

---

Databasesystemer, forår 2006  
IT Universitetet i København

**Forelæsning 3: E-R modellering**

16. februar 2006

Forelæser: Rasmus Pagh

---

# — Forelæsningen i dag —

---

- Datamodellering – hvad, hvorfor og hvordan?
- “Business rules”
- Grundliggende Entity-Relationship (E-R) modellering
- Mere om E-R modellering

# — Vigtige begreber fra tidligere forelæsninger —

- Datamodel
- Instans af relation
- Skema for en relation
- Database skema: Samlingen af relationskemaer i en database
- Attributter

# — Hvad er datamodellering? —

---

*Datamodellering* er en proces, der bruges til at skabe en datamodel for et givet problemområde

Når man modellerer data, der skal gemmes i en relationel database, skal et *slutprodukt* af datamodelleringen være et relationelt database skema.

Af hensyn til processen og til dokumentation af systemet, vil (god) datamodellering ofte indeholde adskillige andre aktiviteter end at opskrive relationskemaer:

- Definition af “business rules” .
- Tegning af E-R diagrammer.
- Opskrivning af supplerende datadefinitioner.

# — Hvorfor datamodellering? —

---

En god og korrekt repræsentation af data er af fundamental vigtighed for at:

- Databasen skal kunne understøtte de ønskede typer af operationer og forespørgsler.
- Databasen og associerede programmer let skal kunne tilpasses nye ønsker til funktionalitet, eller ændringer i problemområdet.

Kendetegn for en god konceptuel datamodel:

- Afspejler alle relevante aspekter af problemområdet præcist.
- Letforståelig og utvetydig, også med begrænset teknisk viden.
- Kan systematisk oversættes til en konkret logisk datamodel, der er udgangspunktet for implementation.

# — Database designmetodologi —

---

Denne forelæsning og de tre følgende dækker den dominerende metodologi for design af relationsdatabaser:

1. Identifikation (og dokumentation) af alle relevante *entiteter* og *relationer*, samt de “business rules” der gælder for dem.
2. Udformning af en grafisk repræsentation i E-R (eller EER eller UML) notation. (I dag og næste uge).
3. Konvertér modellen til et antal relationsskemaer (næste uge), og “constraints” (om to uger).
4. Eliminér eller reducer redundans ved at dele relationer. Denne proces kaldes **normalisering**. (Om tre uger.)

---

**“Business rules”**

---

## — Business rules —

---

MDM anbefaler, at man som en del af datamodelleringen opskriver alle relevante “business rules”, der gælder for problemområdet.

En *business rule* er et udsagn, der definerer eller begrænser et aspekt af det problemområde, vi modellerer.

Reglerne skal være præcise og utvetydige, og **forståelige for enhver, der forstår problemområdet**. De er således i høj grad et redskab til kommunikation mellem “business people” og databaseudviklere.

I princippet skal mængden af business rules beskrive *alle ikke-tekniske aspekter* af databasen. (Her regnes udformning af brugerflade, rapporter, etc., som tekniske aspekter).

## — Business rules – eksempler —

---

- *Enhver medarbejder skal have registreret præcis én lønkonto.*
- *For medarbejdere med samme funktion må lønnen højst variere med 20%.*
- *Til en medarbejder, der fylder 40, 50, eller 60 år i indeværende måned, udbetales et gratiale på 5000 kr.*
- *En medarbejder er en person, der har en gældende ansættelseskontrakt.*
- *En person er kendetegnet ved et unikt CPR nummer, et fornavn, et efternavn, eventuelt en adresse, og eventuelt et firmanavn.*

# — Kendetegn ved gode business rules —

MDM opskriver flg. kendetegn ved en god business rule:

- Declarative – what, not how
- Precise – clear, agreed-upon meaning
- Atomic – one statement
- Consistent – internally and externally
- Expressible – structured, natural language
- Distinct – non-redundant
- Business-oriented – understood by business people

---

## Grundliggende entity-relationship (E-R) modellering

---

# Hvad er en entity?

---

Ordet *entity* bruges i to betydninger:

- **Entity instance.** Et objekt, et sted, en hændelse eller et begreb.
- **Entity type.** En samling af entitetsinstanser. Forekommer i E-R diagrammer.

## Eksempler på entity instances:

“Danmark” (et sted), “Die Hard” (et abstrakt objekt), “Bruce Willis” (et fysisk objekt), “databaseekspert” (et begreb).

## Eksempler på entity types:

“LAND”, “FILM”, “SKUESPILLER”, “KVALIFIKATION”.

## — Attributter for entity types —

---

En entity type vil have tilknyttet et antal *attributter*, som er egenskaber eller karakteristika for den.

At bestemme hvilke attributter, der er relevante, er en del af datamodelleringen.

### **Eksempel:**

Entity typen SKUESPILLER kunne f.eks. tilknyttes attributterne *navn*, *adresse*, *køn*, og *fødselsdato*.

## — Sammenhænge mellem entity types —

Sammenhænge mellem entity types kaldes i E-R modellen for *relationships*.

Igen er relationship en samlebetegnelse for to begreber:

- **Relationship instance.** En sammenhæng eller kobling mellem to eller flere entitetsinstanser.
- **Relationship type.** En samling relationship instances, der forbinder to eller flere entitetstyper. Forekommer i E-R diagrammer.

**Eksempel på relationship instance:** (“Die Hard”, “Bruce Willis”)

**Eksempel på relationship type:** “spiller i” (forbinder entitetstyperne SKUESPILLER og FILM)

(Forelæsningen fortsætter ud fra slides, der følger med MDM.)

# Første del af gruppeprojektet

---

- Kan hentes på kursushjemmesiden – deadline 2/3.
- Udvikler jeres kompetence i E/R modellering.
- Regn med at lave fejl! I får chancen for at rette dem i næste del, efter feedback fra hjælpelæreren.
- Har I fundet jeres gruppemedlemmer?

# — **Vigtigste ting i denne forelæsning** —

- Business rules
- E-R begreber: Entity type, relationship type, stærke og svage entity types, attribut, grad af relationship, cardinality constraints.
- Grafisk repræsentation af E-R diagrammer.

## Næste gang

---

Modellering med Extended E-R (EER) notation:

- Subtype/supertype relationer (“nedarvning”).
- Clustering.

Hvordan man kommer fra en E-R model til relationskemaer.