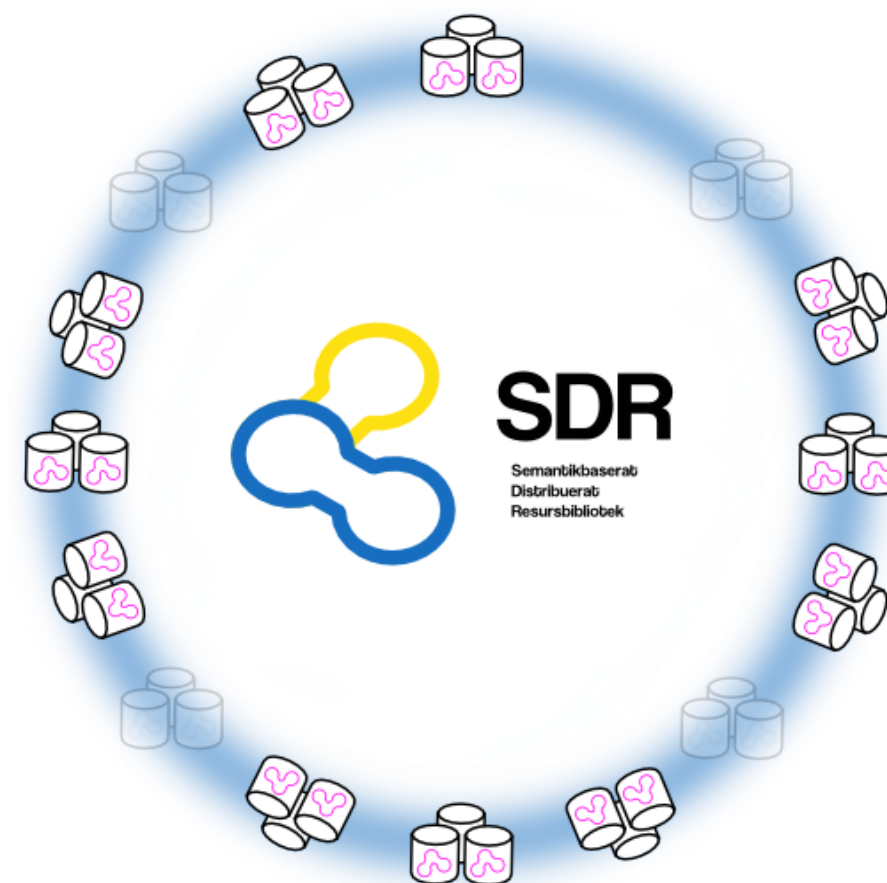


MARIANELA GARCIA LOZANO, FARSHAD MORADI, EDWARD TJÖRNHAMMAR



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.

Slutrapport för SDR - Semantikbaserat Distribu- erat Resursbibliotek

Marianela Garcia Lozano, Farshad Moradi, Edward Tjörnhammar

Slutrapport för SDR -
Semantikbaserat Distribuerat
Resursbibliotek

Titel Slutrapport för SDR - Semantikbaserat Distribuerat Resursbibliotek

Title SDR End Report - Semantic Distributed Repository

Rapportnummer / Report no FOI-R--2608--SE

Rapporttyp / Report type Användarrapport / User report

Utgivningsår / Year 2008

Antal sidor / Pages 48

Kund / Customer FM

Forskningsområde 2. Operationsanalys, modellering och simulering

Research area 2. Operational Research, Modelling and Simulation

Delområde 21. Modellering och simulering

Sub area code 21. Modelling and Simulation

Projektnummer / Project no E7120

Godkänd av / Approved by Martin Rantzer
Chef, Informationssystem

ISSN ISSN-1650-1942

FOI Totalförsvarets forskningsinstitut
Informationssystem
164 90 STOCKHOLM

Sammanfattning

SDR är förkortningen för ett *Semantikbaserat distribuerat resursbibliotek*. Det är en distribuerad infrastruktur med vars hjälp olika organisationer och grupper skall kunna dela med sig och utnyttja varandras resurser. Resurserna kan vara heterogena och vara allt från projektrelaterade filer, simuleringskomponenter, sensordata till utdelade datorer som fungerar som beräkningsresurser. Infrastrukturen är en hybridlösning mellan en indextjänst, ett filsystem och ett middleware. Den bygger på en blandning av tekniker från P2P, Grid och Semantic Webb. Användningsområdena för SDR är många och vi har pekat på några i denna rapport (t.ex. Modellbaserad förmågeutveckling). SDR är också namnet på ett två-årigt projekt som har löpt under verksamhetsåren 2007-2008. Det har bedrivits inom ramen för FoT-området Modellering och Simulering.

Denna rapport beskriver projektet och verksamheten som bedrivits, infrastrukturen och dess ingående komponenter samt de samarbeten och resultat som erhållits. Rapporten avslutas med dels en sammanfattning och diskussion av de frågeställningar som framkommit under projektets gång, dels en titt på framtiden och de planer som finns på fortsättning.

Nyckelord

SDR, Semantik, Distribuerade system, Resursbibliotek, Lagring, Robusthet, Skalbarhet, DHT, Hashtabell, Grids, Ontologi, RDF, SPARQL, STIL, Modellbaserad förmågeutveckling

Abstract

SDR stands for Semantic Distributed Repository. It is a distributed infrastructure intended to alleviate problems in information exchange between different organizations. Resources are considered heterogeneous and can range from project specific files, simulation components, sensor data to service descriptions. The infrastructure is a hybrid between an index service, a filesystem and a middleware for distributed semantic resource lookup. SDR leverages from techniques from the domains of P2P, Grid and Semantic Web. The uses for SDR are bountiful and we have specified a few of them in this report (e.g. STIL, *Förmågebaserad modellutveckling*).

SDR is also the name of a two year project running between 2007 and 2008. It has been conducted within the frame of modelling and simulation.

This report describes the project, its activities, infrastructure, components and collaborative results. The report is concluded with a summary and discussion related to the discovered challenges as well as a glimpse of the future.

Keywords

SDR, Semantic, Distributed, Repository, Storage, Robustness, Scalability, DHT, Hash Table, Grids, Ontology, RDF, SPARQL, STIL

Innehåll

1	Introduktion	11
1.1	Projektet	11
1.2	Mål och nytta	11
1.3	Genomförande	12
1.4	Rapportstruktur	13
2	Semantikbaserat distribuerat resursbibliotek	15
2.1	SDR i sitt sammanhang	16
2.1.1	Relaterade ansatser	17
2.2	Användningsområden	19
2.2.1	Fördelarna med semantisk sökning	19
2.2.2	Modellbaserad förmågeutveckling	20
2.2.3	STIL	21
2.2.4	Cloud Computing	21
3	Arkitektur	23
3.1	Funktioner	24
3.2	Systemkomponenter	24
3.3	Säkerhet	26
3.4	Datamodell	27
3.5	Semantisk sökning	29
4	Samarbetsprojekt	31
4.1	REVVA och SDR	31
4.1.1	Krav	31
4.1.2	Resultat	32
4.2	Komponentbaserad modellutveckling och SDR	33
4.2.1	Resultat	34
4.3	SM4ALL och SDR	35
4.3.1	Krav	36
4.3.2	Resultat	36
5	Sammanfattning & diskussion	37
5.1	Fortsättning	37
6	Appendix	39

A GSN	41
B SRML-exempel	43
C SPARQL-exempel	45
Litteraturförteckning	47

1 Introduktion

1.1 Projektet

Projektet "Semantikbaserat Distribuerat Resursbibliotek" (SDR) är ett tvåårigt uppdrag från Försvarsmakten som startades 2007. Det är en fortsättning, fokusering och fördjupning av ett av de viktigare verksamhetsspåren inom det tidigare genomförda NätSim-projektet¹. Syftet med SDR-projektet är att:

- stödja insatsförmågor/verksamheter med behov av resurser med information från modellering och simulering (M&S),
- uppnå högre tillgänglighet, enklare återanvändning och högre utnyttjandegrad av resurser.

SDR-infrastrukturen bygger på en vidareutveckling av de NätSim-komponenter som låg till grund för den distribuerade resurshanteringen och säkerheten.

1.2 Mål och nytta

Såväl militärt som civilt finns ett stort behov av att kunna samarbeta och interagera över organisationsgränser i form av dynamiska virtuella organisationer. Samarbete innebär utbyte av resurser och information. För att kunna tillgodogöra sig den information som utbyts behöver man ha samma begrepps-förståelse. För att kunna åstadkomma detta, d.v.s. utbyta och dela information och resurser, krävs en plattform som tillhandahåller en gemensam begreppsapparat, tjänster och infrastruktur. Ett distribuerat resursbibliotek är en central del i detta. Olika organisationer och grupper ska i detta distribuerade resursbibliotek, med hjälp av en överliggande gemensam plattform, kunna dela med sig och utnyttja varandras resurser. Resurserna kan vara allt från projektrelaterade filer, simuleringsmodeller, komponenter, sensordata och verktyg till beräkningskraft och beskrivningar av olika personers kompetenser. Viktigt att notera i detta sammanhang är att det handlar om heterogena resurser.

SDR är tänkt att hantera flera sorters resurser samtidigt. Målsättningen för projektet är att skapa ett säkert och distribuerat resursbibliotek som, baserat på öppna standarder och tekniker, är säkert, operativsystemsoberoende,

¹NätSim-projektet (Nätverksbaserad modellering och simulering) bedrevs under åren 2004-2006 och hade det övergripande målet att stödja modellering och simulering som en effektiv metod eller verktyg inom det Nätverksbaserade försvaret. Inom ramen för det projektet ingick att utveckla en skalbar, modulär och tjänsteorienterad arkitektur som kunde erbjuda tjänster för att underlätta nätverksbaserad M&S såsom distribuerad resurshantering och modellexekvering, komponentbaserad M&S, datorstödd samverkan, säkerhet, med mera. För att läsa mer om NätSim-projektet rekommenderas rapporterna [4, 5, 6].

organisationsoberoende, transparent för användaren, konsistent, versionshanterat, medmera. Resursbiblioteket skall möjliggöra snabb och flexibel informationsinhämtning och -delning. Koncept, resurser och deras relationer skall även tillgängliggöras på ett transparent och generellt vis. Nyttan och målet kan sammanfattas i följande punkter:

- Målet är att tillhandahålla ett verktyg med vars hjälp (i en nätverksbaserad miljö) det är möjligt att:
 - erbjuda en distribuerad infrastruktur
 - erbjuda säker resursdelning
 - ha fokus på hög tillgänglighet och robusthet
 - vara oberoende av underliggande plattformar
 - på ett enkelt sätt hitta rätt resurs(er) d.v.s. semantisk sökning
 - tillvarata den kunskap som utvecklades inom NätSim-projektet.
- Nyttan är att:
 - användare på ett enkelt sätt skall kunna samarbeta över organisationsgränser
 - användare enkelt kan hitta de resurser de behöver för att lösa sina uppgifter
 - användare enklare kommer kunna lösa gemensamma problem
 - man uppnår en högre utnyttjandegrad av resurser.

1.3 Genomförande

Tre huvudspår identifierades tidigt som väsentliga för att uppnå projektets syfte och mål.

A Vidareutveckling

- Funktionalitet: Vi har undersökt hur sökfunktionen kunde förbättras samt hur vi kunde stödja matchning och sammansättning av komponenter.
- Säkerhetsaspekter: Vi undersökte säkerhetsaspekterna utifrån kundens behov och designade ett säkerhetsramverk.
- Användarvänlighet: En viktig del i användandet av SDR är skapandet och sökandet av resurser och information. För att SDR ska kunna användas i en större skala var det av vikt att göra dessa funktioner mer användarvänliga. Framför allt var det viktigt att se hur skapandet av metadata kunde underlättas och automatiseras.

B Identifiering av samarbetspartners

- Försök till att identifiera lämpliga samarbetspartners (både nationella och internationella) har bland annat krävt studier av relaterade ansatser inom området.

C Analys av användningsområden inom ledningssystem

- SDR är en generell arkitektur som hittills framför allt har inriktats mot M&S-området. Det tredje spåret inom projektet har varit att undersöka om och hur SDR skulle kunna användas för att stödja Försvarsmaktens ledningssystem.

Fokus under projektets första år (2007) låg framför allt på punkt A - Vidareutveckling och frågeställningar, då det är grunden för SDR-arbetet. En förhoppning var att under året även etablera samarbete med användare för att på så sätt identifiera nya krav och användningsområden för SDR. Denna satsning lyckades och ett antal samarbeten med olika projekt inleddes. Förra årets arbete finns beskrivet i rapporten [9].

En del av de inledda samarbetena har fortsatt och fördjupats under detta år (2008). De finns beskrivna i kapitel 4. Resultaten som erhållits inom dessa samarbeten har lett till ytterligare förbättringar av infrastrukturen och användarvänligheten. Internationellt har vi genom uppstarten av ett nytt EU-projekt fått kontakt med partners runt om i Europa som sysslar med relaterade frågor. Vi hyser förhoppningen att detta samarbete kommer att leda till flera positiva synergieffekter i framtiden. Arbetet med punkt C i listan har bl.a. lett till en fortsättning av SDR-projektet i form av ett nytt projekt kallat *"Effektiv informationshantering av M&S-resurser för ledningssystem"*. Detta projekt kommer att fokusera mindre på infrastruktur och mer på att utreda hur M&S-relaterade informationstjänster effektivast och enklast kan beskrivas och hittas. Projektet skall även undersöka hur man lättast lägger in information i systemet för att det skall vara så smidigt som möjligt för användaren att hitta. Slutligen kommer vi även att titta på hur man lättast formulerar frågor som maximerar kvaliteten på sökträffarna. Detta är av speciellt intresse när man har behov av komplexa sökträffar, d.v.s. när man söker efter flera resurser eller tjänster som skall uppfylla ett gemensamt behov och/eller passa ihop. Idag klarar SDR av den typen av sökningar, men det är tyvärr en arbetskrävande process. Detta hoppas vi kunna förändra.

1.4 Rapportstruktur

Denna rapport beskriver SDR-projektet, verksamheten som bedrivits under de båda verksamhetsåren (2007 och 2008), arkitekturen, de hanterade frågeställningarna, samt resultatet. Dessutom redovisas kvalitetssäkringen av projektets verksamhet som framför allt har skett genom samverkan.

Kapitel 2: Här tar vi upp några av de utmaningar som utvecklingen av en infrastruktur av SDR-typ medför. Vidare går vi in på relaterade ansatser och hur SDR står sig i förhållande till dessa. Kapitlet avslutas med

exempel på hur SDR skulle kunna användas generellt och även för Försvarsmakten, detta för att orientera läsaren i möjligheterna med SDR.

Kapitel 3: I detta kapitel beskrivs SDR från ett teknisk perspektiv och har som syfte att fördjupa den intresserade läsaren i dess uppbyggnad, beteende och funktion. Arkitekturen för systemet beskrivs ur ett flertal perspektiv.

Kapitel 4: Detta kapitel beskriver de samarbeten som har bedrivits under SDR:s båda verksamhetsår, status samt de resultat som har erhållits. Här får läsaren ytterligare exempel på hur SDR kan användas.

Kapitel 5: Slutkapitlet ger en sammanfattning och diskussion av projektverksamheten, rapporten och resultaten. Vidare pekar vi även på framtida arbete och utmaningar som är kvar att lösa för att kunna stödja insatsförmågor och verksamheter med behov av av resurshanteringsstöd.

Appendix: För den verkligt intresserade läsaren tillhandahålls fördjupningar och exempel inom valda områden.

2 Semantiskbaserat distribuerat resursbibliotek

Som beskrivits i avsnitt 1.1 är huvudsyftet med SDR att stödja hanteringen av resurser och då i synnerhet inom M&S-området. Därtill ligger det i SDR:s roll att stödja olika verksamheter med samverkan kring externa resurser. I konstruktionen av den nödvändiga arkitekturen, har vi ställts inför en mångfald av utmaningar.

Bland annat har utmaningarna bestått i att bygga en säker distribuerad infrastruktur som är tillräckligt användarvänlig, robust och skalbar för att den ska klara av försvarets krav och behov på hantering av resurser. Aspekter av detta är frågeställningar relaterade till resursers ägarskap, underhåll, utnyttjande och slutligen rättigheter. Därtill kommer utmaningen att skapa informationsmodeller som gör det möjligt att på ett syntaktiskt och semantiskt korrekt sätt tillgängliggöra, söka, byta och utnyttja resurser samt deras metadata.

För att sammanfatta och kategorisera problematiken som SDR-projektet ställts inför följer här en kortfattad lista på utmaningar. Dessa utmaningar har direkt påverkat utformningen av SDR-arkitekturen.

Säkerhet: Resursers tillgänglighet i systemet måste regleras. Eftersom arkitekturen är distribuerad kan flera organisationer delta och man kan inte anta att dessa har full tillit till varandra.

Rättigheter: Resurser kan ha licenser och olika typer av klassificeringar bundna till sig.

Tillförlitlighet: I en distribuerad arkitektur där det finns system som man inte själv äger är det svårt att uppnå tillförlitlighet hos data, både vad gäller kvalitet och korrekthet. Vidare behöver systemet kunna hantera det faktum att det inte går att lita på att andras system är uppe och fungerar, m.a.o är tillgängliga.

Arkitektur, nätverk, infrastruktur: För att möjliggöra samarbete över organisationsgränser behöver den underliggande arkitekturen vara skalbar, robust och modulär. Vidare skall användaren, utan att behöva större kunskap om de tillgängliga resurserna eller deras lagringsplats, lätt kunna söka och finna det hon/han behöver. Systemet behöver alltså hantera data på ett transparent vis.

Olika plattformar: Arkitekturen och infrastrukturen måste kunna hantera olika typer av operativsystem, uppkopplingar och program.

Olika organisationskulturer: Olika organisationer kan ha olika sätt att benämna samma sak vilket medför att samma resurs får olika namn och

beskrivningar. Systemet behöver kunna hantera dessa semantiska olikheter.

Funktionella utmaningar: Detta gäller aspekter som integrering med andra verktyg, sökning, indexering av resurser, autentisering, auktorisering, etc.

De utmaningar och krav som SDR har ställts inför har gjort att arkitekturen och prototypimplementationen utvecklats stegvis och undan för undan förbättrats.

2.1 SDR i sitt sammanhang

I detta avsnitt försöker vi se på SDR i sitt sammanhang. Först betraktar vi det inifrån ett infrastruktur- och informationssystemsperspektiv. Sedan vänder vi blicken utåt och tittar på relaterade ansatser och hur SDR förhåller sig till dem.

SDR kan beskrivas både som en infrastruktur och ett informationssystem. Vi tittar på vad SDR har plockat från vardera samt på vad som gör SDR till en hybrid mellan dessa.

Infrastrukturer och SDR

SDR har utvecklats mot en distribuerad lösning och är uppbyggd som ett Peer-to-Peer (P2P) system för att öka skalbarheten. P2P-system har som främsta syfte att möjliggöra fildelning. I P2P-system tillåts noderna (peers) ansluta sig samt falla bort på ett dynamiskt vis. För filsökning brukar någon enkel nycelordsbaserad sökform användas. Därtill är det vanligt för noder inom samma system att använda samma strategier för att dela data. Strukturen och möjligheten att hitta data är strikt beroende av att noder inte deltar i systemet med intentionen att förstöra meddelandeföringen. Vi talar därmed om ett homogent men distribuerat P2P-system när vi ser på SDR. SDR nyttjar strukturen hos ett homogent distribuerat system för att fullgöra semantisk indexering samt hantera noder. En viktig del av infrastrukturer är abstraktionslager eller så kallade "middleware". Dessa är skapade för att underlätta datamanipulering. Detta innebär ofta en abstraktion av datarepresentation och transparens av datalokalitet¹. Vilket exempelvis kan användas för att hantera semantisk data på dess lägsta nivå i form av beskrivandelogik². Abstraktionslagret gör att SDR-infrastrukturen är helt transparent för användaren.

¹ Transparens av datalokalitet innebär att användaren är ej medveten om var datat är beläget fysiskt.

² Beskrivandelogik är ett relativt ungt forskningsområde, på engelska "description logic". För den intresserade läsaren se avsnitt 3.5 samt [2].

Informationssystem och SDR

Det andra sättet att se på SDR är som ett informationssystem. Vi ser på informationssystem som möjliggörare för datautbyte och tillhandahållare av strukturer för att representera och finna information. Informationssystemens roller är inte helt tydliga och i och med att språken som används för data-beskrivningar genomgår hierarkisk utveckling (exemplifierat med utvecklingen från XML till OWL) gör även stödsystemen det. Grovt sett kan funktionaliteten hos informationssystem delas upp i filsystem- och indextjänster. SDR har inslag av båda.

Filsystemtjänster: används traditionellt i syfte att tillhandahålla lokal och direkt dataåtkomst. Eftersom filorganisationen kan omstruktureras godtyckligt hålls konsistensnivån³ till enstaka filer, d.v.s. användaren är själv skyldig att hålla konsistensnivån mellan filer, tänk specifik katalogstruktur eller namnkonvention. För enskilda filer hålls konsistens genom att relatera block⁴, man journalför operationer på blocken och man har fragmenteringshantering. SDR:s resurser kan betraktas som filer.

Indextjänster: används för uppslag av data vilken returnerar annan data. En indextjänst kan betraktas som en databas, vilket i viss mån även gäller för filsystem. Den stora skillnaden är att indextjänstens data enbart kan nås genom att ställa frågor och invänta svar. Vilket beror på att indextjänsten använder sig av ytterliggare konsistensregler för lagrad data. Resurser och dess tillhörande data i SDR, se avsnitt 3.4, läggs till och hämtas från systemet genom att göra förfrågningar. Dessa operationer kräver konsistenshållning.

2.1.1 Relaterade ansatser

SDR agerar som ett abstraktionslager där funktionen har gått mot en hybrid-lösning mellan filsystem och indextjänst. Denna utveckling har skett baserat på kraven som ställts på systemet. Relaterade tekniker hamnar i gränslandet mellan dessa huvudgrupper.

Freenet

Freenet⁵ är ett distribuerat anonymiseringsverktyg för datalagring. Precis som hos SDR ligger inte ansvaret för en enskild fil hos en specifik nod utan den fragmenteras för att sedan distribueras och replikeras till övriga noder.

³Konsistensnivå, den nivå för vilken konsistens skall hållas. Konsistens uppnås vid analys av delkomponenter (i detta fall filfragment), där dessa återfinns i ett korrekt tillstånd. Olika delkomponenter kan innehålla tillstånd för vilka alla är ansedda att vara korrekta, för dessa fall krävs konsistensregler.

⁴Med block menar vi block av data s.k. filfragment.

⁵<http://freenetproject.org/>

Freenet är dock mer generellt än klassiska P2P-system då den även tillhandahåller funktion för tjänster som exempelvis publikation av hemsidor, forum och meddelandesändning. Skillnaden mot SDR är att det inte finns en säkerhetsarkitektur, förutom anonymisering, samt avsaknaden av semantiska sökformer.

GNUnet

GNUnet⁶ är utvecklat i syfte att dela information anonymt men är till skillnad från Freenet helt inriktat på fildelning. Sökningsfunktionaliteten är nyckelordsbaserad och filer publiceras i GNUnet-systemet men delas sedan ut från sin lokala plats. Skillnaderna är att SDR tillhandahåller semantisk sökning samt att användare i SDR alltid är autentiserade samt auktoriserade när de nyttjar någon annans resurs.

Hadoop

Hadoop⁷ är ett distribuerat filsystem där data distribueras och replikeras över flera maskiner. Vid sökning filtreras och sammanförs datafragment till en fil. SDR vidgar dock konceptet distribuerat filsystem genom att kräva att resurser beskrivs semantiskt.

HBase

Hbase⁸ är ett virtuellt tabellhanteringssystem som bygger på Hadoop. Det kan användas för att bygga en distribuerad databas där ansvaret för nyckelrymden för en tabell delas över flera noder. Varken Hadoop eller HBase fanns när SDR designades. Man hade annars kunnat tänka sig att använda denna teknik. Istället används idag en distribuerad hashtabell⁹ för att dela ansvaret för fildelar och nycklar.

Pig

Pig¹⁰ är också ett system som bygger på Hadoop. Det tillhandahåller distribuerade databasfunktioner liknande HBase. Pig använder sig av ett eget språk PigLatin, vilket liknar SQL (ett globalt och spritt standardiserat databasfrågespråk). Pig skulle kunna användas för att stödja semantiska datauppslag men det skulle dock inte kunna stödja en mer flexibel datamodell, d.v.s stöd för flera ontologier och härledning av semantisk data, vilket SDR gör.

⁶<http://gnunet.org/>

⁷<http://hadoop.apache.org/core/>

⁸<http://hadoop.apache.org/hbase/>

⁹En hashtabell är en datastruktur med mycket snabb indexering. I kapitel 3 förklaras hashtabeller närmre.

¹⁰<http://hadoop.apache.org/pig/>

SDB

SDB¹¹ (tycks ej vara en akronym) är en semantisk frågemotor som kompilerar frågor för att matchas mot en optimerad databasstruktur. Detta innebär även att den kan användas för att skapa databasscheman ifrån ontologier. SDB skall enligt påstående vara skalbar. SDB skulle kunna användas i SDR men ej tillsammans med den distribuerade hashtabell som för närvarande används.

2.2 Användningsområden

I detta avsnitt vill vi ge läsaren en inblick i hur SDR skulle kunna användas samt peka på vilka behov och funktionalitet SDR skulle kunna täcka och bidra med. Vi tar upp fördelarna med semantisk sökning samt pekar på två försvarsmaktssponsrade projekt med behov som ligger inom SDR:s visionsområde. Inom Försvarsmakten existerar det fler användningsområden där SDR:s teknikstomme skulle kunna bidra med mervärde, men de två som vi tar upp här innehåller exempel på representativa problem som SDR skulle kunna användas för att lösa.

I kapitel 4 beskriver vi de faktiska samarbeten och SDR-tillämpningar som har bedrivits inom projektverksamheten.

2.2.1 Fördelarna med semantisk sökning

Sökfunktionen, som är en av de mest centrala delarna i SDR, är gjord framför allt med tanke på att man vill söka efter flera komponenter som ska passa ihop. Semantisk sökning innebär en möjlighet för användaren att söka på koncept, egenskaper och resursers relationer till andra resurser, vilket underlättar och effektiviserar användningen av resurserna. I normala fall när en användare gör en sökning med t.ex. en vanlig sökmotor eller i en wiki är det oftast ett specifikt ämne som man är intresserad av. Dessa sökningar görs oftast baserade på nyckelord och kräver en viss förkunskap om domänen. Resursen (informationen) som eftersöks är i sin tur oftast taggad med nyckelord som matchas mot sökningens nyckelord. Svårigheten uppstår när man vill göra mer komplexa sökningar, exempelvis i situationer som kräver flera samverkande komponenter av olika typ. Nedan följer två exempel på sådana situationer.

- Flera simuleringskomponenter som går att exekvera tillsammans eftersöks: "Alla simuleringsmodeller som är kognitiva pilotmodeller, skrivna i Java och som har använts med flygplansmodeller av typen JAS39 Gripen".
- En grupp som skall utföra ett uppdrag behöver sättas samman: "Alla personer med kunskap om XX, YY eller ZZ, som vid något tillfälle har samarbetat och som är tillgängliga för uppdrag nu".

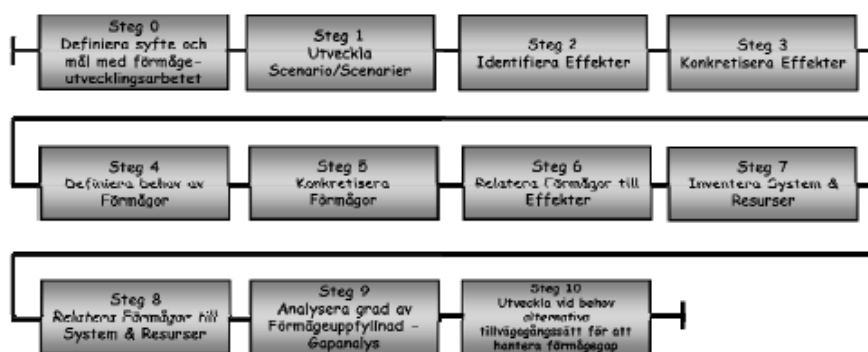
¹¹<http://jena.sourceforge.net/SDB/>

2.2.2 Modellbaserad förmågeutveckling

Modellbaserad förmågeutveckling (MF) är ett angreppssätt för förmågeutveckling inom Försvarsmakten. Metoden är baserad på metodik från områden som Simulation Based Acquisition, M&S, Enterprise Architecture, Enterprise Architecture Frameworks (MODAF), ISO/IEC 15288 Systems Engineering Life Cycle Processes, Effect Based Approach to Operations (EBAO) samt H Mål. För mer information om MF, se [1].

Utgångspunkten är att förmågeutveckling ska leda till att uppnå en eller flera effekter som i sin tur leder till att man når en målbild (eller ett s.k. End State, enligt EBAO). Med effekt avser man resultatet av en handling (ett agerande), där resultatet kan vara en beteende- eller tillståndsförändring hos något system. För att uppnå den önskade effekten behöver vissa handlingar *kunna* utföras, även kallat att ha en förmåga. Dessa förmågor tillhandahålls av system. I modellbaserad förmågeutveckling består systemen i sin tur av olika resurser som delas in i personal, materiel, processer samt information.

Metoden för modellbaserad förmågeutveckling består av 11 steg som kan utföras iterativt, se figur 2.1. I dessa steg är det framför allt intressant att titta på de senare stegen, d.v.s. steg 7 och framåt: (steg 7) inventera system & resurser, (steg 8) relatera förmågor till system & resurser, (steg 9) analysera grad av förmågeuppfyllnad - gapanalys. Det som sker i dessa steg är att man inventerar relevanta tillgängliga eller planerade resurser (system) samt matchar hur väl dessa resurser överensstämmer med de förmågebehov man har. Dessa metodsteg skulle kunna lösas med hjälp av SDR.



Figur 2.1: Bild tagen från rapporten [1] som illustrerar de 11 metodstegen.

Givet att man beskriver resurser, anger status på dem (ex. är tillgänglig, under utveckling resp. är planerad) samt lagrar dessa resursbeskrivningar i SDR skulle man kunna formulera semantiska frågor för att söka efter de resurser som uppfyller vissa förmågor och även passar ihop. Man skulle även snabbt kunna

få fram svar på vilka resurstyper som saknas och därmed även vilka förmågor som kan leveras eller ej.

Sett till de tidigare stegen i processen är det även värt att notera att det är mycket dokument som produceras. Dessa skulle med fördel kunna lagras och göras sökbara i SDR. Vid nya förmågeutvecklingsprojekt skulle då det gamla framtagna materialet finnas lätt tillgängligt och erfarenheter enklare kunna återanvändas.

2.2.3 STIL

STIL (ett ramverk för Samordnade Tjänster för Information och Ledning) är en ansats för samverkan i samhället. Det är resultatet av ett samarbete mellan Krisberedskapsmyndigheten och Försvarmakten. I den så kallade "gula boken" beskriver man en vision som är väldigt lik SDR:s:

"Organisationer och myndigheter blir alltmer beroende av att kunna utnyttja och utbyta information med hög tillgänglighet, säkerhet och effektivitet. I samhället finns mycket att vinna på att stödja utbyte av informationsresurser från flera aktörer anpassat efter kända, men också helt nya icke förutsägbara behov." [14]

I den gula boken beskrivs översiktligt förutsättningar, grundprinciper och teorier kring hur man kan bygga upp ett ramverk för att samordna tjänster för lednings- och beslutsstöd. Man tar även upp, utan att gå in på djupet, den struktur som samverkan kan byggas på samt frågeställningar såsom säkerhetsaspekter. På de flesta punkter kan SDR påstås tillfredsställa kraven för informationsinfrastrukturen som ligger till grund för STIL-ramverket.

2.2.4 Cloud Computing

Cloud computing är ett populärt begrepp myntat för att beskriva användandet av avlägsna tjänster utan visshet om dess interna struktur. I Cloud Computing kan tjänster begäras utan kunskap om dess lokalitet, hur den kontrolleras. Exempel på en sådan tjänst är datalagring.

Detta synsätt på infrastruktur och tjänster stämmer in mycket bra på SDR. Användare och noder som deltar är inte intresserad av antalet deltagande noder utan enbart av huruvida data finns eller ej. SDR klarar av att operera självständigt, t.ex. när noder tillkommer eller faller ifrån distribueras och kopieras (replikeras) datan helt anpassat efter antalet kvarvarande noder. På grund av detta är resursernas tillgänglighet inte beroende av var deras data lagras och följer därför principerna kring cloud computing.

3 Arkitektur

Detta kapitel ger en teknisk översikt av arkitekturen och är ämnad för läsaren som är intresserad av att veta hur SDR är uppbyggt och hur viktiga SDR-funktioner fungerar.

SDR hämtar mycket av sitt beteende från det faktum att den underliggande datamodellen är en distribuerad hashtabell. För läsaren som inte är bekant med hashtabeller gör vi här en kort genomgång av vad det är.

Hashtabell är en datastruktur som har som yttersta syfte att indexera data. För detta används en så kallad hashfunktion som omvandlar datan till nyckelform. Specifikt för nyckeln är att dess längd är betydligt kortare än datan, annars kunde datan användas direkt. Då nyckelintervallet är mycket mindre än dataintervallet kan det uppstå kollisioner när två olika datamängder omvandlas till samma nyckel, även känt som *postfacks-principen*. Kollisioner är dock ett hanterbart problem. Det som är trevligt med hashtabellen är att man erhåller ett enkelt och snabbt sätt att hitta och hämta data.

En distribuerad hashtabell (DHT) delar upp ansvaret för nyckelintervallet på deltagande noder. Hashfunktion och intervalllängd är fixvärden för "ringen"¹. Många kända P2P-system² är baserade på denna typ av indexering.

Som tidigare beskrivits, se avsnitt 2.1, kan man se SDR som ett P2P-serversystem där nätverkstopologin definieras av en distribuerad hashtabell. Varje enskild organisation kan delta med en eller flera noder på, med avseende till nätverket, identiska villkor. Rent logiskt delas inte organisationer upp i grupper utan alla ligger platt runt samma "ring". Detta är inte obligatoriskt utan organisationer skulle kunna, rent teoretiskt, delas upp i gruppintervall, vilket bl.a. skulle kunna vara intressant för att tillhandahålla datalokalitet (med detta menas att data får grupptillhörighet och även kan antas, då det är organisationsbaserat, begränsas rent geografiskt). SDR liknar DNS³, men uppfyller betydligt hårdare robusthetskrav, mer om detta i avsnitt 3.2.

Den nuvarande lösningen att använda SDR som ett distribuerat serversystem är inget egentligt krav utan en anpassning till arvssystem. Utmärkande för SDR är dess resursorienterade natur där vi med "resurs" talar om en utökning av filkonceptet. Resurser beskriver alltid något men resursen i sig kan även, precis som en fil, innehålla data. För att göra beskrivningen av designen och arkitekturen tydligare delar vi in kapitlet i följande avsnitt: funktioner, säkerhet, systemkomponenter, datamodell och semantisk sökning.

¹Vi säger "ring" för att betona att nyckelrymden är begränsad och bearbetas cirkulärt och inte taxonomiskt.

²Skype, eMule, YaCy, I²P, BitTorrent, m.fl.

³DNS är en förkortning för *Domain Name System* eller på svenska domännamssystemet. Det är ett system som förenklar adressering av datorer på IP-nätverk (som till exempel Internet). Systemet översätter en textbaserad adress till ett IP-nummer (datorns adress) och tvärtom, men är inte begränsad till denna typ av uppslagning.

3.1 Funktioner

Konsumenterna av SDR:s tjänster kommunicerar med SDR-noderna genom ett webbtjänst-gränssnitt. SDR tillhandahåller resursrelaterade funktioner så som:

Autentisering: Alla användare loggar in på systemet och autentiseras med hjälp av certifikat. Alla resurser kan identifieras med hjälp av en ägare (som kan vara en person, roll eller grupp)⁴.

Auktorisering: Kontrollerar att någon har rättigheter att utföra en operation på en resurs⁵.

Modifiera beskrivning: Ändrar beskrivningen för en specifik resurs

Modifiera data: Ändrar datainnehållet för en specifik resurs

Radera resurs: All resursrelaterad data förstörs hos ansvariga noder

Skapa/modifiera access-roll-lista: Skapar/ändrar access för de resurser som nyttjar aktuell accesskontrollista och användargrupplista (på engelska *User Group List* (UGL)).

Skapa/modifiera ontologi: Skapar eller förändrar semantisk information som deltagande noder kan använda för att beskriva sina resurser

Skapa resurs: Data skapas, sprids och replikeras till ansvariga noder

Sökning: Förmåga att hitta all inlagd data

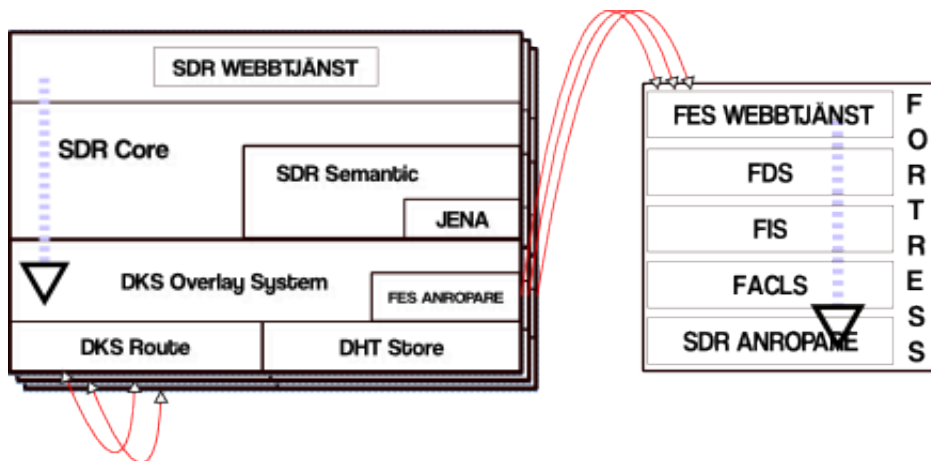
3.2 Systemkomponenter

SDR har delats upp i en rad viktiga delkomponenter vars gemensamma funktion utgör "SDR". Målet har varit att dessa konceptmässigt skall fungera fristående för att underlätta diskussion samt förenkla utbyte av komponenter. Hela infrastrukturen återfinns visualiserad i fig 3.1 och för att ytterligare klargöra systemet beskriver vi kortfattat några av de stödsystem som förekommer i bilden.

Fortress: Fortress är, som kommer att beskrivas i avsnitt 3.3, den tjänstesamling som hanterar accessrättigheter och verifierar identiteter.

⁴ Autentisering sker genom att användarens lösenord hashas och krypteras med sin privata nyckel, som SDR *Certificate Authority* (CA) har utfärdat. CA är den enhet som utfärdar digitala certifikat åt andra. Ett certifikat är ett dokument som intygar att någon är den som den utger sig för att vara. Utfärdaren är någon som certifikatsgranskaren litar på.

⁵ Auktorisering sker genom uppslag och analys av *accesskontrollistan* (på engelska *Access Control List* (ACL)). ACL är en lista med rättigheter kopplade till ett visst objekt. Denna lista kan gälla för en person, en grupp eller en roll samt gälla för olika typer av operationer såsom läsa, skriva, exekvera, o.s.v.



Figur 3.1: Figurens båda huvuddelar avbildar de ingående komponenterna i dels SDR-noden och dels säkerhetssystemet Fortress. Kommunikation initierad av användare via webbtjänster sker i nedåtgående riktning (lila pilar). Kommunikation initierad av infrastrukturen är utritad med röda pilar. Kommunikation mellan noder sker i de undre lagren medan kommunikation med klienter sker i de övre lagren. Alla resursanrop mellan SDR-noder sker transparent och auktoriseras genom "Fortress" (se avsnitt 3.3 för säkerhetsdetaljer).

SDR Core: SDR Core ansvarar för att hålla konsistens hos data samt att denna fasas ut ur systemet när den inte längre används.

SDR Semantic: SDR Semantic är en del av SDR Core. Det är ett system för att bearbeta semantisk data, d.v.s. utföra härledningsarbete och validering av RDF⁶/RDFS⁷/OWL⁸ formaterad data. Semantic är SDR:s modul för att ställa olika typer av frågor. Systemet använder en systemkomponent vid namn "Jena"⁹.

Jena: är en del av "SDR Semantic" och kan ses på som ett mellanlager för att möjliggöra semantisk datamanipulation i en vid mening. Jena är ett verktyg för grafmanipulation vilket fokuserar på dekomposition av beskrivandelogik.

DKS Overlay System: Distributed Kernary System(DKS) är en typ av distribuerad hashtabell med väsentligt mycket bättre robusthet än jämför-

⁶<http://www.w3.org/RDF/>

⁷<http://www.w3.org/TR/rdf-schema/>

⁸<http://www.w3.org/TR/owl-ref/>

⁹<http://jena.hpl.hp.com>

bara tekniker som Chord¹⁰ och Pastry¹¹. Inom SDR används en modifierad kodbas vilken är baserad på SICS (Swedish Institute of Computer Science) implementation "dks"¹².

Vår modifierade DKS står för nätverkskommunikationen inom SDR. SDR:s kommunikation mellan noder sker indirekt genom att skriva och läsa data från den distribuerade hashtabellen. Det är den distribuerade hashtabellen som är ansvarig för:

- Datalagring. Data hashas genom någon global förutbestämd funktion. Varje nod har en nodidentifierare som används för att bestämma vilket nyckelintervall den är ansvarig för.
- Datareplikering. Replikering sker genom intervallskiftning.
- Datamigrering. Migrering genom en naturlig intervallomfördelning.

FES-anroparen: är en del av "DKS Overlay System" och vet hur "FES", ingångstjänsten för "Fortress", skall anropas. Denna används för att rådfråga om auktorisation.

DKS Route: är en del av "DKS Overlay System" och hanterar routning av data.

DHT Store: är en del av "DKS Overlay System" och ansvarar för den långsiktiga lagringen av data. Man kan välja att lagra tabellen som en nod ansvarar för på ett godtyckligt sätt: som en bunt textfiler, i en databas, med eller utan kryptering, osv.

3.3 Säkerhet

För SDR-användarna innebär säkerheten att de måste vara identifierbara (autentiserade) samt inneha rättigheter (auktoriserade) för att kunna hantera resurser och relaterad data. Vid skapandet av en resurs i systemet är användaren skyldig att deklarerat vilken accesskontrollista som skall användas. ACL:er måste därför finnas fördefinierade eller skapas tillsammans med resurserna. På samma sätt krävs en användargrupplista om accesskontrollistan innehåller rättigheter relaterade till grupper eller roller. En och samma ACL kan återanvändas för flera resurser. Eftersom ACL:er lagras som resurser i SDR kan de hämtas direkt med hjälp av sitt namn. En accesskontrollista är en XML-baserad fil där det står vem som får göra vad. Rättigheter kan anges baserat på användare, roll eller grupp. Kopplingen mellan användare, roller och grupper sker i användargrupplistorna.

Centralt för säkerheten i systemet är att alla anrop är tydligt identifierbara samt att alla datauppslag i systemet kräver auktorisation. I varje meddelande

¹⁰<http://pdos.csail.mit.edu/chord/>

¹¹<http://freepastry.org/>

¹²<http://dks.sics.se>

skickas både användarnamn och en verifierbar hemlighet (d.v.s. den är signerad med en personlig nyckel). Denna hemlighet, vilken lagrats i SDR, används därefter för att säkerställa identitet.

Säkerhetsaspekterna av SDR har lösts genom att bygga ut systemet med kompletterande tjänster under det samlade arbetsnamnet "Fortress". Tjänsterna är i sin tur uppdelade i:

FES: Fortress Enforcement Service är ingångspunkten för all kommunikation som rör säkerhetsfunktioner. Denna används lokalt på varje SDR-nod för uppslag kring resursrättigheter. FES använder sig av FDS för att auktorisera anrop.

FDS: Fortress Decision Service bidrar med icke ontologiskt beskriven semantisk härledning kring huruvida en person har accessrättigheter eller ej. FDS frågar i sin tur FIS om information kring anrop, t.ex. användarinformation som eventuell gruppstillhörighet, ACL-namn och vilken policy som skall brukas.

FIS: Fortress Information Service svarar på förfrågningar kring användarinformation och säkerhetspolicy.

FACLS: Fortress ACL Service lagrar och underhåller ACL:er i systemet, denna förlitar sig helt och hållet på SDR. Man krypterar all data innan den lagras med en för Fortress gemensam nyckel.

3.4 Datamodell

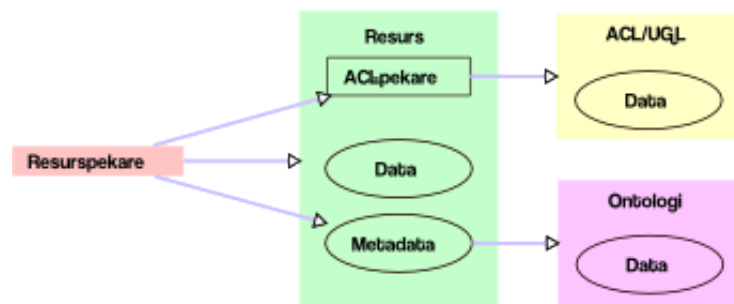
Datamodellen i SDR är till för att möjliggöra semantisk sökning. Den är uppdelad i tre huvudgrupper, Resurser, ACL:er och Ontologier vilka tillsammans innehåller fyra primärtyper. Primärtyperna har konsistenskrav för att systemets representation av resurser inte skall bryta samman. Om konsistenskraven är uppfyllda tillåts användaren manipulera primärtyperna direkt. Primärtyperna visualiseras i figur 3.2 och består av resursdata och metadata, acldata samt ontologidata.

Nedan beskriver vi först primärtyperna och därefter dersa respektiva konsistenskrav.

Resurser

Resurser är en abstraktion för att tala om data och dess tillhörande beskrivning, accessrättigheter, osv. Resurser har enbart ett namn men bär ingen data i sig och är i sin tur uppdelade i data och metadata.

- Data. Med detta menas icke beskrivande data, d.v.s. alla typer av information som användaren är intresserad av att lagra men som ej kan representeras på ett semantiskt sätt.



Figur 3.2: De olika primärtyperna samt deras relationer. Cirkclar betecknar faktisk data som användaren lägger till i systemet.

- Metadata. Metadata är beskrivande data, d.v.s. information en dator själv kan resonera kring, med syfte att på ett konsekvent sätt strukturera kunskap om resursen med hjälp av en ontologi.

ACL:er/UGL:er

Dessa består av roll- och accesslistor för alla resurser. De beskriver vilka grupper, organisationer, som existerar samt hur rättigheter för specifika primärtyper ser ut. En ACL/UGL får läggas till av vem som helst men enbart dess ägare har rätt att radera listan och då enbart om dess borttagande ej bryter mot något konsistenskrav. En ACL/UGL får modifieras antingen av ägaren eller den som har administratörsrättigheter över ACL:en/UGL:en.

Ontologier

Ontologier är domänbeskrivningar som beskriver vilka konstruktioner metadata-bekrivningen tillåts anta. Det är upp till oss människor att gemensamt tolka omvärlden och sammanfatta denna förståelse i en ontologi.

Konsistenskrav

Kraven på datamodellen kan nästan uttydas från figur 3.2:

- Resurser. En resurs är fri att existera utan tillhörande basdata, men den är ej tillåten att existera utan sin beskrivande data (metadata).
- ACL:er. Varken resurser eller ontologier kan existera utan en giltig ACL-pekare. Den data ACL:en pekar ut kommer vara relaterad till en UGL som i sig kräver ytterligare en ACL-beskrivning vilken ej får introducera användarcykler.

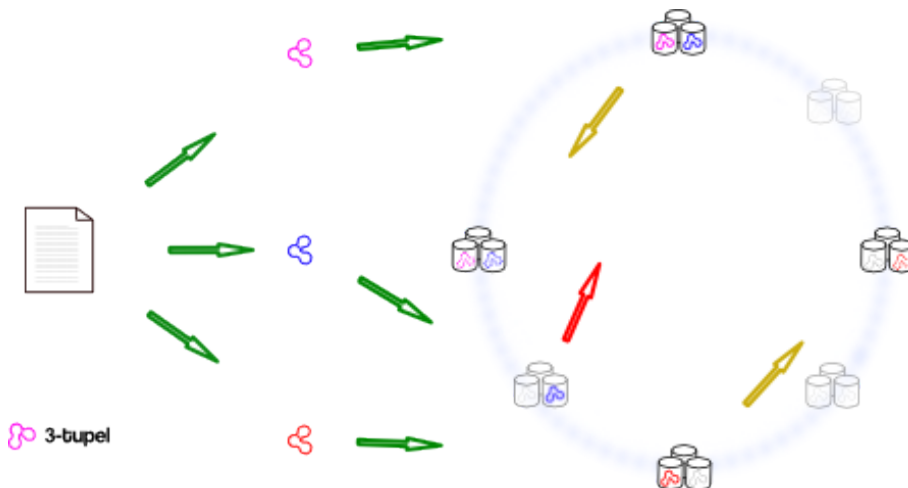
- Ontologier. Minst en ontologi måste finnas tillgänglig för varje given metadatabeskrivning.

Något som i dagsläget inte görs men borde läggas till är att varje datamängd som användaren lagrar hashas för att säkerställa att den inte blivit korrupt.

3.5 Semantisk sökning

För att förstå semantisk sökning i SDR måste man först ha en djupare förståelse för vad vi menar med semantik. Semantiska beskrivningar utgår alltid ifrån koncept och deras inbördes relationer. För att beskriva relationer mellan koncept använder man 3-tupler, alltså data på formen (s, p, o) , där noderna "s" står för subjekt, "p" predikat och "o" objekt. Ett exempel på en sådan 3-tupel är (rosa, äger, bollen). I Jena lagras informationen i strukturer s.k. grafer. 3-tuplerna är inget mindre än byggstenarna för att sätta samman graferna. Man har i syfte att berika konstruktionen, med härledning av nya 3-tupler, skapat "beskrivandelogik", en teori för analys och konstruktion av tabellbaserade semantiska härledningsmetoder.

Det är just utifrån beskrivandelogiken som OWL(Web Ontology Language) har konstruerats. Med detta i bakhuvudet förstår vi att man kan dela upp all OWL-baserad data i 3-tupler och sedan dra slutsatser utifrån den.



Figur 3.3: Figuren visar en resursbeskrivning som delas upp i 3-tupler och distribueras ut på ringen.

Efter att en resursbeskrivning översatts till 3-tupelform distribueras dem därefter ut på SDR-ring. Varje nod får skyldigheten att ansvara för en viss mängd tupler efter bästa förmåga. Detta kan innebära att tupler måste migrera

då noden lämnar ringen. Alternativt kan det betyda att tupler måste replikeras för att den föregående ägaren har kraschat. Detta kan enbart upptäckas om det finns kvarvarande, tidigare replikerade, tupler på ringen. Detta samspel mellan noderna återges i figur 3.3. Grön pil i figuren representerar uppdelningen och distributionen av en OWL-datamängd i tupelform. En gul pil beskriver ett meddelande mellan replikerande noder och en röd pil beskriver migrering av tupler p.g.a. att ägaren bestämt sig för att lämna ringen.

Sökningen är baserad på det faktum att varje fråga är en textuell representation av en graf med 3-tupler där det för varje tupel existerar minst en icke-variabel nod. Den nod, vilken mottagit en sökförfrågan, samlar in de relaterade 3-tuplerna och från dessa skapar en graf. Därefter sker matchning mot den insamlade grafen.

För den intresserade läsaren kan det vara intressant att notera att det existerar ett NP-komplett problem vilket är det generella fallet av det som kallas för subgrafisomorfism.

Vi vill passa på att nämna att alla former av semantiska beskrivningar mynnar ut i någon logikform och att system för att hantera sådan data lokalt, d.v.s. icke distribuerade logikbaserade sökalgoritmer, har funnits sedan Prologs¹³ dagar. I Prolog är dock n-tupelrepresentationen mer generell.

¹³Prolog är ett logikprogrammeringsspråk.

4 Samarbetsprojekt

Ett av målen med årets arbete har varit att vidareutveckla SDR så att det blir anpassat till användarnas behov samtidigt som SDR utvärderas ur ett användarperspektiv. Då användarna teoretiskt sett kan komma från många olika typer av organisationer och därefter ha olika behov och bakgrund var det nödvändigt att ha representativa användarfall med relevanta och trovärdiga användarscenarier. Syftet med dessa scenarier var även att få en plattform från vilken det gick att utvärdera fördelarna och demonstrera användningen av SDR och SDR:s tjänster.

I projektets början fördes samtal med olika intressenter från Försvarsmakten såsom Webb-portalen, ett delprojekt inom SamM&S-projektet, UtvC i Enköping och andra. Av diverse skäl som att de inte hade tid eller att SDR fortfarande var för omoget blev det ingen fortsättning på dessa samtal. Istället fick vi se oss om efter samarbeten på närmare håll, d.v.s. internt på FOI. Detta ledde slutligen till olika grupper av användare, var och en med olika resursbiblioteksbaseade behov. Detta kapitel behandlar dessa samarbeten, deras mål och vilka resultat som uppnåts.

4.1 REVVA och SDR

På FOI pågår sedan 2006 ett projekt om verifiering, validering och ackreditering (VV&A)¹ av simuleringsmodeller kallat REVVA2, vilket finansieras av European Defence Agency (EDA). Projektet löper över tre år och är en fortsättning på en förstudie som genomfördes 2003. Målet är att ta fram en standard för VV&A av simuleringsmodeller. Förutom Sverige (FOI och FMV) deltar Frankrike, Nederländerna, Kanada, Danmark och Storbritannien.

4.1.1 Krav

Under REVVA2-processen genereras och sammanställs ett antal produkter av olika slag som behöver lagras och återanvändas för olika ändamål. För detta krävs ett resursbibliotek och en av uppgifterna i REVVA2 är att definiera krav på ett sådant. I REVVA2 är informationen konceptuellt uppdelad i två huvudkategorier som kan lagras separat: organisationens gemensamma

¹Syftet med VV&A är att identifiera hur användbar en simuleringsmodell är för ett specifikt syfte. Verifiering är en process för att säkerställa att simuleringsmodellen är implementerad på rätt sätt, d.v.s. att den implementerade simuleringsmodellen stämmer överens med kravspecifikationen och den konceptuella modellen. Validering är en analys av om simuleringsmodellen är rätt för sitt syfte, d.v.s. att rätt modell används. Acceptans är en experts utlåtande om till hur stor grad simuleringsmodellen och dess användning ska accepteras. Ackreditering fyller ungefär samma syfte som acceptans men innefattar ett specifikt godkännande av den organisation som ska använda simuleringsmodellen. Processen kring ackreditering ser olika ut i olika organisationer varför acceptans är en mer generell benämning.

information och projektets information. Den förstnämnda innehåller information om simuleringsmodellen och dess utveckling. Detta innefattar information om referenten, d.v.s. det system man vill modellera, samt modeller på konceptuell, formell och exekverbar form. Därtill skall input, output och simuleringsmodellens beteende specificeras. Projektspecifik information avser kunskapsbasen för VV&A-projektet, d.v.s. det som är direkt relaterat till VV&A-arbetet. Detta kan exempelvis vara olika typer av VV&A-specifika produkter samt VV&A-relaterade referenser (metoder, riktlinjer etc.). Generellt är kraven på ett VV&A-resursbibliotek inte mer specifika än för andra tillämpningar. Det som är mer specifikt är själva gränssnittet mot användarna samt den REVVA-specifika funktionaliteten.

Troligen behöver man ett verktyg som stöder VV&A-processen och använder ett resursbibliotek för de produkter som skapas och används. Detta ligger utanför det aktuella arbetets omfång men bör beaktas vid en eventuell fortsättning. För att kunna göra detta krävs också en mer stabiliserad VV&A-metodik. Eftersom projektet pågår fram till 2009 utvecklas och förändras REVVA2 kontinuerligt.

4.1.2 Resultat

Inom SDR gjordes 2007 en analys av hur väl kraven på ett VV&A-resursbibliotek stämmer överens med de tjänster som SDR kan erbjuda. För detta sammanställdes ett antal användarfall som användes som underlag vid diskussioner och arbetsmöten kring frågeställningen, se dessa i rapporten [9]. Det preliminära resultatet av detta arbete var att SDR mer än väl täcker in de behov som behövs för att stödja VV&A processen. Eftersom REVVA2, som tidigare nämnts, är ett projekt under utveckling och dess produkter och processer kontinuerligt förändras får detta ses som ett pågående arbete.

Under året har det utvecklats en miljö där REVVA-användaren m.h.a *Graphical Goal Structure Notation* (GSN), se [3], skall kunna modellera grafer som beskrivs i RDF och sedan hanteras av SDR. GSN är en grafnotation som används för att modellera en argumentationsstruktur, en formell notation vilken genereras från en uppritad bild. Framför allt används det som ett kommunikationsmedel för att beskriva hur ett påstående kan visas vara sant med hjälp av bevis. GSN är ett verktyg som används inom flera discipliner, bl.a. försvaret, kärnkraft-, flyg- och järnvägsområdet, för framför allt riskanalys. Huvudkomponenterna i argumentationsstrukturen består av ett huvudmål, strategier för att nå målet samt nödvändiga delmål.

I miljön som är under utveckling kan REVVA-användaren argumentera för om ett simuleringsmål är möjligt eller inte. Exempel på ett sådant mål är en simulering som har som mål att träna piloter. I appendix A finns exempel på hur GSN-notationen ser ut samt ett exempel på hur en GSN-graf kan se ut.

4.2 Komponentbaserad modellutveckling och SDR

Ett viktigt användningsområde för SDR är som stöd vid utveckling av simuleringar genom komponentbaserad utvecklingsmetodik. Denna utvecklingsmetodik kräver en process där (prefabricerade) simuleringskomponenter, som tillsammans kan utgöra en simulering, identifieras, testas mot varandra (matchas) mot det önskade resultatet (slutsimuleringen), och slutligen sätts ihop. Dessa komponenter kan i många fall vara tillverkade av olika individer och organisationer. För att kunna använda komponenterna bör de vara lagrade på ett sätt som gör dem lättillgängliga och lättidentifierbara för utvecklaren.

Komponentbiblioteket måste tillåta att olika organisationer kan tillgängliggöra sina komponenter och samtidigt kunna utnyttja komponenter utvecklade av andra organisationer. Dessutom bör sökningen vara så exakt som möjligt för att undvika irrelevanta resultat och förenkla processen med att testa komponenterna mot varandra, d.v.s. matcha dem.

Under 2006-2008 utvecklades vid FOI en process för komponentbaserad modellutveckling, där SDR var en viktig beståndsdel. Simuleringarna i denna process beskrivs genom "Simulation Reference Markup Language", som är ett formaliserat språk i syfte att beskriva simuleringar [12]. Komponenterna är paketerade i form av "Base Object Models" [11]. BOM har ursprungligen tagits fram för att möjliggöra uppbyggnad av objektsmodellerna i "High Level Architecture" [8] samt att tillhandahålla återanvändbara modellkomponenter för snabb utveckling av HLA-federater och federationer. BOM är dock generell i sin natur och därmed inte begränsad till HLA, d.v.s. det kan användas för utveckling av alla typer av simuleringar. BOM-konceptet baserar sig på antagandet att simulerings- och federationsbeståndsdelar kan extraheras och återanvändas som byggblock eller komponenter.

Fabriceringen av komponenter börjar med att en beskrivning av slutsimuleringen tas fram. Denna beskrivning bör helst vara formaliserad för att stödja en automatiserad analys av innehållet. Därefter tolkas beskrivningen och en lista av de typer av behövda komponenter tas fram. Utifrån denna lista söker utvecklaren genom lagrade komponenter för att hitta de komponenter vilka bäst passar behovet. De identifierade komponenterna testas sedan mot varandra för att avgöra om de passar ihop både syntaktiskt och semantiskt. Till slut kontrolleras att komponenterna tillsammans levererar det önskade resultatet, d.v.s. det som ingick i beskrivningen av simuleringen.

Denna processen är realiserad i en agentbaserad miljö, där varje agent har ansvar för att exekvera en del av processen, såsom sökning, filtrering, matchning och ihopsättning av komponenter. För att kunna genomföra syntaktisk och semantisk matchning av BOM-komponenter har en regelbaserad metod tagits fram [7]. I metoden förses varje BOM med en ontologisk beskrivning vilken är representerad med OWL (Web Ontology Language) [13]. Denna beskrivning kallas Semantic BOM Attachment, SBA. Reglerna används tillsammans med respektive komponents SBA, i syfte att dra slutsats kring komponenternas ömsesidiga sammansättningsbarhet. För att implementera och logiskt

resonera kring reglerna används en s.k. inferensmotor (inference engine). Reglerna är uppdelade i tre grupper: syntaktiska regler, statisk-semantiska regler och dynamisk-semantiska regler. De syntaktiska reglerna kontrolleras med syntaktisk information såsom namn på händelser och antal parametrar. De statisk-semantiska reglerna används för att jämföra och kontrollera dataklasser, enhetsklasser och användningsområden. Därtill utförs typkontroll, inom den semantiska rymden, för sändare och mottagare i syfte att säkerställa att information tolkas rätt på båda sidor.

De dynamisk-semantiska reglerna avser beteendet hos BOM:ar och kompositioner under körning. Varje BOM innehåller ett tillståndsdigram som beskriver hur komponenten ändrar tillstånd under körning. Genom att använda dynamisk-semantiska regler kan man säkerställa att komponenter i en komposition ändrar tillstånd på ett konsekvent sätt och i enlighet med kompositionens önskade beteende.

Den beskrivna agentmiljön använder SDR som komponentbibliotek och kommunicerar med den genom ett gränssnitt baserat på webbtjänster. Sökningen är semantisk och använder sig av de ontologier, vilka refereras genom komponenternas SBA samt en övergripande ontologi för M&S-området.

4.2.1 Resultat

I samarbetet med SDR-projektet har en miljö för komponentbaserad utveckling tagits fram. Den stödjer dels lagring och sökning av BOM-baserade simuleringskomponenter samt identifiering av ingående SRML komponenterna i en simuleringsbeskrivning. Åtkomsten till denna miljö sker via ett specialanpassat webbbaserat grafiskt användargränssnitt, se figur 4.1. Genom detta kan en användare, vilken önskar att sätta samman en simulering av olika simuleringskomponenter, mata in sitt SRML-dokument², systemet går igenom dokumentet, identifierar de ingående komponenterna samt domänen de tillhör och bygger en lämplig SPARQL-fråga³. I frågan kan man se att vi antar att simuleringskomponenterna är beskrivna i BOM-formatet och att man har använt sig av en BOM-ontologi för att beskriva dem. Om det finns komponenter lagrade i SDR som matchar sökningen kommer en lista med dessa komponenter att presenteras för användaren. Denne kan då välja att ladda ner dem (förutsatt att användaren har rätt att göra så) och gå vidare i komponentmatchningsprocessen.

Resultaten har varit mycket lovande och en jämförelse med vanlig nyckelordssökning är på gång.

²se exempel på en sådan beskrivning i appendix B

³se exempel på en sådan fråga i appendix C



Figur 4.1: Det grafiska webbaserade användargränssnittet för SDR som är specialanpassat för den komponentbaserade utvecklingen

4.3 SM4ALL och SDR

Under förra året deltog SDR-projektet i en projektrelaterad EU-projektsansökan kallad *Smart Homes for All - An Embedded Middleware Platform for Pervasive and Immersive Environments For All*, (SM4All). Ansökan gjordes inom EU-utlysningen ICT-2007.3.7 inriktad mot nätverksbaserade inbyggda kontrollsystem. System med målsättningen att skapa arkitekturer och ramverk som framför allt stödjer sammansättningsbarhet (composability) och skalbarhet, med hjälp av tjänster som tillhandahåller dynamisk konfiguration, ontologier, säkerhet och tillförlitliga tjänster.

Tanken med SM4All-förslaget är framför allt att utveckla en innovativ plattform för smarta och inbyggda tjänster. Detta genom användningen av en miljö vilken använder sammansättning och semantisk interaktion mellan tekniker i syfte att uppnå dynamisk tjänstekonfiguration. En del av plattformen innehåller även ett distribuerat resursbibliotek. Förslaget kommer att appliceras på ett scenario med privata hem och sjukhem, för vård av personer med olika förmågor och behov (unga, gamla, friska och sjuka). Målet är helt enkelt att göra deras vardag lättare.

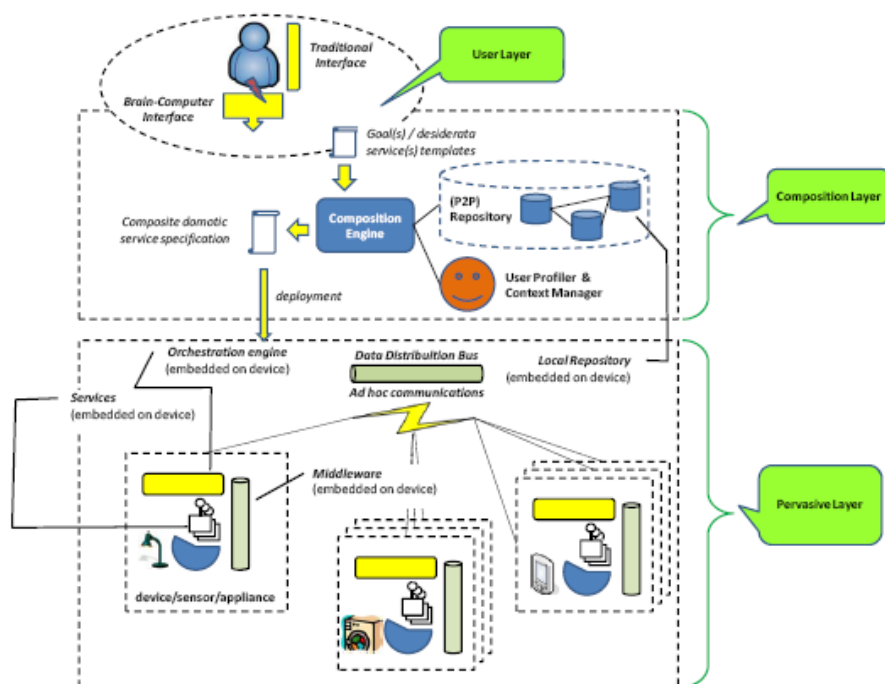
SM4All-konsortiet består av 10 medlemmar från 5 länder inom både industri, myndigheter och högskolor.

4.3.1 Krav

Kraven SM4All-förslaget har på sig är många, men de som är mest relaterade till SDR-projektet och som även är det som FOI kommer att jobba med är de krav som är relaterade till mellanlagret i SM4All-arkitekturen. Mellanlagret är de delar som har hand om resursbibliotek samt tjänstehantering, inklusive tjänstebeskrivning, -sammansättning och -sökning, se figur 4.2.

4.3.2 Resultat

SM4All-projektet är nyligen uppstartat (september 2008) och befinner sig ännu i uppstartsfasen. Vi räknar däremot med att utbytet och resultaten som generas via detta samarbete kommer att ytterligare motivera SDR som en platform för semantisk interoperabilitet.



Figur 4.2: Den logiska vyn av SM4All-arkitekturen

5 Sammanfattning & diskussion

I denna rapport har vi beskrivit SDR-projektet med dess verksamhet, användningsområden, arkitektur, samarbeten och resultat.

Under 2007 och 2008 har SDR-projektet producerat en säker distribuerad infrastruktur för robust semantisk resursdelning. Syftet har varit att stödja insatsförmågor och verksamheter med fokus på resurshantering samt att uppnå högre tillgänglighet, enklare återanvändning och högre utnyttjandegrad av dessa.

Verksamheten har fokuserat på att utveckla infrastrukturen samt att kvalitetsssäkra och sprida resultat medelst samarbeten. Inom projektet har det även publicerats rapporter och artiklar, bland andra årsrapporten [9] och artikeln [10].

I ett försök att påvisa SDR:s skalbarhet och nytta har vi tittat på olika användningsområden. Vi har kommit fram till att SDR är så pass generellt att det går att använda i de flesta situationer där hantering av semantisk information är åtråvärt. Speciellt när man önskar hitta flera heterogena resurser som skall passa ihop t.ex. sätta ihop simuleringar av flera komponenter, sätta ihop grupper med flera förmågor, sätta ihop förmågor av flera system, o.s.v.

SDR kan i grova drag ses som ett P2P-system med semantisk indexering av filer. Hanteringen av semantisk metadata, de inbyggda säkerhetsmekanismerna som "Fortress" tillhandahåller, samt det faktum att SDR är utformat som ett serversystem är de egentliga skillnaderna från klassiska fildelningstjänster såsom Napster och Kazaa. Därtill kan tilläggas att SDR tillhandahåller en betydligt högre robusthet p.g.a. sina förmågor till automatisk replikering och migrering av data.

SDR:s arkitektur tillhandahåller användaren transparenta resurser, vilket får som effekt att varken lokalitet eller säkerhet behöver beaktas. Detta brukar populärt även kallas för *Cloud computing*.

Kvalitetssäkringen av projektets verksamhet har framför allt skett genom samarbeten över projektgränserna. Detta har lett till ett antal prototyper vilka bygger på SDR som en infrastruktur.

5.1 Fortsättning

Under arbetet med SDR har det visat sig att informationshantering är en arbetskrävande och många gånger omständlig process. Därför kommer ett treårigt fortsättningsprojekt "*Effektiv informationshantering av M&S-resurser i ledningssystem*" att startas upp 2009.

Svårigheten har många gånger varit att säkerställa att en resurs hanteras, lagras och görs sökbar på ett effektivt sätt. Speciellt svårt är detta då användaren behöver använda flera olika resurser för att lösa sitt problem.

En lösning på problemet med att hitta rätt resurs har varit att införa me-

tainformation samt att införa ontologier, s.k. domänbeskrivningar som även beskriver relationer och egenskaper mellan olika begrepp. Ontologier har ofta pekats ut som en nödvändig komponent för att kunna åstadkomma semantisk interoperabilitet. Problemet är att det ofta innebär ett stort och tidskrävande arbete att dels skapa en ontologi som är användbar för många, dels tagga upp information med metainformation. Alternativa metoder för att komma runt detta börjar dock dyka upp. Därför har huvudfrågan för fortsättningsprojektet blivit hur M&S-relaterade informationstjänster effektivast och enklast beskrivs, hittas, sammanställs och lagras, d.v.s. hanteras. Projektet kommer även undersöka hur man lättast lägger in information i systemet för att det ska vara så smidigt som möjligt för användaren att hitta. Användaren skall inte tvingas lägga ner en massa tid på att specificera extrainformation vid sökning. Slutligen kommer vi även att titta på hur man lättast formulerar sökfrågor och maximerar kvaliteten på sökträffarna.

Tanken med projektet är att försöka effektivisera och underlätta informationshantering och då speciellt stödja de insatsförmågor som använder tjänstebaserade ledningssystem d.v.s. C 107 (Förmåga att genomföra informations-tjänst), I 306/307 (Förmåga att stödja taktisk/operativ chef med underrättelser som underlag för beslut) samt I 315 (Förmåga till inhämtning via öppna källor -stöd för detta-).

Vidare kommer projektet att verka för att aktivt samarbeta och föra dialog med projekt och användare inom FM (exempelvis FOI-projektet Situations och hotanalys för BG2011), FMKE (exempelvis Knowledge Support-gruppen) och EU (fortsätta på det redan inledda samarbetet med projektet Smart Homes for All - SM4ALL) för att sprida och nyttiggöra de erhållna resultaten.

Projektet kommer framför allt att jämföra, implementera och utvärdera några av de nya informationshanteringsmetoderna med syftet att förbättra och underlätta informationshanteringen. Projektet bygger på resultat från SDR-projektet men kommer inte att ha infrastrukturen som huvudfokus längre.

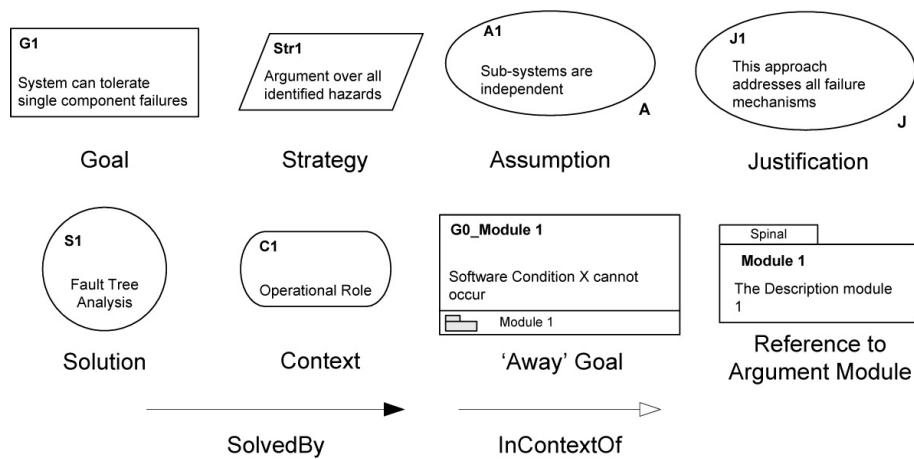
Projektet kommer under de tre planerade åren att leverera utvärdering av metoder för tjänsthantering, ge metodrekommendationer samt ta fram prototypsystem. Detta kommer innebära att projektet behöver undersöka existerande lösningar, dels kommersiella och dels open source lösningar. Om lämpligt kommer även egna metoder att tas fram och utvärderas. För kvalitetssäkringen kommer så långt som möjligt metoder att utvärderas medelst användartester och resultat att publiceras vid internationella konferenser.

6 Appendix

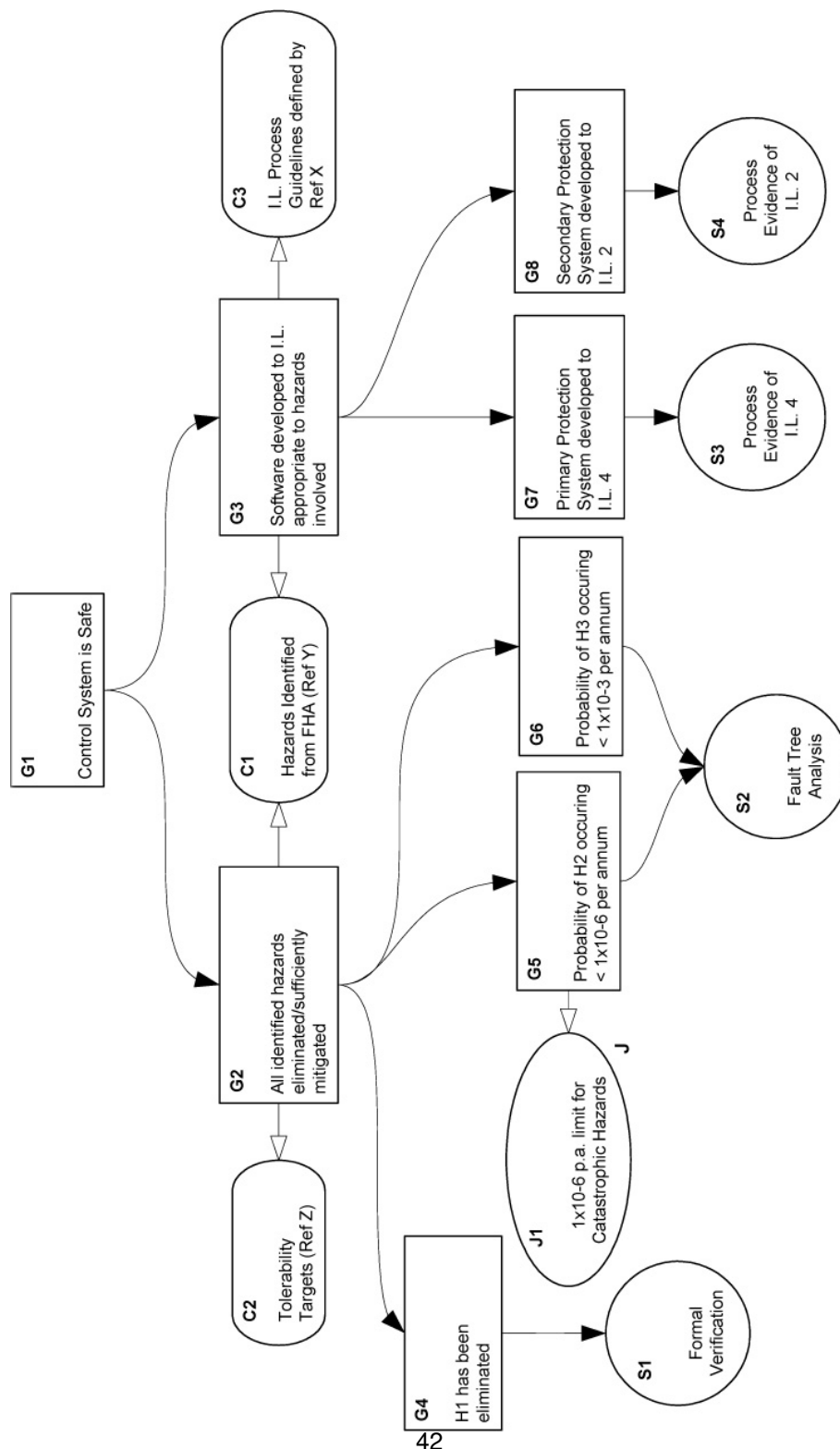
A GSN

I detta appendix visas exempel på hur GSN-notationen kan se ut, se A.1 samt på hur en GSN-graf kan se ut, se A.2. Grafexemplet beskriver hur man kan argumentera för att ett system är säkert.

Bilderna i detta appendix är från <http://safetyengineering.wordpress.com/2008/04/04/the-goal-structuring-notation-gsn/>.



Figur A.1: Exempel på hur notationen för GSN ser ut.



Figur A.2: Ett enkelt exempel på en GSN-graf.

B SRML-exempel

Detta är ett exempel på ett SRML-dokument. Det är en nerbantad version av de SRML-exempeldokument som finns. I denna SRML-beskrivning vill man ha en simulering med två federater, en tank och en truppenhet. I ursprungsversionen hade man även en GPS-federat. Dessa enheter skall samverka och har vissa egenskaper som t.ex. att tanken är av typen Tiger och att truppenheten består av 20 soldater.

```

1 <?xml version="1.0"?>
2 <!--
3 BattleTracking.xml is an example implementation of "SRML
   Light" as defined in
4 the masters thesis "Composability and Reuse in Simulation
   Development using
5 BOMs". This SRML Document models a simulation where a GPS
   system tracks a battle
6 between a tank unit and a troop unit. This is defined as a
   simulation with three
7 major components; a tank unit, a troop unit and one
   positioning system. This
8 file describes the components and their interconnections.
9 It is assumed that the internal logic of each component (such
   as simulation
10 failovers of the GPS system, calculating firing speed of the
   units and how they
11 move around as well as their formations, possible break-downs
   etc) is included
12 in the BOMs as needed, which is why this file only details the
   high-level model
13 of the simulation.
14 -->
15 <Simulation xmlns="urn:x-schema:SRML.xdr">
16 <Script Type="text/javascript" />
17 <!--
18 This class defines a Tank unit, that expresses its position in
   latitude and
19 longitude. The tank type is a "Tiger" and the unit contains 3
   tanks. It should
20 support events that broadcasts its position to components
   interested in it, and
21 being able to fire at the troop unit.
22 -->
23 <ItemClass name="TankUnit">
24 <TankUnit latitude="0" longitude="0" tankCount="3"
   type="Tiger">
25 <Script Type="text/javascript">
26 BroadcastEvent(all, "Position", this.latitude,
   this.longitude);
27 SendEvent(TroopUnit, "Fire");

```

```

28 </Script>
29 </TankUnit>
30 </ItemClass>
31 <!--
32  This class defines a Troop Unit, that is similar to the
    TankUnit. The major
33  differences are that it defines a larger set of soldiers in
    the unit, and does
34  not supply a type.
35 —>
36 <ItemClass name="TroopUnit">
37 <TroopUnit latitude="0" longitude="0" soldierCount="20">
38 <Script Type="text/javascript">
39 BroadcastEvent(all, "Position", this.latitude,
    this.longitude);
40 SendEvent(TankUnit, "Fire");
41 </Script>
42 </TroopUnit>
43 </ItemClass>
44 </Simulation>

```

Listing B.1: Ett nerbantat SRML-dokument baserat på ett exempel med en simulering som innehåller en tankenhet och en truppenhet som skall samverka.

C SPARQL-exempel

Exempel på den SPARQL-fråga som fås vid parsning av SRML-filen i appendix B. Det frågan uttrycker är att man är intresserad av alla simuleringskomponenter som antingen är av tank- eller trupptyp. Dessa komponenter skall sedan uppfylla vissa egenskaper såsom t.ex. att tanken skall vara av typen Tiger och att truppenheten skall innehålla 20 soldater.

```

1 PREFIX ontology: <http://sdr.foi.se/ontology/ontology.owl#>
2 SELECT ?x
3 WHERE { {?x ontology:isBOMContextOf ?x0 .
4 ?x0 ontology:name "TankUnit" .
5 ?x0 ontology:isEntityTypeOf ?x00 .
6 ?x00 ontology:name "latitude" .
7 ?x00 ontology:value "0" .
8 ?x00 ontology:name "longitude" .
9 ?x00 ontology:value "0" .
10 ?x00 ontology:name "tankCount" .
11 ?x00 ontology:value "3" .
12 ?x00 ontology:name "type" .
13 ?x00 ontology:value "Tiger"} UNION
14 {?x ontology:isBOMContextOf ?x1 .
15 ?x1 ontology:name "TroopUnit" .
16 ?x1 ontology:isEntityTypeOf ?x10 .
17 ?x10 ontology:name "latitude" .
18 ?x10 ontology:value "0" .
19 ?x10 ontology:name "longitude" .
20 ?x10 ontology:value "0" .
21 ?x10 ontology:name "soldierCount" .
22 ?x10 ontology:value "20"}}
```

Listing C.1: Den resulterande SPARQL-fråga som fås vid parsning av SRML-dokumentet B.1

Litteraturförteckning

- [1] Front End Strategy AB. Metod för modellbaserad förmågeutveckling. Technical Report 33875/2008, FMV, 2006.
- [2] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.
- [3] Colin Brain. Graphical argumentation handbook. SE Validation Limited, november 2007. ISBN 0-9550107-2-1.
- [4] M. Eklöf, L. Ferrara, M. Garcia Lozano, F. Moradi, D. Nordqvist, and M. Persson. Nätverksbaserad modellering och simulering - prototypen. Användarrapport/User report FOI-R-2144-SE, FOI, 2006.
- [5] M. Eklöf, M. Garcia Lozano, Moradi Farshad, D. Nordqvist, and J. Ulriksson. Nätverksbaserad modellering och simulering - prototypen. Användarrapport/User report FOI-R-1767-SE, FOI, 2006.
- [6] M. Eklöf, M. Garcia Lozano, F. Moradi, D. Nordqvist, M. Sparf, and J. Ulriksson. Nätverksbaserad modellering och simulering - arkitekturen. Metodrapport/Methodology report FOI-R-1439-SE, FOI, 2004.
- [7] et al F. Moradi. A rule-based approach to syntactic and semantic composition of bombs. *Proceedings of the 11th International Symposium on Distributed Simulation and Real Time Applications, DS-RT '07*, October 2007. Crete, Greece.
- [8] <https://www.dmsomil.com>, 2007.
- [9] M. Garcia Lozano, F. Moradi, M. Persson, and P. Svan. Sdr - ett semantiskt baserat resursbibliotek. Användarrapport/User report FOI-R-2349-SE, FOI, 2007.
- [10] Marianela Garcia Lozano, Farshad Moradi, and Rassul Ayani. Sdr: A semantic based distributed repository for simulation models and resources. In *AMS '07: Proceedings of the First Asia International Conference on Modelling & Simulation*, pages 171–176, Washington, DC, USA, 2007. IEEE Computer Society.
- [11] SISO. Guide for base object model implementation (bom) use and implementation. Technical Report SISO-STD-003.0-DRAFT-V0.11, Simulation Interoperability Standards Organization (SISO), 2005.
- [12] <http://www.w3.org/TR/2002/NOTE-SRML-20021218/>, 2007.

- [13] <http://www.w3.org/2004/OWL/>, 2007.
- [14] LEDS CIO. CIO Övlt Mikael Hagenbo, editor. *STIL-ramverk Samordnade Tjänster för Information och Ledning*. Försvarsmakten, stil-9-6 edition, 2007.

