



# Arkitekturbaserad ledningssystemutveckling

Slutrapport

NIKLAS HALLBERG, HELENA GRANLUND, JOACHIM HANSSON,  
PETER LITSEGÅRD, THOMAS SUNDMARK, SOFIE PILEMALM OCH ANNIE PILEMALM

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](http://www.foi.se)

FOI-R--3079--SE  
ISSN 1650-1942

Användarrapport  
November 2010

**Informationssystem**

Niklas Hallberg, Helena Granlund, Joachim  
Hansson, Peter Litsegård, Thomas Sundmark,  
Sofie Pilemalm och Annie Pilemalm

# Arkitekturbaserad ledningssystemutveckling

Slutrapport

<b>Titel</b>	<b>Arkitekturbaserad ledningssystemutveckling: Slutrapport</b>
<b>Title</b>	<b>Architecture-based C2 systems development</b>
<b>Rapportnr/Report no</b>	<b>FOI-R--3079--SE</b>
<b>Rapporttyp Report Type</b>	<b>Användarrapport</b>
<b>Månad/Month</b>	<b>November/November</b>
<b>Utgivningsår/Year</b>	<b>2010</b>
<b>Antal sidor/Pages</b>	<b>48 p</b>
<b>ISSN</b>	<b>ISSN 1650-1942</b>
<b>Kund/Customer</b>	<b>Försvarmakten</b>
<b>Projektnr/Project no</b>	<b>E53078</b>
<b>Godkänd av/Approved by</b>	<b>Magnus Jändel</b>

**FOI, Totalförsvarets  
Forskningsinstitut**

**Avdelningen för  
Informationssystem**

**Box 1165**

**581 11 Linköping**

**FOI, Swedish Defence  
Research Agency**

**Information Systems**

**Box 1165**

**SE-581 11 Linköping**

## Sammanfattning

Denna rapport sammanfattar det resultat som producerats inom projektet *Arkitekturbaserad ledningssystemutveckling* (2008-2010). Projektets syfte var att tillhandhålla Försvarmakten kunskap avseende arkitekturbaserad utveckling av ledningssystem som är praktiskt och vetenskapligt kvalitetssäkrat. Att utveckla ledningssystem med rätt funktionalitet och egenskaper är både tidskrävande och kostsamt och andelen lyckade utvecklingsprojekt är relativt låg. Under många år har förutsättningarna för att åstadkomma ”bra” system studerats, men ännu återstår många aspekter att beakta.

Projektet *Arkitekturbaserad ledningssystemutveckling* har bidragit med kunskap inom en rad områden avseende Försvarmaktens utveckling av ledningssystem. Exempelvis har projektet bidragit med kunskap avseende (1) vilket behov av stöd föreligger vid utveckling av ledningssystem inom Försvarmakten, (2) relevanta principer och utvecklingsmetoder samt (3) erfarenheter kring modellbaserad systemutveckling.

Projektet har haft nära kopplingar till flera pågående mer kunskapsstillämpade projekt inom Försvarmakten så som *Modellbaserad förmågeutveckling*. Detta för att de frågeställningar som studerades i projektet skulle ha hög relevans för Försvarmaktens utveckling av ledningssystem. Kunskapsöverföringen från projektet har skett genom att projektets medarbetare aktivt deltagit i dessa mer tillämpande projekt samt genom rapporter. Inom projektet har även en rad vetenskapliga publikationer producerat, i syfte att vetenskapligt kvalitetssäkra erhållna resultat. En viktig del i att tillämpa projektets resultat inom Försvarmakten kommer att leva vidare i transferprojektet *Metod för kravställning av FM ledningssystem*.

Nyckelord: Ledningssystemutveckling, arkitekturbaserad utveckling, slutrapport

## Summary

This report summarizes the result produced by the project *Architecture-based C<sup>2</sup> systems development* (2008-2010). The project's objective was to provide the Armed Forces with knowledge relating to architecture-based development of C<sup>2</sup> systems that is practically and scientifically quality assured. Developing C<sup>2</sup> systems with the right functionalities and features is both time consuming and costly, and the proportion of successful development projects is relatively low. For many years the conditions to achieve "good" systems have been studied, but there are still many aspects left to explore.

The project *Architecture-based C<sup>2</sup> systems development* has contributed with knowledge in several areas relating to the Armed Forces' development of C<sup>2</sup> systems. For example, the project has contributed with knowledge relating to (1) which needs of support exist for the development of C<sup>2</sup> systems within the Armed Forces, (2) relevant principles and development methods, and (3) experiences regarding model-based systems development.

The project has had close links with several more applied projects, such as the *Model-based capability development* project. These links were established to ensure that the issues studied in the project would have high relevance for the Armed Forces' development of C<sup>2</sup> systems. Knowledge transfer from the project has been achieved through the project's employees actively participating in these more applied projects and through reports. The project has also produced a number of scientific publications in order to assure the quality of its scientific results. An important part of the application of the project's results within the Armed Forces will live on in the transfer project *Method for specifying requirements of FM C<sup>2</sup> systems*.

Keywords: C2 systems development, architecture-based development, final report

# Innehållsförteckning

<b>1</b>	<b>Inledning</b>	<b>7</b>
1.1	Projektet .....	7
1.1.1	Frågeställningar.....	8
1.1.2	Nytta/effekter för Försvarsmakten.....	9
1.1.3	Kunskapsspridning.....	9
1.1.4	Rapporter och vetenskapliga bidrag .....	9
<b>2</b>	<b>Genomförande</b>	<b>11</b>
2.1	Behovsanalys .....	11
2.2	Identifiering av utvecklingsprinciper och -metoder.....	11
2.3	Fallstudier.....	12
2.4	Utveckling av kunskapsdatabasen.....	12
<b>3</b>	<b>Vad är arkitekturbaserad ledningssystemutveckling</b>	<b>13</b>
<b>4</b>	<b>Varför behövs arkitekturbaserad ledningssystemutveckling</b>	<b>16</b>
<b>5</b>	<b>Vad behövs för arkitekturbaserad ledningssystemutveckling</b>	<b>17</b>
5.1	Gemensam terminologi.....	17
5.2	Systemavgränsningar och kontextuell förståelse.....	18
5.3	Behovsanalys och kravhantering .....	18
5.4	Modellering.....	21
5.5	Kvalitetssäkring .....	23
5.5.1	Användarmedverkan .....	23
5.5.2	Kvalitets- och beslutsgrindar .....	24
5.6	Utnyttja forskning.....	24
5.6.1	Datorstödet Illuminatio .....	24
5.6.2	Transferprojekt .....	25
<b>6</b>	<b>Diskussion</b>	<b>26</b>
<b>7</b>	<b>Referenser</b>	<b>28</b>



# 1 Inledning

Att utveckla system är ett hantverk som kräver insikter baserade på ett flertal olika kompetenser. Det finns många fallgropar inom ett flertal områden och det krävs att det finns en ömsesidig respekt mellan utvecklare från olika kunskapsdiscipliner samt mellan utvecklare och användare (Hallberg, Pilemalm Westerdahl et al 2008; Hallberg, Hansson, Jungert et al 2009). Andelen lyckade projekt har varit och är fortfarande låg (Söderström, 2010; Kasser, 2007; Chaos, 1995). Ingen egen kunskapsdisciplin har i kraft av sig själv all den kunskap som krävs för att framgångsrikt genomföra utveckling.

Att utveckla system är ett utmanande och tidskrävande hantverk (Collin, 2003; Boehm and Papaccio, 1998). Utmaningarna med att utveckla system ligger i att erhålla förståelse för den framtida användningen, omsätta denna förståelse till behov samt att nyttja dessa för att bestämma vilka krav systemet ska uppfylla. Krav skall omsättas i en design, som skall realiseras till ett faktiskt system. Vidare skall utvecklade system svara mot såväl behov som krav (Hallberg, Pilemalm Westerdahl et al, 2008). Detta innebär ett omfattande behov av att kunna tolka, omsätta och kommunicera information mellan varje utvecklingssteg. Det har visat sig nödvändigt att iterera utvecklingsstegen för att åstadkomma system som motsvarar de förväntningar som ställs. Det vill säga, att göra ”rätt sak, första gången” har inte varit det normala inom systemutveckling. Sedan många år finns en strävan att effektivisera systemutveckling, vilket innebär att bygga rätt system på rätt sätt första gången (Arthur, 1992). Att hitta en universell metod har dock visat sig omöjligt (Brooks, 1995). Detta orsakas av att genomförandet av utvecklingsaktiviteter i hög grad påverkas av de specifika förutsättningarna som råder för varje projekt - så som vilket system som skall byggas, användarrepresentanters tillgänglighet samt utvecklarnas erfarenheter av att bygga liknande system.

Att utveckla rent tekniska system är svårt, men att utveckla socio-tekniska system (exempelvis ledningssystem) är betydligt svårare. Att avhandla samtliga dessa svårigheter och utmaningar i en rapport, bok eller ett projekt är inte möjligt. För att belysa denna problematik från olika håll har projektet innefattat såväl intervjustudier, som litteraturstudier och fallstudier. Samtliga fallstudier har utgått från en specifik frågeställning, relaterad till ledningssystemutveckling.

Syftet med denna rapport är att sammanfattande beskriva de resultat som framkommit under projektet *Arkitekturbaserad ledningssystemutveckling*.

## 1.1 Projektet

Projektet *Arkitekturbaserad ledningssystemutveckling* (2008-2010) syfte var att tillhandhålla Försvarmakten kunskap avseende arkitekturbaserad utveckling av



ledningssystem som är praktiskt och vetenskapligt kvalitetssäkrat. Detta sker genom att de erfarenheter som finns inom Försvarsmakten, hos leverantörer, stödjande organisationer och i vetenskapssamhället, struktureras och tillgängliggörs. En viktig del i detta har varit att projektets medarbetare aktivt medverkat i olika ledningssystemutvecklingsprojekt och andra relaterade initiativ inom Försvarsmakten.

### **1.1.1 Frågeställningar**

De generella frågeställningarna som legat till grund för projektet är: (1) vilka egenskaper och vilket innehåll skall ingå i arkitekturramverk, arkitekturer och utvecklingsmetoder för modellbaserad ledningssystemutveckling, (2) hur skall ovanstående nyttjas för att utveckla framtida ledningssystem, och (3) hur skall kunskapen avseende ovanstående förvaltas?

De specifika frågeställningarna för verksamheten 2008 var:

- Vilket behov av stöd föreligger vid utveckling av ledningssystem inom Försvarsmakten?
- Vilka relevanta principer och ansatser till arkitekturramverk, arkitekturer och utvecklingsmetoder finns och nyttjas idag?

De specifika frågeställningarna för verksamheten 2009 var:

- Vilken nationell och internationell erfarenhet avseende modellbaserad utveckling av ledningssystem (ex. MODAF) finns idag?
- Hur säkerställs i modellbaserad ledningssystemutveckling att de framtida användarnas behov tillgodoses?
- Hur säkerställs tillvaratagandet av de erfarenheter som gjorts under utveckling och i användning av befintliga ledningssystem, vid utveckling av nya ledningssystem?

De specifika frågeställningarna för verksamheten 2010 var:

- Vilka erfarenheter (goda respektive dåliga) finns avseende modellbaserad systemutveckling?
- Hur skall metoder, ansatser och principer för systemutveckling inom Försvarsmakten utformas?
- Hur skall ovanstående nyttjas för att utveckla ledningssystem?
- Hur har modeller etc. historiskt nyttjats vid utveckling inom Försvarsmakten och hur kan denna kunskap användas för att utforma modellbaserad förmågeutveckling?
- Hur skall kunskapen avseende ovanstående förvaltas?

### 1.1.2 Nytt/effekter för Försvarmakten

Under projektet har Försvarmakten erhållit stöd i ett antal projekt och uppdrag av projektets medarbetare, i form av expert- och direktstöd. Exempel på detta är LEDS CIO målsättningsarbete avseende Informationshantering, LEDS VHU stöd till utvecklingen av Försvarmaktens verksamhetsledning och stöd avseende modellbaserad förmågeutveckling inklusive H-MÅL ny. Projektet har även bidragit med att samla erfarenheter som görs inom ledningssystemutveckling, att analysera dessa samt att återföra detta till Försvarmakten. Vidare har projektet möjliggjort uppbyggnaden av kompetens avseende arkitektur och systemutveckling vid FOI, en resurs som redan avropas i ett flertal andra verksamheter.

### 1.1.3 Kunskapsspridning

Den kunskap som byggts upp inom projektet har överförts och kommer att överföras till Försvarmakten och FMV via:

- rapporter,
- projektmedarbetarnas medverkan i olika projekt och uppdrag,
- transferprojekt ”Metod för kravställning av FM ledningssystem”,
- de intervjuer som genomfördes där de intervjuade erhöll information om verksamheten, samt
- projektets referensgrupp bestående av personer från Försvarmakten och FMV (Hallberg, 2008).

### 1.1.4 Rapporter och vetenskapliga bidrag

Följande rapporter är producerade inom projektet:

- Hansson, J. Granlund, H., Hallberg, N., Pilemalm, S., and Annie Pilemalm (2010) *Kvalitetssäkring vid kravhantering: Granskning av formuleringar*. FOI-R--3070—SE.
- Pilemalm, S. (2010) *Direktstöd – kravhantering FMLS/SWECCIS*. FOI Memo 3388.
- Litsegård, P. & Sundmark, T. (2010) *Illuminatio: Informationssystem för att tillgängliggöra vetenskapligt kvalitetssäkrad kunskap avseende utveckling av ledningssystem*. FOI Memo 3358.
- Granlund, H. och Hansson, J. (2010) *CAiSE, Hammamet Tunisien, 7-11 juni 2010*, FOI MEMO 3303.

- Hallberg, N., Pilemalm, S., Sparf, M. och Sjödin, L. (2009) *Modellbaserad utveckling: Omvärldsanalys*, FOI Memo 2842.
- Hallberg, N., Hansson J., Jungert E., Westerdahl L., Pilemalm S., Granlund, H., Sundmark, T., Litsegård, P., Kylesten, B., Hunstad, A., Rankin, A. and Eriksson, H. (2009) *Ledningssystemsutveckling: Fallstudier kring kravhantering, modellering och kvalitetssäkring*, FOI-R--2892—SE.
- Sundmark T. och Hallberg N. (2009) *Prototyp av databas för metoder och principer avseende arkitekturbaserad utveckling av ledningssystem*, FOI Memo 2911.
- Hallberg, N., Pilemalm, S., Westerdahl, L., Jungert, E., Eriksson, H., Andersson, L., & Lindmark, F. (2008) *Principer och metoder för systemutveckling*, FOI-R--2628--SE.
- Hallberg, N., Pilemalm, S., & Westerdahl, L. (2008) *Behovsanalys avseende Försvarsmaktens utveckling av ledningssystem*, FOI Memo 2443.

Följande vetenskapliga bidrag är producerade inom projektet.

- Hallberg, N., Andersson, R., and Ölvander, C. (2010) Agile architecture framework for model driven development of C2 systems. *Systems Engineering* Vol. 13, No. 2, pp 175-185.
- Hallberg, N., Pilemalm, S., and Timpka, T. (2010) Quality Driven Requirements Engineering for Development of Crisis Management Systems. *BMC Medical Informatics and Decision Making*. Submitted
- Hallberg, N., Jungert, E., and Pilemalm, S. (2010) Ontology for Systems Development. *International Journal of Software Engineering and Knowledge Engineering*. Submitted
- Pilemalm, S., Hallberg, N., Sparf, M. and Niclason, T. (2010) Practical Experiences of Model-based Development - Case studies from the Swedish Armed Forces. *Systems Engineering Journal*. Submitted

## 2 Genomförande

De initiala stegen i projektet utgjordes av en behovsanalys vars syfte var att identifiera vilket behov av stöd som finns inom Försvarets utveckling av ledningssystem (Hallberg, Pilemalm, & Westerdahl, 2008). Behovsanalysen följdes av en litteraturstudie avseende utvecklingsprinciper och metoder (Hallberg, Pilemalm, Westerdahl, et al 2008). Litteraturstudien låg till grund för beskrivningar av olika utvecklingsmetoder och -principer. Baserat på behovsanalysen, litteraturstudien samt problem och frågeställningar som lyfts i andra verksamheter utformades frågeställningarna för nio fallstudier. Varje fallstudie hade en väl avgränsad frågeställning och ett specifikt syfte (Hallberg, Hansson, Jungert et al 2009; Hansson, Granlund, Hallberg & Pilemalm, 2010). Ett övergripande syfte för samtliga ingående fallstudier var dock att studera fenomenen och bidra till utveckling av ansatser och kunskap kring systemutveckling. Exempelvis syftade fallstudien *Erfarenheter av modellbaserad utveckling* (Appendix E: Fallstudier) till att studera hur modellbaserad utveckling fungerade inom ramen för Försvarets program PRIO. Vidare hade fallstudien *Erfarenheter av MODAF* till syfte att skapa ett frågeinstrument för att utvärdera tillämpningen av MODAF. Under projektets genomförande har arbete bedrivits, kontinuerligt och parallellt, med att skapa en kunskapsdatabas med avsikt att dokumentera och sprida kunskap om arkitekturbaserad utveckling av ledningssystem (Sundmark & Hallberg, 2009; Litsegård & Sundmark, 2010).

### 2.1 Behovsanalys

Behovsanalysen baserades på intervjuer med 16 respondenter från Försvaretsmakten, FMV och FOI. Dessa representerade skilda nivåer, områden och organisationer som är involverade i Försvaretsmakten ledningssystemutveckling. Syftet var att identifiera vilket behov av stöd och förbättringar som föreligger i Försvaretsmakten ledningssystemutveckling. Den information som erhöles under intervjuerna analyserades med avseende på att identifiera behov. Dessa behov kategoriserades och strukturerades. Genomförandet och det fullständiga resultatet av behovsanalysen beskrivs i Hallberg, Pilemalm och Westerdahl (2008) samt komprimerat i (Appendix A: Behov av stöd för utveckling av ledningssystem).

### 2.2 Identifiering av utvecklingsprinciper och -metoder

Med utgångspunkt i behov och erfarenheter avseende systemutveckling identifierades ett antal utvecklingsprinciper som är viktiga vid utveckling av system (Appendix B: Utvecklingsprinciper) samt begrepp centrala för systemutveckling (Appendix C: Begrepp inom systemutveckling), en generisk

utvecklingsprocess samt ett antal utvecklingsmetoder (Appendix D: Utvecklingsmetoder). Detta har skett genom litteratursökningar, dokumentstudier och en generell, erfarenhetsbaserad kunskapsinventering av de deltagande forskarnas olika kompetensområden. Dessa finns mer ingående beskrivna i Hallberg, Pilemalm, Westerdahl et al (2008).

## **2.3 Fallstudier**

Inom projektet genomfördes nio fallstudier för att mer i detalj studera fenomen relaterade till Försvarmaktens utveckling av ledningssystem (Hallberg, Hansson, Jungert et al, 2009; Hansson, Granlund, Hallberg & Pilemalm, 2010). Varje fallstudie har utgått från en specifik frågeställning och nyttjat en för denna anpassad metodik. Fallstudierna har i första hand genomförts inom sådan verksamhet som projektets medarbetare har varit involverade i, vilket har gett en naturlig återkoppling av genererad kunskap till verksamheten.

## **2.4 Utveckling av kunskapsdatabasen**

Inom projektet utvecklades en databas för den kunskap som genereras i projektet. I dess nuvarande form utgör den en fungerande prototyp som agerar som (1) ett medium för att sprida kunskap till projektets intressenter, (2) ett arbetsredskap för projektmedarbetare då dessa deltar i olika verksamheter, samt (3) ett organisatoriskt minne för forskargruppen. Under arbetet har flera olika tekniska möjligheter för att realisera databasen studerats. Successivt i detta arbete har fokus flyttas från ”inmatning” och ”lagring” av information till att handla om att ”hitta” och ”nyttja” information (Litsegård & Sundmark, 2010).

### 3 Vad är arkitekturbaserad ledningssystemutveckling

Det finns i litteraturen ingen entydig definition av *arkitekturbaserad ledningssystemutveckling*. Inte heller begreppen *arkitekturbaserad utveckling* eller *ledningssystem* är entydigt definierade. För att skapa en förståelse för begreppet arkitekturbaserad ledningssystemutveckling är det nödvändigt att klarlägga den fullständiga betydelsen av de ingående begreppen, så som *system*, *ledningssystem* och *systemutveckling* samt *modellbaserad utveckling* vilken är en närliggande företeelse.

#### Vad är ett system?

Ett system är en samling komponenter som har organiserats för att åstadkomma en eller flera specifika funktioner (ANSI/IEEE, 2007). System kan vara tekniska, organisatoriska eller kombinationer av dessa. System består av komponenter (som kan vara andra system) och existerar i en systemomgivning. Både komponenter och systemomgivning kan betraktas som system. Vad som är vad avgörs av vilket system som är i fokus.

#### Vad är ett ledningssystem?

Ett ledningssystem är en speciell typ av system som utgör alternativt utför ledningen av andra system. Generellt har ledningssystem tre uppgifter (1) samla in information, (2) beslutsfattande, och (3) delge information och order. Inom Försvarsmakten, definieras ledningssystem som bestående av människor och tekniska stöd, vilka är organiserade och som följer givna arbetsprocedurer. Inom Försvarsmakten finns i huvudsak två typer av ledningssystem, (a) insatsledningssystem och (b) verksamhetsledningssystem. I andra sammanhang ses ledningssystem som det tekniska system som stödjer beslutsfattare att genomföra ledning.

#### Vad är systemutveckling?

Systemutveckling är processen att utveckla system. Det existerar en stor mängd olika processer (Hallberg, Pilemalm, Westerdahl et al 2008). Det finns ett antal begrepp som beskriver utvecklingsprocessens genomförande. *Iterativ utveckling* kännetecknas av återkoppling från senare faser till tidigare för att göra omarbetningar, exempelvis ta fram en ny version av kravspecifikationen. *Inkrementell utveckling* innebär att operativa delar av systemet levereras efter hand och att varje ny version av systemet har en ökad funktionalitet. *Evolutionär utveckling* är både iterativ och inkrementell. *"Agile" utveckling* innebär att med korta iterationer successivt bygga upp systemets funktionalitet (Larman, 2004).

#### Vad är modellbaserad utveckling?

Modellbaserad utveckling omfattas av (1) utvecklingsprocesser, (2) metamodeler och (3) relationer och möjlighet till transformationer (Sottet et al,

2006; Romero et al, 2007; Hallberg, Pilemalm, Sparf et al 2009).

Utvecklingsprocessen syftar till att skapa och successivt förfina modeller av systemet, där ytterligare information läggs till i efterkommande steg och iterationer (Sottet et al, 2006). I modellbaserad utveckling utgör modeller indata till och resultat från samtliga steg i utvecklingen, till dess att det slutliga systemet erhållits (Balasubramanian et al, 2006).

### **Vad är arkitekturbaserad utveckling?**

Det finns flera varianter av arkitekturbaserad utveckling i litteraturen, även om begreppets användning är begränsad. Nedan beskrivs fem varianter.

#### *Variant 1: Modellbaserad utveckling*

Modellbaserad utveckling kan mycket väl ses som en variant av arkitekturbaserad utveckling, där abstrakta modeller successivt utvecklas till ett system. I vissa modellbaserade ansatser fokuserar de initiala modellerna främst på omgivningen till system.

#### *Variant 2: Nu-läge versus bör-läge*

Denna variant bygger på att utvecklingen baseras på skillnaden mellan hur systemet fungerar respektive hur det skall fungera. Detta "skall fungera" kan avse såväl nutid som framtid. Arbetsprocessen är:

1. Identifiera och beskriv det nuvarande systemet som en arkitektur
2. Identifiera och beskriv hur systemet skall fungera som arkitektur
3. Identifiera skillnaderna och hur dessa skall elimineras

Att skapa arkitekturbeskrivningar är resurskrävande och inte helt ovanligt förbises därför steg 1. Vilket försvårar att nyttja nuvarande system och delsystem. Exempel på en ansats som baserade enbart på "bör-läget" är "*Business Process Reengineering (BPR)*" (Hammer & Champy, 1993).

#### *Variant 3: Arkitekturdrivande krav -> Arkitektur -> System*

En variant av arkitekturbaserad utveckling som bygger på att:

1. Identifiera de arkitekturdrivande kraven
2. Skapa en arkitektur
3. Nyttja arkitektur för att designa system

#### *Variant 4: Arkitektur "by accident"*

Denna variant bygger på studier av snarlika system till det som skall utvecklas, i syftet att identifiera "goda" egenskaper (arkitektur). Dessa egenskaper dokumenteras och nyttjas sedan som grund för att utveckla systemet. Därtill hör dokumentation av "best practices" på lösningar avseende design.

#### *Variant 5: "Enterprise Architecture"*

Arkitekturbaserad utveckling avser utveckling som nyttjar sammanhängande beskrivningar av omgivningen och/eller systemet, det vill säga beskrivningar av dess/deras arkitektur. Detta görs utifrån ett helhetsperspektiv, avseende såväl dynamiska som statiska egenskaper. Arkitekturen för systemomgivningen nyttjas för att identifiera de behov som det utvecklade systemet skall tillfredsställa samt övriga krav som skall ställas på systemet. Exempel på detta är ”Enterprise Architecture” och dess användning för att styra organisationers IT-stöd. Syftet med arkitekturbeskrivningen av systemet som skall utvecklas är att ge en översikt av systemdesignen. En sådan översiktlig helhetssyn uppnås genom den typ av beskrivningar som inte detaljerat återger enskilda komponenter och funktioner, utan istället anger hur de olika komponenterna skall integreras i systemet.



## **4 Varför behövs arkitekturbaserad ledningssystemutveckling**

Traditionell utveckling av system likväl ledningssystem har baserats på olika dokument, till exempel kravspecifikation och designspecifikation, som produceras i olika aktiviteter under projektens gång. Dessa har sedan legat till grund för det arbete som sker i efterkommande aktiviteter. Problemet med denna ansats är att mycket information går förlorad mellan olika aktiviteter och att spårbarhet är svår att upprätthålla. Att nyttja modeller för att överföra information till efterkommande aktiviteter har flera fördelar, men svårigheten är bland annat att hitta rätt abstraktionsnivå på dessa modeller. Vidare krävs relativt många olika modeller för att möta behovet under utvecklingsarbetet.

Tanken i arkitekturbaserad utveckling är att det är samma modell som successivt förädlas och vidareutvecklas under hela utvecklingsprocessen. För att hantera komplexitet så nyttjas datorbaserade verktyg och möjligheten att betrakta modellen ur olika vyer. Denna ansats har sina utmaningar och problem, men även fördelar i form av spårbarhet och möjligheten att kontinuerligt uppdatera modellen. Det är exempelvis möjligt att ändra tidig information, vilket så skapar ett senare genomslag. I behovsanalysen som genomfördes i projektet lyftes det fram behov av exempelvis spårbarhet, en gemensam dokumentationsform samt återanvändning av resultat och erfarenheter. Samliga av dessa behov skulle kunna tillgodoses med hjälp utav arkitekturbaserad ledningssystemutveckling.

## 5 Vad behövs för arkitekturbaserad ledningssystemutveckling

Under behovsanalysen identifierades en mängd behov som inte har en direkt betydelse för arkitekturbaserad ledningssystemutveckling, så som mer resurser och tillgång till rätt kompetens. Flera andra behov, så som väldefinierad utvecklingsprocess och tydligt ansvar, ligger utanför projektets fokus och hanteras i andra verksamheter. Projektet nöjer sig därmed med att påtala existensen av dessa behov (Hallberg, Pilemalm & Westerdahl, 2008).

### 5.1 Gemensam terminologi

I utveckling av ledningssystem krävs samverkan mellan organisationer, grupper och individer med skilda kompetenser och bakgrunder. Effektiv samverkan utan missförstånd förutsätter att de som interagerar delar ett gemensamt språk. Att inte ha en gemensam begreppsapparat kan leda till missförstånd som kan få allvarliga konsekvenser för utvecklingen. I projektet har ett arbete bedrivits med att identifiera, strukturera och relatera viktiga termer och begrepp inom systemutveckling (Hallberg, Pilemalm, Westerdahl et al 2008; Hallberg, Jungert & Pilemalm, 2010). Resultatet utgörs av en ontologi som innehåller fyra övergripande kategorier av begrepp; (1) Generella begrepp, (2) Beskrivningsbegrepp, (3) Utvecklingsprocessbegrepp och (4) Systemuppträdandebegrepp.

- De generella begreppen innefattar: System, Ledningssystem, Systemomgivning, Komponent, Vy, Arkitektur, Arkitekturramverk, Ramverk och Systemutveckling.
- Beskrivningsbegreppen innefattar begrepp som bidrar till att beskriva systemet i dess olika utvecklingsfaser, så som Utsaga, Behov, Krav, Design och Realisering.
- Utvecklingsprocessbegreppen innefattar aktiviteter som genomförs under systemutveckling. Dessa aktiviteter är Kontextanalys, Behovsanalys, Kravhantering, Design, Realisering, Införande, Verifiering, Validering och Exploatering.
- Systemuppträdandebegrepp innefattar begrepp som beskriver hur system ser ut, fungerar och uppträder. Dessa begrepp innefattar Funktion, Kapacitet, Förmåga, Egenskap, Tjänst och Användbarhet.

## 5.2 Systemavgränsningar och kontextuell förståelse

Att avgöra systems gränser är en av de allra första aktiviteterna som måste genomföras (Sommerville, 2001). När väl dessa gränser är identifierade och fastställda kan själva utvecklingsarbetet påbörjas. Ett av de första steg är att få en förståelse för den kontext som system skall användas i. Detta steg innefattar att erhålla förståelse för vilken miljö samt vilka intressenter, system, styrande dokument etc som påverkar och påverkas av systemet så väl vid utveckling som i drift. I detta ingår vidare att besluta vilka av dessa som skall tas hänsyn till vid utformningen av system. Till exempel vilka intressenters behov skall beaktas vid utformningen av system.

Projektet har identifierat behovet av tydligt fastställda systemgränser, så väl i behovsanalysen som i till exempel fallstudien *Kvalitetsdriven kravhantering för utveckling av avvikelsehantering* (Hallberg, Pilemalm & Westerdahl, 2008; Hallberg, Hansson, Jungert et al, 2009). Dock har inga specifika metoder för detta identifierats. En generell möjlighet är att nyttja en modell för att kommunicera vad som är kontext och vad som är det system som skall utvecklas. Vidare har behovet av kontextanalys identifierats, vars syfte är att analysera den omgivning i vilket det system som skall utvecklas avses att fungera. Flera av de metoder som beskrivs i Hallberg, Pilmalm, Westerdahl et al (2008) kan nyttjas för att genomföra kontextanalys i form av verksamhetsanalys och intressentanalyser, vilka är vanliga former av kontextanalys. *Verksamhetsanalys* fastställer hur verksamheten bedrivs, exempelvis avseende organisation, processer och informationsflöden. *Intressentanalys* fastställer vilka intressenterna till ett system är. Det är vidare deras upplevelser av problem, brister och möjligheter som motiverar utvecklingen.

## 5.3 Behovsanalys och kravhantering

Det är för alla former av systemutveckling betydelsefullt och centralt att basera utvecklingen på de behov som intressenter och verksamheter har och i de allra flesta fall att uttrycka dessa som krav på systemet. Att genomföra behovsanalys innebär att identifiera och dokumentera intressenters och verksamheters behov. Det innefattar såväl uttalade som outtalade behov (Hallberg, Andersson, & Westerdahl, 2005; Ficalora & Cohen, 2009; Kano, 1995). Kravhantering syftar till att fastställa krav på system, där varje krav skall vara korrekt formulerat, entydigt, förståeligt, komplett, konsistent, verifierbart, modifierbart, spårbart och prioriterat samt att kravmängden är detaljerad till en nivå lämplig för den fortsatta utvecklingen (Wieggers, 2003).

Projektet har identifierat behovet av adekvata behovsanalyser och kravhantering vid flera delstudier, exempelvis i behovsanalysen avseende stöd för utveckling av ledningssystem (Hallberg, Pilemalm & Westerdahl, 2008), litteraturstudien avseende modellbaserad utveckling (Hallberg, Pilemalm, Sparf et al 2009) och fallstudien *Erfarenheter av modellbaserad utveckling* (Hallberg, Hansson, Jungert et al 2009). Ett viktigt moment inom behovsanalyser och kravhantering är att prioritera, vilket är nödvändigt då det inte är möjligt eller ens önskvärt att bygga systemlösningar för alla behov (Karlsson, Wohlin, & Regnell, 1998).

*Prioritering* görs för att fokusera på det som ger störst nytta för intressenterna, baserat på tillgängliga resurser. I samtliga utvecklingsprojekt görs prioriteringar explicit eller implicit, exempelvis avseende vilka intressenter som skall ges möjlighet att påverka utformningen av systemet och även i vilken omfattning de skall göra det. Prioritering görs också ett flertal gånger under utvecklingen, exempelvis avseende intressenter, behov, krav och designval (Hallberg, 1999).

Projektet har beskrivit ett flertal metoder och ansatser för att genomföra behovsanalyser och kravhantering (Hallberg, Pilemalm, Westerdahl et al, 2008; Hallberg, Hansson, Jungert et al, 2009). Exempel på metoder för att fånga intressenternas upplevelser är intervjuer, deltagande observationer och dokumentanalyser, The Critical Incident Technique (CIT) och Future workshops.

- Intervjuer genomförs för att samla information om och förstå individers erfarenheter och upplevelser. De är lämpliga när utvecklaren vill fånga enskilda aktörers perspektiv (Bernard, 2000).
- Deltagande observation av individers agerande är en metod som härstammar från antropologi och etnografi, men som har visat sig vara effektiv då det gäller förändringsarbete och teknikstött förändring av arbetsuppgifter (Hasu & Engeström, 2000).
- Dokumentanalyser används typiskt i den initiala utvecklingen, exempelvis för att studera verksamhetsstrukturer, verksamheter, visioner, mål, aktörer och aktiviteter samt för att identifiera utsagor som grund för behovsanalyser.
- The Critical Incident Technique (CIT) bygger på individers självrapportering av moment som de upplever som kritiska, för att uppnå verksamhetens mål (Flanagan, 1954). Här ryms moment som stödjer såväl som motverkar uppfyllandet av verksamhets mål. Ölvingson, Hallberg, Timpka, och Greenes (2002) samt Hallberg, Timpka och Eriksson (1999) beskriver användandet av CIT inom kravhantering.
- Future workshops involverar användare genom att låta dem reflektera över sin situation och de problem som återfinns i relation till den, förändringsbehov i verksamheten samt behov av informationstekniskt

stöd för förändringar (Kensing & Halskov Madsen, 1993). Future Workshops genomförs i tre faser: kritikfas, fantasifas och implementationsfas; där medverkande först beskriver den egna arbetssituationen och dess problem, sedan genererar futuristiska lösningar på problemen och i nästa steg omvandlar dessa till tekniskt och organisatoriskt implementerbara lösningar (Jungk, & Müllert, 1987).

Projektet har också beskrivit en metod *kundrösttabell* för att identifiera behov ur utsagor.

- Kundrösttabell (eng: Voices of Customers Table, VCT) är ett av många kvalitetsverktyg som utvecklats inom Quality Function Deployment (QFD) (Mazur, 1992). I kundrösttabellen antecknas kundens önskningar i form av utsagor. Dessa utsagor analyseras därefter utifrån (1) vem som anses behöva dem, (2) vad de anser sig behöva, (3) när de behöver det, (4) var det behöver användas, (5) varför behovet finns och (6) hur de vill att det skall fungera. Analys med stöd av kundrösttabell har som mål att formulera faktiskt behov. Kundrösttabellens primära funktion är att underlätta analysen av vad kunden (i systemutveckling motsvaras *kund av användare* eller *intressent*) egentligen behöver.

Projektet har identifierat och beskrivit en metod *användningsfall* för att omsätta behov till krav.

- Användningsfall beskriver hur system används eller är tänkta att användas. Det består av en text som beskriver själva användningen, samt en figur som illustrerar aktörers interaktion med systemet (Kulak & Guiney, 2003). Hallberg, Ölvander och Törne (2006) beskriver en mall för användningsfall. I Hallberg, Pilemalm och Timpka (2010) beskrivs hur användningsfall nyttjas för att omsätta behov till krav.

Projektet har också beskrivit en ansats *Kvalitetsdriven kravhantering* för att, baserat på ovanstående metoder, genomföra behovsanalyser. Ansatsen har också utvärderats i två fallstudier *Kvalitetsdriven kravhantering för utveckling av insatsledningssystem* och *Kvalitetsdriven kravhantering för utveckling av avvikelshantering*. Vidare studerades i fallstudien *Kravhantering och MODAF* hur denna ansats kan nyttjas som underlag för en MODAF-baserad modellering.

- Kvalitetsdriven kravhantering är en ansats för att identifiera krav genom: (1) Insamling av data, (2) Identifiering av utsagor, (3) Identifiering av behov, (4) Analys av behov, (5) Identifiering av krav och (6) Analys av krav (Hallberg, Pilemalm, & Timpka, 2010; Hallberg, Andersson & Westerdahl, 2005).

Att identifiera behov och krav på ett tillfredsställande sätt är nödvändigt, dock inte tillräckligt. Behov så väl som krav måste kommuniceras till den fortsatta utvecklingen, vilket innebär att de måste dokumenteras på ett adekvat sätt. I

projektet studerades och utvecklades en ansats för att formulera, granska och kvalitetssäkra behov och krav (Hansson, Granlund, Hallberg et al 2010). Ansatsen kombinerar ett flertal etablerade förfaringssätt till en sammanhållen metod i fyra steg; (1) *Formulera behov*, (2) *Granska behov*, (3) *Formulera krav* och (4) *Granska krav*. Syftet med ansatsen är att överbrygga ett vanligt problem vid dokumentation av krav, att besluta vilken typ av information om kravet som skall beskrivas samt hur (Karlsson, Wohlin, & Regnell, 1998). Ansatsens perspektiv är på formulering som en systemoberoende svårighet vid dokumentering, inte bara av krav utan även av behov. Exempel på förfaringssätt som nyttjas i de fyra stegen är behovs- och kravtabeller, checklistor och inspektioner.

- Behovs- och kravtabeller utgör en strukturerad dokumentation av behov respektive krav för att skapa förutsättningar för vidare systemutveckling. Tabellen innehåller bland annat unika identifierare, spårbarhet mellan behov och krav samt formuleringen av behov eller krav.
- Checklistor används vid granskning av behov och krav. Ett antal kriterier har identifierats, bland annat ifrån Davies et al (1993) arbete med kvalitetsattribut för krav. Behov och krav granskas emot dessa uppsatta kriterier och checklistan tillåter att de granskade objekten markeras om de ej uppfyller ett av kriterierna.
- Inspektioner utvecklades av Fagan (1986) och utgör ett systematiskt sätt för en grupp, bestående av ett fåtal personer, att systematiskt granska och korrigera komplexa dokument från ett förutbestämt perspektiv. I projektets ansats har inspektionen utvecklats för att passa perspektivet på granskning av formulering.

## 5.4 Modellering

Att nyttja olika former av modeller för att beskriva företeelser och system ur olika perspektiv och att nyttja dessa modeller är fundamentalt i all form av arkitekturbaserad utveckling. Att använda modeller inom systemutveckling har länge varit vanligt, exempelvis för att beskriva, begripliggöra och dokumentera komplexa samband och system i olika utvecklingsfaser. Modeller nyttjas även för att beskriva och dokumentera systemomgivningar, så väl existerande som tänkta (Hallberg & Fransson, 2001; Hay, 2003; Sommerville, 2001; Friedenthal, Moore & Steiner, 2008). Inte heller modellbaserad utveckling är något nytt fenomen (Hallberg, Pilemalm, Sparf et al 2009).

Modellbaserad utveckling ses som en möjlig väg att komma tillrätta med en del av de svårigheter som finns i systemutveckling. Exempelvis syftar den till att system bättre skall motsvara existerande behov samt att utvecklingen och även senare modifieringar skall gå fortare att genomföra. Modellbaserad utveckling

utgår vanligtvis 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. Rent akademiskt finns det dock fortfarande många kunskapsluckor då det gäller hur modellbaserad utveckling skall genomföras.

Projektet identifierade ett flertal behov som relaterar till modellering, så som exempelvis behov av

- att modellera de förmågor som karakteriserar verksamheter och ledningssystem.
- gemensamma dokumentationsformat för resultat, erfarenheter och kunskap så att dessa kan återanvändas.
- spårbarhet för att följa upp förändringar och motivera nyttan med genomfört arbete (Hull, Jackson & Dick, 2005).

Hallberg, Pilemalm, Sparf et al (2009) beskriver en litteraturstudie kring modellbaserad utveckling. Projektet beskriver vidare ett antal metoder etc. för modellering och modellbaserad utveckling.

- Unified Modeling Language (UML) är ett standardiserat modelleringsspråk för visualisering, specificering, konstruktion och dokumentation (OMG, 2001). Språket inkluderar en rad notationer och diagram för att beskriva såväl dynamiska som statiska egenskaper hos system. Dock är språket mer uppskattat bland tekniker än av exempelvis ekonomer och verksamhetsledare.
- Användningsfall är en modell för hur system används eller är tänkta att användas (Kulak & Guiney, 2003).
- Misuse cases är en utveckling av traditionella användningsfall, som också beskriver hur systemet inte får nyttjas (Sindre & Opdahl, 2000). Genom att visa på möjliga hot i samma modell som de funktionella kraven ges möjligheten att komplettera modellen med funktionella krav som kan reducera konsekvenserna av hoten.
- Prototyper är modeller av system. Dessa kan användas i kommunikationen med användare för att ge dessa möjligheter att påverka utformningen av systemet. De kan också nyttjas som ett medium att kommunicera mellan de som designar respektive realiserar systemet.
- Arkitekturramverk är en metod för utveckling av systemarkitekturer för att erhålla sammanhängande och konsekventa modeller. Det finns många arkitekturramverk, exempelvis MODAF, DoDAF och TOGAF. Försvarsmakten har fattat ett beslut om att använda MODAF.

Flera av de fallstudier som genomförts inom projektet har direkt anknytning till modellbaserad utveckling.

- Fallstudien *Modellbaserad utveckling* beskriver, baserad på en litteraturstudie, vad modellbaserad utveckling är, vad det syftar till och hur det bedrivs.
- Fallstudien *Erfarenheter av MODAF:s* 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.
- Fallstudien *Erfarenheter av modellbaserad utveckling* beskriver erfarenheterna av modellbaserad utveckling i form av fördelar, nackdelar och utmaningar med utgångspunkt från Försvarmaktens Program PRIO.

## 5.5 Kvalitetssäkring

Betydelsen av att kontinuerligt kvalitetssäkra delresultat i utvecklingsarbete kan inte överskattas (Collin, 2003). Till exempel, små brister i en kravspecifikation får stora konsekvenser om dessa upptäcks så sent som i det färdiga systemet. Framförallt är det kostsamt att åtgärda brister som ”hängt med ett tag” (Boehm & Papaccio, 1998). Fel och brister måste därmed identifieras och åtgärdas så fort som möjligt. Det finns två typer av kvalitetssäkring: verifiering och validering.

- Verifiering syftar till att avgöra om en design alternativt ett system motsvarar de fastställda kraven.
- Validering syftar till att avgöra om kraven, designen eller systemet motsvarar det faktiska behovet, det vill säga om dessa ger önskad effekt för intressenterna.

I projektet har främst två metoder för kvalitetssäkring beaktats, användarmedverkan samt kvalitets- och beslutsgrindar.

### 5.5.1 Användarmedverkan

Tillgång till användarrepresentanter med insikt i användningssituationen är centralt för att säkerställa en hög kvalitet hos utvecklade system. Medverkan av användare motiveras oftast av att det är de som känner sin domän bäst och som vet vilket behov av stöd de har. Det finns många olika sätt att involvera användarna, till olika grad samt att ge dem olika mandat för att påverka. Användarna kan till exempel involveras för att stödja analysen av utsagor för att identifiera behov och de kan utvärdera prototyper för att säkerställa att exempelvis inga ”oanvändbara” funktioner implementeras (Hallberg, Ölvander, Johansson, & Törne, 2006).



Projektet beskriver flera metoder som kan nyttjas i samband med användarmedverkan, så som *Future workshop*, *kundrösttabell* och *prototyping* (Hallberg, Pilemalm, Westerdahl et al, 2008).

### **5.5.2 Kvalitets- och beslutsgrindar**

Det arbete som berör kvalitets- och beslutsgrindar beskrivs främst i fallstudien *Kvalitets- och beslutsgrindar* (Hallberg, Hansson, Jungert et al, 2009). Hansson, Granlund, Hallberg et al (2010) bygger vidare på detta arbete och beskriver hur kvalitets- och beslutsgrindar kan utformas för behovs- och kravspecifikationer.

*Fallstudien Kvalitets- och beslutsgrindar* baseras på en omfattande litteraturstudie som beskriver hur kvalitets- och beslutsgrindar utformas och används. Det finns många positiva erfarenheter från användandet av kvalitets- och beslutsgrindar, men det finns även fallgropar. Ett av de största problemen är grindar som saknar ”tänder”. Dessa kan släppa igenom projekt som inte håller tillräckligt hög kvalitet. Utvecklade och införandet av kvalitets- och beslutsgrindar måste anpassa till den verksamhet som de skall stödja.

*Fallstudien Kvalitetssäkring vid kravhantering* beskriver en metod för att formulera, granska och kvalitetssäkra behov och krav (Hansson, Granlund, Hallberg et al 2010). Till metoden har material i form av tabeller, checklistor och handledningar tagits fram för att stödja utförandet i respektive moment. Delar av detta innefattar beskrivningar av hur granskning och inspektion av behov respektive krav skall ske, vilket kan nyttjas som grund för att utforma kvalitets- och beslutsgrindar avseende behovs- och kravspecifikationer.

## **5.6 Utnyttja forskning**

I den behovsanalys, som genomfördes i projektet, framkom det behov av att bättre kunna tillgodogöra sig forskning i utvecklingsarbete. Det innefattade behov att bättre utnyttja forskningsresultat för att erhålla förbättringar av utvecklingsarbetet, så väl med avseende på utvecklingsmetodik som på innehåll.

### **5.6.1 Datorstödet Illuminatio**

Illuminatio är ett verktyg för att öka tillgängligheten till vetenskapligt kvalitetssäkrade metoder och principer för utveckling av ledningssystem. Verktyget bygger på indexering av dokument och en kompetent sökmotor (Apache Solr) för fritextsökning. Fritextsökning gör det möjligt att hitta termer oavsett var de omnämns i dokumentet. Vidare är det möjligt att genomföra sökningar på närbesläktade begrepp som t.ex. synonymer och akronymer. Begrepp söks dessutom automatiskt på både svenska och engelska, så att till

exempel en sökning på ”kravhantering” också ger träffar på dokument som innehåller begreppet ”Requirements engineering”.

Genom nyttjandet av avancerad sökteknologi, istället för en relationsdatabas, har fokus därigenom flyttats från *hur* beskrivningen av identifierade metoder och principer lagras till att istället *hitta* och *nyttja* information. Detta medför även att de som producerar material till verktyget kan koncentrera sig på innehållet i, istället för på formen vilken informationen beskrivs. De ansatser som Illuminatio bygger på skapar dessutom stora möjligheter att integrera systemet med andra system, exempelvis Kupal (Waern, Lindahl, Bengtsson, & Westerdahl, 2010), för att öka tillgängligheten.

### 5.6.2 Transferprojekt

Transferprojektet *Metod för kravställning av FM ledningssystem* är ett tillämpningsprojekt baserat på kunskap som erhållits i FoT-projektet Arkitekturbaserad ledningssystemutveckling. I transferprojektet ska systemutvecklare från Totalförsvarets Forskningsinstitut (FOI) tillsammans med representanter från arbetsgruppen från FMLS TS och ledningsstödssystemet SWECCIS ta fram, tillämpa, utvärdera och dokumentera en metod för kravställning av ledningssystem, anpassad för FMs behov. FOI kommer även att direkt stödja arbetet med kravställningen av SWECCIS. Bakgrunden till projektet är dels svårigheter att generellt kravställa system och dels att FMLS TS/SWECCIS upplevt specifika problem i sitt utvecklings- och kravarbete; problem som är direkt kopplande till otillräcklig struktur och spårbarhet i utvecklingsprocessen. Projektet söker lösa dessa problem genom att formalisera tidigare erfarenheter och kunskaper, både generellt systemutvecklingsmässiga och försvarsmaktsspecifika, till en ensad, operativt fungerande metod för kravställning av ledningssystem.

## 6 Diskussion

Att utveckla system är komplicerat och ett projekt på 3 år kan inte lösa de frågeställningar som i princip härrör från 1960-talet. Frågeställningen är ”hur skall ändamålsenliga system åstadkommas, på en förutsägbar tid och till en förutsägbar kostnad” (Brooks, 1995). Andelen ”lyckade” utvecklingsprojekt har i princip inte ökat sedan 1994 (Chaos 1994; Kasser 2007; Söderström, 2010).

Projektet Arkitekturbaserad ledningssystemutveckling har haft som övergripande mål att stödja Försvarmakten och relaterade organisationer med vetenskapligt kvalitetssäkrat som dessa kan nyttiggöra vid utveckling av ledningssystem. För att få en uppfattning om vilket behov av stöd som finns genomfördes initialt en behovsanalys med personer som är involverade i utvecklingen av Försvarmaktens ledningssystem. Denna behovsanalys kom att fungera som en grund för projektet, men även som underlag för framtida projekt och kunskapsutvecklingsinitiativ.

Projektets huvudbidrag innefattar kunskap avseende:

- det behov av stöd som finns inom Försvarmaktens utveckling av ledningssystem.
- metoder och principer för systemutveckling.
- modellbaserad utveckling.
- att tillämpa Kvalitetsdriven kravhantering för utveckling av ledningssystem.
- att baserat på kravhantering skapa MODAF-baserade modeller.
- praktiska erfarenheter av MODAF.
- utveckling av ett intervjuinstrument för att utvärdera erfarenheter av MODAF.
- praktiska erfarenheter av modellbaserad utveckling.
- hur kvalitets- och beslutsgrindar har tillämpats.
- metodik för att skriva bättre behov och krav.
- utveckling av en metod för att kvalitetssäkra behovs- och kravformuleringar.
- hur datorstöd för att tillgängliggöra vetenskapligt kvalitetssäkrat kunskap om systemutveckling skall utformas.

Det finns många frågeställningar och kunskapsluckor som inte kunnat beaktas i projektet på grund av en begränsad budget. Projektet har dock bidragit till att

lägga en kunskapsgrund för framtida projektet. Framtida forskning inom området bör i än högre grad få en aktionsforskningsprägel. Detta betyder att forskningen kopplas än närmre den verksamhet som kunskapsutvecklingen är avsedd att stödja. Vidare behöver medarbetarna få utrymme för att vetenskapligt förkovra sig, vilket kan öppna nya möjligheter att öka kvalitén på utvecklade system. Ett exempel på detta är att innan 2008, då projektet startade, hade forskargruppen *Arkitektur och systemutveckling* enbart en mindre andel forskning relativt alla de avtappande projekt som gruppens medarbetare var involverade i. Detta höll på att dränera gruppen på så väl kompetens som medarbetare. Detta projekt har sedan dess bidragit till att ny kunskap har byggts upp, vilken relativt omgående omsatts i pågående tillämpade projekt. Vidare har projektet resulterat i ett transferprojekt, som kommer att tillämpa den kompetens som byggs upp kring kravhantering. Projektet har därmed vitaliserat gruppen och efterfrågan av gruppens kompetens har kraftigt ökat, vilket medfört att gruppen är dubbelt så stor som tidigare.

## 7 Referenser

Bernard, R. H. (2000) *Social Research Methods. Qualitative and Quantitative Approaches*, Sage Publications. Thousand Oaks, CA.

ANSI/IEEE Std 1471 (2007), *Recommended Practice for Architectural Description of Software-intensive Systems*. <http://www.iso-architecture.org/ieee-1471/index.html>

Arthur, J. L. (1992). *Improving Software Quality: An Insider's Guide to TQM*. New York: Wiley.

Balasubramanian, K., Gokhale, A., Karsai, G., Sztipanovits, J., & Neema, S. (2006) Developing Applications Using Model-driven Design Environments. *Computer*, 39, 2, 33-40.

Bergman, B. & Klefsjö, B. (1994) *Quality from Customer Needs to Customer Satisfaction*. London: McGraw-Hill.

Bernard, R. H. (2000) *Social Research Methods. Qualitative and Quantitative Approaches*, Sage Publications. Thousand Oaks, CA.

Boehm, B. W. & Papaccio, P. N. (1998) Understanding and controlling software costs. *IEEE Transaction on Software Engineering*, 14, 1462 – 1477.

Brooks, F. P. Jr. (1995). *The Mythical Man-Month: Essays on Software Engineering*. Addison-Wesley.

Carmel, E., George, J. F., & Nunamaker, J. Jr. F. (1992) Supporting Joint Application Development (JAD) and Electronic Meeting Systems: Moving the CASE Concept Into New Areas of Software Development. In J. F. Nunamaker (Ed.), *Proceedings of the Twenty-Fifth Hawaii International Conference on System Sciences*, 331-342.

Carroll, J. M. (2000) *Making Use. Scenario-based Design of Human-Computer Interactions*, Massachusetts Institute of Technology, MA.

Chaos, The Standish Group, 1995, <http://www.standishgroup.com/> (2004-02-27).

Collin B. (2003) *IT-kvalitet: Verksamhets- & Effektivitetsutveckling*. Studentlitteratur, Lund Sweden.

Davies, A., Overmyer, S., Jordan, K., Caruso, J., Dandashi, F., Dinh, A., Kincaid, G., Ledebøer, G., Reynolds, P., Sitaram, P., Ta, A., & Theofanos, M. (1993) Identifying and Measuring Quality in a Software Requirements Specification In *Proceedings of the 1 st International Symposium on Software Metrics*, 141–152.

Fagan, M. E. (1986) Advances in Software Inspections. *IEEE Transactions on Software Engineering*, SE-12(7): 744–751.

- Ficalora, J. P. & Cohen, L. (2009) *Quality Function Deployment and Six Sigma: A QFD Handbok*. Boston, Pearson Education.
- Flanagan, J. C. (1954). The Critical Incident Technique. *Psychological Bulletin*, 51, 327-58.
- Friedenthal, S., Moore, A., & Steiner, R. (2008) *A Practical Guide to SysML: The Systems Modeling Language*. Morgan Kaufmann. Amsterdam.
- Hallberg N., Hansson J., Jungert E., Westerdahl L., Pilemalm S., Granlund H., Sundmark T., Litsegård P., Kylesten B., Hunstad A., Rankin A. & Eriksson H. (2009) *Ledningssystemsutveckling: Fallstudier kring kravhantering, modellering och kvalitetssäkring*, FOI-R--2892—SE.
- Hallberg N. (1999) Incorporating User Values in the Design of Information Systems and Services in the Public Sector: A Methods Approach. [Dissertation No. 596] Linköping Studies in Science and Technology.
- Hallberg, N. (2008) Referensgrupp för FoT-projektet: *Arkitekturbaserad ledningssystemutveckling*. FOI Memo 2373.
- Hallberg, N. & Fransson, J. (2001) *UML baserad metamodell för modellering av bekämpningskedjor*. Linköping, FOI 2001, (FOI-R--0159--SE).
- Hallberg, N. & Hallberg, J. (2006) The Usage-Centric Security Requirements Engineering (USEr) Method. In *Proceedings of Information Assurance Workshop*, IEEE.
- 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., Jungert, E., & Pilemalm, S. (2010) Ontology for Systems Development. *International Journal of Software Engineering and Knowledge Engineering*. Submitted.
- Hallberg, N., Pilemalm, S., & Timpka, T. (2010) Quality Driven Requirements Engineering for Development of Crisis Management Systems. *BMC Medical Informatics and Decision Making*. Submitted.
- Hallberg, N., Pilemalm, S., & Westerdahl, L. (2008) *Behovsanalys avseende Försvarsmaktens utveckling av ledningssystem*. FOI 2008, FOI Memo 2443.
- Hallberg, N., Pilemalm, S., Sparf, M., & Sjödin, L. (2009) *Modellbaserad utveckling: Omvärldsanalys*, FOI Memo 2842.
- 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., 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).
- Hallberg, N., Ölvander, C., Johansson, M., & Törne, A. (2006) *Utvärdering av användningsfall avseende tekniskt ledningssystem för urban miljö*. Linköping, FOI 2006, (FOI Memo 1914).
- Hammer, M. & Champy, J. (1993). *Reengineering the Corporation*. New York, NY: Harper Collins Publisher.
- Hansson, J. Granlund, H., Hallberg, N., Pilemalm, S., and Annie Pilemalm (2010) *Kvalitetssäkring vid kravhantering: Granskning av formuleringar* FOI-R--3070—SE.
- Hasu, M. & Engeström, Y. (2000) Measurement in Action: an Activity-theoretical Perspective on Producer-user Interaction. *International Journal of Human-Computer Studies*, 53, 61-89.
- 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.
- Ingmarsson, M., Eriksson, H., and Hallberg, N. (2009) Towards Integration of Different Media in a Service-oriented Architecture for Crisis Management. *Proceedings of the 15th International Conference on Distributed Multimedia Systems (DMS'2009)*, Boston, MA, September 10-12. pp 146-151.
- Jungk, R. & Müllert, N. (1987) *Future workshops: How to Create Desirable Futures*. London, England, Institute for Social Inventions.
- Jürjens, J. (2005) *Secure Systems Development with UML*. Springer-Verlag Berlin Heidelberg 2005.
- Kano, N. (1995). Upsizing the Organization by Attractive Quality Creation. In G.K. Kanji (Ed.), *Proceedings of the First World Congress on Total Quality Management* (pp. 60-72). London: Champman & Hall.
- Karlsson, J., Wohlin, C., & Regnell, B. (1998). An Evaluation of Methods for Prioritizing Software Requirements. *Information and Software Technology*, 39, 939-947.
- Kasser, J.E. (2007) *A Framework for Understanding of Systems Engineering*. Cranfield: The right Requirement.

Kensing, F. & Halskov Madsen, K. (1991). Generating Visions: Future Workshop and Metaphorical Design. In J. Greenbaum & M. Kyng (Eds.). *Design at Work: Cooperative Design of Computer Systems* (pp. 155-168). Hillsdale, New Jersey: Lawrence Earlbaum.

Kruchten, P. H. (2004) *The Rational Unified Process. An Introduction*, Addison-Wesley.

Kulak, D. & Guiney, E. (2003) *Use cases: Requirements in context*. Addison-Wesley, Upper Saddle River, NJ.

Larman, C. (2004) *Agile & Iterative Development*. Addison-Wesley, Boston.

Litsegård, P. & Sundmark, T. (2010) Illuminatio: Informationssystem för att tillgängliggöra vetenskapligt kvalitetssäkrad kunskap avseende utveckling av ledningssystem. FOI Memo 3358.

Mazur, G. (1992) Voice of the Customer Table: A Tutorial. In *Transactions from the Fourth Symposium on Quality Function Deployment*, 105-111.

McGovern, J., Ambler, S. W., Stevens, M. E., Linn, J., Jo, E. K. & Sharan, V. (2003) *The Practical Guide to Enterprise Architecture*. Prentice Hall PTR, New Jersey.

Mizuno, S. & Akao, Y. (eds.). (1994) *QFD: The Customer-Driven Approach to Quality Planning and Development*. Tokyo: Asian Productivity Organization.

OMG (Object Management Group) (2001) *Unified Modeling Language Specification*, Version 1.4.

Pilemalm, S. & Hallberg, N. (2008) Exploring Service-Oriented C2 Support for Emergency Response for Local Communities. In *Proceedings of the 5th International Conference on Information Systems for Crisis Response and Management (ISCRAM 2008)*, Washington, DC.

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](http://www.jot.fm/issues/issues%2007%20October/paper10)

Sindre, G. & Opdahl, A. L. (2000) Eliciting Security Requirements by Misuse Cases, TOOLS Pacific 2000: 37th International Conference on Technology of Object-Oriented Languages and Systems, Sydney, Australien.

Sommerville, I. (2001) *Software engineering*, 6th edition. Addison-Wesley, Boston, MA.

Sommerville, I., Sawyer, P. & Viller, S. (1998) Viewpoints for requirements elicitation : a practical approach. In *Proceedings of the Third IEEE International Conference on Requirements Engineering*, Colorado Springs, Colorado, USA.



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.

Waern, Å., Lindahl, B., Bengtsson, J., & Westerdahl, L. (2010) Nyttiggörande av forskning KUPAL. Projektsammanfattning v9.0. FOI Memo 3243.

Wieggers, K. E. (2003) *Software requirements*. Microsoft press, Redmond WA.

Schuler, D. & Namioka, A. (eds.) (1993). *Participatory Design: Principles and Practices*. Hillsdale, NJ: Lawrence Earlbaum.

Sundmark T. och Hallberg N. (2009) *Prototyp av databas för metoder och principer avseende arkitekturbaserad utveckling av ledningssystem*. FOI Memo 2911.

Söderström, J. (2010) *Jävla skitsystem*. Söderström.

Ölvingson, C., Hallberg, N., Timpka, T., and 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*, vol 68, no1-3, pp 165 – 174.

## Appendix A: Behov av stöd för utveckling av ledningssystem

I projektets initiala skede genomfördes en behovsanalys avseende vilket stöd samt vilka förutsättningar och aktiviteter inom FMs verksamhet som behövs för att möjliggöra utveckling av ledningssystem (Hallberg, Pilemalm & Westerdahl, 2008). I detta ingick att avgöra vad som är viktigt för att lyckas med utvecklingen och för att slutresultatet, det vill säga ledningssystemet, skall bli så ändamålsenligt som möjligt. De identifierade behoven avser att ge en grund för det fortsatta arbetet i projektet, men även att vara till nytta för andra initiativ som syftar till att effektivisera och förbättra FMs utveckling av ledningssystem.

Nedan är behoven strukturerade i områdena: Ledning, Utvecklingsprocess och aktiviteter, Kompetens, Utvecklingsstöd och Övrigt.

### **Ledning**

Denna kategori av behov omfattar: Ledning och styrning samt Samverkan.

#### *Ledning och styrning*

Det finns behov av övergripande koordinering och styrning av FMs ledningssystemutveckling. I detta ingår att ha en tydlig roll och ansvarsfördelning samt regler för hur ledningssystemutveckling skall genomföras. Grunden för denna ledning och styrning måste baseras på fastställda målsättningar i ledningssystemutvecklingen. Vidare skulle detta nyttjas för att samordna aktiviteterna och utvecklingsinitiativ för att kunna återanvända resultat över projektgränser.

#### *Samverkan*

Det finns behov av samverkan mellan olika aktörer inom ledningssystemutvecklingsområdet. Denna samverkan måste koordineras och styras.

### **Utvecklingsprocess och aktiviteter**

Denna kategori omfattar: Ledningsutvecklingsprocess, Behovsanalyser, Kravhantering, Design och Utvärdering.

#### *Ledningsutvecklingsprocess*

Det finns behov av en tydlig och väl definierad ledningsutvecklingsprocess för att erhålla en förutsättning för effektiv utveckling av ledningssystem. Ökad tydlighet skulle medföra bättre samverkan såväl som minimerat informationstapp, vilka båda skulle bidra till högre kostnadseffektivitet i ledningssystemutveckling. I denna utvecklingsprocess är hänsyn till integration av ledningsmetodik- och teknikutveckling att betrakta som nödvändigt.

#### *Behovsanalyser*

Det finns behov av att genomföra behovsanalyser, i syftet att säkerställa erhållandet av nytta för verksamheten.

#### *Kravhantering*

Det finns behov av att genomföra kravhantering, vilket innefattar att specificera, prioritera och säkerställa konsistens. Vidare måste krav avseende COTS (Commercial off-the-shelf) kunna hanteras.

### *Design*

Det finns behov av att hitta tillfredsställande lösningar som även fungerar för FM som helhet, och inte bara för enskilda delar. Design inbegriper även förfarandet med att säkerställa samma och liknande behov för samma tekniska lösning, i syftet att minska kostnader för utveckling, drift och underhåll.

### *Utvärdering*

Det finns behov av att resultatet av ledningssystemutveckling utvärderas. En sådan utvärdering fokuserar på att kontrollera själva genomförandet såväl som måluppfyllnaden, i syfte att förbättra utvecklingsprocessen. I detta ingår att:

- utvärdera levererade produkter och resultat
- utvärdera befintliga ledningssystem i bruk, för att skapa en grund för utvecklingen av framtida system
- säkra kvaliteten hos koncept i skarpa övningar, för att undvika att koncept enbart förblir teoretiska beskrivningar.

### **Kompetens**

Denna kategori omfattar: Utvecklingsteam, Användarmedverkan, Utnyttja erfarenheter och Utnyttja forskning.

### *Utvecklingsteam*

Det finns behov av ett flertal olika och kompletterande kompetenser i samband med utvecklingen av ledningssystem. Exempel på ingående kompetenser i utvecklingsteam är teknisk kompetens, systemutvecklingskompetens, militär kompetens, taktisk kompetens, ledningskompetens, verksamhetskompetens och pedagogisk kompetens.

### *Användarmedverkan*

Det finns behov av att kunna involvera användarrepresentanter i utvecklingsarbetet. Detta för att nyttja befintlig kompetens så effektivt som möjligt, för att säkerställa att de nyttoeffekter som eftersträvas faktiskt erhålls och att inte ”oanvändbara” funktioner och system utvecklas.

### *Medvetenhet*

Det finns behov av en god förståelse då det är användningen av teknik som ger effekt. Ökad insikt i att dåliga IT system (för-)stör fungerade verksamhet bör också betraktas som integrerat i behovet av medvetenhet.

### *Utnyttja erfarenheter*

Det finns behov av att överföra sammantagna erfarenheter, specifikationer och system från ett projekt till andra.

### *Utnyttja forskning*

Det finns behov att bättre utnyttja forskningsresultat för att erhålla förbättringar av utvecklingsarbetet, så väl med avseende på utvecklingsmetodik som på innehåll.

### **Utvecklingsstöd**

Denna kategori innehåller Utvecklingsmetoder, Dokumentation, Spårbarhet, Modellering, Verktysstöd, Systemkarta och Helhetsbild

#### *Utvecklingsmetoder*

Det finns behov av tillfredställande utvecklingsmetoder. Dessa skall vara gemensamma, bland annat för att medarbetare lättare skall komma in i andra projekt och verksamheter.

#### *Dokumentation*

Det finns behov av gemensamma dokumentationsformat för resultat, erfarenheter och kunskap. Detta för att underlätta personalomsättning och förbättra informationsflödet i utvecklingsprojekt.

#### *Spårbarhet*

Det finns behov av spårbarhet för att följa upp förändringar och motivera nyttan med genomfört arbete. Spårbarhet skall återfinnas mellan verksamhetens behov, krav, kravställare och realiseringar samt lösningar och faktiska förmågor.

#### *Modellering*

Det finns behov av att modellera de förmågor som karaktäriserar verksamheter och ledningssystem. Detta innefattar modellering som stödjer liknade aktiviteter och för att främja användarmedverkan i utvecklingsprocessen.

#### *Verktysstöd*

Det finns behov av olika former av datorbaserade verktysstöd för att underlätta utvecklingsarbetet.

#### *Systemkarta*

Det finns behov av systemkartor och interoperabilitet mellan system. Dessa innehåller med nödvändighet information om vilka ledningssystem som behöver kommunicera med varandra.

#### *Helhetsbild*

Det finns behov av att tydligt åskådliggöra det tänkta systemet och samma systems roll i dess omgivning. I detta ingår att beskriva systems externa egenskaper och dess sammanhang.

### **Övrigt**

Denna kategori innehåller Systemanskaffning, Säkerhet och Resurser

#### *Systemanskaffning*

Det finns behov av en effektiv systemanskaffning som är anpassad till pågående förändringarna i FM och som tillhandahåller stöd för upphandling mot industrin.

***Säkerhet***

Det finns behov av effektiva säkerhetslösningar, som inte blir begränsande för användandet av funktioner och system.

***Resurser***

Det finns behov av utökade resurser i form av tid, pengar, personal och faciliteter för ledningssystemutveckling. I detta ingår att vidareutveckla faciliteter för experiment, simulering och träning.

## Appendix B: Utvecklingsprinciper

Detta avsnitt beskriver ett antal centrala *principer* som är viktiga för att lyckas med systemutveckling. Dessa utvecklingsprinciper beskrivs utförligare i (Hallberg, Pilemalm, Westerdahl et al, 2008) och med relationer till de identifierade behoven avseende vilket stöd samt vilka förutsättningar och aktiviteter inom FMs verksamhet som behövs för att möjliggöra utveckling av ledningssystem.

- *Väldefinierad utvecklingsprocess* med tydliga aktiviteter och mål, vilka är införstådda och accepterade av berörda parter.
- *Kompetens* i form av tillgång till rätt personal måste finnas under hela utvecklingen med bibehållen kontinuitet.
- *Tydligt ansvar* måste vara delgivet till grupper med ansvaret att genomföra utvecklingsprojekt. Det vill säga de måste ha erhållit tydliga mandat.
- *Tillgång till användarrepresentanter* med direkt insikt i användningssituationen är centralt för att säkerställa en hög kvalitet hos utvecklade system.
- *Tydliga systemgränser*. Innan några andra utvecklingsaktiviteter initieras är det viktigt att tydligt definiera vilket system som skall utvecklas och vad som tillhör systemomgivningen (Sommerville, 2001).
- *Kontextuell förståelse* för den omgivning (kontext) som systemet skall fungera i är nödvändig för att kravställa det.
- *Tydlighet avseende begrepp* är nödvändigt för att undvika missförstånd som kan få allvarliga konsekvenser för utvecklingen.
- *Intressentanalys* fastställer vilka intressenterna till ett system är. Det är deras upplevelse av problem, brister och möjligheter som motiverar utvecklingen.
- *Behovsanalys* är nödvändig för att fastställa vilka behov intressenterna har, vilket innefattar såväl uttalande som outtalade behov (Kano, 1995).
- *Kravhantering* är att fastställa och dokumenterar de krav den vidare utvecklings skall baseras på. Krav utgör ”bryggan” som binder samman behoven med designen.
- *Designa för användning* är att ta hänsyn till hur användarna skall använda system i dess utformning. I detta ingår att ta hänsyn till användarnas begränsningar. Prototyper av systemets gränssnitt och

rekommendationer för design kan nyttjas för att säkerställa att system fungerar som användarna har tänkt sig.

- *Spårbarhet* skall upprätthållas dubbelriktat för att säkerställa att det är ”rätt” system som utvecklas och för att kunna motivera exempelvis val av design och lösningar mot behov. Exempel på spårbarhet är ”behov - krav” och ”krav - design” (Hallberg, 1999; Hull, Jackson & Dick, 2005).
- *Prioritering* görs för att fokusera tillgängliga resurser på det som ger störst nytta för intressenterna. Prioriteringar görs (explicit eller implicit) i samtliga projekt, men om valet av lösningars skall baseras på intressenternas behov krävs spårbarhet.
- *Kontinuerlig kvalitetssäkring*. För att säkerställa att det är rätt system som utvecklas krävs kontinuerlig kvalitetssäkring av resultaten från de ingående aktiviteterna (Collin, 2003).
- *Modellering* och modeller användas för att beskriva, begripliggöra och dokumentera komplexa samband och system i olika utvecklingsfaser. Det nyttjas även för att beskriva och dokumentera systemomgivningar, så väl existerande som tänkta (Hallberg & Fransson, 2001; Hay, 2003; Sommerville, 2001; Friedenthal, Moore & Steiner, 2008).
- *IT-säkerhetsaspekter* måste beaktas tidigt i utvecklingen och integreras med övrig utveckling (Hallberg & Hallberg, 2006).

## Appendix C: Begrepp inom systemutveckling

Ett problem in systemutvecklingsområdet är att begrepp inte används stringent. Hallberg, Pilemalm, Westerdahl et al (2008) och Hallberg, Pilemalm och Jungert (2010) beskriver en ontologisk ansats för att definiera de centrala begreppen och deras relationer inom systemutveckling. Ontologin innehåller fyra övergripande kategorier av begrepp; (1) Generella begrepp, (2) Beskrivningsbegrepp, (3) Utvecklingsprocessbegrepp och (4) Systemuppträdandebegrepp.

- Generella begrepp innefattar: System, Ledningssystem, Systemomgivning, Komponent, Vy, Arkitektur, Arkitekturramverk, Ramverk, Systemutveckling.
- Beskrivningsbegreppen innefattar begrepp som bidrar till att beskriva systemet i dess olika utvecklingsfaser, så som Utsaga, Behov, Krav, Design och Realisering.
- Utvecklingsprocessbegreppen innefattar aktiviteter som genomförs under systemutveckling. Dessa aktiviteter är Kontextanalys, Behovsanalys, Kravhantering, Design, Realisering, Införande, Verifiering, Validering och Exploatering.
- Systemuppträdandebegrepp innefattar begrepp som beskriver hur system ser ut, fungerar och uppträder. Dessa begrepp innefattar Funktion, Kapacitet, Förmåga, Egenskap, Tjänst och Användbarhet.

### Systemutvecklingsaktiviteter

Det finns ett antal olika sätt att beskriva systemutveckling baserat på ingående *aktiviteter*. I projektet har följande aktiviteter nyttjas för att beskriva systemutveckling: Kontextanalys, Behovsanalys, Kravhantering, Design, Realisering, Verifiering och Validering. Exploatering är en viktig aktivitet, men som inte riktigt passar in i den process som de övriga utgör. Mer omfattande beskrivning vad de innefattar finns i Hallberg, Pilemalm, Westerdahl et al (2008).

- Kontextanalys syftar till att analysera den omgivning i vilket det system som skall utvecklas avses fungera i. Verksamhetsanalys och intressentanalyser är vanliga former av kontextanalys.
- Behovsanalyser syftar till att fånga intressenternas och verksamheternas verkliga behov (Hallberg, Andersson, & Westerdahl, 2005; Ficalora & Cohen, 2009).
- Kravhantering syftar till att fastställa krav på system, där varje krav skall vara korrekt formulerat, entydigt, förståeligt, komplett, konsistent, verifierbart, modifierbart, spårbart och prioriterat samt att kravmängden är detaljerad till en nivå lämplig för den fortsatta utvecklingen (Wiegers, 2003).



- Design innebär att beskriva hur systemet ska utformas och fungera, avseende såväl synliga som osynliga aspekter av systemet som utvecklas. Resultatet är därmed en eller flera modeller av systemet som utvecklas.
- Realisering innebär att omsätta designen till ett fungerande system, genom att systemet införskaffas, systemkomponenter införskaffas och konfigureras till ett fungerande system eller att system byggs från grunden.
- Verifiering syftar till att avgöra om en design alternativt ett system motsvarar de fastställda kraven.
- Validering syftar till att avgöra om kraven, designen eller systemet motsvarar det faktiska behovet, det vill säga om dessa ger önskad effekt för intressenterna.
- Exploatering syftar till att studera om ett existerande system kan nyttjas även för andra ändamål; ge nytta som det ursprungligen ej var avsett för.

## Appendix D: Utvecklingsmetoder

I detta kapitel listas de metoder som har studerats under projektet, där begreppet metod skall beaktas i sin vidaste mening. De beskrivna metoderna ska ses som goda exempel som kan tillämpas var och en för sig, i olika kombinationer och på olika sätt, beroende på projektets syfte och kontext. Metoderna beskrivs utförligare i Hallberg, Pilemalm, Westerdahl et al (2008) liksom deras relation till identifierade behov.

- Dokumentanalyser betecknar strukturerade studier av dokument. Detta används typiskt i den initiala utvecklingen, exempelvis för att studera en verksamhets struktur, verksamhet, vision, mål, aktörer och aktiviteter samt för att identifiera utsagor som grund för behovsanalyser. Dokumentanalyser kan användas både vid verksamhetsbeskrivning och för att finna behov och förändringar, exempelvis i måldokument, och vid identifiering av krav på IT-stöd.
- Deltagande observation är observation av människor som agerar i deras naturliga miljö (Hasu & Engeström, 2000). Metoden härstammar från antropologi och etnografi men används idag inom systemutveckling. Metoden har visat sig var effektiv då det gäller förändringsarbete och teknikstöd förändring av arbetsuppgifter.
- Intervjuer används för att samla information om och förstå människors erfarenheter och upplevelser; hur de konstruerar sin verklighet. De når en snävare grupp av individer än exempelvis enkäter och kan vara lämpliga när forskaren/utvecklaren vill fånga exempelvis enskilda aktörers perspektiv (Bernard, 2000)
- The Critical Incident Technique (CIT) bygger på användares självrapportering av incidenter som är kritiska i deras arbete, både incidenter som leder till tillfredsställelse och incidenter som leder till upplevda problem (Flanagan, 1954). Insamlad information kan nyttjas som grund för att genomföra utvecklingsinsatser.
- Future Workshops syftar till att aktivt involvera användare i systemutvecklingsprocesser genom att låta dem själva reflektera över sin situation och problem i relation till den, förändringsbehov i verksamheten och behov av informationstekniskt stöd för förändringar (Kensing & Halskov Madsen, 1991). Future Workshops genomförs i de tre faserna kritikfas, fantasifas och implementationsfas där användare först beskriver den egna arbetssituationen och dess problem, sedan genererar futuristiska lösningar till problemen och i nästa steg omvandlar dessa till tekniskt och organisatoriskt implementerbara lösningar (Jungk, & Müllert, 1987).

- Unified Modeling Language (UML) är ett standardiserat modelleringsspråk för visualisering, specificering, konstruktion och dokumentation (OMG, 2001). Språket inkluderar en rad notationer och diagram för att beskriva såväl dynamiska som statiska egenskaper hos system.
- Användningsfall beskriver hur system används eller är tänkta att användas. Det består av en text som beskriver själva användningen, samt figur som illustrerar aktörers interaktion med systemet (Kulak & Guiney, 2003). I Hallberg, Ölvander och Törne (2006) beskrivs en mall för användningsfall.
- Prototyper är modeller av system. Dessa kan användas i kommunikationen med användare för att ge dessa möjligheter att påverka utformningen av systemet. De kan också nyttjas som ett medium att kommunicera mellan utvecklare samt att kommunicera med de som skall realisera systemet.
- Prototypbaserad utveckling är en metod för att utveckla system. Det finns en rad tekniker för utvärdering av prototyper, exempelvis scenariobaserad utvärdering. Scenarier är beskrivningar av konstruerade användarsituationer med avseende på arbetsuppgifter som ska utföras med hjälp av systemet; en beskrivning av en aktivitet definierad som en rad, ofta sekventiella, handlingar (Carroll, 2000; Hallberg, Ölvander, Johansson, & Törne, 2006). I scenariobaserade utvärderingar interagerar användarrepresentanter med systemet och utfallet kan mätas både kvantitativt, med avseende på exempelvis felfrekvens, och kvalitativt, med avseende på subjektiva upplevelser av systemet.
- Tjänsteorienterad arkitektur (Service Oriented Architecture, SOA) ger möjligheten att utveckla distribuerade system som organiseras som en struktur för kommunikation mellan system genom kommunicerande tjänster (Pilemalm & Hallberg, 2008; Ingermarsson, Eriksson & Hallberg, 2009; McGovern, Ambler, Stevens, Linn, Jo, & Sharan, 2003).
- Quality Function Deployment (QFD) är ett kvalitetssystem med ursprung i japanskt kvalitetstänkande, och är nära besläktat med Total Quality Management (TQM) (Mizuno, S. & Akao, 1994; Bergman & Klefsjö, 1994). QFD främst lämpat för att stödja identifieringen av användarnas behov och omvandlingen av dessa behov till krav på systemet, och därefter till bestämning av designattribut (funktioner, egenskaper mm). QFD består av en filosofi, ett antal kvalitetsverktyg och appliceringsmodeller (Hallberg, 1999).
- Kundrösttabell (eng: Voices of Customers Table, VCT) är ett av många kvalitetsverktyg som utvecklats inom QFD (Mazur, 1992). I

kundrösttabellen skrivs kundens önskningsin som utsagor. Dessa utsagor analyseras därefter utifrån (1) vem som anses behöva dem, (2) vad de anser sig behöva, (3) när de behöver det, (4) var det behöver användas, (5) varför behovet finns och (6) hur de vill att det skall fungera. Analys med stöd av kundrösttabell har som mål att kunna formulera faktiskt behov. Kundrösttabellens primära funktion är att underlätta analysen av vad kunden egentligen behöver.

- Användarmedverkan motiveras oftast av att det är användarna som känner sin domän bäst och som vet vilket behov av stöd de har. Det finns många olika sätt att involvera användarna, till olika grad samt att ge dem olika mandat för att påverka. Ansatser som betonar aktiv användarmedverkan är exempelvis Deltagande design and Joint Application Design (JAD) (Schuler & Namioka, 1993; Carmel, George, & Nunamaker, 1992).
- Arkitekturramverk kan ses som en metod för utveckling av systemarkitekturer för att erhålla sammanhängande och konsekventa modeller. Det finns många arkitekturramverk, exempelvis MODAF, DoDAF och TOGAF.
- Misuse cases vilket är en utveckling av traditionella användningsfall (Sindre & Opdahl, 2000). Genom att visa på möjliga hot i samma modell som de funktionella kraven ges möjligheten att komplettera modellen med funktionella krav som kan reducera konsekvenserna av hoten.
- Viewpoints är inte specifikt en metod för att utvinna säkerhetsgenskaper under systemutvecklingen (Sommerville, Sawyer, & Viller, 1998). Dock fungerar metoden för att identifiera såväl funktionella som icke-funktionella krav för hela systemet. Viewpoints beskriver uppgifter ur användares perspektiv. I dessa skall det framgå vem användaren är (roll) och vilken uppgift användaren löser. Kopplat till varje användare finns ett antal concerns (angelägenheter) som är tillämpliga för användaren.
- UMLsec är en vidareutveckling av UML för att föra in säkerhetsgenskaper redan i kravfasen (Jürjens, 2005).
- Kvalitetsdriven kravhantering är en metod för att identifiera krav som baseras på kvalitetsverktyg som finns beskrivna i QFD litteraturen och genomförs i sex steg: (1) Insamling av data, (2) Identifiering av utsagor, (3) Identifiering av behov, (4) Analys av behov, (5) Identifiering av krav och (6) Analys av krav (Hallberg, Andersson & Westerdahl, 2005).
- Rational Unified Process (RUP) är ett processramverk som omfattar hela utvecklingscykeln för system, i första hand mjukvara (Kruchten, 2004).

RUP stödjer evolutionär systemutveckling och är i hög grad modellbaserad baserat på Unified Modeling Language (UML).

- Verksamhetsutvecklingsmetoden för Ledningssystem (VUM-LS) bygger på de principer och ansatser från RUP, UML, användarcentrerad systemutveckling, principer om användbarhet och Quality Fuctionen Deployment. Metoden har utvecklats baserat på erfarenheter från utvecklingsprojekt inom FM gjorda av FOI. Målet med VUM-LS är att beskriva en beprövad metod för verksamhetsutveckling där ledningssystemets samtliga delar; metod, personal, organisation och teknik, utvecklas i harmoni.

## Appendix E: Fallstudier

I projektet genomfördes följande delstudier i form av 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, (6) Erfarenheter av modellbaserad utveckling, (7) Kvalitets- och beslutsgrindar, (8) Databas: Metoder och principer för utveckling av ledningssystem, och (9) Kvalitetssäkring vid kravhantering.

Fallstudie 1-8 finns beskrivna i Hallberg, Hansson, Jungert et al (2009) och den nionde fallstudien finns beskriven i rapporten Hanson, Granlund, Hallberg & Pilemalm (2010). En fortsättning på den åttonde fallstudien återfinns i Litsegård & Sundmark, (2010).

- Fallstudien *Modellbaserad utveckling* beskriver vad modellbaserad utveckling är, vad den syftar till och hur den 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 bedrivas iterativt och det anses viktigt att kontinuerligt säkerställa kvalitén på de modeller som skapas.
- Fallstudien *Kvalitetsdriven kravhantering för utveckling av insatsledningssystem* 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, 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 för framgång och den så kallade kundrösttabellen är ett användbart verktyg för att tolka användares utsagor och formulera dessa som behov.

- Fallstudien *Kvalitetsdriven kravhantering för utveckling av avvikelshantering* beskriver tillämpningen av Kvalitetsdriven kravhantering 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.
- Fallstudien *Kravhantering och MODAF* beskriver 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.
- Fallstudien *Erfarenheter av MODAF* beskriver ett intervjuinstrument för att inhämta erfarenheter kring användningen av MODAF. Motivet för detta är att MODAF har valts av Försvarsmakten som arkitekturramverk, samtidigt som det saknas vetenskapliga studier kring tillämpningen av MODAF. Vad är det i tillämpningen som är svårt, hur kan detta hanteras och vilka är dess styrkor? I fallstudien tillämpas intervjuinstrumentet dels för att inhämta erfarenheter kring MODAF, men också för att validera instrumentet. Resultatet 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 Försvarsmakten. Införandet kräver även att fördelarna med ett gemensamt ramverk tydligt presenteras. Intervjuinstrumentet fanns fungera väl, och kan nyttjas för betydligt mer omfattande studier.
- Fallstudien *Erfarenheter av modellbaserad utveckling* beskriver erfarenheterna av modellbaserad utveckling i form av fördelar, nackdelar och utmaningar med utgångspunkt från Försvarsmaktens 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.

- Fallstudien *Kvalitets- och beslutsgrindar* beskriver 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 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.
- Fallstudien *Databas: Metoder och principer för utveckling av ledningssystem* har studerat hur ett system för att dokumentera och tillgängliggöra den kunskap som byggts upp inom projektet skall utformas; vars syfte är att lagra vetenskapligt kvalitetssäkrade principer och metoder avseende systemutveckling. Systemdesignen har förändrats avsevärt under fallstudiens gång då det gått från att ha varit databascentrerat till att slutligen ha blivit dokumentcentrerat. Den slutgiltiga prototypen av systemet, *Illuminatio*, fokuserar tack vare denna förändring på att presentera den lagrade informationen på ett mer intuitivt och lättillgängligt sätt. Detta åstadkoms huvudsakligen genom att använda sig av ett användargränssnitt som påminner mycket om det gränssnitt som Google använder för sin välkända sökmotor.
- Fallstudien *Kvalitetssäkring vid kravhantering* beskriver en metod för att formulera, granska och kvalitetssäkra behov och krav. En kravspecifikation med hög kvalitet är nödvändig för att lyckas med systemutveckling. Dessvärre innehåller många kravspecifikationer brister som kan undvikas redan då krav nedtecknas genom att författaren har kunskap om hur krav ska formuleras och är noggrann vid formuleringen. Genom granskning av behov och krav kan brister och felaktigheter detekteras innan specifikationen nyttjas i det vidare utvecklingsarbetet. Studien, som innefattar litteraturoversikt, metodutveckling samt initial verifiering, har resulterat i en metod som skall reducera antalet brister i specifikationer. Metoden omfattar att formulera behov, formulera krav, granska behov samt granska krav. Formulera behov och krav görs utifrån specifika attribut som skall tillgodose att rätt information dokumenteras. Dessa formuleringar



granskas och kvalitetssäkras. Till metoden har material i form av tabeller, checklistor och handledningar tagits fram för att stödja utförandet i respektive moment. Inom ramen för studien genomfördes en initial verifiering av momentet formulera krav. Tio respondenter med erfarenhet av systemutveckling deltog i verifieringen, där en mängd krav jämfördes med krav som omformulerats med studiens metod. Respondenterna ansåg att de omformulerade kraven var bättre formulerade och strukturerade, entydiga samt test- och verifierbara, men för specifika. De omformulerade kraven tillförde struktur och tydlighet, men begränsade förståelsen för systemet.