

SISU

analys

Nr. 1

KONCEPTUELL MODELLERING

- * Grundbegrepp
- * Praktisk tillämpning
- * Forskning

Svenska Institutet för Systemutveckling

ISSN 0282-9924

Copyright SISU

INNEHÅLLSFÖRTECKNING

FÖRORD	3
SAMMANFATTNING	4
1. KONCEPTUELL MODELLERING - EN INTRODUKTION	5
2. KONCEPTUELL MODELLERING I PRAKTIKEN	16
3. FORSKNING OM KONCEPTUELL MODELLERING	39
4. STANDARDISERING	55
5. MER ATT LÄSA	60
6. UTBILDNING	63

FÖRORD

SISU-analys är en tidskrift som utges fyra gånger per år. Varje nummer behandlar ett tema - ett aktuellt problemområde och söker då ge en samlad bild av området ur olika aspekter. Efter en grundläggande introduktion presenteras såväl praktiska erfarenheter som aktuella problemställningar och utvecklingslinjer på den internationella forskningsfronten.

Detta första nummer av SISU-analys är en **kollektiv insats** i ordets rätta bemärkelse. Många personer från olika företag har lämnat värdefulla bidrag.

Lars Axelsson, Kommundata, redogör för sina erfarenheter i avsnitt 2.6. **Janis Bubenko jr, SISU**, har svarat för kapitel 3 och 6. **Christer Dahlgren, Ericsson**, har nedtecknat erfarenheter i avsnitt 2.12. **Kjell Degerstedt, Lantmäteriverket**, har författat avsnitt 2.8. **Eva Lindencrona, SISU**, har svarat för den grundläggande introduktionen i kapitel 1. **Håkan Lövgren, Volvo-Data**, beskriver sin organisations erfarenheter i avsnitt 2.9. **Peter Lindström, Programator**, berättar om sitt företags erfarenheter i avsnitt 2.5. **Björn Nilson, FRI**, har skrivit kapitel 4 om standardiseringssträvanden i databas- och modelleringsområdet. **Bror Noren, Vattenfall**, sammanfattar Vattenfalls syn på modellering i avsnitt 2.13. **Ulf Olofsson, Televerket** har svarat för avsnitt 2.7. **Anders Persson, Data Logic**, berättar om modelleringserfarenheter i avsnitt 2.4. **Eskil Swende och Claas Åkesson, IRM Consult**, bidrar med intressanta erfarenheter i avsnitt 2.3. **Håkan Wall, Saab Scania** redogör för sitt företags erfarenheter i avsnitt 2.10. **Benkt Wangler, SISU**, har svarat för insamling, redigering och sammanfattning av erfarenheterna i kapitel 2. Benkt har dessutom utarbetat "Mer att läsa" kapitlet nr. 5. **Hans Willars, Plandata**, har svarat för erfarenheterna enligt avsnitt 2.2 och **Björn Wåhlberg, Skandia**, har utarbetat underlaget för avsnitt 2.11.

SISU tackar alla medverkande för deras insatser och hoppas att resultatet kan stimulera läsaren till fortsatt inträngning i detta problem- och metodområde.

SAMMANFATTNING

Detta nummer av SISU analys söker först ge en allmän introduktion till vad konceptuell modellering (KM) är och vilket är dess syfte. Därefter behandlar det följande frågor: (1) Vilka erfarenheter har man av praktisk tillämpning av KM?, (2) Vad händer på forskningsfronten?, (3) Vad händer på standardiseringsfronten?, och (4) Var kan man läsa/lära sig mer om KM?

Konceptuell modellering är en formell teknik att beskriva information och data genom att i första hand beskriva den avgränsade verksamhet eller verklighet som en tillämpnings data "handlar om". Detta innebär att man måste klart och entydigt definiera de begrepp, objekt och samband som man vill skall gälla i verksamheten. I en sådan verksamhetsdefinition ingår även definition av restriktioner för vilka samband kan gälla och hur dessa kan eller får förändras med tiden. "Data" får en semantiskt klarare definition om de "hängs upp" på denna verksamhetsbeskrivning. **Konceptuellt schema** är det namn som internationellt används för systembeskrivningar av denna typ.

Av 20 tillfrågade kontaktpersoner i ISVI:s medlemsföretag inkom 12 med korta redogörelser för sina erfarenheter av KM (se kapitel 2). Dessa kan sammanfattas på följande sätt.

KM-tekniken är på stark frammarsch och praktiskt taget samtliga tillfrågade redovisar positiva erfarenheter. KM bidrar till bättre begrepps- och systemdefinitioner, bättre kommunikation med systemens beställare/användare och ger mer exakt underlag för den efterföljande system- och databaskonstruktionen. Man får ett bättre grepp om företagets "informationsresurser". Normalt använder man KM redan i tidiga etapper av systemutvecklingen och då som ett stöd till andra, parallella analysaktiviteter. Det underlättar en "datadriven" systemarkitektur.

Det är lätt att lära sig läsa och förstå KM-beskrivningar men svårare att utforma dessa. Problem uppstår framför allt när man skall "konceptualisera" och "abstrahera" en verksamhet. Vad är en "bra" abstraktion? Tumregler önskas! Behov finns också för mer "hjälpssamma" och avancerade datorstöd. En realistisk modell kan bestå av 10.000-tals definitioner som man skall hålla reda på. Forskningen inriktar sig främst på att utveckla mer "kraftfulla" språk och verktyg för KM så att så mycket "kunskap" om en verksamhet som möjligt kan hanteras. Den forskning som nu bedrivs kommer att ligga till grund för 5:e och senare generationers systemutvecklingsverktyg. Målet, på sikt, är att en högnivå systemspecifikation — ett konceptuellt schema — skall vara tillräcklig för att automatiskt generera ett körbart system. Ett annat viktigt forskningsområde inom KM är utveckling av bättre metodik och arbetsformer för konstruktion av konceptuella schemata. I kapitel 3, som handlar om forskning, ger vi också tips om internationella samfund, konferenser och publikationer som behandlar frågeställningar av relevans för KM.

Kapitel 4 belyser olika standardiseringssträvanden i detta och angränsande områden. Av speciellt intresse här är ISO:s arbete på att söka "standardisera" riktlinjer för "goda" språk för konceptuell modellering. På detta område har standardiseringsarbetet ändrat karaktär och är "pådrivande" i stället för att vara en ex post facto aktivitet. Kapitel 5 och 6, slutligen, ger tips om böcker och kurser för den som vill fördjupa sina kunskaper i konceptuell modellering.

1. KONCEPTUELL MODELLERING - EN INTRODUKTION

1.1. Vad är en konceptuell modell?

Uttrycket 'konceptuell modell' används ofta som synonym till 'data modell'. Vidare förekommer 'objektmodell', 'informationsmodell', 'begreppsmodell' o.s.v. Alla uttrycken är ganska oklara och alla saknar vedertagna betydelser. Ett sätt att beskriva vad en konceptuell modell är för något är att ange vad den används till:

En konceptuell modell används för att beskriva information eller data.

1.1.1. Datorberoende beskrivning.

Att beskriva information eller data är emellertid inget nytt. I traditionella programmeringsspråk t ex finns alltid mekanismer för att beskriva data. Ett exempel är de delar av COBOL-språket som används i dess DATA DIVISION.

```
01 ARTIKELPOST
02 ARTNR PIC...
02 LEVNAMN PIC...
02 A'PRIS PIC...
02 BEST-ANTAL PIC...
```

Figure 1-1: Databeskrivning i COBOL.

I databashanterare finns mekanismer eller hela språk för att beskriva data. Sådana språk kallas vanligen DDL — Data Definition Language — och finns i såväl traditionella databashanterare (t ex DL1/IMS, SCHEMA-DDL/CODASYL) som i relationshanterare.

Programspråkens och databashanterarnas databeskrivningar uttrycks i begrepp som post, segment, set, tabell, fält, term, variabel, nyckel osv. Många av dessa begrepp har att göra med eller har sitt ursprung i datorers sätt att arbeta. En postförekomst t ex är den logiska mängden data som överföres till eller från ett sekundärminne vid en skriv- eller läsoperation i ett program. En 'setförekomst' är en accessväg — en koppling — mellan postförekomster, osv.

```
CREATE TABLE L (LEVNAMN (CHAR (25), NONULL),
ADDRESS (CHAR (35)),
TELEFON (CHAR (7)))

CREATE TABLE A (ARTNR (CHAR (6)), NONULL)
NAMN (CHAR (20)),
APRIS (CHAR (5)),
BESTPKT (CHAR (4)))
```

Figure 1-2: Databeskrivning för relationshanterare.

Så kallade konceptuella modeller skiljer sig från de traditionella databeskrivningarna (programmeringsspråkens och databashanterarnas) framförallt i det att databeskrivningen görs i **datorberoende** begrepp. Exempel på sådana begrepp är

- objekt/entitet
- attribut/egenskap
- förhållande/association
- händelse

Dessa s.k. "modelleringsbegrepp" har ingen direkt motsvarighet i datorers sätt att arbeta. Snarare kan man säga att de är begrepp som filosofer, kunskapsteoretiker och i viss mån lingvister arbetat med sedan långt tillbaka.

1.1.2. Beskrivning av "formaliserad" information.

Vi utgick ifrån att konceptuella modeller används för att beskriva information eller data. När vi talar om information eller data i detta sammanhang avses alltid information eller data i ett informations- eller datasystem. Informationssystem behöver inte vara datoriserade. För konceptuell modellering har detta ingen betydelse, även om det i praktiken vanligen är i samband med datoriserade informationssystem som konceptuell modellering tillämpas.

Dagens informationssystem är antingen formaliserade eller icke-formaliserade. Formaliseringen avser den information som hanteras i systemen, och huruvida denna uttrycks på ett formaliserat eller icke-formaliserat sätt.

Datoriserade informationssystem är oftast — men inte nödvändigtvis — formaliserade. När man inom eller mellan företag använder tele-

konferenssystem (som KOM eller MEMO) för informationsspridning och informationsutbyte kan vi tala om datoriserade, icke-formaliserade informationssystem. Vi kan

FRÅNVARORAPPORT:

NAMN	FR O M	T O M
Lena	85-03-01	85-03-03
Lars	85-04-15	85-04-19
Annika	85-04-22	85-04-26
Lisa	85-05-02	85-05-04
.		
.		

FORMALISERAD
INFORMATION/DATA

FRÅNVARORAPPORT

Lena var sjuk måndag, tisdag och onsdag första veckan i mars. Både Annika och Lars var borta i april. Lars var i Spanien hela vecka 16 och Annika...

ICKE-FORMALISERAD
INFORMATION/DATA

Figure 1-3: Formaliserad resp icke-formaliserad information

finna exempel på formaliserade, icke-datoriserade informationssystem i manuella-, blankett-, formulär- eller kortbaserade system. Vanligast är dock för närvarande att datoriserade informationssystem är formaliserade och att icke-datoriserade informationssystem är icke-formaliserade (utvecklingen går mot en mindre skarp gräns mellan formalisering och icke-formalisering samt mot datoriserade system som hanterar formaliserad, halv-formaliserad..., icke-formaliserad information på ett integrerat sätt).

Konceptuella modeller används idag för att beskriva formaliserad information eller formaliserade data.

1.1.3. Schema eller modell?

När man säger att en konceptuell modell används för att beskriva (formaliserad) informa-

tion eller data menar man därmed också vanligen att det är beskrivningen som är modellen.

En konceptuell modell beskriver information eller data genom att den beskriver den del av en verklighet/verksamhet som informationen eller data "handlar om".

ISO – Internationella Standardiserings Organisationen kallar en sådan beskrivning för ett **konceptuellt schema**. I fri översättning från ISO-rapporten "Concepts and Terminology for the Conceptual Schema and Information Base" finner vi att beskrivning av en verksamhet/verklighet enligt ISO består av två processer (se fig 1-5).

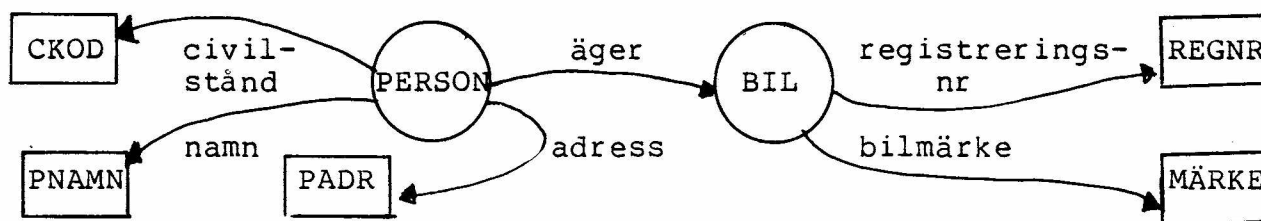


Figure 1-4: En konceptuell modell?

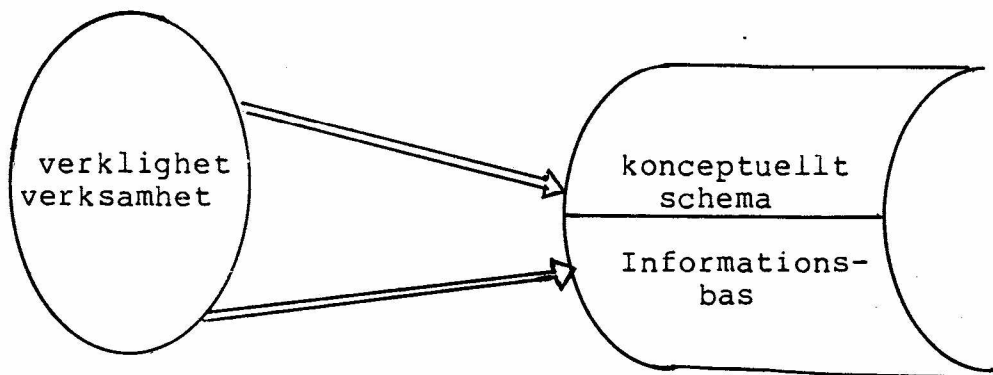


Figure 1-5: Beskrivning av en verksamhet/verklighet.

1. Klassificering, abstraktion, generalisering av verkligheten/verksamheten samt fastställande av regler om och i verkligheten.
2. Registrering av fakta och händelser i verkligheten.

Regler i det konceptuella schemat avbildar regler som existerar i verkligheten och som påverkar skeenden i verkligheten. Därmed kommer reglerna i det konceptuella schemat också att påverka innehållet i informationsbasen och förändringar i informationsbasens innehåll. Något förenklat kan vi säga att man här betraktar en konceptuell modell såsom bestående av schema och informationsbas. Om 'mo-

dellen' omfattar enbart beskrivningen, eller såväl beskrivning som förekomster, har betydelse när man diskuterar och ställer krav på att beskrivningen skall omfatta dynamiska egenskaper. Med ISO's synsätt skall man i det konceptuella schemat beskriva inte bara statiska egenskaper, dvs strukturen i informationsbasen, utan också dynamiska egenskaper, d.v.s. hur informationsbasen kan förändras ge-

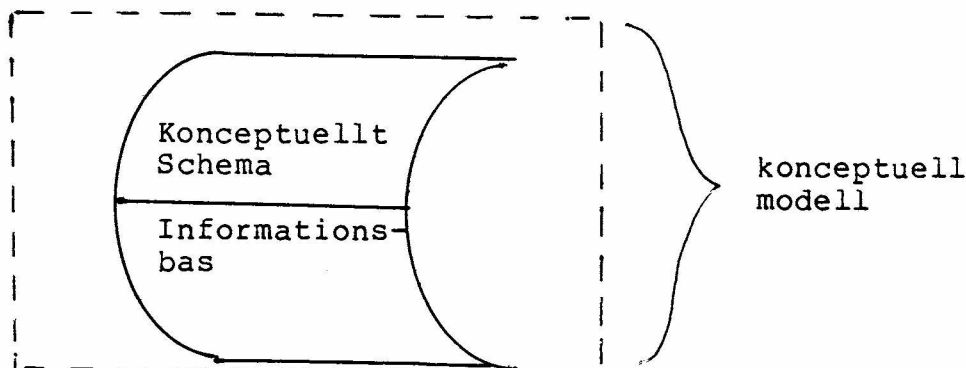


Figure 1-6: Konceptuell modell.

nom tillägg, borttag och ändringar i sitt innehåll. När ett konceptuellt schema skall beskriva inte bara modellens statiska utan även dess dynamiska egenskaper ställs nya krav på det språk i vilket schemat skall uttryckas.

1.1.4. Regler och villkor.

En databeskrivning uttryckt i t.ex. COBOL-språket kan — förutom att den är uttryckt i dataorienterade begrepp — sägas vara 'semantiskt fattig'. Beskrivningen är inte tillräckligt innehållsrik för att en läsare enkelt skall kunna förstå den verklighet/verksamhet som indirekt har avbildats.

I exemplet i figur (1-1) kan vi ana att här avbildas en verksamhet där det förekommer artiklar och leverantörer. Vi kan gissa att artikelnr identifierar artiklar, men hur är det med leverantörsnamn och leverantörer? Kan en artikel levereras av mer än en leverantör? Levereras alla artiklar från någon leverantör? Kan en leverantör leverera många olika artiklar? Varierar a'priset för en artikel med leverantören? Eller avses ett a'pris som är fixt för en artikel oberoende av vem som är leverantör? Är beställningsantalet för en artikel konstant eller varierar det med vem som är leverantör? Finns det leverantörer som inte levererar artiklar? Hur länge räknas man som leverantör efter det man slutat leverera artiklar?

Vid konceptuell modellering eftersträvar man att beskriva 'mer' av verkligheten/verksamheten än vid traditionella databeskrivningar. Man beskriver de företeelser och de samband mellan företeelser som informationen eller data "handlar om". Man beskriver de regler och villkor som gäller för dessa företeelser och samband och som därmed kommer att återspeglas i information om dem. IOS-rapporten anger som en riktlinje för det konceptuella schemat, den s.k. 100%-principen:

Alla relevanta, generella, statiska och dynamiska aspekter, d.v.s. alla lagar, regler och villkor som finns i verkligheten/verksamheten, skall beskrivas i det konceptuella schemat. Ett informationssystem definieras av de regler och villkor som beskrivs i det konceptuella schemat, ej av regler som beskrivs på annat sätt, t ex i applikationsprogram.

Att det konceptuella schemat skall uttrycka enbart konceptuellt relevanta aspekter av data, uttrycks i ISO-rapporten i den s.k. **konceptualiseringsprincipen**:

Ett konceptuellt schema skall enbart omfatta konceptuellt relevanta aspekter — statiska som dynamiska — av verksamheten/verkligheten och därmed inkludera alla aspekter som har att göra med data representation, data organisation, åtkomst, lagring, format, presentation, o.s.v.

När det gäller sambandet mellan de regler och villkor som gäller i verkligheten/verksamheten och de uppgifter som lagras i en informationsbas anger ISO-rapporten som riktlinjer:

- det som är 'omöjligt' i verkligheten/verksamheten bör vara 'otillåtet' i den konceptuella modellen
- det som är 'otillåtet' i verkligheten/verksamheten bör vara 'icke-önskvärt' i den konceptuella modellen

Det är t.ex. i verkligheten omöjligt att vara äldre än sina biologiska föräldrar och då bör detta vara otillåtet i en konceptuell modell. Att vid en viss tidpunkt samtidigt vara gift med två personer är i Sverige otillåtet. Information om ett sådant förhållande bör vara 'icke-önskvärt' i ett informationssystem, d.v.s. resultera i ett felmeddelande eller motsvarande från infor-

mationssystemet.

1.2. Hur ser en konceptuell modell ut?

Det finns idag kanske ett hundratal mer eller mindre olika modelleringsansatser. Med modelleringsansats menas här ett 'språk' och en metodik — ett tillvägagångssätt — att formulera och framställa ett konceptuellt schema. Ett modellerings-språk omfattar vanligen

- en uppsättning modelleringsbegrepp
- en formell notation
- en grafisk notation.

Utgående från modelleringsbegreppen kan man klassificera ansatserna som

- databasorienterade
- objektorienterade
- logikorienterade.

Gränserna mellan dessa klasser är oskarpa. För flera ansatser är det svårt att säga till vilken klass de hör. Om man följer ISO's synsätt och den s.k. konceptualiseringsprincipen samt kravet på att en konceptuell modell skall bygga på datoroberoende begrepp kan man här bortse från de databasorienterade ansatserna. Med databasorienterade ansatser avses de databeskrivningsmöjligheter (DDL) som erbjuds i databashanterarna.

De objektorienterade ansatserna bygger på begrepp såsom:

- objekt/entitet
- attribut/egenskap
- förhållande/relation/association
- händelse

Olika objektorienterade ansatser utnyttjar olika delmängder av dessa begrepp. De s.k. EAR-ansatserna (Entity-Attribute-Relationship) utnyttjar — som namnet anger — begreppen entitet — attribut och relation. Den s.k. NIAM ansatsen utnyttjar begreppen objekt och association osv.

Det finns inga allmänna definitioner av de grundläggande modelleringsbegreppen. Mycket generellt kan man säga att objekt används för att i modellen representera sådant som vi betraktar som företeelser i verkligheten/verksamheten. Vi representerar sådana företeelser med objekt som vi, i ett informationssystem,

vill registrera information om. Exempel på företeelser som naturligt representeras av objekt är personer, bilar, fastigheter, artiklar, or- der, leverantörer, osv.

I en del ansatser (och i ISO arbetet) skiljer man mellan objekt som representerar företeelser (icke-lexikala objekt) och objekt (lexikala objekt) som utgör namn eller värden. De lexikala objekten – namnen, värdena – används för att beskriva eller referera till icke-lexikala objekt.

Attribut representerar den information om objekt, som man vill registrera i informationssys-

tem. Förhållande representerar samband mellan företeelser, d.v.s. förhållanden råder mellan objekt. Förhållanden kan inte existera om inte objekten existerar i modellen. Händelser representerar företeelser som karakteriseras av att de alltid är förknippade med en tidpunkt, den tidpunkt vid vilken de inträffade. Händelser påverkar modellen genom att tillföra eller ta bort objekt, genom att upprätta eller ta bort förhållanden mellan objekt, genom att förändra objektens attribut osv.

De modelleringsbegrepp som används i en modelleringsansats har vanligen en mer precis definition inom respektive ansats.

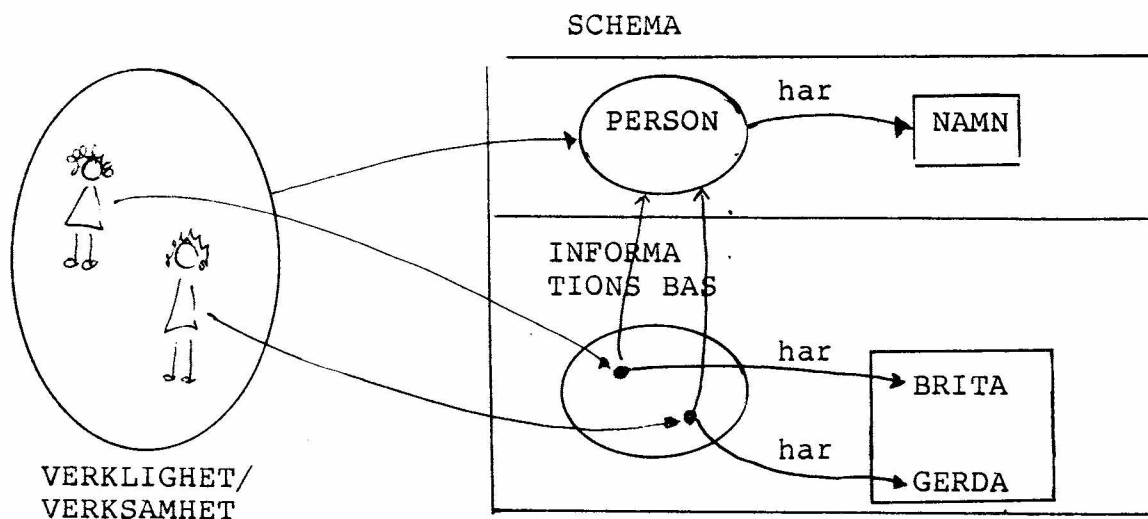


Figure 1-7: Icke-lexikala respektive lexikala objekt.

Dvs ett modelleringsbegrepp kan ha olika definitioner i olika ansatser. Ett exempel på detta är begreppet attribut som inom EAR-ansatserna definieras som en funktion men som inom andra ansatser har andra definitioner och där inte nödvändigtvis motsvarar en funktion.

De logikorienterade ansatserna bygger främst på predikatlogikens grundbegrepp. Huvudsyftet för denna klass av ansatser är att kunna resonera om en verklighet/verksamhet, att kunna visa att mängden av definierade regler är icke-motsägelsefullt och att kunna dra slutsatser på basis av regler och fakta i modellen. Många senare ansatser utgör en kombination av objektorientering och logikorientering.

1.2.1. Grafisk notation

Nästan alla modelleringsansatser har någon form av grafisk notation. Grafer har visat sig

vara speciellt lämpliga vid kommunikation mellan människor. Arbetet med att utforma en konceptuell modell kräver intensiv kommunikation mellan människor. Därför används grafer speciellt vid **utformning** av en konceptuell modell - ett schema.

På sikt kanske grafer även blir en lämplig form för kommunikation mellan människa och dator. Grafisk notation kan då i ökad utsträckning komma att spela en roll vid användning av konceptuella modeller. Olika modelleringsansatser använder olika uppsättningar symboler i sina grafer. Det pågår en diskussion om för- och nackdelar med olika uppsättningar symboler - olika ritsätt. Förutom de som argumenterar för ett speciellt ritsätt förekommer även de som anser att ritsättet i sig - ofta förenklat till "fyrkanter eller ovaler" - är oväsentligt. Forskning kring bilder och symboler blir alltmer aktuell för vårt område och kan kanske på sikt bidra med resultat av betydelse för

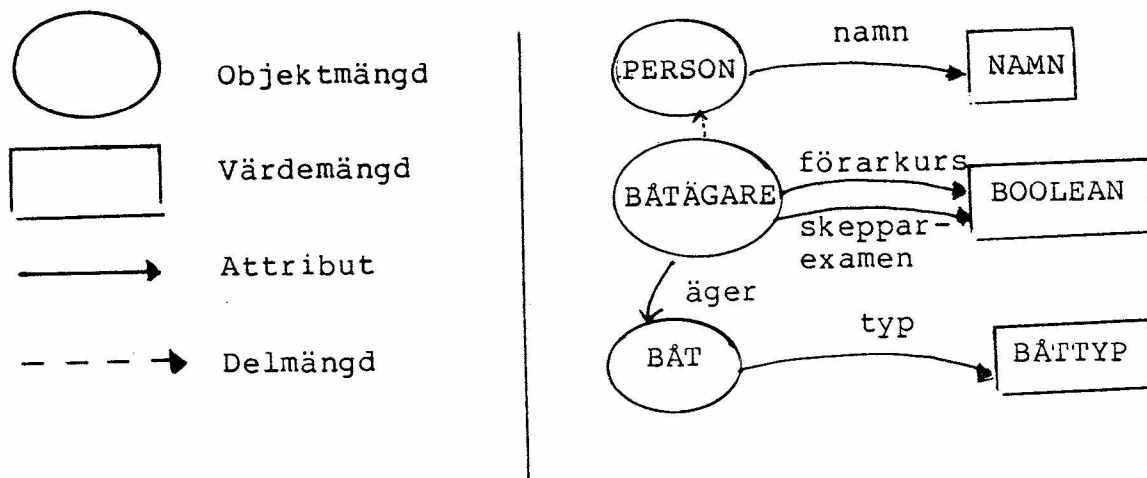


Figure 1-8: Exempel på grafisk notation enligt SIMOL (SISU's Modeling Language)

grafisk notation.

Den grafiska notationen för olika ansatser skiljer sig inte bara i vilka symboler man använder utan även i vad – för vilka modelleringsbegrepp – man har symboler.

I notationen för EAR (se figur 1-7) t ex finns inga grafiska symboler för attribut. Grafen visar enbart objekt och relationer mellan dessa. I t ex SIMOL och NIAM notationerna ritas man ut attributen i graferna. Att en notation ger möjlighet att uttrycka ett modelleringsbegrepp – säg attribut – innebär naturligtvis inte att en graf måste innehålla sådana. Likaså

kan man naturligtvis – i en praktisk situation – i en graf inkludera uppgifter om t ex attribut även om det inte definierats någon speciell symbol för detta. Sunt förnuft avgör hur mycket och vad som i en viss situation är lämpligt att ta med i en viss graf.

I flera ansatser finns grafisk notation för att uttrycka vissa mycket enkla regler och villkor. Vanligast förekommer notation för att uttrycka om en association råder mellan en eller många objekt av en viss typ.

Så snart man vill uttrycka något mer komplexa regler eller villkor – t ex att något förut-

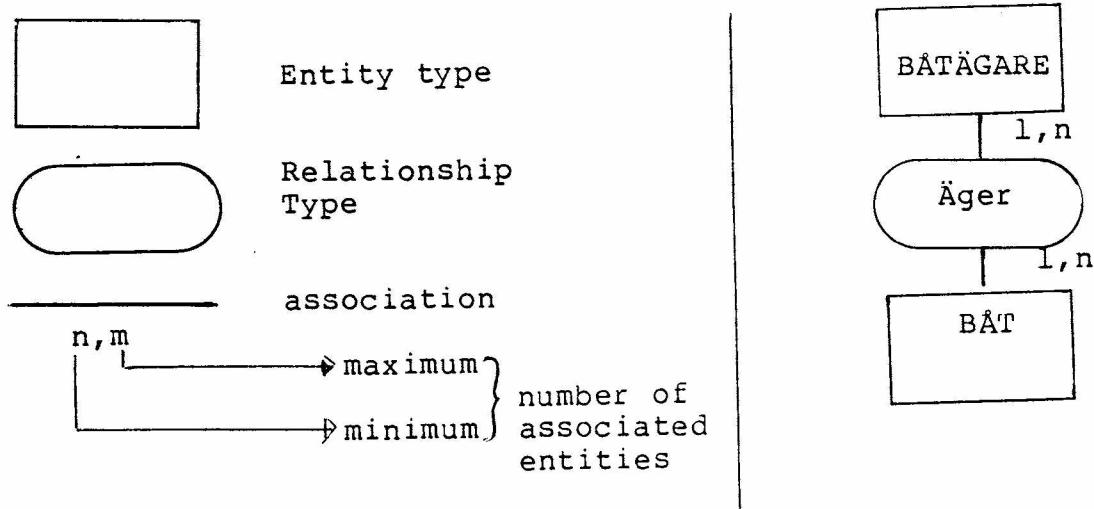


Figure 1-9: Grafisk notation för Entity Relationship ansatsen (enligt ISO).

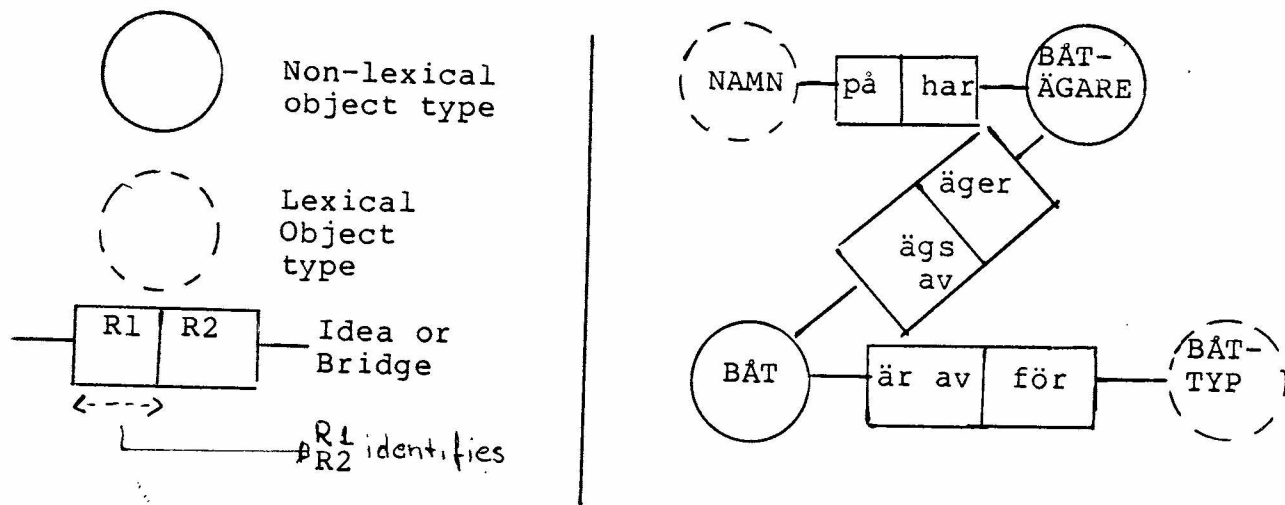


Figure 1-10: Exempel på grafisk notation för NIAM ansatsen (enligt ISO).

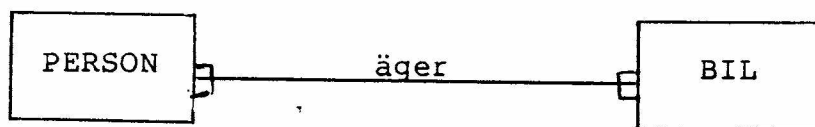


Figure 1-11: Grafisk notation enligt SASMO för att uttrycka ett många till många samband mellan objekt.

sätter något annat, att ett värde kan härledas eller beräknas ur andra värden, att vissa objekt eller värden ömsesidigt utesluter eller implicerar varandra osv – räcker inte den grafiska notationen till. När man på ett entydigt sätt vill uttrycka ett konceptuellt schema krävs en – till den grafiska notationen – kompletterande, formell notation.

1.2.2. Formell notation

I stort sett alla modelleringsansatser omfattar ett formellt (icke grafiskt) språk för att uttrycka ett konceptuellt schema.

De formella språken för konceptuella scheman är vanligen 'kraftfullare' än den grafiska notationen.

CONCEPTUAL SCHEMA	Båtägande
ENTITY-TYPE DESCRIPTION	namn namn förarbevis skepparexamen
RELATIONSHIP-TYPE	äger
DEMENSION COLLECTION	2 båtägare båt
CARDINALITY	båtägande 1,n båt 1,n
DESCRIPTION	

Figure 1-13: Konceptuellt schema uttryckt i ett formellt språk för EAR-ansatsen (enligt ISO).

entitetsmängd LEVERANTÖR
 för mängdens element gäller
 existerar från NYLEVERANTÖR
 till BORTLEVERANTÖR.

är PERSON.
 har attribut levnamn
 avbildar på LEVNAMN
 enligt E:A,A:-
 tilldelas initialt, fixt
 attribut-slut.
 har attribut levererar
 avbildar på ARTIKEL
 enligt F:F, A:A.
 tilldelas initialt, ändras vid NYART
 attribut slut.
 har attribut adress
 avbildar på ADRESS
 enligt E:F, A:-
 tilldelas

Figure 1-12: Del av ett konceptuellt schema uttryckt i SIMOL notation.

Kraftfullheten har framförallt att göra med väl procedurella som deklarativa ansatser vid
 möjligheten att uttrycka regler och villkor. Så- specifikation av regler och villkor förekommer.

```

CONSTRAINT .....
is declared as ...
begin
  for each ...
  condition ...
  holds ...
end
end
  
```

Figure 1-14: Exempel på uppbyggnad av en proceduriell deklARATION av en regel i NIAM språket.

De deklarativa ansatserna representeras främst av predikatlogik-orienterade specifikationer.

Hur 'kraftfulla' de språk som skall användas för att uttrycka ett konceptuellt schema behö-

ver vara diskuterat såväl inom forskningen som i det internationella standardiseringsarbetet. En preliminär kravspecifikation för sådana språk återfinns i ISO-rapporten "Assessment Guidelines for Conceptual Schema

CONSTRAINT $\forall x$ (BÅTÄGARE (x) \longrightarrow $\exists y$ (BÅT (y) & äger (x,y)

Figure 1-15: Exempel på möjlig regelspecifikation i predikatlogik.

Language proposals" ISO/TC97/SC5/WG3
1984-06-09.

schema har en syntax. Syntaxen för beskrivningspråket måste i sin tur uttryckas i något 'språk'.

Ett språk för att uttrycka ett konceptuellt

- (R1) conceptual-schema = "begin"
"CONCEPTUAL-SCHEMA called" schema-name ";"
nolot-declaration {nolot-declaration}
lot-declaration {lot-declaration}
{subtype-declaration}
{idea-declaration}
bridge-declaration {bridge-declaration}
{phrase-declaration}
{constraint-declaration}
end".
- (R2) nolot-declaration = "NOLOT called" (nolot-name | nolot-name-list).
- (R3) lot-declaration = "LOT called" (lotname | lotname-list).
- (R4) subtype declaration nolot-subtype | lot-subtype.
- (R5) nolot-subtype = " NOLOT called"
(nolot-name-1 | nolot-name-1-list)
is subtype-of NOLOT called" nolot-name-2 ";".
- (R6) lot-subtype = "LOT called" (lot-name-1 | lotname-1-list)
is subtype-of LOT called" lot-name-2 ";".

Figure 1-16: Del av formell syntax för beskrivning av NIAM schema deklARATIONSSPRÅK (ur ISO-rapporten).

Det finns alternativa 'språk' för att definiera syntax för ett annat språk.

```
<attribute declaration> ::=  
  attribute-begin <attribute identifier>;  
  range <range identifier>;  
  <mapping clause>;  
  | inverse <attribute identifier>;1  
  <derivation declaration>;  
  | <constraint specification>|  
  | initialize<expression>;1  
  attribute-end | <attribute identifier > ;1  
  
<attribute identifier> ::= <lower case identifier>  
<range identifier> ::= entity type identifier |  
  <L-object type identifier>  
  
<mapping clause> ::= mapping (<mapping specification>)  
<mapping specification> ::= <mapping type>, <mapping property>  
<mapping type> ::= 1:1 | 1:M | M:1 | M:M  
<mapping property> ::= T:T | T:P | P:T | P:P  
  
<derivation declaration> ::=  
  derivation<derivation specification>  
<derivation specification> ::=  
  <set-by clause>!<derivation rule>  
<set-by clause> ::= set-by <event type identifier list>  
<derivation rule> ::= <simple derivation rule> |  
  <structured derivation rule>
```

Figure 1-17: CMOL språkets syntax uttryckt i BNF.

1.3. När används konceptuell modellering?

Konceptuell modellering har traditionellt använts i samband med databassystem. Under de senaste åren har konceptuell modellering i ökande utsträckning kommit att användas i de mycket tidiga faserna i systemutvecklingsprocessen. På en del företag ser man möjligheter att använda konceptuell modellering som ett hjälpmedel för underhåll och förvaltning av system.

Det finns anledning att anta att konceptuell modellering framöver kommer att användas — med olika preciseringsgrad och för olika syften — under hela systemutvecklingsprocessen.

1.3.1. Konceptuell modellering i samband med databassystem

Uttrycken konceptuellt schema, konceptuell modell, konceptuell nivå lanserades av ANSI/SPARC 1975. ANSI/SPARC¹ presenterade då den sk tre- schema-arkitekturen för databassystem. Denna föreslog att det i varje databassystem skulle finnas databeskrivningar på tre nivåer; den interna nivån, den externa nivån samt den konceptuella nivån.

Databeskrivningen på den interna nivån — det interna schemat — beskriver hur data ligger fysiskt lagrat och hur data kan åtkommas. På den externa nivån kan det förekomma ett godtyckligt antal externa schemata. Varje externt schema beskriver en delmängd av data i databasen, strukturerad och uttryckt på ett applikations- (-program) anpassat sätt. Databeskrivningen på den konceptuella nivån — det konceptuella schemat — representerar "the enterprise's view of the data", dvs utgör en för alla användare gemensam beskrivning av data i databasen. Denna beskrivning skall vara fri från interna och externa aspekter, dvs från lagrings-, åtkomst-, representations- och presentationsaspekter.

ANSI/SPARC föreslog inte något speciellt språk för att uttrycka ett konceptuellt schema. Efter det att ANSI/SPARC-förslaget kom har ett stort antal språk — avsedda för att uttrycka databeskrivningar på den konceptuella nivån — presenterats.

Även om den konceptuella nivån vanligen inte har kommit att realiseras enligt ANSI/SPARC's intentioner, så har den dock kommit att bli en

realitet med praktisk förankring. Den konceptuella nivån har kommit att utgöra, inte en del i ett databassystem, men en bas, ett underlag, för databaskonstruktion. De språk som föreslagits får därmed sin användning i samband med databaskonstruktion. För modeller uttryckta i olika språk finns mer eller mindre formaliserade övergångar till olika databashantarens datastrukturer definierade. T.ex. för övergång från en modell uttryckt i EAR-språket till en nätverksstruktur eller för övergång från en modell uttryckt i NIAM-språket till en relationsstruktur osv.

Det hittills dominerande användningsområdet för konceptuell modellering har varit i samband med databaskonstruktion.

1.3.2. Konceptuell modellering tidigt i systemutvecklingsprocessen

Det finns en rad olika, mer eller mindre formella och väldefinierade modeller och beskrivningstekniker som används i de inledande etapperna av en systemutvecklingsprocess. Exempel är MBI, Väggrafteknik, Verksamhetsanalys enligt ISAC, Experimentell Systemutveckling, SYSDOC. Utmärkande för en del av dessa metoder är att analys av information eller data koncentreras till analys av s.k. komponent- och precedensrelationer. I komponentanalysen bestämmer man t.ex. informationsmängders komponenter uttryckta som delinformationsmängder, meddelandetyper och termer. I precedensanalysen analyseras t.ex. vilka informationsmängder som krävs för att producera en viss given informationsmängd. Analys av såväl information som data koncentreras där till analys av inbördes relationer mellan informations- eller dataenheter.

Användningen av och intresse för konceptuell modellering i de mycket tidiga faserna av systemutvecklingsprocessen ökar. Tekniken används då ofta som ett komplement till mer traditionell metodik. Det finns även användare som förespråkar konceptuell modellering som en alternativ metod för verksamhetsanalys. Den resulterande modellen beskriver då i första hand en verksamhet snarare än information eller data i ett informationssystem. Övergången från en generell verksamhetsmodell till en modell av information eller data är inte väldefinierad.

¹ American National Standards Institute/Systems Planning And Requirements Committee

Ytterligare ett syfte med konceptuell modellering i de tidiga faserna, som framhålls allt oftare, är teknikens användbarhet vid kommunikation mellan de personer som kan en verksamhet/verklighet och systemutvecklare. Tekniken understöder en problemorienterad snarare än en dataorienterad kommunikation och kravspecifikation. Vidare, om man utvecklar en detaljerad konceptuell modell tvingas man utreda och precisera de i (en del av) verksamheten centrala begreppen. Det framhålls att man därmed skapar en för (del av) verksamheten gemensamt 'verksamhetsspråk'.

1.3.3. Konceptuell modellering vid underhåll och förvaltning av system

I många företag har man idag stora mängder

data i sina databaser. Innehållet i databaserna är beskrivet i respektive databashanterares DDL. Dvs innehållet är beskrivet primärt avseende de interna – och externa – aspekterna. Databeskrivningarna är därmed i första hand databasadministratörer och programmerares verktyg. Nya typer av tillämpningar, nya användarkategorier, nya systemutvecklingsverktyg, nya kategorier av systemutvecklare m.m. ställer krav på konceptuella beskrivningar av innehållet i existerande databaser för att dessa skall kunna utnyttjas på ett bra sätt.

Intressant i detta sammanhang blir framtidens 'Data Dictionary' eller i ISO terminologi 'Information Resources Dictionary' där en konceptuell modell har en central funktion.

2. KONCEPTUELL MODELLERING I PRAKTIKEN

Datamodellering har under lång tid använts som ett hjälpmedel vid främst databasutformning. I takt med nya forskningsrön, arbetet inom ISO och det ökande intresset för s k Information Resource Management (IRM) har medvetandet om betydelsen av modellering vuxit. Samtidigt har begreppet vidgats, så att man numera hellre talar om konceptuell modellering (KM). Detta för att understryka att syftet är att identifiera, avgränsa och beskriva begrepp som förekommer inom en verksamhet på ett problemorienterat sätt och som alltså inte tar hänsyn till informationsbehandlingsmässiga eller datatekniska aspekter.

2.1 Enkätundersökning – redovisning av erfarenheter

För att i någon mån utröna i vilken utsträckning, på vilket sätt och till vad KM används idag har vi för detta nr av SISU ANALYS bett en del personer med erfarenhet av KM att kort redogöra för detta. Dessförinnan hade vi dock sänt ut en liten enkät i ärendet bland medlemmarna i SISU:s intressentförening. På denna enkät erhöll vi 22 svar, vilka bl a ger vid handen att

- 13 säger sig ha erfarenhet av KM, varav 11 i praktisk tillämpning och 1 på försök.
- Diverse olika metoder, med kända och okända namn, har använts.
- 12 av de 13 har använt KM som hjälpmedel vid verksamhetsanalys, 10 vid databasutformning och 5 vid IRM.
- För det mesta har någon form av datorstöd använts, i några fall av typen data dictionary.

De företag och organisationer som enligt enkäten har erfarenhet av KM samt ytterligare några som vi känt till har vi bett ge en kort beskrivning av sin syn på KM och sina erfarenheter. Resultatet av denna förfrågan redovisas under punkt 2.2 till 2.13 nedan och sammanfattas kort i avsnitt 2.14.

2.2 ABC – metoden för begrepps- och flödesmodeller Bakgrund, syfte, tillämpningar

Hans Willars, PlanData

2.2.1 Allmänt om Begrepps- och Flödes-Modeller (BFM)

All utveckling och datorisering i ett företag bygger på att man kan arbeta utifrån **modeller** av företaget och dess verksamhet. **Relevans** och **”consensus”** är nyckelord för att sådana modeller skall bli fruktbara.

På PlanData anser vi att den modelltyp som kallas ”Begreppsmodell” är mycket relevant för verksamhetsutveckling, och att consensus kring begreppsmodeller är särskilt värdefullt. En begreppsmodell är, som namnet anger, en modell av företagets unika begrepp och idéer. Jämsides med begreppsmodellen, som kan verka ganska abstrakt, arbetar vi med modeller av företagets flöden: de konkreta saks- resp data-flödena mellan arbetsstationer där något utträttas.

PlanDatas metod för praktiskt modellbygge förutsätter att det är ledningen och andra verksamhetssakkunniga som engagerar sig. Metoden innefattar ett gruppdynamiskt arbetssätt, där modellerna växer fram med hjälp av kartong, häftmassa och tejp ”på tapeten” enligt ABC-principen: Alla Bidrar till Consensus.

Resultatet kan ligga till grund för bl a affärsutveckling, ansvarsfördelning, planering/utveckling av ADB samt förbättrad kommunikation och samverkan i företagsärenden.

2.2.2 Några ”grundbultar” i modellarbetet

Enkelhet: Det har visat sig att arbete med modellerna underlättas väsentligt om de grundläggande principerna är generella och begripliga. Idén med BFM skall kunna förklaras på 10 minuter!

Förankring i människan: Det är inte ADB- och databasteknik som motiverar modellarbete, utan människans kognitiva resurser. Begreppen utgör tänkandets byggstenar. Begrepp

och flöden motsvarar väsentligen att — resp hur — kunskaper. Modellerna främjar samverkan människa — teknik.

Växelverkan mellan flöde och begrepp: Det är genom att följa "affärsförlopp" i flödet som man stöter på de begrepp som skall definieras i begreppsstrukturen. Denna ger i sin tur innebörd och stadga åt de saker och data som hanteras i flödet.

Affärsidé först, sedan data: Alla kan förstå att data alltid är data om någonting, och att detta "någonting" måste vara utklarat för att data inte skall hänga i luften. Vi skiljer därför mellan idébärande och databärande begrepp. Det är de idébärande begreppen som "spänner upp" verksamhetsidén och ger struktur och sammanhang åt data. Databegreppen är å andra sidan de som kan åsättas värden av något slag.

Successiv detaljering: Många metoder har lett vilse i detaljeringsträskan! ABC-metodens symboler är valda för att möjliggöra en behovs- och användarstyrd detaljering. Ex: Databegrepp införs endast i den utsträckning som modellarbetets syfte motiverar en sådan detaljering.

Kompetensutnyttjande: Den verksamhetskunskap som skall modelleras finns sällan hos en enda individ. Med ABC-metoden kan flera kompetensprofiler samverka mot en gemensam bild, där var och en har bidragit med sina "pusselbitar". Consensus ger kvaliteten!

Verkligheten är en överenskommelse! Begreppen och deras inbördes struktur är något som måste uppfinnas — de är inte givna en gång för alla, färdiga att bara upptäckas. Begreppen formas i för-logiska, intuitiva processer som stryps i sin linda av översystematisering. ABC-metoden är därför avpassad för ett gradvis införande av logik och systematik. Och människans intuition är ju en resurs som vi skall utnyttja, inte försöka befria oss ifrån!

2.2.3 Tillämpning av BFM: synpunkter och exempel

Kraftfulla utvecklingsmetoder skall sättas in där deras styrka kommer till sin rätt, nämligen på det som är **viktigt** och **svårt**. Det är därför bra om deltagarna i ett modellarbete har förmågan att fokusera på detta, var och en inom sitt kompetensområde. Som regel är sådana

människor både dyra och upptagna, varför deras tid skall utnyttjas effektivt. Därför bör modellarbetet föregås av rejäla förberedelser, där deltagarna intervjuas och "drivande frågeställningar" formuleras. De mentala förberedelserna där alla har börjat tänka i vissa banor kan inte nog poängteras för att ge det gemensamma arbetet en flygande start.

De förberedda frågorna används under modellarbetet för att driva detta framåt. Men det är deltagarna själva som skall formulera svaren som begrepps- och flödes-strukturer! ABC-metoden möjliggör ett "sokratiskt" arbetsätt där alla kan aktiveras både intellektuellt och fysiskt, och det är mycket viktigt för resultatets förankring att inte metodhandledaren "tar över" för att i ren välmening visa "hur det är".

När är det då värdefullt att göra ett modellarbete?

Inledningsvis antyddes breda tillämpningsfält för BFM, och några exempel från vår praktik kan belysa detta.

A. "Informationsförsörjningen kring vår nya produktlinje."

Vid begreppsanalys i ett tillverkande företag hamnar man alltid i produktbegreppet, naturligt nog. I detta fall såg man genast hur den aktuella trenden mot kundorderstyrning tvingade fram en förändring av synen på "produkt". Befintlig terminologi (ex: aggregat, funktionsenhet) blev oanvändbar, och nya ord måste formuleras. Redan innan man kunde börja prata om informationsbehov, så kunde man alltså se hur produktionsutvecklingen krävde en begreppsutveckling som krävde en utveckling av företagsspråket! (Språket är också en viktig resurs i modellarbetet, och inget som till varje pris skall ersättas med en formell symbolik.) Därefter var det relativt enkelt att nå consensus om en produktstruktur som tillfredsställde olika synvinklar: konstruktörens, produktionsens, marknadsföringens och försäljarens.

Med flödets hjälp kunde man sedan plocka fram det begreppsliga sammanhanget kring produkt "från offert till leverans", och en testdatabas byggdes med Dbase3 på en PC enligt principen om kongruens mellan begrepps- och datastruktur. Där kunde olika intressenter göra en första test av sina mest kända info-behov och bedöma informationsergonomin.

B. "Verksgemenskap och decentralisering"

I ett statligt verk kommer man att decentralisera ansvaret för ADB så att det följer verksamhetsansvaret, i första hand länsvis. Det centrala ansvaret för styrning och riksövergripande planering/redovisning medför samtidigt ett behov av verksgemenskap för data i olika avseenden: definition, tillgänglighet, kvalitet, etc. Begreppsmodellen bildar utgångspunkt för arbetet med verkets **dataarkitektur**, som skall ge svar på frågorna:

- Vilka data skall vi ha? (Strukturen)
- Vem skall få göra vad med data? (Ansvar)
- Var skall data finnas? (Lokalisering)

Dataarkitekturen bör inte uttryckas i datatermer, dvs i termer av någon basprogramvara för datahantering. Det skulle ju betyda att man uttrycker målet **efter** det att man har valt medlet för att uppnå målet! En annan tillämpning av begreppsmodellen gäller **delsystemering**, särskilt hur man avgränsar innehållet i olika ADB-system (både centrala och lokala). När detta illustreras i modellen ser man tydligt de begreppsmässiga integrationsområden där det är särskilt viktigt att nå consensus. En nödvändig men "lagom" projektsamordning kan uppnås. Modeller kommer att bli ett levande styrmedel i verksamheten.

C. "Nya grepp i landstingets diabetesvård"

Man hade observerat att den medicinska vetenskapen gjorde goda framsteg inom diabetes, men diabetikerna i stort blev inte bättre i motsvarande grad. Slutsatsen blev att det måste finnas faktorer av betydelse för vårdens resultat som man tog för litet hänsyn till i praktiken. Vilka faktorer? Vad hindrade deras genomslag?

Under ett endags-seminarium byggdes först en begreppsmodell över allt som har betydelse för diabetikernas tillstånd. Arbetsgruppen var bred: distriktsläkare, sjuksköterska, dietist, professor, psykolog och två patienter! Centrala frågor av typen "Vem är patient?" och "Vad är en bra vård?" kunde penetreras och belysas. I termer av begreppen formulerades sedan problemställningar som bearbetades fram till åtgärdsförslag, och även detta gjordes med ABC-metoden.

2.2.4 Vem behöver modeller?

Av praktikfallen ovan bör framgå att modeller kan utföras i flera syften och inom vitt skilda verksamheter. Ett gemensamt drag är att de alltid bidrar till överblick och sammanhang, och detta är ju något som "alla" behöver! Marknaden för modellarbete blir då den växande skara som inser detta behov, och merparten av dessa tycks finnas utanför ADB-avdelningarna.

2.2.5 Hur bli en bra "modelleringskonsult"?

Avslutningsvis några tips till hugade modellarbetsledare. Det går inte att lära sig jobbet genom att bara läsa på. Det finns ett antal "begreppsanalytiska knep" som kräver en hel del träning för att tillämpa. Första villkoret är ju att man överhuvudtaget märker att det är dags att sätta in något som får arbetet att komma ur ett felaktigt spår, en låsning, etc. Sedan gäller det att formulera sig så att deltagarna själva ser ljuset. I botten bör man ha ett genuint intresse för deltagarna och deras "värld", med öppenhet, konsultattityd och servicebeteende som hörnstenar. Det är arbetsamt och ibland svårt att lotsa fram en grupp till den modell som de "känner" för.

Men roligt!

2.3 IRM-strategi och datamodellering

Eskil Swende, Claas Åkesson,
IRM CONSULT

2.3.1 Grundidé

Informationsbehov förändras ständigt

Datastruktur baserad på själva verksamheten är stabil

Slutsats: Systemarkitekturen och systemutvecklingen ska därför baseras på den stabila datastrukturen, dvs vara datadriven.

Vår IRM-strategi och datamodellering baserar sig på en praktisk användning av normaliserings teorin. Bo Sundgren vid SCB insåg tidigt behovet av att lagra data för att möta oplanerade informationsbehov. Han bidrog till en metodutveckling på SAS, som sedan har bred-

dats genom erfarenheter från en rad svenska storföretag inom tillverkningsindustrin, banker, försäkringsbolag och inom kommunal verksamhet.

2.3.2 Marknad

Marknaden för en datadriven utveckling finns främst hos större företag, som har insett att en oplanerad systemstruktur byggd på användarens tillfälliga behov leder till tilltrasslade system. Det är dessa problem som avses med det sk "spagettisyndromet". Marknaden håller nu på att breddas till företag som i förväg vill undvika att hamna i kaos.

2.3.3 Produkter

Huvudprodukten är **datamodellering** som genomförs i 2-dagars datamodellseminarium enligt ett noggrant planerat arbetssätt. Seminariet inleds med utbildning av deltagarna (ca 10 personer som väl känner den aktuella verksamheten) i tekniken att göra datamodell. Idén är att de som kan verksamheten ska stödjas i planeringsarbetet och inte att experten ska försöka lära sig verksamheten. Ledningen av ett seminarium är ett arbete där tidigare erfarenheter ofta är avgörande för att få fram en stabil datamodell av hög kvalitet. Även om själva metoden är mycket enkel är valet av "objekt" med bra nycklar en grannliga uppgift.

Utbildning av instruktörer för datamodellseminarier ute i företagen är en annan produkt. Dessa instruktörer kan sedan samordna dataplaneringen och med expertstöd bygga upp en fungerande data-administration i sina resp företag.

Konstruktion av en datadriven systemstruktur innebär datamodellering av hela eller delar av verksamheten och en verksamhetsbeskrivning där viktiga operativa funktioner identifieras. Dessa analyseras sedan tillsammans i en IRM-matris där det bestäms vilka funktioner som bär ansvaret för resp data. Detta bildar sedan grunden för en **ideal datadriven utvecklingsplan**. Därefter kan det mödosamma arbetet starta att avveckla "spagettistrukturen" och övergå till en stabil datadriven systemstruktur, ett arbete som tar flera år.

Övergången till en **datadriven utvecklingsmodell** är också ett omfattande arbete både vad gäller idéförsäljning och anpassning av resp företags nuvarande utvecklingsmodell. Produkten innebär hjälp med idéförsäljningen och det praktiska genomförandet både vad gäller anpassning av modell och metodstöd till enskilda projekt.

Realisering av datamodellen i den **fysiska databas-designen** är avgörande för införandet av en datadriven ansats. En plan som inte används är ju inte mycket värd. Produkten innebär stöd vid design och revision och till hösten presenteras ett utbildningspaket för hur en datamodell ska omsättas i praktisk verklighet. Det är viktigt att **en** metod kan användas hela vägen så att det slutliga resultatet kan följas upp mot den gjorda planen.

Idéförsäljningen kring **datadriven IRM-strategi** är en viktig produkt. Idén med data-driven ansats måste accepteras av datafolk, användare och ledningen om införandet ska lyckas. Idéförsäljning sker i form av kurser, seminarier och föredrag både generellt och anpassade till resp föredrag.

IRM CONSULT svarar också för distribution inom Skandinavien av ett news-letter som kommer ut varannan månad och som innehåller beskrivning av nya metoder och arbetssätt tillsammans med intervjuer av ledande auktoriteter inom området. Provex kan erhållas från IRM CONSULT.

2.3.4 Användarens roll

Som tidigare nämnts är användarens roll viktig vid framtagande av en datamodell. Metodens enkelhet gör det också lätt för användaren att aktivt delta i planeringen. Vi lägger därför stor vikt vid valet av deltagare till ett seminarium så att kunskap från hela verksamhetsområdet finns representerat. Vi betonar operativ kunskap mer än lednings- och stabskunskap. Metoden inbjuder också efter seminariet till en data-driven utvecklingsmodell där användaren kan ta en aktiv roll i prototyparbetet. Detta innebär en mera påtaglig beskrivning av det framtida systemet än traditionella kravspecar medger.

SAAB-SCANIA INFÖRANDE AV IRM

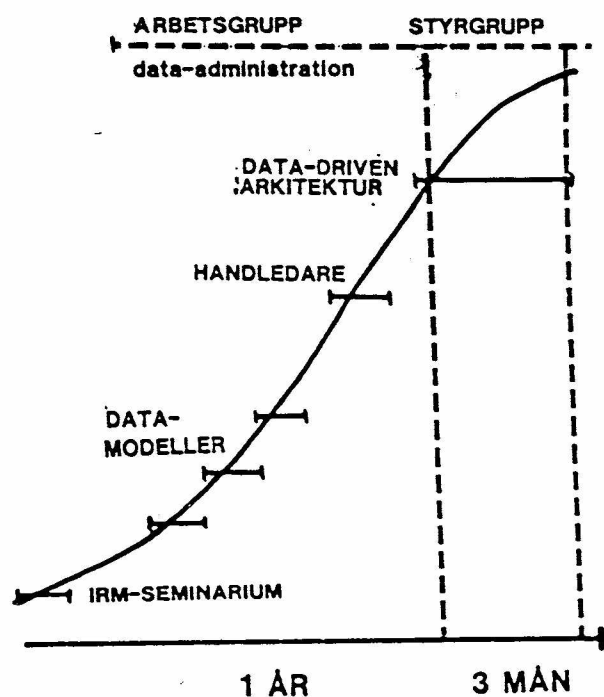


Figure 2-1: Införandet av en datadriven ansats måste ses som en process där man stegvis bygger upp acceptans för idén.

2.3.5 Införandet av datadriven IRM-strategi

Vi rekommenderar alltid att företaget startar med datamodellering av ett mindre projekt; så operativt som möjligt och med goda utsikter att lyckas. Detta för att snabbt erhålla egna positiva erfarenheter av en datadriven ansats. Efter detta byggs erfarenheterna på med ytterligare modellering i några projekt. Därefter är det lämpligt att företaget skaffar egna instruktörer för att driva samordning och starta etablerandet av en data-administration. Först därefter är det lämpligt att starta framtagning av en datadriven systemstruktur. Då finns det egna kunskanden och förankring är gjord hos datafolk, användare och ledningen. I figur 2-1 beskrivs hur införandet har gjorts vid Saab-Scantias flygdivision.

2.4 Konceptuell modellering vid Data Logic

Anders Persson, Data Logic

2.4.1 Inledning

Har vi överblick över vår verksamhet, upplever vi några väsentliga problem, kan vi styra den på ett bra sätt och vilka förändringsåtgärder behöver vi göra för att påverka den?

Ovanstående frågor leder ofta till någon form av AU-arbete. Arbetet syftar till att kartlägga nuläget, fastställa hur den nya situationen skall se ut, beskriva hur förändringsarbetet skall bedrivas samt genomföra förändringsarbetet på ett kontrollerat sätt.

Ordet kartlägga innebär här, att vi studerar ett avgränsat område inom verksamheten ur vissa intressanta aspekter för att sedan beskriva detta ("rita kartor"). Resultatet, dvs våra olika

kartor, syftar endast till att få en ökad kunskap om verksamheten vilken kan förmedlas till flera människor.

Ju mer kunskap om verksamheten vi önskar, desto fler kartor ur olika aspekter och skalor måste vi framställa. Våra kartor måste ju självfallet också samverka och utgå från en gemensam (formaliserad) referensram.

I AU-arbetet framställer vi kartor som beskriver nuläget och kartor som beskriver den nya situationen i verksamheten (verksamheten efter det att förändringsarbetet är genomfört). För att bibehålla och utöka/återanvända kunskapen om verksamheten, behöver vi också arkivera våra kartor på ett enkelt sätt.

2.4.2 Modellering

Konceptuell modellering är en term som har myntats i forskarvärlden och som grovt är det samma som vad vi kallar begreppsmodellering. Emellertid anser vi att ordet konceptuell modellering är svårt att förstå och att ansatsen är för snäv. Vi vill därför bara tala om modellering och modellering ur en given typ.

Modellering innebär framställning av "kartor" över verksamheten. Detta sker inom ett **formaliserat koncept** både vad avser arbetet och den begreppsapparat som används.

Syftet är att erhålla kunskap om verksamheten vilken kan uppfattas av många människor. Häri ingår också i verksamheten överenskomna begrepp och regler.

Modellering kan genomföras på olika detaljeringsnivåer ("världskarta" kontra "Detaljschema") och kan belysa olika aspekter på verksamheten såsom:

- Mål och målprocess.
- Funktioner, processer och händelser.
- Ärenden.
- Formell organisation.
- Informationsflödet.
- Begrepps-/data-modell.

Vilka kartor som skall framställas och vilket detaljeringsdjup dessa skall ha, beror helt på vad vi vill studera.

Modellering kan bedrivas i utredningsarbete på olika nivåer eller i formaliserade projekt.

Data Logic har sedan länge använt olika former av modellering både i systemutvecklingsprojekt och i utredningsarbete på olika nivåer. Modellering är på ett naturligt sätt integrerat i Data Logics systemutvecklingsmodell och Strategisk Informations-Analys (SIA). På senare tid har vi även integrerat våra arbetsformer med synsätt typ IRM/Data-administration.

De beskrivningstekniker vi främst använder är:

- Dataflödesdiagram ("bubbeltekniken"), som används till att beskriva verksamhetens funktioner/aktiviteter/händelser med tillhörande informationsflöde. Denna teknik ingår i ett större koncept kallat "Strukturerad analys".
- Datamodellering både på grov och detaljerad nivå. Beskrivningstekniken bygger på de välkända normalformerna samt på vilka händelser som data utsätts för i verksamheten.

Det har visat sig att modellering blir ett allt viktigare verktyg. Detta gäller inte bara i utvecklingsarbetet utan även för det normala linjearbetet i verksamheten. Allt i den takt som nya hjälpmedel tas fram inom ADB, kommer modellering att spela allt större roller och till en del utgöra "verksamhetens källkod".

För att effektivt kunna använda verktyg av typ relationsdatabaser och fjärde generationens språk, spelar datamodellering en helt avgörande roll. Vi anar också vilka möjligheter ett bra "kartarkiv" typ "Data Dictionary" har att erbjuda.

Om man skall lyckas med modellering, måste detta bedrivas med ett stort engagemang av personal från linjeverksamheten. "Kartorna" är ju helt baserade på kunskapen om verksamheten.

Modellering bedrivs ofta i en arbetsgrupp bestående av både linjepersonal och utredare/systemutvecklare. Om man exempelvis önskar att ta fram en datamodell över en verksamhet är det ofta lämpligt att göra detta i ett intensivt arbetsseminarium under två till tre dagar. Linjepersonalen genomför inte själva modelleringen men förstår dess syfte, resultat och arbetssteg. Dessa spelar istället en aktiv roll i kunskapsöverförandet. Utredaren/system-

utvecklaren behärskar modelleringskoncepten till fullo och baserat på linjepersonalens agerande översätter han detta till en konceptuell modell som alla är överens om.

2.4.3 Data Logic och SISU

Inom SISU:s ram bedrivs ett antal projekt som berör detta område och inom vilket Data Logic aktivt deltar i arbetsgrupper. Dessa är:

- Utveckling av datorstödet RAMATIC.
- Information Dictionary.
- Modellering i systemutvecklingsprocessen.
- Ärendemodellering.

2.5 Objekt- och datamodellering vid Programator

Peter Lindström, Programator

2.5.1 Användningsområden

Tekniken objekt- och datamodellering används av Programatorkonsulter under utvecklingskedan såsom:

- verksamhetsanalys
- informationsanalys
- databasutformning

Den modelleringsansats vi utgår från är Codd's Entity relationship.

A. Verksamhetsanalys (VA)

Ett av verksamhetsanalysens resultat är en kravspecifikation för ett verksamhetsområdes informationsförsörjning. Objektmodeller används i detta sammanhang som ett hjälpmedel och komplement för att:

- beskriva informationssystemens informationsstruktur på en definierad nivå
- peka ut informationsansvar
- identifiera gemensamma objekt
- stämma av och höja kvalitet rörande informationssystemets informations-samband.

Allt detta göra i syfte att:

- skapa konstruktionsunderlag
- skapa bedömningsunderlag (för utvärde-

ring av ex standardpaket)

- skapa underlag för diskussion och realisering av decentraliserade register/databaser.

B. Informationsanalys (IA)

Objektmodeller preciseras under informationsanalysen (IA). Preciseringarnas syfte och inriktning är att beskriva alla objekt, termer och relationer för den information som man beslutar skall lagras. Resultatet blir alltså en komplett beskrivning av vad som skall lagras i systemet.

Om ingen verksamhetsanalys föregått informationsanalysen så omfattar IA de syften med tekniken som beskrivits under VA.

C. Databasutformning

Efter det att alla objekt, termer och relationer preciserats enligt fördefinierad nivå bryts den ned till 4:e normalform. Objektmodellen har då övergått till en datamodell beskriven på en strikt logisk nivå. Denna datamodell kan sedan användas som underlag för transaktionsanalys och logisk databaskonstruktion.

2.5.2 Ett par projekterfarenheter

Tekniken objekt- och datamodellering används sällan fristående från något utvecklingsprojekt. En objektmodell för ett helt företag upplevs ofta som oöverskådlig och resultatet som ganska begränsat. Vår erfarenhet är därför att det är bättre att göra upp objekt- och datamodellen per informationssystem och att beskriva dessa enligt fördefinierade nivåer beroende på användningsområde.

Denna teknik har vi använt oss av vid utveckling av applikationer inom bank- och försäkringsverksamhet, tillverkande industri, vårdsektorn och statliga myndigheter.

Vi har också gjort den erfarenheten under själva utarbetandet av modellen, att det är lättare att göra en objektmodell för ett litet avgränsat område än för ett stort och icke i förväg väl avgränsat d:o.

Dessutom tror vi på ett successivt utarbetande i flera definierade steg snarare än att utarbeta en normaliserad datamodell i ett enda svep. Vi har därför på Programator försökt de-

finiera tre nivåer på objekt- och datamodellen i syfte att ge vägledning. Detta illustreras i figur 2-2.

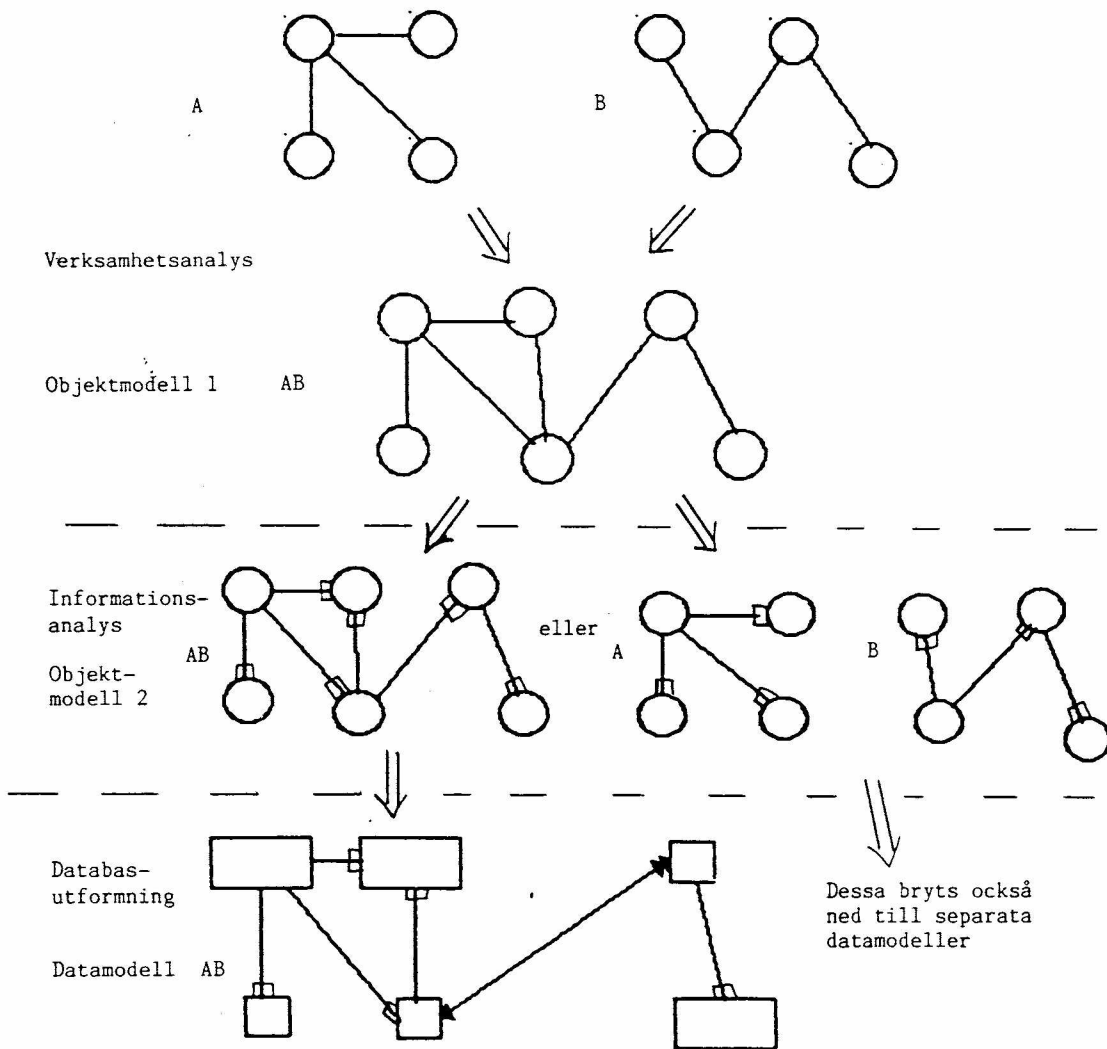


Figure 2-2: Flera lokala objektmodeller integreras, om så är nödvändigt, till en. Denna bryts sedan ned.

2.5.3 Användarmedverkan/utbildning

Kvaliteten på en objektmodell bestäms av dem som gör upp den samt deras kunskap om aktuellt område (informationssystemet) verksamhet. De som har bäst kunskap i detta avseende är användarna av systemet. Därför ställer vi krav på att dessa skall vara med vid utarbetande av en objektmodell.

Ett objektmodellseminarium föregås nästan alltid av en introduktion eller kortare utbildning på någon timme där syfte, teknik och synsätt förklaras och illustreras. Målgruppen är främst de användare som skall delta.

2.5.4 Datorstöd vid objekt- och datamodellering

Datorstödet DAsAK, vilket körs på bl a IBM PC, ger stöd vid:

- objektmodellering
- normalisering
- sammanställning av datamodell
- transaktionsanalys.

Objekten, deras relationer och termer, beskrivs. Objekten normaliseras och sammanställs till en datamodell, där även volymer anges. Transaktionerna beskrivs med avseende

på vilka logiska poster dessa använder. Transaktionerna analyseras. Här kan man nu se vilka transaktioner som är "tungta". Möjlighet finns att ändra sökvägar/ingångar för att testa alternativa lösningssätt.

2.5.5 Upplevda problem

De problem vi upplevt i detta sammanhang är bl a att:

- modellspråket är begränsat och därmed användningen av tekniken
- nivå- och definitionsproblem ex. Vad är ett objekt, vilka kriterier skall ett objekt uppfylla, osv.

2.6 Det enkla är kraftfullt

Lars Axelsson, Kommundata

I ett och ett halvt år har vi arbetat med datamodellering inom Kommundata. Ett tiotal modeller har tagits fram. Intresset för den enkla, mer "primitiva" datamodellering som vi tillämpar, är stort. Metoden är användbar men långtifrån problemfri.

Modellerna har avbildat olika verksamheter: Gatu- och parkförvaltning, fastighetsrelaterad information, kommunal planering, kurs- och konferensadministration, kommunal räddningstjänst m m. Ett par av modellerna har syftat till att skapa en översiktlig kartbild av samspelande verksamheter. Huvudinriktningen är dock, att utifrån väl avgränsade verksamheter, ta fram underlag för databasutformning.

2.6.1 "Gymnastik" — ett syfte

Endast ett par av modellerna har fortsatt i projekt med databasutformning etc. I ett par fall har syftet varit att stimulera till diskussion om en gemensam begreppsapparat och beskrivning av verksamheten. Den kommunala planeringsmodellen har bl a använts som en gemensam referensram för diskussioner mellan olika kommunala och statliga intressenter (REKO 84:5, SCB).

I övrigt är det ett viktigt syfte i ett inledningskede, att över huvud taget praktisera datamodellering. Det finns gott om knepiga modelleringssituationer. Regelbunden "gymnastik" är nödvändig.

2.6.2 Två beskrivningssätt

Den modelleringsansats vi använder är SASMO-metoden (Swende, Åkesson). Metoden bygger på arbetsseminarier med verksamhetskunniga. Deltagarna får först en halv dags utbildning i beskrivningstekniken.

Två beskrivningssätt används omväxlande i arbetet med att gruppera data i verksamheten, grafer med objekt och relationer samt tabeller. Kombinationen av "rena" termfria grafer och exemplifierade tabeller är nyttigt. Tabellerna fyller en pedagogisk uppgift och är särskilt bra för att förtydliga svåra objekt och relationer (händelsobjekt och flerställiga relationer skapar t ex ofta problem).

Modelleringen är lämplig i ett inledningskede av utvecklingsarbetet. Metodens styrka är att den möjliggör en effektiv och korrekt kommunikation mellan verksamhets- och systemexperter. De verksamhetskunniga blir i själva verket huvudaktörer i modelleringsarbetet. Metoden skapar en engagerad dialog med verksamhetsexperterna.

2.6.3 Ny utvecklingsmodell

Tillsammans med andra användarinriktade metoder som rutinskisser och prototyping, vilka också främjar samspel och kreativitet mellan verksamhets- och systemexperter, bör metoden bidra till att vi kan vända blad och lämna de tunga kravspecifikationsmodellerna. Med en förbättrad kommunikation mellan användare och ADB-specialister kan vi få mer snurr på systemutvecklingen.

I en ny utvecklingsmodell bör datamodellering ingå i en "startsträcka" med beskrivningar av projektidén, verksamheten och den aktuella datastrukturen. I Utvecklingshandboken SVEA beskriver vi hur olika metoder kan samverka i en ny utvecklingsmodell (Kommundata, maj 1985, 82 s).

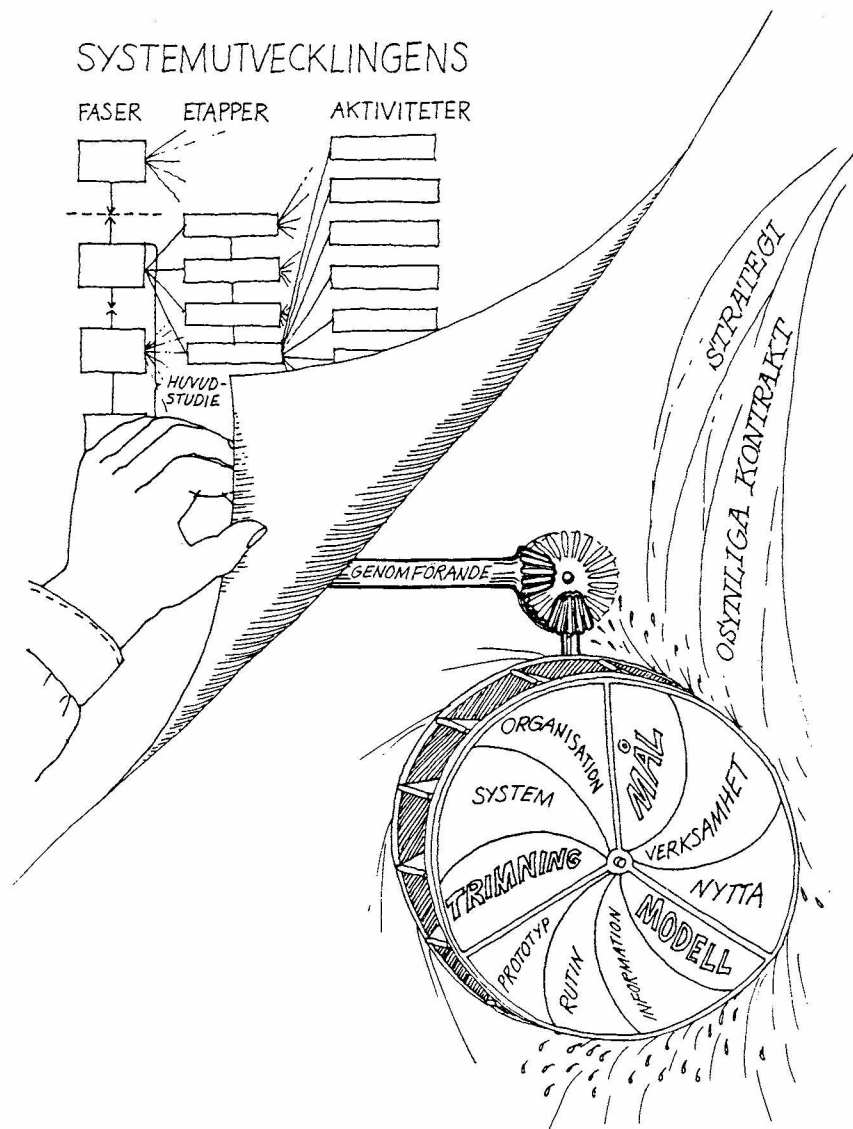


Figure 2-3: Vi sätter snurr på utvecklingen

2.6.4. Problem finns

Modellering enligt den metod vi använder är i vissa avseenden känslig och problematisk.

- Om seminariedeltagarna inte är samlade inom ett väl avgränsat kompetensområde är det svårt att uppnå auktoritativa ställningstaganden.
- Arbetsformerna är krävande. För att få med dem som är mitt uppe "i smeten", dvs som kan verksamheten, är det nödvändigt att arbeta koncentrerat och intensivt under ett par dagar.
- Det är lätt för användare att halka in på beskrivningar av informationsbehov. Att i seminarieform gruppera data och att beskriva beroenden mellan olika data är

svårare och kräver handledare som kan analysera, generalisera, avgränsa etc och som kan förhandla och agera på scenen med en användargrupp.

- Beskrivningstekniken medger bara beskrivningar av "stillbilder" med vissa regler och villkor.

Det positiva överväger dock. Seminarieformerna ger en värdefull gruppdynamik. Begrepp i verksamheten nagelfars effektivt och normaliserade tabeller kan tas fram utan alltför mycket teoretiserande.

2.6.5 "Kraftfullhet"

I en del modellerings-ansatser betonas "kraftfullheten" som ligger i att beskrivningstekni-

ken klarar diverse antaganden, regler och villkor. Betoningen av komplexiteten har tyvärr ofta skymt en annan "kraftfullhet", som gör det möjligt för användare och ADB-specialister att mötas i en klagörande dialog.

Det är svårt att ha många bollar i luften samtidigt. I ett tidigt skede i systemutvecklingen har den förenklade lagringsinriktade datamodelleringen en viktig uppgift. Genom SISU-samarbetet kan vi förhoppningsvis förbättra den beskrivningsmässiga kraftfullheten.

Med hjälp av datorstöd bör vi kunna koppla den "primitiva" datamodellen till beskrivningar av mer komplexa regler. Vi har själva börjat använda Excelerator (Index Technology Corporation, Cambridge) i detta arbete. Med hjälp av verktyget går det att "explodera" sig ned i olika beskrivningsnivåer.

Med verktygsstöd kan vi säkert förbättra systemdokumentation och öppna ADB-tillämpningar för förändringar och för systemförvaltningens krav på insyn. Men användarnas och vår egen svaghet för resultatnriktad frejdighet i utvecklingsarbetet, gör nog att standardiseringsorganens rekommendationer om att specifik tillämpningar till hundra procent, förblir en vällovlig utopi.

2.7 Erfarenheter av datamodellering vid Televerket

Ulf Olofsson, Televerket

2.7.1 Inledning

ADB-service har använt datamodellering i några olika projekt sedan ett år tillbaka. Erfarenheterna hittills är positiva. Dock har vi sett ett behov av datorstöd (data dictionary) om vi skall kunna utnyttja denna metodik fullt ut. Vidare är det viktigt att utveckla metoder för hur man kopplar informationsbehov och händelsekedjor till datamodellen. Arbete med detta pågår.

Datamodellering används idag som en del av verksamhetsanalysen och för databasutformning. Vi diskuterar f n om och i vilken form det kan användas för informationsadministration (IRM). Inom det senare området har vi idag redan gjort en del arbeten. Vi har kartlagt och dokumenterat samtliga system och systemsamband som finns för gemensamma system in-

om televerket. Vi har även utarbetat en metod för gemensam termnamnsättning.

2.7.2 Första projektet

För ungefär ett år sedan gjorde vi vårt första försök att beskriva ett verksamhetsområde med hjälp av en datamodell. Verksamhetsområdet var planering av riksnätet. Vi engagerade IRM CONSULT AB för att med deras hjälp genomföra datamodelleringen och för att själva få ytterligare kunskap/erfarenhet (utöver teoretiska) inom området. Modelleringen genomfördes under verksamhetsanalysfasen. Senare i arbetet låg modellen till grund för databas-konstruktion. Systemet utvecklas med Mimer i Vax-miljö. Erfarenheterna av detta första försök var positiva. Så vi har gått vidare och genomfört datamodeller inom ytterligare ett 10-tal projekt. Företrädesvis inom nätområdet, men även för kund-system. Den modelleringsmetodik vi använder är SASMO.

2.7.3 Erfarenheter

I samtliga fall utom ett har datamodellen varit en bra grundsten att bygga vidare på för konstruktion av system. Det fall som inte lyckades så bra var ett försök att med hjälp av en datamodell integrera flera verksamhetsområden. Den slutsats vi drog av detta är att innan man försöker integrera olika verksamheter bör man ha en övergripande modell. Detta för att ha en bild över den totala verksamheten. Ett annat sätt är att göra modeller över de olika verksamheterna och sedan integrera. I annat fall blir arbetet för komplicerat bl a beroende på att modellering måste göras på olika detaljeringsnivåer. En annan erfarenhet vi gjort är att transformeringen av datamodellen till en databas fungerar väl om databashanteraren är av relationstyp. I de fall andra databashanterare används (nät eller hierarkisk) har det inte varit möjligt att göra databaskonstruktionen direkt från modellen. I dessa fall har modellen endast fungerat som en grov skiss för konstruktionen.

Användarna/kunderna har hittills uppfattat datamodellering positivt. Framförallt för att det utgör ett bra hjälpmedel för kommunikation mellan användare och systemutvecklare. Användarna har också använt modellen för att i vissa fall ändra strukturen eller klassificeringen inom sitt verksamhetsområde.

Vårt arbetssätt, som innebär att speciella personer genomför själva datamodelleringen med en projektgrupp eller enbart användare har visat sig vara bra, då den som leder arbetet inte förväntas ha applikationskunskap utan denna kunskap finns i projektgruppen. Detta arbetssätt ger en större dynamik i projektgruppen än om den som leder styr för mycket.

2.7.4 Utbildning och konstruktion av en modell

Modelleringsarbetet har bedrivits parallellt med övriga systemeringsaktiviteter. För att ta fram en datamodell arbetar vi på följande sätt:

- Datamodelleringsseminarium
- Renritning och framtagning av tabeller för vissa objekt eller relationer
- Uppföljning och presentation av modellen och tabeller
- Projektgruppen förfinar och använder modellen som komplement i sitt analysarbete.

Datamodelleringsseminariet genomförs i internatform med ca 10 personer från projektgruppen och/eller användarna. Seminariet leds av två personer som är kunniga på datamodellering men inte nödvändigtvis inom verksamhetsområdet. Seminariet tar 2 dagar att genomföra. Efter seminariet finns en modell som ritas ren och ett antal tabeller tas fram. De frågetecken som uppstår vid framtagning av tabeller och vid renritningen noteras. Detta arbete görs i huvudsak av de som lett seminariet. När detta är klart, ett par dagar efter seminariet, träffas gruppen igen och slutför modelleringsarbetet. Därefter fastslås en preliminär modell och datamodellen används av projektgruppen i deras vidare analys och konstruktionsarbete. Utbildning för att delta i ett modelleringsseminarium och för att förstå datamodellen har hittills varit en del av seminariet, ca 3-4 timmar. Detta sätt att arbeta har tagits väl emot av användarna. En mycket vanlig kommentar i detta sammanhang är "inte visste jag att jag kunde lära mig så mycket om datamodellering på så kort tid". För att få fler som kan leda arbetet med att ta fram datamodeller låter vi någon delta först som observatör (dock krävs aktivt deltagande i seminariet) och sedan som hjälp till seminarieleddningen.

För att dra nytta av datamodellerna utöver att

de är komplement till övriga systemutvecklingsaktiviteter är det fördelaktigt att ha ett data dictionary för att administrera data. Vidare vore det önskvärt att ha ett designhjälpmedel vid framtagandet av modellerna. Detta hjälpmedel kan med fördel vara kopplat till ett data dictionary. Hittills har vi, i vårt arbete, inte använt några maskinella hjälpmedel.

Slutligen kan konstateras att modellering är ett bra sätt att arbeta för analys och konstruktion av system. Dock återstår en del arbete att integrera modellering med övriga systemeringsaktiviteter och konstruktionshjälpmedel.

2.8 Konceptuell modellering vid Lantmäteriverket – erfarenheter

Kjell Degerstedt, Lantmäteriverket

Vid LMV, Gävle, har arbete med KM skett sedan ungefär 6 månader tillbaka. Det hela började med att vi hade haft god framgång med ett 4G-verktyg för relationsdatabaser samtidigt som vi sökte efter systemeringsmetoder. Vi har ännu ej bestämt oss för vilka systemeringsmetoder vi ska införa. KM blir dock med stor sannolikhet ett av våra systemeringsverktyg pga den framgång vi hittills haft med metoden. Inom LMV har vi sedan flera år haft ett utvecklingsprojekt där vi försökt datorisera vår interna ekonomiadministration. I projektet har bl a en kravspecifikation med ISAC-metoden utarbetats och försök att upphandla lämpliga programvaror har gjorts. Pga våra speciella krav och pga att marknaden inte är så väl anpassad till statlig verksamhet har vi inte lyckats hitta lämpliga program. Samtidigt finns inom fastighetsavdelningen på LMV en mycket komplicerad handläggningsrutin, som är anpassad till gällande lagstiftning inom fastighetsområdet och som man tidigt insåg måste egenutvecklas. Ett starkt behov av integrering mellan EA-rutinerna och handläggningsrutinerna finns också pga att de lokala kontoren i organisationen betraktar dessa två rutiner som en. Utöver detta finns inom verket en stor produktionsavdelning som har ett mycket stort behov av ett produktionsadministrativt system, vilket också finns beskrivet i diverse kravspecifikationer från olika delar av organisationen. Kravspecifikationerna beskriver huvudsakligen de olika delsystemens dynamiska egenskaper. På ett mycket sent stadium i projektet blev LMVs dataenhet, KD, inkopplad som rådgivande konsult. KD såg som

sin uppgift att verka som katalysator och försöka jämka ihop alla användarkrav, KD uppfattade att de största bristerna var dålig förståelse mellan olika användargrupper, vilket hade sin grund i begreppsförvirring och dåliga beskrivningar av systemens statistiska egenskaper. Bland många av användarna fanns dessutom en känsla av att det egentligen var ett enda gemensamt system man pratade om.

Med dessa utgångspunkter beslutade vi att introducera KM. Utbildning av användarna har skett direkt i arbetet med modellen, vilket vi upplevt som lyckat eftersom användarna var väl insatta i problemet och dessutom kände stark motivation för att lösa problemet.

Den variant av KM vi valt är en egen version som bygger på svensk litteratur inom området. Projekterfarenheterna har medfört att vi beslutat att förändra metoden på vissa punkter. Vissa begrepp har visat sig vara oklart definierade, t ex kopplingar och kopplande egenskaper.

Ur KDs synvinkel uppfattas metoden nästan som ett trolleri. Från ett mycket förvirrat läge med många ineffektiva diskussioner med ständiga missförstånd bland användarna har vi uppnått en situation där alla begrepp upplevs som mycket exakt definierade och där verkningssgraden i diskussioner mellan användarna har blivit mångdubblad. Dessutom har en fullständig modell för alla delsystem erhållits. Denna modell har visat sig vara betydligt enklare än vad man trodde i utgångsläget. Modellen kommer nu att bli föremål för implementering i ett 4G-språk.

Under arbetets gång har ett datorstöd utvecklats av fastighetsavdelningen med hjälp av vårt 4G-språk. Verktöget kan kort beskrivas som ett data dictionary för KM. Modellen kommer att dokumenteras och underhållas med detta verktyg.

Användarna har inte haft några större svårigheter att lära sig KM trots att den enda utbildning som getts har varit enligt "naturmetoden". Man har naturligtvis haft svårigheter med att finna de rätta abstraktionerna och objekten, men detta har mer med problemets komplexitet att göra. Modellen har omarbetats många gånger men har slutligen antagit en stabil form med hög tilltro hos användarna.

2.9 Modellering hos Volvo Data

Håkan Lövgren, Volvo Data

Varje människa som studerar någon del av verkligheten (ett system) måste bygga sig en bild (en modell) av hur denna fungerar. Detta gäller i högsta grad vid AU-arbete. Modellering inom AU har gjorts lika länge som begreppet AU existerat. Det som förändrats genom tiden är de grundbegrepp kring vilka modellerna byggs upp. (T ex INFORMATIONSMÄNGD, TERM ...) Valet av grundbegrepp är av stor vikt. Om man ej har något begrepp för att uttrycka ett visst fenomen, kan man ju inte heller bygga någon modell. Grundbegreppen kommer också att styra det sätt på vilket man arbetar för att få fram modellen.

2.9.1 Tillbakablick

Om vi ser tillbaka ett 15-tal år i tiden kan vi lägga märke till att begreppen förändrats. I slutet av 60-talet kom SIS/RAS. Flertalet företag tolkade SIS/RAS och byggde egna systemutvecklingsmodeller. Volvo tog fram den s k RASKA-modellen. De tunga begreppen inom problemorienterat arbete var: FUNKTION, MÅL, BESLUT och INFORMATION. Grundfilosofin var: För att uppfylla vissa mål var varje beslutsfattare tvungen att ha viss information. Man analyserade fram informationsbehov och delade upp dessa i TERMer. Termerna utgjorde 'gränssnitt' mot det utrustningsorienterade arbetet.

Det som sedan hände vid Volvo var att metoden 'Normalisering' började användas, till en början främst för databaskonstruktion. Normaliseringen förde med sig nya grundbegrepp, främst OBJEKT och RELATION. Dessa nya grundbegrepp gjorde sakta men säkert intrång i den 'problemorienterade världen'. Från att tidigare ha talat om att information bestod av termer övergick man till att tala om information om objekt, vilka i sin tur hade termer. Det som hänt på senare tid är att objektänkandet fått en allt mer framskjuten roll. Tidigare tog man ofta fram objekten i samband med informationsanalysen. Numera är objekten bland det tidigaste man söker i ett utredningsarbete. Den största fördelen med detta är att risken för begreppsförvirring minimeras, vilket i sin tur leder till kortare utredningstider och högre precision.

2.9.2 Dagens arbetssätt

Volvo data bedriver en ganska omfattande kursverksamhet. I den kurs som behandlar problemorienterat arbete (8 dagar) rekommenderas att man mycket tidigt gör en grov analys av de i verksamheten ingående objekten och deras relationer. Dessutom rekommenderas att identifierande termer (ID-termer) tas fram. Det resultat man får fram förändras och förfinas under det fortsatta utredningsarbetet. När man slutligen når fram till det läge då informationsbehoven är kända, görs en kontroll mot normalformerna. Arbetssättet har anammats av flera, dock ej alla, projekt med gott resultat.

2.9.3 Metoder

För 'den tidiga objektmodellen' finns ett notationssätt. Någon strikt metod används ej. Modelleringsaktiviteterna bedrivs i storgrupp som består av:

- En person som har stor erfarenhet av dataanalys
- Utredare, som vanligen genomgått utbildning (8-dagarskurs)
- Verksamhetskunnig personal som får information (1/2 dag omedelbart före modelleringstillfället).

Den dataanalytiskunnige och utredaren (a) fungerar som katalysatorer och tar fram (ritar) modellen. Den verksamhetskunniga personalen har, i varje fall till en början, endast som uppgift att beskriva sin verksamhet och att fortlöpande kontrollera modellen.

På senare tid har en metod kallad 'Händelseanalys' använts i ett par projekt, med gott resultat. Metoden som ännu är i utvecklingsskedet finns bl a beskriven i dokumentationen från IAS-85, en konferens som anordnades av SISU i mars 1985. Parallellt med 'objektmodellen' används dessutom V-grafsteknik eller 'dubbelteknik' (Yourdan). De olika vyerna (flöde- och objektmodell) stäms av mot varann.

2.9.4 Datorstöd

För att kunna hålla ihop resultaten har ett datorbaserat Data dictionary visat sig användbart. Volvo Data använder idag produkten Datamanager.

I detta dictionary lagras (i idealfallet) beskrivningar av: FUNKTIONER (verksamheter/delverksamheter), INFORMATIONSMÄNGDER, TERMER, OBJEKT och RELATIONER. För detta ändamål har Volvo Data tagit fram en egen medlemsstruktur i dictionary.

Försök med designhjälpmedel, Data Designer och Designmanager, har gjorts. Dessa har dock ej givit önskvärt resultat, varför de ej används. De främsta orsakerna till att denna typ av datorstöd ej används är:

- Man förlorar kontrollen. Speciellt farliga är de hjälpmedel som säger sig utföra någon form av normalisering.
- Omständliga, ofta kommandobaserade, användarinterface.
- Begränsade outputmöjligheter (Grafik).

2.9.5 Dataadministration

Inom Volvo-koncernen finns idag ett fåtal etablerade dataadministrationsfunktioner. Flertalet av dessa har byggts upp med utgångspunkt från ADB-systemen. Det finns dessutom exempel på dataadministrationsfunktioner som byggs/byggts upp med utgångspunkt från verksamheten. Om den sistnämnda kategorin kan generellt sägas att de som adresserat ett specifikt problem eller projekt lyckats bäst.

2.9.6 Framtiden

Systemutvecklingsarbetet kommer att förändras. I princip kan sägas att begreppet ADB-SYSTEM kommer att luckras upp och i stället kommer begreppet DATA att fokuseras. Modelleringsarbetet kommer att bli mycket väsentligt. I och med att nya tekniker gör det möjligt att direkt hämta information ur data-lagren kommer kraven på korrekt modellering att skärpas. I dag kan en dåligt genomförd analys (och därmed felaktig registerstruktur) ofta rättas upp i de program som producerar information. I morgon finns inte denna möjlighet. Dataadministrationen (eller IRM- funktionen) kommer givetvis att få en mycket central roll. Det är denna funktion som ansvarar för kartorna över objekten; de objekt som (i varje fall under detta sekel) kommer att vara det centrum kring vilket allt AU/SU-arbete bedrivs.

2.10 Datamodellering inom Saab-Scania's Flygdivision

Håkan Wall, Saab Scania

2.10.1 Bakgrund

Vi, liksom de flesta andra företag som har skaffat datasystem för det ena verksamhetsområdet efter det andra under en lång följd av år, har behov av att på något sätt få informationsbehandlingen strukturerad.

I sökandet efter angreppsmetoder kom datamodelleringen in i bilden. Vi bestämde oss för att pröva tekniken i mindre skala för att successivt använda den över större områden. Syftet är alltså att se om vi kan låta datastrukturen i företaget ligga till grund för hur vi strukturerar vår informationsbehandling såväl på detaljnivå som i vidare bemärkelse.

2.10.2 Datamodelleringsteknik

Vi har valt att datamodellerna på det sätt som har utvecklats inom SAS och går under benämningen SASMO.

Det går ut på att man i seminarieform under två dagar arbetar fram en datamodell. De flesta deltagarna i seminarier är representanter från den verksamhet modellen ska göras för och de erhåller erforderlig utbildning i början av seminariet.

Ett typiskt modellseminarium genomförs av ett tiotal användare, ett par systemutvecklare och två handledare. Användarnas deltagande och engagemang är en helt nödvändig ingrediens eftersom de sitter inne med sakkunskaper. Modellerandet uppfattas positivt både av dessa och av systemutvecklarna. De personer som ska bli mest involverade i det fortsatta arbetet med modell och system får lite extra utbildning och insikt i hur modellen sedan ska förädlas och användas.

2.10.3 Applikationer som har modellerats

Vi har datamodellerat för applikationer avseende:

- Kalkylsammansättning
- Inköp
- Dokumentadministration
- Arbetsstudie

- Terminal- och kommunikationsnät

i samband med att system för dessa var aktuella att utveckla. Ett försök har också gjorts att datamodellera för ett större verksamhetsområde, nämligen Produktstöd Militära produkter.

Ingen av dessa applikationer är ännu satt i drift och för ett par av dem vilar utvecklingen av olika skäl. Vi tycker ändå att datamodellerandet i sig självt är ett såpass nyttigt inslag i systemutvecklingsprocessen att vi har satsat en hel del på att tillägna oss tekniken och ha egna modelleringshandledare. Ambitionerna är tills vidare att varje systemutvecklingsprojekt ska datamodelleras.

2.10.4 Datamodelleringen i systemutvecklingsprocessen

Vi finner det naturligt att datamodellering genomförs ganska tidigt i ett systemutvecklingsprojekt. Om verksamhetsanalys enligt MBI-metoden är aktuell är det lämpligt att modelleringen kommer in senast i metodsteg 7. Den kan dock göras tidigare men inte innan en ordentlig avgränsning av verksamhetsområdet har gjorts.

Om datamodellering görs i syfte att i grövre mening strukturera upp datalagring för en verksamhet kan den naturligtvis göras när som helst. Det naturliga i det fallet är dock att passa på när det är aktuellt att starta någon systemutveckling i området.

2.10.5 Användning av datamodellen

Direkt i anslutning till att modellen tas fram får den sin användning i det att användarna utreder och pratar sig samman om begreppsapparaten och får den överförd till systemutvecklarna. Den utgör ju också en struktur som kan uttryckas i tabeller vilka logiskt sett utgör organisationen av den datalagring som erfordras. Vi kan alltså använda den som underlag för databaskonstruktion men måste då ta hänsyn till en mängd andra faktorer som framkommer i senare delar av utvecklingsprocessen.

Många gånger återkommer delar av en datamodell i andra modeller och det som tidigare har modellerats kan man dra nytta av i senare projekt. Vi får en samordning av begreppsapparaten inte bara kring ett speciellt system-

projekt utan successivt i hela företaget.

För området Militärt Produktstöd har vi från datamodellen grupperat lämpliga "datalager" och analyserat vilka funktioner i verksamheten som har ansvaret för deras innehåll respektive intresse av att utnyttja dem. Vi fick fram en bra översikt över hur informationsbehandlingen i området behöver ordnas. Detta tas med som underlag då upprättande av nya system och datalager planeras och kan också få till följd att man omarbetar befintliga.

2.10.6 Erfarenheter

Datamodellering ger ett värdefullt tillskott i samband med nyutveckling av system eller överordnade diskussioner om strukturen på data och dess lagring.

Användarna engageras på ett bra sätt i systemutvecklingen och tar sitt ansvar för att datalagringen blir riktig. De drar samtidigt fördelar av att prata sig samman och blir medvetna om vad datalagren kommer att innehålla. Detta ökar möjligheten till att senare utnyttja deras innehåll exempelvis med användarnära programhjälpmedel. I de fall utveckling av system och datalager genomförs av användarna själva så är datamodelleringen ett utmärkt sätt att få rätt struktur på sin lagring.

Vi har ännu inte fått fram rutiner och arbetsätt för att effektivt ta tillvara det datamodellen har att ge och vi behöver också gå in mera för aktiv samordning av de modeller som kommer fram. I det sammanhanget framträder behovet av datorstöd.

2.10.7 Datorstöd

Ett data dictionary, gärna med förmåga att stötta ritandet av själva datamodellen, bör vara det idealiska hjälpmedlet vid datamodellering. Vi har inte så mycket i den vägen men sneglar en del åt DASAK. Det vi har sett av det är dock ännu så länge inte tillräckligt lätt att använda. Ett data dictionary är givetvis inte ett separat datamodelleringsstöd utan bör användas i alla delar av systemutvecklingen.

2.10.8 Sammanfattning

Datamodellering är säkert en metod som bör användas i systemutvecklingsprocess och system-/dataplanering. Det känns riktigt och

praktiskt att analysera datastrukturen på det sättet. Man har användning för resultatet i flera sammanhang och det verkar klart i linje med IRM-filosofin.

För vår del har vi bara börjat modellerandet och känner behov av att:

- öva upp skickligheten
- bli bättre på att omsätta modellen till datalager
- samordna modeller, definitioner etc effektivt
- skaffa datorstöd för modellerandet integrerat med den övriga systemutvecklingsprocessen.

2.11 Erfarenheter av KM hos försäkringsbolaget Skandia

Björn Wählberg, Skandia

I det följande beskrivs en del erfarenheter från modelleringsprojekt inom Skandia.

2.11.1 Syfte, användningssätt

Syftet med vårt modelleringsarbete har varit

- Att definiera och samordna olika intressentgruppers syn på verksamhetsområdet.
- Att underlätta samordningen av samma slags data i olika system.
- Att ge underlag för bedömningen av kommande krav på databaser etc.
- Att få en gemensam begreppsapparat.

Genom att framställa en objektmodell har vi erhållit

- modeller över systemets lagringskrav
- underlag för fortsatt analys och konstruktion
- underlag för dimensionering av resurser och planering av lagrad information
- hjälpmedel för kommunikation med användarna och för ADB-personalen
- underlag för genomgång och avstämning av informationsbehoven.

Objektmodellen har i och med detta blivit grundstommen för allt fortsatt arbete med att analysera och utveckla systemet. Vid utformningen av systemet har man snabbt kunnat tala om för användarna om det var ett rimligt eller orimligt krav de kom med. Det har också varit lätt att skapa logiska poster eftersom

objektmodellen brutits ned så pass långt att naturliga logiska poster erhållits. Man har även kunnat se hur delsystem hängde ihop med det totala systemet.

2.11.2 Applikationer

Tillämpningarna utgörs av nationella och internationella försäkringssystem. När det gäller internationella återförsäkringar har det varit extra givande med objektmodellering eftersom den begreppsflora som existerar där är ganska svårbegriplig och komplex. Inte ens alla som håller på med detta kände till hur allt hängde ihop. Det visade sig också finnas begrepp som har olika betydelse beroende på i vilket sammanhang man använder sig utav begreppet.

2.11.3 Modelleringsarbete, ett exempel

Först framställde vi utredare en objektmodell över den verksamhet systemet skulle leva i. Denna lade vi sedan åt sidan. Anledningen till att vi själva ritade en objektmodell var att vi ville ha en grund att stå på för vidare diskussioner. Därefter hade vi ett användarseminarium där vi först lärde ut grunderna för hur man ritar och beskriver en objektmodell. Efter detta delades användarna upp i olika grupper beroende på vilken typ av verksamhet de höll på med och dessa grupper fick rita varsin objektmodell avseende hur de såg på verksamheten.

När dessa grupper hade ritat sina objektmodeller hade vi en gemensam genomgång då vi ritade en ny objektmodell utifrån vad de olika grupperna hade kommit fram till. Den objektmodell som blev resultatet av detta har varit ganska stabil.

Efter detta har vi finslipat objektmodellen parallellt med analysen av de funktioner som skall ingå i systemet.

2.11.4 Utbildning, metod, datorstöd

Den metod där vi använt objektmodellering är SAK-metoden. Den version av SAK som vi använder oss av är en företagsanpassad version och i den har vi beskrivit hur objektmodellering går till. Alla som går vår interna SAK-utbildning får alltså lära sig grunderna till modellering.

Inget grafiskt datorstöd har använts utan vi ritade på papper. Det datorstöd vi haft tillgång

till är IBM's Data Dictionary där vi matade in alla objekt för att sedan koppla upp relationer mellan objekten.

Ett önskemål är att man skall kunna rita objektmodellen grafiskt och att det skall finnas grafik med intelligens bakom, dvs flyttar man ett objekt så skall alla relationer hänga med, tar man bort ett objekt skall alla relationer till objektet försvinna.

2.11.5 Allmänna erfarenheter

Positivt har varit att man lätt har kunnat förstå hur allt hänger ihop. Det har också varit lätt att diskutera med användarna kring objektmodellen, i och med att användarna kunnat se sin egen situation i modellen. Det är lätt att få tag i gränssnittet mellan de olika delsystemen.

2.12 Konceptuell modellering inom Ericssonkoncernen

Christer Dahlgren, Ericsson

Ericsson är en koncern med verksamhet i över hundra länder. Verksamheten är uppdelad i åtta affärsområden som var och en består av flera företag. Vissa av de större företagen är indelade i enheter. Egen ADB-verksamhet kan förekomma på sådan enhetsnivå, på bolagsnivå, på affärsområdesnivå eller genom utnyttjande av ett centralt placerat bolag för databehandling. Ofta sker databehandlingen både på lokal och minst en mer central nivå.

Databehandlingen befinner sig i olika utvecklingsstadier på olika ställen inom gruppen, varför det är svårt att svara generellt på frågor om var Ericsson står i datamodelleringsfrågor. För hela gruppen finns dock viss sammanhållning och erfarenhetsutbyte genom de centrala koncernstaberna som finns knutna till koncernledningen. Denna redovisning är grundad på erfarenheter på koncernstabsnivå men är inte hundra procentigt täckande det som sker inom koncernen.

En mer dataorienterad syn på ADB har funnits inom Ericsson de senaste åren och hösten -84 tog koncernstaben initiativet till en gemensam satsning på "IRM" inom gruppen. Syftet med denna var inte att standardisera hanteringen av data utan att skapa ett gemensamt grund-

koncept och att få ett forum för utbyte av erfarenheter inom denna "nya" syn på databehandling.

En av anledningarna till det pågående arbetet med datamodellering är en förändrad syn på data och dess betydelse för koncernen. Våra användare har t ex mycket skilda krav på åtkomst till data. Den ena ytterligheten representeras av de användare som använder en eller några få centrala applikationer och som vet exakt vad dessa data står för, vilka koder som betyder vad och vilket status m a p uppdatering etc som data har. Andra användare kan precisera sina krav utgående från sin verksam-

het men vet inte ens om det finns lagrade data som beskriver detta.

En av grundpelarna i den förändrade grundsynen är en modell för utveckling och förvaltning av datasystem som betonar uppdelningen av arbetet i databehandling och datastrukturering. Vår utvecklingsmodell visar dessutom vilket resultat som uppnås vid olika utvecklingsskeden men beskriver inte i detalj hur utvecklingsarbetet skall bedrivas.

Utveckling och förvaltning för datastrukturering sker sålunda separerat från utveckling och förvaltning av databehandlingsdelar.

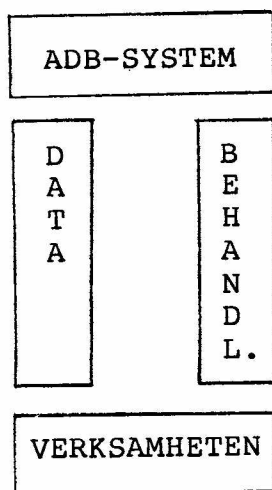


Figure 2-4: Utveckling och förvaltning för datastrukturering sker skilt från utveckling och förvaltning av databehandling.

Såväl datastrukturering som databehandling måste bygga på verksamheten. Genom att datastruktureringen utförs separerat från framtagning av bearbetningsdelarna, blir det lättare att använda hjälpmedel som använder befintliga datastrukturer för att framställa rapporter och terminalbilder.

Anledningen till att datamodellering/konceptuell modellering har använts inom Ericsson är att förstärka synen att data har ett värde i sig och att datas existens inte skall kopplas samman med dess bearbetning (program). Detta synsätt påverkar alla nivåer från verksamhetsanalys till utformning av databaser som omfattar begränsad mängd data. De praktiska erfarenheterna rör framför allt framställning av datamodeller som sedan har realiserats med hjälp av relationsdatabashanteringssystem, framförallt för att belysa databehovet inom ett verksamhetsområde.

En datastruktur som utformas verksamhetsanpassat blir mer flexibelt användbar än en som utvecklas anpassat efter en applikation. Dataelementens samband beskrivs i datamodeller som, mer eller mindre totalt, omfattar all den data som behövs för att beskriva aktuell del av verksamheten samt de relationer som råder mellan ingående dataelement. Dessa datamodeller blir oberoende av hur realiseringen av data senare kommer att göras till databaser. De är endast tidsmässigt korrelerade till bearbetningen, så att när en viss bearbetning skall realiserats måste berörd del av datamodellen finnas som databas.

Datamodellering har använts för ett flertal applikationer inom helt skilda områden, t ex för att beskriva vårt interna datakommunikationssystem, för att analysera reexportproblematiken och för att beskriva framställning av kunddesignade chips. Dessa exempel omfat-

tar såväl teknisk som administrativ databehandling och visar på bredden av vår användning av datamodellering. Arbetssätten har varit likartade och erfarenheterna är mycket samstämmiga. Skillnaderna mellan teknisk och administrativ databehandling har varit försumbara.

Modelleringsarbetet har, i enlighet med vår systemsyn, i allmänhet bedrivits som ett alternativt sätt att beskriva databehovet. Vid datamodelleringen har inte realisationsbegränsningar beaktats och ingen hänsyn har tagits till befintlig datastruktur.

Användare har alltid medverkat aktivt i framtagning av datamodellerna. Ibland har ADB-specialister medverkat i detta begreppsdefinitionsarbete men det tycks inte finnas någon positiv korrelation mellan sådan medverkan och det slutliga resultatet.

För de användare som medverkat i datamodelleringsseminarier har kunskapen varierat från de som aldrig har funderat över skillnaden mellan termförekomst och de värden som termerna kan anta till de som läst allt om konceptuell modellering. Det verkar inte heller finnas någon direkt koppling mellan storleken på den teoretiska kunskapen om konceptuell modellering och det färdiga resultatet. Vi har genomfört seminarium med teoretiska fulländade seminarieledare som inte alls uppnådde förväntat resultat medan liknande möten ledda av seminarieledare med mindre teoretisk grund varit fullträffar. Vad som verkar svårast är att få de som behärskar tekniken, speciellt om de är vana vid konventionell systemering, att låta bli att dominera utvecklingsprocessen.

Datamodelleringsseminarierna har i allmänhet genomförts som två-dagars seminarier, oftast geografiskt skilt från den normala arbetsplatsen. Medverkande i seminarier har fått en grundläggande bild av konceptuell modellering under en halv dag innan seminariet. Ibland har denna grund varit tillräcklig, ibland har ytterligare teori behövts innan arbetet med modellerna kunnat påbörjas. Ytterligare förklaringar och bakomliggande teorier har successivt presenterats alltefter behovet uppstått. Det viktiga är att inte presentera all den teori som seminarieledaren behärskar utan att beskriva det som behövs för att arbetet skall klaras av. Den viktigaste egenskapen hos seminarieledaren är att vara lyhörd för vad delta-

garna behöver.

De första modelleringsseminarierna inom Ericsson genomfördes med hjälp av konsulter från olika bolag. Senare har motsvarande genomförts helt i egen regi. Kvaliteten på konsulter som medverkat har varierat från de som förstått att datamodellering är populärt till de som kan få fram ett användbart resultat tillsammans med en grupp människor. De metoder som använts har framför allt varit av typ Objekt — Relationer mellan objekt och haft en grafisk form. Ibland har relationerna namngivits i själva modellen ibland inte. Normaliseringsteori har alltid funnits med i metodansatserna, i några fall har en alltför stark betoning av dessa teorier skrämt de medverkande "användarna". I flera fall har beskrivningssättet justerats under pågående seminarium så att det bättre passar aktuell applikation och de medverkande.

De datorstöd som använts har framförallt varit relationsdatabashanterare i vilka datamodellen realiserats så att begärda bearbetningar kunnat simuleras. Vilken relationsdatabashanterare som använts har berott på vilken som enklast stått till buds. Den slutliga realiseringen har ofta utformats för annan databashanterare än den som använts för att simulera bearbetningen. Sådana prototyprealiseringar har genomförts såväl under pågående seminarium som vid uppföljningsmöten efteråt. Vid något seminarium har persondator använts för att direkt dokumentera den modell som överenskommit så att deltagarna fått den med sig hem för vidare funderingar. Detta har varit mycket värdefullt och kommer säkert att göras fler gånger.

2.13. Informationsmodellering inom Vattenfall

Bror Norén, Vattenfall

2.13.1. Tilltänkta användningsområden

Informationsmodellering eller konceptuell modellering röner stort intresse inom Vattenfall sedan några år tillbaka. Detta beror bl a på att vi arbetar en hel del med grundläggande kriterier och åtgärder för ökad datakvalitet och integrerbarhet i de informationssystem som utvecklats inom Vattenfall. För detta utgör informationsmodellering ett hjälpmedel för både informationsflödesanalys och databas-

konstruktion.

Kortfattat kan sägas att vi är intresserade att utnyttja informationsmodellering som hjälpmedel för

- övergripande verksamhetsanalys
- konstruktion av aktiva totalintegrerade "information dictionaries" som "logiska" databaser i databassystem (t ex som överbyggnad till relationsdatabaser)
- integration av informationsbehandlings-system
- användarorienterad systemanalys för utformning av dataunderlag i användarkravspecifikationer
- databasadministration, underhåll av databaser
- gemensam "systembild" användare, systemkonstruktör, dataadministratör, databasadministratör
- informationsadministration, övergripande ansvar för system o datastrukturer
- logisk strukturering av metadatabaser i distribuerade informationsbehandlings-system dvs övergripande databaser för administration av systemnodernas databaser
- grafisk visualisering (bildskärm, plotter etc) av logiska strukturer i databaser

2.13.2. Använda modelleringsmetoder

Två olika modelleringsmetoder, dels en relationsansats och dels en egen associationsansats enligt följande beskrivning, prövas inom Vattenfall.

A. Relationsansats (SASMO)

Den SAS-utvecklade relationsmetoden SASMO, som är kommersiellt tillgänglig har tillämpats i ett antal projekt inom Vattenfall. Inom ramen för projekten har därvid sk SASMO-seminarier avhållits. Till en början leddes dessa seminarier av SAS dataplaneringsgrupp. Senare har vi dock utbildat egen perso-

nal för ledning av sådana seminarier.

De systemutvecklingsprojekt SASMO-metodiken tillämpats på har i allmänhet befunnit sig i förstudiefas eller tidig konstruktionsfas. Som allmänt omdöme med utgångspunkt från våra erfarenheter av SASMO kan sägas följande:

Metoden är enkel att lära in och kan snabbt börja tillämpas som ett kommunikationshjälpmedel mellan systemutvecklare och användare. Den seminarieteknik som SAS utvecklat för databasmodellering har visat sig utmärkt. Två intensiva arbetsdagar räcker, att i en grupp om ca 8 a 10 personer bestående av användare och ADB-personal lära in tekniken, etablera fungerande kommunikation användare/systemutvecklare samt utveckla en grundmodell.

Genomgående tillämpning av metoden från förstudiefas till konstruktion och driftsättning har inte prövats. Styrkan i metoden ligger, som vi ser det, i att man enkelt får en grundmodell som man kan "diskutera" över. Detta underlättar kommunikationen med användarsidan.

B. Associationsansats, "Lindqvistmetoden"

Inom Vattenfall utvecklas olika informations-system för processdatabehandling. De verksamheter dessa system skall betjäna präglas av krav på stor exakthet i informationsbehandlingen trots ständig "omvärldsförändring". Kraven på strukturell stabilitet (överblickbarhet) och förändringsbarhet i databassystemen är därför mycket stora.

Vid Vattenfall har därför utvecklats en teknologi för att bygga system som motsvarar dessa krav. Ett sådant system, benämnt KK-info, för nationell driftledning av elkraftproduktion och elkraftöverföring har varit i drift inom Vattenfall sedan närmare 10 år.

Databaslösningen i detta system utgör i dagens språkbruk i princip en 5:e generationens databaslösning med helt programberoende datalagring på typnivå samt innehållande ett aktivt totalintegrerat "information dictionary". Den strukturella lösningen i detta system var vid systemets tillkomst mycket svår att förklara beroende på att databasprincipen dels var "annorlunda" och dels på att en relevant begreppsvärld för denna teknik då ännu saknades.

Databaslösningen har dock sedan något år analyserats i detalj och försetts med dels en informationsmodellansats med komplett notationsätt och dels med en metodik för överföring av modell till termkatalogen i "information dictionary". Detta har sedan givit upphov till en metodansats för generell tillämpning i systemutvecklingsprojekt. Metoden har kallats Lindqvistmetoden efter den person (John Lindqvist, Vattenfall) som ledde utvecklingsarbetet av KK-infosystemet i början av 1970-talet.

B. 1 Modellansats

Grundprinciperna i KK-info-systemets logiska databas liksom i det konceptuella notationsätt som utvecklats utgår från "e-meddelandeteknik". Sålunda används e-meddelanden av nedanstående typer

- Typ V
Objekt, egenskap, värde
- Typ O
Objekt, egenskap, objekt

Dessa e-meddelandetyper har sedan "kompletterats" med klassificering av objekten i **objektklasser** och i **objektdelklasser** samt kategoriindelning av objektklasserna i objektklasskategorier enl följande.

- Typ Vk

Objektklasskategori, objektklass, objektdelklass, egenskap, värde
- Typ Ok

Objektklasskategori, objektklass, objektdelklass, egenskap, (objektklasskategori), objekt

Med utgångspunkt från dessa modifierade e-meddelanden har en konceptuell modellansats av associativ typ konstruerats. För den grafiska notationen i denna tillämpas kriterier och symboler enligt mängdläran.

För konstruktionen av den logiska databasen löses relationstabeller ut ur den konceptuella modellen. Dessa relationssamband kan sedan automatiskt sammanställas till relationer på 3:e eller 4:e NF. Modellens strukturella uppbyggnad möjliggör mycket komplicerade "nyckelsamband" och tidssamband utan att

den blir svåröverblickbar.

Kategori-indelning av objektklasserna i objektklasskategorier medverkar till att synonymproblemen i databassystemets termer löses. Varje term kan därigenom alltid hänföras direkt till rätt "region" på "kartan" (= konceptuella modellen) dvs till rätt systempunkt på hög nivå i bakomliggande verklighet. Synonymproblemen i termerna löses för övrigt till sin helhet m h a kombination modell/termkatalog genom att varje term kan placeras i rätt systempunkt. Genom att införa ett begrepp "objektdelklass" som alltid får representera "tid" (t) löses hur "tid" (datum, år, månader, sek, -sek etc) skall representeras i modell och termkatalog. På detta sätt kan termer associerade till även komplexa tidssamband ändå enkelt knyts till rätt objektklass.

Som tidigare sagts avbildas exakt den konceptuella modellen strukturellt i termkatalogen. För avbildning av associationerna (pilarna) i modellen används därvid e-meddelande typ kategori i Ok. Vid strukturella förändringar i databasen genereras ny logisk databas dvs termkatalog. Eftersom termkatalogen i sig utgör en "parametrisering" av modellen så kan modellen även till sin helhet lagras i den fysiska databasen. Genom att tillfoga ett grafiskt hjälpmedel kan sålunda alltid en aktuell modell för databasens strukturella innehåll tas ut på bildskärm eller plotter och i framtiden kanske även accesser till databasen göras direkt "från" den konceptuella modellen.

Man kan säga att datamodellering vanligen handlar om att "skapa en struktur i sina data-samband i ett system". I denna metod kan man snarare tala om att skapa en "exakt" bild (systembild) av en verksamhet i vilken man kan placera in data. Denna "systembild" är gemensam för flera inblandade parter. Sålunda kan användarna identifiera sig i denna, ADB-teknikerna som skall realisera den fysiska lösningen identifiera sig i denna, datorsystemet "identifiera sig" i denna (genom att den logiska databasen utgör en direkt översättning av den konceptuella modellen). Metoden innebär även möjligheter för användarsidan att i detalj specificera sina datastrukturer redan i kravspecifikationen i olika systemutvecklingsprojekt. Vid tillämpning av metoden för databas-konstruktion kan också sägas att informationsanalys, konstruktion av informationsmodell/termkatalog och konstruktion av logisk

databas utgör en och samma process.

Trots att det "ursprungliga systemet", konstruerat efter dessa principer, varit i drift så länge, så är det först nu när principen analyserats och försetts med metodansats (konceptuell modell/logisk databas) som metoden kan tas i bruk för konstruktion av nya system.

Denna teknik förutsätter engagerad användarmedverkan för att ge alla nyttovärden. Beställare/användare är ju de enda som kan överblicka bakomliggande verksamhet dvs de som kontinuerligt under konstruktionens gång, kan garantera att informationsanalys/modell/konstruktion ligger i linje med "fastställd systembild".

Den fysiska lagringen av data i ett databasprojekt kan göras på olika sätt. En metod, som prövas på Vattenfall, är att använda en relationsdatabas för datalagring och låta den logiska databasen enligt Lindqvistmetoden utgöra en överbyggnad till denna. Detta tillämpas för närvarande i ett större systemutvecklingsprojekt (DAROS) inom Vattenfall.

Metoden "konceptuell modell/information dictionary" där termkatalogen i dictionaryet kan utgöra en logisk databas i en databas-konstruktion har många löftesrika dimensioner. Metoden ger total överblick över data på termnivå liksom över dataförekomsternas kvalitetsparametrar. Detta tillsammans med att synonymproblem lösts (termkatalogen innehåller "allt" utom termnamn) ger möjligheten till konstruktion av gemensamma data-lager, meningsfylld integration av system m m. Detta är kanske mest intressant i tekniska system av processkaraktär men kan tillämpas inom vilken typ av databasorienterad systemutveckling som helst.

2.14. Sammanfattning

I det följande redovisas sammanfattningsvis en del av de viktigaste synpunkterna i bidragen ovan utgående från vissa frågeställningar som vi i vår förfrågan bett att få belysta. Dessa svarar ungefär mot följande 6 delavsnitt.

2.14.1. Användningsområde, syfte

Som det kanske viktigaste syftet med KM framhålls att man med detta hjälpmedel kan reda ut den "begreppsförvirring" som ofta vi-

sar sig föreligga då flera människor, kanske från olika delar av en verksamhet, kommunicerar. Modelleringsarbetet medför att användare pratar sig samman om vilka begrepp som skall användas, innebörden i dessa och sambanden mellan dem. Redan i och med detta är mycket vunnet, eftersom man på så sätt får en stadigare grund för en fortsatt systemutveckling. I bästa fall tänker man sig åstadkomma en samordning av begreppsapparaten för hela företaget och i förlängningen en förbättrad informationsadministration med stöd av ett Information Dictionary.

Ett viktigt användningsområde för den konceptuella modellen är databaskonstruktion och här anses det väsentligt att man åstadkommer "kongruens mellan begrepps- och datastruktur". Systemarkitekturen skall vara "datadriven" och inte baserad på mer eller mindre tillfälliga informationskrav.

2.14.2 Hur skall modelleringsarbetet bedrivas

Konceptuell modellering tycks vara en aktivitet som bör komma in på "startsträckan" vid en systemutveckling. Den kan dock i viss mån utföras parallellt med annat arbete och som ett komplement till andra metoder. Genomgående betonas vikten av att verksamhets-kunnig personal deltar i modelleringsarbetet, som gärna bedrives i ett koncentrerat seminarium på ett par dagar där också en eller flera modelleringsexperter deltar. Det anses dock viktigt att experten/seminarieledaren inte styr arbetet i alltför hög grad. Den gruppdynamik som uppstår anses värdefull och även om den konstruerade modellen inte skulle komma till uttrycklig användning ytterligare, så har seminariet ändå varit nyttigt genom den förhöjda "consensus" som erhållits.

2.14.3. Utbildning

Det tycks vara lätt att få människor att förstå nyttan med att reda ut begreppen, inte bara för att de är viktiga då det gäller att utveckla ADB-system, utan också för att en väldefinierad begreppsstruktur är väsentlig för företagets verksamhet i stort. Det är uppenbarligen också lätt att lära ut typer av modelleringsteknik som här har använts.

Utbildningen sker lämpligen i anslutning till ett modelleringsseminarium. I något fall använ-

der man 1/2 dag strax före och för en annan räcker det med 10 minuter i början av seminariet. Särskilda kurser för metodexperter förekommer i form av 8-dagars-kurs eller som successiv utbildning under medverkan i modelleringsarbetet.

2.14.4. Metodik

Ett antal olika mer eller mindre kända metoder används. Exakt vad dessa innebär låter sig knappast förstås utan grundligare beskrivningar än vad som ges här. Vanligt förekommande inslag tycks dock vara

- grafer
- tabeller, relationer
- normalisering
- händelser, händelseanalys

Det är uppenbart att framför allt grafisk beskrivning är betydelsefull då det gäller att åskådliggöra en begreppsstruktur. Vid sidan av detta förekommer ofta olika slag av tabeller. Gemensamt för de metoder som här förekommer tror jag är att de i första hand syftar till att få fram och klargöra huvudbegreppen inom ett område. Mer hinner man knappast på två dagar. Allmänna regler och villkor är ofta svåra att hitta, förutom att de är besvärliga att uttrycka på ett väldefinierat, men ändå lättförståeligt sätt.

2.14.5. Datorstöd

Ofta har olika slag av datorstöd använts, t ex

- designhjälpmedel som DASAK eller Excelerator
- Data dictionaries som t ex DATA-MANAGER

Några har gjort egna datorstöd, men också de som arbetat helt manuellt har upplevt behovet av ett stöd, framför allt av data dictionary-typ, helst också med grafikmöjlighet.

2.14.6. Problem

Trots den entusiasm inför KM som hjälpmedel i systemutveckling och de positiva erfarenheter som redovisas ovan, är konceptuell modellering en teknik där problem på många olika nivåer återstår att lösa. Bland det som antyds i bidragen finns bl a.

- Problem som har att göra med hur olika företeelser i verkligheten skall avbildas i modellen. Vad skall vara objekt, vad attribut etc? Här kan i viss mån modelleringsansatsen och även syftet med modelleringen bli styrande, vilket man måste vara medveten om.
- Vad är det som skall modelleras, verkligheten/verksamheten eller ett tänkt eller verkligt informationssystem/databas? Det är lätt att oavsiktligt halka in på beskrivning av rena informationskrav, vilket vanligtvis inte är avsikten.
- Hur får man med mer av regler, villkor och dynamik hos objektsystemet i modellen? Hur skall notationen se ut för att samtidigt vara enkel och så rik att den ger möjlighet att beskriva någonting mer än enbart vissa huvudobjekt och deras samband.

Det är bl a svårigheter av denna typ, som vi hoppas komma närmare lösningen till, genom det arbete som bedrivs av SISU tillsammans med SISUs olika intressenter.

3. FORSKNING OM KONCEPTUELL MODELLERING

I detta kapitel ges en populär och översiktlig bild av den forskning som bedrivs nationellt såväl som internationellt och som anknyter till konceptuell modellering (KM). Området KM är långt ifrån väl avgränsat. Området är under snabb utveckling och påverkas ständigt av forskning och tillämpningar i flera angränsande data- och systemvetenskapliga områden. Vi börjar kapitlet med att belysa dessa relationer. Därefter tar vi upp ett flertal aktuella problemställningar som är föremål för forskning och utveckling.

Det kan också vara intressant att veta vilka internationella samfund, arbetsgrupper, konferenser och symposia brukar behandla temata av relevans för forskning i KM. Vi ger en orientering om detta i avsnitt 3.4.

Det måste betonas att informationen i detta kapitel ej gör anspråk på fullständig täckning av forskningen. En sådan sammanställning skulle kräva flera års arbetsinsats. Vi tror dock att det som presenteras här kan för intresserade läsare utgöra en lämplig "inkörsport" i den aktuella KM-forskningen.

3.1 En teoribaserad syn på konceptuell modellering

Vad är en konceptuell modell egentligen? Uppfattningen härom varierar starkt beroende på vem man frågar. Några ser det som någonting som beskriver strukturen och betydelsen (semantiken) hos data i en databas. Andra ser det som en beskrivning av en "teori" om en tillämpning — en verksamhet. Bägge synsätten är rimliga och riktiga. Teori-synsätt är det mest omhuldade i forskarkretsar och avspeglar den influens på konceptuell modellering — och databehandling överhuvudtaget — som har sitt ursprung i sk kunskapsbaserade system och femte generationens datorsystem.

ISO:s 100%-princip (se kapitel 1) kan också tolkas som att varje idag befintligt datasystem — bestående av en databas samt ett antal bearbetande program — kan ses som ett system som förverkligar en "teori om en tillämpning". En "vanlig faktura" som framställs av datasys-

temet kan ses som en logisk konsekvens (en slutsatsdragning) av en given mängd "premisser". Hur kan man nu säga att ett datasystem representerar en teori?

En teori om ett visst objektsystem kännetecknas av att man först definierar ett språk som beskriver vilka typer av utsagor man får göra om tillämpningen. Exempel på detta ges i figur 3-1.

- (a) x är en anställd
- (b) y är ett projekt
- (c) x arbetar i projekt y
- (d) x leder projekt y
- (e) x arbetade y timmar dagen d
- (f) x är en person
- (h) Anställd x har månadslönen y

Figure 3-1: Utsagetyper

I våra datasystem definieras detta formella språk av dels databasschema (posttypbeskrivningar m m) dels av indatabeskrivningar, som ju utsäger vilka typer av utsagor om tillämpningen kan tillföras systemet. Givetvis kan vi formulera ovanstående typer av utsagor mer "kompakt", t ex genom en predikatlogisk notation ANST (x), PROJ (y), ARBPROJ (x,y), osv men det är oväsentligt för detta resonemang.

I tillägg till språkdefinitionen har en teori också ett antal axiomatiska antaganden (grundantaganden) om hur dessa utsagor (formler) får manipuleras och skapa nya utsagor (formler). Vi skall inte krångla till framställningen med detta här utan bara säga att dessa, förenklat sett, motsvarar logiska och matematiska grundlagar.

Ett informationssystem kan nu innehålla en mängd 'fakta' — utsagor — av den typ som definierats i figur 3-1. Exempel på detta visas i figur 3-2. Enligt [6] kallas detta en "informationsbas".

Inom området kunskapsrepresentation är förhållandet det omvända — åtminstone för när-

varande. Så kallade kunskapsbaserade system kännetecknas av förhållandevis stora mängder av generell kunskap: regler och antaganden om ett objektsystem. Faktamängden antas här vara förhållandevis liten. Den centrala frågan här är inte effektiv faktalagring utan systemets förmåga att 'resonera', att dra slutsatser av såväl allmän som specifik karaktär. Frågan kan sägas omfatta såväl problemet hur olika slag av kunskap kan representeras som problemet att finna effektiva metoder för slutsatsdragning.

Hos de 'klassiska' programmeringsspråken såsom t ex Algol, COBOL och FORTRAN, har stora, permanenta faktamängder alltid varit styvmoderligt behandlade. Dessa och andra 'klassiker' såsom LISP, Simula och Pascal har goda mekanismer för strukturering och bearbetning av data, men då skall dessa först vara 'inlästa' i datorns primärminne. Intresset har alltså fokuserat på begrepp och mekanismer för utformning av algoritmer för effektiv och problemnära manipulation av fakta i datorns primärminne. Andra problem som har stått i centrum är formell specifikation av program, synkronisering av parallella processer, bevis av algoritmers korrekthet och analys av andra egenskaper hos program (algoritmer). Men intresset har i allmänhet ej gällt hantering av stora och permanenta datamängder — dvs sådana data som 'överlever' ett programs exekvering och 'delas' av många användare.

Området programmeringsspråk, programmeringsmetodik och programmeringsteori är tveklöst det mest etablerade. En mängd lovande forskningsresultat och idéer om nya språkströmmar ständigt till. En i KM-sammanhang intressant ansats är det sk "Abstract Data Type" begreppet (ADT). En ADT specificerar en klass av objekt med viss datastruktur och definierar de operationer som får utföras på objekten. De klassiska exemplen på ADTs är köer och 'stackar'. En kö ses som ett objekt som 'innehåller' andra objekt ordnade i något avseende, t ex efter ankomsttid. Tillåtna operationer är, säg, att ta första elementet i kön och att lägga till ett element sist i kön.

Men en ADT kan i princip även användas för att specificera, säg, objekttypen "anställd" och dess olika attribut såsom namn, adress och lön. Tillåtna operationer på denna ADT skulle

kunna vara t ex att skapa (införa) en anställd, att ändra dennes namn eller att öka dennes

lön.

Även om tanken bakom ADT-begreppet främst har varit att underlätta strukturering och modularisering av stora programsystem, kan vi lätt se sambandet till det objektorienterade synsättet inom konceptuell modellering. Under de senaste fem åren har insikten att de tre områdena DB (Databaser), PS (Programmeringsspråk) och KR (kunskapsrepresentation) har mycket att erbjuda varandra vuxit allt starkare. Flera arbetskonferenser har anordnats (se mer om detta i avsnitt 3.4) med representanter för dessa tre inriktningar. Man har lagt fram och diskuterat sina synsätt, metoder och teorier och undersökt vilken kunskap som kan 'exporteras' från en inriktning till en annan.

Dessa konferenser har brottats med en rad problem, i synnerhet de första i raden. Forskare från de tre områdena DB, och PS och KR har talat skilda språk och haft skilda begreppsramar. Kommunikationsproblem har funnits även inom områdena — forskare har 'bekänt sig till' olika synsätt och terminologi. Med tiden har man dock kunnat märka ett ökande intresse för varandras problemställningar och resultat. Insikten växer också att framtida informationssystem behöver byggas med kunskap från samtliga dessa (och ytterligare) forskningsområden.

De 'exportartiklar' som man för närvarande ser som viktiga och ömsesidigt intressanta hos DB-, PS- och KR-områdena är följande:

DB-området kan exportera kunnande om effektiv hantering av stora mängder data, (parallella) transaktioner, hantering av tillförlitlighetsfrågor, återstarts- och säkerhetsproblem. I DB-området finns även kunskap om hur man effektivt hanterar 100-tals parallella användare som "delar" en gemensam faktabas.

KR- och AI-områdets främsta exportartiklar är, som vi diskuterat ovan, representation och hantering av mer generell kunskap och språk med vars hjälp 'resonerande' system och 'expertsystem' kan byggas. Det är ingen tvekan om att framtida, databasorienterade informationssystem kommer att ha (viss, begränsad) förmåga att 'resonera', att ge (viss) 'expert-hjälp' och att ge råd, hjälp och anvisningar till ovana användare. Hantering av 'nära naturligt'

språk är därför en annan viktig exportartikel från AI-området.

PS-områdets viktiga exportartiklar är främst Abstract Data Type begreppet och olika metoder för formell specifikation och analys av system och program (t ex ur korrekthetssynvinkel). PS-området är det — matematiskt sett — mest 'mogna' vilket innebär att det nått längre än KR-, och i synnerhet DB-området, vad avser exakt definition av de begrepp och mekanismer man använder. Detta underlättar kommunikation (förståelse) och möjligheten att bygga på varandras resultat.

Forskningsområdet KM — Konceptuell Modellering, som ligger någonstans emellan DB- och KR/AI-området — besväras av denna brist på stringens och formella definitioner. Även om situationen håller på att förbättras, så råder i detta område brist på exakta definitioner och det medför svårigheter att kommunicera resultaten till varandra. Många modelleringsansatser är 'löst' beskrivna i naturligt språk och med enkla grafer. Detta gör att ansatserna är svåra att jämföra och utvärdera i syfte att förbättra dem.

Det finns ytterligare två forskningsområden med omedelbar relevans för Konceptuell Modellering. Det första är det 'traditionella' området Informationssystem och Systemering (IS) och det andra är det under senare år alltmer expanderande området KIS- KontorsInformationssystem.

Det traditionella IS-synsättet kan kanske karaktäriseras som "vi har data (information) och vet precis hur vi vill behandla dessa". Grundtanken bakom traditionell IA — informationsanalys — är just detta. Arbetssättet var då att skissera processer som utbyter data (informationsmängder) med varandra och med funktioner i en organisation (informationsflödesmodeller). Detta kan naturligtvis göras på olika hierarkiska nedbrytningsnivåer. Under senare år har man dock kunnat märka en klar trend att komplettera eller integrera det funktions-, process- och flödesorienterade synsättet med det mer objektsystem- och databasorienterade synsättet. Detta karaktäriseras av att man söker kartlägga och strukturera data om en tillämpning och söka bestämma vilka samband som råder mellan data av olika slag. Beräkningar, bearbetningar och utmatningar behöver inte alltid specificeras i för-

väg utan 'spontant' anges när informationsbehov uppstår.

Detta senare synsätt gör antagandet att den grundläggande datastrukturen och datasambanden (reglerna) är mer 'stabila' än behovet av olika informationsuttag och bearbetningar. Detta synsätt leder till mer 'flexibla' och anpassningsbara informationssystem och det kännetecknar ju också de sk 4:e och 5:e generationens systemutvecklingsspråk och hjälpmedel. Det som den 'traditionella' IS-ansatsen ändå kan 'exportera' är kunnandet och systematiken att analysera verksamheter, beslutsprocesser och behov av informationsför-sörjning.

KIS — kontorsinformationssystem — är ett område på frammarsch. Avancerade arbetsstationer och lokala nät har medfört att enkla och helt lokala ordbehandlingsrutiner kompletteras med databaser, grafik och möjlighet till inbördes kommunikation. KIS blir mer och mer "informationssystem" i ordets rätta bemärkelse. De blir mer och mer komplexa och kräver avancerade metoder och verktyg för att byggas.

Vad som skiljer KIS från traditionella IS är framför allt decentralisering av data och bearbetning samt ansatserna att integrera 'vanliga' strukturerade data i databaser med fri text, bildinformation och ljud. I tillägg till detta introduceras nya "informationsobjekt" såsom dokument (och versioner av dokument), dossier, brevkorgar, kuvert, osv. Modellering och specificering av ett KIS omfattar allt detta och även modellering av ärendehantering och människa-datordialoger.

Konceptuell Modellering intresserar alltför forskare inom KIS-området. KM-ansatserna måste dock vidareutvecklas för att kunna hantera modellering av kontorstillämpningar. Kunskapsexporten från KIS-området blir sålunda främst kunskap om decentralisering, kommunikation och hantering av information representerad av i detta sammanhang nya former såsom ljud och bild.

Detta avsnitt kan nu sammanfattas. Vi har kortfattat beskrivit forskningsområdena DB — databaser, KR — kunskapsrepresentation, PS — programmeringsspråk, IS — informationssystem och KIS — kontorsinformationssystem. Vi har också diskuterat vilken kun-

skap de olika områdena kan exportera till varandra. Var står då KM – konceptuell modellering – i förhållande till dessa områden. Vi ser KM som en generell metodik att på en problemlnära nivå beskriva tillämpningar, eller rättare sagt objektsystem och kunskap om dessa. Den har viktiga beröringspunkter med samtliga ovanstående områden. KM är inte bunden till en viss typ av objektsystem. Objektsystemet kan vara en databas, ett system av informationsprocesser och flöden, en verksamhet eller en kontorstillämpning. För att vidareutveckla KM-metodik är import av kunskap från de ovan nämnda områdena viktig.

Mycket talar för att framtida, x:te generationens systemutvecklingsspråk kommer att vara språk för specifikation av 'exekverbara' konceptuella modeller. Dessa system kommer samtidigt att besitta egenskaper som vi idag associerar med de skilda områdena AI/KR, PS och KIS. Alltså: Kunskaperna från de olika områdena integreras och leder till nya och effektiva systemutvecklingsverktyg.

3.3 Några aktuella forskningsområden

Som framgått av föregående avsnitt är konceptuell modellering ett område som delvis överlappar eller berör flera andra områden såsom Databasteknologi, Kunskapsbaserade System, Programmeringsspråk, Informationssystem och Kontorsinformationssystem. Området Konceptuell modellering har således många aspekter och en forskare kan betona och intressera sig för olika aspekter KM beroende på sin bakgrund och syftet med forskningsverksamheten. En forskare med bakgrund i verksamhetsanalys kan intressera sig för modelleringens roll och plats i de tidiga faser av systemutvecklingsprocessen. En forskare med bakgrund i databasteknologi kanske intresserar sig för hur konceptuella modeller kan användas som underlag för databaskonstruktion. En forskare med bakgrund i programmeringsspråk kan finna problem som rör formell specifikation och analys av modeller de mest intressanta.

Gemensamt för alla dessa insatser är dock ett – mer eller mindre uttalat – mål att ytterligare höja språknivån (närmare användaren) för specifikation och utveckling av informationssystem och att tillhandahålla hjälpmedel som gör att mer 'intelligenta', mer 'robusta' och mer användaranpassade system kan till-

verkas.

Detta sker genom forskningsinsatser inom en rad besläktade problemområden såsom

- utveckling av mer uttrycksfulla modelleringsspråk som medger modellering av 'komplexa verkligheter' och statiska såväl som dynamiska regler för dessas beteende
- komplettering av modelleringsansatserna med nya former för representation av information såsom bild och ljud
- metodik för formell analys av konceptuella modellers 'korrekthet'
- utveckling av datorstöd för konceptuell modellering
- utveckling av applikationsgeneratorer baserade på KM
- modeller och modellering i distribuerade miljöer
- modelleringsmetodik: olika problem som sammanhänger med tillämpning av KM på större, komplexa praktikfall

I tillägg till dessa problemställningar kan vi notera en rad intressanta bidrag till KM från området AI och då speciellt från sk Kunskapsbaserade system.

I följande avsnitt kommer vi att kortfattat berätta om dessa forskningsområden. Mängden av publikationer (artiklar, konferenspresentationer, institutionsrapporter) som berör problemställningar inom dessa forskningsområden är, minst sagt, enorm. Det är också ofta svårt att direkt peka på publikationer som på ett uttömmande sätt behandlar ett problemområde. Vi avstår därför att ge direkta referenser till artiklar m m. I stället hänvisar vi till avsnitt 3.4 och till kapitel 5 där i detta sammanhang intressanta konferenspublikationer (proceedings) och böcker anges. Ytterligare referenser till artiklar kan erhållas där.

3.3.1 Modelleringspråk

allmänt kan sägas att forskningen här syftar till att förverkliga **100%-principen** dvs att det konceptuella schemat skall vara en fullständig

definition av informationssystemet för en viss tillämpning och omfatta, förutom en definition av informationsstrukturen, även definition av alla statiska och dynamiska regler (se även kapitel 1, där detta diskuteras).

Några **statiska** regler har vi illustrerat i figur 3-3, t ex att "ett projekt x måste ha en och högst en ledare y". Detta måste gälla vid alla tidpunkter (tillstånd) i informationsbasen. Om vi har en informationsbas som för ett visst projekt saknar utsaga om vem som leder det eller om vi har i informationsbasen utsagor att två personer leder ett visst projekt så är informationsbasen vid det ögonblicket ej i överensstämmelse med vad som föreskrivits i schemat.

En **dynamisk** regel föreskriver hur en informationsbas får förändras. Exempel på sådana regler är "En anställds lön får ej minska" och "En gång direktör — alltid direktör". Om informationsbasen i ett visst tillstånd innehåller utsagan "Jan's lön är 10 K" och i nästa tillstånd utsagan "Jan's lön är 9 K", så bryter denna **tillståndsförändring** mot den dynamiska regeln i schemat.

Det finns vid det här laget mer än 50 olika modelleringsspråk (mer eller mindre väl) definierade och publicerade. De flesta av dem är under utveckling och även nya ansatser är ständigt på väg. "På ytan" skiljer sig dessa ansatser ofta inte mycket från varandra. De flesta tillämpar begrepp såsom entitet, attribut, förhållande, m m. Skillnaderna uppträder när man betraktar hur de kan uttrycka olika typer av statiska och dynamiska regler och vilken formalism de tillämpar för detta.

Gemensamt för de olika ansatserna är också någon form av grafisk notation som uttrycker informationsstrukturen — typer av utsagor som får förekomma i informationsbasen. En sådan grafisk beskrivning har vanligen formen av en graf där noderna betecknar klasser av objekt och bågarna mellan noderna betecknar relationer eller associationer. En graf är ett utmärkt hjälpmedel för kommunikation mellan användare och systemspecialister, men den har endast en begränsad uttrycksförmåga. De flesta statiska och dynamiska regler går ej att uttrycka i grafform.

Detta är ett dilemma vid systemutveckling. Man eftersträvar enkelhet — lätthet att förstå — och samtidigt "kraftfullhet" vad avser ut-

trycksförmågan. Vi har dock i större praktiska tillämpningar konstaterat att förståelse om ett objektsystem och dess olika objekt och begrepp kräver ej enbart en strukturbeskrivning utan även en beskrivning av vilka statiska och dynamiska lagar och regler som styr systemets beteende. De språk som hittills publicerats för beskrivning av regler och villkor är baserade på matematik — logisk notation och är således ej särskilt "användarvänliga". En intressant forsknings — utvecklingsuppgift är att förse dessa med en "överbyggnad" som gör dem lättare att förstå för en användare. Detta är angeläget eftersom modellering av en tillämpning utan att ta med regler och villkor ger endast en ofullständig (och kanske rentav fektig) bild av verkligheten eller — rättare sagt — det objektsystem (konkret eller abstrakt) man vill nå konsensus om.

Gemensamt för de flesta nyare ansatser och språk för modellering är också strävan att uttrycka regler och villkor på ett så **deklarativt** (VAD-orienterat) sätt som möjligt. Till exempel, regeln "En anställds lön får ej minska" vill vi uttrycka med en eller ett fåtal "formler" och göra det endast på ett ställe. För att uttrycka en sådan formel måste vi ha en notation för att kunna referera till olika tillstånd i informationsbasen eller kunna referera till värdet på attributet "lön" vid olika tidpunkter. En **procedurell** (HUR-orienterad) definition av denna regel skulle motsvara ett avsnitt i ett program som kontrollerar om det nya värdet för variabeln "lön" är mindre än det gamla värdet. Det finns för- och nackdelar med bägge uttrycksformerna. Allmänt kan sägas att deklarativa (dvs VAD-orienterade) specifikationer ger bättre möjlighet till analys av schemabeskrivningens korrekthet (motsägelsefrihet) och ger även informationssystemet en viss möjlighet att "resonera" om sin konceptuella modell. Nackdelen är (ännu så länge) mindre god effektivitet vid implementering och svårigheten att deklarativt uttrycka viss typ av kunskap om objektsystemet, t ex hur ett "optimalt" värde för viss variabel beräknas. Framtiden tycks ligga i ansatser och språk där deklarativa och procedurella definitioner kan samexistera.

Konceptuell modellering har traditionellt tillämpats främst vid administrativa system och "hårt formaliserade, formaterade och strukturerade" data. Data har då representerats i form av teckensträngar enligt olika format. Vissa ansatser talar om dessa data-objekt

som 'lexikaliska' objekt. Genom att informationsteknologin breder ut sig till en rad nya tillämpningsområden såsom Kontorsinformationssystem, CAD/CAM, Geografiska databaser & system, m m uppstår krav att konceptuellt kunna hantera andra former för representation av information såsom fri text, ljud och bild. Forskningsarbete pågår (och vissa resultat finns redan) när det gäller att i samma informationsbas integrera information representerad såväl strukturerat på traditionellt sätt som representerad med fri text, bilder och ljudsekvenser.

Som ett exempel kan vi tänka oss att information om en anställd i en personaldatabas kan omfatta såväl 'vanliga formaterade data', såsom personnummer, namn och adress, som vederbörandes biografi (fri text), foto och röstsignatur. Modelleringspråk som kan definiera konceptuella schemata som omfattar även dessa nyare former för representation av information är under utveckling. Detta öppnar givetvis helt nya möjligheter för informationshantering men radar samtidigt upp en rad nya, intressanta men svåra, forskningsproblem. Till exempel, hur definiera operationer över bildinformation? Vi kanske vill söka information om alla anställda som "är mörkhåriga och har mustasch" eller alla anställda som "liknar Alvar Persson". Hur skall vi i detta sammanhang definiera operationen "liknar"?

Många av dagens databaser och informationssystem lider av problemet att de betraktar den aktuella tillämpningen i ett smalt **tidsperspektiv**. Databasen innehåller endast för ögonblicket aktuella objekt och dessas senaste attributvärden och relationer (såvida inte speciella arrangemang införts). Vid en förändring i objektsystemet gör man, genom en transaktion, en uppdatering av motsvarande attributvärde och "glömmer bort" det gamla värdet. Konsekvensen av detta är att man inte kan ställa frågor till informationsbasen om **historiska** tillstånd — man kan inte analysera (eller resonera om) objektsystemet i ett utsträckt tidsperspektiv. Vill man t ex, analysera den anställdes Peters löneutveckling genom att betrakta hans lön vid olika (historiska) tidpunkter så måste man redan vid systemutveckling förutse detta behov och införa därför lämpliga datastrukturer. Vissa nyare ansatser och språk för modellering tar utgångspunkten att betrakta i modellen **all information om objektsystemet i ett utsträckt tidsperspektiv**.

Detta leder till databaser som "aldrig glömmes", där begreppet "uppdatering" ej förekommer och som kan svara på frågor om godtyckliga historiska tillstånd i objektsystemet. Forskningen på detta område undersöker dels lämpliga språkformer för att definiera **konceptuella schemat med temporal dimension** dels hur en sådan **temporal — eller historisk — databas** effektivt skulle kunna implementeras. Ny minnesteknologi (t ex optiska minnen) kommer att göra dessa "historiska databaser" till en praktisk realitet inom 5 år.

Språk för konceptuell modellering förbättras och utvecklas även genom att modellering börjar tillämpas för andra objektsystem än de som traditionellt betraktas i dagens databasystem. Konceptuell modellering är ju en mycket generell beskrivningsmetodik och de "objektsystem" som man beskriver kan vara, exempelvis, organisationsstrukturer, målstrukturer, beslutsstrukturer, problemsamband, flöden av information och material, hantering av dokument, dossier och ärenden i kontorsmiljöer, produktspecifikationer i CAD/CAM-system, osv. Alla dessa tillämpningar har mycket gemensamt vad avser grundläggande begrepp för modellering. Men var och en uppvisar också vissa speciella önskemål och krav vad avser modelleringsspråkens begreppsapparat och uttrycksförmåga.

Vart är vi då på väg? Kommer vi att om några år ha ett avancerat, allmänt utbrett och accepterat språk för modellering? Allting tyder på att nya generationers (5:e och 6:e) systemutvecklingshjälpmedel och verktyg kommer att baseras på avancerade och objektorienterade modelleringsspråk och att effektiva "maskiner" kommer att finnas som direkt kan "ta" en systembeskrivning i dessa språk och generera exekverbara modeller — produktionssystem. Det är dock mycket osannolikt att något speciellt språk kommer att accepteras som standard (se även kapitel 4). I likhet med programmeringsområdet, där en stor mängd av språk florerar, kommer vi att möta en stor mängd nya modelleringsspråk och tillhörande verktyg. Det kommer att vara svårt att för en viss tillämpning avgöra för- och nackdelarna hos de olika språken — åtminstone för en icke-specialist på området. Inom ISO finns en arbetsgrupp (TC97/SC5/WG3) [7] som håller på att ta fram en kravspecifikation för språk för konceptuell modellering. Kravspecifikationen är ganska allmänt skriven, lämnar ett stort

tolkningsutrymme och det återstår att se vilken effekt den kommer att ha på språkutvecklingen i detta område.

3.3.2. Metoder för analys av modeller

Detta forskningsområde omfattar insatser för att utveckla metoder som förbättrar våra möjligheter att analysera modellers "godhet" och "korrekthet" i olika avseenden. Vi kan här skilja på två problemområden.

1. Hur kontrollera och bevisa att den modell man konstruerat verkligen motsvarar den bild av "verkligheten" eller objektsystemet som systemets ägare eller användare har? Är modellen "bra" för sitt syfte? Vi kan kalla detta för **validitetsproblemet**.
2. Är modellen "korrekt"? Detta område omfattar allt ifrån enkla syntaktiska problem till komplicerade semantiska analyser.

Validitetsproblemet kan knappast tacklas med formella, matematiska metoder. Det är tänkbart att formella metoder kan utvecklas som analyserar en föreslagen modell och föreslår tänkbara strukturella förändringar för att göra modellen mer "homogen". Det stora problemet är dock att kontrollera om den "konceptualisering" av ett objektsystem som användarna och modelleringsspecialister skisserat är **valid**, dvs motsvarar användarnas bild av systemet. För att komma någon vart på detta problem måste metoder utvecklas som ökar användarnas **förståelse** av vad modellen står för och representerar. Validitetsproblem såsom "Är de angivna reglerna riktiga?" och "Saknas det regler i vår specifikation av en modell? Har vi glömt något?" är teoretiskt/formellt inte möjliga att tackla. I detta område finns även semantiska problem. En konceptuell modell är ju resultatet av flera (oftast många) personers sammanjämkade uppfattning om en verklighet. Om personen A använder begreppet "vårdtillfälle" hur kan vi avgöra om det verkligen motsvarar samma sak som begreppet "vårdtillfälle" hos person B?

Vägen mot bättre validering av modeller tycks således ligga i att utveckla metoder som förbättrar kommunikation och förståelse kring modeller och vad de representerar. Möjligheten att skapa "konceptuella skisser" — exe-

kverbara konceptuella modeller — bör sannolikt bidra till en bättre förståelse av den modell man specificerat.

Korrekthetsproblemet avser modellens syntaktiska korrekthet och kontroll av att den inte innehåller logiska motsägelser. I princip skulle detta kunna ske helt mekaniskt. Att kontrollera en modell för syntaxfel är enkelt. Att, däremot, formellt kontrollera huruvida de regler och begrepp som en modells schemadefinition omfattar är inbördes fria från motsägelser är, teoretiskt sett, ett omöjligt problem. Vi kan inte garantera att en sådan analys-algoritm "terminerar", dvs att vi någonsin får ett svar om modellen är **konsistent** eller ej. Och även om vi fick ett svar uppstår genast nästa fråga "vad är det som är icke-konsistent?". Om vår modell består av, säg, 10 000 regler och villkor är det till föga hjälp att endast få reda på att den innehåller motsägelser, dvs att den är **icke-konsistent**. Forskningen på detta område måste således i första hand utveckla praktiska, tumregelsbaserade (heuristiska) metoder att dels bestämma om modellen är icke-konsistent dels ge anvisningar om var man skall söka felet.

Forskningen har tyvärr inte kommit särskilt långt varken på validitetsproblemet eller på korrekthetsproblemet. Det är för närvarande svårt att sja om utvecklingen på detta analysområde. Det finns tecken som tyder på att framtiden kanske ligger i stora, globala modeller som är gemensamma för alla användare och som man förutsätter vara valida och konsistenta i alla avseenden. En tes som förs fram av vissa forskare är att stora, integrerade modeller som är konsistenta knappast är (praktiskt) realistiska. I stället förordar de löst koplade, decentraliserade och öppna system som kan kommunicera med varandra och utbyta såväl data som 'kunskap' om varandra. Varje litet system måste givetvis bygga på en konsistent konceptuell modell men systemen är av sådan begränsad storleksordning att korrektheten (lokalt sett) inte utgör ett problem. Globalt sett finns däremot ingen konsistens i detta "system av system", dvs den konceptuella modellen hos system A kan strida emot system B:s konceptuella modell. Vad A och B måste komma överens om är meningen — **semantiken** — hos de meddelanden som de kan utbyta. För att ett sådant system av system skall kunna utvecklas måste det också finnas metoder för hur delsystemen kan "lära sig"

betydelsen av nya typer av meddelanden som införs i systemet av nytillkomna delsystem eller genom att vissa delsystem förändras och utvecklas. Detta synsätt, ännu så länge på ett mycket tidigt forskningsstadium, närmar sig människors sätt att agera och kommunicera där var och en har en egen "konceptuell modell av verkligheten" och där man vid kommunikation bara delvis "förstår" varandra.

3.3.3 Datorstöd

Konceptuella modeller av realistisk storleksordning omfattar definition och beskrivning av hundratals entitetstyper (objekttyper) samt tusentals attribut och olika slag av regler och villkor. Utveckling av lämpliga datorstöd är ett angeläget forskningsområde. Man behöver datorstöd dels för att dokumentera, hålla reda på och söka bland tusentals olika definitioner dels behöver man stöd för att utföra olika analyser och kontroller (se föregående avsnitt) av det konceptuella schemat i olika utvecklingsstadier. För att dokumentera enklare informationsstrukturer (entiteter, attribut) kan man ofta göra bruk av befintliga sk 'data dictionaries'. Dessa enkla verktyg kan dock ej dokumentera mer komplicerade samband, regler och villkor och ej heller har de någon direkt förmåga att kontrollera och analysera det som dokumenterats. Nya typer av 'dictionaries' med starkt förbättrade egenskaper krävs därför om de skall vara till nytta vid konceptuell modellering.

En önskvärd egenskap hos datorstöd för KM är givetvis att dessa har förmåga att arbeta med grafiska representationer av modeller. Ett bra datorstöd skall då inte bara vara ett "ritverktyg" utan även kunna "förstå" en konceptuell graf och lagra dess logiska innehåll för vidare bearbetning och analys. Ett sådant grafiskt orienterat verktyg, RAMATIC, håller på att utvecklas inom SISU:s programområde 3.

En annan önskvärd egenskap hos datorstöd för KM är att kunna tillsammans med den formella och grafiska beskrivningen även kunna hantera skisser, informella beskrivningar, beslutsprotokoll från arbetsmöten och medge kommunikation (brev, diskussioner) mellan medlemmarna i en modelleringsgrupp.

Utveckling och forskning kring lämpliga datorstöd för KM befinner sig bara i ett mycket tidigt skede. Ett antal ansatser och system finns

men ännu har väl ingen av dem fått någon mer utbredd tillämpning. Dels är dessa verktyg relativt primitiva dels saknar de vanligen lämpliga funktioner för analys, hantering av informell information och för kommunikation. En annan anledning till den relativt begränsade användningen kan vara att själva KM-tekniken befinner sig — praktiskt sett — i en introduktionsfas. Man har ringa nytta av ett verktyg om man ej behärskar själva KM-tekniken och metodik i samband med KM. En annan bromsande faktor vid utveckling av datorstöd har varit den stora mängden av mer eller mindre olika modelleringspråk. "Marknaden" för ett verktyg som bara hanterar **ett enda** språk har varit begränsad.

Till datorstöd för konceptuell modellering bör man också räkna de system som i princip arbetar som 'exekverbara konceptuella modeller'. I sådana system definierar man först ett konceptuellt schema som anger vilka typer av utsagor om ett objektsystem som systemets informationsbas får innehålla. I schemat anger man också statiska såväl som dynamiska regler som reglerar vad som får sägas om objektsystemet och hur informationsbasen får förändras. Ett antal sådana (mer eller mindre avancerade) system finns i experimentell tillämpning. Vissa av dem har även en grafisk användar" interface" mot schemat såväl som mot informationsbasen.

De flesta systemen omfattar även frågespråk mot informationsbasen, möjlighet att definiera formulär för in/utmatning av data samt meny-styrd användarinteraktion. Vi betraktar dem som datorstöd för modellering därför att de ger möjlighet att snabbt bygga upp ett konceptuellt schema och en tillhörande informationsbas ("rapid prototyping"). Detta system kan sedan användas för olika experiment och test av regler och villkor. För "prototyping" av konceptuella modeller har även logikprogrammeringsspråket PROLOG i flera fall använts.

3.3.4 Metodik vid konceptuell modellering

Konceptuell modellering är en relativt ny teknik och en rad metodproblem väntar på sin lösning eller åtminstone på en mer ingående analys.

Ett övergripande problem är hur KM kommer in i den traditionella systemutvecklingspro-

cessen och vilken roll den spelar i dess olika etapper. Kanske innebär KM att en ny modell för systemutveckling bör utvecklas? Skall en KM utvecklas gradvis och parallellt med problemanalys, verksamhetsanalys och informationsbehovsanalys? Hur skall då den konceptuella modellen relateras till de modeller som utvecklas i dessa parallella etapper? Skall man böja med att göra en global konceptuell modell eller skall man först noga specificera informationskrav och bygga en modell i enlighet med dessa krav? Som synes finns det många frågeställningar om vilka olika idéer och uppfattningar råder i praktikfältet såväl som bland forskare. Det är också mycket tveksamt om en bra metodik går att teoretiskt "forska fram" i detta sammanhang. För det behövs det ett mycket aktivt engagemang från praktikfältet. Givetvis finns det ingen "sanning" i detta metodavseende — en metod blir inte bra enbart på sina teoretiska grunder. Den behöver också kunna fungera och accepteras i praktiska miljöer.

Ett stort metodproblem vid konceptuell modellering är hur arbetet på en stor modell (som i slutskedet förväntas innehålla 10 000-tals 'element', dvs entitetstyper, attribut, regler) kan uppdelas på flera parallellt arbetande modelleringsansvariga. Ett tillhörande problem, som kräver fortsatta forskningsinsatser, är metodik för hur lokalt utvecklade konceptuella schemata (s k 'local views') kan integreras till ett gemensamt globalt schema. (det s k 'view-integration'-problemet).

Övriga metodproblem rör utveckling av arbetsprinciper för en 'dataadministratör' eller en 'informationsresursadministratör'. Detta är givetvis ett praktiskt problem men dess lösning bör bli en del på den forskning om KM som nu pågår och som kan ge en antydning om hur framtida "information dictionaries" kommer att vara beskaffade.

Konceptuell modellering innebär i stor utsträckning arbete med människor och deras uppfattning om en viss verklighet eller ett visst objektsystem. Bra språk och metoder för KM bör förstärka deltagarnas förståelse av ett visst system. Här finns alltså även plats för kognitionspsykologisk forskning kring de beskrivningstekniker, bildspråk och verktyg som tillämpas vid KM.

Ett ofta noterat problem vid KM i praktiken är

att de som utformar konceptuella schemata råkar i **konceptualiseringsproblem**. Skall man betrakta ett visst fenomen i objektsystemet som en entitet eller ett attribut eller ett förhållande eller ... ? Även här är det inte sannolikt att forskning kan lämna nämnvärd hjälp. Dock vore det ur praktisk synvinkel önskvärt om några riktlinjer eller tumregler kunde utarbetas som minskade risker för dåliga konceptualiseringar.

3.3.5. AI-importen

På senare år har samarbete och korsbefrukning skett mellan områdena **databaser** och området **AI**, speciellt då dess delområden **kunskapsrepresentation** och **kunskapsbaserade system**. Vi planerar att ägna ett framtida nummer av SISU-analys åt kunskapsbaserade system och dess betydelse för systemutvecklingsområdet. Därför skall vi här endast kortfattat ange några av de 'importartiklar' som för närvarande utövar ett icke ringa inflytande på forskningen kring konceptuell modellering.

Olika metoder för **kunskapsrepresentation**, de må vara **logikbaserade**, **proceduriella**, baserade på s k **semantiska nät** eller **frame**-baserade, påverkar konceptuell modellering i riktning att kunna representera och hantera mer avancerad kunskap — och ej enbart data — om ett visst objektsystem. Förlängningen av detta är att konceptuella modeller kommer att (i viss utsträckning) kunna "resonera" om det aktuella objektsystemet. Den skarpa distinktionen mellan schema och informationsbas kommer i viss mån att suddas ut.

Resonerande system som bygger på icke-standardlogik (man resonerar t ex om vad som är möjligt eller sannolikt) kan, även de, få intressanta framtida tillämpningar inom KM-området.

En användares gränssnitt mot en informationsbas kan göras mer "intelligent" genom att tillämpa resultat från området **förståelse och bearbetning av naturligt språk**. Kanske kan forskningen inom detta område även lämna bidrag till utformning av datorstöd som hjälper en modellkonstruktör vid konceptualisering av ett visst objektsystem.

Intressant forskningsarbete pågår även för att koppla resonerande system till befintliga, stora databaser. På detta sätt utvecklas kunskap

och teknik hur man effektivt kan utnyttja befintliga, stora dataresurser och förse dem med en 'intelligent front-end' mot användaren.

Även om många problem återstår att lösa tycks utvecklingen vara på väg från DBMS (Data Base Management Systems) till KBMS — Knowledge Base Management Systems. Temat har varit uppe på flera konferenser under det senaste året. Det har dock varit svårt att få forskarna eniga om hur ett KBMS bör definieras och vilka krav som bör ställas på det.

3.4 Att följa forskning och utveckling

Att följa forskning och utveckling inom området konceptuell modellering innebär, liksom för andra forskningsområden, att man aktivt deltar i sådana samfund, arbetsgrupper och konferenser som har konceptuell modellering på sitt program och att man håller sig ajour med det som publiceras i ämnet i form av artiklar, rapporter och böcker. I kapitel 5 ger vi förslag till några läroböcker och anger dessas lämplighet för läsare med olika bakgrund och erfarenhet om modellering. I detta avsnitt begränsar vi oss till den mer forskningsorienterade verksamheten och ger tips om olika samfund, intresse- och arbetsgrupper samt publikationer vars verksamhet resp innehåll anknyter till konceptuell modellering.

3.4.1. IFIP — International Federation for Information Processing

Inom IFIP sker utvecklings- och forskningsarbete främst inom dess olika tekniska kommittéer (TC:s) och dessas arbetsgrupper (WG:s). Arbetsgrupperna anordnar regelbundet även så kallade arbetskonferenser. Två av IFIP:s arbetsgrupper är av speciellt intresse för konceptuell modellering, WG2.6 och WG8.1.

IFIP WG2.6 DATA BASES

IFIP Technical Committee 2 on Programming har en arbetsgrupp **WG2.6 on Data Bases**. Denna arbetsgrupp har en lång tradition inom konceptuell modellering och dess arbete kan sägas fokusera på språk, formalismer och metoder för utveckling och definition av konceptuella schemata. Ursprungligen bestod gruppen mest av databasfolk men på senare tid har flera forskare från AI — Kunskapsrepresentation anslutit sig till arbetet.

WG2.6 har under årens lopp svarat för ett flertal arbetskonferenser. 1973 ägde den, vid det här laget klassiska arbetskonferensen **DATA BASE MANAGEMENT** [9] rum på ön Korsika. Flera banbrytande idéer presenterades här och lade en viktig grund för fortsatt utveckling på området (t ex **Abrial's** Binära modellansats, **Sundgren's** Infologiska ansats, m fl).

Nästa milstolpe i WG2.6:s serie var arbetskonferensen **MODELLING IN DATA BASE MANAGEMENT SYSTEMS** [11] år 1976. Medan 1973 års konferens var mer traditionellt databasorienterad var 1976 års konferens betydligt mer modelleringsorienterad. Ett flertal "högnivå" modelleringsspråk presenterades här och bland författarna förekommer namn som **Nijssen, Senko, Bracchi, Falkenberg, Biller, Weber** m fl. Såväl utvidgningar av relationsmodellen som förslag till olika så kallade "semantiska datamodeller" (det som vi kallar för konceptuella modeller) presenterades och en, vid den tiden, sedvanlig debatt om dessa ansatsers för- och nackdelar ägde rum.

Nästa WG2.6-arbetskonferens hade titeln **ARCHITECTURE AND MODELS IN DATA BASE MANAGEMENT SYSTEMS** [12] och den ägde rum 1977 i Nissa. Konferensen fördjupade diskussionen på vissa punkter i förhållande till 1976 års konferens, bl a med avseende på modellering i ett utsträckt tidsperspektiv.

Det dröjde ända till januari 1985 innan WG2.6 arrangerade sin nästa arbetskonferens **DATA-BASE SEMANTICS: DS-1** [10] i Hasselt, Belgien. De artiklar som presenterades täckte ett brett spektrum av problem (se föregående avsnitt) såsom t ex **Modeling static and Dynamic Aspects of Information Systems, Integrity Constraints, Conceptual Models in Prolog, Denotational Semantics for Administrative Databases, English Queries to Databases, Temporal Aspects of Databases**, m m.

Arbetsgruppen 2.6:s nästa arbetskonferens, DS-2, kommer att gå av stapeln i november 1986 i Algarve, Portugal. Konferensens rubrik blir **KNOWLEDGE AND DATA** vilket klart markerar den intressegemenskap som håller på att utvecklas mellan DB- och AI-områdena.

Intresserade av WG2.6:s verksamhet kan vända sig till Janis Bubenko (SISU), som är medlem i WG2.6, eller till WG2.6:s ordförande Pro-

fessor **Robert Meersman**, Department WNIF, Limburgs Universitair Centrum, Universitaire Campus, B-3610, Diepenbeek, Belgien.

IFIP WG8.1 DESIGN AND EVALUATION OF INFORMATION SYSTEMS

WG8.1 hör under IFIP:s tekniska kommitté TC8 och dess intresseområde är **metoder och principer för analys och konstruktion av datorbaserade informationssystem**. Som vi tidigare påpekat, utgör konceptuell modellering numera en viktig komponent vid informationssystemspecifikation och utveckling. Flertalet av medlemmarna i WG8. 1 är intressenter även i WG2.6. På senare år har WG8.1 attraherat även forskare inom områdena **Software Engineering** och **AI – Kunskapsbaserade System**.

Bl a följande arbetskonferenser och tillhörande publikationer, arrangerade av WG8.1 är av intresse med avseende på konceptuell modellering.

- **FORMAL MODELS AND PRACTICAL TOOLS FOR INFORMATION SYSTEM DESIGN**, Oxford, 1979 [15] , innehöll ett flertal föredrag om eller med anknytning till konceptuell modellering.
- **AUTOMATED TOOLS FOR INFORMATION SYSTEMS DESIGN**, New Orleans, 1982 [16] var orienterad mot datorstöd vid programutveckling och systemutveckling. Av speciellt intresse för konceptuell modellering var presentation av TAXIS-systemet (Borgida et al) som kan ses som en av de första ansatserna att skapa en exekverbar konceptuell modell.
- **INFORMATION SYSTEMS DESIGN METHODOLOGIES: A COMPARATIVE REVIEW**, Noordwijkerhout, 1982 [13] , den sk CRIS-1 konferensen, tillhör WG8. 1:s mer kända konferenser. 13 olika "systembeskrivningsmetoder" presenteras genom att alla har fått i uppdrag att "lösa" samma praktikfall. De flesta av de presenterade metoderna inrymmer "någon form av" konceptuell modelleringsaktivitet. Konferensdokumentationen bör vara av intresse för dem som vill studera hur olika systemutvecklingsmetoder har integrerat (mer eller mindre) konceptuell modellering i sina "metodked-

lor". Denna konferens följs året efter av en "metodjämförande" konferens **INFORMATION SYSTEM DESIGN METHODOLOGIES: A FEATURE ANALYSIS**, i York, England.

- Den senaste i raden av WG8.1-konferenser har titeln TFAIS- 85. **THEORETICAL AND FORMAL ASPECTS OF INFORMATION SYSTEMS**. Den avhölls 1985 i Sitges, Katalonien. Praktiskt taget samtliga föredrag behandlade olika ansatser och aspekter på konceptuell modellering. Bland deltagarna var AI-inslaget märkbart.

Nästa arbetskonferens i WG8.1-serien har titeln **ENVIRONMENTS TO SUPPORT INFORMATION SYSTEM DESIGN METHODOLOGIES**. Den kommer att hållas i Bretton Woods, NH, USA, 4-6 september och sålunda behandla datorbaserade verktyg och "utvecklingsmiljöer" för systemutveckling.

Intresserade av WG8.1:s verksamhet kan vända sig till dess ordförande Prof **Arne Sölvberg**, NTH, N-7034, Trondheim, Norge eller till gruppens sekreterare Janis Bubenko jr, SISU.

3.4.2. ACM – Association for Computing Machinery

De ACM-tidskrifter som ofta innehåller stoff av intresse och relevans för konceptuell modellering är, i första hand

- **ACM TODS – Transactions on Database Systems**
- **ACM TOIS – Transactions on Office Information Systems**

Dessa utkommer kvartalsvis. ACM har dessutom en mängd sk SIG:s — speciella intressegrupper. Dessa utger, mer eller mindre regelbundet, nyhets-bulletiner. De intressegrupper som en modelleringsintresserad i första hand bör bevaka är SIGMOD (SIG on **Management of Data** som utger **SIGMOD RECORD**, SIGOA (**Office Automation**) samt SIGART (**Artificial Intelligence**) som ger ut **SIGART Newsletter**. Man bör även bevaka **ACM Computing Surveys** som ibland innehåller översiktsartiklar av intresse för konceptuell modellering. De ACM-organiserade, årliga konferenser som är av direkt intresse för modellering är SIGMOD-

konferenser (se t ex [18, 4]) och SIGOA-konferenser (se t ex [5]). Även andra ACM-konferenser kan vara av intresse i modelleringssammanhang. Ett kalendarium som listar såväl ACM:s arrangemang som andra internationella konferenser återfinns i ACM:s månadstidning **Communications of the ACM**.

3.4.3. IEEE — Institute of Electrical and Electronics Engineers

IEEE's "datasektion" — **Computer Society** — har nyligen startat en teknisk kommitté för **Database Engineering**. Man kan förvänta att dess tidskrift, som planeras utkomma kvartalsvis, kommer att innehålla för konceptuell modellering relevant material.

Det händer emellertid att även IEEE's övriga datatidskrifter såsom **IEEE Computer**, **IEEE Software** och **IEEE Transactions on Software Engineering** ibland (dock mera sällan) innehåller material av intresse och relevans för KM. T ex oktobernumret, 1983, av **IEEE COMPUTER** kan rekommenderas för dem som önskar en bred introduktion i området **Knowledge Representation**.

3.4.4. VLDB — Very Large Data Bases

En internationell samling av forskare i databasområdet står bakom **VLDB Endowment** (VLDB-stiftelsen) vars syfte är att arrangera årliga VLDB-konferenser. Den första VLDB-konferensen hölls 1975 i USA. Under de år som gått har VLDB besökt praktiskt taget samtliga världsdelar. I år, 1985, står Stockholm och SYSLAB (och indirekt SISU) som värd för den 11:e internationella VLDB-konferensen. Samtliga år som VLDB hållits har konceptuell modellering (eller "semantic data modeling", "data modeling" eller "information modeling" som det ofta kallas) varit en tung punkt i ett VLDB-program. T ex den kända ER-modellen (Entity — Relationship) introducerades på VLDB-1975. Årets VLDB innehåller en hel del material relaterat till konceptuell modellering. Speciellt märkbart är inslaget av kunskapsbaserade system — **Knowledge Database Management**.

Intresserade av de gångna årens konferenspublikationer kan vända sig till VLDB-stiftelsens ordförande Professor Alfonso F. Cardenas, Department of Computer Science, 3731 Boelter Hall, UCLA, Los Angeles, CA

90024, USA eller vice ordförande Janis Bubenko jr, SISU. Intresserade av årets Stockholms- VLDB kan kontakta organisationskommitténs ordförande Lars Söderlund, SISU.

3.4.5. Korsbefruktning AI och Databaser

I avsnitt 3.2 berättade vi om vissa 'närmanden' som görs mellan områdena DB, AI och PS (programmeringsspråk). Dessa försök att närma de olika områdena till varandra och ta vara på varandras erfarenheter har manifesterat sig i ett antal arbetskonferenser.

En av de första var **Workshop on Data Abstraction, Databases and Conceptual Modeling**, Pingree Park, Colorado, 1980. Konferensdokumentationen [3] består huvudsakligen av diskussionsinlägg från företrädare för olika forskningsinriktningar.

Efterföljaren till Pingree Park konferensen är Intervale-konferensen (New Hampshire, USA, 1982) som har resulterat i den välkända boken **On Conceptual Modeling — Perspectives from Artificial Intelligence, Databases and Programming Languages** [1]. Tonvikten vid denna konferens var lagd på översiktliga forskningsrapporter som speglade viktiga utvecklingstendenser i de olika områdena och analyserade möjliga beröringspunkter mellan dessa (se även recensionen i kapitel 5).

Hösten 1984 arrangerades ett uppmärksammat och välbesökt symposium **EXPERT Database Systems** [8] på Kiawah Island, South Carolina, USA. Syftet var att söka korsbefrukta områdena **Database Management** och **Expertsystem** för att främja forskning och utveckling kring mer "intelligenta" databashanteringssystem. I symposiet deltog forskare med starkt varierande bakgrund. Vi återger här konferensens huvudtemata.

- Constraint and Rule Management
- Expert Database Systems
- Intelligent Database Access
- Knowledge Base Management
- Knowledge Representation
- Logic Programming and Databases
- Natural Language
- Requirements for Expert Database Systems
- Reasoning
- Semantic Query Optimization
- Specification Methodologies

Konferensens relevans för 'mer avancerade former' av konceptuell modellering förefaller klar.

En arbetskonferens, med liknande syfte som Kiawah-symposiet, och med titeln **Large Scale Knowledge Base and Reasoning Systems [2]**, arrangerades i februari, 1985, i Islamorada, Florida, USA. Syftet var även här att samla forskare med bakgrund i databaser, AI, logik, m m och med intresse att utveckla "storskalig kunskapshandlingsteknik". Konferensbidragen här var huvudsakligen av typen diskussionsinlägg (position papers). Islamorada-konferensen följs i juni, 1985, av en arbetskonferens på Kreta med i stort sett motsvarande innehåll men är då främst avsedd för europeiska deltagare.

3.4.6. Övriga källor och aktiviteter

Några ytterligare källor bör omnämnas. Tidsskriften **INFORMATION SYSTEMS** (utgivare: Pergamon Press, utkommer kvartalsvis) innehåller ofta bidrag som är relevanta för detta område. Samma utgivare publicerar kvartalsvis även **DECISION SUPPORT SYSTEMS** som tangerar området "intelligenta databaser" och därmed KM.

I USA arrangeras regelbundet de s k **International Conferences on Entity Relationship Approach**. Även om ER-modellens upphovsman står bakom dessa konferenser är det inte bara ER-typen av modelleringsansatser som tas upp på dessa.

I Norden finns relativt många aktiva forskare med intresse för databaser, AI och konceptuell modellering. Under de senaste 4-5 åren har det för de flesta blivit en tradition att på vårkanten samlas till ett 3 dagars seminarium i det gästfria Tammerfors. Symposieserien kallas **Scandinavian Research Seminars on Information Modeling and Database Management**. Mer information om dessa seminarier kan fås hos deras värd och arrangör **Hannu Kangassalo**, Computer Science Department, University of Tampere, Tampere, Finland.

Var forskas det då om konceptuell modellering i Sverige? **SYSLAB** i Stockholm och Göteborg har en ganska lång tradition på detta område. Inom laboratoriet finns såväl forskare som ser på KM ur ett "databasperspektiv" som forskare

re som ser på KM med utgångspunkt i AI och Kunskapsrepresentation. Vissa forskare har även anlagt ett "verksamhetsperspektiv" på KM. Laboratorierna SSRC — **Software Systems Research Center** vid Högskolan i Linköping och UPMAIL — **Uppsala Programming Methodology and Artificial Intelligence Laboratory** kan också sägas vara intresserade av KM men då ur en logikbaserad och kunskapsrepresentativ utgångspunkt. HIPLAB — **Heuristic Information Processing Laboratory** vid Lunds Universitet har en liknande arbetsinriktning.

Samtliga laboratorier ovan har skriftserier ur vilka rapporter på begäran kan erhållas.

3.5 Slutkommentar

Som ovan framgått är mängden av olika publikationer, konferenser och aktiviteter av potentiell relevans för konceptuell modellering mycket stort. Ändå gör vi inte anspråk på att ha täckt in det relevanta utbudet mer än till viss del.

Forskningen inom området konceptuell modellering är inte välavgränsad och utvecklingen löper samtidigt i olika riktningar. Olika begreppsapparater och formalismer prövas i syfte att göra ansatserna mer väldefinierade och "kraftfulla". För en "nybörjare" kan det vara svårt att orientera sig i mängden av olika begreppsramar, språk och metoder och skilja på substantiella bidrag och "hjul som uppfunnits på nytt". Detta är dock typiskt för ett ämnesområde under snabb utveckling och det uppvisar många likheter med utvecklingen på området programmeringsspråk eller området 4.e generationens systemutvecklingsverktyg. KM har dessutom problemet att den är relaterad till så många andra discipliner och kunskapsområden.

Det kommer säkert att dröja 5-10 år innan mer avancerade former av konceptuell modellering kommer att få en bredare tillämpning i praktikfältet. Det är dock redan nu konstaterat att enklare, partiella (dvs ej fullständiga) modeller uppfattas som värdefulla, praktiska hjälpmedel i olika (främst de tidiga) etapper vid systemutveckling (se även kapitel 2). Erfarenheter från dessa tillämpningar utgör en viktig kunskapskälla för fortsatt forskning kring såväl modelleringsspråk som modelleringsme-

todik. Omvänt synes det viktigt att professionella i praktikfältet redan nu orienterar sig om pågående forskning för att bättre kunna förbereda sina organisationer för kommande informationsteknologiska nyheter och förändringar.

4. STANDARDISERING

Björn Nilsson, FRI

4.1 Varför det som sker sker.

Det existerar flera alternativa strukturer inom vilka standardiseringens olika aspekter med anknytning till databasområdet kan presenteras. Ett vanligt sätt är att utgå från de olika standardiseringsorganens aktiviteter och interna organisation, ett annat att utgå från områdesvisa aktivitetsbeskrivningar, ett tredje att utgå från producerade standards. Varför har vi då inte valt någon av dessa enkla vägar?

Ett problem för dem som söker meningsfulla sammanhang i standardiseringens eller databasvärldens moras är att nycklarna, de bakomliggande motiven till olika företeelser, ofta är fördolda. Orsaker till olika fenomen penetreras mycket sällan i officiella protokoll och publikationer. Syftet med denna enkla exposé är att exponera vissa samband, som haft eller har en styrande effekt på arbetet inom databasområdet. Vi kommer att begränsa oss i huvudsak till arkitektur- och schemafrågor samt i viss mån ta upp ett antal frågor kring informations- och datakataloger i allmänhet.

4.2 Några observationer.

Dagens standarder och normer inom databasområdet i vid mening är ett hopkok med undermåligt inre sammanhang och med bristfällig täckning. Organisationerna inom standardiseringsområdet både internationellt och nationellt försöker med varierande framgång få bättre grepp över utvecklingen. Även komplexa organisationer typ ISO, International Standards Organisation, med stor inneboende tröghet tvingas organisera om i snabb takt.

Målen bakom standarders framkomst är av helt olika karaktär i olika sammanhang och ofta motstridiga. Standarder och normer presenteras i fel tidsordning och läsningar uppkommer mellan standarder i onödan genom blandning av olika aspekter inom samma standard, vilka rätteligen borde uppträda i olika, oberoende standarder.

Standardiseringsarbetet håller på att ändra karaktär. Från att ha varit en ex post facto aktivitet som värderar marknadens existerande,

etablerade kandidater förvandlas verksamheten till att snarare bli styrande för marknaden och att ligga nära forskningsfronten.

Inom parentes kan nämnas att Sverige i förhållande till landets storlek är väl representerat inom området informationsteknologi inom framför allt ISO men även CCITT, teleunionernas standardiseringsorgan. Inom förvaltningen leder SIS starka länkar till statskontoret till en god koordinering av förvaltningens standardiseringssträvanden. Det som enligt kända byråkratiska spelregler bort leda till en kraftig tidsmässig eftersläpning förbyts ibland i ett framsynt arbete kring s k tekniska normer.

Vart leder nu förändringen av standardiseringsverksamhetens villkor — och framför allt, inom vilka områden bör vi satsa våra begränsade resurser?

4.3 Systemarkitekturer i omvandling.

En satsning av resurser kräver en god referensram till vilken resurserna kan allokeras och gentemot vilken effekter kan diskuteras. På senare tid har ett antal arkitekturer vuxit fram med olika utgångspunkter. Dessa ser just nu ut att konvergera — en väsentlig process.

Inom databasområdet kan vi något förenklat säga att lagringsstrategin på den logiska nivån gått från enkla strukturer i databehandlingens barndom via ett mellanstadium med komplexa datastrukturer för att i det skede vi befinner oss just nu åter orienteras mot enklare strukturer. Codd's relationsteori har haft ett mycket starkt genomslag, kanske främst på modelleringsplanet. Än mer strängt normaliserade strukturer av binär typ vinner också terräng. Samtidigt kan vi konstatera att intresset för lagringsorienterade syntaktiska aspekter av data alltmör i debatten trängs ut av semantiska aspekter, dvs sådana som orienteras mot information eller det data står för. I Sverige lades tidigt en grundval till ett informationsorienterat synsätt genom Börje Langefors och andras arbete, men internationellt är det först under den allra senaste tiden som genombrottet skett. Semantikens och modelleringens värld är dock mycket svårhanterlig både teore-

tiskt och praktiskt. En komplex dataorienterad värld ersätts av en annan, informationsorienterad — men varför?

Då den holländska standardiseringskommissionen i samband med lanseringen av CODASYL 1973 DDL/DML förslagen önskade få dessa accepterade som standarder inom ISO kom detta inte som någon överraskning. Sverige hade t ex inom SIS och statskontoret diskuterat motsvarande åtgärd redan under SAAB D22 tiden. Redan då var frustrationen stor, ingen visste riktigt vad begreppet databashanteringssystem skulle stå för och många kände sig kallade att fylla detta lockande begrepp med innehåll eller att definiera det samma. Det fanns ett antal kandidater som kallade sig databashanteringssystem på marknaden och en allmän insikt om den enorma potentialen i ett standardiserat databasgränssnitt för de konventionella programspråken, främst COBOL och FORTRAN, började växa fram.

Emellertid var det redan för sent att genomföra en standardisering som eventuellt kunde utesluta något av de stora kandidaterna. IBM hade det hierarkiskt orienterade IMS i drift på bred front och relationsorienteringen spirade i bakgrunden, även om ingen kommersiell marknad fanns. CODASYL orienteringen förordades starkt inom ANSI av ett antal stora leverantörer bland andra företrädare av Charlie Bachman — upphovsmannen till den typ av diagram som vanligen beskriver nätverksdatabaser. Någon typ av kompromiss måste skapas, men hur?

Den lösning som fanns som en räddare i nöden hade skissats inom IBMs användarförening GUIDE/SHARE redan 1970. Vanligen hänförs idén till den 1975 publicerade så kallade ANSI/SPARC arkitekturen (se även [17]). Denna arkitektur, som utarbetades av en planeringskommitté inom American National Standards Institute, införde ett nytt begrepp som skulle bli fokuspunkten i nära nog allt senare databasrelaterat arbete — det konceptuella schemat.

I en tre-schema-arkitektur separerar det konceptuella schemat interna, lagringsorienterade aspekter från externa, program-presentationsoorienterade aspekter. Tanken är att det konceptuella schemat skulle kunna frigöras från allt vad som hade med lagrings- och pre-

sentationsorienterade begrepp och influenser att göra. Det konceptuella schemats separation av den faktiska lagringen från utnyttjandet medgav alltså att en given underliggande fysisk databas skulle kunna nyttjas av program skrivna mot en relationsdatabas lika väl som sådana program skrivna för en CODASYL-orienterad nätverksdatabas.

Här fanns alltså det Columbi ägg som genom en extra schemanivå mellan internt lagringsformat och programmets åtkomstformat introducerade närmast ett totalt dataoberoende. Idén tillät samexistens mellan programvara orienterad mot de stora tillverkarnas huvudprodukter och föreföll kunna bevara de enorma inverteringar i programvara som kundbasen stod för utan att rubba marknaden i någon speciell riktning. Den berömda arkitekturen baserad på tre schemanivåer är alltså inte den logiska utan snarare den politiska nödvändighetens barn.

Det konceptuella schemat har senare visat sig vara en idé som haft en mycket stor bärkraft av helt andra skäl än de som ursprungligen var styrande. Speciellt sedan modellering alltmer utvecklats mot en företagsstrategisk aktivitet och sedan begreppet information som gemensam resurs nått förbi slagordsstadiet. Redan ANSI/SPARC såg möjligheterna, men det är först genom ISO-arbetet med start 1976 som det konceptuella schemat tacklats från dessa utgångspunkter (se [6]).

Under tiden har dock krafter av en annan art verkat. Programvara, även basprogramvara som databashanterare är i minskande utsträckning en affär för maskinvaruleverantörerna enbart. Speciellt tydligt är detta inom smådatormarknaden, men även stordatormarknaden ser fenomenet klart. Kunderna framställer allt starkare krav på att kunna sammanfoga komponenter från olika programvaruleverantörer till fungerande totalsystem. Parallellt med detta har leverantörerna av programvara alltmer sett en komponentmarknad som attraktiv då totalinvesteringen i fullständiga programvaror är överväldigande. Detta blir speciellt accentuerat i en informationskatalogorienterad, distribuerad omgivning. Vad är då utvägen?

Amerikanska National Bureau of Standards lämnade mycket tidigt ett uppdrag till Computer Corporation of America för att utveckla en

generell databashanterarkitektur med definierade gränssnitt mellan komponenter med givna funktioner. Ett utkast till arkitektur presenterades av NBS 1982, framförallt som svar på ett kundtryck. Man skulle alltså kunna skapa sitt ideala, anpassade system genom att sätta samman komponenter som en schema-hanterare, en databashanterare, ett recovery-system, en formulärhanterare, ett tp-system, ett säkerhetssystem med någon vald kryptoalgoritm — allt från skilda tillverkare. Något var allt detta knutet till tankar om paketering i ADA packages, något till förhoppningar om en komponentorienterad industri kring C. De som har praktiska erfarenheter av integration av programvarukomponenter skrivna av samma leverantör för en egenutvecklad miljö anar redan här att en uppgift av denna art med komponenter från fristående leverantörer ligger på gränsen till det otänkbara om komponenterna definieras på låg nivå, vilket är tanken här. Mycken kritik har riktats mot förslaget, som dock innehåller ett antal intressanta ideer. Arkitekturen i denna tappning accentuerade databasadministratörens roll på samma sätt som tidigare ANSI/SPARC arbetet gjorde men utvecklade tankarna på användargränssnitt betydligt längre. Tanken om familjer av standarder inom en arkitekturs ram var lanserad. Vi känner igen den från annat håll, eller hur?

ISO OSI Open Systems Interconnection kan ses som ett mycket seriöst försök att konstruera en grundläggande referensmodell inom kommunikationsområdet. Inom denna isoleras olika fenomen och de olika lagren bildar i viss mån en bas för kompatibla familjer av protokoll, eller snarare standarder för gränssnitt. Effekten av detta arbete har varit mycket stor, och just referensmodellen har städad debatten inom kommunikationsområdet väsentligt. Samtidigt är denna referensmodell naturligtvis en arkitektur i sig, med mycket stora likheter med de försök till referensmodeller inom databasområdet som bl a gjorts inom ISOs grupp kring konceptuella modeller. Vad har så kommunikation och databaser gemensamt?

Skillnaden mellan att transportera data mellan olika applikationer som verkar mot samma databas i vid mening och att transportera data mellan olika applikationer som samverkar är inte speciellt stor. Om vi tänker oss en del av de samverkande applikationerna som databasapplikationer (servers) blir skillnaderna i problem på de översta lagren i modellen mini-

mala, jämfört med tre-schema-modellen. Det grundläggande problemet är också gemensamt — att på något sätt beskriva vilken information man vill ha, helst utan att behöva ta hänsyn till hur den är lagrad eller överförs från minnen eller över transmissionsmedier. Samma problem leder ofta till samma lösningar. Application layer i OSI motsvarar alltså ganska nära till dess allmänna inriktning Conceptual level nivån i tre-schema-arkitekturen, Presentation layer External level samt Session layer och underliggande lager Internal level och underliggande nivåer. Kan dessutom ett databasanrop ses som riktat mot intern lagring eller någon databas tillgänglig över datanät, likgiltigt vilket, så har vi alltså en gemensam struktur och referensmodell mellan kommunikation, en lokal databas och distribuerade databaser. I praktiken har detta på senare tid lett till en organisatorisk samordning mellan aktiviteterna inom området konceptuella schemaspråk och OSI.

Som en parentes kan nämnas att Sverige arbetar med utbytesformat för data, dvs närmast en aktivitet inom Presentation layer samt med standardisering av datatermer vilket ligger över gränsytan mellan Presentation och Application layer.

Går då det konceptuella schemats roll att beskriva närmare och vilka kandidater till schemaspråk finns? Är rentav en standard på gång?

4.4 Det konceptuella schemat.

Ett databassystem agerar normalt som en länk för mänsklig kommunikation. Liksom för direkt mänsklig kommunikation krävs en underliggande grammatik med syntaktiska och semantiska egenskaper kopplade till struktur och vokabulär. Det konceptuella schemat koncentreras alltså på presentationsoberoende, semantiska aspekter. Vad innehåller det egentligen?

I databasfallet utgör det konceptuella schemat en allmän överenskommelse om en gemensam uppfattning om en verksamhet, ett objektsystem, dvs det universum databasen är ägnat att beskriva. En grundläggande konsekvens av de begrepp som används i det konceptuella schemat är alltså en harmonisering av ett företags begrepps värld. Det konceptuella schemat skall alltså beskriva ett objekt-

systems uppträdande. De regler som definieras i ett sådant schema är alltså också därigenom (något förenklat) regler för hur beskrivningen av objektsystemet kan utvecklas och manipuleras.

Av humaneffektiva skäl och bekvämlighet vill vi låta olika användare välja olika presentationsformer av data på den externa nivån — antingen för direkt presentation i formulär eller för presentation till program som vidarebearbetar data. Dessa beskrivs i externa schemata. Samtidigt vill vi av lagringstekniska och andra skäl som har med maskineffektivitet att göra kunna utforma passande strukturer för intern lagring. Dessa beskrivs i interna schemata. I denna omgivning ger det konceptuella schemat möjligheter till tolkning av interna och externa representationer, dvs gör tolkning av dessa möjliga. Dessutom skall det konceptuella schemat se till att meningen, dvs informationen, inte förfalskas och distorteras i transformationerna mellan intern och extern nivå.

En kort summering av de centrala rollerna för det konceptuella schemat kan göras i fyra enkla meningar: Att ge en allmän bas för förståelse av objektsystemets beteende. Att definiera den tillåtna utvecklingen och möjligheterna att förändra informationen om objektsystemet i databasen. Att utgöra bas för tolkning (tolkning) av externa och interna syntaktiska formuleringar vilka representerar information om objektsystemet. Att utgöra basen för avbildningar mellan interna och externa schemata.

Inom ISO har vidare en mycket hög ambitionsnivå definierats för konceptuella schemaspråk i form av två principer [6]. Den första — som kallas konceptualiseringsprincipen — säger att ett konceptuellt schema endast skall innehålla konceptuellt relevanta aspekter av både statisk och dynamisk natur om objektsystemet och att alla aspekter av fysisk representation i databaser och accessvägar liksom alla aspekter av extern representation för användare skall uteslutas. Den andra — som kallas hundraprocentprincipen — säger vidare att alla relevanta statiska och dynamiska aspekter, dvs regler, lagar, etc, skall beskrivas i det konceptuella schemat — databassystemet kan inte hållas ansvarigt för regler som beskrivs på annan plats som i t ex applikationsprogram. Har vi nu en tillräcklig bas för att ut-

forma ett bra språk för det konceptuella schemat?

Inom ISO har under den senaste tvåårsperioden utformats ett utkast till värderingskriterier för konceptuella schemaspråk [7]. Det kan direkt sägas att en allmänt accepterad teoretisk bas för ett sådant arbete inte går att finna. Trots orienteringen mot klassisk logik i formuleringen av detta dokument är denna bas naturligtvis inget annat än en trossats — definitivt ingen nödvändighet. Det ligger i själva verket en stor fara i en alltför rigid ram, då denna kan spärra ut innovativa ansatser med andra utgångspunkter i grunden än de normerna står för. Det är naturligtvis ingen nyhet för de flesta att den klassiska logiken brottas med en mängd fundamentala problem. Vi tror snarare att helt nya ansatser behövs för att på längre sikt ge ett genombrott inom modelleringssfären.

De tre schemata som beskrivits ovan omfattar ju det mesta som har med användning, lagring och presentation av data att göra. Skulle vi inte kunna tänka oss dessa som basen i det man normalt kallar ett data dictionary?

4.5 Administration av informations- och dataresurser.

Om det konceptuella schemat under den senaste femårsperioden utgjort den korg i vilken man samlat de problem som väntar på en lösning, är i dagsläget begreppet "Information Resource Dictionary" en stark kandidat till att överta denna roll. Tanken är att de resurser i form av data och program som finns inom en organisation skall kopplas till organisatoriska, personella och ekonomiska begrepp samt sönderdelas i komponenter vars beroendeförhållanden skall dokumenteras för att bilda en bas för både administration, planering och faktiskt datautnyttjande — dessutom skall naturligtvis utvecklingsmiljön för system och program samt driften stödjäs.

Vi kan knappast säga att de arbeten som utförts inom NBS och ANSI till dags dato inger några större förhoppningar. Själva grundansatsen går ut på att definiera de begrepp som behövs för IRM-aktiviteten. Det innebär att konstruera ett konceptuellt schema för denna typ av verksamhet. Denna ansats kan visserligen betraktas som utbildande i ett tidigt skede, men att definiera hur en administrativ om-

givning skall se ut kan knappast vara standardiseringsorganisationernas egentliga syfte. De utkast till förslag till standarder som nu presenteras förefaller alltför rigida för att tjäna sina syften.

I Sverige drivs just nu inom statskontoret ett arbete inom datakatalogområdet. Det är ännu för tidigt att sja om arbetets utgång, men området är viktigt, svårt och mycket intressant. SISU, liksom flera av medlemsföretagen, är för övrigt representerade i arbetet.

Vad som krävs inom databasområdet är just nu att en lika hållbar referensmodell skapas här som inom kommunikationsområdet. Den situation föreligger inte i dagsläget. Innan detta skett kan knappast data dictionary eller datakatalogsbegreppet bli tillräckligt väl specificerat och uppdelat i komponenter för att möjliggöra en allvarligt syftande samling kring relevanta standardiseringsaktiviteter inom området.

En ytterligare komplicerande faktor är den nu-

varande trenden mot decentraliserade och distribuerade system. Kärnpunkten i distribuerade databaser kan beskrivas som lokal autonomi och global samverkan, dvs varje enskild databas skall i normalfallet kunna bearbeta och hantera lokala data men databaserna måste också kunna samverka vid behov av information som finns fördelade över dem. Det finns ingen täckande teori inom detta område, men det förefaller självklart att problemets kärna ligger inom området informations- och databeskrivning.

4.6 Janus

Standardiseringsarbetet står inför en utmaning av mycket stora mått. Dels ändras rollen för arbetet, dels ändras de teknologiska grundförutsättningarna i snabb takt — praktikkens värld har i mångt och mycket sprungit ifrån teorins. Ett betydligt närmare samarbete mellan det avancerade forskningsfältet och praktikfältet måste uppnås för ett fortsatt framtida produktivt arbete.

5. MER ATT LÄSA

För den som vill läsa mer på egen hand listar vi här ett antal böcker som i större eller mindre utsträckning behandlar området konceptuell modellering. Vissa av böckerna har inriktning mot databasdesign, naturligt nog, eftersom detta är ett av de traditionella användningsområdena för modellering. Andra är mer renodlat inriktade mot verklighetsorienterad begreppsmodellering. En del har en klart praktisk inriktning medan andra är mer filosofiska, teoretiska eller formalistiska.

För att hjälpa den som är intresserad har vi grupperat böckerna efter ungefärliga förkunskapskrav. För varje bok ges också en kort beskrivning av innehåll och inriktning. Rubriken på resp. delavschnitt utgöres av bokens titel.

5.1 För nybörjaren

5.1.1 Databaser och datamodeller

Sundgren, Bo
Studentlitteratur, Lund. 1981

Den här enkla och instruktiva boken vänder sig till den som vill ha översiktiskunskap om databaser och datamodeller. Den diskuterar också kort verklighetsorienterade datamodeller och begrepp som förekommer i detta sammanhang. Hur en verklighetsorienterad datamodell kan representeras i olika slag av databasorienterade datamodeller (relationsmodellen, CODASYL och den hierarkiska modellen) visas också. Boken lämpar sig väl för självstudier.

5.1.2 Konceptuell modellering - Informationsanalys

Bubenko, Janis Jr
Lindencrona, Eva
Studentlitteratur, Lund. 1984

Det här är en bok som är både introducerande och ganska djup. Den börjar relativt allmänt med att diskutera motiv för modellering samt att beskriva grundläggande problem, ansatser och begrepp. Några följande kapitel lägger en grund vad gäller matematik, relationsmodellen och logik. Därefter behandlas händelser och olika regler och villkor, vilket för fram till

det kapitel som beskriver ett av författarna uppfunnet språk, CMOL, för konceptuell modellering. CMOL är mycket generellt och omfattande och delvis rätt svårt. Beskrivningen, som delvis utgöres av syntax i BNF, är dessvärre inte särskilt lättillgänglig. Bokens avslutande del är utpräglat praktisk och behandlar diverse tillämpningsaspekter.

Sammanfattningsvis kan sägas att boken är grundläggande, innehållsrik och fränsett kapitlet om CMOL också lättläst, varför den bör lämpa sig väl för var och en som vill sätta sig in i området. Att den dessutom är skriven på svenska gör den ju inte mindre intressant i vår del av världen.

5.2. För dig med viss erfarenhet

5.2.1 Data Models

Tsichritzis, Dionysios C.
Lochovsky, Frederick H.
Prentice Hall Inc., Englewood Cliffs,
New Jersey. 1982

Denna bok kan sägas utgöra en sammanfattning av nuläget inom området datamodellering. Den börjar med att utförligt förklara och exemplifiera en stor mängd av de begrepp som kommer till användning då det gäller strukturer, regler, villkor och operationer mot datamodeller/databaser. Därefter beskrivs ingående ett antal av de vanligaste förekommande typerna av modeller, närmare bestämt

- Relationsmodeller
- Nätverksmodeller
- Hierarkiska modeller
- Entity-relationship-modeller
- Binära datamodeller
- Semantiska nätverk
- Infologiska modeller

Bokens sista del behandlar sättet att använda datamodeller, vad gäller främst databasutformning.

Boken innehåller många instruktiva exempel och är tämligen uttömmande inom sitt område. Den bör vara utomordentligt användbar, både för praktikern och den mer teoretiskt intresserade.

5.2.2 Data and Reality

Kent, William
North-Holland Publishing Company.
1978

”Data and Reality” är en bok som på djupet diskuterar innebörden i många av de begrepp som brukar användas i samband med konceptuell modellering. De första kapitlen tar upp sådant som:

- entiteter
- vad är ett informationssystem
- namngivning
- förhållanden (relationship)
- attribut
- typer, kategorier, mängder

De följande kapitlen diskuterar först begreppet modell i allmänhet och tar sedan upp några olika modelltyper, för att i ljuset av detta speciellt studera modellering av ”relationships”.

Så långt är boken huvudsakligen kritisk och visar på problem utan att egentligen ange några lösningsvägar. Avslutningsvis skisseras dock en modelleringsansats som, enligt författarens åsikt, innehåller de ”rätta” konstruktionselementen för konceptuella modeller.

Boken är mycket intressant för den som på djupet vill tränga in i den begreppsvärld inom vilken man rör sig i modelleringskretsar.

5.2.3 Database Analysis and Design

Robinson, Hugh
Chartwell-Bratt, England;
Studentlitteratur, Sverige. 1981

Detta är en introducerande och lättläst, men trots detta grundlig och då det gäller databaser ganska heltäckande bok. Den består av tre delar benämnda Fundamentals, Analysis och Design.

Den första delen handlar om motiven bakom databaser och arkitekturen hos databassystem samt ger exempel på olika typer av databassystem och datamodeller. Här redogöres också för modeller på olika nivåer i förhållande till datorsystemet.

Bokens mellersta del behandlar informationsanalys och handlar till rätt stor del om koncep-

tuella modeller. Den modelleringsansats som används är objektorienterad och här finns också ett avsnitt om normalisering.

Den sista delen handlar om den slutliga utformningen av databasen, där den konceptuella modellen utgör en utgångspunkt. Här diskuteras också sådant som säkerhetsaspekter och databasadministration. Det sista kapitlet handlar om Data Dictionary.

Som helhet kan man säga att boken har en klar praktisk inriktning och innehåller många exempel. Den är dock ganska CODASYL-orienterad.

5.3 För specialisten

5.3.1 Information Modelling

Bubenko, Janis A. Jr, Editor
Studentlitteratur, Sverige;
Chartwell-Bratt, England. 1983

Denna bok innehåller artiklar, som använts i en vetenskaplig kurs i modellering. De är skrivna av internationellt erkända storheter inom området, är genomgående av hög kvalitet och delvis av mycket hög teoretisk nivå.

Bland det som behandlas är

- Metodik för specifikation av informationskrav
- Semantiska datamodeller
- Lingvistiska aspekter
- Teori för relationsdatamodellen
- Metoder och verktyg för databasutformning.

Boken lämpar sig bäst för den som är intresserad av att sätta sig in i de teoretiska grundvalarna för informationsmodellering eller lära känna ”state of the art” inom området.

5.3.2 On Conceptual Modelling

Edited by Brodie, Michael L
Mylopoulos, John
Schmidt, Joachim W
Springer-Verlag New York Inc. 1984

Den här boken behandlar ämnet konceptuell modellering ur tre olika perspektiv, nämligen från AI-, databas- och programmeringsspråksynpunkt. Den leder sitt ursprung till ett veten-

skapligt symposium som hölls 1982 med deltagande av internationellt välkända forskare inom dessa tre områden. Boken tar alltså upp många av de senaste forskningsfrågorna och rönen. Den är därför knappast lämpad för den rene nybörjaren utan rätt stor vana vid de grundläggande begreppen krävs nog.

För att skapa en plattform för en läsare som inte är insatt i alla tre områdena så utgör de tre första kapitlen en kort introduktion till vart och ett av dem. Övriga kapitel svarar mot olika författares bidrag till konferensen. Varje kapitel avslutas med en sammanfattning av den diskussion som förekom i anslutning till bidraget vid symposiet. Härigenom får man också en del synpunkter för och emot och en viss uppfattning om vad som är problematiskt och värt att diskutera inom området.

5.3.3 Database Design Methodology

Vetter, M
Maddison, R.N
Prentice-Hall International, Inc.
Englewood Cliffs, N.J1981

Syftet med denna bok är, som namnet anger, att beskriva en metodik för databasutformning. I denna metodik spelar konceptuell modellering en viktig roll.

Efter ett inledande kapitel om bokens uppläggning m m, följer en ganska klar och lättfattlig beskrivning av den matematiska begreppsapparat som brukar användas i modellerings-sammanhang. Därefter diskuteras modellering av verkligheten utifrån ett mängdteoretiskt och relationsorienterat synsätt. Den sista

tredjedelen behandlar de mer databastekniskt orienterade faserna i databasdesignprocessen.

Boken är ganska teoretisk och formell, men innehåller också många förklarande figurer och tabeller. Med tanke på den metodiska uppbyggnaden bör boken vara av intresse för envar som är verksam inom databasdesignområdet.

5.3.4 Design of Database Structures

Teorey, Toby J
Fry, James P
Prentice-Hall International, Inc.
Englewood Cliffs, N.J1982

Detta är en mycket innehållsrik bok, som beskriver hela databasutformningsprocessen, alltifrån formulering och analys av informationskrav, via konceptuell modellering till fysisk databasdesign. Bokens senare delar är klart beräkningsinriktade.

De kapitel som behandlar informationsmodellering gör detta utifrån ett praktiskt, metodiskt perspektiv, där man går ganska långt då det gäller att specificera modellens detaljer. Ungefär en fjärdedel av boken behandlar de inledande stegen av databasdesignprocessen, medan återstoden är orienterad mot databasteknik.

Den här boken ger en hel del information om och synpunkter på konceptuell modellering, men är framför allt värdefull för den som är intresserad av en beräkningsmässig analys av olika datastrukturer o dyl.

6. UTBILDNING

Hur kan man i Sverige utbilda sig i konceptuell modellering? På den privata sidan finns fragment och grundläggande avsnitt av konceptuell modellering med i flera konsultföretags och utbildningsföretags kurser och utbildningsprogram. Ämnet går under olika namn såsom objektmodellering, datamodellering, begreppsmodellering och informationsmodellering. Det är ovanligt att modellering här presenteras i form av en egen, specialiserad kurs. Vanligast är att man tar med modelleringsavsnitt i kurser med större bredd såsom verksamhetsanalys, databasutformning eller IRM - Information Resource Management. Det är vidare vanligt att man i dessa sammanhang lär ut ett visst speciellt språk (notation) för modellering. En kartläggning av detta utbildningsutbud i modellering har dock inte ingått i våra ambitioner med detta nummer av SISU-analys.

På den offentliga sidan är det främst fråga om universitetsutbildning och då i form av enskilda kurser i ADB eller som komponent i den systemvetenskapliga utbildningslinjen. Oss veterligt ges grundläggande utbildning i KM vid universiteten i Stockholm och Göteborg. En kurs planeras även för årskurs 3 vid KTHs relativt nystartade datalinje. Vissa aspekter av modellering ingår givetvis även i andra slag av universitetskurser såsom databaser och databasmetodik, informationsanalys och i ett flertal kurser i området AI och kunskapsrepresentation.

Nedan anger vi preliminära skisser för tre kurser i konceptuell modellering som avses att hållas i SISU:s regi under 1985/86.

KM1 är en endags, orienterande kurs för personer som vill få en uppfattning om denna tekniks grundbegrepp och tillämpning.

KM2 är avsedd för användare som under specialisthandledning skall medverka vid utformning av konceptuella schemata om delar av sin organisation eller verksamhet.

KM3 förutsätter KM2 och är till för modelleringsspecialister, dvs personer som skall ansvara för utformningen av sådana modeller.

KM1 - EN ÖVERSIKTSKURS

Konceptuell modellering är ett relativt ungt metodområde. De tekniker och metoder som utvecklas där syftar till en bättre överblick och förståelse av det verksamhetsområde som är föremål för systemutveckling samt till en stringent och problemorienterad beskrivning av information och data inom området. En konceptuell modell beskriver och definierar vad information om ett verksamhetsområde (objektsystem) avser, dvs begrepp, objekt, associationer, förhållanden, attribut och händelser. Men en modell omfattar även en mängd andra antaganden om objektsystemet såsom olika regler och villkor i verksamheten. En sådan modell är ett nödvändigt underlag för att sedan utveckla och konstruera ändamålsenliga databehandlingssystem. Målet för denna kurs är att ge deltagarna en ingående orientering om och förståelse av begrepp, metoder och olika ansatser vid konceptuell modellering och belysa denna tekniks roll vid analys och specificering av användares informationskrav och vid systemkonstruktion.

KURSEN VÄNDER SIG TILL

administrativ och/eller teknisk personal i ansvarig ställning och som vill orientera sig om en ny och kraftfull teknik för verksamhets- och begreppsanalys och specificering av underlag och krav på databasorienterade system. Kursen förutsätter ej tidigare erfarenhet av ADB.

INNEHÅLL

- Systemutvecklingsprocessen och dess olika faser (med tonvikt på de inledande etapperna).
- Analys- och beskrivningsmetoder och modeller i de inledande etapperna: verksamhetsflöden, informationsflöden, konceptuella modeller, m fl samband mellan dessa modeller.
- Varför konceptuell modellering?
- Grundbegrepp för konceptuell modellering: språket och dess roll, objektsystem,

utsagor, fakta, regler, ostrukturerad vs formell strukturerad beskrivning av ett objektsystem.

- Begrepp för modellering: objekt (entiteter), objekttyper, egenskaper, attribut, förhållanden, relationer, m.m.
- Enkel grafisk notation för objektorienterad modellering.
- Något om metodik vid modellering. Orientering om olika modelleringsansatser, hjälpmedel och verktyg.
- Exempel på praktiska tillämpningar vid verksamhetsanalys, informationsanalys och vid databaskonstruktion och dataadministration.

OMFATTNING: 1 dag

KURSMATERIAL

Kompendium, overheadbilder, tidskriftsartiklar.

KM2 - KONCEPTUELL MODELLERING - GRUNDKURS

Översikt-kunskaper i konceptuell modellering har förmedlats i kursen K1. Denna kurs syftar till att ge deltagarna grundläggande kunskaper och färdigheter i objektorienterad konceptuell modellering och dithörande metoder och tekniker. Målet för kursen är att deltagarna efter genomgången kurs skall kunna medverka vid utveckling och specifikation av enklare slag av konceptuella modeller. Orientering ges även om relationsmodellen och dess förhållande till objektorienterade modeller. Kursen ger också en bas för att tillägna sig mer avancerade kunskaper i detta område och kunna ha förståelse för egenskaperna hos olika metoder och ansatser i detta område.

KURSEN VÄNDER SIG TILL

administrativa utredare och övrig personal som skall medverka vid kravspecifiering och modellering av verksamheter och informationssystem.

INNEHÅLL

- Varför modellering? Modellers roll i systemutvecklingsprocessen och vid systemförvaltning (rekapitulation från K1).
- Konceptionella modeller och dess samband med andra modeller såsom verksamhets (flödes) modeller och informationssystemmodeller (rekap. från K1).
- Grundbegrepp för objektorienterad modellering (objekt, attribut, förhållande, association, relation, schema, fakta vs regler m m).
- Teknik och notation för formell beskrivning. (Mängder, relationer och funktioner. Grafisk notation för modellering.)
- Relationsmodellen (grundbegrepp, relationsalgebra, normalisering, m m.)
- Beteende, regler och villkor i objektsystem.
- Överblick över några vanliga ansatser för modellering.
- Grupparbete och övningar i konceptuell modellering. Arbete på praktikfall.

OMFATTNING: 3 dagar

KURSMATERIAL

Janis Bubenko jr och Eva Lindencrona: Konceptuell modellering, Studentlitteratur, Lund 1984.

Praktikfallsbeskrivning, kopior av overheadbilder, artiklar.

KM3 - KONCEPTUELL MODELLERING - FORTSÄTTNINGSKURS

Syftet med denna kurs är att ge fördjupade kunskaper och färdigheter i objektorienterad konceptuell modellering. Kursen förutsätter förkunskaper motsvarande KM2. På kursen lärs ut ett komplett språk för konceptuell modellering (CMOL) och övning ges i dess tillämpning. Vidare ges träning i modelleringsmetodik. Orientering ges också om olika hjälpmedel för modellering samt om olika Artificiell Intelligens-baserade tekniker för kunskapsrepresentation i administrativa system. Kursen ger en nödvändig grund för praktisk metodtillämpning och metodutveckling resp anpassning.

KURSEN VÄNDER SIG TILL

administrativa utredare, metodansvariga, systemutvecklare och programmerare med f6rkunskaper motsvarande KM2 och som vill tilll6mpa konceptuell modelleringsteknik i sin organisation.

INNEH6LL

- Formell notation f6r modellering: orientering om satslogik, predikatlogik, modellteori
- Relationsmodellen - normaliseringsteori
- CMOL - ett komplett spr6k f6r konceptuell modellering. Genomg6ende av dess syntax och semantik.
- 6vningar och genomg6ng av praktikfall.
- Modellering med tidsdimension.
- Kunskapsrepresentation i administrativa system: 6versikt 6ver metodik fr6n AI-omr6det med relevans f6r (framtida) konceptuell modellering.

– Metodik vid modellering:

- * modellering i systemutvecklingsprocessen
- * fr6n informationskrav till konceptuell modell
- * representations- och avbildningsproblem
- * integration av 'lokala' modeller
- * anv6ndarmedverkan
- * fr6n konceptuell modell till databas.

OMFATTNING: 3 dagar

KURSMATERIAL

Janis Bubenko jr och Eva Lindencrona: Konceptuell modellering, Studentlitteratur, Lund, 1984.

6vningsuppgifter. Praktikfall.

REFERENCES

- (1) M.L. Brodie, J. Mylopoulos, J.W. Schmidt.
On Conceptual Modeling: Perspectives from Artificial Intelligence Databases and Programming Languages.
Springer Verlag, 1983.
- (2) M.L. Brodie, Editor.
Islamorada Workshop on Large Scale Knowledge Base and Reasoning Systems.
not yet published, contact editor at CCA, Four Cambridge Center, Cambridge, Mass., 02142, 1985.
- (3) ACM.
Proceedings of the Workshop on Data Abstraction, Databases and Conceptual Modeling.
ACM SIGMOD Record Vol. 11, No. 2, edited by M.L. Brodie and S.N. Zilles, 1981.
- (4) D.J. DeWitt, G. Gardarin, Editors.
SIGMOD 83 Proceedings.
ACM SIGMOD Record, Vol, 13, No. 4, 1984.
- (5) C.A. Ellis, Editor.
Second ACM-SIGOA Conference on Office Information Systems.
ACM Inc., USA, 1984.

- (6) J.J. van Griethuisen et al (edt).
Concepts and terminology for the conceptual schema and the Information Base.
 Report ISO TC97/SCS/WG3, 1982.
- (7) ISO TC97/SC5/WG3.
Assessment Guidelines for Conceptual Schema Language Proposals.
 ISO Working Group (DRAFT) Report, Edited by J.J. van Griethuisen, June, 1984.
- (8) L. Kerschberg, Editor.
Expert Database Systems: Proceedings of the first international workshop on EDS, Kiawah Island, SC, USA, Oct. 24-27, 1984.
 Institute of Information Management, Technology and Policy, Univ, of South Carolina, Columbia, SC, 29208, 1984.
- (9) J.W. Kimbie, K.I. Koffeman, Editors.
Data Base Management.
 North Holland-American Elsevier Publ. Co, 1974.
- (10) T.B. Steel, R. Meersman, Editors.
Database Semantics, DS-1: IFIP TC2 Working Conference organized by WG2. 6.
 To be published by North-Holland Publishing Co., Amsterdam, 1985.
- (11) G.M. Nijssen, Editor.
Modelling in Data Base Management Systems.
 North-Holland Publ. Co, Amsterdam, 1976.
- (12) G.M. Nijssen, Editor.
Architecture and Models in Data Base Management Systems.
 North-Holland Publ. Co., Amsterdam, 1977.
- (13) T.W. Olle, H.G. Sol, A.A. Verrijn- Stuart, Editors,
Information Systems Design Methodologies: a Comparative Review.
 North-Holland Publ. Co., 1982.
- (14) T.W. Olle, H.G. Sol, C.J. Tully, Editors.
Information Systems Design Methodologies: a Feature Analysis.
 North-Holland Publ. Co., 1983.
- (15) H.J. Schneider, Editor.
Formal Models and Practical Tools for Information System Design.
 North-Holland Publ. Co, Amsterdam, 1979.
- (16) H.J. Schneider, Editor.
Automated Tools for Information Systems Design.
 North-Holland Publ. Co, Amsterdam, 1982.
- (17) D. Tsichritzis, A. Klug (eds).
The ANSI/X3/SPARC DBMS Framework.
 Information Systems, Vol 3, Nr 4, 1978.
- (18) B. Yormark, Editor.
SIGMOD 84 Proceedings.
 ACM SIGMOD Record, Vol. 14, No. 2, 1984.