

# Översättning av modelldata

från GDMO-modeller till division nättjänsters  
"vanliga" datamodeller

*Stig Berild*

**Spridningsförbehåll:**

Denna rapport får endast spridas och användas inom de organisationer som deltar som parter i TRIAD-projektet. ©TRIAD december 1993

*Rapporterna beställs från:  
SISU, Electrum 212, 164 40 Kista, Fax 08-752 68 00.  
Rapporterna är endast tillgängliga för Triad-parterna och är avgiftsfria.*

# Innehåll

<b>1. Förutsättning</b>	<b>3</b>
<b>2. Aktuella begreppsmodeller</b>	<b>4</b>
<b>3. Översättningsmodell</b>	<b>7</b>
<b>4. Översättningsförslag</b>	<b>9</b>
4.1 Managed object class	9
4.2 Attribute type	9
4.3 MOC attribute type	
4.4 Super/subclass	11
4.5 Identification	12
4.6 Action	13
4.7 Notification	20
<b>5. Översättningar</b>	<b>21</b>
5.1 Förutsättningar	21
5.2 Exempel	21

# **Översättning av modelldata från GDMO-modeller till division nättjänsters ”vanliga” datamodeller**

Observera, att den här rapporten har till syfte att initiera en debatt, inte att ge en färdig lösning. Problematiken kommer säkert att bli föremål för diskussion under ganska lång tid. Dels är inte de principiella tankegångarna färdigutvecklade, dels är modellerna i sig under vidareutveckling och förfining.



# 1. Förutsättning

I TRIAD-rapporten om GDMO-standarderna, del A, diskuterades olika principer och förutsättningar för datautbyte. Bland annat konstaterades att mottagaren måste förstå avsändarens verksamhetsmodell eller den verksamhetsmodell som upprättats som "medlare" mellan sändare och mottagare. För detta krävs att mottagaren har kunskap om den begreppsapparat (begreppsmodell) som använts vid formuleringen av verksamhetsmodellen. Inför den direkta överföringen måste mottagaren också vara klar över den syntax som används för att avgränsa och relatera överföringens olika beståndsdelar. Syntaxen formuleras sannolikt i begreppsmodellens terminologi.

Avsändaren antas vara ett open system eller någon form av abstrakt data-samordnare av data från open systems. Mottagare antas vara ett godtyckligt system för vilket GDMO-data är av värde. Av detta framgår att avsändaren har sin verksamhetsmodell uttryckt i termer av Management Information Model (se de s.k. GDMO-standarderna 10165-x). Mottagaren antas ha sin motsvarande verksamhetsmodell uttryckt i termer av den inom division nättjänster (N) använda begreppsmodellen.

Olika typer av protokoll kan i princip användas vid den konkreta överföringen av verklighetsdata. Här förutsätts enbart att mottagaren kan avskilja och förstå den inkommande dataströmmens olika komponenter.

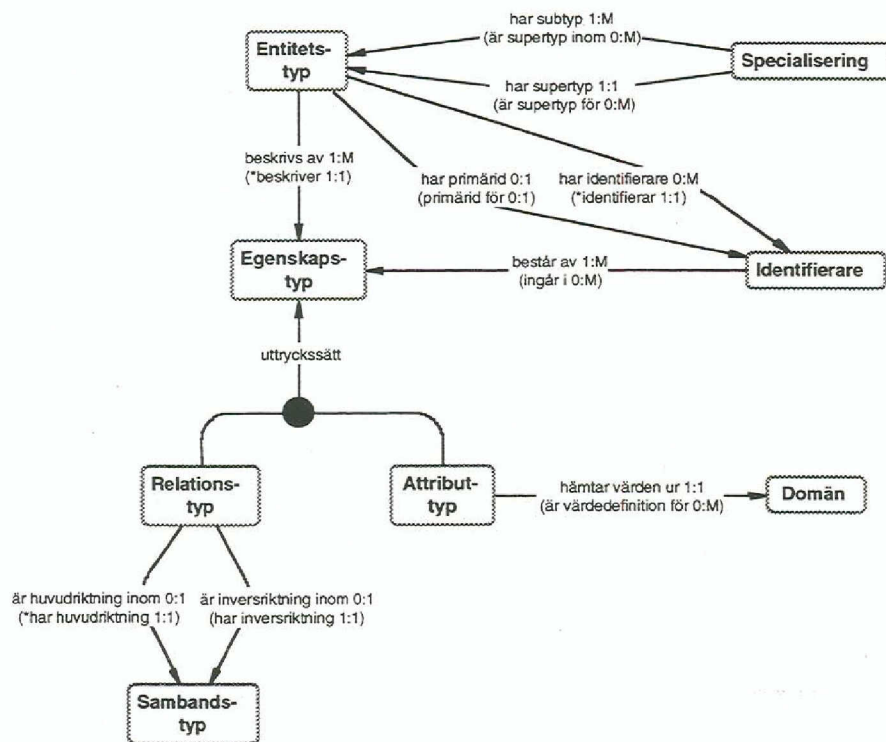
Som förutsättning gäller en idealiserad situation. Antingen antas N-modellen vara (halv)automatiskt genererad enligt generella regler för hur de båda modellernas begrepp hänger ihop eller gäller att GDMO-verksamhetsmodellören och N-verksamhetsmodellören båda uppfattar den modellerade verkligheten på samma sätt trots att de uttrycker den i sitt eget "språk". På så vis får vi möjlighet diskutera de båda modellernas begrepp utan att behöva ta hänsyn till verksamhetsspecifika förutsättningar.

I många reella sammanhang existerar troligen redan de båda verksamhetsmodellerna, dessutom etablerade utifrån specifika omständigheter, villkor, hänsyn, mm. Inga generella regler kan då tillämpas. Varje komponent i respektive verksamhetsmodell måste analyseras förutsättningslöst i och för relatering till komponenter i den andra modellen. Det går mycket väl att etablera en utökad begreppsmodell där både GDMO- och N-modellen, kompletterade med relateringsbegrepp, som kan formulera hur komponenter i en verksamhetsmodell uttryckt enligt GDMO-modellens begrepp relateras till komponenter i en verksamhetsmodell uttryckt enligt N-modellens begrepp, finns med. När en sådan relatering mellan två verksamhetsmodeller etablerats kan verklighetsdata om den specifika modellerade verksamheten utväxlas. Eftersom syftet med denna rapport i första hand är att titta på GDMO- och N-modellernas generella egenskaper, likheter och olikheter, ligger definitionen av en sådan utökad begreppsmodell utanför rapportens ram.

## 2. Aktuella begreppsmodeller

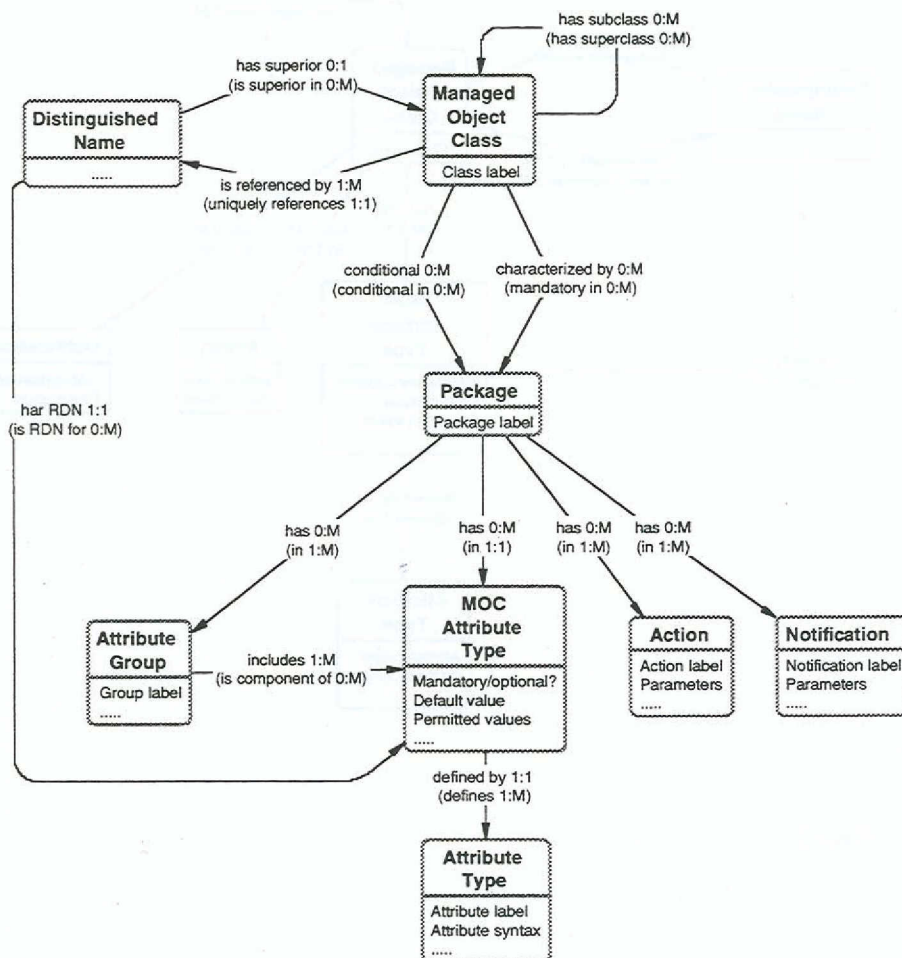
I fortsättningen används Telmod's begreppsapparat och grafiska notation för att uttrycka båda begreppsmodellerna. Telmod-begreppsmodellen är för övrigt mycket snarlik N's begreppsmodell.

N's begreppsmodell visas i figur 1. Endast de delar av relevans för den aktuella översättningssituationen har inkluderats.



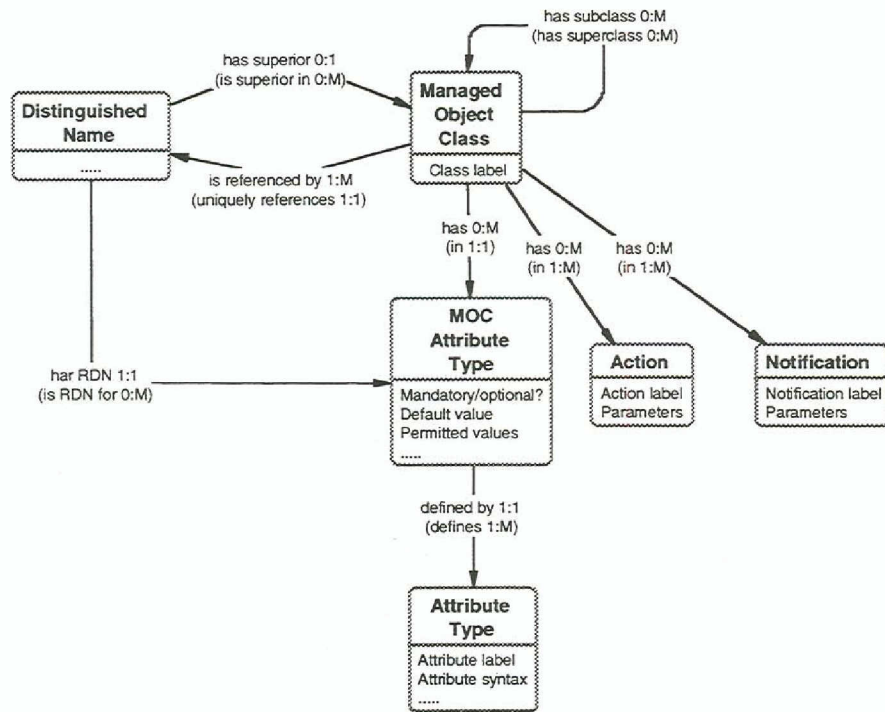
Figur 1

Management Information Model har utseende enligt figur 2 (se vidare TRIAD-rapport nr K25, del B). Observera att figuren inte ger en fullständig bild av modellen. Förhoppningsvis är de för en översättningssituation väsentliga delarna beskrivna, åtminstone tillräckligt mycket för att föra ett resonemang kring översättningsproblematiken.



Figur 2

Attribute group är en gruppering med syftet att underlätta formulering av operationer. Konstruktionen tillför ingen uttryckskraft av värde för att förklara betydelsen av överförda data. Package formulerar indirekt ett existensvillkor. De data som överförs har förhoppningsvis redan skapats hos avsändaren på ett mot Package-definitionerna konsekvent sätt. Tillämpar mottagaren Package-definitioner är det mottagarens ensak att hantera inkommande data i enlighet med dessa definitioner. Package tillför alltså heller ingen uttryckskraft av värde för att förklara betydelsen av överförda data. Management Information Model inskränks därför till vad som redovisas i figur 3. Situationen blir givetvis annorlunda om/när de olika begreppen i standarden 10165-7 tillfogas.



Figur 3

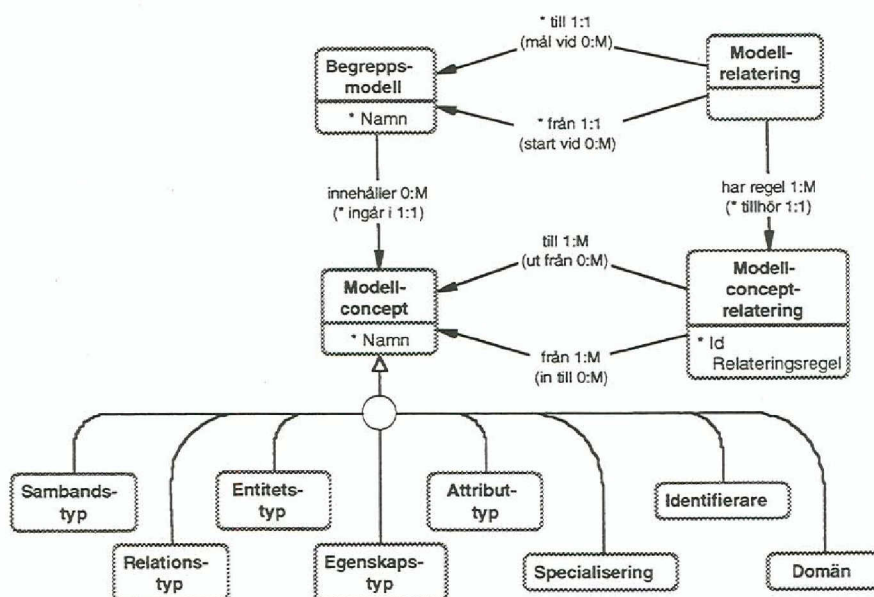


# 3. Översättningsmodell

Primärt gäller det att översätta från konstruktioner i figur 3 till konstruktioner i figur 1. Översättningen är inte nödvändigtvis reflexiv. Ta exv från flervärdig egenskapstyp i en N-modell till relation i en relationsmodell. Motsatta vägen kan antingen bli en egenskapstyp eller en entitetstyp beroende på översättningsintelligens och kunskap om relationens syfte. Dubbelriktad översättning kräver alltså två uppsättningar översättningsregler.

Vi har redan konstaterat att GDMO-modellen och N-modellen är uttryckta i Telmod-notation och att dess begreppsapparat överensstämmer med N-modellens. Telmod-modellen befinner sig på fundamental level eller IRD Definition Schema Level i enlighet med standarden ISO/IEC 10027. Dess syfte här är att kunna uttrycka begreppsmodeller. Detta syfte skulle ju kunna utvidgas till att inkludera definitioner av översättningsprinciper mellan modellbegrepp, vilket vi också väljer att göra.

Varje modellbegrepp kan bli föremål för översättning eller utgöra resultatet av en översättning. Alltså måste Telmod kompletteras med en gemensam generalisering eller superclass till vilken översättningsreglerna kan referera. Vi antar också att det generellt gäller att ett eller flera "in-begrepp" svarar mot ett eller flera "ut-begrepp". Tillåter vi sedan Telmod att uttrycka översättningar mellan många begreppsmodeller, skulle den utökade modellen kunna se ut enligt figur 4. Observera att de ursprungliga sambanden (se motsvarande dem i figur 1) har utelämnats av överskådlighetsskäl.



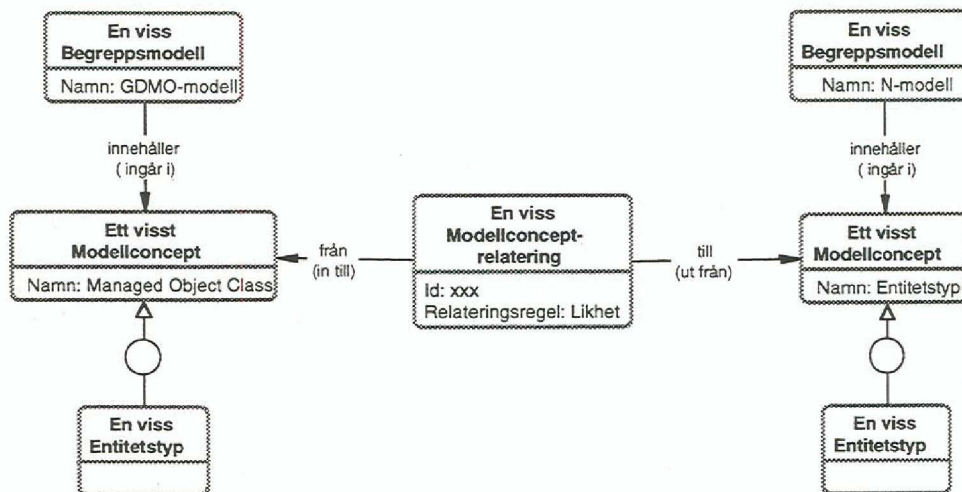
Figur 4

Relateringsformuleringen skulle kunnat formuleras med betydligt striktare formalism. Den skulle även kunna uttryckas i vanlig text under hänvisning till de båda modellernas begrepp. Förhoppningsvis är den valda nivån en rimlig kompromiss för denna första diskussionsomgång.

# 4. Översättningsförslag

## 4.1 Managed object class

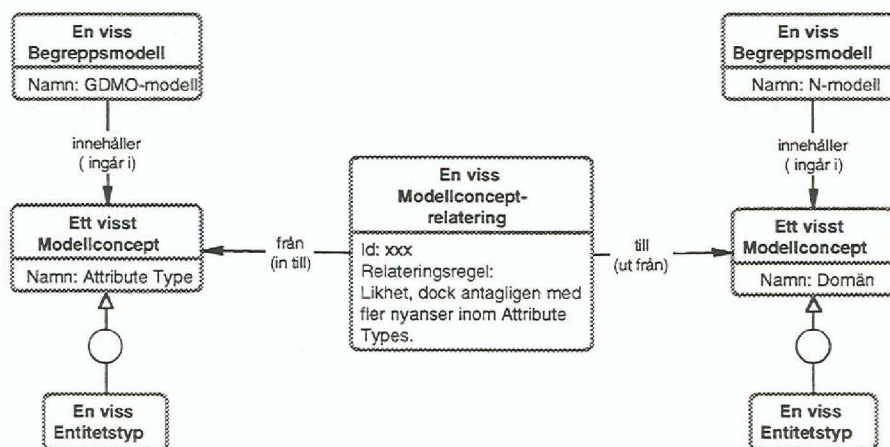
Begreppet Managed object class i GDMO-modellen svarar direkt mot begreppet Entitetstyp i N-modellen. Figur 5 visar den del av en förekomstnivå (uttryckt i enlighet med "schemat" i figur 4) som svarar mot de aktuella begreppens relatering.



Figur 5

## 4.2 Attribute type

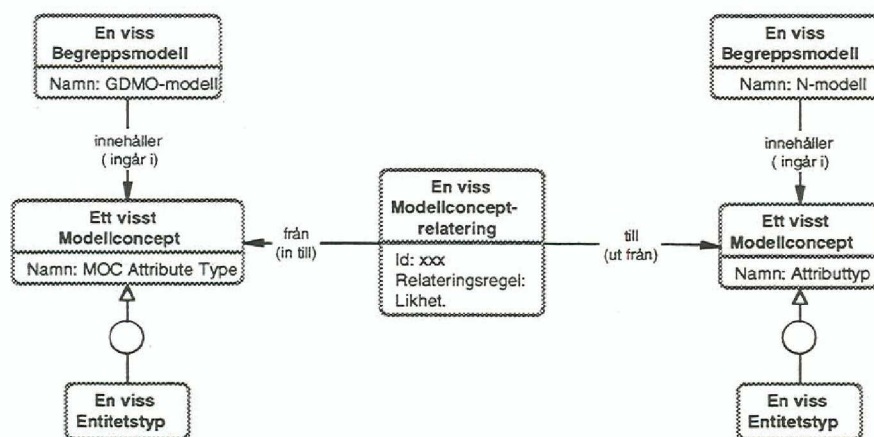
En Attribute type svarar närmast mot en Domän. En nyansskillnad kan förekomma såtillvida att Attribute type tycks vara formulerad betydligt mer med hänsyn till dess användning för att uttrycka någon egenskap hos en Managed object class, dvs det finns sannolikt betydligt fler Attribute types i GDMO-modellen än Domäner i N-modellen.



Figur 6

### 4.3 MOC attribute type

En MOC attribute type är en direkt motsvarighet till en Attributtyp.

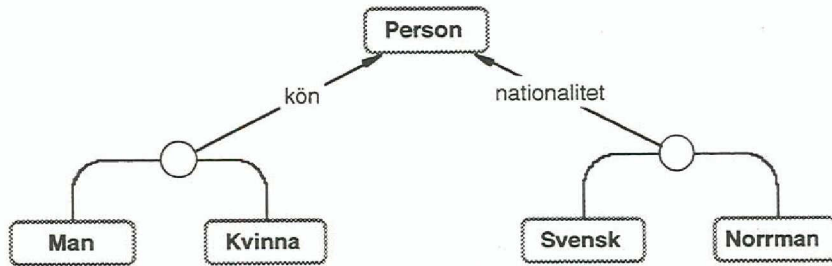


Figur 7



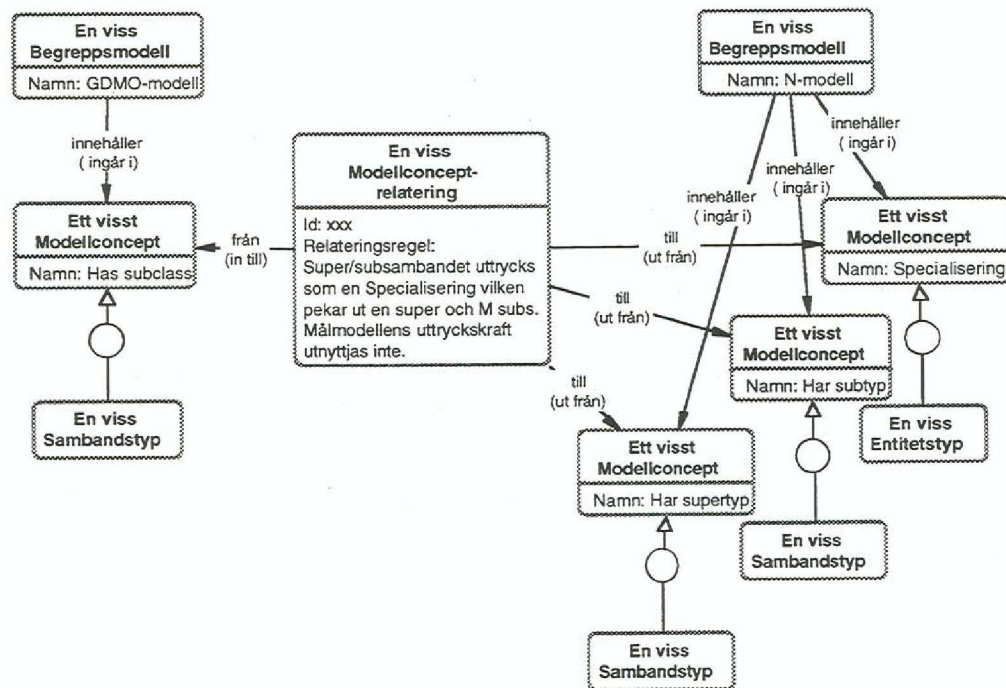
## 4.4 Super/subclass

I GDMO-modellen kan förekomma ett antal subclasses givet en viss Managed Object Class. Vad denna specialisering beskriver omnämns inte, än mindre om deras subclass-närvaro möjligtvis föranleds av olika specialiseringssyften. Att ange Man, Svensk, Norrman, Kvinna som subclasses till Person ger inte samma uttryckskraft som att dela in i två specialiseringar, var och en med sitt syfte. Specialisering 1 anlägger aspekten "kön" och för subclasses Man och Kvinna. Specialisering 2 anlägger aspekten "nationalitet" med Svensk och Norrman som subclasses. Se figur 8.



Figur 8

Vid översättningen till N-modell finns inga möjligheter att gissa aspekter eller syften varför vi får anta att endast ett syfte gäller och att det är självförklarande.



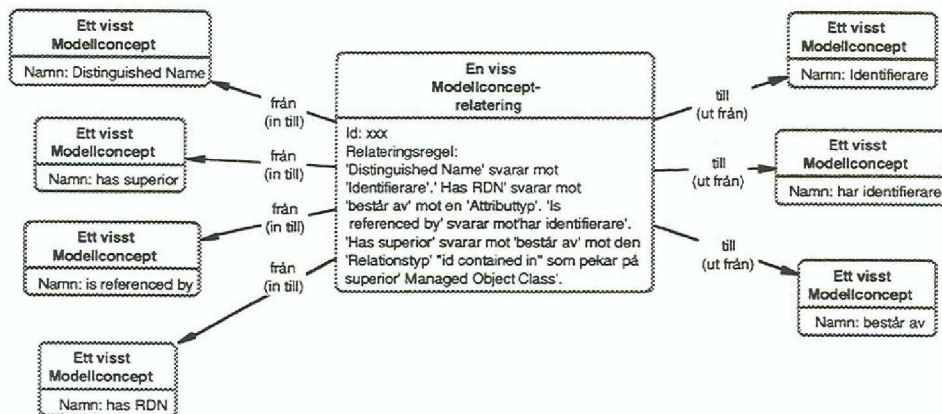
Figur 9

## 4.5 Identification

En identifierare (Distinguished Name) för en Managed Object Class är uppbyggd av ett lokalt direkt beskrivande MOC Attribute Type (Relative Distinguished Name) plus identifieringen av en närmast överordnad eller bestämmande Managed Object Class (vilken i sin tur kan ha en överordnad osv. i en rekursiv definition). Identifieringen för en Managed Object Class pekas alltså ut genom referens till ett RDN (som på översta nivå blir ett DN) plus referens till max en bestämmande Managed Object Class.

N-modellen ger betydligt större frihetsgrader i och med att valfri kombination av Relationstyper och Attributtyper (generaliserade till det gemensamma Egenskapstyp) kan pekas ut. Attributtyperna blir tillsammans motsvarande RDN. Varje utpekad Relationstyp pekar vidare på en Entitetstyp vars identifiering avses ingå som del i den totala identifieringen. Genom att ett flertal relationstyper kan utpekas, kan ett antal entitetstypers identifiering ingå i den totala identifieringen. Dessas identifierare kan i sin tur vara uppbyggda på motsvarande sätt, osv.

Både GDMO- och N-modellen tillåter specifikation av flera alternativa identifierare. För ett visst Distinguished Name pekas bara en attributtyp ut. Relationstyper existerar ännu inte i GDMOs grundmodell (se dock 10165-7). Även när dessa inkluderats är det oklart om dessas semantik på något sätt kommer att användas vid identifieringen. Vi etablerar därför en speciell sambands- typ för ändamålet med beteckningen "id-contains" från superklassen och med inversen "id-contained in". Figur 10 uttrycker det hela i en något förenklad version för överskådlighetens skull. N-modellens "har primärid" används ej.



Figur 10

## 4.6 Action

Tolkningen av Action behöver utredas. Här följer ett försök till resonemang kring olika tolkningar av Action och deras påverkan på översättningsprinciperna.

Action står antingen för en sammansatt management operation eller har att göra med beteendet hos en resource. I det första fallet underförstås att varje del-operation berör samma managed object. Denna operationsgruppering saknar relevans ur ett förklaringsperspektiv eftersom en sådan gruppering inte berör någon avbildning eller abstraktion av den aktuella verkligheten. Grupperingen formulerar endast ett sätt att handskas med data om en resource representerad av ett managed object.

Gäller det andra fallet kan minst tre perspektiv ansättas:

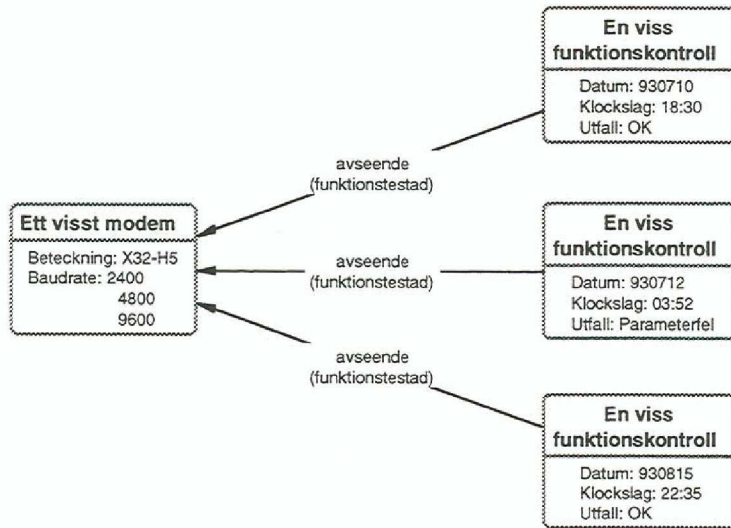
- a. Datautbytet kan innehålla uppgifter om unika actions som en resurs utfört.

I verklighetsmodellen uppstår resursen som ett managed object. En utförd action kan återfinnas som ett attribute till aktuellt managed object på samma sätt som en baudrate. Motsvarande attribute type kan vara flervärdig, dvs klara flera actions eller baudrates. Figur 11.

Ett visst modem	
Beteckning:	X32-H5
Baudrate:	2400 4800 9600
Funktionskontroll:	0710-18:30 0712-03:52 0815-22:35

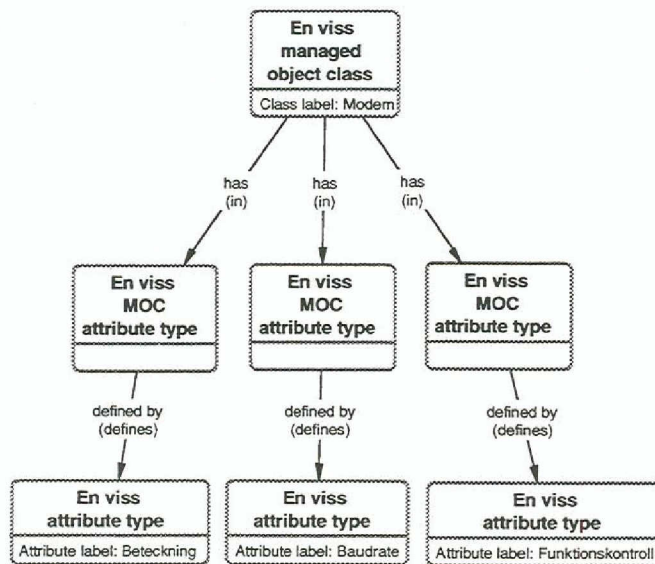
*Figur 11*

Behöver en action beskrivas med mer än ett värde, textsekvens e.dyl. upphöjs det till ett eget managed object med sina egna beskrivningsattribut samt relateras till resursens managed object. Se figur 12.



Figur 12

Verksamhetsmodellerna kan i första fallet formuleras med hjälp av befintlig GDMO-modells begrepp Managed object class, MOC attribute type och Attribute type.



Figur 13



En kompaktare grafisk form visas i figur 14 a och en ännu kompaktare i figur 14 b.

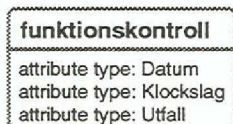
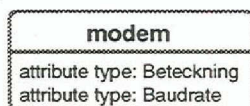


Figur 14 a

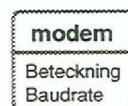


Figur 14 b

Med funktionskontroll upphöjd till en managed object class blir kompaktformen enligt figur 15. Eftersom relationship types mellan managed object types fortfarande är under arbete i 10165-7 (och i ofärdigt skick), etableras inga sådana. Ett alternativ vore att etablera någon form av standardiserad, fördefinierad relationship type för detta ändamål (se figurerna 18 och 19).

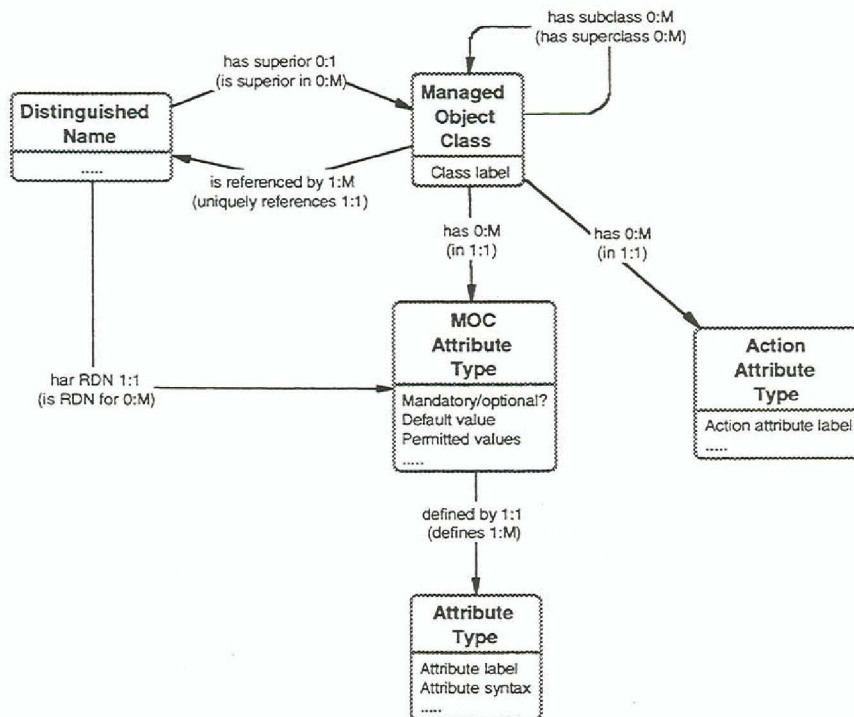


Figur 15 a



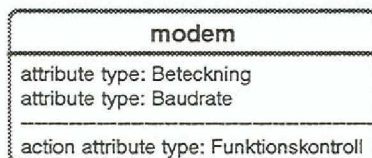
Figur 15 b

Önskar man precisera att det föreligger en distinktion mellan ett "vanligt" attribute type och ett "action" attribute type, måste begreppsmodellen kompletteras. Se figur 16.



Figur 16

Verksamhetsmodellen kan då presenteras enligt figur 17. Action attribute types visas under strecket.

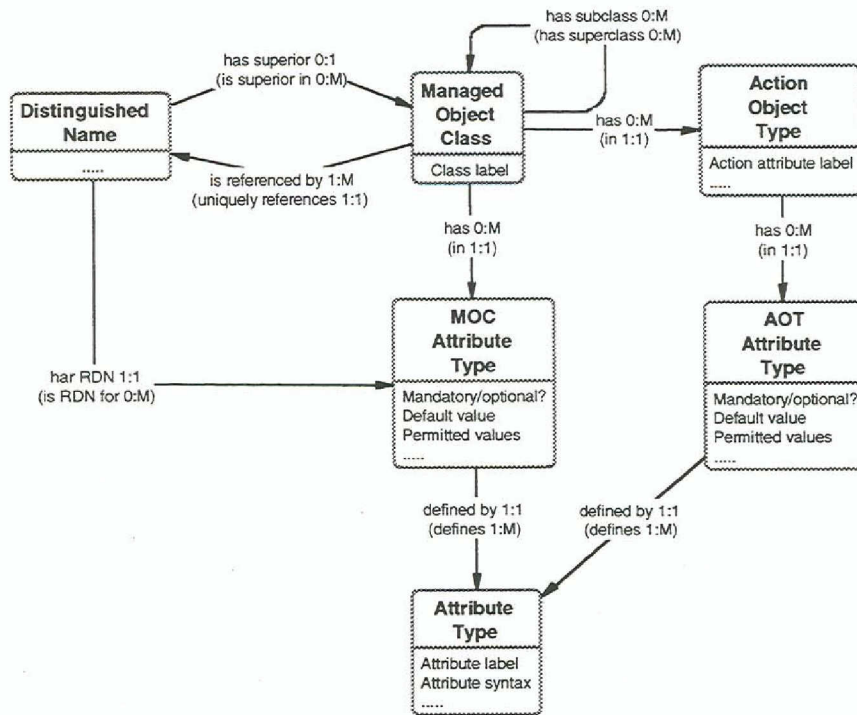


Figur 17 a



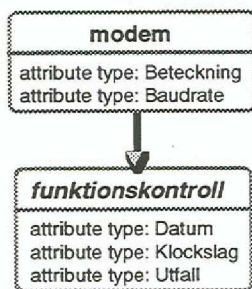
Figur 17 b

Önskar man precisera att det föreligger en distinktion mellan en "vanlig" managed object class och en "action" object class blir begreppsmodellen istället enligt figur 18.

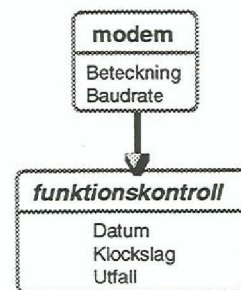


Figur 18

Motsvarande verksamhetsmodell, med action object types kursiverade, visas i figur 19.



Figur 19 a

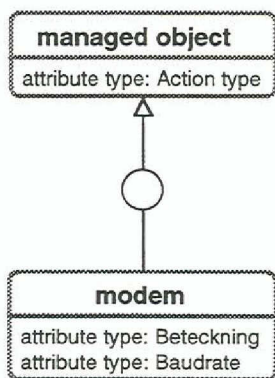


Figur 19 b

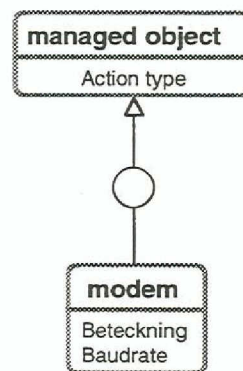
I GDMO-modellen finns begreppet action. Under den givna förutsättningen verkar ovanstående resonemang indikera att figur 16 är det alternativ som ligger närmast den befintliga GDMO-modellen, dvs att jämföra GDMO-modellens Action med figurens Action attribute type.

- b. Datautbytet kan innehålla uppgifter om vilka typer av actions som en viss resurs är kapabel att utföra eller ombeds utföra.

Antingen är detta en karakteristik som tillfogas resursens managed object som ett flervärdigt attribute eller beskrivs varje action type som en självständig företeelse i verklighetsmodellen. Eftersom varje managed object i princip kan ha actions etablerar man rimligtvis en superclass på verksamhetsmodellnivån dit antingen attribute type Action type placeras eller dit en specifik managed object class Action type relateras. Se exempel på attribute type alternativet i figur 20.

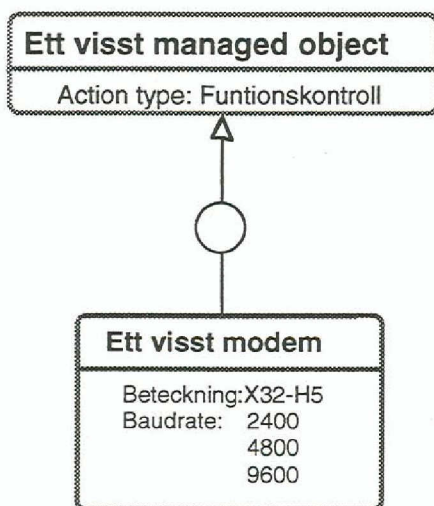


Figur 20 a



Figur 20 b

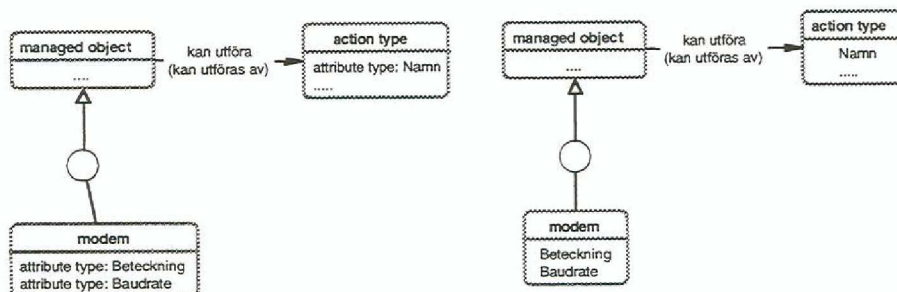
Exempel på verklighetsmodellnivå visas i figur 21.



Figur 21



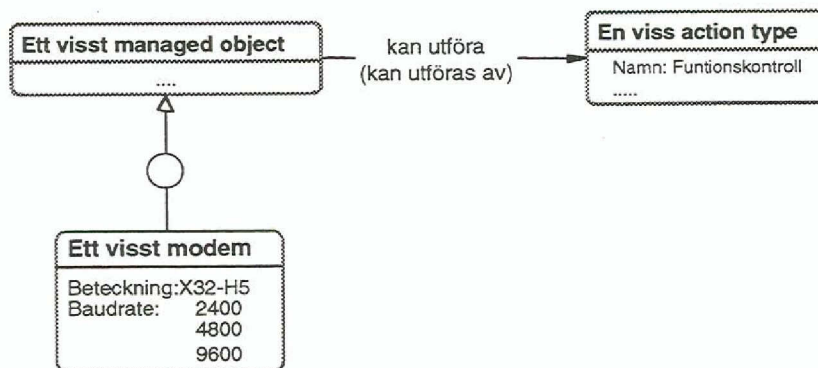
Med action type som object class blir verksamhetsmodellen enligt figur 22.



Figur 22 a

Figur 22 b

Exempel från verklighetsmodellnivån återfinns i figur 23.



Figur 23

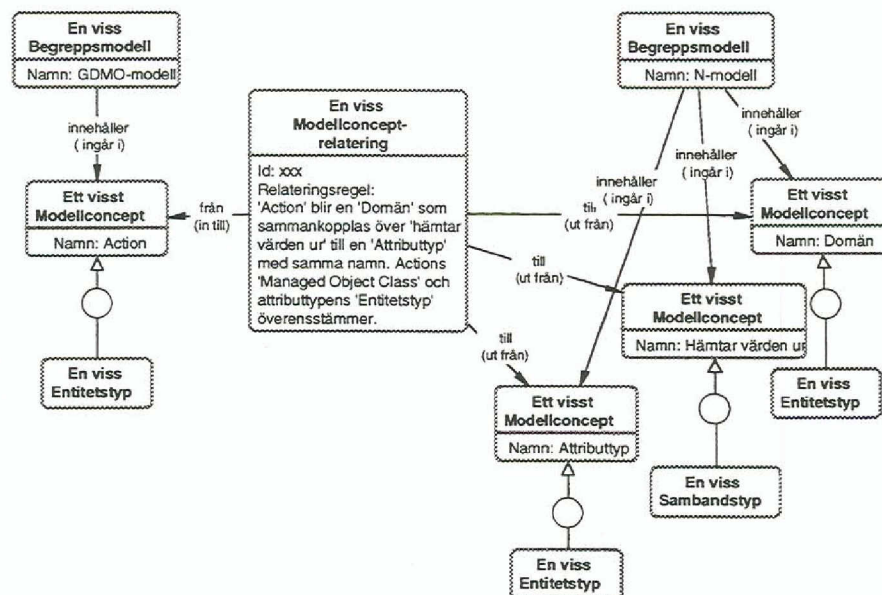
Ingen förändring från den ursprungliga GDMO-modellen behövs, dvs action "döljs" bland övriga modelleringsbegrepp.

- c. Det är bara intressant att formulera vilka typer av actions som en viss resurstyp kan utföra.

Denna förutsättning kan vara intressant i samband med kontroll och styrning av vilka actions som är tillåtna mot resurser tillhöriga viss resurstyp. Dock uppträder inte uppgifterna på verklighetsmodellnivå. Mao ingen påverkan på GDMO-modellen.

För översättningen utgår vi från att alternativ a gäller som förutsättning. Situationen är dock inte entydig eftersom operationstypen Action knappast kan avse inträffade actions, endast önskan om att en sådan utförs. Inte heller tycks det finnas utrymme hos övriga operationstyper att begära uppgifter om inträffade actions, dvs "historiska" uppgifter.

Någon motsvarighet till Action attribute type finns inte i N-modellen varför vi väljer att gruppera dessa under en "vanlig" attributtyp. Benämningen på en viss Action attribute type framgår ju likafullt men under en inte lika distinkt klassificering. Eftersom en och samma Action kan beskriva flera Managed Object Classes måste också en Action etableras som Domän. I brist på bättre alternativ väljs Action-namnet för både Domän och aktuell relaterad Attributtyp.



Figur 24

## 4.7 Notification

Notification är ännu något oklarare än Action. Det framgår i olika tolkningar av standarden ett en Notification kan avse fler än en resurs. Vi bortser här från detta och antar för enkelhets skull att Notification kan hanteras på samma sätt som Action. Se figur 24, ovan men ersätt modellconceptet med "Namn: Notification".

# 5. Översättningar

## 5.1 Förutsättningar

Med översättningsreglerna givna mellan de båda begreppsmodellerna kan vi i princip ta en verksamhetsmodell uttryckt enligt GDMO-modellen och överföra den till en verksamhetsmodell uttryckt enligt N-modellen. Hade översättningsreglerna i den utökade Telmod-modellen varit formellt definierade och kompletta, skulle arbetet kunna utföras med automatik. I dagsläget måste man för varje komponent eller komponentgrupp gå till respektive relateringsregel för att se vilka handgrepp som erfordras.

Vid den reella överföringen av verklighetsdata skulle samma formella definition, om den finns, kunna tolkas och för översättningen. Är den uttryckt i mer eller mindre löpande text under relateringsregel måste varje förutsättning implementeras.

Sker översättningen till en redan, sedan gammalt befintlig, verksamhetsmodell enligt N-modell måste varje komponent i källmodellen analyseras förutsättningslöst.

## 5.2 Exempel

Ta följande exempel. För enkelhets skull antar vi att principerna för domänvärdens inkodning överensstämmer mellan parterna. Vi bortser också från att formulera vad som konstituerar korrekt fullständig identifiering.

Vi önskar överföra "En anställd med namn Svensson och telefonnummer 999999 och 888888 har som reparatör beläggningsgraden 85%. Som reparatör har han utfört reparation r2 och inspektion i3 och i4". Han fungerar även som leveranskontrollant.

Syntaxen enligt GDMO-förutsättningar (figur 3) skulle kunna vara

```
MOC [<class label>#<id>:[<attribute label>: <värde>[,  
<värde>]...;]...;].../[<action label>: <värde>[, <värde>]...;]...;]...;
```

```
SUB [<class label>#<id>:<class label>#<id> [,<class  
label>#<id>]...;]...;
```

Som synes uttrycks syntaxen i termer av GDMO-modellens begrepp. <Id> genereras och har relevans endast i och för överföringen, så att korrekta referenser kan formuleras. Överföringen blir

```
MOCanställd#001:namn:Svensson;telefonnummer:  
999999,888888;;reparatör#111: beläggningsgrad:85%;/  
reparation:r2;inspektion:i3,i4;;;
```

```
SUBanställd#001:reparatör#111,leveranskontrollant#222;;
```

Detta måste översättas till en syntax enligt N-modellens begrepp och villkor (figur 1), för att mottagaren ska kunna förstå innehållet. Exv

```
ENT [<entnamn>#<id>:[<attnamn>:<värde>[,<värde>]...;]...;];
```

```
SUB [<entnamn>#<id>:[<aspekt>,<entnamn>#<id>];]...;
```

Notera att N-modellen inte skiljer på Attribute types och Action attribute types samt att en specialisering är aspektindelad med villkoret att subclasserna under viss aspekt är varandra ömsesidigt uteslutande.

Med automatgenererad målverksamhetsmodell uppstår inga namnkonflikter. Eventuella nygenererade namn baseras entydigt på vad som framgår av regeltexten. Vid separat implementering utgår vi ifrån att den så korrekt som möjligt avbildar relateringsvillkoren vid översättningen.

Den översända sekvensen, omformulerad enligt ovan beskrivna regler (avsnitt 4), skulle kunna bli:

```
ENTanställd#001:namn:Svensson;telefonnummer:  
999999,888888;;reparatör#111:beläggningsgrad:85%;  
reparation:r2;inspektion:i3,i4;;;
```

```
SUBanställd#001:profession,reparatör#111;  
förtroendeuppdrag,leveranskontrollant#222;;
```







# PUBLIKATIONER AV TRIADGRUPPEN

## Verksamhetskrav på informationsadministration

- V 1: IA och verksamhetens krav – erfarenheter från offentlig förvaltning
- V 2: Fallstudie av IA-projektet vid Televerket
- V 3: IA-erfarenheter från företag och myndigheter
- V 4: Den gemensamma informationsmarknaden – en referensram för handlingsfrihet och konkurrenskraft
- V 5: ...fråga är guld. Lokal affärsstyrning utifrån den egna verksamhetens data

## Modellering

- N 1: Modelleringsansatser för begrepps- och datamodellering – Beskrivning och försök till jämförelse
- N 2: Generering av konceptuella modeller från policydokument
- N 3: Espritprojektet Tempora
- N 4: Prövning av regelbaserad metodik inom Posten
- N 5: En kokbok i remodellering – utkast
- N 6: Datorstöd för modellintegration
- N 7: Modellbaserad kunskapsinsamling
- N 8: Modellkvalitet
- N 9: Samband mellan dokument och modeller
- N 10: Modelleringshandboken
  - 1 – Översikt
  - 2 – Modelleringsledarens bashandledning
  - 3 – Modellering i grupp
  - 4 – Kommunikation
  - 5 – Arbetsgångar
  - 6 – Modelleringsväskan
  - 7 – Objektorienterad verksamhetsanalys
  - 8 – Basmodeller
  - 9 – Regelmodellering i praktiken
  - 10 – Business Process Reengineering
  - 11 – Namnsättning
  - 12 – Tolkning av grafiska modeller
- N 11: Ett+Ett=Ett – Två praktikers erfarenheter av modellintegrering

## Kunskapsförmedling

- H 1: Handledarutbildning för modelleringsledare, avancerad
- H 2: Slutrapport HUMLA prototyp
- H 3: Utbildning i Informationsadministration
- H 4: Spridning av Hybris – en fallstudie vid Telia

## Uttagssystem

- U 1: Hybris i Unix-miljö
- U 2: DEBRIS
- U 3: Hybris DOS/PimWin på Posten
- U 4: Program för sökning i databaser – en marknadsöversikt
- U 5: Att nå och förstå data – möjligheter och begränsningar

## Katalogprinciper

- K 1: IRDS
- K 2: IRDS Modeller och modellnivåer
- K 3: Koppling begreppsmodell – relationsmodell
- K 4: IBM:s Repository Manager – en introduktion
- K 5: IBM:s Repository Manager: Datamodelleringsbegreppen
- K 6: IBM:s Repository Manager: Begreppsmodellering i Information Model
- K 7: IBM Repository Manager: Attribut- och värdemodellering i Enterprise Submodel
- K 8: Navigering i Repository
- K 9: TRIAD Newsletter – IRDS inom ISO. Dagsläget
- K 10: TRIAD Newsletter – ISO/IRDS. Händelseutvecklingen 91/92
- K 11: Samverkan mellan resurskataloger – visioner eller behov
- K 12: AD/Cycle I Information Model – Processer och informationsflöden mellan processer
- K 13: AD/Cycle I Information Model – Info Flows inom Processmodellen
- K 14: AD/Cycle I Information Model – Relationsdatabasmodellering
- K 15: AD/Cycle I Information Model – Härlednings-specifikationer i begreppsmodellen
- K 16: IA-prototyp
- K 17: Repository AD/Cycle – International Users Group
- K 18: RAD-konferensen i Chicago, 1992
- K 19: Vad händer inom ANSI-IRDS?
- K 20: Information Warehouse – vad är det?
- K 21: CDF – en översikt
- K 22: PCTE – en översikt
- K 23: XLII – en öppen och flexibel utvecklingsmiljö
- K 24: Hybris IA/DÄ – En IA-prototyp vid Telia
- K 25: Introduktion till GDMO-standarderna
- K 26: OpenODB – en introduktion
- K 27: ANSI/X3H7 "Object Information Management"
- K 28: Object Management Group

## KORT OM TRIAD

*Triad är namnet på ett treårigt samarbetsprojekt kring informationsadministration och dataadministration, IA/DA, som Telia, Posten, Ericsson, Statskontoret och SISU bedriver. Syftet är att utveckla parternas synsätt, metoder och hjälpmedel inom detta område.*

*Arbetet inom Triad är uppdelat i delprojekt som är sammanförda i tre block.*

*Beställarblocket vänder sig dels till dem som är verksamhetsansvariga och måste ta ställning till IA-/DA-satsningar, dels till dem som har ansvaret för IA/DA inom en organisation. Delprojekten inom detta block arbetar med att formulera verksamhetens krav på IA/DA samt studerar och beskriver roller, organisation och arbetsformer för IA-/DA-arbete.*

*Utförarblocket vänder sig till dem som arbetar med IA/DA.*

*Delprojekten arbetar med modellering, data- och resurskataloger samt uttagssystem.*

*Kunskapsförmedling är det block som ser till att resultaten kommer Triad-parterna till godo. Detta sker bland annat genom kurser, seminarier samt genom att rapporter som denna ges ut.*