12th ICCRTS

"Adapting C2 to the 21st Century"

Cross-Domain Ontology Resolution in Net-Centric Command and Control

Track 1: C2 Concepts, Theory, and Policy

Authors: Dale E. Lichtblau, Steven P. Wartik

Point of Contact: Steven P. Wartik

Institute for Defense Analyses

4850 Mark Center Drive
Alexandria, VA 22311-1882

703-845-6646

swartik@ida.org, del@ida.org

Cross-Domain Ontology Resolution In Net-Centric Command and Control

Abstract

Experts have argued that the success of Net-Centric Operations and Warfare (NCOW) depends upon the ability of net-centric environment (NCE) users—both human and automated—to readily discover useful information and Web-based services. Effective discovery requires, in turn, effective meta-data "tagging." It was argued that no single, overarching classification scheme is adequate to provide the semantic support required for the successful deployment of such core enterprise services as discovery, collaboration, mediation, and storage. What was needed was a way to support *multiple taxonomies* with automatic taxonomy evolution using machine learning and intelligent agent technology. This paper analyzes the underlying reasons for this claim and shows that what is really needed is a way to allow multiple *ontologies* (along with their taxonomic correlates) with cross-domain (i.e., inter-ontology) resolution (translation) to coexist in a net-centric environment. After surveying some apparent theoretical limits to ontology commensurability, it describes a conceptual framework that is sufficient to enable information and Web-services interoperability for command and control in an NCE. It compares this framework to current semantic web approaches to show what the framework contributes.

1. Introduction

It should be apparent that the success of Net-Centric Operations and Warfare (NCOW) depends upon the ability of net-centric environment (NCE) users—both human and automated—to readily discover and then use relevant information and Web-based services. Minimally, discovery requires the effective "tagging" of the information and Web services that are being offered for sharing. Tagging is just another name for cataloging or classifying information or Web services to allow them to be found more easily. An earlier paper [1] argued that no single, overarching classification scheme was likely to be fully adequate for the effective discovery of core enterprise services or information. What was needed, it suggested, was a way to support *multiple classification schemes*, ideally with the suitability of each classification scheme to be evolved automatically by intelligent software agents. This paper recaps the underlying reasons for the need for multiple classification schemes—multi-faceted classification—and then argues that what is really needed is a way to allow multiple *ontologies* (along with their taxonomic correlates) with cross-domain (i.e., inter-ontology) resolution (translation) to coexist in a net-centric environment. After surveying some apparent theoretical limits to ontologic commensurability, this paper describe a conceptual framework that is sufficient to enable information and Web-services interoperability for command and control in an NCE. It compares this framework to current semantic web approaches to show what the framework contributes.

2. The Problem

Why is multi-faceted classification needed? The simplest and most compelling reason is that the Department of Defense (DoD) will never be able to enforce the use throughout the enterprise of an information and services "tagging" *standard* (i.e., a *single* classification scheme)—even if a reasonably good standard could be devised. The argument is straightforward. The military Services, the Office of the Secretary of Defense, the supporting and unified combatant commands—i.e., the components of the DoD enterprise—all have different responsibilities and different ways (functions) of meeting those responsibilities. Even when a function is common among the different uniformed Services, the particular way in which the function is performed usually differs between DoD components. And there are simply cultural differences that will never be fully resolved. (For reasons which are sketched below, it is not obvious that a *single* classification scheme sufficient to encompass *all* of the multifarious kinds of information and information services regularly used throughout DoD is even possible. The necessary conceptual tools are just not available.)

First some theoretical background. What exactly is the problem this paper is trying to solve? President Lincoln used to ask, "If you call a tail a leg, how many legs does a dog have? Four! Calling a tail a leg doesn't make it a leg." But a dog's leg can be called a "leg." It might also be called a "canine locomotive appendage"—or simply a CLA, in DoD acronymese. If the Air Force calls a dog's leg a "leg," and the Army calls it a "CLA," and each Service says they have so many legs and CLAs stockpiled, how can one assess the Army's capabilities vis-à-vis those of the Air Force without knowing that the two concepts are really equivalent? Simply put, how can one determine (without asking or consulting a Service-specific lexicon) that the Air Force's "leg" and the Army's "CLA" refer to the same thing? More generally, how can one determine the semantic equivalence of two syntactically distinct concepts?[1]

Conversely, how can one determine the semantic non-equivalence of two syntactically equivalent concepts (e.g., "tank" (*qua* container) and "tank" (*qua* armored attack vehicle))?[2] In terms of net-centric command and control, how are authorized users of the Global Information Grid (GIG) to "discover" the number of tanks (armored vehicles) allocated to the 1st Infantry Division and on-hand and fully operational at a certain time?

Finally, what should be done with a concept that has no analogue? The US Army has three concepts of gender: male, female, and unknown. The Marine Corps has four: male, female, other, and unknown. "Gender" is clearly a concept that exists in both the Army and the Marine Corps. But how should the Army model the gender of an individual that the Marine Corps designates as "other"?

How will the Global Directory Service (GDS)[3] enable a GIG user, with the proper security credentials and "need to know," to find such information? Is the "meta-data" prescribed to be specified and published in a meta-data registry or catalog per the DoD Net-

---

1 The two *names* of the concepts in our example ("leg" and "CLA") differ *syntactically* (i.e., orthographically). By hypothesis, they are equivalent *semantically* (i.e., they *refer* to the same thing and, when properly understood, have roughly the same *connotation*).

2 This is not a vacuous intellectual exercise. A few years ago the US Air Force, Navy, and Marines each had different definitions—with significant implications in terms of readiness reporting—for "available full-up round" of the AIM-9 missile. The definitional differences turned on the Services' assignment of different values to "time to assemble, time to bench-check as serviceable, and time to make ready for immediate load-out."

3 The GDS is one of the Defense Information Systems Agency's (DISA) Net-Centric Enterprise Services (Core Services). It purportedly "[p]rovides an enterprise-wide service for [the] identification [of] and other pertinent information about people, objects and resources, and makes it accessible from any place at any time (http://www.disa.mil/main/prodsol/1_enterprise.html)."

Centric Data Strategy[4] really sufficient to locate needed information (or relevant services, as the case may be)?

As argued in [1], a Google™-like search is often ineffective, especially if the search is to be automated. A recent Google™ search for "tank" returned 122,000,000 results. Searching for "tank" and "military," Google™ returns about 8,420,000 results. To make Google™ and similar Internet search engines more useful, humans intuitively refine the search process by including terms that confine the search to the appropriate domain. The online version of WordNet®—which, by the way, gives five definitions for the noun form of "tank"[5]—has *military* as one of five *domain categories* applicable to "tank" when defined as "an enclosed armored military vehicle; has a cannon and moves on caterpillar treads." A *domain category* in WordNet® is a *topical classification*. In other words, WordNet® makes explicit the mechanism humans rely upon intuitively in disambiguating terms and fixing their meaning. Ultimately, it's the linguistic context in which a term is used that determines its meaning, or, as Wittgenstein famously put it, "the meaning of a word is its use in the language."[6] The meaning of a term emerges from and is dependent upon the way it is used *in* the language and *by* a community of language users. This reference to and need for *language user communities* is reflected in DoD's Communities of Interest (COI) approach to implementing its net-centric data strategy.[7] Indeed, one role of a DoD COI is to "define COI-specific vocabularies and taxonomies," vocabularies "to improve data exchange within COI and among COIs" and taxonomies "to improve precision discovery." Both the theoretical and practical problems inherent in this latter are the focus of this paper.

3. Theoretical Issues in Information Discovery

First, a few (stipulative) definitions. The terms "ontologies," "taxonomies," "vocabularies," "meta-data," *et cetera* have become the DoD buzz-words *du jour*. They are not always used consistently or coherently. In philosophy, ontology is the study of being, *per se*.[8] In computer and information science (and in the DoD), *an* ontology is an account of the things (objects, entities) of interest (in a domain of interest). The account specifies the things (that exist or comprise the domain), as well as their attributes (properties) and the

4 http://www.dod.mil/cio-nii/docs/Net-Centric-Data-Strategy-2003-05-092.pdf

5 http://wordnet.princeton.edu/perl/webwn

6 Wittgenstein, Ludwig, *Philosophical Investigations*, §43. (This view is not—as with any philosophical issue—without considerable controversy. It nicely sloganizes the view that for any formal data modeling effort to be useful for automatic *knowledge* discovery, contextual features—entity attributes, their domains, and *importantly their interrelationships*—must be well formed and easily "navigated.")

7 See, for example, Todd, Michael, "Implementing the Net Centric Data Strategy using Communities of Interest," October 20, 2005, http://colab.cim3.net/file/work/caf/resources/10_20_05/DRM_COI_Net_CentricDataStrategy_Todd_2005_10_20.ppt#612,1,Transforming the Way the DoD Manages Data Implementing the Net Centric Data Strategy using Communities of Interest, for one of many discussions of the critical role that COIs are intended to play in the net-centricity information environment.

8 An excellent discussion of the differences (and similarities) between philosophical ontology and computer science ontology can be found in Smith, Barry, "Ontology and Information Systems," http://ontology.buffalo.edu/ontology(PIC).pdf.

relations[9] that obtain between the things of the domain. It's typical to think of ontologies as consisting of both individual existing things (individuals or instances) and the concepts that characterize them. *Person* is a concept; Dave Alberts is an individual person, an instance of the concept *person*, as it were. A taxonomy is that which you get by selecting all of the concepts of an ontology that are related by (something like) a "is-a-(kind of)" relation and then documenting the results in a way that preserves the "is-a" relation of the ontology. A taxonomy is just a handy "is-a"-based classification. A *good* taxonomy will also be one in which all of the *X*'s that "are-a-kind-of" *Y* are mutually exclusive (the sub-domain *X*'s do not overlap) and collectively exhaustive (there aren't any *Z*'s to be found (that "are-a-kind-of" *Y*) that aren't among the *X*'s). Ideally, every information system ontology will exhibit a good taxonomy. A vocabulary is an account of both the *language* used to give an account of an ontology and its imbedded taxonomy (for example, the terms "concept," "entity," "attribute," "relation") as well as of the ontology itself (that is, of the things that actually comprise the domain of interest). In a genuine sense, a vocabulary is meta-data.[10] It describes what language (data) is *used* within a domain of interest by *mentioning* that language. It's basically a more human-accessible form of what we now call an ontology, with or without a taxonomy. The central issue of this paper is how to effect ontology resolution when searching for information (or services) outside of one's COI. By "ontology resolution" we mean the (partial) alignment of two distinct ontologies in order to facilitate information (or services) discovery within a COI other than one's own. Each ontology provides the context necessary to verify the presence or absence of the sought for information. One can infer safely by analogy that if "tank" *is-a* "armored combat vehicle" within one's own COI, say $COI_{A,}$ and *a-kind-of* "military weapon" in another's COI ($COI_{B)}$, then the two concepts are likely to be the same; however, if "tank" appears as *a-kind-of* "freight car" in still another COI ($COI_{C)}$, then the two concepts are most likely distinct. This simple example is illustrated in Figure 1.

Figure 1 depicts a fragment of three COI ontologies. The concept "tank" is syntactically equivalent in all three. But on the assumption that the two relations "is-a" and "a-kind-of" are semantically equivalent (or nearly so) and that the concept "armored combat vehicle" is qualitatively closer (in the ontological space that subsumes all three ontologies) to "military vehicle" than it is to "freight car," the tanks of COIs *A* and *B* are "equivalent" but the tanks of COIs *A* and *C* (and of *B* and *C*) are "distinct." It is just this kind of reasoning that has to become automated if a net-centric GIG is to fully live up to our expectations.

While this paper has been framing the discussion in terms of information (and information services) discovery, the information exchange issue faces the same problem. For ex-

[9] A (dyadic) *relation* (e.g., "X is the daughter of Y") can be also viewed as an *attribute* ("is the daughter of Y") of X, reducing the definitional machinery to just things and their attributes.

[10] "Data about data" is the customary formula. "Meta" is Greek for "after." (Metaphysics came by its name because the "book" in which Aristotle wrote of things metaphysical (and ontological) was physically placed after his "book" on physics in a collected edition.) In logic (or in any formal discipline) it's always important to bear in mind the distinction between the object language (the language being reasoned about) and the meta-language, the language in which the reasoning (about the object language) is carried out. The Knight in Lewis Carroll's *Alice Through the Looking Glass*, used Alice's ignorance of this "use-mention" distinction to thoroughly befuddle poor Alice. Remember too that meta-data is also data and can be talked about using meta-meta-data (meta-(meta-data)), *ad infinitum*.

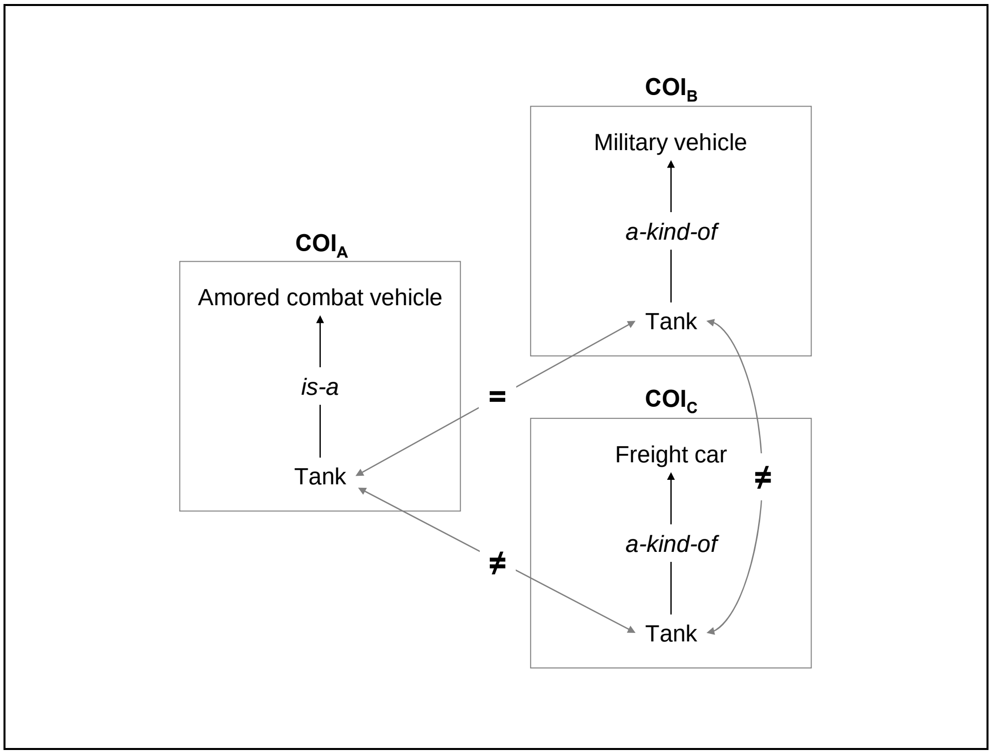

Figure 1. Discovery via Ontology Resolution

ample, the Joint Command, Control, and Consultative Information Exchange Data Model (JC3IEDM) [7] is a key element of the Multilateral Interoperability Programme (MIP), a 26 nation effort under the aegis of North Atlantic Treaty Organization (NATO), with the objective of achieving international interoperability of command and control information systems.[11] The JC3IEDM is "intended to represent the core of the data identified for exchange across multiple functional areas…. [I]t lays down a common approach to describing the information to be exchanged in a command and control (C2) environment." DoD's Universal Joint Task List (UJTL) [8] is a warfighting mission area-focused "menu of tasks in a command language, which serve[s] as the foundation for capabilities-based planning across the range of military operations."[12]

As an example of the reasoning pattern, the term "terrain" is examined for the challenges involved. Both the JC3IEDM and the UJTL have a concept of terrain as illustrated in Figure 2. Between the seven types of terrain in the UJTL and the five specific types available in the JC3IEDM, only one ("mountainous") is common (on a syntactic basis). (The definitions provided in the JC3IEDM documentation add little but obvious synonymity by way of explication. "Flat," for example, is defined as "terrain…characterized as broadly level.") Are the two concepts—*UJTL terrain* and *JC3IEDM terrain*—equivalent, even roughly? How would one decide? How can one represent terrain information, generated on the basis of the UJTL ontology, for exchange via the JC3IEDM to a coalition partner? And could this ever be done automatically? Actually, the situation is a

[11] See http://www.mip-site.org/010_Public_Home_News.htm.

[12] CJCSM 3500.04D, August 2005, directive current as of 17 August 2006, http://www.dtic.mil/cjcs_directives/cdata/unlimit/m350004.pdf.

little more complicated. The UJTL provides additional and rather robust ways to characterize terrain, and terrain is only one of four major facets used to classify land. (The other three are geological features; man-made terrain features, including urbanization; and landlocked waters.) The UJTL descriptors for terrain listed in Figure 2 are only "general characteristics of land areas." Additional features include terrain relief, terrain elevation, terrain slope, terrain firmness, terrain traction, vegetation, and terrain relief features. These features (attributes), along with their allowable values, need to be compared with the analogous geographic features (and accompanying codes) in the JC3IEDM. In addition to the already mentioned geographic-feature-terrain-code, the JC3IEDM can represent a geographic feature in term of bottom-hardness, solid-surface-composition, status-category (liquid-body, liquid-surface, solid-surface), status-surface-recirculation-indicator (will or will not recirculate as a result of rotor downwash), surface-category, type-category, and type-subcategory. These attributes are defined generically as "terrain characteristics to which military significance is attached."

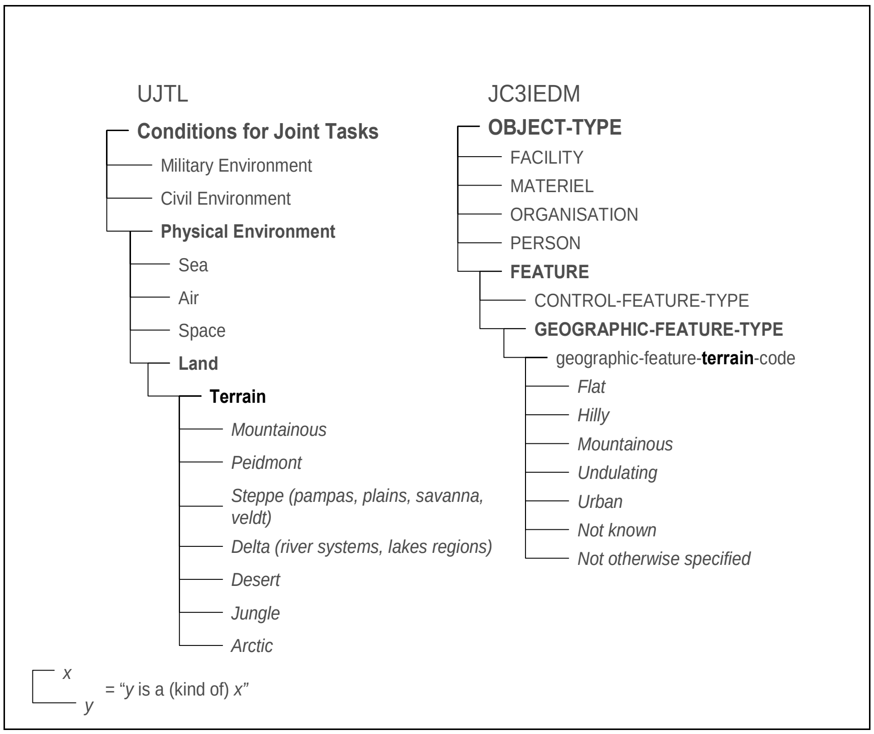

Figure 2. Terrain in the UJTL and JC3IEDM Ontologies

With this brief survey of some of the fundamental issues in terms of ontology resolution that underlie—and threaten to undermine—effective discovery and information exchange in a net-centric environment, it is time to look at possible solutions.

4. Ontology Resolution: One Approach

One such possible solution to what we are calling ontology resolution is a tool developed by John Li of Teknowledge Corporation. His Lexicon-based Ontology Mapping tool—LOM for short—matches terms between source and target ontologies and assigns an alignment confidence rating, a number between 0 and 1, to each putative match [3]. Li's work uses the Suggested Upper Merged Ontology (SUMO) [4] and its descendant, Mid-Level Ontology (MILO) [5], as its context bases. It also uses WordNet® [6] for synonym discovery and comparison. Using these resources LOM has achieved precision and recall rates of 71% and 57%, respectively.[13] This kind of performance makes it well suited to what Li calls "first-cut" comparisons, part of a scenario in which LOM identifies likely matches for subsequent human verification.

LOM demonstrates the feasibility of automatically resolving term equivalence across ontologies. But with only a 71% precision rate, this technology is not (yet) practical for daily application in an environment as large as the NCE. A human must filter out the irrelevant 29% of retrieved records, a significant effort.

In what follows, this paper extends Li's approach by considering only those ontologies that are related to command and control. It does so by discarding SUMO and MILO as context, using instead the ontology that informs the JC3IEDM. A domain-specific ontology seems likely to yield better performance in ontology resolution than the general-purpose ontologies of SUMO and MILO, at least with respect to the domain of primary interest, C2.

The remainder of this paper is organized as follows. Section 5 presents an overview of Li's approach. Section 6 presents this paper's extensions to Li's work. Section 7 covers the framework in which the approach might be used. Section 8 summarizes findings.

5. An Overview of Li's Approach

Li, like most researchers in the area, assumes that information is captured in an ontology. He further assumes an ontology is represented in a formal language that gives precise meaning to kinds of terms and the relationships between them. As is appropriate to semantic web and NCE research, he uses ontologies formally represented in the Web Ontology Language (OWL) [2]. This provides, among other things:

- A hierarchical class model.
- Properties of and relations between classes.
- Individuals, each of which is a member of one or more classes.

LOM takes as input a source ontology and a target ontology, both represented in OWL. The tool's output is a list of the "terms" of the source ontology that "match" terms of the target. A "term" is, roughly, the name of a class, a property or relation, or an individual.

LOM uses a four-step algorithm:

- ***Step 1: Match whole terms.*** In the first step LOM looks for matching names. If both ontologies contain a term with the same name, these terms are considered to match. The whole-term-matching step is quite literal in its treatment of terms: "ObjectItem" and "Object-Item" do not match. (It does, however, ignore case distinctions. "Object-Item" and "object-item" match.)
- ***Step 2: Match word constituents.*** LOM next divides "compounded" terms into their components by considering capitalization, concatenation, and punctuation.

[13] Precision is the percentage of retrieved records that are relevant. Recall is the percentage of relevant records that are retrieved. For an excellent overview of precision and recall, see http://www.hsl.creighton.edu/hsl/Searching/Recall-Precision.html.

Thus "ObjectItem" and "Object-Item" are treated as "Object Item" and would match. LOM also uses stemming and permutations, and uses a stop list to filter out prepositions and similar, usually irrelevant, words. Therefore, "BirthDate" and "Date-of-Birth" match.

- ***Step 3: Match synonym sets.*** LOM then uses WordNet® to generate synonym sets for each term of each pair of terms of the input ontologies provisionally considered to be equivalent. LOM then performs a term-by-term comparison of these synonym sets. If the sets are equivalent, then the two terms are equivalent. For instance, LOM determines that "capability" and "ability" match, as do "motor vehicle" and "automobile."
- ***Step 4: Match types.*** In its final step, LOM uses a predefined set of mappings from WordNet® words mapped to SUMO/MILO terms. LOM finds SUMO/MILO terms for words in the synonym sets identified in step 3, then sees if each source ontology term has a counterpart term in the target ontology.

Each step of the algorithm assigns a confidence-factor score to a pair of matched terms. The first step is a binary 0 or 1 score: either two (whole) terms match or they don't. The remaining steps assign a number between 0 and 1 based on the ratio between the number of matched words and the average number of words being considered. Furthermore, each step uses an empirically derived weighting factor that reflects the decreasing confidence in matches for the successive steps. The final score for a matching pair at each step is the maximum value of its score from an individual step times the step's weighting factor.

The overall result of the algorithm is therefore a sequence of candidate source/target term pair matches, ordered by the alignment confidence computed for each pair. An implementation of the algorithm would also include a minimum confidence value to filter out pairs whose alignment is calculated as highly unlikely.

An ontology developer/maintainer could use these results to add equivalence definitions to his ontology. He would first want to verify each match by examining supplied textual definitions and term context. This would help him find and eliminate terms with variant meanings (e.g., tank, which would satisfy whole-term matching). It would also let him ascertain whether lower-scoring results are in fact valid "matches."

6. A JC3IEDM-Based Extension to Li's Approach

Li's algorithm described in Section 5 is, by design, a general-purpose approach. As such it is an important tool for comparing ontologies from arbitrary domains. However, the authors of this paper believe that many ontology comparisons will be between ontologies from a common domain. As an example, consider ontologies that model the JC3IEDM and the UJTL. Both are predominantly C2-oriented; they share a concern for the same domain, namely C2.

The authors' goal is to dramatically improve accuracy in ontology resolution, at this point between different sub-domains of a single major domain (i.e., C2). This paper therefore proposes an extension of Li's algorithm that incorporates both domain-specific context and technology-specific properties. The domain-specific approach improves on the use of SUMO and MILO by providing analysis of C2-related terms. The technology-specific properties capitalize on features of OWL. The rest of this section presents the proposed algorithm. The relationship to Li's algorithm is maintained by using the same step numbers where possible, or by showing intermediate steps using alphabetic suffixes.

- ***Step 1: Match whole terms***. This step is virtually identical to LOM's first step, though as discussed below the ranking approach differs. The algorithm compares pairs of names of conceptually similar elements: class names to class names, property names to property names, and individual names to individual names. In some OWL dialects an individual can also be a class. Such an element is treated as a candidate for comparison in two element categories.
- ***Step 1a: Match terms in element definitions.*** An OWL element typically has a comment that is intended to convey the element's meaning in natural language. In the authors' experience these comments typically contain a lot of domain-specific jargon and as such could be used in the element comparison process. A confidence factor is only computed if both elements have such a comment. Stop lists are also employed to filter out common words such as prepositions, and a stemming algorithm is used to reduce words to their roots (eliminating plural and infinitive forms).
- ***Step 2: Match word constituents.*** This step is identical to LOM's second step. As in step 1, it is applied to element names.
- ***Step 3: Match synonym sets.*** This step is similar to LOM's third step. However, it uses the results of step 2 to prioritize synonyms. If WordNet® generates a set of synonyms for a given term, it looks for set members that include terms from the element's definition. This increases the confidence that the intended meaning has been correctly picked.
- ***Step 3a: Perform property comparisons.*** The source and target ontologies contain information about classes/individuals and their properties. This step examines the classes and individuals identified as possibly related and examine their properties. If matches in some of their properties have been identified, this step uses that information. For instance, if classes C1 and C2 both have a "has-parent" property, it examines the cardinality of the property in each class. Suppose one is functional and the other requires exactly two instances (Figure 3). This would indicate that the first denotes a strict hierarchy—a taxonomy, perhaps—whereas the other is a network, possibly describing a genealogy. This should weaken confidence in the possible equivalence of C1 and C2. (It would certainly change confidence in the near equivalence of the two "has-parent" properties.) These kinds of tests may be applied to other aspects of properties, such as whether the properties are functional, reflexive, symmetric, or transitive. Property equivalence can also be tested by simply comparing their respective extensions.

  Of course, the absence of corresponding matching properties does not guarantee the non-equivalence of two classes. Two classes with different but not inconsistent properties may be equivalent, with the different properties simply reflecting different views or perspectives of the same concept by different COIs. For instance, military services share concepts but describe them using different terminology.
- ***Step 3b: Classify the ontologies.*** A set of likely equivalences between elements of the two ontologies has now been populated. This information can be employed by

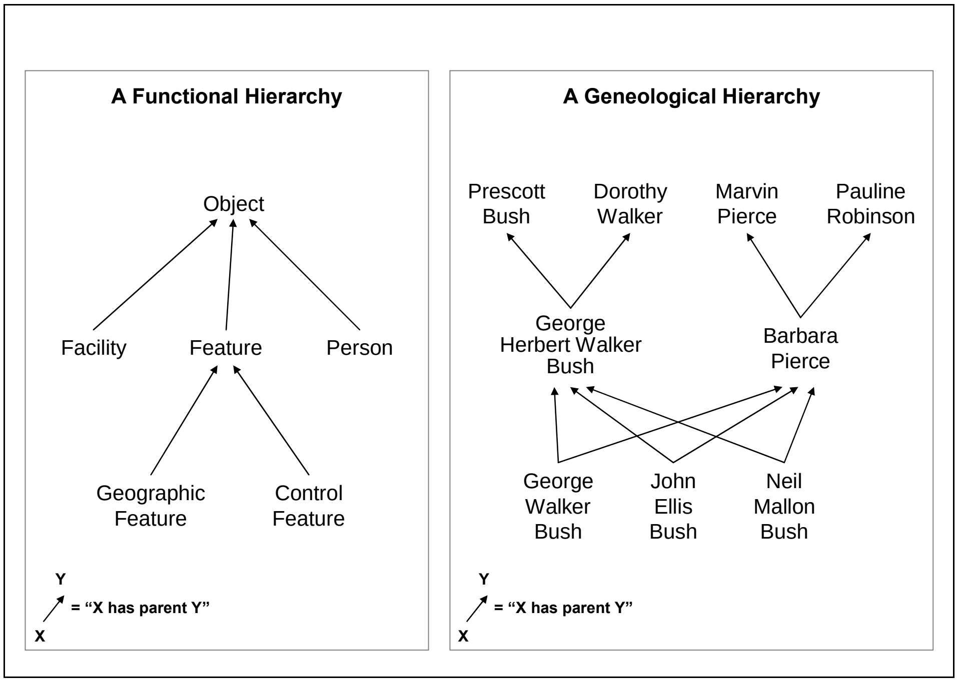

Figure 3. Property Comparisons

a description logic reasoner to check the classes for consistency [9]. Input to the reasoner includes the two ontologies, along with the equivalence statements posited to be true. The reasoner will report whether these statements lead to any contradictions, that is, classes which can have no members. Each contradiction indicates a mistaken assumption and indicates that an equivalence does not in fact exist.

A reasoner can also determine all of the classes of which an individual is a member. This property is exploited to test conceptual similarity confidence. Suppose we have individuals I1 and I2 in both source and target ontologies. If the reasoner identifies I1 as belonging to a class of which I2 cannot be a member, then I1 and I2 cannot denote the same concept. (That I1 cannot be identified as belonging to any class in the target ontology is inconclusive. It only means that not enough information is available to fully categorize I1.)

- ***Step 4: Map terms to JC3IEDM terms.*** This step corresponds to LOM's fourth step, but the JC3IEDM ontology is used instead of SUMO and MILO. The step attempts to categorize a class as equivalent to a JC3IEDM class. Since all JC3IEDM sibling classes are disjoint, two classes can only be equivalent if they are subclasses of the same class. Moreover, it can use the term mappings to place source and target ontology concepts into the taxonomy of JC3IEDM classes.

  LOM performs this step using a custom-created, predefined set of mappings from WordNet® terms to SUMO/MILO terms. The approach in this paper also uses a mapping in this step, though it can be created with less effort because much of it can be generated automatically. Simply start with high-level JC3IEDM terms such as "Facility" and "Action." Use WordNet® to identify the common synonyms and hypernyms of these terms. Then map the synonyms and hypernyms back to the JC3IEDM terms.

The overall ranking approach is similar to LOM's. It assigns a 0 or 1 rating to terms in the first step, then assigns a fractional value (between 0 and 1) in steps 1a, 2, 3, and 4. However, it allows the results of steps 5 and 6 to override the rankings of steps 1–3. Steps 3a and 3b have the potential to demonstrate conclusively that two concepts are *not* equivalent. These steps also have the potential to increase confidence in equivalence by showing fundamental and formal similarities, so the ranking allows them to increase a value computed in previous steps.

Steps 3a and 3b make the algorithm for computing rankings more convoluted than that employed by LOM. However, only after step 3 is the information available to perform them.

7. A Framework for Ontology Resolution

The algorithm in Section 6 yields a set of rankings. This section discusses how to use those rankings.

LOM's confidence factor scores were judged to have a 71% precision rate. That rate makes LOM a good tool for a human analyst, but the figure is not nearly good enough for use in a fully automated mode when matches are accepted without human review, especially during life-risking military C2 operations. Therefore, the algorithm in this paper would be used by a software agent whose task is not to determine equivalences between terms but rather, given a source ontology, to discover target ontologies that probably contain conceptually equivalent terms. The agent would search the NCE for ontologies (by checking registries). On finding an ontology, it would analyze terms in the ontology and generate a report of confidence factor scores. A human analyst would analyze this report and decide which terms are in fact matches. The analyst would then modify the source ontology to record, for each match, the term in the target ontology to which it is equivalent.

Barring an unforeseen technological leap, it is unlikely that an automated agent could ever be trusted to determine the equivalence of two terms for which no formal relationship already exists. A more probable scenario is one in which an agent would determine that two ontologies describe the same domain. This kind of evaluation would be based on an overwhelming similarity between the terms in a source and target ontology. That is, some large percentage of terms in the source ontology would correspond to terms in the target ontology (the term pairs would have a high confidence factor). For any non-trivial ontology, the likelihood of its terms matching those in another ontology and yet not being conceptually related to that ontology is low.

8. Summary

The NCE is intended to be an environment in which automated agents can discover information and services. A prerequisite of discovery is that the agent be able to infer the "meaning" of a discovered term. Ontologies have been proposed as one mechanism to enable inference. For this mechanism to work, there must exist some means to discover and state the conceptual similarities that exist between ontologies.

This paper builds on LOM, an existing tool to help analysts create mappings between ontologies. The extension to LOM capitalizes on the similarities we expect to exist between two ontologies that deal with command and control. In particular, the use of the JC3IEDM provides a more specific context for term resolution than does LOM's use of SUMO and MILO. This should improve the precision of the confidence factors the algorithm generates. The approach is clearly less of a general-purpose aid than LOM, but C2 is an important application area. The NCE will contain many ontologies that include C2 concepts. A domain-specific algorithm should fill a comfortable niche.

Like LOM, the framework in which this paper's approach is used requires human assessment of confidence factors. Technology has not reached the point where automated tools can be trusted to infer contextual similarity. However, any improvement that decreases analysts' workloads should be welcome.

9. References

[1] Lichtblau, Dale E., et al., "Taxonomic and Faceted Classification for Intelligent Tagging and Discovery in Net-Centric Command and Control," 2006 CCRTS, San Diego, CA.

[2] OWL Web Ontology Language Reference, February 2004. See http://www.w3.org/TR/owl-ref.

[3] Li, John, "LOM: A Lexicon-based Ontology Mapping Tool." See http://reliant.teknowledge.com/DAML/I3con.pdf.

[4] Pease, A., Niles, I., Li, J., "The Suggested Upper Merged Ontology: A Large Ontology for the Semantic Web and its Applications," in Working Notes of the AAAI-2002 Workshop on Ontologies and the Semantic Web.

[5] Niles, I., and Terry, A., "The MILO: A general-purpose, mid-level ontology," Proc. International Conference on Information and Knowledge Engineering (IKE'04), Las Vegas, Nevada. June 21-24, 2004.

[6] Fellbaum, C. (ed.), *WordNet: An Electronic Lexical Database*, MIT Press, 1998.

[7] The Joint C3 Information Exchange Data Model Overview, MIP-NATO Management Board, December 2006. See http://www.mip-site.org/publicsite/04-Baseline_3.0/JC3IEDM-Joint_C3_Information_Exchange_Data_Model/JC3IEDM-Overview-UK-DMWG_Edition_3.1_2006-12-08.pdf.

[8] CJCSM 3500.04D, August 2005, directive current as of 17 August 2006, http://www.dtic.mil/cjcs_directives/cdata/unlimit/m350004.pdf.

[9] Sirin, E., et al., *Pellet: A practical OWL-DL reasoner*, Journal of Web Semantics, 2006.

Author Biographies

STEVEN WARTIK is a research staff member in the Information Technology and Systems Division of the Institute for Defense Analyses, where he has worked since 1997. His background is in Computer Science and Software Engineering, and he now concentrates on information modeling, especially with respect to command and control systems. Currently he is studying the role that OWL-based ontologies can play in a net-centric environment. He received his Bachelor's degree from the Pennsylvania State University in

1977, and his Ph.D. in Computer Science and Engineering from the University of California at Santa Barbara in 1983.

DALE LICHTBLAU is a senior member of the research staff in the Information Technology and Systems Division of the Institute for Defense Analyses. He specializes in intelligent systems, data and functional architectures for large-scale information systems, and information assurance. In addition to his current work on cross-domain ontology issues, he is helping to devise an enterprise-level information framework for the U.S. Army. He received his Ph.D. degree, specializing in tense logic, from the Department of Philosophy of the Ohio State University in 1976.