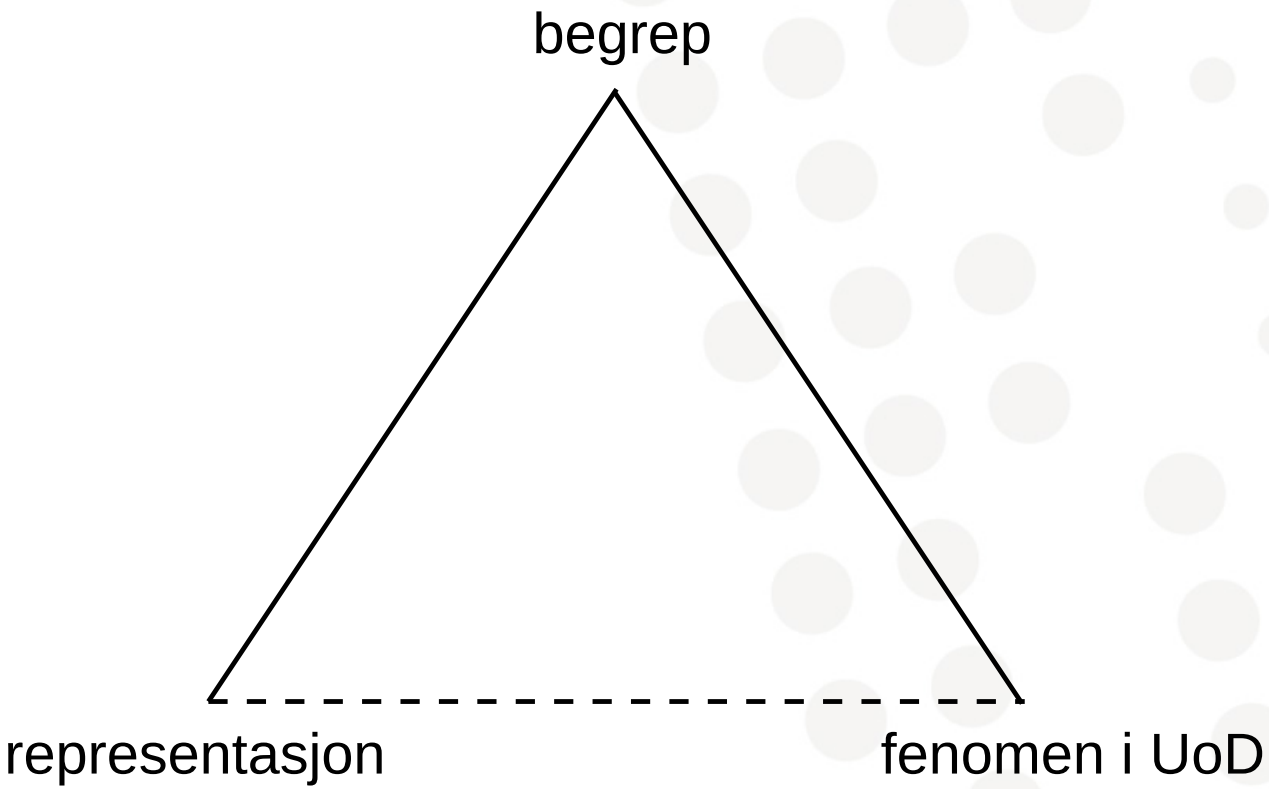
INF1300

Grunnbegrepene i ORM: fakta, begreper, roller, faktatyper, broer, entydighetsskranker, totale roller, funksjonelle avhengigheter

UNIVERSITETET I OSLO

Ogdens trekant



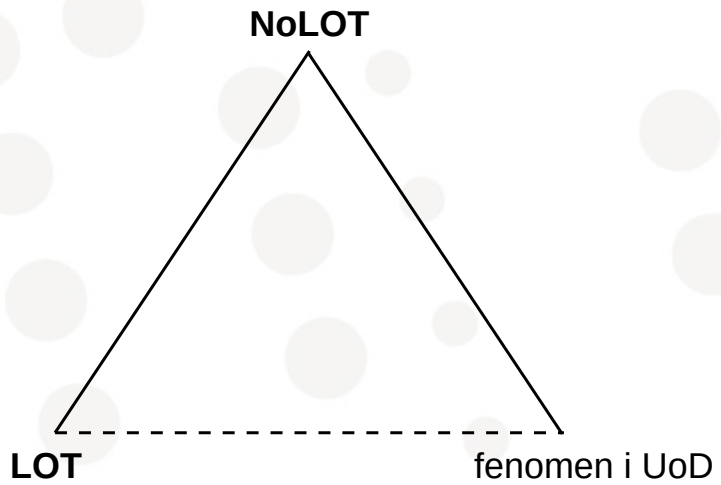
Ogdens trekant og ORM

- I ORM tegner vi begreper og representasjoner som hhv. heltrukne og stiplede rektangler med runde hjørner
- Eksempel:
  - **Begrepet** Person og **representasjonen** Fødselsnummer tegnes slik:

- I ORM brukes ordet **objekttyper** som en felles betegnelse på begreper og representasjoner

ORM vs stORM

- Verktøyet stORM bruker en gammel ORM-dialekt (NIAM–Natural language Information Analysis Method)
- I stORM brukes sirkler i stedet for rektangler
- Et *begrep* kalles en **NoLOT** (Non Lexical Object Type)
- En *representasjon* kalles en **LOT** (Lexical Object Type)



Setningers aritet - 1

- Den elementære setningen
  - *Studenten med navn Anne fikk i emnet med emnekode INF1010 resultatet karakteren B*

  inneholder tre begreper: *student*, *emne* og *resultat*
- Antall begreper i en setning kalles *setningens aritet*
- Vårt eksempel har aritet 3

Setningers aritet - 2

- Setninger med aritet 1 kaller vi unære
- Setninger med aritet 2 kaller vi binære
- Setninger med aritet 3 kaller vi ternære
- Man kan konstruere elementære setninger med vilkårlig høy aritet
- Elementære setninger med aritet > 3 er sjeldne, så vi gir dem ikke egne navn (n-ære setninger)

Unære setninger

- Disse er vanlige i dagligtalen, men sjeldne i en informasjonsmodell
- Eksempel: *En person med navn Siri sover*
- En unær setning kan alltid erstattes av en binær setning hvor ett av begrepene har boolsk representasjon
- Eksempel: *En person med navn Siri har sovestatus med boolsk verdi* true
- Unære setninger er tillatt i ORM, men ikke i modelleringsverktøyet stORM

Binære setninger

- Disse er vanlige i dagligtalen og enda vanligere i en informasjonsmodell
- Eksempel: *En person med navn Siri eier en bil med reg.nr. DL12345*
- Binære (og unære) setninger er alltid elementære
- En ORM informasjonsmodell baserer seg nesten i sin helhet på binære setninger
- Vi skal neste uke se hvordan vi med et knep kan uttrykke elementære setninger med aritet n > 2 ved hjelp av binære setninger

Setning med aritet 4

Eksempel på en elementær setning med aritet 4:

- *31.8.2011 lånte Per kr 200 000 av Skandiabanken*

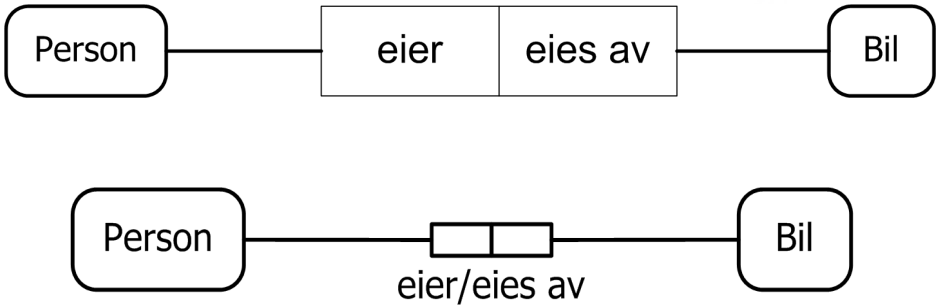
Fakta og setningers dype struktur

- Setningen: *En person med navn Siri eier en bil med reg.nr DL12345* kan også leses baklengs:
- *En bil med reg.nr DL12345 eies av en person med navn Siri*
- De to setningene har samme meningsinnhold
- Lingvistene kaller dette meningsinnholdet for setningenes dype struktur
- Vi informatikere kaller et slikt setningspar *et faktum*, og vi sier at ORM-metoden er *faktaorientert informasjonsmodellering*

Roller og faktatyper

- Se på setningsparet: *En person med navn Siri eier en bil med reg.nr DL12345* og *En bil med reg.nr DL12345 eies av en person med navn Siri*
- Her kan vi åpenbart få lignende fakta ved å bytte ut forekomsten *Siri* med et annet navn og/eller forekomsten *DL12345* med et annet reg.nr.
- Vi sier at begrepet Person spiller rollen eier overfor begrepet Bil, og at Bil spiller rollen eies av overfor Person
- Et slikt rollepar mellom to begreper kalles **en** (binær) **faktatype**

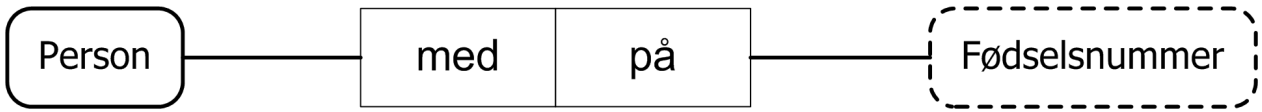
Faktatyper i ORM

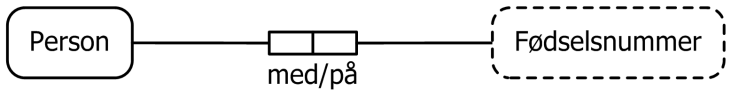


- Vårt eksempel på en faktatype mellom begrepene Person og Bil tegnes slik i ORM1 øverst, en mer kompakt skrivemåte ORM2 (brukes i læreboka) nederst. (Microsofts Visio har maler for en variant av ORM1)
- De to tegnemåtene er ekvivalente og kan brukes om hverandre

Broer

- En **bro** er en forbindelse mellom et begrep og en representasjon
- Her er de to måtene vi har i ORM for å tegne broen mellom Person og Fødselsnummer:





Setningstyper

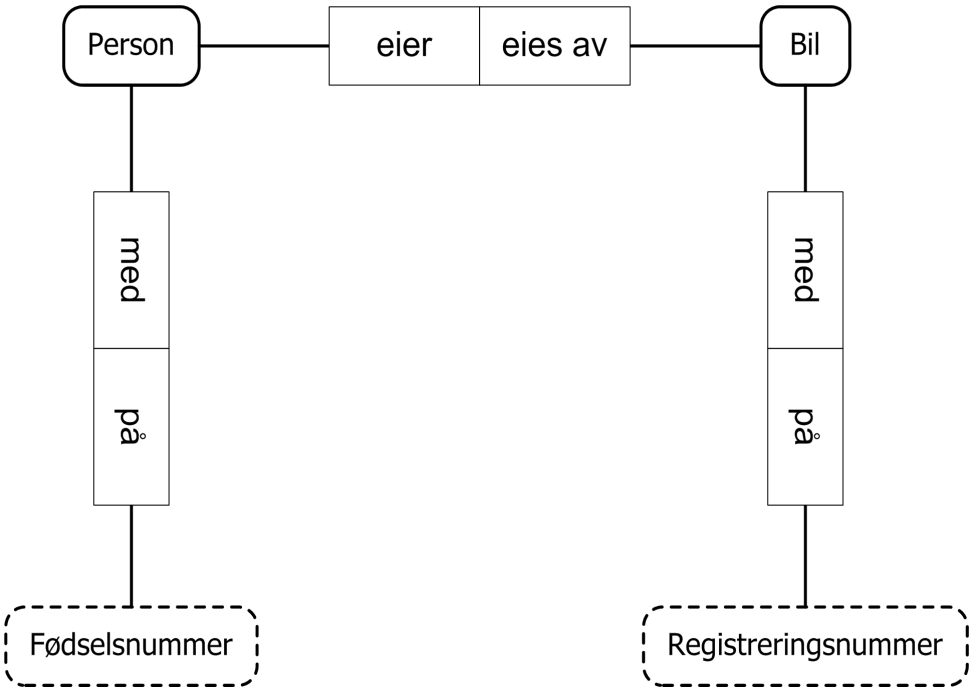
- Ordet **setningstype** er en felles betegnelse på faktatyper og broer
- Broer er alltid binære – de forbinder ett begrep og én representasjon
- Faktatyper kan ha vilkårlig antall roller (aritet)
  - hver rolle skal være knyttet til eksakt ett begrep
  - et begrep kan spille flere roller i samme faktatype

Rollenavn

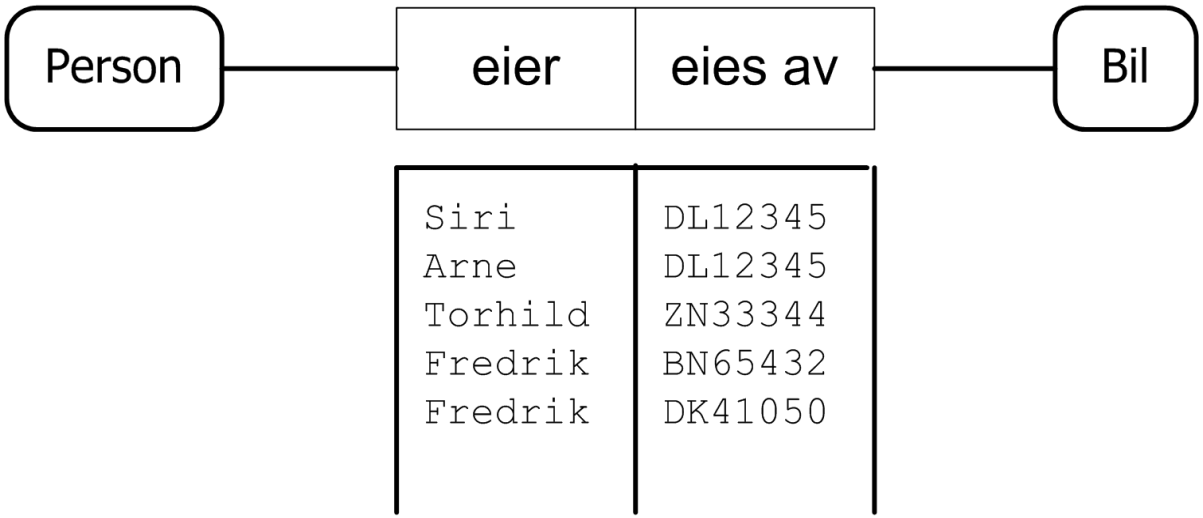
- I faktatyper bør alle rollenavn inneholde et verb (Hvis ikke, er det grunn til å tro at rollenavnet er dårlig valgt)
- I broer er det vanlig med preposisjoner som rollenavn
- De to vanligste rolleparene er
  - med/for
  - med/på

Fakta—setningers dype struktur

Dette diagrammet kan leses begge veier:

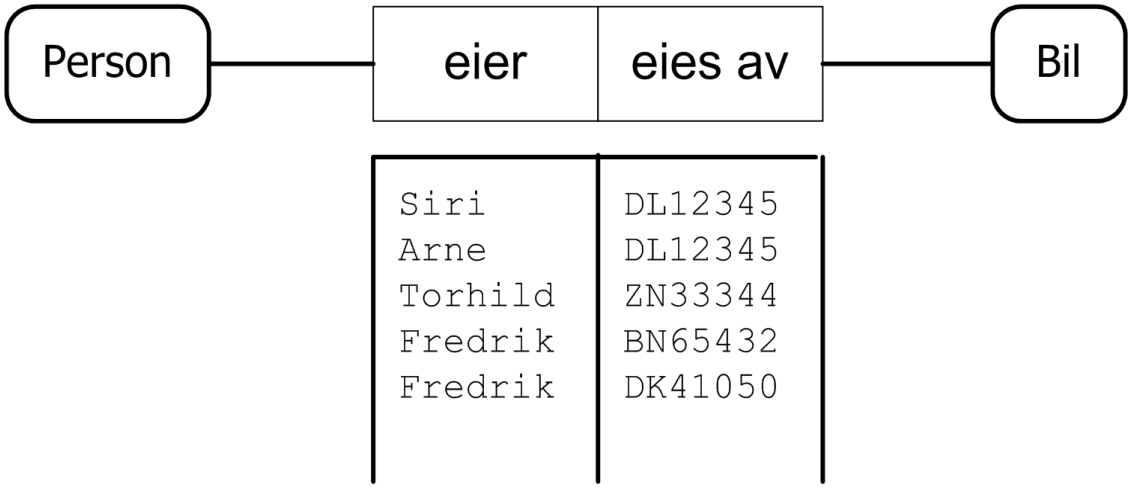


Faktatype med forekomsttabell



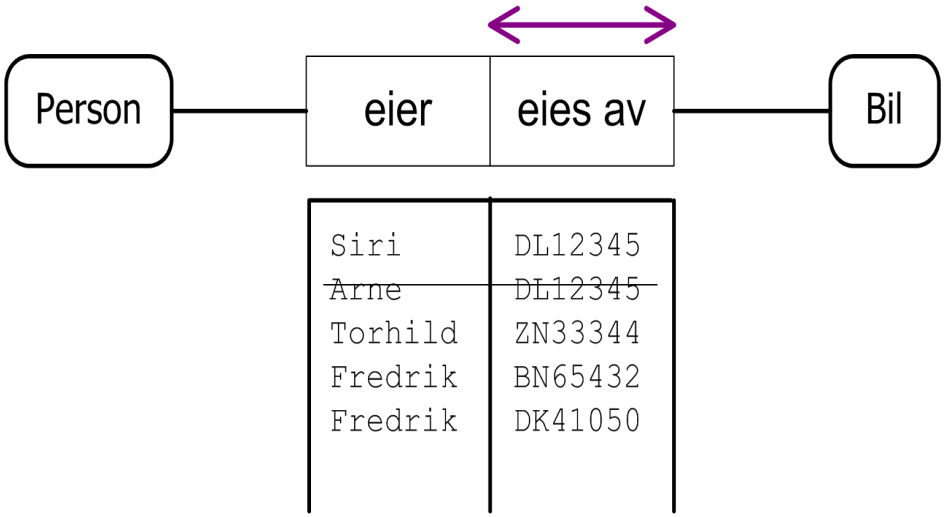
- Legg merke til at en faktatype med forekomsttabell gir oss like mange fakta som det er linjer i forekomsttabellen

Entydighetsskranker - 1



- Anta at vi har en forretningsregel som sier at en bil bare kan ha én eier (mens en person kan eie flere biler)
- Da kan ikke både Siri og Arne være eier av DL12345

Entydighetsskranker - 2



- I ORM-diagrammet setter vi en pil med spiss i begge ender over rollen hvor samme forekomst ikke kan gjentas i forekomsttabellen
- Pilen kalles en **entydighetsskranke**
- Entydighetsskranker kan gå over flere roller—da er det forekomstkombinasjonen i rollene som ikke kan gjentas

Bruk av forekomstdiagrammer

- Forekomstdiagrammer er ikke en del av ORMmodellen, men et nyttig, og ofte nødvendig, hjelpemiddel for å få riktige skranker i modellen
- Derfor er det ikke så nøye hvilken representasjon vi bruker i forekomsttabellene
- Eksempelvis kan vi ha brukt et ansattnummer eller fødselsnummer for å representere personer, mens vi i forekomsttabellene bruker navn
- I så fall antar vi for anledningen at alle personer har forskjellige navn

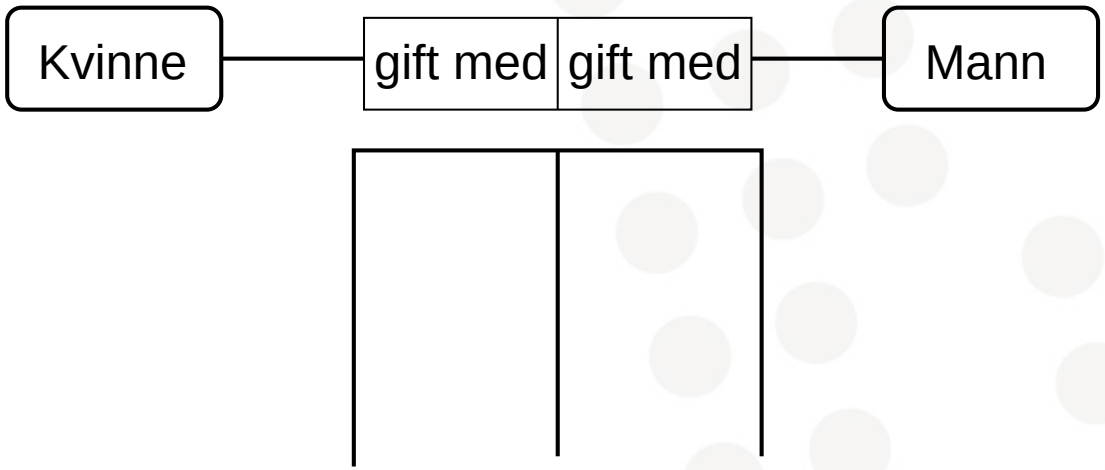
Ekteskap - 1

- Vi skal nå se på en faktatype mellom en kvinne og en mann kalt ekteskap
- Uten entydighetsskranke(r) ser modellen slik ut:

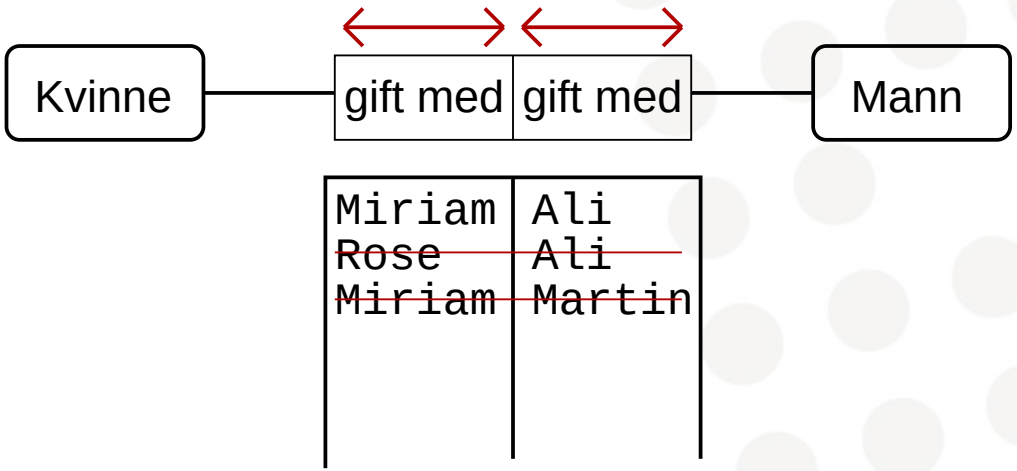
- Hvilke(n) entydighetsskranke(r) skal vi ha?
- Lag forekomsttabell og sett på entydighetsskranke(r)!

Ekteskap - 2

- De fleste har vel foreslått modellen nedenfor
- Lag forekomsttabell



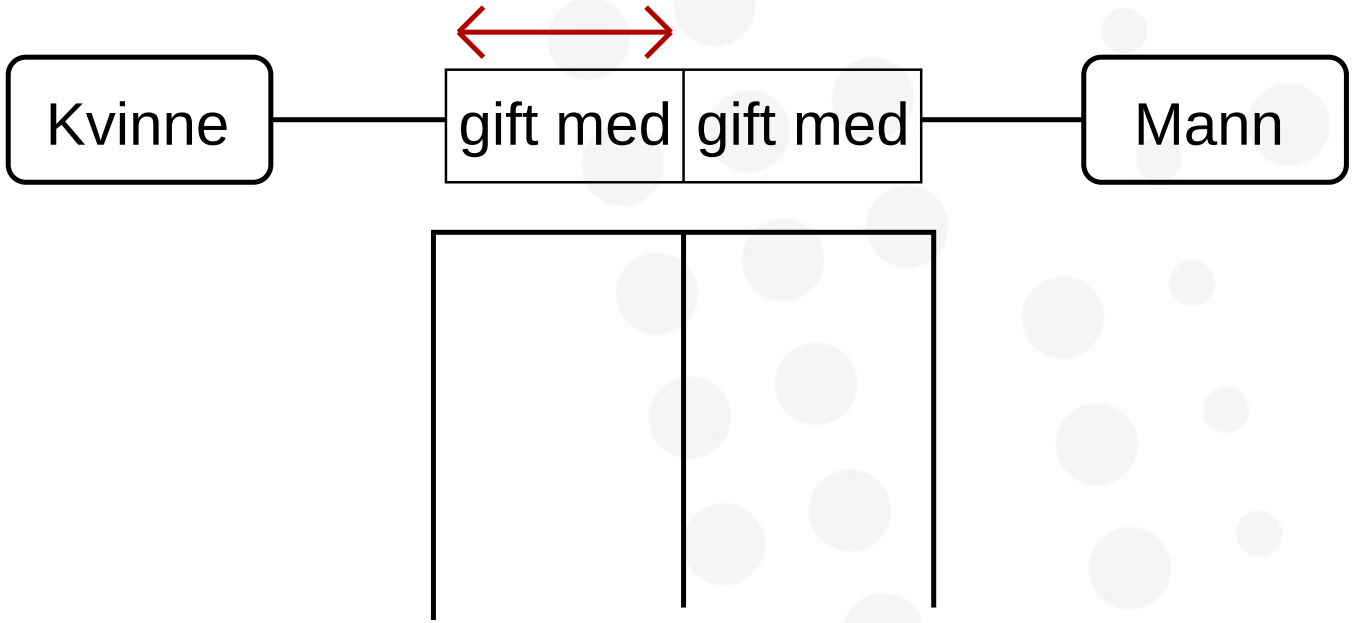
Monogami



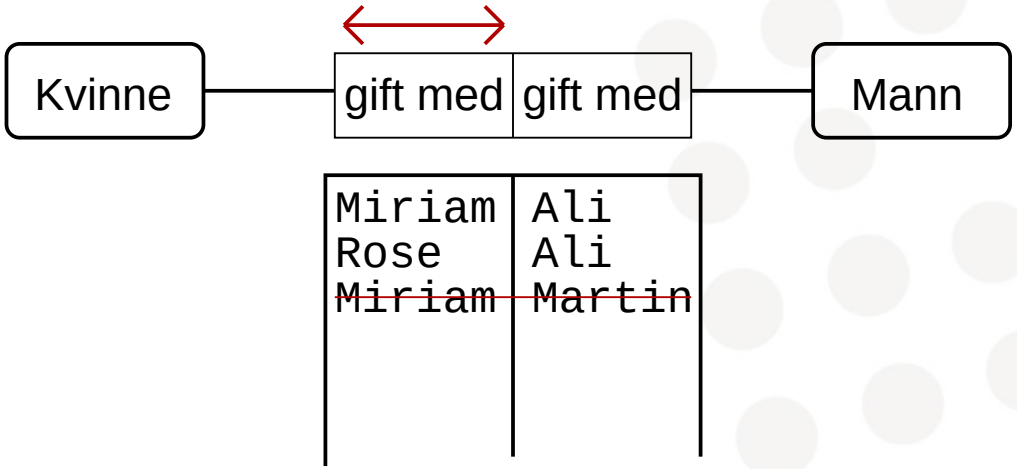
- I Norge er dette eneste lovlige ekteskapsform
- Vi kaller den en 1:1 (én-til-én) faktatype mellom (begrepene) Kvinne og Mann (idag mellom Person og Person)

Ekteskap - 3

- En annen mulighet er nedenstående modell
- Vi lager forekomsttabell:



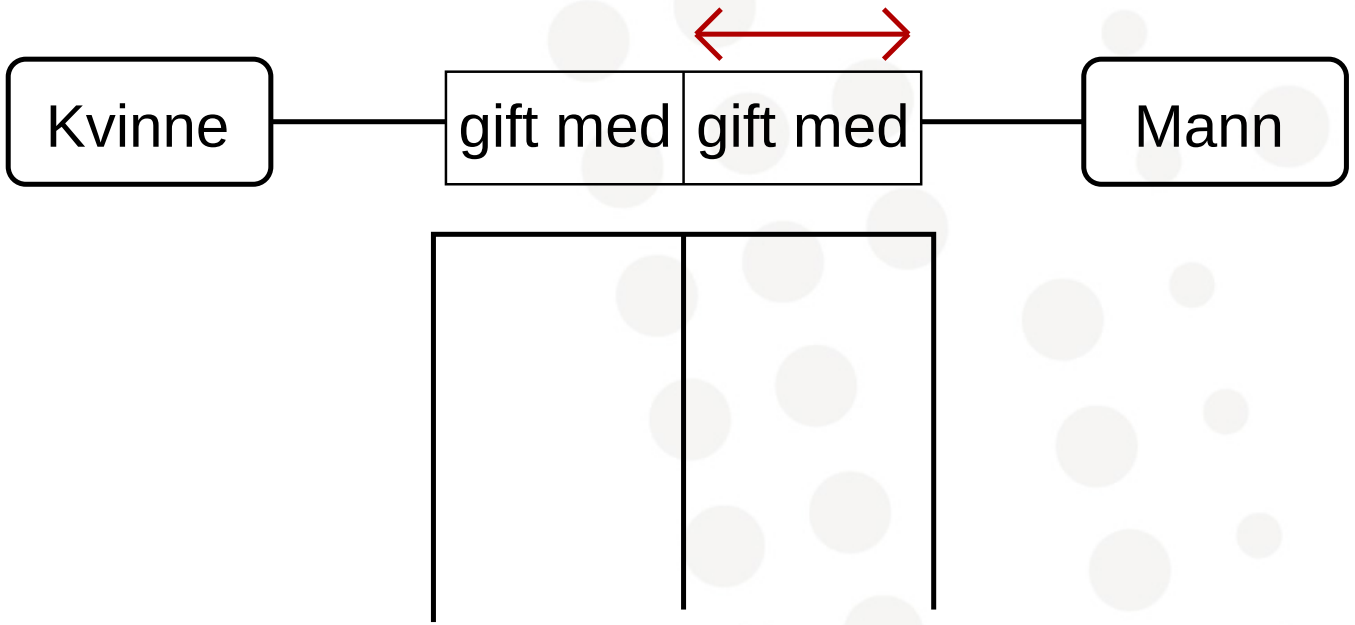
Polygyni



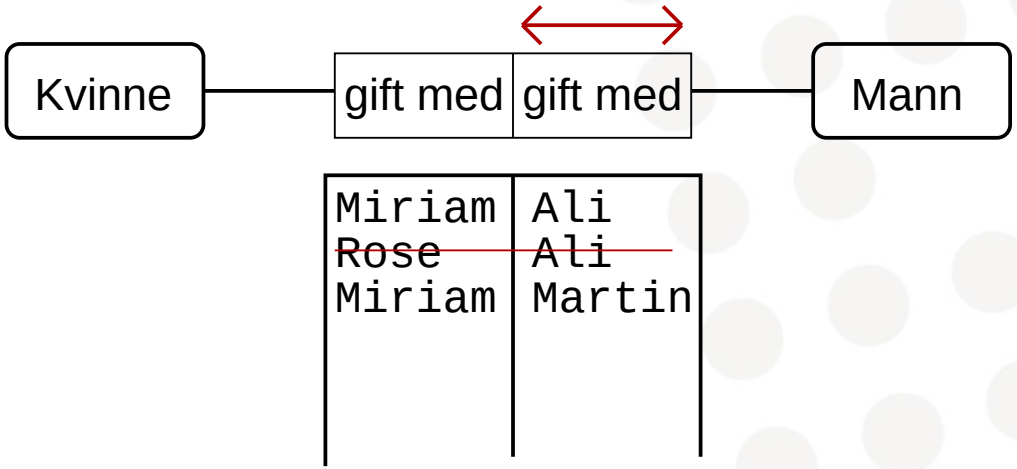
- Polygyni (flerkoneri) er ikke en uvanlig ekteskapsform
- Dette er en n:1 (mange-til-én) faktatype fra Kvinne til Mann

Ekteskap - 4

- En tredje mulighet er nedenstående modell
- Vi lager forekomsttabell:



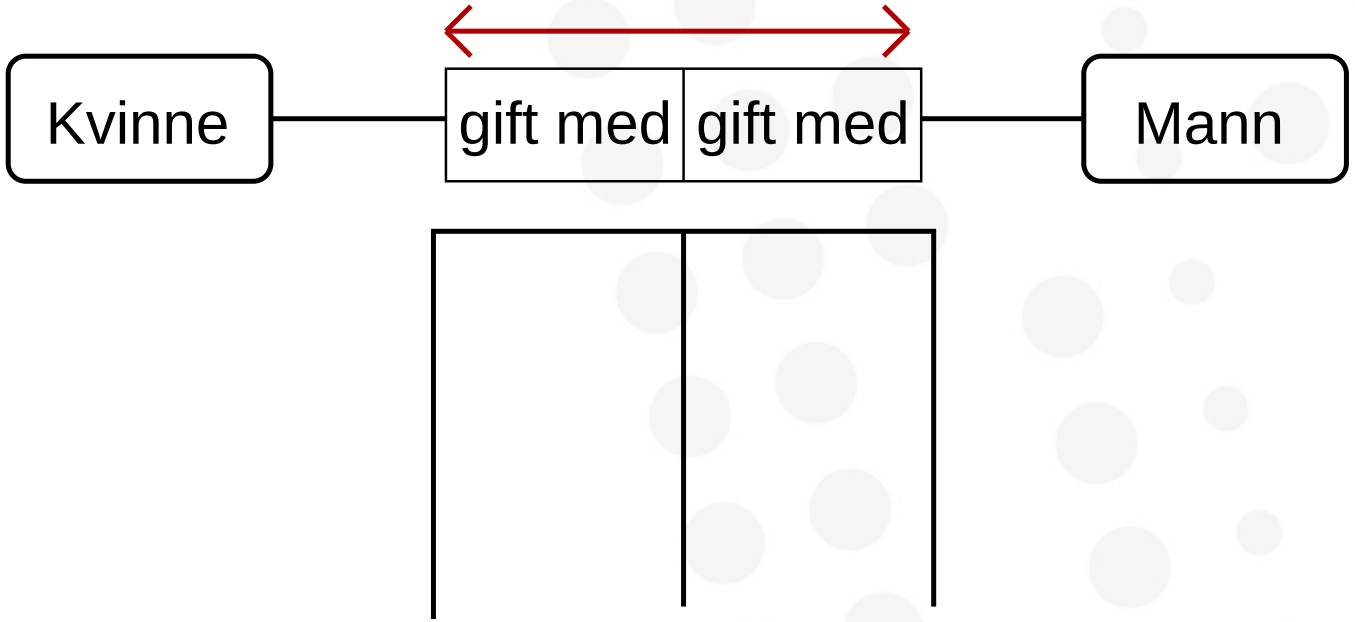
Polyandri



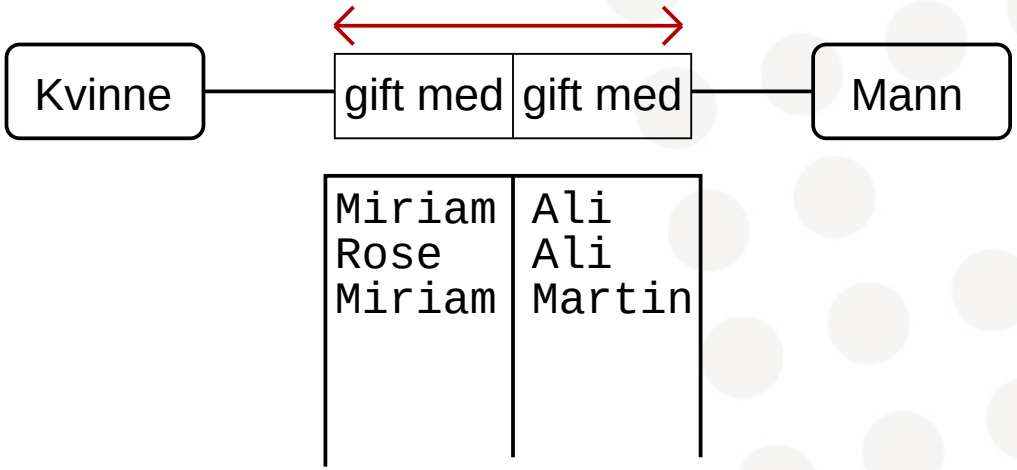
- Polyandri (flermanneri) var nok vanligere før, men forekommer fortsatt, blant annet i Polynesia
- Dette er en 1:n (én-til-mange) faktatype fra Kvinne til Mann

Ekteskap - 5

- En siste mulighet er nedenstående modell
- Vi lager forekomsttabell:

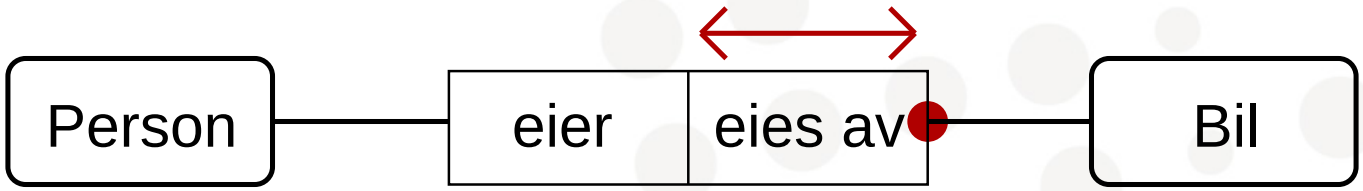


Polygami



- Ekte polygami (flergifte) har vel aldri vært en sosialt anerkjent samlivsform. Unntaket måtte kanskje være hippietidens kollektiver.
- Dette er en m:n (mange-til-mange) faktatype

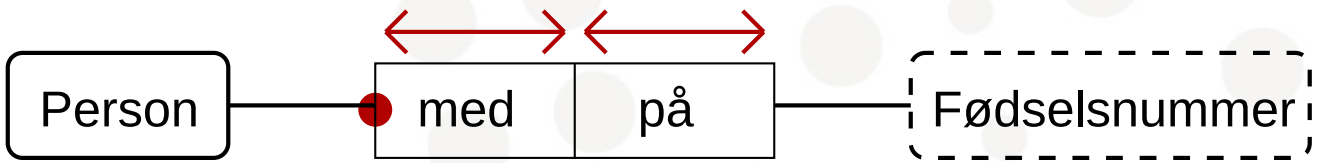
Totale roller



- Dersom alle biler har en eier, sier matematikerne at vi har en total funksjon fra Bil til Person (den er definert for alle forekomster av Bil)
- Vi sier at rollen eies av er en **total rolle** for Bil og markerer det med en kule (liten fylt sirkel) på rollen
- **Merk**: Det at rollen er total, gjør at hver gang vi legger inn en bil i databasen, må vi *samtidig* registrere hvem som eier bilen

Perfekt bro - 1

- En 1:1 bro der begrepsrollen er total, kalles en **perfekt bro**

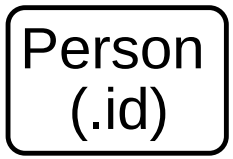


- Perfekte broer er så vanlige at vi har en egen kortform for dem (de implisitte rollenavnene er med/på eller med/for (with/of)):

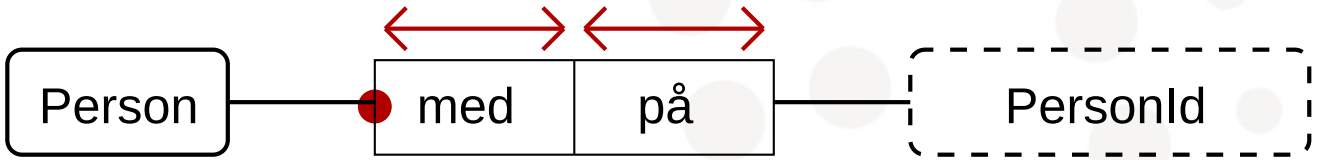
De to tegnemåtene er ekvivalente

Perfekt bro - 2

- Hvis vi har en perfekt bro hvor navnet på representasjonen er lik begrepsnavnet med et suffiks, har vi en enda mer kompakt notasjon:



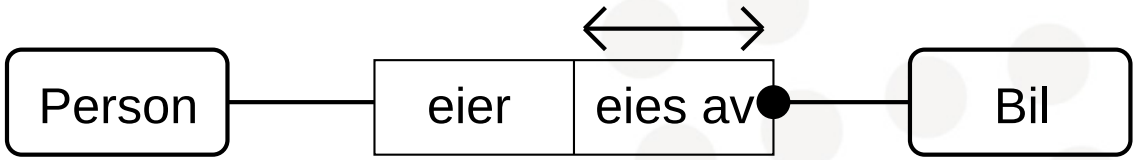
- som er en kortform for



Mer om entydighetsskranker

- Merk at en faktatype (det gjelder også broer) alltid skal ha (minst) en entydighetsskranke
- Hvis ikke, kunne samme faktum bli lagret vilkårlig mange ganger
- Merk også at en kort entydighetsskranke er strengere enn en lang
- Det er feil å la en lang entydighetsskranke dekke en kort

Funksjonelle avhengigheter



- Det at en bil alltid har én, og bare én, eier, gjør at hvis vi vet hvilken bil vi har (dvs. at vi kjenner bilens reg.nr), så vet vi også hvilken person som eier bilen
- Faktatypen fra Bil til Person definerer altså en *funksjon* fra forekomstene av Bil til forekomstene av Person som til en gitt bil gir oss eieren
- Vi sier at Person er *funksjonelt avhengig* av Bil, eller at vi har en FD (Functional Dependency) fra Bil til Person