SUPPORTING STAKEHOLDER COLLABORATION IN INFRASTRUCTURE PROJECTS USING ONTOLOGY MERGER

Nora M. El-Gohary [1] and Tamer E. El-Diraby [2]

ABSTRACT

Semantic (ontology-based) collaborative web tools are promising means for supporting stakeholder collaboration for integrated infrastructure development. However, the development of a single ontology to serve the variety of collaborating stakeholders is not feasible given the multidisciplinary nature of the domain and the uniqueness of each project. As such, to efficiently develop and deploy such collaborative tools, ontology merging techniques are required for facilitating interoperability among ontologies of collaborating stakeholders. However, since none of the existing ontology merging tools allow for the merging of ontology axioms, there is still a need for developing tools for axiom merging to provide full interoperability between stakeholder ontologies. This paper discusses the relation between knowledge base integration and ontology axiom merging. It then presents an existing methodology for knowledge base integration. Further, a discussion of this methodology is provided in the context of ontology axiom merging. Finally, the paper proposes a semi-automatic approach for ontology axiom merging.

KEY WORDS

infrastructure, collaboration, stakeholders, ontology, interoperability, merging.

INTRODUCTION

Collaborative development of infrastructure systems aims at integrating the business and engineering processes between different stakeholders and across various projects to assure consistency in design, coordinated construction, consideration of sustainability, and better handling of infrastructure interdependency. The need for such collaborative integrated infrastructure development has been emphasized by many researchers (Kazi and Charoenngam 2003, Chinowsky and Rojas 2003, Wilson et al. 2001).

Semantic web-based tools are promising means for supporting stakeholder collaboration for integrated infrastructure development. These tools could provide useful services such as providing content-based information browsing, assuring timely input from relevant parties, disseminating the right information to the right party, and capturing corporate / team memory

[1] PhD Candidate, Department of Civil Engrg., 35 St. George Street, Univ. of Toronto, Toronto, ON, Canada, M4Y 2P8, Phone +1 416/978-5964, FAX 416/978-6813, nora.el.gohary@utorotno.ca

[2] Assistant Professor & Director, Centre for Info. Sys. in Infrastructure & Construction, Department of Civil Engrg., 35 St. George Street, Univ. of Toronto, Toronto, ON, Canada, M4Y 2P8, Phone +1 416/978-8653, FAX 416/978-5054, tamer@ecf.utorotno.ca

throughout the life cycle of the collaboration. The core of semantic tools is the development of ontologies. *Ontologies* are a common vocabulary of terms, definitions, and inter-relationships with explicitly defined and machine understandable semantics. Semantics are defined using axioms (rules). By defining shared domain theories, ontologies allow for both individuals and machines to communicate more effectively. Ontologies are thus the foundation of content-based information access and semantic interoperability over the web (El-Diraby et al. 2005). However, the development of a single ontology to serve the whole domain is not feasible given the multidisciplinary parties that are involved and the unique nature of each project. As such, semantic communication is still hindered due to lack of interoperability between different ontologies.

The development of ontology merging techniques thus becomes essential. *Ontology Merging* is the process of deriving a new ontology based on two or more existing ontologies in order to facilitate interoperability between the existing ontologies (McGuinness et al. 2000). Generally, merging two ontologies involves: 1) finding semantic correspondences between both ontologies (i.e. mapping) and 2) merging both ontologies based on the established correspondences. Different methodologies and approaches for ontology mapping and merging are being developed and used, such as Chimaera, PROMPT, ONION, GLUE, FCA-Merge, IFMap, KAON, Edamok, Matchmaker, and OBSERVER. However, none of these initiatives allow for merging of axioms. Since axioms are necessary to define the semantics of terms (Gruninger and Fox 1995) and to provide meta-knowledge about concepts and relations, there is still a need to develop tools for axiom merging to provide full interoperability between ontologies to support seamless exchange of semantic information.

INTEGRATION OF KNOWLEDGE BASES VS. INTEGRATION OF ONTOLOGY AXIOMS

A variety of approaches for integrating knowledge bases have been introduced in the literature (Zobel et al. 2005), i.e. combining pieces of information and rules originating from multiple sources. The review of these different approaches is beyond the scope of this paper. This section will only present brief background information on three areas: 1) the problem of rule integration, 2) Lin's approach to knowledge base integration (1996), which was re-introduced and discussed by Zobel et al. (2005), and 3) some elements of Wang's et al. approach for multiple rule sets integration (1998).

Both, the literature of ontology integration and the review of existing ontology mapping and merging tools, reveal that an approach for the merging of ontology axioms has not yet been put forth. On the other hand, various approaches for the integration of knowledge bases have been presented in the literature by experts in the domains of database and logic. This paper argues that there is a great resemblance between the problem of knowledge base integration and that of ontology axiom merging. Both deal with the integration of a multiple set of rules that come from multiple sources; and both face similar integration problems, such as: redundancy, subsumption, and conflict. As such, both the knowledge base merging and the ontology axiom merging problems can be subsumed under the general problem of rule merging. As such, this paper attempts to learn from and extend on the work done in knowledge base integration area.

Lin (1996) proposed an approach for merging knowledge bases that resolves conflicts in the combined knowledge base. In his approach, he defines a knowledge base as a finite set of sentences. He also defines a function ***w*** that assigns each of the knowledge bases a non-negative number representing the relative degree of 'trust' in the knowledge base. He further introduces a merging operator *merge* as a mapping from a set of knowledge bases and ***w*** to a new merged knowledge base. According to Lin, it is desirable to have the merged knowledge base: 1) contain a *maximal* amount of information from each knowledge base, and 2) differ *minimally* from each knowledge base. As such, if there is no conflict among the original knowledge bases to be merged, it is desirable that the merged knowledge base be simply the union of the original knowledge bases. If a conflict exists among the original knowledge bases, the merging is performed using the *merge* operator such that the merge result is consistent and is based on the weighted majority principle. Consistency means that conflicts among the knowledge bases are always resolved. Weighted majority means that the opinion of the majority prevails. As such, if three knowledge bases of equal ***w*** values are to be merged and two knowledge bases support C while only one opposes C (i.e. supports $\neg$C), then the merged knowledge base will support C. Lin mathematically proves that his merging approach is consistent, complete and reflects the weighted majority principle.

Wang et al. (1998) propose a genetic-algorithm based approach for merging multiple rule sets into a single rule set. This paper will not discuss the details of the approach, as it is not relevant to the context of the paper. However, the paper will highlight two relevant features of the approach: 1) elimination of redundancy and 2) elimination of subsumption. If two rules are found to be matching, Wang's et al. *fusion* operator removes one of these rules, and thus eliminates the redundancy. Similarly, if a rule subsumes another rule, the *fusion* operator eliminates the subsumed rule.

LIN'S METHODOLOGY FOR INTEGRATION OF KNOWLEDGE BASES

The main principles of Lin's approach for integration of knowledge bases were presented above. This section will illustrate his methodology through few simple examples. Suppose K_1 and K_2 are two knowledge bases to be merged, while each knowledge base consists of a finite set of clauses (sentences). These sentences could be either 'atoms', propositions that cannot be decomposed; or 'propositional compositions', propositions that are combined using logical operators ($\vee$, $\wedge$, $\supset$) to create more complex statements. Accordingly, if A, B and C represent individual atoms such as "C.C.C. Company is an electrical subcontractor", then the following statements are defined as follows:

- A means 'A is true'
- $\neg$A means 'A is not true'
- A $\wedge$ B means 'A is true and B is true'
- A $\vee$ B means 'A is true or B is true'
- A $\supset$ B means 'A implies B' , in other words, 'if A is true then B is true'

Two formulas are called equivalent, if the truth values of both formulas are the same under all possible interpretations. Logical equivalence is denoted by $\equiv$, for example, (A $\supset$ B) $\equiv$ ($\neg$A$\vee$B). The following is a brief description of Lin's methodology for knowledge base integration (Lin 1996 and Zobel et al. 2005):

Step 1: For each knowledge base, consider all clauses (whether atoms or propositional compositions) that cause a conflict with any other knowledge base that is being merged. Other clauses that are irrelevant to the conflict are preserved. For example, suppose that the following three statements are the source of conflict, each statement belonging to one knowledge base, and all three knowledge bases have a same weight of 1:

K_1: A

K_2: $A \supset B$

K_3: $A \wedge \neg B$

Step 2: Put each clause into a logically equivalent disjunctive normal form (DNF). A DNF knowledge base is composed of a set of disjuncts (statements combined using 'or' operator) that contain only logical conjunction ('and' operator). For example:

K_1: A (one statement: {A})

K_2: $A \supset B \equiv \neg A \vee B$ (two statements: $\{\neg A\}$ & $\{B\}$)

K_3: $A \wedge \neg B$ (one statement: $\{A \wedge \neg B\}$)

Step 3: For each disjunctive clause, assign the weight of its respective knowledge base to each atom (in the superscript of the atom). For example:

K_1: A^1

K_2: $\neg A^1 \vee B^1$

K_3: $A^1 \wedge \neg B^1$

Step 4: Taking only one disjunct from each knowledge base at a time, form all possible union combinations of disjuncts. For example:

Combination 1: $A^1 \wedge \neg A^1 \wedge A^1 \wedge \neg B^1$

Combination 2: $A^1 \wedge B^1 \wedge A^1 \wedge \neg B^1$

Step 5: For each combination, cancel true and false pairs of atoms and then calculate its total weight as the sum of its individual atom weights. For example:

Combination 1: $A^1 \wedge \neg A^1 \wedge A^1 \wedge \neg B^1 = A^1 \wedge \neg B^1$ (total weight = 2)

Combination 2: $A^1 \wedge B^1 \wedge A^1 \wedge \neg B^1 = A^2$ (total weight = 2)

Step 6: Select the combinations such that the weights of their inconsistent parts are minimum, i.e. the combination that provides the largest total weight. In case the opposing and supporting forces for a set of combinations are in balance (i.e. there exists a tie between total weights), then these combinations are combined together by disjunction. For example:

$K_{combined}$: $(A \wedge \neg B) \vee (A)$

APPLYING LIN'S METHODOLOGY TO INTEGRATION OF ONTOLOGY AXIOMS

As discussed above, there is a great resemblance between the problem of knowledge base integration and that of ontology axiom merging. Therefore, this section will try to apply Lin's methodology for knowledge base integration to ontology axiom merging. For example, suppose that two stakeholders are collaborating over the development of a street project, stakeholder 1 being the project manager with 25 years of experience in the infrastructure domain and stakeholder 2 being a subcontractor with 10 years of experience. To facilitate the collaboration process, both stakeholders are using a web ontology-based portal that provides a variety of collaboration services. Each stakeholder is using an ontology that represents his process knowledge and the way he performs his business processes. As such, both stakeholders need to merge their ontologies to create a combined process structure for using the portal services. Suppose that ontology 1 (of stakeholder 1) is given a weight of 2, while ontology 2 is given a weight of 1; since stakeholder 1 is the project manager and has more experience in the subject domain. Generally, axioms in the subject ontologies provide semantics and constraints on the interpretation of ontology terms and / or define the business rules of stakeholders. Note that this section will consider the problem of ontology axiom merging only, as the problem of concept and relation merging is outside the scope of this paper.

Further, suppose that both ontologies have only two conflicting sentences, one belonging to each ontology, which need to be merged in a consistent way that preserves maximal amount of axioms and applies majority-rule: O_1: $A \supset B$ and O_2: $A \supset \neg B$, with $w(O_1)=1$ and $w(O_2)=2$; where A represents the atom "the stakeholder is a subcontractor" and B represents the atom "is authorized to approve change orders". Obviously, there is a potential conflict among theses two sentences; if A becomes true, O_1 will conclude B (will authorize the subcontractor to approve change orders) and O_2 will conclude $\neg B$ (will NOT authorize the subcontractor to approve change orders). Applying Lin's methodology to this case, will provide the following merge results:

Step 1:
O_1: $A \supset B$
O_2: $A \supset \neg B$

Step 2:
O_1: $A \supset B \equiv \neg A \vee B$
O_2: $A \supset \neg B \equiv \neg A \vee \neg B$

Step 3:
O_1: $\neg A^1 \vee B^1$
O_2: $\neg A^2 \vee \neg B^2$

Step 4:
Combination 1: $\neg A^1 \wedge \neg A^2$
Combination 2: $\neg A^1 \wedge \neg B^2$
Combination 3: $B^1 \wedge \neg A^2$
Combination 4: $B^1 \wedge \neg B^2$

Step 5:
Combination 1: $\neg A^1 \wedge \neg A^2 = \neg A^3$ (total weight = 3)

Combination 2: $\neg A^1 \wedge \neg B^2 = \neg A^1 \wedge \neg B^2$ (total weight = 3)
Combination 3: $B^1 \wedge \neg A^2 = B^1 \wedge \neg A^2$ (total weight = 3)
Combination 4: $B^1 \wedge \neg B^2 = \neg B^1$ (total weight = 1)

Step 6:
$O_{combined}$: $[(\neg A) \vee (\neg A \wedge B) \vee (\neg A \wedge \neg B)] \equiv \neg A$

Now, to see how the merge results may or may not change according to different ontology weight assignment possibilities, consider the following three cases for the same O_1 and O_2 described above. The merged axiom result ($O_{combined}$), for case 2 and case 3, may be derived using the same methodology described above.

- Case 1: $w(O_1)=1$ and $w(O_2)=2$; has corresponding $O_{combined}$: $\neg A$ (the case presented above)
- Case 2: $w(O_1)=2$ and $w(O_2)=1$; has corresponding $O_{combined}$: $\neg A$
- Case 3: $w(O_1)=1$ and $w(O_2)=1$; has corresponding $O_{combined}$: $\neg A$

Subsequently, if we analyze the merge results produced by Lin's approach for the above example of axiom integration, we will notice the following three issues. First, Lin's methodology yields the following statement as the merge result: $\{\neg A\}$. However, this statement does not represent a 'business rule', but just a simple atom. As such, it seems that if Lin's methodology is applied, as is, to axiom merging, it might not yield the 'desired merging output' (an integrated business rule). In other words, $\{\neg A\}$ does not define the implication if A occurs (if the stakeholder is a subcontractor). Defining the implication of A was the purpose of both original axioms (O_1 and O_2), which the combined statement ($\{\neg A\}$) failed to achieve. The question of what should happen if A occurs, B (authorize the subcontractor to approve change orders) or $\neg B$ (do NOT authorize the subcontractor to approve change orders), remains unanswered. Second, dealing with both axioms under consideration, both stakeholders would (in the general case) like the result of the merge to be $A \supset \neg B$, since the ontology of higher weight supports this statement. But Lin's methodology yields $\neg A$, which does not comply with such 'common sense' requirement. Third, even with different weight assignments, the merge result is the same. The merge result, for the above-mentioned example, does not change despite the change of weights.

The above three points show that while Lin's methodology is suitable for the problem of knowledge base integration, it might not be the case for ontology axiom merging. As such, the following modifications are proposed to Lin's approach to render it more applicable for ontology axiom merging. As mentioned above, the main aim for the subject two rules, ($A \supset B$) & ($A \supset \neg B$), is to define what is the implication of A; in other words, which implication should be selected (B or $\neg B$). To decide on that and subsequently come up with a merged axiom, the antecedents of the rule (the left hand-side part of the rule) should be 'temporarily' added to each conflicting axiom by conjunction. By doing that, it is 'hypothetically' assumed that A is true, in order to discover which implication would have governed, if A had occurred. The following briefly describes the proposed modified merging methodology:

Step 1: Same as Lin's step 1.
O_1: $A \supset B$
O_2: $A \supset \neg B$

Step 2: Add the antecedents of the rule (the left hand-side part) to each axiom by conjunction ('and' operator), as follows:
O_1: $A \wedge (A \supset B)$
O_2: $A \wedge (A \supset \neg B)$

Step 3: Same as Lin's step 2.
O_1: $A \wedge (A \supset B) \equiv A \wedge B$
O_2: $A \wedge (A \supset \neg B) \equiv A \wedge \neg B$

Step 4: Same as Lin's step 3.
O_1: $A^1 \wedge B^1$
O_2: $A^2 \wedge \neg B^2$

Step 5: Same as Lin's step 4.
Combination 1: $A^1 \wedge B^1 \wedge A^2 \wedge \neg B^2$

Step 6: Same as Lin's step 5.
Combination 1: $A^1 \wedge B^1 \wedge A^2 \wedge \neg B^2 = A^3 \wedge \neg B^1$ (total weight = 4)

Step 7: Same as Lin's step 6.
$O_{combined}$: $(A \wedge \neg B) \equiv A \wedge (A \supset \neg B)$

Step 8: Remove the antecedents of the rule, which was previously added in step 2, from the combined axiom to come up with the integrated axiom, as follows:
$O_{integrated}$: $A \supset \neg B$

Similarly, if $w(O_1) = 2$ and $w(O_2) = 1$ this approach results in $O_{integrated}$: $(A \supset B)$. This again shows that, for this example, the modified approach yields 'common sense' merge results. Also, if $w(O_1) = w(O_2) = 1$, this approach will result in $O_{integrated}$: {}. Both sentences will cancel, as both ontologies have equal weight. This might be one of the cases at which user input will be required for accepting the merge result or alternatively deciding to select one of the original two rules.

Similarly, this modified approach could be applied for axioms written in predicate logic. For example, consider the following situation:
O_1: subcontractor(x) $\supset$ is_authorized_to_approve_change_orders(x), while $w(O_1) = 1$
O_2: subcontractor(x) $\supset \neg$ (is_authorized_to_approve_change_orders(x)), while $w(O_1) = 2$
$O_{integrated}$: subcontractor(x) $\supset \neg$ (is_authorized_to approve_change_orders(x))

PROPOSED APPROACH FOR INTEGRATION OF ONTOLOGY AXIOMS

Based on literature review and subsequent analysis, this paper proposes an approach for axiom merging composed of two main steps: 1) establishing mapping relationships between sets of ontology axioms, 2) providing the user with merging recommendations.

AXIOM MAPPING

The main objective of the axiom mapping component is to define relationships between the axioms of two or more ontologies. Suggestions will be based on pre-defined mapping algorithms. Given two sets of axioms, each corresponding to one ontology, A_1 and A_2, four types of mapping relationships will be distinguished between any given two axioms:

1. Equality mapping: two axioms are asserted to correspond to one another, i.e. match 100% in terms of predicates and operators. For example, axioms a_{11} and a_{22} will have an 'equality' mapping designation:
   - Axiom a_{11}: $A \wedge B \supset P$
   - Axiom a_{22}: $A \wedge B \supset P$
2. Concept-subsumption mapping: a concept (or more) of one axiom is a 'child' of a concept of the other axiom. For example, if G is a sub-concept of F and Y is a sub-concept of X, then the following two axioms will have a 'concept-subsumption' type of mapping:
   - Axiom a_{13}: $F \wedge Y \supset R$
   - Axiom a_{24}: $G \wedge X \supset R$
3. Rule-subsumption mapping: one axiom is subsumed by the other axiom. For example, axiom a_{26} will have a 'rule-subsumption' type of mapping in relation to axiom a_{15}:
   - Axiom a_{15}: $K \wedge L \supset S$
   - Axiom a_{26}: $K \wedge L \wedge M \supset S$
4. Inconsistency mapping: the use of both axioms in one ontology would raise an inconsistency in that ontology (i.e. you can derive both a statement and its negation from both axioms). For example, the following two axioms will have a 'inconsistency' type of mapping:
   - Axiom a_{17}: $N \supset T$
   - Axiom a_{28}: $N \supset \neg T$

The output of the axiom mapping step will be used as an input to the axiom merging stage. A more elaborate presentation and description of the mapping methodology will be presented in future work.

AXIOM MERGING

The proposed axiom merging approach is characterized by *consistency*, *majority-rule, preservation of maximal amount of axioms, adaptability, and user-interaction*. *Consistency* ensures that conflicts among the merged set of axioms is resolved, i.e. a statement and its negation cannot be derived from the same set of axioms (Lin 1996). *Majority-rule* means that the view of two stakeholders overrules the view of a third stakeholder, if the views of the three stakeholders (as presented in their own original ontologies) are considered of equal value (Zobel et al. 2005). *Preservation of maximal amount of axioms* indicates that: 1) the merged set of axioms represents the maximal amount of axioms originating from each ontology axiom set, 2) the merged set of axioms differs minimally from each original ontology axiom set (Lin 1996). Since inclusion of maximal amount of axioms may not be desirable, because it may lead to over-axiomatization (i.e. including too many rules in the merged ontology), the merging is not fully automated and is only provided as a suggestion to the user. As such, a user may waive the maximal requirement and change the suggested merged axioms. On the other hand, *adaptability* means that the merging result changes based on the circumstances of collaboration; i.e. the merging approach considers criteria such as level of ontology development, experience level, role of each stakeholder, type of project, type of contract, region, and work environment in suggesting the value of each stakeholder view, and consequently in resolving conflicts among axioms). Note that these values are only suggested, and as such could be changed by the user if desired. Also, these values could be initially defined by the user. Finally, *user interaction* refers to the fact that the merging process is not fully automated. Mergings are only suggested to the user, which could accept the suggestions or modify them, as desired. The axiom merging step takes as input:

1. Two or more sets of axioms, each corresponding to one ontology: A_1 to A_n, where n is the number of ontologies to be merged.
2. A mapping between axiom sets defining how A_1 to A_i are related, as discussed above.
3. Optional assignments of the relative values of each ontology (non-negative numbers).

As such, if $a_{1i} \epsilon A_1$ and $a_{2j} \epsilon A_2$, then merging between a_1i and a_2j is performed based on the previously defined mapping relationships as follows:

1. Eliminating redundancy: One of a_{11} and a_{22} is automatically maintained in the merged ontology, if both axioms are asserted to have 'equality' mapping.
2. Elimination of subsumption: A suggestion is made to the user to merge both a_{13} and a_{24}, if both axioms have 'subsumption' mapping. A merged axiom (combination of a_{13} and a_{24}) is also suggested to the user. The merged axiom eliminates specialized concepts and keeps generalized concepts. For example, if G is a sub-concept of F, Y is a sub-concept of X, and as such the following two axioms are asserted to be related by 'concept-subsumption', then the suggested merged axiom will be:
   - Axiom a_{13}: $F \wedge Y \supset R$
   - Axiom a_{24}: $G \wedge X \supset R$
   - Suggested merged axiom: $F \wedge X \supset R$

   Another example, dealing with 'rule-subsmption' mapping, would result in the following suggested merged axiom:
   - Axiom a_{15}: $K \wedge L \supset S$
   - Axiom a_{26}: $K \wedge L \wedge M \supset S$
   - Suggested merged axiom: $K \wedge L \supset S$
3. Elimination of inconsistency: If a_{17} and a_{28} are asserted to have 'inconsistency mapping' and $w(A_n)$ is the weight of an ontology (which represents the value of such ontology with respect to the others), a suggestion is made to the user to merge both

axioms based on the above-defined three principles: consistency, majority-rule, and preservation of maximal amount of axioms. For example:

- Axiom a_{17}: $N \supset \neg T$, while $w(A_1)= 2$
- Axiom a_{28}: $N \supset T \wedge Z$, while $w(A_2) = 1$
- Suggested merged axiom: $N \supset \neg T \wedge Z$

4. All other axioms (which do not raise a redundancy, subsumption or inconsistency problem) will be maintained in the merged ontology, so that the 'maximal amount of axioms' principle is maintained.

CONCLUSION

This paper discussed knowledge base integration in the context of ontology axiom merging. The paper also proposed an approach for axiom merging composed of two main steps: 1) establishing mapping relationships between sets of ontology axioms, 2) providing the user with merging recommendations. The proposed axiom merging approach is characterized by *consistency*, *majority-rule, preservation of maximal amount of axioms, adaptability, and user-interaction*. Merging recommendations also depend on user-defined criteria, such as level of ontology development, role of stakeholder / organization, type of project, type of contract, region, and work environment.

REFERENCES

Chinowsky, P.S. and Rojas, E.M. (2003). "Virtual Teams: Guide to Successful Implementation." ASCE, *J. of Mgmt in Engrg.*, 19(3), 98-106.

El-Diraby, T.E. and Wang, B. (2005). "E-Society Portal: Integrating Urban Highway Construction Projects into the Knowledge City." ASCE, *J. of Constr. Engrg. and Mgmt.*, 131(11), 1196–1211.

Gruninger, M. and Fox, M.S. (1995). "Methodologies for the Design and Evaluation of Ontologies." *Workshop on Basic Ontological Issues in Knowledge Sharing*.

Kazi, A.S. and Charoenngam, C. (2003). "Facilitating Inter-Enterprise Information Exchange in One-of-a-Kind Settings." *Elec. J. of Info. Tech. in Constr.*, 8, 319-340.

Lin, J. (1996). "Integration of Weighted Knowledge Bases." *Artificial Intelligence*, 83, 363-378.

McGuinness, D.L., Fikes, R., Rice, J., and Wilder, S. (2000). "An Environment for Merging and Testing Large Ontologies." *Proc., 7th Intl. Conf. on Principles of Knowledge Representation and Reasoning (KR2000)*, Breckenridge, Colorado, USA, April 12-15, 2000.

Wang, C., Hong, T., Tseng, S., and Liao, C. (1998). "Automatically Integrating Multiple Rule Sets in a Distributed-Knowledge Environment." *IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews*, 28(3), 471- 476.

Wilson, I., Harvey, S., Vankeisbelck, R., and Samad, A. (2001). "Enabling the Construction Virtual Enterprise: The OSMOS Approach." *Elec. J. of Info. Tech. in Constr.*, 6, 83-110.

Zobel, C.W., Rees, L.P., and Rakes, T.R. (2005). "Automated Merging of Conflicting Knowledge Bases, Using a Consistent, Majority-Rule Approach." *Computers & Operations Research*, 32, 1809-1829.