

Krav på IA
Nästa Generation Modellering
Avancerad utbildning för handledare
Katalogprinciper
Verktyg
Informationspridning

Rapport K nr 1: IRDS
Rapport K nr 2: IRDS Modeller och modellnivåer
Rapport K nr 3: Kopping begreppsmodell - relationsmodell
Rapport K nr 4: IBM:s Repository Manager- en Introduktion
Rapport K nr 5: IBM:s Repository Manager: Datamodelleringsbegreppen
Rapport K nr 6: IBM:s Repository Manager: Begreppsmodellering i Information Model
Rapport K nr 7: IBM Repository Manager: Attribut- och värdemodellering i Enterprise Submodel

IBM Repository Manager: Attribut- och värdemodellering i Enterprise Submodel

Stig Berild

Rapporten är skriven i och för TRIAD delprojekt Katalogprinciper.

Spridningsförbehåll:

Denna rapport får endast spridas och användas inom de organisationer som deltar som parter i TRIAD-projektet.
© TRIAD-parterna 1991

Innehåll

IBM Repository Manager: Attribut- och värdemodellering i Enterprise Submodel

1. Inledning.....	1
2. Attribute Type.....	2
3. Info Type.....	4
4. Value.....	8
5. Symbol och Symbol Set.....	10
6. Värderestriktioner.....	14
7. Sammanfattning.....	18

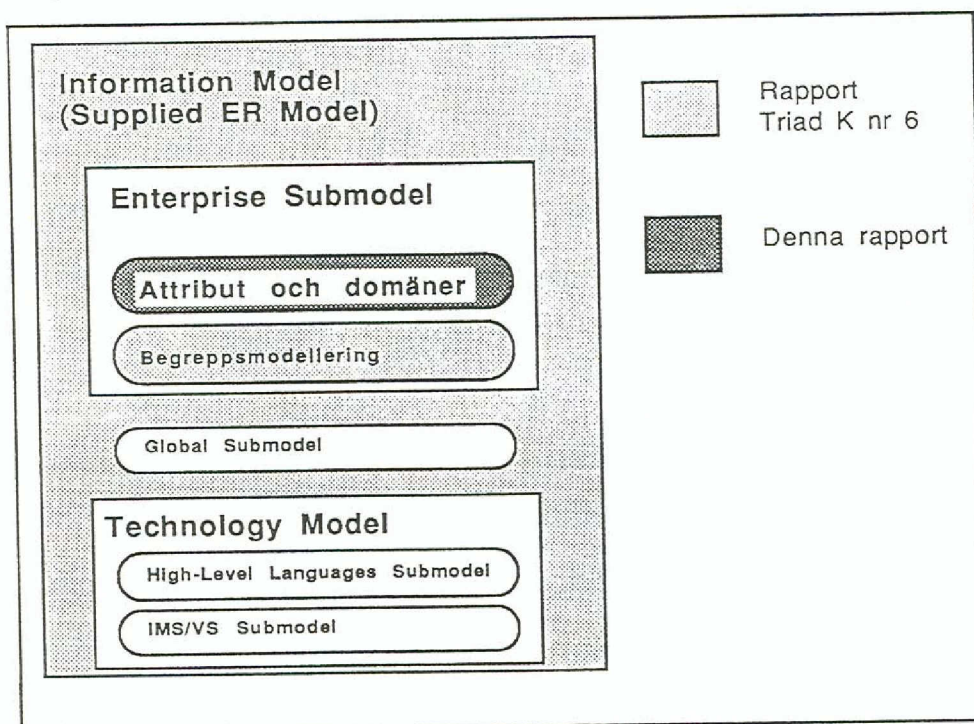
1. Inledning

Rapporten [Triad K nr 6] ger först en allmän översikt över Information Model (IM) i Repository Manager (RM). Därefter ägnas huvuddelen av rapporten åt de entity types och attribute types som är till för att dokumentera begreppsmodeller. Denna del av IM är förhållandevis avancerad och i vissa delar definierad till en detaljeringsgrad som sällan diskuteras eller åtminstone sällan ses exakt preciserad. Dit hör hanteringen av attribut samt deras avbildning på domäner och värden.

I allmänhet noterar man i dessa sammanhang endast distinktionen mellan attribut och domän, ibland ges inte ens attribut någon semantisk mening. I vissa ansatser noteras behovet av att skilja på den representation, som används vid lagring av attributvärden och den som används för olika presentationsbehov (intern - extern representation). IM arbetar med ytterligare preciseringar. Vi har ansett dessa värda en separat rapport, dels för att de inte är helt självförklarande, dels för att visa på en genomtänkt struktur. Den kan mycket väl tjäna som grund för fortsatta diskussioner inom detta, ännu inte färdigpenetrerade, område.

Vi använder samma terminologi och grafiska notation som förklarats under avsnitt 2 i [Triad K nr 6].

Figur 1 visar var inom IM denna rapport hör hemma.



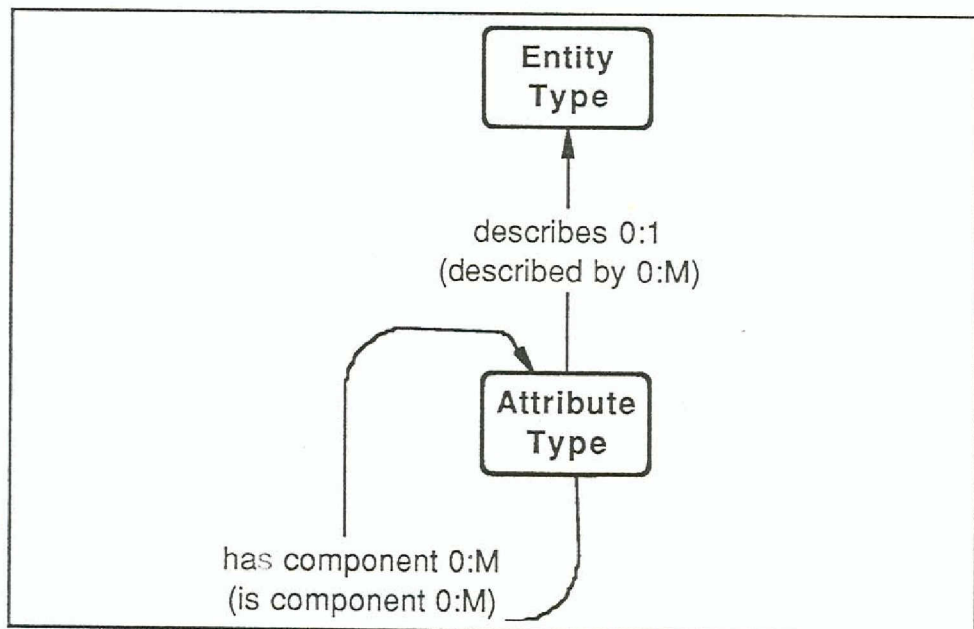
FIGUR 1

2. Attribute Type

I [Triad K nr 6] konstaterade vi att ATTRIBUTE TYPE används för att beskriva ENTITY TYPE med hjälp av värden och symboler. En <attribute type> beskriver endast en <entity type>. Tillfälligtvis tillåts en <attribute type> att även existera utan befintlig DESCRIBES-koppling till viss <entity type>. Viktigt att komma ihåg är att <relationship types> inte kan beskrivas med <attribute types> i IM.

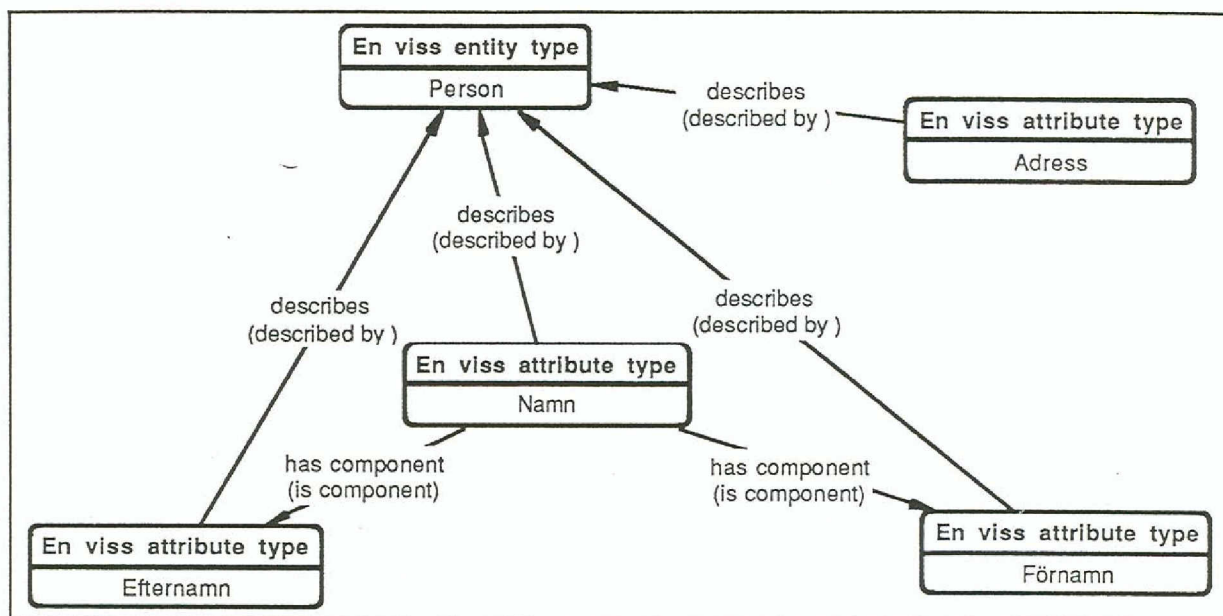
Som synes kan en <attribute type> vara sammansatt av ett antal <attribute type>-komponenter ner till valfritt djup. Exempel ges i avsnitt 4.1 i [Triad K nr 6]. Intressanta attribute types för ATTRIBUTE TYPE är, förutom dess **name** och **description**, bla **minimum** och **maximum** förekomster per <entity type>. Namnet måste vara unikt per <entity type>.

Det kan vara värt att notera att komponent-<attribute types> även måste kopplas som vanliga <attribute types> till samma <entity type>, som den sammansatta <attribute type>. Med hjälp av egenskapen *ordered* för en relationship type, kan sorteringsordningen mellan komponenterna över relationship type HAS COMPONENT i viss mån styras, vid behov. Figur 2 visar schemat så långt.



FIGUR 2

Figur 3 visar ett exempel på förekomstnivå hämtad från [Triad K nr 6]. Av figuren framgår att *person* beskrivs med attribute type *namn* och *adress* men även av namns komponenter *förnamn* och *efternamn*.



FIGUR 3

3. Info Type

Samma värde kan uttryckas på olika sätt. Exv heter saker olika på olika språk. Kanske vill man använda en förkortning till en specialist men fulla benämningen till en oinsatt. Kanske vill man presentera värdet genom olika media (text, tal, bild ...). Oavsett formuleringssätt är det, för den som förstår språket, fråga om samma "värde" i semantisk mening. En uppsättning dylika abstrakta värden, som på goda grunder kan antas vara tolkbara i en verksamhet genom någon av de representationsformer, som används (under förutsättning att man kan det aktuella "språket"), definieras i IM-schemat under beteckningen INFO TYPE. Grundtanken är mao att separera värden från dess symbolmässiga representation.

Den som så önskar, skulle kunna se en <info type> som en förenklad form av <entity type>. I förenklingen ligger att det enda som beskrivs hos en <info type> är en eller flera alternativa identifieringar, som var och en utgörs av endast en <attribute type>. Med detta betraktelsesätt är det naturligt att också uppleva <info types> som självständiga företeelser, tillgängliga för många <attribute types> att referera till. Låt oss återknyta till <info type> *land* från [Triad K nr 6]. Antag att *person* behöver beskrivas med *bosatt i* och *skattepliktig i*. Antag vidare att *företag* beskrivs med *har verksamhet i*. I samtliga 3 fall är det fråga om att ange ett land, för *har verksamhet i* kanske flera länder. I förutsättningen ligger också en verksamhets-gemensam uppfattning om innebörden av begreppet *land*. Modellen blir enligt figur 4.

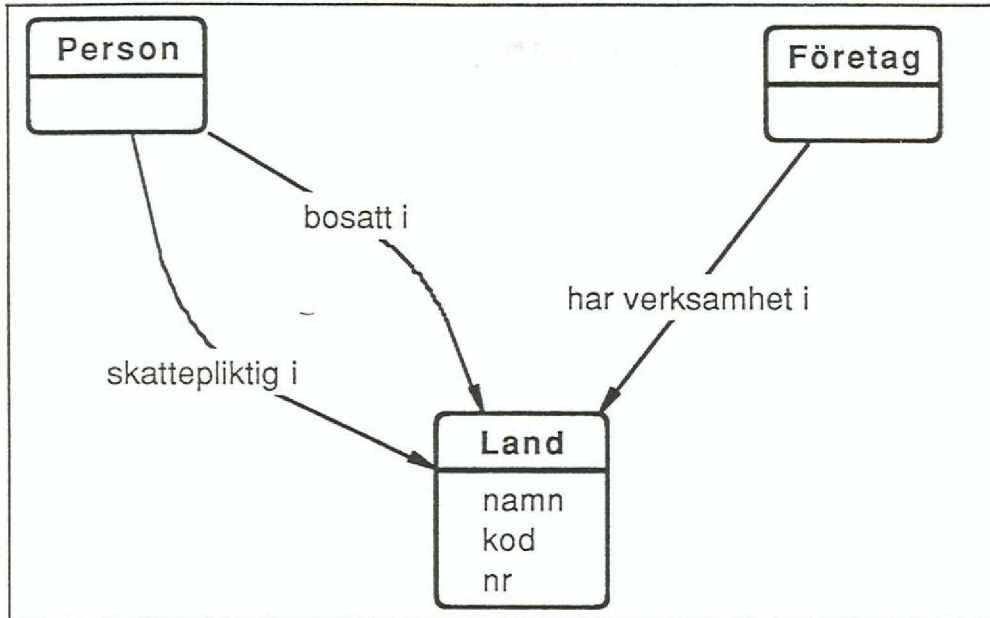


FIGUR 4

Figuren visar 2 <entity types>, 3 <attribute types> och 1 <info type>. Även alternativen enligt figurerna 5 och 6 är acceptabla. Dock bortfaller i figur 5 möjligheten/flexibiliteten att uttrycka samma sak på flera sätt. Man har bundit sig vid en exakt symbolform. Att figuren 6 har noterat *land* som en <entity type> kan som sagt, beroende på vilken "verklighet" som beskrivs, också vara acceptabelt. De fortsatta resonemangen utgår från alternativet i figur 4.

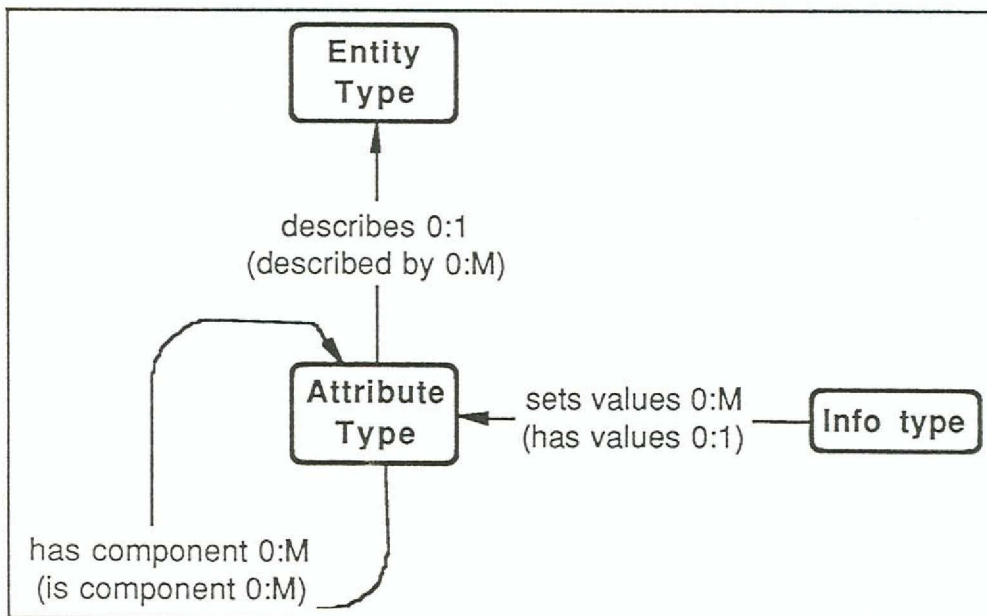


FIGUR 5



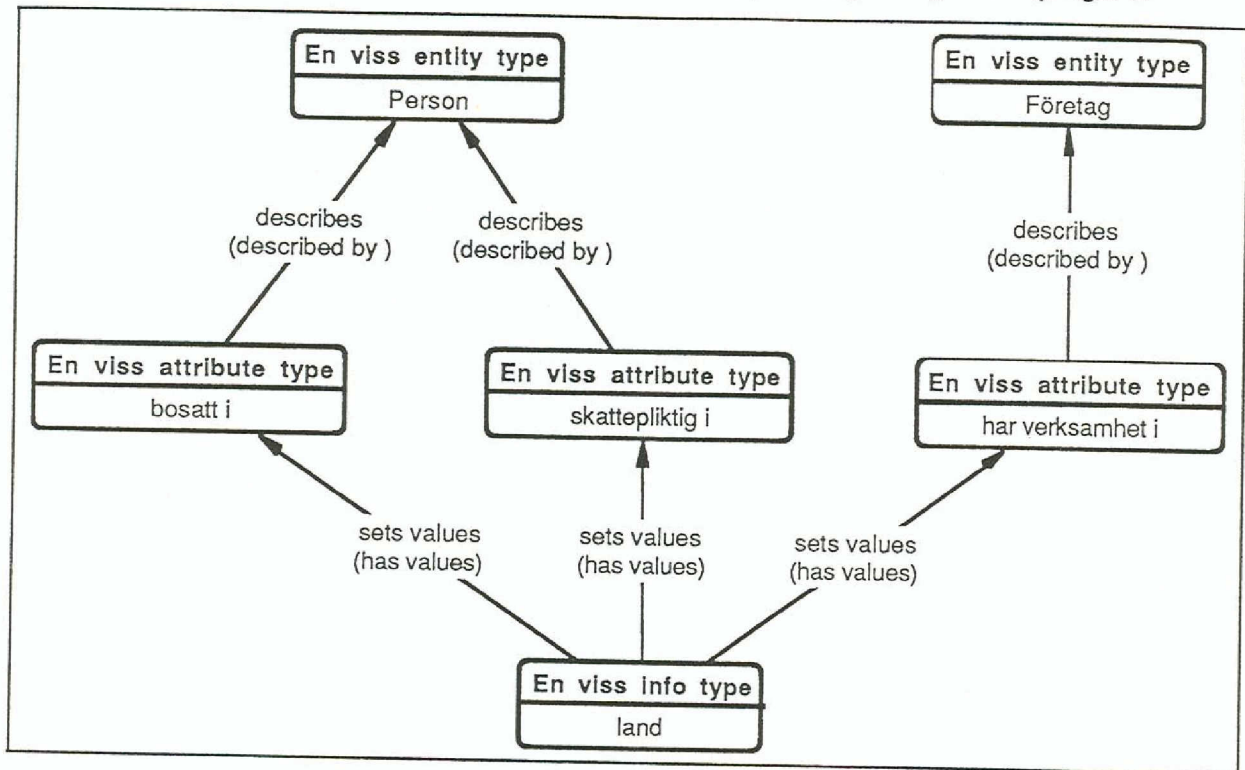
FIGUR 6

INFO TYPE är oberoende av ATTRIBUTE TYPE eller ENTITY TYPE-tillhörighet. Många <attribute types> kan referera till samma <info type>. Vi noterar även att schemat inte föreskriver att varje <attribute type> har en referens till en <info type>, men detta ska endast ses som en frihet under löpande arbetsprocess. Begreppsmodeller som ska bli föremål för realisering i datastrukturer måste rimligtvis kompletteras även med <info type>.



FIGUR 7

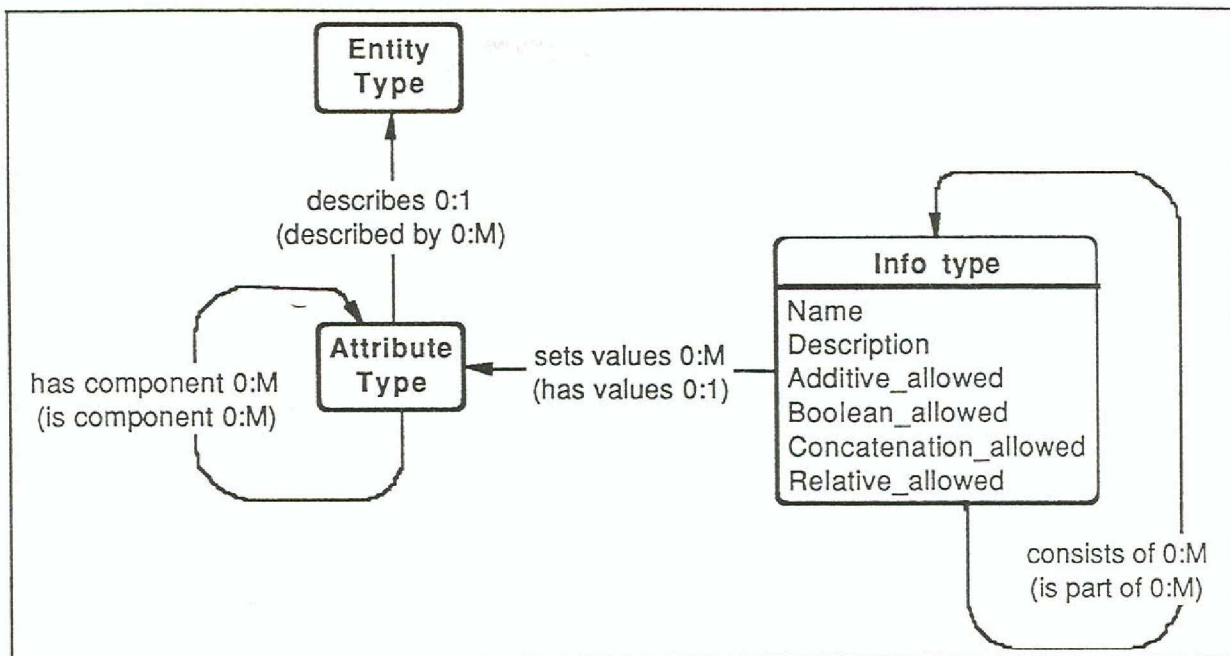
Schemat ger en förekomstnivå baserad på exemplet i figur 4 enligt figur 8.



FIGUR 8

En <info type> kan vara enkel eller sammansatt. En sammansatt <info type> svarar direkt mot en sammansatt <attribute type>. Det betyder att strukturen exakt måste överensstämja mellan de båda. Förutom att en sammansatt <attribute type> kopplas till en sammansatt <info type>, måste varje ingående komponent även parvis kopplas ihop. INFO TYPE-strukturen byggs upp med hjälp av attribute type CONSISTS OF, se figur 9.

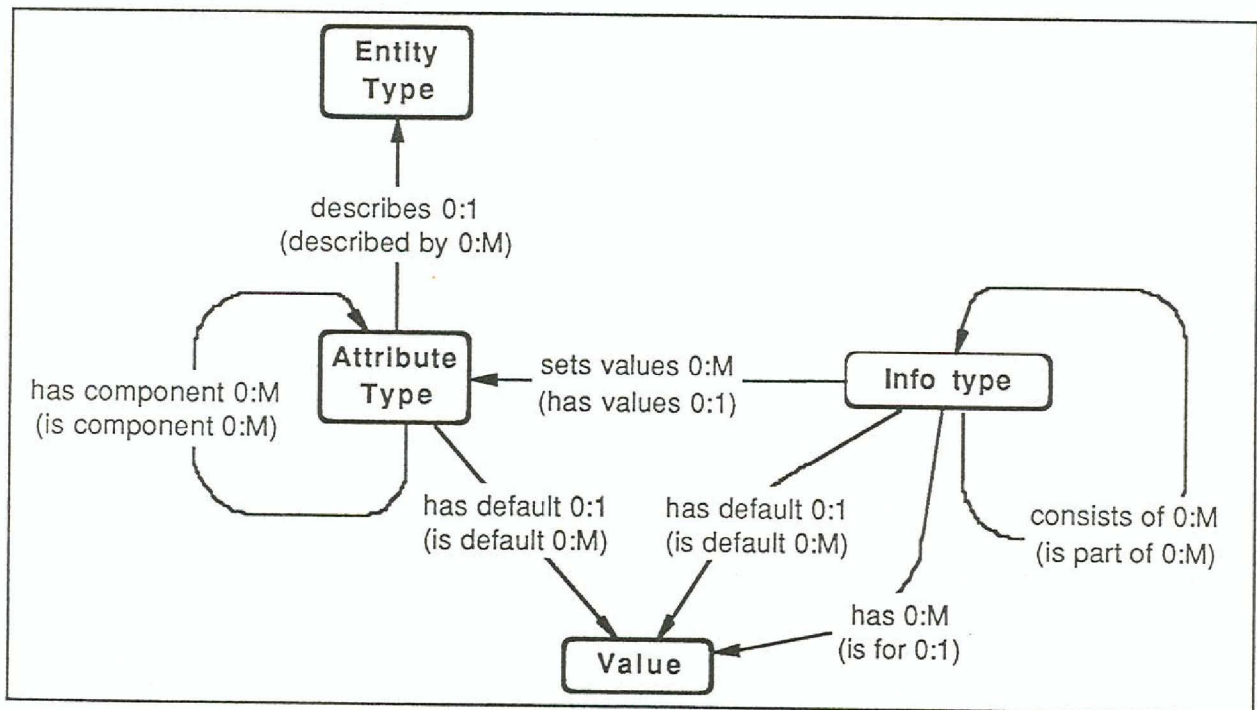
INFO TYPE beskrivs med **name** och **description**. Därutöver finns möjligheter att uttrycka vissa restriktioner på hur värdena inom viss <info type> får användas. Dit hör **additive_allowed**, **boolean_allowed**, **relative_allowed** och **concatenation_allowed**. Anges för dessa Y(es) kan ett värde användas mot additionsoperator, logiska operatorer, relationsoperatorer och sammanlänkingsoperator, respektive. N(o) innebär att operatorerna inte kan användas på denna type av värden. Dessa uppgifter är utmärkta exempel på beskrivning som hör till värdet i sig och inte till dess aktuella symbolrepresentation.



FIGUR 9

4. Value

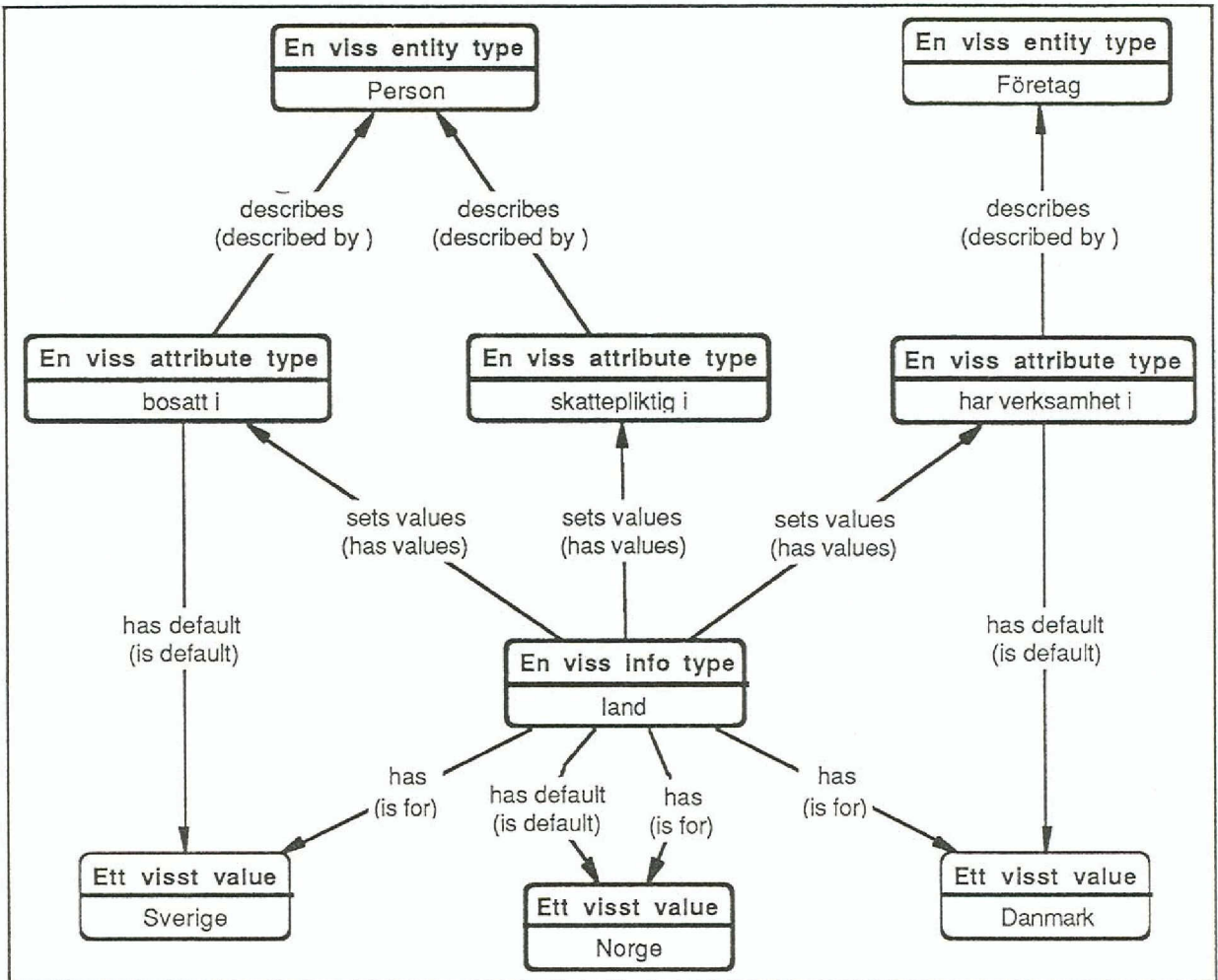
En viss <info type> står som samlingsnamn för en viss uppsättning värden, ett visst värdeförråd. Tillåtna värden kan uttryckas i allmänna termer som restriktioner (se under avsnitt 6) eller genom en explicit uppräkningslista. I det senare fallet finns respektive värde definierat under entity type VALUE. En <info type> kan alltså vara relaterad med ett antal <values> medan ett <value> definitionsmässigt hör till endast en <info type>. IM tycks inte ställa krav på att ett <value> alltid hör till en <info type>, men detta måste nog anses vara normalfallet. Möjligtvis kan värden användas iför att uttrycka <value ranges>, se avsnitt 6, vara exempel på undantag. Som en extra finess finns möjlighet att specificera ett av de uppräknade värdena som standardvärde (HAS DEFAULT), att användas om inget annat sägs. Samma möjlighet erbjuds en <attribute type>, vilket rimligtvis måste innebära att detta, när det specificerats, ges prioritet över det, som anges under relaterad <info type>. Se vidare det utvidgade schemat i figur 10.



FIGUR 10

Om de aktuella länderna för <info type> *land* är Sverige, Norge och Danmark och det normala är att Norge anges, blir det tidigare exemplet utökat enligt figur 11. Där har även angivits att standardvärdet för just *bosatt i* inte ska vara Norge, utan Sverige. Företags normala verksamhet anses vara förlagd till Danmark.

Standardvärde behöver givetvis inte alls anges varken för ATTRIBUTE TYPE eller INFO TYPE. Exempelvis har inget angivits för *skattepliktig*.



FIGUR 11

5. Symbol och Symbol Set

Därmed över till den del av schemat som reglerar hur värden kan representeras. En <info type> kan representeras i form av en eller flera alternativa <symbol sets>. Man har möjlighet att ange vilket <symbol set> som gäller i normalfallet (HAS DEFINED). Övriga, alternativa <symbol sets> relateras via HAS ALTERNATIVE. Enligt manual kan ett <symbol set> höra till flera <info types>. Anledningen till detta verkar något dunkel. Tills vidare konstaterar vi att ett <symbol set> i normalfallet endast hör till en enda <info type>. Vi är nu nere på en nivå där ett stort antal preciseringar behövs för en fullödig beskrivning av representationens uppbyggnad och villkor. Entity type SYMBOL SET beskrivs med ett 30-tal attribute types. Vi berör här bara några få som exempel på vad som beskrivs.

Förutom **name** och **description** finns bla attribute types för att beskriva

typ av symbol

type, alphanumeric_indicator, other type <voice, image, message, ...>

vilket format som gäller

format, null_representation, infinity_representation

speciella numeriska format

fixed_float, precision, rounded_truncated

om varje symbol explicit anges

itemized

bedömd längd i antal tecken

minimum_length, median_length, maximum_lengh

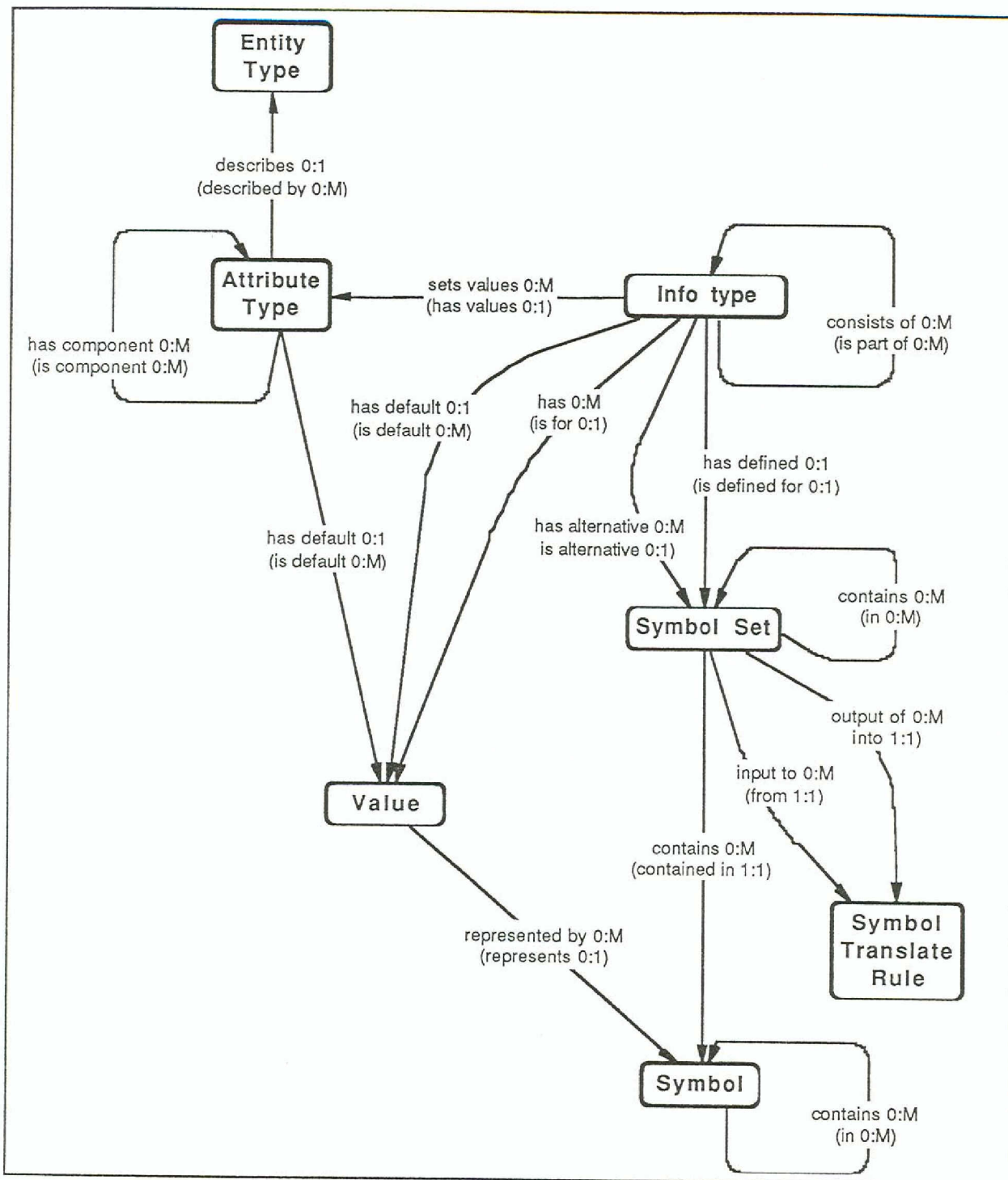
om viss "standard"-symbol ska läggas till symbolen

prefix, suffix

Om ett <symbol set> är "itemized" refererar det till ett antal <symbols> på samma sätt som en <info type> till ett antal <values>. Varje <value> refererar till en eller fler <symbols> som en konsekvens av att en <info type> relateras till en eller flera <symbol sets>. Sammankopplingen mellan VALUE och SYMBOL sker med hjälp av REPRESENTED_BY. Både SYMBOL SET och SYMBOL kan vara sammansatta, dvs ha en komponentstruktur (genom CONTAINS). Vad vi kan förstå står dessa strukturer i överensstämmelse med deras sammansatta motsvarigheter inom INFO TYPE och ATTRIBUTE TYPE.

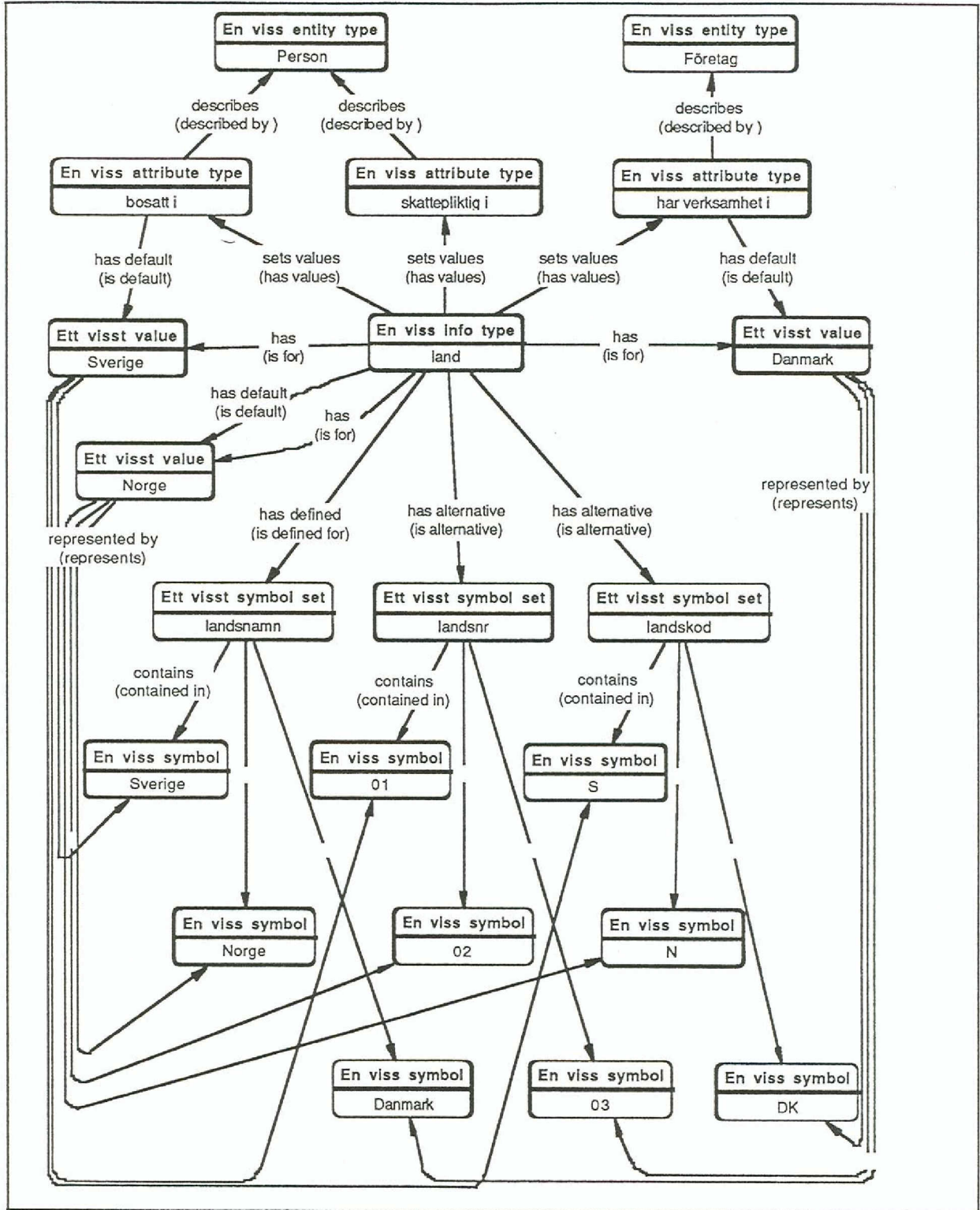
I de fall det finns explicita <values> och dessa har <symbol>-relatering, kan man indirekt få fram att två eller flera specifika symboler står för samma sak, samma <value>. Finns inga explicita <values> och därmed ingen exakt "översättning", återstår möjligheten att ändå relatera <symbol sets> via en översättningsalgoritm. Dyliga specificeras under entity type SYMBOL TRANSLATE RULE. Där redovisas bla algoritmen (**rule_text**) tillsammans med vilket språk den är skriven i (**rule_text_language**). Exempelvis översättning av <info type> *datum* enligt <symbol set> *svensk datumstandard* till <symbol set> *amerikansk datumstandard* beskrivs lämpligen i en <symbol translate rule>. Översättningen specificeras alltid mellan parvisa <symbol sets>, där det ena är input till och det andra output från algoritmen.

Schemat i kompletterat skick visas i figur 12.



FIGUR 12

Fullföljer vi land-exemplet i figur 11 med att inkludera de tre alternativa representationssätten angivna i figur 5 blir resultatet enligt figur 13. Observera att vi av utrymmesskäl utelämnat en del *contains* och *represented by*.



FIGUR 13

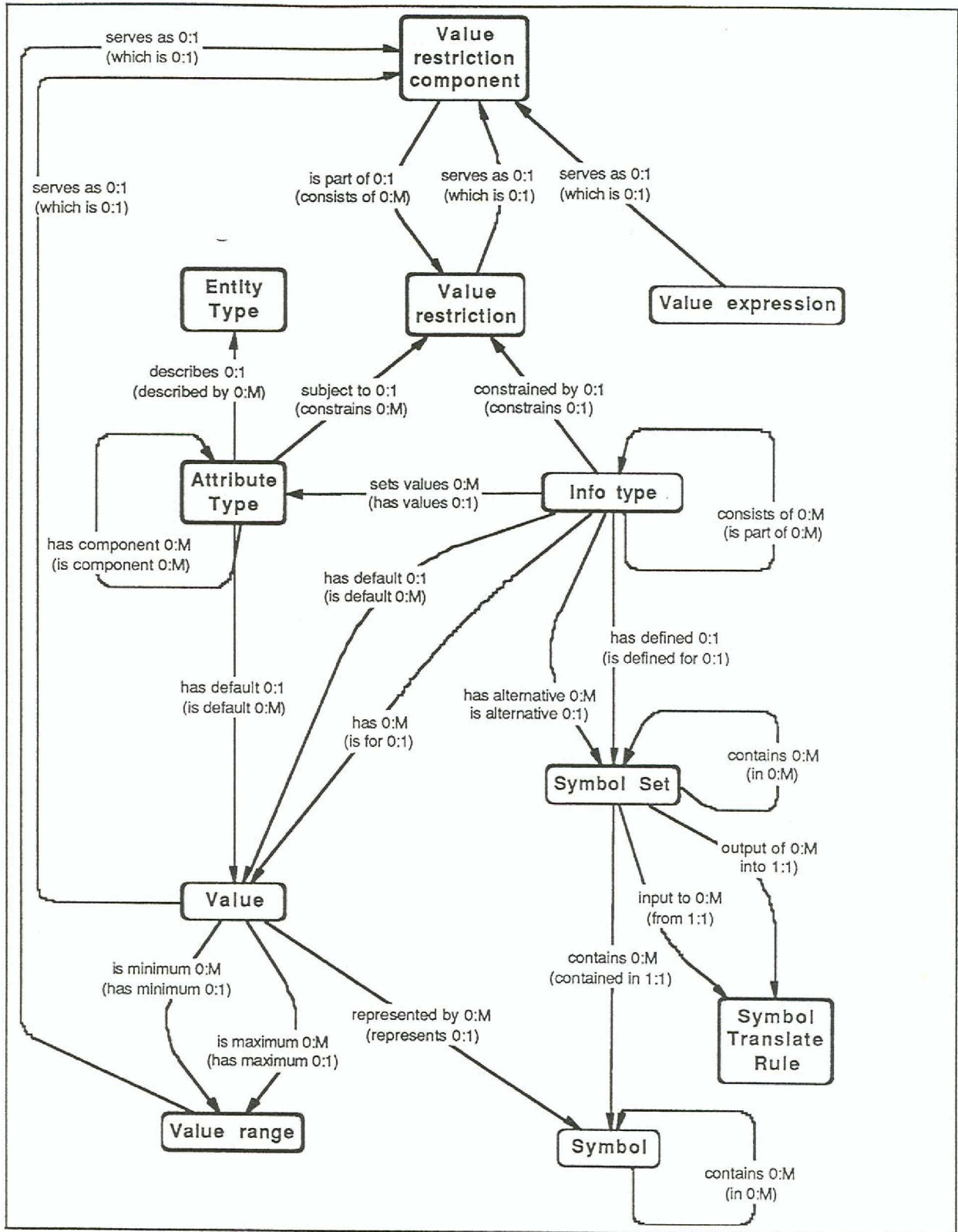
6. Värderestriktioner

Ett sätt att reglera tillåtna värden för en <info type> är att explicit ange dem genom ett antal <value>-relateringar så som avsnitt 4 beskrivit. Även i övriga fall finns möjlighet att reglera tillåtna värden genom entity type VALUE RESTRICTION och dess olika beståndsdelar. En <value restriction> byggs upp av en eller flera komponenter (av typ VALUE RESTRICTION COMPONENT). Varje <value restriction component> formulerar en restriktion. Om två eller fler komponenter specificerats för en <value restriction> kompletteras dess beskrivning med en logisk operator (operator), som ska gälla mellan samtliga komponenter. För att klara behovet av en blandning av logiska operatorer mellan komponenterna, kan en <value restriction> i sig ingå i rollen som komponent (genom relateringen SERVES AS).

VALUE RESTRICTION COMPONENT kan, förutom som VALUE RESTRICTION, alternativt även uttryckas genom ett specifikt värde (VALUE), som VALUE RANGE eller som VALUE EXPRESSION. Även dessa relateras till VALUE RESTRICTION via SERVES AS. En <value range> definierar ett intervall av acceptabla värden. Intervallet har en nedre (HAS MINIMUM) och en övre (HAS MAXIMUM) gräns. Är restriktionen av mer komplicerad art kan en <value expression> formuleras. Där beskrivs villkoret (expression_text) tillsammans med använt språk för formuleringen (expression_text_language).

Restriktioner kopplas naturligt till entity type INFO TYPE men även ATTRIBUTE TYPE kan utsättas för restriktion (genom SUBJECT TO). Denna restriktion bör i så fall ses som en kompletterande restriktion utöver den som gäller för den till en <attribute type> eventuellt kopplade <info type>.

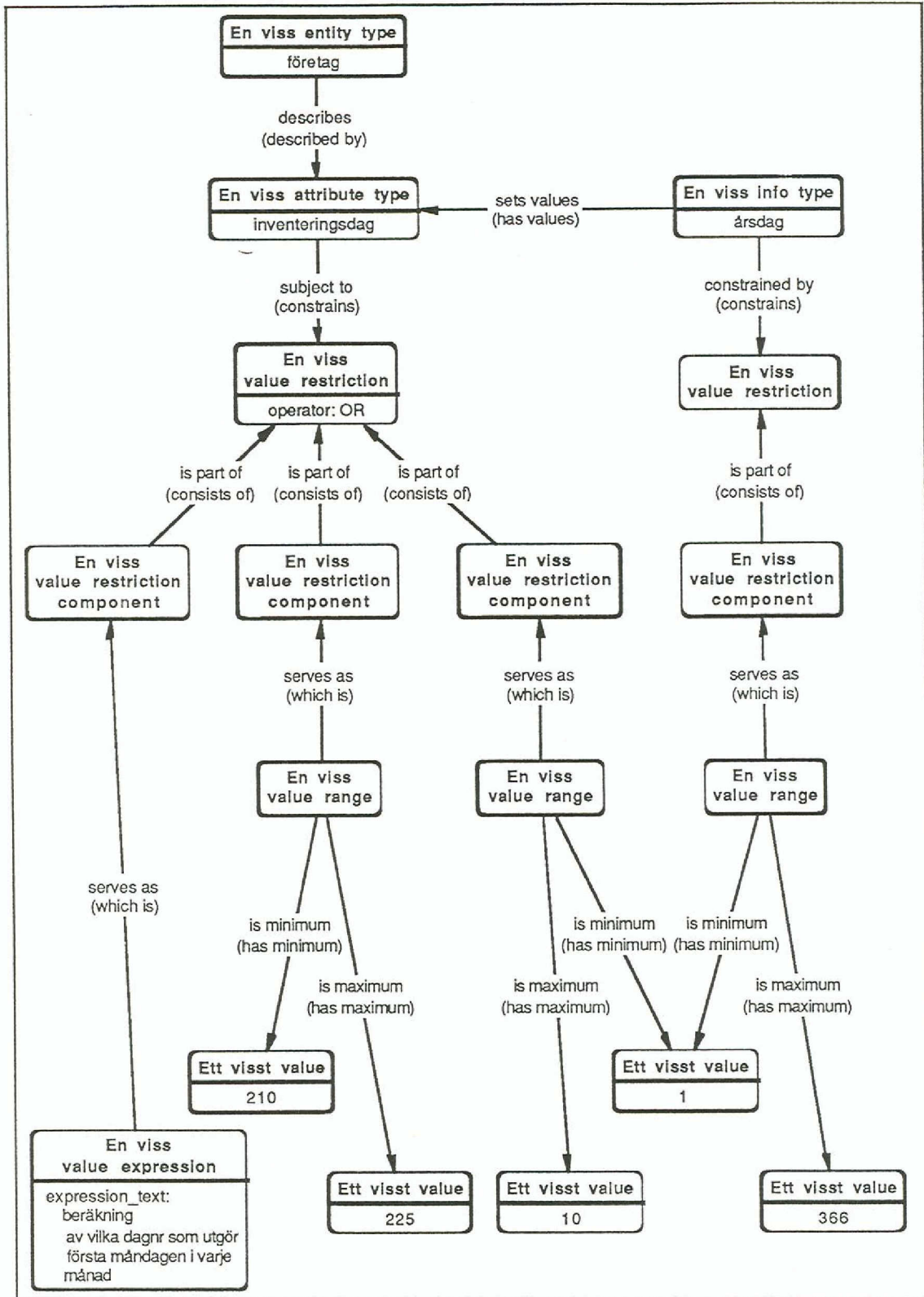
Den slutliga modellen visas i figur 14.



FIGUR 14

Som avslutande exempel tar vi <attribute type> *inventeringsdag* för <entity type> *företag*. Vi bortser för enkelhets skull från symbol-delen. Acceptabla dagar av årets 366 dagar är dels de första 10 dagarna, dels dagarna 210-225, dels varje måndag i ny månad. Se figur 15.

Som synes har varken VALUE RESTRICTION, VALUE RESTRICTION COMPONENT, VALUE EXPRESSION eller VALUE RANGE något eget namn. De har sin relevans endast så länge det finns relateringar med dess omvärld.



FIGUR 15

7. Sammanfattning

Det är inte möjligt, utan en djupare analys, att bedöma om det redovisade schemat i alla delar hänger ihop på ett konsekvent sätt. Exv har VALUE en attribute type **value**, vars värde ska överensstämma med motsvarande för den av de relaterade symbolalternativen, som via dess <symbol set> är kopplad till aktuell <info type> över relationship type HAS DEFINED. Upprätthålls detta beroende? Vi har heller inte kunnat få klarhet i hur fritt "expressions" kan uttryckas. Kan en sådan exv innehålla referenser till andra komponenter i schemat? På det hela taget måste annars det redovisade schemat för attribut och värdehantering anses vara ambitiöst, något som dess komplexitet ger belägg för.

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 nytta 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 bord 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.