

# Interne notater

## STATISTISK SENTRALBYRÅ

34/4

17. februar 1984

### OM BRUKEN AV STANDARDPROGRAM I STATISTISK SENTRALBYRÅ

Av

Håkon Berby, Hanne Modahl og Tormod Solvin

#### INNHOOLD

	Side
1. INNLEDNING .....	2
2. BRUK AV STANDARDPROGRAM I BYRÅET I DAG .....	3
2.1 Innledende merknader om oppgaven, programtyper og data .....	3
2.2 Registrering og klargjøring av data .....	5
2.2.1 Registrering, kontroll og korreksjon av data .....	5
2.2.2 Redigering, kopling og omkodning av data og andre filbehandlingsfunksjoner .....	6
2.3 Resultatuttak .....	8
2.3.1 Ulike typer resultatuttak med standardprogram .....	8
2.3.2 Produksjon av råtabeller og innledende undersøkelse av data .....	8
2.3.3 Produksjon av trykkeklare tabeller .....	10
2.3.4 Statistisk analyse .....	12
2.3.5 Standardprogram for utlisting, slipp- og skjemautfylling ...	14
2.3.6 Programpakker for resultatuttak med fil- og databehandlings- funksjoner .....	14
2.4 Oversikt over programmer på Byråets maskiner .....	15
2.5 Om bruk av standardprogram på fagkontorene til nå .....	16
2.5.1 Innledning .....	16
2.5.2 Programutvalg .....	17
2.5.3 Fagkontorenes bruk av standardprogram .....	18
3. STANDARDPROGRAM I TIDEN FRAMOVER .....	19
3.1 Innledning .....	19
3.2 Hjelpemidler for programmerere .....	20
3.3 Registrering og klargjøring av data .....	20
3.3.1 Registrering, kontroll og korreksjon av data .....	20
3.3.2 Redigering, kopling og omkodning av data foruten andre filbehandlingsfunksjoner .....	21
3.4 Resultatuttak .....	22
3.4.1 Produksjon av råtabeller og rapportgenerering .....	22
3.4.2 Trykkeklare tabeller .....	22
3.4.3 Statistisk analyse .....	24
3.5 Program som håndterer databaser .....	25
3.5.1 Generelt om databasesystemer og språk for å håndtere databaser	25
3.5.2 Utvikling og bruk av databaser i Byrået .....	30
3.5.3 Tilknytning til eksterne databaser .....	32
3.6 Andre programsystemer .....	33
3.6.1 Standardprogrammer for grafisk framstilling .....	33
3.6.2 Standardprogram via mikromaskiner .....	33
4. AVSLUTTENDE MERKNADER .....	34

## 1. INNLEDNING

Målet for virksomheten i Byrået er å lage den statistikken som samfunnet har behov for til enhver tid. Disse målene søkes realisert i Byråets statistikkproduksjonsplaner.

Statistikken lages nå i det alt vesentlige ved hjelp av EDB. Rasjonell statistikkproduksjon krever da at EDB-verktøyene er så godt som mulig tilpasset Byråets statistikkproduksjonsplaner. Det betyr for det første at programmene bør være tilpasset det statistikkprodukt som Byrået publiserer. For det annet betyr det at programmene bør være tilpasset slik at de kan brukes også når behovene og statistikkproduktet endrer seg. Hvis programmene ikke er brukbare når behovene endrer seg, må vi utvikle eller skaffe oss nye programverktøy som kan tilfredsstille disse behovsendringene.

For effektiv produksjon bør dessuten programmene være slik at saksbehandlerne selv kan spesifisere uttak av statistikken fra dataene som er lagret ved hjelp av EDB-utstyr. Jo flere ledd som deltar i produksjonen av statistikken, dess mer tid og ressurser går med til kommunikasjon. Det øker mulighetene for kommunikasjonssvikt med feil og forsinkelser som resultat.

Notatet vil behandle følgende spørsmål: Først, hva er dagens situasjon? Hvilke programverktøy har vi? Hvordan er situasjonen for brukerne av disse programmene på fagkontorene? Deretter, hvilke muligheter har vi i tiden framover til å ta nye programverktøy i bruk? Notatet vil ikke behandle valg av konkrete program for den framtidige utviklingen. Egne arbeidsgrupper bør evaluere ulike alternativ for de ulike oppgavene og søke å veie dem mot kravspesifikasjoner for at vi skal få hensiktsmessige programverktøy.

## 2. BRUK AV STANDARDPROGRAM I BYRÅET I DAG

### 2.1 Innledende merknader om oppgaver, programtyper og data .

De av Byråets databehandlingsoppgaver som dette notatet dekker er dataregistrering, klargjøring av data og sluttbruk. Sluttbruk betyr her vanlig statistikkproduksjon og analyse av data i ulike former.

Byrået har til det siste dekket oppgavene klargjøring og sluttbruk av data ved hjelp av EDB. I løpet av de siste årene er dataregistreringsfunksjonen også blitt understøttet av EDB-program; først i en spesielløsning for industristatistikken, siden i standardprogram for dataregistrering, kontroll og korreksjon (DOKS).

Vi kan dele inn standardprogrammene i to kategorier på grunnlag av kjøremåte. Den første kategorien er program for satsvis kjøring. En kjøring kan bestå av ett eller flere program som sendes inn i maskinen i form av en ferdig sats. Satsen inneholder all informasjon om hva kjøringen skal omfatte med påkalling av program og datafiler, direktiv til standardprogram, etc. Brukeren "bestemmer på forhånd" alle spesifikasjonene i satsen. Eventuelle feil i parametre eller data må inspiseres i utskrift etter at satsen er ferdig kjørt, feilene må rettes og hele (eller noen ganger deler av) satsen kjøres på ny. De fleste program for satsvis kjøring stiller krav til at brukerne kan styrespråket (JCL) på den aktuelle maskinen i tillegg til at de må kunne standardprogrammets språk. Dette kompliserer gjerne bruken av de satsvis orienterte programmene.

Den andre kategorien er program for interaktiv behandling. Brukeren sitter da i en dialog med standardprogrammet ved en terminal til maskinen, f.eks. en skjermterminal. Brukeren kan f.eks. ledes av programmet til å velge behandlingsvei ved at programmet presenterer "menyer" av alternative behandlingsveier. Brukeren kan også bli bedt om å gi spesifikasjoner i det bestemte kommandospråket til programmet innen bestemte rammer som programmet gir i form av skjermbilde. Brukeren får feilmeldinger i løpet av arbeidsprosessen og behøver ikke å gå lange veier tilbake for å rette og fortsette arbeidet.

En blanding av de to kategoriene kan også forekomme. F.eks.: Brukeren kan spesifisere oppgaven i dialog med et interaktivt program som bruker opplysningene til å forme og sette i gang en sats av "gammel type". Brukeren vil få meldinger om rent formelle feil i forhold til programmet i løpet av dialogen og kan rette disse umiddelbart. Derimot må han/hun vente til satsen er kjørt på maskinen og listen er ferdig før det er

mulig å si noe om resultatet svarer til forventningene. Ved feil må brukeren gå tilbake til det interaktive programmet og rette de spesifikasjonene som ble gitt der.

Et annet skille mellom programkategorier kan gjøres på størrelse og omfang av programmene. Større systemer kalles gjerne programpakker. Det er ikke noe skarpt skille mellom standardprogram og programpakker, men en programpakke dekker flere funksjoner enn et standardprogram. Utviklingskostnadene kan være meget høye, og ressursforbruket på datamaskinen er vanligvis høyt. Sammenlignet med standardprogrammene har de imidlertid et betydelig bedre tilbud av hjelpemidler, og er vanligvis lette å bruke og bedre dokumentert. En del slike programpakker omfatter både funksjoner for klargjøring av data og for sluttbruk av data. Noen pakker kan være sterke på ett område og svakere på andre.

Noen programpakker med svært mange og ulike typer funksjoner, karakteriseres som informasjonshåndteringssystemer (IH-systemer). Systemene markedsføres som verktøy som skal dekke det behovet en hel organisasjon har for håndtering av informasjon. Systemene håndteres ved hjelp av programmeringsspråk (4. generasjonsspråk) som ligger på høyere nivå enn de programmeringsspråk (3. generasjonsspråk) som vi hittil har brukt. Disse systemene bygger bl.a. på databaseteknologi for håndtering av data.

Vi har ulike klassifiseringer av data. En generell klassifisering skiller mellom mikrodata, makrodata og metadata. Mikrodata er individualdata eller data som refererer seg til den enkelte statistiske enheten. Makrodata er mikrodata som er aggregert ved hjelp av kjennemerker som refererer seg til de statistiske enhetene. Metadata er dokumentasjon av data og kalles også datakatalog i deler av litteraturen.

Data kan være lagret på ulike måter. Lagringsmåten bestemmer bl.a. tilgjengeligheten av data og utformingen av program og språk for å håndtere dataene. En måte å lagre data på er enkle sekvensielle filer av den typen som har vært dominerende i Byrået til nå. Adkomst til den enkelte dataposten skjer ved å lese postene på filen i sekvens inntil vi finner de(n) vi ønsker. Filer av denne typen kan vi lagre på magnetbånd eller platelager. De mest brukte standardprogrammene i Byrået har vært orientert mot data lagret i slike filer. En annen måte å lagre data på er enkeltstående filer med direkte adkomst til den enkelte datapost via en nøkkel uten å måtte lese de øvrige postene i filen. Slike filer ligger på platelager. En tredje måte å lagre data på er i databaser. Denne måten å lagre data gir også direkte adkomst, men det er mulig å finne fram til de ønskede dataene via ulike nøkkelbegrep. Databaser vil i praksis ligge på platelager. Databasene krever andre typer programmeringsteknikker enn de sekvensielle filene.

Standardprogram for å håndtere databaser vil være mer avanserte og kompliserte i sin oppbygging, men vil gjerne ha betydelig mer effektive og brukervennlige språk enn våre gamle standardprogram. De før nevnte 4. generasjonsspråk og IH-systemer faller inn under denne kategorien. Mer om databaser og 4. generasjonsspråk finnes i avsnitt 3.1 om databasesystemer.

## 2.2. Registrering og klargjøring av data

Byrået har programmer for registrering, kontroll, korreksjon, redigering, sammenkopling og omkoding av data.

### 2.2.1 Registrering, kontroll og korreksjon

Byrået har utviklet to programsystemer som er i bruk i dag for kontroll og korreksjon av data.

KOKS er av den eldre typen for satsvis kjøring. Når vi bruker KOKS, registrerer vi dataene i en serie av satsvise kontroll- og korreksjonskjøringer basert på feillister. Vi kan kontrollere at kjennemerkene har gyldige verdier, sammenhengen mellom kjennemerkene og at det ikke finnes dubletter av datapostene. KOKS ble utviklet i 1971-72.

DOKS gir brukeren samtidig dataregistrering, kontroll og korreksjon i interaktiv dialog i én arbeidssekvens via skjermterminal. Det gir en betydelig effektivitetsgevinst i forhold til KOKS. Det er også mulig å kontrollere og korrigere data som er registrert separat. Denne muligheten er særlig brukt ved større datamengder. Programmet styrer dialogen med brukeren via menyer av handlingsalternativ. DOKS er utviklet og installert på en NORD-maskin i 1980-83 og finnes dessuten i en forenklet versjon på IBM. NORD-DOKS kan utføre de samme typene kontroller som KOKS. IBM-DOKS mangler foreløpig dublett-kontroll.

## 2.2.2 Redigering, kopling og omkodinger av data og andre filbehandlingsfunksjoner

### Filbehandling generelt

De filbehandlingsprogrammene som Byrået har i dag, er alle beregnet på sekvensielle datafiler på magnetbånd eller platelager som kan være på detalj- (mikro-) nivå eller aggregerte tabell- (eller matrise-) filer. Brukerne må kunne JCL-språket for den aktuelle maskinen for å bruke dem.

### SFB

Programmet er beregnet på kopling, redigering og utlisting av filer. SFB kan kople, velge ut og redigere data fra inntil to filer. Dataene kan legges ut på en fil på platelager, magnetbånd eller skrives ut på en liste med eller uten hodetekster. Vi styrer SFB ved hjelp av parametriske direktiver.

### PMPP

Programmet kan kople, selektere, kode om og redigere data fra inntil to filer og legge data ut på inntil tre filer. Programmet kan også utføre beregninger, produsere lister og lage enkle tabeller. Direktivenegis i form av forkortede nøkkelord med parametre og kan bl.a. kombineres med Cobolprogram for mer kompliserte oppgaver. Programmet er beregnet på bruk av programmerere og egner seg dårlig som verktøy for brukerne på fagkontorene.

TREKKE

Programmet trekker et tilfeldig antall records fra en fil med inntil 3000 strata. Vi legger de uttrukne dataene på en fil for videre behandling. Bruksområdet er utvalgsundersøkelser etc. Programmet styres ved hjelp av parametriske direktiver. TREKKE behandler sekvensielle filer i satsvise kjøring. Programmet ble utviklet på Honeywell og er konvertert til IBM.

STRIKE

Dette er et standardprogram for å kople sammen data fra flere filer. Programmet er utviklet ved Statistiska Centralbyrån (SCB) i Sverige for kjøring på IBM-maskin. Direktivene til programmet består av faste nøkkelord, parametre og diverse skilletegn. SCB er i ferd med å endre brukerspråket til STRIKE slik at det får den samme syntaksen som TAB68 som også er produsert av SCB og er beskrevet senere. Vi vet ikke hvor langt dette arbeidet er kommet nå.

STRIKE kan lese inntil 30 innfiler og skape et ubegrenset (i praksis) antall utfiler. Funksjonene omfatter konkatenering, samsortering, formatkonvertering, redigering, fordeling av matchede data på en eller flere utfiler, rutiner for selektering etc. Det er mulig å utføre flere av disse funksjonene i den samme aktiviteten.

Omkoding av data

Omkodingsfunksjonen har i tidligere utviklete Byråprogram vært lagt til et separat program. Tendensen i standardprogrampakker som er tilgjengelige på markedet, er å integrere omkodingsfunksjonen med annen filbehandling og tabellering. De programmene som Byrådet har i dag, er alle basert på parametriske direktiver, og brukerne må kunne maskinens JCL-språk.

SOP

Programmet ble utviklet i Byrådet i 1974 på Honeywell. Det kan foreta inntil 120 omkodinger av ulike typer. Kontinuerlige verdiskalaer kan kodes om til diskrete. Avledete kjennemerker kan dannes direkte på grunnlag av kjennemerkeverdier eller etter beregninger. Vi kan også legge resultatet av beregninger (summer, prosent etc.) i utpostene. Det er mulig å kode om alle eller et utvalg av postene i én datafil. SOP brukes ved hjelp av parametriske direktiver.

## Andre standardprogram med omkodingsfunksjoner

I denne gruppen finner vi PMPP, rapportgeneratoren EASYTRIEVE foruten SPSS og SAS som er standardpakken for statistisk analyse og tabellproduksjon etc. De tre sistnevnte er behandlet blant sluttbehandlingsprogrammene.

Noen standardprogram kan lage avledete kjennemerker direkte og kan til en viss grad nyttes uten at det er nødvendig å omkode i en egen aktivitet. Eksempel på dette er TAB68.

## 2.3 Resultatuttak

### 2.3.1 Ulike typer av resultatuttak med standardprogram

Resultatuttaksoppgavene i Byrået omfatter for det første statistikkproduksjon som ender dels direkte i NOS-publikasjoner, dels i råtabeller som enten lagres for å svare på henvendelser fra publikum eller brukes som underlag for manuell framstilling av tabeller eller figurer i NOS. For det annet omfatter resultatuttaket ulike former for statistisk analyse. For det tredje omfatter det ulike typer utlistinger og rapporter.

### 2.3.2 Produksjon av råtabeller og innledende undersøkelser av data

#### STP

Programsystemet ble utviklet i Byrået i 1972 på IBM-maskin og siden konvertert til Honeywell. STP blir styrt ved hjelp av parametriske direktiver. Programmet kan lage tre ulike typer tabeller. Den første typen er i form av en enkel krysstabell beregnet på bruk ved intervjuundersøkelser. De to andre typene er beregnet på de "vanlige" tabellene i ordinær statistikkproduksjon og kan ha inntil ni kjennemerker i forspalten og ett eller flere i hodet. STP kan både aggregere hyppighetstall (opptelling av sumtall (ett eller flere kjennemerker kan summeres). Det er mulig å spesifisere ulike typer av beregninger på de aggregerte dataene. Tabellene kan skrives rett ut på papir eller legges på plate/magnetbånd for videre behandling. Den første tabelltypen kan produseres i et uendelig antall fra en usortert sekvensiell fil. STP kan produsere maksimalt fem tabeller av de to andre tabelltypene i én kjøring fra en sortert sekvensiell fil. For disse tabelltypene kan vi ha tekster i hodene, mens vi i forspaltene får skrevet ut de kjennemerkeverdiene som er representert i datafilen.



Programmerere på Systemkontoret, og etterhvert også øvede brukere på fagkontorene, lager ofte relativt kompliserte tabeller ved å sette sammen satser med påfølgende sorteringer, SFB, SOP og STP. Mellomresultatene fra hver aktivitet i en slik sats ligger på platelager. Slike satser kan bli ganske store og ikke alltid enkle å holde vedlike. Det siste gjør seg særlig gjeldende ved personsifte på et oppdrag da dokumentasjonen ikke alltid er like god. Store satser stiller store krav til JCL-kompetanse. Grunnen til å lage slike satser er fordelene ved at brukerspråkene ikke inneholder noen prosedyre. Derved blir utviklingstiden relativt liten i forhold til å skrive Cobol-program.

#### STP-KOMBI

STP-KOMBI ble utviklet i 1981 på NORD: Dette er en overbygning til STP for å gi brukerne på fagkontorene et hjelpemiddel til enkelt å lage enkeltstående tabeller selv. Brukeren gir direktivene skrittvis i en veiledende dialog med programmet via skjermterminal. Kommandoene er parametriske, men kjennemerkene kan defineres i en variabelkatalog og refereres til ved hjelp av navn. STP-KOMBI er mer brukervennlig enn standard STP da brukeren heller ikke behøver å sortere dataene først. Når brukeren har avsluttet sin tabellspesifikasjon og STP-KOMBI har godkjent den, vil programmet produsere en sats med (i) sortering av datafilen og (i) parameteroppsett for produksjon av "standard" STP. Deretter overføres satsen til Honeywell hvor den blir kjørt. Resultatet kommer ut i form av en vanlig STP "råtabell" eller på magnetbånd/platefil for senere behandling.

Systemet har vist seg å være en god start for nye brukere på fagkontorene. Mer erfarne brukere på fagkontorene har imidlertid større nytte av standard STP og beskrivelsen av denne da den gir mer fleksibilitet.

#### STRAIGHT RUN

Programmet har vært benyttet særlig ved intervjuundersøkelser for å finne frekvensfordelingene til kjennemerkene i undersøkelsen før endelige tabeller spesifiseres og kjøres.

TAB68

TAB68 er et tabellprogram utviklet ved SCB i Sverige for kjøring på IMB-maskin. Direktivene til TAB68 gis i form av faste nøkkelord, parametre, diverse skilletegn og egendefinerte navn. Vi navngir og definerer datafilens kjennemerker og interne variabler. Vi kan lage avledete kjennemerker i TAB68-kjøringer og til en viss grense unngå egen omkodingsaktivitet. Programmet kan produsere hyppighetstall og sumtall. Det er også mulig å beregne prosent, gjennomsnittstall og deltotaler selv om det er tungvint å beregne på aggregerte tall (i motsetning til f.eks. STP3). Vi kan programmere særskilte utvalg, redigeringer, beregninger etc. i egne subrutiner som TAB68 kaller opp for hver record før dataene går inn i tabelleringsrutinen.

Programmet kan operere på en usortert sekvensiell datafil. For enkelte oppgaver må innfilen være sortert. For å bruke basisversjonen av TAB68, må brukerne kunne bruke IBM's system (CMS) for terminalbruk og styrespråket (JCL) for kjøring av jobben under OS.

Det er laget en "overbygging" (TAB68GEN) som forenkler bruken av TAB68 for skjermterminal. TAB68GEN ble utviklet for Folke- og boligtellingsen 1980, men kan også brukes til andre oppdrag som forberedes for bruk av TAB68GEN. Vi har med denne overbyggingen mulighet for å lagre og hente fram standarddefinisjoner og -direktiver. Brukerne styrer TAB68GEN via en interaktiv dialog etter menyprinsippet med løpende veiledning.

Det er også laget et program (CMSTAB) som gjør det mulig å lese datafiler definert under CMS og få tabellene presentert på dataskjerm for inspeksjon før vi skriver dem ut på papir. CMSTAB er beregnet på å styre TAB68-kjøringer for test- og opplæringsformål.

2.3.3 Produksjon av trykkeklare tabellerTEKSTTAB

Dette er et programsystem for å opprette og vedlikeholde tekster til tabeller, kople tall og tekster og skrive ut ferdige tabeller. TEKSTTAB gjør det mulig å skrive ut et komplett sett av ferdig sideredigerte og paginerte tabeller til en hel publikasjon med noter til tabellene. Tallmatrisene til tabellene må lages utenfor systemet med standardprogram (f.eks. STP) eller med spesialprogram som legger matrisene ut på bånd/platefil. Det medfører at systemet kan produsere alle typer av tabeller.

Systemet er satsvis orientert og direktivene til programmet er parametriske. Det er først og fremst anvendbart til publikasjoner som gjentas over flere perioder. Det kan ofte være komplisert å bruke programmet. Det krever derfor høy EDB-kompetanse av brukerne og god dokumentasjon av satsene for vedlikehold.

#### PRINTERTAB

Programmet er spesielt laget for å lage trykkekklare tabeller for Intervjukontoret. Programmet kjøres satsvis på en sekvensiell datafil. Brukerne formulerer både tekstene og direktivene for tabelldefinisjonen i den samme satsen. Programmet kan gi ferdige tabeller fra rådata. Hver kolonne og linje defineres ved hjelp av én variabel. I motsatt fall, dvs. normalt, må vi forberede innfilen med avledete variabler ved hjelp av omkoding. Dette er ofte komplisert prosess og krever høy EDB-kompetanse av brukerne. Vedlikeholdet kan også bli komplisert.

#### NATBLES

Programmet ble utviklet i sin første versjon på Forskningsavdelingen i Byrået i 1970 på IBM og konvertert til Honeywell. Det er spesielt beregnet på å lagre, vedlikeholde, manipulere og skrive ut tidsserier og nasjonalregnskapstabeller. Direktivene til Honeywell-versjonen er parametriske og programmet er beregnet på satsvis kjøring.

En ny versjon av NATBLES er nå utviklet på NORD. Programmet utfører de samme funksjonene, men er annerledes i bruk. Brukeren arbeider via dataskjerm i en interaktiv dialog med programmet. Programmet instruerer brukeren som skrittvis gir direktiver på grunnlag av menyer. Tabellene kan vises på dataskjermen eller skrives til platelager for utlisting.

Dataene til tabellene kan tastes inn via skjermterminalen, men vil som regel hentes inn utenfra i form av ferdige matriser laget av eksterne program.

Foreløpig er NORD-versjonen ikke tatt i full bruk. Den gamle versjonen på Honeywell vil bli brukt parallelt med den nye, en tid framover.

#### TAB68

TAB68 har gode muligheter for teksting av tabeller. Det egner seg derfor godt for produksjon av trykkekklare tabeller for offsettrykking. En del fagkontorpersonale har nevnt at de synes det er tungt å

spesifiserer tekstene til tabellene ved hjelp av TAB68. Stort sett ser det derved ut til at programmene for produksjon av trykkeklare tabeller er tunge å bruke på ett eller flere områder. Det kan være fordi det er vanskelig å tekste tabeller med denne formen for programutstyr og at andre alternativ må søkes.

#### 2.3.4 Statistisk analyse

Byråets arbeid med samfunnsøkonomiske planleggings- og prognosemodeller stiller særlig store krav til programutstyret. Det er da også på dette feltet at bruken av programpakker er mest utbredt. Økonomisk analysegruppe og sosiodemografisk forskningsgruppe nytter blant annet programpakkene DATSY og TROLL. Gruppe for ressursregnskap nytter også TROLL.

DATSY er et generelt verktøy for dataorganisering, modellformulering og matriseberegning. Programsystemet ble utviklet ved Norsk Regnesentral etter oppdrag fra Byrået, og er videreutviklet i Byrået. DATSY som er satsvis orientert brukes til programmering av økonomiske modeller, i første rekke MODIS. Systemet er utformet med sikte på å kunne håndtere store modeller. Endringer i modellene skal kunne foretas enkelt og pålitelig.

TROLL er utviklet ved Massachusetts Institute of Technology (MIT) og ble tatt i bruk i Byrået i 1978. TROLL anvendes til løsning av mellomstore modeller og økonometriske analyser. Systemet er interaktivt og brukervennlig. TROLL er installert på Norges Banks IBM-maskin. Bruken av TROLL har økt sterkt, og stadig flere kontorer tar den i bruk.

TSP - Time Series Processor - er en programpakke for økonometrisk analyse. Den er satsvis orientert, men kjøring fra terminal er mulig. TSP-programmer skrives i et eget kommandospråk som er forholdsvis enkelt å lære. TROLL dekker imidlertid det meste av det TSP kan utføre og bruken av TSP har derfor gått sterkt ned. Byrået tar ikke sikte på å opprettholde systemkompetanse på TSP framover. Pakken vil imidlertid være tilgjengelig for bruk.

#### SPSS

Ved utvalgsundersøkelser nytter vi programpakken SPSS som er utviklet spesielt for analyse av samfunnsvitenskapelige data. SPSS er særlig godt egnet til analyse av tverrsnittsdata. Den tilbyr et sett av analyseteknikker og er utstyrt med filbehandlingsfunksjoner. I analysedelen finnes blant annet program for tabulering, regresjon, statistiske tester, korrelasjoner og faktoranalyse. Med filbehandlingsdelen kan

vi utføre beregninger, kode om gamle og nye variabler (betinget og ubetinget), selektere enheter, telle enheter og veie enheter (for oppblåsing). SPSS og den interaktive versjonen SCSS blir mye brukt i samfunnsvitenskapelige miljøer. Norsk Samfunnsvitenskapelig Datatjeneste (NSD) bruker systemet og vi utveksler i dag data med NSD i SPSS-format.

SPSS er brukervennlig og har meget god dokumentasjon. SPSS er installert på Honeywell-maskinen til SDS. Det finnes også i en versjon for IBM-maskin (bedre oppdatert) og på NORD, men er ikke installert på disse maskinene.

#### DDPP

Programpakken er tilgjengelig på Honeywell og er beregnet på statistisk analyse av data. Den har vært en del benyttet av saksbehandlere som har lært å bruke den på Universitetet i Oslo. Pakken har en del begrensninger i forhold til SPSS og SAS som begge må vurderes som mer brukervennlige og hensiktsmessige for omfattende statistisk analyse. Intervjuavdelingen bruker DDPP til "straight run-kjøring" før endelige tabeller spesifiseres for kjøring på STP1 og andre tabellprogram.

#### GLIM

Dette er et system for å programmere generaliserte lineære modeller og for analyse av data. Systemet er meget godt utbygd for statistisk analyse. Det er imidlertid bare effektivt på små datamengder og har også begrensede muligheter for manipulering av data. GLIM er heller ikke i stand til å håndtere manglende verdier i dataceller. Systemet har også muligheter for å lage grafiske framstillinger via skjermterminal og linjeskriver. Systemet er ikke spesielt godt dokumentert og er installert på Honeywellmaskinen til SDS.

I tillegg til de brukerorienterte programpakken har vi et subrutinebibliotek, NAG (Numerical Algorithm Group), som er svært nyttig ved komplisert numerisk behandling. De vanligste teknikker i forbindelse med matematiske, statistiske og numeriske metoder er dekket i programpakken. NAG må brukes av en programmerer som subrutiner i et hovedprogram i f.eks. FORTRAN. NAG har vært installert på Honeywell-maskinen til SDS siden 1979 og er nå tatt inn på Byråets IBM-maskin i Kongsvinger.

### 2.3.5 Standardprogram for utlisting, slipp- og skjemautfylling

Programmer som dekker utlistingsfunksjonen er SFB og PMPP. De er omtalt under de generelle filbehandlingsprogrammene foran.

#### PPS

Programmet er beregnet for å adressere selvklebende slipper og skjema. Vi kan redigere utskriften på slippene/skjemaene fra alle eller et utvalg enheter på innfilen. PPS leser sekvensielle filer og vi styrer det ved hjelp av parametriske direktiver. Programmet kjøres satsvis på Honeywell-maskinen til SDS.

### 2.3.6 Programpakker for resultatuttak med fil- og databehandlingsfunksjoner

#### EASYTRIVE

Dette er en programpakke for rapportgenerering med funksjoner for filbehandling, muligheter for omkodning etc. Pakken er utviklet av PANSOPHIC i USA fra slutten av 1960 årene og etterhvert utbygd med flere funksjoner i løpet av 1970 årene. Den består av en basisdel og en rekke tilleggsmoduler som vi eventuelt må kjøpe i tillegg. Pakken er beregnet som et verktøy for programmerere, men er ikke verre tilgjengelig enn Byråets standardprogram på tilsvarende oppgaver. Vi kan spesifisere enkle oppgaver med enkle direktiver. Kravene til brukerkompetanse stiger med mer kompliserte oppgaver. Språket er nærmere et vanlig programmeringsspråk enn våre standardprogram, men er begrenset med faste funksjoner.

Rapportgeneratoren i EASYTRIVE kan lage tabeller med tekster i hodene og utlistinger (inklusive utfylling av slipper og skjema).

Filbehandlingsfunksjonene omfatter lesing av ulike typer filer. Oppdatering av ikke-sekvensielle filer krever kjøp av tilleggsmoduler. Ellers kan vi selektere data, koble to og to filer, sortere filer, samsortere flere filer, søke i datafiler og redigere filer.

EASYTRIVE er godt dokumentert, men ikke så brukervennlig som mer moderne systemer. Det er beregnet på satsvis kjøring.

SAS (Statistical Analysis System)

Dette er en programpakke med hovedvekten på statistisk analyse, men har også omfattende muligheter for sluttbehandlingsoppgaver som tallproduksjon, annen rapportgenerering og ulike filbehandlings- og omkodingsfunksjoner. Disse funksjonene er tilgjengelig i en "basispakke", mens noen funksjoner er tilgjengelig i separate moduler som vi eventuelt må kjøpe i tillegg. Den grafiske pakken SAS/GRAPH har Byrået allerede anskaffet. Blant de andre produktene finner vi tidsserieanalyse (SAS/ETS), operasjonsanalyse (SAS/OR) og skjemahåndtering (SAS/FSP). Det er også mulig å håndtere databaser ved hjelp av SAS. Danmarks Statistikk bruker SAS til en tidsseriedatabase. SAS har et godt system for å danne datakataloger (metadata).

Delen for statistisk analyse omfatter krysstabellering, ulike former for varianser og regresjoner, korrelasjons- og faktoranalyse, lineære modeller og matrisebehandling.

SAS har forskjellige rapportgenereringsfunksjoner. Noen er enkle og lette å bruke, andre mer kompliserte og krevende. Tabelleringsfunksjonen har visse begrensninger, men er grovt sett på høyde med TAB68 og EASYTRIEVE.

Filbehandlings- og datahåndteringsfunksjonene omfatter kopiering, redigering, oppdatering, konkatenering, sortering og samsortering av filer.

2.4 Oversikt over programmer på Byråets maskinerHONEYWELL

Klargjøring av data:

- KOKS (kontroll/korreksjon)
- SFB (filbehandling)
- PMPP (filbehandling/omkoding)
- SOP (omkoding)
- TREKKE (trekking av utvalg)

Resultatuttak:

- STP (råtabeller)
- STRAIGHT RUN (envariable fordelinger)
- INTERTAB (trykkeklare tabeller for intervjuundersøkelser)
- TEKSTTAB (trykkeklare tabeller)
- NATBLES (trykkeklare nasjonalregnskapstabeller)

## Resultatuttak: (forts.)

- GLIM (Generaliserte lineære modeller)
- SPSS (statistisk analyse)
- DATSY (økonomiske modeller, MODIS etc.)
  
- TSP (tidsserieanalyse)
- NAG (numerisk analyse)
- PPS (adressering av skjema og slipper)

NORD

## Registrering og klargjøring av data:

- DOKS (registrering, kontroll, korreksjon)

## Resultatuttak:

- STP-KOMPI (råtabeller, satsvis kjøring på data på Honeywell)
- NATBLES (nasjonalregnskapstabeller)

IBM

## Klargjøring av data:

- DOKS
- TREKKE (trekking av utvalg)
- STRIKE (kobling av datafiler)
- SAS (diverse filbehandlingsfunksj.)
- EASYTRIEVE (diverse filbehandlingsfunksj.)

## Resultatuttak:

- TAB 68 (råtabeller og trykkeklare tabeller)
- INTERTAB (trykkeklare tabeller for intervjuundersøkelser)
- SAS (statistisk analyse)
- EASYTRIEVE (tabellproduksjon)
- TROLL (modeller, økonometriske problemer, tidsserieanalyse)
- NAG (numerisk analyse)

2.5 Om bruk av standardprogram på fagkontorene til nå2.5.1 Innledning

Opplysningene om fagkontorenes bruk av standardprogram er basert på uformelle samtaler med saksbehandlere og andre som utfører EDB-arbeid på



5. kontor og i gruppen for EDB-arbeid ved Underavdelingen for intervjuundersøkelser. Utvalget er neppe representativt og gir ikke et fullstendig bilde av forholdene. Samtalene har allikevel gitt nyttig informasjon om bruken av de ulike programmene, hvordan brukerne oppfatter programmenes anvendbarhet, opplæringssituasjon og noe om hva brukerne kan tenke seg av forbedringer.

### 2.5.2 Programutvalg

De Byråutviklede standardprogrammene ble laget av systemerere/programmerere som til stadighet fikk mer eller mindre like oppgaver innen endel statistikkområder. Disse oppgavene ble programmert med en kombinasjon av mer eller mindre like spesialprogram (i Cobol, RPG og noe Assembler) og maskinleverandørens "utility"-programmer for sortering og redigering. Spesifikasjonene for oppgavene var til en viss grad orientert mot hullkortbasert teknologi. Resultatet var KOKS, STP, SOP og SFB som er i bruk i dag. I den senere tid har DOKS kommet til erstatning for KOKS og TAB68 (på IBM) til erstatning for STP. Men fremdeles baserer Byrådet en vesentlig del av statistikkproduksjonen på de gamle programmene mens nyere verktøy blir relativt lite brukt. Et unntak for dette gjelder Forskningsavdelingen, som foruten å bruke det Byråutviklede NATBLES, har tatt i bruk flere programpakker som dels er utviklet for Byrådet (DATSY), dels programpakker som er kjøpt inn fra ulike produsenter av programverktøy.

### 2.5.3 Fagkontorenes bruk av standardprogram

Fra midten av 1970-åra ble svært mange statistikker produsert stort sett med standardprogram. Det var programmerere på Systemkontoret som brukte standardprogrammene som verktøy for å løse oppgavene som var spesifisert på fagkontorene. Unntaket var Intervjukontoret, senere Underavdelingen for intervjuundersøkelser og Forskningsavdelingen. Programmererne på Systemkontoret ble i stadig større grad opptatt med å lage og vedlikeholde satser av direktiver til standardprogram. Denne oppgaven var programmererne stort sett overkvalifisert til. Ut fra det synet at det var ressursløsning å la programmerere utføre denne funksjonen samtidig med at Systemkontoret stadig lå på etterskudd, ble det bestemt at standardprogramarbeide skulle føres over til fagkontorene. Dette skulle frigjøre arbeidskraft ved Systemkontoret til mer kvalifiserte EDB-oppgaver. En del oppgaver er i de siste par årene blitt overført til fagkontorene, men fremdeles består mange programmereres arbeid av å fylle ut direktiver og lage standardprogramsatser.

Hvordan er så arbeidssituasjonen på fagkontorene for de som bruker de Byråutviklede standardprogrammene?

På underavdelingen for intervjuundersøkelser utføres arbeidet i en gruppe som er spesialister på å bruke standardprogram. På de andre kontorene har en del saksbehandlere tatt programmene i bruk selv.

Felles for de to gruppene er at de arbeider med forskjellige standardprogram som har forskjellig syntaks. Dette gjør arbeidet noe tungt. Det er nødvendig å omstille seg til en viss grad fra program til program. Dessuten er det nødvendig å kunne det relativt kompliserte styreprogramspråket (JCL) for den maskinen som blir brukt i tillegg til sorteringsprogrammet og andre hjelpeprogram. Situasjonen for saksbehandlerne er derved at de ikke bare må lage isolerte standardprogram, men må sette dem sammen til store og kompliserte satser ved hjelp av styrespråket til maskinen. I løpet av relativt kort tid kan én saksbehandler få mange "tunge" satser som hun/han må både dokumentere og holde velike. Siden tid er en knapp ressurs, kommer saksbehandleren lett i samme situasjon som EDB-personalet. Rettelser må ofte gjøres raskt, og dokumentasjonen blir ikke alltid rettet. Enkelte satser blir i tidsnøden (kanskje) ikke dokumentert i det hele tatt. Ved personskifte kan det da være en tung jobb å finne ut hva og hvordan. Fagkontoret kan derved lett komme i samme situasjon som når det er personskifte på Systemkontoret.

Et forhold som har medført endel problemer, er at fagkontorene også produserer datafiler. Disse er ofte blitt mangelfullt dokumentert eller ikke dokumentert i det hele tatt (mangler filbeskrivelse). Dette skaper problemer for arkivering av data og gir problemer ved gjenbruk av data.

Vi har altså en arbeidssituasjon hvor vi bruker i og for seg enkle standardprogram, men hvor samlingen av standardprogram i satser blir kompliserte. Disse kompliserte satsene krever stor innsikt i alle verktøyene som vi bruker. Saksbehandlere får etter eget utsagn, for tiden ikke tilstrekkelig opplæring til denne arbeidssituasjonen.

På Forskningsavdelingen har situasjonen vært noe anderledes. De tok tidlig i bruk programvarer beregnet på sluttbrukere. Programmene har dels vært utviklet på Forskningsavdelingen, dels laget for Forskningsavdelingen og dels innkjøpt. Forskernes arbeidssituasjon ved bruk av programmene har nok vært noe lettere enn ved Underavdelingen for intervjuundersøkelser og fagkontorene ellers, som beskrevet foran. Derimot har den gruppen som kjører program for forskerne, hatt problemer på linje med de som er beskrevet for fagkontorene.

### 3. STANDARDPROGRAM I TIDEN FRAMOVER

#### 3.1 Innledning

Etter hvert som vi utvider bruken av standardprogram på fagkontorene til alle statistikkområder, vil de problemene som er nevnt foran, melde seg med økt tyngde. Hvordan kan vi møte denne situasjonen?

For det første kan opplæringen utvides, og Systemkontoret kan bygge ut et såkalt brukerservicesenter for å veilede fagkontorene i utførelsen av EDB-arbeide. For det annet kan vi søke å bygge en overbygging over de eksisterende standardprogrammene med felles syntaks for alle funksjonene, eller ta i bruk nye programpakker hvor alle de nødvendige funksjoner er bygd inn. Dette innebærer et prioriteringsproblem. Fagkontorene har de fleste systemene på Honeywell mens det er besluttet å satse på videreutvikling på IBM-maskinen på Kongsvinger.

Vi har i dag én overbygning over STP (STP-KOMBI). Den er imidlertid bare brukbar til enkeltstående og enkle tabelloppdrag på filer som er klargjort. Det er ikke mulig å bruke den til mer krevende oppgaver, som det er mange av i Byrået. Den har vært nytting for enkelte i opplæringsituasjon. For saksbehandlere som har fått en viss kompetanse, blir det enklere å bruke standard STP som også må brukes når STP-KOMBI ikke strekker til. Dette tilsier at en overbygning må dekke alt det vi i dag utfører med standardprogram-satser. Det må dessuten være enkelt å bruke den. STP-KOMBI er imidlertid bare aktuell på Honeywell-maskinen. Den videre utvikling vil i hovedsak måtte foregå på IBM-maskinen. Den overbygningen som er laget for TAB68 på IBM, kan muligens utvikles videre. SCB utvikler i dag en overbygning (DIRIGENTEN) som også vil dekke filbehandlingsfunksjoner.

Alternativt kan vi også finne fram til én eller flere programpakker som kan utføre disse funksjonene på IBM-maskinen og som har større mulighet enn de programmene vi har i dag. Det er tilgjengelig programverktøy som selv allokterer lagerplass til data mellom aktiviteter etc. og hvor brukeren bare trenger å definere arkivfilene og ett generelt område som systemet selv organiserer bruken av uten noen form for komplisert JCL for brukeren. Slike systemer har f.eks. sorteringsfunksjoner som er svært enkle å bruke ved hjelp av samme type syntaks som de øvrige funksjonene. Sorteringene spesifiseres i "flukt" med de andre funksjonene (f.eks. tabeller) uten mellomliggende JCL.

For tiden skjer det en utvikling av programverktøy for to ulike brukerkategorier. Den ene er saksbehandlere som skal gjøre større eller mindre

deler av EDB-arbeidet selv. Den andre er EDB-personale som skal utføre større og mer krevende oppgaver, bl.a. oppbygging av databaser, som i sin tur kan benyttes av saksbehandlere med verktøy beregnet for dem. For begge gruppene tilbys 4.generasjonsspråk som har kommandoer som utfører større funksjoner som før krevde en beytdelig programmererinnsett med de tradisjonelle programmeringsspråkene COBOL, FORTRAN etc.

Dette kapittel vil behandle endel nye programprodukter som vi nå har oversikt over, og eventuell utvikling av produkter vi har tilgang til nå.

### 3.2 Hjelpemidler for programmerere

#### TOOLPACK

TOOLPACK er et av de nye 4.-generasjonshjelpemidlene. Det er en samling programverktøy som er beregnet på å effektivisere utvikling og vedlikehold av FORTRAN77-programmer. Systemet er ikke maskinavhengig, men det er mulig å spesialutruste det på den maskinen hvor det blir installert.

TOOLPACK er ennå ikke ferdig utviklet, men blir trolig tilgjengelig sommeren 1984. Systemet vil bli distribuert av NAG (som leverer programpakken av samme navn).

### 3.3 Klargjøring av data

#### 3.3.1 Redigering, kontroll og korreksjon av data

##### DOKS og ADDERS

IBM-DOKS er et mye enklere og vel også mer primitivt system enn NORD-DOKS. Det ble laget raskt med relativt lite ressursbruk i forhold til NORD-DOKS. IBM-DOKS var mer eller mindre en nødløsning, og byrået har nå kjøpt dataregistreringssystemet ADDERS fra firmaet Top Data. Foruten dataregistreringsfunksjoner har ADDERS et godt utbygd system for å spesifisere kontroller for gyldige verdier og å rette feil i dataene. Via spesielle utganger fra ADDERS har vi mulighet til selv å legge inn egne program for ytterligere kontroller, som f.eks. sammenhengen mellom ulike kjennemerker (sammenhengskontroller). Det bør også være mulig å legge inn dublett-kontroll via en slik utgang.

GRAN78

Systemet er utviklet for IBM-maskiner ved SCB i Sverige. Det er basert på separat registrering og en satsvis kjøring som produserer en feilliste. Feillisten kan siden brukes ved terminalbasert oppretting og inspeksjon av datafilen.

3.3.2 Redigering, kobling og omkodning av data foruten andre filbehandlings-funksjonerEASYTRIEVE og PRO/grammar

Byrået har basisversjonen av EASYTRIEVE (se foran) og kan eventuelt kjøpe tilleggsmoduler for (i) å lese/skrive et ubegrenset antall filer og (ii) oppdatere VSAM-filer. Systemet er satsvis orientert med de vanligste krav til JCL-bruk for slike systemer. Systemet er gammelt og produsenten har stanset videreutviklingen og i stedet lansert et nytt system, PRO/grammar, til erstatning. Dette inneholder både basis- og tilleggs-funksjonene til EASYTRIEVE, foruten flere andre muligheter. PRO/grammar vil bli videreutviklet av produsenten.

PRO/grammar er også basert på satsvis kjøring, men PRO/grammar er enklere i bruk enn tradisjonelle satsbaserte programmeringsverktøy og standardprogrammer. PRO/grammar kan løse alle sorterings-, filbehandlings- og tabellfunksjoner hvor vi i dag bruker egne og maskinleverandørers standardprogram. Med PRO/grammar kan vi sette sammen et antall slike aktiviteter uten mellomliggende JCL. Vi trenger bare å definere arkivfiler inn og ut foruten ett generelt arbeidsområde.

Vi bør vurdere EASYTRIEVE og PRO/grammar i forhold til de nye 4. generasjonsverktøyene, som også kan brukes til registrering, kontroll, filbehandling og resultatuttak.

### 3.4 Resultatuttak

#### 3.4.1 Produksjon av råtabeller

##### EASYTRIEVE-PRO/grammar

Rapportgenerator delen i disse systemene er på høyde med eller bedre enn STP2-3. Den bør kunne brukes på oppgaver hvor det faller hensiktsmessig, parallelt med eller som alternativ til TAB68.

##### SAS

Vi bør også vurdere i hvilken grad det er formålstjenlig å bruke SAS til vanlige tabellproduksjon. Det vil i så fall kreve kjøp av en tilleggsmodul for tabellering. Denne er inspirert av TPL som er nevnt nedenfor, men er ikke så godt utbygd med funksjoner som TPL. SAS, derimot, har et godt utbygd metadatasystem.

##### STP

Enkelte har etterlyst et effektivt program for å produsere toveis hyppighetstabeller som er like enkelt å bruke som STP1 og som kan brukes på alle typer filer. STP1 var opprinnelig skrevet for en IBM-maskin under DOS og må eventuelt konverteres til OS. Vi bør vurdere om vi skal konvertere det og lage en overbygning til det for bruk via skjerm eller om et annet mer tidsmessig program som har variabelkatalog, kan fylle denne funksjonen.

#### 3.4.2 Trykkeklare tabeller

##### PRINTERTAB

Et eget utviklingsprosjekt er i gang for å tilpasse dette programmet til metadatabasesystemet for den kommunaløkonomiske databasen. Programmet er ofte avhengig av at dataene tilrettelegges i et eget delsystem. Dette systemet kan være et metadatasystem for en database eller andre filbehandlingsprogram. Programmet kan være tungvint i bruk og egner seg ikke så godt for oppgaver som endres over tid og for mer kompliserte engangsoppdrag.

##### TAB68

Det interaktive systemet for å generere direktiver (TAB68GEN) kan vi gjøre allment tilgjengelig. Vi vurderer å bygge ut den generaliserte versjonen eller ta i bruk DIRIGENTEN og koble den til metadatasystemet for Byrådet.

TPL

Dette er et produkt fra Bureau of Labour Statistics i USA, og det er svært utbredt. TPL har meget gode tabelleringsmuligheter og er basert på et språk som kan lage kompliserte sammensatte tabeller med få programutsagn. TPL er ikke godt utbygd når det gjelder metadatasystem.

TEXT82

Dette er et matrise- og tekstbehandlingssystem som SCB planlegger å utvikle. Vi bør følge med i utviklingen av dette systemet og vurdere om det kan være aktuelt for oss.

GAMBLE

Dette er en tabellgenerator som er utviklet ved SCB. Den vil bli utvidet med bedre tekstbehandlingsfunksjoner og håndtering av metadata ved å koble den til metadatabasen under AXIS (om AXIS senere i notatet).

INTERTAB

Systemet er under utvikling i en internasjonal arbeidsgruppe (SCP, Statistical Computing Project). Danmark, Finland, Polen, Øst-Tyskland, Romania, Tsjekkoslovakia og Tyrkia er deltakerland. Spesifikasjoner er utarbeidet, og programmeringen er i gang. Systemet ligner mye på TPL, men har ikke alle de mulighetene som TPL har. Imidlertid vil INTERTAB ha et godt utbygd metadatasystem. Dette kan gjøre det mer enkelt å bruke enn TPL, gi god dokumentasjon og systemene som vi bruker det til, vil bli enkle å vedlikeholde.

Andre muligheter

Flere av de programmene som vi hittil har benyttet til å framstille trykkeklare tabeller, har vært oppfattet som tungvinte i bruk av ulike grunner. En grunn kan være at både TEKSTTAB, PRINTERTAB og TAB68 på ulike måter lager tekstene på en indirekte måte. Vi bør derfor se på helt andre alternative framstillingsmåter og vurdere disse mot eventuelle videreutviklingsperspektiver for de programmene vi har. Et alternativ er å ta tallmatriser fra ett eller flere av de ulike aggregerings-/tabellprogrammene inn i et tekstbehandlingssystem. Vi kan så gjøre trykkeklare tabeller ferdige der. Fordelen med et tekstbehandlingssystem er at det kan oppfattes som en mer direkte måte å framstille tekstene på i forhold til de fleste programmene som "kan" tekste tabeller. En aktuell funksjon som utvikles på mikromaskiner, er tekstbehandling. Vi bør undersøke mulighetene for kommunikasjon mellom IBM-kompatible mikro-

maskiner og hovedmaskin. Dessuten bør vi undersøke de mulighetene de nyere 4. generasjonsspråkene gir.

### 3.4.3 Statistisk analyse, modellkonstruksjon

#### SAS

Bruken av SAS kommer til å bli sterkt utvidet. Den vil etter hvert overta etter SPSS. I tillegg til de funksjonene som SPSS dekker, har SAS også mulighet for mer komplisert behandling av data. Resultater fra tidligere analyser kan danne grunnlaget for nye. Dessuten finnes et matrisemanipuleringsspråk, et modellformuleringsverktøy for både lineære og ikke-lineære modeller og en rekke tidsserie analysemetoder. Overføring av data til og fra flere databasesystemer er også integrert. Noen av disse funksjonene må kjøpes som tilleggsmoduler.

#### GENSTAT

Dette er en programpakke for statistisk analyse og programmering av generaliserte lineære modeller og ikke-lineære modeller. Analysedelen omfatter bl.a. multivariat analyse, klyngeanalyse og tidsserieanalyse. GENSTAT er utstyrt med logikkfunksjon for å programmere prosedyrer. Dette kan bl.a. nyttes til å programmere og lagre makroer som brukeren siden kan nytte i tillegg til et standard makrobibliotek for å lag endelige program.

Vi vet ennå ikke noe om hvordan systemet er i praktisk bruk og om hvor god dokumentasjonen er.

#### ISISMA

Systemet er utviklet ved Universitet i Århus. Det er et verktøy for tidsserieanalyse og formulering av økonomiske modeller. Det vurderes om ISISMA, eventuelt en videreutvikling av dette systemet, kan erstatte DATSY.

#### MODULECO

MODULECO er under utvikling i Frankrike. Det dekker omtrent de samme områder som ISISMA og vurderes også som alternativ til DATSY.



## LISREL

LISREL er et program som utfører faktoranalyse, stianalyse og estimering av simultane likningssystem og økonomiske modeller på stukturform. Det er opprinnelig utviklet ved Universitetet i Uppsala, og er tilgjengelig (versjon 3) på Honeywell-maskinen til SDS.

### 3.5 Program som håndterer databaser

#### 3.5.1 Generelt om databasesystemer og språk for å håndtere databaser

En database er en velorganisert samling av data, som defineres og manipuleres ved hjelp av et databasehåndteringssystem (DBHS).

Et DBHS består av språk for å:

- definere data (deriblant integritets- og sikkerhetsregler)
- operere på data (registrering, endring, fremhenting)

At data er velorganisert betyr at de er strukturert og lagret slik at vi kan dekke flere informasjonsbehov på en enkel og fleksibel måte uten at data trenger å være lagret på mer enn ett sted. I motsetning til tradisjonelle systemer har et databasesystem sentralisert metadatahåndtering. Det vil si at opplysninger om datastruktur, internrepresentasjon osv. lagres i en metadata-base i tilknytning til DBHS, atskilt fra brukerprogrammene. Alle operasjoner på databasen utføres av et felles delsystem. Brukerprogrammene blir dermed enklere og mer uavhengig av hvordan data er lagret.

Språkene for å håndtere data i en database brukes enten innen et vanlig programmeringsspråk (vertsspråk) eller som selvstendige sluttbrukerspråk for spørring, oppdatering, rapportgenerering, statistisk analyse osv.

Det er tre hovedtyper av generelle DBHS (databasespråk):

1. Hierarkiske (DL/I)
2. Nettverkssystemer (CODASYL)
3. Relasjonsdatabasesystemer (SQL)

Språkene (DL/I, CODASYL og SQL) er alle postorienterte datamodellspråk. Data struktureres/grupperes logisk i posttyper (kalles segment i DL/I og relasjonstabell i SQL). Forskjellen mellom språkene ligger i den måten som relasjonene mellom datapostene er representert og i hvordan vi beskriver operasjonene på data.

Hierarkiske DBHS representerer og lagrer data i trestrukturer, dvs. relasjoner av typen over-/underordnet. Et underordnet objekt (post, segment) kan bare ha ett overordnet objekt og søkingen må starte i toppen av trestrukturen. Fysisk lagringsorden og fysiske pekere (adresser) i databasepostene benyttes for å representere de konseptuelle relasjonene.

Et nettverksdatabasesystem har ikke denne begrensningen. En data-post kan være relatert til flere både over- og underordnede poster. Vi kan oppnå raske søk i en nettverksstruktur. Dette skjer ved hjelp av fysiske pekerkjeder i postene.

Relasjonsdatabasespråkene er språk på høyere nivå enn de to foregående fordi de ikke er knyttet til bestemte fysiske datastruktureringsmetoder. En relasjonsdatabase er bygget opp av "flate" filer på tabellform. Postene inneholder dataelementer på samme nivå uten gjentakelsesfelt (normaliserte filer) og har ikke intrarekord-strukturer. Relasjoner mellom relasjonstabeller (filer) er representert ved såkalte "fremmede" nøkler (logiske pekere, referansefelt / koblingsfelt), som inneholder verdien på primærnøkkelen til den relaterte relasjonstabellen).

Relasjonsdatabasespråkene skiller seg fra de språkene (DL/I og CODASYL) som håndterer hierarkiske og nettverksdatabaser, ved at de opererer på hele filer (relasjonstabeller), dvs. på mange poster, i en operasjon. På dette punktet er det en klar analogi til tradisjonell filbehandling med operasjoner for kobling (match, join) av filer og selektering av poster. De fleste DBHS-leverandører tilbyr i dag denne type språk for spørring og rapportgenerering beregnet på sluttbrukere. Språkene brukes mot utsnitt av databasen som for brukerne fortøner seg som en vanlig fil (relasjonstabell). Et slikt databaseutsnitt (konseptuell el. begrepsmessig datamengde) kan inneholde data fra ulike deler av databasen (de data som trengs i den aktuelle applikasjon).

Med de såkalte datamodellspråkene (konseptuelle databasespråk) forsøkes det å gå et skritt videre for å skape en datauavhengig og problemnær beskrivelse av den virkeligheten som skal representeres i en database.

De fleste av disse benytter begreper som enhets- eller objektklasse, relasjon (mellom enheter) og attributt (egenskap, kjennemerke til enhetene). Eksempler på denne type språk er Bo Sundgrens infologiske datamodell (brukes ved SCB) og det norskutviklede SYSDOC-språket. IBM utvikler et tilsvarende språk i den såkalte utvidede relasjonsmodellen (RM/T). Bortsett fra SYSDOC (SYSTEMATOR) foreligger ingen av de nevnte språk ennå som et programprodukt. De brukes for å spesifisere datasystemer på en implementeringsuavhengig måte (kravspesifikasjon). Denne oversettes deretter til et bestemt DBHS, ved hjelp av bestemte konstruksjonsregler.

#### 4. generasjons språk/verktøy

I de senere år har det kommet på markedet systemutviklingsverktøy som ofte betegnes som 4. generasjons språk eller applikasjongsgeneratorer. Maskinkodespråk, ASSEMBLER-språk og høgnivåprogrammeringsspråk som COBOL, FORTRAN, BASIC, skulle da svare til 1., 2. og 3. generasjon). 4. generasjons-språkene er et resultat av en generalisering av grunnfunksjonene i administrativ data-behandling, og består av interaktive verktøy for å lage systemer (konstruere skjermbilder, rapporter, prosedyrer osv.). De frigjør programmerere og sluttbrukere fra å ha inngående kjennskap til operativsystem, styrekommandoer (JCL) o.l. I forhold til tradisjonell systemutvikling (med COBOL) rapporteres meget store besparelser i programmeringskostnader (60-90%) med denne typen verktøy. De muliggjør større brukermedvirkning og systemene kan bygges mer skrittvis, lettere endres etc. På grunn av store krav til maskinressurser er applikasjongsgeneratorene til nå mest blitt benyttet for å lage prototyper.

De mest avanserte av disse produktene betegnes også som informasjons-håndteringssystemer. De fleste av disse systemene er laget med utgangspunkt i et databasehåndteringssystem, men er utvidet med så slagkraftige språk/utviklingsverktøy at behovet for tradisjonell programmering nesten forsvinner. Systemene kan slik sett sies å håndtere informasjon direkte, ikke bare data (representerer informasjon) som må bearbeides av programmere for å kunne presenteres som meningsfylt informasjon for brukerne.

Et informasjonshåndteringssystem (IH-system) inneholder funksjoner for å beskrive og dokumentere data (kjennemerker og enheter), database-funksjoner, skjermbildehåndtering med dialogutforming, rapportgenerator, spørrespråk, prosedyrespråk, tekstbehandling og ofte også grafiske funksjoner og enkel statistisk analyse. Det skal være et integrert sett av verktøy for systemutvikling. Kjernen, eller "hjernen", i et slikt system er metadata-systemet (datakatalog). Her lagres alle beskrivelser av data og funksjoner, og alle komponentene i IH-systemet (databasehåndtering, terminalstyring, rapportgenerator etc.) er koblet til denne.

De fleste produkter av denne type er utviklet for IBM-maskiner. De dyreste er de som er fullt utbygd med eget databasehåndteringssystem og terminalmonitor (AS, RAMIS II, TIS, IDMS, ADABAS, ADR). Andre, som f.eks. UFO, benytter IBM's basissoftware (VSAM, CICS), men har lignende 4. generasjonsspråk (datakatalog, skjermbildefinering, enkelt spørrespråk, rapportgenerator og prosedyrespråk). Også på NORD-maskiner finnes tilsvarende systemer (SYSTEMATOR, UNIQUE, ACCESS).

## Statistiske databaser

Statistiske databaser brukes for å produsere, presentere og analysere statistikk. Det er altså særlig bruken av data som er spesiell for statistiske databaser. I prinsippet kan det tas ut statistikk (sumdata) fra enhver database, og statistiske beregninger utføres i større eller mindre grad i de fleste administrative og tekniske anvendelser. Databaser som kun brukes til statistikkformål (som i Byrået), har dog ganske karakteristiske trekk, både m.h.p. logisk oppbygging, lagringsstruktur og prosesseringsmønster.

Statistikk-baser inneholder ofte svært mange dataelementer (kjennemerker) selv om få typer av enheter (objekter) er representert. De er gjerne også meget omfattende i volum fordi data samles inn for et stort antall enheter (ved f.eks. totaltelling) og/eller for mange tidsperioder.

Mens f.eks. de fleste administrative systemer til enhver tid skal gi et oppdatert "situasjonsbilde" av et fenomen, er sammenlignbarhet og utvikling over tid vel så viktig i statistiske anvendelser. Tidfesting av data, tidsseriehåndtering, har derfor en sentral plass i statistiske databaser.

Statistiske databaser er stabile i den forstand at data som er registrert, kontrollert og korrigert, senere sjelden blir endret. Fjerning vil eventuelt gjelde hele eller større sammenhengende deler av databasen. Tilgang av data skjer også som regel i store satsvise mengder og periodisk. Kontrollrutinene er ofte meget komplekse bl.a. fordi rådataene mange ganger selv er avledede (aggregerte). Registrering og kontroll kan dog skje uavhengig av (utenom) annen databaseaktivitet (ofte i separat system) siden det er først når et vesentlig utvalg av dataene er lastet inn, at det kan tas ut ny og pålitelig statistikk.

Uttak fra en statistisk database (stat. mål som antall, sum, gjennomsnitt) berører typisk mange enheter, men relativt få kjennemerker. I statistisk analyse er det derimot et stort behov for å studere mange ulike kombinasjoner av kjennemerker og mange forskjellige utvalg av enheter. Et statistisk databasespråk må derfor ha fleksible og slagkraftige funksjoner for selektering av enheter, ekstrahering, gruppering og aggregering av kjennemerker. Siden mange funksjoner og aggregering av store datamengder er svært ressurskrevende, er det ofte nødvendig å lagre mange avledete og aggregerte data. Av denne grunn og av sikkerhetmessige årsaker vil endel statistisk databaser kun inneholde aggregerte data (makrodata).

Databaser med individualdata dekker flere statistikkbehov og gir størst fleksibilitet ved uttak, men stiller store krav til sikring/personvern. Konfidensielle data beskyttes generelt i databaser ved passord og autorisasjonsregler. Data som kun brukes til statistikk, kan dessuten anonymiseres. Selv statistiske spørringer med telling/summering etter klassifiserende kjennemerker, kan avsløre individualopplysninger. Statistiske databasesystemer, som håndterer "åpne", konfidensielle data, må derfor ha spesielle sikkerhetsmekanismer for å undertrykke resultater fra spørringer.

Dokumentasjon (metadata) er av stor betydning i statistiske databaser da basene inneholder mange kjennemerker og har lang levetid, men ofte infrekvent bruk. Statistiske data skal også dekke framtidige uforutsette behov, og slik gjenbruk av data krever pålitelig og lett tilgjengelig dokumentasjon.

Generelle prinsipper for konstruksjon av databaser som datamodellering og normalisering er også gyldige for statistiske anvendelser. Det gjelder særlig for databaser som primært opprettes for registrering, kontroll og oppdatering. Videre behandling av mikrodata stiller særlige krav til rask sekvensiell behandling. Selv normaliserte filer vil ofte ha veldig lange poster med mange kjennemerker per enhet. Transponering (vertikal oppsplitting) av postene er derfor mye anvendt i statistisk databehandling. Fullstendig transponering (kolonnevis lagring) er spesielt effektivt ved behandlinger hvor få variabler, men mange enheter aksesseres (benyttes bl.a. i RAPID og SCSS). Lagring av tidsserier bryter ofte prinsippet om normalisering. Grunndatafilene opprettes oftest med nye utgaver for hver tidsperiode. En får dermed flere generasjoner av samme filtype (såkalt horisontal deling). I typiske tidsseriedatabaser med aggregerte data lagres derimot seriene som vektorer/matriser (altså ikke normalisert). Matriseorganiseringen brukes generelt mye for makrodata (bl.a. i TROLL og AXIS) da dette oftest er den raskeste og minst plasskrevende lagringsmetoden for statistiske tabeller.

#### DHBS for statistiske anvendelser

Innen statistisk databehandling benyttes generelle DBHS mest for å dokumentere og håndtere grunndata (mikrodata). Ofte brukes databaseteknikken kun ved registrering, kontroll og oppdatering mens statistikkfunksjonene utføres av standardprogram (tabellgeneratorer) som bearbeider sekvensielle filer produsert fra databasen. Også der hvor hovedformålet med databaseløs-

ninger er å oppnå en økt databeredskap for mikrodata gjennom bedre sam- ordning, organisering og dokumentering, større fleksibilitet i uttak, an- vendes generelle DHBS som f.eks. ADABAS. Det relasjonsorienterte systemet RAPID fra Statistics Canada er mest utbredt blant statistikk-byråer, mens SIR (Scientific Information System) er mye utbredt innen forskningsinstitu- sjoner.

Flere statistiske sentralbyråer (Sverige, Østerrike, Frankrike, Canada) har utviklet rene statistiske DBHS (AXIS, LASD, ARGOC/C, CANSIM). Disse inneholder funksjoner for å definere og lagre statistiske data (even- tuelt aggregere fra mikronivå). De er spesielt beregnet på interaktiv pre- sentasjon og distribusjon av statistikk.

Også statistiske analysepakker (SAS, TROLL etc.) har databasefunk- sjoner. De er særlig egnet for oppbygging av makrodatabaser hvor analyse- behovene er store. Foreløpig er de i mindre grad egnet som rent produksjons- tekniske hjelpemidler. Det er derfor viktig å ha brukervennlige og flek- sible grensesnitt mellom databasesystemer og analysesystemer.

### 3.5.2 Utvikling og bruk av databaser i Byrådet

#### Kommunaløkonomisk databasesystem

Systemet som er utviklet for den kommunaløkonomiske databasen (trinn I), består av et metadatasystem (dokumentasjonssystem), et generelt metadatasystem ekstraseringsprogram, et system for inntasting av ferdig- kontrollerte data og opplegg for produksjon av faste tabeller (PRINTERTAB). En del hjelperutiner for tilrettelegging av filer er laget med EASYTRIEVE. I trinn 2 av prosjektet utvikles integrerte registrering/kontroll- og korreksjons- funksjoner. I trinn 3 (analyse, flere uttaksformer) vil det bl.a. bli utviklet grensesnitt til analysepakker som TROLL og SAS. Ekstraseringspro- grammet vil bli videreutviklet for å oppnå enklere resultatuttak.

Siden innholdet i databasen etterhvert vil bli utvidet med flere variable, er det lagt vekt på å utvikle/bruke generelle programsystemer. Det skilles klart mellom en logisk begrepsmessig beskrivelse av data (variable, enhet, emne) og den fysiske datadefinisjon (fil-, rekord-, felt-struktur). Et uttak (logisk datamengde, databaseutsnitt) spesifiseres på begrepsnivå slik at brukerne ikke trenger å ha kjennskap til de underliggende datastrukturene. Avledede variable kan beskrives i metadatabasen og, hvis de ikke lagres redundant i en datafil, beregnes de ved uttak. Systemet inneholder i dag et standardprogram for ekstrahering av data som er lagret/dokumentert ved hjelp av metadatasystemet til den kom.øk. databasen.

### Databasesystemer som vurderes eller bør vurderes

Byrået bør vurdere ulike systemer som håndterer databaser interaktivt. Minst ett av dem bør trolig også prøves for å vinne erfaring med denne typen verktøy før beslutning om anskaffelse blir gjort. Vi bør trolig satse på et system som nytter et 4. generasjonsspråk for programmering. Systemet bør også tilfredsstillende kravene til et informasjonshåndteringssystem.

### SAS

Blant de systemene vi har i Byrået er SAS godt utbygd for å håndtere data i databaser. SAS er trolig mest anvendbar på databaser med små volumer av mikrodata, på deldatabaser eller på aggregerte databaser basert på vanlige statistikkproduksjonsfiler. Det vil i tilfelle være aktuelt å bygge ut SAS med en tilleggsmodul for skjermhåndtering og kanskje også en modul for tids-serieanalyse (ETS).

Alternative databasehåndteringssystemer er det mange av for IBM-maskiner. Blant statistiske databasehåndteringssystemer finner vi bl.a. RAPID fra Statistics Canada, SIR som er et kommersielt system, foruten AXIS, LASD og ARGOS-C som alle tre er statistiske makrodatabasehåndteringssystemer. Generelle databasehåndteringssystemer som ADABAS, DATACOM/DB, IDMS, TIS, RAMIS II og andre bør også vurderes. Programprodukter fra IBM som vi eventuelt bør se på er, f.eks. (i) AS (Application System) som i en del år tidligere bare var tilgjengelig på IBM's datasentre, men som nå er tilgjengelig for brukere på egen maskin og (ii) Relasjonsdatabasesystemene SQL/DS, DB2, DXT og QMF. Systemkontoret vil legge fram et prosjektforslag for å vurdere et statistikkdatabasesystem for Byrået basert på bruk via et 4.generasjonsspråk.

### AXIS

Systemet er utviklet ved SCB i Sverige. Det er et system for å lagre og hente data fra en statistisk makrodatabase. SCB bruker AXIS til den regionalstatistiske databasen (RSDB) tidsseriedatabasen (TSDB) og delområdestatistikk-databasen (DSDB). Brukerne kommuniserer interaktivt med databasene etter menyprinsippet via skjermterminal. Data som skal legges inn i databasene kan aggregeres med andre systemer som f.eks. TAB68 og leveres på matriseform til AXIS sammen med dokumentasjonen (metadataene). Det er ikke mulig å koble flere matriser når de først er lagt inn i AXIS. Slike koblinger må utføres før dataene lades i databasen.

LASD

Det østerrikske statistiske sentralbyrået har utviklet LASD (Datenbank). Systemet har lignende funksjoner og oppbygging som AXIS, men med bedre muligheter for manipulering av dataene. LASD brukes på tilsvarende oppgaver som AXIS. Det er bare mulig å lade databasen ved satsvis kjøring, mens bruken av dataene kan skje enten satsvis eller interaktivt. Det er ikke mulig å oppdatere makrodatabasen direkte, det må utføres satsvis fra mikrodata.

ARGOS-C

Systemer er utviklet ved det statistiske sentralbyrået i Frankrike (INSEE). Det håndterer aggregerte data som er organisert som matriser/ tabeller. ARGOS-C er meget fleksibelt i håndtering av dataene og det er enkelt å utføre beregninger, aggregering, selektering og kobling av flere tabeller. Dokumentasjonen er lagret sammen med tabellene og brukes ved uttak og manipulering av dataene. ARGOS-C er ikke fullgodt utbygd for å søke etter data på grunnlag av metadataene slik som AXIS og LASD. INSEE bruker tabellutskrifter fra ARGOS-C direkte i publikasjoner og som inndata til fotosettingsprogrammer.

En arbeidsgruppe ved Systemkontoret er i ferd med å sammenlikne og vurdere AXIS, LASD og ARGOS-C i forbindelse med utvikling av en tids-seriedatabase.

3.5.3 Tilknytning til eksterne databaser

I de siste årene har det både i inn- og utland, vært bygd ut en rekke databaser med direkte tilkopling via terminal. Det finnes både bibliografiske databaser og statistiske databaser. Hver database dekker mer eller mindre avgrensede emneområder, men flere databaser er ofte tilgjengelige via ett søkesystem. Pr. 1980 fantes over 650 bibliografiske og 750 faktadatabaser (statistiske databaser er blant de sistnevnte).

De bibliografiske databasene gjør det mulig å søke etter litteratur på grunnlag av forfatter, emneord i tittel etc. Det største søkersystemet er DIALOG, som har vært prøvd med gode resultater ved Byråets bibliotek. I Norge er flere databaser tilgjengelige i POLYDOC-systemet ved Norsk senter for informatikk A/S.

Flere internasjonale organisasjoner tilbyr nå statistikk i databaser i tillegg til, og i økende grad istedenfor, i bokform. Det er tilfelle for FN med underorganisasjoner, Verdensbanken, Det internasjonale pengesfondet, OECD og EF. Alle hovedområder for presentasjon av statistikk er representert. FN utgir en oversikt over internasjonalt tilgjengelige statistiske databaser i Directory of International Statistics.



## 3.6 Andre programsystemer

### 3.6.1 Standardprogrammer for grafisk framstilling

Gode grafiske programpakker har en vesentlig fordel. De gjør det mulig på en rask måte å ta fram alternative grafiske framstillinger av et problem. Dette gir så muligheter for å velge den framstillingsmåten som i praksis viser seg å gi det beste "bilde" av det problemet som vi ønsker å belyse.

Byrået har anskaffet to grafiske programmer; GIMMS og SAS/GRAPH.

#### GIMMS

GIMMS er laget av Geographical Information Systems Ltd. i Skottland. Programmet er spesielt beregnet på å tegne kart, men har også muligheter for å lage stolpe-, sektor- og punktdiagram. Programmet kan tegne diagrammer med inntil fire farger avhengig av utstyret for uttegning. I tillegg til plotter er det mulig å ta ut diagrammer på skjerm og lagre informasjon om diagrammene på platelager. Forskjellige Tektronix-skjermer kan brukes, men ikke de som Byrået har tilknyttet Norges Banks maskin for bruk av TROLL .

#### SAS

SAS har flere muligheter for diagramtegning enn GIMMS. Systemet har også innlagt norgeskart med fylkesgrenser. Foreløpig kan vi ikke bruke SAS/GRAPH i Oslo da vi mangler terminalutstyr.

Nyere informasjonshåndteringssystemer har gode muligheter for grafisk framstilling på skjerm og plotter i flere farger. Programvareprodusenter som f.eks. ISSCO Graphics, tilbyr flere avanserte grafikkpakker. Vi bør trolig vinne erfaring med de programmene vi har nå og se om de tilfredsstillende behovene før vi ser mer på hva andre systemer har å tilby av muligheter.

### 3.6.2 Standardprogrammer via mikromaskiner

I disse dager skjer det en meget sterk utvikling på mikromaskinområdet. Tidligere dreide dette seg stort sett om enkeltstående maskiner med få kommunikasjonsmuligheter. Utviklingstendensene går bl.a. i retning av å knytte sammen flere mikromaskiner i lokale nett med kommunikasjon med sentralmaskin. Ved å fordele arbeidsoppgavene mellom mikromaskinene og sentralmaskinen, kan vi avlaste sentralmaskinen for mange oppgaver. Eksempler på funksjoner og systemer som blir tilgjengelige på mikromaskin, er tekstbehandling, grafikk og IH-systemer som CULLINET, Focus, Ramis III etc. Ved fordeling på flere mikromaskiner kan vi få parallell behandling istedenfor at alle brukerprogram må behandles i serie av sentralmaskinen.

Et problemområde som skulle ligge godt til rett for en slik løsning, er produksjon av publikasjoner. Databehandlingen kan foregå på sentralmaskin mikromaskin eller ved samspill mellom en mikromaskin og sentralmaskin. Tallmatrisene kan så overføres til tekstbehandlingssystemet på en mikromaskin og flettes sammen med den tekstlige delen. Tallmateriale kan også danne grunnlag for grafikk som kan tas inn i en publikasjon.

For best å kunne vurdere mulighetene for framtidig valg, bør vi ha vunnet noe praktisk erfaring i bruk av mikromaskiner med kommunikasjon med hovedmaskin. Systemkontoret vil derfor anskaffe to mikromaskiner som kan kobles til IBM-maskinen og prøve dem på konkrete oppgaver.

#### 4. AVSLUTTENDE MERKNADER

I arbeidet med å finne fram til de best egnede programverktøyene må vi hele tiden ha målsettingen for virksomheten i Byrået for øye; produksjon av statistikk som samfunnet har behov for. Dette er nedfelt i Byråets program for produksjon av statistikk. Programmet kommer til uttrykk i den løpende statistikken og fagkontorenes planlegging av ny statistikk og endringer i den løpende statistikken. Behovene for ny statistikk har sitt utspring i nye krav fra et samfunn i forandring som også krever endringer i den løpende statistikken.

Dette berører imidlertid et problem. Hvordan kommer behovene til uttrykk? Hva bør være grunnlaget for å tilpasse EDB-verktøyene til behovene? Kan vi nøye oss med å se på spesifikasjonene som fagkontorene utarbeider som grunnlag for Systemkontorets EDB-systemer? Trolig er ikke dette tilstrekkelig. De program- og maskinverktøyene som vi til enhver tid har tilgjengelig, setter rammer for de behovene som vi kan tilfredsstille. Dette skjer også i Byrået, kanskje i sterkere grad for saksbehandlere som er svært "nær" EDB-systemene. Saksbehandlere som selv har brukt standardprogram på fagkontorene produserer den statistikken som programutrustningen gir muligheter for. Dessuten har de fremhevet det som en fordel at de vet hvordan programmene virker når de skal lage spesifikasjoner for Systemkontoret. De lager spesifikasjoner som er tilpasset programmene.

Dette betyr at vi ikke bare kan lage standardprogrammer og kjøpe programpakker som dekker de oppgavene som foreligger i spesifikasjonene til Systemkontoret (og i Byråets publikasjoner). Det er to alternativer til denne måten å finne egnete programprodukter. Den første er å sørge

for en kontinuerlig kontakt med oppdragsgivende fagkontorer og miljøer utenfor Byrået. Den andre er å undersøke hvilke alternativer som nye programprodukter kan tilby av statistikkpresentasjon og drøfte disse internt på Systemkontoret og med fagkontorene. Begge disse mulighetene må benyttes. Representanter for fagkontorene og forskningsgruppene får via sine kontakter i miljøer utenfor Byrået informasjon om alternative måter å presentere statistikken på og om verktøy som benyttes. Systemkontoret har også kontakt med EDB-avdelinger i andre lands statistiske byråer og får informasjon om hvilke programprodukter som tas i bruk og disse produktenes muligheter. Et godt samarbeid mellom fagkontorene og Systemkontoret bør derfor gi muligheter for å finne fram til hensiktsmessige programprodukter.

Nye programverktøy som vi anskaffer, vil selvsagt også ha grenser for hva slags statistikkprodukt som vi kan framstille. Men vi bør ikke sette for stramme grenser. Systemene bør gi oss muligheter til å vokse. Det vil gi mulighet til fleksibilitet uten at vi må skifte system ved små behovsendringer. Det kan imidlertid være grunn til å sette minimumsgrenser for systemenes egenskaper. Det gjelder f.eks. brukernes muligheter til selv å løse sine problemer uten spesielt stor JCL-kompetanse og det er enkelt å starte å bruke systemene på enkle problemer.