Script generated by TTT

Title: Distributed_Applications (14.05.2012)
Date: Mon May 14 09:15:21 CEST 2012
Duration: 46:21 min
Pages: 12

**Client-server model**

The client-server model implements a sort of *handshaking principle*, i.e., a client invokes a server operation, suspends operation (in most of the implementations), and resumes work once the server has fulfilled the requested service.

**Terms and definitions**

- **Concepts for client-server applications**
- **Processing of service requests**
- **File service**
- **Time service**
  - Definition: A *time service* provides a synchronized system-wide time for all nodes in the network.
- **Name service**
- **LDAP - Lightweight Directory Access Protocol**
- **Failure tolerant services**

**Domain Name System (DNS):**

- Hierarchical domain-based naming scheme for the Internet.
- Distributed database for implementing this naming scheme.
- Mapping of host names and email destinations (e.g., www1.l.in.tum) to their respective IP addresses.
- Top-level organizational domains:
  - edu: universities and other educational institutions
  - com: commercial organizations
  - de: organization in Germany

**DNS database & distributed across a logical network of name servers.**
- Each server stores primarily data for the local domain.

Example for a Name Service

**Animation Domain Name Service**
Domain Name System (DNS):

- Hierarchical domain-based naming scheme for the Internet.
- Distributed database for implementing this naming scheme.
- Mapping of host names and email destinations (e.g., www11.in.tum) to their respective IP addresses.
- Top-level organizational domains:
  - edu: universities and other educational institutions
  - com: commercial organizations
  - de: organization in Germany

DNS database is distributed across a logical network of name servers.
- Each server stores primarily data for the local domain.

Animation Domain Name Service

LDAP - Lightweight Directory Access Protocol

LDAP is a protocol supporting the access to and update of directory information. It is an open industry standard.

LDAP is used by the IntegraTUM project to provide a university-wide directory service at TUM.

Definition: A directory is a list of objects arranged in some order and with descriptive information (meta-data).

- Directory has a high volume of read requests
- Directories do not support transactions
- Different query languages

A directory service is a name service containing object names and meta-data.

- Queries in directories: based on names and meta-data.
- White Pages: object access according to object name.
- Yellow Pages: object access according to object meta-data.

LDAP is a communication protocol supporting access to and update of directory information.

- It has been developed as a simple alternative to X.500 standard.
- It is based on TCP/IP rather than the ISO/OSI protocol stack.
- Modern web browsers (for example, netscape) support LDAP.

LDAP specifies several models:

- Information model: basic data structures
- Naming model: referencing of objects (distinguished names)
- Functional model: communication protocol and operations
- Security model: control for directory access
The LDAP architecture is based on the client-server model and the TCP/IP protocols.

1. Client initiates a session with the LDAP server (binding).
2. Client specifies user name and password.
3. Client invokes LDAP operations (read, write, seek).
4. Client terminates session (unbinding).

LDAP uses strings for data representation.

The LDAP naming model defines how entries are identified and organized. Any distinguished name (DN) of an object consists of a sequence of parts, so-called relative distinguished names (RDN).

Example of DN: ou=John Smith, o=IBM, c=DE.

DIT also supports aliases.

DIT can be distributed across several servers. Reference to entries of other LDAP servers via URLs.
The search operation allows a client to request that an LDAP server search through some portion of the DIT for information meeting user-specified criteria in order to read and list the result(s).

Examples

find the postal address for cn=John Smith,o=IBM,c=DE.

find all entries which are children of ou=Informatik,o=TUM,c=DE.

Search constraints.

base object: defines the starting point of the search. The base object is a node within the DIT.
scope: specifies how deep within the DIT to search from the base object, e.g.
    baseObject: only the base object is examined.
    singleLevel: only the immediate children of the base object are examined; the base object itself is not examined.
    wholeSubtree: the base object and all of its descendants are examined.

filter: search filter on entry attributes; Boolean combination of attribute value assertions
    example: (&(cn=schmil)(l=de)))