
Krav på IA

■ Nästa Generation Modellering

Avancerad utbildning för handledare

Katalogprinciper

Uttagssystem

Informationspridning

Detta exemplar är avsett för
"stinsning" d v s remiss inför
slutlig version.
Slutdatum: 6 januari 1992.
Svar till SISU, Lars Bergman,
Box 1250, 164 28 Kista

■ Rapport N nr 1: Modelleringsansatser för begrepps- och datamodellering: – Beskrivning och försök till jämförelse

Modelleringsansatser för begrepps- och datamodellering: – Beskrivning och försök till jämförelse

Harriet Dahlgren,
Mats Gustafsson
Lars-Åke Johansson
SISU

Spridningsförbehåll:

Denna rapport får endast spridas och användas inom de organisationer som deltar som parter i TRIAD-projektet.

© TRIAD-parterna nov 1991

Rapporten är skriven i och för TRIAD delprojekt Nästa Generation Modellering.

Innehåll

Modelleringsansatser för begrepps- och data- modellering: – Beskrivning och försök till jämförelse

FÖRORD.....	2
1.0 Inledning.....	3
2.0 Modeller och modellering	4
2.1 Olika former av modellering.....	4
2.2 Konceptuell modell och datamodell.....	6
3.0 Metoder och modellering	9
3.1 Metoder	9
3.2 Framtagning av modeller.....	13
4.0 Varför modellera?	15
4.1 Användningsområden för modellering	15
4.2 Krav och önskemål på konceptuell modellering	16
4.3 Syfte med konceptuell modellering	16
5.0 Modelleringsansatser & modelleringsspråk	20
5.1 Referensramar och kunskaper	20
5.3 Typer av modelleringsansatser	23
6.0 Konceptuell modellering – grundbegrepp och operationer	25
6.1 Grundbegrepp.....	25
6.2 Operationer.....	29
7.0 En nivåindelad modelleringsteknik.....	31
8.0 Övergång från konceptuell modell till databasmodell	34
8.2 Från konceptuell modell till relationsdatamodell (tabeller).....	35
8.3 Från relationsdatamodell till modell med effektivitetsmässiga hänsyn.....	36
9.0 Presentation av någramodelleringsansatser.....	37
10.0 Sammanställning och försök till jämförelse.....	65
10.1 Karakterisering av ansatserna.....	66
10.2 Skiss till tänkbar värdering av ansatserna	68

FÖRORD

Denna rapport bygger vidare på arbete som ursprungligen utfördes inom DA- (Dataadministrations-) projektet - ett samarbetsprojekt inom SISU:s ram. I detta projekt deltog representanter för ett stort antal företag och organisationer. Bidrag till föreliggande rapport har därför gjorts av fler personer än av oss som står som författare. Vi vill därför passa på att tacka alla bidragsgivare, speciellt inom DA-projektets metod- och modelleringsgrupp - ingen nämnd och ingen glömd.

Författarna

1.0 Inledning

Denna rapport vänder sig till de som är intresserade av metoder och beskrivningstekniker i samband med modellering. Verksamhetsmodellerare, IA-berörda och systemutvecklare, för att nämna några.

Rapporten behandlar metoder och beskrivningstekniker för såväl konceptuell modellering med speciell inriktning på begrepps-, objekt- och datamodellering. Denna skrift är ingen färdig guide för vilken metod och teknik som skall användas vid ett visst modelleringstillfälle. Istället ger den en allmän översikt över metoder och tekniker för modellering. Några representativa ansatser har studerats mer i detalj, med användning av ett genomgående exempel.

Någon fullständig uppräkningslista av modelleringstekniker och metoder är inte möjligt att tillhandahålla med dagens utbud med säkerligen över 200 ansatser för konceptuell modellering. Det är inte heller avsikten med denna skrift.

Rapporten är upplagd enligt följande. I kapitel 2 presenteras några olika slag av modeller och modellering. Konceptuella modeller diskuteras mer detaljerat i detta avsnitt.

Det tredje kapitlet tar upp metoder i samband med modellering. Metoder är avgörande i modelleringssammanhang. Metoden och beskrivningstekniken bör gå hand i hand.

Användningsområden för och syften med modellering diskuteras i rapportens fjärde del. Den följs av ett avsnitt om modelleringsansatser och modelleringsspråk där ett antal krav och kriterier listas och en typindelning görs.

Kapitel 6 tar upp grundbegrepp och operationer inom konceptuell modellering. Hur man inom en och samma modelleringsteknik kan använda varierande detaljrikedom och uttrycksstyrka, för att anpassa modellerna till olika personer som skall använda dem, diskuteras i kapitel 7.

Vad som gäller när en konceptuell modell skall överföras till en databasmodell berörs i det därpå följande avsnittet. I kapitel 9 presenteras, för sex stycken modelleringsansatser. Presentationerna omfattar en informell beskrivning av viktiga komponenter och inslag i ansatserna samt en exempelmodell.

Slutligen, i kapitel 10, görs en sammanfattning av egenskaper hos modelleringsansatserna med inslag av jämförelse på några punkter.

2.0 Modeller och modellerir

En modell är alltid skild från det i verkligheten (objektsystemet) som den skall avbilda. Vilken utformning en modell får är hårt knutet till det syfte som finns för modelleringen. Modellen är alltid en förenkling av sin förlaga.

Några kategorier av modeller är:

- * Ikoniska modeller
- * Analoga modeller
- * Symboliska modeller
- * Verbala modeller

Ikoniska modeller liknar sitt original både till utseende och med de egenskaper de har. Det gör att de är lätta att sätta sig in i, men också att de är svåra att ändra.

Analoga modeller använder sig av speciella egenskaper inom modellen, som motsvarar egenskaper i objektsystemet. Exempel på detta är kurvor och diagram.

Symboliska modeller representerar verkligheten med symboler (t ex bokstäver). Det är den vanligaste men mest abstrakta modelltypen. De representationsregler som används är inte alltid uppenbara.

Verbala modeller använder naturligt språk som beskrivningsmetodik. De är flexibla modeller med stor uttryckskraft men inte speciellt exakta.

2.1 Olika former av modellering

Modellering av tekniska applikationer har en mycket längre tradition än modellering av administrativa verksamheter.

Behovet av att modellera flera aspekter av en verksamhet har blivit större i och med strävan efter en utveckling av bra och samordnade datorsystem.

Idag bedrivs i varje välplanerat projekt för utveckling av informationssystem såväl **verksamhetsanalys** som **informationsanalys**. Ett användbart sätt att presentera resultat från dessa analyser är i form av modeller.

Olika former av modellering kan göras för att fånga olika aspekter. I många avseenden överlappar de. Det finns inte någon klar enhetlig definition av olika modelleringsslag. Här följer ändå en uppräknig av några former av modellering och några frågor som de kan beröra. Uppräknigingen skall inte ses som exakt eller uttömmande:

Verksamhetsmodellering - Hur kan en bild av en verksamhet skapas med avseende på ett visst syfte? Vilka mål och behov finns? Vilka uppgifter har och kommer verksamheten att få? Vilka funktioner skall finnas? Hur ser organisationen ut? Hur skall ansvarsfördelningen vara?

Vilka resurser disponeras?

Målmodellering - Vilka mål har verksamheten? Vilka delmål bidrar till mer övergripande mål? Finns det mål som är motstridande? Vilka åtgärder och problem kan urskiljas?

Begreppsmodellering - Vilka begrepp används inom verksamheten? Hur definieras dessa begrepp? Är uppfattningen om begreppen samstämmig? Hur generella är begreppen?

Objektmodellering - Hur kan enskilda företeelser i verksamheten, som det är intressant att hålla information om, klassificeras till "objektklasser" och samband mellan dem? Finns olika nivåer på vilka man samlat vill kunna tala om viktiga företeelser i verksamheten?

Datamodellering - Vilka begränsningar och speciella egenskaper har objekt i verksamheten? Vilka samband råder mellan dem? Vad behöver vi veta om verksamhetens begrepp och data?

Informationsmodellering - Vilken information hanteras i verksamhet? Vad kännetecknar informationen? Till vilka objekt kan den sägas vara kopplad?

Databasmodellering - Hur skall data struktureras för att kunna lagras i form av en databas? Hur skall datastrukturerna optimeras och effektiviseras för att resultera i en databas med goda egenskaper och prestanda?

Konceptuell modellering - Vilka är verksamhetens centrala begrepp/objekt och vad gäller för dessa i form av samband, begränsningar och regler av såväl statisk som dynamisk natur? Om man kan få en enhetlig, heltäckande modell av detta utgör det ett utmärkt underlag för andra mindre eller mer detaljerade modeller. Konceptuell modellering inrymmer ofta begrepps-, objekt- och datamodellering av semantisk art.

Funktionsmodellering - Vilka funktioner finns i verksamheten? Vilka delfunktioner? Vem är ansvarig för dem? Hur går de till? Vilka uppgifter ingår?

Flödesmodellering - Vilka informations- och sakflöden kan man identifiera i verksamheten? Kan de effektiviseras? Behövs det datorstöd i vissa arbetspunkter med tunga arbetsuppgifter avseende informationsbelastning eller dylikt?

Konceptuell modellering - begrepps-, objekt- och semantisk datamodellering - kommer att behandlas mer ingående i denna skrift.

Däremot ges knappast något utrymme för funktionsmodellering o dyl.

2.2 Konceptuell modell och datamodell

Inom området konceptuell modellering/datamodellering finns en omfattande flora av begrepp för modeller av olika slag. Man talar bl a om:

- * datamodeller
- * databasmodeller
- * semantiska datamodeller
- * objektmodeller
- * begreppsmodeller
- * konceptuella modeller

Alla har det gemensamt att de, med hjälp av en viss, utvald normativ uppsättning av modelleringsbegrepp beskriver eller avbildar vissa aspekter av den del av en verksamhet som ett informationssystem handlar om.

Skillnader som kan finnas mellan de olika typerna av modeller kan beröra två aspekter:

- * omfattningen av modellerna dvs vilka aspekter av verksamheten/verkligheten som skall beaktas och avbildas i modellen
- * hur pass implementerings-/datoroberoende de använda modelleringsbegreppen är.

2.2.1 Konceptuell modell

ISO (Internationella StandardiseringsOrganisationen), har tagit fram en rapport som behandlar begrepp och språk för konceptuell modellering ("Concepts and Terminology for the Conceptual Schema and the Information Base") [2]. Man gör i denna en distinktion mellan modell och schema.

Det man vanligen brukar kalla konceptuell modell eller (semantisk) datamodell motsvarar med ISO-terminologi ett konceptuellt schema. En konceptuell modell skall här också innefatta en sk informationsbas, som innehåller registrerade fakta och händelser i verkligheten/verksamheten.

Enligt ISO skall ett konceptuellt schema inte bara innehålla en strukturbeskrivning av innehållet i informationsbasen utan också ange regler och villkor för hur informationsbasen får förändras

d v s vilka tillägg, borttag och ändringar som är giltiga.

Vid konceptuell modellering, enligt ISO:s synsätt, syftar man till att åstadkomma en utökad beskrivning jämfört med den man vanligen gör vid "traditionell" datamodellering. För vad denna utökade beskrivning skall omfatta

finns riktlinjer i form av två principer, kallade 100%-principen och konceptualiseringsprincipen.

Det kan vara intressant att återge innehållet i dessa två principer här:

* **100 %-principen**

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

* **Konceptualiseringsprincipen**

Ett konceptuellt schema skall bara omfatta konceptuellt relevanta aspekter, statiska och dynamiska, av den avgränsade verkligheten/verksamheten, och därmed exkludera alla aspekter som har med datarepresentation, dataorganisation, åtkomst, lagring, format, presentation, etc, att göra.

I ett databehandlingssystem ligger viktiga antaganden om den bakomliggande verksamheten inbyggda i datastrukturer, programkod och lagringsstrukturer. Konceptuell modellering syftar till att göra dessa antaganden om verksamheten explicita och möjliga att uppfatta.

Det viktiga att konstatera här är den starka betoningen av att det bara är *konceptuellt relevanta* aspekter som skall tas med - dvs alla implementerings- och datorberoende aspekter skall "silas bort" - *alla regler* i verksamheten skall avbildas och ingå i modellen - och att *dynamiska* aspekter skall beaktas.

Det säger sig självt att en modelleringsansats för konceptuell modellering i denna mening måste förse modelleraren med en rikare uppsättning modelleringsbegrepp och ha ett mer utvecklat modelleringsspråk för t ex beskrivning av regler, än en modelleringsansats som syftar till att beskriva en delmängd av detta, exempelvis statiska datamängder.

En avgränsning som görs i många modelleringsansatser är att enbart beakta de statiska aspekterna och därmed förknippade regler av enklare typ.

2.2.2 De tre "klassiska" datamodellerna

Man brukar tala om de tre "klassiska" datamodellerna, tre huvudtyper av datamodeller som de flesta tillgängliga databashanterarna bygger på:

- * CODASYL-(nätverks-)datamodellen
- * den hierarkiska datamodellen
- * relationsdatamodellen

Var och en av dessa ansatser har sin egen uppsättning struktureringsbegrepp för att strukturera data i en databas. De brukar därför också ibland kallas för databasmodeller.

Nätverksmodellen strukturerar data i posttyper och set-typer, den hierarkiska modellen (en begränsad nätverksmodell) i poster och segment, relationsmodellen slutligen strukturerar data i tabeller.

De databasorienterade ansatserna använder modelleringsbegrepp som har sitt ursprung i begrepp som används för att strukturera data i filer och databaser på datorers sekundärminnen.

Numer anses de därför inte fristående vara idealiska modelleringsansatser på den konceptuella nivån, där med ISO-terminologi "endast konceptuellt relevanta aspekter skall beaktas".

Relationsansatsen, med sina enkla struktureringsprinciper och välutvecklade matematiska förankring har nått stor spridning och användning. Till denna ansats har dessutom olika påbyggnader gjorts i syfte att kunna göra datadefinitionerna semantiskt rikare.

2.2.3 Semantisk datamodellering

När man idag talar om datamodellering menar man egentligen inte modellering enbart med hjälp av någon av de tre klassiska datamodellerna. I stället talas om datamodellering med faser där en rikare semantik och en högre abstraktionsnivå, som minskar beroendet av datorlagringstyp, används.

Om den renodlade relationsansatsen används så görs det alltid i kombination med någon objektorienterad ansats. Den objektorienterade modelleringen föregår den "tabellorienterade" modelleringen.

I en systemutvecklingsprocess måste en övergång göras från den problemorienterade konceptuella modellen till en databasmodell - ett schema för en databas. Beroende på hur pass rik och "konceptuellt inriktad" modellen är, blir denna övergång mer eller mindre komplicerad och omedelbart transparent - steget blir mer eller mindre långt att ta.

3.0 Metoder och modellering

3.1 Metoder

3.1.1 Inledning

I denna skrift studeras och jämförs ett antal modelleringsansatser med inriktning på s.k. begrepps- eller datamodelleringstekniker. Det finns en hel mängd olika typer av modelleringstekniker och en större studie skulle ha behövt genomföras om man vill jämföra olika typer av modelleringstekniker över flera metoder.

Metoderna bygger på ett flertal olika synsätt, som resulterar i att olika typer av tekniker växer fram och anses lämpliga.

3.1.2 Att förändra verksamheter

Arbete med att förändra verksamheter är ett mycket komplicerat arbetsområde. En mängd aspekter av en verksamhet måste aktualiseras för att man skall kunna komma fram till bra förslag innan förändringar genomförs. Åminstone om man i någon mån vill prognosticera effekterna innan förändringarna genomförs.

I många verksamheter är informationsförsörjningen en allt viktigare del för att, bland annat, kunna arbeta effektivt i verksamheten. Den är ofta ett konkurrensmedel.

Många anser numera att det är av vikt att inte förändra informationssystemen isolerat ifrån andra förändringar. Förslagen till ändring av informationssystemen bör helst komma upp i samband med att man genererar ideer till förändring av verksamheten som helhet.

Allt fler systemutvecklingsmetoder har därför intagit ett synsätt som bygger på detta sammanvävande av förändringar av informationssystem och verksamhetsförändringar i övrigt.

En viktig del av att förändra verksamheter är att människor i organisationen behöver kommunicera kring att hitta goda inriktningar på förändringen. Verksamheter överhuvudtaget är beroende av att människor kommunicerar och detta accentueras under förändringsprocesser.

Synsätt kring medverkan för berörda i förändringsprocesser spelar naturligtvis in på graden av kommunikation. Problemet är dock ofta att göra kommunikationen kring förändringsprocesser effektiv.

3.1.3 Att göra beskrivningar

En mängd aspekter av en verksamhet måste kunna hanteras när man funderar på att förändra en verksamhet. Ett sätt att försöka överblicka dessa aspekter är att mer explicit försöka beskriva dem.

Det kan finnas flera skäl till att man behöver göra explicita beskrivningar. Ett kan vara att beskrivningarna för den enskilde stimulerar tänkandet så att man blir mer systematisk och att man lättare identifierar problemen samtidigt som man försöker hitta mål och lösningar till problemen. Ett annat skäl kan vara att beskrivningen ökar kommunicerbarheten *mellan* människor. Kommunikationen kan på detta sätt förstärkas. Man kan på kortare tid identifiera om man är överens eller att man har skilda uppfattningar exempelvis om "hur det ser ut idag".

Man kan naturligtvis komma fram till att man *inte* har en gemensam uppfattning. Detta är ju emellertid också ett *resultat* till följd av kommunikationen och man får gå vidare utifrån detta.

Aspekterna kring kommunicerbarhet är naturligtvis viktiga i en verksamhet där det i regel gäller att samarbeta mellan människor.

Modellering i samband med förändringsarbete, och metoder därför, har därför blivit alltmer uppmärksammat.

Man kan också nämna att många modelleringstekniker har drag av *överblick*. Detta för att människan har olika svårigheter att samtidigt uppfatta en stor mängd aspekter av en "verklighet". Detta kan sägas vara ytterligare ett skäl till att skapa explicita modeller.

3.1.4 Beskrivningar för olika syften

Det kan röra sig om beskrivning av:

- verksamhetsfunktioner (kluster av verksamhetsuppgifter som skall utföras)
- aktiviteter och sekvenser av aktiviteter
- explicita verksamhetsregler
- relevanta "fenomen" i verksamheten (objekt, relationer, attribut för objekten etc.)
- vilka informationssystem som finns i en verksamhet och vilka datalager dessa utnyttjar
- vilka organisatoriska enheter som finns i en verksamhet och vilka ordervägar som finns

- vilka sociala grupperingar som finns i verksamheten och hur de interagerar
- vilka problem och orsakssammanhang dessa har
- vilka överordnade och vilka underordnade mål som finns
- hur information flödar mellan olika delar i verksamheten
- vilka fysiska flöden som finns i verksamheten
- vilka informationssystem som finns i verksamheten samt vilka ägare dessa har
- etc.

Man kan dessutom behöva beskriva detta med avseende på tidsaspekten. Detta för att:

- a) kunna beskriva hur saker och ting ser ut i dagsläget
- b) hur man vill att de skall se ut i en framtid (inriktning på förändringen).

Aspekter som har med teknik och maskinresurser måste kunna hanteras för att ingå i en konstruktiv process.

3.1.5 Modellering

En modell är en abstraktion av det man uppfattar som verklighet. Den är dessutom förenklad genom att man tar ut (väljer ut) ett antal faktorer som man anser vara relevanta. Antalet faktorer kan ofta inte bli för stort, eftersom man då förlorar överblicken.

Förenklandet kan ha risker. Dels kan det vara för få aspekter som tas med, dels kan det vara *fel* aspekter som är utvalda.

Modellera behöver man bli för att kunna beskriva ett nuvarande läge i verksamheten och för att kunna beskriva ett önskat läge.

Att beskriva nuvarande läge i en verksamhet blir ofta en förklaringsmodell som används för att "berörda aktörer" i en verksamhet skall komma fram till en gemensam uppfattning om hur saker och ting förhåller sig. Ofta behöver man också bygga en modell av en lösning av något man tror behövs i verksamheten. Detta i arbete med att söka sig fram mot en "specifikation" av vad man uppfattar som en lösning. Det kan krävas mycket eftergranskning kring en sådan modell för att avgöra om det man avser att skapa har rätt egenskaper (uttryckta i specifikationen).

Det blir ofta för dyrt att skapa lösningen konkret och i full skala för att avgöra om det var rätt lösning. Modeller kan beskriva lösningen till mindre kostnad.

3.1.6 Modelleringsbegrepp och deras samband

Eftersom man ofta måste beskriva ett stort antal aspekter av en verksamhet för att förändra denna, måste man definiera och använda sig ett ganska stort antal modelleringsbegrepp. Dessa modelleringsbegrepp har samband.

Att beskriva vilka modelleringsbegrepp man ser och vilka samband de har kan uttryckas i en s.k. metamodell. Den är bra för att göra klart vilka modelleringsbegrepp som finns och vad de betyder.

3.1.7 Att gruppera modelleringsbegrepp

För att överhuvudtaget kunna överblicka alla intressanta aspekter av en verksamhet blir man förr eller senare tvungen att strukturera och gruppera dessa aspekter i olika modeller på så sätt att dessa blir intressanta att studera ur en viss synvinkel eller sammanhang. Vissa aspekter förutsätter dessutom vissa andra för att överhuvudtaget kunna uppträda. För att man, t ex skall kunna tala om begreppet relation så måste det finnas något att relatera.

En sådan grupp av beskrivande aspekter låter man utgöra en modelleringsteknik eller beskrivningsteknik.

Utöver att man grupperar modelleringsbegrepp i modelleringstekniker, kan det finnas behov av samband mellan modelleringstekniker. (Sambandsaspekter mellan olika delar i olika beskrivningstekniker). Dessa kan vara mycket viktiga kopplingar som en metod kan ta sikte på.

Exempel kan vara att "objekt" skapas i "verksamhetsfunktion", eller att "attribut" tilldelas värde i en "rutinaktivitet". Dessa samband är också en typ av modelleringsbegrepp.

3.1.8 Hur uppstår en metod?

Om en person eller en grupp av personer har ideer, eller har erfårit fördelar med att genomföra arbetet på ett visst sätt och detta har givit positiva effekter på ett eller annat sätt, vill man ofta förmedla information om detta arbetssätt till andra. Andra människor, inom eller utom organisationen, kan ju ha för avsikt att genomföra liknande arbetsprocesser.

En metod har börjat se dagens ljus.

Att inte använda en metod är att genomföra arbetet helt planlöst eller att använda sig av en "trial and error"-ansats, vilket innebär att man försöker på en mängd olika sätt tills man tycker att "det går vägen".

Ett dylikt arbetssätt kan antas ta enorma resurser i anspråk, då varje projekt som arbetar med förändring måste hitta sina egna sätt att lösa uppgifterna - även om det i andra projekt har uppstått kunskap om sätt att effektivt genomföra liknande uppgifter. En metod är ju systematiserad kunskap om att genomföra en arbetsprocess. (För ytterligare kriterier se nedan.)

En metodbeskrivning kan finnas tillgänglig i verksamheterna för olika grupper att använda sig av när man kommer i projektsituationer som metoden behandlar. Metoden blir kommunicerbar. En bra och väl beskriven metod kan vara ett bra "kapital" bli när nya personer kommer in i organisationen.

En metod talar ofta om att vissa modeller skall skapas. Metoden beskriver ofta hur dessa skall handhas och vad man skall göra för att uppnå visst resultat, t ex kvalitetsresultat. Modellererna är ofta ett sätt att se på "verkligheten" tills man har uppnått "det konkreta" i verksamheten.

Att modellera är således ett sätt att överhuvudtaget kunna gripa över ett stort antal samverkande faktorer.

3.1.9 Kriterier för att det skall röra sig om en metod

Att arbeta med en metod kan därför sägas vara sättet att ta sig fram, på ett strukturerat sätt, i en arbetsprocess till ett visst mål.

Ett högre krav på definitionen av en metod kan vara att det dessutom också skall finnas utsagor redovisade om *varför* man skall arbeta på ett visst sätt. Således utsagor om godheten av att arbeta på visst sätt och effekterna av detta. Dessa bör vara uttryckta som antaganden.

Läsaren till metoden kan då ta ställning till lämpligheten av metoden och han kan dessutom lära av att genomföra arbetet med hjälp av metoden.

En metod innehåller ofta ett antal arbetssteg som har ett visst förhållande till varandra. En del kan vara sekvensiella, andra parallella.

Om metoden också kan ge svar på *varför* kan tillämparen lättare ta ställning till situationsanspassning av metoden, dvs ta ställning till om den situation som man har för handen täcks av de situationer som metoden tar upp.

3.2 Framtagning av modeller

3.2.1 Funktions- och datamodellering

Inom systemutveckling är det vanligt att arbeta med modeller av olika slag. En modell är ju en förenkling av en verklighet (det som modelleras - ofta kallat objektsystemet) gjord ur ett visst perspektiv, ett visst syfte.

Olika modeller beskriver olika aspekter av objektsystemet. Vanligt i de tidiga faserna av en utvecklingsprocess är att utnyttja dels någon form av "funktionsmodellering", där viktiga och relevanta funktioner i verksamheten analyseras, funktioner som kommer att vara berörda av det framtida informationssystemet. De olika funktionernas samband kan också beskrivas. Sambanden kan vara av informations- och/eller materiell karaktär.

Vanligt numer är också att objektsystemet modelleras ur informations-, datasynpunkt så att företeelser och regler i en verksamhet modelleras i form av objekt, relationer mellan objekt samt attribut för objekt och relationer.

De olika modellerna, funktionsmodellerna och de konceptuella modellerna, fångar alltså olika aspekter av den verklighet där det blivande systemet skall verka.

I olika ansatser kan de olika inriktningarna betonas mer eller mindre dvs en ansats prioriterar datamodellering och tonar ner funktionsmodelleringen och vice versa.

En metod som prioriterar datamodelleringen är IRM-tanken (Information Resource Management). Där ligger som grundläggande princip, att systemen bör byggas utifrån en stabil grund, som ej så lätt förändras med tiden. Uppgiften är alltså att hitta de basobjekt i verksamheten som existerar under lång tid och där förändringar snarare berör egenskaper (attribut) för dessa objekt. Krav på information som olika delar av en verksamhet kan ha är betydligt mer utsatta för förändringar. De kan variera starkt över tiden.

Andra skolor betonar starkare funktionsmodelleringens roll, att avgränsa och analysera funktioner i verksamheten, dess uppgifter och ansvar, beslut som tas i funktionerna på grundval härav, samt resulterande krav på informationsförsörjning.

Man kan alltså bekänna sig till och hävda den ena eller andra skolan och respektive ansats fördelar. En annan väg, som har provats i ett antal företag och organisationer, är att erkänna att båda modellerna behövs och att de i samspel kan uppvisa en del synergieffekter.

Funktionsmodellering studerar och avgränsar problem, verksamhet och informationsbehov och avgör vad den konceptuella modellen skall omfatta. Den ger också en bakgrund till att tala om delmodeller, lokala konceptuella modeller eller lokala modeller på olika nivåer ("vyer").

Konceptuell modellering upprättar en vokabulär så att de funktionsinriktade modellerna med resultat, kan diskuteras, beskrivas och analyseras på ett enhetligt, tydligt och begripligt sätt för alla. De konceptuella modellerna definierar (mer eller mindre omfattande beroende på modelleringsansats) vad informationssystemet skall handla om.

Detta är en sida av systemutveckling - att tillgodogöra sig synergieffekter av "parallell" modellering av olika typer inom faser av en systemutveckling.

4.0 Varför modellera?

4.1 Användningsområden för modellering

Det finns ett stort antal användningsområden för konceptuell modellering. Nedan presenteras några verksamheter där modellering är ett kraftfullt hjälpmedel.

Verksamhetsanalys på övergripande nivå.

Inom verksamhetsmodellering beskrivs ofta dels vilka verksamhetsfunktioner som är aktuella, men också vilka viktiga objekt som finns eller kommer att finnas i verksamheten, och omkring vilka information behöver kunna hanteras.

För verksamhetsutveckling i form av affärsutveckling stödjer konceptuell modellering förståelse av befintliga begrepp såväl som utveckling av nya begrepp. För organisationsutveckling och administrativ utveckling har man stor nytta av att med modellens hjälp kunna identifiera arbetsuppgifter och ansvarsområden.

Förutom mer datorberoende typer av konceptuell modellering är det aktuellt med modellering av funktioner och informationsflöden, för detta ändamål.

Informationsanalys.

Inom systemutveckling bör de informationsbehov som identifierats i en verksamhet brytas ner i detalj så att man kan se exakt hur denna information är uppbyggd, vilka verksamhetsobjekt informationen refererar till och hur informationen skall representeras. Modellering kan användas till planering såväl som till utveckling och förvaltning av informationssystem.

Datamässiga avgöranden och design avseende hur önskad information skall vara representerad i de informationssystem som byggs.

Således vilka 'värdemängder' eller vilka domäner som finns eller behöver kunna användas för att uttrycka information, kan specificeras med datamodellering.

Beskrivning av hur data skall ligga i olika databaser.

Hur data skall distribueras på lämpligt sätt för att bli kunna vara effektiva att hitta och uppdatera i enlighet med de prestandakrav som ställs mot aktuellt system ifråga om svarstider och tillgänglighet.

På denna datorberoende nivå kan databasmodellering utnyttjas förutom mer semantisk datamodellering.

4.2 Krav och önskemål på konceptuell modellering

Konceptuell modellering bör utgöra:

1. Ett hjälpmedel gemensamt för slutanvändare, dataadministratörer och systemutvecklare dvs ett och samma hjälpmedel för utveckling, konstruktion, drift och användning av systemet.
2. Ett hjälpmedel för storskalig informationshantering dvs för att överblicka data på alla nivåer i mycket stora organisationer.
3. Ett hjälpmedel som kan medverka till ingenjörsmässigt, standardiserat och likformigt underhåll dvs ett industriellt sätt att se på informationshantering och systemutveckling.
4. Ett hjälpmedel som utgör central filosofi i ett IH-verktyg - inget glapp mellan analys och konstruktion.
5. Ett hjälpmedel som är uppbyggt på ett sådant sätt att det är pedagogiskt och självinstruerande för att söka vidare i ett datalager; välj först ett adekvat datalager och sök sedan djupare i detta.
6. Ett hjälpmedel som gör det möjligt att helt skilja logisk struktur från fysisk struktur av data. Överblickbarhetskrav kan då ställas på datalagrets logiska struktur, medan effektivitetskrav kan ställa på datalagrets fysiska struktur. Med andra ord - ingen kompromiss mellan överblickbarhetskrav och effektivitetskrav på datalagret.
7. Ett hjälpmedel (eller en del i ett sådant) för att kunna integrera system och därigenom få förutsättningar för att distribuera kvalitetsdeklarerade data.
8. Ett hjälpmedel som bildar grunden i en datakatalog.
9. Ett hjälpmedel som ger stöd för att anpassa företagets informationsbehandling vid omstrukturering av verksamheten (från central till decentral organisation och vice versa).

4.3 Syfte med konceptuell modellering

4.3.1 Tolkning och modell

Att göra en modell innebär många saker. För det första skiljer man på sakområdet (det som modelleras) och modellen. Modellen är en abstraktion och kategorisering av det som modelleras och detta i sin tur har styrts av hur väl man förstår sakområdet och det bakomliggande syftet eller vad man skall använda modellen till. En viktig egenskap hos modellen är att den dokumenterar överenskommelser och ett gemensamt synsätt.

Eftersom modellen skall representera ett sakområde så måste den utgå från objekt och företeelser som kan identifieras, avgränsas och definieras ("Objekt-modellering"). Sakområdet är inte objektivt givet utan vi vet att vår kunskap om sakområdet påverkar vad vi ser och hur vi modellerar.

Till varje modell hör en tolkning av modellen t ex namn i modellen kopplas till företeelser och objekt i sakområdet. Tolkningsföreskrifter kan finnas dokumenterade vid sidan av modellen.

Modellens stora värde ligger i att den ger en struktur som utsäger inom sakområdet kan hängas upp på. Strukturen bygger på en kunskap om sakområdet och är därför ytterst värdefull.

Vid processtyrning är det självklart att modellen är grunden för alla tekniska lösningar. Annars skulle ej processtyrningen fungera exakt som man vill. För administrativa sakområden har det hittills inte varit lika självklart att utgå från en dokumenterad kunskap om sakområdet i form av en modell. Mindre formella och approximativa metoder har ofta använts.

Varje datoriserad lösning bygger dock på en modell och den praktiska användningen bygger på en tolkning av samma modell.

4.3.2 Resurs och system

Hur stor del av ett sakområde skall modelleras? Kan det göras på olika sätt (olika urval, olika detaljering, etc)?

Detta är grundläggande frågor som varit upphov till dagens problem med att integrera de system som utvecklas på olika avdelningar i ett företag.

Genom urval kan många kombinationer uppstå som synbart är nya modeller. De bygger dock på en gemensam större modell för hela sakområdet (även om den ej finns konstruerad) som "försörjer" olika möjliga vyer.

Att skapa ordning och reda mellan ett antal system kräver att vi går tillbaka till verksamheten och en modell av denna. Detta bygger på två viktiga insikter, dels att verksamhetens modell är viktig för förståelse av ett system, dels att verksamhetens modell kan användas för att integrera flera system (gemensam referensram).

4.3.3 Begrepp och data

Datamodellering omfattar den del av en verksamhetsmodell som behövs för att beskriva begrepp och data.

Begrepp skapas i en konceptuell modell och står för företeelser eller objekt i sakområdet. Data är informationsbärare för begrepp. För att tolka data behövs tolkningsinstruktioner eller beskrivning av data. En definition av data innehåller förklaring av det begrepp som data skall representera.

Den konceptuella modellen syftar både till att skapa överblick över begrepp och att skapa entydighet mellan begrepp. Begrepp förekommer i "sammanhang" och måste alltid kvalificeras med detta sammanhang t ex begrepp som används för att uttrycka egenskaper hos ett objekt.

Begrepp konstrueras ej isolerade från varandra. Tvärtom skall begreppen kunna stödja varandra. Man kan skilja på generiska begrepp (typ datum) och applikationsbegrepp (typ leveransdatum). De generiska begreppen bildar en bas vid definitionen av applikationsbegrepp.

De minsta dataenheter som definieras - dataelementtyper - kan förtecknas och beskrivas i en dataelementtypförteckning. Hur data används tillsammans kan förtecknas och beskrivas i en meddelandetyppförteckning och en meddelandesyntax. Hur begrepp hör ihop med objekt i verksamheten visas i den konceptuella modellen.

4.3.4 Datamodellering

En ADB-verksamhet brukar delas upp i olika delar av praktiska skäl. När man vill upptäcka hur allt fungerar spelar dessa gränser mindre roll. Helheten och de grundläggande problemen vid datorisering är det som fångar intresset.

Utvecklingsmodeller, arbetsfördelning, ansvarsfördelning, projektstyrning, etc har dock en indirekt påverkan på hur vi uppfattar problemet med datorisering. Det är svårt att skilja på dessa två aspekter. Ändå är det nödvändigt att titta på de grundläggande problemen för att pröva hållbarheten hos nya metoder.

Hur ska man uttrycka och representera kunskap om en verksamhet som underlag för datorisering? Hur skall denna kunskap hållas levande? Hur skall man skapa effektiva redskap eller verktyg för att hantera denna kunskap? Hur skall denna kunskap komma till uttryck i tekniska lösningar?

Skillnaden mellan att kartlägga och representera verksamhetsorienterad kunskap och att realisera en lösning med hjälp av ADB-teknik är ett dilemma. Hur väl man lyckas att harmonisera ett verksamhetsorienterat och utvecklingsorienterat angreppssätt har stor betydelse för att underlätta datorisering.

4.3.5 Sammanfattning av syftet.

Det är viktigt att inte blanda ihop datamodellering med organisatoriska åtgärder för att införa dataadministration. Datamodellering som teknik fungerar oavsett hur administrativa regler sätts,

t ex om modellering görs per system, per verksamhetsområde eller för hela verksamheten. Ordning och reda behövs både i enskilda delar och för alla delar tillsammans.

En konceptuell modell kan utnyttjas som en karta över verksamheten och dela upp den i lämpliga "affärsområden" eller göras tillräckligt detaljerad för ett område för att sedan tjäna som konstruktionsunderlag. Att modellen kan

nivåindelas är en väsentlig egenskap liksom att de olika nivåerna kan kopplas samman.

Syftet är att skapa stabila strukturer som data kan hängas upp på och som kan fungera som en bas (ev en tänkt bas) för försörjning av data till system med sitt informationsbehov (en "vy"). Modellens stora fördel är att dess struktur och begrepp är tillräckliga för att svara mot ett sakområde. Det svarar mot sakområdets komplexitet.

I synsättet ligger en stor möjlighet att utveckla data som en resurs. Det är den första utvecklingsmetod som inte är inriktad mot systemutveckling utan mot resursutveckling och mot försörjning av ad-hoc behov.

Metoder för systemutveckling har äntligen blivit något annat än att utveckla isolerade system för att uppfylla tillfälliga behov som låses fast i system. Verksamhetens struktur uttryckt i en konceptuell modell är den stabiliserande gemensamma faktorn. Den konceptuella modellen ger ett disciplinerat sätt att representera denna kunskap.

5.0 Modelleringsansatser & modelleringsspråk

Med en modelleringsansats brukar man mena:

- * ett modelleringsspråk, och
- * en metodik - ett tillvägagångssätt att formulera, framställa och förändra ett schema/en modell

Ett modelleringsspråk består av:

- * en uppsättning modelleringsbegrepp
- * en grafisk notation
- * en verbal/textuell notation som kan vara mer eller mindre strukturerad och formell.

I en grafisk notation, som har uppenbara fördelar ("en bild säger mer än tusen ord"), kan man dock i regel bara uttrycka vissa enkla regler och villkor.

Om man vill uttrycka mer komplicerade och omfattande regler och villkor ökar behovet av en mer "uttrycksfull" verbal/textuell notation i tillägg till den grafiska beskrivningen. En sådan notation kan vara mer eller mindre formell beroende på önskad grad av precision.

5.1 Referensramar och kunskaper

Olika typer av modelleringsansatser kan vara olika lämpliga för olika kategorier av människor, som skall arbeta med modellerna. Olika människor har olika referensramar när det gäller att förstå uttrycksformen för en viss modelleringsteknik.

Det är alltså inte bara fråga om att tala om uttrycksförmåga, man måste dessutom fråga för vem det skall uttryckas.

Det finns ett antal aspekter, som måste beaktas när det gäller att diskutera lämplighet hos modelleringsansatser:

- * Vilken kunskap och erfarenhet har de som skall modellera av att abstrahera och modellera över huvudtaget?
- * Vad behöver uttryckas och vilken uttrycksförmåga behöver modelleringsansatsen därmed ha?

- * Vilka referensramar har de som skall arbeta med modellerna? Vilka typer av modelleringstekniker är man van vid (skalmodeller, symbolmodeller, olika typer av modelleringsbegrepp, etc)?
- * Vilka behov av överblick har man? Hur kan detta balanseras med behovet av att se detaljer?

Om man vill att informationssystemanvändare i ett företag skall vara med och beskriva önskat informationsinnehåll i ett nytt system, så bör de begrepp som används i modelleringstekniken för att beskriva informationsinnehåll vara väl anpassade till det speciella ändamålet och till personerna ifråga, så att man inte tröttnar ut de som medverkar i utformningsarbetet med besvärliga och olämpliga beskrivningssätt.

5.2 Några egenskaper hos modelleringsansatser

Det finns många egenskaper som skulle kunna användas för att göra typindelningar och jämförelser mellan modelleringsansatser. Nedan listas några exempel på grupper av egenskaper.

Egenskaperna är i huvudsak karakteriserande och i sig inte värderande dvs de efterfrågar t ex bara om vissa saker behandlas eller finns med i en modelleringsansats. Kriterier för värdering av modelleringsansatser, dvs som dessutom utsäger att ett utfall för visst kriterium är bättre än ett annat, förutsätter naturligtvis också att de grundläggande värderingar och principer som detta bygger på och motiveras av lyfts fram och redovisas.

1. SYSTEMUTVECKLINGSBEGREPP OCH KOPPLINGAR

Vad säger ansatsen om t ex modelleringsnivåer?

Vilka nivåer - hur relateras de till varandra?

Förhållande till problem- och verksamhetsanalys, motsvarande modeller?

Förhållande till databaskonstruktion?

Finns flera distinkta arbets- och utvecklingssteg för att ta fram en modell?

2. KONSTRUKTIONSMETODIK OCH KVALITETSSTYRNING

Vilka principer rekommenderas för modellkonstruktion: från allmän diskussion om objektsystemet, från specificerade informationskrav, kombination?

Finns metodik för konceptualisering och integration?

Hur valideras modellen?

Hur kontrolleras fullständigheten?

Hur kontrolleras konsistens?

Finns det kriterier för kvalitetsmässigt 'goda' modeller?

Finns koppling till eller metodik för databaskonstruktion?

3. PRAKTISKA ASPEKTER OCH VÄRDERINGAR

Är modelleringsansatsen beroende av tillämpningsområde?

Hur bedöms användarnas (utbildade resp. icke utbildade) möjligheter att medverka, förstå och kontrollera specifikationsarbetet?

Erforderlig utbildningsvolym?

Modelleringsansatsens flexibilitet, inkrementalitet, 'modelleringsfrihet', 'beteckningsfrihet', etc?

Finns datorhjälpmedel?

För vad finns datorstöd?

4. TILLÄMPADE MODELLERINGSBEGREPP

Entitet/entitetstyp eller motsv?

Förhållande/förhållandetyp eller motsv?

Händelse/händelsetyp - dynamiska aspekter?

Attribut: funktionella, flervärda, strukturerade?

Globala attribut, "mängdattribut"?

Distinktion mellan lexikala och icke-lexikala objekt (dvs "names of things" och "things themselves")?

Regler och villkor?

Hantering av dynamiska samband: hur hanteras tidsdimensionen och referens till data vid olika tidpunkter?

5. MODELLERINGSSPRÅKET

Finns formellt definierad syntax för arbete med tillämpade begrepp?

Speciellt: hur definieras regler och villkor?

Finns grafisk notation?

Vad kan uttryckas med den notationen?

Uttrycksfullhet och uttryckskraft?

Det är självklart inte möjligt, och inte syftet, att i denna rapport göra en jämförelse mellan olika modelleringsansatser på alla dessa punkter. Delar av dessa har dock använts implicit och explicit vid det beskrivnings- och sammanfattningarbete som redovisas i kapitlen 9 och 10 i denna rapport.

5.3 Typer av modelleringsansatser

Man kan möjligen diskutera ett antal grova typer av modelleringsansatser:

1. Term(namn-)baserade modelleringstekniker
2. Enkla associativa objekt- och begreppsmodelleringstekniker
3. Binära objektorienterade modelleringsansatser
4. Objektorienterade modelleringsansatser av EAR-typ
5. Avancerade konceptuella modelleringsansatser med språk för definition av statiska och dynamiska regler

5.3.1 Term(namn-)baserade modelleringstekniker

I dessa modelleringstekniker används inte något objektbegrepp. I stället relateras enbart informations- eller dataenheter till varandra.

Ett exempel är att komponentanalys bedrivs. Man bryter ner större informationsenheter till mindre enheter - dataflöden i deldataflöden och vidare till dataelement. Man bryter ner informationsbehov eller informationsmängder i delinformationsbehov/mängder, meddelandetyper, termgrupper och enkla termer.

Den renodlade relationsansatsen kan ses som en exemplifiering av denna typ av modelleringsteknik. Man beskriver relationer mellan dataenheter i form av sk funktionella beroenden. På grundval av dessa kan man sedan dekomponera en "universal"-relation i mindre relationer eller man kan utgående från en uppsättning funktionella beroenden mellan dataenheter/attribut/termer syntetisera fram en mängd relationer för ett databasschema.

5.3.2 Enkla associativa objekt- och begreppsmodelleringstekniker

Dessa modelleringsansatser, som i regel används i tidiga skeden av en systemutvecklingsprocess, analyserar och beskriver samband mellan objekt eller begrepp. Sambanden är semantiska dvs de har en innebörd som ges av de namn man ger dem.

Man gör här i regel ingen distinktion mellan olika kategorier av objekt (entitet, förhållande, händelse) eller skiljer mellan begrepp som refererar till objekt av icke-lexikal eller lexikal typ.

Syftet är att fånga begrepp, att diskutera kring begreppen, att notera begreppen, att vara en "inkörspart" till fortsatt modellering.

Enkla associativa modelleringstekniker hålls medvetet enkla i den meningen att de använder få modelleringsbegrepp och har en enkel notation. Beskrivningssättet är i regel uteslutande grafiskt.

En viktig fråga beträffande dessa modelleringstekniker är huruvida de innehåller eller är kopplade till en vidare konceptuell modellering eller om de stannar vid detta.

En risk som måste poängteras när man stannar på alltför grov och överskådlig nivå är att man kommer till en "koncensus" kring begrepp, en koncensus som vid fortsatt detaljering av modellen inte visar sig hålla. En alltför grov beskrivning kan alltså leda till en "falsk" begreppsgemenskap. Man trodde att man var överens, men i själva verket var man det inte.

5.3.3 Objektorienterade modelleringsansatser

Objektorienterade modelleringsansatser använder modelleringsbegrepp som har kunskapsteoretisk, filosofisk och lingvistisk bakgrund och som inte har någon motsvarighet i datorers sätt att arbeta.

Typiska begrepp är objekt/entitet, attribut/egenskap, förhållande/relation/association och händelse.

Beroende på vilka "delmängder" av de "objektorienterade" modelleringsbegreppen som används, brukar man urskilja:

- * Binära ansatser
- * Entity-Attribute-Relationship (EAR)-ansatser

De binära ansatserna använder sig av begreppen objekt och binärt förhållande (association). Ett binärt förhållande är alltid mellan två objekt. I NIAM, som är ett exempel på en binär ansats, skiljer man mellan lexikala objekt (värden) och icke-lexikala objekt. Ett binärt förhållande kan definieras mellan två objekt av samma slag eller mellan ett icke-lexikalt objekt och ett lexikalt objekt.

EAR-ansatserna använder, som namnet utsäger, begreppen objekt/entitet, attribut och förhållande (relationship). Ett attribut är en namngiven association mellan en entitet, eller ett förhållande, och ett attributvärde. En begränsning som gäller är att attribut skall vara "en-värda" (funktionella). Attributvärdens existens beror på existensen av entiteter och förhållanden.

5.3.4 Avancerade konceptuella modelleringsansatser med regelorientering

I sådana ansatser finns i tillägg till grundläggande modelleringsbegrepp som objekt, attribut och förhållande, också välutvecklade språk för specificering av regler och villkor av olika slag.

Förutom regler av statisk natur, kan ansatserna t ex genom användande av händelse- och tidsbegrepp också uttrycka dynamiska förhållanden.

6.0 Konceptuell modellering – grundbegrepp och operationer

6.1 Grundbegrepp

6.1.1 Objekt

De olika företeelser i en verksamhet som man vill kunna göra olika typer av utsagor om representeras i en konceptuell modell av objekt (entiteter). Objekt kan ha samband - associationer.

Kategorier av objekt

Ofta har en modelleringsansats fler modelleringsbegrepp än så - man kategoriserar objekt i t ex objekt/ entitet, attribut och förhållande/ relation. På engelska: Entity - Attribute - Relationship, vilket leder fram till de s k EAR-modelleringsansatserna.

Om en modelleringsansats använder sig av flera kategorier av objekt uppstår vid modellering genast frågan, vad som skiljer dessa olika kategorier åt . När skall man representera en företeelse som ett objekt, som ett förhållande eller som ett attribut?

Skillnaden definieras ofta med hjälp av regler för objektens existens i modellen. Ett objekt kan ha en av andra objekt oberoende existens i modellen, medan ett förhållande bara kan existera om de objekt som förhållandet gäller mellan redan existerar.

Andra modelleringsansatser, som syftar till att också fånga dynamiska aspekter, kan ha ytterligare kategorier av objekt, t ex händelse.

Inte alla modelleringsansatser som använder samma modelleringsbegrepp, definierar dem på samma sätt. Olika restriktioner kan gälla, t ex måste ett attribut i en EAR-modell vara "funktionellt" dvs det får bara ha ett värde. Vill man t ex representera att en person (entitet) kan ha flera telefonnummer så får man antingen, om det är semantiskt riktigt, introducera flera attribut,

t ex bostadstelefonnummer och arbetstelefonnummer. Om båda är arbetstelefonnummer får man introducera arbetstelefonnummer-1 och arbetstelefonnummer-2 eller kanske t o m introducera telefon som en entitet och representera det med hjälp av förhållanden mellan person-entiteter och telefon-entiteter.

Andra modelleringsansatser tillåter "mångvärda" attribut.

Typer och klasser av objekt

I en konceptuell modell vill man i regel inte representera enskilda objekt och enskilda förhållanden utan istället typer eller klasser av objekt och förhållanden.

Att klassificera objekt och bestämma vilka objekt som skall tillhöra en viss objektclass är en viktig modelleringsaktivitet. Att klassificera innebär att bestämma klass och typ för ett objekt eftersom man brukar säga att en objektclass definieras av en objekttyp.

Ett individuellt objekt sägs vara en instans/ förekomst av en objekttyp alternativt en medlem i en objektclass.

Om typbegreppet inte används explicit görs ändå motsvarande distinktion genom att man pratar om objekt resp objektförekomst.

Objekt och namn på objekt

I många modelleringsansatser skiljer man mellan objekt och namn på objekt. Namn motsvarar här term. 'ABC 378' kan sägas vara ett namn för ett bilregistreringsnummer, '412 99' ett namn för ett postnummer, etc.

Objekten kallas då "icke-lexikala objekt" medan namnen kallas "lexikala objekt".

I NIAM (Nijssen Information Analysis Method) som är ett exempel på en binär, objektorienterad modelleringsansats, används modelleringsbegreppen icke-lexikal objekttyp (NON Lexical Object Type = NOLOT) och lexikal objekttyp (Lexical Object Type = LOT).

I CMOL (Conceptual MOdelling Language) kallas lexikala objekttyper för datatyper, i TelMod-B och DM-FYR för domäner och i Tempora-ERT för värdeklasser.

Komplexa objekt

Komplexa objekt är objekt som har en "inneboende" struktur - som består av andra objekt.

Komplexa objekt kan vara användbara för två syften: dels för att under analysens gång inte behöva göra modellen detaljerad ner till lägsta nivå omedelbart, utan kunna avvakta med att "dekomponera" objekt i den till ett senare skede. På så sätt skulle man kunna tillämpa ett hierarkiskt nedbrytningsförfarande vid modellering.

De kan också vara användbara för att dölja detaljer i den detaljerade modellen vid presentation för olika användar- och intressentkategorier, dels under modelleringens gång och dels för att tjäna som kartor över databasens innehåll när systemet är i drift.

Det krävs dock att semantiken i och notationen för komplexa objekt är väl definierad för att de skall kunna användas på ett stringent sätt och utgöra en användbar "abstraktionsmekanism".

6.1.2 Samband

Samband (relationer, förhållanden) fungerar inte, liksom objekt, som fria beståndsdelar i modeller.

Sambanden knyter ihop objekt antingen direkt eller via andra samband. De kan råda mellan en och samma objekttyp eller mellan två olika objekttyper. Om ett samband involverar fler objekttyper än så, se nedan.

Namn på samband ger dem en betydelse, en semantik. Ibland utnyttjar man det faktum att en relation kan betraktas ur flera perspektiv och anger då namn (och därmed semantik) för varje riktning av relationen.

I grafisk notation anges ett samband ofta med en linje som förbinder objekttyper. Samband namnsätts ofta i båda riktningar och en pil (t ex) på relationslinjen anger läsriktning för namnen.

Olika personer kan vilja betrakta sambandet i olika riktningar varför det generellt är svårt att tala om att ett samband har en riktning som är huvudriktning. Regler för vilket namn som tillhör vilken riktning för ett samband är dock viktiga för läsbarheten av grafen.

"Relationsobjekt/relationship/förhållande"

Ett objekt/entitet skall kunna existera oberoende av andra objekt i modellen. Dvs det skall kunna vara meningsfullt att ett objekt/entitet existerar i modellen även om det inte har några samband till andra objekt.

I en del modelleringsansatser finns speciella begrepp för att modellera företeelser som är beroende av andra objekt för sin existens men som man likväl vill kunna beskriva och ange attribut för. Modelleringsbegrepp som relationsobjekt, relationship type, förhållandetyp används för detta. De har ofta också speciell grafisk representation.

I andra ansatser modellerar man likväl sådana företeelser som objekt, och anger genom specifikation av en existensregel att dess existens är beroende av andra objekt. Här framgår alltså inte lika tydligt och omedelbart, som om speciellt begrepp med speciell symbol används, sådant existensberoende direkt i den grafiska beskrivningen.

Förhållanden behöver dock inte alltid vara binära, utan förhållanden kan gälla mellan fler än 2 objekt, mellan 3, 4 eller kanske mellan ett variabelt antal objekt. Så länge som förhållanden gäller mellan ett konstant antal objekt, kan de speciella modelleringsbegreppen av typ relationsobjekt/förhållandetyp användas. Om däremot antalet är variabelt måste förhållandet betraktas som ett objekt/entitet vars existens är beroende av ett variabelt antal objekt/entiteter av andra typer.

Om speciell symbol (av nodtyp) används för sådana relationsobjekt /förhållandetyper, kan de associeras till de objekt/entitetstyper de är beroende av med sambandslinjer. Om de istället symboliseras med relationslinjer uppstår ett

problem om förhållandena är av högre ordning än 2. Relationslinjer måste då dras till relationslinjer, vilket innebär att det blir omöjligt att avgöra om förhållandet gäller mellan de, säg, 3 objekt/entitetstyper som återfinns i de båda relationslinjernas ändpunkter, eller om förhållandet är ett förhållande mellan en objekt/entitetstyp och en annan förhållandetyp.

6.1.3 Regler

Vilka olika typer av regler kan man vilja urskilja i en konceptuell modell och hur kan man beskriva dem?

En del regler byggs in i de använda modelleringsbegreppen. En del regler kan beskrivas i den grafiska notationen. Andra regler beskrivs i annan verbal - informell eller formell form.

En del regler kan vara lokala i den meningen att de gäller och kan knytas till en viss objekttyp

t ex, andra regler återigen är av mer global natur - de involverar fler objekttyper och samband som kan förekomma mellan dem.

En uppdelning av regler kan göras i restriktionsregler och härledningsregler. Restriktionsregler anger vilka begränsningar som skall gälla för modellen och bildar grunden för att kunna avgöra om ett påstående om objektsystemet är korrekt eller rimligt.

Restriktionsregler som förekommer i de flesta modelleringsansatser och som ofta ingår i den grafiska beskrivningen är:

- * avbildningsrestriktioner för relationer/attribut
- * optionalitet el totalitet/partialitet för relationer/attribut

Avbildningsrestriktioner talar om hur många förekomster som kan involveras i ett samband. Ett minimumantal och ett maximumantal brukar anges.

En total relation involverar alla förekomster av det som relationen avser till skillnad från om relationen är partiell (då minimumantalet som avbildas kan vara 0).

Andra restriktionsregler kan begränsa tillåtna värden på attribut. Sådana regler kan vara fristående eller kopplade till modelleringsbegreppet domän och definiera exempelvis format och intervall för värden.

Härledningsregler syftar till att beskriva hur ett påstående om ett objekt, om ett attribut etc, kan härledas ur andra påståenden. Hur härleds t ex ett attributs värde? Vilka är medlemmarna i en härledbar entitetsklass? Med reglernas hjälp kan t ex hanteringen av värden bli ekonomisk. En härledningsregel som får fram nettolön givet bruttolön och skatt kan t ex medge att inte redundant information om lön lagras:

NETTOLÖN = BRUTTOLÖN - SKATT

Regler på formen OM...SÅ... (sk. produktionsregler) lämpar sig för härledning. Mer eller mindre utsvävningar kan göras av villkorsdel och slutsatsdel som ingår i regeln. T ex:

OM det finns X personer anställda i ledningen och Y personer anställda i övrigt
SÅ är den totala mängden av anställda X+Y.

Dynamiska regler används alltför sällan i modelleringsansatser. Det är svårt att beskriva verkligheten på ett rättvisande sätt om tidsaspekter och förändring utesluts. Att beskriva en föränderlig bild är dock mycket svårare än att fånga en statisk ögonblicksbild. Strävan vid utformandet av konceptuella scheman bör enligt ISOs principer vara att fånga en stabil bild men också att göra beskrivningar av dynamiska aspekter.

Exempel på tidsbundna regler är inträffandevillkor och händelseregler men även mer generella verksamhetsregler. Livscyklar för objekt, hur de skapas och tas bort, hör också hit. "Händelse" förekommer också som ett modelleringsbegrepp som inrymmer tidsaspekt.

6.2 Operationer

Klassificering

Klassificering kan betecknas som en operation på förekomster av objekt som ger klasser av objekt som resultat. Inom konceptuell modellering finns entitetsklasser, relationsklasser men även värdeklasser och ibland händelseklasser. Plockar man ut en förekomst ur en klass kallas operationen instansiering.

Generalisering/specialisering

Generalisering innebär att man, genom abstraktion, bildar mer generella begrepp. De samband som gäller mellan de speciella begreppen och det generella begreppet kallas generiska samband eller ISA- (is a/är ett)-samband. Motsatsen till generalisering kallas specialisering.

Om man säger att begrepp, eller generiska objekt, karakteriserar objekttyper, kan vi som exempel säga att objekttypen PERSON är en generalisering av objekttyperna MAN och KVINNA, alternativt att MAN och KVINNA är specialiseringar av PERSON.

I en generisk hierarki ärvs egenskaper och attribut som tillhör de generiskt överordnade objekten av de specialiserade objekten. Har personer ett personnummer så har också män och kvinnor personnummer.

Generiska samband används i många modelleringsansatser. Terminologin varierar men den grundläggande andemeningen är densamma. T ex talar man

ibland om superobjekt - subobjekt. Så t ex är i exemplet ovan PERSON ett superobjekt och MAN och KVINNA subobjekt.

Aggregering

Aggregering innebär att ett antal attribut aggregeras eller grupperas till ett sammansatt objekt. Attributen ses som komponenter eller delar till objektet. Delarna kan återfås med s k dekomponering. PERSON kan t ex ses som ett aggregat av namn, adress, telefon, etc.

Association

En annan operation är association. Det innebär att bilda ett mängdobjekt från en samling objekt som utgör dess medlemmar. T ex kan en samling fotbollsspelare associeras till ett fotbollslag som är ett mängdobjekt.

7.0 En nivåindelad modelleringsteknik

En viss modelleringsansats har ofta ett mindre antal viktiga kärnbegrepp omkring vilka hela modelleringstekniken är byggd. Det är ofta de grundläggande kärnbegreppen, som antas avgöra hur bra ansatsen är för olika användningsområden. Om man skall lära känna en modelleringsansats så är det ofta en god idé att verkligen lära känna dess viktiga modelleringsbegrepp för att förstå hela modelleringsansatsen och dess potentialer.

Vi kan tala om ett antal aspekter hos en modelleringsansats eller en beskrivningsteknik:

- * Antalet modelleringsbegrepp, mängden samband mellan modelleringsbegrepp
- * Exaktheten i definitionen av varje begrepp, inklusive utsagor avseende varför det är viktigt att utnyttja dem för beskrivning och på vilket sätt. Detta speciellt för grundbegreppen i ansatsen.
- * De facetter som finns av de grundbegrepp som modelleringstekniken tillhandahåller inom ramen för de aspekter som en modelleringsteknik verkligen försöker sikta in sig på. (Exempel är super/subrelationer och komplexa objekt - båda för grundbegreppet 'objekt'.)

Förutom att en ansats har en viss uppsättning grundmodelleringsbegrepp, som kan vara ganska olika mellan ett olika ansatser för modellering, så kan antalet övriga begrepp som man sedan bygger på med variera. Dessutom kan djupet i definitionen variera starkt för de olika begreppen.

Mindre utförligt definierade begrepp kan ofta leda till icke önskade tolkningar av begreppen samt missförstånd av hur man skall använda dem. Ett till synes enkelt begrepp kan vid användning i litet mer komplexa modelleringssituationer blir svåransvärt och leda till tveksamheter avseende hur det skall användas. Exempelvis hur bestämningar av super/sub-uttrycksmöjligheten för objekt skall användas. Vad kan i det fallet sägas om ömsesidig uteslutning, uttömmande specialisering osv.

En exakt definierad modelleringsansats kan ofta uppfattas som krånglig och svår att komma in i. Likaså kan en ansats, som har en mångfald av olika begrepp att använda sig av uppfattas som svår att använda, eftersom det finns så mycket att välja bland.

En modelleringsansats vars olika modelleringsbegrepp är exakt definierade och där det finns ett antal begrepp att använda sig av upplevs som mer kraftfull. Det

kan vara väldigt viktigt när det råder flera komplexa samband och förhållanden i verkligheten.

Det finns dessutom ett antal olika personer (intressentgrupper) i en verksamhet som har anledning att bli inblandade i arbetet med att ta fram eller att ta del av modeller. Beroende på vad för typ av intressenter dessa personer är vill de kanske arbeta med olika modeller. T ex kan en företagsledning bli irriterad om det finns en massa detaljer i modellerna som inte är relevanta för den intressentgrupp som man representerar.

Om modellerna för en viss modelleringsansats därför omfattar en mängd detaljer och dessa alltid visas för samtliga intressenter kan modelleringsansatsen få ett rykte om sig att vara svår. För vissa intressenter till modellerna måste det ju dock finnas stor detaljering i modellerna för att dessa skall bli kvalitativa och för att rätt information skall kunna överföras till en viss annan kategori läsare.

Ett exempel kan vara en databaskonstruktör som skall skapa en databas eller en person som skall skriva ett program för en viss arbetsrutin som har modellerats.

Om man alltså både har kravet att kunna göra enkla översiktliga modeller för vissa kategorier intressenter och samtidigt för andra kunna uttrycka en hög grad av detaljrikedom och exakthet, förefaller det som att man har behov av att kunna visa olika *versioner* av en modell. I dessa versioner används en detaljrikedom och en 'projicering' på vissa typer av modelleringsbegrepp så att modellen är relevant för läsarnas behov av modellen.

Man visar alltså *rätt version* av modellen *för rätt intressentgrupp*.

På detta sätt kan man arbeta med en modelleringsansats, som dels är kraftfull där stor uttrycksfullhet och detaljrikedom krävs, dels ger möjlighet att kunna presentera översiktliga modeller som har exakt rätt detaljeringsdjup och facettering i begreppen så att det passar t ex en företagsledning.

Man kan tala om följande dimensioner av en modell:

- * GRAD AV DETALJERING av varje modelleringsbegrepp, t ex hur mycket attribut och restriktioner för begreppet entitet tas med i modellen.
- * GRAD AV PROJICERING, vilka begrepp har tagits med för modellen och vilka har undertrycks.
- * GRAD AV AGGREGERING, har man slagit samman flera instanser av ett modelleringsbegrepp under andra modelleringsbegrepp, som används speciellt för översiktliga modeller. Exempel: komplexa objekt och 'verkliga, exakta' objekt.

För att kunna använda en modelleringsteknik till olika ändamål och att samma begrepp kan användas så långt som möjligt, kan man tänka sig att dela in

modelleringstekniken i nivåer så att man bara 'ser' de delar som är nödvändiga för att modellera de aspekter som är relevanta för en viss användningssituation.

Ofta är det fråga om att en viss detaljering liksom en viss grad av projicering och aggregering är användbar för en viss användningssituation. Några begrepp upplevs som överflödiga. Nedan följer en förteckning med några tänkbara modelleringsbegrepp som kan användas vid olika tillfällen.

* Användning: Övergripande verksamhetsmodellering

Begrepp: Objekt, attribut, relationer, verksamhetsfunktioner, grova informationssamband, mål, verksamhetsuppgifter.

* Användning: Begreppsmodellering i samband med initial projektstart

Begrepp: Objekt är självskrivna, men ibland är klassificering av dessa i exempelvis attribut, relation och objekt, inte nödvändiga att göra här. Funktion, informationsbehov (ibland grova flöden)

* Användning: Informationsanalys

Begrepp: Objekt, relationer, attribut, domäner, restriktioner på värdemängder, beroenden, härledning, funktioner, flöden, grova processer, begrepp som används i olika processer

* Användning: Förvaltning, dataadministration

Begrepp: Objekt, relationer, attribut, informationssystem, verksamhetsfunktioner, databaser, domäner, härledning, databaser, ansvarsområden, (objekt, relationer, attribut, domäner i olika informationssystem och databaser), etc.

* Användning: Systemutformning och fysisk realisering

Begrepp: Objekt, relationer, sökvägar, kardinalitet, uttags- och uppdateringsfrekvenser. Formell beskrivning av behandlingsregler och kontroller

En slutsats av ovanstående resonemang är att de modelleringstekniker som används bör ha en sådan uppsättning av begrepp med en sådan detaljrikedom att de kan tas fram och användas för ett användningsområde i taget. Man har alltså möjlighet att uttrycka detaljer när man behöver detta, t ex inom ramen för databasdesign.

8.0 Övergång från konceptuell modell till databasmodell

Ett krav på ansatser för konceptuell modellering är att det inte skall finnas "något glapp mellan analys och konstruktion". En modelleringsansats bör alltså inrymma eller i varje fall ha koppling till en metodik för hur man för modellen vidare inom konstruktionssteget.

Konstruktionsprocessen innebär att från en problemorienterad kravspecifikation utforma ett schema för en databas med tillhörande processer. Enligt Bubenko och Lindencrona [1], kan denna process principiellt indelas i två steg: i det första steget görs en övergång till ett preliminärt databasschema strukturerat på det sätt som aktuell databashanterare föreskriver.

Är databashanteraren av relationstyp, skapas ett preliminärt relationsdatabasschema som definierar vilka relationstabeller som är aktuella och deras innehåll. Är databashanteraren av nätverkstyp beskrivs schemat med hjälp av posttyper, settyper osv. Syftet är att "utveckla ett eller ett antal alternativa databasscheman som 'logiskt' ligger så nära det konceptuella schemat som möjligt".

Det andra steget syftar till att ta fram ett optimerat databasschema, där omstruktureringar görs för att ta hänsyn till effektivitetsmässiga aspekter. Detta kan innebära att man får denormalisera, återinföra redundans, definiera index för snabbare åtkomst, etc.

Den detaljerade konceptuella modellen skall utgöra bästa möjliga underlag för detta steg där design- och konstruktionsbeslut av annan art än under specifikationsarbetet tas.

Viktiga frågor att ställa sig när man betraktar en viss modelleringsansats är alltså:

- * Är den konceptuella modellen tillräckligt detaljerad för att utgöra ett gott konstruktionsunderlag. Dvs finns det i denna *alla* uppgifter av "problemorienterad natur", som konstruktören behöver eller måste han/hon tillfoga egna antaganden om sådana
- * Finns det en god och glappfri koppling mellan metodik för de olika stegen. Finns det detaljerade regler för hur uppgifter i det konceptuella schemat skall/kan representeras i aktuellt databasschema

8.2 Från konceptuell modell till relationsdatamodell (tabeller)

När man vill skapa en relationsstruktur i form av tabeller så kan detta ske på alternativa sätt. Det finns dock ett antal regler som är vanliga att använda för att styra denna "transformation". Vissa regler är mer givna, i andra fall kan man göra på ett antal olika sätt och får då bestämma sig för vilken regel som är lämpligast.

Ofta är det så att dessa regler uppfattas på olika sätt i olika praktiska tillämpningar och därför kan det vara lämpligt att fokusera dessa noggrannare. Vi ger ett axplock bland de kanske viktigaste reglerna.

I detta avsnitt anges regler som innebär övergång till "nedbrutna" tabeller som ej tar hänsyn till driftsmässiga förhållanden. I nästa avsnitt (8.2) tas regler upp som rör övergång till en tabell där man tar hänsyn till de driftsmässiga förhållandena. Detta innebär t e x att man denormaliserar genom att dubbellagra i syfte att skapa en driftsmässigt mer effektiv tabell.

Övergång till "nedbruten" (normaliserad) struktur berör bl a reglerna:

- **Mångvärda egenskaper,**
vilket ofta leder till att ytterligare en tabell skapas för den aktuella egenskapen (ex: person har flera telefonnummer).
- **Skapande av speciella kolumnnamn**
(kan behövas om databashanteraren har restriktioner i detta avseende).
- **Representation av super/sub-förhållanden i tabeller.**
Detta kan göras på olika sätt. Subobjektet har ett antal extra egenskaper som måste hanteras på lämpligt sätt. Kunskapen kan läggas i program eller genom att vissa kolumner inte utnyttjas för superobjekten då superobjekt och subobjekt ligger i samma tabell, etc.
- **Strukturerade egenskaper.**
Om man (till äventyrs) har egenskaper som är sammansatta så kan detta exempelvis lösas genom att skapa flera separata kolumner.
- **Ett-till-Ett - relationer.**
Detta kan lösas på olika sätt i den "nedbrutna" tabellstrukturen. Ett sätt är att skapa en särskild kopplingstabell för denna situation. Ett annat sätt är att man kan slå ihop tabellerna. Ett tredje sätt är att använda främmande nycklar.
- **Om man har härledda egenskaper**
i begreppsmodellen, så får man avgöra om dessa skall lagras i egna kolumner eller beräknas när applikationen exekverar (och verksamhets-händelserna börjar inträffa). Dessa beslut kan dock även ha

effektivitetsmässiga implikationer, vilket också då kommer att beröra övergången till den fysiska nivån.

8.3 Från relationsdatamodell till modell med effektivitetsmässiga hänsyn

Har man skapat nedbrutna (normaliserade) tabeller utifrån en begreppsmodell, så kan det vara så att man vill göra ett antal ytterligare justeringar innan man får tabeller som man kan använda för att skapa en stor driftsmässig databas. Dessa justeringar hänför sig mestadels till effektivitetsmässiga aspekter, men kan också ha att göra med restriktioner som en viss databashanterare har.

Effektivitetsmässiga justeringar i tabellstrukturen kan föranledas av att körningen mot databasen inte blir tillräckligt snabb utan man blir tvungen att ändra i sina tabellstrukturer (så att man bildar den fysiska). Ett exempel är att om man använder sig av en tabell mycket ofta och om man vid dessa tillfällen även behöver läsa en kolumn i en annan tabell så kan man välja att dubbellagra denna kolumn.

Några exempel på åtgärder (förändringar i tabellstrukturen) som kan vara aktuella när man övergår från en logisk tabellstruktur till en fysisk är:

- Dubbellagring av kolumn i ytterligare en tabell (för snabbare åtkomst).
- Hopslag av en eller flera tabeller till en, för uppgifter som ofta söks ihop.
- Uppdelning av tabeller.
- Definierade index för tabeller som exempelvis har många förekomster och som ofta söks samt där man har krav på snabb åtkomsttid.

9.0 Presentation av några modelleringsansatser

I detta avsnitt presenteras kortfattat 6 st modelleringsansatser för konceptuell modellering (begrepps-, objekt- och datamodellering). Beskrivningarna av ansatserna syftar till att ge en överblickande orientering om dem - viktiga modelleringsbegrepp och egenskaper i övrigt samt att illustrera dem med hjälp av ett exempel.

För illustration har använts ett genomgående exempel, som har modellerats i de olika ansatserna. Exemplet beskrivs först på svenska. Strävan har varit att låta den språkliga beskrivningen och de grafiska beskrivningarna ligga så nära varandra som möjligt.

De 6 ansatserna är:

CMOL (Conceptual MODELing Language)

NIAM (Nijssens Information Analysis Method)

SASMO (inkl variant PV-SASMO)

DM-FYR

TelMod-B

Tempora-ERT.

Som synes är listan långt ifrån fullständig om man ser till vilka modelleringsansatser som är kända och använda inom praktikfält och forskningsvärld i Sverige. Att få en fullständig täckning har heller inte varit avsikten - av praktiska skäl har ett urval måst göras. De valda ansatserna täcker dock ett visst spektrum - från mer "enkla" språk till mer välutvecklade - även om några av dem är ganska likartade.

De gjorda beskrivningarna följer en och samma mall, som upptar rubrikerna:

Bakgrund

Modelleringsbegrepp

Beskrivningsteknik/notation

Förhållande till andra modelltyper, metodik och verktyg

Litteratur

Kapitlet avslutas med en sammanfattande sammanställning där ansatserna på några punkter jämförs med varandra.

Alla grafiska modeller, med undantag av NIAM-modellen, har upprättats med hjälp av CASE-skalverket Ramatic, utvecklat vid SISU. I Ramatic har i princip en miljö använts för varje ansats. Eftersom de olika miljöerna i nuläget är utvecklade till olika djup i detta verktyg, har i en del fall vissa modifikationer gjorts, t ex med användandet av funktioner för fritt ritande i Ramatic (CMOL). NIAM-grafen är gjord med ritprogrammet McDraw II på Macintosh.

En kursverksamhet för "internutbildning" - exempelbeskrivning

Ett kurstillfälle ges vid ett specifikt kursdatum och avser en speciell kurs. Kurstillfället har ett elevpris och ett kurspris.

En kurslokal, som har en lokaladress och en lokalbeteckning, används vid varje kurstillfälle. Ett kurstillfälle identifieras av den unika kurslokalen ihop med kurs och kursdatum.

En kurs har en identifierande kurskod, kursnamn samt en kursbeskrivning. Kursen behandlar exakt ett ämne som definieras av en ämneskod och det ämnesområde som ämnet tillhör. Ämnesområdet har en specifik områdeskod och en områdesbeskrivning.

Kurserna har inte garanterat behöriga lärare. Lärare är anställda och behöriga för en eller flera kurser. De kan också hålla i flera kurser. En eller flera lärare kan ge ett och samma kurstillfälle.

Elever är också anställda. De kan preliminärt bokas för ett godtyckligt antal kurser. Kurserna kan helt sakna sådana bokningar. Vid ett definitivdatum gör eleven en definitiv bokning av exakt ett kurstillfälle. Bokningen framgår av kurstillfället och eleven som har bokat. Ett kurstillfälle måste ha minst en definitiv bokning. En elev kan ha godtyckligt många avslutade kurser.

En avslutad kurs, som är genomgången av exakt en elev, avser ett kurstillfälle och kan utpekas med hjälp av dessa två. Till en avslutad kurs hör en uppgift om intyg har utfärdats (J/N), ett antal närvarotimmar (angivet i ett timantal) samt ett avslutningsdatum.

En avdelning har ett identifierande avdelningsnummer. Avdelningen har minst en anställd. Varje anställd kan bara tillhöra en avdelning.

Anställda refereras till med ett anställningsnummer. De har exakt ett namn (bestående av ett efternamn och ev. flera tilltalsnamn), personnummer och kön. Telefonnummer kan de ha godtyckligt många och eventuellt dela med andra. Dessutom har de en geografisk placering och en befattningskod.

CMOL - a Conceptual MOdelling Language

Bakgrund

CMOL är ett modelleringsspråk utvecklat inom SYSLAB (Systemutvecklingslaboratoriet), Stockholms universitet. Det utvecklades delvis som en syntes av ett flertal olika andra modelleringsspråk och bl a med ett pedagogiskt syfte.

Modelleringsbegrepp

Grundbegreppen är entitetstyp, förhållandetyp, händelsetyp, datatyp samt attribut/selektor. Entiteter, förhållanden och händelser är sk icke-lexikala objekt, medan värden (förekomster av datatyper) är lexikala objekt. Datatyper kan vara sammansatta av andra datatyper (enkla och/eller sammansatta) och kallas då strukturerade datatyper.

Samband mellan objekttyper kallas attribut (el. selektor). Delmängdsförhållanden (generiska samband) mellan entitetstyper och mellan händelsetyper anges. I ett CMOL-schema definieras de olika typerna genom att olika formella egenskaper anges för dem. "Definitionsmallar" finns föreskrivna för entitets-, förhållande-, händelse- och datatyper.

För en entitetstyp anges t ex namn, delmängder samt eventuella mängdattribut dvs attribut som hör till mängden som sådan. Dessutom anges händelser som inför respektive avlägsnar element i mängden, attribut och vilka attribut eller kombinationer av attribut som unikt identifierar en entitet i mängden. För varje attribut anges avbildningsrestriktioner ("mapping", totalitet/partialitet) för antalet element som är associerade till entiteten via detta attribut och dito för dess invers. Vilka händelser eller enligt vilken regel attributet tilldelas, om attributet kan ändras och i så fall av vilka händelser anges också för attributen.

Beskrivningsteknik/notation

CMOL har en grafisk notation och en textuell/formell notation. I den grafiska notationen anges entitetstyper, förhållandetyper, datatyper samt attributs- och delmängdssamband. Händelsetyper kan illustreras grafiskt men knyts ej till andra objekttyper i grafen. De grafiska symbolerna framgår av figur 1.

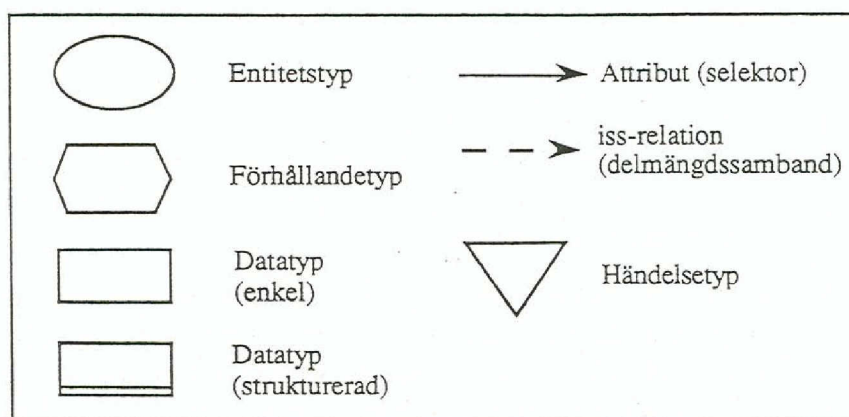


FIG. 1: CMOL - SYMBOLER I DEN GRAFISKA NOTATIONEN

I den formella notationen anges, förutom det som finns grafiskt beskrivet, ytterligare ett antal egenskaper för entitetstyper, händelsetyper, datatyper och attribut, enligt de definitionsmallar som nämnts ovan. Ett exempel på en specifikation av en entitetstyp finns i figur 2.

```

entity-begin ANSTÄLLD;
  existence on-event ANSTÄLL; off-event AVSKEDA;
  generics  $\forall x$  (ANSTÄLLD(x) --> PERSON(x));

  attribute-begin anstnr;
    range ANSTNR; mapping(1:1, T:P);
    inverse anst-nr-för; derivation set-by ANSTÄLL;
  attribute-end

  attribute-begin tot_lön;
    range LÖN; mapping(1:M, T:P); inverse tot-löntagare;
    derivation
      tot_lön(x) = grundlön(x) +
        sum({plön(y) : UPPDRAG(y) & anst(y) = x});
  attribute-end

  identifier anstnr, persnr,
    (telnr, enamn.namn);
entity-end ANSTÄLLD;

```

FIG.2: DELAR AV EN FORMELL BESKRIVNING AV ENTITETSTYPEN ANSTÄLLD

Den formella notationen är välutvecklad och precist definierad i en syntaxbeskrivning. Den bygger på matematisk/logisk notation. I denna kan regler och villkor, som restriktionsregler, härledningsregler, inträffanderegler och inträffandevillkor (för händelser) specificeras.

Selekteringsuttryck som som resultat ger ett, eller en mängd av, objekt kan formuleras och satsen som används i händelser eller procedurer i den konceptuella modellen kan skrivas

(se exempel i figur 3).

```

if not exists(x : ANSTÄLLD(x) & anstnr(x) = a)
then goto alarm else
insert grundlön (x) = 4500;

```

FIG.3: ETT EXEMPEL PÅ EN VILKORSATS I CMOL SOM STYR UTFÖRANDET AV SATSER I EN PROCEDUR

Förhållande till andra modelltyper, metodik och datorstöd

Ett CMOL-schema kan utvecklas genom gradvis detaljering. I tidiga verksamhetsanalysfaser kan ett konceptuellt schema på grov nivå, som upptar de viktigaste objekttyperna och deras samband, utvecklas. Fördjupad

funktionsanalys introducerar successivt alltmer detaljerade informationskrav och det konceptuella schemat kan kompletteras med nya objekttyper, samband och regler. Det är på detaljerad informationsanalysnivå som hela "batteriet" av modelleringsmöjligheter utnyttjas fullt ut.

Lokala konceptuella scheman för informationsbehov kan upprättas och sedan, vid behov, integreras. Att integrera (samordna, harmonisera) scheman är ingen trivial uppgift, men en del allmänna riktlinjer för detta finns beskrivna i ansatsen. Alternativt kan lokala scheman upprättas för varje avgränsad funktion/process och avse funktionens hela "informationssfär" dvs dess in- och utgående informationskrav.

Ett fullständigt och detaljerat konceptuellt schema enl CMOL syftar till att beskriva den formaliserade information som skall omfattas av ett informationssystem samt hur den produceras dvs härleds från "externa" händelser/transaktioner. Det konceptuella schemat är ett "dictionary" för den information och de regler som systemet omfattar.

Implementering av ett konceptuellt schema kan tänkas ske i två steg - en preliminär transformation som överför det konceptuella schemat till ett eller flera alternativa databasschema enligt aktuell databasmodell/typ av DBMS (relationsdatabas, hierarkisk databas, CODASYL/nätverks-databas). Vissa riktlinjer för en sådan transformation till relations- och CODASYL-databas finns.

De flesta regler och villkor i ett CMOL-schema måste implementeras med hjälp av procedurer. I nästa steg sker en omstrukturering i effektiviserande syfte utifrån operationella krav och data (tidskrav, belastningar, volymer, frekvenser, etc).

Datorstöd: MOLOC är ett exekverbart modelleringsspråk som stödjer CMOL och som grundar sig på Prolog. Det är nu också försett med ett grafiskt interface för upprättande av grafer av CMOL-typ.

Ett enkelt datorstöd för den grafiska notationen i CMOL finns inlagt i Ramatic. Stödet är dock ofullständigt och i behov av vidareutveckling.

Litteratur

- [C1] Bubenko, J.& Lindencrona, E.: *Konceptuell modellering - Informationsanalys*, Studentlitteratur, 1984.

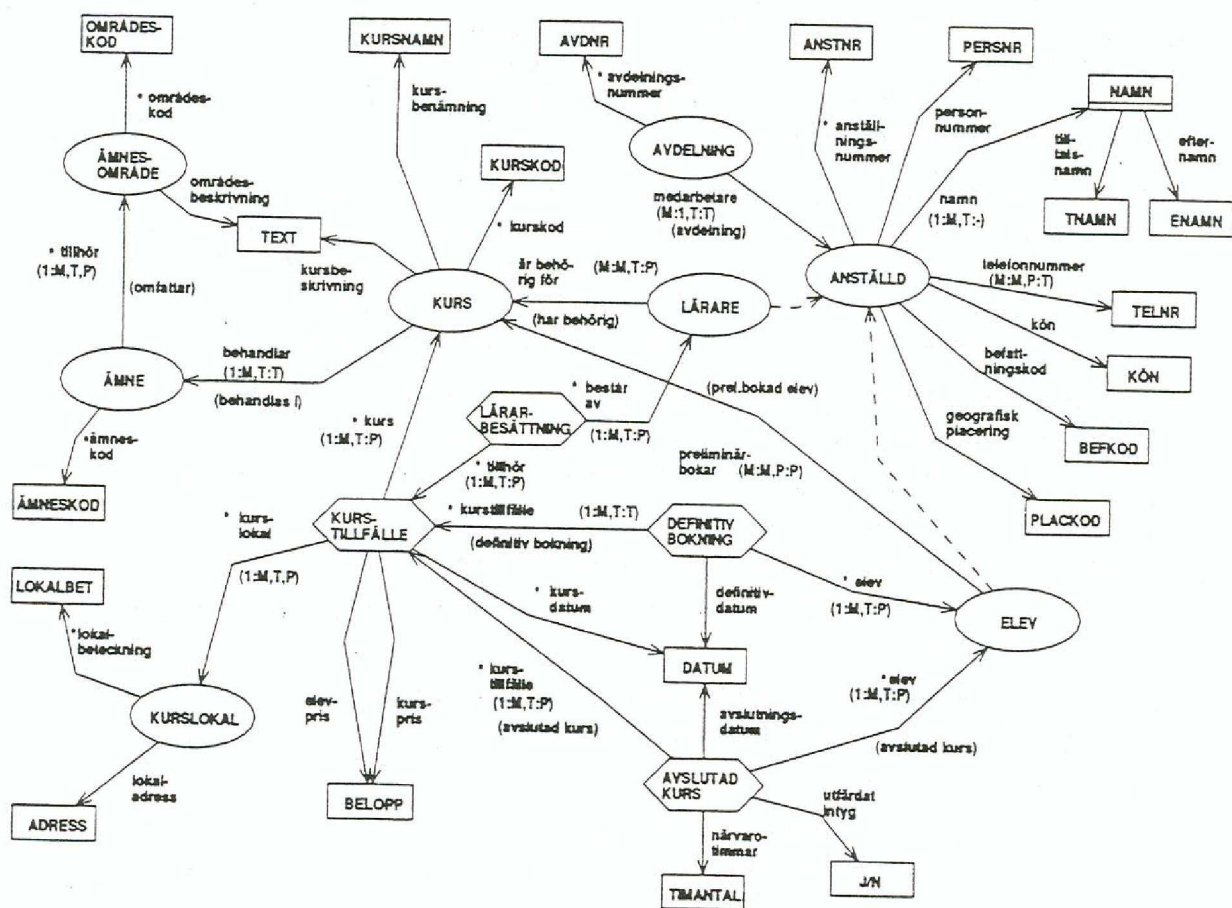


FIG. 4: GRAFISK CMOL-MODELL ÖVER KURSVERKSAMHETENS BEGREPP.

NIAM - Nijssens Information Analysis Method.

Bakgrund

Utvecklad av G.M. Nijssen m fl vid Control Data Belgium, Inc.

Modelleringsbegrepp

I NIAMs konceptuella modellering skiljer man mellan icke-lexikala och lexikala objekt. Lexikaliska objekttyper kallas LOTs (beskriver klasser av lexikala objekt som namn, titel, etc) och icke-lexikala objekttyper kallas NOLOTs (beskriver klasser av icke-lexikala objekt som person, stad, bil, etc.).

En association (binär relation) mellan icke-lexikala objekt kallas IDEA, en association mellan ett icke-lexikalt och ett lexikalt objekt kallas BRIDGE och en association mellan två lexikala objekt kallas PHRASE.

För associationer anges roller i båda riktningar (t ex PERSON bor-i STAD och STAD är-bostadsort-för PERSON). Subtyps-indelning används. Vidare anges ett antal olika typer av s k restriktionregler (se nedan).

Ett konceptuellt schema kallas i NIAM en "conceptual grammar".

I NIAMs funktionsmodellering beskrivs funktioner, informationsflöden (strömmar av meddelanden), informationsbaser (lagringsplatser för meddelanden) och omgivande enheter ("environment"). Funktioner bryts ner i delfunktioner. Procedurbeskrivningar för formaliserbara funktioner ingår i den "konceptuella grammatiken".

Beskrivningsteknik/notation

Det finns en grafisk notation där NOLOTs, LOTs, IDEA TYPEs, BRIDGE TYPEs och subtyp-förhållanden anges.

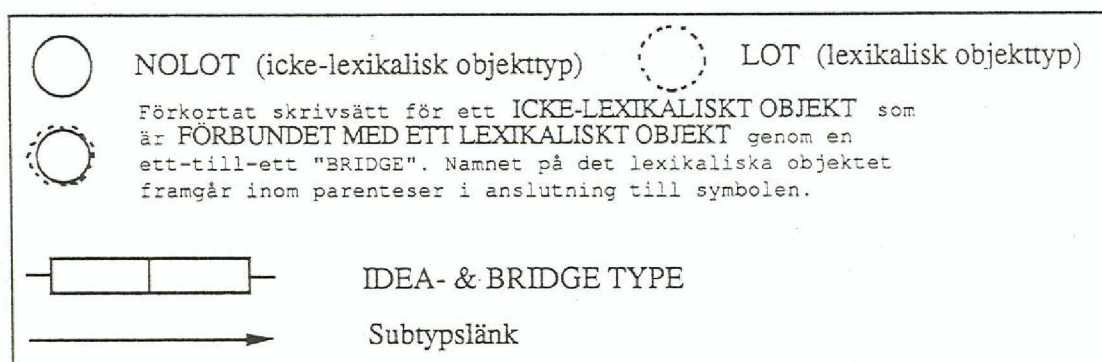


FIG. 5: SYMBOLER I NIAM:S GRAFISKA NOTATION

Det finns också möjligheter att i graferna ange en hel del olika typer av restriktionsregler, t ex identifieringsregler (motsv s k "mapping"-regler dvs 1:1, 1:M, etc), delmängdsregler mellan roller, ömsesidigt uteslutande ("exclusive") subtyper, unikhetsregler (att en kombination av roller unikt identifierar ett icke-lexikalt objekt), etc.

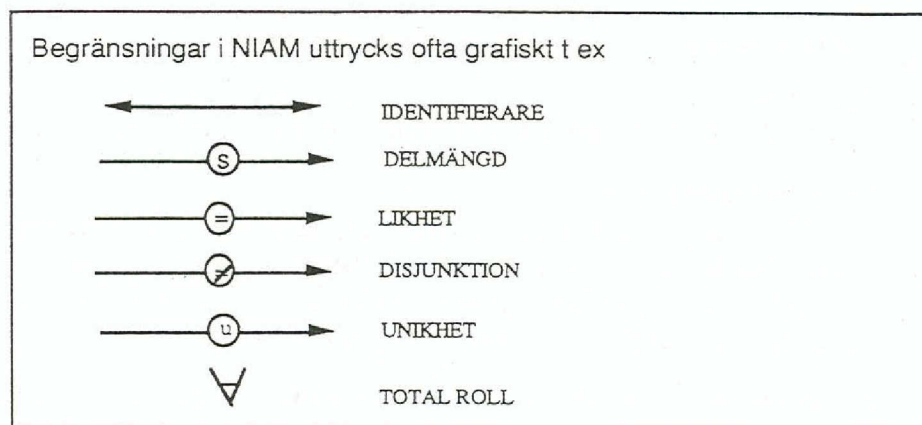


FIG.6: GRAFISK SYMBOLISERING AV REGLER OCH BEGRÄNSNINGAR

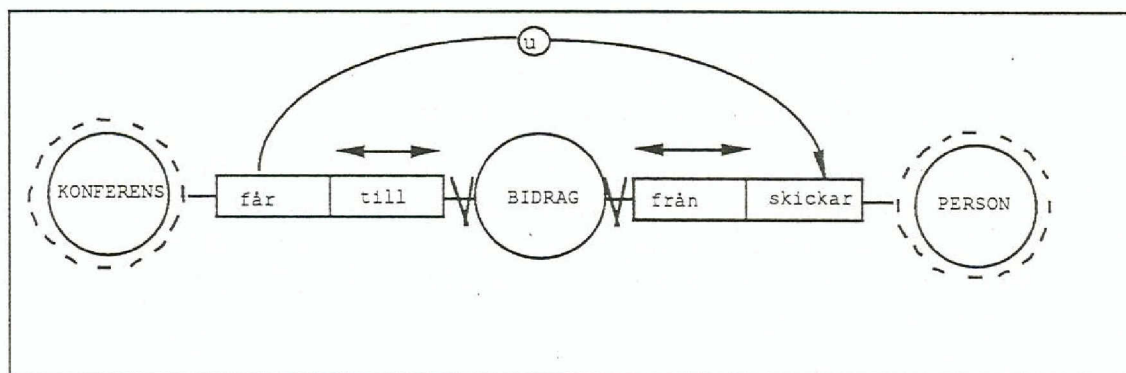


FIG.7: EXEMPEL PÅ ANVÄNDNING AV UNIKHETSREGLN SAMT REGLER FÖR IDENTITET OCH TOTALITET.

Detta innebär att NIAM har en ovanligt rik grafisk notation. Grafema kallas ISDs (informationsstrukturdiagram).

Funktionsmodelleringen beskrivs också i grafisk form. Grafema kallas IFDs (informationsflödesdiagram).

I tillägg till den grafiska notationen finns en formell notation kallad RIDL (Reference and IDEa Language). I detta kan alla IFDs och ISDs specificeras samt ytterligare ett antal regler. Ett exempel återfinns i figur 8. Även procedurer för funktionerna kan specificeras.

```

begin grammar
  add conceptual grammar KURSVERKSAMHET;
  .
  add nolot ANSTÄLLD, KURS, KURSTILLF, LÄRARE ...;
  add nolot ELEV subtype of ANSTÄLLD;
  .
  .
  add lot NAMN, BELOPP, DATUM ...;
  add idea-type LÄRARBEHÖRIGHET
    roles (KURS har behörig and LÄRARE behörig för);
  .
  add bridge-type TIMNÄRVARO
    roles (AVSL KURS närvaro and TIMANTAL tillhö);
  add constraint ...;
end grammar;

```

FIG.8: EXEMPEL PÅ EN "CONCEPTUAL GRAMMAR" I RIDL

Förhållande till andra modelltyper, metodik och datorstöd

NIAM föreskriver i sig ingen speciell systemutvecklingsmodell, utan inriktar sig som namnet säger på informationsanalys.

Gången som föreslås är komprimerat följande:

Lista alla funktioner som informationssystemet avses stödja, bryt ner funktionerna i delfunktioner till en nivå där informationsflödena och bearbetningen i funktionerna kan beskrivas klart. En hierarki av IFD:s blir resultatet. Låt varje informationsflöde ge upphov till ett informationsstrukturdiagram (alltså ett lokalt konceptuellt schema för varje informationsflöde). Beskriv regler och funktioner formellt i RIDL. De lokala schemana överlappar sannolikt - integrera dem till ett schema för informationssystemet. Översiktsskildrar som bara upptar viktiga NOLOTs och IDEA TYPEs kan användas. Datorstöd finns utvecklade. Med hjälp av ISDIS (Information Systems Design and Implementation System - CDC-produkt) kan det konceptuella schemat lagras, uppdateras och kontrolleras. ISDIS kan också generera förslag till databasschema för relationsdatabaser och Codasyl-databaser.

Litteratur

- [N1] *Concepts and Terminology for the Conceptual Schema and the Information Base*, ISO TC97/SC5/WG3, 1982.
- [N2] Verheijen & Van Bekkum: *NIAM: An Information Analysis Method, i Information Systems Design Methodologies: A Comparative Review*, North-Holland, 1982.

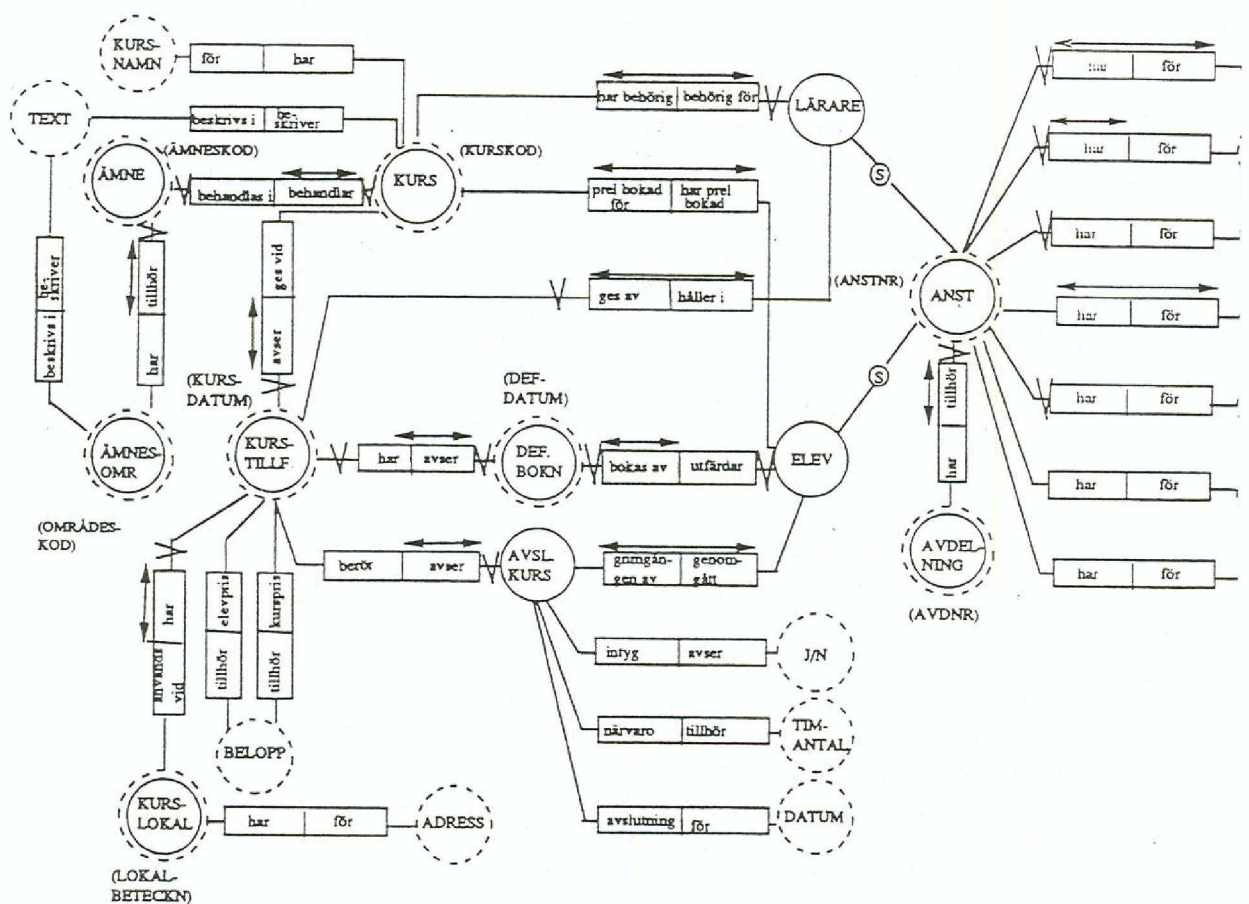


FIG. 9: GRAFISK NIAM-MODELL ÖVER KURSVERKSAMHETENS BEGREPP.

SASMO

Bakgrund

SASMO - datamodellering har ursprung i idéer av Bo Sundgren [S4] och har vidareutvecklats inom SAS och senare IRM Consult. SASMO-modellering ingår som del i IRMA- och SVEA-metodikerna.

Modelleringsbegrepp

Centrala modelleringsbegrepp är objekt och relation. Ett enskilt objekt och en enskild relation kallas objekt förekomst respektive relations förekomst.

Relationer (M:M) kan vara av ordning två eller högre, dvs relatera två eller flera objekt. Relationer av högre ordning än två symboliseras genom att M:M-relationer "vilar på" andra M:M-relationer, och kan, om lämpligt och eget "Id" finns, objektifieras dvs istället modelleras som ett objekt.

Objekt och relationer beskrivs förutom av sina relationer till andra objekt också av dataelement/termer. Dessa dataelement/termer anges ofta direkt i de till objekt och relationer svarande tabellerna (relationstabeller). För objekt anges en eller en kombination av dessa dataelement (om de unikt utpekar förekomster av objektet) som nyckel eller "Id". Övriga dataelement/termer kallas egenskapstermer. Relationer (M:M) lånar sina nycklar från de objekt de relaterar.

En indelning av relationer i så kallade relationstyper görs med avseende på avbildningsrestriktioner (1:1, 1:M, M:M). M:M-relationer namnges med substantiv i singular form. 1:M-relationer bör namnges, med en verbform. Relationerna är oriktade - inverser används i regel inte.

I samband med datamodelleringen exemplifieras objekt- och relations förekomster i form av tabeller. Sju huvudregler för övergång från datamodell till tabeller (s k tabell-precisering) finns angivna

Modelleringsbegreppen är medvetet få och "avskalade" i avsikt att inte komplicera begreppsapparaten. Begrepp som existensberoende (kallat existensregel - tvingande relation), uteslutande relationer, generiska samband genom ram-/rollobjekt finns dock i modelleringsspråket och används i tillämpliga fall.

I PV-SASMO, en variant av SASMO som används inom Volvo Personvagnar har det utvecklats vissa alternativa tolkningar och användningar av modelleringsbegreppen. Så t ex markeras de M:M-relationer som skall bilda tabeller med en asterisk. Beroende på hur denna asterisk placeras när relationer gäller mellan relationer tolkas relationen olika, vilket har konsekvenser när man i ett senare skede "detaljerar" modellen bl a genom objektifiering av relationer. Vidare anges s k existensregler med 0 och 1. Om ett existensberoende också innebär ett identifieringsberoende mellan objekt symboliseras detta med en "tjock" relationslinje.

Beskrivningsteknik/ notation

De använda notationerna är grafisk notation, tabeller med exempel och beskrivande text.

I den grafiska notationen symboliseras objekt av rektanglar och relationer med linjer som förbinder de objekt som relateras.

Avbildningsrestriktioner för relationer dvs max-angivelse symboliseras med "gaffel" på relationslinjen i resp relations ända. Om optionalitet anges noteras detta med hjälp av ettor på relationerna. Ibland kan min-, medel-, max-angivelser noteras i anslutning till relationerna.

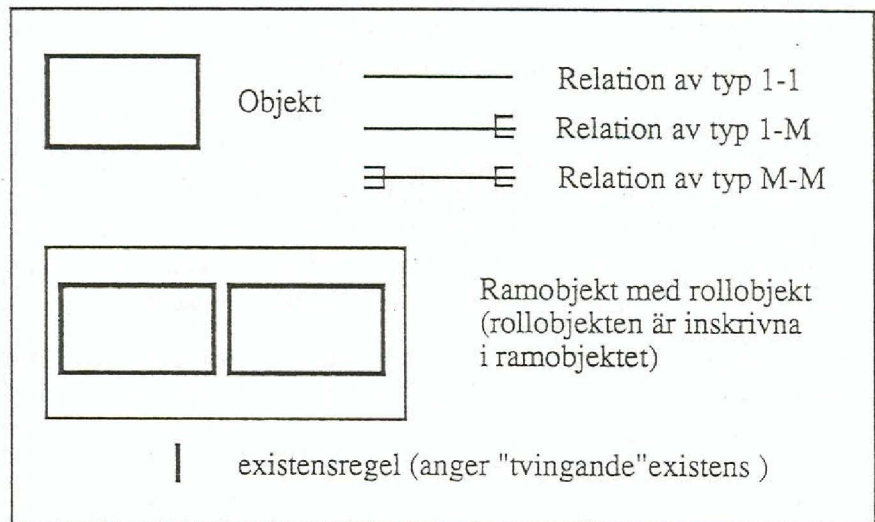


FIG 10: SYMBOLER I SASMO:S GRAFISKA NOTATION

Uteslutande relationer kan indikeras genom att en båge dras över dem.

Generiska samband kan beskrivas genom i varandra inskrivna rektanglar, där det yttre objektet är ramobjekt och de inre är rollobjekt (specialiseringar av det mer generella ramobjektet).

De varianter som som ovan nämnts förekommer i PV-SASMO har följande symbolrepresentation:

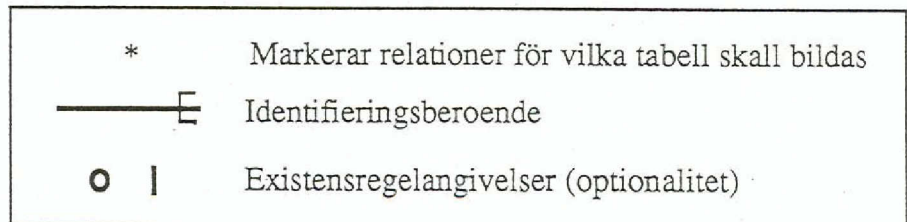


FIG. 11: SKILLNADER I GRAFISK NOTATION I PV-SASMO

I tillägg till den grafiska notationen finns s k objektlistor. I dessa förtecknas de objekt som ingår i modellen och för varje objekt anges nyckel samt görs en definition av objekten i form av beskrivande text i naturligt språk.

I tabellnotationen anges i huvudet på varje kolumn om kolumnen representerar en nyckel (heldragen linje överst), en främmande nyckel (streckad linje överst) eller en egenskap (ingen markering).

Förhållande till andra modelltyper, metodik och datorstöd

Som tidigare nämnts ingår SASMO-ansatsen som en del i IRMA-metodiken (IRMA = Information Resource Management Architecture). IRMA-metodiken syftar till att framställa en "stadsplan" som underlag för planering och utbyggnad av en infrastruktur för en verksamhets informationsbehandling.

IRMA-metodiken innehåller ett antal arbetssteg, vari en datamodell tas fram. Verksamheten indelas i funktioner och funktionerna relateras till datamodellens komponenter i form av en s k IRM-matris. I matrisen framgår användning av och ansvar för data i funktionerna.

IRM-matrisen utgör sedan den karta varifrån en ideal utvecklingsplan tas fram i de efterföljande arbetsstegen. Planen analyseras i anslutning till de kritiska framgångsfaktorer, som utarbetas tillsammans med ledningen.

Arbetet utmynnar i en stadsplaner rapport och förslag till och beslut om fortsatt arbete innehållande aktiviteter som berör etablering av dataadministrationsfunktion, teknologifrågor, säkerhetsaspekter, etc.

SVEA-metodiken används vid utveckling av nya system enligt den stadsplan som tagits fram. I denna används också datamodellering enligt SASMO men nu med inriktning på den aktuella tillämpningen. I samspel med rutinskissning beskrivs tabellvyer och genomförs en "in/utdataprecisering". Datamodellen preciseras.

Enligt vår kännedom finns i nuläget inget datorstöd som på ett nära sätt helt stödjer alla delar av SASMO-tekniken.

Datorstöd för varianten PV-SASMO finns dock skapat i Ramatic.

Litteratur

- [S1] Axelsson, L. & Ortman, L.: *Utvecklingshandboken SVEA. Modell och metoder för administrativt utvecklingsarbete*, Kommundata, 1985
- [S2] *Databasanpassning*. IRM Consult.
- [S3] *Datamodellering - begrepp och definitioner*. IRM Consult.
- [S4] IRM Consult säljer stadsplaneringsidén, artikel i *SISU informa*,
Nr 88/3-4.
- [S5] Sundgren, B.: *Databaser och datamodeller*, Studentlitteratur, 1981.

9 Presentation av några modelleringsansat

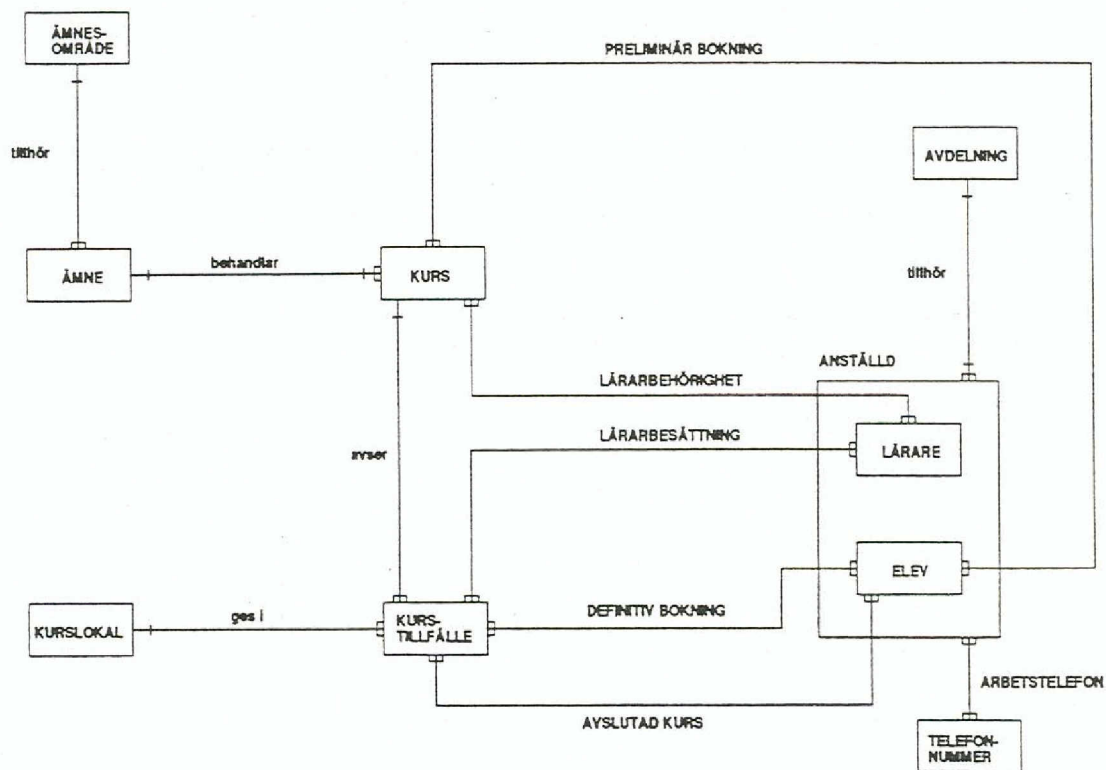


FIG. 12: GRAFISK SASMO-MODELL ÖVER KURSVERKSAMHETENS BEGREPP.

DM-FYR

Bakgrund

Inom FYR-projektet (ett SISU tillämpningsprojekt med deltagande av Stockholms Teleområde, Televerket Nät, Statskontoret och Svenska Institutet för Systemutveckling) har ett modelleringsspråk för begrepps- och datamodellering kallat DM-FYR tagits fram. Språket bygger på SASMO, men en hel del utvidningar och påbyggnader har gjorts bl a i syfte att klarare distingera mellan begrepps- och relationsdatamodell- ("tabell"-)nivån.

Modelleringsbegrepp

Modelleringsbegreppen är objekt, relation, attribut och domän. Relationer kan vara av typ M:M, M:1- och 1:1. Som i SASMO särbehandlas M:M-relationer på så sätt att de kan ses som "existensberoende objekt" och kan tilldelas attribut. M:1, och 1:1-relationers båda aspekter ("riktningar") beaktas, dvs både relationen och dess invers namnges. En pilspets på relationslinjen anger läsriktningen och avser relationens namn. Motsatt riktning mot pilen avser den inversa relationen vars namn är inramat inom parenteser.

Avbildningsrestriktioner anges för alla typer av relationer och ligger till grund för indelningen i relationstyper (1:1, M:1 och M:M). Ett "M" (symboliserat med en "gaffel" i den grafiska beskrivningen) skall tolkas som 0, 1 eller "många", "1" som 0 eller 1. I de fall värdet 0 inte kan förekomma anges detta explicit genom att min-värdet 1 åsätts relationen (existensregel). Denna senare restriktion anges dock bara för 1:1- och M:1-relationer.

M:M-relationer kan gälla mellan objekt, mellan objekt och M:M-relation eller mellan M:M-relationer.

Objekt och M:M-relationer kan ha attribut. Alla attribut är funktionella dvs kan bara ha ett värde.

Attribut "vilar på" domäner dvs värdemängder från vilka attributen "hämtar sina värden".

Generiska samband mellan objekt modelleras. Det mer generella objektet kallas super-objekt och de specialiserade objekten kallas sub-objekt. Ingen explicit notation för att ange egenskaper för specialiseringar (av typ uttömmande, icke uttömmande) finns utvecklad ännu.

Objekt kan identifieras med ett attribut eller med en sammansättning av attribut eller attribut och (M:1-, 1:1-) relationer till andra objekt. Om en sådan sammansättning utses som primär identifierare kan sammansättningen ses som objektets "Id-begrepp" och kan namnges.

Om alternativa attribut/relationssammansättningar finns med vars hjälp objekt kan identifieras, kan "Id-begreppets" namn bibehållas men dess "innehåll" omdefinieras.

Synonyma namn för objekt, M:M-relationer och attribut kan noteras.

Objekt och M:M-relationer har en livscykel som är bestämd av hur de skapas, förändras och tas bort av dialogsteg i rutiner/delrutiner, som har specificerats med hjälp av rutinskisser (enligt FYR/SVEA-metodik).

För att stödja uppfattningen om och förståelsen av de begrepp som modelleras används s k tabell exempel. Enligt de regler för övergång till en "logisk" relationsdatamodell som finns definierade kan en tabellstruktur härledas från begrepps-/objektmodellen. Tabeller och kolumner i dessa bibehåller de namn som gets objekt, M:M-relationer, övriga relationer och attribut i objektmodellen. Tabellerna kan sedan fyllas med exempelvärden.

DM-FYR understödjer modellering på flera nivåer. En övergripande modell kan först ritas i grafisk notation och efterhand detaljeras med hjälp av formulär. En "objektnivå" och en "tabellnivå" kan urskiljas i tekniken.

Beskrivningsteknik/notation

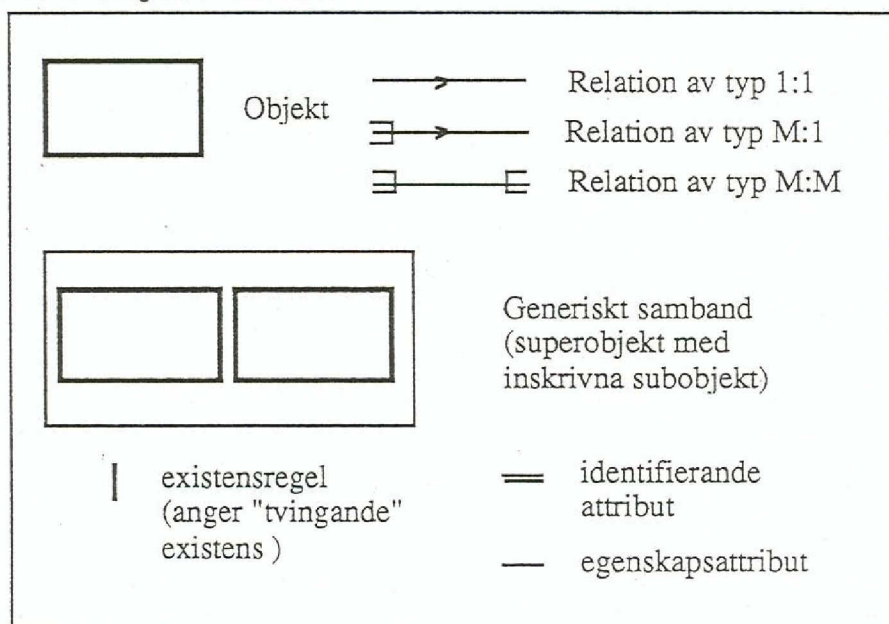


FIG 13: SYMBOLER I DM-FYR:S GRAFISKA NOTATION

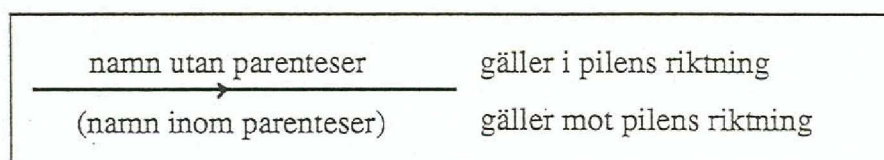


FIG 14: NAMNGIVNING AV RELATIONER

Förhållande till andra modelltyper, metodik och datorstöd

Objekt- och datamodellering enligt DM-FYR bedrivs ihop med rutinskissning. För rutiner/delrutiner som används av funktioner i en verksamhet specificerar

man "vyer" (projektioner) av objekt/relationer (eller tabeller). I arbetet med att specificera sådana vyer av modellen (in- och utdataprecisering) kan upptäckter göras som leder till att objektmodellen och därmed relationsdatamodellen kan behöva kompletteras och förändras.

Som ovan nämnts används i DM-FYR ett starkt iterativt arbetssätt, där man växlar mellan ett begrepps/objektorienterat betraktelsesätt och ett tabellorienterat betraktelsesätt. Bakgrunden är att relationsansatsens tabellstrukturering är välbekant för många vilket genom användandet av exempelvärden bedöms ge positiva effekter för begreppsmodellens kvalitet.

Datorstöd för DM-FYR, liksom för rutinskissning, enkel funktionsmodellering och AU/ADB-planering finns utvecklat. Detta har gjorts inom FYR-projektets ram och med användning av CASE-skalverktyget Ramatic.

Utvecklingsarbetet inom FYR-projektet fortsätter. De enligt planerna närmsta uppgifterna kommer bl a att beröra samordning mellan modeller, design från "logiska" (nominella) tabellstrukturer till "fysiska" (optimerade) tabellstrukturer samt vidare aktivitetsbeskrivning och preciserad "in-/utdata-precisering" enligt SVEA-metodikens "måla skärm"-förfarande.

Litteratur

[F1] *Fyr-delrapport för etapp 1*, 1990-12-17.

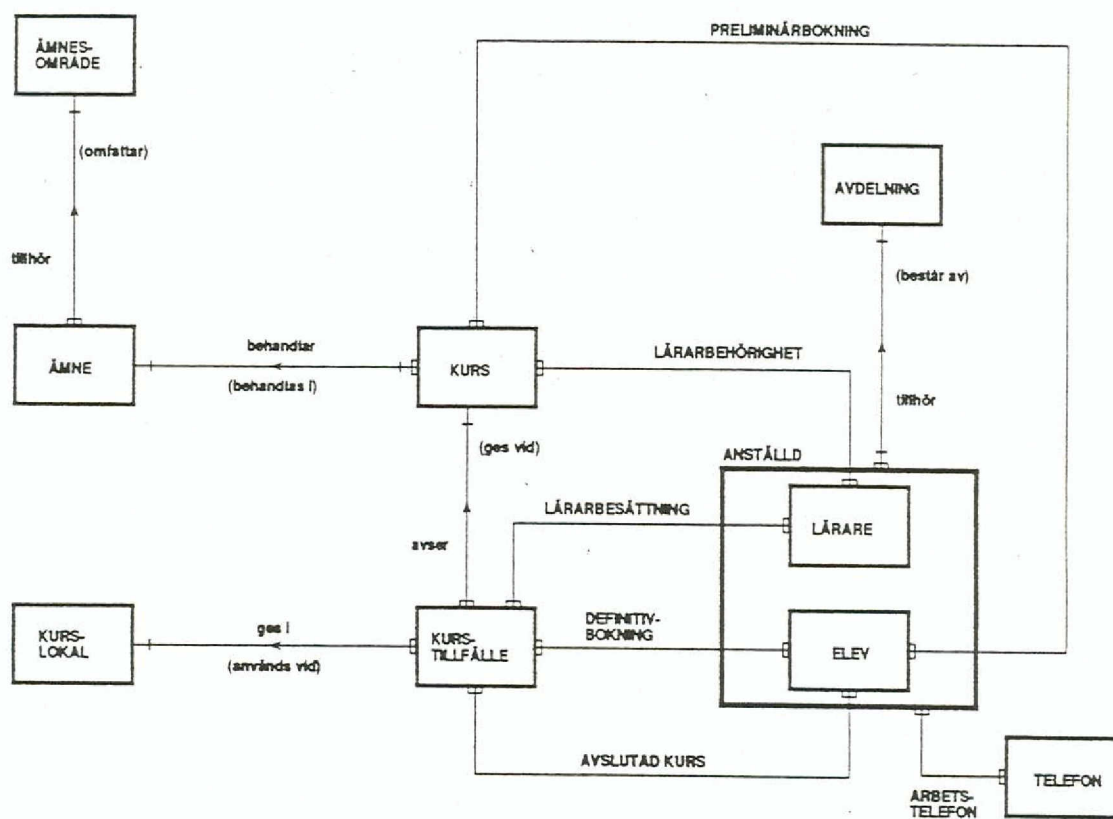


FIG. 15: GRAFISK DM-FYR-MODELL ÖVER KURSVERKSAMHETENS BEGREPP.

TelMod - B

Bakgrund

TelMod-B är utvecklad i samarbete mellan Televerket, Telelogic och SISU som ett förslag till modelleringsansats för det s k IA-projektet (IA= InformationsAdministration).

Utvecklingsarbetet utgick från den datamodelleringsansats som ingick i den s k OBS-metoden

(en systemutvecklingsmodell och metod utvecklad inom Televerket). Avsikten var att tillföra denna ett antal semantiska förstärkningar och det kan vara intressant att återge några av de krav som ursprungligen ställdes upp inför utvecklandet av modelleringsansatsen:

- * det skall vara möjligt att ange domän för 'lexikala' attribut
- * relationer mellan objekt skall uttryckas som verb, där så är möjligt (undantag 'lexikala' attribut)
- * relationers riktning skall framgå på ett väldefinierat och tydligt sätt
- * en relations båda riktningar skall namnges
- * avbildningsrestriktioner skall (kunna) anges
- * om M:M-relationer 'objektifieras' bör man vända gafflarna åt rätt håll
- * generiska relationer skall få användas, dvs det skall finnas en notation för detta
- * någon form av notation för beskrivning av ett objekts livscykel bör övervägas

Modelleringsbegrepp

De grundläggande modelleringsbegrepp som används i TelMod-B är objekt, samband (objektsamband och attribut) samt domän. Begreppen ligger på typ- (eller klass-)nivå, dvs objekt avser egentligen objekttyp/objektklass etc, men i syfte att inte tynga ner begreppsapparaten alltför mycket talar man om objekt, samband, etc. Ett enskilt objekt tillhörigt en viss typ kallas en objekt förekomst. Ett objektsamband är ett samband mellan två objekt (kan vara ett och samma objekt). Varje objektsamband ses och namnges i båda "riktningarna".

Objekt kan vara existensoberoende eller existensberoende. Förekomster av existensoberoende objekt kan, som namnet utsäger, existera oberoende av om andra objekt existerar eller inte.

Är existensen däremot avhängig av att andra objekt existerar, så är objektet existensberoende.

Attribut är samband mellan objekt och domäner. Objekt har attribut och attribut avbildar domäner. Domäner kan anses motsvara standardiserade dataelementtyper.

Generiska samband (ISA-relationer) kan gälla mellan objekt(typer). Flera "oberoende" specialiseringar kan göras av ett och samma objekt, var och en enligt visst specialiseringskriterium. Det objekt som specialiseras, dvs det mer generella objektet kallas super(objekt)typ, de specialiserade objekten kallas sub(objekt)typer. Subtypernas objektmängder inom en specialisering skall vara disjunkta (icke överlappande). En objekt förekomst kan alltså inte tillhöra mer än en subtyp inom en specialisering. En specialisering kan anges vara uttömmande eller icke uttömmande. Är specialiseringen uttömmande så gäller att varje förekomst av supertypen tillhör en subtyp. Är den icke uttömmande gäller alltså att det finns förekomster som inte tillhör någon av de aktuella subtyperna. Det är i princip tillåtet för ett objekt att vara subtyp till mer än en supertyp, i vilket fall objektet ärver egenskaper från samtliga superobjekttyper ("multipelt arv"). Viss försiktighet rekommenderas dock i dessa situationer, eftersom "arvskonflikter" kan uppstå.

För objektsamband och attribut gäller avbildningsrestriktioner. Dessa anges i form av min-, max-angivelser. Objekt identifieras med hjälp av attribut och/eller objektsamband.

Namngivningskonventioner: namn på objekt skall anges i singularis och ha substantivkaraktär.

De skall skrivas med versaler. Detta gäller också namn på domäner. Namn på objekt och domäner skall vara unika. Namn på objektsamband skall anges som verb (gäller både för sambandet och dess invers). Gemena bokstäver används. Detta gäller också för attributnamn. Sambands- och attributnamn skall vara unika inom objekttyp. Ord mellanrum i namn anges med understrykningstecken (eftergift till vissa datorstöd).

En meta(begrepps-)modell finns upprättad över TelMod-B. Den återfinns i [TM2] och [TM4].

Beskrivningsteknik/notation

Den använda notationen är i huvudsak grafisk. Använda grafiska symboler framgår av figur 16.

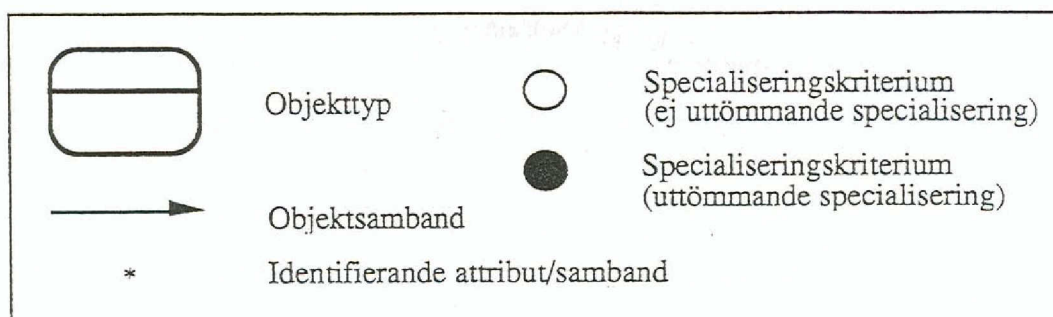


FIG. 16: SYMBOLER I TELMOD-B:S GRAFISKA NOTATION

I grafen beskrivs objekt och samband mellan objekt. Sambandslinjerna är pilförsedda och namngivna - sambandets namn och inversens namn anges, det senare inom parentes. Avbildningsrestriktioner noteras för båda riktningar med kombinationer av icke-negativa heltal och N på formen MIN:MAX.

Identifierande objektsamband markeras med asterisk (*), liksom identifierande attribut, som kan skrivas in i objektsymbolen.

Övriga (beskrivande) attribut för objekt kan också skrivas in i objektsymbolen (se exempel i figur 17) och/eller noteras i en attributlista. Attributens resp domän och avbildningsrestriktion specificeras vid användande av fullständig TelMod-B notation.

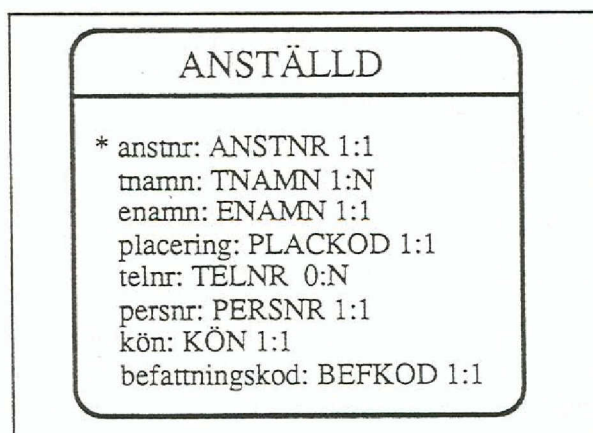


FIG.17: ATTRIBUT FÖR OBJEKT KAN TILLSAMMANS MED DOMÄN OCH AVBILDINGSRESTRIKTIONER ANGES I GRAFEN.

Förhållande till andra modelltyper, metodik och datorstöd

Hur TelMod-B skall ses i förhållande till andra modelltyper linjeras ut i [TM2], vari också riktningar för framtida utvecklingsarbete anges.

För modellering av olika aspekter av verksamheten anges modelltyperna:

Verksamhetens syfte (mål/åtgärdsbeskrivande modeller - *varför*-orientering)

Verksamhetens begrepp/objekt (objekt/begreppsbeskrivande modeller
- *vad*-orientering)

Verksamhetens bedrivande (process/organisationsbeskrivande modeller
- *hur*-orientering)

Verksamhetens regler (villkors/regelbeskrivande modeller - *när*-orientering)

Vid modellering av informationsstöd (för verksamheten) anges modelltyperna:

Informationsstödet data (objekt/databeskrivande modeller - *vad*-orientering)

Informationsstödet applikationer (process/applikationsbeskrivande modeller
- *hur*-orientering)

Informationsstödet regelverk (villkors/regelbeskrivande modeller
- konsekvens-orientering)

Ovanstående får ses som en ramverk inom vilket olika modelltyper och kopplingar/övergångar mellan modelltyper kan bearbetas vidare. Det poängteras att det finns ett antal redan nu i praktik och undervisning använda modelltyper, som successivt skulle kunna lyftas in i ramverket.

Av ovanstående modelltyper är TelMod-B avsedd att användas för modellering av verksamhetens begrepp/objekt. En arbetsgång för metodens "dataaxel" finns föreslagen dvs för att komma "från allmän verksamhetsanalys till körbara datastrukturer". Arbetsgången består av 4 skeden:

1. Affärsinriktad begreppsmodellering
2. Datoriseringsinriktad begreppsmodellering
3. Nominell tabellstruktur
4. Driftmässig tabellstruktur

Affärsinriktad begreppsmodellering syftar till att "uttrycka grundläggande ideer om verksamheten och få upp en karta att göra avgränsningar i". Vid datoriseringsinriktad begreppsmodellering avgränsas och preciseras delar av dessa modeller. Detaljer som är av vikt för datautformning fylls på; t ex kan en "termifiering" av egenskaper göras. Nästa steg innebär en övergång från verksamhetsvärlden till datavärlden - en s k nominell tabellstruktur (en relationsdatamodellen) tas fram enligt angivna transformationsregler. Det 4:e skedet innebär att den nominella tabellstrukturen driftpassas med hänsyn till befintliga prestandakrav för effektiv lösning.

Datorstöd för begreppsmodeller av TelMod-B typ finns i Business Modeler (prototyp för datorstödd verksamhetsanalys utvecklad inom SISU). Ett TelMod-B stöd har också utvecklats med hjälp av Ramatic (CASE-skal prototyp utvecklad inom SISU). Vidare har DEFT använts för framtagning av TelMod-B modeller.

Litteratur

- [TM1] Berghem, A & Jussila, K.: *TelMod, modelleringsansats för televerket*, Stlo, TeleLOGIC Sv, 1989-07-17
- [TM2] Nilsson, B, Willars, H. & Korinek, M.: *TelMod pärmprototyp*, SISU, 7 jan 1991
- [TM3] Wangler, B.: *Förslag till modelleringsansats för IA-projektet*, SISU, 12 sep 1988
- [TM4] Wangler, B. & Lindström, G-G.: *TelMod-B, Ett begreppsmodellerings-språk för Televerket*, Sept 1990

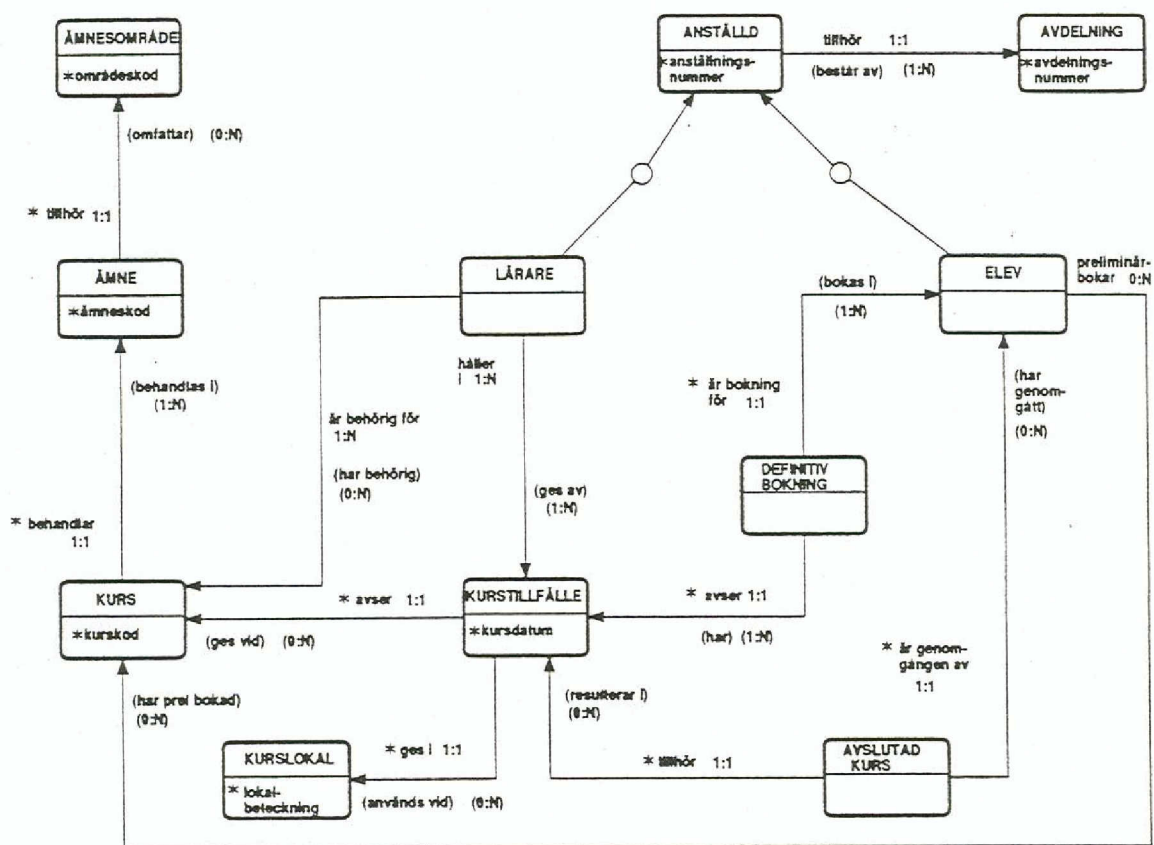


FIG. 18: GRAFISK TELMOD-B-MODELL ÖVER KURSVERKSAMHETENS BEGREPP.

TEMPORA-ERT

Bakgrund

Utvecklad inom TEMPORA-projektet, ett 5-års samarbetsprojekt ingående i det sk ESprit-programmet. I projektet deltar BIM (Belgien), Hitec (Grekland), Imperial College (England), LPA (England), SINTEF (Norge), SISU (Sverige), University of Liege (Belgien) och University of Manchester Institute of Science and Technology (England).

I TEMPORA-projektets ansats för informationssystemutveckling ingår ett flertal komponenter, men beskrivningen koncentreras här till ERT-(Entity-Relationship-Time) ansatsen för konceptuell modellering. På specifikationsnivån finns också två andra modeller, en regelmodell och en processmodell. Av dessa är ERT-modellen den modell som först utvecklades och som nu ses som relativt stabil till sin utformning.

Modelleringsbegrepp

I ERT-ansatsen används modelleringsbegreppen entitet, relation (förhållande), värde, tidsperiod och komplext objekt samt klasser av dessa, dvs entitetsklasser, relationsklasser, värdeklasser, tidsperiodklasser och komplexa objekt-klasser. Värdeklasser kan också vara komplexa dvs innehålla andra värdeklasser som komponenter. Relationer kan gälla mellan två entiteter eller mellan en entitet och ett värde.

Entitets- och relationsklasser kan vara härledda. Till varje sådan klass skall finnas en härledningsregel, som, vid varje tidpunkt anger de entiteter eller individuella relationer, som tillhör klassen.

I ERT-modellen betraktas tidsdimensionen speciellt. Det innebär att man till varje tidsvarierande entitets- och relationsklass, genom tidsstämpling, kan associera en tidsperiodklass.

Den tidsperiod, som genom tidsstämpling av en entitetsklass, associeras till entiteterna i klassen, representerar varje entitets *existensperiod* i modellen dvs perioden från den tidpunkt entiteten föddes i modellen till den tidpunkt då entiteten utgick ur modellen. Tidsperioder som associeras till enskilda relationer i en relationsklass representerar relationens *giltighetstid* dvs tidpunkten från vilken relationen etablerades i modellen till den tidpunkt då relationen utvecklades dvs slutade att gälla. Detta möjliggör att hålla reda på historik för entiteters existens och relationers giltighet i systemet.

Begreppet komplext objekt används i ERT-modellen. Entitetsklasser och värdeklasser kan modelleras som komplexa. En komplex entitet är en abstraktion - en aggregering eller en gruppering - av entiteter, relationer och värden, enkla eller komplexa. Ett komplext värde är en abstraktion - en aggregering eller en gruppering - av värden, enkla eller komplexa. Ingen speciell särskilning görs mellan aggregering och gruppering - vilket som är förhanden kan avläsas genom de kardinalitetsretriktioner som åsätts de komponentrelationer som förbinder

varje komponent med det komplexa objektet. Om många (mer än 1) komponenter tillhöriga viss komponentklass ingår i ett komplext objekt föreligger en gruppering.

Komplexa objekt används, i ERT-modellen, inte bara för att modellera fysiska komponenthierarkier utan också logiska sådana. Det innebär att en och samma komponent kan vara komponent till mer än ett komplext objekt.

Det grundläggande motivet till att inkludera komplexa objekt i ERT-modellen är den möjlighet som detta ger att kunna abstrahera bort från detaljer, som i en speciell situation inte är av intresse.

ERT-modellen är en utvidgad E-R modell, utvidgad i den meningen att den innefattar en tidsdimension. Dessutom skiljer den sig från den ursprungliga E-R-modellen i det att alla associationer mellan objekt ses som relationer ("relationships") och att därmed ingen distinktion görs mellan relationer och attribut ("attributeships").

Beskrivningsteknik/notation

Den grafiska notationen har olika symboler för de olika klasserna av objekt. Så t ex symboliseras komplexa entitets- och värdeklasser med dubbel ramlinje och värdeklass med en indikering i nedre högra hörnet (se figur 19). Härledda entitets- och relationsklasser symboliseras streckade, tidsstämplade entitets- och relationsklasser förses med speciell tidmarkering.

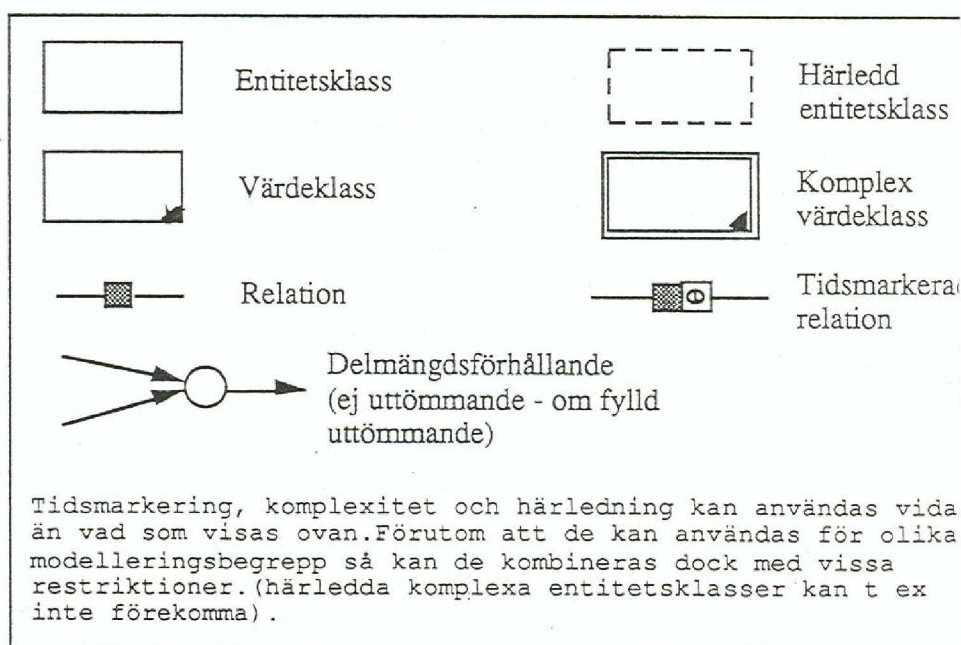


FIG. 19: SYMBOLER I TEMPORA-ERT:S GRAFISKA NOTATION

Komplexa objekt har en inre struktur. Så t ex kan en komplex entitetsklass bestå av andra entitetsklasser (även komplexa) och värdeklasser, medan en komplex värdeklass bara kan bestå av andra värdeklasser (komplexa och/eller enkla).

Grafiskt symboliseras strukturen genom att komponenterna ritas inuti det komplexa objektet och att de förbinds via "ingår i"-relationer till den omgivande ramen. "Ingår i"-relationer symboliseras som andra relationer vilka som helst och innebörden framgår av namngivningen.

Generiska samband (ISA-relationer) symboliseras på samma sätt som i TelMod-B

Alla "vanliga" förhållanden är binära och namnges i båda riktningar. Dock finns ingen riktning angiven för förhållanden.

Avbildningsrestriktioner anges för alla förhållanden i båda riktningar med min-, max-angivelser, på formen (1:M) osv. M kan ersättas av aktuellt heltal.

Förhållande till andra modelltyper, metodik och datorstöd

Som ovan nämnts ingår, på specifikationsnivån, också språk för specifikation av regler och processer. Processer och flöden - triggande och icke-triggande - mellan processer, beskrivs i s k Process Interaction Diagrams (PID). Dessa används i syfte att fånga regler som gäller i den modellerade verksamheten och som beskrivs i regelmodellen. Ett språk, med definierad syntax och semantik för beskrivning av regler, kallat ERL (External Rule Language), finns utarbetat. Detta innehåller bl a ett antal temporala konnektiv (t ex "sometime_in_past", "just_after") med vars hjälp tidsmässiga relationer kan beskrivas.

Utveckling av metodik dvs tillvägagångssätt, riktlinjer och procedurer för framtagning och underhåll av en TEMPORA-modell befinner sig ännu i ett inledningsskede.

En implementering av datorstöd för ERT-modellen finns gjord med hjälp av Ramatic (CASE-skal prototyp utvecklad inom SISU).

Litteratur

[TP1] *The TEMPORA Concepts Manual*, TEMPORA-projektet, Sept. 1990.

10.0 Sammanställning och försök till jämförelse

Det är svårt att ge en fullständig bild av modelleringsansatser och deras relation till varandra. Ovan har några ansatser presenterats och försök till belysande exempel har gjorts. Vilken ansats som är aktuell i en viss modelleringssituation finns inget entydigt svar på men förhoppningsvis kan denna skrift vara till viss hjälp i en valsituation.

Nedan följer en liten uppställning av vad som gäller för de presenterade ansatserna på några olika punkter och uppdelat på 2 avsnitt: det första anger några karakteriserande egenskaper för ansatserna. I det andra avsnittet skissa några förslag till vad en tänkbar, fortsatt jämförelse med värderande inslag skulle kunna tänkas omfatta.

10.1 Karakterisering av ansatserna

10.1.1 Grundläggande modelleringsbegrepp

Vilka typer av modelleringsbegrepp som ansatserna håller sig med säger en hel del om dem. Begreppen kan vara snarlika men t ex benämnas olika. De sammanförs i tabellen nedan till fyra grundtyper av modelleringsbegrepp som de ungefärligt motsvarar : Entitet, Relation, Attribut och Värdeklass. I tabellen framgår också om en icke-grafisk och/ eller en grafisk notation används för begreppen. Om ansatsen saknar modelleringsbegreppet i fråga, så markeras det med —.

Ansats/ Teknik	Modelleringsbegrepp (ungefärliga)			
	Entitet	Relation	Attribut	Värdeklass
CMOL	G, I	G, I	G, I	G, I
NIAM	G, I	G, I	—	G, I
SASMO	G, I	G, I	(G), I	—
DM-FYR	G, I	G, I	(G), I	I
TELMOD-B	G, I	G, I	G, I	I
TEMPORA	G, I	G, I	—	G, I

G Grafisk notation understöds
 (G) Grafisk notation finns men används ej så ofta
 I Icke-grafisk notation understöds

FIG.21: MODELLERINGSBEGREPP I DE PRESENTERADE ANSATSERNA

Modelleringsbegreppen kan variera mycket i sin grad av detaljering och vad som förknippas med varje modelleringsbegrepp. Tex förekommer komplexa entitetsklasser i Tempora-ERT och sammansatta värdeklasser i CMOL, DM-FYR och Tempora-ERT (komplexa värdeklasser).

10.1.2 Relationsbegrepp med regler

I de flesta ansatser är det centralt med namngivningen av de olika instanserna av begrepp. Namngivningen av relationer skiljer sig i de jämförda ansatserna. SASMO använder sig i regel bara av ett namn för relationer, medan övriga ansatser medger två namn, ett för att belysa vardera perspektivet.

Avbildnings- eller kardinalitetsregler som används för relationer är ingalunda lätta att reda ut. I tabellen görs ett försök att visa hur det av en relationsförekomst framgår det minimala och det maximala antalet objekt som involveras av relationen. Här används bl a speciella grafiska symboler på relationslinjen och mapping-notation med min och max.

Objekts beroende av identifikation från andra objekt som de är förbundna med anges i de flesta ansatser grafiskt men icke-grafiskt i DM-FYR och Tempora-ERT, i formulär och via regler respektive.

I alla ansatser tillåts att en relation förbinder ett och samma objekt (rekursiv relation). I NIAM t ex förekommer ömsesidigt uteslutande relationer, dvs dessa kan ej förekomma samtidigt. I NIAM och Tempora-ERT urskiljs inte något särskilt attributbegrepp utan likvärdiga relationer förbinder entiteter med värdeklasser som med entiteter.

Förutom i Tempora-ERT finns i alla ansatser tillgång till en grafisk markering för identifierande relationer och attribut. Det icke-grafiska notationssättet är allmänt mest utvecklat för CMOL, NIAM och Tempora-ERT som utnyttjar speciella språk.

10.1.2 Generiska förhållanden och dynamik

Subtyper eller dylika specialiseringar av entiteter förekommer i alla ansatser. I NIAM, TelMod-B och Tempora-ERT är hanteringen av specialisering mer utvecklad med regler för specialiseringar. Sådana kan också anges i CMOL. T ex kan det specificeras om det är en uttömmande uppdelning som har gjorts. Flera specialiseringsgrupper kan urskiljas från en enskild entitet i Telmod-B och Tempora-ERT, var och en bestämd utifrån ett specialiseringskriterium. Internt måste dessa vara disjunkta. Här understöds dessutom att en entitet är en specialisering av flera andra, dvs att de generiska hierarkierna inte tvingas vara trädstrukturer.

Ansats/ Teknik	Entitetsbegrepp		Dynamik
	Subtyp el dyl	Subtyps- regler understöds	
CMOL	ja	ja (regler)	Icke-statiska beskrivningar används i form av händelser, procedurer
NIAM	ja	ja	procedurer
SASMO	ja	nej	—
DM-FYR	ja	nej	—
TELMOD-B	ja	ja	—
TEMPORA	ja	ja	tidsstämpling, regler

FIG.23: SPECIALISERINGSMÖJLIGHETER AV ENTITETSBEGREPPET I RESP TEKNIK OCH ANVÄNDNINGEN AV DYNAMIK

Vad avser dynamik är den ofta inte prioriterad i begrepps- och datamodelleringsansatsen som sådan, utan hanteras mest i ansatser för funktions- och flödesbeskrivning. Förutom dynamiska aspekter som kan fångas med kopplingar till sådana ansatser förekommer dock händelser, procedurer, regler och tidsstämpling enligt tabellen för de behandlade ansatserna.

10.2 Skiss till tänkbar värdering av ansatserna

En fortsatt värdering av ansatserna/språken skulle kunna grunda sig på följande principer och antaganden:

ISO:s konceptualiseringsprincip

Denna princip utsäger att ett konceptuellt schema (modell) enbart skall omfatta konceptuellt relevanta aspekter dvs det skall exkludera alla implementerings- och datorberoende aspekter.

Ett konceptuellt schema framställs med användande av en modelleringsansats. Följkravet på en modelleringsansats för konceptuell modellering måste rimligen vara att denna ansats med modelleringsspråk (begrepp och notationer) och metodik inte skall innehålla implicita eller explicita modelleringsbegrepp och -konstruktioner, som härstammar från eller är motiverade av implementerings- och/eller datormässiga överväganden. Begreppen och språken skall vara "sant" problem-/verksamhetsorienterade.

ISO:s 100%-princip

Denna princip utsäger att alla (100%) relevanta lagar, regler och villkor som finns/gäller i verksamheten skall beskrivas i det konceptuella schemat. Detta gäller såväl aspekter/regler av statisk som dynamisk typ. De skall alltså inte vara inbakade i applikationsprogram, t ex.

Följkravet på en modelleringsansats blir rimligen att denna måste tillhandahålla begrepp och notationer som möjliggör och underlättar att sådana regler och villkor kan fångas och beskrivas.

I detta sammanhang kan man också notera om regler av härledningstyp (behandlingsregler, bearbetningsregler) kan uttryckas "deklarativt", här tolkat så att de noteras icke-redundant i avskiljbar regelmodell i anslutning till, eller kopplad till den konceptuella modellen eller anges "procedurellt" dvs i anslutning till processer och procedurer som beskrivs i modeller av annat slag med, eller utan, koppling till begrepps-/objektmodellen.

Flera språknivåer

Verksamhetsregler och villkor kan vara mycket komplicerade. För att kunna uttryckas - och med precision - och med beaktande av tidsmässiga och dynamiska inslag - måste språken vara välutvecklade, uttrycksfulla och precisa. Detta är svårt att åstadkomma utan att språket är starkt formaliserat ofta med logik-orienterade inslag. Men det är verksamhetspersoner som har kunskap om verksamhetsförhållanden och -regler och som behöver uttrycksformer och direkta eller indirekta andra tillvägagångssätt för att kunna formulera reglerna och att validera den resulterande modellen.

Det måste alltså ses som en fördel om modelleringsansatsen har inslag som är avsedda att befrämja detta. Vi refererar här till kapitel 7 där det t ex talas om grad av detaljering, grad av projicering och grad av aggregering.

Referenser

- [1] Bubenko, J.& Lindencrona, E.: *Konceptuell modellering - Informationsanalys*, Studentlitteratur, 1984.
- [2] *Concepts and Terminology for the Conceptual Schema and the Information Base*,
ISO TC97/SC5/WG3, 1982.

TRIAD utvecklar IA

Televerket har just tagit första steget in i sin nya IA-organisation och Posten håller på att bygga upp sin nya DA-organisation. Båda organisationerna har sett nyttan att inför 90-talet gå vidare tillsammans i TRIAD-projektet som drivs tillsammans med SISU. Statskontoret deltar också i projektet för att på sikt kunna föra ut nya synsätt och hjälpmedel inom den civila statliga sektorn.

Ericsson Data Services deltar med tyngdpunkten i den del som handlar om att utveckla kompetenta modelleringsledare, delprojektet "Avancerad utbildning för modelleringsledare".

Modelleringsmetoder är centrala i bedrivandet av verksamheten inom informationsadministrationen. Därför arbetar ett delprojekt med utvecklandet av "nästa generation modelleringsmetod" som skall sättas i händerna på informationsadministratören. Siktet är att fördjupa och bredda dagens modelleringsmetoder och där hämta in kunskap från pågående forskning och utveckling internationellt. (faktaruta om IAS91).

Som stöd för informationsadministrationen behövs verktyg. Inom TRIAD arbetar man där inom två områden, kataloger och verktyg.

Delprojektet kataloger arbetar dels med att utforma den informationsmodell som måste kunna täckas av en katalog, dels med att granska och följa utvecklingen av produkter inom området t ex IBM:s "Repository" och Digital's "CDD". Dessutom följer man standardiseringen internationellt kring IRDS. För parterna i projektet liksom för andra organisationer är detta ett tungt område både vad gäller kommande investeringar ekonomiskt och vad gäller kompetenta resurser för en kommande övergång till "repository-världen". - Det inledande skedet syftar till att bygga upp en kunskapsplattform, som sedan kommer att kunna utnyttjas för kravställande och planering och genomförande av övergång från dagens kataloghantering till morgondagens.

Den andra verktygshanterande delen inom TRIAD-projektet, delprojektet "verktyg för informationsadministration", syftar till att ta fram verktyg för uttag och dokumentering av modeller. Betoningen ligger på människa datorgränssnitt och i första skedet görs utveckling av HYBRIS-gränssnittet med prototyper för Posten och för Televerket.

För att hålla ett helhetsperspektiv på projektets delar och för att ha inpassningen av funktionen Informationsadministration i organisationens övriga verksamhet arbetar delprojektet "Krav på IA". I delprojektet arbetar man dels med att kartlägga dagens krav på dataadministration och projicera till morgondagens krav på IA. Dessutom skall man skapa en bild av IA-verksamhetens innehåll och organisation. Från detta i sin tur ställer man krav

på övriga delprojekt. Vilka krav skall ställas på kompetens, metoder, hjälpmedel typ kataloger och gränssnitt?

TRIAD projektet är stort

Budgeten för TRIAD-projektet löper på 10 MSEK per år under en treårsperiod som startar vid kalenderåret 1991 års början och som alltså beräknas avslutad vid utgången av 1993.

TRIAD-projektet är ett tillämpningsprojekt

Det innebär att parterna, Televerket, Posten, Statskontoret, EDS och SISU går in med såväl persontidssatsningar som ekonomiska och att STU, Styrelsen för Teknisk Utveckling, bidrar med ett ekonomiskt tillskott som svarar mot ungefär 40 % av den insatta persontiden.

Öppet för fler deltagare

Parterna i TRIAD-projektet vill gärna öka tempot och bredda perspektivet och vill därför gärna ha fler parter in i projektet. Dessa parter får då enligt SISU:s tårtprincip "betala för en tårtbit, men ät hela tårtan", tillgång till projektets resultat med en insats som ger stor "price performance".

Nya deltagare kan gå in i hela projektet eller i det eller de delprojekt som verkar intressantast. En förutsättning är att man framförallt är beredd att satsa kompetent personal. För de flesta intressenter borda detta vara ett utmärkt sätt att driva personalutveckling för personer t ex inom DA-området, samtidigt som man bygger upp beredskapen inför 90-talets IA-verksamhet.

Kompetensutveckling viktigt resultat

En viktig effekt för parterna av deras medverkan i TRIAD är kompetensutveckling. Man satsar på att ta in personer som så småningom eller redan idag arbetar med DA och IA för att ge dem en djup och "frontlinje"-mässig kompetens. Detta skall utnyttjas när man successivt för in resultaten i den egna organisationen. Projektdeltagarna har alltså en viktig roll som kunskapsförmedlare i den egna organisationen. Dessutom ger projektarbetet deltagarna tillfälle till en egen utveckling inom det professionella området som är unik.

Informationsspridning

Det sjätte delprojektet "Informationsspridning" har till uppgift att sörja för att i första hand parterna men också SISU:s övriga intressenter successivt kan följa och tillgodogöra sig resultat från TRIADprojektet. Seminarier, rapporter och referensgruppsverksamhet är led i den verksamheten.