8. Datenschutz und Zugriffskontrolle in DBS

- **Beobachtung**
  - Es werden immer mehr Daten gespeichert und zwischen Programmen ausgetauscht.
  - Neue Dimensionen beim Sammeln von Daten und dem daraus resultierenden Gefährdungspotential: WWW, Electronic Business, Data Warehouse, Data Mining, . . .
  - Wesentliche Schwachpunkte existierender Schutzkonzepte: mangelnde Differenzierbarkeit und Einheitlichkeit
  - Die Anzahl der Angreifer (Schnüffler, Hacker, Viren, . . .) nimmt zu!
- **BDSG - Übersicht**
  - Legislative Maßnahmen zum Datenschutz
  - Die 10 Gebote
- **Wer sind die Angreifer?**
- **Technische Maßnahmen des Datenschutzes**
  - Zugangs- und Benutzerkontrolle
  - Authentisierung, Übermittlungskontrolle
  - Speicherkontrolle, Zugriffskontrolle
- **Autorisierungsmodell mit expliziten Zugriffsrechten**
- **Zugriffskontrolle in SQL**
  - Definition von Zugriffsobjekten
  - Vergabe und Kontrolle von Zugriffsrechten
  - Probleme des Rechteentzugs
- **Verfeinerung des Autorisierungsmodells**
  - Implizite Autorisierung bei hierarchischer Anordnung von Subjekten, Objekten, Operationen und bei Typhierarchien
  - Rollenkonzept in SQL
- **Sicherheitsprobleme in statistischen DBs**

Datenschutz

- ***Legislative Maßnahmen (Datenschutzgesetze)[1]***
  - **Festlegung, welche Daten in welchem Umfang schutzbedürftig sind**
  - **Vorschriften, die Mißbrauch der Daten entgegenwirken**

    (Festlegung, welche Daten von wem gespeichert werden dürfen, welcher Zugriff auf Daten erlaubt ist, welche Weitergabe der Daten zulässig ist usw.)

- ***BDSG will schutzwürdige Belange der Betroffenen schützen***

a) Allgemeines Verbot der Verarbeitung personenbezogener Daten mit Erlaubnis gewisser Ausnahmen
   ➥ ***Verbotsprinzip mit Erlaubnisvorbehalt***

b) Gewährung spezieller Rechte für die Betroffenen (Auskunft, Berichtigung, Sperrung, Löschung)

c) Einführung besonderer Maßnahmen technischer und organisatorischer Art

- ***Technische Maßnahmen***

1. **Zugangs- und Benutzerkontrolle, Authentifikation**
2. **Übermittlungskontrolle in Rechnernetzen**
3. **Speicherkontrolle: Isolation der Benutzer und Betriebsmittel**
4. **Zugriffskontrolle: Autorisierung des Zugriffs auf gemeinsame Daten**
5. **Datenflußkontrolle beim Datentransport**
6. **Inferenzkontrolle bei statistischen DB**

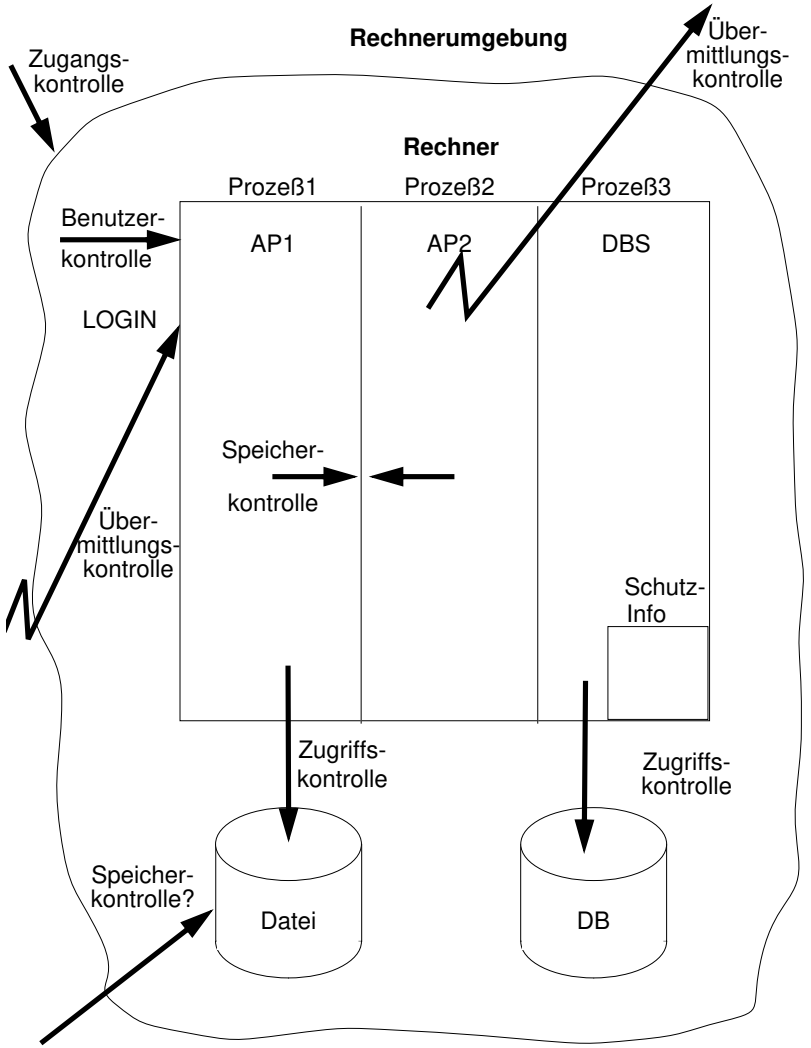
1. http://wwwdbis.informatik.uni-kl.de/misc/gesetze/

BDSG-Anlage (zu § 9 Satz 1)
Technische Maßnahmen

Werden personenbezogene Daten automatisiert verarbeitet, sind Maßnahmen zu treffen, die je nach der Art der zu schützenden personenbezogenen Daten geeignet sind,

1. Unbefugten den Zugang zu Datenverarbeitungsanlagen, mit denen personenbezogene Daten verarbeitet werden, zu verwehren (Zugangskontrolle),
2. zu verhindern, daß Datenträger unbefugt gelesen, kopiert, verändert oder entfernt werden können (Datenträgerkontrolle),
3. die unbefugte Eingabe in den Speicher sowie die unbefugte Kenntnisnahme, Veränderung oder Löschung gespeicherter personenbezogener Daten zu verhindern (Speicherkontrolle),
4. zu verhindern, daß Datenverarbeitungssysteme mit Hilfe von Einrichtungen zur Datenübertragung von Unbefugten genutzt werden können (Benutzerkontrolle),
5. zu gewährleisten, daß die zur Benutzung eines Datenverarbeitungssystems Berechtigten ausschließlich auf die ihrer Zugriffsberechtigung unterliegenden Daten zugreifen können (Zugriffskontrolle),
6. zu gewährleisten, daß überprüft und festgestellt werden kann, an welche Stellen personenbezogene Daten durch Einrichtungen zur Datenübertragung übermittelt werden können (Übermittlungskontrolle),
7. zu gewährleisten, daß nachträglich überprüft und festgestellt werden kann, welche personenbezogenen Daten zu welcher Zeit von wem in Datenverarbeitungssysteme eingegeben worden sind (Eingabekontrolle),
8. zu gewährleisten, daß personenbezogene Daten, die im Auftrag verarbeitet werden, nur entsprechend den Weisungen des Auftraggebers verarbeitet werden können (Auftragskontrolle),
9. zu verhindern, daß bei der Übertragung personenbezogener Datenträger die Daten unbefugt gelesen, kopiert, verändert oder gelöscht werden können (Transportkontrolle),
10. die innerbehördliche oder innerbetriebliche Organisation so zu gestalten, daß sie den besonderen Anforderungen des Datenschutzes gerecht wird (Organisationskontrolle)

Technische Maßnahmen des Datenschutzes



➥ Das „schwächste Glied“ in einer Kette von Maßnahmen bestimmt die Sicherheit des Gesamtsystems!

Wer sind die Angreifer?

- **Klassifikation:**
  - Mitarbeiter (allergrößte Gruppe der Angreifer)
  - Wannabees („Ich will Hacker werden“; sie finden es „cool“)
  - Script-Kidies
  - Hacker
  - Cracker (Zerstören Systeme!)
  - Gurus & Wizards (kriegen fast jede Kiste auf!)
  - Professionelle Industriespione
  - Geheimdienste

Der Chaos Computer Club

(1200 Mitglieder in Deutschland)

- **Ziele:**
  - Freiheit der Information
  - Recht auf Kommunikation
  - Informationelle Selbstbestimmung
  - Forum für kreative Techniknutzer („Hacker“)
  - kritische Analyse und Aufzeigung der Gefahren der Informationsgesellschaft
  - Cyber-Rights, Lobbyarbeit
  - Verbreitung der Hacker-Ethik
  - www.ccc.de

Die Hackerethik

- Der Zugang zu Computern und allem, was einem zeigen kann, wie diese Welt funktioniert, sollte unbegrenzt und vollständig sein!
- Alle Informationen müssen frei sein!
- Mißtraue Autoritäten - fördere Dezentralisierung!
- Beurteile einen Hacker nach dem, was er tut und nicht nach üblichen Kriterien wie Aussehen, Alter, Rasse, Geschlecht oder gesellschaftlicher Stellung!
- Man kann mit einem Computer Kunst und Schönheit schaffen!
- Computer können dein Leben zum Besseren verändern!
- *Mülle nicht in den Daten anderer Leute!*
- *Öffentliche Daten nützen, private Daten schützen!*

Information Warfare

- **Und International?**
  - Gemeinsame Erklärung von CCC, 2600, L0pht, Phrack, Cult of the Dead Cow, Pulhas, !Hispahack u.a.
  - Mehr als 120 Hackergruppen haben unterzeichnet
  - „We - the undersigned - strongly oppose any attempt to use the power of hacking **to threaten or destroy the information infrastructure of a country**, for any reason. Declaring „war“ against a country is the most irresponsible thing a hacker group could do. This has nothing to do with hacktivism or hacker ethics and is nothing a hacker could be proud of.“
  - „The signatories to this statement are asking hackers to reject all actions that seek to damage the information infrastructure of any country. **DO NOT support** any acts of „**Cyberwar**“. Keep the networks of communication alive. They are the nervous system for human progress“.
  - www.hackernews.com/archive.html?010799.html

Zugangs- und Benutzerkontrolle

- **Kernfrage 1:**
  Wie erkennt ein Rechensystem einen berechtigten Benutzer?

  ➥ Frage nach der Identifikation/Authentisierung

- **Organisatorische Maßnahmen**

  (Zugangskontrolle, bauliche Maßnahmen, . . .)

- **Identitätskontrolle (Authentisierung)**

  Nachweis der Identität des Benutzers gegenüber Transaktionssystem bzw. gegenüber BS und DBS

  ➥ Verfahrensklassen bei Authentisierung

- **Was man ist, was man hat, was man weiß**
  - Benutzercharakteristika werden überprüft (Stimme, Handgeometrie, Fingerabdruck, Unterschrift, . . ., „der Körper als Ausweis“).
  - Ausgehändigte Gegenstände ermöglichen Zugriff (Schlüssel für Terminal, maschinell lesbare Ausweise).
  - Authentisierung mittels Wissen (Paßwortmethoden, Frage-Antwort-Methoden, Challenge-Response-Verfahren)

Authentisierung

- **Kernfrage 2:**
  Wie kann ich mich gegenüber einen anderen zweifelsfrei ausweisen?
  Wie kann ich sicher sein, daß eine Nachricht wirklich von dem anderen Sender stammt?

  ➥ Frage nach der Authentisierung von Systemen/Dokumenten

- **Authentisierung**
  - Nachweis der Identität des Benutzers (Netzknoten, Dokument)
  - Authentisierung bezieht sich auf die Quelle der Information (Senderauthentisierung) und auf ihren Inhalt (Datenintegrität).

- **Beidseitige Authentisierung für Rechner-Rechner-Kommunikation**
  - Challenge-Response-Verfahren
  - Einigung auf kryptographisches Verfahren, Schlüsselaustausch
  - Authorization-Encryption: Chiffrierschlüssel mit Gültigkeit für die gesamte Sitzung (*Session*) dient der Autorisierung (ohne bei jedem Nachrichtenaustausch die beidseitige Authentisierung wiederholen zu müssen).

- **Nachrichtenauthentifikation**
  - Unterschriften, Echtheitsmerkmale
  - Der Ersteller besitzt etwas, mit dessen Hilfe er das Dokument authentisch macht.

Übermittlungskontrolle

- **Kernfrage 3:**
  Wie kann ich mit jemand vertraulich kommunizieren?

  ➥ Frage nach der Geheimhaltung der Information

Neben organisatorischen und baulichen Maßnahmen hier vor allem

- **Kryptographische Maßnahmen**
  - Symmetrische Verfahren
    - Schlüssel K wird zum Ver- und Entschlüsseln verwendet (wenig Aufwand)
    - Ersetzungs- und Versetzungsverfahren (DES: *Data Encryption Standard*)
  - Asymmetrische Verfahren
    - Sie beruhen auf dem Einsatz von zwei einander zugeordneten Schlüsseln S (secret) und P (public).
    - RSA-Verfahren ist am bekanntesten (R. Rivest, A. Shamir, L. Adleman).

➥ Sie heißen auch Public-Key-Verfahren
(typischerweise Faktor 1000 langsamer als symmetrische Verfahren).

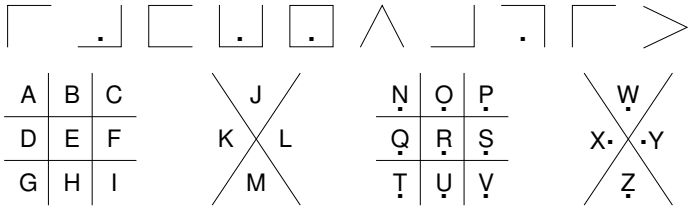
- aber auch: Steganographie

➥ Wer das „Geheimnis“ kennt, kommt auch an die Information!

Kryptographische Verfahren

- **Kryptographie**
  - befaßt sich mit dem Ver- und Entschlüsseln von Nachrichten
  - Sicherheit ergibt sich aus der Qualität des eingesetzten Algorithmus (nicht aus der Verschleierung des Verfahrens ➥ Steganographie).

- **Beispiele aus der Geschichte**
  - Caesar-Chiffre:
    Jeder Buchstabe des Alphabets wird durch seinen Nachfolger ersetzt.
  - Freimaurer (16. Jhd.):
    Ersetzung der Buchstaben durch geometrische Figuren



  - Spanische Geheimschrift (16. Jhd.):
    Ersetzung von Buchstabenpaaren durch spezielle Zeichen:
    vermutlich $25^2 = 625$ Zeichen

- **Verschlüsselungsbeispiel:**

Nachricht M:	Das ist Klartext
Schlüssel K:	azxbazxbazxba
Chiffre C:	xywwusrfeqzphj

- **Schlüsselaustausch**
  - Sender und Empfänger müssen das „Geheimnis“ kennen.
  - Schlüssel K (oder S/P) erlaubt die Entschlüsselung.

Steganographie

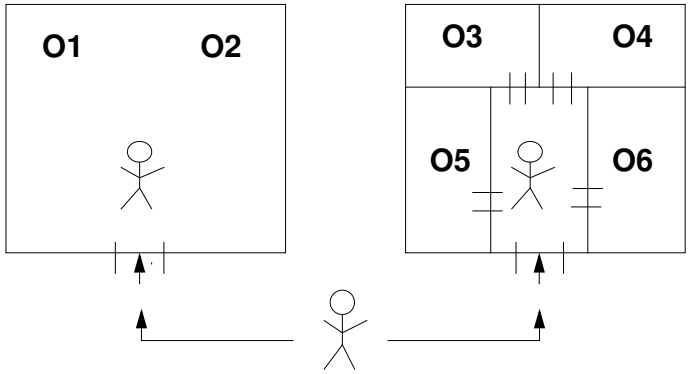
- **Ziel: Verschlüsselte Informationen so zu speichern, daß**
  - niemand diese Informationen findet und daß
  - niemand beweisen kann, daß verschlüsselte Informationen da sind.

  ➥ liefert Argumente für die Gegner staatlich kontrollierter Chiffrierverfahren!
- **Mögliche Anwendung**
  Nutzung der niederwertigsten Bits in Daten vom Typ Bild, Ton, . . .
-

Speicherkontrolle

- **Kernfrage 4:**
  Wie kann ich erreichen, daß die unbefugte Benutzung von Systemressourcen unterbleibt?

  ➥ Frage nach der Verarbeitungskontrolle (bei Prozessen und Dateien)
- **Prozesse und Virtuelle Adreßräume**
  - Isolation durch Prozeß
  - Prozeß ist oft Einheit der Adressierung, der Betriebsmittelvergabe sowie des Schutzes
  - **Analogie:** Haus mit Zimmern



- **Kontrollprobleme bei gemeinsamer Nutzung oder Infiltration**
  - nur Eingangskontrolle
  - Problem des Trojanischen Pferdes
- **Verbesserung**
  - Zugriff auf einzelne Dateien
  - Kontrolle durch Paßwort (Schlüssel)
  - Problem der vielen Schlüssel (Gruppenschlüssel)

  ➥ aber als Zugriffsprinzip: „alles oder nichts“!

‘Trojan Horse’-Problem

- **Problem der Zugriffsbeschränkung bei Programmen**
  („Erschleichung“ der Zugriffsrechte eines Benutzers)

Angriffsbeispiel

Message 3/144 From tamper Dec 24, 98 at 1:19 am 480

Date: Thu, 24 Dec 1998 01:19:18 +0800

To: candy@can.jst.com.ifs.tuwien.ac.at

Subject: candy

Mime-Version: 1.0

--====================5453501:19:17====

Content-Type: text/plain; charset=us-ascii

candy:

software for you.

tamper

--====================5453501:19:17====

Content-Type: application/octet-stream; name=“manager.exe“

Content-Transfer-Encoding: base64

Content-Disposition: attachment; filename=“manager.exe“

TVpQAAIAAAAEAA8A//
8AALgAAAAAAAAAQAAaAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAEAALoQAA4ftAnNIbgBTM0hkJBUaGlzIHByb2dyAAAAAAAAAAAAA
AAAAAAAAEAALoQAA4ftAnNIbgBTM0hkJBUaGlzIHByb2dyYW0gbXVzdCBiZS
BydW4gdW5kZXIgV2luMzINCiQ3AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAVVBFAABMA

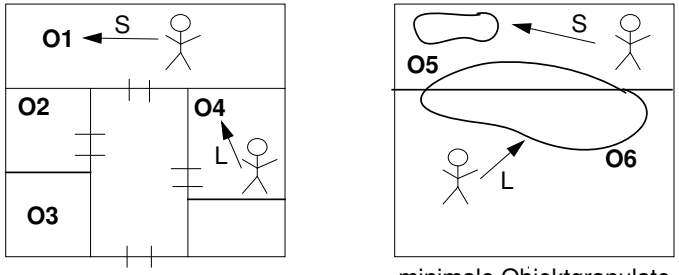
Zugriffskontrolle

- **Kernfrage 5:**
  Wie kann ich erreichen, daß Benutzer mit unterschiedlichen Rechten gemeinsam auf Daten zugreifen können?

  ➥ Frage nach der Zugriffskontrolle (bei Daten)

- **Zugriffskontrolle (Autorisierung)**
  - Vergabe von Zugriffsrechten (Lesen, Schreiben, . . .) auf DB-Objekten, Programmen usw.
  - **Ziele**
    - Verhinderung von zufälligen oder böswilligen Änderungen
    - möglichst weitgehende Isolation von Programmfehlern
    - Verhinderung von unberechtigtem Lesen/Kopieren

- **Analogie**



minimale Objektgranulate
(z. B. durch Sichten)

- **Kontrollkonzepte:**
  - unbeschränkte Zugriffskontrolle (Schlüssel + Privileg)
  - Teilordnung der Nutzungsprivilegien
  - Prinzip des kleinstmöglichen Privilegs
    - Need-to-know-Prinzip
    - **Least Privilege Principle**
  - kooperative Autorisierung (n Schlüssel)

Autorisierungsmodell

- **Explizite Autorisierung[1]:**
  - Der Zugriff auf ein Objekt o kann **nur** erfolgen, wenn für den Benutzer (Subjekt s) ein Zugriffsrecht (Privileg p) vorliegt
  - Autorisierungsregel (o, s, p) legt eine **explizite starke** Autorisierung mit **positivem** Recht fest

- **Schutzinformation als Zugriffsmatrix**

  Subjekte: Benutzer, Programme, Terminals

  Objekte: Programme (Anwendungs-, Dienstprogramme), DB-Objekte (Relationen, Sichten, Attribute)

  Zugriffsrechte: Lesen, Ändern, Ausführen, Erzeugen, Weitergabe von Zugriffsrechten usw., ggf. abhängig von Terminal, Uhrzeit usw.

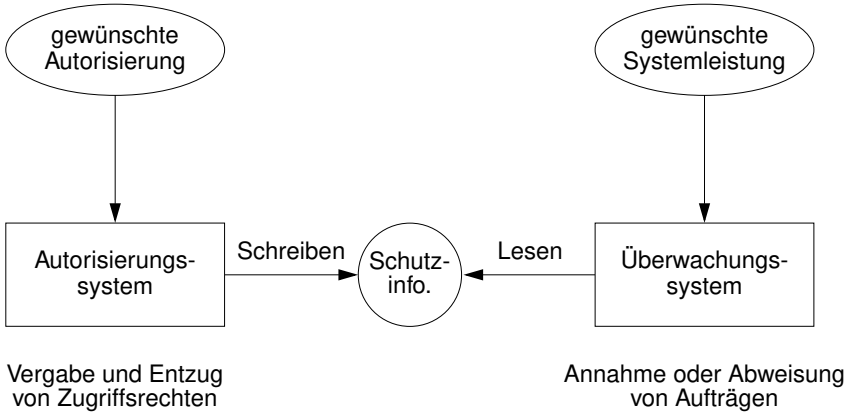
Subjekte, Benutzer	Objekte				
	O1	O2	O3	...	On
B1	P1, P2		P3		Pi
B2		P1	P2, P3		P1
B3		P2, P3	P2		
•					
•					
•					
Bm	P1, P2	Pi	P1		Pi, Pk

- Zugriffsmatrix ist typischerweise **sehr groß und dünn besetzt**
  Welche Realisierungstechniken bieten sich an?

1. Dieses Modell wird im Englischen als Discretionary Access Control (DAC) bezeichnet. Wegen seiner Einfachheit ist DAC weit verbreitet.
   („discretionary“ bedeutet in etwa „nach dem Ermessen des Subjekts“.)

Autorisierungsmodell (2)

- **Autorisierungs[1]- und Überwachungssystem**



- **Autorisierung**
  - zentrale Vergabe der Zugriffsrechte (DBA)
  - dezentrale Vergabe der Zugriffsrechte durch Eigentümer der Objekte
- **Objektgranulat**
  - wertunabhängige oder
  - wertabhängige Objektfestlegung (Sichtkonzept)
- **Wirksamkeit der Zugriffskontrolle beruht auf drei Annahmen:**
  - fehlerfreie Benutzer-Identifikation/-Authentisierung
  - erfolgreiche Abwehr von (unerwarteten) Eindringlingen (vor allem strikte Isolation der Benutzer- und DBS-Prozesse sowie Übermittlungskontrolle)
  - Schutzinformation ist **hochgradig geschützt!**

1. Verfeinerungen des Modells gestatten eine implizite Autorisierung von Subjekten, Operationen und Objekten durch Nutzung entsprechender Hierarchien.

Zugriffskontrolle in SQL

- **Sicht-Konzept erlaubt wertabhängigen Zugriffsschutz**
  - Untermengenbildung, Verknüpfung von Relationen, Verwendung von Aggregat-Funktionen
  - Umsetzung durch **Anfragemodifikation** möglich
- **Vergabe von Rechten**

```
GRANT    {privileges-commalist | ALL PRIVILEGES}
         ON accessible-object TO grantee-commalist
         [WITH GRANT OPTION]
```

- **Objekte** (accessible-object)
  - Relationen bzw. Sichten
  - aber auch: Domänen, Datentypen, Routinen usw.
- **Zugriffsrechte** (privileges)
  - SELECT, INSERT, UPDATE, DELETE, REFERENCES, USAGE, TRIGGER, CONNECT, EXECUTE, . . .
  - Attributeinschränkung bei INSERT, UPDATE und REFERENCES möglich
  - Erzeugung einer „abhängigen“ Relation erfordert REFERENCES-Recht auf von Fremdschlüsseln referenzierten Relationen.
  - USAGE erlaubt Nutzung spezieller Wertebereiche (character sets).
  - dynamische Weitergabe von Zugriffsrechten: WITH GRANT OPTION (GO: dezentrale Autorisierung)
- **Empfänger** (grantee)
  - Liste von Benutzern bzw. PUBLIC
  - Liste von Rollennamen

Zugriffskontrolle in SQL (2)

- **Beispiele:**
  - GRANT SELECT ON Abt TO PUBLIC
  - GRANT INSERT, DELETE ON Abt
    TO Mueller, Weber WITH GRANT OPTION
  - GRANT UPDATE (GEHALT) ON Pers TO Schulz
  - GRANT REFERENCES (Pronr) ON Projekt TO PUBLIC
- **Rücknahme von Zugriffsrechten:**

> **REVOKE** [GRANT OPTION FOR] privileges-commalist
> ON accessible-object FROM grantee-commalist
> {RESTRICT | CASCADE}

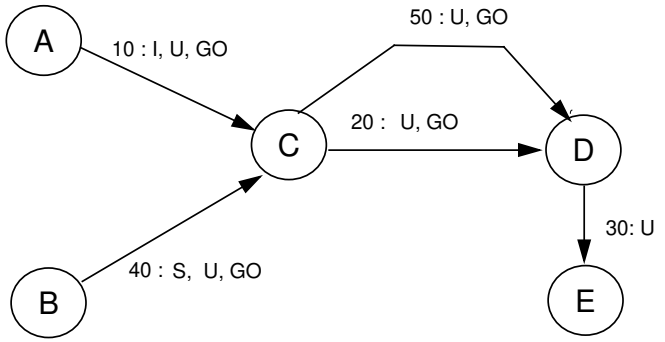
Beispiel: **REVOKE SELECT ON Abt FROM Weber CASCADE**

- **ggf. fortgesetztes Zurücknehmen von Zugriffsrechten**
- **Wünschenswerte Entzugssemantik:**
  Der Entzug eines Rechtes ergibt einen Schutzzustand, als wenn das Recht nie erteilt worden wäre.
- **Probleme:**
  - Rechteempfang aus verschiedenen Quellen
  - verschiedene Entzugssemantiken:
    - zeitabhängige Interpretation
    - zeitunabhängige Interpretation
- **Führen der Abhängigkeiten in einem *Autorisierungsgraph* erforderlich**

Zugriffskontrolle in SQL (3)

- **Autorisierungsgraph mit zeitabhängiger Interpretation:**
  Der Entzug eines Rechtes ergibt einen Schutzzustand, als wenn das Recht nie erteilt worden wäre.
  - ➥ Vergabe von Zeitstempeln für jede Autorisierungsoperation erforderlich
- **Beispiel:**

10 : A: GRANT INSERT, UPDATE ON Pers TO C WITH GRANT OPTION
20 : C: GRANT UPDATE ON Pers TO D WITH GRANT OPTION
30 : D: GRANT UPDATE ON Pers TO E
40 : B: GRANT SELECT, UPDATE ON Pers TO C WITH GRANT OPTION
50 : C: GRANT UPDATE ON Pers TO D WITH GRANT OPTION



70 : A: REVOKE INSERT, UPDATE ON Pers FROM C

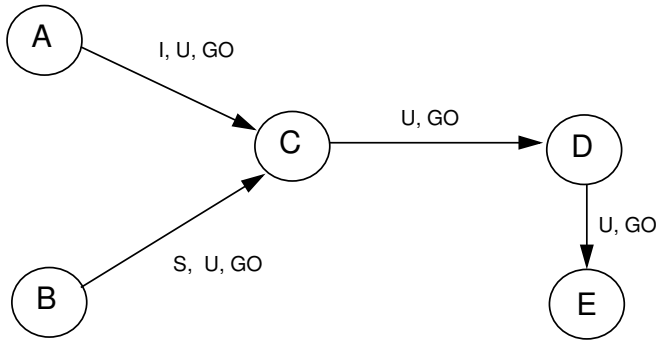
Zugriffskontrolle in SQL (4)

- **Autorisierungsgraph mit zeitunabhängiger Interpretation:**
  Der rekursive Entzug eines Rechtes wird nicht fortgesetzt, sobald der Geber noch mindestens ein gleiches Recht für das Objekt von einer unabhängigen Quelle hat.

  ➥ Überprüfung der Quellenunabhängigkeit bei jeder Rechtevergabe erforderlich

- **Beispiel:**

  A: GRANT INSERT, UPDATE ON Pers TO C WITH GRANT OPTION
  C: GRANT UPDATE ON Pers TO D WITH GRANT OPTION
  D: GRANT UPDATE ON Pers TO E WITH GRANT OPTION
  B: GRANT SELECT, UPDATE ON Pers TO C WITH GRANT OPTION
  C: GRANT UPDATE ON Pers TO D WITH GRANT OPTION



A: REVOKE INSERT, UPDATE ON Pers FROM C

Verfeinerungen des Autorisierungsmodells

- **Implizite Autorisierung**
  - hierarchische Anordnung von Subjekten, Objekten und Operationen
  - explizite Autorisierung auf einer Hierarchiestufe bewirkt implizite Autorisierungen auf anderen Hierarchiestufen

- **Negative Autorisierung**
  - stellt ein Verbot des Zugriffs ($\neg$p) dar
  - kann explizit und implizit erfolgen

- **Schwache Autorisierung**
  - kann als Standardeinstellung verwendet werden
    (Leserecht eines Objektes für gesamte Gruppe, die aus Teilgruppen besteht)
  - erlaubt Überschreibung durch starkes Verbot
    (Teilgruppe erhält explizites Leseverbot)
  - Schreibweise: [. . .]

- **Autorisierungsalgorithmus**

  wenn es eine explizite oder implizite **starke** Autorisierung (*o, s, p*) gibt,
  dann erlaube die Operation

  wenn es eine explizite oder implizite **starke negative** Autorisierung (*o, s,* $\neg$*p*) gibt
  dann verbiete die Operation

  ansonsten

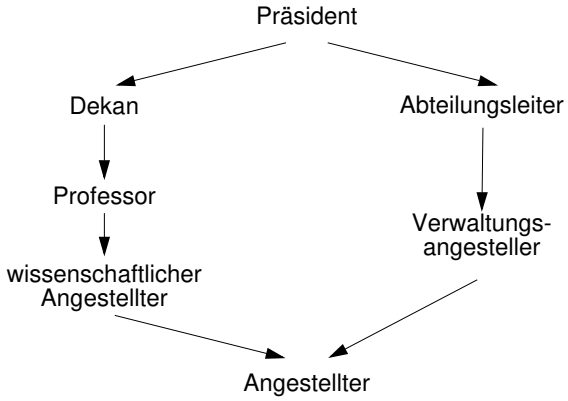
  wenn es eine explizite oder implizite **schwache** Autorisierung [*o, s, p*] gibt,
  dann erlaube die Operation

  wenn es eine explizite oder implizite **schwache negative** Autorisierung [*o, s,* $\neg$*p*] gibt,
  dann verbiete die Operation

  sonst verbiete die Operation

Verfeinerungen des Autorisierungsmodells (2)

- **Implizite Autorisierung von Subjekten**
  - Einführung von Rollenhierarchien
  - zwei ausgezeichnete Positionen
    - eine eindeutige Rolle mit der maximalen Menge an Rechten (z.B. Präsident, Systemadministrator)
    - eine eindeutige grundlegende Rolle (z.B. Angestellter, Hiwi)



- **Explizite positive Autorisierung**

  ➥ implizite positive Autorisierung auf allen höheren Hierarchiestufen

- **Explizite negative Autorisierung**

  ➥ implizite negative Autorisierung auf allen niedrigeren Hierarchiestufen

Rollenkonzept in SQL

- **Rollenkonzept**
  - bisher: (explizite) Zuordnung von Zugriffsrechten zu Benutzern
  - SQL99 erlaubt die Definition von Rollen
  - **Ziel:** Vereinfachung der Definition und Verwaltung komplexer Mengen von Zugriffsrechten
    - Erzeugung von Rollen und Vergabe von Zugriffsrechten (Autorisierungen)
    - Kontrolle der Aktivitäten (Einhaltung der vorgegebenen Regeln)

- **Wichtige Rollen**

  - **Systemadministrator**
    - Sie „besitzt“ sämtliche Ressourcen des DBS und ist zur Ausführung einer jeden DB-Anweisung autorisiert.
    - Rolle verwaltet eine DBS-Instanz, die mehrere DBS umfassen kann.
    - Bei DB2/UDB gibt es beispielsweise zwei Untergruppen: Systemkontrolle und Systemwartung.

  - **DB-Administrator**
    - Rolle gilt für eine spezielle DB mit allen Zugriffsrechten.

  - **Anwendungsentwickler**
    - typische Zugriffsrechte: Verbindung zur DB herstellen (CONNECT), Tabellen erzeugen, AWPs an DB binden
    - Zugriffsrechte beziehen sich auf Menge spezieller DB-Objekte.
    - Kapselung von Rechten durch AWP bei statischem SQL

  - **Endbenutzer**
    - Rechte für Ad-hoc-Anfragen
    - CONNECT- und EXECUTE-Rechte für AWPs

Rollenkonzept in SQL (2)

- **Definition von Rollen**

  CREATE ROLE Revisor
  CREATE ROLE Hauptrevisor

- **Vergabe von Rechten**

  GRANT INSERT ON TABLE Budget TO Revisor

- **Zuweisung von Rollen**

  GRANT role-granted-commalist TO grantee-commalist
  [WITH ADMIN OPTION]

  - Rollen werden Benutzern und Rollen **explizit** zugewiesen.
  - WITH ADMIN OPTION erlaubt die Weitergabe von Rollen.
  - Beispiel: GRANT Revisor TO Weber

- **Entzug von Rollen**

  REVOKE role-revoked-commalist
  FROM grantee-commalist

  - Beispiel: REVOKE Revisor FROM Hauptrevisor
  - WITH ADMIN OPTION ist „vorsichtig“ einzusetzen;
    kein fortgesetztes Zurücknehmen vorgesehen

Rollenkonzept in SQL (3)

- **Anwendung**

  - Momentaner Rechtebesitz

Revisor	:	P1, P2
Hauptrevisor	:	P3, P4
Benutzer Schmidt	:	P1

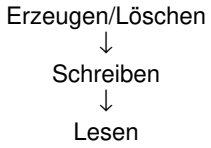
  - GRANT Revisor TO Hauptrevisor WITH ADMIN OPTION

  - GRANT Hauptrevisor TO Schmidt

  - Revoke Revisor FROM Hauptrevisor

Verfeinerungen des Autorisierungsmodells (3)

- **Implizite Autorisierung von Operationen**
  - Festlegung von Operationshierarchien



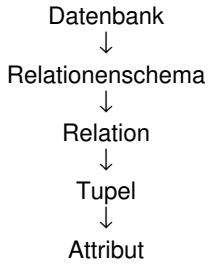
- **Explizite positive Autorisierung**

  ➥ implizite positive Autorisierung auf allen niedrigeren Ebenen

- **Explizite negative Autorisierung**

  ➥ implizite negative Autorisierung auf allen höheren Ebenen

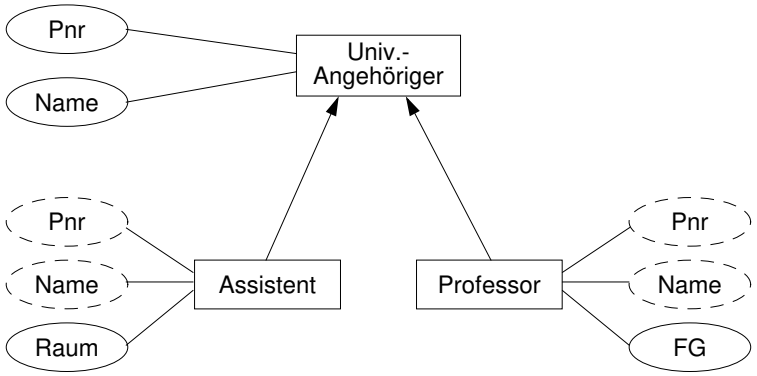
- **Implizite Autorisierung von Objekten**
  - Festlegung von Granularitätshierarchien



- **Implikationen sind abhängig von Operationen**
  - Explizites Erlaubnis zum Lesen einer Relation impliziert das Recht, das Schema der Relation zu lesen
  - Explizite Erlaubnis zum Lesen eines Objektes impliziert automatisch, alle Objekte feinerer Granularität zu lesen
  - Definition einer Relation hat keine Implikationen

Verfeinerungen des Autorisierungsmodells (4)

- **Implizite Autorisierung in Typhierarchien**
  - weitere Dimension der Autorisierung durch Generalisierung/Spezialisierung



- **Benutzergruppen**
  - Verwaltungsangestellte dürfen die Namen aller Univ.-Angehörigen lesen
  - Wissenschaftliche Angestellte dürfen Namen und Fachgebiet (FG) aller Professoren lesen

- **Anfragen**
  - Q1: Lese die Namen aller Univ.-Angehörigen
  - Q2: Lese Namen und Fachgebiet aller Professoren

- **Drei Grundregeln**
  - Ein Zugriffsrecht auf einen Objekttyp impliziert ein gleichartiges Zugriffsrecht auf die vererbten Attribute in den Subtypen
  - Ein Zugriffsrecht auf einen Objekttyp impliziert auch ein gleichartiges Zugriffsrecht auf alle ererbten Attribute
  - Ein in einem Objekttyp definiertes Attribut ist mit den Zugriffsberechtigungen auf seinem Supertyp nicht zugreifbar

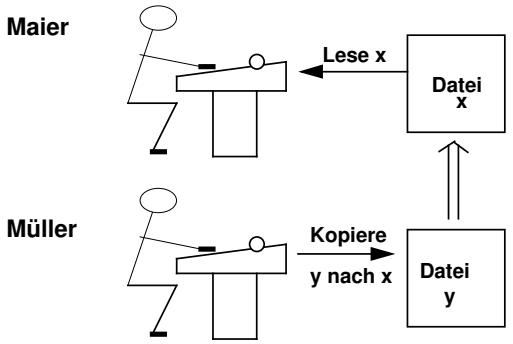
Verfeinerungen des Autorisierungsmodells (5)

- **Hierarchische Objekt-/Subjektklassifikation nach Sicherheitsstufen**
  - Sicherheitshierarchie (besonders im militärischen Bereich): streng geheim, geheim, vertraulich, unklassifiziert
  - hierarchische Klassifikation von Vertrauenswürdigkeit (bei Subjekten) und Sensitivität (bei Objekten)
    - clear (s), mit s als Subjekt (clearance)
    - class (o), mit o als Objekt (classification)

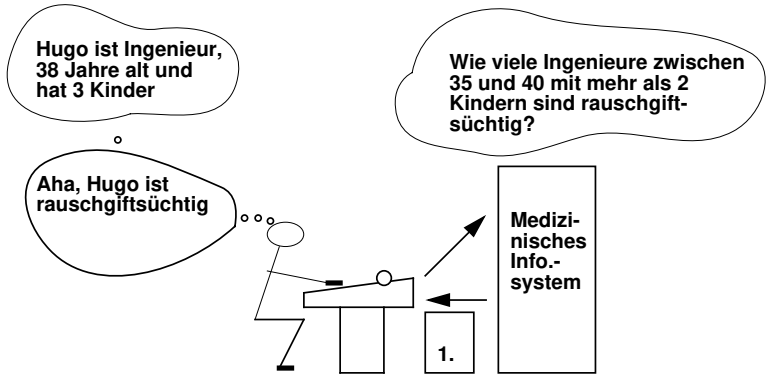
- **Mandatory Access Control:**
  - MAC-Modell realisiert hierarchische Objekt-/Subjektklassifikation mit zwei Zugriffsregeln
    - Ein Subjekt s darf ein Objekt o nur lesen, wenn
      class (o) ≤ clear (s)
    - Ein Objekt o muß mit mindestens der Einstufung des Subjektes s geschrieben werden:
      clear (s) ≤ class (o)

    ➥ kein „write down“ möglich
  - bietet potentiell größere Sicherheit, aber Benutzer unterschiedlicher Sicherheitsstufen können nur schwer zusammenarbeiten

Datenflußkontrolle



Inferenzkontrolle



Sicherheitsprobleme in statistischen DB

- **Datenschutzforderung:**
  Zu Forschungszwecken sind personenbezogene Daten zu anonymisieren. Es ist lediglich die Verwendung statistischer Funktionen wie AVG, MIN, MAX, COUNT erlaubt.

- **Einzelwerte dennoch oft ableitbar bei**
  - selektiven Anfragen (kleine Treffermengen)
  - Ergebnisverknüpfung mehrerer Anfragen

- **Beispiel:**
  - statistische Anfragen auf Pers ohne Attribute Pnr und Name
  - Wissen über bestimmte Personen (z. B. Alter, Beruf, Familienstand, Firmenzugehörigkeit) kann leicht für gezielte Anfragen genutzt werden.

```
SELECT COUNT (*)
FROM Pers
WHERE Alter = 51 AND Beruf = 'Operateur'

SELECT AVG (Gehalt)
FROM Pers
WHERE Alter = 51 AND Beruf = 'Operateur'
```

  ➥ Bei mehr als einem Treffer kann Treffermenge durch weitere Bedingungen reduziert werden.

- **Abhilfemöglichkeiten:**
  - Antwortausgabe nur, wenn Treffermenge über festgelegtem Grenzwert liegt
  - Überprüfung, ob mehrere Anfragen aufeinander aufbauen
  - gezielte Einstreuung von kleineren Ungenauigkeiten

Sicherheitsprobleme in statistischen DB (2)

- **Bsp: N = 13, M = 5**

Name	Geschlecht	FB	Beginn	Abi-Note	VD-Note
Abel	W	Inf	1997	1.6	1.5
Bebel	W	Etech	1997	2.7	2.2
Cebel	M	Etech	1996	1.5	1.3
Damm	W	Inf	1995	1.0	1.0
Ehrlich	M	Bio	1996	2.8	2.6
Fuchs	M	Etech	1990	2.5	1.8
Grommel	M	Inf	1997	1.3	1.2
Heinrich	W	Chem	1996	2.5	2.0
Ibsen	M	Inf	1998	1.6	1.6
Jahn	W	Bio	1999	1.3	1.2
Kramer	W	Math	1995	2.8	2.2
Lustig	M	Etech	1989	1.6	1.8
Müller	M	Inf	1997	1.4	1.3

- **Statistische DB**

#Werte:	Geschlecht	:	2
	FB	:	10
	Abinote	:	31

Sicherheitsprobleme in statistischen DB (3)

- **Charakteristische Formel C:** C = (Geschlecht='W' $\wedge$ FB='Etech')

  1. COUNT (C) =

  2. SUM (C : VD-Note) =

  oder

  1. COUNT (C $\wedge$ VD-Note = 2.1) =

  2. COUNT (C $\wedge$ VD-Note = 2.2) =

- **Forderung**

  |C| > 1

  SUM (C : VD-Note) verboten

  (C: $\underbrace{\quad}_{1}$)

  SUM (Etech : VD-Note) - SUM (Etech $\wedge$ M : VD-Note)

  COUNT(C) = N - COUNT($\neg$C)

Zusammenfassung

- **BDSG regelt die Verarbeitung personenbezogener Daten**
  - Verbotsprinzip mit Erlaubnisvorbehalt
  - Technische Maßnahmen (die sog. Zehn Gebote) müssen stets den veränderten Randbedingungen der Datenverarbeitung angepaßt und neu interpretiert werden.
- **Aufeinander abgestimmte Sicherheitskonzepte sind wesentlich**
  - Zugangs- und Benutzerkontrolle
  - starke Verfahren zur Authentisierung
  - kryptographische Maßnahmen zur Datenübertragung
  - Isolation der Prozesse
  - Prinzip der Zugriffskontrolle: **Least Privilege Principle**
  - Sicherungsanforderungen gelten allgemein in Rechensystemen und insbesondere zwischen Anwendung und DBS.

➥ Das „schwächste Glied" in der Kette der Sicherheitsmaßnahmen bestimmt die Sicherheit des Gesamtsystems!

- **Zugriffskontrolle in DBS**
  - wertabhängige Festlegung der Objekte (Sichtkonzept)
  - Vielfalt an Rechten erwünscht
  - zentrale vs. dezentrale Rechtevergabe
  - verschiedene Entzugssemantiken bei dezentraler Rechtevergabe
  - Rollenkonzept: vereinfachte Verwaltung komplexer Mengen von Zugriffsrechten
- **Sicherheitsprobleme**
  - Datenflußkontrolle
  - Inferenzkontrolle
  - Wenn Zusatzwissen vorhanden ist, lassen statistische DBs die Individualisierung von anonymisierten Daten zu.