



---

# Independent Review of Emerging Semantic Web Technologies Supporting the Defense Training Environment

## Authors

Mr. Mark Phillips (MASA Group Inc)  
Dr. Barry Smith (University at Buffalo and NCOR)  
Dr. Lowell Vizenor (Alion Science and Technology)  
Mr. Scott Streit (Intervise Inc)

**Version 0.0:** 15 November 2010

## Table of Contents

1	Introduction .....	4
1.1	Problem Statement.....	4
1.1.1	Consequences of not proceeding.....	4
1.2	Vision .....	5
1.2.1	Operational Problem Statement .....	5
1.2.2	A Proposed Solution: Ontology and Web-Based Architecture .....	5
2	An Ontology-Based Strategy .....	5
2.1	The Business Case: Benefits of the Ontology-Based Approach .....	7
2.1.1	Benefits in Cost-Effectiveness .....	8
2.1.2	Benefits in Data-Quality .....	8
2.2	Case Study: The Ontology Strategy as Applied in Biomedicine .....	9
2.3	Ontological Models of Interest.....	10
2.3.1	UCore.....	10
2.3.2	UCore SL .....	11
2.3.3	C2 Core.....	11
2.3.4	aXiom.....	12
2.3.5	Air Force Enterprise Vocabulary Team (EVT) SAF/USM/XC.....	13
2.3.6	Biometrics .....	13
2.3.7	NASA .....	14
2.3.8	NOAA.....	14
2.3.9	NextGen.....	15
3	<b>Software Architecture Industry Best Practice</b> .....	15
3.1	Alternatives Considered .....	15
3.1.1	The PHP Solution .....	15
3.1.2	The .Net Solution .....	16
3.1.3	The Enterprise Java Solution.....	16
3.2	Philosophy of Architecture.....	16
3.2.1	W3C Specifications ahead of JSR Specifications .....	16
3.2.2	JSR Ahead of De Facto Standards .....	17
3.2.3	Compositional Patterns to Create Software Systems .....	17
3.2.4	Use Design Patterns When Creating Custom Code .....	17
4	<b>Business Functions of the Solution</b> .....	18
4.1.1	Software Benefits.....	18
4.1.2	Security Benefits .....	18
4.2	Text Search .....	19
4.3	Materialize .....	19
4.3.1	Performance .....	20
4.4	Case Study: NITRD (Ontology and Service Oriented Architecture) .....	20
5	Governance, Standards and Evaluation.....	21
5.1	Ontology Life-Cycle .....	21

5.2	Ontology Governance.....	21
5.3	Change Management.....	21
5.3.1	Authoritative Data .....	22
5.3.2	Evaluation .....	22
5.4	Education & Training .....	22
5.4.1	Training of Ontologists.....	23
	Sample Topics to be covered in an Ontology Curriculum .....	23
6	Tool & Technology Survey.....	24
6.1	Rationale for the Use of Ontologies .....	24
6.1.1	ANDEM: Architecture Neutral Data Exchange Model (ANDEM).....	26
7	References.....	26

DRAFT

# 1 Introduction

The Department of Defense is working at all levels to rationalize its data management strategy (Stenbit, 2003). However, though this strategy is broad in its application, its reach has thus far not extended to specialized areas of interest such as modeling and simulation.

Over many years DoD modeling and simulation, systems have evolved largely independently of moves towards net-centricity in the wider world of DoD data management. These systems have changed little in their fundamental nature and associated workflow strategies from the way they were prior to land-based command and control and phenomena such as the Semantic Web and the pervasive use of data based on digitized mapping products. The result has been a community that has developed its own standards, methods and technologies to address issues that it felt it was most qualified to resolve.

Now, however, the rapid development of net-centric technologies and methods provides new opportunities for the modeling and simulation community within the DoD and in fact offers opportunities to bring together communities of practice (such as C2, logistics) in ways that can help to bring greater coordination in use of data and systems and enable more rapid configuration of models that can bridge separate domains.

The specific components that we shall address here, and which we believe can bring significant early advantages to the M&S community in its adoption of the net-centric approach, include:

1. Ontologies
2. Strategy for the identification and treatment of authoritative data
3. Service oriented architectures (including Web Services)
4. Semantic search
5. Reasoning and analysis engines
6. Data Interoperability

Each one of these components is of value in its own right. Taken together, however, they will create a platform in whose terms the M&S community of the future will be able to leverage the net-centric data strategy in ways which will enable the creation of a more rapid, more unified, more flexible and more realistic approach to joint defence training in the future.

## 1.1 Problem Statement

United States Joint Forces Command (USJFCOM) required an independent assessment of ontological tools, techniques and methods as they might apply to the future Joint Training Enterprise.

### 1.1.1 Consequences of not proceeding

In what follows we recommend the systematic adoption of ontology-based technology that we believe to be consistent with the M&S software needs of the future. Information systems as a whole are evolving at a rapid pace, and many elements of the existing DoD modeling and simulation infrastructure are marked already by features which are out-of-date. The strategy described in what follows is designed to be maximally future proof. The consequences of not proceeding with this strategy are increased expense, decreased responsiveness to DoD training needs, and cumulating inability to take advantage of the abilities of current and future software and web-based infrastructures.

## **1.2 Vision**

### **1.2.1 Operational Problem Statement**

Our goal in what follows is to provide an operational perspective on the problems that must be solved when creating joint modeling and simulation systems and scenarios in an effective and maximally inexpensive way that will enable agility in development, bring improved interoperability of the systems used by the separate stakeholders, and guarantee a high level of faithfulness to the real world of warfighting operations in order to enable rigorous and realistic collective joint training.

These problems arise because multiple different kinds of warfighter are associated with multiple kinds of data, with no common oversight or common documentation as to data formats, no common modes of access to relevant content, no common frameworks for data retrieval and reasoning, and no generally applicable strategy for combination. To create joint simulation scenarios these different kinds of data must be combined together, and currently such combination requires new manual effort to be applied each time, not only because of differences of data formats and vocabulary, but also because of different kinds of authoritative and non-authoritative data that must be identified and used.

### **1.2.2 A Proposed Solution: Ontology and Web-Based Architecture**

The solution we propose involves two dimensions:

1. Creation of a consensus-based ontology framework
2. Adoption of industry standard open source architecture

We will deal with each of these in turn.

## **2 An Ontology-Based Strategy**

Every defense-related organization is confronting the problems caused by lack of coordination in presentation and handling of data. One standard solution to these problems, both within the DoD and in other areas of government, science and industry, turns on the creation of consensus-based ontologies, which are controlled structured vocabularies that can be used for consistent presentation of

data in ways that bring about also benefits in the handling of data for example in allowing more effective retrieval and reasoning.

We propose the application of the ontology-based approach to the problems confronted in the attempt to collective M&S scenarios for defence training. Here, our problems are more complex than is standardly the case, since the problems caused by lack of coordination in presenting and handling of data arise not only at the point where data is used for modeling and simulation, but already at the stage of real world operations. To address this complexity, we propose a three-step strategy for ontology creation, following Mandrick, 2010:

First, *develop* a *C2 Ontology (C2O)*, a small consensus-based controlled vocabulary to serve as the basis for the description (e.g., tagging) of C2 data, including the C2 data needed for modeling and simulation purposes. This ontology should be developed using current best practices and standard operating procedures for ontology development, including automatic realization of the net-centric approach since data annotated with an ontology thereby becomes automatically identifiable through the corresponding Uniform Resource Identifiers (URIs). It should rest on a strategy of *maximal realism*: seeking not a *data model*, but a *reality model*. The ontology should be based on military doctrine, using the common terms used by the warfighters themselves. It should draw wherever possible on existing ontology efforts in the C2 and related domains, and strive, where necessary, for consistency with NIEM Core, JC3IEDM, and the UCore and UCore SL initiatives.

FM 2-01.3 Intelligence Preparation of the Battlefield (Battle-Space)	FM 101-5 Staff Organization and Operations	FM 3-0 Operations	FM 5-0 The Operations Planning Process
FM 6-0 Mission Command: Command and Control of Army Forces	FM 3-07 Stability Operations	FM 3-24 Counterinsurgency	FM 6-99.2 US Army Report and Message Formats
FM 1-02 Operational Terms and Graphics	FM 6-01.1 Knowledge Management	Fm 3-05.40 Civil Affairs Operations	FM 3-60 Targeting

Figure 1: Sample of C2-Related Doctrine (from Mandrick, 2010)

Second, to realize a plug-and-play approach, we create *extensions* of C2O to cover specific modeling and simulation domains. (For an example see Figure 2 below.) The suite of exntesions will include a generic M&S ontology(SimC2O), which will consist of those terms of common interest to all M&S endeavors, together with a number of more specific extensions of SimC2O ranging across domains such as Range Training, Order of Battle scenarios and Mission Threads. In synchrony with this, C2 Core Ontology extensions would be created for specific operational domains of interest such as Logistics, Counterinsurgency, or Civil Information Management. The goal here is that each Community of

Interest (COI) with data annotation needs should embrace a single, incremental strategy of synchronized development of extensions of the C2 Core Ontology to meet their terminology needs, with a governance process to ensure change management, coordination, availability of authoritative data sources, and to provide dedicated cross-community training and pilot testing initiatives.

Third, *incentivize* the use of the C2O and its network of extensions. The goal is to create a situation in which C2O is used by all major participants along the C2 data chain, since the creators of data will see the benefits this will bring (discussed in 2.1 below). For this, we must incentivize leaders of the relevant communities also to contribute to the maintenance of the ontology (because the coherence of their own work depends upon its being of high quality, on its including the needed terms, and on its being up-to-date). The assumption is that, as the benefits of the core and extensions approach become manifest, more resources will accrue to the project, including resources devoted to SimC2O.

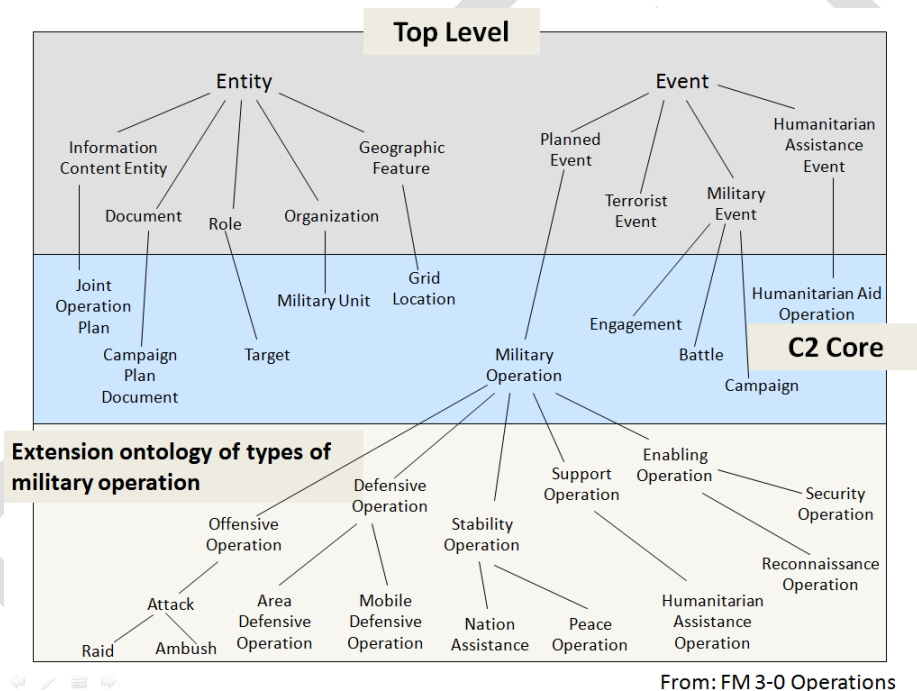


Figure 2: An example of an ontology of military ontologies created according to the Core and Extensions approach

### 2.1 The Business Case: Benefits of the Ontology-Based Approach

In this section we describe the business case for this ontology-based approach. What will the actions recommended in this report provide the DOD? These benefits can be described under three headings.

First, our recommendations will bring benefits in terms of data handling in the initialization phase of scenario preparation.

Second, they will bring benefits in terms of human factors and of data quality, in bringing about a situation where there is more effective coordination of data gathering efforts, and also concomitant benefits in greater realism of simulations.

Third, they will bring benefits in allowing the use of allow application of a rich and growing body of off-the-shelf industry standard state-of-the art software (see Section 3 below).

### **2.1.1 Benefits in Cost-Effectiveness**

The availability of a C2 Core Ontology and of an expanding set of authoritative data sources annotated in its terms will allow M&S specialists to collaborate with ontologists in creating a SimC2 Core Ontology, the common platform for the new approach to M&S scenario development made possible by current net-centric technology, in which the continual need for investment of manual effort in data preparation will be eliminated.

**Enhanced Coordination:** The core-and-extensions strategy will, first, all a more effective coordination of ontology development work across a large population of COIs, and bring also the automatic coordination of applications of the resultant ontologies in the description of data.

**Division of Expertise:** The strategy of orthogonal modules will allow exploitation of the division of expertise on the part of different COIs and subject-matter experts in a way which at the same time ensures consistent interoperation of the whole.

**Training:** The ontology-based approach provides more effective use of resources in the training of those who will be involved in the creation and application of software resources, since the same standard operating processes for ontology development and application and for data use will be adopted in all domains. Personnel can be trained once, and their expertise used multiple times.

### **2.1.2 Benefits in Data-Quality**

**Incremental Strategy for Improvement:** The ontology-based approach provides an incremental strategy for quality improvement of the data flowing from warfighting communities into the M&S information systems. Annotation with common ontologies allows authoritative data to be maintained in ways that make it more easily retrievable, and allow redundancies and gaps to be more easily identified. The problem of data silos (or data cemeteries) is avoided because the ontologies themselves are based on doctrine, are well-disseminated, and are being used at every stage in the operations and M&S data pipeline.

**Enhanced Realism:** Because M&S ontologies are based directly on operations-based ontologies, the approach will bring greater realism of models and simulations.

**Easy Generalizability:** The strategy of core and extensions will allow the strategy to be generalized into new areas to meet new sorts of training needs in the future.

**Enhanced Understandability:** The ontologies provide a cleaner separation of issues of presentation (data models) from issues of meaning (reality models). XML Schema, often used in DoD message standards, is better suited for



specifying the format and structure in which data is exchanged (data model) than specifying the meaning of the data (reality model). This means that the meanings of XML messages are difficult to communicate beyond the communities of origin, and makes their content difficult to aggregate for more general purposes of analysis.

**Net-Centricity:** Since all the ontologies developed will be thoroughly net-aware, and will be made available through industry best practice web services as described in Section 3 below), the strategy of presenting data in terms of these ontologies guarantees an automatic adoption of the net-centric approach.

**Governance:** The ontology-based approach allows for more effective governance of the creation and use of the authoritative data sources formulated in their terms. (See Section 5.2 below.)

**Interoperability:** Above all, the easy combinability of ontologies and of data resources will create for the M&S domain an environment in which plug-and-play modules for different types of joint scenarios can be easily developed and reused without further manual effort. This creates greater flexibility, and a more rapid response in addressing scenario generation needs.

## 2.2 Case Study: The Ontology Strategy as Applied in Biomedicine

The benefits described above are illustrated by the success of the Open Biomedical Ontologies Foundry Initiative (Smith et al., 2007), which addresses an analogous large-scale data integration problem in the field of biomedical research, where multiple model organism species (above all mouse, zebrafish and fly) are used to create genetic counterparts of human disease phenomena, yielding multiple kinds of data which must be generated in ways which allow comparison both between the model organism species and with human beings. Here the problem is: how to integrate data across species and data formats to accomplish complex tasks (e.g. comparing diseases in mouse to counterparts in human)?

The first step of the solution consisted in the creation in 1999 of the Gene Ontology (GO), a common controlled vocabulary to describe (tag) gene and protein data using the same terms across all species and following the strategy of maximal realism. Thus the GO is not a *data model*, but rather a *reality model*, and the GO is correspondingly built out of common terms – such as ‘cell division’ or ‘protein secretion’ – used by the biologists themselves, together with both logical and natural-language definitions.

The second step consists in the creation of multiple coordinated extensions of the GO. These are needed because, while the GO applies to all organism species, it contains only terms relating to normal, healthy organisms in their normal environments. Thus it has no terms for specific kinds of diseases, it has no coverage of experimental processes carried out in the lab or clinic. Taking the GO as foundation, multiple biology groups agreed to develop their ontologies in a synchronized way as extensions from the GO. In addition they created an organization – the OBO (Open Biomedical Ontologies) Foundry – with a

governance process to ensure coordination and to provide dedicated cross-community training and pilot testing initiatives.

At the same time they were able to draw on the considerable success of the GO to *incentivize* the use of this suite of extension ontologies. The GO is now used by thousands of research and industry groups, who have become involved in the maintenance of the ontology. Multiple communities are now contributing to the development of ontologies such as the Protein and Cell ontologies, as well as the Ontology for Biomedical Investigations and the Infectious Disease Ontology as extensions of the GO.

The OBO Foundry ontologies serve as attractors for new users, who see these ontologies as guaranteeing semantic interoperability of their data annotations with the massive installed base of annotations to the GO. Some communities for this reason allow their existing terminologies to die, and agree to use, instead, the more carefully created of the Foundry, and to contribute to their further development. More resources and more expertise thereby accrue to the Foundry project, as the success of coordinated ontology development is demonstrated across a wider range of biological phenomena. The net effect is that the data flowing from model organism experimenters into the information systems of different clinical research groups is incrementally becoming of higher quality, and is becoming more easily integrated together into a single consistent virtual representation of biological and clinical phenomena.

The OBO Foundry ontologies are being adopted also by the biological and clinical modeling and simulation community within the framework of the Virtual Physiological Human ([http://en.wikipedia.org/wiki/Virtual\\_Physiological\\_Human](http://en.wikipedia.org/wiki/Virtual_Physiological_Human)), a multi-billion dollar transnational project to create an ontology-based framework that will allow researchers in human disease to create disparate but integrated computer models of the mechanical, physical and biochemical functions of a living human body.

## 2.3 Ontological Models of Interest

### 2.3.1 UCore

The Universal Core (UCore)<sup>1</sup> is a US Federal Government information sharing initiative that is supported by the US Departments of Defense, Energy, Justice, and Homeland Security, by the Intelligence Community, and by other national and international agencies. The UCore vision is to improve information sharing by defining and exchanging a small number of important, universally understandable concepts across a broad stakeholder base in order to improve the degree of data interoperability between known and unanticipated users while achieving cost and time savings through standardization, modularity, and reuse. In its current form, the UCore 2.0 is well adapted to realizing this strategy of information sharing on the basis of universally understood terms and can serve as a consensus starting point for the construction of successive tiers of more

---

<sup>1</sup> <http://ucore.gov/>.

specific terminologies tailored to the needs of specialist groups of users, or Communities of Interest (COI). COIs can, in turn, create new data models and vocabularies tailored to meet their unique requirements and thus go beyond the narrow set of UCore terms. By providing an evolving resource of common terms UCore 2.0 serves as a central hub designed to maintain a broad community perspective. The long-term goal is that these common terms will create a common reference platform allowing data from diverse COIs to be understood by systems across the DoD, other federal agencies, and perhaps even coalition partners. One limiting factor is that the semantics of UCore lack the logical and ontological expressiveness to validate the consistency of extensions of UCore into lower level domains and subdomains.

### **2.3.2 UCore SL**

The Army Net-Centric Data Strategy (ANCDs)<sup>2</sup> Center of Excellence created UCore SL to supplement the semantics of UCore 2.0. One area in particular where stronger semantics are especially important is in the extension of UCore semantics into domains and COIs. The UCore 2.0 taxonomy is more akin to a controlled vocabulary than an ontology. The UCore 2.0 taxonomy does not include relations with domain and range declarations or disjointness, equivalence, and union axioms. These additional logical resources are necessary to validate, in an automated way, the consistency of domain and COI extensions of UCore. UCore SL [xxx] employs the W3C's Web Ontology Language (OWL) to enable semantic validation of this kind for both individual extensions of UCore as well as the combined set of all extensions. UCore SL offers the entirety of the UCore 2.0 taxonomy (and the relations found in the XML Schema) in a form which satisfies the need of users for enhanced logical resources. It provides for logical decomposition of terms and definitions, the ability to reason logically on the basis of the content of these definitions, and thereby also enhanced support for the creation of consistent extension modules.

### **2.3.3 C2 Core**

C2 Core is a DoD project sponsored by U.S. Joint Forces Command (USJFCOM) and the Office of the Assistant Secretary of Defense/Network and Information Integration (OASD/NII) and is the first major implementation of UCore. The objective of C2 Core is to develop an open standard-supporting, extensible markup language (XML)-based command and control (C2) data exchange. C2 Core is exploring a combined top-down/bottom-up methodology, which both extends semantics down from UCore while also addressing the bottom-up requirements for information exchange brought by specific user groups. The C2 Core follows the same approach as UCore insofar as it attempts to identify a set of terms that is core across the C2 domain. These are terms that are common across C2 that are likely to be used by multiple COIs that fall under the C2 domain. These terms may not be universally understood across all domains, but they should be universally understood relative to the C2 domain. In order to ensure logical consistency is through a top- down extension of UCore

---

<sup>2</sup> Army Net-Centric Data Strategy (ANCDs) <http://data.army.mil/>.

2.0 terms, logically defined using the resources of UCore SL, and applying the result to create a C2 conceptual data model called 'C2 Core CDM', which contains over 200 high-frequency terms that define the C2 domain. These terms pertain to situational awareness, structuring a military organization, planning and assigning tasks, decision making, and assessing progress. Examples of potential targets for extensions of the existing C2 Core include sub-domains such as Strike, Unit Readiness, Planning and Operations, and the Military Decision Making Process (MDMP). Experience in creating UCore SL has yielded a proven process for creating such extensions which results in definitions which are optimized for use both by humans (for teaching and doctrine writing) as well as use by computers (in validation and reasoning).

#### **2.3.4 aXiom**

aXiom is a Space and Naval Warfare (SPAWAR) Systems Center, Charleston, S.C., research and development project that uses a semantic net-centric solution to enable discovery of "self-evident truths" from disparate data sources for improved situational awareness. The aXiom project has leveraged various Defense Department, Department of Homeland Security (DHS) and federal government sponsor needs to explore advanced concepts and technologies relating to semantics and service oriented architecture.

The aXiom Context Technology is based on semantic technologies such as an ontology-based metadata management system, inference engines and a semantic rules engine and deploys an ontology-driven Architecture as the basis of all configuration operations within the aXiom framework. The use of ontologies are also foundation to the following aXiom capabilities: Concept-Data Mapping, User-Driven Concept Model Enhancement, Nonobvious Relationship Discovery, Dynamic Data Discovery, Context-Based Data/Function Security, and Concept-Data Mapping.

All know data services are mapped to the aXiom core ontology, which in turn is linked to other domain- or COI-specific ontologies such as antisubmarine warfare. A semantic search service allows users, then, to search for common concepts across all data sources without having to specify or have foreknowledge of these data sources. Additionally, because the aXiom framework enforces information security in an integrated fashion, users are only allowed to see those elements of data for which they are explicitly authorized.

The aXion core ontology provides:

- A single integration point for new data sources (i.e., they can be integrated into the overall data model simply by mapping their elements to the aXiom core ontology).
- An increase in richness and variety of data without disrupting user interfaces.
- A lower refactor rate for analytical functions or other automated processes that rely on the mapping service.

Unfortunately aXion was not developed in coordination with existing C2Core related ontology efforts, though we believe that such coordination can be established in the future.

### 2.3.5 Air Force Enterprise Vocabulary Team (EVT) SAF/USM/XC

The Air Force EVT was established to support COIs and to work cross-COI issues. The EVT functions and responsibilities are defined in the Community of Interest (COI) Primer Version 3.0 Draft 22

(<https://afkm.wpafb.af.mil/DocView.asp?DocID=6104467>) as follows:

- Provide guidance to COI members on COI standard work and the creation of the deliverables in the vocabulary package.
- Provide technical support to COIs in the formal representation of vocabularies as ontologies.
- Facilitating vocabulary management across COIs to encourage reuse and avoid inconsistencies or other conflicts.
- Perform formal technical review of vocabularies and provide tech/review report to the COI Coordination Panel for consideration, evaluating vocabularies to ensure their completeness, accuracy, and ability to enable net-centric capabilities.
- Provide vocabulary version control and configuration management of vocabulary artifacts.
- Provide technical support to web service developers in proper utilization of formalized vocabularies (ontologies) by web services.

### 2.3.6 Biometrics

Beginning in 2008, the Biometrics Task Force (re-designated the Biometrics Identity Management Agency on March 23, 2010) sponsored an ontology development project involving CUBRC Inc., Symbolic Systems Inc., and the University at Buffalo. The output of the project was a set of three ontologies:

- Upper Biometric Ontology
- Fingerprint Ontology,
- Iris Ontology

which cover the domains of biometric fingerprint and iris characteristic capture, extraction of discriminatory information, and comparison of this information against stored reference data. The ontologies were developed according to the strategy of core and extensions by extending from UCore-SL and the Relation Ontology and by reusing applicable terms from the Information Artifact Ontology (IAO) and Phenotypic Quality Ontologies (PATO). The 3 biometric ontologies, now collectively referred to as the Biometrics Ontology, have since become one of the Biometric Enterprise data architecture products (<http://www.biometrics.dod.mil/CurrentInitiatives/architecture.aspx>).

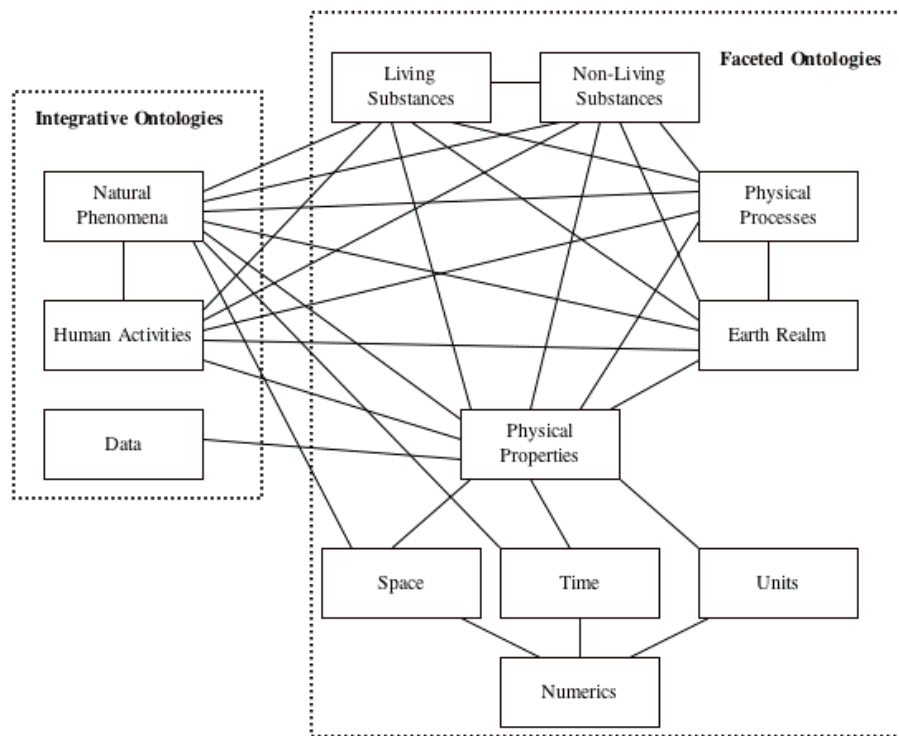


Figure 3: The Core Sweet Ontologies and Their Interrelationships

### 2.3.7 NASA

In the context of the SWEET (Semantic Web for Earth and Environmental Terminology) initiative, NASA has created a highly modular suite of some 200 ontologies incorporating some 6000 terms (<http://sweet.jpl.nasa.gov/>) organized using a modified version of the core and extensions approach as is outlined in 1.2.2 above, with the core ontologies represented in 3. The power of a simple ontology-based approach has been demonstrated by NASA above all in supporting effective information retrieval.

### 2.3.8 NOAA

Ontologies developed by the National Oceanic and Atmospheric Administration (NOAA) include

- ontologies to support the work of the United States Coast Pilot
- ontology resources for marine meteorology
- ontologies to support representation of the data captured in the forms used in the US Ship Arrival and Notification System (SANS)
- ontology resources to be used for markup of maritime documents
- an ontology and XML vocabulary (schema) for the features defined in the IHO Transfer Standard for Digital Hydrographic Data (S-57 standard) and DNC (Digital Nautical Chart).

A selection of ontologies and terminologies for marine domains is made available through the Marine Metadata Initiative (MMI) at:

<http://marinemetadata.org/community/teams/ont/ogcowlharmonization/ontrep>.

Ruttenberg (2009) is a statement of the realist approach to ontology development applied to the marine domain.

### **2.3.9 NextGen**

NextGen is a Congressionally-mandated initiative to modernize the U.S. Air Transportation System in order to increase capacity and reliability, improve safety and security, and minimize the environmental impact of aviation. These improvements to the air transportation system are to be achieved through space-based navigation, digital communications, layered adaptive security, and net-centric information access for operations. Integrated surveillance, the integration of weather data into decision-making, and advanced automation of Air Traffic Management will also be required for the transformation to NextGen.

The mission of the Net-Centric Operations Division (NCOD) of the Joint Planning and Development Office (JPDO) is to manage policies and strategies for information sharing and coordinate investment and development of network-enhancing capabilities. Key attributes for information sharing include: making data discoverable, accessible, understandable, accurate, and timely as well as publishing service-level agreements and establishing a secure, collaborative information-sharing environment.<sup>3</sup> The NCOD is implementing a Semantic Service-Oriented Architecture (SOA) in order to enhance service discoverability, interoperability and understandability through the use of semantic, machine-interoperable service descriptions. Central to this effort is the development of a NextGen Enterprise Ontology that will specify a precise and reusable terminology that will facilitate information sharing across multiple agencies and communities. The long term vision is broad but is being executed in small steps. The NCOD is working with various NextGen stakeholders to identify high-value information exchanges and working with their respective subject matter experts to precisely specify the meaning conveyed in these information exchanges in the NextGen Enterprise Ontology.

## **3 Software Architecture Industry Best Practice**

### **3.1 Alternatives Considered**

Today three basic software paradigms exist for the creation of web based systems: the PHP based solution, the .Net based solution, and the Enterprise Java solution. To better understand these alternative, it is important to describe each technology and its relative benefits.

#### **3.1.1 The PHP Solution**

PHP is a lightweight web language for the processing on web sites. PHP is an interpreted language allowing the rapid development of basic database oriented

---

<sup>3</sup> Department of Defense Net-Centric Data Strategy, May 9, 2003.

web sites. According to Nexen.net, PHP accounts for roughly one-third of all web sites, internationally. PHP runs seamlessly with the Apache Web Server and MySQL. The initial configuration of PHP takes less than 5 minutes to setup. There are over 2 million PHP developers. PHP is best used in a pure web page environment against a structured database. Retrieving, storing and displaying pages is simple and fast. These well-defined applications significantly benefit from the simplicity of PHP.

The major disadvantage of PHP is that it is not intended for extended computer processing. PHP suffers in the area of multi-level security, service-oriented architectures and sophisticated semantic web processing. Semantic web processing requires sophisticated index schemes and dependent libraries that are only available in Java.

### **3.1.2 The .Net Solution**

The .Net solution is a proprietary solution requiring a Microsoft only approach to processing. This violates the basic concept of multiple sources for solution and the fact that software must not dictate an operating system or hardware. The .Net solution is proprietary and is therefore viewed as less than optimal.

### **3.1.3 The Enterprise Java Solution**

Enterprise Java leverages specifications that commenced in 1996. These specifications detail everything from messaging to security to storage. Use of these specifications requires no additional costs and provides most processing out of the box.

According to O'Reilly, Enterprise Java makes up roughly 14 percent of all web solutions. Considering that applications must have critical mass in size to consider an enterprise platform, there is significant market penetration for this solution.

The major disadvantage of Enterprise Java is long setup times for the initial configuration. If PHP, Apache and MySQL take five minutes, and Enterprise Java Solution of JBoss, Java, MySQL and Eclipse takes roughly 16 hours. Fortunately, this cost occurs only once per project.

## **3.2 Philosophy of Architecture**

### **3.2.1 W3C Specifications ahead of JSR Specifications**

We recommend adoption wherever possible of World Wide Web Consortium (W3C) specifications. W3C develops interoperable technologies (specifications, guidelines, software, and tools) to lead the Web to its full potential. W3C is higher level than the Java Specification Requests (JSR) specifications due to the fact that the W3C specifications are language implementation independent.

Most importantly, W3C specifications include the Resource Description Framework (RDF), which allows standardized *representation* of data, and the Web Ontology Language (OWL) standard, which allows inferencing and is built on the basis of RDF.



### 3.2.2 JSR Ahead of De Facto Standards

Java Specification Requests (JSRs) are the actual descriptions of proposed and final specifications for the Java platform. Java adheres to the write once run anywhere paradigm and is routinely used to implement W3C specifications. JSRs may result in containers, application programmer interfaces (API) or remote procedure calls (RPC), among other techniques.

Using our practical example, a remote calling mechanism for accessing RDF may be web services through the Simple Object Access Protocol (SOAP) or JSR 181 or alternately representational state transfer (REST) or JSR 311. Alternately we could use the de facto standard of Apache Axis.

In exploring the problem in more detail, we first consider each technology. JSR 181 defines an annotated Java syntax for programming Web Services.

The principal goal of the specification is to provide a simplified model for web services development that is easy to learn and quick to develop. The specification focuses on enabling the commonly needed forms of web services required for achieving robust, maintainable, and highly interoperable integration. Using this specification, removes the need to hand code web services. Using java annotations, the web services description language (wsdl) is automatically generated and supported on a variety of java platforms. This is design consistent with the write once run anywhere paradigm.

Next, consider JSR 311. JSR 311 develops an API for providing support for RESTful (Representational State Transfer) Web Services in the Java Platform using annotations. Initially, REST was a de facto standard promoted by Google and others. As REST gained acceptance, the JSR 311 represented REST as a write once run anywhere paradigm. REST is uniform resource indicator (URI) oriented, using the hyper text transfer protocol (http) methods, get, post, put and delete. Finally, Axis is the de facto standard for wsdl oriented web services. Whereas, JSR 181 only allows data types supported by java and non-java languages (.net, visual basic, etc), Axis allows more flexibility in the use of data types. The tradeoff for this flexibility is that web services may not run in both the .net and java paradigms.

This examination leads to a conclusion. Since they are virutally for free, applications shall implement web services with JSRs 181 and 311. Since Axis is not part of a specification and may not have long term support, and since standards exist for similar functionality, we remove the consideration of Axis. **De Facto Standards ahead of Custom Development**

An excellent rule for controlling engineering costs is to avoid custom software development. An example of de facto standards includes the google widget toolkit (gwt). Another example, from the past was XDoclet. Both techniques allow the developer to save time in creating software. The gwt and XDoclet techniques are not standards, but provide productivity enhancements to justify their usage. XDoclet was ultimately replaced by annotations in the java software development kit (sdk) 5. A practical example of de facto standards is the use of the Jena api for the creation of RDF systems. This api is extremely well designed and requires very little code to produce RDF implementations. No implementation specification exists at this point and the cost reduction and quality improvement makes Jena a worthwhile choice.

### 3.2.3 Compositional Patterns to Create Software Systems

In Design Patterns, Erich Gamma told us that composition is inherently higher level than inheritance. Using the specification oriented approach, we compose systems by combining specifications. This entire process is composition. A practical example of this type of composition is the use of JSR 181 annotations in conjunction with Jena for an RDF implementation and doing this within the same java class.

### 3.2.4 Use Design Patterns When Creating Custom Code

In software engineering, a design pattern is a general reusable solution to a commonly occurring problem in software design. A design pattern is not a finished design that can be transformed directly into code. It is a description or template for how to solve a problem that can be used in many different situations. Object-oriented design patterns typically show relationships and interactions between classes or objects without specifying the final application classes or objects that are involved. Algorithms are not thought of as design patterns, since they solve computational problems rather than design problems. Since we established software development is an expensive activity, consider lessening the number of development hours by using pre-defined design patterns. Mark Grands book Design Patterns In Java offers 47 unique patterns. Use of these patterns improves the quality of software by using previously vetted interactions.

## **4 Business Functions of the Solution**

### **4.1.1 Software Benefits**

Following best practices in the creation and application of ontologies will facilitate an M&S solution that can rely on software resources that are standards-based, lightweight, scalable, secure, and deterministic. This standards-based software requires minimal development and configuration resources. It makes it possible for the M&S community to leverage existing components, allowing both a services oriented architecture and semantic processor available on day one without software development.

Service-oriented architectures are inherent in a number of products and specification. They allow adoption of a best-of-breed approach, drawing on specifications and tools that already have large acceptance within both the DOD and the international commercial installed base. Moving in the same direction as the rest of industry allows the leveraging of existing personal and technical resources with a minimal learning curve.

This technology agnostic service-oriented architecture uses the two prevailing transport mechanisms and already developed and configured security solutions to provide a multi-level secure suite.

This solution is an analytic tool for retrieval, manipulation and sharing of information. Using a variety of interfaces, a user retrieves relevant data from multiple databases that are either housed locally or externally. Data Analysts review data in a variety of ways including web pages, web services and workflow. Analysts view geospatial data through a map interface.

### **4.1.2 Security Benefits**

A common question asked when working with web architectures is, "How do you secure an entire web architecture?" followed, skeptically, by the statement: "Currently we cannot secure Web services, but we can secure the client web application". The solution is an implementation of an architecture that ensures a secure solution irrespective of the type of access (web browser or web services).

The client access to system requires no knowledge of the target environment. Our solution is elegant and implementation agnostic. The solution has a security

capability that demonstrates how JEE provides confidentiality, authentication and access control. These mechanisms are available to arbitrate access irrespective of the access mechanism, i.e. the web, enterprise beans, or web services.

The following characteristics describe the security architecture:

- The application will run on any web browser, web server, or EJB server
- The smallest configuration changes are required to move from EJB server to EJB Server
- JEE is the host and target environment

The objectives for the security architecture are:

- To maximize the use of definitional constructs while minimizing the amount of written software
- To maximize the portability across all platforms while minimizing software changes
- To maximize the use of generalized constructs that are typical of EJB development
- The same JAAS/Login module is used for two-way SSL from a browser or web service

## 4.2 Text Search

A user performs a text search against all available data sources. These data sources include those available through Web Services. Text searches search for matching values in the database. For example, if a text search is for “Smith,” the results may be for a person with the same surname or a street named “Smith Street.” The results from a text search bring back the **Uniform Resource Identifiers** (URIs) of all documents in which the term is contained.

## 4.3 Materialize

The URIs return the source of the RDF document. The source may be a RDF or non-RDF document stored locally or in a remote location.

For example, a URI may point to a M&S Word Document (.doc) stored in a database located across the network. The URI goes across the network as an HTTPS link. This allows an encrypted data exchange via SSL. The user’s web browser knows how to visualize the document returned based on its MIME type. In this case, the web browser will visualize the .doc file with M&S Word.

Progressively, all data available for use in M&S scenario development is captured in RDF format, using terms from the C2O or from one or its extensions. This will allow more sophisticated searches, since it allows for example use of the relations connecting terms in the ontologies to be built into the search, as in: Send me all documents referring to Baghdad or to one of its suburbsPersist

Persist stores a model in Lucene as well as a triple store. Lucene indexing includes all attributes and an overall document. The triple store persists the model.

### 4.3.1 Performance

The architecture merges combinatorial based retrieval from Lucene and Solr to find chosen Uniform Resource Indicators (URIs). This solutions can retrieve 3,000 resources from 10 Teraabytes of data in less than 250 milliseconds. Upon receipt of the URIs triple stores retrieve models in less than 25 millisecond per models assuming model sizes of less than 100 kilobytes.

### 4.4 Case Study: NITRD (Ontology and Service Oriented Architecture)

Dr. George O. Strawn, Director of the National Coordination Office for the Networking and Information Technology Research and Development Program requested the following:

- 1 A specification-oriented Service-Oriented Architecture for representing, securing and publishing data.
- 2 An ontology-based solution allowing the dynamic inclusion of new ontologies.
- 3 An interchange format for communicating research allocations and outcomes.
- 4 The hosting of the production solution.
- 5 Service level agreements for this externally facing solution.
- 6 Leveraging of Open Source Solutions minimizing product cost.
- 7 Support for all chosen products.

Each of these requests was met with the following **solution**. The details of the "how" for this choice and alternatives considered follows this section.

Objectives 1, 4 and 6 were met by the selection of the following products and architecture. The operating system is Ubuntu 10.04, and the Service-Oriented Architecture is performed by the use of JBoss 5.1.0. JBoss is an implementation of the following used specifications: Java Sever Faces, Enterprise Java Beans, RESTful Web Services, SOAP Web Services, Java Access and Authorization Service. The total product costs for the entire solution was zero. MySQL and lucene provide the storage mechanism for all data, and OpenJena provides the semantic web processing.

Objectives 2 and 3 required the composition of an existing ontology for Funding with the creation of two ontologies. Other organizations accepted the two created ontologies of GrantFunding and Outcome as standards. NITRD agreed to hold and support these two ontologies.

The solution for Objectives 2 and 3 require the use of ontologies. Instantiating the ontology readily provides an interchange format.

Items 5 and 7 are the responsibility of the integration vendor. The integration vendor is a contributor to a variety of open source products and, therefore,

provides a guarantee for the Service Level Agreements and supports all production and development tools. Simply put, the large installed base of these tools creates a healthy support environment.

The result of this work is available at <http://dashboard.nitrd.gov> resulted in a semantic interchange format, three visualizers and a secure web environment, allowing the public to view data in ways never seen before. Costs were significantly minimized and the system was built with change in mind. Query results occur in less than 1 second providing the type of response expected by users.

Additionally, NITRD works as a Good Samaritan in technology hosting two ontologies and promoting the extensibility and use of Open Source.

## 5 Governance, Standards and Evaluation

### 5.1 Ontology Life-Cycle

We can distinguish the following stages in the ontology life-cycle,

- Ontology editing
- Ontology evaluation
- Ontology publishing
- Ontology application and feedback from users
- Ontology maintenance
- Ontology versioning

### 5.2 Ontology Governance

The strategy we propose requires collaborative development of two sets of ontologies, each involving distributed teams of ontologists and software/database developers and subject-matter experts.

### 5.3 Change Management

Each ontology will have its own editorial team. In addition, we propose that there will be two Change Control Boards (CCBs), the first with responsibility for changes in the C2O and in its operations-specific extensions; the second with responsibility for SimC2O and over its M&S-specific extensions. Each CCB will include representatives of the editorial teams and of the relevant stakeholders.

The responsibility of each CCB is to make decisions regarding whether or not proposed changes to the corresponding ontologies should be implemented. For effective change management it will be necessary to define a clear process for dealing with change requests submitted on-line in a timely and transparent fashion. It will be necessary also to determine a clear policy on versioning.

To ensure successful realization of the core plus extensions strategy, the CCB will have responsibility in addition for making decisions concerning changes to the suite of extension ontologies built around the core. Both CCBs will need to develop and apply policies designed to resolve disputes, for example where two COIs claim the authority to develop ontology content for a single domain, or where it is not clear to which extension ontology the terms relating to a given domain should belong.

### **5.3.1 Authoritative Data**

We envision the establishment of an Authoritative Data Committee (ADC), with representatives from both the operations (C2) and M&S (SimC2) domains. The tasks of the ADC will be to determine authoritative data sources for M&S work; to ensure that authoritative data sources are maintained in consistent fashion and in a way that involves effective use of appropriate ontologies and technical (software) resources.

### **5.3.2 Evaluation**

If ontologies and authoritative data sources are to be effective vehicles for ensuring faithfulness of models and simulations to the operational reality of the warfighter, they must be subject to empirically-based methods of evaluation (a) of their adequacy to the corresponding doctrine, (b) of their effectiveness in supporting M&S software. We propose a joint evaluation working group, comprising members from both the CCB and the ADC to address issues of evaluation and to institute pilot testing where necessary. The working group can in addition co-opt persons with expertise for specific domains to provide peer review of corresponding ontology content.

## **5.4 Education & Training**

The increasing use by governments and by scientific and commercial organizations of data management strategies centered on ontology technology is bringing a growing need for ontology expertise and thus for investment in the education and training of ontologists. Unfortunately there are very few formal training opportunities for ontologists, and few formal qualifications in ontology, so that organizations needing to hire ontologists often have difficulties in identifying qualified candidates.

It is already clear that the resultant need for persons with ontology expertise goes far beyond the current availability of appropriately trained personnel. Organizations seeking to hire ontologists often face difficulties in identifying qualified candidates since there is no professional organization that certifies ontologists and very few educational institutions that offer formal education and training in ontology.

In the M&S domain, the establishment of programs for the training of ontologists would at the same time provide additional benefits by helping in the development of a body of knowledge not only concerning the techniques of ontology but also concerning important successes and failures. In this way, it would help those

working in semantic technology and related fields to recognize where ontology can be successfully used, and at the same time to avoid a variety of characteristic errors -- and resultant project failures -- that have affected ontology initiatives in recent years.

#### 5.4.1 Training of Ontologists

To be effective, ontologies need to be created by teams whose members command quite specific skills. Based on the surveys conducted in connection with the 2010 NIST Ontology Summit (<http://ontolog.cim3.net/cgi-bin/wiki.pl?OntologySummit2010>), only one academic program was identified as devoted to education in applied ontology (a Masters program at the University of Buffalo). In addition, we identified some 21 university programs offer at least one ontology-centered course as part of computing programs typically at the masters level. There are also commercial organizations offering training courses designed to familiarize customers with specific ontology-based products. Available training opportunities for professionals do not meet existing needs, above all because they do not cover the human factors elements needed to ensure coordinated ontology development and use, and they do not cover the logical and semantic aspects of ontology technology. Above all, apart from the University at Buffalo program, none of the existing opportunities do justice to the fact that ontology is interdisciplinary.

On the basis of a thorough international survey of ontology expertise needs by government and industrial organizations, the NIST Ontology Summit compiled a model curriculum covering the knowledge and skills that should be developed in an ontology program, some extracts of which are provided in the following table:

<b>Sample Topics to be covered in an Ontology Curriculum</b>
<b>Core Skills (abilities required for developing, improving ontologies, and applying ontologies)</b>
Clarifying the purpose of a given ontology, understanding potential deployment, performing requirements analysis
Analyzing existing legacy models and data that are relevant to a given project
Judging what kinds of ontologies are useful for a given problem (including: knowing when ontologies are not useful)
Preliminaries to ontology development (Identifying, evaluating and using software tools that support ontology development; assembling an ontology from reusable modules)
Managing ontologies across their life cycle requirements analysis and planning managing a systematic update process versioning, documentation, error trackers evaluating and improving ontologies (finding errors via manual term-by-term inspection, solving interoperability problems, decomposing large ontologies into interconnected modules)
Documenting ontologies (e.g., providing natural language definitions and providing concise explanations for axioms)

<p><b>Human Factors</b></p> <ul style="list-style-type: none"> <li>Training people in development and use of ontologies</li> <li>Coordination of ontology development efforts distributed among multiple stakeholder groups</li> </ul>
<p><b>Core Knowledge</b></p>
<p>The basic terminology of ontology (ontologies vs. databases; relation of ontology to conceptual modeling and data modeling)</p>
<p><b>Theoretical foundations</b></p> <ul style="list-style-type: none"> <li>first-order logic, basics of description logic, modal logic, second-order logic, set theory</li> <li>knowledge representation, conceptual modeling, data modeling; metadata</li> </ul>
<p><b>Building and editing ontologies</b></p> <ul style="list-style-type: none"> <li>human aspects (application of classification principles, manual auditing, coordination of ontology modules and core-extension strategies)</li> <li>software tools (Protégé, ...)</li> <li>addressing interoperability problems among ontologies</li> <li>ontology evaluation</li> </ul>
<p><b>Examples of ontologies, illustrating different methodologies</b></p> <ul style="list-style-type: none"> <li>upper-level ontologies (BFO, DOLCE, SUMO, ...)</li> <li>mid-level, domain-spanning ontologies (PSL, ...)</li> <li>domain ontologies (GO, Enterprise Ontology, ...)</li> </ul>
<p><b>Examples of ontology applications (successes and failures)</b></p> <ul style="list-style-type: none"> <li>as controlled vocabularies / standards, to achieve coordination among humans</li> <li>to solve interoperability problems among external data resources</li> <li>reasoning with ontology content</li> <li>improve search and retrieval</li> <li>natural language processing</li> <li>decision support, situational awareness, information fusion, anomaly detection</li> </ul>
<p><b>The Semantic Web, Software and Architecture (sample topics)</b></p> <ul style="list-style-type: none"> <li>Introduction to the Semantic Web</li> <li>URIs and namespaces</li> <li>XML and XMLS Datatypes</li> <li>RDF and RDF/XML</li> <li>OWL, including OWL-2 and the profile EL++.</li> <li>Design patterns used in the creation of Semantic Web Solutions</li> <li>SPARQL</li> <li>Inference and reasoning</li> <li>Graph visualization</li> <li>JBoss</li> <li>Putting it all together: loading and accessing an ontology.</li> </ul>

## 6 Tool & Technology Survey

### 6.1 Rationale for the Use of Ontologies



The strategy for improving the ability of developers of M&S software to integrate data from disparate sources into an internally consistent and properly formatted package is to focus on the data initialization phase. The problem is to integrate as rapidly as possible many different kinds of data in the form of a scenario. Currently, these data cause excess problems for the scenario designer because the data coming from different sources are not employing common formats, vocabularies and so on. They are in addition marked by redundancies which can be resolved only with ad hoc manual effort. As a consequence, current modeling and simulation integration efforts are marked by long and difficult integration periods which can last anywhere from months to years.

The proposed solution is the initiation of a large scale, centrally coordinated effort to bring about the following ends: 1) a common suite of realistic C2 and related ontologies for describing warfighting situations and 2) a governance process which allows this suite of ontologies to evolve and expand in a maximally consistent and useful way. There are a number of ways that the use of a common set of ontologies, maintained by domain experts committed to the acceptance of tested best practices and vetted by a community of authorities in a well-documented governance process, can improve the overall time and cost of modeling and simulation integration efforts:

- *Improved understandability* – The use of ontologies to represent the meaning of data in an external, application neutral format is an improvement over the current state of affairs where the meaning of the data is stored in the heads of developers or buried in application code, both of which make understanding the data time consuming and error prone.
- *Improved reusability* – The strategy of a *reality*-centric approach over an *application*-centric approach means that the ontologies are designed in such a way as to be reusable by a large and varied community of users. Application-centric approaches are difficult to integrate with one another and often lead to additional one-off solutions.
- *Improved extensibility* – The use of a common ontology makes it possible for different efforts to create mission specific extensions of the common ontology—i.e. create new terms when necessary and reuse terms when possible.
- *Improved discoverability* – The use of a common ontology makes it possible for groups to discover and understand the data assets of other groups, thereby reducing the number of redundant efforts.

The use of a common ontology – or better, of a common suite of ontology modules designed for interoperability – along with an effective governance process can bring about a network effect [reference] where the value of the ontology exponentially increases as more people use it. When combined with the employment of open source technologies and practices by other Federal government agencies to establish a legal and technical framework to reduce cost and waste by adopting appropriate open technologies for DOD use, the result could be a web oriented architecture within which data services, tools, and services of importance to data initialization could be made more discoverable, composable, and increasingly re-used.

### 6.1.1 ANDEM: Architecture Neutral Data Exchange Model (ANDEM)

ANDEM is intended to reconcile differences between the various object formats (High Level Architecture (HLA), Test and Training Enabling Architecture (TENA), etc.) in support of the development and execution of mixed-architecture Live, Virtual, and Constructive (LVC) environments. What this means for the war-fighter: the effective and efficient reusing of multiple architecture products regardless of service, component, or development tool. The independent format allows mapping any interoperability architecture DEM to a common language. Once mapped, it will support reuse in multiple interoperability environments. Architecture specific Data Exchange Models can be mapped to each other through the use of gateways to bridge the multiple LVC architectures. However, this approach requires developers who are familiar with the models involved. The challenge is to accelerate and automate as much of the mapping process as possible. The objective of ANDEM is to extract a single data exchange metamodel from the metamodels for TENA, HLA, DIS, and CTIA. The ANDEM metamodel should express the same data exchange capabilities as any TENA, HLA, or DIS object model. The relation between ANDEM metamodel and other metamodels is a one to many relation, instead of a many to many relation--i.e. a point to point mapping from each metamodel.

The developers of ANDEM recognize that the use of ontologies and Semantic Web technology will improve the ability to create, maintain and compose object models. Ontologies will permit and facilitate archiving and maintaining interoperability knowledge that is typically lost, or kept only by original designers.

## 7 References

Mandrick, William C2 Core Ontology Study Report./ Prepared for the U.S. Army CIO G6, 15 December 2010

Smith, B. et al. The OBO Foundry: Coordinated Evolution of Ontologies to Support Biomedical Data Integration, **Nature Biotechnology**, 25 (11), November 2007, 1251 -1255.

Ruttenberg, A. The realist approach to building ontologies for science, [http://www.stateofthesalmon.org/agencypartnerships/downloads/SaIDAWG\\_ppts\\_1109/Ruttenberg-realistapproach.pdf](http://www.stateofthesalmon.org/agencypartnerships/downloads/SaIDAWG_ppts_1109/Ruttenberg-realistapproach.pdf), 2009.

Stenbit J.P. Department of Defense Memorandum, **DOD Net-Centric Data Strategy**, Dated 9 May 2003.