11 Verzeichnisdienste

11.1 Architektur des Domain Name Service (DNS) im Internet

11.2 Protokolle des DNS

11.3 Verzeichnisdienste nach ISO/OSI

11.1 Architektur des Domain Name Service (DNS) im Internet

Abbildung Namen ⇒ Internet-Adressen (IP-Adressen)

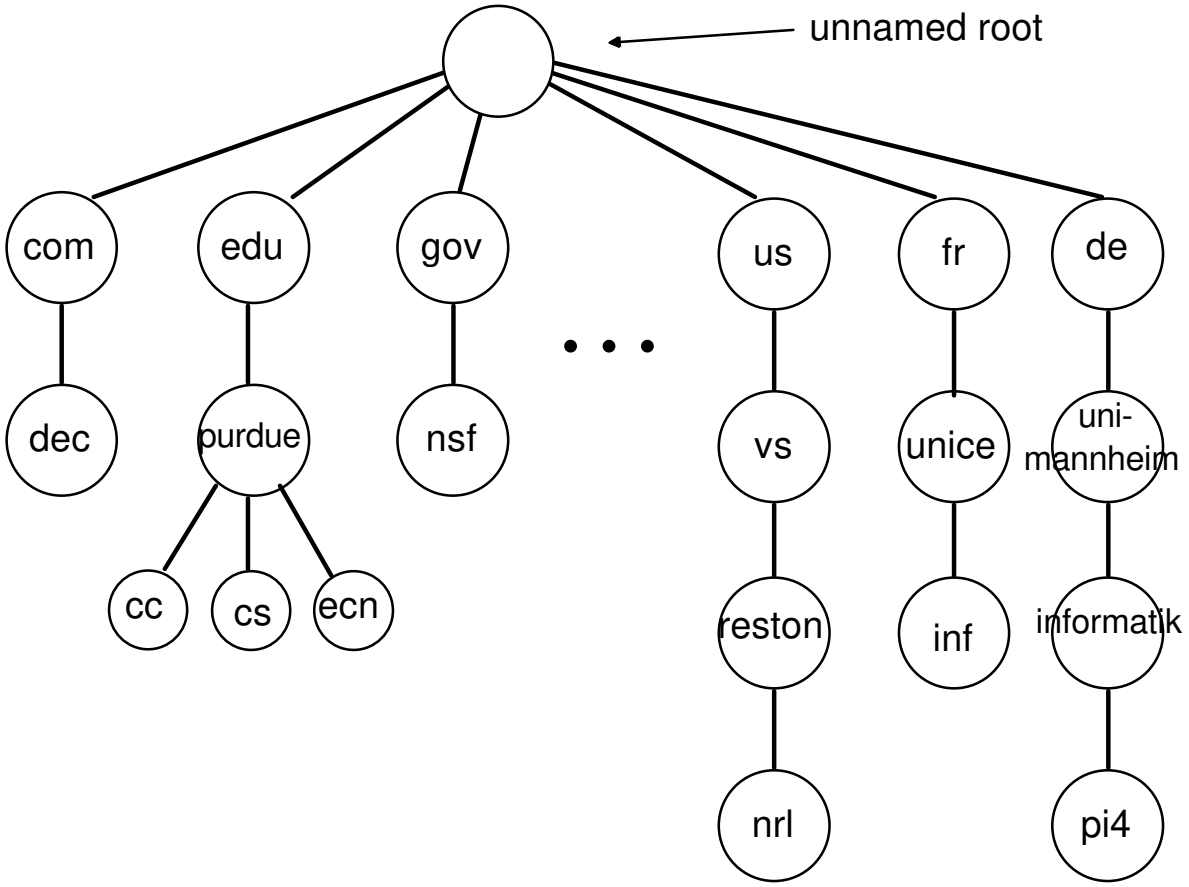
Lokale Rechnernamen befinden sich in /etc/hosts oder werden mittels Yellow-Pages abgefragt.

Die Namen im INTERNET sind hierarchisch strukturiert, z.B.:

de
uni-mannheim.de
uni-karlsruhe.de
informatik.uni-mannheim.de
pi4.informatik.uni-mannheim.de
pi3.informatik.uni-mannheim.de

Jede Hierarchiestufe entspricht einer **Domäne** von Adressen. Für jede Domäne existiert ein Name-Server, der die Hosts seiner Domäne kennt.

Beispiel für Domänen



Für jede Domäne gibt es eine administrative Stelle für die Namensvergabe.

Domain Name Service

- Menge von kooperierenden Name-Servern
- In der Regel ein Name-Server pro Domäne plus ein Back-Up-System

Namensauflösung

Prinzipiell

Top-Down, beginnend mit dem Root Server

Probleme

hoher Overhead, Root wird zum Flaschenhals

Daher

2-stufiger Auflösungsmechanismus:

- Klient kontaktiert einen lokalen Name-Server
- Wenn keine lokale Namensauflösung möglich ist, Durchlaufen der gesamten Hierarchie

Überwiegend werden lokale Namen benötigt ⇒ Effizienzgewinn. Effizienzsteigerung für nicht-lokale Namen: Caching im lokalen Name-Server

Algorithmus zur Namensauflösung (1)

- Die DNS-Client-Software heißt "name resolver"
- Der name resolver kennt die Adresse von mindestens einem name server (= Knoten im Baum des DNS). Dies ist der lokale name server, meist in Blatt-Knoten im DNS-Baum.
- Der name resolver baut eine Request-PDU auf ("domain name query") und sendet sie an den name server. Dabei verlangt er entweder "recursive resolution" oder "non-recursive resolution".
- Der name server prüft, ob er die domain name query lokal beantworten kann.
  - Falls ja, sendet er die Antwort an den Client
  - Falls nein und "recursive resolution" verlangt ist, kontaktiert er einen oder mehrere weitere name server im Baum, bis er die Antwort hat. Jeder name server muß mindestens einen root server kennen (mit IP-Adresse und DNS port).
  - Falls nein und "non-recursive resolution" verlangt, meldet er dem Client den Namen eines anderen name servers, den er versuchen könnte

Algorithmus zur Namensauflösung (2)

- Jeder name server hat einen Cache für Einträge, die von einem anderen name server geholt wurden. Die Cache-Einträge werden mit einem Time-Out versehen ("Time-to-Live"). Wird eine gesendete Information im Cache gefunden, so erhält der name resolver (DNS Client) diese Information zusammen mit der Adresse des zuständigen name servers im Baum.
- Manche name resolver haben eigene Cache-Speicher

11.2 Protokolle des DNS

0	16	31
IDENTIFICATION	PARAMETER	
NUMBER OF QUESTIONS	NUMBER OF ANSWERS	
NUMBER OF AUTHORITY	NUMBER OF ADDITIONAL	
QUESTION SECTION . . .		
ANSWER SECTION . . .		
AUTHORITY SECTION . . .		
ADDITIONAL INFORMATION SECTION . . .		

0	16	31
QUERY DOMAIN NAME . . .		
QUERY TYPE	QUERY CLASS	

0	16	31
RESOURCE DOMAIN NAME . . .		
TYPE	**CLASS**	
TIME TO LIVE	**RESOURCE DATA LENGTH**	
RESOURCE DATA . . .		

11.3 Verzeichnisdienste nach ISO/OSI

ISO/CCITT Verzeichnis Standards

- 9594/1 The Directory - Overview of Concepts, Models and Services (=X.500)
- 9594/2 The Directory - Models (=X.501)
- 9594/3 The Directory - Access and System Services Definition (=X.521)
- 9594/4 The Directory - Distributed Operations (=X.518)
- 9594/5 The Directory - Protocol Specification (=X.519)
- 9594/6 The Directory - Selected Attribute Types (=X.520)
- 9594/7 The Directory Selected Object Classes (=X.521)
- 9594/8 The Directory Authentication Framework (=X.509)

Zweck des Verzeichnisses

Informationen über Objekte der realen Welt (Telekommunikations- und Informationsverarbeitungsobjekte) können gespeichert und abgefragt werden, z.B.: Personen, Organisationen, Länder, Orte, Computersysteme, Anwendungsinstanzen

Die Funktionen des Verzeichnisdienstes werden benötigt von

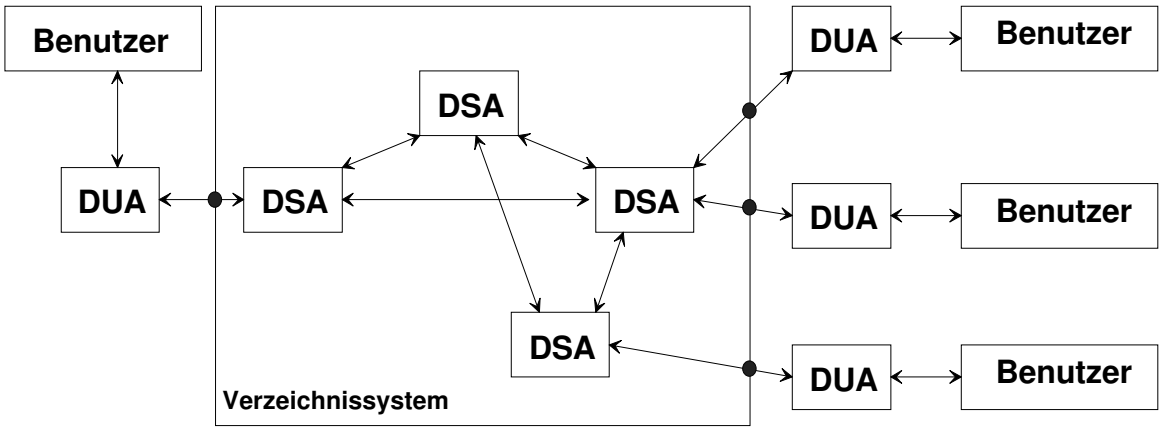
- Personen
- OSI Anwendungen
- OSI Managementprozessen
- OSI Schichtinstanzen
- Telekommunikationsdiensten (X.400 MHS, Teletex ...)

Eigenschaften eines Verzeichnisdienstes

- Globales (weltweit) verteiltes Verzeichnis mit replizierten Daten
- Abfragerate > > Aktualisierungsrate
- kein Bedarf für ein augenblickliches globales Commitment der Aktualisierungen; übergangsweise sind Inkonsistenzen sind akzeptierbar
- Benutzer (Personen, Computerprogramme) können die Informationen des Directories lesen und ändern
- benutzerfreundliche Namensgebung, besonders für Personen, Benutzung von Alias-Namen, 'yellow-page'-Dienst (gelbe Seiten)
- isoliert den Benutzer von Änderungen der Abbildung von Namen auf Adressen

11.3.1. Funktionales Modell

Das Verzeichnis und seine Benutzer



- ● **Zugriffspunkt**

↔ **Interaktion**

DUA: Directory User Agent

DSA: Directory System Agent

Direcory User Agent (DUA)

- Anwendungsprozeß
- arbeitet im direkten Dialog mit dem Verzeichnis auf Anweisung eines Benutzers (Person oder Computer-programm)

Directory System Agent (DSA) Anwendungsprozeß

- Teil des Directories, hält Teile der Informationen des Directories bereit und stellt keinen, einen oder mehrere Zugriffspunkte für DUAs zur Verfügung
- Die DSAs sind vernetzt und implementieren ein verteiltes Verzeichnis

11.3.2. Organisatorisches Modell

Directory Management Domain (DMD)

- Menge von einem oder mehreren DSAs und einem oder mehreren DUAs, verwaltet durch eine einzelne Organisation
- Administration DMD (ADDMD)
- Private DMD (PRDMD)

11.3.3. Informationsmodell

Beschreibt die logische Struktur der Informationen, die vom Verzeichnisdienst verwaltet werden.

Directory Information Base (DIB)
Menge von Informationen (Verzeichnis-Einträge), die durch das Verzeichnis verwaltetet werden

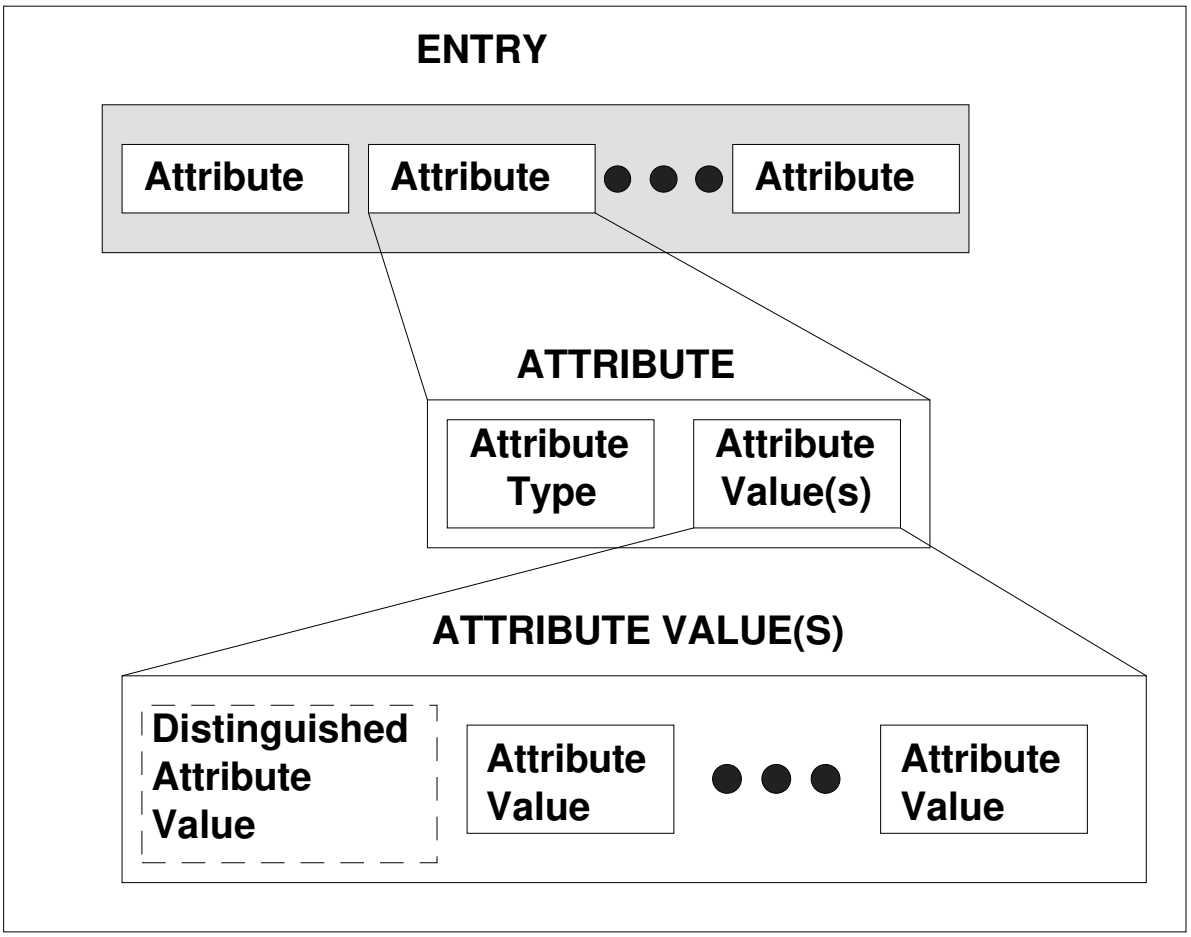
Verzeichnis-Eintrag:
Sammlung von Informationen über ein einziges Objekt

Objektklasse

- identifizierte Familie von Objekten, die bestimmte Eigenschaften gemeinsam haben
- jedes Objekt gehört zu mindestens einer Objektklasse
- für jedes einzelne Objekt existiert genau ein Objekteintrag in der DIB und zusätzlich keiner oder mehrere Alias-Einträge

Struktur eines Eintrags

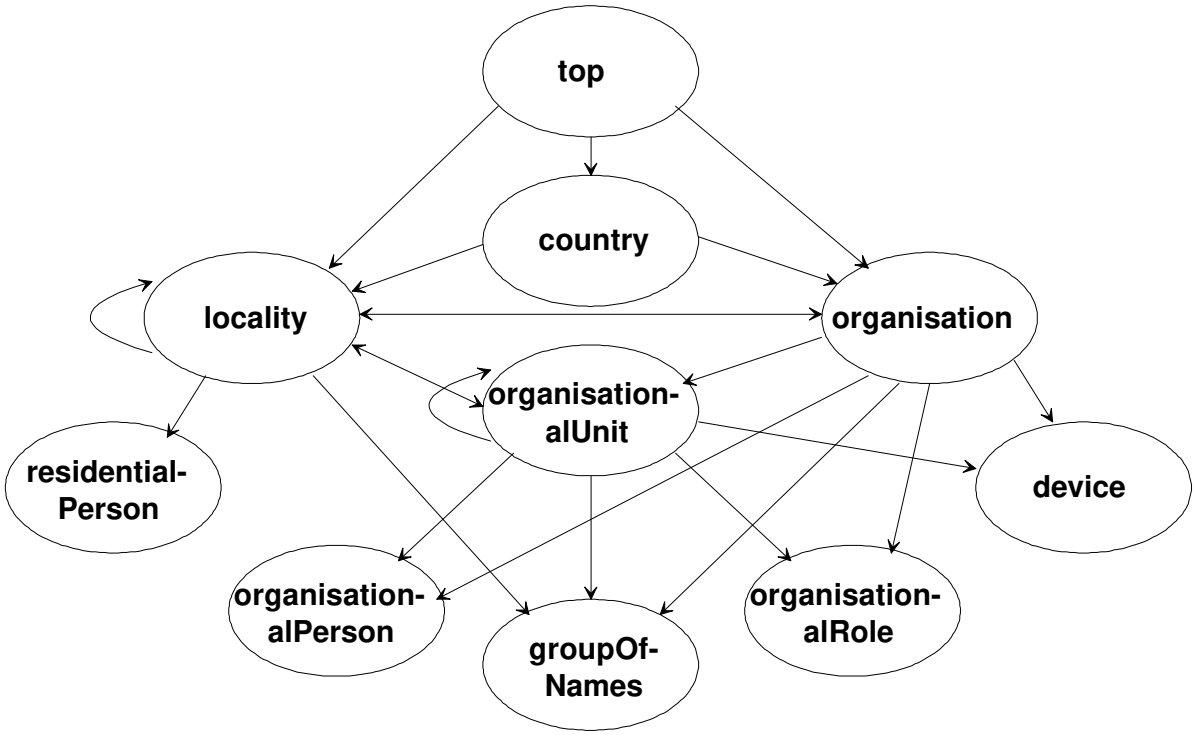
Menge von Attributen, jeder mit einem Attributtyp und einem oder mehreren Werten



Directory Information Tree (DIT)

- Die Verzeichnis-Einträge in einer Baumstruktur angeordnet
- Diese basiert auf den hierarchischen Beziehungen, die zwischen Realwelt-Objekten gefunden werden
- Unterstützt das systematische Finden von Einträgen und das verteilte Management der DIB

Beispiel:



Namensgebung

Unterscheidbarer Name:
Jeder Eintrag in der DIB hat einen unterscheidbaren Namen, welcher eindeutig und unzweifelhaft den Eintrag identifiziert. Die Eigenschaft des unterscheidbaren Namens leitet sich aus der Baumstruktur der Information ab.

Relativ unterscheidbarer Name (RDN):
Dieser Name ist eindeutig nur relativ zu dem Vorgänger des Eintrags. Er besteht aus einer Teilfolge von Attributwerten, die die unterscheidbaren Werte des Eintrags enthält.

Beispiele für Attributtypen

Common Name

Serial Number

Country Name

Locality Name

Postal Address

Postal Code

Organization Name

Organization Unit Name

Telephone Number

ISDN-Address

Presentation Address

Telephon Number

11.3.4 Verzeichnis-Dienst

Der Verzeichnisdienst stellt die folgenden Operationen für DUAs zur Verfügung:

Operationen zum Zuordnen und Aufheben der Zuordnung

- Bind
- Unbind

Leseoperationen

- Read
- Compare
- Abandon

Suchoperationen

- List
- Search

Änderungsoperationen

- Add Entry
- Remove Entry
- Modify Entry

Verteilte Operation

Die DSAs benutzen die folgenden Operationen für ihre Zusammenarbeit:

Lesedienste

- Chained Read
- Chained Compare

Suchdienste

- Chained List
- Chained Search

Änderungsdienste

- Chained Add Entry
- Chained Remove Entry
- Chained Modify Entry

Verzeichnis-Protokolle

Spezifikation der Anwendungsdienstelemente und des Anwendungskontextes für zwei Protokolle, zur Kooperation zwischen DUAs und DSAs in Schicht 7:

- Directory Access Protocol (DAP)
- Directory System Protocol (DSP)

Authentifikation

Bereitstellung von Authentifikationsdiensten durch das Verzeichnis für seine Benutzer.

Zwei Ebenen der Authentifikation

- Einfache Authentifikation
  Beruht auf der Benutzung von unterscheidbaren Namen und Passwort eines Benutzers.

- Strenge Authentifikation
  Basiert auf starken Verschlüsselungsalgorithmen (z.B. Public-Key-Kryptographie (RSA)