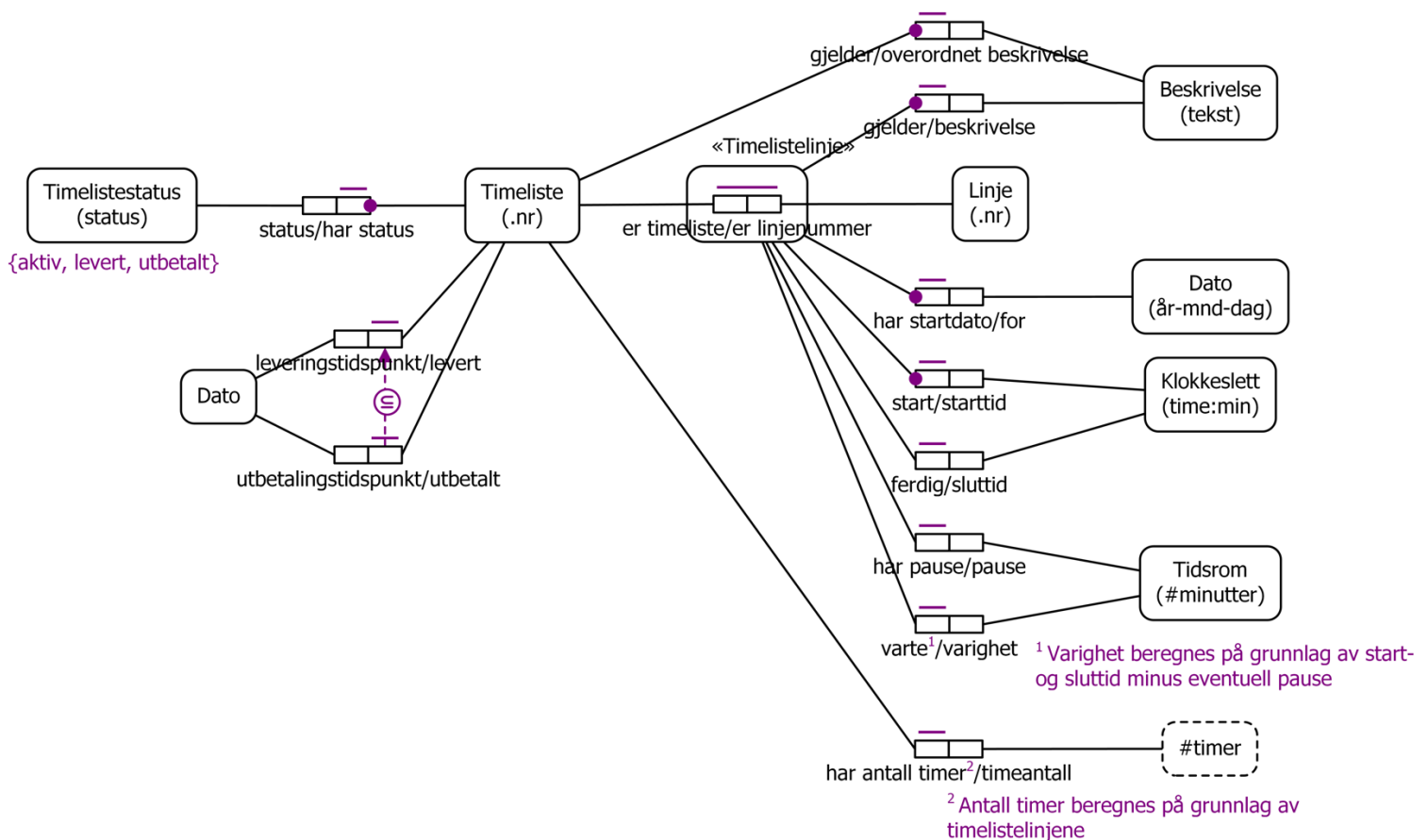


Innleveringsoppgave 3

En driftig timeansatt sommerhjelp ved universitetet i Uqbar var lei av å skrive timelister for hånd. Han har laget en liten database for å hjelpe seg. Idéen hans var å samle alle opplysninger han skulle rapportere, i en liten database, for så å skrive ut en rapport fra databasen hver gang han skulle be om penger.

Han hadde bare tenkt å bruke databasen selv, så han trengte ikke å legge til rette for at det kunne være flere brukere av databasen. Derimot trengte han å vite noe om hva han hadde jobbet med til enhver tid, når han hadde jobbet, hvilke timer som allerede var rapportert og når – og dessuten gjerne når han kunne vente penger inn på konto. Etter en del grubling og tegning kom han frem til følgende ORM-modell:



Basert på denne ORM-modellen fikk han en database med følgende tabeller:

```
Timeliste(timelistenr, status, [levert], [utbetalt], beskrivelse)
Timelistelinje(timelistenr, linjenr, startdato, starttid, [sluttid], [pause], beskrivelse)
```

Legg merke til at ORM-modellen har to faktatyper med anmerkninger om at verdiene (forekomstene) beregnes på grunnlag av verdiene i andre faktatyper. Vi har ikke gjennomgått hvordan dette skal realiseres, men én måte å gjøre det på, er ved hjelp av SQL

views. Så varigheten av en timelistelinjeaktivitet beregnes fra verdiene i Timelistelinje ved hjelp av et view (som kan benyttes i SQL-spørringer på samme måte som vanlige tabeller)

Varighet(timelistenr, linjenr, varighet)

Tilsvarende beregnes timeantall for hver timeliste fra verdiene i Varighet ved hjelp av et view

Timeantall(timelistenr, timeantall)

For å kunne vurdere om databasen kan være nyttig, har du bestemt deg for å prøve databasen selv.

Oppgave 1. Opprett en database på ditt eget databaseområde ved å

- i. kopiere filene `timelistedb.sql`, `timeliste.txt` og `timelistelinje.txt` til et sted på ditt eget område (klikk på lenkene over, så får du tak i filene),
- ii. logge inn på PostgreSQL med `psql -h dbpg-ifi-kurs -U brukernavn`, og
- iii. kjøre kommandoen `\i timelistedb.sql`.

Oppgave 2. Bruk SQL til å finne ut

1. hvilke timelistelinjer som er lagt inn for timeliste nummer 3
2. hvor mange timelister det er
3. hvor mange timelister som det ikke er utbetalt penger for
4. ved hvor mange tilfeller klokka passerte midnatt mens vedkommende var på jobb
5. antall timer som det ikke er utbetalt penger for
6. hvor mange timer det totalt ble jobbet i juli.

Oppgave 3. Legg inn data om følgende ved hjelp av SQL insert-setninger:

| Timeliste | | | | |
|-------------|----------|------------|--------------|----------------------------|
| timelistenr | status | levertdato | utbetaltdato | beskrivelse |
| 8 | utbetalt | 2016-07-29 | 2016-08-10 | Planlegging av neste trinn |
| 9 | levert | 2016-08-03 | | Detaljering av neste trinn |
| 10 | aktiv | | | Skriving av rapport |

| Timelistelinje | | | | | | |
|----------------|---------|------------|----------|---------|-------|----------------|
| timelistenr | linjenr | startdato | starttid | sluttid | pause | beskrivelse |
| 8 | 1 | 2016-07-25 | 10:15 | 17:30 | 50 | diskusjoner |
| 8 | 2 | 2016-07-27 | 12:45 | 14:00 | | konkretisering |
| 9 | 1 | 2016-07-27 | 15:15 | 18:45 | 70 | del1 |
| 9 | 2 | 2016-07-28 | 10:00 | 14:00 | 35 | del2 |
| 9 | 3 | 2016-07-28 | 21:00 | 04:15 | 90 | del3 |
| 9 | 4 | 2016-08-02 | 13:00 | 17:00 | | del4 |
| 10 | 1 | 2016-08-03 | 10:50 | 16:10 | 40 | kap1 |
| 10 | 2 | 2016-08-05 | 18:00 | | | kap2 |

Oppgave 4. Lag SQL-spørringer som skriver ut de dataene du la inn i oppgave 3.