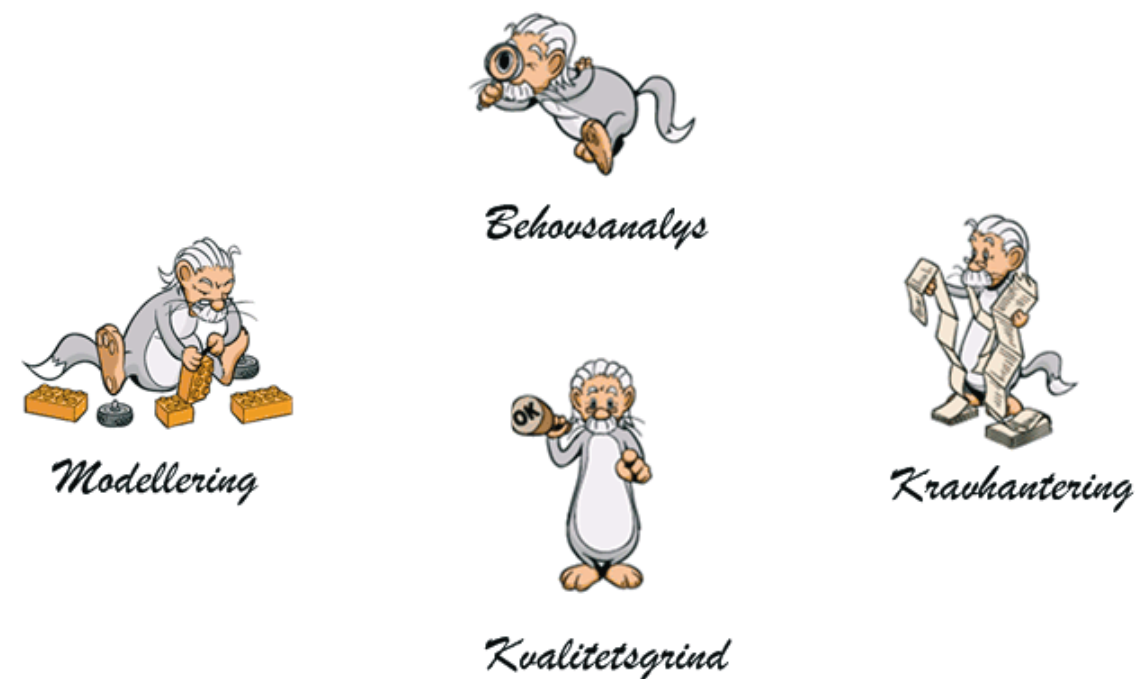




Ledningssystemsutveckling

Fallstudier kring kravhantering, modellering och kvalitetssäkring

NIKLAS HALLBERG, JOACHIM HANSSON, ERLAND JUNGERT, LARS WESTERDAHL,
SOFIE PILEMALM, HELENA GRANLUND, THOMAS SUNDMARK, PETER LITSEGÅRD,
BIRGITTA KYLESTEN, AMUND HUNSTAD, AMY RANKIN OCH HENRIK ERIKSSON



FOI är en huvudsakligen uppdragsfinansierad myndighet under Försvarsdepartementet. Kärnverksamheten är forskning, metod- och teknikutveckling till nytta för försvar och säkerhet. Organisationen har cirka 1000 anställda varav ungefär 800 är forskare. Detta gör organisationen till Sveriges största forskningsinstitut. FOI ger kunderna tillgång till ledande expertis inom ett stort antal tillämpningsområden såsom säkerhetspolitiska studier och analyser inom försvar och säkerhet, bedömning av olika typer av hot, system för ledning och hantering av kriser, skydd mot och hantering av farliga ämnen, IT-säkerhet och nya sensorers möjligheter.



FOI
Totalförsvarets forskningsinstitut
Informationssystem
Box 1165
581 11 Linköping

Tel: 013-37 80 00
Fax: 013-37 81 00

www.foi.se

FOI-R--2892--SE
ISSN 1650-1942

Användarrapport
December 2009

Informationssystem

Niklas Hallberg, Joachim Hansson, Erland Jungert,
Lars Westerdahl, Sofie Pilemalm, Helena
Granlund, Thomas Sundmark, Peter Litsegård,
Birgitta Kylesten, Amund Hunstad, Amy Rankin
och Henrik Eriksson

Ledningssystemsutveckling

Fallstudier kring kravhantering, modellering och kvalitetssäkring

Titel	Ledningssystemsutveckling: Fallstudier kring kravhantering, modellering och kvalitetssäkring
Title	C2 systems development: Case studies regarding requirements engineering, modeling and quality assurance
Rapportnr/Report no	FOI-R--2892--SE
Rapporttyp Report Type	Användarrapport
Månad/Month	December/December
Utgivningsår/Year	2009
Antal sidor/Pages	91 p
ISSN	ISSN 1650-1942
Kund/Customer	FM
Kompetensklass	25 Arkitektur och systemutveckling

Extra kompetensklass

Projektnr/Project no	E53078
Godkänd av/Approved by	Magnus Jändel

FOI, Totalförsvarets Forskningsinstitut	FOI, Swedish Defence Research Agency
Avdelningen för Informationssystem	Information Systems
Box 1165	Box 1165
581 11 Linköping	SE-581 11 Linköping

Sammanfattning

Att utveckla system är en utmaning, med många svårigheter som måste hanteras. De allra största utmaningarna vid all typ av utveckling är att specificera vad som skall skapas och sedan med precision realisera något som motsvarar detta. Denna rapport beskriver erfarenheter av frågeställningar inom systemutveckling som behöver beaktas och förstås vid etablering av modellbaserad utveckling. Syftet är att tillhandhålla Försvarmakten kunskap kring främst kravhantering, MODAF, modellbaserad utveckling samt kvalitetssäkring vid utveckling. Rapporten bygger på åtta fallstudier i vilka specifika frågeställningar relaterade till ledningssystemutveckling studerats. Tre av fallstudierna har beröring på behovsanalys och kravhantering. Fyra av fallstudierna berör modellbaserad utveckling. Två av fallstudierna berör MODAF. En av fallstudierna beskriver en databas avsedd för att lagra metoder och principer som funnits välfungerade.

Resultat beskriver erfarenheter av att tillämpa en ansats för kravhantering av ledningssystem för så väl insats som verksamhetsledning. Studierna visar att grund principen för metoden fungerar väl, med mindre anpassningar till tillämpningsområdet. Resultat beskriver också att modellbaserad utveckling inte betyder att det är möjligt att bortse från ”gamla sanningar” så som behovet av nära samverkan mellan domänexperter och utvecklare. Vidare beskrivs att MODAF upplevs som ett väl fungerade ramverk, som dock inte är trivialt att tillämpa utan kommer att kräva omfattande utbildningsinsatser. Resultat beskriver även ett intervjuinstrument för att samla erfarenheter av att tillämpa MODAF samt en arkitektur för en kunskapsdatabas avsedd för vetenskaplig kvalitetssäkrad kunskap avseende systemutveckling.

Nyckelord: Kravhantering, modellering, beslutsgrindar, MODAF

Summary

Systems development is a challenge, with many difficult tasks to take into consideration. The major challenge in all kinds of development is to specify what should be developed and then realize this. This report presents experiences concerning issues in systems development that need to be considered and understood when establishing model-based systems. The objective is to provide the Swedish Armed Forces with knowledge about requirements engineering, MODAF, model-based development and quality assurance in development. The report is based on eight case studies addressing specific issues related to development of management systems. Three of the case studies concern needs analysis and requirements management. Four of the case studies are about model-based development. Two of the case studies relate to MODAF. One of the case studies presents a knowledge database used to store methods and principles that have been well-functioning.

The results present experiences in applying an approach for requirements engineering of management systems for operations as well as business management. The outcome shows that the basic principle of the method works well, with minor adjustments to the scope. The results also suggest that model-based development has to acknowledge "old truths" which include the need for close collaboration between domain experts and professional developers. Furthermore, the results show that experiences when using MODAF is that it works well as a framework. However, it is not trivial to implement and will require extensive training. The results also describe an instrument used for interviews when gathering experiences of applying MODAF and the architecture of a knowledge database designed for quality-assured scientific knowledge relating to software development.

Keywords: Requirements engineering, modeling, decision gates, MODAF

Innehållsförteckning

1	Inledning	9
2	Bakgrund	11
3	Resultat	12
3.1	Modellbaserad utveckling.....	12
3.2	Kvalitetsdriven kravhantering för utveckling av insatsledningssystem	12
3.3	Kvalitetsdriven kravhantering för utveckling av avvikelsehantering.....	13
3.4	Kravhantering och MODAF	13
3.5	Erfarenheter av MODAF.....	13
3.6	Erfarenheter av modellbaserad utveckling.....	14
3.7	Kvalitets- och beslutsgrindar	14
3.8	Databas: Metoder och principer för utveckling av ledningssystem.....	15
4	Diskussion	16
5	Fallstudier	17
5.1	Fallstudie: Modellbaserad utveckling	17
5.1.1	Inledning.....	17
5.1.2	Metod	18
5.1.3	Resultat	18
5.1.4	Diskussion	25
5.2	Fallstudie: Kvalitetsdriven kravhantering för utveckling av insatsledningssystem	27
5.2.1	Inledning.....	27
5.2.2	Metod	28
5.2.3	Resultat	28
5.2.4	Diskussion	33
5.3	Fallstudie: Kvalitetsdriven kravhantering för utveckling av avvikelsehantering.....	34

5.3.1	Inledning.....	34
5.3.2	Metod	35
5.3.3	Resultat.....	37
5.3.4	Diskussion.....	43
5.4	Fallstudie: Kravhantering och MODAF	44
5.4.1	Inledning.....	44
5.4.2	Metod	45
5.4.3	Resultat.....	45
5.4.4	Diskussion.....	48
5.5	Fallstudie: Erfarenheter av MODAF	49
5.5.1	Inledning.....	49
5.5.2	Metod	51
5.5.3	Resultat.....	51
5.5.4	Diskussion.....	56
5.6	Fallstudie: Erfarenheter av modellbaserad utveckling.....	57
5.6.1	Inledning.....	57
5.6.2	Metod	59
5.6.3	Resultat.....	60
5.6.4	Diskussion.....	63
5.7	Fallstudie: Kvalitets- och beslutsgrindar	66
5.7.1	Inledning.....	66
5.7.2	Metod	67
5.7.3	Resultat.....	67
5.7.4	Diskussion.....	78
5.7.5	Grindar, roller och ansvar	79
5.7.6	Kriterier.....	80
5.7.7	Slutsatser	80
5.8	Fallstudie: Databas: Metoder och principer för utveckling av ledningssystem	81
5.8.1	Inledning.....	81
5.8.2	Metod	81
5.8.3	Resultat.....	82
5.8.4	Diskussion.....	85
6	Referenser	86

1 Inledning

Att utveckla ledningssystem är en utmaning, med många svårigheter att hantera och fallgropar att undvika (Hallberg et al 2008). Att avhandla samtliga dessa i en rapport, bok eller ett projekt är inte möjligt. För att belysa denna problematik från olika håll har i denna rapport beskrivits åtta fallstudier i vilka specifika frågeställningar relaterade till ledningssystemutveckling studerats.

En av de allra största utmaningarna vid all systemutveckling är att specificera kraven (Kasser, 2007). Det spelar ingen roll hur ”bra” den utvecklingsmetod som nyttjas är, hur omfattande designen är eller hur avancerade tekniker som nyttjas. Är ingångsvärdena (behoven) felaktiga eller otillräckliga kommer systemet inte att motsvara de ställda förväntningarna och kostsamma omarbetningar kommer att krävas. Tre av fallstudierna har beröring på behovsanalys och kravhantering, vilket syftar till att tillhandhålla Försvarmakten kunskap avseende behovsanalys och kravhantering.

Försvarmakten lägger ner omfattande resurser på att utveckla en modellbaserad ansats för att utveckla system¹. Begreppet modellbaserad utveckling är dock inte entydigt definierat, och det är också oklart vad det finns för fördelar och utmaningar med att anamma en sådan strategi. Fyra av fallstudierna berör modellbaserad utveckling, vilket syftar till att tillhandhålla Försvarmakten kunskap kring tillämpning av modellbaserad utveckling.

Försvarmakten har valt MODAF som det beskrivningsramverk som skall ligga till grund för modellering av system på samtliga nivåer². Två av fallstudierna berör MODAF. Syftet med dessa fallstudier är att tillhandhålla Försvarmakten kunskap kring tillämpning av MODAF samt hur traditionell kravhantering kan nyttjas som grund för modellering.

Det finns ingen brist på ansatser, aspekter, koncept, metoder, tekniker och principer för systemutveckling (Hallberg et al 2008). Tvärtom finns dessa i överflöd. Utmaningen är istället att hitta de som fungerar väl inom Försvarmakten och att tillse att dessa tillämpas. En av fallstudierna berör en databas avsedd för att lagra metoder och principer som funnits välfungerade. Syftet är att på sikt Försvarmakten skall ha tillgång till kunskap om vetenskapligt och praktiskt beprövade metoder och principer som fungerar väl för utveckling av system.

Denna rapport är en redovisning av den verksamhet som bedrivits i det FoT-finansierade projektet *Arkitekturbaserad ledningssystemutveckling*. Arbetet som ligger till grund för rapporten syftar till att beskriva erfarenheter av

¹ C LEADS (2008) ”C LEADS beslut avseende fortsatt arbete med Modellbaserad Förmågeutveckling, bilaga” 2008-11-18, HKV 09 947: 78917

² FM/CIO/0028/09 Instruktion för beskrivning av system inom FM

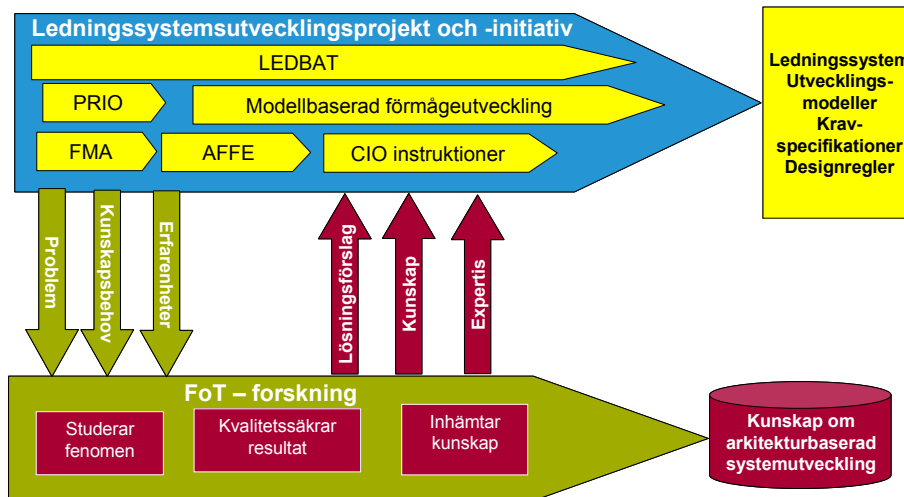
systemutveckling som behöver inarbetas, beaktas och förstås vid etablering av modellbaserad systemutveckling.

Rapporten består av följande delstudier:

1. Modellbaserad utveckling
2. Kvalitetsdriven kravhantering för utveckling av insatsledningssystem
3. Kvalitetsdriven kravhantering för utveckling av avvikelshantering
4. Kravhantering och MODAF
5. Erfarenheter av MODAF
6. Erfarenheter av modellbaserad utveckling
7. Kvalitets- och beslutsgrindar
8. Databas: Metoder och principer för utveckling av ledningssystem.

2 Bakgrund

Projektet *Arkitekturbaserad ledningssystemutveckling* syftar till att tillhandahålla Försvarmakten stöd avseende arkitekturbaserad utveckling av ledningssystem. Målet med projektet är att tillhandahålla kunskap och kompetens avseende arkitekturbaserad utveckling av ledningssystem, som är kvalitetssäkrad – vetenskapligt grundad och praktiskt tillgänglig. Detta skall ske genom att projektets medarbetare aktivt medverkar i olika ledningssystemutvecklingsprojekt och andra relaterade initiativ inom Försvarmakten. Detta för att inhämta kunskap avseende svårigheter, kunskapsbehov och erfarenheter inom området för att härfter kunna använda denna kunskap som grund för att genomföra fortsatt forskningsverksamhet inom FoT-projektet. Kunskap som erhålls inom FoT-projektet avseende möjliga lösningar återförs i sin tur till projekten och andra liknade verksamheter (Figur 1). Under projektets första år genomfördes dels en behovsanalys av vilket stöd vid utveckling av ledningssystem som föreligger, dels kartlades principer och metoder för systemutveckling (Hallberg et al 2008; Hallberg, Pilemalm & Westerdahl, 2008). Denna rapport beskriver resultat från det andra året av projektet.



Figur 1. Beskrivning av relationen och informationsflödet mellan Försvarmakten ledningssystemutvecklingsinitiativ och FoT-projektet.

3 Resultat

Syftet med denna rapport är att tillhandhålla Försvarmakten kunskap kring främst kravhantering, MODAF, modellbaserad utveckling samt kvalitetssäkring vid utveckling. Denna del av rapporten presenterar kort resultatet från de åtta fallstudierna: (1) Modellbaserad utveckling, (2) Kvalitetsdriven kravhantering för utveckling av insatsledningssystem, (3) Kvalitetsdriven kravhantering för utveckling av avvikelshantering, (4) Kravhantering och MODAF, (5) Erfarenheter av MODAF, (6) Erfarenheter av modellbaserad utveckling, (7) Kvalitets- och beslutsgrindar och (8) Databas: Metoder och principer för utveckling av ledningssystem.

3.1 Modellbaserad utveckling

Denna fallstudie beskriver vad modellbaserad utveckling är, vad det syftar till och hur det bedrivs. Studien är baserad på en litteraturstudie. Modellbaserad utveckling anses komma tillrätta med en del av de svårigheter som finns i systemutveckling. Generellt gäller för de i litteraturen beskrivna ansatserna att dessa utgår från en eller flera modeller av den omgivning som systemet skall stödja och nyttjas i. Dessa modeller används för att specificera de system som skall byggas. Vissa ansatser beskriver även modeller för att realisera systemet. Metamodeller som beskriver relationer mellan modellerna nyttjas för att automatiskt generera modeller utifrån tidigare skapade modeller. Dessa metamodeller ger också möjlighet till modifieringar i en modell och får genomslag i övriga relaterade modeller. Modellbaserad utveckling bör drivas iterativt och det anses viktigt att kontinuerligt säkerställa kvalitén på de modeller som skapas.

3.2 Kvalitetsdriven kravhantering för utveckling av insatsledningssystem

Denna fallstudie presenterar en ansats för att identifiera krav avseende ledningssystem för genomförande av insatser. Ansatsen är baserad på Quality Function Deployment (QFD), användningsfall, prototyper och scenario utvärderingar. Den föreslagna strategin utgörs av tio aktiviteter: (1) Datainsamling, (2) Identifiera utsagor, (3) Fastställa behoven, (4) Analys och strukturering av behov, (5) Prioritering av behov, (6) Utveckling av användningsfall, (7) Analys av krav, (8) Utveckling av scenarier och prototyper, (9) Utvärderingar av användningsfall och (10) Revidering av användningsfall och krav. Ansatsen tillämpas och utvärderas i en studie. Resultatet av fallstudien visar att medverkan av slutanvändare i kravhanteringsprocessen är avgörande och den så kallade kundrösttabellen är ett användbart verktyg för att tolka användares utsagor och formulera dessa som behov.

3.3 Kvalitetsdriven kravhantering för utveckling av avvikelshantering

Att specificera system så dessa erhåller rätt funktionalitet är en av de mest kritiska aktiviteterna inom systemutveckling. För att kontinuerligt utveckla en organisation/verksamhet, som så Försvarsmakten, är det centralt att kunna hantera avvikelser, vilket kan stödjas med olika former av IT-system. Inom Försvarsmakten används ett antal olika IT-system för avvikelshantering, men det finns en ambition att införa ett gemensamt system. För att uppnå detta så har Försvarsmakten upphandlat ett system som varit i pilotdrift under 2009. Denna fallstudie syftar till att studera tillämpningen av en ansats för kravhantering, benämnd Kvalitetsdriven kravhantering. Detta genom att tillämpa ansatsen för att identifiera behov av IT-system för avvikelshantering inom Försvarsmakten.

Fallstudien genomfördes i tre steg (1) anpassning av ansatsen Kvalitetsdriven kravhantering att passa utveckling av ett IT-stöd för verksamhetsfunktion, (2) tillämpa ansatsen och dokumentera gjorda erfarenheter samt (3) sammanställa och presentera resultatet.

Metoden befanns fungera tillfredställande. Processmodellen upplevdes som ett bra stöd då användningsfall skapades. Totalt 430 behov identifierades. Dessa innefattade varför Försvarsmakten behöver avvikelshantering och vilket stöd de som är involverade i avvikelshantering behöver.

3.4 Kravhantering och MODAF

MODAF är ett beskrivningsramverk och det finns behov av att studera hur traditionella kravhanteringsmetoder kan nyttjas för att lägga grund för de modeller som skall beskrivas. I denna fallstudie undersöks hur användningsfall och en övergripande processbeskrivning kan nyttjas för att börja beskriva de initiala MODAF modellerna. Resultatet visar att det inte finns helt entydiga relationer mellan användningsfallen och MODAF vyerna. Det fanns en tendens att beskriva förmågor på alltför hög detaljeringsnivå. Men så väl användningsfall som processmodellen gav stöd för att skapa modeller enligt MODAF.

3.5 Erfarenheter av MODAF

MODAF har valts av Försvarsmakten som arkitekturramverk. Det saknas dock vetenskapliga studier kring tillämpningen av MODAF. Vad är det i tillämpningen som är svårt, hur kan detta hanteras och vilka är dess styrkor? Denna fallstudie syftar till att skapa ett intervjuinstrument för att inhämta erfarenheter kring användningen av MODAF. I fallstudien tillämpas

intervjuinstrumentet dels för att inhämta erfarenheter kring MODAF, men också för att validera instrumentet.

Resultat visar att för det syfte MODAF är utvecklat fungerar det väl. Men det är inte någon "silverkula" som fungerar för allt. Vidare är MODAF relativt komplext samtidigt som de verktyg som stödjer modelleringen är svåra att nyttja. Därmed kommer det att krävas en omfattande utbildningsinsats för att lyckas införa MODAF i Försvarmakten. Införandet kräver även att fördelarna med ett gemensamt ramverk tydligt presenteras. Intervjuinstrumentet fanns fungera väl, och kan nyttjas för betydligt mer omfattande studier.

3.6 Erfarenheter av modellbaserad utveckling

I denna fallstudie studeras erfarenheterna av modellbaserad utveckling i form av fördelar, nackdelar och utmaningar med utgångspunkt från Försvarmaktens Program PRIO. Fallstudien visar att det inom PRIO upplevts många av de utmaningar och problem som generellt förknippas med modellbaserad utveckling. Projektet illustrerar inneboende brister i många modellbaserade ansatser. De viktigaste utmaningarna är vikten av förändringsledning, helhetsperspektiv samt tid och resurser. När det gäller brist avseende kompetens kan detta hänföras till att modellbaserade ansatser tenderar att i otillräcklig omfattning uppmärksamma behovet av användarmedverkan och datainsamling till grund för konstruktion av modellerna.

3.7 Kvalitets- och beslutsgrindar

Syftet med denna fallstudie är att beskriva hur kvalitets- och beslutsgrindar utformas och används. Studien är baserad på en litteraturstudie. Utformningen av kvalitetsgrindar ser på en övergripande nivå relativt lika ut mellan olika industrier och företag. På en mer detaljerad nivå kan det skilja sig mycket eftersom grindarna är anpassade efter individuella organisations- eller projektmål. Många positiva erfarenheter från användare av grindsystem har dokumenterats men litteraturstudien visar på att det finns fallgropar. Ett av de största problemen är grindar som saknar "tänder", vilka kan släppa igenom projekt som inte håller tillräckligt hög kvalitet. Ett annat problem är att avgöra på vilken nivå kriterierna skall ligga och hur de skall utformas. Det finns idag inga entydiga svar på hur dessa frågor skall hanteras.

3.8 Databas: Metoder och principer för utveckling av ledningssystem.

För att dokumentera och tillgängliggöra den kunskap som byggs upp inom projektet *Arkitekturbaserad ledningssystemsutveckling*, skapas en kunskapsdatabas vars syfte är att lagra vetenskapligt kvalitetssäkrade principer och metoder avseende systemutveckling.

För att tillgängliggöra ovan nämnda metoder och principer utvecklas ett antal gränssnitt mot kunskapsdatabasen. Exempel på sådana gränssnitt är:

- Inmatning/underhåll
- Sökfunktion, beslutsstöd
- Handboksgenerering

4 Diskussion

I denna rapport beskrivs åtta fallstudier, var av sex rör kravhantering och modellering ur olika perspektiv. Syftet med rapporten är att tillhandhålla Försvarsmakten kunskap kring främst kravhantering, MODAF, modellbaserad utveckling samt kvalitetssäkring vid utveckling. Fallstudier visar att nyttjandet av modellerbaserad utveckling, så som MODAF baserad utveckling, inte medför att de ”gamla” sanningarna inom systemutveckling kan bortses från. Det är fortfarande av största vikt att tydligt fastställa systemgränser. Det vill säga, vad skall beskrivas och utvecklas. Vidare är den information som ligger till grund för modellering viktig. Det vill säga, vad skall åstadkommas och vilka problem/behov skall tillfredställas. Fördelarna med en modellbaserad ansats är dock att spårbarhet kan erhållas, vilket bland annat ger bättre möjlighet till överblick av utvecklingen.

Kvalitetsgrindar har nyttjats relativt länge inom systemutveckling, och det finns en övergripande samsyn på hur dessa skall utformas. Men för att passa en viss verksamhet måste den mer detaljerade nivå anpassas efter individuella organisations- eller projektmål. Det finns många positiva erfarenheter av kvalitetsgrindar. Detta indikerar att det vore lämpligt att vidare studera möjligheten att införa någon form av effektivt grindsystem i Försvarsmakten i och med övergången till modellbaserad utveckling. En av de största utmaningarna med att få till effektiva grindar är att de måste ges ”vassa tänder”, som inte släpper igenom resultat med för låg kvalitet. Ett annat problem är att avgöra på vilken nivå kriterierna skall ligga och hur de skall utformas. Det finns idag inga entydiga svar på hur dessa svårigheter skall hanteras.

Det finns många ytterligare frågeställningar som behöver beaktas och studier av möjligheter av vad som kan göras för att underlätta och stärka modellbaserad utveckling. Dessa frågeställningar kan hämtas från befintlig litteratur om vad som är viktigt vid systemutveckling. Exempel på detta är kommunikation med representanter för verksamheten och användarna, påbörja prioritering så tidigt som möjligt i utvecklingen. Standarder och verktyg kommer säkert att öka framgången för modellbaserad utveckling, men för att lyckas krävs att ett antal frågeställningar beaktas.

5 Fallstudier

Detta kapitel presenterar åtta fallstudier (1) Modellbaserad utveckling, (2) Kvalitetsdriven kravhantering för utveckling av insatsledningssystem, (3) Kvalitetsdriven kravhantering för utveckling av avvikelshantering, (4) Kravhantering och MODAF, (5) Erfarenheter av MODAF, Arkitekturramverk, (6) Erfarenheter av modellbaserad utveckling, (7) Kvalitets- och beslutsgrindar och (8) Databas: Metoder och principer för utveckling av ledningssystem. Varje fallstudie beskrivs med en sammanfattning, inledning, genomförande, resultat och en diskussion.

5.1 Fallstudie: Modellbaserad utveckling

(Denna fallstudie presenteras mer utförligt i Hallberg, N., Sparf, M., Sjödin, L. och Pilemalm, S. (2009) Modellbaserad utveckling: Omvärldsanalys. FOI MEMO 2842)

Att utveckla system som motsvarar ställda förväntningar är inte enkelt. Modellbaserad utveckling anses komma tillrätta med en del av de svårigheter som finns i systemutveckling. Denna fallstudie beskriver vad modellbaserad utveckling är, vad den syftar till och hur den bedrivs. Det pågår ett arbete inom Försvarsmakten med att införa en ansats för modellbaserad utveckling. Syfte med denna fallstudie är att tillhandahålla Försvarsmakten kunskap om modellbaserad utveckling. Fallstudien är baserad på en litteraturstudie. Generellt gäller för beskrivna ansatser att dessa utgår från en eller flera modeller av den omgivning som systemet skall stödja och nyttjas i. Dessa modeller används för att specificera de system som skall byggas. Vissa ansatser beskriver även modeller för att realisera systemet. Metamodeller som beskriver relationer mellan modellerna nyttjas för att automatiskt generera modeller utifrån tidigare skapade modeller. Dessa metamodeller ger också möjlighet till modifieringar i en modell och får genomslag i övriga relaterade modeller. Modellbaserad utveckling bör drivas iterativt och det anses viktigt att kontinuerligt säkerställa kvalitén på de modeller som skapas.

5.1.1 Inledning

Att utveckla system är svårt och tidskrävande (Collin, 2003; Kasser, 2007). Förståelse för en framtida användning skall omsättas i behov, som nyttjas för att specificera krav som systemet skall uppfylla. Krav skall omsättas till en design, som skall nyttjas för att realisera ett system (Hallberg et al, 2008). Detta system skall motsvara de behov som kravställt. Detta innebär ett omfattande behov av att kunna tolka, omsätta och kommunicera information mellan varje utvecklingssteg. Det är dock svårt att överföra information mellan olika

utvecklingsaktiviteter och mellan olika iterationer när dessa är beskrivna i separata dokument.

Olika former av modeller har länge nyttjas inom systemutvecklingen. Men modellbaserad utveckling har nu bättre förutsättningar än tidigare då verktygsstöd och standarder finns tillgängliga (Kulkarni & Reddy, 2008). Syftet med denna fallstudie är att beskriva vad modellbaserad utveckling är, vad det syftar till och hur den bedrivs.

5.1.2 Metod

Denna fallstudie baseras på en litteraturstudie avseende modellbaserad utveckling. Sökord som nyttjades var ”Model based development” och ”Model driven development” efterhand som litteratur identifierats har uppföljningar av nyckelreferenser gjorts och läsaren hänvisas till dessa.

5.1.3 Resultat

Grunden för modellbaserad utveckling omfattas av (1) utvecklingsprocesser, (2) metamodeller och (3) relationer och möjlighet till transformationer (Sottet et al 2006; Romero et al 2007). Utvecklingsprocessen syftar till att skapa och successivt förfinas modeller, där ytterligare information läggs till i efterkommande steg och iterationer (Sottet et al, 2006). I modellbaserad utveckling utgör modeller indata till och resultatet från samtliga steg i utvecklingen tills det slutliga systemet erhållits (Balasubramanian et al 2006). En typisk utvecklingsprocess tar sin utgångspunkt i domänmodeller som beskriver den omgivning i vilket systemet skall användas, exempelvis verksamhetsmodeller, modeller som beskriver uppgifter systemet skall lösa samt modeller som representerar de tänkta användarna. Utifrån dessa skapas modeller som representerar systemet som skall utvecklas, exempelvis modeller avseende processer, användarinteraktion, krav, design och programkod. Sottet et al (2006) hävdar att i praktiken är dessa modellbaserade utvecklingsprocesser iterativa. Den första iterationen kan då vara att skapa initiala modeller vilka förfinas under varje iteration. Transformation mellan en modell till en annan är information om relationerna mellan dessa modeller. Denna information kan nyttjas av transformationsverktyg för att automatisera processen att skapa en modell baserad på en annan (Kühne, 2006).

Metamodeller är centrala i modellbaserad utveckling, utan dessa skulle enskilda modeller bli isolerade beskrivningar (Gitzel & Korthaus, 2004; Kühne, 2006). Dessa är modeller av modelleringspråk för att beskriva modellerna och deras relationer. Metamodeller ger förutsättningar att kunna transformera olika typer av modeller mellan varan, manuellt eller automatiskt (Mellor, Clark, & Futagami, 2003; Kühne, 2006).

5.1.3.1 Användning

Det finns ett stort intresse för modellbaserad utveckling inom programvaruutveckling, ofta med modelldriven arkitektur (eng. Model Driven Architecture, MDA) som grund. MDA baseras på tre steg (ex. Siegel, 2005). I första steget produceras modeller avseende verksamhet och beteende utan att ta hänsyn till vilken teknik som senare skall nyttjas. Dessa modeller skall vara *plattformsoberoende* (eng. Platform Independent Model, PIM). I andra steget, utifrån de modeller som skapats i första steget skapas en eller flera modeller som är beroende av vald teknologi. Det vill säga de är *plattformberoende* (eng. Platform-Specific Model, PSM). De plattformberoende modellerna skall ses som bryggan till implementeringen, som inte tillför något ytterligare avseende verksamhet och beteende. Flera olika typer av modeller kan nyttjas för att beskriva dessa nivåer (Fardoun et al 2009). I det tredje steget genereras programkod.

Baserat på modeller som beskriver användningskontexten, går det att skapa applikationer vars gränssnitt automatiskt anpassas med hänsyn till kontexten vid användningen. Dessa är exempelvis användbara vid utveckling av webbapplikationer som används av heterogena användargrupper (Kapitsaki, et al 2009). Porres et al (2008) redogör för en modellbaserad ansats för att beskriva och införa handlingsregler (eng: guidelines) i beslutsstödssystem. När handlingsreglerna behöver förändras räcker det med att modifiera modellerna för att anpassa beslutsstödet.

Nunes och Schwabe (2006) beskriver en ansats för att skapa prototyper av användargränssnitt baserat på modeller, som nyttjas för att generera programkod. På så vis kan de snabbt åstadkomma förändringar och möjliggör för icke-tekniskt insatta personer att medverka i utvecklingen. Molina et al (2008; 2009) redogör för en process och en notation för att beskriva processer och skapa modellbaserade användargränssnitt utifrån dessa. Sousa et al (2008) beskriver en modellbaserad process i tre faser, *Conception, management* och *application* för att utifrån verksamhetsprocesser skapa användargränssnitt till applikationer. Hallberg, Andersson och Ölvander (2009) beskriver en modellbaserad process och ett ”lättviktigt” arkitekturramverk för att utveckla ledningssystem.

5.1.3.2 Tillämpning

Vid införande av modelldriven utveckling är det viktigt att ta hänsyn till de tekniska, organisatoriska och sociala förutsättningar som råder (Selic, 2003). För att lagra information behövs förutom metamodeller och modeller någon form av datalager, men detta är inte tillräckligt. Effektiv modellbaserad utveckling förutsätter dessutom tillgång till exempelvis verktyg för modellbeskrivning och validering av modeller samt beslutsstöd (Romero et al 2007). Modellerna som enbart används för dokumentation har ett begränsat värde, då de snart skiljer sig från den verklighet de representerar. Att kunna automatiskt skapa system

baserat på modeller som en framgångsfaktor (Selic, 2003). Kulkarni och Reddy (2008) har lång erfarenhet av att använda modellbaserad utveckling för att utveckla informationssystem. Dessa erfarenheter innefattar kravet på att avsätta ordenligt med resurser för att i det initiala utvecklingsarbetet välja rätt arkitektur och utvecklingsstrategi samt att införa dessa i metamodeller, vilket de hävdar betalar sig i senare faser.

Att använda sig av olika former av modeller har länge varit vanligt inom systemutveckling (Hay, 2003; Sommerville, 2001; Friedenthal, Moore, & Steiner, 2008). Inom modellbaserad utveckling ses modeller som representationer av ett system (Mohagheghi, Dehlen, & Neple, 2009). Modeller kan användas för att beskriva existerande system och systemkontexter, samt för att beskriva krav på och design av framtida system (Hay, 2003). Inom systemutveckling ses modellering som en användbar teknik för att hantera komplexitet (Hallberg och Fransson, 2001). Modellerna utgör ett verktyg för att analysera systemegenskaper som grund för att exempelvis identifiera krav och för att kommunicera med användarna (Jacobson, Ericsson, & Jacobson, 1995). Modeller möjliggör också att olika aspekter av problemområden i verksamheten belyses från olika perspektiv. De gör det också möjligt att dela in komplexa system i mer överskådliga och hanterbara delar (Hull, Jackson, & Dick, 2005; Carlock & Fenton, 2001).

Att modellera verksamheter innebär att identifiera vilka aktörerna är, vilka aktiviteter de utför, hur dessa aktiviteter och aktörer är organiserade samt vilka resurser aktörerna nyttjar. Dessa modeller utgör en grund för utveckling av såväl tekniska stödsystem som förbättring av existerande verksamhetsstrukturer och processer (Hull, Jackson, & Dick, 2005). Att ta fram sådana modeller kräver (1) en utvecklingsstrategi, (2) en metod för att konstruera modellen, och (3) ett modelleringsspråk för att beskriva modellen.

5.1.3.3 Fördelar

I modellbaserad utveckling skapas de initiala modellerna utan hänsyn till vilka underliggande teknologier som skall användas (Selic, 2003). Detta är en fördel då specificering av funktioner och beteende kan göras oberoende av förändringar i de teknologier som nyttjas. Modelldriven utveckling ökar möjligheten för dem som inte är tekniska experter att aktivt delta i att designa och bygga system (Balasubramanian et al, 2006). Det är vidare användbart för utvecklare att kunna exekvera sina modeller i simulerade miljöer för att kunna verifiera och validera dessa (Selic, 2003). En ytterligare fördel med ett modellbaserat angreppssätt är att det stödjer inkrementell iterativ utveckling som är nödvändig vid utveckling av komplexa system (Selic, 2003).

Vid modellbaserad utveckling av verksamheter är det möjligt att separera verksamhet från informationssystem, samtidigt som kopplingarna mellan dessa bibehålls. På så vis kan verksamheten utvecklas utan att hänsyn till att

begräsningar till tekniska begräsningar måste göras, samtidigt som informations-system som stöder verksamheten snabbt kan realiseras (Lee, 2005; Kulkarni & Reddy, 2008). Lee (2005) menar vidare att modellbaserad utveckling möjliggör verksamhetsdriven utveckling istället för den mer traditionellt teknikdrivna. Modellbaserad utveckling stödjer snabba anpassningar baserade på validering av funktionella krav tillsammans med användare, men det finns fortfarande brister avseende hur icke-funktionella krav skall hanteras (Kulkarni & Reddy, 2008).

5.1.3.4 Modellkvalité

Det är viktigt att vid denna typ av modellering som första steg bestämma målet med de modeller som skall skapas och därefter fokusera framtagandet på detta mål. Stora modeller blir fort ohanterliga och oanvändbara. Det är också svårt att avgöra vilket som är den viktiga informationen i alltför omfattande modeller. Att ta fram modeller ofta tidskrävande, och än mer resurser krävs för att underhålla och uppdatera dem. Informationsmodeller har ofta längre giltighet än processmodeller (Hay, 2003).

Selic (2003) hävdar att för att vara användbara och effektiva måste modeller vara: (1) *abstrakta*, (2) *begripliga*, (3) *korrekta*, (4) *förutsägbara* och (5) *kostnadseffektiva*. Den viktigaste egenskapen är abstraktiviteten, vilken innebär att en modell skall ge en förenklad beskrivning av det som den representerar. Detta genom att till exempel ta bort eller dölja detaljer som är inte är relevanta och därmed skapa en bättre förståelse för det som är viktigt. Modeller skall enbart beskriva de egenskaper som är av intresse. Begriplighet relaterar till modellens förmåga att förmedla en komplex företeelse så att den är begripligt för dem som skall använda modellen. Förutsägbarhet innebära att modeller skall kunna användas för att förutsäga beteende hos det som de representerar. Kostnadseffektivitet innebär att modellerna måste vara betydligt billigare att konstruera och analysera än det är att direkt bygga systemet. Andra som skriver om kvalité avseende modeller är exempelvis Mohagheghi, Dehlen och Neple (2009) samt Moody och Shank (2003).

Mohagheghi et al (2009) anger sex kvalitetsaspekter på modeller som är viktiga att förstå och uppnå till en tillfredställande nivå. Dessa är *riktighet* (eng: correctness), *fullständighet* (eng: completeness), *konsistens* (eng: consistency), *begriplighet* (eng: comprehensibility), *begränsad* (eng: confinement) och *förändlighet* (eng: changeability). De beskriver vidare hur dessa kvalitetsaspekter relaterar till olika företeelser, roller och verktyg i modellbaserad utveckling.

Moody och Shank (2003) anger åtta kvalitetsfaktorer; *riktighet* (eng: correctness), *fullständighet* (eng: completeness), *anpassningsbarhet* (eng: flexibility), *enkelhet* (eng: simplicity), *konsistens* (eng: integration; med denna kvalitetsfaktor avses "consistency"), *begriplighet* (eng: understability), *integritet* (eng: integrity) och möjlighet till *realisering* (eng: implementability). De

beskriver vidare relationerna mellan dessa kvalitetsfaktorer samt till vilka som skall medverka i att säkerställa att tillräcklig kvalitet uppnås.

5.1.3.5 Olika typer av modeller

En utmaning med modellbaserad utveckling är att det inte räcker med en typ av modell utan flera olika modeller behövs. Enligt Danieles (2002) kan modeller inom systemutveckling delas upp i Konceptuella modeller, Specifikationsmodeller och Implementationsmodeller. Konceptuella modeller beskriver en situation i omgivningen. Specifikationsmodeller definierar vad systemet skall göra och funktioner som skall tillhandhållas. Dessa är oberoende av vilken teknologi som skall nyttjas. Implementationsmodeller beskriver hur programmet utformas avseende begränsningarna i den valda teknologin. I och med detta är det också eftersträvansvärt att åstadkomma spårbarhet mellan de olika modellerna; exempelvis så att förändringar i en modell får genomslag i relaterade modeller.

En processmodell har till uppgift att visualisera det flöde av aktiviteter, information och resurser som skall förädla något. Det finns ett flertal likartade definitioner av begreppet process. Exempelvis ”en process är en kedja av aktiviteter som återkommande förädlar input till output”. Denna definition av en process anser dock många vara otillräcklig. Ett bättre sätt att beskriva en process är enligt Ljungberg och Larsson (2001) ”En process är ett repetitivt använt nätverk av i ordning länkade aktiviteter som använder information och resurser för att transformera ’objekt’ in till ’objekt’ ut, från identifiering till tillfredsställelse av kundens behov”.

En strukturell modell visar ett systems olika komponenter och hur dessa relaterar till varandra Sommerville (2001). En komponent kan till exempel vara en organisatorisk enhet, ett delsystem eller en befattning i en befattningsstruktur.

Prototyper är modeller av system. Dessa användas främst för att ge användare en möjlighet att påverka utformningen av systemet, innan det färdiga systemet finns tillgängligt. Inom prototyputveckling skiljs på explorativ och evolutionär utveckling. Explorativ prototyputveckling innebär att snabbt ta fram en rad modeller av systemet för att identifiera användarnas behov (Cooper, 2001). Evolutionär prototyputveckling syftar till en gradvis utveckling och modifiering av en modell av det slutliga systemet. Vidare kan prototyper vara horisontella eller vertikala. Med horisontella prototyper ges en överblick av systemet men dessa har begränsad funktionalitet. Med vertikala prototyper realiserar funktionalitet i vissa begränsade delar av systemet (Avison & Fitzgerald, 1995). Prototyper utgörs av pappersmodeller, PowerPoint-bild, webbsidor och mjukvaruapplikationer. Det finns ett antal verktyg som stöder framtagande av prototyper. Att nyttja prototyper ger högre kvalitet på de färdiga systemen, lägre utvecklingskostnad samt högre acceptans och förståelse av systemet från användarna.

5.1.3.6 Genomförande av modellering

För att uppnå höga kvalitativa resultat vid modellering är processen för att utveckla modellerna lika viktig som det modelleringspråk/ramverk som används. Vanliga ansatser för att fånga domänkunskap är; intervjuer med domänexperter, analys av befintlig dokumentation, observation av befintliga arbetssätt och modellering i grupp. (Stirna & Persson, 2007) Vid val av genomförandesätt för modellering måste hänsyn tas till de förutsättningar som finns.

Utveckling av modeller i grupp kan bedrivas utifrån olika utgångspunkter. Konsensusdrivet deltagande (eng: consensus-driven) respektive samrådsdrivet deltagande (eng: consultative) är två varianter (Stirna & Persson, 2007). Konsensusdrivet deltagande baseras på att deltagarna äger modellerna och gemensamt bestämmer/styr över deras innehåll. Samrådsdrivet deltagande baseras på att analytiker/modellörer utvecklar modeller som sedan domänexperter validerar, där utvecklingen av modellerna ofta sker genom att nyttja en annan ansats som intervjuer, deltagande observation med mera. Det finns två huvudsakliga fördelar med ett konsensusdrivet deltagande (Stirna & Persson, 2007):

- Kvaliteten på modellerna blir bättre genom att resultatet är framtaget i konsensus.
- Involverar intressenterna i beslutsprocessen tidigt, som i sin tur underlättar acceptans och engagemang. Detta blir extra viktigt då modelleringen fokuserar på att uppnå en förändring i den aktuella domänen.

Ovanstående kan verka lätt och självklart samt att vem som helst, när som helst, kan genomföra en konsensusdriven modellering. Erfarenheterna visar dock på att ett antal förutsättningar måste vara uppfyllda för att ansatsen med konsensusdriven modellering skall lyckas. (Stirna & Persson, 2007):

- Modelleringsteamet måste få en tydlig uppgift där syftet med modelleringen klart framgår.
- Tillräcklig tid och resurser måste allokeras för både projektgruppen samt övriga deltagare från organisationen.
- Modelleringsteamet måste ha mandat att utveckla eller förändra såväl verksamhetsprocesser som tekniska processer, procedurer, koncept och regler.
- Teamet måste vara välbalanserat i relation till kunskapsområdet som avhandlas.
- Modelleringen bör ledas/stödjäs av erfarna modelleringsledare.

Enligt Stirna och Persson (2007) finns det även andra kriterier som avgör om konsensusdriven modellering är ett lämpligt angreppssätt (Tabell 1).

5.1.3.7 Stöd för modellering

I detta avsnitt beskrivs tre olika typer av stöd för att genomföra modellbaserad utveckling, (1) verktyg, (2) notationer och (3) arkitekturramverk.

5.1.3.7.1 *Verktyg*

En viktig del av angreppssättet vid modellbaserad utveckling är datorbaserat stöd. Systemen som skall utvecklas är komplexa och har relationer och beroenden som är svåröverskådliga. Mängden information som till exempel beskriver kontexten, kraven, strukturen och beteendet avseende det tänkta systemet blir så pass omfattande att det inte är görligt att hantera genom att skriva på papper, framförallt på grund av att systemutveckling i princip alltid bedrivs iterativt. Det innebär att information (t ex krav) som traditionellt sett dokumenteras i tidiga faser kan förändras, tas bort eller kompletteras allteftersom utvecklingen fortskrider. Det är inte heller sannolikt att samma verktyg är lämpligt att använda i alla faser av utvecklingen. Detta ställer krav på att användaren kan överföra hela eller relevanta delar av informationen mellan olika datorbaserade stöd. Det är ofta frågan om stora datamängder. Det viktiga vid val av verktyg är precis som vid systemutveckling att se till att de behov utvecklarna har möts av verktyget.

Det finns även behov av att genomföra analyser av relationer mellan olika informationsobjekt. En vanlig förekommande analys är att spårbart kontrollera att kraven är behandlade i designen. Att hantera detta i form av utskrivna dokument är ofta inte hållbart i längden. Stödet avseende detta kan självklart variera mellan verktygen. Konfigurationsledning är en mycket viktig del av att hantera informationen; det kommer i dessa databaser och verktyg att finnas behov av att frysa informationen/data för att kunna utveckla mot något konkret och fastställt.

Val av verktyg är dock inte enkelt och ofta baserat på tidigare erfarenheter. Individer som använt ett verktyg under många år byter inte frivilligt till ett annat verktyg som klarar av motsvarande uppgifter.

5.1.3.7.2 *Notationer*

Unified Modeling Language (UML) är ett standardiserat modelleringsspråk för visualisering, specificering, konstruktion och dokumentation (OMG, 2001). Ursprungligen utvecklades UML för mjukvaruintensiva system, men har också visat sig vara användbart som grund för att beskriva de flesta typer av system. Språket inkluderar en rad notationer och diagram för att beskriva såväl dynamiska som statiska egenskaper hos systemen och kan tillämpas vid konstruktion av t ex verksamhetsmodeller, strukturmodeller, processmodeller

och informationsmodeller. UML innehåller möjligheten att skapa egna tillägg till notationerna vilket möjliggör anpassning av språkets syntax och semantik. Systems Modeling Language (SysML) är ett generellt tillämpbart modellerings-språk som är en dialekt av UML med utökningar/förändringar för att möta kraven för användning inom disciplinen Systems Engineering (Friedenthal, et al 2008).

5.1.3.7.3 Arkitekturramverk

Arkitekturramverk används för att utveckla arkitekturer som sammanhängande och konsekventa. Ramverk definierar hur modeller skall organiseras och presenteras. För att uppnå detta används någon form av metamodell som bidrar till att med ett enhetligt språk beskriva arkitekturer, ingående element och deras relationer (MODAF, 2009). Det existerar ett antal arkitekturramverk som exempelvis Zachmans (1987), Ministry of Defence Architectural Framework (MODAF), NATO Architectural Framework (NAF), Department of Defense Architectural Framework (DoDAF) och The Open Group Architecture Framework (TOGAF) (Schekkerman, 2003; MODAF, 2009). MODAF är valt av Försvarsmakten som beskrivningsramverk.

MODAF är framtaget för MOD (UK Ministry of Defence) och används som styrning vid arbete med och framtagning av verksamhetsarkitekturer (eng: enterprise architecture). MODAF bidrar med en strukturerad beskrivning och en uppsättning enhetliga uttryck för beskrivning av komplexa arkitekturer. Syftet med ett arkitekturramverk som MODAF är att definiera de komponenter som ingår i en arkitektur, till exempel organisationer, processer och informationsflöden samt dess relationer till varandra. MODAF kan även beskriva till exempel systemarkitekturer där gränssnitt, dataspecifikationer och systemkomponenter ingår samt hur relationerna mellan dessa ser ut. Informationen som presenteras i ett arkitekturramverk som MODAF kallas vyer. Syftet med vyerna är att presentera system- eller verksamhetselement utifrån olika perspektiv. Vad perspektiven skall beskriva beror på vilka av systemets intressenter som informationen är avsedd för; olika intressenter har olika roller och därmed skiftande behov av information om system- eller verksamhetselementen (MODAF, 2009).

5.1.4 Diskussion

Modellbaserad utveckling är vida omskrivet i litteratur om systemutveckling. Det finns många som föreslår olika ansatser för modellbaserad utveckling men generellt gäller att de tar sin utgångspunkt i abstrakta modeller, som är oberoende av teknologi, för att senare med mer konkreta modeller realisera systemen. Därigenom finns en vision om att förändringar i den teknikberoende modellen skall få genomslag i alla efterföljande modeller. De flesta processerna är också iterativa för att modellerna successivt skall kunna utvecklas vidare. De allra flesta nyttjar UML i någon form som notation för modelleringen. Flera påtalar

vikten av att kvalitetssäkra de framtagna modellerna under utvecklingsprocessen. Metamodeller anses vara av vikt för att uppnå konsistens mellan modeller och möjlighet att automatiskt transformera en modell till en annan typ av modell. Alternativt att automatiskt kunna validera modeller mot varandra.

Det finns ett behov av adekvata verktyg för att kunna genomföra effektiv modellbaserad utveckling. Det finns alltför få vetenskapliga studier avseende erfarenheter av att tillämpa MODAF.

5.2 Fallstudie: Kvalitetsdriven kravhantering för utveckling av insatsledningssystem

(Denna fallstudie presenteras mer utförligt i Hallberg, N., Pilemalm, S. & Timpka, T. (2010) Quality Driven Requirements Engineering for Development of Crisis Management Systems. Working paper.)

Inom ledning av insatser finns det behov av olika former av stöd, som informationssystem kan tillfredsställa. Utveckling av system för så komplexa verksamheter är dock svårt. Andelen framgångsrika systemutvecklingsprojekt är låg, till stor del orsakad av bristfällig kravhantering. En stor utmaning ligger i att identifiera användarnas behov och att uttrycka dessa som krav. Denna fallstudie presenterar en ansats för att identifiera krav avseende ledningssystem för krishantering. Ansatsen är baserad på Quality Function Deployment (QFD), användningsfall, prototyper och scenario utvärderingar. Den föreslagna ansatsen genomförs i tio aktiviteter: (1) Datainsamling, (2) Identifiera utsagor, (3) Fastställa behoven, (4) Analys och strukturering av behov, (5) Prioritering av behov, (6) Utveckling av användningsfall, (7) Analys av krav, (8) Utveckling av scenarier och prototyper, (9) Utvärderingar av användningsfall och (10) Revidering av användningsfall och krav. Ansatsen tillämpas och utvärderas i projekt avseende tekniskt beslutsstöd för insatser i urban miljö. Resultatet av studien förmedlar hur slutanvändare kan medverka i kravhanteringsprocessen samt att kundrösttabellen är ett användbart verktyg för att tolka användares utsagor och formulera dessa som behov.

5.2.1 Inledning

Ledning av insatser vid kriser är kritiska och komplexa aktiviteter, där misstag kan få förödande konsekvenser. Informationssystem har förmåga att stödja ledningen av krishantering för att, exempelvis, tillhandahålla lägesbild, stödja beslutsfattande, samordna de operativa insatserna och stödja informationsutbyte (Jefferson, 2006). Att införa informationssystem leder inte per automatik till effektivare insatser. Dessa system måste uppfylla en rad kriterier för att vara användbara, exempelvis måste informationssystemen vara tillräckligt flexibla för att stödja det kreativa och kontinuerliga beslutsfattandet. (Mendonca, Jefferson & Harrald, 2007).

Informationssystem har potential att stödja de flesta typer av verksamheter att nå högre effektivitet, högre kvalitet av producerat resultat och öka samverkan mellan involverade aktörer. Men svårigheterna med att utveckla informationssystem är välkända (Wiegers, 2003, Kasser, 2007). Young (2001) hävdar att omkring 80 % av alla brister i informationssystem har sitt ursprung i de tidiga

faserna. Utmaningar till att nå ökad kvalitet hos utvecklade system innefattar även att kunna kommunicera med användarna för att identifiera de behov som är viktigast att tillfredställa (Kasser, 2007). För detta krävs notationer och metoder som ger användarna möjlighet att aktivt delta i utvecklingen. En ytterligare utmaning är spårbarhet (Hull, Jackson, & Dick, 2005; Daneva, 2003).

Denna fallstudie utvärderar en ansats för kravhantering, Kvalitetsdriven kravhantering (eng: Quality Driven Requirements Engineering, (QRE)) för att identifiera krav på ledningssystem vid insatser för exempelvis krishantering. Metoden bygger på Quality Function Deployment (QFD), och omfattar användningsfall, användarmedverkan, prototyper och scenarier utvärderingar. Syftet är att tillhandahålla Försvarmakten kunskap gällande en ansats för specificering och validering av krav och funktionalitet avseende insatsledningssystem.

5.2.2 Metod

Fallstudien studien utfördes i tre steg: (1) en teoretisk justering av QRE med ett argumenterande konstruktionstänkande (Rittel, 1973), (2) tillämpning av den justerade QRE i projektet avseende teknisk beslutsstöd för insatser i urban miljö (Ahlberg, et al 2006), och (3) en utvärdering av de erfarenheter som gjorts.

5.2.3 Resultat

Detta avsnitt beskriver enbart den resulterade metoden och de erfarenheter som erhöles då den tillämpades. Metoden Kvalitetsdriven kravhantering finns beskriven i sin ursprungliga form i Hallberg, Andersson, och Westerdahl (2005).

QRE består av tio steg, (1) Datainsamling, (2) Identifiera utsagor, (3) Identifiera behov, (4) Analys och strukturering av behov, (5) Prioritering av behov, (6) Utveckling av användningsfall, (7) Analys och strukturering av krav, (8) Utveckling av scenarier och prototyper, (9) Utvärderingar av användningsfall och (10) Revidering av användningsfall och kraven. Varje steg i QRE presenteras nedan med dess syften, hur de genomfördes i studien, resultaten, och de erfarenheterna.

Steg 1: Datainsamling

Syfte: Att samla in information som skall användas för att identifiera behov.

Genomförande: Information samlas in om t.ex. användarna, verksamheten, verksamhetsmål, idéer om framtida system och användarnas erfarenheter. I fallstudien utfördes en dokumentanalys, omfattande tre rapporter som beskriver problem, behov av stöd, krav och tekniska lösningar på internationella insatser i urbana miljöer. Rapporterna innehöll resultatet av intervjuer, observationer och

diskussioner med användarrepresentanter som gjorts under förstudien till projektet (Ahlberg, et al 2006).

Resultat: Beskrivningar i form av rapporter.

Erfarenhet: Det hade varit bättre med en direkt kontakt mellan systemutvecklare och företrädare för användare men den korta tidsramen för behovs- och kravanalys tillät inte detta. Information som fanns tillhanda innehöll dock utförliga beskrivningar, framtagna av olika forskare och som presenterade fenomen utifrån ett flertal olika perspektiv.

Steg 2: Identifiera utsagor

Syfte: Att identifiera utsagor som innehåller information är av betydelse för den fortsatta utvecklingen.

Genomförande: Noggrann genomläsning av den insamlade informationen och identifiering av de delar av texten som kan användas för att identifiera behov. I fallstudien kopierades utsagor från de rapporter som identifierades i Steg 1 och som samlats i ett dokument. En övergripande kategorisering gjordes av uttalanden, där liknande utsagor placerades tillsammans.

Resultat: En lista med utsagor. I studien identifierades 80 utsagor som relevanta för den fortsatta analysen. Storleken på de utsagor som identifierades varierar från del av mening till tre meningar.

Erfarenhet: Denna aktivitet ansågs vara relativt rakt på sak. Det svåra är dock att välja utsagor av lämplig omfattning. Utsagorna skall vara begripliga men samtidigt inte alltför omfattande så att de täcker ett flertal behov.

Steg 3: Identifiera behov

Syfte: Att ur utsagorna identifiera behov.

Genomförande: Kundröstattabellen (eng: Voice of the Customer Tabell (VCT)) används som stöd för att analysera de identifierade utsagorna genom att betrakta dem utifrån olika perspektiv (Tague, 2005). I fallstudien utfördes VCT-analysen i MS Word.

Resultat: En uppsättning ostrukturerade och delvis överlappande behov. I studien identifierades 128 behov.

Erfarenhet: Att utföra analys med stöd av VCT är ett tidskrävande arbete. Det visade sig dock vara en klar fördel att kategorisera utsagorna i Steg 2. Detta eftersom liknande utsagor ofta speglade samma eller liknande behov, fast ur olika perspektiv, vilket underlättade tolkningen av utsagorna. Vikten av att välja utsagor av lämplig storlek blev uppenbar, då omfattande utsagor tenderar att innehålla irrelevant information, medan uttalanden som är för smala gör det omöjligt att få fram behov utan att återvända till källan. Dessutom blir spårbarhet

tydligare om en utsaga enbart ger upphov till ett behov. Spårbarhet till källan är nödvändig när tolkningen av utsagorna är tvetydig.

Steg 4: Analys och strukturering av behov

Syfte: Kategorisera och strukturera behoven för att öka kvalitén på formuleringarna och skapa överskådlighet.

Genomförande: Liknande behov grupperas till kategorier. Därefter grupperas kategorierna samman med relaterade kategorier och bildar nya kategorier av en högre abstraktionsnivå tills en sammanhängande struktur erhålls. Efterhand tas dubletter bort och formuleringar av behov förbättras i en iterativ process tills kvaliteten i uppsättningen av behov anses tillfredsställande. Studien genomfördes i MS Word, där behov grupperades tillsammans och med olika nivåer av rubriker skapades en hierarki. En systemutvecklare nyttjades för att granska resultatet, vars synpunkter sedan användes för att förbättra resultatet ytterligare.

Resultat: En uppsättning strukturerade behov. I studien erhöles 14 kategorier, på högsta nivå, och totalt 97 behov.

Erfarenhet: Med en stor skärm eller en projektor är det möjligt att arbeta datorbaserat med kategoriseringen. Traditionellt brukar denna typ av aktivitet genomföras som en gruppbaserad brain-stormingsaktivitet med Post-it-lappar. Omfattande tid sparades dock på att slippa överföra behoven till Post-it-lappar och sedan gå tillbaka till någon form av datorstöd. För granskning fungerade MS Words funktion "Spåra ändringar" alldeles utmärkt. Användningen av MS Word gjorde det enklare att spara resultatet, jämfört med stora pappersark vilka brain-storming aktivitet ofta resulterar i. Att inleda varje behov med en enhetlig formulering fanns vara nödvändigt för att kunna använda sorteringsfunktionen i MS Word, och för att identifiera dubletter och lättare kunna gruppera liknande behov. I studien användes initial formuleringar i form av "Kunskap om ..." och "Stöd till ...".

Steg 5: Prioritering av behoven

Syfte: Att identifiera de behov som är av störst vikt att tillfredställa.

Genomförande: De identifierade behoven prioriterades efter hur viktiga de ansågs vara baserat på skalan *Mycket viktigt*, *Viktigt* och *Låg vikt*. Denna prioritering skall helst göras av någon med en omfattande domän kunskap. I studien ombads projektledningen att rangordna de 97 behoven, baserat på hur viktiga de ansåg dem vara. Projektledningen ombads att enbart klassa ett totalt behov som *Mycket viktiga*, vilka skulle ligga till grund för att skapa användningsfall.

Resultat: Den första prioriteringen av de 97 behoven resulterade i att 50 gavs med högsta prioritet. Efter en omprioritering av dessa 50, valdes 22 behov ut att ligga till grund för utveckling av användningsfall.

Erfarenhet: Prioriteringen upplevdes betydligt mer tidskrävande och svårare än väntat. Det konstaterades att det var svårt att *välja bort* behov.

Steg 6: Utveckling av användningsfall

Syfte: Att ta fram användningsfall, baserade på de prioriterade behoven.

Genomförande: En mall användes för att skapa användningsfall baserat på behoven. En systemutvecklare skapade användningsfallen och en annan systemutvecklare validerade dessa mot behoven och förslog förbättringar. En enhetlig början av alla krav användes; som i detta fall utgjordes av "systemet skall ...".

Resultat: 22 användningsfall utvecklades. I dessa beskrevs 99 funktionskrav och 81 icke-funktionella krav.

Erfarenhet: Användning av en fastställd mall upplevdes mycket användbar för att utveckla användningsfall. Att använda en systemutvecklare som extern granskare av användningsfallen visade sig väsentligt öka deras kvalitet. Flera förbättringar genomfördes av användningsfallen, speciellt då det gäller de identifierade kraven.

Steg 7: Analys och strukturering av krav

Syfte: En strukturerad uppsättning krav baserade på användningsfall.

Genomförande: Kategoriindelning utfördes med hjälp av MS Word genom att gruppera liknande krav och med flera nivåer av rubriker skapas hierarkier.

Resultat: En strukturerad uppsättning krav.

Erfarenhet: Struktureringen av kraven som ger en överblick av vilka krav som bör tillfredställas. Vidare var den enhetliga inledningen av formulering av krav användbar för sortering funktionerna i MS Word, för att ta bort dubletter och för att gruppera liknande krav.

Steg 8: Utveckling av scenarier och prototyper

Syfte: Att konstruera scenarier och en prototyp som återspeglar de valda användningsfallen.

Genomförande: Fem av användningsfallen valdes ut av projektledningen för att utvärderas. För att illustrera de funktioner som användningsfallen representerade, togs scenarier och en prototyp fram och realiserade i MS PowerPoint. Två interna utvärderingar genomfördes för att säkerställa att scenarierna och prototypen inte skulle innehålla uppenbara felaktigheter och inkonsekvenser. I dessa utvärderingar deltog projektmedlemmar med militära, tekniska och människa-system interaktions kompetenser. Resultaten av utvärderingarna användes för att finjustera scenarierna och prototyperna.

Resultat: Ett ramsscenario som presenterade förutsättningarna, fem miniscenarier som presenterade funktionaliteter representerande envar av de fem utvalda användningsfallen.

Erfarenhet: De utvärderingar som gjorts under utvecklingen visade sig viktiga. Speciellt människa-system interaktion och militär kompetens medförde en väsentlig förbättring av scenarier och prototyper.

Steg 9: Utvärderingar av Användningsfallen

Syfte: En formativ utvärdering av användningsfallen och därmed den funktionalitet som dessa representerade.

Genomförande: Scenariona utvärderades under en fem timmar lång session. I utvärderingen deltog tre användarrepresentanter, samtliga utbildade för insatser i internationella miljöer. Fyra systemutvecklare deltog och presenterade scenariona, hanterade prototypen, ledde diskussionen och förde anteckningar. Först presenterades ramsscenario. Därefter presenterades var och ett av de fem miniscenarierna, med stöd av prototypen, följt av en diskussion. Under diskussionen ombads användarrepresentanterna att ge sin syn på scenarierna och prototypen baserat på följande frågor.

- Skulle de presenterade funktionerna vara tillämpliga i situationer som scenariot illustrerade?
- Finns det andra situationer där de redovisade funktionerna skulle vara nyttiga?
- Hur skulle en mer trolig användningssituation se ut?
- Finns det andra funktioner och annan information som skulle vara användbar i situationer motsvarande scenariot?

Resultat: Information som kan användas för att förbättra användningsfallen. Ytterligare ett stort antal iakttagelser om hur systemen för detta sammanhang bör utformas kom upp under utvärderingen. Viktigaste egenskaper är att systemen måste vara enkla att använda och information snabbt måste kunna spridas till de berörda. Det ansågs att stora positiva effekter skulle uppnås med en automatiserad administration avseende materialhantering, vilket skulle frigöra personal för direkta uppgifter avseende hantering av den givna situationen. De användningsfall som utvärderades ansågs återspegla användbara funktioner.

Erfarenhet: Den power-point-baserad prototypen upplevdes väl illustrera olika funktioner. Under utvärderingen förslogs många möjligheter hur systemen för detta sammanhang skall utformas. Till exempel att de måste vara enkla att använda och att informationen snabbt måste meddelas till de berörda. Det är viktigt att efter en utvärdering vara beredd på att ta hand om även de föreslagna förbättringarna som inte alltid är direkt kopplande till de användningsfall som utvärderas.

Steg 10: Finjustering av Användningsfall och kraven

Syfte: Att uppdatera användningsfall och krav.

Genomförande: Anteckningar och inspelningar från utvärderingen analyserades med avseende på tre frågor: (1) information som kan användas för att förbättra användningsfallen, (2) ytterligare funktioner för hantering av situationer i stadsmiljö och (3) andra aspekter som är viktiga för att ta hänsyn till vid utveckling av ledningssystem för användning i stadsmiljö. Resultatet användes för att förbättra användningsfallen och kraven.

Resultat: Validerade användningsfall och krav.

Erfarenhet: Fyra av de fem användningsfallen genomgick omfattande omarbetningar utifrån resultatet av utvärderingen. Det var dock främst förbättringar och tillägg av mer information. Utvärderingen upplevdes väsentligen leda till förbättrade och mer detaljerade användningsfall.

5.2.4 Diskussion

Utveckling av informationssystem lider ofta av bristfällig kravhantering och kommunikation mellan systemutvecklare och användarrepresentanter (Kasser, 2007). Denna fallstudie presenterar en ansats för kravidentifiering som bygger på en gedigen behovsanalys, spårbarhet från källor till krav, samt stöd för att validera funktioner och krav av användarrepresentanter. Utan en noggrann behovsanalys kommer resterande utveckling att tvingas bygga på gissningar. Det finns dock en brist på förståelse av vikten av detta samt en bristande förståelse av att behovsanalys inte är triviala. Syftet med fallstudien var inte att göra en fullständig behovsanalys eller kravspecifikation, utan att utvärdera ansatsen för Kvalitetsdriven kravhantering.

Att ha tillgång till användarrepresentanter är viktigt, eftersom omfattande omarbetning härigenom kan undvikas, särskilt när det gäller tolkningen av utsagor vid analysen för att identifiera de verkliga behoven (Hallberg, Timpka, & Eriksson, 1999). Vidare måste utvecklingen baseras på de faktiska och mest framträdande behoven (Ibid). Det vill säga, det är viktigt att prioritera behoven och inte vänta med prioriteringen till kravspecificeringen. Att validera så väl behov som krav är viktigt. Fallstudien visade att varje utvärdering påtagligt förbättrar kvalitén. Det är svårt för användarrepresentanter att bedöma traditionella kravspecifikationer. I fallstudien nyttjades en prototyp och ett antal scenarier för att kommunicera kring krav med användarrepresentanter. Studien visade på att detta fungerade effektivt för att validera användningsfall, och därmed också kraven. Deltagarna som medverkade i utvärderingen kom snabbt in i situationen och bidrog med värdefulla kommentarer. Det hade säkerligen varit mycket svårare för dem att yttra sig, om enbart en lista av krav lagts fram.

5.3 Fallstudie: Kvalitetsdriven kravhantering för utveckling av avvikelshantering

Att specificera system så dessa erhåller rätt funktionalitet är en av de mest kritiska aktiviteterna inom systemutveckling. För att kontinuerligt utveckla en organisation/verksamhet, så som Försvarsmakten, är det centralt att kunna hantera avvikelser, vilket kan stödjas med olika former av IT-system. Inom Försvarsmakten används ett antal olika IT-system för avvikelshantering. Det finns dock en ambition att införa ett gemensamt system. För att uppnå denna ambition har Försvarsmakten upphandlat ett system vilket varit i pilotdrift under 2009. I denna fallstudie studeras tillämpningen av en ansats för kravhantering, benämnd *Kvalitetsdriven kravhantering*. Ansatsen tillämpas för att identifiera behov av IT-system för avvikelshantering inom Försvarsmakten.

Fallstudien genomfördes i tre steg; (1) anpassning av ansatsen för att passa utveckling av ett IT-stöd för verksamhetsfunktion, (2) tillämpa ansatsen och dokumentera gjorda erfarenheter samt (3) sammanställa och presentera resultatet.

Metoden befanns fungera tillfredställande. Processmodellen upplevdes som ett bra stöd då användningsfall skapades. Totalt 430 behov identifierades. Dessa innefattade varför Försvarsmakten behöver avvikelshantering och vilket stöd de som är involverade i avvikelshanteringen behöver.

5.3.1 Inledning

Trots att IT-system inte längre är någon ny företeelse så är fortfarande andelen misslyckade IT-projekt relativt hög (Kasser, 2007). De vanligaste bristerna, och samtidigt de som är mest kostsamma att korrigera, är de som görs i de tidiga faserna av utvecklingsprocessen (Young, 2001). En gedigen behovsanalys är därmed inte bara ett underlag för vidare utveckling, utan också viktigt för att motivera varför system och systemfunktioner skall införas. Att hantera avvikelser³ är en central ledningsfunktion för de allra flesta verksamheter. Avvikelshantering utgör grunden för en kontinuerlig utveckling, exempelvis genom att förhindra tillbud och att anpassa verksamheten för att motsvara nya förutsättningar.

I denna fallstudie studeras tillämpningen av en ansats för kravhantering, Kvalitetsdriven kravhantering, vid utveckling av IT-stöd för verksamhetsledningsfunktioner och avvikelshantering (Hallberg, Andersson, & Westerdahl, 2005; Hallberg, Ölvander, & Törne, 2006). Försvarsmakten har ett stort antal IT-

³ Avvikelse är ett icke-uppfyllande av ett krav (SS-EN ISO 9000:2005)

applikationer som stöd för sin verksamhetsledning. Syftet med denna fallstudie är att tillhandahålla Försvarmakten kunskap gällande en ansats för behovsanalys och specificering av krav avseende ledningssystem för verksamhetsledning. I fallstudien tillämpas ansatsen för att identifiera behov av ett IT-system för avvikelshantering inom Försvarmakten.

5.3.1.1 Bakgrund

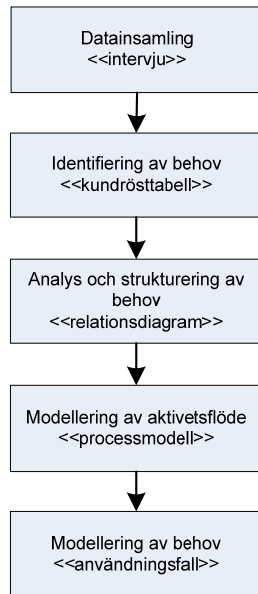
Inom Försvarmakten nyttjas ett flertal olika IT-system för avvikelshantering. Det finns dock planer på att införa ett gemensamt IT-system. Den bakomliggande orsaken till detta är bland annat behovet av att kunna sammanställa statistik avseende vilka avvikelser och typer av avvikelser som förekommer samt att kunna identifiera systematiska avvikelser. Under 2006 togs en kravspecifikation avseende IT-system för hantering av avvikelser, förbättringsförslag och erfarenheter fram. Kravspecifikationen låg till grund för upphandlingen av ett IT-system under 2007. Systemet har dock ännu inte tagits i bruk men i början av 2009 initierades en pilotdrift på tre pilotförband. Under pilotdriften har ett antal frågor uppkommit vilka behöver besvaras innan systemet tas i drift.

Det projekt inom vilket fallstudien genomföres syftade till att utföra en behovsanalys för att studera hur ett IT-system skall stödja Försvarmaktens avvikelshantering. Behovsanalysen belyser vilka behov som finns inom olika verksamhetsgrenar, hur de kan stödjas av ett system, vilka roller som ingår i hanteringen av avvikelser samt vilka uppgifter de olika rollerna har. Behovsanalysen skall ligga till grund för ett beslutsunderlag om huruvida Försvarmakten skall införa ett gemensamt avvikelshanteringssystem.

5.3.2 Metod

Fallstudien genomfördes i tre steg; (1) anpassning av ansatsen Kvalitetsdriven kravhantering för att passa utveckling av ett IT-stöd för verksamhetsfunktion, (2) tillämpa ansatsen och dokumentera gjorda erfarenheter samt (3) sammanställa och presentera resultatet.

Kvalitetsdriven kravhantering tillämpades i projektet i fem steg; (1) datainsamling, (2) identifiering av behov, (3) analys och strukturering av behov, (4) modellering av aktivitetsflöde och (5) modellering av behov (Figur 2).



Figur 2. Beskrivning av hur behovsanalysen genomfördes.

Inledande förberedelser för fallstudien var ett projektmöte där bakgrund och uppdrag föredrogs. En projektplan framarbetades och fastställdes. Tillämpning av Kvalitetsdriven kravhantering i fallstudien genomfördes enligt nedan.

Steg 1 – Datainsamling: En lista med intressenter ur pilotförbanden samt andra intressenter med betydande roller avseende avvikelshantering inom Försvarmakten togs fram. En intervjuguide skapades som mall för de intervjuer som skulle genomföras. Intervjuerna genomfördes och dokumenterades.

Steg 2 – Identifiering av behov: Utifrån intervjuerna identifieras utsagor vilka sammanställdes i en kundrösttabell varifrån sedan utsagorna omvandlades till behov.

Steg 3 – Analys och strukturering av behov: Behoven som identifierats utifrån utsagorna analyserades och strukturerades i ett relationsdiagram.

Steg 4 – Modellering av aktivitetsflöde: Ett avvikelseärendes hantering modellerades genom att beskriva ett avvikelseärendes gång, vilka aktiviteter som utförs samt vilka personer som genomför dessa.

Steg 5 – Modellering av behov: Användningsfallsdiagram och användningsfallsbeskrivningar skapades för att beskriva vilka aktiviteter som behövs för att handha avvikelseärenden. De aktivitetsflöde som skapades i Steg 4 ligger till grund för detta steg.

5.3.3 Resultat

Resultatet av fallstudien beskrivs baserat på de fem steg som behovsanalysen genomfördes med avseende på (1) hur ansatsen tillämpades, (2) resultatet av steget samt (3) gjorda erfarenheter.

5.3.3.1 Datainsamling – intervjuer

Tillämpning: En intressentanalys genomfördes för att fastställa vilka intressenter som vara relevanta att intervjua avseende avvikelshantering. En intervjuguide togs fram för att ge intervjuaren riktlinjer för intervjuerna samt säkerställa att rätt områden belystes under intervjun. Intervjuguiden delades in i sju huvudavsnitt; (1) bakgrund, (2) avvikelshantering, (3) problem vid avvikelshantering, (4) styrkor i dagens avvikelshantering, (5) framtida avvikelshantering, (6) frågor kring pilotstudie samt (7) avslutning. Därefter genomfördes intervjuerna på plats hos intressenterna (bortsett från en som genomfördes via konferenstelefon) med diktafon och stödanteckningar som hjälpmedel. Intervjuerna transkriberades inte utan nedtecknades i sin helhet, dock inte ordagrant.

Utfall: Nio intressenter fastställdes som relevanta och valdes ut för intervju. Av dessa intervjuer visade sig sex möjliga att genomföra, då tiden ej ansågs räcka till för några ytterligare intervjuer. Den avsatta tiden för varje intervju var två timmar.

Erfarenhet: En diskussion om valet av datainsamlingsmetod, enkäter eller intervjuer, uppstod. Fördelen med enkäter ansågs vara mindre tidsåtgång men nackdelarna var att frågorna måste vara mer precist utformade och att det inte finns utrymme för att utveckla svaren. Möjligheterna till att föra en diskussion med de intervjuade bedömdes ge mer indata och därmed ett fördelaktigare resultat. Den avsatta tiden på två timmar var mer än väl tilltagen. De flesta intervjuer varade runt 70-80 minuter, även om några ville ställa frågor och vidareutveckla sitt resonemang efter att diktafonen stängts av.

Intervjuguiden var utformad för att inte föra in respondenterna på pilotstudien, utan för att beskriva avvikelshandlingen inom deras område såsom den fungerar idag (med fördelar och nackdelar) samt vilka behov och förbättringar som är tänkbara. Respondenterna ombads även beskriva hur de ville att en framtida avvikelshantering skall fungera. Detta visade sig vara svårt för de respondenter som varit involverade i pilotstudien, då de ofta återkopplade till det system de testat. I deras beskrivningar av framtida avvikelssystem beskrevs ofta ett system som liknade det system som varit aktuellt under pilotstudien med dess funktioner och flöde. Analysen av intervjumaterialet blev således någon mån inriktat mot pilotsystemet och de behov som framkom måste beaktas med en viss försiktighet då de inte är helt objektiva.

Transkribering ansågs inte nödvändig då identifieringen av behov tas ur flera intervjuer som lagts samman till en gemensam behovsbeskrivning. De enskilda

respondenternas exakta formuleringar blir då av mindre betydelse. Det viktiga är att texten innehåller korrekta tolkningar av respondenternas utsagor.

Att välja ut respondenter visade sig mer tidskrävande än förväntat. I framtida tillämpningar av Kvalitetsdriven kravhantering bör det övervägas om inte detta bör vara en egen aktivitet som föregår datainsamlingen.

5.3.3.2 Identifiering av behov – kundrösttabell

Tillämpning: För att identifiera de behov som respondenterna framhållit i intervjuerna skapades kundrösttabeller (Tabell 1). En kundrösttabell innehåller respondentens utsaga, vem (som har behovet), vad (behövs det till), när (vilket tillfälle), var (fysisk plats), varför (finns behovet), hur (deras lösning) och behovet. Utifrån intervjuerna hämtas utsagan ur vilken behoven identifierades. En utsaga är några meningar vilka skall belysa något behov som kunden har. Utsagan bör inte vara för lång men måste ändå ge ett visst sammanhang för att kunna ge möjlighet till att identifiera behovet med stöd av de olika frågorna (vem, vad, när, var, varför, hur). Den slutgiltiga formuleringen av behovet skall inte beskriva lösningar. Tabellerna granskades och reviderades av olika personer för att skapa ett så utförliga kundbehov som möjligt.

Tabell 1: Exempel på Kundrösttabell.

Utsaga	Vem	Vad	När	Var	Varför	Hur	Behov
Säkinsp ställer krav på att Försvarsmakten har ett fungerande avvikelshanteringssystem, även för mark-säkerhet, vilket inte finns idag.	FM	Rapportera olyckor, tillbud, iakttagelser	Vid avvikelser		Efterfölja Säkinsp krav.	RMM	Stöd med att hantera avvikelser på ett fungerande sätt. Uppfylla krav avseende hantering av avvikelser

Utfall: Varje intervju resulterade i en kundrösttabell om cirka 50 utsagor.

Behoven ligger till grund för resterande arbete med att utforma ett beslutsunderlag. Således skapar en noggrann analys goda förutsättningar för ett korrekt beslutsunderlag.

Erfarenhet: Då flera respondenter varit med i pilotstudien och säger sig ha behov av ett IT-system, visar det sig att flera av de behov som framkommit är direkt kopplade till den genomförda pilotstudien.

Det finns en stor spridning av behov inom de olika verksamhetsgrenarna samt vilka egenskaper ett IT-system bör ha. I utsagorna förekommer det vissa motsägelser, dels ifrån samma respondent men även också mellan de olika respondenterna. Motsägande behov liksom allt för spretiga behov kan givetvis inte tillgodoses utan att samtliga parter enas om ett gemensamt behov.

5.3.3.3 Analys och strukturering av behov – relationsdiagram

Tillämpning: För att kunna analysera behoven strukturerades dessa i ett relationsdiagram. I anslutning till varje behov markerades från vilken intervju de härstammade för att på så sätt uppnå spårbarhet. Behoven beskrevs efter en gemensam form och sorterades för att kunna påvisa vilka behov som uppkom flera gånger. Redundanta behov slogs samman och liknande behov beskrevs bara en gång. Behoven samlades i kategorier med liknande behov och underkategorier skapades. Arbetet utfördes i grupp och diskussioner, om vilka kategorier som skulle skapas samt till vilka kategorier behoven skulle hänföras, genomfördes.

Utfall: Då samtliga kundrösttabeller var kompletta sammanfogades behoven till en lista (430 behov hade identifierats). När all kategorisering var klar hade två olika nivåer av kategorier skapats, en nivå för vad verksamhetsområdena har för behov och vilka behov de har av systemet, samt en nivå för de behov Försvarsmakten har av ett avvikelshanteringssystem.

Behoven för verksamhetsområdena delades in i fem huvudnivåer; (1) hantering av avvikelser, (2) rutiner, (3) rättigheter, (4) information samt (5) utbildning.

Försvarsmaktsnivån indelades i sju huvudnivåer; (1) styrning, (2) mål, (3) kvalitetsfrämjande verksamhet, (4) utbildning, (5) arbetsmiljö och värdegrundsrelaterade, (6) IT-system samt (7) definition av avvikelse.

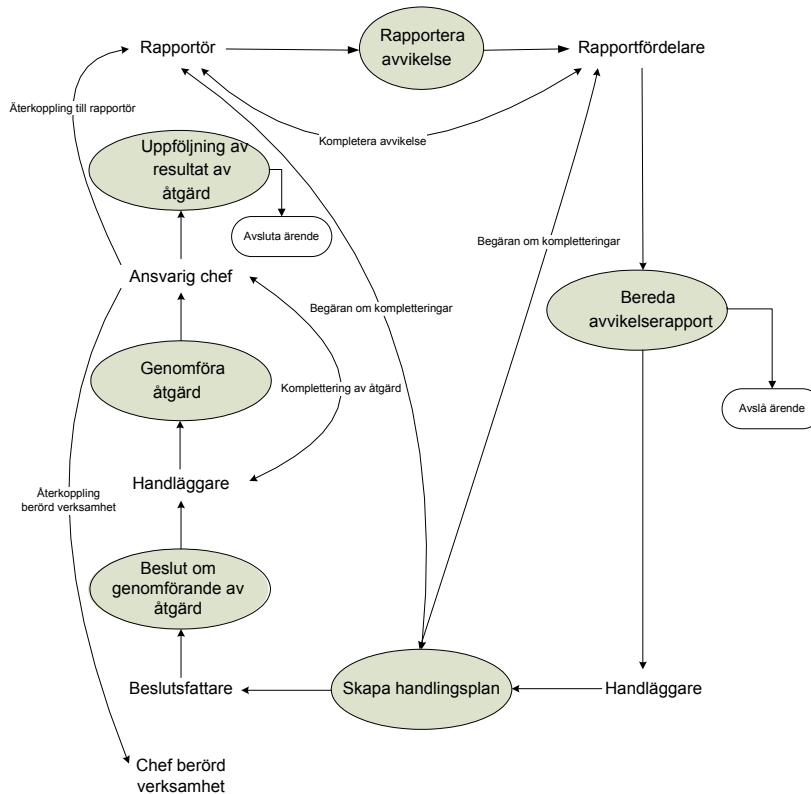
Erfarenhet: Genom att försöka beskriva behoven med samma struktur och sortera dem var det möjligt att eliminera eller slå samman en annars ohanterlig mängd behov. Uppdelningen av behov i kategorier och underkategorier var tidskrävande men resultatet gav en mer överskådlig lista än när alla behov var samlade i kundrösttabeller. För att klara av att ta tillvara på samtliga behov som identifierats var det nödvändigt att skapa en nivåindelning. En hel del av de behov som identifierats berör inte avvikelshanteringssystem i sig utan berör till exempel Försvarsmaktens arbetssätt eller dokument som styr avvikelshantering.

5.3.3.4 Modellering av aktivitetsflöde

Tillämpning: För att skapa en förståelse för hur avvikelshantering inom Försvarsmakten skall ske på en övergripande nivå skapades en processmodell

(Figur 3). Modellen beskriver hur avvikelseärenden hanteras från det att en rapportör rapporterar ett ärende, via beslut om åtgärd, genomförande av åtgärd samt återkoppling till rapportören.

Utfall: Processmodell som beskriver aktivitetsflödet i avvikelshantering innefattande aktiviteter och aktörer.



Figur 3. Processmodell.

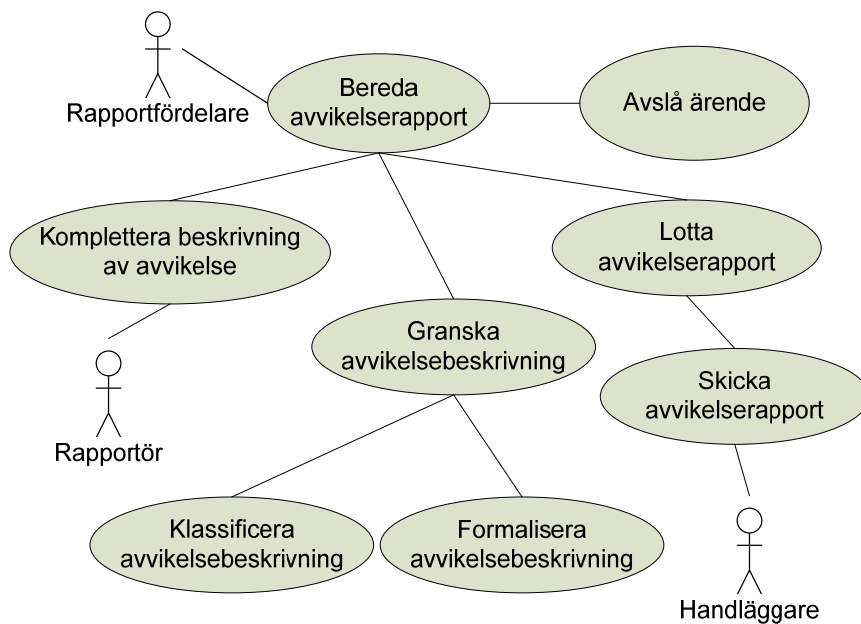
Erfarenhet: Att skapa en processmodell ansågs nödvändigt för att få en gemensam förståelse för en önskvärd avvikelshantering enligt vad som beskrevs i intervjuerna. Samtidigt som modellen utvecklades så identifierades ett antal roller som skulle komma vara involverade i avvikelshantering.

5.3.3.5 Modellering av användningsfall

Tillämpning: För att illustrera och konkretisera behoven modellerades dessa med hjälp av användningsfall. Användningsfallen beskriver vilka aktörer som interagerar med systemet samt vilka aktiviteter de utför. Användningsfallen beskrivs med hjälp av användningsfallsbeskrivningar samt

användningsfallsdiagram. Användningsfallsbeskrivningarna beskrivs i en tabell. Tabellen innehåller en kortfattad beskrivning av användningsfallet, vilken aktör som initierar användningsfallet, vem som har användning av användningsfallet samt vilka aktörer som stödjer användningsfallet. Funktionella och icke-funktionella krav på systemet beskrivs samt en referens till vilka behov som användningsfallet realiserar. Användningsfallsdiagrammet är en visualisering av användningsfallsbeskrivningen. Diagrammet presenterar användningsfallet visuellt, genom att visa vilka aktörer som deltar och de aktiviteter som inträffar.

Utfall: Elva användningsfall skapades, vilka beskrevs med användningsfallsdiagram och användningsfallsbeskrivningar. Tabell 2 och Figur 4 illustrerar en användningsfallsbeskrivning och ett användningsfallsdiagram vilka påvisar hur en rapportfördelare får en avvikelserapport från en rapportör och hur denne bereder en avvikelserapport för vidarebefordran till en handläggare. Rapportfördelaren kan begära kompletterande beskrivning från rapportören om denne anser att avvikelserapporten är för vag eller på annat sätt ofullständig. Rapportfördelaren granskar, klassificerar och formaliserar avvikelserapporten innan denne lottar den vidare till en ansvarig handläggare. Rapportfördelaren har även möjlighet att avslå ärendet om denne anser att ärendet inte är en avvikelse eller om det på annat sätt är irrelevant.



Figur 4. Exempel på användningsfallsdiagram.

Erfarenhet: Processmodellen var en bra utgångspunkt vid utformandet av användningsfallen. De roller som användes och de centrala händelserna var definierade i aktivitetsflödet. Utformandet av vilka aktiviteter som var nödvändiga för varje enskild händelse var tidskrävande och diskuterades ingående. Att utforma funktionella krav på systemet för vissa aktiviteter var både komplicerat och svårt, till exempel vilket stöd skall systemet ge rapportfördelaren vid lottning.

Tabell 2: Exempel på användningsfallsbeskrivning.

Identifierare:	UC 1.2 (ver: 0.5)
Namn:	Bereda avvikelserapport
Kortfattad beskrivning:	Aktören granskar den inkomna rapporten och kompletterar vid behov. Aktören klassificerar rapporten innan denna lottas vidare till ansvarig enhet/handläggare.
Nytta:	Handläggaren får en beskrivning av en avvikelse samt stöd för att formulera en avvikelserapport.
Behov som motiverar:	1.2 Beredning av avvikelser 1.2.1 Rapportfördelning 3 Rättigheter
Primära aktörer:	Rapportfördelare
Secondära aktörer:	Handläggare
Stödjande aktörer:	
Användningsfrekvens:	
Förutsättningar:	
Funktionella krav:	
Icke-funktionella krav:	
Författare:	Niklas Hallberg, Joachim Hansson
Datum:	2009-10-12

5.3.4 Diskussion

En förutsättning för att utveckla system med precision och kvalité är insikt i verksamheten och de blivande användarnas situation. Förståelse för de behov som finns är viktigt för att kunna motivera införskaffandet och införandet av system. I denna fallstudie beskrivs nyttjande av Kvalitetsdriven kravhantering för att genomföra en behovsanalys avseende IT-stöd för en verksamhetsfunktion för avvikelshantering.

I inledningen av projektet var den kompetens som representanten för Försvarmaktens tillförde av högsta betydelse; då bakgrund till uppdraget beskrevs, då det upphandlade systemet demonstrerades samt vid identifieringen av intressenter. Men även domänkunskap som representanten för Försvarmakten tillförde var central för att kunna tolka utsagorna och översatt dessa till behov.

Arbetsgruppen fick en demonstration av det upphandlade systemet vilket kan ha påverkat arbetsgruppen arbete, främst vid konstruerandet av processmodellen över aktivitetsflödet. Demonstrationen ansågs dock vara nödvändig för att förstå respondenterna.

Framtagandet av behov kan ses som en iterativ process där flera personer bidrog till resultatet. Syftet med att involvera flera personer är att minimera risken för att utsagorna feltolkas. I detta steg bidrog Försvarmakten med kunskap och djupare insikter samt tolkningar ur ett Försvarmaktsperspektiv, vilket för utomstående kan upplevas obegripligt.

Respondenterna gav vitt skilda beskrivningar av dagens avvikelshantering. Detta skapade ett visst improvisationsmoment under intervjuerna då bakgrundsinformation om moment i avvikelshanteringsprocess saknades. Med en större insikt i varje enskild verksamhets avvikelshantering torde intervjuerna ha kunnat ge mer koncisa frågor vilket borde ha resulterat i en bättre bild av behoven. En brist under intervjuerna kan även ha utgjorts av otillräcklig kunskap om Försvarmaktens arbetsdokument. Under intervjuerna togs ibland dessa dokument upp utan att vidare frågor kring dokumenten kunde ställas.

5.4 Fallstudie: Kravhantering och MODAF

MODAF är det arkitekturramverk som Försvarsmakten beslutat att använda för att beskriva sin verksamhet. Det finns behov av att studera hur traditionella kravhanteringsmetoder kan nyttjas för att lägga grunden för de modeller som skall beskrivas. Syftet med den här fallstudien är att undersöka om det utifrån resultatet av en kravhanteringsprocess, i form av användningsfall och övergripande processbeskrivningar, går att skapa de initiala MODAF modellerna. Resultatet visar att det inte finns helt tydliga relationer mellan användningsfallen och MODAFs vyer. Det fanns i fallstudien en tendens att beskriva förmågor på alltför låg detaljeringsnivå. Men så väl användningsfall som processmodellen gav stöd för att skapa modeller enligt MODAF.

5.4.1 Inledning

För att underlätta hanteringen av komplexa system krävs någon form av arkitekturiell beskrivning (Maier & Rechtin, 2002). Detta gäller även för verksamheter och organisationer, och då så kallade Enterprise Architectures (Johanson & Ekstedt, 2007). Försvarsmakten anser sig ha behov av ett arkitekturramverk för att beskriva sin verksamhet och sina system. För att tillgodose dessa behov har Försvarsmakten beslutat att nyttja MODAF v1.2 som ramverk vid utveckling av arkitekturer⁴.

I denna fallstudie studeras hur resultatet av kravhantering kan nyttjas för att skapa beskrivningar enligt MODAF. Försvarsmakten har beslutat att nyttja MODAF för att beskriva sina system på samtliga nivåer. Det saknas studier avseende hur krav på förmågor skall specificeras som underlag inför MODAF-baserade modellutveckling. Syftet med denna fallstudie är att tillhandahålla Försvarsmakten kunskap gällande en ansats för att nyttja krav i form av användningsfall som underlag för MODAF-baserade modellutveckling. Fallstudien tar sin utgångspunkt i de användningsfall som skapats i fallstudien *Kvalitetsdriven kravhantering för utveckling av avvikelshantering*, vilken beskriver hur en behovsanalys genomförts avseende införandet av ett gemensamt avvikelshanteringssystem inom Försvarsmakten (Hallberg, Andersson, & Westerdahl, 2005). Resultat av studien utgörs av elva användningsfall och en processmodell. Användningsfallen beskriver användarnas interaktion med systemet (Kulak & Guiney, 2003). Processmodellen beskriver de roller och centrala aktiviteter som ingår i hanteringen av avvikelser. Användningsfallen och processmodellen användes som ingångsvärde för denna studie.

⁴ FM/CIO/0028/09 Underbilaga 1.4 09 100:58386

5.4.2 Metod

Fallstudien genomfördes i tre steg (1) skapa en ansats för att utifrån användningsfall och processbeskrivning ta fram MODAF modeller, (2) tillämpa ansatsen och dokumentera gjorda erfarenheter samt (3) sammanställa och presentera resultatet.

Ansats för att skapa MODAF modeller baserat på användningsfallen och processmodellen genomfördes i sex steg. Det första steget bestod i att noggrant granska och analysera användningsfallen, samt identifiera förmågor relaterade till dessa. Förmågorna dokumenterades i respektive användningsfallsbeskrivning. I ett andra steg modellerades förmågorna i de strategiska vyerna enligt MODAF. I det tredje steget utvecklade operativa vyer utifrån förmågorna samt med stöd av den övergripande processmodell som tagits fram. I det femte steget utvecklades systemvyer baserade på de operativa vyerna. I det sjätte steget, utvärderades modellerna av en MODAF expert som ej deltagit i det tidigare arbetet. Kommentarer från utvärderingen nyttjades för att förbättra MODAF modellerna. I fallstudien nyttjades verktyget MOOD⁵ för att skapa MODAF-modeller.

När MODAF används i utvecklingssyfte anses det lämpligt att använda först strategiska vyer (StV), därefter operationella vyer (OV) och slutligen systemvyer (SV).

Strategiska vyer beskriver förmågor och förhållandet mellan dessa. Operationella vyer beskriver den logiska strukturen av organisationen. Detta sker genom att beskriva logiska enheter, uppgifter och aktiviteter samt deras relationer. Systemvyerna är lösningsorienterade. I systemvyer presenteras krav på lösningar och existerande lösningar.

5.4.3 Resultat

Detta avsnitt presenterar hur användningsfallen och processmodellen nyttjades för att skapa MODAF beskrivningar, avseende tillämpning, utfall och erfarenheter. Sist i detta avsnitt presenteras den ordning i vilket MODAF vyerna skapades samt till vilka vyer som information hämtades från användningsfallen respektive processmodellen.

5.4.3.1 Modifiering av användningsfall

Tillämpning: Till var och ett av användningsfallen identifierade förmågor, istället för krav.

Utfall: Det identifierades 42 förmågor till användningsfallen.

⁵ <http://www.mood.co.uk/> (2009-11-23)

Erfarenheter: Redan tidigt i arbetet påpekades att det var för många ”förmågor” som identifierades i användningsfall och dessa hävdades vara på ”för låg nivå”. Orsaken ansågs ligga i att användningsfall syftar till att visa användares interaktion med system. Att beskriva förmågor utifrån användningsfall visade sig därmed inte vara helt oproblematiskt. Ofta uppstod diskussioner kring om det verkligen var en förmåga eller systemfunktion eller aktivitet som identifierats.

5.4.3.2 Strategiska vyer

Tillämpning: Det 42 förmågorna beskrevs i Capability Taxonomy (StV-2) samt grupperades med relaterade förmågor. Efterhand reducerades antalet förmågor och samtliga på nivå tre och nedåt (dvs mer detaljerade) togs bort.

Senare i processen skapades även en Operational Activity to Capability Mapping (StV-6).

Utfall: Utfallet utgörs av två vyer:

- Capability Taxonomy (StV-2): En huvudförmåga *Hantera avvikelser* med 9 stödjande förmågor, exempelvis *bereda avvikelseärende* och *bereda beslutsunderlag*.
- Operational Activity to Capability Mapping (StV-6): En matris med sex förmågor med ett till ett förhållande till sex aktiviteter. Exempelvis så mappades förmågan *Bereda beslutsunderlag* till aktiviteten *Handlägga* som bestod av sju aktiviteter till exempel *Identifiera åtgärd* och *Beskriva åtgärd*.

Erfarenheter: Det konstaterades vid modelleringen att de ur användningsfallen identifierade förmågorna, trots tidigare diskussioner, hade en alltför låg detaljeringsnivå. Därför reduceras antalet förmågor vid beskrivningen av de strategiska vyerna. I anslutning till detta uppstod en diskussion om vad som är en förmåga och vad som är en aktivitet i MODAF. En förmåga kan ses som en aktivitet eller en funktion av någon annan.

5.4.3.3 Operationella vyer

Tillämpning: Baserat på de modellerade förmågorna och användningsfallen utvecklades en Operational Node Relationship Description (OV-2) som visar logiska noder och kommunikationen mellan dessa.

Vidare skapades en Operational Activity Model vy (OV-5) med simbanor, även denna med stöd av processmodellen, samt en Operational Information Exchange Matrix (OV-3) i vilken OV-2's ”needlines” beskrivs.

Utfall: En Operational Node Relationship Description (OV-2) innefattande fem noder: (1) *Rapporteringsnod*, (2) *Fördelningsnod*, (3) *Handläggningsnod*, (4) *Beslutsnod* samt (5) *Åtgärdsnod*. Vidare skapades även Operational Activity Model (OV-5) innefattande sex aktiviteter på högsta nivån och fem simbanor.

Exempel på aktivitet är *Handlägga* och *Utvärdera* vilka finns i simbanan som representerar noden *Handläggningsnod*. Varje aktivitet beskrivs med en eller flera underliggande nivåer. En Operational Information Exchange Matrix (OV-3) skapades.

Erfarenheter: Det upplevdes som svårt att inte använda de roller som finns beskrivna i användningsfallen som noder. Ett försök gjordes med att skapa ”abstrakta” noder men det lyckades inte.

Processmodellen ansågs utgöra ett bra stöd för att beskriva aktiviteterna på högsta nivån i OV-5. För att få en överblick över hur aktiviteter relateras till de ovan skapade noderna kan dessa presenteras som ”simbanor”. Simbanor upplevdes som mycket användbara för att tydligt beskriva vilken nod som ansvarar för respektive aktivitet.

5.4.3.4 Systemvyer

Tillämpning: Två systemvyer konstruerades Function to Operational Activity/Service Function Traceability Matrix (SV-5) samt Functionality Description (SV-4).

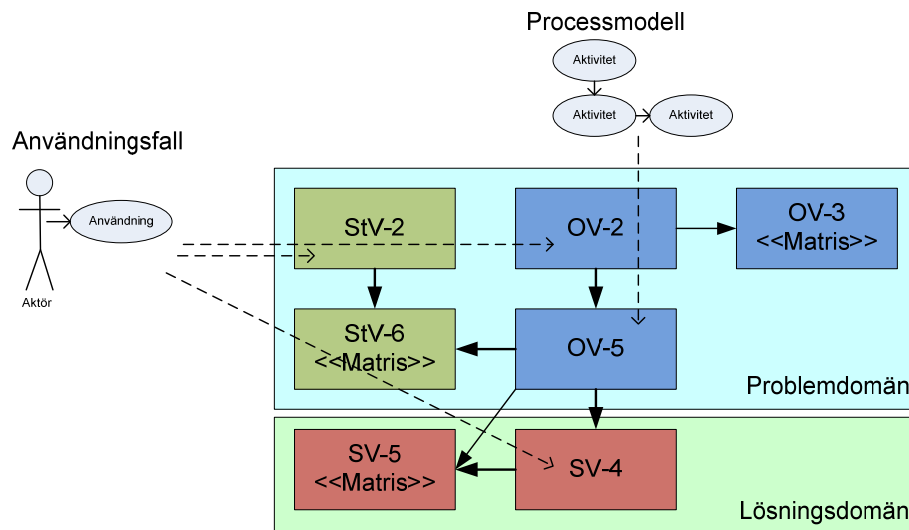
Utfall: Sex Functionality Description (SV-4) vyer skapades. Vidare skapades en Function to Operational Activity/Service Function Traceability Matrix (SV-5).

Erfarenhet: Vid konstruktionen av SV-4 kom användningsfallen till nytta. SV-4:an blev dock betydligt mer detaljerade än motsvarande användningsfall.

5.4.3.5 Modelleringsprocess

Användningsfall och processmodellen fungerade som stöd vid konstruktionen av flera vyer (Figur 5). Den modelleringsprocess som iterativt utvecklades under fallstudien innehåller följande steg.

1. StV-2 skapas baserat på förmågor identifierade i användningsfallen.
2. OV-2 skapas baserat på roller identifierade i användningsfallen.
3. OV3 skapades baserat på OV-2
4. OV-5 skapas baserat på identifierade aktiviteter beskrivna processmodellen och roller identifierade i användningsfallen. En detaljering av aktiviteterna genomfördes.
5. StV-6 skapades baserat på StV 2 och OV-5
6. SV-4 skapades baserat på användningsfallen, men detaljerades ytterligare.
7. SV-5 skapades baserat på de identifierade funktionerna i SV-4 samt aktiviteterna ifrån OV-5.



Figur 5. Illustration av modelleringen, samt i vilka vyer som information ur användningsfall och processmodellen nyttjas

5.4.4 Diskussion

Att utgå från användningsfall och processmodeller i MODAF modellering är möjligt. Det viktiga är att dock ha koll på de olika systemnivåer som finns. I denna studie ansågs övergripande användningsfall fungera bäst för att beskriva förmågor medan mer nära IT-stödet fungerade bäst för att beskriva funktioner i SV-4. Processmodellen kom att visa sig mer värdefull än vad som först troddes, då den fungerade som ett medium för att kommunicera kring vid skapandet av OV-5. En slutsats är dock att mer studier kring hur behovsanalyser och traditionell kravanalys kan stödja framtagande av MODAF modeller behövs. Vidare behövs studier avseende hur MODAF modeller kan nyttjas för att dokumentera resultatet av såväl behovs- som kravanalyser.

5.5 Fallstudie: Erfarenheter av MODAF

MODAF har valts av Försvarsmakten som arkitekturramverk. Det saknas dock vetenskapliga studier kring tillämpningen av MODAF. Vad är det i tillämpningen som är svårt, hur kan detta hanteras och vilka är dess styrkor? Denna fallstudie syftar till att skapa ett intervjuinstrument för att inhämta erfarenheter kring användningen av MODAF. I fallstudien tillämpas intervjuinstrumentet dels för att inhämta erfarenheter kring MODAF, men också för att validerar instrumentet.

Resultat visar att för det syfte MODAF är utvecklat fungerar det väl. Men det är inte någon allmängiltig lösning som fungerar för allt. Vidare är MODAF relativt komplext samtidigt som de verktyg som stödjer modelleringen är svåra att nyttja. Därmed kommer en omfattande utbildningsinsats att krävas för att lyckas införa MODAF i Försvarsmakten. Införandet kräver även att fördelarna med ett gemensamt ramverk tydligt presenteras. Intervjuinstrument fanns fungera väl, och kan nyttjas för betydligt mer omfattande studier.

5.5.1 Inledning

Försvarsmakten är en komplex och svåröverskådlig organisation. Detta medför att det är svårt att leda och genomföra utveckling av Försvarsmakten, då det inte är möjligt att fullt ut se effekterna av genomförda åtgärder. Det är inte heller möjligt att överskåda relationerna mellan IT-system, materiel, processer etc och hur dessa bidrar till Försvarsmaktens förmågor. Detta i sin tur gör det omöjligt att bedöma hur väl och kostnadseffektiva olika alternativa realiseringar är. Ett av de första ramverk för att skapa översikt av komplexa organisationer var Zachmans arkitekturramverk (Zachman, 1987). Med arkitekturramverk som baseras på metamodeller erhålls relationer mellan modeller och objekt. Det finns ett stort antal arkitekturramverk, exempelvis DoDAF, NAF och TOGAF (Schekkerman, 2003). Försvarsmakten har valt att basera sitt ramverk på MODAF.

MODAF är ett arkitekturramverk framtaget för brittiska MOD (Ministry of Defence) och används för att skapa verksamhetsövergripande arkitekturer. MODAF bidrar med en strukturerad beskrivning och en uppsättning enhetliga uttryck för beskrivning av komplexa arkitekturer. Objekten i modellerna och deras relationer beskrivs och regleras i MODAFs metamodell M5. Modeller skapas och presenteras i så kallade vyer. Vyerna representerar exempelvis system- eller verksamhetselement utifrån olika perspektiv. Vyerna i MODAF är grupperade i sju perspektiv:

- **All Views (AV)** Ger summarisk information rörande arkitekturmodellen och tillhandahåller information som gör det möjligt att leta i och indexera modellen.
- **Strategic Views (StV)** Dokumenterar den strategiska bilden rörande förmågor och hur dessa utvecklas över tiden. Detta understödjer förmågeplanering och styrning.
- **Operational Views (OV)** Dokumenterar operationella processer, relationer samt deras omgivning för att möjliggöra operationell analys och kravutveckling.
- **System Views (SV)** Dokumenterar systems (resursers) funktionalitet och interkonnektivitet för att stödja systemanalys och livscykelhantering.
- **Technical Views (TV)** Dokumenterar standarder och begränsningar som skall gälla för den övergripande arkitekturen.
- **Service Oriented Views (SOV)** Dokumenterar tjänsters funktionalitet, begränsningar och inbördes beroenden.

Acquisition Views (AcV) Dokumenterar program, beroenden, tidsscheman och programstatus för att tillhandahålla styrinformation samt programsynkronisering.

I fallstudie:

- utvecklas ett intervjuinstrument för att samla in erfarenheter av att tillämpa MODAF
- studeras vad som är MODAF styrkor respektive svagheter samt vad som är svårt och hur dessa svårigheter kan hanteras.

Fallstudien syftar till att tillhandahålla Försvarmakten kunskap avseende erfarenheter av att tillämpa MODAF samt ett intervjuinstrument för att kontinuerligt studera tillämpningarna för att baserat på detta kunna vidareutveckla exempelvis tillämpningsanvisningar.

Framtagandet av intervjuinstrumentet baserades på *Critical incident technique* (CIT). Att använda sig av CIT inspirerade intervjuer har tidigare visat sig fungera väl för liknande utvärderingar (Flanagan, 1954; Ölvingson, Hallberg, Timpka, & Greenes, 2002)

För att inhämta erfarenheter genomfördes intervjuer med fyra personer involverade i att tillämpa MODAF. Intervjuerna riktades mot att identifiera problem och svårigheter vid användning av MODAF, respektive framgångsfaktorer för nyttjande av ramverket.

5.5.2 Metod

Studien utfördes i tre steg: (1) framtagande av ett intervjuinstrument baserat på CIT, (2) genomförande av intervjuer, (3) analys av intervjumaterialet och (4) modifiering av intervjuinstrumentet.

Intervjuerna spelades in, parallellt med att intervjuanteckningar gjordes. Anteckningarna gjordes av en annan person än intervjuaren, detta för att få bästa möjliga fokus i och kvalitet på anteckningarna. Efter respektive genomförd intervju användes inspelningen för att komplettera anteckningarna. För bästa möjliga läsbarhet gjordes därmed inte en fullständig och detaljerad transkription av intervjun.

Analysen av intervjumaterialet genomfördes i form av en läsning av transkriberade intervjuer, markering av yttranden som bedömdes ha intresse och relevans för fallstudien.

Slutligen innebar genomförandet av fallstudien en utvärdering av intervjuinstrumentet. Utfallet av intervjuanalysen är naturligt beroende av kvaliteten på intervjuguiden.

5.5.3 Resultat

Resultatet presenteras i två huvuddelar: (1) Intervjuinstrumentet och (2) resultatet av analysen av intervjusvaren.

5.5.3.1 Intervjuinstrument

Intervjuinstrument syftar till att erhålla kunskap avseende erfarenheter relaterade till tillämpningen av MODAF. Detta görs bland annat för att kunna skapa tillämpningsanvisningar.

Intervjuinstrument består av följande frågeområden:

- Bakgrund
- MODAF erfarenhet
- Problem och svårigheter vid MODAF
- Framgångsfaktorer för MODAF användning vid systemutveckling

I frågeområdet *Bakgrund* ingår följande frågor:

- Vilken är din nuvarande tjänst?
- Vad är din huvudsakliga arbetsuppgift?
- Hur många år har du arbetat med systemutveckling?
- Vilka roller och vilket ansvar har du haft?
- Vilka verksamheter och projekt är du för tillfället engagerad i?
- Vilka systemutvecklingsmetodiker har du erfarenhet av?

- Vilka, om några, datorstöd har du erfarenhet av i samband med systemutveckling?

I frågeområdet *MODAF erfarenhet* ingår följande frågor:

- Hur länge har du jobbat med MODAF?
- Kan du kort beskriva hur MODAF nyttjats i projekt som du varit engagerad i?
- Följdes MODAF strikt eller gjordes avsteg?
- Varför har man valt att nyttja MODAF inom dessa projekt?
- Vilken upplever du är den största nyttan med att använda MODAF?
- Upplever du att det finns något som är dåligt med MODAF?
- Har du erfarenhet av att nyttja andra arkitekturramverk?

I frågeområdet *Problem och svårigheter vid tillämpning MODAF* ingår följande frågor:

- Kan du redogöra för de tre mest påtagliga problemen alternativt svårigheterna som ni haft relaterat till MODAF användning?
- Uppskattningsvis, hur vanligt förekommande upplever du att dessa problem och svårigheter är?
- Vilka konsekvenser medförde ovanstående problem/svårigheter för projektet och resultatet?
- Hur hanterades dessa problem?
- Ser du någon eller några åtgärder som skulle kunna förebygga eller lösa dessa problem?

I frågeområdet *Framgångsfaktor för tillämpning av MODAF* vid systemutveckling ingår följande frågor:

- Kan du redogöra för tre framgångsfaktorer för ett framgångsrikt nyttjade av MODAF?
- Hur kan man åstadkomma dessa framgångsfaktorer?
- Generellt, vilka är det tre mest kritiska momenten för att systemutvecklingsprojekt, i vilket MODAF nyttjas, skall nå framgång?

5.5.3.2 Resultatet av intervjuerna

5.5.3.2.1 Respondenternas bakgrund

Två kvinnor respektive två män intervjuades. Tre av respondenterna har militär bakgrund som utbildade officerare. Samtliga har en högre civil utbildning inom systemutvecklingsområdet.

Respondenterna ser sig själva ha en kompetensmässig bredd inom områden som verksamhets-, förmåge- och systemutveckling. De anser att begreppet förmågeutveckling handlar om att uttrycka behov oberoende av teknisk och organisatorisk realisering. Detta medan begreppet systemutveckling skapar associationer till tekniska realiseringar. Med ett svenskt perspektiv på systemutveckling används ofta begreppet holistisk systemsyn, vilket infattar betydligt mera än det rent tekniska.

Respondenterna har från två till 12 års erfarenhet av systemutveckling. De har också en bred erfarenhet av att använda olika typer av systemutvecklingsmetodiker och datorstöd för systemutveckling.

5.5.3.2.2 Erfarenheter av MODAF

Erfarenheterna av arbete med MODAF varierar mellan drygt ett upp till tre år. Inom respondentgruppen finns även erfarenhet av arbete med andra arkitekturramverk.

5.5.3.2.3 MODAF

En av respondenterna påpekade att drivkraft för att satsa på MODAF är utmaningen att identifiera, beskriva och analysera behov och till behoven relaterade krav. Det påpekas exempelvis av en respondent att det existerar och produceras betydande volymer av kravdokument, men att kvaliteten inte därmed behöver vara hög. Krav är i allmänhet svåra att verifiera att de är uppfyllda. De är gärna formulerade på en detaljerad teknisk nivå, som kan vara svår att koppla till en verksamhets mera övergripande krav.

MODAF är baserat på det amerikanska ramverket DODAF, men DODAF är mera orienterat mot tekniska system, ledningssystem. MODAF har däremot ett fokus på resurser och lösningar, vilket speglas i den bredare uppsättning vyer som MODAF omfattar. Initialt hanterade DODAF inte humanorienterade frågor.

MODAF:s tjänstevy är enligt flera av respondenterna ett huvudsakligen svenskt bidrag till MODAF. Detta poängteras av några respondenter som en tydlig styrka med detta arkitekturramverk, genom att det möjliggör en tjänstebaserad utveckling, och att denna kan samordnas med behov och krav relaterade till övriga vyer. En av respondenterna reflekterar dock kring om tjänstevyns bidrag försvarar arbete och kostnader lagt på detta. Enligt denna respondent måste en separat begreppsapparat byggas upp för tjänstevyn, samtidigt som mera funktionella begrepp också behövs inom en funktionell arkitektur.

Respondenterna påpekar - med variationer i hur budskapet formuleras – den roll MODAF kan spela för att:

- Beskriva verksamheter
- Definiera vad man vill beskriva och till vilken nivå

- Strukturera information och därigenom verksamheter, förmågor och system
- Säkra interoperabilitet mellan verksamheter, förmågor och system
- Dokumentera för att säkra spårbarhet i verksamhets-, förmåge- och systemutveckling
- Minimera översättningsproblem mellan verksamheter, förmågor och system.

Flera respondenter lyfter fram förmågeperspektivet som en särskild styrka och rent av unik egenskap hos MODAF. Ramverket täcker allt i samma ramverk, som en respondent formulerade sig. MODAF inbjuder till att analysera i första hand förmågor som sedan kan brytas ner till verksamhetsaktiviteter som sedan är spårbara till systemkrav.

Försvarsmakten strävar efter att åstadkomma modellbaserad förmågeutveckling. I detta ligger, enligt en av respondenterna, att definiera ett förmågegap som bedöms kunna lösas av exempelvis ett nytt eller modifierat förbandssystem alternativt materielsystem. Systemlivscykeln för sådana system är ett antal år, från idé till operativ användning och slutligen skrotning av systemet. Detta accentuerar vikten av övergripande och strukturerad styrning, vilket tillämpning av MODAF skulle kunna bidra till.

5.5.3.2.4 MODAFs brister

MODAF har även brister och svagheter. Tidshantering i MODAF är endast övergripande i termer av faser och epoker. Ekonomi och affärsmässighet hanteras inte alls i MODAF, samtidigt som detta är något som måste beaktas för att veta om system och förändringar är ekonomiskt realiserbara. Det finns en risk att utveckla med inriktning på perfekta system, utan att tvingas göra ekonomiskt baserade prioriteringar.

Att MODAF inte är egenutvecklat inom Försvarsmakten kan ge en förändringsproblematik; ”not invented here”-relaterade problem kan existera. Huruvida detta är en svaghet med MODAF eller dess tillämpning kan dock diskuteras. MODAF:s komplexitet indikerar därmed ett tydligt behov av att utbilda för att underlätta införande av MODAF inom Försvarsmakten.

En komplexitet tillkommer också från respektive domän där MODAF används, genom att en uppsättning begrepp och koncept måste definieras. Det är svårt att göra detta på ett enkelt och stringent sätt.

5.5.3.3 Problem och svårigheter

I intervjuerna beskrivs olika problem och svårigheter med MODAF samt konsekvenser dessa medför. Hur dessa kan hanteras eller förebyggas av preventiva åtgärder berördes också.

Komplexiteten hos MODAF återkommer i intervjuerna som en utmaning. Det manifesterar sig på olika sätt, exempelvis i metamodellens komplexitet, som ställer höga krav på modellerare.

Val av abstraktionsnivå är ett relaterat problem. För olika aktörer kan vad som utgör en konkret lösning respektive övergripande aktivitet skilja sig markant. Vid MODAF-modellerande utgör detta tydliga problem då dessa nivåer måste hanteras och relateras. Respondenterna poängterar vikten av att veta vilken frågeställning som skall hanteras och hur lösningen avses kunna uppnås. Likaså gäller det att veta vad som räcker som lösning; det är viktigt att avgränsa hur stor del av situationen som skall modelleras.

Verktyg som används för att dokumentera modeller är svår använda, i och med att de upplevs som komplicerade. De kommer därmed att kräva relativt omfattande utbildningar. Existerande verktyg implementerar MODAF på olika sätt. Det ställer betydande krav på dem som skall nyttja MODAF, speciellt avseende MODAF:s metamodell. Respondenterna menar att det vore önskvärt att verktygen hade inbyggt betydligt mera kunskap om metamodellen, för att underlätta för MODAF-användare att hantera objekt och kopplingar mellan dessa. Det är inte rättfram och helt strikt följa MODAF och att förstå var relationer finns. Relationer är ofta inte enkla ett-till-ett relationer, de kan mycket väl gå via ett antal andra objekt.

Bristande förändringsbenägenhet kan medföra svårigheter att införa MODAF. Vidare uppstår problem när MODAF möter verkligheten. Det vill säga att det som i teorin verkar enkelt visar sig relativt ofta vara betydligt mer komplicerat att åstadkomma i praktiken. Såväl för enkla som för komplicerade modeller innebär detta problem. Att göra precis lagom omfattande modeller är en svår avvägning.

Väl utformade utbildningsinsatser krävs för att lyckas med införande av MODAF. Detta gäller såväl utbildning av de som modellerar, som av kund-representanter inom modellerade verksamheter. Sådan utbildning kräver tid, inlednings även för att helt enkelt bekanta sig med varandra. Vidare indikerar detta vikten av att ha tillgång på bra modellerare.

5.5.3.4 Framgångsfaktorer

MODAF uppmuntrar till ett strukturerat arbetssätt och strukturerat informationshantering samt återanvändning av information. Detta kräver bra beslutsunderlag för att kunna identifiera de frågeställningar och problem som skall hanteras med MODAF modellering. Vidare kräver det bra dokumentation, vilket även säkrar spårbarhet. Likaså underlättar detta att visa för kunden hur man har tolkat verksamheten och därmed möjliggöra justeringar.

För att lyckas med att införa MODAF krävs en stark förändringsledning inom Försvarmakten. Vidare krävs att en acceptans för att MODAF som beskrivningsramverk skapas. Respondenterna påtalar vikten av en tydlig

problemdefinition, förmåga att fatta beslut och väl utformad och inriktad utbildning för att nå framgång vid MODAF-införandet.

5.5.4 Diskussion

Respondenterna beskrev ett antal olika brister med MODAF. De stora utmaningarna såg de ligga i att lyckas förankra dess användning inom Försvarsmakten. De beskrev styrkor hos MODAF som exempelvis en ansats för att utgå från de förmågor som behövs och inte tekniska realiseringar, samt att en styrka med MODAF ligger i att kunna beskriva olika fenomen i samma modell. Det påpekas samtidigt att MODAF inte är någon form av ”mirakelmedicin” för utveckling. Att lyckas med MODAF-baserad utveckling, ställer samma krav som i all annan utveckling; tydliga systemgränser, vad som skall beskrivas och vad som inte skall beskrivas. Vidare är det viktigt att det underlag som ligger till grund för modellerna är korrekt. MODAF innefattar ingen självriktande funktion. Slutligen måste formerna för att bedriva modelleringsarbetet utformas, exempelvis hur domänexperter bör nyttjas i framstagande av modeller.

Intervjuguiden fungerade väl, även om den inte följdes i alla detaljer. Den utgör ett välfungerade underlag för intervjuerna och säkrade att de viktiga frågeställningarna togs upp.

5.6 Fallstudie: Erfarenheter av modellbaserad utveckling

Modellbaserad utveckling är ett område på frammarsch, forskningsmässigt och tillämpningsmässigt. Trots detta saknas studier som behandlar praktiska erfarenheter av modellbaserad utveckling och införande av modellbaserade system och arbetssätt. I denna studie fokuseras erfarenheter av modellbaserad utveckling i form av fördelar, nackdelar och utmaningar med utgångspunkt från Försvarsmaktens Program PRIO. PRIO har arbetat modellbaserat i konfigureringen av affärssystemet SAP för successivt införande i Försvarsmaktens verksamhet. Det pågår ett arbete inom Försvarsmakten med att införa en ansats för modellbaserad utveckling. Syfte med denna fallstudie är att tillhandahålla Försvarsmakten praktiskt erfaren kunskap avseende modellbaserad utvecklingen. Syftet med fallstudien är således att generera kunskap avseende modellbaserad utveckling, med speciellt fokus på Försvarsmakten.

Studien visar att PRIO upplevt många av de utmaningar och problem som generellt förknippas med modellbaserad utveckling och att projektet dessutom illustrerar inneboende brister i många modellbaserade ansatser. De viktigaste utmaningarna är vikten av förändringsledning, helhetsperspektiv samt tid och resurser. När det gäller brist avseende kompetens kan detta hänföras till att modellbaserade ansatser tenderar att i otillräcklig omfattning uppmärksamma behovet av användarmedverkan och datainsamling till grund för konstruktion av modellerna.

Trots detta är det angreppssätt som PRIO valt, att börja i begränsad, experimentell skala med den modellbaserade ansatsen (ett förband) och sedan successivt förflytta det modellbaserade systemet ut i hela verksamheten (alla förband), riktig. För fortsatt arbete, i PRIO och i alla Försvarsmakten projekt som tillämpar modellbaserad utveckling i någon form, föreslås kontinuerlig utvärdering och ett bättre och mer strukturerat tillvaratagande av erfarenheter, inte minst mellan Försvarsmakten projekten. En erfarenhetsdatabas föreslås som stöd för dokumentation av sådana erfarenheter.

5.6.1 Inledning

Det finns ett stort intresse för modellbaserad utveckling inom systemutveckling (ex. Siegel, 2005, Molina et al 2008, Sousa et al 2008, Hallberg et al 2009). I litteraturen beskrivs i relativt hög utsträckning fördelar med och förutsättningar för modellbaserad utveckling. De fördelar som nämns innefattar bland annat att de initiala modellerna tas fram utan hänsyn till vilka underliggande teknologier som skall användas (Selic, 2003), vilket ökar möjligheten för dem som inte är

tekniska experter att aktivt delta i utvecklingen (Balasubramanian et al, 2006) samt att modellbaserade angreppssätt stödjer inkrementell iterativ utveckling som är nödvändig vid utveckling av komplexa system (Selic, 2003). Det har vidare hävdats att det vid modellbaserad utveckling av verksamheter är möjligt att separera verksamhet från informationssystem, samtidigt som kopplingarna mellan dessa erhålls (Lee, 2005; Kulkarni & Reddy, 2008) och att modellbaserad utveckling möjliggör verksamhetsdriven utveckling istället för mer traditionellt teknikdrivna ansatser (Lee, 2005).

När det gäller förutsättningar pekar studier på vikten av att ta hänsyn till de tekniska, organisatoriska och sociala förutsättningar som råder (Selic, 2003) och på behov av verktyg för datalagring, för modellbeskrivning och validering av modeller (Romero et al 2007). Det talas även om vikten av att avsätta ordenligt med resurser för att i det initiala utvecklingsarbetet välja rätt arkitektur och utvecklingsstrategi samt för att införa dessa i metamodeler (Kulkarni & Reddy, 2008). Andra förutsättningar som relaterar till modellerna hävdas vara att dessa är: (1) abstrakta, (2) begripliga, (3) korrekta, (4) förutsägbara och (5) kostnadseffektiva, för att vara användbara och effektiva Selic (2003).

Däremot finns få vetenskapliga studier avseende praktiska erfarenheter av att tillämpa modellbaserade utvecklingsansatser. Den forskning som finns diskuterar just förutsättningar och fördelar. Vidare har den oftast ett tekniskt perspektiv där t ex olika arkitekturramverk jämförs eller modellkvalitet diskuteras. Studier med ett praktiker, ”how to do it” eller organisationsperspektiv saknas. Detta trots att tillämpningen av den modellbaserade utvecklingsprocessen i sig är central för att uppnå ett ”lyckat” resultat eller användbart system. Ofta förutsätts exempelvis implicit att den kunskap och information som skall ligga till grund för modellerna finns tillgänglig och är oproblematiske. Hur sådan kunskap skall inhämtas och inarbetas i modellerna behandlas inte. Kopplat till detta är användarmedverkan i modellbaserad utveckling är ett outforskat område.

Bristen på mer erfarenhets inriktade, ”lessons learned” orienterade studier gäller både utvecklingsprocessen för och införandet av modellbaserade system, samt både generellt och i militära kontexter.

Syftet med fallstudien *Erfarenheter av modellbaserad utveckling* är därmed att studera ett modellbaserat systemutvecklingsprojekt ur ett praktiker perspektiv och därur identifiera erfarenheter i form av fördelar, nackdelar, utmaningar, problem och framgångsfaktorer. Målet är ”lessons learned” för modellbaserad utveckling och införande av modellbaserade system, generellt och i kontexten Försvarsmakten.

Studien har genomförts som en fallstudie i vilket Försvarsmaktens Program PRIO har fått tjäna som ett exempel på modellbaserad utveckling. Program PRIO syftar till att utveckla och införa en integrerad affärslösning inom Försvarsmaktens områden; ekonomi, personal, logistik, ledning och styrning. All

denna verksamhet skall stödjas av ett gemensamt system och skall ersätta ungefär 70 av Försvarmaktens nuvarande verksamhetsledningssystem. Förändringen från många små system med egen information till ett gemensamt system där informationen delas anses som en nödvändig förändring för att effektivisera verksamheten. Den modellbaserade tillämpningen i PRIO relaterar till konfigureringen av valt affärssystem (SAP) där modellering med hjälp av MODAF genomförts av de av Försvarmaktens verksamhetsprocesser som systemet skall stödja.

5.6.2 Metod

I fallstudien har Program PRIO studerats ur ett praktiker perspektiv genom att genomföra intervjuer med respondenter kopplade till programmet. Genomförandet av fallstudien har skett i 4 steg; (1) skapa intervjuinstrument, (2) välja respondenter, (3) genomföra intervjuer och (4) analysera intervjuer.

Intervjuinstrumentet var semi-strukturerat och bestod av 22 frågor för en riktad öppen intervju. Frågorna var organiserade under två huvudsakliga teman; PRIO och SAP system samt Modellbaserad utveckling med specifika frågor rörande erfarenheter, problem, framgångsfaktorer et c kopplade till respektive tema. Vidare innehöll intervjuguiden frågor om respondenternas yrkes- och erfarenhetsmässiga bakgrund av relevans för modellbaserad utveckling.

För intervjuerna valdes sju respondenter med olika befattningar i eller nära kopplat till Program PRIO. Respondenterna valdes för att representera olika perspektiv och erfarenheter i program och Försvarmakten verksamhet inom såsom "beställare", "leverantörer" och "mottagare" av systemet. Respondenterna har därmed innefattat t ex verksamhetschefer, projektledare, projektkoordinatorer och konsulter/utvecklare.

Totalt sex intervjuer genomfördes. Varje intervju tog 1-1,5 timme i anspråk och genomfördes på respondenternas respektive arbetsplatser. Samtliga intervjuer utom en spelades in (en fick inte spelas in av säkerhetsskäl kopplade till den lokal i vilken intervjun genomfördes). Minnesanteckningar togs vid samtliga intervjuer. Av de inspelade intervjuerna gjordes senare fullständiga transkriptioner.

En kvalitativ analys har utförts på det transkriberade materialet, samt på anteckningar från den intervju där det rådde inspelningsförbud. Den kvalitativa analysen skedde i två steg. I första steget läste en av forskarna materialet. Utsagorna från respondenterna sorterades i olika kategorier. Dessa kategorier samlades sedan under större teman och beskrevs i löpande text. Texten, samt transkriberingsmaterialet och anteckningar lästes sedan av två ytterligare forskare som modifierade texten och de övergripande temana.

5.6.3 Resultat

Analysen av intervjuerna resulterade i 5 övergripande teman; modellbaserad utveckling, kompetens, förändringsledning, genomförande och återkoppling. Resultatet redovisar fördelar och nackdelar inom varje tema samt de utmaningar som respondenterna påtalat.

5.6.3.1 Modellbaserad utveckling

Rationalisering, kontroll och identifiering av dubbla aktiviteter uttrycks som fördelar med modellbaserat arbetssätt. Respondenterna hävdade att genom att skapa processmodeller av relevanta delar av Försvarmakten erhålls en överblick över alla delsystem och var av aktiviteter som genomförs ”dubbelt” kan identifieras. Elimineringen av dubbelarbete ger rationaliseringseffekter samt ger möjlighet att bringa ordning och reda i organisationen.

Respondenterna menar att modellerna ger en möjlighet att betrakta Försvarmaktens organisation på ett nytt sätt. Utifrån modellerna kan en medvetenhet skapas om organisationens funktioner och hur de är kopplade till varandra. De kan betraktas från olika vyer och med olika lager av djuphet. Respondenterna ansåg detta utgöra en värdefull pedagogisk vinst bland annat för att visa hur förändringar i en del av systemet kan ha oönskade effekter på samma system sett från en annan vy eller för att vid införande av ett nytt system, en ny produkt eller ny funktionalitet beskriva såväl tekniska som verksamhetsmässiga effekter av förändringen.

Flera respondenter ansåg det vara en utmaning att skapa förståelse i organisationen för det hjälpmedel ett modellbaserat arbetssätt utgör och att sedan lyckas etablera det som ett arbetssätt.

När det gäller praktiska erfarenheter ses kravspecifikation för systemet som en nackdel av de respondenter som kan sägas utgöra leverantörer. De påtalar att kraven på systemet vid upphandling och implementering redan var satta och statiska; det vill säga de var ej möjliga att anpassa till förändringar eller rådande realitet i form av ekonomiska eller tekniska begränsningar.

De modeller som utvecklades av Försvarmaktens processer inför upphandlingen av affärssystemet ses av leverantörerna däremot som en bra förberedelse från Försvarmaktens sida. Dock ansågs modellerna något för detaljerade. Mottagarna påpekar i sin tur att vissa saker som relaterar till modellerna i sig, såsom informationsklassning har fungerat väldigt bra.

5.6.3.2 Kompetens

Att modellera Försvarmaktens processer till att överföra dem till ett affärssystem är på många sätt en utmaning. Respondenterna påtalade behovet av att arbetet förutsätter en mycket gedigen kunskap om verksamheten. Den mängden kunskap ansågs inte kunna besittas av en och samma person utan flera personer,

med omfattande erfarenhet, måste samverka. Respondenterna hävande att kärngruppen i programmet som deltar i ”workshops” med leverantörer har inte ensamma den kompetensen. De måste tilldelas den från linjeorganisationen. Samtliga respondenter påtalar att detta inte alltid skett och det är en nackdel.

Konsekvenser av detta är att misstag har gjorts och som behövt rättas till i efterhand med sena förändringar som följd. Förändringar påtalas av både leverantörer och verksamma i programmet som olyckliga. När verksamheten prövar systemet och identifierar misstagen ses de som fel hos systemet och inte som fel i modellen som systemet bygger på. Respondenterna menar att det är svårt för användare som inte är vana vid ett modellbaserat arbetssätt att koppla de upplevda felen till modellen, och inte till systemet.

En möjlighet som påtalades är att lägga in den granskning som sker av det som beslutats under ”workshops” tidigare för att på så sätt fånga upp misstag i ett tidigare skede än under det slutliga systemtestet i verksamheten.

5.6.3.3 Förändringsledning

Flera respondenter påtalade problemet med att det under arbetet varit oklart om vilket mandat som finns, dels för faktiskt genomförande av projektet, dels för den förändringsledning som krävs för att projektet skall lyckas. För förändringsledning känner både leverantörer och verksamma i programmet att tydligt uppdrag saknas.

Respondenterna menade att när en bärkraftig förändringsledning saknas innebär detta en svår utmaning och påverkar organisationen som helhet, dels genom att organisationen inte är säker på att förändringarna över huvudtaget skall genomföras, dels för att den inte vet vad förändringarna kommer att innebära. En konsekvens av att förändringsledningen inte genomförs på en övergripande organisationsnivå är att utvecklingsprojektet tvingas ta sig an denna uppgift ad-hoc mässigt då det krävs. Förändringsledning har inte setts som en primär uppgift i arbetet inom PRIO, samt att kompetens och resurser saknas i för att utföra ett genomtänkt förändringsarbete.

En strategi för förändringsarbetet från utvecklingsprojektets sida där linjerepresentanterna från verksamheten utgör en kompetensresurs till programmet ansågs som önskvärd. Under processens gång tar linjerepresentanterna med sig information till verksamheten och när införandet av systemet är slutfört tar de hem systemet och processmodeller för vidare förvaltning. Den strategin har dock inte ansetts fungera fullt ut eftersom de linjerepresentanter programmet har tillgång till inte representerar de ”rätta” användarna och inte har tillräcklig kunskap om ett modellbaserat arbetssätt. Programmet upplever att de varit svårt att få förståelse från linjen för programmets aktiviteter. För att åtgärda detta har processform för linjerepresentanterna etablerats.

En viktig del av förändringsledning är utbildning. Utbildningen av personal relaterat till programmet har i det initiala skedet varit obefintlig enligt flera av respondenterna.

5.6.3.4 Genomförande

Flera respondenter har påtalat osäkerhet i olika former vad det gäller mandat för genomförande av programmet. Undantaget är de respondenter som representerar leverantörerna. De uttrycker snarare sig som säkra på genomförande, då de som en fördel upplever att Försvarmakten har en rak beslutsprocess och att fattade beslut ligger. De faktorer som bidrar till osäkerhet runt genomförandet på olika sätt är istället tid, kompetensresurs, särskiljning av programmet från övrig verksamhet och kultur/vana.

Respondenterna uttrycker på flera olika sätt att det är en utmaning för de verksamma i programmet att arbetet har tagit tid. Dels är det svårt att hålla målbilden fokuserad och inte låta den förändras varefter nya erfarenheter erhålls. Dels påverkas den vanliga organisationen av programmets utsträckning i tid såtillvida att den får en ambivalent inställning till dess faktiska genomförande. En del respondenter menar att detta fått till konsekvens att verksamheten inte fullt ut prioriterat programmets genomförande så högt som är kutym i Försvarmakten för genomgripande projekt.

En annan faktor som ansågs påverka är att i Försvarmakten finns en kultur att bestämma förändringar på dokument nivå, vilka sedan inte alltid genomförs i organisationen så att faktisk organisation och gällande dokument inte är överrensstämmande. I en modellbaserad utveckling av organisation krävs, tvärt emot rådande kulturell förståelse för vikten av att modellerna stämmer med verkligheten.

Särskiljningen av programmet från övrig verksamhet är en tredje faktor som påtalas av respondenterna. Särskiljningen innebär programmet hanterar kontakterna med leverantören av systemet SAP, gör att programmet behöver en nära relation kontakt med organisationen som kompetensresurs för att lyckas. Denna kontakt har saknats och detta ökar osäkerheten på mandatet för genomförande. Ett konkret exempel på bristande kontakt är de som arbetat med modellerna och den verksamhet de modellerat, inte fysiskt befunnit sig på samma plats. Kopplingen mellan beställare, utförare och mottagare har saknats. Införande 1 är dock ett positivt, tydligt steg i genomförandefasen och har medfört att känslan av osäkerhet kring genomförandet har minskat.

5.6.3.5 Återkoppling/utvärdering

Systematisk utvärdering av och ansvar för projektets delar är ytterligare ett område som är vagt för flera av de verksamma i programmet. I första hand upplever de att någon borde ta mandatet för att utvärdera deras arbete. I andra hand kan de tänka sig att utvärderandet av projektet skall göras av dem själva. I

det senare fallet behövs ramar för projektet att utvärdera sig mot. Den systemförändring och möjlighet till systemstöd som programmets arbete utgörs som en bra utveckling och en fördel. Däremot är den tekniska lösningen med SAP och det sätt förändringen utförs på en utmaning och ett arbete som bör utvärderas och ges återkoppling på. En respondent påpekar explicit att ”lessons learned” inte hunnits med då man är så mycket uppe i processen men att den biten bör inarbetas i programmets kommande faser.

5.6.4 Diskussion

Modellbaserad utveckling är ett relativt nytt och komplext område och tillämpning av modellbaserade ansatser genomförs därför lämpligen stegvis och experimentellt där erfarenheter samlas i en erfarenhetsbas innan ansatsen tillämpas i stor skala för en hel organisation (Harpst, 2009). Program PRIO syftar visserligen ultimatum till hela Försvarmakten men systemet levereras iterativt i olika versioner och i första steget har en begränsning valts att görs till ett förband. Detta verkar vara ett helt adekvat angreppssätt enligt litteraturen kring modellbaserad utveckling.

Respondenterna ger en i vissa delar likartad erfarenhetsbild men skiljer sig i andra delar. Fördelar med modellbaserad utveckling uttalas tydligare de av verksamma och drivande aktörerna i programmet, medan leverantörerna av affärssystemet är mer tveksamma och påtalar flera konkreta problem som uppkommit under processen: t ex alltför detaljerade modeller och allt för låsta krav. Att förändringsledning saknats är dock både beställare, leverantör och mottagare överrens om. Det är värt att notera att respondenterna, trots att de alla deltagit i programmet och i praktiken arbetat modellbaserat, i de allra flesta fall är osäkra på vad modellbaserad utveckling är.

5.6.4.1 Erfarenheter i ett forskningsperspektiv

De erfarenheter som respondenterna ger uttryck för speglar i stort den forskning som finns inom området modellbaserad utveckling. Trots att man, framför allt ifrån beställarsidan, talar mycket om fördelarna med angreppssättet och även nämner en del saker som fungerat bra är det tydligt att PRIO har upplevt många av de problem och utmaningar som kan sammankopplas med ansatsen.

Erfarenheter innefattar framför allt vikten av förändringsledning, och helhetsperspektiv, utmaningar i den modellbaserade utvecklingsprocessen i sig, kopplat både till modellerna i sig och till den datainsamlingsprocess som ligger till grund för dem, samt ”klassiska” systemutvecklingsproblem i form av brist på tid och resurser.

Bristen på förändringsledning är den stora erfarenheten ur programmets arbete. Modell- och arkitekturbaserade ansatser såsom t e x Enterprise Architecture (EA) betonar att detta är den absolut viktigaste framgångsfaktorn för ett modellbaserat

arbete och vid införande av nya angreppssätt och system (Hart, 2003). Om personal inte informeras om och motiveras till att ta till sig förändringar minskar deras acceptans för ovanstående, vilket självklart försvårare införandeprocessen. Att ha mandat för att genomföra förändringarna och sälja in dem i verksamheten är fundamentalt (Hart, 2003) I programmet insågs initialt inte vikten av förändringsledning och det har påverkat införandeprocessen negativt. Viktigast av allt, mandatet för att förändra har saknats. Att programmet fortsättningsvis får ett sådant mandat är absolut fundamentalt för ett fortsatt ”lyckat” införande med system som används och arbetssätt som blir accepterade.

Helhetsperspektiv är en annan fundamental aspekt i modellbaserade utvecklingsansatser. Modellerna skall relatera till en organisations övergripande tydliga och definierade mål och enstaka projekt skall arbeta mot, spegla och omsätta målen i konkret verksamhet. Kopplingen mellan övergripande strategier och individuella projekt bör vara tydlig, likaså kopplingen och samarbetet mellan olika projekt och mellan de aktörer (t ex beställare, leverantör, mottagare) som berörs av de modellbaserade aktiviteterna och systemen. Detta har inte varit fallet i PRIO som till viss del drivits isolerat och upplevt alltför lite stöd från linje-verksamheten. Även om resultatet av PRIO kommer att påverka hela Försvarsmakten behövs fortsättningsvis en diskussion om och strategier för PRIOs koppling till och kontinuerliga integrering med den skarpa verksamhet programmet skall stödja och mellan de olika typer av aktörer och projekt som ingår där.

Modellerna i utvecklats PRIO upplevdes av leverantörerna som alltför detaljerade. Att fokusera ”rätt” modeller som speglar projektets syfte och dessutom hålla dem på ”rätt” detaljnivå är något som flertalet studier pekar ut som svårt (t ex Selic, 2003).

Vad forskningen och de modellbaserade tillämpningarna i sig dock tenderar att negligera är processen för insamling av data och information som ligger till grund för modellerna – processen för kunskaphämtning. Denna process diskuteras ofta överhuvudtaget inte utan kunskaper förväntas på ett oproblematiserat sätt ”bara finnas där”. Respondenternas uttalande kring detta illustrerar tydligt att så uppenbarligen inte är fallet. Tvärtom har påtalats exempel på att det ibland samlats in ”fel” information och därmed byggts in fel i modellerna och vidare in det konfigurerade systemet. Att modellerna är korrekta är fundamentalt i modellbaserad utveckling eftersom de påverkar alla efterföljande aktiviteter och byggs in i tekniska lösningar. I fallet PRIO handlade den felaktiga informationen mycket om bristen på tillgång till ”rätt” kompetenser. De personer som var representerade i modelleringsgrupperna hade ofta otillräcklig kunskap om de processer som skulle modelleras. Detta relaterar till traditionell användarmedverkan i systemutveckling och problemet att få ”rätt” användare representerade. De som här domänkunskapen eller utgör systemets slutanvändare är sällan de som har mest tid att delta i utvecklingsarbetet.

Brist på tid och resurser är ofta ett problem i många typer av utvecklings- och förändringsarbete. Forskningen har konstaterat att modellbaserade ansatser är tids- och resurskrävande (Kulkarni och Reddy, 2008). Genomförandet av PRIO utgör inget undantag.

5.6.4.2 "The way ahead"

Flera respondenter nämnde behovet av utvärdering och återkoppling som fundamentalt för fortsatt PRIO arbete. Behov av utvärdering av modellbaserad utveckling har även identifierats i andra Försvarsmakten projekt (Pilemalm, Kylesten och Allberg, 2009). Studierna av dessa projekt (FMLS TS, NBG 11 och MARKUS) visar att de Försvarsmakts projekt som idag tillämpar ngn form av modellbaserad utveckling arbetar isolerat och inte lär sig av varandras erfarenheter. Ökat samarbete mellan projekten är inte nödvändigt bara för helhetsperspektivet i sig utan skulle kunna generera just ett värdefullt erfarenhets- och "lessons learned" utbyte. Ett datorstöd i form av exempelvis en erfarenhetsdatabas skulle kunna underlätta strukturerad dokumentation av och därmed öka nyttan av ett sådant utbyte.

Trots att Program PRIO upplevt flera av de utmaningar och problem man ofta sammankopplar med modellbaserad utveckling har man levererat en första version av det konfigurerade affärssystemet, har påbörjat version två och ser med tillförsikt på framtiden. Man har tagit till sig vissa av erfarenheterna och jobbar idag mer pro-aktivt med t ex förändringsledning. Exempelvis har man börjat ta fram information och utbildningspaket runt PRIO, modellbaserad utveckling och vad det nya systemet kommer att innebära för verksamheten i form av förändrade arbetssätt. Att fortsätta på den vägen och ackumulativt ta till sig erfarenheter – egenupplevda och de en kontinuerlig utvärdering pekar på – i PRIO, i forskning, generellt och i militärkontext, framför allt i andra av Försvarsmaktens modellbaserade projekt - är ett bra sätt att successivt bygga in den modellbaserade ansatsen med tillhörande system i Försvarsmaktens skarpa verksamhet. Att ta tillvara på de "lessons learned" som beskrivs i denna studie är ett steg på den vägen.

5.7 Fallstudie: Kvalitets- och beslutsgrindar

I utvecklingen av nya produkter är det nödvändigt att en viss kvalitet upprätthålls. För att fånga upp projekt som inte håller tillräckligt god kvalitet kan kvalitets- eller beslutsgrindar användas. I ett projekt får ett delresultat passera kvalitetsgrinden och gå vidare till nästa fas endast om det uppfyller ett antal fördefinierade kriterier. Syftet med denna fallstudie är att beskriva hur kvalitets- och beslutsgrindar utformas och används. Studien är baserad på en litteraturstudie.

Utformningen av kvalitetsgrindar ser på en övergripande nivå relativt lika ut mellan olika industrier och företag. På en mer detaljerad nivå kan det skilja sig mycket eftersom grindarna är anpassade efter individuella organisations- eller projektmål. Många positiva erfarenheter från användare av grindssystem har dokumenterats men litteraturstudien visar på att det finns flera fallgropar. Ett av de största problemen är grindar som saknar ”tänder”, vilka kan släppa igenom projekt som inte håller tillräckligt hög kvalitet. Ett annat problem är att avgöra på vilken nivå kriterierna skall ligga och hur de skall utformas. Det finns idag inga entydiga svar på hur dessa svårigheter skall hanteras.

5.7.1 Inledning

Inom dagens produktutveckling är konceptet grindar väl utbrett och en mängd varianter av dess tillämpning finns. Grindarna kan ha olika benämningar, t.ex. kvalitetsgrindar eller beslutsgrindar⁶, men det övergripande syftet är i stort sett detsamma. Grindar fungerar som milstolpar för beslutsfattande under utvecklingsprocessen i ett projekt och vid varje grind utvärderas arbetet och ett beslut fattas huruvida projektet skall fortgå.

Begreppet kvalitetsgrindar dök för första gången upp hos NASA på 60-talet, men grindarna användes då främst som ett kontrollverktyg av amerikanska myndigheter snarare än som en del av en utvecklingsprocess (Valeri, 2004). Nästa generation av kvalitetsgrindar blev känd genom den modell Cooper (1990) tog fram för produktutveckling, Stage-Gate®-modellen. Denna modell är än idag den mest tillämpade och används inom många olika industrier, bland annat transport-, och kemiindustrierna men även för finansiella tjänster och i mindre konsumentföretag (Valeri, 2004; Phillips, 1999; Shaw et al., 2001). I Nordamerika använder så mycket som 80-85% av produktutvecklingsföretagen grindkonceptet (Cooper, 2008).

⁶ Engelska ”quality gates” och ”decision gates”

Sedan början på 90-talet har ett antal studier av organisationer med tillämpade grindssystem genomförts. Syftet har varit att öka förståelsen för vad som leder till framgång i produktutveckling och framförallt hur grindar används som beslutsstöd för att lägga ner olönsamma projekt (Hart, 2003). Studierna visar även att mer forskning inom området behövs och att många organisationer tillämpar systemet på ett felaktigt sätt vilket leder till ineffektiva grindar (Cooper, 2008, Schmidt, 2005). Trots att resultaten visar många fördelar med användningen av grindkonceptet, visar de även att rätt förutsättningar krävs för att lyckas. Exempel på områden som behöver utforskas närmare är utformningen av kriterier, flexibilitet i grindar samt tydligt definierade roller.

Denna fallstudie ger en beskrivning av hur kvalitetsgrindar och beslutsgrindar används och utformas idag för att säkerställa att delresultaten håller tillräcklig kvalitet. Inom Försvarsmakten pågår ett kontinuerligt utvecklingsarbete av förband och materielsystem. Syftet med denna fallstudie är att tillhandahålla Försvarsmakten kunskap avseende hur kvalitets- och beslutsgrindar tillämpats och vilka erfarenheter som finns. Resultaten är uppdelade i två delar, varav den första beskriver den övergripande processen för ett grindsystem och den andra är en genomgång av tidigare forskning på området. Detta avslutas sedan med en diskussion som bland annat tar upp de vanligaste problemen med implementering och användning av grindsystem.

5.7.2 Metod

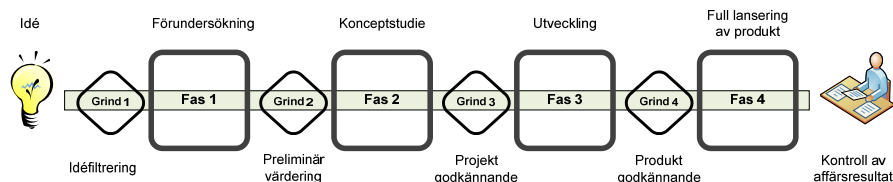
Denna fallstudie är baserad på en litteraturstudie. Sökord som använts är ”decision gates” och synonymer till denna term. Under arbetets gång har ett antal benämningar på grindar stötts på, exempelvis ”quality gates”, ”stage gates”, ”phase gates” och ”development gates”. Några konsekventa skillnader på dessa begrepp har inte identifierats och samtliga benämningar i denna fallstudie har därför översatts till kvalitetsgrindar.

5.7.3 Resultat

Resultaten från litteraturstudien är uppdelad i två delar. Del 1, *Den övergripande processen* beskriver på en konceptuell nivå de grundläggande delarna av ett grindsystem. Beskrivningen bygger på den ursprungliga Stage-Gate®-modellen, framtagen av R.G. Cooper (1990), en modell som senare legat till grund för många efterföljande varianter (bl.a. Ettlé, 2007; Phillips, 1999; Valeri, 2004). I del 2, *Litteraturöversikt - implementering och användning av grindsystem* presenteras forskningsresultat från studier som tidigare gjorts på tillämpade grindsystem.

5.7.3.1 Den övergripande processen

På konceptuell nivå kan ett system beskrivas som en process bestående av (1) ett antal faser⁷ (Figur 6). I varje fas ingår ett antal aktiviteter för att uppnå de utsatta målen vilket krävs för att man skall kunna gå vidare till nästa grind i utvecklingsprocessen. Processen består även av (2) ett antal grindar, där beslut fattas om huruvida ett projekt uppfyllt de fördefinierade kriterierna och kvalitetskraven.



Figur 6. Konceptuell modell av faser och grindar. Omarbetad från Cooper (1990).

5.7.3.1.1 Faser

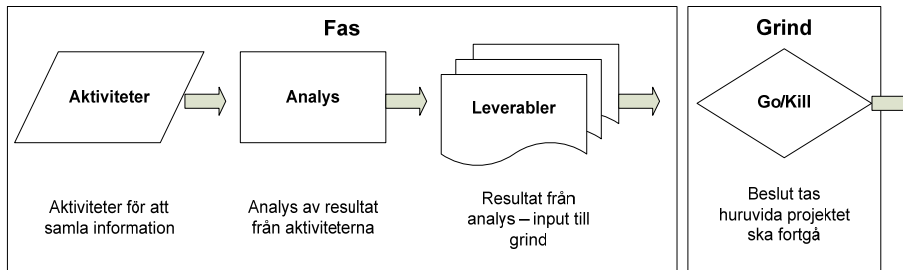
Varje fas består av ett antal aktiviteter som skall utföras med målet att uppnå kraven för nästa kvalitetsgrind (Figur 4). Faserna är tvärorganisatoriska vilket innebär att aktiviteter från olika funktioner sker parallellt. En roll, projektledaren, har huvudansvar för samtliga aktiviteter. Uppgiften kräver god kunskap om vilka kriterier som skall uppfyllas i varje steg. Syftet med att ha en huvudansvarig för hela projektet är att se till att ansvaret inte flyttas mellan avdelningar och grupper i de olika faserna.

5.7.3.2 Grindar

Slutet av varje fas motsvarar en viktig tidpunkt i kvalitetsgranskningen och i besluten, vilka representeras av en kvalitetsgrind. Beslut fattas enligt ett antal fördefinierade kriterier som bland annat innehåller krav på kvalitet, goda utsikter för marknadsföring, en fungerande ekonomisk plan och analys av tekniska möjligheter. Det är grindvakternas uppgift att avgöra huruvida kraven uppfyllts i den föregående fasen och således har grindvakten också till uppgift att fatta beslut om projektets framtid ("go" eller "kill"). Syftet med grindarna är att hindra olönsamma projekt att gå vidare eller att fånga upp svårigheter i ett tidigt skede. Samtliga grindar i ett projekt har en liknande struktur och består av; (1) *input* (eller leverabler), resultaten från det aktuella steget; (2) *kriterier*, de

⁷ Hur många faser en process består av varierar oftast beroende på organisationen, det kan vara upp till tio (Phillips, 1999), men hamnar ofta någonstans mellan fyra och sju (Cooper, 2008).

fördefinierade kraven som skall uppnås; (3) *output*, beslut som fattas vid grinden om huruvida projektet skall fortgå eller inte samt en projektplan för nästa steg i de fall där detta är aktuellt (Figur 7).



Figur 7. Konceptuell modell över en fas och en grind. Omarbetat från Cooper (2008)

5.7.3.3 Grindvakterna

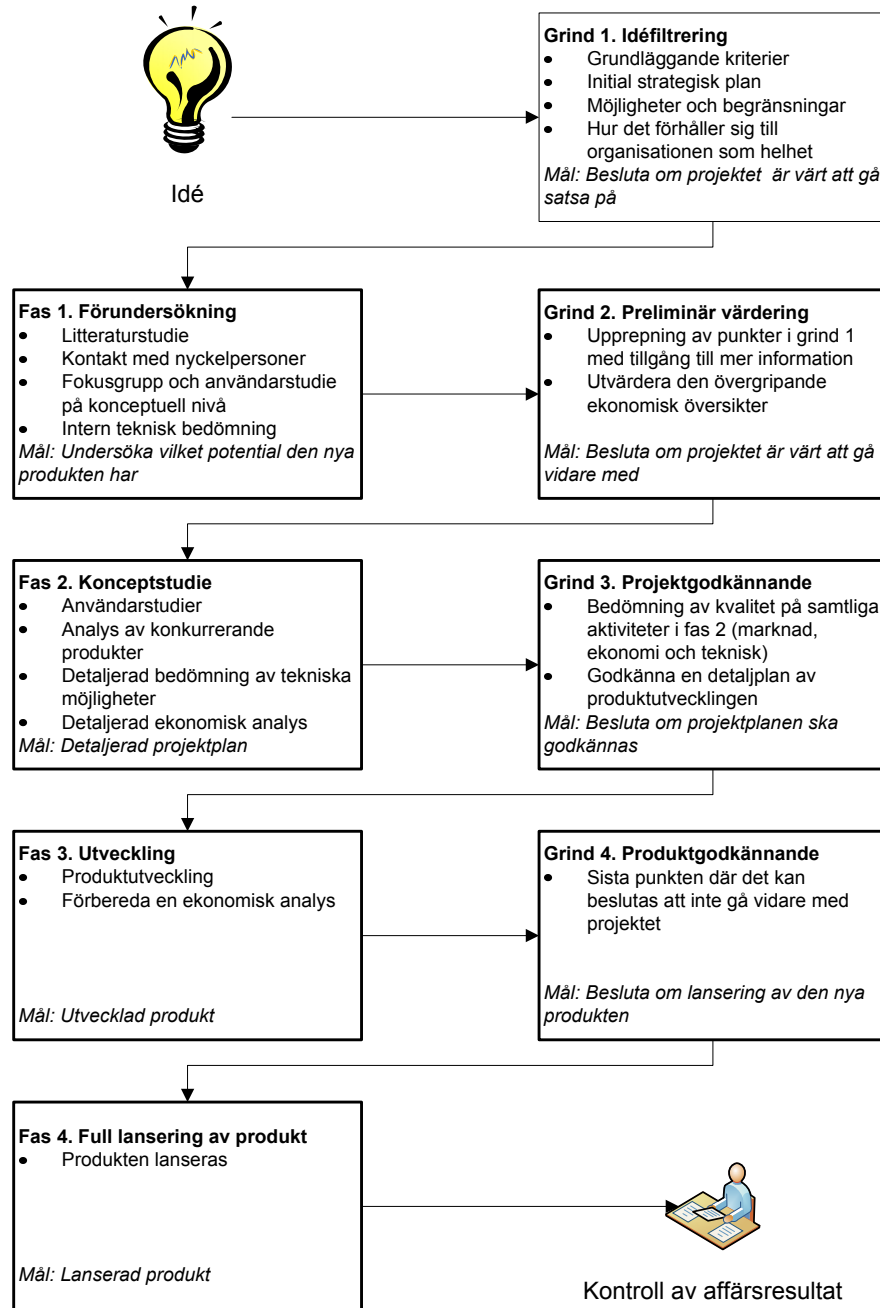
Grindvakterna representeras oftast av den yttersta ledningen, eftersom de måste ha befogenhet att fatta beslut om projektet skall fortgå eller läggas ner. Deras uppgifter är framförallt att:

- Granska kvaliteten på resultaten från aktuellt steg.
- Värdera kvaliteten på projektet utifrån ett ekonomiskt perspektiv.
- Godkänna planen för nästa steg samt avsätta de nödvändiga resurserna. Detta utförs genom att de ger omdömen av typ ”go”, ”kill”, ”hold” eller ”recycle”.

Vid grinden kontrolleras därmed kvaliteten på resultatet från den föregående fasen, projektgruppens arbetsinsats, om projektet fortfarande är attraktivt utifrån en ekonomisk och affärsmässig synvinkel samt om projektplanen är välgrundad. Behöver vissa delar förbättras eller göras om för att passera grinden och gå vidare till nästa fas ges omdömet ”recycle”. Behövs mer information för att ett beslut skall kunna tas ges omdömet ”hold” till dess mer information erhållits.

5.7.3.3.1 Kriterier

Kriterierna består som tidigare nämnts av ett antal krav som rör kvalitet, ekonomi och marknadsföring. Kriterierna är ofta framtagna av grindvakterna snarare än av projektmedlemmarna och dess relevans vid en viss tidpunkt kan skifta. Exempelvis kan tekniska möjligheter och marknadsintresse vara av större vikt i början av ett projekt medan kvalitet och ekonomiska kriterier väger tyngre under de senare delarna (Schmidt, 2005).



Figur 8. Den övergripande processen steg för steg. Bygger på beskrivning av Cooper (1990)

5.7.3.4 Processen steg för steg

En process initieras av en ny produktidé och slutar i bästa fall med lansering denna idé. Efter en tid bör återkoppling ske genom en undersökning av produktens framgång. Vid den första grinden skall ett beslut fattas huruvida en idé är värd att lägga resurser på. Inga ekonomiska kriterier är dock med i denna första filtrering. Målet är att hålla en låg budget i början och främst arbeta med att ta fram underlag för beslut angående produktens potential. Eftersom varje steg i processen generellt sätt kostar mer än de föregående, är en av huvuduppgifterna att inhämta nödvändig information om risker i projektet. I takt med att kostnaderna ökar bör därmed antal risker minskas (Cooper, 2008). En översiktlig beskrivning av aktiviteter som ingår i varje fas finns i Figur 8.

5.7.3.5 Litteraturoversikt

Nedan presenteras resultaten från litteraturstudien i tre delar: (1) En anpassningsbar modell, (2) Grindarnas utformning och flexibilitet samt (3) kriterier.

5.7.3.5.1 En anpassningsbar modell

Ett antal omfattande studier tyder på att användningen av en modell med grindsystem som innehåller ett antal utvecklingsfaser och utvärderingstillfällen är utbredd inom produktutvecklingen (Hart, 2003; Cooper, 2008). Företag världen över modifierar och anpassar faser och grindar till olika organisationsstrukturer och projekt. Riskfaktorer och ett projekts omfattning är exempel på faktorer som kan komma att påverka antalet faser och grindar som används. Denna typ av effektivisering och anpassning är enligt Stage-Gate®'s grundare Robert Cooper (2008) absolut nödvändig för att modellen skall kunna nyttjas på bästa sätt i varje projekt.

Cooper (2008) menar att det inte finns någon aktivitet eller några kriterier som är direkt obligatoriska utan Stage-Gate® innehåller enbart rekommendationer. En typ av anpassning kan t.ex. vara att utförandet av flera aktiviteter, från olika faser, sker simultant. Detta kan innebära att en aktivitet från en senare fas flyttas tillbaka till en tidigare, eller att det finns ett visst överlapp mellan faserna (Cooper, 2008; Schmidt, 2005).

En studie av produktutveckling inom fordonsindustrin visar på att egenmodifierade modeller ökar effektiviteten (Ettlie, 2007). Huvudsyftet med studien var att undersöka hur Stage-Gate® användes vid innovativ produktutveckling som utnyttjar ny avancerad produktutvecklingsteknik, i form av t.ex. virtuella möten. Projektledare i 72 olika projekt tillfrågades och utav dem uppgav 48,6 % att de arbetade utefter den traditionella Stage-Gate®-modellen och omkring 33 % ansåg att de använde en modifierad version av den ursprungliga Stage-Gate®-modellen. Resultaten visade att företag som arbetar

mer decentraliserat och utnyttjar virtuella verktyg i högre grad använde en modifierad modell, mer anpassad efter förutsättningarna i projektet. Resultaten visar även att företagen som anpassat sin modell i högre grad optimerade avvägningen mellan kostnad och kvalitet och att de i allmänhet minskade produktionstiden med 50 % utan att kvaliteten minskades.

Dessa resultat kan jämföras med en studie där en utvärdering av sex olika välkända företag som använder Stage-Gate®, bl.a. Kodak, General Electric och Bombardier (Phillips, 1999) utförts. Varje modell jämfördes mot den generiska fyrstegsmodellen, ursprungligen framtagen av Cooper (Figur 6). Resultaten visade att samtliga företag i hög utsträckning följde Stage-Gate®'s ursprungliga principer. En iakttagelse var att tvärorganisatoriska företag i högre grad höll sig till den mest grundläggande fyrstegsmodellen medan organisationer med en starkare funktionell struktur ökade både antalet faser och antalet grindar i varje fas. Detta kan bero på att en tvärorganisatorisk organisation har fler informella granskningar (ibid.).

Studien av Phillips (1999) inkluderar inga resultat om vilken organisationsstruktur som har mest framgång med avseende på sina produkter⁸ men en diskussion förs kring det faktum att ett ökat antal faser ger möjlighet att betona vissa faser som är särskilt avgörande för det aktuella projektet och att detta kan öka konkurrenskraften. Det utökade antalet faser placerades generellt sätt omkring koncept- och utvärderingsfasen. Phillips (1999) föreslår att det ökade antalet faser och grindar snarare bör finnas omkring utvecklingsfasen än utvärderingsfasen, detta eftersom en majoritet av budgeten spenderas här.

Ett exempel på en anpassning av Coopers ursprungliga modell (Figur 6) är Exxon Research and Engineering Companys (ERE) förslag (Cohen, 1998). Förslaget bygger på att utvidga processen till att även inkludera grundforskning. Ett införande av grindkonceptet redan i grundforskningen förväntades ge möjlighet för decentraliserade forskningsgrupper att skapa en bättre struktur för arbetet. Detta dels genom att samtliga grupper har en gemensam projektplan, dels genom att grindarna skapar spårbarhet samt en möjlighet att påverka projekts riktning (ibid.).

Målet med denna utvidgning av modellen var att bättre kunna nyttja forskningsresultat för kommersiella syften, för att minska kostnaderna och för ökad framgång (Cohen, 1998). Utvidgningen inkluderade ett tillägg av grindar i början av projektet. Forskningsresultaten skulle på så vis ligga till grund för bland annat identifiering av nya möjligheter, utvärdering av nya koncept och planering av utvecklingsfasen (ibid.). Vidare föreslås en komprimering av de två

⁸ En annan studie av Daimler Chrysler i Brasilien visar resultaten att tvärorganisatoriska projektgrupper påverkar det ekonomiska resultatet på ett positivt sätt. Andra egenskaper som hade positiv inverkan på ekonomin var en detaljerad ekonomisk plan, stark ekonomisk ledning samt användandet av verktyg och tekniker för att hålla kostnaderna nere (Ibusuki, 2005).

första grindarna i den ursprungliga modellen. Resultaten visade att det fanns flera fördelar med detta, bl.a. att ledningen involverades i ett tidigt skede, att kvalitetskontrollerna ökade samt att forskarna fick ökad förståelse för konsumenternas behov.

5.7.3.5.2 Grindarnas utformning och flexibilitet

Trots att grindar är centrala för att hantera risker, ofta ekonomiska, har många användare bristande förståelse för dem och enligt Cooper (2008) verkar detta vara den svåraste utmaningen i användningen av ett Stage-Gate®. Är grindarna ineffektiva kommer även utvecklingsprocessen att vara det och möjligheten att fatta ett korrekt beslut om projektets framtid försvåras markant. Det finns en rad faktorer vilka kan påverka grindarnas effektivitet, bl.a. flexibilitet, grindvakternas involvering och ansvarsfördelning (bl.a. Cooper, 2008; Flohr, 2008; Valeri, 2004).

Grindar utan tänder

Grindarnas utformning beskrivs som den svagaste punkten för många som använder Stage-Gate® och det vanligaste problemet är att grindarna inte har några så kallade ”tänder” (Cooper, 2008; Schmidt, 2005). Med detta menas att ingen sätter stopp för projekt utan att när de väl passerat den första grinden så rullar de bara på. Problemet, menar Cooper (2008), är att enbart den först grinden används som ett ”go”- alternativt ”kill”-beslut, och de övriga grindarna används som ett möte för uppdatering av projektets utveckling.

Grindvakter

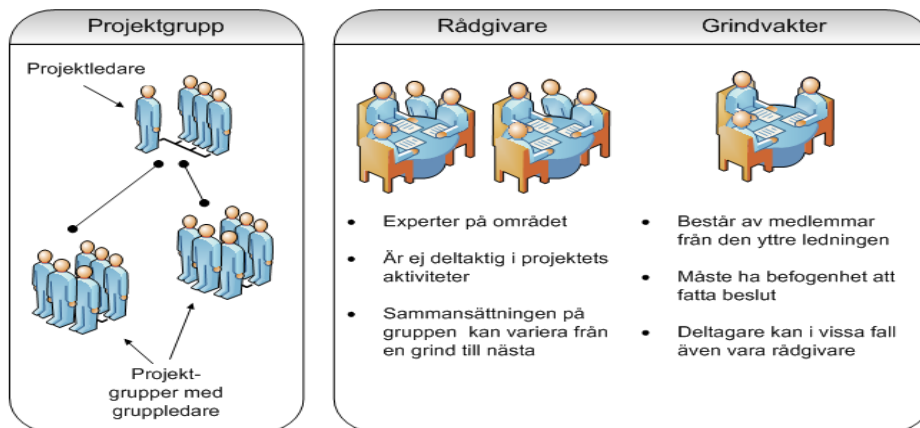
Grindvakter består oftast av medlemmar från den yttersta ledningen, dvs. de som äger nödvändiga resurser för att ett projekt skall kunna fortgå. Att välja rätt grindvakter är ett vanligt förekommande problem, vilket leder till oklarheter kring ansvar och befogenheter (Cohen, 1998; Cooper, 2008; Schmidt, 2005). Cooper (2008) menar att grindvakter bör vara tvärfunktionella och representera olika delar av företaget, såsom marknadsföring, försäljning och teknik. Gruppen skall heller inte ha för många medlemmar utan vara begränsad till enbart de nödvändigaste. Schmidt (2005) poängterar hur viktigt det är att en skillnad görs mellan deltagare i projektet, de som utför aktiviteterna, och de som är grindvakter. Cohen (1998) menar att utöver att tillhandahålla strategisk insikt måste de kunna uppmuntra kreativitet, upptäcka risker och se till att projektet håller rätt fokus. En annan lärdom var att det är nödvändigt att vara flexibel; inte alla grindar och steg är nödvändiga i samtliga projekt utan bör anpassas. Ytterligare en svårighet som identifierades av Cohen (1998) var att grindvakter har dubbla roller. Utöver att fatta beslut om ett projekts framtid skall de även se till att rätt förutsättningar finns för att uppnå nödvändiga krav, identifiera potentialen hos nya idéer samt vägleda projektgruppen så den hamnar på rätt spår.

Under ett möte skall resultaten från den aktuella fasen enbart presenteras ytligt. En rapport med resultaten bör ha distribuerats före mötet så att grindvakterna är väl förberedda. Utöver detta behöver grindvakterna ha kunskap om andra projekt i företaget för att kunna fatta beslut om vilka projekt som skall prioriteras (Schmidt, 2005).

De vanligast förekommande problemen som identifierats med grindvakter är (Cooper, 2008) att (1) styrelsen utser "favoritprojekt" som tilldelas fler fördelar, ofta i form av svaga grindkontroller; (2) grindmöten ställs in i sista minuten; (3) grindmöten hålls men inga beslut fattas; (4) nyckelpersoner saknas vid mötet och beslut kan därför inte tas; (5) endast en grindvakt finns; (6) "go" och "kill" beslut grundas på åsikter eller spekulationer, alternativt personliga intressen.

Ansvar och roller

Svårigheter med arbetsfördelning och ansvar gäller inte bara för grindvakter. I en studie av Busy och Payne (1998) intervjuades projektledare inom en stor försvarskoncern för vapentillverkning. Studien visade att projektledarnas bedömning av hur mycket tid som skulle läggas på en aktivitet varierade kraftigt. De fann även att deras bedömning av tid för olika aktiviteter inte alltid överensstämde med den bedömning som gjordes av medarbetarna i projektet. Projektledarnas bedömning byggde ofta på deras individuella mål som i sin tur influerades av deras roll i organisationen. Vidare visade resultaten att erfarna utvecklare var mer tidspessimistiska, producerade mindre detaljrika projektplaner och i högre grad konsulterade kollegor. Generellt sett visade resultaten på att erfarenheterna av området inte nödvändigtvis innebar god projektplanering.



Figur 9. Grupper vid grindar. Omarbetad från Valeri (2004).

Shaw et al. (2001) implementerade Stage-Gate® i kemiindustrin och fann att medarbetarna hade något av en negativ attityd till modellen och kände sig påtvingade densamma. Samarbetet mellan de som lägger upp och planerar projekten och de som utför arbetet fungerade inte under själva implementeringen. Shaw et al. (2001) tror att metoden på sikt kan resultera i stora tidsvinster för företaget, utan att äventyra möjligheten för nya innovativa idéer. I en modell framtagen av Valeri (2004) ges ett förslag på hur rollfördelningen kan se ut och även hur denna kan användas för att grindarna skall vara mer flexibla (Figur 9). Modellen bygger i grunden på den ursprungliga Stage-Gate®, dock med fokus på att grindarna skall vara flexibla och motverka flaskhalsar i utvecklingsprocessen, ett problem som rapporterats av användare av Stage-Gate®. Modellen har implementerats på två företag med existerande NPD⁹ processer. Resultatet från implementeringen uppskattades av båda företagen. Det mest intressanta i den föreslagna modellen var fördelningen av roller och ansvar, och förslaget att tillåta att en fas påbörjas innan den föregående är avslutad (Figur 9).

Enligt modellen av Valeri (2004) deltar tre skilda grupper i processen: (1) projektgrupp, (2) rådgivare och, (3) grindvakter. Projektgruppens huvuduppgift är att planera och genomföra projektet. Rådgivarna består av experter och styrelsemedlemmar som inte är direkt involverade i projektet och grindvakterna är de personer som har befogenhet att fatta beslut angående projektet. Medlemmar i den sistnämnda gruppen skall ha en helhetssyn över alla projekt för att kunna prioritera rätt projekt.

För att grindarna skall vara flexibla föreslår Valeri (2004) att faserna till viss del kan tillåtas överlappa varandra. Detta bör ske systematisk och inte resultera i att kriterier förbises. Enligt modellen sker en referentgranskning under slutskedet av fasen och när detta är fullbordat kan nästa fas påbörjas. Detta hindrar att flaskhalsar uppstår om inte alla aktiviteter och projektgrupper är klara vid samma tidpunkt. Då referentgranskningen är helt klar hålls ett möte med projektledaren, rådgivarna och grindvakterna där ett slutgiltigt beslut fattas.

Rekommendationer

En lista på rekommendationer för att grindsystemet skall fungera bättre har tagits fram av Schmidt (2005):

- Grindvakterna skall utgöras av andra befattningshavare än de som utför aktiviteterna i projektet för att bedömningen skall bli objektiv och dessutom krävs en beslutsfattare som har befogenheter att fatta avgörande beslut.
- Tydligt definierade kriterier skall finnas tillgängliga för samtliga involverade i projektet.

⁹ New Product Development (NPD)

- Beslut för ”go” eller ”kill” skall tas av en grupp grindvakter från olika expertområden och inte enbart vara en persons ansvar.
- Grindvakter skall roteras, det är mindre troligt att en grindvakt som gett ”go” vid en grind kommer att terminera ett projekt vid nästa grind.
- Övervakningen skall öka allteftersom mer information blir tillgänglig och kostnaderna går upp.
- Virtuella beslutsmöten kan hållas, forskning pekar mot att grindvakter fattar bättre beslut om de inte är i samma rum.

5.7.3.5.3 Kriterier

Vid varje grind sker en utvärderingsprocess där ett visst antal fördefinierade kriterier skall ha uppnåtts. Exakt hur dessa kriterier skall utformas och huruvida det är samma kriterier som används vid varje grind i beskrivningarna finns det i nuvarande modeller inga tydliga regler redogjorda för (Hart, 2003). Generellt sätt pekar de flesta resultaten mot att de flesta använder olika kriterier vid varje grind (Flohr, 2008; Valeri, 2004).

Det finns olika strategier för hur kriterier kan användas, t.ex. kan flexibla kriterier innebära att kriterier delas upp i en hierarkisk kategorisering där vissa kriterier *måste* uppfyllas till en viss grind medan andra *bör* uppnås, vilket i praktiken innebär att dessa i vissa fall kan senareläggas (Schmidt, 2005).

Andra exempel på strategier är att antingen ha generella kvalitetsrekommendationer som kan användas i samtliga projekt inom ett företag, eller ha flexibla kriterier som kan omdefinieras och anpassas till varje enskilt projekt (Tabell 3) (Flohr, 2008). Fördelen med den förstnämnda är att resultaten mellan projekten kan jämföras och arbetsinsatsen inte blir lika omfattande för varje enskilt projekt. Nackdelen är att kriterierna måste anpassas till olika projekt och därmed blir mer abstrakta. Är det samma kriterier för samtliga projekt föreslår författarna att en kriteriekatalog utnyttjas. Denna katalog används i samtliga projekt men ger viss möjlighet att bortse från eller lägga till några kriterier för att passa projekts förutsättningar. Detta ökar spårbarheten och möjligheten att jämföra resultat mellan olika projekt. Fördelen med att ha mer löst definierade kriterier vid kvalitetsgrindarna är att dessa i större utsträckning kan anpassas till projektet, allteftersom mer information blir tillgänglig. Omfattningen av ledningens arbete blir heller inte lika stor (Tabell 3). Vidare diskuterar Flohr (2008) huruvida kriterierna skall vara tydligt definierade i ett tidigt skede av ett projekt. Fördelen med detta är att ledningen då har större möjlighet att påverka. Utöver detta kommer kriterierna att bli tydligare för utvecklarna och ha större inverkan på projektet som helhet.

Tabell 3. Fördelar och nackdelar med kriteriekatalog. Omarbetad från Flohr (2008).

	Fast kriteriekatalog	Ingen kriteriekatalog
Fördelar	<ul style="list-style-type: none"> - Kvalitativ policy för samtliga projekt - Definiering av kriterier enbart en gång - Viss möjlighet att skräddarsy efter projektets behov 	<ul style="list-style-type: none"> - Kriterier perfekt anpassade efter projektets behov
Nackdelar	<ul style="list-style-type: none"> - Svårare att definiera kriterierna eftersom de ska anpassas till samtliga projekt - Lägre acceptans hos projektdeltagarna - Kriterier måste tolkas med hjälp av externa resurser 	<ul style="list-style-type: none"> - Generellt sätt krävs mer resurser för definiering av kriterier eftersom de måste definieras för varje enskilt projekt - Enskilda definitioner försvårar möjligheten att jämföra projekt

Flohr (2008) menar även att en metod för att definiera kriterier krävs. Denna bedömning baseras på resultaten från en studie av ett stort antal studentgrupper. Resultaten visade att då kriterier enbart skapades utifrån tidigare erfarenheter, vilket var mest tidseffektivt, gav detta inte något bra kriterieunderlag. Tvärtom uppfyllde en sådan strategi enbart hälften av alla nödvändiga kriterier och dessutom skapades många triviala kriterier (Flohr, 2008). Författaren föreslår att kriterier bör tas fram systematiskt utifrån abstrakta företagsmål. Detta kan göras genom att applicera en kvalitetsmodell eller metod, exempelvis GQM (Goal-Question-Metric) (ibid.). Detta säkerställer högre kvalitet men kräver mer omfattande resurser i början av projektet.

Hart (2003) har gjort en omfattande studie av hur kriterier används vid kvalitetsgrindar inom NPD. Studien involverade 166 projektledare för företag som utvecklar och tillverkar industrivaror. Idag används en mängd olika tekniker inom NPD för att ta fram kriterier¹⁰ och enligt författarna krävs, utöver de kommersiella utvärderingsteknikerna, även tekniska utvärderingar vid varje grind, som säkerställer kvalitén hos produkten med avseende på funktion, form och produktion.

Resultaten visade att användningen av kriterier varierade mycket mellan företagen, men vissa trender gick att utläsa. Generellt sett vägde tekniska möjligheter, intuition och marknadsundersökningar tungt vid de första grindarna medan kriterier relaterade till produktens prestation, kvalitet och att hålla sig inom budget var mer avgörande i de senare faserna. Vissa kriterier användes enbart vid någon enstaka grind, t.ex. hur unik produkten är eller hur intuitivt

¹⁰ Exempel på tekniker var beta-testing, perceptuell mapping, Quality Function Deployment (QFD) (Hart et al, 2003)

tilltalande produkten är. Andra kriterier återkom vid de flesta grindar, t.ex. kundnöjdhet, marknadspotential och produktens övergripande prestanda. För en mer detaljerad beskrivning av de vanligaste kriterierna vid varje grind se Hart (2003). Ett intressant resultat från studien var att liknande mönster av kriterieanvändning återfanns i utvärderingarna oavsett företagets storlek.

Hart (2003) menar att studien presenterar en beskrivning av hur användningen av kriterier ser ut idag och att en viktig fråga för framtiden är att förstå vilken betydelse de har vid varje grind. Kontextuella variabler som tidspress och hög konkurrens måste tas med i beräkningen i en sådan studie.

Gorschek och Wohlin (2006), med bakgrund i systemutveckling, har tagit fram en modell som skall fungera som stöd för kommunikation mellan olika aktörer delaktiga i en utvecklingsprocess, RAM (Requirements Abstraction Model). Målet var att skapa en möjlighet för hantering av krav på flera abstraktionsnivåer. Detta gjordes genom att krav på olika nivåer omarbetades antingen uppåt genom att beskriva dem på en högre abstraktionsnivå med avseende på produktionsmålet alternativt brytas ner dem till mer konkreta delar. Detta för att säkerställa att krav inte hamnar för långt utanför de övergripande produktionsmålen men även att se till att krav inte är för abstrakta för utvecklarna. RAM föreslår fyra abstraktionsnivåer¹¹ men lämnar antal nivåer öppet för att skräddarsys efter behov.

Under arbetet med modellen framgick att denna arbetsmetod har potential att kartlägga vilken effekt ett visst krav kommer att ha på helheten och vilka problem som kan uppstå. Den ger även en större möjlighet för olika aktörer att diskutera kraven, eftersom dessa finns på flera olika abstraktionsnivåer (ibid.).

5.7.4 Diskussion

Trots att användningen av grindsystem är utbredd och erfarenheten från många företag varit positiv har ett flertal studier visat att svårigheter finns. Den mest använda modellen, Stage-Gate®, beskrivs på en konceptuell nivå och innefattar en hel utvecklingsprocess. Modellen är inte skapad för att mikrostyra en process utan varje organisation skall kunna tolka och anpassa modellen efter sin organisationsstruktur samt kunna utnyttja andra ”best practise” - metoder (Cooper, 2008). Vidare bör hänsyn även tas till organisationsstorlek, budget och projektets omfattning (ibid.) Detta innebär att tillämpningen kan skilja sig mycket mellan olika företag liksom de problem som uppstår.

¹¹ De fyra abstraktionsnivåerna (på engelska): product level, feature level, functional level, component level

5.7.5 Grindar, roller och ansvar

Utifrån studierna som presenterats i denna fallstudie är det vanligaste problemet att grindarna används på ett felaktigt sätt. Svårigheter förknippat med detta är bl.a. val av grindvakter, grindvakternas engagemang samt grindarnas flexibilitet.

Grindvakterna bär ett stort ansvar för att grindarna skall fungera effektivt. De måste vara välinformerade, engagerade och ha god förmåga att se vilka projekt som har goda framtidsutsikter (Cohen, 1998; Cooper, 2008). Saknar grindarna ”tänder”, dvs. att grindvakterna inte sätter stopp för olönsamma projekt blir systemet ineffektivt. För att grindvakterna skall ha möjlighet att fatta ett korrekt beslut krävs att de är välinformerade. Detta innebär dock inte att de måste känna till alla detaljer, en mikrostyrd process blir för byråkratisk och processen förlorar sin effektivitet pga. tidsbrist (Cooper, 2008; Valeri, 2004). Enbart nödvändig information skall därför presenteras för grindvakterna. Ytterligare metoder för att motverka stopp i processen är de förslag som presenterats av Valeri (2004), där projektdeltagarna påbörjar nästa fas medan resultaten granskas av rådgivare och grindvakter.

Dåligt engagemang hos grindvakterna pekar på brister i organisationskulturen och enligt Cooper (2008) är detta ett vanligt problem. Exempel på detta är grindvakterna som är sena, oförberedda eller uteblir från möten. Om inte grindvakterna, vanligtvis medlemmar från den yttersta företagsledningen, tar grindmöten på allvar kan detta skapa en stämning som sprider sig neråt i organisationen.

Många organisationer har svårt att identifiera vilka som skall utses till grindvakter (Cohen, 1998, Cooper, 2008). Det kanske viktigaste är att grindvakterna är objektiva gentemot projektet, dvs. att de inte själva är delaktiga i arbetet med aktiviteterna eller har ett särskilt personligt intresse för det. Detta kan leda till särbehandling av projekt och större svårigheter att sätta stopp för olönsamma projekt (Schmidt, 2005; Valeri, 2004). Grindvakter skall vara personer som har befogenhet att godkänna alternativt sätta stopp för projekten. Det skall även vara personer som har en helhetssyn på verksamheten och utifrån detta kan prioritera mer lönsamma projekt. Detta innebär däremot inte att samtliga medlemmar i ledningen måste vara delaktiga. I en stor organisation kan det senare snarare resultera i att ansvaret faller mellan stolarna.

Schmidt (2005) föreslog att grindvakterna skulle skiftas vid varje grind. Anledningen till detta var att det annars finns risk för att individer som godkänt ett projekt vid ett tillfälle har svårt att säga nej vid nästa tillfälle. Ett annat tips var att hålla virtuella möten, detta eftersom detta minskade risken för att grindvakter godkände olönsamma projekt.

De två sistnämnda förslagen på hur grindar kan hanteras går dock inte helt i linje med de svårigheter som nämnts tidigare, exempelvis att ansvar faller mellan stolarna. Det finns en motsättning i att grindvakterna skall känna ansvar,

inspiration, engagemang samt vara insatta i varje projekt samtidigt som de byts ut vid varje beslutspunkt eller enbart medverkar i virtuella möten. En avvägning mellan alla dessa faktorer måste därför göras baserat på organisationens förutsättningar.

5.7.6 Kriterier

Hur kriterier skall utformas och vilka kriterier som är viktiga är en annan svårighet vid tillämpning av ett grindsystem. Är inte kriterierna av tillräckligt god kvalitet kommer heller inte produkten vara det. Detta kommer även att försvåra arbetet för utvecklarna samt för grindvakterna som måste göra bra utvärderingar av resultaten.

Genom att utnyttja kriteriekataloger med bestämda kriterier som är till för samtliga projekt kan spårbarheten ökas. Detta innebär att kvaliteten på olika produkter kan jämföras. Problemet med detta är dock att kriterier som skall anpassas till varje projekt blir för abstrakta och måste tolkas utifrån varje enskilt projekt (Flohr, 2008).

5.7.7 Slutsatser

För att få ett system med kvalitetsgrindar att fungera krävs många samverkande faktorer. Antalet faser och grindar skall anpassas till den specifika organisationen och varje projekts behov. Det krävs även en god organisationskultur, tydliga kriterier och god kommunikation mellan de olika aktörerna.

Många frågor kvarstår dock, framförallt behövs en bättre förståelse för hur grindvakternas roll skall definieras hur grindarna skall bli mer effektiva.

Kriteriers utformning är en annan viktig utmaning som idag ännu inte har några tydliga riktlinjer.

5.8 Fallstudie: Databas: Metoder och principer för utveckling av ledningssystem

För att dokumentera och tillgängliggöra den kunskap som byggs upp inom projektet *Arkitekturbaserad ledningssystemsutveckling*, skapas en kunskapsdatabas vars syfte är att lagra vetenskapligt kvalitetssäkrade principer och metoder avseende systemutveckling.

För att tillgängliggöra ovan nämnda metoder och principer utvecklas ett antal gränssnitt mot kunskapsdatabasen. Exempel på sådana gränssnitt är:

- Inmatning/underhåll
- Sökfunktion, beslutsstöd
- Handboksgenerering

5.8.1 Inledning

Under projektets gång har det framkommit ett önskemål om att lättare kunna spara och tillgängliggöra kunskap och kompetens som erhålls inom olika studier.

Syftet med denna fallstudie är att ta fram en databas för lagring och tillgängliggörande av de erfarenheter och den kunskap som byggs upp inom projektet, samt att skapa ett systemstöd för att administrera och söka bland denna information. Målbilden på sikt med detta arbete är att tillhandahålla Försvarmakten en källa till vetenskapligt och praktiskt kvalitetssäkrad information avseende metoder och principer för systemutveckling.

Målet med arbetet är att vid Q3 2010 ha en körbar kunskapsdatabas, i form av ett systemstöd, innehållande vetenskapligt kvalitetssäkrade metoder och principer för ledningssystemsutveckling.

5.8.2 Metod

Fallstudien genomförs i följande steg:

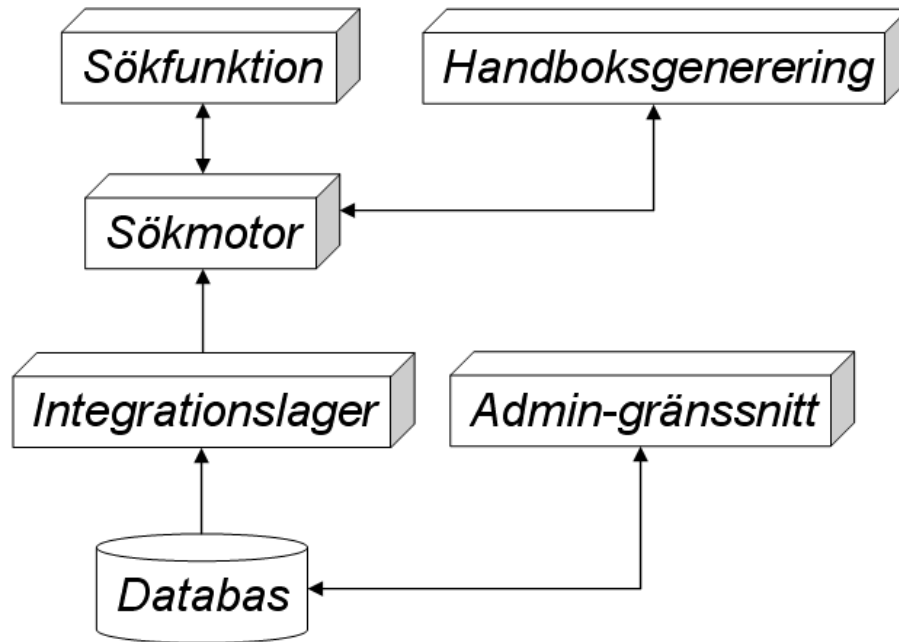
1. Framtagande av prototyp (Q3 2009)
2. Vidareutveckling av databasschema (Q4 2009)
3. Implementering av systemstöd (Q2 2010)
4. Informationsinhämtning och testning (Q3 2010)

5.8.3 Resultat

I följande avsnitt presenteras, på en konceptuell nivå, kunskapsdatabasen samt de gränssnitt för inmatning/underhåll, sökning samt handboksgenerering som identifierats.

5.8.3.1 Arkitektur

Figur 9 ger en översiktlig bild över systemstödet uppbyggnad. Aktuella teknologier har medvetet utelämnats.



Figur 9. Systemskiss systemstöd för arkitekturbaserad ledningssystemutveckling

5.8.3.2 Datamodell

Eftersom den *fysiska* datamodellen använder sig av engelsk notation, skrivs dessa efter respektive klass/egenskap. Två övergripande klasser har identifierats; *metod* och *princip*. Dessa är likartade men den väsentliga skillnaden att en *metod* kan ha *indata* (input) och generera ett *resultat* (result). I övrigt är dessa klasser identiska och innefattar följande attribut:

Namn (*method/principle*)

Beskrivning (*description*)

Syfte (*purpose*)

Skapande datum (*created*) – tidstämpel då objektet skapades

Modifierings datum (*modified*) – tidstämpel då objektet senast förändrades

Utöver dessa klasser förekommer ytterligare ett antal *beskrivande* klasser som vid behov kan associeras till metoder och principer, för att på så sätt mer utförligt beskriva det aktuella objektet.

För den underliggande datamodellen gäller generellt att de beskrivande objekten kan associeras till flertalet olika övergripande objekt (metoder/principer). Utöver detta gäller att de övergripande objekten kan ha godtyckligt antal associationer till andra övergripande objekt. På så sätt är det enkelt att åskådliggöra beroenden eller referenser mellan olika metoder/principer.

5.8.3.2.1 Referens (*reference*)

Klassen Referens beskriver de källreferenser som nyttjas av de lagrade metoderna/principerna. De egenskaper som kan anges för en referens är:

Titel (*title*)

Författare (*author*)

Källa (*source*)

Sidreferens (*pageno*)

5.8.3.2.2 Notation (*notation*)

Klassen Notation beskriver vilken typ av notation som kan appliceras på en given metod/princip.

Namn (*notation*)

5.8.3.2.3 Förutsättning (prerequisite)

Klassen Förutsättning Beskriver vilka förutsättningar som gäller för en given metod/princip.

Namn (*prerequisite*)

5.8.3.2.4 Genomförande (implementation)

Klassen Genomförande Beskriver vilka genomförandesteg som gäller för en aktuell metod/princip.

Namn (*implementation*)

5.8.3.2.5 Aktivitet (activity)

Klassen Aktivitet Beskriver vilka aktiviteter som en given metod/princip kan appliceras på.

Namn (*activity*)

5.8.3.3 Administrationsgränssnitt

För att kunna fylla databasen med information, samt underhålla denna, implementeras ett administrationsgränssnitt mot databasen. Via detta gränssnitt skall det vara möjligt för administratörer att skapa, modifiera samt ta bort olika objekt i databasen på ett strukturerat och säkert sätt.

5.8.3.4 Integrationslager

För att överföra informationen från databasen till sökfunktionen utvecklas ett *integrationslager*. Syftet med detta lager är att:

- *Denormalisera* den underliggande, normaliserade, datamodellen
- *Transformera* den denormaliserade informationen för att passa schemat i *Solr*

Resultatet av kombinationen av databas/integrationslager/sökfunktion är att användaren kan:

- I realtid få förslag på existerande objekt i databasen vid inmatning av sökbegrepp
- Se objektrelationer t.ex. ”vilka metoder/principer har referenser till denna rapport?”
- Se vilka metoder/principer som är beroende av t.ex. en *förutsättning* eller ett *genomförandesteg*
- Se vilka metoder som är relaterade till en given metod och/eller princip

5.8.3.5 Sökfunktion

För att underlätta åtkomsten till objekt lagrade i databasen har möjligheten att erbjuda en sökfunktion undersökts. Anledningen till detta är att en relationsdatabas, såsom den som använts för att upprätta den fysiska datamodellen, är väl lämpad för *lagring* av objekt samt för att *upprätthålla regelverket mellan dessa* (referentiell integritet) men är i dagsläget mindre lämpad för *fritextsökning*. Därav, för att kunna erbjuda kraftfulla möjligheter till fritextsökning, har sökmotorn *Solr* valt att användas.

5.8.3.6 Solr

Valet av *Solr* är baserat på erfarenheter samt en omfattande global acceptans. *Solr* har visat på god förmåga till skalbarhet och driftsäkerhet. Utöver detta erbjuder den funktionalitet som kommer väl till pass i ett system som detta. *Solr* är dessutom modulärt uppbyggd vilket gör det möjligt att tämligen enkelt utöka sökmotorn med ytterligare funktionalitet om det blir nödvändigt. I dagsläget bedöms dock att den erbjuder långt mer funktionalitet än vad som kan behövas inom ramen för detta projekt.

5.8.3.7 Handboksgenerering

Det väsentliga syftet med projektet är att stödja Försvarsmakten med kunskap i dess strävan att nyttja modellbaserad utveckling vid framtagandet av nya ledningssystem. Ett led i att stödja detta är, i stor utsträckning, att kunna automatisera sammanställandet av riktlinjer baserat på den kunskap som finns representerad i databasen. Användare skall därför, utifrån ett antal angivna kriterier kunna beställa ett utdrag ur databasen för att automatiskt generera en enklare form av handbok, i form av t.ex. ett Word-dokument.

5.8.4 Diskussion

Det som återstår i arbetet med att vidareutveckla sökfunktionen, implementera administrationsgränssnittet till databasen samt att fylla databasen med information. När detta är genomfört följer sedan tester av systemstödet i form av tekniska tester såväl som lättare användartester.

6 Referenser

- Ahlberg, S, Bergman, J., Dahlman, J., Folkesson, M., Forsgren, R., Hallberg, N., Hasewinkel, H., Hedström, J., Johansson, M., Kylesten, B., Söderman, U., Törne, A., & Ölvander, C. (2006) *Teknik för ledning av insats i urban miljö - TEBE II slutrapport*, FOI 2006, FOI-R--2176—SE.
- Allberg, H., Kylesten, B., & Pilemalm Sofie (2009) *Utvärdering av modellbaserad förmågeutveckling - påverkan på personalens förståelse, kunskap och arbetssätt - Analys av uppgift och beskrivning av initial projektplan*. FOI 2009, FOI Memo 2808.
- Avison, D. E. & Fitzgerald, G. (1995) *Information Systems Development: Methodologies, Techniques and Tools*. The McGraw-Hill Companies.
- Balasubramanian, K., Gokhale, A., Karsai, G., Sztipanovits, J., & Neema, S. (2006) Developing Applications Using Model-driven Design Environments. *Computer*, 39, 2, 33-40.
- Busby, J. S. & Payne, K. (1998). The Situated Nature of Judgment in Engineering Design Planning. *Journal of Engineering Design*, 9(3):273-93 (September).
- Carlock, P. G. & Fenton, R. E. (2001) System of systems (SOS) enterprise systems engineering for information-intensive organizations. *System Engineering*, 4, 4, 242-261.
- Cohen, L. Y., Kamienski, P. W., & Espino, R. L. (1998). Gate system focuses industrial basic research. *Research Technology Management*, 41(4), 34-37.
- Collin B. (2003) *IT-kvalitet: Verksamhets- & Effektivitetsutveckling*. Studentlitteratur, Lund Sweden.
- Cooper, K. K. (2001) *Rapid Prototyping Technology. Selection and Application*. Marcel Dekker Inc, New York.
- Cooper, R. G. (1990). Stage-gate systems: A new tool for managing new products. *Business Horizons*, 33(3), 44-54.
- Cooper, R. G. (2008). Perspective: The stage-gates® idea-to-launch process - update, what's new, and NexGen systems. *Journal of Product Innovation Management*, 25(3), 213-232.
- Daneva, M. (2003) Lessons Learnt from Five Years of Experience in ERP Requirements Engineering, *Proceedings of the International Conference of Requirements Engineering*, Los Alamitos, CA.
- Daniels, J. (2002) Modeling with a sense of purpose. *IEEE Software*, 19, 1, 8-10.

- Ettlie, J. E. (2007). Modified stage-gate® regimes in new product development. *Journal of Product Innovation Management*, 24(1), 20-33.
- Fardoun, H. Montero, F., & Jaquero, V. L. (2009) eLearnXML: Towards a model-based approach for the development of e-Learning systems considering quality, *Advances in Engineering Software*. In press.
- Flanagan, J. C. (1954). The Critical Incident Technique. *Psychological Bulletin*, 51, 327-58.
- Flohr, T. (2008). Defining suitable criteria for quality gates. In Lecture Notes In Computer Science; 5338 (245-256). *Proceedings of the International Conferences on Software Process and Product Measurement*. Springer-Verlag Berlin, Heidelberg.
- Friedenthal, S., Moore, A., & Steiner, R. (2008) *A Practical Guide to SysML: The Systems Modeling Language*. Morgan Kaufmann. Amsterdam.
- Gitzel, R. & Korthaus, A. (2004) The Role of Metamodeling in Model-Driven Development. *Proceedings of the 8th World Multi-Conference on Systemics, Cybernetics and Informatics (SCI2004)*.
- Gorschek, T., & Wohlin, C. (2006). Requirements abstraction model. *Requirements Engineering*, 11(1), 79-101.
- Hallberg, N. & Fransson, J. (2001) *UML baserad metamodel för modellering av bekämpningskedjor*. Linköping, FOI 2001, (FOI-R--0159--SE).
- Hallberg, N. Andersson, R. & Ölvander, C (2009) Agile Architecture Framework for Model Driven Development of C2 Systems. *J Systems Engineering*. In press.
- Hallberg, N., Andersson, R., & Westerdahl, L. (2005) *Quality-driven process for requirements elicitation: the case of architecture driving requirements*. Linköping, FOI 2005, (FOI-R--1576--SE).
- Hallberg, N., Pilemalm, S., & Westerdahl, L. (2008) *Behovsanalys avseende Försvarsmaktens utveckling av ledningssystem*. FOI 2008, FOI Memo 2443.
- Hallberg, N., Pilemalm, S., Westerdahl, L., Jungert, E., Eriksson, H., Andersson, L. & Lindmark, F. (2008) *Principer och metoder för systemutveckling*. FOI 2008, FOI-R--2628--SE.
- Hallberg, N., Sparf, M., Sjödin, L. och Pilemalm, S. (2009) *Modellbaserad utveckling: Omvärldsanalys*. FOI MEMO 2842.
- Hallberg, N., Timpka, T. & Eriksson, H. (1999) The Medical Software Quality Deployment Method. *Methods of Information in Medicine*, 38, 1, 66-73.
- Hallberg, N., Ölvander, C., & Törne, A. (2006) *Behovs- och kravanalys avseende tekniskt beslutsstöd för operationer i urban terräng*. Linköping, FOI 2006, (FOI-R--2037--SE).

- Harpst, G. (2009) *Executing Revolution, Six Disciplines Publishing*, Findlay, Ohio.
- Hart, S. (2003). Industrial companies' evaluation criteria in new product development gates. *Journal of Product Innovation Management*, 20(1), 22-36.
- Hay, D. C. (2003) *Requirements analysis: From business views to architecture*. Prentice Hall, Upper Saddle River, New Jersey.
- Hull, E., Jackson, K., & Dick, J. (2005) *Requirements engineering*, 2nd edition, Springer, New York.
- Ibusuki, U. (2005). Cost management in the stage-gate system. *AACE International Transactions*, 1-4.
- Jacobson, I., Ericsson, M., & Jacobson, A. (1995) *The object advantage: Business process reengineering with object technology*. Addison-Wesley, Reading, MA.
- Jefferson, T. L. (2006) Evaluating the role of information technology in crisis and emergency management. *The journal of information and knowledge management systems*, 36, 3, 261-264.
- Johanson, P. & Ekstedt, M. (2007) *Enterprise Architecture: Models and analyses for Information Systems Decision Making*. Studentlitteratur, Lund.
- Kapitsaki, G. M., Kateros, D. A., Prezerakos, G. N., & Venieris, L. S. (2009) Model-driven development of composite context-aware web applications. *Information and Software Technology*, 51, 8, 1244-1260.
- Kasser, J. E. 2007. *A Framework for Understanding of Systems Engineering*. The right Requirement, Cranfield UK.
- Kulak, D. & Guiney, E. (2003) *Use cases: Requirements in context*. Addison-Wesley, Upper Saddle River, NJ.
- Kulkarni, V. & Reddy, S. (2008) A model-driven approach for developing business applications: experience, lessons learnt and a way forward. *Proceedings of the 1st conference on India software engineering conference*, 21-28.
- Kühne, T. (2006) Matters of (Meta-) Modeling. *J. Software Systems Model*, 5, 369-385.
- Lee, J. (2005) Model-driven business transformation and the semantic web. *Communications of the ACM*. 48, 12, 75-77.
- Ljungberg, A. & Larsson, E. (2001) *Processbaserad verksamhetsutveckling*. Studentlitteratur AB. Sverige.
- Maier M. W. & Rechtin, E. (2002) *The art of systems architecting*. CRC Press.

- Mellor, S. J., Clark, A. N., & Futagami, T. (2003) Model-Driven Development. *IEEE Software*, 20, 5, 14-18.
- Mendonca, D., Jefferson, T. & Harrald, J. (2007) Collaborative adhocracies and mix-and-match technologies in emergency management, *Communications of the ACM*, 50, 3, 44-49.
- MODAF 2009, Ministry of Defence Architecture Framework, <http://www.modaf.org.uk> (2009-11-30)
- Mohagheghi P., Dehlen, V., & Neple, T. (2009) Definitions and approaches to model quality in model-based software development – A review of literature. *Information and Software Technology*. In press.
- Molina, A. I., Redondo, M. A., & Ortega, M. (2009) A methodological approach for user interface development of collaborative applications: A case study. *Science of Computer Programming*, 74, 754-776.
- Molina, A. I., Redondo, M. A., Ortega, M. & Hoppe, U. (2008) CIAM: A Methodology for the Development of Groupware User Interfaces. *J Universal Computer Science*, 14, 9, 1435-1446.
- Moody, D. L. & Shanks, G. G. (2003) Improving the quality of data models: empirical validation of a quality management framework. *Information Systems*, 28, 6, 619-650.
- Nunes, A. N. & Schwabe, D. (2006) Rapid Prototyping of Web Applications combining Domain Specific Languages and Model Driven Design. *Proceedings of the 6th international conference on Web engineering (ICWE'06)*, 263, 153-160.
- OMG (Object Management Group) (2001) *Unified Modeling Language Specification*, Version 1.4.
- Phillips, R. (1999). Comparative study of six stage-gate approaches to product development. *Integrated Manufacturing Systems*, 10(5), 289-297.
- Porres, I., Domínguez, E., Perez, B., Rodríguez, A., & Zapata, M. A. (2008) A Model Driven Approach to Automate the Implementation of Clinical Guidelines in Decision Support Systems. *Proceedings of the 15th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems (ECBS 2008)*, 210-218.
- Rittel, H. J. W. & Webber, M. M. (1973) Dilemmas in a general theory of planning. *Policy Sciences*, 4, 2, 55-169.
- Romero, J. R., Rivera, J. E., Dur'an, F., and Vallecillo, A. (2007) Formal and Tool Support for Model Driven Engineering with Maude. *J of Object Technology*, 6, 9, 187-207. <http://www.jot.fm/issues/issues 2007 October/paper10>

- Schekkerman, J. (2003) *How to survive in the jungle of enterprise architecture framework: Creating or choosing an enterprise architecture framework*. Trafford, Victoria, BC, Canada.
- Schmidt, J.B. (2005). Gate decisions: The key to managing risk during new product development. In Kahn, K.B. (ed.), *The PDMA Handbook of New Product Development*, John Wiley & Sons, New Jersey, (2nd ed.) pp. 331-348.
- Selic, B. (2003) The Pragmatics of Model-Driven Development. *IEEE Software*, 20, 5, 19-25.
- Shaw, N.E., Burgess, T. F., Hwarng, H. B & de Mattos, C. (2001). Revitalizing New Process Development in the U.K., Fine Chemicals Industry. *International Journal of Operations & Production Management*, 21(8):1133-51.
- Siegel, J. (2005) Why use the model driven architecture to design and build distributed applications? *Proceedings of the 27th international conference on Software engineering*, 37.
- Sommerville, I. (2001) *Software engineering*, 6th edition. Addison-Wesley, Boston, MA.
- Sottet, J. S., Calvary, G., & Favre, J. M. (2006) Towards Model Driven Engineering of Plastic User Interfaces. *Lecture Notes in Computer Science*, 3844, 191-200.
- Sousa, K., Mendonça, H., Vanderdonckt, J., Namahn, E. R., & Vandermeulen, J. (2008) User interface derivation from business processes: a model-driven approach for organizational engineering. *Proceedings of the ACM symposium on applied computing*, 553-560.
- Stirna, J. & Persson, A. (2007) Ten Years Plus with EKD: Reflections from Using an Enterprise Modeling Method in Practice. *Proceedings of the Eleventh International Workshop on Exploring Modeling Methods in Systems Analysis and Design (EMMSAD'07)*, 99-108.
- Tague, N. R. (2005) *The Quality Toolbox*. ASQ Quality Press.
- Valeri, S. G. (2004). Improving the flexibility of new Product development (NPD) through a new quality gate approach. *Journal of Integrated Design & Process Science*, 8(3), 2004-2036.
- Wieggers, K. E. (2003) *Software requirements*. Microsoft press, Redmond WA.
- Young, R. R. (2001) *Effective Requirements Practice*. Addison Wesley, Boston.
- Zachman, J. A. (1987) A Framework for Information Systems Architecture. *IBM Systems Journal*, 26, 3, 276-292.

Ölvingson, C., Hallberg, N., Timpka, T., & Greenes R. A (2002) Using the critical incident technique to define a minimal data set for requirements elicitation in public health, *International Journal of Medical Informatics*, 68, 1-3, 165 – 174.