INF115 Lecture 8: *Data Modelling with ER*

Adriaan Ludl
Department of Informatics
University of Bergen

Spring Semester 2021

INF115 Practical Aspects

Next lecture: Wednesday

24.02.2021 from 14:15 to 16:00

Mandatory assignment released this Friday (19.02.2021) !

Chapter 7: *Data modelling with ER (Part 2)*

Learning Goals:

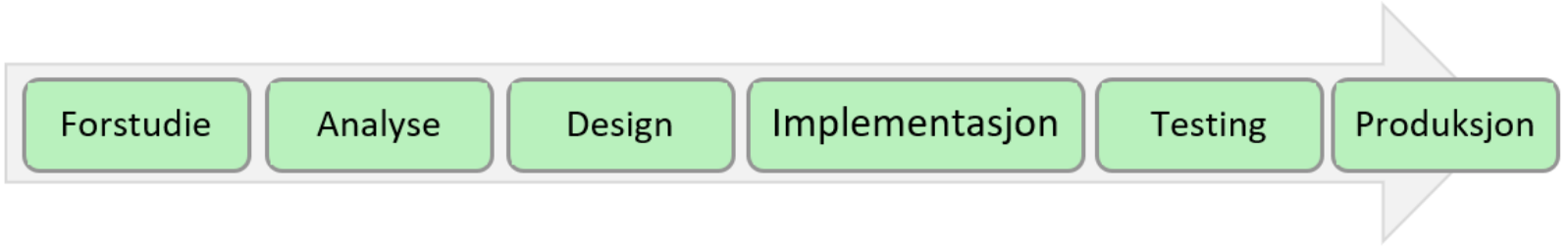
- Information System and Database **Development with ER** (Entity Relationship)
- Concrete, Abstract and Composite **Objects**
- Frequently Used **Modeling Patterns** and Strategies
- Modelling **Traps / Pitfalls**

Databases and Information Systems

- An **Information System** is a system that enables collecting, storing, using, transmitting, and distributing *information*. [Wikipedia]
  - Humans
  - Machines
  - Networks
  - Programs/applications
  - Manual routines
  - **Databases**
- A database is usually a part of a *larger* information system.

System Development

What?

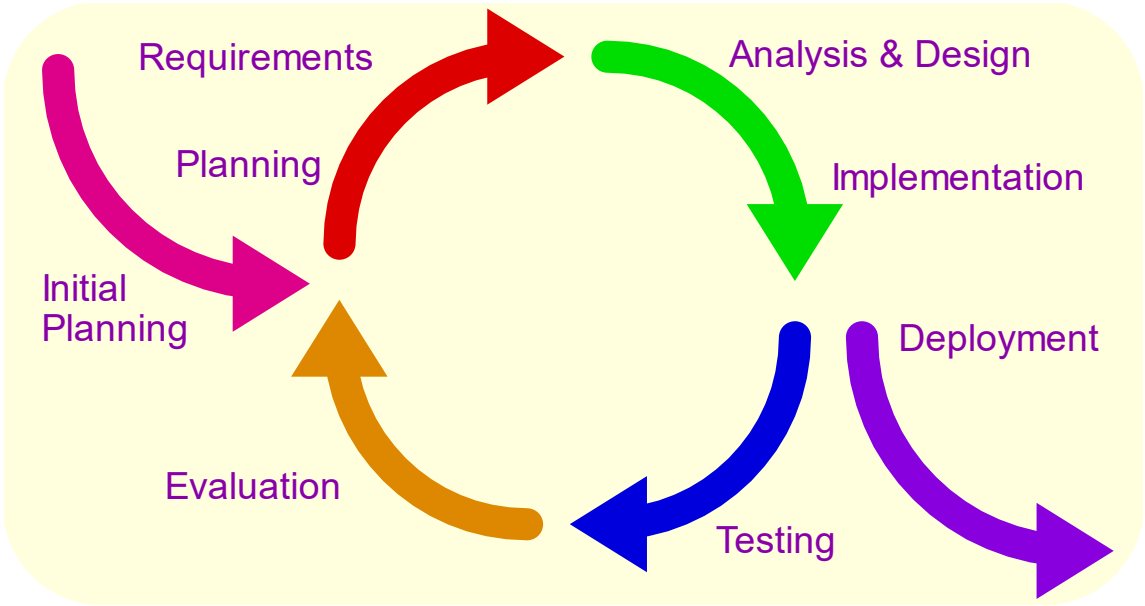


How?

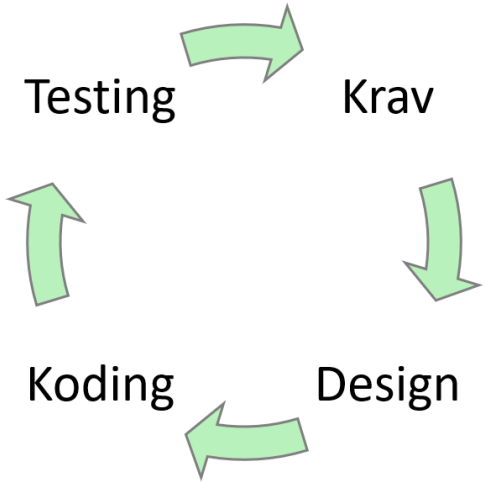
The development **process** that leads from *requirements (« what »)* to a *system (« how »)*.

A. Classical development methods use **phases**.

B. Newer methods are more ***agile*** *(smidig)* and ***iterative.***



[Wikipedia]



Database design from abstraction to implementation

What?

- **Conceptual** (begrepsmessig) **database design**
  - Describes the database in a way independent of given technology
  - **E/R-diagram**
- **Logical database design**
  - Logical table structure independent of a specific DBMS
  - **E/R-diagram with foreign keys** (Ch. 8)
- **Physical database design**
  - Physical table structure adapted to a given DMBS

How?

Generate Database

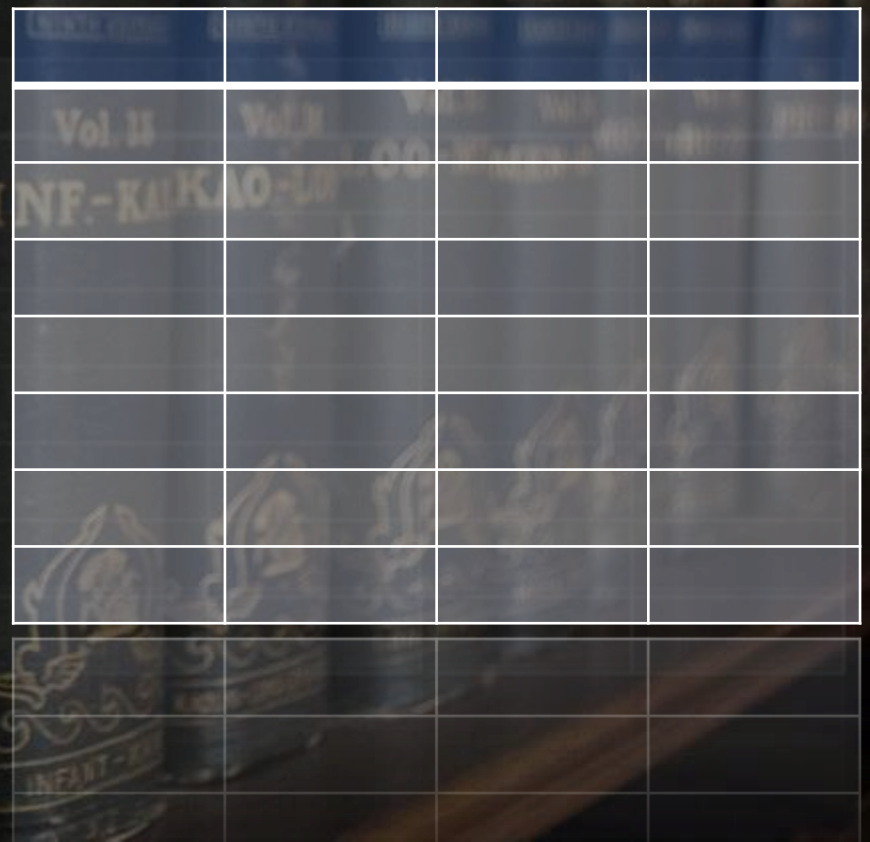
Models and Perspectives

- Useful to make a model before building something complicated:
  - Drawings,
  - 3D-models, …
- **Theoretical models** give a **simplified view of reality**. Simplification is justified to focus on essential traits (aspects) that have to be understood and at the same time neglect what is non-essential. [Wikipedia]
- **Perspectives**:
  - ***Process*** oriented: What will the system do ?
  - ***Information*** oriented: What will the system represent ?
  - ***Object*** oriented: A system is composed of «communicating objects».

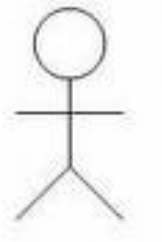
Representing Data in Tables

We have already seen examples of how **data can be represented in tables**.

We will now see more examples…



Physical objects (can usually be touched)



More abstract objects: *Loan (Lån)*

LNr	KNr	Beløp	Start	År	Rente	Type
27630364643	50401	1550000	02.09.2015	20	5.5	Serie
27630382732	50104	65000	01.01.2019	8	7.6	Annuitet
27630393982	20932	1050000	20.07.2012	20	5.0	Annuitet
27630338829	90237	300000	10.12.2017	10	6.5	Annuitet
27630391882	70238	1250000	05.08.2018	15	5.5	Serie

- A loan is not a physical object, but it is quite «*concrete*».
- Are there foreign keys here ?
- Store just what we need: We can **generate** the repayment schedule (nedbetalingsplan).

Composite objects: Ordre og Ordrelinje

OrdreNr	Ordredato	KNr
20505	20.8.2019	5022
20506	20.8.2019	5009
20507	20.9.2019	5188

An **order** is composed of a «header» …

… and several «**lines**».

OrdreNr	VNr	Pris	Antall
20505	10830	29.90	1
20505	77033	109.50	5
20506	10830	29.90	1
20506	44939	57.60	3
20506	65081	109.50	5
20507	12088	109.50	2

One-to-many relationship between an order and a line on the order.

Tables as Relationships: *ProsjektArbeid*

PNr	AnsNr	AntTimer
1001	42	12
1001	71	44
1003	123	4
1003	42	21
1003	2	76
1003	93	59
1005	2	40
1005	19	7
1007	27	23
1012	2	15
1012	42	42

Every row represents an **instance of the relationship.**

The table represents the **many-to-many relationship** between projects and employees.

Quiz on Data modelling with ER (part 3)

Please answer the practice quiz on mitt.uib now ☺

(you can take it again later if you want)

Link:

- https://mitt.uib.no/courses/27455/quizzes

Break !
Lecture resumes in 15 minutes

Events: Temperature measurements

StedNr	Dato	Temp
53	30.04.2019	7
53	12.05.2019	8
53	20.05.2019	6
54	20.05.2019	11
54	20.07.2019	4

An **event** is something

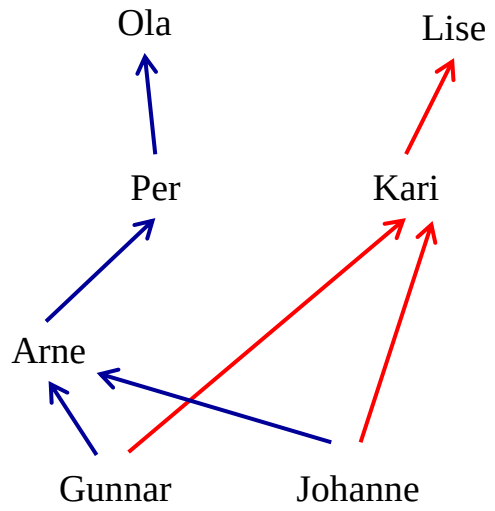
that **happens** at a given **place**

at a given **time**.

- What is an ***appropriate*** **primary key** in this table ?
- What if we want to make multiple measurements per day ?

Tables and hierarchies: Genealogical trees (Slektstrær)

Id	Fornavn	Mor	Far
1	Ola		
2	Lise		
3	Per		1
4	Kari	2	
5	Arne		3
6	Johanne	4	5
7	Gunnar	4	5



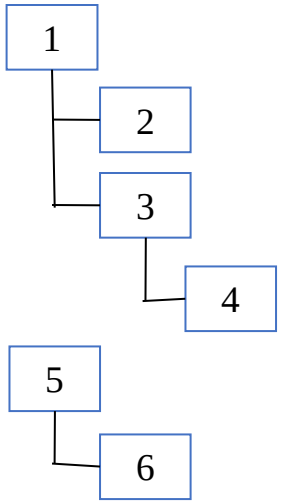
Both father and mother relationships are *one-to-many relationships*.

Table representation: Let every child «point to» their father/mother.

- Are there **foreign keys** here ?

Tables and hierarchies : Discussion forum

Id	Avsender	Dato	Melding	SvarPå
1	kahn	23.02.2019	Bla bla bla ...	
2	ok88	23.02.2019	Bla bla bla ...	1
3	jwh	24.02.2019	Bla bla bla ...	1
4	kahn	24.02.2019	Bla bla bla ...	3
5	kasper	24.02.2019	Bla bla bla ...	
6	jesper	25.02.2019	Bla bla bla ...	5

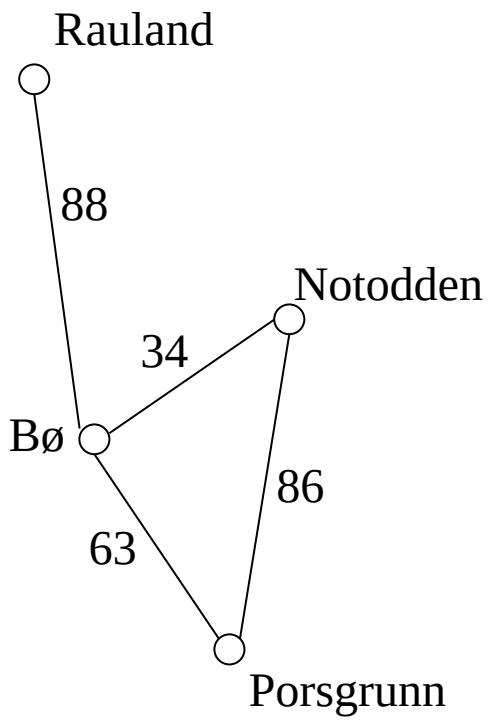


- A forum has many discussion threads.
- Every thread is a **hierarchy** (a top-down tree).
  - A post can have many answers.
  - A post can start a new thread, or answer a previous post (innlegg).

➢ Are there **foreign keys** in this table ?

Tables and graphs (networks): Distance Matrix

Fra	Til	Km
Bø	Porsgrunn	63
Porsgrunn	Notodden	86
Bø	Notodden	34
Bø	Rauland	88



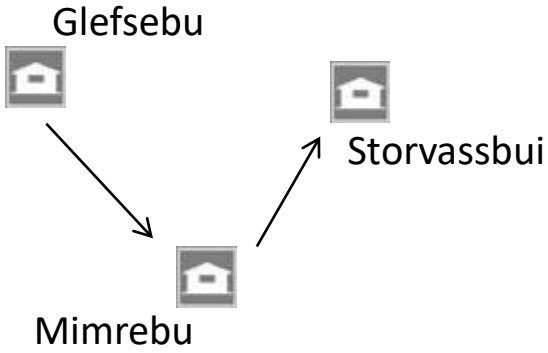
- A ***graph*** is a collection of **nodes** (cities) and **edges** (paths).

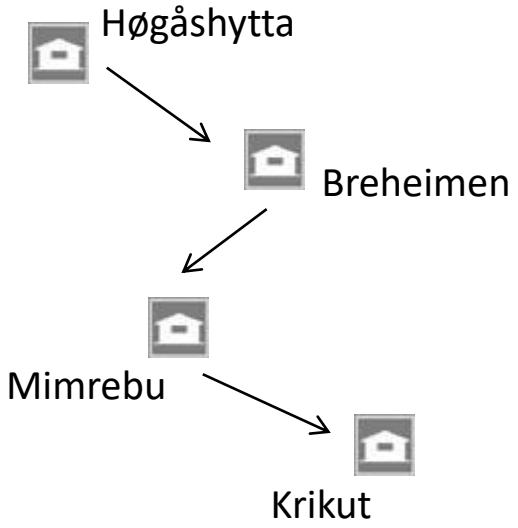
- Every edge is a row in the table.
- Nodes (cities) can be stored in a separate table.

➢ Store paths in both directions separately ?

Tables and graphs (networks): Hike (fottur)

TurNr	StartHytte	Gradering
56	Glefsebu	Lett
77	Høgåshytta	Middels





TurNr	Dag	TilHytte
56	1	Mimrebu
56	2	Storvassbui
77	1	Breheimen
77	2	Mimrebu
77	3	Krikut

Modelling Patterns / Strategies

There are useful data **modelling patterns** or **strategies** can be applied to many typical problems (**modelleringsmønstre** eller **modelleringsklisjee)** [Skagestein].

Examples:

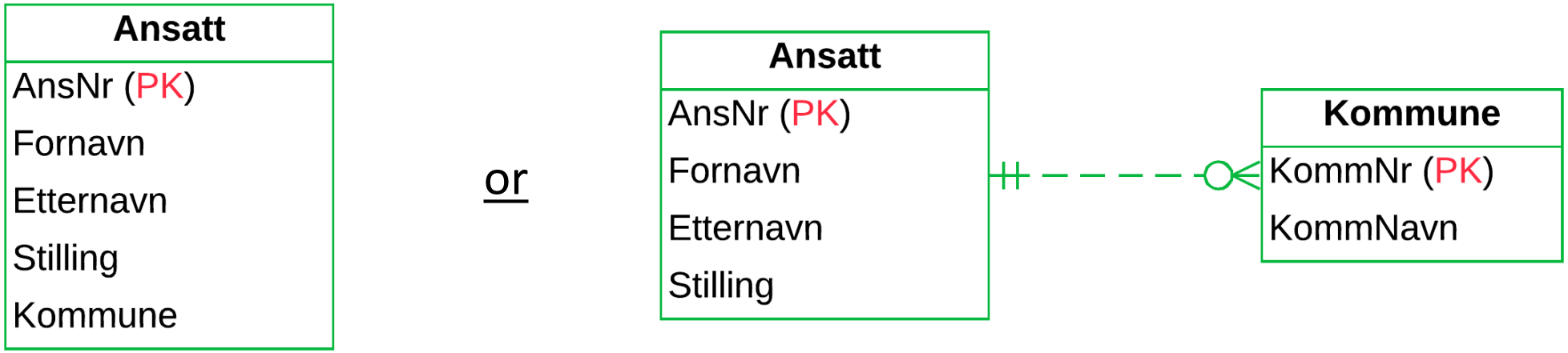
- «Is-an» hierarchies (a cat is an animal with four legs …)
- «Has-an» hierarchies (a product ist built from …)

- A paper form is built from a header and lines …

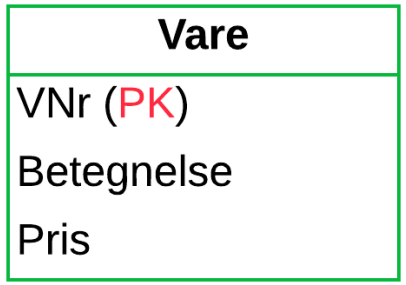
- Should we store only data on the current situation?
- Or also historical data ?

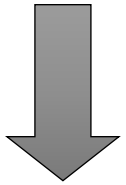
Entity or attribute?

- It is an important question to consider whether a **quantity** shall be an *entity* or an *attribute of another entity*.
- If we only store the name of a **municipality** (***kommune***), then it could be an attribute in *Ansatt*.
- However, *if we need to store more information about the municipality*, then we should introduce it as *an actual entity*.

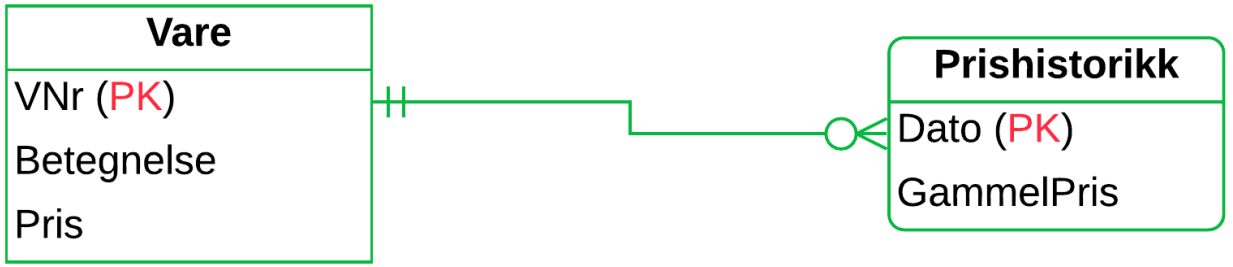


Current or historical data ?





- Only store current price ?
- What did the product cost before ?
  - Store price changes or daily values ?
  - Work on time periods.
- *Time stamp* as (partial) identifier (PK) in the ***entity*** that represents **events**.



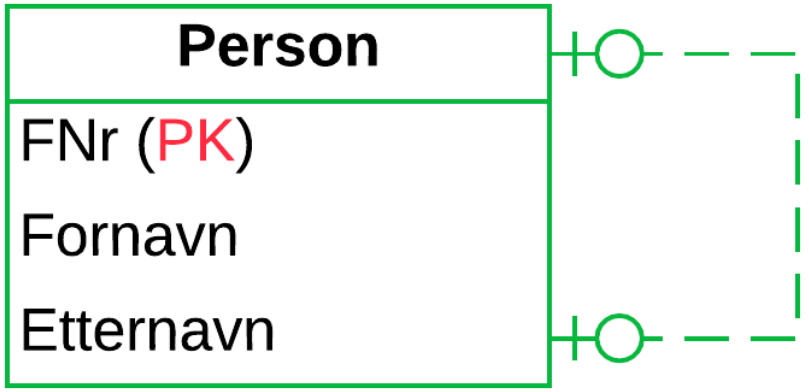
Header/line pattern (master/detail)

Many **registration forms** and **reports** are structured in two parts:

- A **header** and a number of **lines**.
- Examples:
  - An **order overview** has the order date and the client number in the header, and in addition lines each of which gives the product number and the number of units odered.
  - A **time sheet** has the employee number and time period in the header, and then a number of lines that specify the project number and the number of hours worked on the project.
  - A **transcript of** records (vitnemål) has the student number, start and end year in the header, and then a number of lines each with course code, exam date and grade.

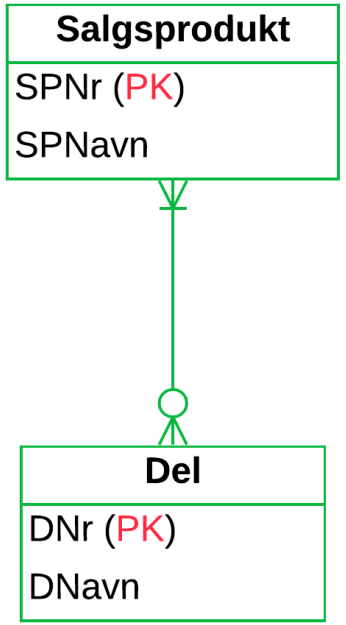
Self referencing relationship

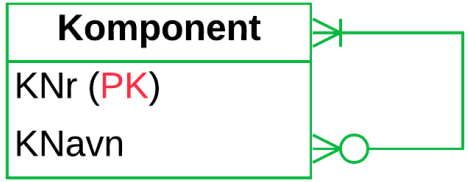
- It is entirely possible to make relationships that involve only one entity type. These are called **self-relationships** **(egenforhold).**
- For example reltationships between humans – friends, partners, spouses, teammates, colleagues, boss, etc ...



Hierarchies

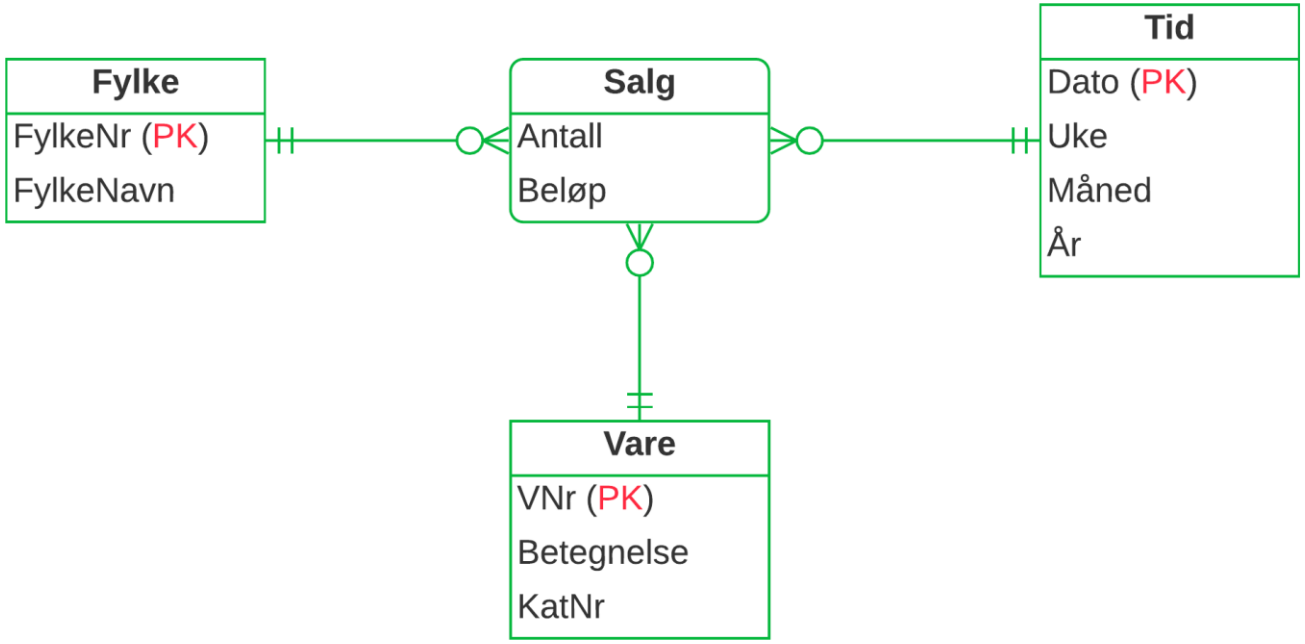
- Self referencing relationships can be used to model arbitrarily **deep hierarchies**: genealogical trees, catalogs/maps, discussion forum …
- Can the number of levels vary ? Upper limit ?
- Which model gives the simplest queries ?





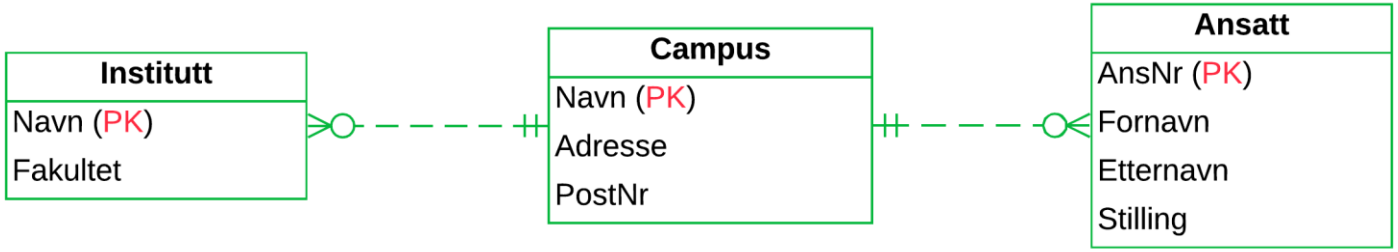
Star schema

- A **data warehouse** contains aggregated data from one or several databases and potentially other data sources. The **data warehouse** is used for **strategic decisions**.
- Data models for a **data warehouse** are often based on a ***central* fact table** joined to multiple **dimension tables,** where time is often one of the dimensions.



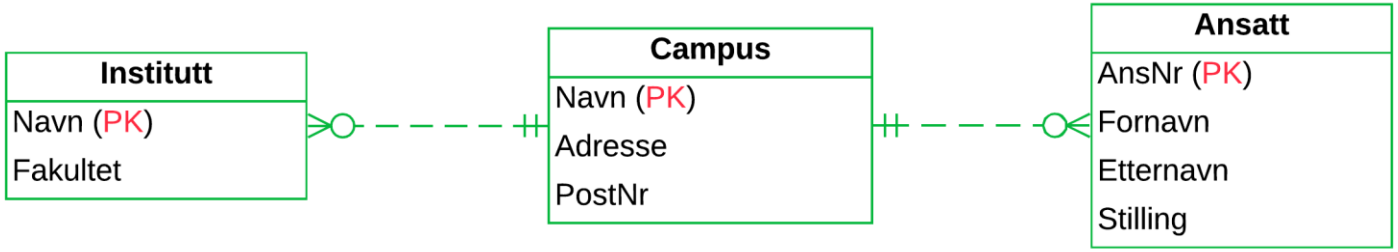
Modelling traps/pitfalls: fan trap

- An incomplete data model: we did not get a relationship between *Institute* and *Employee*.
- From a given institute we find the corresponding campus, but from there we have a ***branching*** (vifte = fan) to *Ansatt*.

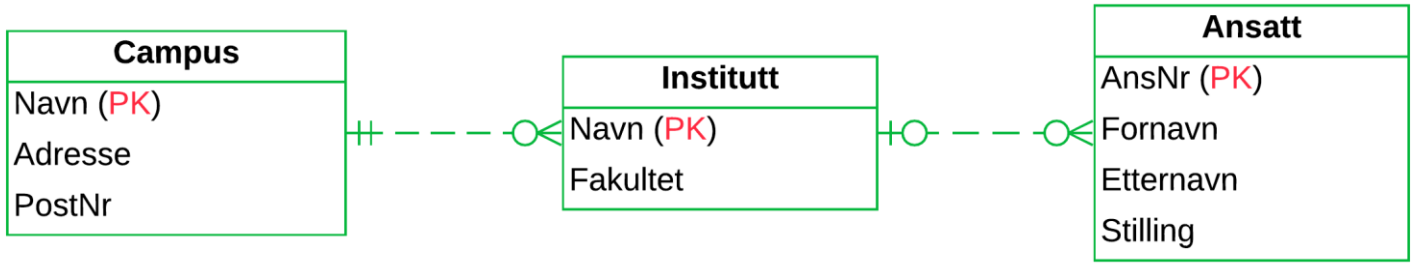


Modelling traps/pitfalls: fan trap

- An incomplete data model: we did not get a relationship between *Institute* and *Employee*.
- From a given institute we find the corresponding campus, but from there we have a ***branching*** (vifte = fan) to *Ansatt*.

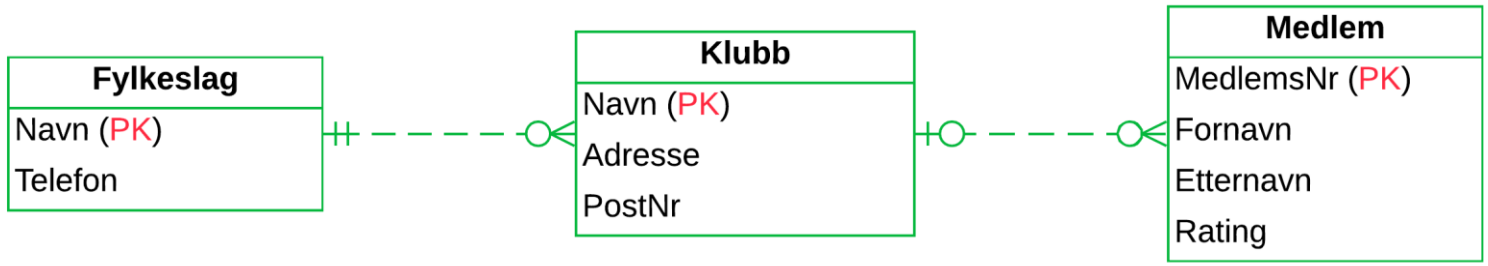


- **Solution**:



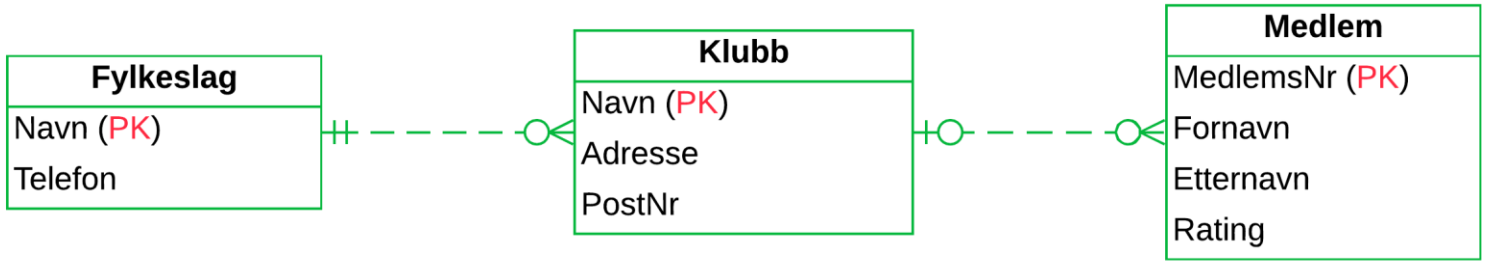
Modelling traps: chasm trap

- An incomplete data model: we did not specify to which county club (fylkeslag) the clubless members belong.
- The model has «0 or 1» for the relationship between Club and Member (Medlem), this gives a **chasm** (kløft) in Local Club.

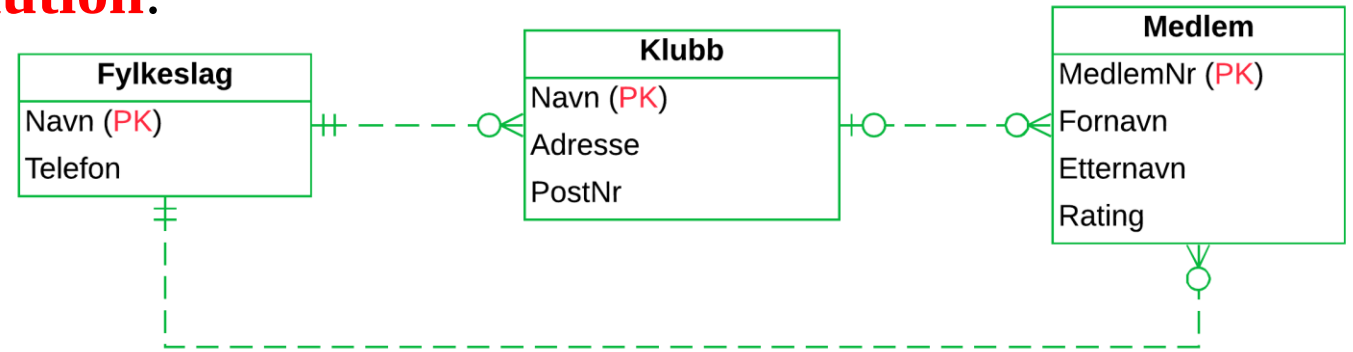


Modelling traps: chasm trap

- An incomplete data model: we did not specify to which county club (fylkeslag) the clubless members belong.
- The model has «0 or 1» for the relationship between Club and Member (Medlem), this gives a **chasm** (kløft) in Local Club.



- **Solution**:



Quizz on Data modelling with ER (part 4)

Please answer the practice quizz on mitt.uib now ☺

(you can take it again later if you want)

Link:

- https://mitt.uib.no/courses/27455/quizzes

Chapter 7: *Data Modelling with ER*

Summary of part 1:

- Goals of data modelling and database design.
- **ER** and **UML** are **visual modelling languages**.
- **Entities** with attributes and identifiers (PK).
- One-to-one, one-to-many and many-to-many **relationships**.
- **Weak** entities and **identifying** relationships.
- **Common abstraction methods.**

Chapter 7: *Data modelling with ER (Part 2)*

Summary of ER part 2:

- Information System and Database **Development with ER** (Entity Relationship)
- Concrete, Abstract and Composite **Objects**
- Frequently Used **Modeling Patterns** and Strategies
- Modelling **Traps**

INF115 Practical Aspects

Next lecture: Wednesday

24.02.2021 from 14:15 to 16:00

Mandatory assignment released this Friday (19.02.2021) !