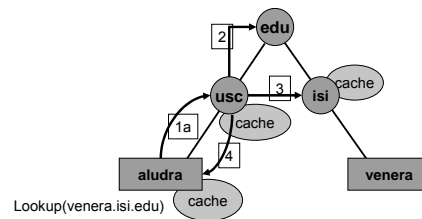
Lookup(venera.isi.edu)

Scalability of naming

- **Scalability**
  - **Ability to continue to operate efficiently as a system grows large, either numerically, geographically, or administratively.**
- **Affected by**
  - **Frequency of update**
  - **Granularity**
  - **Evolution/reconfiguration**
- **DNS characteristics**
  - **Multi-level implementation**
  - **Replication of root and other servers**
  - **Multi-level caching**

Closure

- **Closure binds an object to the namespace within which names embedded in the object are to be resolved.**
  - **"Object" may as small as the name itself**
    - **GNS binds the names to namespaces**
    - **Prospero binds enclosing object to multiple namespaces**
    - **Tilde and quicksilver bind users to namespaces**
    - **NFS mount table constructs system centered namespace**
  - **Movement of objects can cause problems**
    - **When closure is associated with wrong entity**

Other implementations of naming

- Broadcast
  - Limited scalability, but faster local response
- Prefix tables
  - Essentially a form of caching
- Capabilities
  - Combines security and naming
  - Traditional name service built over capability based addresses

Advanced Name Systems

- DEC's Global Naming
  - Support for reorganization the key idea
  - Little coordination needed in advance
- Half Closure
  - Names are all tagged with namespace identifiers
    - DID - Directory Identifier
    - Hidden part of name - makes it global
    - Upon reorganization, new DID assigned
    - Old names relative to old root
  - But the DID's must be unique - how do we assign?

Prospero Directory Service

- Multiple namespace centered around a "root" node that is specific to each namespace.
  - Closure binds objects to this "root" node.
- Layers of naming
  - User level names are "object" centered
  - Objects still have an address which is global
  - Namespaces also have global addresses
- Customization in Prospero
  - Filters create user level derived namespaces on the fly
  - Union links support merging of views

Resource Discovery

- Similar to naming
  - Browsing related to directory services
  - Indexing and search similar to attribute based naming
- Attribute based naming
  - Profile
  - Multi-structured naming
- Search engines
- Computing resource discovery

The Web

- Object handles
  - Uniform Resource Identifier (URI's)
  - Uniform Resource Locators (URL's)
  - Uniform Resource Names (URN's)
- XML
  - Definitions provide a form of closure
    - Conceptual level rather than the "namespace" level.

LDAP

- Manage information about users, services
  - Lighter weight than X.500 DAP
    - Heavier than DNS
  - Applications have conventions on where to look
    - Often data is duplicated because of multiple conventions
  - Performance enhancements not as well defined
    - Caching harder because of less constrained patterns of access