Towards a rule-based matcher selection

Malgorzata Mochol[1] and Anja Jentzsch[1]

Freie Universität Berlin, Königin-Luise-Str. 24-26, D-14195 Berlin, Germany
mochol@inf.fu-berlin.de, mail@anjajentzsch.de

Abstract. The central problems w.r.t. interoperability and data integration issues in the Semantic Web are schema and ontology matching approaches. Today it takes an expert to determine the best algorithm and a decision can usually be made only after experimentation, so as both the necessary scaling and off-the-shelf use of matching algorithms are not possible. To tackle these issues, we present a rule-based evaluation method in which the best algorithms are determined semi-automatically and the selection performs prior to the execution of an algorithm.

1 Introduction

In recent decades an ever-growing number of ontologies have been developed and deployed in numerous fields like knowledge management, information retrieval, multimedia, Web Services, and the Semantic Web (SW). Considering the decentralized nature of the SW development and since SW envisions a Web of machine-understandable information which can be automatically accessed by and exchanged among semantics-aware applications, an exponential growth in the number of ontologies and ontology-based applications is to be expected. Many of these ontologies will describe similar domains, though using different terminologies, since there will always be more than one representation of any domain of discourse [16], while other ontologies will have overlapping domains. The dissemination of ontologies across the various research communities has simultaneously generated an emerging plethora of tools and methodologies to build, maintain and manage, merge, map, and match ontologies. Among these methods, matching algorithms occupy the key role in facilitating the overall success of the SW. They are involved in almost every phase of an ontology engineering and management process and are crucial to interoperation and interoperability between Web applications using different but related ontologies [11]. Additionally, a fundamental requirement for the realization of the SW vision is the use of tested and proved ontology matching algorithms capable of dealing with the *heterogeneity of ontological sources* available on the Web in terms of representation and natural languages, varying degrees of maturity and granularity levels, and divergent views of the modelled domains. The importance of the matching issue is also reflected by the large number of matching algorithms [6, 7, 10, 13, 17, 18, 22, 23] and the variety of the adaption of such algorithms in various areas of computer science like Web Service, P2P or Grids [3]. As stated in [8], in the upcoming years, we can expect work on matching issues not only to continue but to increase mostly because of the growing interest in the semantic heterogeneity tasks from both research and industry. In spite of all these gains and activities, current matching algorithms cannot be optimally used in ontology matching tasks as envisioned by the SW community, mainly because of the inherent dependency between approaches and ontological properties. In this paper, we introduce a possible solution to this issue – a Metadata-based Ontology MAtching (MOMA) framework based on a reuse-paradigm that, when given a set of ontologies to be matched, takes into account the capabilities of existing matching algorithms and suggests appropriate matchers for application. A matching framework requires a highly adaptive selection of available matching services capable of taking full advantage of the broad spectrum of ontologies found across the network.

The remain of the paper is organized as follows: Through analyzing the open issues within the ontology matching domain we identify the main requirements, which a matcher selection framework has to satisfy (Sec. 2). Sec. 3 introduces the high level architecture of the proposed framework while Sec. 4 specifies the framework's Knowledge Base. The matcher selection process and the framework evaluation together with the related work are described in Sec. 5 and 6, respectively. The paper is closed with an outlook on the future activities (Sec. 7).

2 Matcher Selection - Requirements

We have based the analysis of the framework requirements on the main open issues within ontology matching domain:

1. Due to the heterogeneity of the existing ontological sources and the diversity of applications in which matchers are to be applied, it is generally known that an *overarching ontology matching algorithm* capable of serving every ontology type, delivering the desired output, and dealing with excessive application requirements *will not be realized in the foreseeable future*. We have recognized the need for a strategy that strives to optimize the matching process whilst aware of the inherent dependencies between matchers, their execution characteristics, required output, and the types of ontologies they are able to process.

2. We have had to tackle the *issues of heterogeneity and multiplicity of existing ontology matching approaches* and to support the researchers and developers interested in the subject of matching by giving them the chance to get an overview of the still growing collection of algorithms.

3. We have noticed that developers of ontology-based applications traditionally "reinvent" matching approaches, coding new algorithms rather than reusing existing components. Since software reuse can enhance the quality of components and the productivity of developers, and reduce development time [14], we have recognized the potential of this strategy in the matcher context.

4. What is also required are *techniques and tools capable of dealing with different ontological sources* [2] and satisfying the requirements of emerging applications. Since matcher users, e.g researchers/developers, need support in choosing appropriate matchers for their particular application, we have had to facilitate selection and, in turn, access to and use of existing matchers.

5. Further requirement is to consider, rather than ignore, the *problems of the different matchers*. In particular, to tackle the weaknesses of the matchers, there is a need for a tool/framework that exploits the valuable ideas embedded in current approaches, while at the same time takes into account their limitations.

Concluding, the situation in the ontology matching domain requires a highly flexible selection of available matching services in order to take full advantage of the broad spectrum of ontologies and ontology matching algorithms across the network and to satisfy the various requirements of the users.

3 Architecture

Matcher users consult the MOMA Framework for recommendations regarding matchers adequate for their applications. This applies to both humans, e.g. ontology engineers looking for means to compare ontological sources, and machine users seeking automatized methods e.g. to generate mediation ontologies. In order to (semi-)automatically decide which algorithm suits which ontological input, information regarding the approach and the input sources must be linked together. Hence, our MOMA uses (semantical) descriptions of both matching algorithms and ontologies, which are then joined by means of rules (cf. Fig. 1).

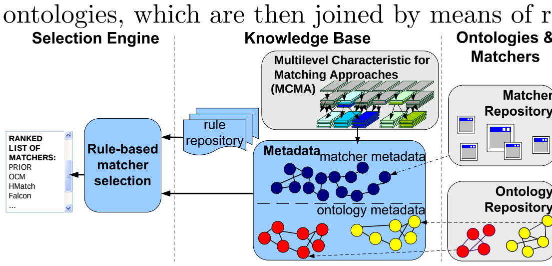

Fig. 1. Architecture of the (semi-)automatic MOMA Framework

Ontologies & Matchers consists of an *ontology repository* (ontologies to be matched) and a *matcher repository* (approaches which will be evaluated for their suitability w.r.t. the given ontologies).

Knowledge Base (KB) - Since it is important to recognize cross application needs and define a matcher characteristic that allows comparison of different approaches and the subsequent selection of suitable algorithms. Specifically, we have collected the various features of matching approaches (input, output, etc.) and targeted application, identified those that have an impact on the selection of appropriate matching approaches, and built a matcher characteristic that serves as the basis for the decision regarding suitability issue. The development resulted in a hierarchical tree of features called *Multilevel Characteristic for Matching Approaches* (*MCMA*)[1]. Furthermore, the KB includes a description of available matching algorithms (*matcher metadata*) that may be selected for application and the metadata of ontological sources (*ontology metadata*) to be matched. Since the ontology metadata is not provided along with the ontological

[1] For detailed information regarding MCMA, the reader is referred to [19]

sources, our MOMA generates automatically the description using information from the particular ontology and additional sources. As we are unable to generate the matcher metadata automatically, our framework considers only matchers which are available in the system and have been annotated manually. The KB is completed by a predefined set of rule statements stored in the *rule repository* that describes the dependencies between the matchers and ontological sources.

Selection Engine, the core of the MOMA, is responsible for the selection of algorithms applicable for a specific input. The matcher and ontology metadata allow the engine to automatically compare the characteristic of the input with the constraints of the available algorithms and, by means of rules, detect the most suitable algorithms which can best deal with the particular ontologies.

4 Metadata and rules

Our MOMA uses additional information about ontologies (ontology metadata) and matching approaches (matcher metadata) in order to determine which of the latter are appropriate in a given context. To ensure a rich and formal representation of the semantics of the metadata we have modelled this information in ontological form (according to established ontology engineering methodologies [9]) and implemented it using OWL.

4.1 Matcher Metadata

Since the suitability of the given matching approaches is determined with careful consideration of a number of factors relevant for the selection, the matcher metadata is intended to capture information about existing ontology matchers. To specify the contents of the metadata model we have applied our MCMA: since MCMA was developed in a systematic way by the utilization of (i) analysis of the requirements collected during the development of various SW-based applications, (ii) analysis of the literature, (iii) collaborations with ontology/software engineers and experts in ontology matching, (iv) our findings in the context of different case studies in ontology engineering, it was a matter of course to consider MCMA as a basis for our matcher metadata. Furthermore, in order to collect data regarding existing matchers, which will be considered within the selection process, we have developed a questionnaire based on MCMA. The survey allows us to analyze existing matchers, collect data relevant to the matcher selection and deliver the weighting of the matchers w.r.t. the particular feature.

4.2 Ontology Metadata

Not only the matching approaches but also the incoming sources need to be described in the corresponding metadata. For this purpose, we have applied the information model described in [20, 12], which is a contextual model for SW resources. In the following we sketch the individual feature categories and identify those, which are relevant to the matching issue:

syntactic features offer quantitative and qualitative information about the ontology and its underlying topology. The quantitative features such as the number of specific ontological primitives have an impact on the matcher selection since the input size affects the matching performance and quality.

semantic features are related to the formal semantics of the representation language and the meaning of the ontology content. These features play a crucial role in matcher selection since they directly affect the matcher execution, e.g. features such as formality level restrict the number of applicable matching algorithms because the matcher may handle only a particular formality level.

heuristic and pragmatic features refer to authoring and historical data of an ontology (e.g. when, by whom, for what purpose it was developed). Such features are not crucial to the matcher selection, however, in the later development, we could imagine to analyze e.g. version or the engineering process issues.

4.3 Rule repository

For ontologies to be matched, the selection engine must decide which matchers satisfy the requirements and obtain the desired outputs. The engine is aware of the background information detailing the available approaches and the input properties. However, in order to automatically infer which algorithms suit the concrete inputs, it needs explicit knowledge concerning the dependencies between these algorithms and the sources on which they can operate successfully. We have formalized this knowledge in terms of *rules-statements* that determine which elements (which matchers) fulfill the rule conditions. The rules[2] are the result of analyzing publications within the ontology matching domain and collaborations with experts, and have been empirically confirmed in some projects [1, 21].
mandatory rules decide whether ontologies are appropriate for matching:

- *domain rules* Match ontologies only if they describe similar domains.
- *natural language* Match ontologies if they use the same natural language.

selection rules evaluate the matchers in order to find suitable approaches for the given input (examples):

- *no instances* If ontologies have no instance data, matchers capable of dealing with scheme can be applied
- *representation language* Only matchers capable of dealing with the representation language of the incoming sources can be applied (e.g. if the sources are formalized in OWL only matchers which can handle OWL can be applied)
- *formality level* Apply only matchers which can deal with the formality level of the incoming sources
- *input size - concepts* Consider only matchers which can deal with the given amount of concepts (e.g. if the incoming sources have over 500 concepts each, suitable matchers are those which can match this amount of concepts)

[2] For implementation purposes, MOMA rules have been implemented in SWRL, a rule language for the SW; `http://www.w3.org/Submission/SWRL/`

5 Framework in action

After the elaboration of the requirements and the description of the framework components, we concentrate mainly on the MOMA functionality(cf. Fig. 2).

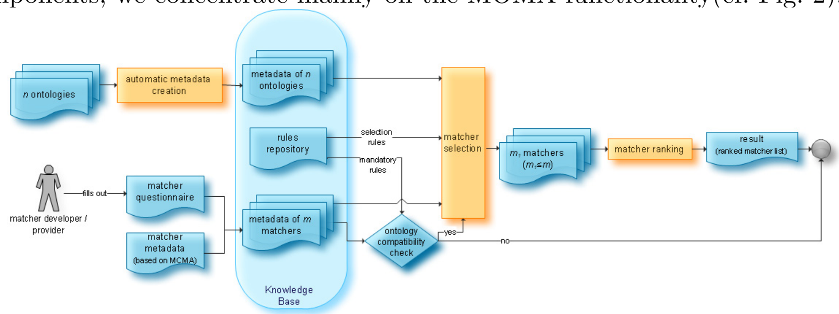

Fig. 2. Matcher evaluation process - Flow chart

Metadata extraction/generation: Since it is impossible to automatically generate the matcher metadata, the approaches must be annotated manually using the MCMA-based online survey and so collected data has been converted into instances of the matcher metadata. To be able to evaluate the matching approaches, the framework needs also additional data regarding incoming sources. The ontologies to be matched are loaded into the system, whereby MOMA automatically generates the ontology metadata utilizing the OntoMeta [4] approach.

Rule execution: As both ontology and matcher metadata have been generated and stored in a Sesame repository along with the SWRL selection rules, the framework creates SeRQL queries on the repository. For this purpose, we have integrated into the framework the SWRL Engine[3] conducting reasoning SWRL rules in RDF graphs and adopted it for our purposes by augmenting the OWL support. Having both matcher and ontology metadata with rules set in KB and the extended SWRL Engine, the framework is ready for the crucial step – rule execution – which (when rule body = true) links two ontologies (in the case of mandatory rules) or ontology with matcher (in the case of selection rules). The rule execution starts with the conversion of rules into SeRQL queries to select the variables defined in the rule's body and place the rule conditions into the `SELECT`'s `WHERE` part. Based on the query's set of results, the framework inserts new triples defined in the rule's head into the Sesame repository (cf. Ex. A, B):

A: SWRL rule tests whether the given ontologies, use the same nat. language:

```
meta:hasNaturalLanguage(?ontology1, ?language) AND meta:hasNaturalLanguage(?ontology2,?language)
-> matcherIsCompatibleConcerningNaturalLanguage(?ontology1,?ontology2)
```

B: SWRL rule tests whether a matcher can handle formal ontologies:

```
matcher:Informal(?matcher, ?formal) AND swrlb:greaterThanOrEqual(?formal,"4") AND
meta:hasFormalityLevel(?ontology, meta:formal)
-> matcherIsCompatibleAccordingFormalityLevel(?matcher, ?ontology)
```

Since each property created by the SWRL Engine between a matcher and the ontologies (on the basis of the selection rules) means that the matcher conforms

[3] `http://www.ag-nbi.de/research/swrlengine/`

to this particular condition, the framework allocates the weighting (using our survey scale from 0 to 8 with 8 being the best score) to the analyzed matcher thus increasing the suitability level of the particular approach. In the next step, the framework ranks the remaining m_1 matchers[4] according to their suitability regarding the particular set of ontological sources; an outcome of the MOMA selection engine is a ranked list of matching approaches.

6 Evaluation

The MOMA evaluation was performed in two phases: the first step, which was dedicated to the overall relevance of the MCMA (cf. Sec. 3), was carried out using the expert judgment method, while the second was performed regarding the reliability of MOMA predictions in the context of real-world situations and was conducted by applying the case study method. Since the goal of case study is to obtain as a complete picture as possible of what and why is going on in an instance [5], and the most critical issue is the appropriate instance selection, we need representative cases for our tests. We found such cases in the context of the OAEI[5], which aims to establish a consensus for the evaluation of alignment approaches by setting up an evaluation campaign and benchmark tests to assess the strengths and weaknesses of the existing approaches. In the course of the campaigns the participants sample their matchers on the test cases provided by the contest organizers, who at the end evaluate the matchers based on their suitability for a particular case. In the following we report on the evaluation conducted on the basis of two test cases from the OEAI2006[6]:

anatomy test case covers the domain of body anatomy and consists of *two ontologies* with an approx. size of several 10k classes and relations: the Foundational Model of Anatomy[7] and the OpenGalen Anatomy Model[8]

food test cases consists of AGROVOC[9] (terminology of agriculture, forestry, fisheries, food and related domains) and the NAL Agricultural Thesaurus[10] (vocabulary reference tool for agricultural and biological terms).

For each of the incoming sources our MOMA Framework automatically generated the ontology metadata (FMA and OpenGalen metadata for the anatomy test case and AGROVOC and NAL metadata for food test case). In the next step the metadata for the matching approaches, which are available in the system (i.e. matchers participated in the OEAI2006), was created on the basis of the online survey filled out by each matcher developer and evaluated the matcher domain expert. Having the matcher and the ontology metadata for each test case, the

---

4 In the course of matcher selection the number of suitable matchers (m_1) may be reduced in comparison to the preliminary candidates number ($m_1 \leq m$).

5 *Ontology Alignment Evaluation Initiative*; `http://oaei.ontologymatching.org`

6 By the time of writing matcher metadata for OAEI2007 has not been completed.

7 `http://sig.biostr.washington.edu/projects/fm/AboutFM.html`

8 `http://www.opengalen.org/`

9 `http://www.fao.org/aims/ag_intro.htm`

10 `http://agclass.nal.usda.gov/agt/agt.shtml`

selection engine fired the rules from the repository. As the mandatory rules were satisfied by the ontological sources, the framework evaluated (by considering the selection rules) and then ranked (by cumulating the weighings) the matchers according to the requirements based on the particular sources.
Results - anatomy case: We have observed that the results of the MOMA selection process performed prior to the execution of a matching algorithm are very similar to the results achieved by the OAEI2006 (cf. Tab. 1).

Results - Anatomy test case				
OAEI Campaign 2006			**Rule-based evaluation methodology**	
Approach	**Terms mapped**	**Ranking**	**Approach**	**Ranking**
NIH (AOAS)	2966	1	-	-
ISLab HMatch	2963	2	PRIOR	1
PRIOR	2590	3	ISLab HMatch	2
Falcon	2204	4	Falcon	3
-	-	-	...	...
-	-	-	NIH (AOAS)	7

Table 1. Anatomy task: OAEI 2006 Campaign vs. MOMA-based results

The only major difference occurs in the NIH (AOAS): although during the contest the NIH (AOAS) approach had the highest number of mappings along with numerous mappings, incl. a significant amount that corresponded with those that were also found in other systems, it achieved further place in the ranking delivered by our framework. However, if we take into account that, during the OAEI 2006, the NIH approach had a significant number of mappings not found by any other system, this might be responsible for the different findings.
Results - food case: In our opinion, if we consider that from eight OAEI2006 matching approaches available in the MOMA system, the first four in our ranking are the same that achieved the best results in the campaign, the MOMA outcomes are relatively similar to results achieved by the OAEI2006 (cf. Tab. 2).

Results - Food test case				
OAEI Campaign 2006			**Rule-based evaluation methodology**	
Approach	**Precision/Recall**	**Ranking**	**Approach**	**Ranking (Points)**
Falocn	0.83 / 0.46	1	-	-
RIMOM	0.81 / 0.50	2	-	-
PRIOR	0.71 / 0.45	3	PRIOR	1
ISLab HMatch	0.61 / 0.46	4	ISLab HMatch	2
-	-	-	RIMOM	3
-	-	-	Falcon	4
-	-	-	AUTOMS	5
-	-	-	NIH (AOAS)	6
-	-	-	DSSim (MAOM)	7
-	-	-	OWL CtxMatch	8

Table 2. Food task: OAEI 2006 Campaign vs. MOMA-based results

Furthermore, since MOMA merely aims to suggest potentially suitable matchers for a given set of ontological sources, we do not have to achieve *exactly* the same ranking results as in the case of ranking based on the real execution of matchers; we aim to deliver a sort of recommendations for the future analysis.

6.1 Related work

As far as we know only one similar to the MOMA Framework solution has been developed: OntoMas – Ontology Mapping Assistant [15][11] developed in Inter-

[11] http://www.polytech.univ-nantes.fr/ontomas/

operability Research for Networked Enterprises Applications and Software (INTEROP) NoE is a didactic tool that assists in and teaches the implementation of matching process. Its matching tools classification is based on the classification presented in [8]. However, the description of the approaches is much more rudimentary than the MCMA characteristic. The basic description of the OntoMas considers only the syntactic aspects of the input ontologies which is motivated by the fact that the tool must be usable by novice users.

7 Conclusion

We have presented a (metadata- and rule-based) framework that evaluates and ranks matching approaches according to their suitability for a given set of ontological sources. With our MOMA Framework we intend to contribute to the tackling of real-world challenges that are commonly agreed upon testbeds and benchmarking with the aim of ensuring seamless interoperability and integration of the various SW technologies. Furthermore, we wish to reduce the number of candidates that can be considered for application and recommend matchers which should be the main focus within the reuse strategy for the given ontological sources. Regarding the requirements defined in Sec. 2 the MOMA:

- contributes to the optimization of the matching process by taking a first step to the propagation of the reuse paradigm in the ontology matching domain (req. 1) and to the exploitation of the existing matching approaches (req. 3) by proposing a solution based on the selection of approved algorithms;
- tackles the issues of heterogeneity of existing ontology matchers (req. 2) and supports developers by providing “recommendations” regarding suitable matchers and giving them the chance to adopt matchers that have already been tried and tested in particular domains/applications/tasks (req. 4).
- exploits the valuable ideas embedded in current algorithms in order to find and, in turn, apply the existing approach in related context (req.4 & 5).

The future work we will be dedicated to: the development of a web service-based MOMA access, the automatical extraction of matcher metadata and, in turn, further collection of matcher data as well as the extension of the rule repository considering further syntactic (e.g. average path length) and heuristic features.
Acknowledgements: This work has been partially supported by the "InnoProfile - Corporate Semantic Web" project funded by the German Federal Ministry of Education and Research (BMBF) and the BMBF Innovation Initiative for the New German Länder - Entrepreneurial Regions.

References

1. C. Bizer, R. Heese, M. Mochol, R. Oldakowski, R. Tolksdorf, and R. Eckstein. The Impact of Semantic Web Technologies on Job Recruitment Processes. In *Proc. of the Int. Conf. Wirtschaftsinformatik*, pages 1367–1383, 2005.
2. S. Castano, A. Ferrara, and S. Montanelli. Methods and Techniques for Ontology-based Semantic Interoperability in Networked Enterprise Contexts. In *Proc. of the Workshop On Enterprise Modelling and Ontologies for Interoperability*, 2004.

3. S. Castano, A. Ferrara, S. Montanelli, and G. Racca. Matching techniques for resource discovery in distributed systems using heterogeneous ontology descriptions. In *Proc. of the Int. Conf. on IT: Coding and Computing (ITCC'04)*, pages 360–366. IEEE Computer Society, 2004.
4. Yao Chen. Automatic Generation of Ontology Metadata. Technical report, Diploma-Thesis, Freie Universitt Berlin, April 2006.
5. L.-E. Datta, L. G. Morra, and A. C. Friedlander. *Program Evaluation and Methodology Division. Case Study Evaluations.* US General Accounting Office, 1990.
6. H.-H. Do and E. Rahm. COMA-a system for flexible combination of schema matching approaches. In *Proc. of the 28th Intl. Conf. on Very Large Databases*, 2002.
7. A.-H. Doan, J. Madhavan, P. Domingos, and A. Halevy. *Handbook on Ontologies*, chapter Ontology Matching: A Machine Learning Approach, pages 385–516. Springer, 2004.
8. J. Euzenat and P. Shvaiko. *Ontology matching*. Springer, 2007.
9. M. Fernandez-Lopez and A. Gomez-Perez. Overview and analysis of methodologies for building ontologies. *Knowledge Engineering Review*, 17(2):129–156, 2002.
10. F. Giuchiglia and P. Shvaiko. Semantic Matching. *The Knowledge Engineering Review (KER) Journal*, 18(3):265–280, 2004.
11. W. Hu, G. Cheng, D. Zheng, X. Zhong, and Y. Qu. The Results of Falcon-AO in the OAEI 2006 Campaign. In *Proc. of the Int. Workshop on Ontology Matching colloc. with the ISWC-2006*, 2006.
12. J. Hartmann, E. Paslaru Bontas, R. Palma, and A. Gmez-Prez. DEMO - A Design Environment for Metadata About Ontologies. In *Proc. of the ESWC2006*, 2006.
13. J. Madhavan, P. A. Bernstein, and E. Rahm. Generic Schema Matching with Cupid. In *Proc. of the 27th VLDB Conference*, pages 48–58, 2001.
14. M. A. Lemley and D. W. O'Brien. Encouraging software reuse. *Stanford Law Review*, 49(2):255–304, 1997.
15. M. Harzallah M. Huza and F. Trichet. OntoMas: a tutoring system dedicated to ontology matching. In *Proc. of the Int. Workshop on Ontology Matching colloc. with the ISWC2006*, 2006.
16. J. Madhavan, P.A. Bernstein, P. Domingos, and A.Y. Halevy. Representing and Reasoning about Mappings between Domain Models. In *Proc. of the 18th Nat. Conf. on AI and 4th Conf. on Innovative Applications of AI*, pages 80–86, 2002.
17. D. McGuinness, R. Fikes, J. Rice, and S. Wilder. The Chimaera ontology environment. In *Proc. of the 17th National Conference on AI*, pages 1123–1124, 2000.
18. S. Melnik, H. Garcia-Molina, and E. Rahm. Similarity Flooding: A Versatile Graph Matching Algorithm and Its Application to Schema Matching. In *Proc. of the 18th International Conference on Data Engineering (ICDE02)*, 2002.
19. M. Mochol, A. Jenztsch, and J. Euzenat. Applying an Analytic Method for Matching Approach Selection. In *Proc. of the Int. Workshop on Ontology Matching colloc. with the ISWC2006*, 2006.
20. E. Paslaru. Using Context Information to Improve Ontology Reuse. In *Proc. of the Doctoral Consortium at the CAISE'05*, 2005.
21. E. Paslaru, M. Mochol, and R. Tolksdorf. Case Studies on Ontology Reuse. In *Proc. of the 5th International Conference on Knowledge Management*, 2005.
22. J. Poole and J. Campbell. A Novel Algorithm for Matching Conceptual and Related Graphs. *Conceptual Structures: Appl., Impl. and Theory*, pages 293–307, 1995.
23. G. Stumme and A. Maedche. FCA-MERGE: Bottom-up merging of ontologies. In *Proc. of the 17th IJCAI 2001*, pages 225–230, 2001.