

Krav på IA

Nästa Generation Modellering

Avancerad utbildning för handledare

**Katalogprinciper**

Uttagssystem

Informationspridning

**Rapport K nr 1: IRDS**

Rapport K nr 2: IRDS Modeller och modellnivåer

Rapport K nr 3: Koppling 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

# IRDS Information Resource Dictionary System.

En kortfattad beskrivning av ISO:s förslag till standard.

## Örjan Jonsson

Televerket, Avdelningen för Grundteknik

### Spridningsförbehåll:

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

© TRIAD-parterna 1991

Rapporten är skriven i och för TRIAD delprojekt Katalogprinciper.

# IRDS

Information Resource Dictionary System.  
En kortfattad beskrivning  
TRIAD Katalogprinciper - ARBETSRAPPORT

## INNEHÅLLSFÖRTECKNING

1. SAMMANFATTNING.....	3
2. BAKGRUND .....	4
3. VAD ÄR ETT IRDS ? .....	5
3.1. Begreppsapparaten .....	5
3.2. Innehållet i IRDS .....	7
3.3. Arkitekturen i IRDS .....	9
3.4. Hur kan man använda IRDS ? .....	11
4. VAD ÄR PROBLEMEN ? .....	13
5. VILKEN BETYDELSE FÅR ISO-IRDS ? .....	14
6. REFERENSER .....	15

Information Resource Dictionary System. En kortfattad beskrivning

TRIAD Katalogprinciper - ARBETSRAPPORT

TRIAD Katalogprinciper Rapport K nr 1  
Information Resource Dictionary System. En kortfattad beskrivning- Arbetsrapport  
Örjan Jonsson  
© TRIAD-parterna/SISU 1991

# 1. SAMMANFATTNING

Hur man skall samordna den information som hanteras vid systemutveckling är ett högaktuellt problem. Att det skulle finnas en modell som vore direkt anpassad till varje företags specifika behov, syns vara en orimlighet. Att det däremot finns en generell modell som går att anpassa till specifika behov är IRDS-standardiseringsarbetet ett bevis på. Denna rapport försöker ge en övergripande förklaring till det arbete som försigår inom ISO.



## 2. BAKGRUND

Sedan lång tid tillbaka har man lagrat delar av den information man använder sig av i ett systemutvecklingsarbete, i vad man brukar kalla datakataloger. I en sådan är det vanligt att man kan hantera begrepp såsom termer, poster, program, program som uppdaterar post "A" osv. Datakatalogens begränsning bestod i att den var mycket statisk eftersom den datamodell som katalogen byggde på inte var anpassningsbar för nya behov.

I modern systemutveckling ställs högre krav på hantering och samordning av information. Denna samordning är nödvändig, bla eftersom det blir allt vanligare att man använder sig av olika slag verktyg vid systemutvecklingen ( sk CASE-verktyg).

En sådan samordning sker lämpligen via ett gemensamt "repository", "encyclopedia" eller resurskatalog som är den benämning som bl a används av [RDF-CASE] och i fortsättningen av denna rapport. I resurskatalogen lagras all information som tillhör ett projekts alla olika faser. Informationen kan exempelvis vara text, figurer, diagram, källkod, beskrivningar av data, funktioner, projektdeltagare och aktiviteter. I resurskatalogen lagras även data om de relationer som finns mellan olika informationsobjekt. Lagringen ska ske på ett sådant sätt att alla CASE-verktyg kan tolka informationen och återanvända denna.

Dagens CASE-verktyg upplevs dock ofta som "isolerade öar" utan förmåga att kommunicera sinsemellan. Detta beror på att en etablerad standard saknas för att kunna samordna informationen mellan verktygen.

Både ANSI och ISO arbetar med standardisering för att lösa ovan nämnda problematik. Denna benämns IRDS, (Information Resource Dictionary System). ANSIs förslag godkändes som standard 1988. De leverantörer som idag påstår sig följa IRDS-standarderna syftar till denna ANSI-standard. ISOs förslag är till större delen ännu inte godkänt som standard, men betraktas som mera heltäckande och utgör bakgrundsmaterial för denna rapport.

Styrkan i IRDS ligger i den flexibilitet man har i att utforma en informationsmodell som är anpassad till den egna organisationen. Man kan sedan anpassa denna modell alltefter nya behov dyker upp.

## 3. VAD ÄR ETT IRDS ?

Ett IRDS adresserar främst tre områden

- den ger en flexibel datamodell
- den svarar för generella tjänster som versionshantering och säkerhet
- den ger informationsgränssnitt i form av skärmbilder för slutanvändare, bryggor till andra resurskataloger samt åtkomst av data för CASE-verktyg och tillämpningar

ISOs standard är till stora delar, fortfarande bara ett förslag, men det grundläggande konceptet utgör en nyligen antagen standard kallad IRDS FRAMEWORK, beskriven i [ISO 10027].

I denna referensram definierar ISO dels den begreppsapparat man använder sig av men också en generell IRDS-arkitektur i form av komponenter och gränssnitt mellan dessa.

### 3.1. Begreppsapparaten

Grunden i IRDS är baserad på en relationsmodell, där SQL används som specifikationspråk, med följande fyra definierade datanivåer:

"IRD Schema Definition Level"	- Definitionsschema	nivå 4
"IRD Definition Level"	- Definition	nivå 3
"IRD Level"	- Innehåll	nivå 2
"Application Level"	- Tillämpning	nivå 1



		Application Level Pair	IRD Level Pair	IRD Definition Level Pair
4	IRD Definition Schema Level			IRD Definition Schema
3	IRD Definition Level		IRD Schema	IRD Definition
2	IRD Level	Application Schema	IRD	
1	Application Level	Application		

I praktiken använder man främst de två mittersta nivåerna i en resurskatalog (Definition och Innehåll), Det är på nivå 2 och 3 som de flesta systemutvecklare kommer att komma i kontakt med IRDS. Här återfinns beskrivningar och definitioner av de tillämpningar man arbetar med, medan de övriga nivåerna finns med främst för att erbjuda en fullständig bild. Begreppet "Level Pair" används för att visa att varje gång man tänker sig en operation mot data i en IRD, så opererar man på två nivåer. Den översta nivån är alltid den som innehåller definitionerna av den underliggande nivån. Ett annat sätt att uttrycka detta är att varje objekt på nivå N+1 finns instansierad på nivå N. Den övre nivån benämns "schema" för den underliggande nivån. Exempelvis utgör "Application Schema" på nivå 2 en beskrivning över "Application" på nivå 1.

Man använder ofta begreppen META-data och META-META data när man diskuterar flexibiliteten i ovanstående modell. Med META-data avses den beskrivning som finns om det data som lagras. Man avser alltså IRD Level nivån. Med META-META data avses den beskrivning som finns av META-data-nivån, och man befinner sig alltså på IRD Definition Level.

## 3.2. Innehållet i IRDS

De exempel som återfinns i detta kapitel ger endast anspråk på att försöka förtydliga strukturen i IRDS-standarden och avser inte att försöka ge en komplett bild av strukturen.

Modellen bygger på typer och instanser/förekomster av dessa typer. En objekttyp "TABELL" kan ha instanserna/förekomsterna "LÖNE\_TABELL" och "PERSON\_TABELL" definierade på underliggande nivå. I stället för de allmänt nyttjade begreppen "Entity" och "Relationship" använder sig ISO av begreppen "Object" och "Association".

På definitions-schemanivån uttrycks de definitioner som ligger till grund för resterande del av kataloginnehållet. För att vara tydliga har ISO valt att uttrycka dessa med hjälp av SQL. Ändringar och tillägg på denna nivå sker via standardisering av ISO. Några exempel på hur typerna på definitions-schemanivån kan vara instansierade på definitionsnivån kan vara:

Object-type	Association-type	Attribute-type
IRDS-USER	HAS	LENGTH
TABLE	CONTAINS	ADDED_BY
COLUMN	PROCESSES	
DOCUMENT	PROCESSED_BY	
PROGRAM		
CONTAINS		
LENGTH		

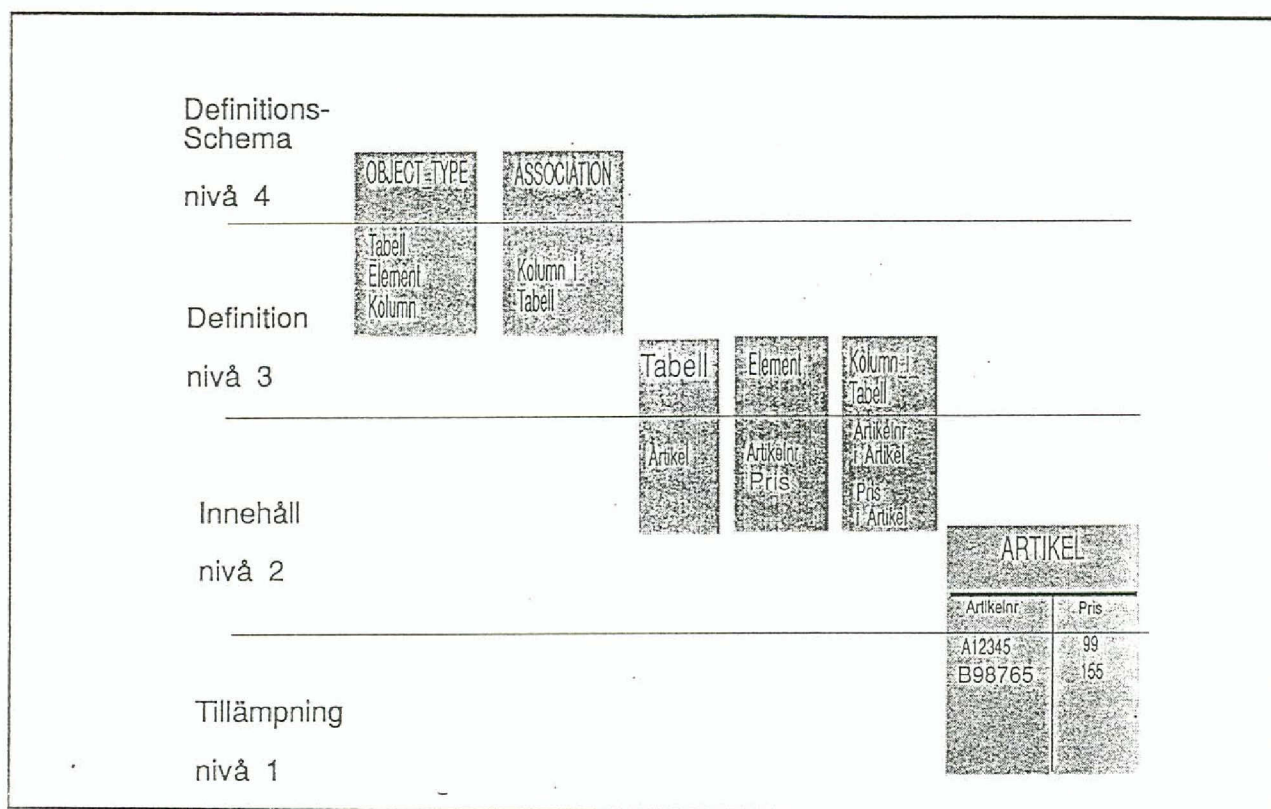
Observera att CONTAINS och LENGTH finns definierade som objekt av typen "Association-type" respektive "Attribute-type". Det är också möjligt att definiera "reverserande" relationer som PROCESSES och dess motsats PROCESSED\_BY.

Definitions-nivån är beskriven i standarden i form av 32 st tabeller. Innehållet i dessa tabeller representerar den tillåtna strukturen i den aktuella resurskatalogen. Det är denna nivå som ger en flexibilitet att modellera de begrepp och data man arbetar med.

Innehållsnivån motsvarar den datanivå som man traditionellt brukar beskriva för en tillämpning i form av postbeskrivningar och tabellbeskrivningar. Analys- och design- information är ett annat exempel på information som lagras på denna nivå.



Tillämpningsnivån utgör den del av informationen som man beskriver och kontrollerar med hjälp av sitt datakatalogsystem. Det är dessa data som utnyttjas i traditionella tillämpningar som faktureringsystem eller kundreskontra, och som lagras i traditionella databaser. Observera att lagring av dessa data inte sker inom IRDS och att IRDS inte heller uttryckligen kräver att dessa data lagras i en databas, utan data kan finnas under ett filhanteringssystem av godtycklig typ.



Figuren ovan är ett förenklat exempel på en kedja av IRDS-definitioner som hör samman. På definitions-schemanivån finns de objekttyper som definieras av standarden dvs OBJECT\_TYPE och ASSOCIATION. OBJECT\_TYPE har instanserna Tabell, Element och Kolumn på definitionsnivån. Dessa bildar i sin tur objekttyper för vad som definieras på tillämpningsnivån. Artikel är en instans av typen Tabell. Artikelnr är en instans av typen Element. Dessa bildar i sin tur objekttyper för vad som lagras i operativa databaser ex ett artikelregister.

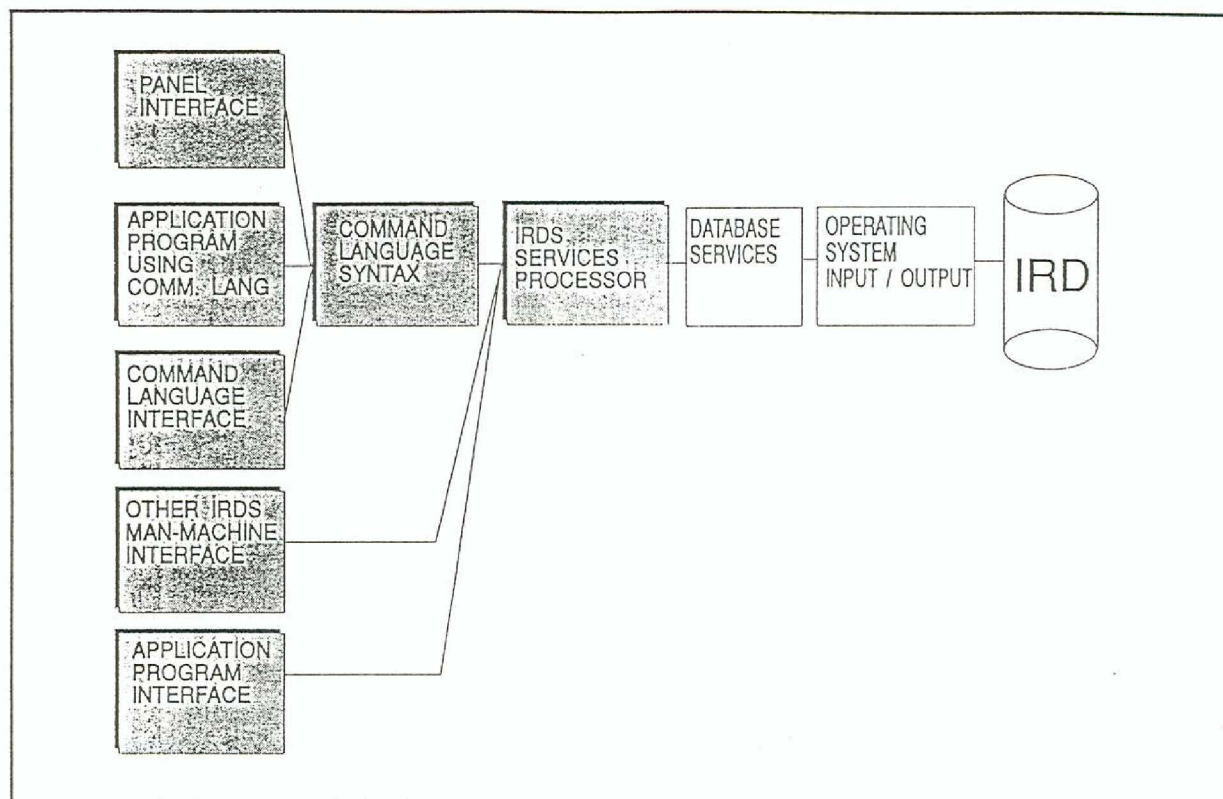
För dem som önskar läsa mera om modellnivåerna rekommenderas Triad-rapporten

"Stig Berild: IRDS:Modeller och modellnivåer".



### 3.3. Arkitekturen i IRDS

Förutom specifikation över hur man lagrar och definierar data i en IRDS, så innehåller ISO- standarden även ett antal definierade komponenter och gränssnitt enligt modellen nedan.



ISO anser att en del av en resurskatalogs roll är att kunna fungera i en miljö, där processer kommunicerar med varandra, och har därför utformat arkitekturen till att kunna hantera detta.

Till de viktigaste gränssnitten hör:

#### IRDS Panel Interface

Syftar till att definiera utseendet på de formulär som används för att uppdatera data eller datadefinitioner i en IRDS. Tanken är att ett gemensamt utseende ska finnas hos alla leverantörers implementering av IRDS-standarderna. Man definierar dock inte ingående skärmbilders utseende i detalj utan anger olika areor där

informationen skall placeras. Man anger även logiken i människa-maskin-dialogen.

Standarden definierar följande presentationsareor:

-**Status**, visar vilken IRD som accessas eller vad IRDS arbetar med som t ex "uppdaterar katalogen" eller "hämtar information".

-**Schema**, visar möjligheterna att navigera i ett visst IRD schema

-**Händelser**, visar vilka möjligheter man har att byta till underliggande formulär. Innehåller dessutom en "GODKÄNNANDE"-funktion där man bekräftar en gjord ändring.

-**Meddelande**, varningar och felmeddelanden presenteras

-**Hjälp**, vid begäran så presenteras hjälp-information här

### IRDS Command Language Interface

Standard för att möjliggöra åtkomst av data och datadefinitioner genom ett kommandospråk. (jämför SQL). Knutet till detta gränssnitt är också sättet man utnyttjar kommando-språket på. Ett separat gränssnitt finns för om man nyttjar ett applikationsprogram, färdiga paneler eller ställer frågor direkt i kommandospråket.

### IRDS Services Interface

Fungerar som ett generellt kommunikationsgränssnitt mellan dataprocesser och ger åtkomst till data i IRD. Detta gränssnitt är det som har kommit längst i standardiseringsväg. Ett förslag finns framtaget i form av en "Committee Draft" [CD 10728], men är fortfarande ej godkänt.

Via detta gränssnitt kan man nå och manipulera data på nivå 2 och 3. En applikation (t ex ett CASE-verktyg) som försöker nå data, gör alltid detta på uppdrag av en viss användare och det är dennes identitet som avgör behörighet. Åtkomst av data sker alltid genom att man "navigerar" från en etablerad utgångspunkt "Start Object" genom strukturen till det objekt man vill nå, "End Object". Först när man är framme kan man manipulera dess data. Interaktionen sker alltid i form av transaktioner. Den tid vilken en användare är ansluten till IRDS kallas en "session". En session kan bestå av en eller flera transaktioner. Varje session inleds med en "OPEN IRDS" och avslutas med en "CLOSE IRDS". Däremellan kan enskilda transaktioner utnyttja COMMIT eller ROLLBACK.

För övriga komponenter som finns i vänstra delen av figuren ovan, har det inte fastställts, att någon standardisering kommer att ske. De komponenter som finns i figurens högra del kommer inte att definieras inom ISOs ramverk, utan t ex

Database Services Interface förutsätts finnas implementerad i den DBMS som katalogen realiserar i.

För att möjliggöra överföring av data mellan olika IRDS kommer en definition att göras på export/import-fileras innehåll. Denna standard kommer dock att vara åtskild från IRDS.

### 3.4. Hur kan man använda IRDS ?

De tjänster som en IRDS ska kunna fullgöra enligt ISO, är dels sådana som man förväntar sig ska finnas i de flesta DBMS, dels några som är specifika för just IRDS.

Bland de tjänster som ISO anser att den underliggande databashanteraren bör svara för finns bl a:

- **Fördefinition av giltiga värden i attribut**, t ex värdet i attributet DOKUMENTFORMAT skall vara någon av A4, A4L, A5 eller A3
- **Accesskontroll** i betydelsen att användare XXX får inte läsa tabell YYY
- **"Audit trail"** d v s man skall kunna se att användare XXX uppdaterade tabell YYY den 23 oktober 1990
- **Gräns- och default-värden i attribut**, d v s att attributet ARTIKELNR är större än 0 och mindre än 999 999 eller att attributet MOMSPLIKTIG alltid sätts till JA om inte annat anges
- **Integritet i databasen** innebär att en transaktion alltid utförs helt eller inte alls
- **Fråge- och rapportmöjligheter**, t ex definiera nya frågor och vyer
- **Åtkomst till data från andra maskiner**

Bland de tjänster som IRDS bör svara för finns:

- **Namnsättning**, skall garantera att varje definierat objekt har ett unikt namn
- **Statushantering** av innehåll, en specifikation kan vara preliminär, godkänd eller inaktuell
- **Versionskontroll**, det måste gå att lagra definitioner i flera versioner
- **Uppdelning**, ett system kan bli uppdelat i mindre system. Ett visst givet objekt kan dock tillhöra endast ett definierat system



- **Indelning**, snarlikt begreppet ovan med den skillnaden att indelning mera görs med syfte att styra användares åtkomst till endast ett visst system eller en viss version av system
- **Skapande av kopior**
- **Analys av verkan vid förändring**, d v s man skall kunna få besked om hur en viss förändring påverkar systemet i stort

Några exempel på olika användningsområden för en resurskatalog kan vara:

**Dataadministration**, detta är det primära och viktigaste användnings-området. Innebär att definiera och underhålla beskrivningar över datastrukturer, definitioner och processer.

**Databasadministration**, fysisk datamodellering, tuning och definitioner av recovery - och restartrutiner

**Nätverksadministration**, Komponenter och förbindelser i nätets arkitektur kan beskrivas,

**Projekthantering**, för planering, uppföljning, granskning och kontroll av ett projekts hela livscykel

**Dokumentation**, bl a genom att man själv kan definiera olika typer av rapporter

**Kvalitetssäkring**, genom att man kan tillförsäkra sig om att man följer regler beträffande metoder, dokumentation o s v.

**Systemutveckling**, beskrivning för systemerare hur användarkrav övergår till specifikationer och design, hjälper programmerare att konstruera och testa program, stöd vid underhåll i form av datadefinitioner och process-beskrivningar.

## 4. VAD ÄR PROBLEMEN ?

Man har i arbetet med ISO-IRDS utgått från att specifikationen skall ske i någon form av standardiserat specifikationspråk. Alternativen som fanns var NDL för nätverksdatabaser och SQL för relationsdatabaser. Ganska naturligt valde man SQL.

Inom det standardiseringsarbete som pågår inom ANSI har man däremot använt sig av en ER-modellerings teknik för att beskriva standarden.

ANSI tillåter t ex:

- en relation mellan fler än två entiteter
- attribut till relationer
- relationer kopplade till relationer

Den stora skillnaden i synsätt mellan de två olika förslagen har lett till att samarbetet mellan dessa inte har fungerat bra. Denna situation snabbar inte upp standardiseringsarbetet i en period där marknaden efterlyser ett klart besked om hur standarden skall se ut.

Man är inom ISO medveten om att det finns ett stort behov av att utbyta information mellan olika IRD. Ett exempel kan vara att man vill låta nya projekt hämta information till ett lokalt IRD från ett centralt IRD. Tyvärr så säger inte standarden någonting om hur detta skall lösas. Problemet för de som specificerar standarden är att den grundläggande tekniken för vad man önskar inte alltid finns tillgänglig. T ex innan man kan uppnå en distribuerad heterogen lösning på katalogproblematiken, så finns det, redan i befintliga standarder som SQL, grundproblem som är olösta.

Ex: så finns det ingen överenskommen lösning till "Two Phase Commit", dvs en ömsesidig bekräftelse på att man kan låta en uppdatering ske mellan flera databaser.

Ett annat problem man har ställts inför är att SQL inte riktigt räcker till att beskriva en så pass komplex sak som en katalog eftersom datadefinitionsspråket i SQL har sitt ursprungliga syfte i att hantera nivå 2 i IRDS-modellen. Numera använder sig man inom IRDS-arbetet av syntax från kommande SQL2-standard. Detta måste dock koordineras med dem som specificerar denna standard och allt detta tar tid. Dessutom gör det IRDS-standarderna mera svårtolkad för dem som inte är insatta i kommande SQL2-syntax.



## 5. VILKEN BETYDELSE FÅR ISO-IRDS ?

Det finns ett stort behov av en internationell standard och det hade varit trevligt att kunna avsluta denna rapport med att påstå att ISO-IRDS snart skulle finnas som realiserade och färdiga produkter. Så är dock inte fallet. Dagens situation är betydligt mer komplex!!!

Nuläget är att det finns ett antal alternativa lösningar till hur man skall organisera sin information. Förutom ISOs arbete så föreligger en standard från ANSI som är godtagen i USA. Redan idag hävdar ett antal leverantörer att man följer denna standard. Till denna så finns det förslag till utökningar från både IBM och Digital. IBMs förslag till ANSI är inte överensstämmande med den egna produkten MVS/Repository som kommer att ingå som en kärna i AD/Cycle. Digital's förslag till ANSI är baserat på ett nytt objektorienterat synsätt. Inom UNIX-världen så lanserar ECMA (European Computer Manufacturers Association) den specifikation till integrationsplattform, PCTE som ett "UNIX repository" följande ANSI-standarderna.

Mångfalden av produkter och förslag är ett resultat av det intresse för kataloger som vi kunder visat under senare år. Ett troligt scenario är att IBMs Repository kommer att bilda en form av "de-facto" standard på marknaden (i alla fall i betydelsen att de flesta kataloger kommer att kunna ta emot och överföra information till en sådan katalog).

Som formell standard på kort sikt kommer ANSI att gälla ytterligare några år. Detta understryks av det faktum att amerikanska myndigheter kräver följsamhet med standarden vid sina uppköp.

Under tiden fortsätter ISO's standardiseringssträvanden inom IRDS-området. Dagsläget är att man diskuterar en "omstart" i form av en IRDS2. Det finns faktiskt många fördelar med en sådan omstart.

ANSI och ISO kommer att närma sig varandra eftersom IRDS2 inte kommer att specificeras med SQL. Däremot kommer man att utgå ifrån ISO's redan godtagna "FRAMEWORK" [ISO 10027].

En annan fördel är att man på så vis även skulle kunna behandla de förslag till förbättringar som har inkommit till ANSI, från både IBM och Digital.

Även om det är långt kvar till en global standard för resurskataloger, så för det pågående arbetet med sig att en samordnad syn uppstår efter hand. Standardiseringsarbetet påverkar produkter och produkterna påverkar i sin tur standardiseringen.

## 6. REFERENSER

- [ISO 10027] Information Resource Dictionary System  
ISO/IEC 10027, Framework
- [CD 10728] Information Resource Dictionary System  
ISO/IEC CD 10728, Services Interface
- [RDF-CASE] CASE, En plattform för 90-talets systemutveckling  
Söderström, Jonsson, Fägersten, Tenevall  
DF Förlag 1990, ISBN 0-903969-42-4
- [BERILD] IRDS:Modeller och modellnivåer  
TRIAD, delprojekt Katalogprinciper  
Rapport nr: 3



## 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 TRIAD-projektet. Seminarier, rapporter och referensgruppsverksamhet är led i den verksamheten.