

# **CDIF**

## **– en översikt**

*Stig Berild*

**SISU & SVERIGES TEKNISKA ATTACHÉER**

©TRIAD Augusti 1993



# Innehåll

- 1. Inledning 3**
  - 1.1 Problemområde 3
  - 1.2 Förutsättningar för datautbyte 3
  - 1.3 CDIF:s syfte och omfattning 4
  - 1.4 Problem 6
  
- 2. CDIF-arbetets organisation 8**
  
- 3. Existerande CDIF-dokument 9**
  
- 4. Meta-meta-modellen 10**
  
- 5. Existerande Metamodeller 13**
  - 5.1 CORE 13
  - 5.2 ERA 16
  - 5.3 DFD 17
  - 5.4 DATA INVENTORY 18
  - 5.5 Presentation Model 20
  - 5.6 Utbyggbarhet 21
  
- 6. Överföringsformat 22**
  - 6.1 Formatspecifikation 22
  
- 7. Pågående och planerade aktiviteter 24**
  - 7.1 Meta-modeller 24
  - 7.2 Syntax 25
  - 7.3 "Varudeklaration" 25
  
- 8. Några avslutande kommentarer 26**





# 1. Inledning

## 1.1 Problemområde

Triad-rapport K 22 om PCTE innehåller ett inledande resonemang om olika typer och grader av integrering. Den inledningen skulle i princip passa lika bra i en rapport om CDIF. Läs den därför gärna först. Denna rapport går direkt in på en översiktlig beskrivning av CDIF, dess bakgrund, innehåll och framtidsperspektiv. Den läsare som endast önskar en allmän orientering om CDIF kan hoppa över avsnitten 5.1-5.6 samt kapitel 7.

## 1.2 Förutsättningar för datautbyte

CDIF står för Case Data Interchange Format och är dels beteckningen på ett antal standarddokument, dels en kommitté inom vilket arbetet bedrivs. Mer om detta nedan. Syftet är att underlätta överföring mellan Case-verktyg av de typer av data som dessa normalt hanterar. Det är med andra ord fråga om datautbyte (data interchange), inte datadelning (data sharing). Se figur 1.



Figur 1

Data skickas mellan verktyg i enlighet med en överenskommen syntax och semantik. Samordningen innebär endast att mottagaren kan förstå vad avsändaren har skickat. Samordningen gäller endast data (inte process eller användargränssnitt) och endast momentant i samband med en överföring.

Syntaxen beskriver den form eller struktur, t ex ordningsföljd och avskiljningsymboler, som används för att ordna de data som överförs, så att de entydigt kan "packeras" av avsändaren och lika entydigt kan "packas upp" av mottagaren. Överföringstekniken som används för det fysiska överföringsmediet i kombination med en syntax brukar kallas protokoll.

Semantiken representerar innebörden av det som skickas. För att sändare och mottagare ska kunna förstå samma sak av det som skickats, måste de ha samma uppfattning om innebörden av de symboler som sänds. Ett sätt att åstadkomma detta är att definiera använda symboler och begrepp i en begreppsmodell. Begreppsmodellen är en abstraktion av den verklighet (universe of discourse) man avser att utbyta data om. Förutom begreppen kan modellen innehålla begreppens samband och regler för deras användning m m. T ex kan modellen dels innehålla en definition av vad begreppen *Person* och *Härfärg* betyder, dels ett samband som talar om att *Härfärg* är något som beskriver en *Person*.

Att skicka "Pelle, röd" från A till B säger antagligen inte B så mycket om inte B känner mycket väl till vad A brukar prata om. I bästa fall kan använd syntax ge information om att först kommer ett namn på någonting, därefter, avskilt av ett kommatecken, följer en symbol som representerar någon egenskap för detta någonting. B kan alltså, med hjälp av syntaxen, dela upp det inkomna meddelandet i två beståndsdelar, var och en med sin generella innebörd, enligt

<namn på något> , <namn på egenskap>

Om Pelle är namnet på solen och röd är valören på solens påverkan på horisontens molnskärm vid en solnedgång, eller något helt annat, framgår däremot inte.

Antag nu att A och B lyckats komma överens om en viss syn på "världen" i och för överföring av data, d v s en gemensam begreppsmodell för överföring (som bl a inkluderar *Person* och *Härfärg* enligt ovan).

Att skicka och tolka "*Person: Pelle, Härfärg: röd*" är nu inga problem under förutsättning att de olika beståndsdelarna kan benämnas upp. Förmodligen innehåller deras modell många fler begrepp och samband. För enkelhets skull har A och B kommit överens om att använda en enda generell syntax för många behov enligt:

<objekttyp> : <namn> , <attributtyp> : <värde>

Med detta givet kan A packa in och B packa upp respektive beståndsdel. Delarna har sina givna roller, varav vissa har direkt koppling till den överenskomna begreppsmodellen och därmed till den semantiska aspekten. En förutsättning för ett meningsfullt utbyte är med andra ord en gemensamt accepterad och förstådd begreppsmodell. Observera att A och B var för sig kan arbeta efter helt andra modeller, ha gjort helt andra abstraktioner av den aktuella verkligheten. Huvudsaken är att de båda förstår innehållet i den gemensamma modellen, är villiga att uttrycka sig i termer av denna modell i samband med överföring och, inte minst, har förmåga att överföra data från den interna modellen till den gemensamma i samband med överföringen. Mer om det nedan.

### 1.3 CDIF:s syfte och omfattning

I CDIF kallas en abstraktion eller begreppsmodell enligt ovan för **meta-modell**. Är bara A och B inblandade, kan de förmodligen sätta sig ner vid ett bord och muntligt komma överens om vilka begrepp som ska användas och vad de betyder. Kommer C in i bilden, blir det lite jobbigare men är fortfarande hanterbart. D representerar en utländsk produkt och behöver hjälp av en tolk. E och F kommer med många nya begrepp. G använder en ordbehandlare och ett speciellt tabellformat för att dokumentera sin meta-modell. C och F har börjat använda grafiska symboler, dock av olika typ, för att åskådliggöra sina respektive meta-modeller, o s v. Det är inte lika enkelt att komma överens längre. Varför då alla dessa verktyg, räcker det inte med ett enda? Dels kan de täcka in olika faser av ett systems livscykel, dels kan de ha olika funktioner, användargränssnitt m m som passar olika användare olika bra, dels är det realistiskt att anta att större organisationer helt enkelt inte vill eller kan ha total kontroll och samordning av verktygsinköp.



Sammanjämkningen till en gemensamt accepterad meta-modell underlättas betydligt om förhandlingarna såväl som slutresultatet uttrycks i ett gemensamt, för alla förståeligt språk. Vi har redan använt sådana språkelement, exempelvis *objekttyp* och *attributtyp*, för att förklara syntaxen ovan. Meta-modeller formuleras och åskådliggörs med fördel grafiskt. Språkets begrepp, samband, egenskaper, regler m m inklusive en specifik grafisk notation kallas i CDIF för **meta-meta-modell**. CDIF:s meta-meta-modell är en variant av en Entity-Relationship-modell. Den grafiska notationen är okomplicerad.

Alltså, CDIF innehåller en definition av en meta-meta-modell med vars hjälp ett antal meta-modeller kan formuleras. En meta-modell är en abstraktion av, eller uttrycker semantiken hos, de data som är avsedda att överföras enligt överenskommen syntax, i allmänhet Case-data. En meta-meta-modell uttrycker en abstraktion av meta-modeller.

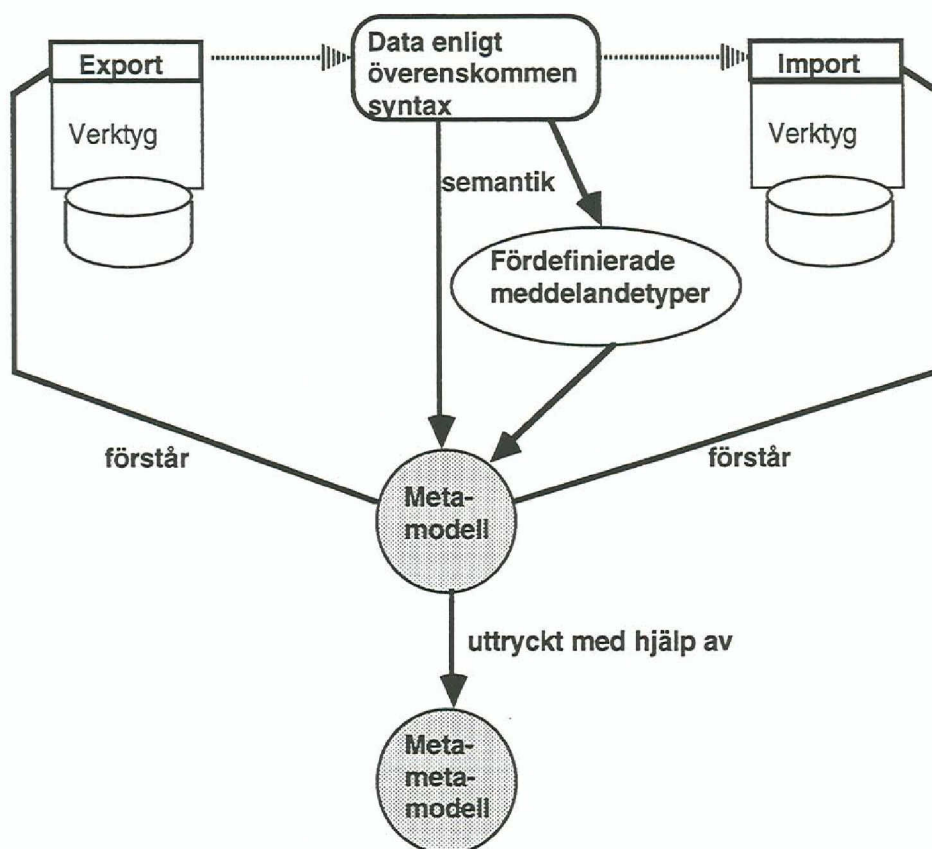
I en överföring, vilken som helst, kan semantiken antingen vara inkluderad, som i det enkla exemplet ovan, eller vara fördefinierad i form av en överenskommen meddelandetyper. CDIF-överföringar innehåller alltid semantiken. Jämför med EDIFACT inom vilket mycket omfattande meddelandetyper finns definierade.

CDIF omfattar också ett antal fördefinierade meta-modeller. Flera är under utveckling. De som så önskar kan även formulera egna meta-modeller i enlighet med den givna meta-meta-modellen. Samtliga med kunskap om CDIF:s meta-meta-modell har också möjlighet att tolka formulerade meta-modeller.

CDIF innehåller därutöver en alldeles specifik meta-modell, nämligen den som definierar olika typer av grafik och grafiska symboler. Genom denna fördefinierade meta-modell kan olika grafiska symboler, deras placering i ett koordinatsystem samt deras relatering till vad de symboliserar i såväl meta-modellen som bland reella data eller förekomster, föras över till mottagaren.

Förekomstnivån, d v s den nivå där de data som överföringen egentligen motiveras av finns, kallas i CDIF för **modell**.

Observera att meta-modeller endast utgör en slags neutral översättningsplattform. Avsändaren har data lagrade enligt det egna verktygets interna format och meta-modell, uttryckt i enlighet med verktygets egen interna meta-meta-modell. Data kanske ligger lagrade enligt någon variant av relationsmodellen (meta-meta-modellen) abstraherade i en meta-modell, vars begrepp och betydelse endast delvis överensstämmer med den neutrala överföringsmodellen. Samma sak gäller mottagaren, vars data kan ligga strukturerade enligt något egenutvecklat filformat och med egna interna begrepp. En överföring innebär att avsändaren måste transformera sin lokala modell till gemensam CDIF-meta-modell. Mottagaren måste göra motsvarande i omvänd ordning. Lyckas avsändaren transformera till en CDIF-meta-modell är det klart att överföra data, inte bara till en mottagare utan till alla som klarar att transformera från en given CDIF-meta-modell till egen intern form.



Figur 2

## 1.4 Problem

Att kunna tolka en meta-modell innebär inte automatiskt att den är användbar. Ibland är en transformation från en intern modell till en gemensam CDIF-baserad meta-modell enkel och klar, exempelvis om de interna begreppen och CDIF-begreppen överensstämmer till namn och innebörd. En transformation är även enkel om begreppens innebörd stämmer överens men inte namnen.

Ibland kan begreppens isolerade innebörd överensstämma, men "uttryckskraften" i den interna modellen och CDIF:s meta-meta-modell skilja sig åt. Hur transformerar man exempelvis ett specialiseringssamband enligt CDIF:s meta-meta-modell till en relationsmodell, som inte explicit kan formulera denna typ av samband?

Ibland finns inte entydig begreppsöverensstämmelse. I dessa fall får man antingen försöka göra så gott det går med de risker det kan föra med sig, eller helt enkelt avstå.

Vem ansvarar för innehållet? Är det den som levererar eller den som valt att ta emot leveransen? I en CDIF-baserad överföring borde underförstått gälla att avsändare och mottagare har att ta ansvar för sin respektive sida av en eventuell transformation. Dock säger detta ingenting om överföringens fullständighet. Ta t ex begreppet *Person* som

i CDIF-modellen står för "nu eller tidigare anställd", medan det i modellen hos avsändare A står för "nu anställd". När detta begrepp används kommer mottagare B förmodligen att tolka överförda data som samtliga "nu eller tidigare anställda" vilket kan ge oönskade konsekvenser. Varken A eller B har gjort något fel. Boven i dramat är en något oprecis CDIF-modell.

En liknande situation uppstår om B begär och förväntar sig alla personer med röd hårfärg och det endast för en del av A:s personer finns denna uppgift registrerad. Svaret blir ofullständigt.

Det går säkert att finna många fler liknande, oklara situationer.

Ett intressant test på om det är möjligt att genomföra en säker kommunikation mellan två eller flera verktyg kan vara att viss data kan transporteras fram och tillbaka igen utan att förvanskas längs vägen.



## 2. CDIF-arbetets organisation

Arbetet med CDIF utförs av en Technical Committee inom Electronic Industries Association (EIA). De standarder som tas fram är:

*designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvements of products, and assisting the purchaser in selecting and obtaining with minimum delay the proper product for his particular need.*

Man tar fram två olika typer av standarder, dels den ambitiösare *Recommended Standard* för vilken en formellt definierad framtagningsprocess tillämpas, dels en *Interim Standard* som är ett dokument som bedöms ha allmänt intresse och med en livslängd av tre år. Tanken är att bärkraftiga Interim Standarder under denna period kan hinna vidareutvecklas inom EIA eller inom annat standardiseringsorgan. CDIF-dokumenten EIA/IS-81, -82, -83 (se nedan) är alla, i sina existerande versioner, exempel på den senare kategorin.

EIA är ett ANSI-ackrediterat standardiseringsorgan vilket tillåter EIA att sända sina alster till ISO.

CDIF Technical Committee har för närvarande ca 40 medlemmar. Bland dem återfinns såväl Case-leverantörer (exempelvis Intersolv, LBMS, CADRE, Sybase och Oracle) som Case-användare (exempelvis Boeing och DuPont). I första hand antas USA-baserade företag som medlemmar men även andra accepteras i praktiken. Som i de flesta standardiseringssammanhang är man tacksam för de insatser som erbjuds, oavsett varifrån de kommer. Medlemskap kostar mellan 100-500 dollar beroende bl a på rösträtt.

# 3. Existerande CDIF-dokument

CDIF har hittills presenterat tre dokument:

## **EIA/IS-81: CDIF – Framework for Modeling and Extensibility**

Innehåll:

- Översiktlig beskrivning av meta-meta-modellen
- Beskrivning av använd grafisk notation
- Specifikation av tillåtna datatyper
- Något om en avsändares och mottagares roller
- Exakt definition av meta-meta-modellens komponenter

## **EIA/IS-82: CDIF – Transfer Format Definition**

Innehåller tre separata delar:

- Part 1: General Rules for CDIF Syntaxes and Encoding
- Part 2: CDIF Transfer Format Syntax – SYNTAX.1
- Part 3: CDIF Transfer Format Encoding – ENCODING.1

## **EIA/IS-83: CDIF – Standardised CASE Interchange Meta-modell**

Innehåll:

Part 1: Semantiska modeller

Definition av fyra ämnesområden (subject areas):

- en generell del (CORE)
- en ER-modell (ERA)
- en flödesmodell (DFD)
- en utbruten del av ER-modellen som beskriver olika aspekter på attribut, värden m m (DATA INVENTORY)

Part 2: En presentationsmodell

Definition av:

- generella grafiska objekt, exempelvis ikoner
- den grafiska representationen av visst semantiskt objekt

De kommande avsnitten behandlar översiktligt meta-meta-modellen (EIA/IS-81), de semantiska modellerna (EIA/IS-83, part 1) och presentationsmodellen (EIA/IS-83, part 2) samt principer för överföringssyntax (EIA/IS-82, part 1). För detaljuppgifter och information om övriga delar hänvisas till CDIF-dokumenterna.

## 4. Meta-meta-modellen

Denna modell är en variant av Entity-Relationship-modellen (se figur 3). Typiska egenskaper är att:

- modellen enbart har binära samband (relationships).
- relationship types kan ha attribute types.
- specialiseringshierarkier kan ange sk "multiple inheritance".
- det inte finns några namngivna inverssamband, dock antalsuppgifter (constraints) åt båda hållen.
- attribute (types) har en entydig relatering till ett Object (type) och liknar därför mer vanliga attribute types än domäner.
- DataType motsvarar domän och ligger som attribute type under MetaAttribute.

Anta att vi vill utbyta data om processer och deras informationsflöden, exempelvis att processen P1 skickar informationsflödet I1 till processen P2 (bland tusen andra processer och informationsflöden dem emellan). P1, P2 och I1 är alla exempel på objects inom en viss modell i CDIF:s terminologi.

En mellan parterna överenskommen meta-modell av informationsflödesbegrepp reglerar vilka typer av företeelser och samband som får hanteras. I denna meta-modell återfinns säkert begreppen *Informationsflöde* och *Process* samt sambanden *skickas från* och *skickas till*. Begreppen är exempel på två MetaEntities, sambanden på två MetaRelationships. *Skickas från*, t ex, sammanbinder två MetaEntities, nämligen *Informationsflöde* med *Process* med en semantisk riktning från *Informationsflöde* till *Process*. I själva verket har man inom CDIF valt att bygga meta-modeller så att ett visst MetaRelationship alltid utgår från en enda MetaEntity och till en enda MetaEntity, d v s så kallade binära samband. Begreppen MetaEntity och MetaRelationship med flera (som abstraktioner av meta-modell-nivån) och deras möjliga samband regleras i meta-meta-modellen. Se figur 3 där det just beskrivna från/till-förhållandet formuleras genom *HasSource* respektive *HasDestination*. Specialiseringar beskrivs genom *HasSubtype*-sambandet o s v.

Både en meta-modell och en meta-meta-modell är exempel på modeller av begrepp, samband, m m, som uttrycker abstraktioner av någonting. Om nu meta-meta-modellen kan användas för att uttrycka meta-modeller, varför inte också använda den för att uttrycka sig själv? Detta har CDIF tagit fasta på genom att beskriva meta-meta-modellen i samma grafiska notation som "vanliga" meta-modeller (se figur 3).

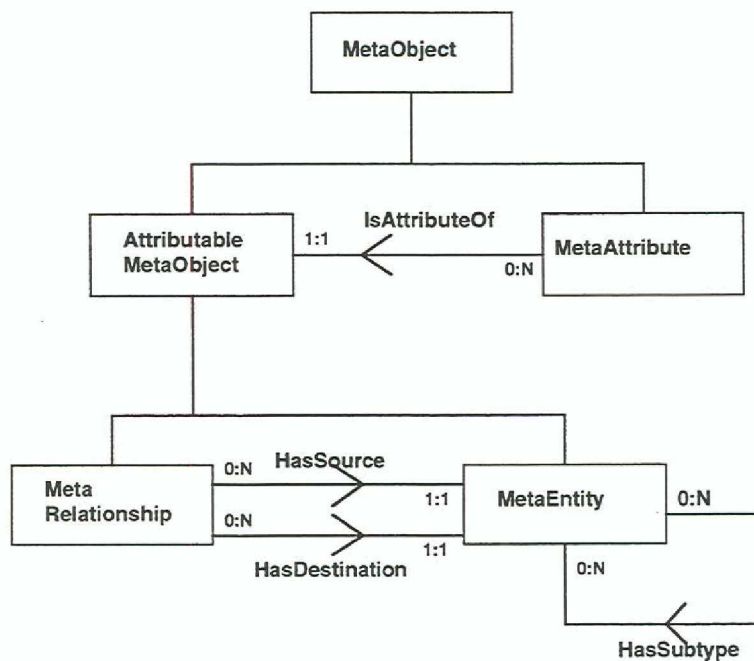
Att notera är att man har valt att konsekvent undvika suffixet "Type" för begreppen. För att poängtera att detta är en meta-meta-modell för att uttrycka meta-modeller, har begreppen lika konsekvent försetts med prefixet "Meta".

Som synes har sambandets huvudriktning indikerats med en pil. MetaRelationship pekar ut endast en (1:1) MetaEntity som "från-MetaEntity" genom *HasSource*. Motsvarande gäller för "till-MetaEntity" genom *HasDestination*. En MetaEntity kan figurera ihop med noll eller flera (0:N) MetaRelationships över såväl *HasSource* som *HasDestination*.



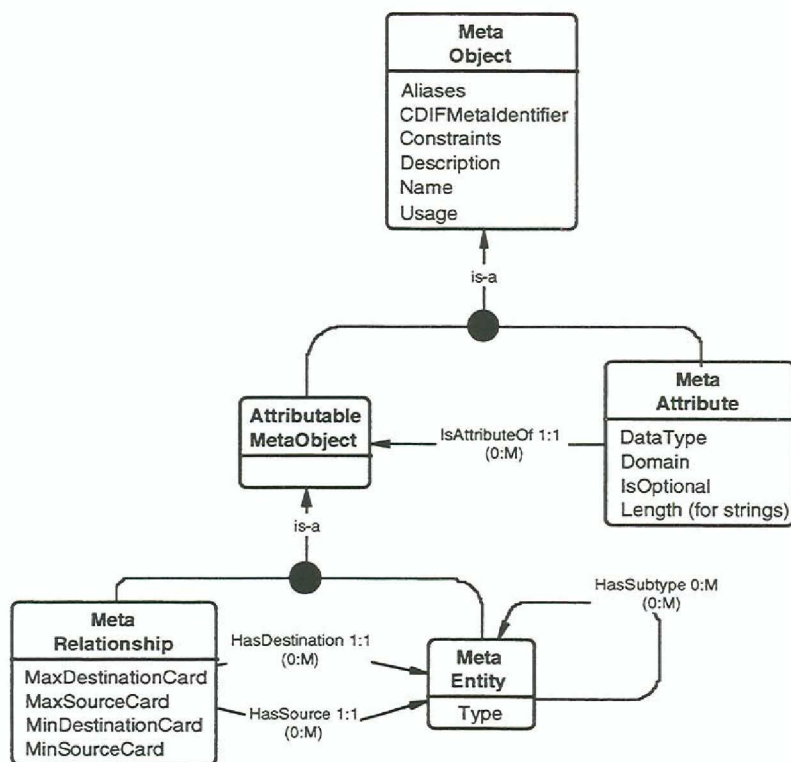
Varje företeelse på meta-modell-nivå är en specialisering av MetaObject.

Specialiseringar visas som en förgrening eller gaffel. Exempelvis är MetaRelationship och MetaEntity båda specialiseringar av AttributableMetaObject.



Figur 3

Samma meta-meta-modell uttryckt i den notation som används i Business Modeler, ett datorstöd som SISU utvecklat, visas i figur 4. Figuren har kompletterats med några attribute types. I fortsättningen använder vi av praktiska skäl denna notation för beskrivning av olika meta-modeller.



Figur 4

CDIF:s meta-meta-modell är en mycket enkel variant av en ER-modell. Eftersom syftet bara är att kunna formulera semantik i form av meta-modeller för överenskommelser om syn på och tolkning av översända data, finns inget egenvärde i att göra modellen onödigt komplex. Tvärtom. Däremot är förutsättningarna för den meta-modell som används för att precisera semantiken vid överföring av data enligt olika varianter av ER-modeller helt annorlunda. Den modellen måste i princip ha tillräcklig uttrycks kraft för att klara alla upptänkliga varianter av ER-modeller.

# 5. Existerande meta-modeller

Förhoppningen är att på sikt kunna definiera standardiserade meta-modeller för att täcka in de flesta eller åtminstone de vanligaste behoven av utbyte av Case-data. Ska varje typ av meta-modell till 100% kunna överföras, måste överförings-modellen, som just beskrivits, i princip vara ett superset av alla inblandade Case-verktygs egna modeller. Problemet uppstår när modeller i vissa punkter är varandra motstridiga. I sådana lägen måste förhandlingar tillgripas för att uppnå en för parterna rimlig kompromiss. En sådan supermodell kan också visa sig vara så rik att olika alternativa transformationsvarianter kan finnas. Följden kan bli att precision går förlorad.

Meta-modell-nivån delas naturligt in i ett antal väl avgränsade delmodeller där varje delmodell beskriver ett visst arbetssteg, typ av beskrivning, användarroll o s v. En sådan delmodell kallas inom CDIF för subject area, ämnesområde på svenska. IS-83 innehåller, som sagt, definition av fyra ämnesområden:

- en generell del (CORE)
- en ER-modell (ERA)
- en flödesmodell (DFD)
- en utbruten del av ER-modellen som beskriver olika aspekter på attribut, värden m m (Data Inventory).

Dessa presenteras grafiskt i följande avsnitt.

## 5.1 CORE

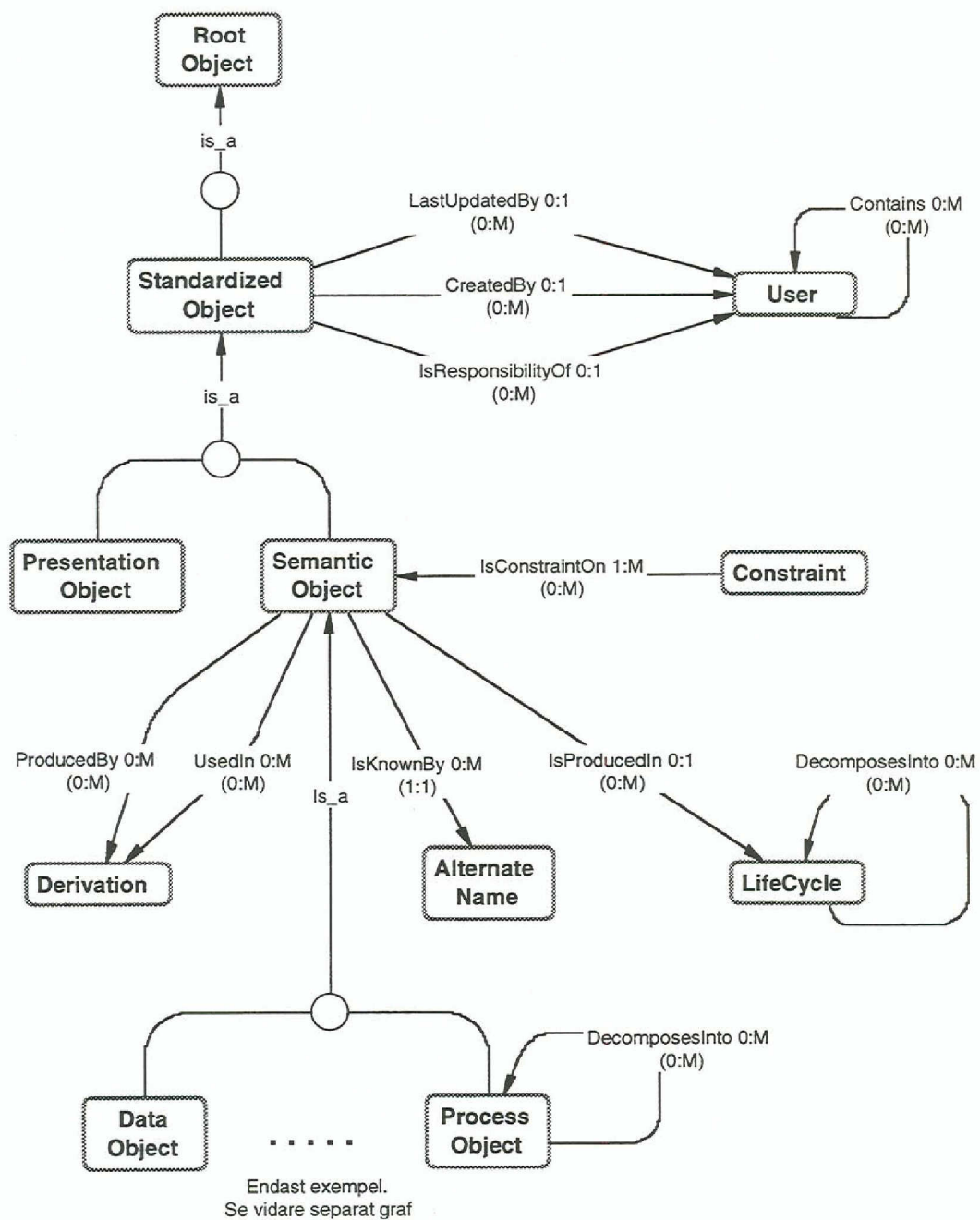
Inom detta ämnesområde återfinns de typer av objekt som har en viss generalitet, d v s de spelar en roll inom flera specifika ämnesområden. Hit hör till exempel de konstruktioner som står för de högsta generaliseringsnivåerna. Alla objekttyper är subtyper till **RootObject**. Alla objekttyper som finns definierade i CDIF-dokument är dessutom specialiseringar av **StandardizedObject**. Här återfinns exempelvis uppgifter om när objekttyper skapats och ändrats. **User** finns med för att beskriva vem som gjort vad med viss objekttyp och vem som har ansvar för den.

De som önskar tillföra egna modeller kan lägga dessa direkt under **RootObject**.

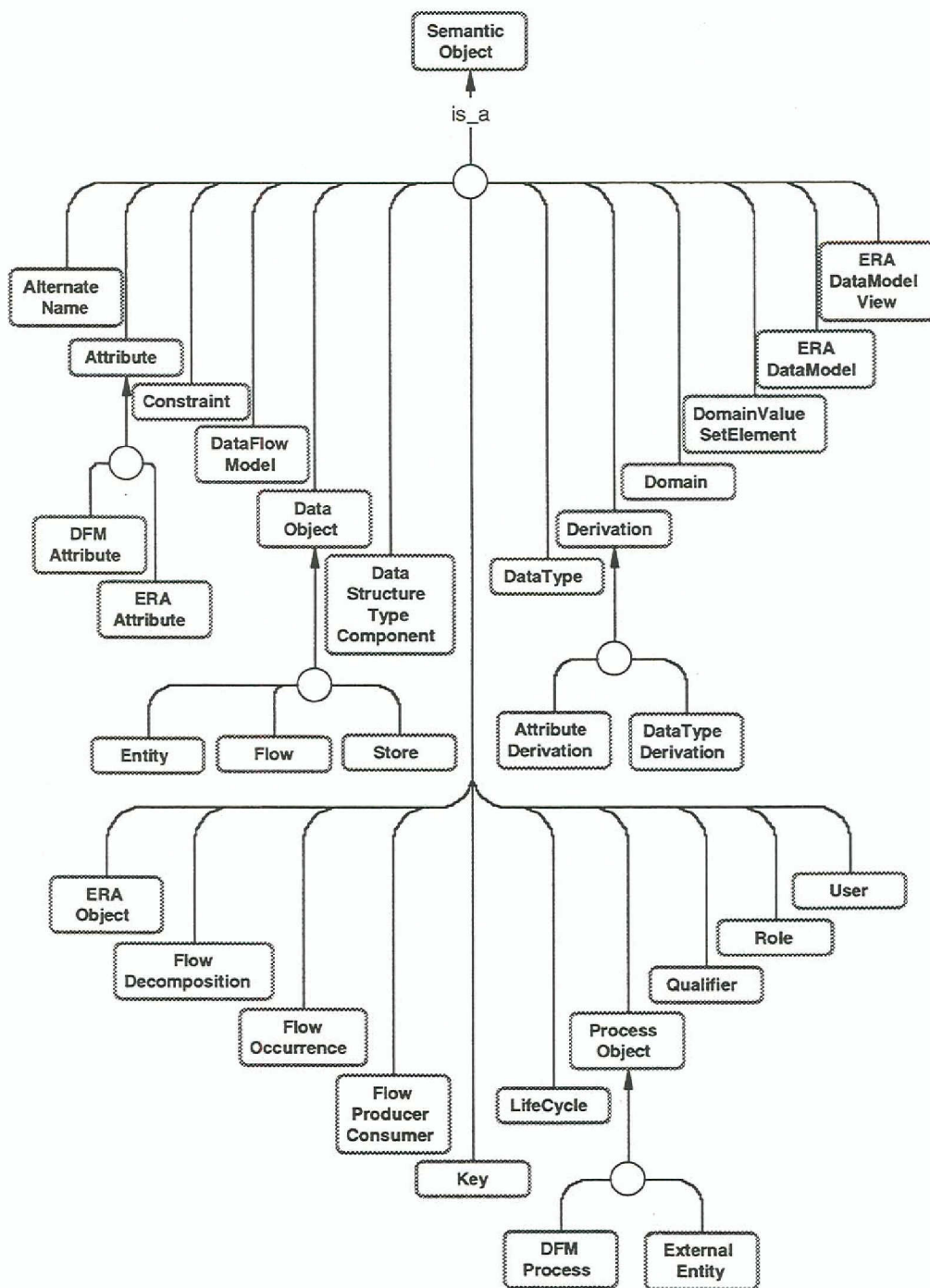
**SemanticObject** är den gemensamma supertypen för alla objekttyper som finns inom de semantiskt beskrivande modellerna, d v s för närvarande de i EIA/IS-83, part 1 definierade ämnesområdena. Part 2 innehåller de meta-modeller som beskriver grafiken. Samtliga objekttyper för detta ändamål ligger under **PresentationObject**.

För beskrivning av övriga objekttyper hänvisas till CDIF-dokumentet.

Förutom de specialiseringar som är visade internt inom varje ämnesområdes-meta-modell, finns de generella specialiseringar som visas i figur 6.



Figur 5



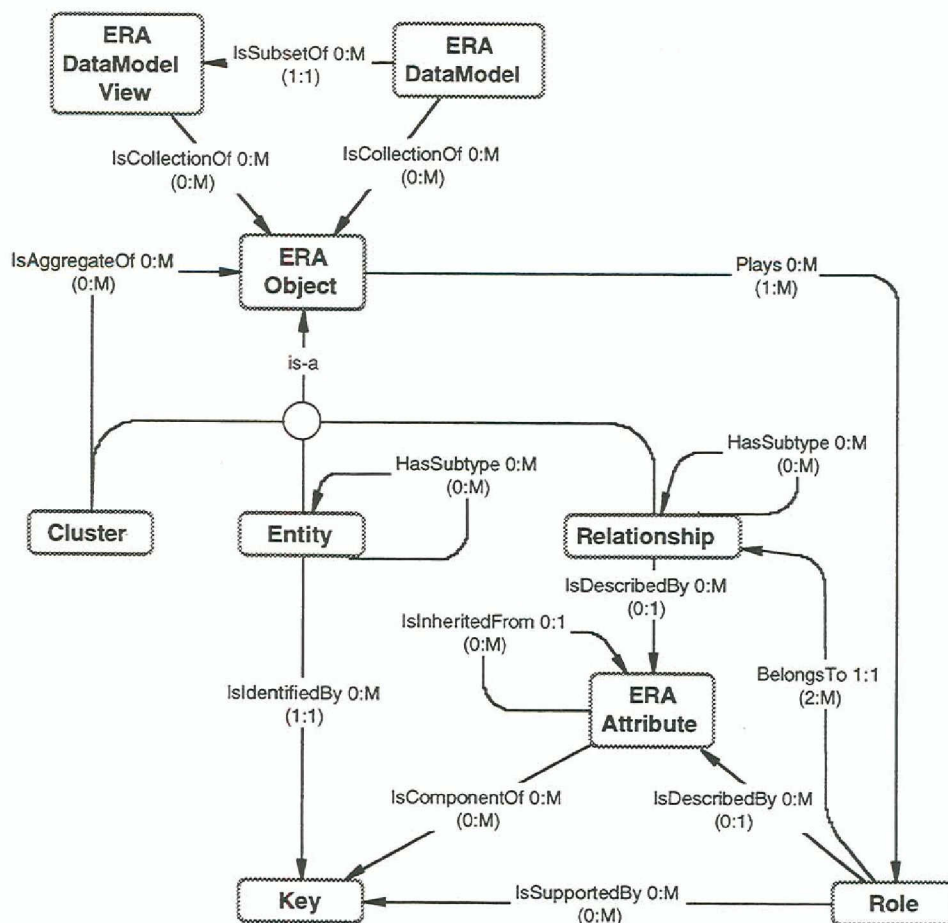
Figur 6



## 5.2 ERA

Denna ämnesområdes-modell klarar olika grader av avancerade ER-modeller. Det må gälla binära samband eller flerställiga samband, med eller utan attribut, det må gälla enkla eller multipla specialiseringsstrukturer, enkla eller komplexa objekt o s v. Eftersom alla objekttyper inom detta ämnesområde är subtyper till *ERAObject* kan enkelt olika önskade varianter av den mycket flexibla, fullständiga modellen skapas. Varje variant blir en *ERADataModelView*, som helt enkelt pekar ut de *ERAObjects* som ska ingå. *Cluster* fungerar som en gruppering av objekttyper för något ändamål. *Entity*, *Relationship* och *Key* är självförklarande. *Role* motsvarar ungefär det som i AD/Cycle Information Model kallas Relationship Link eller varje riktning eller aspekt av en sambandstyp. Eftersom samband här kan vara flerställiga blir minst två Roles inkluderade i varje Relationship. Det Role pekar på behöver inte vara Entities utan kan lika gärna vara Relationship eller Cluster (se koppling till *ERAObject*). Relationships kan alltså stå för samband mellan Relationships. Som synes kan ett Relationship ha attribut. Intressant och ovanligt är att varje Role också kan ha attribut. Se exempelvis PCTE:s meta-meta-modell (Triad-rapport K 22).

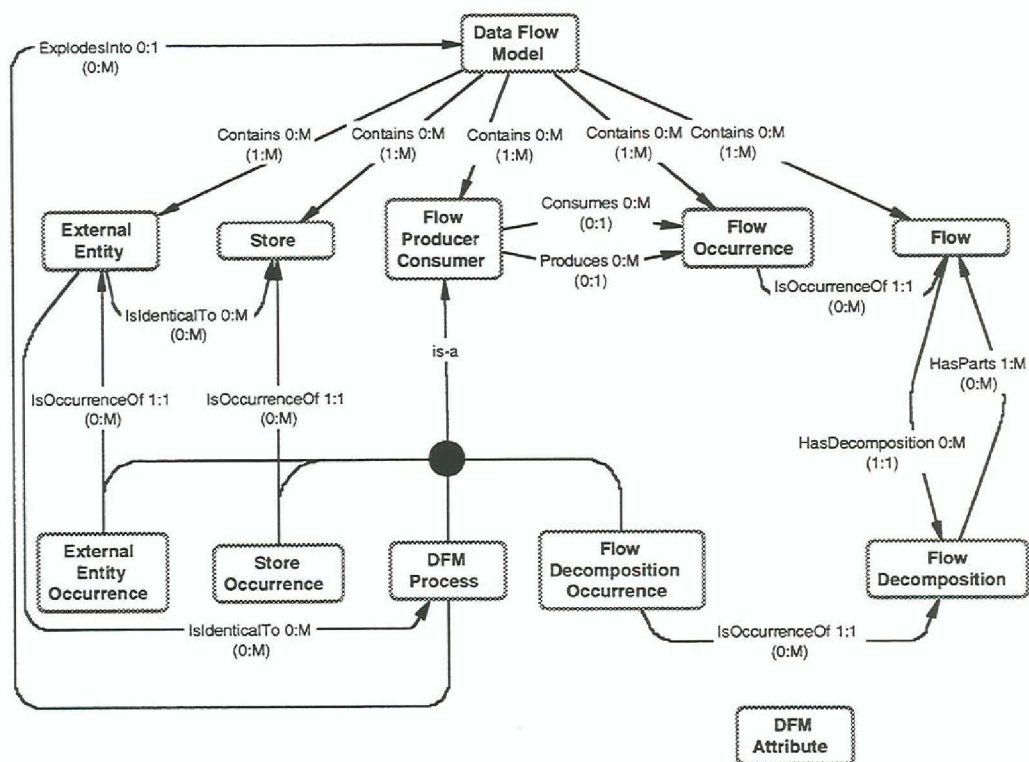
Den noggranne frågar sig säkert varför inte Entities har några kopplingar till *ERAAttribute*. De finns, men mycket indirekt! I figur 6 framgick att *Entity* är en subtyp till *DataObject*. I figur 9, där framförallt attribut- och värdesambanden visas, kommer det att framgå att *DataObject* beskrivs av (*IsDescribedBy*) *Attribute*. Frågan är om den principen är så finurlig. Eftersom *DFMAttribute* (attribut i Data Flow Model) också är en subtyp till *Attribute*, kan *Entity* beskrivas med ett sådant. Borde inte varje *DFMAttribute* ses enbart som en användning av ett *ERAAttribute* och ERA-modellen konsekvent stå för en abstraktion, d v s att *Entity* alltid bara kopplar sig till *ERAAttribute*. Så är ju fallet med *Relationship*. Kanske är detta ett tecken på faran att ha alltför heltäckande meta-modeller (supersets). Modellen blir så komplex att risken för inkonsekvenser, fel och misstolkningar ökar.



Figur 7

### 5.3 DFD

Data Flow Diagram meta-modell har en hel del likheter med motsvarande modell inom AD/Cycle Information Model, som utförligt beskrivits i Triad-rapport K 12. Jag nöjer mig därför med att visa modellen i figur 8 och hänvisa till till K12 och CDIF-dokumentet.

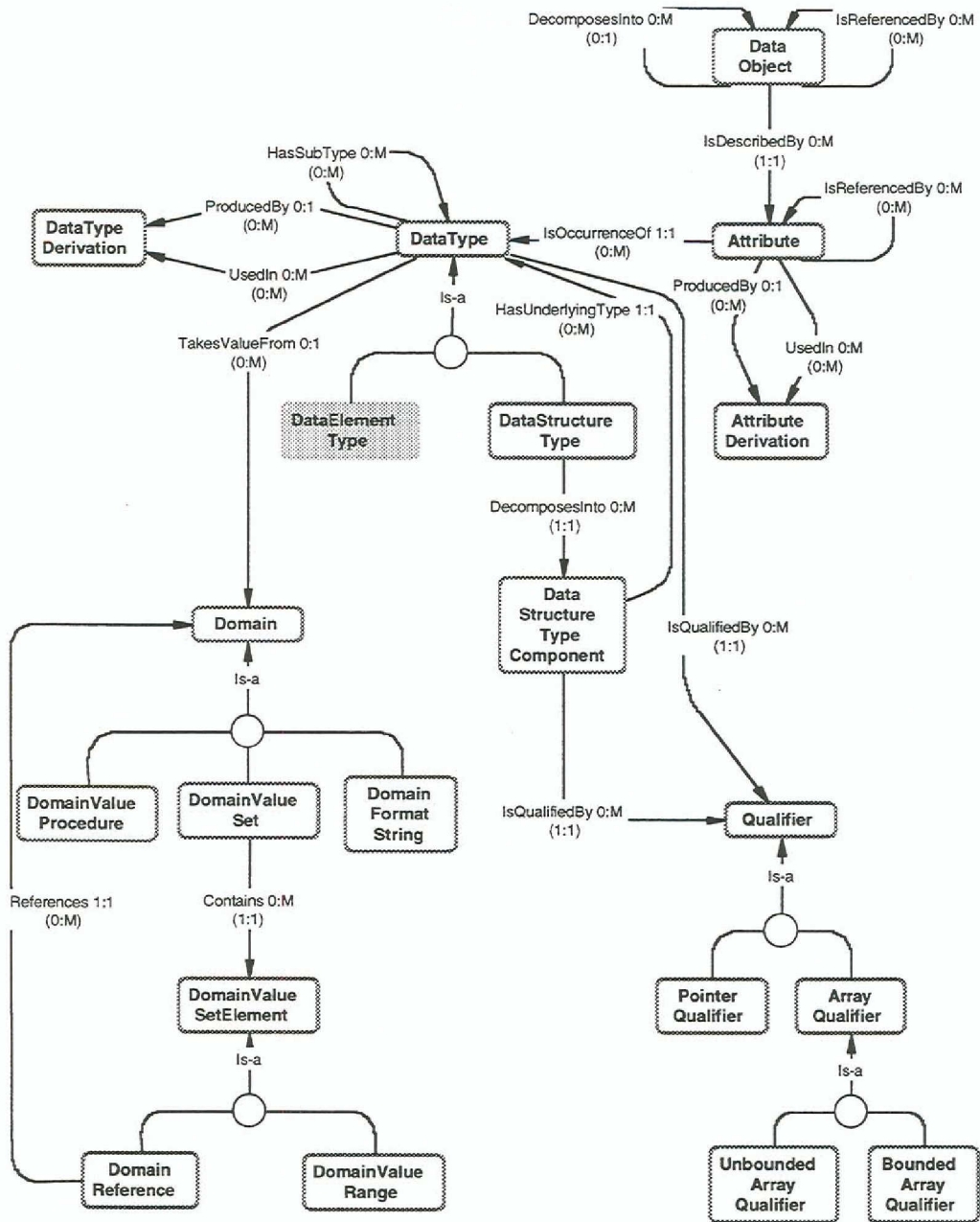


Figur 8

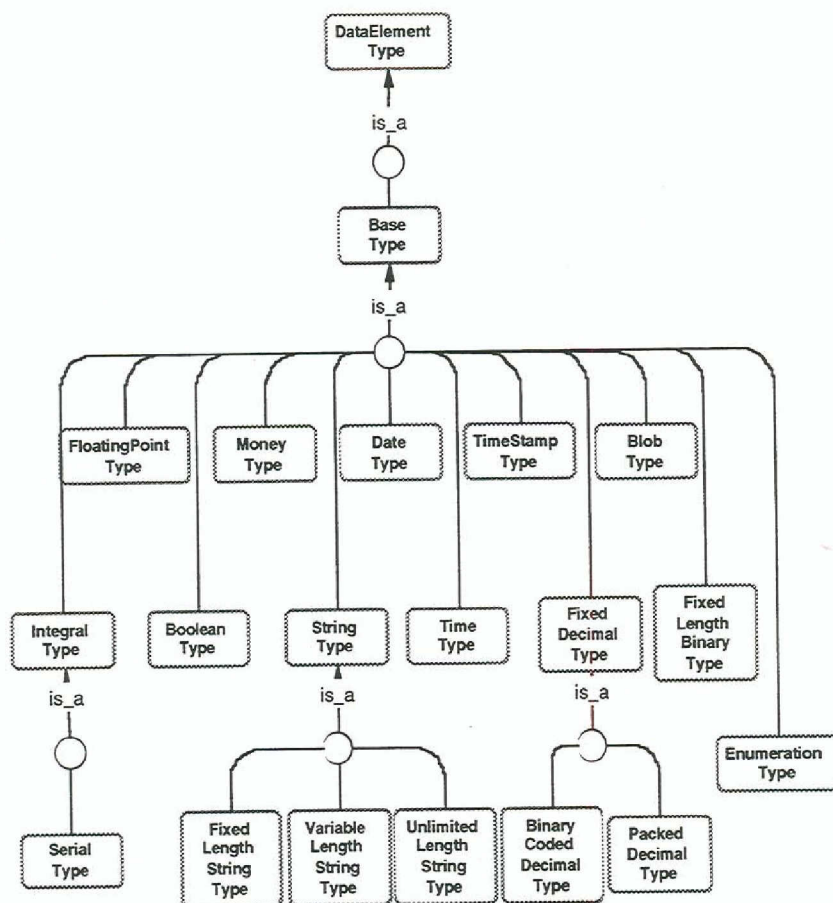
## 5.4 DATA INVENTORY

Inom detta ämnesområde beskrivs form, värdeområde, sort, struktur, m m för de typer av värden som är relevanta (se figur 9). En sådan *Data Type* beskriver uttrycksformen för ett *Attribute* (se *IsOccurrenceOf*). Motsvarigheten till CDIF:s *Data Type* i AD/Cycle Information Model är *Info Type*. *Domain* formulerar värdeområden. Någon exakt uppdelning i abstrakta värden (*Info Type*) och olika sätt att symbolmässigt formulera dessa (*Symbol Set*), som finns i AD/Cycle, tycks inte existera i CDIF. Däremot har man en mycket finindeldad substruktur för elementära datatyper (se figur 10). Sammansatta datatyper (*Data Structure Type*) kan definieras till godtycklig komplexitet genom att en komponent är en *Data Type* och därigenom kan vara antingen elementär eller sammansatt.





Figur 9



Figur 10

## 5.5 Presentation Model

De flesta ämnesområdes-modeller presenteras grafiskt. Förutom de olika modellkomponenternas innebörd och semantiska relateringar kan det ibland finnas anledning att också överföra information om den grafiska form modellen presenterats på sändarsidan. Framförallt är denna typ av information användbar om avsändar- och mottagarsidan båda tillämpar samma meta-modell och samma verktyg för att hantera en meta-modell. Även under andra förutsättningar kan grafiska uppgifter vara av värde, exempelvis för att så långt det är möjligt erbjuda att även olika verktyg kan erbjuda likartat grafiskt utseende (grafiska symboler och deras positioner).

CDIF innehåller en mycket ambitiöst utformad meta-modell för detta. Den begränsar sig för närvarande till sådan grafik som kan representeras med hjälp av **nodes** och **edges**. En node är en mer eller mindre komplex symbol. Dess skala kan ändras, dock inte dess form. En edge är också en mer eller mindre komplex symbol men med syftet att sammanbinda nodes. Den kan ändra form genom att den är töjbar. 3D-grafik kan beskrivas.

Presentation Model kan naturligt delas in i två delar. Den ena delen beskriver utformning av grafiska symboler generellt medan den andra delen beskriver symbolens användning i och för en viss modellkomponent i en viss semantisk modell. Den senare refererar till en viss plats i det grafiska rummet. Finessen med denna uppdelning är uppenbar. Vid en överföring räcker det med att föra över de olika typerna av symboler en gång. Därefter följer helt enkelt de uppgifter som justerar (exempelvis omskalning) och placerar ut dessa symboler i den aktuella modellen.

Det skulle föra för långt att gå in på meta-modellens alla detaljer. Den intresserade hänvisas till EIA/IS83, Part 2.

Antagligen kommer Presentation Model snart att behöva kompletteras i och med att Case-verktygen kan utnyttja multimedia.

## 5.6 Utbyggbarhet

Existerande meta-modeller kan utvidgas och nya meta-modeller kan skapas genom att inkludera:

- nya meta-entiteter
- nya meta-samband till existerande meta-entiteter
- nya meta-entitetsupertyper
- nya meta-attribut till existerande meta-entiteter eller meta-samband
- nya värden till existerande meta-attribut

Utvidgningen definieras i en viss sektion av en överföring, se avsnitt 6.2. Observera att ämnesområdes-modeller i CDIF:s perspektiv inte finns lagrade i någon resurskatalog, Case-databas eller dylik. Den entydiga definitionen återfinns endast i EIA/IS83. Sedan måste varje verktyg internt hantera dessa meta-modeller efter behov. En del verktyg kanske endast har en avtolkningsalgoritm baserad på den kunskap som hämtats ur CDIF-dokumentet, andra kanske lagrar de olika ämnesområdes-modellerna i en meta-databas så att avtolkningsfunktionen kan generaliseras mot aktuellt innehåll i metadatabasen o s v. I det förra fallet "dyker utvidgningar upp" enbart i samband med överföringen, i det senare fallet uppdateras först metadatabasen så att erforderlig utvidgningsinformation kan automatgenereras in i överföringen.

Utvidgningar av standardiserade meta-modeller bör användas med måtta och i ordnade former, annars är man snart tillbaka till "gå" igen. Gäller det dock nya ämnesområden för vilka ännu inte meta-modeller tagits fram, återstår förstås inget annat än att ta fram egna modeller.



# 6. Överföringsformat

En överföring, eller en **CDIF Transfer**, äger rum mellan en avsändare och en mottagare i en fördefinierad form. CDIF Transfer innehåller **modelldata** tillsammans med allt som en mottagare behöver för att kunna ta emot och tolka dessa data. Dit hör använd syntax och den aktuella meta-modellen, tillsammans benämnt **CDIF Transfer Format**.

En avsändare eller mottagare är normalt ett Case-verktyg eller en resurskatalogsfunktion. Inget hindrar dock att det exempelvis kan vara en godtycklig process eller ett objekt i en distribuerad miljö. Mellan avsändare och mottagare gäller vissa outtalade ansvarsförhållanden:

Avsändare:

- Ska inte utöka meta-modellen med saker som kan ge överlappande semantik med existerande definitioner.
- Ska leverera allt den kan enligt begärd förfrågan utan hänsyn till importörens "absorbtionsförmåga".
- Ska leverera värden för varje meta attribute som är satt till obligatoriskt (mandatory), även om sådant inte finns (null-värde).

Mottagare:

- Ansvarar för att kontrollera att översända data är korrekta och fullständiga.
- Måste syntaktiskt kunna ta emot alla levererade data, även om den inte kan tolka eller nyttja den.
- Arbetar alltid med hjälp av sin vy av det mottagna, sin Working Meta Model (WMM). Den kan genom direktiv ändras dynamiskt under en överföring.
- Måste ha varje refererad objekttyp definierad i WMM. Framåttreferenser godkänns ej.
- Tolkar utelämnat värde för frivilligt meta attribute som att värdet inte finns hos avsändaren.

## 6.1 Formatspecifikation

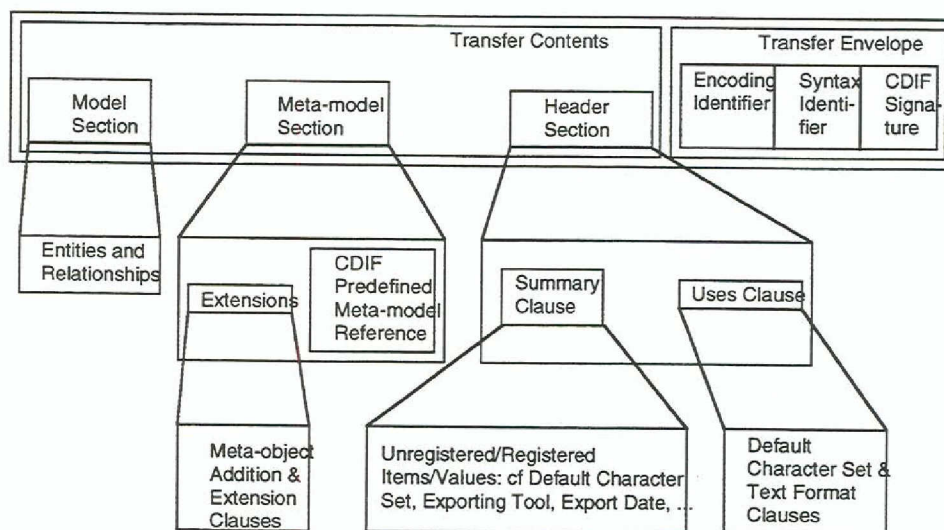
För olika sammanhang och under olika förutsättningar kan det vara fördelaktigt att kunna anpassa den syntax som används för en överföring. Samma data och meta-modeller kan ju packeteras på många olika sätt. CDIF sätter inga begränsningar därvidlag. Befintlig standard definierar i EIA/IS-82 en syntax under namnet SYNTAX.1. Den är uttryckt i en BNF-notation. Andra alternativ är att förvänta framöver. Exempelvis pågår arbete på en ASN.1-anpassad syntax. I princip kan varje syntax, som kan representera alla modelleringsbegrepp i meta-meta-modellen, föra över meta-modelldata och därmed även de data som svarar mot dessa meta-modeller.

Förutom syntax måste en mottagare veta vilken kodningsprincip (text, binär, krypterad, ...) som använts för överförda data. EIA/IS-82 definierar en kodningsprincip under beteckningen ENCODING.1. Denna är textbaserad och därmed läsbar av ett mänskligt öga. Andra varianter kommer sannolikt att tas fram.

En CDIF Transfer består av två delar. Den första delen, **Transfer Envelope**, innehåller dels en unik symbol som talar om att det är fråga om en CDIF-överföring, dels uppgift om vilken syntax och vilken kodningsprincip som används för resten av överföringen. TransferEnvelope har alltid en fast, fördefinierad syntax. Därefter följer **Transfer Contents**. Den delen är i sin tur indelad i tre separata delar där den första innehåller allmän information om den aktuella överföringen (**Header Section**), den andra referens till den aktuella meta-modellen som vid behov är kompletterad med utvidgningar (**Meta Model Section**) och den sista delen innehåller de förekomstdata som är det egentliga innehållet och syftet för överföringen (**Model Section**). I Model Section är varje dataelement angivet tillsammans med typangivelse från meta-modell-delen, allt för att åstadkomma total entydighet.

Varje objekt i en överföring ges en unik intern identifierare enbart för att vid behov unikt kunna skapa referenser mellan objekt inom en överföring. Observera att dessa identifierare inte har någon koppling till de eventuella intern-identifierare, som kan finnas för att identifiera objekt i en modell hos avsändaren. Dessa senare är helt och hållet beroende av hur modeller hanteras internt hos varje verktyg. Vissa använder sig av intern-identifierare, andra inte.

Figur 11 ger en översikt. För detaljinformation hänvisas till EIA/IS-82.



Figur 11

# 7. Pågående och planerade aktiviteter

Huvudsaklig inriktning för CDIF under den närmaste tiden kommer att vara mot utveckling av fler ämnesområdes-modeller, vilket verkar naturligt nu när Bob Matthews på IBM blivit ny vice ordförande i CDIF Technical Committee. Han är en av de ansvariga bakom AD/Cycle Information Model. Arbetet ska organiseras i Working Groups, samordnade genom en Integration Working Group.

Man utför också realistiska tester av olika slag för att kontrollera att både syntax och meta-modeller är rimligt kompletta och korrekta. Ett antal Case-verktygsleverantörer medverkar.

De ansvariga för arbetet med CDIF bedömer att det är värdefullt att ha en löpande samverkan med bl a:

- ISO SC7/WG11 (Software Engineering Data Description and Interchange)
- ECMA PCTE
- IRDS (ANSI och ISO)

Motsvarande åsikter har dessa organ gentemot CDIF.

## 7.1 Meta-modeller

Arbete pågår för att dels se över existerande meta-modeller, dels komplettera dessa med fler ämnesområden. AD/Cycle Information Model och Case-verktygs-modeller, bland andra, influerar dessa arbeten.

Aktuellt för revidering och nyutveckling är:

Foundation Subject Area  
Common Subject Area  
Data Inventory Subject Area  
Data Modeling Subject Area  
Data Flow Model Subject Area  
State/Event Model Subject Area  
Physical Relational Data Base Subject Area  
Presentation Location and Connectivity Subject Area  
Presentation Shape Subject Area  
Presentation Global Subject Area

Målsättningen är att dessa ämnesområdes-modeller först ska lämnas ut som Interim Standards och testas i en prototypfas. Därefter kommer de att ges ut som Recommended Standards och som en följd av detta bli föremål för en mer formell utvärderingsfas.



Nya moduler kan också förutses inom "lower-case tool"-området. Dit hör bl a:

- programdesign
- bildskärmsdesign
- databasdesign
- projektstyrning
- configuration management

## 7.2 Syntax

En ASN.1-anpassad syntax är under utveckling. Eventuellt kommer även en STL-syntax att tas fram (STL= syntax proposed by the IEEE Task Force for Professional Tools for Case Tool Intercommunication, P1175).

## 7.3 "Varudeklaration"

Innebörden av att vara CDIF-anpassad måste vara entydigt definierat för att standarden ska få genomslag på marknaden. Verktyg får inte fritt annonsera att de är "CDIF-compliant" om inte specifika villkor är uppfyllda. Arbete på en kravformulering pågår. Antagligen kommer två anpassningsnivåer att fastställas:

1. Full efterlydnad av alla CDIF-standarder.
2. Efterlydnad av någon specifik del, exempelvis någon ämnesområdes-modell.

Beträffande b-nivån är det alldeles nödvändigt att en Case-leverantör deklarerar om all egen, intern modellsemantik kan uttryckas i CDIF. Åt andra hållet vore det upplysande om de kunde informera om vilka CDIF-konstruktioner de inte kan ta emot eller ta emot med någon form av transformation (i så fall redovisa vilken). Utan dessa uppgifter går det inte att klargöra vilken integration mellan verktyg som i realiteten är möjlig.

# 8. Några avslutande kommentarer

- Dokumenten är föredömligt rediga och självförklarande. Både exempel och frågor/svar finns inkluderade. Tyvärr är dokumenten ganska omfattande till omfånget. Dock beror det snarare på att respektive modell är mycket exakt och utförligt beskriven än att standarden i sig är komplex. Den nya versionen förutspås bli betydligt mer kompakt, utan avkall på läsbarhet.
- CDIF är för närvarande det enda organ vars arbete inom ämnesområden tycks ha ordentlig genomslagskraft. I alla händelser refereras ofta CDIF:s arbete inom detta område.
- CDIF har en mycket hög ambitionsnivå avseende meta-modellerna. Superset-principen gäller. Det behöver dock inte alltid vara en styrka eftersom det kan innebära godtycke eller många olika men likvärdiga sätt att formulera samma sak. Detta kan innebära ett problem för verktygen när de ska transformera en formulering mellan den egna modellen och CDIF-modellen.
- Mycket jobb finns kvar med alla ämnesområden, både vad gäller ändringar och nya områden. Antagligen kommer samarbete med andra organ att behövas.
- Presentationsmodellen är ganska oprövad men mycket detaljerad. Dock finns inget direkt stöd för andra typer av grafer än rena grafiska symboler, exempelvis tabeller, matriser och screen layouts.
- CDIF innehåller inget stöd för (garanti för) att det som överförs är formellt riktigt och fullständigt. Mottagaren måste kontrollera detta.
- Möjligheten till egna utvidgningar av subject-area-modeller kan undergräva integrationssträvanden.
- CDIF är förhållandevis oprövat men ganska riskfritt. Det kräver inga omfattande grundinvesteringar. Man kan börja med överföringar baserade på någon mycket begränsad meta-modell och sedan successivt expandera.
- Ingen bindning till viss syntax erfordras. Utmärkt!
- Det återstår sannolikt en hel del att göra beträffande principer för överenskomelser mellan avsändare och mottagare, fördefinierade utväxlingar, förändringshantering, m m. Kontrakt?
- CDIF ligger inom ett nytt område och det finns flera pågående parallella standarder (exempelvis IRDS Export/Import). Det finns dock förhoppning om



ökat samarbete, exempelvis med ISO/SC7/WG11, så småningom. Full samordning kanske varken är önskvärd eller realistisk.

- Möjlighet att överföra Management Control Data saknas (eller är liten). Hur överförs exempelvis behörighetsinformation? Denna brist behöver ses över antingen genom kompletteringar i meta-modellen eller på annat sätt. Jämför med PCTE och IRDS som båda poängterar vikten av dylika uttrycksmöjligheter.

I en Case-värld under snabb utveckling och där idén med integrering genom resurskataloger inte stabiliserats, står CDIF för en rimligt ambitiös integreringsansats. Att börja med en CDIF-baserad integrering är ganska ofarlig. Att i ett senare läge orientera sig mot en integrering baserad på resurskataloger borde inte innebära några principiella svårigheter.

Till sist ett stort tack till Katarina Kindwall som så brutalt men mycket välmotiverat redigerat rapporten.

