

EFFEKTIV IT

AFFÄRSKOMMUNIKATION

RAPPORT NR 14 – AUGUSTI 1994

OM ARKITEKTUR FÖR SAMVERKANDE INFORMATIONSSYSTEM

Matts Ahlsén

SVENSKA INSTITUTET FÖR SYSTEMUTVECKLING

SISU

Om arkitektur för samverkande informationssystem

Den här rapporten diskuterar arkitekturer för informationssystem speciellt med avseende på samverkan mellan olika datorstödda informationssystem. Bakgrundsmaterialet är dels de seminarier som hållits på SISU med anknytning till projekten Affärskommunikation och Arvet, konferenser och seminarier, samt synpunkter från olika organisationer.

1. Arkitektur och informationssystem

Att utveckla och sätta samman nya eller existerande datorstödda tillämpningar till samverkande informationssystem kräver ett välutvecklat komponentsynsätt (c.f. objektorientering). Detta bör ingå i ett ramverk som möjliggör avgränsning av teknisk plattform, funktionalitet, gränssnitt och ansvarsområden för de ingående informationssystem. En arkitektur för informationssystem (IS-arkitektur) definierar ett sådant ramverk. En IS-arkitektur kan beskriva hur informationssystem är inbördes organiserade, hur de interagerar och hur de är tekniskt och funktionellt avgränsade i förhållande till en organisations verksamhet och processer.

Syftet med en sådan arkitektur är att kunna identifiera och beskriva existerande såväl som planerade system med avseende på komponenter, funktionalitet, avgränsningar och inbördes beroenden. Detta för att möjliggöra samverkan mellan existerande IS och nya tillämpningar. En arkitekturbeskrivning kan då liknas vid en slags referensmodell som identifierar olika vyer eller projektioner på olika abstraktionsnivåer. Arkitektur kan generellt uppfattas som form och funktion, men termen arkitektur används flitigt i sammanhangen IT och informationssystem med varierande och ibland oklar innebörd. Det ligger ett generellt intresse i att klargöra begreppet arkitektur och att relatera det till metodik, beskrivningsteknik och modeller för informationssystem. Några andra motiv är,

- Systemutveckling blir allt mer komponentorienterad och blir mer av systemsammansättning och konfigurering. En "arkitekturmodell" skall hjälpa oss att förstå och relatera komponenterna i denna process. Omvänt kan en sådan "modell" utgöra en målstruktur för renoverings eller migreringsprocesser, genom att hjälpa oss att identifiera återanvändbara komponenter.

- De flesta informationssystem är inte slutna, de kommunicerar på många olika sätt med omgivningen. Systemens gränser förändras och kan i vissa fall vara oklara. Samverkan och samarbete över såväl verksamhets- som organisationsgränser ställer olika aspekter på systemsamverkan i delvis ny belysning (t ex åtaganden, ansvarsformer, styrning), aspekter som till ringa omfattning stöds av existerande IT-strategier och IS-metoder. En arkitektur bör lyfta fram sådana faktorer och relatera dem till olika struktureringsprinciper för informationssystem.
- Kraven växer på att kunna hantera heterogenitet för att åstadkomma samverkan mellan informationssystem, dvs olikheter som är ett resultat av att system utvecklats på olika plattformar, med olika syften och vid olika tidpunkter. Dagens teknik ger goda möjligheter till ett visst implementerings- och plattformsoberoende, men heterogenitet på tillämpningsnivå kommer dock alltid att finnas. En arkitektur bör t. ex. ge vägledning om vilka standarder som kan användas för att hantera heterogenitet.
- Synen på (datorstödd) information och informationshantering förändras. Att se "informationssystemet" som ett övergripande styrsystem med krav på fullständighet och global konsistens blir allt svårare i miljö där ett informationssystem inte längre är *ett* system, utan snarare en federation av många olika system, som dessutom behöver förändras kontinuerligt. En arkitektur bör klargöra olika struktureringsprinciper för sådana federationer.

Det vi här menar med arkitektur avser inte systemutvecklingsmetodik eller språk och modeller för att identifiera och specificera krav på informationssystem. Arkitektur avser inte själva processen att utveckla system, utan snarare den struktur som kan vara målet för en utvecklingsprocess. Arkitekturen är dock starkt relaterad till och kan vara ett hjälpmedel i analys- och design-situationer, den säger hur "det bör vara" inte "hur man gör", i detta avseende kan den stödja vidareutveckling av informationssystem.

Det ligger nära tillhands att jämföra (byggnads) arkitektur med design och arkitektur för informationssystem. Det generella området designteori [Lundequist 1992] börjar uppmärksammas även av "arkitekter" av informationssystem. Vissa försök har också gjorts att jämföra begrepp från arkitekturteorier inom byggnad med IS design [Olerup 1991].

Forskning kring informationssystemutveckling och arkitektur pågår i Sverige vid Linköpings Universitet inom forskningsgruppen VITS [Goldkuhl et al. 1993], där man bl a empiriskt studerar konsekvenser av olika strategier för strukturering av informationssystem. Vid Chalmers och Göteborgs Universitet har Magoulas och Pessi studerat informationssystemarkitektur [Magoulas and Pessi 1991]. Denna forskning drivs för närvarande vidare på Handelshögskolan i Göteborg. Detta arbete är till stor del baserat på Hugosons utveckling av principer för verksamhetsbaserad systemstrukturering [Hugoson 1990].

Internationellt används begreppet informationssystemarkitektur i forskning kring strategier för planering och införande av IT [Earl 1989], såväl som inom ramen för traditionell systemutveckling. Ett relativt nytt område där arkitekturfrågor spelar en framträdande roll är "Enterprise Integration" [Petrie 1992] där man strävar efter att utveckla metodik för att relatera eller integrera olika modeller av informationssystem och verksamheter.

Samverkan är ett primärt syfte med de flesta informationssystem, de faktorer och konstruktionsprinciper som har betydelse för samverkan måste kunna uttryckas inom ramen för en arkitekturbeskrivning.

2. Samverkande informationssystem

Samverkansmodell

Informationsteknologins tillämpning på informationsförsörjning blir allt mer diversifierad. De flesta organisationer är beroende av en mängd separata informationssystem som stöd för olika delverksamheter, inklusive datorstöd för individers personliga produktivitet. Samtidigt växer kraven på samverkan och informationsutbyte mellan verksamheter, vilket kräver bättre samverkan mellan de lokala systemen i den egna organisationen såväl som med andra organisationers informationssystem. En förbättrad samverkan mellan informationssystem syftar till att underlätta situationer där två eller flera avgränsade verksamhetsstödjande informationssystem, kan utnyttjas för att utföra någon verksamhetsöverskridande funktion (eller process). Datorstödda arbetsflöden (ärendehantering) är ett exempel på samverkan mellan informationssystem där ett antal sammanhängande arbetssteg (ärendet) kan utföras av en mängd i sig fristående informationssystem. En annan form av samverkan krävs för att möjliggöra sökning och sammanställningar över en mängd olika informationskällor.

Existerande metoder för systemutveckling ger relativt dåligt stöd för utveckling av samverkande system, metodsynsätt har färgats av den vid en viss tid existerande tekniken för implementering. Som exempel kan nämnas databasteknikens utveckling under det senaste decenniet som i hög grad format systemutvecklarens och datateknikers tanke- och därmed de metoder och synsätt som idag tillämpas för att bygga datorstödda informationssystem [Sundgren 1992]. Men databasteknikens roll förändrats från att ha varit fokuserad på lagring och strukturering av data, till kommunikation och distribuering av information. Databasen är inte längre primärt "informationssystemets minne" [ibid]. Detta har betydelse för hur vi skall utforma informationssystemarkitekturer i framtiden.

Det är vanligt att systemsamverkan diskuteras utifrån huruvida en eller flera organisationer berörs. Datoriserade arbetsflöden är exempel på teknik som idag företrädesvis tillämpas inom ramen för en organisationen, även om inga principiella hinder finns för att hantera arbetsflöden över organisationsgränser. Begreppet inter-organisatoriska system (IOS) betecknar datorstödda informationssystem som används för att dela resurser mellan två eller flera organisationer [Suomi 1992]. Inter-organisatoriska system har studerats utifrån aspekter som strategisk användning av IT, ekonomiska och sociala effekter, samt olika samverkansformer. I det senare ingår koppling till tillämpningsområdet EDI där olika branschspecifika standarder och transporttjänster, utgör teknik för informationsöverföring inom IOS. För att kunna tala om "samverkan" inom IOS börjar man nu vidga EDI till att även omfatta datorstödda affärsprocedurer. Standardisering inom EDI-området försöker idag etablera konsensus kring samverkansformer för sådana procedurer, vilket kan liknas vid utveckling av en arkitekturmodell.

Många av de överväganden som gäller samverkan i IOS gäller naturligtvis även informationssystem i organisationer som är decentraliserade. Skillnaden är att man för inter-organisatorisk samverkan måste hantera frånvaron av övergripande styrning och ansvar, vilket får förutsättas finnas även i starkt decentraliserade organisationer.

Vad menas med olika samverkansformer? Hur är dessa relaterade till informationssystemarkitektur? Kan vi identifiera krav på arkitekturer utifrån olika samverkansformer? En viktig utgångspunkt är att kunna urskilja en samverkansmodell¹, på en övergripande nivå kan detta vara en beskrivning av hur ett antal avskilda delsystem förväntas interagera. Det väsentliga är då att modellen fokuserar på det externa beteendet i form av gränssytor mellan de ingående delsystemen, och sambanden mellan informationssystem och den verksamhet de avses stödja.

En vanligt förekommande uppdelning skiljer meddelandesamverkan från samverkan genom lagrade data². Där meddelandesamverkan står för överföring av informationsobjekt via överenskomna meddelandetyper mellan tillämpningar i avgränsade informationssystem. Denna typ av samverkan kan i princip vara synkron eller asynkron, och realiserats typiskt med något meddelandehanteringssystem sk. postverk, och eventuellt genom att använda publika standarder (t ex baserade på EDIFACT). Samverkan via lagrade data bygger på möjligheten att kunna definiera den information som är gemensam

¹ Begreppet samverkansmodell har också använts för att beskriva beslutsprocesserna vid utveckling av samverkande system [RRV 1991].

² Registersamverkan eller databassamverkan används även som beteckning.

för hela eller delar av en organisation, och som därmed kan tillgodose informationsbehov hos flera informationssystem i olika verksamheter. Detta kan realiseras genom tillgång till en eller flera databaser. Samverkan via lagrade data innebär inte att enskilda tillämpningar eller informationssystem tillåts "gå in i varandras databaser" på ett okontrollerat sätt, utan förutsätter såväl strikt inkapsling som en väl fungerande data-/informationsadministration. På motsvarande sätt behöver inte meddelandesamverkan innebära fokusering på asynkron el-postliknade kommunikation.

En sådan distinktion bör sättas in i ett sammanhang eller ramverk, där begreppet samverkan kan ses från olika perspektiv. Ur ett användar- och verksamhetsperspektiv kan vi betrakta olika samverkansformer för interaktion mellan delverksamheter, där varje nivå tillför komplexitet.

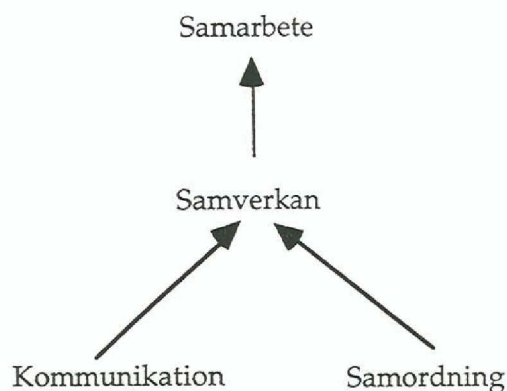


Fig. 1: Exempel på samverkansformer.

På en nivå har vi kommunikation i betydelsen gemensamma begrepp och vokabulär, samt, samordning i form av gemensamma regler för koordinering. Dessa utgör grund för samverkan, vilket innebär gemensamma mål och delad kunskap/information, som i sin tur stödjer samarbete, i betydelsen ansvar och utförande av gemensamma aktiviteter.

För de datorstödda informationssystemen kan vi se samverkan på åtminstone tre olika nivåer av detaljering,

- kooperation: samverkan mellan tillämpningar, där tillämpningarna är komponenter i olika informationssystem inom skilda verksamheter (möjligen i olika organisationer). Tillämpningarna stödjer något gemensamt övergripande mål, t ex realiserat i ett datorstött arbetsflöde eller genom standardiserade EDI-procedurer (scenarios). Dessa kan bygga på informationsförmedling genom lagrade data eller via meddelandeförmedling.

- interoperativitet: att dela funktioner och data, t ex det vi idag kallar "client-server", objektmäklare, transaktioner (konsistens, säkerhet), datadistribution, meddelandeförmedling, EDI-meddelanden, eller delad databas. Tillämpningsgränssnitt (API) mot databas- och transporttjänster.
- kommunikation: protokoll för dataöverföring och mekanismer för anrop. OSI(7), adressering, säkerhet inklusive kryptering och autentifiering.

Informationssystemarkitekturens samverkansmodell skall relatera olika samverkansformer till informationssystemens struktur.

Olika perspektiv på arkitekturer för samverkan

Det finns många olika tillämpnings- och teknikområden där resonemang kring arkitektur är en väsentlig del för beskrivningen och förståelsen av teknikens användning i olika sammanhang. Vi tar upp några sådana områden här i form av olika perspektiv på arkitektur. Dessa är delvis överlappande, och skall inte ses som en klassificering av olika arkitekturer.

I ett verksamhets-informationssystemperspektiv kan man med arkitektur avse strukturen hos ett enskilt system, eller en struktur av ett flertal relaterade informationssystem. Den forskning som refereras ovan omfattar detta perspektiv. Förutom de aspekter som nämnts tidigare, ingår här frågeställningar om huruvida är verksamheten och användare är en del av arkitekturen, och hur verksamhet relateras till informationssystem och omvänt. Samverkan får här betydelsen av olika beroenden mellan avgränsade informationssystem.

I databasperspektivet så handlar arkitektur främst om olika sätt att organisera schemastrukturer (schema-arkitektur), semantisk heterogenitet mellan informations- och datamodeller och schemaintegrering. De senare har att göra med generell modelleringsproblematik, men har kommit att fokuseras under senare år som ett resultat av olika distribuerade/decentraliserade databasarkitekturer. En mängd olika databasarkitekturer har föreslagits inkluderande konventionella centraliserade databaser, distribuerade databaser, multi-databaser och federerade databaser. Samverkan har här betydelsen delad åtkomst till lokala eller autonoma databaser, i form av information och dennas beskrivning (schemata). Arkitektur för federerade databaser [Sheth and Larson 1990] gäller i många avseende också samverkande informationssystem generellt.

Ett tredje perspektiv kan kallas plattformsperspektivet, och är kanske den bild som många associerar med arkitektur och informationssystem. Här handlar arkitektur om komponenter och funktioner inom ramen för olika infrastrukturer eller plattformar, t ex baserade på de facto (industri) och de jure standarder. En term som ofta används för dessa arkitekturer är "middleware".

Exempel på arkitekturer är OSFs "Distributed Computing Environment" (DCE) och objektmäklare av typen OMGs CORBA. Den senare beskrivs ibland som ett ramverk för operationsinvokering med en objektorienterad gränssnittsyntax. Det finns idag en mängd implementeringar baserade på CORBA. Samverkan i detta perspektiv handlar i huvudsak om interoperativitet som det nämnts ovan.

Det fjärde perspektivet kallar vi för agentperspektivet. Agentsynsättet lyfter fram koordinering som det centrala i samverkande system. Visionen här är en informationssystemarkitektur som utgörs av "samverkande intelligenta agenter i nätverk" [Laufman 1994; Papazoglou et al. 1992]. Agent har här betydelsen av ett välavgränsat fristående delsystem med specialiserad funktionalitet, som erbjuder eller utnyttjar funktionalitet till /i andra agenter. "Intelligens" bör här förstås som avancerad funktionalitet. Agentsynsättet kommer huvudsakligen från området artificiell intelligens (AI) och har senare vidareutvecklats under rubriken DAI ("Distributed Artificial Intelligence").

Vi utgår i den här rapporten huvudsak från ett informationssystemperspektiv, men med viss inriktning mot databasperspektivet.

3. Informationssystemarkitektur

Syftet med arkitektur

I sammanhanget informationssystem kan vi beskriva arkitekturer med avseende på ett "system" eller med avseende på flera relaterade "system". Intuitivt betyder "system" här en samling tillämpningar och databaser som stödjer en avgränsad verksamhetsfunktion. Begrepp som "tillämpning", "system", "databas" etc är med nödvändighet allmänna, och deras specifika innebörd är beroende av sammanhang. Ett allmänt syfte med området IS-arkitektur är att sätta in denna typ av begrepp i ett sammanhang för att möjliggöra en mer precis diskussion av informationssystemets uppbyggnad och gränssnitt mot omgivningen.

Systematiska studier av arkitekturbegreppet har gjorts av Magoulas & Pessi [Magoulas and Pessi 1991] som identifierat två olika användningar av termen arkitektur. I ett systemteoretiskt perspektiv där arkitekturen t ex representeras av en referensmodell för vägledning vid strukturering av något system. I den andra användningen av termen, skall IS-arkitekturen underlätta planering och strukturering av en eller flera organisationers samlade informationssystem.

Det finns ett stort antal försök till definition av begreppet informationssystem i litteraturen. Här använder vi begreppet i en intuitiv betydelse för att referera till en mängd helt eller delvis datorstödda funktioner för informationshantering i någon avgränsad verksamhet. Ett syfte med arkitektur är just att möjliggöra avgränsning genom att tydliggöra olika beroenden. I detta avseende blir arkitekturen också ett kommunikationsinstrument eller en förklaringsmodell som kan användas för att tydliggöra de olika relationer som existerar mellan informationssystem och verksamhet.

Vi kan även se IS arkitektur som ett slags förändringsinstrument; med en arkitekturmodell skall man då kunna identifiera komponenter och beroenden i gamla ("legacy") system i en migreringsprocess [Brodie & Stonebraker 1993].

Vad omfattar arkitektur för IS och hur presenteras den? En vanligt förekommande utgångspunkt är att betrakta arkitekturens domän utifrån ett antal olika projektioner och abstraktionsnivåer av system och verksamhet.

Abstraktionsnivåer (specifikationer)	Projektioner				
	Verksamhet	Information	Funktion	Design	Teknologi
Strategisk					
Konceptuell					
Logisk					
Fysisk					

Fig. 2: Projektioner och abstraktionsnivåer för arkitektur.

I matrisen i Fig. 2 representerar elementen olika delområden för en arkitektur, där elementet Konceptuell/Information i matrisen kan representera arkitekturens representation av begreppsmodeller för verksamheten. Till detta skall läggas definitioner av begrepp, deras inbördes relationer i form av en taxonomi, struktureringsprinciper och representationer (modeller) som används för beskriva arkitekturen. Matrisformen som utgångspunkt behöver förvisso inte innebära att en viss arkitektur beskrivs i form av en tvådimensionell matris.

Bland existerande arkitekturmodeller som baseras på ett sådant nivå/projektionstänkande kan nämnas Zachmans ISA [Sowa and Zachman 1992] och i viss mån ANSA [Iggulden 1992]. ISOs förslag till ODP- ("Open Distributed Processing") standard har också en liknande utgångspunkt.

Att systematiskt studera olika strategier eller principer som leder till olika sätt att strukturera informationssystem, kan ge en bild av egenskaper och faktorer som karakteriserar olika IS-arkitekturer. Goldkuhl et al. [Goldkuhl, Pettersson et al. 1993] föreslår ett antal faktorer för analys och jämförelse av olika informationssystemarkitekturer:

- Arkitekturprincip: t ex princip för att organisera verksamhetens informationsförsörjning.
- Syn på data: t ex hur information betraktas som resurs.
- Informationssamverkan: t ex meddelandesamverkan eller samverkan via lagrad data.
- Ansvar för data, informationssystem och arkitektur: t ex centralisering/decentralisering av styrning.
- Tillgänglighet/spridning av data i verksamheten: t ex antaganden om global resp verksamhetslokal tillgänglighet och spridning.
- Förändring – stabilitet: antaganden om förändringsbenägenhet hos data och informationsbehov.
- Organisationsstruktur i relation till informationssystem: beroendeförhållanden.
- Organisations-/verksamhetsförändring i relation till IS: antaganden om hur IS påverkas vid förändringar av organisation och verksamhet.
- Anskaffning av data: t ex antaganden om informationskällor.
- Samband mellan databaser och applikationer: t ex databeroende, avgränsningsprinciper för IS.
- Datastrukturering/datalagring: t ex antagande om komplexitet och redundans
- Databeskrivning: t ex principer för dataadministration och regler för datautbyte
- Driftsmiljöer: t ex oberoende och heterogenitet
- Sårbarhet: konsekvenser

Dessa faktorer har generaliserats fram genom analys av två olika strategier för strukturering av informationssystem.

Konsekvenser av olika arkitektursynsätt

Konsekvenser och resultat av att tillämpa olika principer för strukturering av informationssystem har studerats empiriskt [Goldkuhl 1994][Pettersson 1994b] utifrån de analysfaktorer som nämnts ovan. Med IS-arkitektur avser man här en organisations samlade informationssystem och deras inbördes förhållanden och relation till organisationens (del-) verksamheter.

I dessa undersökningar har man utgått ifrån två olika struktureringsprinciper för informationssystem i form av verksamhetsbaserad resp informationsbaserad (datadriven) strukturering. En av de mest framträdande skillnaderna mellan dessa strategier och de resulterande arkitekturerna, är samverkansmodellen (faktorn Informationssamverkan). Målet har varit att studera praktiska realiseringar av arkitekturer och att analysera dessa i förhållande till principer och avsedda mål. Man har speciellt studerat dessa två strategiers syn på och konsekvenser för ansvarsfördelning och användarinflytande [Pettersson 1994a].

Den informationsbaserade strategin företräds här av IRM och därmed likställda dataadministrativa (eller datadrivna) strategier [ibid.]. Samverkansmodellen baseras på samverkan via lagrade data, d v s databaser. Man kan konstatera att den informationsbaserade strategin bygger på att betrakta information som central, och i många fall global resurs, i verksamheten. Detta leder till ett beroende av att informationsmängder och datadefinitioner är relativt stabila i förhållande till verksamhet och tillämpningar. Denna ansats förutsätter i många fall också en väl utvecklad organisation för data/informationsadministration som en del av informationssystemarkitekturen. En grundläggande strävan i denna strategi är också att göra informationssystemen så oberoende av organisations- och verksamhetsstruktur som möjligt. Data- och informationsmodellering fokuseras i denna strategi, det är dock inte modellering per se som skall kritiseras, utan snarare den centrala roll datamodellen får för struktureringen av informationssystemen samt en strävan att skapa stabila data-definitioner.

Flera företag har idag erfarenheter av informationssystemarkitektur som stödjer decentralisering och samverkan mellan informationssystem. Den i Sverige mest etablerade och definierade ansatsen för decentraliserade arkitekturer torde vara Verksamhetsbaserad Systemstrukturering (VBS) [Hugoson 1990].

VBS baseras på flera grundläggande principer av vilka den mest fundamentala torde vara: att informationssystemen skall samverka på samma grunder som verksamhetens funktioner samverkar. Den centrala frågeställningen som lyfts fram är hur ansvaret för en organisations informationssystem skall fördelas, decentralisering och avgränsning av delsystem blir här centralt. Samverkansmodellen består av en uppsättning riktlinjer för hur meddelandesamverkan skall realiseras.

Meddelandesamverkan baseras här på s k sambandsinformation som definieras utifrån hela verksamhetens perspektiv. Den totala mängden sambandsinformation (kallad sambandsstrukturen) skulle då i princip kunna ses som en för verksamheten global informationsmodell. Dock påpekas i VBS att de ingående "informationstyperna" inte skall vara inbördes relaterade. Det senare torde vara en restriktion som är ämnad att åstadkomma oberoende i

realiseringen av sambandsinformation i termer av meddelandestrukturer. Den verksamhetsbaserade strategin tonar ned behovet av att integrera olika (informationssystem-) modeller från olika informationssystem (c.f. "Enterprise Integration", schema integrering). För att utveckla och underhålla sambandsinformation föreskrivs dock en form av globala begreppsmodeller, samverkanskataloger. Denna typ av kataloger kräver också någon form av central data- eller informationsadministration.

Noteras bör att dessa två exempel på strategier till viss del har samma bevekelsegrunder i att möjliggöra förändring i verksamhet och i informationssystem, men på helt olika sätt. Man kan ifrågasätta det riktiga i en allt för stark fokusering på information som globalt tillgänglig resurs, i allmänhet på grund av att information förefaller att sakna de flesta av de egenskaper vi normalt tillskriver resurser. Å andra sidan så är den form av decentralisering som verksamhetsbaserad företräder, inte önskvärd eller ens möjlig i många organisationer.

Ett annat synsätt på vad informationssystemarkitektur kan vara ges av de många olika struktureringsmodeller som alla på ett relativt likartat sätt försöker identifiera generella komponenter och konstruktionsprinciper för ett eller flera informationssystemens uppbyggnad, inklusive samverkan med omvärlden. Vi kan kalla detta för en konstruktionsorienterad arkitektursyn.

Baserat på 25 års erfarenhet av utveckling av datorstödda verksamhetssystem har Bell-Core [Bellcore 1992] utvecklat en arkitektur kallad OSCA. Det övergripande målet är att åstadkomma interoperativitet mellan system och mellan tillämpningar, möjligen i olika datormiljöer. Utgångspunkten är separering av data, processer och verksamhetsfunktioner samt en strikt modularisering av IS-strukturen.

Arkitekturmodellen beskrivs i termer av skikt ("Layers"), komponenter ("Building Blocks"), gränssnittsdefinitioner ("Contracts"), och en antal konstruktionsprinciper för komponenter och gränssnitt. En komponent är en samling programvaruartefakter som tillsammans implementerar en för verksamheten meningsfull funktion. Ett "kontrakt" representerar ett gränssnitt mot en uppsättning (verksamhets-) funktioner i en komponent, och garanterar åtkomst till dessa funktioner för andra komponenter. Konstruktionsprinciperna syftar till att åstadkomma en hög grad av autonomi mellan komponenter, t ex genom inkapsling och abstraktion via kontrakt. Begreppen koppling och koherens används för att referera till beroende mellan komponenter, resp till funktioners samhörighet inom en komponent.

Arkitekturens "skikt" implicerar inte någon hierarkisk skiktning, utan kan snarare ses som domän inom vilka olika typer av funktionalitet grupperas.

Dataskikt ("Data Layer"): data inbegriper både struktur och beteende och man skiljer på global, privat och redundant data. Globala data förvaltas av någon komponent och skall kunna delas globalt. Privata data är lokala för, och inkapslas av, en komponent, redundanta data är replikerade data med vidhängande konsistenskrav. Data i detta skikt avses vara härledda från någon global informationsmodell (denna är dock ej en del av arkitekturen).

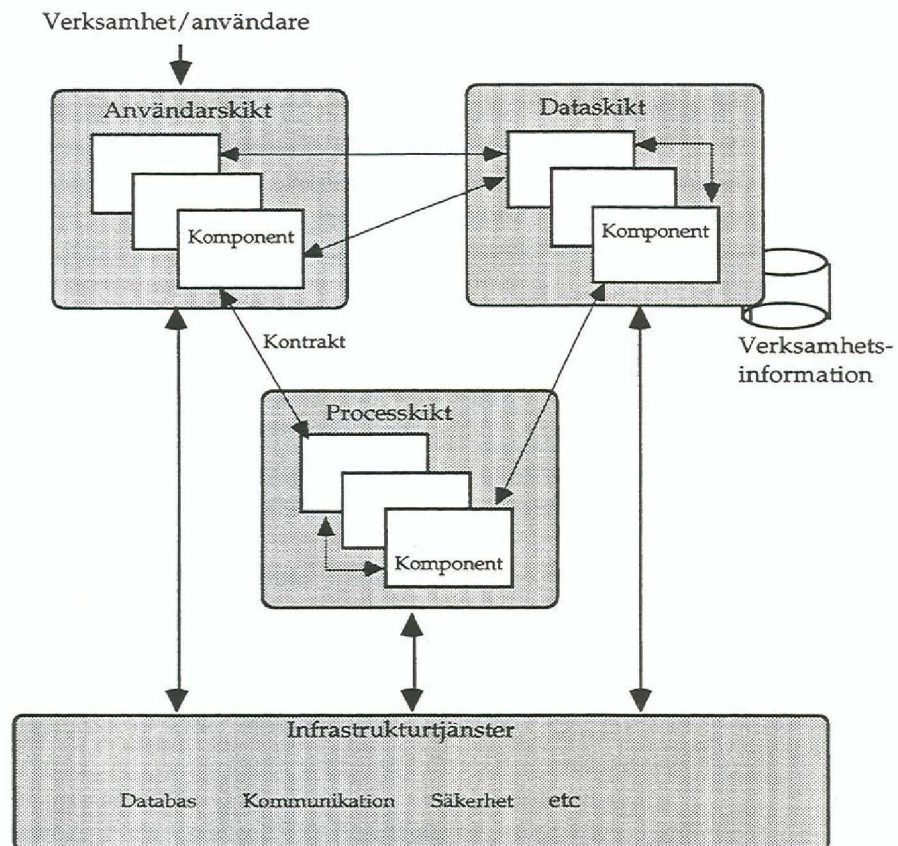


Fig. 3: Arkitekturkomponenter enligt OSCA.

Processkikt ("Processing Layer"): implementerar verksamhetsfunktioner i form av bearbetningslogik som härletts från verksamhetsregler. Användar/verksamhetsskikt ("User Layer"): utgör gräns mot verksamhet och användare. Komponenterna i detta skikt ger användar- och verksamhetsanpassad åtkomst till komponenter i data- eller processkikten.

Infrastrukturdelen av OSCA har rollen av en plattform (eller "middleware") med verksamhetsoberoende funktioner för ex vis kommunikation över system och verksamhetsgränser.

OSCA-arkitekturen ger antydningar om hur information och funktioner som realiserar genom arkitekturen, skall betraktas i förhållande till verksamheten som helhet. Den innehåller dock inte något fullständigt informations/verksamhetsperspektiv på arkitektur. Andra exempel på denna typ av arkitekturer är de olika "applikationsarkitekturer" som många datorleverantörer utvecklat.

Arkitekturklasser

Som utgångspunkt för en mer detaljerad taxonomi, kan vi tänka oss tre huvudsakliga klasser av arkitekturer bestående av konstruktionsorienterade, informationsbaserade och federerade arkitekturer³. Denna klassificering är baserad på arkitekturemas principiella syn på strukturering av informationssystem. En grundtanke är att alla arkitekturklasser skall ha en samverkansmodell som en egenskap. Modellen i Fig.4 skall ses som en uttömmande men möjligen överlappande specialisering av de olika klasserna.

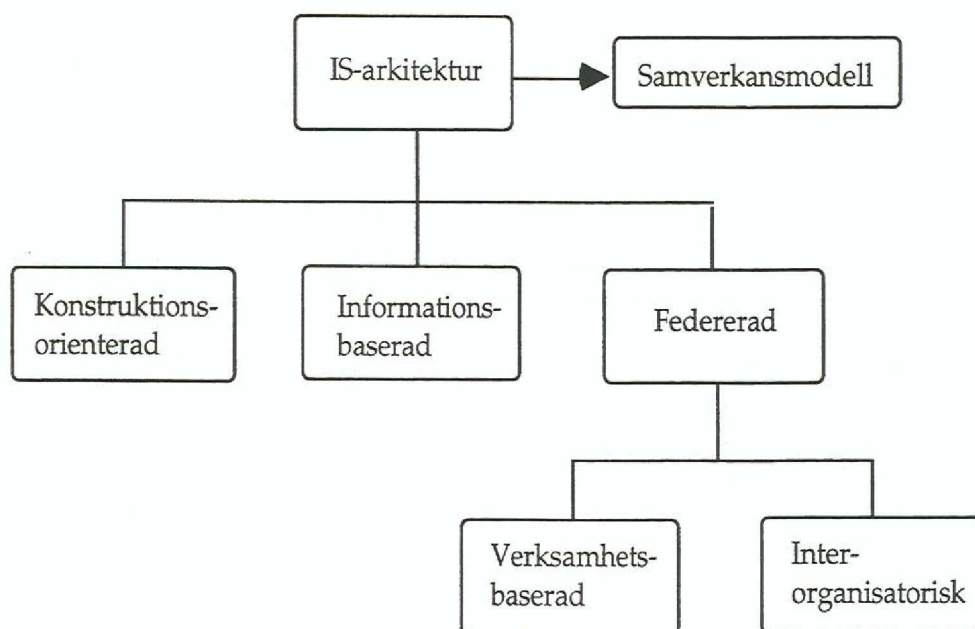


Fig.4: Exempel på arkitekturklasser.

³ I [Magoulas and Pessi 1991] skiljer man på informations-, verksamhets- och interorganisatorisk arkitektur.

En konstruktionsorienterad arkitektur kan i princip realisera målarkitekturer (system) som har egenskaper som också återfinns i informationsbaserade eller federerade system, d v s informationssystem som är kooperativa. Den främsta skillnaden mellan informationsbaserad och federerad arkitektur ligger i samverkansmodellen i form av informationsutbyte. De verksamhetsbaserade och den inter-organisatoriska arkitekturerna är specialiseringar av en federerad arkitektursyn, där en väsentlig skillnad är hur övergripande ansvars eller ledningsfunktioner kan realiseras. Detta har att göra med frågor kring autonomi hos de ingående IS-komponenterna i arkitekturen.

4. Faktorer i samverkansmodellen

Nedan sammanfattas några av de väsentligaste faktorerna som samverkansmodellen för en federerad arkitektur bör vara baserad på.

Avgränsning

Arkitekturen skall klargöra de kriterier som kan användas för avgränsning, vilket tidigare nämnts som ett huvudsyfte.

I en verksamhetsbaserad arkitektur som VBS framhävs ansvarsområden som en fundamental avgränsningsprincip. Avgränsningsproblematiken ses ej där som en teknisk fråga, utan snarare som en utbildningsprocess mellan organisation/funktion och informationssystem. I konstruktionsorienterade arkitekturer, som OSCA, används istället ofta något struktureringsbegrepp för avgränsning (såsom skikt), vilket gör avgränsning till i huvudsak ett teknisk problem.

Avgränsning har också betydelse för att åstadkomma inkapsling av existerande system. Idag är det t ex vanligt att tillämpa komponentteknik för att skapa gränssnitt mot existerande system, vilka sedan successivt kan migreras över till nya arkitekturer. Detta åstadkoms genom inkapsling ("wrapping") av ex vis funktioner, databaser eller hela informationssystem.

Olika former av autonomi

Oavsett princip så handlar avgränsning om att åstadkomma oberoende mellan informationssystem. Med oberoende och därmed autonomi menar vi frånvaron av beroenden som försvårar tillgänglighet, (vidare) utveckling och drift av ett informationssystem i dess lokala miljö.

I det generella fallet kan vi tala om tre former av autonomi med avseende på ett informationssystem:

- design eller konstruktionsautonomi: frihet att välja teknik och principer för realisering av plattformar och tillämpningar, datamodeller och representationsformer etc.
- kommunikationsautonomi: frihet att välja kommunikations- och samverkansform, t ex avseende tidskrav.
- drifts eller exekveringsautonomi: drift och tillgänglighet skall inte kunna styras eller påverkas av andra system.

Dessa motsvarar i stort de olika former av oberoenden som identifieras inom VBS (funktionellt, tidsmässigt, tekniskt och utvecklingsoberoende).

Heterogenitet

Olika former av autonomi leder till olika former av heterogenitet. De svåraste formerna är de som är en konsekvens av designautonomi, och som resulterar i olikheter med avseende på plattformar, implementeringar och semantiken hos data. Det senare kallas ofta semantisk heterogenitet, och är ett område som tilldragit sig stort intresse från forskare inom databasområdet och konceptuell modellering under många år [Johannesson 1993].

Ett sätt att till viss del hantera semantisk heterogenitet är via samverkanskataloger, som bl a förespråkas inom VBS. En sådan katalog är dock alltid svår att utveckla och förändra, p g a att den är global i någon mening. Det som blir kritiskt i en federerad arkitektur är t ex "djupet" i katalogen (hur långt den skall vara specialiserad från början), hur innehållet i katalogen används för att utforma lokala gränssnitt mellan informationssystem, hur förändringar i katalogen skall påverka existerande gränssnitt och kontrakt etc.

I en federerad arkitektur bör dock samverkanskatalogen vara relativt "grund" (endast innehålla de mest generella begreppen/sambanden), vilka sedan används genom lokala specialiseringar inom ramen för kontrakt. Problemen kring heterogenitet och kataloger är också relaterade till schemaarkitektur i multi-/federerade databaser.

En viktig uppgift för arkitektursens samverkansmodell med avseende på heterogenitet, är att den också kan användas för att identifiera de möjliga standarder som kan överbrygga vissa former av heterogenitet.

Informationsutbyte

I ljuset av oberoende och avgränsning mellan enskilda informationssystem, så är möjligheter till effektivt och säkert informationsutbyte kritiskt. En viktig del är också dynamiken i informationsutbytet, d v s protokollaspekter, dessa diskuteras nedan under begreppet koordinering.

I princip kan informationsutbyte generaliseras till att handla om utbyte eller delande av "tjänster", d v s en tjänst kan realiseras av ett antal meddelandeväxlingar. En fördel med detta är att man frigör sig något från meddelandesynsättet som samverkansform, som i vissa fall kan vara begränsande. Det finns en risk att kommunikationsaspekter överbetonas; man "skickar och tar emot data", asynkronitet, flata strukturer etc (c.f. EDIFACT). Som alternativ kan gränssnittsbegreppet användas för att specificera sambandsinformation i form av tjänster. Gränssnittsbegreppet kan vara bättre lämpat för att uttrycka samband på olika abstraktionsnivåer. Meddelandetyper är en form av gränssnitt, funktioner och tjänster är andra. I VBS ges sambandsinformationen med avsikt en klar meddelandekaraktär genom att den förutsätts ingå i en sändar-mottagar relation (1:M). Man skiljer på olika typer av överföringsprinciper: "planerad(successiv)", "begärd" och "fråga-svar". Dessa motsvarar sändarstyrning, mottagarstyrning, resp sporadiska sökningar. Där det för den senare ej finns någon fördefinierad sambandsinformation i form av meddelandetyper.

Vi har tidigare diskuterat meddelandesamverkan vs samverkan via lagrade data, dessa anförs ofta som två skilda samverkansformer som ej bör kombineras. Ingen gemensam datalagring mellan informationssystem för att undvika beroenden kan dock vara en alltför hård restriktion. Gemensam datalagring är en form av resursdelning som inte bör "förbjudas". Med väldefinierade gränssnitt och ev. restriktioner i kombination med inkapsling och behörighet kan beroenden bli kontrollerbara.

Dagens EDI-standarder är baserade på dokument som av tradition använts för att förmedla information mellan organisationer. Standarderna utnyttjar internationellt definierade termer, vilket då gör dem lämpade för inter-organisatoriska system (IOS). Denna tillämpning försvåras dock av den stora mängd av branschvisa specialiseringar av meddelandetyper som utvecklas. EDI likställs ibland med IOS vilket är missvisande eftersom EDI snarare skall ses som en tillämpning för att realisera IOS. Olika EDI-standarder bör däremot utgöra en del av en inter-organisatorisk arkitektur. I princip kan EDI-standarder användas för intra-organisatoriska system, standardernas nuvarande utformning talar dock emot detta.

Reglerad samverkan

Autonomi är ett resultat av strävan att minimera övergripande ansvars- och styrformer, och i vissa fall kan detta medföra frånvaron av global styrning. Detta måste kompenseras med någon annan styrmekanism. Det är vanligt att man då etablerar överenskommelser i form av avtal, kontrakt, för att (i första hand) dokumentera de regler som skall gälla för samverkan mellan informationssystem trots en hög grad av autonomi.

Kontraktprinciper kan tillämpas och realiseras inom flera olika nivåer och vyer i en arkitektur. Det finns tre olika aspekter/användningssätt av begreppet kontrakt i en samverkansmodell: som dokumenterad överenskommelse, som inkapslingsprincip (mot funktioner och data) och som behörighetsmekanism. Kontrakt blir då en del av gränssnittsdefinitionerna i en arkitektur tillsammans med meddelande- och tjänstedefinitioner (eller funktioner).

I VBS definieras all sambandsinformation som ett resultat av överenskommelser mellan berörda (verksamhets-) funktioner. Dessa ses som en slags kontrakt (avtal) vilka reglerar struktur och innehåll för ex vis en mängd meddelandetyper. Kontrakt i OSCA-arkitekturen utgör en inkapslings- och behörighetsmekanism, och i viss mån ett avtal. Motsvarande konstruktion till det vi här kallar kontrakt inom EDI-tillämpningar utgörs av så kallade "interchange agreements". Inom ramen för Öppen EDI försöker man också att formalisera och standardisera dessa som en del av scenarier (se sektion 5).

Ibland diskuteras även "förhandling" i samband med kontraktshantering i informationssystem, detta kan innebära olika principer för hur kontrakt administreras.

Realisering av kontrakt i (federerade) databaser har gjorts där kontrakt utnyttjas för samtliga ovan nämnda aspekter [Ahlsen and Johannesson 1990].

Koordinering

En samverkansmodell måste omfatta principer och tekniker för koordinering, dvs hur olika aktiviteter synkroniseras inbördes och i tiden. I en mycket generell definition kan koordinering sägas handla om att hantera beroenden mellan aktiviteter [Malone and Crowston 1994]. För ett antal informationssystem kan ett sådant beroende utgöras av en delad resurs såsom ett informationslager, som hanteras genom en delad databas, där koordinering kan innebära att maximera tillgång med bibehållen konsistens. En annan typ av beroende kan vara sekvensieringen av stegen i ett datorstött arbetsflöde som omfattar flera informationssystem, och där koordinering bl a innebär att hålla reda på dessa olika parter tillstånd.

Noteras bör att när vi pratar om autonomi ovan så handlar det främst om frånvaron av icke-önskvärda beroenden. När vi här diskuterar koordinering så avser detta hantering av de önskvärda eller nödvändiga beroenden som är en konsekvens av samverkan med andra informationssystem.

Koordinering i någon form finns i alla informationssystem men är inte alltid tydliggjord i systemens olika abstraktionsnivåer och gränssnitt. Ofta är det så att god (fungerande) koordinering ej heller märks. En av egenskaperna hos vårt s.k. systemarv är att de existerande informationssystemen ofta "samverkar" på många olika sätt med användare, andra system, och andra organisationers informationssystem, problemet är att de inte gör det efter någon klar samverkansmodell med vars hjälp olika beroenden och därmed koordineringskrav, kan identifieras (t ex i migreringssituationer eller då delsystem skall bytas ut).

I en inter-organisatorisk arkitektur blir koordinering svårare eftersom det inte alltid är realistiskt, önskvärt eller ens möjligt att förutsätta en övergripande (global) styrning och därmed kontroll, av de aktiviteter som interagerar.

De regler som skall gälla för koordinering beskrivs i en modell (t ex med en tillståndorienterad notation) i ett samverkanskontrakt, ex vis för sekvensiering av en mängd meddelanden.

Rollbegreppet används ibland som ett instrument att beskriva beroenden och koordinering mellan aktiviteter, där en roll utgör en generell beskrivning som kan associeras till ex.vis. individer, funktioner eller informationssystem. Detta försöker man t ex göra inom Öppen EDI.

En viktig aspekt av koordinering hänför sig till hantering av fel- och undantagssituationer. I dagens EDI-tillämpningar utgör detta ibland ett stort problem, eftersom de flesta EDI-standarder inte omfattar sådant stöd.

Koordinering börjar ses som ett (tvärvetenskapligt) FoU-område [Malone and Crowston 1994] som spänner över informationssystem, organisationsteori, ekonomi och sociala och biologiska system.

5. Standardisering

Standardisering handlar till stor del om att etablera konsensus kring vad som är möjligt att standardisera. För interoperativitet är standardiseringssträvandena omfattande och har lett till relativt väletablerade de facto och de jure standarder för bl a plattformar (CORBA, DCE m.fl.), databasprotokoll (t ex DRDA, RDA) och meddelandestandarder baserade på EDIFACT. Vissa försök görs nu att också skapa standarder för IS-arkitektur, vi nämner kort två sådana aktiviteter nedan.

Öppen EDI

Referensmodellen för Öppen EDI är ett exempel på en inter-organisatorisk arkitektur med fokus på utnyttjande av publika standarder. Referensmodellen lägger speciell tonvikt på det grundläggande begreppet "scenarios", vilket i detta sammanhang används för att karakterisera datorstödda affärsprocedurer baserade på EDI-standarder. Definition och beskrivning av scenarios representerar samverkansmodellen i denna arkitektur, och ger bl a stöd för automati, koordinering samt meddelandesamverkan. En annan del av modellen omfattar den plattform som skall tillhandahålla generella tjänster för interoperativitet. Referensmodellen och alternativa beskrivningssätt för scenarios har tidigare presenterats [Ahlsén et al. 1994].

Arkitektur för informationssystem i sjukvården

"Healthcare Information Systems Architecture" är ett standardiseringsprojekt som har påbörjats under 1994. Projektet drivs och finansieras av det europeiska standardiseringsorganet CEN inom området medicinsk informatik (TC251 Medical Informatics). Projektet skall utveckla ett förslag till europeisk standard för informationssystemarkitektur med avseende på IT inom hälso- och sjukvården. Samverkan utgör här ett väsentligt problemområde. Utgångspunkten är bl a ISOs ODP förslag.

6. Arkitektur för nya tillämpningar

De senaste årens utveckling inom IS/IT-området har lett till en viss insikt om behovet av metod och teknik för att hantera system av system eller federationer av delsystem.

Inom databasområdet lades stor möda ned på forskning kring distribuerade databaser t ex för att hantera alla problem kring globala konsistenskrav. Sedan

kom en svängning mot decentraliserade samverkande databaser, eller multidatabaser och federerade databaser, där man släpper på konsistenskraven men tvingas tackla problemen med semantisk heterogenitet. I dagens databasprodukter åstadkoms dock ofta distribution genom replikering av distribuerade data, med globala restriktioner som följd. Man kan också ana ett visst nyvaknat intresse för repository-teknik, t ex genom Microsofts utveckling av sin "Meta Data Manager". En annan trend är att försöka utnyttja plattformsteknik av typen objektmäklare med distributionstransparens, för att skapa globala objektmodeller ("object spaces"). Ovanstående utveckling tenderar att leda tillbaka till centralisering och integrering.

BPR-trenden [Hällström 1993] och fokus på verksamhetsövergripande processer, har därvidlag aktualiserat såväl behov som betydelsen av informationssystemarkitektur som fokuserar samverkan och decentralisering. Det finns även en möjlighet att "bättre" IS-arkitektur kan leda till nya tillämpningar.

Att t ex möjliggöra för mindre företag att utnyttja IT som samverkansplattform kan leda till nya tillämpningar som omfattar mer än konventionell data-kommunikation, som t ex att snabbt kunna bilda konsortier för upphandling eller budgivning i konkurrens med större företag. Ett exempel på en sådan tillämpning är s k IT-baserade arbetsnät.

Ett IT-baserat arbetsnät kopplar med hjälp av olika former av informations- och kommunikationsteknologi, samman ett antal organisationer och utnyttjas under en viss tid för att nå ett gemensamt mål (t ex genomförande av ett projekt, etablerande av ett konsortium). Sådana nät kan omfatta olika typer av organisationer med olika roller och kan vara geografisk distribuerat. Arbetsnätet är ett logiskt nät som kan baseras på olika kommunikationssystem och nättjänster, och består ett antal självständiga parter, eller noder, som kan utgöras av organisationer, individer eller andra informationssystem.

Samverkansmodellen som stöd för informationsdelning och koordinering av arbetsprocesser blir en viktig bas i sådana arbetsnät. Aspekter på koordinering är exempelvis: definition av roller, noders autonomi, åtaganden och rättigheter, samt principer för elektronisk kontraktering. Aspekter på informationsdelning inbegriper sändar- respektive mottagarstyrd informationsförsörjning, utnyttjande av meddelandesystem, informationsbaser eller hybrider av dessa.

Detta är ett exempel bland många andra tillämpningar där informationssystemarkitektur spelar en viktig roll.

Källor

Konferenser och seminarier

Förutsättningar för framtida systemsamverkan, SISU-seminarium, februari 1994.

CORBA (S. Baker), SISU-seminarium, mars 1994.

Interoperable Information Systems (M. Brodie), SISU-seminarium, april 1994.

Liss' 94, Strukturering av informationssystem, Linköpings universitet, april 1994.

2nd Intl conference on Cooperative Information Systems - CoopIS 94, Toronto, maj 1994.

Litteraturreferenser

Ahlsen, M. and Johannesson, P., *Contracts in Database Federations*, In Proceedings of International Working Conference on Cooperating Knowledge Based Systems, Keele, Springer, 1991 (1990).

Ahlsén, M., Pelkonen, H. and Walseth, S., (1994), *Concepts and Notations for Open-edi Scenarios*, Rep.No: Effektiv IT Rapport Nr 4, February 1994, Swedish Institute for Systems Development.

Bellcore, (1992), *The Bellcore OSCA Architecture - Baseline Requirements*, Rep.No: TR-STIS-000915, 1992, Bellcore.

Brodie, M. and Stonebraker, M., (1993), *Darwin: On the Incremental Migration of Legacy Systems*, Rep.No: TR-0222-10-92-165, March 1993, GTE Laboratories.

Earl, M. J., (1989), *Management Strategies for Information Management*, Prentice Hall.

Goldkuhl, G., (1994), *Några problem vid datadriven strukturering av informationssystem*, Föredrag presenterat på LISS'94, Linköping maj 1994.

Goldkuhl, G., Pettersson, K. and Eriksson, O., (1993), *Hur studera realisering och konsekvenser av strategibaserade informationssystemarkitekturer*, Research Report Rep.No: LiTH-IDA-R-93-19, Juni 1993, Institutionen för datavetenskap, Univ. i Linköping.

Hugoson, M.-Å., *Verksamhetsbaserad systemstrukturering*, In Proceedings of NordDATA90, Göteborg, (1990).

Hällström, M., (1993), *Business Process Engineering - synsätt, metoder och datorstöd*, Rep.No: TRIAD Rapport N10:10, December 1993, Svenska Institutet för Systemutveckling.

Iggulden, D., *The ANSA Architecture as a Framework for Heterogeneous Execution Environments*, In Proceedings of The First International Conference on Enterprise Integration Modeling,, (1992).

Johannesson, P., (1993), *Schema Integration, Schema Translation, and Interoperability in Federated Information Systems*,

Laufman, S., *The Information Marketplace: The Challenge of Information Commerce*, In Proceedings of 2nd International Conference on Cooperative Information Systems, Toronto, M. Brodie, M. Jarke and M. Papazoglou (Ed.), University of Toronto Press (1994).

Lundequist, J., (1992), *Om designteorins uppkomst*, Nordisk Arkitekturforskning, Vol. 1992, No. 4.

Magoulas, T. and Pessi, K., (1991), *En studie om informationssystemarkitekturer*, Licenciate Thesis Rep.No: February 1991, Inst. f. ADB, Chalmers & Göteborgs Universitet.

Malone, T. W. and Crowston, K., (1994), *The Interdisciplinary Study of Coordination*, ACM Computing Surveys, Vol. 26, No. 1. 87-119.

Olerup, A., (1991), *Design Approaches: A Comparative Study of Information System Design and Architectural Design*, The Computer Journal, Vol. 34, No. 3.

Papazoglou, M. P., Laufmann, S. C. and Sellis, T. K., (1992), *An Organizational Framework for Cooperating Intelligent Information Systems*, International Journal of Intelligent and Cooperative Information Systems, Vol. 1, No. 1.

Petrie, C. J. (Ed.), (1992), *Enterprise Integration Modelling - Proceedings of the First International Conference*, MIT Press,

Pettersson, K., (1994a), *Informationssystemstrukturering, ansvarsfördelning och användarinflytande - En komparativ studie med utgångspunkt i två informationssystemstrategier*, Licentiatavhandling Rep.No: ISRN LIU-SHS-R--4--SE, Juni 1994, Institutionen för Datavetenskap, Linköpings Universitet.

Pettersson, K., (1994b), *Några problem vid verksamhetsbaserad systemstrukturering*, Föredrag presenterat på LISS'94, Linköping maj 1994.

RRV, (1991), *ADB i samverkan*, Rep.No: Dnr 1988:1655, Riksrevisionsverket.

Sheth, A. P. and Larson, J. A., (1990), *Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases*, ACM Computing Surveys, Vol. 22, No. 3.

Sowa, J. F. and Zachman, J. A., (1992), *Extending and formalizing the framework for information systems architecture*, IBM Systems Journal, Vol. 31, No. 3. 590-616.

Sundgren, B., (1992), *Databasorienterad systemutveckling*, Studentlitteratur.

Suomi, R., (1992), *On the concept of inter-organizational information systems*, Journal of Strategic Information Systems, Vol. 1, No. 2. 93-100.

Effektiv IT-rapporter

- Nr 1 Att Mäta Informationsteknologi – Data om IT i Sverige och utomlands,
Mattias Hällström, december 1993. *IT:s Ekonomi & Management*
- Nr 2 Mätning för Effektiv Systemutveckling,
Tapani Kinnula, mars 1994. *Systemutvecklingens Ledtider & Kvalitet*
- Nr 3 Affärsmässiga Scenarier som bakgrund till Reengineering av Informationssystem,
Lars-Åke Johansson, Mats R Gustafsson, mars 1994. *Systemarvet*
- Nr 4 Concepts and Notations for Open-edi Scenarios,
Matts Ahlsén, mars 1994. *Affärskommunikation*
- Nr 5 Business Process Reengineering – vad är det?
Mattias Hällström, april 1994. *Verktyg för Verksamhetsutveckling*
- Nr 6 Managing Information Technology: The Capital Budgeting Process,
Thomas Falk, Nils-Göran Olve, maj 1994. *IT:s Ekonomi & Management*
- Nr 7 Integrerad Systemutveckling – lärdomar från industrin tillämpade på
systemutveckling, Sten-Erik Öhlund, Lars Bergman, maj 1994.
Systemutvecklingens Ledtider & Kvalitet
- Nr 8 Kunskap för hantering av systemarvet – en första systematisering,
Lars-Åke Johansson, Mats R Gustafsson, Roland Dahl, juni 1994. *Systemarvet*
- Nr 9 Metoder för Business Process Reengineering,
Mattias Hällström. *Verktyg för Verksamhetsutveckling*
- Nr 10 GIATs modell för integrering av logiskt underhåll med utveckling av produktsystem
Lars Bergman. *Systemutvecklingens Ledtider & Kvalitet*
- Nr 11 Ekonomisk värdering av IT-satsningar
Nils-Göran Olve. *IT:s Ekonomi & Management*
- Nr 12 IT i årsredovisningen
Nils-Göran Olve. *IT:s Ekonomi & Management*
- Nr 13 Kalkylmodeller för reverse engineering/reengineering-insatser – vad finns idag?
Lars-Åke Johansson, Mats R Gustafsson. *Systemarvet*

*Svenska Institutet för Systemutveckling,
SISU, bedriver forskning, följer utvecklingen och
förmedlar kunskap om informationsteknologins
tillämpning på informationsanvändning
och informationsförsörjning i företag,
myndigheter och andra organisationer.
Institutet verkar inom detta område som
ett opartiskt nationellt kompetenscentrum.*



Electrum 212, 164 40 Kista
Isafjordsgatan 26
Telefon 08-752 16 00 Telefax 08-752 68 00