DB

- Entwurf
  - Miniwelt
    - Anforderungsanalyse
      - 1. intensiv mit Anwendern
      - 2. id. Organisationseinheiten
      - 3. id. Aufgaben
      - 4. id. Beziehungen
      - 5. Vollständigkeit
      - 6. Eindeutigkeit
  - konzeptioneller Entwurf
    - ER Modell
      - Motivation
      - Grenzen
        - puristisch
        - Anwendungsspezifische Konsistenzbedingungen
      - Modellierungsstruktur
        - 1. Entities
          - wohl unterscheidbarer echter Körper
          - schwache Entitities
        - 2. Relationen
        - 3. Attribute
          - Entities beschreiben
          - Key Attribute: unterstreichen
          - multi valued Attribute (Doppelkreis)
        - 4. Rollen
        - 5. Funktionalitäten / Cardinalities
          - alt. (min-max) Notation
        - 6. Generalisierung / Spezialisierung
          - disjunkt
          - n. disjunkt
          - total
          - partial
        - 7. Aggregation
    - UML (oo)
      - 1. Struktur- / Klassen Diagramm
        - 1. Klasse
          - Klassenname
          - Attribute
          - Methoden
        - 2. Assoziation
          - Kommunikationsstruktur (vgl. Relationen / Rollen) mit Funktionalitäten
        - 3. Generalisierung / Spezialisierung
        - 4. Aggregation
          - "consists of"
        - 5. Composition
          - Aggregation + existenzabhängige Besitzerstruktur
      - 2. Use Case Diagramm
        - Benutzer Sichtweise
      - 3. Interaction Diagramm
        - Sequenz Diagramm
          - Fokus: time sequence of tasks
        - Kollaboration Diagramm
          - Fokus: Nachrichtenaustausch
      - 4. State Diagramm
        - Verhalten und Status eines Objekts
        - fokus: Status transitionen bei externen events
          - (Situation / Action)
      - 5. Activity Diagramm
        - Aktivitäten statt Status
    - semantisches Datenmodell
    - top down vs. bottom up
  - Logische Datenmodelle
    - 1. Netz od. hierarchisches
    - 2. deduktives
    - 3. relationales
    - 4. oo
    - 5. XML
  - Implementierung
  - Validierung
  - Ziele der DB
    - 1. korrekt
    - 2. komplett
    - 3. minimal
    - 4. sicher
    - 5. effizient
    - 6. adaptierbar
- oo
  - Motivation
    - (Klasse entspricht einem Relationenschema)
    - OID
    - Verhalten
      - Verhalten in Objekt einmal gespeichert statt in jeder Applikation
    - Überladen
      - Wiederverwendung von Operationen für semantisch ähnliche operationen
    - Vererbung (Is A)
      - Subtyp erbt alle features des Supertyps
    - Spezialisierung
      - zusätzliche features (durch Modifikation)
    - Verfeinerung
      - hinzufügen oder neudifinieren von Attribute / Operationen
    - Substituierbarkeit
      - Polymorphismus
  - Objekt
    - OID
    - Wert/ Zustand
    - Objekttyp
      - Struktur
      - Verhalten
- Einführung
  - Datenbasis
    - DBMS
    - Motivation
      - 1. Mehrbenutzerzugriff
      - 2. Recovery
      - 3. k. Redundanz
      - 4. Konsistenz
      - 5. Zugriffsmöglichkeiten / Sicherheit
      - 6. Integritätsbedingungen
      - 7. Entwicklungskosten
    - Abstraktionsebenen
      - physische Ebene
      - logische Ebene
      - Sichten
      - => Datenunabhängigkeit
    - Datenmodelle bestehen aus
      - DDL
        - Datenbankschema
        - create, alter
      - DML
        - Datenbankausprägung
        - insert into, update
    - 3 views
      - computer science
      - application
      - user
- relational
  - def.
    - Relation
    - Domäne
    - Relationen Schema
    - Transaktion
      - ACID
    - Schlüssel
  - SQL
    - 1. SELECT * FROM R WHERE A = ...
      - Selektion
    - 2. SELECT DISTINCT A,B FROM R
      - Projektion
    - 3. SELECT * FROM R UNION SELECT * FROM T
      - R U T
    - 4. SELECT * FROM R MINUS SELECT * FROM T
      - R - T
    - 5. SELECT * FROM R,S WHERE A <=/> B
      - Join, auf gleichen Sch(R)
    - 6. SELECT R.A,B,C,D FROM R JOIN S ON R.A = S.A
      - Join (JOIN ... ON ...)
    - 7. SELECT * FROM R,T
      - R x T
      - Kreuzprodukt
    - 8. SELECT R AS B FROM ...
    - 9. BSP
      - 1. SELECT DISTINCT ...
      - 2. FROM ...
      - 3. JOIN ... ON ...
      - 4. WHERE ...
      - 5. GROUP BY ...
      - 6. HAVING ...
      - 7. ORDER BY ...
    - 10. Joins
      - N . (I) J
      - N L (O) J
      - N R (O) J
      - N . O J
      - . . (I) J ON
      - . L (O) J ON
      - . R (O) J ON
      - . . O J ON
  - relationale Algebra
    - prozedural
    - deskriptiv
  - relationales Kalkulus
    - Tupelkalkül
      - {t|P(t)}
      - {t|E u,v (u e T1 & v e T2 & u[A1] = ... & t[person] = ...)} t = <person>
      - bindet eine Variable an Tupel einer Relation
    - Domänenkalkül
      - {<v1,...,vn>|P(v1,...,vn)}
      - {<v1,v2>|E v3,v4...vn (<v1,...,vk> e T1 & v3 = ...)}
      - bindet Sequenz von Domänenvariablen an Domänen von Attributen
    - sichere Ausdrücke
      - Ergebnis des Ausdrucks Teilmenge der Domäne
      - Domäne besteht aus Menge aller Konstanten und Attributwerten, die in der Formel vorkommen oder referenziert werden
  - Konsistenzbedingungen
    - Domain
      - value specifications
    - Tupel
      - connection between values of diff. attributes, fd
    - Relation
      - key constraint, aggreg.
    - Referential
      - foreign key, inclusion, exclusion
    - fd
    - Armstrong Axiome
      - Reflexivität
      - Verstärkung
      - Transitivität
  - NF
    - Motivation
      - Idee
      - Verlustlos
      - Abhängigkeitsbewahrend
    - NFNF
    - 1NF
    - 2NF
      - Anomalieen
      - S# S V# V
      - jedes NSA ist von jedem SK v.fd abhängig
    - 3NF
      - synthese
      - S# S Ort Land
      - kein NSA ist von einem SK transitiv abhängig
    - BCNF
      - dekomposition
      - Jede nicht-triviale Determinate ist SK
    - Nachteile
      - Join über Relationen
  - Probleme
    - 1. Segmentierung
    - 2. impedance Match
    - 3. künstliche Schlüsselattribute
    - 4. fehlendes Verhalten
      - Transformation
      - Operationen werden nicht in DBMS verwaltet