



Håndbok i utvikling av statistikksystemer

Med vekt på IT-metode



Håndbok i utvikling av statistikksystemer

Med vekt på IT-metode

Forord

Håndbok i utvikling av statistikkssystemer er annen - og sterkt reviderte - utgave av "Håndbok for EDB-metode". I forhold til forrige utgave har det skjedd mange vesentlige endringer. Men kanskje er den viktigste endringen en dreining i retning mot anvendelighet for de statistikkfaglige deltakerne i prosjektet. Siden IT griper inn i så å si alle deler av statistikkprosjekter i dag, omfatter nemlig denne utgaven av håndboka i prinsippet **alle deler av et statistikkprosjekt**. Selv om de kapitlene som omhandler IT fremdeles er mest gjennomarbeidet, er målgruppen for håndboka alle som arbeider med utvikling av nye statistikkprosjekter.

Noe av det viktigste Statistisk sentralbyrå venter å oppnå ved å utgi denne håndboka er at når et prosjekt er avsluttet foreligger det en god dokumentasjon av prosjektet. All **dokumentasjon** lagres elektronisk og samlet på et fastsatt sted slik at vi lett kan finne fram til den. På lengre sikt vil dette gi en mer ensartet dokumentasjon av statistikken i Statistisk sentralbyrå enn det vi til nå har sett. Det faktum at alle etterhvert vil komme til å bruke de samme arbeidsteknikkene vil føre til at vi raskere og mer nøyaktig forstår hverandre. **Kommunikasjonen** blir derfor bedre - både mellom IT-medarbeidere, mellom statistikkfaglige medarbeidere og på tvers av fagområdene IT og statistikk.

Den arbeidsmåten håndboka anbefaler (gjøre de riktige tingene, samarbeid, dokumentasjon, sjekklister for hva som må gjøres og så videre) gir dessuten en sikring av **kvaliteten** på produktet.

Håndboka er et resultat av et samarbeid i Statistisk sentralbyrå. En rekke personer både i statistikkmiljøet og i IT-miljøet har bidratt. Metoden er behandlet og vedtatt i IT-utvalget i mai 1998.

Statistisk sentralbyrå
Oslo/Kongsvinger, 10. juni 1998

Svein Longva

Forord fra arbeidsgruppa

Om første versjon av håndboka

Basert på erfaringene med “*Rapport fra prosjektet metodisk systemutvikling*” ble det laget en “minimemote” i 1994. Våren 1996 ble det etablert en arbeidsgruppe bestående av Erik Sjøberg@203@Oslo, Bjørn Roar Joneid@303@Oslo, Jan Erik Wålberg@403@Kvgr, Stein Tore Dale@701@Kvgr og Svein Jåvold@710@Oslo. Denne gruppa utviklet “*Håndbok i EDB-metode*”. Denne håndboka var først og fremst rettet mot IT-personalet som jobbet med **utvikling av statistikkssystemer i Statistisk sentralbyrå**. Gruppen har siden hatt ansvaret for å utvikle og vedlikeholde metoden¹. I tillegg til metodegruppa bidro en mengde andre IT-faglige og statistikkfaglige medarbeidere til håndboka. Leiv Solheim og Rune Gløersen ivaretok synspunkter fra statistikkfaglig metode og prosjektstyring.

Endringer fra forrige versjon

I versjon 2 av metoden er **rollen til de statistikkfaglige medarbeiderne fokusert sterkere**. Fasen “Innledende arbeid ved fagseksjonen” er omdøpt til “*STAT-analyse*”, og denne fasen er utdypet og forbedret av Espen Sørensen@240@Oslo og Jon Holmøy@330@Oslo. At metoden har skiftet navn fra EDB-metode til “*Håndbok i utvikling av statistikk-systemer*” illustrerer også det økte fokus på de statistikkfaglige medarbeiderne. Bruk av Statistisk sentralbyrås **nye strategiske verktøy ORACLE/2000** er innarbeidet i metoden. Metoden støtter derfor nå både SAS og ORACLE eksplisitt. Dessuten har vi forsøkt å gjøre hele håndboka mer strømlinjeformet og enda mer preget av huskelister/eksempler og fremfor alt - **enklere terminologi**. Et meget stort fremskritt er det at det endelig er kommet et elektronisk verktøy for dokumentasjon av SSBs data (DATADOK). Her får hele SSB blant annet en fellesløsning for definisjoner av filer og variabler.

Andre håndbøker

En del forhold som angår statistikkutviklings-prosjekter er beskrevet i andre håndbøker. I disse tilfellene henvises det til disse håndbøkene (*Håndbok i prosjektstyring*, *Revisjonshåndbok*, *Håndbok for programmerere*, *Dokumenasjonsnotat for DATADOK* med flere; se referanselisten bak).

Veien videre

Den delen av håndboka som går på rent statistikk-faglige ting er mindre utviklet enn den delen som går på de IT-faglige. Det som først og fremst gjenstår i arbeidet for å gjøre denne håndboka maksimalt effektiv for bruk i Statistisk sentralbyrå er derfor å integrere statistikkfaglig metode i kapittel 4. Dette gjelder terminologi og statistisk metode (estimeringer, utvalg osv.). Dessuten bør boka gjennomgå slik at den behandler et gjennomgående eksempel fra et typisk statistikkområde.

Tilgang til håndboka

Håndboka får du tilgang til som

- en håndbok på kundestøtte
- et Word-dokument som du selv kan skrive ut fra `q:\dok\handbok\notater\metode98.doc`
- en Intranet-referanse på *SSB-INFO* under pkt. *IT*

Statistisk sentralbyrå

Oslo/Kongsvinger 10. juni 1998

Metodegruppa

¹ “*Rapport fra prosjektet Metodisk systemutvikling*, 30. mars 1992” og “*Skal SSB innføre systemutviklingsmetode?*, 1. november 1993” forteller hvorfor og hvordan SSB tok utgangspunkt i *SE-Express* (basert på “System Engineering”) og ORACLEs *CASE method*.

Innholdsfortegnelse

| | |
|--|-----------|
| 1. Innledning | 5 |
| 1.1 Om metode | 5 |
| 1.2 Om hvem som bør lese hva | 5 |
| 1.3 Om innholdet | 5 |
| 2. Oversikt | 6 |
| 2.1 Prosjektet startes | 6 |
| 2.2 Struktur, prosjektstyring og kvalitetssikring | 7 |
| 3. Prosjektinitiering | 9 |
| 3.1 Prosjektskriv og prosjektgodkjenning | 9 |
| 3.2 Opprett infrastruktur | 10 |
| 4. STAT-analyse-fasen | 11 |
| 4.1 Statistikkfaglig forarbeid | 12 |
| 4.2 Kontekstdiagram | 14 |
| 4.3 Prosjektplan | 15 |
| 4.4 Avklaring med styringsgruppa | 15 |
| 4.5 Statistikkfaglig orientert kravspesifikasjon | 16 |
| 5. IT-analyse-fasen | 21 |
| 5.1 Prototyping | 22 |
| 5.2 Dataflytdiagram | 23 |
| 5.3 Datamodellering | 25 |
| 5.4 Datadefinisjon | 28 |
| 5.5 Dialogdesign (GUI) | 29 |
| 5.6 Detaljspesifikasjon | 31 |
| 6. Gjennomførings-fasen | 33 |
| 6.1 Programmering og testing | 34 |
| 6.2 Fysisk dataflytdiagram | 35 |
| 7. Avslutnings-fasen | 36 |
| 7.1 Dokumentasjonsnotat | 38 |
| 7.2 Sjekk dokumentasjon | 41 |
| 7.3 Igangkjøring | 42 |
| 8. Oppsummering | 43 |
| 9. Ordlister | 45 |
| 10. Vedlegg: Typisk dataflyt og datalagertyper i SSB | 49 |
| 11. Vedlegg: Hvilke verktøy skal brukes til hva? | 50 |
| 12. Referanser | 50 |
| 13. Noen erfaringer etter bruk av metoden | 51 |

1. INNLEDNING

1.1 Om metode

Denne håndboka beskriver en metode for hvordan et statistikkssystem i Statistisk sentralbyrå bør utvikles. Dette gjør den ved å følge prosjektet fra *prosjektskriv*² til et ferdig levert IT-system. Vi har delt metoden i flere trinn (i det følgende kalt *faser*):

- *Prosjektoppstart* sikrer at prosjektet kommer inn i ordnede former
- *STAT-analyse* sikrer oversikt over systemet som skal lages & statistikkfaglig analyse
- *IT-analyse* sikrer detaljert spesifikasjon av det systemet som skal lages
- *Gjennomføring* sikrer testet koding av IT-systemet
- *Avslutning* sikrer at dokumentasjonen blir ferdig og at prosjektet blir ordnet avsluttet

Fasene er igjen delt inn i konkrete teknikker for å gjennomføre fasen. Hele tiden foreslås det hvordan arbeidsoppgavene kan organiseres og utføres for å få det best mulige resultat som utgangspunkt for neste trinn i arbeidet.

Nytten av noen av teknikkene er avhengige av systemets størrelse, men det vil ofte være hensiktsmessig å gjennomføre alle de foreslåtte teknikkene fordi det å gjennomgå teknikkene skaper en arbeidsform som stiller krav til at man tenker grundigere gjennom det systemet som skal lages. Og ikke minst - det genererer en god dokumentasjon av systemet.

1.2 Om hvem som bør lese hva

Trenger alle å lese hele håndboka? Nei, det er ikke nødvendig! Her er et forslag til hvem som bør lese hva:

Ledere som ønsker informasjon om hva metoden kan tilby og hvordan den er oppbygd, bør spesielt studere kapittel 2 og 8. **Prosjektledere** bør også studere kapittel 3. Dette kapitlet kan også med fordel samholdes med "*Håndbok i prosjektstyring*".

Statistikkfaglige medarbeidere fra fagseksjonene finner svært mye av det viktige stoffet i kapittel 4. Det er fagseksjonene som har ansvaret for innholdet i dette kapitlet, og kapitlet er utarbeidet spesielt med tanke på arbeidsflyten i Statistisk sentralbyrå. Men også kapittel 2, 3, 8 gir nyttig informasjon; spesielt for den som er administrativt ansvarlig på fagsiden.

Kundestøtte som har ansvar for tilrettelegging av dokumentasjon bør studere kapittel 3 (organiseringen av Q-disken mm.).

IT-kontor-lederne som har ansvaret for å kvalitetssikre resultatet fra IT-delen av prosjektet bør lese kapittel 2, 3, 7 og 8.

IT-medarbeiderne ("systemutviklerne" og spesielt den som er IT-ansvarlig for prosjektet) bør lese hele håndboka nøye og ikke minst - bruke den som oppslagsverk.

1.3 Om innholdet

Denne håndboka kan naturlig deles inn i tre deler:

- Kapittel 2 (og 8) gir en **oversikt** over metoden. Merk spesielt at kapittel 8 gir en 1-sides tekstlig beskrivelse av hele metoden og en nyttig figur-oversikt!
- Kapitlene 3-7 gir **detaljert** gjennomgang av fasene og teknikkene i metoden.
- Resten av håndboka består av utfyllende stoff (ordliste, referanser og oversikter).

² Metoden kan også brukes uten at prosjektskriv foreligger og arbeidet trenger ikke være organisert som et prosjekt.

Stoffet i kapittel 3-7 er presentert i den rekkefølgen aktivitetene skal utføres. Informasjonen i denne delen av håndboka er meget strengt strukturert; hver teknikk starter med en oversikt som gir noen sentrale fakta om teknikken:

| | |
|----------------------|--|
| Idé | Idéen som ligger bak å bruke teknikken i SSB. |
| Ambisjonsnivå | Hvilket ambisjonsnivå vi legger oss på i SSB. |
| Notat | Henvisning til eventuelt notat som er skrevet i tilknytning til denne teknikken. |
| Mal | Angivelse av hvor malen (eksempelet) finnes lagret på elektronisk form. |
| Ansvarlig | Hvem som er ansvarlig for resultatet av denne teknikken. |
| Utførende | Hvem som bruker teknikken og legger frem resultatet. |

Deretter følger noen punkter om hvordan teknikken konkret brukes:



Beskrivelse:

Gir en beskrivelse av hvordan teknikken utføres



Erfaringer:

Oppsummerer erfaringer som er gjort med denne teknikken i SSB.



Teknisk merknad:

En mer teknisk anmerkning om denne teknikken.

Eksempel

Et eksempel på bruk av teknikken anvendt i SSB. I tilknytning til denne er det listet opp hvilke symboler som er lovlige i teknikken.

- Det utfyllende stoffet består av vedlegg som består av komprimerte beskrivelser av:
 - et typisk IT-system i Statistisk sentralbyrå
 - ordliste som gir en tekstlig beskrivelse av mye brukte metodiske uttrykk.
 - verktøy som brukes til teknikkene
 - referanseliste som gir mer nøyaktige referanser til notater som har tilknytning til metoden.

2. OVERSIKT

2.1 Prosjektet startes

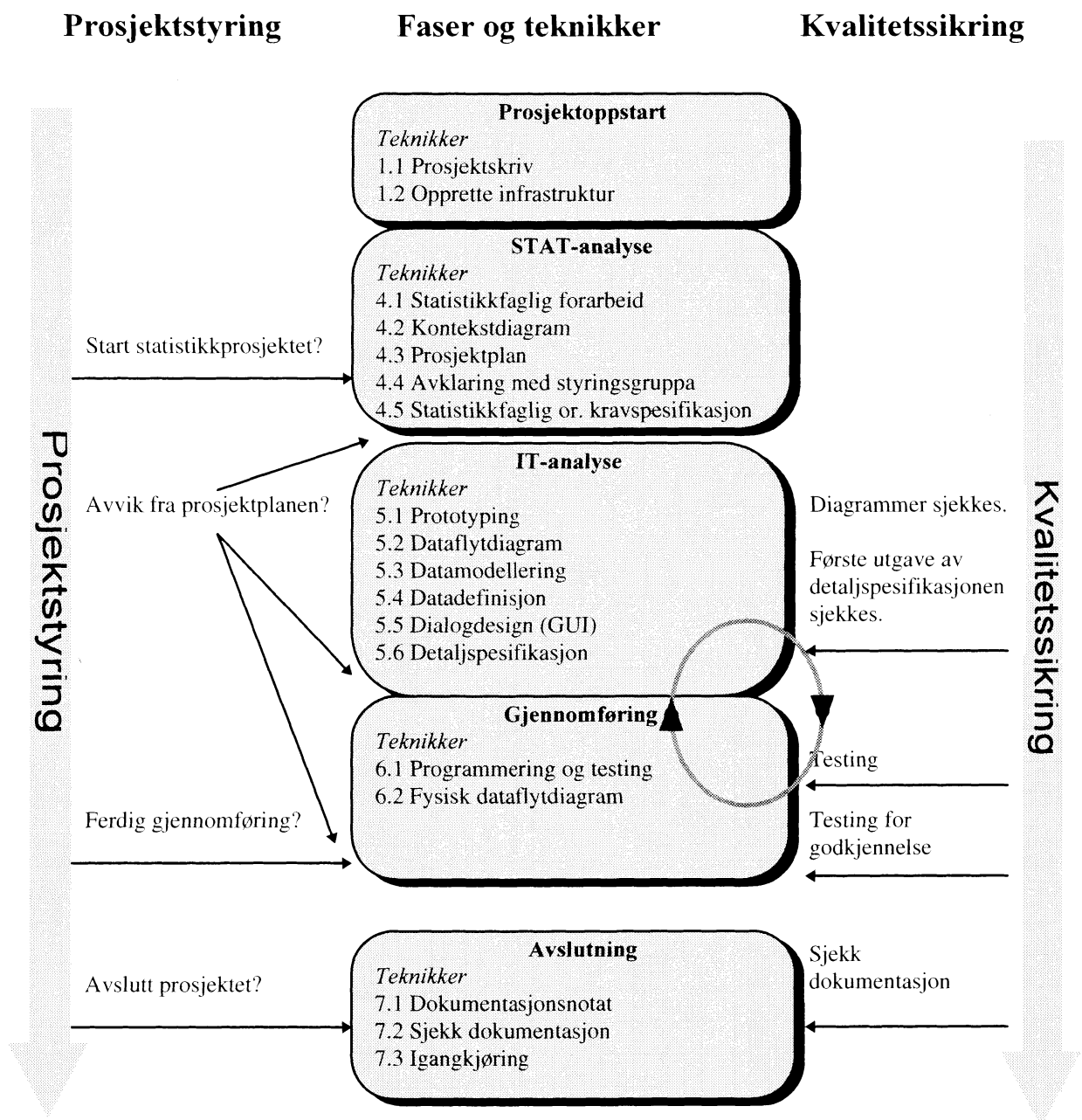
Før vi kan starte arbeidet med å lage statistikkssystemet må prosjektet startes opp. Prosjektene startes ("initeres") ofte etter følgende modell:

- ideer (fra ekstern oppdragsgiver, seksjonsledelse, DM, avdelingsmøte)
- høringsrunder på de relevante seksjonen(e) og evt. eksternt.
- utarbeide prosjektskriv (se håndbok i prosjektstyring).
- prosjektskrivet godkjennes (og implementeres i virksomhetsplan)

Prosjektskrivet er oppdragsgivers bestilling overfor (hoved)prosjektleder og prosjektgruppe. Når prosjektskrivet er godkjent, vil selve prosjektarbeidet starte. "Håndbok i prosjektstyring" foreslår en del prosedyrer i oppstartarbeidet. Vi anbefaler at disse prosedyrene følges og særlig gjelder dette for organiseringen av prosjektet. **Vi vil presisere at IT-kompetanse bør være representert i prosjektgruppen når prosjektet skal inkludere programmering.**

2.2 Struktur, prosjektstyring og kvalitetssikring

Arbeidet med å lage et statistikk-system starter formelt ved at et prosjektskriv som skal regulere et etterfølgende prosjekt godkjennes. Først når prosjektskrivet er godkjent starter prosjektet. Figuren nedenfor viser hvilke faser som gjennomgås og hvordan fasene følger etter hverandre i tid. På figuren er hver fase angitt med en boks med navn og teknikker i fasen. Ved venstre og høyre marg ser vi to piler som viser hvordan henholdsvis *prosjektstyring* og *kvalitetssikring* henger sammen med fasene i arbeidets gang. Fra prosjektstyringen i venstre marg skyter det ut piler som illustrerer hvordan ledelsen griper inn i prosjektet og styrer dette. Fra høyre marg illustrerer pilene tilsvarende hvilke tiltak som treffes for å sikre kvaliteten på delresultatene i prosjektet.



Figur 1.1: Oversikt over arbeidet i et statistikkprosjekt

Arbeidsprosessen er delt opp i *faser* hvor hver fase består av en samling *teknikker*. Teknikkene brukes for å løse de utfordringene som oppstår i arbeidet med å utforme en IT-løsning på det etterspurte system. Som et resultat av en teknikk får vi enten et tekstdokument eller et *diagram*

Metoden kan brukes både med og uten prosjektorganisering. Best resultater gir den med prosjektorganisering³.

Men la oss nå skaffe oss en rask oversikt over hvordan arbeidet i et statistikkprosjekt forløper. Dette gjør vi ved å gå gjennom fasene i tur og orden:

Prosjektinitierings-fasen sørger for at prosjektet kommer igang. Prosjektet utarbeides og det videre arbeidet tilrettelegges. Arbeidet i denne fasen utføres i stor grad av (hoved-)prosjektlederen og den som er IT-ansvarlig i prosjektet.

Vi går nå over i analysedelen av arbeidet. Dette er en sentral del av prosjektet hvor systemet planlegges detaljert, og vi sjekker ut om prinsippene fungerer i praksis. Analyse-fasen er delt i to like viktige deler:

STAT-analyse-fasen ender opp med en *statistikkfaglig kravspesifikasjon* av systemet. Kravspesifikasjonen viser hvordan de statistikkfaglige medarbeiderne ønsker seg at det nye systemet skal være. (Denne delen utføres i hovedsak av det statistikkfaglige personalet.) Her jobbes det etter statistikkfaglige prinsipper, hvilke data systemet skal ta utgangspunkt i, hva som skal være systemets produkter osv. Dessuten lages det en detaljert prosjektplan som avklares med styringsgruppa for prosjektet.

Etter at fagseksjonen har foretatt dette innledende arbeidet, er det klart for å starte opp den mer IT-pregede delen av prosjektet (**IT-analyse-fasen**). Nå samarbeider de statistikkfaglige medarbeiderne og IT-medarbeiderne nært om å skape den best mulige realiseringen av de statistikkfaglige medarbeidernes ønsker - det tilgjengelige ressursertatt i betraktning. I arbeidsprosessen benyttes diagrammer, skriftlige spesifikasjoner og programvare-biter, men til slutt ender vi opp med en tekstlig *detaljspesifikasjon* av hvordan systemet skal lages.


Når analysen anses som ferdig, går prosjektet over i **gjennomførings-fasen** som består av koding og testing utført av IT-medarbeiderne. Nå viser erfaringer fra SSB (og resten av verden) at vi ikke alltid klarer å få i stand en perfekt detaljspesifikasjon av IT-systemet som skal lages før man starter gjennomføringen. Ofte vil det bli oppdaget mangler og uklarheter i detaljspesifikasjonen fra IT-analyse-fasen, og det kan bli nødvendig å gå tilbake for å gjøre endringer i detaljspesifikasjonen - også etter at man har påbegynt gjennomføringen (*iterasjon*). Iterasjonen mellom de to fasene («IT-analyse» og «gjennomføring») symboliseres ved hjelp av en sirkel på figuren. Selv om vi må tilstrebe å få færrest mulig iterasjoner mellom analyse og gjennomføring, må vi holde fast på at det viktigste er at prosjektet ender opp med en best mulig løsning og god dokumentasjon⁴.

Avslutnings-fasen er i hovedsak viet til systematisering av dokumentasjonen i prosjektet - samt å få avsluttet prosjektet på en ryddig måte. Alt arbeid utover dette defineres som et nytt prosjekt - et vedlikeholdsprosjekt.

³ I et prosjekt er flere av følgende representert: *Prosjektleder*, *prosjektgruppe* (inklusive *IT-medarbeidere*, *brukere* og *teknisk prosjektleder*), *styringsgruppe* og *referansegruppe*.

⁴ På den annen side må det understrekes at forutsetningen for å kalle arbeidsformen «metodisk» er at man får klarlagt kravene til det systemet som skal lages før systemet programmeres.

3. PROSJEKTINITIERING

| |
|---|
|  |
| Prosjektinitiering |
| STAT-analyse |
| IT-analyse |
| Gjennomføring |
| Avslutning |

| | |
|----------------------|--|
| Idé | Ryddig start på prosjektet. |
| Mål | Oversikt over prosjektets mål og forventede ressursbruk. |
| Inn til fasen | - |
| Ut av fasen | 1. Prosjektskriv og prosjektgodkjenning 2. Opprett infrastruktur |
| Utførende | Fagseksjon, kundestøtte, IT-leder og (hoved-)prosjektleder (ev. IT-delprosjektleder) |

I denne fasen opprettes **infrastrukturen** rundt IT-prosjektet. Det opprettes blant annet dokumentasjonskataloger på Q-disken. I løpet av prosjektet fylles disse med resultatene fra de teknikkene som anvendes⁵.

3.1 Prosjektskriv og prosjektgodkjenning

| | |
|----------------------|---|
| Idé | Prosjektdrevet arbeidsform |
| Ambisjonsnivå | Brukes i utviklingsprosjekter. I mindre vedlikeholdsprosjekter vil ikke alltid et projektskriv foreligge. |
| Notat | - <i>Håndbok i prosjektstyring</i> |
| Mal | - |
| Ansvarlig | Fagseksjonen |
| Utførende | Fagseksjonen |

Beskrivelse:

1. For utfylling av projektskriv: se *Håndbok i prosjektstyring*.
2. Projektskrivet behandles og ev. godkjennes.

⁵ Dersom et prosjekt er delt opp i underprosjekter (med et tilhørende hierarki av underkataloger), skal hvert delprosjekt dokumenteres på sin respektive underkatalog.

3.2 Opprett infrastruktur

| | |
|----------------------|--|
| Idé | Tilrettelegge for IT-delen av prosjektet. |
| Ambisjonsnivå | Oppfordring til at to eller flere IT-medarbeidere jobber sammen om IT-arbeidet i prosjektet. |
| Notat | - <i>Håndbok i prosjektstyring</i> - <i>Kort innføring i ny navnestandard for UNIX-filer</i> - <i>Datalagring på UNIX i Statistisk sentralbyrå</i> |
| Mål | - |
| Ansvarlig | IT-ansvarlig, (hoved)prosjektleder |
| Utførende | IT-medarbeider (ev. IT-delprosjektleder) og kundestøtte |


Beskrivelse:

- IT-leder har følgende funksjoner ved oppstarten av IT-arbeidet i prosjektet:
 - Han mottar forespørselen fra fagseksjonen og velger ut prosjektdeltakere for IT-delen av arbeidet i prosjektet.
 - Han velger ut en IT-ansvarlig. Denne får funksjonen å lede IT-arbeidet i prosjektet. Tittelen er IT-ansvarlig, IT-koordinator eller IT-prosjektleder, alt etter størrelsen på prosjektet.
 - Han oppretter en prosjektgruppe som består av minst én *IT-medarbeider* og minst én *statistikkfaglig medarbeider*. Hvis dette er samme person, bør det velges en kontaktperson ved den tilhørende IT-gruppa som kan fungere som rådgiver for den statistikkfaglige medarbeideren når han designer statistikksystemet og programmerer dette.
- IT-ansvarlig har ansvaret for følgende:
 - Hvis det finnes et *prosjektskriv* eller et *IT-delprosjekt-skriv*, tar IT-arbeidet utgangspunkt i dette⁶.
 - Hvis prosjektet bruker UNIX, ber IT-ansvarlig kundestøtte om å opprette en stammekatalog på UNIX. Samtidig spesifiserer IT-ansvarlig medlemmer til lese- og skrivegruppene i dette prosjektet.
 - IT-ansvarlig sørger for å be kundestøtte opprette en katalogstruktur for dokumentasjonen på PC. Han gir kundestøtte en liste over prosjektdeltakere.
- Kundestøtte er ansvarlig for:
 - å opprette en stammekatalog på UNIX lese- og skrivegrupper slik der er spesifisert av IT-ansvarlig.
 - å opprette katalogstrukturen på PC: *Q:\DOK\ med underkataloger slik figuren i avsnitt 6.2 viser. Når et prosjekt er delt opp i delprosjekter, kan det være lurt å dele hovedprosjekt-katalogen opp i delprosjektkataloger. Vi får da en struktur som følger navnestandarden på UNIX. *q:\dok\⁷.**
 - at alle som jobber i prosjektet gis skrivetillatelse til den (de) aktuelle katalogen(e).
 - at prosjektnavn er det samme som stammekatalog på UNIX. (Dersom stammen på UNIX er delt inn i substammer, skal den samme stammeinndelingen brukes for prosjektets dokumentasjon på *Q:\DOK\.)*
 - at listen over de som har skrivetillatelse på *Q:\DOK\ er identisk med tilsvarende liste på UNIX.*

⁶ Prosjektskrivet inneholder noen basisopplysninger om prosjektet: Målet med prosjektet, hva prosjektet skal dekke, hvor lang tid det forventes å ta, litt om behov for ressurser og bestemme prosjektets rammebetingelser. Merk at metoden kan brukes selv om det ikke foreligger noe prosjektskriv. Planlegging av tidsforbruk på personer må da gjøres i en *tidsplan* som utarbeides etter at *dataflytdiagram (nivå 1)* er skissert. Timeverkene fordeles da i prosjektet på personer og faser. Ferdig-datoen på hver av fasene kalles for en *milepel*. Det er milepelene som prosjektet følges opp etter. (Eventuell detaljplanlegging ned på enkeltoppgaver innen hver fase er utelukkende til bruk internt i prosjektet.)

⁷ Dette tilsvarer på UNIX: *<stamme>\<type katalog>*

4. STAT-ANALYSE-FASEN

| | |
|--|--|
| Prosjektinitiering | <p>Idé Ryddig overgang fra statistikkseksjonens til IT-medarbeidernes arbeid.</p> <p>Mål Skape et sikkert og gjennomtenkt grunnlag for IT-medarbeidernes arbeid.</p> <p>Inn til fasen</p> <ol style="list-style-type: none"> 1. Prosjektskriv og prosjektgodkjenning 2. Opprett infrastruktur <p>Ut av fasen</p> <ol style="list-style-type: none"> 1. Statistikkfaglig forarbeid 2. Kontekstdiagram 3. Prosjektplan 4. Avklaring med styringsgruppa 5. Statistikkfaglig orientert kravspesifikasjon <p>Utførende Fagseksjon med bistand fra IT-medarbeidere</p> |
| STAT-analyse  | |
| IT-analyse | |
| Gjennomføring | |
| Avslutning | |

“*STAT-analyse*” er en forkortelse for “statistikkfaglig analyse”. Dette innebærer at denne fasen i hovedsak er ment å skulle utføres ved et statistikk-fagkontor i Statistisk sentralbyrå.

Prosjektgruppas medlemmer fra fagseksjonen (de statistikkfaglige medarbeiderne) må nå **analysere** dataene som skal inn og ut av systemet, samt prosessen som skal utføres i systemet. Denne teknikken har vi valgt å kalle for “*statistikkfaglig forarbeid*”. Sammen med IT-medarbeiderne tegnes kontekstdiagrammet opp

Basert på projektskrivet må prosjektgruppen arbeide frem en prosjektplan. IT-medarbeidere er også med på å anslå tidsbruk for å utarbeide systemet - basert på informasjonen som er innhentet i kontekstdiagrammet. Husk å sette av tid til dokumentasjon både underveis og i avslutningen av prosjektet. Arbeidet ender opp en prosjektplan:

- målformulering
- milepælplan
- detaljplan (her vil metoden være til hjelp, ved at for hver teknikk anslås tidsbruk, ansvarlig, utførende og avhengigheter)

Prosjektplanen behandles i styringsgruppa. Prosjektgruppa vil nå få avklart om prosjektet er på riktig spor i forhold til styringsgruppas ønsker. Når prosjektplanen er behandlet og evt. korrigeret, kan fagseksjonens medarbeidere begynne å spesifisere systemet. Denne planleggingen kalles her *statistikkfaglig orientert kravspesifikasjon*⁸ (av systemet).

⁸ En brukers kommentar til fasen

1. Det brukes generelt for lite tid til planlegging i prosjektene, og det diskuteres løsninger i stedet for å klarlegge hvilke problemer prosjektet skal løse.
2. Brukerne er ikke involvert i prosjektet og blir påtvunget løsningene til systemereren.
3. Noen ganger er bruker og systemerer samme person. Da blir det ofte slik at han konsentrerer seg mest om den faglige oppgaven/resultater, ikke spesifisering og dokumentasjon.

4.1 Statistikkfaglig forarbeid

| | |
|----------------------|--|
| Idé | Statistikkfaglig seksjon forbereder/klargjør seg til å spesifisere prosjektplan, herunder IT-arbeidet. |
| Ambisjonsnivå | Obligatorisk |
| Notat | <i>Datafangstprogrammet</i> |
| Mal | - |
| Ansvarlig | (Hoved-)prosjektleder |
| Utførende | Prosjektgruppa |

Beskrivelse:

1. Prosjektskrivet danner grunnlaget, sammen med evt. andre notater/dokumenter.
2. Det vil ofte være flere veier som fører til målet. Uansett må forutsetninger / datagrunnlag samt rammene kartlegges nøyer, før en starter å planlegge i detalj og involvere IT-medarbeiderne.
 - Tolking av prosjektskriv
 - resultatvariable
 - periodisitet, referanseperioder
 - populasjon, ønsket og trekkepopulasjon
 - nedbrytninger:
 - aktivitet
 - geografisk nivå
 - aktualitet:
 - produksjonstid
 - publiseringstid
 - kvalitet
 - annet
 - Kartlegge tilbud av data:
 - eksisterende statistikk
 - ny datainnsamling
 - registre
 - andre kilder
 - Analysere sammenheng og konsistens mellom dataene
 - Kontakt med mulige oppgavegivere (inkl. vurdering av oppgavebyrden):
 - kilder (person / bedrift / bransjeorganisasjon)
 - antall oppgavegivere
 - størrelse på oppgavegivere
 - dekningsgrad (krav/oppfylt)
 - enhetsproblematikk
 - variable
 - tidsfrister
 - periodisitet
 - relevans
 - samordning
 - måling av oppgavebyrde?
 - Teste tilgjengelige data:
 - fastlegge hvilke data som vil være nødvendige
 - få tilgang til dataene
 - hente dataene
 - analysere dataene (enheter, dekning, kvalitet, forsinkelser)
 - Økonomisk aspekt
 - Legale aspekter

3. Konklusjonene av forarbeidet gir prosjektgruppen grunnlag for å utarbeide prosjektplanen.
Sentrale valg er tatt og arbeidet med å spesifisere mål, milepæler og detaljplaner starter.
Teknikkene som må utføres for å utarbeide systemet planlegges som aktiviteter i prosjektet.
4. For prosjekter større enn 1000 timeverk skrives eget prosjektskriv for IT-delen av prosjektet.

Erfaringer:

1. Når datagrunnlaget er registre er ofte dataene ikke godt nok kjent av fagseksjonen. Dette kan ofte føre til at IT-medarbeiderne kommer til å jobbe med for dårlige spesifikasjoner.
2. Teknikken hjelper fagseksjonen i dens forberedelser til IT-arbeidet.

4.2 Kontekstdiagram

| | |
|----------------------|--|
| Idé | Kartlegge omgivelsene til det informasjonssystemet vi nå skal lage. |
| Ambisjonsnivå | Frivillig. Angi kun leverandører og mottakere av data. Kan tegnes manuelt. |
| Notat | - |
| Mal | - |
| Ansvarlig | Prosjektgruppa |
| Utførende | (Hoved-)prosjektleder |

Beskrivelse:





- (Hoved-)prosjektleder arrangerer et møte med de statistikkfaglige medarbeiderne hvor det blir tegnet opp et *kontekstdiagram*.
- Om opptegningen av kontekstdiagrammet:
 - Viser avsendere og mottakere av data og hvilke data som flyter ut og inn av systemet.
 - NB: Det som skjer *inne* i systemet ignoreres!
 - Diagrammet tegnes som grunnlag for *Statistikkfaglig orientert kravspesifikasjonen* fra det *STAT-analyse*.

Erfaringer:

- Diagrammet hjelper med å få startet opp arbeidet med spesifikasjonene. Teknikken gir en god oversikt over hvilke datakilder som inngår i systemet, og er mer nyttig enn det kan se ut til ved første øyekast.
- Etter at diagrammet er ferdig kan det *kopieres* til et nytt diagram som utvides til også å vise innholdet i systemet: dataflytdiagram (DFD1). Men merk at det er først i IT-analysefasen at dataflytdiagrammet tegnes opp **detaljert**.

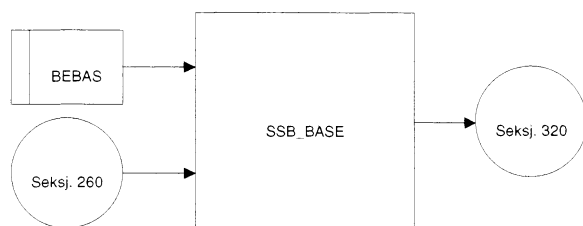
Teknisk anmerkning:

- “*Kontekstdiagram*” ble tidligere kalt “DFD0”.
- Vi bruker samme tegneteknikk i kontekstdiagram og logisk dataflyt (IT-analysefasen), men symbolbruken er noe forskjellig.
- Det er lurt å bruke en tavle når diagrammet tegnes opp. Da blir alle feil tydelige og det øker den felles forståelsen.
- Tillatte symboler** i teknikken “*Data Flow*”:

| Symbol |  |  |  |  |
|-----------|---|---|---|---|
| Betydning | Dataflyt retning | Datalleverandør/ datamottaker | Datalager | Systemet som skal lages |

NB! Vi kan velge om vi vil vise datakilder/datamottakere (○) eller input/output-datakilder (□).

Eksempel:



4.3 Prosjektplan

| | |
|----------------------|-------------------------------------|
| Idé | Planlegge ressursbruk i detalj. |
| Ambisjonsnivå | Se "Håndbok for prosjektstyring". |
| Notat | <i>Håndbok for prosjektstyring.</i> |
| Mal | - |
| Ansvarlig | (Hoved-)prosjektleder |
| Utførende | Prosjektgruppa |

Beskrivelse:

1. Se *Håndbok for prosjektstyring*.

4.4 Avklaring med styringsgruppa

| | |
|----------------------|---|
| Idé | Avklare om prosjektet er på riktig spor i forhold til styringsgruppens ønsker |
| Ambisjonsnivå | Obligatorisk |
| Notat | - |
| Mal | - |
| Ansvarlig | (Hoved-)prosjektleder |
| Utførende | (Hoved-)prosjektleder og styringsgruppe |

Beskrivelse:

1. Ta utgangspunkt i notatet fra "*Statistikkfaglig forarbeid*" og "*Prosjektplan*".
2. (Hoved-)prosjektleder arrangerer et møte med styringsgruppen, der konklusjoner av forarbeidet gjennomgås (notatet). Møtet kan tas i samband med organiseringen av prosjektet.
3. I forarbeidet har man analysert behovene, datakilder og kommet frem til løsningsforslag som man kan arbeide videre med. Møtet vil avklare hvilke alternativer og løsninger prosjektgruppa skal arbeide med.
4. Prosjektplanen godkjennes av styringsgruppen.

Erfaringer:

1. Det er ofte ikke gitt hvilke datakilder som skal brukes og hvilke brukerkrav som skal / kan innfris. Teknikken er nyttig for å avklare dette på et tidlig tidspunkt (slik at bortkastet arbeid med alternativer som forkastes unngås).
2. Styringsgruppa bør inneholde «lederne» til de som skal arbeide med prosjektet, slik at disse kan avsette ressurser (forpliktende) til riktig tid.
3. Avklaringer i styringsgruppa er viktig når prosjektplanen skal utarbeides. Hvis man vet hva man vil ha, vil planleggingen være mindre ressurskrevende.

4.5 Statistikkfaglig orientert kravspesifikasjon

| | |
|----------------------|---|
| Idé | 1. Dokumentasjon av statistikksystemet. 2. Grunnlag for å velge IT-løsning. NB! Ta ikke hensyn til IT-språk eller begrensninger i IT-systemer. |
| Ambisjonsnivå | Spesifikasjonen inneholder en gjennomtenkning av alle hovedpunkter som det ferdige systemet kommer til å inneholde. |
| Notat | <i>Revisjonshåndboka</i> |
| Mal | |
| Ansvarlig | (Hoved-)prosjektleder |
| Utførende | Prosjektgruppa, med evt. hjelp fra referansegruppa (hvis noen) |



Beskrivelse:

- Etter møtet med styringsgruppa, har prosjektgruppa nå beslutning fra styringsgruppen på hvilke løsninger som man skal arbeide med. Nå kan arbeidet med å spesifisere kravene til systemet starte. Kravspesifikasjonen vil danne grunnlaget for det arbeidet som IT-medarbeiderne skal utføre. De som skriver bør være bevisst på følgende:
 - Framgangsmåten bør være at man har et «tenk høyt»-møte (brainstorming) over systemet.
 - Beskriv det dere virkelig vil ha, ikke det som dere tror er teknisk mulig. Beskriv systemet som skal lages ut fra en statistikkfaglig synsvinkel. Dermed begrunner du også hvorfor dette er en fornuftig / nødvendig løsning. Hva som er mulig avgjøres i samarbeid med IT-medarbeiderne i IT-delen av prosjektet.
 - Spesifikasjonen skal omfatte deler som skal inngå i IT-løsningen / systemet. Ikke gå for mye i detaljer, da detaljene skal spesifiseres i IT-analysefasen (kapittel 4.4) i samarbeid med IT-medarbeideren.
 - Bruk egne ord - ikke IT-ord.
 - Hva kan effektiviseres, hva bør beholdes som det er, i arbeidsrutinen..
- Resultatene fra dette organiseres i skrevet på neste side. Denne spesifikasjonen danner grunnlaget for en felles forståelse for systemet og utgjør utgangspunktet for dataflytdiagrammet (og eventuell datamodell) som tegnes i IT-analysefasen.
- Mal for kravspesifikasjonen er gitt nedenfor.
- Revisjonshåndboka* inneholder mer detaljerte opplysninger/råd om de forholdene som skal utredes.



Erfaringer:

- Vær oppmerksom på at prosjekter kan være av ulike typer og at disse krever ulikt forarbeid:
 - *Konverteringsprosjekt* (overføring av et system til ny maskinplattform). Her er hovedsaken at du beskriver hvordan ting var før. Ofte kan konverteringsprosjekter regnes som utviklingsprosjekter fordi vi samtidig bør tenke nytt (om hva vi skulle ønske var mulig) for å forbedre eller utvide det eksisterende systemet.
 - *Utviklingsprosjekt* (utvikling av nye rutiner). Her kreves det mye forarbeid. Ofte kommer systemerene enten for seint eller for tidlig inn i dette arbeidet. Grunnen til dette er at de statistikkfaglige medarbeiderne spesifiserer for detaljert før de tar kontakt med IT-medarbeiderne. Slik kan de komme til å binde prosjektet til ikke-optimale løsninger. På den annen side hender det ofte at IT-medarbeiderne blir satt til å programmere ting som burde vært en del av «statistikkfaglig forarbeid» (se forrige teknikk).
 - *Vedlikeholdsprosjekt* (vedlikehold på et prosjekt som er avsluttet). Her vil ofte IT-medarbeideren kjenne systemet på forhånd, slik at forarbeidet ikke blir så omfattende.
 - *Betalte oppdrag*. Dette er ofte hasteprosjekter hvor spesifikasjonene kommer via telefon. Husk at også her gjelder regelen «hastverk er lastverk». Den statistikkfaglige medarbeideren bør derfor også her, som et minimum, levere en statistikkfaglig orientert kravspesifikasjon til systemereren.

Jho, ESn/- 12.05.1998 SIST ENDRET: 22.05.1998 Q:\DOK\SSB-BASE\analyse\KRAVSPEC.DOC v. 1.2

Avdeling for personstatistikk, Seksjon for persondata

Statistikkfaglig orientert kravspesifikasjon for SSB-BASE

<Innholdsfortegnelse>

1. Innledning

- Referanse til prosjektskriv hvis det finnes⁹

2. Oversikt og krav til systemet

- Mål: Hvilke hovedoppgaver systemet skal løse
- Basis (tidligere systemer, vedtak, standarder mm)
- Avgrensning
- Hva må forbedres i forhold til tidligere systemer
- Systemets forhold til omgivelsene
- Målgruppe
 - definere brukergruppen (antall, område osv.)
 - spesifisere nødvendig kompetanse

⁹ Hvis prosjektskriv ikke finnes, skal innledningen inneholde følgende punkter:

- Bakgrunn for prosjektet
- Interessenter (oppdragsgiver, ev. styringsgruppe m.m.)
- Ansvarsforhold (prosjektleder, IT-ansvarlig, eventuelt styringsgruppas og referansegruppas medlemmer)
- Presentasjon av statistikkfaglige problemer
- Referanser til annen dokumentasjon som kan være av interesse (hvor den finnes, hvem som står ansvarlig for den)
- Hvem som er berørt av endringene
- Systemets plassering i en større helhet

3. Beskrivelse av systemet

Her skal systemet spesifiseres. Alle forhold som berører IT-løsninger må tas med. En behøver ikke å gå inn på de enkelte formler / metoder, men få med i grove trekk hva som skal inn i systemet, hva som skal skje der, og hva som skal ut av systemet.

3.1.1 Datainnsamling

- Innsamlingsmetode:
 - spørreskjema (angi: utforming og registrering)
 - register/EDI (angi: overføring til Statistisk sentralbyrå og bearbeiding)
 - intervju (angi: utforming, registrering og overføring til Statistisk sentralbyrå)
- Utvalg:
 - populasjon:
 - mål
 - trekkefil
 - utvalgsplan (utvalg av enhet)
 - stratifiseringer
 - rulleringer
- Preprinting:
 - grunnlag
 - forrige periodes verdi?
 - minibrev
 - varianter:
 - bokmål/nynorsk
 - én-/flerbedriftsforetak
 - andre
- Statistisk enhet:
 - oppgaveenhet
 - observasjonsenhet
 - analyseenhet
 - rapporteringsenhet
- Variable (kjennemerker)/ kilder:
 - definisjoner
 - terminologi
 - avgrensinger
 - utvalg/registerbasert
 - tabeller (innhold)
 - datakilder/registre
 - kilder til oppdatering/konsistenskontroll (f.eks. BoF)

3.1.2 Datafangst (IT)

- Registreringsrutiner
 - type (registrering mot skjermbilder, optisk lesing/scanning)
 - spesifisere hvordan feil skal rettes opp under datafangsten
 - avmerking (innkommet): Logging/tilbakemelding
 - koding
 - andre kommentarer
- Koblinger mellom datakilder (angi: datalagre og koblingsnøkler)
- Delregister

- Tvangsmulkt-rutiner
- Lagring
 - type data: rådata / ureviderte data
 - rutiner
 - hvor lenge
 - i hvilken grad skal lagring foretas
- Administrativ informasjon (rapporter/lister)

3.3 Revisjon (angi: type og aggregeringsnivå)

- Feilkontroller:
 - dubletter
 - inkonsistenser
 - validitetskontroller
 - ekstremkontroll
 - makrokontroll
- Grafisk revidering (f.eks. *SAS-Insight*)
- Feiloppretting:
 - automatiske rutiner
 - manuelle rutiner (behov for skjermbilder?)
- Frafall:
 - totalt
 - partielt
- Koding av feil, opprettinger, frafall
- Lagring (reviderte data)
- Administrativ informasjon (rapporter)

3.3 Estimeringer og aggregering

- Tidligere rutiner (ved overføring/konverteringsprosjekter)
- Vekting
- Indekser
- Imputeringer:
 - partielt frafall
 - totalt frafall
 - ekstreme verdier
 - metoder (nødvendig grunnlag)
 - koding av imputering og metode
 - godkjenning?
- Fordelinger / periodiseringer
- Hjelpevariable
- Oppblåsing
- Aggregering
 - hvilke data skal aggregeres
 - hva skal det aggregeres til (f.eks. makronivå)
 - hvordan skal aggregeringen gjøres
 - hvordan skal dataene lagres
- Usikkerhetsmål
- Konsistensbehandling
- Sesongjustering
- Statistiske korreksjoner
- Lagring (produksjonsdata - mikronivå)

- Administrativ informasjon (rapporter)

3.4 Publisering


- Statistiske standarder:
 - NACE (aktiviteter)
 - sektorer (NR)
 - produkter
 - prosesser
 - andre
- Publisering:
 - konfidensielle data
 - tabulering
 - tilgjengelighet
 - sted (US, PM, SM, Årbok, NOS, Internett, andre)
 - tidspunkt
 - skreddersydde tabeller?
 - lagring (produksjonsdata - makronivå)
- Feilkilder
 - Utvalgsfeil
 - Målefeil
 - databearbeidingsfeil
 - andre svakheter i statistikken

4. Tanker om eventuelle framtidige utvidelser

Vedlegg

- Ev. definisjoner
- Ev. skjema
- Referanser til relevante notater m.m.

5. IT-ANALYSE-FASEN

| | | |
|--|----------------------|--|
| Prosjektinitiering | Idé | Transformere kravspesifikasjonen til en IT-løsning. |
| STAT-analyse | Mål | Lage spesifikasjonen til det systemet som skal lages. Statistikkfaglige medarbeidere er ansvarlig for den endelige kvaliteten på spesifikasjonen, men den lages i et nært samarbeid mellom statistikkfaglige medarbeidere og IT-medarbeidere. |
| IT-analyse  | Inn til fasen | <ol style="list-style-type: none"> 1. Kontekstdiagram (systemets omgivelser) [frivillig] 2. Statistikkfaglig orientert grovspekifikasjon. |
| Gjennomføring | Ut av fasen | <ol style="list-style-type: none"> 1. Prototyping (ikke obligatorisk) 2. Dataflytdiagram 3. Datamodell (når relasjonsdatabaseløsning) 4. Datadefinisjon 5. Dialogdesign 6. Detaljspesifikasjon |
| Avslutning | Utførende | IT-medarbeidere i nært samarbeid med fagseksjon |

I IT-analysefasen jobber den samme arbeidsgruppa som jobbet sammen i fasen *STAT-analyse*. Nå kommer imidlertid IT-medarbeiderne inn for fullt slik at arbeidsinnstasen fordeles jevnere mellom IT-medarbeiderne og de statistikkfaglige medarbeiderne.

I denne fasen tas det utgangspunkt i spesifikasjonen (og kontekstdiagrammet) fra forrige fase og dette og **detaljer**es og **operasjonaliser**es (overføres til IT-språk) slik at IT-medarbeiderne skal kunne programmere dette i neste fase. Det er et mål å få så få overraskelser som mulig når programmeringen utføres. Bruk av prototyping gjør ofte dette arbeidet mer spennende og effektivt.

Det kan jobbes på flere ulike måter. Teamarbeid er sterkt anbefalt i denne fasen.¹⁰

¹⁰ Forslag til arbeidsform i analysefasen

1. Mål og tema for møtet er fastsatt på forhånd (f.eks. på forrige møte eller via mail).
2. De statistikkfaglige medarbeiderne har tenkt gjennom problemstillingen før møtet og har et notatutkast med seg.
3. Notatet diskuteres i plenum. Bruk tavle og tegn diagram hvis mulig.
4. Etterarbeid fra møtet.
 - De statistikkfaglige medarbeiderne endrer i spesifikasjonene
 - IT-medarbeiderne tegner elektronisk lagret diagram
5. Resultatene fra pkt. 4 er utgangspunkt for neste møte.

5.1 Prototyping

| | |
|----------------------|--|
| Idé | Spesifikasjonen som skal danne grunnlaget for gjennomføringen (programmeringen) av prosjektet blir utarbeidet ved en kombinasjon av «prøving og feiling» og tradisjonelle spesifikasjoner. |
| Ambisjonsnivå | Liten grad av funksjonalitet bygges inn i prototypen. |
| Notat | 1) Grafiske brukergrensesnitt i <i>Håndbok for programmerere</i> 2) Eventuell kommende «styleguide» (se nedenfor) |
| Mal | - |
| Ansvarlig | IT-medarbeider sammen med statistikk-medarbeider (ev. IT-delprosjektleder) |
| Utførende | IT-medarbeidere og statistikk-medarbeidere sammen (IT er redaktør). |

Beskrivelse:

IT-analyse-fasen kan naturlig gjennomføres ved hjelp av utstrakt grad av **prototyping**. Komponentene i prototypingen er de teknikkene som forklares i de neste avsnittene.

1. Hver av teknikkene utføres som beskrevet i punktene under. Når prosjektdeltakerne beslutter at de ikke kommer lenger, går de over til neste teknikk. Når alle teknikkene er utført, foreligger det et detaljspesifikasjon av systemet. Dessuten foreligger et EDB-produkt som inneholder hovedtrekkene av brukergrensesnittet.
2. Dersom det viser seg at den detaljspesifikasjonen som er laget ikke er tilstrekkelig detaljert eller ufullstendig av andre grunner, går prosjektdeltakerne sammen tilbake og jobber videre med én eller flere av teknikkene, alt etter hva som viser seg hensiktsmessig.

Erfaringer:

1. Prototyper kan med fordel brukes når:
 - IT-systemet er interaktivt (baseres tungt på skjermbilder)
 - Behovene ikke kan bli forstått uten operativ erfaring eller at videre design ikke kan komme igang uten at et forberedende system er bygd og kjørt
 - Systemene er meget store og derfor kompliserte¹¹. Systemet kan da spesifiseres bit for bit. Hver bit programmeres og testes ut før neste bit påbegynnes (*inkrementell systemutvikling*.)
2. Farer ved prototyping er blant annet
 - de fremtidige brukerne av systemet kan tro at prototypen er det endelige systemet
 - IT-medarbeiderne lar være å designe systemet skikkelig
 - det blir for mange prototyper i kommunikasjonen mellom partene¹²
3. Fordeler med prototyping er at prototypen
 - virker som kvalitetssikring på hvor god den felles forståelsen for IT-systemet er
 - gir god kommunikasjon og er gøy å jobbe med
 - gir rask respons mellom partene i et IT-prosjekt
4. En hel gruppe brukere bør delta i prototypingen slik at flere interesser blir tilgodesett.
5. Prototyping bør ta utgangspunkt i en *styleguide*. Styleguiden er et spesielt nyttig hjelpemiddel for å få et ensartet brukergrensesnitt slik at statistikkfaglige medarbeidere skal kjenne seg igjen fra program til program og systemløsning til systemløsning. Styleguiden er et dokument som beskriver retningslinjer for design av:

¹¹ ... eller at den totale funksjonen ikke kan bli implementert innen fastsatt tid.

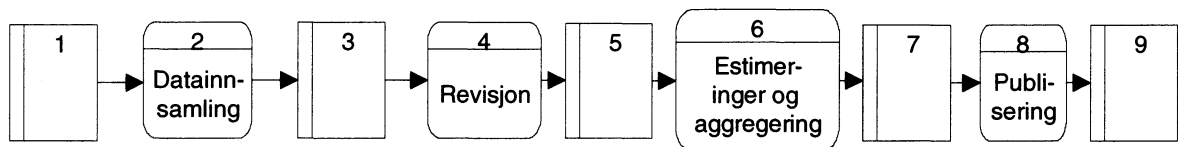
¹² Antall prototyper bør ofte være maksimalt 3 - 4. Bestem antallet før prototypingen påbegynnes.

5.2 Dataflytdiagram

| | |
|----------------------|--|
| Idé | Oversikt |
| Ambisjonsnivå | Obligatorisk. Oversikten bør kunne tegnes opp på én side i små prosjekter. I større prosjekter kan det brukes flere nivå av diagrammene. |
| Notat | Teknisk hjelp: Svein Jåvold (1997): <i>Minihåndbok i ABC Flowchart</i> , q:\dok\notater\handbok\abc.doc |
| Ansvarlig | IT-medarbeider (ev. IT-delprosjektleder) |
| Utførende | IT-medarbeider (red.) sammen med statistikkfaglige medarbeidere |

Beskrivelse:

- IT-ansvarlig samler **alle** prosjektdeltakerne for å tegne *dataflytdiagram* (nivå 1, dvs. DFD1). Dataflyt-diagrammet gir prosjektdeltakerne oversikt over *prosesser (moduler)*, *datalagre (logiske filer)*, *dataflyt* mellom dem, *datakilder* og *datamottakere*. *Prosesser* er aktiviteter som mottar, transformerer og/eller sender ut data. Husk å få med alle datalagre!
- Dataflytdiagram (også kalt *logisk dataflytdiagram*) bygger på den statistikkfaglig orienterte grovspefisikasjonen fra *STAT-analyse-fasen*. Derfor vil de fleste IT-systemene i Statistisk sentralbyrå passe inn i følgende skjema:



- La data flyte i én retning i diagrammet (fra venstre mot høyre eller ovenfra og nedover). Diagrammet får da en tidsakse slik eksemplet på neste side viser. (TIPS: Husk at en prosess alltid kommer mellom to datalagre (og omvendt).)
- Dersom det blir for mange bokser i diagrammet, kan det lønne seg å lage flere delprosesser, hver av dem med et eget diagram. Dette blir da nye dataflytdiagram på nivå 2 (DFD2). På denne måten får vi også en kontroll på at vi har tenkt langt nok, og vi kan være relativt sikre på at DFD1 beskriver en god systemløsning. Diagrammet bør derfor (i alle fall mer uformelt på papir) jobbes med ned på nivå 2 for å rette opp detaljer som kan påvirke diagrammet på nivå 1.
- Det bør foreligge dataflytdiagram nivå 1 også for relasjonsdatabaseløsninger. Disse bør beskrive flyten av data inn til databasen (lasterutiner) og uttrekksrutiner til filer og rapporter
- Sammenhengen mellom dataflytdiagram og datamodellen: Delprosessene i DFD2 kan være til god hjelp i datamodelleringen for å finne entiteter og relasjoner (sammenhenger).




Erfaringer:

- Dataflytdiagram blir ofte blandet sammen med det fysiske dataflytdiagrammet i gjennomføringsfasen. Forskjellen er at logisk dataflyt tegnes **før** systemet er programmert for å finne ut hvordan det **bør** være. Fysisk dataflyt tegnes *etter* at systemet er programmert og forteller hvordan systemet virkelig ble. Fysisk dataflyt er derfor mye mer detaljert enn logisk dataflyt.
- Dataflytdiagram* (DFD1) kunne også kalles for *oversiktsdiagram*. Det er mange grunner til at diagrammet har vist seg nyttig å bruke også i Statistisk sentralbyrå:
 - Å lage diagrammet gir en sjekk (kvalitetssikring) av spesifikasjoner fra statistikkfaglige medarbeidere at spesifikasjonene er riktige og at statistikkfaglige medarbeidere og IT-medarbeider har forstått hverandre.
 - Det er en visuell teknikk og kan forstås av de fleste uten mye trening.
 - Det gir den nødvendige grove oversikten for å kunne velge løsning.

- Det lager en modell som alle kan forstå av det systemet som skal lages.
 - Alle parter kan bruke diagrammet til å følge med hvor langt vi til enhver tid er kommet.
 - Det er en oversiktlig dokumentasjon av systemet til bruk for dem som ikke kjenner systemet fra før.
3. Det er viktig å være forberedt på mange endringer i diagrammet før det endelige utkastet i IT-analysefasen er klart. Avsett tilstrekkelig med tid til dette arbeidet! Den tiden får du godt betalt for senere..

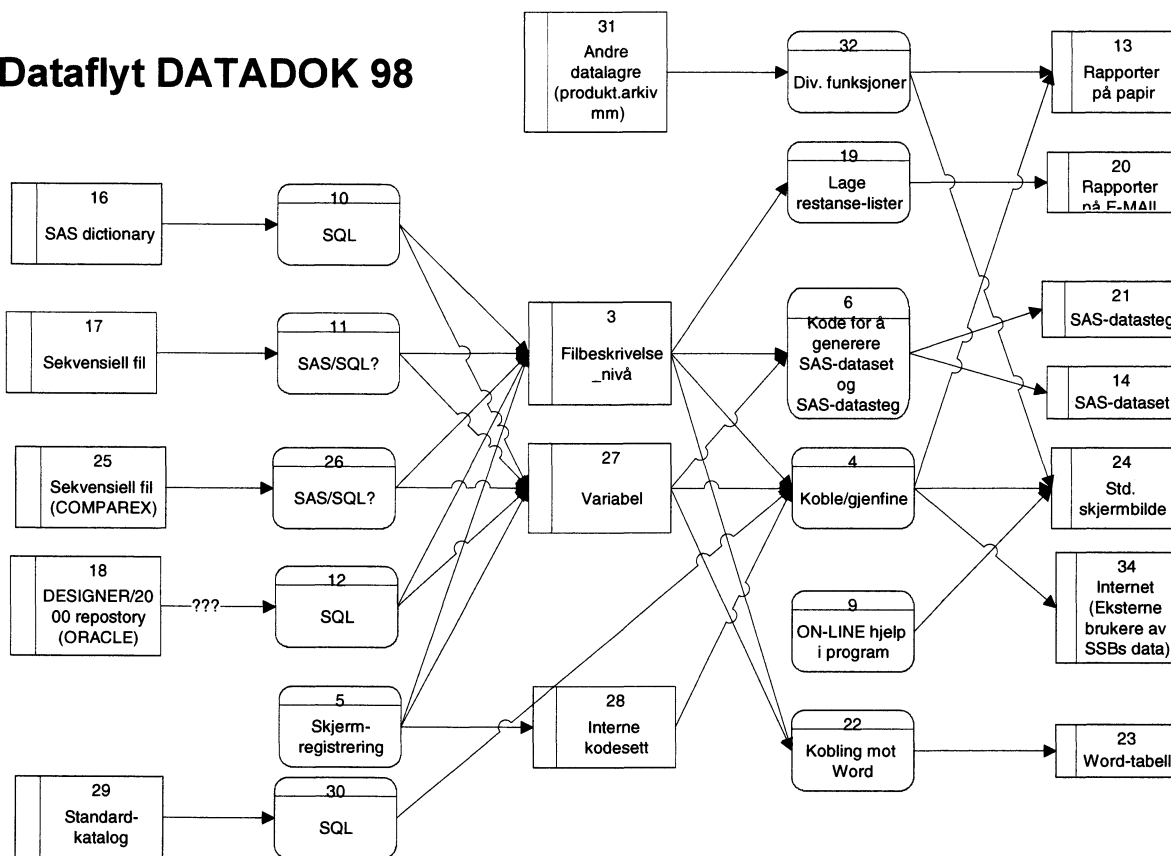
Teknisk anmerkning:

Tillatte symboler i teknikken "Dataflow":

| Symbol |  |  |  |
|-----------|---|---|---|
| Betydning | Dataflytretning | Prosess | Datalager |

Eksempel på dataflytdiagram:

Dataflyt DATADOK 98

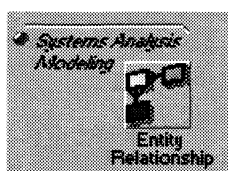


5.3 Datamodellering

| | |
|----------------------|---|
| Idé | Oversikt over, og sammenheng mellom, dataene i systemet. |
| Ambisjonsnivå | Obligatorisk . |
| Notat | Svein Jåvold (1997): <i>Minihåndbok i ABC Flowchart</i> , q:\dok\notater\handbok\abc.doc |
| Mal | - |
| Ansvarlig | IT-medarbeider (ev. IT-delprosjektleder) |
| Utførende | IT-medarbeidere (redaktør) sammen med statistikkfaglige medarbeidere |

Beskrivelse:

1. Når databaseløsningen skal diskuteres, innkaller IT-ansvarlig alle prosjektdeltakerne til et møte hvor det tegnes en *datamodell* på tavla.



2. (Den konseptuelle) datamodellen er en grafisk sammenheng mellom entiteter (objekter) som man tenker skal inngå i IT-systemet. Minimum følgende skal være med i diagrammet:

- navn på entiteten
- attributter (egenskaper) til entiteten
- relasjoner mellom entitetene

- entydig nøkkel (identifikasjon) til entiteten

I produktet *ORACLE DESIGNER/2000* gjøres dette ved teknikken «*Entity-Relationship*».

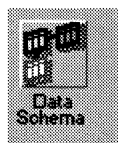


3. Når modellen er ferdig opptegnet, overfører vi den til den fysiske datamodellen, i denne prosessen transformeres og normaliseres modellen slik at

- entiteter til tabeller
- attributter til kolonner

- relasjoner til fremmednøkler
- primære unike identifikatorer til primærnøkler.

I produktet *ORACLE DESIGNER/2000* gjøres dette ved teknikken «*Database Wizard*».



4. I *ORACLE DESIGNER/2000* kan vi nå bruke teknikken «*Data Schema*» for å gjøre tilpasninger i den fysiske datamodellen.

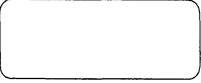

- I denne teknikken kan vi også generere DDL eller tabelldefinisjonen til databasen.

Erfaringer:

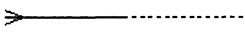

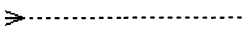
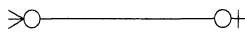


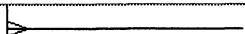
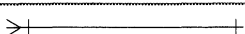



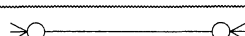
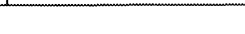
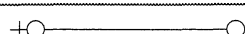
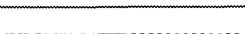
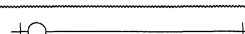
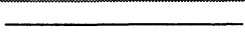
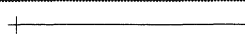
1. I SSB lagrer vi ofte data slik de kommer inn på skjemaene. Av og til ligger til og med hele skjemaet på samme record. Det kan diskuteres om dette krever datamodellering. Vi kan kanskje si at det allerede har foregått en viss (dessverre ofte) ustrukturert datamodellering ved design av skjemaet ved at de dataene (variablene) som hører sammen ofte er plassert nær hverandre på skjemaet.
2. Den konseptuelle (logiske) datamodellen kan gi økt forståelse for sammenhengen mellom data også for datalagring i ikke-relasjonsdatabaser (for eksempel SAS). Det anbefales derfor **aldri** å kjøre et prosjekt som inneholder bruk av data uten å bruke datamodellering.
3. Datamodellering har vist seg vanskelig å beherske på et tilfredsstillende høyt nivå. Datamodelleringen bør derfor foregå som en gruppeprosess under veiledning av en ekspert (intern eller ekstern).

Teknisk anmerkning:

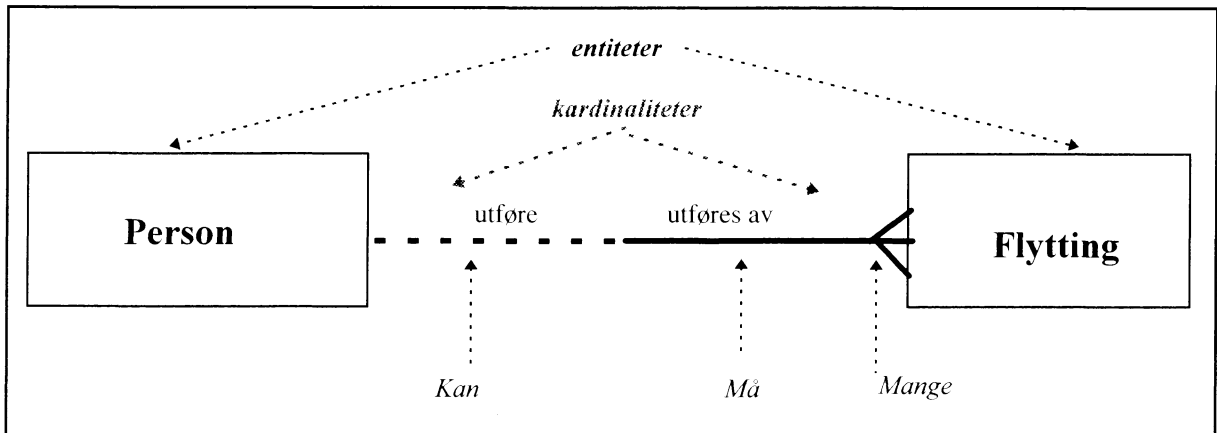
Tillatte symboler i teknikken «Entity Relationship»

| Symbolforklaring | Symbol |
|------------------|---|
| Entitet |  |
| Relasjonslinje |  |

• Relasjoner slik de gjengis i hhv. DESIGNER/2000 og ABC Flowchart:

| V->H: relasjon | H->V: relasjon | DESIGNER/2000 | ABC Flowcharter |
|------------------|------------------|---|---|
| EN | NULL eller MANGE |  |  |
| NULL eller EN | NULL eller MANGE |  |  |
| NULL eller EN | EN eller MANGE |  |  |
| EN | EN eller MANGE |  |  |
| EN eller MANGE | NULL eller MANGE |  |  |
| NULL eller MANGE | NULL eller MANGE |  |  |
| NULL eller EN | NULL eller EN |  |  |
| EN | NULL eller EN |  |  |
| EN | EN |  |  |

• **Anvendelse av relasjoner i datamodellering:**

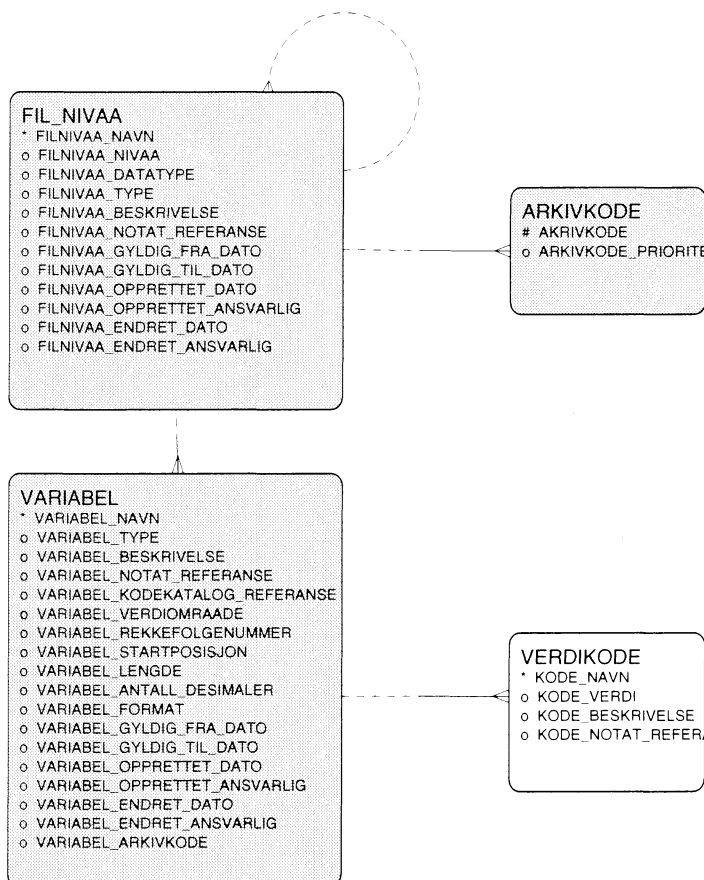


Figur: Elementer i datamodellen.

- Figuren over viser et enkelt eksempel på hvordan datamodellering ved hjelp av ER-metoden kan se ut. Legg merke til betydningen av minimums- og maksimums-kardinalitet.
 - Fra venstre mot høyre leser vi at *en person kan utføre mange flyttinger*.
 - Fra høyre mot venstre leser vi at *en flytting må utføres av én person*.

Eksempel: ORACLE Designer/2000:

Figuren leses slik:



- En fil kan ha flere arkivkoder tilknyttet seg. Men en bestemt arkivkode må være tilknyttet ett bestemt fil.
- En fil kan ha flere variabler tilknyttet seg. Men en variabel må alltid tilhøre en bestemt fil.
- En variabel kan ha flere lovlige (verdi)koder. Men en verdikode må alltid beskrive en bestemt variabel.
- Så et eksempel på litt mer avansert datamodellering: Den øverste relasjonen er en *egenrelasjon*, og den er der fordi det som vi til nå har kalt «fil» egentlig er en sekvens av navneledd (sml. SSBs navnestandard; f.eks: \$SKATT/person/data/g91/aksje). Relasjonen sier at et filnivå kan bestå av mange under-filnivåer, og at et filnivå kan høre til ett overordnet filnivå.

5.4 Datadefinisjon

| | |
|----------------------|--|
| Idé | Data defineres slik at alle i SSB har ON-LINE tilgang til dem. |
| Ambisjonsnivå | Obligatorisk. Alle filer som ikke er midlertidige dokumenteres. |
| Notat | - <i>Datalagring på Unix i Statistisk sentralbyrå</i> - <i>Kort innføring i ny navnstandard for Unix-filer</i> - <i>Brukerveiledning for datadokumentasjonssystemet!</i> |
| Mal | - |
| Ansvarlig | Statistsikk-medarbeiderne (de som eier dataene som beskrives) |
| Utførende | Dataeier. |

Beskrivelse:

1. Data dokumenteres elektronisk i «DATADOK».
2. Følgende regler gjelder for lagring av data:
 - Alle langtidslagrede data og produksjonsdata dokumenteres. Andre data kan dokumenteres hvis ønskelig.
 - Alle data langtidslagres på sekvensielle filer (med fast recordlengde).

Erfaringer:

1. Dokumentasjonen av data i SSB må økes for å unngå feilinformasjon eller bortkastet tid på grunn av udokumenterte data.
2. Midlertidige filer kan med ofte fordel dokumenteres sammen med produksjonsdata, men dette avhenger av bruken.

Teknisk anmerkning:

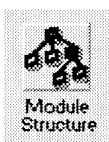
1. I DATADOK registreres **ikke** de fysiske filene som inneholder dataene. Det er **beskrivelsene** av filene som lages i DATADOK.

5.5 Dialogdesign (GUI)

| | |
|----------------------|--|
| Idé | Vise hvordan skjermbildene henger sammen (menysystem). |
| Ambisjonsnivå | Interaktive systemer. Frivillig på mindre prosjekter. |
| Notat | - |
| Mal | - |
| Ansvarlig | IT-medarbeider |
| Utførende | IT-medarbeider |

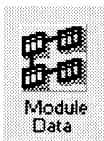
 **Beskrivelse:**

1. IT-medarbeiderne tegner opp et dialogsystem sammen med statistikk-medarbeiderne. Dialogsystemet viser hvordan dialogen mellom programsystemet og brukeren fungerer. I DESIGNER/2000 bruker du følgende teknikker:



2. For å lage menyen bruker vi diagrammet *Module Structure*. Vi gjør da slik:
- Tegn opp hoved-meny-punktet
 - Tegn opp under-meny-punkt
 - Plasser under-menypunktet hierarkisk i forhold til punktet på nivå over.

3. Vi skal nå lage skjermbildene. IT-medarbeiderne og de statistikkfaglige medarbeiderne lager skjermbildene sammen. De tar hensyn til:




- skjermbildet (vinduer, menylinje, knapperad, statuslinje, kommandoknapper, fargebruk)
- feedback (visualisering av ventetid, feilmelding, andre meldinger)
- navnstandarder (variable, programmer)
- standardrutiner som kan kalles opp (Se *Håndbok i programmerere*)



I ORACLE DESIGNER/2000 bruker vi teknikken "*Data Module Diagram*". Det gjør vi slik:

- Velg de tabellene som du vil ha med på skjermbildet (inkludert gjentakelser tabeller i form av master/detail).
- Fyll tabellene med de kolonnene som du vil ha med på skjermbildet.
- Generer relasjonene mellom tabellene.
- Pass på at inndelingen i skjermbilder (Canvas), vinduer (Windows) og blokker av data er riktig.

Også i SAS og andre verktøy kan det lages et interaktivt (ON-LINE) system med mange skjermbilder hvor brukeren kan bevege seg mellom skjermbildene. Det som da tegnes opp er skjermbildene. Bildene bindes sammen med kommandoer som gjør at den som prøver ut systemet ("prototypen") kan bla fram og tilbake i bildene og bli kjent med menysystemet.

 **Teknisk anmerkning:**

Tillatte symboler i teknikken «Dialogdesign»:

| | | |
|------------------|---|---|
| Symbol |  |  |
| Betydning | «Window», «canvas» (skjermbilde), «block» | Forbindelse, valg |

I DESIGNER/2000 benytter teknikken *Data Module Diagram* komponentene «window», «canvas» og «block» samt forbindelseslinjene mellom blokkene. Alle forbindelseslinjer er fremmednøkler.

5.6 Detaljspesifikasjon

| | |
|----------------------|--|
| Idé | 1. Transformere kravspesifikasjon til en IT-faglig spesifikasjon. 2. Hver datalager/fil og hver prosess/program får et eget kapittel i detaljspesifikasjonen. |
| Ambisjonsnivå | Obligatorisk. Detaljspesifikasjonen lages som kapitler på ett eller flere tekstbehandlingsdokumenter. |
| Notat | - |
| Mal | - |
| Ansvarlig | (Hoved-)prosjektleder |
| Utførende | Statistikk-medarbeidere (redaktør) sammen med IT-medarbeidere. |

Beskrivelse:

1. Statistikkfaglige medarbeidere skriver *detaljspesifikasjonen* i Word som ett eller flere dokumenter.
2. Beskrivelsen tar utgangspunkt i *Statistikkfaglig orientert grovspesifikasjon* som statistikkfaglige medarbeidere har skrevet. Detaljspesifikasjonen må ha samme struktur som dataflytdiagrammet (DFD1) (se malen under) Detaljspesifikasjonene baserer seg på det logiske dataflytdiagrammet og eventuelt en datamodell ved at dette/disse fungerer som en «grafisk innholdsfortegnelse» for detaljspesifikasjonen. (Dvs: For hver prosess i (logisk) dataflytdiagrammet lages det et avsnitt eller kapittel i detaljspesifikasjonen. Summen av disse kapitlene utgjør detaljspesifikasjonen.
3. Statistikkfaglige medarbeidere følger malen i eksemplet.
4. For hver prosess på logisk dataflytdiagram lages et kapittel med detaljert beskrivelse. Ta med stoffet fra «*Statistikkfaglig orientert kravspesifikasjon*», men forsøk å få et enda høyere presisjonsnivå (om mulig). Husk blant annet følgende:
 - Operér med DATADOKs navn på variabler
 - Spesifiser alle spesialtilfeller
 - Ikke ta ting for gitt; husk at IT-medarbeiderne ikke kjenner området like godt som statistikkfaglige medarbeidere! Sjekk derfor at begge parter forstår innholdet på samme måte.
 - Nøyaktige formler (ta gjerne med et eksempel)
 - Dersom *Detaljspesifikasjonen* ikke utarbeides ved å starte med teksten fra *Statistikkfaglig orientert kravspesifikasjon*, bør det lages rikelig med henvisninger til kravspesifikasjonen i Detaljspesifikasjonen.

Erfaringer:

1. I små prosjekter holdet det ofte å lage et enkelt dataflytdiagram (DFD1). For større prosjekter er det naturlig å dele dataflytdiagrammet DFD1 opp i flere underdiagrammer (DFD2).
2. Det er stor forskjell på hvor mye IT-medarbeiderne på forhånd kjenner til av systemet som skal lages. Men selv når IT-medarbeideren kjenner systemet godt på forhånd, bør det dokumenteres grundig under arbeidets gang. Dårlig dokumentasjon vil lett kunne føre til manglende nytenkning samt dårlig dokumentasjon for ettertiden.

Eksempel:

SVE/- 24.05.96 SIST ENDRET: 13.08.97 Q:\DOK\SSB-BASE\ANALYSE\INIT.DOC v 3.0

Avdeling for personstatistikk, Seksjon for persondata

Detaljspesifikasjon for Befolkningsbasen

<Innholdsfortegnelse>

1. Innledning

- Henvising til annen relevant dokumentasjon.

2. Datainnsamling

2.1 <Prosess nr d1>

.....Beskrivelse.....

2.2 <Prosess nr d2>

.....Beskrivelse.....

3 Revisjon

3.1 <Prosess nr r1>

.....Beskrivelse.....

3.2 <Prosess.nr r2>

.....Beskrivelse.....

4. Estimeringer og aggregering


..... OSV

5. Publisering

..... OSV

Vedlegg

6. GJENNOMFØRINGS-FASEN

| | |
|--|--|
| Prosjektstart | <p>Idé Programmering og testing.¹³</p> <p>Mål: Ende opp med en ferdig programkode som er i samsvar med spesifikasjonene fra statistikkfaglige medarbeidere og retningslinjene i SSB.</p> <p>Inn til fasen 1. Dataflytdiagram 2. Datamodell 3. Datadefinisjon 4. Eventuell dialogdesign 5. Detaljspesifikasjon</p> <p>Ut fra fasen 1. Kjørbare programmer 2. Dokumentasjon innen det enkelte program 3. Fysisk dataflyt med sammenhengen mellom programmene</p> <p>Utførende IT-medarbeidere med bistand fra fagseksjon</p> |
| STAT-analyse | |
| IT-analyse | |
| <p>Gjennomføring</p>  | |
| Avslutning | |

I denne fasen jobber hovedsaklig IT-medarbeiderne. De **programmerer** ut fra spesifikasjonene og prototypen utarbeidet i forrige fase.

Merk at detaljerte retningslinjer for hvordan programmeringen skal utføres er beskrevet i *Håndbok i programmerere*.

¹³ Denne fasen er i systemutviklingsmetoder fra '70 og '80-tallet oppdelt i én eller flere av fasene «logisk design», «fysisk design» og «konstruksjon». I vår metode er dette forenklet for å unngå for mye detaljregulering. Dette gir også økt fleksibilitet mht. verktøy (operativsystem, programmeringsspråk, databaser osv.).

6.1 Programmering og testing

| | |
|----------------------|--|
| Idé | Programmér med tanke på at andre skal lese programmet! |
| Ambisjonsnivå | Obligatorisk. Følge SSBs standarder for programmering. |
| Notat | 1. <i>Håndbok for programmerere</i> 2. Div. håndbøker om SAS, UNIX osv. |
| Mal | \$FELLES/maler/heading.txt (UNIX) |
| Ansvarlig | IT-medarbeider |
| Utførende | IT-medarbeider + bruker og IT-medarbeider på testing |

Beskrivelse:

- IT-medarbeideren programmerer etter retningslinjer i «*Håndbok for programmerere*».
- Det er 3 typer tester. Hver av testene har som funksjon å finne feil i systemet:
 - *Modultesting*: Testing av enkeltprogrammer
 - *Systemtest*: Test av sammenhengen mellom programmene
 - *Akseptansetesting*: Test på om systemet som helhet kan godtas av brukerne. For konverteringsprosjekter innebærer dette som et minimum at resultater produsert av det nye systemet sammenlignes med resultatene fra det gamle. For nyutviklede systemer sjekkes det at resultatene er i samsvar med forventningene til statistikkfaglige medarbeidere.
- Når en IT-medarbeider driver programutvikling **både** på PC/Windows og på UNIX skal programmene **ikke** kopieres mellom plattformene. For å sikre tilgjengelighet til programmer på PC brukes «q:\dok\<prosjektnavn>» også til å inneholde referanse til kataloger hvor programmer ligger lagret på UNIX (under hvert enkelt prosjekt). I dokumentasjonsnotatet skal det også komme klart fram hvor programmene ligger.
- Programmene må dokumenteres tilstrekkelig. Spesielt viktig er dette for støtte batch-jobber. Eksempel på heading i et slikt program er:

```

/*****
Prosjekt .....: SSB-BASE
Programnavn .....: arb_Bosatt
Skrevet av .....: Bjørn Joneid/24.06.1996
Programmets funksjon .: Kobler filene og tester lovlige koder.
Anmerkning.....: Programmet forutsetter at data er korrekte.
Filer inn.....: $SSB-BASE\data\arbtak.dat
                  $SSB-BASE\data\bosatt.dat
Filer ut.....: SSB-BASE m. arb.takere
Endret .....: 26.11.1996/Erik Sørberg
                  Tester nå på lovlige koder.
Endret.....: 03.05.1998/sve
                  Endret format på variabel "varer" til numerisk.
*****/

```

6.2 Fysisk dataflytdiagram

| | |
|----------------------|---|
| Idé | Oversikt over informasjonssystemet vi <u>har</u> laget. |
| Ambisjonsnivå | Obligatorisk i batch-systemer. Viser sammenhenger mellom programmer og filer. |
| Notat | q:\dok\notater\handbok\abc.doc |
| Mal | - |
| Ansvarlig | IT-medarbeider |
| Utførende | IT-medarbeider |

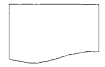






Beskrivelse:

Etter at programmene er kodet og testet må den endelige versjonen av programmene dokumenteres. I batch-systemer tegnes da *fysiske dataflytdiagram(mer)* som viser sammenhengen mellom programmene. Fysisk dataflytdiagram inneholder følgende bestanddeler:

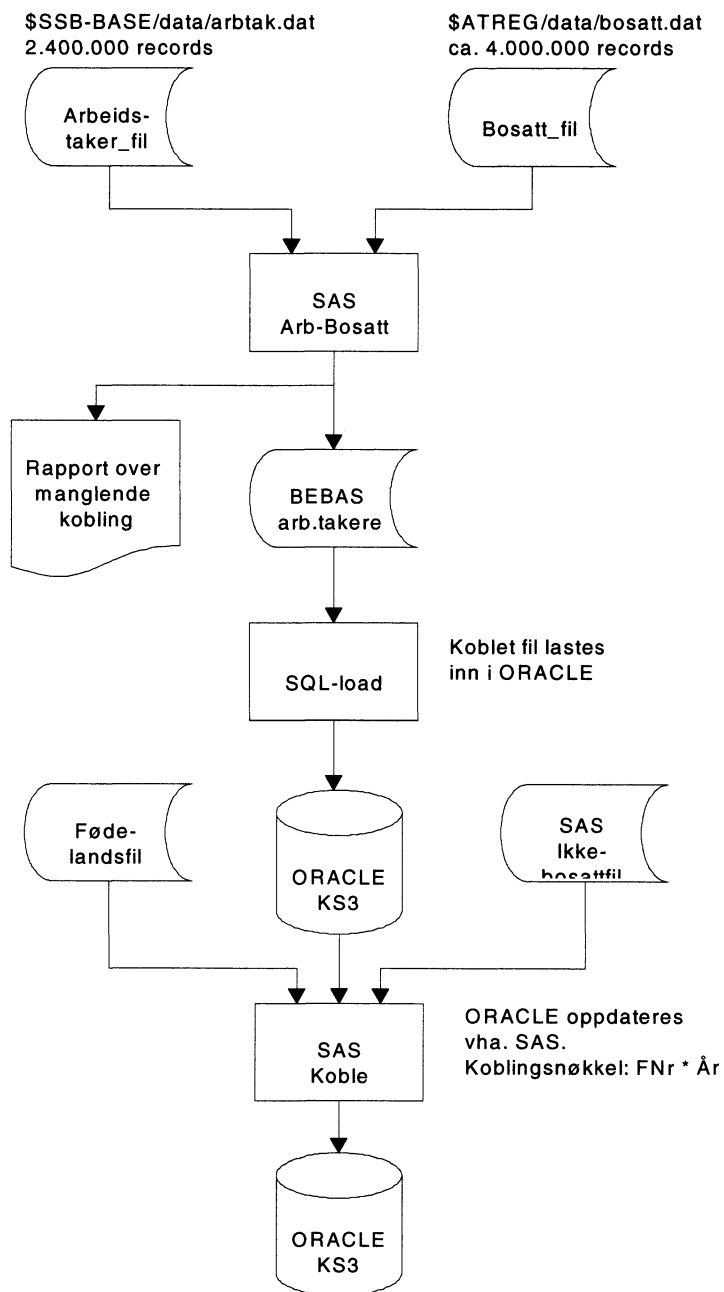
1. Grafiske symboler for filer og programmer og flyten mellom dem.
2. På filer/databasetabeller skal følgende spesifiseres:
 - Navn (inkl. SAS-libname/ORACLE-brukernavn og stammenavn på UNIX)
 - Eventuelt omtrentlig antall records/rader
3. På programmer skal følgende spesifiseres:
 - Programnavn (ikke katalognavn)
 - Beskrivelse av hva programmet gjør
 - Annet er frivillig, men unngå rot!
4. På diagrammet (som helhet) skal det tilføres en dokumentasjonsheading som inneholder:
 - Diagramnavn (automatisk)
 - Versjonsnummer
 - Ansvarlig for programmene i diagrammet
 - Ansvarlig for spesifikasjonene til programmene
 - Hvilke kataloger programmene er lagret på (stammenavn på UNIX). Hvis det er mange «programmer» (funksjoner, subrutiner, prosedyrer) lagret på samme programfil, oppgis filnavnet

Erfaringer:


1. *Fysisk dataflytdiagram* er en teknikk som tradisjonelt har stått sterkt i SSB og som mange kjenner igjen som såkalte «flowchart», «flytkart», «flytskjema», «programflyt» eller «systemkart».
2. Diagrammet ligner på *logisk dataflytdiagram (DFD)*, men brukes for dokumentasjonens skyld og ikke som et verktøy for å *design* de enkelte *prosesser* og *transaksjoner* som inngår i systemet.
3. Hvis det er spesielt viktige skjermbilder i tilknytning til prosesser i IT-løsningen, kan disse tas med i *fysisk dataflytdiagram* i tillegg til prosessene.
4. Det anbefales ikke å bruke *fysisk dataflytdiagram* til å kommentere strukturen i et program. Det enkelte programmet kommenteres utelukkende internt i programmet (se «koding»). Programmer som er kompliserte må ha desto flere kommentarlinjer!
5. **Tillatte symboler** i teknikken «FlowChart»:

| Symbol |  |  |  |  |  |  |  |
|-----------|---|---|---|---|--|---|---|
| Betydning | Dokument (papirtabell) | Skjermbilde | Database/dataset oracle, sas, fame | Sekvensiell fil | Program | Dataflyt-retning | Peker til forts. |

Eksempel:



7. AVSLUTNINGS-FASEN

| | | |
|--|----------------------|---|
| Prosjektinitiering | Idé | I denne fasen ryddes det opp i den dokumentasjonen som allerede eksisterer - slik vi ønsker prosjektet skal framstilles for ettertiden. Den gjenstående dokumentasjonen lages i denne fasen. |
| STAT-analyse | Mål | Lagre informasjon om systemet slik at det er lett å finne fram til og så lett å forstå at den blir brukt i praksis. |
| IT-analyse | Inn til fasen | <ol style="list-style-type: none"> 1. Kontekstdiagram 2. Dataflytdiagram 3. Datamodell 4. Detaljspesifikasjon 5. Eventuelt dialogsystem 6. Dokumentasjon innen det enkelte program 7. Fysisk dataflytdiagram med sammenhengen mellom programmene |
| Gjennomføring | Ut fra fasen | <ol style="list-style-type: none"> 1. Dokumentasjonsnotat 2. Informasjon om prosjektet som helhet |
| Avslutning  | Utførende | Fagseksjon, IT-medarbeidere og IT-ledelse |

I denne fasen jobber igjen IT-emdarbeiderne og de statistikkfaglige medarbeiderne sammen. Hovedpoenget i denne fasen er å **“sy sammen”** og avslutte alt som er laget i løpet av prosjektet. Når denne fasen er slutt skal brukerne være selvhjulpne i forhold til den løsningen som er laget.

7.1 Dokumentasjonsnotat

| | |
|----------------------|--|
| Idé | Samle all dokumentasjon av systemet på ett sted. Skape brukerens bilde av systemet og oversikt over systemet. |
| Ambisjonsnivå | Obligatorisk 1) Lagre den som Word-dokument på <i>q:\dok\...</i> 2) Systemdokumentasjon og brukerdokumentasjon |
| Notat | - |
| Mal | - |
| Ansvarlig | (Hoved-)prosjektleder |
| Utførende | IT-medarbeider og statistikk-medarbeider (redaktører i fellesskap) |

Beskrivelse:

1. Statistikk-medarbeider og IT-medarbeider skriver sammen en endelig dokumentasjon av systemer etter malen gitt nedenfor. Innholdet kan gjerne hentes fra tidligere utarbeidede dokumenter og diagrammer.

Erfaringer:

1. Innhold og struktur vil kunne variere fra prosjekt til prosjekt. I hvert fall i mindre prosjekter er det unødvendig å skille mellom brukerdokumentasjon og systemdokumentasjon (de slås sammen til et «dokumentasjonsnotat»).

Eksempel:

SVE/- 22.08.96 SIST ENDRET: 22.08.96 Q:\DOK\SSB-BASE\DOKNOT.DOC v 1.0

Dokumentasjonsnotat for SSB-BASE

- Forfatter(ne)s navn
- Navn på prosjektet og seksjonen(e)

<Innholdsfortegnelse>

1. Innledning (ev. delvis kopi av prosjektskriv)

- Bakgrunn for prosjektet
- Målgruppe og interessenter (oppdragsgiver, ev. styringsgruppe mm)
- Referanser til annen dokumentasjon som kan være av interesse¹⁴.
- Leseveiledning (hvem som skal lese hva osv.)

2. Oversikt over systemet (eventuelt delvis kopi fra prosjektskrivet)

- Kort beskrivelse av systemet (ev. kopi fra detaljspesifikasjon)
 - hvilke hovedoppgaver systemet skal løse
 - hvem som er berørt av endringene i dette nye systemet i forhold til tidligere
 - oppsummering av muligheter og begrensninger i systemet
- Omgivelsene systemet skal fungere i
 - standarder
 - databasesystem
 - beskrivelse av standardprogramvare (ev. også programutviklingshjelpemidler og programbibliotek)
 - kommunikasjonsutstyr (ev. også datamaskiner som har betydning for systemet)

*When in trouble,
when in doubt;
run in circles,
scream and shout!*

**3. Bruk** (brukerdokumentasjon)

¹⁴ Ansvarsforhold kan i ON-LINE-systemer legges inn i hjelp-funksjonen i programmet:

- kontaktperson ved drift, forslag, endringer og systemfeil
- prosjektleder
- IT-ansvarlig
- eventuelt styringsgruppas og referansegruppas medlemmer

- Forkunnskaper som forventes av statistikk-medarbeideren (inkl. krav til opplæring i bruk av systemet)
- Systemets restriksjoner (hva det kan og hva det ikke kan)
- Liste over datafiler/rapporter som systemet bruker/produserer
- Veiledning for normal bruk av systemet
 - Hvordan sette systemet igang
 - Beskrivelse av brukergrensesnittet som helhet
 - konsept (modell presentert for brukerne)
 - meny-systemet
 - ikonraden
 - Beskrivelse av enkelt-skjermbilder
 - navn, formål, eventuell kommentar og spesiell anmerkning
 - liste over lovlige verdier på skjermfeltene
 - rapporter (med eksempler), muligheter for utskrift
 - meldinger, feilmeldinger og behandling av feil
 - Beskrivelse av eventuelle BATCH-kjøringer¹⁵
 - navn, formål, eventuell kommentar og spesiell anmerkning
 - inndata, utdata, rapporter (med eksempler)
 - Beskrivelse av eventuelle manuelle rutiner
 - Beskrivelse av hjelpesystemet
- Spesial-situasjoner ved bruk av systemet
 - Hvordan håndtere feil? Ev. også liste over feilmeldinger.
 - Hvordan stoppe systemet og komme igang igjen?
 - Hvem skal kontaktes ved uforståelige feil?

4. IT (systemdokumentasjon/oversikt)

- Oversikt over systemet (diagrammer som limes inn fra tegneverktøyet):
 - kontekstdiagram (hvis nødv.) [omgivelser]
 - logisk/konseptuelt dataflytdiagram [oversikt]
 - datamodell [dataoversikt]
 - module data diagram (hvis ORACLE) [brukergrensesnitt + database]
 - module structure diagram (hvis ORACLE) [menyer mm.]
 - fysisk dataflytdiagram (hvis SAS) [detaljer i programmer, menyer mm.]
- Hvor er programmene lagret.
- Beskriv sentrale IT-tekniske fakta ved virkemåten til programsystemet (gjærne også med kopier av headingene til sentrale programmer).
- **Vedlegg** (frivillig)
 - Eventuelle dataflytdiagrammer (av ulike typer) på nivå høyere enn nivå enn oversikt (nivå 1)
 - Tabeller
 - Programlister
 - Spesifikasjoner (kopi eller referat fra “*Statistikkfaglig orientert kravspesifikasjon*”)

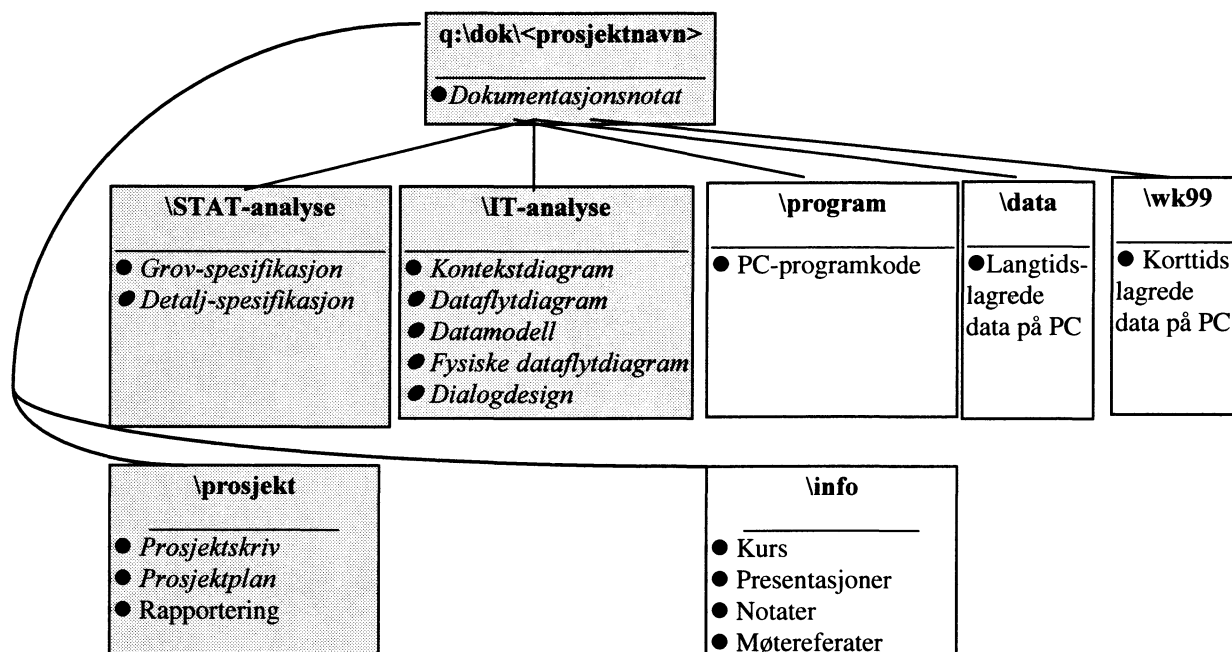
¹⁵ Dette punktet kan i mange tilfeller legges inn som en oversikt i tabellform.

7.2 Sjekk dokumentasjon

| | |
|----------------------|--|
| Idé | Sjekk at dokumentasjonen er i henhold til reglene i SSB. |
| Ambisjonsnivå | Obligatorisk |
| Notat | - |
| Mal | - |
| Ansvarlig | IT-kontorleder |
| Utførende | IT-kontorleder |

Beskrivelse:

1. Det er IT-kontor-lederens oppgave å kvalitetssikre at dokumentasjonen foreligger. Her må det understrekes at kvalitetssikringen **ikke** er kontroll av faglig innhold i dokumentene, men å se til at all dokumentasjon faktisk foreligger, og at den ligger på riktig plass på Q-disken.
2. Underveis i prosjektet er alle viktige produkter lagret på PC under Q-diskens katalog «DOK» med prosjektkatalog bestemt ved navnet etter prosjektet: Q:\DOK\



Figur: Oversikt over hva som ligger hvor på Q-disken

NB! Bare de gule (grå) boksene utgjør obligatoriske underkataloger for prosjektkatalogen.

Erfaringer:

1. Det kan være lurt å lage enda et nivå av kataloger. Eksempler på dette er:
 - Når et prosjekt er delt opp i delprosjekter, kan det være lurt å dele hovedprosjektkatalogen opp i delprosjektkataloger. Vi får da en struktur som følger navnestandarden på UNIX.
 - Noen kataloger blir så fulle av filer at det blir vanskelig å holde oversikten. Det lages da underkataloger. Underkatalogene kan da organiseres slik som punktene på figuren ovenfor viser: q:\dok\
2. Umiddelbart under katalogen q:\dok\

7.3 Igangkjøring

| | |
|----------------------|---|
| Idé | IT-systemet leveres til brukergruppen og dokumentasjonen i SSB. |
| Ambisjonsnivå | Obligatorisk. Brukerne av systemet har testet og godtatt at systemet er i samsvar med spesifikasjonene. Dokumentasjonen godkjent av IT-leder. |
| Notat | - |
| Ansvarlig | (Hoved-)prosjektleder |
| Utførende | IT-medarbeider og statistikk-medarbeider sammen med IT-leder |

Beskrivelse:

1. Statistikk-medarbeidere og IT-medarbeidere tester det endelige systemet sammen. Hvis systemet fungerer som forventet, overleveres systemet til de kommende brukerne av systemet sammen med dokumentasjonen.
2. Bestem hvem som er ansvarlig for endringer i systemet.
3. Send ut dokumentasjonsnotatet i serien *Interne dokumenter*.
4. En tur på byen når alt er OK.
5. Alt som gjøres etter dette i prosjektet er vedlikehold (det vil si at utviklingsprosjektet eller konverteringsprosjektet er avsluttet).

Erfaringer:

Dokumentasjonen er en del av produktet som blir laget. Dårlig dokumentasjon gir et dårlig system.

8. OPPSUMMERING



Prosjektinitiering

❶ *Prosjektskriv* for hele prosjektet. Prosjektskrivet inneholder noen basisopplysninger om prosjektet. Målet med prosjektet, hvor lang tid det forventes å ta og litt om behov for ressurser. (Hoved-)prosjektlederen har ansvaret for denne fasen, og fasen er beskrevet i detalj i «Håndbok i prosjektstyring».

❷ Opprette infrastruktur

- Det opprettes en prosjektgruppe som består av minst én IT-medarbeider ("systemutvikler") og minst én statistikk-faglig medarbeider ("bruker").
- Hvis du bruker UNIX, må kundestøtte lage en stammekatalog på UNIX (Se notat for «Navnestandard på UNIX».)
- Sørg for at kundestøtte oppretter en katalog `q:\dok\ hvor alle som jobber i prosjektet har skrivetillatelse.`
- Prosjektnavn skal være det samme som stammekatalog på UNIX (hvis en slik benyttes i prosjektet).



STAT-analyse

❸ Fagseksjonen starter opp arbeidet med *statistikkfaglig forarbeid*. I denne teknikken gjør de seg kjent med dataene og velger metode. Arbeidet varierer i omfang og form etter type prosjekt.

❹ Når dette er gjort kan statistikk-medarbeiderne sammen tegne en skisse over systemet i et *kontekstdiagram* for å få oversikt over hvilke data som går inn og ut av systemet og hvem som eier disse dataene.

❺ *Prosjektplan* og *Avklaring med styringsgruppa* sikrer at prosjektet er på riktig spor i forhold til styringsgruppas ønsker.

❻ *Statistikkfaglig orientert kravspesifikasjon* danner utgangspunkt for (logisk) dataflytdiagram og eventuell datamodell i IT-analysefasen. Her skal statistikk-medarbeiderne beskrive det systemet de ideelt ønsker seg, uten å bruke IT-språk.



IT-analyse-fasen

I denne fasen IT-analyses det systemet som skal lages (*kravspesifikasjon*). IT-analysen lages i samarbeid mellom statistikk-medarbeideren og IT-medarbeideren. Det anbefales å bruke "prototyping" i prosessen: IT-medarbeiderne og statistikk-medarbeiderne jobber da sammen om å lage og knytte sammen skjermbilder mm.. Arbeidet er ferdig når brukergrensensnittet fungerer tilfredsstillende.

❼ På grunnlag av de statistikkfaglige medarbeidernes «statistikkfaglig orientert kravspesifikasjon» tegner IT-medarbeiderne og de statistikkfaglige medarbeiderne sammen opp en oversikt over systemet (*dataflytdiagram*). For å sikre at diagrammet er så riktig som mulig, lønner det seg ofte å tenke litt mer detaljert enn det en tegner opp på diagrammet. Resultatet skal foreligge i et CASE-verktøy.

❽ IT-medarbeiderne (med hjelp fra de statistikkfaglige medarbeiderne) lager nå en *datamodell* for å få oversikt over sammenhengen i dataene. Får en først til en god datamodell, er systemet godt forstått! Datamodellering skal alltid brukes i store prosjekter og i prosjekter hvor relasjonsdatabaser inngår. Resultatet skal foreligge i et CASE-verktøy.

❾ Alle datalagrene dokumenteres i systemet DATADOK. Dette kalles å lage "*datadefinisjon*".

❿ Hvis det inngår mange skjermbilder i løsningen, tegner IT-medarbeideren en *dialogdesign* som viser sammenhengen mellom bildene og hvordan du kommer fra et bilde til det neste.

⓫ De statistikkfaglige medarbeiderne (i samarbeid med IT-medarbeiderne) skriver *detaljspesifikasjonen* i et tekstbehandlingsverktøy (Word). Hver prosess og datalager fra teknikken «logisk dataflytdiagram» beskrives med et avsnitt eller kapittel i detaljspesifikasjonen. IT-medarbeideren jobber altså aktivt med spesifikasjonen direkte med de statistikkfaglige medarbeiderne (ikke bare mottar dokumenter eller enda verre - får en del «uforpliktende» spesifikasjoner i telefonen).

- Kvalitetssikring av arbeidet kan gjøres ved at de statistikkfaglige medarbeiderne og IT-medarbeiderne sammen går gjennom diagrammene og detaljspesifikasjonen og forsøker å finne flest mulig feil og mangler. Spesifikasjonen er ferdig når det ikke lenger synes å være åpenbare mangler i detaljspesifikasjonen eller diagrammene.



Gjennomførings-fasen

⓬ Programmene kodes i det valgte verktøy: I starten av hvert program lages et «hode» som blant annet angir hva programmet gjør, hvem som har skrevet det og beskrivelse av inn- og ut-filer/tabeller.

⓭ Programkoden testes. Dersom det på tross av kvalitetssikringen i IT-analysefasen skulle oppdages feil eller mangler i detaljspesifikasjonen, er det selvsagt mulig å gå tilbake og gjøre deler av IT-analysefasen om igjen («iterasjon»). Men husk at iterasjoner tar tid!

⓮ IT-medarbeiderne tegner opp *fysisk dataflytdiagram* som viser sammenhengen mellom programmene.

- IT-løsningen gjennomgår en *akseptansetest* på at systemet fungerer i samsvar med de spesifikasjonene de statistikkfaglige medarbeiderne gav.



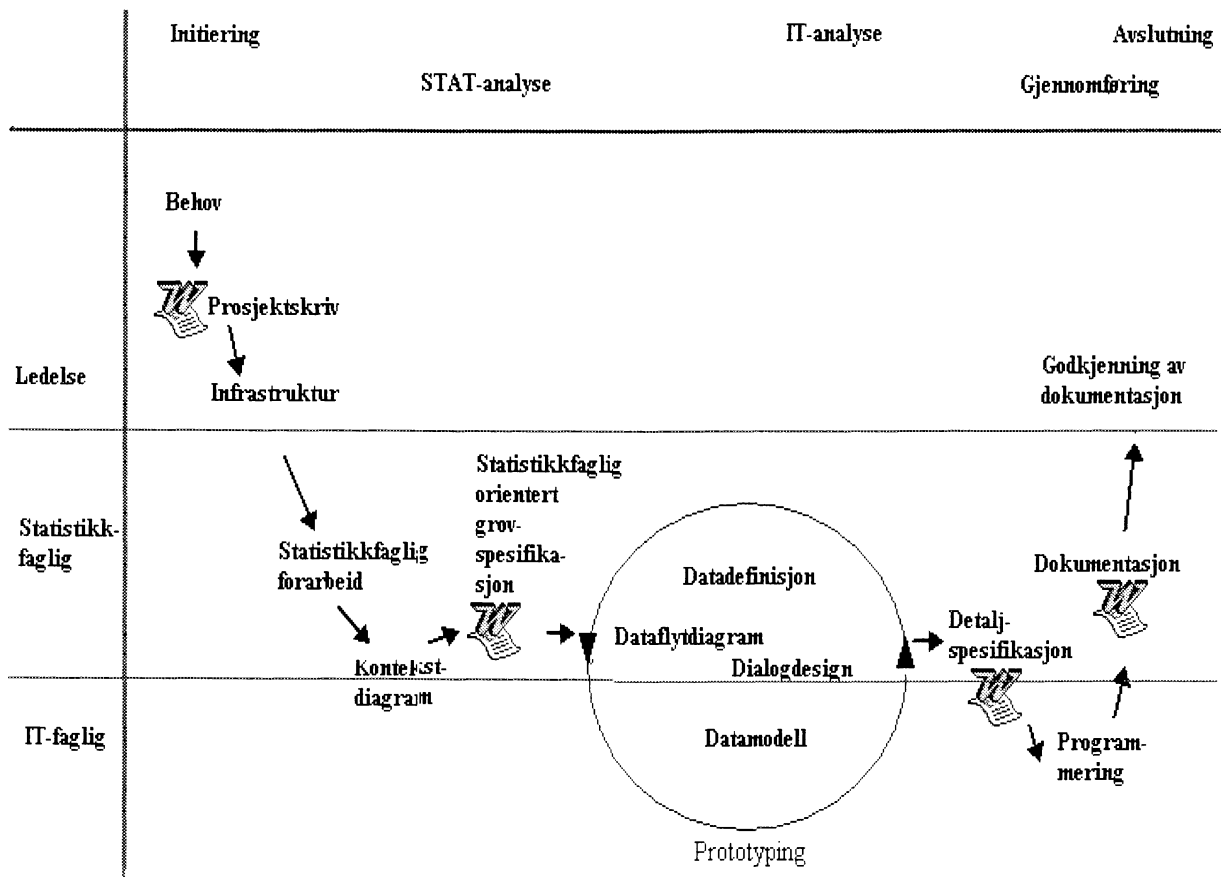
Avslutnings-fasen

⓯ De statistikkfaglige medarbeiderne og IT-medarbeiderne samarbeider om å skrive *dokumentasjonsnotatet*.

⓰ Avdelingens IT-leder påser at dokumentasjonen på katalogen `q:\dok\ er i henhold til retningslinjene`

⓱ Igangkjøring av systemet. Dokumentasjonsnotatet gis ut i serien *Interne dokumenter*.

Vi er nå ved veis ende i prosjektet. La oss sammenfatte det alt det vi har lært i følgende sterkt komprimerte figur:



Figur 8.1: Håndbokens idé presentert i kortform.

9. ORDLISTE

- ✎ angir at dette er en teknikk som inngår i en av fasene i arbeidet med statistikkprosjektet
- ◆ angir at dette er en fase i arbeidet med statistikkprosjektet
- ☺ angir at dette er en deltaker i arbeidet med å lage systemet
- ⇒ angir annet som angår arbeidet med statistikkprosjektet

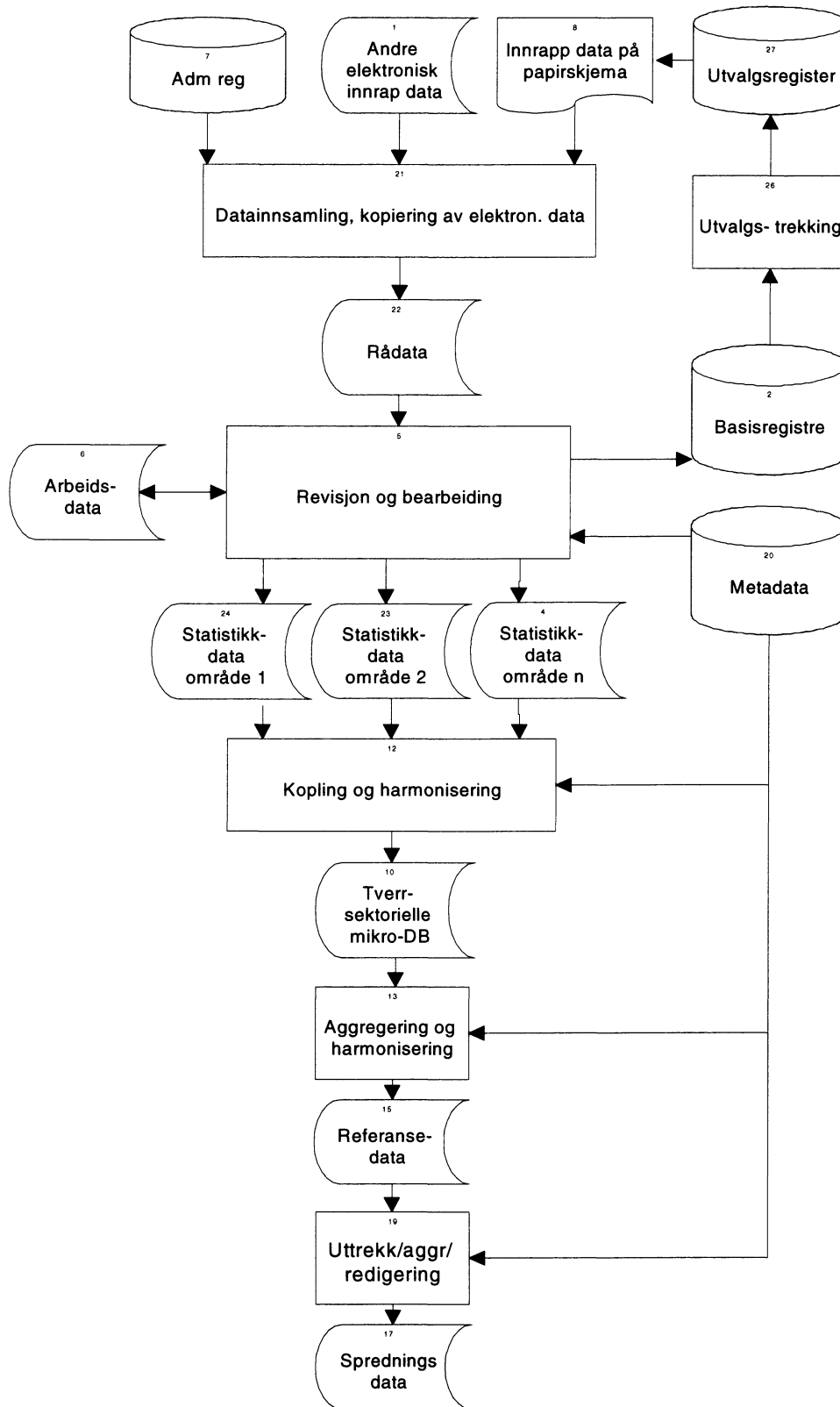
- ⇒ **aktiviteter**: De arbeidsoperasjoner som en prosess kan deles inn i. Se innledningskapitlet!
- ⇒ **analyse**: Hoveddel av metoden for å lage statistikkprosjekter. 2 delt:
 - a) STAT-analyse som utføres av statistikkseksjonenes medarb.for å planlegge innholdet i et system
 - b) IT-analyse som utføres for å omforme kravene fra statistikk-seksjonen til en detaljspesifikasjon som IT-medarbeiderne bruker som utgangspunkt for å lage statistikk-systemet.
 For øvrig brukt som statistisk begrep i forbindelse med analyse av data.
- ◆ **IT-analyse(fasen)**: *Se over!*
- ⇒ **attributt**: Egenskap ved en entitet. Se teknikken *datamodellering!*
- ◆ **avslutning(sfasen)**: Prosjektet avsluttes og det sjekkes at IT-koden og dokumentasjonen er i orden.
- ⇒ **bearbeiding**: Ofte kalt «massering av data», «transformering av data» eller «tilrettelegging av data». Mellom input og lagring av data i SSB (nevnes i *statistikkfaglig orientert grovspesifikasjon*).
- ⇒ **beslutningspunkt**: Ved avslutningen av en fase, før en ny fase påbegynnes, skal det besluttes om neste fase skal gjennomføres. Sammenlign fig. i innledningskapitlet!
- ✎ **blackbox-testing**: Testing av programmer uten å kjenne koden (dvs. sende inn data og undersøke om output-data er riktige).
- ☺ **bruker**: Den som skal bruke IT-systemet som lages. Se særlig fasen *STAT-analyse*.
- ⇒ **CASE-verktøy** (= *Computer Aided Software Engineering*): Verktøy for å automatisere deler av *systemutviklingsprosessen*. Vi har flere typer CASE-verktøy:
 - ⇒ *upper CASE*: administrasjon og dokumentasjon
 - ⇒ *lower CASE*: kodegenerering
 - ⇒ I-CASE: *upper-CASE + lower-CASE*
- ✎ **CBA** (= *Cost benefit analysis*): Se *nytte-kostnadsanalyse!*
- ✎ **CRA** (= *critical requirement analysis*) Ofte kalt “kritisk kravanalyse”!
- ✎ **dataflyt**: Hvordan data beveger seg fra datalager til datalager via prosesser. Dataflyt er av to typer: *logisk dataflyt* og *fysisk dataflyt*. Logisk dataflyt er igjen av to typer: *kontekstdiagram* og *dataflytdiagram nivå 1...n*.
 - ✎ **kontekstdiagram**: Diagram som viser hvilke kilder og mottakere av data som systemet har. Brukes i fasen *STAT-analyse*.
 - ✎ **logisk dataflytdiagram nivå 1** (DFD1): Diagram som viser en grov oversikt over IT-systemet som skal lages med dets omgivelser. Inneholder *prosesser* og *datalagre* og sammenhengen (*dataflyten*) mellom dem. Brukes i fasen *IT-analyse*.
 - ✎ **logisk dataflytdiagram nivå 2** (DFD2): 1. nedbrytning av logisk dataflytdiagram (DFD1). Delprosesser med tilhørende dataflyt. Brukes i fasen *IT-analyse*. (Ikke obligatorisk i vår metode.)
 - ✎ **fysisk dataflyt** (DFDF) Se «*fysisk dataflytdiagram*»!
- ⇒ **datalager**: Logisk fil. Brukes i logiske dataflytdiagrammer.
 - ⇒ **manuelt lager** (mappe i kontorskuff, listing osv.)
 - ⇒ **datamaskinbasert lager** (sekvensiell fil, databasetabell, SAS-datasett osv.)
 - ⇒ **temporært lager** (midlertidig lagring)
- ✎ **datamodell(ering)** (DM): Sammenhengene i data: entiteter, attributter og nøkler. Teknikken som brukes i IT-analysefasen.

- ⇒ **delprosess**: Del av en hovedprosess i DFD1.
- ◆ **design(fasen)**: I vår metode er det en del av *gjennomføringsfasen*. Dette er egne faser i *Systems Engineering*:
 - ⇒ - logisk design
 - ⇒ - fysisk design
- ⇒ **Designer 2000**: CASE-verktøy i *Developer 2000 (ORACLE)*.
- ✍ **detaljspesifikasjoner**: Detaljerte spesifikasjoner fra statistikkfaglige medarbeidere til IT-medarbeider om hvordan systemet skal fungere. Teknikk i IT-analysefasen.
- ✍ **DFD0** = kontekstdiagram (av og til betegnet som *dataflytdiagram nivå 0*).
- ✍ **DFD1** = logisk dataflytdiagram nivå 1.
- ✍ **DFD2** = logisk dataflytdiagram nivå 2.
- ⇒ **diagram**: Resultat av tegneteknikk.
- ✍ **dialogdesign (= dialogsystem)**: Teknikk for å tegne opp sammenhengen mellom skjermbildene. Teknikk i IT-analysefasen.
- ✍ **DM** = *datamodellering*.
- ✍ **dokumentasjon**: Tilrettelegging og presentasjon av informasjon om det som skal gjøres eller allerede er gjort.
- ✍ **ELH (= entity life history)**. Se «*entitetenes livssyklus*»!
- ⇒ **entitet**: Et dataelement/objekt (f.eks. person, foretak, bolig).
- ✍ **entitetenes livssyklus (ELH)**: Historien til en entitet: oppretting, endring, sletting.
- ✍ **estimering**: Planlegging av ressurser: prosjektdeltakernes tidsforbruk, maskinressurser osv.
- ⇒ **event** = *hendelse*.
- ◆ **fase**: Grovste oppdeling av systemutviklingsarbeidet (se innledningskapitlet!). I vår metode er det 4 faser (prosjektinitiering, IT-analyse, gjennomføring og avslutning).
 - ⇒ Hver fase er oppdelt i *aktiviteter*.
 - ⇒ Hver aktivitet er oppdelt i konkrete (arbeids)*oppgaver*.
- ✍ **Flowchart** = *fysisk dataflytdiagram*.
- ✍ **fysisk dataflytdiagram**: Oversikt over systemet som er laget. Skjematisk sammenheng mellom programmer og sekvensielle filer/databasetabeller. Se kapitlet for *gjennomføringsfasen*!
- ⇒ **funksjon**: Begrep i (*logisk*) *dataflyt*-teknikken i *System Engineering*. Ikke benyttet i vår metode!
- ✍ **funksjonsanalyse**: Ligner vår (*logisk*) *dataflyt*analyse.
- ⇒ **fysisk design**: Begrep ikke benyttet i vår metode. Se *design(fasen)*!
- ◆ **gjennomføring(sfasen)**: Koding, testing og dokumentasjon av dette.
- ⇒ **hendelse (= event)**: Begrep som kommer mer inn med objektorientering. En hendelse setter i gang («trigger») en prosess.
- ⇒ **IE** = *Information Engineering*.
- ◆ **avslutning**: Iverksetting av systemet som lages (installering m.m.) Se fasen *avslutning*.
- ⇒ **Information Engineering**: *Systemutviklingsmetode*, konkurrent til *Systems Engineering*.
- ⇒ **Inkrementell metode**: *Modell for systemutviklingsmetode* som baseres på at systemet ikke kan spesifiseres i detalj før kodingen starter. Motsatt av «*fossefallsmodellen*». Ved *inkrementell* utvikling utvikles systemet bit for bit, for så til slutt å ende opp med et komplett system (som å bygge med legoklosser). Man velger gjerne ut den viktigste delen av det nye systemet, og lager denne først. Deretter bygger man på med stadig nye deler, slik at man til slutt får et komplett system. Inkrementell systemutvikling Det er mange fordeler med denne formen for systemutvikling, men det er alltid en fare for at en ny modul kan utløse feil/svakheter i systemet, noe som vil føre til forsinkelser i prosjektet.
- ✍ **inspeksjon**: (sml. *Fagans* metode for gjennomgang av hvorvidt dokumenter tilfredsstillers kvalitetskrav).
- ⇒ **installasjon**: Se *avslutningsfasen*.
- ⇒ **iterasjon**: Gjentatte gjennomganger av en fase med gradvis bedre resultat (forhåpentligvis).
- ☺ **IT-medarbeider** (= systemutvikler = IT-ansatt). Person ansatt på en IT-seksjon/gruppe.

- ✍ **JAD** (= *Joint Application Development*). Se «*lagarbeid*».
- ✍ **koding**: Programmering. Se også «*faser*».
- ⇒ **kravspesifikasjon**: (Godkjente) spesifikasjoner fra brukerne av systemet.
- ✍ **kritisk kravanalyse** (*CRA*). Teknikk i IT-analysefasen i *System Engineering*. Brukes ikke i vår metode.
- ⇒ **kvalitetssikring** (= *QA*). Kvalitetssikring er ivaretatt på mange måter i vår metode. Følgende teknikker:
 - ✍ *IT-analysefasen*: Tegne sammen diagrammer for å sjekke at alle har samme *modellen* av systemet som skal lages.
 - ✍ *Gjennomføringsfasen*: *Testing* av programkoden m.m.
 - ✍ *Avslutningsfasen*. «*Sjekk dokumentasjon*» og «*Igangkjøring*».
 - ✍ I den grad arbeidet utføres i et prosjekt med flere deltakere kontrolleres resultatene automatisk og løpende.
- ✍ **lagarbeid**: Arbeidsform hvor alle prosjektdeltakerne jobber sammen. IT-medarbeiderne lager diagrammer og statistikkfaglige medarbeidere skriver tekst. Se fotnote i *IT-analysefasen*.
- ✍ **logisk design**: Fase i *System Engineering*. Benyttes ikke i vår metode. Se *design!*
- menysystem**: Skjema over alle skjermbildene og deres tilhørende programmer. Se *dialogdesign*.
- ⇒ **metode**: Begrepet betegner formalismen bak en metodisk arbeidsform. Brukes her ensbetydende med *systemutviklingsmetode* (se kapittel 1). En sentralt element i håndboka.
- ☺ **metodeansvarlig**: Gruppe med spesielt ansvar for vedlikehold og opplæring i IT-metoden.
- ⇒ **milepel**: Dato når en *fase* eller viktig *aktivitet* i prosjektet er avsluttet.
- ⇒ **modul**: Samling av programmer som hører naturlig sammen.
- ✍ **nytte-kostnadsanalyse** (= *Cost Benefit Analyse (CBA)*). Teknikk for på forhånd å beregne kostnader og gevinster ved å utføre et prosjekt sammenlignet med ikke å utføre prosjektet.
- ⇒ **objektorientering**: Programmering basert på å modellere *objekter* som har egenskaper (*attributter*) som utfører handlinger (prosedyrer) + arving fra «foreldre»/(parent)-objekter.
- ⇒ **oppgave**: Den minste arbeidsoperasjon i metoden. Se innledningskapitlet og *fase!*
- ⇒ **produkter**: Resultat av teknikk (*diagram, spesifikasjon, program* m.m.)
- ✍ **programspesifikasjoner**: Se *detaljspesifikasjoner!*
- ⇒ **prosess**: Enhet i en grov oppdeling av IT-systemets bearbeiding av data. Komponent i (*logisk dataflyt*-diagrammet. Se innledningskapitlet og *IT-analysefasen*.
- ⇒ **prosjekt**: Arbeidsform hvor personer fra ulike faggrupper jobber sammen for å nå et mål innen en tidsfrist. Engangsoppgave med et klart mål og klare ressurs- og tidsrammer. Se innledningskapitlet!
- ☺ **prosjekteier**: Den som har satt i gang prosjektet og skal bruke resultatet.
- ◆ **prosjektinitiering**: Se under «*faser*»!
- ☺ **prosjektgruppe**: Summen av de personene som deltar i prosjektet.
- ☺ **prosjektleder**: Den som leder prosjektet. (Dersom prosjektgruppa består av bare én person, er denne personen (hoved-)prosjektleder.) Se innledningskapitlet!
- ✍ **prosjektskriv**: SSBs tradisjonelle *prosjektskriv*. Se kapittel for *prosjektinitiering!*
- ⇒ **prosjektstyring** (PS): Rutiner for å planlegge og følge opp prosjektet. Dette ligger på siden av *systemutviklingsmetoden*, men de er nyttige å bruke sammen (synergieffekt).
- ✍ **prototyping**: Se kapittel for *IT-analysefasen*.
- ⇒ **QA** = *Quality assurance*. Se *kvalitetssikring*.
- ⇒ **RAD** = *Rapid application development*: Utvikling vha. ferdigskrevne standardmoduler og kodegenerering (se *kodefasen*).
- ☺ **referansegruppe**: Gruppe av spesielt kvalifiserte personer som kan delta i vurdering av diverse oppgaver i prosjektet.
- ⇒ **SE-Express**: Engelsk *systemutviklingsmetode* levert av *ISI A/S*. Utgangspunktet for SSBs *systemutviklingsmetode*.
- ⇒ **SE-ONLINE**: Hjelpfunksjon til *SE-Express*. Finnes på PC.
- ⇒ **SSADM**: *Systemutviklingsmetode* som ble utarbeidet som standard for forvaltningen i Storbritannia. Forløper for *System Engineering*.

- ◆ **STAT-analyse:** Fase i arbeidet med statistikkprosjektet. Statistikkmedarbeiderne identifiserer sine krav til det statistikk-systemet som skal lages.
- ⇒ **styleguide:** Retningslinjer for hvordan du skal lage skjermbilder og *dialoger* i grafiske brukergrensesnitt.
- ☺ **styringsgruppe:** Prosjektets øverste beslutningsmyndighet. (Hoved-)prosjektlederen rapporterer til denne.
- ⇒ **SU = systemutvikling.**
- ⇒ **systemering:** Arbeidet med å analysere IT-delen av et system. Tilsvarende *IT-analysefasen* og deler av *prosjektinitieringsfasen*.
- ⇒ **System Architect:** CASE-verktøy brukt i SSB 1992-1996, levert av *Popkins Software* (USA).
- ⇒ **System Engineering (SE):** Kommersialisering av *SSADM*. *SE-Express* er en forenkling av SE. Vår metode er igjen en forenkling av SE-Express.
- ☺ **systemutvikler:** IT medarbeider som utformer løsningen for IT-systemet.
- ✍ **systemkart = fysisk dataflytdiagram.**
- ⇒ **systemutvikling (SU):** Innbefatter hele arbeidet med utforming av et informasjonssystem, både systemering og koding (dvs. alle fasene i *systemutviklingsmetoden*).
- ⇒ **systemutviklingsmetode (SU-metode):** *Metoden* en bruker når en vil arbeide med *systemutvikling*. Se definisjon i innledningskapitlet! Sentral del av håndboka.
- ✍ **teknikk:** Standardisert måte å løse et problem på. *Teknikkene* er presentert i kapitlene fra *prosjektinitiering* til *avslutning*.
- ☺ **teknisk prosjektleder:** IT-faglig person som leder den IT-faglige delen av prosjektet.
- ✍ **testing:** Teknikker for å finne feil.
 - ✍ *modultest:* programtest
 - ✍ *systemtest:* test av sammenhenger mellom programmer
 - ✍ *akseptansetesting:* test av systemet som helhet
- ⇒ **testplan:** Plan for å teste ferdige produkter (*program, spesifikasjon* m.m.).
- ⇒ **tidsplan:** Oversikt over hva de enkelte prosjektdeltakerne skal gjøre i de ulike *fasene* i prosjektet.
- ⇒ **transaksjon:** Begrep i (*logisk*) *dataflytdiagrammet* i *System Engeneering*. ikke benyttet i vår metode.
- ⇒ **vaske data:** Begrep fra *datawarehouse*. Konsistenssjekking m.m. av inputdata.
- ✍ **white-box-testing:** Testing om programmet er riktig ved å lese programkode.
- ⇒ **3. normalform:** Se *datamodellering* i kapitlet om *IT-analysefasen*.

10. VEDLEGG: Typisk dataflyt og datalagertyper i SSB



Figur: Typisk dataflyt og datalagertyper i Statistisk sentralbyrå.

11. VEDLEGG: Hvilke verktøy skal brukes til hva?

I denne håndboka er det lagt vekt på å beskrive utviklingen av et statistikk-system mest mulig uavhengig av verktøy. Her er likevel noen gode råd om verktøybruk:

| Type data | Datamodell | Dataflyt | Fysisk dataflyt | Dialogsystem | Definisjoner |
|----------------------------------|------------------------------------|--------------------|--------------------|---------------------|--------------------------------|
| Sekvensielle filer ¹⁶ | ABC Flowcharter/ Developer 2000 | ABC Flowcharter | ABC Flowcharter | - | DATA-DOK |
| SAS-datasett | ABC Flowcharter/ Developer 2000 | ABC Flowcharter | ABC Flowcharter | ABC Flowcharter | DATA-DOK SAS |
| FAME | - | - | ABC Flowcharter | ABC Flowcharter | FAME |
| ORACLE | Developer 2000 | Developer/2000 | - | Developer/2000 0 | DATA-DOK Developer/ 2000 |

12. REFERANSER

- Svein Jåvold (1998): "Dokumentasjonsnotat for Datadok", *Interne dokumenter*, SSB
- Håkon Berby (1994): "Datalagring på Unix i Statistisk sentralbyrå, Retningslinjer, 2. reviderte utgave", *Interne dokumenter*, 96/14, SSB
- Frank Foyn (red.) (1998): *Revisjonshåndbok*, SSB.
- Rune Gløersen (red.) (1996): "Håndbok i prosjektstyring", *Interne dokumenter*, SSB
- Knut Vidar Hoholm og Svein Jåvold (1992): *Rapport fra prosjektet Metodisk systemutvikling*, SSB
- Sverre Hove (red.) (1993): *Datafangsprogrammet*, Utredning fra et utvalg nedsatt av statistikkmøtet, SHO, LSO, BBL, BOT, JLY, HMM/02.06.93, SSB.
- Svein Jåvold & Knut Vidar Hoholm (1993): *Skal SSB innføre systemutviklingsmetode?*, SSB
- Svein Jåvold (1995): *Minimetode*, SSB.
- Svein Jåvold (1997): q:\dok\notater\handbok\abc.doc (*Minihåndbok i ABC Flowchart*)
- Truls Thirud & Svein Jåvold (red.) (1996): "Håndbok for programmerere", *Interne dokumenter*, 97/12, SSB
- Stein Turtumøygard & Håkon Berby (1995): "Kort innføring i ny navnestandard for UNIX-filer", *Interne dokumenter*, 95/17, SSB

¹⁶ Brukt i forbindelse med *Visual Basic* eller lignende.

13. NOEN ERFARINGER ETTER BRUK AV METODEN

Når fagseksjonen vet hva IT-medarbeiderne har behov for og IT-medarbeiderne vet hva fagseksjonen har behov for, er grunnlaget for et godt prosjektarbeid lagt! Når fagseksjonen vet hva IT-medarbeiderne har utført og IT-medarbeiderne vet hva fagseksjonen har utført, er grunnlaget for en god oppfølging lagt!
Eспен Sørensen

Jeg har faktisk begynt å bruke metodeopplegget dere har laget, dokumentasjonen og alt. Det TAR mye tid å dokumentere, men det er faktisk litt moro også, og faktisk temmelig nyttig iblant...
Ove Viggen

Jeg ser, når jeg leser håndboka, at den vil være et nyttig verktøy også for slike som meg, som skal følge opp prosjektene. Og jeg merker meg at IT-kontorleder er ansvarlig for at det blir utarbeidet dokumentasjon. Jeg tror det vil virke positivt, også for den jobben jeg er satt til å utføre.
Rune Gløersen

Å bruke en felles metode for systemutvikling og dokumentasjon er en forutsetning for å organisere våre data hensiktsmessig for produksjon, gjenbruk og analyse på tvers av SSBs mangfoldige statistikk- og analyseområder.
Håkon Berby

Intet er mer oppklarende enn en god figur. Dataflytdiagrammer er helt nødvendige for kommunikasjon i prosjektet.
Leiv Solheim

De sist utgitte publikasjonene i serien Statistisk sentralbyrås håndbøker

- | | | | |
|----|---|----|--|
| 45 | Håndbok i datasikkerhet og fysisk sikring. 1994. 53s. | 55 | Nordisk statistikk på CD-ROM: Veiledning. 20s. |
| 46 | Telefonkatalog. 1997. 91s. | 56 | PC-Axis versjon 2.2: Brukerhåndbok. 69s. |
| 47 | EØS-avtalen. Det statistiske samarbeid og konsekvenser for Statistisk sentralbyrås statistikkproduksjon. 1994. 55s. | 57 | Produktregister versjon 4.0: Brukerveiledning. 49s. |
| 48 | Håndbok i tilsettingssaker. 1994. 32s. | 58 | Håndbok i prosjektstyring. 20s. |
| 49 | Oppgaveplikt og tvangsmulkt. 1995. 55s. | 59 | Personalreglement for Statistisk sentralbyrå. 22s. |
| 50 | Emneinndeling 1995. 1995. 43s. | 60 | Produktnummerkatalog pr. 28.02.1996. 55s. |
| 51 | Intervju: EDB-arbeidsbok. 1995. | 61 | Innkjøpshåndbok. 1996. |
| 52 | Intervju: EDB-oppslagsbok. 1995. | 62 | Timeplan versjon 3.0: Brukerveiledning. 16s. |
| 53 | Intervju: Opplæring og administrasjon. 1995. | 63 | Håndbok i EDB-metode. 52s. |
| 54 | Internkontroll: Revidert utgave 1997. 25s. | 64 | Publiseringshåndbok: Regler og retningslinjer for publisering i Statistisk sentralbyrå. 93s. |



Statistisk sentralbyrå
Statistics Norway