



NORGES TEKNISK-  
NATURVITENSKAPELIGE UNIVERSITET  
INSTITUTT FOR DATATEKNIKK OG INFORMASJONSVITENSKAP

Faglig kontakt under eksamen:  
Terje Brasethvik  
Tlf: 73 59 34 43 / 90 95 91 85

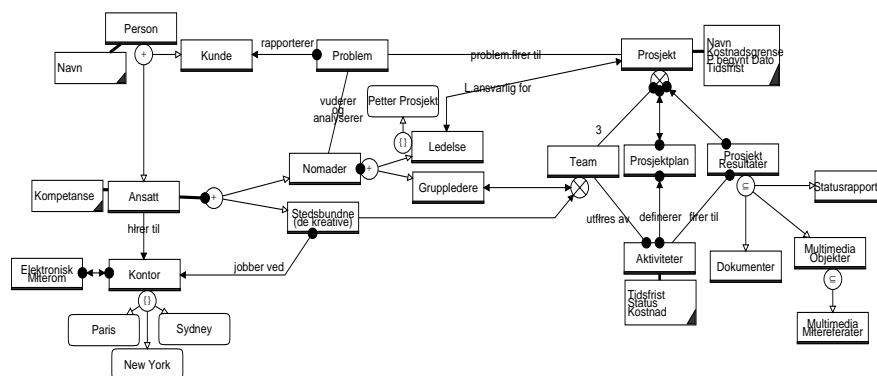
**KONTINUASJONSEKSAMEN  
I FAG 78050 (INKL 45160) SYSTEMERING 1**

Tirsdag 3. august 1999

Løsningsantydning

**Oppgave 1 – Referent Modell (20%)**

Lag en Referent-modell over de viktigste begrepene fra den "prosjektverdenen" som er beskrevet.

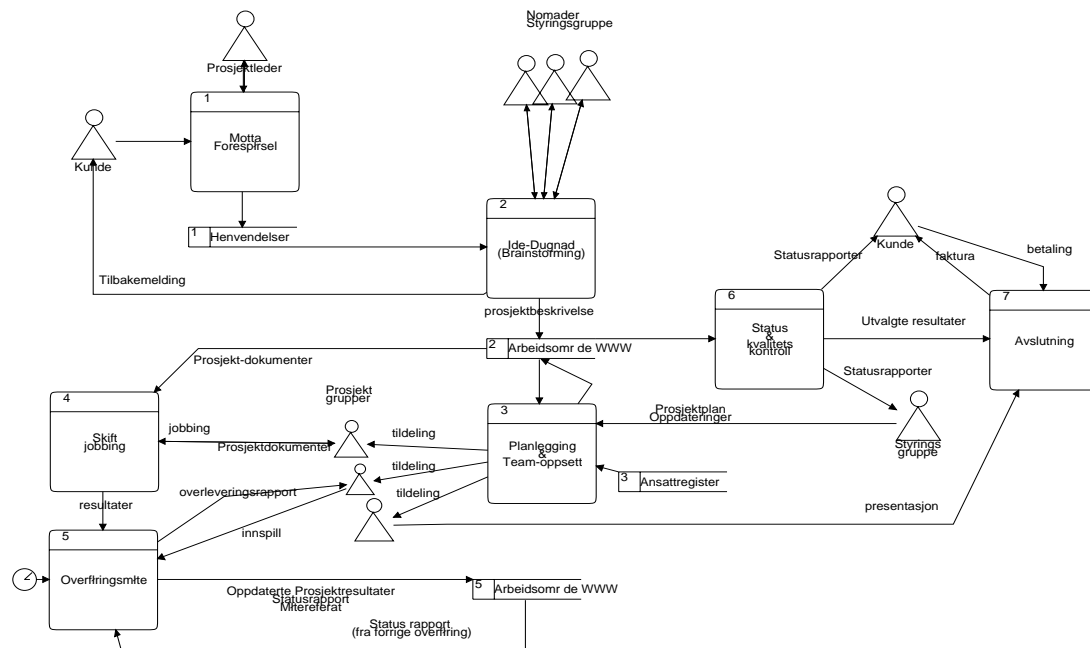


Kommentarer:

- De viktigste begrepene fra oppgavebeskrivelsen er *Kunder*, de forskjellige typer *Ansatte* og deres tilhørende *Kontor*, samt selve begrepet *Prosjekt* og de prosjekt-relaterte begrep som *Prosjekt-grupper* ("team"), *Prosjektplan*, *Aktiviter* og *Resultater*.
- Alle ansatte er enten *Nomader* eller *Stedsbundne*. I modellen er det markert at alle ansatte kan høre til et kontor, men bare de stedsbundne er knyttet til ett (og bare ett) kontor. Nomadene er igjen delt inn i *Ledelse* og *Gruppeledere* (ikke viktig for besvarelsen)
- Kundene rapporterer inn problemer som vurderes av nomadene og som eventuelt fører til et prosjekt.
- Et prosjekt er definert som bestående av 3 prosjektgrupper, en prosjektplan og et sett prosjekt resultater. Prosjektplanen definerer flere aktiviteter som utføres av prosjekt-gruppene og fører til prosjektresultater. Man kunne tenkes å utelate prosjektplanen som eget begrep i modellen og bare definere prosjektet til å bestå av aktivitetene som utføres direkte.

## Oppgave 2 – Prosessmodellering (DFD) (40%)

a) Lag et overordnet dataflytdiagram (DFD) for systemet.

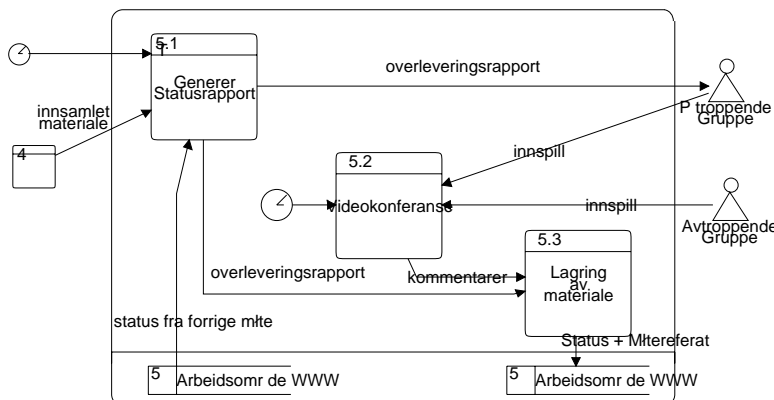


### Kommentarer:

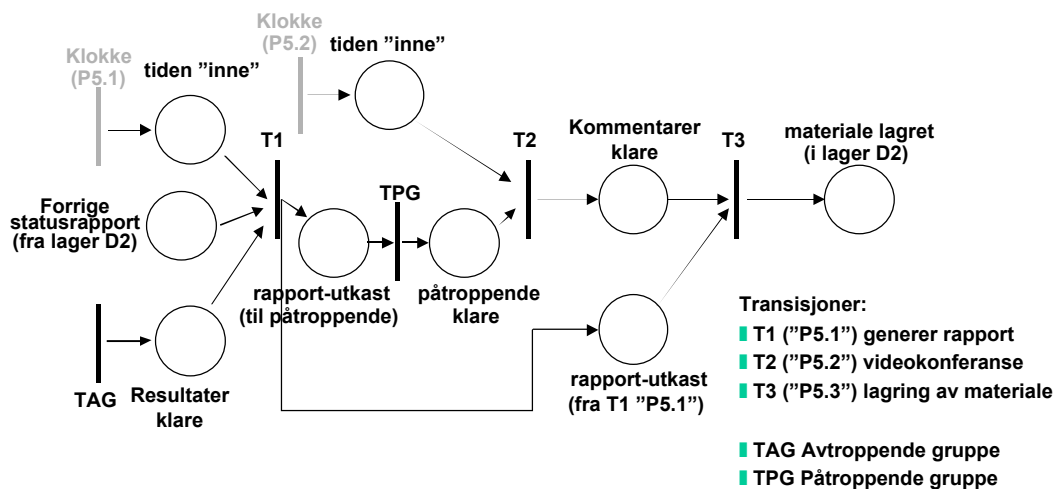
- P1: Alle henvendelser mottas av systemet, kontrolleres av en prosjektleder og lagres midlertidig.
- P2: Ide-dugnad omfatter utsendelse av henvendelsen på høring, problemanalyse, diskusjon og skriving av prosjektbeskrivelse.
- P3: Styringsgruppen definerer prosjektplanen og oppgaver tildeles til de utvalgte prosjektgrupper ("team"). Styringsgruppen kan til enhver tid lese og oppdatere prosjektplanen, noe som igjen kan føre til en ny tildeling av aktiviteter til gruppene.
- P4: Skiftjobbing med tilgang til prosjektdokumenter
- P5: Overføringsmøtet starter ved at systemet samler inn resultater fra aktiv prosjektgruppe. Vi kan tenke oss at disse sammenliknes med status fra forrige overlevering mellom disse to gruppene, og at det utfra dette genereres en overføringsrapport som sendes til påtroppende gruppe. På klokkeslett starter videokonferansen mellom de to grupper. Når møtet er ferdig oppdaterer systemet status og lagrer møtereferat i systemet.
- P6: Status og kvalitetskontroll kan utføres av kunder og styrings-grupper etter behov.
- P7: Avslutning omfatter innsamling og overlevering til kunden, samt utsendelse av faktura og mottak av betaling.
- Merk at D5 "Arbeidsområde WWW" er det samme datalager som D2, det er bare kopiert i figuren for å gi bedre layout.

b) *Dekomponer prosessen for "overleveringsmøtet" et trinn ned.*

NB! Det kunne være tvil i oppgaveteksten om hvilken prosess som skulle dekomponeres, enten avslutningsmøtet mellom to skiftgrupper eller det endelige avslutningsmøtet med overlevering av prosjektet til kunden. Ordrbruken i oppgavetekst og "case-beskrivelsen" er ikke helt entydig. Det var ment å være overføringsmøtet mellom 2 skiftgrupper, slik det er vist i modellen under. Hvilket møte studenten har valgt er imidlertid ikke viktig for evalueringen av besvarelsen. Det viktige er at dekomponeringen er korrekt i forhold til case't for den prosessen som er valgt og også konsistent i forhold til besvarelsen i a). Samme kommentar gjelder også videre i oppgavene c) og d).



c) *Lag en Petri-nett- modell tilsvarende dataflyt-diagrammet du lagde i oppgave b).*



Petrinettet som er vist er mest mulig "direkte oversatt" fra DFD-modellen i b) ved å lage en transisjon for hver prosess og ved å la de ønskede flyter bli til plasser i nettet. Vi har i tillegg lagt inn en transisjon for de eksterne agentene *avtroppende-* og *påtroppende gruppe* (agenter tilsvarer transisjoner ettersom de både kan "produsere" og "konsumere" token) som markerer at disse mottar og leser rapport utkastet fra T1 ("P5.1") før videokonferansen starter. I tillegg har vi indikert transisjoner for de to triggende klokkene som er markert i P5.1 og P5.2, men klokker og triggere er en del av PrM språket, og ikke en del av "standard" DFD notasjon.

Vi har i oversettelsen utelatt flyten "innspill" fra DFD modellen etter som en transisjon i petrinettet antas å skje på "0" tid. Vi kan dermed ikke modellere på samme måte som DFD, hvor vi har latt P5.2 pågå i 1,5 timer og hvor de enkelte grupper kommer med kontinuerlige innspill og kommentarer. Petrinettet dekomponeres i transisjonene, slik at vi for å modellere dette kunne dekomponere T2 og lage et eget nett som beskriver møtet mere i detalj.

d) *Sammenlign de to modellene fra oppgavene b) og c).*

Som nevnt kan vi rent direkte "oversette" mellom de to, ved å la flyter i DFD tilsvare plasser og la prosesser tilsvare transisjoner. Videre kan datalagre "oversettes" til plasser og agenter til transisjoner som vist i b). Imidlertid er det en del forskjeller i de to språkene, noe som gir modellene ulike egenskaper og en slik oversetting blir dermed ikke nødvendigvis helt riktig/naturlig.

- Et Petrinett er strengt formelt (matematisk) definert som bestående av plasser og transisjoner. Modellen utgjør dermed et sekvensielt transisjonsnettverk, der overgangene bestemmes av "fyringsregler". Nettet markeres med "token" for å indikere ressursbruk og samtidighet. Transisjoner kan skje når alle input-plasser har token og antas å ta null tid. En transisjon konsumerer et token fra alle input-plasser og generer et token til alle output-plasser.

Formålet med de formelle definisjonene bak petrinettet og dets fyringsregler er nettopp at modellen skal kunne simuleres og eksekveres. Petrinett modellene er dermed populære som utgangspunkt for arbeidsflytsystemer, man bruker da gjerne utvidelser av petrinett, f.eks. BNM (Behaviour Network Model) som i tillegg tillater nærmere spesifisering av ressursene og pre/post kondisjoner til firing av transisjoner.

- Selv om en DFD modell på samme måte kan sies å bestå av flyter og prosesser, så er den ikke matematisk begrunnet på samme måte som et petrinett. Vi har ingen formell definisjon av hva en prosess er, når denne kan starte, hvor lenge den pågår eller hvor mange ganger den evt. leser og sender informasjon på de respektive inn- og ut-flyer. DFD modellen inneholder ikke noe tidsperspektiv og vi kan bare delvis - og nokså tilfeldig - klare å lese sekvensen av modellen (ved å tolke prosessnavn/nr. og flytnavn).

For at en DFD modell skal kunne simuleres eller eksekveres, må den dermed tilføres en del informasjon for å få frem tidsperspektivet og sekvens, hvilken informasjon som sendes, ressursbehov etc. Et eksempel på hvordan slik informasjon kan tilføres er PrM-språket med sine timere/delays, triggere/terminatorer, porter m.m. (slik det er indikert i prosess P5.1 oppgave b)

### **Oppgave 3 – Systeminnføring (20%)**

a) *Hvilke aktiviteter inngår naturlig i innføringen av et informasjonssystem i en organisasjon.*

Eksempler på aktiviteter er:

- Installasjon av systemet
- Akseptansetest
- Konvertering av data, overføring av informasjon
- Integrasjon av systemet med omgivelser (teknologisk)
- Omlegging av arbeidsrutiner, opplæring av brukere, etablere brukerstøtte (dokumentasjon, helpdesk, ...)
- Oppfølging og etterarbeid, herunder foreberedelser av nye leveranser, vedlikehold (leveranseplanlegging), arkivering av nyttig lærdom fra prosessen etc.

Har her sett bort fra de ting som naturlig er integrert i utviklingsprosessen ellers, så som brukerdeltakelse underveis i utviklingen, solid forankring av systemet på alle nivå i organisasjonen, systemtesing osv.

- b) *Det er i pensum beskrevet spesielt noen faktorer (eller "lover") som kjennetegner utviklingen – eller livsløpet – til et system etter at det er kommet i drift i en organisasjon. Hvilke faktorer er dette og hvilke tiltak må man sette i verk for å håndtere en slik utvikling av systemet?*

Boka kap. 6.3 diskuterer hvordan datasystem utvikler seg over tid, etter at de er satt i drift. Vi kan på en måte si at det her er snakk om en slags form for "slitasje", ettersom systemet stadig må forandres, korrigeres, tilpasses og videreutvikles. Det er spesielt angitt 3 "lover" eller karakteristika ved denne utviklingen:

- Kontinuerlig forandring: Systemet vil i løpet av sin levetid gjennomgå kontinuerlig forandring, helt til det regnes som mere lønnsomt å "fryse" utviklingen og heller erstatte systemet, bygge det på nytt eller kassere det.
- "Strukturell degenerering": Etterhvert som systemet forandres og modifiseres vil det bli mere og mere ustrukturert.
- Gradvis vekst: Det er en generell trend at systemet vokser over tid.

I boka kap. 6.3.4 er det vist at en slik utvikling forsterkes og situasjonen forverres dersom man ikke setter av nok tid og ressurser til å administrere og planlegge vedlikehold systemet (de såkalte A og P type aktiviteter). I forelesningene har vi definert begrepene "Endringskontroll" og "Leveranseplanlegging" som eksempler på denne type aktiviteter.

#### **Oppgave 4 – Ytelsesvurdering (20%)**

- a) Ytelsesbetraktninger kan sies å være sentrert rundt 3 typer parametre: *Ytelsesinformasjon (P)*, *Systeminformasjon (S)* og *Belastning (W)*. Beskriv – eller gi eksempler på – hva slags informasjon det her er snakk om.

- Ytelsesinformasjon (P)
  - Prosesseringsrate [arbeid pr. tidsenhet]
    - throughput - jobber, transaksjoner, interaksjoner pr. sekund
    - Instruksjonsrate
    - Kapasitet
  - Svartid ("Responsiveness") [Tid]
    - Responstid
    - Turnaroundtid
    - Reaksjonstid
  - Utnyttelsesgrad
    - Prosent/del av ressurs som brukes
- Systeminformasjon (S)
  - Tjenester - de operasjoner som tilbys fra systemet til omgivelsene
  - Struktur - HW og SW komponenter
  - Individuelle komponenter og sub-systemer
- Last informasjon (W)
  - "Work" - de operasjoner som "pådyttes" systemet.
  - "Load" - mengde jobber som systemet må håndtere (ankomstfrekvens, parallelitet, forgreining)
  - Sekvens

b) Hva slags fremgangsmåter har vi for å få tak den type informasjon som er nødvendig for å utføre ytelsesbetraktninger ?

- Estimering
  - Statistikk
  - Beregninger
  - Kapasiteter
- HW - spesifikasjoner
  - Gjenbruk av komponenter, bruk av standard programvare
- Benchmark
  - DeFacto ”standarder”, målinger for gitte applikasjoner/situasjoner
  - Sier noe om hva man kan forvente
- Monitorering - Måling
  - HW - Tellere, Klokke
  - Snapshots, Logger, ...
- Verktøy (Bl.a. debuggere)