

EFFEKTIV IT

SYSTEMARVET

RAPPORT NR 3 – MARS 1994

AFFÄRSMÄSSIGA SCENARIER

SOM BAKGRUND TILL

REENGINEERING

AV INFORMATIONSSYSTEM

Lars-Åke Johansson

Mats R Gustafsson

SVENSKA INSTITUTET FÖR SYSTEMUTVECKLING

SISU

Innehåll

Inledning	2
Några begreppsdefinitioner	3
Verksamhetsscenarier	4
Exempel på ett fall inom ramen för scenarierna	15
Summering	17
Verksamhetsscenarier och åtgärder som kan bli aktuella i en strategi	18
Referenser	22

Inledning

SISU arbetar tillsammans med parterna i det nystartade Effektiv-IT-programmet med att systematisera kunskap kring hantering och förnyelse av systemarvet, med fokus på områden som intressenterna anser är speciellt relevanta och som har stor ekonomisk betydelse.

Det finns kunskaper inom området att hämta främst internationellt. De står att finna under en mängd rubriker och delområden. Till exempel metodik för reverse engineering och reengineering, remodellering och verktyg. Det kan gälla arbetsformer i projekt och strategiuppbyggnad för att hantera existerande system och deras migrering mot nya strukturer eller programförståelse, ekonomiska aspekter på reengineering av informationssystem etc.

Vad som starkt betonas i denna skrift är att insatser av typen reverse engineering och reengineering av informationssystem är mer eller mindre meningslösa om de inte inriktas och motiveras från ett verksamhetsmässigt perspektiv. Det betyder att det måste finnas verksamhetsmässiga orsaker till insatser av denna typ.

Det är förmågan att identifiera och avgränsa dessa verksamhetsmässiga perspektiv och utforma en lämplig ansats för reverse engineering och reengineering som är avgörande för om insatserna skall lyckas. Det antas i denna skrift att det inte enbart finns ett sätt att gå tillväga i alla situationer, utan att det snarare är fråga om att "plocka ihop" ett tillvägagångssätt baserat på den situation man befinner sig i.

Det är viktigt att för varje specifikt fall kunna utforma en lämplig reengineering-insats och strategi. Det bör göras i förhållande till den affärsmässiga verksamhetssituationen, och till de intentioner som gäller för verksamheten. Systemarvet-projektet kommer under det första året bl a att arbeta med att strukturera och ställa samman beskrivningar av ett antal verksamhetssituationer för vilka man kan diskutera olika strategier för reverse engineering/reengineering-insatser. Det råder brist på sådana beskrivningar, även internationellt.

I denna skrift försöker vi precisera ett antal verksamhetssituationer - eller scenarier - för att skilja ut dem och diskutera dem närmare. Vi ställer frågan: Hur ser dessa scenarier ut och vilka tänkbara typer av insatser krävs för att man skall kunna nå en önskad effekt?

Sammanställningen gör inte anspråk på att vara fullständig utan utgör ett förslag på situationer som kan vara vettiga att utreda närmare. Förslagen är tänkta att diskuteras och skärpas ytterligare längre fram i projektet.

Några begreppsdefinitioner

Vi vill i detta avsnitt definiera ett antal begrepp som i fortsättningen kommer att användas. Det förekommer alternativa definitioner. De vi utgår från i rapporten grundar sig på en översättning av Chikofsky et al, 1990.

Med **systemarvet** menar vi här den relativt stora mängd av system som utvecklades under 70- och 80-talen och som uppfattas vara behäftade med problem. Det kan vara att de passar dåligt för dagens informationssystemkrav från verksamheten, är dåligt beskrivna, utnyttjar gamla systemplattformar, uppfattas som svårhanterade, passar dåligt till en ny verksamhetsstruktur, är dyra att köra eller i övrigt är svåra att anpassa till nya krav.

Med **reverse engineering** avses processen att analysera och undersöka ett system i syfte att identifiera systemets komponenter och deras relationer, samt att skapa representationer av systemet i en annan form eller på en högre abstraktionsnivå.

Med **reengineering**, eller renovering, menar vi här analys och förändring av ett system i syfte att rekonstruera det i en ny förbättrad form samt att därefter implementera systemet i den nya formen.

Man kan också säga att reengineering egentligen innefattar både reverse engineering och forward engineering.

Med **forward engineering (systemutveckling)** menas här den traditionella processen att gå från högnivåabstraktioner och logiska, implementeringsoberoende specifikationer till fysisk implementation av ett system.

Med **systemunderhåll/systemförvaltning** menas här modifiering av ett system efter leverans för att rätta fel, förbättra prestanda eller andra egenskaper, eller anpassning av systemet till en förändrad omgivning.

Med **remodellering** avses skapandet eller återskapandet av beskrivningar (modeller) av system, av de verksamhetsbegrepp vars representationer systemet hanterar eller av den verksamhet som systemet stöder.

Verksamhetsscenarier

Ny kunskap är under framväxt inom området reverse engineering/reengineering. En mängd verktyg, principiella arbetssätt och annan kunskap har utvecklats på senare tid.

Det råder däremot brist på kunskap och anvisningar om hur dessa ansatser skall kunna tillämpas i olika situationer. Framförallt är det ett problem att kunna sätta samman användning av vissa typer av verktyg, olika analyssteg och ett successivt arbetssätt baserat på de verksamhetsmässiga förutsättningarna.

I det följande har vi försökt sätta upp några typscenarier för ett antal verksamhetsmässiga situationer, och diskuterar hur man kan sätta samman olika verktyg och arbetssätt för att få avsedd effekt i verksamheten.

Det är av olika skäl ofta dessa verksamhetsmässiga förutsättningar som i grunden driver och motiverar intresset för systemarvet.

Projektet har vid flera tillfällen, även internationellt, kunnat notera att man har svårt att sätta samman kunskap om hur man skall kunna utveckla lämpliga reverse engineering/reengineering-strategier på basis av olika verksamhetsmässiga situationer.

Ett av skälen till detta är säkert att man inte förmår att explicit göra klart för sig vilken den verksamhetsmässiga situationen egentligen är, samt att ur denna härleda de åtgärder som kan vara lämpliga att vidta.

Scenarierna är framtagna dels från SISU:s intressentorganisationer, dels från fall som vi funnit i litteraturen, i internationella rapporter eller på andra sätt.

Förhållandena i de beskrivna verksamheterna är således den viktigaste anledningen till att reverse engineering-insatser görs, men scenarierna gör inte anspråk på att vara fullständigt beskrivna. De är ämnade att visa att olika situationer kräver olika ansatser i form av reverse engineering/reengineering för att få effekt.

Vi kan ställa upp följande viktiga typer av scenarier:

1. Förändringar och underhåll av informationssystemen kostar för mycket för verksamheten – det måste finnas alternativ.

I scenariot upplevs underhållskostnaderna som för stora och man skapar åtgärder för att gå igenom systemarvet i syfte att försöka sänka kostnaderna. Det är emellertid svårt att få fram information om hur höga dessa kostnader idag är, men det har noterats, bland annat i en SISU-undersökning, att underhållskostnaderna i sig ofta inte upplevs som särskilt betungande om underhållsarbetet i övrigt fortgår väl.

Det är när det förutom underhållskostnaderna också uppkommer problem att få systemet att uppträda som man vill, som man noterar att de upplevs som besvärande.

Detta kan medföra att man inleder studier av vad man kan göra med sitt system för att få ner underhållskostnaderna. Detta kan i sin tur leda till att man förändrar strukturen hos systemet så att man kan utföra underhåll billigare.

Det är dock så att det idag saknas referensramar ifråga om vad som är normal underhållskostnad för en viss typ av system i en viss typ av verksamhet.

En första analys måste innebära att man undersöker vad de höga underhållskostnaderna består i. Är det stora förändringar som genomförs? Består kostnaderna i att systemen är så ostrukturerade och dåligt beskrivna att det tar lång tid att förstå dem och kunna genomföra förändringar? Har de som tidigare underhållit systemen nu helt andra arbetsuppgifter så att det också av den anledningen är svårt för nya personer att sätta sig in i systemen?

Därefter måste man ta reda på om det fortsatta underhållet av systemen kommer att innebära ytterligare stora förändringar. Detta kan vara fallet då verksamheterna kräver att så sker. Till exempel när nya tjänster måste kunna erbjudas mot kund.

Beroende på svaren på ovanstående frågor måste man sätta samman en strategi av åtgärder som leder fram till att underhållskostnaderna kan sänkas.

Om förväntade nya framtida krav är små inom verksamheten, kanske befintlig kod och befintliga databasstrukturer kan förändras och struktureras om, så att de blir lättare att justera och underhålla. Är däremot förväntade ändringskrav inom verksamheten stora, kanske man bör skriva om vissa delar helt, och samtidigt skapa en struktur som blir lättare att bygga ut. Man bör i det fallet vara noggrann med att de nya delarna blir väl beskrivna och dokumenterade, så att de fortsättningsvis kan ändras i enlighet med nya krav.

Allmänt kan också sägas att en analys av underhållskostnadernas storlek måste sättas i relation till någonting. Helst då till liknande verksamheter med samma typ av informationsstöd. Annars kan man riskera att ge sig på saker som har marginalmässig sämre effekt i verksamheten än andra åtgärder.

Åtgärder för att få ned underhållskostnaderna måste kanske samordnas med åtgärder för att också kunna möta framtida krav. Man skall således inte bara sänka kostnaderna i förhållande till de justeringar som görs idag, utan också i förhållande till de verksamhetsmässiga förändringar, och därav framkomna krav, som måste realiseras längre fram.

2. Verksamheten måste förberedas på att olika verksamhetsdelar skall kunna säljas bort och på att företagsdelar kan komma att köpas in.

Inom tillverkande industri har olika delar av verksamheten ganska klara uppgifter i förhållande till den färdiga produkten. Så är det t ex inom bilindustrin. Inom diversifierade företag däremot kan man syssla med produkter tillhörande helt olika branscher, vilket innebär att de olika företagen lever ett relativt självständigt liv.

I den senare gruppen kan man lättare sälja delar av företaget eller koncernen, eftersom de är mer löst sammansatta. Även inom första gruppen lägger man nu mera ut vissa delar av den totala produktionen på entreprenad.

Man kan t ex i ett biltillverkande företag vara inne på att beställa hela motorer med vissa bestämda egenskaper hos en utomstående partner. Det innebär att man i princip kan sälja hela sin motordivision och ha ett beställningsförhållande till denna nu självständiga organisation.

För att kunna genomföra denna typ av förändringar måste informationssystemen ha sådana gränssytor att verksamheterna ägandemässigt och styrningsmässigt kan separeras på ett sådant sätt att de "egna" informationssystemen ändå fungerar tillfredsställande. Informationssystemen skall kunna behålla sitt värde trots förändringarna.

Detta är dock inte alltid så lätt att åstadkomma eftersom informationssystemen själva skapar hinder. Systemen måste då förändras så att det blir möjligt att skapa sådana gränssytor. Detta kan orsaka ganska stora omstruktureringar, beroende på förändringsgrad och på hur utgångssystemen ser ut.

Om ett företag arbetar förhållandevis integrerat i avseende på de delar man vill skilja av, och om informationssystemen dessutom avspeglar denna struktur, kan det bli ganska mycket arbete med att skapa gränssytor där man vill ha dem enligt den nya verksamhetsstrukturen. Det kan vara svårt att, utifrån den verksamhetsmässiga nivån, avgöra exakt var gränsen i systemet går mellan den ena och den andra delen. Arbetet på den verksamhetsmässiga nivån är viktigt för att gränssytorna skall bli reella och grundade i den samverkan och styrning man verkligen vill ha. Det är viktigt att beskriva vad man kommer fram till. Beskrivningssätt för detta är inte vanliga men håller på att växa fram.

Ofta är det lämpligt att definiera avgränsningar genom att definiera vilka meddelanden som behöver utbytas mellan de olika verksamhetsdelarna. Dessa meddelanden kan sedan implementeras genom att systemdelarna skapar och sänder dem. Det blir sedan det mottagande informationssystemets ansvar att ta emot dem och sedan utföra de olika uppgifter som är baserade på att ett visst meddelande mottagits.

Det börjar komma stöd för att införa sådana arkitekturer från olika leverantörer. Vissa större företag har skapat sig egna generella paket som stödjer denna typ av strukturer.

Vissa program måste kanske omstruktureras ordentligt, så att de kan skapa de avsedda meddelandena, och arbeta på ett sådant sätt att de realiserar tankegångarna som finns i dessa arkitekturer.

3. Verksamheten tar beslutet att den skall styras med autonoma verksamhetsdelar. En kontraktstyrd meddelandesamverkan införs.

I scenariot tar en verksamhet beslutet att den skall styras genom uppdelning i självständiga enheter med egna resurser. Varje verksamhetsdel kommunicerar med övriga genom meddelandesamverkan. Denna avgränsning av verksamhetsdelar gör att informationssystem måste delas och strukturer skapas så att man klarar av att kommunicera med meddelanden. Detta kräver att informationssystem kan modifieras så att den nya arkitekturen uppnås på olika plan.

Vid denna struktur får man gå tillväga på ett sätt som liknar det under situation 2. Gränssnitten är viktiga att fastställa. De måste beskrivas. De kontrakt man tar upp i denna situation måste också beskrivas noggrant så att de blir reella i verksamheten. Kontrakten är primärt en verksamhetsangelägenhet och inte en informations-systemaspekt.

Om man har tillgång till ett aktivt modelleringsverktyg kan det vara aktuellt att noggrant beskriva kontrakten och associera dessa till beskrivningar av hur meddelandena exakt är uppbyggda och vad de består av.

Om olika delverksamheter som skall separeras är hårt integrerade, kan det innebära mycket arbete att lägga gränssnitt på lämpliga ställen i informationssystemen. Speciellt om dessa också är hårt integrerade och inte strukturerade med avseende på de olika verksamhetsuppgifter som skall utföras.

Om system som skall delas är äldre och mindre väl strukturerade, kan det ibland vara lämpligt att bygga nytt för någon eller några av de verksamhetsdelar som skall stödjas. Man "mörklägger" samtidigt de delar av informationssystemen som motsvarar verksamhetsdelar som delats av. Denna typ av mörklägning kan vara aktuell när vissa delar varit speciellt utsatta och är nedslitna p g a mycket "patchrättningar". För den "andra delen" använder man relevanta delar av det gamla informationssystemet.

Detta kan innebära att det blir lättare att ta hand om det gamla systemet och att man samtidigt skaffar sig en del som är bättre strukturerad för framtiden, med bättre och mer förändringsvänliga informationssystem.

4. Verksamhetens centrala funktioner bryts ner och ett antal regionala enheter skapas som är duplikat av varandra och som stödjer sin respektive geografiska del. Viss samordning sker centralt.

Vissa verksamheter, bl a myndigheter, genomför en medveten decentralisering. Man skapar relativt självständiga enheter som skall stödja avgränsade geografiska områden. Detta sker från en utgångspunkt där verksamheten har varit relativt centraliserad. Systemen måste omstruktureras så att de blir mer självständiga var för sig. De lokala systemen är i princip duplikat av varandra.

Samordning av de regionala delarna klaras genom att en ny, förminskad central verksamhetsdel skapas, med informationssystem som är anpassade för den nya uppgiften. Samordningen består av att nya styrdirektiv ges till de regionala delarna; i övrigt får dessa lösa sina uppgifter på sitt eget sätt.

Eftersom det gäller stora förändringar av styrningen bör en verksamhetsanalys av lämplig typ inledas för att man skall få klart för sig vad som skall utföras centralt och regionalt. Bland annat måste sambanden mellan den centrala delen och de regionala delarna preciseras. Den ansats för verksamhetsanalys som man använder bör alltså innehålla stöd för detta arbete. Dessutom måste en funktions-/processanalys och en begreppsanalys göras för de två olika "nivåerna" för att klargöra vilka krav som finns mot informationssystemen.

När det gäller de existerande informationssystemen kan dock ofta olika beslut och regler återanvändas i den nya styrstrukturen. Många olika typer av rutiner och besluts-kriterier som tidigare utfördes och användes centralt kommer att utföras i de regionala delarna, med hjälp av informationssystem som dupliceras och som regionerna själva ansvarar för.

När man använder gamla rutiner och verksamhetsregler regionalt, kan det dock vara väsentligt att klart beskriva dessa explicit i förhållande till varje regionalt system. Detta om det blir aktuellt att förändra informationssystemen och deras regler till följd av visst regionalt självbestämmande eller till följd av att nya lokala regler och politiska beslut tillkommer. Detta kan göras med hjälp av modern begreppsmodellering med associerad regelmodelleringsmöjlighet. Detta skall göras i verksamhetsmässiga termer så att berörda personer kan förstå och verifiera modellerna, inklusive regelbeskrivningarna. Det duger inte att beskriva regler i programkod.

Det kan dessutom vara aktuellt att realisera samverkan mellan centrala och regionala delar med hjälp av en meddelandsamverkanstruktur som också avspeglar sig direkt i hur informationssystemen utväxlar datoriserade meddelanden. I detta fall är det dock också viktigt att meddelanden beskrivs separat liksom vilka motsvarigheter de ingående termerna har i de regionala delarnas och i den centrala delens begreppsapparat.

De regionala delarna bör i princip ha ansvar för sina egna beskrivningar av vad som hanteras i de regionala systemen. Speciellt om informationssystemen och de verksamhetsmässiga förutsättningarna skiljer sig åt i större utsträckning.

5. Man vill hitta/skapa ett nytt sätt (skapa en ny plattform) för att förändra och vidareutveckla system.

I vissa fall är det inte de höga underhållskostnaderna i sig och det att man vill hitta ett sätt att sänka dem som är det viktigaste, utan man vill skaffa sig ett helt nytt sätt att förändra och vidareutveckla system. I detta fall tänker man oftast på helt nya system, men man kan ibland även använda de gamla systemen. Dessa kräver dock ganska omfattande omstruktureringar och förbättringar.

Skälet till att man vill ha en sådan flexibel plattform för förändring och vidareutveckling av systemen ligger naturligtvis på det affärsmässiga planet. Det kan t ex vara så att man går in i en konkurrenssituation, som kräver att man måste kunna ändra sina produkter. Det kan gälla förmågan att snabbt kunna skapa allianser med nya samarbetspartners för samarbete kring olika delprodukter etc.

Vilka kriterier kan man då ställa upp för en ny plattform för underhåll och vidareutveckling? För det första måste de existerande systemen vara bra beskrivna. Detta såväl på den konceptuella nivån som avseende hur olika lösningar är gjorda, på databasnivå och i programmoduler etc.

Dessutom måste delarna vara väl strukturerade med väldefinierade gränssytor så att man förstår vad varje del gör. Avgränsningarna av dessa delar är dessutom viktiga för att man skall kunna använda dem på olika ställen i systemen, i takt med att krav ändras.

Delar skall kunna läggas till genom att man använder gränssytor till existerande delar. Nya delar skall vara väl avgränsade och beskrivas väl - inga snabbt påbyggda "fixar".

Avgränsningarna bör kännetecknas dels av att man hanterar konceptuella objekt och deras egenskaper och händelser/operationer, dels av speciella datatekniska operationer som orsakas av att man har en speciell datateknisk miljö. Dessa aspekter är grundläggande vid en uppdelning av existerande system.

Formen för att åstadkomma en ny plattform för förändring och vidareutveckling kan vara att göra en mer sammanhållen CASE-baserad utvecklingsmiljö.

Den nya plattformen kan också kännetecknas av att man bygger upp en miljö med återanvändbara komponenter, som man försöker använda i så hög grad som möjligt, när man förändrar och bygger nya system och delsystem.

De återanvändbara komponenterna bör vara mycket väl beskrivna, dels så att man kan hitta dem om man söker efter komponenter med vissa egenskaper, dels så att man verkligen kan förstå vad en viss komponent gör och vilka premisser den bygger på.

Komponenterna bör dessutom vara tillgängliga i ett verktyg (någon form av repository) så att man lätt kan söka och navigera bland dem.

6. En verksamhet skall bygga utveckling och vidareutveckling med återanvändbara komponenter.

En variant på ovanstående verksamhetsmässiga intention är att verksamheten också uttalar att vidareutveckling och förändring skall ske genom att man skapar och utnyttjar återanvändbara komponenter.

Detta kan naturligtvis ske inom ramen för föregående rubrik, eftersom man vet att det är rationellare ur olika synpunkter. Det är däremot så att utvecklandet av återanvändbara komponenter kräver vissa grundinvesteringar för att man skall nå upp till en sådan nivå att de kan återanvändas.

Dels måste man avgränsa och strukturera komponenterna, dels måste man ha möjligheter att verkligen hitta komponenterna för att kunna göra bedömningen och ta beslutet att återanvända dem.

Man vill alltså legalisera och skapa medvetenhet kring dessa investeringar.

Återanvändning kan för övrigt ske på olika nivåer. Det kan både vara objektdefinitioner och objektoperationer som kan tänkas att återanvändas, men också modeller, systemdelar och hela system.

7. Verksamheten måste skapa nya verksamhetstjänster med existerande informationssystemstruktur, eftersom marknaden kräver detta.

I scenariot utsätts verksamheten för nya krav från marknaden. Nya tjänster måste skapas. Samtidigt är tillgängliga resurser knappa, vilket innebär att nya system som klarar de nya kraven inte utan vidare kan anskaffas. Man måste kunna klara de nya tjänsterna med de system som finns. Man måste på ett eller annat sätt hitta vägar att förändra de existerande systemen, så att de klarar att leverera den information som krävs för att stödja de nya uppgifterna.

Detta kräver att struktureringen av systemet är god så att man kan bygga ut med delar som klarar de nya tjänsterna.

Ett exempel på nya tjänster kan vara att man utöver leverans av en viss produkt också erbjuder kunden att ordna olika tillstånd från myndigheter för användning av produkten. Detta kräver att man samlar in särskilda uppgifter från kunden, vilket måste göras i nya anpassade delrutiner.

Denna situation kräver att man tillämpar en förändringsstrategi där man verkligen har klart för sig vad som finns i de existerande systemen och vilka premisser (verksamhetsregler) man egentligen bygger på. Det kan göras genom lämplig remodellering. Med hjälp av de nya beskrivningarna, speciellt om strukturen förbättrats, kan man lättare gå in i systemen och göra lämpliga utvidgningar.

Det duger inte att man på ett patch-orienterat sätt bara bygger på med delar för att klara de nya uppgifterna för stunden. Då är risken för fel hög och man får svårigheter att senare kunna gå in i systemen och förstå vad som egentligen görs. Man sliter ner systemen.

Dessutom bör de nya begreppen som hanteras i systemutbyggnaderna noggrant beskrivas på en verksamhetsmässig nivå så att berörda personer förstår dem. De nya reglerna som hanteras i informationssystemen bör också beskrivas.

8. Den existerande datorleverantören byter miljöer och verksamheten tvingas flytta alla sina informationssystem till ett eller flera nya operativsystem.

I scenariot moderniserar datorleverantören sina datorsystemmiljöer. Ibland blir man tvungen att ta större strategiska beslut och t ex byta operativsystem för sina maskiner. Detta kan ha mycket stor betydelse för de tillämpande kunderna och deras informationssystem. De kan bli tvungna att göra mycket stora ändringar i sina systemstrukturer.

Ibland har man i en verksamhet kanske kört sina system längre än man ursprungligen tänkt göra och detta gör att leverantören känner sig föranlåten att ha synpunkten att också kunden måste vara beredd att ta en del av kostnaderna.

Man kan argumentera att denna situation inte är någon verksamhetsmässig förutsättning som är intressant när man talar om strategier för reverse engineering/reengineering. Detta är nog sant. Emellertid kan denna situation utgöra en stor restriktion som verksamheten mer eller mindre plötsligt står inför. Speciellt om man inte haft en löpande uppsikt över dessa förändringar av förutsättningarna. Detta kan innebära att verksamheten plötsligt får ta en stor kostnad för att göra dessa förändringar, så att man kan köra sina system på en ny maskinplattform, eller liknande, som leverantören vill föra in.

Det kan vara fråga om att ett operativsystem måste bytas ut mot ett annat, som är något annorlunda och som innebär att systemen inte kan exekvera som förut. Det kan vara att databashanterarleverantören för in nyheter, vilket gör att man måste ändra det sätt, på vilket man tidigare "accessat" databaserna.

Det kan också vara så att stordatorlösningar ersätts med client/server-lösningar. Detta kräver en del ombyggnad av systemen.

Det kan också vara fråga om att hela datorer ersätts med modernare maskiner med nya operativsystem, vilket innebär att applikationer mer radikalt behöver ändras för att överhuvudtaget fungera.

I dessa fall kan det vara väsentligt att förändra systemen på ett riktat sätt, så att man försöker skapa en nivå på applikationerna som är riktad mot att göra dem så miljöberoende som möjligt. Det ökar möjligheterna att flytta dem mellan olika miljöer. Även när man förändrar existerande applikationer bör förändringarna genomföras på ett sådant sätt att detta arkitekturtänkade tillämpas.

I de fall då existerande applikationer fungerar väl och fyller viktiga informationsbehovsuppgifter, kan det bli aktuellt att se till att de bli körbara också i nya miljöer.

Skall applikationer köras under längre tidsperioder bör de också beskrivas så att man vet vad de gör utifrån ett verksamhetsmässigt perspektiv. Behandlingen i olika program kan med fördel uttryckas med hjälp av regelbeskrivning.

9. Verksamheten har fått nya legala krav genom att nya lagar har införts vilket gör att de gamla systemen inte oförändrade kan utnyttjas som förut.

Lagregler ändras, vilket gör att man får problem med gamla system. Det innebär ofta att premisserna på vilka systemen är byggda helt ändras. Regler inbyggda i program måste skäras ut. Andra måste sättas in och anslutas till andra typer av åtgärder. Hela programstrukturen måste ifrågasättas. Andra typer av data som skall lagras måste ändras. I regel är ändringarna så kraftiga att man måste göra en ny analys av vad som skall göras i systemet. Dessa krav måste sedan kopplas till vad som egentligen görs i det nuvarande systemet. Redan det sistnämnda är ofta svårt nog att ta reda på.

Är systemen dessutom dåligt strukturerade kan det bli en mycket besvärlig uppgift att ändra systemen i den grad som krävs.

Exempelvis kan nya lagregler innebära att förutsättningarna för att skapa unik identifiering helt ändras, genom att man inte får använda personnummer som man gjort förut. Möjligheterna att kunna använda personnummer kommer förmodligen att begränsas genom politiska beslut framöver.

Detta kan innebära att förutsättningarna helt ändras för hur man söker och hanterar de "objekt" som finns i systemet. Nya uppgifter kanske måste samlas in för att identifiera personer. Nya behandlingar och nya regler kan bli aktuella i systemet.

Förmodligen är det så att det krävs en mycket god strukturering av ett system för att man skall klara av att förändra det på detta sätt.

Ett annat exempel kan vara att företag som säljer till konsument, förutom moms, måste ta ut en helt ny avgift på sina produkter, till exempel i samband med en ny miljölagstiftning. Detta innebär nya redovisningar och kräver olika grad av förändring av informationssystemen.

Om det rör sig om välstrukturerade informationssystem kan man klara denna typ av ganska radikala förändringar på ett överblickbart sätt. Om däremot informationssystemen är mindre välstrukturerade kan denna typ av förändringar bli svårare att genomföra.

Förändringar bör dock genomföras så att en bättre strukturering genomförs eller upprätthålls. Man bör ta fram tydliga beskrivningar på verksamhetsnivå av de begrepp och regler som hanteras i systemet och i de olika programmen.

En förbättrad beskrivning bör användas då denna typ av förändring troligen kommer att innebära att komplexiteten i vad som hanteras ökar.

10. Marknaden kräver att ny kvalitetsstyrning av processer införs vilket inte kan klaras med de befintliga systemen.

Kvalitetsstyrning börjar bli ett allt viktigare konkurrensmedel. Man skall inte bara utföra de interna processerna i sig, utan man måste också hantera ett antal kvalitetskontrollmoment i processen, för att säkerställa att kvaliteten ligger inom vissa värden.

Man måste dessutom kunna visa för en yttre part att dessa kvalitetshöjningar verkligen utförs, för att försäkra denne om att vissa kvalitetskrav existerar och vidmakthålls.

Detta är nya krav som allt fler organisationer kommer att konfronteras med. Detta i takt med att olika kvalitetsstandarder blir allt vanligare och att trycket ökar från olika kunder på att dessa kvalitetskrav skall vara beaktade i processerna.

Informationssystemen som stödjer processerna måste klara de nya kraven. Kontroller skall utlösas, data skall samlas in och olika rapporter och dokument skall skapas, etc. Olika regler måste implementeras och nya åtgärdsstrukturer skapas, beroende på vad som händer vid olika utfall av kontrollerna.

I många fall kan konsekvenserna och förändringarna för informationssystemen bli ganska stora för att upprätthålla denna kvalitetsstyrning.

Åtgärderna mot existerande system kan ligga på två plan. Dels skall processerna ändras till följd av kvalitetuppföljningsarbetet. Därmed skall även de sammanhängande informationssystemen förändras, så att de nya rutinerna skall kunna upprätthållas. Dels måste man ofta höja kvaliteten i informationssystemen i sig, så att inte brister där orsakar en kvalitetsförsämring i verksamhetsprocesserna.

Det kan innebära att systemen måste beskrivas bättre så att förändringar och rättningar kan genomföras säkrare. Det kan också innebära att man måste strukturera systemen i sig bättre så att möjligheterna att hitta fel ökar, bland annat innan ett förändrat system sätts i drift igen.

11. Arbetsformerna har helt ändrats genom att man gör saker på ett nytt sätt. Helst vill man byta de gamla systemen.

Ibland kommer man på att man kan utföra arbetet i en verksamhet på ett helt nytt sätt. Det kan ibland vara föranlett av att informationsteknologin ger nya möjligheter, men ibland kan en idé eller en analys av själva sättet att utföra arbetet, göra att man kommer på ett bättre sätt att utföra det.

För att nå sådana effekter måste man kanske bygga om sina informationssystem. Ibland kan det finnas möjlighet att förändra befintliga system så att det nya sättet att utföra rutinerna på kan stödjas.

Olika stora förändringar kan behöva göras i det gamla systemet, beroende på hur olikt det nya sättet att arbeta är i förhållande till det gamla. Hur väl och överblickbart systemet går att förändra, beror på hur väl strukturerat och beskrivet det är.

12. Ny teknologi blir tillgänglig och det blir uppenbart att vissa delar av informationsteknologin måste bytas ut. Andra delar gör vad de skall – gränssnitt måste skapas mellan de nya och de gamla delarna.

Ett exempel på detta är att inmatning av data blir helt omodern genom att data kan fångas när man gör det operativa arbetet i verksamheten. T ex kan streckkodsläsning i försäljningskassor ersätta separat inmatning av vad som säljs. Hela det separata inmatningsledet kan då automatiseras. Däremot utför de efterföljande systemen den behandling som verksamheten vill, vilket gör att dessa kan behållas.

Detta scenario är egentligen ett exempel på att helt ny informationsteknologi blir tillgänglig och innebär att vissa delar av verksamhetsarbetet kan göras på ett helt nytt sätt.

Det kan innebära att en större del av verksamheten påverkas och att stora delar av det informationsteknologiska stödet blir påverkat. Men det kan också innebära att endast en del av verksamheten blir påverkad och att större delen av de existerande systemen kan behållas, eftersom de i övrigt gör rätt saker.

Ett exempel på det senare är de fall där streckkodsläsning i kassor direkt kan kopplas till beställningssystem och prissättningssystem.

Genom de nya delarna i systemet uppenbarar sig en del nya möjligheter att ställa samman information och dra slutsatser. Man kan till exempel analysera köpbeteende hos olika kundgrupper utifrån de marknadsföringsåtgärder som vidtas vid en viss tidpunkt.

I detta fall gäller det att skapa någon form av gränssnitt mellan de nya och de gamla delarna. Eventuellt kan detta ske genom ett meddelandesamverkansgränssnitt.

Exempel på ett fall inom ramen för scenarierna

Ett intressant scenariefall är shippingföretaget Wallenius, ett medelstort företag som aktivt tar sig an migrering och informationssystemstöd i en förnyad verksamhet.

Där har man påbörjat en process för att göra en reengineering-insats både avseende informationssystemen i sig och för hela den plattform (miljö) som man arbetar på.

Verksamheten är mitt i denna process och man börjar se vissa effekter av dessa insatser.

Det intressanta med detta fall är att det rör sig om ett medelstort företag som är effektivt i olika avseenden, men som inte har råd med att ha egna metodavdelningar. I stället har de antagit utmaningen och försöker lösa uppgiften med egen kunskap och med hjälp av kunskap som kan hämtas utifrån på lämpligt sätt.

En av slutsatserna verkar vara att också mindre företag kan utnyttja moderna miljöer på ett offensivt sätt och hitta nya sätt att använda informationsteknologin.

Det går åt en hel del fantasi och framåtanda, men samtidigt ger smidigheten i den mindre organisationen en möjlighet.

Förutsättningarna

Förutsättningarna var att:

Verksamheten får ökade och förändrade krav på sig genom att den blir allt mer internationaliserad och bedrivs på en mängd platser i världen. Dessutom vill kunderna att tjänsterna förändras så att man transporterar godset mer från dörr till dörr. Vidare vill kunderna ha fortlöpande information om var deras gods befinner sig samt, redan under transporten, ha information om huruvida skada har uppstått, så att man inte behöver upptäcka detta först vid framkomsten till destinationen. Administration av skadeanmälningar och ersättningskrav kan då påbörjas redan under resans gång.

Den miljö i vilken informationssystemen arbetade tenderade att bli stängd och dyr. Det innebar att man inte, tillräckligt fritt och spontant, kunde komma åt information i den befintliga miljöns databaser. Det innebar också att de existerande stordatorerna var dyra att köra. En client/server-lösning skulle bli billigare.

Driftskostnaden för att köra systemen misstänktes vara onödigt hög.

Åtgärderna blev att hela datormiljön byttes ut. En client/server-miljö lyftes in och ersatte stordatorerna som använts förut. Ett 4:e generationens verktyg byttes ut mot ett paket varmed man kan ställa fria SQL-frågor mot databas. PC infördes som klienter till UNIX-servern.

Flera av de gamla systemen skrivs nu om och utvidgas för att kunna hantera de nya verksamhetsuppgifterna. Verksamhetsanalys och modellering används för att ta reda på de nya verksamhetskraven innan utvecklingsarbete startas.

Effekterna

Effekterna har blivit att driftskostnaderna har sjunkit. Verksamheten håller på att få system som bättre kan stödja de nya verksamhetsuppgifterna och kraven från kunderna. Man kan dessutom lättare söka fritt efter den information man behöver för olika beslutssituationer.

Summering

Inom området reverse engineering och reengineering av informationssystem finns det en mängd verktyg och kunskaper för att utföra olika deluppgifter i samband med att man hanterar, och i olika avseenden, förbättrar systemarvet.

Det saknas dock i hög grad kunskap om hur man skall sätta samman dessa verktyg och kunskaper för att komma åt svårare, mer sammansatta situationer i större verksamheter.

I denna skrift har vi försökt strukturera upp ett antal verksamhetssituationer i vilka man antas vilja utnyttja systemarvet och samtidigt göra någon typ av förändring i verksamheten.

Det antas vidare att, beroende på vilken typ av situation det gäller, man behöver sätta samman ett antal åtgärder av reverse engineering och reengineering-typ för att få önskade effekter i verksamheten på kort och lång sikt.

Dessa serier av åtgärder bildar strategier som är lämpliga för respektive situation. Vi har skisserat och exemplifierat vilka sammansättningar av åtgärder som kan vara aktuella för de olika verksamhetssituationerna.

Kategoriseringen görs för att göra verksamhetssituationerna tydligare i syfte att diskutera olika sammansatta strategier. I verkliga fall är det ofta fråga om att flera av situationerna är aktuella.

I följande tabell har en sammanställning av de olika verksamhetsmässiga scenarierna gjorts.

Verksamhetsscenarier och åtgärder som kan bli aktuella i en strategi

Typscenario	Åtgärder på verksamhetsnivån	Åtgärder på informations-systemnivån	Åtgärder på implementeringsnivån
1. Verksamheten upplever att underhållet kostar för mycket pengar.	Utredning: Vad är normala underhållskostnader. Kommer verksamheten att förändras mycket? På vilket sätt måste man preparera för ett bättre underhåll?	Vad är det som gör att underhållskostnaderna är höga? Skapa bättre struktur. Skapa bättre beskrivningar.	Beskriv hur implementering hänger samman med den konceptuella nivån.
2. Verksamheten måste ha självständiga delar så att de t ex är avyttringsbara.	Vad är verksamhetsmässigt önskade delar? Vad är ansvarsmässiga och juridiska delar?	Ev nya avgränsningar och definitioner av nya informationssystemgränser. Definition av samband. Bättre beskrivningar.	Ev ny implementering eller reorganisering. Beskrivning i förhållande till den konceptuella nivån
3. Verksamheten skall styras med autonoma verksamhetsdelar	Avgränsning av verksamhetsdelar, deras uppgifter och gränssnitt samt definition av samband.	Skapa en strategi för hur systemen skall delas och avgränsas med hänsyn till den nya verksamhetsmässiga strukturen. Beskriv samband väl. Skapa goda beskrivningar.	Skapa realisering i enlighet med den nya autonoma delsystemstrukturen. Skaffa meddelandeöverföringsstöd så att meddelanden kan utväxlas enligt angivet ansvar. Lagring skall ske lokalt i varje del. Samverkan med andra delar skall ske via meddelanden.

4. Verksamheten regionaliseras och den centrala delen blir mindre.	Vad som skall utföras måste noggrant analyseras i ett verksamhetsmässigt perspektiv. Samband mellan delarna måste definieras.	System kan behöva delas och relevanta delar återanvändas ute i regionerna resp. i den mindre, centrala delen. Beskrivning av vad som finns i resp del viktig. Sambanden måste vara klara.	En ny implementering behöver i en del fall göras och strukturen skall spegla den nya verksamhetsstrukturen. Beskrivning och kopplingen till den konceptuella nivån viktig. Plattform för meddelandutväxling skapas om denna inriktning avses
5. Bättre struktur för att förändra informationssystem	Gör klart vad den nya miljön skall klara. Vad är de verksamhetsmässiga kraven? Vilka förändringar kommer man att förvänta. Av vilken typ?	Definiera noga vad den nya plattformen skall bestå av. Viktigt med definition av ambitionsnivå. Viktigt med beskrivning samt navigeringsstöd för att hitta	Strukturering av vad som finns/skall implementeras måste troligen förbättras. Beskrivningen och kopplingen till den konceptuella nivån viktig
6. Vidareutvecklingsbara verksamhetsdelar och informationssystem skall skapas genom återanvändbara komponenter.	Avgör vad återanvändningen skall syfta till. Vilka verksamhetsmässiga effekter vill man ha? Vilken nivå skall återanvändningen ligga på? Vem betalar? Upprätta kalkyl.	Avgräsning av lämpliga komponentdelar enligt tidigare utredning. Viktigt med beskrivning.	Utveckling av nya komponenter och avgräsning av gamla. Koppling av komponenter och deras beskrivning viktig. Oftast nödvändigt med beskrivningsverktyg för navigering.
7. Nya verksamhetstjänster måste skapas fast med existerande system.	Vilka verksamhetsmässiga tjänster skall utvecklas. Hur viktiga är de? Nyttoanalys för att motivera olika ambitionsnivåer.	Specifikation av de nya delarna och hur de skall hänga samman med de gamla delarna. Noggrann beskrivning av de nya delarna. Ev. verksamhetsorienterad beskrivning av de gamla delarna om man behöver gå in i dem.	God och strukturerad implementering av de nya delarna. Ev. behöver implementering av de gamla delarna förbättras om man misstänker att man behöver gå in i dem. Ett samverkansgränssnitt kan behöva skapas mellan nytt och gammalt.

<p>8.</p> <p>Man tvingas flytta informationssystem till nya plattformar pga leverantörsändringar. Restriktioner för verksamheten.</p>	<p>Vilka effekter har restriktionerna på den verksamhetsmässiga och affärsmässiga nivån? Kan det finnas positiva effekter att hämta för uppostringarna i samband med restriktionerna? Finns det anledning att omstrukturera så att man kan utnyttja den nya plattformen på ett bättre sätt och få extra effekter?</p>	<p>Om möjligt bör man förbättra beskrivningarna av vad som görs i systemen om man behöver gå in i dem i framtiden.</p> <p>Om man skapar delar på den nya plattformen, vad ligger i varje del?</p>	<p>Skapa en bra gränssnitt mellan de gamla systemen och gränsytan mot plattformen. Utnyttja moderniteterna i den nya plattformen.</p> <p>Kan arkitekturen förbättras i applikationen baserat på den nya plattformen?</p>
<p>9.</p> <p>Nya lagar kräver förändringar i existerande system.</p>	<p>Analys av hur de nya reglerna ser ut som skall in i systemet. Vad saknas i de gamla systemen? Vad är kostnad och nytta med att föra in ändringarna?</p>	<p>Noggrann beskrivning av begrepp och vad de nya reglerna består i (regelmodellering). Beskriv även andra regler i det existerande systemen så att man vet vad man byter ut och ändrar. Bygg på begreppsbeskrivningar. Ev. förbättra strukturen i systemet</p>	<p>Implementera de nya delarna i särskilda moduler så att man bidrar till att skapa en bättre struktur totalt.</p> <p>Beskrivningen i förhållande till den konceptuella nivån viktig.</p>
<p>10.</p> <p>Ny kvalitetsstyrning måste in i de existerande systemen</p>	<p>Vilken kvalitetsstyrning skall uppnås? På kort och lång sikt? Vilka effekter kan detta ge?</p>	<p>Specifikation av hur de nya förändringarna och påbyggnaderna skall föras in i de existerande systemen. Hur kan en bättre strukturering göras genom detta arbete? Bättre beskrivning av de nya och gamla delarna.</p>	<p>Implementering av de nya delarna på ett välstrukturerat sätt i särskilda moduler.</p>

11. Arbetsformerna har helt ändrats men man tvingas försöka utnyttja existerande system
- Utred vilka effekter det nya sättet att arbeta kommer att ge. Lönar det sig att bygga helt nya system? Vad är riskerna med detta? Kan man återanvända väsentliga delar av det man har? Det nya systemstödet beskrivs från en verksamhetsmässig synvinkel.
- Specificera vad det nya systemstödet skall kunna klara. Om man skall bygga nytt, specificera och ev köp komponenter. Om befintliga delar kan användas, avgränsa de delar som kan användas och skapa gränssnitt till dessa. Koppla in de avgränsade delarna i den nya strukturen.
- Nya delar implementeras strukturerat. Implementeringsnivån beskrivs väl i förhållande till den konceptuella nivån. De avgränsade existerande delarna förbättras eventuellt strukturellt och beskrivs.
12. Helt ny teknologi gör att man byter ut vissa delar av existerande system
- Vad innebär det i effekter och uppoffringar att utnyttja den nya tekniken? Vilka ytterligare effekter kan man få förutom de direkt kortsiktiga? Kan man mot låg marginalkostnad skapa ytterligare effekter?
- Var skall gränssnittet mellan gammalt och nytt ligga? Hur skall man skapa gränssnittet? Beskriv vilka begrepp som används och vad som görs i den nya delen. Beskriv gränssnittet.
- Implementera de nya delarna på ett strukturerat sätt. Beskriv hur implementeringen gjorts och hur dessa hänger samman med den konceptuella nivån. Implementera gränssnittet på ett strukturerat sätt, eventuellt via en ORB.

Referenser

Capers Jones, Applied Software Measurement, Assuring Productivity and Quality, Software Productivity Research, Inc, Mc Graw Hill, 1991, ISBN 0-07-032813-7.

Chikofsky, Elliot J., Cross II, James H., Reverse Engineering and Design Recovery: A Taxonomy, IEEE Software, January 1990.

Documentation, 3rd Reverse Engineering Forum, September 15 - 17, 1992, Northeastern University, Burlington, Massachusetts, USA.

Hefedh, M., Fatma, M., Ali M., Reusing Software: Issues and Research Directions, Departement de Mathématique et d'Informatique, Université de Québec à Montréal, School of Engineering and Computer Science, Oakland University, University of Tunis II, Tunisia and University of Ottawa, 1992.

IT 2000, Effektiv IT, Förutsättningar för ett nytt utvecklingsprogram inom informationsteknologins tillämpningsområden, En förstudie, Ds 1993:43, Näringsdepartementet, Regeringskansliets offsetcentral, 1993.

Object Management Group (OMG), The Common Object Request Broker: Architecture and Specification (CORBA), OMG Document Number 91.12.1, Revision 2.1.

Sneed, H., M., Economics of Software Re-engineering, Software Engineering Service, Rosenheimer, Muncken, Germany, Software maintenance: Research and Practice, Vol. 3, 1991.

Ulrich, W., M., Re-engineering: Defining an Integrated migration framework, Tactical Strategy Group, Aptos, CASE-trends, 1992.