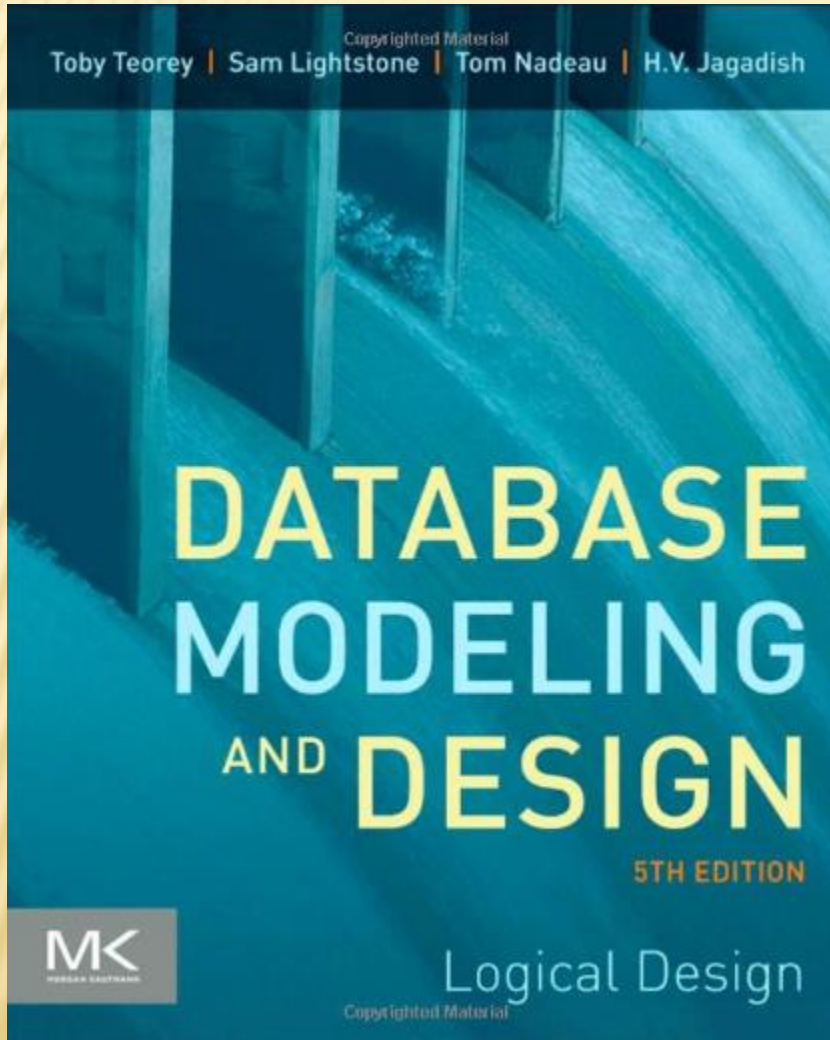


Prof. Dr. Ermir Rogova

DATABAZAT

Pjesa 1

LITERATURA



Database Modeling
and Design: Logical
Design, 5th Edition

Toby Teorey, Sam Lightstone,
Tom Nadeau, H.V. Jagadish

HYRJE

- ✘ Teknologjia e databazave ka evoluar shpejt këto tri dekadat e fundit
- ✘ Shumë sisteme të specializuara të databazave (orientuar në objekte, hapsinore, multimediale, etj) kanë gjetur përdomin substancial në komunitetet shkencore dhe ingjinierike
- ✘ Megjithatë, sistemet relacionale (tabelore) mbetin teknologjia dominuese e databazave për ndërmarrjet biznesore

MENAGJIMI I DATABAZES

- ✘ Komponenti themelor në databazen relacionale është atributi ose kolona
- ✘ Kjo është njësia më e vogël që ka kuptim në botën reale, p.sh. Emri, mbiemri, etj.
- ✘ Një grup artikujsh që kanë lidhje ndër vete trajtohen si një njësi dhe quhen *record* (shënim), rresht apo tuple
- ✘ Një koleksion rreshtash të një lloji formon tabelën
- ✘ Databaza është një objekt më kompleks. Është një koleksion i tabelave të ndërlidhura të llojeve të ndryshme.
- ✘ Motivimi për përdorimin e databazave është aftësia e saj për të siguruar qasje shumë përdoruesëve , integrimi i të dhënave, qasje më e lehtë në to dhe me pak redundance (duplikim i panevojshëm)

SISTEMI I MENAGJIMIT TË DATABAZAVE (DBMS)

- ✘ Sistemi i menagjimit të databazave është një sistem sofverik i përgjithshuar për manipulimin e databazave

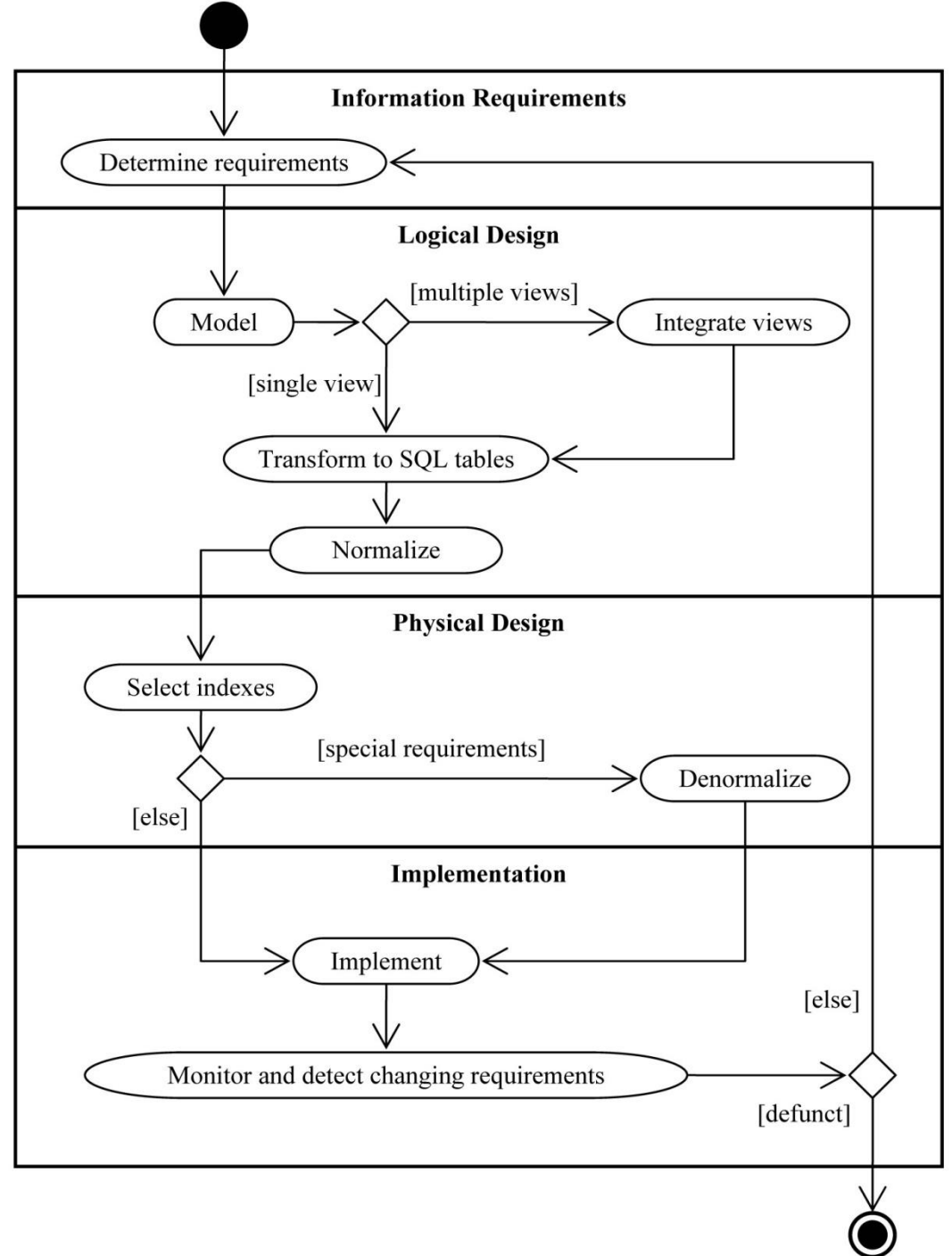
- ✘ Ky sistem mbështet:
 - + pamjet logjike (skemat, subskemat),
 - + pamjen fizike (metodat e qasjes, klasteret e të dhënave)
 - + gjuhën e definimit të të dhënave (DDL)
 - + gjuhën e manipulimit të të dhënave (DML)
 - + dhe shërbime të ndryshme si:
 - ✘ Menagjimi i transaksioneve,
 - ✘ integriteti i të dhënave
 - ✘ restaurimi pas dështimit
 - ✘ Siguria

- ✘ Sistemet relacionale të databazave sigurojnë edhe nivel të lartë të pavarshmërisë së të dhënave

CIKLI JETËSOR I DATABAZËS

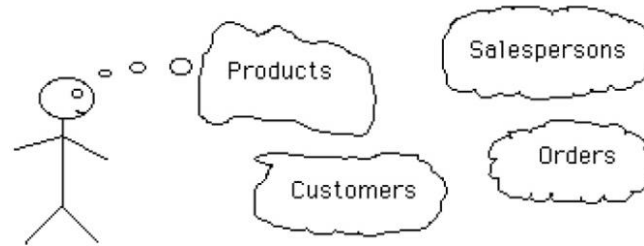
- ✘ Cikli jetësor i databazës inkorporon hapat themelor të përfshirë në:
 - + dizajnimin e skemës globale të databazës logjike
 - + shpërndarjen e të dhënave përmes një rrjeti kompjuterik, dhe
 - + definimin e skemave lokale specifike
- ✘ Kur dizajni kompletohet, cikli jetësor vazhdon me implementimin e mirëmbajtjen e databazës

CIKLI JETËSOR



ANALIZA E INFORMACIONEVE

Step I Information Requirements (reality)



- ✘ Nevojat e databazes përcaktohen duke intervistuar prodhuesit dhe shfrytëzuesit e të dhënave
- ✘ Informatat e marra shfrytëzohen për të krijuar specifikacionet formale të nevojave/kërkesave
- ✘ Këto specifikacione përfshijnë:
 - + të dhënat e nevojshme për procesim,
 - + marrëdhëniet në mes të dhënave, dhe
 - + platforma softverike për implementimin e databazës

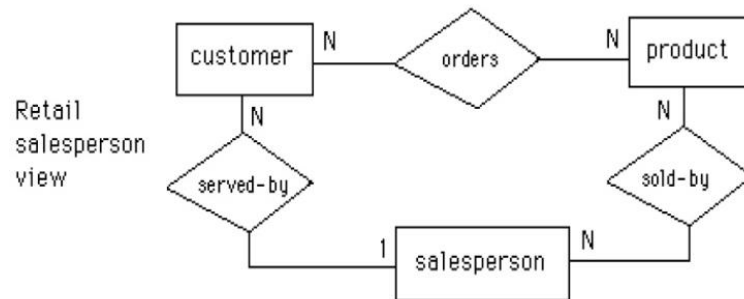
DIZAJNIMI LOGJIK

- ✘ Skema globale, e cila është diagram konceptual i modelit të të dhënave.
- ✘ Ajo paraqet të gjitha të dhënat dhe marrëdhëniet në mes tyre.
- ✘ Zhvillohet duke përdorur teknika si ER ose UML
- ✘ Konstruktet e modelit duhen përfundimisht të transformohen në tabela

MODELIMI KONCEPTUAL

Step II Logical design

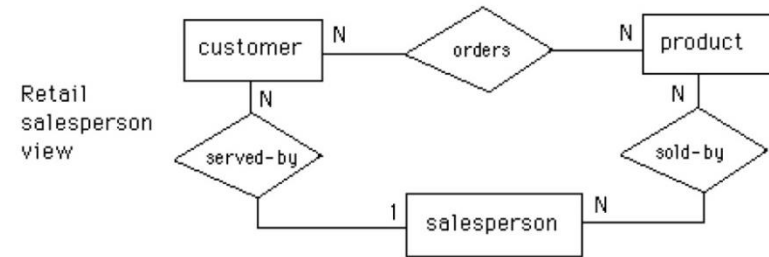
Step II.a Conceptual data modeling



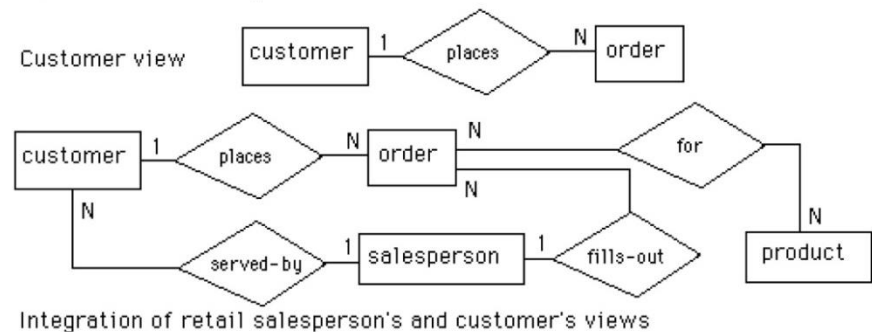
- ✘ Kërkesat analizohen dhe modelohen duke përdorur diagramet ER ose UML
- ✘ Këto përfshijnë tipare të ndryshme që do të shtjellohen më vonë, si semantika për marrëdhëniet opcionale, marrëdhëniet trinare, supertipet dhe nëntipet, etj.

INTEGRIMI I PAMJES

- ✘ Zakonisht, kur dizajni është i madh dhe më shumë persona janë të përfshirë në analizën e kërkesave, paraqiten disa pamje të të dhënave dhe marrëdhënieve në mes tyre
- ✘ Këto rezultojnë në mospërputhje për shtak të ndryshimeve në taxonomi, kontekst ose perceptim



Step II.b View integration



- ✘ Për të eliminuar redundancën dhe mospërputhjet në model, këto pamje duhen të racionalizohen dhe konsolidohen në një pamje të vetme globale.
- ✘ Integrimi i pamjes kërkon përdorimin e mjeteve të ER-it siç janë identifikimi i sinonimeve, agregacioni dhe gjeneralizimi

TRANSFORMIMI I MODELIT NE TABELA

Step II.c Transformation of the conceptual data model to SQL tables

Customer

cust-no	cust-name

Product

prod-no	prod-name	qty-in-stock

Salesperson

sales-name	addr	dept	job-level	vacation-days

Order

order-no	sales-name	cust-no

Order-product

order-no	prod-no

```
create table customer
(cust_no integer,
 cust_name char(15),
 cust_addr char(30),
 sales_name char(15),
 prod_no integer,
 primary key (cust_no),
 foreign key (sales_name)
 references salesperson,
 foreign key (prod_no)
 references product):
```

- ✘ Duke u bazuar në rregullat e paracaktuara, çdo entitet, së bashku me marrëdhëniet në mes tyre transformohet në tabela, duke eliminuar ato të panevojshmet

NORMALIZIMI I TABELAVE

Step II.d Normalization of SQL tables

Decomposition of tables and removal of update anomalies.

Salesperson

sales-name	addr	dept	job-level

SalesVacations

job-level	vacation-days

- ✘ Ndodh kur në tabelat e caktuara ku një apo disa attribute mvaren nga nje apo disa attribute të tjera
- ✘ Mvarshmëritë funksionale (functional dependencies) derivohen nga modeli konceptual dhe nga analiza e kerkesave
- ✘ Ato paraqesin mvarshmëritë në mes elementeve që janë identifikatorë unikatë si dhe attributeve të entiteteve, dhe janë mjet i rëndësishëm në normalizim

DIZAJNIMI FIZIK

- ✘ Dizajnimi fizik përfshin përzgjedhjen e indeksëve (metoda të qasjes), particionimin, dhe grumbullimin e të dhënave
- ✘ Metodologjia e dizajnit logjik e thjeshtëson qasjen në dizajnimin e databazave të mëdha duke e zvogluar numrin e mvarshmërive të të dhënave që duhen të analizohen
- ✘ Integriteti i të dhënave qëndron përmes normalizimit të tabelave të krijuara kur data modeli është transformuar në model relational
- ✘ Qëllimi i dizajnit fizik është optimizimi i performancës .
- ✘ Në këtë hap mund të kemi nevojë të bëjmë edhe denormalizimin

IMPLEMENTIMI, MONITORIMI DHE MODIFIKIMI

- ✘ Kur dizajni kompletohet databaza mund të krijohet duke përdorur DDL-në e nje DBMS.
- ✘ Pastaj DML mund të përdoret për të pyetur dhe azhuruar databazën, si dhe për te vendosur indexët dhe kufizimet, si p.sh. Integriteti referencial
- ✘ Gjuha SQL përmban DDL-në dhe DML-në (*create, select*)
- ✘ Kur databaza fillon punën, monitorimi tregon se a po arrihen kerkesat e performancës
- ✘ Nëse jo, atëherë duhet të bëhen modifikime.
- ✘ Modifikime behen edhe kur kerkesat për performancë ndryshojnë
- ✘ Cikli jetësor vazhdon me monitorin, ri-dizajnim dhe modifikim

PËRMBLEDHJE

- ✘ Njohja e teknikave për modelimin e të dhënave dhe dizajnimin e databazave është e rëndësishme për praktikuesit e databazave dhe zhvilluesit e aplikacioneve
- ✘ Cikli i jetës i databazës tregon hapat e nevojshëm për qasjen metodike në dizajnimin e databazës, prej dizajnit logjik, i cili është i pavarur nga mjedisi i sistemit, deri te dizajni fizik, që bazohet në detalet e DBMS së zgjedhur për implementimin e databazës.
- ✘ Në mesin e qasjeve të ndryshme të modelimit, modelet e të dhënave ER dhe UML janë më të përdorurat sot për shkak të thjeshtësisë dhe lexueshmërisë së tyre