

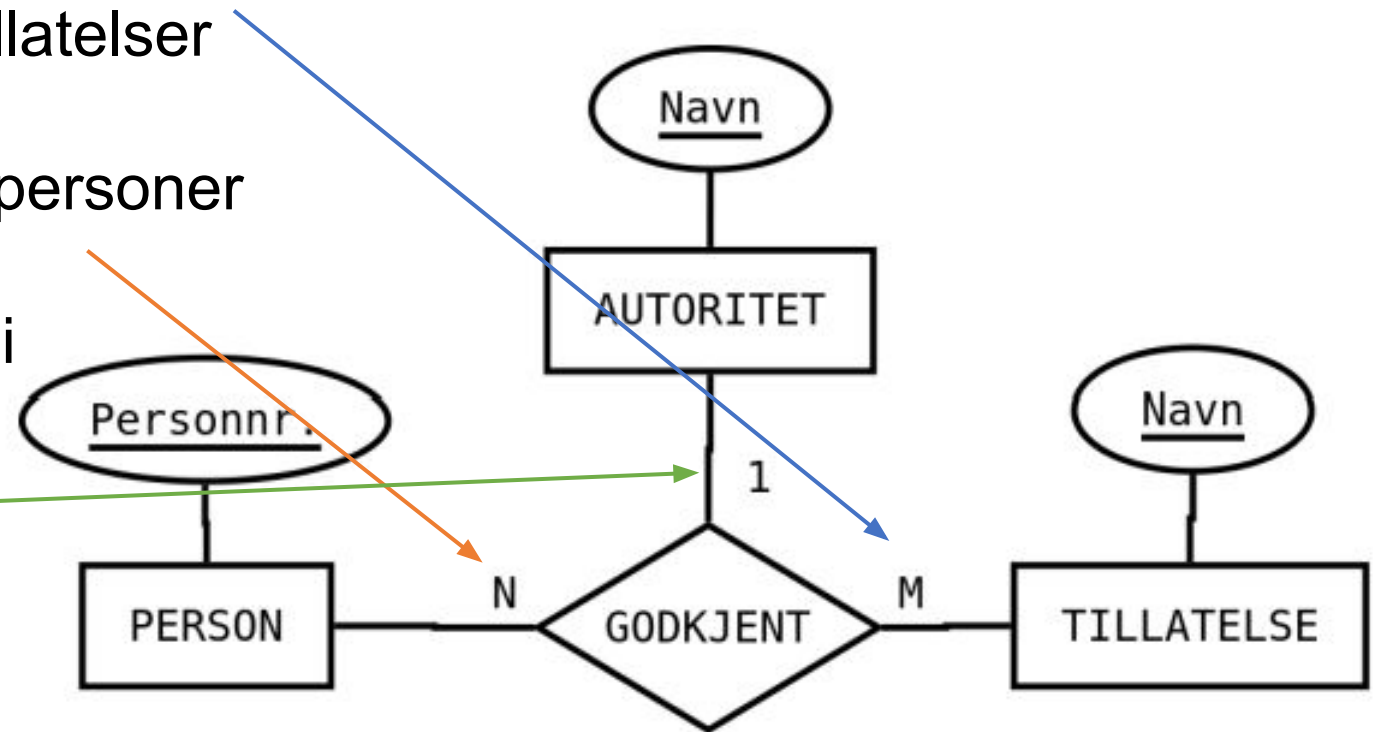
IN2090

Avansert ER og realisering



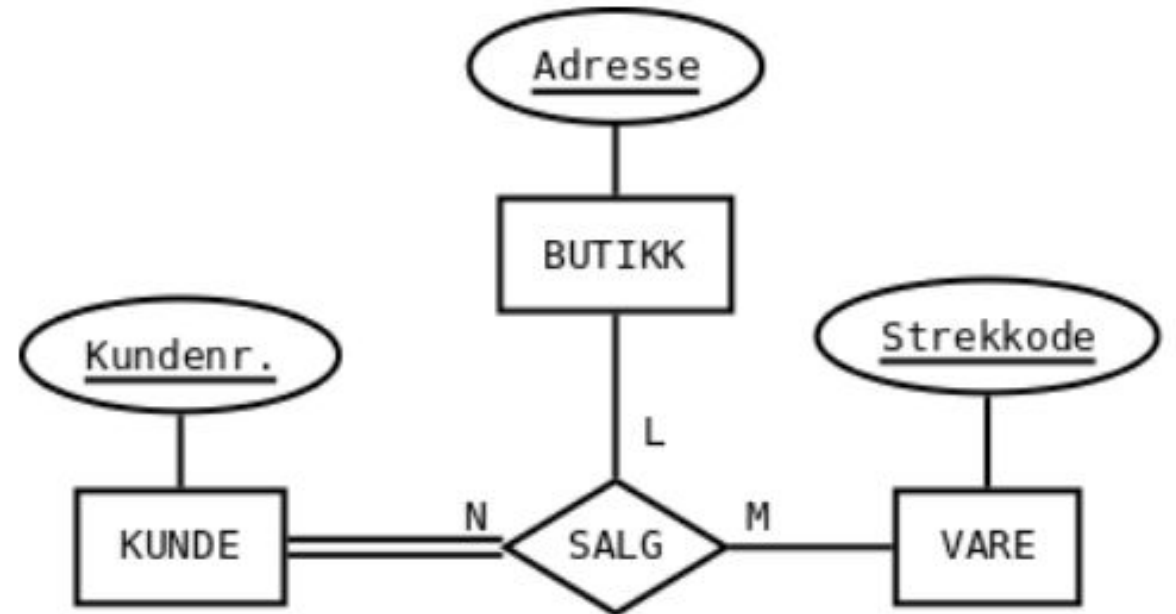
Ternære relasjoner

- En person kan få mange tillatelser fra en autoritet
- En autoritet kan gi mange personer samme tilatelse;
- men kun én autoritet kan gi en bestemt tillatelse til en bestemt person.



Ternære relasjoner

- En kunde kan kjøpe samme vare fra mange butikker (L);
- en butikk kan selge mange varer til samme kunde (M);
- en butikk kan selge samme vare til mange kunder (N).
- En kunde må ha kjøpt minst én vare fra en butikk.



Realisering

1. (Vanlige/normale) entiteter blir tabeller/relasjoner
2. Svake entiteter blir også tabeller/relasjoner:
 - PN: PN identifiserende entitet + PN svak entitet

Relasjoner:

3. 1 : 1, fire valg:
 - Sammensmelting (merge)
 - Legg til FN fra etitet_1 til etitet_2s PN (og omvendt)
 - Ny relasjon/tabell
4. 1 : N, to valg:
 - Ny relasjon/tabell
 - Legge FN inn i N-siden
5. N : M: blir alltid en ny tabell/relasjon
6. Ternære: tilsvarende som for M:N-relasjoner (uavhengig av kardinaliteter)
7. Flerverdiattributter blir alltid en ny tabell/relasjon



Entiteter

- Lag en tabell av entiteten, og:
 - Ta med alle vanlige attributter
 - For sammensatte attributter, ta bare med "underattributtene"
 - Velg en PN. Er det flere nøkler velg en PN og de andre som KN
- Merk:
 - Utleidet/Derived (de stiplede) skal IKKE tas med
 - Flerverdiattributter skal IKKE være med (disse blir en egen tabell)

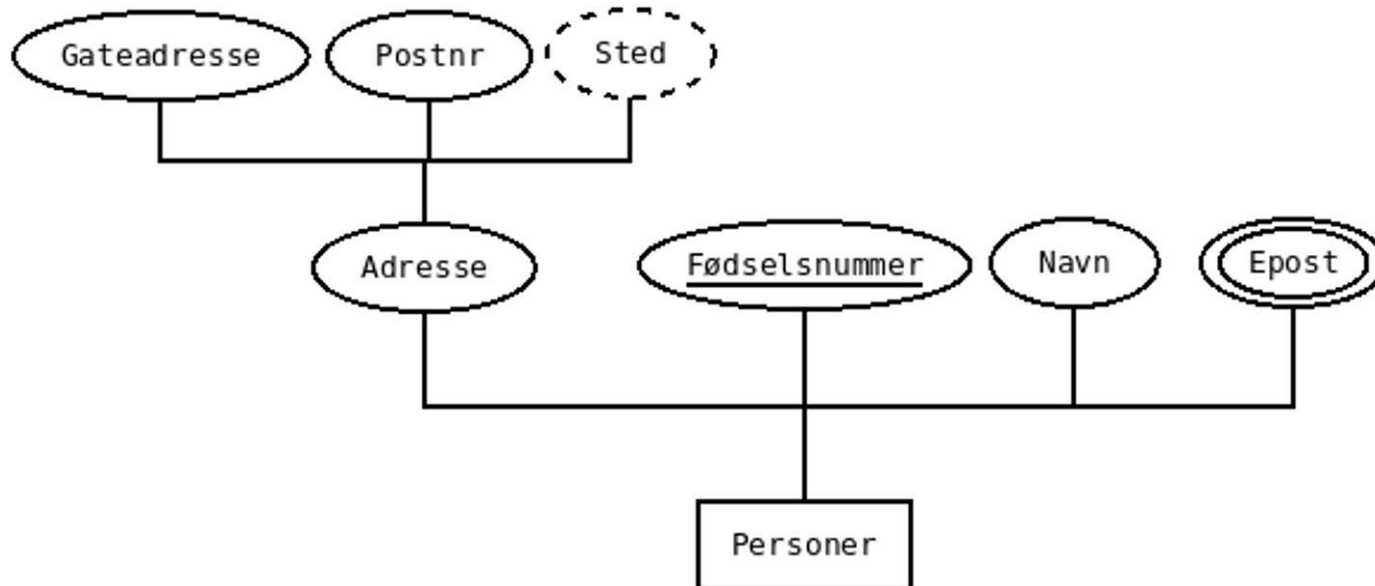


Eksempel

- Personer(Fødselsnummer, Navn, Gateadresse, Postnr)

Flerverdiattributter blir alltid en ny tabell. Tar med PN fra entiteten. Får da:

- Epost(Fødselsnummer, Epost)



Svake entiteter

- Lag en tabell av den svake entiteten, og:
 - Ta med alle vanlige attributter
 - For sammensatte attributter, ta bare med "underattributtene"
 - Legg til PN fra den identifiserende entiteten. Da blir nøkkelen du la til fremmednøkkel til den identifiserende entiteten.
 - PN blir kombinasjonen av den svake nøkkelen og PN fra den identifiserende entiteten
- Merk:
 - Utledet/Derived (de stiplede) skal IKKE tas med
 - Flerverdiattributter skal IKKE være med (disse blir en egen tabell)



Eksempel

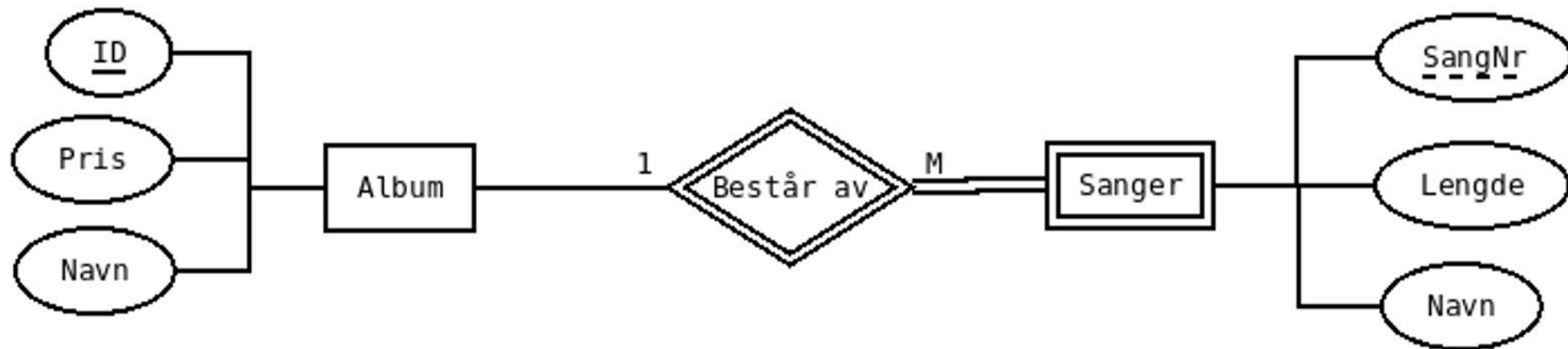
- Sanger(SangNr, AlbumID, Lengde, Navn)

Her blir fremmednøkkelen:

- Sanger(AlbumID) --> Album(ID),

Altså:

Sanger sin AlbumID refererer til Album sin ID



Mange til mange relasjoner (N:M)

- Lag alltid en ny tabell
- Ta med relasjonsattributter (de som går ut av relasjonen)
- Ta med PN fra begge entitetene. Disse danner en ny PN i den nye tabellen (altså felles strek under begge)

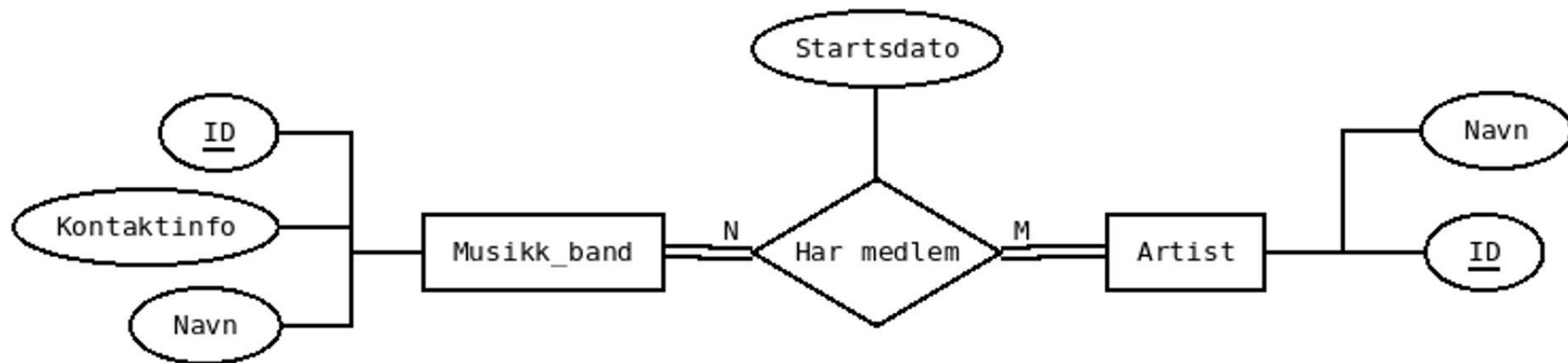


Eksempel

- HarMedlem(BandID, ArtistID, Startsdato)

FNer:

- HarMedlem(BandID) --> Musikk_band(ID)
- HarMedlem(ArtistID) --> Artist(ID)



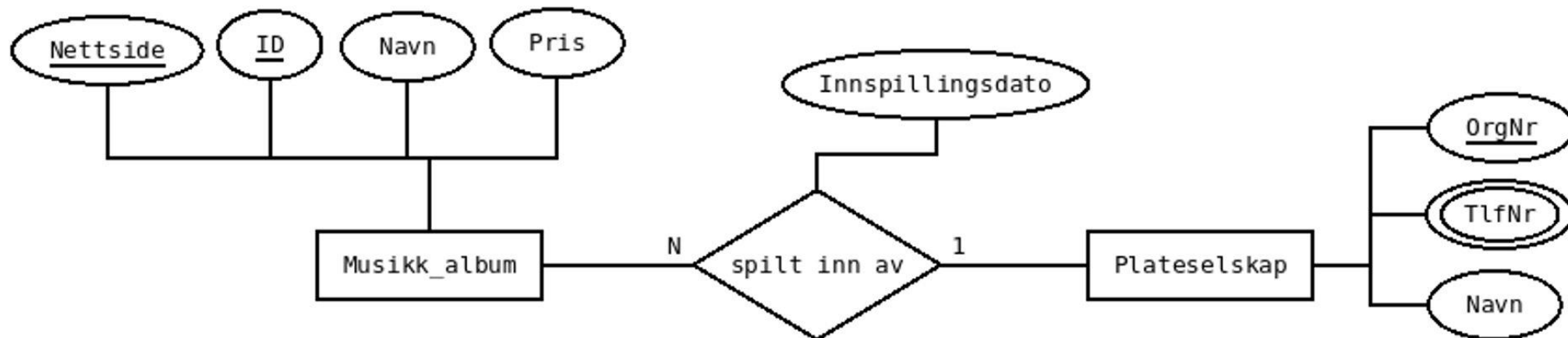
En til mange relasjoner (1:N eller N:1)

- To forskjellige måter å gjøre det på:
 1. Flytt relasjonssattributter (de ut fra relasjonen) til N-siden og ta med PN fra 1-siden. Husk den nye FNen. PN blir den som var på N-siden fra før.
 2. Lage en ny tabell av relasjonen og ta med PN fra begge sider.



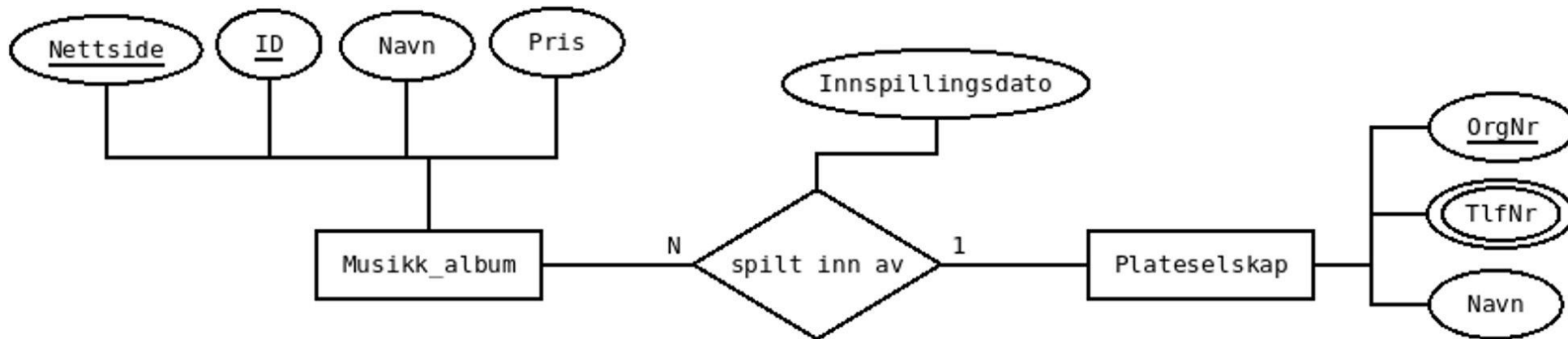
Eksempel alternativ 1

- Musikk_album(ID, Nettside, Navn, Pris, Innspillingsdato, PlateselskapOrgNr)
- PN: ID
- KN: Nettside
- FN: Musikk_album(PlateselskapOrgNR) --> Plateselskap(OrgNr)



Eksempel alternativ 2

- SpiltInnAv(AlbumID, SelskapOrgNR, Innspillingsdato)
- FNER:
 - SpiltInnAv(AlbumID) --> Musikk_album(ID)
 - SpiltInnAv(SelskapOrgNr) --> Plateselskap(OrgNr)
- Denne metoden brukes **sjeldent**. Brukes hvis det er få tupler/attributter i entiteten på N-siden.



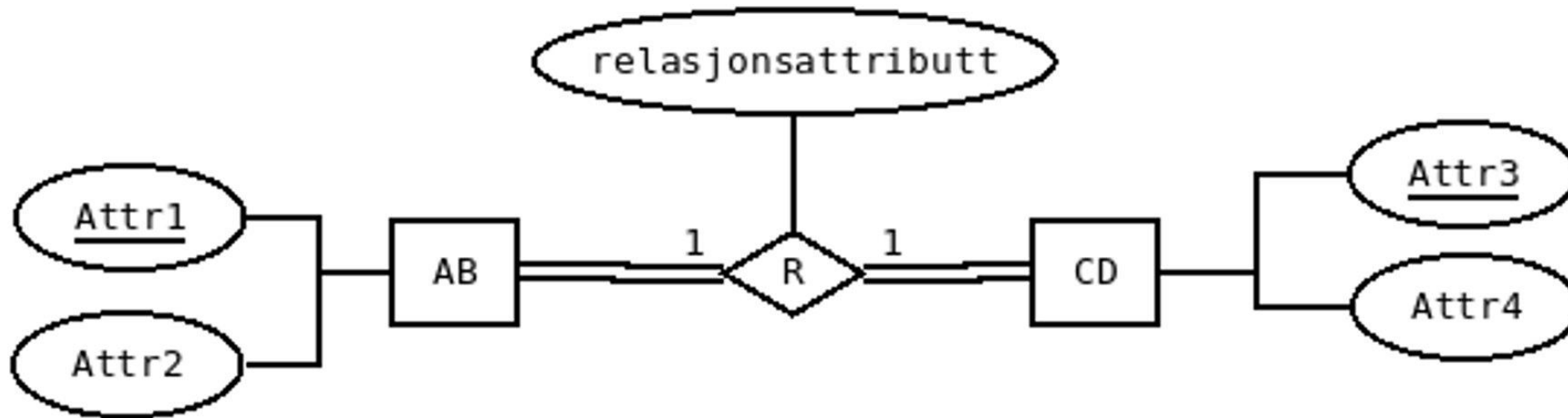
1 til 1 relasjoner

- 3 forskjellige måter å gjøre dette på:
 1. Sammensmetning:
 - Når begge entiteter har total deltakelse inn til relasjonen.
 - Lag en ny tabell og ta med alle attributter fra begge entitetene + relasjonsattributter.
 - Valgfritt hvilken av de to PNene man velger som ny PN i den nye tabellen. Den andre blir en KN.
 - Husk å oppdatere FNer.
 2. (**Foretrukket**) Velg en av sidene og ta med relasjonssattributter og PN fra motsatt side. Hvis det er en total deltakelse (dobbelt strek) med i relasjonen, velg den siden. Husk at dette blir en FN til relasjonen på motsatt side.
 3. "Brukes sjeldent". Formålet er å kryssreferere PNer.
 - Lag en ny tabell med relasjonsattributt og begge PNene.
 - Velg PN.
 - Husk å legge til FNer til disse PNene



Eksempel alternativ 1

- R(Attr1, Attr2, Attr3, Attr4, felles)
- Her kunne vi valg Attr3 som PN og Attr1 som KN.

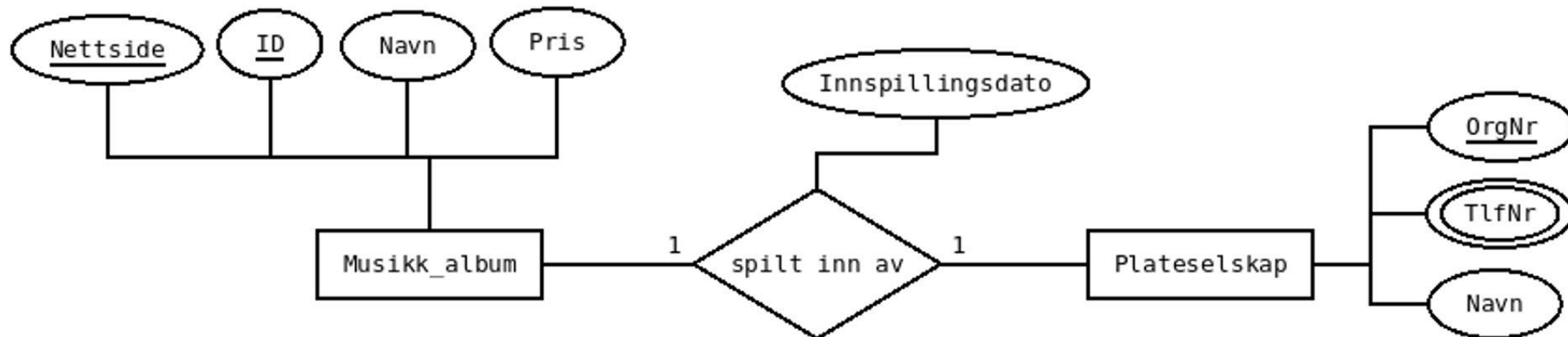


Eksempel alternativ 2

- Musikk_album(ID, Nettside, navn, Pris, innspillingsdato, PlateselskapOrgNr)
- FN: Musikk_album(PlateselskapOrgNR) --> Plateselskap(OrgNr)

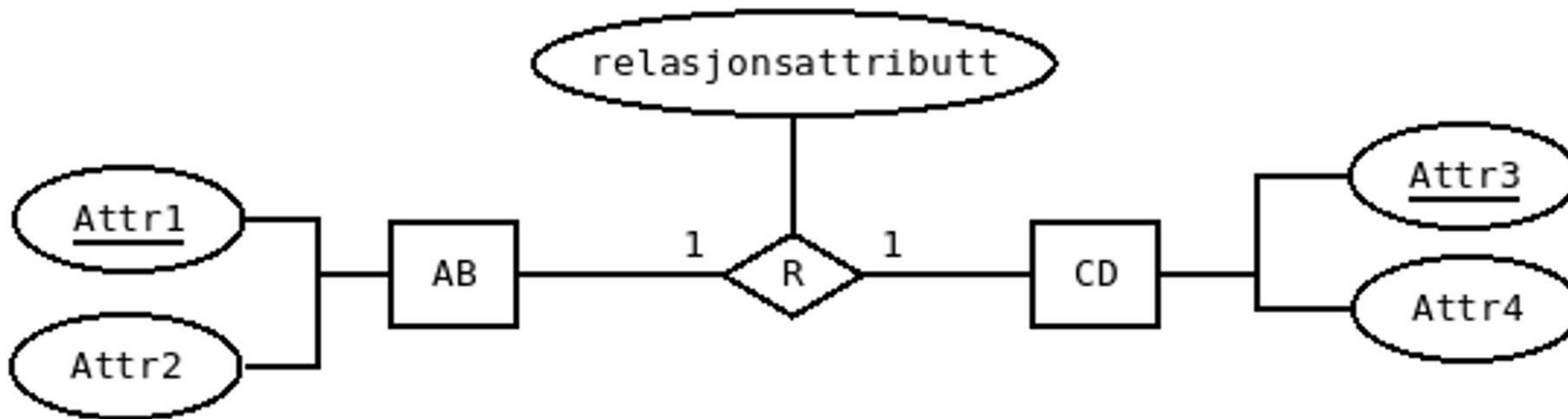
Eller:

- Plateselskap(OrgNr, Navn, Innspillingsdato, albumID)
- FN: Plateselskap(albumID) --> Musikk_album(ID)



Eksempel alternativ 3

- Nytab(Attr1, felles, Attr3)
- FNER:
 - Nytab(Attr1) --> AB(Attr1)
 - Nytab(Attr3) --> CD(Attr3)



Ternære

- Alltid en ny tabell:
 - Ta med relasjonsattributter (de ut fra relasjonen) og alle **nøkkel**attributer (de som inngår i PKene til alle entitetene)
 - Gitt en 1:N:M-relasjon ta med PN fra alle 3: lag en ny PN av kombinasjonen fra N, M.
 - Gitt en N:M:P-relasjon ta med PN fra alle 3: lag en ny PN med kombinasjonen av alle.
 - Gitt en 1:**1**:N-relasjon. Får man to nøkler. 1:N og **1**:N. Velg en som PN og en som KN
 - Gitt en 1:1:1-relasjonen gir 3 nøkler. Velg 1 PN og 2 KN

Husk FNer



Eksempel

Person(personnr)

- KN/PN: {personnr}

Firma(orgnr)

- KN/PN: {orgnr}

Stilling(navn)

- KN/PN: {navn}

Ansatt(person, firma, stilling)

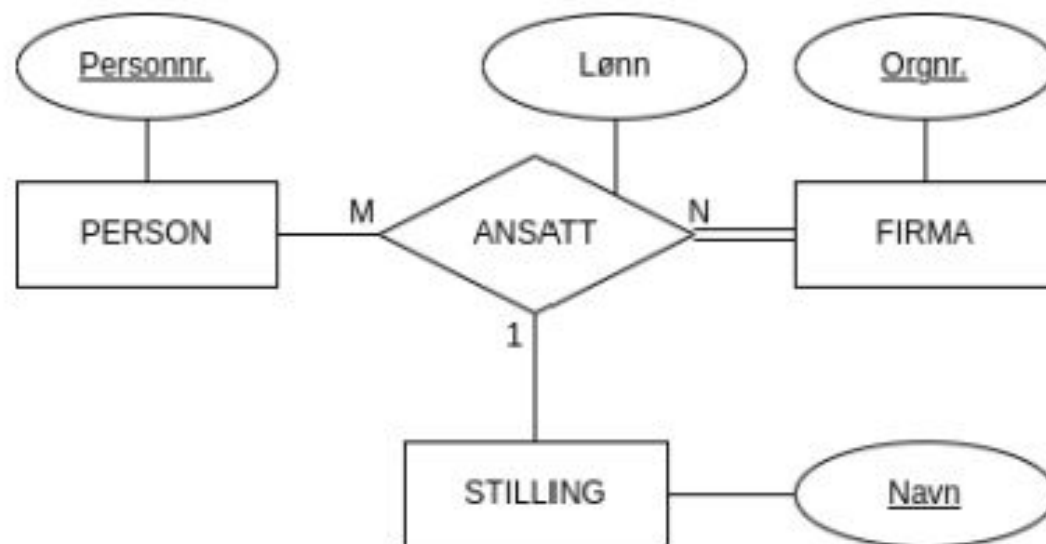
- KN/PN: {person, firma}

- FNER:

Ansatt(person) -> Person(personnr)

Ansatt(firma) -> Firma(orgnr)

Ansatt(stilling) -> Stilling(navn)



Jobb med ukesoppgaver/oblig1

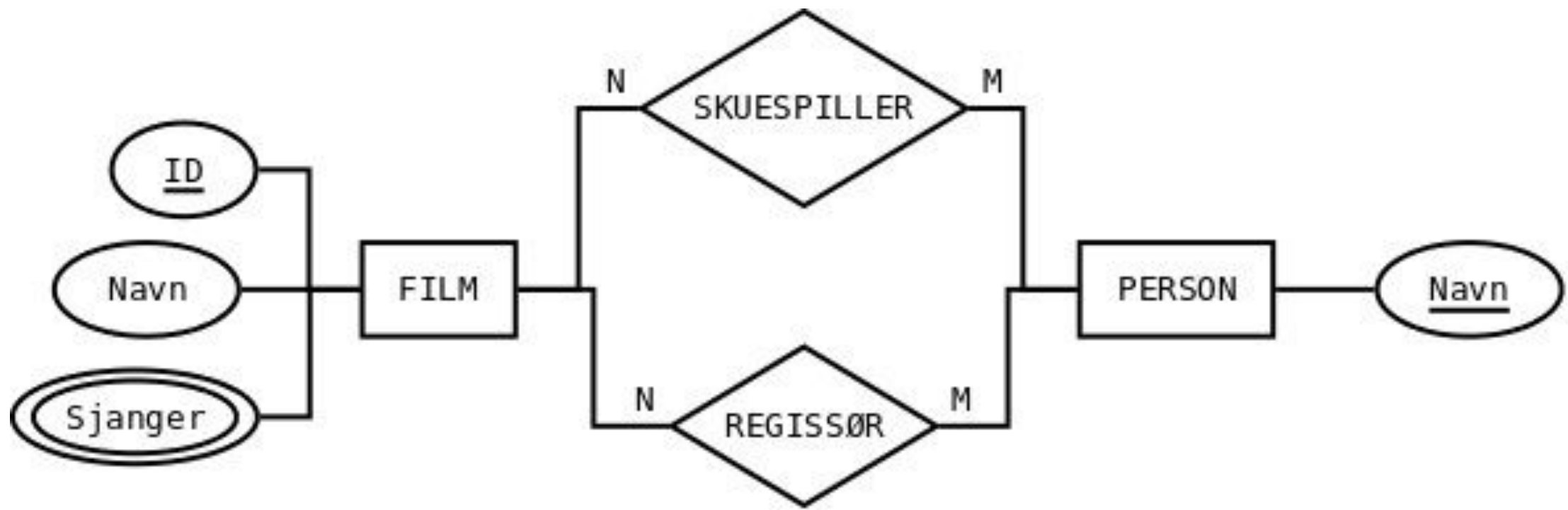
- Oblig1 (Modellering): [oblig1.pdf \(uio.no\)](#)
- **Frist for Oblig1 (Modellering): 28. September kl 23.59!**
- Ukesoppgaver (uke 4: Modellering: ER-modellering 2)
[IN2090-ukesoppgaver: Uke 4 – Universitetet i Oslo \(uio.no\)](#)



Løsningsforlag ukesoppgaver uke 4

1. Vi vil lage oss en liten filmdatabase.
 - Alle **filmene** har **navn**, men ettersom flere filmer kan ha samme navn, gir vi også **unike ID-er** til filmene. For hver film ønsker vi å lagre **en eller flere sjangre**.
 - Til slutt ønsker vi også info om hvilken **skuespiller** (eller skuespillere?) som spilte hovedrollen
 - For disse kan vi anta at alle har **unike navn**.
 - og hvilken **regissør** (regissører?) som regisserte filmen.
 - For disse kan vi anta at alle har **unike navn**.

Prøv å unngå overflødige entiteter og/eller attributter



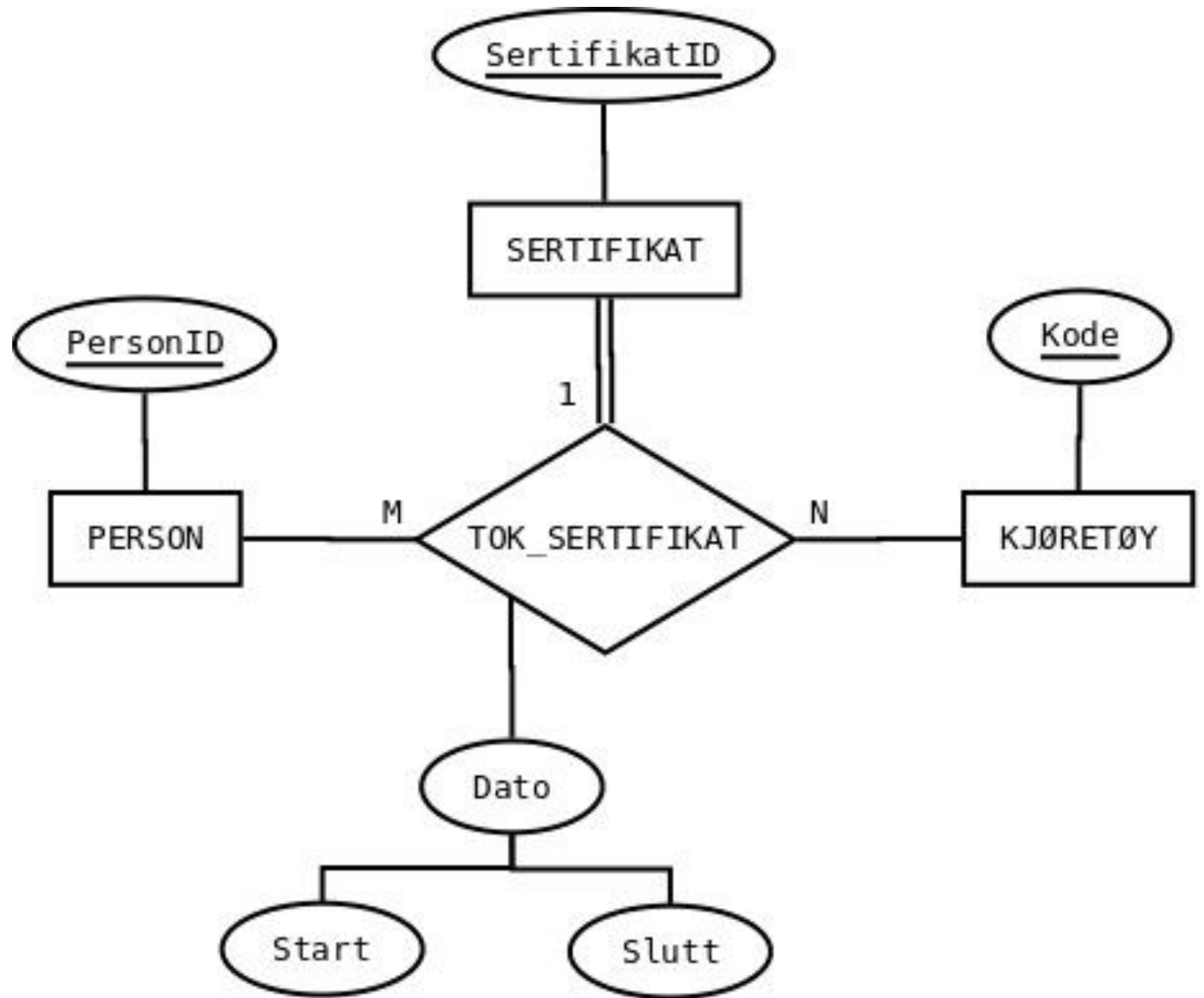
Oppgave 2 - Sertifikater

- Vi skal modellere at **personer** har **sertifikat** og hvilke **kjøretøy** man har sertifikat for.
- En person kan ha sertifikat for flere kjøretøy, og hvert sertifikat er gyldig fra en **startdato** til en **sluttdato**.
- Modeller en ternær relasjon mellom **Person**, **Sertifikat** og **Kjøretøy**. Sett passende attributter til hver entitet, og husk at alle entiteter skal ha en primærnøkkel (som kan bestå av ett eller flere attributter). Modellen skal uttrykke følgende:

Gitt en person og et sertifikat kan det være flere kjøretøy; gitt en person og et kjøretøy kan det kun være ett sertifikat; og gitt et sertifikat og et kjøretøy kan det være mange personer.

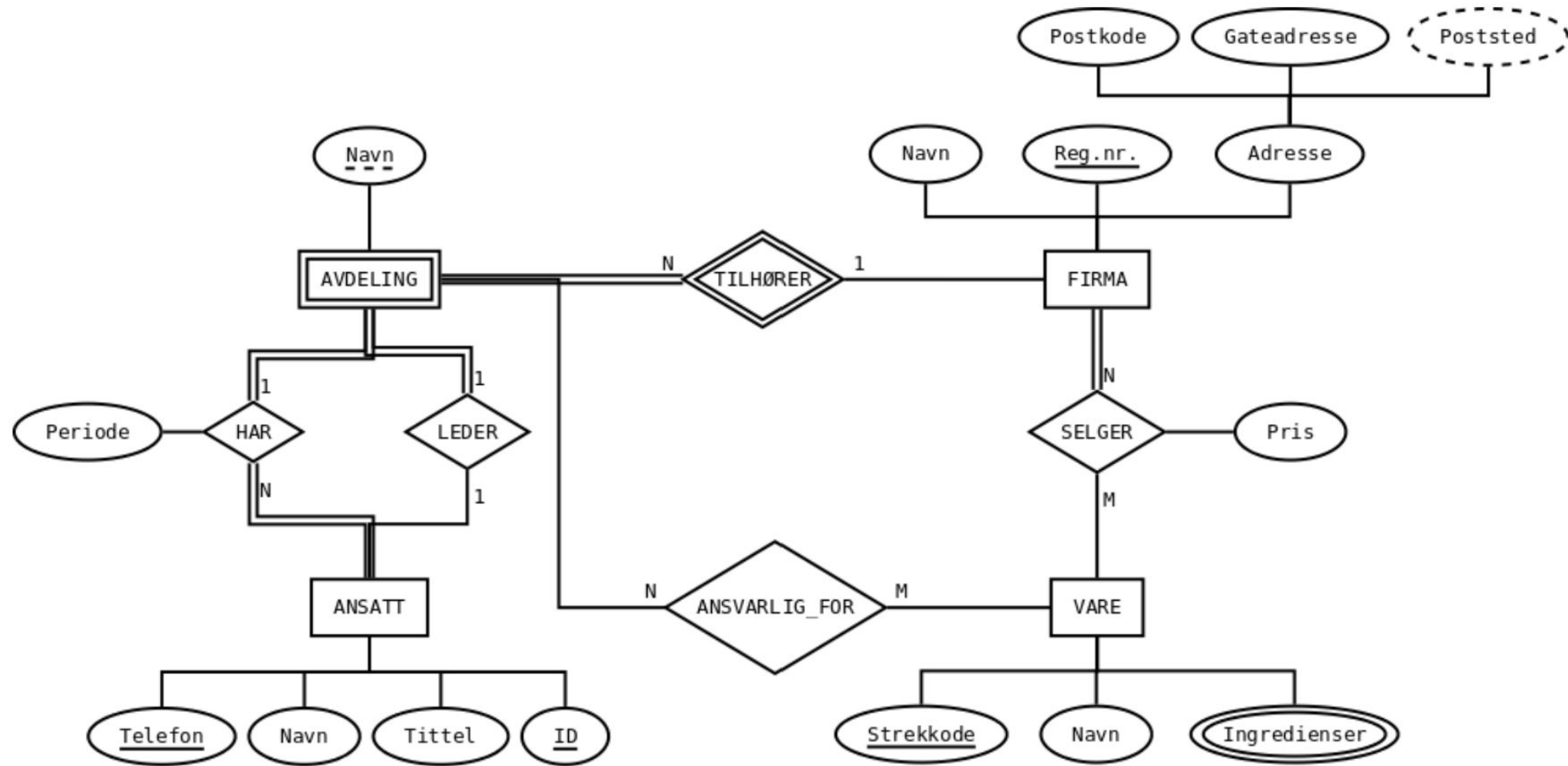
Dato med start og slutt, kunne også vært satt på entiteten SERTIFIKAT som en sammensatt attributt dato, hvis vi antar at datoen gjelder for sertifikatet I seg selv og ikke kombinasjonen PERSON, KJØRETØY, SERTIFIKAT.

Dersom datoen settes på SERTIFIKAT vil alle som har samme sertifikat også ha samme dato, derfor gir det mer mening å sette dato på relasjonen.



Realisering

Oppgave 3 - Varer og ingredienser



Løsningsforslag - Oppgave 3

Entiteter (ikke-svake)

Firma(regnr, navn, postkode, gateadresse)

KN: {regnr}, PN: {regnr}

Vare(strekkode, navn)

KN: {strekkode}, PN: {strekkode}

Ansatt(id, telefon, navn, tittel)

KN: {id}, {telefon}, PN: {id}

Løsningsforslag - Oppgave 3

Svake entiteter

Avdeling(navn, firma)

KN: {navn, firma}, PN: {navn, firma}

Fremmednøkler:

Avdeling(firma) -> Firma(regnr)

Putter leder inn i avdeling:

Avdeling(navn, firma, leder)

KN: {navn, firma}, {leder}, PN: {navn, firma}

Fremmednøkler:

Avdeling(firma) -> Firma(regnr)

Løsningsforslag - Oppgave 3

N-M-relasjoner

Ansvarlig_for(avdeling, firma, vare)

KN: {avdeling, firma, vare}, PN: {avdeling, firma, vare}

Selger(firma, vare, pris)

KN: {firma, vare}, PN: {firma, vare}

Fremmednøkler:

Ansvarlig_for(avdeling, firma) -> Avdeling(navn, firma)

Ansvarlig_for(vare) -> Vare(strekkode)

Selger(firma) -> Firma(regnr)

Selger(vare) -> Vare(strekkode)

Løsningsforslag - Oppgave 3

fler-verdi attributter

Ingrediens(vare, ingrediens)

KN: {vare, ingrediens}, PN: {vare, ingrediens}

Fremmednøkler:

Ingrediens(vare) -> Vare(strekkode)

Løsningsforslag - Oppgave 3

Endelig skjema

Firma(regnr, navn, postkode, gateadresse)

Vare(strekkode, navn)

Ansatt(id, telefon, navn, tittel)

Avdeling(navn, firma, leder)

Ansvarlig_for(avdeling, firma, vare)

Selger(firma, vare, pris)

Ingrediens(vare, ingrediens)

Fremmednøkler:

Avdeling(firma) -> Firma(regnr)

Ansvarlig_for(avdeling, firma) -> Avdeling(navn, firma)

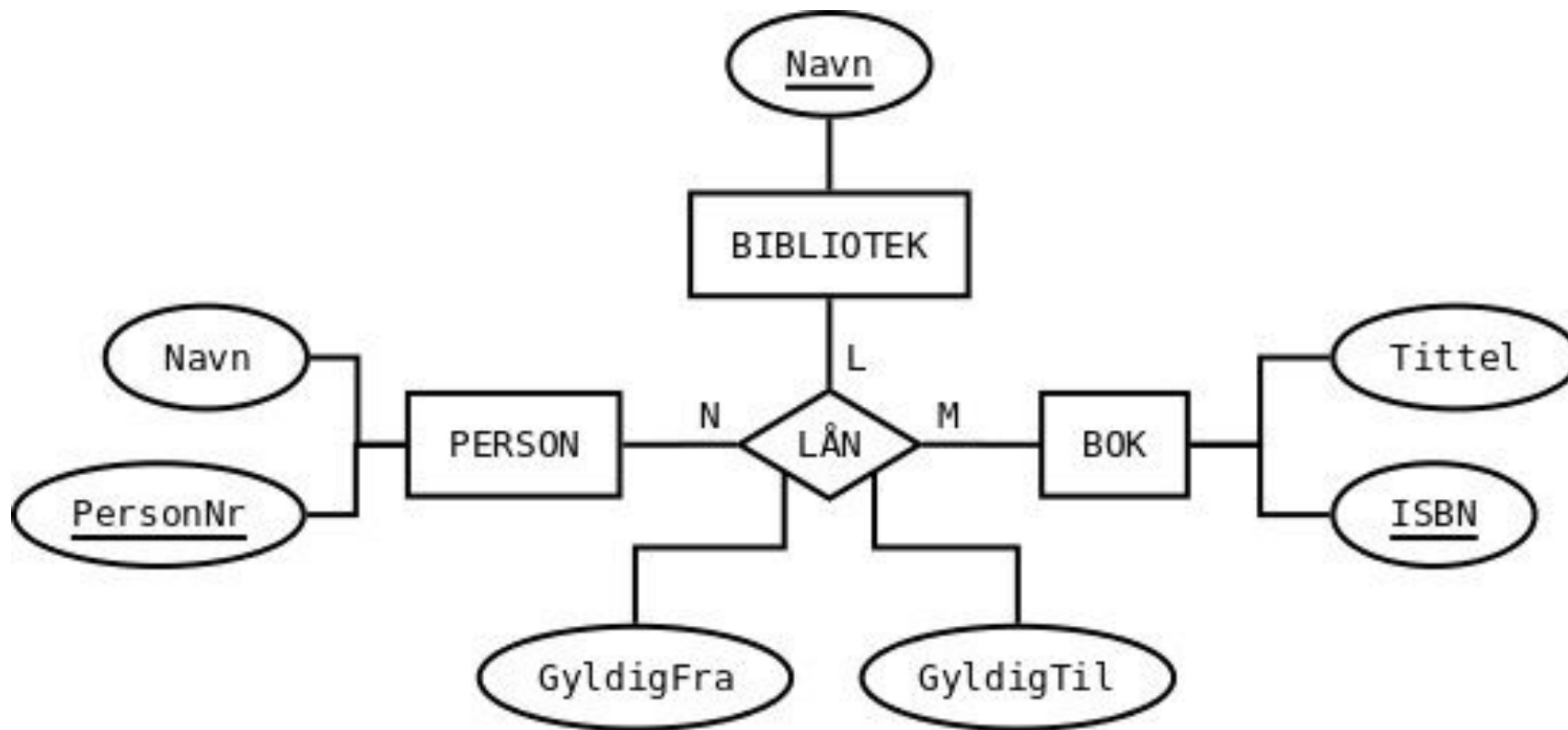
Ansvarlig_for(vare) -> Vare(strekkode)

Selger(firma) -> Firma(regnr)

Selger(vare) -> Vare(strekkode)

Ingrediens(vare) -> Vare(strekkode)

Oppgave 4 - Lån



Oppgave 4 - Lån

Person(persnr, navn)
KN: {persnr}, PN: {personnr}

Bibliotek(navn)
KN: {navn}, PN: {navn}

Bok(isbn, tittel)
KN: {isbn}, PN: {isbn}

Lån(person, bibliotek, bok, gyldigfra, gyldigtil)
KN: {person, bibliotek, bok}, PN: {person, bibliotek, bok}

Fremmednøkler:
Lån(person) -> Person(persnr)
Lån(bibliotek) -> Bibliotek(navn)
Lån(bok) -> Bok(isbn)