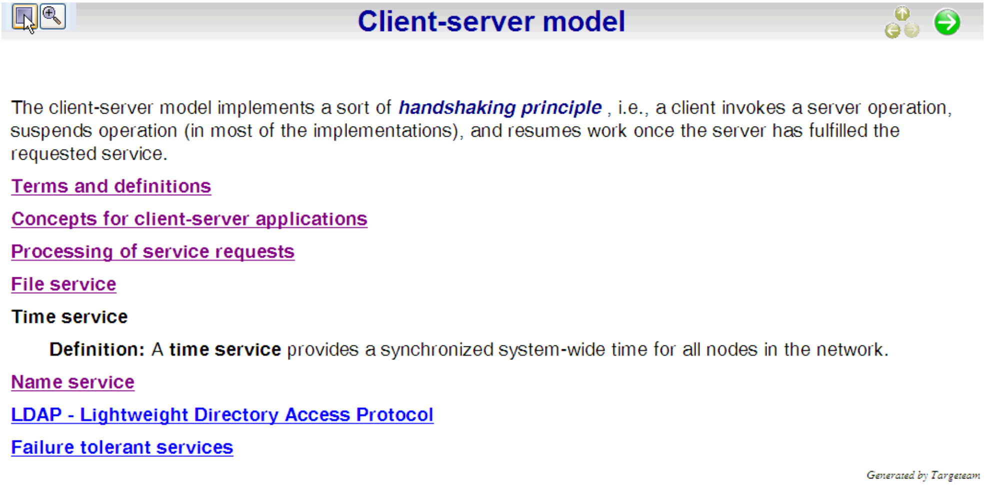
Script **generated by TTT**

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

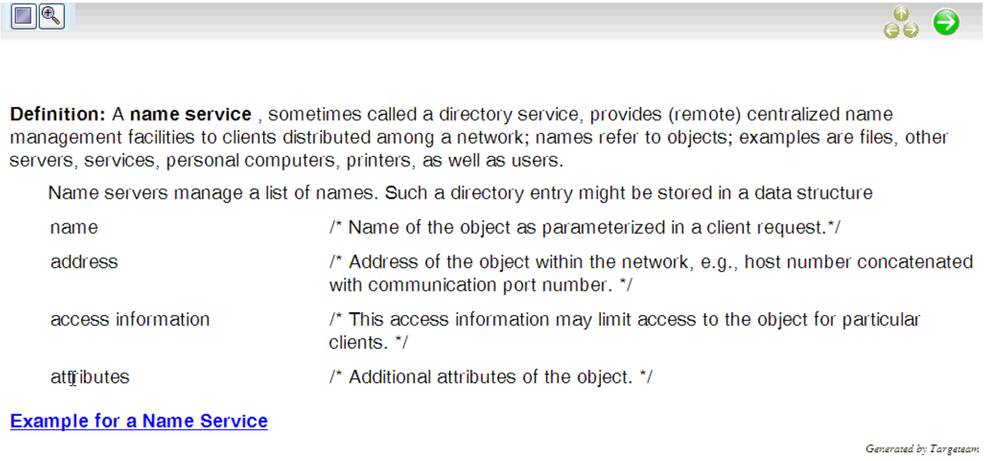
Definition: A **time service** provides a synchronized system-wide time for all nodes in the network.

Name service

LDAP - Lightweight Directory Access Protocol

Failure tolerant services

Generated by Targeteam



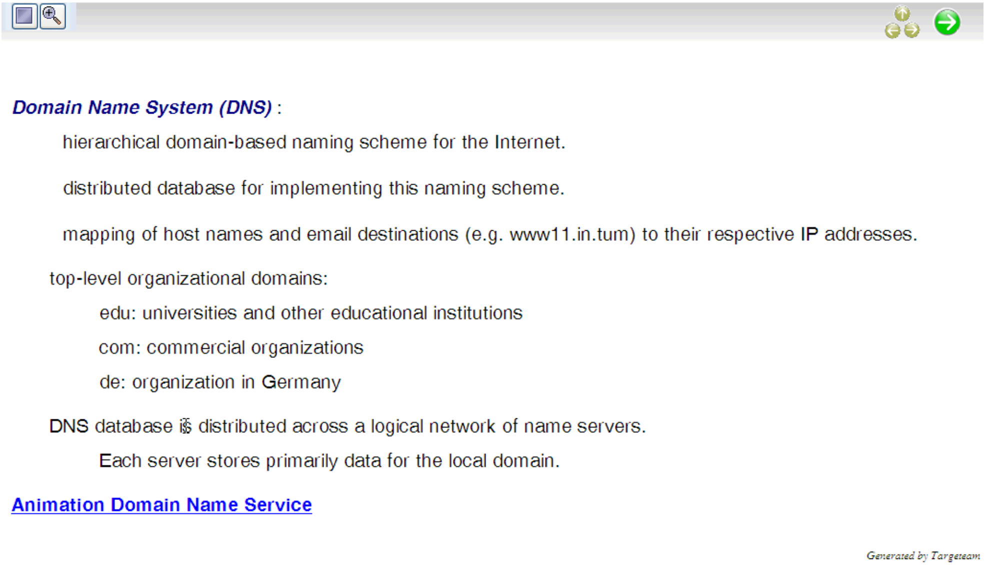
Definition: A **name service** , sometimes called a directory service, provides (remote) centralized name management facilities to clients distributed among a network; names refer to objects; examples are files, other servers, services, personal computers, printers, as well as users.

Name servers manage a list of names. Such a directory entry might be stored in a data structure

name	/* Name of the object as parameterized in a client request.*/
address	/* Address of the object within the network, e.g., host number concatenated with communication port number. */
access information	/* This access information may limit access to the object for particular clients. */
attributes	/* Additional attributes of the object. */

Example for a Name Service

Generated by Targeteam



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:

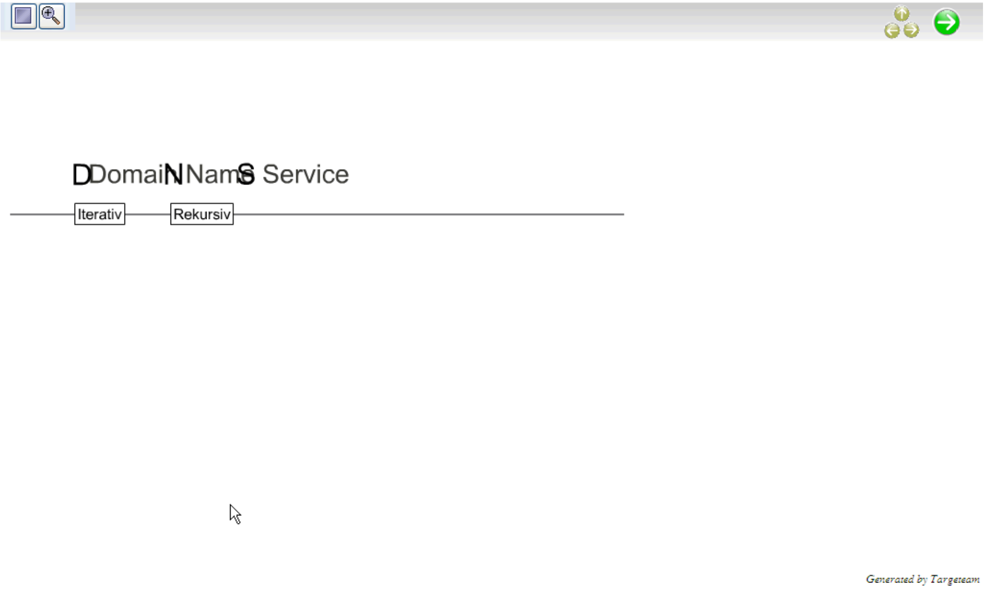
- edu: universities and other educational institutions
- com: commercial organizations
- de: organization in Germany

DNS database is distributed across a logical network of name servers.

Each server stores primarily data for the local domain.

Animation Domain Name Service

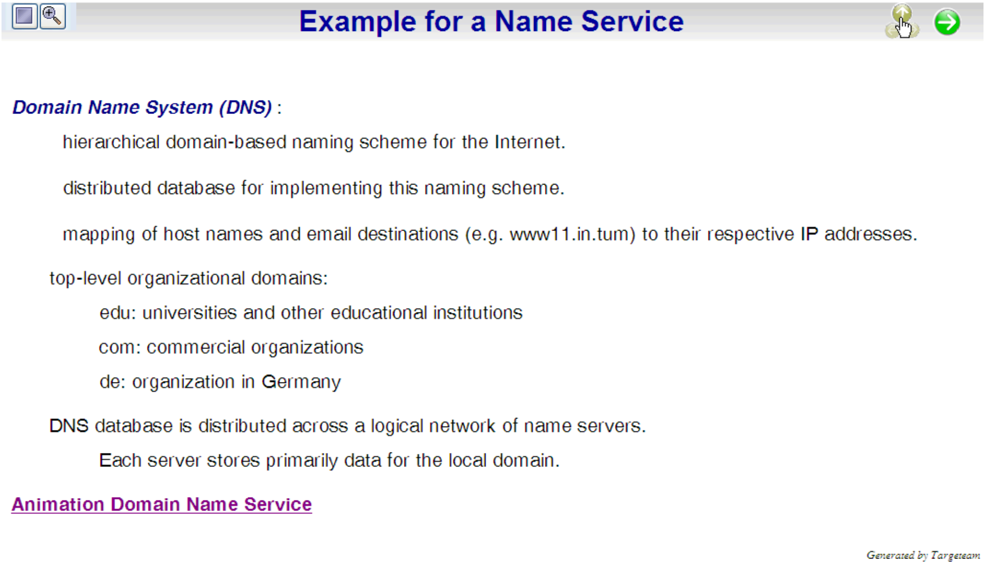
Generated by Targeteam



DDomaiNNamS Service

Iterativ Rekursiv

Generated by Targeteam



Example for a 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:

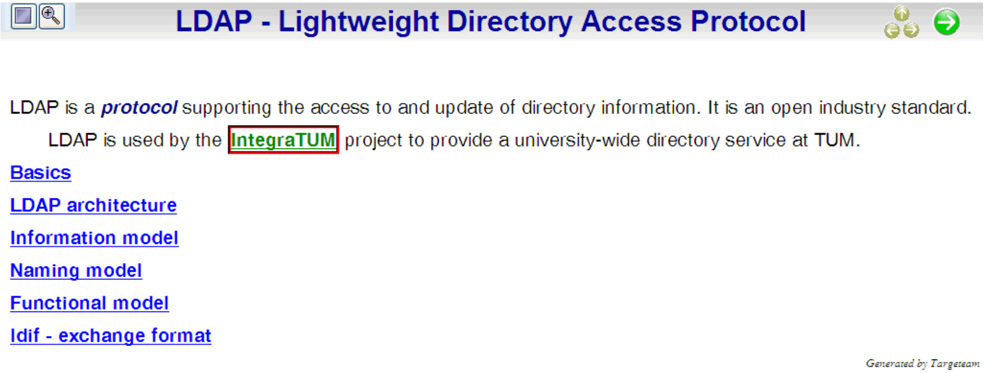
- edu: universities and other educational institutions
- com: commercial organizations
- de: organization in Germany

DNS database is distributed across a logical network of name servers.

- Each server stores primarily data for the local domain.

Animation Domain Name Service

Generated by Targeteam



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.

Basics

LDAP architecture

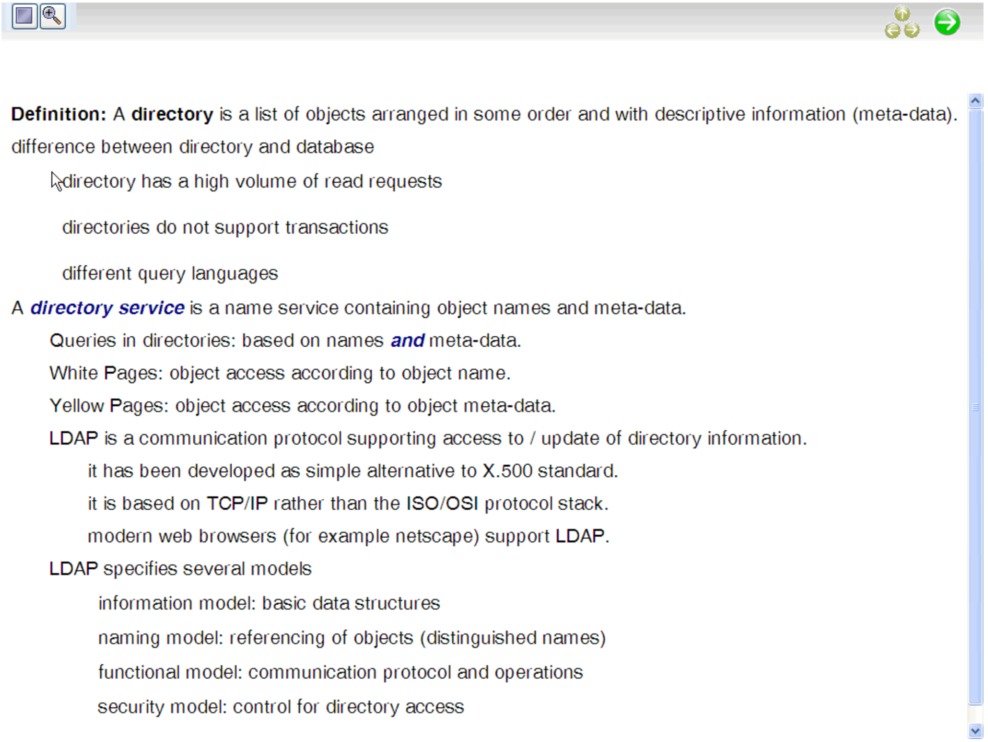
Information model

Naming model

Functional model

ldif - exchange format

Generated by Targeteam



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

difference between directory and database

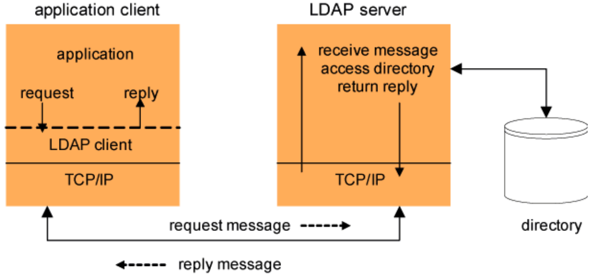
- directory has a high volume of read requests
- directories do not support transactions
- different query languages

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 / update of directory information.
  - it has been developed as 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

LDAP architecture

The LDAP architecture is based on the client-server model and the TCP/IP protocols.



LDAP uses strings for data representation.

General interaction process

1. Client initiates a session with the LDAP server (binding).

   Client specifies a name or an IP address and port (e.g. port 389) of the LDAP server.

   Client specifies user name and password.

2. Client invokes LDAP operations (read, write, seek).
3. Client terminates session (unbinding).

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.

Basics

LDAP architecture

Information model

Naming model

Functional model

ldif - exchange format

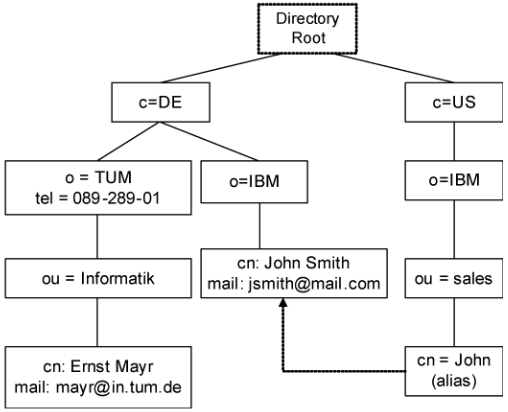
The LDAP naming model defines how entries are identified and organized. Any ***distinguished name*** (DN) of an object consists of a sequence of parts, so-called ***relative distinguished names*** (RDN).

The entries in an LDAP directory are hierarchically structured as tree (Directory Information Tree, DIT).

Example of DN: cn=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.



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.

Basics

LDAP architecture

Information model

Naming model

Functional model

ldif - exchange format

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=schmi*)(!(c=de)))

Code example