

SISU **rappport**

nr 9

Kvalitet hos konceptuella scheman

**Sven-Bertil Wallin
Benkt Wangler
Rolf Wohed**

SISU

**Svenska Institutet för Systemutveckling
Box 1250, 164 28 KISTA**

Kvalitet hos konceptuella scheman

ISSN: 0282-9924

Copyright
SISU - Svenska Institutet för Systemutveckling
Juni 1990

Innehåll

1. Inledning	1
2. Kvalitetsbegreppet	3
3. Användningsområden för konceptuella modeller	5
3.1 Informationssystemutveckling.....	5
3.2 Verksamhetsutveckling.....	6
4. Egenskaper hos konceptuella modeller	8
4.1 Inledning.....	8
4.2 Egenskaper hos en kvalitativt bra modell	9
4.3 Medel att nå kvalitet hos schemat	13
5. Diagnosverktygets uppbyggnad	19
5.1 Idé och önskade egenskaper.....	19
5.2 Expertsystem, vad är det?.....	20
5.3 Expertsystemets utseende idag.....	20
6. Slutsatser	22

1. Inledning

Dagligen utvecklas och dokumenteras hundratals datamodeller, verksamhetsmodeller, begreppsmodeller, dataflödesmodeller, osv. På basis av dessa byggs sedan informationssystem. Det finns då stor risk att logiska fel och andra kvalitativa brister hos dessa problemnära modeller och specifikationer fortplantar sig till de implementerade systemen, eller i varje fall försenar och försvårar utvecklingsarbetet.

Att avhjälpa sådana fel och brister kostar mer ju senare de upptäcks under utvecklingsprocessen. Beslut som tas och designunderlag etc som produceras i tidiga faser har ofta mycket stor betydelse för det färdiga systemets detaljer. Ett fel begånget i en tidig fas av systemutvecklingen medför större rättningsarbete ju senare det upptäcks. Om felet inte upptäcks förrän systemet tagits i drift kan de dessutom förorsaka direkt skada.

För att komma till rätta med dessa problem har man på många företag infört särskilda rutiner för kvalitetssäkring i form av planerade och strukturerade genomgångar av specifikationer, tillsammans med experter på verksamheten. Detta angreppssätt syftar, vad gäller tidiga faser av systemutvecklingsprocessen, främst till att stämma av specifikationen i förhållande till verksamhetsområdet, dvs dels att avslöja avvikelser från detta, dels att försäkra sig om att man fått med allt som beställaren avsett.

Några frågor man nu kan ställa är: Kan vi samla och systematisera kunskap och erfarenhet med vars hjälp vi på ett tidigt stadium skulle kunna analysera de problemorienterade specifikationerna i syfte att upptäcka kvalitativa brister? Vad bör för övrigt, i detta sammanhang, förstås med kvalitet, dvs vilka egenskaper vill vi att den färdiga modellen eller specifikationen ska ha? Vilken ytterligare kunskap krävs för att kunna ställa diagnoser av detta slag? Kan denna samlade kunskap rent av byggas in i ett CASE-verktyg, så att eventuella brister upptäcks redan då specifikationerna produceras?

För att undersöka dessa frågor har det bedrivits ett projekt på SISU som syftar till att klargöra begreppet kvalitet med avseende på konceptuella modeller, som ofta kallas objektsmodeller och datamodeller. Avsikten har varit att ange explicita 'kvalitetskriterier', om möjligt på en sådan konkretionsnivå att de kan byggas in i ett datorstöd. Eftersom man rimligen inte kan förutsätta att samma slag av kvalitetskriterier gäller för alla typer av specifikationer har vi i detta projekt koncentrerat oss på den statiska delen av det konceptuella schemat, dvs utan händelser och andra regler än avbildningsrestriktioner för attribut.

Naturligtvis spelar också den (grafiska och/eller textuella) notation som används stor roll, såtillvida att man inte kan ställa samma krav på en modelleringsansats med små uttrycksmöjligheter som på en med stora. Det modelleringspråk som vi närmast haft i åtanke vid våra diskussioner inom projektet är det inom SISU utvecklade SIMOL. Det är en binär associativ, objektbaserad ansats bestående av såväl en grafisk som en textuell notation. Den fortsatta diskussionen i denna artikel avser med andra ord i första hand kvalitet hos konceptuella scheman beskrivna i SIMOL. De kvalitetskriterier som anförs är dock generella i den meningen att de kan sägas gälla för alla modelleringspråk som har motsvarande uttryckskraft.

Parallellt med denna verksamhet har utveckling av ett expertstöd för diagnosticering av konceptuella scheman pågått. Detta är avsett att komplettera det SISU-utvecklade CASE-skalet RAMATIC. Kapitel 5 i denna rapport behandlar frågor som är aktuella i detta sammanhang.

2. Kvalitetsbegreppet

Med kvalitet hos en produkt eller tjänst avses enligt den internationella standardiseringsorganisationen, ISO, dess " ...förmåga att tillfredställa kundens eller användarens behov och förväntningar". Man skulle också kunna säga att någonting har hög kvalitet om det fyller sitt syfte.

Den senare formuleringen ställer användningsområdet i centrum. För att kunna avgöra om någonting har hög kvalitet bör man alltså känna till vad det ska användas till. Detta innebär alltså även att om någonting kan användas till mer än en sak, kan det också ha olika kvalitet med avseende på de olika användningsområdena. Till skillnad från många andra slag av systemspecifikationer kan konceptuella scheman användas i en rad olika sammanhang, förutom för informationssystemutveckling, t ex för affärs- och organisationsutveckling.

Av detta följer, för det första, att man vid en analys av kvalitetsbegreppet med avseende på konceptuella scheman, först borde undersöka vilka de potentiella användningsområdena för konceptuell modellering är, och därefter räkna upp kvalitetskravet för vart och ett. För det andra innebär det att man vid kvalitetsdiagnos av ett specifikt schema strängt taget bör veta vilket det avsedda användningsområdet är. Kapitel 3 nedan innehåller en genomgång av de mest uppenbara användningsområdena för konceptuella scheman.

Ett konceptuellt schema kan göra nytta på flera olika sätt. Det man först tänker på är antagligen att det ska utgöra ett bra underlag för utvecklingsaktiviteter, t ex databasutformning, där det utgör en del av utgångsmaterialet. Det är samtidigt viktigt att schemat innehåller så mycket som möjligt av den information som krävs. Detta ställer krav på modelleringsansatsens uttrycksmöjligheter, men också på att schemat utformats på ett sätt som underlättar transformationen från konceptuellt schema till t ex databasstruktur.

En annan generell nyttoeffekt av att konstruera ett konceptuellt schema är att begrepp som används i en verksamhet får en klarare definition, eftersom man vid konceptuell modellering måste ta ställning till begreppens exakta innebörd. Detta medför att andra

relaterade specifikationer, typ funktionsbeskrivningar och dataflödesmodeller, får en tydligare innebörd, vilket är bra oavsett vad de ska användas till.

Också andra slag av positiva effekter förekommer, t ex skapande av konsensus omkring en begreppsapparat och skapande av nya användbara begrepp. Här är det dock snarast 'kvaliteten' hos själva modelleringsprocessen som är avgörande. I detta sammanhang får man inte heller glömma den utbildningseffekt som blir resultatet för dem som deltar i en modelleringsprocess.

I vårt projekt är dock syftet att enbart granska schemat som sådant, oavsett hur det har producerats. Modelleringsprocessen och de kvalitetsaspekter som kan läggas på denna är därför inte av primärt intresse och lämnas således i fortsättningen därhän.

En genomgång av olika användningsområden har dock övertygat oss om att kvalitetskraven inte varierar särskilt mycket med hänsyn till dessa, utan på det hela taget är generella. Rent allmänt kan dock sägas, att om schemat riktar sig till slutanvändare så bör man betona enkelhet och om det ska utgöra underlag för utformning av implementerbara systemkomponenter så krävs en hög detaljeringsgrad. Detta talar för att det borde finnas möjlighet att ha scheman på olika detaljerings- och abstraktionsnivåer.

Bland viktiga, generella kvalitetskriterier kan följande nämnas:

- syntaktisk och semantisk korrekthet
- god begriplighet, låg komplexitet
- målinriktning, relevans
- stabilitet, flexibilitet
- hög fullständighet vad gäller täckning av områdets verksamhetsregler

Vi avstår här från en mer djupgående diskussion av innebörden i, och bakgrunden till, dessa kvalitetskrav, för att istället diskutera vilka generella egenskaper hos schemat som är av betydelse i sammanhanget och som alltså kan utgöra utgångspunkt i en kvalitetsdiagnos.

3. Användningsområden för konceptuella modeller

I det följande beskrivs ett antal verksamheter där konceptuell modellering kan göra nytta. Gränserna och förhållandena mellan de olika områdena är inte klara, men av praktiska skäl har dessa ändå grupperats efter ungefärlig samhörighet. För varje grupp ges en kort karakteristik.

3.1 Informationssystemutveckling

3.1.1 Planering

Planering bedrivs i syfte att avgöra vilka informationssystem som behövs för att stödja en verksamhet och vilka egenskaper dessa informationssystem ska ha. En detaljerad analys av verksamhetsfunktionernas informationsbehov är nödvändig. Resultatet av denna analys uttrycks med fördel i form av konceptuella scheman. Dessa kan sedan integreras till ett schema för t ex varje funktion. Med detta som underlag kan sedan en uppdelning i lämpliga, kommunicerande informationssystem utföras.

3.1.2 Utveckling

Här är det fråga om att utveckla informationssystem och dess beståndsdelar. En vidareutvecklad och fullt detaljerad konceptuell modell utgör en specifikation av det önskade informationssystemet, som kan utgöra underlag för egenutveckling av systemet med eller utan hjälp av 4GL, eller för val av standardsystem. I det förra fallet utgör den konceptuella modellen det viktigaste underlaget för framför allt databasutformning, men är också en naturlig grund för dialog- och programutformning.

3.1.3 Förvaltning och drift

I samband med användning av informationssystem utgör det konceptuella schemat en användartillvärd 'karta' över systemet och dess beståndsdelar - framför allt över innehållet i dess databaser.

Förvaltning av informationssystem brukar anses innefatta korrigering, förbättring, anpassning och sanering av systemet. Den konceptuella modellen utgör, i sin egenskap att vara den viktigaste delen av systemdokumentationen, en given utgångspunkt i detta arbete. I detta sammanhang kan den konceptuella modellen eventuellt tänkas realiserad inom ramen för en data- och informationskatalog (information resource dictionary, repository).

Även vid vidareutveckling av informationssystem är det konceptuella schemat i sin egenskap av informationskarta viktigt. Det är i detta sammanhang viktigt att också det konceptuella schemat underhålls, så att det ständigt avspeglar det aktuella läget.

3.2 Verksamhetsutveckling

3.2.1 Affärsutveckling

Affärsutveckling handlar om att analysera och klargöra den begreppsapparat som används då man uttrycker ett företags affärsidé eller ett företags eller en verksamhets mål. Syftet kan vara att definiera och avgränsa väsentliga begrepp eller att, i en utvecklings-situation, ifrågasätta företeelser eller att finna nya fruktbara tanke-mönster. Ett konceptuellt schema hjälper här till att öka förståelsen för begreppen och leder ofta till uppkomsten av nya användbara begrepp.

3.2.2 Organisationsutveckling

Syftet med organisationsutveckling är att definiera/avgränsa innehåll i organisationsplaner eller ansvarsområden och arbetsuppgifter. Ett konceptuellt schema utgör här den fasta grund som behövs för att åstadkomma klarhet.

3.2.3 Administrativ utveckling

Administrativ utveckling innebär att vidareutveckla och förbättra en verksamhet. Med anledning av detta utförs en analys av verksamheten och dess funktioner. För att då få förståelse för verksamhetens och dess olika funktioners uppgifter kan ett konceptuellt schema vara till stor nytta.

Sammanfattning

Det finns som synes ett stort antal ganska olikartade användningsområden för konceptuell modellering. Det ligger därför nära till hands att tro att vi har starkt varierande kvalitetskrav för olika områden. Vi har dock kommit till slutsatsen att skillnaderna mest är en fråga om behov av detaljeringsgrad i modellen. Gemensamt för alla användningsområden är ju att man vill definiera och bringa klarhet i den mängd företeelser som hanteras i en verksamhet, det må vara för att utveckla och använda ett informationssystem eller i något annat syfte. Detta talar för att man borde ha tillgång till verktyg som ger modellören/modellanvändaren möjlighet att se modellen på olika detaljeringsnivåer.

Med utgångspunkt från vår analys har vi kommit fram till att de egenskaper som redovisas i nästa kapitel kan vara av intresse då man bedömer ett schemas kvalitet, dock i något växlande grad beroende på det avsedda användningsområdet. Vi har dock inte ansett oss kunna ge några anvisningar för viktning med hänsyn till användningsområde utan lämnar tills vidare detta öppet för bedömning vid varje enskilt tillfälle.

4. Egenskaper hos konceptuella modeller

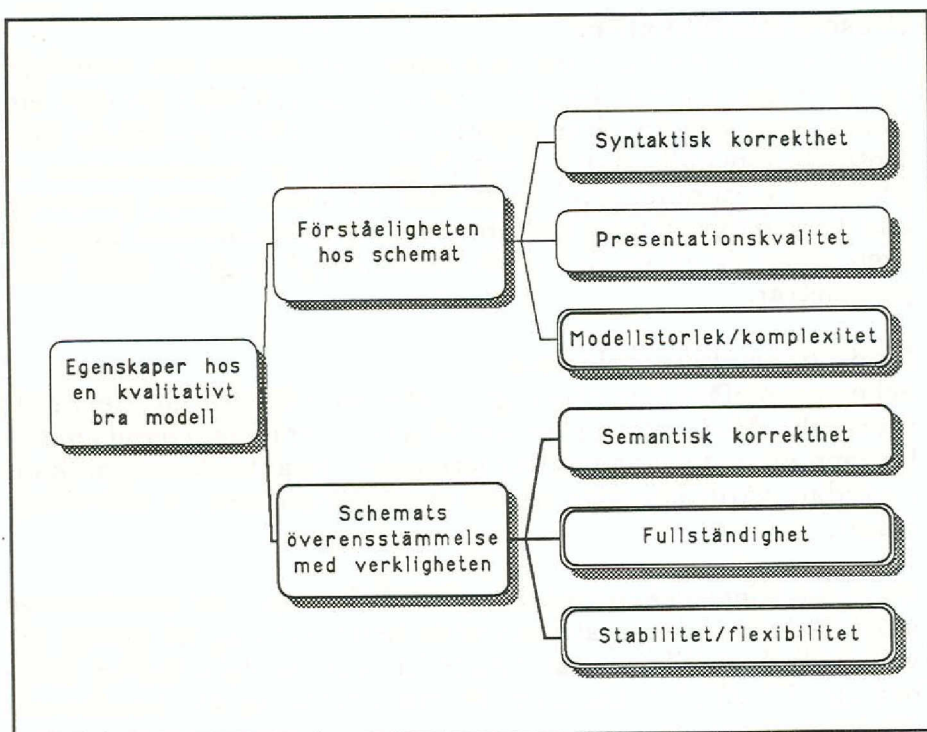
4.1 Inledning

I det följande beskrivs ett antal väsentliga och grundläggande egenskaper hos konceptuella scheman, som i princip är möjliga att bygga in i ett datorstöd. Egenskaperna ska utgöra utgångspunkten för ett diagnostiseringsverktyg byggt på tumregler, vilket interaktivt kan användas som stöd vid schemautformning. Hjälpmidlet ska påpeka fel och brister samt komma med förslag till rättelser och förbättringar.

De uppräknade egenskaperna är inte oberoende av varandra. Man kan dock dela upp dem i två klasser; en klass för egenskaper hos schemat som sådant, dvs oavsett det avbildade verksamhetsområdet, och en för egenskaper som har med schemats sätt att avbilda det specifika verksamhetsområdet att göra.

En kort beskrivning ges för varje egenskap tillsammans med en indikation av på vilket sätt en analys av denna egenskap skulle kunna användas i det nämnda datorstödet.

Därefter beskrivs ett antal tekniker för att åstadkomma dessa kvalitetsegenskaper hos en modell.



Figur 1. Kvalitetskriterier för en konceptuell modell.

Syftet med modellen påverkar kvalitetsaspekterna i varierande grad. En modell framtagen på förstudenivå i ett projekt har andra krav på sig avseende fullständighet, stabilitet och komplexitet än en som är framtagen som underlag för databasdesign. Det är främst egenskaperna Modellstorlek/komplexitet, Fullständighet och Stabilitet/flexibilitet där kvalitetskraven är beroende av syftet. Dessa är markerade med dubbla ramar i figur 1 ovan.

För att exemplifiera de olika kvalitetsegenskaperna har vi valt att genomgående visa ett exempel föreställande en tänkt utlåningsverksamhet i ett bibliotek. Detta är mycket förenklat för att de olika aspekterna ska framträda tydligt.

4.2 Egenskaper hos en kvalitativt bra modell

4.2.1 Förståeligheten hos schemat

Syntaktisk korrekthet

Med detta avses här huruvida schemat är syntaktiskt korrekt och konsistent, dvs internt motsägelsefritt. Kontroller kan utföras automatiskt, antingen i samband med att modellen skapas, eller i samband med att någon annan analys utförs. Felaktigheter påpekas.

Till syntaktisk korrekthet kan också räknas trohet mot benämningkonventioner inom modelleringsansatsen, i den mån sådana finns. Kontroll av detta kan ske vid behov. Avvikelse påpekas.

Presentationskvalitet

Här avses i första hand grafisk kvalitet, dvs sådant som överskåd-
lighet, undvikande av korsande pilar, antal symboler per dm², etc.
Denna egenskap kan inte avgöras enbart utifrån modellens kon-
ceptuella innehåll, utan kräver tillgång till en representation av
grafiken. Varierande grad av stöd kan fås av ett verktyg genom att
låta det generera förslag utifrån en verbal representation. Man kan
även tänka sig att ett datorstöd "ordnar upp" en graf enligt vissa
parametrar.

Modellstorlek/komplexitet

Schemats omfång har naturligtvis betydelse för hur lätt det är att
överblicka. Med omfång eller storlek avses här i första hand antalet
begrepp, dvs antalet namngivna företeelser. Exakt vilken signifikans
ett sådant värde har eller hur det ska användas är dock tills vidare
oklart.

En abstraktionsmekanism inbyggd modelleringsverktyget skulle ge
möjlighet att i presentationen välja komplexitetsgrad genom att
välja/välja bort olika slag av konstruktioner t ex delmängder, attri-
but etc. På så sätt kan man få rätt blandning av detaljer och
överblick.

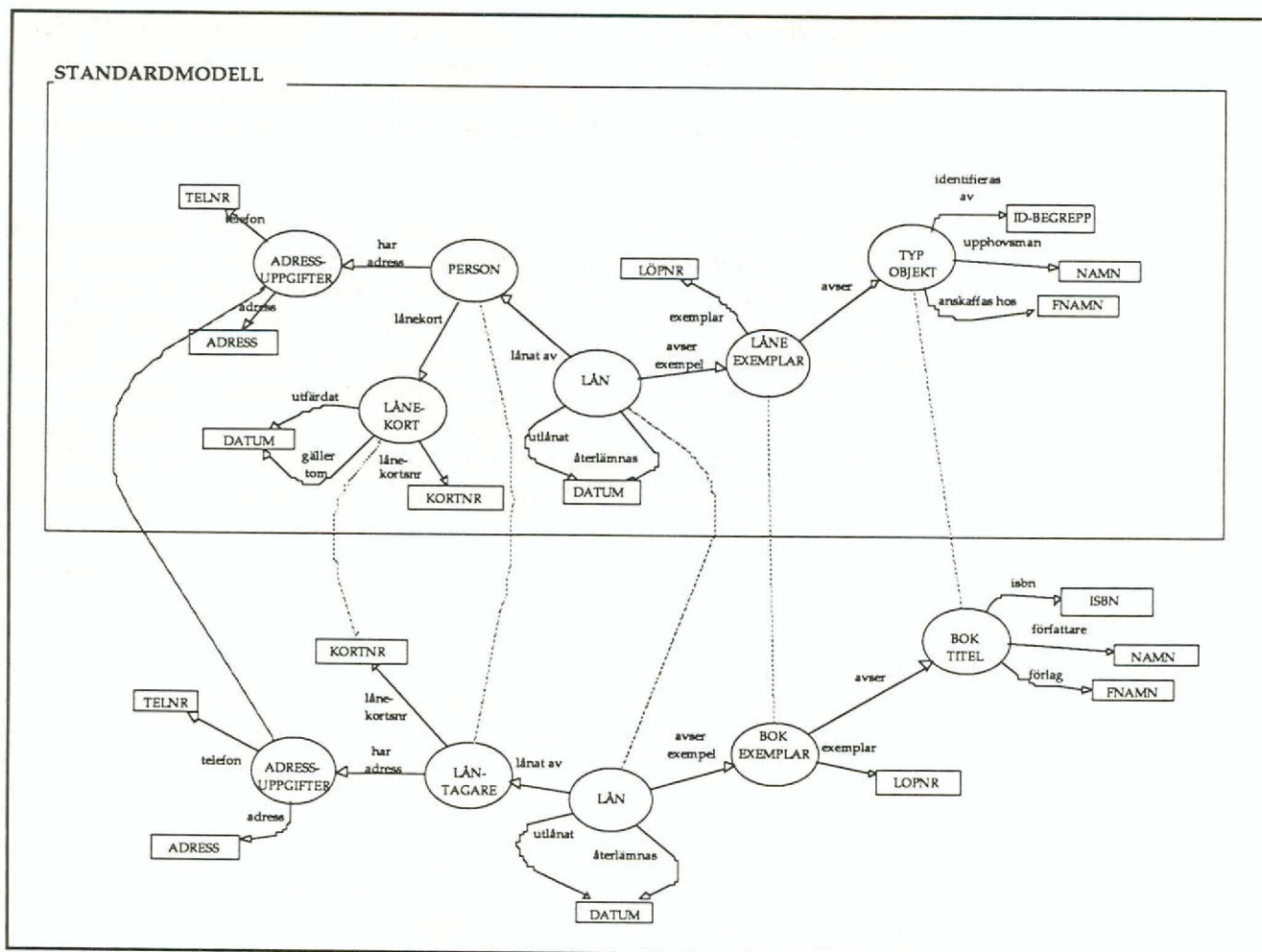
4.2.2 Schemats överensstämmelse med verkligheten

Då vi som människor bedömer riktigheten eller kvaliteten hos t ex
ett konceptuellt schema, gör vi det utifrån den tolkning vi gör av
schemat mot bakgrund av kunskap vi har om den begreppsvärld som
används. Kunskap om verksamhetsområdet är med andra ord av
central betydelse vid kvalitetsdiagnostik. Framför allt är det fråga
om huruvida de begrepp som förekommer i schemat används på det
sätt som är normalt inom verksamhetsområdet.

Semantisk korrekthet

Här är det fråga om huruvida schemat avbildar det avsedda verk-
samtetsområdet på ett korrekt sätt, dvs om det ansluter till vad som
är brukligt inom området eller vad som tidigare befunnits eller
fastställts vara rätt och riktigt. För att detta ska kunna avgöras av
ett datorstöd, krävs att verksamhetskunskap, t ex i form av tidigare
producerade scheman, finns representerad i dess kunskapsbas,
liksom att det finns möjlighet att koppla nya begrepp till redan
kända. Det senare kan kanske ske i form av generiska kopplingar till
de tidigare kända begreppen på det sätt som indikeras i Fig 2 nedan.
Avvikelse från 'standard' kan då upptäckas och ifrågasättas av
verktyget.

I exemplet har vi en standardmodell för en låneverksamhet som är
generell för att kunna gälla alla möjliga låneverksamheter. En
lånemodell för ett litet bibliotek som endast lånar ut böcker kopplas
mot denna för att man ska se att alla nödvändiga objekt finns med
- därmed inte sagt att man väljer att ha med alla objekt i standard-
modellen, se kopplingen till LÅNEKORT i figuren.



Figur 2. En lånemodell i förhållande till sin "normalmodell".

Fullständighet

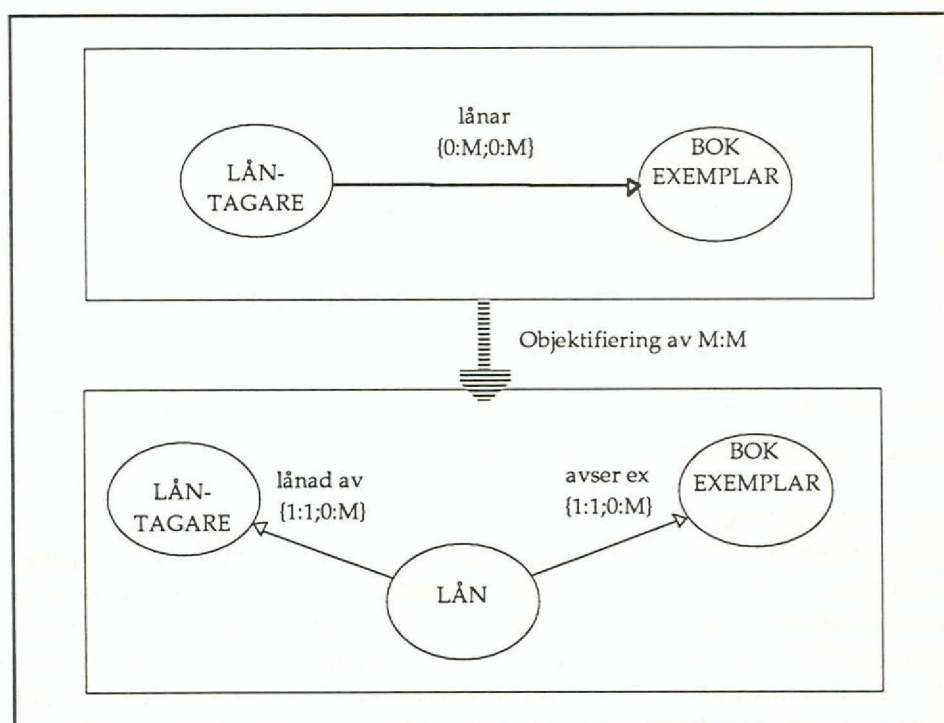
Här gäller det i vilken utsträckning schemats innehåll täcker objekt-systemet, dvs hur stor del av verksamhetens regler (business rules) som finns representerade i schemat. Ett möjligt sätt att komma åt detta skulle vara att jämföra schemat med någon annan beskrivning av objektsystemet. T ex genom att, som nämns ovan, låta verktyget generera alla möjliga utsagor i naturligt språk utifrån modellen. Dessa utsagor skulle sedan granskas av verksamhetsexperter i en process som i detta fall främst syftar till att kontrollera att allt finns med.

Stabilitet/flexibilitet

Naturligtvis vill man också att införda begrepp ska vara stabila och inte förändras över tiden liksom att det ska vara lätt att införa förändringar när det blir nödvändigt, dvs att en förändring i omvärlden föranleder små förändringar i schemat. Att bedöma stabilitet över tiden förutsätter dock vissa överväganden beträffande den framtida utvecklingen inom verksamhetsområdet. Hur detta ska kunna automatiseras är svårt att se. Rent allmänt torde dock gälla att objekttyper på hög generaliseringsnivå är stabilare än mer specialiserade. Vid en stabilitetsanalys skulle därför objekt kunna

klassificeras efter generaliseringsnivå. De mest specialiserade objekten skulle därefter kunna granskas mer i detalj. Man kan också tänka sig att ha scheman på olika abstraktionsnivåer. Samma modell kan då ses på olika nivåer (Se även avsnittet om komplexitetsnivåer).

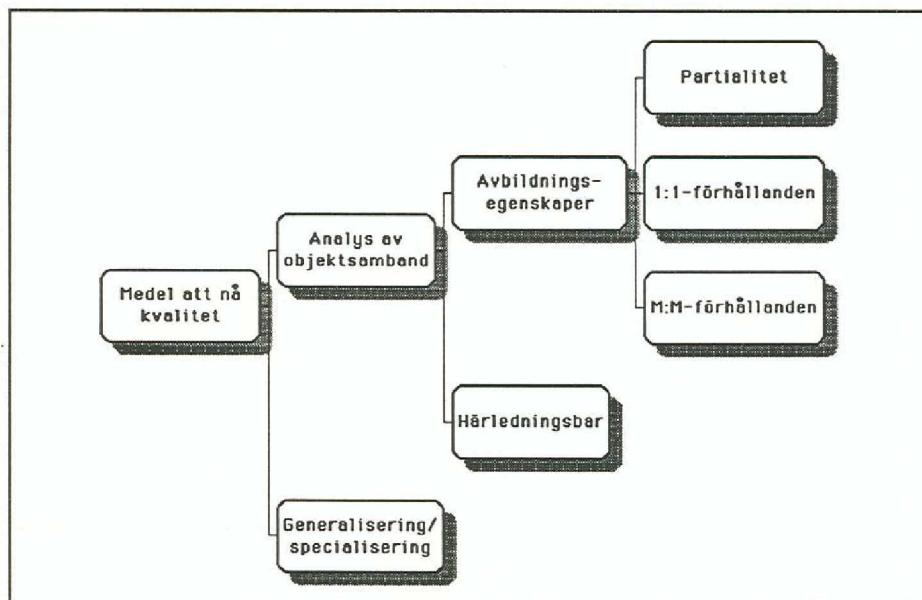
Hög flexibilitet innebär att ändringar och tillägg är lätta att göra. Allmänt kan väl anses att tillägg av ett attribut är en mindre ändring än tillägg av en ny objekttyp. Detta skulle tala för att man redan från början inför objekttyper mer eller mindre överallt där det möjligen kan komma ifråga, t ex att man objektifierar M:M-förhållanden och specialiserar eller dekomponerar objekt utöver vad som från början förefaller nödvändigt. Figur 3 visar ett exempel på hur högre flexibilitet kan uppnås. Här innebär uppbyggnaden av M:M-förhållandet (många till många) att vi kan lägga till information om varje enskilt lån.



Figur 3. Objektifiering (Uppbrytning) av M:M-förhållande.

4.3 Medel att nå kvalitet hos schemat

Figur 4 nedan beskriver några tänkbara sätt att höja kvaliteten på schemat enligt diskussionen i föregående avsnitt.

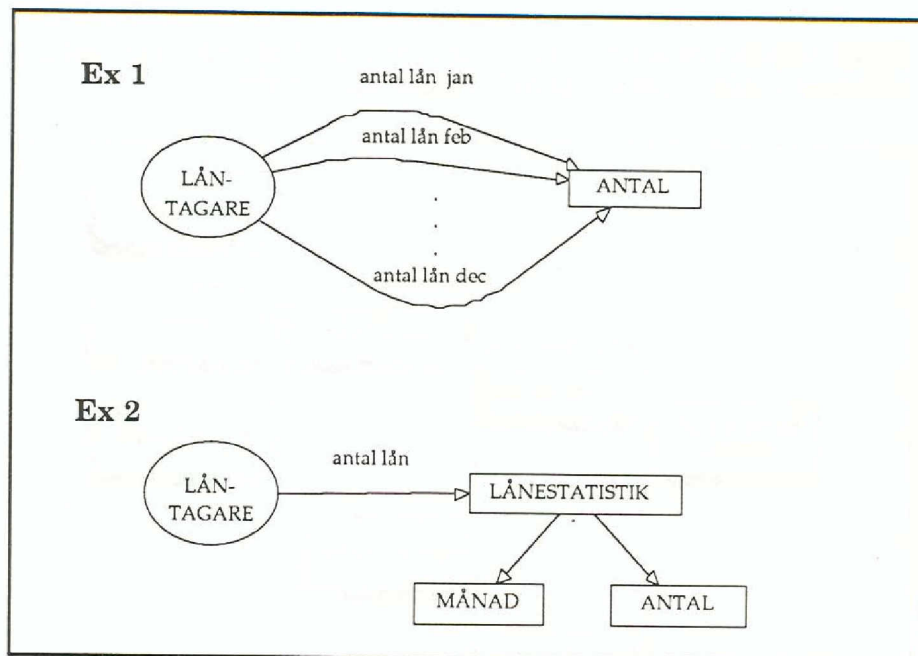


Figur 4. Medel att nå bra kvalitet hos ett konceptuellt schema.

4.3.1 Analys av objektsamband

Med detta avses antal och karaktär hos samband av olika slag, t ex attributsamband och generiska samband. Egenskapen kan analyseras dels genom beräkning av kvantifierbara storheter, som t ex antal och frekvens av olika slags samband och dels genom att man detaljgranskar vissa kritiska typer.

Avbildningsegenskaper



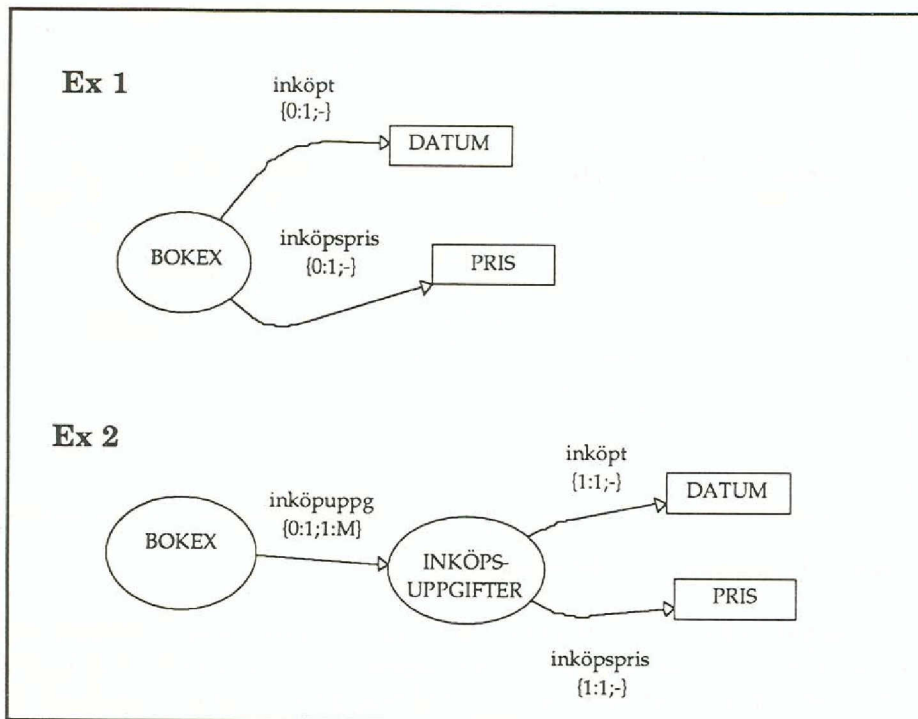
Figur 5. Transformering till lägre komplexitetsgrad.

Värden på de kvantifierbara storheterna utgör en sorts mått på omfång eller komplexitet. Hur de ska användas är tills vidare oklart. I exemplen i figur 5 är attribut/objekt-kvoten, som kan sägas vara ett mått på strukturell komplexitet, $12/2$ i exempel 1 och $3/4$ i exempel 2. De flesta håller nog med om att exempel 2 är en bättre framställning än exempel 1. Den är både klarare och mer flexibel. Det senare eftersom den enklare låter sig utvidgas till flera år.

Partialitet

Med partiella attribut menas de egenskaper som inte är obligatoriska för ett visst objekt. Många partiella attribut kan tyda på att man inte brutit upp modellen tillräckligt, då dessa innebär en osäkerhet i modellen. Se figur 6.

Detta innebär en lägre kvalitet både avseende förståelighet och fullständighet. Många partiella attribut kan också vara ett tecken på att det finns beroenden dem emellan som kan ge upphov till nya objekt, detta kan vara ett problem när det gäller databasdesign.

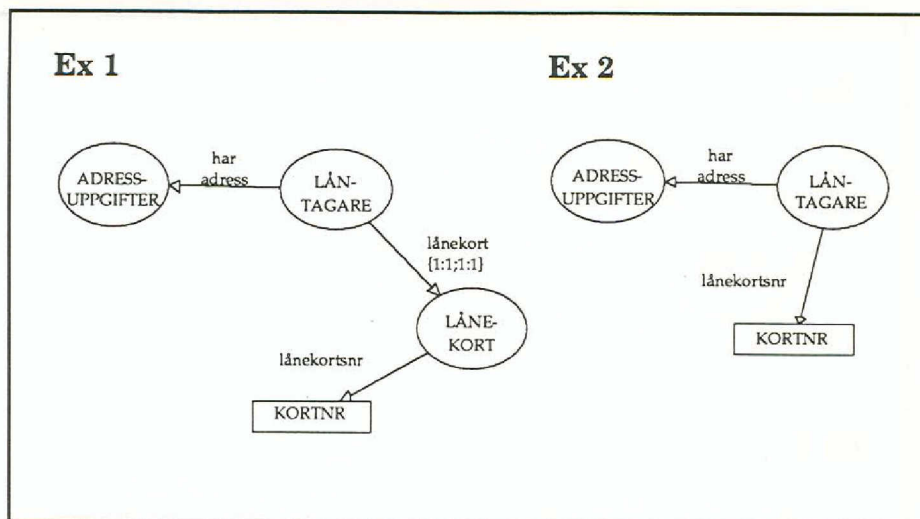


Figur 6. Partiella attribut.

I ex 1 i figur 6 ser vi att ett bokexemplar kan ha ett inköpspris och/eller ett inköpsdatum. Vi ser inte om de har ett inbördes beroende, attributen är båda partiella. I ex 2 däremot visar vi att ett bokexemplar kan ha inköpsuppgifter, om sådana finns måste både inköpsdatum och inköpspris anges. Modellen har ökat i komplexitet men vunnit i semantik.

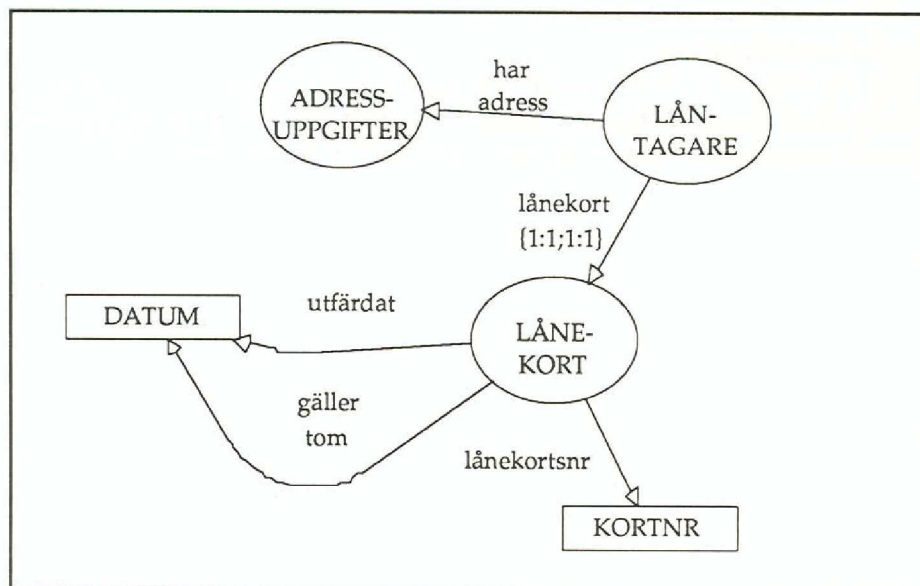
1:1-förhållanden

Ett renodlat 1:1-förhållande mellan entitetstyper kan tyda på att det som egentligen bör betraktas som ett enda objekt har blivit uppdelat i två, något som kan vara, men inte behöver var, tveksamt.



Figur 7. Ej nödvändigt 1:1-förhållande.

I exempel 1 i figur 7 ser vi att en låntagare måste ha ett lånekort, och endast ett. Lånekortet har ett lånekortsnummer. I detta fall kan vi säga att biblioteket representerar låntagaren med ett lånekort. Vi kan med andra ord göra som i exempel 2, ange lånekortsnummer som attribut direkt på låntagaren. Vi måste dock vara försiktiga så att vi inte tappas semantik i schemat. Är det låntagaren som lånar en bok – eller lånekortet? I figur 8 ställs frågan på sin spets då vi har ytterligare attribut på lånekortet. Vi vill här veta när det är utfärdat, respektive till vilket datum det gäller. Här är det mindre naturligt att lägga dessa attribut på låntagaren. För tydlighetens skull bör lånekort behållas som objekt.

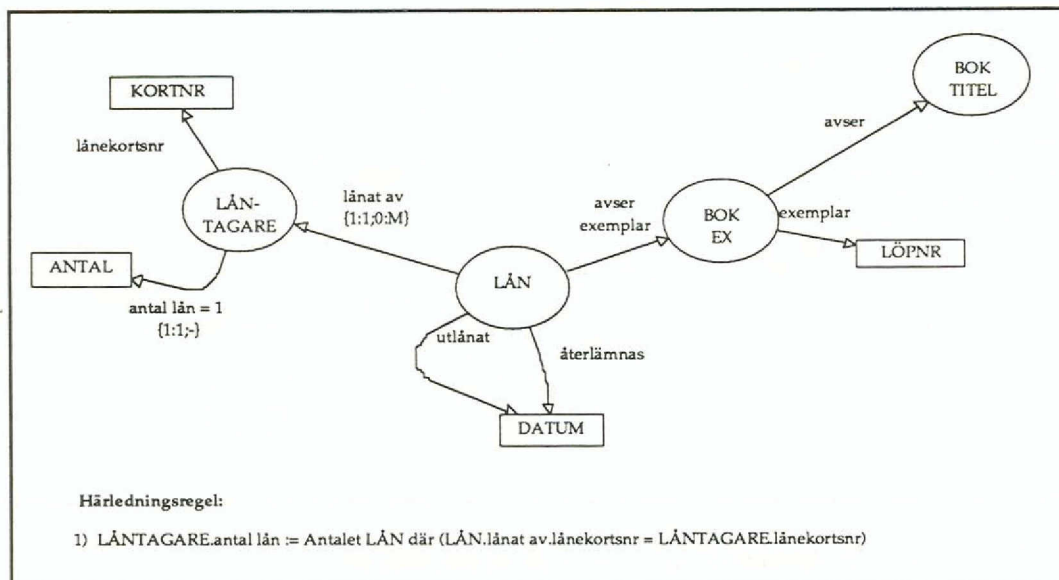


Figur 8. Semantiskt välmotiverat 1:1-förhållande.

M:M-förhållanden

Attribut som är flervärda i pilens riktning kan ge upphov till misstag från normaliseringssynpunkt i samband med databasdesign. Se exempel i figur 2 i avsnitt 3.

Härledbarhet



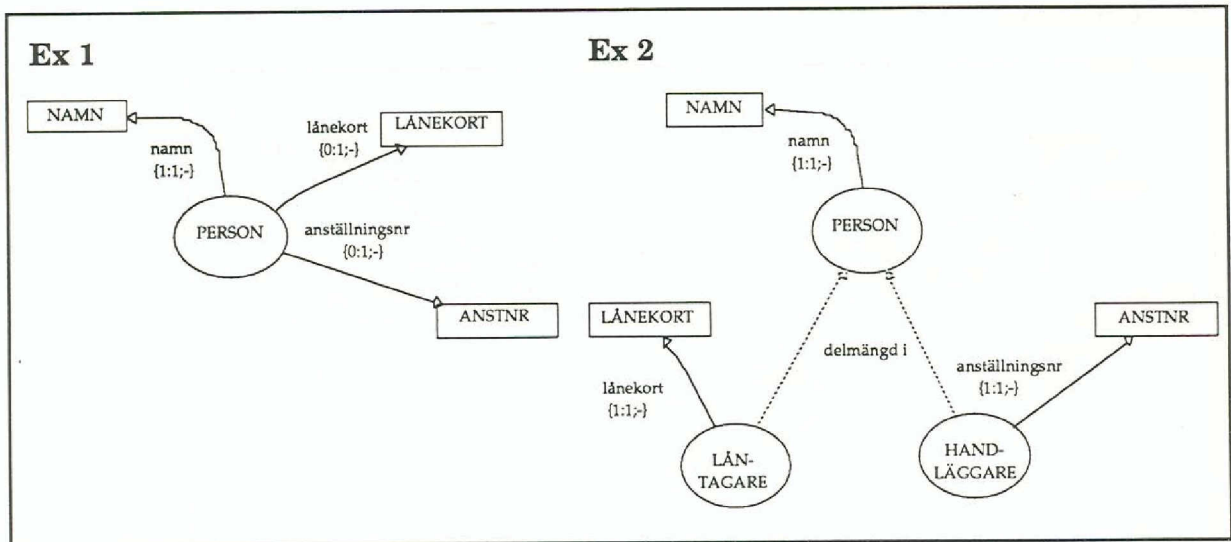
Figur 9. Härledbarhet.

Samband mellan attribut kan ibland formuleras som härledbara. Se t ex figur 9, där attributet "antal lån" kan härledas genom att ackumulera det antal lån låntagaren har (via attributet "har lån").

Om härledbarhet inte markeras i schemat kan det ge upphov till redundans i den så småningom producerade databasen. I figur 9 har vi markerat "antal lån" genom att placera ett likhetstecken och en hänvisning till ett härledningsuttryck efter namnet. Det vore önskvärt om diagnosticeraren kunde hjälpa till att upptäcka härledbarhet i sådana fall då härledbara storheter införs i modellen utan att det markeras. Det är dock för ögonblicket svårt att se hur detta skulle gå till. Härledbara objekttyper kan sägas motsvara det man i databassammanhang brukar kalla externa vyer.

4.3.2 Generalisering/specialisering

Vad gäller generiska samband kan man också tänka sig att verktyget ger förslag till möjliga generaliseringar eller specialiseringar utifrån attributlikhet eller partialitet i attribut.



Figur 10. Specialisering för att undvika partiella attribut.

Vi ser i exemplet att en person kan ha lånekort och/eller anställningsnummer. Genom att specialisera preciserar vi de roller en person måste inneha för att ha respektive attribut. Vi får då totala attribut som ger schemat mer semantik.

Vi kan också tänka oss att verktyget ser likheter i attribut för olika objekt och föreslår ett generellt objekt som en generalisering av dessa.

5. Diagnosverktygets uppbyggnad

5.1 Idé och önskade egenskaper

Ett viktigt syfte med analys av kvalitetsbegreppet, sett ur diagnosverktygets synvinkel, är att få en grov uppskattning av vilka typer av fel eller tveksamheter som är mest talrika. Med hjälp av denna uppskattning kan man sedan fokusera diagnosen på just dessa aspekter i syfte att avlägsna de värsta felen först. Förutom att detta leder till högre effektivitet vid konstruktionen av schemat, innebär det också en kvalitetsvinst för användaren, genom att denne tidigt kan upptäcka att han är på fel spår och därigenom slipper onödigt specificeringsarbete.

Diagnosticering av schemat tänkes ske utifrån de egenskaper som nämnts ovan. Många av dessa låter sig dessvärre inte datoriseras så enkelt. Detta gäller framförallt de som har att göra med kunskaper om verksamhetsområdet. En möjlighet är här att systemet frågar användaren, då det inte har tillräcklig information om området. Kontroll av redundans i schemat är ett exempel på en sådan egenskap.

Egenskapsanalysen kan i vissa fall utgöra underlag för heuristiska regler som resulterar i förslag till förbättringar. Dessa heurismer kan delas in i grupper, där varje grupp avser en viss typ av misstänkt fel. Varje grupp får sedan en prioritet som beror på hur fundamental den anses vara. Avgörande för detta kan t ex vara att man först vill få bort de fel som har de värsta konsekvenserna eller de som är flest till antalet. De grupper av heurismer som har den högsta prioriteten blir sedan de som behandlas först vid diagnosen.

5.2 Expertsystem, vad är det?

Eftersom diagnosstödet är ett expertsystem kan det här vara på sin plats med en utförligare förklaring av ordet. Expertsystem är en term som ofta används ganska vårdslöst. Ursprungligen betecknade termen ett system som försökte simulera beteendet hos en expert inom ett visst begränsat område. För att åstadkomma det använde man en speciell teknik som brukar kallas KunskapsBaserade System (KBS). Dvs, har man inte ett KBS som kan sägas simulera beteendet hos en expert så är det inget expertsystem, utan det är ett KBS. Ett KBS är ett programpaket med ett speciellt tolkningsmaskineri som kallas inferensmaskin. En inferensmaskin kan tolka kunskap, vanligtvis uttryckt som regler, och applicera den på data. Den vanligaste formen av KBS använder en typ av regler som kallas produktionsregler (om <villkor på data uppfyllt> så <slutsats om data>). För att man ska kunna dra så många slutsatser som möjligt från sina data så lönar det sig ofta att försöka göra inferenser (slutsatsdragning) både framlänges och baklänges.

Framåtinferens går till så att man går igenom alla regler och ser efter vilka regler som har uppfyllda villkor för det tillgängliga datat. Då är också regeln uppfylld och man kan dra slutsatsen som står i slutet på regeln. Dessa nya slutsatser kan sedan adderas till datat och proceduren upprepas.

Vid bakåtinferens börjar man med en slutsats och försöker uppfylla villkoret i motsvarande regel. För att uppfylla villkoret kan man behöva en slutsats från en annan regel som i sin tur kan behöva ytterligare en annan slutsats osv. På så sätt kan man bevisa en slutsats som annars inte kunde hittas, på grund av att beräkningen med framåtinferens tar för lång tid. Med andra ord, bakåtinferens gör att man kan koncentrera sökandet till det som behövs för att dra de viktiga slutsatserna. Vid framåtinferens bevisar man alla slutsatser, även sådana man inte är intresserad av och som inte behövs för att härleda andra viktiga slutsatser. Eftersom beräkningstiden ökar exponentiellt med antalet regler och data är det viktigt att man inte gör för många onödiga inferenser.

5.3 Expertstödet utseende idag

Expertsystemet är implementerat i Prolog, som ett produktionssystem med ett antal regler. Varje heurism motsvaras av minst en regel. För att minska tolkningssvårigheter har syntaxregler alltid den högsta prioriteten. Verksamhetsregler kontrollerar att de verksamhetsspecifika konventionerna följs. Av skäl som nämns nedan har vi för närvarande inte implementerat denna typ av regler. Allmänna kvalitetsregler representerar kunskap om vad som allmänt kan anses vara bra kvalitet hos ett schema.

Vid utvecklingen av diagnosstödet har vi hittills koncentrerat oss på konstruktionen av heurismerna eftersom det är dessa som upptäcker brister i schemat, vilket ju är diagnosens huvudsakliga syfte.

Nuvarande implementering läser ett schema ur RAMATIC:s databas och översätter detta till Prolog-predikat. Utifrån dessa arbetar sedan heurismerna.

I nuvarande version av systemet har vi inte lagt in någon interaktion med användaren. Innan vi gör detta vill vi först lösa problemet att representera verksamhetskunskap, så att detta kan tas med i diagnosen. Bland många tekniker för att bygga upp verksamhetskunskap, studerar vi speciellt induktion som ett medel att åstadkomma hierarkiska kunskapsstrukturer. Dessa kan i sin tur användas för att applicera generell verksamhetskunskap på ett specifikt schema.

Reglerna i expertsystemet har en relativt restriktiv syntax. Det har fördelen att även andra än systemkonstruktören kan förstå dem, eftersom man inte behöver ha någon djupare kunskap om diagnossystemets uppbyggnad. Som exempel kan vi ta en regel som upptäcker ett syntaxfel:

"här ska det stå en lämplig text att skriva ut till användaren"

Rule Id A23'

If

Name Of Attribute Is Containing_upper_case_letter

Then

Name Of Attribute Violates_this_rule.

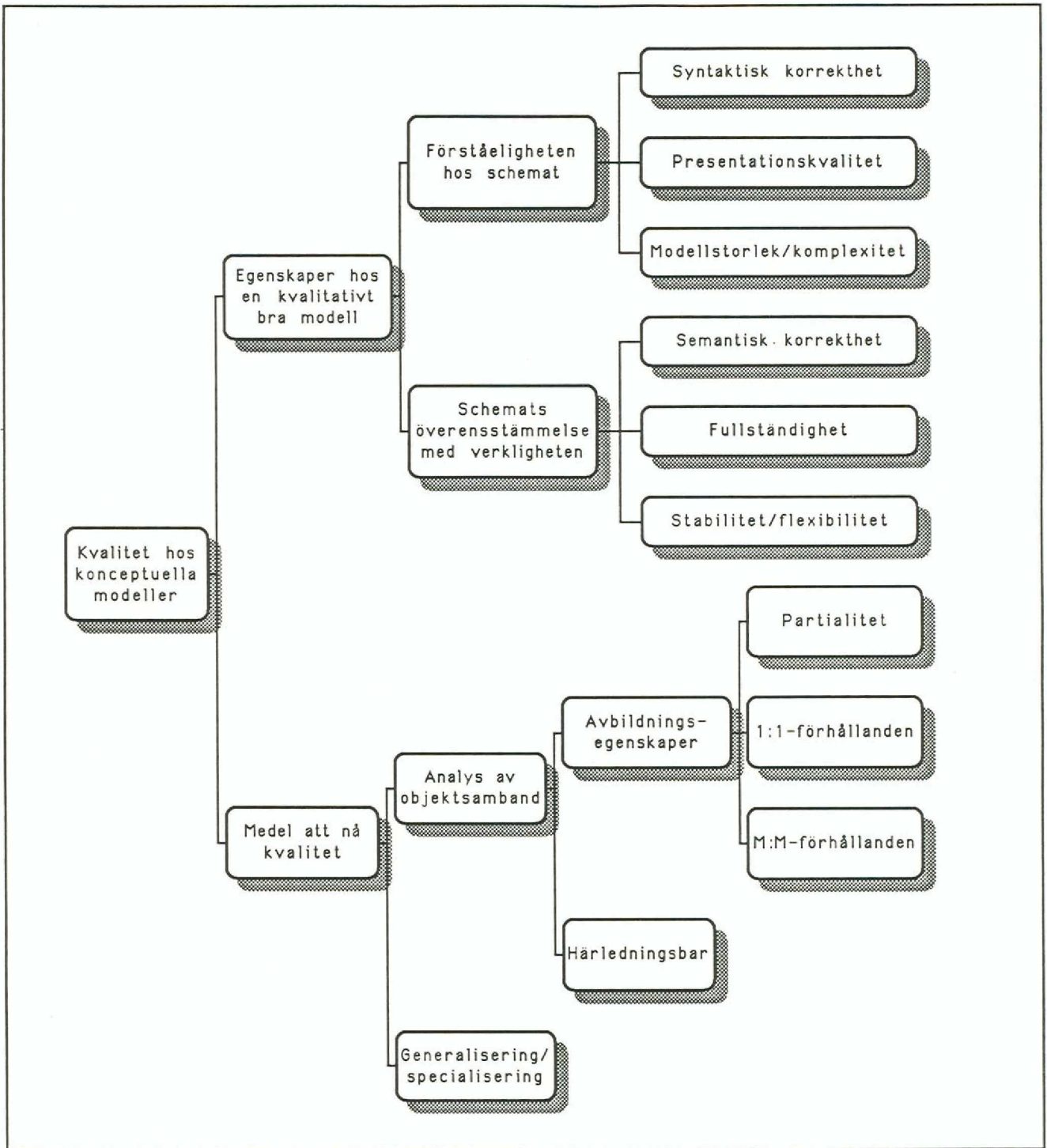
Den här regeln kommer att upptäcka alla attributnamn som innehåller en eller flera stora bokstäver och etablera detta som faktum i diagnosen.

Naturligtvis kan det också vara motiverat att etablera andra fakta än sådana som rör rena felaktigheter. Det kan t ex vara så att vissa företeelser i schemat förutsätter andra. Detta innebär att en slutsats i en regel måste ingå som villkor i en annan. Sådana kedjor av regler medför att också mer komplicerade kontroller kan utföras. I framtida versioner av verktyget kan slutsatsdelen även tänkas innehålla operationer som t ex att rita förslag till användaren på dennes skärm.

6. Slutsatser

Som nämnts har vårt arbete hittills varit inriktat mot konceptuell modellering. Vi tror dock att mycket av våra resultat kan generaliseras eller modifieras så att de även kan gälla andra slag av specifikationer, t ex verksamhetsbeskrivningar eller programspecifikationer. Naturligtvis spelar även här den använda formalismen stor roll.

Det är vår uppfattning att verktygsanknutna analys/diagnoshjälpmedel av den typ som här har diskuterats är möjliga att konstruera och att de kan vara till stor hjälp vid utveckling av informationssystem. Särskilt viktig anser vi den semantiska aspekten vara. Dessvärre är det också den som ställer sig svårast att implementera på ett från databehandlings- och användningssynpunkt effektivt sätt. Att lösa detta problem är en betydande utmaning för framtiden.



Figur 11. Egenskaper hos konceptuella modeller och medel för att nå kvalitet.