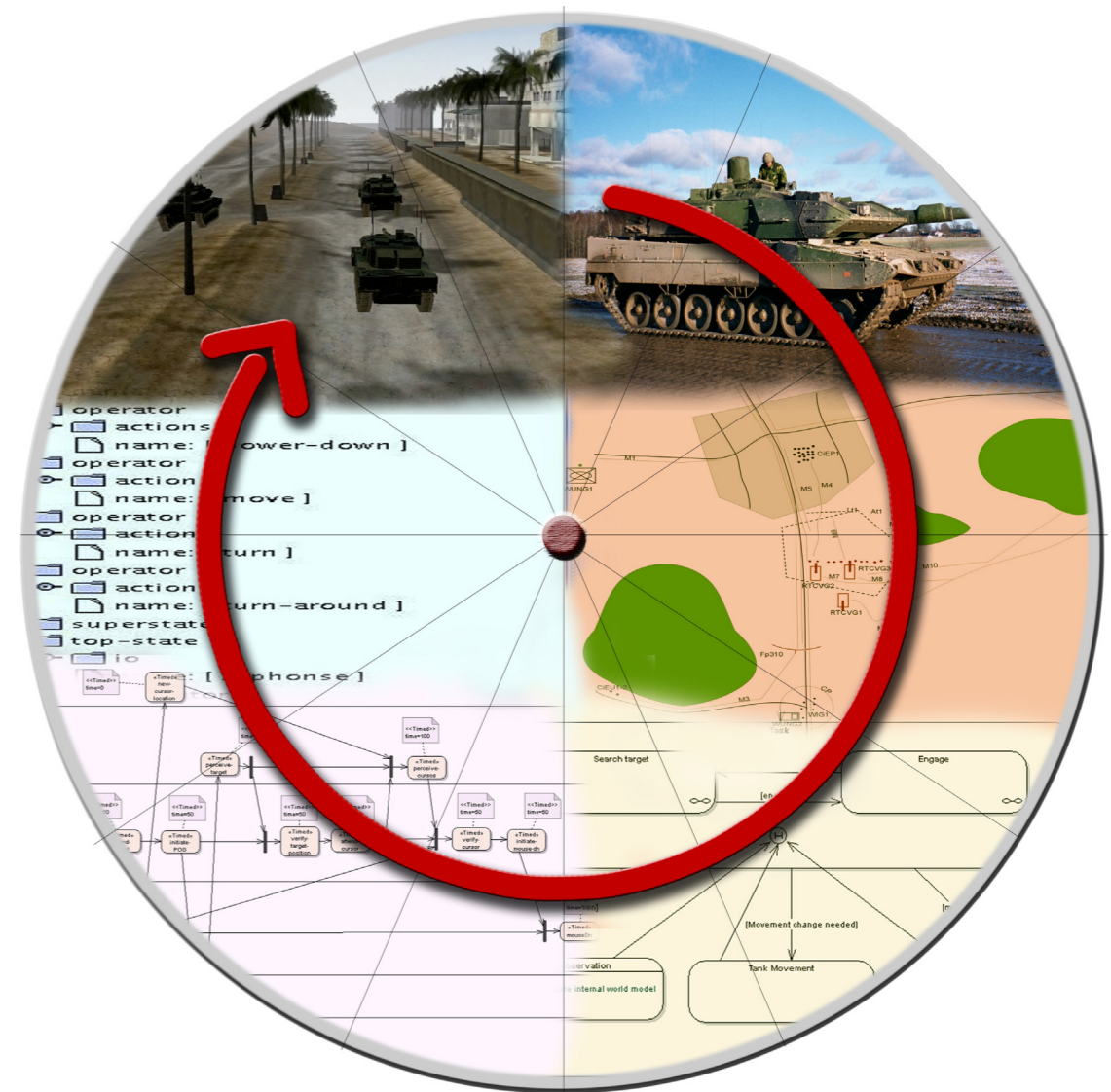


STEN-ÅKE NILSSON, JOHAN PELO, MIKAEL LUNDIN, PETTER ÖGREN



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.

Sten-Åke Nilsson, Johan Pelo, Mikael Lundin,  
Petter Ögren

# Från CGF till HBR och operationsområdesanalys

Årsrapport 2008

Titel	Från CGF till HBR och operationsområdesanalys
Title	From CGF to HBR and Operation area analysis
Rapportnr/Report no	FOI-R--2627--SE
Rapporttyp Report Type	Användarrapport User report
Sidor/Pages	55 p
Månad/Month	December
Utgivningsår/Year	2008
ISSN	ISSN 1650-1942
Kund/Customer	FM
Forskningsområde Programme area	2. Operationsanalys, modellering och simulering 2. Operational Research, Modelling and Simulation
Delområde Subcategory	21 Modellering och simulering 21 Modelling and Simulation
Projektnr/Project no	E7118
Godkänd av/Approved by	Martin Rantzer
FOI, Totalförsvarets Forskningsinstitut Avdelningen för Informationssystem	FOI, Swedish Defence Research Agency Information Systems
164 90 Stockholm	SE-164 90 Stockholm

## Sammanfattning

Forskningsområdet "Datorgenererade aktörer" (CGF) bygger på många olika vetenskaper som datalogi, M&S, AI och fysiologi. Begreppet CGF har under senare år ofta ersatts av begreppet HBR, Human Behavior Representation, för att i högre grad införliva de mänskliga egenskaperna hos aktörerna. För att göra detta krävs kunskap inom psykologi och kognitionslära. På detta sätt har man breddat definitionen av CGF och visar att denna teknologi kan användas i många sammanhang där man inte primärt vill infoga en "stridskraft".

Projektet har under senare år identifierat en intresse- och trendförskjutning från de användningsområden som kunde skönjas under CGF förstudie 2001, till närliggande områden som påvisar i vissa stycken annorlunda och nya behov av syntetiska aktörer.

Denna förskjutning för med sig konsekvenser för forskningen inom mänsklig beteenderepresentation (HBR). Behovet av användning av syntetiska aktörer i större skala, och för skarpa frågeställningar, har vuxit kraftigt. Produktionen av syntetiska aktörer för att möta dessa behov har inte ökat i takt med behoven. Den kunskap som finns om arkitekturer och datalogisk utformning av ramverk och bibliotek för syntetiska aktörer behöver kompletteras med insikt om hur produktionsprocessen ska utformas för att möta dessa behov.

Svenska försvarets inriktning att i allt högre grad delta i internationella fredsbevarande insatser kräver kunskap om betydelsen av kulturella och sociala skillnader och effekter. Dessa är i dag mycket viktiga att kunna påvisa i utbildning och träning, både vad gäller inverkan på samarbete såväl som vilka effekter som det har i fråga om kontakt med befolkningen i ett insatsområde.

I mångt och mycket är utvecklingen av en "CGF med mänskliga egenskaper" eller inom HBR beroende av att många personer med olika kompetenser kan arbeta tillsammans. FOI är väl rustat för forskning inom HBR, eftersom där finns kompetenser inom alla ovan nämnda vetenskaper.

Kunskap och förståelse för betydelsen av sociala och kulturella faktorer är av direkt intresse för de FM styrkor som skall verka i ett område. Den som ämnar beskriva dessa faktorer i en simuleringsmodell behöver en teoretisk grund för att beskriva hur dessa faktorer påverkar beteende. För att modellen skall vara användbar för FM vid träning, utbildning samt efteranalys, krävs fortsatta forskningsinsatser för att finna en vetenskaplig grund som modeller för militära ändamål kan baseras på.

Nyckelord: M&S, CGF, datorgenererade styrkor, simulering, kognitiv modellering, HBR, arkitekturer, MSI, OneSAF, VV&A

## Summary

The area of "Computer Generated Forces" (CGF) is based on many sciences such as computer science, M&S, AI and physics. The concept of CGF has in recent years often been replaced by the concept of HBR, Human Behavior Representation, to better incorporate the human characteristics of the players. To do this requires knowledge in psychology and cognitive science. This has broadened the definition of CGF, and shows that this technology can be used in many contexts where it is not primarily intended to insert a "battle force".

The CGF project at FOI has in recent years identified an interest and trend shift from the uses that could be detected during the CGF study in 2001, to nearby areas that demonstrate somewhat different and new needs of synthetic actors.

This shift brings along implications for research in human behavior representation (HBR). The need for the use of synthetic actors in a larger scale, and the sharp questions, has grown rapidly. The production of synthetic actors to meet those needs has not increased in correspondence with needs. The knowledge of architectures and computational design of the frameworks and libraries for synthetic actors need to be supplemented with an understanding of how the production process should be designed to meet those needs.

The increasing focus of the Swedish Armed Forces to participate in international peacekeeping operations requires knowledge of the importance of cultural and social differences and effects. Today these are very important to be able to demonstrate in education and training, both in terms of impact on cooperation as well as the effects it has in terms of contact with people in a field of action.

The development of a "CGF with human characteristics" or within HBR is depending on that many people with different skills can work together. FOI is well equipped for research in HBR, because there exist skills in all the above mentioned sciences.

Knowledge and understanding of the importance of social and cultural factors are of immediate interest to forces that are to operate in an area. Anyone who intends to describe these elements in a simulation model needs a theoretical basis for describing how these factors affect behavior. To be useful for the Swedish Armed Forces for training, education, and after-analysis, modelling requires further research efforts to find a scientific basis upon which models for military purposes can be based.

Keywords: M&S, CGF, computer generated forces, simulation, cognitive modelling, HBR, architectures, MSI, OneSAF, VV&A

# Innehållsförteckning

<b>1</b>	<b>Inledning</b>	<b>7</b>
1.1	Syfte med denna rapport .....	7
1.2	Bakgrund.....	7
1.3	Läsinstruktion .....	8
<b>2</b>	<b>CGF-projektet vid FOI</b>	<b>9</b>
2.1	Nyttan med arbetet .....	11
2.2	Definition av CGF, intresseområden.....	11
2.3	Användning av CGFer .....	11
<b>3</b>	<b>Resultat</b>	<b>13</b>
3.1	Forskningsområdet .....	13
3.2	Utvecklingsprocessen .....	13
3.3	Metoder och verktyg .....	13
3.3.1	Datainsamling .....	13
3.3.2	Kunskapsbearbetning .....	14
3.3.3	Ramverk för kognitiv modellering .....	14
3.3.4	PMF, Performance Modifier/Moderator Functions.....	15
3.3.5	VV&A.....	15
3.3.6	Guidelines .....	16
3.3.7	Utvärdering av verktyg: AI.Implant och STAGE .....	16
3.3.8	Ramverk för simulering .....	17
3.3.9	Prototyping.....	17
3.3.10	Hifi Engine.....	18
3.3.11	Modellbibliotek/biblioteksstruktur/arkitektur .....	18
3.4	Internationellt samarbete .....	19
3.4.1	NATO HFM 128 .....	19
3.4.2	AFRL-FLSC .....	19
3.5	Närliggande användnings- och utvecklingsområden .....	19
<b>4</b>	<b>Diskussion</b>	<b>21</b>
<b>5</b>	<b>Förkortningar</b>	<b>23</b>
<b>6</b>	<b>Litteraturförteckning</b>	<b>25</b>
<b>7</b>	<b>Bilaga 1, Reserapport 17th BRIMS</b>	<b>27</b>
<b>8</b>	<b>Bilaga 2, NATO RTO HFM-128</b>	<b>33</b>
<b>9</b>	<b>Bilaga 3, RTO Award</b>	<b>37</b>
<b>10</b>	<b>Bilaga 4, Hifi Engine</b>	<b>39</b>

10.1	Projects.....	40
10.2	Publications .....	41
<b>11</b>	<b>Bilaga 5, Athena</b>	<b>43</b>
11.1	Introduction.....	43
11.1.1	Motivation .....	44
11.2	Background .....	46
11.2.1	“AgentInterface”.....	46
11.2.2	The Merlin framework.....	46
11.3	The framework.....	49
11.3.1	Interfaces.....	49
11.3.2	Simulation engine interfaces .....	49
11.3.3	Simulation utility objects .....	50
11.3.4	HBR interfaces .....	51
11.3.5	The Soar adapter .....	51
11.4	Conclusion.....	54
11.5	Future work.....	55

# 1 Inledning

Sedan flera år har datorgenererade aktörer använts i många olika sammanhang men även idag satsas stora belopp t.o.m. på grundforskning inom området. Forskningsområdet har kopplingar till och bygger på många olika vetenskaper som t.ex. datalogi, M&S, AI, fysiologi såväl som psykologi och kognitionsvetenskap. Det förutses ha en hög potential för området M&S med många efterlängtrade, effektivitetshöjande tillämpningar.

## 1.1 Syfte med denna rapport

Denna rapport är en redovisning av 2008 års verksamhet men innehåller också en kort sammanställning av verksamhet inom CGF-området som bedrivits vid FOI de senaste 7 åren. Denna fleråriga verksamhet är noggrannare beskriven i de referenser som anges.

Rapporten kan tänkas fungera som en portal för alla som vill ha en introduktion till och insikt i både forskningsområdet och användningsmöjligheterna med datorgenererade aktörer.

Arbetet är finansierat med medel inom ramen för FoT M&S.

## 1.2 Bakgrund

Modellering och simulering (M&S) är ett kraftfullt verktyg som används för att stödja utveckling av framtida militära koncept och system. Modellering och simulering används också till att förbättra träning och användning av befintliga styrkor och utrustningar, samt för att i förväg analysera militära operationer i nya miljöer.

Det finns ett antal teknologier som har stor betydelse för resultatet när man utnyttjar simuleringar för militära ändamål. Datorgenererade styrkor (Computer Generated Forces, CGF) är en sådan teknologi. Datorgenererade styrkor som används i simuleringar erbjuder representationer av militära enheter på olika aggregationsnivåer, och används alltmer för att befolka distribuerade simuleringar.

Det har sedan början av 90-talet förekommit försök att modellera förband med ett så intelligent beteende att det inte fordras någon mänsklig operatör alls för att övervaka dem. Det mest kända av dessa tidiga försök var det s.k. Soar/IFOR-projektet (Soar, "State, Operator And Result"/Intelligent Forces), där man försökte skapa modeller av piloter som skall kunna fungera i alla normala situationer som inträffar under flyguppdrag. Man utvecklade också modeller av flygstridsledare som kommunicerar med de flygande pilotmodellerna under uppdragen. Dessa modeller har använts i ett antal övningar under åren. Bland annat kan man nämna de amerikanska STOW-E, 1994, samt Coyote- och Roadrunner-övningarna, 1998, som var flera dygn långa, fullständiga simulationer av luftkrig med hundratals agenter och bemannade simulatorer. Syftet med dessa CGFer har framförallt varit att skapa större, mer realistiska och meningsfulla simuleringar för taktiska övningar genom att simulera egna och fiendliga förband.

Datorgenererade förband kan också med fördel användas i förhandsanalys av planerade militära operationer, taktikutveckling, materielanskaffningsprocesser och utveckling av nya system. Dessa olika användningsområden ställer olika krav på de datorgenererade styrkorna och på modellerna som styr deras beteende. Om man t.ex. i en materielanskaffningsprocess vill studera det operativa utfallet av införandet av ett hjälmsikte behöver man ha mer detaljerade modeller av flygförarnas visuella perception än om de modellerade flygförarna "bara" skall vara motståndare i en BVR-strid.

Den nya krigföringen handlar i stor utsträckning om informationsövertag och att utifrån det kunna förutse motståndarens handlande. I och med detta kommer detaljerad modellering av beslutsfattandet och kognitionen hos fienden att behövas för att



informationsstriden skall kunna modelleras. Det kommer inte att räcka med enkla modeller av simulerade beslutsfattare, utan striden måste kunna simuleras på många olika aggregationsnivåer. Det kommer inte räcka med en utnötningssalgorithm för att avgöra striden, eftersom inte sidan med mest eldkraft nödvändigtvis vinner. Asymmetrisk krigföring kommer att höra till vardagen och för att kunna förutspå utfallet här måste man kunna modellera på individnivå. Det är också först då man kan få fungerande prediktiva analyser av hur informationskriget fungerar samt vilken inverkan kulturella och sociologiska faktorer har.

### **1.3 Läsinstruktion**

Rapporten är i huvudsak uppdelad i två avsnitt, ”CGF-projektet vid FOI” och ”Resultat”, Det första ger information om definitioner inom CGF-världen, visar en modell för utveckling, pekar ut de vetenskapliga grunderna samt diskuterar något om användning och nytta för Försvarmakten. Den andra delen redovisar innehållet i modellen efter vilken arbetet genomförts. Framför allt redovisas ett antal arbetsinsatser som gjorts avseende framtagning och/eller utvärdering av metodik och verktyg. Dessa är viktiga ingredienser i M&S-processen vid framtagning och användning av datorgenererade aktörer. Texten innehåller ett stort antal hänvisningar till andra rapporter och verk, varav några finns i utdrag i bilageform i denna rapport.

Främst de egenproducerade rapporterna bifogas denna rapport på en CD.

## 2 CGF-projektet vid FOI

År 2001 gjordes en förstudie med titeln ”Datorgenererade styrkor – metoder och möjligheter” (Castor et al, 2002) vid FOI. Denna rapport gav förutom bakgrund, beskrivning av nytta med och arkitekturgenomgång också förslag till inriktning av kommande arbete inom området datorgenererade aktörer.

Under 2002-2005 bedrevs projektet ”Datorgenererade styrkor” vid FOI. Syftet med projektet var att:

- kartlägga området ”Datorgenererade styrkor” och ”Datorgenererade beteendemodeller” både nationellt och internationellt,
- införskaffa kunskap inom området för att därigenom kunna bistå Försvarmakten i sådana frågor,
- skapa ett kontaktnät nationellt och internationellt,
- skapa ett ramverk för gemensamt nyttjande av beteendemodeller inom Försvarmakten.

Efter att ha genomfört förstudien under 2001 och fortsatt arbete år 2002 om CGF och CGF-utveckling fann projektet vid julen 2002 att andra forskare redan tänkt i ungefär samma banor om hur ett sådant arbete skall bedrivas. Från boken ”Modeling Human and Organizational Behavior” (Pew & Mavor, 1998) återges nedanstående text som beskriver ett rekommenderat sätt att arbeta inom detta område. Författarna har sammanställt en arbetsgrupps förslag till ett ramverk för arbetet med att utveckla “models of human behavior for use in military simulations” på beställning av DMSO. De fyra huvudpunkterna i detta ramverk är:

### **Samla och gör data om mänsklig prestation användbart**

Denna punkt handlar om att samla data som beskriver alltifrån grundläggande mänskliga förmågor till data som visar hur de militära enheterna i verkligheten agerar, samarbetar och kommunicerar. Dessa data behövs för många ändamål: för modellutveckling, för verifiering och validering, ackreditering och sist men inte minst för utveckling av nya teorier om hur människor och grupper av människor fungerar enskilt och tillsammans.

### **Utveckla procedur för att ackreditera modeller över mänskligt beteende**

För ackreditering krävs väl genomarbetade processer för verifiering och validering, samt för beskrivning och dokumentation över funktion och användning.

### **Satsa på att utveckla modeller för vissa fokuserade områden**

Andra rekommenderade och fokuserade aktiviteter för att komma vidare inom området är enligt nedan.

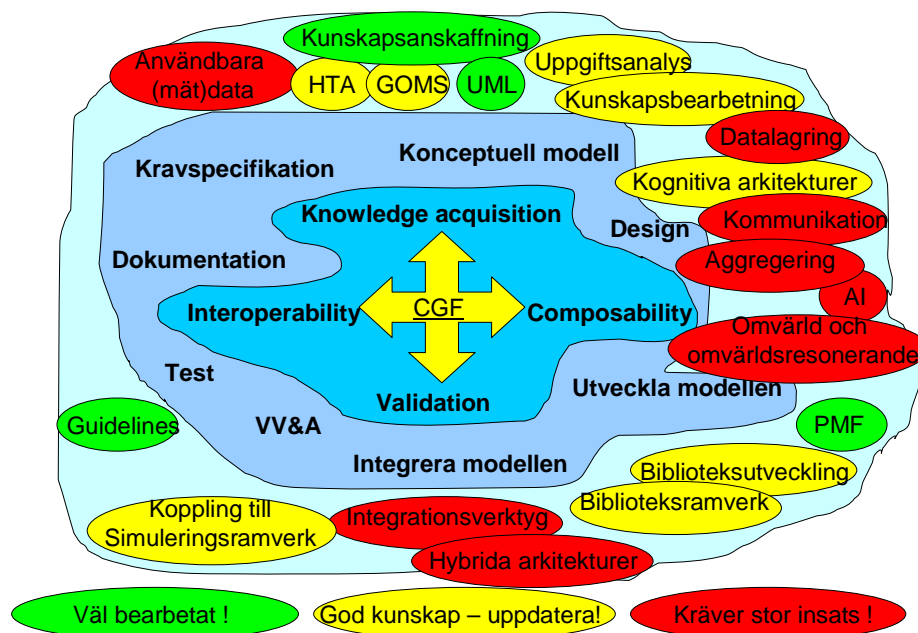
- Utveckla metoder för uppgiftsanalys. Att göra detaljerade beskrivningar av hur uppgifter och procedurer utförs och hur strukturer ser ut och används vid militära engagemang är grundläggande för att man skall kunna modellera datorgenererade styrkor som betar sig på ett trovärdigt sätt.
- Noggrant beskriva ändamål och möjligheter med de modeller som tas fram (som svar på en behovsanalys).
- Följ en väl utformad utvecklingsplan och satsa resurser på vissa fokuserade områden.

- Skapa arbetslag med nödvändiga kompetenser, t.ex. militära specialister, forskare som är experter inom kognitiv psykologi, social psykologi, sociologi, beteendevetenskap, datavetenskap och simuleringsteknologi.
- Utveckla prototyper.
- Uppmuntra återanvändning av modeller och funktioner. Med en tillräckligt genomtänkt design och arkitektur skall det gå att bygga simuleringsmodeller av grundläggande block och funktioner.
- Satsa tillräckliga resurser. Det är inte troligt att generella modeller kommer att kunna tas fram genom att snabbt komplettera någon eller några av de idag mest använda arkitekturerna.
- Det är inte troligt att *en* arkitektur är mest lämpad att använda för alla behov som finns. Olika arkitekturer har olika starka och svaga sidor. Det är därför troligt att flera arkitekturer måste användas eftersom kraven och behoven ofta är väldigt olika. Det har också prövats att kombinera flera arkitekturer och detta visar sig kunna fungera. Att satsa resurser på att utveckla en arkitektur som kan klara av alla krav är en ganska osäker investering förutom att man får räkna med en avsevärd tidsrymd för ett sådant arbete. Många av de existerande arkitekturerna har funnits 10-20 år och fortsätter att utvecklas.

### Stöd utveckling av teorier och grundläggande forskning inom nödvändiga områden

På grund av den särskilda karaktär som t.ex. beslutsfattande, situationsmedvetande, lärande och modellering av organisationer har i den militära världen så kan man inte förvänta sig att civil kunskap eller spelindustri skall föra utvecklingen särskilt mycket framåt inom dessa områden. Från (Wallin et al, 2003).

Ett sätt att beskriva några av elementen i ovanstående strategi har gjorts i figur 1 nedan. Den innehåller förenklat de tre lagren forskningsområde, utvecklingsprocess samt verktyg och metoder, och antyder relationer (närhet) dem emellan. Färgläggningen av metoder och verktyg visar kunskapsläget inom projektgruppen 2006 – 07.



Figur 1. Modell av CGF-utveckling, begreppssammanställning och kunskapsläge.

Under 2006 - 2008 har verksamheten framför allt innehållit:

- internationellt samarbete,
- fortsatt bearbetning av biblioteksstruktur,
- framtagning av teknik för att koppla modellarkitekturer till simuleringsramverk som STAGE, Merlin/Avalon och OneSAF,
- användardiskussioner med FLSC och deras operativa M&S-behov inom området.

Även behovet av icke-traditionell CGF-utveckling och användning för modellering av kulturella och sociala faktorerers inverkan har börjat belysas under 2008.

## 2.1 Nyttan med arbetet

Datorgenererade styrkor (CGF) är en term som används för att beskriva modeller och simuleringar som erbjuder representationer av militära styrkor, och har traditionellt använts för att befolka simuleringar med objekt och aktörer för att skapa en realistisk miljö.

Genom att använda datorgenererade styrkor kan man minimera kostnaden för organisation, ledning och dirigerings av enheter som behövs för en effektiv och meningsfull simulering. Det skulle ur försvarets synvinkel vara värdefullt att ha ett bibliotek av generiska styrkor som representerar olika militära och icke-militära enheter på olika aggregationsnivåer. Dessa skulle kunna användas som ”plug’n’play”-moduler, som via ett givet gränssnitt kan kopplas till de simuleringar där de behövs. Användningsområdena är många, t.ex. träning, utbildning och analys.

## 2.2 Definition av CGF, intresseområden

Begreppet CGF har under senare år ofta ersatts av begreppet HBR, Human Behavior Representation. På svenska används ofta datorgenererade aktörer eller syntetiska aktörer. På detta sätt breddar man definitionen av CGF och visar att denna teknik kan användas i många sammanhang där man inte primärt vill infoga en ”stridskraft”. En CGF kan utgöras av en entitet eller grupp som har ett automatiserat, förutbestämt eller förprogrammerat uppträdande. HBR å sin sida uppfattas som ägare av åtminstone några typiska människoliknande beteenden. HBR kan innehålla både de kognitiva funktionerna och såväl rörelseförmåga som rörelsemönster som visualiserade individer.

HBR har alltså många fler intressenter och många fler användningsområden. Psykologer vill kunna bygga modeller av hur människan fungerar i takt med att kunskapen ökar som förklaring och bekräftelse på psykologiska teorier.

M&S-samfundet vill kunna bygga och använda CGFer för att befolka simulerade världar. En datorgenererad aktör med kognitiva förmågor kan då vara särskilt värdefull.

## 2.3 Användning av CGFer

Användningsområdet för CGFer är mångfaldigt. Exempel på användning är:

- som intelligenta motståndare och medspelare, individer och grupper vid träning,
- för befolkning av den simulerade världen (för taktikutveckling),
- för SBA (simulation based acquisition) i designfrågor,
- som testpersoner vid test och analys,
- vid utformning av beslutsstödsystem,

- i inledande faser i taktikutveckling med enbart automatiserade styrkor,
- som funktion i taktikstöd till beslutsfattare i realtid.

CGFer kan också användas som automatiska operatörer t.ex. vid mångfallssimuleringar samt som automatiska operatörer inom många forskningsområden.

## 3 Resultat

### 3.1 Forskningsområdet

Rapporten (NATO RTO TR-47 /SAS-017, 2001) målar upp forskningsområdets utveckling och (åter-) användning av CGFer i fyra riktningar i form av:

- KA/KE , Knowledge Acquisition/Knowledge Engineering,
- composability,
- validation,
- interoperability.

Detta är fyra huvudområden inom M&S-processen och de är särskilt viktiga inom HBR. För varje del beskrivs en "roadmap" för att nå framgång inom HBR-utveckling och återanvändning. För att kunna få en snabbare, säkrare och mer effektiv utveckling förutses inriktning mot formalism, semantisk interoperabilitet, automatiserade processer och lagring av kunskap i "repositories", semantiskt sökbara databaser. Vid FOI pågår forskning inom alla dessa områden idag.

### 3.2 Utvecklingsprocessen

Metodikutveckling har bearbetats och beskrivits för utvecklingsprocessen: kunskapsinsamling, komponentbaserad utveckling, återanvändning, och för verifiering och validering.

I (Wallin et al, 2005) beskrivs på ett överskådligt sätt metodik för utveckling och användning av modeller av mänskligt beteende. Den här beskrivna metodiken är FEDEP-liknande och visar kopplingen till ovanstående forskningsområden.

### 3.3 Metoder och verktyg

Här följer kortfattat en genomgång av ramverk, metoder och verktyg som kan användas för skapande av CGF-modeller och kognitiva modeller och som vi har erfarenheter av i projektet.

#### 3.3.1 Datainsamling

Inom projektet har två ganska stora arbeten gjorts för datainsamling enligt beskrivning av KA/KE. Det första var en studie av Strid i bebyggelse vid en övning i Norrköping 2005. Det andra arbetet har genomförts vid FLSC, där ett antal flygförare djupintervjuades om sitt agerande i flygstridssituationer. Dessa arbeten finns rapporterade i (Nilsson et al, 2004) respektive Kunskapsanskaffning CGF 05 (ppt-presentation 23 maj 2005).

Utdrag från rapporten:

"Slutsats

Alla tre metoderna för kunskapsinhämtning (via dokumentation, via instruktioner och intervjuer samt via observation) upplevdes som viktiga för att kunna genomföra uppgiftsanalysen. En "after action review" med genomgång av situationerna med de övande och övningsledarna hade varit önskvärd. Utifrån kraven rekommenderar därför gruppen ordningen:

- 1) Studera dokumentation, manualer, etc (ger grundläggande kunskaper för gruppen)

- 2) Delta i övningsledarnas genomgångar
- 3) Observera övningarna, gärna tillsammans med övningsledare eller annan expert
- 4) Intervjua de övande tillsammans med expert i en "after action review" "

Och abstract från presentationen:

"CGFer (Computer Generated Forces) utgör med- och motspelare i simulerade miljöer. Den mest begränsande faktorn för utvecklingen av valida intelligenta agenter med mänskliga egenskaper är den formella beskrivningen av vad den verkliga operatören faktiskt gör. En orsak till att det är svårt att göra denna är att det ofta gäller att fånga och beskriva "tyst kunskap". Verktyg för att underlätta denna kunskapsinhämtning saknas i stor utsträckning och processen är mer en konst än en vetenskap. Projektet "Datorgenererade styrkor med mänskligt beteende" har under våren 2005 genomfört en sådan process vid FLSC (Flygvapnets Luftstrids Simulerings Center) som ett led i att utveckla agenter som är användbara i FLSC-miljön och som är mer intelligenta än dagens CGFer. Erfarenheter och rekommendationer från denna process kommer att presenteras."

### 3.3.2 Kunskapsbearbetning

Det vore önskvärt att kunna representera den insamlade informationen på ett enhetligt sätt, gärna i form av mallar som samtidigt kan beskrivas och förstås av både kunskapsinhämtarna och de som implementerar underlaget i simuleringsmodeller. Exempel på sådana mallar är bl.a. Hierarchical Task Analysis (HTA) och GOMS. På senare tid har också UML förts fram som en kandidat.

Erfarenheter av HTA och UML2 finns i projektet framför allt genom examensarbeten som genomförts framgångsfullt.

Uppgiften i ett av examensarbetena bestod i att analysera UML2, för att undersöka huruvida man kan använda dess diagram för att beskriva beteende och förhoppningsvis kunna svara på följande frågor.

- Är UML2 ett lämpligt verktyg för att beskriva beteende?
- För vilka arkitekturer (regelbaserade, ANN, Fuzzy Logic,...) är UML bra/mindre bra ?
- Vilka fördelar/nackdelar har UML jämfört med t.ex. HTA eller GOMS?
- Kan UML beskriva beteendemodifierare?

En detaljerad utvärdering genomfördes genom att utgå från material för hur en stridsvagnspluton utför en del av ett uppdrag, beskriva detta i UML och utifrån UML-diagrammet implementera beteendet i arkitekturen Soar för att slutligen visualisera med hjälp av lämplig grafikmotor, i detta fall Battlefield (Larsson, 2005).

### 3.3.3 Ramverk för kognitiv modellering

I (Castor et al, 2002) finns en förteckning av ramverk för kognitiv modellering, verktyg för CGF-modellering och ett antal ramverk, scenariogenereringssystem listade och beskrivna. Andra listor och beskrivningar av kognitiva ramverk finns i (Pew & Mavor, 1998) samt i (Gluck & Pew, 2005).

I (Castor et al, 2002) finns även en jämförelse av användbarhet och funktionalitet gjord mellan en av de mer populära kognitiva arkitekturerna, Soar och den svenskframtagna Agent Factory som är en produkt från Pitch Kunskapsutveckling.

### 3.3.4 PMF, Performance Modifier/Moderator Functions

Sättet att infoga effekter av yttre och inre faktorerers inverkan på den modellerade CGFen kallas ofta för PMF. Dessa funktioner påverkar de kognitiva förmågorna och är nödvändiga att infoga i modellen för att få fram t.ex. effekter av trötthet, känslor, hunger. Exempel på hur detta kan implementeras finns beskrivet i (Nilsson et al, 2004).

### 3.3.5 VV&A

Rapporterna (Wallin et al, 2005), (Nilsson & Yi, 2006), (Gluck & Pew, 2005) och NATO RTO HFM 2008 innehåller alla beskrivningar och några förslag till metodik för verifiering och validering av HBR-modeller. Detta är ett av de svåraste arbetsområden man kan ge sig in på. Scott Harmon i (Harmon et al) uttrycker sig t.o.m. som så att det är så svårt så det kanske inte är värt att över huvudtaget ge sig in på! Eller är det bara en myt? – Se tabell 1 nedan.

Myth	Explanation
Users are good sources of HBR requirements.	Users may understand the people in the situations they want simulated but they do not usually understand what about that human behavior their simulation can ignore. Thus, even if they appreciate the need for an HBR in their simulation (not a common occurrence), they tend to state that need in unrealistic terms.
A good referent for an HBR is a human doing the same job.	Identifying the corresponding human as a referent for its simulation may seem like a good idea but everything about that human is probably not well known. Human referents also tempt one to expect its abstraction to perform exactly like that human rather than its abstraction.
A valid HBR is as realistic (i.e., error = 0 in all property dimensions for all dependencies) as possible.	Again, HBRs are necessarily abstractions of real humans. We do not understand human nature well enough to accurately model all of human behavior. Like other simulations, HBRs should only represent what the purpose requires. Even if technically possible, a perfectly accurate HBR would be prohibitively expensive to develop and use.
A good HBR is stochastic just like humans.	Human behavior appears stochastic due to its high complexity and chaotic nature. Some aspects of human behavior lend themselves to stochastic representation but treating all of human behavior as random ignores the goal-directed nature inherent to all humans, a key property.
A good HBR is logical just like humans.	Humans seldom behave logically. They can, however, explain their behavior logically but that explanation rarely agrees with the real phenomena underlying their behavior.
“Fair Fight” is a clear and testable criterion for HBR validity.	The success of the fair fight criterion depends entirely upon the observer and, thus, cannot be objective. This will lead to irreconcilable differences between observers about whether the HBR actually met the criterion. Good validation criteria are both observable and observer-independent.



The experts will recognize invalid HBR behavior when they see it.	Experts may recognize some invalid behavior when they see it but the complex nature of human behavior will lead to many false positives. Experts have often declared quirky HBR behavior as distinctly “human” when it actually resulted from implementation errors.
Validating HBRs is too hard so why do it or even try to understand it.	This defeatist perspective only leads to accepting poorly performing HBRs. Like any validation task, a reasonably simple discipline can produce acceptable and cost effective results. Good understanding of HBR validity can even simplify the difficulty of abstracting the parts of human behavior necessary to achieve a purpose thereby reducing the developmental costs and risks.

Tabell 1. Common Myths about HBR Validation (Harmon et al).

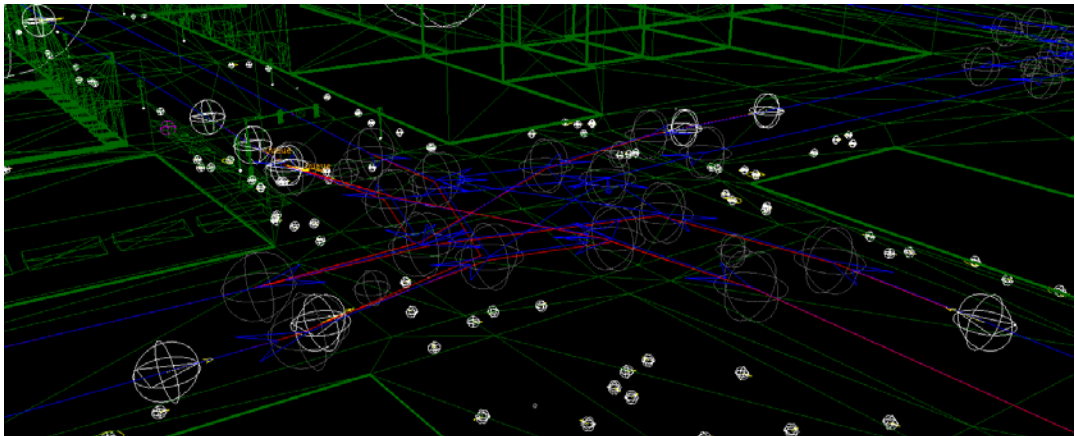
### 3.3.6 Guidelines

Rapporten (Wallin et al, 2005) erbjuder riktlinjer för konstruktion och implementering av CGFer. Rapporten från (NATO RTO HFM128) ger ytterligare kunskap om HF (Human Factors) och kognitionsvetenskapens betydelse och inverkan på CGF/HBR-utveckling. Projektet medverkade aktivt i denna grupp, vars arbete belönats med ett välrenommerat pris. Se vidare avsnitt 3.4.1 samt bilaga 3.

### 3.3.7 Utvärdering av verktyg: AI.Implant och STAGE

Området syntetiska aktörer har fått en allt större spridning, inte minst inom spelindustrin. Som ett led i att försöka ta del av den utvecklingen har projektet gjort en förnyad översyn av möjligheter till att använda kommersiella verktyg som plattform. De verktyg som i år varit föremål för utvärdering är *AI.Implant*, en programvarukomponent som används i vissa kommersiella spel, samt *STAGE*, ett simuleringsramverk som redan används vid FOI Båda tillhandahålls av företaget Presagis.

#### *AI.Implant*



Figur 2. Virtuellt trafikerad korsning. Linjerna representerar navigationsbanor och hinder.

*AI.Implant* är en programvarukomponent som tillhandahåller artificiell intelligens för att befolka datorspel och simuleringar med syntetiska aktörer. I korthet så möjliggör *AI.Implant* för de simulerade aktörerna att ta kontextberoende beslut samt röra sig på ett sätt som uppfattas realistiskt av användaren.

## **STAGE**

*STAGE* är ett simuleringsverktyg avsett för träning och analys. Verktöget används vid FLSC, som är en av CGF-projektets intressenter och avnämare. I syfte att underlätta resultatöverföringen har projektet utvärderat verktöget.

*AI.Implant* kan också användas integrerat i *STAGE*. En sådan konstellation av verktyg skulle på sikt vara av stort intresse att använda som försöksplattform för fortsatt forskning och prototyputveckling även för andra intressenter.

### **3.3.8 Ramverk för simulering**

#### **3.3.8.1 ModSAF**

Uppkomsten av CGF- och SAF (Semi-Automated Forces)-simuleringar skedde genom DARPA:s (Defence Advanced Research Projects Agency) försök med SIMNET-programmet under den senare hälften av 80-talet. SIMNET bestod av en serie stridsvagnsimulatorer som var sammankopplade via nätverk och som använde ett gemensamt kommunikationsprotokoll (SIMNET-protokollet) för att dela på statusinformation. I början av projektet räckte det med de sammankopplade simulatorerna för att träna små grupper. Det blev dock snart klart att för att simulera större taktiska övningar med tillräckligt många aktörer behövdes en mer skalbar metod än att enbart koppla till fler simulatorer. För att tillgodose detta behov utvecklade man datorprogram som möjliggjorde att mänskliga simulatoroperatörer kunde skapa ett antal simulerade objekt, som t.ex. stridsvagnar eller flygplan, och kontrollera deras beteende.

Sedan dess har utvecklingen inom området fortsatt och CGF-programmen ökat i antal och komplexitet. Det mest kända exemplet är förmodligen ModSAF (Modular Semi-Automated Forces) som under 90-talet växte från att vara ett enkelt system för att försörja simuleringar med markbaserade plattformar till ett komplext simuleringssystem med förmåga att simulera stora mängder av mark-, luft- och sjöplattformar, samt andra omgivningsobjekt. ModSAF har en relativt enkel kunskapsrepresentation, och simuleringarna blir inte tillräckligt detaljerade och realistiska. Inom det amerikanska försvaret har trenden gått mot kognitiv modellering och agentteknologi när man skall utveckla CGF-tillämpningar.

#### **3.3.8.2 OneSAF**

Utvecklingen av OneSAF kan benämnas som ett lyckat projekt med i storleksordningen ca etthundra utvecklare, inklusive konsulter som SAIC och MITRE. Målet har varit att skapa nästa generations CGF-verktyg som ett gemensamt ramverk för U.S.Army's alla M&S-domäner: ACR (Advanced Concepts and Research), TEMO (Training, Exercises, and Military Operations), och RDA (Researches, Development and Acquisition). Därmed ersätter OneSAF tidigare Semi-Automated Forces som ModSAF, CCTT SAF och JANUS. En beskrivning på svenska av OneSAF finns i (Pelo et al., 2007).

### **3.3.9 Prototyping**

Under verksamhetsåren har enkla modeller, funktionsmodeller, prototyper tagits fram i arbetet för att lära oss, testa och visualisera funktioner som är användbara i CGFer.

Nedan följer ett antal exempel på olika typer av prototyper som FOI arbetat med.

- Arkitekturfunktionstest, kognitiva, PMF: (Castor et al., 2002), (Wallin et al., 2003)
- Motion capture (Nilsson et al., 2004), (Lundin, 2004)
- Implementation/integration av data/system från KA/KE, UML: (Larsson, 2005), (Kunskapsanskaffning CGF 2005)

- Modellimplementationer (Pelo et al., 2007)

### 3.3.10 Hifi Engine

Hifi Engine är en testbädd som erbjuder implementering, test och visualisering av CGFer och som projektet använt för att demonstrera prototyper.

Mer beskrivet om Hifi Engine finns i bilaga 4

### 3.3.11 Modellbibliotek/biblioteksstruktur/arkitektur

En stor del av projektiden har använts för att enligt de senaste kunskaperna om komponentbaserad utveckling utveckla och beskriva ett förslag till hur man ska bygga modellbibliotek för att lagring, användning och återanvändning av modeller och komponenter av modeller skall kunna ske på bästa sätt. Sammanfattande resultat av detta arbete finns i bilaga om Athena (se bilaga 5) och kan också hämtas från FOI CGF-årsrapport 2007 (Pelo et al, 2007) i kapitlet om virtuell rotekamrat där det senaste inom projektet om modellbibliotek finns beskrivet. Detta bibliotek är idag sammankopplat med MERLIN så att beteendekomponenterna kan sättas samman och exekveras på samma sätt som flyg- eller robotmodeller. Projektet har drivits som ett forskningsprojekt utan några direkta krav från användare.

#### 3.3.11.1 FLSC-användare

FLSC har ett behov av att befolka sina miljöer med olika typer av datorgenererade styrkor. De datorgenererade styrkor som man hittills använt har främst utvecklats i STAGE. FLSC ser gärna att denna utveckling drivs vidare för att förbättra CGF-funktionaliteten och då även utöka simulatoranläggningens möjligheter. Inom FLSC har man även en god bild av vilka krav som ska ställas på CGFer inom anläggningen och dess verksamhet.

Några konkreta exempel på behov av ny funktionalitet som diskuterats anges nedan.

1. Generering av många förband/"omvärldsbrus" (liknande dagens STAGE med navigeringar och beteende kopplat till brytpunkter för luft-, mark- och sjö mål, samt automatisk följning av vägar för fordon och personer).
2. Entiteter som skall interagera mot bemannade kabiner, t.ex. attackflygplan som gör undanmanöver om de får robotskott mot sig, människor som flyr om det släpps en bomb i området.
3. Smarta motståndare, kanske framförallt jaktflygplan och attackflygplan/UAV som kan följa en förplanerad navigering, men efter speciella trigger-event flyga mer taktiskt efter någon typ av egen AI/riskbedömning/erfarenhetsbank.
4. Kunna populera omvärld/målområden med "övriga entiteter" som inte har någon annan funktion än att skapa andra objekt än med- och motspelare. Exempelvis folksamlingar, fordon, som på ett eller annat sätt får ett antal vägar, stigar eller ett geografiskt område att röra sig fritt i, gärna med slumpade parametrar så att om man kör samma scenario flera gånger får olika utseende på "bruset". Detta spar på sikt mycket tid, då man slipper bygga ett antal variationer kring samma tema.

Under hösten 2008 har projektet påbörjat en aktivitet för att möta behovet under punkt 3 ovan, d.v.s. att skapa CGFer för jaktstrid med långräckviddiga robotar, s.k. BVR-strid (Beyond Visual Range). Aktiviteten är motiverad av FLSCs behov av automatiska motståndare när man tränar uppdrag där det inom området finns många flygplan från båda sidor.

Ett utkast till konceptuell algoritm har tagits fram där man använder en s.k. *model predictive control* (MPC)-ansats för att styra CGFen tillsammans med en datastruktur innehållande ett antal simulerade hot-robot-hypoteser. MPC innebär att man hela tiden

simulerar utfallet av ett antal styrkommandon och sedan väljer den bästa av dessa. Detta förfarande upprepas hela tiden i styralgoritmen. De simulerade hot-robot-hypoteserna används för att utvärdera olika styralternativ, vilket möjliggör att CGFen tillgodogörs den kunskap som finns inbyggd i jaktrobotarna som finns tillgängliga i Merlin-arkitekturen.

## 3.4 Internationellt samarbete

### 3.4.1 NATO HFM 128

Som kort beskrivits i tidigare årsrapporter har projektet deltagit i NATO RTO HFM 128.

Arbetsgruppens resultat finns samlat i en slutrapport NATO HFM128-Report, "Human Behavior Representation in Constructive Simulation". Slutmötet i maj 2007 var en workshop, NATO RTO HFM 143 Specialist Meeting on Human Behavior Representation in constructive modeling . Sedan dess har sammanställning av arbetet gjorts och Executive Summary finns i bilaga, hela rapporten finns i sin nuvarande form med på CGF-samlings-CD.

Gruppens arbete har tilldelats pris av NATO RTO Board för utomordentliga vetenskapliga insatser till nytta för NATO och dess medlemsstater, se vidare bilaga 3, "Citation for the 2008 RTO Scientific Achievement Award".

### 3.4.2 AFRL-FLSC

Ett 6-årigt samarbete startades 2005 mellan FLSC, FOI och AFRL (Air Force Research Laboratory) i MESA för utbyte, utveckling och samarbete om International Mission Training Research (IMTR) och Distributed Mission Training (DMT). Ett av verksamhetsområdena var Kognitiv Modellering med avsikt att ta fram och använda modeller av syntetiska aktörer. Ett informationsutbyte genomfördes inom det område som CGF-projektet arbetat med under sin projekttid. Sedan 2007 har dock fortsatt samarbete gått i stå, delvis beroende på oklar finansiering, delvis annorlunda inriktning av arbetet främst hos gruppen vid AFRL.

## 3.5 Närliggande användnings- och utvecklingsområden

Projektet har identifierat en intresse- och trendförskjutning från de användningsområden som kunde skönjas under 2001, till närliggande områden som påvisar i vissa stycken annorlunda och nya behov av syntetiska aktörer. Det finns i huvudsak tre grunder för detta:

- Mognaden hos många CGF-ramverk, t.ex. OneSAF och STAGE är nu sådan att den inkluderar användning av tillräckligt bra tillämpningar inom CGF eller tillåter genom öppen arkitektur att anpassa eller länka till modeller som fyller mer krävande behov.
- Svenska försvarets inriktning att i allt högre grad delta i internationella fredsbevarande insatser kräver kunskap om betydelsen av kulturella och sociala skillnader och effekter. Dessa är i dag mycket viktiga att kunna påvisa i utbildning och träning, både vad gäller inverkan på samarbete såväl som vilka effekter som det har i fråga om kontakt med befolkningen i ett insatsområde.
- Svenska försvarsmaktens deltagande i militära internationella konflikter kräver likaså en ny förståelse för kulturella skillnader och dessas effekter i konfliktområdet.

Denna förskjutning för med sig konsekvenser för forskningen inom HBR. Behovet av användning av syntetiska aktörer i större skala, och för skarpa frågeställningar har vuxit kraftigt. Produktionen av syntetiska aktörer för att möta dessa behov har inte ökat i takt med behoven. Den kunskap som finns om arkitekturer och datalogisk utformning av ramverk och bibliotek för syntetiska aktörer behöver kompletteras med insikt om hur produktionsprocessen ska utformas för att möta dessa behov.

Kunskap och förståelse för betydelsen av sociala och kulturella faktorer är av direkt intresse för de FM styrkor som skall verka i ett område. Den som ämnar beskriva dessa faktorer i en simuleringsmodell behöver en teoretisk grund för att beskriva hur dessa faktorer påverkar beteende. För att modellen skall vara användbar för FM vid träning, utbildning samt efteranalys, krävs stora forskningsinsatser för att finna en vetenskaplig grund som modeller för militära ändamål kan baseras på.

(Lundin, 2008) påvisar dessa behov ytterligare och är en genomgång och sammanfattning av kunskap inom området idag.

## 4 Diskussion

Sedan forskningen kring detta område initierades har Försvarmaktens förutsättningar förändrats radikalt. M&S av storskaliga militära operationer utgör inte längre det enda centrala tillämpningsområdet. Vidare finns en större konkurrens om en minskande mängd tillgängliga medel för forskning och utveckling. Har då Försvarmakten fortfarande samma behov av datorgenererade styrkor? Från forskningsperspektivet vill vi påstå att behovet snarare är större idag än när det första forskningsprojektet initierades 2001.

Nyttan för Försvarmakten med syntetiska aktörer ligger främst i dess användning, vid simulatorer och träningsanläggningar, bland annat för att åstadkomma:

- förbättrad kvalitet på träningen genom ökad realism,
- ökad kapacitet genom minskat behov av mänskliga aktörer.

Därmed är det också viktigt att belysa forskningens roll i värdekedjan för befolkning av simulerade världar. Idag drivs den utvecklingen inom området av spelindustrin. Användning av syntetiska aktörer för militära ändamål ställer speciella krav på utformning, användning och validitet för dessa modeller. Även om det kan finnas fall där kommersiella spelprodukter kan användas utan modifieringar, så kräver merparten av de tillämnningar som projektet studerat att beteenden kan skraddarsys för ändamålen. Således behöver kunskapen konkretiseras och förpackas i form av prototyper för att komma FM tillgodo.

Även om användningen av syntetiska aktörer kan vara till stor nytta för FM, så är framtagningen förknippad med kostnader. I jämförelse med andra utvecklingsprojekt och kontinuerlig vidareutveckling av simulatoranläggningar kan en satsning på syntetiska aktörer lätt framstå som ett högriskprojekt. För att åstadkomma en ökad användning av syntetiska aktörer inom FM simulatoranläggningar behövs sannolikt en strategi som:

- kan motivera den högre risken,
- är förknippad med resurser för utveckling.

Därmed inte sagt att området Human Behavior Representation inte behöver fortsatt forskning. Krasst sett är merparten av dagens militära operationer förorsakade av mänskligt beteende. I många fall utgörs dessutom den förväntade effekten av dessa operationer av att förändra beteenden och uppfattningar. Forskning kring HBR lär oss inte bara nya sätt att representera mänskligt beteende; den kan även ge unika insikter om detta område utifrån ett militärt perspektiv.



## 5 Förkortningar

AI	Artificiell intelligens
ANN	Artificiella Neurala Nätverk
BVR	Beyond visual range
CGF	Computer Generated Forces
DMSO	Defense Modeling and Simulation Office
FEDEP	Federation Development and Execute Process
FLSC	Flygvapnets Luftstrids Simulerings Center
FM	Försvarmakten
GOMS	Goals, Operators, Methods and Selection Rules
HBR	Human Behaviour Representation
HTA	Hierarchical Task Analysis
KA/KE	Knowledge Acquisition/Knowledge Engineering
M&S	Modellering och simulering
ModSAF	Modular Semi-Automated Forces
PMF	Performance Modifier Functions
SAF	Semi-Automated Forces
SBA	simulation based acquisition
UML	Unified Markup Language
VV&A	Verification, Validation & Accreditation





## 6 Litteraturförteckning

M. Castor, Niklas Wallin, S. Å. Nilsson, F. Moradi. Datorgenererade styrkor – metoder och möjligheter, December 2002. FOI-R--0688--SE.

K. Gluck and R. Pew. *Modeling Human Behavior with Integrated Cognitive Architectures*, 2005.

S. Y. Harmon, C. W. D. Hoffman, A. J. Gonzalez., R Knauf., V.B. Barr, Validation of Human Behavior Representations.

J. Larsson, Final Thesis, The Unified Modeling Language – Modeling Computer Generated Forces. Lith-IDA-Ex—05/78—SE, 2005.

M. Lundin, Simulating the effects of mental workload on performance. LIU-KOGVET-D—04/xx—SE, Linköping Electronic Press, Sweden, 2004.

M. Lundin, Sociocultural Models in Computer Generated Forces. FOI-MEMO 2510, 2008.

NATO RTO-TR-HFM-128, Human Behavior Representation in Constructive Simulation, 2008. Preliminary version.

NATO RTO-TR-047/SAS.017, Human Behaviour Representation, 2001.

S. Å. Nilsson, H. Allberg, M. Lundin, N. Wallin, and M. Castor. Datorgenererade styrkor med mänskligt beteende, 2004. FOI-R--1489--SE.

S. Å. Nilsson, C. Yi. VV&A of CGF: A REVVA Application?, 2006. FOI-R--2117--SE.

J. Pelo, M. Lundin and N. Wallin. Årsrapport 2007 CGF, 2007. FOI-R—2400—SE.

R., Pew, & A.Mavor, 1998. *Modeling Human and Organizational Behavior: Application to Military Simulations*. National Research Council.

N. Wallin, M. Lundin, S. Å. Nilsson, and H Hasewinkel. Datorgenererade styrkor, December 2003. FOI-R--1076--SE.

N. Wallin, S. Å. Nilsson, M. Lundin, and M. Castor. Guidelines för utveckling och användning av modeller av mänskligt beteende, December 2005. FOI-R--1837--SE.

Kunskapsanskaffning CGF 2005. PowerPoint-presentation från M&S-presentationsdag 23 maj 2005.



## 7 Bilaga 1, Reserapport 17th BRIMS

Reserapport 17th BRIMS & Spring SIW 2008

### **Inledning**

17th Conference on Behavior Representation in Modeling and Simulation (BRIMS) och Spring 2008 Simulation Interoperability Workshop (SIW), i år i Providence, Rhode Island, var för andra året i rad samlokaliserade. Konferenserna är fortsättningsvis, tillsammans med IITSEC, några av de mest aktuella för FOIs forskningsområde syntetiska aktörer och M&S i allmänhet. Enligt huvudsponsorn SISOs styrgrupper har deltagarantalet på båda konferenserna sjunkit med hälften sedan 5 år tillbaka, men BRIMS har åtminstone fler icke-accepterade bidrag än förra året. Dessutom tror man på en ökning, åtminstone inom området social- och kulturell simulering, till 2009. Överlag är bidragen till båda konferenserna mycket varierande både i kvalitet och ämnesinriktning.

### **Tematiska områden**

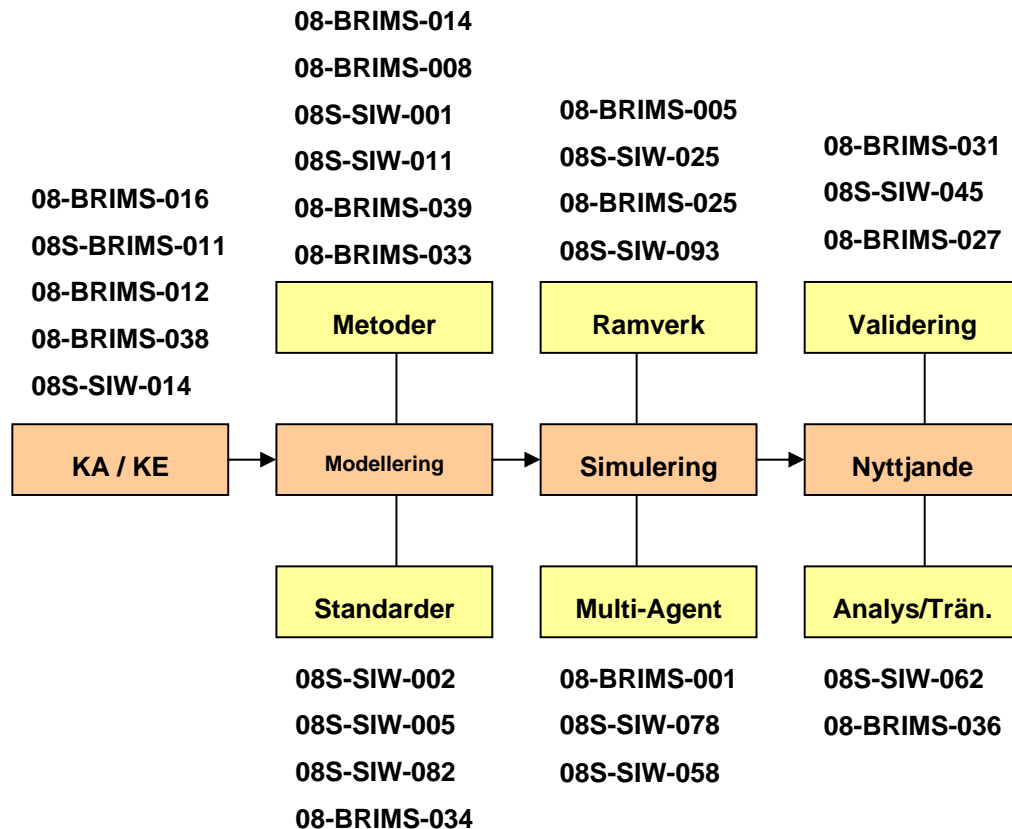
BRIMS konferensbidrag sträcker sig från detaljerade och empiriskt validerade kognitiva modeller till stridssimuleringar på taktisk nivå, tillsammans med enstaka nya ramverk och "concept papers" där begrepp och behov klagörs. I traditionell anda förekom paneldebatter där bland annat modeller av samma fenomen jämfördes (08-BRIMS-031), samt gemensamma standarder för omvärldar diskuterades (08-BRIMS-034). Antalet bidrag var tillräckligt litet för att parallella sessioner inte skulle behövas.

SIW har ett tydligare fokus på simulering och färre bidrag med en tydlig modelleringsansats. Med fem parallella sessioner kunde dock inte allt bevakas utan en övergripande orientering gjordes. Generellt handlar bidragen om följande.

- Gemensamma standarder och språk. Två exempel är *Common Battle Management Language (C-BML)* (08S-SIW-082) och *Military Scenario Definition Language (MSDL)* (08S-SIW-002).
- Distribuerade simuleringar, HLA och relaterade tjänster (t.ex. datatrafikövervakning). Pitch AB visade bl.a. hur Google Earth kunde användas för att visualisera och testa HLA-federationer och scenarier (08S-SIW-093).
- Nya tekniker för simulering och ramverk, till exempel (08S-SIW-025) där en ny approach till simulering ökar datorkraften med upp till 15000 gånger för simuleringar av Monte Carlo-typ.
- Modelleringsprojekt av militära operationer och materiel. Till exempel (08S-SIW-011) där olika resolutioner av en projektilmodell jämförs, eller (08S-SIW-014) där Marine Corps simuleras i tre olika operationer med IWARS.
- Validering. En hel session var ämnad åt standardisering och generell metodik för VV&A. Bland annat representerades FOI av Constantinos Giannoulis som presenterade resultat från REVVA-gruppen (08S-SIW-048).

### Relevans för verksamhetsområden

Följande schema beskriver hur de mest bevakade bidragen kan vara relevanta för verksamheter inom FOI och syntetiska aktörer. Av dessa beskrivs sedan de tio mest intressanta och/eller relevanta bidragen. KA (Knowledge Acquisition) och KE (Knowledge Engineering) diskuterades inte i högre grad som temaområde, men olika metoder och typer av källmaterial återfinns i de presenterade modelleringsansatserna.



08-BRIMS-001

### Visual Agent-Based Model Development with Repast Symphony

En endagskurs som var av intresse av två anledningar: Först nyheterna i den senaste versionen av Repast: Förbättringar sedan tidigare version (Repast J) är bland annat integration med analysverktyg som MATLAB, SPSS, tekniker som GIS, men framförallt nya begrepp för att beskriva systemet agenterna verkar i: *agent*, *context*, *projection-space*, *projection-network*. Den andra nyttan var att prova visuell modellering som metod för att snabbt bygga logiken i en agent. En plugin till Eclipse IDE användes för att bygga agenter med hjälp av ett GUI. Detta bestod av symboler för programkodsnära begrepp som *for/while*, *property*, och agentnära begrepp som *decision*, *task*, och *behavior*. Det fanns även en trädstruktur, *model score*, för att specificera tidigare nämnda systembegrepp. Ambitionen med Symphony är att flytta mycket av det som tidigare krävde lusläsning av APIer, till GUI och wizards.

08S-SIW-058

**Towards a Lexicon for the Modeling and Simulation of Emergent Behavior Systems**

En mycket grundlig och noggrann studie av begreppet *emergens*, där en historisk genomgång av artiklar inom biologi, fysik, ekonomi, och sociologi ligger till grund för den matematiskt formaliserade ontologi som är målet med studien. Med *emergens* menas att ett system får nya egenskaper vid en sammansättning av många systemelement: En tegelsten har egenskapen *vikt* och ett hus av tegelsten har summan av varje tegelstens vikt, men beroende på olika *strukturella relationer* (arkitekturer) uppstår även emergenta egenskaper som gör att olika hus skiljer sig. Författaren tar även upp en historia av multi-agent system (MAS) och hur *emergens* blivit relevant inom dessa, samt hur ontologin som utvecklats passar in i MAS. Dessa formalismer kan vara mycket användbara vid utvecklingen av sociala modeller och beskrivningen av dessa.

08S-SIW-025

**Simulating Parallel Overlapping Universes in the Fifth Dimension with HyperWarpSpeed Implemented in the WarpIV Kernel**

Bakom denna näst intill "oseriösa" titel kan det potentiellt mest banbrytande SIW-bidraget finnas, genom en ny och smart användning av parallella processer. De tre spatiala dimensionerna samt tidsdimensionen är grunden för de flesta simuleringarna. De största simuleringarna är ofta av Monte Carlo typ, d.v.s. ett mycket stort antal processer med slumpmässiga ingångsvärden testas för att se hur en eller flera variabler påverkas. *Fifth Dimension* är en kloning av simuleringen där mekanismer som ändrar världens tillstånd genomgår en *split* och återgår de senare, genomgår de en *merge*. På detta sätt behövs inte alla processer köras som i Monte Carlo-fallen, utan simuleringen påminner mer om en smart breddenförstökning, där varje ny nod i sökrymden med samma värde smälter samman. Simuleringen kan vidare enkelt köras framåt och bakåt. Ansatsen är i sin linda, men jämfört med Monte Carlo metoden går simuleringarna 236 gånger snabbare.

08S-SIW-011

**Experiences from a Multi-Resolution Modeling case study using Model Driven Architecture**

Ett bidrag som tydliggör vad vi menar med *upplösning* i en simuleringsmodell, och en föreslagen lösning till hur ett modellbibliotek med modeller med olika upplösningar effektivt skulle kunna skapas. Grunden är *model-driven architecture* (MDA), en modelleringsansats som nu är några år gammal. Poängen är att man separerar de funktionella specifikationerna och implementationerna av dessa. Detta för att uppnå återanvändning, interoperabilitet och portabilitet. Exemplet som tas upp är en vapenmodell som kan beskrivas i ett antal upplösningar (där en finare upplösning innefattar fler variabler och komplexare beräkningar). Genom att utnyttja MDAs metabeskrivningar kommer man i framtiden kunna utveckla system där man snabbt kan välja rätt grad av upplösning på sina simuleringar.

08S-SIW-001

**Simulating Decision Making where Cooperative Agents Use a Hybrid Reasoning Framework**

Ett mångårigt modelleringsprojekt (med avtappningar till SIW) där RePast representerar agenter i en GIS-baserad omvärld. Simuleringen består av agenter vilka representerar olika roller i ett högkvarter: Commander, Air Defence Commander, Information Warfare

Commander, Sea Combat Commander, och Strike Commander. Dessa utvecklar CONOPS (CNOcept of OPerationS) och utvärderar riskerna för dessa, och utifrån vald CONOP skapas sedan ett uppdrag. Agenterna nyttjar andra ramverk för att lösa operativa problem med hjälp av ett flertal kända kunskapsmetoder: *Casebased Reasoning*: Används för att hitta en liknande CONOP utifrån ett bibliotek. Ramverk: NaCoDAE. *Bayesianska nät*: Väljer parametrar som mest sannolikt reducerar risken i krigszonen. Ramverk: JavaBayes. *Neurala nät*: Används för att utvärdera föreslagna formeringar av förband. Ramverk: JOONE. Dessutom används ett (icke-beskrivet) *expertsystem med fuzzy logic* för att kontrollera handlingar och val av ovanstående teknologier.

08-BRIMS-016

### **Integration of Military Occupational Characteristics into Human Performance Models**

En kanadensisk ansats att skapa prestationsmodeller i IPME utifrån yrkesbeskrivningar. Man vill komplettera de befintliga operatörsmodellerna (fysiska egenskaper, antropometri, läggning, tillstånd) med information från yrkesrollspecifikationer som: Jobbklassifikation, funktionell arbetsbeskrivning, ansvar, krav, gradbeteckning, arbetstidslängd, samt kunskaper och färdigheter. Det handlar alltså om kunskapsområden från AFRL-FOI-samarbetets Mission Essential Competencies workshops. En plug-in till IPME med yrkesklassifikationer ger möjligheten att påverka uppgiftsnätverken (till exempel förkorta tiden en uppgift tar givet att en färdighet finns) och operatörsmodellerna.

08-BRIMS-024

### **PSTK: A Toolkit for Modeling Dynamic Power Structures**

Soartechs bidrag till trenden DIME/PMESII/EBAO/CA inom C2. PSTK har utgått från en analys av begreppet *makt*. Attribut till makt som instrumentalitet, relativitet, multipla källor, kvantifierbarhet, och universalitet har ingått i designen av detta ramverk. Makt ses som en skalär av *kapital \* förmåga*, där kapital går att lagra medan förmåga kan gälla endast för ögonblicket. PSTK bildar ett sorts spel där spelarna (agenterna) i varje drag försöker påverka varandra, antingen positivt eller negativt, genom att påverka varandras makt. Detta sker genom en förändring, transaktion, av någon av följande typer av makt (som även diskuteras): *politisk, militär, ekonomisk, social/ideologisk*. Agenterna agerar utifrån *lines of influence* (LOI) där en agent påverkar en annan agents maktattribut genom en tredje agent. PSTK sponsras av US DoD och har testats av underrättelseanalytiker med upp till 148 separata agenter, 1023 LOI, 900 mål, och 1164 kontexter.

08-BRIMS-033

### **A General Framework for Cross-cultural Interactions in Virtual Environments**

Bidraget beskriver en nystartad ansats till en generell beskrivning av kultur, inom ramen för virtuell interaktion. Inledningsvis konstateras att Hofstede's variabler är svårtolkade för överföring till specifika situationer. Istället har man fokuserat på vilka variabler som har relevans för kulturell interaktion, individer emellan: Fysisk karaktäristik, social karaktäristik, språk och kommunikativ förmåga. Man går än mer in i detalj för att beskriva uttal av språk, tonläge, poser, kroppskontakt, ansiktsuttryck, blick, samt känslomässiga uttryck. En ontologi (MACBETH: Measuring Accurate Behavior for Effective Training of Humans) har utvecklats för att tydliggöra de kulturella fenomenen (t.ex. institutioner, värderingar, organisationer, normer, artefakter, nationer), men ett komplett träningsramverk är ännu inte utvecklat.

08-BRIMS-039

### **A Language for Modeling Cultural Norms, Biases and Stereotypes for Human Behavior Models**

Fokuset för denna forskningsansats är att definiera ett ”språk” för tolkning av kulturell kunskap. Språket ska kunna användas av en etnolog för att ”spela in” etnografisk data för användning i syntetiska aktörer. Vidare fokuserar man på kulturella normer och mentala modeller för kulturella stereotyper. Här baserar man sig på *schemateori* och framförallt D’Andrade (1992)-teori om att kulturellt betingat resonering kan ses grundat i en uppsättning kulturella scheman. En viktig aspekt som kommer fram är att det inte enbart handlar om att följa en uppsättning regler, utan det finns även en motivation att uppfattas av andra som regelföljande (ett exempel som tas upp är synen på alkohol inom Islam). För att generera de kulturella beteendena skapas nätverk av scheman, där olika handlingar har vikter till olika regler (att erbjuda en muslim alkohol triggar både en viktad länk till dennes kulturella schema och dennes uppfattning om personen som erbjuder alkohol). En prototyp med stereotypa drag för tyskar och irakiska sunni-muslimer har utvecklats.

08-BRIMS-008

### **Modeling the Human Decision Making Process in Maritime Interdiction Using Conceptual Blending Theory**

Ett av två BRIMS-bidrag från Singapore. Framförallt viktigt ur teoretisk synvinkel då det ger ett exempel på tillämpning av *Conceptual Blending Theory* (CBT) (se Fauconnier & Turner, 1993-). CBT är liksom Soar en *Unified Theory of Cognition* och beskriver en fundamental men undermedveten process för hur ny information integreras med tidigare och eventuellt ger inläring och kunskapsbyggande i vardagligt tänkande och språk. Grundelementen *membrane*, *data ticket*, *procedural ticket*, *connector*, och *frame* har biologiska motsvarigheter i hjärnans synapser. I detta bidrag modelleras OODA-loopen (Observe, Orient, Decide, Act) hos en hamnkontrollant med CBT som grund. Hamnkontrollantens problem är att bedöma hotbilden från okända fartyg, utifrån potentiella händelseförlopp (självordsbombning, bordning, vapenattack på kort avstånd, självmordskollision, raketattack, och användning av lättantändligt bränsle). Olika händelseförlopp representeras i ett *blending network*, där en *data ticket* kan ha attribut från text ett okänt fartyg. Ansatsen syftar i det långa loppet till att skapa ett beslutsstödssystem för hamnkontrollanter utifrån den modellerade beslutsprocessen baserad på CBT.





## 8 Bilaga 2, NATO RTO HFM-128

# Human Behavior Representation in Constructive Simulation

### Executive summary

#### Human factors (HF) are underrepresented in military modeling

In current military operational models, the human aspect is still often represented in a mechanistic way, bearing little resemblance to observations, as if all humans always act the same way in a situation much as a machine would. In reality, human behavior is not deterministic. Human behavior depends upon a complex set of human traits and states interacting with an equally complex set of environmental variables. Behavior is not a random effect, but neither is it deterministic. Behavior is the consequence of situational impact on the human pursuing goals moderated by personal traits and states in a partially observable environment. Without proper representation of behavior, and the reasons behind the behavior, the validity of the model may be seriously flawed, making its performance and predictions questionable.

Although considerable progress has been made over the last several decades in modeling of cognitive processes, these models and modeling architectures have not been designed to solve operational problems. Conversely, operational architectures do not cope well with cognition and seldom reflect more than rudimentary human physical capabilities. The conceptual model for representation of the actions of personnel in military simulations that the HFM 128 study group is striving toward would have balanced capabilities for both the HF and operational aspects, while allowing for efficient and valid studies.

#### Including HF

HF that are relevant to operations have been identified in previous NATO reports and in books on human behavior representation with weaknesses or shortfalls pointed out in existing military simulations. In general, the knowledge of HF is more advanced than the implementation in these models. The difficulty that model developers face is how to incorporate HF knowledge without creating models of such complexity that they become unusable. This report provides a number of suggestions to limit the complexity, while using solid HF knowledge. A key recommendation is to represent human states as explicit variables that affect performance in a transparent way. Human states can be measured and provide a basis for expressing the stresses experienced by personnel that are thought to be important factors to military outcomes (exhaustion, thermal load, mental workload, etc.). It is time for the operational military modeling community to begin incorporating advances from HF and cognitive science into Human Behavior Representations (HBRs) for constructive simulation.

#### Operational models with integrated behavior

Effects based operations have a profound impact on the way an operational problem is solved and consequently also on the requirements placed on simulation models. The operational task is to accomplish goals rather than to perform a prescribed sequence of tasks (characteristic of SOPs, leaving the choice of how to accomplish the goal to the

assigned unit. In effects based operations, the assigned unit needs more information about the global context, is in the best position to evaluate the local context to generate options, and needs the decision authority to enact those options that seem favorable. Modeling this approach to operations requires increased representation of cognition, in which coordinated units must have their own representation of cognitive processes that capture assessment, judgment, and decision making. The choices allow for variable behavior, but the effect of various stressors experienced by the unit may also need to be modeled, further affecting behavior and performance. Generation of behaviour and automated understanding of that behavior by other entities is an additional challenge to the modeler. This report attempts to provide some guidance on how human behavior models can be extended to capture these effects in military modeling and simulation.

The validation of models remains a concern. Although numerous models of operators or platforms have been created for military simulations, few have been rigorously verified and validated beyond the physics and SOP characterization of behavior. The development of HF sub-models based on the scientific literature can deliver validity at the sub-model level. However, combining those sub-models in an operational model requires verification of the sub-models as well as the integrated ensemble. Substantial funding should be allocated for this stage of the development, particularly if data must be collected. It is important that the integrated model be tested with use cases to ensure validity, and we strongly recommend describing the validation method used when presenting the simulation results so that the audience can make an informed judgment about the suitability of the conclusions.

Special considerations are necessary when aggregated units are used as entities in a constructive simulation. Teams and larger units have additional HF properties that do not exist at the individual level (such as cohesiveness and mutual support). On the other hand, some individual characteristics become insignificant to the team level of performance (particularly individual skills such as marksmanship) for statistical reasons although some remain critical (particularly those that affect the team as a whole, such as fatigue). It is recommended that the simulation be scaled by aggregating elements where necessary and incorporating the associated HF appropriate to that aggregation.

### **Discussion of recent HF knowledge**

The main features of perception and cognition as well as sources of behavioral variation are discussed in detail. The increasing importance of decision making places emphasis on the higher cognitive levels. In perception, detection may be the most basic aspect, but the meaning of what is perceived for the situation is crucial. Perception is the input to cognition. Sense making and the various memory functions underlie the higher cognitive functions needed to set out a tactic or take reasoned decisions. Also the feedback from cognition, controlling perception to look for additional information must be represented.

Weaknesses in HF knowledge are revealed by the demands of the latest international operations. The effects of soldier behavior on the attitude of the local population is receiving increasing interest in the field, but the scientific knowledge in this field is still fragmented and has not reached a useful level of modeling.

**Application of this document**

This document attempts to provide concrete guidance regarding integrated HF modeling. Former advice to NATO included a large number of high-level recommendations, which are mostly still valid. We felt that we should try to push forward at the practical level. The follow-up will show how successful we were in indicating a way forward.

Included are numerous schemes and examples dealing with how HF can be implemented in a structured way. In Chapter IX, Section 2, these considerations are synthesized in an overall scheme and a 19-step process to guide practitioners and analysts through an HF reinforced study case, called good practice. We have chosen to avoid the arrogance of referring to our recommended process as “best practice,” but we hope the reader will find it useful.



## 9 Bilaga 3, RTO Award

### CITATION for the 2008 RTO SCIENTIFIC ACHIEVEMENT AWARD

**HFM-RTG 128 and HFM-RSM143 on**

#### “Human Behaviour Representation in Constructive Simulation”

The R&T challenge taken on by both HFM-128 and HFM-143 was how to advance the state-of-technology on computer constructive simulations of human-centered military activities. In current computer models of military operations, the human aspect is often represented in a mechanistic way bearing little resemblance to observed behaviours in real military missions. Without a proper representation of behaviour, and the reasons behind this behaviour, the validity of the model is seriously flawed, making its performance and predictions questionable.

The conceptual model for representation of the actions of personnel in military simulations that RTG-128 presents in its report, strives towards balanced capabilities for both the human factors *and* operational aspects, while allowing for efficient and valid analyses and studies that military operators rely upon for experimentation and operational planning.

HFM RSM-143 culminated the work of RTG-128 by providing a forum for the promulgation of current knowledge of human behaviour in constructive simulation. This was essential to further the field of human modeling for NATO Modeling & Simulation (M&S).

The quality of each RTG members’ scientific contribution, as well as their knowledge of the military domain and the breadth of their combined expertise, was an essential ingredient for this authoritative contribution to NATO M&S capability needs. The RTG report offers a seminal analysis of the state of the art as well as new insights for an appropriate synthesis for incorporating advances from human factors and cognitive science into NATO’s operational M&S.

However, the most prominent contribution of the RTG members is their effort to push M&S science forward to the practical level. Included in their report are numerous schemes and examples dealing with how human factors can be implemented in a structured way. These considerations are synthesized in an overall scheme and a 19-step process to guide practitioners and analysts through a human factors reinforced study case, called good practice. It is not for them, but certainly it is up to us, the NATO RTO Board, to refer to this 19-step process as “best practice”!



## 10 Bilaga 4, Hifi Engine

HiFi Engine is a simulator engine developed at the Department of Man System Interaction (MSI) at the Swedish Defence Research Agency (FOI). HiFi Engine is developed based on open source components, which handle different functionalities, e.g. physics, sound and network communication, etc. (Figure 1). HiFi Engine is mainly used for simulations, and a core of functionality adds specific capability to the simulation engine. HiFi Engine is an important tool for simulations and has been used in numerous research projects (Kylesten et al., 2004) to create experiment platforms. For instance, in a motion based Combat Vehicle simulation, with the purpose of studying multimodal threat information (Carlander, Eriksson & Oskarsson, 2007). It is possible to both simulate air and ground vehicles. Current work aim at utilizing tactile information cueing in brown/white situation during helicopter landing.

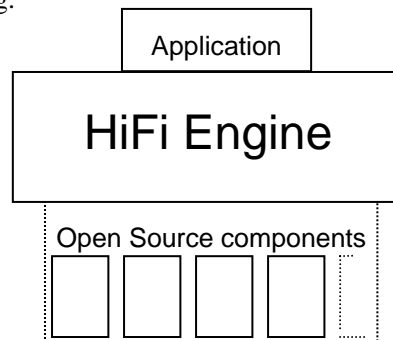


Figure 1. The architecture of software developed using HiFi Engine. The application is built on top of HiFi Engine and can utilize all functionality HiFi Engine provides.

HiFi Engine is also the backbone in the simulator facility at MSI, the MSI Laboratory, where it is responsible for creating an immersive simulator environment for operators placed in the simulator. (Figure 2)

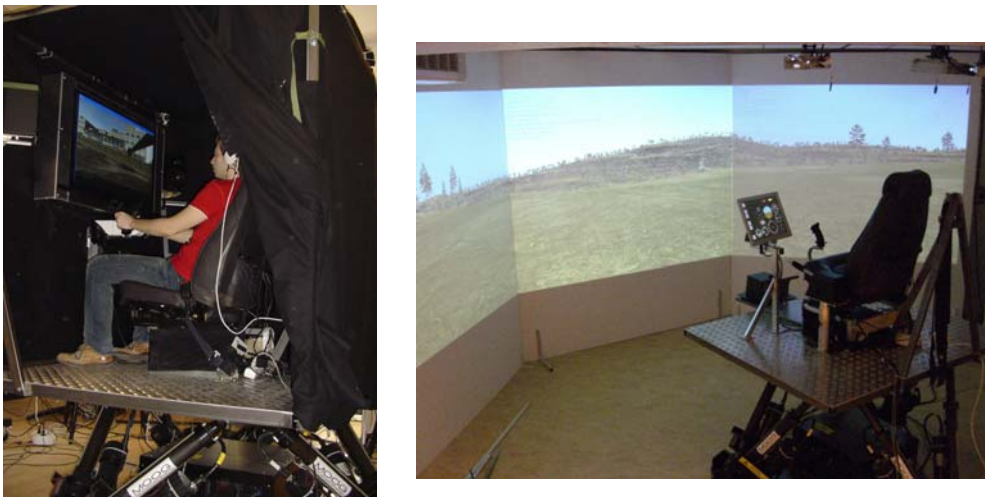


Figure 2. Two different setups at the MSI Laboratory. On the left is HiFi Engine providing motion information to the motion platform and visual and sound stimuli in a Combat Vehicle 90-simulator. On the right is the setup adapted for helicopter simulation.



## 10.1 Projects

The following projects have developed simulator applications using HiFi Engine:

- Operator Site 2003-2005, Multimodal Displays 2006-2007 developing prototypes for intuitive displays utilizing vibro-tactile, 3D audio and visual displays.
- GENESIS, analysis methods for modern digital telecommunication systems to prevent intersystem-interference problems in military and Homeland-Security applications.
- NBCPatria, simulator of a vehicle used for indicating NBC pollutions in a virtual environment.
- KoLäge, a simple command and control system used to investigate in what situations it is crucial to have a 3D presentation of the virtual environment, when commanding ground forces.
- Helicopter simulation. HiFi Engine has been used to create an immersive helicopter simulator in which studies are to be carried out on how tactile displays could be utilized in brown/white out situations.
- MOSART, a modular software environment developed for research projects to evaluate results in a larger context by setting up more advanced simulations and demonstrators.
- Electronic Warfare Simulator (EWSim), a HLA-based framework that simulates the effect of electronic warfare in distributed multispectral scenarios.

Currently, the following projects are taking advantage of the functionality in HiFi Engine:

- MERLIN, development of MERLIN based behaviour models and services for CGF's. The project is taking advantage of functionality in HiFi Engine and it also provides a natural platform to demonstrate the results in.
- Flexible autonomous UGV systems, development of a demonstrator where the mission is to solve a surveillance task using multiple UGV's that cooperate.
- Computer Generated Forces, utilizes HiFi Engine as platform when evaluating behaviour of the CGF's developed.
- GENESIS, analysis methods for modern digital telecommunication systems to prevent intersystem-interference problems in military and Homeland-Security applications.

## 10.2 Publications

Ahlberg Simon, Bergman Joakim, Dahlman Joakim, Folkesson Martin, Forsgren Robert, Hallberg Niklas, Hasewinkel Håkan, Hedström Johan, Johansson Mattias, Kylesten Birgitta, Söderman Ulf, Törne Anders, Ölvander Christin, (2006) *Teknik för ledning av insats i urban miljö - TEBE II slutrapport*. FOI-R--2176--SE, User report

Eriksson Lars, Lindahl Björn, Hedström Johan (2006) *Konceptutveckling av operatörsstöd. 2006*, FOI-R--2210—SE, User report

Sennersten Charlotte, Alfredson Jens, Castor Martin, Hedström Johan, Lindahl Björn, Lindley Craig, Svensson Erland, (2007) *Verification of an experimental platform integrating a Tobii eyetracking system with the HiFi game engine*. FOI-R--2227--SE, Methodology Report

Martin Insulander (2008) *Development of a Helicopter Simulation for Operator Interface Research* (Master's Thesis DRAFT 2)

Stenumgard Peter, (2007) *Arkitekturbeskrivning GENESIS som info till anbudsfordran NTK*



# 11 Bilaga 5, Athena

## 11.1 Introduction

This document presents a framework for development and use of Human Behaviour Representations (HBR<sup>1</sup>) in Modelling and Simulation (M&S). The framework is developed at the Swedish Defence Research Agency (FOI) and is produced by a project called "Computer Generated Forces", investigating how to use HBR models within M&S for the Swedish Defence. A main objective is to create a foundation for a software library of reusable HBR models that can be used in different simulation software. The framework is a partly result of that objective.

The framework is not supposed to serve as a new cognitive architecture nor a simulation framework for HBR or Computer Generated Forces, CGF. The framework is used to facilitate integration of HBR models into existing and future simulation systems, computer games, etc. By adapting HBR models and engines, the framework ensures that different models can work together in a simulation, that they can be reused and that models can easily be configured, composed and integrated with minimum effort.

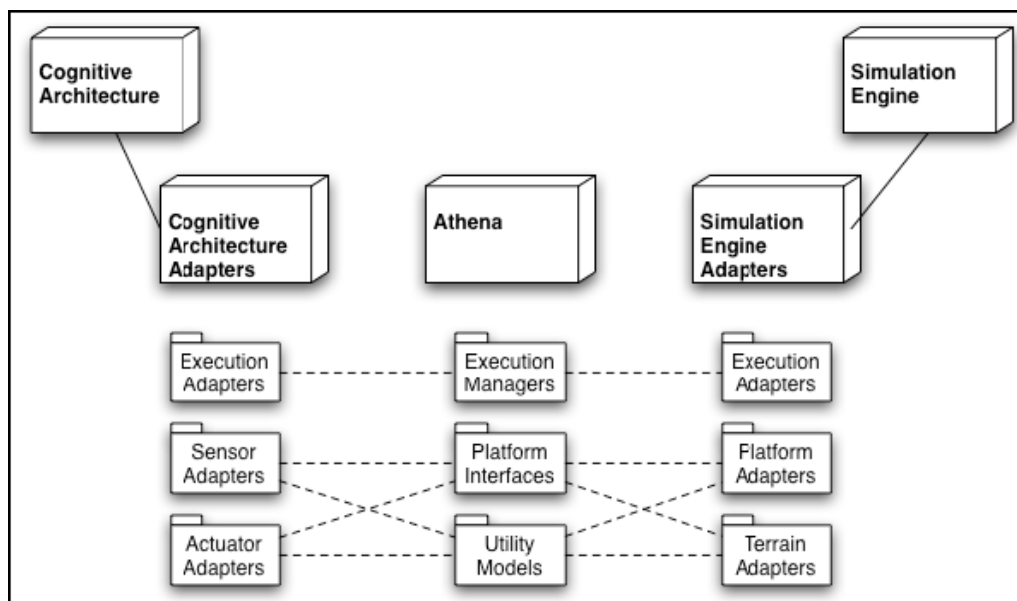


Figure 3. *The Athena framework "glues" simulation engines and HBR models together.*

By creating reusable components of HBR, a model library could be created. HBR could then be picked from the library and "plugged" into different simulation systems.

Later in this section, the document presents some motives why a framework is needed; in section 2, it describes previous work and the Merlin framework that the Athena framework

<sup>1</sup> Here also meaning Artificial Intelligence (AI) and Computer Generated Forces (CGF) technology

uses. In section 3, a more detailed description of the Athena framework is presented. Conclusion and benefits for using the framework is presented in the last section, section 4.

### 11.1.1 Motivation

Human Behaviour Representation in M&S has mainly been used to populate simulations with team-mates, opponents and neutrals in order to create more realistic scenarios with a larger number of entities. Typically, such simulations involve less human influenced HBR with strong rational behaviour often founded on optimization methods of artificial intelligence.

In the last ten years, the use of more human-like models have been proposed in order to create more realistic representations, and to create HBR for studying and reproducing real-life phenomena observed from military operators. Cognitive science, artificial intelligence and computer science has applied theories of human behaviour into software called cognitive architectures. Several architectures have been developed to aid in HBR development each with its own theory of human decision-making, learning, memory, etc. Thus, different cognitive architectures have different characteristics inherited by the resulting HBR.

Sometimes hybrid architectures are used to complement each other, where each architecture handles its own sets of mechanisms, or where architectures communicate data between each other. A model of vision could for example filter objects in the environment before they are sent to the decision making model.

To explain observed behaviour from real-life situations in different and sometimes extreme environments these architectures are sometimes modified through “performance modifiers”. The modifiers add individual characteristics and account for external stressors such as heat, noise, darkness etc. or internal stressors such as anxiety, fear, aggression, etc.

As implicated above, the use of HBR in M&S spans a whole range of military applications. There are lots of different theories, architectures, algorithms, etc that try to model different aspects of human behaviour. There is no single candidate that can adequately perform within all these applications. As most software engineers, HBR developers have developed their own toolboxes and sub-models to be used in the development. There are also numerous CGF architectures that are bound to its internal simulation framework (or more often, the CGF’s are part of the simulation framework).

Creating HBR models could be very time consuming. A lot of time invested in knowledge elicitation and elaboration together with theories about human cognition will produce more realistic HBR. As a software component, such models should also be developed for:

- Reuse. It should be possible to reuse sub-models of HBR to create new variants and new behaviour. It should also be possible to reuse complete HBR models in different simulation systems and applications.
- Easy of use. Non-programmers should easily be able to compose, configure and inject HBR models into simulation software. HBR model developers could

concentrate on modelling behaviour. Standardized formats could provide guiding principles for both HBR- and simulation software developers.

- Extensibility. Using component based software engineering principles, it should be possible to extend and adapt models for different problems.

To support the development and use of HBR a framework to aid in these processes would be beneficial. Because of the diverse uses of HBR models, such a framework should provide tools to support different HBR models and simulation systems. A component based framework could also be used to collect models and sub-models into libraries of HBR, favouring reuse and cooperative development.

## 11.2 Background

### 11.2.1 “AgentInterface”

FOI has previously developed simulation interfaces for the cognitive architecture Soar. A framework, called AgentInterface was developed and used together with the SGIO<sup>2</sup> interface to configure agents and inject them into simulation systems (FLAMES<sup>TM</sup>, ACM, FENIX<sup>3</sup>). The AgentInterface defined interfaces, adapters, a component model and an HLA adapter. Although highly configurable, AgentInterface lacked runtime component loading, XML syntax and was targeted towards a single cognitive architecture, Soar. New models had to be added and compiled with the application and the configuration files were, although human-readable, hard to parse and understand.

### 11.2.2 The Merlin framework

Merlin is a C++ software framework developed at the Swedish Defence Research Agency, and was initiated to form a foundation for a library of software models associated with air combat simulations. Merlin was developed as a component-based framework, so that models and sub-models easily could be replaced without changing the framework, dependent applications or dependent models. Merlin defines a set of interfaces for assembling and interacting with components and services that can be used by models and applications.

With its own component model Merlin uses an “object factory process” to manage creation, composition and destruction of components. To use this service, a developer must conform to a set of standard interfaces that are used by Merlin for object management.

---

<sup>2</sup> A C++ interface for external applications to interact with Soar

<sup>3</sup> FLAMES is simulation framework, ACM is an open source flight simulator, FENIX is a flight simulator developed at FOI

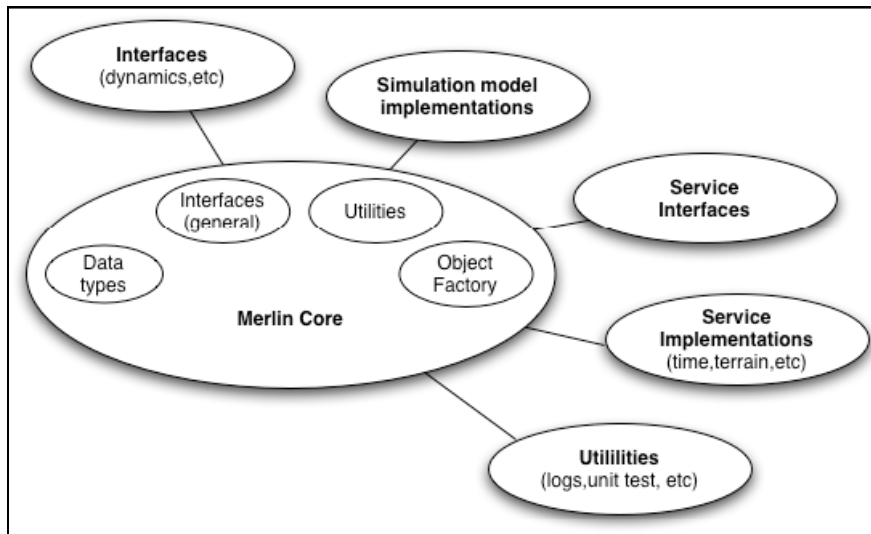


Figure 4. **Components of the Merlin framework**

Merlin also consists of some basic data types for math, geometry, etc. Together with the object management service these components constitute the “core” of Merlin, and are general enough to be used by a variety of simulation applications.

In addition, Merlin also provides more domain dependent interfaces for platforms, radars, propulsion, guidance, etc. These interfaces are primarily intended for air combat simulations, although many interfaces are generic enough to be used by other models and applications. Existing systems typically have their own set of interfaces and utilities that needs to be used, while still using the Merlin object management service.

### **Object Factory Process**

The object factory process is a method of creating objects in runtime from textual strings. These strings are organized in an XML description that contains the name of the objects, describes their relationships with other objects and the initialize values of their attributes. By parsing the XML description, the factory process can create the objects, aggregate them, establish relationships, set the initial values and then deliver the objects to an application or other models.



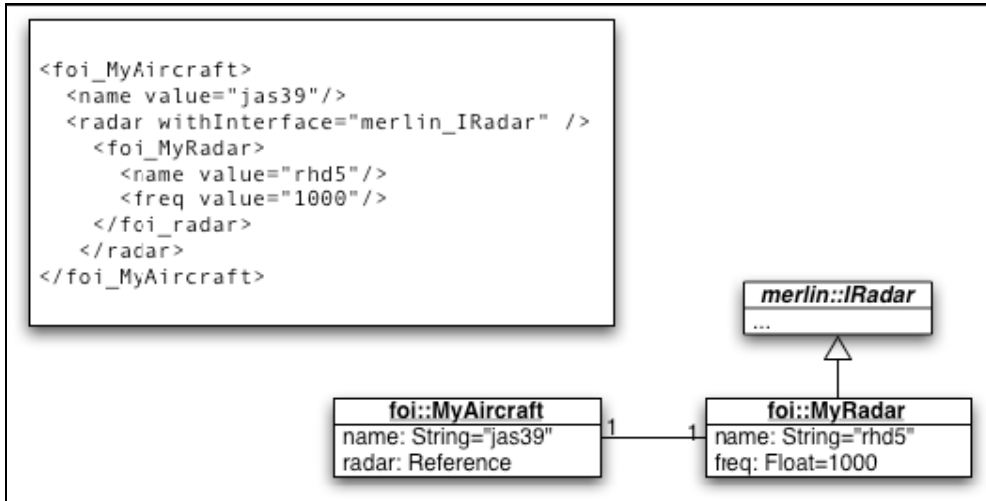


Figure 5. *Upper right, an example of an XML description, that will generate the objects, outlined in the lower left of the figure.*

When the factory process encounters the name of an object, it searches through one or more directories for binary files for the class definition. It loads the file as a dynamic load library (DLL), and instantiates the class into an object. It then sets the object's initial values specified in the XML file.

Since, relationships are established through interfaces, the most specialized form of any object does not need to be known. The factory process can load class definitions in runtime which also makes it possible exchange objects sharing the same interfaces without changing dependent models or the application. Also, replacement of objects can be done just by modifying the XML description either by hand or in a graphical editor. Typically, all binary files that hold class definitions are stored in a common folder, a model library, so that the user easily can pick which models to use.

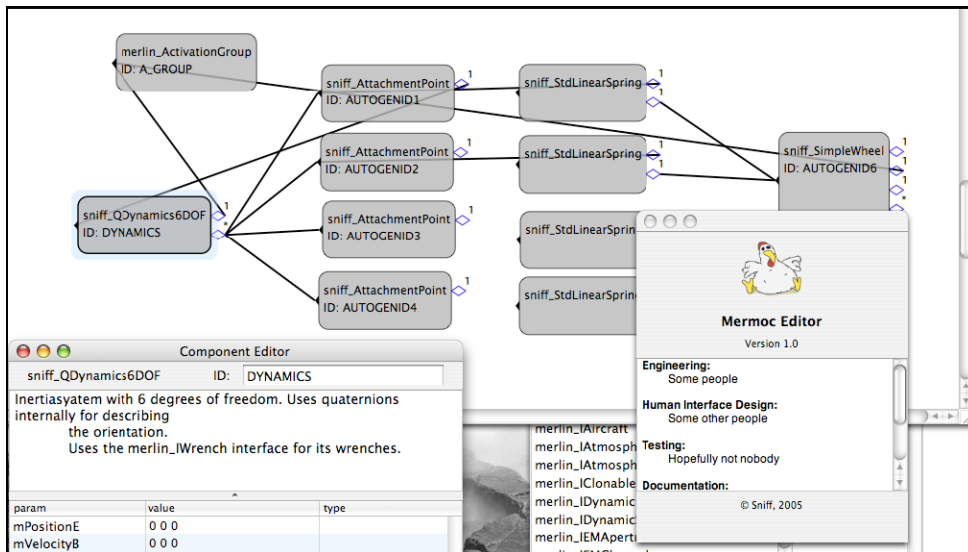


Figure 6. *A diagram editor that can be used to generate XML descriptions just by connecting components and entering values.*

Merlin also provides special objects called “services”, that typically that typically only will be instantiated once in a simulation. Time managers, terrain managers, atmosphere are examples of such objects. These components can be configured and replaced just as other components, Merlin just provide some default implementations. The difference is that service components are obtained automatically through subscriptions.

### 11.3 The framework

Athena is a C++ framework to aid in use of HBR in simulations. The framework uses the Merlin core modules (data types and the factory process) to maintain a component-based approach. On top of Merlin, Athena adds managers and interfaces for handling HBR models, called agents. Operator interfaces, that could aid the agents in their tasks, are also provided. Some examples are: navigation aids, identification aids, platform sensors, platform controls, etc. Athena provides some implementations of these interfaces (for example a waypoint controller), but a developer may create custom implementations.

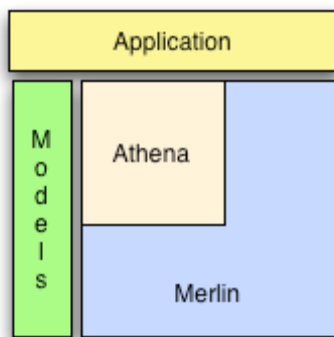


Figure 7 *Athena uses some interfaces and components of Merlin. An application will typically use services from both Athena (i.e. agent management) and Merlin (i.e. time management). An application may also interact directly with the models, although most interactions will be done through the interfaces*

The framework is HBR architecture independent. This feature comes with a cost: when a new architecture is introduced, it has to be adapted to meet the requirements of the framework. Once the adapter is created it can easily be used to configure the architecture for the current simulation problem using standard formats or editors. When the architecture is “adapted”, HBR models and their sub models can be reused, composed and configured using the same formats. By using Athena’s interfaces, adapters for custom HBR architectures may be created. Athena currently provides one such adapter for using Soar as the behaviour engine.

#### 11.3.1 Interfaces

Athena could be considered as a set of adapters that glues HBR models to simulation engines. On the simulation side, the glue consists of interfaces that adapt the simulation engines models so that they can be reflected by the HBR models in a uniform manner. On the opposite side there are interfaces that are used to adapt HBR engines to the framework.

#### 11.3.2 Simulation engine interfaces

Simulation engine interfaces are interfaces that should reflect typical objects and sub-objects in a military simulation engine. Since, Merlin was designed to adapt air combat simulations, it already provides some of these interfaces. For example, Merlin has interfaces for rigid bodies, kinematics, terrain, atmosphere, etc. Athena therefore reuses some of the interfaces, while adding its own sets of interfaces for identification, navigation, communication and other interfaces reflecting typical objects that an operator may interact with.

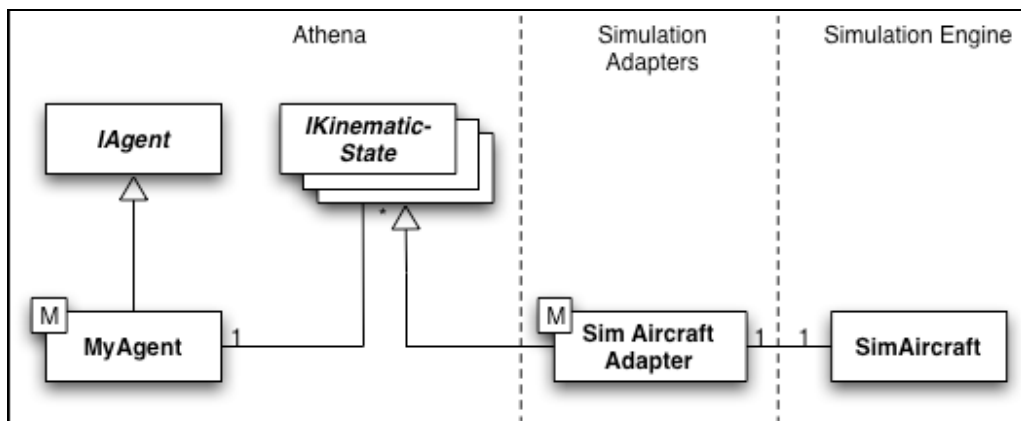
### An Example

This is a simple example of how a platform from an existing simulation engine could be adapted so that an Athena agent could manoeuvre the platform.

First, we assume that there is an existing aircraft model in the simulation engine, called SimAircraft. We also assume that we already have created an agent called MyAgent that depends on a few generic interfaces that could describe an aircraft or any other platform.

We start by creating an adapter called AircraftAdapter that should provide the interfaces that our agent depends on. We add a reference to the SimAircraft object, so that the adapter may transfer and transform data from the SimAircraft object to the MyAgent object through the interfaces. The MyAgent object knows how to interact with the interfaces but know nothing about the vehicle that actually implements these interfaces. Remember that this is only a programmatically way of creating higher interoperability by connecting components. How the agent will operate the platforms is coded in its knowledge system.

Once this is done, any other Agent object that depend on the same sets of interfaces may operate the SimAircraft object through the adapter, thus the adapter only need to be created once. Since most simulation engines uses object oriented approaches the adapter may often be split into several adapters for low-level objects in the simulation. This will create a more flexible system and the possibility to reuse adapters for several simulation objects (both tanks and aircrafts have positions, rotations, etc.)



### 11.3.3 Simulation utility objects

Athena also provides default implementations of some of its interfaces. For example, Athena provides interfaces for waypoint handling. A simulation engine may adapt its own waypoint handling functions to the interface as in the previous example. However, sometimes this can be tedious or the simulation engine does not provide a waypoint handler. In that case Athena provides an implementation of a waypoint handler that can be used. Other examples of such implementations are communication devices, terrain

navigation helpers, identification aids, etc. As usual, a user may create custom implementations of such objects.

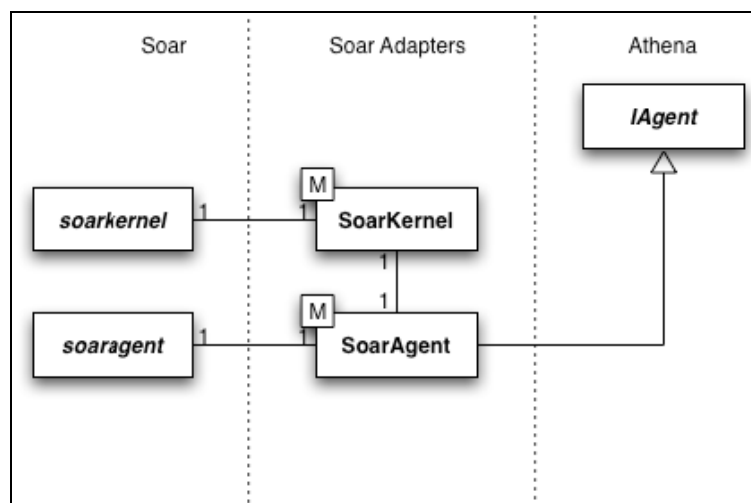
### 11.3.4 HBR interfaces

Currently, the only interface that is used for integration of HBR models is the Agent interface. It should be used to adapt HBR representations to the Athena framework.

### 11.3.5 The Soar adapter

Soar is a cognitive architecture, developed at the University of Michigan, that is used to model human behaviour and general intelligence. Soar can be run by itself or can interact with simulation or game engines through a C kernel API or through a C++ API called SML (Soar Markup Language). Both API's support execution controls and information exchange between Soar and external simulators. The Soar plug'in was created to simplify configuration of the Soar kernel, and the Soar agents but also to further facilitate the information exchange between a Soar agent and a simulator.

The Soar plug'in contains several components, where the SoarKernel, SoarAgent and SoarWME's are the most important.



### SoarKernel

The SoarKernel is an adapter for the Soar kernel class in SML, `sml::Kernel`. Through the adapter a kernel can be easily be configured by changing the adapter's properties in the XML description. The configuration properties are:

- Module. Which binary Soar kernel (DLL) to use. Enables use of customized kernels.
- Process mode. The kernel can be set to run in embedded mode (same thread), in a different thread or through a network (a different computer, a Soar Debugger for instance).
- Optimized or non-optimized execution mode

```

<foicgf_SoarKernel>
  <implements>
    <interfaces>
      <foicgf_1SoarKernelManagerClient/>
    </interfaces>
  </implements>
  <id value="default"/>          <!-- unique identifier -->
  <module value="SoarKernelSML"/> <!-- Soarkernel DLL -->
  <thread value="current"/>      <!-- Thread (current,remote) -->
  <optimization value="true"/>   <!-- Optimization (true,false) -->
  <remoteip value="127.0.0.1"/>  <!-- IP of remote kernel instance -->
  <port value="12121"/>
</foicgf_SoarKernel>

<!-- More kernels?... -->

```

Several adapters may be created in the same application, so that different agents may use different kernels. A kernel is only instantiated if it is used by an agent. Soar kernels are stored in a SoarKernelManager container that can be globally accessed by agents to retrieve kernels.

### SoarAgent

The SoarAgent is an adapter for the Soar agent class in SML, sml::Agent. Every SoarAgent needs a SoarKernel as its behaviour engine. An agent may also need sensors, actuators, filters, etc. For flexibility, the SoarAgent is a generic container of components that transfer data between simulation models and the Soar Working Memory. These components are called SoarInputs and SoarOutputs. Through the adapter, an agent may be configured by editing its properties in the XML description. The configuration properties are:

- Update interval, how many updates per second.
- Name, the name of the agent
- Kernel, which kernel to use. Specified by an id that should match a kernel id.
- Runtloutput, specifies whether to use Soar's runtloput
- DecisionsPerCycle, how many decisions the agent will create in one cycle.
- Productions, the filename of the Soar production rules

```

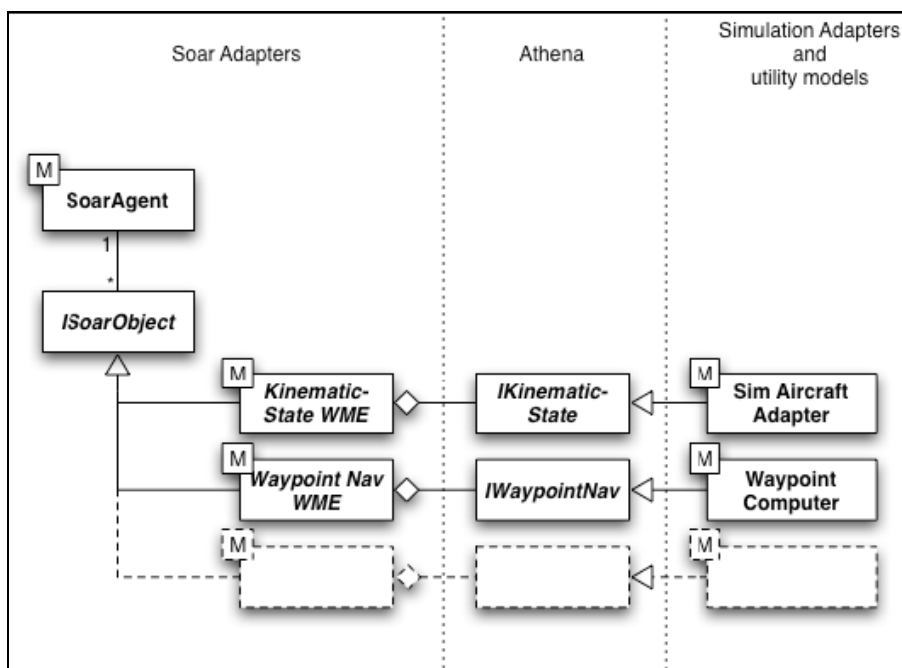
<foicgf_SoarAgent>
  <implements>
    <interfaces>
      <merlin_ITimeManagerClient/> <!-- Get time manager service -->
      <athena_IAgentManagerClient/> <!-- Get agent manager service -->
      <foicgf_ISoarKernelManagerClient/> <!-- Get soar kernel service -->
      <merlin_IActivationEntity/>
      <athena_IAgent/>
    </interfaces>
  </implements>
  <mActivationGroup withInterface="merlin_IActivationGroup" IDREF="A1"/>
  <mStepSize value="1"/> <!-- Updates/sec -->
  <name value="agent1"/> <!-- Unique name -->
  <kernelid value="kalle"/> <!-- Which kernel to use -->
  <runTilOutput value="1"/> <!-- Run until output, 0=false, 1=true -->
  <decisionsPerCycle value="1"/> <!-- Decisions per cycle -->
  <productions value="$PWD/Agents/Agent1.soar"/> <!-- Productions file -->
  <!-- To be continued.. -->

```

**SoarInput**

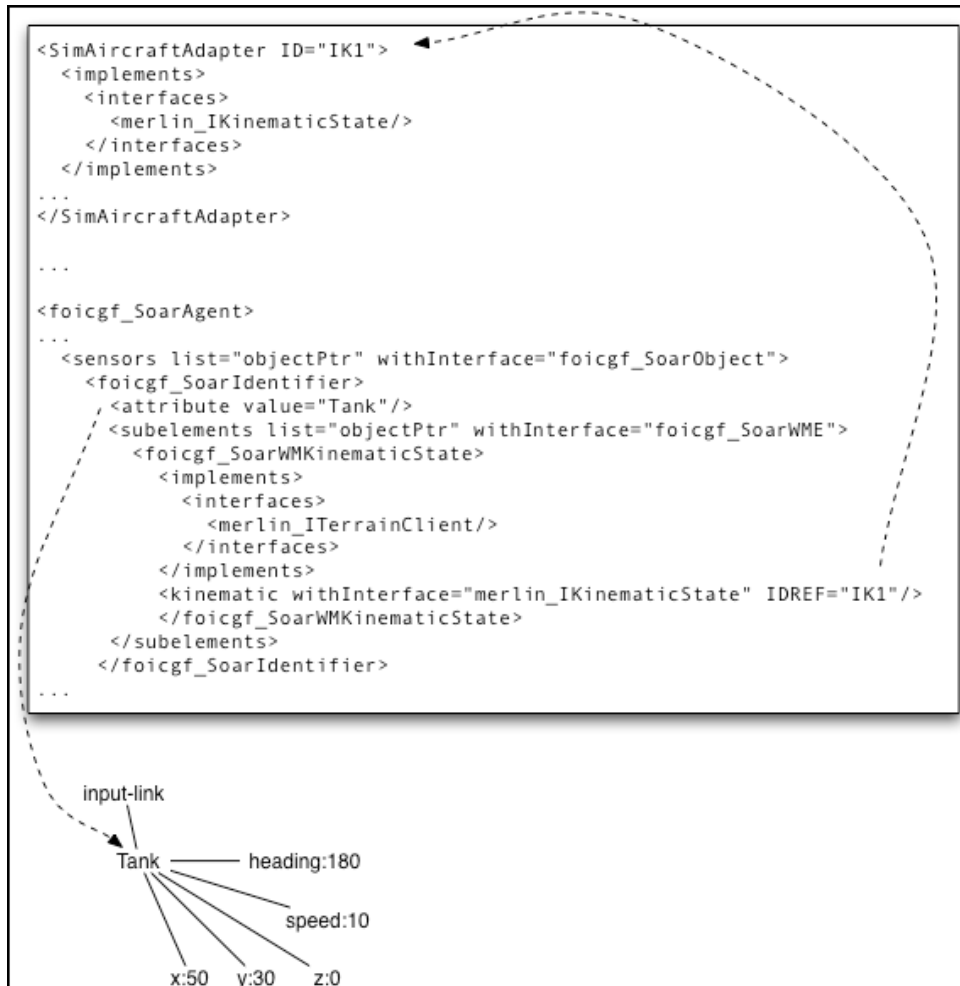
Every SoarInput is specialized to transfer data from a simulation object to a Soar working memory structure. The SoarInput’s use simulation interfaces and often there is a one-to-one correspondence between a SoarInput and an interface i.e. there is one SoarInput that transfers waypoint data, there is one SoarInput that transfers geometric data, etc. What information that is transferred is decided by the SoarInput implementation, as usual there maybe several different implementations of a SoarInput that uses the same interfaces.

Soar’ working memory is organized as a graph of symbols, with a root node called “input-link”. From the root node it is possible to navigate the graph through identifiers, while the leaf nodes are called simple working memory elements. A SoarInput may contain both identifiers and simple wme’s. Sometimes it is desirable to create intermediate nodes between the SoarInput (or actually its elements) and the root node. This can be done by adding a SoarIdentifier that is an adapter to sml::Identifier. For example, a waypoint data input and terrain data input might both be put under a common “navigation” node.



## SoarOutput

SoarOutput is a wrapper for sending Soar commands to a simulator. When a SoarAgent wants to send a command it puts symbols on a node called “output-link”. Every SoarOutput registers a command token to the agent. When the agent discovers that the Soar kernel creates a command symbol that matches the output token, it forwards that command to the corresponding SoarOutput component. The SoarOutput then transforms and sends the command with its parameters to appropriate interfaces.



## 11.4 Conclusion

Athena and Merlin provide a framework for integrating HBR models into simulation engines. The integration is done through custom adapters that are created from standard interfaces. Once the integration of an HBR model or HBR engine is done, new type of models can be added and configured just by configuring an XML description. The HBR models can therefore be delivered as configurable components, put in a library and used in a “plug’n’play” fashion. By using graphical- or XML editors, cognitive modellers can quickly configure agents and use them in various simulation environments. They can therefore focus on behaviour modelling without dwelling into software integration problems.

## **11.5 Future work**

The framework will always be in a progressive development stage, since new demands will lead to additions of new models and adapters. Once, created the models will persist within the framework and reused. Apart from adding new models the framework needs to be evaluated with different types of HBR architectures and simulation software. Interfaces need to be updated and added to suit new applications. Also, Merlin is still under development, bringing new features inherited by the Athena framework.