MAFRA – An Ontology MApping FRAmework for the Semantic Web

Nuno Silva and João Rocha
DEI - ISEP – IPP
Porto – Portugal
Nuno.Silva@dei.isep.ipp.pt

Abstract: Ontology mapping intends to define how data specified according to an ontology (source ontology) may be transformed into data specified according to another ontology (target ontology). MAFRA framework aims to cover all phases of the ontology mapping process, including discovery of similarities between elements in ontologies and selection and specification of required transformations.
MAFRA implementation adopts a mapping declarative strategy. MAFRA implementation adopts an open architecture in order to maximize and answer Semantic Web requirements.
MAFRA is being developed in the context of KAON and is being applied in the European project Harmonise, which aims to provide solutions for (semi-) automatic interoperability between major operators in tourism e-business.
Keywords: Ontology Mapping, Semantic Web, Semantic Bridge

1. Introduction

The current WWW is a great success with respect to the amount of stored documents and the number of users. However, the ever-increasing amount of information on the Web places a heavy burden of accessing, extracting, interpreting and maintaining information on the human users of Web. Tim Berners-Lee, the inventor of the WWW, coined the vision of Semantic Web, suggesting means for annotation of Web resources with machine-processable metadata providing them with background knowledge and meaning (see [1]). Ontologies as means for conceptualizing and structuring domain knowledge are seen as the key to enabling the fulfillment of the Semantic Web vision.

However, the de-centralized nature of the Web makes indeed inevitable that communities will use their own ontologies to describe their data. In this vision, ontologies are themselves distributed and the key point is the mediation between distributed data using mappings between ontologies. Thus, complex mappings and reasoning about those mappings are necessary for comparing and combining ontologies, and for integrating data described using different ontologies. Existing information integration systems and approaches (e.g., TSIMMIS [4], Information Manifold [5], Infomaster[1], MOMIS[2]), are "centralized" systems of mediation between users and distributed data sources, which exploit mappings between a single mediated schema and schemas of data sources. Those mappings are typically modeled as views (over the mediated schema in the local-as-view approach, or over the sources schemas in the global-as-view approach), which are expressed using languages having a formal semantics. For scaling up to the Web, the "centralized" approach of mediation is probably not flexible enough, and distributed systems of mediation are more appropriate.

Building on this idea and on existing work, we introduce MAFRA, an Ontology MApping FRAmework (MAFRA) for distributed ontologies in the Semantic Web. MAFRA represents

[1] http://infomaster.stanford.edu/infomaster-info.html
[2] http://sparc20.ing.unimo.it/Momis/

an approach and conceptual framework providing a generic view onto the overall distributed mapping process. In particular, in this paper we focus on the methodology, representation and execution aspects of mappings. However, the proposed framework offers support in all parts of the ontology mapping life cycle.

Motivations and basic ontology mapping definitions are introduced in next section. In section 3 the underlying conceptual architecture of MAFRA is described and in section 4 is presented the developed methodology. In section 5 we give some hints on the MAFRA implementation within the KAON[3] initiative. A short discussion of related and future work is given in section 6.

2. Motivations

Ontology mapping task semantically relates any two ontologies. Considering ontology as a specification of a conceptualization [9], an infinite number of possible mapping relations between two ontologies may exist. Based on the large range of ontology mapping problem, it is important to define goals and quality indicators. Five characterization vectors were identified and defined as important goals:

- **Applicability** concerns what type of mapping relations are possibly solved in the system;
- **Semantic Expressivity** concerns to how explicit and described is the mapping specification;
- **Modularization** concerns with the system characteristic to be build upon the combination of small, simple modules into a more complex whole;
- **Reutilization** of components, namely concerning applying knowledge created from previous mapping experiences;
- **Declarativity** concerns the capabilities of the system to supply conditions to domain expert focus on semantic instead of in *how-to*. Maximizing declarativity, will improve quality and productivity, while minimizing software development and customization.

One of the important issues in describing this works tends to be the used terminology. Next paragraphs try to cope with this issue.

Ontology entity is either a concept or a property of ontology. In context of MAFRA, the ontology representation language used is RDFS[4]. Concepts are defined as *rdfs:Class* and properties as *rdfs:Property*.
Ontology Instances are the data represented according to (and coherent with) the ontology. In the context of MAFRA, ontology instances are represented in RDF. The source ontology instances are transformed in (new) target ontology instances.
Ontology Mapping Document or simply Ontology Mapping is the document specifying the procedures required to transform source ontology instances and the conditions that must hold to perform such transformations. Ontology Mapping Document life cycle represents the states the ontology mapping document passes through its life. Empirically, an ontology mapping document life cycle resembles either an ontology object or project life cycle, including analysis, requirements specification, project, simulation, implementation, tests and validation, maintenance and release.
Ontology Mapping Process is the set of activities associated to the ontology mapping document and the activities to execute it over a set of source ontology instances. Basically, it may be decomposed in two sub processes:

- Ontology Mapping Document specification, which is effectuated at ontological level;

[3] http://kaon.semanticweb.org
[4] http://www.w3.org/TR/rdf-schema/

- Ontology Mapping execution over a set of source ontology instances, carried out at data level.

3. Conceptual Framework

An ontology mapping process, as defined in [2], is the set of activities required to transform instances of source ontology into instances of target ontology. Studying the process and analyzing different approaches from the literature we observed a set of commonalities and assembled them into the MAFRA conceptual framework, outlined in Figure 1. The framework consists of five horizontal and four vertical modules. Horizontal modules correspond to five fundamental phases occurring in any ontology mapping process. The vertical modules interact with the horizontal phases along the entire ontology mapping process providing specific functionalities.

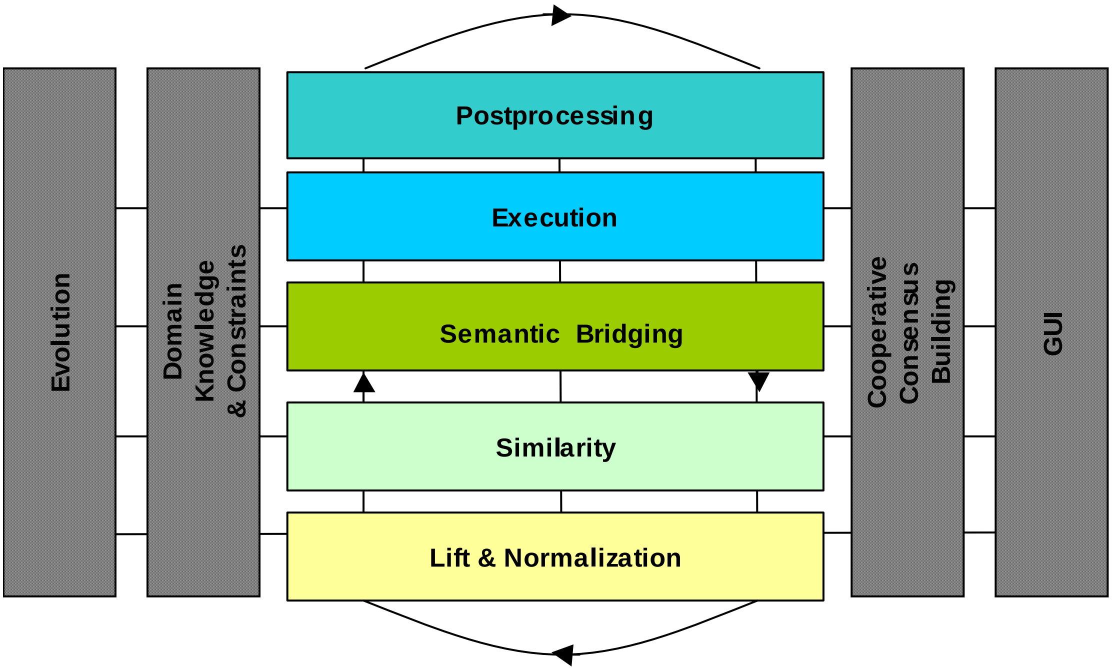

Figure 1 – MAFRA Conceptual Framework

3.1. Horizontal Dimension of MAFRA

Within the horizontal dimension, we identified the following five modules corresponding to five process phases:

Lift & Normalization focuses on coping with language and lexical heterogeneity [3] between source and target ontology. Language lift focuses on minimizing syntactical and structural representation heterogeneity, while lexical normalization focuses on minimizing ontology contents lexical heterogeneity. Lexical normalization acquires special relevance in normalizing abbreviations and acronyms.

Similarity Discovery aims to discover and establish similarities between source ontology entities and target ontology entities.

Semantic Bridging uses the similarities computed in the previous phase to establish correspondences between entities from the source to target ontology. Complementarily it defines the transformations required to transform source instances into the most similar target instances.

Execution actually transforms instances from the source ontology into target ontology instances by evaluating the semantic bridges defined earlier. In general two distinct modes of operation are possible, namely offline (static, one-time transformation) and online (dynamic,

continuous mapping between source and the target) execution depending on operation environment and requirements.
Post-processing takes the results of the execution module to check and improve the quality of the transformation results. The most challenging task of post-processing is establishing object identity - recognizing that two instances represent the same real-world object.

3.2. Vertical Dimension of MAFRA

The vertical dimension of MAFRA contains modules that interact with horizontal modules during the overall mapping process. Four modules have been identified:
Evolution focus on maintaining semantic bridges obtained by the "Semantic Bridge" module, in synchrony with the changes in the source and target ontologies.
Cooperative Consensus Building. The cooperative consensus building aspect is responsible for establishing a consensus on semantic bridges between two communities participating in the mapping process. This is a requirement, as one has to choose frequently from multiple alternatively possible mappings. Automating the mapping process as much as possible reduces the amount of human involvement required to achieve consensus.
Domain Constraints and Background Knowledge. The quality of similarity computation and semantic bridging may be dramatically improved by introducing background knowledge and domain constraints, e.g. by using glossaries to help identify synonyms or by using lexical ontologies, such as WordNet or domain-specific thesauri, to identify similar concepts. **Graphical User Interface**. Mapping is a difficult and time consuming process, which is not less difficult than building an ontology itself, i.e. deep understanding of both conceptualizations required on human side, thus extensive graphical support must be given and it is a separate issue how this can be achieved in an optimal way. The graphical user interfaces (GUI) is further elaborated in section 0.

4. Mapping methodology

This section describes the MAFRA ontology mapping methodology (MAFRA-M), pointing out requirements, approaches and decisions. In many aspects the methodology and ontology-mapping document are tightly related since the description must conform to the methodology specifications.

MAFRA-M intends to provide guidelines in the ontology mapping process, especially in the representation and execution of semantic relations. Therefore, MAFRA-M specially focuses on the Semantic Bridging, Execution and Evolution modules of MAFRA framework.

Semantic bridge is the most important concept in an ontology mapping methodology. A semantic bridge (bridge for short) is a set of elements describing:

- **What entities**, from both source and target ontology, are semantically related;
- **How** source instances are transformed into target instances;
- **Conditions** must hold to execute the bridge?

What

In order to answer this question, two type of semantic bridges are introduced:

- **Concept Bridge** used to describe the semantic relations between (source and target) concepts.
- **Property Bridge,** used to specify the semantic relations between (source and target) properties, either relations or attributes.

Furthermore, previous semantic bridges are affected by concept and property characteristics. In special, semantic bridges must reflect:

- **Identity**: a concept instance has its own identity while a property has not. In fact, a property value is associated with a concept instance, which implies concept instance is a container of property values. The *hasBridge* construct is used to acquire these characteristics into the ontology mapping specification.
- **Hierarchical structure:** concept and property form distinct hierarchy structure in the ontology. Hierarchies are supported in MAFRA-M through *subBridgeOf* construct, which constructs itself a hierarchy of semantic bridges. Bridges hierarchy defines the possibility to apply super bridges to concepts and properties that have not specific bridges, but their parents do.
- **Cardinality**, specifies the dimension of source and target entities involved in the semantic bridge. It is characterized by a pair of values in the form of *x:y*, where *x* defines the number of source entities and *y* the number of target entities. Concept bridge cardinality is always *1:1*, meaning that one and only one source concept corresponds to one and only one target concept. This does not mean one source concept is related to one single target, or that each concept instance corresponds to one single target concept. Property bridge cardinality ranges from *1:1* to *n:m*. All cardinality combinations are possible except *0:1* (at least a source value is necessary to create a target property is defined) and *x:0* (no target property value is created). A property bridge cardinality is directly related to the transformation service applied to it (see next).

How

Transforming source data into target data is the main goal of the ontology mapping process, and reflects the complexity and number of possible ontology mapping relations. The definition of how the transformation are performed assumes special relevance and is distinct depending on concept or property bridge:

Concept bridge *translates* source instances into target instances. Basically, a reclassification and renaming process occurs through a concept bridge, i.e. a source instance of a certain type gives rise to a new target instance of another type (i.e. reclassification) and a new identity (i.e. renaming). The translation procedure applied in concept bridges is universal, i.e. independently of the semantic relation associated to the concept bridge the required translation process is always the same.

Property bridge *transforms* source instance properties into target instance properties. It defines which and how source properties are transformed into which target properties. Possible transformations includes, copy of attribute and relations, concatenation and split of attributes, transformation of alphanumeric attribute values through regular expressions, default values, etc. Due to the multitude of potential required transformations, a universal transformation procedure is impossible. Hence, an open set of transformation procedures is required in an ontology mapping system.

Conditions

Semantic mapping is often dependent on the data values and not only on the ontologies being mapped. Hence it requires condition specification at development time and its verification at execution time. Two main problems occur:

- How to specify and execute conditions. MAFRA-M adopted an explicit specification of conditions in contrast to transformation process hard-coded conditions. Therefore, condition specification is treated just like any other transformation argument.
- How to control execution based on the result of condition verification. For that, the A*lternative Bridge* is suggested. Alternative Bridge is a set of mutually alternative

semantic bridges, either concept or property bridges (in an specific alternative bridge only one type of bridges may exists). Bridges inside an alternative bridge are ordered, and self-conditioned. The first successful bridge is executed, while the others are rejected.

5. Implementation

MAFRA is currently under development within the KAON Ontology and Semantic Web Framework. Five modules of MAFRA have been implemented: the normalisation module, the automatic similarity discovery module, the semantic bridging representation, the graphical user interface and the execution engine. A 3rd party solution was adopted concerning the Lift phase.

Lift & Normalisation phase implementation

Both ontologies must be normalized to a uniform representation (e.g. RDF/S, DAML, OWL) in order to eliminate syntax differences and making semantics differences between the source and the target ontology more apparent [2]. The lift process is beyond our efforts and therefore we seeked for a third party solution, in which the NormTool fits perfectly our requirements. Furthermore, one essential step of this first phase is normalization. Normalization is the process whereby the ontologies lexical elements are reduced to a common set of specifications. Three distinct ordered tasks are performed in our approach: (i) tokenization of the vocabulary, (ii) elimination of resulting stop words and (iii) expansion of acronyms. The result is a list of normalized *lexica* providing better support to the Similarity Discovery phase. Some ontology languages provides support for lexical dimension. RDF/S suggests the very *naïf rdfs:label*, while KAON extensions to RDF/S provide very powerful, mechanisms for muti-language support.

Similarity Discovery

Several different similarity measures have been proposed in literature. We distinguish between strategies and algorithms even if both are combined in a similarity measure approach. We adopted a multi-strategy and multi-algorithm process. The first strategy focuses on acquiring a **lexical similarity** between each entity in source entity with each and all entities in target entity. For that, WordNet and an altered Resnik [6] algorithm are used. Subsequently, a next step calculates the so-called **property similarity** that is responsible to acquire the similarity between concepts based on their properties, either attributes or relations. The bottom-up similarity intends to propagate the similarity (or dissimilarity) from lower parts of the taxonomy to the upper concepts. It uses the property similarity as input and propagates the values to the top. This similarity gives a good overall view of similarity between taxonomies. Complementarily, the top-down similarity propagates similarities from top to bottom, and assumes special relevance when top-level concepts have a higher or lower similarity.

Semantic bridging

This simple principle motivate our approach in semantic bridge specification following the evidence that RDFS ontologies normally rely and exploit the underlying OO part of RDFS, namely the taxonomic structure in the form of a graph, and in particular cases, the form of a tree. The semantic bridging phase is divided in five distinct steps:

1. First, concept bridging chooses according to the similarities found in previous phase, pairs of entities to be bridged. The same source entity may be part of different bridges. Two distinct cases may arise: First, the source concept corresponds to either one of the target concepts. This implies that the source instance will give rise to one instance of just one of the target concepts. Second, the source concept corresponds to many distinct target concepts, which implies that the source instance will give rise to one instance of many target concepts. The automatic process tries to find the best choice based on heuristics and lexical relations. For example, if the target concepts have the source concept as hypernym that tends to show that source instance should be translated to either one of the target

concepts. The antonym relation (extracted from WordNet) may also be used for confirming of this case. On the other hand if no hypernym relation exist it tends to correspond to the second case.

2. Second, the property-bridging step is responsible to specify the matching properties for each concept bridge. As for concepts, a property may be part of several matches, which implies the same two cases previously mentioned for concepts. Therefore, the same strategy may be used in here. It is important to emphasize that properties in our approach are of two types, distinguishing between attributes and relations. If source and target properties are of different types the transformation specification information is required, where the domain expert is asked to supply this information.
3. Third, the inferring step focus in endowing the mapping with bridges for concepts that do not have a specific counterpart target concept. In fact, a source concept *cs1* may not always have a target concept counterpart *ct1*. However, if a match exists between the source concept *cs0* (a super concept of *cs1*) and *ct0*, than an implicit similarity exists between *cs1* and *ct1*. This scenario is depicted in Figure 2. Even if the concept *Employee* has no direct counterpart in the target ontology, instances of this concept should be translated into *Academics* instances. This can be automatically inferred because *Employee* is sub concept of *Person*, which in turn is bridged with *Academics*. However this is not always a straightforward solution because ambiguity arises in some situations. To infer a bridge to *PhD_Student* concept is one of such situations. This concept is sub concept of two concepts, which means that any instance of *PhD_Student* is also an instance of both *Employee* and *Student*. However, such qualification do not exists in target ontology. In these situations we use available domain knowledge, namely the exploitation of previous mappings where such concepts were bridged. However, for the moment this decision is up to the domain expert. Inferred bridges are always sub bridges of some higher bridge and should not state the target entity. In this example, the process creates an inferred bridge that relies on between *Person* and *Academics* to execute the translation. This is called encapsulation in the OO paradigm.

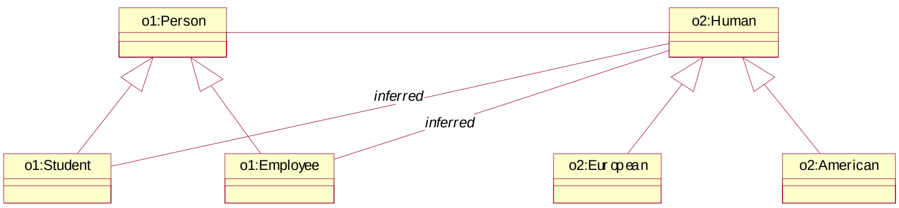

Figure 2 - Inferring best possible bridge

4. Fourth, the refinement step intends to improve quality of bridges between a source concept and sub concepts of target concepts. In fact this is a complementary procedure of the similarity phase. Besides this step to be optional, it becomes important if a good mapping quality is necessary.
5. Fifth, the transformation specification step intends to associate a transformation procedure to the translation, in a way that source instance may be translated into target instances. This task may be automated in some extend, especially in well-known situations, which can be acquired through experience. However this task is fundamentally a domain expert step. There are two main issues that are extremely dependent on the domain expert: (i) the alternative bridge conditions specification arising in concept bridging and property bridging, and (ii) the specification of mapping between different types of properties.

Graphical-user interface

A screen-shot of the user interface for mapping specification is presented in Figure 3. In this example two ontologies have been opened side by side, and in between an instance of the semantic bridging ontology is created using a graphical user interface using the drag&drop mechanism.

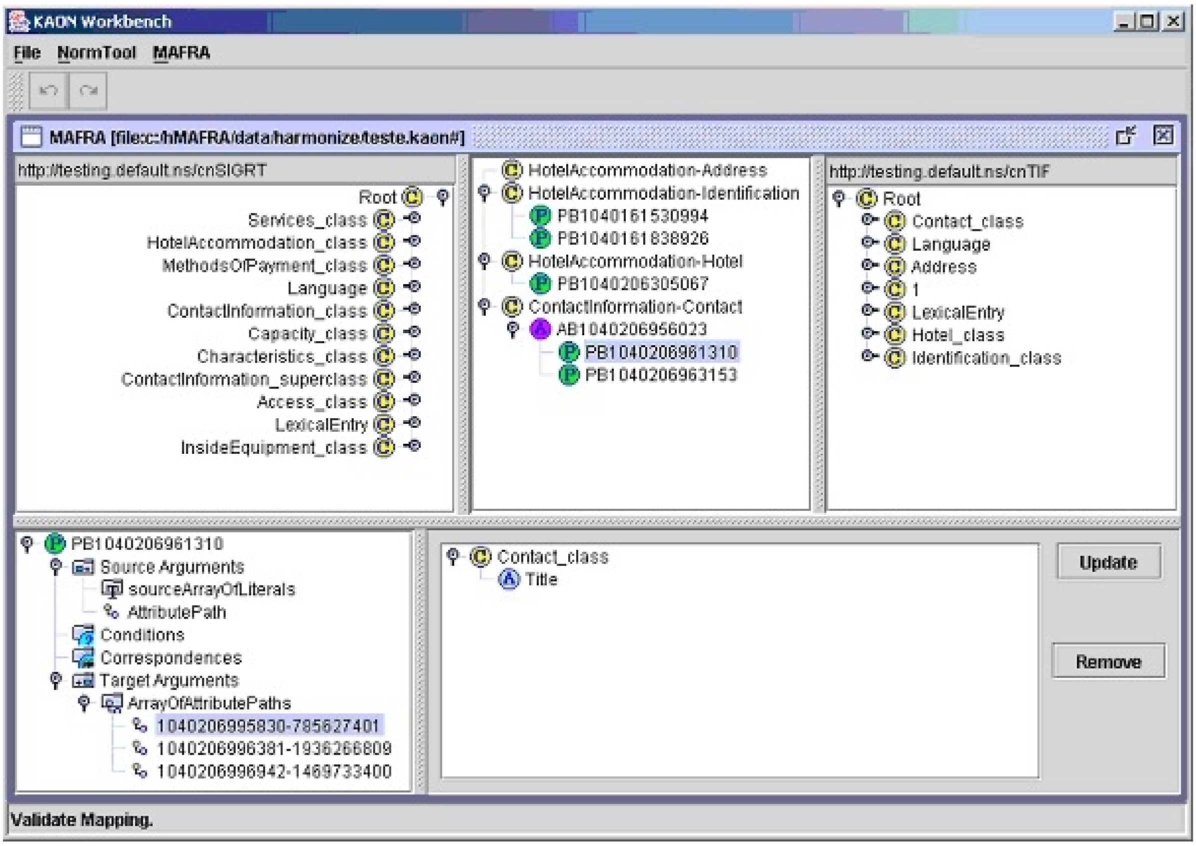

Figure 3 - Screen shot of MAFRA User Interface

The developed mapping tool represents the domain expert interface with the similarity and semantic bridging modules, and the possibility to interact within the mapping process. The user participation is fundamental and must be promoted. We adopted a tree view similar to the most common ontology editors. The mapping tool defines two tree views for the ontologies being mapped (in the left and in the right) and a central tree view representing the mapping. Bridges are manipulated through drag and drop actions.

The execution engine has been implemented in Java, exploiting the features of KAON, and it represents the first step of out efforts in developing a general translation engine for SBO instances (instances respecting the Semantic Bridge Ontology, part of the same effort [7]).

6. Conclusion and Future Work

Ontologies may be used for achieving a common consensus within a user community about conceptualizing, structuring and sharing domain knowledge. We have motivated that it is unrealistic to assume that one single ontology for different communities of users is realistic in real-world applications. We argue that decentralization has been one of the key elements for the scalability of the World Wide Web and its underlying applications. In order to balance the autonomy of each community with the need for interoperability, mapping mechanisms between ontologies have been proposed. In this paper we presented the Ontology Mapping Framework (MAFRA) supporting the interactive, incremental and dynamic ontology mapping process in the context of the Semantic Web. In this paper a specific focus has been set on the semantic bridging phase where we have provided a detailed description of a semantic bridge meta-ontology, that is instantiated when mapping between two domain ontologies. MAFRA is being applied in the European project Harmonise, where real world ontologies are being mapped in order to improve business between tourism intervenients. Complete examples will be presented in final manuscript.

References

[1] Tim Berners-Lee, Mark Fischetti (Contributor), Michael L. Dertouzos; “Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web”; 1999.

[2] Philip A. Bernstein; Erhard Rahm; “On Matching Schemas Automatically”; MSR-TR-2001-17; February 2001.

[3] P.R.S. Visser, D.M. Jones, T.J.M. Bench-Capon e M.J.R. Shave; “An Analysis of Ontology Mismatches; Heterogeneity versus Interoperability”; AAAI Spring Symposium on Ontological Engineering; Stanford University, California, USA; - also appeared as AAAI Technical Report SS-97-06, A. Farquhar and M. Gruninger (eds.); 1997.

[4] J. Hammer and H. Garcia-Molina and K. Ireland and Y. Papakonstantinou and J. Ullman and J. Widom; “Information Translation, Mediation, and Mosaic-Based Browsing in the TSIMMIS System”; Exhibits Program of the Proceedings of the ACM SIGMOD International Conference on Management of Data, page 483; San Jose, California; June 1995.

[5] Alon Y. Levy and Anand Rajaraman and Joann J. Ordille; “Querying Heterogeneous Information Sources Using Source Descriptions”; in Proceedings of VLDB-96; 1996.

[6] P. Resnik; “Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language"; Journal of Artificial Intelligence, volume 11-11, pp 95-130; 1999.

[7] A. Maedche, B. Moltik, N. Silva and R. Volz; “MAFRA —An Ontology MApping FRAmework in the Context of the Semantic Web”; in Proceeding of the EKAW’2002, Siguenza, Spain; 2002.

[8] J. Y. Park, J. H. Gennari, & M. A. Musen; “Mappings for Reuse in Knowledge-based Systems”; SMI Report Number: SMI-97-0697; 1997.

[9] T. R. Gruber; "Towards Principles for the Design of Ontologies Used for Knowledge Sharing"; Formal Ontology in Conceptual Analysis and Knowledge Representation, N. Guarino and R. Poli (editors); Kluwer Academic Publishers, Deventer, The Netherlands; 1993.