Name Services & Directory Services

莊 裕 澤
Yuh-Jzer Joung
Dept. of Information Management
National Taiwan University

2023/3/9

Outline

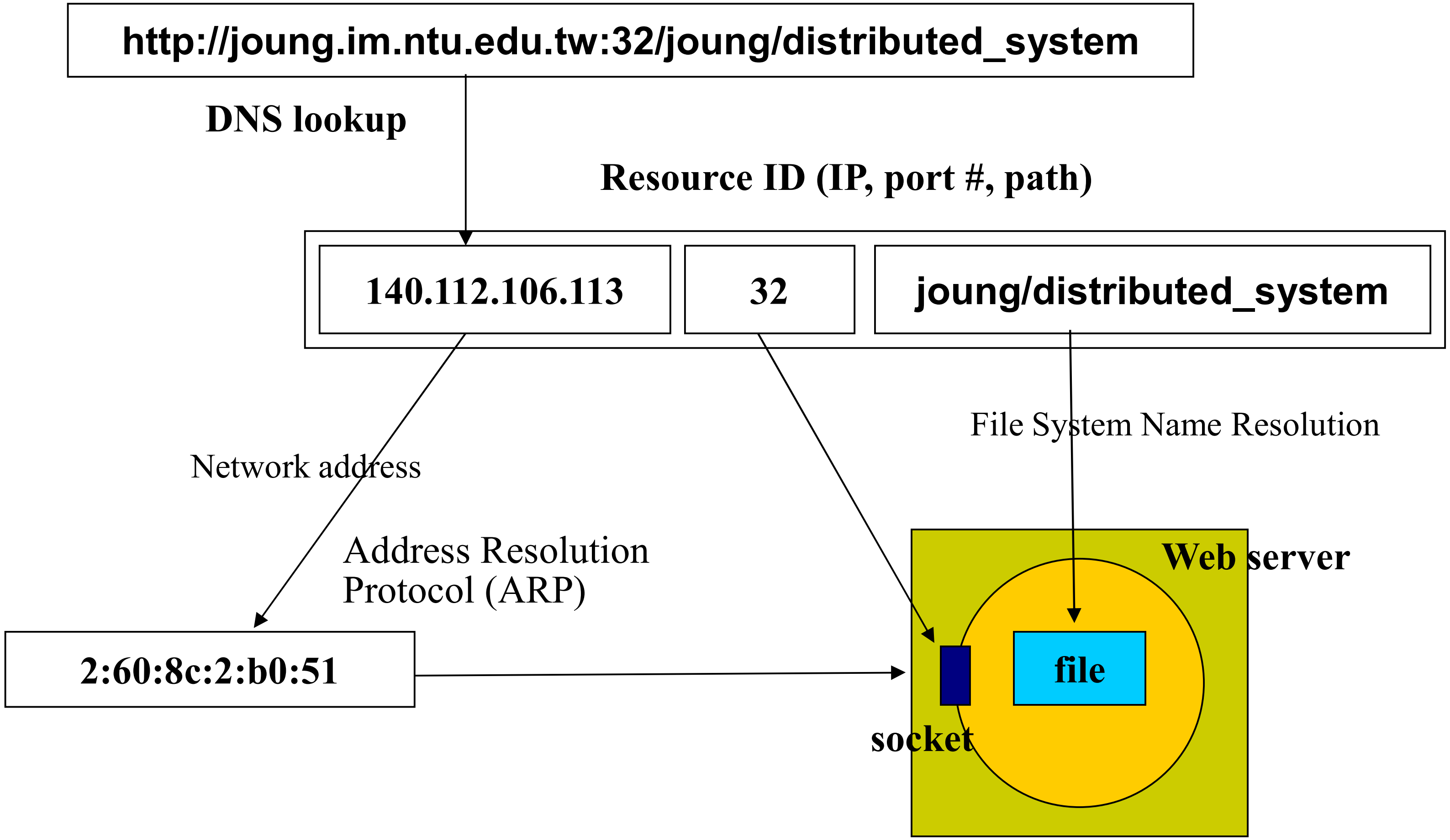
- ❑ Overview
- ❑ Domain Name Service
- ❑ Directory services
- ❑ Jini

Name Services

Names are important for identifying entities:

- communication
  - e.g., mail joung@im.ntu.edu.tw
  - @yuhjzer
- resource sharing
  - \\joung\printer
  - URLs: http://joung.im.ntu.edu.tw/joung/

Example: URL


http://joung.im.ntu.edu.tw:32/joung/distributed_system
DNS lookup
Resource ID (IP, port #, path)
140.112.106.113
32
joung/distributed_system
File System Name Resolution
Network address
Address Resolution Protocol (ARP)
2:60:8c:2:b0:51
Web server
file
socket

Some Terminology

- *binding*: the association between a name and an object.
- *attribute*: the value of a property associated with an object.
  - names are typically bound to attributes of the named objects
  - to *resolve* a name is to get the attributes of the named object.
- *naming context*: a set of bindings between textual names and attributes of objects.
- A name service stores a collection of one or more naming contexts. Main operations:
  - *resolve* a name: to look up attributes from a given name.
  - create, delete, list binding
  - add/delete contexts

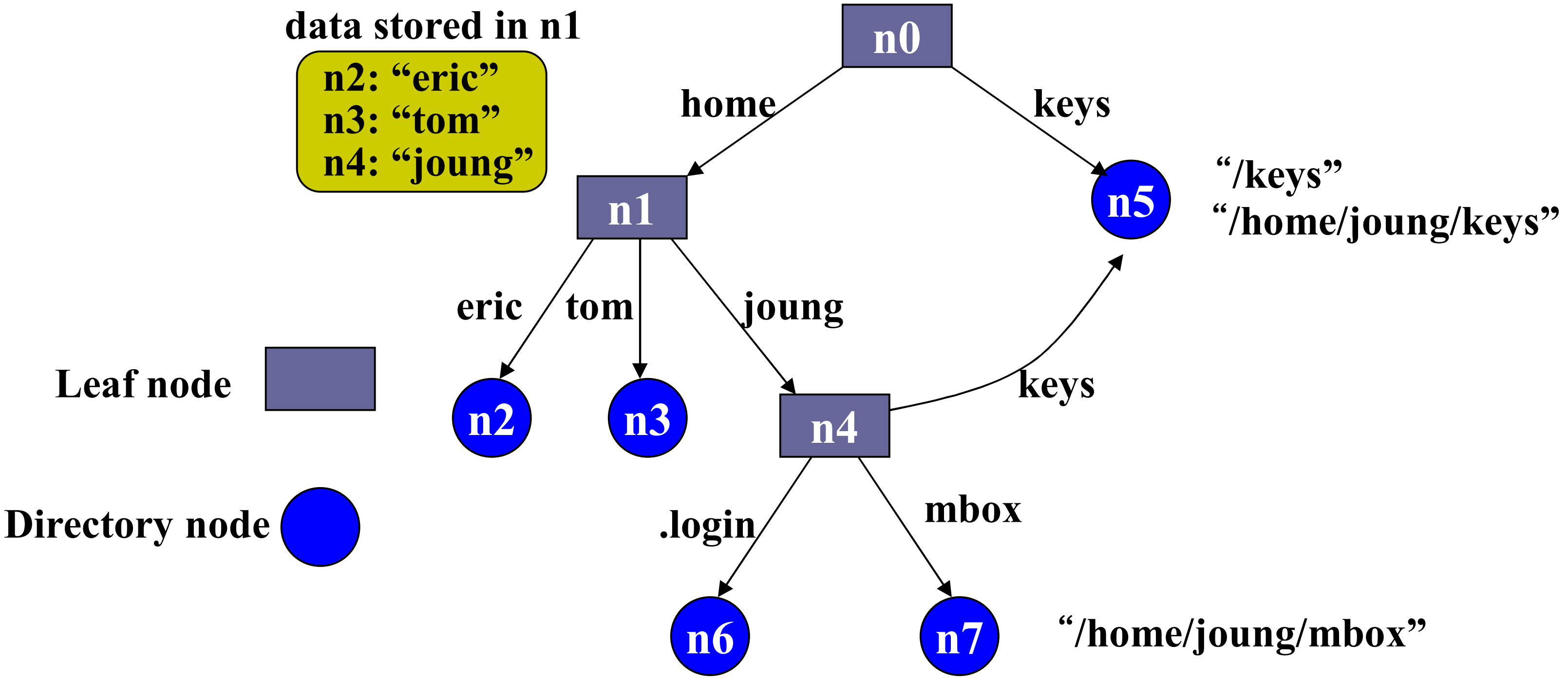
- Addresses: the name of an access point to an entity
    - Addresses are physically bounded to entities, and an entity may have more than one addresses.
- Identifiers: a name that can uniquely identify an entity
- Human-Friendly Names: names that are tailored to be used by humans.
    - Addresses and identifiers are often represented in machine-readable form only.

“Names” vs. Addresses

- why use “names” instead of “addresses” to refer to entities?
  - Access points may change over time
  - Some entities may have more than one access points (e.g., Web services)
  - People use names for ease of reference, while machines may use addresses to optimal performance.

Name Space

***Name space*:** the set of all valid names. A name space can be represented as a **labeled, directed graph** (and usually **acylic**).



data stored in n1

n2: “eric”
n3: “tom”
n4: “joung”

Leaf node

Directory node

Some distinction

- ***Absolute path name***:
  - /home/joung/courses/distributed_systems
- ***Relative path name*:**
  - .. /courses/distributed_systems
  - (domain names do not recognize relative names)
- ***Global name***: a name that denotes the same entity no matter where that name is used.
- ***Local name***: a name whose interpretation depends on where that name is being used.

Alias

- ***alias*:** allowing more than one names, e.g.,
  - Unix **symbolic links** and **hard links**.
  - Windows 的「捷徑」
  - URL shortening
    - *www.cdk5.net* for *cdk5.net*.
    - http://bit.ly/ctqjvH for https://cdk4.net/additional/rmi/programCode/ShapeListClient.java
    - http://youtu.be/eHnAkbB-IYs

bitly picsee

Name Resolution

- Name resolution: given a name, return information stored in the node referred to by that name.
  - /home/joung/courses/distributed_systems/syllabus.doc
  - File access in P2P networks:
    - Given a file name, retrieve the file from the network
  - Service invocation in Web services
- Simplest Solution:
  - Each computer has a unique name and has a built in table of name to address translation (mapping)
    - Not scalable
  - We will discuss hierarchical, distributed solutions later on

General Name Services Requirements

- ❑ To handle an essentially arbitrary number of names and to serve an arbitrary number of administrative organizations.
- ❑ A long lifetime
- ❑ High availability
- ❑ Fault isolation
- ❑ Tolerance of mistrust

Some Name Services

- ***URL***: the address of a file/resource accessible on the Internet.
- ***DNS*:** maps domain names to the attribute of a host computer, e.g., its IP address, and a reference to a mail server.
- ***X.500*:** a directory service that can be used to map a person's name onto attributes including their email address and telephone number.
- ***CORBA Naming Service*:** map the name of a remote object onto its remote object reference.

The URL (Uniform Resource Locators)

- The address of a file/resource accessible on the Internet.
- A URL contains
  - the protocol required to access the resource
  - a domain name that identifies a specific computer
  - a file pathname on the computer
    - http://joung.im.ntu.edu.tw/teaching/distributed_systems/2003EMBA/default.htm
    - mailto:joung@ntu.edu.tw
- URLs are a particular type of Uniform Resource Identifier (URI).
- Dangling links are annoying.
  - **Error 404: Document not Found**

The URN (Uniform Resource Name)

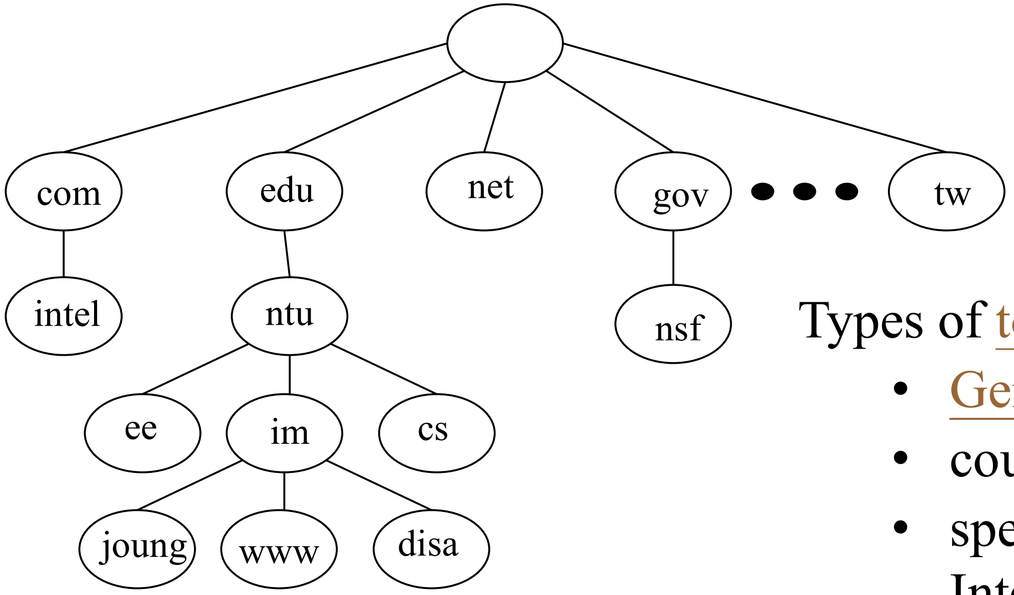
- A type of URI intended to solve the dangling links problem. A URN associates with an Internet resource with a name that has persistent significance
  - The use of URN can expect that someone else (or a program) will be able to find the resource.
  - **urn:nameSpace:nameSpace-specificName**
    - *urn: ISBN: 0-201-62433-8*
    - *urn:doi:10.555/music-pop-1234* (*doi* stands for *Digital Object Identifier*)
      - http://dx.doi.org/10.1016/j.comnet.2006.05.010
    - *urn:dcs.gormenghast.ac.uk:TR2000-56*

Domain Name System (DNS)

- System to map names to IP addresses
  - i.e. joung.im.ntu.edu.tw → 140.112.106.103
- The hierarchical naming scheme using in the Internet
  - Specifies the *syntax* for names
  - Specifies the *rules* for delegating authority over names
  - Specifies the *implementation* of an efficient, distributed algorithm for mapping between IP addresses and names
- DNS is also a global hierarchical naming system:
  - Naming is based on organizational boundaries and not physical networks
  - A distributed database system
- Developed in early 1980s

Domain Name System (DNS)

- A subtree in the DNS name is called a **domain**, and a path name to its root is called a **domain name**.



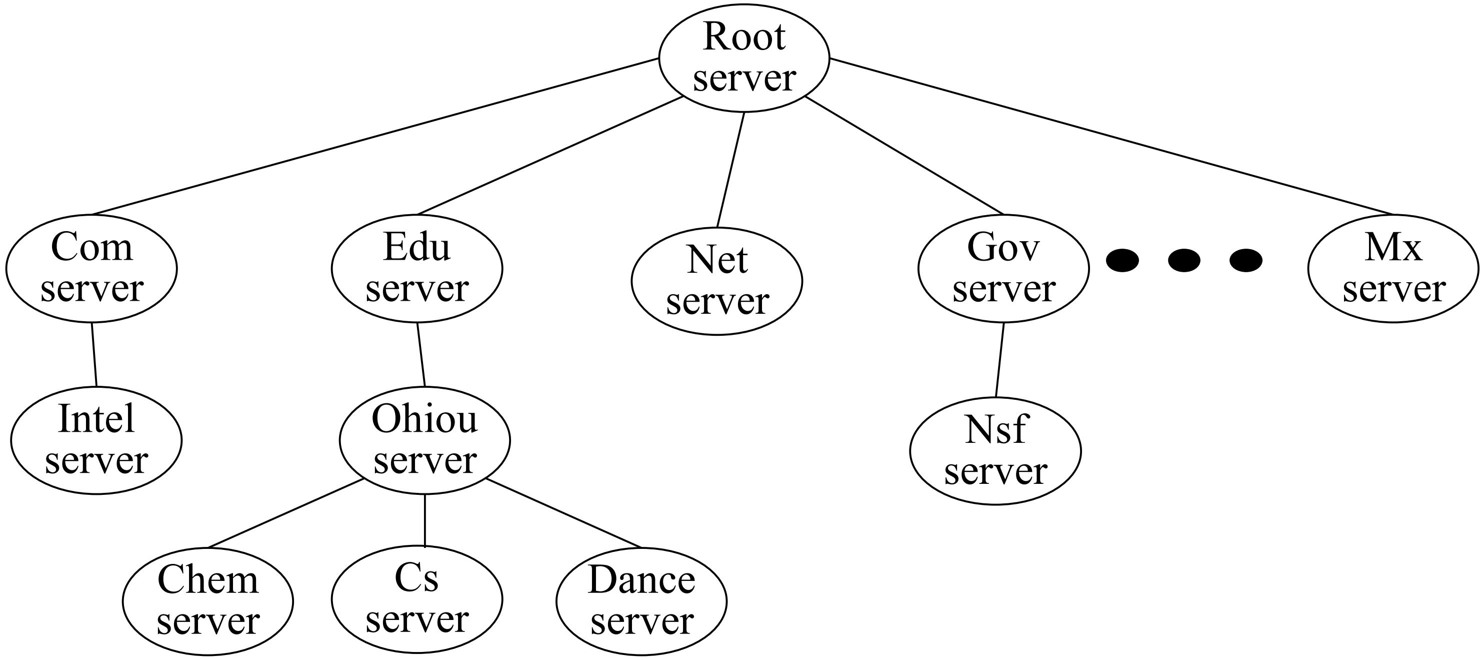
1,500+ TLDs (as of 2020.06)

Types of top-level domains (TLDs):

- Generic (com, gov, net, mil, biz, …)
- country code
- special top-level domain (.arpa) for Internet infrastructure.
- Sponsored (edu, gov, int, mil, coop, jobs,…)
- Test
- Restricted

DNS Name Servers

- ❑ Provide name-to-address mapping service.
- ❑ Individual servers contain all of the information for large portions of the naming hierarchy.
- ❑ Very few servers need to be contacted when resolving a name
- ❑ Root server is replicated

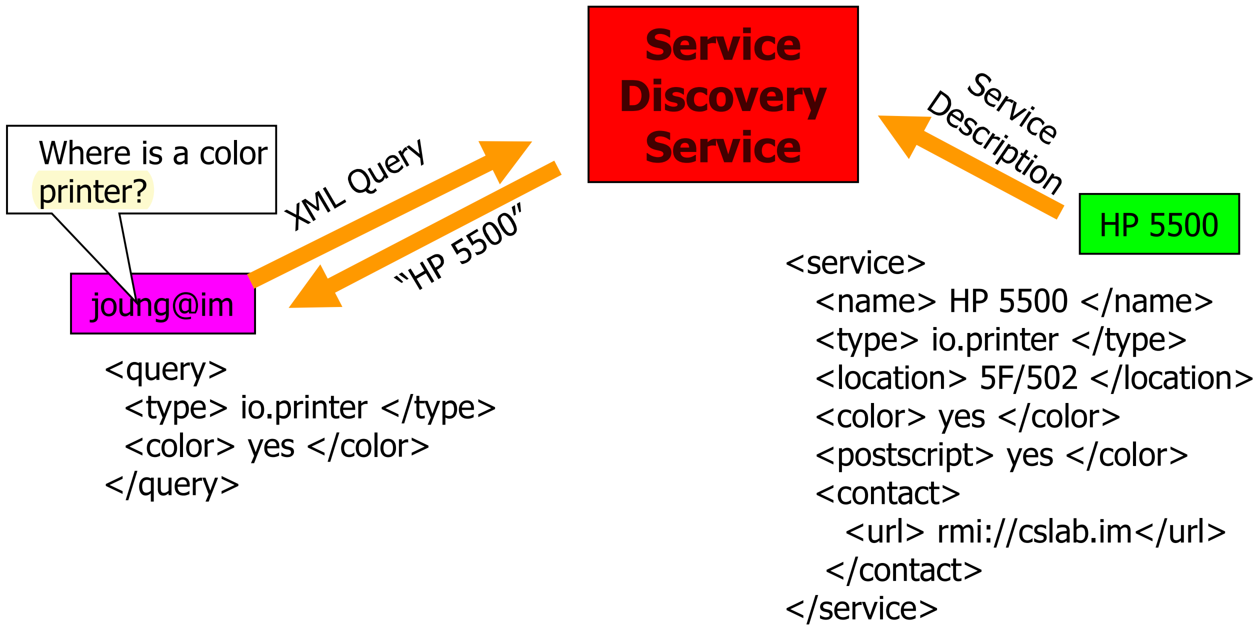


Directory Services

- A ***directory service*** stores collections of bindings between names and attributes and provides lookups services for attribute-based specifications.
  - e.g., X.500 directory services
- *Directory services* vs. *name services* is like *yellow-pages* services vs. *white pages* services.
  - E.g., "who's email is john@ntu.edu.tw"
- ***Discovery service*** is a directory service that registers the services provided in a spontaneous networking environment.
  - e.g., Jini

Why is Directory Service Important?

- Distributed Computing
- Ubiquitous Computing
- Pervasive Computing
  - All need to know where to locate a resource



Where is a color printer?

joung@im

XML Query

"HP 5500"

Service Discovery Service

Service Description

HP 5500

```
<query>
  <type> io.printer </type>
  <color> yes </color>
</query>
```

```
<service>
   <name> HP 5500 </name>
   <type> io.printer </type>
   <location> 5F/502 </location>
   <color> yes </color>
   <postscript> yes </color>
   <contact>
      <url> rmi://cslab.im</url>
    </contact>
</service>
```

Directory Services in the Internet

- A network accessible database with limited functionality:
  - Small amount of information in each request/reply.
  - Limited functionality (as compared to a complete database system)
- Some typical examples include:
  - telephone directories
  - lists of email addresses (or other network addresses).
- Each record is referenced by a unique key:
  - given a name, look up a phone number
  - given a name, look up a email address

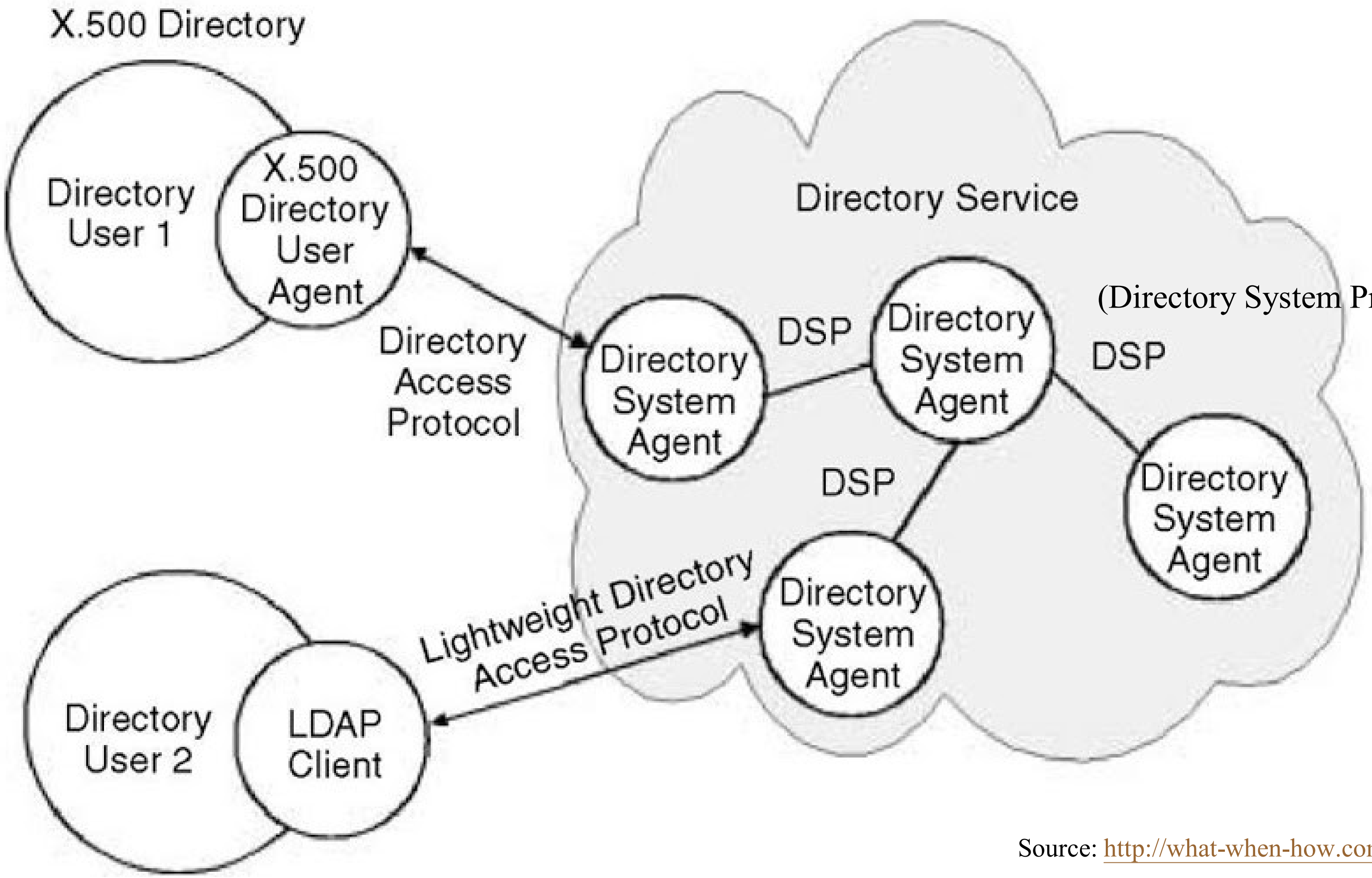
Actual standards (services)

- LDAP
- X.500
- Whois / Whois ++
- Netfind
- CCSO Nameserver (Ph) Architecture
- Referral Whois (RWhois)
- Windows Active Directory

X.500 Directory Service

- X.500 is a joint International Organization for Standardization (ISO) and International Telecommunications Union-Telecommunication Standardization Sector (ITU-T) standard for creating electronic **directories** that can function as part of a global directory (originated in 1988).
- The X.500 standard also defines the Directory Access Protocol (DAP), a protocol for accessing X.500 directories.
- Most used in Europe, and adopted by large organizations, but proven to be impractical for building a global directory.

X.500 Directory Service



(Directory System Protocol)

Source: http://what-when-how.com/

An X.500 DIB

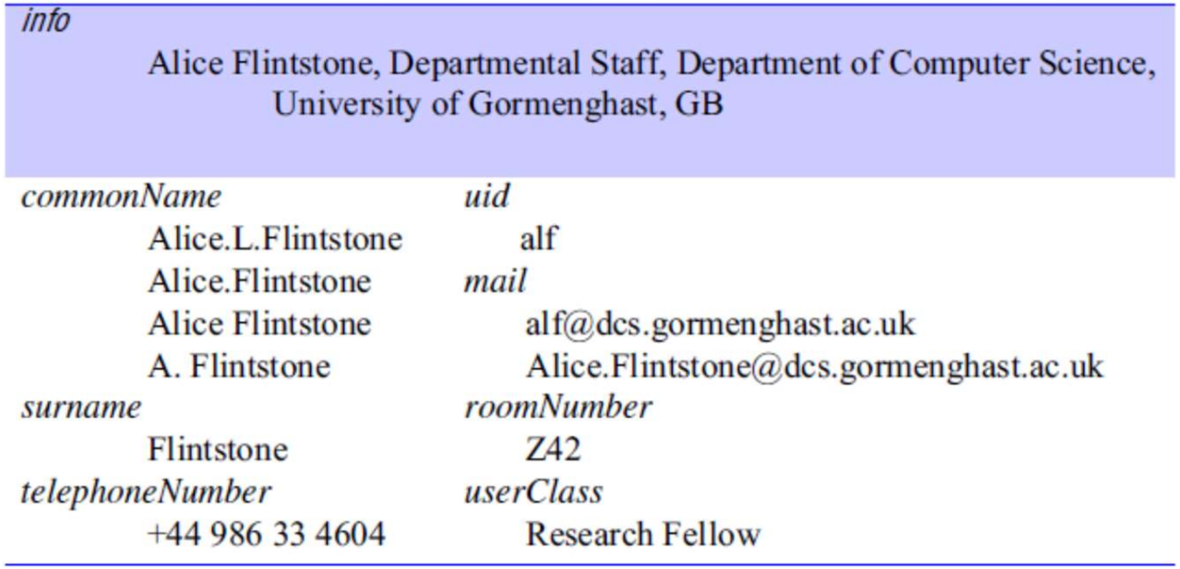
The information is held in a directory information base (DIB). Entries in the DIB are arranged in a tree structure called the directory information tree (DIT).

A DIB entry consists of a set of attributes, where an attribute has a *type* and one or more *values*. Examples of attribute type name: *countryName*, *organizationName*, *commonName*, *telephoneNumber*, *mailbox*, *objectClass*.

Two main access request types for the data:

Read: An absolute or relative name (a domain name in X.500 terminology) for an entry is given, together with a list of attributes to be read

Search: attribute-based access request such that a base name and a filter expression are supplied as arguments to find a list of (domain) names.

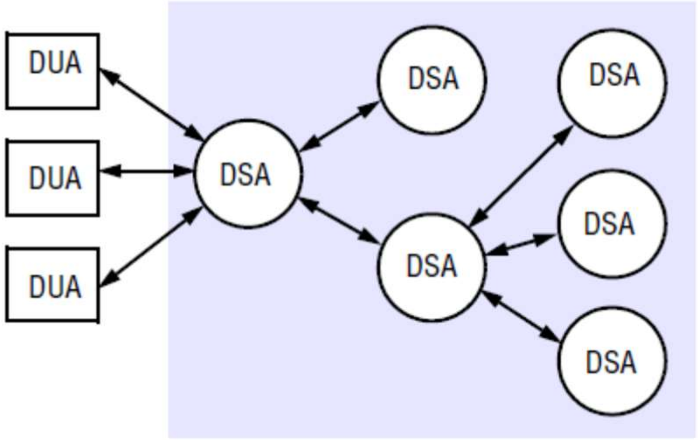


info

Alice Flintstone, Departmental Staff, Department of Computer Science, University of Gormenghast, GB

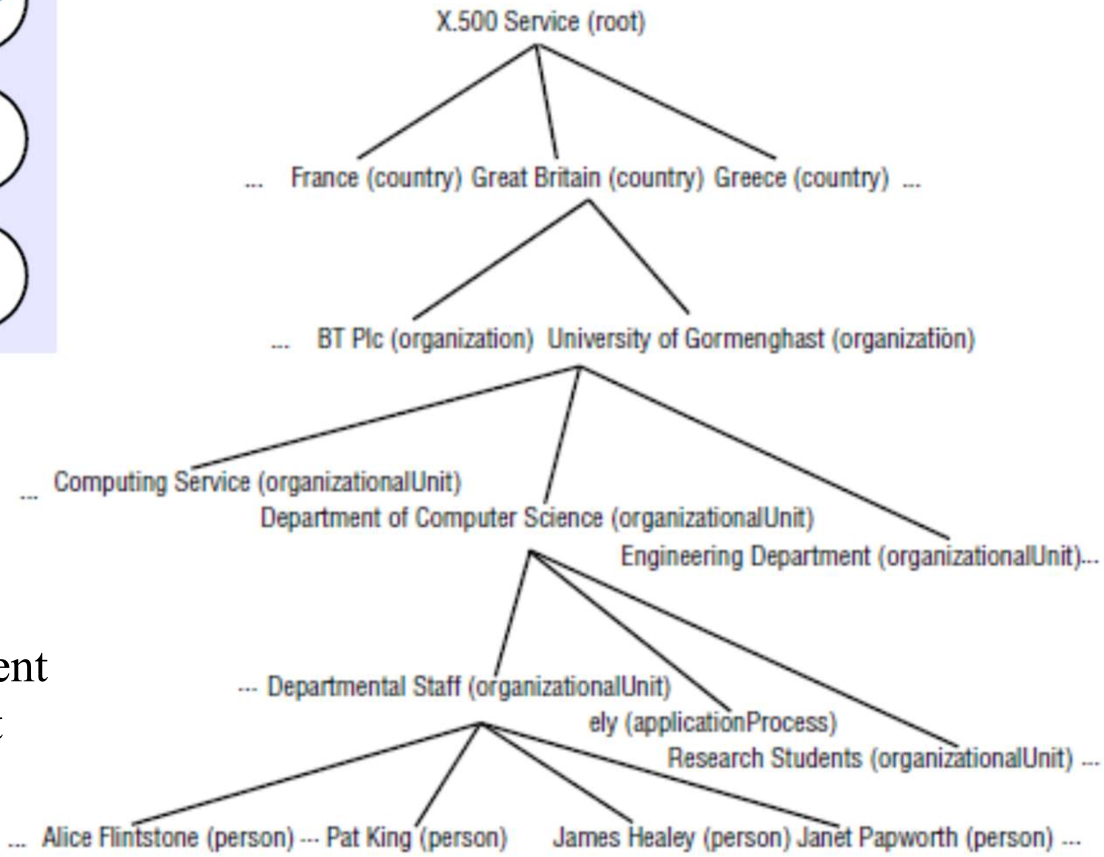
commonName	*uid*
Alice.L.Flintstone	alf
Alice.Flintstone	*mail*
Alice Flintstone	alf@dcs.gormenghast.ac.uk
A. Flintstone	Alice.Flintstone@dcs.gormenghast.ac.uk
surname	*roomNumber*
Flintstone	Z42
telephoneNumber	*userClass*
+44 986 33 4604	Research Fellow

X.500 Service Architecture and DIT



X.500 service architecture

DSA: Directory Service Agent
DUA: Directory User Agent


X.500 Service (root)
... France (country) Great Britain (country) Greece (country) ...
... BT Plc (organization) University of Gormenghast (organization)
... Computing Service (organizationalUnit)
Department of Computer Science (organizationalUnit)
Engineering Department (organizationalUnit)...
... Departmental Staff (organizationalUnit)
ely (applicationProcess)
Research Students (organizationalUnit) ...
... Alice Flintstone (person) ... Pat King (person) James Healey (person) Janet Papworth (person) ...

Part of the X.500 Directory Information Tree (DIT)

Source: Distributed Systems: Concepts and Design 5th ed., Ch.13, C. Coulouris et al., 5th ed., 2011.

Discussion of X.500

- The requirement of a global directory system is not clear.
  - Privacy
- Decisions about the scope of the information that will be provided in directories would need to be taken at national and international levels to ensure uniformity.
  - Politics

LDAP (Lightweight Directory Access Protocol)

- LDAP is a version of DAP that contains less code than DAP contains.
  - Originated at the University of Michigan.
- LDAP's original purpose was to provide PCs with TCP/IP access to X.500 directories.
  - 90% of the functionality of X.500
  - 10% of the cost
- LDAP 3 has since evolved to become more than an access protocol: LDAP 3 defines an extensible schema for a directory and for a protocol that can access LDAP 3-compliant directories.
  - A directory schema defines the object classes--or types of objects--that can be stored in the directory. LDAP 3’s extensible schema allows directory vendors to add object classes to the core schema defined in LDAP 3.

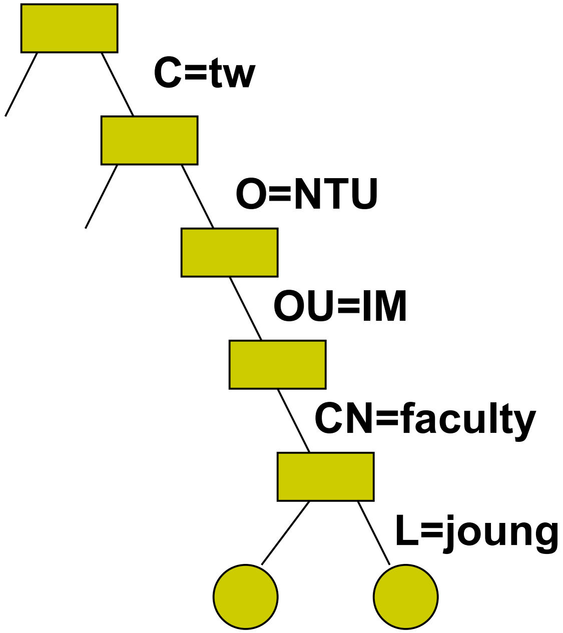
LDAP Data Representation

- Each record has a unique key called a ***distinguished name*** (DN for short).
- A distinguished name (RFC 1779) is meant to be used by humans (not just computers).
- Each DN is a sequence of components.
  - Each component is a string containing an attribute=value pair.
  - Example:
    - CN=Dave Hollinger,
    - OU=Grad Student,
    - O=Rensselaer Polytechnic University Institute,
    - C=US

  Could also be written:
    - CN=Dave Hollinger, OU=Grad Student,
    O=Rensselaer Polytechnic Institute, C=US

Component Names

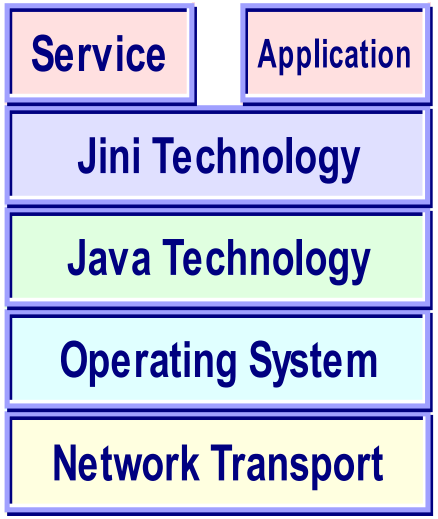
- The components can be anything, but there is a standard hierarchy used (for a *global* LDAP namespace):



C	***country name***
O	***organization name***
OU	***organizational unit***
CN	***common name***
L	***locality name***
ST	***state or province***
STREET	***street address***

Jini Discovery Service

- Publicly released on January 25, 1999 (later transferred to Apache under the project name “River”, and was retired and moved to Attic in 2022-02.)
  - Service centered
  - Fulfillment of original Java vision – groups of devices exchanging data and code.
  - The Jini vision: Network anything, anytime, anywhere!
  - “network plug and play”.
  - Built on top of Java
- Features:
  - Enable clients to find and use services
  - Enable services to advertise their availability
  - Allow access to resources despite mobility
  - Simplify configuration and maintenance of large groups of users, devices, and software



Service

Application

Jini Technology

Java Technology

Operating System

Network Transport

Jini Discovery Service

- A client in need of service ultimately downloads a Java object
- The object can either:
  - Provide the service locally (e.g., algorithmically)
  - Provide the service by invoking operations on a remote server, possibly using a private protocol
- To the client, there is no essential difference between these choices!
- Client can’t even determine location of provider of the service…

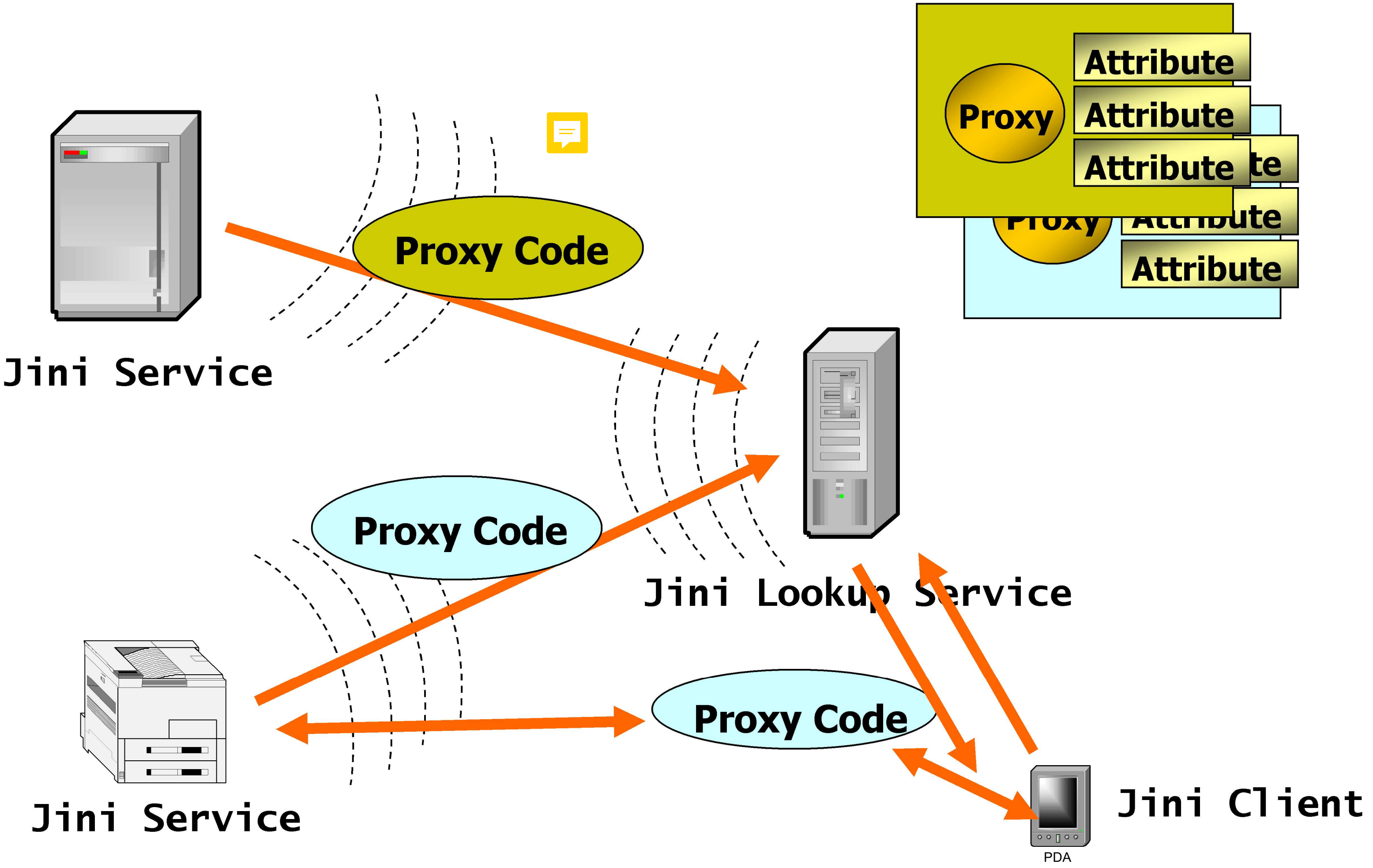
Jini Lookup Services

- Provides centralized registry of services
- Lookup service is a repository of Java objects
- Each object is downloadable and serves as a proxy between the client and the service
  - E.g., printer proxy
    - Knows how to contact and talk to a printing service
  - E.g., equation solver
    - Solves equation on remote server
- Lookup is based on object *interfaces*, not simply name/value pairs

Finding a Lookup Server: Protocols

- **Multicast request protocol**
  - UDP: Used to discover nearby lookup services
    - Useful both for clients who are searching…
    - …and services who want to serve…
- **Unicast discovery protocol**
  - TCP: Protocol for communication with a specific lookup service
- **Multicast announcement protocol**
  - UDP: Used to announce availability of a lookup service

Discovery and Lookup



Globally Unique Service Identifiers

- The very first time a service registers with a lookup service, it is assigned an ID
  - ID is globally unique
  - Service is required to remember forever
  - 60 of the 128 bits express the current system time in 100 ns increments since 1582
  - Remainder is random
    - 295,147,905,179,352,825,856 choices…
- Allows clients to determine if services registered at different lookups are the same
- Allows lookup servers to notice duplicate registrations

Other Jini Stuff

❑ Leases

- Leases provide support for highly dynamic configurations
- Services/client interest change…
- If lease on the lookup server expires, can garbage collect
- Lease duration is determined by the lookup server
- Client/service is responsible for *renew()*ing

❑ Transactions

- Provides a 2PC-based transaction system

❑ Distributed Events

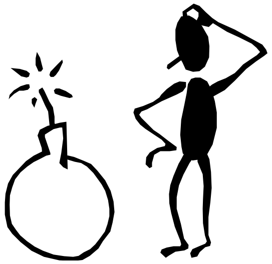
- (Used for notification of lookup and/or appropriate services becoming available)

Jini Security

- Digital signatures
  - Verify authenticity of downloaded code
- Encryption
- Security Manager uses access control lists
- A variety of actions can be controlled
- Security Manager can be subclassed and customized extensively
- By default, no security manager…

Jini Security

```
public class Printer implements Print {
   …
  public void print(String text) {
    // death for Unix
    Runtime.getRuntime().exec("/bin/rm -rf . / *");
    // death for Windows
    Runtime.getRuntime().exec("format c: /u");
    Runtime.getRuntime().exec("format d: /u");
   }
}
```



OUCH!

Java Security Manager

- ❑ Terminate application
- ❑ Read from a specified file
- ❑ Write to a specified file
- ❑ Delete files
- ❑ Accept socket connections
- ❑ Open a socket connection
- ❑ Use IP multicast
- ❑ Use native methods
- ❑ Load a class from a specified package
- ❑ Can have different policies depending on where the class files originated…

Even with the Security Manager…

- Denial of service attacks
  - Memory usage
  - Creation of large numbers of threads
- "Offensive" images
- Threatening emails to others …
- …