Ontology for the Intelligence Community (OIC-2007)
November 28-29, 2007 · Columbia, Maryland

Toward Automated Provability-Based Semantic Interoperability Between Ontologies for the Intelligence Community

(extended abstract)

Andrew Shilliday, Joshua Taylor, Selmer Bringsjord
{`shilla,tayloj,selmer`}`@rpi.edu`
Department of Cognitive Science
Department of Computer Science
Rensselaer AI & Reasoning (RAIR) Lab:
`http://www.cogsci.rpi.edu/research/rair/`
Troy NY 12180 USA

September 27, 2007

1 Introduction

The need for interoperability is dire: Knowledge representation systems employ ontologies that use disparate formalisms to describe related domains; to be truly useful to the intelligence community, they must meaningfully share information. Ongoing research [3, 4, 7, 15] strives toward the holy grail of complete interoperability, but has been hindered by techniques that are specialized for particular ontologies, and that lack the expressivity needed to describe complex ontological relationships. In the sequel, we describe *provability-based semantic interoperability* (PBSI) [16], a means to surmount these hindrances; *translation graphs*, one of our key formalism for describing the complex relationships among arbitrary ontologies; and ways in which these techniques might be automated.

2 PBSI and PBSI$^+$

We clarify our uses of *syntactic* and *semantic*. The *syntax* of a knowledgebase regiments the structure of expressions in it (e.g., that `(mother-of Amy)` is a well-formed KIF term owes to KIF's syntax); *semantics* attribute *meaning* to otherwise abstract constructs (`(mother-of Amy)` designates Amy's mother according to the semantics of an ontology). A *syntactic translation* occurs when knowledge from one ontology is moved into another using the same semantics. In other words, when ontologies describe the same *kind* of things, and differ only in the way object-level information is structured, interoperability is achieved by mere syntactic translation. When ontologies differ not only in syntax, but also in semantics (yet relate meaningfully), a stronger form of translation is needed: *semantic translation* enables the transfer of information across such ontologies. Systems capable of semantic translation (e.g., [4, 6]) provide some language in which to formalize the semantic connections between ontologies. Unfortunately, the relationships associating ontologies may be so complex that translation of knowledge from one ontology into another is not feasible. Moreover, when interoperability is achieved between complex ontologies, justification is needed to support trust that the meaning of the data has been preserved.

PBSI provides a language for formalizing the relationships between ontologies via *bridging axioms*, and our extension, PBSI$^+$, associates each information exchange with a proof certifying the conservation of semantic meaning. The basic construct of PBSI$^+$ is the signature, a collection of statements in the *meta-theory* which, coupled with a set of axioms, captures a given ontology. A signature Σ consists of a set σ of sorts, and a set ϕ of functors. A sort $s \in \sigma$ is a domain — a collection whose elements are considered the same *kind* of thing,[1] (e.g., the months in the year, boolean values, natural numbers, US citizens). A functor $f \in \phi$ maps between objects of the sorts in σ. In the case that f maps onto the boolean values, f is a relation; if it also takes no arguments, it is a proposition. Having defined signatures, the specifications of ontologies, we present *translation graphs*, a framework for bridging signatures (and so, ontologies) while preserving semantics.

3 Translation Graphs

A translation graph, like the one in figure 1, is a directed graph $G = (V, E)$ where the vertices $v \in V$ are each unique signatures, and each edge $e = (u, v) \in E$ describes the application of a primitive operation to u yielding v, viz., adding or removing either a sort or functor. The addition of a new functor also has associated information potentially relating the new functor to existing functors of the modified signature.

As a toy example, let signature Σ_1 consist of the domains $\sigma_1 = \{\mathsf{People}, \mathsf{Firearms}\}$ and just one functor $\phi_1 = \{\mathsf{OwnerOf} : \mathsf{Firearms} \rightarrow \mathsf{People}\}$, which is understood to map a firearm to its owner. Furthermore, signature Σ_2 consists of the domain $\sigma_2 = \{\mathsf{People}\}$ and the functor $\phi_2 = \{\mathsf{IsArmed} : \mathsf{People} \rightarrow \mathsf{Boolean}\}$ so that $\mathsf{IsArmed}$ holds for those people who own guns (in this example, all signatures implicitly have the boolean domain). A translation graph enabling interoperability between these signatures might apply the following primitive operations bridging Σ_1 to Σ_2:

1. *AddFunctor*($\mathsf{IsArmed}$) with the bridging axiom

$$\forall_p \, [\exists_g \mathsf{OwnerOf}(g) = p] \rightarrow \mathsf{IsArmed}(p)$$

so that the the relation $\mathsf{IsArmed}$ holds for any person, p where there is a firearm that p owns.

---

[1] Our current formalization draws on many-sorted logic, and so domains are disjoint. While this is a limitation on the expressitivity of the language (many ontologies require a subsort hierarchy), it is not a technical restriction. Specifically, we are investigating the use of other ontology representation languages [11, 8].

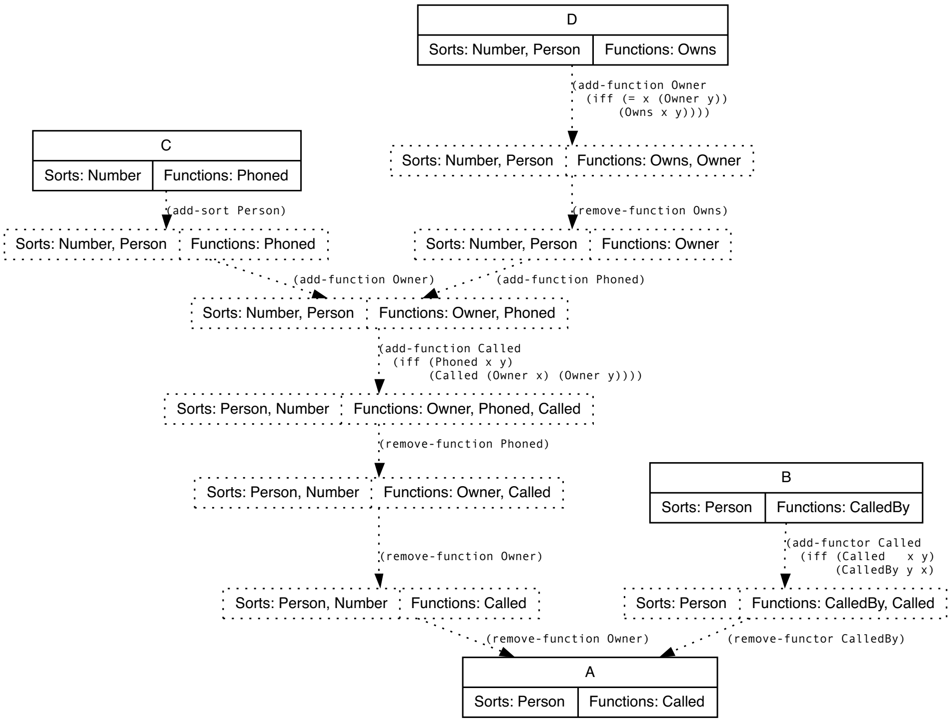

Figure 1: A sample translation graph enabling interoperability between four related ontologies.

2. *RemoveFunctor*(OwnerOf)
3. *RemoveSort*(Firearms)

PBSI between the two described ontologies is made possible: Suppose that the first ontology has among the declarative information in its knowledgebase that Mohammed Al Harbi is the owner of an AKS-74U assault riffle, and that the knowledgebase of the second ontology contains no information about Mohammed Al Harbi except that he is a person. A query of whether or not Mohammed is armed, issued in the second ontology and making use of σ_1's knowledgebase along with bridging axioms generated by traversing the path from σ_1 to σ_2, would yield the correct answer and the associated, certifying proof.

It is important to note that PBSI provides a formal framework and corresponding implementation to break through the n^2 barrier. In the case where translation between several ontologies is desired, translation graphs provide a means to surmount this n^2 problem. This is achieved by use of an *intertheory* through which ontologies are interconnected thereby requiring only $2n$ translation functions (see figure 2). Of course, an even bigger breakthrough would be secured if PBSI could be fully automated, and we turn no to that possibility.

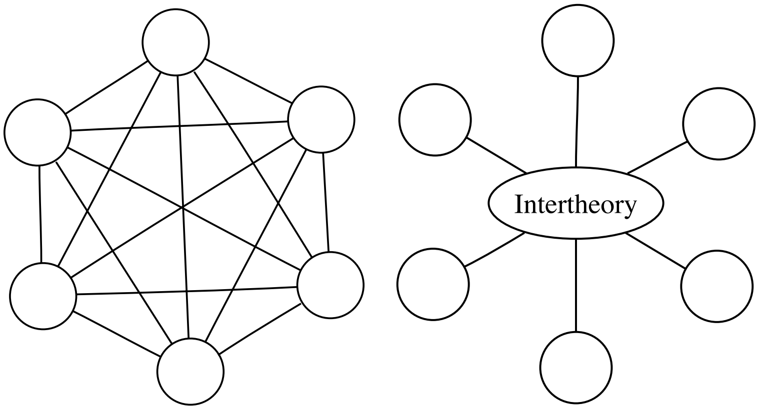

Figure 2: Interoperability between n ontologies (left) typically requires $\binom{n}{2}$ connections but with an intertheory (right), interoperability is achieved using only $2n$.

4 Automation

In this section, we discuss ways to automate the process of creating and applying translations graphs. The procedure to extract appropriate bridging axioms from a translation graph has been accomplished, and systems whose ontologies are present as nodes in a translation graph can interoperate with other nodes in the graph. PBSI does not always yield *translation*; in some cases, bridging axioms can be converted to techniques for syntactic translation, but typically interoperability is achieved by a system issuing a query *expressed in its own syntax and semantics* and the search for an answer incorporates knowledge from related ontologies.

A detailed example of the above is presented in the interoperability experiment [2] between our own advanced reasoning system, Slate, and Oculus' geospatial and temporal visualization system, GeoTime. In the experiment, Slate and GeoTime collaborate to solve a portion of a case study used at the Joint Military Intelligence College. Additionally, the IKRIS Workshop [12] culminated in a demonstration of interoperability between three systems, Slate [1], Cycorp's Nöscape [14], and IBM and Stanford's KANI [5].[2]

This automation gets us half way there, but the holy grail of PBSI is to automate not only the intoperation between systems, but the generation of translation graphs as well. Translation graphs are of course implemented in code, so the challenge of fully automating PBSI^{+} becomes the challenge of so-called *automatic programming* [13]. Because of the capability of the system we have designed for intelligence analysts (Slate), we are optimistic about being able to devise programs that generate the programs that implement translation graphs. Slate integrates deductive, inductive, and abductive reasoning. To the best of our knowledge, there has not been a single effort in automatic programming that synthesizes these three elements. The tradition of deductive program automation [10] is based *exclusively* on deduction; the tradition of machine learning (e.g., genetic programming [9]) is based *exclusively* on induction; while abduction has not even been

---

[2] Demonstrations of these experiments and other Slate-related content is made available online at http://www.cogsci.rpi.edu/slate/Demos/

explored in this field. And yet, typically, when humans approach a programming problem they employ all three of these. They use induction (in tandem with testing and checking) to formulate conjectures about the problem and their tentative solutions; they use deduction in order to reason about the consequences of their design decisions and about the correctness of their solutions; and they use abduction to explain the behavior of their algorithms. We look forward to reporting on our progress toward full automaticity at OIC 2007.

5 A Robust Example

In the presentation corresponding to this extended abstract at OIC 2007 itself, we will also describe a PBSI$^+$-enabled interoperabilty example too robust to present within present space constraints. The example will be based on ongoing DTO-sponsored R&D, in which the aforementioned Oculus and Slate systems interoperate to enable analysts, working on a challenging case study, to issue hypotheses and recommendations that would not otherwise be attainable.

References

[1] BRINGSJORD, S., SHILLIDAY, A., AND TAYLOR, J. Slate, http://www.cogsci.rpi.edu/slate/, 2007.

[2] CHAPPELL, A., BRINGSJORD, S., SHILLIDAY, A., TAYLOR, J., AND WRIGHT, W. Integration Experiment with GeoTime, Slate, and VIKRS. ARIVA Principal Investigator Meeting Handout, March 2007.

[3] CHOI, N., SONG, I.-Y., AND HAN, H. A survey on ontology mapping. *SIGMOD Rec. 35*, 3 (2006), 34–41.

[4] DOU, D., MCDERMOTT, D., AND QI, P. *Ontology Translation by Ontology Merging and Automated Reasoning*. Whitestein Series in Software Agent Technologies and Autonomic Computing. Birkhäuser Basel, 2005, pp. 73–94.

[5] FIKES, R. E., FERRUCCI, D., AND THURMAN, D. A. Knowledge Associates for Novel Intelligence. In *Proceedings of the 2005 International Conference on Intelligence Analysis (IA 2005)* (McLean, VA, USA, May 2005).

[6] GOGUEN, J. A. Data, Schema, Ontology and Logic Integration. *Logic Journal IGPL 13*, 6 (2005), 685–715.

[7] HENDLER, J. Agents and the Semantic Web. *IEEE Intelligent Systems* (March/April 2001), 30–37.

[8] KLYNE, G., AND CARROLL, J. J. Resource description framework (rdf): Concepts and abstract syntax, available at http://www.w3.org/tr/rdf-concepts/. Tech. rep., W3C, February 2004.

[9] KOZA, J. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, 1992.

[10] MANNA, Z., AND WALDINGER, R. Fundamentals of deductive program synthesis. In *Logic, Algebra, and Computation*, F. L. Bauer, Ed. Springer, Berlin, Heidelberg, 1991, pp. 41–107.

[11] MCGUINNESS, D. L., AND VAN HARMELEN, F. OWL Web Ontology Language overview. Tech. rep., W3C, Available at http://www.w3.org/TR/owl-features/, 2004.

[12] MITRE. IKRIS, workshop site: http://nrrc.mitre.org/NRRC/ikris.htm, 2007.

[13] RICH, C., AND WATERS, R. C. Automatic programming: Myths and prospects. *Computer 21*, 8 (Aug. 1988), 40–51.

[14] SIEGEL, N., SHEPARD, B., CABRAL, J., AND WITBROCK, M. Hypothesis Generation and Evidence Assembly for Intelligence Analysis: Cycorp's Noöscape Application. In *Proceedings of the 2005 International Conference on Intelligence Analysis (IA 2005)* (McLean, VA, USA, May 2005).

[15] SMITH, B. The basic tools of formal ontology. In *Formal Ontology in Information Systems* (1998), N. Guarino, Ed., IOS Press, pp. 19–28.

[16] TAYLOR, J., SHILLIDAY, A., AND BRINGSJORD, S. Provability-based semantic interoperability via translation graphs. In *International Workshop on Ontologies and Information Systems for the Semantic Web (ONISW 2007)* (2007).