SEMANTIC TECHNOLOGY FOR
INTELLIGENCE, DEFENSE, AND SECURITY

STIDS2012

Semantic Technologies in Cyber Security

THE 7TH INTERNATIONAL CONFERENCE
ON SEMANTIC TECHNOLOGIES
OCTOBER 23 - 26, 2012
Mason Inn Conference Center
George Mason University
Fairfax, Virginia Campus

Conference Proceedings

Paulo C. G. Costa
Kathryn B. Laskey
(Eds.)

GEORGE MASON UNIVERSITY
C4I CENTER
DATA TACTICS CORPORATION
YarcData A CRAY COMPANY
KADSci Knowledge & Decision Sciences

Preface

The 7th International Conference on Semantic Technologies for Intelligence, Defense, and Security (STIDS 2012) provides a forum for academia, government and industry to share the latest research on semantic technology for defense, intelligence and security applications.

Semantic technology is a fundamental enabler to achieve greater flexibility, precision, timeliness and automation of analysis and response to rapidly evolving threats. This year we have the following topics:

- Creating an interoperable suite of public-domain ontologies relevant to intelligence analysis covering diverse areas:
- Ontologies and reasoning under conditions of uncertainty
- Semantic technology and ontological issues related to:
    - Source credibility and evidential pedigree
    - Use of sensing devices including security, e.g. global infrastructure grid (GIG),
      images and intelligence collection in general
- Usability issues relating to semantic technology
- Best practices in ontological engineering

Fairfax, VA, October 2012.

Paulo Costa and Kathryn Laskey
STIDS 2012 Chairs

STIDS 2012 Committees

STIDS 2012 Program Committee

Rommel Carvalho	George Mason University
Werner Ceusters	SUNY at Buffalo
Paulo Costa	George Mason University
Mike Dean	Raytheon BBN Technologies
Katherine Goodier	L3 Technologies
Richard Haberlin	EMSolutions Inc.
Terry Janssen	Lockheed Martin
Mieczyslaw Kokar	Northeastern University
Kathryn Laskey	George Mason University
Nancy Lawler	US Department of Defense
Dan Maxwell	KaDSci, Inc.
Leo Obrst	MITRE Corporation
Mary Parmelee	MITRE Corporation
Barry Smith	SUNY at Buffalo
Simon Spero	UNC Chapel Hill
Gheorghe Tecuci	George Mason University
Andreas Tolk	Old Dominion University
Brian Ulicny	Vistology, Inc.
Amanda Vizedom	Wind River Consulting, LLC
Duminda Wijesekera	George Mason University

Best Paper Award Committee

Amanda Vizedom (Chair)	**Wind River Consulting, LLC**
Mike Dean	Raytheon BBN Technologies
Simon Spero	UNC Chapel Hill
Andreas Tolk	Old Dominion University
Brian Ulicny	Vistology, Inc.

STIDS Steering Committee

Paulo Costa	George Mason University
Mike Dean	Raytheon BBN Technologies
Katherine Goodier	NIC, Inc.
Kathryn Laskey	George Mason University
Leo Obrst	MITRE Corporation
Barry Smith	NCOR, Univ. at Buffalo

STIDS 2012 Organizing Committee

General Chairs:

Paulo Costa	George Mason University
Kathryn Laskey	George Mason University

Local Team (GMU)

Debra Schenaker (Administrative Chair)
Priscilla McAndrews
Nicholas Clark
Alexandre de Barros Barreto
Cheol Youn Park
Karen Tai

STIDS 2012 Platinum Sponsor

DATA TACTICS CORPORATION

Data Tactics Corporation (DTC) has been developing and implementing mission-focused capabilities to the Intelligence Community and DOD for years; providing superior service and leading innovation. Whether it is data mining, data correlation, data retrieval, information security or cloud computing, Data Tactics understands the challenges that face our client-base and our peers across the industry. With our vast knowledge, professional expertise and dedication, Data Tactics is prepared and committed to designing, implementing and sustaining customized solutions to meet the customers' mission requirements.

Data Tactics Corporation is a small business solely focused on mission –relevant solutions that bring industry recognized experts in the field of Specialized Cloud Enterprise Architecture, Cyber Security, Geospatial Engineering, System / Software Development, Data / System Integration, and Operations and Maintenance (O&M) / Sustainment support. We measure that support at our end-user. The staff is qualified to identify report, resolve and support a myriad of complex data, storage, security and system problems. Our success has been proven time and again at traditional sites but also in tactical forward deployed environments.

Our Mission

- To Design, Develop, Deliver and Manage State-of-the-Art Technological Capabilities for Our Client's Enterprise that Supports Our Client's Mission Objectives
- To See Our Performance across the Service Lifecycle through Our Client's Lens.
- Our Work Contributes to Our Client's Success because we Design, Deliver and Sustain those Services to Work in the Client's Environment, by Client's Personnel to Achieve Client Success

Vision Statement

- To Establish an Enduring Relationship of Trust with Our Client Based Solely on Our Performance
- To Deliver a Product or Service that becomes Second-Nature to Our Client's Personnel and a Seamless Part of Our Client's Business Operations
- To Remain a Creative, Disruptive and Leading Research, Development and Rapid Deployment Institution Where Our Shared Intellect, Hard Work and Vanguard for Our Client's Trust make a Positive Difference in the Lives of Our Employees, the Success of Our Clients and the Security of Our Country

STIDS 2012 Golden Sponsor

Big Data is new, but we're not. We've got years of experience in data management and a reputation for hardware performance and reliability that stretches back decades, with thousands of man-years of investment into this technology. As a Cray company, YarcData has the resources to provide the highest performance processing capabilities and visionary data management resources – it means our Big Data appliance for graph analytics is innovative, but solid.

It's an exciting time. We're new, but we're established, and we're ready to change the way you value and leverage Big Data.

STIDS 2012 Best Paper Award Sponsor

KaDSci, LLC is small veteran owned company that was founded in 2008 with the goal of finding and providing solutions to the nation's and industries' most vexing decision related research, technology, and analysis challenges. Toward that end KaDSci has assembled a small highly skilled team of collaborating scientists from many disciplines as well as technically savvy and creative professionals with a passion for solving hard problems.

STIDS 2012 Keynote Speaker

Mike Dean

A Pragmatic View of the Semantic Web and Ontologies

Wednesday, October 24, 2012

Mike Dean is a Principal Engineer at Raytheon BBN Technologies where he's worked for 28 years in distributed computing, command and control applications, information assurance, and (since 2000) the Semantic Web. He was Principal Investigator for Integration and Transition in the DARPA Agent Markup Language (DAML) Program, which catalyzed the Semantic Web, and has served on W3C Working Groups for RDF, OWL, and RIF. He continues to develop, lead, or consult on the development of Semantic Web tools, ontologies, datasets, and applications for DoD, IC, and commercial customers. He holds a B.S. in Computer Engineering from Stanford University.

Abstract: I continue to be excited about the accomplishments and potential of the Semantic Web and related technologies. In this talk, I'll review some of those accomplishments, present some applications, discuss various emerging technologies (Wikidata, GeoSPARQL, SILK, PROV, ontology design patterns, Big Data, stream processing, etc.), identify some things that are still missing, and perhaps offer some predictions for the future.

STIDS 2012 Keynote Speaker

Jay Holcomb

An Insider Perspective of how Semantic Technologies are supporting the DHS efforts in Cybersecurity

Thursday, October 25, 2012

Jay Holcomb recently joined the Department of Homeland Security's National Protection & Programs Directorate (NPPD), Office of Cybersecurity and Communications (OC&C), Critical Infrastructure Cyber Protection and Awareness (CICPA) department. He is an internationally renowned expert in Cybersecurity and the Program Lead for Cyber Integration at the Control Systems Security Program/ICS-CERT. His presentation will bring an insider perspective of how Semantic Technologies are supporting the DHS efforts in Cybersecurity.

STIDS 2012 Best Paper Award

The Best Paper Award is meant to recognize the excellence of our technical program and contributors, as well as to promote the continuing efforts of our community to push forward the state of the art in Semantic Technologies for Intelligence, Defense, and Security.

The award was presented to the authors of the paper chosen by the STIDS 2012 awards committee as the best contribution appearing in the conference proceedings. Dr. Daniel Maxwell, President of KaDSci, Inc., announced the award at a special session on Thursday, October 25. The awardees received a check in the value of US$450.

Best Paper Award Winner of STIDS 2012

Ontological Considerations for Uncertainty Propagation in High Level Information Fusion

by Mark Locher and Paulo Costa

Honorable Mentions

The following two papers also made it to the last phase of the selection process, and were both highly recommended by the award committee for an honorable mention as outstanding contributions to the conference (order is irrelevant):

Best-practice Time Point Ontology for Event Calculus-based Temporal Reasoning

by Robert Schrag

Using Ontologies in a Cognitive-Grounded System: Automatic Action Recognition in Video-Surveillance

by Alessandro Oltramari and Christian Lebiere

Table of Contents

Technical Papers

Position Papers

Technical Papers

Ontological Considerations for Uncertainty Propagation in High Level Information Fusion

Mark Locher
George Mason University and SRA, International
Fairfax VA USA
mlocher@gmu.edu

Paulo C. G. Costa
George Mason University
Fairfax VA USA
pcosta@gmu.edu

Abstract— **Uncertainty propagation in a level 2 high level information fusion (HLIF) process is affected by a number of considerations. These include the varying complexities of the various types of level 2 HLIF. Five different types are identified, ranging from simple entity attribute refinement using situation status data to the development of a complete situation assessment assembled from applicable situational fragment data. Additional considerations include uncertainty handling in the input data, uncertainty representation, the effects of the reasoning technique used in the fusion process, and output considerations. Input data considerations include the data's relevance to the situation, its credibility, and its force or weight. Uncertainty representation concerns follow the uncertainty ontology developed by the W3C Incubator Group on Uncertainty Reasoning. For uncertainty effects of the fusion process, a basic fusion process model is presented, showing the impacts of uncertainty in four areas. Finally, for output uncertainty, the significance of a closed-world versus open-world assumption is discussed.**

Keywords - High level fusion, input uncertainty, process uncertainty, output uncertainty, uncertainty representation

I. INTRODUCTION

The past 20 years have seen an explosion of systems and techniques for collecting, storing and managing large and diverse sets of data of interest to a number of communities. These data are collected by a wide variety of mechanisms, each of which has varying considerations that influence the uncertainty in the data. In order to provide useful information for a particular question or problem, the relevant data ("evidence") must be identified, extracted and then fused to provide insight or answers to the question or problem. The information fusion community has developed a widely accepted functional layered model of information fusion. These layers can be divided into low level and high-level fusion. At all levels, the data going into a fusion process is recognized as having uncertainty, which affects in various ways the degree of certainty in the output of the process. Low-level fusion has been widely explored, primarily through the radar tracking community, and issues of uncertainty determination and propagation are well understood [1].

High-level fusion, on the other hand, requires reasoning about complex situations, with a diversity of entities and various relationships within and between those entities. This reasoning is often expressed symbolically, using logic-based approaches [2]. There has been significant work in using ontological approaches in developing fusion techniques and some of these approaches have taken uncertainty considerations into account (e.g. [3] [4] [5] [6]). Various techniques exist to model and propagate uncertainty in a fusion process, with varying strengths and difficulties. This suggests that their relative performance in a fusion system should vary significantly depending on the types and nature of the uncertainties within both the input data and the context of the problem set modeled with the fusion system. Unfortunately, there is no consensus within the fusion community on how to evaluate the relative effectiveness of each technique. Work in this area will be hampered until the evaluation question is at least better defined, if not resolved.

The International Society for Information Fusion (ISIF) chartered the Evaluation of Technologies for Uncertainty Reasoning Working Group (ETURWG) to provide a forum to collectively address this common need in the ISIF community, coordinate with researchers in the area, and evaluate techniques for assessing, managing, and reducing uncertainty [7]. In its first year, ETURWG defined its scope and developed the uncertainty representation and reasoning evaluation framework (URREF) ontology. The URREF ontology aims to provide guidance for defining the actual concepts and criteria that together comprise the comprehensive uncertainty evaluation framework [8]. It is evident that part of the issue in evaluating different uncertainty representation systems is to properly understand how a high-level fusion process works and how uncertainty is propagated through the process.

This paper aims to help establish the various considerations about how uncertainty affects a HLIF process. It will begin by defining what is meant by a HLIF process, and then focus on one class of HLIF, the level 2 HLIF. From there, it will define a taxonomy of Level 2 HLIF, where increasing complexity of level 2 HLIF types have additional uncertainty considerations. Then it explores uncertainty propagation issues associated with uncertainty in the input data, the uncertainty effects of both the fusion reasoning process and the representation scheme, and the output uncertainty. It concludes with a top-level discussion of an overall mathematical approach applicable to these considerations.

II. DEFINITION OF HIGH-LEVEL FUSION

A widely accepted definition of High-Level Information Fusion (HLIF) is that it refers to the fusion processes classified as level 2 and above within the revised Joint Directors of Laboratories data fusion model. This model establishes five functional levels, as defined in [9] and repeated in Table 1 below.

Table 1: JDL Fusion Levels [9]

Level	Title: Definition
0	**Signal / Feature Assessment: Estimate signal or feature state. May be patterns that are inferred from observations or measurements, and may be static or dynamic, and may have locatable or causal origins**
1	**Entity Assessment: Estimation of entity parametric and attributive states (i.e. of individual entities)**
2	**Situation Assessment: Estimate structures of parts of reality (i.e. of sets of relationships among entities and implications for states of related entities.)**
3	**Impact Assessment: Estimate utility/cost of signal, entity or situation states, including predicted utility / cost given a system's alternative courses of action**
4	**Process Assessment: A system's self-estimate of its performance as compared to desired states and measures of effectiveness.**

A key item is that these assessments are not just a combination of information, but they are also analytic judgments. For example, a level 2 fusion process is more than a unified display of information (e.g. a common operational picture); rather, it requires explicit statements about how certain specific elements of reality are structured, in order to address specific questions that a user of that process wants answered. Level 2 fusion essentially answers the question "what is going on?" Level 3 fusion addresses "what happens if …?", where "if" is followed by a possible action or activity (level 3 is often predictive). Level 4 involves steering the fusion system, including adjusting data collection based on an assessment of already-collected data. There has been some discussion regarding the boundary between level 1 and level 2. Das, for instance, considers identification and object classification as beyond level 1, suggesting that this type of fusion should be a level 1+ [10]. Steinberg, on the other hand, considers this to be clearly level 1 [9]. Sowa's ontological categories provide insight into this question, and can be used to illuminate some factors on uncertainty propagations considerations. In the present work, these ontological categories were used as a basis for defining a taxonomy of level 2 fusion.

III. TAXONOMY OF LEVEL 2 HLIF

Sowa defined twelve ontological categories, and together they comprise a very attractive framework for analyzing fusion processes at level 2. He suggests that one way of categorizing entities in the world is to consider them from three orthogonal aspects [11]. The first is whether they are physically existing or abstract. Abstract entities are those that have information content only, without a physical structure. This includes the idea of geometric forms or canonical structures (e.g. idea of a circle), or entities like computer program source code.

The second aspect defining the ontological categorization is whether the entity is a continuant (i.e., having time-stable recognizable characteristics) or an occurrent (i.e., significantly changing over time). This means that an entity can either be an object (a continuant) or a process (an occurrent – also called an event). The third and final aspect of his ontological categorization is the degree of interrelatedness with other objects and processes. At the independent level, an entity is considered by itself, without reference to other entities. At the relative level, an entity is considered in single relation to another entity. Finally, the idea of mediating takes into account two items: the number and complexity of the various interrelationships among the entities, and the unifying idea – its purpose or reason – that allows one to define a situation or a structure that encompasses the relevant entities [11].

The combination of these three aspects results in the 12 ontological categories shown in Table 2. Table 3 provides a more detailed definition of each ontological category and provides some examples.

A key point in looking at this ontological categorization is that one must understand the context and viewpoint from which a given entity is categorized, and that changes to either of these two might result in different categorizations for the same entity. To illustrate this point, an airplane can be considered as either an independent object flying in the air, or a complex mediating structure with thousands of component objects and processes that work together for the purpose of achieving aerial flight. The viewpoint one takes depends on the context one is interested in. In the airplane example, it depends on whether one is tracking a particular aircraft using a variety of sensors, or attempting to determine the various capabilities of a new aircraft type.

Table 2: Sowa's Categories [11]

	Physical		**Abstract**	
	Continuant	**Occurrent**	**Continuant**	**Occurrent**
Independent	**Object**	**Process**	**Schema**	**Script**
Relative	**Juncture**	**Participation**	**Description**	**History**
Mediating	**Structure**	**Situation**	**Reason**	**Purpose**

It is tempting to suggest that Sowa's three relationship levels correspond to the JDL levels 1 / 2 / 3 (i.e., Independent, Relative, and Mediating, respectively). However, this has at least three major problems. First, Sowa's relative level is focused on a single relationship between two entities, while JDL level 2 can (but does not have to) consider multiple relationships in and between multiple entities. Second, JDL level 2 situation assessment includes making assessments about the purpose or reason for the situation. This reason or purpose is the key characteristic that distinguishes one situation from another. A raucous sports team victory celebration, a protest and a riot share many entities and relationships, but understanding the reason/purpose behind it can make a significant difference to a chief of police. Third, there are level 1 inferences that depend on the existence of fixed relationships between entities.

To illustrate the latter point above, consider the case of an intercepted radar signal that has been classified as having come from a specific type of radar system. Now let us suppose that the radar type is tightly associated with a larger system, such as the AN/APG-63 radar on older versions of the US F-15 aircraft [12]. If one has detected the APG-63 radar, one also has very high confidence that one has detected an F-15 aircraft. This F-15 object identification occurs because there is a fixed relationship between the two objects (it's not a 100% relationship, as the APG-63 is also installed on fourteen United

States Customs and Border Protection aircraft [13]). This situation is a clear example of a fixed relationship between entities (i.e., AN/APG-63 used in F-15 fighters) that supports a level 1 object identification, thus making it applicable to directly associate JDL level 1 to Sowa's Independent relationship.

Table 3: Definitions [11]

	Definition	Examples
Object	**Any physical continuant considered in isolation**	**Any specific existing item (e.g. car serial number 123, etc.)**
Process	**The changes that occur to an object over time, with a focus on the changes**	**Explosion, most action verbs**
Schema	**The form of an continuant**	**Circle, language concepts for classes of objects (e.g. cat, airplane)**
Script	**The time or time-like sequence of an occurrent**	**Process instructions, software source code, radar track file**
Juncture	**Time-stable relationship between two objects**	**Joint between two bones, connection between parts of a car**
Participation	**Time-varying relationship between two objects, or a process related to an object**	**Artillery firing a shell, radio communication between two people**
Description	**An abstraction about the types of relationships that can exist between continuants**	**The idea behind concepts like "join", " "separate", "works for", "mother of", etc.**
History	**The recorded information about an occurrence as it relates to one or more continuants**	**Video file of a traffic intersection**
Structure	**A complex continuant with multiple sub-continuants and many relationships. Focus is on the stability of the continuant**	**Composition of an army, layout of a chemical plant**
Situation	**A complex occurrent with multiple continuants and many relationships. Focus is on the time sequence of changes among the objects and processes**	**A birthday party, road traffic in a metropolitan area**
Reason	**The intention behind a structure**	**Differentiates a chemical weapon factory from a fertilizer factory**
Purpose	**The intention driving a situation**	**Intention that differentiates going to war from conducting a military exercise**

Now consider the case where the radar is associated with a Surface-to-Air Missile (SAM) system, such as the Tin Shield acquisition radar and the SA-10 Grumble SAM system. The SA-10 system consists of multiple separate vehicles, not a single vehicle. The radar vehicle is physically separate from the other vehicles. It is possible for the Tin Shield radar to be used as a stand-alone search radar [14]. In this case, detection of the Tin Shield radar signal may indicate the presence of the SA-10, but it may not.

A key differentiator between JDL levels 1 and 2 is the focus on an object versus on multiple objects in relationship to each other. Yet, as illustrated by the two later examples, a JDL level 1 assessment can use techniques that are grounded in Sowa's relative level. In general, determining an object's level 1 attributes and states often depends on fusing different sensor outputs of processes that an object has undergone – thus making use of participation level information.

Using Sowa's categories, one can create the taxonomy of level 2 situations shown in Figure 1. This taxonomy ranges widely in complexity and analytic inferences required. There are five cases presented in the Figure, each created by first determining whether one is dealing with a known situation, or whether the situation itself must be inferred. In general, the least complex case is for known situations where one is determining / refining the attribute of an entity. This case straddles the level 1 / 2 line. It is object / process identification where the relationship between elements within the object of interest may vary. An example is the radar / vehicle case above. The defined situation is that a Tin Shield radar has been detected at a particular location. The question is whether an SA-10 battery (a higher level object) is at that location, or whether the radar is operating in a stand-alone mode (whether operationally, for system testing, or for system maintenance). The inferences generally are based on schema-based evidential reasoning (e.g. "there is a 95% chance that this radar will be associated with an SA-10 battery in its immediate vicinity").

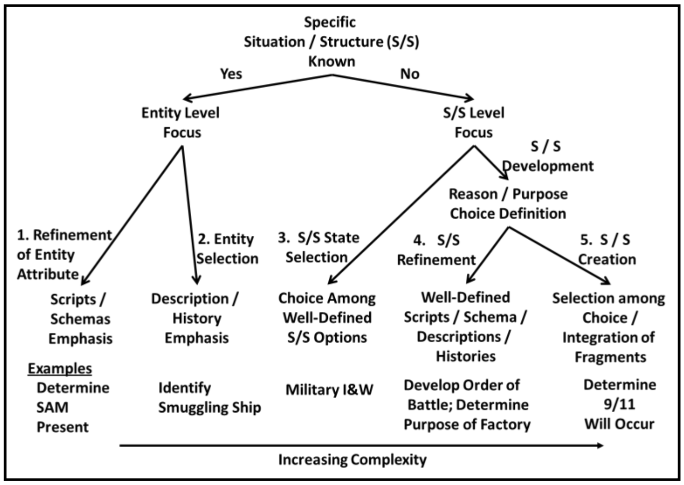


Figure 1: Types of Situation Assessments

The second case is a step up in complexity, where the situation is well defined but the objective is to identify a specific object of interest within the situation. For example, one might have very credible evidence that a terrorist group will attempt to smuggle a radiological bomb into the United States via a freighter. In this case, the situation itself is known (one knows the purpose / intention), but the actors may be hidden. Inferring which freighter (an object identification) is a likely carrier of the bomb is the question of interest. Another example would be to determine who committed a robbery of a bank, when one has a video of the act itself (the situation is a robbery). In this case, the evidence is extracted from a variety of sources, which can be classified as being junctures, participations, histories or descriptions.

The inferential process generally becomes more complex when the specific situation itself is not known, but must be inferred. The taxonomy outlines three such cases, each with an increasing level of complexity. The first is when the specific situation is not known, but there is a set of well-defined situation choices to select from. This case is a situation version of a state transition. A classic example is the military indications and warning question, which can be raised when an increase in activity at military locations in a country is detected. The question then becomes "what is the purpose of the activity?" Four major choices exist: a major military exercise, suppression of domestic unrest, a coup d' etat, or preparing to go to war. Each is a relatively well-defined situation with known entities, attributes and relationships. The selection among them becomes a pattern-matching exercise.

The next level of complexity occurs when not only is the situation itself unknown, the situation itself must be developed. Unlike the case above, the issue now is not choosing among a set of possible situations but to build the situation from the data. This case can be divided into two subcases. In the first subcase, one has a series of templates that can be used in developing aspects of the situation. For example, in developing an enemy order of battle for a standing nation-state's military, one has a basic understanding of the objects and relationships that constitute a modern military force. A country may not have all of the elements, and the organizational structure will vary. Yet, it is very likely that the structure and deployment will follow patterns similar to those used by other countries.

The second subcase is the most complex situation. Here, one must develop a situation where the basic purpose itself must be determined. For example, consider the case when a government agency is notified that something is significantly amiss, with enough information to spark interest, but not enough to understand what is happening. In that case, the evidence must be assembled without a common template to guide the fusion. Rather, the evidence must be fused using fragmentary templates, that themselves must be integrated to provide the overall situation. Integrating the data to "connect the dots" that could have predicted the September 11, 2001 commercial airliner strikes on the World Trade Center and the Pentagon falls into this category. Note also that this case also straddles the level 2 / level 3 fusion line, since determining the purpose in this case has a predictive element with possible courses of actions and outcomes.

IV. UNCERTAINTY PROPAGATION IN HLIF

In any fusion process, one follows a fundamental reasoning process, which logically uses a series of reasoning steps, often of an "if, then" form. Beginning with a set of events, we form a chain of reasoning to come to one or more conclusions. Figure 2a models a simple case, while Figure 2b gives an example of that case. More complex structures can be easily created [15].

The ETURWG found that within this fundamental process there were at least four areas for uncertainty considerations: the uncertainty in the input data, the uncertainty associated with representation within the fusion system, the uncertainty effects of the reasoning process, and the resultant uncertainty in the outputs of the process [7, 8]. The subsections below address some of the ontological considerations associated with the first three factors. Issues associated with output uncertainty are treated in section V.

A. Uncertainty in the Input Data

All conclusions are ultimately grounded on evidence, drawn from a variety of data sources. But often evidence is "inconclusive, ambiguous, incomplete, unreliable, and dissonant." Any conclusions drawn from a body of evidence is necessarily uncertain. Schum [15] found that one must establish the credentials of any evidence used in a reasoning process. These credentials are its relevance to the question / issue at hand, its credibility, and its weight or force [16]. This suggests that one should elaborate on the fundamental reasoning process from Figure 2 with the additional items shown in Figure 3.

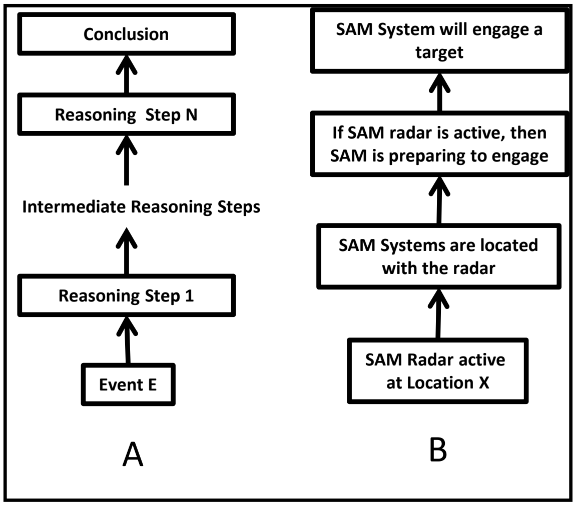

Conclusion
Reasoning Step N
Intermediate Reasoning Steps
Reasoning Step 1
Event E
A
SAM System will engage a target
If SAM radar is active, then SAM is preparing to engage
SAM Systems are located with the radar
SAM Radar active at Location X
B

Figure 2: Fundamental Reasoning Process

Data becomes evidence only when it is relevant. Relevance assesses whether the evidence at hand is germane to the question(s) being considered. Irrelevant information makes no contribution to the conclusion drawn, and potentially confuses the fusion process by introducing extra noise. Evidence can be either positively (supportive) or negatively (disconfirmatory) relevant to a particular hypothesis. Any analytic effort is obliged to seek and evaluate all relevant data.

Once data is shown to be relevant to a particular problem (i.e., it becomes evidence), Schum points out that there is an important but often overlooked distinction between an event (an object, process, juncture or participation in Sowa's ontological categories) and the evidence about that event or state. That is, Joe's statement "I saw Bob hit Bill with a club" does not mean that such event actually happened, and should be seen only as evidence about it. Credibility establishes how believable a piece of evidence is about the event it reports on. Schum identified three elements of credibility [17]; the ETURWG added self-report as a distinct element (see Table 4 for elements and definitions) [7].

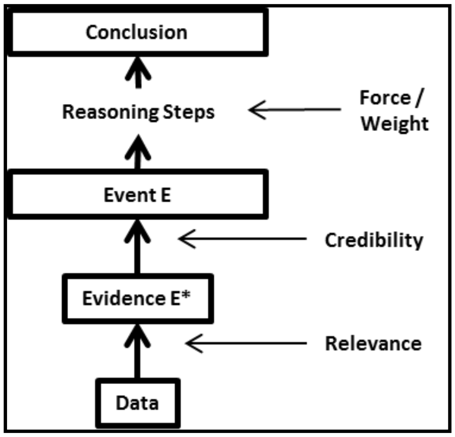


Figure 3: Evidential Factors

Table 4: Elements of Evidential Credibility

Veracity: Source is telling what it believes to be true (note that the source may be deceived)
Objectivity: Source has received the evidence on which it based its reporting. This includes consideration of system biases and false alarms
Observational Sensitivity: Source has the ability to actually observe what it reports (e.g. Observer actually has the visual acuity needed to see what was going on, or an electronic intercept was of such low quality the operator guessed part of the conversation)
Self-Report: Source provides a measure of its certainty in its report (e.g. a human source hedges her report with "it's possible that…" or a sensor reports that detection was done at a signal to noise ratio of 4)

The force (or weight) of the event establishes how important the existence of that event is to the conclusion one is trying to establish. By itself, the event "Bob hit Bill with a club" would have a significant force in establishing a conclusion that Bill was seriously injured. It would have less force in establishing that Bill was committing a violent act and needed to be stopped at Bill, and even less force in concluding that Bob was angry at Bill. Figure 3 shows that credibility can have an effect on the force of an event on the conclusion. For example, if the credibility of Joe's testimony about Bob hitting Bill with a club is low, the certainty of a conclusion that Bob's hitting was the cause of Bill's injuries would be less than if Joe testimony's credibility was high. Schum investigated a number of different ways in which considerations about data credibility could affect the overall conclusions. One of his most interesting findings is that, under certain circumstances, having credible data on the credibility of a data source can have a more significant force on the conclusion than the force of the event reported in the data [15].

B. Uncertainty in the Representation

Uncertainty varies in its forms and manifestations. Therefore, the uncertainty representation scheme used has an effect on what can or cannot be expressed. To see this, one first needs to have an understanding on the different types of uncertainty. The W3C Incubator Group exploring uncertainty reasoning issues for the World Wide Web developed an initial ontology of uncertainty concepts, shown in Figure 4 [18].

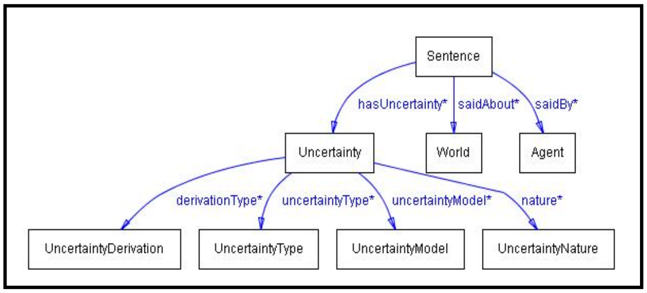


Figure 4: Uncertainty Ontology

A Sentence is a logical expression in some language that evaluates to a truth-value (formula, axiom, assertion). For our purposes, information will be presented in the form of sentences. The World is the context / situation about which the Sentence is said. The Agent represents the entity making the Sentence (human, computer etc.). Uncertainty is associated with each sentence, and has four categories. Three of those are described in Table 5, along with their significance for uncertainty propagation in a HLIF process.

Table 5: Definition of Uncertainty Categories

Uncertainty Derivation
Objective: Derived in a formal way, repeatable derivation process. Significance - level of uncertainty can be reliably estimated
Subjective: Judgment, possibly a guess. Significance - Level of uncertainty may be unpredictable
Uncertainty Nature
Aleatory: Uncertainty inherent in the world Significance - Additional data will not resolve uncertainty
Epistemic: Uncertainty in an agent due to lack of knowledge Significance - Uncertainty could be resolved by additional evidence gathering, which eliminates the lack of knowledge
Uncertainty Type
Ambiguity: Referents of terms are not clearly specified Significance - The same evidence may not distinguish between two or more possibilities
Empirical : Sentence about a world is either satisfied or not satisfied in each world, but it is not known in which worlds it is satisfied; this can be resolved by obtaining additional information (e.g., an experiment) Significance - Uncertainty can be resolved with additional information
Randomness (Type of empirical uncertainty): sentence is an instance of a class for which there is a statistical law governing whether instances are satisfied Significance - The empirical uncertainty has a predictable basis for making an estimate, using the appropriate statistical law
Vagueness: No precise correspondence between terms in the sentence and referents in the world Significance - Uncertainty due to a lack of precision
Incompleteness: information about the world is incomplete / missing Significance - Uncertainty increases because assumptions / estimates of information must be used, rather than the actual information. May not have a basis for making an estimate
Inconsistency: no world can satisfy the statement. Significance - Data is contradictory; must resolve source of contradiction (Can occur when deception is used)

The last category in the ontology is Uncertainty Model, capturing the various approaches that can be used to model uncertainty in a reasoning process. These include (but are not limited to):

- Bayesian Probability Theory
- Dempster-Shaffer Evidence Theory
- Possibility Theory
- Imprecise Probability approaches
- Random Set Theory
- Fuzzy Theory / Rough Sets
- Interval Theory
- Uncertainty Factors

A critical item in uncertainty propagation is the proper fit between the types of uncertainty in the input data and in the model(s) used in the fusion reasoning process. Failure to account for all of the uncertainty types in the input data can result in an erroneous process output. A classic survey of uncertainty models, with a discussion on applicable uncertainty types, is given in [19], with a recent review of the state-of-the-art in [20]

C. *Uncertainty in the HLIF Fusion Process*

To explore the ontological considerations of the uncertainty propagation in a HLIF fusion process, we need to have a basic fusion process model. We will concentrate on the level 2 fusion process only, and leave out significant detail on the processes at the other levels. Figure 5 shows this model. The first thing to observe is that the raw data can come in at any level, as evidenced by the incoming arrows at the right side of the figure. The model does not require that all data be signal or feature (Level 0) data, which is then aggregated into higher-level conclusions. For instance, object identification data (level 1) could come from an on-scene observer or from an image analyst reporting on an image. Communications intercepts or human reporting could provide evidence on relationships (level 2) or future intentions (level 3). Note that if a level 3 fusion process is active, its outputs could affect the level 2 process in two places. It can either be a controlling variable in the fusion process itself, or it can affect the interpretation and extraction of evidence. However, a level 3 process will have an effect only if it has separate evidence that is not being used in the level 2 fusion process (otherwise one has circular reporting).

There are four basic processes in this model. The first is the ***fusion*** process itself, which is usually some form of a model-based process. These models most often take the form of Bayesian networks [10, 21, 22], although alternative approaches have been proposed using graphical belief models [23] and general purpose graphical modeling using a variety of uncertainty techniques [14].

Another important aspect of this model that must be emphasized is that not all of the evidence that goes into the model-based process is (or is assumed to be) in an immediately usable form. Some data must have the appropriate evidence extracted from it. This is where the uncertainty considerations associated with representation within the fusion system come into play. For example, the raw level 2 data may be a series of people association data, which must be combined into a social network analysis to reveal the full extent of the relationships.

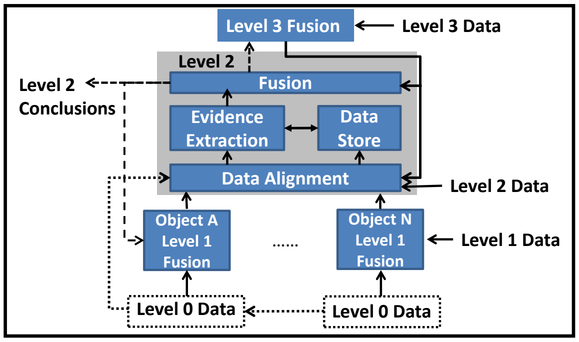


Figure 5: Level 2 Fusion Process Model

Another example may be that one is interested in whether two ships met and transferred cargo in the open ocean. Suppose that you have a track file on each ship which has long revisit rates between collections. This does not provide an obvious indication that the ships met and stopped for a while. But the track files show that both ships were on tracks that did put them at a common location at a given period, and that the average speed dropped significantly during the time a meeting could have occurred (implying that the ships may have stopped for a while). Given this data, one could conclude with some level of certainty that they did meet and stopped to transfer something. This level of certainty is driven by at least two factors: the quality of the track file data (establishing how certain one is in concluding that the tracks allowed them to meet), and how likely is it that two ships showing these track characteristics actually would have met and stopped.

A significant part of the ***evidence extraction*** process could be comparison to historical or reference data. For example, a vehicle may be moving outside of a normal shipping lane / airway or off-road. This requires a reference to a map base. For this reason, the process model includes a *data store*, for both reference information and for previous data.

The last part of the model is a ***data alignment*** process. Data may come in with different reference bases, and need to be aligned to a common baseline in order to be used in the extraction and fusion processes.

Finally, note that the level 2 process includes the possibility of a direct use of level 0 data. An area of active research is the multi-source integration of level 0 data that is not of sufficient quality, or that does not have enough quantity to allow a high quality single-source conclusion.

V. Mathematical Construct

A. *Model*

Several authors have developed mathematical constructs for use in assessing the uncertainty of a situation assessment [2, 25]. Our model is a version of the one put forth by Karlsson [26], modified using the terminology put forth by Franconi [27]. Karlsson's version focuses only on relationships, and does not explicitly include predicates and attributes. While one can model predicates and attributes using relationships, it is cleaner to separate the entity space from the attribute space. In addition, the construct formed in this paper acknowledges level 2 HLIF as explicitly including entity attributes as well as relationships between entities. Including attributes as separate from entity relationships, rather than defining relationships to include attribute states makes this clearer. Per [27], the language consists of:

E_n, the 1-ary predicates

A_k, the attributes (stated as 2-ary predicates)

R_p, n-ary predicates for all relationships

There is an interpretation function $I = \langle D, \cdot^I \rangle$ where domain D is a non-empty set = $\Omega \cup B$, Ω is the set of all entities, B is the set of all attribute values and $\Omega \cap B = \emptyset$. Then

$E_i^I \subseteq \Omega$

$A_i^I \subseteq \Omega \text{ X } B$

$R_i^I \subseteq \Omega \text{ X } \Omega \text{ X} \ldots \text{ X } \Omega = \Omega^n$

x_i are the specific instances and $x_i \in \Omega$

We can make at least three uncertainty assessments. For any specific entity tuple $(x_1,\ldots, x_n)$, we have a level of uncertainty as to whether that tuple is a member of a specific relationship. For a generic uncertainty measure u_T, the basic equation for whether a tuple is correctly associated with a defined relationship is

$$u_{Tj}((x_1,\ldots, x_n)_j \in R_j \mid E_{B,} S, I) \qquad (1)$$

where E_B is the body of evidence used in making the assignment, and S, I are any already known situation or impact states. A similar equation holds for attribute uncertainty.

We can also have uncertainty as to whether a relationship that we see in the data is the relationship of interest. Given a set of k possible relationship and a body of evidence E_B for a particular relationship $R_{current}$, we can assess the following uncertainty:

$$u_{Rk}((R_{current} = R_k \mid E_{B,} S, I) \qquad (2)$$

Again, a similar uncertainty equation holds for attribute uncertainty. Situation assessment depends on the relationships in the situation. A situation then can be defined as

$$S \overset{def}{=} (R_1, \ldots., R_{k,} A_1, \ldots. A_n) \qquad (3)$$

Finally, we have an uncertainty measure u_s. Given a set of m possible situations and a body of evidence E_B for a particular relationship $S_{current}$, we can assess the following uncertainty:

$$u_s(S_{current} = S_m \mid E_B, I) \qquad (4)$$

In addition to uncertainties in the evidence and in the reasoning process, equation (4) also allows us to account for uncertainties in the situation definition. Equation 3 implies that every situation can be precisely defined as a set of specific relationships and attributes. But what if a relationship or attribute is missing in a particular situation instance? For example, a canonical birthday celebration in the United States includes a cake with a number of lit candles on it. If there are no candles on the cake, does this mean it is not a birthday celebration?

B. *Application to Situation Assessment Taxonomy*

We can use this model to better understand the varying complexities of the different situation assessment cases given in section 3. For the simplest case, entity attribute refinement, we see that we have a very simple situation ("emitter operational in the environment"). From the existence of one object (the Tin Shield radar), we are inferring the existence of a second object (the SA-10 SAM system). This is a binary relation, based on a Sowa Juncture (x_1, x_2). With this binary relation, we are operating with a single instance of equation (1). We only have the uncertainty measure for "Tin Shield" and "SA-10" to be in juncture. For the second case, entity selection, we again have a defined situation, but now are seeking a specific object within multiple choices of objects. We are operating at the level of equation (2) – we are seeking a specific relation that ship i is the ship of interest. Based on the evidence, we will create multiple tuples for the different relationships that could lead us to the ship (using equation (1)) and then combine the results to get to equation (2).

For the third case, structure / situation selection, we invoke equation (4) as the basic equation. We are choosing between multiple choices as to what the situation is. We use equation (1) to determine if various relationships exist, and based on those findings, determine which situation model is the correct one for this body of evidence. For the fourth case, structure / situation refinement, we again use equations (1) and (4). But we also use equation (2) to determine what the exact set of relationships is. Case 4 differs from case 3 in that we are trying to determine what the relationships are that are appropriate for this situation (or structure).

For the fifth case, structure / situation creation, we have all of the uncertainties addressed above, and we add an uncertainty not immediately obvious in the generic equations. Relook equation (4). One of the stated requirements is that we are selecting among a set of defined situations. This essentially is a closed world assumption. However, in case 5 we are building the situation, rather than determining which situation among a choice of situations is the applicable one. We still have a number of models to choose from, but they are more fragmentary than in previous cases. The previous cases represent more of a "pieces of the puzzle" approach, where one is assembling the puzzle according to one or more available pictures to help guide you. Case 5 represents the case where we one is assembling the puzzle without a picture or set of pictures to guide one. Rather, you are assembling the puzzle guided by basic puzzle rules about matching shapes and picture colors. So, in case 5, we are also determining what the applicable S_ks are.

VI. Discussion

Up to this point we have been able to attest the existence of a number of uncertainty propagation considerations when analyzing a level 2 HLIF. Most of these are not necessarily obvious at a first glance, which suggests the importance of a framework that supports the analytical process. The framework proposed in this paper is meant for supporting the analysis of processes occurring at JDL fusion level 2, and an important aspect of it is the ability to correlate such processes with the uncertainty considerations raised so far. Figure 6 summarizes these considerations as they relate to the heart of the basic process model shown in Figure 5.

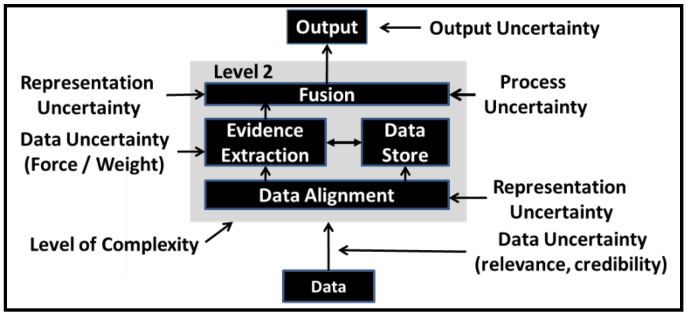

Output
Output Uncertainty
Level 2
Representation Uncertainty
Fusion
Process Uncertainty
Data Uncertainty (Force / Weight)
Evidence Extraction
Data Store
Data Alignment
Representation Uncertainty
Level of Complexity
Data Uncertainty (relevance, credibility)
Data

Figure 6: Level 2 HLIF Uncertainty Considerations

The taxonomy of level 2 HLIF types discussed in section 2 defines the complexity of the uncertainty considerations that must be accounted for. Five different types are identified, ranging from simple entity attribute refinement using situation status data to the development of a complete situation assessment assembled from applicable situational fragment data. The uncertainty in the input data / evidence must be assessed for relevance, credibility, and force / weight, per the ontology of evidence presented in Laskey et al. [17]. The representation uncertainties that drive the modeling methodologies can be classified per the uncertainty ontology developed by the W3C Incubator Group for Uncertainty Reasoning [18]. A variety of different models can be used to properly capture the aspects of uncertainty in the data [19, 20]. Finally, the output uncertainty strongly depends on the a priori identification of possible situation choices, or upon having a fusion process that allows for an effective open world assumption. These uncertainty considerations are the beginning of understanding how to evaluate the effectiveness of various uncertainty management methods in high-level fusion.

REFERENCES

[1] D. L. Hall, J. Llinas, “Multi-Sensor Data Fusion” in *Handbook of Multisensor Data Fusion: Theory and Practice (2nd ed)*, CRC Press, pp 1-14, 2009

[2] D. A. Lambert, “A Blueprint for Higher Level Fusion Systems”, in Information Fusion, pp 6-24, Elsevier, Vol 10, 2009

[3] M. M. Kokar, C. J. Matheus, K. Baclawski, J. A. Letkowski, M. Hinman, J. Salerno, “Use Cases for Ontologies in Information Fusion” Proceedings of the 7th International Conference on Information Fusion (2004), retrieved from http://vistology.com/papers/Fusion04-UseCases.pdf on 1 Jun 2012.

[4] E. G Little, G. L Rogova, “Designing Ontologies for Higher Level Fusion”, Information Fusion, pp 70-82, Elsevier, Vol 10, 2009

[5] P. C. G. Costa (2005) *Bayesian Semantics for the Semantic Web*. Doctoral Thesis, School of Information Technology and Engineering, George Mason University. Fairfax, VA, USA, 2005.

[6] R. N Carvalho (2011) *Probabilistic Ontology: Representation and Modeling Methodology*. Doctoral Thesis, School of Information Technology and Engineering, George Mason University. Fairfax, VA, USA, 2011.

[7] Evaluation of Technologies for Uncertainty Reasoning Working Group (ETURWG) website, http://eturwg.c4i.gmu.edu/?q=aboutUs , retrieved May 19, 2012.

[8] P. C. G. Costa, K. B. Laskey, E. Blasch, A. Jousselme, “Towards Unbiased Evaluation of Uncertainty Reasoning: The URREF Ontology”, Proceedings of the 15th International Conference on Information Fusion (To Be Published)

[9] A. N. Steinberg, C. L. Bowman, “Revisions to the JDL Data Fusion Model,” *Handbook of Multisensor Data Fusion: Theory and Practice (2nd ed)*, CRC Press, pp 45 - 68, 2009

[10] S. Das, *High-Level Data Fusion*, Boston MA (USA): Artech House, 2008

[11] J. Sowa, *Knowledge Representation: Logical, Philosophical and Computational Foundations*, Grove CA (USA): Brooks/Cole, Pacific, 2000

[12] http://www.raytheon.com/capabilities/products/apg63_v3/

[13] http://www.p3orion.nl/variants.html

[14] Air Power Australia website, http://www.ausairpower.net/APA-Acquisition-GCI.html#mozTocId55304, as retrieved on June 2, 2012

[15] D. A. Schum, *The Evidential Foundations of Probabilistic Reasoning*, New York: John Wiley and Sons, Inc., 1994.

[16] D. Schum, “Thoughts About a Science of Evidence, ” University College London Studies of Evidence Science, retrieved from 128.40.111.250/evidence/content/Science.doc on June 2, 2012

[17] K.B Laskey, D. A. Schum, P. C. G Costa, T. Janssen, “Ontology of Evidence”, Proceedings of the Third International Ontology for the Intelligence Community Conference (OIC 2008), December 3-4, 2008

[18] K.J. Laskey, K. B. Laskey, P. C. G. Costa, M. M. Kokar, T. Martin, T. Lukasiewicz, Uncertainty Reasoning for the World Wide Web, W3C Incubator Group Report 31 March 2008. Retrieved from http://www.w3.org/2005/Incubator/urw3/XGR-urw3-20080331/

[19] P. Walley, “Measures of uncertainty in expert systems”, Artificial Intelligence, 83(1), May 1996, pp. 1-58

[20] B. Khaleghi, A. Khamis, F. O. Karray, “Multi-sensor Data Fusion: A Review of the State-of-the-Art”, Information Fusion (2011), doi: 10.1016/j.inffus.2011.08.001

[21] K. B. Laskey, P. C. G. Costa, T. Janssen, “Probabilistic Ontologies for Multi-INT Fusion” in Proceedings of the 2010 conference on Ontologies and Semantic Technologies for Intelligence, 2010

[22] A. N. Steinberg, “Foundations of Situation and Threat Assessment” in *Handbook of Multisensor Data Fusion: Theory and Practice (2nd ed)*, CRC Press, pp 437 -502, 2009

[23] R. G. Almond, *Graphical Belief Modeling*, New York NY (USA): Chapman and Hall, 1995

[24] P. P. Shenoy, “Valuation-Based Systems for Bayesian Decision Analysis,” *Operations Research*, pp 463 – 484, Vol 40, No 3, May-June 1992 ,

[25] P. Svensson, “On Reliability and Trustworthiness of High-Level Fusion Decision Support Systems: Basic Concepts and Possible Formal Methodologies”, 9th International Conference on Information Fusion, Florence (Italy), 10-13 July 2006, retrieved online from http://www.isif.org/fusion/proceedings/fusion06CD/Papers/51.pdf on May 6, 2012

[26] A. Karlsson, Dependable and Generic High-Level Algorithms for Information Fusion – Methods and Algorithms for Uncertainty Management, Technical Report HS-IKI-TR-07-003, University of Skovde, retrieved 15 Sep 2011 from his.diva-portal.org/smash/get/diva2:2404/FULLTEXT01.

[27] E. Franconi, Description Logic Tutorial Course, downloaded from http://www.inf.unibz.it/~franconi/dl/course/ on 1 May 2012

Using Ontologies in a Cognitive-Grounded System: Automatic Action Recognition in Video Surveillance

Alessandro Oltramari
Department of Psychology
Carnegie Mellon University
Pittsburgh, Pennsylvania 15217
Email: aoltrama@andrew.cmu.edu

Christian Lebiere
Department of Psychology
Carnegie Mellon University
Pittsburgh, Pennsylvania 15217
Email: cl@cmu.edu

***Abstract*—This article presents an integrated cognitive system for automatic video surveillance: in particular, we focus on the task of classifying the actions occurring in a scene. For this purpose, we developed a semantic infrastructure on top of a hybrid computational ontology of actions. The article outlines the core features of this infrastructure, illustrating how the processing mechanisms of the cognitive system benefit from knowledge capabilities in fulfilling the recognition goal. Ultimately, the paper shows that ontologies can enhance a cognitive architecture's functionalities, allowing for high-level performance in complex task execution.**

I. Introduction

The automatic detection of anomalous and threatening behaviour has recently emerged as a new area of interest in video surveillance: the aim of this technology is to disambiguate the context of a scene, discriminate between different types of human actions, eventually predicting their outcomes. In order to achieve this level of complexity, state-of-the-art computer vision algorithms [1] need to be complemented with higher-level tools of analysis involving, in particular, knowledge representation and reasoning (often under conditions of uncertainty). The goal is to approximate human visual intelligence in making effective and consistent detections: humans evolved by learning to adapt and properly react to environmental stimuli, becoming extremely skilled in filtering and generalizing over perceptual data, taking decisions and acting on the basis of acquired information and background knowledge.

In this paper we first discuss the core features of human 'visual intelligence' and then describe how we can simulate and approximate this comprehensive faculty by means of an integrated framework that augments ACT-R cognitive architecture (see figure 1) with background knowledge expressed by suitable ontological resources (see section III-B2). ACT-R is a modular framework whose components include perceptual, motor and memory modules, synchronized by a procedural module through limited capacity buffers (refer to [2] for more details). ACT-R has accounted for a broad range of cognitive activities at a high level of fidelity, reproducing aspects of human data such as learning, errors, latencies, eye movements and patterns of brain activity. Although it is not our purpose in this paper to present the details of the architecture, two specific mechanisms need to be mentioned here to sketch how the system works: i) *partial matching* - the probability that two different knowledge units (or *declarative chunks*) can be associated on the basis of an adequate measure of similarity (this is what happens when we consider, for instance, that a bag is more likely to resemble to a basket than to a wheel); ii) *spreading of activation* - when the same knowledge unit is part of multiple contexts, it contributes to distributionally activate all of them (like a chemical catalyst may participate in multiple chemical transformations). Section 7 will show in more details how these two mechanisms are exploited by the cognitive system to disambiguate action signals: henceforth, we will refer to this system as the *Cognitive Engine*. As much as humans understand their surroundings coupling perception with knowledge, the *Cognitive Engine* can mimic this capability by leveraging scene-parsing and disambiguation with suitable ontology patterns and models of actions, aiming at identifying relevant actions and spotting the most anomalous ones.

In the next sections we present the different aspects of the *Cognitive Engine*, discussing the general framework alongside specific examples.

II. The Conceptual Features of Visual Intelligence

The territory of 'visual intelligence' needs to be explored with an interdisciplinary *eye*, encompassing cognitive psychology, linguistics and semantics: only under these conditions can we aim at unfolding the variety of operations that visual intelligence is responsible for, the main characteristics of the emergining representations and, most importantly in the present context, at reproducing them in an artificial agent.

As claimed in [3],"events are understood as action-object couplets" (p. 456) and "segmenting [events as couplets] reduces the amount of information into manageable chunks" (p. 457), where the segment boundaries coincide with achievements and accomplishments of goals (p.460). Segmentation is a key-feature when the task of disambiguating complex scenarios is

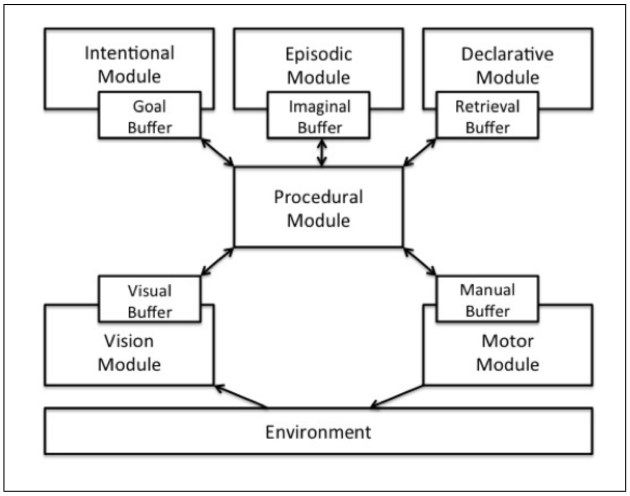


Fig. 1. ACT-R modular structure elaborates information from the environment at different levels.

considered: recognition doesn't correspond to the process of making an inventory of all the actions occurring in a scene: a selection process is performed by means of suitable 'cognitive schemas' (or *gestalts*, e.g. up/down, figure/ground, force, etc.), which carve visual presentations according to principles of mental organization and optimize the perceptual effort" [4]. Besides cognitive schemas, conceptual primitives have also been studied: in particular, [5] applied Hayes' naïve physics theory [6] to build an event logic. Within the adopted common sense definitions, we can mention i) *substantiality* (objects generally cannot pass through one another); ii) *continuity* (objects that diachronically appear in two locations must have moved along the connecting path); iii) *ground plane* (ground acts as universal support for objects).

As far as action-object pairs are central to characterize the 'ontology of events', verb-noun 'frames' are also relevant at the linguistic level[1]; in particular, identifying roles played by objects in a scene is necessary to disambiguate action verbs and highlight the underlying goals. In this respect, studies of event categorization revealed that events are always *packaged*, that is distinctly equipped with suitable semantic roles [8]: for example, the events which are exemplified by motion verbs like walk, run, fly, jump, crawl, etc. are generally accompanied with information about source, path, direction and destination/goal, as in the proposition "John ran out of the house (*source*), walking south (*direction*) along the river (*path*), to reach Emily's house (*destination/goal*)"; conversely, verbs of possession such as have, hold, carry, get, etc. require different kind of semantic information, as in the proposition "John (*owner*) carries Emily's bag (*possession*)". Note that it is not always the case that all possible semantic roles are filled by linguistic phrases: in particular, *path* and *direction* are not necessarily specified when motion is considered, while *source* and *destination/goal* are (we do not focus here on agent and patient which are the core semantic roles).

As this overview suggests, there is an intimate connection between linguistics, cognition and ontology both at the level of scene parsing (mechanism-level) and representation (content-level). In particular, in order to build a visual intelligent system for action recognition, three basic functionalities are required:

- **Ontology pattern matching** - comparing events on the basis of the similarity between their respective pattern components: e.g., *a person's burying an object* and *a person's digging a hole* are similar because they both include some basic body movements as well as the act of removing the soil;
- **Conceptual packaging** - eliciting the conceptual structure of actions in a scene through the identification of the roles played by the detected objects and trajectories: e.g. if you watch *McCutchen hitting an homerun*, the Pittsburgh Pirates' player number 22 is the 'agent', the ball is the patient, the baseball bat is the 'instrument', toward the tribune is the 'direction', etc.).
- **Causal selectivity**: attentional mechanisms drive the visual system in picking the causal aspects of a scene, i.e. selecting the most distinctive actions and discarding collateral or accidental events (e.g., in the above mentioned *homerun* scenario, focusing on the movements of the first baseman is likely to be superfluous).

In the next section we describe how the *Cognitive Engine* realizes the first two functionalites by means of combining the architectural features of ACT-R with ontological knowledge, while **Causal selectivity** will be addressed in future work.

III. Building the Cognitive Engine

A. The Context

The *Cognitive Engine* represents the core module of the Extended Activity Reasoning system (EAR) in the CMU-Minds Eye architecture (see figure 2). Mind's Eye is the name of the DARPA program[2] for building AI systems that can filter surveillance footage to support human (remote) operators, and automatically alert them whenever something suspicious is recognized (such as someone leaving a package in a parking lot and running away – see also [9]). In this framework, visual intelligent systems play the role of filtering computer vision data, suitably coupling relevant signals with background knowledge and – when feasible – searching for a 'script' that ties together all the most salient actions in a scene. This comprehensive capability requires intensive information processing at interconnected levels: basic optical features (low-level), object detection (mid-level) and event classification (high-level). EAR has been conceived to deal with the last one: in particular the *Cognitive Engine* receives outputs from the Immediate Activity Recognition module (IAR), which collects the results of different pre-processing algorithms and adopts learning–based methods to output action probability distributions [10].

[1] We refer here to the very broad notion of 'frame' introduced by Minsky: "frames are data-structure for representing a stereotyped situation, like being in a certain kind of living room, or going to a child's birthday party" [7].

[2] http://www.darpa.mil/Our_Work/I2O/Programs/Minds_Eye.aspx

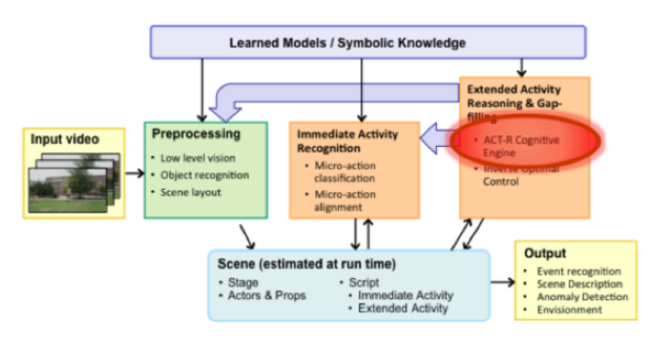


Fig. 2. CMU Mind's Eye architecture

Specific parsing functions are included in EAR to convert the IAR output into sequences of quasi-propositional descriptions of atomic events to be fed to the *Cognitive Engine*. For example, the sample video strip in figure 3 can be converted into *(a)*:

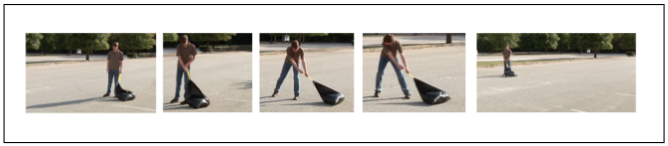


Fig. 3. Significative moments of a composite action

(a) Person1 Holds Bag2 + Person1 Bends Over + Person1 Drags Bag2 + Person1 Stops.

These sequences reflect the most likely atomic events (so called 'micro-actions', 'micro-states' and'micro-poses') occurring in the environment, detected and thresholded by machine vision algorithms. The addition symbol exemplifies temporal succession while numbers stand for entity unique identifiers. For the sake of readability, we omit here the temporal information about start and end frames of the single atomic-events, as well as spatial coordinates of the positions of objects. Leveraging the semantic properties of sequences like *(a)*, the *Cognitive Engine* aims at generalizing over action components and distill the most likely 'unifying story': for instance, figure 3 depicts a person hauling an object to the top left side of the scene. Ontology patterns [11] of action play a key-role in the process of sequence disambiguation: in this regard, III-B reviews some of the core patterns we adopted in the recognition mechanisms of the *Cognitive Engine* and outlines the basic classes and properties of the ontology of actions used for high-level reasoning. The benefits of using ontologies for event recognition in the context of the Mind's Eye program have been also discussed in [12], although our two approaches differ both in the theoretical underpinnings (as the next sections will show, we propose a hybridization of linguistic and ontological distinctions rather than embracing ontological realism) and in the general system design (in [12] the authors outline a framework in which ontological knowledge is directly plugged into visual algorithms, while in our proposal ACT-R is exploited as an intermediate module to bridge the vision and the knowledge levels, stressing the role of cognitive mechanisms in action understanding).

B. The Knowledge Infrastructure

1) Ontology patterns of actions: In recent years,'Ontology Design Patterns' (or just 'ontology patterns') have become an important resource in the areas of Conceptual Modeling and Ontology Engineering: the rationale is to identify some minimal conceptual structures to be used as the *building blocks* for designing ontologies [13]. Ontology patterns are small models of entities and their basic properties: the notion originates in [14], where the author argues that a good (architectural) design can be achieved by means of a set of rules that are packaged in the form of patterns, such as 'windows place', or 'entrance room'. Design patterns are then assumed as archetypal solutions to design problems in a certain context. Ontology patterns are built and formalized on the basis of a preliminary requirement analysis, which can be driven either by applications tasks or by specific problems in the domain of interest. In our context, ontology patterns enable the classification of actions by means of pinpointing the basic semantic roles and constituent atomic events of relevant actions. In these regards, table I shows the composition of the core ontology patterns used in the *Cognitive Engine*: e.g. an instance of the action-type 'pick-up' depends on the occurrence of at least four basic components (C1-C4), namely 'bend-over', 'lower-arm', 'stand-up' (necessary body-movements) and 'holding' (referring to the interaction between a person and an object); moreover, those action-verbs require specific conceptual roles to be exemplified, respectively, *protagonist* for the first and the third component, *agent* for the second and the fourth (which includes also 'patient' as object-role). But what did inspire our modeling choices? How could we identify those roles and atomic events? Which rules/principles allowed us to assemble them in that very fashion? In order to answer to these questions, in the next section we introduce HOMINE, the Hybrid Ontology for the Mind's Eye project.

2) Ontology of actions: Ontologies play the role of 'semantic specifications of declarative knowledge' in the framework of cognitive architectures [15]. As [16], [17], [18], [19] demonstrate, most research efforts have focused on designing methods for mapping large knowledge bases to the ACT-R declarative module. Here we commit on taking a different approach: instead of tying to a single monolithic large knowledge base, we built a hybrid resource that combines different semantic modules, allowing for high scalability and interoperability. Our proposal consists in suitably linking distinctive lexical databases, i.e. WordNet [20] and FrameNet [21] with a computational ontology of actions, plugging the obtained semantic resource in the dynamic mechanisms of the ACT-

TABLE I
ONTOLOGY PATTERNS OF ACTIONS FOR THE COGNITIVE ENGINE

Action	Role1	Role2	Role3	Role4	Object	C1	C2	C3	C4
Arrive	self-mover	theme				walk	stop		
Leave	self-mover	theme				walk	exit		
Give	agent	carrier	agent		patient	holding	transport	drop	
Take	carrier	agent	agent		patient	transport	drop	holding	
Exchange	agent	agent	agent		patient	give	take	swap	
Carry	agent	carrier	agent		patient	holding	transport	pull	
Pick-up	protagonist	agent	protagonist	agent	patient	bend-over	lower-arm	stand-up	holding
Put-down	agent	protagonist	agent	figure1	patient	holding	bend-over	lower-arm	on
Bury	protagonist	agent	protagonist	agent	patient	bend-over	lower-arm	fill-with-tool	stand-up
Dig	protagonist	agent	agent	protagonist	patient	bend-over	lower-arm	dig-with-tool	stand-up
Haul	protagonist	agent	agent	agent	patient	bend-over	extend-arm	holding	drag

R cognitive architecture (see IV). Accordingly, HOMINE is built on the top-level of DOLCE-SPRAY [22], a simplified version of DOLCE [23]: we used DOLCE-SPRAY as a general model for aligning WordNet (WN) and FrameNet (FN) – following the line of research of [24]: figure 4 shows some selected nodes of DOLCE backbone taxonomy. The root of the hierarchy of DOLCE-SPRAY is ENTITY, which is defined as anything which is identifiable by humans as an object of experience or thought. The first distinction is among CONCRETE-ENTITY, i.e. objects located in definite spatial regions, and ABSTRACT-ENTITY, whose instances don't have spatial properties. In the line of [25], CONCRETE-ENTITY is further split in CONTINUANT and OCCURRENT, namely entities without inherent temporal parts (e.g. artifacts, animals, substances) and entities with inherent temporal parts (e.g. events, actions, states) respectively. The basic ontological distinctions are maintained: DOLCE's ENDURANT and PERDURANT match DOLCE-SPRAY's CONTINUANT and OCCURRENT. The main difference of DOLCE-SPRAY's top level with respect to DOLCE, is the merging of DOLCE's ABSTRACT and NON-PHYSICAL-ENDURANT categories into the DOLCE-SPRAY's category of ABSTRACT-ENTITY. Among abstract entities, DOLCE-SPRAY's top level distinguishes CHARACTERIZATION, defined as mapping of *n*-uples of individuals to truth values. Individuals belonging to CHARACTERIZATION can be regarded to as 'reified concepts', and the irreflexive, antisymmetric relation CHARACTERIZE associates them with the objects they denote. Whether CHARATERIZATION is formally a metaclass, and whether CHARACTERIZE bears the meaning of set membership is left opaque in this ontology.

HOMINE's linguistic-semantic layer is based on a partition of WN related to verbs of action, such as 'haul', 'pick-up', 'carry', 'arrive', 'bury' etc. WN is a semantic network whose nodes and arcs are, respectively, synsets ("sets of synonym terms") and semantic relations. Over the years, there has been an incremental growth of the lexicon (the latest version, WordNet 3.0, contains about 120K synsets), and substantial enhancements aimed at facilitating computational tractability. In order to find the targeted group of relevant synsets, we basically started from two pertinent top nodes[3], move #1 and move#2[4]. As one can easily notice, the former synset denotes a change of position accomplished by an agent or by an object (with a sufficient level of autonomy), while the latter is about causing someone or something to move (both literally and figuratively). After extracting the sub–hierarchy of synsets related to these generic verbs of action, we introduced a top-most category 'movement-generic', abstracting from the two senses of 'move' (refer to figure 5 for the resulting taxonomy of actions).

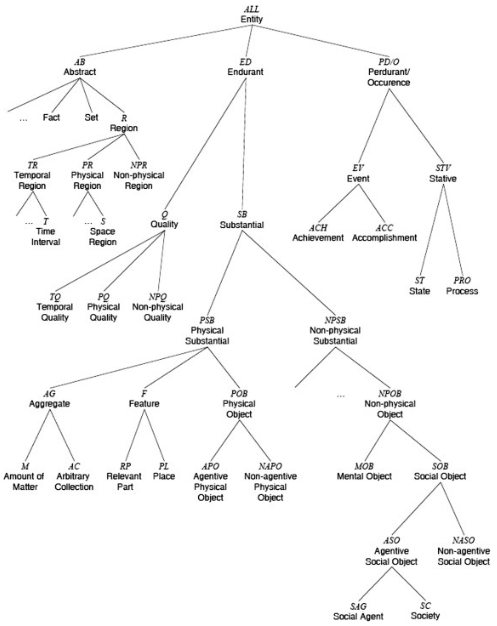


Fig. 4. An excerpt of DOLCE-SPRAY top level

FrameNet (FN) is the additional conceptual layer of HOMINE. Besides wordnet-like databases, a computational lexicon can be designed from a different perspective, for example focusing on frames, to be conceived as orthogonal

[3] AKA Unique Beginners (Fellbaum 1998).

[4] 01835496 move#1, travel#1, go#1, locomote#1 (change location; move, travel, or proceed) "How fast does your new car go?"; "The soldiers moved towards the city in an attempt to take it before night fell". 01850315 move#2, displace#4 (cause to move or shift into a new position or place, both in a concrete and in an abstract sense) "Move those boxes into the corner, please"; "The director moved more responsibilities onto his new assistant".

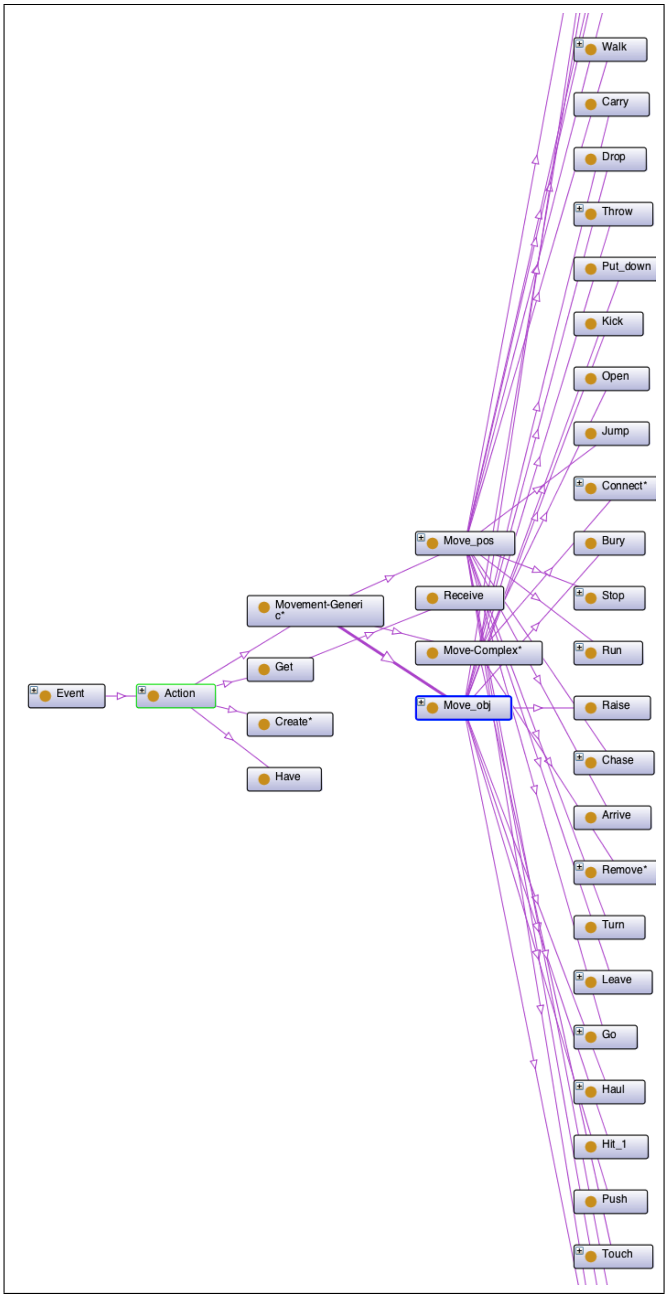


Fig. 5. An excerpt of HOMINE backbone taxonomy

to domains. Inspired by frame semantics [26], FN aims at documenting "the range of semantic and syntactic combinatory possibilities (valences) of each word in each of its senses" through corpus-based annotation. Different frames are evoked by the same word depending on different contexts of use: the notion of 'evocation' helps in capturing the multi-dimensional character of knowledge structures underlying verbal forms. For instance, if you consider the *bringing* frame, namely an abstraction of a state of affairs where sentient agents (e.g., persons) or generic carriers (e.g. ships) bring something somewhere along a given path, you will find several 'lexical units' (LUs) evoking different roles (or frame elements - FEs): i.e., the noun 'truck' instantiates the 'carrier' role. In principle, the same Lexical Unit (LU) may evoke distinct frames, thus dealing with different roles: 'truck', for example, can be also associated to the vehicle frame ('the vehicles that human beings use for the purpose of transportation'). FN contains about 12K LUs for 1K frames annotated in 150000 sentences. WN and FN are based on distinct models, but one can benefit from the other in terms of coverage and type of information conveyed. Accordingly, we have analyzed the evocation-links between the action verbs we have extracted from WN and the related FN frames: those links can be generated through 'FN Data search', an on–line navigation interface used to access and query FN[5]. Using a specific algorithm [27], WordNet synsets can be associated with FrameNet frames, ranking the results by assigning weights to the discovered connections [28]. The core mechanism can be resumed by the following procedure: first of all the user has to choose a term and look for the correspondent sense in WordNet; once the correct synset is selected, the tool searches for the corresponding lexical units (LUs) and frames of FrameNet. Afterwards, all candidate frames are weighted according to three important factors: the similarity between the target word (the LU having some correspondence to the term typed at the beginning) and the wordnet relative (which can be the term itself - if any - and/or its synonyms, hypernyms and antonyms); a variable boost factor that rewards words that correspond to LU as opposed to those that match only the frame name; the spreading factor, namely the number of frames evoked by that word:

$$\boxed{\frac{similarity(wordnet\_relative, target\_word) * BoostFactor}{spreading\_factor(wordnet\_relative)}}$$

If DOLCE-SPRAY provides the axiomatic basis for the formal characterization of HOMINE[6], and WN and FN computational lexicons populate the ontology with linguistic knowledge, SCONE is the selected framework of implementation[7]. SCONE is an open–source knowledge-base system intended for use as a component in many different software applications: it provides a LISP-based framework to represent and reason over symbolic common–sense knowledge. Unlike most diffuse KB systems, SCONE is not based on OWL (Ontology Web Language[8]) or Description Logics in general [30]: its inference engine adopts marker–passing algorithms [31] (originally designed for massive parallel computing) to perform fast queries at the price of losing logical completeness and decidability. In particular, SCONE represents knowledge as a *semantic network* whose nodes are locally weighted (*marked*) and associated to arcs (*wires*[9]) in order to optimize basic reasoning tasks (e.g. class membership, transitivity, inheritance

[5] https://framenet.icsi.berkeley.edu/fndrupal/index.php?q=luIndex

[6] For instance, DOLCE adapts Allen's temporal axioms [29], which are considered as state of the art in temporal representation and reasoning.

[7] http://www.cs.cmu.edu/~sef/scone/

[8] http://www.w3.org/TR/owl-features/

[9] In general, a *wire* can be conceived as a binary relation whose domain and range are referred to, respectively, as A-node and B-node.

of properties, etc.). The philosophy that inspired SCONE is straightforward: from vision to speech, humans exploit the brain's massive parallelism to fulfill all recognition tasks; if we want to build an AI system which is able to deal with the large amount of knowledge required in common-sense reasoning, we need to rely on a mechanism which is fast and effective enough to simulate parallel search. Accordingly, SCONE implementation of marker–passing algorithms aims at simulating a pseudo-parallel search by assigning specific marker bits to each knowledge unit. For example, if we want to query a KB to get all the parts of cars, SCONE would assign a marker M1 to the A-node CAR and search for all the statements in the knowledge base where M1 is the A-wire (domain) of the relation PART-OF , returning all the classes in the range of the relation (also called 'B-nodes'). SCONE would finally assign the marker bit M2 to all B-nodes, also retrieving all the inherited subclasses[10]. The modularization and implementation of HOMINE with SCONE allows for an effective formal representation and inferencing of core ontological properties of events, such as: i) participation of actors and objects in actions; ii) temporal features based on the notions of 'instant' and 'interval'; iii) common-sense spatial information.

The *Cognitive Engine* is the result of augmenting ACT-R with HOMINE: in general we refer to ACT-R including the SCONE extra-module as ACT-RK, meaning 'ACT-R with improved Knowledge capabilities' (the reader can easily notice the evolution from the original ACT-R architecture – figure 1 – to the knowledge-enabled one – figure 6). We engineered a SCONE-MODULE as a bridging component between the cognitive architecture and the knowledge resource: this integration allows for dynamic queries to be automatically submitted to HOMINE by ACT-RK whenever the visual information is incomplete, corrupted or when reasoning with common-sense knowlege is needed to generalize over actor and actions in a scene. In this way, the *Cognitive Engine* is able to overcome situations with missing input: ACT-R mechanisms of partial matching and spreading activation [2] can fill the gap(s) left by the missing atomic events and retrieve the best–matching ontology pattern. In the last section of the paper we describe how *Cognitive Engine* performs action-recognition task for the example orginally sketched in figure 3.

IV. Using the *Cognitive Engine* for action recognition: an example

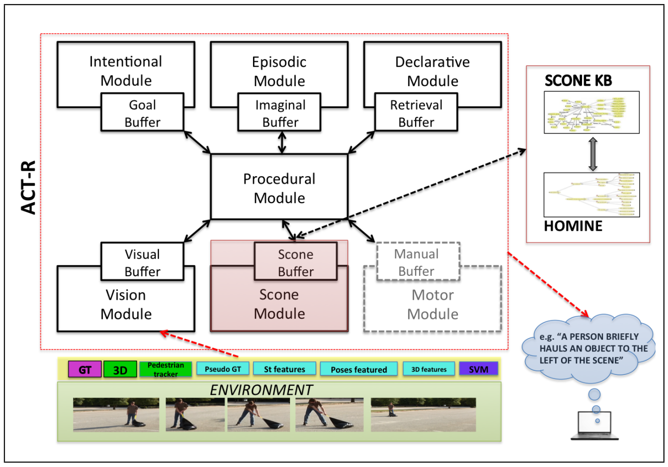


Fig. 6. The *Cognitive Engine*

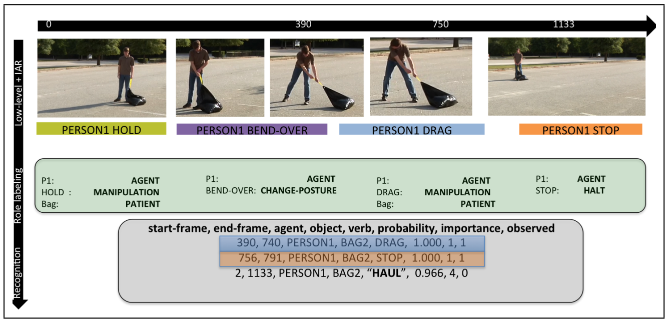
In the context of the Mind's Eye program, a visual intelligent systems is considered to be successful if it is able to process a video-dataset of actions[11] and output the probability distribution (per video) of a pre-defined list of verbs, including 'walk', 'run', 'carry', 'pick-up', 'haul', 'follow', 'chase', etc[12]. Performance is measured in terms of consistency with human responses to stimuli (*Ground-Truth*): subjects have to acknowledge the presence/absence of every verb in each video. In order to meet these requirements, we devised the *Cognitive Engine* to work in a human-like fashion (see section II), trying to disambiguate the scene in terms of the most reliable conceptual structures. Because of space limitations, we can't provide here the details of a large-scale evaluation: nevertheless, we can discuss the example depicted earlier in the paper (figure 3) in light of the core mechanisms of the *Cognitive Engine*. Considering figure 7, the *Cognitive Engine* parses the atomic events extracted by IAR, namely 'hold' (micro-state) and 'bend-over', 'drag', 'stop' (micro-actions), associating frames and roles to visual input from the videos. This specific information is retrieved from the FrameNet module of HOMINE: frames and frame roles are assembled in suitable knowledge units and encoded in the declarative memory of ACT-RK. As with human annotators performing semantic role labeling [32], the *Cognitive Engine* associates verbs denoting atomic events to corresponding frames. When related mechanisms are activated, the *Cognitive Engine* retrieves the roles played by the entities in the scene, for each atomic event: for example, 'hold' evokes the *manipulation* frame, whose core role *agent* can be be associated to 'person1' (as showed in light-green box of the figure). In order to prompt a choice within the available ontology patterns of action (see table I), sub-symbolic computations for *spreading activation* are executed [2]. Spreading of activation from the contents of frames and roles triggers the evocation of related ontology patterns. As mentioned in the introduction, *partial matching* based on similarity measures and *spreading of activation* based on compositionality are the main mechanisms used by *Cognitive Engine*: in particular, we constrained semantic similarity within verbs to the 'gloss-vector' measure computed over WordNet synsets [33]. Base-level activations of verbs actions have been derived by frequency analysis of the American National Corpus: in particular, this choice reflects the fact that the more frequent is a verb, the more is likely to be activated

[10] Far from willing to deepen a topic that is out of scope to treat in this manuscript, we refer the reader to [31] for details concerning marker–passing algorithms.

[11] http://www.visint.org/datasets.html.

[12] This list has been provided in advance by DARPA.

by a recognition system. Additionally, strengths of associations are set (or learned) by the architecture to reflect the number of patterns to which each atomic event is associated, the so-called 'fan effect' controlling information retrieval in many real-world domains [34].



```
start-frame, end-frame, agent, object, verb, probability, importance, observed
390, 740, PERSON1, BAG2, DRAG,  1.000, 1, 1
756, 791, PERSON1, BAG2, STOP,  1.000, 1, 1
2, 1133, PERSON1, BAG2, "HAUL",  0.966, 4, 0
```

Fig. 7. A Diagram of the Recognition Task performed by the *Cognitive Engine*. The horizontal black arrow represents the sequence time framing while the vertical one represents the interconnected levels of information processing. The light-green box displays the results of semantic disambiguation of the scene elements, while the gray box contains the schema of the output, where importance reflects the number of components ina detected pattern (1-4) and *observed* is a boolean parameter whose value is 1 when a verb matches an IAR detection and 0 when the verbs is an actual result of EAR processing.

The core sub-symbolic computations performed by the *Cognitive Engine* through ACT-RK can be expressed by the equation in figure 8:

$$\boxed{A_i = \ln \sum_j t_j^{-d} + \sum_k W_k S_{ki} + \sum_l MP_l Sim_{li} + N(0,\sigma)}$$

Fig. 8. Equation for Bayesian Activation Pattern Matching

- **1st term**: the more recently and frequently a chunk *i* has been retrieved, the higher its activation and the chances of being retrieved. In our context *i* can be conceived as a pattern of action (e.g., the pattern of HAUL), where t_j is the time elapsed since the j^{th} reference to chunk *i* and *d* represents the memory decay rate.
- **2nd term**: the contextual activation of a chunk *i* is set by the attentional weight S_{ki} given the element *k*, the element *i* and the strength of association between an element *k* and the *i*. In our context, *k* can be interpreted as the value BEND-OVER of the pattern HAUL in figure 7.
- **3rd term**: under partial matching, ACT-RK can retrieve the chunk *l* that matches the retrieval constraints *i* to the greatest degree, computing the similarity Sim_{li} between *l* and *i* and the mismatch score MP (a negative score that is assigned to discriminate the 'distance' between two terms). In our context, for example, the value PULL could have been retrieved, instead of DRAG. This mechanism is particularly useful when verbs are continuosly changing - as in the case of a complex visual input stream.
- **4th term**: randomness in the retrieval process by adding Gaussian noise.

Last but not least, the *Cognitive Engine* can output the results of extra-reasoning functions by means of suitable queries submitted to HOMINE via the SCONE-MODULE. In the example in figure 7, object classifiers and tracking algorithms could not detect that 'person1' is dragging 'bag2' by pulling a rope: this failure in the visual algorithms is motivated by the fact that the rope is a very tiny and morphologically unstable artifact, hence difficult to be spotted by state-of-the-art machine vision. Nevertheless, HOMINE contains an axiom stating that:
"For every x,y,e,z such that P(x) is a person, GB(y) is a Bag and DRAG(e,x,y,T) is an event e of type DRAG (whose participants are x and y) occurring in the closed interval of time T, there is at least a z which is a proper part of y and that participates to e"[13].
Moreover, suppose that in a continuation of the video, the same person drops the bag, gets in a car and leaves the scene. The visual algorithms would have serious difficulties in tracking the person while driving the car, since the person would become partially occluded, assume an irregular shape and would be no more properly lightened. Again, the *Cognitive Engine* could overcome these problems in the visual system by using SCONE to call HOMINE and automatically perform the following schematized inferences:

- Cars move;
- Every car needs exactly one driver to move[14];
- Drivers are persons;
- A driver is located inside a car;
- If a car moves then the person driving the car also moves in the same direction.

Thanks to the inferential mechanisms embedded in its knowedge infrastructure, the *Cognitive Engine* is not bound to visual input as an exclusive source of information: in a human-like fashion, the *Cognitive Engine* has the capability of coupling visual signals with background knowledge, performing high-level reasoning and disambiguating the original input perceived from the environment. In this respect, the *Cognitive Engine* can be seen as exemplifying a general perspective on artificial intelligence, where data-driven learning mechanisms are integrated in a knowledge–centered reasoning framework.

V. Conclusion

In this paper we presented the knowledge infrastructure of a high-level artificial visual intelligent system, the *Cognitive Engine*. In particular we described how the conceptual specifications of basic action types can be driven by an hybrid semantic resource, i.e. HOMINE and its derived ontology patterns: for each considered action verb, the *Cognitive Engine* can identify typical FrameNet roles and corresponding lexical fillers (WordNet synsets), logically constraining them

[13] Note that here we are paraphrasing an axiom that exploits Davidsonian event semantics [35] and basic principles of formal mereology (see [25] and [36]). Also, this axiom is valid if every bag has a rope: this is generally true when considering garbage bags like the one depicted in figure7, but exceptions would need to be addressed in a more comprehensive scenario.

[14] With some exceptions, especially in California, around Mountain View!

to a computational ontology of actions encoded in ACTR-K through the SCONE Knowledge-Base system. Future work will be devoted to improve the *Cognitive Engine* and address *causal selectivity* (see II) using (1) reasoning and statistical inferences to derive and predict goals of agents and (2) mechanisms of abduction to focus on the most salient information from complex visual streams. We also plan to extend the system functionalities in order to support a wider range of action verbs and run tests on a large video dataset.

Acknowledgments

This research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-10-2-0061. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

References

[1] D. A. Forsyth and J. Ponce, *Computer Vision, A Modern Approach*. Prentice Hall, 2004.

[2] J. Anderson and C. Lebiere, *The Atomic Components of Thought*. Erlbaum, 1998.

[3] B. Tversky, J. Zachs, and B. Martin, "The structure of experience," in *Understanding events: From Perception to Action*, T. Shipley and T. Zacks, Eds., 2008, pp. 436–464.

[4] L. Albertazzi, L. Van Tonder, and D. Vishwanath, Eds., *Perception Beyond Inference. The Information Content of Visual Processes*. The MIT Press, 2010.

[5] J. M. Siskind, "Grounding language in perception," *Artificial Intelligence Review*, vol. 8, pp. 371–391, 1995.

[6] P. J. Hayes, "The second naïve physics manifesto," in *Formal Theories of the Common Sense World*, J. Hobbes and R. Moore, Eds. Ablex Publishing Corporation, 1985.

[7] M. Minsky, "A framework for representing knowledge," in *Mind Design*, P. Winston, Ed. MIT Press, 1997, pp. 111–142.

[8] A. Majid, J. Boster, and M. Bowerman, "The cross-linguistic categorization of everyday events: a study of cutting and breaking," *Cognition*, vol. 109, pp. 235–250, 2008.

[9] P. W. Singer, *Wired for War*. The Penguin Press, 2009.

[10] P. Maitikanen, R. Sukthankar, and M. Hebert, "Feature seeding for action recognition," in *Proceedings of International Conference on Computer Vision*, 2011.

[11] M. Poveda, M. C. Suarez-Figueroa, and A. Gomez-Perez, "Ontology analysis based on ontology design patterns," in *WOP 2009 Workshop on Ontology Patterns at the 8th International Semantic Web Conference (ISWC 2009). Proceedings of the WOP 2009.*, W. . . W. on Ontology Patterns at the 8th International Semantic Web Conference (ISWC 2009), Ed. WOP 2009 Workshop on Ontology Patterns at the 8th International Semantic Web Conference (ISWC 2009), 2009. [Online]. Available: http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-516/pap05.pdf

[12] W. Ceusters, J. Corso, Y. Fu, M. Petropoulos, and V. Krovi, "Introducing ontological realism for semi-supervised detection and annotation of operationally significant activity in surveillance videos," in *the 5th International Conference on Semantic Technologies for Intelligence, Defense,and Security (STIDS 2010)*, 2010.

[13] A. Gangemi and V. Presutti, "Ontology design patterns," in *Handbook on Ontologies*, ser. 2nd Edition, S. Staab and R. Studer, Eds. Springer, 2009.

[14] C. Alexander, *The Timeless Way of Building*. Oxford Press, 1979.

[15] A. Oltramari and C. Lebiere, "Mechanism meet content: Integrating cognitive architectures and ontologies," in *Proceedings of AAAI 2011 Fall Symposium of "Advances in Cognitive Systems"*, 2011.

[16] J. Ball, S. Rodgers, and K. Gluck, "Integrating act-r and cyc in a large-scale model of language comprehension for use in intelligent systems," in *Papers from the AAAI workshop*. AAAI Press, 2004, pp. 19–25.

[17] S. Douglas, J. Ball, and S. Rodgers, "Large declarative memories in act-r," in *Proceedings of the 9th International Conference of Cognitive Modeling*, 2009.

[18] B. Best, N. Gerhart, and C. Lebiere, "Extracting the ontological structure of cyc in a large-scale model of language comprehension for use in intelligent agents," in *Proceedings of the 17th Conference on Behavioral Representation in Modeling and Simulation*, 2010.

[19] B. Edmond, "Wn-lexical: An act-r module built from the wordnet lexical database," in *Proceedings of the 7th International Conference of Cognitive Modeling*, 2006, pp. 359–360.

[20] C. Fellbaum, Ed., *WordNet, An Electronic Lexical Database*. MIT Press, Boston, 1998.

[21] J. Ruppenhofer, M. Ellsworth, M. Petruck, and C. Johnson, "Framenet: Theory and practice," June 2005.

[22] G. Vetere, A. Oltramari, I. Chiari, E. Jezek, L. Vieu, and F. M. Zanzotto, "Senso comune, an open knowledge base for italian," *TAL - Traitement Automatique des Langues*, vol. 39, no. Forthcoming, 2012.

[23] C. Masolo, A. Gangemi, N. Guarino, A. Oltramari, and L. Schneider, "WonderWeb Deliverable D17: The WonderWeb Library of Foundational Ontologies," Tech. Rep., 2002.

[24] A. Gangemi, N. Guarino, C. Masolo, and A. Oltramari, "Sweetening wordnet with dolce," *AI Magazine*, vol. 3, pp. 13–24, Fall 2003.

[25] P. Simons, Ed., *Parts: a Study in Ontology*. Clarendon Press, Oxford, 1987.

[26] C. J. Fillmore, "The case for case," in *Universals in Linguistic Theory*, E. Bach and T. Harms, Eds. New York: Rinehart and Wiston, 1968.

[27] A. Burchardt, K. Erk, and A. Frank, "A wordnet detour to framenet," in *Sprachtechnologie, mobile Kommunikation und linguistische Resourcen.*, ser. Computer Studies in Language and Speech, B. S. Bernhard Fisseni, Hans-Christian Schmitz and P. Wagner, Eds. Frankfurt am Main: Peter Lang, 2005, vol. 8, pp. 408–421.

[28] A. Oltramari, "Lexipass methodology: a conceptual path from frames to senses and back," in *LREC 2006 (Fifth International Conference on Language Resources and Evaluation)*. Genoa (Italy): ELDA, 2006.

[29] J. F. Allen, "An interval based representation of temporal knowledge," in *7th International Joint Conference on Artificial Intelligence*. Vancouver: IJCAI, Morgan Kaufmann, 1983, pp. 221–226, vol.1.

[30] F. Baader, D. Calvanese, D. L. Mcguinness, D. Nardi, and P. F. Patel-Schneider, Eds., *The Description Logic Handbook : Theory, Implementation and Applications*. Cambridge University Press, 2003.

[31] S. Fahlman, "Using scones multiple-context mechanism to emulate human-like reasoning," in *First International Conference on Knowledge Science, Engineering and Management (KSEM'06)*. Guilin, China: Springer–Verlag (Lecture Notes in AI), 2006.

[32] D. Gildea and D. Jurafsky, "Automatic labelling of semantic roles," in *Proceedings of 38 th Annual Conference of the Association for Computational Linguistics (ACL-00)*, 2000, pp. 512–520.

[33] T. Pedersen, S. J. Patwardhan, and M. Michelizzi, "Wordnet :: Similarity: Measuring the relatedness of concepts," in *Demonstration Papers at HLT-NAACL*, 2004, pp. 38–41.

[34] L. Schooler and J. Anderson, "The disruptive potential of immediate feedback," in *Proceedings of the Twelfth Annual Conference of The Cognitive Science Society*, 1990, pp. 702–708.

[35] R. Casati and A. Varzi, Eds., *Events*. Aldershots, USA: Dartmouth, 1996.

[36] R. Casati and A. Varzi, *Parts and Places. The Structure of Spatial Representation*. Cambridge, MA: MIT Press, 1999.

Best-practice time point ontology for event calculus-based temporal reasoning

Robert C. Schrag
Digital Sandbox, Inc.
McLean, VA USA
bschrag@dsbox.com

***Abstract*—We argue for time points with zero real-world duration as a best ontological practice in point- and interval-based temporal representation and reasoning. We demonstrate anomalies that unavoidably arise in the event calculus when real-world time intervals corresponding to finest anticipated calendar units (e.g., days or seconds, per application granularity) are taken (naively or for implementation convenience) to be time "points." Our approach to eliminating the undesirable anomalies admits durations of infinitesimal extent as the lower and/or upper bounds that may constrain two time points' juxtaposition. Following Dean and McDermott, we exhibit axioms for temporal constraint propagation that generalize corresponding naïve axioms by treating infinitesimals as orthogonal first-class quantities and we appeal to complex number arithmetic (supported by programming languages such as Lisp) for straightforward implementation. The resulting anomaly-free operation is critical to effective event calculus application in commonsense understanding applications, like machine reading.**

***Index Terms*—temporal knowledge representation and reasoning, event calculus, temporal ontology best practices, temporal constraint propagation**

I. Introduction

Machine reading technology recently has been applied to extract temporal knowledge from text. The event calculus [8] presents appropriate near-term targets for formal statements about events, time-varying properties (i.e., fluents), and time points and intervals. While at least one implemented event calculus-based temporal logic [2] also has included calendar dates and clock times, most classical event calculus treatments address real-world time only abstractly. None so far has adopted the carefully crafted formulation of points (instants), intervals, dates, and times in Hobbs' and Pan's RDF temporal ontology [4]—which correctly treats all time units as intervals. We say, "correctly," because the casual treatment of a calendar or clock unit as a time point unavoidably leads to undesirable anomalies. This point may be subtle—ISO standard 8601 [3] pertaining to representation of dates and times states, "On a time scale consisting of successive steps, two distinct instants may be expressed by the same time point," and also (unfortunately, apparently circularly) defines an instant as a "point on the time axis." We hope, by demonstrating anomalies resulting from incorrect time point treatment and by presenting effective correct implementation techniques, to motivate future best-practice event calculus-based applications.

II. Event calculus ontology and axioms

We have implemented a temporal reasoning engine for an event calculus variant including the following ontological elements.

- Time intervals are convex collections of time points—intuitively, unbroken segments along a time axis.
- The ontological status of time points is an issue contended here. We argue that in the best practice they are taken to be instants with no real-world temporal extent, while naïvely (we argue incorrectly) finest anticipated calendar or clock units—which actually are intervals—have been taken as time "points." We take a time point to be a degenerate time interval—one whose beginning and ending points both are the time point itself.
- Fluents are statements representing time-varying properties—e.g., the number of living children a person has.
- The events of interest occur at individual time points and may cause one or more fluents to change truth value. E.g., the event of adopting an only child will cause the fluent hasChildren(Person, 0) to become false and the fluent hasChildren(Person, 1) to become true.

Figure 1 exhibits axioms defining the predicates we use to say when fluents "hold" (are true) and when events "occur" (happen).

holdsThroughout(fluent, interval) ↔ ∀(point): pointInInterval(point, interval) → holdsAt(fluent, point)
holdsThroughout(fluent, interval) ↔ ∀(sub): hasSubInterval(interval, sub) ∧ holdsThroughout(fluent, sub)

holdsAt(fluent, point) ↔ ∃(interval): intervalIsPoint(interval, point) ∧ holdsThroughout(fluent, interval)

holdsWithin(fluent, interval) ↔ ∃(sub): hasSubInterval(interval, sub) ∧ holdsThroughout(fluent, sub)

occursWithin(event, interval) ↔ ∃(point): pointInInterval(point, interval) ∧ occursAt(event, point)

Figure 1. Axioms relating holds and occurs predicates. Variables appearing on the left-hand side of an initial implication are universally quantified. Variables introduced on the right-hand side are quantified as indicated. The predicates relating time points and intervals are defined in the appendix.

Informally, a fluent holds throughout an interval *I* iff it holds at every point and throughout every subinterval contained by *I*. It holds (or occurs) within *I* iff it holds (or occurs) within some subinterval (or point) contained by *I*.

In the naïve approach, it's perfectly acceptable to assert that a fluent holds or that an event occurs "at" a specific "point" on the calendar or clock. We believe that under the preferred approach, in which the only (true) points directly accessible delimit the boundaries of measured time units, such assertions (or even queries) should be rare—perhaps limited to issues of legal status (e.g., one reaches the age of majority at exactly 12:00 midnight on one's 21st birthday). Thus, we commend preferred use of holdsWithin and occursWithin to replace naïve use of holdsAt and occursAt.

Besides being correct, the preferred approach is also more robust. In the naïve approach, supposing an enterprise decides to enhance its represented granularity from days to hours, it will need to replace all existing occurrences of holdsAt with holdsWithin (because its working definition of a "point" will have changed). As such, naïve approach users might as well avoid holdsAt and just use holdsWithin, which has equivalent semantics when its interval argument is a time point.

A given event calculus application also will include axioms to indicate which transition events initiate or terminate which fluents, as summarized by Schrag [7]. We don't need that much detail here, however, to demonstrate our concerns about undesirable anomalies arising from the naïve approach.

III. Anomalies Arising from the Naïve Time Point Approach

We discuss the following anomalies.

A. Inability to order time points within a finest represented time unit (e.g., a calendar day—see section A)

B. Inability to avoid inferred logical contradiction when contradictory statements hold at different real-world times within a finest represented time unit (see section B)

C. Inability to order real-world events occurring within a finest represented time unit (see section C)

D. Inability to avoid inferred logical contradiction when real-world events occur within a finest represented time unit and initiate contradictory fluents (see section D)

The time map in Figure 2 illustrates these anomalies, as discussed in the following subsections.

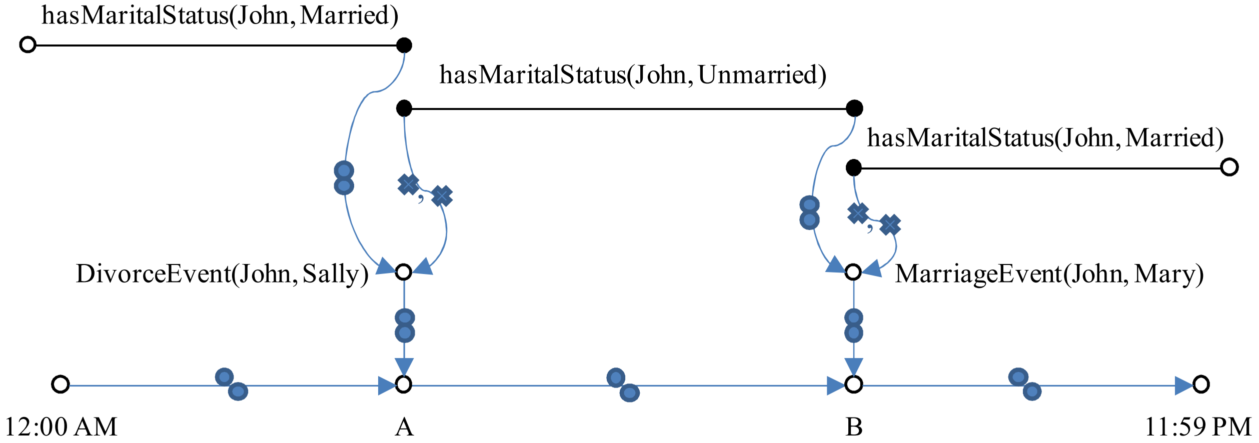

hasMaritalStatus(John, Married)
hasMaritalStatus(John, Unmarried)
hasMaritalStatus(John, Married)
DivorceEvent(John, Sally)
MarriageEvent(John, Mary)
12:00 AM
A
B
11:59 PM

Figure 2. Time map illustrating naïve approach anomalies. Fluent observations (top) include fluents and the intervals throughout which they hold. Dark-filled points indicate that associated fluents are known not to hold beyond their intervals' beginning or ending. Constraint graphics (with arrows) are defined in Figure 9, in the appendix. Transition event occurrences (middle) include the events and points where these occur. Contradictory fluents cannot overlap temporally, and, per event calculus convention, initiated fluent observations begin immediately after triggering transition events. The calendar (bottom) shows the initial and final minutes of a given day, plus two included time points, ordered as shown.

A. Inability to order time points

As is apparent in Figure 2, this basic problem underlies the other three listed above. In the naïve approach, the only way to order time points is to associate them with distinct finest calendar or clock units. Suppose days are the finest time unit represented. We'd like to assert the point-wise temporal relations (i.e., constraints) Figure 2 indicates, but in the naïve approach such constraints would be contradictory—all the points shown would resolve to the same calendar day's time "point," which cannot precede itself. This anomaly can be particularly troubling in the representation of statements extracted by machine reading from news articles, which frequently exhibit only calendar dates but cover sequences of events occurring within single days. The option of discarding such fine ordering information—and treating all within-day events as if they were simultaneous—is equally problematic. Rendering event orderings correctly is critical to representing causality—just one fundamental element of a true commonsense understanding that machine reading is hoped ultimately to support.

Even when our representation isn't fine enough to specify absolutely when during a given day (e.g.) a time point occurs, when we can order the points, we can avoid contradictions resulting from an incorrect presumption of simultaneity. Absent total (or even partial) ordering, we also can still hypothesize orders that might not lead to contradictions.

B. Inability to order contradictory holds statements

A person can't be both married and unmarried at the same time, as would be required if all the constraint-linked points in Figure 2 were collapsed onto a single day "point." In the naïve approach, it is (from a real-world perspective) as if we forced every marriage or divorce (indeed, every event) to occur at the stroke of midnight.

C. Inability to order events

In the naïve approach, we can say that a person divorced one spouse and married another on the same day, but we can't say in what order these events occurred.

D. Inability to order occurs statements initiating contradictory fluents

Without the ability to order events, we don't know whether any axiom proscribing polygamy has been violated or not. An implementation might take one position or another, depending on the order in which it happened to visit the transition events and to apply its rules for initiating and terminating fluents, detecting contradictions, and propagating constraints.

IV. Temporal Constraint Representation and Propagation

Compared to an application's finest represented calendar or clock unit, available real-world information may be more or less precise. E.g., we may know the year that a given event occurred but not the month or the day. If our finest represented units are days, this gives us an earliest and a latest possible date on which the event could have occurred (the first and last days of the year given). We use the notation distance(a, b, [x, y]) to indicate that the number of finest time units along a path from time point a to time point b has as a lower bound x and as an upper bound y.

Rather than expose our system-internal time units, we provide a user interface in terms of calendar and clock units—affording users source code-level robustness against future granularity enhancements. A distinguished calendar/clock point (e.g., the beginning point of the interval for 12:00 midnight, January 1, 1900) affords a reference against which the distance to other dates/times is calculated.

We refer to an asserted distance statement (or to a user-provided statement from which it is derived) as a temporal constraint.

Real-world information also may give us only qualitative information about the relationship between two time points—e.g., one is before or one is after the other. The following two figures exhibit axioms to define qualitative relations among time points—Figure 3 following the naïve approach, Figure 4 the preferred one. (See also Figure 9 in the appendix for graphical definitions of these relations.) Notice that the only difference between these two axiom sets is in their representation of the smallest possible distance between any two time points. In the naïve approach, it is one finest time unit. In the preferred approach, it is arbitrarily small—taken to be infinitesimal.

timePointEqualTo(a, b) ↔ distance(a, b, [0, 0])
timePointLessThan(a, b) ↔ distance(a, b, [1, ∞])
timePointGreaterThan(a, b) ↔ distance(a, b, [−∞, −1])
timePointGreaterThanOrEqualTo(a, b) ↔ distance(a, b, [0, ∞])
timePointLessThanOrEqualTo(a, b) ↔ distance(a, b, [−∞, 0])
hasNextTimePoint(a, b) ↔ distance(a, b, [1, 1])
hasPreviousTimePoint(a, b) ↔ distance(a, b, [−1, −1])
timePointTouching(a, b) ↔ distance(a, b, [−1, 1])
timePointGreaterThanOrTouching(a, b) ↔ distance(a, b, [−1, ∞])
timePointLessThanOrTouching(a, b) ↔ distance(a, b, [−∞, 1])

Figure 3. Axioms defining qualitative relations between time points in the naïve approach, where finest time units are treated as "points" and the smallest possible distance is one such time unit

timePointEqualTo(a, b) ↔ distance(a, b, [0, 0])
timePointLessThan(a, b) ↔ distance(a, b, [ϵ, ∞])
timePointGreaterThan(a, b) ↔ distance(a, b, [–∞, −ϵ])
timePointGreaterThanOrEqualTo(a, b) ↔ distance(a, b, [0, ∞])
timePointLessThanOrEqualTo(a, b) ↔ distance(a, b, [–∞, 0])
hasNextTimePoint(a, b) ↔ distance(a, b, [ϵ, ϵ])
hasPreviousTimePoint(a, b) ↔ distance(a, b, [−ϵ, −ϵ])
timePointTouching(a, b) ↔ distance(a, b, [−ϵ, ϵ])
timePointGreaterThanOrTouching(a, b) ↔ distance(a, b, [−ϵ, ∞])
timePointLessThanOrTouching(a, b) ↔ distance(a, b, [–∞, ϵ])

Figure 4. Axioms defining qualitative relations between time points in the preferred approach, where all time units are treated as intervals and we use an infinitesimal (denoted ϵ) to separate points that are (in the limit) "adjacent"

Both approaches use infinity (denoted ∞) to represent the largest possible distance between time points. Handling this in temporal constraint propagation (computing tightest distance bounds, considering all constraints) requires axioms defining non-standard arithmetic, as in Figure 5. Figure 6 exhibits axioms for the constraint propagation process in which Figure 5's arithmetic axioms are applied. Note that all but the last of Figure 5's axioms handle only the infinities specially. By treating the positive infinitesimal denoted ϵ as the imaginary number *i* (as in [2][5][6]) and by appealing to complex arithmetic, we can use the same axioms to support propagation in both approaches.

Note that in the naïve approach using only real numbers all the imaginary parts will be zero. The only substantive difference between the two approaches' computational complexity for constraint propagation is that the preferred approach enables finer (and thus more numerous unique) constraints.

Implementation is straightforward for addition and arithmetic negation in a programming language such as Lisp that supports complex numbers and arithmetic. While complex numbers with unequal real and/or imaginary parts are incomparable with respect to magnitude, in our imaginary-as-infinitesimal interpretation the real parts always dominate and the imaginary parts are compared only when the real parts are equal—per the last axiom defining finite>, in which the predicates real>, real=, and imaginary> invoke the indicated comparisons on the real and imaginary parts of their arguments.

infinite(–∞)
infinite(∞)

infinite+(–∞, –∞, –∞)
infinite+(∞, ∞, ∞)
infinite+(a, –∞, –∞) ← ¬infinite(a)
infinite+(–∞, b, –∞) ←¬infinite(b)
infinite+(a, ∞, ∞)← ¬infinite(a)
infinite+(∞, b, ∞) ← ¬infinite(b)
infinite+(a, b, a + b) ← ¬infinite(a) ∧ ¬infinite(b)

infinite–(–∞, ∞)
infinite–(∞, –∞)
infinite–(a, –a) ← ¬infinite(a)

infinite>(∞, –∞)
infinite>(a, –∞) ← ¬infinite(a)
infinite>(∞, b) ← ¬infinite(b)
infinite>(a, b) ← ¬infinite(a) ∧ ¬infinite(b) ∧ finite>(a, b)

finite>(a, b) ← real>(a, b) ∨ (real=(a, b) ∧ imaginary>(a, b))

Figure 5. Axioms supporting constraint propagation arithmetic (addition, subtraction, and comparison) over temporal duration bounds of infinite extent

distance(b, a, [–y, –x]) ↔ distance(a, b, [x, y]) ∧ infinite–(x, –x) ∧ infinite–(y, –y)
distance(a, b, [w, y]) ← distance(a, b, [x, y]) ∧ distance(a, b, [w, z]) ∧ infinite>(w, x)
distance(a, b, [x, z]) ← distance(a, b, [x, y]) ∧ distance(a, b, [w, z]) ∧ infinite>(y, z)
distance(a, c, [mo, np]) ← distance(a, b, [m, n]) ∧ distance(b, c, [o, p]) ∧ infinite+(m, o, mo) ∧ infinite+(n, p, np)

Figure 6. Axioms for propagating lower and upper temporal bounds to infer tightest bounds considering all constraints

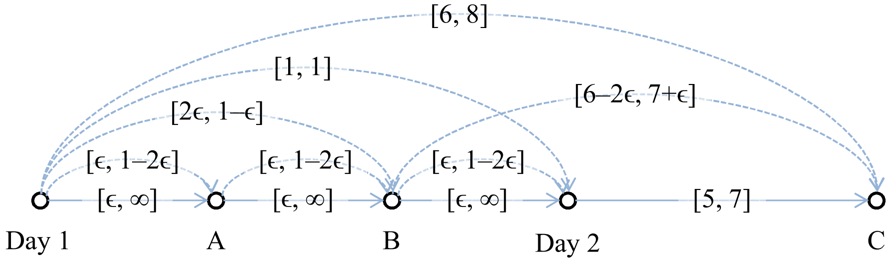


Figure 7. Raw (solid arrow) and inferred/propagated (dashed arrow) constraints, with lower and upper bounds, in the preferred approach. Constraints have directions indicated by arrows (all oriented from left to right)

V. How the Preferred Approach Avoids Anomalies

To see how constraint propagation works—and avoids anomalies—in the preferred approach, see Figure 7, which supposes days are our finest time unit.

By way of raw constraints, we know that points A and B both fall between Day 1 and Day 2, that A follows B, and that point C is between five and seven days after Day 2. For clarity, Figure 7 omits the [ϵ, ∞] constraint from Day 1 to B and from A to Day 2, as well as many inferred constraints relating pairs of points not connected in the figure. The two-dimensional (in the implementation, complex) arithmetic treating infinitesimal and non-infinitesimal quantities orthogonally effectively maintains qualitative point ordering—both within finest represented calendar or clock unit boundaries (e.g., relating points A and B) and across them (relating B and C). See Figure 8.

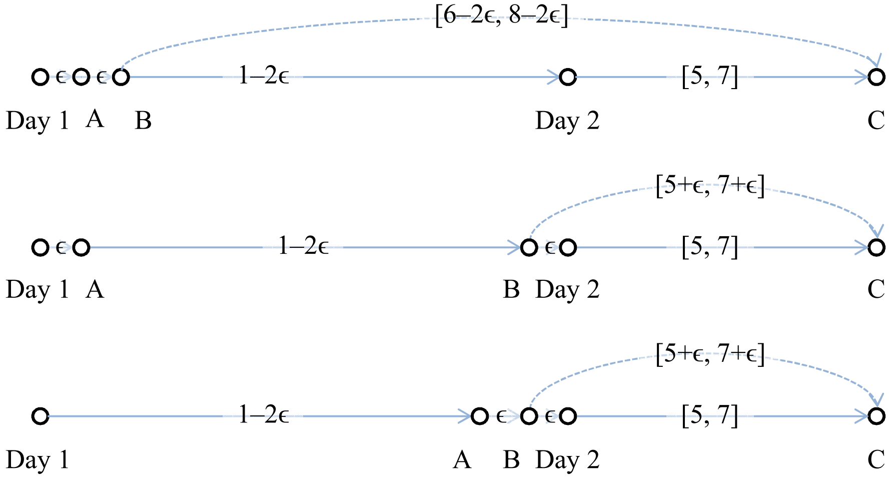


Figure 8. Extreme cases for the time points A and B in Figure 7, including (at the extremes) greatest lower and least upper bounds in the inferred constraints shown there

As we explained in section III, resolving this time point ordering anomaly simultaneously resolves the other three anomalies described there as well. Now, we also can order the events that occur at time points and avoid spurious contradictions that arise from the naïve approach's inability to order events and fluent observations. When our finest time units are days, we no longer have to pretend that all events occur at the stroke of midnight. With appropriate ordering of events, we'll be able to put machine reading in a better position to support commonsense understanding of causality.

VI. Summary

We have demonstrated temporal reasoning anomalies that arise when implementation of the event calculus naively follows classical treatments that casually treat finest represented calendar or clock time intervals as "points." We have presented axioms and described implementation techniques to resolve these anomalies when all time intervals are correctly treated as time intervals and when time points are taken to be instants with zero real-world duration extent. We argue that this preferred approach, rather than the naïve one, is needed for the event calculus to be useful in applications, like machine reading, intended to support commonsense understanding including causality.

Acknowledgment

Thanks to other participants in DARPA's Machine Reading research program that supported our temporal reasoning implementation—especially to other members of the SAIC evaluation team, including Global InfoTek (the author's former employer).

References

[1] J. Allen, "Maintaining knowledge about temporal intervals," in Communications of the ACM. 26, pp. 832–843, November 1983.

[2] T. Dean and D. McDermott, "Temporal data base management," Artificial Intelligence, vol. 32, pp. 1–55, 1987.

[3] International Standards Organization, "Data elements and interchange formats—information interchange—representation of dates and times," international standard ISO 8601:2004(E), third edition, 2004.

[4] J. Hobbs and F. Pan, "An ontology of time for the semantic web," ACM Transactions on Asian Language Information Processing, Vol. 3, No. 1, pp. 66–85, March 2004.

[5] R. Schrag, J. Carciofini, and M. Boddy, "Beta-TMM Manual (version b19)," Technical Report CS-R92-012, Honeywell SRC, 1992.

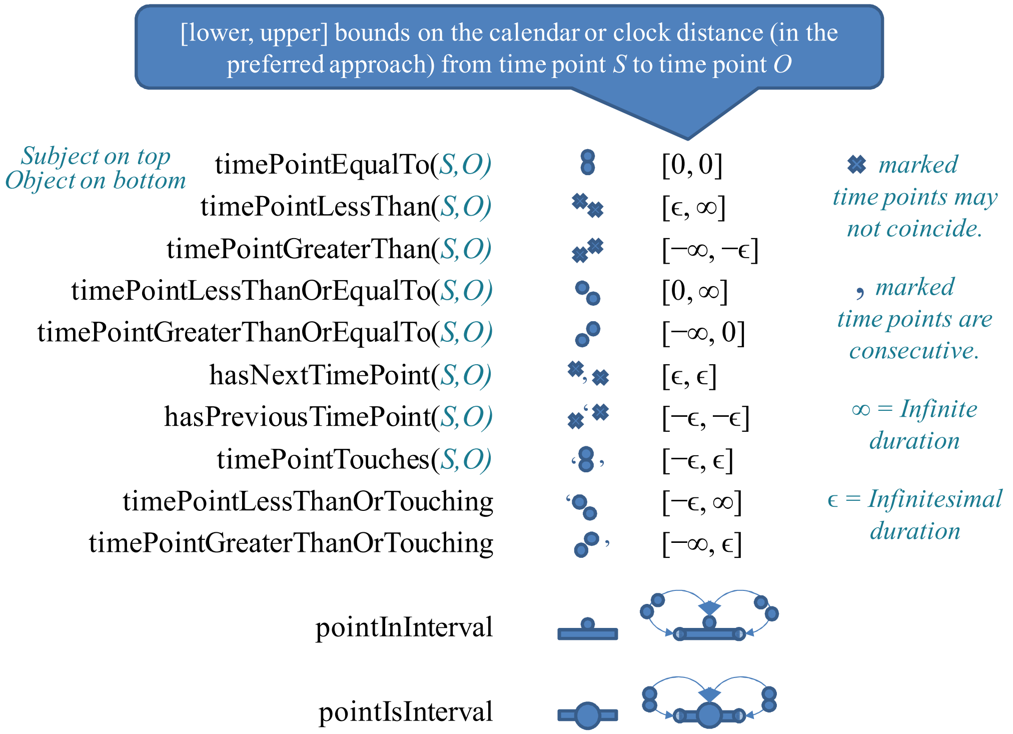
[6] R. Schrag, M. Boddy, and J. Carciofini. "Managing disjunction for practical temporal reasoning," in Principles of Knowledge Representation and Reasoning: Proceedings of the Third International Conference (KR-92), pp 36–46, 1992.

[7] R. Schrag, "Exploiting inference to improve temporal RDF annotations and queries for machine reading," 7th International Conference on Semantic Technologies for Intelligence, Defense, and Security (STIDS), 2012.

[8] M. Shanahan, "The event calculus explained," in Artificial Intelligence Today, ed. M. Wooldridge and M. Veloso, Springer Lecture Notes in Artificial Intelligence no. 1600, pp.409–430, 1999.

Appendix: Time point and interval relations

The set of predicates illustrated in Figure 9 (repeated from Figure 4) supports every qualitative binary time point relation over the time point distance landmark values indicating equality, adjacency, and lack of constraint above or below. (A user also may specify arbitrary bounds on the number of time units intervening between any two points.) As illustrated in Figure 10 selected examples, this point orientation yields a much broader set of qualitative interval relations than does Allen's classical formalism [1], which is purely interval oriented, without points.


[lower, upper] bounds on the calendar or clock distance (in the preferred approach) from time point S to time point O
Subject on top
Object on bottom
✖ marked time points may not coincide.
, marked time points are consecutive.
∞ = Infinite duration
ϵ = Infinitesimal duration

timePointEqualTo(S,O)	[0, 0]
timePointLessThan(S,O)	[ϵ, ∞]
timePointGreaterThan(S,O)	[−∞, −ϵ]
timePointLessThanOrEqualTo(S,O)	[0, ∞]
timePointGreaterThanOrEqualTo(S,O)	[−∞, 0]
hasNextTimePoint(S,O)	[ϵ, ϵ]
hasPreviousTimePoint(S,O)	[−ϵ, −ϵ]
timePointTouches(S,O)	[−ϵ, ϵ]
timePointLessThanOrTouching	[−ϵ, ∞]
timePointGreaterThanOrTouching	[−∞, ϵ]
pointInInterval	
pointIsInterval	

Figure 9. Qualitative relations over time points, with graphical icons that we use to illustrate the definitions of point-and-interval relations (here) and interval-interval relations (in Figure 10). Such illustrated definitions include beginning and ending points super-imposed on interval icons, to elucidate the constraints.

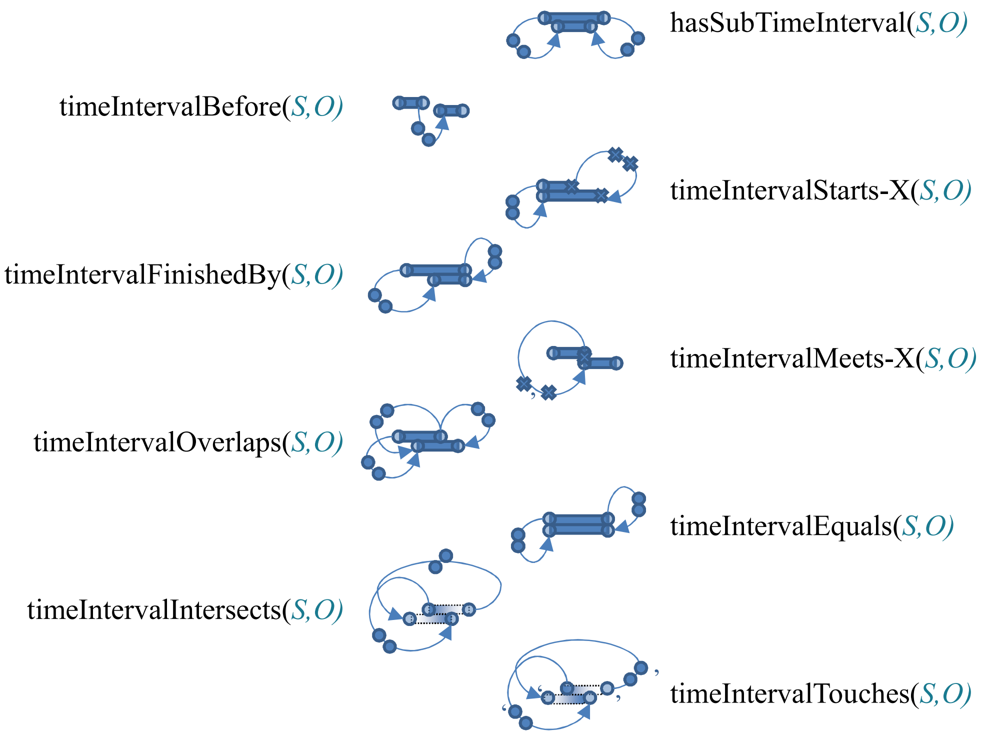


Figure 10. Selected relations over time intervals (with defined time point relations indicated)

Constellation

A Prototype ISO/IEC 11179 Metadata Registry

Gramm Richardson
U.S. Department of Defense
gpricha@tycho.ncsc.mil

Elli Schwarz
SRA International
eliezer_schwarz@sra.com

Abstract—Different systems across the government, as well as in the private sector, use different country names or country codes to represent the notion of a "country" within a particular problem domain. These systems may choose to represent countries using a particular standard for county names and country codes. Often times these systems find themselves interacting with other systems that may use another standard for country representation. This makes it difficult to compare and link country-related data in a consistent fashion. We describe our work on the Constellation system using the ISO/IEC 11179 metadata standard to register the various country code sets in a common metamodel. This facilitates management, querying, updating and mapping the elements within the code sets.

Keywords: metadata, country codes, ontology

I. INTRODUCTION

There exist numerous international and national standards for country and country code representations. Some are designed to represent countries within a certain domain, such as the ITU-T e.164 [1] codes to represent telephone dialing codes for countries, or the ICAO [2] codes to represent country prefixes for airplane tail numbers. Other codes are attempts at international or national standardization, such as ISO 3166 [3] codes and NGA Geopolitical Codes [4]. Each of these standards has its own terminology and criteria for inclusion in its list.

Unfortunately, there is no unambiguous, standard definition of the term "country" [5]. Many country code sets contain entries for entities that might not be thought of as countries in the common usage of the word. A code set may consider a semi-autonomous or dependent entity to be a country in its own right, or it may include non-country placeholders such as "reserved" or "unknown". Some code sets may list a region or entity for practical, political, or diplomatic considerations, notwithstanding the entity's precise legal status.

To further complicate matters, these country lists are not static. Dependent territories may become independent, civil wars may split countries, two countries can unify, or a country may simply decide to change its official name. To keep up with changing realities, many of these code sets or standards organizations publish updates to their lists from time to time. This adds a chronological dimension to the maintenance of county code sets.

All of the above factors make it necessary to maintain these code sets together in one registry that can facilitate the management, querying and updating of these code sets. This registry can also provide a framework for tackling the challenge of mapping entities from one code set to another.

This rest of this paper describes the Constellation metadata registry system, which uses the ISO/IEC 11179-3 Edition 3 registry metamodel [6] standard to register and map country code sets. We will describe in more detail the nuances of common country code management challenges. We will discuss our approach to designing a country code registry using an OWL ontology based on the ISO/IEC 11179 metamodel, and explain how we handle updates. We will also describe our algorithm used to match countries across code sets.

II. COUNTRY CODE MANAGEMENT CHALLENGES

The complex nature of country data poses several challenges for its management in a registry:

- A country/geopolitical entity may have an official name and several alternate names, and some of these names may be in multiple languages.
- In some country code standards, there may be multiple code formats for each country. For example, in ISO 3166-1, each country has trigraphs, digraphs, and numeric codes, whereas other standards may have only one code format per country.
- One country may have multiple codes in one format, such as in the ICAO Nationality Marks code set. In that code set, South African aircraft can bear the nationality marks "ZS", "ZT", or "ZU".
- Multiple countries in a single code set may share the same code, such as in ITU-T e.164, where 25 countries share the country dialing code "1".
- A geopolitical entity may be a dependency of another country, like a state, territory, province, or outlying area. In ISO 3166, these entities are listed in a separate code set for dependencies, ISO 3166-2. The code set ISO 3166-1 is used for what it considers to be "top-level" (usually independent) countries. In ITU-T e.164, the dependency may be explicitly written out as part of the country name in parenthesis, as in the case of "Greenland (Denmark)". In other code sets, the administrator is ignored.
- Some code sets may have entries for regions (such as Europe or Asia) or transnational groups (such as EU, UN, or NATO) which are not traditionally thought of as countries.

- Code sets change over time. New versions of code sets might be released, and updates to individual entities in the code set, like code or name changes or even spelling corrections, might be issued.

Using an ontology can be the first step toward managing some of the above complexities. The UN FAO (Food and Agriculture Organization) ontology [7] illustrates one approach to add some degree of structure to the attributes of a country or region. It provides an OWL ontology with properties such as fao:nameOfficial and fao:nameShort for the different forms of a country name (with a language tag to indicate the language of the name), fao:validSince and fao:validUntil for valid dates for a particular country, and fao:isAdministeredBy to represent the administering country. It also provides many other additional properties of importance to countries, such as fao:sharesBorderWith, fao:predecessorOf, fao:memberOf, and other useful properties.

Additionally, SKOS [8] can be used to provide some level of abstraction to the concept of a country and its name and code representations. Using the SKOS vocabulary in OWL provides the skos:Concept class, and instances of this class can represent countries, with properties such as skos:prefLabel to represent the preferred name, and skos:altLabel to represent other names (with language tags on the literal to represent the language of the name). SKOS Mapping Properties such as skos:closeMatch and skos:broadMatch can be used on these country instances to map similar countries or country relationships. SKOS Documentation Properties such as skos:note or skos:changeNote can be used to further describe a country and changes to a country.

Methods of supplying the country code for a SKOS country concept have also been proposed in [9]. One possibility mentioned there is adding new properties for the different types of codes (iso3166:twoLetterCode or iso3166:numericalCode), or using a skos:prefLabel with a special private language tag to indicate the code type (such as using the skos:prefLabel property with "FR"@x-notation-twoletter as the literal).

SKOS-XL [10] has been proposed to further extend SKOS. It provides a class skosxl:Label to further abstract the notion of a name from the country it represents, so the name can have its own properties independent of the country itself. Thus, a date or other provenance information pertaining to the name can be accommodated [11]. The Library of Congress proposed an additional ontology, MADS/RDF [12], which builds on SKOS but provides additional classes and properties designed to model geographic and other kinds of names, as well as thesauri and other controlled value lists. The Library of Congress MARC [13] codes use the MADS/RDF ontology to represent its list of geographic areas.

Using these ontologies are a good start toward registering country code metadata in a way that manages many of the complexities listed above. However, we cannot expect that each country code set we want to register will provide their data in this fashion. Some existing code sets are provided as CSV files, with columns mapping country names to country codes, without any schema at all. Many other code sets are available only as tabular data embedded in web pages or text documents that we converted to CSV. Therefore, it is important that we allow any vocabulary or data format to be used in each particular code set, and rely on our own internal metamodel to accommodate all of these diverse data models in a uniform fashion.

Furthermore, it is important that whatever internal metamodel we use not be proprietary, and be able to handle updates to the data without losing the data contained in earlier versions. Using a standard metamodel would enable a more widespread use and understanding of our system, and would also enable it to be used by other kinds of data besides country codes, to facilitate integration with a wider range of problem domains. Maintaining a version history of the data would be of great use if the system were to integrate with other systems that contain data from an earlier point in time. To accommodate all these issues, we chose to develop the Constellation system using the ISO/IEC 11179 metamodel standard [6] to register our country code metadata. This standard, with some of our own minor extensions, enables us to build a system that can not only register countries, codes, and mappings among these countries, but also handle different versions of the various code sets and updates.

III. Implementing the ISO/IEC 11179 Metamodel in OWL for Constellation

The goals of the Constellation country code metadata registry are to represent the metadata using a consistent terminology, provide a uniform way of querying the data, manage updates without disrupting previous versions of the data, and facilitate storing relationships between data elements.

The ISO/IEC 11179 metamodel describes a variety of classes, attributes, and associations between classes useful for representing metadata about country objects. In Constellation, we implemented these classes and attributes in an OWL ontology. We represent the set of all countries in a code set as an instance of the Conceptual_Domain class, and the set of country codes in that code set as a Value_Domain. Each Value_Domain can represent one country code format (e.g., digraph or numeric). In most code sets we registered, there is only one code format for each country, so there would be one Value_Domain. In other code sets, for example ISO 3166-1, there are three code formats for each country – the trigraph, digraph, and numeric codes. Each of these formats would be a separate Value_Domain within the Conceptual_Domain for ISO 3166-1. The Value_Domain is made up of a set of Permissible_Values that contain the code (known as the "permitted value") for a country.

Each country entry is modeled as a Value_Meaning within a Conceptual_Domain. The Conceptual_Domain is thus made up of a set of Value_Meanings. Each country can contain several names (official names or other forms of the name), in multiple languages. In order to separate the concept of "country" from that of its name, we use the 11179 Designation class to represent a label or name for a country Value_Meaning. This Designation contains a "sign" property containing the actual country name, and a language identifier property to represent the language used for that name. We use a Designation_Context to describe the "acceptability" of a Designation within the context of a Conceptual_Domain. The acceptability ratings are described in ISO/IEC 11179 as being on a scale of: preferred, admitted, deprecated, obsolete and superseded. Only one Designation per language is “preferred” in a given Context; we use “admitted” to represent the other forms of the name.

Value_Meanings and Permissible_Values each contain a property for begin_date and optional end_date. This is used to represent the time period when the code set considers that value to be part of its official list. Instances of these classes without an end_date are considered to be the latest valid entry. We extended the 11179 standard to add these date fields to the Designation_Context as well. If a code set has several versions (such as when new countries are added, names or codes change, etc.) we can represent this with multiple instances of the class, each with a different date range. A diagram depicting an example of some instances of these classes can be found in Fig. 1.

The 11179 standard also provides a way to depict relationships among concepts. We use this feature to represent relationships among countries, such as when an entity is part of another country or is administered by another country. We also use this feature to represent relationships among countries that are likely to be close matches (i.e. the country named “United States” in the different code sets). These matches can be generated manually or by machine. Constellation’s semi-automated country matching algorithm [14] suggests matches based on the similarity of the names of countries in different code sets. The suggestions are then evaluated by a person who marks them as either correct or incorrect. These human judgments are recorded as rules that are used when automatically aligning entities in different code sets. We explain our approach to store these relationships in more detail later.

The Constellation system can thus be used to keep track of countries, country names, country codes, relationships among countries, and different versions of all of these pieces of information. This system has been successfully applied to over 15 different code sets, and it is easy to add additional ones. Table 1 shows some of the code sets we’ve used along with a brief description of how the code set is used.

TABLE I. CODE SETS REGISTERED IN CONSTELLATION

Code Set	**Description**
International Organizations	
International Civil Aviation Organization	Aircraft nationality marks based on the Chicago Convention on International Civil Aviation, as reported to ICAO by national administrations. Used as the prefix of an aircraft tail number.
International Olympic Committee	Codes identifying the National Olympic Committees/National Teams participating in the Olympics
ISO 3166-1, ISO 3166-2	Entities which are members of the UN or one of its specialized agencies and parties to the Statute of the International Court of Justice, or registered by the UN Statistics Division. Part 2 of the standard includes dependencies of the entities in Part 1.
UN FAO Geopolitical Ontology	AGROVOC, FAOSTAT, FAOTERM - code sets used for agricultural statistics and projects purposes
UN M.49 Area Codes	Used by the United Nations for statistical purposes
U.S. Government	
Census Schedule C	Used by the US Census Bureau as well as the Army Corps of Engineers
Treasury International Capital Reporting	Designations identifying countries in data files on international portfolio capital movements reported to the US Treasury Department via the Treasury International Capital reporting system.
GSA Geographic Locator Codes	Used by US federal agencies for reporting data to the Federal Real Property Profile.
NGA Geopolitical Codes (and dependencies)	Codes for political entities in the NGA GEOnet Names Server (Formerly FIPS 10-4).
Industry	
ITU-T e.164	Recommendation that defines structure for telephone numbers, including country dialing codes
ITU-T e.212	Defines the code used in the Mobile Country Code portion of an IMSI (International Mobile Subscriber Identifier)
International Union of Railways	Standard numerical country coding for use in railway traffic. Used as the owner’s code (3rd and 4th position) of a 12-digit wagon identification number.

IV. DATA INGESTION AND UPDATES

In order to facilitate the easy ingestion of data of all types, we have two main ingestion workflows: ingesting CSV files and RDF files. For CSV, we require some basic columns such as country name (with separate columns for preferred names, and other languages), columns for dates, and columns for country codes. The column headers need to be one of several that we have pre-defined. In order to ensure that all data is ingested into the system in a uniform fashion, we first convert the CSV into a general-purpose RDF format suited for easy conversion to our OWL representation of the 11179 format. We also take RDF country data in any format (such as UN FAO data, Library of Congress MARC codes, and country currency data, each of which uses a different ontology) and convert that to the general-purpose RDF format using SPARQL 1.1 scripts custom written for each of these RDF ontologies. Once this data is in the general-purpose RDF format, it is then ingested using another SPARQL 1.1 script to convert the general-purpose RDF to RDF conforming to our OWL implementation of the 11179 metamodel.

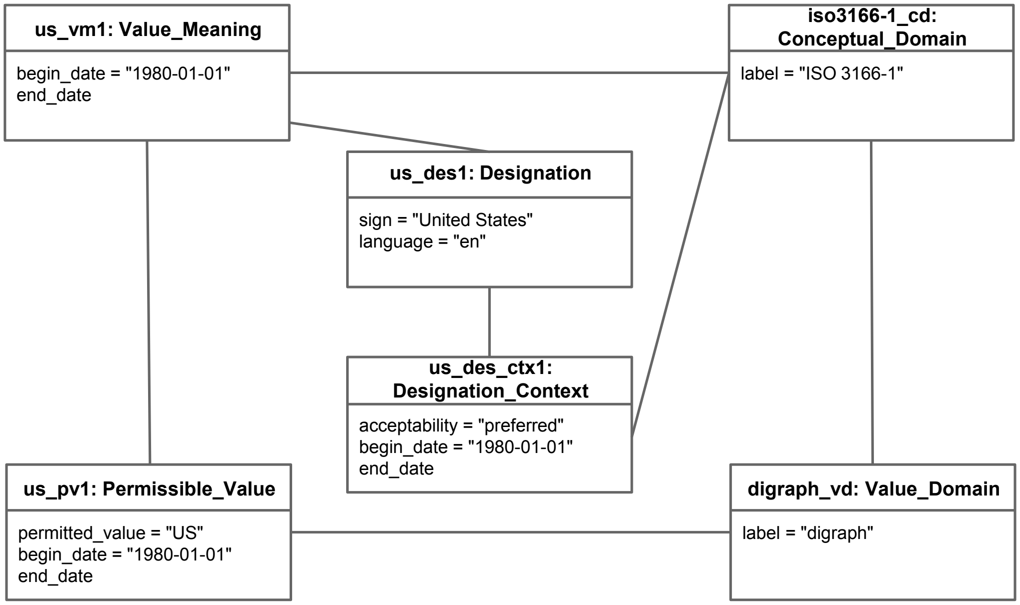

us_vm1: Value_Meaning
begin_date = "1980-01-01"
end_date
iso3166-1_cd: Conceptual_Domain
label = "ISO 3166-1"
us_des1: Designation
sign = "United States"
language = "en"
us_des_ctx1: Designation_Context
acceptability = "preferred"
begin_date = "1980-01-01"
end_date
us_pv1: Permissible_Value
permitted_value = "US"
begin_date = "1980-01-01"
end_date
digraph_vd: Value_Domain
label = "digraph"

Figure 1. UML object diagram showing an example of Constellation's use of ISO/IEC 11179 metamodel, edited for clarity

Updates to the country code sets are performed in a purely additive fashion. No statements are actually removed from the RDF store when performing update operations on country, country code, or country name data. Each of these entities may be updated separately, allowing for incremental updating of code sets. In the case of ISO 3166-1, updates are issued on an irregular basis every few months as update newsletters. The last full version of ISO 3166-1 was published in 2006, and keeping that code set current requires implementing the updates described in the newsletters. These newsletters might correct a spelling mistake in a name, change one numeric code to another, add a new country, or describe other changes. As stored in the Constellation metadata registry, country entities, codes, and country names each have begin_dates and optional end_dates associated with them. In the case of country names, the dates are associated with the acceptability of its usage in a particular Designation_Context. If a code set removes an entry, it is not actually deleted from our database, but it is marked with an end_date reflecting the date this entry was removed from the code set. Any data that has an end_date is not considered part of the current set of values but as part of an earlier version of the code set.

This use of dates on Designation_Contexts is an extension to the ISO/IEC 11179 metamodel being used in Constellation. With this extension we can record a country name change in a particular standard. For example, Libya in ISO 3166-1 has changed its name. In 2006, the country was identified in ISO 3166-1 by its official long-form English name, "the Socialist People's Libyan Arab Jamahiriya", in addition to a short form of the name. Following that country's civil war in 2011, the ISO 3166 Maintenance Agency issued an update to the country's name in a November, 2011 newsletter, which removed the long-form English name from the entry for the country.

To reflect this change in Constellation, an end_date value is added to the Designation_Context relating the former name to the code set. A new Designation_Context reflecting the name's new status (in this case, "deprecated") is added and given a begin_date. The RDF statements express the fact that a given country name ceased to be accepted and began to be deprecated on a particular date. If, rather than simply being removed, the name was changed, new statements would be added to relate the new name to the existing country and describe its usage acceptability, context, and the dates when it was used. Fig. 2 shows an RDF diagram using date fields to deprecate the old long-form name of Libya.

V. Country Matching and Relations in ISO/IEC 11179

When choosing a metamodel, there are many ways to model the relationships between countries across code sets. Our first approach was a country-centric approach, where we would define a unique URI for each country. Constellation's semi-automated country matching algorithm [14] was used to determine which countries were the same or similar across code sets. That URI would be used in all code sets as the Value_Meaning representing the notional country.

However, that approach proved problematic for many reasons. First and foremost, two different code sets may not have the same complement of values, so a given URI might not have statements in each code set. Additionally, we don't know that each standard refers to the exact same country, even if the same name is used. For example, one code set may have an entry for United States, which would include all states and dependent territories. Another code set may have separate entries for the United States, excluding territories, and separate entries for each of the territories. A code set may even include the territories as part of its definition of United States yet still have separate entries for some of these territories. For these reasons, having one URI for United States that would be shared across code sets clearly would not be appropriate, since each code set may have a slightly different interpretation of what is indicated by the country name.

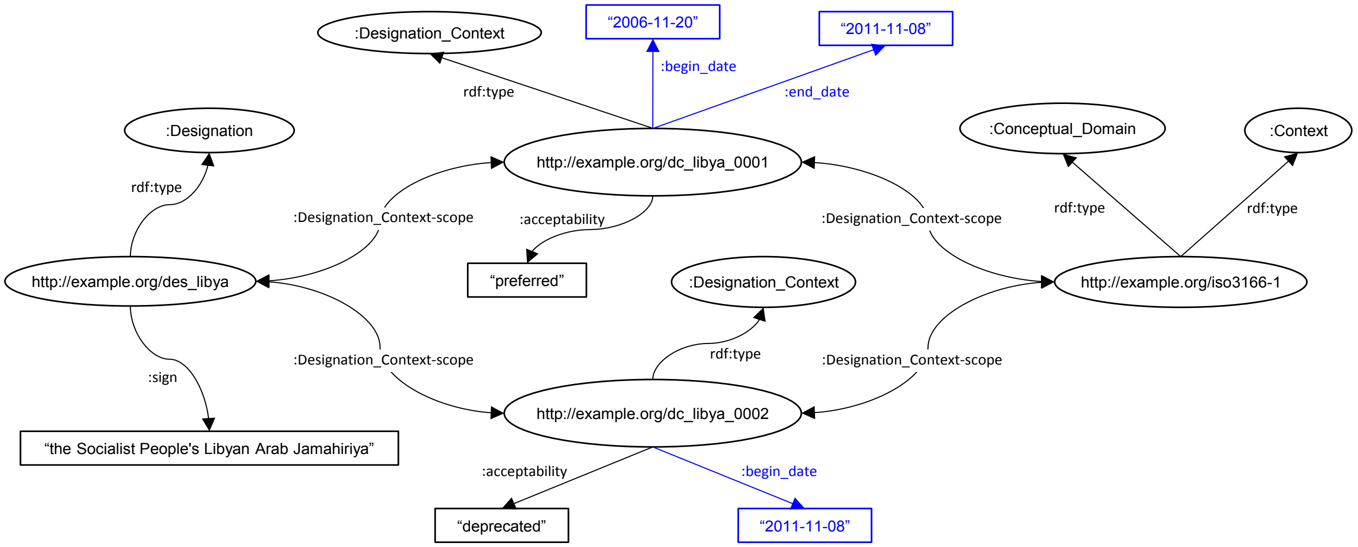

:Designation_Context
"2006-11-20"
"2011-11-08"
:begin_date
:end_date
rdf:type
:Designation
http://example.org/dc_libya_0001
:Conceptual_Domain
:Context
rdf:type
:Designation_Context-scope
:acceptability
:Designation_Context-scope
rdf:type
rdf:type
http://example.org/des_libya
"preferred"
:Designation_Context
http://example.org/iso3166-1
:Designation_Context-scope
rdf:type
:Designation_Context-scope
:sign
http://example.org/dc_libya_0002
"the Socialist People's Libyan Arab Jamahiriya"
:acceptability
:begin_date
"deprecated"
"2011-11-08"

Figure 2. RDF diagram showing how Constellation handles deprecated country names

Another example of this problem is that in some standards the country China includes Hong Kong and Macau, whereas in other standards each one has its own disjoint representation. If we had one URI for China, there would be ambiguity as to what is meant by that URI—is that the URI of all of China and its dependencies, or of just mainland China? Another example is Sudan and South Sudan—one code set might have a separate entry for South Sudan (which recently became independent from Sudan), as well as for Sudan itself. However, another code set may contain one entry for Sudan, meaning both Sudan and South Sudan. This may be based on the different dates of the code set, if one code set wasn't yet updated after South Sudan's independence, or the code set may not recognize South Sudan's independence.

Another issue with using a unique URI for each country is that two code sets may use completely different names for the same country. The reason that different names may be used in a given code set may be politically motivated. The country identified in the international ISO 3166 standard as "Myanmar" is referred to by the name "Burma" in official U.S. Government documents. The entity identified as "Taiwan, Province of China" in ISO 3166 is called "Chinese Taipei" by the International Olympic Committee. Although these entries have different names, technically they are referring to the same entity.

In all of the above cases, it is debatable whether it makes sense to use the same URI for the notional country across all code sets. Since each code set has its own idea of what an entry actually refers to, it is very difficult to determine if two code sets are using a country name in exactly the same way [15]. Therefore, we decided that each code set would use its own set of URIs (unique Value_Meanings) for its own values. Instead of relying on a common URI to map countries from one code set to another, we use 11179 Relations, which provide a way to link countries across code sets. For the names of the relationships, we use the SKOS vocabulary terms where appropriate (such as skos:closeMatch or skos:broadMatch). Use of skos:exactMatch and owl:sameAs was avoided for the same reasons we chose not to use the same URI. The 11179 standard doesn't provide date properties for these Relations, but we can add these fields to keep track of versions just as we did for countries above.

VI. Querying Challenges Using the ISO/IEC 11179 Metamodel

The generic nature of the 11179 metamodel adds a great deal of complexity and abstraction to the representation of the data. This poses a challenge for querying, since even a simple query getting all country codes for a given country name can involve traversing a large amount of RDF, resulting in a lengthy and difficult to read SPARQL query. The 11179 Relations which we used to link related concepts to each other also adds a great deal of complexity and extra statements. This is because the 11179 relations model is best suited to scale to ternary, quaternary, and higher-order relations, but it adds additional overhead when dealing with simpler binary relations, as will be explained below.

We attempted to provide shortcuts in the data we ingested, but this resulted in losing some of the benefits of 11179, particularly when it came to updates. We were able to simplify querying using shortcuts such as adding an rdfs:label directly to a Value_Meaning, instead of using Designations with a "sign" property, eliminating an extra statement traversal. However, this did not allow for dates to be provided for the label itself. Eliminating Designation_Context and adding alternate name forms directly in the Designation posed a similar problem managing the acceptability ratings. Since we don't want to actually delete any data from our system, in order to keep previous versions of data we needed these abstractions of Designation and Designation_Context, so we can maintain dates and acceptability ratings on the Value_Meaning and Designation_Context objects independently.

We experienced similar problems using shortcuts for 11179 Relations. In the 11179 metamodel, traversing the graph from one Concept to another Concept related by a Relation requires stepping through three intermediate objects rather than just a single predicate. We attempted to add convenience predicates (such as skos:broader) for these Relations to provide only one statement linking the two Concepts. As a result of this simplification, the SPARQL queries using the convenience predicates were much shorter and easier to read, but the convenience predicates lacked much of the descriptive power of the 11179 Relations. Fig. 3 shows a simple example of the way that relationships are represented in the 11179 metamodel, compared to how they are represented in SKOS.

Due to our issues with shortcuts, we determined that they were not a suitable approach for Constellation, and as a result we have some long, complex queries. We are exploring the use of SPIN [16] functions to pre-define query patterns for some of the complex parts of the 11179 metamodel. We would then call these functions in our queries. Although this may not improve query efficiency (unless the SPARQL implementation incorporates some efficiencies or caching for the SPIN functions), it should help a great deal with query readability and maintainability.

VII. Conclusion

We have shown how the 11179 metamodel can be used to register, query, and track updates to country code data. We have also demonstrated how 11179 can be used to track relationships among countries, such as country group memberships and administration. We have also shown how we can link similar countries together using 11179 relationships.

Applications of our work extend beyond just country code mapping. We have used it to model country currencies, and even to store thesaurus information, including taxonomies (such as the FAA Aviation Safety Thesaurus and the ETDE/INIS Joint Thesaurus of nuclear energy terminology).

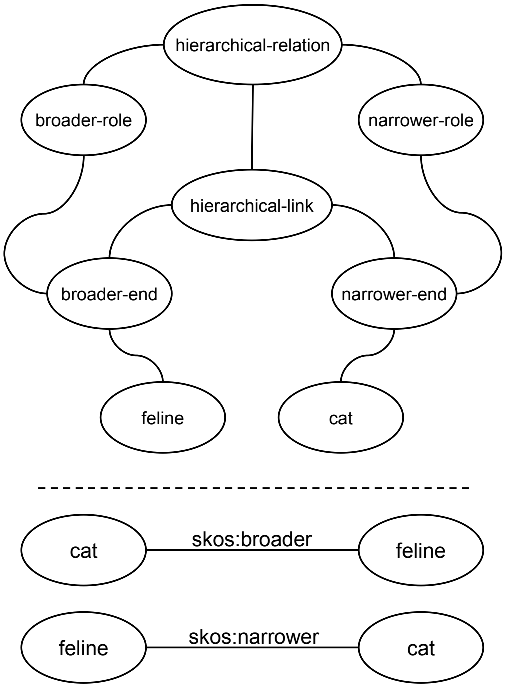


Figure 3. Top - broader and narrower relations represented via the 11179 metamodel. Bottom - broader and narrower relations represented in SKOS.

We are currently experimenting with applying this research to automated compliance challenges. The 11179 metamodel is useful for registering the metadata related to system policies and rules. We can then track changes to these rules, and relationships between different rules, in the same way we track changes and relationships in country code data. The Constellation registry, using the 11179 metamodel, can thus be used to address these challenges across a variety of metadata.

References

[1] "Operational Bulletin No. 991 Annex - List of ITU-T Recommendation E.164 assigned country codes." ITU-T Telecommunication Standardization Bureau, 01-Nov-2011.

[2] "Aircraft Nationality Marks and Common Marks as notified to ICAO." International Civil Aviation Organization - Air Navigation Bureau (ANB), 08-Jun-2009.

[3] "ISO 3166-1 - Codes for the representation of names of countries and their subdivisions -- Part 1: Country codes." International Organization for Standardization, 15-Nov-2006.

[4] T. Palmer, "Geopolitical Entities and Codes (Formerly Federal Information Processing Standards Publication 10-4: Countries, Dependencies, Areas of Special Sovereignty, and Their Principal Administrative Divisions)." National Geospatial-Intelligence Agency, Apr-2010.

[5] "In quite a state," *The Economist*, vol. 395, no. 8677, pp. 62–63, 10-Apr-2010.

[6] R. Gates and K. Keck, Eds., "ISO/IEC FDIS 11179-3:2012(E) - Information technology — Metadata registries (MDR) — Part 3: Registry metamodel and basic attributes." ISO/IEC JTC1, 08-Jan-2012. Available: http://metadata-standards.org/Document-library/Documents-by-number/WG2-N1651-N1700/WG2N1675_Editors-Final-Sneak-Peek-FDIS_11179-3.pdf

[7] "FAO Geopolitical ontology," *Food and Agriculture Organization of the United Nations*, 18-Jan-2011. [Online]. Available: http://www.fao.org/countryprofiles/geoinfo/geopolitical/resource/.

[8] A. Miles and S. Bechhofer, Eds., "SKOS Simple Knowledge Organization System Reference." The World Wide Web Consortium, 18-Aug-2009.

[9] J. Voss, "Encoding changing country codes for the Semantic Web with ISO 3166 and SKOS," in *Metadata and Semantics*, New York, NY: Springer Science+Business Media, LLC, 2009, pp. 211–221.

[10] A. Miles and S. Bechhofer, Eds., "SKOS Simple Knowledge Organization System eXtension for Labels (SKOS-XL) Namespace Document - HTML Variant." The World Wide Web Consortium, 18-Aug-2009.

[11] B. DuCharme, "Improve your taxonomy management using the W3C SKOS standard," *IBM developerWorks*, 10-May-2011. [Online]. Available: http://www.ibm.com/developerworks/xml/library/x-skostaxonomy/index.html. [Accessed: 24-May-2012].

[12] MODS/MADS Editorial Committee, Ed., "MADS/RDF Primer." Library of Congress, 10-May-2012.

[13] "MARC List for Geographic Areas," *Library of Congress Authorities and Vocabularies*, 26-Apr-2011. [Online]. Available: http://id.loc.gov/vocabulary/geographicAreas.html.

[14] G. Richardson, "Automated Country Name Disambiguation for Code Set Alignment," *Research and Advanced Technology for Digital Libraries*, vol. 6273, pp. 498–501, Sep. 2010.

[15] H. Halpin, P. Hayes, J. McCusker, D. McGuinness, and H. Thompson, "When owl:sameAs Isn't the Same: An Analysis of Identity in Linked Data," in The Semantic Web – ISWC 2010, vol. 6496, P. Patel-Schneider, Y. Pan, P. Hitzler, P. Mika, L. Zhang, J. Pan, I. Horrocks, and B. Glimm, Eds. Springer Berlin / Heidelberg, 2010, pp. 305–320

[16] H. Knublauch, Ed. "SPIN - SPARQL Syntax" The World Wide Web Consortium. [Online]. Available: http://www.w3.org/Submission/spin-sparql/

Rapid Argumentation Capture from Analysis Reports: The Case Study of Aum Shinrikyo

Mihai Boicu, Gheorghe Tecuci, Dorin Marcu
Learning Agents Center, Volgenau School of Engineering, George Mason University, Fairfax, VA 22030

Abstract— **The availability of subject matter experts has always been a challenge for the development of knowledge-based cognitive assistants incorporating their expertise. This paper presents an approach to rapidly develop cognitive assistants for evidence-based reasoning by capturing and operationalizing the expertise that was already documented in analysis reports. It illustrates the approach with the development of a cognitive assistant for assessing whether a terrorist organization is pursuing weapons of mass destruction, based on a report on the strategies followed by Aum Shinrikyo to develop and use biological and chemical weapons.**

Knowledge engineering, learning agent shell for evidence-based reasoning, problem reduction and solution synthesis, agent teaching and learning, intelligence analysis, cognitive assitant, argumentation, weapons of mass destruction.

I. INTRODUCTION

We research advanced knowledge engineering methods for rapid development of agents that incorporate the knowledge of human experts to assist their users in complex problem solving and to teach students. The development of such systems by knowledge engineers and subject matter experts is very complex due to the difficulty of capturing and representing experts' problem solving knowledge.

Our approach to this challenge was to develop multistrategy learning methods enabling a subject matter expert who is not a knowledge engineer to train a learning agent through problem solving examples and explanations, in a way that is similar to how the expert would train a student. This has led to the development of a new type of tool for agent development which we have called *learning agent shell* [1]. The learning agent shell is a refinement of the concept of *expert system shell* [2]. As an expert system shell, the learning agent shell includes a general inference engine for a knowledge base to be developed by capturing knowledge from a subject matter expert. The inference engine of the learning agent shell, however, is based on a general divide-and-conquer approach to problem solving, called problem reduction and solution synthesis, which is very natural for a non-technical subject matter expert, facilitates agent teaching and learning, and is computationally efficient. Moreover, in order to facilitate knowledge reuse, the knowledge base of the learning agent shell is structured into an ontology of concepts and a set of problem solving rules expressed with these concepts. The ontology is the more general part of the knowledge base and is usually relevant to many applications in the same domain, such as military or medicine. Indeed, many military applications will require reasoning with concepts such as military unit or military equipment. Thus, when developing a knowledge-based agent for a new military application, one may expect to be able to reuse a significant part of the ontology of a previously developed agent. The reasoning rules, however, are much more application-specific, such as the rules for critiquing a course of action with respect to the principles of war versus the rules for determining the strategic center of gravity of a force. Therefore the rules are reused to a much lesser extent. To facilitate their acquisition, the learning agent shell includes a multistrategy learning engine, enabling the learning of the rules directly from the subject matter expert, as mentioned above.

We have developed increasingly more capable and easier to use learning agent shells and we have applied them to build knowledge-based agents for various applications, including military engineering planning, course of action critiquing, and center of gravity determination [3].

Investigating the development of cognitive assistants for intelligence analysis, such as Disciple LTA [4] and TIACRITIS [5], has led us to the development of a new type of agent development tool, called *learning agent shell for evidence-based reasoning* [6]. This new tool extends a learning agent shell with generic modules for representation, search, and reasoning with evidence. It also includes a hierarchy of knowledge bases, the top of which is a domain-independent knowledge base for evidence-based reasoning containing an ontology of evidence and general rules, such as the rules for assessing the believability of different items of evidence [7]. This knowledge base is very significant because it is applicable to evidence-based reasoning tasks across various domains, such as intelligence analysis, law, forensics, medicine, physics, history, and others. An example of a *learning agent shell for evidence-based reasoning* is Disciple-EBR [6].

The development of a knowledge-based agent for an evidence-based reasoning task, such as intelligence analysis, is simplified because the shell already has general knowledge for evidence-based reasoning. Thus one only needs to develop the domain-specific part of the knowledge base. However, we still face the difficult problem of having access to subject matter experts who can dedicate their time to teach the agent. This paper presents a solution to this problem. It happens that there are many reports written by subject matter experts which already contain significant problem solving expertise. Thus, rather than eliciting the expertise directly from these experts, a junior professional may capture it from their reports.

We will illustrate this approach by considering a recent

This research was partially supported by the National Geospatial-Intelligence Agency and by George Mason University. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the National Geospatial-Intelligence Agency or the U.S. Government.

report from the Center for a New American Security, “*Aum Shinrikyo: Insights Into How Terrorists Develop Biological and Chemical Weapons”* [8]. This report provides a comprehensive analysis of this terrorist group, its radicalization, and the strategies followed in the development and use of biological and chemical weapons. As stated by its authors: “… this is the most accessible and informative opportunity to study terrorist efforts to develop biological and chemical weapons” [8, p.33]. “This detailed case study of Aum Shinrikyo (Aum) suggests several lessons for understanding attempts by other terrorist groups to acquire chemical or biological weapons” [8, p.4]. “Our aim is to have this study enrich policymakers’ and intelligence agencies’ understanding when they assess the risks that terrorists may develop and use weapons of mass destruction” [8, p.6].

Indeed, this report presents in detail two examples of how a terrorist group has pursued weapons of mass destruction, one where it was successful (sarin-based chemical weapons), and one where it was not successful (B-anthracis-based biological weapons). We will show how we can use these examples to train Disciple-EBR, evolving it into a cognitive assistant that will help intelligence analysts in assessing whether other terrorist groups may be pursuing weapons of mass destruction. Notice that this process operationalizes the knowledge from the report to facilitate its application in new situations.

We first present a brief summary of the Aum report. Then we explain the process of evidence-based hypothesis analysis using problem reduction and solution synthesis. Finally we present the actual development of the cognitive assistant.

II. Aum Shinrikyo: Insights Into How Terrorists Develop Biological and Chemical Weapons [8]

The first section of the report describes the creation of the Aum cult by Chizuo Matsumoto in 1984 as a yoga school. Soon after that Aum started to develop a religious doctrine and to create monastic communities. From the beginning the cult was apocalyptic, believing in an imminent catastrophe that can be prevented only by positive spiritual action. In 1988, the cult started to apply physical force and punishments toward its members to purify the body, and started to commit illegalities.

The second section of the report analyzes the biological weapons program. The cult first tried to obtain botulinum toxin, but it failed to obtain a deadly strain. However, the cult released the toxin in 20 to 40 attacks in which, luckily, nobody died. Possible causes of the failure were identified as ineffective initial strain of C. botulinum, unsuitable culture conditions, unsterile conditions, wrong post-fermentation recovery, and improper storage conditions. Similarly, the anthrax program and its failure are analyzed.

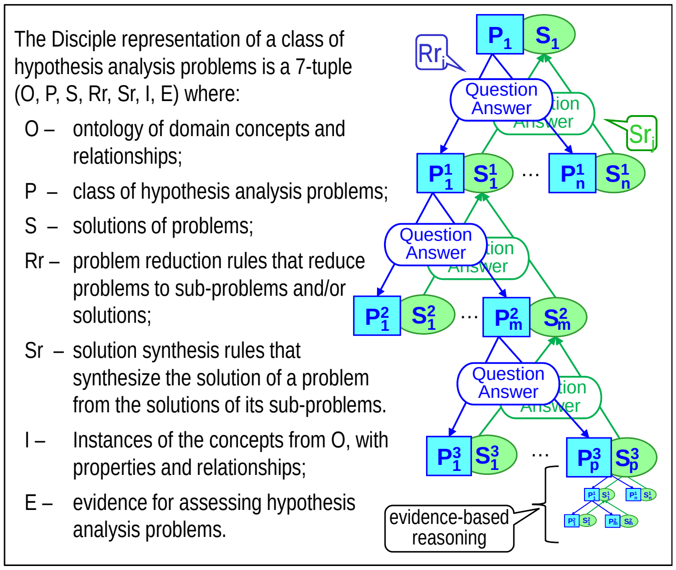
The third section of the report analyzes the chemical weapons program. While other chemical agents were tested during the program, the main part of the program was based on sarin. Although the program had some problems with mass production, it was generally successful, and produced large quantities of sarin at various levels of purity. Aum performed several attacks with sarin, including: (1) an ineffective attack on a competing religious leader in 1993; (2) an attack, in June 1994, with a vaporization of sarin, intended to kill several judges – the vapors were shifted toward a neighborhood, killing 8 persons and injuring 200; (3) several attacks in the Tokyo Subway on 20 March 1995, killing 13 and injuring thousands.

The fourth section of the report summarizes the main lessons learned: (1) chemical weapons capabilities seem more accessible than biological capabilities for mass killing; (2) effective dissemination is challenging; (3) recurred accidents in the programs did not deter their pursuit; (4) during the transition to violence some leaders joined while others were isolated or killed; (5) law enforcement pressure was highly disruptive even though it was not an effective deterrent; (6) the programs and attacks were conducted by the leadership group only, to maintain secrecy; (7) the hierarchical structure of the cult facilitated the initiation and resourcing of the programs but distorted their development and assessment; (8) contemporaneous assessment of the intentions and capabilities of a terrorist organization are difficult, uncertain and even misleading; (9) despite many mistakes and failures, successes were obtained as a result of the persistence in the programs.

III. Hypothesis Analysis with Disciple-EBR

A class of hypothesis analysis problems is represented in Disciple-EBR as the 7-tuple (O, P, S, Rr, Sr, I, E), as shown in Figure 1. The ontology O is a hierarchical representation of both general and domain-specific concepts and relationships. The general (domain-independent) concepts are primarily those for evidence-based reasoning, such as different types of evidence. The two primary roles of the ontology are to support the representation of the other knowledge elements (e.g. the reasoning rules), and to serve as the generalization hierarchy for learning. The hypothesis analysis problems P and the corresponding solutions S are natural language patterns with variables. They include first-order logic applicability conditions that restrict the possible values of the variables.

A problem reduction rule Rr expresses how and under what conditions a generic hypothesis analysis problem P_g can be reduced to simpler generic problems. These conditions are represented as first-order logical expressions. Similarly, a



The Disciple representation of a class of hypothesis analysis problems is a 7-tuple (O, P, S, Rr, Sr, I, E) where:

O – ontology of domain concepts and relationships;

P – class of hypothesis analysis problems;

S – solutions of problems;

Rr – problem reduction rules that reduce problems to sub-problems and/or solutions;

Sr – solution synthesis rules that synthesize the solution of a problem from the solutions of its sub-problems.

I – Instances of the concepts from O, with properties and relationships;

E – evidence for assessing hypothesis analysis problems.

Figure 1. Disciple representation of a class of hypothesis analysis problems.

solution synthesis rule Sr expresses how and under what conditions generic probabilistic solutions can be combined into another probabilistic solution [9]. As mentioned, Disciple-EBR already contains domain-independent problem reduction and solution synthesis rules for evidence-based reasoning.

Disciple-EBR employs a general divide-and-conquer approach to solve a hypothesis analysis problem. For example, as illustrated in the right-hand side of Figure 1, a complex problem P_1 is reduced to n simpler problems $P^1_1, \ldots, P^1_n$, through the application of the reduction rule Rr_i. If we can then find the solutions $S^1_1, \ldots, S^1_n$ of these sub-problems, then these solutions can be combined into the solution S_1 of the problem P_1, through the application of the synthesis rule Sr_j. The Question/Answer pairs associated with these reduction and synthesis operations express, in natural language, the applicability conditions of the corresponding reduction and synthesis rules, in this particular situation. Their role will be discussed in more detail in the next section.

Specific examples of reasoning trees are shown in Figures 5, 6, and 11, which will be discussed in the next section. In general, a top-level hypothesis analysis problem is successively reduced (guided by questions and answers) to simpler and simpler problems, down to the level of elementary problems that are solved based on knowledge and evidence. Then the obtained solutions are successively combined, from bottom-up, to obtain the solution of the top-level problem.

Figure 2 presents the reduction and synthesis operations in more detail. To assess hypothesis H_1 one asks the question Q which happens to have two answers, A and B. For example, a question like "Which is an indicator for H_1?" may have many answers, while other questions have only one answer. Let's assume that answer A leads to the reduction of H_1 to the simpler hypotheses H_2 and H_3, and answer B leads to the reduction of H_1 to H_4 and H_5. Let us further assume that we have assessed the likeliness of each of these four sub-hypotheses, as indicated at the bottom part of Figure 2. The likeliness of H_2 needs to be combined with the likeliness of H_3, to obtain a partial assessment (corresponding to the answer A) of the likeliness of H_1. One similarly obtains another partial assessment (corresponding to the answer B) of the likeliness of H_1. Then the likeliness of H_1 corresponding to the answer A needs to be combined with the likeliness of H_1 corresponding to the answer B, to obtain the likeliness of H_1 corresponding to all the answers of question Q (e.g., corresponding to all the indicators).

We call the two bottom-level syntheses in Figure 2 *reduction-level syntheses* because they correspond to reductions of H_1 to simpler hypotheses. We call the top-level synthesis *problem-level synthesis* because it corresponds to all the known strategies for solving the problem.

The likeliness may be expressed using symbolic probability values that are similar to those used in the U.S. National Intelligence Council's standard estimative language: {no possibility, a remote possibility, very unlikely, unlikely, an even chance, likely, very likely, almost certain, certain}. However, other symbolic probabilities may also be used, as discussed by Kent [10] and Weiss [11]. In these cases one may use simple synthesis functions, such as, min, max, average, or weighted sum, as shown in Figure 2 and Figure 3 [12].

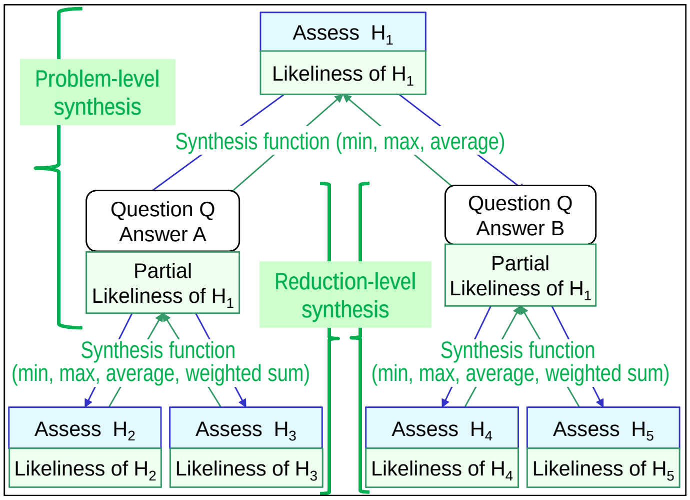

Assess H_1
Likeliness of H_1
Problem-level synthesis
Synthesis function (min, max, average)
Question Q
Answer A
Partial Likeliness of H_1
Reduction-level synthesis
Question Q
Answer B
Partial Likeliness of H_1
Synthesis function (min, max, average, weighted sum)
Synthesis function (min, max, average, weighted sum)
Assess H_2
Likeliness of H_2
Assess H_3
Likeliness of H_3
Assess H_4
Likeliness of H_4
Assess H_5
Likeliness of H_5

Figure 2. Hypothesis assessment through reduction and synthesis.

As indicated above, Disciple-EBR includes general reduction and synthesis rules for evidence-based reasoning which allow it to automatically generate fragments of the reduction and synthesis tree, like the one from Figure 3. In this case the problem is to assess hypothesis H_1 based on favoring evidence. Which is a favoring item of evidence? If E_1 is such an item, then Disciple reduces the top level assessment to two simpler assessments: "Assess the relevance of E_1 to H_1" and "Assess the believability of E_1". If E_2 is another relevant item of evidence, then Disciple reduces the top level assessment to two other simpler assessments. Obviously there may be any number of favoring items of evidence.

Now let us assume that Disciple has obtained the solutions of the leaf problems, as shown at the bottom of Figure 3 (e.g., "If we assume that E_1 is believable, then H_1 is very likely to be true." "The believability of E_1 is likely.") Notice that what is really of interest in a solution is the actual likeliness value. Therefore, an expression like "The believability of E_1 is likely" can be abstracted to "likely." Consequently, the reasoning tree in Figure 3 shows only these abstracted solutions, although, internally, the complete solution expressions are maintained.

Having obtained the solutions of the leaf hypotheses in Figure 3, Disciple automatically combines them to obtain the likeliness of the top level hypothesis. First it assesses the

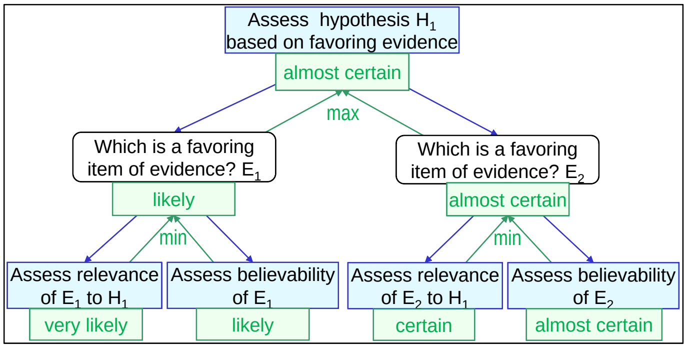

Assess hypothesis H_1 based on favoring evidence
almost certain
max
Which is a favoring item of evidence? E_1
likely
min
Which is a favoring item of evidence? E_2
almost certain
min
Assess relevance of E_1 to H_1
very likely
Assess believability of E_1
likely
Assess relevance of E_2 to H_1
certain
Assess believability of E_2
almost certain

Figure 3. Automated hypothesis assessment through reduction and synthesis.

inferential force of each item of favoring evidence (i.e., E$_1$ and E$_2$) on H$_1$ by taking the min between its *relevance* and its *believability*, because only evidence that is both relevant and believable will convince us that a hypothesis is true. Next Disciple assesses the inferential force of the favoring evidence as the max of the inferential force corresponding to individual items of evidence because it is enough to have one relevant and believable item of evidence to convince us that the hypothesis H$_1$ is true. Disciple will similarly consider disfavoring items of evidence, and will use an on balance judgment to determine the inferential force of all available evidence on H$_1$.

To facilitate the browsing and understanding of larger reasoning trees, Disciple also displays them in abstracted (simplified) form, as illustrated in the bottom right side of Figure 5. The top-level abstract problem "start with chaos and destruction" is the abstraction of the problem "Assess whether Aum Shinrikyo preaches that the apocalypse will start with chaos and destruction" from the bottom of Figure 5. The abstract sub-problem "favoring evidence" is the abstraction of "Assess whether Aum Shinrikyo preaches that the apocalypse will start with chaos and destruction, based on favoring evidence." This is a specific instance of the problem from the top of Figure 3 which is solved as discussed above. The user assessed the relevance and the believability of the two items of evidence EVD-013 and EVD-014, and Disciple automatically determined and combined their inferential force on the higher-level hypotheses.

IV. Agent Development Methodology

Figure 4 presents the main stages of evolving the Disciple-EBR agent shell into a specific cognitive assistant for hypotheses analysis. The first stage is system specification during which a knowledge engineer and a subject matter expert define the types of problems to be solved by the system. Then they rapidly develop a prototype, first by developing a model of how to solve a problem, and then by applying the model to solve typical problems. During the next phase they use the developed sample reasoning trees to develop a specification of the system's ontology and use that specification to design and develop an ontology of concepts and relationships which is as complete as possible. Finally they use the system to learn and refine reasoning rules, which may also require the extension of the ontology.

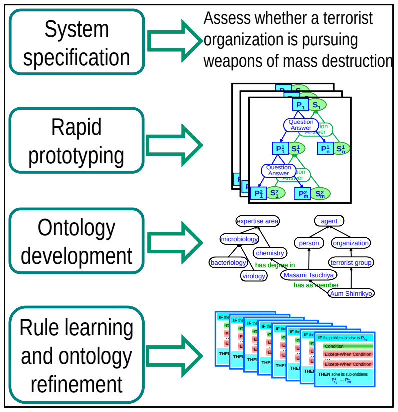
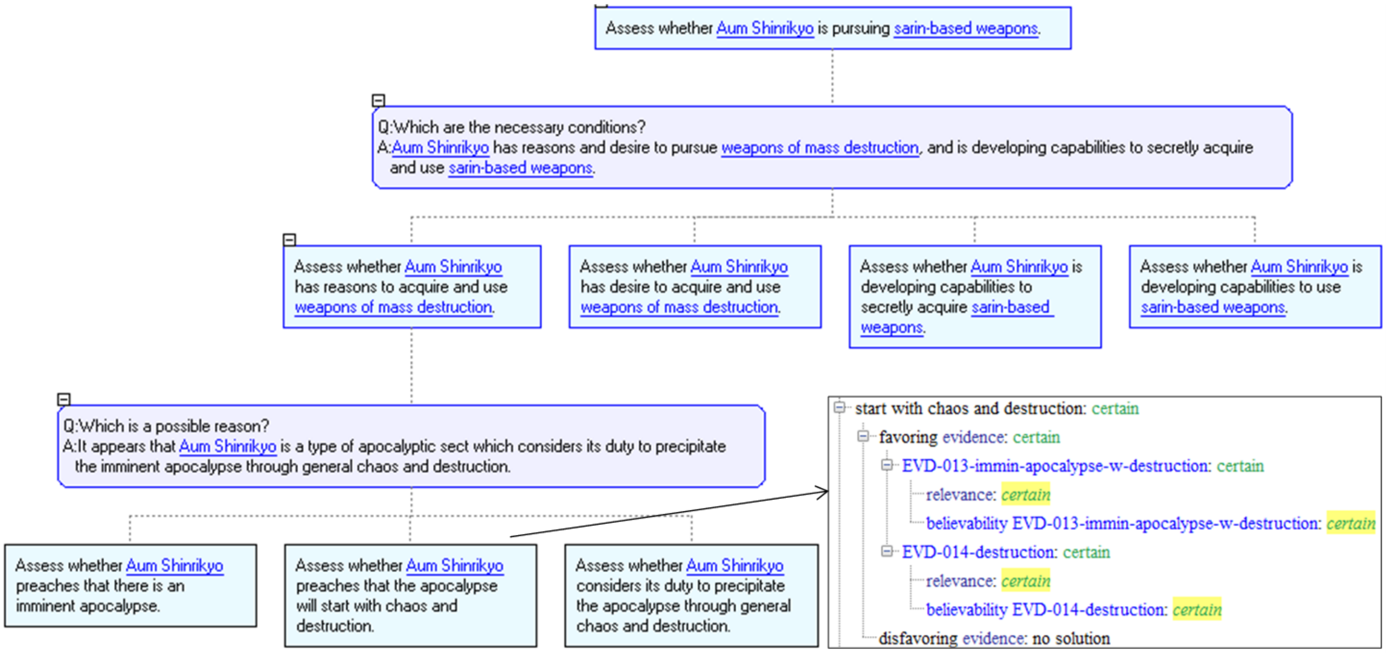


Figure 4. Main agent development stages.

In the next section we will illustrate the development of a cognitive assistant that will help assess whether a terrorist organization is pursuing weapons of mass destruction. The main difference from the above methodology is that we capture the expertise not from a subject matter expert, but from the Aum report [8].

V. Capturing the Expertise from the Aum Report

The Aum report presents in detail two examples of how a terrorist group has pursued weapons of mass destruction. We will briefly illustrate the process of teaching Disciple-EBR based on these examples, enabling it to assist other analysts in assessing whether a terrorist group may be pursuing weapons of mass destruction. For this, we need to frame each of these examples as a problem solving experience imagining, for instance, that we are attempting to solve the following hypothesis analysis problem:



Assess whether Aum Shinrikyo is pursuing sarin-based weapons.

Q:Which are the necessary conditions?
A:Aum Shinrikyo has reasons and desire to pursue weapons of mass destruction, and is developing capabilities to secretly acquire and use sarin-based weapons.

Assess whether Aum Shinrikyo has reasons to acquire and use weapons of mass destruction.

Assess whether Aum Shinrikyo has desire to acquire and use weapons of mass destruction.

Assess whether Aum Shinrikyo is developing capabilities to secretly acquire sarin-based weapons.

Assess whether Aum Shinrikyo is developing capabilities to use sarin-based weapons.

Q:Which is a possible reason?
A:It appears that Aum Shinrikyo is a type of apocalyptic sect which considers its duty to precipitate the imminent apocalypse through general chaos and destruction.

Assess whether Aum Shinrikyo preaches that there is an imminent apocalypse.

Assess whether Aum Shinrikyo preaches that the apocalypse will start with chaos and destruction.

Assess whether Aum Shinrikyo considers its duty to precipitate the apocalypse through general chaos and destruction.

start with chaos and destruction: certain
favoring evidence: certain
EVD-013-immin-apocalypse-w-destruction: certain
relevance: certain
believability EVD-013-immin-apocalypse-w-destruction: certain
EVD-014-destruction: certain
relevance: certain
believability EVD-014-destruction: certain
disfavoring evidence: no solution

Figure 5. Detailed and abstract fragments of the hypothesis analysis tree.

Assess whether Aum Shinrikyo is pursuing sarin-based weapons.

We express the problem in natural language and select the phrases that may be different for other problems. The selected phrases will appear in blue, guiding the system to learn a general problem pattern:

Assess whether ?O1 is pursuing ?O2.

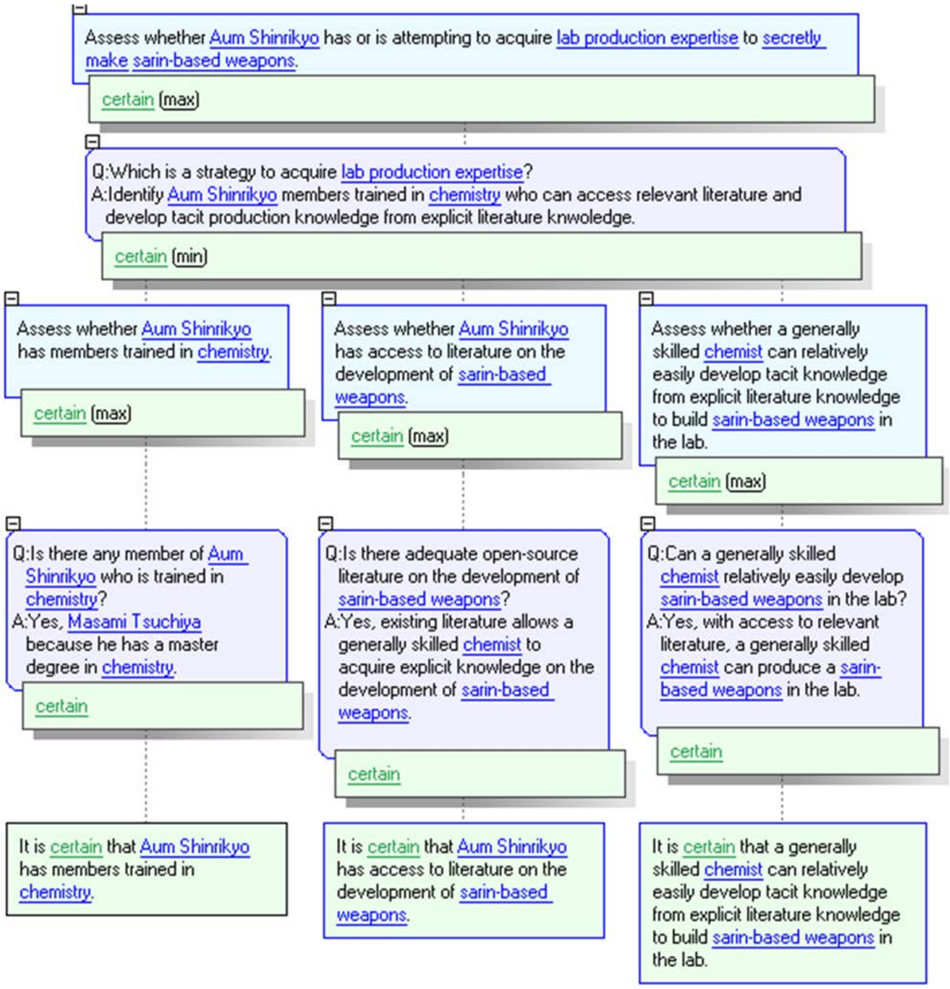
Then we show Disciple how to solve the hypothesis analysis problem based on the knowledge and evidence provided in the Aum report. The modeling module of Disciple-EBR guides us in developing a reasoning tree like the one from the right hand side of Figure 1. The top part of this tree is shown in Figure 5.

The main goal of this stage is to develop a formal, yet intuitive argumentation structure [12-15], representing the assessment logic as inquiry-driven problem reduction and solution synthesis. Notice that, guided by a question-answer pair, we reduce the top-level hypothesis assessment problem to four sub-problems. We then reduce the first sub-problem to three simpler problems which we declare as elementary hypotheses, to be assessed based on evidence. Once we associate items of evidence from the Aum report with such an elementary hypothesis, Disciple automatically develops a reduction tree. For example, we have associated two items of favoring evidence with the second leaf-problem and Disciple has generated the reasoning tree whose abstraction is shown in the bottom-right of Figure 5. After we have assessed the relevance and the believability of each item, Disciple has automatically computed the inferential force and the likeliness of the upper level hypotheses, concluding: “It is certain that Aum Shinrykio preaches that the apocalypse will start with chaos and destruction.”

The other hypothesis analysis problems are reduced in a similar way, either to elementary hypotheses assessed based on evidence, or directly to solutions. For example, based on the information from the Aum report, the problem “Assess whether Aum Shinrykio is developing capabilities to secretly acquire sarin-based weapons” is reduced to the problems of assessing whether Aum Shinrykio has or is attempting to acquire expertise, significant funds, production material, and covered mass production facilities, respectively. Further, the problem “Assess whether Aum Shinrykio has or is attempting to acquire expertise in order to secretly make sarin-based weapons” is reduced to the problems of assessing whether it has or is attempting to acquire lab production expertise, mass production expertise, and weapons assessment expertise, respectively. Then the problem “Assess whether Aum Shinrykio has or is attempting to acquire lab production expertise in order to secretly make sarin-based weapons” is solved as indicated in Figure 6. As one can see, the strategy employed by Aum Shinrykio was to identify members trained in chemistry who can access relevant literature and develop tacit production knowledge from explicit literature knowledge. This strategy was successful. A member of Aum Shinrykio was Masami Tsuchiya who had a master degree in chemistry. Moreover, there is open-source literature from which a generally-skilled chemist can acquire explicit knowledge on the development of sarin-based weapons. From it, the chemist can relatively easily develop tacit knowledge to produce sarin-based weapons in the lab.

The Aum report provides the knowledge and evidence to solve the initial problem, explaining the success of Aum Shinrykio in pursuing sarin-based weapons.

At this stage Disciple only uses a form of non-disruptive learning from the user, automatically acquiring reduction and synthesis patterns corresponding to the specific reduction and synthesis steps from the developed reasoning tree. These patterns are not automatically applied in problem solving because they would have too many instantiations, but they are suggested to the user who can use them when solving a similar problem which, in this case, is “Assess whether Aum Shinrikyo is



Assess whether Aum Shinrikyo has or is attempting to acquire lab production expertise to secretly make sarin-based weapons.

certain (max)

Q:Which is a strategy to acquire lab production expertise?
A:Identify Aum Shinrikyo members trained in chemistry who can access relevant literature and develop tacit production knowledge from explicit literature knwoledge.

certain (min)

Assess whether Aum Shinrikyo has members trained in chemistry.

certain (max)

Q:Is there any member of Aum Shinrikyo who is trained in chemistry?
A:Yes, Masami Tsuchiya because he has a master degree in chemistry.

certain

It is certain that Aum Shinrikyo has members trained in chemistry.

Assess whether Aum Shinrikyo has access to literature on the development of sarin-based weapons.

certain (max)

Q:Is there adequate open-source literature on the development of sarin-based weapons?
A:Yes, existing literature allows a generally skilled chemist to acquire explicit knowledge on the development of sarin-based weapons.

certain

It is certain that Aum Shinrikyo has access to literature on the development of sarin-based weapons.

Assess whether a generally skilled chemist can relatively easily develop tacit knowledge from explicit literature knowledge to build sarin-based weapons in the lab.

certain (max)

Q:Can a generally skilled chemist relatively easily develop sarin-based weapons in the lab?
A:Yes, with access to relevant literature, a generally skilled chemist can produce a sarin-based weapons in the lab.

certain

It is certain that a generally skilled chemist can relatively easily develop tacit knowledge from explicit literature knowledge to build sarin-based weapons in the lab.

Figure 6. Sample problem reduction and solution synthesis tree.

pursuing B-anthracis-based weapons". The overall approach used by Aum Shinrykio was the same but, in this case, the group was not successful because of several key differences. For example, Endo, the person in charge of the biological weapons was not an appropriate expert: "Endo's training, interrupted by his joining Aum, was as a virologist not as a bacteriologist, while in Aum's weapons program he worked with bacteria" [8, p.33]. While there is open-source literature from which a generally-skilled microbiologist can acquire explicit knowledge on the development of B-anthracis-based weapons, "producing biological materials is a modern craft or an art analogous to playing a sport or speaking a language. Though some aspects can be mastered just from reading a book, others relevant to a weapons program cannot be acquired this way with rapidity or assurance" [8, p.33].

The rapid prototyping stage (see Figure 4) results in a system that can be subjected to an initial validation with the end-users.

The next stage is that of ontology development. The guiding question is: What are the domain concepts, relationships and instances that would enable the agent to automatically generate the reasoning trees developed during rapid prototyping?

The questions and answers that guide the reasoning process not only make very clear the logic of the subject matter expert, but they also drive the ontology development process, as will be briefly illustrated in the following.

From each reasoning step of the developed reasoning trees, the knowledge engineer identifies the instances, concepts and relationships mentioned in them, particularly those in the question/answer pair which provides the justification of that step. Consider, for example, the reduction from the bottom-left of Figure 6, guided by the following question/answer pair:

Q: Is there any member of Aum Shinrikyo who is trained in chemistry?
A: Yes, Masami Tsuchiya because he has a master degree in chemistry.

This suggests that the knowledge base of the agent should include the objects and the relationships shown in Figure 7. Such semantic network fragments represent a specification of the needed ontology. In particular, this fragment suggests the need for a hierarchy of agents (covering Aum Shinrikio and Masami Tsuchiya), and for a hierarchy of expertise domains for weapons of mass destruction (including chemistry). The first hierarchy might include concepts such as organization, terrorist group, person, and terrorist, while the second might include expertise domain, virology, bacteriology, microbiology, and nuclear physics. The semantic network fragment from Figure 7 also suggests defining two features, has as member (with organization as domain and person as range), and has master degree in (with person as domain and expertise area as range).

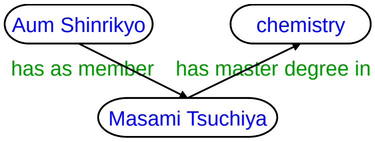


Figure 7. Ontology specification.

Based on such specifications, and using the ontology development tools of Disciple-EBR, the knowledge engineer develops an ontology that is as complete as possible by importing concepts and relationships from previously developed ontologies (including those on the semantic web), and from the Aum report.

The next stage in agent development is that of rule learning and ontology refinement. First one helps the agent to learn applicability conditions for the patterns learned during the rapid prototyping stage, thus transforming them into reasoning rules that will be automatically applied for hypotheses analysis.

From each problem reduction step of a reasoning tree developed during rapid prototyping the agent will learn a general problem reduction rule (or will refine it, if the rule was learned from a previous step), as presented elsewhere (e.g., [3, 9, 16]), and illustrated in Figure 8.

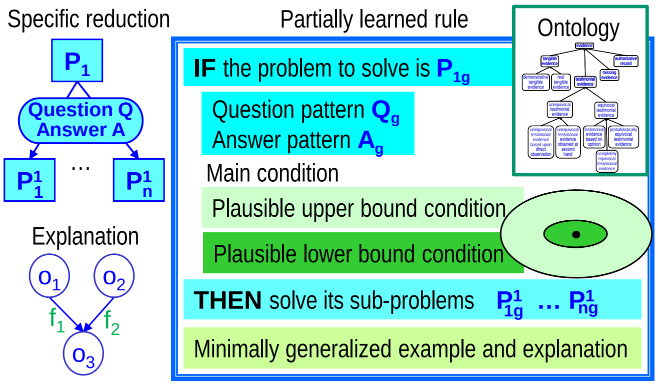

Specific reduction
P_1
Question Q
Answer A
P^1_1
…
P^1_n
Explanation
o_1
o_2
f_1
f_2
o_3
Partially learned rule
Ontology
IF the problem to solve is P_{1g}
Question pattern Q_g
Answer pattern A_g
Main condition
Plausible upper bound condition
Plausible lower bound condition
THEN solve its sub-problems P^1_{1g} … P^1_{ng}
Minimally generalized example and explanation

Figure 8. Rule learning from a specific reduction.

The left part of Figure 8 shows a specific problem reduction step and a semantic network fragment which represents the meaning of the question/answer pair expressed in terms of the agent's ontology. This network fragment corresponds to that defined by the knowledge engineer for this particular step, during the rapid prototyping phase, as illustrated in Figure 7. Recall that the question/answer pair is the justification of the reduction step. Therefore we refer to the corresponding semantic network fragment as the explanation of the reduction step.

The right hand side of Figure 8 shows the learned IF-THEN rule with a plausible version space applicability condition. The rule pattern is obtained by replacing each instance and constant in the reduction step with a variable. The lower bound of the applicability condition is obtained through a minimal generalization of the semantic network fragment, using the entire agent ontology as a generalization hierarchy. The upper bound is obtained through a maximal generalization.

One, however, only interacts with the agent to identify the explanation of the reduction step, based on suggestions made by the agent. Then the agent automatically generates the rule. For instance, based on the reduction from the left-hand side of Figure 6, and its explanation from Figure 7, Disciple learned the rule from Figure 9.

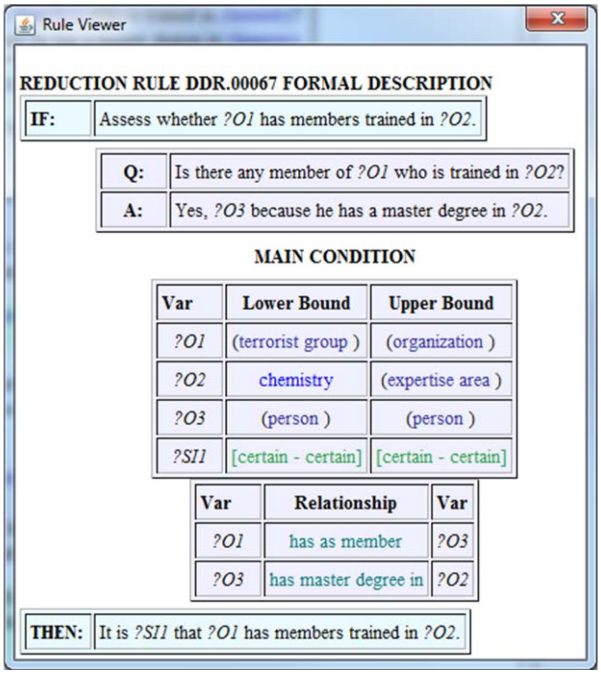
Finally one teaches the agent to solve other problems. In this case, however, the agent automatically generates parts of the reasoning tree, by applying the learned rules, and one critiques its reasoning, implicitly guiding the agent in refining the rules. For example, based on the explanation of why an instance of the rule in Figure 8 is wrong, the agent learns an except-when plausible version space condition which is added to the rule, as shown in Figure 10. Such conditions should not be satisfied in order to apply the rule.

Correct reductions lead to the generalization of the rule, either by generalizing the lower bound of the main condition, or by specializing the upper bound of one or several except-when conditions, or by adding a positive exception when none of the above operations is possible.

Incorrect reductions and their explanations lead to the specialization of the rule, either by specializing the upper bound of the main condition, or by generalizing the lower bound of an except-when condition, or by learning the plausible version space for a new except-when condition, or by adding a negative exception.

The goal is to improve the applicability condition of the rule so that it only generates correct reductions.

At the same time with learning new rules and refining previously learned rules, the agent may also extend the ontology. For example, to explain to the agent why a generated reduction is wrong, one may use a new concept or feature. As a result, the agent will add the new concept or feature in its ontology of concepts and features. This, however, requires an adaptation of the previously learned rules since the generalization hierarchies used to learn them have changed. To cope with this issue, the agent keeps minimal generalizations of the examples and the explanations from which each rule was learned, and uses this information to automatically regenerate the rules in the context of the new ontology. Notice that this is, in fact, a form of learning with an evolving representation language.



REDUCTION RULE DDR.00067 FORMAL DESCRIPTION

IF:	Assess whether ?O1 has members trained in ?O2.

Q:	Is there any member of ?O1 who is trained in ?O2?
A:	Yes, ?O3 because he has a master degree in ?O2.

MAIN CONDITION

Var	Lower Bound	Upper Bound
?O1	(terrorist group)	(organization)
?O2	chemistry	(expertise area)
?O3	(person)	(person)
?SI1	[certain - certain]	[certain - certain]

Var	Relationship	Var
?O1	has as member	?O3
?O3	has master degree in	?O2

THEN:	It is ?SI1 that ?O1 has members trained in ?O2.

Figure 9. Learned rule.

The trained agent may now assist an analyst in assessing whether other terrorist groups may be pursuing weapons of mass destruction. For instance, there may be some evidence that a new terrorist group, the Roqoppi brigade, may be pursuing botulinum-based biological weapons. The analyst may instantiate the pattern "Assess whether ?O1 is pursuing ?O2" with the name of the terrorist group and the weapon and the agent will generate the hypothesis analysis tree partially shown in Figure 11, helping the analyst in assessing this hypothesis based on the knowledge learned from the Aum report.

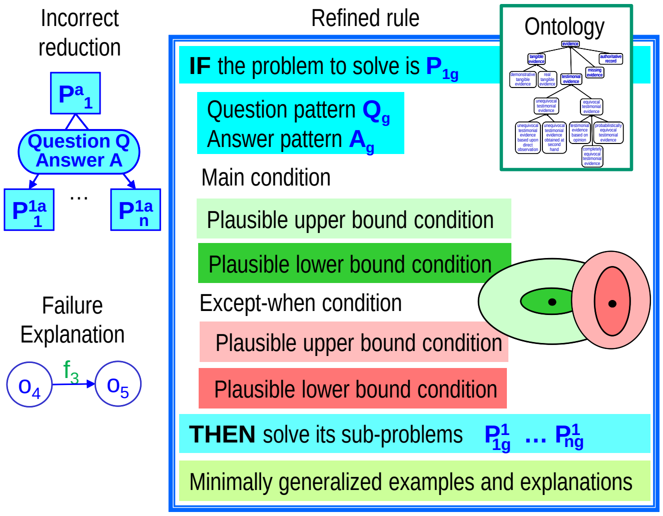

Incorrect reduction
P^a_1
Question Q
Answer A
P^{1a}_1
…
P^{1a}_n
Failure Explanation
o_4
f_3
o_5
Refined rule
Ontology
IF the problem to solve is P_{1g}
Question pattern Q_g
Answer pattern A_g
Main condition
Plausible upper bound condition
Plausible lower bound condition
Except-when condition
Plausible upper bound condition
Plausible lower bound condition
THEN solve its sub-problems P^1_{1g} … P^1_{ng}
Minimally generalized examples and explanations

Figure 10. Rule refined based on a negative example and its explanation.

VI. Final Remarks

We have briefly presented an approach to the rapid development of cognitive assistants for evidence-based reasoning by capturing and operationalizing the subject matter expertise from existing reports. This offers a cost-effective solution to disseminate and use valuable problem solving expertise which has already been described in lessons learned documents, after-action reports, or diagnostic reports.

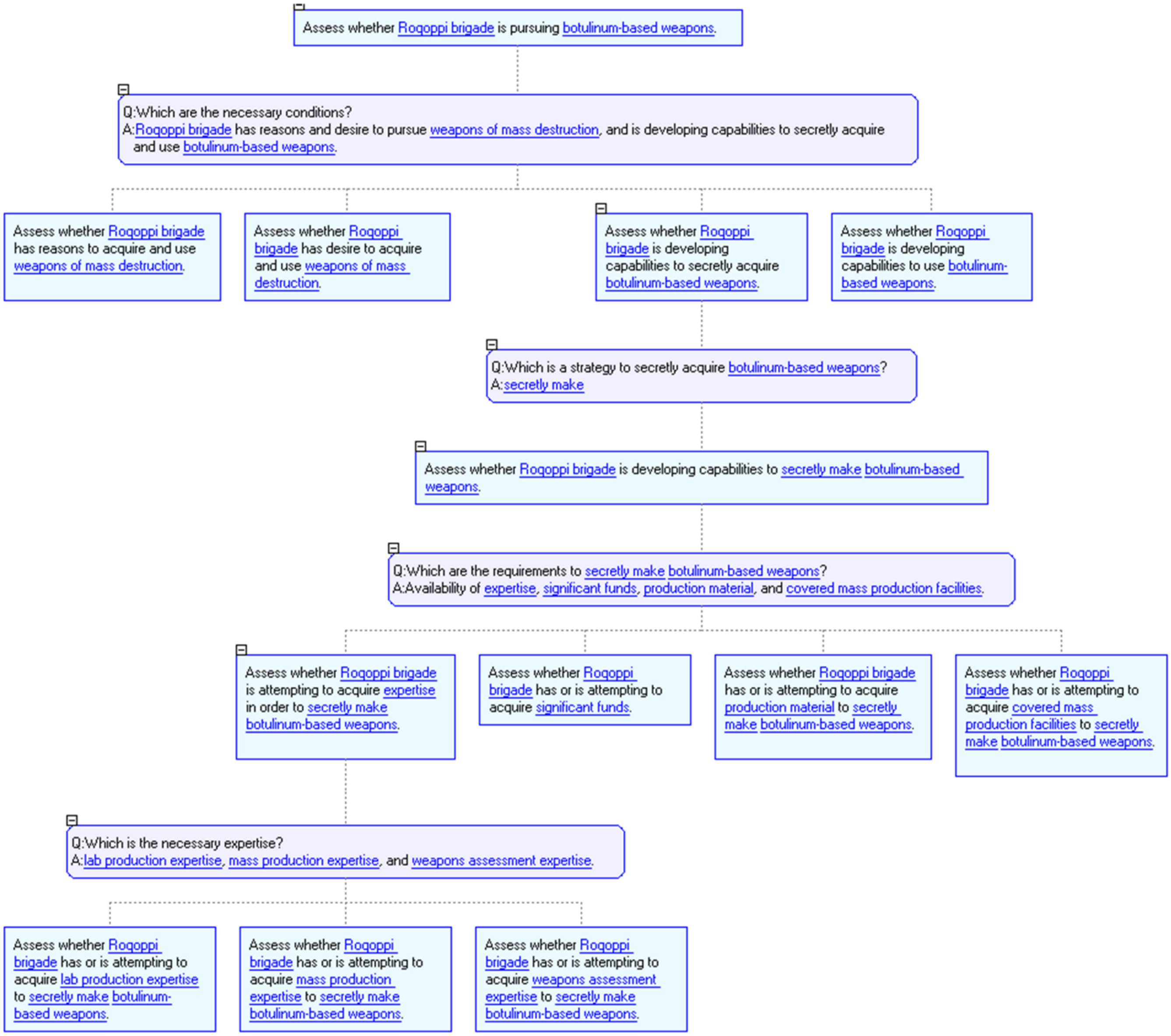
References

[1] Tecuci G. (1998). *Building Intelligent Agents: An Apprenticeship Multistrategy Learning Theory, Methodology, Tool and Case Studies*, San Diego: Academic Press, ISBN:0126851255.

[2] Clancey W.J. (1984). NEOMYCIN: Reconfiguring a rule-based system with application to teaching. In: Clancey, W.J., Shortliffe, E.H. (eds.) *Readings in Medical Artificial Intelligence*, pp.361-381. Reading, MA: Addison-Wesley.

[3] Boicu M., Tecuci G., Stanescu B., Marcu D. and Cascaval C.E. (2001). Automatic Knowledge Acquisition from Subject Matter Experts, in *Proceedings of the Thirteenth International Conference on Tools with Artificial Intelligence (ICTAI)*, pp. 69-78. 7-9 November 2001, Dallas, Texas. IEEE Computer Society, Los Alamitos, California.

[4] Boicu M., Tecuci G., Ayers C., Marcu D., Boicu C., Barbulescu M., Stanescu B., Wagner W., Le V., Apostolova D., Ciubotariu A. (2005). A Learning and Reasoning System for Intelligence Analysis, *Proceedings*



Assess whether Roqoppi brigade is pursuing botulinum-based weapons.

Q:Which are the necessary conditions?
A:Roqoppi brigade has reasons and desire to pursue weapons of mass destruction, and is developing capabilities to secretly acquire and use botulinum-based weapons.

Assess whether Roqoppi brigade has reasons to acquire and use weapons of mass destruction.

Assess whether Roqoppi brigade has desire to acquire and use weapons of mass destruction.

Assess whether Roqoppi brigade is developing capabilities to secretly acquire botulinum-based weapons.

Assess whether Roqoppi brigade is developing capabilities to use botulinum-based weapons.

Q:Which is a strategy to secretly acquire botulinum-based weapons?
A:secretly make

Assess whether Roqoppi brigade is developing capabilities to secretly make botulinum-based weapons.

Q:Which are the requirements to secretly make botulinum-based weapons?
A:Availability of expertise, significant funds, production material, and covered mass production facilities.

Assess whether Roqoppi brigade is attempting to acquire expertise in order to secretly make botulinum-based weapons.

Assess whether Roqoppi brigade has or is attempting to acquire significant funds.

Assess whether Roqoppi brigade has or is attempting to acquire production material to secretly make botulinum-based weapons.

Assess whether Roqoppi brigade has or is attempting to acquire covered mass production facilities to secretly make botulinum-based weapons.

Q:Which is the necessary expertise?
A:lab production expertise, mass production expertise, and weapons assessment expertise.

Assess whether Roqoppi brigade has or is attempting to acquire lab production expertise to secretly make botulinum-based weapons.

Assess whether Roqoppi brigade has or is attempting to acquire mass production expertise to secretly make botulinum-based weapons.

Assess whether Roqoppi brigade has or is attempting to acquire weapons assessment expertise to secretly make botulinum-based weapons.

Figure 11. Part of an automatically generated hypothesis analysis tree.

of the Twentieth National Conference on Artificial Intelligence, AAAI-05, Pittsburgh, Pennsylvania, USA, July 9-13.

[5] Tecuci, G., Marcu, D., Boicu, M., Schum, D.A., Russell K. (2011). Computational Theory and Cognitive Assistant for Intelligence Analysis, in *Proceedings of the Sixth International Conference on Semantic Technologies for Intelligence, Defense, and Security – STIDS*, pp. 68-75, Fairfax, VA, 16-18 November.

[6] Boicu, M., Marcu, D., Tecuci, G., Schum, D. (2011). Cognitive Assistants for Evidence-Based Reasoning Tasks, *AAAI Fall Symposium on Advances in Cognitive Systems*, Arlington, VA, 4-6 November.

[7] Boicu M., Tecuci G., Schum D. (2008). Intelligence Analysis Ontology for Cognitive Assistants, in *Proceedings of the Conference "Ontology for the Intelligence Community: Towards Effective Exploitation and Integration of Intelligence Resources,"* Fairfax, VA, 3-4 December.

[8] Danzig R., Sageman M., Leighton T., Hough L., Yuki H., Kotani R. and Hosford Z.M. (2011). *Aum Shinrikyo: Insights Into How Terrorists Develop Biological and Chemical Weapons*, Center for a New American Security, Washington, DC, July.

[9] Tecuci G., Boicu M. (2010). Agent Learning for Mixed-Initiative Knowledge Acquisition, *Final Report for the AFOSR Grant # FA9550-07-1-0268*, Learning Agents Center, Fairfax, VA 22030, February 28.

[10] Kent S. (1994). Words of Estimated Probability, in Steury D.P., ed., *Sherman Kent and the Board of National Estimates: Collected Essays*, Center for the Study of Intelligence, CIA, Washington, DC.

[11] Weiss C. (2008). Communicating Uncertainty in Intelligence and Other Professions, *International Journal of Intelligence and CounterIntelligence*, 21(1), 57–85.

[12] Schum D.A. (2001). *The Evidential Foundations of Probabilistic Reasoning*, Northwestern University Press.

[13] Tecuci G., Schum D.A., Boicu M., Marcu D. (2011). *Introduction to Intelligence Analysis: A Hands-on Approach with TIACRITIS*, 220 pages, George Mason University.

[14] Wigmore J.H. (1937). *The Science of Judicial Proof*. Boston, MA: Little, Brown & Co.

[15] Toulmin S.E. (1963). *The Uses of Argument*. Cambridge Univ. Press.

[16] Tecuci G., Boicu M., Boicu C., Marcu D., Stanescu B., Barbulescu M. (2005). The Disciple-RKF Learning and Reasoning Agent, *Computational Intelligence, Vol.21, No.4*, pp. 462-479.

Developing an Ontology of the Cyber Security Domain

Leo Obrst[a], Penny Chase[b], Richard Markeloff[a]

The MITRE Corporation

[a]McLean, VA

[b]Bedford, MA

{lobrst, pc, rmarkeloff}@mitre.org

***Abstract*— This paper reports on a trade study we performed to support the development of a Cyber ontology from an initial malware ontology. The goals of the Cyber ontology effort are first described, followed by a discussion of the ontology development methodology used. The main body of the paper then follows, which is a description of the potential ontologies and standards that could be utilized to extend the Cyber ontology from its initially constrained malware focus. These resources include, in particular, Cyber and malware standards, schemas, and terminologies that directly contributed to the initial malware ontology effort. Other resources are upper (sometimes called 'foundational') ontologies. Core concepts that any Cyber ontology will extend have already been identified and rigorously defined in these foundational ontologies. However, for lack of space, this section is profoundly reduced. In addition, utility ontologies that are focused on time, geospatial, person, events, and network operations are briefly described. These utility ontologies can be viewed as specialized super-domain or even mid-level ontologies, since they span many, if not most, ontologies -- including any Cyber ontology. An overall view of the ontological architecture used by the trade study is also given. The report on the trade study concludes with some proposed next steps in the iterative evolution of the Cyber ontology.**

***Index Terms*—ontology, malware, cyber, trade study.**

I. Introduction

This report is a trade study to support the development of a Cyber ontology. In this section we present the goals of both the Cyber ontology effort and this report. The following sections discuss the ontology development methodology and various ontologies and standards that could be utilized to extend the Cyber ontology. This report concludes with some proposed next steps in the iterative evolution of the Cyber ontology.

The ultimate goal of this effort is to develop an ontology of the cyber security domain, expressed in the OWL language, that will enable data integration across disparate data sources. Formally defined semantics will make it possible to execute precise searches and complex queries. Initially, this effort is focused on malware. Malware is one of the most prevalent threats to cyber security, and the MITRE team's work on the Malware Attribute Enumeration and Characterization (MAEC) language [1] provides a store of knowledge that can be readily leveraged.

As the scope of the ontology expands, the underlying conceptual framework will be provided by the Diamond Model of malicious activity [2], shown in Figure 1. The four corners of the diamond, Victim, Infrastructure, Capability, and Actor (the one threatening the victim), account for all the major dimensions of a malicious cyber threat.

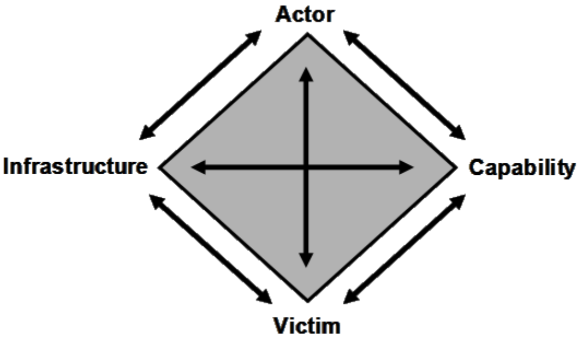


Fig. 1. The Diamond Model of malicious activity (from [2]).

The primary goals of this document are to explain the process followed in developing the Cyber ontology and catalog the sources upon which it is based. A secondary goal is to provide a compilation of resources useful for constructing semantic models in the cyber security domain.

II. Ontology Development Methodology

This section identifies the general methodology employed in the ontology development process, along with the specific methodology used to develop the Cyber ontology.

A. General Methodology

In general, the ontology development methodology employed here is called a "middle-out" approach. This means that it contains aspects of top-down analysis and bottom-up analysis. Bottom-up analysis requires understanding the semantics of the underlying data sources which are to be integrated. Top-down analysis requires understanding the semantics of the end-users who will actually use the resulting ontology-informed, semantically integrated set of data sources, i.e., the kinds of questions those end-users want to ask or could ask, given the enhanced capabilities resulting from the semantic integration of those data sources (e.g., questions that require temporal integration or reasoning, as over integrated timelines of events). See references [3-8].

These kinds of analyses result in the development of competency questions [7, 8]. These are the questions that need to be asked of the ontology in order to provide the targeted value to the users. As such, these questions can be viewed as the queries that need to be executed. These queries, in turn, can be viewed as a test procedure that indicates when the ontology development is sufficiently complete for a given stage of development, i.e., when those queries return results that are accurate, sufficiently rich, and at the right level of granularity as judged by a subject matter expert (SME).

Capturing the right competency questions is part of the requirements analysis phase of ontology development. These help identify use cases and scenarios. Taken together, the competency questions, uses cases, and scenarios enable the requirements to be fleshed out.

The key to ontology development here is of course an understanding of the cyber domain, which drives the kinds of entities, properties, relationships, and potentially rules that will be needed in the ontology.

B. *Specific Methodology*

More specifically, the methodology used for the current ontology development is based on the following principles, focused on parsimony and reuse:

Reuse of existing ontologies: Existing ontologies are reused where possible. The methodology of reuse consists of the following steps:

A. Establish the base of possible existing ontologies in the domain areas of interest, including foundational, mid-level, utility, and reference ontologies.

B. When developing the current Cyber ontology, incorporate classes and properties (and definitions) that exist in the best of the ontologies of (A).

C. When the number of classes and properties incorporated from a given ontology of (A) into the Cyber ontology grows large, consider directly importing the given ontology into the Cyber ontology, and establishing equivalence relations between the classes of the (A) ontology and the classes of the Cyber ontology.

Harvesting of existing schemas, data dictionaries, glossaries, standards: Other structured and definitional resources are used when available, as a form of knowledge acquisition of the domain. These resources are analyzed for the kinds of entities, relationships, properties, attributes, and the range of values for those, expressed in the resource. Where it makes sense, and as correlated with other Cyber database schemas and expressed analyst questions and interests (and their decompositions), these entities, relationships, properties, and values are incorporated into the Cyber ontology, after refinement according to ontological engineering principles.

Keeping it simpler: Where possible, the simpler ontological approach is chosen. This can mean that, for example, where the choice is between a 4-D spacetime or a 3-D space and time conceptualization, the 3-D conceptualization is chosen because it is generally simpler for non-ontologists to understand.

C. *Cyber Ontology Architecture*

The final product of the ontology development methodology described above will be an ontology that consists of a number of modular sub-ontologies, rather than a single, monolithic ontology. Ontologies can be grouped into three broad categories of upper, mid-level and domain ontologies, according to their levels of abstraction [9]:

- Upper ontologies are high-level, domain-independent ontologies that provide common knowledge bases from which more domain-specific ontologies may be derived. Standard upper ontologies are also referred to as foundational or universal ontologies.
- Mid-level ontologies are less abstract and make assertions that span multiple domain ontologies. These ontologies may provide more concrete representations of abstract concepts found in the upper ontology. There is no clear demarcation point between upper and mid-level. Mid-level ontologies also encompass the set of ontologies that represent commonly used concepts, such as Time and Location. These commonly used ontologies are sometimes referred to as utility ontologies [10].
- Doman ontologies specify concepts particular to a domain of interest and represent those concepts and their relationships from a domain specific perspective. Domain ontologies may be composed by importing mid-level ontologies. They may also extend concepts defined in mid-level or upper ontologies.

These categories and their roles in ontology architecture are shown in Figure 2, reproduced from [9]. A further discussion can be found in [10].

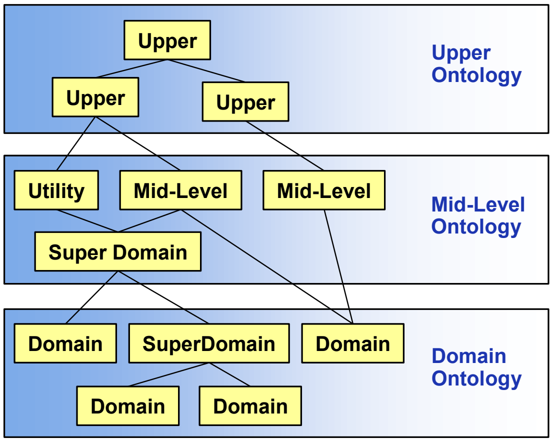


Fig. 2. Ontology architecture

Figure 3 depicts the expected architecture of the Cyber ontology. Each rounded box represents a major category of concepts. These concepts can be arranged along a level of abstraction continuum from broad and general to domain-specific. The larger bounding boxes represent separate ontologies that span multiple concept categories. The ontologies shown in Figure 3 and the sources they are based on are described in the following section.

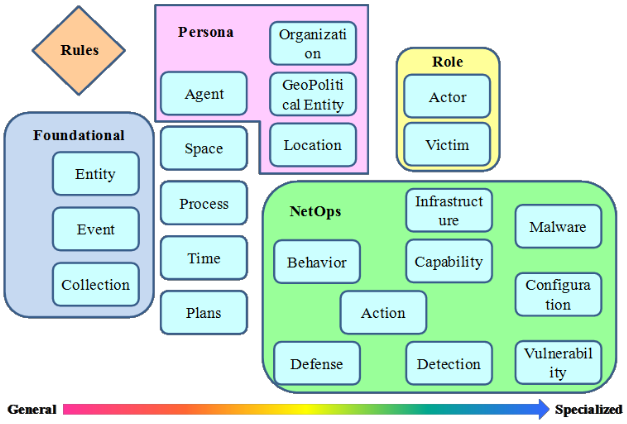


Fig 3. The Cyber ontology architecture

III. Resources for the Malware and Cyber Ontologies: Ontologies, Schemas, and Standards

There exist a variety of resources that can lay the groundwork for a Cyber ontology. This section presents a survey of those resources that we consider to be particularly applicable and important. These are not limited to ontologies, but also include taxonomies, lexica, and schemas.

A. Malware Resources

Published attempts to systematically categorize malware include one ontology [11] and three descriptive languages implemented in XML [1, 12, 13]. Also worthy of mention is an attempt at categorizing malware traits [14].

XML is a technology for defining text documents for information exchange, and the structure and content of a particular type of XML document is dictated by an XML schema. XML schemas offer enumerations of concepts and shared vocabularies for specific domains that can be useful as a basis for ontology development. However, XML schemas do not define formal semantics for the terms they contain, and are therefore not equivalent to ontologies.

1) Swimmer's Ontology of Malware Classes

A paper by Morton Swimmer [11] is the only non-trivial attempt to construct an ontological model of malware that we could identify. Swimmer's ontology is intended to enable data exchange between security software products. Swimmer's taxonomy of malware classes is shown in Figure 4.

Swimmer's malware class hierarchy is relatively simple. It organizes malware into well-known categories such as Trojan horse, virus, and worm. This may not be useful for malware instances that exhibit either behaviors from multiple classes or novel behaviors not associated with any recognized class.

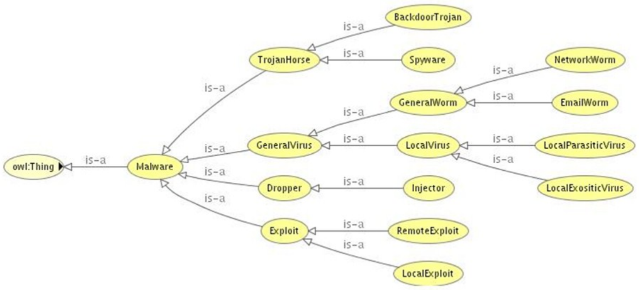


Fig. 4. Swimmer's malware class hierarchy (from [11]).

In Swimmer's taxonomy of malware characteristics, all malware characteristics belong to one of three high-level classes:

- Payload. This is assumed to be programmed with malicious intent.
- Vector. This defines how the malware is deployed or spread.
- Obfuscation. Characteristics for evading detection.

In describing vector characteristics, Swimmer coins the term "insituacy" to mean "the state the Malware strives to be in through its actions".

2) MAEC: Malware Attribute and Enumeration Characterization

MAEC is intended as a language for addressing all known types, variants, and manifestations of malware. Current signature-based malware detection techniques identify malware using a single metadata entity (e.g., a file hash), and MAEC's primary goal is to provide a more flexible method for characterizing malware based on patterns of attributes such as behaviors, artifacts, and attack patterns. This stands in contrast with Swimmer's work, which is focused on predefined malware families and discernible intent.

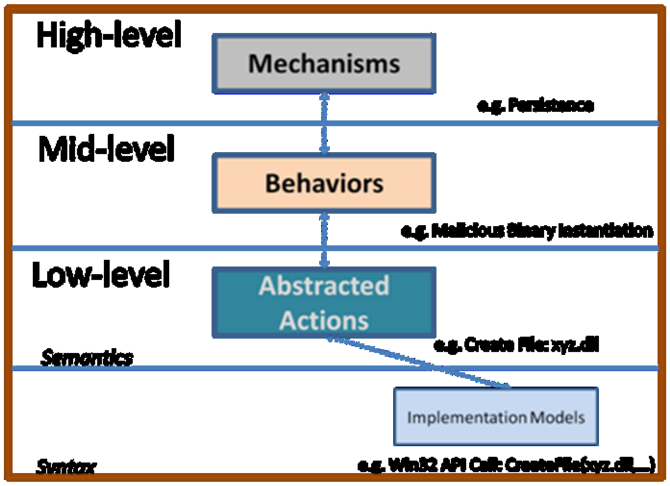


Fig. 5. The MAEC architecture

MAEC has a tiered architecture, as shown in Figure 5. At its lowest level, MAEC strives to portray what an instance of malware does by describing its actions, such as hardware accesses and system state changes. A distinction is drawn between semantics and syntactics by abstracting actions away from their implementations. This facilitates correlation between malware instances that do similar things at a low-level but with different implementations (such as malware targeted at different platforms).

MAEC's middle level describes malware behaviors. Behaviors serve to organize and define the purpose behind low-level actions, whether in groups or as singletons. Behaviors can represent discrete components of malware functionality at a level that is useful for analysis, triage, detection, etc.

MAEC's top level summarizes malware in terms of its mechanisms. Mechanisms are organized groups of behaviors. Some examples would be propagation, insertion, and self-defense. Since there is likely a low upper bound on the number of possible mechanisms, they can be useful in understanding the composition of malware at a very high level.

There are other resources such as the Industry Connections Security Group (ICSG) Malware Metadata Exchange Format [12], and Zeltser's Categories of Common Malware Traits [14], which space limitations preclude us from elaborating.

B. *Languages for Cyber Security Incidents*

Howard and Longstaff's seminal work [15] represents an early attempt to establish a common language for describing computer and network security incidents. Since then, industry and standards organizations have promulgated several languages for describing computer and network security incidents. Some of the prominent ones are described below. These languages all share the goal of facilitating information sharing across the cyber security community.

OpenIOC is an XML format for sharing intelligence related to cyber security incidents. Intelligence is organized as Indicators of Compromise (IOCs), which represent patterns that suggest malicious activity. OpenIOC has been developed by MANDIANT [13] and offered as an open standard. MANDIANT's products are widely used by defense contractors, and consistency with OpenIOC facilitates processing information from the Defense Industrial Base (DIB). OpenIOC includes around 30 separate XML schemas that describe various classes of objects that can be used to detect suspicious activity, such as MD5 hashes, registry keys, IP addresses, etc. The OpenIOC schemas are probably the most comprehensive descriptions of these types of objects available. The MAEC team incorporated the OpenIOC objects into MAEC and subsequently the OpenIOC objects formed the starting point for CybOX objects (CybOX is discussed in Section III.H).

IODEF [16] is a specification, in the form of an XML schema, developed by the IETF Extended Incident Handling (INCH) Working Group of the Internet Engineering Task Force (IETF) [17]. IODEF is an information exchange format for Computer Security Incident Response Teams (CSIRTs). It also provides a basis for the development of interoperable tools and procedures for incident reporting.

The VERIS framework [18] is used by Verizon Business [19] to collect security incident data from anyone who volunteers to submit it. These data are collected using a Web application [20]. The goal is to collect data of sufficient quantity and quality to support statistical analyses. Verizon's data collection is based on what they refer to as the A4 Threat Model. In this model, security incidents are regarded as a series of events where an organization's information assets are adversely affected. These events have four descriptive dimensions:

- Agent: Whose actions affected the asset
- Action: What actions affected the asset
- Asset: Which assets were affected
- Attribute: How the asset was affected.

The details of the VERIS model are available online in a Wiki format [18].

C. *Attack Patterns and Process Models*

The literature offers a number of attempts to create taxonomies and conceptual models of cyber attacks and attack patterns. Howard and Longstaff's [15] attack model is shown in Figure 6. In their model, an attacker uses a tool to exploit a vulnerability. This produces an action on a target (which together comprises an event). The intention is to accomplish an unauthorized result.

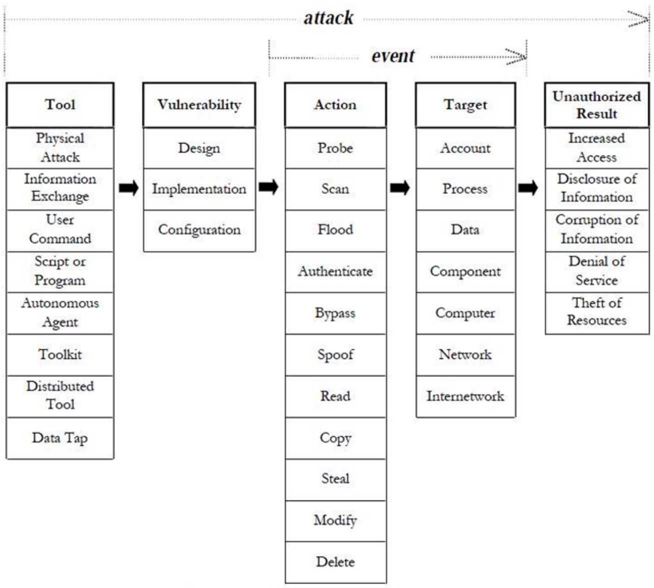


Fig. 6. Howard and Longstaff's model of computer and network attacks (from [15]).

A more recent work in a similar vein [21], presented at the 2007 IEEE International Symposium on Network Computing and Applications, delineates a model for the attack process that consists of the following phases:

- Reconnaissance. The search for information about potential victims.
- Gain Access. Gaining access, at the desired level, to a victim's system.
- Privilege Escalation. Escalate the initial privilege level, as necessary.
- Victim Exploration. Gaining knowledge of the victim's system, including browsing files, searching user accounts, identifying hardware, identifying installed program, and searching trusted hosts.

- Principal actions. Taking steps to accomplish the ultimate objective of the attack, such as installing malicious software or compromising data integrity.

This model is shown in flowchart form in Figure 7, reproduced from [21].

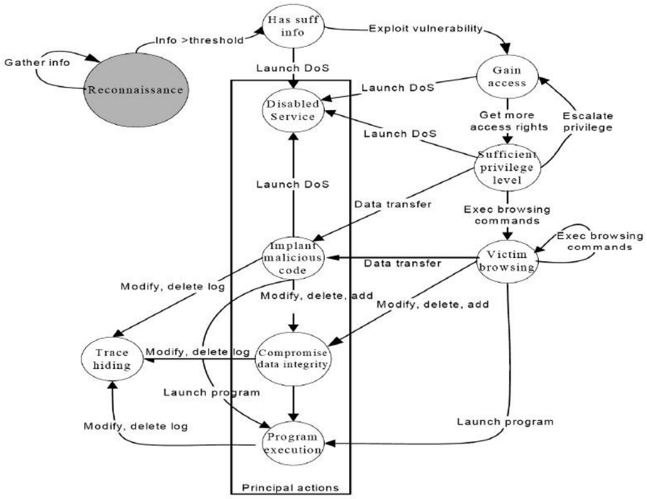


FIG. 7. A proposed attack process model (from [21]).

Relevant discussions of attack phases can also be found in blog postings by Bejtlich [22] and Cloppert [23].

The CAPEC catalog [24] defines a taxonomy of attack patterns. The CAPEC catalog currently contains 68 categories and 400 attack patterns. Attack patterns are modeled after object-oriented design patterns, and by design they exclude low-level implementation details. Categories are containers for related attack patterns. The patterns are more or less aligned with the top two MAEC layers, and categories roughly correspond to MAEC mechanisms.

The WASC Threat Classification [25] is similar to CAPEC.

D. Foundational Ontologies for the Cyber Ontology

Modeling choices are made in the development of foundational ontologies that have a downward impact on mid-level and domain ontologies. We cannot describe some of these ontological choices here, but invite the reader to see [9].

There are several foundational ontologies that could be considered for use in the Cyber ontology. These range from Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) [26], Basic Formal Ontology (BFO) [27], Object-Centered High-Level REference ontology (OCHRE) [28], Generic Formal Ontology (GFO) [29], Suggested Upper Merged Ontology (SUMO) [30], Unified Foundational Ontology (UFO) [31, 32], and Cyc/OpenCyc [33-35].

E. Utility Ontologies

The Cyber ontology will necessarily include concepts from domains that transcend cyber security, such as notions concerning people, time, space, and events. Where possible, the Cyber ontology will import existing ontologies to provide descriptions of these concepts. In this section we very briefly catalog the utility ontologies that we would consider for inclusion in the Cyber ontology.

1) Persons

Modeling the Actor and Victim nodes in Figure 1-1 will entail an ontological description of persons, their social roles and relationships, and their relationships to things. Among the available ontologies that might address this need, we include Friend Of A Friend (FOAF) [36], DOLCE Social Objects [37] which includes social roles and organizations.

2) Time

The Cyber ontology will need to be able to express notions of time instances and intervals, as well as concepts related to clock and calendar time. Various theories of the structure of time have been proposed; see [38] for a survey. Of particular interest is Allen's Interval Algebra for temporal reasoning [39]. Allen's calculus defines 13 basic relations between two time intervals.

There are two W3C standard ontologies of temporal concepts, OWL-Time [40] and time-entry [41]. They both provide similar vocabularies for expressing facts about temporal intervals and instants, while time-entry also includes the concept of an event. Both ontologies contain object properties that implement the Allen relations. Also included in the ontologies are classes and relations for expressing intervals and instants in clock and calendar terms. Both ontologies include the concept of a time zone, and a separate global time zone ontology is available [42].

3) Geospatial

The Cyber ontology may require geospatial concepts to describe the physical locations of people or infrastructure. See [43] for a comprehensive survey of available geospatial ontologies. Another source of information about geospatial ontologies is the Spatial Ontology Community of Practice (SOCoP) [44]. SOCoP is chartered as a Community of Practice under the Best Practices Committee of the Federal CIO Council.

The two-dimensional analog to Allen's Interval Algebra for qualitative spatial representation is the Region Connection Calculus 8 (RCC-8) [45], so named because eight basic relations comprise the calculus. RCC theory can be extended to support reasoning about regions with indeterminate boundaries [46].

If it is the case that a significant portion of the geospatial information to be described by the Cyber ontology is in the form of text mentions of place names, then the GeoNames Ontology [47] may be suitable for inclusion in the ontology. Although GeoNames does not support RCC-8, it has relations such as locatedIn, nearby, and neighbor. It is accompanied by a knowledge base containing 140 million assertions about 7.5 million geographical objects that span the globe. A typical use for GeoNames is to infer what country a given town, city, or region is located in.

F. Events and Situations

Events are entities that describe the occurrences of actions and changes in the real world. Situations represent histories of action occurrences. In this context at least, situations are not equivalent to states. Events and situations are dynamic and challenging to model in knowledge representation systems.

As in the temporal and spatial domains, logic formalisms have been created for representing and reasoning about events and situations. These are the event calculus [48] and situation calculus [49]. Both calculi employ the notion of fluents. A fluent is a condition that can change over time. The main elements of the event calculus are fluents and actions, and for the situation calculus they are fluents, actions and situations.

Notions of events and situations are included in several of the ontologies previously described. DOLCE, GFO, Cyc, and time-entry all have Event classes. GFO has a class named History that corresponds to the concept of a situation, and Cyc has a Situation class. BFO's ProcessualEntity class has subclasses that correspond closely to events and situations.

Ontologies for events and situations include a DOLCE extension for descriptions and situations [50], a proposed upper event ontology [51], and an ontology for Linking Open Descriptions of Events (LODE) [52].

G. Network Operations

A network operations (NetOps) OWL ontology was developed in 2009 by MITRE as part of the data strategy effort supporting the NetOps Community of Interest (COI). The NetOps ontology includes entities and events, and represents mission threads of interest to US federal government network management.

H. Other Cyber Resources

There are a number of other resources that can be mined for concepts, abstractions, and relationships between entities that may be suitable for inclusion in a Cyber ontology.

Common Event Expression (CEE) [53] is intended to standardize the way computer events are described, logged, and exchanged. Some of these events would naturally correspond to malware actions and behaviors. The CEE components most relevant to cyber security ontology development are the Common Dictionary and Event Expression Taxonomy (CDET). The dictionary defines a collection of event fields and field value types that are used throughout CEE to specify the values of properties associated with specific events. The taxonomy specifies event types. Examples of event types are user login, service restart, network connection, privilege elevation, and account creation.

A recent foundational schema for the cyber domain is Cyber Observable Expression (CybOX) [54]. CybOX is designed for the specification, capture, characterization and communication of events or stateful properties observable in the cyber domain in support of a wide range of use cases. MAEC and CEE both leverage CybOX for describing cyber objects, actions, and events. An emerging schema is the Structured Threat Information Expression (STIX) [55], which provides an overarching framework for describing threat information, including adversaries, tactics, techniques and procedures (TTPs), incidents, indicators, vulnerabilities, and courses of actions. Malware is included under the heading of TTPs. STIX references other schemas and cyber information, including MAEC, CybOX, CVE, and CPE.

Security Content Automation Protocol (SCAP) [56] is a suite of specifications that standardize the format and nomenclature by which security software products communicate software flaw and security configuration information. In its current incarnation [57], SCAP is comprised of seven specifications:

- eXtensible Configuration Checklist Description Format (XCCDF) [58], a language for authoring security checklists/benchmarks and for reporting results of checklist evaluation.

- Open Vulnerability and Assessment Language (OVAL) [59], a language for representing system configuration information, assessing machine state, and reporting assessment results.

- Open Checklist Interactive Language (OCIL) [60], a framework for expressing a set of questions to be presented to a user and corresponding procedures for interpreting responses to these questions.

- Common Platform Enumeration (CPE) [61], a nomenclature and dictionary of hardware, operating systems, and applications.

- Common Configuration Enumeration (CCE) [62], a nomenclature and dictionary of security software configurations.

- Common Vulnerabilities and Exposures (CVE) [63], a nomenclature and dictionary of security-related software flaws.

- Common Vulnerability Scoring System (CVSS) [64], an open specification for measuring the relative severity of software flaw vulnerabilities

Of these standards, the ones most germane to developing a Cyber ontology would be OVAL, CPE, CCE and CVE. Parmelee [65] has outlined a semantic framework for these four standards built upon loosely-coupled modular ontologies. Parmelee's framework is intended to simplify data interoperability across automated security systems based on the OVAL, CPE, CCE and CVE standards.

IV. CYBER ONTOLOGY DEVELOPMENT: NEXT STEPS

The current Cyber ontology is focused primarily on malware and some preliminary aspects of the so-called 'diamond model', which includes actors, victims, infrastructure, and capabilities. Necessarily, more of the infrastructure and capabilities were developed first; however, even these are not yet developed to the level of detail that is warranted, i.e., expanding on behavioral aspects and events, in particular that are the core of Cyber, would make it more useful. These are our next steps.

Acknowledgment

© 2012, The MITRE Corporation. All Rights Reserved. The views expressed in this paper are those of the authors alone and do not reflect the official policy or position of The MITRE Corporation or any other company or individual.

References

[1] MAEC - Malware Attribute Enumeration and Characterization. [Online] http://maec.mitre.org/.

[2] Ingle, J. Organizing Intelligence to Respond to Network Intrusions and Attacks. *Briefing for the DoD Information Assurance Symposium.* Nashville, TN, 2010.

[3] Fernandéz, M., Gómez-Pérez, A. and and Juristo, N. METHONTOLOGY: From Ontological Art to Ontological Engineering. *AAAI97 Workshop on Ontological Engineering, Spring Symposium Series*. Stanford University, 1997. pp. 33-40.

[4] Fernández M. et al. Building a Chemical Ontology Using Methontology and the Ontology Design Environment. *IEEE Intelligent Systems.* January/February 1999. Vol. 14, 1. http://www.aifb.uni-karlsruhe.de/Lehrangebot/Sommer2001/SemanticWeb/papers/chemical_ontology.pdf.

[5] Fernández, M. Overview of Methodologies for Building Ontologies. Workshop on Ontologies and Problem-Solving Methods: Lessons Learned and Future Trends. (IJCAI99). August 1996.

[6] Gómez-Pérez, A., Fernández, M. and de Vicente, A. Towards a Method to Conceptualize Domain Ontologies. *ECAI '96Workshop on Ontological Engineering.* Budapest, Hungary : s.n., 1996. pp. 41-52.

[7] Gruninger, M. and Fox, M. S. Methodology for the design and evaluation of ontologies. Montreal, 1995.

[8] Uschold, M. and Gruninger, M. Ontologies: Principles, Methods, and Applications. 1996. Vol. 11, 2, pp. 93-136.

[9] Obrst, L. Ontological Architectures. [ed.] Johanna Seibt, Achilles Kameas Roberto Poli. Chapter 2 in Part One: Ontology as Technology in the book: TAO – Theory and Applications of Ontology, Volume 2: Computer Applications. Springer, 2010.

[10] Semy, S., Pulvermacher, M. and Obrst, L. Toward the Use of an Upper Ontology for U.S. Government and U.S. Military Domains: An Evaluation. *MITRE Technical Report, MTR 04B0000063.* November 2005.

[11] Swimmer, M. Towards An Ontology of Malware Classes. [Online] January 27, 2008. http://www.scribd.com/doc/24058261/Towards-an-Ontology-of-Malware-Classes.

[12] IEEE-SA - Industry Connections. [Online] http://standards.ieee.org/develop/indconn/icsg/malware.html.

[13] MANDIANT: Intelligent Information Security. [Online] http://www.mandiant.com.

[14] Zeltser, L. Categories of Common Malware Traits. *Internet Storm Center Handler's Daiiry.* [Online] Sept. 25, 2009. http://isc.sans.edu/diary.html?storyid=7186.

[15] Howard, J. D. and Longstaff, T. A Common Language for Computer Security Incidents. [Technical Report]. Sandia National Laboratories, 1998.

[16] Cover Pages Incident Object Description and Exchange Format (IODEF). [Online] http://xml.coverpages.org/iodef.html.

[17] Internet Engineering Task Force. [Online] http://www.ietf.org/.

[18] VERIS Framework. [Online] https://verisframework.wiki.zoho.com/.

[19] Verizon Business. [Online] http://www.verizonbusiness.com/.

[20] Verizon Incident Classification and Reporting. [Online] https://www2.icsalabs.com/veris/incidents/new#/welcome.

[21] Gadelrab, M., El Kala, A. and Deswarte, Y. Execution Patterns in Automatic Malware and Human-Centric Attacks. *IEEE International Symposium on Network Computing and Applications.* 2008.

[22] Bejtlich, R. TaoSecurity: Incident Phases of Compromise. [Online] June 6, 2009. http://taosecurity.blogspot.com/2009/06/incident-phases-of-compromise.html.

[23] Cloppert, M. [Online] Oct. 14, 2009. http://computer-forensics.sans.org/blog/2009/10/14/security-intelligence-attacking-the-kill-chain/.

[24] CAPEC - Common Attack Pattern Enumeration and Characterization. [Online] http://capec.mitre.org/.

[25] The Web Application Security Consortium/Threat Classification. [Online] http://projects.webappsec.org/w/page/13246978/Threat-Classification.

[26] Laboratory for Applied Ontology - DOLCE. [Online] http://www.loa-cnr.it/DOLCE.html.

[27] Basic Formal Ontology (BFO). [Online] http://www.ifomis.org/bfo.

[28] Schneider, L. How to Build a Foundational Ontology -- The Object-Centered High-level Reference Ontology OCHRE. *Proceedings OF THE 26TH Annual German Conference on AI, KI 2003: Advances In Artificial Intelligence* . 2003.

[29] General Formal Ontology (GFO). [Online] http://www.onto-med.de/ontologies/gfo/.

[30] Niles, I., and Pease, A. Towards a Standard Upper Ontology. [ed.] Chris Welty and Barry Smith. Proceedings of the 2nd International Conference on Formal Ontology in Information Systems (FOIS-2001). 2001.

[31] Guizzardi, G., Wagner, G. Some Applications of a Unified Foundational Ontology in Business. [ed.] Michael Rosemann and Peter Green. *Ontologies and Business Systems Analysis.* IDEA Publisher, 2005.

[32] Guizzardi, G., Wagner, G. Towards Ontological Foundations for Agent Modeling Concepts using UFO. Agent-Oriented Information Systems (AOIS), selected revised papers of the Sixth International Bi-Conference Workshop on Agent-Oriented Information Systems. Springer-Verlag, 2005.

[33] Cycorp, Inc. [Online] http://cyc.com/cyc/technology/whatiscyc_dir/whatsincyc.

[34] Cycorp, Inc. [Online] http://cyc.com/cyc.

[35] OpenCyc.org. [Online] http://www.opencyc.org/.

[36] The Friend of a Friend (FOAF) project. [Online] http://www.foaf-project.org/.

[37] Masolo, C. et al. Social Roles and their Descriptions. *Proceedings of KR'2004.* 2004. pp. 267-277.

[38] Hayes, P. A Catalog of Temporal Theories. *Technical Report UIUC-BI-AI-96-01.* s.l. : Univerisity of Illinois, 1996.

[39] Allen, J. F. Maintaining knowledge about temporal intervals. *Communications of the ACM.* 1983.

[40] Hobbs, J. R. and Pan, F. An Ontology of Time for the Semantic Web. CM Transactions on Asian Language Processing (TALIP): Special issue on Temporal Information Processing. 2004. Vol. 3, 1, pp. 66-85.

[41] Pan, F. and Hobbs, J. R. Time in OWL-S. *Proceedings of the AAAI Spring Symposium on Semantic Web Services.* s.l. : Stanford University, 2004. pp. 29-36.

[42] A Time Zone Resource in OWL. [Online] http://www.isi.edu/~hobbs/timezonehomepage.html.

[43] Ressler, J., Dean, M. and Kolas, D. Geospatial Ontology Trade Study. [ed.] Terry Janssen, Werner Ceuster Leo Obrst. *Ontologies and Semantic Technologies for Intelligence.* Amsterdam, Berlin, Tokyo, Washington D.C. : IOS Press, 2010, Chapter 11, pp. 179-212.

[44] Spatial Ontology Community of Practice (SOCoP). [Online] http://www.socop.org/.

[45] Randall, D., Cui, Z. and and Cohn, A. A spatial logic based on regions and connection. *Proceedings of the 3rd International Conference on Principles of Knowledge Representation and Reasoning.* Cambridge, MA, 1992. pp. 165-176.

[46] Gotts, A. Cohn and N. The 'Egg-Yolk' representation of regions with indeterminate boundaries. [ed.] P. Burrough and A. M. Frank. *Proceedings, GISDATA Specialist Meeting on Geographical Objects with Undetermined Boundaries.* Francis Taylor, 1996. pp. 171-187.

[47] GeoNames Ontology - Geo Semantic Web. [Online] http://www.geonames.org/ontology/documentation.html.

[48] Kowalski, R. and Sergot, M. A Logic-based Calculus of Events. *New Generation Computing* . 1986. Vol. 4, pp. 67–95.

[49] Reiter, R. The frame problem in the situation calculus: a simple solution (sometimes) and a completeness result for goal regression. [ed.] Vladimir Lifshitz. *Artificial intelligence and mathematical theory of computation: papers in honour of John McCarthy.* San Diego, CA : Academic Press Professional, Inc., 1991. pp. 359-380.

[50] Gangemi, A. and Mika, P. Understanding the Semantic Web through Descriptions and Situations. *Proceedings of CoopIS/DOA/ODBASE.* 2003. pp. 689-706.

[51] Kaneiwa1, K. Iwazume, M. and Fukuda, K. An upper ontology for event classifications and relations. *AI'07 Proceedings of the 20th Australian joint conference on Advances in artificial intelligence* . 2007.

[52] LODE: Linking Open Descriptions of Events. [Online] http://escholarship.org/uc/item/4pd6b5mh.

[53] Common Event Expression: CEE, A Standard Log Language for Event Interoperability in Electronic Systems. [Online] http://cee.mitre.org/.

[54] CybOX – Cyber Observable Expression. [Online] http://cybox.mitre.org/

[55] STIX-whitepaper. [Online] http://measurablesecurity.mitre.org/docs/STIX-Whitepaper.pdf

[56] The Security Content Automation Protocol (SCAP) - NIST. [Online] http://scap.nist.gov/.

[57] Quinn, Waltermire, Johnson, Scarfone, Banghart. The Technical Specification for the Security Content Automation Protocol (SCAP): SCAP Version 1.1 (DRAFT). Gaithersburg, MD : NIST, 2011. SP800-126.

[58] XCCDF - The eXtensible Configuration Checklist Description Format - The Security Content Automation Protocol (SCAP) - NIST. [Online] http://scap.nist.gov/specifications/xccdf/.

[59] OVAL - Open Vulnerability and Assessment Language. [Online] http://oval.mitre.org/.

[60] OCIL - The Open Checklist Interactive Language - The Security Content Automation Protocol (SCAP) - NIST. [Online] http://scap.nist.gov/specifications/ocil/.

[61] CPE - Common Platform Enumeration. [Online] http://cpe.mitre.org/.

[62] Common Configuration Enumeration (CCE): Unique Identifiers for Common System Configuration Issues. [Online] http://cce.mitre.org/.

[63] CVE - Common Vulnerabilities and Exposures. [Online] http://cve.mitre.org/.

[64] Common Vulnerability Scoring System (CVSS-SIG). [Online] http://www.first.org/cvss/.

[65] Parmelee, M. *Toward an Ontology Architecture for Cyber-Security Standards.* George Mason University, Fairfax, VA : Semantic Technologies for Intelligence, Defense, and Security (STIDS) 2010.

A Policy-Based Dialogue System for Physical Access Control

Mohammad Ababneh
Department of Computer Science
George Mason University
Fairfax, VA, USA
mababneh@gmu.edu

Duminda Wijesekera
Department of Computer Science
George Mason University
Fairfax, VA, USA
dwijesek@gmu.edu

James Bret Michael
Department of Computer Science
Naval Postgraduate School
Arlington, VA, USA
bmichael@nps.edu

***Abstract*—We prototype a policy-based dialog system for providing physical access control to secured facilities and smart buildings. In our prototype system, physical access control policies are specified using the eXtensible Access Control Markup Language. Based on the policy and the user's presence information, our dialog system automatically produces a series of questions and answers that, if correctly answered, permit the requester to enter the secure facility or smart building. The novelty of this work is the system's ability to generate questions appropriate to the physical location, time of day, and the requester's attributes.**

***Index Terms*—Dialogue, Question Answering, Voice Recognition, VXML, XACML, Access Control Policy, Security**

I. OVERVIEW

We developed a prototype policy-based system for guarding entry to physical facilities, such as smart buildings. The system interacts with potential entrants using a spoken dialogue. Our physical access control system uses the OASIS consortium's eXtensible Access Control Markup Language (XACML) standard [1] and the W3C's Voice eXtensible Markup Language (VXML) [2] for specifying the dialogue.

In order to generate dialogues from physical access control policies specified in XACML, we generate so-called "VXML Voice forms" from XACML policy rules. In this paper we describe our initial implementation of the prototype. Given that emerging mobile applications use interactive voice commands such as Apple's Siri, Google's Andriod S-Voice, and Microsoft Windows Speech Recognition, we envision that new applications would emerge for interactive voice-based access to resources.

The dialogues we generate are in the form of a question and an (acceptable) answer. In our prototype, questions are generated using a grammar for words or phrases—belonging to a restricted vocabulary—that are taken from an XACML rule's subject-attribute values [3][4]. Answers should conform to a grammar that is linked to the rule's data type, and acceptable answers are those that provide the values that match the values specified in the XACML policy. The class of grammars may come from and be checked against a database or other sources of subject attributes.

For each question, user input is collected and stored in a variable. These variables are used to generate an XACML request that is passed to an XACML policy decision point (PDP). The PDP is responsible for the decision of either granting or denying access. If access is granted, the system sends a control-system message to an actuator that unlocks the entry door to the facility.

We succeeded in generating a question for each rule in the policy. Our existing implementation uses a small number of policy rules and their conversion. We are currently working on scaling this implementation by addressing the run-time and automatic conversion of a large number of rules, in addition to developing the capability to dynamically generate the grammars using .grxml files [3].

Due to advancements in mobile applications and the emergence of voice user interfaces (VUI) as well as their being provided as a service in the cloud, new access control mechanisms are needed. When completed, our current architecture and implementation will serve as a testbed for further research and development.

A. Potential Applications

The following are potential applications for our system:

- *Mobile computing:* The adaptation of the voice technology and VUI in mobile computing (e.g., Apple's Siri, Google's Android S-Voice, Microsoft Windows Speech Recognition, Research in Motion’s BlackBerry) introduces challenges in using the technology in order to accomplish more sophisticated tasks such as access control to either resources and services locally on the devices or remotely at data centers via cloud services. We envision this system to be useful for providing ubiquitous policy-based automated physical access control from mobile devices, where hands-free usage is valued or required.

- *Military and first-responder applications:* It is important to provide hands-free user interfaces to military personnel, especially those participating in combat operations, as well as first responders (e.g., firefighters, police, paramedics). These personnel have to access support facilities (e.g., gates at a forward operating bases or police sub-stations) and equipment (e.g., fire trucks) (see [5]).
- *Information systems:* In general this access control can be used to provide appropriate access to any information system using a VUI.
- *Electronic Commerce and Business:* This new access control approach can be used in performing transactions in e-commerce applications accessed via mobile computing devices.
- *Physical Access Control:* Access control to critical facilities can be automated and access can be granted or denied based on an enterprise’s policy. Policy languages such as XACML are standardized to unify policies across enterprises and reduce administrative load. This is all hands-free and it only depends on answers to questions that represent the subject’s attributes. A sample dialogue for access control is illustrated in Table 1.

1	U	Hello
	S	Welcome, Please Say who you are?
2	U	I am User01 (Alice)
	S	Please say your Password or enter it using the key pad
3	U	Pass01
	S	Ok, I got that. Now tell me Why do you need to access the building (Role)
4	U	Professor
	S	What is your office number
5	U	4429
	S	In which floor is the dean’s meeting room
6	U	5th
	S	What is the time on the clock to your right
7	U	7:30
	S	Are you sure
8	U	Yes
	S	Is it raining outside?
9	U	Yes
	S	Is it cold or warm?
10	U	Cold
	S	Is the temperature below 40 degrees?
11	U	No
	S	Can you estimate the range that the temperature might be in?
12	U	50 to 60
	S	Ok, granted access

U: User, S: System

TABLE 1. A SAMPLE DIALOGUE FOR ACCESS CONTROL

The original intention of access control policy languages such as XACML was to deal with systems and access control enforcement points, but not humans. In our approach we use XACML to drive human interaction through dialogues with the system. This new approach has implications on the way dialogues are generated and access control decisions are taken. As mentioned above, many potential applications might take advantage of this approach. To be applicable, many aspects of human voice interactions should be studied—more research on this but it is outside of the scope of the work we report on here.

One important aspect of such an approach is the scale of implementation, especially in case of physical control. The user of such a system will likely not use the system if the system takes a long time to make an access decision for each individual of a large number of humans waiting in crowds such as at a sporting event (e.g., a World Cup football match). The processing time of an access request initiated by a human entity is going to be different than the time initiated by an automated process or application. This is another area of research we have left to future work.

II. BACKGROUND

In this section we discuss the most relevant standards and technologies to our research.

A. VoiceXML

VoiceXML (VXML) is the Voice Markup Language developed and standardized by the W3C’s Voice Browser Working Group. It is intended for creating audio dialogues that feature synthesized speech, digitized audio, recognition of spoken and Dual Tone Multi-Frequency (DTMF) key inputs, recording of spoken input, telephony, and mixed initiative conversations. VXML is similar to HTML in the textual arena in providing an interface between a user and the Web, using a voice interface. Its purpose is to bring the advantages of web-based development and content delivery to interactive voice response (IVR) applications. All Web technologies are still relevant in any voice interface, such as services, markup languages, linking, URIs, caching, standards, accessibility, and cross-browser [2].

The most important terms in VXML are:

- *Form:* Forms define an interaction that collects values for a set of form-item variables. Each field may specify a grammar that defines the allowable inputs for that field. If a form-level grammar is present, it can be used to fill several fields from one utterance
- *Block:* An item is a component of a form that presents information by synthesizing a phrase of text into speech to the user.
- *Field:* An item is a component of a form that gathers input from the user by synthesizing a textual phrase into speech for the user. The user must provide a value for the field before proceeding to the next element in the form.
- *Menu:* A menu presents the user with a choice of options and then transitions to another dialog based on that choice

B. XACML

XACML is an OASIS standard XML-based language for specifying access control policies [1]. In a typical XACML usage scenario, a subject that seeks access to a resource submits a query through an entity called a Policy Enforcement Point (PEP). The PEP, responsible for controlling access to the resource, forms a request in the XACML request language and sends it to the PDP. The PDP in turn evaluates the request and sends back one of the following responses: accept, reject, error, or unable to evaluate, with the PEP allowing or denying access to the requester accordingly, as shown in Figure 1.

Figure 1 contains the following entities:

- *Policy Set:* A set of policies.
- *Policy:* A set of rules, an identifier for the rule-combining algorithm, and (optionally) a set of obligations. May be a component of a policy set.
- *Rule:* A target, an effect, and a condition. A component of a policy.
- *Combination Algorithm:* The procedure for combining the decision and obligations from multiple policies.
- *Subject:* An actor whose attributes may be referenced by a predicate.
- *Target:* The set of decision requests, identified by definitions for resource, subject, and action that a rule, policy, or policy set is intended to evaluate.
- *Resource:* Data, service or system component.
- *Attribute:* All entities are identified using attributes.

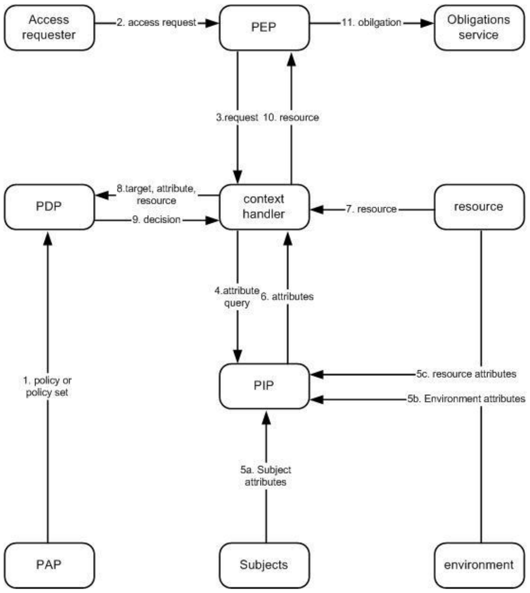


Fig. 1. XACML's Data-Flow Diagram

- *Predicate:* An evaluable statement about attributes.
- *PEP:* Governing entity of a resource.
- *PDP:* The entity that evaluates access requests.
- *PIP:* The entity that fetches attribute values for the PDP..
- *PAP:* The entity that retains polices.
- *Context Handler:* The entity that converts decision requests to XACML requests.

Some of the currently available implementations of the XACML specification are for example OpenXACML, enterprise-java-xacml from Google code, HERAS XACML, JBoss XACML, Sun Microsystem's XACML, and WSO2 Identity Server, which is used in this implementation.

III. IMPLEMENTATION PLATFORMS

In this section we introduce briefly the implementation platforms of the standards and technologies used to implement our prototype.

A. Voxeo for Speech Recognition

Voxeo Prophecy is a standards-based platform for speech, IVR, and Software Implemented Phone (SIP) applications for Voice over Internet Protocol (VoIP) applications [6]. Some of the capabilities integrated into the platform are: automatic speech recognition, speech synthesis, and visual programming. Prophecy provides libraries to create and deploy IVR or VoIP applications, including VXML and Call Control (CCXML) browsers with speech recognition and synthesis engines, and a built-in SIP soft-phone. Prophecy supports ASP, CGI, C#, Java, PERL, PHP, Python, and Ruby and has a built-in web server that supports PHP and Java applications. It complies with VXML and CCXML standards.

B. XACML Implementation – WSO2

WSO2 Identity Server [7] provides security and identity management of enterprise web applications, services, and APIs. WSO2 full implementation supports identity management, single sign-on, Role-based Access Control (RBAC), fine-grained access control, LDAP, OpenID, SAML, Kerberos, OAuth, WS-Trust, and the XACML 2.0/3.0.

C. Programming and development

In addition to the above two major platforms, programming languages such as Java, Java Server Pages (JSP), and Java Script (JS) were used to accomplish the integration and interoperability work between these platforms and thus enabled us to develop our prototype.

IV. OUR APPROACH

The core of our work transforms an access control policy into a voice platform supported voice language. We use XACML and W3C VXML for these two purposes.

In order to generate a dialogue between a user and an access control system to make it possible for the system to make a decision to whether grant or deny access, the access control policy is transformed into VXML. The rules in each policy are read and transformed into VoiceXML blocks and forms. The entire policy is parsed using a DOM parser and then each rule element is converted to a VXML block for the user interface translating text to speech (TTS), posing the question to the user, and then waiting for the user's response through voice recognition. The details of how TTS and voice recognition technologies are outside the scope of our current work; we are implementing these services through an integrated Voice Application Development Environment introduced in Section III.A. Figure 2 depicts the high-level architecture for our prototype system.

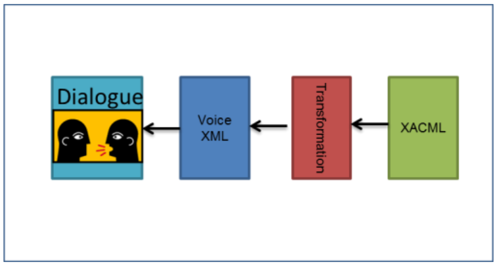


Fig. 2. High level Approach Architecture

A more detailed illustration of this policy voice (VXML) is shown in Figure 3.

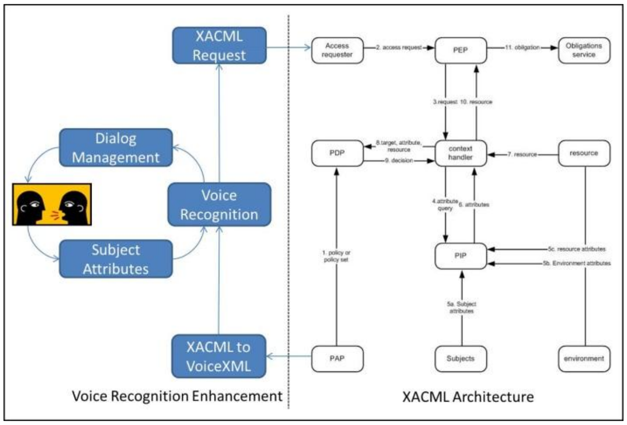


Fig. 3. Dialogue-Policy integration

A. A Working Scenario

We used the Voxeo Prophecy IVR platform (see www.voxeo.com/products/voicexml-ivr-platform.jsp)—including the webserver, designer, SIP, and application manager—to develop our application for voice recognition. We use a scenario to illustrate how the application works.

Our scenario starts with a XACML policy file, with the rule shown in Figure 4.

```
<?xml version="1.0" encoding="UTF-8"?>
<Policy RuleCombiningAlgId="identifier:rule-
combining-algorithm:deny-overrides"
PolicyId="urn:oasis:names:tc:example:SimplePolic
y1"
xsi:schemaLocation="urn:oasis:names:tc:xacml:2.0
:policy:schema:cd:04 http://docs.oasis-
open.org/xacml/access_control-xacml-2.0-policy-
schema-cd:04.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"
xmlns="urn:oasis:names:tc:xacml:2.0:policy:schem
a:cd:04">
<Description> Med Example Corp access control
policy
</Description>
<Target/>
<Rule Effect="Permit"
RuleId="urn:oasis:names:tc:xacml:2.0:example:Sim
pleRule1">
<Description> Any subject with an e-mail name in
the med.example.com domain can perform any
action on any resource.
</Description>
<Target>
<Subjects>
<Subject>
<SubjectMatch
MatchId="urn:oasis:names:tc:xacml:1.0:function:r
fc822Name-match">
<AttributeValue
DataType="urn:oasis:names:tc:xacml:1.0:data-
```

```
type:rfc822Name"> @med.example.com
</AttributeValue>
<SubjectAttributeDesignator
DataType="urn:oasis:names:tc:xacml:1.0:data-
type:rfc822Name"
AttributeId="urn:oasis:names:tc:xacml:1.0:subjec
t:subject-id"/>
</SubjectMatch>
</Subject>
</Subjects>
</Target>
</Rule>
</Policy>
```

Fig. 4. A sample XACML rule

Using JSP, we load the XACML file into a Document Object Manager (DOM) object. We read the rules inside the XACML document and link each rule to a VXML block/form. The JSP script extracts the attribute's value from the DOM's rule element and passes it to the Voxeo designer application, which converts it to VXML. Figure 5 shows an example of a VXML file.

```
<?xml version="1.0" encoding="UTF-8"?>
<vxml xmlns="http://www.w3.org/2001/vxml"
  xmlns:xsi="http://www.w3.org/2001/
  XMLSchema-instance"
 xsi:schemaLocation="http://www.w3.org/2001/vxml
   http://www.w3.org/TR/voicexml20/vxml.xsd"
   version="2.0">
  <form>
  <field name="e-mail">
     <prompt>What is your e-mail address?
     </prompt>
     <grammar src="email.grxml"
             type="application/srgs+xml"/>
  </field>
  <block>
<submit next="http://www.example.com/user.asp"/>
  </block>
 </form>
</vxml>
```

Fig. 5. A sample VXML

In order to create the question, a phrase is inserted before the attribute value in the form of "What is?" or "Is your?" followed by the attribute name extracted from the RuleID of the DOM's rule element. Our current implementation supports the yes/no answers to "Is your?" type of questions. We are in the process of enlarging the question formation syntax to support other question formats.

In this way a question will be generated for every rule in the policy file. The human user then needs to answer the questions posed by the system.

The next step is to collect attribute "VoiceXML variable" values generated throughout the dialogue and use them to generate an XACML request. Figure 6 shows an example of a request.

```
<Request>
<Subject>
<Attribute
AttributeId="urn:oasis:names:tc:xacml:1.0:subjec
t:subject-id"
DataType="urn:oasis:names:tc:xacml:1.0:data-
type:rfc822Name">
<AttributeValue>mababneh@@med.example.com
</AttributeValue>
</Attribute>
<Attribute AttributeId="group"
DataType=http://www.w3.org/2001/XMLSchema#string
Issuer="admin@gmu.edu">
<AttributeValue>Developers</AttributeValue>
</Attribute>
</Subject>
<Resource>
<Attribute
AttributeId="urn:oasis:names:tc:xacml:1.0:resour
ce:resource-id"
DataType="http://www.w3.org/2001/XMLSchema#anyUR
I">
<AttributeValue>http://server.example.com/code/d
ocs/developer-guide.html</AttributeValue>
</Attribute>
</Resource>
<Action>
<Attribute
AttributeId="urn:oasis:names:tc:xacml:1.0:action
:action-id"
DataType="http://www.w3.org/2001/XMLSchema#strin
g">
<AttributeValue>read</AttributeValue>
</Attribute>
</Action>
</Request>
```

Fig. 6. A sample XACML request

The XACML request will be sent to the XACML implantation of choice. In our case, we have chosen WSO2 Identity Server version 3.2. It has a distinguished, modern, and high-performance XACML implementation with a service-oriented implementation option. The Voxeo development environment supports both HTTP requests and web services. Our choice was to enforce the policy by accepting the XACML request through a web service interface with the WSO2 XACML PEP. The PDP will take a decision based on the attribute values collected from the dialogue and matches of values of subjects and resources in the XACML implementation (see Figure 1). The grant or deny decision will then be ready to be returned back to the application. In our current case, it should be translated to a physical access decision as to whether to open a door.

The XACML's access decision is based on the output of the policy and rule combining algorithms. Following the standard, the rule and policy should evaluate to true in order to grant access; otherwise it would produce "indeterminate" or "not applicable." In case there are multiple applicable rules and policies, the final access decision is the result of the logical combination of these algorithms.

Our work, in its final state, will illustrate the use of XACML to control access to resources through building dialogues with human users. There are efforts proposing access control systems through XACML interfacing with other data models. In the published literature, a majority of this integration effort is with Web Services [8]. Most of this harmonizing work relied on the use of the de facto Simple Object Access Protocol (SOAP) [9] messages in the Web Services architecture to extract security-related attributes and use them in XACML for the purpose of access control [10] [11].

Security Assertion Markup Language (SAML) profile for XACML is heavily relied on when there is a need to use additional subject's attributes that are administered by other authorities to evaluate access control requests [12]. SAML and other message exchange protocols can be the means through which XACML can interface with other data models. Our work is different by trying to let XACML reach the human user directly and initiating a dialogue with him, manage the dialogue, and then decide whether to allow access.

V. NEXT STEPS

Our next steps in this work would be:

- Finishing the XACML implementation
- Being able to generate requests and responses and execute them
- Determining the best way to use grammars
- Looking into the best way to generating VoiceXML from XACML: there are options to evaluate such as DOM and XSLT

Some of the follow-on research items we are pursuing are:

- Integrating presence information with the dialog access control system. It is critical for an automated system to authenticate to the system the speaker or requester of access to avoid certain attacks, such as by verifying the physical presence and human nature of the speaker.
- Making the dialog as short as possible. In a spoken dialog or question-answer system, it is different than filling a form using a keyboard and a mouse. It can take more time to say the voice block (form) than filling it by hand or via a keyboard. In some cases, we might need to make a quick access decision through dialog which might require thinking of ways to reduce the time required to collect attributes and make a correct XACML decision. Maybe asking only the most difficult or important questions according to their weight can reduce the number of questions without affecting the accuracy. An analysis similar to the one in [13] implementing item response theory [14] [15] might be helpful to this work.
- Some policy sets might have a large number of policies and rules with different combination algorithms. It would be interesting to see how this can affect our spoken policy-based access control (question-answer) system.
- After being able to generate dialogues from policies, it would be interesting to see if we can generate rules from dialogues.
- In any dialogue, it is important to guarantee the privacy of what is spoken. In order to answer a question the user has to say things that might be considered to be sensitive (e.g., a unique government identity card number, such as a social security number) and the user might be reluctant to answer the question in public, which might affect the attributes collected in order to build the request and thus the decision might not be correct.

VI. CONCLUSION

In this work, we have presented a novel approach to generate a dialogue for the purpose of physical access control from a standard access control policy language. This policy language driven interaction with the user or authorization requester is generated at runtime and is implemented in a standard language.

ACKNOWLEDGMENT

This material is based upon work supported by the US Air Force Office of Scientific Research under grant FA9550-09-1-0421.

REFERENCES

[1] OASIS, eXtensible Access Control Markup Language (XACML), available at URL: https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml, last accessed Aug. 6, 2012.

[2] World Wide Web Consortium, Voice Extensible Markup Language (VoiceXML)(VXML), available at URL: http://www.w3.org/Voice/, last accessed Aug. 6, 2012

[3] World Wide Web Consortium, Speech Grammar Recognition Specification, white paper available at URL: http://www.w3.org/TR/speech-grammar/, last accessed Aug. 6, 2012.

[4] J. E. Hopcroft and J. D. Ullman, Introduction to Automata Theory, Languages, and Computation, Addison-Wesley, 1979.

[5] Thomas Massie and Duminda Wijesekera, “TVIS: Tactical Voice Interaction Services for Dismounted Urban Operations,” in Proceeding of Military Communications Conference, IEEE, pp. 258-264, San Diego, Calif., Nov. 17-19, 2008.

[6] URL: http://www.Voxeo.com, last accessed Aug. 6, 2012.

[7] URL: http://www.wso2.com, last accessed Aug. 6, 2012.

[8] OASIS, Web Services, available at URL: https://www.oasis-open.org/standards

[9] W3C, Simple Object Access Protocol (SOAP), available at URL: http://www.w3.org/TR/soap

[10] Chongshan Ran and Guili Guo, “Security XACML Access Control Model Based On SOAP Encapsulate,” International Conference on Computer Science and Service System, IEEE, pp. 2543-2546, Nanjing, China, 27-29 June 2011.

[11]Ardagna, C.A., De Capitani di Vimercati, S., Paraboschi, S., Pedrini, E., Samarati, P., Verdicchio, M. , “Expressive and Deployable Access Control in Open Web Service Applications,” IEEE Transactions on Services Computing, Volume 4, Issue 2, pp. 96-109, April-June 2011.

[12] OASIS, Security Assertion Markup Language (SAML), SAML 2.0 profile of XACML v2.0, available at URL: http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-saml-profile-spec-os.pdf

[13] Ahmed A. L. Faresi and Duminda Wijesekera, “Preemptive Mechanism to Prevent Health Data Privacy Leakage,” in Proceedings of International Conference on Management of Emergent Digital EcoSystems, ACM, pp. 17-24, San Francisco, Calif., Nov. 21-23, 2011,

[14] F. B. Baker, The basics of item response theory, ERIC Clearinghouse on Assessment and Evaluation, 2001.

[15] F. B. Baker, Item response theory: Parameter estimation techniques, vol. 176, CRC, 2004.

AUTHOR BIOGRAPHIES

MOHAMMAD ABABNEH is a doctoral student in the Information Technology program at George Mason University.

DUMINDA WIJESEKERA is an Associate Professor of Computer Science at George Mason University, where he serves as Program Director of the Doctor of Philosophy program in Information Security and Assurance. He is a member of the university’s Center for Security Information Systems.

JAMES BRET MICHAEL is a Professor of Computer Science and Electrical and Computer Engineering at the Naval Postgraduate School, Arlington, Virginia.

A Semantic Approach to Evaluate the Impact of Cyber Actions on the Physical Domain

Alexandre de Barros Barreto
Instituto Tecnológico de Aeronáutica
São José dos Campos, SP, Brazil
Email: adebarro@c4i.gmu.edu

Paulo Cesar G. Costa
George Mason University
Fairfax, VA, USA
Email: pcosta@gmu.edu

Edgar T. Yano
Instituto Tecnológico de Aeronáutica
São José dos Campos, SP, Brazil
Email: yano@ita.br

Abstract—**Evaluating the impact that events within the cyber domain have on a military operation and its critical infrastructure is a non-trivial question, which remains unanswered so far in spite of the various research efforts addressing it. The key issue underlying this question is the difficulty in correlating cyber and physical behaviors in an integrated view, thus allowing for real-time analysis. This paper addresses the issue with the development of an ontology-based framework in which the cyber and physical behaviors are integrated in a consolidated view, using a combination of open standards protocols and semantic technologies. In our approach, the mission and its physical aspects are modeled using a business process language (e.g., BPMN) and an information infrastructure based on Simple Network Management Protocol (SNMP). In this scheme, changes in the environment are captured using the output of sensor components existing in the infrastructure. In order to ensure a complete and integrated analysis of the accruing data, we have developed a Cyber Situation ontology (in OWL) and a methodology for mapping the cyber and the physical domains. In this framework, mission data from the environment is retrieved and fused using an engine based on the Semantic Web Rule Language (SWRL). The output of this process is then presented to an analyst in a way that only the most important information needed to support his/her decisions is shown. To validate our approach, a real air traffic scenario was modeled and many simulated flights were generated to support of our experiments.**

I. Introduction

With the increasing automation of processes and systems that are part of critical infrastructures supporting military and vital civilian operations, the cyber domain became one of most important aspects in strategic planning.

Society's dependence on this domain [1] has reached a point in which it is now considered as a new dimension of war, together with air, land and sea. In this new paradigm, a key aspect is to understand how actions performed in the cyber domain (space and time) affect the operations taking place in the other domains, so one can leverage actions in the cyber domain as tools to achieve the campaign objectives [2], [3]

Unfortunately, this is no trivial task, since it requires correlating cyber and physical behaviors in an integrated view that allows tasks to be evaluated in real time. The complexity embedded in this requirement implies, among other things, that an IT manager supporting critical infrastructures must be able to access all relevant data pertaining to the network and translate it to the support team in a way that allows them to understand the real impact of cyber threats to the network and what it means to the overall mission. Existing tools and methodologies cannot provide this level of information, and are not suitable to support complex cyber threat assessment in real situations. This is a major gap that to our knowledge has not been successfully filled, in spite of the relatively large body of research focused on the subject.

This paper addresses this gap by proposing a semantic framework that fuses physical and cyber data collected from existing sensors and retrieving information that is relevant to the assessment of cyber impact. It is designed to support analysts with an integrated view, one that correlates actions in the cyber domain with effects in other domains, allowing the evaluation of its impact on the operational objectives.

The proposed framework and its main aspects are illustrated and evaluated via a simulated air traffic scenario, which includes a large number of simulated flights.

This paper is organized as follows. Section II presents the main concepts necessary to understand the framework being proposed, as well as a sample of the most relevant approaches attained so far to address the problem. Section III describes the framework for evaluating the impact of a cyber attack on an operation occurring in the physical domain. The approach is discussed in Section IV, and illustrated with an analysis of a fictitious air traffic scenario build specifically to evaluate our research. Finally, Section V presents a few considerations and issues that must be addressed in future research aimed to improve the approach.

II. Background and Related Research

The main concept to present is *mission*. As discussed in [4], a mission is the task (or set of tasks), together with its (their) associated purpose, that clearly indicates the action to be taken assigned to an individual or unit.

Three other important concepts are *Situation Awareness*, *Impact Assessment* and *Threat Assessment*. The first, as described in [5], is the perception of the elements of the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status into the near future to enable decision superiority.

The second important concept, Impact Assessment, involves the task of estimating the effects on situations of planned or estimated/predicted actions by the participants, including interactions between action plans of multiple players [6].

The third and last concept, Threat Assessment, can be understood as an expression of intention to inflict evil, injury, or damage. The focus of threat analysis is to assess the likelihood of truly hostile actions and, if they were to occur, projected possible outcomes [6].

From a general perspective, the second and third concepts can be seen as being part of the first, but with a difference in their focus. More specifically, while impact assessment looks for an "internal" understanding (i.e., what is happening and why should I care?), threat assessment seeks the same understanding from the enemy's viewpoint (i.e., how they can hurt us). More important to our research is the fact they all these concepts imply a means to assess the mission. In other words, all must go through the process of specifying and maintaining a reasonable degree of confidence in mission success, which is linked to the concept of Mission Assurance [7].

Literature on the subject of measuring effectiveness of a mission points to two major approaches. The first is to use the concept of *task* as the evaluation basis, while the second instead focuses to the *effects* [8]. The framework presented in this paper adopts the second approach.

The main approach to provide mission understanding involves using a set of distributed sensors to detect intrusions and to uncover attack paths. The preliminary research on the subject is due to Denning [9] and Bass [10]. Schneier [11] proposed the use of an attack-tree to measure effect, which allows understanding of the relationships between attacks, as well as how one attack over a cyber asset affects other assets. In spite of the advances above cited, the problem of determining the impact of a cyber attack on a (mission) task still persists, since no methodology exists to effectively map cyber assets to tasks. Furthermore, these techniques are not capable of dealing with some common types of cyber attacks, rendering them unsuitable for impact assessment in the current state of the art in cyber warfare. For instance, when an attack is new (e.g. a zero-day attack), its signature is unknown and there will be no attack-tree associated with it. As a result, it will be extremely difficult to identify its attack pattern by the time it occurs.

The above limitation illustrates the need for new approaches. A more comprehensive one would involve identifying attacks, highlighting significant events and then understanding the importance of them in a system [12]. To assess the importance of events, one must understand how the process of planning and implementing a mission works. Topological Analysis of Network Vulnerability (TVA) [13] is meant to provide such understanding. TVA supports an analyst in measuring the impact of a threat through the evaluation of topological aspects of the environment. The main weakness of this approach is the absence of an explicit mapping between the mission and the infrastructure supporting it. As a result, this becomes yet another cognitive burden implicitly assigned to the analyst, a solution that clearly does not scale well with the increasing complexity of the operational environment.

Another related approach can be summarized by the work on Mission-Oriented Risk and Design Analysis (MORDA) [14] and on the Security Optimization Countermeasures Risk and Threat Evaluation System (SOCRATES) [15]. In this approach, all components that exist in the problem (mission, resources and threats) are mapped and used in the analysis. However, the mapping process is very complex and requires continuous iteration with the human analyst (i.e. human-in-the-loop), who needs to provide constant feedback and input to the methodology. As a consequence of its demand for human interaction, this approach tends to be applied in the planning phase, while being less suitable to the more time intensive environment found in real time decision making scenarios.

Another methodology that relates to the problem addressed in this paper is Cyber Mission Impact Assessment (CMIA) [7], [16]. CMIA presents a way to (manually) associate mission and infrastructure, and use the resulting association to support the assessment of mission assurance.

In a typical analytical process using CMIA, each attack is simulated and its associated impact is calculated. Then, all attacks and assets are correlated and the paths with the highest cost are prioritized. The major deficiency of this approach is its inability to evaluate more than one attack simultaneously, which prevents an assessment of the synergistic effect of coordinated attacks. This is a major liability, since in most cases the enemy would attempt to achieve an overall effect with parallel attacks that is much greater than the sum of the isolated effects of these same attacks.

The above mentioned works are a representative subset of current research related to evaluation of the impact of cyber threats, and can thus support the claim that the research problem remains unsolved. In summary, each approach suffers from in at least one of the two issues that can be singled out as the main causes for this situation. The first is the lack of a correlation (and, in some cases, computation) between the main components that are needed for impact assessment, the mission and its supporting infrastructure. The second cause for failures is the inability to provide real-time analysis of these two components and their interactions. The proposed framework is meant to address both, with a unique combination of semantic technologies, operations research, and simulation, which we explain in the next Section.

III. Evaluating the Impact of Cyber Threats

This paper proposes ARGUS, a new Framework that evaluate the impact of a cyber attack on a mission. ARGUS is comprised of four main phases: 1) modeling of mission, 2) modeling of network architecture, 3) collecting cyber and mission information, and 4) developing impact assessment. These phases are depicted in Figure 1.

As implied in the diagram, the core idea within ARGUS is to capture the mission and infrastructure information and consolidate it in an integrated data representation, which allows for a comprehensive analysis to be performed.

A. Modeling of Mission

The first phase in ARGUS involves modeling of mission, which is achieved by the use of a business process language.

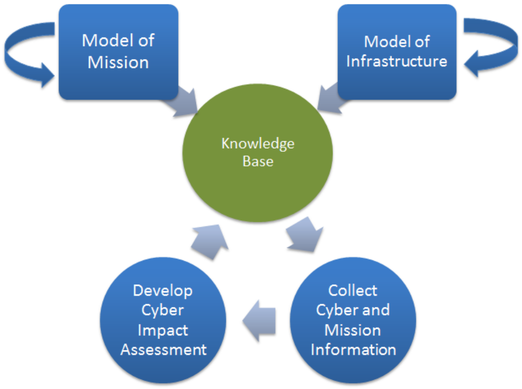

Model of Mission
Model of Infrastructure
Knowledge Base
Develop Cyber Impact Assessment
Collect Cyber and Mission Information

Figure 1. ARGUS major phases

The goal of this phase is to capture the most important information of the mission within the model. Importance here, of course, is measured with respect to its relevance to impact assessment, and includes the tasks, relationships between the tasks, objectives, resources required to develop the mission and, finally, performer (i.e., entity or set of entities that has the responsibility to perform the mission).

In our current research, we leveraged previous experience within our group and made the design decision of capturing these aspects using the Business Process Modeling Notation (BPMN) language [17]. However, any business modeling language with the ability to capture the information described above could have been used and, therefore, might be used with the framework in the future.

One of the most important features of the ARGUS is its reliance on semantic technologies to ensure consistency when used in multiple domains. Therefore, although a business modeling language is used as the basis for information elicitation (BPMN, in the current implementation of the ARGUS), all information captured is stored in an ontology-based information representation repository. The ontology supporting the repository was developed using the most recent version of the W3C recommended OWL 2 Web Ontology Language [18]. In fact, to illustrate the advantages of using an ontology-based framework, it should be emphasized that we didn't have to actually develop a mission ontology from scratch, but we simply imported and made some adaptations to existing work by others. That is, the ontology itself is an adaptation of the one defined in D'Amico et al. [19], while architecture is based on that of Mateus et al. [20].

In our context, the main concept in a mission is activity (see figure 2). An activity has a set of pre and post conditions and one goal. His goal is to produce one or more effects over a resource. An activity can be measure, enabling that can be understand the state of the mission's components.

Due to its main focus on business, BPMN lacks native support for some of the mission information that needed to be captured. Thus, we had to extend its basic structure to accommodate our representational requirements. Figures 2 and 3 illustrate some of the extended attributes (marked with a circle in the figures), which are present in the mission ontology supporting the repository.

The use of a business language (BPMN in the current implementation) was not only convenient as a development tool for the framework, but also proved to be rather suitable for capturing the main aspects of a mission, especially when it is used in civilian environments such as air traffic management, nuclear power plants, and others. Its business-oriented notation made it easier to accommodate the concepts of a mission in the Air Traffic Domain that we are using in the evaluation of the research, while also having a relatively straightforward mapping to the associated concepts in the mission ontology.

One example of a business-oriented concept being mapped to the mission ontology is that of a Pool. To model a mission, an analyst starts by describing the Organizations that participate in the process of accomplishing the mission. These can be squadrons, sectors, departments, battalions, or any functional structure involved with the mission details. Pool is the BPMN concept used to describe such organizations.

We expect the currently developed mapping to be relatively robust when applied along with the framework to other domains. Table I summarizes of the mapping developed in this initial phase of our research.

Table I
MAPPING BPMN TO THE MISSION ONTOLOGY

Concept Source	
Mission Model	**BPMN**
Organization	Pool
System	Lane
Activity	Task
Service	Performer
Condition	Gateway or Event

The ARGUS approach only builds mappings between automated processes, although BPMN is able to support non-automated ones. A service is understood as the entity responsible for performing tasks (activities), while a system is a collection of services. To ensure a proper correlation between business and infrastructure data, the analyst must describe where the service is provided, using his address and ports.

The framework supports the identification of relevant information from raw data captured by the sensors. In order for this to be accomplished, information regarding the effect, conditions and service level are described using rules. More specifically, an *effect* is the result, outcome, or consequence of an *action* (task) over a resource. Further, a *condition* can be understood as the state of the environment or of a situation in which a performer (service) performs or is disposed to perform an task. Finally, *service level* refers to the minimum (or maximum, depending on the requirement) standard that a service is expected to reach with confidence.

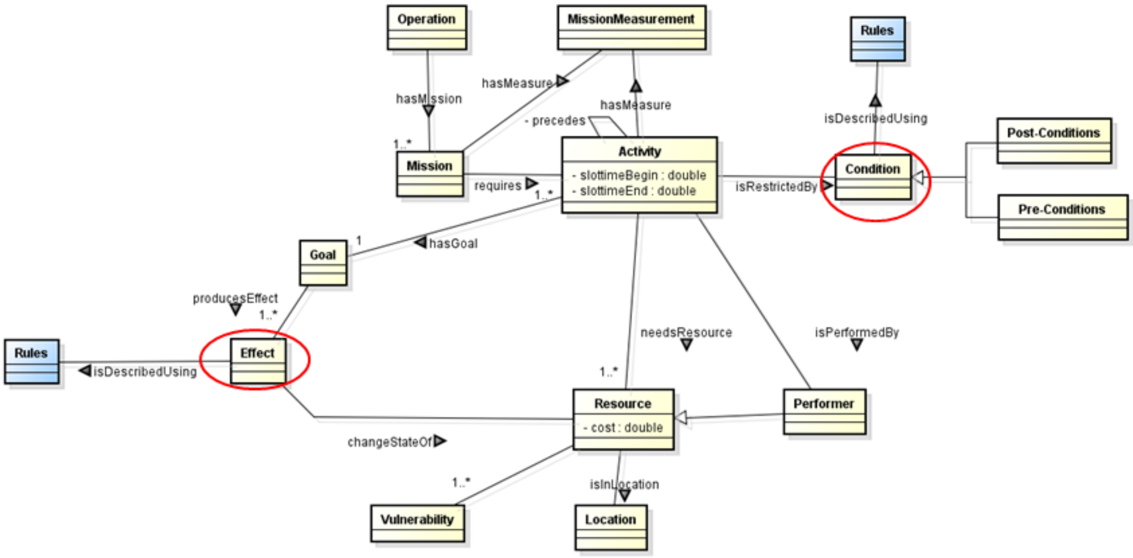


Figure 2. The Mission Ontology

B. Modeling of Network Architecture

The second phase in ARGUS, modeling of network architecture, is in fact performed almost in parallel with the first. In this phase, all information about the infrastructure is captured using Simple Network Management Protocol (SNMP) [21] and stored in the ontology-supported information representation repository. The main concept in the ontology used to represent the infrastructure is Cyber Asset, which is also depicted in Figure 3. Cyber Assets are responsible for to host one or more service (which is who performs the activities needed by the mission). Through services, ARGUS maps the infrastructure in mission and vice versa.

Another important concept from BPMN is that of a *performer*, which was mapped to the mission ontology as *service* (cf. Table I). In BPMN, the performer concept defines the resource that is responsible for an activity. It can be specified in

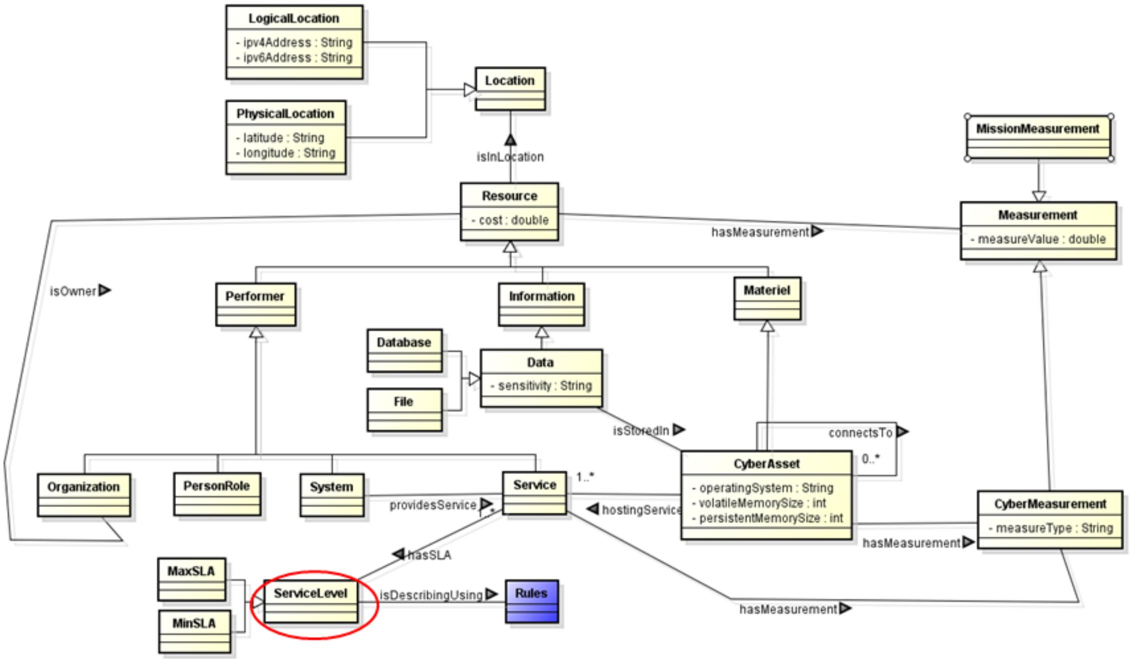


Figure 3. The Resource Ontology

the form of a specific individual, a group, an organization role or position, or an organization. Due to the above mentioned mapping, in ARGUS performers are services, which explains the need for analysts to specify the implementation address during the modeling. In other words, the correlation between the services and the cyber assets is made automatically by the framework via SNMP queries, which collect the UDP/TCP ports of the services via two tables residing in the Management Information Base (MIB) of each of the network hosts (*tcpConnLocalPort* and *udpLocalPort*).

To build the network archiecture and its variations, the framework performs queries on the other three tables residing in each host's MIB, the *ipRouteDest*, the *ipRouteMetric*, and the *ipRouteNextHop*. The combination of the information retrieved from these tables allows the Framework algorithm to infer the neighbors of the host, as well as the network distance between the host and nodes that were eventually discovered via the routing protocol embedded in the framework algorithm. Finally, the framework uses changes in those attributes (e.g. nodes added, nodes deleted, changes in nodes IP route metrics, etc.) as parameters for inferring the network dynamics. Besides the network information mentioned above, the framework also uses SNMP to retrieve a set of other infrastructure properties, such as memory (persistent and volatile) size, operating system, uptime, etc. It is outside the scope of this paper to explain in detail the framework algorithms and how each network parameter is assessed, more information on these details can be obtained from the work at the GMU/ITA C2 testbed (cf. [22]).

C. Collecting Cyber and Mission Information

The third phase in ARGUS involves the collection of relevant information. In this case, the criteria for information to be considered relevant is related to the value it adds to the overall understanding of the environment (i.e. how it improves situation awareness). This assessment is performed in accordance with the general scheme depicted in Figure 4.

The main concept in the scheme is *Situation*, which is an event or set of events that are meaningful to the mission. In ARGUS, events can be captured in any different ways. In our first implementation, we can retrieving the data existing in the SYSLOG Database [23] or by capturing network packets via a packet capture (PCAP) interface (e.g.through an intrusion detection system) [24]. Once an event is captured, the framework uses rules to classify it as being part of a situation. As previously mentioned, these rules will be applied to information retrieved from the network sensors and inserted into the framework through the BPMN's and Ontology's interfaces (cf. Figures 2, 3, and 4).

The design choice for describing the rules was the Semantic Web Rule Language (SWRL) [25]. SWRL extends a set of OWL axioms to include Horn-like rules, thus enabling Horn-like rules to be combined with an OWL knowledge base. The expressiveness achieved by this rule scheme is key to the framework's ability to capture aspects that cannot be easily captured using OWL, such as utilization of resources, mission requirements, and others.

Once all information needed from the business and infrastructure is retrieved, the events are captured from the sensors' input, and classified in accordance with relevant situations using rules. Then the framework is ready to evaluate the impact of the current state of the system on its main mission. In ARGUS, this evaluation is performed through four distinct types of analysis: *dependence paths*, *temporal*, *cost*, and *history degradation*.

The first type of analysis, *dependence paths*, aims to uncover problems in topology that have the potential to affect the accomplishment of the mission. The typical questions involved in this analysis include (but are not limited to) the following:

- In this state of the system, can the mission goal be reached?
- If task C fails, is there any path left to reach the goal?

The second type of analysis, *temporal*, seeks to define a window of interest in which the problem is solvable. The typical questions that are raised in this type of analysis include but are not limited to:

- What tasks need to be monitored at time T?
- How much time is needed to finish the task and accomplish its objective?

The third type of analysis, *cost*, is meant to identify when the cost starts to become a serious threat to the task execution. In other words, it evaluates the cost / benefit ratio of each task with respect to the overall mission. The typical questions to be answered in this analysis include:

- How much does this task cost?
- Do the benefits of this task justify the costs involved in its execution?
- If task C is compromised, does an alternative route have an acceptable cost?

The last type of analysis, *history degradation*, has the goal of understanding how fast the infrastructure is degrading. Its typical questions can be similar to the ones in each of the above tasks, but with a focus on the way the infrastructure assets are degrading and its associated impact on the overall mission.

D. Developing Impact Assessment

The fourth phase in ARGUS, *impact analysis*, is the main part of the framework. In order for this phase to be executed in real time, so the impact evaluation would be done as the mission unfolds, we have developed the reference implementation depicted in Figure 5.

The Cyber Situation Awareness engine (CyberSA Engine) is comprised of six modules. The first is the BPMN Module, which performs the tasks of getting mission information from a BPMN file, parsing it, and mapping the retrieved concepts to the mission ontology.

The SNMP and SYSLOG modules perform queries on all hosts and on the SYSLOG Server, respectively. When the associated answers are received, the module parses and converts them to the format they will be used in the system.

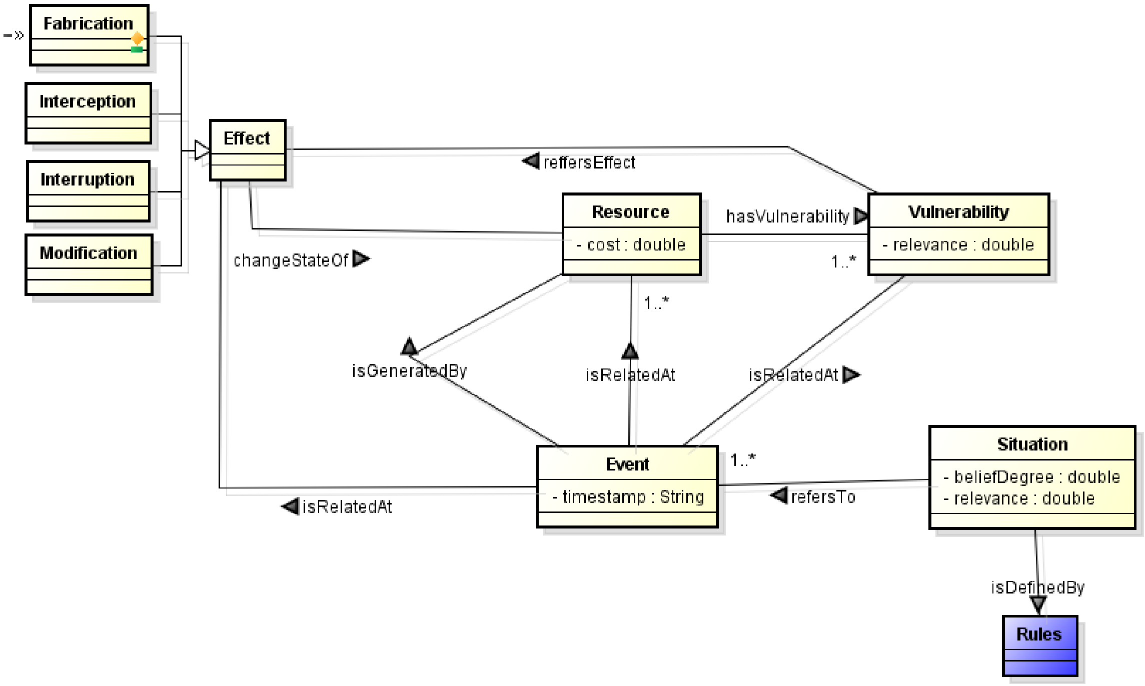


Figure 4. Capturing the Details of an Event

The PCAP module retrieves event data from the network. However, analysing the retrieved raw data is a time consuming and non-trivial task, so in our implementation we have made the design decision of using an external tool, TSHARK [26]. This tool is a terminal-oriented version of Wireshark designed for capturing and displaying packets when an interactive user interface is not necessary or not available. It has a set of filters that produces information in a format that is more readable to analysts.

Once the four modules above collect and process their respective information, the result needs to be made available in a consistent way so the CyberSA Engine can provide it to the users. This consistency is also achieved with the support of semantic technologies, via the implementation of a Semantic Fusion Module. The main services this module provides are making inferences and applying rules, which were written by analysts using the GUI.

The Semantic Fusion Module uses two libraries to provide its features. The first is the OWL-API [27], a Java API and reference implementation for creating, manipulating and serializing OWL Ontologies. The second is Pellet [28], which is an OWL 2 reasoner that provides standard and cutting-edge

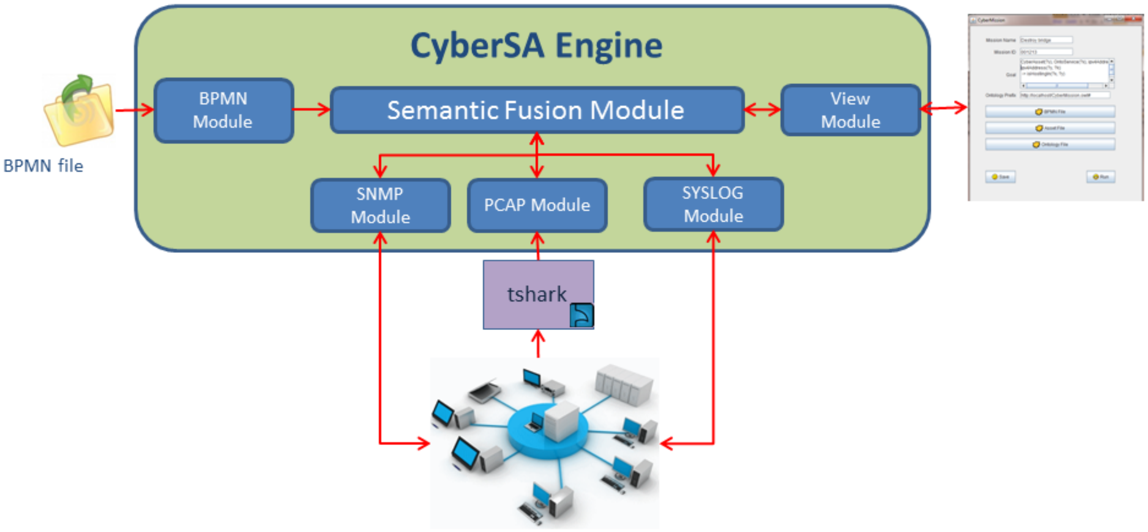
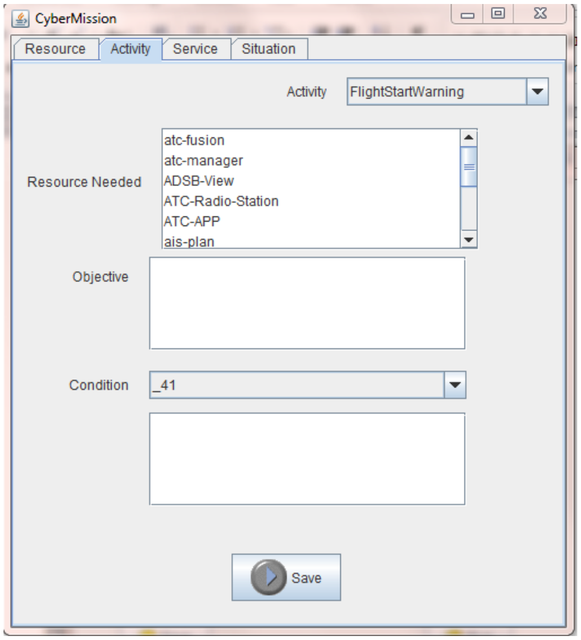


Figure 5. The CyberSA Engine

reasoning services for OWL ontologies.

The last module of the CyberSA Engine is the View Module, which provides the interface to analysts. The main goals of this interface are to allow analysts to provide information the system cannot obtain automatically, and to write the rules used by the system's inference engine.

Figure 6 is an example of a typical form of the system's GUI, in this case one that allows the analyst to setup a task. In the combo box depicted in the figure (named as "Activity"), the analyst chooses the type of activity he wants to set, as well as the associated fields - which are shown in a contextual fashion with support from the mission ontology. In the example, the analyst chose the activity "FlightStartWarning", and was then presented with three fields. In the first field, the analyst is presented with the resources that he needs to do the task. In the remaining two fields, the analyst is expected to describe, using rules in SWRL syntax, how to measure the task progress and the conditions this measure will be performed.



Activity: FlightStartWarning

Resource Needed:
atc-fusion
atc-manager
ADSB-View
ATC-Radio-Station
ATC-APP
ais-plan

Objective

Condition: _41

Figure 6. The ARGUS User Interface

By means of this GUI, the system will guide the analyst through a process in which he will be able to define the activity, the cost of resources, the service's SLA, and other rules that must be defined given the relevant situations. The View Module also provides classification of the event (i.e. the situation(s) it pertains to).

IV. Discussion

A simulation of an air traffic scenario was developed to evaluate the framework, verifying its ability to generate the relevant situation assessment and present it to the analyst. The simulation is based on a real scenario, located at the Campos basin in Brazil, where a heavy helicopter operation is held to support maritime oil platforms sixty to eighty miles offshore. The mission described in this scenario thus involves air traffic service where the aircraft consume the smaller amount of fuel and the system generates a low number of collision resolution events. A collision resolution event happens when two aircraft fly within a distance (vertical or horizontal) that is smaller than the safety rules defined by law.

The simulation includes three distinct air traffic services organizations (cf. Figure 7). The first is the AIS (Aeronautical Information Service), which has the responsibilities of inserting the flight plan into the system and getting all clearance necessary for the aircraft to fly. The second service modeled is the Radio Station, which gets information on flight tracks (i.e. aircraft) within its area of coverage and sends it to the APP (Ground-controlled Approach) Service. Finally, the APP service performs three main tasks: fuse track information, present it in a controller view and generate alerts to be used by a monitoring system.

The simulation was developed using the C2 Simulation Testbed [22], a joint project between the C4I Center at George Mason University (GMU) and the C2 Lab at the Instituto Tecnológico de Aeronáutica (ITA) in Brazil. The testbed allows the emulation of any infrastructure behavior and the simulation of all aspects of the physical environment (aircraft flights, collisions, etc). The current evaluation scenario includes fourteen aircraft that take off from three different airports and go to the oil platforms. The flight plan was developed to generate collision warnings, allowing the framework to generate situations of interest. A view of this scenario using the C2 Simulation Testbed is presented in Figure 7.

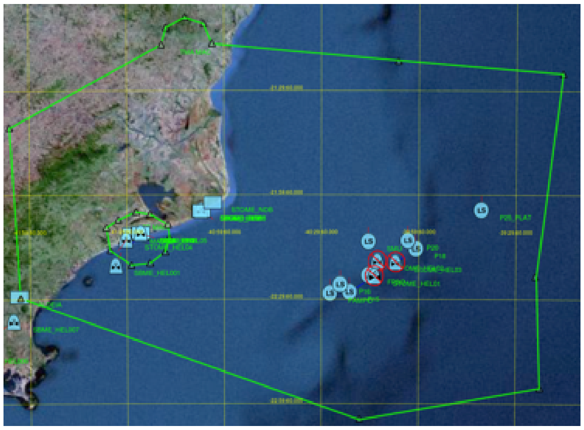


Figure 7. The Simulation in VRForces

A major aspect that is needed for the framework to define relevant situations is the proper definition of the rules by analysts. Among other things, these rules formally establish to the system the conditions that restrict the task, the goal of mission in general, the objective of each task, and other aspects that are important in filtering the raw data coming from the sensors. In addition to these aspects, another key use of rules is to create relations that are not explicit in the domain. As an example, the link between cyber assets and services can be defined by this simple rule:

$CyberAsset(?y), OntoService(?x), ipv4Address(?x, ?k), ipv4Address(?y, ?k) \rightarrow isHostingIn(?x, ?y)$. Therefore, it is fair to say that the combination of SWRL rules and OWL 2 statements to link the physical and cyber domains is at the heart of the system's goal of evaluating mission impact.

V. Future Research

This paper presented an approach for connecting the cyber and physical domains, with the objective of assessing the impact that actions in the former have in the latter. This is research in progress in an area where clear answers are usually not attainable, mostly due to the complexity as well as to the level of subjectivity involved in real time impact assessment. As such, the framework presented here should be seen as a first step of a steep ladder. Yet, it is a firm step, since after attempting various approaches we remain convinced that the solution to this problem relies in a combination of techniques where semantic technologies and simulation play a major role.

The software modules, including the ontology and some of the rules, that together comprise the framework are already implemented, and we are currently evaluating its performance via the C2 Simulation Testbed. Preliminary results are promising and should be available soon. Our future work path includes aspects such as the usability of the system, and others that rely on semantic technologies to alleviate the reliance on analysts to provide domain knowledge in the form of SWRL rules.

Acknowledgment

The authors recognize VT MÄK and Scalable Network Technologies for providing the tools and support needed to develop the Testbed. Our gratitude is also extended to Latin-Media SA, who provided support for the testbed in Brazil.

References

[1] G. Eason, B. Noble, and I. N. Sneddon, "On Certain Integrals of Lipschitz-Hankel Type Involving Products of Bessel Functions," *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, vol. 247, no. 935, pp. 529–551, Apr. 1955. [Online]. Available: http://rsta.royalsocietypublishing.org/content/247/935/529

[2] M. Endsley, "The application of human factors to the development of expert system for advanced cockpits." in *Annual Meeting of Human Factors and Ergonomics Society*. Human Factors Society, 1987, pp. 1388–1392.

[3] J. Boyd, "OODA Loop." Center for Defense Information, Tech. Rep., 1995.

[4] DoD, *DODAF. DoD Architecture Framework Version 2.0 - Volume 1: Introduction, Overview, and Concepts.*, DoD Std., 2009.

[5] J. Salerno, M. Hinman, and D. Boulware, "A situation awareness model applied to multiple domains," in *Proceedings of SPIE*, vol. 5813, 2005, p. 65.

[6] E. Bosse, J. Roy, and S. Wark, *Concepts, models, and tools for information fusion*, A. House, Ed. Artech House, 2007 2007.

[7] S. Musman, M. Tanner, A. Temin, E. Elsaesser, and L. Loren, "A systems engineering approach for crown jewels estimation and mission assurance decision making." in *IEEE Symposium on Computational Intelligence in Cyber Security (CICS)*, 2011.

[8] M. J. Fiebrandt, C. Mills, and T. Beach., "Modeling and simulation in the analysis of a joint test and evaluation methodology," in *Spring Simulation Multiconference*, vol. 3. Society for Computer Simulation International, 2007, pp. 251–256.

[9] D. E. Denning, "An intrusion-detection model," *IEEE Transactions on Software Engineering*, vol. 13, pp. 222–232, 1987.

[10] T. Bass, "Multisensor Data Fusion for Next Generation Distributed Intrusion Detection Systems," in *IRIS National Symposion*, 1999.

[11] B. Schneier, "Attack trees: Modeling security threats," Dr. Dobb's journal, December 1999.

[12] O. S. Saydjari, "Cyber defense: Art to Science." *Communications of the ACM - Homeland Security*, vol. 47, no. 3, March 2004.

[13] S. Jajodia, S. Noel, and B. O'Berry, "Topological Analysis of Network Attack Vulnerability." *Managing Cyber Threats*, vol. 5, pp. 247–266, 2005.

[14] S. Evans, D. Heinbuch, E. Kyle, J. Piorkowski, and J. Wallner, "Risk-based systems security engineering: stopping attacks with intention," *IEEE Security and Privacy*, vol. 2, pp. 59–62, 2004.

[15] D. L. Buckshaw, G. S. Parnell, W. L. Unkenholz, D. L. Parks, J. M. Wallner, and O. S. Saydjari, "Mission Oriented Risk and Design Analysis of Critical Information Systems," *Military Operations Research*, vol. 2, pp. 19–38, 2005.

[16] S. Musman, M. Tanner, A. Temin, E. Elsaesser, and L. Loren, "Computing the impact of cyber attacks on complex missions." in *2011 IEEE International Systems Conference (SysCon)*, 2011, pp. 46–51.

[17] OMG, *Business Process Model and Notation (BPMN) 2.0*, http://www.omg.org/spec/BPMN/2.0, OMG Std., 2011.

[18] W3C, *OWL 2 Web Ontology Language*, http://www.w3.org/TR/owl2-overview/, W3C Std., October 2009.

[19] A. D'Amico, L. Buchanan, J. Goodall, and P. Walczak, "Mission Impact of Cyber Events: Scenarios and Ontology to Express the Relationships between Cyber Assets, Missions, and Users." AFRL/RIEF, Tech. Rep. OMB No. 0704-0188, December 2009.

[20] C. J. Matheus, M. M. Kokar, K. Baclawski, J. A. Letkowski, C. Call, M. Hinman, J. Salerno, and D. Boulware, "SAWA: An assistant for higher-level fusion and situation awareness," *Proceedings of SPIE*, vol. 5813, no. 1, pp. 75–85, 2006. [Online]. Available: http://link.aip.org/link/?PSI/5813/75/1&Agg=doi

[21] J. Case, M. Fedor, M. Schoffstall, and J. Davin, *A Simple Network Management Protocol (SNMP)*, The Internet Engineering Task Force (IETF) Std. RFC 1157, May 1990. [Online]. Available: http://www.ietf.org/rfc/rfc1157.txt

[22] A. B. Barreto, M. Hieb, and E. T. Yano, "Developing a Complex Simulation Environment for Evaluating Cyber Attacks," in *I/ITSEC*. I/ITSEC, 2012, will be published in I/ITSEC 2012.

[23] R. Gerhards, *The Syslog Protocol*, http://tools.ietf.org/html/rfc5424, IETF Std. rfc5424, March 2009.

[24] E. Nemeth, G. Snyder, S. Seebass, and T. Hein, *UNIX System Administration Handbook*. Prentice Hall, 2000.

[25] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean, "SWRL: A Semantic Web Rule Language Combining OWL and RuleML," http://www.w3.org/Submission/SWRL/, W3C Member Submission, May 2004.

[26] Wireshark, "Wireshark," http://www.wireshark.org/, 2012.

[27] M. Horridge and S. Bechhofer., "The OWL API: A Java API for OWL Ontologies." *Semantic Web Journal 2(1), Special Issue on Semantic Web Tools and Systems,*, pp. 11–21, 2011.

[28] "Pellet: OWL 2 Reasoner for Java," http://clarkparsia.com/pellet/, 2012.

D2RCrime: A Tool for Helping to Publish Crime Reports on the Web from Relational Data

Júlio Tavares
University of Fortaleza - UNIFOR
Fortaleza/CE, Brazil
julio.at@gmail.com

Vasco Furtado
University of Fortaleza - UNIFOR
Fortaleza/CE, Brazil
furtado.vasco@gmail.com

Henrique Santos
University of Fortaleza - UNIFOR
Fortaleza/CE, Brazil
hensantos@gmail.com

Eurico Vasconcelos
University of Fortaleza - UNIFOR
Fortaleza/CE, Brazil
euricovasconcelos@gmail.com

Abstract—**In the Law Enforcement context, more and more data about crime occurrences are becoming available to the general public. For an effective use of open data, it is desirable that the different sources of information follow a pattern, which allows reliable comparisons. In addition, it is expected that the task of creating a correspondence between the pattern and the internal representations of each source of information is not a steep learning curve. These two conditions are hardly found in the actual stage, where open data about crime occurrences refer to the data disclosed by each police department in its own way. This paper proposes an interactive tool, called D2RCrime, that assists the designer/DBA of relational crime databases to make the correspondence between the relational data and the classes and properties of a crime ontology. The ontology plays the role of a pattern to represent the concepts of crime and report of crime, and is also the interface to publish on-the-fly relational crime data. This correspondence allows the automatic generation of mapping rules between the two representations, what allows for access to relational data from SPARQL. An evaluation of D2RCrime is done with DBA/system analysts who used the tool for establishing correspondences between relational data and the ontology.**

Index Terms—**Internet, Semantic Web, Knowledge Engineering, Law Enforcement, Open Government.**

I. Introduction

The culture of participation and collaboration on the Web could not leave out the public sector. New forms of relationships between citizens and governments are also emerging from a new attitude on the tract of government information and public service on the Internet. This new approach, understood here as Government 2.0 (while complying with the Web 2.0), relies on governments as open platforms to provide information [1].

In the Law Enforcement context, more and more data about crime occurrences are becoming available to the general public. In the U.S. and Britain in particular, police departments quickly realized that they should open data to encourage participation by the population. For an effective use of open information, it is desirable that the different sources of information follow a pattern, which allows, for instance, making reliable comparisons. Here, when we mention a pattern, we refer to a language with the power to represent information about both the provenance and the meaning of the concepts that should be available. Moreover, it is expected that the task of creating a correspondence between the pattern and the internal representations of each source of information is not a steep learning curve. These two conditions are hardly found in the actual stage in the context of opening data about crime occurrences. The usual process is each police department to define its own way to disclose its data by creating intermediary representations (typically spreadsheets[1]) that must constantly be updated. Alternatively, the police departments develop their own APIs[2] that are characterized by their specificity. In brief, each department spends time and resources to define its own way to disclose its data.

This paper proposes a method to guide the process of opening crime data that aims to mitigate the aforementioned problems. This method relies on ontologies for representing the concepts of crime and crime report. The *crime* ontology defines the basic concepts and properties used in the context of Law Enforcement to define a crime occurrence. The *crime report* ontology defines the basic information necessary to characterize the report of a crime occurrence such as the source of the report, the date and time of the report, its description, and so on.

We have designed an interactive tool that assists the designer/DBA to make the correspondence between the relational data and the classes and properties of the crime ontology. This correspondence allows us to automatically generate the mapping rules between the two representations, which conducts the process of accessing relational data from SPARQL. Unlike the majority of approaches that replicate the relational data into another repository, we based our proposal

[1] See http://www.atlantapd.org/crimedatadownloads.aspx in Atlanta
[2] See http://sanfrancisco.crimespotting.org/api for San Francisco

on the D2R Server [2]. D2R is a system for publishing relational data on the Web. The D2R Server enables Resource Description Framework (RDF) and HTML browsers to navigate the content of non-RDF databases, and allows applications to query a database using the SPARQL query language over the SPARQL protocol. This approach relieves the data owner of concerns about the integrity and consistency of the replicated data. Finally, an evaluation of D2RCrime is done with DBA/system analysts who used the tool for establishing correspondences between relational data and the ontology.

II. REPRESENTING CRIME REPORTS

Two ontologies are at the core of our proposal. They intend to represent the concepts of crime and report of crime. Our representation of crime is not restricted to the information that nowadays has been disclosed by police departments worldwide. However some information is mandatory to define a unique instance. A crime has at least a type, a date and time (imported from the time ontology [3], a precise address (geographical coordinates), and a description. Information about the people involved such as the perpetrator(s), the witnesses and the victim(s) may also be inserted, but it is not mandatory.

The crime ontology is basically a hierarchy for inferential purposes. It was modeled so that it is possible to map the various classifications of crime type. We define the crime events as specializations of the Event class, from the Event Ontology [4]. According to the Event Ontology, “an event is an arbitrary classification of a space/time region, by a cognitive agent. An event may have a location, a time, active agents, factors and products.” To describe where a crime occurred geographically, we use the ontology wgs84[3] to express location in terms of latitude and longitude.

Typically, a detailed identification of the people involved is not open information due to privacy concerns. However, this varies according to different countries, sources and cultures. In Brazil, for instance, the media naturally discloses homicide victims. In the US, raw crime data does not include the victim’s name.

We defined a crime ontology inspired by the Criminal Act Ontology in the context of the OpenCyC Project, and also took into consideration the FBI Uniform Crime Report[4] standard. The report of crime refers to a particular crime and has information about the reporting itself. The identification of the reporter, the time and date of the report, and links to external sources are examples of this kind of information. As a report of crime contains basic provenance information, in order to represent these latter features, we imported the Provenance Model Language 2 (PML2) ontology [5]. Even though the Open Provenance Model (OPM) [6] and its Open Provenance Model Ontology (OPMO) are becoming widely used for provenance exchange, we have chosen to use PML2 because it includes classes and properties to represent the trustworthiness of the sources and credibility of the information. These properties are important because our ultimate goal is to combine crime open data from a large variety of sources that sometimes can even be anonymous. The *CrimeReport* class is a subclass of *pmlp:Information*. We have also used some specific properties to describe a report, such as *pmlp:hasCreationDateTime* (hour of the report), *pmlp:hasDescription* (text of the report), and pmlp:hasSource (entity that published the report).

The complete ontology is described in [15]. Figure 1 shows a piece of this ontology describing a particular crime (homicide). This is the most refined level of detail that we have proposed. Doing so, we aim to keep the tradeoff between simplicity and generality while providing good coverage.

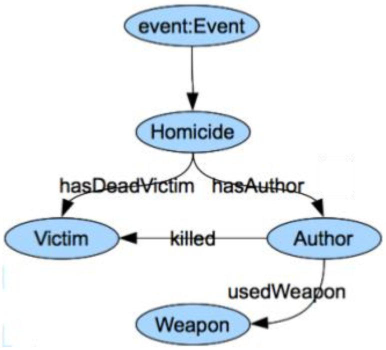


Fig. 1. Piece of the crime ontology for the description of homicide

III. ASSISTING THE MAP BETWEEN RELATIONAL DATA AND THE CRIME ONTOLOGY

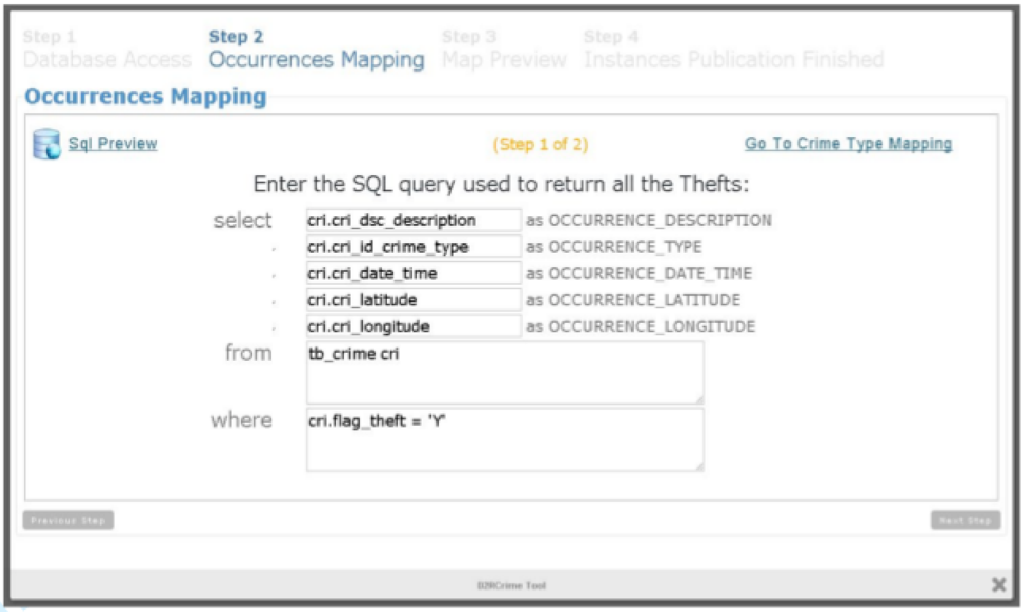
The definition of a language to be used as a pattern for opening data on criminal incidents is only the first step of the proposed method. Patterns require community acceptance, therefore a key aspect is how friendly the use of the pattern is. Thus it is essential that the correspondence between information represented in the pattern and information represented in the databases of the police departments be easily established. In this section we describe how the proposed method seeks to accomplish this. It relies on two assumptions i) as crime data are originally stored in relational databases, the Web publication thereof should not require data replication, and ii) the task of associating the original data with the ontology should not require learning another programming language.

A. Publishing Relational Data on the Web

To achieve the first requirement, we have chosen to base our method on systems that map relational data to RDF on-demand such as Asio Semantic Bridge for Relational Databases[5], D2R[6] [2], SquirrelRDF[7], and UltraWrap[8] [7]. In these methods, an application (typically a Web server) takes requests from the Web and rewrites them to SQL queries. This on-the-fly translation allows the content of large

[3] http://www.w3.org/2003/01/geo/
[4] http://www.fbi.gov/about-us/cjis/ucr/ucr

[5] http://www.bbn.com/technology/knowledge/asio_sbrd
[6] http://www4.wiwiss.fu-berlin.de/bizer/d2r-server/
[7] http://jena.sf.net/SquirrelRDF
[8] http://www.cs.utexas.edu/~miranker/studentWeb/UltrawrapHomePage.html



Enter the SQL query used to return all the Thefts:

```
select  cri.cri_dsc_description    as OCCURRENCE_DESCRIPTION
      , cri.cri_id_crime_type      as OCCURRENCE_TYPE
      , cri.cri_date_time          as OCCURRENCE_DATE_TIME
      , cri.cri_latitude           as OCCURRENCE_LATITUDE
      , cri.cri_longitude          as OCCURRENCE_LONGITUDE
from    tb_crime cri
where   cri.flag_theft = 'Y'
```

Fig. 2. Example of a SELECT clause to define the concept of THEFT

databases to be accessed with acceptable response times without requiring data replication.

The World Wide Web Consortium (W3C) has recognized the importance of mapping relational data to the Semantic Web by starting the RDB2RDF incubator group (XG) to investigate the need for standardization. In particular, we have chosen to use an approach based on the D2R server. D2R is an open and free system for publishing relational data on the Web. It enables RDF and HTML browsers to navigate the content of non-RDF databases, and allows applications to query a database using the SPARQL query language over the SPARQL protocol.

The operation of D2R is through the interpretation and execution of rules, described in the Data to Relational Query language (D2RQ [8]), for mapping the equivalence between an ontology and a relational database.

D2RQ consists of a mapping language between relational database schema and RDFS/OWL ontologies. The D2RQ platform creates an RDF view of the relational database, which can be accessed through Jena, Sesame, and the SPARQL query language. D2RQ's main elements are *ClassMap* and *PropertyBridge*. The *ClassMaps* represent the classes of an ontology and associates them with a table or a view of a database. The *PropertyBridges* are linked to one or more *ClassMaps* and are mainly used to connect the columns in a table with the properties (attributes) present in an ontology. Usually, they are filled with literal values, but can also make references to URIs that designate other resources.

With *PropertyBridges* it is possible to specify conditional restrictions that can be used to filter a specific domain or range of information. Using the Join structure, it is also possible to specify the mapping between multiple tables and a class or a property in the ontology. Another quite usual feature is the *TranslationTable* structure, which allows 1 to n mapping (table to classes).

The performance of more complex mappings, whereby it may be necessary to access a Web service or to use conditional structures and external sources of data, can be made through the *javaClass* structure, which allows the use of Java classes to perform the mapping.

In practice, it is very difficult to implement mapping just with simple correspondences like one-to-one table to classes. There is often the need to handle more complex structures, including the javaClass, which requires an effort that the designer is not always able to make. For instance, a tuple of a table that describes crime data must be mapped into instances of different classes such as robbery, theft, homicide, etc. Our idea then was to provide a tool that facilitates this process of mapping to the case of criminal data.

B. The D2RCrime Tool

D2RCrime provides resources to support the publication of reports of crimes in RDF, from relational databases. In particular, the goal is to help designers and/or DBA who do not have extensive knowledge in semantic technologies. The ontology of crimes described above is used to guide an interactive process with a designer/DBA. The basic premise is that D2RCrime mapping between the ontology classes and the database tables can be obtained interactively by asking the designer to write SQL queries for retrieving tuples from the database that describe a particular class (or property) of the ontology. The aim is thus to use a language largely dominated by designers/DBA and allows them to easily describe the concepts represented in the ontology of crimes. Figure 2 shows an example of how this dialog occurs in D2RCrime.

It asks the designer to complete a SELECT clause to retrieve all the thefts from the database of crime occurrences (tb-crime in the Figure). The tool also asks that the response contain the date, time, location and description of each theft. For each SELECT clause made by a designer/DBA, D2RCrime transforms the query into an N3 rule. The process is iterative and new questions will be carried out until all the classes and properties of the ontology have been described in terms of SELECT clauses. At the end of the process, the entire mapping is performed using D2RQ and therefore can be executed on the D2R Server. Frame 1 illustrates the mapping between tables and classes. The crime report and theft classes are mapped there.

D2RCrime transforms the SQL into D2RQ elements. To do this, the following mapping is done: Aiming to accelerate the elicitation of the requirements for the mapping, D2RCrime identifies which database field is associated with the type of crime. It then proposes a customized interface in which it is possible to associate the values of crime type with the corresponding ontology classes.

```
// CrimeReport - In the ClassMap below
   it is defined that the instances are
   generated with the class
   "crime:CrimeReport"

map:CrimeReport a d2rq:ClassMap;
  d2rq:dataStorage map:database;
  d2rq:uriPattern "crimereport/
      @@tb_cri_crime.CRI_IDCRIME@@";
  d2rq:class crime:CrimeReport;
  d2rq:classDefinitionLabel "CrimeReport";
  map:CrimeReport__label a
  d2rq:PropertyBridge;

  d2rq:belongsToClassMap map:CrimeReport;
  d2rq:property rdfs:label;
  d2rq:pattern "CrimeReport
  #@@tb_cri_crime.CRI_IDCRIME@@";

// Theft [OCURRENCE_TYPE] -
   In the ClassMap below, it is defined
   that the instances are generated with
   the class "crime:Theft".
   Note the d2rq:condition for
   selecting the adequate type of crime

map:Theft a d2rq:ClassMap;
  d2rq:dataStorage map:database;
  d2rq:uriPattern "Theft/@@tb_cri_crime.
      CRI_IDCRIME@@";
  d2rq:class crime:Theft;
  d2rq:condition "tb_cri_crime.
    tcr_idtipo_crime=1 or
    tb_cri_crime.tcr_idtipo_crime=4";
  d2rq:classDefinitionLabel "Theft";

map:Theft__label a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:Theft;
  d2rq:property rdfs:label;
  d2rq:pattern "Theft #@@tb_cri_crime.
      CRI_IDCRIME@@";
```

Frame 1. Example of the code in D2RQ generated by D2RCrime

During the dialogue process, D2RCrime offers the possibility for the designer to see how the instances of the classes (crime reports) have been built. A widget to plot crimes on the spot where they occurred shows the values of each report. Figure 3 shows an example of this.

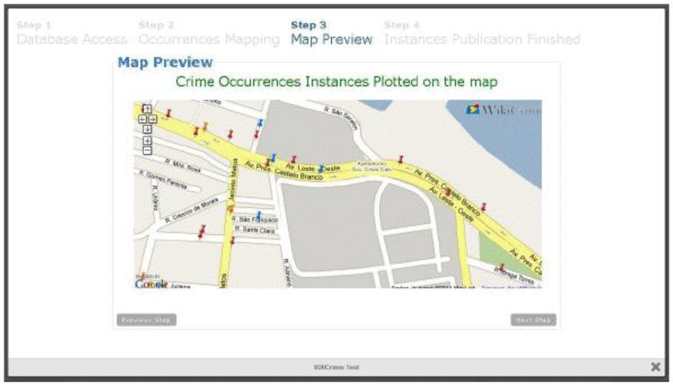


Fig 3 Preview of the instances of crime reports plotted in the map

IV. EVALUATION

Our approach proposes a new method of mapping between relational databases and structured data in RDF. We are not aware of similar tools or approaches that are able to perform the RDF2RDF mapping intuitively using SQL clauses. Because of this, we had difficulty choosing what would be the most appropriate way to validate our hypothesis for the comparison and experiments. To alleviate this issue, we decided to compare D2RCrime with the D2RServer tool itself, which automates the generation of D2RQ code for mapping the relational data into RDF.

In order to analyze the hypotheses raised in this paper, an empirical study was conducted aimed at assessing: 1) the representational power of the proposed ontology to represent criminal events; 2) whether the task of creating correspondence by means of the proposed tool is not actually a “steep learning curve” and whether the tool is user friendly and intuitive, enabling and facilitating the proposed mapping process.

A. Methodology

The study was conducted in two stages. In the first stage, a battery of tests of “translation” of information on crimes was conducted in the laboratory, based on the proposed ontology. The battery was based on non-probabilistic and intentional samples (50 each) from police agencies. The choice of samples was based on two factors: the requirement that the police agencies have their information about crimes published, and the interest in evaluating the ontology in different countries (criminal law) and in different languages.

In the second stage, tests were conducted with users to analyze whether the D2RCrime tool softens the “steep learning curve” found in the data-opening process. For such, a sample of 10 users — 5 analysts and five DBAs, all with experience in DBMSs and SQL language — were invited to publish data on crimes in two sessions.

The first session used the D2RCrime tool in conjunction with the proposed ontology. The second session was conducted without introducing the tool, encouraging users to perform the publication without support of the tool. To do so, we used the automatic mapping generation resource (generate-mapping) available in the D2RServer software. This procedure automatically generates a mapping file expressed in D2RQ language, which reflects the structure of the relational database to be mapped.

All the users who took part in the tests had good knowledge on SQL language and little or no knowledge on semantic technologies, representing the scenario usually found in an IT staff. The proposed method takes this fact into account, utilizing the System Analysts’ and DBAs’ prior knowledge in SQL and not exposing them to the need to learn the set of tools required for publishing content on the Semantic Web.

As a methodology for performing the test, users were exposed to a document with different data models, which were aimed at representing the tables related to the storage of criminal occurrences. Thus, different data modeling was distributed among the user groups, so that there would be a significant representation of the main scenarios found in the databases of police departments. The use of different models was aimed at assessing the generality of this approach. The following performance factors were used for the tests conducted:

1) Success in the mapping activities, which indicates whether it was possible to complete the mapping test within the allotted time (30 minutes);

2) RDF Mapping, which reflects the quantity of concepts and properties of the ontology that were successfully mapped to RDF for those users who finished the tasks (item 1);

3) Correctness of the generated vocabulary, which reflects whether the published data met the main concepts described in the ontology;

4) Autonomy which is the number of users that have finished the activities without human guidance at the time (only with the specification of the activity).

B. Results: Ontology Coverage

As mentioned before, the proposed crime ontology was based on the current initiatives of open crime data. For the purpose of evaluating the completeness of the ontology coverage, we compared the concepts represented therein with four samples of crime datasets in different countries: Oakland, US; FBI, US; London, UK; and Fortaleza, BR. A table describing the main concepts used in this comparison is available at http://www.wikicrimes.org/ontology/table.htm. In general the main concepts were correctly mapped. Most of the types of reports open to the public refer to crimes against property (robbery, thefts, burglary, etc.) and crimes against life (murder, attempted murder, etc.). Problematic cases refer to types of crimes that are generic, such as “anti-social behavior“ or “disturbing the peace.” Typically this involves several types of crimes that differ from country to country. In US, for instance, prostitution is a crime that could be classified as anti-social behavior. In Brazil, prostitution is not crime. We decide not to drill down in each one of these cases; we created the generic classes to represent them.

C. Results: User Interaction

Figure 4a shows the results obtained from the tests, in which D2RCrime was used according to the indicators outlined in Section IV.A. Figure 4b shows the results for the case in which the D2R tool was used.

Taking into account that the users had no prior knowledge in the use of the tool or semantic technologies, the tests showed that the tool is a viable alternative to easily provide for the opening of data. This strengthened our hypothesis that the use of the SQL metaphor is a good heuristic for the success of the method. The high percentage obtained in the “RDF mapping” and “Correctness of vocabulary” indicators can be used to demonstrate the effectiveness of the method. During the experiments, it was also proven that this approach obtained good acceptance due to the fact that it is not necessary to invest time in semantic technologies/tools that are often not of direct interest to such users.

Regarding the “the number of activities done in the time constraint” indicator, we found that each concept of the ontology was mapped, with the aid of the tool, taking one minute on average. It was also perceived that the process of mapping the last concepts was always performed faster than mapping the initial concepts: after mapping the first concepts, the users acquire the minimum experience in the tool, enough to perform the subsequent tasks even more quickly.

Regarding the “RDF mapping” indicator, there were slight indications of mapping and usability failures. In one of the tests, the tool did not properly format a string informed by the user for the “date” field, causing the respective property of the ontology not to be mapped successfully. The “date” field is more prone to situations such as this, because several SQL functions are applied thereto (e.g.: substring) to format the data.

In order to make a comparative analysis, we conducted the same test with other users, but this time using a different methodology. We chose to use the tool provided by the D2R itself, where — given a relational database — the automated mapping functionality (generate mapping) is responsible for generating the mapping file starting from the structure of a relational database. In order to do so, the tool generates an RDF vocabulary according to the database, taking into account the table names as the ontology class names and the table columns as the ontology properties. The following aspects drove the choice of the D2R tool:

1) Independence of paid license;

2) Ease of use;

3) Availability on the market;

4) Ability to be used in a 30-minute test without the need for special infrastructure.

Approaches such as the Asio Semantic Bridge for Relational Databases — ASBRD[9], SquirrelRDF[10], and RDBToOnto [9] are methods that are close to our approach, but require a considerable learning curve, due largely to the need for specific configurations and the need to manipulate the mapping file manually. Tools such as Oracle Semantic Technologies and the ASIO SBRD itself require paid software licenses.

As the methodology for conducting this second phase of testing, a document containing the information needed to perform the installation of D2R Server software was made available to the users, as well as the procedures to generate the automatic mapping of the relational database and test whether the publication of the data was successful. Before beginning the tests, the basic operation of the D2RQ mapping file was explained to the users, detailing its main structures and compulsory components (*ClassMaps* and *PropertyBridges*). After these procedures, the users then began the tasks related to publication of the data.

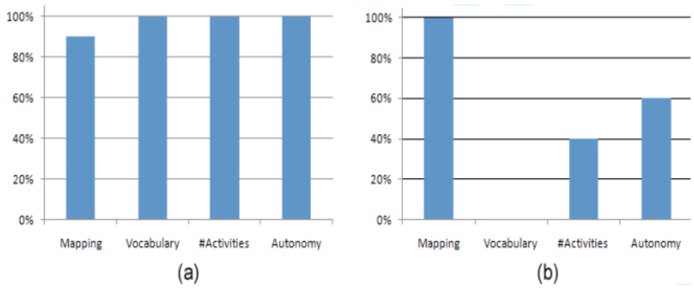


Fig. 4. Results of the evaluation (a) with the use of D2RCrime and (b) with the D2R standard tool

Figure 4b reflects the results of the testing, according to the same aforementioned indicators. The “RDF mapping” (100%) demonstrates that the approach is stable and is able to perform the mapping of the various types of data among the tables and columns involved. The “Correctness of vocabulary” indicator, however, got a very low percentage (0%). This is obviously due to the fact that using only the D2R, the classes and fields of the ontology cannot be generated. The D2R tool generates its own vocabulary created in an ad hoc way. This reflects a common fragility found in automated mapping approaches: although the data are mapped to RDF, in order for them to be able to actually represent the local domain and its respective relationships to be mapped, the mapping device must undergo a series of customizations to relate the generated instances efficiently.

The “the number of activities done in the time constraint” indicator (40%) shows that not all tests could be completed in the stipulated time. This is due to the fact that users had to learn how to configure the D2RServer software in order for the automatic mapping to be generated, confirming the fact that — even for a task that is simple to perform — a higher learning/difficulty curve is already shown to be present for the completion of the mapping tasks due to the need to learn about semantic tools.

D. Discussion

As a general result, the data obtained showed the proposed method as a viable alternative to easily provide for the opening of data on the Semantic Web. The D2RCrime tool is shown to be an effective alternative to lessen the steep learning curve required in this process.

It is important to stress that the automatic mapping generated by the D2R Server software does not provide integration with standardized ontologies accepted by the community (e.g.: GeoNames, Time, PMLP, Sioc, etc.), which somewhat hinders the context of data integration and reuse of information. Using the D2RCrime tool, the data are published using a proposed ontology that foresees this entire scenario of integration/mash-up of information.

It is also important to highlight that in order for semantic applications to be integrated more deeply to the published data, it’s necessary to replace the vocabulary generated automatically with RDF vocabularies that are standardized, accepted by the community, widely known, and publicly accessible. The generated mapping can be freely edited. However, in order to do so, the user must have all of the knowledge about how the mapping method and syntax work.

V. RELATED WORK

Metatomix’s Semantic Platform[11] and RDBtoOnto[12] [9] are examples of automatic tools that generate a populated ontology in RDF. In the case of the first, the mapping is done through a graphical eclipse plugin. Other structured sources can map to the same ontology allowing data integration under the same ontology. DB2OWL [10] automatically generates ontologies from database schemas, but it does not populate the ontology with instances. The mapping process is performed from the detection of particular cases for conceptual elements in the database, then the conversion is realized through the mappings from these components present in the database to their counterparts in the ontology.

Triplify [11] is a lightweight plug-in that exposes relational database data as RDF and Linked Data on the Web. There is no SPARQL support. The desired data to be exposed is defined in a series of SQL queries. Triplify is written only in PHP but has been adapted to several popular web applications (WordPress, Joomla, osCommerce, etc.).

ODEMapster[13] is a plugin for the NeOn toolkit, which provides a GUI to manage mappings between the relational database and RDFS/OWL ontologies. The mappings are expressed in the R2O language.

[9] http://www.bbn.com/technology/knowledge/asio_sbrd

[10] http://jena.sourceforge.net/SquirrelRDF

[11] http://www.metatomix.com

[12] http://www.tao-project.eu/researchanddevelopment/demosanddownloads/RDBToOnto.html

[13] http://neon-toolkit.org/wiki/ODEMapster

Asios' SBRD (Semantic Bridge for Relational Databases) enables integration of relational databases to the Semantic Web by allowing SPARQL queries over the relational database. An initially OWL ontology is generated from the database schema, which can then be mapped to a defined domain OWL ontology. The refinement of the ontology is done by means of Snoogle [12]. Snoogle converts the initial mappings to SWRL/RDF or SWRL/XML. It also allows two ontologies to be viewed on screen and then the correspondence between their classes can be generated, as well as attributes thereof. This whole process of mapping is accomplished via a visual interface.

This two-step approach followed by Asio requires a significant effort by the user compared with the approach we have proposed. For non-experts, it requires learning of two sets of tools. SquirrelRDF8 is a tool that allows relational databases to be queried using SPARQL. This tool takes a simplistic approach by not performing any complex model mapping like D2RQ. One of the most significant limitations of this approach is that it is not possible to use SPARQL queries searching for properties.

VI. CONCLUSION

In this paper we have described a method that relies on the representation of ontologies as a pattern to represent the concepts of crime and report of crimes. Besides a pattern, the ontologies are the interface to publish relational crime data on-the-fly. We have also proposed an interactive tool, called D2RCrime, which assists the designer/DBA to make the correspondence between the relational data and the classes and properties of the crime ontology. This correspondence allows automatic generation of the mapping rules between the two representations that conduct the process of access of relational data from SPARQL.

Open issues persist and will drive our future research. Open data may come from different sources. It will be necessary to have mechanisms to compare and check whether the information refers to the same fact. Creating mechanisms to automatically identify these repetitions is a challenge to be pursued. Another challenge, also due to the fact that information comes from different sources, is the need to account for the credibility of information automatically. When sources are known, such as official sources, the attribution of credibility is natural. However, the credibility of non-official information sources is difficult to be assigned. Methods for computing reputation and trustworthiness of the sources as in [13] [14] are examples of how this can be addressed.

Finally it is important to point out that the main advantage of having open crime data is the possibility that it will be used to provide services to citizens. Examples of this are alerts about how dangerous a certain place is and suggestions of safe routes. Such information can be enriched with data coming from popular participation, for example, via collaborative mapping. An example of collaborative mapping in Law Enforcement is WikiCrimes[14] [13]. WikiCrimes aims to offer a common interaction space among the public in general, so that people are able to report criminal facts as well as keep track of the locations where such crimes occur. We have integrated D2RCrime to WikiCrimes in which the instances retrieved by WikiCrimes from the Police Department's relational databases via D2RCrime are plotted directly on the digital map (for further details see [15]). Doing so, a set of services provided by WikiCrimes is available to the citizens. It is possible to receive alerts about dangerous places and to receive alerts by email as well. Apps for running on iPhones and Android smartphones also exist.

ACKNOWLEDGMENT

This work was supported in part by the CNPq under Grants 55977/2010-7 and 304347/2011-6 .

REFERENCES

[1] D. Lathrop, L. Ruma, "Open government: Collaboration, transparency, and participation in practice", in O'Reilly Media, 2010.

[2] C. Bizer, R. Cyganiak, "D2R Server - Publishing Relational Databases on the Semantic Web", in Poster at the 5th International Semantic Web Conference, 2006.

[3] J.R. Hobbs, F. Pan, "An ontology of time for the semantic web". In ACM Transactions on Asian Language Information Processing (TALIP), 66–85, ISSN 1530-0226.

[4] Y. Raimond, S. A. Abdallah, "The event ontology", 2006. Available: http://purl.org/NET/c4dm/event.owl.

[5] D. McGuinness, L. Ding, P. Pinheiro da Silva, C. Chang, "PML2: A Modular Explanation Interlingua", in Proceedings of the AAAI 2007 Workshop on Explanation-Aware Computing, Vancouver, British Columbia, Canada, July 22-23, 2007.

[6] L. Moreau, B. Clifford, J. Freire, J. Futrelle, Y. Gil, P. Groth, N. Kwas-nikowska, S. Miles, P. Missier, J. Myers, B. Plale, Y. Simmhan, E. Stephan, and J. Van Den Bussche, "The Open Provenance Model — Core Specification (v1.1)", in Future Generation Computer Systems, 2010.

[7] J. F. Sequeda, R. Depena, D. Miranker, "Ultrawrap: Using SQL Views for RDB2RDF", in Poster at the 8th International Semantic Web Conference (ISWC2009). Washington DC, US, 2009.

[8] C. Bizer, A. Seaborne, "The D2RQ Platform v0.7 - Treating Non-RDF Relational Databases as Virtual RDF Graphs". Available: http://www4.wiwiss.fu-berlin.de/bizer/d2rq/spec/20090810.

[9] F. Cerbah, "RDBToOnto: un logiciel dédié à l'apprentissage d'ontologies à partir de bases de données relationnelles", Strasbourg, 2009.

[10] N. Cullot, R. Ghawi, K. Yétongnon, "DB2OWL: A Tool for Automatic Database-to-Ontology", in Proceedings of the 15th Italian Symposium on Advanced Database Systems (SEBD), 2007.

[11] S. Auer, "Triplify – Light-Weight Linked Data Publication from Relational Databases", in Proceedings of the 18th World Wide Web Conference (WWW2009).

[12] H. Wang, C. Tan, Q. Li, "Snoogle: A search engine for the physical world", in IEEE Infocom, 2008.

[13] V. Furtado, L. Ayres, L. de Oliveira, M. Vasconcelos, C. Caminha, J. D'Orleans, J, "Collective Intelligence in the Law

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

Enforcement: The WikiCrimes System", in Information Science, 2010.

[14] I. Pinyol, J. Sabater-Mir, G. Cuni, "How to talk about reputation using a common ontology: From definition to implementation", in Proceedings of the Ninth Workshop on Trust in Agent Societies, Hawaii, USA. pp: 90-101. 2007.

[15] . J. Tavares, V. Furtado, H. Santos, "Open Government in Law Enforcement: Assisting the publication of Crime Occurrences in RDF from Relational Data", in AAAI Fall Symposium on Open Government Knowledge: AI Opportunities and Challenges, Arlington, VA, 2011.

The *Why Agent*

Enhancing user trust in automation through explanation dialog

Rob Cole
Raytheon Company
Intelligence and Information Systems
State College, PA, U.S.A.

Michael J. Hirsch
Raytheon Company
Intelligence and Information Systems
Orlando, FL, U.S.A.

Jim Jacobs
Raytheon Company
Network Centric Systems
Ft. Wayne, IN, U.S.A.

Robert L. Sedlmeyer
Indiana University – Purdue University
Department of Computer Science
Ft. Wayne, IN, U.S.A

***Abstract*— Lack of trust in autonomy is a recurrent issue that is becoming more and more acute as manpower reduction pressures increase. We address the socio-technical form of this trust problem through a novel decision explanation approach. Our approach employs a semantic representation to capture decision-relevant concepts as well as other mission-relevant knowledge along with a reasoning approach that allows users to pose queries and get system responses that expose decision rationale to users. This representation enables a natural, dialog-based approach to decision explanation. It is our hypothesis that the transparency achieved through this dialog process will increase user trust in autonomous decisions. We tested our hypothesis in an experimental scenario set in the maritime autonomy domain. Participant responses on psychometric trust constructs were found to be significantly higher in the experimental group for the majority of constructs, supporting our hypothesis. Our results suggest the efficacy of incorporating a decision explanation facility in systems for which a socio-technical trust problem exists or might be expected to develop.**

Keywords-Semantic modeling; Maritime Autonomy; Trust in Autonomy; Decision Explanation.

I. Introduction

Large organizations such as the Department of Defense rely heavily on automation as a means of ensuring high-quality product, as well as cost control through manpower reduction. However, lack of user trust has repeatedly stood in the way of widespread deployment. We have observed two fundamental forms of the problem: the technical and the socio-technical form. The technical form is characterized by user reservations regarding the ability of a system to perform its mission due to known or suspected technical defects. For example, an automated detection process might have a very high false positive rate, conditioning operators to simply ignore its output. Trust in such a situation can only be achieved by addressing the issue of excessive false detections, a technical problem suggesting a purely technical solution. As another example, consider a situation in which automation is introduced into a purely manual process characterized by decision making in high-pressure situations. In such a situation, operators might reject automation in favor of the trusted, manual process for purely non-technical reasons. In other words, in the absence of any specific evidence of limitations of the automation, the automation could nonetheless be rejected for reasons stemming from the social milieu in which the system operates. This is the socio-technical form of the problem.

One might address the socio-technical problem through education: train the operators with sufficient knowledge of system specifications and design detail to erase doubts they may have regarding the automation. Such an approach is costly since every operator would have to be trained to a high degree. Operators would essentially have to be system specialists. Instead, we propose an approach intended for non-specialist operators, stemming from the insight that the socio-technical trust problem results from a lack of insight into system decision rationale. If an operator can be made to understand the *why* of system behavior, that operator can be expected to trust the system in the future to a greater degree, if the rationale given to the operator makes sense in the current mission context.

Explanation mechanisms in expert systems have focused on the use of explicit representations of design logic and problem solving strategies [1]. The early history of explanation in expert systems saw the emergence of three types of approaches, as described in Chandrasekaran, Tanner, and Josephson [2]. Type I systems explain how data matches local goals. Type 2 systems explain how knowledge can be justified [3]. Type 3 systems explain how control strategy can be justified [4]. A more detailed description of these types is given by Saunders and Dobbs [5, p. 1102]:

> Type 1 explanations are concerned with explaining why certain decisions were or were not made during the execution (runtime) of the system. These explanations use information about the relationships that exist between pieces of data and the knowledge (sets of rules for example) available for making specific decisions or choices based on this data. For example, Rule X fired because Data Y was found to be true.
>
> Type 2 explanations are concerned with explaining the knowledge base elements themselves. In order to do this, explanations of this type must look at knowledge about

This research was supported by Raytheon Corporate IR&D.

knowledge. For example, knowledge may exist about a rule that identifies this rule (this piece of knowledge) as being applicable ninety percent of the time. A type 2 explanation could use this information (this knowledge about knowledge) to justify the use of this rule. Other knowledge used in providing this type of explanation consists of knowledge that is used to develop the ES but which does not affect the operation of the system. This type of knowledge is referred to as deep knowledge.

Type 3 explanations are concerned with explaining the runtime control strategy used to solve a particular problem. For example, explaining why one particular rule (or set of rules) was fired before some other rule is an explanation about the control strategy of the system. Explaining why a certain question (or type of question) was asked of the user in lieu of some other logical or related choice is another example. Therefore, type 3 explanations are concerned with explaining how and why the system uses its knowledge the way it does, a task that also requires the use of deep knowledge in many cases.

Design considerations for explanations with dialog are discussed in a number of papers by Moore and colleagues ([6], [7], [8] and [9]). These papers describe the explainable expert systems (EES) project which incorporates a representation for problem-solving principles, a representation for domain knowledge and a method to link between them. In Moore and Swartout [6], hypertext is used to avoid the referential problems inherent in natural language analysis. To support dialog with hypertext, a planning approach to explanation was developed that allowed the system to understand what part of the explanation a user is pointing at when making further queries. Moore and Paris [8] and Carenini and Moore [9] discuss architectures for text planners that allow for explanations that take into account the context created by prior utterances. In Moore [10], an approach to handling badly-formulated follow-up questions (such as a novice might produce after receiving an incomprehensible explanation from an expert) is presented that enables the production of clarifying explanations. Tanner and Keuneke [11] discuss an explanation approach based on a large number of agents with well-defined roles is described. A particular agent produces an explanation of its conclusion by ordering a set of text strings in a sequence that depends on the decision's runtime context. Based on an explanation from one agent, users can request elaboration from other agents.

Weiner [12] focuses on the structure of explanations with the goal of making explanations easy to understand by avoiding complexity. Features identified as important for this goal include syntactic form and how the focus of attention is located and shifted. Eriksson [13] examines answers generated through transformation of a proof tree, with pruning of paths, such as non-informative ones. Millet and Gilloux [14] describe the approach in Wallis and Shortliffe [15] as employing a user model in order to provide users with explanations tailored to their level of understanding. The natural language aspect of explanation is the focus of Papamichail and French [16], which uses a library of text plans to structure the explanations.

In Carenini and Moore [17], a comprehensive approach toward the generation of evaluative arguments (called GEA) is presented. GEA focuses on the generation of text-based arguments expressed in natural language. The initial step of GEA's processing consists of a text planner selecting content from a domain model by applying a communicative strategy to achieve a communication goal (e.g. make a user feel more positively toward an entity). The selected content is packaged into sentences through the use of a computational grammar. The underlying knowledge base consists of a domain model with entities and their relationships and an additive multi-attribute value function (a decision-theoretic model of the user's preferences).

In Gruber and Gautier [18] and Gautier and Gruber [19] an approach to explaining the behavior of engineering models is presented. Rather than causal influences that are hard-coded [20], this approach is based on the inference of causal influences, inferences which are made at run time. Using a previously developed causal ordering procedure, an influence graph is built from which causal influences are determined. At any point in the influence graph, an explanation can be built based on the adjacent nodes and users can traverse the graph, obtaining explanations at any node.

Approaches to producing explanations in MDPs are proposed in Elizalde et al. [21] and Khan, Poupart and Black [22]. Two strategies exist for producing explanations in BNs. One involves transforming the network into a qualitative representation [23]. The other approach focuses on the graphical representation of the network. A software tool called Elvira is presented which allows for the simultaneous display of probabilities of different evidence cases along with a monitor and editor of cases, allowing the user to enter evidence and select the information they want to see [24].

An explanation application for JAVA debugging is presented in Ko and Myers [25]. This work describes a tool called *Whyline* which supports programmer investigation of program behavior. Users can pose "*why did*" and "*why didn't*" questions about program code and execution. Explanations are derived using a static and dynamic slicing, precise call graphs, reachability analysis and algorithms for determining potential sources of values.

Explanations in case-based reasoning systems are examined as well. Sørmo, Cassens, and Aamodt [26] present a framework for explanation and consider specific goals that explanations can satisfy which include transparency, justification, relevance, conceptualization and learning. Kofod-Petersen and Cassens [27] consider the importance of context and show how context and explanations can be combined to deal with the different types of explanation needed for meaningful user interaction.

Explanation of decisions made via decision trees is considered in Langlotz, Shortliffe, and Fagan [28]. An explanation technique is selected and applied to the most significant variables, creating a symbolic expression that is converted to English text. The resulting explanation contains no mathematical formulas, probability or utility values.

Lieberman and Kumar [29] discuss the problem of mismatch between the specialized knowledge of experts providing help and the naiveté of users seeking help is considered. Here, the problem consists of providing explanations of the expert decisions in terms the users can understand. The *SuggestDesk* system is described which advises online help personnel. Using a knowledgebase, analogies are found between technical problem-solution pairs and everyday life events that can be used to explain them.

Bader et al. [30] use explanation facilities in recommender systems to convince users of the relevance of recommended items and to enable fast decision making. In previous work, Bader found that recommendations lack user acceptance if the rationale was not presented. This work follows the approach of Carenini and Moore [17].

In Pu and Chen [31], a "*Why?*" form of explanation was evaluated against what the researchers termed an Organized View (OV) form of explanation in the context of explanations of product recommendations. The OV approach attempts to group decision alternatives and provide group-level summary explanations, e.g. "these are cheaper than the recommendation but heavier." A trust model was used to conduct a user evaluation in which trust-related constructs were assessed through a Likert scale instrument. The OV approach was found to be associated with higher levels of user trust than the alternative approach.

The important of the use of context in explaining the recommendations of a recommendation system was investigated in Baltrunas et al. [32]. In this study of point-of-interest recommendation, customized explanation messages are provided for a set of 54 possible contextual conditions (e.g. "this place is good to visit with family"). Even where more than one contextual condition holds and is factored into the system's decision, only one can be utilized for the explanation (the most influential one in the predictive model is used). Only a single explanatory statement is provided to the user.

Explanation capabilities have also been shown to aid in increasing user satisfaction with and establishing trust in complex systems [34, 35, 36]. The key insight revealed by this research is the need for *transparency* in system decision-making. As noted by Glass et al., "users identified explanations of system behavior, providing transparency into its reasoning and execution, as a key way of understanding answers and thus establishing trust. [37]" Dijkstra [38] studied the persuasiveness of decision aids, for novices and experts. In one experiment, lawyers examined the results of nine legal cases supported by one out of two expert systems. Both systems had incomplete knowledge models. Because of the incomplete models, the expert systems routinely gave opposite advice on each legal case. This resulted in the lawyers being easily mis-led. Therefore, adequate explanation facilities and a good user-interface must provide the user with the transparency needed to make the decision of trusting the system. Rieh and Danielson [39] Outline four different explanation types of decision aids. Line-of-reasoning explanations provide the logical justification of the decision; justification explanations provide extensive reference material to support the decision; control explanations provide the problem-solving strategy to arrive at the decision; and terminological explanations provide definition information on the decision. In each case, the amount of transparency in the decision-making process is a factor in the trust of the user.

Our approach to providing transparency, the *Why Agent*, is a decision explanation approach incorporating dialog between the user and the system. Rather than attempting to provide monolithic explanations to individual questions, our dialog-based approach allows the user to pose a series of questions, the responses to which may prompt additional questions. Imitative of natural discourse, our dialog approach allows a user to understand the behavior of the system by asking questions about its goals, actions or observables and receiving responses couched in similar terms. We implemented our approach and conducted an evaluation in a maritime autonomy scenario. The evaluation consisted of an experiment in which two versions of an interface were shown to participants who then answered questions related to trust. Results of the experiment show response scores statistically consistent with our expectations for the majority of psychometric constructs tested, supporting our overall hypothesis that transparency fosters trust. The rest of this paper is organized as follows. Section II describes the problem domain and the technical approach. Experiments and results are presented in Section III. In Section IV, we provide some concluding remarks and future research directions.

II. Technical Approach

A. *Domain Overview*

Our approach to demonstrating the Why Agent functionality and evaluating its effectiveness consisted of a simulation-based environment centered on a maritime scenario defined in consultation with maritime autonomy SMEs. The notional autonomous system in our scenario was the X3 autonomous unmanned surface vehicle (AUSV) by Harbor Wing Technologies [1]. Raytheon presently has a business relationship with this vendor in which we provide ISR packages for their AUSVs.

The X3 was of necessity a notional AUSV for our demonstration because the actual prototype was not operational at the time of the Why Agent project. For this reason, a live, on-system demonstration was not considered. Instead, our demonstration environment was entirely simulation-based. An existing route planning engine developed under Raytheon research was modified to serve as the AUSV planner. Additional code was developed to support the simulation environment and Why Agent functionality, as described below.

B. *Software Architecture*

Our software architecture consists of four components interacting in a service-oriented architecture, as shown in Figure 1.

The Planner component performed route planning functions based on a plan of intended movement. A plan of intended movement is input in the form of a series of waypoints. These waypoints, along with environmental factors, such as weather forecast data, are used in the planning algorithm to determine

[1] http://www.harborwingtech.com

an actual over-ocean route. The planner was a pre-existing component developed on R&D that the Why Agent leveraged for the demonstration. Modifications made to the planner to support the Why Agent project include changes to expose route change rationale to the controller and inform the controller of weather report information.

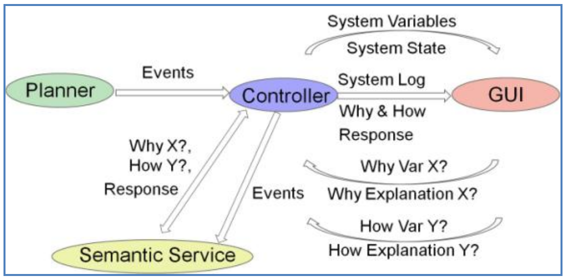


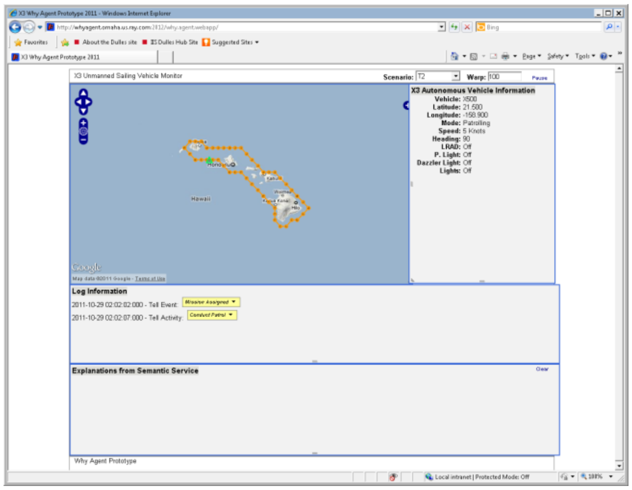
Figure 1: SW architecture for Why Agent.

The Controller represents the embodiment of the majority of the simulated AUSV decision logic and simulation control logic. Because we did not employ an actual AUSV for the Why Agent project, much of the decision logic of an actual AUSV had to be simulated for our demonstration, logic implemented in the Controller. The input to the Controller consisted of a test control file that defined the event timeline for the simulation. In addition to orchestrating simulation events defined in the control file, the Controller mediated queries and responses between the user interface and the semantic service.

The graphical user interface was implemented as a web application. Two versions of the GUI were developed, one with and one without the Why Agent explanation facility. The Why Agent version is shown in Figure 2. It has four screen regions: a map, a status panel, a log data panel and an explanation panel. The map, implemented with Google Map technology, shows the current location and route of the AUSV. The status panel shows various AUSV status values, such as location, speed, current mode, etc. The log panel shows a time-stamped series of event descriptions. Various items in the log panel are user-selectable and have context-sensitive menus to support the user interface functionality of the Why Agent facility. When a user makes a selection, the response from the semantic service is shown in the bottom (explanation) panel. Additionally, responses in the explanation panel are also selectable for further queries. In this manner, the user can engage in a dialog with the system.

The semantic service contains the knowledgebase underlying the decision rationale exposed by the Why Agent. The knowledge consists of event and domain ontology models represented in web ontology language (OWL) format. The semantic service provides responses to queries from the controller through queries against its underlying models.

An example of a domain model is shown in Figure 3. Relationships in this figure encode potential queries linking concepts and events that can be displayed in the user interface. For example, the activity *ConductPatrol* relates to the function *MissionExecution* through the relationship *servesPurpose*. This relationship is statically associated with the query why? at the user level. Thus, the existence of this link connected with the node *ConductPatrol* implies a why? option being made available to the user in the context-sensitive menu for the *ConductPatrol* item. When the user selects the *ConductPatrol* item and the associated *why?* option, a query is generated that contains IDs associated with the *ConductPatrol* node and the *servesPurpose* link. The linked node, in this case *MissionExecution*,is then returned to the user as the result of a query against the associated OWL model.


X3 Why Agent Prototype 2011 - Windows Internet Explorer

http://whyagent.omaha.us.ray.com:2012/why.agent.webapp/

X3 Unmanned Sailing Vehicle Monitor

Scenario: T2   Warp: 100

X3 Autonomous Vehicle Information
Vehicle: X500
Latitude: 21.500
Longitude: -158.900
Mode: Patrolling
Speed: 5 Knots
Heading: 90
LRAD: Off
P. Light: Off
Dazzler Light: Off
Lights: Off

Log Information
2011-10-29 02:02:02:000 - Tell Event: Mission Assigned
2011-10-29 02:02:07:000 - Tell Activity: Conduct Patrol

Explanations from Semantic Service

Why Agent Prototype

Figure 2: General GUI for Why Agent interface.

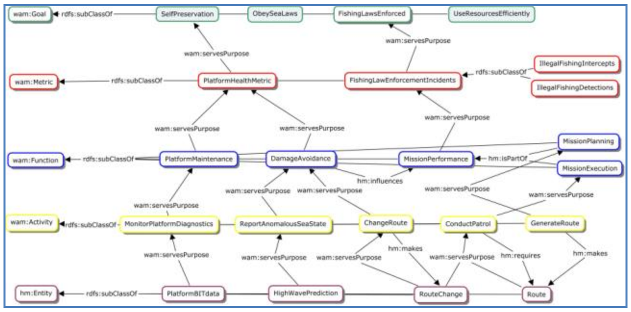


Figure 3: Example domain model.

III. Experimentation

Our evaluation approach consisted of an experiment in which the Why Agent was the treatment. Two versions of a prototype operator interface were developed. One version incorporated the Why Agent functionality and the second did not. The two versions were otherwise identical. Screenshots of the two interface versions are presented in Figures 4 and 5.

A. Demonstration Scenario

The demonstration scenario consisted of autonomous fishing law enforcement in the Northwestern Hawaiian Islands Marine National Monument. The CONOP for this mission is as follows:

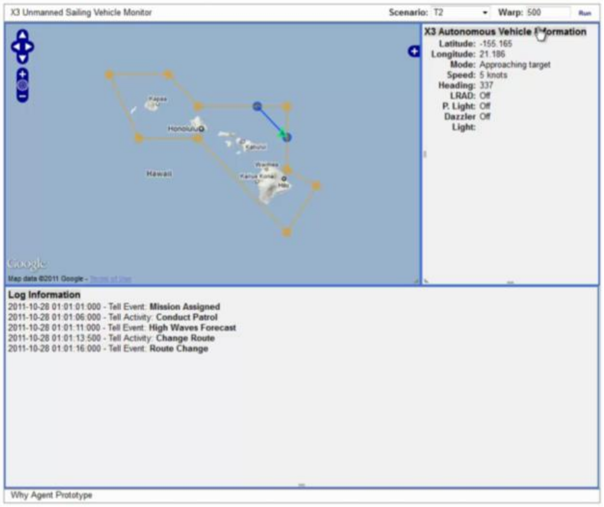
- The AUSV operator selects waypoints corresponding to a patrol area.

- The AUSV route planner finds a route through the waypoints and a patrol is conducted.

- RADAR is used to detect potential illegal fishing vessels (targets)
- Targets are investigated visually after AUSV closes to an adequate proximity.
- Automated analysis of the visual data is used to confirm the target is engaged in illegal fishing.
- Targets engaged in illegal activity are visually identified for subsequent manned enforcement action.

Non-lethal self-defensive actions can be taken by the AUSV in the presence of hostile targets.

To support this demonstration, a software-based simulation environment was developed. The demonstration consisted of capturing video of user interactions with the baseline and Why Agent versions of the operator interface while a scripted series of events unfolded over a pre-determined timeline.



X3 Unmanned Sailing Vehicle Monitor — Scenario: T2 — Warp: 500

X3 Autonomous Vehicle Information
Latitude: -155.165
Longitude: 21.186
Mode: Approaching target
Speed: 5 knots
Heading: 337
LRAD: Off
P. Light: Off
Dazzler: Off
Light:

Log Information
2011-10-28 01:01:01:000 - Tell Event: **Mission Assigned**
2011-10-28 01:01:06:000 - Tell Activity: **Conduct Patrol**
2011-10-28 01:01:11:000 - Tell Event: **High Waves Forecast**
2011-10-28 01:01:13:500 - Tell Activity: **Change Route**
2011-10-28 01:01:16:000 - Tell Event: **Route Change**

Why Agent Prototype

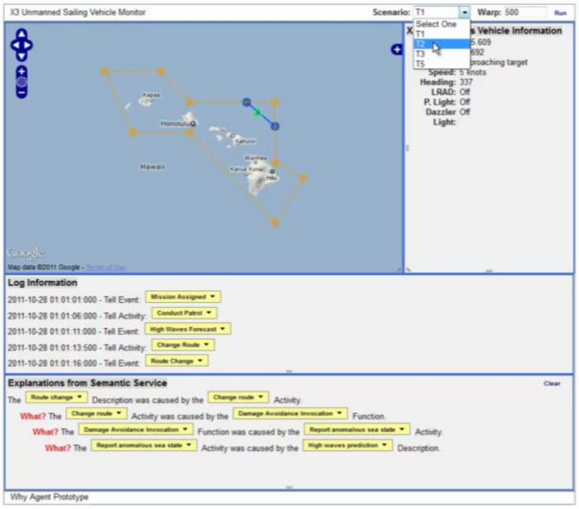
Figure 4: Operator interface without the Why Agent functionality.

B. Experimental Design

Our experiment consisted of a single-factor, randomized design. The factor is interface type and has two levels: baseline (control) and Why Agent (experimental). Thus, we have two treatment levels, corresponding to the two factor types. The experimental subjects were Raytheon employees, recruited across multiple Raytheon locations, during the project.

Our general hypothesis is that **the Why Agent fosters a more appropriate level of trust in users than the baseline system**. By utilizing the information provided by the Why Agent, users will be more able to calibrate their trust [33]. To test this hypothesis, we needed to operationalize the concept of "more appropriate level of trust" and thereby derive one or more testable hypotheses. We accomplished this through the following operationalization.

Trust in a particular system, being an unobservable mental aspect of a user, necessitates the use of psychometric readings of constructs related to the overall concept of trust. Given the broad nature of this concept, multiple constructs should be defined. Using our domain insight and engineering judgment, we selected the following set of five psychometric constructs: 1. General Competence, 2) Self-Defense, 3) Navigation, 4) Environmental Conservation and 5) Mission. Each construct is intended to capture the users' belief regarding the system's ability to effectively perform in regard to that construct, i.e. the user's level of trust for that construct. For example, the construct *Mission* attempts to encompass user attitudes toward the ability of the system to successfully execute its mission. The Environmental Conservation construct was included as an example of a construct under which we would not expect to see a difference in psychometric responses.



X3 Unmanned Sailing Vehicle Monitor — Scenario: T1 — Warp: 500

Heading: 337
LRAD: Off
P. Light: Off
Dazzler: Off
Light:

Log Information
2011-10-28 01:01:01:000 - Tell Event: Mission Assigned ▾
2011-10-28 01:01:06:000 - Tell Activity: Conduct Patrol ▾
2011-10-28 01:01:11:000 - Tell Event: High Waves Forecast ▾
2011-10-28 01:01:13:500 - Tell Activity: Change Route ▾
2011-10-28 01:01:16:000 - Tell Event: Route Change ▾

Explanations from Semantic Service
The Route change ▾ Description was caused by the Change route ▾ Activity.
What? The Change route ▾ Activity was caused by the Damage Avoidance Invocation ▾ Function.
What? The Damage Avoidance Invocation ▾ Function was caused by the Report anomalous sea state ▾ Activity.
What? The Report anomalous sea state ▾ Activity was caused by the High waves prediction ▾ Description.

Why Agent Prototype

Figure 5: Operator interface with the Why Agent functionality.

For each construct, we have a set of possible trust levels and a set of psychometric participant response scores. Define these as follows (for this study, k=5):

- Set of k constructs C = $\{c_j : 1 \leq j \leq k\}$
- Set of trust levels L = $\{low, high\}$
- Psychometric participant response scores for each construct:

  Control: $R^C = \{r_j^C : 1 \leq j \leq k\}$

  Experimental: $R^E = \{r_j^E : 1 \leq j \leq k\}$

Here, we take the simplest possible approach, a binary trust level set. We simply assume that the trust level for a particular construct should either be low or high, with nothing in between. Clearly, many other trust models are possible. To operationalize the notion of "more appropriate level of trust", we need to define, for each construct, a ground truth assignment of trust level. Thus, we need to define the following mapping T:

- Mapping of construct to trust level: $T(j) \in$ L
  - $T(j) = low$: People *should not* trust the system regarding construct j
  - $T(j) = high$: People *should* trust the system regarding construct j.

Additionally, we need to map the elements of the trust set to psychometric scale values. In other words, we need to normalize the scale as follows:

- Mapping of trust level to psychometric scale values

$$S: S(low) = 1; S(high) = 5.$$

At this point, we can define the concept of "appropriate level of trust" in terms of the psychometric scale through a composition of the above mappings *S* and *T*. In other words, for each construct, the appropriate level of trust is the psychometric value associated with the trust level assigned to that construct:

- Appropriate Level of Trust *with respect to design intent* $A = \{a_j : 1 \leq j \leq k\}$

For each construct c_j, the appropriate level of trust a_j for that construct is given by

$$a_j = S(T(j)),\ 1 \leq j \leq k \qquad (1)$$

A key aspect of the above definition is the qualifier *with respect to design intent*. We assume the system functions without defects. *With respect to design intent* simply means "it should be trusted to accomplish X if it is designed to accomplish X." We make this assumption for simplification purposes, fully acknowledging that no real system is defect-free. In the presence of defects, the notion of *appropriate level of trust* becomes more complex.

Having defined *appropriate level of trust*, we are finally in a position to define the key concept, *more appropriate level of trust.* The intuition underlying this notion is the observation that if one's trust level is not appropriate to begin with, any intervention that moves the trust level toward the appropriate score by a greater amount than some other intervention can be said to provide a "more" appropriate level of trust. The Why Agent specifically exposes information associated with the purpose of AUSV actions. Such additional information serves to build trust [33]. If the psychometric score for the experimental group is closer to the appropriate trust level than the score for the control group, then we can say that the experimental treatment provided a more appropriate level of trust for that construct. Formally, we define this concept as follows:

- More appropriate level of trust: Given observed response scores r_j^C and r_j^E for construct *j*, the experimental response r_j^E reflects a more appropriate level of trust when the following holds

$$r_j^E - r_j^C < 0 \text{ if } a_j = 1 \qquad (2)$$
$$r_j^E - r_j^C > 0 \text{ if } a_j = 5 \qquad (3)$$

We expect the Why Agent to affect observed trust levels only for those constructs for which relevant decision criteria are exposed during the scenario. In these cases, we expect Equations (2)-(3) to hold. In all other cases, we do not. For example, since the AUSV is not designed to protect marine life, we assert that the appropriate level of trust for the Environmental Conservation construct is "low." However, we do not expect to observe response levels consistent with Equations (2) – (3) unless dialog exposing decision rationale relevant to this concept is included in the scenario.

Based on this reasoning, we expect the effect of decision explanation to be one of pushing response scores up or down, toward the appropriate trust level but only in cases where explanation dialog related to the construct under test is exposed. In other cases, we expect no difference in the response scores, as indicated in Table 1. We note that the null hypotheses are derived as the complementary sets to the equations in Table 1. E.g., the 'low, with relevant dialog' null hypothesis equation would be $r_j^E - r_j^C \geq 0$.

A total of 44 control and 50 experimental subjects were recruited for the Why Agent study. The experiment was designed to be completed in one hour. Following a short orientation, a pre-study questionnaire was presented to the participants. The pre-study questionnaire contained questions regarding participant demographics and technology attitudes. The purpose of the pre-study questionnaire was to determine whether any significant differences existed between the experimental and control groups. Following the pre-study questionnaire, participants were given a short training regarding the autonomous system and their role in the study. Participants were asked to play the role of a Coast Guard commander considering use of the autonomous system for a drug smuggling interdiction mission. Following the training, participants were shown the scenario video which consisted of several minutes of user interaction with either the baseline or Why Agent interface. Following the video, participants completed the main study questionnaire. The system training was provided in a series of powerpoint slides. Screenshots taken from the study video were provided to the participants in hardcopy form, along with hardcopies of the training material. This was done to minimize any dependence on memory for participants when completing the study questionnaire.

Table 1: Expected responses as a result of decision explanation.

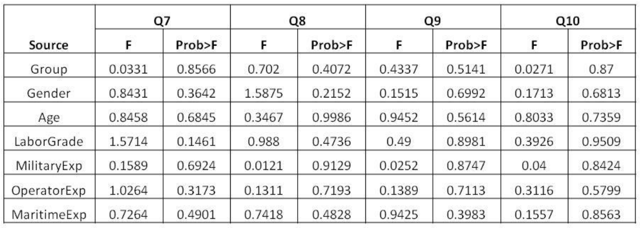
		Experimental Condition	
		With relevant dialog	Without relevant dialog
Construct trust level	Low	Experimental response less than control response $r_j^E - r_j^C < 0$	Experimental response indistinguishable from control response $r_j^E - r_j^C = 0$
	High	Experimental response greater than control response $r_j^E - r_j^C > 0$	Experimental response indistinguishable from control response $r_j^E - r_j^C = 0$

C. *Experimental Results*

To investigate whether significant differences exist between the control and experimental groups in terms of responses to the technology attitudes questions, ANOVA was performed. The results are shown in Table 2. Cronbach reliability coefficients, construct variances and mean total response scores are shown for the control and experimental groups in Tables 3 and 4.

To investigate whether significant differences exist between the control and experimental groups in terms of responses to the study questions, ANOVA was performed. For this study, we focused our analysis on individual constructs. Thus, we do not present any statistics on, for example, correlations among responses related to multiple constructs for either the control or experimental group. The results are shown in Table 6.

Table 2: ANOVA computations analyzing differences between control and experimental groups, for technology attitude questions.



Source	Q7		Q8		Q9		Q10	
	F	Prob>F	F	Prob>F	F	Prob>F	F	Prob>F
Group	0.0331	0.8566	0.702	0.4072	0.4337	0.5141	0.0271	0.87
Gender	0.8431	0.3642	1.5875	0.2152	0.1515	0.6992	0.1713	0.6813
Age	0.8458	0.6845	0.3467	0.9986	0.9452	0.5614	0.8033	0.7359
LaborGrade	1.5714	0.1461	0.988	0.4736	0.49	0.8981	0.3926	0.9509
MilitaryExp	0.1589	0.6924	0.0121	0.9129	0.0252	0.8747	0.04	0.8424
OperatorExp	1.0264	0.3173	0.1311	0.7193	0.1389	0.7113	0.3116	0.5799
MaritimeExp	0.7264	0.4901	0.7418	0.4828	0.9425	0.3983	0.1557	0.8563

Table 3: Cronbach reliability coefficients, construct variances, and means for control group.

Control Results						
	Variances					
Construct	Q1	Q2	Q3	Total	Cronbach Alpha	Mean
1	0.492	0.306	0.348	2.20	0.72	11.11
2	0.710	0.517	NA	1.79	0.63	6.43
3	0.720	0.319	NA	1.05	0.02	7.30
4	0.911	0.670	NA	2.02	0.43	6.73
5	0.953	0.586	NA	2.23	0.62	7.34

T-test results for each construct are shown in Table 5. Two p-values are shown for each construct; p1 represents the p-value resulting from use of the pooled variance while p2 represents the p-value resulting from use of separate variances.

The ANOVA results shown in Table 2 indicate that the experimental and control groups did not significantly differ across any attribute in terms of their responses to the technology attitudes questions. In other words, we do not see any evidence of a technology attitude bias in the study participants.

Table 4: Cronbach reliability coefficients, construct variances, and means for control group.

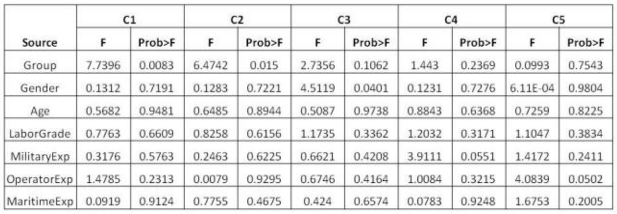
Experimental Results						
	Variances					
Construct	Q1	Q2	Q3	Total	Cronbach Alpha	Mean
1	0.286	0.262	0.449	1.94	0.73	12.06
2	0.689	0.694	NA	2.18	0.73	7.22
3	0.480	0.367	NA	1.17	0.56	7.64
4	0.571	0.621	NA	1.92	0.76	7.14
5	0.898	0.629	NA	2.05	0.51	7.46

Table 5: T-test computations for each construct.

Construct Hypothesis Tests				
	p-values			
Construct	p1	p2	Null Hypothesis	Result
1	0.001	0.001	Experimental score is not greater than Control score	Reject Null Hypothesis
2	0.004	0.004	Experimental score is not greater than Control score	Reject Null Hypothesis
3	0.058	0.059	Experimental score is not greater than Control score	Accept Null Hypothesis
4	0.158	0.159	Experimental score is equal to Control score	Accept Null Hypothesis
5	0.348	0.347	Experimental score is not greater than Control score	Accept Null Hypothesis

For constructs one and two, the experimental response was greater than the control response (p = 0.001 and 0.004, respectively), consistent with our expectations. For construct four, environmental conservation, we see no significant difference between the experimental and control responses (p = 0.16), which is also consistent with our expectations as this construct had no associated decision explanation content exposed to the experimental group. The experimental response for construct 3 was not significantly higher than the control response, which is inconsistent with our expectations, although the difference is only marginally outside the significance threshold (p = 0.059).

Table 6: ANOVA computations analyzing differences between control and experimental groups, for study questions.



Source	C1		C2		C3		C4		C5	
	F	Prob>F	F	Prob>F	F	Prob>F	F	Prob>F	F	Prob>F
Group	7.7396	0.0083	6.4742	0.015	2.7356	0.1062	1.443	0.2369	0.0993	0.7543
Gender	0.1312	0.7191	0.1283	0.7221	4.5119	0.0401	0.1231	0.7276	6.11E-04	0.9804
Age	0.5682	0.9481	0.6485	0.8944	0.5087	0.9738	0.8843	0.6368	0.7259	0.8225
LaborGrade	0.7763	0.6609	0.8258	0.6156	1.1735	0.3362	1.2032	0.3171	1.1047	0.3834
MilitaryExp	0.3176	0.5763	0.2463	0.6225	0.6621	0.4208	3.9111	0.0551	1.4172	0.2411
OperatorExp	1.4785	0.2313	0.0079	0.9295	0.6746	0.4164	1.0084	0.3215	4.0839	0.0502
MaritimeExp	0.0919	0.9124	0.7755	0.4675	0.424	0.6574	0.0783	0.9248	1.6753	0.2005

While the test results indicate moderate support for the efficacy of the Why Agent approach, they are decidedly mixed, so it is not possible to draw any definitive conclusions. As discussed below, we recognize that a number of significant limitations also hinder the application of our results. A pilot study would have helped to create a stronger experimental design and recruit a more representative sample population, but this was not possible due to budget and schedule constraints. Nevertheless, the study has provided initial evidence for how and to what extent the Why Agent approach might influence trust behavior in autonomous systems, and given impetus for continued investigations.

Construct Reliability: Referring to Table 4, we see that reliability coefficients for some constructs are not above the commonly-accepted value of 0.7. Had schedule permitted, a pilot study could have uncovered this issue, providing an opportunity to revise the questionnaire.

Experiment Limitations: Clearly a variety of limitations apply to our experiment. One is that participants did not interact directly with the system interface; instead entire groups of participants were shown a video of someone else interacting with the system. Also, the participants were not drawn from the population of interest. Consequently, our results may not apply to that target group. Additionally, subjects were asked to play a role with much less information than a real person in that role would have. Also, as noted by a reviewer, the experimental design does not allow us to determine whether decision correctness is related to trust when clearly it should be; an intervention that raises trust regardless of correctness is not desirable. Finally, execution of the experiment could have been improved. In particular, our maritime autonomy SME noted: The Mode should have reflected the simulation events; The LRAD light should have illuminated during the approach phase with an audio warning; The subjects should have been trained on the nonlethal defense functions.

Semantic Modeling: A potentially significant drawback to our approach is the manually-intensive nature of the semantic modeling effort needed to populate our knowledgebase. Identifying ways to automate this process is a key area of potential future work related to this effort.

IV. Concluding Remarks

We draw the following specific conclusions based on the quantitative results reported above. First, the experimental and control groups do not significantly differ across any attribute in terms of their responses to the technology attitudes questions. The experimental and control groups do not significantly differ across any non-Group attribute in terms of their responses to the study questions with the exception of gender differences for construct. Construct reliability is low in some cases, indicating the need for a prior pilot study to tune the psychometric instrument. We accept the null hypothesis for construct 4 and reject it for constructs 1 and 2, as predicted under our assumptions. We cannot reject the hypothesis associated with construct 3, although this is a very marginal case. The results of construct 5 are contradictory to our expectations. Overall, we conclude that the Why Agent approach does increase user trust levels through decision transparency.

References

[1] B. Chandrasekaran and W. Swartout, “Explanations in knowledge systems: the role of explicit representation of design knowledge,” *IEEE Expert* vol. 6, no. 3, pp. 47-19, 1991.

[2] B. Chandrasekaran et al., “Explaining control strategies in problem solving,” *IEEE Expert* vol. 4, no.1, pp. 9-15, 1989.

[3] William R. Swartout. “XPLAIN: a system for creating and explaining expert consulting programs,” *Artificial Intelligence*, vol. 21, no. 3, pp. 285-325, 1983.

[4] William J. Clancey, “The epistemology of a rule-based expert system — a framework for explanation,” *Artificial Intelligence*, vol 20., no. 3, pp. 215-251, 1983.

[5] V. M. Saunders and V. S. Dobbs, “Explanation generation in expert systems,” in *Proceedings of the IEEE 1990 National Aerospace and Electronics Conference*, vol. 3, pp. 1101-1106, 1990.

[6] J. Moore and W. Swartout, “Pointing: A Way Toward Explanation Dialog,” *AAAI Proceedings*, pp.457-464, 1990.

[7] Swartout et al., 1991. “Explanations in knowledge systems: design for explainable expert systems,” *IEEE Expert*, vol. 6, no. 3, pp. 58-64, 1991.

[8] Johanna D. Moore and Cécile L. Paris, “Planning text for advisory dialogues,” in *Proceedings of the 27th annual meeting on Association for Computational Linguistics*, 1989.

[9] Giuseppe Carenini and Johanna D. Moore, “Generating explanations in context,” in *Proceedings of the 1st international conference on Intelligent user interfaces*, 1993.

[10] J. D. Moore, “Responding to ‘HUH?’: answering vaguely articulated follow-up questions,” in *Proceedings of the SIGCHI conference on Human factors in computing systems: Wings for the mind*, 1989.

[11] M.C. Tanner and A.M. Keuneke, “Explanations in knowledge systems: the roles of the task structure and domain functional models,” *IEEE Expert*, vol. 6, no. 3, 1991.

[12] J. L. Weiner, “BLAH, a system which explains its reasoning,” *Artificial Intelligence*, vol. 15, no. 1-2, pp. 19-48, 1980.

[13] Agneta Eriksson, “Neat explanation of Proof Trees,” in *Proceedings of the 9th international joint conference on Artificial intelligence*, vol. 1, 1985.

[14] C. Millet and M. Gilloux, “A study of the knowledge required for explanation in expert systems,” in *Proceedings of Artificial Intelligence Applications*, 1989.

[15] J.W. Wallis and E.H. Shortliffe, "Customized explanations using causal knowledge," in *Rule-based Expert Systems*, Addison-Wesley, 1984.

[16] K. N. Papamichail and S. French, “Explaining and justifying the advice of a decision support system: a natural language generation approach,” *Expert Systems with Applications*, vol. 24, no. 1, pp. 35-48, 2003.

[17] Carenini and Moore, “Generating and evaluating evaluative arguments,” *Artificial Intelligence*, vol. 170, no. 11, pp. 925-952, 2006.

[18] T. R. Gruber and P. O. Gautier, “Machine-generated explanations of engineering models: A compositional modeling approach,” *IJCAI*, 1993.

[19] Patrice O. Gautier and Thomas R. Gruber, “Generating Explanations of Device Behavior Using Compositional Modeling and Causal Ordering,” *AAAI*, 1993.

[20] B. White and J. Frederiksen, “Causal model progressions as a foundation for Intelligent learning,” *Artificial Intelligence*, vol. 42, no. 1, pp. 99-155, 1990.

[21] F. Elizalde et al., “An MDP approach for explanation. Generation,” In *Workshop on Explanation-Aware Computing with AAAI*, 2007.

[22] O. Z. Khan et al., “Explaining recommendations generated by MDPs,” In *Workshop on Explanation Aware Computing*, 2008.

[23] S. Renooij and L. Van-DerGaa, “Decision making in qualitative influence diagrams,” In *Proceedings of the Eleventh International FLAIRS Conference*, pp. 410–414, 1998.

[24] C. Lacave et al. “Graphical explanations in bayesian networks,” In *Lecture Notes in Computer Science*, vol. 1933, pp. 122–129. Springer-Veralg, 2000.

[25] Andrew Ko and Brad Myers, “Extracting and answering why and why not questions about Java program output,” *ACM Transactions on Software Engineering and Methodology*, vol. 20, no. 2, 2010.

[26] F. Sørmo et al., “Explanation in case-based reasoning – perspectives and goals,” *Artificial Intelligence Review*, vol 24, no. 2005, pp. 109–143, 2005.

[27] A. Kofod-Petersen and J. Cassens, “Explanations and context in ambient intelligent systems, in *Proceedings of the 6th international and interdisciplinary conference on Modeling and using context*, 2007.

[28] C. P. Langlotz et al., “A methodology for generating computer-based explanations of decision-theoretic advice,” *Med Decis Making*, vol. 8, no. 4, pp. 290-303, 1988.

[29] H. Lieberman and A. Kumar, “Providing expert advice by analogy for on-line help,” in *Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, pp. 26-32, 2005.

[30] Baderet et al., “Explanations in Proactive Recommender Systems in Automotive Scenarios,” *Workshop on Decision Making and Recommendation Acceptance Issues in Recommender Systems Conference*, 2011.

[31] P. Pu and L. Chen, “Trust building with explanation interfaces,” *in: 11th International conference on Intelligent User Interfaces*, pp. 93-100, 2006.

[32] Baltrunas et al., “Context-Aware Places of Interest Recommendations and Explanations,” in *1st Workshop on Decision Making and Recommendation Acceptance Issues in Recommender Systems*, (DEMRA 2011), 2001.

[33] J. D. Lee K. A. See, Trust in Automation: Designing for Appropriate Reliance. *Human Factors*, vol 46, no 1, pp. 50-80, 2004.

[34] D. L. McGuinness et al., “Investigations into Trust for Collaborative Information Repositories: A Wikipedia Case Study,” in *Workshop on the Models of Trust for the Web*, 2006.

[35] I. Zaihrayeu, P. Pinheiro da Silva, and D. L. McGuinness, “IWTrust: Improving User Trust in Answersfrom the Web,” in *Proceedings of the 3rd International Conference on Trust Management*, pp. 384-392, 2005.

[36] B. Y. Lim, A. K. Dey, and D. Avrahami, “Why and why not explanations improve the intelligibility of context-aware intelligent systems,” in *Proceedings of the 27th international conference on Human factors in computing systems*, pp. 2119-2128, 2009.

[37] A. Glass, D. L. McGuinness, and M. Wolverton, “Toward establishing trust in adaptive agents,” in *Proceedings of the 13th international conference on Intelligent user interfaces*, pp. 227-236, 2008.

[38] J. J. Dijkstra, “On the use of computerised decision aids: an investigation into the expert system as persuasive communicator,” Ph.D. dissertation, 1998.

[39] S. Y. Rieh and D. R. Danielson, “Credibility: a multidisciplinary framework.” In Annual Review of Information Science and Technology, B. Cronin (Ed.), Vol. 41, pp. 307-364, 2007.

The URREF Ontology for Semantic Wide Area Motion Imagery Exploitation

Erik Blasch
Air Force Research Lab
Rome, NY, 13441
erik.blasch@rl.af.mil

Paulo C. G. Costa, Kathryn B. Laskey
C4I Center - George Mason University
Fairfax, VA, 22030
{pcosta, klaskey}@c4i.gmu.edu

Haibin Ling
Temple Universtity
Philadelphia, PA 19122
hbling@temple.edu

Genshe Chen
Information Fusion Tech.
Germantown, MD 20874
gchen@intfusiontech.com

Abstract— *Today's information fusion systems (IFSs) require common ontologies for collection, storage, and access to multi intelligence information. For example, ontologies are needed to represent the connections between physics-based (e.g. video) and text-based (e.g. reports) describing the same situation. Situation, user, and mission awareness are enabled through a common ontology. In this paper, we utilize the uncertainty representation and reasoning evaluation framework (URREF) ontology as a basis for describing wide-area motion imagery (WAMI) analysis to determine uncertainty attributes. As part of the Evaluation of Technologies for Uncertainty Representation Working Group (ETURWG), both the URREF and a WAMI challenge problem are available for research purposes. We provide an exemplar schema to link physics-based and text-based uncertainty representations to explore a common uncertainty demonstration.*

Keywords: *Hard-soft Information Fusion, Performance Evaluation, Uncertainty Reasoning, Knowledge Representation, Ontology, Measures of Effectiveness.*

I. INTRODUCTION

A fundamental goal of information fusion is to reduce uncertainty by combining information from multiple sources. When inputs come from disparate, heterogeneous sources, there is a need for a unified, common, and standardized semantic understanding of the information being fused, and also of the associated uncertainty. Ontologies [1] provide a means for such shared semantic understanding, thus enabling interoperability among systems in application domains such as command and control, emergency response, and information sharing [2]. In this work, we focus specifically on the need for interoperable representations of uncertainty. Figure 1, taken from [3], depicts the transformation of evidence from sensors through a fusion system to produce outputs reported to users. The fusion system employs uncertainty representation and uncertainty for machine processing and user interaction, refinement, and understanding [3, 4, 5, 6].

The evaluation of how uncertainty is processed is dependent on system-level metrics such as timeliness, accuracy, confidence, throughput, and cost [7], which also are information fusion quality of service (QoS) metrics [8]. Future large complex information fusion systems will require performance evaluation [9] and understanding of the connections between various metrics [10]. It is a goal of the *Evaluation of Technologies for Uncertainty Representation Working Group* (ETURWG) to formulate, test, and evaluate different approaches to uncertainty representation and reasoning. The URREF ontology provides a common semantic understanding to support evaluation of the uncertainty aspects of IF systems.

Information fusion system-level metrics include timeliness (how quickly the system can come to a conclusion within a specified precision level), accuracy (where can an object be

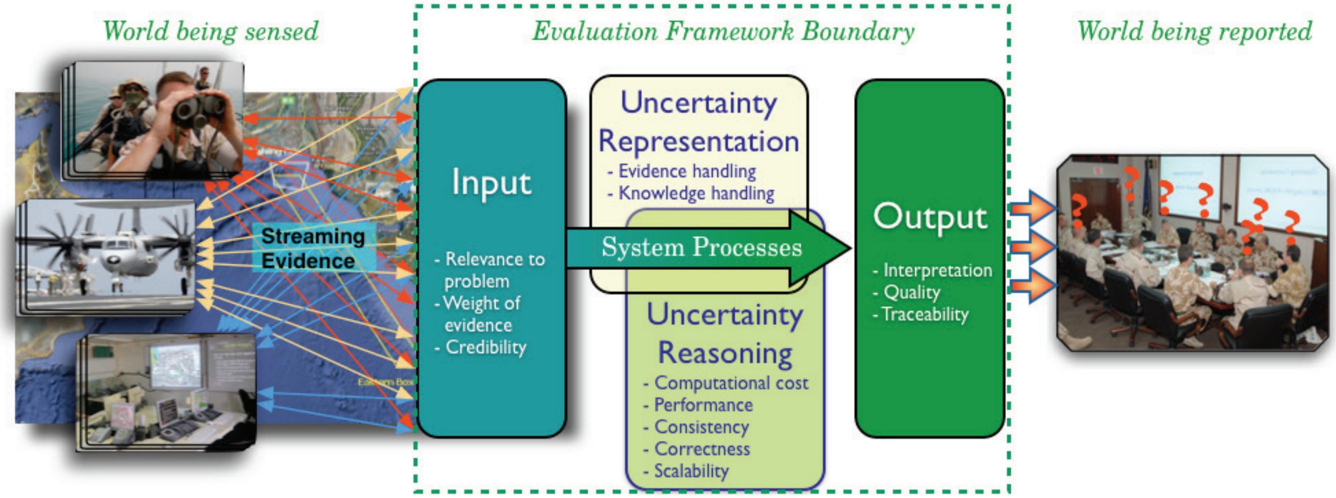


Figure 1 - Boundaries of the Uncertainty Representation and Reasoning Evaluation Framework [3].

found for a specified localization level), and confidence (what level of a probability match for a defined recall level). Clearly, different choices in uncertainty representation approaches will affect the achievable timeliness, accuracy, and confidence of a system, and therefore must be considered when evaluating both the system's performance as a whole [11] and the specific impact of the uncertainty handling approach. Yet, when evaluating timeliness (or any other system-level metrics), one will likely find some factors not directly related to the handling of uncertainty itself, such as object tracking and identification report updates (i.e., Level 1 fusion) [12, 13, 14], situation and threat assessment relative to scenario constraints (i.e., Level 2/3 fusion) [15], overall system architectures (e.g. centralized, distributed, etc.) [12], data management processes and feedback / input control processes (i.e., Level 4 fusion considerations) [16], and user-machine coordination based on operating systems (i.e., Level 5 fusion) [17], and others. In other words, evaluation of the uncertainty handling aspect of a fusion system is closely related to, yet distinct from, evaluation of the performance of the system overall.

Key to the various Data Fusion Information Group (DFIG) [18] levels of information fusion is *evaluation*. For example, there have been efforts in comprehensive tracking [19, 20], object classification [21], and situation awareness evaluation [22], which focus on measures of performance (MOPs). Future evaluations will include high-level information Measures of Effectiveness (MOEs) [23] that include uncertainty characterization [24].

Along with the URREF ontology, the ETURWG has also developed a series of use cases. The purpose of the use cases is to provide concrete realizations of the range of problems to which the URREF is intended to apply, to help ensure that the framework can address this range of problems. One use case is the use of *Wide-Area Motion Imagery* (WAMI) for Level 1 fusion [25, 26, 27, 28]. Other computer vision working groups [29] are exploring semantic technology with datasets that are not necessary focused on uncertainty, but have a rich set of ontologies and datasets for collaboration and comparisons.

The paper investigates the use of URREF for WAMI tracking. Section II explores the issues of uncertainty characterization and Section III, the uncertainty evaluation framework. Section IV presents a WAMI tracking use case using the URREF for timeliness, accuracy, and confidence. Section V provides and discussion and Section VI conclusions.

II. The Uncertainty Representation Problem

The Information Fusion community envisions effortless interaction between humans and computers, seamless interoperability and information exchange among applications, and rapid and accurate identification and invocation of appropriate services. As work with semantics and services grows more ambitious, there is increasing appreciation of the need for principled approaches to representing and reasoning under uncertainty. Here, the term "uncertainty" is intended to encompass a variety of aspects of imperfect knowledge, including incompleteness, inconclusiveness, vagueness, ambiguity, and others. The term "uncertainty reasoning" is meant to denote the full range of methods designed for representing and reasoning with knowledge when Boolean truth-values are unknown, unknowable, or inapplicable. Commonly applied approaches to uncertainty reasoning include probability theory [30], expert systems [31], fuzzy logic, subjective logic [32, 33], Dempster-Shafer theory, DSmT [34], and numerous other techniques.

To illustrate the challenges of evaluating uncertainty representation and reasoning in information systems, we consider below a few reasoning challenges faced within the World Wide Web domain that could be addressed by reasoning under uncertainty [1]. Uncertainty is an intrinsic feature of many of the required tasks, and a full realization of the World Wide Web as a source of processable data and information management services [3] demands formalisms capable of representing and reasoning under uncertainty such as:

- *Automated agents* (e.g., to exchange Web information);
- *Uncertainty-laden data*. (e.g., terrain information);
- *Non-sensory collected information* (e.g., human sources);
- *Dynamic composability* (e.g., Web Services); or
- *Information extraction* (e.g., indexing from large databases)

These problems are all related to information fusion, involve both text-based [35] and physics-based [36] data, and can be easily extrapolated to represent the more general classes of problems found in the sensor, data, and information fusion. A recent example of hard-soft fusion uses a controlled natural language (CNL) for data-to-decisions [37].

III. The Uncertainty Evaluation Framework

The uncertainty representation and reasoning evaluation framework (URREF) includes both hard (e.g. imaging, radar, video, etc.) and soft (e.g., human reports, software alerts, etc.) sources, which require integration for uncertainty MOEs.

Effectiveness relates to a system's capability to produce an effect. Benefits of fusion include providing locations of events, extending coverage, and reducing ambiguity and false alarms. The goal of the IFS is to support users in their tasks to provide refined information, reduce time and workload, or enable complete, accurate, and quality task completion. Effectiveness includes *efficiency*: doing things in the most economical way (good input to output ratio), *efficacy*: getting things done, (i.e., meeting objectives), and *correctness*: doing "right" things, (i.e., setting right thresholds to achieve an overall goal - the *effect*). The MOEs support system-level management and design verification, validation, testing, and evaluation. The URREF output step involves the assessment of how information on uncertainty is presented to the users and, therefore, how it impacts the quality of their decision-making process.

Key aspects of effectiveness include quality of service (QoS) and quality of information, also known as information quality (IQ). QoS relates to the ability of a system to provide timely and dependable data transmission. QI relates to the fitness for purpose of the content. QoS and QI metrics can be utilized for hard-soft semantic information fusion [38, 39, 40, 41]. Representing and measuring QI typically requires addressing the semantics of the domain and the problem. Thus, ontologies are an indispensible tool for measuring QI [42]. Because QI is inherently focused on uncertainty, probabilistic ontologies [43] are useful for representing QI metrics.

The URREF ontology, whose main concepts are depicted in Figure 2, is a first step towards a common framework for evaluating uncertainty in fusion systems. These core classes are subclasses of the top level class, which in OWL is called *Thing*. The core of the ontology is the *Criteria class*, which drives the development of the elements of the subclasses (Section II.B). The Uncertainty Classes were either taken or adapted from the Uncertainty Ontology developed by the W3C's URW3-XG [1]. The ontology must also be used as a high-level reference for defining the actual evaluation criteria items that will comprise a comprehensive uncertainty evaluation framework. Other main class definitions include:

- A *source class* is the origin of the information. A physical sensor is one important example of a source; where natural language inputs from a human is another.

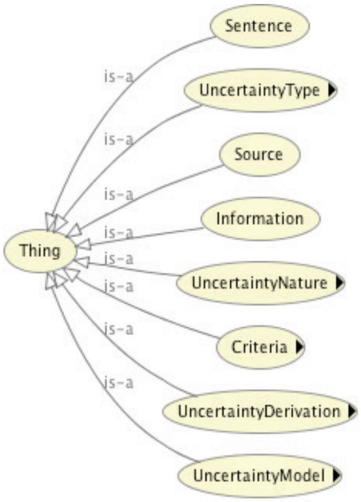


Figure 2 – The URREF ontology main classes.

- A *Sentence class* captures an expression in some logical language that evaluates to a truth-value (e.g., formula, axiom, assertion).
- A *Uncertainty Derivation* class refers to the way it can be assessed which is decomposed into:

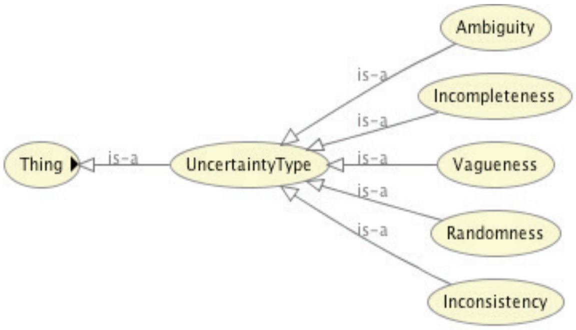


Figure 3 – URREF Ontology: Uncertainty Type Class.

1) *Objective Subclass*: (e.g., factual and repeatable derivation process).
2) *Subjective Subclass:* (e.g., a subject matter expert's (SME's) estimation).

- A *Uncertainty Model class* contains information on the mathematical theories for the representing and reasoning with the uncertainty types.

A. Uncertainty Type Class

Uncertainty Type is a concept that focuses on underlying characteristics of the information that make it uncertain. Its subclasses are *Ambiguity*, *Incompleteness*, *Vagueness*, *Randomness*, and *Inconsistency*, all depicted in Figure 3. These subclasses were based on the large body of work on evidential reasoning by David Schum [31].

B. Criteria Class

The *Criteria Class* is the main class of the URREF ontology, and it is meant to encompass all the different aspects that must be considered when evaluating information uncertainty handling in multi-sensor fusion systems. Figure 4 depicts the Criteria Class and its subclasses:

1) *Input Criteria:* encompasses the criteria that directly affect the way evidence is input to the system. It focuses on the source of input data or evidence, which can be tangible (sensing or physical), testimonial (human), documentary, or known missing.

- *Relevance to Problem* assesses how a given uncertainty representation is able to capture why a given input is relevant to the problem and what was the source of the data request.
- *Weight or Force of Evidence* measures how a given uncertainty representation is able to capture the degree to which a given input can affect the processing and output of the fusion system. Ideally, the weight should be an objective assessment and the representation approach must provide a means to measure the degree of impact of an evidence item with a numerical scale such as value of information [24].
- *Credibility,* also known as believability, comprises the aspects that directly affect a sensor (soft or hard) in its ability to capture evidence. Its subclasses are *Veracity*, *Objectivity*, *Observational Sensitivity*, and *Self-Confidence.*

2) *Representation Criteria:* encompasses the criteria that directly affect the way information is captured by and transmitted through the system. These criteria can also be called interfacing or transport criteria, as they relate to how the representational model transfers, passes, and routes information within the system.

- *Evidence Handling:* is a subclass of representation criteria that apply particularly to the ability of a given representation of uncertainty to capture specific characteristics of incomplete evidence that are available to or produced by the system. The main focus is on measuring the quality of the evidence by assessing how well this evidence is able to support the development of a conclusion. It has subclasses *Conclusiveness*, *Ambiguousness*, *Completeness*, *Reliability*, and *Dissonance.*

- *Knowledge Handling:* includes criteria intended to measure the ability of a given uncertainty representation technique to convey knowledge. Its subclasses are *Compatibility* and *Expressiveness* (which is further divided into the subclasses *Assessment*, *Adaptability*, and *Simplicity*)

3) *Reasoning Criteria:* contains criteria that directly affect the way the system transforms its data into knowledge. These can also be called process or inference criteria, as they deal with how the uncertainty model performs operations with information. It has the following subclasses:

- *Correctness* measures of the ability of the inferential process to produce results close to the truth. In cases where there is no ground truth to establish a correct answer (including a simulated ground truth), the representation technique can still be evaluated in terms of how its answers align with what is expected from a gold standard (e.g. subject matter experts, etc.).

- *Consistency* assesses of the ability of the inferential process to produce the same results when given the same data under the same conditions.

- *Scalability* evaluates how a representational technique performs on a class of problems as the amount of data or the problem size grows very large. Scalability could be broken down into additional sub-criteria.

- *Computational Cost* computes the number of resources required by a given representational technique to produce its results.

- *Performance* includes metrics to assess the contribution of the representational model toward meeting the functional requirements of an information fusion system. Other system architecture factors also affect these metrics. This criterion is divided into subclasses *Timeliness* and *Throughput*.

4) *Output Criteria* relates to the system's results and its ability to communicate it to its users in a clear fashion. It has the following subclasses:

- *Quality* serves to assess the informational assessment of the system's output. It includes *Accuracy* and *Precision* as subclasses. It is common to see in the literature the same concepts with different names. For example, accuracy sometimes is used as a synonym of precision; and sometimes precision is a refinement of accuracy. As one makes the granularity coarser, one can expect that the system will have a better accuracy. Precision can also be used to determine bounds on the certainty of the reported result.

- *Interpretation* refers to the degree to which the uncertainty representation and reasoning can be used to guide assessment, to understand the conclusions of the system and use them as a basis for action, and to support the rules for combining and updating measures.

The above concepts are being explored within the ETURWG, which is making use of this ontology (shown in Figure 4) to support the development of uncertainty evaluation criteria over a set of information fusion use cases. The interested reader should refer to the group's website for more specific details (http://eturwg.gmu.edu). Note that the URREF ontology is not supposed to be a definitive reference for evaluation criteria, but simply an established baseline that is

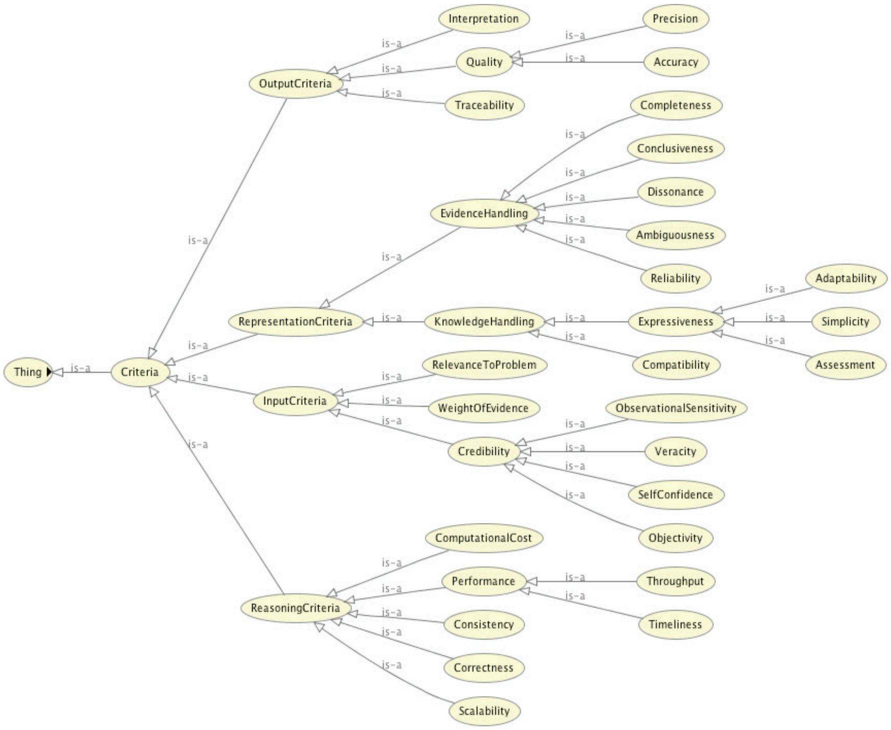


Figure 4 – URREF Ontology: Criteria Class.

coherent and sufficient for its purposes. This approach privileges the pragmatism of having a good solution against having an "ideal" but usually unattainable solution. For instance, a definitive reference would involve having universally accepted definitions and usage for terms such as "Precision." This is clearly infeasible. The approach also takes into consideration that more important than naming a concept is to ensure that it is represented clearly and distinctly within the ontology so as to ensure the consistency for such applications as hard-soft fusion.

To assure utility and acceptability of the URREF ontology, most of its concepts have been drawn from seminal work in related areas such as uncertainty representation, evidential reasoning, and performance evaluation. The ontology has built on the URW3 uncertainty ontology [1]. Also, the structure and viewpoint adopted in the ontology development have been tuned to addressing the uncertainty evaluation problem and its associated perspective (e.g. how information is handled within a fusion system). Next, we present simultaneous tracking and identification (ID) application using the URREF.

IV. EXAMPLE – WAMI

Wide area motion imagery (WAMI) systems provide imagery and video surveillance of large areas.

A. Schema

A *schema* for image processing is shown in Figure 5 for the Cursor on Target (CoT) program [44]. As detailed, the schema provides target type and identification (ID) allegiance, time stamps, and coordinate locations (much as the DFIG level 1 object assessment information of target track and ID information). While the schema is simple, and worked well [45], for purposes of information transmission, processing, exploitation, and dissemination, future developments could include uncertainty fields from the URREF ontology. It is important to assess which semantic content is most relevant for operational information fusion management and systems design.

```
<?xml version='1.0' standalone='yes'?>
<event version="2.0"
    uid="J-01334"
    type="a-h-A-M-F-U-M"
    time="2005-04-05T11:43:38.07Z"
    start="2005-04-05T11:43:38.07Z"
    stale="2005-04-05T11:45:38.07Z" >
  <detail>
  </detail>
  <point lat="30.0090027" lon="-85.9578735" ce="45.3"
             hae="-42.6" le="99.5"  />
</event>
```

Figure 5 – Cursor on Target Schema [46]

In order to determine what uncertainty attributes can be added to such a message passing schema, there are three issues (1) what, (2) how much, and (3) which ones. For the case of physics-based (video) and textual-based reports, we need to determine what semantic content could be useful. One simple case is that either a human analyst can report a "friendly" in the *uid* field, or a machine tracker could extract the information from the video to update the *uid* field of "friendly". One example of "friendly" could be from extracted text and video exploitation of a blue vehicle. What is obviously missing from the CoT schema is some notion of uncertainty with the measurements and information as to the confidence, timeliness, and position accuracy. While the entire URREF cannot, and should not, be considered for the schema updates, as a message passing service for the ontology, the first issue is to calculate possible uncertainty metrics that could go into the schema.

B. Metrics to Support the URREF Ontology

For the metrics available in the Cursor on Target Schema, we seek measures of confidence, accuracy, and timeliness, as related to *uid*, *time*, and *point*; respectively.

- Credibility / Confidence: evaluates the ability to discern an object based on a known target. Classification is the target match, while identity is target allegiance. If targets are of known entities, it can be assumed that the targets not classified could pose an ID uncertainty. Using a Bayesian approach for this example, we determine the relative probability from the likelihood values of the object, versus of target clutter $\ell_{O|c}$, where c_j is for $j = 1, ..., n$ clutter types:

$$Pr_{O|c} = \frac{[\ell_{O|C}]}{\Sigma_{c_j \in C} [\ell_{O|c_j}]} \quad (1)$$

Using plausibility, uncertainty is everything unknown

$$U_L = 1 - Pr_{O|c} \quad (2)$$

- Timeliness: evaluates when the system knows enough information to make a decision versus when it was collected. For the purpose of this analysis we simulate the deadtime for an *input time delay* (TD_i) for a decision i, as related to the user achieving a control decision [46]. Likewise, in the action selection requires time as modelled as an output time delay (TO_i). The updated state-space representation is:

$$\dot{\boldsymbol{x}}(t) = -A\boldsymbol{x}(t) + B\,u(t - TD_i)$$
$$\mathbf{y}(t) = C\,\mathbf{x}(t - TO_i) + D\,\mathbf{u}(t) \quad (3)$$

To determine the estimation parameters of A and B, as well as the output analysis of C and D, we model the importance of the information processing as related to the cognitive observe-orient-decide-act (OODA) functions. Uncertainty is defined as the decision time difference of arrival:

$$U_T = \mathbf{x}(t - TO_i) - \mathbf{x}(t - TD_i) \quad (4)$$

- Accuracy: evaluates how the real world track estimates from the measurements compare to the ground truth. For the purpose of this analysis, the real world is reduced to a specified track estimate x_M, as related to ground truth x_T. Using a root-mean square error, we have:

$$U_L = \sqrt{(x_M - x_T)^2 + (y_M - y_T)^2} \quad (5)$$

Accuracy can be determined versus the ability to track a target exactly: 1 - U_L. Other aspects could include track purity for track-to-track association [46] for situation awareness including:

- Specificity: evaluates how much of the real world clutter is reduced such as reducing the false alarms. While we do not simulate, we can deduce from the track confidence.

- Situation Completeness: evaluates how much of the real world the system knows. For the purpose of this analysis the real world is reduced to a specified region of space (the volume of interest, VOI) during a given time interval (the time interval of interest, TOI).

C. Wide Area Motion Imagery Example

WAMI has gained in popularity as it affords advanced capabilities in persistence, increased track life, and situation awareness, but it also poses new challenges such as low frame updates (timeliness) [47, 48]. Leveraging developments from computer vision [49, 50, 51, 52, 53], methods are being applied as part of the ETUWG [29]. The persistence coverage affords such methods as multiple object and group tracking [54, 55, 56], road assessment and tracking [57, 58], contextual tracking [59, 60], and advances in particle filtering [59, 61]. Because of the numerous objects and their movements, there are opportunities for linear road tracking, but also there is a need for nonlinear track evaluation [62] such as the randomized unscented transform (RUT) filter [63] for accuracy assessment. These issues will be important for future work.

We utilize the results from a WAMI tracker for track location accuracy, the pixels on target for classification for target identity (e.g. credibility), and the timeliness to make a decision. We are tracking four targets with an on-road analysis with a nominated target of interest, as shown in Figure 6. Vehicles turning off road are not considered as part of the user defined targets of interest. Note: the entire Columbus Large Image Format (CLIF) WAMI image collection has been presented in previous papers with discussions with the entire video data set [27] (see the ETURWG website).

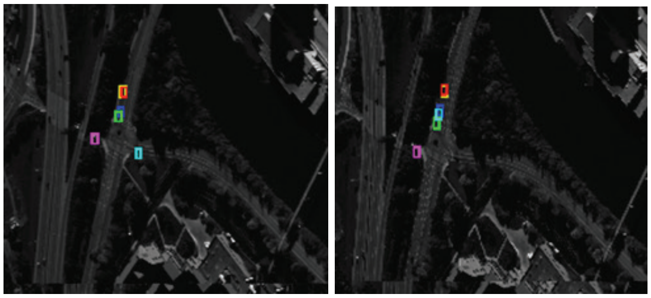


Figure 6 – WAMI Tracking.

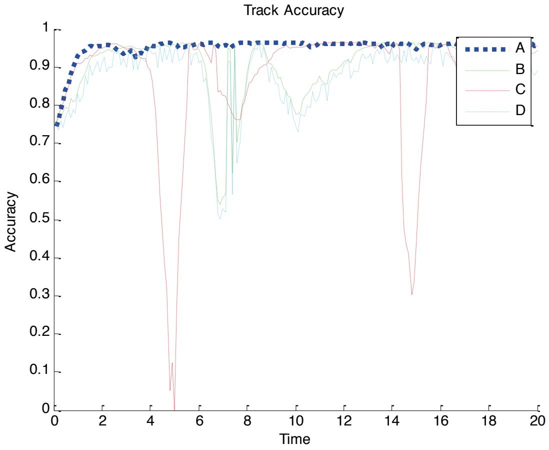


Figure 7 – Target Accuracy

Figure 7 plots the target accuracy (which is the inverse of the typical plots that show the target tracking error). Figure 8 combines the track accuracy in a unified display plot showing the target confidence (uid) and the accuracy. The confidence is shown as solid lines and the timeliness presented as the black humps where the time intervals are shown as: orient (t = 2.5-5), observe (t = 5-10), decide (t = 10-13) and act (t =13-18) time steps.

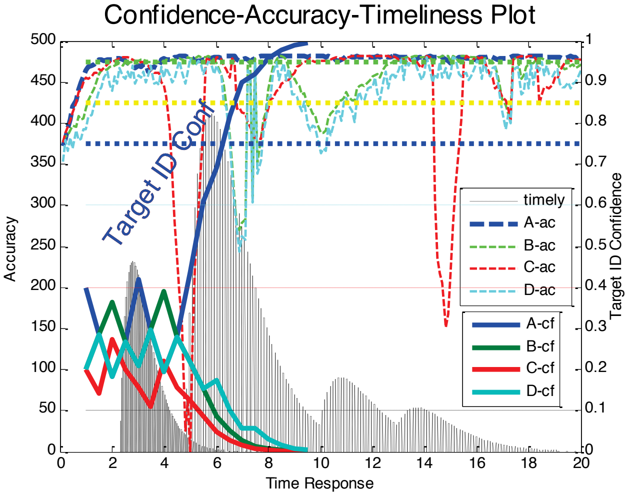


Figure 8 – Confidence-Accuracy-Timeliness Results.

Using the above information, we combine the credibility /confidence, accuracy, and timeliness (CAT) for a semantic notion of fused uncertainty in Figure 9 (where the normalized values are $U_T = U_C + U_T + U_A$). Together, the combined uncertainty could be a ontology field in an updated CoT schema to give the user a quality assessment of a machine processed semantic representation of uncertainty.

V. Discussion

Figure 9 shows a case for unified uncertainty estimation and is meant for discussion. Given the choice to utilize the URREF ontology, there are issues associated with choosing an ontology representation that can work within a message passing schema. If only one field was available, say ut, then is it appropriate to normalize the uncertainty and combine for purposes of the schema? For this case, only one target was nominated (like the CoT program), from which we see that the combined evidence supports a reduction in uncertainty; namely decreased track error, increased plausibility and hence ruling out the uid error, and the timeliness in decision making.

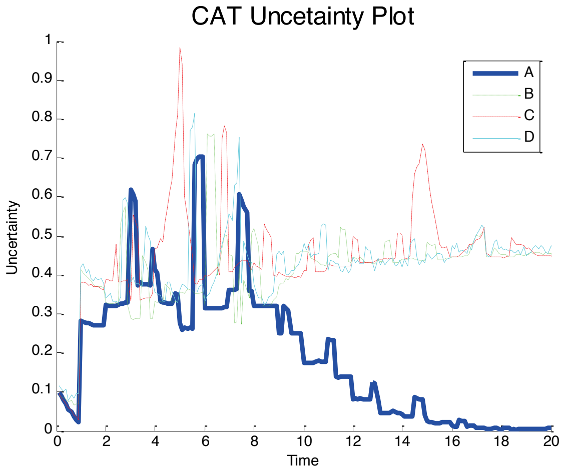


Figure 9 – Objective Semantic of Uncertainty.

VI. Conclusions

Characterizing the uncertainty is important in information fusion (IF) processes. Evaluation of IF systems presents various challenges due to the complexity of fusion systems and the sheer number of variables influencing their performance. Developing the operational semantics will include issues of representation, reasoning, and policy which need to be considered for command and control [64]. Representing uncertainty has an overall impact on system performance that is hard to quantify or even to assess from a qualitative viewpoint. The ETURWG technical considerations unearthed many issues that demand a common understanding that is only achievable by a formal specification of the semantics involved [65, 66].

In the paper, we utilized the current URREF ontology in relation to an established schema (Cursor on Target) to support the development of a specific use case for wide-area motion imagery (WAMI) simultaneous tracking and identification. We also presented a visual analytic method for uncertainty metrics and analytics. Future work includes group tracking, activity analysis, hard-soft fusion, and contextual understanding.

More specific requirements to evaluate a set of use cases and associated data sets designed by the ETURWG are accessible through our webpage [http://eturwg.c4i.gmu.edu]. Although it is clear that the URREF ontology is not a definitive reference for all types of information fusion activities, it has proven to be a discussion towards a common framework.

Acknowledgment

The authors recognize the contributions of the members of the International Society of Information Fusion's ETURWG in formulating the URREF ontology, which was discussed in several of its meetings and includes the WAMI use case.

References

[1] K.B. Laskey, K.J. Laskey, P.C.G. Costa, M. Kokar, T. Martin, and T. Lukasiewicz (eds.) "Uncertainty Reasoning for the World Wide Web: W3C Incubator Group Report." World Wide Web Consortium. Available at http://www.w3.org/2005/Incubator/urw3/XGR-urw3/.

[2] E. P. Blasch, E. Bosse, and D. A. Lambert, *High-Level Information Fusion Management and Systems Design*, Artech House, 2012.

[3] P. C. G. Costa, K. B. Laskey, E. Blasch and A-L. Jousselme, "Towards Unbiased Evaluation of Uncertainty Reasoning: The URREF Ontology," *Int. Conf. on Info Fusion*, 2012.

[4] E. P. Blasch and P. Hanselman, "Information Fusion for Information Superiority," *IEEE Nat. Aerospace and Electronics Conference,* 2000.

[5] E. Blasch, "Level 5 (User Refinement) issues supporting Information Fusion Management" *Int. Conf. on Info Fusion*, 2006.

[6] E. Blasch, "User refinement in Information Fusion", Chapter 19 in *Handbook of Multisensor Data Fusion 2^{nd} Ed,* (Eds.) D. Hall, and J. Llinas, CRC Press, 2008.

[7] D. L. Hall and J. M. Jordan, *Human Centered Information Fusion*, Artech House, 2010.

[8] E. Blasch, M. Pribilski, B. Roscoe, *et. al.*, "Fusion Metrics for Dynamic Situation Analysis," *Proc SPIE,* Vol. 5429, Aug 2004.

[9] E. Blasch, I. Kadar, J. Salerno, M. M. Kokar, S. Das, G. M. Powell, D. D. Corkill, and E. H. Ruspini, "Issues and Challenges in Situation Assessment (Level 2 Fusion)," *J. of Advances in Information Fusion,* Vol. 1, No. 2, pp. 122 - 139, Dec. 2006.

[10] E. Waltz and J. Llinas, "System Modeling and Performance Evaluation," Ch 11 in *Multisensor Data Fusion Systems*, Artech House 1990.

[11] J. Llinas, "Assessing the Performance of Multisensor Fusion Processes," Ch 20 in *Handbook of Multisensor Data Fusion*, (Eds.) D. Hall and J. Llinas, CRC Press, 2001.

[12] P. Hanselman, C. Lawrence, E. Fortunano, B. Tenney, and E. Blasch, "Dynamic Tactical Targeting," *Proc. of SPIE,* Vol. 5441, 2004.

[13] K. C. Chang and Robert M. Fung, "Target Identification with Bayesian Networks in a Multiple Hypothesis Tracking System", *Optical Engineering*, Vol. 36, No. 3, pp.684-691, March, 1997.

[14] E. Blasch, *Derivation of A Belief Filter for High Range Resolution Radar Simultaneous Target Tracking and Identification*, Ph.D. Dissertation, Wright State University, 1999.

[15] C. Yang and E. Blasch, "Pose Angular-Aiding for Maneuvering Target Tracking", *Int. Conf. on Info Fusion*, July 2005.

[16] G. Chen, D. Shen, C. Kwan, J. Cruz, M. Kruger, and E. Blasch, "Game Theoretic Approach to Threat Prediction and Situation Awareness," *J. of Advances in Information Fusion,* Vol. 2, No. 1, 1-14, June 2007.

[17] E. Blasch, I. Kadar, K. Hintz, J. Biermann, C. Chong, and S. Das, "Resource Management Coordination with Level 2/3 Fusion Issues and Challenges," *IEEE Aerospace and Electronic Systems Magazine,* Vol. 23, No. 3, pp. 32-46, Mar. 2008.

[18] E. Blasch, "Sensor, User, Mission (SUM) Resource Management and their interaction with Level 2/3 fusion" *Int. Conf. on Info Fusion*, 2006.

[19] E. Blasch and S. Plano, "DFIG Level 5 (User Refinement) issues supporting Situational Assessment Reasoning," *Int. Conf. on Info Fusion,* 2005.

[20] K. C. Chang, Z. Tian, S. Mori, and C-Y. Chong, "MAP Track Fusion Performance Evaluation," *Int. Conf. on Information Fusion*, 2002.

[21] E. P. Blasch, O. Straka, J. Duník, and M. Šimandl, "Multitarget Performance Analysis Using the Non-Credibility Index in the Nonlinear Estimation Framework (NEF) Toolbox," *Proc. IEEE Nat. Aerospace Electronics Conf (NAECON)*, 2010.

[22] R. Carvalho and KC Chang, "A Performance Evaluation Tool for Multi-Sensor Classification Systems," *Int'l Conf. on Information Fusion*, 2009.

[23] J. Salerno, E. Blasch, M. Hinman, and D. Boulware, "Evaluating algorithmic techniques in supporting situation awareness," *Proc. of SPIE,* Vol. 5813, April 2005.

[24] E. Blasch, P. Valin, E. Bossé, "Measures of Effectiveness for High-Level Fusion," *Int. Conference on Information Fusion*, 2010.

[25] P.C.G. Costa, R.N. Carvalho, K.B. Laskey, and C.Y.Park, “Evaluating Uncertainty Representation and Reasoning in HLF systems”, *Int. Conference on Information Fusion*, 2011.

[26] O. Mendoza-Schrock, J. A. Patrick, and E. P. Blasch, “Video image registration evaluation for a layered sensing environment.” in IEEE Nat. Aerospace Electronics Conf. (NAECON), 2009.

[27] H. Ling, Y. Wu, E. Blasch, G. Chen, H. Lang, and L. Bai, “Evaluation of visual tracking in extremely low frame rate wide area motion imagery,” *Int. Conf. on Information Fusion*, 2011.

[28] R. Pelapur, S. Candemir, F. Bunyak, M. Poostchi, G. Seetharaman, and K. Palaniappan, “Persistent Target Tracking Using Likelihood Fusion in Wide-Area and Full Motion Video Sequences,” *Int. Conf. on Info. Fusion*, 2012.

[29] P. Liang, G, Teodoro, H. Ling, E. Blasch, G. Chen, and L. Bai, “Multiple Kernel Learning for Vehicle Detection in Wide Area Motion Imagery,” *Int. Conf. on Info Fusion*, 2012.

[30] E. Blasch, G. Seetharaman, M. Talbert, K. Palaniappan, and H. Ling, “Key Elements to Support Layered Sensing Dismount Tracking,” *Proc. NATO SET 178-RWS 017 Workshop on Detection of Dismounted Combatants*, Sept. 2011.

[31] D. Schum, “The Evidencial Foundations of Probabilistic Reasoning,” Northwestern University Press, 1994.

[32] P. Walley, “Measures of Uncertainty In Expert Systems,” Artificial Intelligence, 83(1), May 1996, pp. 1-58.

[33] L. M. Kaplan, S. Chakraborty, and C. Bisdikian, “Subjective Logic with Uncertain Partial Observations,” *Int. Conf. on Information Fusion*, 2012.

[34] A. J∅sang and R. Hankin, “Interpretation and Fusion of Hyper Opinions in Subjective Logic,” *Int. Conf. on Information Fusion*, 2012.

[35] J. Dezert, D. Han, Z-g. L, and J-M. Tacnet, “Hierarchical DSmP transformation for decision-making under uncertainty,” *Int. Conf. on Information Fusion*, 2012.

[36] A. Auger and J. Roy, “Expression of Uncertainty in Linguistic Data,” *Int. Conference on Information Fusion*, 2008.

[37] H. Ling, L. Bai, E. Blasch, and X. Mei, “Robust infrared vehicle tracking across target pose change using L1 regularization,” *Int’l Conf. on Information Fusion*, 2010.

[38] A. Preece, D. Pizzocaro, D. Braines, D. Mott, G. de Mel, and T. Pham, “Integrating hard and soft Information Sources for D2D Using Controlled Natural Language,” *Int. Conf. on Information Fusion*, 2012.

[39] W. Perry, D. Signori, and J. Boon, “Exploring the Information Superiority: A Methodology for Measuring the Quality of Information and its Impact on Shared Awareness,” *RAND Corporation*, 2004.

[40] M. E. Johnson and K. C. Chang, “Quality of Information for Data Fusion in Net Centric Publish and Subscribe Architectures,” *Int. Conf. on Information Fusion*, 2005.

[41] M. A. Hossain, P. K. Atrey, and A. El Saddik, “Modeling Quality of Information in Multi-Sensor Surveillance Systems,” *IEEE Int. Conf. on Data Engineering Workshop*, 2007.

[42] C. Bisdikian and L. M. Kaplan, M. B. Srivastava, D. J. Thornley, D. Verma, and R. I. Young, “Building principles for a quality of information specification for sensor fusion,” *Int. Conf. on Information Fusion*, 2009.

[43] E. P. Blasch, “Ontological Issues in Higher Levels of Information Fusion: User Refinement of the Fusion Process,” *Int. Conf. on Info Fusion,* 2003.

[44] R. Carvalho, P. Costa, K. Laskey, and KC Chang, “PROGNOS: Predictive Situational Awareness with Probabilistic Ontologies,” *Int’l Conf. on Information Fusion*, 2010.

[45] M J. Kristan, J. T. Hamalainen, D. P. Robbins, and P. J. Newer, , “Cursor-on-Target Message Router User’s Guide,” *MITRE Product – MP090284,* 2009. (accessed on line)

[46] R. A. Shulstad, “Cursor on Target: Inspiring Innovation to Revolutionizing Air Force Command and Control,” *Air and Space Power Journal.* Vol. 4, Dec., 2011.

[47] E. Blasch, R. Breton, P. Valin, and E. Bosse, “User Information Fusion Decision Making Analysis with the C-OODA Model,” *Int. Conf. on Info Fusion*, 2011.

[48] E. Blasch and P. Valin, “Track Purity and Current Assignment Ratio for Target Tracking and Identification Evaluation,” *Int. Conf. on Info Fusion*, 2011.

[49] K. Palaniappan, F. Bunyak, P. Kumar, et al., “Efficient feature extraction and likelihood fusion for vehicle tracking in low frame rate airborne video,” *Int. Conf. on Information Fusion*, 2010.

[50] K. Palaniappan, R. Rao, G. Seetharaman, "Wide-area persistent airborne video: Architecture and challenges," *Distributed Video Sensor Networks: Research Challenges and Future Directions*, Springer, Part V, Chapter 24, pp. 349 – 371, 2011.

[51] Y. Wu, H. Ling, E. Blasch, G. Chen, and L. Bai, “Visual Tracking based on Log-Euclidean Riemannian Sparse Representation,” *Int. Symp. on Adv. in Visual Computing - Lecture Notes in Computer Science*, 2011.

[52] X. Mei, H. Ling, Y. Wu, E. P. Blasch, and L. Bai, “Minimum Error Bounded Efficient L1 Tracker with Occlusion Detection,” *IEEE Computer Vision and Pattern Recognition*, 2011.

[53] Y. Wu, E. Blasch, G. Chen, L. Bai, and H. Ling, “Multiple Source Data Fusion via Sparse Representation for Robust Visual Tracking,” *Int. Conf. on Info Fusion*, 2011.

[54] Y. Wu, J. Wang, J. Cheng, H. Lu, E. Blasch, L. Bai, and H. Ling, “Real-Time Probabilistic Covariance Tracking with Efficient Model Update,” *IEEE Trans. on Image Processing*, **21**(5):2824-2837, 2012.

[55] X. Zhang, W. Li, W. Hu, H. Ling, and S. Maybank, “Block covariance based L1 tracker with a subtle template dictionary,” *Pattern Recognition*, 2012.

[56] E. P. Blasch and T. Connare, “Improving Track maintenance Through Group Tracking,” *Proc of the Workshop on Estimation, Tracking, and Fusion; A Tribute to Yaakov Bar Shalom*, 360 –371, May 2001.

[57] T. Connare, E. Blasch, J. Schmitz, F. Salvatore, and F. Scarpino, “Group IMM tracking utilizing Track and Identification Fusion,” *Proc. of the Workshop on Estimation, Tracking, and Fusion; A Tribute to Yaakov Bar Shalom,* Monterey, CA, 205 -220, May 2001.

[58] M. Wieneke and W. Koch, “A PMHT Approach for extended Objects and Object Groups,” *IEEE T. Aerospace and Electronic Systems*, vol. 48, no. 3, July 2012.

[59] C. Yang and E. Blasch, “Fusion of Tracks with Road Constraints,” *J. of. Advances in Information Fusion,* Vol. 3, No. 1, 14-32, June 2008.

[60] X. Shi, H. Ling, E. Blasch, and W. Hu, “Context-Driven Moving Vehicle Detection in Wide Area Motion Imagery,” *Int’l Conf on Pattern Recognition (ICPR),* 2012.

[61] E. D. Marti, J. Garcia, and J. L Crassidis, “Improving Multiple-Model Context-Aided Tracking through an Autocorrelation Approach,” *Int. Conf. on Info Fusion*, 2012.

[62] L. Snidaro, I. Visentini, K. Bryan, and G. L. Foresti, “Markov Logic Network for context integration and situation assessment in a maritime domain,” *Int. Conf. on Info Fusion*, 2012.

[63] F. Papi. M. Podt, Y. Boers, G. Battistello, and M. Ulmka, “Constraints Exploitation for Particle Filtering based Tagret Tracking,” *Int. Conf. on Info Fusion*, 2012.

[64] Z. Duan, X. Rong Li, and J. Ru, “Design and Analysis of Linear Equality Constrained Dynamic Systems,” *Int. Conf. on Information Fusion*, 2012.

[65] O. Straka, J. Duník, and M. Šimandl, “Randomized Unscented Kalman Filter in Target Tracking,” *Int. Conf. on Information Fusion*, 2012.

[66] L. Scholz, D. Lambert, D. Gossink, and G. Smith, “A Blueprint for Command and Control: Automation and Interface,” *Int. Conf. on Info Fusion*, 2012.

[67] E. Blasch, P. C. G. Costa, K. B. Laskey, D. Stampouli, G. W. Ng, J. Schubert, R. Nagi, and P Valin, “Issues of Uncertainty Analysis in High-Level Information Fusion – Fusion2012 Panel Discussion,” *Int. Conf. on Info Fusion*, 2012.

[68] P. C. G. Costa, E. P. Blasch, K. B. Laskey, S. Andler, J. Dezert, A-L. Jousselme, and G. Powell, “Uncertainty Evaluation: Current Status and Major Challenges – Fusion2012 Panel Discussion,” *Int. Conf. on Info Fusion*, 2012.

Using Semantic Web Technologies to Develop Intrinsically Resilient Energy Control Systems

Frederick Sheldon and Daniel Fetzer
Oak Ridge National Laboratory
Oak Ridge, TN 37831, U.S.A.
{sheldonft, fetzerdt}@ornl.gov

Jingshan Huang
University of South Alabama
Mobile, AL 36688, U.S.A.
huang@southalabama.edu

Jiangbo Dang and Dong Wei
Siemens Corporation, Corporate Research and Technology
Princeton, NJ 08540, U.S.A.
{jiangbo.dang, dong.w}@siemens.com

David Manz
Pacific Northwest National Laboratory
Richland, WA 99354, U.S.A.
david.manz@pnnl.gov

Thomas Morris
Mississippi State University
Mississippi State, MS 39762, U.S.A.
morris@ece.msstate.edu

Jonathan Kirsch and Stuart Goose
Siemens Corporation, Corporate Research and Technology
Berkeley, CA 94704, U.S.A.
{jonathan.kirsch, stuart.goose}@siemens.com

***Abstract*—To preserve critical energy control functions while under attack, it is necessary to perform comprehensive analysis on root causes and impacts of cyber intrusions without sacrificing the availability of energy delivery. We propose to design an intrinsically resilient energy control system where we extensively utilize Semantic Web technologies, which play critical roles in knowledge representation and acquisition. While our ultimate goal is to ensure availability/resiliency of energy delivery functions and the capability to assess root causes and impacts of cyber intrusions, the focus of this paper is to demonstrate a proof of concept of how Semantic Web technologies can significantly contribute to resilient energy control systems.**

***Index Terms*—cybersecurity, energy control system, ontology, knowledge base, semantic annotation, data integration.**

I. Introduction

Our energy infrastructure depends on energy delivery systems comprised of complex and geographically dispersed network architectures with vast numbers of interconnected components. These systems provide critical functions to provide information and automated control over a large, complex network of processes that collectively ensure reliable and safe production and distribution of energy. The energy utilities are modernizing these vast networks with millions of smart meters, high speed sensors, advanced control systems, and a supporting communications infrastructure. This additional complexity brings benefits, but also increases the risks of cyber attacks that could potentially disrupt our energy delivery. These systems must maintain high availability and reliability even when under attack. After a security incident has been detected, the incident response team needs the ability to investigate and determine the root cause, attack methods, consequences, affected assets, impacted stakeholders, and other information in order to inform an effective response. The response team needs this information in the short term in order to contain or eradicate the attack, recover compromised equipment, and restore normal operation. The team also needs to determine counter-measures to prevent recurrence and possibly collect evidence to legally prosecute the offenders. This analysis and response must be done without interrupting the availability of the energy delivery systems.

To address the aforementioned challenges, this paper presents the design and architecture of *InTRECS*, an InTrinsically Resilient Energy Control System. The ultimate goal of InTRECS is to provide tools and technologies to ensure the availability/resiliency of energy delivery functions, along with the capability to assess root causes and impacts of cyber intrusions. To meet these goals, InTRECS extensively applies Semantic Web technologies, including cybersecurity domain ontologies, a comprehensive knowledge base, and semantic data annotation & integration techniques. Semantic Web technologies are built upon ontologies, which are formal, declarative knowledge models and have been shown to play critical roles in knowledge representation and acquisition.

In this paper, we argue that applying Semantic Web technologies in InTRECS affords several benefits compared to typical approaches that utilize relational databases:

- While relational databases focus on *syntactic* representation of data and lack the ability to explicitly encode semantics, Semantic Web technologies support rich *semantic* encoding, which is critical in automated knowledge acquisition.
- Powerful tools exist for capturing and managing ontological knowledge, including an abundance of reasoning tools readily supplied for ontological models, making it much more convenient to query, manipulate, and reason over available data sets. As a result, semantics-based queries, instead of SQL queries, are made possible.
- Advances in an energy delivery system (EDS) require changes to be made regularly regarding underlying data models. In addition, more often than not, it is preferable to represent data at different levels and/or with different abstractions. There are no straightforward methods for performing such updates if relational models are adopted.
- Semantic Web technologies better enable EDS researchers to append additional data into repositories in a more

This manuscript has been authored by contractors of the U.S. Government (USG) under contract DE-AC05-00OR22725. Accordingly, the USG retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for USG purposes.

> flexible and efficient manner. The formal semantics encoded in ontologies makes it possible to reuse data in unplanned and unforeseen ways, especially when data users are not data producers, which is now very common.

While our ultimate goal is to ensure availability/resiliency of energy delivery functions and the capability to assess root causes and impacts of cyber intrusions, the focus of this paper is to demonstrate a proof of concept of how Semantic Web technologies can significantly contribute to resilient energy control systems. The rest of the paper is organized as follows. Section II gives a brief review on related research in ontologies and semantic annotation & integration, respectively. Section III describes the overall architecture of InTRECS, followed by methodology details for developing domain ontologies & knowledge base and performing data annotation & integration. Section IV demonstrates our preliminary experimental results. Finally, Section V concludes with future research directions.

II. Related Work

A. Ontologies in Energy Delivery Control and Cybersecurity

Energy delivery control systems comprise complex network architectures that may contain hundreds of specialized cyber components and may extend across wide geographical regions. Cyber attack investigation involves examining large volumes of data from heterogeneous sources. Researchers are facing the challenge of how to maintain the integrity of data derived from diverse sources across distributed geographic areas ([1-7]). These research efforts have resulted in various ad-hoc proprietary formats for storing and analyzing data and maintaining respective metadata. Different parties are likely to adopt different formats according to specific needs. Therefore, the seamless communication among different parties, along with the knowledge sharing and reuse that follow, become a non-trivial problem. Turnitsa and Tolk [8] discussed in depth multi-resolution, multi-scope, and multi-structure challenges during data exchange between different models.

Semantic Web technologies that are based on domain ontologies can render tremendous help. Ontologies are declarative knowledge models, defining essential characteristics and relationships for specific domains of interest. As a semantic foundation, ontologies greatly help domain experts to formally define domain knowledge in terms of *data semantics* (intended meanings) rather than *data syntax* (forms in which data are represented). Reasons for developing ontologies include, but not limited to: (i) to share domain information among people and software; (ii) to enable reuse of domain knowledge; (iii) to analyze domain knowledge and make it more explicit; and (iv) to separate domain knowledge from its implementation. There exist some domain ontologies in cybersecurity and related areas, e.g., Intrusion Detection System Ontology [1], Network Security Ontology [2], Process Control Ontology [4], INSPIRE Ontology [5], and GE SADL Host Defense Ontology [7]. These ontologies provide metadata and standard terminologies in respective domains.

B. Semantic Data Annotation & Integration

Semantic data annotation & integration can bring critical impacts and benefits to data analysis and management. Semantic annotation (tagging) systems can be divided into manual, semi-automatic, and automatic ones [9]. In manual tagging systems (Sema-Link [10] for example), users employ controlled vocabularies from some ontology to tag documents. Such a manual process is time-consuming and requires deep domain expertise, in addition to the inconsistency issue. Semi-automatic tagging systems improve manual tagging systems by automatically parsing documents and recommending potential tags. Human annotators only need to select tags from candidates suggested by the system. Automatic semantic tagging systems offer further improvement by parsing and tagging documents with ontological concepts and instances in a fully automatic way. Zemanta [11] is such an example. By suggesting contents from various sources, such as Wikipedia, YouTube Flickr, and Facebook, Zemanta disambiguates terms and maps them to the Common Tag Ontology [12]. Dang et al. have developed one of the largest comprehensive, domain-independent ontological knowledge base, UNIpedia+ [13], which covers around 11 million named English entities. Based on UNIpedia+, they further developed an automatic tagging system [14] to produce semantically linked tags for given data. The information system architecture in the Los Angeles Smart Grid project [15] enabled analytical tools and algorithms to forecast energy load and identify load curtailment response through semantically meaningful data.

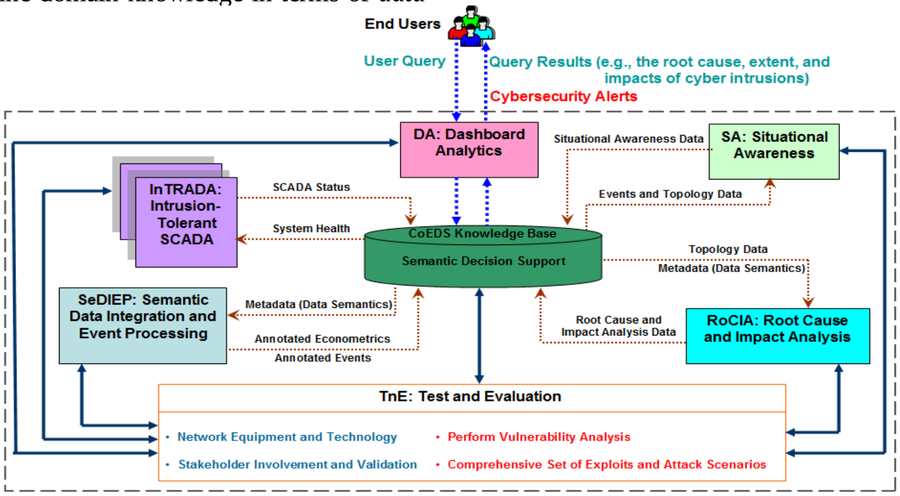

End Users
User Query
Query Results (e.g., the root cause, extent, and impacts of cyber intrusions)
Cybersecurity Alerts
DA: Dashboard Analytics
Situational Awareness Data
SA: Situational Awareness
InTRADA: Intrusion-Tolerant SCADA
SCADA Status
System Health
Events and Topology Data
CoEDS Knowledge Base
Semantic Decision Support
Topology Data
Metadata (Data Semantics)
SeDIEP: Semantic Data Integration and Event Processing
Metadata (Data Semantics)
Annotated Econometrics
Annotated Events
Root Cause and Impact Analysis Data
RoCIA: Root Cause and Impact Analysis
TnE: Test and Evaluation
• Network Equipment and Technology
• Stakeholder Involvement and Validation
• Perform Vulnerability Analysis
• Comprehensive Set of Exploits and Attack Scenarios

Fig. 1. Overall architecture of InTRECS system.

III. Methodology

A. InTRECS Overall Architecture

Figure 1 illustrates the overall architecture of InTRECS, which is decomposed into six subsystems.

- *Intrusion-Tolerant SCADA (InTRADA)*
  We will develop a survivable SCADA system based on intrusion-tolerant replication [16]. InTRADA will be capable of guaranteeing correct operations and excellent performance even when part of the system has been compromised and is under the control of an intelligent attacker.
- *Cybersecurity Ontologies and Knowledge Base for Energy Delivery Systems (CoEDS)*
  CoEDS knowledge base (KB) contains domain ontologies, a resource description framework (RDF) repository, a SPARQL RDF query engine, and an inference engine. The KB will provide end users with a unified and consistent data layer for analyzing data at the semantic level.
- *Semantic Data Integration and Processing (SeDIEP)*
  Our focus is to develop an automatic semantic data annotation & integration engine for tagging data sources based on the metadata defined in CoEDS ontologies. An event-processing engine will handle dynamic events and generate security alerts.
- *Root Cause and Impact Analysis (RoCIA)*
  RoCIA provides the basis to detect cyber incidents and investigate the root cause, attack methods, consequences, affected assets, impacted stakeholders, attackers' identity, and other metrics to inform an effective response. RoCIA will leverage the Cyber Security Econometrics System (CSES) and the inference and query engines provided within CoEDS KB to assist EDS stakeholders in evaluating cybersecurity investments and to provide an economic impact assessment of on-going cyber intrusions.
- *Dashboard Analytics and Situation Awareness (DaSA)*
  Dashboard analytics includes a user graphical user interface (GUI) to support interactions between end users and InTRECS. Situational awareness will be performed for end users. We will also support reasoning through the inference engine in CoEDS.
- *Test and Evaluation (TnE)*
  Implemented modules will automatically configure the test suite environment to the appropriate start state for the test case. A portal will provide the information and documentation and will execute the test case. We will also develop a test suite in an end-user setting, including a set of denial of service (DOS), reconnaissance, and network packet integrity exploits targeting SCADA, remote terminal unit (RTU), and network architecture vulnerabilities.

InTRECS will be constantly active to intrinsically provide resiliency, i.e., correct operations and excellent performance. At the same time, a DaSA GUI will guide end users to generate queries out of data derived from diverse sources. Query results, e.g., the root cause, extent, and impacts of the cyber intrusion, can then be provided back to end users. InTRECS will also push security alerts up to end users. Both query results and alerts are regarded as semantic decision support to end users because they extensively utilize Semantic Web technologies, namely, domain ontologies, RDF triples resulting from semantic annotation, and inferences & analysis performed at the semantic level.

B. CoEDS Domain Ontologies and Knowledge Base

There are four components in CoEDS KB: (i) CoEDS domain ontologies, (ii) an RDF repository, (iii) a SPARQL RDF query engine, and (iv) an inference engine. Through automatic data integration and logic reasoning, CoEDS KB will be able to provide a unified and consistent data layer for analyzing data *at the semantic level*. It will thus assist end users to effectively obtain real-time decision support, so that they can (i) obtain health status updates of SCADA replicas, (ii) analyze and better understand the root cause, extent, and impacts of an attack, (iii) acquire situational awareness, and (iv) recommend courses of action.

1) *Interaction between CoEDS and other InTRECS subsystems:* CoEDS KB actively exchanges information with other subsystems of InTRECS on a regular basis.

- InTRADA receives system health and status information from CoEDS KB, and incorporates such knowledge to enhance its fault-detection algorithms. This will enable InTRADA to more rapidly reconfigure itself in the event of a cyber attack by helping it distinguish between performance faults caused by a malicious application and by more benign issues such as transitory network problems. InTRADA sends to CoEDS KB status updates regarding the health of the replicas, hence providing data for future cyber attack analysis.
- SeDIEP obtains the data semantics, i.e., ontological metadata, from CoEDS KB and utilizes such metadata during the automatic semantic annotation. Annotated data, including cybersecurity econometrics, dynamic events, etc., are stored back into CoEDS KB to construct and continuously update the central data repository in the KB.
- CoEDS KB provides RoCIA with topology data as well as the data semantics essential for performing root cause and impact analysis. RoCIA supplies CoEDS KB with root cause and impact analysis data, including attack signatures, attack locations, exploits, consequences, countermeasures, model parameters, network components, security requirements, threats, vulnerabilities, and stakeholders.
- CoEDS KB furnishes DaSA with dynamic events and electric grid components and topology data, both of which are in an annotated form. DaSA sends back situational awareness data to CoEDS KB. In addition, the KB also provides the Correlation Layers for Information Query and Exploration (CLIQUE) and Traffic Circle, two visual analytics tools in DaSA, with interoperability for behavior model-based anomaly detection.

2) *Motivation for developing CoEDS ontologies:* Among existing ontologies in cybersecurity and related areas (mentioned in Section II), there is *not a single one that is comprehensive enough* to cover a complete set of concepts and relationships for the purpose of this research. In particular, with regard to the fields of SCADA status, root cause analysis, situational awareness, electric grid components and topology, cybersecurity econometrics, cost benefit analysis, and complex event processing, all aforementioned existing ontologies are missing some necessary concepts within these critical fields. Even in the case that a specific concept of our interest is contained in some existing ontology, more often than not, the semantics defined in such an ontology need to be extended and customized before this concept can be utilized within InTRECS system. In brief, Energy Control Systems (ECS) end users lack a comprehensive, customized conceptual model, which prevents the energy sector from leveraging enhanced knowledge acquisition processes brought by Semantic Web technologies. Such a situation motivates us to develop CoEDS domain ontologies.

3) *Ontology development principles:* We have observed seven practices suggested by Smith et al. [17]: the ontology should (i) be freely available; (ii) be expressed using a standard language or syntax; (iii) provide tracking and documentation for successive versions; (iv) be orthogonal to existing ontologies; (v) include natural language specifications of all concepts; (vi) be developed collaboratively; and (vii) be used by multiple researchers. In particular, we propose a *decomposition* methodology as the strategy for coming up with orthogonal ontologies. Our methodology is similar to those used in the database normalization theory, third normal form (3NF) for example. We first began with concepts from possibly many sub-domains in one large set, followed by the identification of dependencies or overlaps among these concepts, and we finally proceeded to decompose all concepts based on their identified dependencies. Our preliminary design is to develop seven sub-ontologies in CoEDS: SCADA status, root cause & impact, situational awareness, grid component & topology, cybersecurity econometrics, cost benefit, and complex event processing. Consequently, we achieved the orthogonality feature, i.e., the non-overlapping feature, for CoEDS domain ontologies.

4) *Knowledge-driven ontology development procedure:* The ontology development was not from scratch. Instead, to (i) take advantage of the knowledge already contained in existing ontologies and (ii) reduce the possibility of redundant efforts, we have reused, extended, and customized a set of well-established concepts from existing domain ontologies. In addition, popular upper ontologies, e.g., the Basic Formal Ontology (BFO), was imported into our ontologies. The ontology development was driven by domain knowledge and decomposed into five stages, as suggested by Uschold and Gruninger [18]: (i) specification of content; (ii) informal documentation of concept definitions (by domain experts); (iii) logic-based formalization of concepts and relationships between concepts; (iv) implementation of the ontology in a computer language; and (v) evaluation of the ontology, including the internal consistency and the ability to answer logical queries. As illustrated in Figure 2, these five stages are essentially ongoing and iterative because end users' needs will change as their understanding of the domain evolves. In this iterative, knowledge-driven approach, both ontology engineers and domain experts have been involved, working together to capture domain knowledge, develop a conceptualization, and implement the conceptual model. The ontology construction process has taken place over a number of iterations, involving a series of interviews, evaluation strategies, and refinements. Standard revision-control procedures have been utilized.

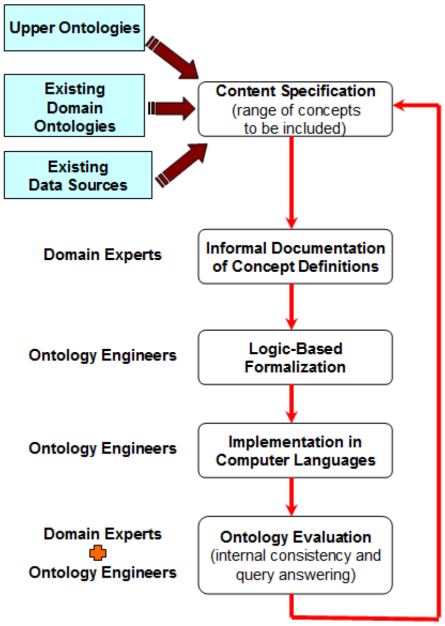

Upper Ontologies
Existing Domain Ontologies
Existing Data Sources
Content Specification (range of concepts to be included)
Domain Experts
Informal Documentation of Concept Definitions
Ontology Engineers
Logic-Based Formalization
Ontology Engineers
Implementation in Computer Languages
Domain Experts
Ontology Engineers
Ontology Evaluation (internal consistency and query answering)

Fig. 2. Knowledge-driven, iterative ontology development.

5) *Ontology format and development tool:* There are different formats and languages for describing ontologies, all of which are popular and based on different logics: Web Ontology Language (OWL) [19], Open Biological and Biomedical Ontologies (OBO) [20], Knowledge Interchange Format (KIF) [21], and Open Knowledge Base Connectivity (OKBC) [22]. We have chosen the OWL format recommended by the World Wide Web Consortium (W3C). OWL is designed for use by applications that need to process the content of information instead of just presenting information to humans. As a result, OWL facilitates greater machine interpretability of Web contents. We have chosen Protégé, an open-source ontology editor developed by

Stanford [23], as our development tool over other available tools such as CmapTools and OntoEdit.

6) CoEDS KB components – RDF Repository, Query Engine, and Inference Engine: Based on the formal knowledge defined in CoEDS ontologies, heterogeneous data sources can be annotated and integrated into a central repository. Note that data sources to be integrated include structured, semi-structured, or unstructured data, the interoperability thus becomes an obstacle during knowledge discovery. We adopt RDF, a model for data interchange recommended by the W3C, to handle such a challenge. RDF specifically supports the evolution of schemas over time without requiring all the data consumers to be changed. The generic structure of RDF allows structured, semi-structured, and unstructured data to be mixed, exposed, and shared across different applications, thus helping to handle the data interoperability challenge. Following automatic semantic data annotation (see Section III.C), RDF triples will be indexed and accumulated into a central repository. SPARQL Protocol and RDF Query Language (SPARQL) [24] is a query language recommended by W3C to retrieve and manipulate RDF data. End users of InTRECS system will be guided by a GUI to automatically generate RDF queries across semantically integrated sources. These queries will then be executed by a SPARQL-based query engine.

The RDF data repository and query answering are not enough for an effective and comprehensive knowledge acquisition. Suppose that some facts do not exist in any original data sources, they will thus not be stored in the RDF repository. But such information may be critical to end users. To obtain the ability to acquire previously implicit knowledge, we will incorporate an inference engine (a.k.a. logic reasoner). Compared with traditional relational database techniques, inference engines provide a more expressive method for querying and reasoning over available data sets. Thus, ontology-based (a.k.a. semantics-based) queries, instead of traditional SQL queries, are possible. Ontology-based queries improve traditional keyword-based queries in several ways. (i) Both *synonymous* terms (those having same meaning) and *polysemous* terms (those having different meanings) can be included to obtain more results that are relevant to the user query. (ii) Semantic relationships among terms often reveal extra clues hidden in disparate data sources. Such relationships can be explicitly discovered to further improve the quality of query answering. Consequently, we will be able to acquire hidden knowledge and information that was originally implicit and unclear, yet critical, to end users. With a logic reasoner, CoEDS repository will work as a comprehensive knowledge base.

7) *Sesame framework for RDF repository, SPARQL RDF query engine, and inference engine:* We have preliminarily chosen Sesame framework [25] to store and manage the RDF repository. Sesame is an open-source Java framework for the storage and querying of RDF data. The framework is fully extensible and configurable with respect to storage mechanisms, inferencers, RDF file formats, query result formats, and query languages. In addition, Sesame offers a JBDC-like user API, streamlined system APIs, and a RESTful HTTP interface supporting the SPARQL protocol for RDF. Moreover, Sesame contains a built-in inference engine, and various reasoning tasks, e.g., subsumption and contradiction reasoning, can be performed.

C. Semantic Data Annotation and Event Processing

According to the formal domain knowledge, including a global metadata model, defined in CoEDS, heterogeneous data sources can be annotated and seamlessly integrated into a central RDF data repository, which will serve as a unified and consistent data layer for data analytics applications.

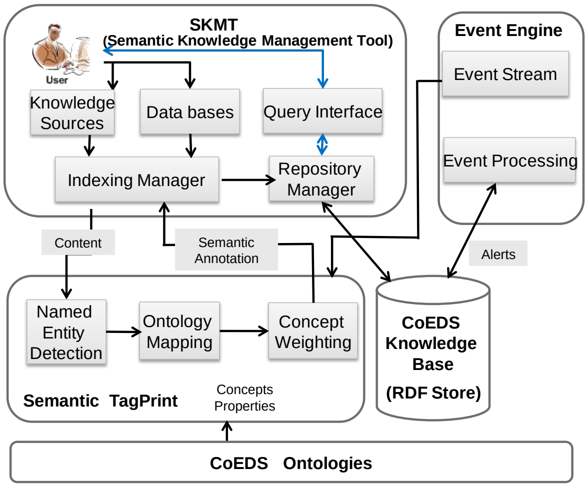


Fig. 3. Semantic data annotation and event processing (SeDIEP).

1) System overview: Semantic data annotation and event processing (SeDIEP) subsystem manages various data sources and automatically annotates and integrates data at semantic level. As shown in Figure 3, there are three major components in the subsystem: (i) Semantic TagPrint, (ii) Semantic Knowledge Management Tool (SKMT), and (iii) Event Engine. Semantic TagPrint is an automatic semantic tagging engine that annotates structured data and free text using ontological entities from CoEDS ontologies. SKMT manages heterogeneous data sources for semantic annotation and integration. Event engine feeds the semantic tagging engine with dynamic events. It also generates alerts with the support from CoEDS through modified RDF queries and the semantic reasoning.

Heterogeneous data sources will be annotated and seamlessly integrated into a central RDF data repository based on CoEDS ontologies. This data repository will serve as a unified and consistent data layer for further analyzing data at the semantic level. Our core technologies can substantially reduce design-to-execution time for application domains of data integration, visualization, and analysis.

- Meaningful data. Our system will annotate terms in text with their corresponding concepts in CoEDS ontologies by finding their meanings and analyzing their context.
- Scalability. Indexed data are stored and managed in a repository. Collected and initially processed data can be incrementally analyzed and indexed.
- Easy integration. Various data sources can be seamlessly integrated along with their semantic indexes.

2) *Deep annotation and integration:* Data sources to be integrated contain structured, semi-structured, or unstructured data. As discussed in the previous section, we adopt RDF to handle the data interoperability challenge. Semantic data annotation is the process of tagging source files with metadata predefined in ontologies such as names, entities, attributes, definitions, and descriptions. Herein, we use terms of "semantic annotation" and "semantic tagging" interchangeably. The annotation provides extra information contained in metadata to existing pieces of data. Metadata are usually from a set of ontological entities (including concepts and instances of concepts) predefined in ontologies. For unstructured data such as free text, we will use a tagging engine to align them with ontological entities and generate semantic annotations. For structured data including database data, the annotation will take two successive steps: (i) first we will annotate data source schemas by aligning their metadata with ontological entities; (ii) according to annotated schemas we will then transform original data instances into RDF triples. We refer to such annotation as "deep" annotation – this term was coined by Goble, C. in the Semantic Web Workshop of WWW 02. It is necessary to annotate more than just data source schemas because there are situations where the opposite "shallow" annotation (i.e., annotation on schemas alone) cannot provide users with the desired knowledge. Following semantic data annotation, RDF triples will be indexed and accumulated into a central repository.

3) *Unified view over original data sources and cost-efficient analysis:* All semantic tags will be generated from a global metadata model, i.e., CoEDS ontologies, our tool thus provides a unified view over original data sources at the semantic level. As discussed before, our RDF query and reasoning engines will provide users with more meaningful and relevant information from semantically annotated and integrated data sources. In addition, semantic relationships among tags provide us with additional clues and will further improve the quality of retrieved results. Given a set of candidate results to be returned to users, we will calculate the semantic similarity between each result and the user query using semantic features such as (i) *hypernym*, which defines the *superClassOf* relationship and (ii) *holonym*, which defines the *partOf* relationship. We will then rank these results by their respective semantic similarities. Consequently, users can be presented with more relevant query results.

4) *Semantic event processing:* Dynamic events will be fed to our Semantic Tag Print, which will annotate these events with semantic tags. Then events are represented as RDF triples, accompanied with event attributes such as timestamps and probabilities. With the support from CoEDS, SeDIEP will transform these tagged events into SPARQL queries. We will perform event filtering, correlation, and aggregation or abstraction using semantic matching, rules, and similarity evaluations. Moreover, we will detect event patterns on event streams with temporal semantic rules. As a result, high-risk vulnerabilities and threats can be predicted, and security alerts will then be automatically generated and rendered to users when facing potential cyber intrusions.

5) *Core Components in SeDIEP:* Figure 3 shows three major components in SeDIEP to semantically integrate various data sources and event streams.

a) Component one: Semantic TagPrint is an automatic semantic tagging engine that annotates structured data and free text using ontological entities. Three modules were designed for this component.

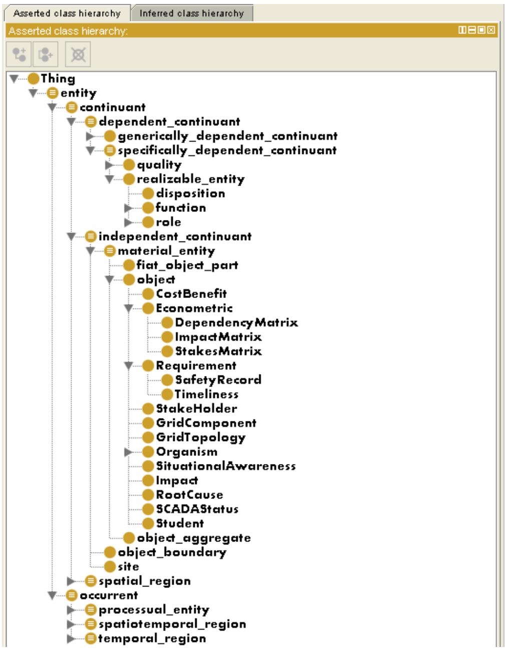
- Named Entity Detection: This module extracts named entities, noun phrases in general, from the input text. We adopt Stanford Parser [26] to detect and tokenize sentences, and assign Part-of-Speech (PoS) tags to tokens. Entity names will be extracted based on PoS tags.
- Ontology Mapping: This module maps extracted entity names to CoEDS concepts and instances with two steps: Phrase mapping and Sense mapping. Phrase mapping will match the noun phrase of an entity name to a predefined concept or instance. Sense mapping will utilize a linear-time lexical chain algorithm to disambiguate terms that have several senses defined in ontologies.
- Ontology Weighting: This module utilizes statistical and ontological features of concepts to weigh semantic tags. We then annotate the input text using the semantics with higher weights.

b) Component two: SKMT collects original text and sends annotation results to Repository Manager, whose main role is to manage RDF repository (store) and to communicate with Query Interface. These components altogether provide a unified view over original data sources at the semantic level. Users will be guided by a GUI to automatically generate RDF queries across semantically integrated data sources. These queries will then be executed by a SPARQL-based RDF query engine. As discussed earlier in this subsection, we can calculate the semantic similarity between each candidate query result and the user query using semantic features such as *hypernym* and *holonym*. These query results can then be ranked by their respective semantic similarities. Consequently, we are able to render users more accurate and desired query results.

c) Component three: Event Engine annotates dynamic events and stores them as RDF triples. It will then generate SPARQL queries and perform event filtering, correlation, and aggregation or abstraction with the semantics defined in CoEDS ontologies.

IV. Preliminary Experimental results

In this ongoing research, we have developed a preliminary version of CoEDS domain ontologies and knowledge base to demonstrate a proof of concept of how Semantic Web technologies can significantly contribute to resilient energy control systems. We also exported instances into an RDF data repository within the Sesame framework.



```
Thing
  entity
    continuant
      dependent_continuant
        generically_dependent_continuant
        specifically_dependent_continuant
          quality
          realizable_entity
            disposition
            function
            role
      independent_continuant
        material_entity
          fiat_object_part
          object
            CostBenefit
            Econometric
              DependencyMatrix
              ImpactMatrix
              StakesMatrix
            Requirement
              SafetyRecord
              Timeliness
            StakeHolder
            GridComponent
            GridTopology
            Organism
            SituationalAwareness
            Impact
            RootCause
            SCADAStatus
            Student
          object_aggregate
        object_boundary
        site
      spatial_region
    occurrent
      processual_entity
      spatiotemporal_region
      temporal_region
```

Fig. 4. Protégé GUI screen shot exhibiting some CoEDS concepts.

A. CoEDS Ontologies

As discussed earlier in Section III.B, we have developed seven sub-ontologies in CoEDS: *SCADA Status Ontology*, *Root Cause & Impact Ontology*, *Situational Awareness Ontology*, *Grid Component & Topology Ontology*, *Cybersecurity Econometrics Ontology*, *Cost Benefit Ontology*, and *Complex Event Processing Ontology*. The purpose of such a decomposition strategy is to achieve the orthogonality feature, i.e., the non-overlapping feature among different CoEDS sub-ontologies. After individual sub-ontologies were developed, we then imported them into CoEDS. If future modifications are needed for any sub-ontology, the changed schema information will be automatically integrated into CoEDS ontologies. Figure 4 demonstrates a screen shot from Protégé GUI, which exhibits a portion of CoEDS concepts. Note that the well-defined, general-purpose structure from the Basic Formal Ontology (BFO), a popular upper ontology across different disciplines and research areas, was preserved in the ontology schema. Statistic information for all seven sub-ontologies is summarized in Table I. In total, CoEDS ontologies contain 269 concepts, 232 object properties, and 110 data properties.

TABLE I. Statistics for CoEDS Ontologies

Sub-Ontology	**Statistic Information**		
	Total Number of Concepts	***Total Number of Object Properties***	***Total Number of Data Properties***
SCADA Status Ontology	35	23	12
Root Cause & Impact Ontology	37	21	9
Situational Awareness Ontology	39	27	15
Grid Component & Topology Ontology	51	39	17
Cybersecurity Econometrics Ontology	38	25	20
Cost Benefit Ontology	33	19	18
Complex Event Processing Ontology	36	28	19

B. CoEDS Knowledge Base

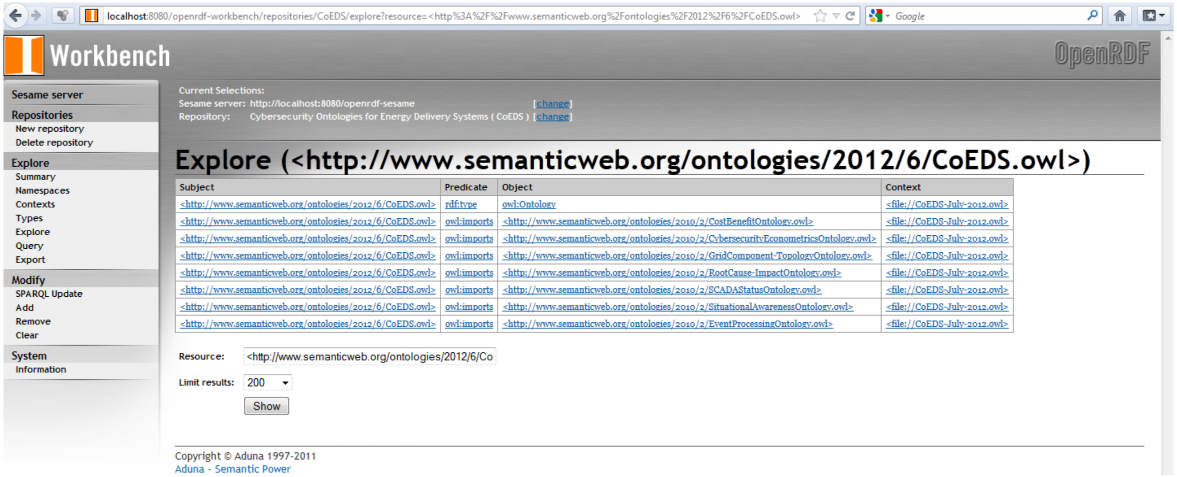
The current CoEDS KB contains a total of 1,223 facts (a.k.a. axioms in Protégé). Details can be found in Table II.

TABLE II. Statistics for CoEDS Knowledge Base Axioms

Axiom Category	**Statistic Information**
Class Axioms	460
Subclass Axioms	268
Equivalent Class Axioms	57
Disjoint Class Axioms	135
Object Property Axioms	217
Data Property Axioms	108
Individual Axioms	236
Annotation Axioms	202

C. Sesame Framework to Manage Data Repository

Within the Sesame framework we exported all ontological instances into an RDF data repository for future storage and management. Figure 5 is a screen shot from Sesame GUI, where the seven sub-ontologies and the overall CoEDS ontologies were clearly demonstrated. Being an open-source Java framework, Sesame framework can be readily extended and configured for the storage and querying of RDF data. Moreover, a JBDC-like user API, streamlined system APIs, and a RESTful HTTP interface are offered in Sesame as well.



Current Selections:
Sesame server: http://localhost:8080/openrdf-sesame [change]
Repository: Cybersecurity Ontologies for Energy Delivery Systems (CoEDS) [change]

Explore (<http://www.semanticweb.org/ontologies/2012/6/CoEDS.owl>)

Subject	Predicate	Object	Context
<http://www.semanticweb.org/ontologies/2012/6/CoEDS.owl>	rdf:type	owl:Ontology	<file://CoEDS-July-2012.owl>
<http://www.semanticweb.org/ontologies/2012/6/CoEDS.owl>	owl:imports	<http://www.semanticweb.org/ontologies/2010/2/CostBenefitOntology.owl>	<file://CoEDS-July-2012.owl>
<http://www.semanticweb.org/ontologies/2012/6/CoEDS.owl>	owl:imports	<http://www.semanticweb.org/ontologies/2010/2/CybersecurityEconometricsOntology.owl>	<file://CoEDS-July-2012.owl>
<http://www.semanticweb.org/ontologies/2012/6/CoEDS.owl>	owl:imports	<http://www.semanticweb.org/ontologies/2010/2/GridComponent-TopologyOntology.owl>	<file://CoEDS-July-2012.owl>
<http://www.semanticweb.org/ontologies/2012/6/CoEDS.owl>	owl:imports	<http://www.semanticweb.org/ontologies/2010/2/RootCause-ImpactOntology.owl>	<file://CoEDS-July-2012.owl>
<http://www.semanticweb.org/ontologies/2012/6/CoEDS.owl>	owl:imports	<http://www.semanticweb.org/ontologies/2010/2/SCADAStatusOntology.owl>	<file://CoEDS-July-2012.owl>
<http://www.semanticweb.org/ontologies/2012/6/CoEDS.owl>	owl:imports	<http://www.semanticweb.org/ontologies/2010/2/SituationalAwarenessOntology.owl>	<file://CoEDS-July-2012.owl>
<http://www.semanticweb.org/ontologies/2012/6/CoEDS.owl>	owl:imports	<http://www.semanticweb.org/ontologies/2010/2/EventProcessingOntology.owl>	<file://CoEDS-July-2012.owl>

Resource: <http://www.semanticweb.org/ontologies/2012/6/Co

Limit results: 200

Copyright © Aduna 1997-2011
Aduna - Semantic Power

Fig. 5. Screen shot from Sesame repository management.

V. Conclusion

To preserve critical energy control functions while under attack, it is necessary to perform comprehensive analysis on the root cause, extent, and impacts of cyber intrusions without sacrificing the availability of energy delivery. We proposed to develop InTRECS, an intrinsically resilient energy control system, to address these challenges. Semantic Web technologies, which play critical roles in knowledge representation and acquisition, have been extensively adopted in our system. The focus of this ongoing research is to demonstrate a proof of concept of how Semantic Web technologies can significantly contribute to resilient energy control systems. We justified the research motivation, described our methodology in detail, and exhibited preliminary experimental results. Future research directions include, but are not limited to, (i) continue CoEDS ontology development towards delivering a highly stable and more usable version; (ii) incorporate query and inference engines into the knowledge base for end users to better analyze root causes and impacts of cyber intrusions; and (iii) implement SeDIEP subsystem.

Acknowledgment

This research was partially supported through the U.S. Department of Energy (DOE) Higher Education Research Experiences (HERE) program for Faculty at the Oak Ridge National Laboratory, Oak Ridge, Tennessee, sponsored by the U.S. Department of Homeland Security (DHS).

References

[1] J. Undercoffer, A. Joshi, and J. Pinkston, “Modeling Computer Attacks: An Ontology for Intrusion Detection,” *RAID 2003, LNCS 2820*, pp. 113-135, 2003, Springer-Verlag Berlin Heidleberg, 2003.

[2] A. Simmonds, P. Sandilands, and L. Ekert, “An Ontology for Network Security Attacks,” *Proc. the 2nd Asian Applied Computing Conference (AACC-04)*, LNCS 3285, pp. 317-323, 2004.

[3] W. Wang and T. Daniels, “A Graph Based Approach toward Network Forensic Analysis,” *ACM Transactions on Information and Systems Security*, Vol. 12, No. 1, Article 4, Pub. Date: Oct. 2008.

[4] J. Hieb, J. Graham, and J. Guan, “An Ontology for Identifying Cyber Intrusion Induced Faults in Process Control Systems,” *Critical Infrastructure Protection III*, IFIP AICT 311, pp. 125-138, 2009.

[5] G. Isaza, A. Castillo, M. Lopez, L. Casillo, and M. Lopez, “Intrusion Correlation Using Ontologies and Multi-agent Systems,” *Proc. 4th International Conference on Information Security and Assurance (ISA-10)*, pp. 355-361, Miyazaki, Japan, June 23-25, 2010.

[6] M. Choras, R. Kozik, A. Flizikowski, and W. Holubowicz, “Ontology Applied in Decision Support System for Critical Infrastructures Protection,” *IEA/AIE2010*, LNAI, pp. 671-680, 2010.

[7] B. Barnett, A. Crapo, and P. O’Neil, “Experiences in Using Semantic Reasoners to Evaluate Security of Cyber Physical Systems,” *General Electric Internal Report GridSec*, 2012.

[8] C. Turnitsa and A. Tolk, “Knowledge Representation and the Dimensions of a Multi-Model Relationship,” *Proc. the 40th Conference on Winter Simulation (WSC-08)*, pp. 1148–56, 2008.

[9] L. Reeve and H. Han, “Semantic Annotation for Semantic Social Networks Using Community Resources,” *AIS SIGSEMIS Bulletin*, vol. 2, pp. 52-56, 2005.

[10] S. Wiesener, W. Kowarschick, and R. Bayer, “SemaLink: An Approach for Semantic Browsing through Large Distributed Document Spaces,” *Proc. the 3rd International Forum on Research and Technology Advances in Digital Libraries*, p. 86, 1996.

[11] Zemanta. http://www.zemanta.com/.

[12] Common Tag. http://www.commontag.org/.

[13] K. Murat, J. Dang, and S. Uskudarli, “UNIpedia: A Unified Ontological Knowledge Platform for Semantic Content Tagging and Search,” *Proc. the 4th IEEE International Conference on Semantic Computing*, Pittsburg, PA, USA, 2010.

[14] K. Murat, J. Dang, and S. Uskudarli, “Semantic TagPrint: Indexing Content at Semantic Level,” *Proc. the 4th IEEE International Conference on Semantic Computing*, Pittsburg, PA, USA, 2010.

[15] Y. Simmhan, Q. Zhou, and V.K. Prasanna,“Semantic Information Integration for Smart Grid Applications,” *Chapter 19, Green IT: Technologies and Applications*, pp. 361–80, 2011.

[16] J. Kirsch, S. Goose, Y. Amir, and P. Skare, “Toward Survivable SCADA,” *Proc. the Annual Cyber Security and Information Intelligence Research Workshop (CSIIRW-11)*, Oak Ridge, 2011.

[17] B. Smith, M. Ashburner, C. Rosse, J. Bard, W. Bug, W. Ceusters, L. Goldberg, K. Eilbeck, A. Ireland, C. Mungall, N. Leontis, P. Rocca-Serra, A. Ruttenberg, S. Sansone, R. Scheuermann, N. Shah, P. Whetzel, and S. Lewis, “The OBO foundry: coordinated evolution of Ontologies to support biomedical data integration,” *Nature Biotechnology*, 25(11):1251–1255, 2007.

[18] M. Uschold and M. Gruninger, “Ontologies: principles, methods, and applications,” *Knowledge Engineering Review*, 11(2):93-155, 1996.

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

[20] OBO. http://www.obofoundry.org/.

[21] KIF (Knowledge Interchange Format). http://logic.stanford.edu/kif/.

[22] OKBC. http://www.ai.sri.com/okbc/.

[23] Protégé. http://protege.stanford.edu/.

[24] SPARQL. http://www.w3.org/TR/rdf-sparql-query/.

[25] Sesame. http://www.openrdf.org/doc/sesame/.

[26] D. Klein and C.D. Manning, “Accurate Unlexicalized Parsing,” *Proc. the 41st Meeting of the Association for Computational Linguistics*, pp. 423-430, 2003.

Exploiting inference to improve temporal RDF annotations and queries for machine reading

Robert C. Schrag
Digital Sandbox, Inc.
McLean, VA USA
bschrag@dsbox.com

***Abstract*—We describe existing and anticipated future benefits of an end-to-end methodology for annotating formal RDF statements representing temporal knowledge to be extracted from text, as well as for authoring and validating test and/or application queries to exercise that knowledge. Extraction is driven by a target ontology of temporal and domain concepts supporting an intelligence analyst's timeline tool. Both the tool and the methodology are supported at several points by an implemented temporal reasoning engine, in a way that we argue ultimately advances machine reading technology by increasing both sophistication and quality expectations about temporal annotations and extraction.**

***Index Terms*—temporal knowledge representation and reasoning, extracting formal knowledge from text, machine reading, annotation interfaces and validation**

I. Introduction

Machine reading—that is, automatic extraction of formal knowledge from natural language text—has been a longstanding goal of artificial intelligence. Effective extraction into RDF has the potential to make targeted knowledge accessible in the semantic web. We recently supported a large-scale evaluation of temporal knowledge extraction from text by providing RDF/OWL ontology for target statements and a corresponding reasoning engine for query answering. Along the way, we discovered…

- How inference could improve annotation—the manual extraction of formal temporal statements—and question authoring for evaluation or for applications.
- How, coupled with annotation and question authoring processes, inference could ultimately drive more sophisticated machine reading capabilities.

II. Temporal knowledge representation and reasoning for timeline development

Our temporal logic is based loosely on the event calculus [10], as follows.

A time interval is a convex collection of time points—intuitively, an unbroken segment of a time line. Time intervals begin and end with time points, which may be constrained relative to each other or relative to a calendar. The ontology includes a rich set of relational properties over time points and intervals, and the reasoning engine will calculate tightest inferable bounds between any two points and will detect contradictory time point relation sets.

A fluent is an object-level, domain statement (e.g., FluentA: attendsSchool(Jansa LubljanaUniversity)) whose truth value is a function of time. It is taken to be true at time points where it holds and not to be true at time points where it does not hold. We reify a fluent in an observation—a meta-level statement whose object is a fluent, whose subject is a time interval, and whose predicate is a holds property (e.g., holdsThroughout(FluentA Interval1), when FluentA is observed over Interval1, corresponding to, say, September, 1980).

The events of interest to us, which we call transition events, occur at individual time points and may cause one or more fluents to change truth value. We represent events (like the birth of Jansa) as objects with attribute properties like agent, patient, and location, and we relate events to time intervals with an occurs property (e.g., occursAt(BirthOfJansa Point2), where Point2 is associated with an interval corresponding to the date September 17, 1958). As usual with the event calculus, such events can initiate fluents (e.g., occursAt(BirthOfJansa Point2) initiates FluentB: alive(Jansa Interval3), where Interval3 is begun by Point2) or terminate them (e.g., DeathOfJansa…). The temporal reasoning engine implements appropriate axioms to perform fluent initiation and termination.

Note that an observer may report information about the temporal extent of a fluent without communicating anything about initiation or termination. E.g., if text says *Abdul and Hasan lived next door to each other in Beirut in 1999,* we don't know when Abdul or Hasan may have moved to or from Beirut. When text says *Abdul moved to Beirut in 1995 and emigrated in 2007,* we use the properties clippedBackward and clippedForward regarding the fluent residesInGPE-spec(Abdul BeirutLebanon) to indicate initiation and termination by anonymous (unrepresented) transition events, so that we can also initiate or terminate temporally under-constrained like-fluent observations (e.g. *Abdul lived in Beirut during the 1990s*).

The reasoning engine's implementation, using AllegroGraph, Allegro Prolog, and Allegro Common Lisp from Franz, Inc., can answer any conjunctive query. While not yet heavily optimized, it is at least fast enough to support machine reading system evaluation over newspaper articles where cross-document entity co-reference is not required.

The combined extraction and reasoning capability was conceived to support an intelligence analyst's timeline tool in which a GUI would be populated with statements about entities (e.g., persons) of interest extracted from text. Our evaluation of machine reading capabilities was based on queries similar to those we would have expected from such a tool's API. It also supposed the analysts could formulate their own, arbitrary questions, such as Query 1: *Find all persons who were born in Ljubljana in the 1950s and attended Ljubljana University in the 1980s, the titles that they held, the organizations in which they held these titles, and the maximal known time periods over which they attended and held these titles.*

III. LESSONS LEARNED AND REALIZED IN IMPLEMENTATION

This indirect, query answering style of machine reading evaluation makes it especially important that we perform effective quality control of formal queries in the context of the formal statements we expect to be extracted from answer-bearing documents. We thus developed the test query validation approach illustrated in Figure 1. Considering Query 1's formalization (see Figure 10 in section IV.B), it's worth noting that we used the methodology illustrated here to debug a number of subtle errors occurring in our earlier (manual) formulations. When each such formulation did not result in the answers expected, we traced inference to identify a point of failure, corrected this, and then iterated until correct.

Our machine reading technologists told us early on that they preferred macro-level relational interfaces that would streamline away micro-level details of time points and intervals. We thus provide a language of flexible specification strings (spec strings) that expand to create time points, intervals, and relations inside our reasoning engine. We also provide ontology to associate the temporal aspects Completed, Ongoing, and Future with fluents (e.g., *he used to live there* vs. *he is living there* vs. *he is going to live there*) and with the reporting dates of given articles, to afford a relational interface reasonably close in form to natural language sources. For the Completed or Future aspects, we also can capture quantitative lag or lead information (e.g., *he lived there five years ago* or *he will move in five days from now*).

IV. LESSONS LEARNED WITH IMPLEMENTATION PROPOSED

Here, we propose some further approach elements that we expect to lead to high-quality temporal annotations, including…

A. Interfaces and workflows deliberately designed to support capture of all statements specified as extraction targets (see section A)

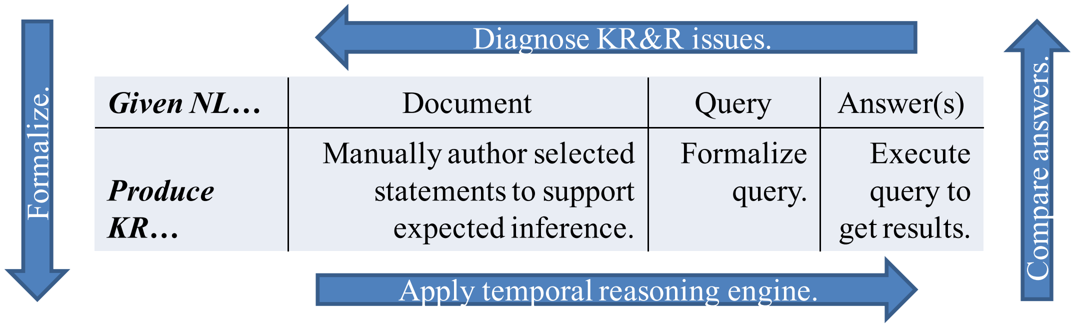
B. Graphical time map display including fluents and events (see section B)
   - On-line inference to elucidate inter-relationships and potential contradictions
   - Visual feedback to let users help assure quality themselves
   - Time map-based widgets supporting user knowledge entry

C. Technology adaptable to test or application question authoring (see section C)

D. Quantitative temporal relation annotation evaluation (see section V.A).

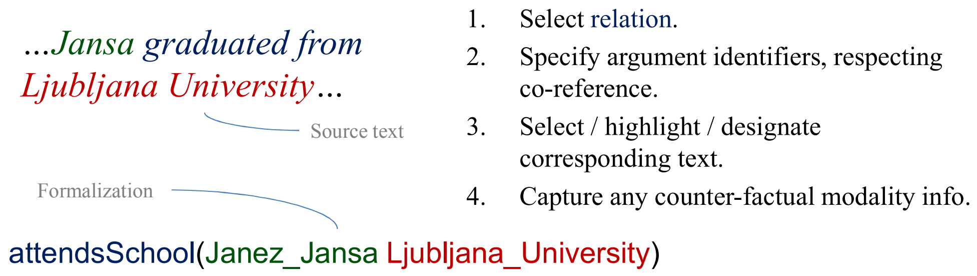
A. Annotation workflows

Fluents are simple statements that we can readily represent in RDF. The example in Figure 2 focuses on the fluent about one Janez Jansa attending Ljubljana University—only on the fluent, not the full observation including temporal information (i.e., only that Jansa attends the school, not when). The technology needed to annotate such information is well understood and (excepting perhaps the last bullet about modality) has been well enough exercised that we may routinely expect good results. This includes multi-frame GUIs where a user can produce stand-off annotations by highlighting text and by clicking in drop-down boxes select relations, classes, and instances. In part because these tools have preceded reading for formal knowledge extraction, they may not use our intended representation internally—i.e., they may for historical reasons internally use a representation (e.g., XML that is not RDF) tailored to linguistic phenomena rather than associated with any formal ontology.



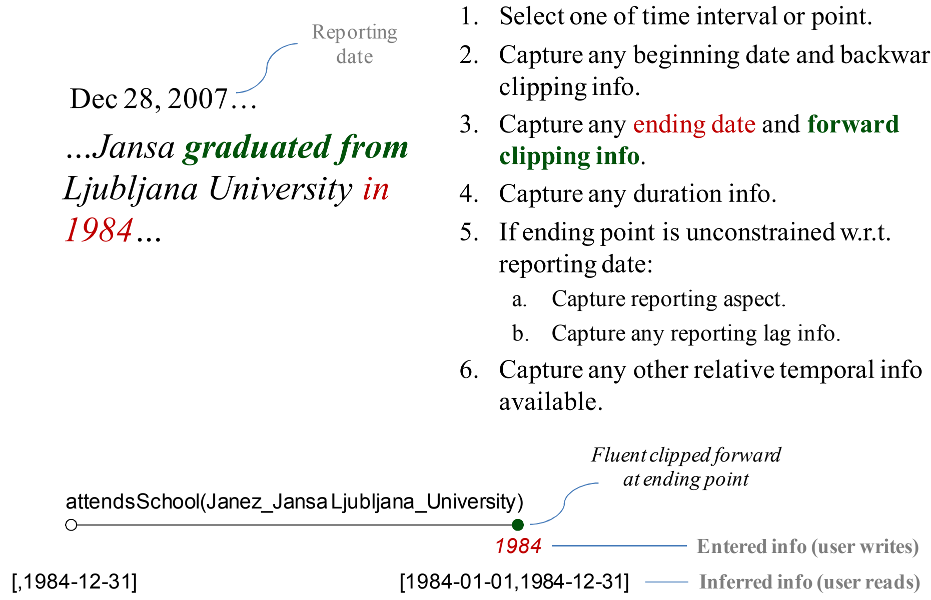
Given NL…	Document	Query	Answer(s)
Produce KR…	Manually author selected statements to support expected inference.	Formalize query.	Execute query to get results.

Figure 1. We validate test queries by making sure that natural language (NL) and formal knowledge representation (KR) versions of documents, queries, and answers agree, diagnosing and debugging as necessary.



1. Select relation.
2. Specify argument identifiers, respecting co-reference.
3. Select / highlight / designate corresponding text.
4. Capture any counter-factual modality info.

Figure 2. The workflow to annotate a fluent



1. Select one of time interval or point.
2. Capture any beginning date and backward clipping info.
3. Capture any ending date and **forward clipping info**.
4. Capture any duration info.
5. If ending point is unconstrained w.r.t. reporting date:
   a. Capture reporting aspect.
   b. Capture any reporting lag info.
6. Capture any other relative temporal info available.

Figure 3. The workflow to annotate a holds statement (a fluent observation)

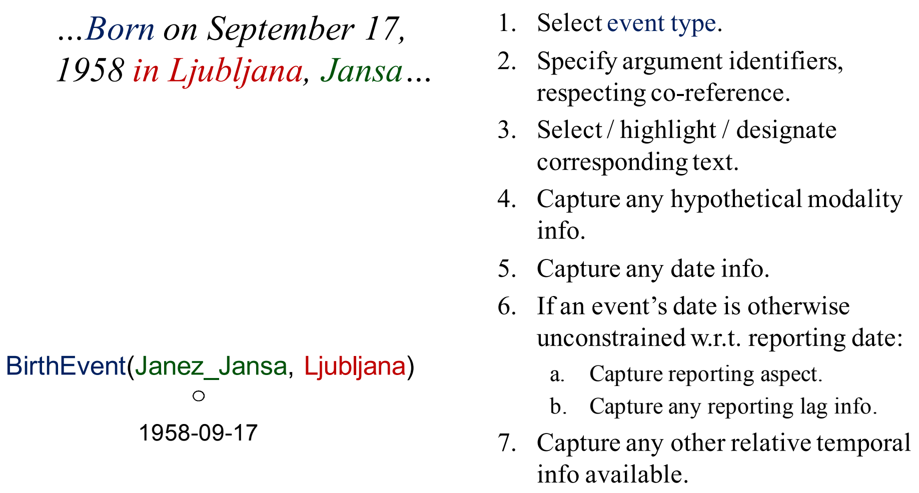
Capturing the temporal information associated with the given observation of a fluent in a holds statement requires following a sequence of actions and decisions in a deliberately designed workflow, as outlined in Figure 3. We have highlighted, by color- and typeface-coding, some temporal elements in the source text, along with corresponding steps in the workflow and elements of the associated graphical representation.

Addressing the workflow step by step, we see that:

- There is no reason to believe Jansa attended school for only one day (the time unit associated with a time "point" in our machine reading evaluation), so we choose a time interval (and the predicate holdsThroughout) rather than a time point (and holdsAt). Schrag [8] argues that holdsAt almost never is appropriate, and in future this step may be omitted.
- We find no beginning date information. (In the absence of such information, there is no benefit to asserting backward clipping.)
- We find (and have highlighted) a coarse-grained ending date (1984). We indicate that our fluent is clipped forward, assuming that Jansa no longer attends the school after graduation.
- There is no duration information. (We don't know how long Jansa was at school.)
- The ending point is well before the reporting date, so we skip to the next step.
- There is no other relevant temporal information.
- To indicate clipping, the graphic fills the time point symbol (making it solid).

Our reasoning engine expands the entered coarse date 1984 into earliest and latest possible calendar dates bounding the observation's ending point. It also infers an upper bound on its beginning time point.

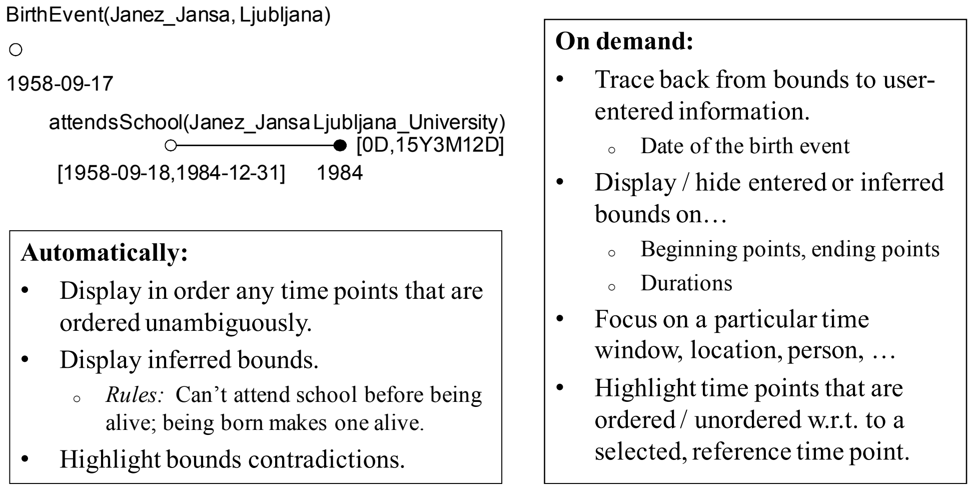
As illustrated in Figure 4, we invoke a similar workflow for event occurrence. Because our representation for events is simpler than that for observations, this workflow has fewer steps. Our ontology treats birth as a fluent transition event—it occurs at a given time point, and it causes a transition of the vital status of the person born (from FuturePerson to Alive). Our graphical representation here accordingly just depicts a single time point (not an interval). We can use basically the same workflow to capture a non-transition event (e.g., a legal trial) that occurs over more than one time point.



...Born on September 17, 1958 in Ljubljana, Jansa...

1. Select event type.
2. Specify argument identifiers, respecting co-reference.
3. Select / highlight / designate corresponding text.
4. Capture any hypothetical modality info.
5. Capture any date info.
6. If an event's date is otherwise unconstrained w.r.t. reporting date:
   a. Capture reporting aspect.
   b. Capture any reporting lag info.
7. Capture any other relative temporal info available.

Figure 4. Workflow to annotate a transition event


BirthEvent(Janez_Jansa, Ljubljana)
○
1958-09-17
attendsSchool(Janez_Jansa Ljubljana_University)
[0D,15Y3M12D]
[1958-09-18,1984-12-31]
1984

Automatically:
- Display in order any time points that are ordered unambiguously.
- Display inferred bounds.
  - *Rules:* Can't attend school before being alive; being born makes one alive.
- Highlight bounds contradictions.

On demand:
- Trace back from bounds to user-entered information.
  - Date of the birth event
- Display / hide entered or inferred bounds on…
  - Beginning points, ending points
  - Durations
- Focus on a particular time window, location, person, …
- Highlight time points that are ordered / unordered w.r.t. to a selected, reference time point.

Figure 5. A time map with both a fluent and a transition event

B. Displaying integrated time maps

Figure 5 illustrates a time map including both the birth event and the school attendance fluent from earlier figures. It also suggests functional requirements to be satisfied automatically/by default and upon user demand. Note that we now have automatically displayed—from on-line temporal inference—a lower bound on the fluent observation's beginning date: Jansa could not have attended school until after he was born. (The "day" time point granularity used in our machine reading evaluation leads to some non-intuitive effects, like not being alive until the day after one is born. We can easily correct this using an interval constraint propagation arithmetic including infinitesimals [6][7][8].) We've also indicated bounds on the fluent observation's duration (calculated as ending date bounds interval minus starting date bounds interval). Propagating effects like this can maximize visual feedback to users, expanding their basis for quality judgments about the information they enter. If any inferred bound seemed odd, a user could click on it to identify which of his/her own entered information (then highlighted in the display) might be responsible. The time map display tool would automatically launch such an interaction when it detected a contradiction among inputs.

The time map displayed in Figure 6 includes all the information from the source text that is necessary to answer Query1. The last fluent observation (at bottom right, where Jansa is prime minister) exercises workflow steps that earlier time map elements don't. We have no ending date for this observation, but we do have present tense reference to Jansa as *the prime minister,* so we appeal to the reporting aspect Ongoing. From the source text *he was elected prime minister on November 9, 2004,* we can bound the observation's beginning point.

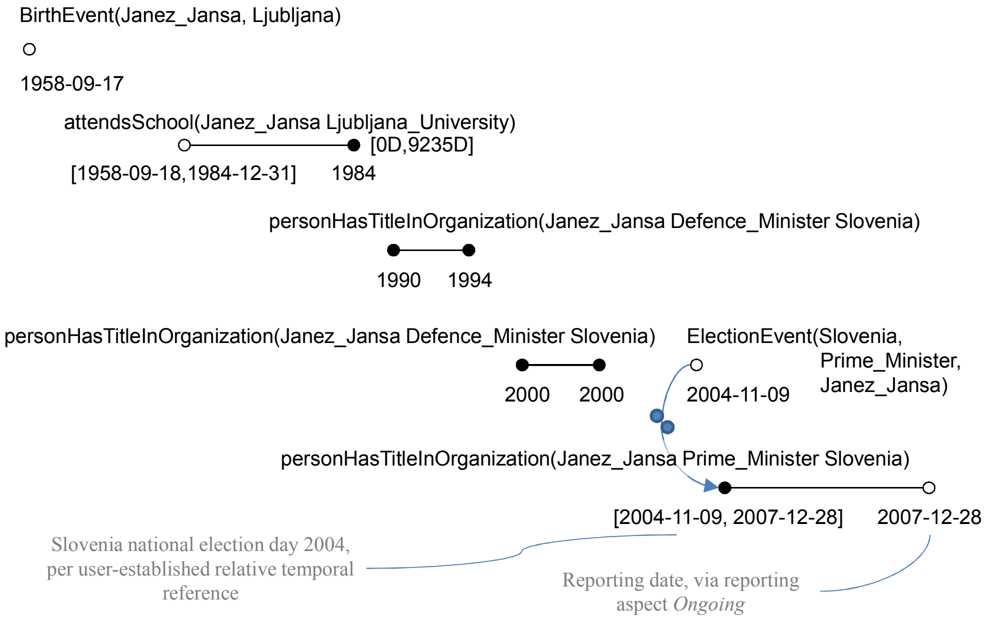

BirthEvent(Janez_Jansa, Ljubljana)
1958-09-17
attendsSchool(Janez_Jansa Ljubljana_University)
[0D,9235D]
[1958-09-18,1984-12-31]
1984
personHasTitleInOrganization(Janez_Jansa Defence_Minister Slovenia)
1990
1994
personHasTitleInOrganization(Janez_Jansa Defence_Minister Slovenia)
2000
2000
ElectionEvent(Slovenia, Prime_Minister, Janez_Jansa)
2004-11-09
personHasTitleInOrganization(Janez_Jansa Prime_Minister Slovenia)
[2004-11-09, 2007-12-28]
2007-12-28
Slovenia national election day 2004, per user-established relative temporal reference
Reporting date, via reporting aspect Ongoing

Figure 6. A time map with more statements extracted from the same article

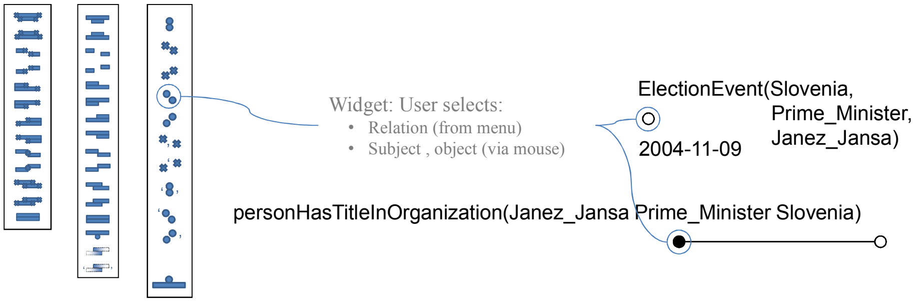

Widget: User selects:
Relation (from menu)
Subject , object (via mouse)
ElectionEvent(Slovenia, Prime_Minister, Janez_Jansa)
2004-11-09
personHasTitleInOrganization(Janez_Jansa Prime_Minister Slovenia)

Figure 7. Using the GUI to establish relative temporal reference

Our user also has entered the election event. An election is not necessarily a fluent transition event, at least in that an elected candidate does not always take office immediately. So, we rely on the user to establish relative temporal reference between the election event and the fluent observation's beginning. See the depicted constraint, whose entry is illustrated in Figure 7. Establishing relative temporal reference requires the selection of a pair of time points and/or intervals to be related and of an appropriate temporal relation between them. Here, we just need the time point at which the election occurs to be less than or equal to the time point at which Jansa takes office.

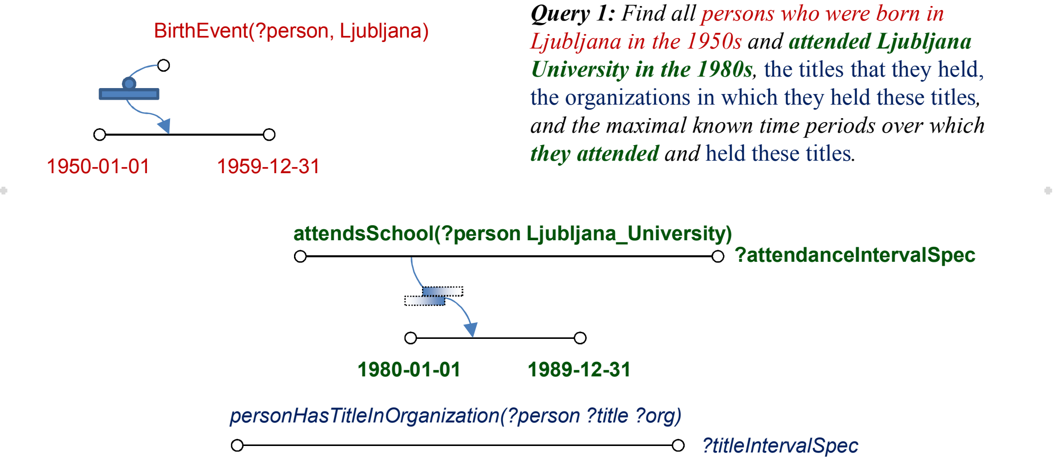
While a few common relations may be all that most users will ever need, we do have a lot of relations [8] that a user could in principle choose from. We should be able to provide access to these effectively, so that our user is empowered without being overwhelmed.

Figure 8 shows formal statements that would be created directly by the user's actions (i.e., not also including those created indirectly by inference) in entering the information reflected in our finished time map. We have highlighted fluents and some other key statements, each of which appears near several related statements. Our time map represents Jansa's birth event in a non-standard way, repeated here in different color type, beside the italicized, standard statements. We have not similarly formalized the event of Jansa's election as PM, and Figure 8 includes just statements about that event's point of occurrence.

Clearly, we can do a lot of formal work for the user behind the scenes.

```
      F_school: attendsSchool (Janez_Jansa Ljubljana_University)
                holdsThroughout(F_school I_school)
                clippedForward(F_school I_school)
                hasTimeIntervalSpecString(I_school [,1984])
                hasPersonBorn(birth Janez_Jansa)
                occursAt(birth P_birth)
                hasTimePointSpecString(P_birth 1958-09-17)          BirthEvent(Janez_Jansa, Ljubljana)
                hasPersonBorn(birth Janez_Jansa)
                hasBirthEventGPE-spec(birth GPEspec)
                hasCityTownOrVillage(GPEspec ljubljana_Ljubljana_Slovenia)
                hasNationState(GPEspec Slovenia)
                type(Defence_Minister MinisterTitle)
   F_PTIO_DM_1: personHasTitleInOrganization(Janez_Jansa Defence_Minister Slovenia)
                holdsThroughout(F_PTIO_DM_1 I_PTIO_DM_1)
                clippedBackward(F_PTIO_DM_1 I_PTIO_DM_1)
                clippedForward(F_PTIO_DM_1 I_PTIO_DM_1)
                hasTimeIntervalSpecString(I_PTIO_DM_1 [1990,1994])
   F_PTIO_DM_2: personHasTitleInOrganization(Janez_Jansa Defence_Minister Slovenia)
                holdsThroughout(F_PTIO_DM_2 I_PTIO_DM_2)
                clippedBackward(F_PTIO_DM_2 I_PTIO_DM_2)
                clippedforward(F_PTIO_DM_2 I_PTIO_DM_2)
                hasTimeIntervalSpecString(I_PTIO_DM_2 [2000,2000])
     F_PTIO_PM: personHasTitleInOrganization(Janez_Jansa Prime_Minister Slovenia)
                holdsThroughout(F_PTIO_PM I_PTIO_PM)
                clippedBackward(F_PTIO_PM I_PTIO_PM)
                hasBeginningTimePoint(I_PTIO_PM I_PTIO_PM_beginning)
                hasTimePointSpecString(Slovenia_2004_Election_Day 2004-11-09)
                timePointGreaterThanOrEqualTo(I_PTIO_PM_beginning Slovenia_2004_Election_Day)
                hasReportingAspect(I_PTIO_PM Ongoing)
                ref(annotation I_PTIO_PM)
                annotation(document annotation)
                hasReportingChronusSpecString(document 2007-12-28)
```

Figure 8. Formal statements associated with the time map in Figure 6


BirthEvent(?person, Ljubljana)
1950-01-01
1959-12-31
attendsSchool(?person Ljubljana_University)
?attendanceIntervalSpec
1980-01-01
1989-12-31
personHasTitleInOrganization(?person ?title ?org)
?titleIntervalSpec

Query 1: *Find all persons who were born in Ljubljana in the 1950s and* ***attended Ljubljana University in the 1980s***, the titles that they held, the organizations in which they held these titles, *and the maximal known time periods over which* ***they attended*** *and* held these titles.

Figure 9. The time map covering our Query1

C. Adaptation to test or application question authoring

We might reuse much of the same machinery in a question authoring interface, in which a user can formalize a query, as illustrated for Query1 in Figure 9. This time map display is even less cluttered than the one for this query's supporting statements, for a couple of reasons.

- We are making general statements, rather than specific ones, so don't use as many dates or long identifiers. Rather, we use variables (here beginning with ?).
- We are asking about only one answer (set of variable values satisfying the query) at a time. The supporting statements in our earlier time map include three separate sets of bindings for the variables ?title and ?org.

We have introduced intervals to represent *the 1950s* and *the 1980s,* and we have selected time point/interval relationships appropriate to the query's conditions. These relationships are associated with particular idioms used in our formalization in Figure 10.

```
            hasPersonBorn(?birth ?person)
            hasBirthEventGPE-spec(?birth ?GPEspec)
            hasCityTownOrVillage(?GPEspec ljubljana_Ljubljana_Slovenia)
            hasTimeIntervalSpecString(?I_range_birth [1950-01-01,1959-12-31])
            occursWithin(?birth ?I_range_birth)
            hasTimeIntervalSpecString(?I_range_school [1980-01-01,1989-12-31])
            holdsWithin(?F_school ?I_range_school)
            maximallyHoldsThroughout(?F_school ?I_school)
            hasTimeIntervalSpecString(?I_school ?attendanceIntervalSpec)
?F_title:   personHasTitleInOrganization(?person ?title ?org)
            maximallyHoldsThroughout(?F_title ?I_title)
            hasTimeIntervalSpecString(?I_title ?titleIntervalSpec)
```

Figure 10. Formalization of the query in Figure 9's time map, covering Query 1

Our query asks about the "maximal known time periods" over which the fluents hold, and we associate (via a query authoring workflow step) an "interval spec" variable with each fluent's observation interval. Per our formalization, this will be bound, on successful query execution, to a string that describes lower and upper bounds on the observation interval's beginning point, ending point, and duration. The formalization uses the properties occursWithin (for *born in the 1950s*) and holdsWithin (for *attended school in the 1980s*) to accommodate the temporal relations selected for the query authoring time map. We know to use maximallyHoldsThroughout (vice the less restrictive holdsThroughout) for the fluents' observation intervals because the query's author has included (via the invoked widget) associated spec string variables.

Thus, it appears that we might enable non-specialists to author effective test queries (or, in a transition/application setting, domain queries), without requiring the intervention of a KR specialist. One angle on this proposed work might be to determine the extent to which readers who are not (temporal) knowledge representation specialists can perform such tasks consistently—alternatively, to determine the amount of training (e.g., pages of documentation, number of successfully completed test exercises) required to qualify an otherwise-non-specialist to perform the task well. That said, rather than "dumb down" the task, to accommodate non-expert readers, we propose to ratchet up annotator performance expectations—to achieve the highest-quality results possible so that we can drive research regarding extraction of temporal knowledge by machines from text to new levels of sophistication. The machine reading researchers whose systems are under evaluation quite reasonably ask, before they embark on a mission of technological advancement, "Is this task feasible for humans, with acceptable consistency?" We'd like to answer that question in the best way that we can.

V. Related Work and Proposed Advances

Beyond test questions and answers, the entire machine reading community would benefit from having a large volume of good temporal logic annotations available. Time is a key topic in language understanding, engendering much current community interest. TimeML [4], which emphasizes XML annotation structures rather than RDF ontology and relationships, has been used in the TempEval temporal annotation activities (see, e.g., www.timeml.org/tempeval2/) and advanced as an international standard [5]. We are interested in exploring the synergy between this work and ours.

Others have applied limited temporal reasoning in post-processing of temporal annotations, to…

A. Compute the closure of qualitative pairwise time interval relations, as one step in assessing a machine reader's precision and recall performance (see section A)

B. Ascertain the global coherence of captured qualitative relations (see section B).

Our implementation can go further, as described below.

A. Quantitative temporal relation annotation evaluation

Evaluating temporal annotations typically has been limited to (Allen's [1]) qualitative relations (e.g., before, overlaps, contains), and quantitative information about dates and durations typically has been evaluated only locally—at the level of temporal expressions (AKA "TIMEXs" [3]). The reasoning applied has been strictly interval-based, neglecting important quantitative information about dates and durations widely available in text. This approach is taken by Setzer et al. [9], e.g.

Our temporal reasoning engine, which is point-based, naturally accommodates arbitrary bounds on the metric durations that separate time points and uses global constraint propagation to calculate earliest and latest possible dates/times for any point (including the beginning and ending points of all temporal intervals), as well as tightest bounds on durations.

This approach also usually affords sufficient global perspective for a robust recall statistic. Adapting the standard approach for evaluating interval relations [11], we can discard from our gold standard annotations any redundant relations until we determine a set spanning globally calculated bounds. Then we can count members of this spanning set whose addition to a user's candidate set results in tightening of bounds in the latter, to determine recall.

Only when every member of a set of points is unrelated to the calendar (i.e., we have only point ordering and interval duration information) do we lack calendar bounds supporting meaningful recall assessment. Then, however, by choosing any point in a connected set to serve as a reference (in place of the calendar), we can apply the same approach as above.

It may reasonably be argued that at some threshold of representational complexity the brute force transitive closure-and-spanning tree approach to computing recall and precision of an extracted knowledge base (set of statements) must become impractical. Our quantitative temporal statements are certainly richer than the typical qualitative ones, and (depending on knowledge base size) we may be pushing up against this threshold with them. Our query answering evaluation paradigm is more broadly applicable, presuming inference over extracted statements remains tractable—for the queries of interest.

B. Ascertaining global coherence

Waiting until annotation is done to infer bounds and detect contradictions neglects opportunities to give annotator's (e.g., time map-based) feedback and receive their best-effort corrections. As Bittar et al. [2] comment, "Manually eliminating incoherencies is an arduous task, and performing an online coherence check during annotation of relations would be extremely useful in a manual annotation tool." We propose this.

VI. SUMMARY

We have outlined existing and anticipated future benefits of an end-to-end methodology for…

- Annotating formal RDF statements representing temporal knowledge to be extracted from text
- Authoring and validating test and/or application queries to exercise that knowledge.

These capabilities are supported by an implemented temporal reasoning engine. They and the engine are intended to support a timeline tool conceived for use by intelligence analysts. We have explained how these benefits can advance machine reading technology by increasing both sophistication and quality expectations about temporal annotations and extraction.

ACKNOWLEDGMENT

Thanks to other participants in DARPA's Machine Reading research program—especially to other members of the SAIC evaluation team, including Global InfoTek (the author's former employer).

REFERENCES

[1] J. Allen, "Maintaining knowledge about temporal intervals," in Communications of the ACM. 26, pp. 832–843, November 1983.

[2] A. Bittar, P. Amsili, P. Denis, L. Danlos, "French TimeBank: An ISO-TimeML annotated reference corpus," in Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, pp. 130–134, 2011.

[3] L. Ferro, L. Gerber, I. Mani, B. Sundheim, and G. Wilson, "TIDES 2005 standard for the annotation of temporal expressions," MITRE Corporation, 2005.

[4] J. Pustejovsky et al., "TimeML: Robust specification of event and temporal expressions in text," AAAI Technical Report SS-03-07, 2003.

[5] J. Pustejovsky, K. Lee, H. Bunt, and L. Romary, "ISO-TimeML: an international standard for semantic annotation," in Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC), Malta. May 18-21, 2010.

[6] R. Schrag, J. Carciofini, and M. Boddy, "Beta-TMM Manual (version b19)," Technical Report CS-R92-012, Honeywell SRC, 1992.

[7] R. Schrag, M. Boddy, and J. Carciofini. "Managing disjunction for practical temporal reasoning," in Principles of Knowledge Representation and Reasoning: Proceedings of the Third International Conference (KR-92), pp 36–46, 1992.

[8] R. Schrag, "Best-practice time point ontology for event calculus-based temporal reasoning," 7th International Conference on Semantic Technologies for Intelligence, Defense, and Security (STIDS), 2012.

[9] A. Setzer, R. Gaizauskas, and M. Hepple, "The role of inference in the temporal annotation and analysis of text," Language Resources and Evaluation v. 39, pp. 243–265, 2005.

[10] M. Shanahan, "The event calculus explained," in Artificial Intelligence Today, ed. M. Wooldridge and M. Veloso, Springer Lecture Notes in Artificial Intelligence no. 1600, pp.409-430, 1999.

[11] X. Tannier and P Muller, "Evaluation metrics for automatic temporal annotation of texts," in Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08), 2008.

Horizontal Integration of Warfighter Intelligence Data

A Shared Semantic Resource for the Intelligence Community

Barry Smith
University at Buffalo, NY, USA

Tatiana Malyuta
Data Tactics Corp. VA, USA

William S. Mandrick
Data Tactics Corp. VA, USA

Chia Fu
Data Tactics Corp. VA, USA

Kesny Parent
Intelligence and Information Warfare Directorate (I^2WD) CERDEC, MD, USA

Milan Patel
Intelligence and Information Warfare Directorate (I^2WD) CERDEC, MD, USA

Abstract - **We describe a strategy that is being used for the horizontal integration of warfighter intelligence data within the framework of the US Army's Distributed Common Ground System Standard Cloud (DSC) initiative. The strategy rests on the development of a set of ontologies that are being incrementally applied to bring about what we call the 'semantic enhancement' of data models used within each intelligence discipline. We show how the strategy can help to overcome familiar tendencies to stovepiping of intelligence data, and describe how it can be applied in an agile fashion to new data resources in ways that address immediate needs of intelligence analysts.**

Index Terms—**semantic enhancement, ontology, joint doctrine, intelligence analytics, intelligence data retrieval.**

I. Introduction

The horizontal integration of warfighter intelligence data is described in Chairman of the Joint Chiefs of Staff Instruction J2 CJCSI 3340.02A [1] in the following way:

> Horizontally integrating warfighter intelligence data improves the consumers' production, analysis and dissemination capabilities. Horizontal Integration (HI) requires access (including discovery, search, retrieval, and display) to intelligence data among the warfighters and other producers and consumers via standardized services and architectures. These consumers include, but are not limited to, the combatant commands, Services, Defense agencies, and the Intelligence Community.

Horizontal integration is achieved when multiple heterogeneous data resources become aligned or harmonized in such a way that search and analysis procedures can be applied to their combined content as if they formed a single resource. We describe here a methodology that is designed to achieve such alignment in a flexible and incremental way. The methodology is applied to the source data at arm's length, in such a way that the data itself remains unaffected by the integration process.

Ironically, attempts to achieve horizontal integration have often served to consolidate the very problems of data stovepiping which they were designed to solve. Integration solution A is proposed; and works well for the data and purposes for which it was originally tailored; but it does not work at all when applied to new data, or to existing data that has to be used in new ways. Such failures arise for a variety of reasons, many of which have to do with the fact that integration systems are too closely tied to specific features of the (software/workflow) environments for which they have been developed. We propose a strategy for horizontal integration which seeks to avoid such problems by being completely independent of the processes by which the data store to which it is applied is populated and utilized. This strategy, which draws on standard features of what is now called 'semantic technology' [2], has been used successfully for over ten years to advance integration of the data made available to bioinformaticians, molecular biologists and clinical scientists in the wake of the successful realization of the Human Genome Project [3, 4]. The quantity and variety of such data – now spanning all species and species-interactions, at all life stages, at multiple granularity levels, and pertaining to thousands of different diseases – is at least comparable to the quantity and variety of the data which need to be addressed by intelligence analysts. As we describe in more detail in [5], however, today's dynamic environment of military operations (from Deterrence to Crisis Response to Major Combat Operations) is one in which ever new data sources are becoming salient to intelligence analysis, in ways which will require a new sort of agile support for retrieval, integration and enrichment of data. We will thus address in particular how our strategy can be rapidly reconfigured to allow its application to emerging data sources.

The strategy is one of a family of similar initiatives designed both to rectify the legacy effects of data stovepiping in the past and to counteract the problems caused by new stovepipes arising in the future. It is currently being applied within the DCGS-A Standard Cloud (DSC) initiative, which is part of the Distributed Common Ground System-Army [6], the principal Intelligence, Surveillance and Reconnaissance (ISR) enterprise for the analysis, processing and exploitation of all US Army intelligence data, and which is designed to be interoperable with other DCGS programs. The DSC Cloud is a military program of record in the realm of Big Data that is accumulating data from multiple diverse sources and with high rapidity of change. In [5, 7] we described how the proposed strategy is already helping to improve search results within the DSC Cloud in ways that bring benefits to intelligence analysts. In this communication, we present the underlying methodology describing also how it draws on resources developed in an incremental way that takes account of lessons learned in successive phases of application of the methodology to new kinds of data. Here we provide only general outlines. Further details and supplementary material are presented at [8].

II. Overcoming Semantic Stovepipes

Every data store is based on some data model which specifies how the data in the store is to be organized. Since communities that develop data stores do so always to serve some particular purpose, so each data model, too, is oriented around some specific purpose. Data models have been created in uncoordinated ways to address these different purposes, and they typically cannot easily be modified to serve additional purposes. Where there is a need to combine data from multiple existing systems, therefore, the tendency has been to invest what may be significant manual effort in building yet another data store, thereby contributing further to a seemingly never-ending process of data stovepipe proliferation.

To break out of this impasse, we believe, a successful strategy for horizontal integration must operate at a different level from the source data. It must be insulated from entanglements with specific data models and associated software applications, and it must be marked by a degree of persistence and of relative technological simplicity over against the changing source data to which it is applied.

The strategy we propose, which employs by now standard methods shared by many proponents of semantic technology [2], begins by focusing on the terms (labels, acronyms, codes) used as column headers in source data artifacts. The underlying idea is that it is very often the case that multiple distinct terms $\{t_1, \ldots, t_n\}$ are used in separate data sources with one and the same meaning. If, now, these terms are associated with some single 'preferred label' drawn from some standard set of such labels, then all the separate data items associated with the $\{t_1, \ldots t_n\}$ will become linked together through the corresponding preferred labels.

Such sets of preferred labels provide the starting point for the creation of what are called 'ontologies', which are created (1) by selecting a preliminary list of labels in collaboration with subject-matter experts (SMEs); (2) by organizing these labels into graph-theoretic hierarchies structured in terms of the *is_a* (or subtype) relation and adding new terms to ensure *is_a* completeness; (3) by associating logical definitions, lists of synonyms and other metadata with the nodes in the resultant graphs. One assumption widespread among semantic technologists is that ontology-based integration is best pursued by building large ontology repositories (for example as at [9]), in which, while use of languages such as RDF or OWL is standardized, the ontologies themselves are unconstrained. Our experience of efforts to achieve horizontal integration in the bioinformatics domain, however, gives us strong reason to believe that, in order to counteract the creation of new ('semantic') stovepipes, we must ensure that the separate ontologies are constructed in a collaborative process which ensures a high degree of integration among the ontologies themselves. To this end, our strategy imposes on ontology developers a common set of principles and rules and an associated common architecture and governance regime in order to ensure that the suite of purpose-built ontologies evolves in a consistent and non-redundant fashion.

III. Defining Features of the SE Approach

Associating terms used in source data with preferred labels in ontologies leads to what we call 'Semantic Enhancement' (SE) of the source data. The ontologies themselves we call 'SE ontologies', and the semantically enhanced source data together form what we call the 'Shared Semantic Resource' (SSR). To create this resource in a way that supports successful integration, our methodology must ensure realization of the following goals, which are common to many large-scale horizontal integration efforts:

- It must support an incremental process of ontology creation in which ontologies are constructed and maintained by multiple distributed groups, some of them associated with distinct agencies, working to a large degree independently.
- The content of each ontology must exist in both human-readable (natural language) and computable (logical) versions in order to allow the ontologies to be useful to multiple communities, not only of software developers and data managers, but also of intelligence analysts.
- Labels must be selected with the help of SMEs in the relevant domains. This is not because these labels are designed to be used by SMEs at the point where source data are collected; rather it is to ensure that the ontologies reflect the features of this domain in a way that coheres as closely as possible with the understanding of those with relevant expertise. Where necessary – for instance in cases where domains overlap – multiple synonyms are incorporated into the structure of the relevant ontologies to reflect usage of different communities of interest.
- Ontology development must be an arms-length process, with minimal disturbance to existing data and data models, and to existing data collection and management workflows and application software.
- Ontologies must be developed in an incremental process which approximates by degrees to a situation in which there is one single reference ontology for each domain of interest to the intelligence community.
- The ontologies must be capable of evolving in an agile fashion in response to new sorts of data and new analytical and warfighter needs.
- The ontologies must be linked together through logical definitions [10], and they must be maintained in such a way that they form a single, non-redundant and consistently evolving integrated network. The fact that all the ontologies in this network are being used simultaneously to create annotations of source data artifacts will in turn have the effect of virtually transforming the latter into an evolving single SSR, to

which computer-based retrieval and analysis tools can be applied.

The ontology development strategy we advocate thus differs radically from other approaches (such as are propounded in [11]), which allow contextualized inconsistency. For while of course source data in the intelligence domain will sometimes involve inconsistency – the data is derived, after all, from multiple, and variably reliable, sources –, to allow inconsistency among the ontologies used in annotations would, from our point of view, defeat the purposes of horizontal integration.

To achieve the goals set forth above, we require:

- A set of ontology development rules and principles, a shared governance and change management process, and a common architecture incorporating a common, domain-neutral, upper-level ontology.
- An ontology registry in which all ontology initiatives and emerging warfighter and analyst needs will be communicated to all collaborating ontology developers.
- A simple, repeatable process for ontology development, which will promote coordination of the work of distributed development teams, allow the incorporation of SMEs into the ontology development process, and provide a software-supported feedback channel through which users can easily communicate their needs, and report errors and gaps to those involved in ontology development.
- A process of intelligence data capture through 'annotation' [12] or 'tagging' of source data artifacts [7], whereby the preferred labels in the ontologies are associated incrementally with the terms embedded in source data models and terminology resources in such a way that the data in distinct data sources, where they pertain to a single topic, are represented in the SSR in a way that associates them with a single ontology term. Currently the annotation process is primarily manually driven, but it will in the future incorporate the use of Natural Language Processing (NLP) tools. Importantly, the process of annotation incrementally tests the ontologies against the data to which they must be applied, thereby helping to identify errors and gaps in the ontologies and thus serving as a vital ontology quality assurance mechanism [12].

IV. ONTOLOGICAL REALISM

The key idea underlying the SE methodology is that the successful application of ontologies to horizontal data integration requires a process for creating ontologies that is independent of specific data models and software implementations. This is achieved through the adoption of what is called 'ontological realism' [13], which rests on the idea that ontologies should be constructed as representations, not of data or of data models, but rather of the types of entities in reality to which the data relate.

The first step in the development of an ontology for a domain that has been identified as a target for intelligence analysis is thus *not* to examine what types of data we have about that domain. Rather, it is to establish in a data-neutral fashion the salient types of entities within the domain, and to select appropriate preferred labels for these types, drawing for guidance on the language used by SMEs with corresponding domain expertise. In addition, we rely on authoritative publications such as the capstone Joint Publication (JP) 1 of Joint Doctrine and the associated Dictionary (JP 1-02) [14, 15] (see Figure 1), applying adjustments where necessary to ensure logical consistency. The resultant preferred labels are organized into simple hierarchies of subtype and supertype, and each label is associated with a simple logical definition, along the lines illustrated (in a toy example) in Table 1.

vehicle =def: an object used for transporting people or goods

- **personnel carrier** =def. a vehicle that is used for transporting persons
- **tractor** =def: a vehicle that is used for towing
- **crane** =def: a vehicle that is used for lifting and moving heavy objects

vehicle platform=def. means of providing mobility to a vehicle

- **wheeled platform**=def. a vehicle platform that provides mobility through the use of wheels

Table 1. Fragments of asserted ontologies

V. REALIZATION OF THE STRATEGY

There is a tension, in attempts to create a framework for horizontal integration of large and rapidly changing bodies of data, which turns on the fact that (1) to secure integration the framework needs to be free from entanglements with specific data models; yet (2) to allow effective representation of data, the framework needs to remain as close as possible to those same data models.

This same tension arises also for the SE approach, where it is expressed in the fact that:

(1) The SSR needs to be created on the basis of persistent, logically well-structured ontologies designed to be reused in relation to multiple different bodies of data; yet:

(2) To ensure agile response to emerging warfighter needs, its ontologies must be created in ways that keep them as close as possible to the new data that is becoming available locally in each successive stage.

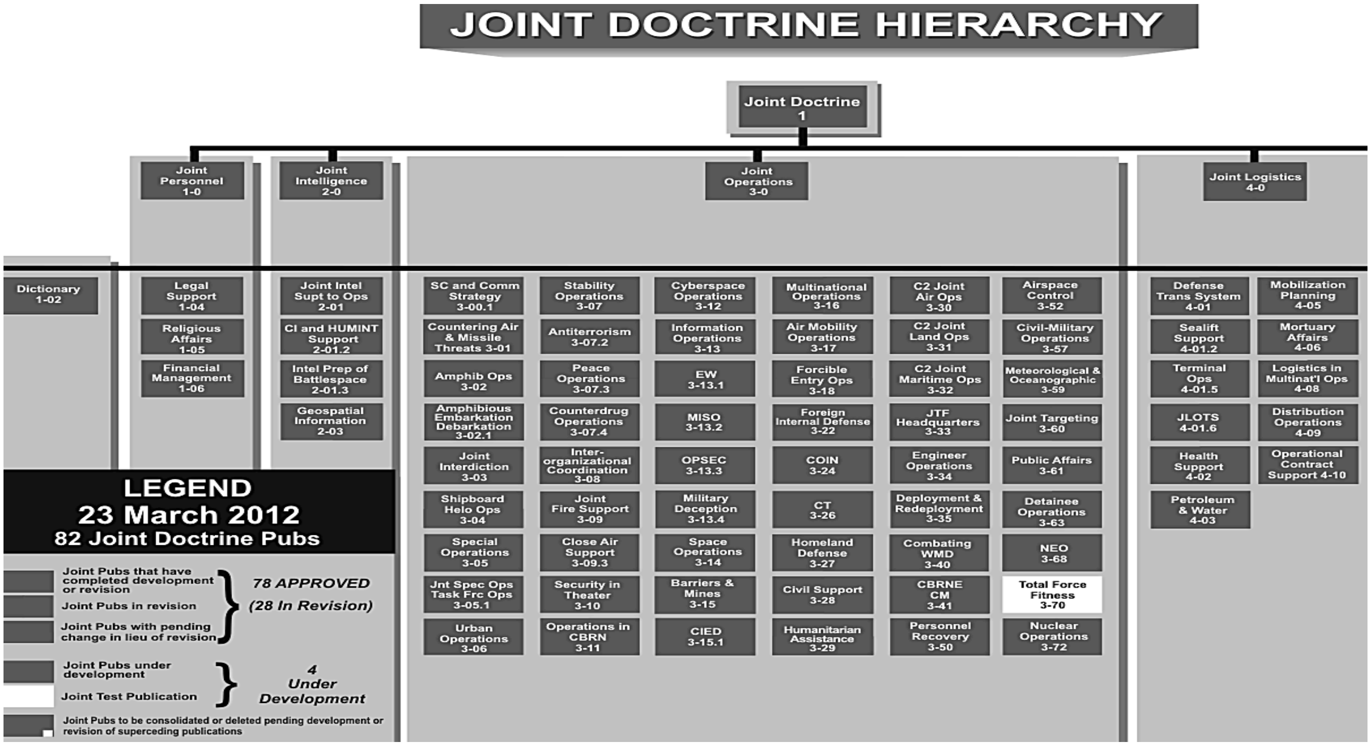


Figure 1 - Joint Doctrine Hierarchy

To resolve this tension, the SE strategy incorporates a distinction between two sorts of ontologies, called '*reference*' and '*application*' ontologies, respectively. By '*reference ontology*', we mean an ontology that captures generic content and is designed for aggressive reuse in multiple different types of context. Our assumption is that most reference ontologies will be created manually on the basis of explicit assertion of the taxonomical and other relations between their terms. By '*application ontology*', we mean an ontology that is tied to specific local applications. Each application ontology is created by using ontology merging software [16] to combine new, local content with generic content taken over from relevant reference ontologies [17,18], thereby providing rapid support for information retrieval in relation to particular bodies of intelligence data but in a way that streamlines the task of ensuring horizontal integration of this new data with the existing content of the SSR.

A. Principle of Single Inheritance

Our ontologies are 'inheritance' hierarchies in the sense that everything that holds (is true) of the entities falling under a given parent term holds also of all the entities falling under its *is_a* child terms at lower levels. Thus in Figure 2, for example, everything that holds of 'vehicle' holds also of 'tractor'. Each reference ontology is required to be created around an inheritance hierarchy of this sort that is constructed in accordance with what we call the *principle of asserted single inheritance*. This requires that for each reference ontology the *is_a* hierarchy is asserted, through explicit axioms (subclass axioms in the OWL language), rather than inferred by the reasoner. In addition it requires that this asserted *is_a* hierarchy is a monohierarchy (a hierarchy in which each term has at most one parent). This requirement is imposed for reasons of efficiency and consistency: it allows the total ontology structure to be managed more effectively and more uniformly across distributed development teams – for example by aiding positioning and surveyability of terms. It brings also computational performance benefits [23] and provides an easy route (described in Section V.E below) to the creation of the sorts of logical definitions we will need to support horizontal integration. The principle of asserted single inheritance comes at a price, however, in that it may require reformulation of content – for example deriving from multi-inheritance ontologies already developed by the intelligence community – that is needed to support the creation of the SSR. Again, our experience in the biomedical domain is that such reformulation, while requiring manual effort, is in almost all cases trivial, and that, where it is not trivial, the effort invested often brings benefits in terms of greater clarity as to the meanings and interrelationships of the new terms that need to be imported into the SE framework.

B. A Simple Case Study

Imagine, now, that there is a need for rapid creation of an application ontology incorporating preferred labels to describe artillery units available to some specific military unit called 'Delta Battery'. Such an ontology is enabled, first, by selecting from existing reference ontologies the terms needed to address the data in hand, for example of the sort used in Table 1. Second we define supplementary terms needed for our specific local case, as in Table 2.

Some of these terms may later be incorporated into corresponding asserted ontologies within the SE suite. For our present purposes, however, they can be understood as being simply combined together with the associated asserted ontology terms using ontology merging software, for example as developed by the Brinkley [17,19,17] and He [20,21] Groups. Because of the way the definitions are formulated, it is then possible to apply an automatic reasoner [22] to the result of merger to infer new relations, and thereby to create a new ontology hierarchy, as in Figure 2. Note that, in contrast to the reference ontologies from which it is derived, such an application ontology need not satisfy the principle of single inheritance. Note, too, that the definitions are exploited by the reasoner not only to generate the new inferred ontology, but also to test its consistency both internally and with the reference ontologies from which it is derived.

artillery weapon = def. device for projection of munitions beyond the effective range of personal weapons
artillery vehicle = def. vehicle designed for the transport of one or more artillery weapons
wheeled tractor = def. a tractor that has a wheeled platform
tracked tractor = def. a tractor that has a tracked platform
artillery tractor = def. an artillery vehicle that is a tractor
wheeled artillery tractor = def. an artillery tractor that has a wheeled platform
Delta Battery artillery vehicle=def. an artillery vehicle that is at the disposal of Unit Delta
Delta Battery artillery tractor=def. an artillery tractor that is at the disposal of Unit Delta

Table 2: Examples of supplementary terms and definitions

The strategy is designed to guarantee

(1) that salient reference ontology content is preserved in the new, inferred ontology in such a way that

(2) the latter can be used to semantically enhance newly added data very rapidly, and thereby

(3) bring about the horizontal integration of these data with all remaining contents of the SSR.

While ontology software has the capacity to support rapid ontology merger and consistency checking, we note that the inferred application ontology that is generated may on first pass fail to meet the local application needs. Thus, multiple iterations and investment of manual effort are needed.

Requiring that all inferred ontologies rest on reference ontology content serves not only to ensure consistency, but also to bring about what we can think of as the *normalization* [23] of the evolving ontology suite. (This is in loose analogy with the process of normalization of a vector space, where a *basis* of orthogonal unit vectors is chosen, in terms of which every vector in the whole space can be represented in a standard way.)

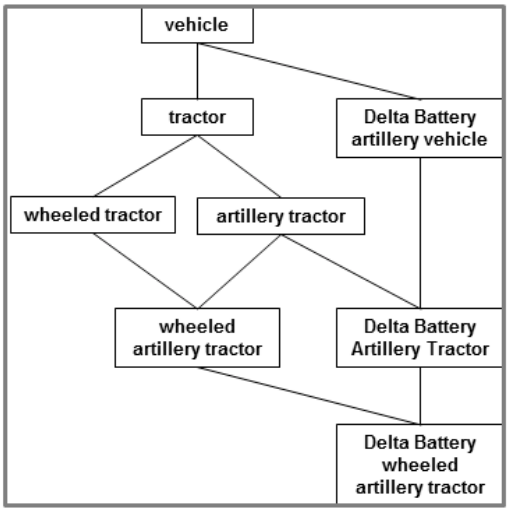

vehicle
tractor
Delta Battery artillery vehicle
wheeled tractor
artillery tractor
wheeled artillery tractor
Delta Battery Artillery Tractor
Delta Battery wheeled artillery tractor

Figure 2. Inferred ontology of Delta Battery artillery vehicles. Child-parent links are inferred by the reasoner from the content of merged reference ontologies and from definitions of the supplementary terms. Note that some terms have multiple parents.

A suite of normalized ontologies is easier to maintain, because globally significant changes – those changes which potentially have implications across the entire suite of ontologies – can be made in just one place in the relevant reference ontology, thereby allowing consequent changes in the associated inferred ontologies to be propagated automatically. This makes ontology-based integration easier to manage and scale, because when single-inheritance modules serve to constrain allowable sorts of combinations, this makes it easier to avoid problems of combinatorial explosion.

C. *Modularity of Ontologies Designed for Reuse*

The reference ontologies within the SE suite are to be conceived as forming a set of plug-and-play ontology modules such as the Organization Ontology, Geospatial Feature Ontology, Human Physical Characteristics Ontology, Event Ontology, Improvised Explosive Device Component Ontology, and so on. These modules need to be created at different levels of generality, with the architecture of the higher level reference ontologies being preserved as we move down to lower levels.

Each module has its own coverage domain, and the coverage domains for the more specific modules (for example *artillery vehicle*, *military engineering vehicle*) are contained as parts within the coverage domains of the more general modules (for example *vehicle*, *equipment*). It is our intention that the full SE suite of ontologies will mimic the sort of hierarchical organization that we find in the Joint Doctrine Hierarchy [15], and our strategy for identifying and demarcating modules will wherever possible follow the demarcations of Joint Doctrine. The goal is to specify a set of levels of greater and lesser generality: for example *Intelligence*, *Operations*, *Logistics*, at one level; *Army Intelligence*, *Navy Intelligence*, *Airforce Intelligence*, at the next lower level; and so on. Ideally, the set of modules on each level are non-redundant in the sense that (1) they deal with non-overlapping domains of entities and thus (2) do not contain any terms in common. In this way the more general content at higher levels is inherited by the lower levels and thus does not need to be recreated anew. As the history of doctrine writing shows, drawing such demarcations and ensuring consistency of term use in each sibling domain on any given level is by no means easy. Here, however, we will have the advantage that the ontology resource we are creating is not designed to serve as a terminology and doctrine set for use by multiple distinct groups of warfighters. Rather, it is designed for use behind the scenes for the specific purpose of data discovery and integration. Thus it is assumed that disciplinary specialists will continue to use their local terminologies (and taxonomies) at the point where source data is being collected, even while, thanks to the intermediation of ontology annotation, they are contributing to the common SSR. At the same time, community-specific terms will wherever possible be added to the SE ontology hierarchies as synonyms. This will contribute not only to the effectiveness of ontology review by SMEs but also to the applicability of NLP technology in support of automatic data annotation.

Our goal is to build the SE ontology hierarchy in such a way as to ensure non-redundancy by imposing the rule that, for each salient domain, one single reference ontology module is developed for use throughout the hierarchy. Creating non-redundant modules in this way is, we believe, indispensable if we are to counteract the tendency for separate groups of ontology developers to create new ontologies for each new purpose.

D. *Benefits of Normalized Ontology Modules*

The grounding in modular, hierarchically organized, non-redundant, asserted ontology modules brings a number of significant benefits, of a sort which are being realized already in the biomedical ontology research referred to above [3]. First, it creates an effective division of labor among those involved in developing, maintaining and using ontologies. In particular, it allows us to exploit the existing disciplinary division of knowledge and expertise among specialists in the domains and subdomains served by the intelligence community. To ensure population of the ontologies in a consistent fashion, we are training selected SMEs from relevant disciplines in ontology development and use; at the same time we are ensuring efficient feedback between those who are using ontologies in annotating data and those who are maintaining the ontologies over time in order to assure effective update, including correction of gaps and errors.

Second, it ensures that the suite of asserted ontologies is easily surveyable: developers and users of ontologies can easily discover where the preferred label equivalents of given terms are to be found in the ontology hierarchy; they can also easily determine where new terms, or new branches, should be inserted into the SE suite. Thus, where familiar problems arise when mergers are attempted of independently developed ontologies and terminology content, the incremental approach adopted here implies that mergers will be applied almost exclusively only (1) to the content of reference ontologies developed according to a common methodology and reviewed at every stage for mutual consistency and (2) to application ontology content developed by downward population from the evolving ontology suite.

E. *Creating Definitions*

The principle of single inheritance allows application of a simple rule for formulating definitions of ontology terms, whereby all definitions are required to have the form:

an *S* = Def. a *G* which *D*s

where ‘S’ (for: species) is the term to be defined, ‘G’ (for: genus) is the immediate parent term of ‘S’ in the relevant SE asserted ontology, and ‘D’ (for: differentia) is the species-criterion, which specifies what it is about certain G’s which makes them S’s. (Note that this rule can be applied consistently only in a context where every term to be defined has exactly one asserted parent.)

As more specific terms are defined through the addition of more detailed differentia, their definitions encapsulate the taxonomic information relating the corresponding type within the SE ontology to the sequence of higher-level terms by which it is connected to the corresponding ontology root. The task of formulating definitions thereby serves as a quality control check on the correctness of the constituent hierarchies, just as awareness of the hierarchy assists in the formulation of coherent definitions.

A further requirement is that the definitions themselves use (wherever possible) preferred labels which are taken over from other ontologies within the SE suite. Where appropriate terms are missing, the SE registry serves as a feedback channel through which the corresponding need can be transmitted to those tasked with ontology maintenance. The purpose of this requirement is to bring it about that the SE ontologies themselves will become incrementally linked together via logical relations in the way needed to ensure the horizontal integration of the data in the SSR that have been annotated with their terms. And as more logical definitions are added to the SE suite, the more its separate modules begin to act like a single, integrated network. All of this brings further benefits, including:

- Lessons learned in experience developing and using one module can be easily propagated throughout the entire system.
- The value of training in ontology development in any given domain module is increased, since the results of such training can easily be re-applied in relation to other modules.
- The incrementally expanding stock of available reference ontology terms will help to make it progressively easier to create in an agile fashion new application ontologies for emerging domains.

- The expanding set of logical definitions cross-linking the ontologies in the SE suite will mean that the use of ontology reasoners [22] for quality assurance of both asserted and inferred ontologies will become progressively more effective. These same reasoners will then be able to be used to check the consistency of the resultant annotations; and when inconsistencies are detected, these can be flagged as being of potential significance to the intelligence analyst.

VI. From Data to Decisions: An Example

Suppose, for example, that analysts are faced with a large body of new data pertaining to activities of organizations involved in the financing of terrorism through drug trafficking. The data is presented to them in multiple different formats, with multiple different types of labels (acronyms, free text descriptions, alphanumeric identifiers) for the types of organizations and activities involved.

To create a semantically enhanced and integrated version of these data for purposes of indexing and retrieval, analysts and ontology developers can use as their starting point the Organization Ontology which has already been populated with many of the general terms they will need across the entire domain of organizations, both military and non-military, formal and informal, family- or tribe- or religion-based, and so on. It will also contain the terms they need to define different kinds of member roles, organizational units and sub-units, chains of authority, and so on.

Adherence to the SE principles ensures that the Organization Ontology has been developed in such a way as to be interoperable, for example, with the Financial Event and Drug Trafficking Ontologies. Portions of each of these modules can thus be selected for merger in the creation of a new, inferred ontology, which can rapidly be applied to annotation of the new drug-financed terrorism data, which thereby becomes transformed from a mere collection of separate data sources into a single searchable store horizontally integrated within the SSR.

VII. Upper-, Mid-and Lowest-Level Ontologies

The SE suite of ontologies is designed to serve *horizontal* integration. But, it depends also on what we can now recognize as a *vertical* integration of asserted ontologies through the imposition of a hierarchy of ontology levels. In general, the SE methodology requires that all asserted ontologies are created via downward population from a common top-level ontology, which embodies the shared architecture for the entire suite of asserted ontologies – an architecture that is automatically inherited by all ontologies at lower levels.

Here, the *level* of an ontology is determined by the level of generality of the types in reality which its nodes represent. The Upper Level Ontology (ULO) in the SE hierarchy must be maximally general – it must provide a high-level domain-neutral representation of distinctions between objects and events, objects and attributes, roles, locations, and so forth. For this purpose we select the Basic Formal Ontology 2.0 (BFO), which has been thoroughly tested in multiple application areas [8,24]. Its role is to provide a framework that can serve as a starting point for downward population in order to ensure consistent ontology development at lower levels. Since almost all SE ontology development is at the lower levels within the hierarchy, BFO itself will in most cases be invisible to the user.

The Mid-Level Ontologies (MLOs) introduce successively less general and more detailed representations of types which arise in successively narrower domains until we reach the Lowest Level Ontologies (LLOs). These LLOs are maximally specific representation of the entities in a particular one-dimensional domain, as illustrated in Table 3.

Some MLOs are created by adding together LLO component modules, for example, the Person MLO may be created by conjoining person-relevant ontology components from Table 3 such as: Person Name, Person Date, Hair Color, Gender, and so on. More complex MLOs will involve the use of reasoners to generate ontologies incorporating inferred labels such as ‘Male Adult’, ‘Female Infant’, and so on, along the lines sketched in Section V.B above.

Person Name (with types such as: FirstName, LastName, …)
Hair Color (with types such as Grey, Blonde, …)
Military Role (with types such as: Soldier, Officer, …)
Blood Type (with types: O, A, …)
Eye Color (with types: Blue, Grey, …)
Gender (with types: Male, Female, …)
Age Group (with types: Infant, Teenager, Adult, …)
Person Date (with types: BirthDate, DeathDate, …)
Education History (with types: HighSchoolGraduation, …)
Education Date (with types: DateOfGraduation, …)
Criminal History (with types: FirstArrest, FirstProsecution, …)
Citizenship (based on ISO 3166 Country Codes)

Table 3. Examples of Lowest Level Ontologies (LLOs)

Figure 3 illustrates the rough architecture of the resultant suite of SE ontologies on different levels, drawing on the top-level architecture of Basic Formal Ontology.

VIII. Conclusion

In any contemporary operational environment, decision makers at all levels, from combatant commanders to tactical-level team leaders, need timely information pertaining to issues ranging from insurgent activity to outbreaks of malaria and from key-leader engagements to local elections. This requires the exploitation by analysts of a changing set of highly disparate databases and other sources of information, whose horizontal integration will greatly facilitate this data to decision cycle.

The SE strategy is designed to create the resources needed to support such integration incrementally, with thorough testing at each successive stage, and one of our current pilot projects is designed to identify the problems which arise when the SE methodology is applied to support collaboration across distinct intelligence agencies, including exploring how independently developed legacy ontologies can be incorporated into the framework.

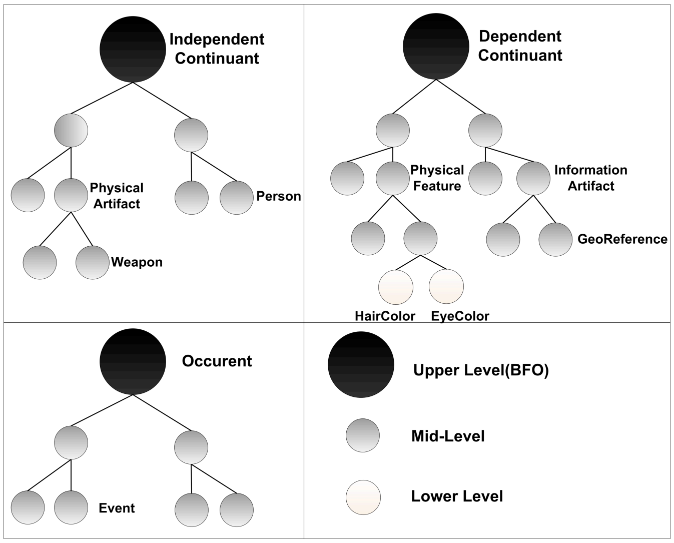


Figure 3. Organization of asserted ontologies

Our work on using SE ontologies for purposes of annotation has been executed thus far both manually and with NLP support. The results of this work have been found useful to indexing and retrieval of large bodies of data in the DSC Cloud store. In our next phase we will test its capacity to support rapid creation of application ontologies to address emerging analyst needs. In a subsequent, and more ambitious phase, we plan to explore the degree to which the idea of semantic enhancement can be truly transformative in the sense that it will influence the way in which source data are collected and stored. We believe that such an influence would bring a series of positive consequences flowing from the fact that the asserted ontologies will be focused automatically upon (i.e. represent) the same entities in the battlespace that the operators, analysts, and war-planners are concerned with, and they would treat these entities in the same intuitively organized way. Thus while at this stage all SE ontologies are free of entanglements with specific source data models, our vision for the future is that the success of the approach will provide ever stronger incentives for the use of SE ontologies already in the field. These incentives will exist, because using such ontologies at the point of data collection will guarantee efficient horizontal integration with the contents of the SSR, thereby giving rise to a network effect whereby not only the immediate utility of the collected data will be increased, but so also will the value of all existing data stored within the SSR.

References

[1] Chairman of the Joint Chiefs of Staff Instruction. J2 CJCSI 3340.02A.

[2] P. Hitzler, M. Krötzsch and S. Rudolph, *Foundations of Semantic Web Technologies*, Chapman & Hall, 2009.

[3] Barry Smith, et al., “The OBO Foundry: Coordinated Evolution of Ontologies to Support Biomedical Data Integration”, *Nature Biotechnology*, 25 (11), November 2007, 1251–1255.

[4] Fahim T. Imam, et al., “Development and use of Ontologies Inside the Neuroscience Information Framework: A Practical Approach”, *Frontiers in Genetics*, 2012; 3: 111.

[5] Barry Smith, et al., “Ontology for the Intelligence Analyst”, *Crosstalk: The Journal of Defense Software Engineering* (forthcoming).

[6] Distributed Common Ground System - Army (DCGS-A) What is it? *Pentagon Army Posture Statement*, 27 December 2011.

[7] David Salmen, et al., “Integration of Intelligence Data through Semantic Enhancement”, *Proceedings of the Conference on Semantic Technology in Intelligence, Defense and Security* (STIDS), George Mason University, Fairfax, VA, November 16-17, 2011, CEUR, Vol. 808, 6–13

[8] Supplementary material on Semantic Enhancement: http://ncorwiki.buffalo.edu/index.php/Semantic_Enhancement

[9] http://ontolog.cim3.net/cgi-bin/wiki.pl?OpenOntologyRepository.

[10] Chris J. Mungall et al., “Cross-product extensions of the Gene Ontology”, *Journal of Biomedical Informatics* 44 (2007), 80–86.

[11] Douglas B. Lenat, “CYC: a large-scale investment in knowledge infrastructure”, *Communications of the ACM*, 38 (11), 1995 33-38.

[12] David P. Hill, et al., “Gene Ontology Annotations: What they mean and where they come from*”, BMC Bioinformatics*, 2008; 9(Suppl 5): S2.

[13] Barry Smith and Werner Ceusters, “Ontological Realism as a Methodology for Coordinated Evolution of Scientific Ontologies”, *Applied Ontology*, 5 (2010), 139–188.

[14] Joint Publication 1, Doctrine for the Armed Forces of the United States, Chairman of the Joint Chiefs of Staff. Washington, DC. 20 March 2009.

[15] Joint Electronic Library: The Joint Publications.

[16] Z. Xiang, et al., “OntoFox: Web-Based Support for Ontology Reuse”, *BMC Research Notes*. 2010, 3:175.

[17] Marianne Shaw, et al., “Generating Application Ontologies from Reference Ontologies”, *Proceedings, American Medical Informatics Association Fall Symposium*, 2008, 672-676.

[18] James Malone and Helen Parkinson, “Reference and Application Ontologies.”

[19] James F. Brinkley et al., “Project: Ontology Views.”

[20] http://www.hegroup.org/ontoden/.

[21] J. Hur, et al., “Ontology-based Brucella vaccine literature indexing and systematic analysis of gene-vaccine association network”, *BMC Immunology* 2011, 12:49

[22] OWL 2 Reasoners, http://www.w3.org/2007/OWL/wiki/Implementations.

[23] Rector, A. L. “Modularisation of Domain Ontologies Implemented in Description Logics and Related Formalisms including OWL”. *Proceedings of the 2nd International Conference on Knowledge Capture*, ACM, 2003, 121–128.

[24] Pierre Grenon and Barry Smith, “SNAP and SPAN: Towards Dynamic Spatial Ontology”, *Spatial Cognition and Computation*, 4: 1 (March 2004), 69–103.

Position Papers

A Study of MEBN Learning for Relational Model

Cheol Young Park, Kathryn Blackmond Laskey, Paulo Costa, Shou Matsumoto
Volgenau School of Engineering
George Mason University
Fairfax, VA USA
[cparkf, klaskey, pcosta]@gmu.edu, smatsum2@masonlive.gmu.edu

Abstract— **In the past decade, Statistical Relational Learning (SRL) has emerged as a new branch of machine learning for representing and learning a joint probability distribution over relational data. Relational representations have the necessary expressive power for important real-world problems, but until recently have not supported uncertainty. Statistical relational models fill this gap. Among the languages recently developed for statistical relational representations is Multi-Entity Bayesian Networks (MEBN). MEBN is the logical basis for Probabilistic OWL (PR-OWL), a language for uncertainty reasoning in the Semantic Web. However, until now there has been no implementation of MEBN learning. This paper describes the first implementation of MEBN learning. The algorithm learns a MEBN theory for a domain from data stored in a relational Database. Several issues are addressed such as aggregating influences, optimization problem, and so on. In this paper, as our contributions, we will provide a MEBN-RM (Relational Model) Model which is a bridge between MEBN and RM, and suggest a basic structure learning algorithm for MEBN. And the method was applied to a test case of a maritime domain in order to prove our basic method.**

Keywords: **Probabilistic ontology, Multi-Entity Bayesian networks, PR-OWL, Relational Model/Database, Machine Learning, Statistical Relational Learning**

I. Introduction

Statistical Relational Learning (SRL) is a new branch of machine learning for representing and learning a joint distribution over relational data [1, 2]. As its name suggests, it combines statistical and relational knowledge representations. A relational model represents a domain as a collection of objects that may have attributes and can participate in relationships with other objects. Relational representations are expressive enough for important real-world problems, but until recently have not supported uncertainty. This gap has been filled by SRL methods. Statistical relational knowledge representations combine statistical and relational approaches, allowing representation of a probability distribution over a relational model of a domain. SRL methods allow such representations to be learned from data.

Examples of representation languages for SRL include Probabilistic Relational Models (PRMs), Markov Logic Networks (MLNs), Relational Dependency Networks (RDNs), Bayesian Logic Programs (BLPs), Join Bayes Net (JBN), and Multi-Entity Bayesian Networks (MEBN) [2, 3, 4, 5, 6, and 7].

A comparison of some of the above models is given in [1]. Typically, SRL models provide a representation for relational knowledge, along with methods for both induction and deduction. Relational representations provide both class and instance models. A class model describes statistical information that applies to classes of objects. For example, a class model might describe the false positive and false negative rates for a class of sensor. The instance model is generated from the class model by a deduction method. For example, the instance model would be used to infer the probability that a given detection is a false positive. An induction method learns structure and parameters of a domain theory from observations. For example, induction would be used to learn the false positive and false negative rates from a data set annotated with ground truth.

SRLs have been applied to problems such as Object Classification, Object Type Prediction, Link Type Prediction, Predicting Link Existence, Link Cardinality Estimation, Entity Resolution, Group Detection, Sub-graph Discovery, Metadata Mining, and so on [2].

This paper is concerned with the Multi-Entity Bayesian Networks (MEBN), a relational language that forms the logical basis of Probabilistic OWL (PR-OWL), a language for uncertainty reasoning in the Semantic Web [7, 8]. PR-OWL has been extended to PR-OWL 2, which provides a tighter link between the deterministic and probabilistic aspects of the ontology [9]. MEBN extends Bayesian networks to a relational representation. A MEBN Theory, or MTheory, consists of a set of Bayesian network fragments, or MFrags, that together represent a joint distribution over instances of the random variables represented in the MTheory [7].

However, until now there has been no implementation of induction or learning for MEBN or PR-OWL. This paper describes such an implementation. We follow an approach used by other SRL models [1] and use Relational Database (RDB) to store the observations from which the representation is learned.

This paper focuses a basic learning algorithm that addresses the following issues:

1. Developing a bridge of MEBN and RDB;
2. Developing basic structure and parameter learning for MEBN.

Ultimately, a relational learning algorithm should address issues such as aggregation of data, reference uncertainty, type uncertainty, and continuous variable learning. These issues will be considered for future research.

Our learning method is exact, and assumes discrete random variables, and complete data. It will be evaluated by the inference accuracy test.

In Section 2, we give a brief definition of MEBN and RM as background. In the Section 3, we introduce the MEBN-RM Model. In Section 4, we present the basic structure learning algorithm. The application of the algorithm is described in the Section 5.

II. Multi-Entity Bayesian Networks (MEBN) and Relational Model (RM)

A. Multi-Entity Bayesian Networks (MEBN)

MEBN extends Bayesian Networks (BNs) to represent relational information. BNs have been very successful as an approach to representing uncertainty about many interrelated variables. However, BNs are not expressive enough for relational domains. MEBN extends Bayesian networks to represent the repeated structure of relational domains.

MEBN represents knowledge about a domain as a collection of MFrags, an MFrag is a fragment of a graphical model that is a template of probabilistic relationships among instances of its random variables. Random variables in an MFrag can contain ordinary variables which can be filled in with domain entities. And MFrag includes context, input, and resident node for restriction of entity, reference of node, and random variable respectively. We can think of an MFrag as a class which can generate instances of BN fragments, which can then be assembled into a Bayesian network [7].

B. Relational Model (RM)

In 1969, Edgar F. Codd proposed RM as a database model based on first-order predicate logic [10]. RM is composed of Relation, Attribute, Key, Tuple, Instance, and Cell. Relational database which is the most popular database is based on RM.

III. MEBN-RM Model

As a bridge of MEBN and RM, we suggest MEBN-RM Model which provides a specification for how to match elements of MEBN to elements of RM. Key nodes in MEBN are the context and resident node. To understand this easily, we use the following example of the university relational model.

Course	
Key	Difficulty
c1	low
c2	high
c3	high
c4	low
c5	med
c6	low

Registration		
Course Key	Student Key	Grade
c1	s1	low
c1	s2	high
c2	s2	high
c2	s4	low
c3	s5	med
c4	s6	low

Student	
Key	Advisor
s1	p4
s2	p2
s3	p3
s4	p1
s5	p5
s6	null

Professor	
Key	Major
p1	SYST
p2	OR
p3	OR
p4	CS
p5	SYST
p6	OR

Table 1. Example of university relational model

A. Context Node

In MFrags, context terms (or nodes) are used to specify constraints under which the local distributions apply. Thus, it determines specific entities on an arbitrary situation of a context. In MEBN-RM model, we define four types of data structure corresponding to context nodes: Isa, Slot-filler, Value-Constraint, and Entity-Constraint type.

Type	Name	Example
1	Isa	Isa(Person, P), Isa(Car, C)
2	Value-Constraint	Height(P) = high
3	Slot-Filler	P = OwnerOf(C)
4	Entity-Constraint	Friend(A, B)

Table 2. Context Node Types on MEBN-RM Model

1) Isa

In MEBN, the Isa random variable represents the type of an entity. In a RM, an entity table represents a collection of entities of a given type. Thus, an entity table corresponds to an Isa random variable in MEBN. Note that a relationship table whose primary key is composed of foreign keys does *not* correspond to an Isa RV. A relationship table will correspond to the Entity-Constraint type of Context Node.

2) Value-Constraint

In a case, a value of attribute can limit keys which are related with only the value. For example, Consider Table 1, in which we have the course table with the difficulty attribute. (In our definition, Attribute is descriptive Attribute and Key is Primary Key)

The course table has instances of the key (e.g., c1, c2, c3, c4, c5, and c6). And if we want to focus on a case of the entity with “high” value of the attribute, it will be {c2, c3}. In this case, for the entity, any group of elements related with any attributes can be derived. We encode this into “Difficulty (Course) = high” in MEBN.

3) Slot-Filler

In the table 1, the professor key is used on the student table by a foreign key, Advisor. The foreign key is not primary key in the student table. In this case, the connection will be expressed by “Professor = Advisor (Student)” in MEBN. And its instance will be that s1’s advisor is p4 and so on.

4) Entity-Constraint

The registration table is a relationship table which is a bridge between the course and student entity. In this case, obviously, the registration table will be an intersection group. And this is described as “Registration (Course, Student)” in MEBN.

B. Resident Node

In MFrags, Resident Node can be described as Function, Predicate, and Formula of FOL with a probability distribution. FOL Function consists of arguments and an output, while FOL Predicate consists of arguments and no output, but Boolean output. We define the following relationship between elements of RM and MEBN.

RM	Resident Node
Attribute	Function/ Predicate
Key	Arguments
Cell of Attribute	Output

Table 3. Resident Node Types of MEBN-RM Model

For example, in the table 1, the grade of the registration table is the function having the course and student keys as arguments. Its output will be the cell of the grade such as low, med, and high. On the other hand, if the domain type of the grade is Boolean, it will be the predicate in MEBN.

IV. The Basic Structure Learning For MEBN

To address the issues in Section 1, we suggest a basic structure learning algorithm for MEBN. The initial ingredients of the algorithm are a dataset of RM, a Bayesian Network Structure searching algorithm, and a size of chain. For the parameter learning, we only use Maximum Likelihood Estimation (MLE). The algorithm focuses on discrete variables with complete data. We utilize a standard Bayesian Network Structure searching algorithm to generate a local BN from the joined dataset of RM. To avoid infinite loops, we employed the size of chain. Thus, the process of searching structure will finish in the size of chain.

Firstly, the algorithm creates the default MTheory. All keys of DB are defined as entities of MEBN theory. One default reference MFrag is created. For the all of tables of DB, the dataset for each table is retrieved and, by using the BN structure searching algorithm, a graph is generated from the dataset. If the graph has a cycle and undirected edge, a knowledge expert for the domain sets the arc direction. Based on the revised graph, an MFrag is created. Until the size of chain is reached, the joined datasets which are derived by “Join” command in SQL are retrieved. The graphs related to the joined datasets are generated in the same way as the above. If any nodes of the new generated graph are not used in any MFrags, create the resident node having the name of the dataset of the graph on the default reference MFrag and the new MFrag for the dataset. If not, only make edges between resident nodes in the different MFrags. Lastly, for all resident nodes in the MTheory, LPDs are generated by MLE.

V. Case Study

To evaluate the algorithm, we used a dataset which came from the PROGNOS (Probabilistic OntoloGies for Net-centric Operation Systems) [11, 12]. The purpose of the system is to provide higher-level knowledge representation, fusion, and reasoning in the maritime domain.

The PROGNOS includes a simulation which provides the ground truth information for the system. The simulation uses a given single entity Bayesian Network (we use this term to discriminate the SSBN from Multi Entity Bayesian Networks) in order for sampling data. The simulation generates 85000 persons, 10000 ships, and 1000 organization entities with various values of attributes. The data for these entities are stored in the relational database.

For the evaluation of the model, the training and test dataset was generated by the simulation. Using the basic structure learning for MEBN, the PROGNOS MTheory was derived as shown in Figure 2. In the model, a total of four MFrags were generated such as the default reference, org_members, person, and ship MFrag.

To generate a SSBN from this MTheory, we assume that we have one person, ship, and organization. They are related as ship_crews (Ship S, Person P) and org_members(Organization O, Person P). We queried the isShipOfInterest node with the several evidence nodes located in the leaf nodes. Figure 3 presents the result SSBN in which the nodes of the ship and person entity are connected each other.

To compare the accuracies of the results, we used the single entity Bayesian Network which was used for the sampling. Thus, the single network provided another query result with the same evidence. Figure 1 shows the Receiver Operating Characteristic (ROC) Curve which describes accuracy of the result of the learned MTheory and single entity Bayesian Network. The areas under curves are shown in Table 4.

Model	AUC
Learned MTheory	0.874479929
Single Entity Bayesian Network	0.87323784

Table 4. AUC of Learned MTheory and Single Entity Bayesian Network

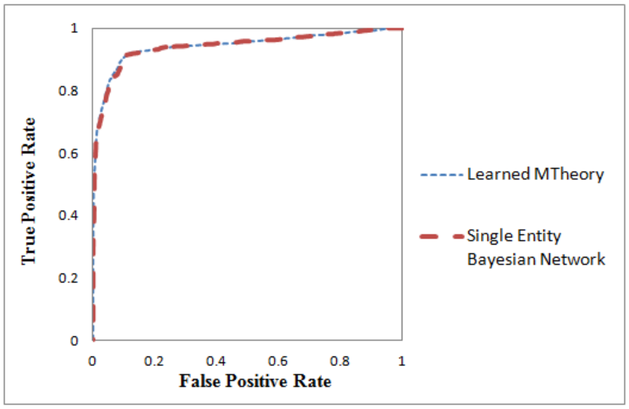


Figure 1. ROC of Learned MTheory and Single Entity Bayesian Network

As we can see from Figure 1 and Table 4, the results of accuracy of the learned MTheory and the single entity Bayesian Network are almost the same. This means that the learned MTheory well reflected the data of the relational database which was sampled using the single entity Bayesian Network.

In this paper, we only compared the learned MTheory to the true model which was the single entity Bayesian Network. This result proves that our approach reflects the true model correctly. However, the result of this paper is only the beginning and baseline for a full MEBN Learning method, because we didn’t address the aggregating influence problem which is the important issue in SRL models.

VI. DISCUSSION AND FUTURE WORK

Because of a flood of complex and huge data, efficient and accurate methods are needed for learning expressive models incorporating uncertainty. In this paper, we have introduced a learning approach for MEBN. As a bridge between MEBN and RM, MEBN-RM Model was introduced. For induction, the Basic Structure Learning for MEBN was suggested.

Recently, we are studying about a heuristic approach which called as the Framework of Function Searching for LPD (FFS-LPD) to address the aggregating influence problem. We plan to expand the learning algorithm in order to include continuous random variables.

References

[1] Hassan Khosravi, Bahareh Bina. A Survey on Statistical Relational Learning. In Proceedings of Canadian Conference on AI'2010. pp.256~268

[2] Getoor, L., Tasker, B.: Introduction to statistical relational learning. MIT Press, Cambridge, 2007

[3] Domingos, P., Richardson, M.: Markov logic: A unifying framework for statistical relational learning. In: Introduction to Statistical Relational Learning, ch. 12, pp. 339–367, 2007

[4] Nevile, J., Jensen, D.: Relational dependency networks. In: An Introduction to Statistical Relational Learning

[5] Kersting, K., de Raedt, L.: Bayesian logic programming: Theory and tool. In: Introduction to Statistical Relational Learning

[6] Oliver Schulte, Hassan Khosravi, Flavia Moser, and Martin Ester. Join bayes nets: A new type of bayes net for relational data. Technical Report 2008-17, Simon Fraser University, 2008. also in CS-Learning Preprint Archive.

[7] Laskey, K. B.,MEBN: A Language for First-Order Bayesian Knowledge Bases. Artificial Intelligence, 172(2-3), 2008

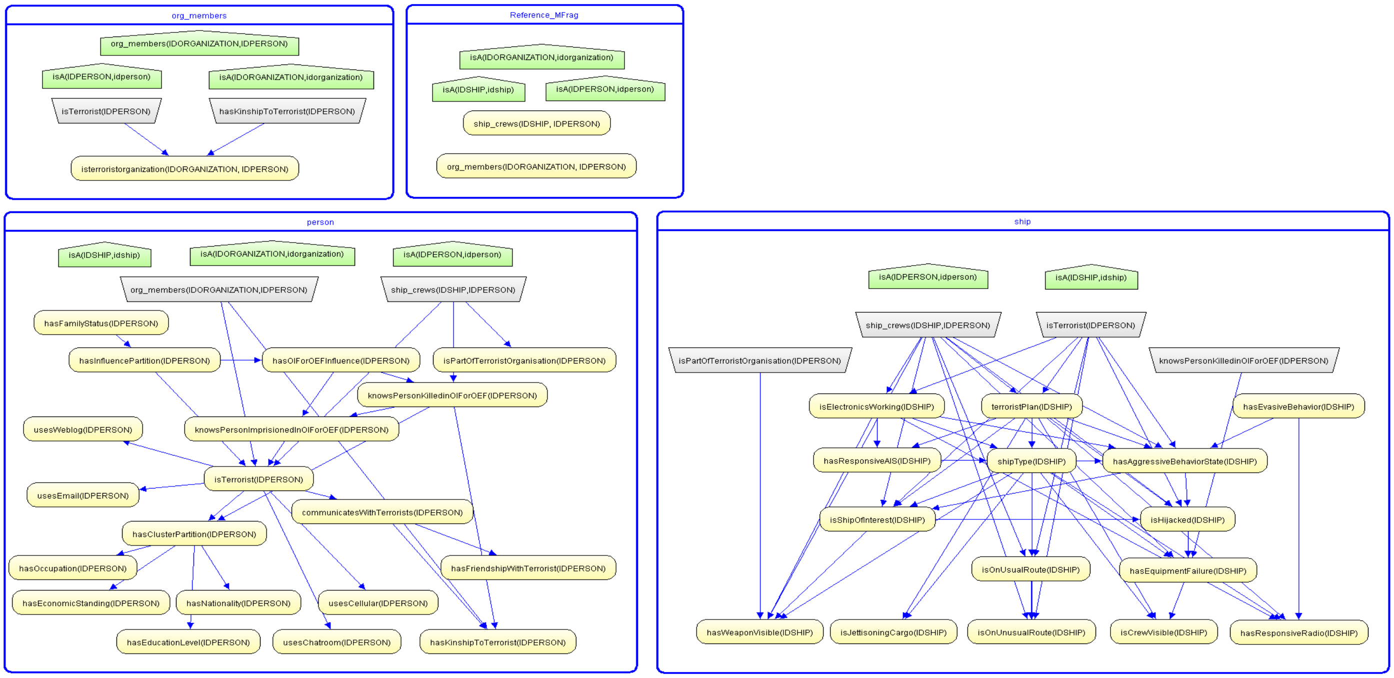
[8] Paulo C. G Costa, Bayesian Semantics for the Semantic Web. PhD Dissertation, George Mason University, July 2005. Brazilian Air Force.

[9] Rommel N. Carvalho, Probabilistic Ontology: Representation and Modeling Methodology, PhD Dissertation, George Mason University, July 2011.

[10] Codd, E.F. "A Relational Model of Data for Large Shared Data Banks". Communications of the ACM, 1970

[11] P.C.G. Costa, K.B. Laskey, and KC Chang, “PROGNOS: Applying Probabilistic Ontologies To Distributed Predictive Situation Assessment In Naval Operations.” Proceedings of the 14th Int. Command And Control Research and Technology Symposium, Washington, D.C., USA, 2009.

[12] R. N. Carvalho, P. C. G. Costa, K. B. Laskey, and K. Chang, “PROGNOS: predictive situational awareness with probabilistic ontologies,” in Proceedings of the 13th International Conference on Information Fusion, Edinburgh, UK, Jul. 2010.



org_members
org_members(IDORGANIZATION,IDPERSON)
isA(IDPERSON,idperson)
isA(IDORGANIZATION,idorganization)
isTerrorist(IDPERSON)
hasKinshipToTerrorist(IDPERSON)
isterroristorganization(IDORGANIZATION, IDPERSON)
Reference_MFrag
isA(IDORGANIZATION,idorganization)
isA(IDSHIP,idship)
isA(IDPERSON,idperson)
ship_crews(IDSHIP, IDPERSON)
org_members(IDORGANIZATION, IDPERSON)
person
isA(IDSHIP,idship)
isA(IDORGANIZATION,idorganization)
isA(IDPERSON,idperson)
org_members(IDORGANIZATION,IDPERSON)
ship_crews(IDSHIP,IDPERSON)
hasFamilyStatus(IDPERSON)
hasInfluencePartition(IDPERSON)
hasOIForOEFInfluence(IDPERSON)
isPartOfTerroristOrganisation(IDPERSON)
knowsPersonKilledinOIForOEF(IDPERSON)
usesWeblog(IDPERSON)
knowsPersonImprisionedInOIForOEF(IDPERSON)
isTerrorist(IDPERSON)
usesEmail(IDPERSON)
communicatesWithTerrorists(IDPERSON)
hasClusterPartition(IDPERSON)
hasOccupation(IDPERSON)
hasFriendshipWithTerrorist(IDPERSON)
hasEconomicStanding(IDPERSON)
hasNationality(IDPERSON)
usesCellular(IDPERSON)
hasEducationLevel(IDPERSON)
usesChatroom(IDPERSON)
hasKinshipToTerrorist(IDPERSON)
ship
isA(IDPERSON,idperson)
isA(IDSHIP,idship)
ship_crews(IDSHIP,IDPERSON)
isTerrorist(IDPERSON)
isPartOfTerroristOrganisation(IDPERSON)
knowsPersonKilledinOIForOEF(IDPERSON)
isElectronicsWorking(IDSHIP)
terroristPlan(IDSHIP)
hasEvasiveBehavior(IDSHIP)
hasResponsiveAIS(IDSHIP)
shipType(IDSHIP)
hasAggressiveBehaviorState(IDSHIP)
isShipOfInterest(IDSHIP)
isHijacked(IDSHIP)
isOnUsualRoute(IDSHIP)
hasEquipmentFailure(IDSHIP)
hasWeaponVisible(IDSHIP)
isJettisoningCargo(IDSHIP)
isOnUnusualRoute(IDSHIP)
isCrewVisible(IDSHIP)
hasResponsiveRadio(IDSHIP)

Figure 2. Generated PROGNOS MTheory

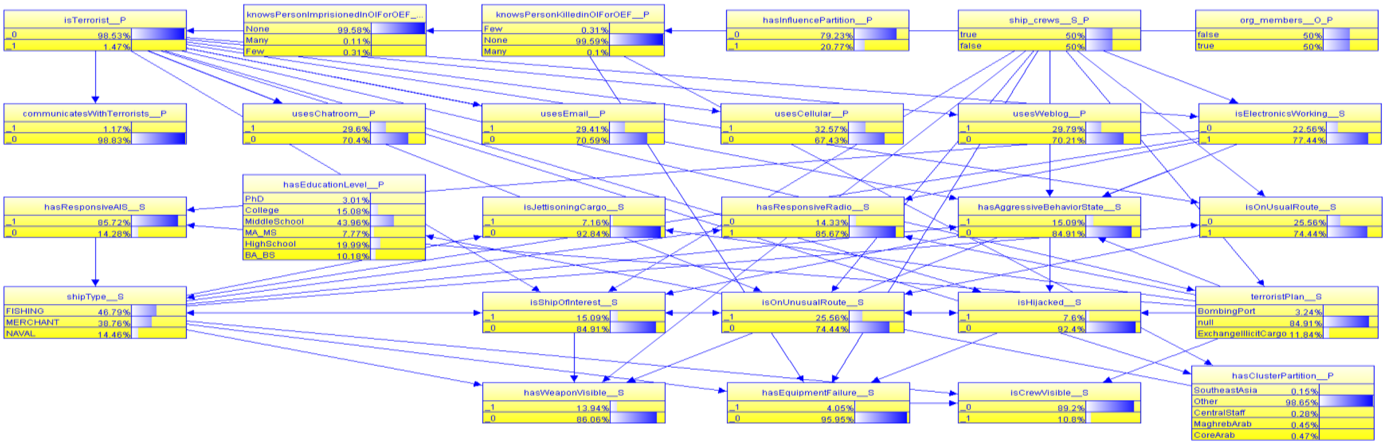


Figure 3. Generated SSBN of PROGNOS MTheory

Social Sifter: An Agent-Based Recommender System to Mine the Social Web

M. Omar Nachawati, Rasheed Rabbi, Genong (Eugene) Yu, Larry Kerschberg and Alexander Brodsky

Dept. of Computer Science
George Mason University
Fairfax, VA, USA
{mnachawa, rrabbi, gyu, kersch, brodsky} at gmu.edu

Abstract— With the recent growth of the Social Web, an emerging challenge is how we can integrate information from the heterogeneity of current Social Web sites to improve semantic access to the information and knowledge across the entire World Wide Web, the Web. Interoperability across the Social Web sites make the simplest of inferences based on data from different sites challenging. Even if such data were interoperable across multiple Social Web sites, the ability of meaningful inferences of a collective intelligence [1] system depends on both its ability to marshal such semantic data, as well as its ability to accurately understand and precisely respond to queries from its users. This paper presents the architecture for Social Sifter, an agent-based, collective intelligence system for assimilating information and knowledge across the Social Web. A health recommender system prototype was developed using the Social Sifter architecture, which recommends treatments, prevention advice, therapies for ailments, and doctors and hospitals based on shared experiences available on the Social Web.

Keywords: social semantic search; collective knowledge systems; recommender systems, OWL; RDF; SPARQL

I. INTRODUCTION

Since its inception, the World Wide Web has always overwhelmed users with its vast quantity of information. The advent of Social Webs, coined Web 2.0, has placed an additional burden on Web search engines. While the established algorithms that Web search engines employ are effective in surfacing the most popular results through hyperlink analysis, as demonstrated by the Hubs and Authorities algorithm [2] and the PageRank algorithm [3], those results are not necessarily relevant despite popularity and these algorithms have fallen short of solving the problem of information overload [1, 2, 3] on the World Wide Web.

The research into natural language understanding [4] attempts to close that gap. However the quality of machine generated semantics still pales in comparison to that of humans. This became a core challenge for the Semantic Web or Web 3.0, where information is made available in structured, machine-friendly formats allowing machines not only to sort and filter such data, but also to combine data from multiple Web sites in a meaningful way and allow inferences to be made upon that data. While semantic query languages, such as SPARQL, can provide a database-like interface to the World Wide Web, it is only as good as the quantity and quality of information that is made available in structured, machine readable formats, such as RDF and OWL .

Conventionally, finding answers to questions and learning from the knowledge mine existed on the Social Web has primarily been a manual process. It requires a lot of intelligence in sifting through the mountains of Social Web pages using only a keyword-based Web search engine, which is akin to a primitive pitch-fork in Semantic Web terms. More recently, however, Social Web sites have begun to embrace Semantic Web technologies such as RDF and OWL, and have been offering much more machine-friendly data, such as geo-tagged images on Flickr, Friend Of A Friend (FOAF) exports in FaceBook and hCalendar [7] tagged events on Blogger. Such developments have sparked the evolution of the Social Web into a collective knowledge system [1], where the contributions of the user community are aggregated and marshaled with knowledge from other heterogeneous sources (e.g., web pages, news and encyclopedia articles, and academic journals) in a synergy dubbed the *Social Semantic Web*.

While the Semantic Web focuses on data to enable interoperability among heterogeneous semi-structured web pages, the focus of the Social Semantic Web vision is to create a system of collective intelligence by improving the way people share and explore their own and others knowledge and experience [1]. Work on the Social Sifter promotes that grand vision and expands on the research done on the patented Knowledge Sifter architecture [7, 8, 9], as well as the Personal Health Explorer [11], undertaken at George Mason University. As a proof of concept, we have designed a social health knowledge and recommender system based on the Social Sifter platform that utilizes the Social Semantic Web to provide precise search results and recommendations.

The rest of this paper is organized as follows: section II discusses related work, section III describes the Social Sifter architecture and a brief description of the prototype system. Section IV highlights the experimental results, and Section V identifies the possible future work on the Social Sifter platform.

II. RELATED WORK

A. Knowledge Sifter and Personal Health Explorer

Semantic systems belong to a class of systems that make use of ontologies, context awareness and other semantic methods to make informed recommendations. Such research in semantic search at George Mason University began with WebSifter [8, 9, 10], an agent-based multi-criteria ranking system to select semantically meaningful Web pages from multiple search engines such as Google, Yahoo, etc. The work further led to a patent [8]. Knowledge Sifter (KS) [8] is motivated by WebSifter [7,8], but is augmented with the advanced use of semantic web ontologies, authoritative sources, and a service-oriented plug-and-play architecture. Knowledge Sifter is a scalable agent-based web services framework that is aimed to support i) ontology guided semantic searches, ii) refine searches based on relevant feedback, and iii) accessing heterogeneous data sources via agent-based knowledge services. Personal Health Explorer (PHE) is an enhancement of KS to perform semantic search in biomedical domain. PHE leverages additional features of a personal health graph to be identified, categorized, and reconstituted by providing links to the user to rate individual results and return to previous queries and update information through a semantically supported path.

KS and PHE are able to obtain more relevant search results than classic search engines; while the result is very general, it leaves room to make it more personalized. Both KS and PHE make multifaceted efforts towards realizing the Semantic Web vision, primarily focusing on the formal ontological sources. PHE provides facilities to include a user's Personal Health Record (PHR), which entails additional permission and access control which may be constrained by HIPAA regulations. Interestingly, both of these systems did not use the data available on the Social Web, namely Wikipedia, YouTube, Flickr, Facebook, LinkedIn, etc. This is where Social Sifter makes its contribution.

B. *BLISS and Cobot*

Other attempts to utilize Web 2.0 technology to enhance the quality and relevance of health recommendation systems include bookmarking, crowd sourcing, crowd tagging and harvesting user recommendations. The Biological Literature Social Ranking System (BLISS) is one such prototype system that allows users to bookmark and promote their recommendation to communities of special interest, facilitate the annotation and ranking by the community, and present the results to allow other users to get the recommendations based on community ranking [6]. The bookmarking approach is useful in establishing the authoritativeness of information over the long term because it uses social voting or ranking [5].

The Cobot system uses social conversation and social tagging (preference) to enhance the health recommendations. Three techniques are noteworthy: (1) user-initiative dialogue in capturing user's intent, (2) social tagging in establishing the authoritativeness of social information, and (3) case-based semantic reasoning in utilizing social knowledge for recommendation [5].

C. *Semantic Analytics on Social Networks*

A multi-step engineering process is described in [9] to utilize social knowledge. These steps are common procedure to across the initiatives to transform the social web information to semantic knowledge.

Social Sifter adheres to the underlying framework of Knowledge Sifter [9], the knowledge manipulation mechanism of PHE [10], and engineering process for semantic association of [11] to leverage an integrated semantic search engine and recommender system.

III. The Social Sifter Architecture

Social Sifter, an enhancement of the existing Knowledge Sifter (KS), is a collection of cooperating agents that are exposed through web services and exhibits a Service-Oriented Architecture (SOA)-based framework.

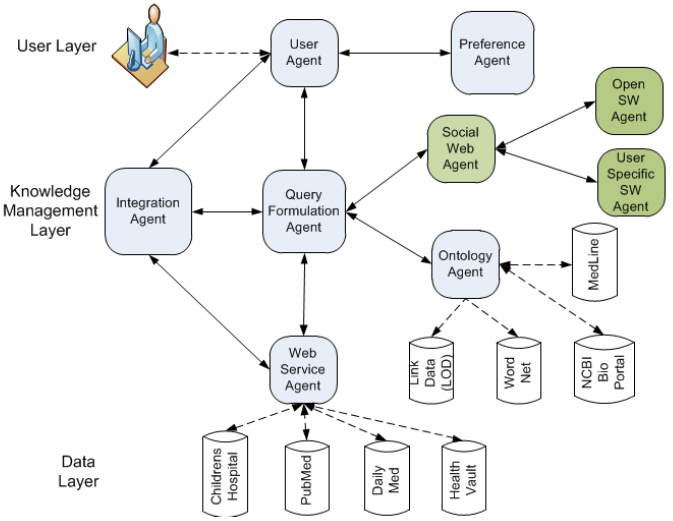


Figure 1. Social Sifter Architecture – Tiers and Components

Depending on the functionality, agents are allocated into three different architecture layers – i) the User Layer, ii) the Knowledge Management Layer, and iii) Data Layer. The User Layer consists of the User and Preferences agents, and manages all user interaction and data preferences. The Knowledge Management Layer handles the support for semantic search, access to data sources, and the ranking of search results using technologies like the Ontology, Social Web Crawling, Ranking, Query Formulation, and Web Services agents. The Data Layer consists of the data repositories that provide authoritative information and documents. The hierarchy of the architecture layers is already defined in KS; three additional agents were added, with an alteration of the underlying algorithm to perform the execution flow into the Social Sifter.

Social Web agent basically collaborates with following two agents to manipulate social web information.

Open SW agent performs open search within the blogs, related support groups etc.

User Specific SW agent identifies user social identities across the web and conducts Collaborative Filtering by processing social tags, user participation and responses available on the social webs.

IV. Health Recommender System

As a proof-of-concept, we are building a health recommender system using our Social Sifter architecture that provides health recommendations for any type of sickness, disease or disorder. The present system does not do any natural language processing on user queries, and therefore is limited as to what it can accept as a valid query. Currently, the system accepts a comma delimited list of words that relate to a specific ailment and returns a list of relevant descriptions of the ailment, therapy options, doctors, and treatment centers as collected from the Social Semantic Web from our knowledge Management Layer. We intend for future versions of the health recommender system to allow for unrestricted language queries by performing natural language processing to transform the unstructured query input into a more structured format, acceptable by the Social Sifter architecture.

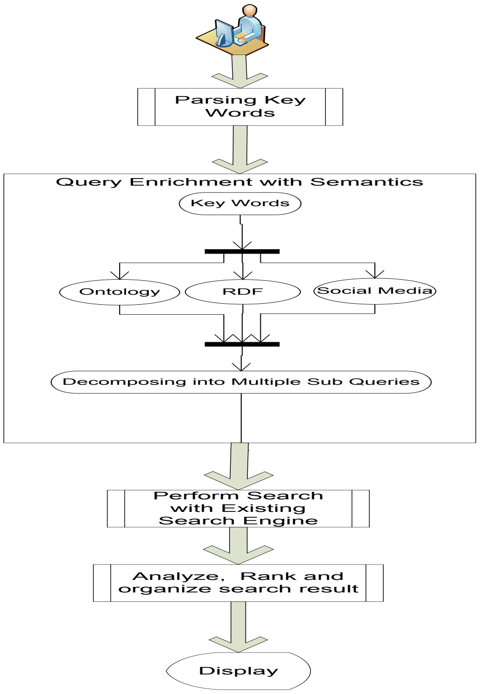

Parsing Key Words
Query Enrichment with Semantics
Key Words
Ontology
RDF
Social Media
Decomposing into Multiple Sub Queries
Perform Search with Existing Search Engine
Analyze, Rank and organize search result
Display

Figure 2. Social Sifter work flow diagram

A. *Scenario for Pancreatic Cancer*

Consider the case when a user is exploring recommendations for *pancreatic cancer*. According to the NIH, treatment options include surgery and biliary stents. The NIH also lists links to support groups, among which CancerCare.org features a social question-answer forum that is categorized by topic. Our inference agent for health recommendations takes advantage of this domain knowledge in attempting to provide better quality recommendations than what would be available from a general Web search engine. Let us walk through the steps of the health recommender system for this particular query.

Query Submission: User logs into the health recommender system website and enters the following query terms "pancreatic cancer."

Query String Preparation:

i) The ***User Agent*** parses the query string to identify key words.

ii) The ***Preference Agent*** collects context information, including the user's IP address, a query session identifier, and the best geographic location estimate available for that user. It tries to create a User Profile by indexing friendship and affiliation information to generate the user's Social Graph.

iii) User Agent passes the SPARQL query and the collected User Profile information to the Query Formulation Agent.

Query Refinement: The ***Query Formulation Agent*** then attempts to enrich the original SPARQL query by:

i) ***Semantic Query Decomposition***: It will generate multiple sub-queries that generalize and specialize the term *pancreatic cancer* based on the health-domain ontology from The National Center for Biomedical Ontologies (NCBO), a BioPortal and MedLine (Medical Literature Analysis and Retrieval System Online), which is a bibliographic database of life sciences and biomedical information.

ii) ***Marshalling***: selected data will be marshaled with the amassed folksonomy from the Social Web Agent. The inference engine will also generate queries based on the results of any cluster analysis from data crawled from the Social Web, which may pick up, for instance, other ailments that people have discussed together with *pancreatic cancer*.

iii) ***Ranking***: The end result of this meta-search is a weighted tree of sub-queries, where weights are assigned based, among other features, on the static nature of the sub-query generated (heuristically) as well as the importance of the source (back-reference analysis).

Post Query Processing: Once all sub-queries have been defined, the Web Service Agent passes them to the Data Layer, which accordingly runs the queries and itself ranks each result, based on many factors, including relevance (ontological), importance (back-reference based) and belief (Bayesian-based inference from Social Semantic Web).

Result Scrutinizing: The results are then returned to the Integration Agent, which combines different classes (based on the results from the classifier) of results based on a total ordering derived from the aggregated ontology, and back-reference analysis. The agent also performs a clustering analysis on the result set to further group the results and perform statistical calculations on the groups of results before passing them to the User Layer.

Result displaying: The User Layer then displays the grouped and ranked results according to the preferences selected by the user.

B. *Query life cycle for Pancreatic Cancer in Social sifter*

The life cycle of a query in Social Sifter, e.g., searching for "pancreatic cancer", is as follows: (1) a user allows access to his profile, (2) Sifter culls information from his social networks, (3) Sifter initiates targeted information harvesting, (4) Sifter conducts semantic inference and reasoning, and (5) Sifter presents socially- and semantically-renked results are to the user.

C. *Social Sifter Prototype*

The Social Sifter prototype has been implemented to use information retrievable from Facebook using Graph API in gathering the information about the users. In Facebook, each user can have feeds, likes, activities, interests, music, books, videos, events, groups, checkins, games, and his personal information, like hometown and related locations. These provide a very rich base for understanding the intension of a user when he is searching on the Web.

Social Sifter combined both semantic reasoning and social ranking to better understand user's intention and present the results to users, based on initial search keywords or phrases provided. The algorithm for the currently implemented search is described as follows.

(1) Login: User logs into his Facebook using OAuth authentication. The program gets the authorized token and uses it to access user's information with user's concurrence.

(2) Information Retrieval: The system retrieves the information about the user (Feeds, Likes, Activities, Interests, Music, Books, Photos, Videos etc.) and uses them in supporting the targeted harvesting of information and formulating the social ranking of results in categories.

(3) Social ranking – A simple algorithm is used to calculate the social weights of the harvested information in each category. The algorithm is basically counting the occurrences of keywords or phrases in each category.

(4) Social context – The user's background information is used in refining the search results or filtering the results. One specific example is the location information. The home location of the person is generally used to limit the places to be searched and returned.

(5) Semantic result presentation – The results are presented to users in groups: people, groups, events, places, events, pages, or posts. The current implementation is limited to use the categories or semantics of Facebook. The actions in Facebook link objects and people. They are the bases for our search engine in weighing the harvesting strategies. They are also important in ranking the results and the categories when presenting the search results to users. The current implementation used the same social ranking strategy described in (3).

D. Proactive Social Search

The existing Facebook semantics do not capture the semantic of health queries. For health problems, users may be interested in finding out the cure of certain diseases, which is not captured by the current set of actions available in Facebook. Customized actions can be implemented using the Facebook Open Graph, but it is beyond the scope of this paper.

V. EXPERMENTAL FINDINGS

The Social Sifter prototype has been implemented. The Facebook Graph API was used as the basis for harvesting social network information about the user. Social information was used in two aspects – understanding the user's intention (context) and ranking results (social semantic ranking). The two aspects showed improved search results. For example, the searching case using phrase – "pancreatic cancer" can be compared using three different engines – Google, Facebook, and Social Sifter. Social Sifter provided integrated results and used social ranking to rearrange the categories depending on users profile information. Location is determined based on user provided current living locations. More testing is being carried out to determine metrics to assess the quality of social semantic search recommendations.

VI. CONCLUSIONS

Social semantic search is an integration of social networks and semantic search. Semantic search provides rich means in enhancing search, especially the user's intent and semantic reasoning. Social search involves people and links to their social graphs. In this paper, a prototype social semantic search engine, Social Sifter, has been presented. The lessons learned from the implementation showed two areas for improving search accuracy: social contextual information (user intent understanding) and social semantic ranking (results relevance).

The current implemented prototype system is limited in the use of the semantic reasoning. The crawling of data should be expanded to other social media and social networks. Integration of these results into a standard semantic data store is necessary to realize the power of semantic reasoning. Further study directions are: (1) to integrate mature ontologies, (2) to define customized actions to demonstrate the approach in health domain, and (3) to use the reasoning power of semantics.

REFERENCES

[1] T. Gruber, "Collective knowledge systems: Where the Social Web meets the Semantic Web," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 6, no. 1, pp. 4–13, Feb. 2008.

[2] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *J. ACM*, vol. 46, no. 5, pp. 604–632, Sep. 1999.

[3] L. Page, S. Brin, R. Motwani, and T. Winograd, *The PageRank Citation Ranking: Bringing Order to the Web*. 1999.

[4] A. Ntoulas, G. Chao, and J. Cho, "The infocious web search engine: improving web searching through linguistic analysis," in *Special interest tracks and posters of the 14th international conference on World Wide Web*, New York, NY, USA, 2005, pp. 840–849.

[5] J. M. Gomez, G. Alor-Hernandez, R. Posada-Gomez, M. A. Abud-Figueroa, and A. Garcia-Crespo, "SITIO: A Social Semantic Recommendation Platform," in 17th International Conference on Electronics, Communications and Computers, 2007. CONIELECOMP '07, 2007, p. 29–29

[6] "hCalendar 1.0 · Microformats Wiki." [Online]. Available: http://microformats.org/wiki/hcalendar. [Accessed: 15-Apr-2012].

[7] L. Kerschberg, W. Kim, and A. Scime, "WebSifter II: A Personalizable Meta-Search Agent Based on Weighted Semantic Taxonomy Tree," in *International Conference on Internet Computing*, Las Vegas, NV, 2001

[8] L. Kerschberg, W. Kim, and A. Scime, "Personalizable semantic taxonomy-based search agent," U.S. Patent 7117207Oct-2006

[9] L. Kerschberg, H. Jeong, Y. Song, and W. Kim, "A Case-Based Framework for Collaborative Semantic Search in Knowledge Sifter," *Case-Based Reasoning Research and Development*, vol. 4626/2007, pp. 16–30, 2007.

[10] T. G. Morrell and L. Kerschberg, "Personal Health Explorer: A Semantic Health Recommendation System," workshop on Data-Driven Decision Support and Guidance System (DGSS), 28th IEEE International Conference on Data Engineering, Arlington, VA April 1, 2012.

[11] Boanerges Aleman-Meza, Meenakshi Nagarajan, Cartic Ramakrishnan, Li Ding, Pranam Kolari, Amit P. Sheth, I. Budak Arpinar, Anupam Joshi, Tim Finin. Semantic Analytics on Social Networks: Experiences in Addressing the Problem of Conflict of Interest Detection. *WWW 2006,* May 23–26, 2006, Edinburgh, Scotland. ACM 1-59593-323-9/06/0005.