04B0000075

MITRE TECHNICAL REPORT

Netcentric Semantic Linking:
An Approach for Enterprise Semantic Interoperability
Revision 1

October 2004

Mary K. Pulvermacher
Suzette Stoutenburg
Salim Semy

Sponsor:	ESC	**Contract No.:**	FA9721-04-0001
Dept. No.:	D500	**Project No.:**	0304752600

MITRE
Corporate Headquarters
Bedford, Massachusetts

MITRE Department Approval

Mary K. Pulvermacher
Senior Principal Staff

Abstract

As threats around around the world become ever more complex, the DoD must seek ways to minimize the lag time required for synthesizing information and providing the resultant intelligence to the Warfighter. Key to this goal is providing a Command and Control (C2) machine to machine (M2M) environment where rapid and flexible exchange of information with new, and often unanticipated, trading partners is possible.

To enable M2M semantic interoperability in dynamic environments, semantics must be expressed in a manner precise enough for humans and machines to understand them. This entails capturing data and application semantics using a standard language and making implied semantics explicit. Extending the Web to give Web-based information well-defined meaning in a standard way is the vision of the Semantic Web. Key to the vision of the Semantic Web is the ability to capture semantics in ontologies across multiple domains and link these ontologies to interconnect related concepts. One approach for facilitating system interoperability and data integration in a network centric environment is to harness the power of linked ontologies to promote semantic interoperability across the enterprise.

Therefore, our purpose was to explore Network Centric Semantic Linking as a potential solution for integration across the U.S. Military C2 Enterprise. Our one staff-year research effort accomplished three goals. First, we investigated approaches for semantically linking ontologies across military domains using the proposed international standard Web Ontology Language (OWL). Second, we formed opinions on the applicability of semantic linking to the military domain by using this technology to develop a semantic linking test case that addressed an existing mission problem. Finally, we assessed the value of using standard upper ontologies in a military environment. This paper presents the results of our research and gives details on the mission test case we implemented.

Keywords: Semantic Web, semantic linking, semantic mapping, ontology, ontology mapping, standard upper ontology, target validation, interoperability, Netcentric, OWL

Executive Summary

Background

Synthesized information must get to the Warfighter faster through machine to machine (M2M) environments.

As threats around the world become ever more complex, the DoD must seek ways to minimize the lag time required for synthesizing information and providing the resultant intelligence to the Warfighter. Key to this goal is providing a Command and Control (C2) machine to machine (M2M) environment where rapid and flexible exchange of information with new, and often unanticipated, trading partners is possible.

To enable machine to machine semantic interoperability in dynamic environments, data and application semantics must be expressed in a manner precise enough for humans and machines to understand them. This entails capturing semantics using a standard language and making implied semantics explicit. Extending the Web to give Web-based information well-defined meaning in a standard way is the vision of the Semantic Web.

M2M interoperability requires standard semantic specification.

Semantic linking could be a solution to semantic interoperability across domains.

Key to the vision of the Semantic Web is the ability to capture data and application semantics in ontologies across multiple domains and link these ontologies to interconnect related concepts. One approach for facilitating system interoperability and data integration in a network centric environment is to harness the power of linked ontologies.

Therefore, our goal was to explore Network Centric Semantic Linking as a potential solution for integration across the U.S. Military C2 Enterprise. We investigated approaches for semantically linking ontologies across military domains using the proposed international standard Web Ontology Language (OWL). We formed opinions on the applicability of semantic linking to the military domain by developing a semantic linking test case that addressed an existing mission problem. Finally, we assessed the value of using standard upper ontologies in a military environment. This paper presents our results and gives details on the mission test case we implemented.

Study goal was to explore Netcentric Semantic Linking as a potential solution for Integration across the C2 Enterprise.

Mission Focus: Target Validation

We automated portions of the target validation process to show how semantic linking across military domains could meet a real mission need. We built an ontology to capture concepts in the Air Operations Database (AODB) and an ontology to capture concepts in the Military Intelligence Database (MIDB). Each ontology was linked to a separate relational database that simulated representative AODB and MIDB data, respectively. These ontologies were linked using a reference ontology of Geographic Area of Interest concepts. We demonstrated the ability to retrieve information from across multiple domains to address operational needs using a semantic linking approach.

Conclusions

Our experiences on this Netcentric Semantic Linking study led us to the following conclusions.

- Semantic Linking is Powerful but Complex.
    - Semantic Web has power but is difficult.
    - Linking via a Reference Ontology is powerful and extensible.
    - Domain and modeling expertise are critical.
    - Linking ontologies to databases was more challenging than anticipated.
- Better Semantic Web Tools are Needed.
    - Ontologies are powerful but non-trivial.
    - OWL tools are immature.
- Semantic Web is Maturing Rapidly and Warrants DoD Attention.
    - Semantic Web is here to stay and maturing rapidly.
    - Semantic Web will be an integral part of a Netcentric approach in the DoD.
    - Semantic Web training and best practices are needed.
    - Developing ontologies with reuse in mind will pay off in the long-term.

Recommendations

Our recommendations regarding using Semantic Web technologies in a Netcentric approach are directed to MITRE and our government sponsors. These recommendations are listed below.

- Monitor innovations in Semantic Web tools and standards.
- Invest in training on Semantic Web technologies.
- Support development of Ontology Best Practices for the DoD community.
- Consider how to apply Semantic Web technology in the development of future mission capabilities.
- Develop ontologies with reuse in mind.

Acknowledgements

This Netcentric Semantic Linking study was conducted as part of the FY04 MITRE research effort titled “Semantic Integration of the C2 Enterprise” with Principal Investigator Marwan Sabbouh. We gratefully acknowledge the help of the many people who assisted us, but most especially Jeff Johnston and Jack Sexton, who served as our key domain experts. Jeff suggested the target validation mission use case and provided invaluable links and contacts to data sources while Jack tutored us on airspace mission operations. We give our thanks to Tom Buck, Hans Royer, and other Northrup Gumman team members for hosting a Technical Exchange Meeting and sharing valuable insights. We would also like to thank the technical support team at Network Inference for their prompt response to all our pleas for help. Finally, thank you Leo Obrst for selflessly sharing your Semantic Web knowledge, participating in the upper ontology study, and providing a thorough review of this paper

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1 Context

Commercial and government organizations are moving toward greater use of Web technologies, both on the Internet and on controlled intranets. Use of these Web-based networks is leading to unprecedented levels of data exchange. However, being able to exchange bits over a network does not mean that the meaning of the data is understood. Interoperability across systems implies that the data can not only be exchanged, but also correctly interpreted and operated upon. Use of standard Web technologies is decreasing the amount of custom coding needed for system interoperability. For example, the Extensible Markup Language (XML)[1] provides a standard data syntax that eliminates the need to create custom tools to parse the incoming data stream. However, being able to parse and read data does not mean that people or applications understand it. Custom coding and a-priori knowledge is still required to ensure that the data is interpreted correctly.

As the tempo of operations increases, military, and government operators want to decrease the manual effort and associated lag time needed for systems to interoperate. This is critical in an environment where rapid and flexible exchange of information with new, and often unanticipated, trading partners is required. One approach for facilitating dynamic system interoperability is to move toward capturing data and application semantics in a manner precise enough for humans and machines to understand them. This entails capturing semantics using a standard language and making implied semantics explicit (e.g., What does a distance of 27.5 mean? Is it 27.5 miles? Kilometers? Inches?). Extending the Web to give Web-based information well-defined meaning is the vision of the Semantic Web. Key to the vision of a Semantic Web is the ability to capture semantics in ontologies and link these ontologies to interconnect related concepts. One approach for facilitating system interoperability and data integration in a network centric environment is to harness the power of linked ontologies in a Semantic Web.

1.2 Study Purpose

Explore Network Centric Semantic Linking as a potential solution for integration across the U.S. Military Command and Control Enterprise.

This study explored network centric semantic linking as a potential solution for integration across the U.S. Military Command and Control (C2) enterprise. This task grew out of a Fiscal Year (FY)

---

[1] http://www.w3.org/XML/

2003 semantic web research effort[2] that concluded that semantic linking is key to the realization of the Semantic Web and could be a *solution to promoting semantic interoperability across the enterprise.*

This one staff-year study had three goals. The first goal was to investigate approaches for semantic linking of ontologies using the proposed international standard Web Ontology Language (OWL). At the start of this effort OWL was on track for becoming a World Wide Web Consortium (W3C) recommendation. A W3C recommendation is their vernacular for an international standard. Because OWL was emerging as the international standard for data and application semantics, we believed it was critical to gain experience using OWL and develop opinions on recommended approaches for semantic linking using OWL. Therefore, we made it a goal to use OWL and OWL capable tools in our Netcentric Semantic Linking investigation.

Our second goal was to develop a semantic linking test case that spanned military domains. We wished to test our hypothesis that semantic linking could be used as an approach for integration across the U.S. C2 enterprise by actually trying it. Because we believe that the best way to learn is through direct experience, we insisted on actually applying semantic linking to a test case that spanned military domains. The desired impacts of these first two goals were to 1) form opinions on the value and difficulty of semantically linking ontologies in a military context, 2) form these opinions based upon practical knowledge and experience, 3) gain practical experience using OWL and OWL tools, and 4) share our results widely, both within MITRE and with our customers.

Our final goal was to assess the value of using standard upper ontologies in a military domain. Standard upper ontologies are touted as a tool for semantically linking ontologies. However, there is no consensus on the value of using standard upper ontologies to link domain ontologies. The desired impact of this task was to form and share opinions on the value of mapping ontologies to a standard upper ontology within a military context.

1.3 Document Organization

Section 2 provides an overview of the semantic web – what it is and why one should care. An overview of ontology mapping approaches is given is Section 3. We provide a summary of the results of our standard upper ontology investigation and references to obtain more complete results in Section 4. Section 5 describes our mission use case. We describe the target validation use case and how we applied semantic linking to address the problem. We also step through our process and findings in implementing this mission use case. Section 6 on potential semantic linking research extensions is targeted to other Semantic Web researchers. Our conclusions and

---

[2] JBI Agent Based Architecture FY03 Mission Oriented Investigation and Experimentation (MOIE). Further information is available to MITRE employees at http://info.mitre.org/mtp/control?Transaction=org.mitre.mtp.mtpHomepage.ViewProjectHomepageTransaction&mtpNumber=923 or http://employeeshare.mitre.org/p/pulver/transfer/JABA%20Results%20CD/OpenMeFirst.htm

recommendations may be found in Sections 7 and 8, respectively. Finally, Appendix A gives some sample ontology queries including the queries we used to demonstrate Netcentric semantic linking with our mission use case.

2 Semantic Web Background

2.1 What is the Semantic Web

> "The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation."
> -- Tim Berners-Lee, James Hendler, Ora Lassila, The Semantic Web,
> Scientific American, May 2001

n well-defined meaning, better enabling computers and people to work in cooperation" [BHL01]. A key point is that the Semantic Web extends the current World Wide Web incrementally. The current Web is designed to present information to people. The Semantic Web extends the current Web to make web information meaningful to machines (software) by giving it well defined meaning (i.e., semantics). These semantics are made explicit through ontologies that are then exploited by software applications.

So what is an ontology? The term ontology is not new. It originates in philosophy and has been around since the 18th century.[3] What is relatively new is the adoption of ontologies in the Web community, with the corresponding use of Web technologies (Web addressing, universal character set, XML, etc.) as their foundation. A commonly cited definition of ontology in the Web community is "the specification of a conceptualization" [Gru93]. Another useful definition of ontology may be found in [OWL04 pp. 3-4]. In an ontology, concepts and their relationships to other concepts are specified precisely to support machine interpretation. A concept may be thought of as a resource identified by a Uniform Resource Identifier (URI). Resources may either exist on the Web (e.g., a document that may be retrieved) or be represented on the Web (e.g., a person). An ontology captures information about these resources and the relationships between them.

Web ontology languages are founded on a language called Resource Description Framework (RDF)[4] and its subsequent extension called RDF Schema, together referred to as RDF/S. RDF/S represents resources as sets of triples, where each triple consists of either (Resource, Property, Resource) or (Resource, Property, PropertyValue). These triples collectively constitute a graph. A simplified vehicle ontology is shown in Figure 1. OWL[5] is a semantic extension of RDF/S, providing more expressive power. On 10 February 2004, the W3C announced that RDF and OWL were W3C Recommendations, effectively making them Web standards.[6]

---

[3] An interesting history of the term ontology may be found in [GFC04 pp 1-5].

[4] http://www.w3.org/RDF/

[5] http://www.w3.org/2004/OWL/

[6] http://www.w3.org/2004/01/sws-pressrelease

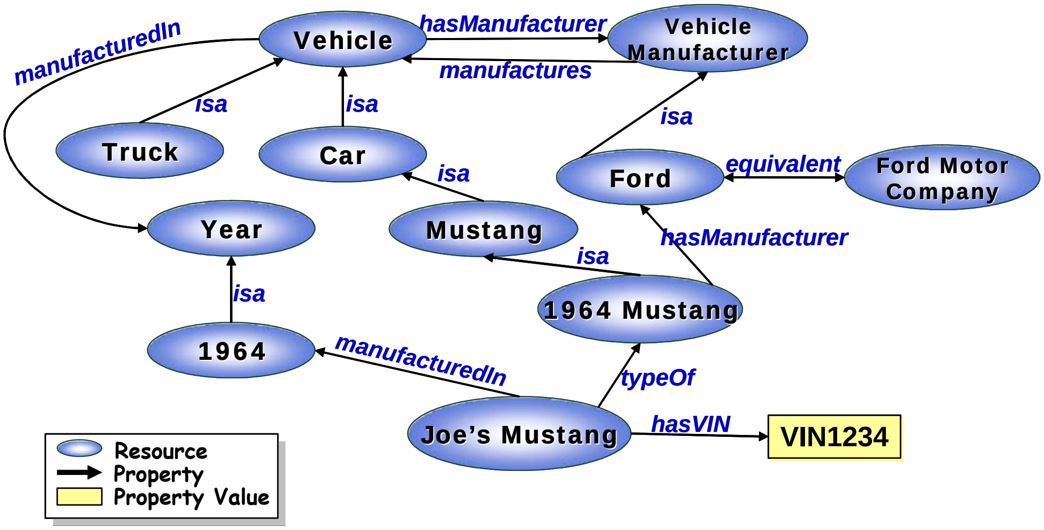


Figure 1. Sample Ontology

OWL has three increasingly-expressive sublanguages: OWL Lite, OWL DL, and OWL Full.[7] OWL Lite supports classification hierarchy and simple constraints. For example, while OWL Lite does support cardinality constraints, their values are restricted to either 0 or 1. OWL DL, named due to its correspondence to description logics, provides for maximum expressiveness while retaining computational completeness (all conclusions are guaranteed to be computed) and decidability (all computations will finish in finite time). OWL Full offers maximum expressiveness and the syntactic freedom of RDF, but no computational guarantees. Note that both RDF and OWL observe the Open World Assumption, that new knowledge can always be added to what already exists.

Ontologies provide a mechanism for machines to perform simple inferencing by combining facts together to form new facts or conclusions. For example, given that *Mary* is the *spouseOf Jim* and *spouseOf* is a symmetric property, an inference engine can conclude that *Jim* is the *spouseOf Mary* without having that fact explicitly asserted. OWL provides the semantic expressiveness that enables machines to inference on the ontology or on facts (i.e., instance data mapped to the ontology) based upon things like relations between concepts (equivalent, disjoint, etc.), property characteristics (inverse, transitive, symmetric, etc.) or cardinality constraints (e.g., birthmother has exactly one value). Constraints and the capability to combine facts to make inferences allow machines to solve tedious problems, combine facts to discover new information, and help prevent certain misunderstandings. They do not solve all problems.

2.2 Why the Semantic Web?

So why should we care about the Semantic Web and its associated technologies? First, even if realization of the Semantic Web is years away, the associated technologies can be applied to a

---

[7] http://www.w3.org/TR/2004/REC-owl-guide-20040210/#OwlVarieties

wide range of today's enterprise challenges and the potential exists for even greater capabilities in the future. In [DOS03] the business case for the semantic web is outlined as follows, "The organization that has the best information, knows where to find it, and can utilize it the quickest wins." This same maxim applies to operational challenges within DoD. Semantic Web technologies contribute to enhanced Decision Support, Information Sharing and Knowledge Discovery by enabling the ability to specify the meaning of concepts in a standard way and then link them across organizations and Communities of Interest (COI).

Specifically, mapping terms to a domain ontology provides a context for that term that can be exploited by software applications. One example application is the ability to perform a concept-based search rather than a text-based search. While search engines today are very powerful, they often result in a glut of information. For example, on July 6, 2004, typing *mustang* into the Google™ search engine resulted in approximately 3,720,000 hits. With the capability to search on the concept mustang, as a type of car, there is the potential to filter many of the undesired "hits."

Ontologies provide the potential to transform search engine technology by using context information to guide the user through query refinement. For example, if one provided a semantic search engine the same text string, the semantic search engine could discover that *mustang* exists in many contexts and could ask the user for clarification on whether they intended this as a kind of horse, kind of car, the name of an entity, or any other discovered association. This power could be applied in search engines that support military intelligence analysts, allowing them to locate and synthesize information more quickly and accurately.

Use of ontologies and inference engines could improve data and application discovery. For example, a software application asked to find all cars manufactured by Ford in 1964, could use the ontology shown in Figure 1 to discover *1964 Mustang* as a valid response to this query. Ontologies could also be used for application discovery as evidenced by the initiatives to use ontologies for Web Service discovery and integration such as OWL-Services[8] and Web Service Modeling Ontology.[9]

Mapping or linking concepts between ontologies can also provide more rapid and agile data integration. For example, the tedious aspects of synthesizing intelligence information could be automated for intelligence analysts, freeing them for more complex integration tasks. Subsequent sections of this paper provide more detail on this potential application of Semantic Web technologies, as testing this hypothesis was the focus of our study.

Ontologies can help with data mediation. Ontologies can be used to mediate between data sets for the purpose of converting attributes (e.g., feet to meters), derivations (e.g., ancestors of a person using transitive *hasParent* property), or mapping between instances (e.g., NoFireOperationsAreas are the union of all NoFireAirspaces and NoStrikeTarget).

There is also great potential value for future applications that exploit the information contained in ontologies. At present, a very small proportion of the data exposed on the Web is marked up

---

[8] http://www.daml.org/services/owl-s/

[9] http://www.wsmo.org/

using Semantic Web vocabularies like RDF and OWL. As more data gets mapped to ontologies, the potential exists to achieve the same exponential growth in value that exists in the WWW. Furthermore, the importance of Semantic Web technologies has been recognized within the military domain for some time. [Bou03] describes several projects that resulted in ontologies relevant to the military domain. Also, the Defense Advanced Research Projects Agency (DARPA) has been a driving force in the advancement of Semantic Web technologies through its DARPA Agent Markup Language (DAML)[10] program and other DARPA initiatives such as DARPA High Performance Knowledge Bases (1996-1999) and Rapid Knowledge Formation[11] (2000-2004).

Finally, we should care about Semantic Web technologies because they are here to stay and evolving rapidly. The Semantic Web is a concept championed by the W3C. The intellectual capital of at least thousands of intelligent individuals is being applied to it. The concepts underlying the Semantic Web are not new. Rather, the Semantic Web is a coalescence of research from many fields including Artificial Intelligence, Knowledge Representation, Database Management Systems, Information Retrieval, Natural Language Processing, Mathematics and Logic. The difference is that the Semantic Web is applying these results on a global scale via the WWW. Further evidence that the Semantic Web is here to stay is that applications of these technologies have moved beyond the research community; they are now being embraced by large commercial companies. Two examples cited in [Bou04] are the use of RDF by Adobe and Sun. Finally, the number of conferences and meetings with Semantic Web themes is exploding, providing further indication of the large and growing interest in these technologies.

---

[10] http://www.daml.org/

[11] http://dtsn.darpa.mil/ixo/programs.asp

3 Netcentric Semantic Linking

Ontologies are viewed by many as a mechanism to facilitate information integration and interoperability between heterogeneous information sources. [Bou03] states that information integration from heterogeneous sources can be addressed at the structural, syntactic, or semantic levels. In this section, we discuss ways to semantically link ontologies as one mechanism to address information integration at the semantic level.

3.1 Linking versus Mapping

We termed this study “Netcentric Semantic Linking” because we wanted to investigate the feasibility of interconnecting domain ontologies as a network centric approach for C2 enterprise integration. Over the course of our research, we found that we should have used the term “mapping” rather than “linking.” To some people, linking implies a one-to-one relationship. This is indeed what is done on the web today with hyperlinks. However, connecting related concepts in ontologies is rarely that simple. The term mapping more accurately reflects the potential for many-to-many relationships. Because the task was already established and underway, we retained the title “Netcentric Semantic Linking.” However, a more accurate title would have been “Netcentric Semantic Mapping.” Therefore, we use the terms linking and mapping somewhat interchangeably in this document.

3.2 Ontology Categories

Ontologies may exist at many levels of abstraction. We group ontologies into three broad categories of upper, mid-level and domain ontologies. Figure 2 is a graphical depiction of these notional levels along with some sample concepts that may be found at each level.

An upper ontology, as defined by [Phy02], is a high-level, domain-independent ontology, providing a framework by which disparate systems may use a common knowledge base and from which more domain-specific ontologies may be derived. The concepts expressed in such an ontology are intended to be basic and universal concepts to ensure generality and expressivity for a wide area of domains. An upper ontology is often characterized as representing common sense concepts, i.e. those that are basic for human understanding of the world [KSD01]. Thus, an upper ontology is limited to concepts that are meta, generic, abstract and philosophical.[12] Standard upper ontologies are also sometimes referred to as foundational ontologies[13] or universal ontologies [Co102].

A mid-level ontology serves as a bridge between abstract concepts defined in the upper ontology and low-level domain specific concepts specified in a domain ontology. While ontologies may be mapped to one another at any level, the mid-level and upper ontologies are intended to provide a mechanism to simplify the mapping of concepts across domains. Mid-level ontologies may

---

[12] http://suo.ieee.org

[13] http://www.opencyc.org/

provide more concrete representations of abstract concepts found in the upper ontology. This ontology category also encompasses the set of ontologies of commonly used concepts, such as Time and Location, which are sometimes referred to as utility ontologies.

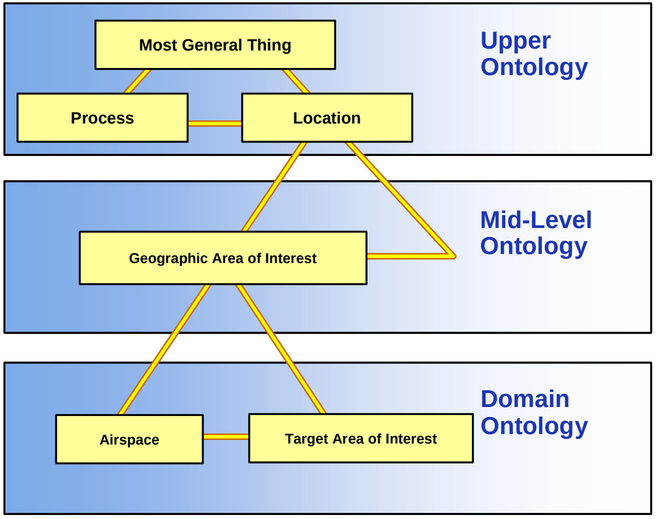


Figure 2. Ontology Categories

A domain ontology specifies concepts particular to a domain of interest and represents those concepts and their relationships from a domain specific perspective. While the same concept may exist in multiple domains, the representations may widely vary due to the differing domain contexts and assumptions. Domain ontologies may be composed by importing mid-level ontologies. They may also extend concepts defined in mid-level or upper ontologies. Reusing well established ontologies in the development of a domain ontology allows one to take advantage of the semantic richness of the relevant concepts and logic already built into the reused ontology. Using common mid-level and upper ontologies is intended to ease the process of integrating or mapping domain ontologies.

3.3 Semantic Linking/Mapping Research

There are many papers published related to semantic mapping using ontologies. [KS03] provides an excellent survey of ontology mapping approaches. Our literature survey, readings, and our own test case convince us that ontology mapping is still an emerging field.

In our research, we've discerned two central themes regarding semantic mapping – it is difficult and it requires significant expertise. Semantic mapping is manual, tedious, and error-prone. There are no fully automated methods for semantic mapping. All approaches we surveyed required user interaction. Even the extensive survey of 35 works by [KS03] failed to discover a fully automated method for ontology mapping. Much of the research on this topic addresses tools to assist users in reducing the effort required in ontology mapping. Also, it is not surprising that semantic mapping, even with the assistance of tools, requires a significant amount of expertise on the part of the person performing the mapping. This person must work with domain experts to understand each domain ontology and its usage in depth, and must have sufficient knowledge to consider the consequences of a mapping when those consequences are often difficult to discern. With an open world assumption new knowledge may always be added. Therefore, if two autonomous ontologies are linked, a change in one of them could have implications on the mapping. It is impossible to anticipate every change and can be difficult to understand the implications of a discovered change. [Kle01] provides a good summary of the problems that can be associated with the use of multiple ontologies (differences in how a domain is conceptualized and differences in how the concepts are specified to include differences in modeling, terminology, and encoding).

A summary of our semantic mapping readings is beyond the scope of this paper. However, we'd like to share a couple findings from our literature survey that we found interesting. One paper on an approach for ontology mapping that uses machine learning techniques to find mappings included the following insightful quote "…on the Semantic Web, the largest benefits of ontology matching come from matching the most heavily used ontologies; and the more heavily an ontology is used for marking up data, the more data it has." [Doa03]

Another interesting discussion is the comparison of ontology mapping to database schema integration in [KS03 pp. 26-27]. The authors note that many practitioners, primarily from database backgrounds, see these approaches as similar. The authors state that while techniques used for database schema matching or integration might be of interest to ontology matching practitioners, there are substantial differences that should be taken into account. They cite a comparative survey by [NK02] and summarize the areas where ontologies and database schemata are different from the perspective of evolution as:

"Database schema evolution aims to preserve the integrity of data itself, whereas ontology evolution is more complex since ontologies can be seen as data themselves, and a typical query on an ontology could result in elements of the ontology itself.

Database schemata do not provide explicit semantics for their data, whereas ontologies are logical systems, and hence the intended semantics is explicitly and formally specified.

Database schemata are not sharable or reusable, usually they are defined over a specific database, whereas ontologies are by nature reusable and typically extend others.

Traditionally, database schema development and update is a centralized process, whereas ontology development is more decentralized and collaborative.

Database schema evolution should take into account the effects of each change operation on the data, like addition of a new class; in ontologies, however, the number of knowledge representation primitives is much higher and more complex: Cardinality constraints, inverse properties, transitive properties, disjoint classes, definition of logical axioms, type-checking constraints.

Databases make a clear distinction between schema and instance data, whereas in rich knowledge representation languages used for ontology modeling it is difficult to distinguish where the ontology ends and the instances begin."

3.4 Semantic Linking/Mapping Approaches

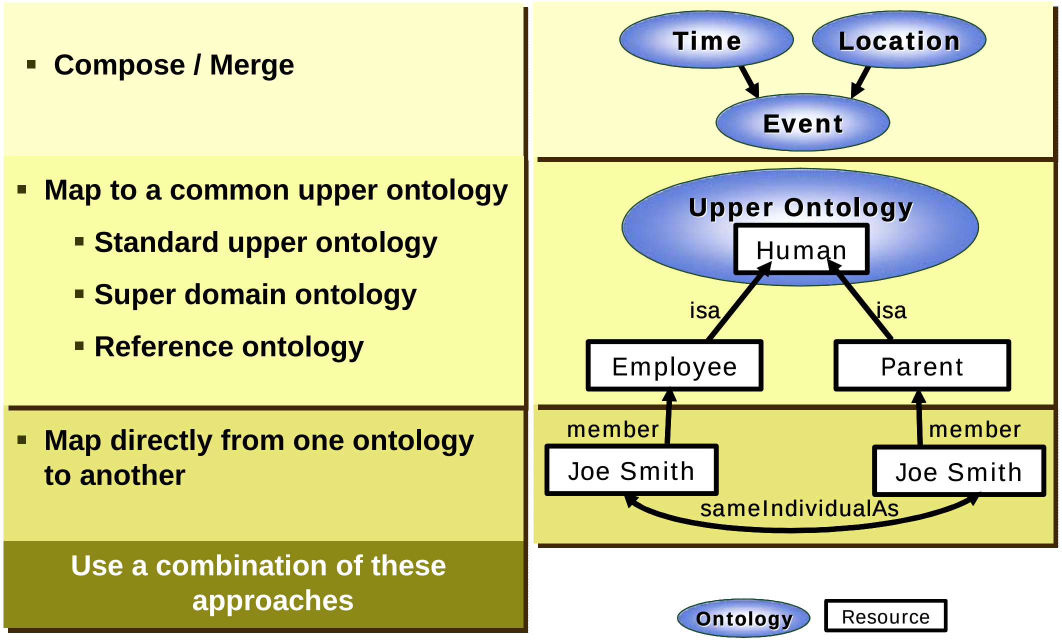
As mentioned above, there has been much research on the problem of mapping multiple ontologies. These approaches are relevant whether one wants to use multiple ontologies together within an application or wishes to attempt to reuse a previously created ontology in the creation of a domain ontology. This section summarizes ontology mapping approaches, focusing on ontology to ontology mapping. Other ontology mapping approaches are mentioned for completeness, but detailed discussions of them are beyond the scope of this paper.

3.4.1 Ontology Meta-Model Mapping

We do not address mapping between different ontology languages or meta-models, as we expect that the emergence of OWL as international standard should simplify ontology mapping approaches. If OWL is adopted as the semantic vocabulary standard, then tools to assist in inter-ontology mapping no longer need to deal with language differences. This of course ignores existing ontologies developed in other vocabularies. [Kle01] describes the four types of language level mismatches that can occur when ontologies written in different languages are combined as mismatches in syntax, logical representation, semantics of primitives, and language expressivity.

3.4.2 Ontology Linking

There are several approaches for ontology linking. These approaches include composition, merging, and several forms of ontology to ontology mapping. These approaches are defined in more detail below. Another option, not explicitly discussed, is a combination of these approaches. The ontology linking options discussed in this section are summarized graphically in Figure 3.



- Compose / Merge
- Map to a common upper ontology
  - Standard upper ontology
  - Super domain ontology
  - Reference ontology
- Map directly from one ontology to another

Use a combination of these approaches

Figure 3. Ontology Linking Options

3.4.2.1 Compose

Composing a new ontology by reusing an existing ontology is one form of ontology linking. This concept is especially relevant when one considers the creation of "utility" ontologies of commonly used concepts. With the existence of such ontologies, ontology designers can compose their domain ontologies using these utility ontologies and inherit the concepts and inferencing capabilities provided by them. Utility ontologies could be analogous to high quality software libraries of commonly used functions. Further, concepts in the utility ontology could be mapped to concepts in an upper ontology without the need for users of the utility ontology to be aware of these mappings. Because it is early in the Semantic Web evolution, few utility ontologies exist. However, they are emerging, as evidenced by the DARPA funded effort to create a standard Time ontology.[14]

3.4.2.2 Merge

Merging can also be loosely considered a form ontology linking. [NM00] defines merging to be the process used to "create a single coherent ontology that includes the information from all the sources." Early Semantic Web research dealt with the development of tools to assist in ontology merging. Clearly, this approach *alone* would not scale to C2 enterprise level where our focus lies. However, merging may provide a partial solution, especially within a domain.

---

[14] http://www.daml.org

3.4.2.3 Map Directly from Ontology to Ontology

Another form of ontology linking is ontology to ontology mapping. Ontology to ontology mapping includes mapping to a standard upper ontology, to a common upper (or super domain) ontology, to a reference ontology, or directly from one domain ontology to another domain ontology.

As previously mentioned, one approach touted for ontology mapping or integration is to use a standard upper ontology. Further information on this approach is discussed in Section 4.

Closely related to standard upper ontologies, is mapping to a common upper ontology. Whereas a standard upper ontology is intended to contain domain-independent, universal concepts, a common upper ontology would contain concepts common across a large domain – in essence, a super domain ontology. A super domain ontology could contain core elements common across the domain that could then be extended within the sub-domains. One could consider this to be a common vocabulary for that domain. Sample super domains could be portions of the U.S. Military (C2, intelligence, logistics, etc.), biology (processes, experiments, functions, etc.) or finance. Ontologies can be layered so what is considered "upper" is relative as shown in Figure 4's notional layering.

Ontologies can also be mapped to a reference ontology that includes key concepts, but no instance data. We decided upon this approach for our mission test case and subsequently found a description of this approach. [KS03] states that a "…Reference ontology is an agreed understanding that favours the sharing of knowledge, and is not supposed to be populated." While reference ontologies are not necessarily instance data free, this was the approach we used in our mission use case implementation.

The final form of ontology to ontology mapping is to map directly from one domain ontology to another domain ontology. Much of the research on tools to assist in ontology mapping focus on this approach. Many methodologies, theories, and approaches have been considered. We found [KS03] to be a good summary of the current state of the art of ontology mapping.

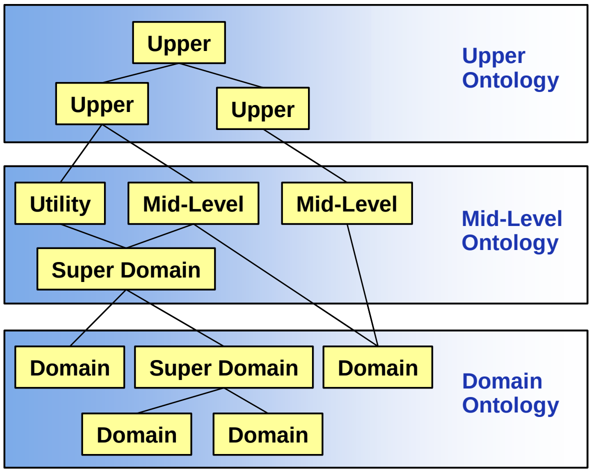


Figure 4. Notional Ontology Layering

3.4.3 Ontology Versioning

Ontology versioning may be considered to be another type of ontology linking. An ontology can be mapped to previous and subsequent versions of itself. In fact, OWL includes built-in constructs of *owl:versionInfo*, *owl:priorVersion*, *owl:backwardCompatibleWith* and *owl:incompatibleWith* to assist in this process. How to manage ontology evolution is a large topic and is beyond the scope of this paper.

3.4.4 Other Types of Ontology Mapping

Other types of ontology mapping include mapping an ontology to instance data, a taxonomic standard, or an application. Commercial tools exist that allow one to map an ontology to one or more relational databases to populate an ontology with instance data. We provide more details on this approach in Section 5 as this is the approach we used in our test case. However, an ontology may also be mapped to any other form of structured or semi-structured data such as the output of a web service or an XML-formatted file. In fact, Network Inference[15] plans to announce soon a capability to use their tools to link ontologies to web services. [OLW03] describes an e-commerce example of mapping a taxonomic standard (specifically the Universal Standard Products and Services Classification (UNSPC)[16]) to an ontology and discusses goals for mapping an ontology to an application's data structures.

---

[15] http://www.networkinference.com/

[16] http://www.eccma.org/

4 Upper Ontology Task

As mentioned in Section 1.2, one goal of this study was to assess the value of using standard upper ontologies in a U.S. Military domain. One approach touted for linking ontologies is to use a standard upper ontology. Although there are several efforts to develop standard upper ontologies to facilitate mutual understanding, there is no consensus on the value of this approach, and in fact some sources doubt the merits of using upper or universal ontologies. For example, Colomb states that it "is extremely doubtful that these universal ontologies can be used as the basis for ontologies supporting interoperating information systems because information systems are largely concerned with institutional facts, which are enormously variable. Institutional facts depend heavily on context and background" [Col02, p.29]. Therefore, we researched standard upper ontology initiatives to assess their applicability to a U.S. Military domain.

As described in Section 3.2, upper ontologies are intended to define foundational concepts used in both mid-level and domain ontologies. In theory, the mapping between domain ontologies becomes easier if the ontologies to be mapped are derived from a standard upper ontology.

Our evaluation of the applicability of a Standard Upper Ontology within U.S. Government and U.S. Military domains was primarily a paper-based study. We conducted a literature survey to better understand what an upper ontology is and how one may be applied. We investigated upper ontology initiatives attempting to define a Standard Upper Ontology, including the IEEE Standard Upper Ontology Working Group (SUO WG)[17] and The WonderWeb Consortium.[18] Under the SUO WG, we evaluated the Suggested Upper Merged Ontology (SUMO)[19] and Upper Cyc[20] Ontology. From the WonderWeb Consortium we considered Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE).[21] We also summarized key ontological choices that must be made by standard upper ontology developers.

Our evaluation criteria were based on our judgment of what is important from a U.S. Government perspective. These evaluation criteria included: licensing, structure, maturity, and ontological distinctions. Our bias was toward an open license and a modular, mature ontology. Figure 5 contains a summary of our evaluation findings.

---

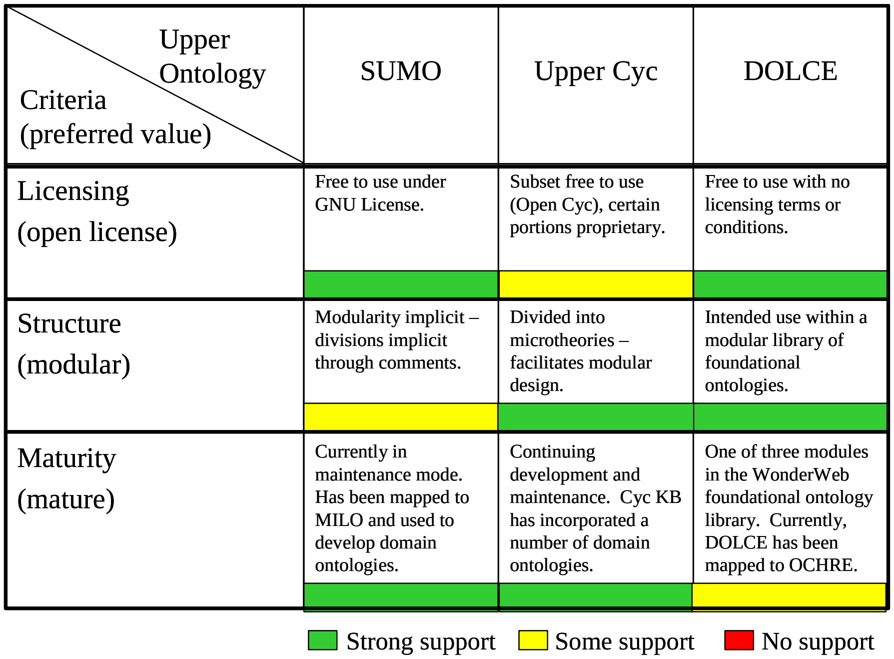
[17] http://suo.ieee.org/

[18] http://wonderweb.semanticweb.org/

[19] http://ontology.teknowledge.com/

[20] http://www.cyc.com/

[21] http://wonderweb.semanticweb.org/deliverables/documents/D18.pdf



Upper Ontology Criteria (preferred value)	SUMO	Upper Cyc	DOLCE
Licensing (open license)	Free to use under GNU License. 🟩	Subset free to use (Open Cyc), certain portions proprietary. 🟨	Free to use with no licensing terms or conditions. 🟩
Structure (modular)	Modularity implicit – divisions implicit through comments. 🟨	Divided into microtheories – facilitates modular design. 🟩	Intended use within a modular library of foundational ontologies. 🟩
Maturity (mature)	Currently in maintenance mode. Has been mapped to MILO and used to develop domain ontologies. 🟩	Continuing development and maintenance. Cyc KB has incorporated a number of domain ontologies. 🟩	One of three modules in the WonderWeb foundational ontology library. Currently, DOLCE has been mapped to OCHRE. 🟨

🟩 Strong support 🟨 Some support 🟥 No support

Figure 5. Upper Ontology Evaluation Summary

In the process of our evaluation, we reached five primary conclusions. First, an open license is essential for the U.S. Government to facilitate information sharing. Second, it is difficult to use an upper ontology as it is intended today, i.e. mapping a domain ontology to an upper ontology to reuse or refine concepts that exist in the upper ontology. This is because there is no agreed upon standard upper ontology, few proven implementations, and little guidance to help discern the impact of using a particular upper ontology concept within a domain. Third, upper ontologies are maturing, and thus there is hope that mapping to them will become easier. Fourth, ontology developers should, at a minimum, consider the contents of upper ontologies as they design their mid-level and domain ontologies. A standard upper ontology is created by experts and can provide a theoretical foundation and conceptual model even if one does not actually map to it. Although there was no single best upper ontology, our current bias is to use DOLCE as a conceptual framework for mid-level and domain ontologies. DOLCE is modular, has an open license, and builds on ontological engineering practices begun in Cyc and continued in SUMO. Also, we see the approach of developing a library of foundational ontologies (rather than a monolithic ontology as is the case with SUMO and Upper Cyc) as promising. Our final conclusion is that utility ontologies that capture commonly used concepts would be valuable within U.S. Government domains and could decrease costs for ontology designers.

5 Semantic Linking Mission Use Case

5.1 Mission Problem

5.1.1 Motivation

One of our goals was to develop a semantic linking test case that spanned military domains to demonstrate the potential value of semantic linking as an approach for cross-domain semantic integration. We decided to use target validation as our mission use case. We chose target validation because it is a process that crosses military domains, a real mission problem exists where semantic linking could provide a solution, and it was suggested by a MITRE domain expert. Also, implementers in both domains are moving toward a service oriented approach, and therefore semantic web technologies could be considered as part of a logical migration within such an approach.

5.1.2 What is Target Validation?

Target development is a core Air Force Intelligence discipline in support of Combat Operations involving the systematic examination of potential targets. This process identifies the critical components of a target and its vulnerabilities to attack and includes the following five major functions: [22]

- Target analysis,
- Target validation
- Documentation
- Nomination, and
- Collection and exploitation requirements.

Target validation is also an important Joint Air Operations mission. According to the *Command and Control for Joint Air Operations* publication, targeting is complicated by the requirement to deconflict duplicative targeting by different forces or echelons within the same force and to integrate the attack of those targets with other components of the joint force.[23] Specifically, the targeting responsibilities of the Joint Force Air Component Commander include the following: [24]

- Direct and ensure deconfliction of joint air operations,
- Synchronize joint air operations,

---

[22] USAF Intelligence Targeting Guide, Air Force Pamphlet 14-210 Intelligence, http://www.fas.org/irp/doddir/usaf/afpam14-210/part05.htm#page41

[23] Command and Control for Joint Air Operations, Joint Publication 3-30, 5 June 2003, http://www.dtic.mil/doctrine/jel/new_pubs/jp3_30.pdf

[24] Ibid

- Coordinate with the appropriate components; agencies/liaison elements for synchronization and deconfliction with land and naval operations,
- Coordinate with the appropriate components' agencies/liaison elements for tasking of the air capabilities/forces made available, and
- Coordinate with the joint force special operations component commander's special operations liaison element for integration, synchronization, and deconfliction with special operations.

Object key objectives of target validation are to avoid friendly fire and to avoid sensitive enemy targets (churches, schools, hospitals, etc.). Therefore, target validation requires knowledge of friendly (blue) and enemy (red) force positions on the battlefield.

5.1.3 Target Validation Today

Target validation is performed today using blue force information in the form of airspace data from the Air Operations Database (AODB) and red force information from the Modernized Integrated Data Base (MIDB). The Air Operations Center extracts airspace data from the AODB to create and distribute the Airspace Control Order (ACO). Intelligence analysts receive the ACO, manually parse the message, extract airspaces of interest, and enter them into their intelligence database as intelligence areas of interest. This allows the analyst to go to one location to retrieve information needed to validate their targets.

The effect is that Airspace information is extracted from the ACO and converted into Intelligence Area of Interest (AOI) information to make it available in a single location for target validation tools. This approach results in loss of data since airspace information is forced from a 3-dimensional or 4-dimensional representation into a 2-dimensional intelligence AOI. Also, this approach necessitates co-mingling of blue and red force data, which is not preferable, and the airspace information being used may become out of date.

Once the ACO data is integrated into the MIDB, the analyst confirms that the candidate targets being considered for attack are on the Joint Target List (JTL). The JTL is a consolidated list of selected targets considered to have military significance in the combatant commander's area of responsibility.[25] An analyst must also confirm that the candidate targers are not No Strike targets (locations that must not be targeted) nor Restricted targets (sensitive targets that should be avoided). Therefore, candidate targets are checked against the No Strike Target List and the Restricted Target List. Additionally, an analyst must confirm that none of the candidate targets will result in friendly fire or fratricide. This is the step in which target locations are checked against planned locations of friendly forces, as specified in the ACOs. Figure 6 illustrates a partial functional flow of the current process.

---

[25] http://www.dtic.mil/doctrine/jel/doddict/data/j/02907.html

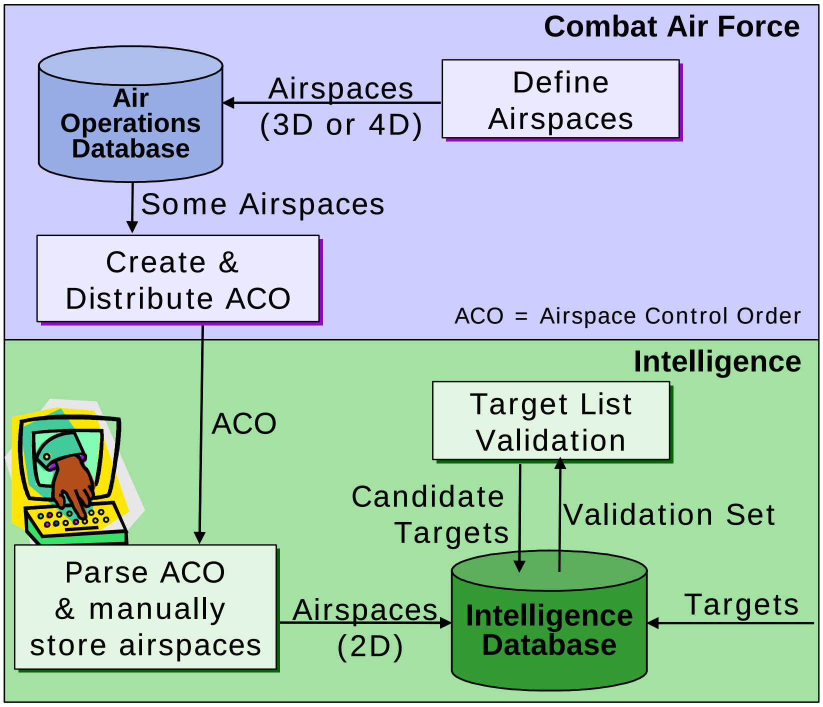


Figure 6. Current Target Validation Process

5.2 Applying Semantic Linking as a Potential Solution

Our hypothesis was that the current target validation process could be enhanced by applying semantic linking technology to bridge the Combat Air Force and Intelligence domains. Our proposed approach was to use ontologies to link relevant AODB and MIDB data to allow target validation to occur using complete and up to date information directly from both authoritative sources. This strategy also avoids the issue of co-mingling red and blue data and, as we discuss later, it offers a much greater opportunity for extensibility.

Fundamentally, we wanted to develop domain ontologies and an approach for linking them that would allow an analyst access to both the target and airspace information they needed to perform target validation. We developed a Target Ontology to conceptually model targets and key target lists and an Airspace Ontology to conceptually model airspaces. We also developed a Reference ontology to link these ontologies together. We examined several options for linking domain data to the Target and Airspace ontologies. One approach for attaching data is to store the instance data internally as part of the ontology. We eliminated this approach early as we wished to use an approach that allowed us to easily integrate with legacy systems and their data. The approach we favored was to store the instance data externally and link it to the ontology. Our intent was to have one domain ontology link directly to a test version of the legacy database of record and have the other domain ontology link to data made available via web services that access the legacy database. Operational and classification restrictions kept us from linking directly to the legacy database and tool limitations (Network Inference plans to develop, but had not yet implemented, the feature to link ontologies to web services within their suite of tools) kept us from attempting the second approach. Therefore, we developed test databases that contained representative operational data. However, as discussed below, we did not replicate the legacy database schema. Our resulting high level design is shown in Figure 7 below. We built a Target Ontology to abstract relevant data from the MIDB. This entailed creating a model of targets and their relevant semantics. Likewise, we built an Airspace Ontology as a conceptual abstraction of airspace data contained in the AODB. We then built a Reference Ontology to semantically map between the concepts contained in the Target and Airspace ontologies, thereby providing a single interface to access data from either legacy database. Details on our implementation follow.

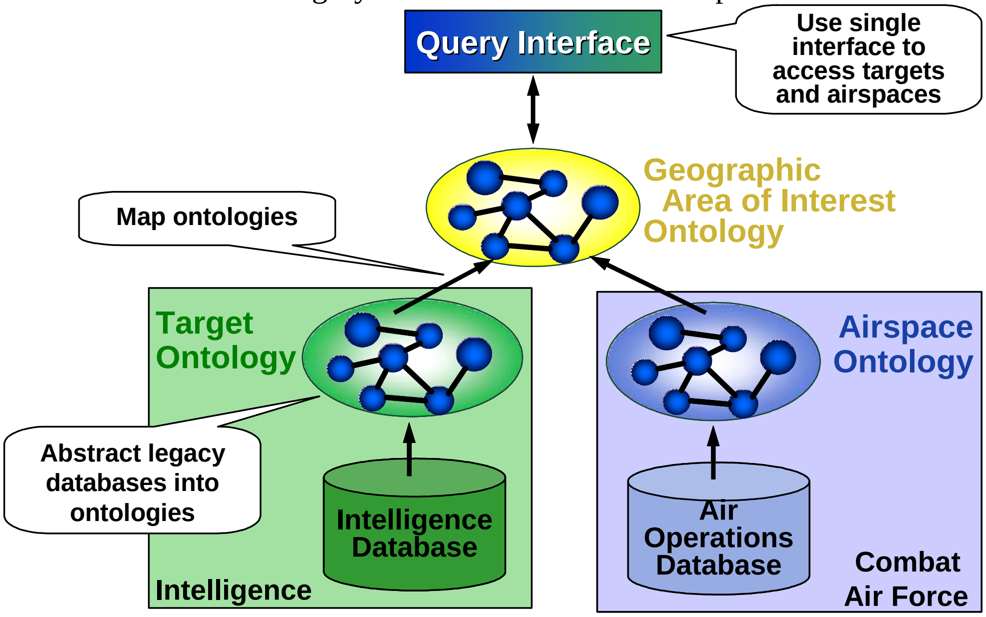


Figure 7. Applying Semantic Linking to Target Validation

5.3 Implementation

There were several steps required to implement our mission use case. A high level overview of our implementation approach is shown in Figure 8 with details to follow. Copies of our ontologies and the client software we created to interface with the inference engine are available to MITRE employees on the MITRE Information Infrastructure (MII) Internal Source Forge site.

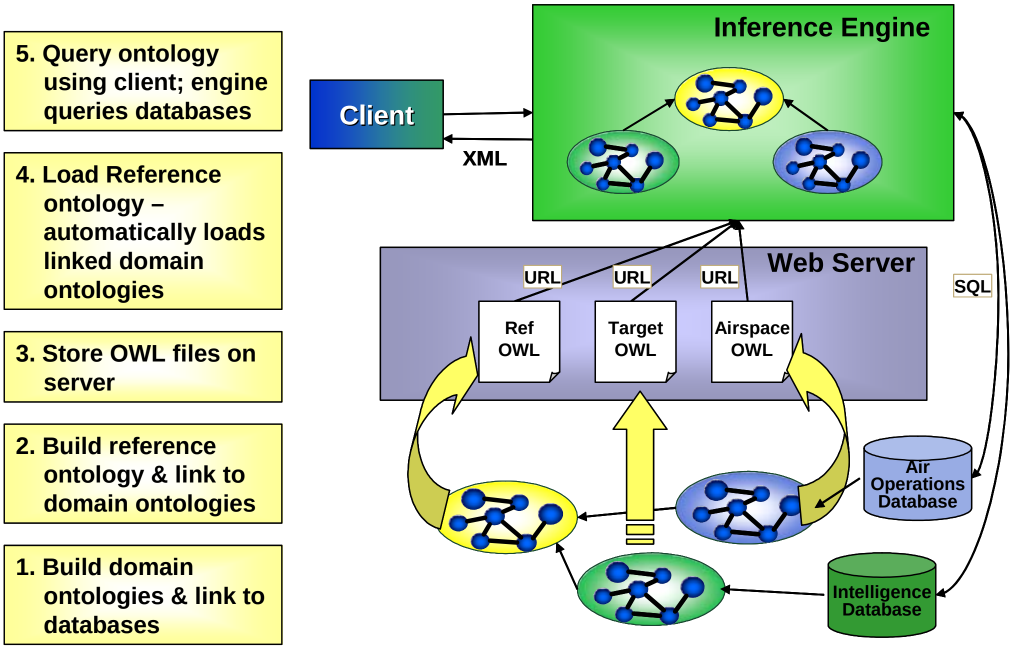


Figure 8. How Our Implementation Works

5.3.1 Select an Ontology Language

As mentioned in Section 1.2, a major goal of this effort was to examine semantic linking using the new international standard Web Ontology Language. Therefore, our goals prescribed the use of OWL in our mission use case. OWL is briefly described in Section 2.

5.3.2 Select Tools

Since OWL was selected as the ontology language, it was necessary to select a tool set that supported the generation of OWL ontologies and an engine that could inference over them. At the start of this project in October 2003, OWL was quite new and not yet a standard. OWL had just become a W3C Candidate Recommendation[26] in August 2003. It quickly progressed to a W3C Proposed Recommendation[27] in December 2003 and became a W3C Recommendation[28] in

---

[26] A Candidate Recommendation is a document that W3C believes has been widely reviewed and satisfies the Working Group's technical requirements. W3C publishes a Candidate Recommendation to gather implementation experience.

[27] A Candidate Recommendation is a document that W3C believes has been widely reviewed and satisfies the Working Group's technical requirements. W3C publishes a Candidate Recommendation to gather implementation experience.

[28] A W3C Recommendation is a specification or set of guidelines that, after extensive consensus-building, has received the endorsement of W3C Members and the Director. W3C recommends the wide deployment of its Recommendations. **Note:** W3C Recommendations are similar to the standards published by other organizations.

February 2004.[29] Considering the status of OWL in October, it was not surprising that there were few OWL-capable tools available to consider. Our experience with FY03 Semantic Web research taught us that use of research level tools (i.e., free and accessible on the internet) meant a sizable time investment in tool integration. We wanted to avoid this since our study was relatively small. We did find one commercial company that offered semantic technologies that used OWL as its native language and provided an integrated suite of capabilities we required. This was Network Inference. Therefore, we purchased a developer's license to their suite of tools.

Network Inference[30] had two native-OWL tools available, Construct™ and Cerebra Server™, that together provide an OWL ontology building tool, OWL inference engine, ability to link ontologies to relational databases to populate them with instance data, and a query interface to the ontology and instance data. Construct™ is a Visio-native graphical modeler that allows knowledge engineers to create and maintain OWL ontologies. Construct™ supports import and export of OWL files and allows mapping of semantic concepts to data in databases and soon to web services. Cerebra Server™ is the inference engine that operates over OWL ontologies. It supports an Application Program Interface (API) and SOAP interfaces, and can store and query ontologies. Instance data can be stored separately in a database and automatically accessed by the inference engine. These Network Inference tools support OWL-DL. Details on how we used these tools in our implementation can be found in Section 5.3.6.

5.3.3 Develop and Populate Domain Ontologies

After selecting and becoming trained on our tool set, our next step was to develop and populate the domain ontologies. In this section we discuss our process and findings. Although we did create the domain ontologies before developing the reference ontology and the queries, the process was not as sequential as implied in this section. Rather, our approach was iterative due to our learning curve and complications encountered with the Network Inference tools.

In general the process was to study the legacy data sources, develop competency questions, then develop the domain ontology in Construct and link the ontology to a relational database of test data. Network Inference stores the database mappings as extensions within the OWL ontology. We then built the queries and tested them using Construct's client interface to the Cerebra Server inference engine. Cerebra Server automatically creates the structured query language (SQL) calls to the database from these mappings. When we had the queries working correctly, we included them in our client interface (discussed in Section 5.3.6).

5.3.3.1 Develop Competency Questions

The first step in developing an ontology is defining its domain and scope. One approach for determining the scope of an ontology suggested by [NM01] is to develop a list of competency questions (i.e., questions that a knowledge base based on the ontology should be able to answer).

---

[29] For more information on the W3C Recommenation Process see http://www.w3.org/2003/06/Process-20030618/tr.html

[30] http://www.networkinference.com

These competency questions are important in that they drive the design of the ontology. We discussed our proposed approach with several domain experts, including an operational targeteer, to confirm that our approach made sense and to get help in delineating a realistic set of competency questions. Our competency questions included the following:

- What are all the active No Fire Areas (or any other airspace type)?
- What are the active Restricted Operations Areas of a given type (i.e., a given subclass of Restricted Operations Areas)?
- Are all the Candidate Targets on the Joint Target List?
- Are any Candidate Targets on the No Strike Target List?
- Are any Candidate Targets on the Restricted Target List?

5.3.3.2 Develop Naming Convention

Our next step was to develop a naming convention for the ontology name, for names of classes and properties within the ontologies, and for the namespaces used. Several projects considered using our Cerebra Server inference engine so we developed a naming scheme to identify the project association for each ontology. Our naming convention for classes and properties within the ontology followed the convention used in both the Protégé[31] open source tool and [NM01]. These conventions were:

Class names start with a capital letter (e.g., *Wine*)

Property names begin with lower case (e.g., *produces*)

For multiple word names, run the words together and capitalize each new word (e.g., *MealCourse or hasDataCode*)

Class names in general will be singular even though a class name represents a collection of objects (e.g., we will use *Wine* rather than *Wines*)

Avoid abbreviations in concept names

Finally, we adopted the namespace naming convention used in the DoD XML Registry for the Aerospace Operations namespace.[32] However, this convention had to be changed due to a bug in the Network Inference tools that required that a namespace name end with a single word (and not use dot separators).

---

[31] http://protege.stanford.edu/index.html

[32] http://diides.ncr.disa.mil/xmlreg/user/prevwhatsnew.cfm

5.3.3.3 Develop Airspace Ontology

The Airspace Ontology models airspace information as defined in the ACO. The ACO defines and establishes special purpose airspace for military operations and notifies all agencies of the effective time of activation and the composite structure of the airspace to be used. We referenced many sources to design the Airspace Ontology including:

- the United States Message Text Format (USMTF) ACO definition, sample ACOs,
- Theater Theater Battle Management Information Core Systems (TBMCS) Information Services which provide an ACO Schema,
- the Cursor on Target XML Schema as a model for point definition,
- the Multilateral Interoperatility Programme (MIP) C2 Information Exchange Data Model (C2EIDM) [33], and
- MITRE domain experts.

In developing the Airspace Ontology we found discussion with domain experts was vital. While official data sources like the USMTF standard provided the needed technical detail, it was discussion with operationally experienced experts that provided us with the understanding needed to develop this semantic model. Discussions on how the ACO was actually used influenced our design. For example, the ACO may define a single physical airspace to have multiple usages. In our design this led to treating each usage as a separate airspace, specifically as a separate instance of the appropriate class of airspace. This means that in our ontology one physical airspace could result in multiple airspace instances, each associated with the same Airspace Control identification number and sharing the same physical characteristics.

Our final Airspace Ontology, shown in Figure 9, includes only a subset of the airspaces defined in the ACO. As we encountered implementation issues we simplified our model to make it easier to identify problems. We modeled three airspace categories, called airspace types in the USMTF. These are the Air Traffic Control airspace, Special Use airspaces, and Restricted Operations Zones. Within each airspace category we modeled a selected set of airspace usages.

As shown in Figure 9, inheritance plays a strong role in this ontology. We defined the Combat Air Force airspace (*CAFAirspace*) to have a set of attributes: name, description, airspace control order identification number (i.e., the specific message), and airspace control means identification number (i.e., the physical airspace). We also added named properties for the airspace shape and the airspace time period. Later, when we linked this ontology to a database we added the attributes of *shapeID* and *timePeriodID* to act as keys to link database tables. The solid lines in the diagram show inheritance (e.g., *SpecialUseAirspace* is a type of *CAFAirspace*).

Because airspaces are commonly referred to by their standard acronym, we created an airspace class named with the standard acronym and made it equivalent to its appropriate class. This

---

[33] http://www.mip-site.org

allows us to reference that airspace class, its properties and its instances using either the full name or the acronym (e.g., either *DropZone* or *DZ*). An equivalent class works well where the exact same concept can be referred to in different ways.

We added color coding to the ontology for clarification. Figure 9 is an export from the Construct tool. However, it is important to note that while the color is stored as part of the graphical representation of the ontology in Construct, it is not part of the OWL ontology itself. We used dark blue color to depict which classes were abstract (i.e., not intended to have instance data) and light blue to depict the classes intended to have instances. The *IntervalTimePeriod* class is shown in white as it was included for context, but not modeled in detail.

We constructed the Airspace ontology, but then made several changes to the design as we learned more about the capabilities of our selected tool, linked the ontology to the database and constructed the queries to extract instance data. Three of these changes are relevant to our overall findings. Our first change was to simplify how we treated instance data constraint checking. OWL ontologies allow the use of XML Schema data types to apply constraints on data values. We wished to use this feature to apply constraints to property values. However, at the time of our implementation Network Inference only supported a small subset of the built-in data types and did not support user-defined data types. Therefore, our property values were defined as either strings or integers, but we could not apply user-defined types to check range of values or apply patterns. One built-in data type we tried to build into our experiment, but had to eliminate was the XML Schema dateTime type. Network Inference tools now recognize the dateTime type, but still are unable to reason using it (i.e., do time calculations).

A second change we made to the ontology was to simplify how we modeled airspace shapes. An airspace shapes may be a line, point, circle, corridor, orbit, polyarc, radarc, or track. Initially, we had planned to model a subset of these shapes in detail. However, we had complications when we attempted to query shape information from our linked airspace database. With the current Network Inference tools there is no way to retrieve data from a linked database using their queries where retrieval of the data requires traversing two links. Our initial ontology modeled a CAFAirspace as having a Shape with that shape being one of a given set of shapes, and that specific shape having a set characteristics. We wanted to query and retrieve the shape characteristics for a given airspace. This required querying across two associations (e.g., airspaceA hasShape shapeB – shapeB is pointC – pointC has latitudeX). This was not possible so we simplified our ontology to list the type of shape (Point, Line, Circle, Orbit, etc) without modeling the shape details.

The final change we made was the most disconcerting, but also the one that has since been addressed in the updated version of the Network Inference tools. After we linked our test airspace database to the Airspace Ontology and began experimenting with queries, we found that our queries retrieved more instance data than we expected. We determined that this was due to the fact that Network Inference tools support only satisfiable queries and not provable queries. A provable query for all instances of a class would return each instance that *definitely is* a member of that class. This was what we expected. A satisfiable query returns all instances that *possibly might* be a member of that class (i.e., It is not impossible that or is not inconsistent that it is a member of the class.). This is what we received. Therefore, to get the instance data we expected, we needed to change the ontology to make it explicitly inconsistent that an airspace of one class could also be an instance of another class. Thus, we defined just about every class as disjoint with every other class. Defining something as disjoint guarantees that an individual that is a member of one class cannot simultaneously be an instance of a specified other class. The disjoint indicators are shown in Figure 9 as red arrows. We found this limitation quite disturbing as it was not operationally true. However, the good news is that the latest version of Network Inference tools supports provable queries.

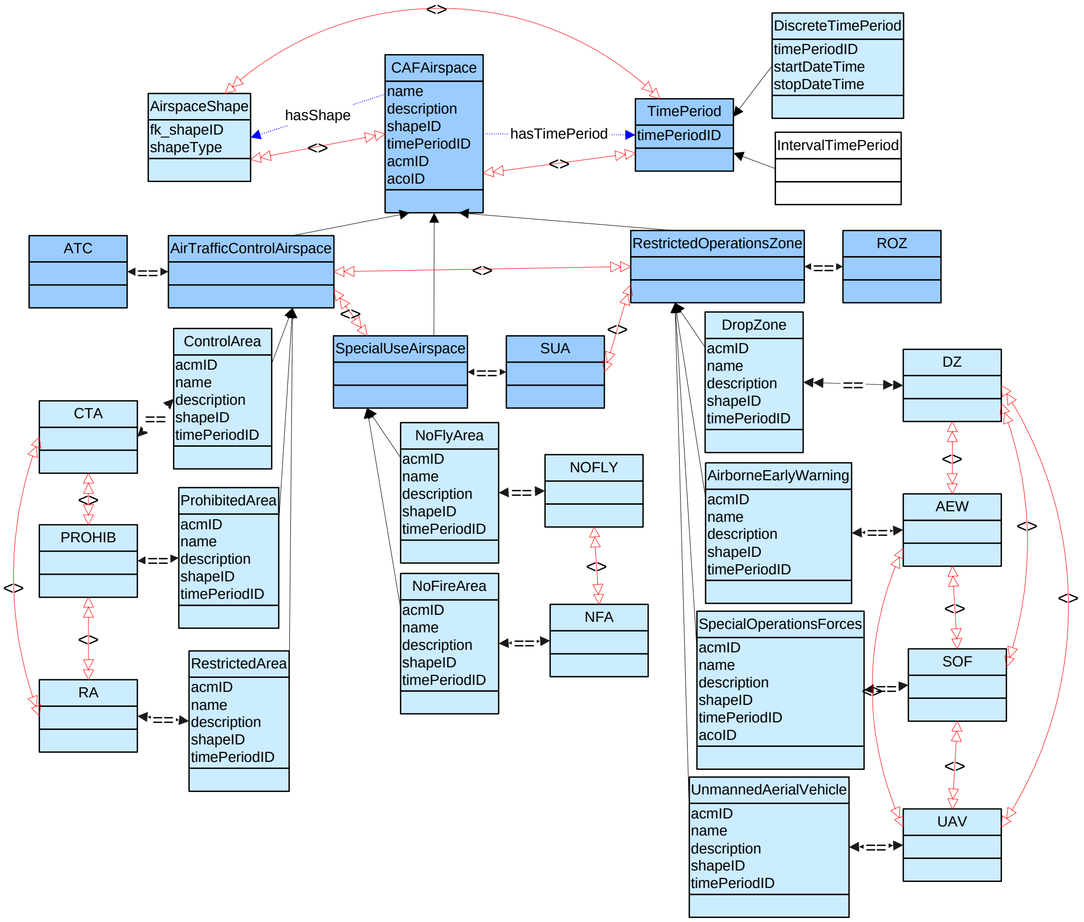


Figure 9. Airspace Ontology

5.3.3.4 Populate Airspace Ontology

Our approach for linking instance data to the Airspace Ontology was to develop a test database that contained representative operational data and link it to the ontology. We used sample ACOs and the results of a TBMCS web service interface to AODB data to obtain operationally realistic data to use as a model for our unclassified test data. We did not try to replicate the operational database schema, but rather developed our test database to align with our ontology. Even then, we found that the design of the database co-evolved with the design of the ontology as discussed in the two examples which follow. This implies that linking legacy databases to ontologies could be quite challenging and more work is needed to more loosely couple ontologies with linked sources of instance data such as databases.

So, our database design was heavily influenced by the tool capabilities. Our initial approach was to model the database to reflect the contents of the ACO and have a table that contained a list of physical airspaces each of which could have multiple usages. However, there was no way to extract a single airspace and map it as a member of multiple airspaces classes based on attribute values that reflect the multiple usages. Therefore, we took the easy approach and created a database table for each class that was to contain instance data (i.e., the light blue boxes in Figure 9). This allowed us to easily populate the classes with instance data and answer our competency question of what are all the airspaces of a particular type. But, this is a "show stopper" for use in an operational environment. It will not be feasible to redesign legacy databases to accommodate limitations in ontology tools.

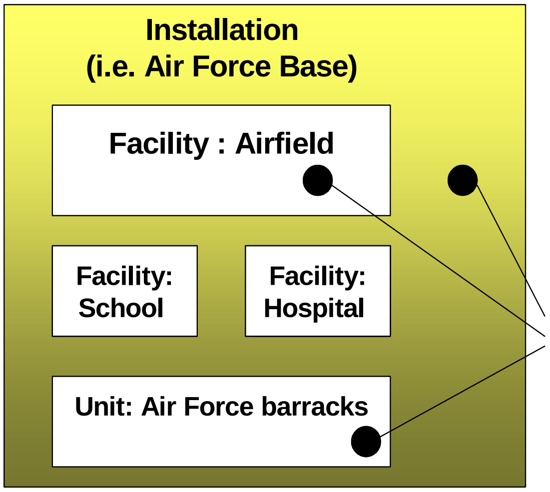
To accommodate linking a specific instance of a named property to a particular airspace, we added unique identifies (i.e., database keys) to both the ontology and the database. With this approach we could have a specific instance of a named property map to multiple airspace instances. While this is not a large change to the ontology, it does demonstrate that database linkages can be tightly coupled to the ontology.

5.3.3.5 Develop Target Ontology

Several sources were referenced to develop the target ontology including:

- MIDB schema,
- TBMCS ODB Target web Services on the Developer's Network (DevNet),
- USMTF target information, and
- MITRE Domain Experts.

With information collected from these sources, we designed the Target ontology to capture the semantics of the relevant components of the concept of Target as represented in the MIDB database. Target is really a desired point of impact, referred to in the MIDB as a Designated Mean Point of Impact (DMPI), essentially specified by coordinates. However, a target can also refer to a Targeted Facility, such as an Air Base or Factory, or a Targeted Unit, which is an organization, such as a group of soldiers. If the target is a facility or unit, there is typically a DMPI specifying the precise point of the target. See Figure 10 below for an example of how a Facility and a Unit can be targeted using DMPI.



DMPI is the desired point of impact.

DMPI may specify a targeted Facility, Unit or any point.

Figure 10. Facility and Unit as Targets

This operational usage had an impact on the design of the ontology. The primary classes of data were *Target*, *DMPI*, *TargetedFacility and TargetedUnit.* We derived the Target List classes (Candidate, Restricted, NoStrike and Joint) by talking with domain experts at MITRE and consulting the Joint Tactics, Techniques, and Procedures for Intelligence Support to Targeting[34] and the USAF Intelligence Targeting Guide.[35] We modeled a target list as all members of the respective class of targets.

Since the intended usage of the ontology was to support the validation of targets, we also had to consider how targets are classified by the targeteer. As described in section 5.1.3, part of the target validation process is to confirm that candidate targets are on the Joint Target List, but are not on either the No Strike Target List or the Restricted Target List.

So, the ontology design had to model that a target can be a facility, unit or point, as well as the fact that a target could be a member of a number of target lists. The ontology was originally designed such that an instance of a valid target could be a member of *TargetedFacility*, *TargetedUnit*, or *DMPI* and also a member of the *Candidate* and *Joint* target lists, but not a member of the *NoStrike* or *Restricted* target lists. However, the limitation of the inference engine only supporting satisfiable queries required that we define the target lists as disjoint from each other as well as disjoint from the types of targets. Therefore, we added the concept of *TargetedEntity* as the as union of *TargetedFacility*, *TargetedUnit*, and *DMPI* and specified *TargetedEntity* as disjoint from *Target*. This was not logical in terms of operational use, but was a necessary workaround until the tools evolved to support provable queries. As mentioned in Section 5.3.3.3, the Network Inference tools now support provable queries. The resultant ontology is shown in Figure 11.

---

[34] Joint Pub 2-01.1, Joint Tactics, Techniques, and Procedures for Intelligence Support to Targeting, 09 January 2003. http://www.dtic.mil/doctrine/jpintelligenceseriespubs.htm

[35] USAF Intelligence Targeting Guide, AF Pamphlet 14-210 Intelligence, 1 February 1998. http://www.fas.org/irp/doddir/usaf/afpam14-210/

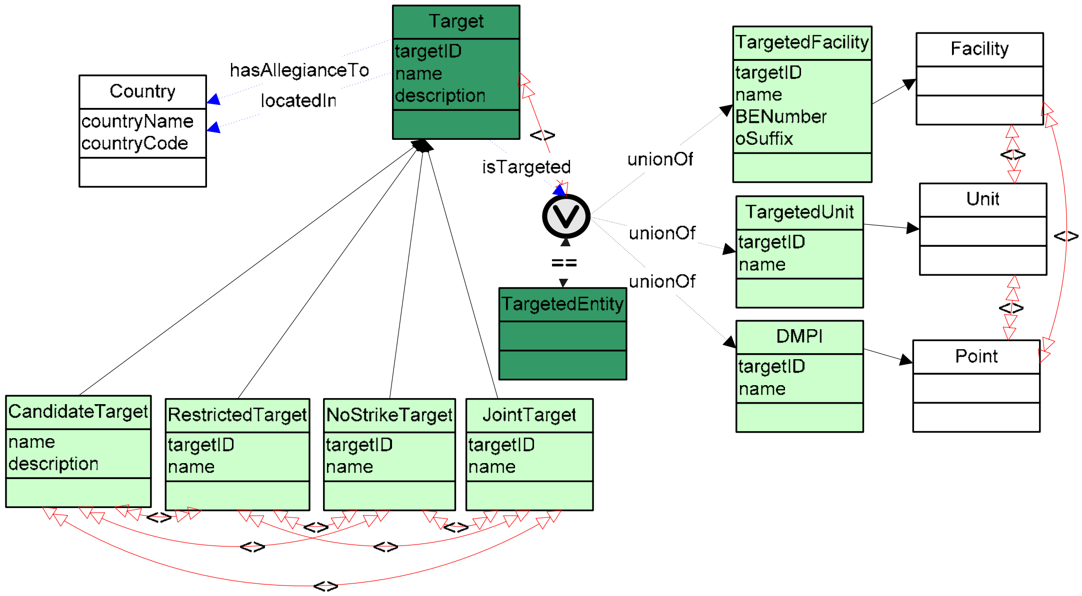


Figure 11. Target Ontology

5.3.3.6 Populate Target Ontology

Since the Network Inference Cerebra Engine did not yet support using web services as a data source for instances, we decided to link the Target Ontology to a simulated operational database to populate it with instance data. This allowed us to explore the difficulty and feasibility of using this approach.

Our intent was to simulate a real target database as closely as possible. To do so, we investigated several approaches for gaining access to MIDB data. Our selected approach was to design the ontology and database using the MIDB database schema and the XML output from the Target Management web services offered by the TBMCS on their Developer's Network (DevNet). Instances were mapped to the concepts shown in light green in the Figure 11. *Target* and *TargetedFacility* (in dark green) are abstract classes (i.e., represent higher level concepts and are not to be instantiated). The concepts shown in white are included for context, but are not modeled in detail.

We created separate database tables for the classes *TargetedFacility*, *TargetedUnit*, and *DMPI*. This design was intuitive and mimics the operational representation in TBMCS and MIDB.

Our major finding was the same as that discussed in Section 5.3.3.4 above. That is, we were forced to build separate tables for each subclass of *Target* (i.e., Target list). This is an operational "show stopper" as it is not feasible to always build separate tables for each type within a class of data. We wished to be able to assign any *TargetedEntity* (be it a *TargetedFacility*, *TargetedUnit*, or *DMPI*) to be a member of one or more target lists (i.e., subclasses of *Target*). However, the limitation of only satisfiable queries forced us to create a separate database table for each subclass of *Target* and manually replicate the data.

We were able to get some unclassified sample data from the Gemini Infrastructure Intelligence Portal, a source of intelligence information available on SIPRNET. We used this data to confirm types and sample values for Facility and Unit data. We created the actual test data ourselves based on this information and validated it with our MITRE domain experts.

5.3.4 Develop and Map Reference Ontology

We designed an ontology to semantically link related concepts across the Airspace and Target ontologies. This Geographic Area of Interest Ontology is a Reference ontology and as such is not intended to contain instance data. After we developed this approach and the possible classes to include, we confirmed the approach and the suggested classes with both domain experts at MITRE and one of the development contractors.

The Reference ontology, shown in Figure 12, contains geographic areas of interest concepts from a military perspective. The root class, *GeographicAreaOfInterest*, has several subclasses. The color coding indicates which classes were modeled in the Reference ontology (yellow), which classes were imported from the Target ontology (green), which classes were imported from the Airspace ontology (blue), and which classes were included only to add further context (white). Intending to experiment with different forms of mapping, we mapped the Airspace ontology using an equivalence relation and the Target ontology using a subclass relationship. Once these ontologies were mapped together, we could access information from any of them through the vendor supplied query interface. This is a powerful capability with significant potential. Further, we demonstrated the ability to introduce new concepts in the Reference ontology (i.e., *RestrictedOperationsArea* and *NoFireOperationsArea*) that joined together domain ontology concepts that were related from a mission perspective. This allowed us to treat the members of the joined classes collectively. See Sections 5.3.5.2 and 5.3.5.3 for more information on our examples of using the complex class constructor *unionOf* to define these new concepts.

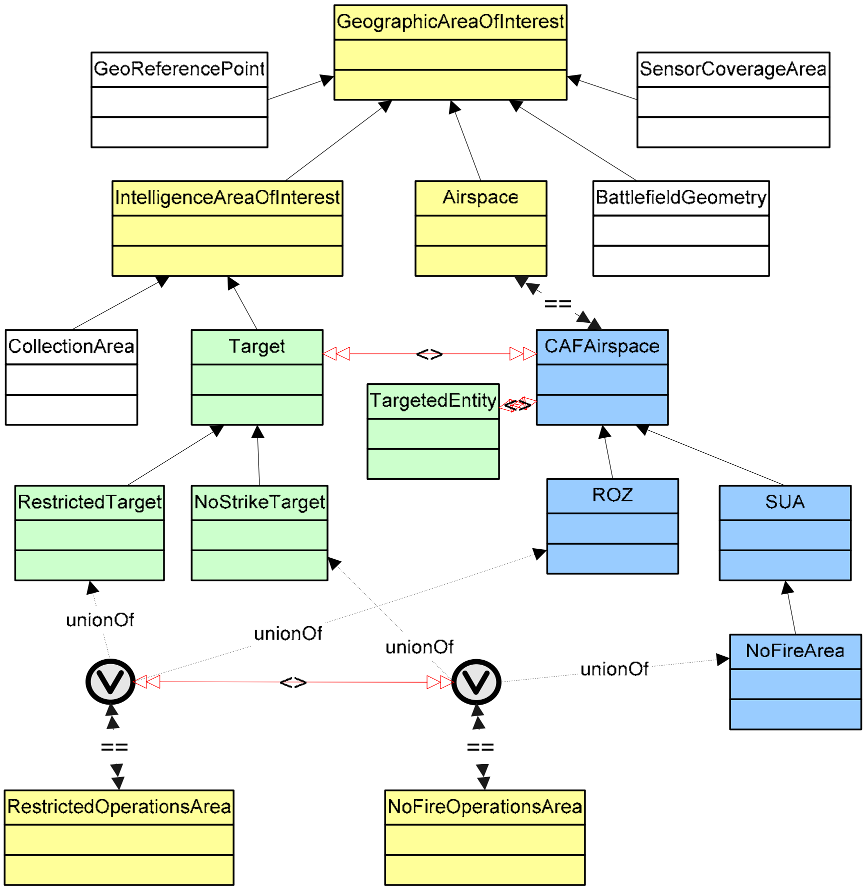


Figure 12. Geographic Area of Interest Reference Ontology

5.3.5 Develop and Test Queries

We developed a set of queries both to answer our competency questions and also to incrementally demonstrate the capabilities afforded by Netcentric Semantic Linking. Ontologies may be queried for information about the ontology itself (e.g., What are all the classes of this ontology? What are all the properties associated with a specific class? What are all the subclasses of a given class?) or for instance data mapped to the ontology (e.g., What are all the members of class *X*?). To query our ontologies, we used the Network Inference query capabilities which are based on the W3C XQuery[36] standard. XQuery is designed to work with the XML data model, but extensions are being proposed to define how OWL should work with XQuery. The Network Inference tools only supported a portion of XQuery.

By necessity, the queries we developed were closely tied to the design of the ontologies and the way the databases were linked to the ontologies. Getting the queries to return the data we

---

[36] http://www.w3.org/XML/Query

expected was an iterative and time-consuming process. This was due to many factors including our learning curve, software bugs in the tools, and the lack of features we desired (and sometimes required) in the tools. The vendor was quite helpful throughout this process and they have made many improvements in their tool suite over the course of this one year project. A robust query approach for OWL ontologies is an area where we see a strong need and expect to see significant improvements in the near term.

In this section we step through a series of queries to demonstrate the potential of semantic linking. For each query we provide a graphical simplification of our linked ontologies to allow the reader to visualize the query result. In these diagrams, the yellow ovals represent selected classes from the Reference ontology while the green and blue ovals represent classes from the Target and Airspace ontologies, respectively. In the Target and Airspace ontologies, the darker colored ovals represent abstract classes (i.e., classes that are not intended to have instances). We also added gray rectangles that show the values that are expected from our test databases. For convenience, we show only the value of the property *name* (i.e., Target name and Airspace name). This provided a "cheat sheet" of expected results during demonstrations of our prototype software. The actual queries we used may be found in Appendix A.

5.3.5.1 Query Each Domain Ontology Individually

The first part of our demonstration included two separate queries to show that once ontologies are mapped together a user or application has access to all the data linked to the mapped ontologies. Figure 13 shows the results of two separate queries. This includes a query for all instances of the class *NoStrikeTarget* and a query for all instances of the class *NoFireArea*. As shown in the figure, the results were three instances of *NoStrikeTarget* and one instance of *NoFireArea*. These two queries successfully demonstrated the ability to access domain data stored in databases by using ontologies as an abstraction layer. Further, these queries show the ability to use a single interface to access data from multiple repositories. This is very powerful. One could easily conceive of a portal that allows a user or application to query on a given concept and gather data from many distributed data sources without the need to know where that data is located or how the data is stored in its native repository.

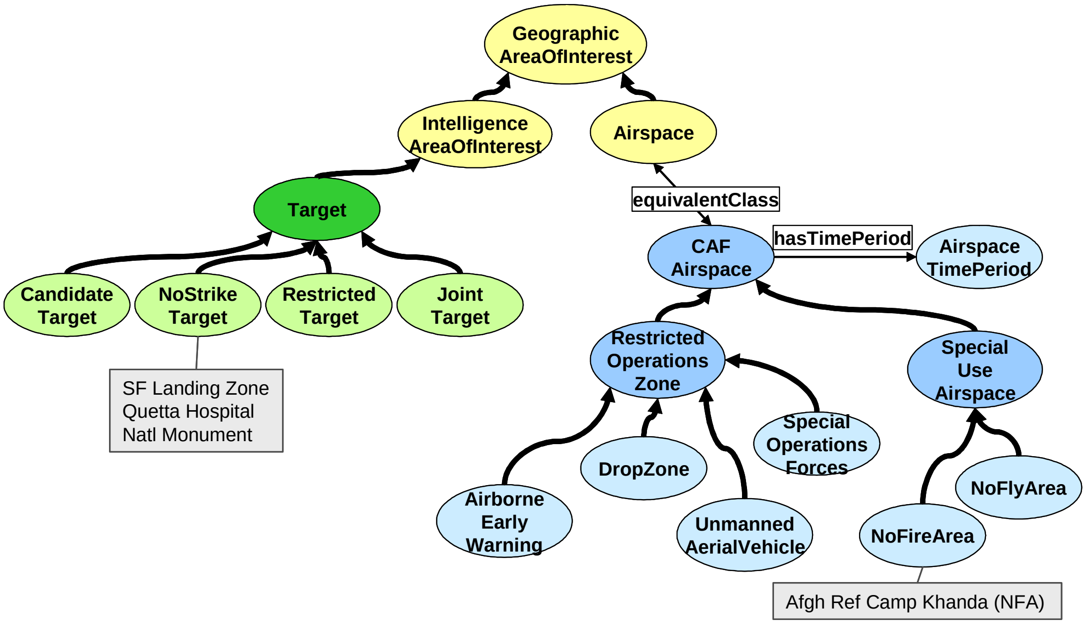


Figure 13. Ontology Queries: All NoStrikeTargets and NoFireAreas

5.3.5.2 Query NoFireOperationsArea

This query shows the ability to add ontological concepts as a bridge between associated concepts in different operational domains. This can be done as a convenience or to meet evolving operational needs. We added the concept *NoFireOperationsArea* to the Reference ontology and defined it as the union of *NoStrikeTarget* from the Target ontology and *NoFireArea* from the Airspace ontology. By doing this we could now consider the members of those two classes collectively. Therefore, our query for all instances of the class *NoFireOperationsArea*, returned all the members of *NoStrikeTarget* as well as all members of *NoFireArea* as shown in Figure 14. Figure 15 is a screen shot of the client interface we developed (described in section 5.3.6) that captures the results of this query.

This query shows another powerful capability that can be exploited to operational advantage. As new concepts are conceived, they can be added to an OWL ontology due to OWL's open world assumption – that not all things are known and new information can always be added to what already exists. Add to that OWL's capability to use class expressions such as *unionOf* and you have the tools to create new concepts that can be exploited by applications. Our query result demonstrates that an application can access instance data resident in both test databases by querying on a single concept. This is a conceptual equivalent to a database join across two different databases that could use different database tools, have different database structures, and reside in different locations. You could also create concepts using other class operators. For example, one might define the class *NonUnitedStatesSensor* as an intersection of the class Sensor with the class of all things not located in the United States.

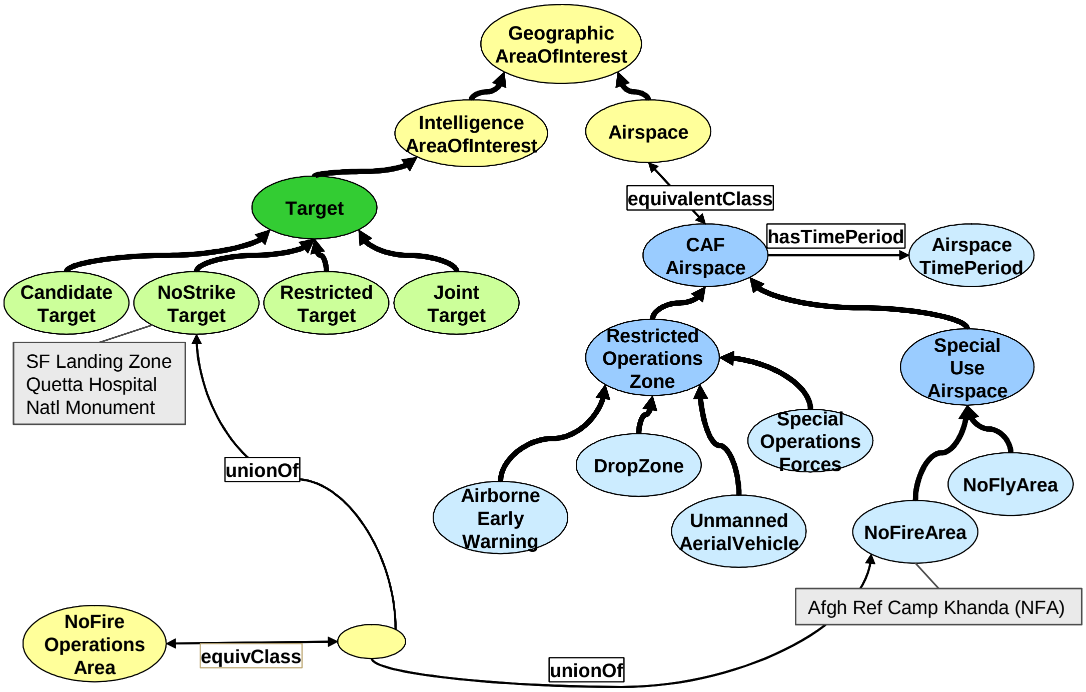
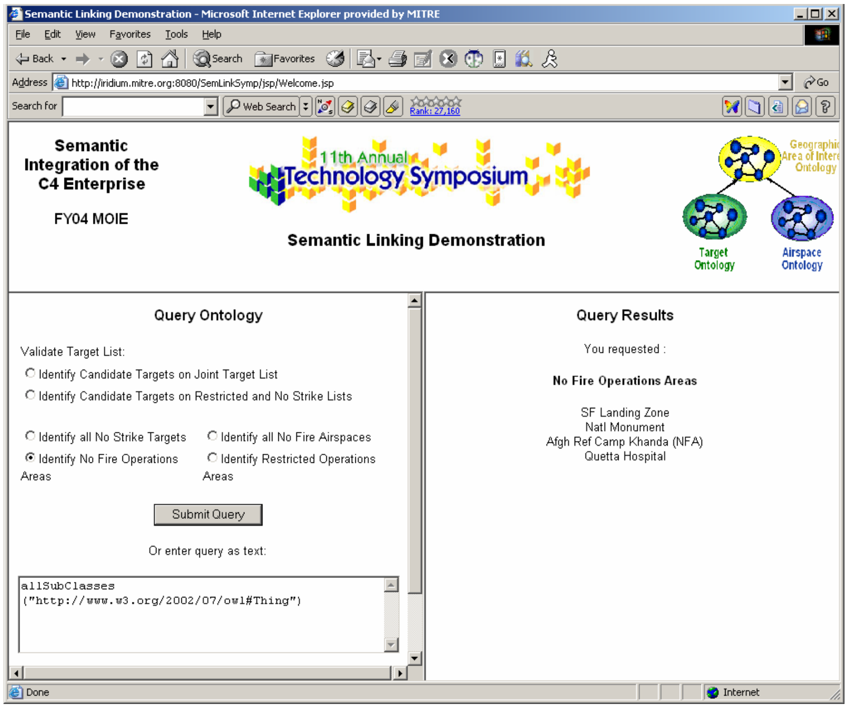


Figure 14. Ontology Query: Identify NoFireOperationsAreas



Query Ontology

Validate Target List:

○ Identify Candidate Targets on Joint Target List
○ Identify Candidate Targets on Restricted and No Strike Lists

○ Identify all No Strike Targets ○ Identify all No Fire Airspaces
● Identify No Fire Operations Areas ○ Identify Restricted Operations Areas

Submit Query

Or enter query as text:

```
allSubClasses
("http://www.w3.org/2002/07/owl#Thing")
```

Query Results

You requested :

No Fire Operations Areas

SF Landing Zone
Natl Monument
Afgh Ref Camp Khanda (NFA)
Quetta Hospital

Figure 15. Query Result: Identify NoFireOperationsAreas

5.3.5.3 Query RestrictedOperationsArea

This query is very similar to the last one, but it shows that one can join concepts at different levels of aggregation. In this case, we defined a new concept called *RestrictedOperationsArea* as the union of all members of the class *RestrictedTarget* from the Target ontology and all members of the class *RestrictedOperationsZone* from the Airspace ontology. Recall that *RestrictedOperationsZone* is an abstract class and has no instances. However, it does have subclasses that have instance data mapped to them. This means that a query for all members of the class *RestrictedOperationsArea* will return all members of *RestrictedTarget* as well as all members of the classes *AirborneEarlyWarning*, *DropZone*, *UnmannedAerialVehicle*, and *SpecialOperationsForces*. These results are shown in Figures 16 and 17.

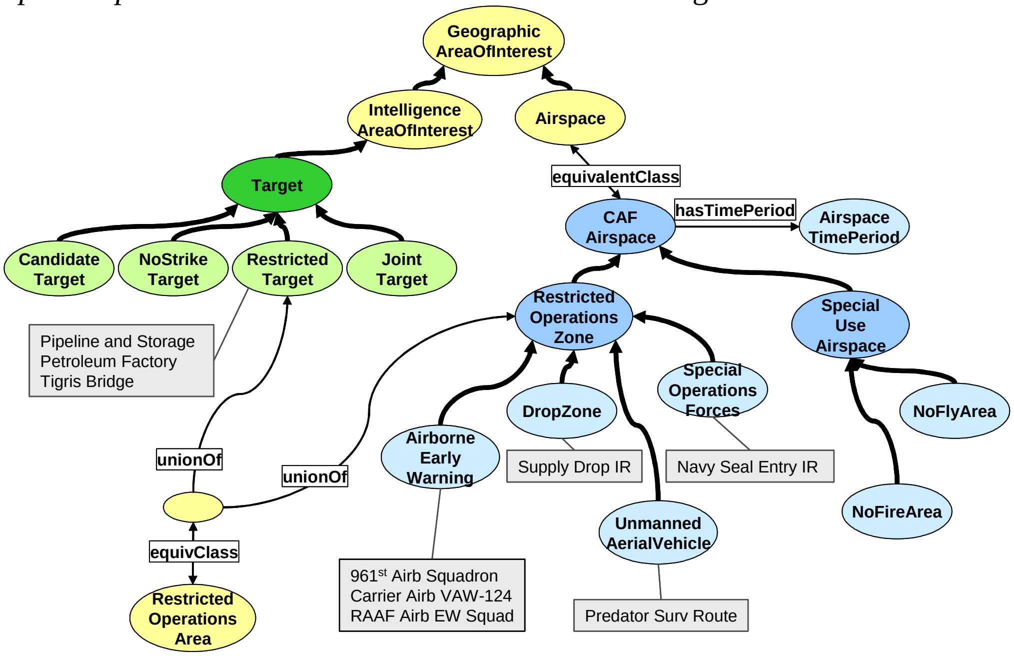
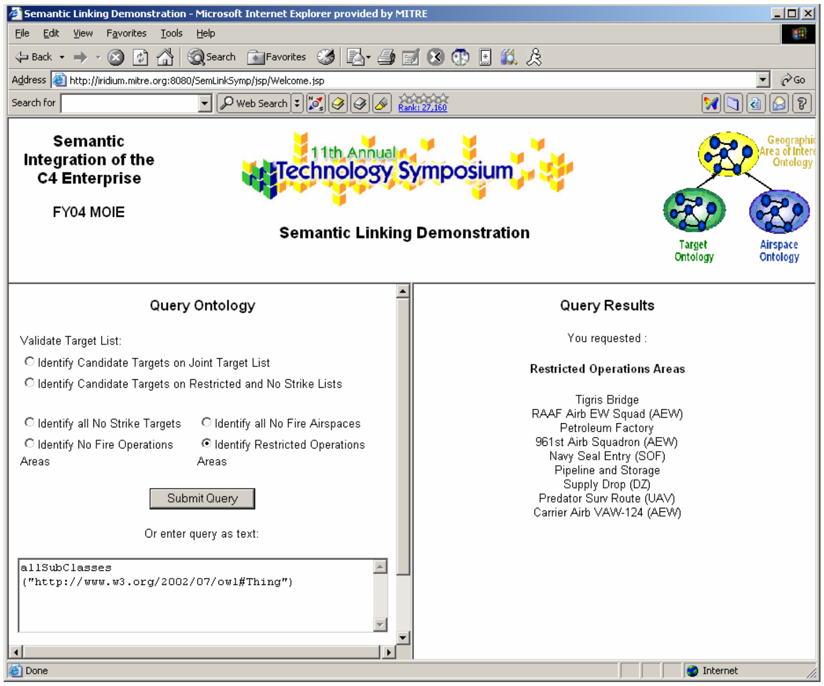


Figure 16. Ontology Query: Identify RestrictedOperationsAreas



Query Ontology

Validate Target List:

○ Identify Candidate Targets on Joint Target List
○ Identify Candidate Targets on Restricted and No Strike Lists

○ Identify all No Strike Targets
○ Identify all No Fire Airspaces
○ Identify No Fire Operations Areas
● Identify Restricted Operations Areas

Submit Query

Or enter query as text:

```
allSubClasses
("http://www.w3.org/2002/07/owl#Thing")
```

Query Results

You requested :

Restricted Operations Areas

Tigris Bridge
RAAF Airb EW Squad (AEW)
Petroleum Factory
961st Airb Squadron (AEW)
Navy Seal Entry (SOF)
Pipeline and Storage
Supply Drop (DZ)
Predator Surv Route (UAV)
Carrier Airb VAW-124 (AEW)

Figure 17. Query Result: Identify RestrictedOperationsAreas

5.3.5.4 Query to Support Target Validation

The two queries discussed in this section support target validation. These queries demonstrate that Netcentric Semantic Mapping can be used to support military operations. As discussed in Section 5.1.3, intelligence analysts must confirm that any candidate targets are on the Joint Target List, but are not on either the No Strike Target List or the Restricted Target List. We tried to formulate a single query but discovered from the vendor that the Network Inference queries did not yet support the *not* operation. Therefore, we divided the query into two separate queries. The first queries whether the targets that are members of the *CandidateTarget* class ARE also members of the *JointTarget* class (the answer is shown in green in Figure 18). The second queries whether any candidate targets ARE either members of either the *NoStrikeTarget* or *RestrictedTarget* class (i.e., are not valid). The invalid target in this case is shown in red in Figure 18.

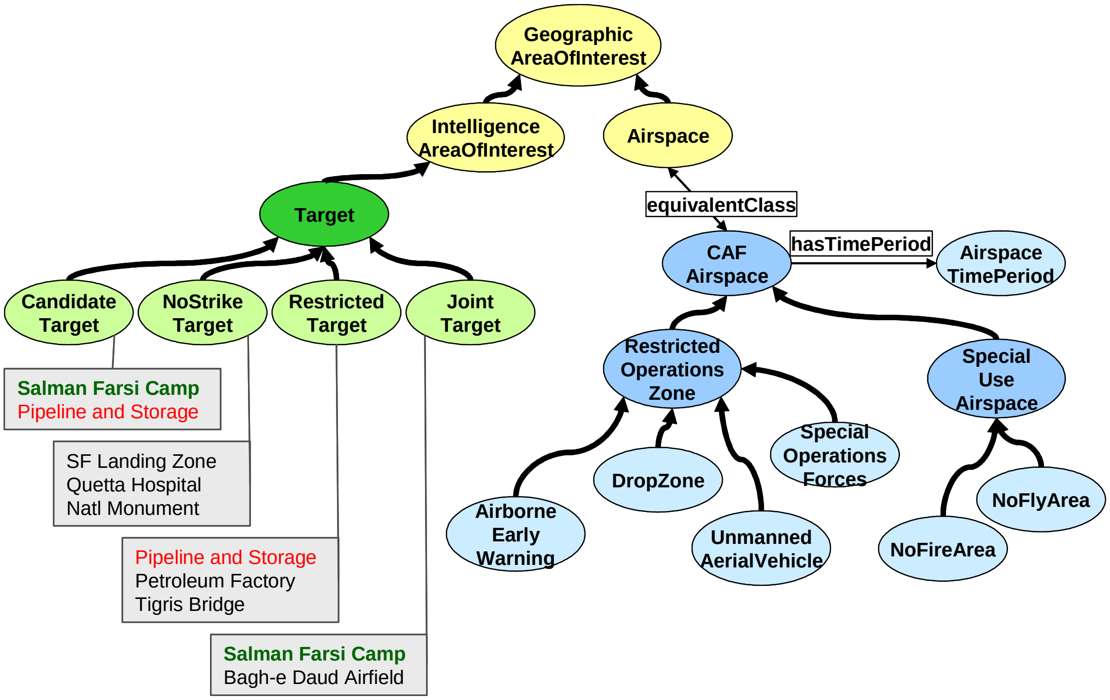


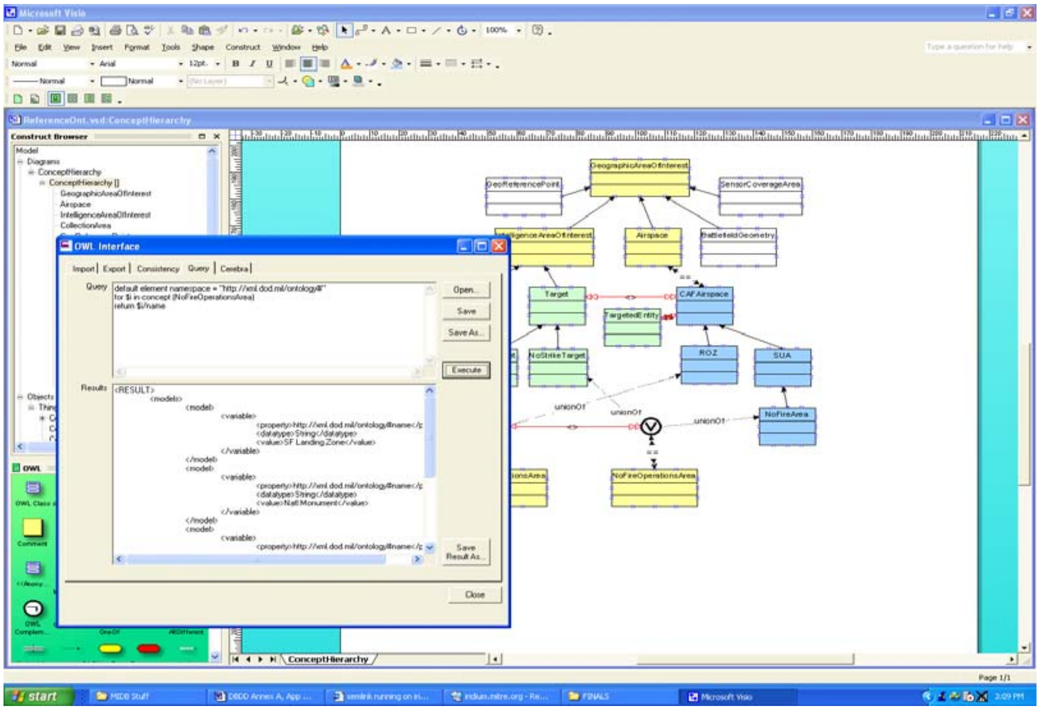
Figure 18. Ontology Queries: Identify CandidateTargets on JointTarget List and on RestrictedTarget and NoStrikeTarget Lists

5.3.6 Develop Client Application

To demonstrate the ability to exploit semantic linking, we built a custom client to interface to the OWL reasoning engine, Network Inference’s Cerebra Server. In a real mission application, the Cerebra Engine would be accessed by another application such as a target validation tool. However, we needed a simple way to demonstrate our mission use case so that the potential power of semantic linking could be observed.

Cerebra Server uses a modified XQuery syntax to query ontologies and their associated instance data and returns results in XML. One can query an ontology that has been loaded into the Cerebra inference engine either using the Construct tool as a client or via a client that uses the Cerebra Server application programming interface (API) or web service interface. Figure 19 is a screen shot of a query and the query result using the Construct client interface.

The intent of our client application was to hide the details of the XQuery-like language used to query ontologies loaded into Cerebra and to translate from the returned XML to something more easily interpreted by users. Figure 20 shows the high level architecture of the Cerebra client we built.



```
default element namespace = "http://xml.dod.mil/ontology#"
for $i in concept (NoFireOperationsArea)
return $i/name
```

```
<RESULT>
        <models>
                <model>
                        <variable>
                                <property>http://xml.dod.mil/ontology#name</p
                                <datatype>String</datatype>
                                <value>SF Landing Zone</value>
                        </variable>
                </model>
                <model>
                        <variable>
                                <property>http://xml.dod.mil/ontology#name</p
                                <datatype>String</datatype>
                                <value>Nat'l Monument</value>
                        </variable>
                </model>
                <model>
                        <variable>
                                <property>http://xml.dod.mil/ontology#name</p
```

Figure 19. Construct Query Interface

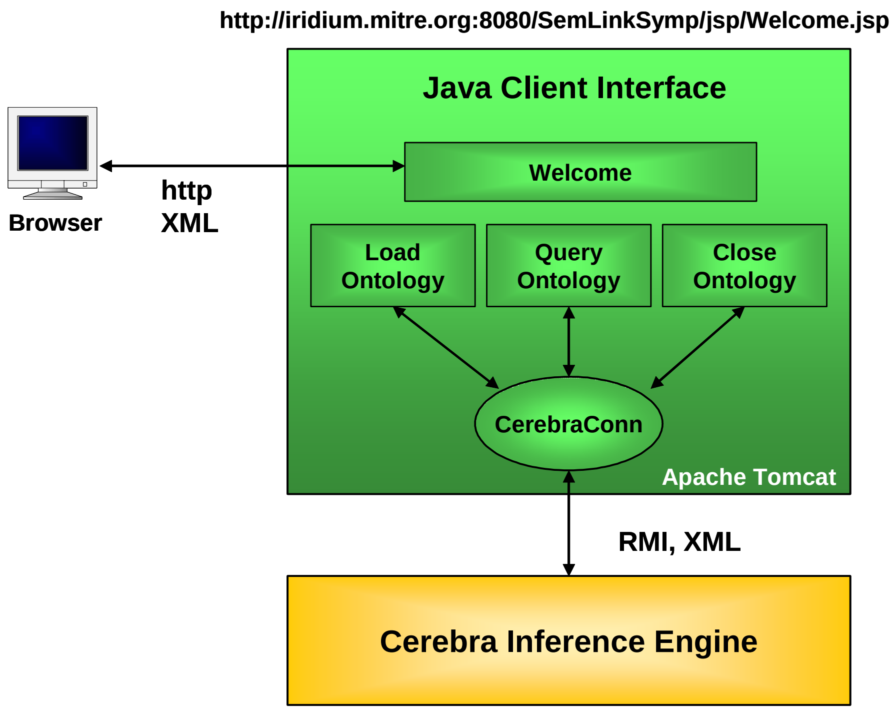
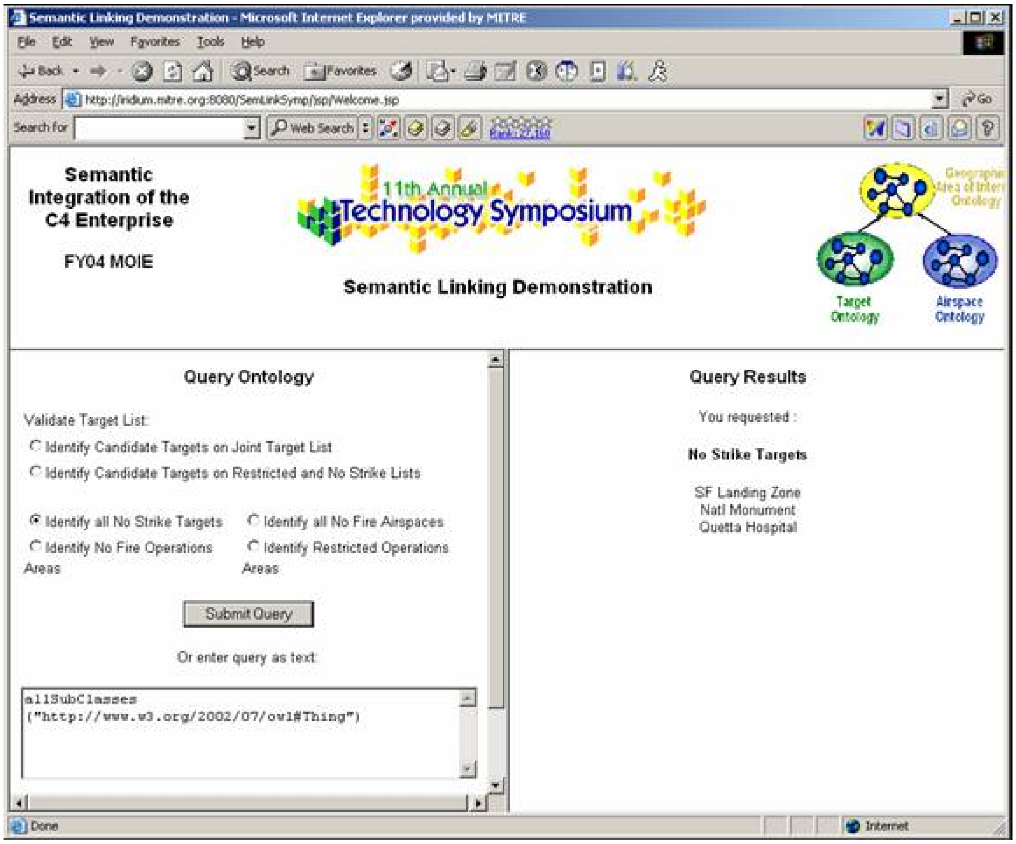


Figure 20. Cerebra Client Design

The custom client simulates an interface to a targeteer, offering the user the ability to retrieve data needed to validate targets. Our semantic linking allows the user to retrieve needed target and airspace data from simulated authoritative sources from a single interface. The client offers the user the ability to load the Airspace, Target or Reference ontologies. Once loaded, the ontology can be queried by selecting pre-canned queries or by specifying a custom query in the text window, as shown in the screen shot in Figure 21.



Address: http://iridium.mitre.org:8080/SemLinkSymp/jsp/Welcome.jsp

Semantic Integration of the C4 Enterprise

FY04 MOIE

Semantic Linking Demonstration

Query Ontology

Validate Target List:

○ Identify Candidate Targets on Joint Target List

○ Identify Candidate Targets on Restricted and No Strike Lists

● Identify all No Strike Targets

○ Identify all No Fire Airspaces

○ Identify No Fire Operations Areas

○ Identify Restricted Operations Areas

Submit Query

Or enter query as text:

```
allSubClasses
("http://www.w3.org/2002/07/owl#Thing")
```

Query Results

You requested :

No Strike Targets

SF Landing Zone
Natl Monument
Quetta Hospital

Figure 21. Custom Client Screen Shot

The client was built in Java, Java Server Pages (JSP), and Java Beans and runs under the Apache Tomcat web server. MySQL was the database server used to simulate portions of the Air Operations and Intelligence databases. The code for the custom client is available to MITRE employees on the MII Internal Source Forge site.

6 Potential Extensions

Over the course of the research we formulated many ideas for future Semantic Web experimentation. Some of the ideas are extensions to what we had planned. Others result from the inevitable descoping that occurs to fit a project within available resources. A few are simply ideas we though would be interesting to study. We include this section primarily for other Semantic Web researchers.

- Develop way to capture standard provenance information for the ontology s a whole.
- Create geographic regions ontology elements based upon a standard Point ontology that builds upon an Air Force Materiel Command Electronic Systems Center (ESC) and MITRE Cursor-on-Target (CoT) effort. See how the ontology elements would map to proposed standard upper ontologies.
- Actually link additional ontologies and data sources to our Reference Ontology to test its extensibility.
- Develop a strategy for integrating our prototype with existing target validation tool sets or with the web service interfaces to the operational databases.
- Populate an onology by mapping it to the results of a web service.
- Design an experiment to actually link to a standard upper ontology.
- Time Experimentation
    - Experiment with the use of the XML Schema built-in dataTime type to perform time calculations (e.g., is time x between time y and time z).
    - Experiment with the OWL version of the DAML Time Ontology.
    - Design an experiment to compare these approaches and then share findings and recommendations.
- Experiment with using user-defined XML Schema datatypes to restrict data values in an ontology
- Experiment with additional property restrictions (e.g., PointAirspaces equivalent to class of airspaces where hasShape=Point).
- Experiment with the use of a standard country Code ontology either from the DAML ontology library [37] or another source.

---

[37] http://www.cs.rochester.edu/~ferguson/daml/

- Experiment with a few of the interesting research tools available. We were exposed to several fascinating approaches and initiatives at the International Semantic Web Conference [38] in October 2003. Several intrigued us with their potential application to the DoD domain.

Investigate what it would take to create dynamic ontological classes that were enabled and disabled via triggers, such as the date and time of an event.

---

[38] http://www.daml.org/ontologies/

7 Conclusions

The results of this Netcentric Semantic Linking study, combined with previous Semantic Web research experience, lead us to several conclusions. These conclusions, discussed below, are divided into three categories.

7.1 Semantic Linking is Powerful but Complex

7.1.1 Semantic linking has promise but is difficult.

As a key component to the vision of a world-wide semantic network of machine interpretable information, semantic linking holds incredible promise. It holds the promise of data and service discovery as well as an ability to perform conceptual queries, potentially across an ever growing set of semantically linked domains. Eventually, as more data and services become semantically mapped, one can envision the same type of network effect occurring that exists with the current World Wide Web.

But, today semantic linking is hard. Semantic linking currently is a difficult, manual process that requires a thorough understanding of each domain to be linked, the data modeling vocabulary used (OWL in our case), and the sometimes difficult to discern implications of the mapping. It remains a research area and as such has only rudimentary tool support to assist the user. Because ontologies are developed to support a specific purpose and often use different conceptualizations of a domain, linking them after the fact is difficult. Standard upper ontologies, utility ontologies of commonly used concepts, and super domain ontologies of core domain concepts offer some hope of aiding in the semantic linking process

7.1.2 Linking via a Reference Ontology is powerful and extensible.

We found that using a reference ontology to map multiple ontologies together was a very powerful approach. Once ontologies are mapped together a user or application can retrieve information about the mapped ontologies or the data linked to them. Our queries demonstrated the ability to extract data from databases that are linked to the mapped ontologies, sometimes through the use of a uniting concept. As shown in Figure 22, one could also link new data sources to a reference ontology, be they new relational databases or other forms of structured data (e.g., from web services or XML formatted files). Mapping additional ontologies to the reference ontology introduces the possibility of discovering new information or relationships not previously discerned. Also, the ability to query on concepts (e.g., all no strike targets) rather than the physical data representation (e.g., specific SQL call) adds a more flexible and extensible capability. Finally, we found the ability to add new classes that unite concepts in different ontologies powerful and were intrigued by the potential to use this capability to join concepts that are not directly mappable, but are related from a mission perspective.

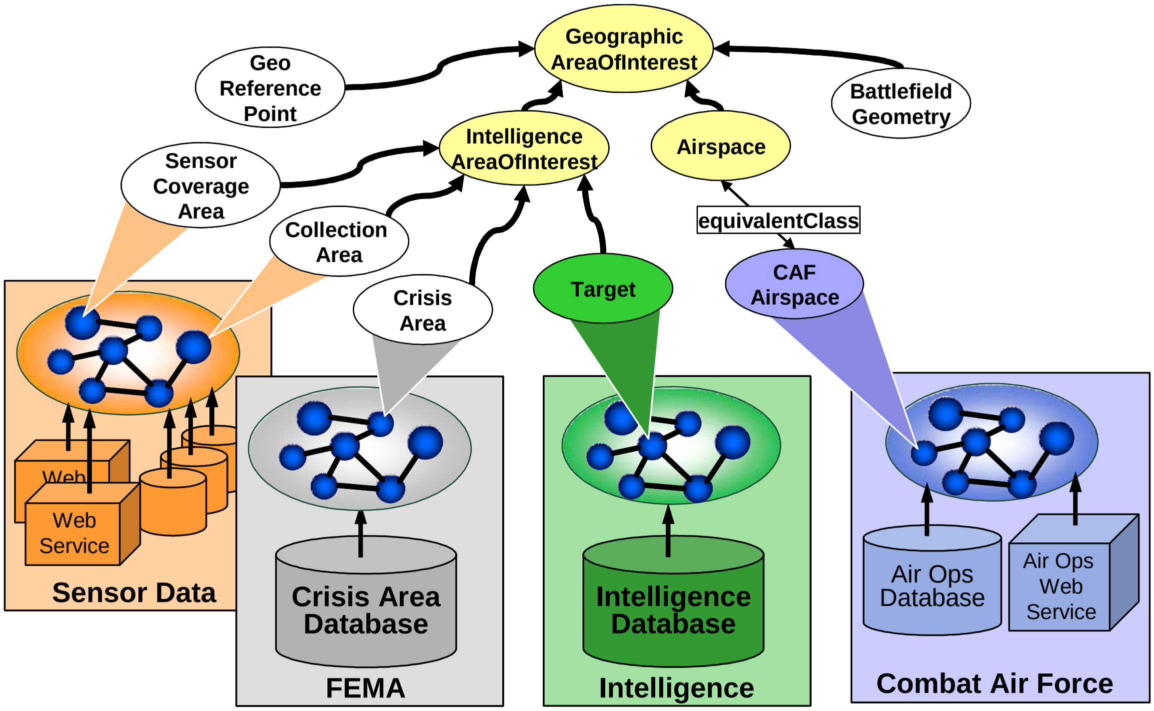


Figure 22. Extensibility of Semantic Linking

7.1.3 Domain and modeling expertise are critical.

We found that an understanding of the domain and the details of the data to be modeled, as well as strong semantic modeling expertise, are critical to the success of developing and linking ontologies. Consensus on how to model data within a domain is very challenging, especially since one wishes to model the concepts to reflect a particular usage and not necessarily to reflect how the data is stored today. Further, semantic linking across domains is even more challenging in that in depth domain expertise within each domain is required.

7.1.4 Linking ontologies to databases was more challenging than anticipated.

We found there to be a very tight coupling between ontologies and their linked databases and we challenge vendors to find a way to more loosely couple these layers. We found a number of problems in linking our ontologies to databases even in the cases in which we developed database schemas specifically for this effort. This suggests that it would even more difficult to build an ontology that links to a legacy database where the database structure cannot be changed. We consider some of the problems we identified to be “show stoppers”, preventing usage in a real application. Network Inference is aware of these issues and is beginning to address them. We also found that different usages (queries) can lead to different ontology designs. This could limit extensibility in that ontologies may have to be modified as new usages are identified.

7.2 Better Semantic Web Tools are Needed

7.2.1 Ontologies are powerful but non-trival.

The formalism of the newly standardized language OWL makes ontologies powerful and able to support machine to machine semantic interoperability. However, this same formalism also presents challenges, since significant skills in multiple areas (domain knowledge, data modeling skills and ontological engineering) are required to develop and use ontologies.

7.2.2 OWL tools are immature.

Tools to support ontology development and linking in OWL are immature. This is not surprising since OWL was just approved as an international standard in February 2004. Integrated and robust tools are needed to create, validate, manage, map and query ontologies to make these technologies more widely accessible to people and applications. More mature tools to explain how an inference engine derived an answer are also required, especially in a military domain. Despite the issues we encountered with the Network Inference tool set, we believe they offer a very promising solution suite, and are encouraged by their plans for enhancements. In fact, Network Inference created a tool that does a very good job of abstracting the details OWL into a easy to use graphical interface and deserves much credit for pioneering an OWL tool suite before the standard was approved. We are encouraged that more and more vendors are developing OWL solutions. The next strides we hope to see in Semantic Web tools are the development of standard rule and query languages.

7.3 Semantic Web is Maturing Rapidly and Warrants DoD Attention

7.3.1 Semantic Web is here to stay and maturing rapidly.

Semantic Web technologies offer great potential for future applications and are already being used by large commercial companies today. It is clear that we need not realize the entire Semantic Web vision to attain value from these technologies. Further, the tempo of change is high for Semantic Web technologies so we expect to see rapid improvements in capabilities.

7.3.2 Semantic Web will be an integral part of a Netcentric approach in the DoD.

Another conclusion we draw from our experience is that the wide applicability and potential offered by Semantic Web technologies will make them an integral part of the DoD’s Netcentric approach. While several challenges must be addressed before the entire Semantic Web vision can be realized, these technologies can be useful within a domain or community of interest today. The real power of Semantic Web technologies is embodied in the future potential of how they could be used – today by humans and simple applications and in the future by applications with more intelligence. Also, in combination with rules, ontologies could provide a large advance in capabilities. While OWL is more expressive than RDF/S, there are still some things that require rules for additional reasoning power. For example, ontologies combined with rules could significantly advance current DoD capabilities through the ability to create and link new concepts or enable dynamic service behavior, perhaps based on dynamic, real world events.

7.3.3 Semantic Web training and best practices are needed.

If Semantic Web technologies are here to stay and can be applied to our customers' needs, then we need to develop skills and best practices for using them. Training in ontology development and use would expedite the ability of the military to take advantage of these emerging technologies. We need to stay abreast of developments in these emerging technologies so we can position our customers to harness their power to increase mission effectiveness. A best practices guide on how to build and use ontologies, that extend existing work (e.g., [NM01]), would be valuable. Further, best practices or conventions within a DoD Community of Interest (COI) could make it easier to map ontologies within that COI. Finally, we believe it would be valuable to develop a standard ontology provenance approach (e.g., to define ontology source, purpose, and lineage) for use within the DoD enterprise. Our bias is to develop and use a DoD provenance ontology to be imported into each DoD domain ontology and populated by the ontology developers. Use of a DoD provenance ontology would allow applications to reason over the provenance data for use in ontology discovery or ontology maintenance (e.g., check versioning).

7.3.4 Developing ontologies with reuse in mind will pay off in the long term.

With a growth in ontology development, reuse of ontologies will be essential to fully leverage the power of the Semantic Web. Analogous to software libraries, as ontologies mature, we envision the emergence of a library of ontologies. From this library, ontology designers can compose their domain ontologies, inheriting concepts and inferencing capabilities provided by these utility ontologies. This is evident in existing efforts to define a standard time ontology.[39] Furthermore, upper or foundational ontologies will serve as a theoretical foundation for domain ontology developers. Foundational ontologies will allow developers to build upon the knowledge and experience already captured in the foundational ontology, increasing the likelihood of developing a semantically rich domain ontology.

---

[39] http://www.daml.org

8 Recommendations

Our recommendations regarding using Semantic Web technologies in a Netcentric approach are directed to MITRE and our government sponsors.

8.1 Monitor innovations in Semantic Web tools and standards.

Support MITRE and DoD participation in W3C working groups and related conferences. Form relationships with vendors and periodically test upgrades and enhancements to tools. Organize technical exchange meetings on Semantic Web technologies and applications.

8.2 Invest in training on Semantic Web technologies.

Support development of classes in this technology and provide the opportunity to advance the education of MITRE and government employees as well as our customers to position ourselves to take advantage of these advances. Encourage contractors to invest in training and prototyping of government solutions using these technologies.

8.3 Support development of Ontology Best Practices for the DoD community.

Fund a small effort to develop Ontology best practices for the DoD. This effort should also include a standard framework for specifying provenance information for DoD ontologies. This effort should also include participating in relevant groups such as the W3C Semantic Web Best Practices and Deployment Working Group.[40]

8.4 Consider how to apply Semantic Web technology in the development of future mission capabilities.

Consider Semantic Web technologies for near-term capabilities or applications to be deployed in two or more years. Small steps today could position programs to harness the power of these technologies as they mature. Support prototyping and experimentation using these technologies. Select a candidate future application to demonstrate how this technology can support a Netcentric enterprise.

8.5 Develop ontologies with reuse in mind.

When developing ontologies, make an active attempt to leverage existing ontologies wherever reasonable. Furthermore, develop your own ontologies to maximize the likelihood of reuse.

---

[40] http://www.w3.org/2001/sw/BestPractices/

References

[BHL01] T. Berners-Lee, T., J. Hendler, and O. Lassila, “The Semantic Web,” Scientific American, vol. 284, no. 5, pp. 34–43, May 2001.

[Bou03] Anne-Claire Boury-Brisset, “Ontology-based Approach for Information Fusion”, 2003. http://www.infofusion.buffalo.edu/conferences_and_workshops/ontology_wkshop_2/ont_ws2_working_materials/BouryBrissetOntologyandFusion.PDF

[Bou04] C. Boulton, “Semantic Web to Take Center Stage at WWW2004”, May 16, 2004, http://www.internetnews.com/dev-news/article.php/3354651.

[Col02] R.M. Colomb, “Use of Upper Ontologies for Interoperation of Information Systems: A Tutorial”, Technical Report 20/02 ISIB-CNR, Padova, Italy, November 2002.

[Doa03] Doan, AnHai, Jayant Madhavan, Robin Dhamankar, Pedro Domingos, Alon Halevy, “Learning to Match Ontologies on the Semantic Web”, The VLDB Journal, Volume 12, Number 4, November 2003, 303-319.

[DOS03] Daconta, Michael C., Leo J. Obrst and Kevin T. Smith, “The Semantic Web”, Wiley Publishing, 2003.

[GFC04] Gómez-Peréz, Asunción, Mariano Frenández-López and Oscar Corcho, “Ontological Engineering”, Springer-Verlag London, 2004.

[Gru93] T.R. Gruber, “A Translation Approach to Portable Ontology Specifications,” Knowledge Acquisition, vol. 5, issue 2, pp. 199-220, June 1993.

[KDS01] Kiryakov, A., K. Simov, M. Dimitrov, “OntoMap: Portal for Upper-level Ontologies,” Proceedings of the international conference on Formal Ontology in Information Systems – Volume 2001, pp 47-58.

[Kle01] M. Klein, “Combining and relating ontologies: an analysis of problems and solutions”, Proceedings of the IJCAI-01 Workshop on Ontologies and Information Sharing, Seattle, USA, August 4-5, 2001, pp 53-62.

[KS03] Kalfoglou, Yannis and Marco Schorlemmer, “Ontology Mapping: the state of the art”, The Knowledge Engineering Review, Vol. 18:1, 1-31, Cambridge University Press, 2003.

[MAB03] Maier, Andreas, J. Aguado, A. Bernaras, I. Laresgoiti, C. Pedinaci, N. Peña, and T. Smithers, “Integration with Ontologies”, Conference paper WM2003, April 2003, Luzern. http://www.ontoprise.de/documents/integration_with_ontologies.pdf

[NK02] Noy, N.F. and M. Klein, “Ontology Evolution: Not the Same as Schema Evolution”, Also as: Smi-2002-0926, University of Stanford, Stanford Medical Informatics, Knowledge and Information Systems, 2002.

[NM00] Noy, N.F. and M.A. Musen, “PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment”, Proceedings of the National Conference on Artificial Intelligence (AAAI), 2000.

[NM01] Noy, N.F and D.L. McGuinness, “Ontology Development 101: A Guide to Creating Your First Ontology”, Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880, March 2001.

[OLW03] Obrst, Leo, Howard Liu, and Robert Way, “Ontologies for Corporate Web Applications”, AI Magazine, Fall 2003.

[Phy02] Phytila, Chris, “An Analysis of the SUMO and Description in Unified Modeling Language”, April 2002.

[SR03] Silva, Nuno and João Rocha, “Ontology Mapping for Interoperability in Semantic Web”, Proceedings of the IADIS International Conference WWW/Internet 2003 (ICWI'2003), Algarve, Portugal, November 2003.

[W3C04] *OWL Web Ontology Language Use Cases and Requirements*, W3C Recommendation, 10 February 2004, available online at: http://www.w3.org/TR/2004/REC-webont-req-20040210/

[WVV01] H. Wache, T. Voegele, U. Visser, H. Stuckenschmidt, G. Schuster, H. Neumann and S. Huber, “Ontology-Based Integration of information – A Survey of Existing Approches,” Proceedings of the IJCAI-01 Workshop on Ontologies and Information Sharing, Seattle, USA, August 4-5, 2001, pp 108-118.

Appendix A Ontology Queries

This This appendix provides samples of Network Inference queries in their XQuery syntax along with the XML result returned by Network Inference Cerebra Server. The syntax documented here was valid as of Cerebra Server version 3.0, November 2003. The syntax has been modified in subsequent releases.

Cerebra Server v3.0 provided the ability to perform taxonomic queries as well as conjunctive queries. A taxonomic query answered questions about classes and individuals which are in the ontology. Table 1 is an example of a taxonomic query. A conjunctive query was used to query instances stored in a database. Table 2 is an example of a conjunctive query.

Table 3 contains the queries we used to support our semantic linking demonstration.

Table 1. Sample Taxonomic Query

Taxonomic Query: What are the subclasses of the *Target* class?

NI Xquery:

```
allSubClasses (“http://xml.dod.mil/ontology#Target”)
```

XML Result:

```
<RESULT>
    <allSubClasses>
        <equivalent>
            <class>http://www.w3.org/2002/07/owl#Nothing</class>
        </equivalent>
        <equivalent>
            <class>http://xml.dod.mil/ontology#RestrictedTarget</class>
        </equivalent>
        <equivalent>
            <class>http://xml.dod.mil/ontology#NoStrikeTarget</class>
        </equivalent>
        <equivalent>
            <class>http://xml.dod.mil/ontology#JointTarget</class>
        </equivalent>
        <equivalent>
            <class>http://xml.dod.mil/ontology#CandidateTarget</class>
        </equivalent>
        <equivalent>
            <class>http://xml.dod.mil/ontology#Target</class>
        </equivalent>
    </allSubClasses>
</RESULT>
```

Table 2. Sample Conjunctive Query

Conjunctive Query example: Identify all *Targets* in the ontology.

NI XQuery:

```
default element namespace = "http://xml.dod.mil/ontology#"
for $i in concept (Target)
return $i/name
```

XML Result:

```
<RESULT>
    <models>
            <model>
                    <variable>
                            <property>http://xml.dod.mil/ontology#name</property>
                            <datatype>String</datatype>
                            <value>Tigris Bridge</value>
                    </variable>
            </model>
            <model>
                    <variable>
                            <property>http://xml.dod.mil/ontology#name</property>
                            <datatype>String</datatype>
                            <value>SF Landing Zone</value>
                    </variable>
            </model>
            <model>
                    <variable>
                            <property>http://xml.dod.mil/ontology#name</property>
                            <datatype>String</datatype>
                            <value>Natl Monument</value>
                    </variable>
            </model>
            <model>
                    <variable>
                            <property>http://xml.dod.mil/ontology#name</property>
                            <datatype>String</datatype>
                            <value>Petroleum Factory</value>
                    </variable>
            </model>
            <model>
                    <variable>
                            <property>http://xml.dod.mil/ontology#name</property>
                            <datatype>String</datatype>
                            <value>Pipeline and Storage</value>
                    </variable>
            </model>
            <model>
                    <variable>
                            <property>http://xml.dod.mil/ontology#name</property>
                            <datatype>String</datatype>
                            <value>Bagh-e Daud Airfield</value>
                    </variable>
            </model>
            <model>
                    <variable>
                            <property>http://xml.dod.mil/ontology#name</property>
                            <datatype>String</datatype>
                            <value>Quetta Hospital</value>
                    </variable>
            </model>
            <model>
                    <variable>
                            <property>http://xml.dod.mil/ontology#name</property>
                            <datatype>String</datatype>
                            <value>Salman Farsi Camp</value>
                    </variable>
            </model>
    </models>
</RESULT>
```

Table 3. Semantic Linking Demonstration Queries

1. Identify No Strike Targets.

NI XQuery:

```
default element namespace = "http://xml.dod.mil/ontology#"
for $i in concept (NoStrikeTarget)
return $i/name
```

2. Identify No Fire Airspaces.

NI XQuery:

```
default element namespace = "http://xml.dod.mil/ontology#"
for $i in concept (NoFireArea)
return $i/name
```

3. Identify No Fire Operations Areas. (links the concepts of No Strike Targets and no Fire Airspaces)

NI XQuery:

```
default element namespace = "http://xml.dod.mil/ontology#"
for $i in concept (NoFireOperationsArea)
return $i/name
```

4. Identify Restricted Operations Areas. (links the concepts of Restricted Target and Restricted Operations Zone)

NI XQuery:

```
default element namespace = "http://xml.dod.mil/ontology#"
for $i in concept (RestrictedOperationsArea)
return $i/name
```

5. Identify Candidate Targets on the Joint Target List.

NI XQuery:

```
default element namespace = "http://xml.dod.mil/ontology#"
for $i in concept (CandidateTarget)
for $j in concept (JointTarget)
where $i/name = $j/name
return $i/name
```

6. Identify Candidate Targets on the Restricted and No Strike Lists.

NI XQuery:

```
default element namespace = "http://xml.dod.mil/ontology#"
for $i in concept (CandidateTarget)
for $j in concept (unionOf(RestrictedTarget,NoStrikeTarget)))
where $i/name = $j/name
return $i/name
```

Glossary

ACO	Airspace Control Order
AODB	Air Operations Database
AOI	Area of interest
API	Application Programming Interface
C2	Command and Control
C2EIDM	C2 Information Exchange Data Model
COI	Communities of Interest
DAML	DARPA Agent Markup Language
DARPA	Defense Advanced Research Projects Agency
DevNet	Developer’s Network
DMPI	Designated Mean Point of Impact
DOLCE	Descriptive Ontology for Linguistic and Cognitive Engineering
FY	Fiscal Year
JSP	Java Server Pages
JTL	Joint Target List
MIDB	Modernization Integrated Data Base
MIP	Multilateral Interoperability Programme
OWL	Web Ontology Language
RDF	Resource Description Framework
RDF/S	RDR and RDF Schema
SQL	Structured query language
SUMO	Suggested Upper Merged Ontology
SUO WG	Standard Upper Ontology Working Group
TBMCS	Theater Battle Management Information Core Systems
UNSPC	Universal Standard Products & Services Classification
URI	Uniform Resource Identifier
USMTF	United States Message Text Format
W3C	World Wide Web Consortium
XML	Extensible Markup Language