Datenbankadministration

4. Zugriffskontrolle

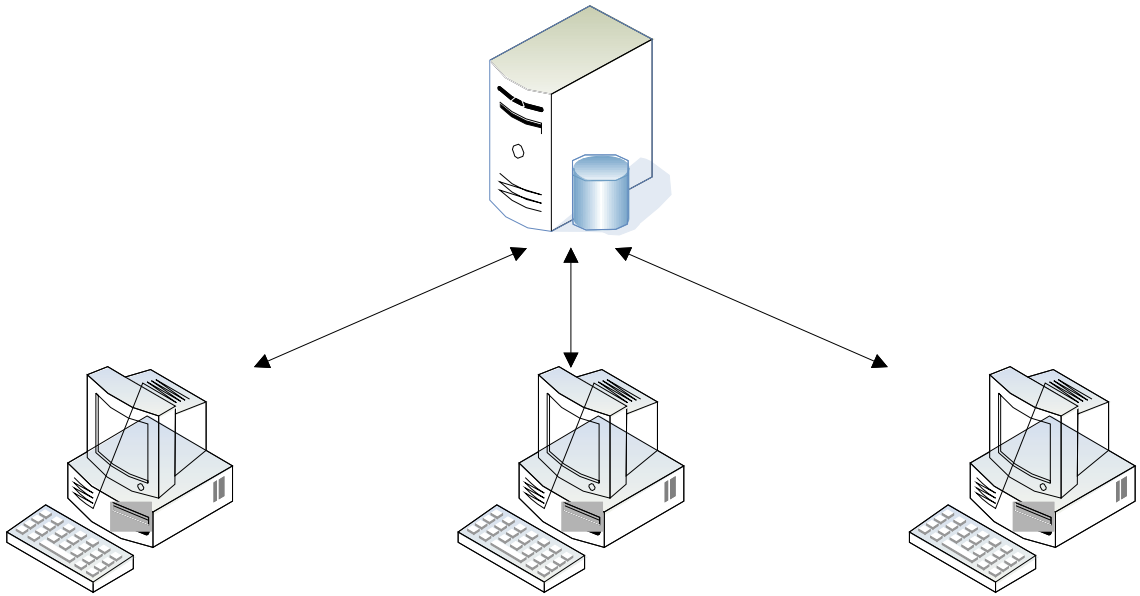
AG DBIS University of Kaiserslautern, Germany

Karsten Schmidt kschmidt@informatik.uni-kl.de
(Vorlage TU-Dresden)

Wintersemester 2008/2009

- **Architektur**

Server mit z.B. DB2 Enterprise Server Edition



Clients, z.B. DB2 Client
oder DB2 Enterprise Server Edition

Katalogisieren

- **Katalogisieren einer Datenbank**
  - Verbindungsaufbau erfordert Katalogisieren
    - Verwaltungsserver
      `CATALOG ADMIN TCPIP NODE <nodename>`
      `REMOTE <host-name> | <ip-address>`
    - Instanz
      `CATALOG TCPIP NODE <nodename>`
      `REMOTE <host-name> | <ip-address>`
      `SERVER <service-name> | <port>`
      `REMOTE_INSTANZ <instance-name>`
    - Datenbank
      `CATALOG DATABASE <dbname> AS <alias>`
      `AT NODE <nodename>`
  - Objekte aus dem Katalog entfernen
    - `UNCATALOG NODE <nodename>`
    - `UNCATALOG DATABASE <dbname>`
  - Aktualisieren des Verzeichniscache
    - `TERMINATE`

Übersicht

- **Was wird heute behandelt - Sicherheitsaspekte**
  - Authentifikation
  - Autorisierung
  - Privilegien
  - Nutzer und Gruppen
- **Aspekte eines Sicherheitskonzepts**
  - Wer darf auf eine Instanz bzw. Datenbank zugreifen?
  - Wo und wie wird ein Nutzer verifiziert?
  - Welche Kommandos darf ein Nutzer absetzen?
  - Welche Daten darf ein Nutzer lesen und ändern?
  - Welche Datenbankobjekte darf ein Nutzer anlegen, ändern und löschen?
  - Welche Rechte darf ein Nutzer weitergeben?

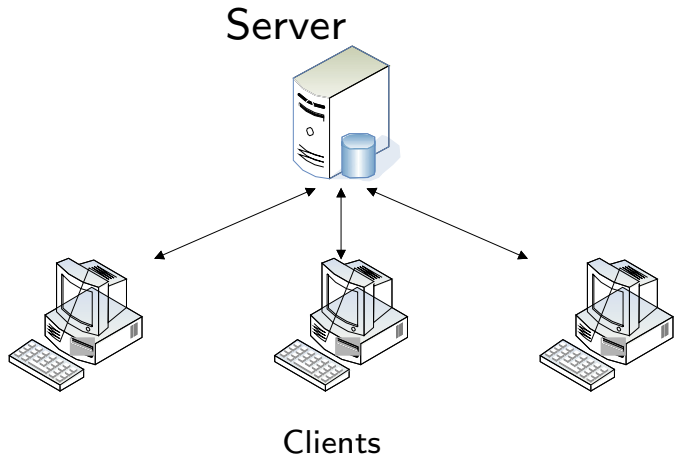
DB2 Sicherheitskonzept

- **DB2 nutzt eine Kombination aus**
  - externen Sicherheitsservices
  - internen Zugriffskontrollinformationen
- **Authentifikation**
  - Identifizierung des Nutzers
  - Kontrolle Nutzername und Passwort
  - Verbindung zur Datenbank

    ```
    CONNECT TO <dbname>
       USER <user> USING <password>
    ```
- **Autorisierung**
  - Was darf der Nutzer und was nicht

Authentifikation

- **Authentifikation (authentication)**
  - Verifizierung der Identität eines Nutzers (Nutzername/Passowrt)
  - durch Betriebssystem oder separates Produkt (Kerberos)
  - durch Server oder Client
- **Art der Authentifikation (authentication type)**
  - werden am Server und am Client gesetzt (müssen übereinstimmmen)



- `UPDATE DBM CFG USING AUTHENTICATION <auth_type>`
- `CATALOG DB <dbname> AT NODE <node_name AUTHENTICATION <auth_type>`

Authentifikation

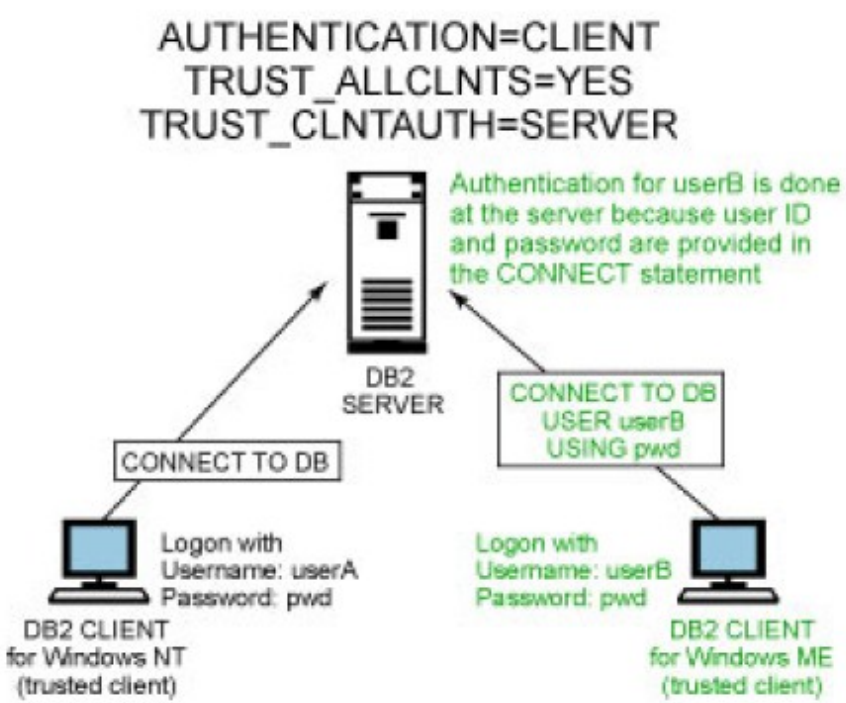
- `<auth_type>`
  - `SERVER / SERVER_ENCRYPT`
    = Authentifikation am Server
  - `CLIENT`
    = Authentifikation am Client
  - `KERBEROS`
  - `KRB_SERVER_ENCRYPT`
    = Kerberos & Server Encrypt; nur für Server
  - `DCE`
    = Authentifikation durch Fremdsoftware; nur für Client

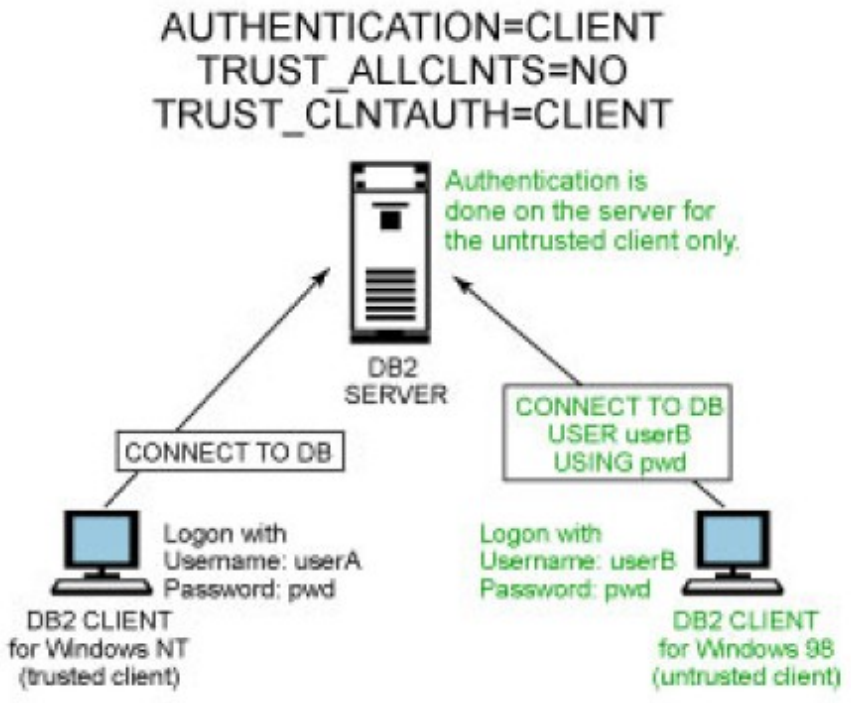
Client-Authentifikation

- **Client-Authentifikation**
  - `TRUST_ALLCLNTS`
    - Vertrauenswürdigkeit des Client-Betriebssystems (!)
    - `UPDATE DBM CFG USING TRUST_ALLCLNTS YES|NO|DRDAONLY`
      - `YES` → allen Clients vertrauen (unabhängig vom Betriebssystem)
      - `NO` → nicht-vertrauenswürdige Clients am Server authentifizieren (z.B. Windows 98 und früher)
      - `DRDAONLY` → Distributed Relational Database Architecture-fähige Hostrechner
  - `TRUST_CLNTAUTH`
    - Festlegung des Authentifikationsortes, falls vertrauenswürdige Clients trotzdem Logininformationen angeben
    - `UPDATE DBM CFG USING TRUST_CLNTAUTH CLIENT|SERVER`

Client-Authentifikation

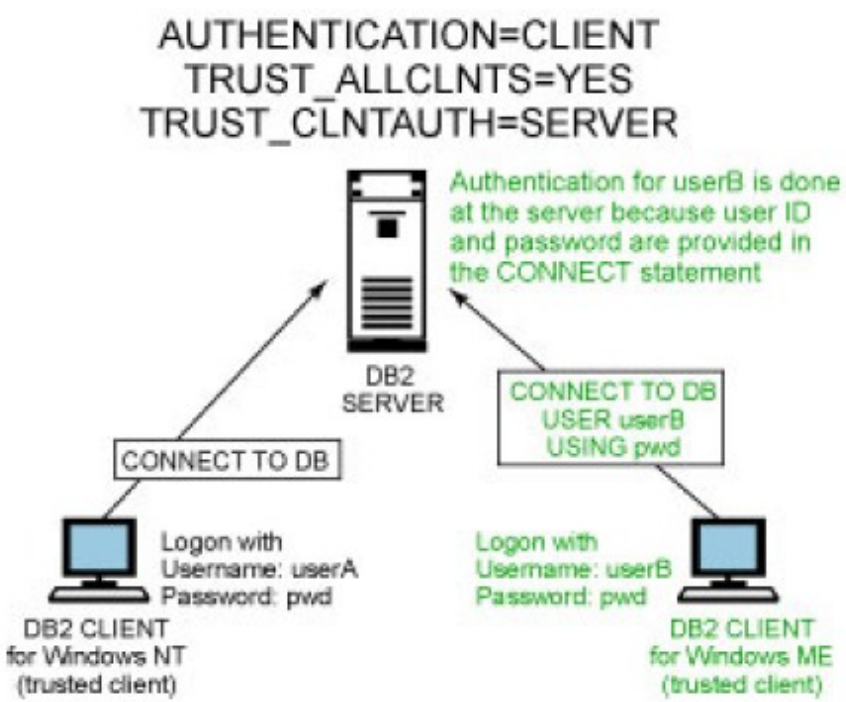
- **Beispiele**

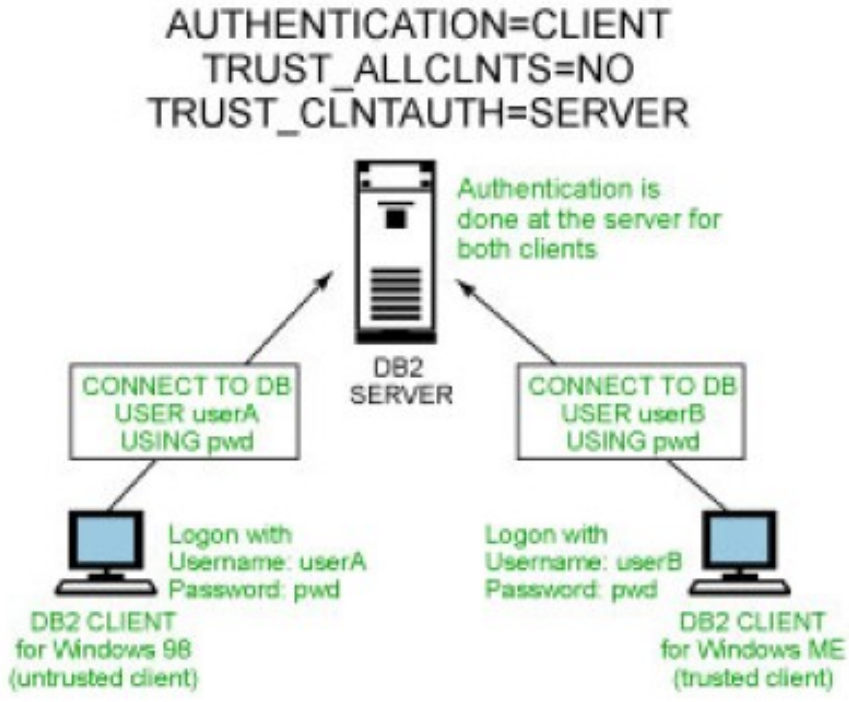




Client-Authentifikation

- **Beispiele**





Autorisierung

Autoritäten

- **Autorisierung (authorization)**
  - Zugriffs- bzw. Ausführungsrechte für Datenbankobjekte
  - Unter Kontrolle des DB2 Database Managers

Gruppe

- `SYSADM`: Systemadministrator
  - Ausführen sämtlicher Verwaltungsaufgaben der DB2 Instanz
- `SYSCTRL, SYSMAINT`: System Control, System Maintenance
  - Keine Berechtigung zum Modifizieren der Instanzkonfiguration
  - Komplette Kontrolle über alle Datenbankobjekte (keine Leserechte!)
  - Ausführen von Verwaltungsaufgaben (Backup & Recovery, Ändern der Datenbankkonfiguration, etc.)

Gruppe User

- `DBADM`: Datenbankdministrator
  - Komplette Kontrolle über eine Datenbank
  - Wird dem Ersteller einer Datenbank automatisch zugewiesen
- `LOAD`
  - Berechtigung zum Laden von Daten
  - Zusätzlich Privilegien auf entsprechende Datenbankobjekte nötig (z.B. Relation laden nur mit `INSERT` Privileg)

Autoritäten (2) - Instanz-Level

- **SYSADM**
  - vergleichbar mit root (Unix) oder Administrator (Windows)
  - SYSADM-Nutzer dürfen als einzige Veränderungen an den Instanz-Einstellungen vornehmen (update dbm cfg)
- **SYSCRTL**
  - ähnlich zum SYSADM, wobei kein Zugriff auf die Daten in der Datenbank möglich ist
  - Achtung: db2 create/drop database/tablespace möglich
- **SYSMAINT**
  - Teilmenge von SYSCRTL
  - db2 create/drop database/tablespace nicht möglich
  - nur Maintenance möglich

Autoritäten

- **Zuweisung von Autoritäten**
  - `SYSADM, SYSCRTL, SYSMAINT`
    - Auf Instanzebene
    - `UPDATE DBM CFG USING SYSADM_GROUP|SYSCTRL_GROUP| SYSMAINT_GROUP <group_name>`
  - `DBADM, LOAD`
    - Auf Datenbankebene
    - `GRANT LOAD|DBADM ON DATABASE <db_name> TO USER|GROUP <auth_name>`
    - `GRANT INSERT ON TABLE <tab_name> TO USER|GROUP <auth_name>`

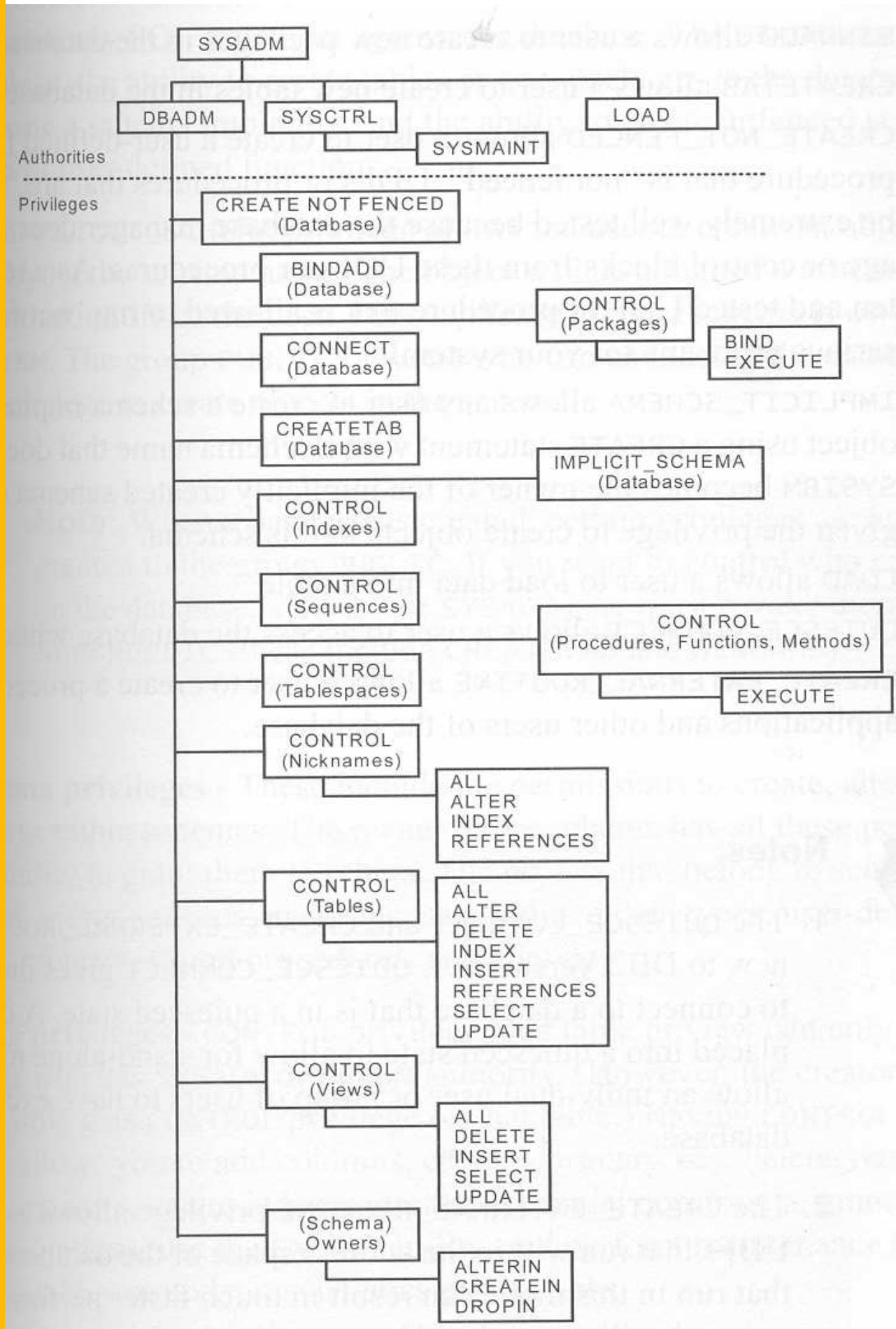
Privilegien

- **Privilegien (privileges)**
  - Recht zum Anlegen bzw. Zugriff auf Datenbankobjekte
  - Zuweisung
    - `GRANT` (DB2 Dokumentation)
    - `GRANT <privilege> ON <obj> <obj_name> TO USER|GROUP <auth_name>`
  - Entzug
    - `REVOKE` (DB2 Dokumentation)
    - `REVOKE <privilege> ON <obj> <obj_name> FROM USER|GROUP <auth_name>`
  - Eigentümer bzw. Erzeuger eines Datenbankobjektes erhält `CONTROL`-Privileg und kann Rechte weitergeben

Privilegien (2)

- **Datenbank-Level**
  - CREATETAB
    - Nutzer können Tabellen anlegen
  - CONNECT
    - Nutzer können sich zur Datenbank verbinden
  - LOAD
    - Nutzer können das LOAD-Kommando ausführen
- **Datenbankobjekt-Level**
  - CONTROL (voller Zugriff auf das Datenbankobjekt)
  - SELECT, INSERT, DELETE, UPDATE
  - ALTER, INDEX,
  - ...

Privilegien



Privilegien

- **Implizite Privilegien**
  - Bei Vergabe von Privilegien/Autoritäten werden automatisch **einige** niedere Prioritäten mit vergeben = implizite Privilegien
  - Bei Entzug des höherwertigen Privileges verbleiben die expliziten und impliziten Privilegien
  - Z.B. `CONTROL` auf Table impliziert alle untergeordneten Privilegien, bei Entfernen des `CONTROL`-Privileges bleiben diese aber erhalten
- **Grant-Option**
  - Bei `GRANT`-Befehl kann `WITH GRANT OPTION` angegeben werden
  - Das Recht darf weiter vergeben werden, aber NICHT wieder entzogen werden
  - Es gibt keine „`WITH REVOKE OPTION`“

Zusammenfassung

- **Katalogisierung von entfernten Datenbanken**
- **Sicherheitskonzept**
  - Authentifikation
  - Autoritäten
  - Privilegien
- **Mehr Information**
  - 730: Tutorial 2