Datamodellering i det virkelige liv

Jan-Thore Bjørnemyr

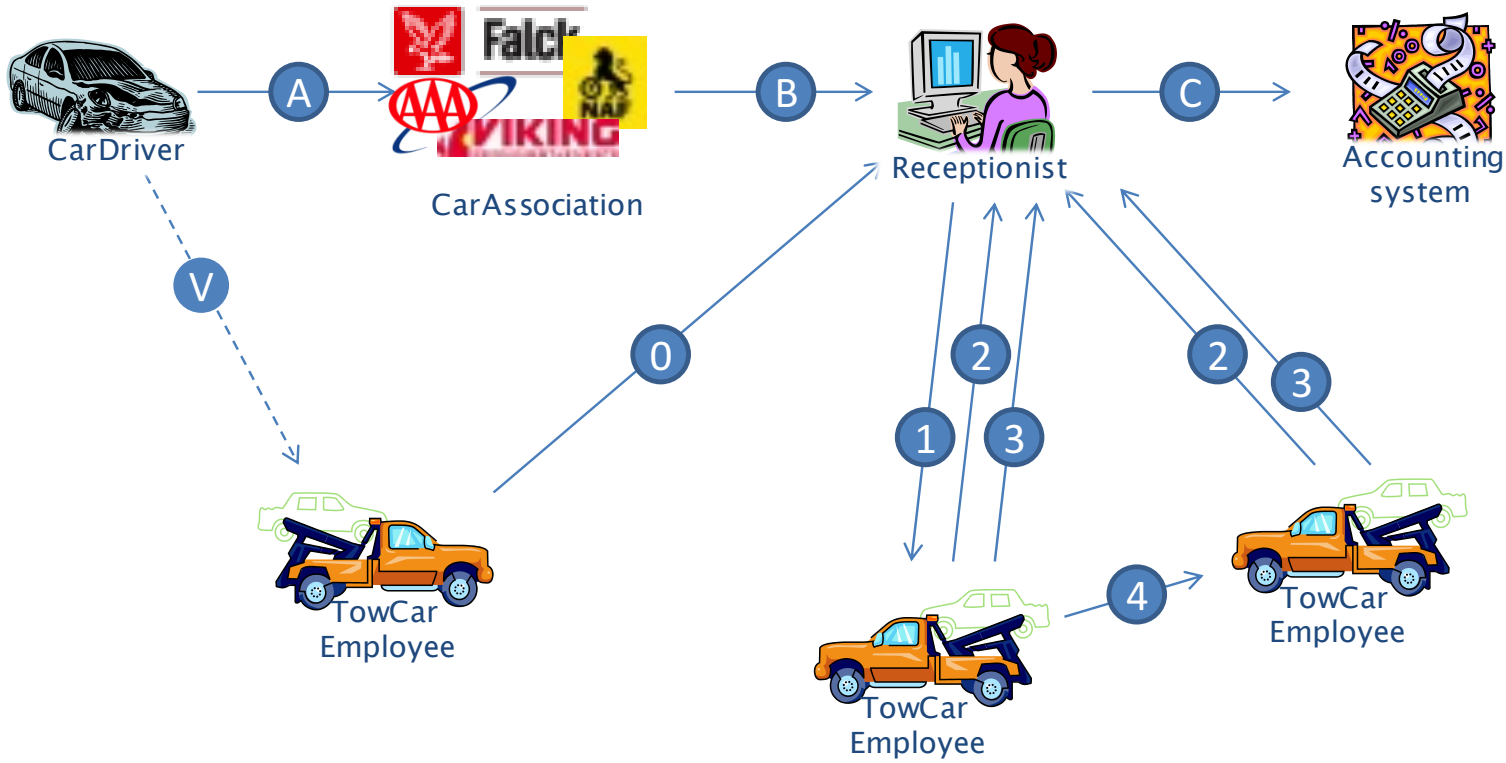
Jan-Thore Bjørnemyr

- Cand. Scient., databehandling 1991
- Jobbet for Ericsson, IBM og Control Data
- Gründer
- Selvstendig konsulent
- Canada, USA, Argentina, Mexico, Colombia, Ungarn, Filippinene, England, Danmark

Datamodell

- Hvorfor lager vi datamodeller?
- Utgangspunktet er et behov!
  - Vi trenger et system
  - Systemet trenger (kanskje) en database
  - Databasen trenger en beskrivelse
  - Beskrivelsen er datamodellen

Eksempel Informasjonssystem

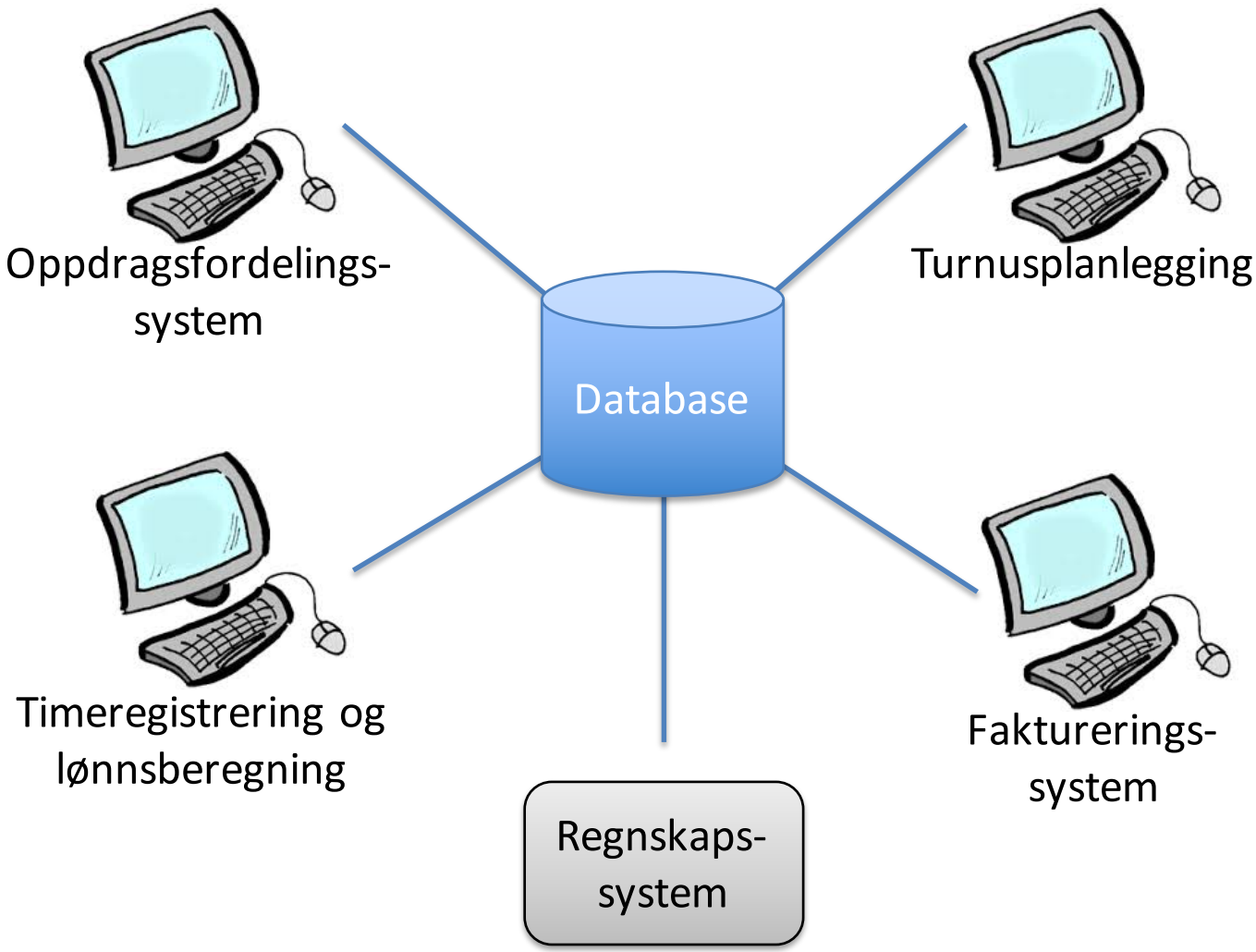


Information entities
- CarDriver
- CarAssociation
- Receptionist
- TowCarEmployee
- AccountingSystem

Functions
- A::Informational
- B::ExternalConnector
- C::SendToAccunting
- V::Informational
- 0::Connector::SMS[0]
- 1::Connector::SMS[1]
- 2::Connector::SMS[2]
- 3::Connector::SMS[3]
- 4::Informational

Forms
- IncidentRegistration
  - FunctionList[1,..]
- DispatchPage
  - FunctionList[1,..]
- WorkReporting
  - FunctionList[C,..]

Informasjonssystem



Databasen

- Felles ressurs
- Felles struktur
- Felles regelverk
- MEN
  - alle systemene ser ikke nødvendigvis alt!

- Databasen må beskrives
  - derfor datamodeller og datamodellering

Litt om ORM og ER

- ORM er ikke ORM
- ORM (= Object Relational Mapping)
- ORM (= Object Role Modelling)
- NIAM (= Natural Language Information Analysis Method)
- ER (= Entity Relationship Method)

ORM vs. ER

ORM

- Konseptuell (+)
- Bottom-up
- Volumiøs (-)
- Presis (+)
- Gir normalisert struktur(+)
- Lite utberedt (-)

ER

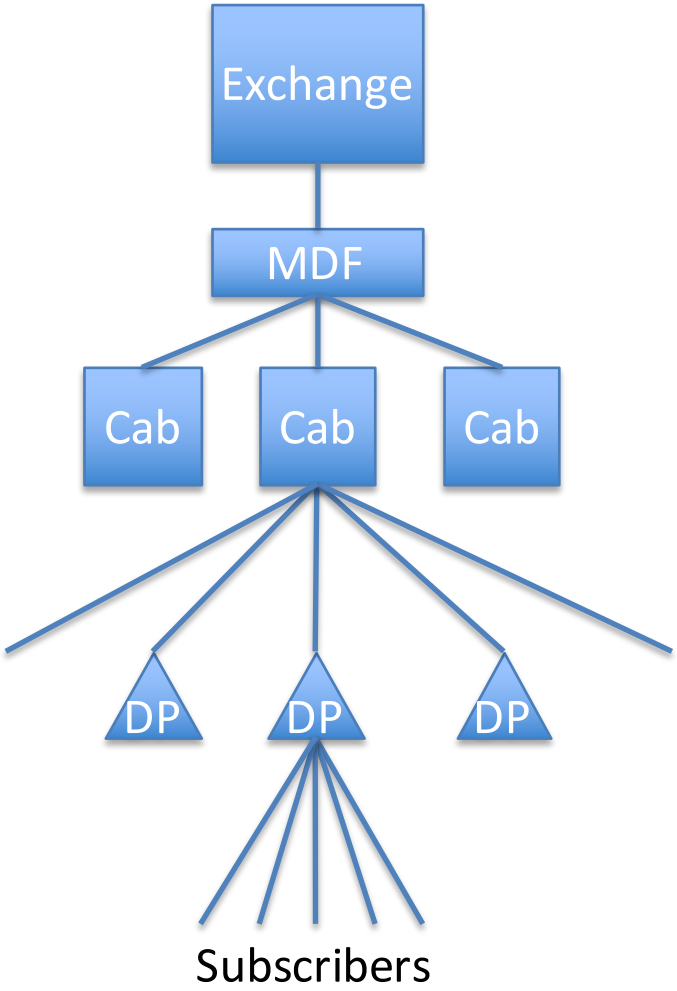
- Tabell modellering (-)
- Top-down
- Kompakt (+)
- Gir god oversikt (+)
- Ikke normalisert struktur (-)
- Veldig utberedt (+)

Min personlige erfaring er at det er enklere å kommunisere en ORM modell med en kunde enn en ER modell

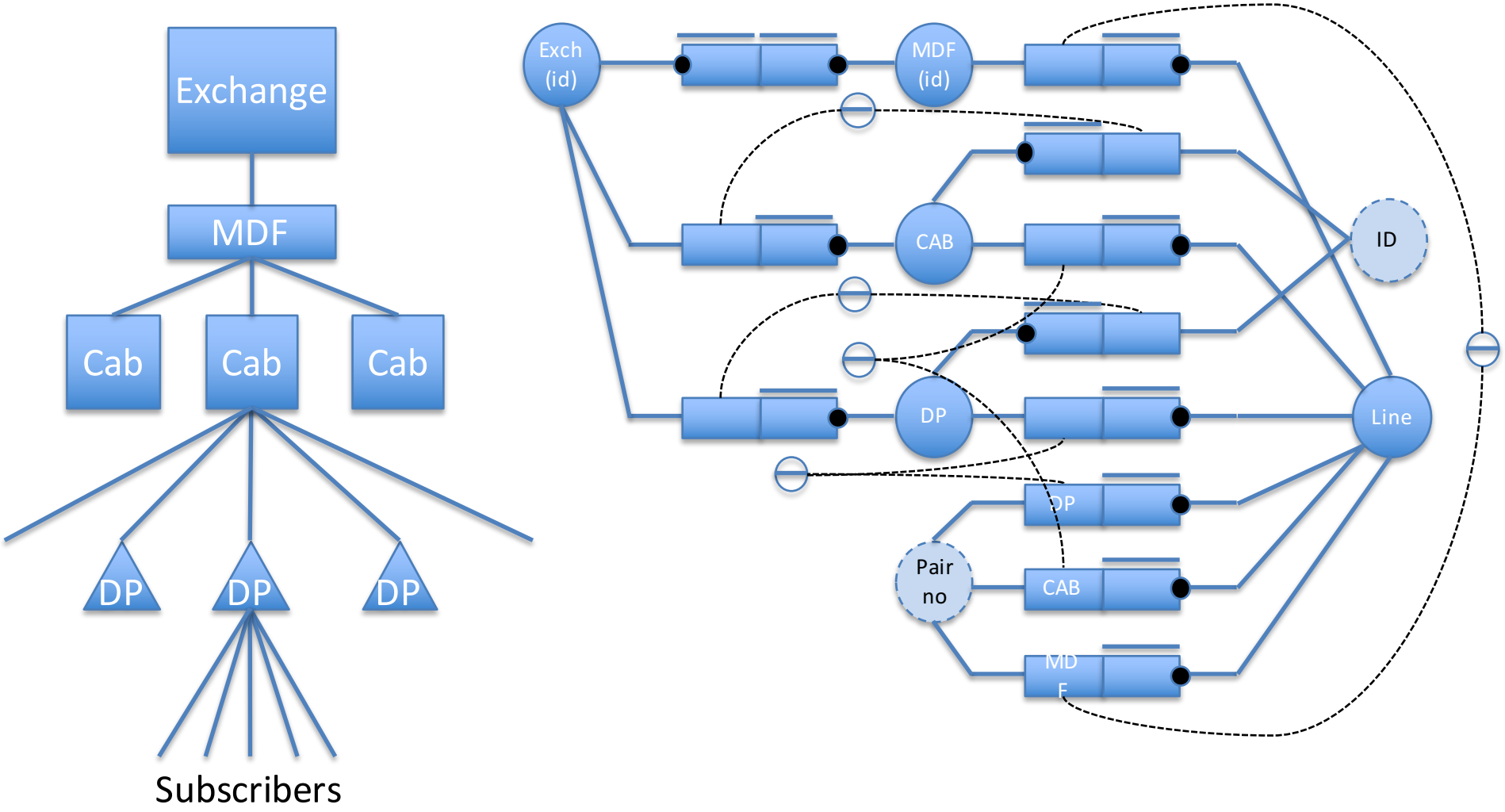
Datamodelleringsprosessen

- Kommunikasjon med kunden
- Verktøy: Munn og ører, spør og lytt
- Pass på at du gjør deg forstått
- Avgrensning
- Finn strukturer, vær kritisk
- Finn regler, let etter motsigelser
- Se forskjell på data og presentasjon
- Test modellen!

Datamodelleringsprosessen

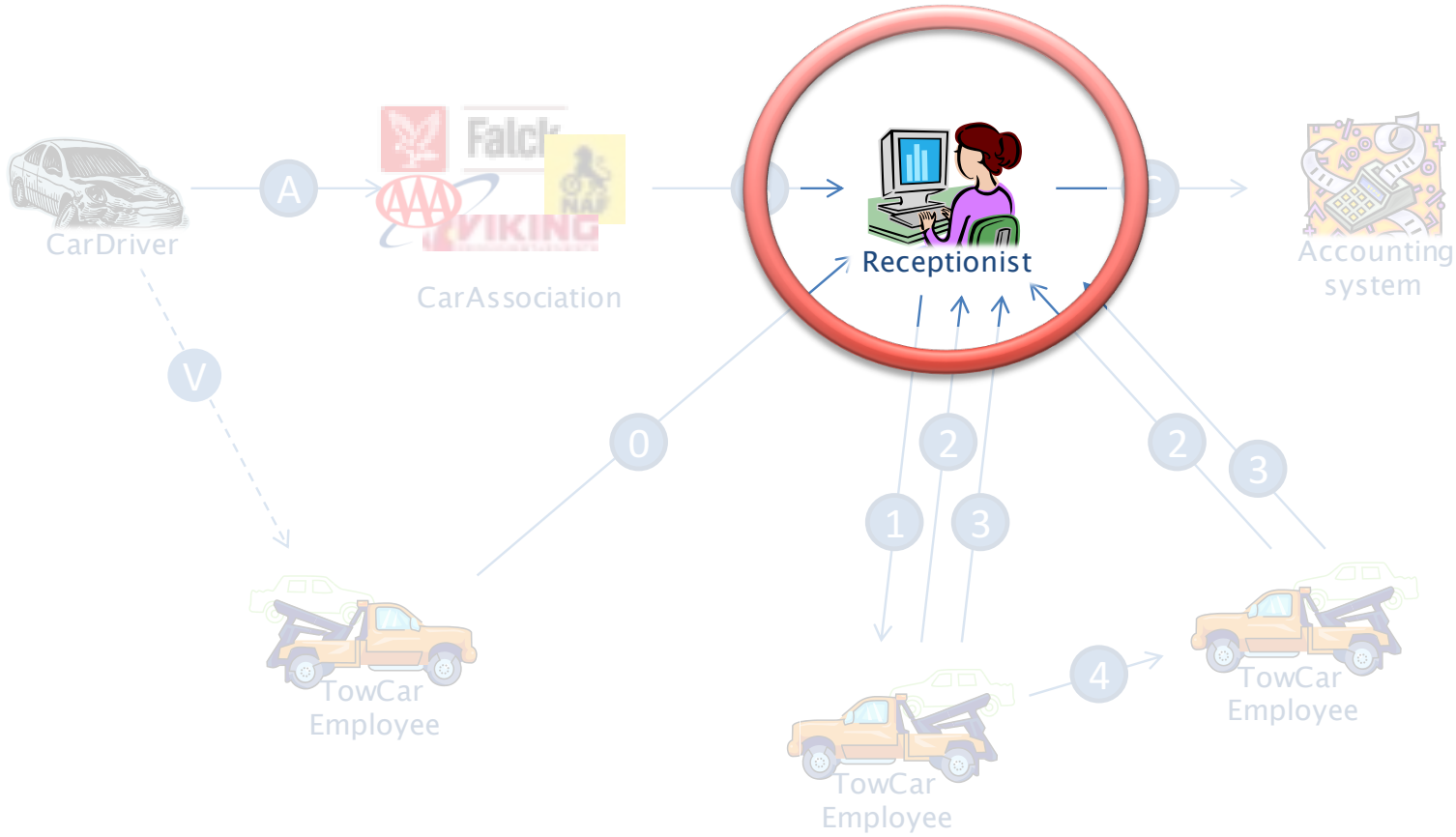


Datamodelleringsprosessen



Noen vanlige problemer

1. Avgrensning



Information entities
- CarDriver
- CarAssociation
- Receptionist
- TowCarEmployee
- AccountingSystem

Functions
- A::Informational
- B::ExternalConnector
- C::SendToAccunting
- V::Informational
- 0::Connector::SMS[0]
- 1::Connector::SMS[1]
- 2::Connector::SMS[2]
- 3::Connector::SMS[3]
- 4::Informational

Forms
- IncidentRegistration
  - FunctionList[1,..]
- DispatchPage
  - FunctionList[1,..]
- WorkReporting
  - FunctionList[C,..]

2. Forståelse av struktur

Hva i all verden er en adresse?

Postadresse, besøksadresse, fakturaadresse, leveringsadresse, … , … , …

2800 Bartons Bluff Ln. Apt #611
Austin, TX, 78746

Grønnegata 11
2317 Hamar

En dagsmarsj øst fra den hvite steinen ved elva

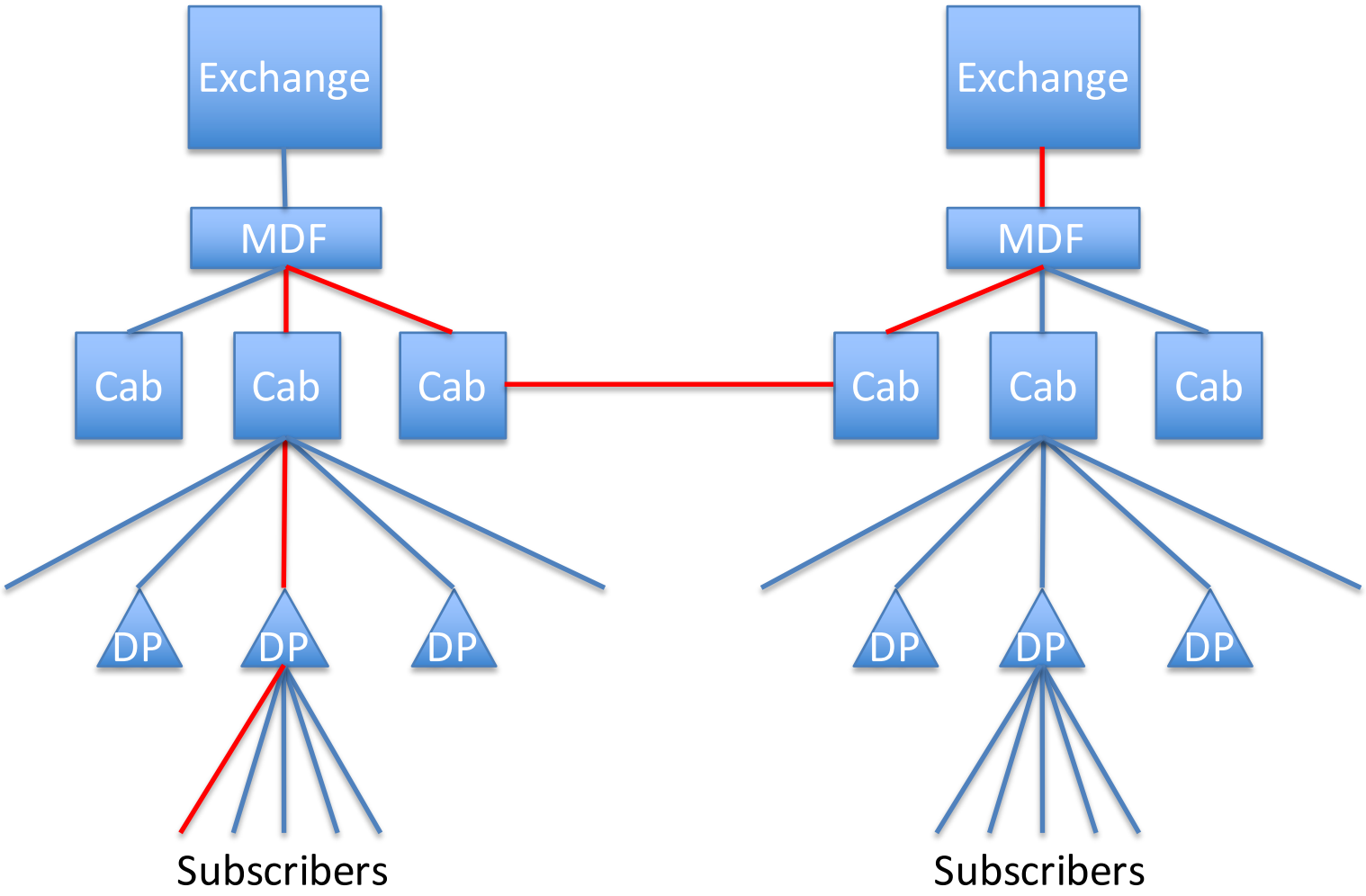
3. Data vs. presentasjon

Hva er et telefonnummer?
Hva er data og hva er presentasjon?

800-MY-APPLE
800-692-7753

Dette er kanskje eksempel på en presentasjon, men det skaper allikevel problemer.
Hvordan skal vi hindre noen i å bruke 800-MY-BPPLE??

4. Kart og terreng



Hvorfor beskrive regler i en modell?

- Det er rimelig å anta at samme regel gjelder for alle applikasjonssystemene
- Regler bør ha:
  - En beskrivelse
  - En implementasjon
  - Implementert ett sted

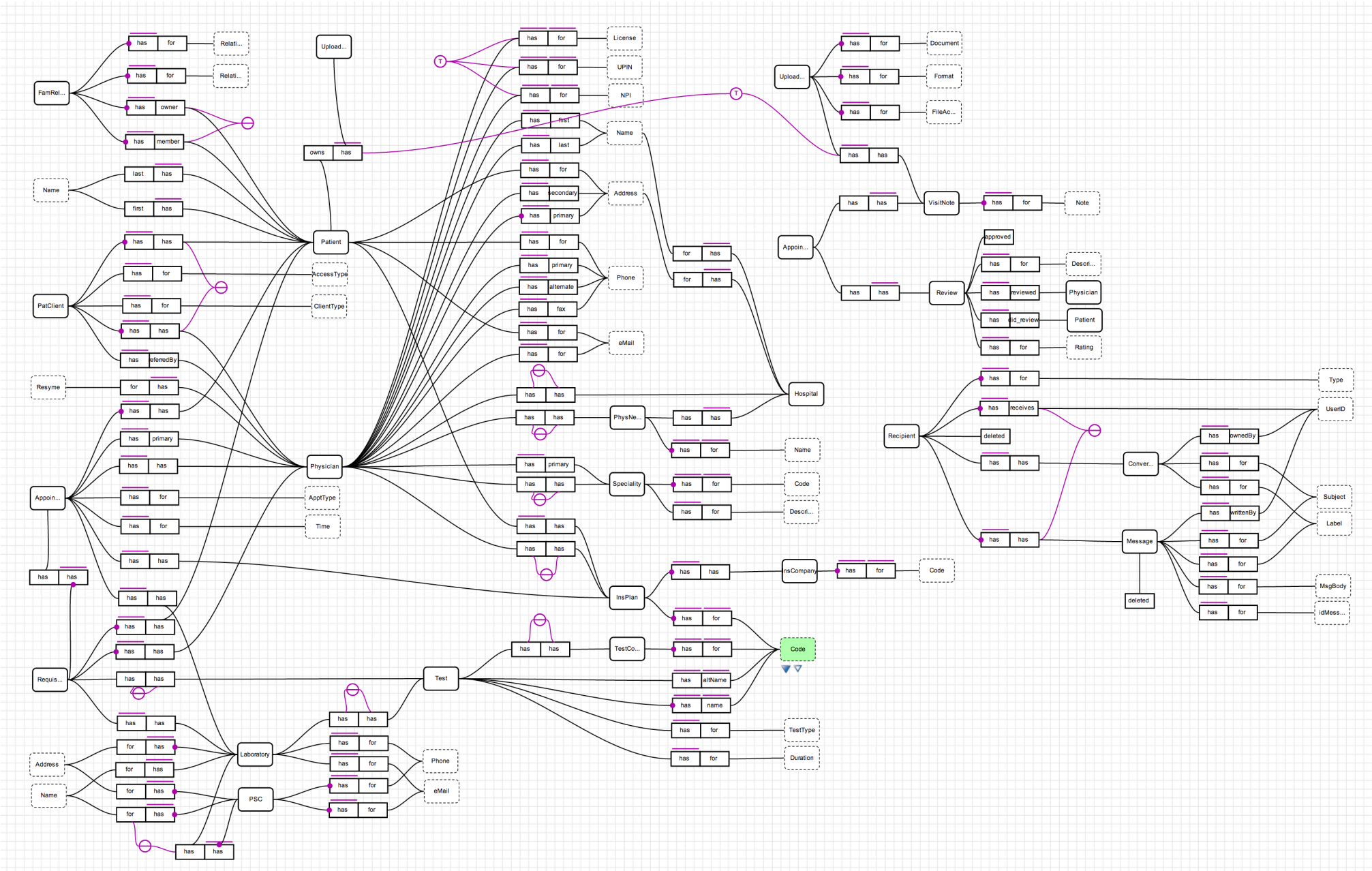
Når er datamodellen ferdig?

- Testing av modellens påstander?
- Er modellen vår egnet?
- Databasen er populert med testdata?
- Databasen har reelle data?
- Hva gjør vi med eksisterende data?
- Hva med historiske data?

Utviklingsfasen

- Skranker er noe HERK!
- Mandatory
- Foreign keys
- Check constraints
- Andre constraints
- Slås på etter hvert!

Datamodell, eksempel



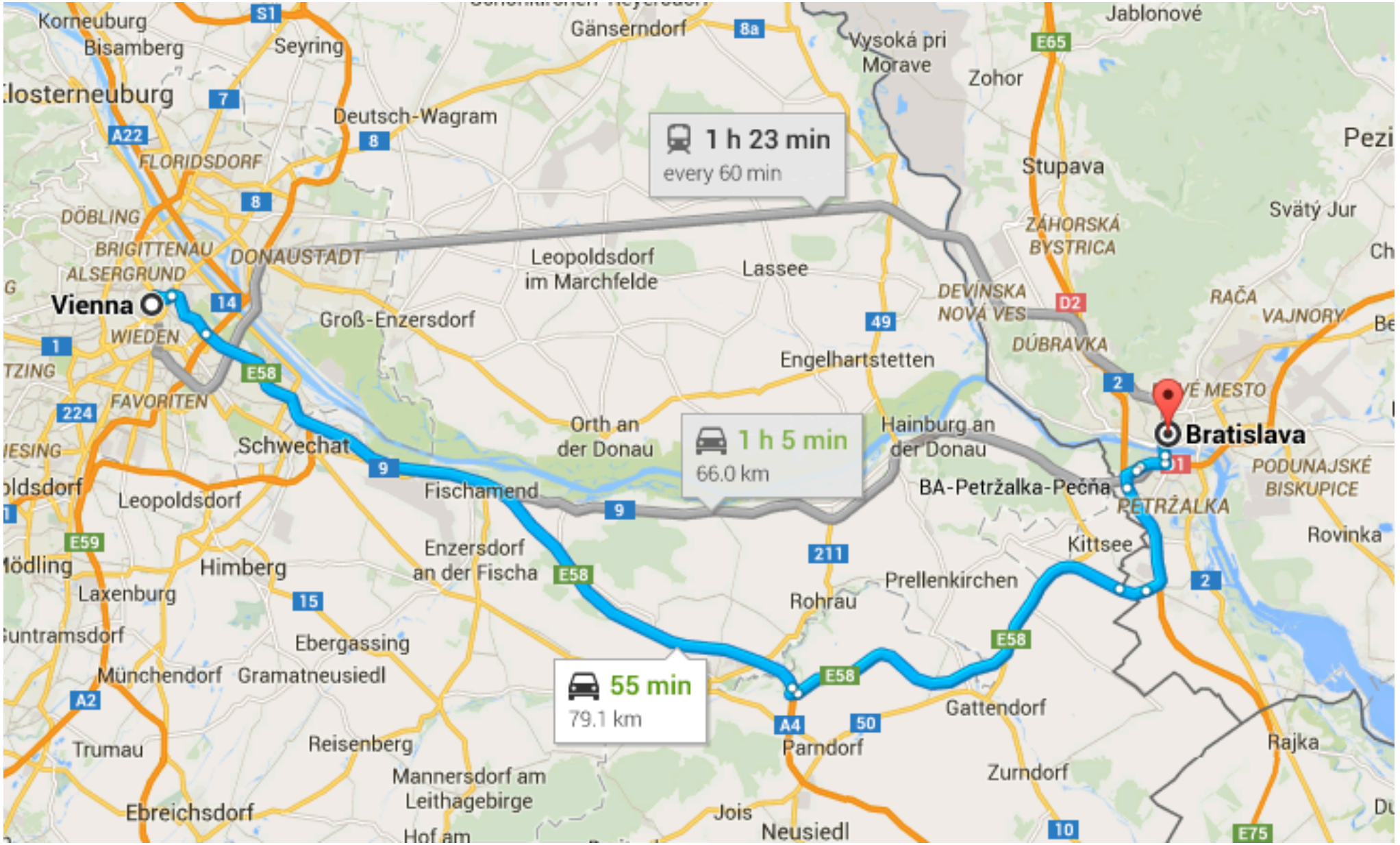
Alt er en prosess

Datamodellen:

- utvikles i parallell med spesifikasjon
- utvikles videre under programmeringen
- får sin egentlige test når reelle data legges inn

Datamodellen er et levende dokument!

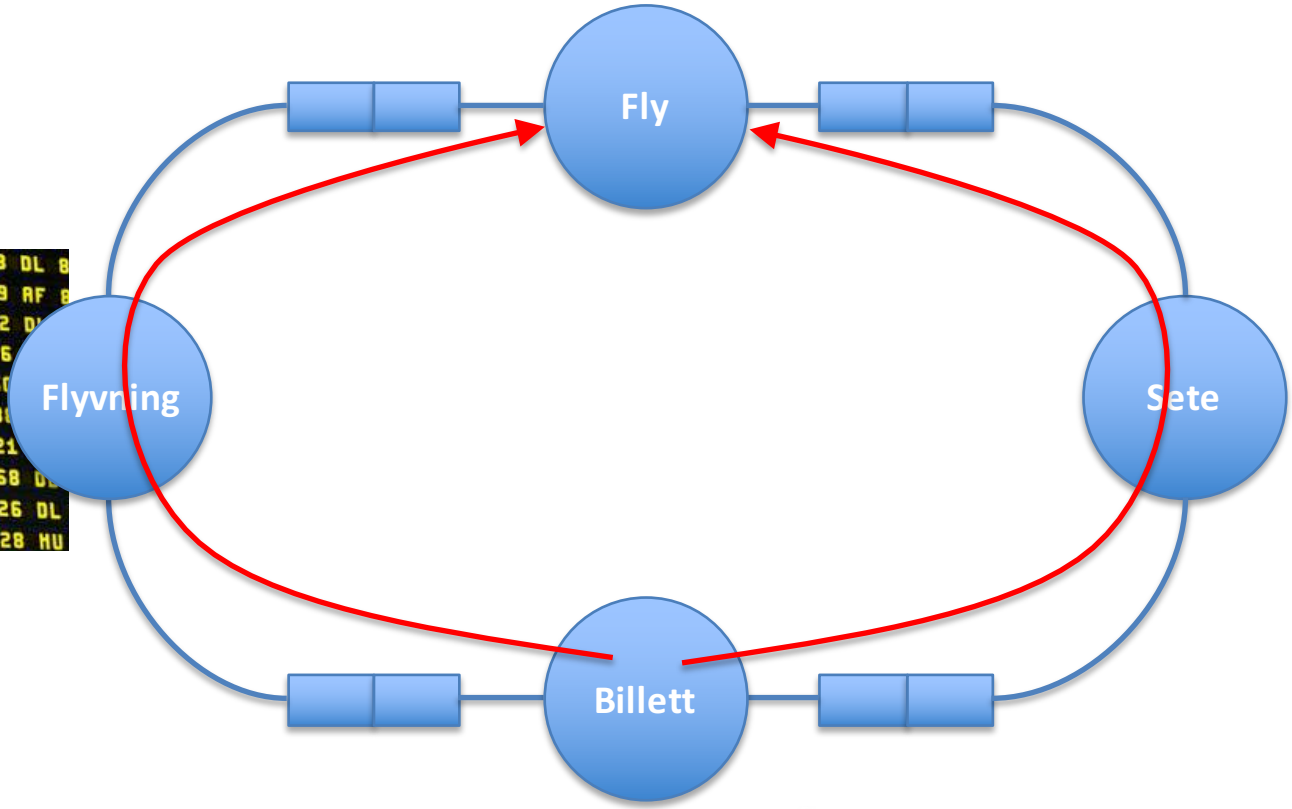
Equivalence-of-path

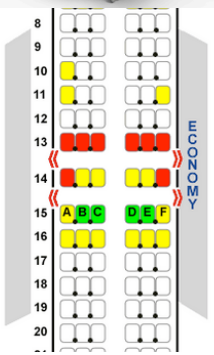


Equivalence-of-path

- Equivalence-of-path er et sett regler som vies liten oppmerksomhet, men som finnes i nesten alle modeller
- Dette er noe som oppstår når det er to forskjellige stier (eller ”paths”) mellom to objekter i modellen
- Disse to stiene er sjeldent uavhengige
- Man ble først klar over problemet da man begynte å lage store billettbestillingssystemer
- Flyselskaper har gjerne mange fly av forskjellige typer, og disse flys i forhold til en gitt turnus og en ruteplan
- Utforingen er å sikre at dataene i databasen faktisk henger sammen (at det faktisk flyr et fly med sete 48H på den datoen som er angitt i billetten)


ATLANTA
PRAGUE
NEW YORK-JFK
CHICAGO
MEXICO
ATLANTA
LOS ANGELES
WASHINGTON
BEIJING


Fly
Flyvning
Sete
Billett


ECONOMY
A B C
D E F


AIRLINE TICKET
AIRLINE TICKET
142
142
BOARDING PASS

Varianter av eop

- Eksempler
  - Klassekontakt skal ha barn i den klassen vedkommende er klassekontakt for
  - Pilot skal være sertifisert for flytypen han skal fly
  - En fotballdommer kan ikke dømme kamper med lag fra egen krets

- Noen eop skranker løses strukturelt,
  noen gjennom subset-skranker
  og noen må rett og slett programmeres
- Generelt skal eop implementeres i databasen

To interessante temaer

(som dessverre må håndteres)

- Brukere og brukerrettigheter
  - Tilgang til skjermbilder
  - Tilgang til funksjonalitet i skjermbildene
  - Tilgang til data i skjermbildene
- Språk og språkuavhengighet
  - Oversettelse av brukergrensesnitt
  - Oversettelse av data

Eksamenstips - ORM

- Lesbarhet

  Det må være mulig å faktisk lese det som er skrevet

- Syntax

  Objekter, relasjoner

- Begrepsdannelse

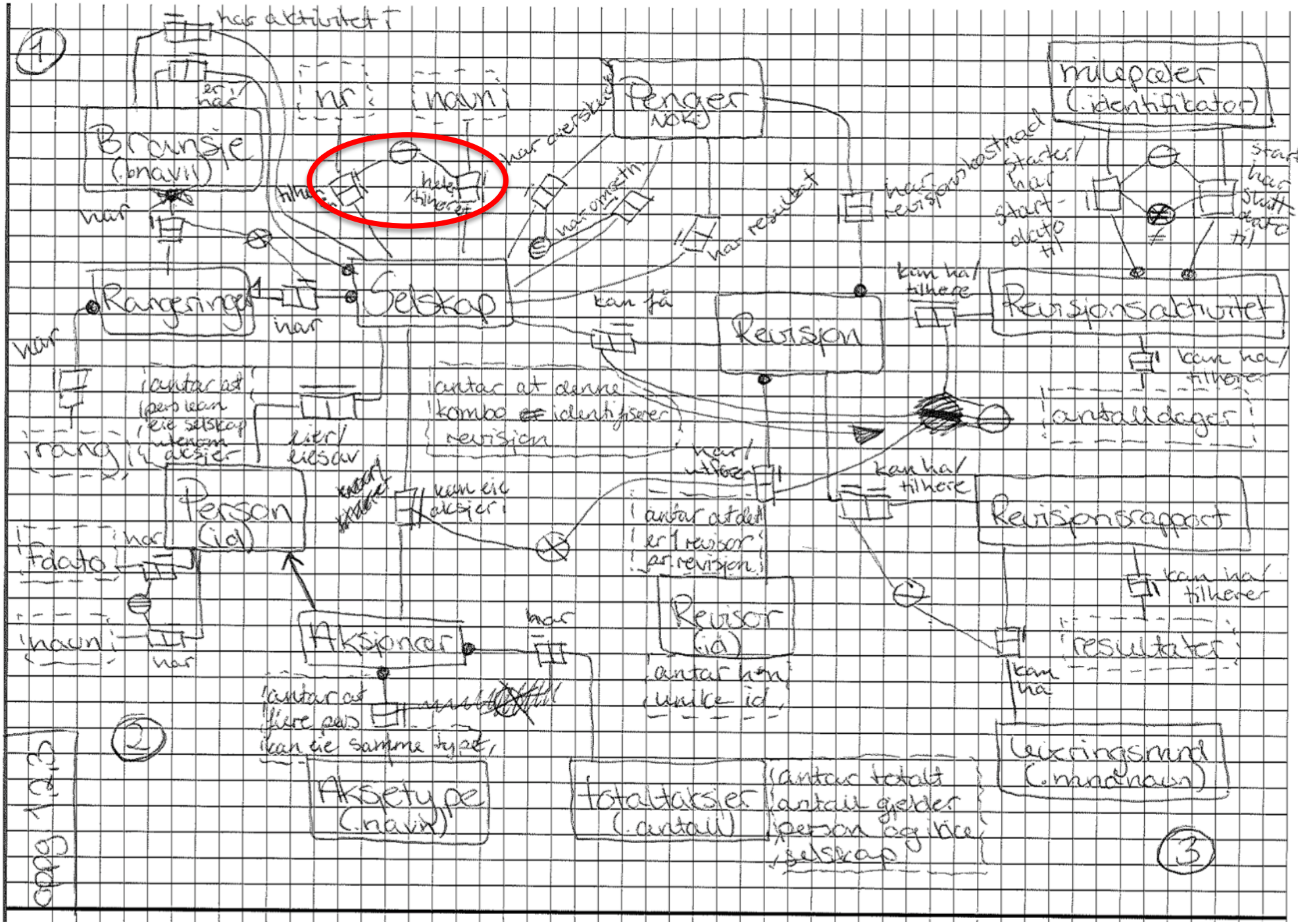
  Riktige objekter

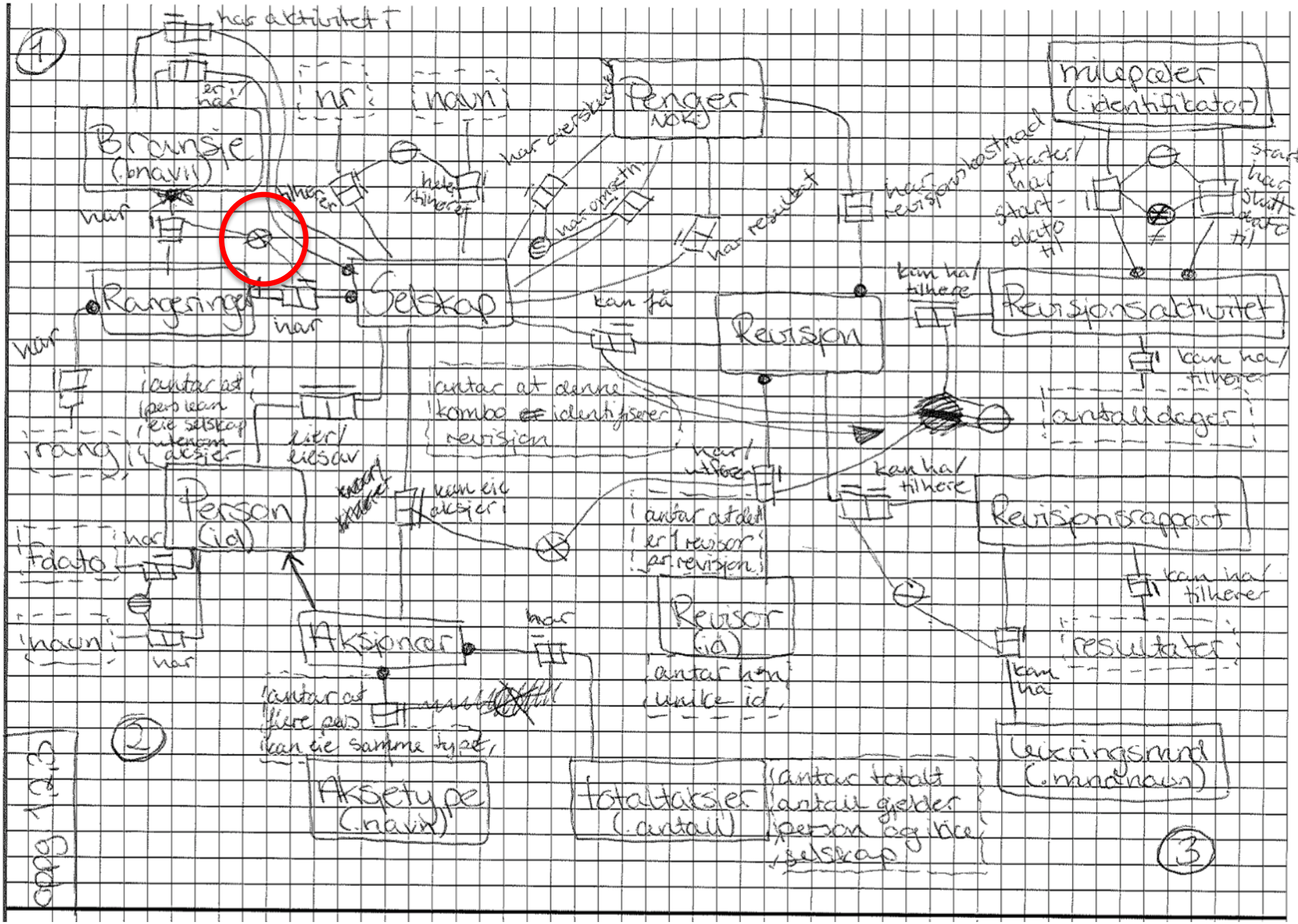
- Avanserte beskrankninger

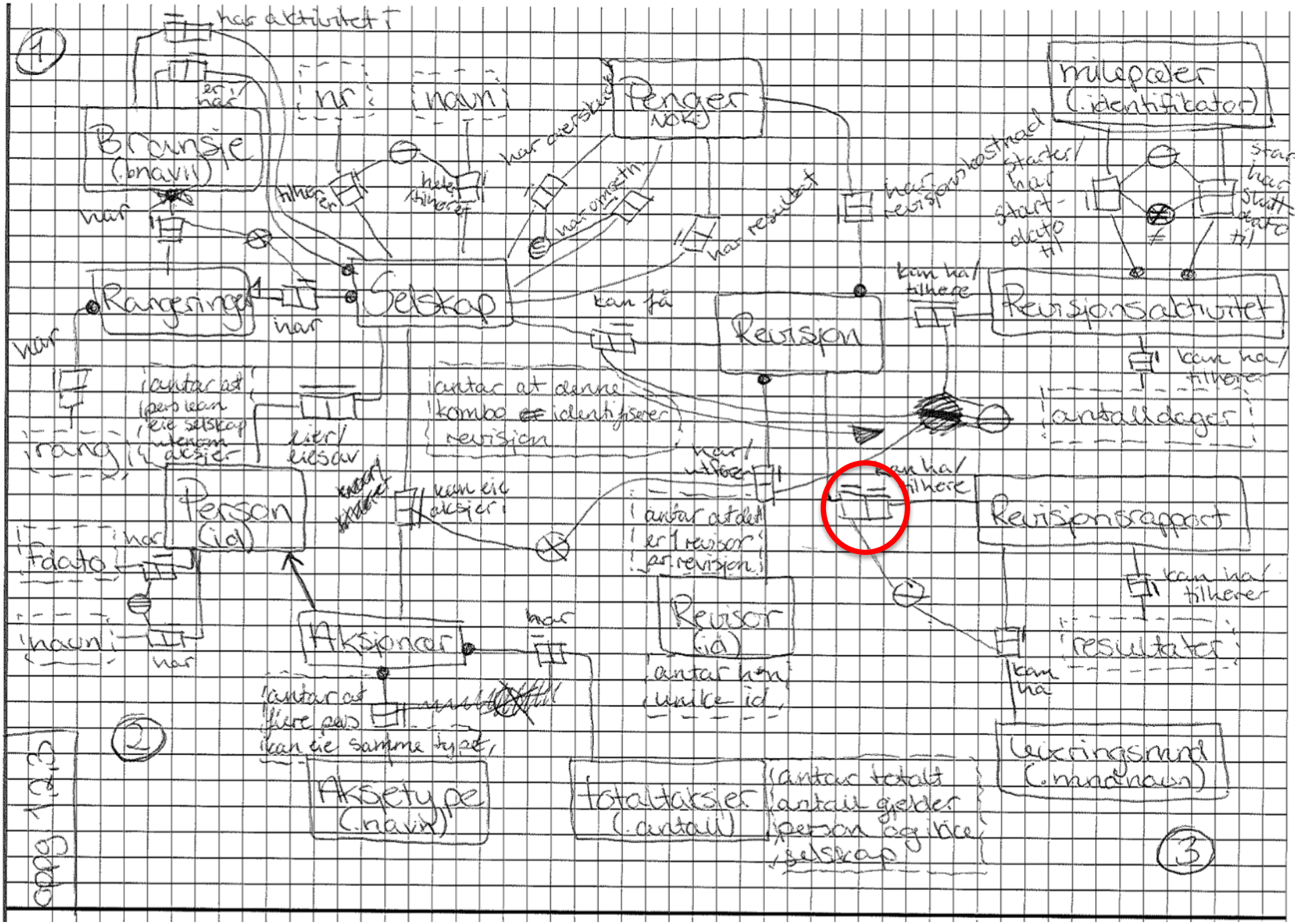
  Delmengde, exclude, total etc.

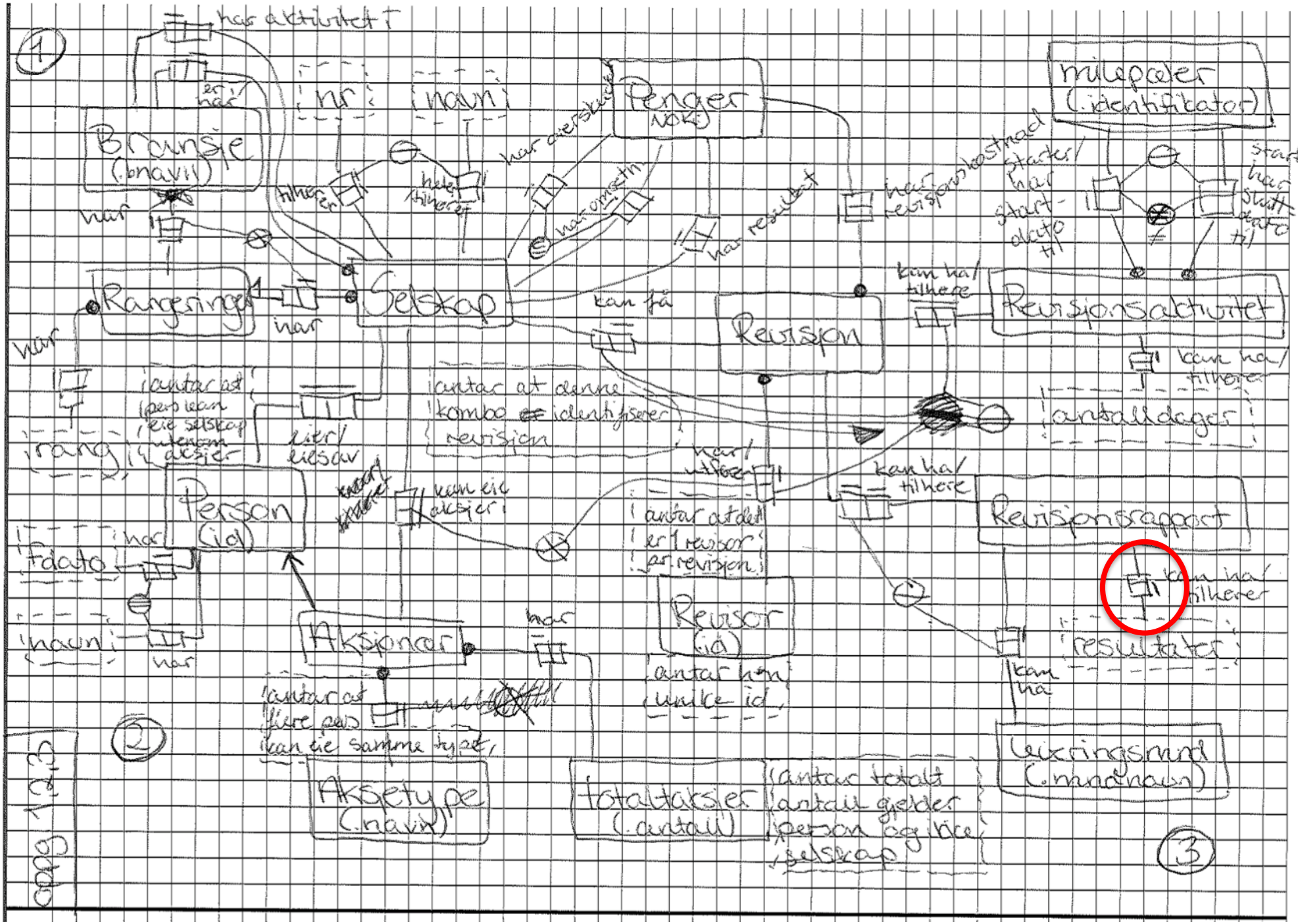
- Forståelige relasjoner

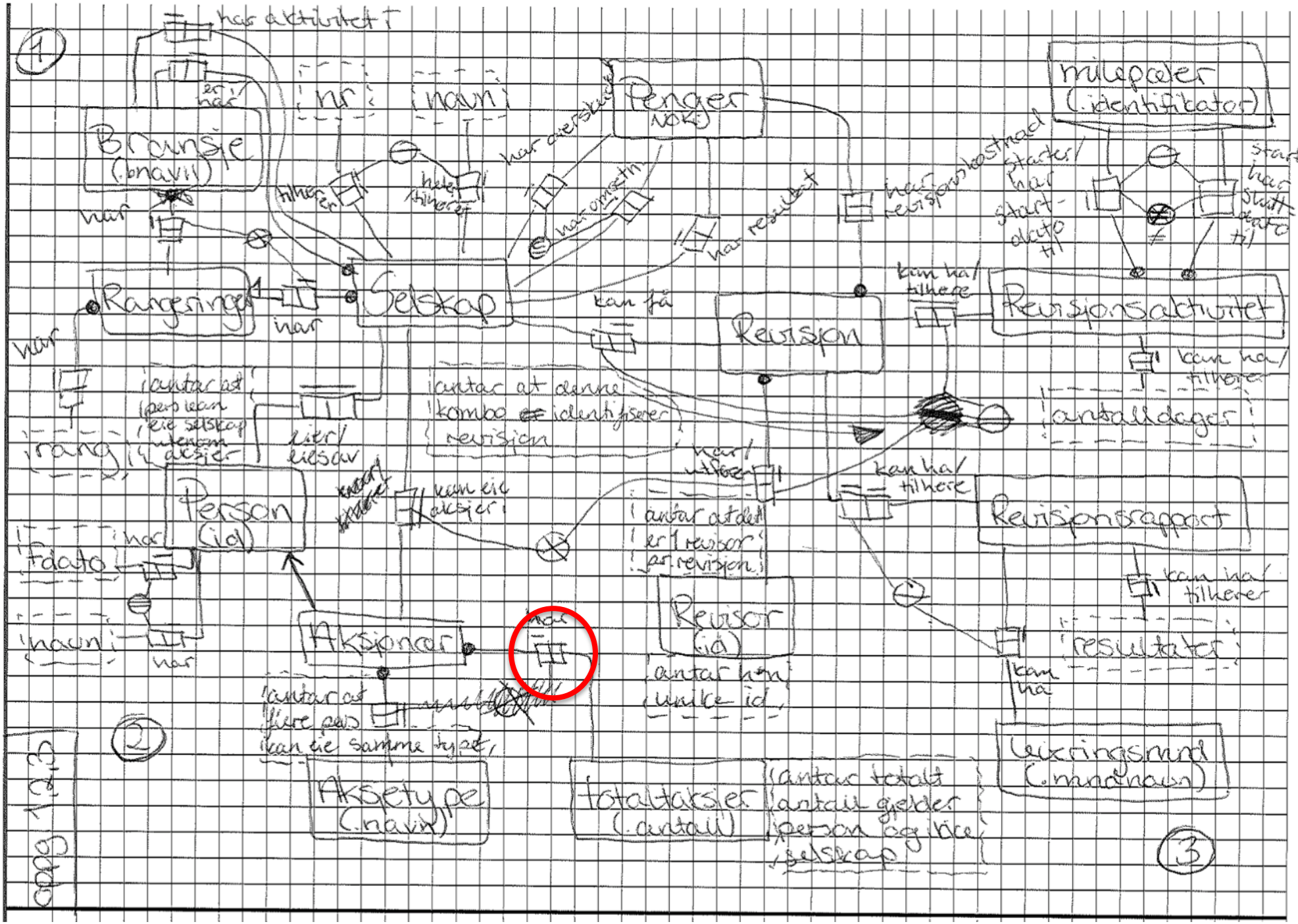
  Fyll på med tekst

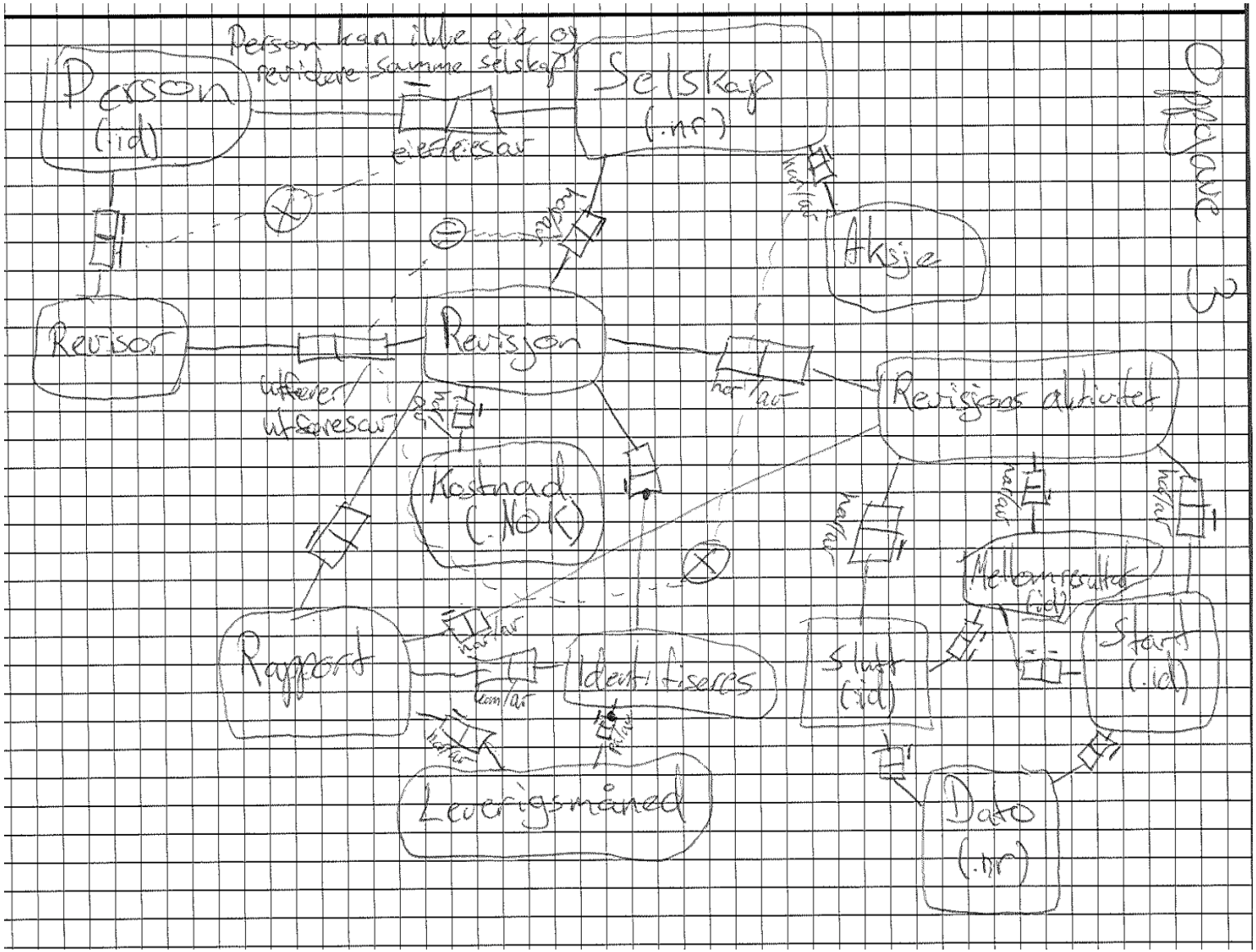


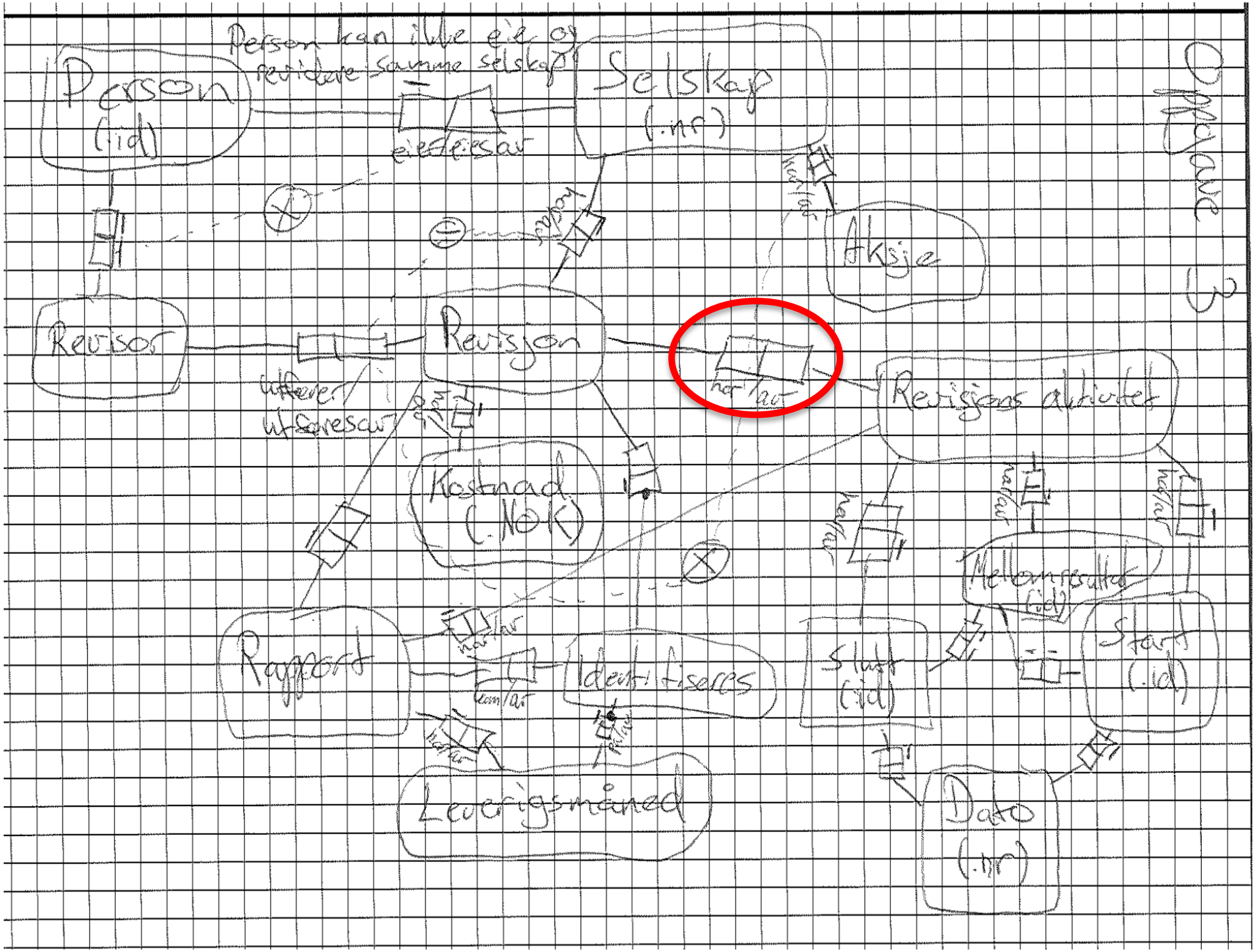




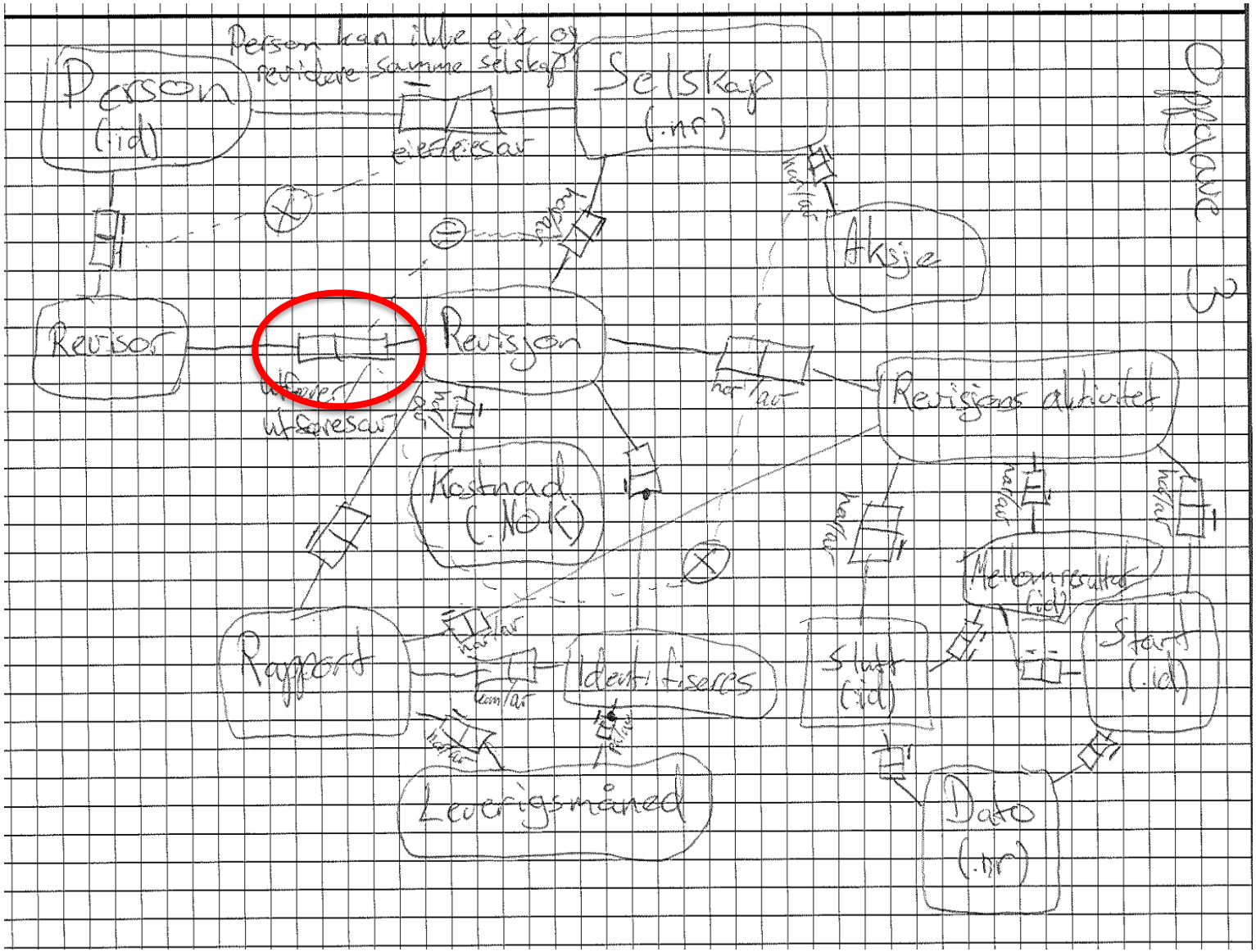


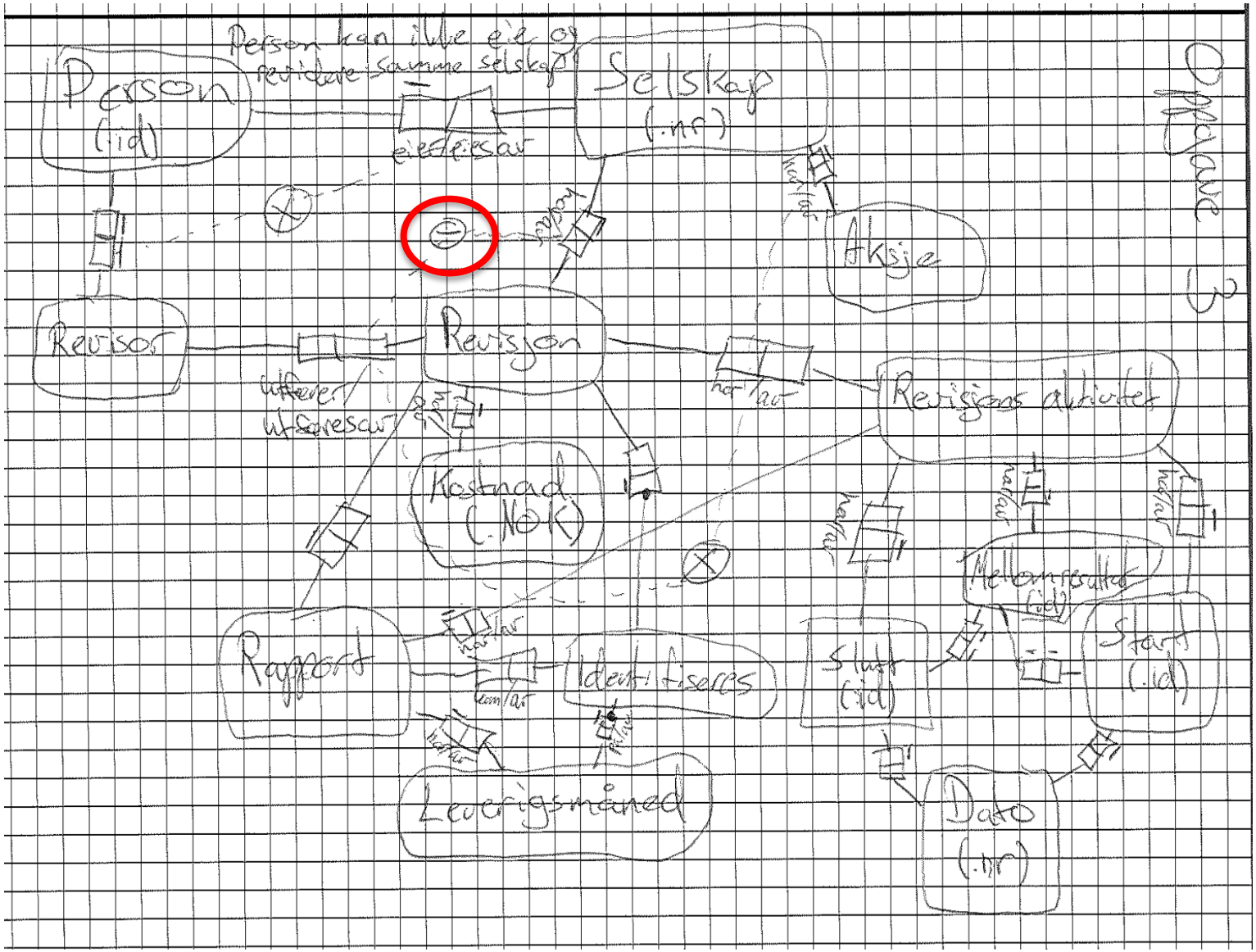


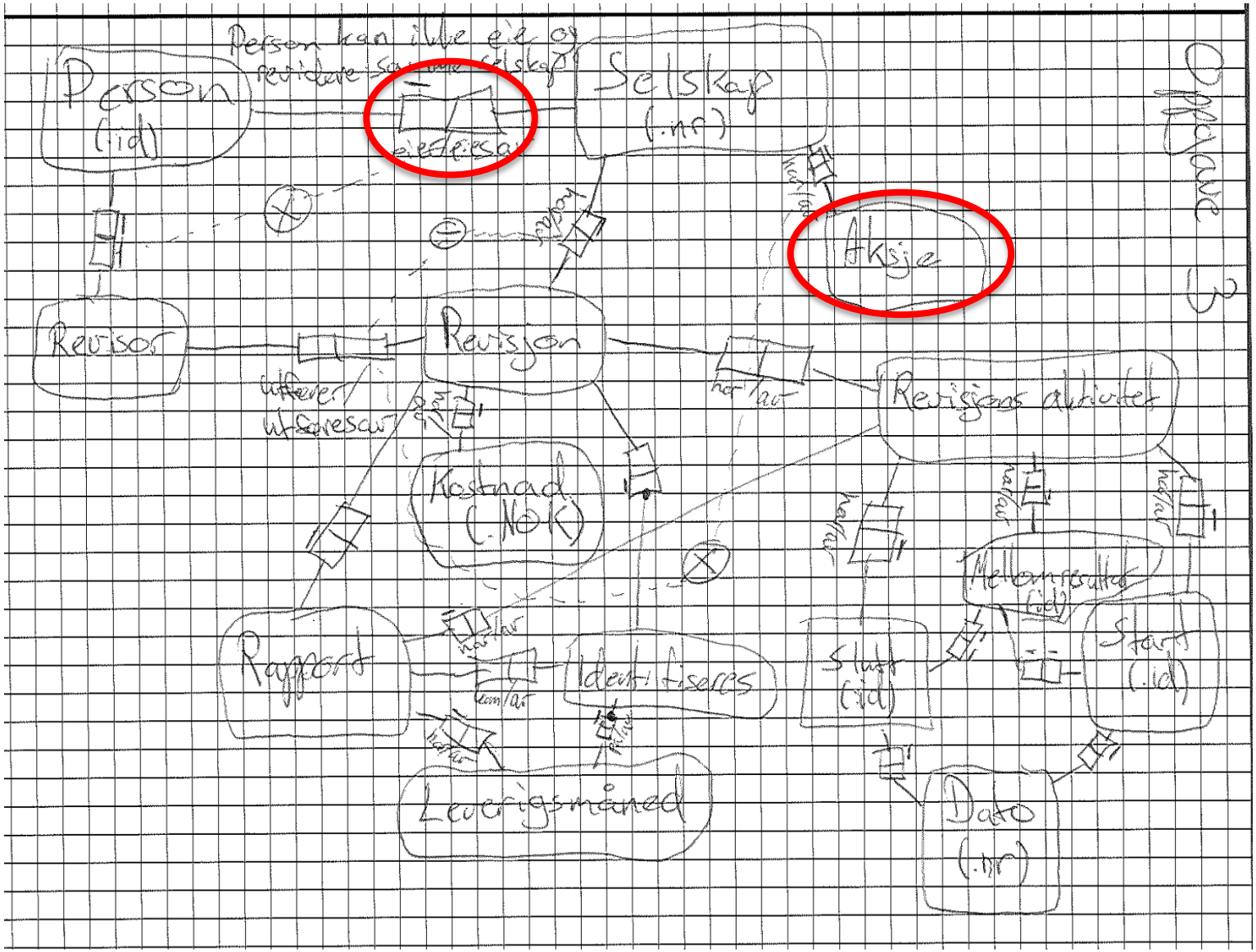

Oppgave 3
Person kan ikke eie og revidere samme selskap
Person
(.id)
eier/eies av
Selskap
(.nr)
har/er
Aksje
rapporterer
Revisor
Revisjon
utfører/
utføres av
har/er
Revisjons aktiviteter
Kostnad
(.NOK)
har/er
Mellomresultat
(.id)
Rapport
har/er
Identifiseres
Sluttresultat
(.id)
Start
(.id)
dannet av
har/er
Leveringsmåned
Dato
(.nr)

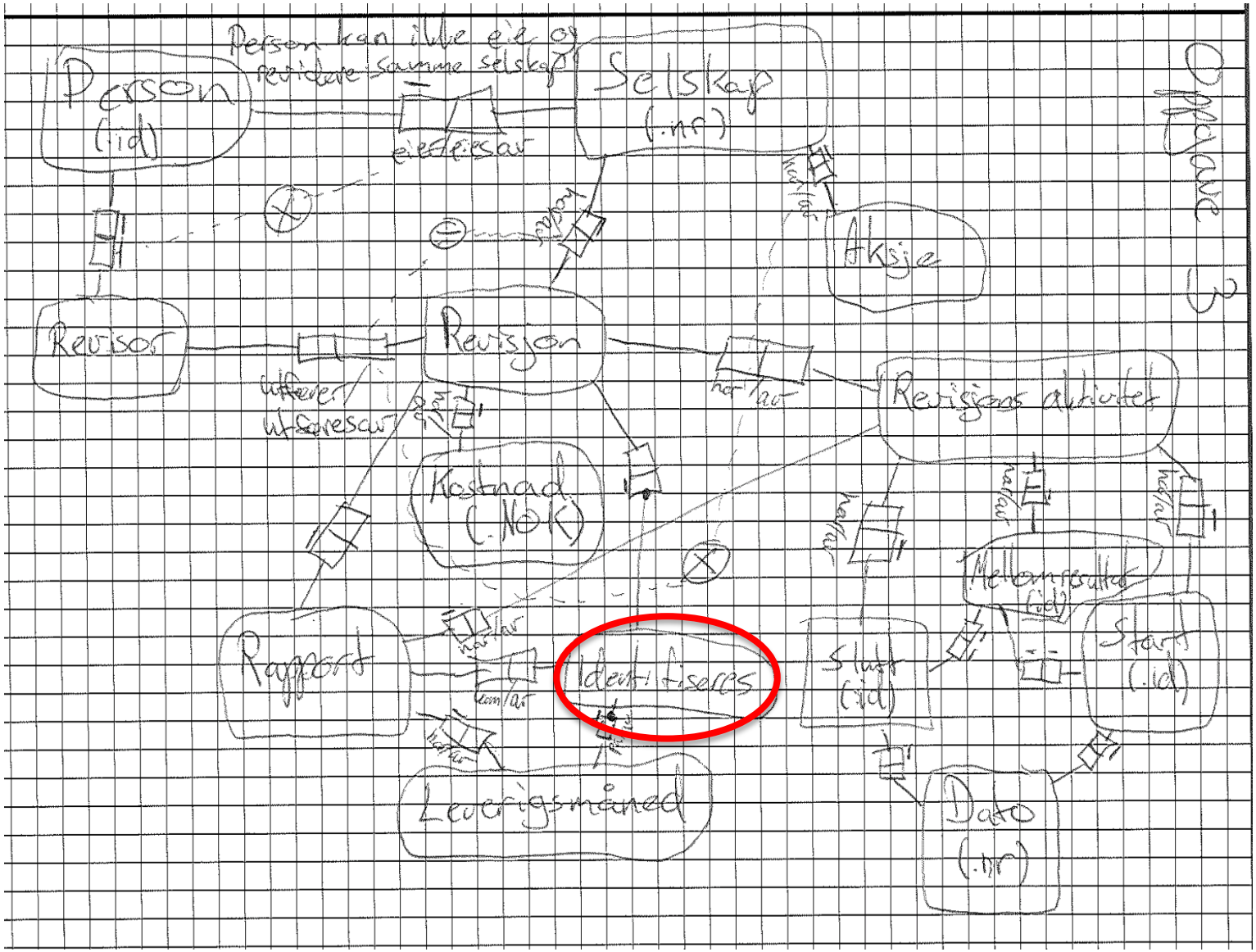


Person kan ikke eie og revidere samme selskap!
Person
(.id)
Selskap
(.nr)
eier/eies av
har/av
Aksje
Oppgave 3
Revisor
Revisjon
utfører/utføres av
Kostnad
(.NOK)
Revisjons aktivitet
har/av
Mellomresultat
(.id)
Start
(.id)
Slutt
(.id)
Rapport
Identifiseres
Leveringsmåned
Dato
(.nr)

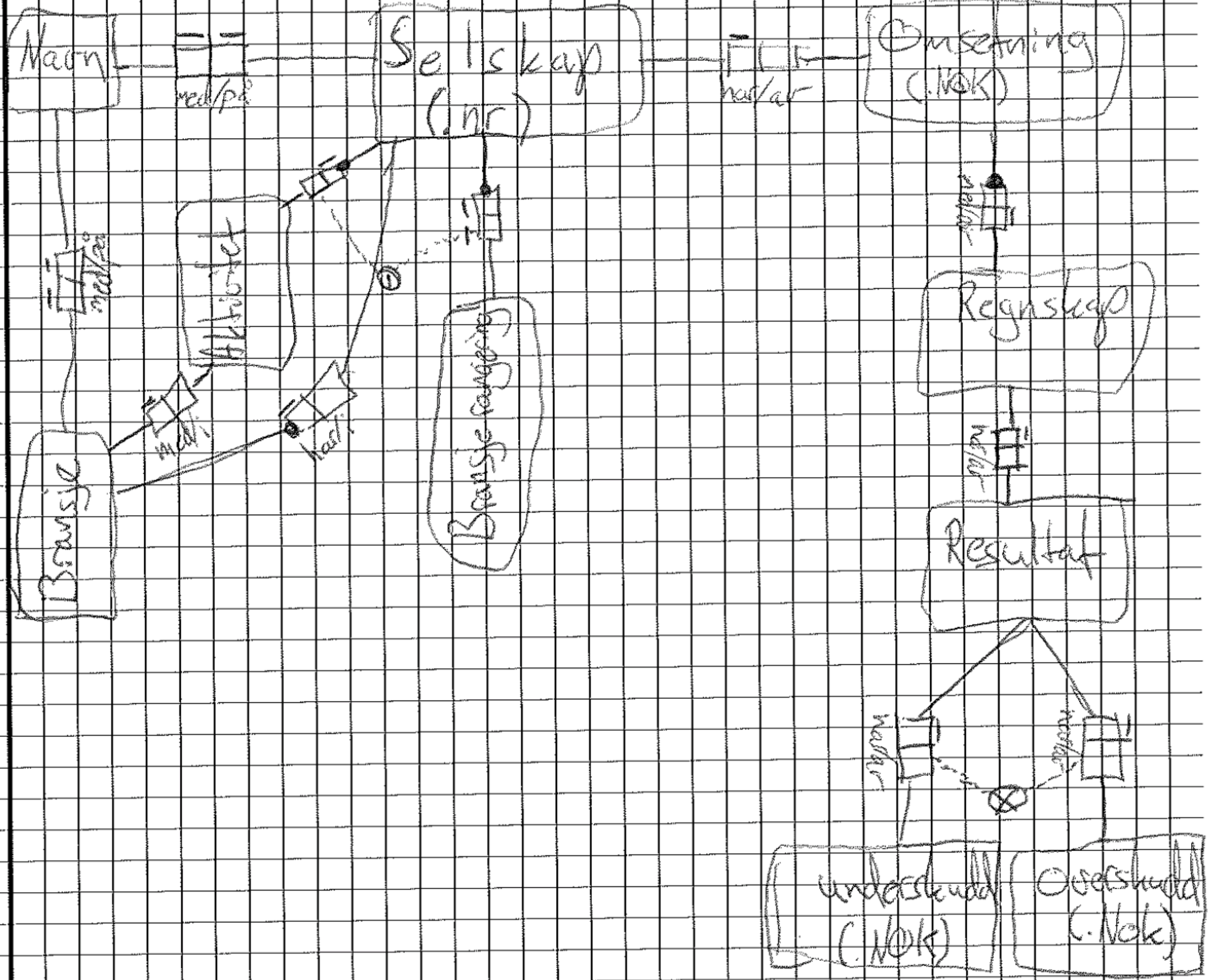

Oppgave 3
Person kan ikke eie og revidere samme selskap
Person
(.id)
Selskap
(.nr)
eier/eies av
Aksje
Revisor
Revisjon
utfører/utføres av
Revisjons aktivitet
Kostnad
(.NOK)
Rapport
Identifiseres
Leveringsmåned
Slutt
(.id)
Mellomresultat
(.id)
Start
(.id)
Dato
(.nr)


Oppgave 3
Person kan ikke eie og revidere samme selskap
Person
(.id)
Selskap
(.nr)
eier/eies av
Revisor
Revisjon
Aksje
Revisjons aktivitet
Kostnad
(.NOK)
Rapport
Identifiseres
Leveringsmåned
Mellomresultat
(.id)
Start
(.id)
Slutt
(.id)
Dato
(.nr)

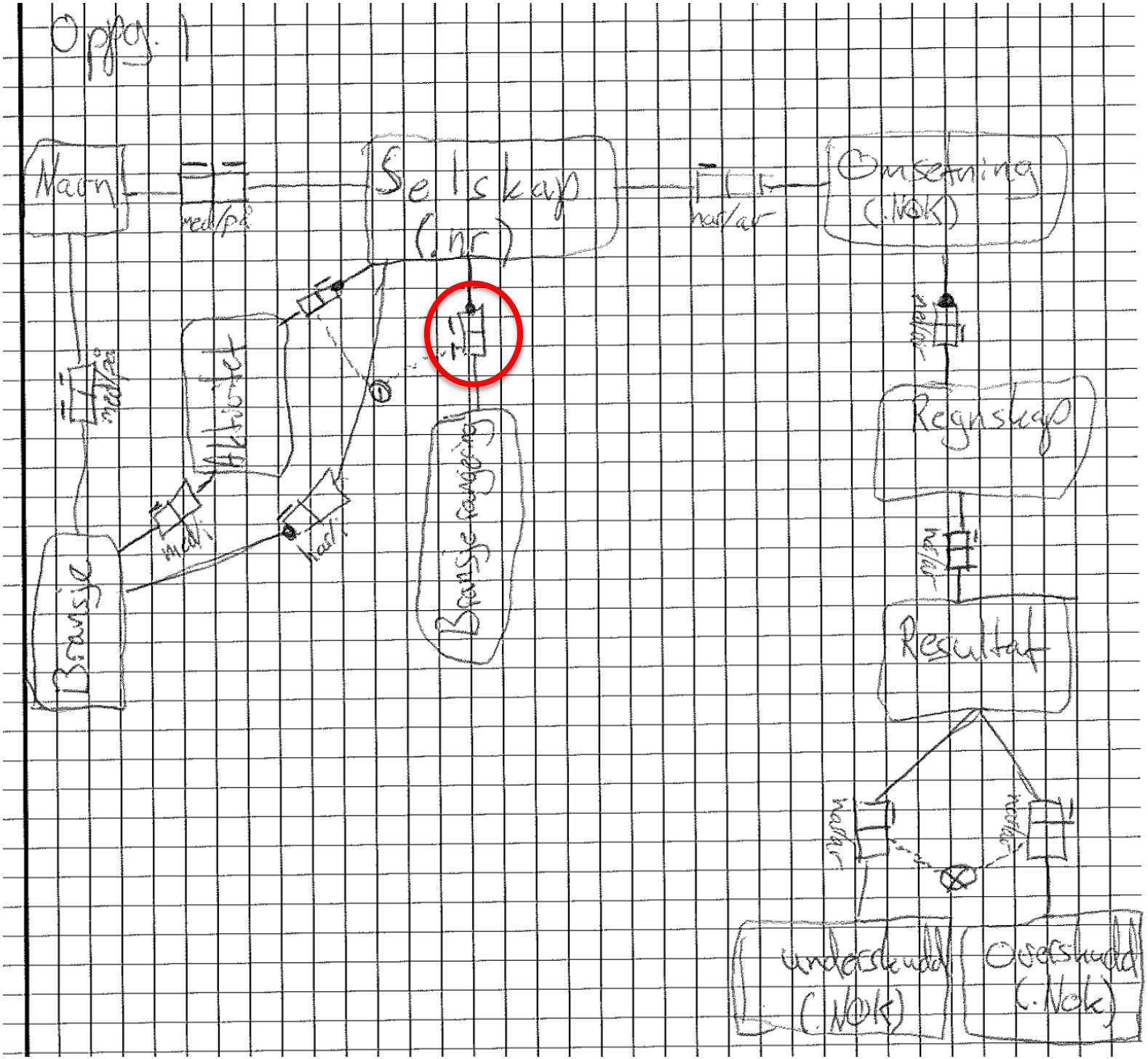

Oppgave 3
Person kan ikke eie og revidere samme selskap
Person
(.id)
Selskap
(.nr)
eier/eies av
Aksje
Revisor
Revisjon
utfører/utføres av
har/tilhører
Revisjons aktiviteter
Kostnad
(.NOK)
Rapport
Identifiseres
Leveringsmåned
Slutt
(.id)
Mellomresultat
(.id)
Start
(.id)
Dato
(.nr)

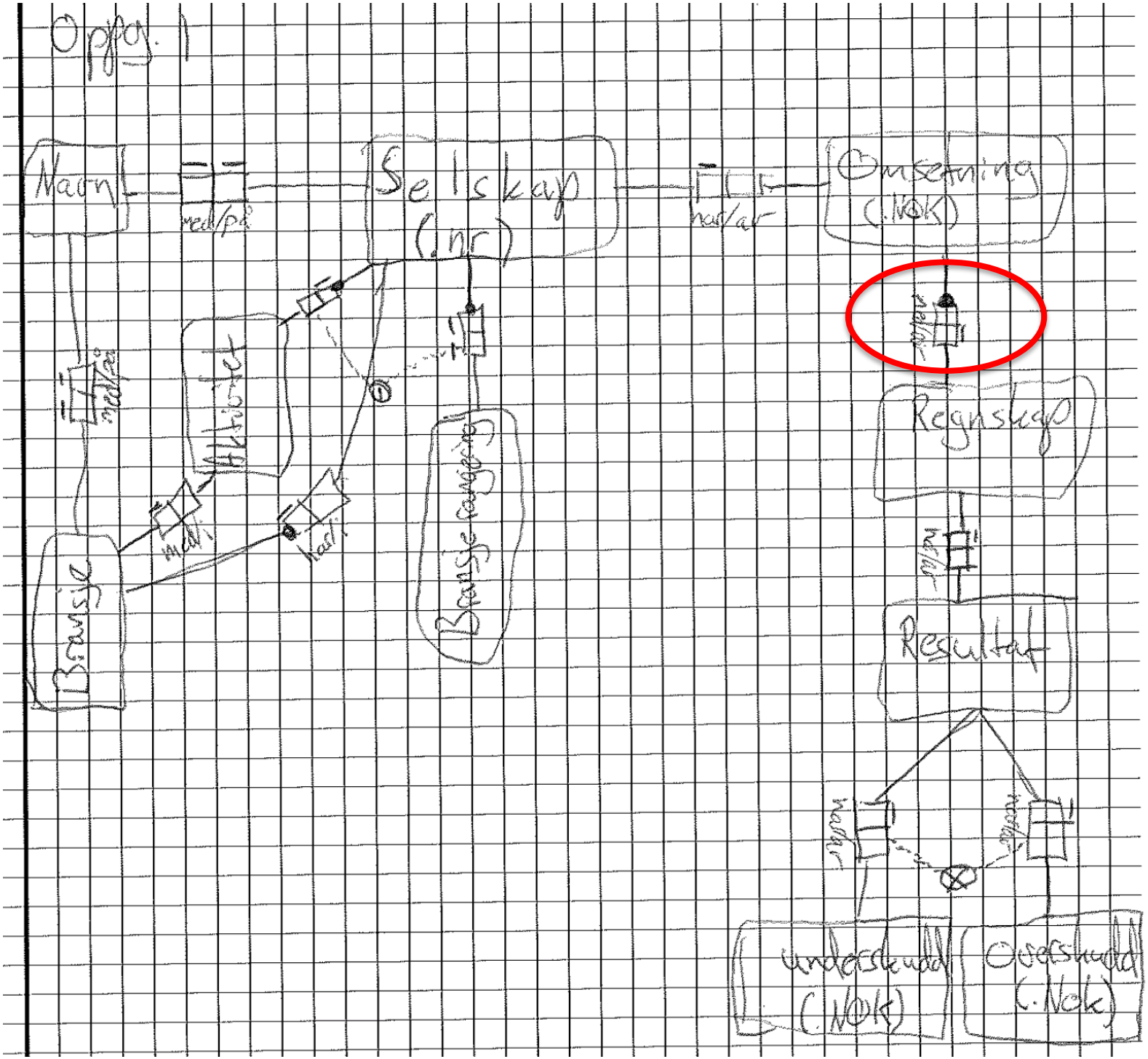

Oppgave 3
Person kan ikke eie og revidere samme selskap
Person
(.id)
Selskap
(.nr)
eier/eies av
har/er
Aksje
Revisor
Revisjon
utfører/
utføres av
har/er
Revisjons aktivitet
Kostnad
(.NOK)
har/er
Mellomresultat
(.id)
Rapport
har/er
Identifiseres
Sluttresultat
(.id)
Start
(.id)
dannet av
har/er
Leveringsmåned
Dato
(.nr)

Oppg. 1

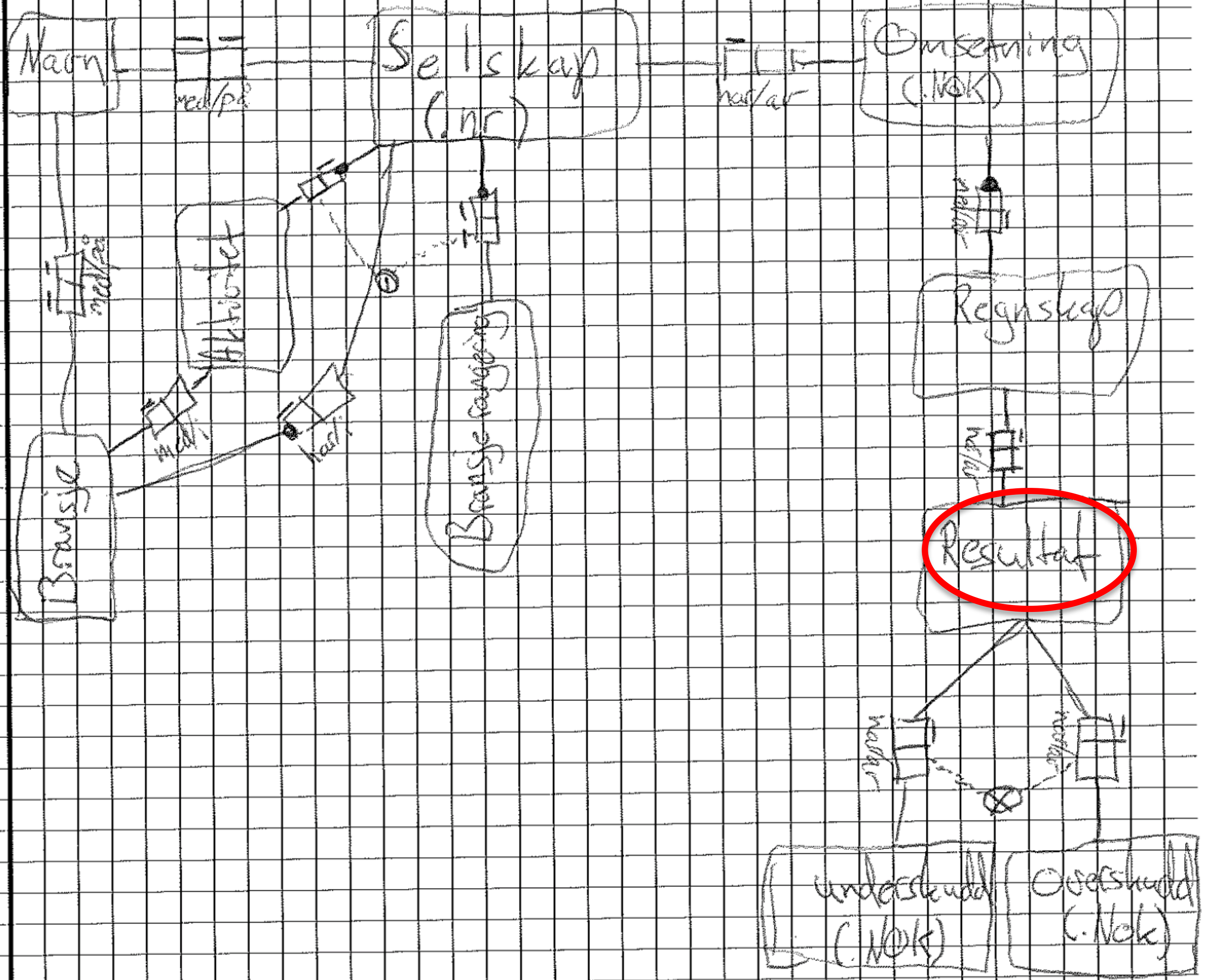


Navn
ved/på
Selskap
(.nr)
har/av
Omsetning
(.NOK)
Aktivitet
Bransje
ved/i
har/i
Bransjetangering
Regnskap
Resultat
har/av
Underskudd
(.NOK)
Overskudd
(.Nok)


Opg. 1
Naam
Selskap
(.nc)
Omsetning
(.NOK)
Regnskap
Resultat
Bransje
Bransje gjennomsnitt
Bransje sammenligning
Overskudd
(.NOK)
Overskudd
(.NOK)

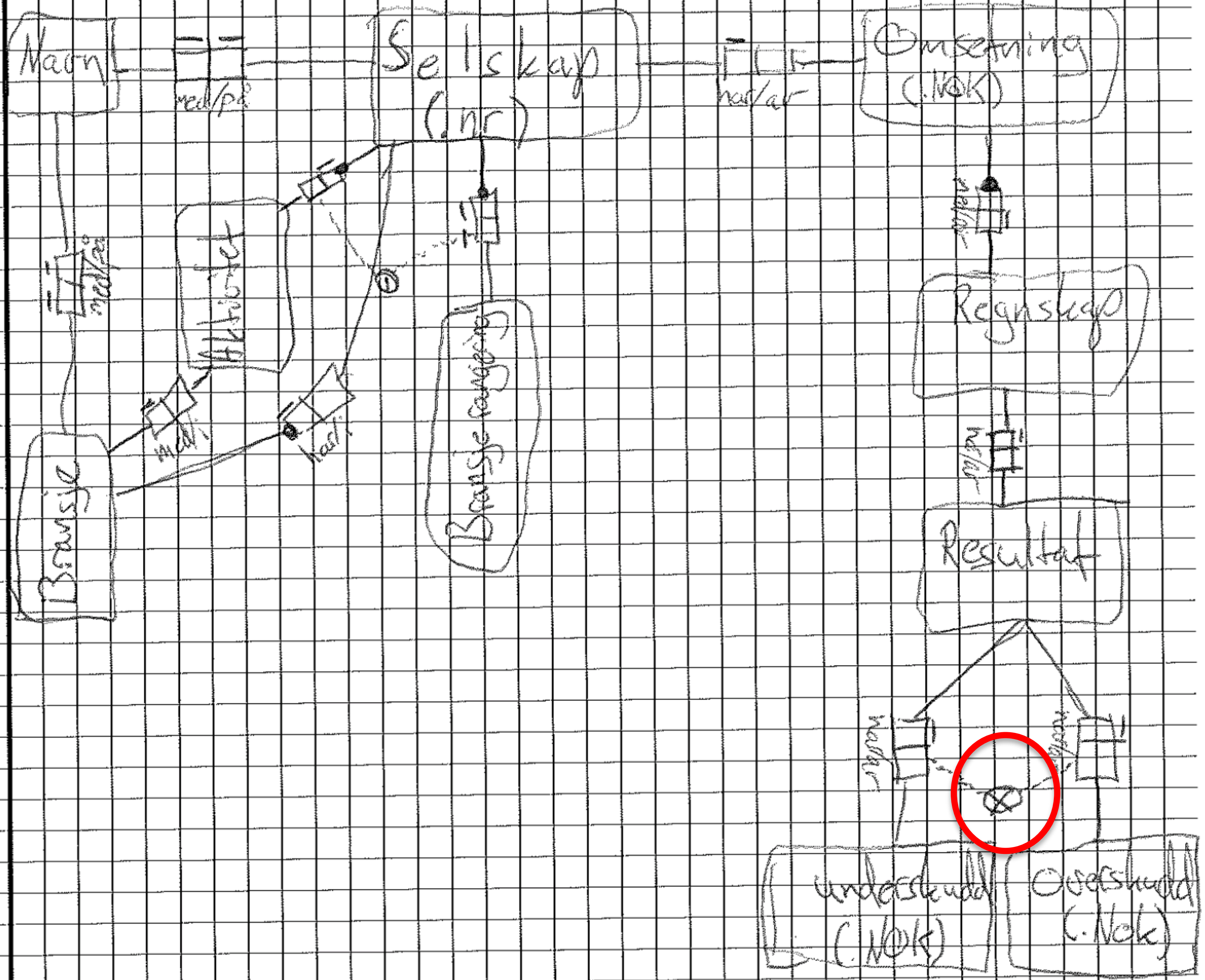

Opg. 1
Naam
Selskap
(.nc)
Omsetning
(.NOK)
Regnskap
Resultat
Bransje
Bransje samsvar
Overskudd
(.NOK)
Overskudd
(.NOK)

Oppg. 1

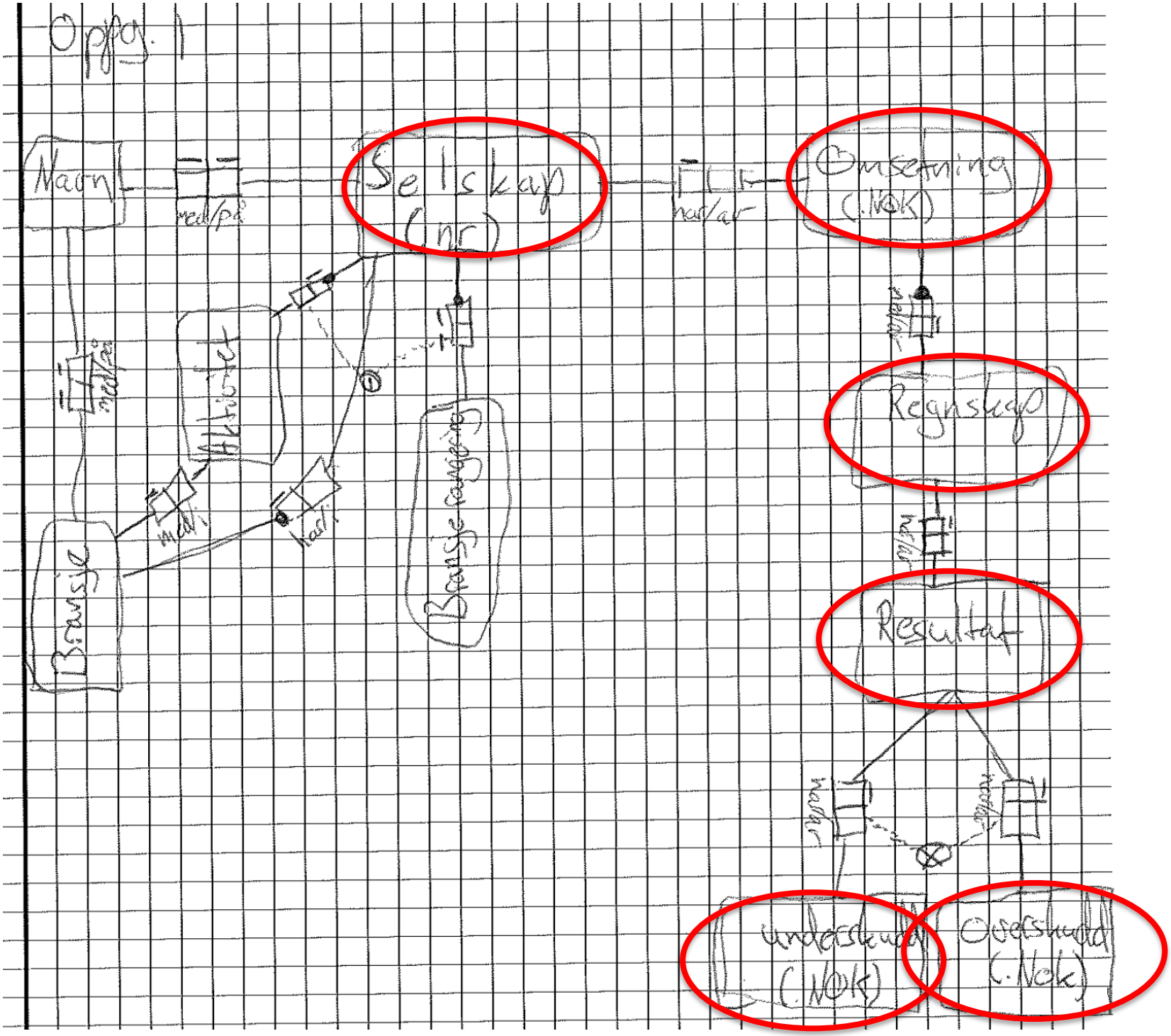


Navn
Selskap
(.nr)
Omsetning
(.NOK)
Aktivitet
Bransje
Bransjefangering
Regnskap
Resultat
Underskudd
(.NOK)
Overskudd
(.Nok)

Oppg. 1



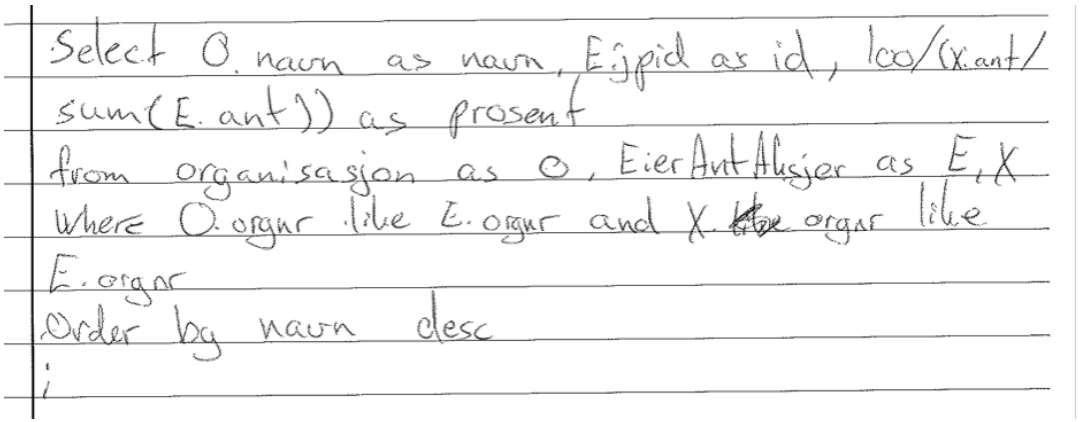
Navn
Selskap
(.nr)
Omsetning
(.NOK)
Aktivitet
Bransje
Bransjetangering
Regnskap
Resultat
Underskudd
(.NOK)
Overskudd
(.Nok)


Opg. 1
Naam
Selskap
(nr)
Omsetning
(NOK)
Regnskap
Resultat
Beholdning
Beholdning
Overskudd
(NOK)

Eksamenstips - SQL

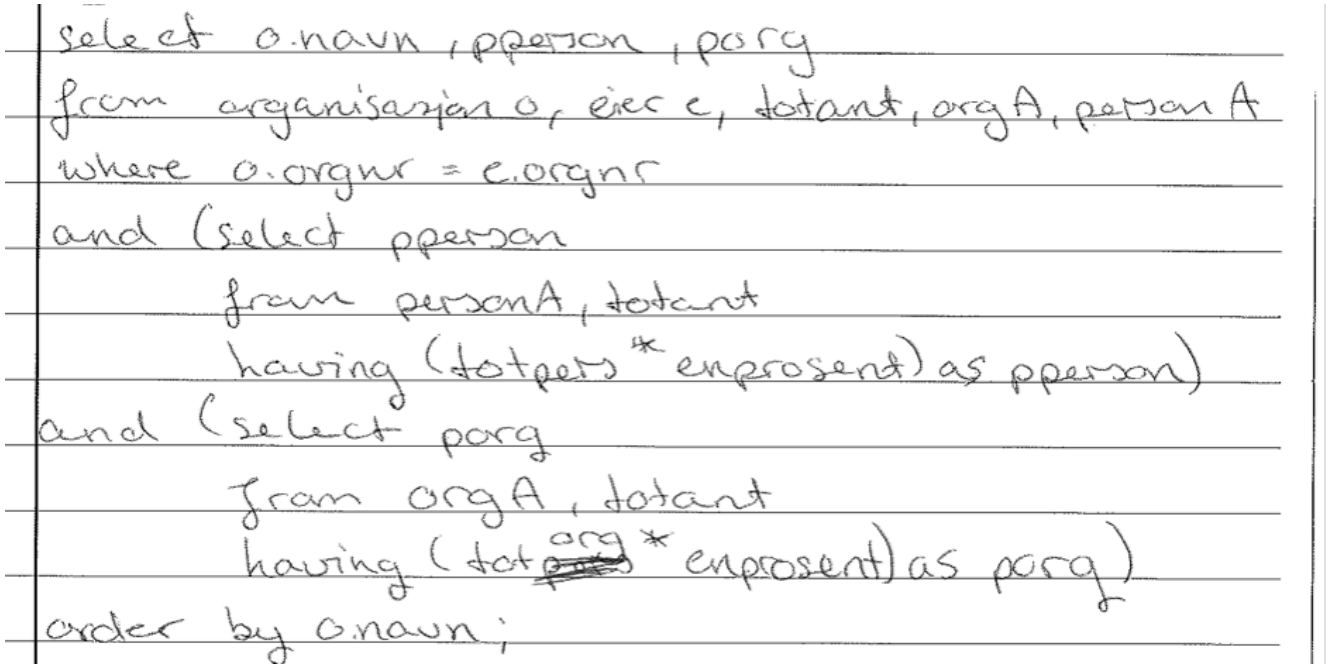
- Lesbarhet
- Syntax
- Husk select .. from .. where
- Ofte er det slik at en gitt oppgave skal teste dere i én bestemt ferdighet (group by, having, union, outer join etc)
- Ikke gjør svarene unødig kompliserte
- Bruk gjerne view for å komme fram til svaret dersom dere trenger flere trinn

Det er ikke alt som er like enkelt å forstå (for sensor)



```
Select O.navn as navn, Eigpid as id, 100/(X.ant/
sum(E.ant)) as prosent
from organisasjon as O, EierAntAksjer as E, X
where O.orgnr like E.orgnr and X.~~like~~ orgnr like
E.orgnr
Order by navn desc
;
```

Det er ikke alt som er like enkelt å forstå (for sensor)



```
select o.navn, pperson, porg
from organisasjon o, eier e, totant, orgA, personA
where o.orgnr = e.orgnr
and (select pperson
        from personA, totant
        having (totpers * enprosent) as pperson)
and (select porg
        from orgA, totant
        having (totorg * enprosent) as porg)
order by o.navn;
```

Takk for oppmerksomheten!