Ontology Verification with Repositories

Michael GRÜNINGER [a], Torsten HAHMANN [b], Ali HASHEMI [a], Darren ONG [a]
[a] *Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, Ontario, Canada M5S 3G8*
[b] *Department of Computer Science, University of Toronto, Toronto, Ontario, Canada M5S 3G8*

Abstract. In this paper we show how the relationships between first-order ontologies within a repository can be used to support ontology verification. We discuss the use of representation theorems and classification theorems to characterize the models of an ontology, and then show how such results can be obtained from notions such as relative interpretation.

Keywords. ontology repository, ontology evaluation, first-order logic, representation theorems

1. Ontology Verification

Verification is concerned with the relationship between the intended models of an ontology and the models of the axiomatization of the ontology. In particular, we want to characterize the models of an ontology up to isomorphism and determine whether or not these models are equivalent to the intended models of the ontology. This relationship between the intended models and the models of the axiomatization plays a key role in the application of ontologies in areas such as semantic integration and decision support.

We can say that two software systems are semantically integrated if their sets of intended models are equivalent. However, systems cannot exchange the models themselves – they can only exchange sentences in the formal language that they use to represent their knowledge. We must be able to guarantee that the inferences made with sentences exchanged in this way are equivalent to the inferences made with respect to the system's intended models – given some input the application uses these intended models to infer the correct output.

In the area of decision support, the verification of an ontology allows us to make the claim that any inferences drawn by a reasoning engine using the ontology are actually entailed by the ontology's intended models. If an ontology's axiomatization has unintended models, then it is possible to find sentences that are entailed by the intended models, but which are not provable from the axioms of the ontology.

Unfortunately, it can be quite difficult to characterize the models of an ontology up to isomorphism. Ideally, since the classes of structures that are isomorphic to an ontology's models often have their own axiomatizations, we should be able to reuse the characterizations of these other structures. In this paper we show how the relationships between first-order ontologies within a repository can be used to support ontology design and evaluation. Throughout the paper, we will focus on the role that model-theoretic properties play in the application of an ontology repository. In particular, we will address the following challenges:

- How can we identify and characterize the classes of structures that are isomorphic to the models of ontologies?
- What is the weakest theory that is required to axiomatize the intended models of an ontology?
- How can the design of new ontologies be driven by investigating the properties of the models of existing ontologies?

All of these questions are being investigated in the context of the COLORE (Common Logic Ontology Repository) project, which is building an open repository of first-order ontologies that serve as a testbed for ontology evaluation and integration techniques, and that can support the design, evaluation, and application of ontologies in first-order logic. All ontologies are specified using Common Logic (ISO 24707), which is a recently standardized logical language for the specification of first-order ontologies and knowledge bases. At the lowest level are theories of general mathematical structures, such as algebraic structures (e.g. semigroups, groups, rings, vector spaces), and combinatorial structures (e.g. orderings, lattices, graphs). These ontologies serve as the basis for the representation theorems for generic ontologies currently within the repository, such as processes, time, mereotopology, and geometry. Future work will design new ontologies for manufacturing standards (such as ISO 10303 STEP (Standard for the Exchange of Product data) and ENV 12204 (Constructs for Enterprise Modelling)), which will extend and integrate the generic ontologies.

The fundamental insight of this paper is that we can use the relationships between ontologies to assist us in the characterization of the models of the ontologies. The objective of the work is the construction of the models of one ontology from the models of another ontology by exploiting the relationships between these ontologies and their modules in the repository.

2. Relationships Between Ontologies

We begin with the relationship between ontologies and their subtheories within the repository. Theories within the repository will be referred to as modules. An ontology is a theory that consists of one or more modules, so that relationships between ontologies can be defined with respect to the relationships between their modules. We first introduce the set of relationships between ontologies and their modules within a repository, and then show how we can use these relationships to characterize the models of the ontologies.

2.1. Extensions

The simplest relationship between ontologies is that of extension, In particular, the notion of conservative extension has played a key role in the study of modular ontologies ([8], [9]):

Definition 1 *Let T_1 be a first-order theory with language $\mathcal{L}(T_1)$ and let T_2 be a first-order theory with language $\mathcal{L}(T_2)$.*

T_2 is a conservative extension of T_1 iff $\mathcal{L}(T_1) \subset \mathcal{L}(T_2)$ and for any formula $\Phi \in \mathcal{L}(T_1)$, we have $T_2 \models \Phi$ iff $T_1 \models \Phi$.

Several results ([15]) illustrate how conservative extensions of theories are related to extensions and substructures of models of the theories. In particular, if for every model $\mathcal{M}_1 \in Mod(T_1)$ there exists a model $\mathcal{M}_2 \in Mod(T_2)$ such that $\mathcal{M}_1 \subset \mathcal{M}_2$, then T_2 is a conservative extension of T_1. Alternatively, if T_2 is a conservative extension of T_1 then for any model $\mathcal{M}_2 \in Mod(T_2)$ there exists a model $\mathcal{M}_1 \in Mod(T_1)$ such that $\mathcal{M}_1 \subset \mathcal{M}_2$.

Since our goal is to construct the models of an ontology by using the models of theories that are already in the repository, these propositions tell us how to capture and reuse existing constructions. If we have constructed the models of one theory by extending models of the other theory, then this is captured in the repository as a conservative extension relationship. If we know that one theory is a conservative extension of another (e.g. by proof-theoretic means), then we know that models of the theory can be constructed from models of the other.

2.2. Relative Interpretation

Different ontologies within the same language can be compared using the notions of satisfiability, extension, and independence. More difficult is to compare ontologies that are axiomatized in different languages; in such cases, we need to determine whether or not the nonlogical lexicon of one ontology can be interpreted in the nonlogical lexicon of the other ontology. In this section, we review the basic concepts from model theory that will supply us with the techniques for comparing ontologies in different languages.

We will adopt the following definition from [5]:

Definition 2 *An interpretation π of a theory T_0 with language L_0 into a theory T_1 with language L_1 is a function on the set of parameters of L_0 such that*

1. *π assigns to $\forall$ a formula $\pi_\forall$ of L_1 in which at most the variable v_1 occurs free, such that*

$$T_1 \models (\exists v_1)\, \pi_\forall$$

2. *π assigns to each n-place relation symbol P a formula π_P of L_1 in which at most the variables $v_1, ..., v_n$ occur free.*
3. *For any sentence σ in L_0,*

$$T_0 \models \sigma \Rightarrow T_1 \models \pi(\sigma)$$

Thus, the mapping π is an interpretation of T_0 if it preserves the theorems of T_0.

Definition 3 *An interpretation π of a theory T_0 into a theory T_1 is faithful iff there exists an interpretation π of T_0 into T_1 and*

$$T_0 \not\models \sigma \Rightarrow T_1 \not\models \pi(\sigma)$$

for any sentence $\sigma \in \mathcal{L}(T_0)$.

Thus, the mapping π is a faithful interpretation of T_0 if it preserves satisfiability with respect to T_0. We will also refer to this by saying that T_0 is faithfully interpretable in T_1.

For example, the work in [10] shows that the PSL-Core theory within the PSL Ontology [11] is interpretable by Reiter's axiomatization of situation calculus, but that this is not a faithful interpretation, since there are sentences consistent with PSL-Core that are not consistent with situation calculus.

Definable equivalence is a generalization of the notion of logical equivalence between ontologies with the same language.

Definition 4 *Two ontologies T_1 and T_2 are definably equivalent iff T_1 is faithfully interpretable in T_2 and T_2 is faithfully interpretable in T_1.*

Similarly, faithful interpretations are a generalization of the notion of conservative extension.

Theorem 1 *T_1 is faithfully interpretable in T_2 iff there is theory T_3 such that T_1 is definably equivalent to T_3 and T_2 is a conservative extension of T_3.*

2.3. Definability

Relative interpretations specify relationships between theories; we are also interested in specifying relationships between models of the theories. We begin with the notion of definable sets within a structure.

Definition 5 *Let $\mathcal{M}$ be a structure with domain M and language L.*

A set $X \subseteq M^n$ is definable in $\mathcal{M}$ iff there is a formula $\varphi(v_1, ..., v_n)$ of L such that $X = \{\langle a_1, ..., a_n \rangle \in M^n \ : \ \mathcal{M} \models \varphi(\langle a_1, ..., a_n \rangle)\}$.

Using this definition, we can adopt the following approach from [17]:

Definition 6 *Let $\mathcal{N}$ be a structure in $\mathcal{L}_0$ and let $\mathcal{M}$ be a structure in $\mathcal{L}$. We say that $\mathcal{N}$ is definable in $\mathcal{M}$ (equivalently, $\mathcal{M}$ defines $\mathcal{N}$) iff we can find a definable subset X of M^n and we can interpret the symbols of $\mathcal{L}_0$ as definable subsets and functions on X so that the resulting structure in $\mathcal{L}_0$ is isomorphic to $\mathcal{N}$.*

Since we will also be interested in characterizing the models of an ontology's axiomatization up to elementary equivalence, we will need a generalization of the notion definability:

Definition 7 *A structure $\mathcal{M}$ is weakly definable in $\mathcal{N}$ iff there exists a structure $\mathcal{K}$ such that $\mathcal{N}$ is elementary equivalent to $\mathcal{K}$ and $\mathcal{M}$ is definable in $\mathcal{K}$.*

The relationship between relative interpretations of ontologies and definability of structures is captured in a straightforward way by the following theorem:

Theorem 2 *If there exists a faithful interpretation of an ontology T_1 into an ontology T_2, then every model of T_2 defines some model of T_1.*

Using techniques that characterize the definable sets within a model, this result can be used to show when relative interpretations between theories do not exist (see [10]).

2.4. Classes of Theories

The notion of definability allows us to distinguish between different kinds of subtheories of an ontology. We will later see how this distinction impacts the verification of ontologies.

Definition 8 *A theory T_2 is a definitional extension of a theory T_1 iff every constant, function, and relation in any model of T_2 is definable in some model of T_1.*

It is easy to see that a definitional extension of a theory T is also a conservative extension of T, although the converse is not true; that is, there are conservative extensions of theories which are not definitional extensions.

Definition 9 *Two theories T_1 and T_2 with disjoint nonlogical lexicons are synonymous iff there exists a theory S in the language formed by taking the union of the lexicons of T_1 and T_2 such that S is a conservative extension of both T_1 and T_2.*

The notion of logical synonymy is stronger than definable equivalence, and it allows us to explicitly axiomatize the mapping between the ontologies as conservative definitions:

Theorem 3 *T_1 and T_2 are synonymous theories iff there exists a set of conservative definitions Σ_{12} with respect to T_1 such that*

$$T_1 \cup \Sigma_{12} \models T_2$$

and there exists a set of conservative definitions Σ_{21} with respect to T_2 such that

$$T_2 \cup \Sigma_{21} \models T_1$$

For example, the alternative axiomatizations of lattices as posets and as algebras demonstrate that these are synonymous theories.

Definition 10 *A module T_{core} in the repository is a core theory iff no function and no relation in models of T_{core} is definable in the models of any other theory unless that theory is synonymous.*

Core theories are the modules that are the building blocks of the repository. The relations that are axiomatized in a definitional extension are those which are definable in a core theory, and in Section 4 we will see how the definitional extensions are related to the models of the core theories.

Definition 11 *A core hierarchy is a set of core theories $T_1, ..., T_n$ such that $\mathcal{L}(T_i) = \mathcal{L}(T_j)$, for all i, j.*

For example, the theories that axiomatize classes of partial orderings form a core hierarchy – all theories have the same language (i.e. the ordering relation $\leq$) and all are extensions of the theory that contains the three axioms for a partial ordering (transitivity, reflexivity, and antisymmetry).

By the following theorem, theories in the same core hierarchy are related by non-conservative extension.

Theorem 4 *If T_1 and T_2 are core theories in the same core hierarchy, then*

$$T_1 \subset T_2 \Leftrightarrow Mod(T_2) \subset Mod(T_1)$$

In other words, models of a core theory in a core hierarchy are models of its subtheories that are modules in the same core hierarchy; extensions only restrict the sets of models, not the structures of the models themselves. This is not in general the case for all core theories; for example, the core theories in the PSL Ontology do not form a core hierarchy, since each theory expands the nonlogical lexicon; as a result, the models of a core theory in the PSL Ontolgy are constructed by extending models of other core theories.

2.5. Related Work

The use of relative interpretations between first-order axiomatized theories as a means to combine smaller theories has been implemented by the Interactive Mathematical Proof System (IMPS). IMPS is a mathematical theorem prover that utilizes a repository of axiomatized mathematical theories linked to each other through relative interpretations using the little theories approach to mechanize traditional tools of classical mathematical reasoning [7]. The use of relative interpretations by IMPS provide the means to transport a theorem from the theory it was proved in to any other theory linked with an interpretation. The IMPS repository is organized around the relative interpretations available between stored theories. Furthermore, IMPS guarantees the consistency of generated proofs based on the notion of relative consistency between theories. Within IMPS there is a set of theories deemed *foundational*, meaning they are regarded or known to be consistent. Since all proofs begin with a foundational theory and any theory developed from another is a conservative extension of the original theory, all theories developed are consistent relative to the original foundational theory [6]. Although the use and definitions of theory interpretations and relative consistency in IMPS are specific to the purpose of theorem proving, it nonetheless shows how such relationships can be utilized to relate and combine theories. Unfortunately, relative consistency proofs alone are insufficient to verify an ontology. Strictly speaking, we only need to show that a model exists in order to demonstrate that a theory is satisfiable. However, in the axiomatization of domain theories, we need a complete characterization of the possible models.

The Information Flow Framework, an application of category theory to knowledge representation, uses theory interpretations for sharing ontologies in distributed settings [16]. Instances of two different ontologies are linked if they share the same *type* subsumption hierarchy. Finding equivalent *types* between those ontologies are done using theory interpretations through a common upper ontology. A virtual ontology is then formed as a fusion of all participating ontologies and used as the complete system through which the sharing of information occurs [16].

Using $\mathcal{E}$-connections for modularization of OWL ontologies has been explored for both the decomposition of existing ontologies [8] and for ontology design [9]. The $\mathcal{E}$-connection language is a formalism that allows the combination of decidable logics in a way that preserves decidability while adding expressiveness [9]. An $\mathcal{E}$-connection is a set of $\mathcal{E}$-connected ontologies that each model a different application domain, while the $\mathcal{E}$-connection itself models the union of all the domains [8]. In the case of using $\mathcal{E}$-connections to decompose a large ontology into modules (a collection of axioms), each module encapsulates some terms of the original ontology. In [8] the definition of semantic encapsulation is given as a component that preserves a basic set of entailments of a term in an ontology. This leads to the partitioning of a large ontology into a collection of modules that are conservative extensions of one another. The relationship of conservative extensions between modules ensures that each module can be reused independent of the rest while retaining the original semantics of its contained terms. An algorithm for partitioning an OWL ontology into $\mathcal{E}$-connected ontologies is provided in detail in [8]. The major limitation of using $\mathcal{E}$-connections is that the use of such modules as a means of refining an ontology (non-conservative extensions of a theory) becomes impossible.

3. Ontology Verification: Core Theories

Any ontology can be partitioned into a set of core theories and a set of definitional extensions to the core theories. We therefore approach the problem of ontology verification from two perspectives. First, we consider the verification of core theories through representation theorems, and then we consider the verification of definitional extensions through classification theorems. In both cases, we show how the relationships between ontologies within the repository can be used to specify properties of the models of the ontologies.

3.1. Representation Theorems

We can evaluate the adequacy of the application's ontology with respect to some class of structures that capture the intended meanings of the ontology's terms by proving that the ontology has the following two fundamental properties:

- Satisfiability: every structure in the class is a model of the ontology.
- Axiomatizability: every model of the ontology is elementary equivalent to some structure in the class.

The purpose of the Axiomatizability Theorem is to demonstrate that there do not exist any unintended models of the theory, that is, any models that are not specified in the class of structures. In general, this would require second-order logics; however, if we assume that the ontology supports interoperability among first-order inference engines which exchange first-order sentences, then we do not need to restrict ourselves to elementary classes of structures when we are axiomatizing an ontology. Since the applications are equivalent to first-order inference engines, they cannot distinguish between structures that are elementarily equivalent. Thus, the unintended models are only those that are not elementary equivalent to any model in the class of structures.

Classes of structures for theories within an ontology are therefore axiomatized up to elementary equivalence – the theories are satisfied by any model in the class, and any model of the core theories is elementarily equivalent to a model in the class.

In this section, we are interested in constructing models of one ontology by combining models of ontologies in repository, or by decomposing the models of the ontology into models of other ontologies in the repository. We will consider two motivating examples – the mereotopology RT and the core theories PSL-Core and Occurrence Trees from the PSL Ontology.

3.2. Relationship to Interpretability

In this section, we show how representation theorems for ontologies are captured by the relationships between modules within the repository. We begin by a model-theoretic definition for the key property that underlies the Satisfiability and Axiomatizability Theorems:

Definition 12 *A class of structures $\mathfrak{M}$ can be represented by a class of structures $\mathfrak{N}$ iff there is a bijection $\varphi : \mathfrak{M} \rightarrow \mathfrak{N}$ such that for any $\mathcal{M} \in \mathfrak{M}$, $\mathcal{M}$ is weakly definable in $\varphi(\mathcal{M})$ and $\varphi(\mathcal{M})$ is weakly definable in $\mathcal{M}$.*

Note that representation is up to elementary equivalence rather than up to isomorphism, since we are using the notion of weak definability between models.

Theorem 5 *A theory T_1 is definably equivalent with a theory T_2 iff the class of models $Mod(T_1)$ can be represented by $Mod(T_2)$.*

We can use this result in two ways – if we have specified Satisfiability and Axiomatizability Theorems for a particular ontology, we can capture this in the repository by identifying the theories that axiomatize the classes of structures that are elementary equivalent to the models of the ontology. Alternatively, if we are attempting to prove the Satisfiability and Axiomatizability Theorems for the ontology, then we can use the relationships between theories in the repository to identify any other theories that are definably equivalent to the ontology; the models for these other theories will provide the classes of structures that characterize the models of the ontology.

The notion of representability allow us to specify the weakest theory required to axiomatize the intended models of an ontology. For example, some process ontologies may represent a timeline as the real numbers, yet the intended models for the timeline simply require a dense linear ordering. Since the theories in the repository are combined to specify a theory that is definably equivalent to the ontology, eliminating any axioms would result in unintended models.

3.3. Examples from Mereotopology

Within formal ontologies, spatial ontologies are amongst the most widely studied ontologies. Our environment is saturated with spatial and spatio-temporal information. Hence, every upper ontology contains some definitions and axioms dealing with spatial information. Within spatial ontologies, mereotopologies capturing topological (contact) and mereological (parthood) relations between regions of space have been most widely studied. First-order ontologies of mereotopologies are in particular interesting because a large set of possible axioms and their interactions can be studied in a clearly defined setting. In the study of mereotopologies, representation theorems helped to establish the relationship between different ontologies. Lattices in general and, more specifically, contact algebras consisting of a lattice and a binary contact relation satisfying certain axioms have proven useful as representations of Whiteheadian mereotopology [14]. Here, we only summarize some of the results to exemplify the usefulness of representation theorems for verifying ontologies.

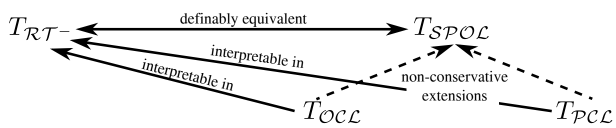

Figure 1. The ontologies and modules $T_{\mathcal{RT}^-}$, $T_{\mathcal{SPOL}}$, $T_{\mathcal{OCL}}$ and $T_{\mathcal{PCL}}$ and their relationships.

For example, the models of the first-order ontology of mereotopology $T_{\mathcal{RT}^-}$ (a generalization of the ontology from [1]) can be represented by the definably equivalent ontology of Stonian p-ortholattices (Stonian, pseudocomplemented, orthocomplemented lattices) $T_{\mathcal{SPOL}}$. The axiomatization of $T_{\mathcal{RT}^-}$ and the precise structure of Stonian details are not important here, but can be found in [13] together with the details of the representation theorem. The main point here is that Stonian p-ortholattices are a natural class of lattices that is definable as the ontology $T_{\mathcal{SPOL}} = T_{\mathcal{OCL}} \cup T_{\mathcal{PCL}} \cup \{S1\}$ consisting of a module for orthocomplemented lattices $T_{\mathcal{OCL}}$ and a module for pseudocomplemented lattices $T_{\mathcal{PCL}}$ strengthened by the Stone identity $(S1)$ $(x+y)^{**} = x^* + y^*$ where * denotes the pseudocomplementation operation. Of course, $T_{\mathcal{OCL}}$ and $T_{\mathcal{PCL}}$ are themselves ontologies; both of them are non-conservative extensions of the ontology of bounded lattices. As mentioned before, these lattices are definable in terms of $\leq$ and thus form a core hierarchy. From the knowledge that $T_{\mathcal{RT}^-}$ is definably equivalent to $T_{\mathcal{SPOL}}$, we can then, for example, deduce that the each of the ontologies $T_{\mathcal{OCL}}$ and $T_{\mathcal{PCL}}$ are interpretable in $T_{\mathcal{RT}^-}$, but not faithfully interpretable in $T_{\mathcal{RT}^-}$. All these classes of lattices have been studied for half a century in lattice theory and are well-understood, cf. [2]. Hence, giving representation theorems for a not-well understood ontology in terms of a better-understood mathematical theories (and their first-order ontologies) helps to verify whether the mereotopology captures the intended models.

Perhaps more well-known is the representation of the models of the popular mereotopology Region Connection Calculus (RCC) [3] by Boolean Contact Algebras (BCA), the latter structures denoted by $\langle A, C\rangle$. For the RCC, we use in the following the theory $T_{\mathcal{RCC}}$ as axiomatized by (RCC1)-(RCC8) from [4]. BCAs, in turn, consist of a standard Boolean algebra $A = \langle A; 1; 0;'; +; \cdot\rangle$ with a bottom element 0, a top element 1 and the sum and intersection operations $+$ and $\cdot$ for which many definably equivalent first-order theories can be found, e.g., in [19]. We denote any such theory based on the single predicate $\leq$ by $T_{\mathcal{BA}}$. Another definition of a Boolean algebra established in [13] is that of a distributive Stonian p-ortholattice. Furthermore, BCAs require a binary contact relation C that is reflexive, anti-symmetric, and extensional as captured by the first-order theory $T_{\mathcal{C}} = \{C1, C2, Ext\}$. For details see e.g. [20]. The axioms (P) and (O) which define the pivotal mereotopological relations of parthood P and overlap O are definitional extensions of $T_{\mathcal{C}}$.

(C1) $x \neq 0 \rightarrow C(x,x)$ (Reflexivity)
(C2) $C(x,y) \leftrightarrow C(y,x)$ (Symmetry)
(Ext) $\forall z(C(x,z) \rightarrow C(y,z)) \leftrightarrow x = y$ (Extensionality)
(P) $P(x,y) \leftrightarrow \forall z(C(x,z) \rightarrow C(y,z))$ (Parthood)
(O) $O(x,y) \leftrightarrow P(x,y) \vee P(y,x)$ (Overlap)

(C0) $\neg C(0,x)$ (Null disconnectedness)
(C3) $C(x,y) \wedge y \leq z \rightarrow C(x,z)$ (Monotonicity of C with respect to $\leq$)
(C4) $C(x, y+z) \rightarrow C(x,y) \vee C(x,z)$ (Topological sum)
(Con) $(x \neq 0 \wedge x \neq 1) \rightarrow C(x,x')$ (Connectedness of x and x')

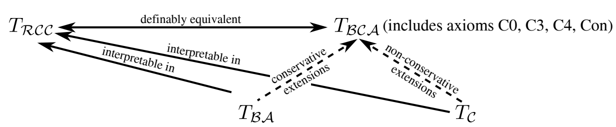

Figure 2. The ontologies and modules $T_{\mathcal{RCC}}$, $T_{\mathcal{BCA}}$, $T_{\mathcal{BA}}$ and $T_{\mathcal{C}}$ and their relationships.

In addition, the axioms $C0, C3, C4, Con$ govern the interplay between the Boolean algebra and the relation C. Notice that in these axioms all relations apart from C, and allfunctions and constants are defined in the Boolean algebra. This results in the theory $T_{\mathcal{BCA}} = T_{\mathcal{BA}} \cup T_{\mathcal{C}} \cup \{C0, C3, C4, Con\}$ being definably equivalent to, for details see $T_{\mathcal{RCC}}$ [20,4]. The mapping between the two theories is defined by three equivalences:
$P(x, y) \leftrightarrow x \leq y$ $\qquad \neg C(x, y) \leftrightarrow x < y$ $\qquad O(x, y) \leftrightarrow x \cdot y \neq 0$

Due to (C0), (C3), (C4), and (Con), we cannot choose arbitrary models from $T_{\mathcal{BA}}$ and $T_{\mathcal{C}}$ and combine them into a model of the theory $T_{\mathcal{BCA}}$. We can however select an arbitrary Boolean algebra and choose a model from $T_{\mathcal{C}}$ accordingly so that we obtain a model of $T_{\mathcal{BCA}}$. In particular, we have to make sure that the equivalence $\neg C(x, y) \leftrightarrow x < y$ is satisfied. The reverse, i.e. taking an arbitrary model of $T_{\mathcal{C}}$ and constructing a model of $T_{\mathcal{BCA}}$ accordingly, is not guaranteed to work.

Both these examples show how representation theorems can be used to obtain several relationships between different ontologies and their modules (see Figures 1 and 2). The first example of the theory T_{RT^-} demonstrated how the intersection of two theories is captured, while the second example of the theory T_{RCC} shows how a module (T_{BA}) can be supplemented by another module (T_C) to build an ontology definably equivalent to another ontology. In these examples, we exploited the relationships between the models of the different theories to characterize the relationships between the ontologies and modules. They also demonstrate that there are different ways in which models of multiple ontologies can be combined to construct complex ontologies. The next section formalizes one special case thereof with especially well-behaved properties.

3.4. Reducibility

Suppose that we wish to provide representation theorems for some new theory T. Even if there is no single ontology (i.e. set of modules) in the repository that is definably equivalent to T, we can still use the models of repository theories to specify a representation of $Mod(T)$. We want to capture the following intuition: the representation of $Mod(T)$ is reducible to the representations of $Mod(T_1)$, ..., $Mod(T_n)$ for some set of modules $T_1, ..., T_n$ in the repository. Moreover, we want to characterize this notion of reducibility by using the relationships among modules in the repository. The question is therefore how we can use the representations of the $Mod(T_i)$ to specify the representation of $Mod(T)$.

As we did with the notion of representation, we begin with a model-theoretic definition of the relationship:

Definition 13 *A class of structures $\mathfrak{M}$ is reducible to the classes of structures $\mathfrak{N}_1, ..., \mathfrak{N}_n$ iff there is a set of surjections $\varphi_i : \mathfrak{M} \rightarrow \mathfrak{N}_i$ such that if $\mathcal{M} \in \mathfrak{M}$ and*

$$\mathcal{M} = \mathcal{M}_1 \cup ... \cup \mathcal{M}_n$$

then $\mathcal{M}_i$ is weakly definable in $\varphi_i(\mathcal{M}_i)$ and $\varphi_i(\mathcal{M}_i)$ is weakly definable in $\mathcal{M}_i$.

As we did with representability, we can show how the model-theoretic notion of reducibility can be characterized by the relationships between modules in the repository:

Theorem 6 *Let T be a theory and let $T_1, ..., T_n$ be a set of modules in the repository.*
$Mod(T)$ is reducible to $Mod(T_1), ..., Mod(T_n)$ iff

- *T faithfully interprets each theory T_i, and*
- *$T_1 \cup ... \cup T_n$ is definably equivalent to a subtheory $T' \subset T$ such that*

$$Mod(T) \subset Mod(T')$$

If a theory T is reducible to a set of theories, then from any model $\mathcal{M}_i \in Mod(T_i)$, we can construct a model $\mathcal{M}$ of T; each such model is isomorphic to a substructure of $\mathcal{M}$. The axioms in $T \setminus T'$ are expressed in the same language as T; in this sense, they are not satisfied by new classes of structures, but instead specify *how* the substructures $\mathcal{M}_i \in Mod(T_i)$ are assembled to construct a model $\mathcal{M}$ of T. If a theory T is not reducible to a set of theories, then there exist models of one theory which cannot be combined with models of other theories to construct a model of T. In this case, T is not a faithful interpretation of one of the theories. For example, the theory $T_{\mathcal{RCC}}$ from the previous section is not reducible to the theories $T_{\mathcal{BA}}$ and $T_{\mathcal{C}}$. The next section gives an example of a reducible theory.

3.5. Examples from the PSL Ontology

The models of the PSL-Core theory [18] provide an example of reducibility. In particular, we can identify three classes of structures that can be combined to construct models of PSL-Core. Two of these are classes of incidence structures which are represented by the decomposition of graphs:

Definition 14 *Let $G = (V, E)$ be a directed graph with no nontrivial cycles, and let $P(E)$ be a partitioning of the edges in G.*
A partitioning incidence structure is the tripartite incidence structure of rank 3:

$$\mathbb{I} = (P(E), E, V, \in)$$

A graph incidence structure is the bipartite incidence structure of rank 2:

$$\mathbb{I} = (E, V, \in)$$

Theorem 7 *Let T_{linear} be the theory of linear orderings, let $T_{partition}$ be the theory of partitioning incidence structures, and let $T_{graph-incidence}$ be the theory of graph incidence structures.*
$T_{pslcore}$ is reducible to $T_{linear} \cup T_{partition} \cup T_{graph-incidence}$.

Thus, any linear ordering, partitioning incidence structure, and graph incidence structure can be combined to construct a structure that is isomorphic to a model of $T_{pslcore}$. The additional axioms in $T_{pslcore}$ specify how the substructures are combined by constraining the mapping of timepoints, activity occurrences and objects to the linear ordering.

3.6. Foundational Theories

The methodology proposed in this section leads to some obvious questions. If all core theories in the repository have representation theorems, how do we avoid infinite regress? Do there exist theories which are foundational in the sense that the models of all other theories are represented using the models of the foundational ones? If so, how can we characterize the models of such foundational theories?

There are three techniques that can be used to address these questions. First, there are some theories that are so general that they can be used to represent the models for a wide range of theories. For example, graphs (symmetric binary relations) and preorders (transitive, reflexive relations) are used to specify the structures for ontologies such as [21], yet we do not provide representation theorems for graphs and preorders. Also note that powerful theories such as Peano Arithmetic do not play the role of foundational theories in this sense, since representability requires that the models of the two theories are equivalent, whereas the models of Peano Arithmetic properly contain many other classes of models as substructures.

In some cases, we can use the notion of mutual interpretability, in which two different theories are definably equivalent so that both can be used as foundational. For example, classes of lattices are often characterized by topological representations.

A third technique is to use structure theorems to specify the models of a theory. In this case, we provide representation theorems for a small class of structures and then specify how all models can be constructed from such “building block" structures. This approach is taken for theories such as linear orderings, groups, rings, and modules.

4. Ontology Verification: Definitional Extensions

The evaluation of the definitional extensions within an ontology is based on the notion of definable sets and the particular property of such sets that they are preserved by automorphisms of the underlying models.

4.1. Classification Theorems

Many ontologies are specified as taxonomies or class hierarchies, yet few ever give any justification for the classification. If we consider ontologies of mathematical structures, we can classify models by using properties of models, known as invariants, that are preserved by isomorphisms. For some classes of structures, invariants can be used to classify the structures up to isomorphism; for example, vector spaces can be classified up to isomorphism by their dimension. For other classes of structures, such as graphs, it is not possible to formulate a complete set of invariants. Nevertheless, even without a complete set, invariants can still be used to provide a classification of the models of a theory. In a sense, classification theorems provide an alternative approach to characterizing the models of a core theory – if a model can be reconstructed up to isomorphism from the invariants alone, then the invariants provide an implicit representation of the models.

This approach can also be justified by the application of ontologies to support semantic integration. If the ontologies of two software applications have the same language, then the applications will be interoperable if they share the semantics of the terminology in their corresponding theories. Sharing semantics between applications is equivalent to sharing models of their theories, that is, the theories have isomorphic sets of models. We therefore need to determine whether or not two models are isomorphic, and in doing so, we can use invariants of the models.

4.2. Relationship to Interpretability

Following this methodology, the set of models for a core theory can be partitioned into equivalence classes defined with respect to the set of invariants of the models. Each equivalence class in the classification of the core theory's models is axiomatized using a definitional extension of the core theory. In particular, each definitional extension of the core theory is associated with a unique invariant for the models of the theory. The different classes and relations that are defined in the extension correspond to different properties of the invariant. In this way, the terminology of an ontology arises from the classification of the models of the core theories with respect to sets of invariants.

Invariants for models may also be related to invariants of their substructures. For example, definitional extensions of PSL-Core axiomatize classes of activity occurrences that arise from the automorphism group of the occurrence structure within a model of PSL-Core, and classes of timepoints arise from the automorphism group of the timeline structure. Other examples of invariants for the PSL Ontology were first discussed in [12]. The most prevalent class of occurrence constraints is that of *markovian* activities, activities whose preconditions depend only on the state prior to their occurrences (e.g., to withdraw money from a bank account, there must be sufficient funds in the account). The class of markovian activities is defined in the PSL definitional extension $state\_precond.def$. In this case, the substructure of a model of PSL characterizes the relations between states and activity occurrences. The invariant that corresponds to this extension is the automorphism group and the classes that are defined in the extension correspond to different classes of groups.

5. Ontology Design

The design of new core theories can be driven by investigating the properties of the models of existing ontologies within the repository. For example, models of existing mereotopologies are represented by classes of lattices (e.g. Boolean lattices and Stonian p-ortholattices). It is possible to design new mereotopology ontologies by searching for other classes of lattices axiomatized within the repository which can be used to represent alternative ontological choices with respect to the parthood and connection relations.

Definitional extensions of a core theory T axiomatize the classes of models in $Mod(T)$. Since the class of models of a core theory is a restriction of the class of models of other theories in the same core hierarchy, nonconservative extensions of a core theory T are often related to the classification of the models of T with respect to invariants.

Another approach to the design of new ontologies is to consider the definability of invariants. Definitional extensions arise when the invariants for models of T are definable in T. In some cases, however, the invariants for models of a theory T may not be definable in T, but rather may be definable in a core theory which is an extension of T.

6. Summary

The concepts and methods discussed in this paper, in particular the relationships of relative interpretation together with conservative and nonconservative extension, can be used to organize the theories within an ontology repository. We have shown how to use these relationships between ontologies to assist us in the characterization of the models of the ontologies. In particular, we can use the notion of interpretability to specify representation theorems, and use the notion of reducibility of structures to construct the models of one ontology from the models of another ontology by exploiting the relationships between these ontologies and their modules in the repository.

References

[1] Asher, N. and Vieu, L. (1995) Toward a geometry of common sense: a semantics and a complete axiomatization for mereotopology. In *Proc. of the 14th Int. Joint Conf. on Artificial Intelligence (IJCAI'95)*, pp. 846–852. Morgan Kaufmann, 1995.

[2] T. S. Blyth. *Lattices and Ordered Algebraic Structures*. Springer London, 2005.

[3] Cohn, A.G., Bennett, B., Gooday, J.M., and Gotts, N.M. (1997) RCC: a calculus for region based qualitative spatial reasoning. *GeoInformatica*, 1:275–316, 1997.

[4] Düntsch, I., Wang, H., and McCloskey, S. (1999) Relation algebras in qualitative spatial reasoning. *Fundamenta Informaticae*, 39(3):229–248, 1999.

[5] Enderton, H. (1972) *Mathematical Introduction to Logic*,Academic Press.

[6] Farmer, W. M. (2000) An Infrastructure for Intertheory Reasoning. In: CADE-17, *Lecture Notes in Computer Science (LNCS)*, D. McAllester, ed., p. 1831:115-131.

[7] Farmer, W. M., Guttman, J. D., Thayer, F. J. (1992) Little Theories. In: CADE-11, *Lecture Notes in Computer Science (LNCS)*, D. Kapur, ed., p. 607:567-581.

[8] Grau, B., Parsia, B., Sirin, E., Kalyanpur, A. Automatic Partitioning of OWL Ontologies Using E-Connections, *Proc. of Description Logic Workshop (DL)*, 2005.

[9] Grau, B., Parsia, B., Sirin, E. (2009) Ontology Integration Using E-connections, pp.293-320, *Modular Ontologies*, Stuckenschmidt et al. (eds.). Springer-Verlag, Berlin.

[10] Gruninger, M. (2009) Definability and Process Ontologies, *Ninth International Symposium on Logical Formalizations of Commonsense Reasoning*, Toronto, Canada.

[11] Gruninger, M. (2009) Process Ontologies, pp. 419-421, *Handbook of Ontologies, 2nd Edition*, S. Staab (ed.). Springer-Verlag, Berlin.

[12] Gruninger, M. and Kopena, J. (2004) Semantic Integration through Invariants, *AI Magazine*, 26:11-20.

[13] Hahmann, T., Winter, M., and Gruninger, M. (2009) Stonian p-ortholattices: A new approach to the mereotopology RT_0. *Artificial Intelligence*, 173:1424–1440, 2009.

[14] Hahmann, T., and Grüninger, M. (2010) Region-based theories of space: Mereotopology and beyond. In *Qualitative Spatio-Temporal Representation and Reasoning: Trends and Future Directions*. IGI Publishing. in press.

[15] Hodges, W. (1993) *Model Theory*. Cambridge University Press.

[16] Kent, Robert E. (2000) The Information Flow Foundation for Conceptual Knowledge Organization. In: Dynamism and Stability in Knowledge Organization. Proceedings of the Sixth International ISKO Conference. Advances in Knowledge Organization 7 (2000) 111-117. Ergon Verlag, Wurzburg.

[17] Marker, D. (2002) *Model Theory: An Introduction*. Springer Verlag.

[18] Menzel, C. and Gruninger, M. (2001) A formal foundation for process modeling, *Second International Conference on Formal Ontologies in Information Systems*, Welty and Smith (eds), 256-269.

[19] Padmanabhan, R. and Rudeanu, S. (2008) *Axioms for Lattices and Boolean Algebras*. World Scientific, 2008.

[20] Stell, J.G. (2000) Boolean connection algebras: a new approach to the Region-Connection Calculus. *Artificial Intelligence*, 122:111–136, 2000.

[21] Troquard, N., Trypuz, R., Vieu, L. (2006) Towards an ontology of agency and action: From STIT to ONTOSTIT+, pp. 179-192, *Proceedings of the Fourth International Conference on Formal Ontology in Information Systems*, B. Bennett and C. Fellbaum (eds.) Baltimore. IOS Press.