Proceedings of I-MEDIA '07 and I-SEMANTICS '07
Graz, Austria, September 5-7, 2007

DKP-OM: A Semantic Based Ontology Merger

Muhammad Fahad, Muhammad Abdul Qadir
Muhammad Wajahat Noshairwan, Nadeem Iftikhar
(Mohammad Ali Jinnah University, Islamabad, Pakistan
{fahad, aqadir, wajahat, nadeem}@jinnah.edu.pk)

Abstract: Accurate mapping and merging of multiple ontologies to produce consistent and coherent merged global ontology is a very important process to enable heterogeneous multi-vendors semantic-based systems to communicate and understand each other. Current systems for ontology mapping and merging are very restricted in term of resolving mismatches or proposing accurate matches with no or minimum human intervention. The suggestions made by these systems do not consider all information available in the semantic knowledge of the ontologies. In this paper we are proposing a system for merging and alignment of OWL-DL ontologies that uses semantics of concepts especially the disjoint knowledge in order to resolve conflicts automatically. We show that in order to provide a complete, consistent and coherent merged ontology, preservation of disjoint knowledge in the definition of ontologies is very helpful. Our system uses heuristics-based approach for mapping concepts of ontologies by analyzing description logic (DL) of concepts and preserves the disjoint knowledge in the merged ontology, too. The concept of validation during the initial stages of ontology mapping not only distinguishes our system from the existing ones' but also reduces the users' dependability for validating the consistency of the generated mappings. We also discuss experimental findings to prove the effectiveness of our approach in merging process.

Keywords: Ontology Mapping and Merging, Disjoint Knowledge Preservation, Ontology Engineering, Knowledge Modelling, Semantic Computing
Categories: H.3, H.4.2, I.2.4

1 Introduction

Ontology is one of the important components of a semantic based information processing systems [Gomez-Perez, 04]. Information processed by such systems is modelled in the form of ontology. To enable these heterogeneous multi-vendors systems communicate and understand each other in order to develop a globally compatible semantic based information system, we need to map and merge the ontologies and generate a unique global ontology [Noy, 01]. Ontology creation is mainly dependant on how the ontologist interprets the domain and models the interpretation, thus source ontologies may model the same knowledge in different ways [Klein 01]. Wiederhold [Wiederhold 94] highlights the ontological mismatches between source ontologies that may arise when ontologists model the same knowledge in different ways. Grosso et al. [Grosso, 98] and Bowers et al. [Bowers, 00] narrate linguistic mismatches when integrating source ontologies that were built using different languages. Klein [Klein 01] combines the work of many researchers and categorizes the mismatches between ontologies as language level mismatches and ontological level mismatches. He also mentioned that ontology mapping and merging systems should be capable enough to find these mismatches and resolve them with minimal or no human intervention to produce consistent and coherent merged ontology. Besides these mismatches, we identified sufficient knowledge omission error [Noshairwan, 07] that would also create problems in onotolgy merging process.

Developing ontology mapping and merging algorithms which can resolve the mismatches with minimum human intervention is a challenge for the developers. Many semi-automatic approaches have been proposed for this purpose. These techniques use many features that can be found in ontologies (labels, structures, instances, semantics) to produce initial mappings between source ontologies for human user [GomezPerez, 04]. Many of these systems produce inaccurate mappings because they assume semantically distinct concepts to be the same as they are using linguistic analysis or instance based matching to find similarities between concepts. Therefore, validation of suggestions and resolution of conflicts is totally dependant on human experts.

Pottinger and Bemstein [Pottinger, 03] formally describes the requirements of a merging process based on similarities. Current ontology merging systems follow these requirements such as element preservation, equality preservation, extraneous item prohibition etc. However, we have observed that one of the important requirements of ontology merging has been overlooked. This requirement is the preservation of Disjoint Knowledge. Qadir [Qadir, 07] raises the significance of disjoint knowledge within the ontology and demonstrates that disjoint knowledge omission could be catastrophic in various situations. Therefore, a system has been proposed to detect disjoint knowledge omission and generate warnings for ontologists. In this paper, we provide an ontology mapping and merging approach, which uses the features of the existing approaches and in addition to these features it also incorporates disjoint knowledge preservation mechanism based on description logic (DL). Analyzing disjoint axioms within the ontologies would raise a new class of conceptualization conflict known as alignment conflict among disjoint relations. These conflicts arise when lexically same concepts within the source ontologies contradict each other with respect to their semantics descriptions of disjointness. When the existing systems merge these ontologies, they ignore the restriction of disjointness and the result would be an incorrect merged ontology. Due to use of more semantics provided in the ontologies to be merged, our approach enhances the accuracy of ontology mapping and merging process and also reduces the human intervention.

Rest of the paper is organized as follows: Section 2 discusses related work. Section 3 presents our framework, “DKP-OM”: the architecture, the working of system and the main mechanisms for finding the mappings between concepts. Section 4 contains our experimental findings and initial report on its effectiveness. Section 5 concludes the paper and shows future directions.

2 Related Work

There are many techniques for ontology mapping and merging based on features that are used for finding similarities between concepts such as instances, labels, structures, etc. Instance based techniques like FCA-Merge [Stumme, 01], and IF-Map [Kalfoglou, 03] consider concepts having the same instance as candidate to be merged. The major drawback is observed when semantically distinct concepts having the common instance are considered to be the same. GLUE [Doan, 04] follows the instance-based and multi-strategy machine learning approach that analyses taxonomic structure for ontology integration. It calculates the probabilities of concept matching to combine results of multistrategies by its subcomponents. QOM [Ehrig, 04] adopts many similarity measures and follows a dynamic programming approach to find correspondences between source ontologies. It uses heuristics by only choosing promising candidate mappings and thus reduces the runtime complexity.

The tools PROMPT [Noy, 00], AnchorPROMPT [Noy, 01] and Chimaera [McGuinness, 00] use the similarity of labels and to a certain extent the structure of ontologies. Prompt follows the cyclic process by finding the initial suggestions based on identical labels, synonyms, and superclasses for top-level classes (structural indications) during ontology merging process. It was designed for specialized terminology for medical ontology; hence it produces less accurate mappings for general ontologies. Moreover when semantically same concepts are modelled with different names, it has no ability to find those matchings. ANCHORPROMPT exploits the graph analysis of ontology where classes are considered as nodes and slots as links between them. It analyses the paths in the subgraph limited by the anchors and determines those classes that are frequently appearing in similar positions on similar paths. Chimaera suggests the user to merge concepts by finding pairs having same labels, same prefix or suffix, substrings and acronym usage. ONION [Mitra, 02] exploits the taxonomic analysis, local definition analysis and formalization of articulation ontologies to merge different ontologies. Articulation ontologies contain concepts and relationships expressed as rules. These rules provide links between source domains while ontology merging process. Cupid [Madhavan, 01] integrates the linguistic and structural approaches to compute normalized correspondence coefficients with the assistance of a precompiled thesaurus. OLA [Euzenat, 03, 04] uses Alignment API that finds correspondences based on terminological, structural and extensional approaches for OWL-Lite ontologies. It finds recursive relationships to find the best match through iteration and searches similarities between similar categories i.e. concepts with concepts, attributes with attributes and so on.

Our sytems follows the hybrid approach and uses linguistic matching and semantic-based formal definition analysis to find correspondences between concepts. It follows the OLA [Euzenat 03, 04] by covering all the possible characteristics of OWL ontologies i.e. terminological, structural and extensional and CUPID [Madhavan, 01] approach in a sense of finding linguistic and structural matching strategies. The use of heuristics that guides system to find mappings and resolves conflicts is similar to the constraints followed by GLUE [Doan, 04]. The use of disjoint knowlegde axioms and validation of initial mappings by considering disjoint relations between concepts distinguishes it from rest of the systems and lessens users' dependability for validating the consistency of the generated mappings.

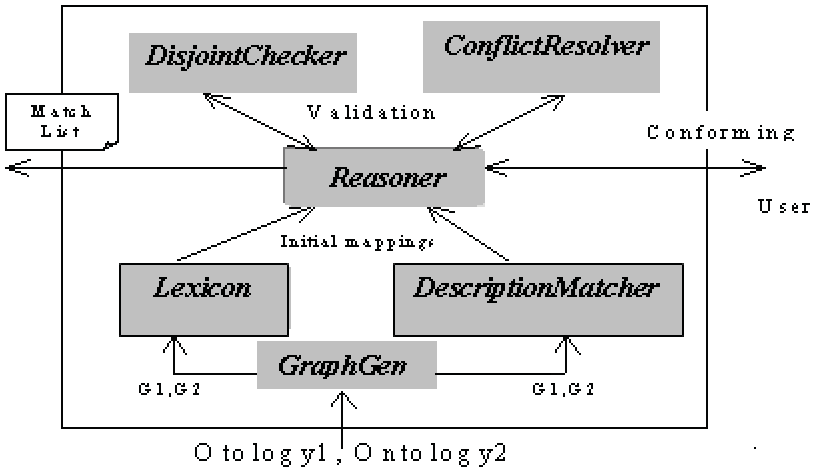

Figure 1: Top-level view of DKP-OM

3 The Design and Working of DKP-OM

Disjoint Knowledge Preserver (DKP) based Ontology Merger (OM) merges ontologies by exploiting semantic knowledge available in the ontologies especially the disjointess of the concepts. Figure 1 shows the top-level view of DKP-OM. In DKP-OM, initial mappings are found by *Lexicon* and *DescriptionMatcher*. *Lexicon* uses string based algorithm and thesaurus lookup to find mappings between concept names. It also does the structural analysis to find correspondences between structural taxonomy between source ontologies. *Lexicon* generally detects (i) concepts having same names, or using same substrings, suffixes, prefixes or acronyms, (ii) concepts having synonyms (same meanings), hyponym less general) or hypernym (more specific) relationships, (iii) concepts having structural similarity with each other i.e. occurrence of concept in the ontologies under consideration, having same parent, siblings, children etc.

DiscriptionMatcher performs the description logic analysis of concepts that reflects semantic similarities by analysing their formal definitions and relations. During analysis it may use the more generic definitions of the concepts in the hirearchy. Generally it detects (i) concepts having same definitions and restrictions or usage of concepts in its definition in intersection or union etc. (ii) concepts having same relationship with other concepts such as parent, child, sibling etc. (iii) overlapping concepts by using heuristics, which increases the chance of correct mappings among them.The probability of correct mappings sent from *DiscriptionMatcher* is greater than the mappings suggested by *Lexicon*. The initial mappings from *Lexicon* and *DiscriptionMatcher* are propagated to the *Reasoner.*

Reasoner does the validation of each mapping found in the previous stage so that the merged ontology stays consistent with reference to the source ontologies. It rejects *lexicon*'s mappings that are linguistically same but semantically different by analysing description logic and then finds one to one correspondence between the concepts of ontologies. Finally it produces consistent, coherent and non-redundant mappings between concepts of source ontologies. The working of *Reasoner* is further explained in the coming sections.

3.1 Reasoning in DKP-OM

Reasoner analyses the concepts to the degree permitted by the formal semantics of the OWL-DL [Gomez-Perez, 04] and validates mappings found by *Lexicon* and *DescriptionMatcher*. It focuses on checking the consistency based on disjoint relationships in merged ontologies. It gets confidence about the accuracy of mappings, which are produced both by *Lexicon* and *DescriptionMatcher*, and detects the conflicting situations caused by wrong mappings. It preserves disjoint knowledge between concepts, and avoids disjoint knowledge omission error. Alignment conflicts among disjoint relations between concepts are detected and suggestions are predicted for conflict resolution.

Reasoner uses *DisjointChecker* to check whether mapped concepts are directly or indirectly (in the hierarchy) disjoint with each other. It rejects the mappings that give rise to conflicting situations. For example consider the ontologies in Figure 2 and assumes that *Lexicon* suggests B and C concepts of ontology 1 are same as the concepts Y and Z of ontology 2, children of B mapped on the children of Y. Assume *Lexicon* also suggests that some child of C is mapped on a child of Y based upon textual similarities. Our disjoint checker would raise an alarm for such mappings because the parents of these two mapped children are disjoint (indirectly) and the mapping would be inconsistent. The conflicts could be resolved either by the conflict resolver or by human expert.

The system has build-in capability to resolve alignment conflicts during ontology merging process with the help of heuristics in the knowledge base as suggested in table 1. *ConflictResolver* checks the knowledge base to get a solution for such conflicts. Some of these heuristics guide the system about the concepts that are candidate for merging but some only predict about their partial mapping by analysing overlapping descriptions. Such overlapping descriptions are useful while analysing mappings came from *Lexicon*. As *Lexicon* only suggest about their linguistic similarity, overlapping heuristics increases the probability and chances of these mapping between concepts to be correct.

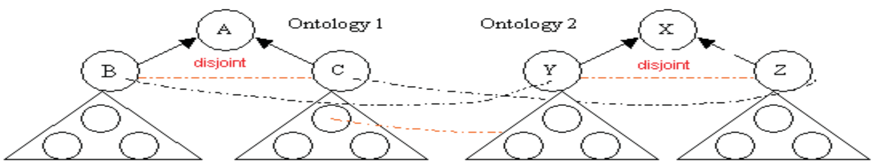

Figure 2: Contradictory Mappings between Concepts of source ontologies

• A concept C_a is made up (union) of n concepts and a concept C_b has same n children, then C_a and Cb are supposed to be candidate for merge.
• If the concept (C_a) is formed by same restriction as restriction in C_b and the type of restriction is some/all value restriction then concepts C_a and C_b are candidate for merge.
• Concept C_a is made up (intersection) of n concepts and a concept C_b has same n children, then the chance of Ca and Cb to be merge increased.
• Concept C_a is union or intersection of some concepts. The concept C_a and C_b are overlapped, if C_b equal to any concept that is used in definition of C_a.
• Concepts C_a and C_b are overlapped; If C_a is sub-concept or super-concept of C_b.
• Concept C_a is based on all value restriction and concept C_b is union of some concepts. The concepts C_a and C_b are overlapped, If the domain concept of restriction is equal or sub concept of C_b or any concept that is used in definition of $C_{b.}$
• Concept C_a is based on maximum or minimum cardinality restriction and concept C_b is union of some concepts. Concepts C_a and C_b are overlapped, If the domain concept of restriction is equal or sub concept of C_b or any concept that is used in definition of C_b.
• Concept (C_b) is formed by same restriction as restriction in C_a and the type of restriction is minimum/maximum cardinality restriction then the concepts C_a and C_b are overlapped.
• Concept (C_b) is formed by same restriction as restriction in C_a and the type of restriction is some/all value or minimum/maximum cardinality restriction then the concepts C_a and C_b are overlapped.

Table 1: Examples of Heuristics

Reasoner suggests the mappings that preserve disjoint knowledge between concepts in merged ontology as they are in source ontologies. While preserving disjoint knowledge between concepts, some conflicts may occur. Resolution criteria for such conflicts should be asked from the user by giving all the possible suggestions. Consider a similar situation with a slight modification to the situation in Figure 2, by removing disjoint knowledge between Y and Z concepts as shown in Figure 3. From the Figure we come to know that B and C are mapped to the Y and Z concepts. Furthermore concept G1 that is a child of Y is mapped onto a child of C. Such conflicts can be resolved automatically by the *Reasoner*. *Reasoner* checks whether concept G or any child of G overlap with concept B or with children of B. If they do not overlap then *Reasoner* accept that mapping to be true and suggest user to map Y to B, and extract subconcepts hierarchy G to be merged with subconcepts hierarchy of C. But if they overlap then it extracts only overlapping concepts of G to be merged with subconcepts of B.

[1] consider G as concept *student* in ontology 2. Due to semantic heterogenity some of the *students* (children of G) are employeed and some are not. In ontolgy 1 *student* is a subchild of C (unemployeed person) and disjoint with B (employees). *DisjointChecker* highlights this situation and *ConflictResolver* suggests employeed *students* to be kept under B and unemployeed *students* under C.

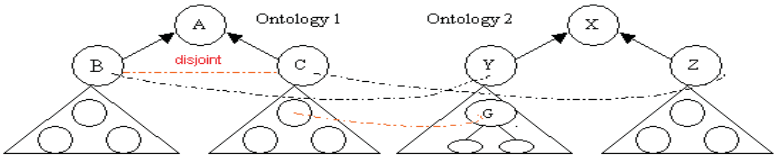

Figure 3: Mappings between Concepts

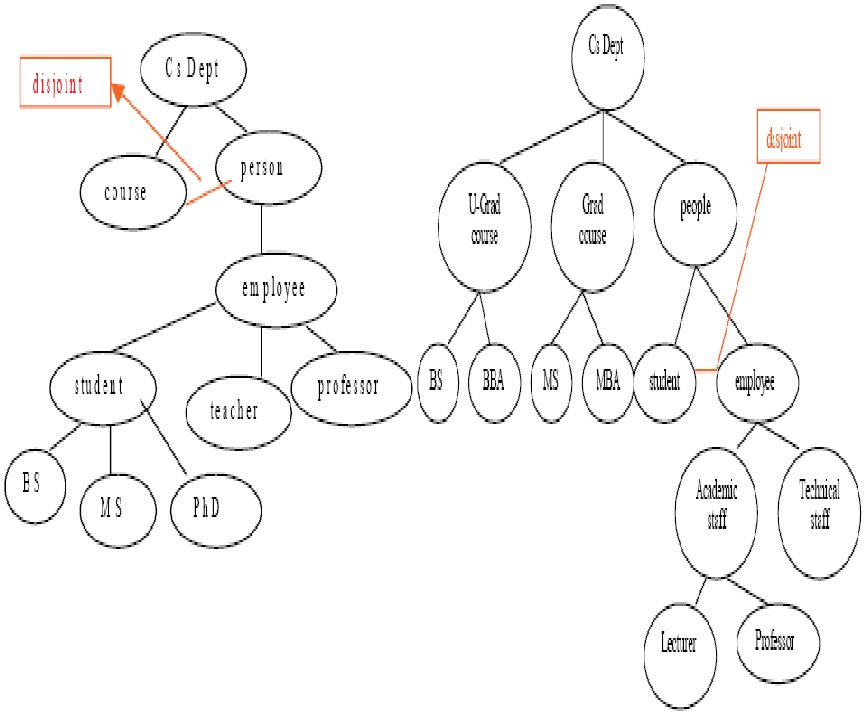

Figure 4: a) University Ontology A b) University Ontology B

4 Evaluation

We have evaluated DKP-OM on several real-world domains to check its matching accuracy and effectiveness. We performed an experiment with DKP-OM by giving two university ontologies having disjoint-axioms. Due to space limitations we only show top-level view of both ontologies in Figure 4 and discuss our high level findings. Suggestions by DKP-OM are represented in Figure 5a. The conflict arises during the merging is shown in Figure 5b. Inaccurate mappings that were rejected are shown in Figure 6. The alignment conflict occurs while mapping the concept *student* of ontology A, with *student* of ontology B. As an instance of only concept *PhD student* can be an *employee* in ontology A but no instance of *student* can be the instance of *employee* as they were made disjoint in ontology B. DKP-OM prompts the user to take the decision for conflict resolution whether to preserve disjoint knowledge as in ontology B or preserve parent child relationship as in ontology A. In the language of the example considered here, one is to preserve disjoint knowledge between *student* and *employee* as in ontology B with children *BS* and *MS*, and extract *PhD* concept and make subconcept of *employee*. Second is to ignore disjointness between *Student* and *Employee* and make *PhD* concept as a child of both concepts by multiple inheritance.

Prediction	C1	C2
similar	Cs Dept	Cs Dept
hyponym	course	Ugrad Course
hyponym	course	Grad course
synonym	person	people
similar	student	student
similar	employee	employee
similar	professor	professor
similar	teacher	lecturer

Fig. a

Conflict	Student	Student
	Employee	Employee
Suggestions	Preserve disjoint knowledge between Student and Employee Maintain BS and MS as children of Student Extract PhD concept and made subchild of employee.	
	Remove disjointness between Student and Employee Make PhD concept as a subchild of both concepts Student and Employee are overlapping concepts	

Fig. b

Figure 5: Mapping results– a) Mappings between source ontologies b) Suggestions for Alignment Conflict in disjoint relations

Rejected Prediction	C1	C2
Linguistic Similarity	BS	BS
Linguistic Similarity	MS	MS

Figure 6: Rejected Mappings

5 Conclusion and Future Work

This paper presents the DKP-OM system that exploits linguistic matching and semantic-based formal definition analysis to find correspondences between concepts to improve accuracy of mappings. DKP-OM preserves disjoint knowledge in merged ontology, thus avoids disjoint knowledge omission error [Qadir, 07]. It checks the consistency of the merged ontology by taking disjoint knowledge axioms into consideration. The system is capable of finding alignment conflicts among disjoint relations and resolving these conflicts by using set of heuristics or user intervention. Our General-purpose heuristics and description logic analysis make the system work with different contexts, domains and subject ontologies. Domain specific heuristics[1] about disjoint knowledge in source ontologies minimize the search space while finding correspondences between concepts and thus reduce the runtime complexity. However domain specific heuristics can only be applied on well-known ontologies where we do not expect any alignment conflict within the subhierarchies of disjoint concepts.

Our ongoing research direction on this topic is to conduct empirical studies to assess the effectiveness of using disjoint knowledge axioms in ontology merging and to compare our system with other ontology merging systems. At the same time, we will further enhance the system based on some heuristics that lead towards consistent merged ontology with avoidance of higher level of user intervention during ontology merging process. We will further enhance the system for dynamic extraction of mapped code chunks to produce automatic merged ontologies.

References

[Bowers, 00] Bowers, S., Delcambre, L.: "Representing and Transforming model-based information", In First Workshop on the Semantic Web at the Fourth European Conference on Digital Libraries, Lisbon, Portugal(2000), 5-18.

[Doan, 04] Doan, A., Madhaven, J., Domingos, P., Halevy, A.: "Ontology matching: A machine learning approach", Handbook on Ontologies in Information Systems, Springer-Verlag,(2004), 397-416.

[Ehrig, 04] Ehrig, M., Staab, S.: "QOM - quick ontology mapping", In Proc. of the Third International Semantic Web Conference (ISWC2004), Springer, LNCS 3298(2004), 683–696.

[Euzenat, 03] Euzenat, J. and Valtchev, P.: "An integrative proximity measure for ontology alignment", In Proc. of the 1st Intl. Workshop on Semantic Integration(ISWC-2003), Sanibel Island, USA(2003), 33–38.

[Euzenat, 04] Euzenat. J. and Valtchev, P.: "Similarity-based ontology alignment in owl-lite", In Proc. of *16th* ECAI-04, Valencia, Spain(2004), 333–337.

[Gomez-Perez, 04] Gomez-Perez, A., Fernández-López, M., Corcho,O.: "Ontological engineer-

ing: With examples from the Areas of Knowledge Management, E-Commerce and the Semantic Web"; Springer, London(2004)

[Grosso, 98] Grosso, W., E., Gennari, J., H., Fergerson, R., W., Musen, M., A.: "when knowledge models collide (how it happens and what to do)", In Proc. of the 11th Workshop on Knowledge Acquisition, Modeling and Management (KAW '98), Banff Canada (1998), 18-23

[Kalfoglou, 03] Kalfoglou, Y., Schorlemmer, M.: "If-map: an ontology mapping method based on information flow theory", Journal of data semantics, Springer Berlin / Heidelberg, LNCS 2800(2003), 98–127.

[1] For example consider family ontology O_a where Male concept is disjoint with Female concept and ontology O_b where Men concept is disjoint with Women concept. If we get a top level mapping of concept Male into Men and Female into Women, then search space would be reduced by only seeking mapping of subchildren concepts of Male into subchildren concepts of Men only and vice versa rather than in all the taxonomy of concepts.

[Klein, 01] Klein, M.: "Combining and relating ontologies: an analysis of problems and solution", In Proc. of Workshop on Ontologies and Information Sharing (IJCAI-01), Seattle, USA(2001), 53-62.

[Madhavan, 01] Madhavan, J., Bernstein, P., Rahm, E.: "Generic schema matching using Cupid", In Proc. of the 27th Intl. Conference on VLDB, Roma, Italy(2001), 49–58

[McGuinness, 00] McGuinness, D., L., Fikes, Rice, J., Wilder, S.: "An environment for merging and testing large ontologies", In Proc. of the 7th Intl. Conference on Principles of Knowledge Representation and Reasoning, Breckenridge, CO, USA(200), 483–493.

[Mena, 00] Mena, E., Illarramendi, A., Kashyap, V., Sheth, A., P.: "OBSERVER: An approach for query processing in global information systems based on interoperation across preexisting ontologies", International Journal DAPD, 8, 2 (2000), 223-271.

[Mitra, 02] Mitra, P., Wiederhold, G.: "Resolving Terminological Heterogeneity in Ontologies," In Proc. of Workshop on Ontologies and Semantic Interoperability at the 15th ECAI, Lyon, France (2002), 45-50.

[Noshairwan, 07] Noshairwan, W., Qadir M., A., Fahad, M.: "Sufficient Knowledge Omission error and Redundant Disjoint Relation in Ontology", InProc. 5th Atlantic Web Intelligence Conference, Fontainebleau, France (June 25-27, 2007)

[Noy, 00] Noy, N. F., Musen, M. A.: "The PROMPT suite: interactive tools for ontology merging and mapping", IJHCS, Elsevier, 59, 6 (2003), 983–1024.

[Noy, 01] Noy, N. F., Musen, M. A.: "Anchor-PROMPT: Using nonlocal context for semantic matching", In Proc. of Workshop on Ontologies and Information Sharing (IJCAI-01), Seattle, USA(2001), 63–70.

[Pottinger,03] Pottinger, R., Bernstein, P.: "Merging Models Based on Given Correspondences" In Proc. of 29th Intl. Conference on VLDB'03, Berlin, Germany (2003), 862-873.

[Qadir, 07] Qadir, M., A., Noshairwan, W.: "Warnings for Disjoint Knowledge Omission in Ontologies" In Proc. of 2nd ICIW'07, Morne, Mauritius (2007), 45-45

[Stumme, 01] Stumme, G., Mädche, A.: "FCA-merge: bottom-up merging of ontologies", In Proc. 17th IJCAI '01, Seattle, WA, USA (2001), 225–230.

[Wiederhold, 94] Wiederhold, G.: "An algebra for ontology composition", In Proc. of Monterey Workshop on Formal Methods, U.S. Naval Postgraduate School, Monterey CA (1994), 56–61