Dissertation
zur Erlangung des akademischen Grades
der Doktorin der Naturwissenschaften
(Dr. rer. nat)

The Methodology for Finding Suitable Ontology Matching Approaches

eingereicht
am Institut für Informatik
des Fachbereichs Mathematik und Informatik
der Freien Universität Berlin

von

Dipl. Inform. Małgorzata Mochól

Tag der Disputation: 29. Januar 2009

1.Gutachter:
Prof. Dr.-Ing. Robert Tolksdorf
Freie Universität Berlin
Fachbereich Mathematik und Informatik
Institut für Informatik
AG Netzbasierte Informationssysteme (AG NBI)

2.Gutachter:
Dr. Charles J. Petrie, Jr.
Stanford University
Computer Science Department
Stanford Logic Group

Dissertation
zur Erlangung des akademischen Grades
der Doktorin der Naturwissenschaften
(Dr. rer. nat)

The Methodology for Finding Suitable Ontology Matching Approaches

eingereicht
am Institut für Informatik
des Fachbereichs Mathematik und Informatik
der Freien Universität Berlin

von

Dipl. Inform. Małgorzata Mochól

Tag der Disputation: 29. Januar 2009

Abstract

Interoperability has gained in importance and become an essential issue within the Semantic Web community. The more standardized and widespread the data manipulation tools are, the easier and more attractive using the Semantic Web approach has become. Though Semantic Web technologies can support the unambiguous identification of concepts and formally describe relationships between concepts, thereby allowing the representation of data in a more meaningful and more machine-understandable way, Web developers are still faced with the problem of semantic interoperability, which stands in the way of achieving the Web's full potential. To attain semantic interoperability, systems must be capable of exchanging data in such a way that the precise meaning of the data is readily accessible, and the data itself can be translated by any system into a form that it understands. Hence, a central problem of interoperability and data integration issues in Semantic Web vision is schema or ontology matching and mapping.

Considering this situation in Semantic Web research, we wish to contribute to the enhancement of (semantic) interoperability by contributing to the ontology matching solution. The number of use cases for ontology matching justifies the great importance of this topic in the Semantic Web. Furthermore, the development and existence of tried and tested ontology matching algorithms and support tools will be one of the crucial issues that may have a significant impact on future development. Therefore, we have developed a Metadata-based Ontology Matching (MOMA) Framework that addresses data integration and the interoperability issue by creating and maintaining awareness of the link between matching algorithms and various ontologies. Our approach allows for a more flexible manual and (semi-)automatic deployment of matching algorithms, depending on the specific requirements of the application (e.g. suitability to certain types of input) to which the matchers are to be utilized. Since it is difficult to theoretically compare the existing approaches due

to the fact that they are based on different techniques, a matcher characteristic that describes the different approaches on various levels of detail is needed. We have hence developed a Multilevel Characteristic for Matching Approaches (MCMA), which forms part of the MOMA Framework and has been utilized for the matcher selection. Taking into account the requirements of the successful deployment of semantic technologies regarding off-the-shelf and easy to use tools, the MOMA Framework should be capable of meeting the demands of different users: humans (Semantic Web experts and ontology matching lay users) and machines (e.g. service/matching providers). For human users, the process of choosing the suitable approach can be carried out manually, while machines require at the very least a semi-automatic selection of appropriate matchers. In manual selection, since the decision depends on multiple criteria (MCMA) and scales are not consistent, we have applied a systematic approach that structures the expectation, intuition, and heuristic-based decision making into a well-defined methodology called Analytic Hierarchy Process (AHP). In order to (semi-) automatically determine which matchers are appropriate for a given application, the MOMA Framework uses additional information on the ontologies (ontology metadata) and available matchers (matcher metadata). The ontology metadata captures information about matching relevant ontology features while the matcher metadata, based on the MCMA, describes the most important characteristics of the matching services. Furthermore, since explicit knowledge about the dependencies between the matching algorithms and the structures on which they operate is needed, we have formalized it into dependency rules statements that, taking into account the characteristic of matching approaches and ontological sources to be matched, determine which elements (i.e. matchers) are to be used for a given set of ontologies.

Since the evaluation aspects of the MOMA Framework are directly related to the usage of the framework in real-world situations, the evaluation of both the AHP and rule-based approaches has been conducted on real-world test cases defined by the Ontology Alignment Evaluation Initiative (OAEI) Campaign. The results of the evaluation process demonstrate the applicability of the MOMA Framework for matcher selection and the accuracy of its predictions.

With the MOMA Framework, which allows for the selection of suitable matching approaches w.r.t the given application requirements, we intend to contribute to the tackling of real world challenges, which are commonly agreed testbeds and benchmarking, with the aim of ensuring seamless interoperability and integration of the various Semantic Web technologies.

Acknowledgments

Though it will not be enough to express my gratitude in words to all the people who supported and helped me, I would still like to give my many thanks to all of them.

First of all, I would like to thank both my supervisors, Prof. Dr.-Ing. Robert Tolksdorf from the Free University of Berlin and Dr. Charles J. Petrie Jr. from Stanford University, for numerous interesting discussions and fair comments on my PhD work. I especially want to thank Prof. Tolksdorf, whose support and mentoring made my thesis work possible. I am very grateful for his patience, motivation, enthusiasm, and criticism during the entire work on my dissertation.

I would like to acknowledge my past and present colleagues from the Working Group Networked Information System (AG NBI) at the Free University Berlin and particularly to Dr. Elena Simperl for great teamwork that resulted in many joint publications, Dr. Klaus Schild for providing me with valuable comments, especially in the very beginning phase of the PhD work, and Dr. Lyndon J.B. Nixon, with whom I shared my office and who patiently answered my frequent questions. I also would like to express my gratitude to my student Anja Jentzsch for her continuous engagement in the matching domain and persistence in assisting me and revising the implementation of the thesis w.r.t my comments.

I would also like to thank all my friends for their patience and support during this challenging time. Special thanks go to Ali Rostami for being a long lasting source of energy during this exhaustive research and for his understanding in the long journey towards my PhD, as well as to my best friend Karolina Polosin for the relaxing talks, on the phone and in person, and her sympathetic ear for each sort of problem.

I would also like to thank my stepfather for his wide-ranging support and motivation over my all PhD years. And finally, I am very very very grateful to my mom. Her love, faith, patience, and understanding during the entire PhD period encouraged me to work hard and to continue pursuing the dissertation. The completion of this thesis would be inconceivable without her help and motivation and . . . her culinary skills over all these years.

Thank you very much!

Małgorzata Mochól
Berlin
November 18, 2008

I would like to dedicate this PhD work to my mom:
without you none of this would ever have happened.

Moją pracę doktorską chciałabym zadedykować mojej mamie:
bez Ciebie to wszystko nie byłoby możliwe.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Abstract In this introductory chapter, we present the major reasons and motivation for our research. We also briefly introduce our approach in which the suitable algorithms for given contexts are selected and the selection process performs prior to the execution of a particular matching algorithm. Furthermore, we explain the term "suitability", which plays a crucial role in our research, and discuss the research methodology applied in the course of our study. In the context of the utilized methodology, we outline the major steps undertaken together with the main contributions of the thesis. In conclusion, we give the reader an overview of the contents of this work.

"Motivation is undoubtedly the single greatest influence on how well people perform. Most productivity studies have found that motivation has a stronger influence on productivity than any other factor."

Steve McConnel, Rapid Development

1.1 Motivation

The World Wide Web, known as “WWW” or “Web”, is, according to World Wide Web Consortium (W3C)[1], the universe of network-accessible information, the embodiment of human knowledge with a body of software, and a set of protocols and conventions. The use of hypertext and multimedia techniques has made it easy for anyone to surf, browse, and contribute to the Web; it is a universal medium for the exchange of data and knowledge. Over the last two decades, the World Wide Web has rapidly evolved into a vast repository containing huge amounts of decentralized information on all matters of interest. The current Web, as stated in [42], can be characterized as the second Web generation. The first generation Web started with handwritten HTML pages while the second generation made the step to machine generated and often active HTML pages. The first two generations were meant for direct human processing (reading, browsing, form-filling), but the third generation Web, the ”Semantic Web“, aims to provide machine processable information. Its main challenge is to find, integrate, and process all (available) information relevant to a particular context. Since most of the Web’s content is primarily designed to be read by humans, machines can parse Web pages for layout, though they cannot automatically process data from a particular Web site without understanding its semantics. For this reason, the next years will be characterized by the transformation of the Web from a document-publication medium intended for human utilization into a medium for intelligent knowledge exchange [85]. In this context, interoperability between systems, not only in the current Web, but especially in the Semantic Web, will become one of the principal issues in achieving the time-to-market demanded in a competitive environment.

Although interoperability is a very broad term containing within it many of the issues of effectiveness with which diverse information resources might fruitfully co-exist in common, **interoperability** is the ability of the system, or the components therein, to provide information portability and inter-application cooperative process control. Furthermore, as stated in [12], interoperability can also be characterized as *a form of system intelligence that enhances the cooperation between component information systems*. Such intelligence is necessary to provide services, locate resources, coordinate, and carry out complex functions across component information systems without the need to know in advance what resource are available and how to acquire them. In terms of interoperability, we always address one (or more) of the six major interoperability levels ranging from network protocols and operating systems to data models and application semantics (cf. Fig. 1.1).

The issues of interoperability and the data integration have long been the subject

[1] `http://www.w3.org` accessed on 02.10.2008

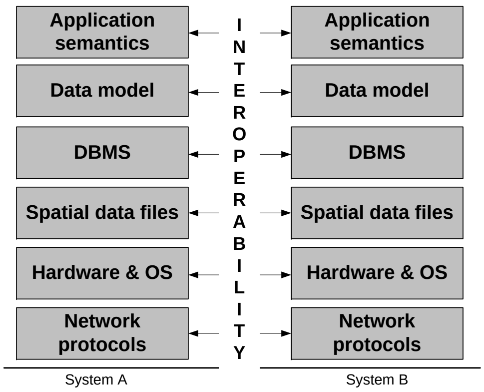


Figure 1.1: Levels of the interoperability [12]

of discussion in Information Society research: In 1993, the International Council of Science (ICSU)[2], together with United Nations Educational, Scientific and Cultural Organization (UNESCO)[3] and the Committee on Data for Science and Technology (CODATA)[4], published a roadmap – Agenda of Action – concerning the role of science in the information society. In it, they intended to promote *"the interoperability principles and metadata standards to facilitate cooperation and effective use of collected information and data."* [99] However, the main contributions to ensure interoperability on the Web have been achieved by (i) widely established standards, e.g. the TCP/IP protocol ensuring transport interoperability, and HTTP and HTML, which provide a standard method to retrieve and present hypertext text documents, (ii) together with multiple strategies developed to facilitate data interoperability. Nevertheless, to achieve overall data interoperability, one must attain syntactic, schematic, and semantic interoperability while respectively taking into consideration syntactic, schematic, and semantic heterogeneities:

syntactic interoperability means the degree to which different applications can share and exploit data (how easy it is to read the data and get representation that can be exploited by applications); it mainly deals with parsing the data correctly [42],

schematic interoperability handles difficulties that arise from the use of different schemas to represent the same data [38],

[2] `http://www.icsu.org/` accessed on 02.10.2008
[3] `http://www.unesco.org/` accessed on 18.10.2008
[4] `http://www.codata.org/` accessed on 02.10.2008

semantic interoperability tackles the problem of understanding data and the difficulty in integrating resources developed using various vocabularies and different perspectives on the data [84]; it deals with defining mappings between terms (e.g. unknown terms and known terms) in the data [42].

Over the last few years, interoperability has gained even more in importance and become a highly essential issue within the Semantic Web community. The more standardized and widespread the data manipulation tools are, including a higher degree of syntactic interoperability, the easier and more attractive using the Semantic Web approach has become. Though Semantic Web technologies can support the unambiguous identification of concepts and formally describe relationships between concepts, thereby allowing representation of data in a more meaningful and more machine-understandable way, Web developers are still faced with the problem of semantic interoperability, which stands in the way achieving the Web's full potential. The problem here is that the cost of establishing semantic interoperability, due to the need for content analysis, is usually higher than what is needed to establish syntactic interoperability [42]. *Semantic interoperability is necessary before multiple applications can truly understand data and treat it as information*; it will thus be, according to [42], a *sine qua non* for Semantic Web. To achieve semantic interoperability, systems must be capable of exchanging data in such a way that the precise meaning of the data is readily accessible, and the data itself can be translated by any system into a form that it understands [84]. Hence, a central problem in interoperability (particularly in semantic interoperability) and data integration issues in Semantic Web vision is schema or ontology matching and mapping [38]. Despite the relatively large number of promising matching approaches and the diversity of ideas and techniques employed, current matching approaches still show important limitations when applied to the emerging Semantic Web [69, 124, 130, 168]. It has also turned out that different exiting matching algorithms are better suited for matching different sets of ontologies since they are tailored to certain ontology types (cf. Sec. 3.5 and 5.1) or cannot be applied across various domains with the same effects (cf. Sec. 5.1.4). The heterogeneity of the matching approaches forces the semantic developers to choose matchers that would be appropriate for their goals. The selection process and, in particular, the case of a wrong choice would be reflected, for example, in the increased cost of the implementation of semantic based applications (cf. Sec. 8.6.2) which, in turn, would retard the maturing and moving of semantic technologies beyond academic applications into a broader industrial scale. Furthermore, today it takes an expert to determine the best algorithm for a particular application, and a decision can usually be made only after complex experimentation, testing and execution of different algorithms, so, as a result, the necessary scaling and off-the-shelf use of matching algorithms are not possible.

The situation described above reigns not only in the "wild", "wide" Semantic Web community but pesters us also in "the privacy of our homes". In one of our first Semantic Web projects[5], we aimed at developing a hotel evaluation and recommendation engine that uses Semantic Web technologies to enhance the quality of an existing hotel search engine and, in turn, that offers a system to evaluate the extensive and constantly changing range of goods offered to the customer [66, 143]. With our approach, we strove to optimize the hotel selection process, raise the quality of travel services, save travelers' time, and significantly reduce the direct as well as indirect travel costs. And so the main task in the project consisted of implementing a prototypical framework to be used within a semantic hotel search engine whereby the evaluation engine returned a ranking of hotels with detailed information regarding the degree of the hotel/profile matching, which then provides a basis for further refinement and re-evaluation. At that point, our difficulties started, as we needed a matching approach capable of matching hotel features with the travellers' profiles. We needed an approach able to deal with our particular ontological sources and their instances, and to deliver the required output within an acceptable time, although we were not well versed in the ontology matching domain, had no overview regarding the diversity of existing matching approaches, and did not know how to find an appropriate solution. We spent a great deal of time searching for suitable matchers, testing them and trying to execute existing approaches, and became rather frustrated wishing for a tool that would facilitate the determination of potentially suitable matching approaches without (prior to) their execution.

The intricate situation within the matching domain caused by some open issues, as we specify in Sec 3.5, has been identified in the broad Semantic Web community (cf. Sec. 5.1.3) as well as in our "house" projects. Our personal bad experiences and difficulties encountered during the project's realization[6] were the main source of our motivation and explain why we have concentrated on the matching issue.

1.2 Our approach

Considering the "global" and "personal" problems mentioned above and the current situation in Semantic Web research, **we wish to contribute to the enhancement of (semantic) interoperability by supporting ontology matching issue with our research**. We have therefore developed a matching framework for ontologies – *Metadata-based Ontology Matching (MOMA) Framework* – that implements an evaluation method in which the algorithms suitable for a given context are selected and

[5] Reisewissen Project;`http://reisewissen.ag-nbi.de/en` accessed on 02.10.2008
[6] Description of our further projects can be found in Sec. 5.1.2

the selection process performs prior to the execution of a particular matching algorithm. Thereby the framework contributes to resolving the data integration and interoperability issue by creating and maintaining awareness of the link between matching algorithms and various ontological sources. Our approach allows for a more flexible manual and semi-automatic deployment of matching algorithms, depending on the particular requirements of the application (e.g. suitability to certain types of input sources) to which the matchers will be utilized. Since it is difficult to theoretically compare the various existing approaches, due to the fact that they are based on different techniques, we need a matcher characteristic that describes the different approaches in various levels of detail; it is important to recognize cross application needs and define a suitable matcher characteristic that allows comparison of different approaches and consequently the selection of algorithms appropriate to the given requirements. We have developed a *Multilevel Characteristic for Matching Approaches (MCMA)*, which forms part of our MOMA Framework and has been utilized for our matcher selection approach. Taking into account the requirements of the successful deployment of semantic technologies regarding off-the-shelf and easy to use tools, our framework should be capable of meeting the demands of different users: humans (Semantic Web experts and ontology matching lay users) and machines (e.g. service/matching providers). For human users, the process of choosing the suitable approach can be carried out manually, while machines require, at the very least, semi-automatic selection of appropriate matchers. In manual selection, since the decision depends on multiple criteria and scales are not consistent, we must involve qualitative and quantitative methodology from the family of the multi criteria decision making process known as Multi Criteria Decision Analysis (MCDA). In particular, our manual approach is based on the systematic approach that structures the expectation, intuition, and heuristic-based decision-making into a well-defined methodology called Analytic Hierarchy Process (AHP) [164]. On the other hand, in order to semi-automatically infer which algorithms suit certain inputs, we need explicit knowledge about the dependencies between these algorithms and the structures on which they operate. We have formalized this knowledge into dependency rules-statements that, taking into account the characteristic of matching approaches and ontological sources to be matched, determine which elements (in this case the matchers) are to be used for a given set of ontologies.

Since the evaluation aspects of the MOMA Framework are directly related to the usage of the framework in real-world situations, the evaluation of both approaches, the manual as well as the semi-automatic, has been conducted on two real-world test cases. The results of the evaluation process demonstrates the applicability of the MOMA Framework for matcher selection and the accuracy of its predictions. With our MOMA Framework, which allows the selection of suitable matching approaches for given application requirements, 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 Semantic Web technologies.

1.2.1 Suitability of the matching approaches

Since the term “suitability” plays a crucial role in our work, in the following, we introduce it briefly, discussing its meaning w.r.t the selection of matching algorithms, and explaining what we mean when we say that the MOMA Framework aims to select the suitable matching approaches.

To elucidate the meaning of the term “suitability” in the context of matcher selection, we have based our explanations on the definition taken from the ISO/IEC 9126-1 standard[7] that addresses the quality model definition. ISO/IEC 9126- 1 is specified by means of six general software characteristics (which are further categorized into subcharacteristics) in order to form a basic set of independent quality characteristics [64, 106]. One of its main characteristics describes **functionality**, which refers to

> *“the capability of the software product to provide functions which meet stated or implied needs when the software is in use under specified conditions* [100, 151].

The functionality characteristic, in turn, includes the subcharacterisitc **suitability**, which describes

> *“the capability of the software product to provide an appropriate set of functions for specified tasks and user objectives* [100, 151].

If we apply this definition to *suitability* in the context of matcher selection, and if we consider the fact that the ISO/IEC 9126 definitions acknowledge that the objective is to meet user needs [10], we may state that the framework selects the appropriate algorithms from the “repository” of available matchers based on a definition of application needs and user requirements. To have a reasonably high expectation that selected components will meet the actual needs, the potential matcher users must be able to specify their needs sufficiently, while the matching approaches must contain enough information at the correct level of precision [67]. Regarding the above functionality and suitability definitions, the term *suitability* in our context means:

[7] `http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=22749`, accessed on 02.10.2008

- the ability to meet the objectives of matcher users, i.e. to satisfy the requirements of human and/or machine matcher users (cf. Sec. 4.2.2)
- that the matcher is able to process the given incoming sources (taking into account the ontologies to be matched and their characteristics such as representation language or size);
- that the matcher is able to deliver required (type of) output; and
- that the matching approach is applicable for the purposes intended:
  - to satisfy the requirements of the application context in which the matcher is utilized and
  - to serve the specified tasks (e.g. reuse of sources, transformation) for which the matcher is deployed.

Furthermore, the selection process is related to the Multilevel Characteristic for Matching Approaches (MCMA) (cf. Sec. 4.2.3), which contains factors relevant and crucial to the decision regarding the suitability of the approaches, i.e. fulfilling the requirements defined in terms of what factors describe the suitability of matching approaches. Since our matching framework, as explained earlier, supports two modes of matcher selection, semi-automatic and manual, and since in both cases the selection process is based on the ranking of matching approaches regarding the given application needs, in the case when the potential matcher candidate does not fulfill the particular requirement, it is not eliminated from the matcher list, but its position on the ranked list is downgraded. Nevertheless, the general definition of the term “suitability” specified above can slightly vary according to the applied detection mode applied:

manual mode Since, in the manual mode, the users can define their requirements in more detail, the framework can take these more specific needs into account, i.e. the matcher that possesses the appropriate documentation is deemed more suitable than one without. And as we have more information and not every requirement is critical, we accept in this selection process the situation that even the matcher with the highest ranking may not necessarily meet all the (potentially very detailed) requirements.

(semi-)automatic mode Since the information regarding e.g. ontologies is gathered automatically, there is less input of data than in the case of the manual selection mode, where information crucial to the selection of matcher is provided by the user. For this reason, the results generated by the semi-automatic selection can be less precise than the outcomes of the manual selection mode.

Throughout the course of the work, we use the words *suitable*, *relevant*, *adequate*, and *appropriate* as synonyms with the meaning as described above.

1.2.2 Methodology of the Thesis

> *"If we knew what it was we were doing, it would not be called research, would it?"*
>
> Albert Einstein

Research, according to [113], is not about discovering the unknown but rather a *"strenuous and devoted attempt to force nature into the conceptual boxes supplied by professional education"*. Furthermore, the same author defines research as activities that contribute to the understanding of a phenomenon, whereby *phenomenon* is a set of behaviors of some entities that a researcher or a group of researchers (research community) are interested in. And *understanding* is the knowledge that allows predictions of the behavior of some facets of this phenomenon. If a particular research community undertakes activities that, in its opinion, contribute to the understanding of the phenomenon, we are talking about *research methods* and *techniques* [194].

Different research problems require different research approaches. In this sense, if we consider the research in the context of the Information Systems and Information Technology discipline, it can be characterized by two paradigms: behavioral science and design science. As stated in [88], the *behavioral science* paradigm seeks to develop and verify theories that explain or predict human or organizational behavior, while the *design science* paradigm seeks to extend the boundaries of human and organizational capabilities by creating new and innovative artifacts. Design science, also called improvement research, is rooted in engineering and the sciences of the artificial, and concentrates on the problem-solving and improving nature of activities [194]. The design science process, as stated in [183], goes though the design circle as shown in Fig. 1.2 and always starts with the *awareness of a problem*. This subprocess of the cycle picks up a problem by comparing the object under consideration with the specifications. The existing knowledge is then used to suggest solutions for the issues in question (*suggestion*), which, in turn, form the basis for the *development* phase that attempts to implement an artifact which, at the end, must be *evaluated*. Development, evaluation, and further suggestion are usually iteratively performed in the course of the research (design) effort. The design research process ends with the *conclusion* that indicates the termination of the issue.

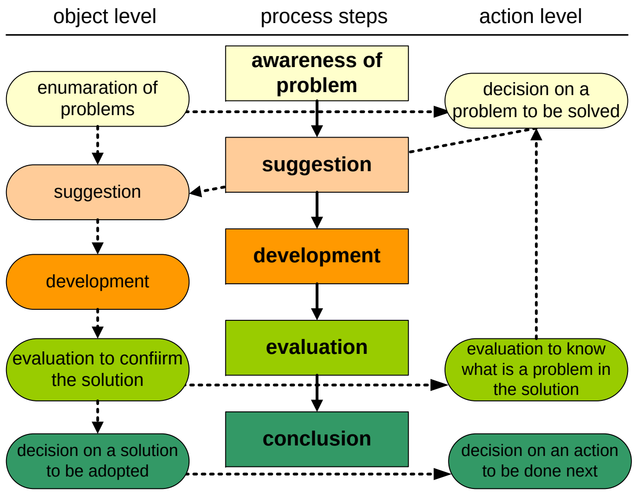


Figure 1.2: Design circle [183]

Since design science seeks to create innovations that define the ideas, practices, and technical capabilities through which the analysis, design, implementation, management, and use of information systems can be effectively and efficiently deployed, it is ideal for the purposes of our research. For this reason, the design circle in Fig. 1.2 reflects not only the methodology applied in our research process but also the structure of the entire thesis (cf. Fig 1.3):

awareness of a problem In the first phase of our research, we analyze the crucial issues in the Semantic Web with respect to (semantic) interoperability and, in particular, ontology matching in order to identify the major problems in this domain. We have found that current matching algorithms cannot be optimally used in ontology matching tasks as envisioned by the Semantic Web community, mainly because of the inherent dependency between approaches, their characteristics and ontology properties (cf. Chapters 1 - 3).

suggestion To tackle the abovementioned problem, we propose a solution – a framework – that, given a set of ontologies to be matched, takes into account the capabilities of existing matching algorithms and suggests suitable matchers to be applied (cf. Chapter 4).

development The suggestion, i.e. the solution, is realized in the form of a prototypical implementation. The implemented framework is based on the matcher characteristic (cf. Chapter 5) that has been developed to describe different matching approaches and that supports both manual as well as (semi-)automatic selection of suitable matching approaches (cf. Chapter 6 and 7, respectively).

evaluation The initial matcher characteristic is evaluated before being applied to the matcher selection process. The revised model is utilized in the selection of the matcher approach which, in turn, is evaluated in the context of case studies (cf. Chapter 8).

conclusion We summarize our work and complete the research project with conclusions as well as prospects for future work (cf. Chapter 9).

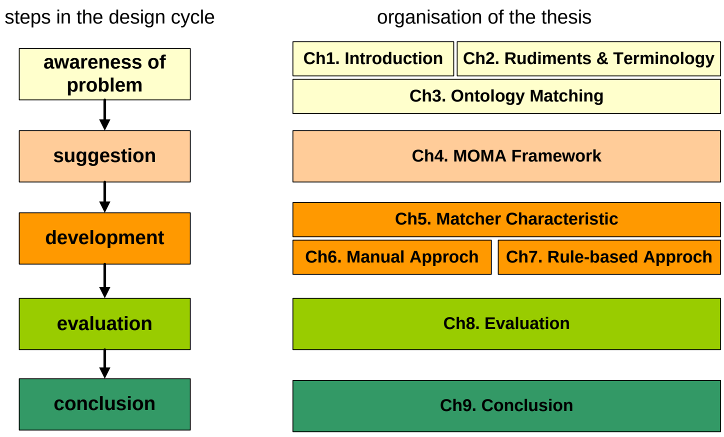


Figure 1.3: Methodology of the dissertation

Our research, since it can be classified as design science, is active with respect to technology (ontology matching) and is engaged in the creation of technological artifacts (framework) that impact people and organizations (users of the framework) [88].

1.2.3 Contributions

As stated in [88, 126], in the context of design science in Information Technology, we discuss four general artifacts as outputs: constructs, models, methods, and instantiations (cf. Tab. 1.1), among which the contributions of our work can be divided into three groups:

method We propose a methodology based on a reuse-paradigm that, given a set of ontologies to be matched, takes into account the capabilities of existing matching algorithms and *suggests appropriate matchers for application*. Our strategy

Output	Description
constructs	conceptual vocabulary and symbols of a problem domain
models	a set of propositions or statements expressing relationships between constructs (abstractions and representations)
methods	a set of steps used to perform a task (algorithms and practices)
instantiations	operationalization of constructs, models and methods (implemented and prototype systems)

Table 1.1: Outputs of design research

strives for an optimization of the matching process through the *selection of existing matchers* that have been tested and successfully applied to various applications and real world scenarios, i.e. the *developed methodology considers the dependencies between matching algorithms and the types of ontologies* they are capable of processing successfully to *recommend suitable matching approaches w.r.t the given application requirements*.

model To select suitable matching approaches, as envisioned in our methodology, we need a set of criteria that allow us to describe and, in turn, compare various matching approaches. Since it is difficult to theoretically compare the approaches, we need a possible matching algorithms characteristic that has an impact on the selection of an approach. To this end, we have built a *Multilevel Characteristic for Matching Approaches (MCMA) - a model that allows us to describe the different facets of matching approaches*. The model contains 6 main criteria, called dimensions, which form the criteria groups. These dimensions contain detailed criteria, called factors, which are specified by the attributes. These dimensions, factors, and attributes form MCMA, which is to be used for the selection of the suitable matching approaches.

instantiations The methodology developed for the selection of appropriate matcher algorithms has been implemented in the form of a framework – Metadata-based Ontology Matching (MOMA) Framework – which, utilizing the MCMA model, supports both manual as well as (semi-)automatic mode of detection of the matcher suitability.

All these contributions – our methodology based on a reuse-paradigm together with the model (MCMA), which describes the different facets of matching approaches, and the implementation of the method in the form of a framework (MOMA Framework) – support the *evaluation of the matching algorithms prior to the execution* and *have reviewed value* since we have validated them by the evaluation testing.

1.3 Overview

Before we go further in the thesis, this section gives the reader a brief overview of the entire work:

Chapter 2 Since only with a thorough understanding of the main expressions and terms is it possible to follow the ideas of our research, the thesis starts with a general introduction of the major terminologies and definitions used in the course of this work. We introduce the vision of the Semantic Web and explain the terms *ontology*, *ontology matching*, and *rules* to be used in this context.

Chapter 3 The explanation of the main terms used in this work is followed by an overview of the ontology matching domain, including an elucidation of the significance of the matching issues within the Semantic Web vision. We provide an overview of the major matcher classifications and outline a few of the existing matchers while taking into account the diversity of these approaches. In this section, the readers' attention will be drawn to the major issues within the ontology matching domain, since most of them are crucial to the work and will be addressed in the course of the thesis.

Chapter 4 In this chapter, we introduce our strategy, which strives for an optimalization of the matching process through the support of the initial step of the ontology matching reuse – the selection of matching approaches suitable to individual cases. Specifically, we describe our **M**etadata-based **O**ntology **MA**tching (MOMA) Framework, which exploits the valuable ideas embedded in current matching approaches, and, at the same time, accounts for their limitations, i.e. for specific input ontologies, it suggests the suitable matching algorithms regarding application requirements. Since the framework should meet the demands of both user groups, humans and machines, the process of choosing the suitable matching approach can be carried out manually (based on the Analytic Hierarchy Process) or (semi-)automatically (rule-based), respectively.

Chapter 5 In this chapter, we describe the development of the **M**ultilevel **C**haracteristic for **M**atching **A**pproaches (MCMA) that includes factors relevant to the matcher suitability decision. We elaborate the process of domain analysis based on experiences with different matchers in the context of ontology engineering, the examination of requirements collected during the development of Semantic Web-based applications, information gathered while collaborating with experts, and research works found in the matching literature. We then detail the developed MCMA that is organized in the form of a taxonomy, where dimensions are defined by sets of factors, and these are described by the attributes. The MCMA allows us to specify (i) the existing matching approaches, (ii) the inputs (ontologies) on which they are able to process successfully, (iii) the results (delivered output) of the matching execution, (iv) the usage of the approach (e.g. domains and tasks in which the particular matcher has been already successfully utilized), (v) the documentation, and (iv) the costs of matcher usage.

Chapter 6 As mentioned before, since the MOMA Framework integrates two different policies regarding the detection of suitable matching approaches for the processing of pre-defined ontological inputs, this section is dedicated to manual selection. We describe the adoption of one of the methods from the family of Multiple Criteria Decision Analysis (MCDA) – the Analytic Hierarchy Process (AHP) – for the purpose of matcher selection. We explain specifically why we have decided to adopt the AHP for our needs, describe how the decision-making process was applied to the matching approach selection, and introduce an AHP-based tool that supports the selection process.

Chapter 7 In the course of this chapter, we outline the (semi-)automatic selection of matcher approaches together with the high-level architecture and the tool developed. This part of the MOMA Framework uses additional information on the ontologies (ontology metadata) and available matchers (matcher metadata) in order to determine which of the latter are appropriate in a given application context. The ontology metadata captures information about matching relevant ontology features, while the matcher metadata, based on the MCMA model, describes the most important characteristics of the matching services like input and output parameters, applied heuristics. Finally, we describe the core of the rule-based MOMA Framework – the selection engine responsible for the decision-making process – by means of prescribed rules grouped in a rule repository.

Chapter 8 This chapter is dedicated to the final phase of our work – the evaluation. We start with a description of the comprehensive expert-based evaluation of the MCMA, which was conducted in two steps: the first was carried out with reviews from anonymous matching experts, and the second was based on comments coming from known experts. The evaluation process led to the refinement of the initially defined characteristic and resulted in a revised MCMA, which has been used within both the AHP and rule-based approaches of the selection of suitable matchers. Furthermore, to verify the accuracy of predictions of both parts of the proposed MOMA Framework, the AHP-based and the rule-based approaches, we evaluated them in the context of the case studies defined by the Ontology Alignment Evaluation Initiative (OAEI) 2006 Campaign. We elaborate on the applied evaluation method, report the findings resulting from the use of our MOMA Framework compared with the results achieved during the OAEI contest, and compare the findings of the manual vs. the results of the semi-automatic matcher selection. We close this section with a very brief discussion on the cost issues related to the matcher selection together with some additional remarks.

Chapter 9 In this closing chapter, we conclude our research work in the ontology matching domain and discuss the result achieved. Furthermore, for future work, we sketch the possible research directions and extensions of our MOMA Framework.

Enjoy the reading!

Chapter 2

Rudiments & Terminology

Abstract Only with a thorough understanding of the main expressions and terms is it possible to follow the ideas of our research. For this reason, we explain the main terminology to be used in this work (cf. Fig. 2.1). We start with the visions and goals of the *Web of data* – Semantic Web – and elucidate the differences between the current HTML-based Web of documents and the Semantic Web. Then we explain the term *ontology*, which is an important concept in context of the entire work. Since ontologies have been increasingly used in biology, sociology, physiology and, in recent years, in knowledge management as well as Computer Science, the definitions of ontology have varied considerably. Therefore we have highlighted the terminology to be used in the course of the dissertation. Beyond this, we introduce a definition of the key term of this work – ontology matching; however, the ontology matching domain will be elaborated in the following chapter.

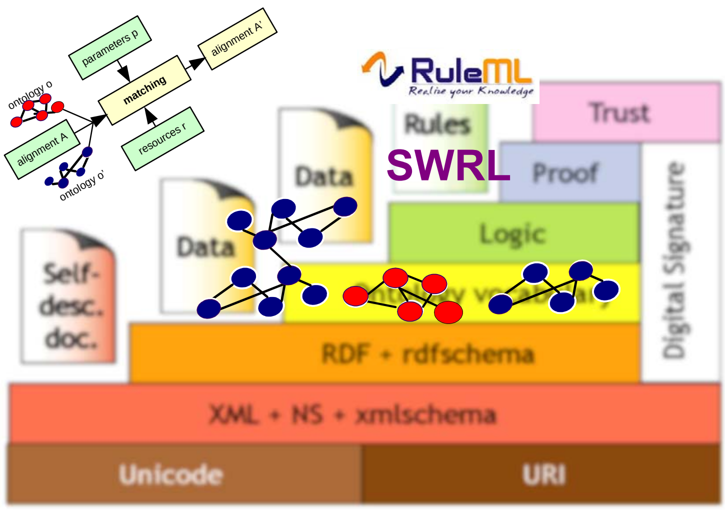


Figure 2.1: Fundamental terminology

2.1 Semantic Web

The World Wide Web can be seen as a success story, both in terms of the amount of available information and the number of people using it. The Web's success is owed to the simplicity of the underlying structures and protocols that ensure easy access to all kinds of resources [43], i.e. it is characterized by easy access to a huge amount of information. The Internet helps its users to deal with documents, though unfortunately not with the accompanying information. As a result of the continuous growth of the Internet, the search for, location, organization, and maintenance of information and knowledge (not documents) has become difficult and complex. Web pages are developed and designed by people for people, and the computer, as a machine, can only present information understandable to humans. On the other hand, existing technologies used by common search engines are usually incapable of serving the expectations of the users, delivering mismatched, irrelevant, or insufficient results, or are unable to deliver the hoped-for responses to more complex queries. For these reasons, the coming years will be characterized by the transformation of the Web from a document publication medium intended for human consumption and utilization into a medium for intelligent knowledge interchange and exchange [85], where data will be published – in addition to classic HTML pages – directly on the Web. The goal is through *"engineering the open standards that make the Web work"* by expanding the Web from *"a Web of documents towards one Web of data and services on everything for everyone."* [22]. This development, called **Semantic Web**, is the joint effort of scientific institutions (MIT[1], Stanford[2], ILRT[3] etc.) and top enterprises (e.g. HP[4], IBM[5], Nokia[6] etc.;) [155] and is led by the W3C.

> *"The Semantic Web provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries" [36]*

The Semantic Web is not an alternative to the existing Web and will not replace the existing HTML-based Internet, rather, it is to extend the Web by adding the semantic (meaning) of the information. In other words, the Semantic Web is an extension of the current Web in which information is given well-defined meaning, thus facilitating the collaboration of computers and people [8]. By supplementing well-defined meaning to the Web, information is made machine-readable and

[1]MIT - Massachusetts Institute of Technology; `http://mit.edu/`, accessed on 10.08.2008
[2]Stanford University; `http://www.stanford.edu/`, accessed on 10.08.2008
[3]ILRT - Institute for Learning & Research Technology; `http://www.ilrt.bris.ac.uk/`, accessed on 10.08.2008
[4]HP Labs; `http://www.hpl.hp.com/`, accessed on 10.08.2008
[5]IBM Research; `http://www.research.ibm.com/`, accessed on 10.08.2008
[6]Nokia Research; `http://research.nokia.com/`, accessed on 10.08.2008

machine-understandable and improves work between humans and computers. This semantic-based vision of the Internet expands the visible *human to human* Web into a Web of *human to human and machine*. The term "Semantic Web" comprises a family of technologies that enable applications to generate and automatically process metadata describing distributed Internet resources. The final vision of the Semantic Web is the use of the Web as a globally distributed database, which can be queried like a local database of today and deployed by the use of applications to perform certain tasks automatically [8]. To achieve the vision of the Semantic Web, networked resources, e.g. websites, are annotated by structured and machine-understandable metadata, which are assigned a well-defined meaning and are interpreted by means of logic rules. Thus, the Semantic Web is not a specification but a *philosophy* that solves well-defined problems with well-defined operations on well-defined data and offers a range of models and languages (e.g. XML, XML Schema, RDF, OWL, SPARQL) to handle the various aspects of document semantics under this well-defined data form [156]. The philosophy of Semantic Web follows certain important principles:

- evolution as a development principle;
- every concept can be identified with URIs;
- resources and relationships are classified semantically; and
- partial information is acceptable and absolute truth is not necessary.

The first person to come up with the idea of the Semantic Web was Tim Berners-Lee, who proposed, in 1998, in [3] the vision of a Semantic Web that enables automated information access and use based on machine-processable semantics (meaning) of data.

> *" The concept of machine-understandable documents does not imply some magical artificial intelligence which allows machines to comprehend human mumblings. It only indicates a machine's ability to solve a well-defined problem by performing well-defined operations on existing well-defined data. Instead of asking machines to understand people's language, it involves asking people to make the extra effort"* [3].

The Semantic Web is described in terms of a *layer cake* (also called *architecture stack*), which defines the reference architecture in the ongoing standardization process. The renowned stack, presented in 2001 and illustrated in Fig. 2.2, includes the fundamental technologies of the Semantic Web. The core technologies have been developed independently or even before the idea of a Semantic Web and are part of the

W3C Recommendations: XML, URI, OWL (Web Ontology Language) and RDF (Resource Description Framework). Since there have been substantial research and development investments in semantic technologies by the research organizations and the major players of the IT sector, the standardization of the fundamental layers of the architecture have now reached a very advanced state, whereas the standardization process of the upper layers (security, logic, proof, trust) is just starting.

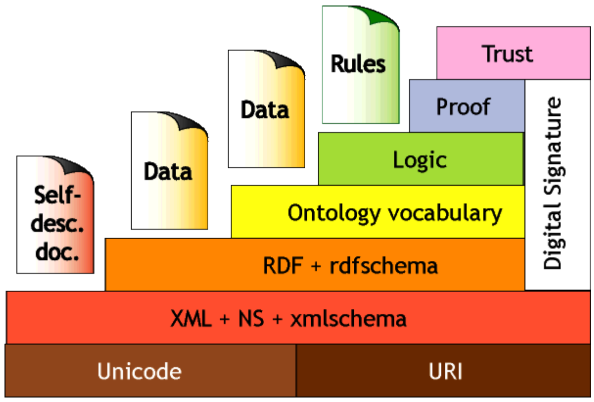


Figure 2.2: Semantic Web layer cake from 2001 [107]

Unicode/URI Layer
If any Semantic Web application is to be enabled to access and use data from other such applications, all data object and data schema must have a URI [114]. A resource, in terms of the Semantic Web, can be anything from physical objects to human beings, and from simple words to complex conceptual structures (this principle provides a measure for what can be included into the Semantic Web). URIs[7] can be used to identify definitions for concepts, especially in the context of ontologies and metadata.

XML / XML Schema Layer
The XML layer adds syntactic interoperability, guarantees the exchangeability of semantic information, and introduces the RDF/RDF Schema layer as the data interoperability layer [51]. Extensible Markup Language (XML)[8] describes the data and focuses on the kind of data it is.

[7] `http://www.w3.org/Addressing/`, accessed on 10.02.2008
[8] `http://www.w3.org/XML/`, accessed on 10.02.2008

RDF / RDF Schema Layer
Resource Description Framework (RDF)[9] allows for the description of resources and how they relate to each other. RDF specifies a data model for publishing metadata on the Web and utilizes XML as serialization syntax for data transmission. It provides a consistent, standardized way of describing and querying Internet resources, from text pages and graphics to audio files and video clips. It is a model and syntax for annotating metadata designed for the exchange of information over the Web. RDF provides (with the other standards like RDFS and OWL) syntactic interoperability between applications on the Web as well as a base layer for building the Semantic Web.

> *"If HTML and the Web made all the online documents look like one huge book, RDF, schema, and inference languages will make all the data in the world look like one huge database"* [7].

An RDF statement is the smallest unit of information expressing a unique fact and is composed of tipple: subject (resource described), predicate (property of the resource), and object (value of the property). RDF uses URIs to name resources and properties. RDFS (RDF Schema) defines and publishes the vocabulary used in the RDF data model. It is a vocabulary description language, which does not provide actual application-specific classes and properties but provides the framework to describe them. Both RDF and RDF Schema are based on XML and XML Schema. The existence of standards for describing data (RDF) and data attributes (RDF Schema) enables the development of a set of readily available tools to read and exploit data from multiple sources [114].

Ontology Layer
Ontologies formally name and describe the central concepts of an application domain and the relationships between them. According to Gruber's definition, an ontology is an explicit formal specification of shared conceptualization [73] and prevents misunderstandings and ambiguities. Ontologies are usually employed as a main representation of the semantic knowledge, and are semantic models for knowledge exchange between human and machine actors that specify a common domain representation defining restrictions of interpretation possibilities of symbols.[10] The most recent development in standard ontology languages and part of the W3C Recommendations is the Web Ontology Language (OWL)[11]. It extends RDFS with terms

[9] RDF/XML Syntax Specification, W3C Recommendation 10.02.2004; `http://www.w3.org/TR/rdf-syntax-grammar`, accessed on 02.10.2008

[10] More details regarding ontologies incl. further definitions and applications can be found in Sec. 2.2.

[11] Web Ontology Language (OWL): `http://www.w3.org/2004/OWL/`, W3C Recommendation 10.02.2004: `http://www.w3.org/TR/owl-features/`, all accessed on 02.10.2008

and concepts for an even more expressive knowledge representation in the form of ontologies. OWL has three incrementaly-expressive sublanguages, OWL Lite, OWL DL, and OWL Full, designed for use by specific communities of implementers and users.[12]

Logic / Proof / Trust Layers

The upper layers represent some of the most complex technical challenges faced by the Semantic Web, and are still at the initial stage of development (a pre-standardization phase) while simple application demonstrations are being constructed. The trust layer defines mechanisms upon which applications decide whether or not to trust the given information, while the logic layer provides the means to specify sets of deductions that can be made from a collection of data together with formalisms for describing steps taken to reach conclusions from given facts. Furthermore, according to Tim Berners-Lee's vision, we can talk about the Semantic Web bus [4], as illustrated in Fig. 2.3, which is based, on the one hand, on ontologies and data, and on the other hand, on logic with rules.

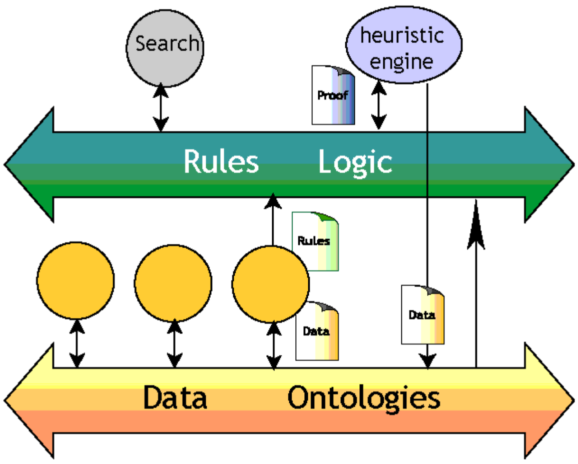


Figure 2.3: Semantic Web bus [4]

[12]Considering the family of the ontology languages, the more specific languages like languages for describing Web Services: Web Service Modeling Ontology (WSMO) and Semantic Markup for Web Services (OWL-S) could also be mentioned here. However, since they are not the main scope of our research work, we do not dwell upon this issue. The interested reader is referred to the corresponding working groups and their web pages: WSMO: `http://www.wsmo.org/`, `http://www.w3.org/Submission/WSMO/`; OWL-S: `http://www.w3.org/Submission/OWL-S/`, all accessed on 24.06.2008).

A *rule axiom* is constructed in the form of an implication between an antecedent (body) and a consequent (head), each of which consists of a (possibly empty) set of atoms. The formula of the rule is:

$$\texttt{antecedent} \rightarrow \texttt{consequent} \quad (2.1)$$

where both antecedent and consequent are conjunctions of atoms written $a_1 \wedge \cdots \wedge a_n$.

Informally, the intended meaning can be read as: Whenever the conditions specified in the antecedent hold (are true), then the conditions specified in the consequent must also hold. An empty antecedent is treated as trivially holding (true), and an empty consequent is treated as trivially not holding (false). Non-empty antecedents and consequents hold if all of their constituent atoms hold [92]. An antecedent/consequent is an assertion, for instance, `parent(x,y)`, meaning `x` is a parent of `y`:

```
Implies(Antecedent(parent(x,y)), Consequent(older(x,y)))
```
(2.2)

means if `x` is a parent of `y`, then `x` is older than `y`.

Rules, as mentioned before, always express a conditional, with an antecedent and a consequent component. When the consequent specifies an action, the aim of satisfying the antecedent is to schedule the action for execution. When the consequent defines a conclusion, the effect is to infer the conclusion [81]. In the Semantic Web world, there are two main rule languages: RuleML and SWRL, which actually are being developed simultaneously with the Semantic Web.

Rule Markup Language (RuleML) RuleML aims to provide a shareable, XML-based rule markup language for rule storage, interchange, and retrieval [90]. RuleML has been developed by the Rule Markup Initiative[13] and can be seen as a step towards defining a shared and modular language specification. The initiative obtains the transformations from and to other rule standards/systems and coordinates the development of tools to elicit, maintain, and execute RuleML rules. It permits both forward (bottom-up) and backward (top-down) rules in XML for deduction, rewriting, and further inferential-transformational tasks, as well as collecting use cases, e.g. on business rules and reactive services.

Semantic Web Rules Language (SWRL) Although OWL is gaining industry-wide acceptance, it is not expressive enough for all applications. Semantic Web

[13] `http://www.ruleml.org/`, accessed on 06.09.2008

Rules Language (SWRL) [14] is used instead to tackle this difficulty [1]. SWRL, originally called ORL (the OWL Rules Language), is based on a combination of the OWL DL and OWL Lite sublanguages of the OWL Web Ontology Language with the Unary/Binary Datalog RuleML sublanguages of the RuleML. As the name implies, SWRL is intended to be the rule language of the Semantic Web. It provides Horn-like rules for both OWL-DL and OWL Lite, including a high-level syntax to represent these rules, and is more powerful than either OWL-DL or Horn rules alone [92]. Since SWRL is based on OWL, all rules are expressed in terms of OWL concepts: classes, properties, individuals, and literals. To provide a formal meaning for OWL ontologies that do not include rules written in the syntax provided by SWRL, there is a model-theoretic semantics that provide generalization of data models (relational or semi-structured) and can deal with uncertain or vague information and identifiable objects [1]. SWRL has significant expressive power: It allows much more expressiveness with properties (binary predicates), negation, and chaining, and includes numerous built-in relations. In addition, one may assert equivalences as well as implications in SWRL. To check if a rule has been satisfied (or if it fires), we need SWRL reasoners.

An example in natural language notation:

If John has Mary as a parent and Mary has Bill as a brother, then this rule requires that John has Bill as an uncle.

Example in SWRL notation ("human-readable notation"):

$$parent(?a, ?b) \wedge brother(?b, ?c) \rightarrow uncle(?a, ?c). \tag{2.3}$$

Even if SWRL is more expressive than RuleML, there is still much work to be done in the future. The current issues and development directions of SWRL are outlined in [182].

[14] `http://www.w3.org/Submission/SWRL/`, accessed on 06.09.2008

Since Tim Berners-Lee's first presentation of the Semantic Web layer cake in 2000 [5], the architecture stack has been modified with the development of the core technologies, and thanks to the effort led by W3C in collaboration with a large number of researchers and industrial partners, work is underway on the development of Semantic Web technologies. The architecture stack has also been adapted with respect to the evolving requirements. The last version of the Semantic Web architecture also includes, besides the above mentioned layers, the query languages (e.g. SPARQL[15]) layer and the user interface and applications layer (cf. Fig. 2.4).

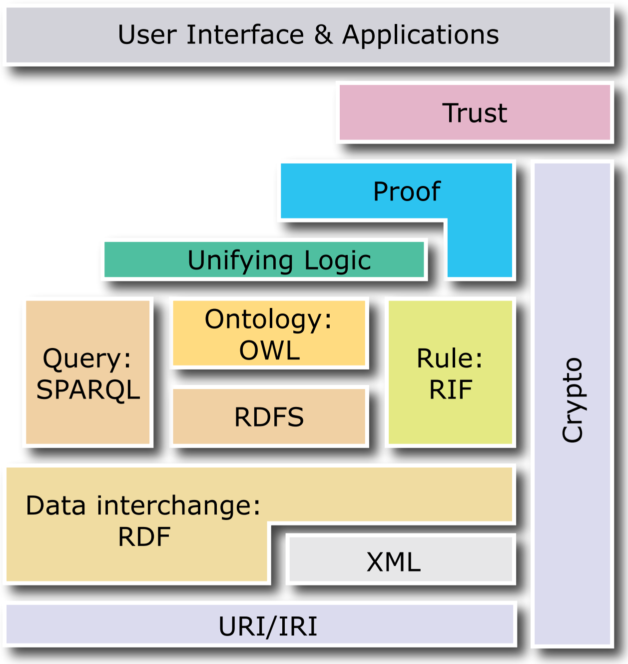


Figure 2.4: Semantic Web layer cake 2007 [22]

Currently, the Semantic Web technologies are maturing and moving out from academic applications into the industrial sector. This is demonstrated by the strong and growing interest of various business sectors like human resources and employment, health care and life sciences, transport, logistics and energy [116] and, on the other hand, by public bodies like the European Commission, which has supported the distribution and transfer of these technologies into the business world in vari-

[15] SPARQL - Query Language for RDF, W3C Candidate Recommendation 14.06.2008; `http://www.w3.org/TR/rdf-sparql-query/` accessed on 18.08.2008

ous European projects like OntoWeb[16], REWERSE[17], SDK project cluster on SEKT[18], DIP[19], and Knowledge Web[20] as well as SUPER[21], and ACTIVE[22].

2.2 Different Views on Ontologies

Although the term "ontology" has a long tradition, especially in philosophy, and has been widely discussed, it seems to have generated much controversy recently since it was adapted for psychology, sociology, computer and information science and since people from these communities often use the term "ontology" with different, partly incompatible meanings in mind [87]. In the following, we introduce the main definitions of the term "ontology" whose meanings depend on the domain in which the ontology is to be applied. We also highlight the usage of ontology in the context of our research within Computer Science.

Philosophical ontology
According to [172], ontology as a branch of philosophy is the science which defines the kinds and structures of objects, properties, processes, relations, etc. in every area of reality. According to the definition in the "Encyclopedia Britannica" [23], an ontology is the theory or study of being as such; it is an area of philosophy that deals with the nature and organization of reality. Traditionally, issues on existence and the state of being were answered by metaphysics, a discipline which goes back to Aristotle and refers to fourteen treatises dealing with what he called "first philosophy". Philosophical ontology looks for the description (not explanation) of the terms of a classification of entities in the universe and can also be called *descriptive* or *realist ontology*. Philosophical ontology describes the categories available inside a given domain of interest and can be unitized into a *formal ontology*, which is a formal theory of non domain-specific entities, attributes, items, or domains, and *material on-*

[16] OntoWeb - Ontology-based Information Exchange for Knowledge Management and Electronic Commerce; http://www.ontoweb.org, accessed on 24.06.2008

[17] REWERSE - Reasoning on the Web with Rules and Semantics Project; http://rewerse.net/, accessed on 24.06.2008

[18] SEKT - Semantic Knowledge Technologies; http://www.sekt-project.com, accessed on 24.06.2008

[19] DIP - Data, Information, and Process Integration with Semantic Web Services Project; http://dip.semanticweb.org/, accessed on 24.06.2008

[20] Knowledge Web EU Network of Excellence; http://knowledgeweb.semanticweb.org, accessed on 24.06.2008

[21] SEUPER - Semantics Utilised for Process management within and between EnterPrises; http://www.ip-super.org/, accessed on 24.06.2008

[22] ACTIVE - Enabling the knowledge powered enterprise; http://www.active-project.eu/, accessed on 15.09.2008

tology, also called *regional ontology*, which is concerned with domain-specific terms, concepts etc. In other words, there is a formal ontology — the ontology of part and whole, of identity and difference, dependence and independence — and there are particular domain specific or regional ontologies, for example, ontologies of geography, medicine, or ecology [173]. Generally, the philosopher-ontologist attempts to establish the truth about reality by finding an answer to the question of existence.

Sociological ontology

> *"Any way of understanding the world, or some part of it, must make assumptions (which may be implicit or explicit) about what kinds of things do or can exist in that domain, and what might be their conditions of existence, relations of dependency, and so on"* [166].

Such an inventory of the categories of being and their relations is, from the *sociological view*, an ontology. In this sense, each specific science, including sociology, may be said to have its own ontology (e.g. individuals, institutions, relations, norms, roles) depending on the particular sociological theory under consideration [166].

Physiologist's ontology

The *physiologists*, in contrast, define ontology as the branch of metaphysics devoted to the study of the nature of being or existence, or the essence of things, including the distinction between reality and appearance and whether mathematical entities exist outside of the mind [35].

Ontology for computer scientists

Computer scientists as well have taken on this term for their needs and use it for describing the domains they wish to implement. In the world of information systems, an ontology is a software (or formal language) artefact designed with a specific set of uses and computational environments in mind. In this context, ontologies aim at capturing domain knowledge in a generic way and provide a commonly agreed understanding of a domain, which may be reused and shared across applications and groups [29]. Ontology is (very largely) qualitative and deals with relations, including the relations between entities belonging to distinct domains of science, as well as between such entities and the entities recognized by common sense. An ontology is often something that is ordered by a specific client within a specific context and in relation to specific practical needs and resources [173]. With applications in fields such as knowledge management, information retrieval, natural language processing, e-Commerce, information integration, and the emerging Semantic Web, ontologies are part of a new approach to building intelligent information systems [60]. They are intended to provide knowledge engineers with reusable pieces of declarative knowledge, which can be — together with problem-solving methods and reasoning services — easily assembled into high quality and cost-effective systems [140]. Thus, a major representation of the semantic knowledge in Computer Science are ontologies that are explicit, formal specification of shared conceptualization [74] (cf. Fig. 2.5).

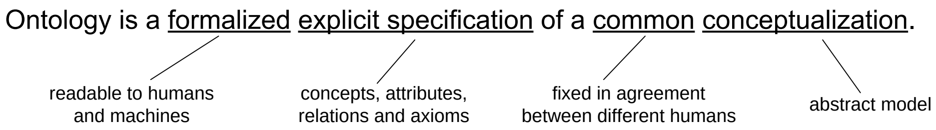


Figure 2.5: Definition of the term *ontology* in the context of Computer Science

Ontologies are targeted to provide the means to formally specify a commonly agreed upon understanding of a domain of interest in terms of concepts, relationships, and axioms, as well as a common vocabulary of an area, and to define — with different levels of formality — the meaning of the terms and of the relations between them [70, 73]. While, according to [87], in computer science, researchers assume that they can define the conceptual entities in ontologies mainly by formal means, in information systems, researchers discussing ontologies are more concerned with understanding conceptual elements and their relationships, and often specify their ontologies using only informal means. Furthermore, as stated in [174], we can distinguish between domain, generic, application, representational, and task ontologies:

domain ontologies capture the knowledge valid for a particular type of domain (e.g. electronic, medical, life science) and are being developed by communities of domain experts (e.g. the Gene Ontology by the Gene Ontology Consortium).

generic ontologies (core / upper ontologies) are valid across several domains; they define general concepts and relationships between them and are currently being standardized by international standardization committees (e.g. SUMO by the IEEE Standard Upper Ontology Working Group)

application ontologies contain all the necessary knowledge for modeling a particular domain (usually a combination of domain and method ontologies that provide terms specific to a particular problem-solving method).

representational ontologies do not commit to any particular domain. Such ontologies provide representational entities without stating what specifically should be represented. A well-known representational ontology is the *frame ontology*, which defines concepts such as frames, slots, and slot constraints that allow the expression of knowledge in an object-oriented or frame-based way.

task ontologies provide terms specific to particular tasks (e.g. hypotheses belong to the diagnosis task ontology).

Ontologies in conjunction with semantic mapping and translation techniques can be used in a semantic integration task to describe the semantic of the information sources and to make the content explicit [188]. At this point, it must be said that most researchers agree that semantic integration is one of the most serious challenges for the Semantic Web today [147]. Furthermore, in some projects, ontologies take on additional tasks, such as querying model and verification, and support the validation and adaptation of machine-processable data [198]. Ontologies support the adaptation of searches specific to the knowledge of the actor and context as well as more relevant search results (with ranking). Since ontologies conceptualize a part of the real world, they can serve as a means of communication between users of an application domain.

In recent years, an ever growing number of ontologies have been developed in numerous computer scientific fields like knowledge management, information retrieval, personalization, multimedia, software engineering, and Web Services. At the same time, the dissemination of ontologies across these research communities has resulted in a plethora of tools and methodologies to build, manage, and maintain ontologies. According to these methodologies[23], engineering ontologies are defined as an iterative process that reflects major similarities to established models from the neighboring research field of Software Engineering. In general, we can say there are six main phases in the ontology building process:

- domain analysis (requirements analysis and knowledge acquisition)
- domain conceptualization
- ontology implementation
- ontology evaluation
- ontology refinement
- ontology maintenance

[23]Cf. [71] for a description of and a comparison among the most relevant ontology engineering methodologies

The first step in the process is usually the definition of the domain and the ontology purpose. Such information can be gathered from domain experts using the question-answer method by means of natural language or machine learning tools [72]. Another method [160] manually extracts common domain knowledge from various sources and integrates it into a target ontology. In this step, most methods try to satisfy in some measure the "Ontologist's Credo" [173], which recommends that the ontology engineer needs to be to some extent familiar with the scope of the ontology. After specifying the domain and purpose of the ontology, e.g. in an ontology requirements specification document [179], the domain knowledge in question must be conceptualized by means of ontological primitives. The specifications of ontology purposes, competency questions, design guidelines, and knowledge sources are the main sources of information for this step. Current methodologies adopt slightly different positions w.r.t. the techniques for conceptualizing ontologies at the knowledge level: in [193] the methodology requires precise, unambiguous text definitions for concepts and relationships, while in [72] the role of axioms in specifying the definitions of the ontological terms and constrains for their interpretation is stressed.

The result of the ontology engineering phases is a preliminary ontology, which is refined in a fine-tuning process towards a target ontology. The evaluation step checks whether the target ontology satisfies the requirements developed in the first phase of the ontology building process and implements eventual refinements until the requirements are fulfilled.

In spite of the fact that ontology engineering is now considered a mature discipline with regard to the Semantic Web, outfitted with a variety of methodologies and tools to build, manage, and merge ontologies [61, 179], most of the currently available ontologies are not aligned with any specific methodology. Instead, they are the results of some application- and domain-dependent building process. It is generally accepted that building an ontology is a challenging, time-consuming and error-prone process. And yet the development of new ontologies fails to tap the full potential of existing domain-relevant knowledge sources.

> *"We need to be able to get the reuse of the data, which we have founded"* [6].

Typically, when implementing an ontology-based application, the underlying ontology is built from scratch, does not look to available ontological knowledge on the Web, and is implicitly tailored to specific application needs, which in turn means that it is not built to be reused in different settings.[24]

[24] The reuse of existing knowledge sources in the context of ontology building, together with our experiences in this area, is addressed in Sec. 5.1.1

Though ontologies and associated ontology management tools have become increasingly popular in recent decades, the dissemination of ontologies, ontology-driven technologies in real-world business contexts, and ontology-based applications as envisioned by the Semantic Web community requires sensitive methodologies to address both technical and economic challenges of ontology engineering. In this context, a further major aspect of ontology engineering is the issue tied to the economic challenges of ontology engineering. Unlike adjacent engineering disciplines, current Semantic Web research ignores several aspects of the engineering processes that are fundamental in real-world business contexts. A significant factor for the success of the Semantic Web and ontology technologies in the industrial sectors is cost (time and money) reduction. In this regard, and in order for ontologies to be built and deployed on a large scale beyond the boundaries of the academic community, one needs not only technologies to assist the implementation process, but also proved and tested means to control the costs of the overall engineering process. Besides feasible tool support, one of the major requirements is the availability of methods, like the *Ontology Cost Model ONTOCOM* [170, 171], that address the business-oriented aspects of ontology development and deployment.

2.3 Ontology Matching - Definition

The most crucial term in our entire research is the term "ontology matching" and for this reason, before we go over to the next chapter that elaborates the state of the art regarding the ontology matching domain and describes different matching approaches, their classifications, and main issues (cf. Sec 3.1), we need to formally define the term "ontology matching". We talk about **ontology matching** when we are talking about a process of finding relationships or correspondences between entities of different ontologies. In particular, it takes two (or more) ontologies, each consisting of a set of discrete entities (which can be classes, properties, rules, predicates, etc.) as input and determines as output the relationships (e.g., equivalence, subsumption) between these entities [54]. The **matching process**, as stated in [56], can be seen as *"a function f, which, from a pair of ontologies to match o and o', an input alignment A, a set of parameters p, and a set of oracles and resources r, returns an alignment A' between these ontologies:"*

$$A' = f(o, o', A, p, r) \tag{2.4}$$

In some cases, one may need to match more than two ontologies. In this case, we are talking about the **multiple matching process**, which, as stated in [56], can be defined as *"a function f, which, from a set ontologies to match* $o_1, ..., o_n$*, an input alignment* A*, a set of parameters* p*, and a set of oracles and resources* r*, returns an alignment* A' *between these ontologies:"*

$$A' = f(o_1, ..., o_n, A, p, r) \tag{2.5}$$

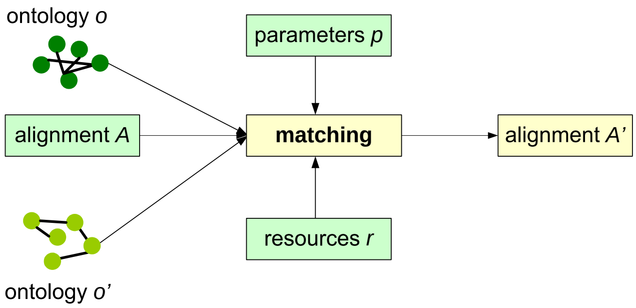


Figure 2.6: Matching process [56]

In the rest of the work, when we talk about ontology matching[25], we assume that the matching process, according to the abovementioned definitions, can occur between two or more ontologies (cf. Fig. 2.6). Furthermore, in the following we will use the terms ontology matching, matchers, and matching approaches interchangeably.

[25] Detailed elaboration on ontology matching domain can be found in Chapter 3.

Chapter 3

Ontology Matching

Abstract Semantic technologies provide a standardized infrastructure to create, use, and exchange machine-understandable information. Among the prerequisites for the realization of this revolutionary concept are matching techniques that are capable of handling the open, dynamic, and heterogeneous nature of Semantic Web information in a feasible manner. This chapter discusses the current situation in the ontology matching domain with particular emphasis on the significant role of matching approaches in relation to the realization of the Semantic Web vision. Furthermore, we provide an overview of the main matcher classifications and adumbrate a few of the existing matchers while taking into account the diversity of these approaches. Even if numerous methods and algorithms have been developed to solve specific issues regarding the Semantic Web, none of these solutions can be deployed for all problems in the ontology matching domain. We outline some of these crucial short, medium, and long-term issues that need to be resolved, i.e. overcoming problems of interpretability and heterogeneity, before the global Semantic Web can be accomplished.

"Matching is the process of finding relationships or correspondences between entities of different ontologies."

Jérôme Euzenat & Pavel Shvaiko: *Ontology Matching*, Springer, 2007

3.1 Ontology Matching Domain

Over the past years, a considerable increase in data and knowledge and, connected with it, growth of the Internet, especially Web services, has compounded the need for semantic interoperability across heterogenous sources [206]. The level of effort is linear in the number of matches to be performed, a growing problem given the soaring number of web data sources and e-businesses to be integrated. A faster and less labor-intensive integration approach is needed to address this issue [158]. Since not only volume but also the size of the real-world data (there are often quite large sources with thousands of concepts and properties) play crucial roles in terms of the variety of applications, the manual identification of the correspondences between given sources has become a virtually impossible task. The matching issue has become a fundamental step in numerous applications like data integration, e-business and semantic query processing [158]. Aside from this, in recent decades an ever-growing number of ontologies have been developed and deployed in numerous computer science fields like knowledge management, information retrieval, personalisation, multimedia, software engineering, Web Services, and the Semantic Web.

The Semantic Web envisions a Web of machine-understandable information which can be automatically accessed by and exchanged among semantics-aware applications. Furthermore, given the decentralized nature of the development of the Semantic Web, 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 [123], while other ontologies will have overlapping domains. In other words, a fully developed Semantic Web will contain numerous, distributed, and ubiquitously available ontologies, and the next generation of Web Services will, as a result of its open design, employ ontologies to describe service capabilities and to mediate interprocess communication. In addition, users will be able to choose among different ontologies to enable mediated access to Web information, or to integrate or transform them into application-specific, customized models. However, to integrate data from disparate ontologies, we must know the semantic correspondences between their elements [47].

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 algorithm occupies the key role in facilitating the global success of the Semantic Web because of its implications in almost every phase of an ontology engineering and management process:

- ontology merging, where a new ontology is created from two source ontologies;
- ontology alignment – a set of correspondences between two or more ontologies; it is an output of the matching process;
- ontology mapping – a directed version of alignment; and
- ontology transformation – a process of expressing the entities of an ontology with respect to the entities of another ontology; the result of this process is a third ontology whereby the two source ontologies remain unmodified [56].

Furthermore, ontology matching is a crucial task to enable interoperation and interoperability between Web applications using different but related ontologies [95]; matching techniques are necessary to provide semantic mappings among ontologies for discovery and the sharing of knowledge and data. Additionally, a fundamental requirement for the realization of the global Semantic Web vision is the use of tested and proved ontology matching algorithms capable of dealing with the *heterogeneity of current ontological sources* available on the Web in terms of representation languages, varying degrees of maturity and granularity levels, natural languages, and divergent views of the modeled domains. The importance of the matching issue is also reflected by the large number of matching algorithms that have been proposed [45, 47, 69, 124, 129, 130, 153, 176], and the variety of the adaption of such algorithms in various areas of computer science like Web Service, P2P, or Grids [28]. The number of approaches participating in the ontology alignment evaluation campaigns (OAEI 2004, 2005, 2006, 2007) organized since 2004 by the Ontology Alignment Evaluation Initiative (OAEI)[1] has also been constantly increasing (cf. Fig. 3.1).

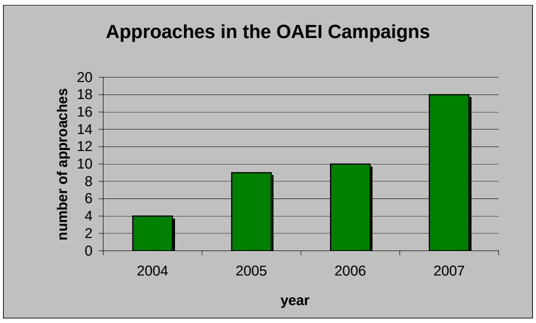


Figure 3.1: Number of matching approaches participating in the OAEI contest

[1] `http://oaei.ontologymatching.org/`, accessed on 10.08.2008

The number of use cases for ontology matching, as described in [54], the comprehensive studies, surveys, and classifications on this topic, given for example in [44, 157, 168], as well as the still growing number of publications (reports, workshops, conferences, and journal papers) related to the ontology/schema matching and semantic matching (cf. Fig. 3.2[2]) are further proof of the significance and justify the great importance of this topic in the Semantic Web.

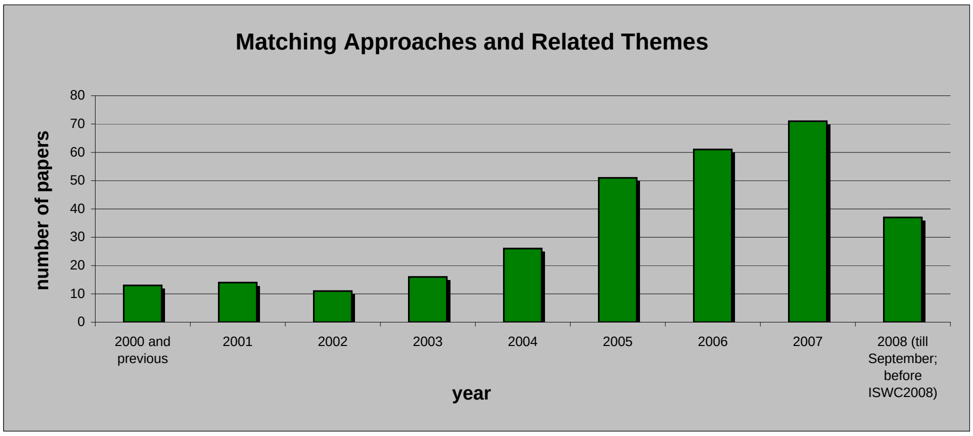


Figure 3.2: Approx. the most relevant matching publications in the period before 2000 to September 2008

Above all, such initiatives as (i) Ontology Matching[3], which propagates the research results and awareness of existing matching efforts, (ii) the Information Interpretation and Integration Conference (I3CON)[4], which was the first in what has become a series of workshops supporting the ontology and schema interpretation and integration research communities, and (iii) the abovementioned Ontology Alignment Evaluation Initiative (OAEI), which has organized evaluation campaigns and whose aim is to evaluate ontology matching technologies, acknowledge not only how fundamental and relevant the matching issue is for the development and propagation of the Semantic Web vision, but also show how problematic and challenging it is.[5]

Considering all these facts, we agree with the authors of the new book "Ontology

[2] Data for the figure from the `http://www.ontologymatching.org/publications.html` accessed on 30.09.2008

[3] `http://www.ontologymatching.org` accessed on 10.08.2008

[4] `http://www.atl.external.lmco.com/projects/ontology/i3con.html` accessed on 15.08.2008

[5] The problems and difficulties within the ontology matching domain are elaborated in Sec. 3.5.

Matching" [56] who expect work on matching issues not only to continue, but to increase in the upcoming year, mostly because of the growing interest in the semantic heterogeneity tasks from both research and industry.

3.2 Classifications of Matcher Approaches

Matching conceptual structures, be they database schemes, XML schemes, conceptual graphs or, as mentioned before, more recently Semantic Web ontologies, is a discipline with a long tradition, which plays a significant role in various areas of computer science such as data integration, data warehouses, agent communication, and Web Service composition. In the following, we provide a general overview of the existing classifications of matching approaches, though we do not claim completeness with regard to the plethora of ways the matchers can be grouped.

3.2.1 Syntactic vs. Semantic Matching

The most general classification of the matching approaches is the categorization that takes into account the way matching elements are computed and the type of similarity relation used. In this context, we distinguish between (cf. Fig 3.3):

syntactic matching, whose key distinction is to map labels (of nodes) and use syntax driven techniques and syntactic similarity measures to look for similarity; in this kind of matching, semantics are not analyzed directly but semantic correspondences are searched for only on the basis of syntactic features [168], and

semantic matching, whose key feature is to map meanings (concepts) and not labels; the mappings are calculated on the basis of schema/ontology elements (e.g., nodes of graphs) by computing semantic relations like equivalence (=), more general ($\sqsupseteq$), less general ($\sqsubseteq$), mismatch ($\perp$), and overlapping ($\sqcap$), instead of computing coefficients, rating match quality in the [0,1] range. [69].

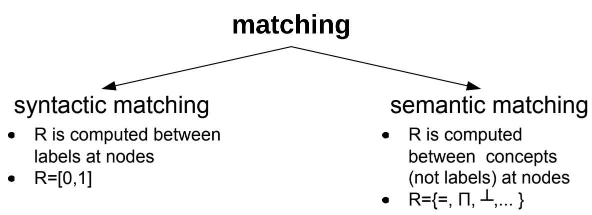


Figure 3.3: Syntactic vs. semantic matching [69]

3.2.2 Bernstein & Rahm Classification

The best known matcher classification is the Bernstein & Rahm classification, which distinguishes between *individual matchers*, which compute a mapping based on a single matching criterion, and *combining matchers*, which use multiple individual matchers [132]. Though this classification was originally intended for schema matching approaches, it can be also applied to systems performing matching between ontologies. We follow the notation form [168] that uses the more general term "matching approaches" instead of "schema matching approaches" from the original publication (cf. Fig. 3.4).

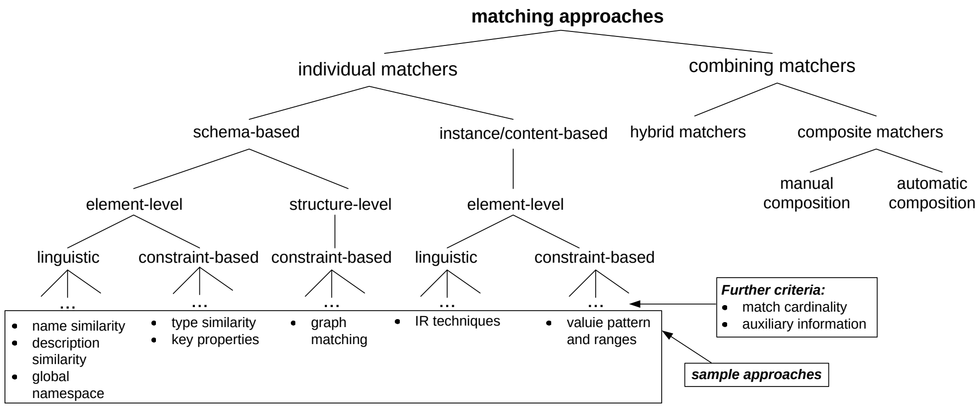


Figure 3.4: Classification of matching approaches[157]

Individual matchers, applying only a single method of matching items e.g. linguistic or taxonomical matchers, can work on instance data (*instance/contents-based matchers*) or consider only structure information, be they relationship types, data types, or schema structures (*schema-only based matchers*). Both algorithms can be applied on individual schema elements such as attributes or concept labels (*element-level matchers*). In addition, schema-only based approaches can deal with combinations of schema elements such as complex schema structures, thus mappings can be computed by analyzing subgraphs (*structure-level matcher*). A single element-level matcher uses linguistic, as well as constraint-based techniques, while a schema-only based matcher uses only the latter.

Combining matchers combine individual matchers so as to overcome their limitations by proposing hybrid and composite solutions; they are divided into two categories: *composite matchers* and *hybrid matchers*. Composite matchers (e.g. GLUE[47],COMA[45], CMC[186]) combine the different results of independently executed matchers whereby the order of the execution of the individual matchers can be assign manually (*manual composition*) or (semi-) automatically (*automatic composition*) so that flexibility of the approaches increases. In contrast, a *hybrid matcher* (e.g.Cupid[124]) does not allow such manual intervention and follows a black box paradigm, in which various individual matchers are melted together to form a new algorithm with a fixed combination of multiple match criteria. This kind of matcher is difficult to extend and enhance.

3.2.3 Shvaiko's Classification

The next possibility regarding how matcher approaches may be categorized is based on the two classifications mentioned previously. It is important to add that this classification, however, concentrates on schema-based approaches that consider only ontology information, not instance data. If we consider the *semantic* matching approaches, it will be beneficial to treat the semantics captured in ontologies at different levels of details. Since there is a need to distinguish between ontology matching techniques that rely on diverse semantic clues, as stated in [168], we can differentiate between techniques with either a heuristic or formal ground (cf. Fig. 3.5).

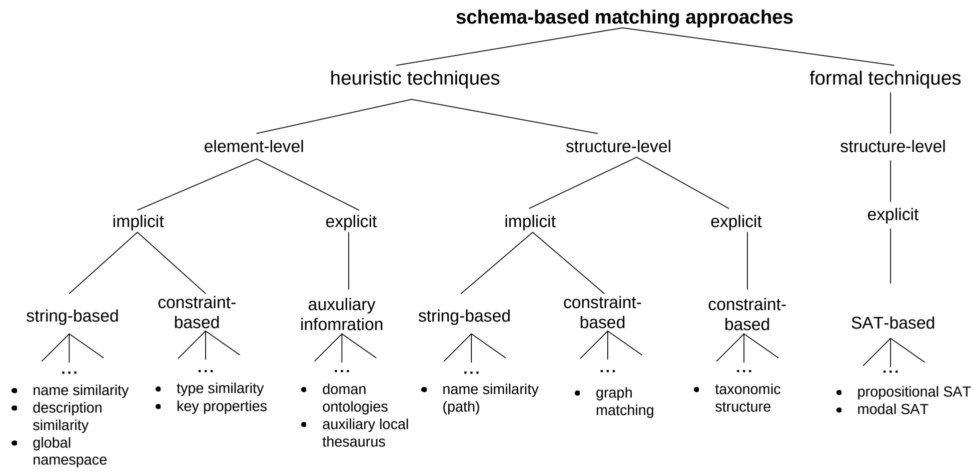


Figure 3.5: A revised classification of schema-based matching approaches [168]

Heuristic techniques try to guess the underlaying relations between similar labels or graph structures. In contrast, *formal techniques* have model-theoretic semantics which are used to justify their inferences [168]. Furthermore, we can differentiate between syntax driven *implicit techniques* (e.g. techniques that consider labels as strings or analyze data types) and *explicit techniques* that exploit the semantics of labels and are based on tools that explicitly codify semantic information (e.g. thesauruses, WordNet[6]).

3.2.4 Three-layer Classification

All three classifications mentioned before are based on three different backgrounds (the first (cf. Sec. 3.2.1) distinguishes the matchers on the basis of similarity types, the Bernstein & Rahm classification (cf. Sec. 3.2.2) differentiates between individual and combining matchers, while Shvaiko's classification (cf. Sec. 3.2.3) distinguishes between diverse semantic clues) and serve as a foundation for the next classification of the elementary matching approaches (cf. Fig. 3.6). This classification includes three main layers [56]:

granularity/input interpretation layer is mainly based on the Bernstein & Rahm classification including element and structure-level matchers as well as classification based on the structure or semantic matchers; in this case, the techniques that exploit the external resources of a domain and common knowledge in order to interpret the input have been categorized under the separate term "external";

basic techniques layer distinguishes between the elementary matching techniques and is motivated by the way a particular method interprets the input information (e.g. a label can be interpreted as a string or word while a hierarchy can be interpreted as graph or taxonomy);

kind of input layer differentiates between the type of input that can be served by a particular algorithm. In this context, we distinguish between terminological methods that can be string-based or linguistic, structural approaches that consider either internal structures (structural) or relations of entities with entities (relational), and algorithms that can work on models (semantics) or data instances (extensional).

Depending on the reader's focus, the grouping can be read both from top down and bottom up. Thus, to reach the middle layer — basic techniques — we can focus on

[6] `http://wordnet.princeton.edu`, accessed on 27.08.2008

the way techniques interpret input information (top down direction) or concentrate on how objects are manipulated (bottom up reading) [56].

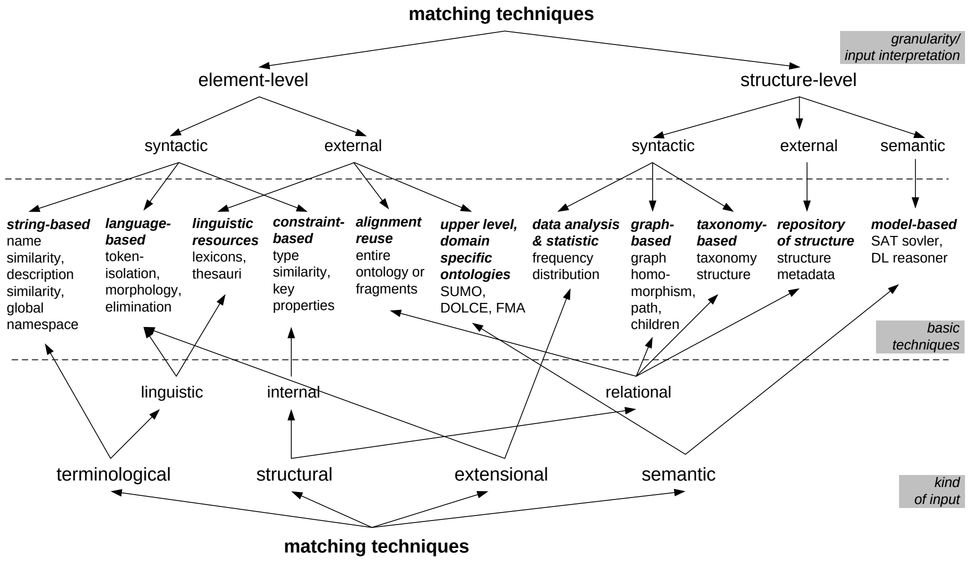


Figure 3.6: A retained classification of elementary matching approaches [56]

All the classifications mentioned provide a common conceptual basis for matching approaches and can be used for the general description of matching systems. They can also help to develop new matching systems or elementary approaches that take into account the existing solutions. However, to be able to analytically compare matching algorithms, a more precise matcher characteristic is needed, and the existing classifications can only serve as one of the basic information sources for the development of the detailed characteristic.

3.3 Compliance of Measures

The most prominent criteria to evaluate the quality of matching approaches are *precision* and *recall*, which have their seeds in information retrieval [196] and have been adapted for the ontology matching issues. According to [56], precision and recall are based on the comparison of the resulting alignment A with a reference alignment R, effectively comparing which correspondences are discovered and which are not (cf. Fig. 3.7). Following this definition:

precision measures the ration of correctly found correspondences (true positive) over the total number of returned correspondences (true positive and false positive); it is meant to measure the degree of correctness of the approach[56]:

$$\texttt{Pr(A,R)} = \frac{|R \cap A|}{|A|} \tag{3.1}$$

recall measures the ratio of correctly found correspondences (true positive) over the total number of expected correspondences (true positive and false negative); it is meant to measure the degree of completeness of the alignment [56]:

$$\texttt{Re(A,R)} = \frac{|R \cap A|}{|R|} \tag{3.2}$$

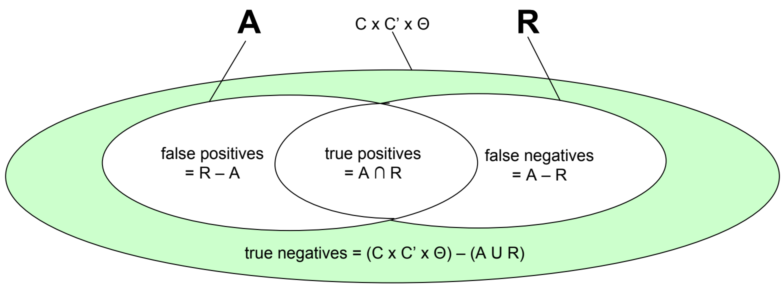


Figure 3.7: Two alignments as a set of correspondences and relations between them [56]

3.4 Existing Matching Approaches

In the following, we give a brief overview of some existing matching approaches, specifically, the most common approaches, as well as matchers that will be used to evaluate our research (cf. Chapter 8). Many more approaches have been left out of this survey; it was neither feasible nor practical to include everything that has been developed to date.

Anatomical Ontology Alignment System (AOAS/NLM/NIH) to aligning concepts is an automatic, rule-based approach, and operates at the schema level, generating mostly point-to-point mappings. The approach uses a combination of domain-specific lexical, structural, and semantic techniques, and utilizes domain specific knowledge (lexical knowledge from external resources, knowledge augmentation, and inference techniques) [205].

AUTOMS (HCONE): Automated Ontology Mapping through Synthesis of methods is an approach for the automatic alignment of ontologies. It exploits the HCONE-approach to ontology merging [110] that uses lexical (or syntactic), structural, semantic (or domain) knowledge and makes use of the intended informal meaning of concepts by mapping them to WordNet senses using the Latent Semantic Indexing (LSI)[7] method [111]. Furthermore, AUTOMS integrates the HCONE-merge method with a lexical matcher COCLU (COmpression-based CLUstering)[195] as well as with matching heuristics that exploit structural features of the source ontologies [109].

COMA++ extends the previous prototype COMA (COmbination of MAtching algorithms) [45]. It is a customizable and generic tool for matching both schemas and ontologies specified in languages such as SQL, XML Schema or OWL. [127]. Since it provides an extensible library of matching algorithms, it is able to parallel composite matchers (utilizing a composite approach to combine different match algorithms) [2]. COMA supports different applications and scheme types (like XML and relational schemes) and provides an extensible library of matching algorithms, a component for combining the results obtained, and extensive functionality for the evaluation of matching effectiveness. One of the weak points in COMA comes, however, from the possibility that the suggested combined methods may prove to be inadequate for complex situations. Since each base matcher performs differently in different conditions, simple, predefined composition methods are incapable of capturing such performance variation [186].

Cupid implements a hybrid matching algorithm comprised of linguistic and structural schema matching techniques, and computes similarity coefficients with the assistance of a pre-compiled thesaurus [167]. Cupid (i) includes automated linguistic-based matching; (ii) is both element-based and structure-based; (iii) is biased toward similarity of atomic elements (i.e. leaves), where much schema semantics is captured; (iv) exploits internal structure, but is not overly confused by variations in that structure; (v) exploits keys, referential constraints, and different views; and (vi) makes context-dependent matches of a shared type definition that is used in several larger structures [124].

[7] `http://www.cs.utk.edu/~lsi/`

DSSim (MAOM) is an ontology mapping system that is used with a multi-agent ontology mapping framework in terms of question answering. The main objective is to assess how applying the belief function can improve correctness of the ontology mapping by combining the similarities originally generated by both syntactic and semantic similarity algorithms. DSSim aims to dynamically produce alignments at runtime. Though its goal is to be monitored through human interaction, the DSSim algorithm is an iterative closed loop which creates mapping without human interaction [139].

Falcon/Falcon-AO is an automatic tool for aligning ontologies based on linguistic matchers (lexical comparison and statistic analysis) and matchers based on graph matching for ontologies that measures the structural similarity between graphs [144]. Falcon provides technologies for **F**inding, **A**ligning, and **L**earning ontologies, and ultimately for **C**apturing knowledge by an **ON**tology-driven approach. The new version of the approach contains three elementary matchers: (i)V-Doc uncovers alignments by revealing the usage (context) of the domain entities in the ontologies to exploit their intended meanings; (ii) I-Sub is a lightweight matcher based on the string comparison; and (iii) GMO uses RDF bipartite graphs to represent ontologies and measures the structural similarities between the graphs by the similarity propagation between domain entities and statements. All three types of matchers are seen as independent components that make up the core matcher library of Falcon-AO. To be able to cope with large-scale ontologies, an ontology partitioner, PBM [96], has been integrated [95].

GLUE is a system that employs learning techniques to semi-automatically create semantic mappings between ontologies [47] and is strongly related to the earlier system *LSD* used in schema integration, i.e. for finding mappings between various local schemas based on the same mediated schema.

ISLab HMatch is a system for dynamically matching distributed ontologies. It can be dynamically configured for adaptation to the semantic complexity of the ontologies to be compared, using a combination of both syntactic and semantic techniques. The matching configuration of ISLab HMatch is selected in an automated way according to a matching policy embedded in the incoming request. Additionally, with HMatch, it is possible to determine one-to-one and one-to-many mappings [27]. The similarity analysis is performed with affinity metrics to determine a concept semantic affinity measure in the range [0,1]. ISLab HMatch supports ontology matching, schema matching, version matching, and directory matching [95].

JHU (APL/Onto-Mapology) is a a Johns Hopkins University Applied Physics Lab ontology mapping software that integrates techniques based on string/text matching, structure/graph matching, and semantic (rule-based/logic-based) matching. It allows users to combine these techniques as desired or to use a hybrid algorithm. This system is dedicated to OWL ontologies [9].

OWL Lite Aligner (OLA) is an ontology matching system designed to balance the components that make up an ontology, e.g., classes, constraints, and data instances. Its goal is to integrate all aspects of OWL-Lite while successfully dealing with cyclic definitions. OLA exploits, besides string-based element-level matchers, also a matcher based on WordNet, iterative fix-point computation. The similarity between nodes of the graphs is based on two foundamental elements: the category of nodes considered, e.g., class, property, and the features of this category, e.g., superclasses, and properties [57, 55]. OLA is designed with the idea of balancing the contribution of each component that composes an ontology (classes, properties, names, constraints, taxonomy, and even instances). In particular, OLA is a family of distance-based algorithms that converts definitions of distances based on all the input structures into a set of equations [169].

OWL-CtxMatch (OCM) has been designed to match OWL DL ontologies. It is an OWL specialized version of the CtxMatch algorithm designed as a general algorithm to discover semantic relationships across distinct and autonomous generic structures. Since OCM performs semantic matching and, as a result, is able to recognize a broad range of relationships between matched entities, i.e. not only equivalence but also subsumption, disjointness, and intersection, the main requirement it imposes on the structures being matched is the necessity to label them with natural language labels [141, 142].

PRIOR is an ontology mapping system based on **P**rofile p**R**opagation and **I**nf**O**rmation **R**etrieval techniques, i.e. the profile of a concept = the concept's name + label + comment + property restriction + other descriptive information. PRIOR exploits both linguistic and structural information to map small ontologies, while integrating classic information retrieval (Indri search engine[8]) to process large ontologies [125].

RiMOM is a tool for ontology alignment that combines different strategies and aims to find the optimal alignment results: edit-distance based strategy, statistical-learning based strategy, and three similarity-propagation based strategies. Each

[8] `http://www.lemurproject.org/indri`, accessed on 28.08.2008

strategy is based on one kind of ontological-information/ approach. Depending on their label and structure similarity factors, the algorithm will favor one or the other kind of strategy, and for this purpose, it uses heuristic rules [184, 203].

SimilarityFlooding (SF) is a hybrid matching algorithm based on the ideas of similarity propagation. Schemas are presented as directed labeled graphs; the algorithm manipulates them in an iterative fixpoint computation to produce mappings between the nodes of the input graphs. The technique uses a syntactic string comparison mechanism of the vertices' names to obtain an initial mapping, which is further refined within the fix-point computation [69, 130].

S-Match is a schema-based schema/ontology matching system that implements a semantic matching approach. It is moreover a hybrid system which performs composition of element level techniques. As output, it delivers semantic relations (equivalence, more general, less general, mismatch, and overlapping) between nodes of graphs that correspond semantically to each other. S-Match libraries contain 13 element-level matchers and 3 structure-level matchers [167].

Considering the classifications of matching approaches that have been introduced in Section 3.2, the matching approaches mentioned above can be summarized in following tables:

- Tab. 3.1 is based on the Bernstein & Rahm classification,
- Tab. 3.2 concentrates on the three-layer classification and additionally
- Tab. 3.3 considers the different inputs of the algorithms.

Schema-based matchers	**Instance-based matchers**	**Mixed matchers**
AOAS (NLM/NIH), AUTOMS (HCONE), Cupid, COMA++, DSSim (MAOM), ISLab HMatch, OCM (OWL-CtxMatch), SF, S-Match	GLUE	OLA, Falcon, RiMOM, PRIOR

Table 3.1: Matchers overview based on the Bernstein & Rahm classification

	Element-level		Structure-level		
Matchers	**Syntactic**	**External**	**Syntactic**	**Semantic**	**External**
AOAS (NLM/ NIH)		linguistic resources (lexical knowledge)	structural knowledge	semantic techniques, rules based techniques	structural knowledge
AUTOMS (HCONE)	language-based (LSI)	linguistic resources (WordNet), domain specific knowledge	taxonomy based (structural features)		
COMA++	string-based, language-based, data types	auxiliary thesauri (synonyms, hypernyms, abbreviations) alignment reuse	DAG (tree) matching		
Cupid	string-based, language-based, data types, key properties	auxiliary thesauri (synonyms, hypernyms, abbreviations)	tree matching weighted by leaves		
DSSim (MAOM)	string-based (concept-name similarity, property name and set similarity)	linguistic resources (WordNet)	graph-based	yes	
Falcon	string-based, language-based	linguistic resources (WordNet)	graph based, structural affinity		
GLUE	constraint-based (domain constrains)	alignment reuse (earlier mapping results)	taxonomy-based (hierarchical structure)		
ISLab HMatch	language-based	linguistic resources (common thesaurus)			
JHU (APL)	string based		graph based	rule-based/logic-based	
OLA	string-based, language-based, data types	linguistic resources (WordNet)	iterative fix-point computation, matching of neighbors, taxonomic structure		
OCM	string-based, language-based	linguistic resources (WordNet)		based on description logic	
PRIOR	string-based, language-based (descriptive information)		graph-based, taxonomy-based (hierarchical structure)		
RiMOM	string-based	linguistic resources (WordNet)	taxonomic structure, similarity propagation		
SF	string-based, datatypes, key properties		iterative fix-point computation		
S-Match	string-based, language-based	linguistic resources (WordNet), sense-based, gloss-based		propositional SAT	

Table 3.2: Matchers overview based on the three-layer classification (extended table from [56])

	Representation language of the matcher input				
Matchers	**Classifications & taxonomy**	**Relational schema**	**XML schema**	**RDF**	**OWL**
AOAS (NLM/NIH)				yes	yes
AUTOMS (HCONE)					yes
COMA++		yes	yes		yes
Cupid		yes	yes		
DSSim (MAOM)				yes	yes
Falcon				yes	yes
GLUE	yes	yes	yes		
ISLab HMatch					yes
JHU (APL)				yes	yes
OLA				yes	yes
OCM	yes				yes
PRIOR				yes	yes
RiMOM					yes
SF		yes	yes		
S-Match	yes		yes		yes

Table 3.3: Matchers overview regarding representation language of the input

3.5 Open Issues within Ontology Matching Domain

Despite its pervasiveness, today, ontology matching is still largely conducted by hand, in an extremely tedious, time-consuming, labor-intensive and error-prone process. As stated in [47], manual matching has now become a key bottleneck in building large-scale information management systems. The same authors argue that the advent of technologies such as the WWW, XML, and the emerging Semantic Web will further fuel information sharing applications and exacerbate the problem. In this context, although the development of tools to assist in the ontology matching process has become crucial for the success of a wide variety of information management applications, there are still, as stated in [146], many steps in the process that could be automated, many points where a tool could make reasonable suggestions, and many conflicts and constraint violations for which a tool could check. It means that there are still a number of crucial short, middle, and long term problems that need to be resolved in order to overcome the interpretability and heterogeneity issues and to come up to realization of the vision of a fully developed Semantic Web.

No overarching matching

Many methods and tools are under development to solve specific problems in the Semantic Web; however, none of these solutions can be deployed for all the existing problems. This statement is also true in the ontology matching field, in which there is no overarching matching algorithm for ontologies capable of serving all (heterogeneous) ontological sources; there is no matching "one fits all". Most of the research in this area proposes new approaches based on different principles and relies on various features. These new approaches solve only small parts of the "greater" problem in the matching domain or are mere "stop gaps" [65].

In Sec. 5.1.2 we outline two case studies as an example of the lack of an overarching matching approach.

"Unused" reuse

It is generally accepted that reuse and reusability of existing ontological sources are very important issues for cost-effective and high-quality ontology engineering. This situation is a consequence of the process in which ontology engineers have recognized the potentials of the "reuse strategy" and have proposed, utilized, and tested a wide range of methodologies [61] for building ontologies from existing sources [16, 74, 150, 181, 193]. Although the development and existence of tried and tested ontology matching algorithms and support tools will be one of the crucial issues that may have a significant impact on future development, for instance, of the vast semantic web-based information management systems, the ontology matching, mapping, and alignment field still does not take seriously the alternative solution, which is to reuse existing approaches during the development of semantic-based applications. Consequently, the reuse of these semantic-based approaches have not yet been analyzed satisfactorily within the Semantic Web realm. Our experiences collected during the development of ontology-based applications [13, 66, 143] confirm previous findings in the literature that building such applications is still a tedious process, due to the lack of tested and proved support tools, and that reuse of existing methods within new application contexts is currently not extensively discussed. When implementing an application using a matching approach, the corresponding algorithm is typically built from scratch, and only small, marginal attempts to reuse existing methods are made.

"Evil" diversity

Since much time and effort have been spent on the development of new ontology alignment and matching algorithms, the collection of such algorithms is still growing (cf. Sec. 3.4). For this reason, those interested in ontology matching researchers, developers, and other users are confronted with a problem: there is an enormous amount of divergent work from different communities who claim some sort of relevance to ontology mapping, matching, alignment integration, and merging [102]. Given this multiplicity, it is difficult to identify the problem areas and comprehend the solutions provided. In this view, the diversity of matching approaches is a weakness rather than a strength. Part of the problem is also the lack of a comprehensive survey, a standard terminology, obscure assumptions or undisclosed technical details, and the dearth of evaluation metrics [102].

In Sec. 5.1, we elaborate the valuable ideas embedded in current matching approaches, and on the other hand, we outline the limitations of the approaches which, for instance, have often been emphasized in the recent literature and in course of collaborating with matching experts.

"Holes" in the approaches

Despite an impressive number of research initiatives in the matching field, current matching approaches, as emphasized in recent literature [168, 69, 130, 124], still feature significant limitations. For example, the majority of existing approaches to ontology matching are (implicitly) restricted to processing particular classes of ontologies and are thus unable to guarantee a predictable quality of results on arbitrary inputs. Furthermore, current matching approaches, though containing valuable ideas and techniques, are tailored to certain types of ontologies and are confined to specific schema types and representation languages [44]. Beyond this, they need to be customized for a particular application setting (like schema or data integration), they assume that a common, or, at least to a large extent, overlapping universe of discourse [33] (e.g. ontology merging), cannot be applied across various domains with the same effect, and do not perform well (or have not been tested yet) on inputs with heterogeneous (graph) structures or on large-sized inputs. Most approaches are restricted to finding 1:1 matching correspondences [130] and, finally, they lack exhaustive testing in real world scenarios and real-world practice.

In Sec. 5.1.2 we describe two case studies illustrating how problems differ sufficiently so that various matchers can or cannot be applied. In particular, we will show that even if in both cases matchers tackling relatively large incoming sources are needed, matchers suitable for the one case would not be appropriate for other.

Furthermore, in Sec. 5.3, we give a brief overview of the results of the matcher survey conducted in the course of our research highlighting some limitations the approaches.

Missing infrastructure

As stated in [206], after years of extensive research and the development of numerous matching approaches, it is time to reap some of the procedures, techniques, and tools gained. The successful deployment of semantic technologies like ontology engineering tools or matching/mapping/merging approaches require that they become off-the-shelf tools, easy to integrate by experts as well as laypeople. In this context, as stated in [56], users (researchers, developers, and others interested in ontology matching) and demanding help in choosing an appropriate matcher or combining the most appropriate matchers for their particular use.

> *"There needs to be some general infrastructure in place where one can easily register, access and use various things such as: ontologies, mappings between ontologies, mapping languages, and translation engines. At a minimum, it should be possible to issue requests such as: "Given this message, encoded using ontology A, please return a translated message encoded using ontology B. And please use this particular mapping and that particular translation engine". Such infrastructure is currently lacking"* [189].

Additional issues

Aside from the problems mentioned above, there are many other aspects of a global nature that need to be resolved. There is the question of what should be matched based upon what needs to be found. Moreover, it is important to avoid performing relatively blind matching while aware of when to stop the matching process. It is also necessary to adapt the systems, i.e. adjust them, not to the data to be processed, but to the issue that needs to be resolved with the particular matcher. Furthermore, open issues like large-scale matching evaluation, discovering missing background knowledge for matching process, user involvement, social and collaborative ontology matching, matcher selection and self-configuration as well as explanation of the matching results that have been identified as main challenges for ontology matching during the 3rd International Workshop on Ontology Matching (OM-2008)[9] collocated with the 7th International Semantic Web Conference ISWC-2008 confirm the diversity of the matching problems and the importance of their step-by-step solving.

[9] `http://om2008.ontologymatching.org/`, accessed on 01.11.2008

Though we most definitely will not be able to "solve all these problems and save the world" in our research work, we are tackling some of these issues (some more strongly than others). We are concentrating on the selection of suitable matching approaches, which, in our opinion, is one of the main issues in the ontology matching domain. By proposing a framework that supports the selection of a relevant matching algorithm suitable w.r.t the given specification while taking into account the definition of the appropriate criteria for this decision making process, we are definitely addressing "missing infrastructure" and "evil diversity" issues and, in some measure, the problems related to "unused" reuse, "holes" in the approaches and "no overarching matching".

Chapter 4

MOMA Framework

Abstract Current matching algorithms cannot be optimally used in ontology matching tasks as envisioned by the Semantic Web community, mainly because of the inherent dependency between approaches and ontological properties. In this chapter, we introduce a possible solution to this problem – a Metadata-based Ontology MAtching (MOMA) Framework (cf. Fig. 4.1) 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 approaches for application. The 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. This can be achieved by applying a flexible tool based on decision-making, on the one hand, and dependency-rules statements, on the other, to support both manual and automatic selection process, respectively.

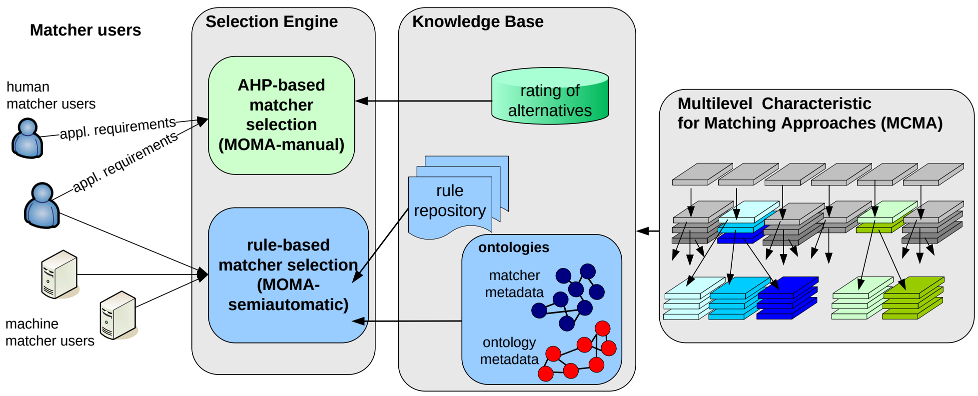


Figure 4.1: MOMA Framework

4.1 Framework Overview

Since the existing matching algorithms cannot be optimally used in ontology matching tasks as envisioned by the Semantic Web community, we need a strategy to remedy the weaknesses and take advantage of the particularity of the various approaches in the selection of suitable matchers. To tackle this issue, the "matching problem" must be clearly seen as a collection of small subproblems, which are dependent on various sets of criteria and circumstances. Furthermore, due to the fact that there is no "general" matching issue, there is also no "general" way to solve the matching issues by only posing the query "find a matching algorithm for two ontologies and deliver a set of relations"(cf. Sec. 3.5). This general query covers indeed every type of ontology and matching algorithm while also giving the same basic information regarding alignments; however, it does not address the specific requirements of a particular application.[1] In our opinion, the existing matching algorithms should be evaluated with respect to the given application requirements prior to their execution, the best approaches should be chosen, and, if necessary, adapted to the particular system. In this context, our approach does not intend to propose a new matching technique in any of the categories presented so far (cf. Sec. 3.2,3.4) but to focus on developing a novel matching strategy which aims to choose existing matching algorithms depending on the given context, i.e. considering characteristics of the ontological inputs, output, and application requirements. Our proposed methodology takes into account the difficulties and limitations of current matching algorithms while, at the same time, considering their advantages. Instead of conjunctively using various matchers in order to increase applicability – a technique employed in hybrid and composite matching approaches – we accept the dependency between existing algorithms and their input sources and extrapolate it to identify the most appropriate matching algorithms for a given set of specific ontology inputs and application context. In comparison to the combined frameworks, our approach allows for a more flexible triggered selection of matching algorithms, depending on their suitability, for instance, to particular phases of the ontology management process. In particular, our strategy is based on dependencies between algorithms and the types of ontologies in which the former are able to process successfully, and factors that influence the matching tasks. For example, for two (or more) characterized ontologies to be matched, having a definition of the problem to be solved along particular requirements regarding the final application, our methodology determines which matching algorithms satisfy the respective specification so as to obtain the desired output. Our methodology has been imple-

[1] Examples on Semantic Web-based applications and their requirements regarding matching approaches are discussed in Sec.5.1.2.

mented in the form of an ontology matching framework that serves as the first step in the reuse-strategy of existing algorithms, since reuse typically involves selection as a one of the main process phases [112]. The necessity of such a selection strategy and the "market" for tools supporting matching issues have been pointed out in the analysis of the existing matching approaches and open issues within the matching field and, as before mentioned, were also confirmed during the 3rd International Workshop on Ontology Matching (OM-2008) (cf. Sec. 3.5).

In the course of this chapter, we present the central requirements regarding the framework by means of the "matcher gaps" presented in Section 3.5 and introduce the main use cases that need to be served by the framework. At the end, we describe the components of the MOMA Framework in the context of the high-level architecture.

4.2 Metadata-based Ontology Matching Framework

To tackle the problem of heterogeneity of existing ontology matchers, and to circumscribe the disadvantages of the particular matcher, a reuse strategy of matching approaches based on the examination of their characteristics is needed. Hence, the aim of our research is develop a methodology and implement it in the form of a framework which, considering the suitability of matching approaches regarding the particular application context, realizes our vision of selecting suitable matching approaches. In specific, we have developed a **Metadata-based Ontology MAtching (MOMA) Framework** based on a reuse-paradigm that, given a set of ontologies to be matched, takes into account the capabilities of existing matching algorithms and suggests appropriate matchers for a given application. Our MOMA Framework deals with the limitations of existing matching approaches by being aware of the link between matching algorithms and the ontologies for which they have been originally designed or to which they have been successfully applied. The framework uses additional information about the ontologies and data about available matching algorithms in order to determine which of the latter are appropriate in a defined context.

The term "framework" has been intentionally chosen for our work since the approach proposed in this thesis is not limited to the ontology matching selection problems but can also be seen as a general and generic selection framework, which, after corresponding adaptations, could be applied to other selection issues. In the case of application of the framework in the context of selection problems different from matching issues, we obviously would not call it MOMA Framework but simply "metadata-based selection framework". The further applications of the developed framework are not the main scope of this work, nevertheless, a possible utilization of the framework together with some required adaptations will be briefly described in Sec. 9.2.

Before we devote our attention to the use cases and high-level architecture of the MOMA Framework, we concentrate on the framework requirements, which have been based on the collection of open issues within ontology matching domain as explained in Sec. 3.5.

4.2.1 Requirements

Requirements and requirement specifications are the crucial issues, not only in the preliminary phase but for the entire software development process. According to [49], software requirement is:

(a) a software capability deployed by the user to resolve a problem or to achieve an objective;

(b) a software capability that must be met or possessed by a system or a system component to satisfy a contract, standards, or other formally imposed documentation.

As stated in [115], the requirements, which are often problem statements, come from potential/future users and other stakeholders of the system. Furthermore, a part of the work is to elicit the requirements from the users, inherent in which is the assumption that stakeholders have demands that must be elicited, analyzed for consistency, feasibility, and completeness, and formulated as requirements. Hence, we have decided to base the analysis of framework requirements on the issues elaborated in the Section 3.5. These issues are the main problems within matching domain which we focus on and solve (at least partially) with our MOMA Framework (cf. (a)). We have analyzed the different matching "gaps" and, in the course of this section, elucidate which of them we have addressed in the MOMA Framework and to what extent. In general, by proposing the framework that supports the selection of matching algorithms that are suitable w.r.t the given specification while taking into account the definition of the appropriate criteria for the decision making process, we have contributed to solving the "missing infrastructure" and "evil's diversity" issues and, to some measure, have addressed the problems related to "unused" reuse, "holes" in the approaches, and "no overarching matching".

no overarching matching Due to the heterogeneity of the ontological sources and the diversity of applications in which matching approaches are to be applied, it is generally known that an overarching ontology matching algorithm capable of serving every ontology type and language, delivering the desired matching results, and dealing with excessive application requirements will not be realized in the foreseeable future. In our opinion, a possible solution to this matching dilemma is to design a new matching strategy that strives to optimize the matching process (by reusing existing approaches) while being aware of the inherent dependencies between matchers, their execution characteristics, required output, and the types of ontologies they are able to process successfully. Our methodology and framework are based on this strategy.

"evil" diversity 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 matching by giving them the chance to get an overview of the still growing collection of such algorithms. For this reason, we propose a tool/framework that not only provides a simple list of existing matchers and/or matcher development initiatives but also gives recommendations regarding suitable matching approaches for a given case. Our MOMA Framework can support researchers, developers of semantic web-based applications and people interested in utilizing ontology matching algorithms by analyzing their expectations and needs and, in turn, by selecting the most suitable approaches to the application.

"unused" reuse *"Would you be the first passenger on an airliner whose parts have just come out of the R&D shop? Or would you prefer to board knowing the aircraft was designed and constructed with parts that have successfully kept planes airborne for years? Practicing engineers, whether they specialize in civil, mechanical, electrical or aerospace engineering, select from components whose characteristics have been tested and proven for safety and efficiency"* [202]. We noticed that developers of ontology-based applications traditionally "reinvent" matching approaches, coding new algorithms rather than reusing existing components. This runs counter to what is practiced in ontology engineering and other areas of software development. Since software reuse can enhance the quality of components, increase the productivity of developers, and reduce development time [117], we recognize the potentials of the "reuse strategy" for matching approaches and have decided to apply this modus operandi for the matcher issue. Proposing a solution based on the selection of approved algorithms, our matching strategy, which has been implemented in our MOMA Framework, is the first step to the exploitation (and circumscription of the disadvantages) of the existing matching approaches.

missing infrastructure As mentioned before, what is required are techniques and tools capable of dealing with different ontological sources [26] and able to satisfy the requirements of emerging applications. MOMA Framework facilitates selection and later could, in turn, allow the access to and use of matchers. Hence, with help of MOMA, the existing matching approaches, which have been tried and tested in particular domains, applications, and tasks, can be applied to related domains, applications, and tasks. Furthermore, users like researchers, developers, and others people interested in ontology matching find help in choosing an appropriate matcher for their particular needs.

holes in the approaches To tackle the "holes" in the approaches, our methodology should consider, rather than neglect or ignore, the problems and limitations of the different matchers. In particular, to cope with the weaknesses of the matchers, our MOMA Framework exploits the valuable ideas embedded in current matching approaches, but at the same time takes into account their limitations – for specific input ontologies, it optimizes the matching results by sorting the candidate matching regarding their suitability for the given case.

Fundamentally, the MOMA Framework 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.

4.2.2 Main Use Cases

During discussions with semantic web-based application developers, researchers, and experts in the ontology matching field, we noticed that there two different types of users interested in the application of matching approaches and the utilization of relevant supportive tools. Consequently, as there are various groups of users who can potentially deploy our framework, we have made a conscious decision to ensure that our MOMA Framework serves both developers and computer scientists, supporting them in their implementation and research work, but also the matching providers, enabling them to utilize our framework in different services tasks. To this end, we have classified the users of the MOMA Framework into two main groups: *human users* and *machine users*. After having identified the user types, we focused our concern on certain requirements and ways of interaction specific to each group of MOMA users (cf. Fig 4.2). And so we have identified:

- *human matcher users* (e.g. ontology engineers, semantic web-based application developers) who, for instance, wish to match two (or more) source ontologies to create a new target ontology based on a combination of existing knowledge resources. Considering the ontology development process, which is mostly not conducted by people with a high level of expertise in the ontology matching domain, there is a need for means to aid them in selecting and applying ontology management tools, including matching algorithms. The process of choosing the suitable (w.r.t the given requirements) approach in this case can occur in both modes, manually as well as (semi-)automatically.

- *machine matcher users* (e.g. service/matching providers, search engines looking for similar ontologies) that play a core role in enabling Web Service interactions. In this case, the process of choosing a suitable matching approach is envisioned to be performed (semi-)automatically.

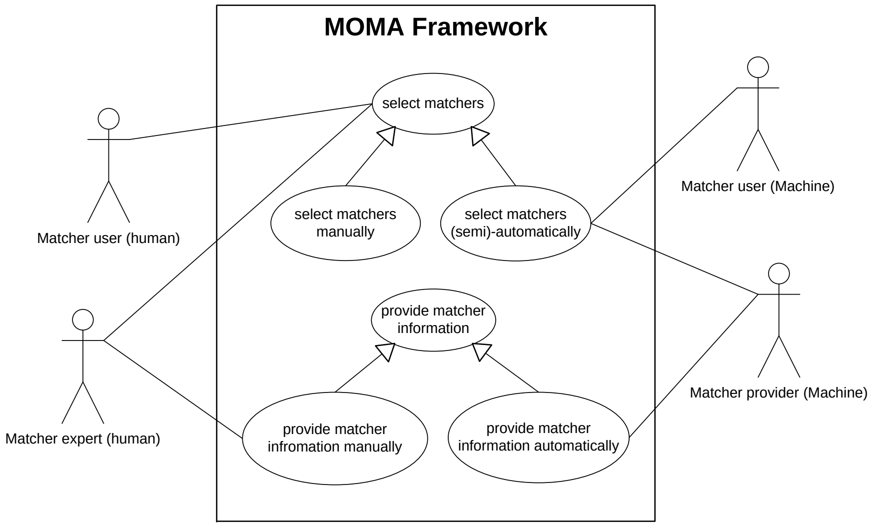


Figure 4.2: MOMA use case diagram

The objective of the MOMA Framework is to supply a tool that offers methods to support the manual as well as (semi-)automatic decision-making process regarding the applicable matchers. Furthermore, to be able to make a decision regarding the relevance of matching approaches, the MOMA Framework needs additional knowledge about matching algorithms, incoming ontological sources, required output, and the application where the matcher is to be deployed. At this point, there are two ways to collect such information: as illustrated in Fig 4.2; data can be provided manually by matcher experts or be automatically delivered by information providers. In our prototypical implementation, some data, e.g. regarding the characteristics of different matching approaches, is gathered by human experts while others, like ontology metadata, can be delivered through the application of an automatic approach.[2]

4.2.3 High-level Architecture of the MOMA

The MOMA Framework, as illustrated in Fig. 4.1, consists of three main components:

MCMA is the **M**ultilevel **C**haracteristic for **M**atching **A**pproaches utilized to describe the matching algorithms, their incoming sources, and feasible output, together with application features in which the matching approach is to be applied;

Knowledge Base includes information (based on the MCMA structure) regarding existing matchers that may be selected for application and sources that are to be matched; it also contains some rule statements that describe the dependencies between the matching approaches and ontologies;

Selection Engine is responsible for the matcher selection which, depending on the mode (manual or semi-automatic), conducts the matcher determination process based on the AHP-approach or predefined rules, respectively.

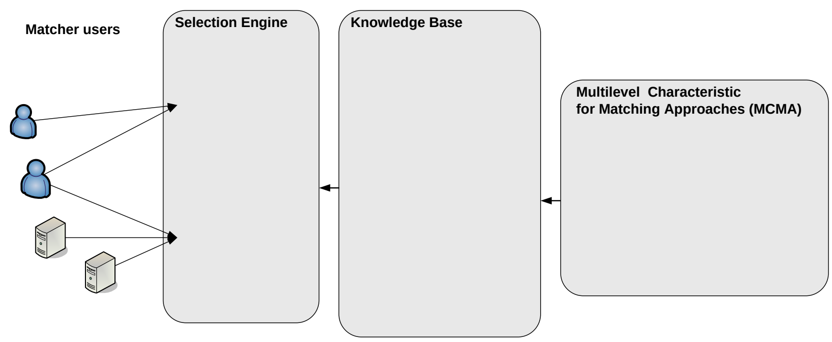


Figure 4.3: Main components of the MOMA Framework

In the following, we concentrate on the main components of the MOMA Framework, analyzing it from the perspective of the manual and semi-automatic modes

[2]For further details concerning the collection of the requisite information, the reader is referred to Sec. 6 and 7.

of selection; the description of the matcher characteristic that has been developed to be utilized in both the manual and the automatic selection processes is followed by the manual and the (semi-)automatic detection of suitable matching algorithms.

Matcher characteristic

Finding the most appropriate matching systems for a particular application is a difficult task, because there are so many different systems and so many different (application)characteristics. In our opinion, 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 various features of matching approaches (together with input, output, costs, etc.) and targeted application, identified those that have an impact on the selection of an appropriate matching approach, and finally built a matcher characteristic that serves as the basis for the final decision regarding the suitability issue (cf. Fig. 4.4). This development process detailed in Chapter 5 results in the a hierarchical tree of features that we call a **Multilevel Characteristic for Matching Approaches** or **MCMA**.

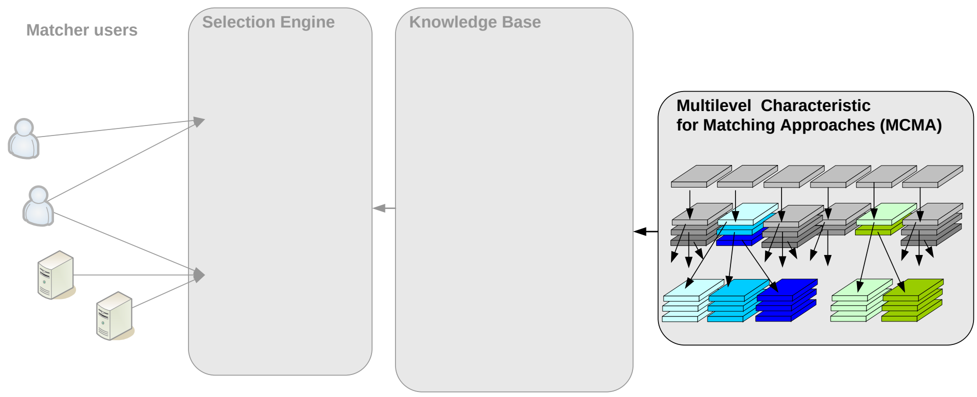


Figure 4.4: Characteristic of matchers within MOMA Framework

Manual approach

To allow the manual selection of matching approaches and thereby serve the human matcher users, we have adopted one of the approaches from the Multi Criteria Decision Analysis (MCDA) family – the Analytic Hierarchy Process (AHP) [165], which uses pairwise comparisons along with a semantic and ratio scale to assess the decision maker's preferences [76]. AHP provides a mathematically rigorous application and a process for prioritization and decision-making. It also allows decision makers to model a complex problem in a hierarchical structure to show the relationships of the goal objectives (find suitable matchers), sub-objectives (our MCMA), and alternatives (different matchers) [63]. AHP allows for:

- the gathering of knowledge about a particular problem to be solved with the requisite matching approach (*appl. requirements* in Fig. 4.5 provided by the framework users),
- the collecting of the quantification of subjective opinions (*rating of alternatives* in Fig. 4.5),
- the comparison of alternatives (*AHP-based matcher selection* in Fig. 4.5) in relation to established criteria (*MCMA* in Fig. 4.5).

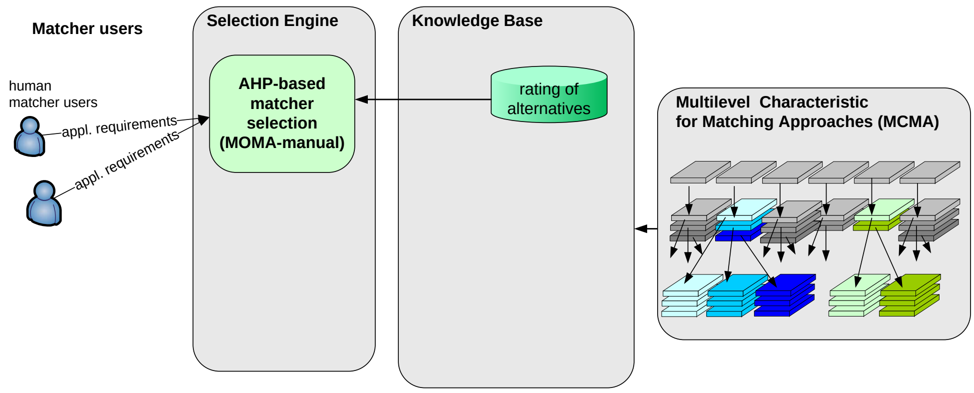


Figure 4.5: MOMA Framework - manual mode

Generally, by reducing complex decisions regarding the suitability of the matching approaches to a series of pairwise comparisons using the defined MCMA and synthesizing the results, decision-makers arrive at the optimal decision based on a clear rationale [164]. In our case, it means that the human users of the MOMA Framework obtain a list of matchers ordered by their suitability to the given context.[3]

[3]For more details, reader is referred to Chapter 6.

Semi-automatic approach

In order to serve the machine matcher users, we need to support a (semi-)automatic matcher selection process. As one possible solution for this issue, we propose a matching framework based on metadata (defined in the form of ontologies) and rules. In other words, to determine automatically which algorithms suit the concrete inputs, explicit knowledge is needed concerning the dependencies between these algorithms and the structures in which they operate. We have formalized this knowledge in terms of dependency rule-statements that determine which elements (in this case which matchers) are to be used or excluded (e.g. if no instance data is available, then apply only scheme matchers or apply only matchers that are capable of dealing with the representation language of the inputs). However, since the characteristic of existing matching approaches cannot be easily collected automatically and the matcher developers and/or matcher experts are needed to provide the required information (cf. Sec. 7.4.1), the matcher selection process can effectively occur only semiautomatically. Therefore, in the following, the terms “semi-automatic” and “automatic” are used interchangeably in the context of the matcher selection mode.
The part of the MOMA Framework that supports the semi-automatic selection mode consists of:

- additional information regarding the ontologies (partially based on MCMA), in particular, the contextual information as to which ontologies have been developed and used (*ontology metadata* in Fig 4.6),
- information regarding existing ontology matching algorithms based on the MCMA (*matcher metadata* in Fig 4.6)
- rules that place the matching algorithms and their characteristics in relation with ontologies; we have formalized the knowledge concerning dependencies between algorithms and the structures on which they operate in terms of dependency rule-statements (*rule repository* in Fig 4.6).
- a selection engine, which, as the core of the MOMA Framework, is responsible for the decision making process by means of rules grouped into a rule repository (*rule-based matcher selection* in Fig 4.6); for a given set of ontologies to be matched, the selection engine must decide which matching algorithms are applicable in order to obtain the desired outputs.

In general, the automatic MOMA Framework resorts to semantically represented metadata on ontologies and matchers in order to express a core set of rules that can be applied to detect algorithms suitable for processing a pre-defined ontological input and extended for particular application purposes.

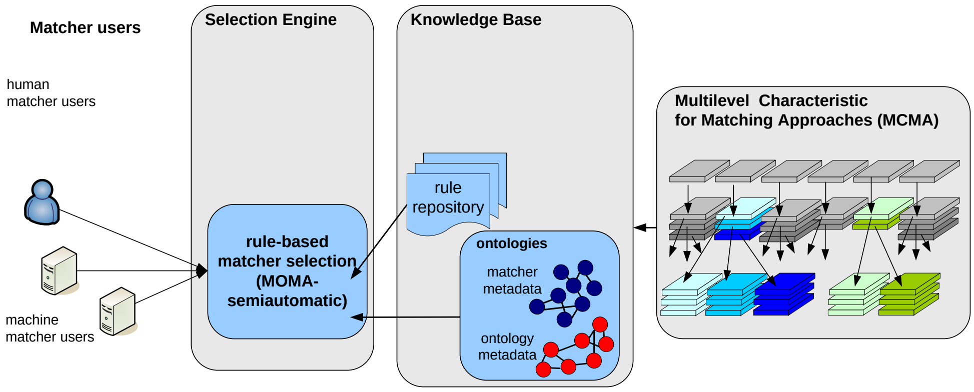


Figure 4.6: MOMA Framework - automatic mode

Due to its generic character (especially regarding the rule-statements which can be extended), the (semi-)automatic approach can be applied in a service-oriented context to enable the discovery and operation of appropriate matching services required to deal with specific ontologies.

4.3 Final remarks on MOMA

The proposed MOMA Framework takes into account the capabilities of existing matching algorithms and suggests appropriate matchers for individual cases. Our MOMA Framework contributes to data integration and interoperability by maintaining awareness of the link between matching algorithms and a wide variety of ontologies. It is the first step toward the reuse of existing ontology matching approaches and contributes to the more optimal utilization of ontology matching tasks as envisioned by the Semantic Web community.

In the next chapters, we concentrate on the development of the MCMA, discuss the manual and automatic MOMA Framework in detail, and elaborate the evaluation of MCMA as well as both MOMA selection modes.

Chapter 5

Matcher Characteristic

Abstract In the following chapter, we describe the development of the matcher characteristic, which is realized in three steps (cf. the blue, green and yellow marks within Fig 5.1): (i) In order to determine the factors relevant for the decision regarding the suitability of particular approaches, the first step of the development process belongs to the analysis of the matcher domain (green part), based on the experiences in ontology engineering, the examination of requirements collected during the development of different Semantic Web-based applications, an analysis of information gathered while collaborating with various experts, and while researching the matching literature. (ii) The factors resulting from the first step are then thematically grouped (blue part) in an iterative process to build, at the end, a multilevel characteristic for matching approaches. (iii) The final development phase is dedicated to the expert-based evaluation (yellow part) and consequential to the refinement of the initially defined *a priori* characteristic, thus creating a revised model.

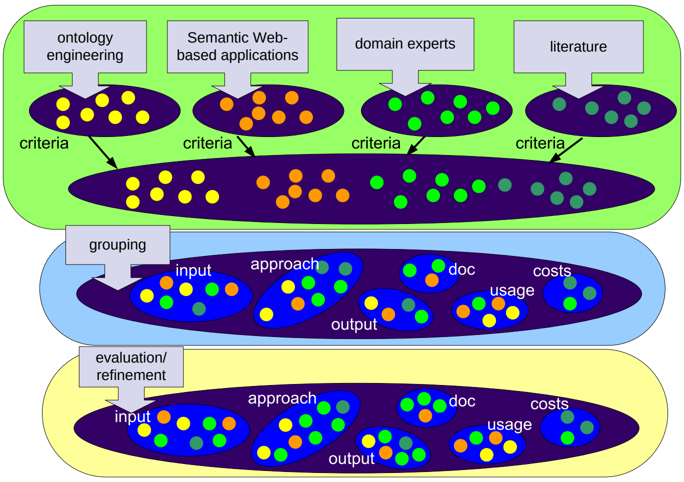


Figure 5.1: Development of the matcher characteristic

5.1 Analysis of the Matcher Characteristic

The indispensable step in the process of finding suitable matching algorithms is to define a matcher characteristic that allows comparison of different approaches and consequently, the selection of appropriate matchers. Thus, our work started with research within the matcher field and analysis of the diversity of the matcher's applications in order to define the attributes that influence the decision regarding the appropriateness of the various approaches. To gather the needed information and because of the fact that the data collection depends on research method applied, the corresponding method needs to be utilized. One of the most common distinctions between the research methods is their classification into *qualitative* and *quantitative* approaches. While quantitative research methods (conclusive research) were originally developed in the natural sciences to study natural phenomena (survey methods, laboratory experiments, etc.), qualitative research methods (exploratory research) were developed in the social sciences to enable researchers to study social and cultural phenomena. Since the qualitative research methods have been designed to help researchers to understand people (i.e. users), their problems, and the circumstances in which they live or work, we have decided to apply this type of research method for the analysis of the matcher domain. Furthermore, in the family of the qualitative research methods, we can choose between study research, ethnography, grounded theory, and action research.

Case study research is an empirical inquiry that investigates a contemporary phenomenon within a real-life context [138]; it is the most common method used in information systems.

Grounded theory is an inductive, theory discovery methodology that allows researchers to develop a theoretical account of the general features of a topic while simultaneously grounding the account in empirical observations or data [138].

Ethnography is an "in-depth" research method, as the researchers spend some time alongside and within the domain analyzed; this method enables a deep insight into the research domain and its problems [138].

Action research aims, as emphasized in [159], to contribute both to the practical concerns of people in an immediate problematic situation and to the goals of social science by joint collaboration within a mutually acceptable ethical framework; this method is asserted in organization development and education but not really in information systems.

Considering the above-mentioned characteristic of each method and the fact that the analysis of the matcher domain is based on:

(a) the examination of requirements collected during the development of different Semantic Web-based applications,

(b) the experiences in ontology engineering,

(c) the researching the matching literature, and

(d) an analysis of information gathered while collaborating with various experts,

we have adopted two approaches: case study research and grounded theory for our purposes (cf. Fig. 5.2). Since grounded theory suggests that there should be continuous interplay between data collection and analysis, and is therefore a specific approach to theory development, we have applied this to the continuous and systematic analysis of the literature and expert opinions ((c) & (d)). Furthermore, due to the importance of the understanding of complex issues within a research domain and since the experiences from real-life applications play a crucial role in our case, we have applied case study research by analyzing the different Semantic Web-based applications involving matching approaches and utilizing it through our empirical findings within the ontology engineering field ((a) & (b)).

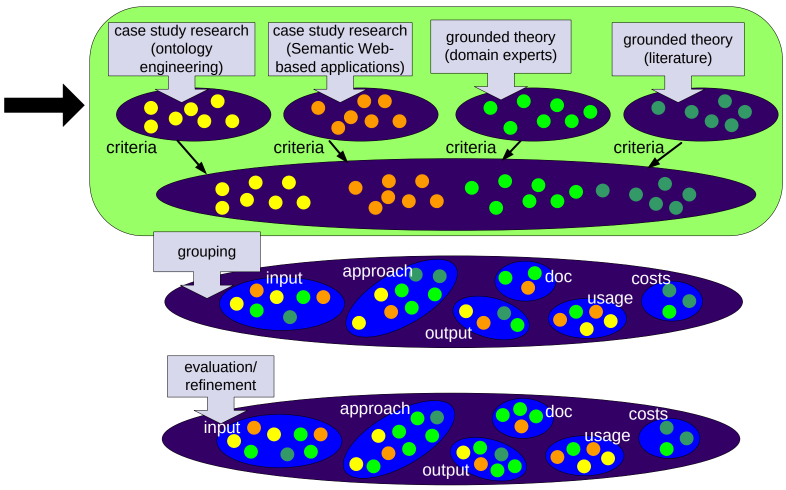


Figure 5.2: Matcher characteristic - analysis

In the following sections, we describe specifically how the matcher characteristic has been developed: by differentiating matching approaches through analyzing the requirements collected during the development of various Semantic Web-based applications ((a); case study research), by taking into account the analysis of the literature together with intensive and systematic collaborations with experts in the ontology matching field, ontology and software engineers ((c) & (d); grounded theory), and by considering our findings within the context of different case studies in ontology engineering ((b); case study research). In order to enable and facilitate the understanding of the process and to emphasize which are the important terms that play a crucial role in the matcher characteristic for selection purposes, we have put in italics the relevant features when describing the different parts of the analysis phase.

5.1.1 Findings within Ontology Engineering

Since ontologies are understood as a means for a shared knowledge conceptualization, reusing existing ontological sources increases application interoperability, on both the syntactic and semantic levels. It is generally accepted in the Semantic Web community that building ontologies from scratch is a challenging, time-consuming, and error-prone task.

> *"Even with emerging web and ontology standards, coordinating ontology development - whether manual or automatic - will prove to be a challenging task. In evolving domains, it is expected that ontologies will not remain static and various versions of ontologies will have to be tracked. Interdisciplinary ontologies may need to be created from existing domain-specific ontologies, domain specific ontologies may need to be merged with more general ontologies, different versions of a single-domain ontology may need to be merged, and new information may need to be merged with existing ontologies. Furthermore, new ontologies may be built by merging information from heterogeneous databases or other information sources."* [54]

Moreover, the development of new ontologies has failed to tap the full potential of the knowledge sources available on the Web. This situation can also be explained by the difficulties related to building reusable ontological sources which must strike a balance between rich conceptualization and application specificity [74]. Furthermore, *reusing existing knowledge* for ontology engineering today is still complicated by serious technical problems that should not be underestimated when deciding on how to build a specific (application) ontology (e.g. from scratch, by reuse, using ontology learning techniques or combinations of the three). The latter problems can be traced back to the difficulties connected to the detection and application of appropriate ontology management algorithms, like matching and merging approaches that are crucial in the process of selection, adaptation, and deployment of relevant sources.

Ontology reuse can be defined as the process in which existing (ontological) knowledge is used as input to generate new ontologies, and it typically starts with the identification of potentially relevant knowledge sources. While most ontologies that have emerged in the last decades can be accessed on the Web, these ontologies usually show significant differences with respect to intrinsic and extrinsic features. In the first category, we mention the *representation language*, the *modelled domain* (represented content), the view on the domain, the *granularity* as well as the *degree of formality* (thesauri, XML-Schemes, UML diagrams, etc.). The second category contains features such as *maturity*, development stage, or underlying methodology. Taking into account, for example, the different representation languages, even when translation tools are available, the source ontologies must be compared and eventually merged. For this purpose, one needs *scheme*-matching algorithms that can deal with the *heterogeneity of the incoming sources*, for instance, w.r.t *their structure*, *granularity*, *degree of formality*, *domain*, and *application* view on the domain.

Although reusing ontologies should reduce engineering costs, as it avoids rebuilding existing ontologies and improves the interoperability between applications, this process is currently associated with huge costs and effort, which may ultimately outweigh its benefits. First of all, as in other engineering disciplines, when we take a look at the costs needed to build an ontology using existing sources, costs involved in finding, becoming familiar with, adapting, and modifying the modules to fit a new context must be considered. Beyond that, building a new target ontology partially means translating between *different representation schemes*, performing scheme matching, or both, and merging sources. The *translation* between representation formalisms is a realistic task only for similar *modelling languages*, and even in this case, current tools face some significant limitations [58, 162, 190].

Reusing an ontology and the efforts associated with this depend significantly on the *complexity* of the conceptual model and the self-descriptiveness or *clarity* of the conceptual model. The *clarity* of the model is mainly influenced by the human-perceived readability. The *complexity* of the ontology, however, must be particularly considered while searching for relevant and applying potentially suitable ontology management algorithms. This complexity depends on three factors: the *size of the ontology*, the *expressivity of the representation language* used, and the *structure of the import graph* containing imported ontologies. The import graph structure can be divided into simple, as in taxonomical *tree structures*, and complex, as in *non-tree structures*. Furthermore, the complexity of the syntax used (representation language) is termed as simple for common taxonomical *hierarchies*, moderate if *further property types* are used, and complex in the case of *restrictions* and *axioms*. The third ontology complexity attribute is related to the *size of the ontology*: *small ontologies* are supposed to contain up to 100 ontological primitives (concepts, relations, instances, and axioms), *middle ontologies* contain up to 1,000 concepts, while ontologies with more than 1,000 concepts are classified as *large ontologies* [170]. Consequently, the costs of reusing existing sources are high in terms of the complex factors mentioned above that, in turn, are directly associated with the application of the ontology management algorithms. This means, for example, that **if the process of alignment and merging of different knowledge sources could be improved in some way, it would have a positive impact on the costs of the entire reuse process**: It would increase the benefits of the reuse strategy and this, in turn, would contribute to balancing the costs involved in the reuse process.

5.1.2 Requirements of Semantic Web-based case studies

Semantic Web technologies are maturing and moving out of academic applications into the industrial and wider "audience". This course is demonstrated (i) by a growing number of people and institutions providing metadata on their personal or institutional web pages in vocabularies based on RDF; (ii) by the strong and growing interest in the semantic topics shown by various commercial sectors; (iii) and by public bodies, like the European Commission, which support the distribution and transfer of these technologies to the business world (cf. Sec. 2.1). One location of such activity was the Knowledge Web EU Network of Excellence, which formed an *Industry Board*[1] to promote greater awareness and faster uptake of Semantic Web technology within European industry. The Knowledge Web, together with this board, aimed to transfer technology from research to industry, promote ontological technologies, propose technological recommendations, and meet industrial application needs. Within this context, some key sectors for the early uptake of Semantic Web technologies were identified: human resources, and health and life sciences [145].

In the following, we elaborate our findings regarding the application of matching approaches in the context of two case studies, in the human resource and health care domains, which were conducted in the course of our two national projects.

Case Study: Human Resource (HR)

In the project *Knowledge Nets*[2], which was a part of the InterVal - Berlin Research Centre for the Internet Economy [3] and was funded by the German Ministry of Re-

[1] `http://knowledgeweb.semanticweb.org/o2i/index-2.html`, accessed on 05.07.2008
[2] Wissensnetze; `http://wissensnetze.ag-nbi.de/` accessed on 07.08.2008
[3] `http://interval.hu-berlin.de` accessed on 10.08.2008

search[4], we picked up on the issue of human resource management as a potential early adopter of semantic technologies; the setting of the project was a typical use scenario for the deployment of Semantic Web technologies considering their application in the e-Recruitment domain [13, 83, 134]. In general, the Knowledge Nets project approached the impact of semantic technologies from a business and technical viewpoint to make predictions about the influence of these new technologies on markets, enterprises, and individuals (cf. Fig.5.3). To this end, the project took a closer look at particular market sectors and application scenarios whereby every scenario included a technological component as well as a deployment component. The former made use of the projected availability of semantic technologies in the coming years while the latter assumed the availability of the information required in machine-readable form. The combination of these two projections allowed us to build e-business scenarios for analysis and experimentations and, on the other hand, to make statements about the implications of the new technology on the participants of the scenario in the early stage of development.

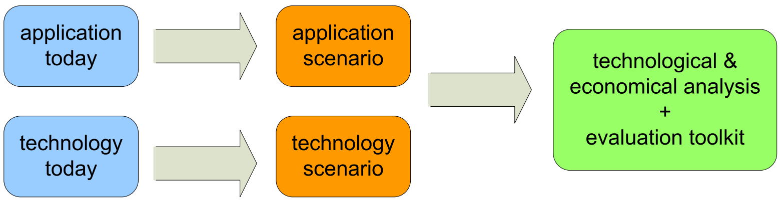


Figure 5.3: Knowledge Net's approach

The first step within the Knowledge Net project towards the realization of the human resources scenario was the creation of a human resources ontology (HR ontology). This ontology was intended for use in a Semantic Web job portal by allowing a uniform representation of job postings and job seeker profiles, and semantic matching in job seeking and procurement tasks. The analysis of requirements revealed the necessity of aligning the resulting ontology with commonly-used domain standards and classifications in order to maximize the integration of job seeker profiles and job postings from different organizations. The usage of the ontology in semantic matching tasks requires it to be represented in a *highly formal representation language*. Human-readable concept names in *various languages* were also necessary in order to make the ontology usable in job portals.

We started the development process by identifying the sub-domains of the application setting (e.g. professional and educational skills, professions or occupations

[4] Bundesministerium für Bildung und Forschung (BMBF), `http://www.bmbf.de/` accessed on 10.08.2008

and industries) and several useful knowledge sources covering them. We noticed that while there are many different existing taxonomies for the classification of job postings, there is no consensus, i.e. common classifications among job portals. The knowledge sources to be reused in this case differed to a large extent regarding *content area*, *granularity*, *representation format* and *degree of formality*. For candidate ontologies, we selected some of the most relevant classifications in the field that have been deployed by national and international agencies and statistic organizations (e.g. occupation classification, industrial sector classification, skills ontology). The selection of the candidate source ontologies was followed by their customization and integration into the target ontology. Due to the application setting, classification standards, like occupation classification and the classification of industrial sectors, had to be completely integrated into the new ontology. Skill ontology was partially used to define concepts representing competencies to describe job requirements as well as job seeker skills. In the end, to develop the target application ontology, we reused the impressive body of domain knowledge available in the form of *classifications*, *taxonomies*, and *ontologies* on the Web.

The complex process of identification of relevant sources or their parts, as well as their integration into the target ontology, was conducted manually. To put it numerically: 15% of the effort necessary to build this ontology was spent on gathering the relevant sources and about 40% of the time was spent on customizing these source ontologies (identifying relevant sub-ontologies). Due to the *heterogeneity of the knowledge sources*, over 45% of the engineering time was used to translate them into the target *representation language* OWL. Refinement and evaluation of the ontology took up the remaining 10% of the total time. Some parts of the total procedure, such as comparing various parts of the reusable sources and searching for overlaps, as well as integrating and customizing, could certainly be automated, thereby shortening the duration of the building work and bringing the cost down. However, since the candidate sources:

- covered different *domains* e.g. occupations and industrial sectors;
- represented different *degrees of formality*;
- differed in the *granularity* of the domain described;
- represented a broad range of *formats* like text files, XML-scheme, DAML+OIL; and
- used different natural languages such as English and German,

no suitable matching or merging algorithms capable of dealing with the heterogeneity of the incoming sources were found in the feasible time. For this reason, most phases within the reuse process, such as selection and merging of relevant sources, even if they could have been performed "automatically" by applying an appropriate algorithm, were conducted manually.

The human resource scenario required matching algorithms, not only in the development of a human resource ontology, but also during the runtime of the application when the HR ontology developed was used to improve the quality of conventional job search engines beyond the multitude of various keyword- and statistics-based algorithms. Given the rich and machine-processable representation of the domain of interest (e-Recruitment), ontology forms the basis for techniques that compute semantic similarities between information resources like job postings and applicant profiles. At this point, a technique that combines annotations using controlled vocabularies with background knowledge regarding a certain application domain was needed. In the prototypical implementation, the domain specific knowledge was represented by concept *hierarchies* for skills and skill levels, an occupational classification, and a *taxonomy* of industrial sectors. With this background knowledge of the recruitment domain (i.e. formal definition of relevant concepts and specification of the relationships between concepts) represented in a machine-understandable format, there was a necessity to compare job descriptions and applicant profiles based on their semantic similarities [153] instead of merely relying on the containment of keywords, as is done by most contemporary search engines. To pinpoint the appropriate jobs for an applicant or a suitable candidate for a job opening, we needed semantic matching approaches capable of dealing with the *highly formal HR ontology* and with the specific application requirements. According to our particular incoming sources (HR ontology) and considering the given application (semantic-based job portal) needs, we identified the following requirements that potential suitable matching approach had to fulfill. The matcher should be able to:

- tackle *two ontologies*, since we were describing both job position posting as well as job seeker;
- deal with the *representation language* of the HR ontology: OWL;
- deal with the *natural language* of the HR ontology, English or German, since they were used separately, depending on in which language the job portal (that uses the ontology) is running;
- tackle *different types of properties*, since the skill ontology contained not only `is-a` properties;
- handle relative *large ontologies* with over 8.000 concepts;
- deal with *ontologies containing instances*;

- *proceed automatically*, since it was supposed to be applied within an online job portal; and
- be able to compare one applicant profile against a multitude of job openings (or the other way around), i.e. it had to perform *1:n matching*.

Case Study: Health and Life Sciences

Our second case study, where matching approaches were required, came from the German Research Foundation[5] funded project *A Semantic Web for Pathology*[6]. The project analyzed the usage of semantic technologies in a retrieval system for image and text data in the medical domain. An ontology, which was to be built, would be used to enable content-based retrieval and guide the automatic semantic annotation of textual pathology reports [18, 19]. The ontology had to cover both domain knowledge (the domain of lung pathology) and application-relevant data, like the structure and content of medical reports, specific to the health-care institution involved in the project. Due to the *complexity of the application domain* and interoperability considerations, the ontology engineering process was to focus on reusing the multitude of medical ontologies instead of building the application knowledge from scratch. Ontology matching techniques therefore played a crucial role in the completion of this task, as we had to merge various ontologies modelling interrelated domains within the final application ontology [136].

The first step towards the development of an appropriate ontology was dedicated to the identification and analysis of over 100 medical ontologies that cover aspects related to our application domain – lung pathology. The result of this phase was a list of potentially relevant knowledge resources which, however, differed to a large extent concerning the *granularity of the conceptualization*, *level of formality*, *implementation language*, etc.:

- SNOMED[7] and DigitalAnatomist[8] describe the anatomy of the lung as well as typical lung diseases and are aligned with UMLS[9];
- *upper-level* UMLS Semantic Network contains generic and core medical concepts as part of the UMLS;
- XML-HL7 is a standard, *XML-based format* for the representation of patient records; and

[5]Deutsche Forschungsgemeinschaft (DFG), `http://www.dfg.de` accessed on 10.08.2008
[6]`http://swpatho.ag-nbi.de` accessed on 05.03.2008
[7]`http://www.snomed.org` accessed on 05.03.2008
[8]`http://sig.biostr.washington.edu/projects/da/` accessed on 05.03.2008
[9]`http://www.nlm.nih.gov/research/umls` accessed on 05.03.2008

- Immunohistology Guidelines are used by domain experts in diagnostic procedures within the medical organization involved in the project [136].

The conceptualization, in which we tailored the sources to the particular needs of our restricted application domain phase, was followed by the translation of the UMLS data to OWL. Simultaneously with the reuse activities, large parts of pathological knowledge were conceptualized using a text-based ontology learning approach, as this knowledge was not covered by any of the identified sources. In order to integrate these alternative development outcomes, we were confronted with the problem of choosing a matching approach capable of dealing with the *size* and the *complexity* of the resulting ontologies. To face these needs, we needed an algorithm that:

- would be able to deal with *upper-level ontologies* like UMLS Semantic Network;
- could handle the different *levels of formality*;
- could deal with different *representation languages* (XML schemes and OWL/RDFS);
- would not need to handle *instance data* since we did not have any instances;
- could address different *natural languages*: English and German.

Furthermore, considering the particularity of the application domain, like medicine-specific core relations and the size of the ontologies, it would be a definitive advantage if the approach had been already tested and proven within the *context* of ontologies covering a medical domain.

HR vs. Health/Life Sciences

Our findings in the context of the two cases described above can be concluded with the observation that, even if both e-Recruitment and medical cases needed matchers tackling relatively large incoming sources, and in both cases matchers would have to deal with different natural languages, approaches suitable for the medical case would not be appropriate for the e-Recruitment case since, for instance:

- in the medical case, we needed schema-based matchers (no instances were available), while for the e-Recruitment scenario, approaches handling instances were required; the pure schema-based matchers would not be able to match the instances of the e-Recruitment incoming sources;

- in the e-Recruitment case, we had domain ontologies as incoming sources, while in the medical case, matchers handling upper-level ontologies were called for; the domain ontology matchers would not be able to process upper-level sources (they would probably deliver no, wrong, or insufficient results);
- in the e-Recruitment case, we had only OWL sources, and approaches dealing with this representation language would fit this case, while in the medical case, we also had some XML sources that forced us to apply matchers dealing with this representation language.

The very brief theoretical comparison of the e-Recruitment and medical requirements corresponds to our practical findings within the cases, and shows that the same matchers are not compatible for different cases and that the matcher selection matters regarding the given applications and their requirements.

5.1.3 Analysis of Expert Opinions

In the course of collaborating with experts in the matching field and developers of particular matchers, we benefited from their experiences with the existing approaches and collected vital matcher data. Analyzing the data gathered enabled us to exploit the valuable ideas embedded in current matching approaches and understand their limitations, and at the same time, we gained insight into the important, from the expert point of view, attributes that need to be considered when describing approaches and searching for suitable ones. According to almost all the interviewees, one of the important limitations of the current algorithms is related to the problem that all algorithms can deal quite well with *two* incoming sources but only some of them serve *more than two ontologies*. Moreover, considering the *number of ontological primitives*, the developers expressed their hope that in the future, taking into account the industrial requirements regarding processing large data sets, there would be more approaches capable of dealing not only with *small sources* but also of matching *larger ontologies with thousands of primitives*. While analyzing the *number of different types of ontological primitives*, the experts found it striking that matching ontologies that contain *axioms* is still a challenge, since very few approaches were capable of handling this type of primitives. Beyond this, if we analyze the heterogeneity of the input sources regarding the *natural* and *representation languages*, the best results in regards to the variety of suitable matching approaches can be achieved, according to domain experts, by sources that use only *one natural* or *one representation language*. If we examine matchers considering sources with and without instances, all approaches are capable of matching *schemes*, but only a small number of the existing systems can handle *instances*. This opinion is also shared by the researchers involved in the Knowledge Web EU Network of Excellence, who stress that:

> *"...schema-based matching solutions have been so far investigated more than the instance-based solutions. We believe that this is an objective trend, since we have striven to cover state of the art systems without bias towards any particular kind of solutions."* [55]

Since the "matcher's world" is wide-ranging and heterogenous, the matchers differ not only in the sources they can handle successfully but also in the different processing types: some systems follow the *black box paradigm* (i.e. automatic execution without human intervention) while others allow or require *user interaction*, e.g. *manual pre-* or *postprocessing* effort. Another aspect discussed with developers and matching field experts was the adaptation of approaches for different application purposes (*integration*, *translation*, *navigation*, *query answering*, *data mediation*, etc.), various application goals (like *local use* or *internet use*) as well as adaptation to *different tasks*, *applications*, and *domains*. If we consider the application goals, the majority of matchers have been developed to be run on local machines while few systems can be accessed on a network or as an online service. According to the opinion of experts, such limitations slow down the realization of the vision of a fully developed Semantic Web.

In the next section, we outline the results of the analysis of the matcher literature, which, in the main, confirm the opinions, issues and characteristics gleaned from the matching experts and presented above.

5.1.4 Literature-based Analysis

In terms of developing a matcher characteristic, we have also analyzed the existing classifications (cf. Sec. 3.2), taking into account in particular the potential suitability of the criteria defined for the purpose of finding matchers that meet the given application requirements. Considering these classifications, the matcher algorithms can be divided along two main groups: *individual algorithms* (e.g. FCA-MERGE [176] or S-Match [69]) which apply only a *single method of matching items*, e.g. linguistic or taxonomical matchers, and combinations of former matchers which attempt to overcome their limitations by proposing *hybrid* and *composite* solutions. A hybrid approach (e.g.Cupid [124]) follows a *black box paradigm*, in which various individual matchers merge to form a new algorithm, while composite matchers allow increased user interaction (e.g. GLUE [47], COMA [45], CMC [186]). Despite the relatively large number of promising matching approaches, their limitations with regard to certain ontology characteristics have often been emphasized in the recent literature [69, 124, 130, 168]. Furthermore, there are a number of constant features that are shared by the majority of systems, but each individual system usually innovates within a particular aspect [55]. The main issues emphasized in the common "matching" literature can be summarized as follows[10]:

- Some approaches assume a common or, at least to large extent, overlapping universe of discourse [33].
- The approaches cannot usually be applied across various domains with the same effect (e.g. Cupid [69]) since they focus on specific application domains, such as books or music.
- They require certain representation (or translation to the suitable format) languages (e.g. COMA [45]); specifically this means that most systems deal only with certain ontology types, such as DTDs, relational schemas, or OWL ontologies, and only a small number of approaches are able to be more universal in handling multiple types of ontologies or deal with different issues of various application domains.
- They require specific natural languages.
- Most of the known matching approaches are capable of dealing with two incoming sources, while few (e.g. DCM [30] or Wise-Integrator [82]) can handle multiple sources.
- They perform well on relatively small inputs with at most hundreds of concepts, and have not been tested or do not scale for real world applications processing complex schemes.
- Some approaches do not perform well on inputs with heterogeneous (graph) structures or are restricted to tree-based concept models (SimilarityFlooding (SF) [130], S-Match [69]).
- The matcher results of most approaches are based on a one-to-one mapping between taxonomies (such as in GLUE [46]), while only a relatively small number of systems offer more complex correspondences like 1:n or m:n matching [55].
- They need some manual pre-processing (such as in GLUE, COMA [44]).

As stated in [27], the main general requirement regarding matching approaches is their applicability to different *ontology specification languages*, with special attention to recent standards of the Semantic Web like OWL. A further general requirement is

[10] Most following matching algorithms have been already introduced in Sec. 3.4

the capability of coping with different *levels of detail* and design choices in describing the knowledge of interest using a certain *language*. In addition, the capability of considering different constructs used in ontology languages is required for matching purposes.

The analysis of the matcher domain, based on findings in ontology engineering, requirements collected during the development of various Semantic Web-based applications, collaborations with matching experts and ontologists, and the analysis of matching literature, has resulted in over 100 features relevant for our matcher characteristic. This set of terms serves as a basis for the development of a matcher characteristic, which is to be applied in context of matcher selection process.

5.2 Development of a multilevel characteristic

The characteristics collected during our research can, on the one hand, be used to describe matching approaches and, on the other, serve as a basis to develop an advanced characteristic to be applied for comparison of different matchers. Due to the fact that, as we noticed, some terms in the identified set of features "belonged together", we decided to order them by building *groups of attributes* (cf. Fig. 5.4).

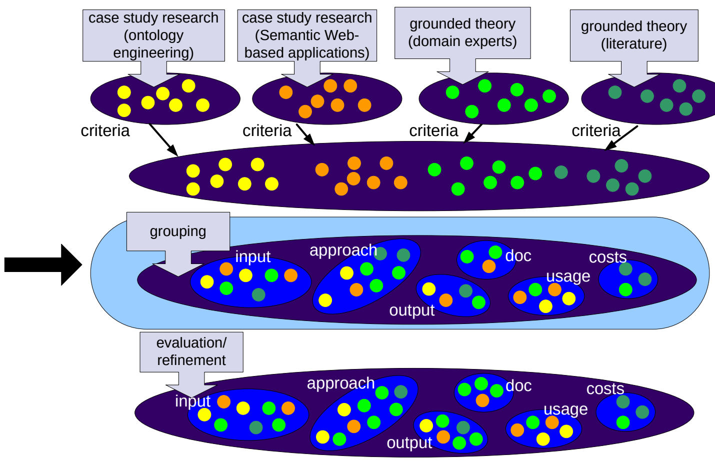


Figure 5.4: Grouping of the matcher characteristics

Each group was assigned a name, a more general term that described the attributes of the group. This generalization process linked with the naming of each new group was conducted several times in the course of an iteration process. We started the process with the single attributes, then we considered the groups that had been created from attributes while taking into account groups that were created in the earlier phase. The process concluded when no two groups were left that, in our opinion, could be brought together. The "grouping process" resulted in six main groups of characteristics called **dimensions**. These dimensions are the main aspects to be taken into account in the search for the suitable matching approaches in the context of a given problem:

- **Input characteristic** takes into account the ontologies to be matched.
- **Approach characteristic** describes the matching algorithms.
- **Output characteristic** defines the desired result of the matching execution.
- **Usage characteristic** considers the different situations in which the approaches can be or have already been used.
- **Documentation characteristic** points out the existence and quality of the documentation
- **Cost characteristic** addresses the costs for the usage of an algorithm.

Since we put the attributes into groups under rather general designations, we have built a hierarchical tree of characteristics that we call a **Multilevel Characteristic for Matching Approaches**, in short **MCMA**, where the child nodes describe and represent the parent node properties (cf. Fig. 5.5). In other words, the dimensions, which form the collection for matcher attributes and build the top layer of the multilevel characteristic, are defined by sets of factors which, in turn, are described by the attributes.

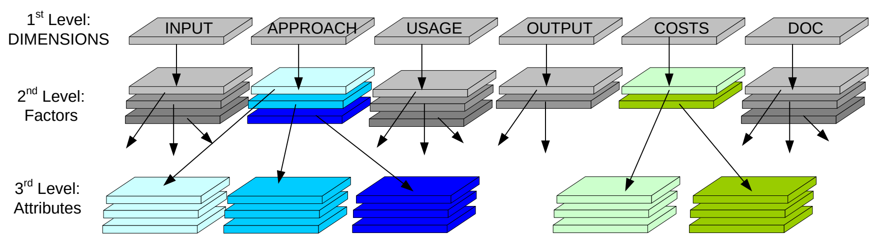


Figure 5.5: Multilevel characteristic for matching approaches (MCMA) with its dimensions, factors and attributes

In the following sections we discuss the factors and attributes of each dimension which belong to the evaluated *a posteriori* MCMA. The evaluation process and the development of the *a posteriori* model from the preliminary *a priori* characteristic is described in Section 8.3. For a clearer introduction of the entire set of factors and attributes, we have placed them together with the corresponding descriptions in the form of tables; this also allow us, to a certain extent, to maintain the hierarchical structure of the developed MCMA (cf. Table 5.1 - 5.6).

5.2.1 Input characteristic

The first step towards the analysis of the matching characteristics is the examination of the matching input. Since the majority of existing approaches for ontology matching are restricted to processing particular classes of ontologies and thus are unable to guarantee a predictable quality of results on arbitrary inputs, the attributes that describe the input are, in our opinion, the most important and relevant criteria that play a crucial role in the selection of appropriate algorithms. Despite the relatively large number of promising matching approaches, their limitations with respect to certain ontology characteristics have often been emphasized in the recent literature [69, 124, 130, 168, 169]. Accounting for the fact that matching algorithms cannot be applied with the same expectations of success, regardless of any dimension of the ontology metadata model, we have identified some syntactic, as well as semantic, ontology features as relevant for the process of reviewing matcher suitability. **Quantitative syntactic features**, such as the size of the ontology which can be expressed in a number of (specific) ontological primitives, affect the matching execution performance, because some matchers perform well only on relatively small inputs, and because the quality of the structured-based matchers typically perform better on simple (graph) structures. In contrast, **semantic features**, such as model type, representation and natural language (since some algorithms require a certain language), input structure (while other matchers do not perform well on heterogeneous structures or can accept only certain types of schemes), and the level of formality, restrict the number of applicable matching algorithms. Furthermore, due to the fact that only a few existing matching approaches are capable of dealing with multiple sources, the number of incoming sources (two or more) that are to be matched play a crucial role in matcher characteristic. Additionally, we have also taken into account that most matchers rely not only on the input to be matched (like schemas or instances), but also on auxiliary information, like matching rules, domain constraints, dictionary, or previous matching decisions. Considering all these facts and terms, we set up the dimension input characteristic as shown in Tab. 5.1.

DIMENSION: INPUT Characteristics	
Attribute	**Description**
Factor: Input Size (algorithm capable of handling:)	
number of ontologies	number of different ontologies to be matched (two or more)
size of input	number of ontological primitives (concepts, properties, axioms) to be matched: *small* (up to 100 primitives), *middle* (101 - 500 primitives), *large* (501 - 1,000 primitives), *extra large* (over 1,000 primitives)
size of instances	number of instances to be matched: *no instances*, *small* (up to 500 instances), *medium* (501 - 1,000 instances), *large* (1,001 - 10,000 instances), *extra large* (over 10,000 instances)
number of concepts	number of concepts to be matched: *small* (up to 100 concepts), *medium* (100 - 500 concepts), *large* (500 - 1,000 concepts), *extra large* (over 1.000 concepts)
number of relations	number of relations to be matched: *small* (up to 30 relations), *medium* (31 - 100 relations), *large* (100 - 1,000 relations), *extra large* (over 1,000 relations)
number of axioms	number of axioms to be matched: *no axioms*, *small* (up to 30 axioms), *medium* (31 - 100 axioms), *large* (100 - 1,000 axioms), *large* (over 1,000 axioms)
Factor: Input category (algorithm capable of handling:)	
glossary	a list of terms with their definitions
thesaurus	a list of important terms (single-word or multi-word) in a given domain and a set of related terms for each term in the list
taxonomy	indicates only class/subclass relationship (hierarchy) [48]
DBschema	often does not provide explicit semantics for their data
ontology	an explicit specification of a conceptual [74]; describes a domain completely [48]
Factor: Input formality level [192] (algorithm capable of handling:)	
(highly/semi) informal ontology	expressed loosely in natural language or in a restricted and structured form of natural language
semi-formal ontology	expressed in an artificial, formally defined language
(rigorously) formal ontology	meticulously defined terms with formal semantics, theorems and proofs of such properties as soundness and completeness
Factor: Input model type [75] (algorithm capable of handling:)	
task ontology	model built for a generic task (e.g. diagnosing)
domain ontology	model of a generic domain (e.g. medicine) or part of the world
application ontology	model built for a specific application; concepts depend on both a particular domain and task, which are often specializations of both the related ontologies
upper-level ontology	model of the common objects that are generally applicable across a wide range of domain ontologies; it describes very general concepts (e.g. space, time)
Factor:Input type (algorithm capable of handling:)	
scheme	schema-based matcher
instance	instance/content-based matchers
Factor: External sources (algorithm is able to handle /to provide:)	
additional user input	most matchers rely not only on the input to be matched (like schemas or instances) but also on auxiliary information
previous matching decision	
training matches	
domain specific resources/ constrains	
domain constrains	
list of valid domain values	
dictionary	
missmatch information	
matching rules	
global schemas	
Factor: Input natural language (NL) (algorithm is:)	
NL-specific (one language)	the approach is dependent on one natural language
NL-specific (many languages)	the approach is dependent on more than one natural language
NL-independent	the approach is language independent
to be continued …	

Attribute	Description
Factor: Input representation language (RL) [192] (algorithm is:)	
RL-specific (one language)	the approach is dependent on one rep. language
RL-specific (many languages)	the approach is dependent on more then one rep. language
RL-independent	the approach is independent on rep. language
Factor: Art of the input representation language (algorithm serves:)	
XML/XMLSchema	the approach serves XML/XMLSchema
relational schema	the approach serves relational schema
RDF(S)	the approach serves RDF(S)
OWL	the approach serves OWL
Factor: Input structure (algorithm capable of handling:)	
tree structure	the approach can handle only tree-structures
DAGs structure	the approach can handle directed acyclic graphs structures
graph structure	the approach can handle (heterogenous) graph structures
is-a relations	the approach can handle is-a relations
heterogeneous relations	the approach can perform additionally to the "is-a" also on other relations

Table 5.1: Input characteristic

5.2.2 Approach characteristic

The second crucial dimension characterizes the matching approaches themselves. The corresponding factors and attributes compile a list of matcher features that have been empirically proven to have an impact on the quality of matching tasks. They consider the common categorization of the approaches, like the classifications presented, for example, in [44, 157, 168] and are briefly described in Section 3.2. Taking into account the classification introduced in [157], we have made distinction between *individual algorithms*, which compute a mapping based on a single matching criteria [69, 176], and combinations of the individual algorithms: *hybrid* and *composite solutions*. A hybrid approach [124] follows a *black box paradigm*, in which various individual matchers are synthesized into a new algorithm, while the composite matchers allow for increased user interactions [45, 47]. The *approach characteristics* also take into account such factors as processing type, matching ground and execution parameter (cf. Table 5.2).

DIMENSION: APPROACH Characteristics	
Attribute	**Description**
Factor: Matcher Type (algorithm is a(n):	
individual matcher	computes a mapping based on a single matching criteria
combined matcher	uses multiple individual matchers
Factor: Granularity/input interpretation of the matcher [56] (algorithm is a:)	
element-level matcher	matcher can be applied on (individual) elements of the input in isolation ignoring their relations with other elements
syntactic element-level matcher	matcher can be applied on (individual) elements of the input using purely syntactic methods
external element-level matcher	matcher can be applied on (individual) elements exploiting auxiliary (external) resources in order to interpret the input
structure-based matcher	matcher looks at the structure of input, be they relationship types, data types, or schema structures; it considers correspondences by
to be continued …	

Attribute	**Description**
	analyzing how elements appear in the structure
syntactic structure-based matcher	matcher interprets the input with regard to it sole structure following some clearly stated algorithms
semantic structure-based matcher	matcher looks at the structure of input using some formal semantics to interpret its input
external structure-based matcher	matcher looks at the structure of information exploiting auxiliary (external) resources in order to interpret the input
Factor: Processing (algorithm supports:)	
manual execution	manual execution
gray box paradigm	semi-automatic execution where the human intervention is possible
black box paradigm	automatic execution without human intervention
manual preprocessing allowed/required	human intervention before execution is allowed or even required
manual postprocessing allowed/required	human intervention after execution is allowed or even required
Factor: Execution Type (algorithm supports:)	
simultaneous execution	the single matching algorithms (within a composite matcher) can be executed simultaneously
sequential execution	the single matching algorithms (within a composite matcher) can be executed sequentially
Factor: Kind of Similarity Relation (algorithm performs:)	
syntactic matching	similarity based on syntax driven techniques and syntactic similarity measures; relation computed between labels at nodes [168]
semantic matching	relation computed between concepts at nodes [168]
Factor: Matcher Level (algorithm can perform on:)	
element level	match performed for individual schema elements
structure level	match performed for complex schema structures
atomic level	elements at the finest level of granularity e.g. attributes in an XML schema [157]
non-atomic (higher) level	e.g. XML elements
Factor: Matching Ground	
heuristic	"guessing" relations between similar labels or graph structures [167]
formal	uses formal techniques (e.g. can have model-theoretic semantics which is used to justify the results) [167]
Factor: Semantic Codification Type(algorithm uses:)	
implicit techniques	syntax driven techniques [167](e.g. considers labels as strings)
explicit techniques	exploit the semantics of labels [167]; uses an external sources for assessing the meaning of labels
Factor: Execution Parameter (algorithm needs:)	
max time of execution	describes the maximum time needed for execution
max disc space for execution	describes the maximum disc space needed
precision	expresses the proportion of relevant retrieved matches [196]
recall	expresses the proportion of relevant documents retrieved [196]

Table 5.2: Approach characteristic

5.2.3 Output characteristic

In addition to the input and approach dimensions, the *output characteristic* (cf. Table 5.3) plays a decisive role in the selection of the suitable matching algorithm. Depending on the requirements given, an application can, for example, call for a matcher that considers only some of elements of the schemes, while other systems might lack a match for all elements. One of the key factors in this dimension is the *cardinality* (global vs. local cardinality) which specifies whether a matcher compares one or more elements of one scheme with one or more elements of another scheme (in some cases the results are based on a one-to-one mapping between taxonomies [46], while in others it may be one-to-many).

DIMENSION: OUTPUT Characteristics	
Attribute	**Description**
Factor: Output type	
deliver relations	the output is not restricted to correspondence of equivalence
deliver value	e.g. matcher used to determine the semantic similarity between concepts
deliver understandable (for humans) results	matcher delivers some explanations of the results
Factor: Matching Cardinality	
global 1:1	relationship cardinalities between entities w.r.t different entities [157] e.g. one-to-one or many-to-many alignments.
global n:1	
global 1:m	
global n:m	
local 1:1	relationship cardinalities between entities w.r.t an individual correspondence [157] e.g., simple or complex correspondences
local n:1	
local 1:m	
local n:m	
Factor: Execution Completeness	
full match	considers all elements of the schemes
partial match	considers only some elements of the schemes
injective match	distinct elements of the domain is mapped to distinct elements of the range
surjective match	all elements of the range are mapped to elements of the domain

Table 5.3: Output characteristic

5.2.4 Usage characteristic

Among the fundamental requirements for the realization of the vision of the fully developed Semantic Web are proven ontology matching algorithms. Though containing valuable ideas and techniques, some of the current matching approaches lack exhaustive testing in real world scenarios. Considering this problem and additionally making allowances for the fact that some of the algorithms cannot be applied across various domains to the same effect [69], it is important to know if a particular approach has already been successfully adapted for different *domains*, *applications* and *tasks*. In addition, the *usage characteristics* dimension also considers different types of matcher usage: matcher utilized by ontology engineers (*applicable by human*) who, for example, look for means to compare sources for building a new ontology or matcher applied by Web Services (*applicable by machine*) seeking automatized methods to generate mediation ontologies (cf. Table 5.4).

DIMENSION: USAGE Characteristics	
Attribute	**Description**
Factor: Usage goal (algorithm is built for:)	
local use	the matcher is run on the local machine
network use	the matcher is accessible on a network
internet-based use	the matcher service is available through internet
Factor: Application Area (algorithm is built for:)	
reuse of sources	the matching approach is deployed for ontology reuse, which may be defined as a process in which available knowledge is used to generate new ontologies
usage of sources	the matching approach is applied for use ontologies (within an application) e.g. to compare profiles
integration	reuse of available source ontologies within a range in order to build a new ontology which serves at a higher level in the application than that of various ontologies in ontology libraries [118]
transformation	associated with the ontology evolution that uses matching for finding the changes that have occurred between two ontology versions
merging	the matching approach is used for integrating the schemas of different databases under a single view
translation	ontology translation is required to translate data sets, generate ontology extensions, and query through different ontologies [50] for example (i) catalog integration - matchers are used to offer an integrated access to online catalogs, (ii) multi agent communication - matchers are used to find the relations between the ontologies used by two agents and translating the messages they exchange, (iii) context matching in ambient computing uses matching approaches of application needs and context information when application and devices have been developed independently and use different ontologies [55]
query answering	can be applied to data integration or P2P information sharing where matching approaches are used to find the relations of ontologies used by different peers
data mediation	applied within web service composition in which matchers are used between ontologies to describe service interfaces in order to compose web services by connecting their interfaces
query reformulation	uses matcher to translate user queries about the web
navigation	uses matchers to annotate web pages with partially overlapping ontologies
Factor: Usage type (algorithm is:)	
human applicable	approach can be used only by humans (human interaction indispensable)
machine applicable	approach can be used by machine as a service
Factor: Adaptation ability (algorithm has been applied for:)	
number of domains	number of different domains the matching approach was applied to
number of applications	number of different applications the matching approach was applied to
number of tasks	number of different tasks the matching approach was applied to
reference of usage	the approach as has been utilized by other users

Table 5.4: Usage characteristic

5.2.5 Documentation characteristic

Because documentation is an essential part of every software product and, for engineering purposes, often more important than the program code [97], the information regarding its quality and clarity can be significant in the selection of an approach. Furthermore, since one of the goals of documentation is to provide sufficient information so that an architecture can be analyzed for suitability to the purpose [32], it can be a determining factor for the selection of a particular algorithm, especially if the algorithm is to be reused in a different context from the domain or application it was originally developed for (cf. Table 5.5).

DIMENSION: DOCUMENTATION Characteristics	
Attribute	**Description**
quality of documentation	quality of the available documentation
clarity of documentation	clarity of the available documentation
clarity of maturity description	clarity of the description of the approach's maturity
availability of examples	are examples of the approach available

Table 5.5: Documentation characteristic.

5.2.6 Cost characteristic

The last dimension, *cost characteristic*, describes the financial factors regarding the (commercial[11]) usage of a single matching approach, like the matcher licence or the access to the appropriate matcher interface (cf. Table 5.6).

DIMENSION: COST Characteristics	
Attribute	**Description**
costs of matcher licence	the costs entailed to acquire the matcher licence
costs of matcher tool licence	the costs entailed to acquire the tools matcher has been developed with
costs of access matcher interface	the costs entailed to acquire the use of the interface

Table 5.6: Cost characteristics.

5.3 Matcher profile

In order to collect data regarding existing matchers, we have developed an online questionnaire *"Characteristic of ontology matching approaches"*[12] based on the MCMA defined in the latter section (cf. Sec. 5.2). The survey allows us to both analyze the existing matchers, and to provide data important for the process of selection of suitable matching approaches. Furthermore, the analysis of the data gathered enables exploitation of the valuable ideas embedded in current matching approaches, and at the same time, accounts for their limitations. The survey, consisting of 37 questions (some questions are mandatory), is divided into 8 groups:

- *introductory questions*, which are related to the developer team and its institution;
- *matcher input size-related questions* aim to collect information regarding the size of the input, which is served by the particular matching approach (e.g. How

[11] At the moment, there is no available commercial automatic matching tool. Nevertheless, in order to deal with such problems in the future, we have taken these criteria into account.

[12] `http://matching.ag-nbi.de` accessed on 05.03.2008

well does your approach support matching of sources with small numbers of axioms?);

- *matcher input type-related questions* gather information regarding the inputs handled by the matcher (e.g. How well does your approach match formal ontologies?);
- *matcher approach-related questions* collect data concerning the matching approach itself (e.g. How well does your approach support black box paradigm?);
- *matcher usability-related questions* provide information regarding the application and usage of the matching approach given (e.g. How well does your approach support sources integration?);
- *matcher output-related questions* gather data concerning the output delivered by the matching approach (How well does your approach support the global / local cardinality?);
- *matcher documentation-related questions* gather data regarding the quality of the available matcher documentation (e.g. How high is the quality of the matcher documentation?); and
- *matcher cost-related questions* collect information on the costs of using matching approach (e.g. How high are the costs for the matcher licence?).

The questionnaire respondents weighed the algorithms according to a scale between 0, the approach does not support the required feature, and 8, the matcher deals extremely well with the given characteristic[13](cf. Table 5.7).

Rating	Meaning
0	no answer / the approach does not support the feature
1	between no support and slight support
2	slight support
3	between slight and good support
4	good support
5	between good and very good support
6	very good support
7	between very good and extremely good support
8	extremely good support

Table 5.7: Rating scale.

In the first phase of the analysis of the existing matching approaches, we decided to involve and examine the matcher that took part in the *Ontology Alignment*

[13]This scale applies to all questions except for the first group since they concern general information, which are more descriptive in nature

Evaluation Initiative 2006 Campaign (OAEI 2006) organized by the Ontology Alignment Evaluation Initiative (OAEI)[14]. OAEI is a coordinated international initiative whose aim is to establish a consensus for the evaluation of available methods for semantic matching and ontology integration. A series of evaluation campaigns has been conducted since 2004[15] to assess the strengths and weaknesses of existing alignment/matching systems, to compare the performance of different techniques, improve evaluation techniques, and to help improve the ontology alignment/matching approaches (cf. Fig. 3.1). During the campaigns, the participants sampled their matchers on the test cases provided by the contest organizers. At the end, the resulting rankings of the matchers were based on their suitability for a particular case.

In the following, we will briefly analyze some data provided by the developers whose matchers took part in the OAEI 2006 Campaign:

- **AOAS (NIH, NLM)** is an approach to align concepts; it is automatic, rule-based, and operates at the schema level, generating mostly point-to-point mappings; it uses a combination of domain-specific lexical techniques and structural and semantic techniques [205].
- **Automs** is a tool for the automatic alignment of ontologies that integrates several matching methods [109].
- **Falcon-AO** is an automatic ontology alignment tool, which strives to provide technologies for **f**inding, **a**ligning and **l**earning ontologies, and ultimately for **c**apturing knowledge through an **on**tology-driven approach [95].
- **ISLab HMatch** is an algorithm for the dynamic matching of distributed ontologies, based on linguistic and structural matching techniques for the evaluation of affinity in terms of concept names and concept contexts [25].
- **MAOM-QA (AQUA)** is an ontology mapping system used with a multi agent ontology mapping framework in the context of question answering [139]
- **PRIOR** is a **pr**opagation and **i**nf**o**rmation **r**etrieval-based ontology mapping system that exploits both linguistic and structural information to map small ontologies, and integrates the Indri search engine[16] to process large ontologies [125].

[14] `http://oaei.ontologymatching.org` accessed on 10.03.2008

[15] OAEI 2004 Campaign: `http://oaei.ontologymatching.org/2004/Contest/`;OAEI 2005 Campaign: `http://oaei.ontologymatching.org/2005/`;OAEI 2006 Campaign: `http://oaei.ontologymatching.org/2006/`, all accessed on 10.03.2008

[16] `http://www.lemurproject.org/indri/`, accessed on 10.10.2008

- **RiMOM** is a tool for ontology alignment by combining different strategies; it integrates multiple strategies: edit-distance based, statistical-learning based, and three similarity-propagation based strategies [203]
- **OWL CtxMatch** is an algorithm developed to match OWL DL ontologies [142].

For certain characteristics, we provide a diagram to show the weighing of a particular matcher concerning the property given and within the description of the results we have used in the scale mentioned above.

5.3.1 Size-related questions

In the first set of questions, the developers rated their approaches regarding the size of the input their matcher can handle. According to our analysis of the answers provided, all the algorithms can deal *well* to *extremely well* with two incoming sources, but only MAOM and ISLab HMatch are capable of serving more than two ontologies *well* and *very well*, respectively. Taking into account the number of ontological primitives within the sources, all the approaches can handle *very well* to *extremely well* small ontologies (up to 100 primitives), and five of them (Falcon-AO, PRIOR, RiMOM, AOAS and ISLab HMatch) can still deal with medium (up to 500) size ontologies. The situation looks a bit different if we consider the large (up to 1,000) and extremely large (over 1,000) sources. Even these ontologies can be handled by the five approaches mentioned, but only PRIOR can tackle the problem extremely well while the others are only at an average (cf. Figure 5.6).

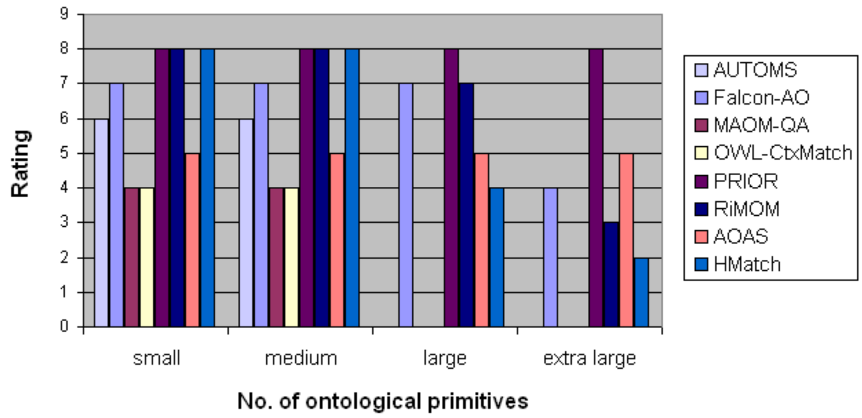


Figure 5.6: How well does the approach serve small/medium/large/extra large ontologies?

While analyzing the number of different types of ontological primitives, we were struck by the fact that the matching of ontologies that contain axioms remains a challenging issue, since so far only four (Falcon, OWL-CtxMatch, RiMOM, ISLab HMatch) algorithms were able to handle this type of primitives (cf. Figure 5.7).

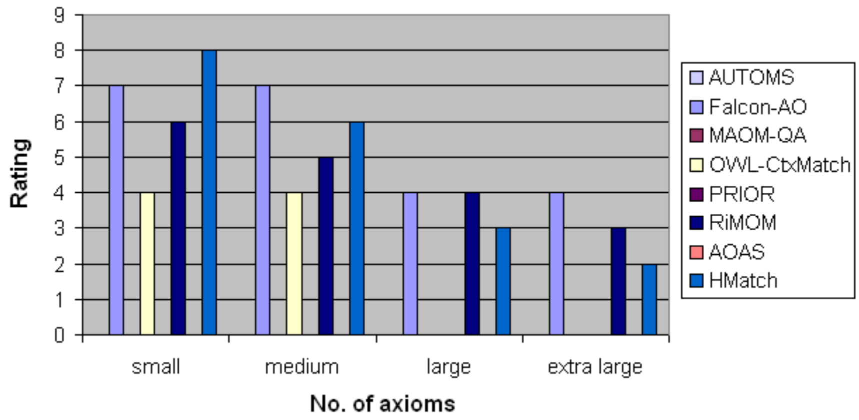


Figure 5.7: How well does the approach match ontologies with different numbers of axioms?

5.3.2 Input type-related questions

One of the important properties of matching approaches is the type of input on which they can operate successfully. To review the "matchability" of the existing approaches regarding the heterogeneity of the incoming sources, in the subsequent set of questions we have concentrated on attributes, which characterize the matcher input, like input category, formality level (highly/semi-informal ontology, semi-formal ontology, rigorously formal ontology, cf. Sec. 5.2.1), input type (instance, schema), and natural and representation languages. First of all, let us analyze the existing matchers according to sources with and without instances: while almost all the approaches can match schemes, only Automs, Prior, and RiMOM can handle instances. Furthermore, if we take a look at the different input categories, we notice that all the algorithms can then deal with taxonomies and ontologies (quite well), but only four of them (Falcon, RiMOM, AOAS, ISLab HMatch) are able to match thesauruses. In addition, Falcon and RiMOM can handle glossaries (cf. Figure 5.8). Beyond this, if we analyze the heterogeneity of the input sources regarding the natural and representation languages, the best results concerning the variety of suitable

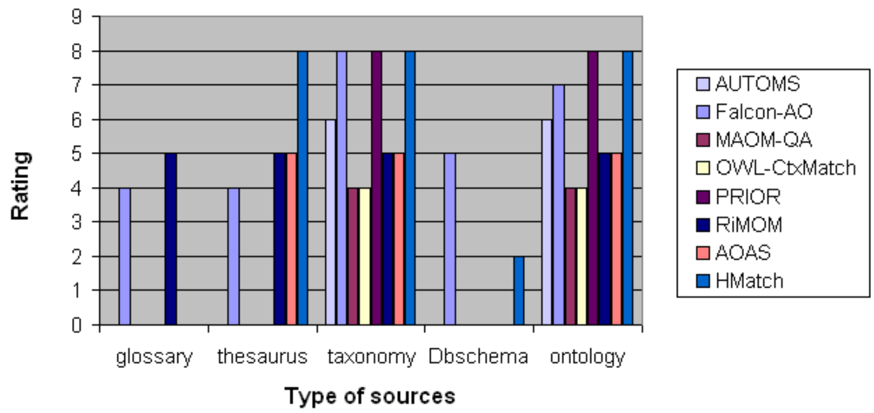


Figure 5.8: How well does the approach handle different types of sources?

matching approaches can be achieved by sources that use only one natural or representation language (Figure 5.9). Only RiMOM is a (natural) language independent approach, meaning that it can also serve multilingual ontologies.

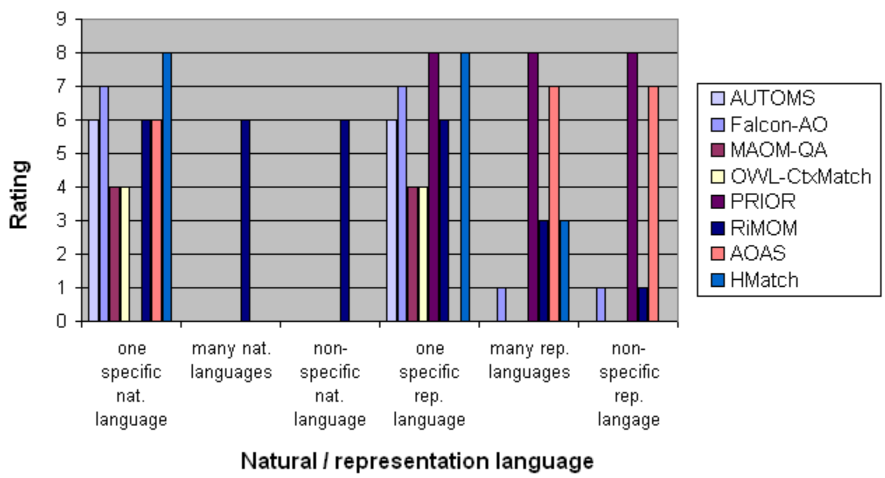


Figure 5.9: How well does the approach handle sources that take into account the rep. and nat. languages?

5.3.3 Approach-related questions

During the ability and suitability examination of different approaches, not only the features of the input but especially the characteristics of the particular matcher itself play a crucial role. Figure 5.10 illustrates how well the approaches take into account the different processing types. While almost all the systems follow the black box paradigm (i.e. automatic execution without human intervention), AOAS and RiMOM require some manual pre-processing, and, additionally, the latter needs some post-processing effort.

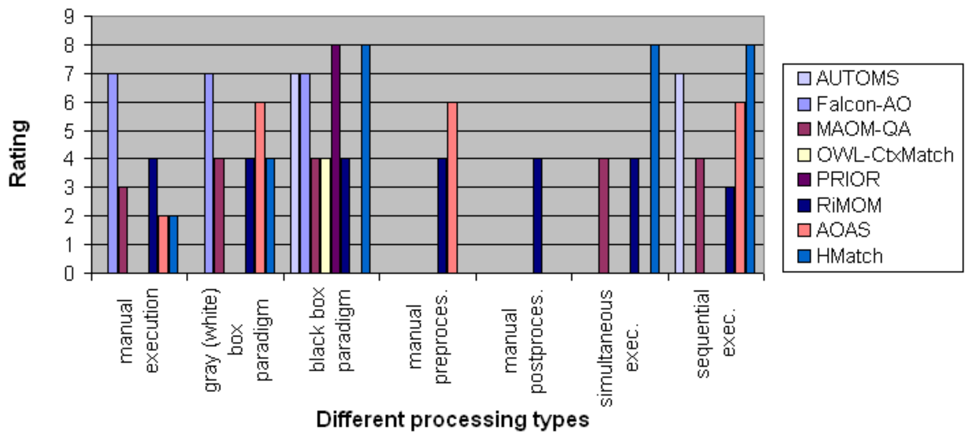


Figure 5.10: How good is the approach for different processing types?

5.3.4 Usability-related questions

Besides the features previously mentioned, the matcher developers were asked to rate the approaches regarding the adaptation for different application purposes (integration, translation, reuse of sources, etc.), various application goals (local use, internet use), as well as adaptation for different tasks, applications, and domains. In Figure 5.11, we show an example of how the approaches behave regarding the local, network, and internet usage. Unfortunately, most of the approaches have been developed to be run on local machines, and only four systems (Automs, Falcon, MAOM, and ISLab HMatch) can be accessible on a network or as an online service.

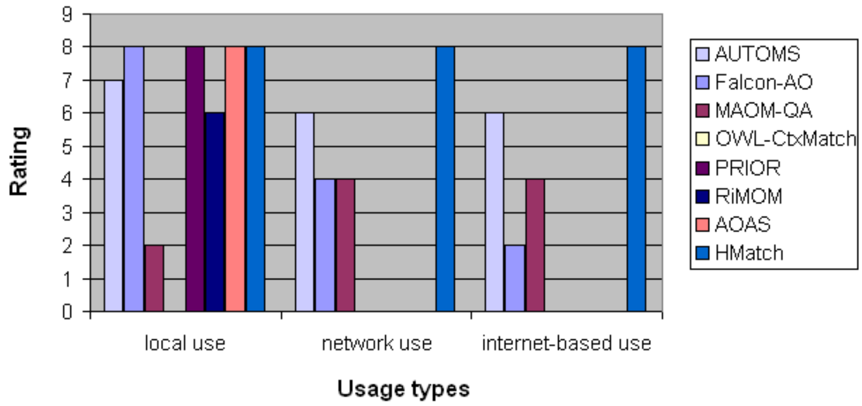


Figure 5.11: How good is the approach for different types of usage?

5.3.5 Output-related questions

The characteristics of the output is a constraint, as well as those of the input. In Figure 5.12 we take a look at the output cardinality, one of the properties describing the output of the approaches. Falcon and ISLab HMatch support (very good and extremely good, respectively) each type of the cardinality with the same intensity. Furthermore, AOAS serves only the global cardinality, RiMOM concentrates on (global and local) 1:1, while OWL-CtxMatch supports (global & local) n:m mapping.

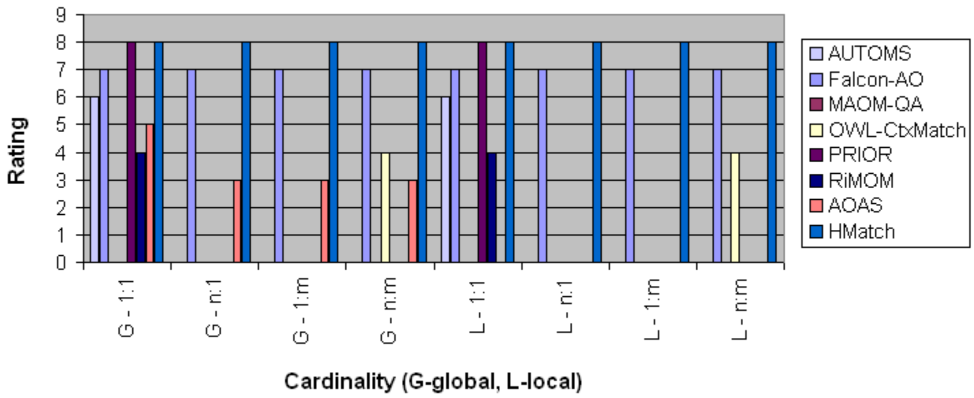


Figure 5.12: How well does the approach support global/local cardinality?

5.3.6 Documentation-related questions

The most subjective characteristic to rate, in our opinion, is the quality and clarity of the matcher documentation. All developers (except the OWL-CtxMatch developers) assign ratings between low (3) and high (5) for each aspect of the matcher documentation queried (cf. Fig. 5.13).

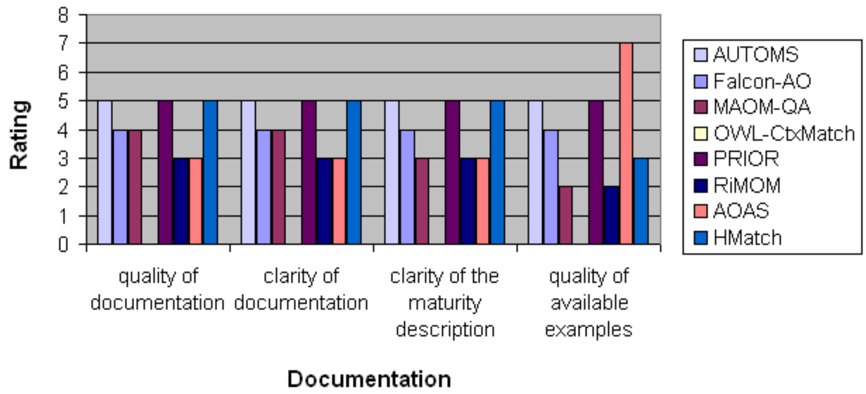


Figure 5.13: How high is the quality of the documentation?

5.4 Final remarks on MCMA

In this chapter, we have described the development process of the multilevel characteristic for matching approaches (MCMA), which consists of dimensions, factors, and attributes. In specific, we have elaborated how we have identified relevant dimensions in matcher context, investigated them, and analyzed the characteristics of these matchers along these dimensions. This inventory characteristic is very precise and goes beyond most published characterizations of matching systems in terms of attributes [56, 98, 102, 157, 169]. The preliminary MCMA has been evaluated in order to assure the quality of the model developed wherein the evaluation process was dedicated to the overall relevance and accuracy of the *a priori* multilevel characteristic regarding the matcher issue.[17] Since the suitability of the given approaches is determined w.r.t the requirements of the application and with careful consideration of a number of factors, the goal of the evaluated and revised MCMA is to serve as a basis for the decision making process for selecting the most appropriate algorithm and, in turn, to establish more precisely the adequacy of a matcher within an application.

[17]For details regarding the evaluation process, the reader is referred to Sec. 8.3.

Chapter 6

Manual Approach

Abstract In this chapter, we elaborate on the manual part of the Metadata-based Ontology Matching (MOMA) Framework that, by applying a flexible methodology and providing a tool based on decision-making process, supports human-based selection of suitable matching approaches. Taking into account the factors defined within the Multilevel Characteristic for Matching Approaches (MCMA), we have determined the suitability of the given matcher with respect to the requirements of the application in which the matcher is to be deployed. To select suitable matchers, we have adopted one of the methods from the family of Multiple Criteria Decision Analysis (MCDA) — the Analytic Hierarchy Process (AHP). In the course of the chapter, we explain why we decided to adopt this particular MCDA method for our purposes and describe how the decision making process has been transformed into the matching approach selection (cf. Fig. 6.1).

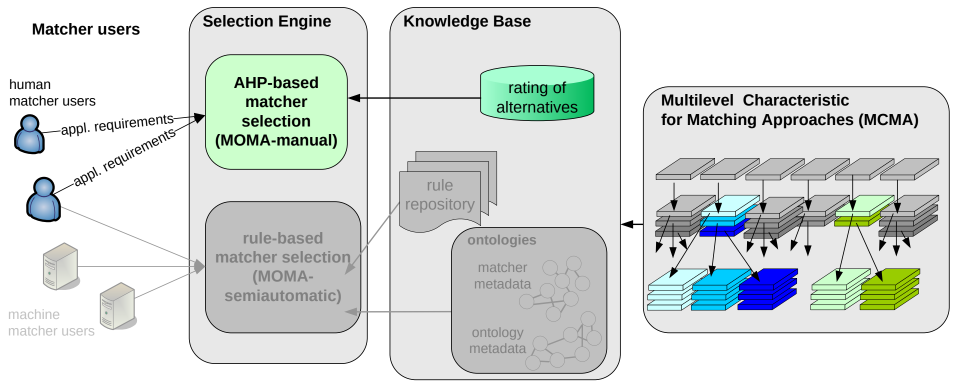


Figure 6.1: Manual MOMA Framework

6.1 Multi-Criteria Decision

Decisions have not only inspired reflection by many thinkers since ancient times, but are also inseparably linked to the classic works in economics, physics and computer science [62]. When people are faced with a decision, they usually make it based on their experience, intuition or feelings, or advice from others. Since a decision is the result of an interaction between different actors influenced by a particular context, it can take on one of three facets: completely rational, completely irrational, or completely non-rational (cf. Fig. 6.2). A *rational decision* consists of the evaluation of all the alternatives and then choosing the one that maximizes the decision maker's satisfaction or needs, while a *non-rational decision* is based on the decision maker's experiences and knowledge, and an *irrational decision* considers only personal aspirations and aversions [76].

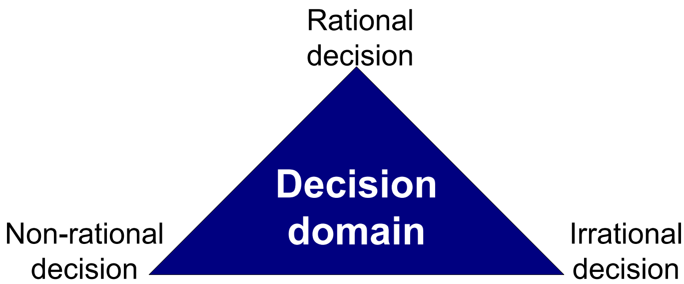


Figure 6.2: Decision domain [76]

As long as decisions (i) rely on a single criterion that serves as the basis for comparing alternatives, or (ii) the scales of the different criteria are consistent, and (iii) numeric measures accurately capture prospective performance, summary statistics or, in some cases, just acting on human instinct may be sufficient for the decision making process. However, when the decision depends on multiple criteria and, additionally, the contributing factors are on scales that do not lend themselves well to comparison, the process becomes very complex and difficult, and the involvement of qualitative as well as quantitative methodologies and tools are indispensable. Consequently, a multi-criteria decision making process is required in such cases. Such a process, also known as a *MultiCriteria Decision Analysis (MCDA)*[1], is a quantitative approach to aid decision makers in the evaluation of decision issues involving numerous and conflicting criteria (variables). Since almost all complex human decisions are multi-criteria decisions, MCDA approaches can be used not only in

[1]Multicriteria Analysis is called Multiple Criteria Decision Making (MCDM) by the American School and Multicriteria Decision Aid (MCDA) by the European School

different (research) domains [120, 78] like energy policy [89], health policy [21], and technology assessment [86], but also in everyday life where personal decisions, such as choosing a new home, depend on heterogenous criteria like price, location, public transportation, and security.

A key characteristic of most MCDA problems is that they generally do not have conclusive or custom solutions, and for this reason the MCDA methods aim to help make recommendations that lead to better decision making. Since a single alternative usually is not as good or superior to all the other considered alternatives in terms of attributes, the alternatives given cannot be directly compared, as is in the case of scalar function. A central idea of most MCDA methods is that the approaches can combine all the criteria defined into a single scalar objective function and the "best" solution awarded the highest score. The common MCDA methodology can be defined as a process based on two main phases, *construction* and *exploitation*, which, in turn, consist of the four steps shown in Fig. 6.3:

- input capabilities (information) that concern and characterize the information accepted by the aggregation procedure;
- a preference elucidation and modelling phase that forms criteria and determines inter-criteria information; this includes trade-offs, lotteries, direct ranking, and pairwise comparison;
- MCAP - Multi-Criteria Aggregation Procedure (to aggregate the alternative evaluations and exploration of the aggregation), a matter of algorithmic and mathematical methods that address a specific decision issue where the mathematics serve as the basis in the search for a core, ranking, sorting, reduction of graph circuit, etc.;
- making recommendations characterizes the information produced by the aggregation procedure.

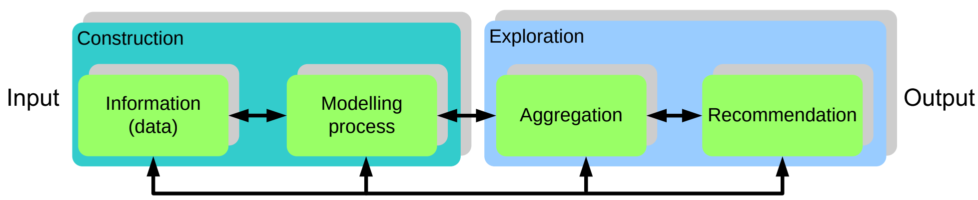


Figure 6.3: General schema of MCDA methods [76]

Since there are many different MCDAs that require different types of raw data and follow different optimization algorithms, there are also different ways these methods can be classified: Some rank options, while others identify a single optimal alternative, still others provide an incomplete ranking or differentiate between acceptable and unacceptable alternatives [120]. If we consider, for instance, the diverse MCAPs of the methods, we can distinguish between:

- the *single synthesizing criterion* approach that establishes an aggregation function and represents at best the decision maker's preferences;
- the *outranking synthesizing* approach that leads to different order structures depending on the preference relations considered, the hypotheses regarding the properties of these relations (transitivity, etc.), and the use of thresholds; and
- the *interactive local judgments* with the trial-and-error approach.

Another classification system, with which the majority of specialists agree, says that MCDA methods can be divided into two main groups regarding regional context:

- methods of *American inspiration*, based on the utility function (e.g. AHP, UTA), that aggregate different criteria (points of view) into one global criterion called utility function; these methods eliminate incomparability between variants; and
- methods of *European (French) inspiration*, based on the outranking relation (e.g. ELECLTRE, PROMETHEE), that take into account the incomparability between variants [204].

There are, of course, many more MCDA classifications, though their examination is not the scope of this work. In the following, we will concentrate on the most common classification of the principal MCDA approaches that distinguishes between elementary methods, outranking methods, and multi-attribute utility and value theories:

Elementary methods are intended to reduce complex problems to a singular basis for the selection of preferential alternatives. Although easy to implement, the elementary methods reduce complex problems to a singular metric and thus can result in an oversimplified and often overly conservative representation of the problem [120]. They are mostly applied to cases in which a problem has just a few alternative solutions due to limited criteria. Such methods can include pros and cons analysis, maximin and maximax methods, or decision tree analysis.

Outranking is based on the principle that one alternative is more dominant, in terms of the degree of dominance, than another, i.e. one alternative tops another if it outperforms on certain criteria or performs at least as well on all other criteria. The preference of two alternatives *a* and *b* regarding a given set of criteria can be expressed by using the following relations: (i) *a* is strongly preferred to *b*, (ii) *a* is weakly preferred to *b*, (iii) *a* is indifferent to *b*, and (iv) *a* cannot be compared to *b*. In this context, an alternative is said to be *dominated* if it performs poorly in some criteria and on a par in others [120], which in turn means that the *dominant alternative* is better or equal in all respects. Outranking is a partially compensatory method that does not rely upon optimization. Some of the most important outranking approaches are:

- ELECTRE method [2] — the first method using an outranking approach [24] — reduces the size of alternative sets by eliminating to a certain degree alternatives dominated by another. The underlying principle stems from processing a system of pairwise comparisons of the alternatives. This procedure involves the use of a system of preference relations (SPR) based on the exploitation of the outranking relation. According to this relation, an action will outrank another if the first action is considered to be at least as good as the second [137]. Furthermore, ELECTRE is based on the assumption that the system of individual (decision maker's) preference relationships can be described by means of a sophisticated analysis of the relationships of preference, indifference, and incomparability, whereby the latter enters as a factor into the final ranking. This complex analysis involving the assessment of several thresholds and preference relationships may be judged as a remarkable effort, though it may lead to cumbersome work. Often it is very difficult to express credible threshold values, and the consequent ranking can end up being hard to understand [137, 204].
- PROMETHEE I and II are based on the same principle as ELECTRE, however, they require very clear, additional information (information between the criteria and information within each criterion). The ranking of the alternatives with respect to certain criteria in PREOMETHEE I is used to calculate a partial pre-order [62, 104].

Outranking methods have been used, for example, in the selection of a solid waste management system, the location of a waste treatment facility, and for nuclear waste management [103].

[2] The first version of ELECTRE, the ELECTRE I, was followed by many others: ELECTRE IS, II, III, IV.

Multi-Attribute Utility and Value Theories (MAUT) is a technique that assigns a utility value in the form of a real number to each criteria, and in this way allows a simple expression of the decision maker's preferences. The decision taken is rationally based, since the MAUT method relies on the assumption that the decision makers are rational: (i) more is preferred to less, (ii) preferences do not change, and (iii) preferences are transitive. With respect to the indicators and data, MAUT approaches can handle different scales as well as different issues. For this reason, they are common and frequently used to solve real life problems. The main consideration of these approaches is: How great is an effect and how important is a criterion relative to the other criteria [120].Within the family of MAUT approaches we can include the following methodologies:

- UTA methodology evaluates the alternatives using a set of utility functions modeled after the decision maker's *a priori* preferences. The method searches for the "optimal" shape of the function using linear programming, i.e. a shape that best reflects the decision maker's preferences [101]. In the real world, the UTA methods have been mainly deployed in financial management, human resource management, and marketing.
- Analytic Hierarchy Process (AHP) uses pairwise comparisons along with a semantic and ratio scale to assess the decision makers' preferences [76]. The AHP algorithm is composed of four main phases: construction of the hierarchical structure of the issue, definition of the preferential information (relative weights) and calculation of the absolute weights, coherence analysis, and construction of the final ranking. All variants are ultimately ranked according to their utilities [204]. AHP is very useful when the decision-making process is complex.
- MACBETH (Measuring Attractiveness by a Categorical Based Evaluation Technique) supports interactive learning in regard to the evaluation issue and the elaboration of recommendations that consider priorities and select alternatives in the decision making process. It is designed for building a cardinal numerical scale to measure the attractiveness of the alternatives in the course of a learning process [52].

In risky project choices, MAUT ensures that the final selection correspondences with the decision maker's preferences. It is, as well, a mechanism used in cases too complex to be handled satisfactorily by intuition, as long as the analysts are aware of the underlying assumptions and bear in mind the decision making process [137]. MAUT approaches are among the most scientifically grounded methods with a firm foundation in decision theory [120].

Mixed Methods are approaches that cannot be classified into the first three groups. The exponent of this kind of MCDA is the QUALIFLEX that uses successive mutation to provide a ranking of the alternative corroboration with the ordinal information [76].

Due to the ranking of the alternatives, and consequently the particular recommendations based on the evaluation of the attributes for each alternative, the weight of the attributes, and the synthesis or aggregate function, the choice of a MCDA method plays a crucial role in the decision process. Since numerous MCDA methods have been proposed to help to deal with complex decision making, it is extremely challenging, if not impossible, for the decision makers to know all the possible approaches. As none of the MCDA methods can be considered appropriate for all the decision issues, and since there is no consensus regarding which method is the most suitable for the various kinds of decision making situations, the decision regarding the suitability of a particular method is a complex and tedious process. Furthermore, as it is difficult to determine which aspects are to be considered in the selection of an appropriate approach [77], the decision usually depends on the problem involved, which model decision makers are most comfortable with, as well as their familiarity or affinity with it. According to [120], selecting an approach from the methods available may be itself an expression of subjective values or a purely pragmatic choice (such as familiarity or perceived ease of implementation). However, according to Guitoni and Martel [76], the proper way to choose the "right" method is

> *"...to select (or adopt) the method that can handle "correctly" the (given) situation..."*.

For this reason they have delineated general guidelines (G1-G6[3]) which aim to support the decision makers in the first step towards the decision making process — in the selection of the method appropriate to their particular issues. In order to find an appropriate MCDA method for the matcher issue, in the following we introduce the Guitoni and Martel's guidelines and sketch out the way we interpreted and applied them to our situation.

Guideline G1: *determine the stakeholder in the decision process*

In our case, the stakeholders are the potential researchers or developers of semantic web-based applications looking for suitable matchers.

[3]Though Guitoni and Martel [76] formulated seven guidelines, in the following we will analyze only the first six of them. As the last guideline treats the tool support of a particular method, it is not relevant for our considerations, since we are developing a tool for the chosen method.

Guideline G2: *consider the decision maker's mindset when choosing a suitable preference elucidation mode (tradeoffs, rating, lottery, pairwise comparison) and order (total filtration, semi-order, partial semi-order, partial pre-order, total pre-order).*

Since pairwise comparisons have been proposed as an effective and intuitive approach to elicit qualitative data for multi-criteria decision making [185], and because they can be used for all decisions based on multi-criteria and for which a method that requires qualifying the qualitative data is to be applied, we have decided to deploy a method that supports pairwise comparisons and delivers a total pre-order of the alternatives.

Guideline G3: *determine the decision issue (problem formulation or statement) pursued by the decision maker.*

B. Roy defines in [161] *decision problematic* as the conception of the way the decision maker envisions the aid to be supplied to the problem in question, based on answers to questions like: What kind of results are to be obtained? With which terms should the problem be posed? Within this context, we have to distinguish between:

- description problematic, which describes alternatives and their consequences in a formalized and systematic way;
- choice problematic, which chooses the best alternative or develops a selection procedure;
- sorting problematic, which categorizes alternatives according to norms; and
- ranking problematic, which ranks the alternatives in decreasing preferences or builds an ordering procedure.

For instance, to choose the appropriate matcher alternatives, we need to rank the alternatives given, i.e. in our case, we have to deal with the choice and ranking problematic.

Guideline G4: *choose the MCAP capable of handling the input information available (which can be expressed as cardinal or ordinal, certain or uncertain, mixed, etc.) and for which the decision maker can provide the information required.*

One should not use a MCDA method conceived to handle cardinal and a specific set of information to process ordinal or uncertain information [76].

Guideline G5: *consider the discrimination power of the criteria, the compensation degree between them (if the decision maker refuses any compensation, then many MCAP's will not be considered), and the intercriteria information needed.*

In the context of compensation degree, differentiation can take place between:

- compensatory compensation - There is absolute compensation between the different evaluations.
- non-compensatory compensation - No compensation is accepted between the different dimensions of criteria and the criteria itself.
- partially compensation - Some kind of compensation is accepted between various dimensions and criteria (most of the methods fall into this group) [76].

In our case, since we do not know exactly who will play the role of the decision maker in a particular case of matcher selection, we must provide some compensatory allowance between the dimensions and criteria, i.e. we need to apply a method that provides for partial compensation.

Guideline G6: *the fundamental hypothesis (e.g. transitivity, dominance in which one option outperforms the others in one state and is on par in all other states, invariance, where different representations of the same choice problem should yield the same preferences) on which all the theoretical and axiomatical developments of a method based are to be met.*

In the case of matcher selection, dominance and invariance play the crucial roles.

After a review of the guidelines regarding the selection of an appropriate MCDA method and definition of the corresponding specification of our matcher case, we can now analyze which method satisfies our requirements. A composition of some of the MCDA methodologies described earlier, together with the particular guidelines and the most important issues crucial for the determination of an appropriate MCDA approach, is illustrated in Tab.6.1[4].

[4]The characteristics vital to the matcher issue are marked in blue.

Method	Guideline									
	G2		G3	G4			G5			G6
	Pref. elucid. mode	Order	Decision probl.	Information*			Discrim. power of criteria	Compens.	Inter-criteria	Hypo-thesis**
				o	c	m				
elem. methods	direct rating	total preorder	choice	X	X	-	absolute	none	-	ind., inv., tran., dom.
UTA	trade offs	total preorder	choice	X	-	-	absolute	partially	indirect	ind., inv., tran., dom.
AHP	pairwise comp.	total preorder	choice ranking	-	X	-	absolute	partially	total explicit	ind., inv., dom.
ELEC-TRE I	pairwise comp.	core	choice	X	X	X	absolute	partially	total explicit	ind., inv., coal.
PROME-THEE I	pairwise comp.	partial semiorder	ranking	X	X	X	non absolute	partially	total explicit	ind., inv., coal.
QUALI-FLEX	pairwise comp.	total semiorder	ranking	X	-	-	absolute	partially	total or partial & explicit	ind., inv.,

*o - ordinal, c - cardinal, m - mixed
**ind.-independence, inv.-invariance, tran.-transitivity, dom.-dominance, coal.-coalition; social choice theory

Table 6.1: Comparisons of MCDA methods based on the selection guidelines (adapted from [76])

Furthermore, the different experiences that have been collected over years during the application of MCDA methods to various domains and cases can be summarized in following findings:

- outranking methods do not always take into account whether over-performance in one criterion can make up for under-performance in another [119];
- ELECTRE and AHP methods are the most reliable and user friendly MCDA methods; the models of preferences proposal in these methods and final rankings generated by them are highly appreciated [204];
- while the UTA method is recommended for decision problems with a larger number of variants and ELECTRE should be applied to smaller instances, the AHP method can be applied in both cases [204].

With these facts and the MCDA guidelines in mind, we have determined that the best approach for tackling our matcher problem is the **Analytic Hierarchy Process (AHP)** method. In our opinion, the main advantage of AHP that should be highlighted is its ability to provide a systematic, validated approach for the consolidation of information regarding alternatives using multiple criteria. As the AHP method has been shown to be a fruitful and varied methodology that has been useful in a wide range of fields like strategic planning, engineering design, project evaluation, impact analysis [197], as well as software selection [91], software engineering, and ontology engineering (in the selection of an appropriate ontology) [121], to name but a few, we feel justified in our choice. Furthermore, AHP belongs to the MAUT approaches, which, as mentioned before, are among the most scientifically grounded methods with a firm foundation in decision theory [120].

Since we have decided to utilize AHP for the matcher selection, in the following sections, we first take a closer look at the AHP methodology and explain its adoption for the matcher issue.

6.2 Analytic Hierarchy Process

The Analytic Hierarchy Process (AHP) developed by Thomas L. Saaty is a basic approach for decision making and a general theory of measurement. Since its introduction in 1970 and first publication in a book in 1980 [163], numerous analysts and decision makers have applied this MCDA method in a variety of domains to solve different problems (the federal government [199], the electric utility industry [122], politics [53], etc.). AHP is a quantitative comparison method used to select optimal alternatives by comparing the alternative solutions based on the relative performance of the criteria. In the AHP objective value information, expert knowledge and subjective preferences can be utilized since

> *"...the experience and knowledge of people are at least as valuable as the data they use"* [197].

The AHP method assists decision makers in organizing their thoughts and judgments with the goal of making better and more effective decisions, selected on the basis of the greatest value of the objective function. Furthermore, AHP takes into account the considerations of Hahn [79] regarding the need for a structured results-based approach to decision making that allows for trade-offs within the systematic method, including all perspectives and considerations. It is based on a theory of ratio scale estimation and enables the application of qualitative criteria in the evaluation of alternatives, i.e. it can be transferred into a ratio scale by using pairwise comparisons of qualitatively expressed measures [103].

> *"A useful feature of the AHP is its applicability to the measurement of intangible criteria along with tangible ones though ratio scale"* [197].

The AHP is a systematic approach developed to structure an expectance-, intuition-, and heuristics-based decision making process into a well-defined methodology founded on sound mathematical principles[11]. This approach helps its users to set priorities and to make the best decision when both qualitative and quantitative aspects of a decision need to be considered[164]; AHP provides a mathematically rigorous application and a proven process for prioritization and decision-making. By reducing complex decisions to a series of pairwise comparisons and then synthesizing the results, decision makers arrive at the best decision based on a clear rationale. It is generally accepted that AHP constitutes one of the best options for aiding multi-criteria decision making, since it compares the relative importance that each criterion has with respect to the others while enabling the relative weight of the criteria to be calculated. It normalizes, at the end, the weights in order to obtain the measures for the existing alternatives.

6.2.1 AHP Steps

In the following, we outline, on a rather abstract level, the main steps of the AHP method: (i) problem definition, (ii) hierarchy building, (iii) alternative indication, (iv) pairwise comparison, and (v) result calculation, in order to elaborate, in the further course of this chapter, how the AHP method supports the matcher selection issue.

STEP #1 Define the problem or the project objectives

STEP #2 - Build a hierarchy of decision
In the second step, AHP provides a means to break down the problem, defined in the previous phase, into subproblems (goal, criteria, sub-criteria, and alternatives) that can be more easily comprehended and subjectively evaluated [11]. The simplest way to depict the decision problem is a three-tier hierarchy: the goal of the decision or the objectives of the problem form the base while the leaf nodes represent the alternative solutions to be compared, and stacked between these two levels are various criteria. In more complex situations, while breaking down the problem into subproblems, the hierarchy developed contains more than one level of criteria. In such cases, the goal at the top and the alternatives expressed by the leaves are connected by the criteria, which are further defined by sub-criteria, and these, in turn, are divided into sub-sub-criteria, etc. (cf. Fig. 6.4).

Hierarchical decomposition applied in the AHP appears, as stated in [165], to be a basic device used by people to deal with diversity and, moreover, allows judgment to be focused separately on the individual properties essential to the formation of a sound decision level. Thus the hierarchical structure provides insight into a complex problem and an overview of the relationship between the particular sub-

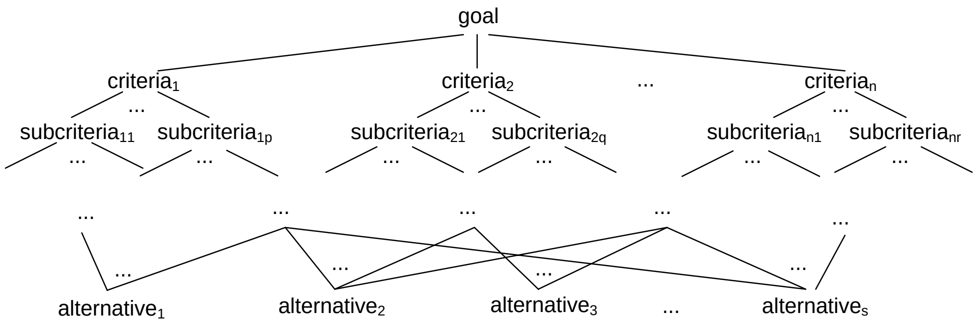


Figure 6.4: AHP hierarchy structure

problems and attributes while helping decision makers to determine whether the issues on each level (attributes, sub-attributes, etc.) are of the same degree of magnitude. Moreover, the AHP allows decision makers to compare more accurately the homogenous elements within one level.

STEP #3 - Indicate a preference for each alternative
In this phase, the AHP method asks the decision makers/domain experts to indicate a preference or priority for each decision alternative in terms of how it contributes to the individual criterion (defined in the previous step). Given the information on relative importance and preferences, a mathematical process is used to synthesize the information and provide a priority ranking of the alternatives given in terms of their overall preference. After the comparison of the alternatives, the relative importance of one solution over another can be expressed. According to Saaty [34], the best approach to handling such a situation is the eigenvector solution[5], which provides relative ranking of the alternatives.

STEP #4 - Build a pairwise comparison
The most effective way to concentrate judgment is to take a pair of elements and compare them on the basis of a singular property, while putting aside the other properties and elements [164]. For each level of criteria (sub-criteria and criteria), a pairwise comparison between the sibling nodes is to be formed and organized into a square evaluation matrix (cf. Equation 6.1), i.e., the comparisons are applied to pairs of homogenous elements on a corresponding level within the AHP hierarchy and use the Saaty's fundamental *"validated for effectiveness"* [165] scale shown in Tab. 6.2.

[5]Calculation example will be shown in Sec. 6.3.3

$$A = \begin{bmatrix} a_{11} & \cdots & a_{1i} & \cdots & a_{1n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{i1} & \cdots & a_{ij} & \cdots & a_{in} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{n1} & \cdots & a_{nj} & \cdots & a_{nn} \end{bmatrix} where \tag{6.1}$$

∀ x = 1, . . . , n ∀ j = 1, . . . , n: $a_{ij} > 0$
∀ i = j: a_{ij} = 1
∀ i = 1, . . . , n ∀ j = 1, . . . , n: a_{ij} = ${a_{ij}}^{-1}$

Intensity of importance	**Definition**	**Explanation**
1	equal importance	two activities contribute equally to the objectives
2	weak	intermediate value
3	moderate importance	experience and judgment slightly favoring one activity over another
4	moderate plus	intermediate value
5	strong importance	experience and judgment strongly favoring one activity over another
6	strong plus	intermediate value
7	demonstrated importance (very strong)	an activity is favored very strongly over another
8	very very strong	intermediate value
9	extreme importance	evidence favoring one activity over another is of the highest possible order of affirmation
reciprocal		if activity *i* has one of the above numbers assigned to it when compared with activity *j*, then activity *j* has the reciprocal value when compared with *i*
rationals	ratios arising from the scale	if consistency is forced by obtaining n numerical values to span the matrix

Table 6.2: Fundamental AHP-scale developed by Thomas L. Saaty [165]

STEP #5 - Calculate the final result

The final weights of the elements at the bottom of the hierarchy are obtained in the process of the hierarchic composition; this is done by adding together all the contributions of the element in each level with respect to the elements in the level above. In the strictest sense, the ratings of each alternative from step #3 are multiplied by the weight of the sub-criteria, carried out in step #4, and then aggregated to obtain local ratings with respect to each criteria. The local ratings are then multiplied by the weights of the criteria (cf. step #4) and aggregated to the overall ratings. The final value is used to make a decision regarding the problem defined in step #1.

6.2.2 Consistency in AHP

An important consideration within the pairwise comparison is the consistency of the judgments made by the decision makers. Since perfect consistency is practically impossible to achieve, a method to measure the degree of consistency among the pairwise judgments provided by the decision makers is needed. In order to verify the consistency of the pairwise comparisons of the evaluation matrix, the maximal eigenvector λ_{max} has to be determined and, consequently, the consistency index (CI) can be calculated:

$$CI = \frac{\lambda_{max} - n}{n - 1}, where \tag{6.2}$$

λ - maximal eigenvector
n - number of items being compared

Furthermore, a measure of consistency used by the AHP for decision support of the pairwise comparisons within the evaluation matrix that need to be revised is known as the consistency ratio (CR) and can be calculated:

$$CR = \frac{CI}{RI}, where \tag{6.3}$$

CI - consistency index
RI - random consistency index

Random consistency index (RI) is the index of a randomly generated pairwise comparison matrix. It depends on the number of elements (n) to be compared and assumes the values shown in the Tab. 6.3.

n	RI	n	RI
2	0	9	1,45
3	0,52	10	1,49
4	0,89	11	1,51
5	1,11	12	1,54
6	1,25	13	1,56
7	1,35	14	1,57
8	1,40	15	1,58

Table 6.3: Average Random Consistency Index (RI)

The consistency ratio is designed so that values of the ratio exceeding 10% (0.1) are indicative of inconsistent judgments. The measure of inconsistency can be used to gradually improve the consistency of judgments [165]. If the degree is lower than 10%, the values of the pairwise comparison are said to be acceptable and the decision process can be continued; otherwise, the decision-makers should rethink and revise the judgments already weighted before continuing with the analysis.

The Analytic Hierarchy Process (AHP) has been identified as the most suitable approach among the methods in the Multiple Criteria Decision Analysis (MCDA) family for adaptation in the context of the matching issue. In the course of the next section, we explain how this particular decision making process has been "customized" to the matching approach selection and give a brief overview of the tool we developed based on the AHP approach to help decision makers "on the way to the suitable matcher".

6.3 AHP in the context of Matcher Selection

Considering that the ontology development process is frequently conducted by people without a high level of expertise in the ontology matching domain, there is a need for means to aid them in the selection process and, in the next step, to apply ontology management tools such as matching algorithms. From our point of view, the obvious solution to making full use (and to limit the disadvantages) of existing matching approaches is a strategy based on the reuse of available matchers wherein the most important step is the selection of appropriate matchers. Both a blessing and a curse is that there are many different approaches, all with various methods of processing and capable of dealing with sources of all sorts and delivering different results (cf. Sec. 3.5); there are miscellaneous matching approaches which have different advantages and disadvantages, depending on the various heterogenous features. However, in view of the diversity of matching approaches, we have one goal: finding approaches suitable for a particular situation without losing sight of the needs of the application or task. The problem here is that a decision over the suitability of a matching approach depends on multiple criteria that cannot be easily compared, so that a decision becomes considerably complex and difficult. Furthermore, to allow the matcher users (e.g. ontology engineers, semantic web-based application developers, researchers) to have an influence on the selection process, on the one hand, and to be able to deliver results that are suited to the given context (i.e. application requirements where the matcher is to be applied), on the other, manual selection of algorithms with respect to the given requirements needs to be supported. To tackle the issue of multiple criteria and to allow the manual selection of matching approaches and thereby serve the human matcher users, as envisioned in our MOMA Framework (cf. Sec. 4.2.2), the involvement of qualitative and quantitative decision making methodologies becomes essential. For this reason, we have adopted one of the approaches from the Multiple Criteria Decision Analysis (MCDA) that is based on a mathematically rigorous method and provides a proven process for prioritization and decision making – the Analytic Hierarchy Process (AHP) – and that ultimately facilitates the selection of matching approaches. By reducing the complex decision regarding suitable matchers to a series of pairwise comparisons and synthesizing the final results, decision makers arrive at the best decision based on a clear rationale – suitable matching approaches [164].

The abstract of this chapter includes a figure that, on a high level, illustrates the architecture of the AHP-based part of the MOMA Framework already described in Sec. 4.2.3. In the course of this section, we delve into the entire AHP-based process of selection matching approaches, including the specification of the tool developed. Since the particular steps of the AHP approach, as sketched in the latter section, can be seen as pieces of a puzzle which need to be put together in order to find the final solution for a given problem, we piece together these separate puzzle parts to determine the suitable approach. In addition, since the application of AHP-based MOMA Framework to a real case study is described in the context of the evaluation of MOMA Framework in Chapter 8, in this section we provide a rather simplified example to better explain the functionality of this part of the framework. For a better understanding of the issue of matcher selection and our proposed solution, we have reduced the de facto situation to the problem of choosing the best matcher among two approaches: OCM (OWL-CtxMatch) and PRIOR[6, 7].

6.3.1 Definition of a Problem

The first step of AHP is to either define the problem to be solved or to outline the project objectives. Considering the diversity of the matching approaches and the goal of the reuse strategy, as described in Sec. 4.2.1, we can formulate the main goal as follows: with respect to existing matching approaches, *determine which are suitable*[8] *to the given context (application/task requirements)*.

Example
With two approaches in hand, OWL-CtxMatch and PRIOR, we wish to find out which of them is (more) suitable for our purposes.

[6]For more details regarding the approaches, the reader is referred to Sec. 3.4

[7]Please note that since it is only a very simplified example used to clarify the matcher issue, the data as well as the achieved results do not express the real quality of the used approaches.

[8]The meaning and utilization of the term “suitability” in the context of matcher selection has been explained in Sec. 1.2.1

6.3.2 Building a "Matcher" Decision Hierarchy

To find an answer to the issue defined in the previous section, the problem must be broken down into subproblems, which, in turn, as provided in the AHP methodology, form a decision hierarchy that includes goal, criteria, sub-criteria (sub-sub-criteria, etc.) and solution alternatives. Choosing the criteria relevant to the problem is one of the most important phases within the AHP-based decision making process. Here, the developers of the decision hierarchy have the freedom and duty to deliberate on its level of consolidation. As recommended in [108], the consolidation of the criteria should only be carried out to the extent that it does not sacrifice important information (some consolidation may be necessary in order not to flood the decision makers with too much detailed, unstructured information). At this point, we must consider:

- what is important for our decision regarding the determination of suitable matching approaches;
- which criteria to include into the decision hierarchy with regard to the impact of the selection of the adequate matching approach; and
- where to include (as criteria, sub-criteria or sub-sub-criteria) the particular criteria.

Furthermore, during the building of our hierarchy, we must follow certain rules so as to ensure that there are sufficient details for the matching issues. Hence, we must:

- illustrate the matching problem as thoroughly as possible but *"not so thoroughly as to lose sensitivity to change in the elements"* [164];
- consider the environment of the matching approaches and the environment of their selection and application;
- identify the matching issues or attributes that affect the decision in terms of the suitability of an approach, and that can contribute to the the final result; and
- identify participants that are relevant to the issue of matcher selection.

Since the decision concerning matchers suitability depends on the hierarchy developed, the compilation of a set of properties is a significant step towards the realization of the AHP-based MOMA Framework. Therefore the development of the decision hierarchy to be applied within the selection process should be based on indepth knowledge of the matcher domain as well as experiences gathered during the development and application of various matching approaches. Furthermore, the hierarchy must be developed in collaboration with domain experts. All these requirements are fulfilled by our (evaluated[9]) Multilevel Characteristic for Matching Approaches (MCMA) described in Sec. 5.2. Our MCMA is built in a hierarchical structure, ranging from the uncertain (upper levels) to the relatively certain (lower levels) criteria; it first defines the particular and then moves to the general attributes (in contrast to the development from the general (upper levels) to the particular (lower levels) [197]). MCMA includes six main criteria – dimensions – that form the superficial collection of matcher attributes corresponding, at the same time, with the first level of criteria according to the AHP methodology (cf. Fig.6.4). These are:

- *input characteristic* that considers the matcher input (ontologies to be matched, auxiliary information);
- *approach characteristic* that describes the matching algorithms themselves;
- *output characteristic* that defines the desired result of the matching execution;
- *usage characteristic* that considers the various situations in which the approaches have been (are to be) used;
- *documentation characteristic* - points out the existence and type of documentation; and
- *cost characteristics* that addresses the the usage costs of the algorithm.

These dimensions are defined by sets of factors (cf. Fig.6.4 - sub-criteria) which are, in turn, described by attributes (sub-sub-criteria). The matcher decision hierarchy (cf. Fig. 6.5) needed for the decision process is utilized by our developed MCMA, whereas the goal defined in Sec. 6.3.1 (level 0 of the decision hierarchy) is connected though three levels of criteria: 1^{st} level - MCMA dimensions, 2^{nd} level - MCMA factors, and 3^{rd} level - MCMA attributes with alternative solutions – different matching approaches.

[9]Evaluation of the MCMA is detailed in Sec. 8.3

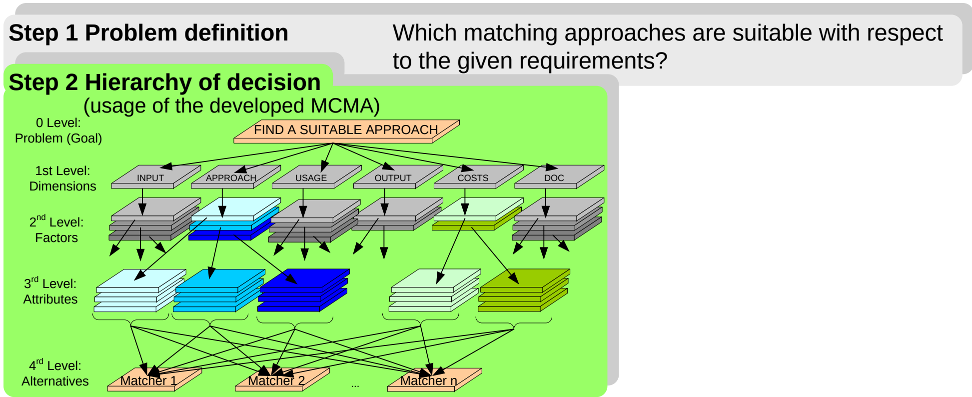


Figure 6.5: Hierarchy structure

Example

For easy understanding, we have reduced the decision hierarchy to three levels in total, and so, in our exemplary matcher decision hierarchy, the goal, as it is described in Sec. 6.3.1, is connected across one level of criteria that affect the final decision: *large input size*, *tree structure* and *instances* (exemplary features of the MCMA), with the possible solutions: matchers *PRIOR* and *OCM*. The dummy hierarchy, which is used in the entire example, is shown in Fig. 6.6.

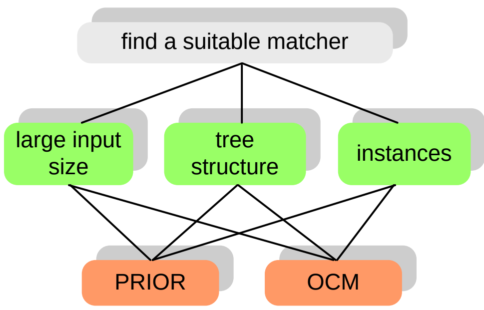


Figure 6.6: Dummy decision hierarchy

6.3.3 Collection of matcher data

To collect data on the various alternatives of matching approaches and to determine the suitable candidates among them, we first need relevant information regarding the alternatives. For this reason, we have developed (following the hierarchical structure of the matching characteristic) an online questionnaire (to be filled out by domain and matching experts, cf. Sec. 5.3) that allows the addition and rating (using a predefined Saaty's scale from 0 to 8; cf. Tab. 6.2) of new matching alternatives. The preliminary comparisons of the different approaches based on the data that form the questionnaire have been reviewed by matcher experts in order to ensure the objectivity of the comparisons. After the comparison of the alternatives, the relative importance of one solution over another is expressed using the eigenvector. The collected data regarding the matcher alternatives from the questionnaire are stored in a matching questionnaire database, while an additional database (AHP database) stores the calculation results (cf. Fig. 6.7 and 6.9, respectively).

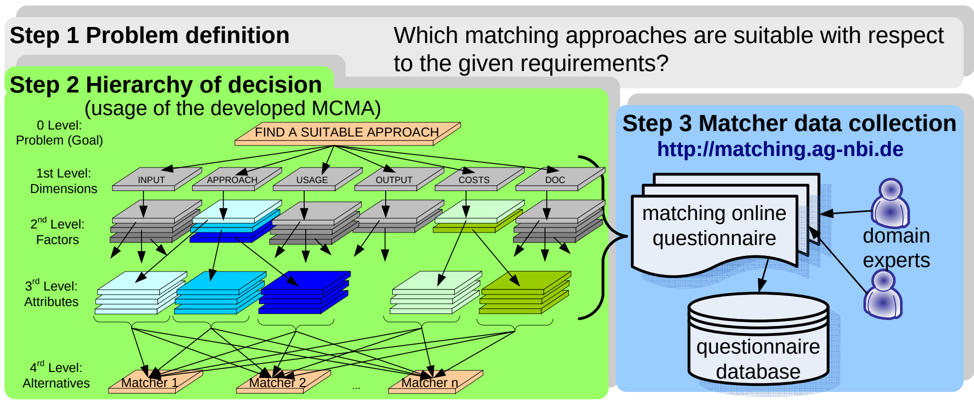


Figure 6.7: Data collection

Example
Assuming that:

- PRIOR can handle *large input size* twice as well as OCM,
- PRIOR can handle *tree structure* three times better than OCM, and
- OCM can deal with *instances* four times better than PRIOR,

the preferences (weightings of the alternatives) can be translated into the following evaluation matrix which, in turn, results in the eigenvector shown in Fig. 6.8.

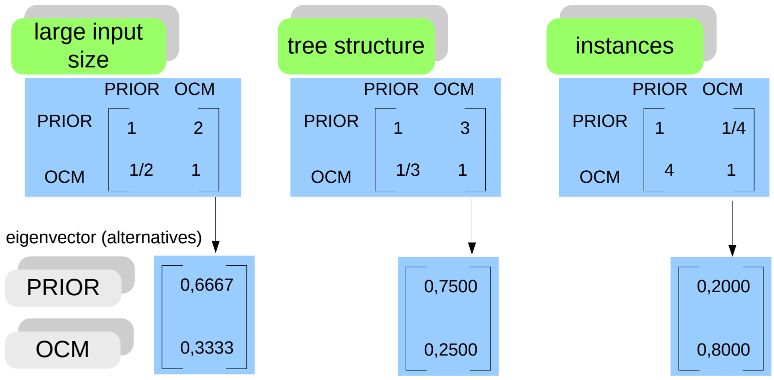


Figure 6.8: Dummy data collection

6.3.4 Building a Pairwise Comparison

In the next step of the matcher selection process, the requirements concerning the potential matching approaches with respect to the task or the application in which the approach is to be utilized must be compared and weighted (cf. Fig. 6.9). This means, for each level of criteria (dimension, factors, and attributes from the MCMA as described in Sec. 6.3.2), pairwise comparisons between the sibling nodes are to be made and organized into a square matrix. To facilitate these pairwise comparisons of the decision criteria based on our MCMA, we have developed a supporting user-friendly tool – as a part of the MOMA Framework – which, at the end, supports the entire process of the manual matcher selection[10].

Since we are acting on the assumption that people who decide on the matcher adoption for a particular application or task have already analyzed the situation given and identified the specific requirements that must be fulfilled by the matching approach, such people are also capable of evaluating and weighting their system specifications with the use of our tool. For each level of criteria, the user builds a pairwise comparison between the sibling nodes: the user weights attributes against attributes, factors against factors, and dimensions against dimensions.

[10]The preliminary basis that gave us initial ideas for the development of our AHP tool was the JAHP tool; `http://www2.lifl.fr/~morge/software/JAHP.html` accessed on 05.05.2007

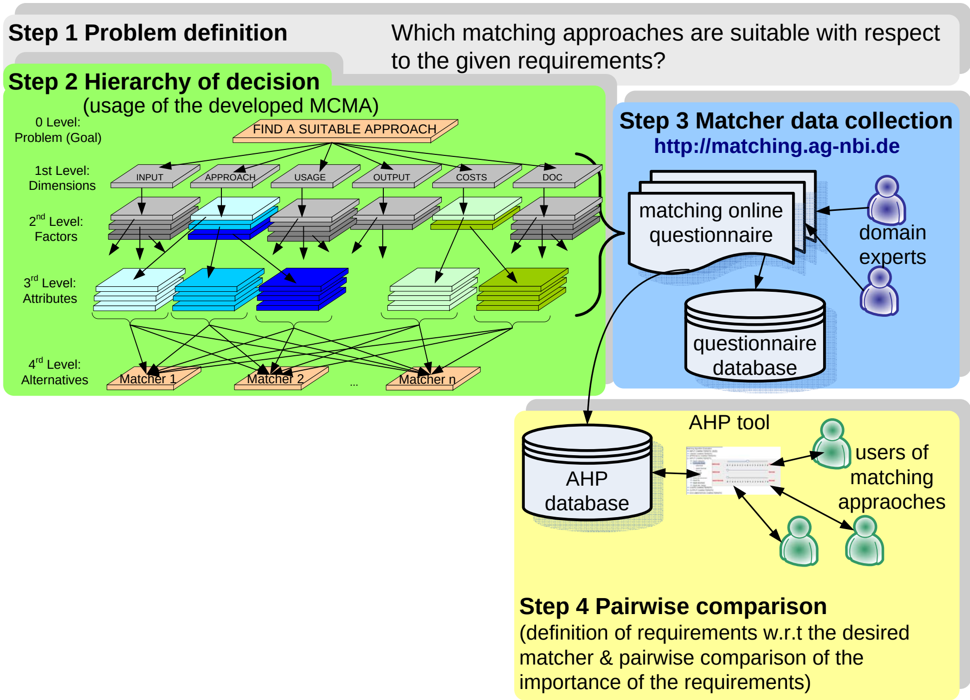


Figure 6.9: Pairwise comparison

Example

Coming back to our simplified example, we have built a pairwise comparison for the criteria *large input size*, *tree structure* and *instances*, according our dummy requirements as follows:

- *Large input size* is “slightly” more important for our dummy application than *tree structure* (AHP scale: 2, cf. Tab. 6.2).
- In our dummy application, we slightly favor the ability to deal with *tree structure* over the ability to handle *instances* (AHP scale: 3, cf. Tab. 6.2).
- In our dummy application, we “moderate(ly) plus” favor handling *large input size* over dealing with *instances* (AHP scale: 4, cf. Tab. 6.2).

We translated these requirements that had been defined using natural language into numbers using the AHP-scale (cf. Tab. 6.2). In doing so, we obtained an evaluation matrix with ratings of the criteria used to calculate the eigenvector that expresses the importance of each weighted criteria with regard to the dummy application requirements (cf. Fig. 6.10).

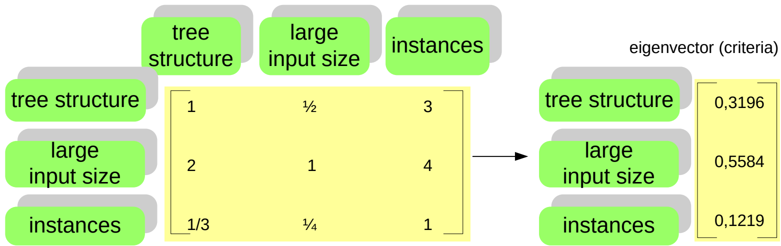


Figure 6.10: Dummy criteria comparison

6.3.5 Calculation of the Final Result

The decision regarding the determination of a suitable matching approach defined in step #1 is based on the ranking of matcher alternatives. The ranking reflects the overall importance of the approach according to the alternative weightings (collecting matcher metadata) performed in step #3, (cf. Sec 6.3.3) as well as criteria weightings regarding the application or tasks requirements (pairwise criteria comparison) done in step #4 (cf. Sec. 6.3.4).

As mentioned above, the tool developed in the MOMA Framework facilitates not only the pairwise comparisons of the criteria, but also by calculating the final result of the decision making process, delivers the ranked list of the matchers considered (cf. Fig. 6.11).

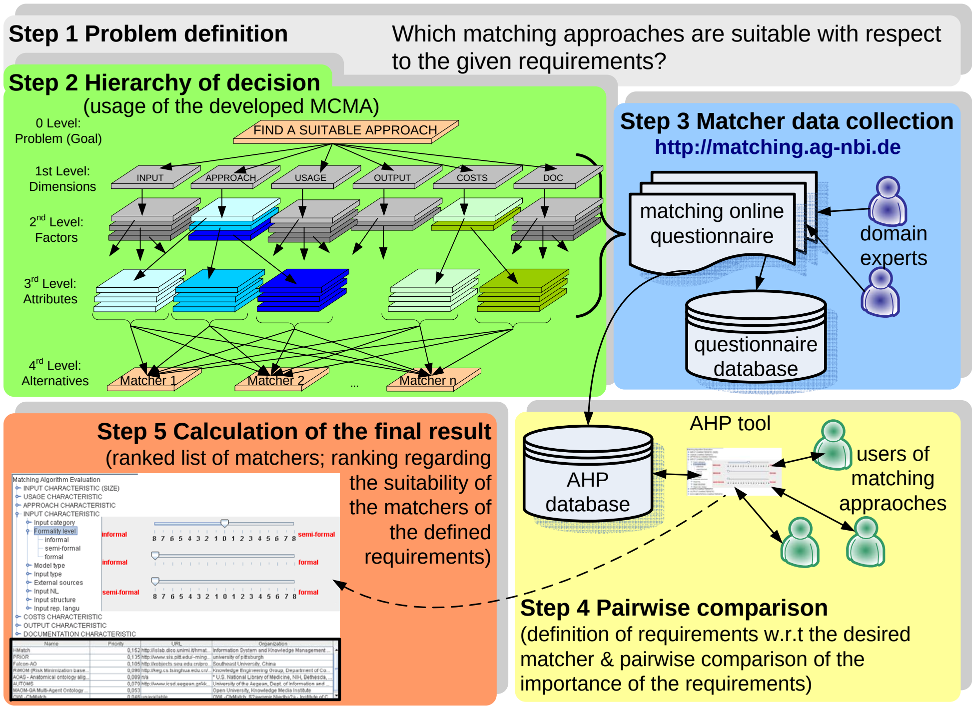


Figure 6.11: Final result

Example
Let us return to our simplified matcher case: We multiplied the ratings of PRIOR and OCM matchers with the weightings of criteria *large input size*, *tree structure*, and *instances*, resulting in the overall ratings of the matcher alternatives. Regarding the given requirements of the perfect (according to the dummy requirements) matcher (cf. Sec. 6.3.4), the matcher that best satisfies our "idea of a perfect approach" in the dummy application is PRIOR (cf. Fig. 6.12).

6.4 Final remarks on AHP-based approach

To allow the manual selection of matching approaches as envisioned in our MOMA Framework (cf. Chapter 4) and to serve the human matcher users, as described in Sec. 4.2.2, we have adopted the Analytic Hierarchy Process (AHP) which uses pairwise comparisons along with a semantic and ratio scale to assess the decision

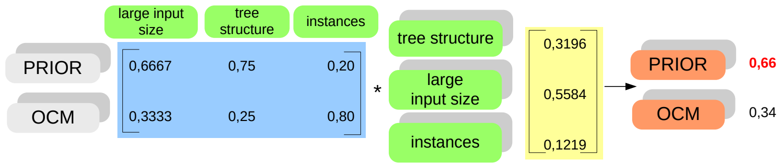


Figure 6.12: Calculation of the final result for dummy application

maker's preferences. The criteria hierarchy used for the matcher selection is based on our MCMA - Multilevel Characteristic of Matching Approaches (cf. Chapter 5), which covers features of particular algorithms along with appropriate incoming sources. In order to further facilitate the complex matcher selection, the process is supported by the tool, which is a part of the MOMA Framework and which delivers a list of matchers ordered according to their suitability. Furthermore, since the AHP method allows the decision makers to analyze results from various perspectives, the decision makers, who decide on the utilization of a particular matching approach in the concrete application, can play what-if scenarios by changing the weights of different attributes, factors, and dimensions. Even when more people participate in the decision process and their performance yields different courses of action, they can work together to reach a consensus on the judgments [197]. Another aspect that has not been addressed is the issue of definition and elimination of irrelevant matcher alternatives. Since we use for the matcher selection the relative measurement of the AHP approach, there is no need, as stated in [164], to improvise notions of relevant and irrelevant alternatives, as is done in utility theory, since, by definition of relative measurement, everything being compared is relevant. At this point, it is also important to add that, due to the fact that previous AHP users in other domains found pairwise comparisons to be an effective and intuitive way to elicit qualitative data, this method has not only gained acceptance in research but in commercial use as well. We hope these findings will also apply to matcher selection.

To verify the accuracy of the predictions of the proposed matcher selection, we have evaluated the manual part of the MOMA Framework in accordance with two real world case studies defined by the Ontology Alignment Evaluation Initiative (OAEI) 2006 Campaign (cf. Sec. 3.1, 5.3). The elaboration on the evaluation method applied, discussion on the results achieved by our approach, and their comparison with the matcher ranking obtained during real processing are described in Sec. 8.4.

Chapter 7

Rule-based Appoach

Abstract While in the preliminary chapter, we described the manual matcher selection based on the decision making approach – AHP, here we concentrate on the (semi-)automatic Metadata-based Ontology Matching (MOMA) Framework, which is based on the rules- and metadata-oriented methodology, and as its manual counterpart, considers existing matching algorithms. The framework uses, in particular, additional data w.r.t. different matches and, utilizing the defined rule set, builds a relation between the matchers (their characteristics) and the given ontologies to exploit the advantages of each of these approaches. Thus, in this chapter we elaborate the main idea of this rule-based approach and detail the high-level architecture of this part of the framework (cf. Fig. 7.1) along with its functionality; we describe the creation and usage of metadata for matchers and ontologies together with the application of the rules in order to determine which matching algorithms are appropriate for individual cases.

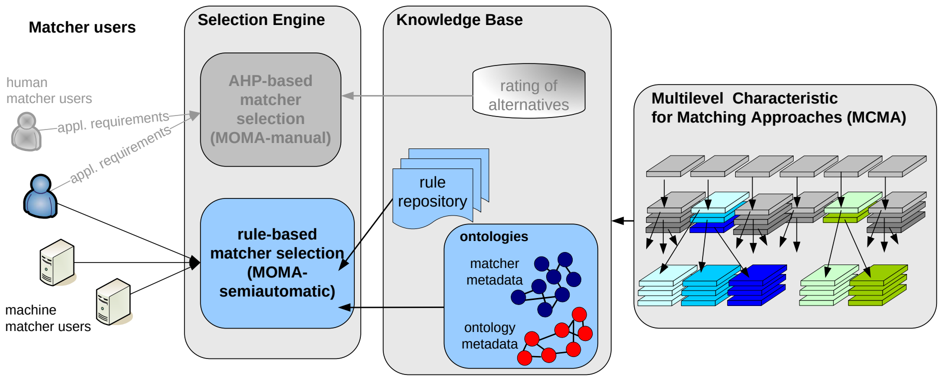


Figure 7.1: Rule-based MOMA Framework

7.1 Rule-based Matcher Selection

The Semantic Web envisions a Web of machine-understandable information which may be automatically accessed by and exchanged among semantics-aware applications. One of the prerequisites for the realization of this revolutionary concept are matching techniques capable of handling the open, dynamic, and heterogeneous nature of Semantic Web information with ease. Since matching methods are acknowledged as a core enabling technology, for example, in the context of mediating Web Service interactions, the selection of the most appropriate matching (service) is envisioned to be performed (semi-)automatically. As a possible solution for the (semi-)automatic matcher selection issue, we propose a methodology based on a Knowledge Base containing metadata (defined in the form of ontologies) and rules, and the execution of the latter. The methodology has been utilized in the semi-automatic Metadata-based Ontology Matching (MOMA) Framework (cf. Sec 4). Given a set of ontologies to be matched, the framework based on a reuse-paradigm takes into account the capabilities of existing matching algorithms and suggests appropriate matchers for a given application. Following the above mentioned goal, the objective of our MOMA Framework is to support both human matcher users, like ontology engineers and semantic web-based application developers (as elaborated in Chapter 6), and machine matcher users, such as service/matching providers or search engines. Due to its generic and (semi-)automatic character, the approach could be applied in a service-oriented context, so as to facilitate the discovering and deployment of appropriate matching services required to deal with specific (previously unknown) ontologies. Furthermore, it means that our framework offers a method and supplies tool to support not only the manual but also (semi-)automatic decision-making process regarding the applicable matchers.

The (semi-)automatic MOMA Framework uses additional information (metadata) regarding ontologies, particularly the structural data and the contextual information with which ontologies have been developed and used, and metadata regarding different matchings. To determine automatically which algorithms suit the concrete inputs, we have accepted the dependencies between existing algorithms and their input sources and extrapolated them to identify the most suitable matching algorithms for a given set of specific ontology inputs and application context (the matching algorithms and their characteristics are placed in relation with the ontologies). Since explicit knowledge concerning the dependencies between the algorithms and, for instance, the structures on which they operate is needed, we have formalized this knowledge in terms of dependency rule-statements that determine which elements, i.e. which matchers, are to be used. Furthermore, to express a core set of rules that is applied to detect algorithms suitable for processing a pre-defined ontological input, our framework resorts to semantically annotated metadata on ontologies and matchers. Thus, our matching framework allows different ontologies to be analyzed and matching algorithms to be checked to determine whether they are appropriate (considering given requirements) for the problem to be solved.

In the course of this chapter, we outline the main components of the (semi-)automatic MOMA Framework, based on the architectural structure. Then we detail the different metadata types together with the set of pre-defined rules on which the approach relies, and conclude the chapter with the description of the execution of the matcher selection process.

7.2 Architecture

Matching users consult the MOMA Framework in order to obtain matching algorithms required for a particular application. This, as detailed in Sec.4.2.2, applies to both humans, e.g. ontology engineers looking for means to compare similar ontological sources, and Web Services seeking automatized methods to, for instance, generate mediation ontologies. In order to (semi-)automatically decide which algorithms suit which particular ontological input, information regarding the approach and the input sources must be brought together. For this reason, the proposed framework uses (semantic) descriptions of both single matching algorithms and Web ontologies, which are then related by means of rules to optimize matching results. The core of the MOMA Framework shown in Fig. 7.2 consists of:

Ontologies & Matchers that represent existing ontologies, be they local (private) ontologies or those available on the Web (public), and matchers available in the system.

- **ontology repository** The framework does not distinguish between local ontologies and ontologies available on the Web, since, after the specification of the source location, both are handled in the same manner. As mentioned before, the framework needs a description (metadata) of the ontological sources; however, since the metadata is unfortunately not provided along with the ontologies, we have decided that our framework takes over this issue and automatically generates the necessary description using the information from the particular ontology and additional sources.[1]

- **matcher repository** The framework needs not only ontological sources, which are to be matched, but also matching approaches, which will be evaluated

[1] The more detailed description of the generation process can be found in Sec. 7.4.

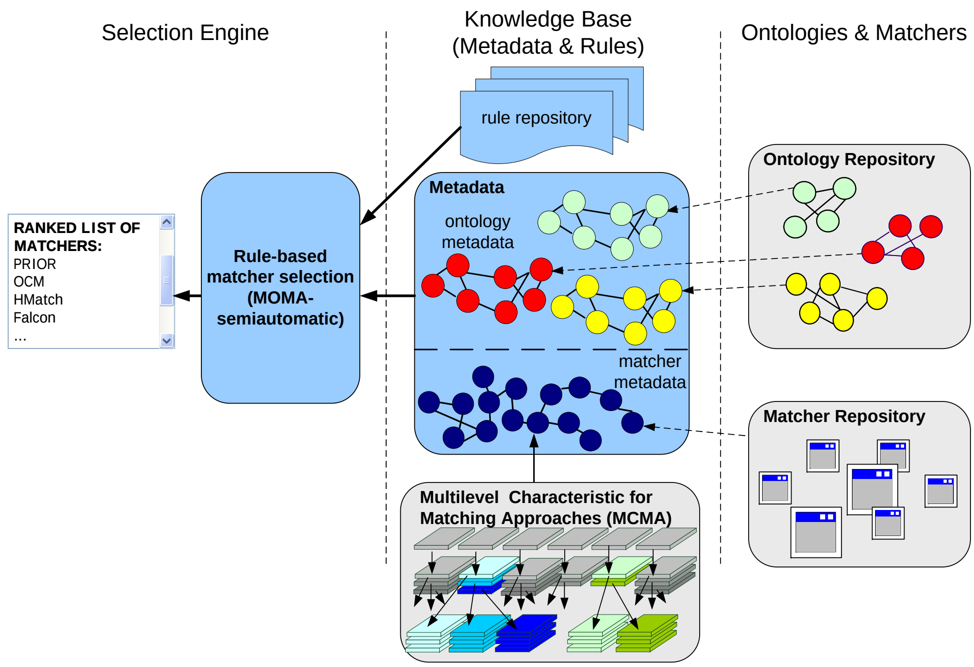


Figure 7.2: Architecture of the (semi-)automatic MOMA Framework

for their suitability w.r.t the given ontologies. However, since we do not execute the matching approaches, we do not need to collect the particular algorithms but instead just their descriptions and characteristics. For this reason, what we call "matcher repository" is simply a kind of overview of matchers that provides the required data (matcher metadata[2]) to the system.

Knowledge Base (KB) - In our opinion, 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 (together with input, output, costs, etc.) and targeted applications, 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 final decision regard-

[2]The more detailed description regarding the matcher metadata and its extraction can be found in Sec. 7.4.

ing suitability. This development process, described in Chapter 5, resulted in our **Multilevel Characteristic for Matching Approaches (MCMA)**. Furthermore, KB as the content of a particular domain or field of knowledge includes, in our case, 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. However, since the ontology metadata is not provided along with the ontological sources, our framework comes into play and automatically generates the description using the information from the particular ontology and additional sources. Furthermore, as we are unable (at this time) to generate the matcher metadata automatically, our framework considers only matchers that are available in the system by being manually annotated. As this collection of knowledge is to be expressed using some formal knowledge representation language, both matcher and ontology metadata are expressed in the form of ontologies. The KB is completed by a predefined set of rule statements stored in the *rule repository* that defines the dependencies between the matching approaches and ontological sources.

Selection Engine, the core of the MOMA Framework, is responsible for the decision making process regarding the selection of algorithms applicable for a specific set of input. The formal descriptions of matchers and incoming ontological sources to be matched allow the selection engine to automatically compare the metadata of the inputs with the constraints of the available algorithms and, by means of generic rules, detect the most suitable algorithms that can best deal with the properties of a particular set of ontologies (i.e. the decision process is conducted by means of ontology and matching metadata as well as rules from the rule repository).

7.3 Metadata & Rules

The term *metadata* has become common with the popularity of the World Wide Web, and its necessity is even more evident, especially when large amounts of information make it difficult to search and retrieve information. Today there is a huge range of metadata definitions that can be divided, for example, into application-oriented definition, definition based on the architecture of metadata, and data-oriented definition [180]. Considering the latter, metadata, literally "data about data", is an increasingly ubiquitous term that is understood in different ways by diverse professional communities who design, create, describe, preserve, and use information systems and resources [68]. However, metadata also describes the content, quality, condition, and other characteristics of data, which, in turn, means that metadata can relieve data users of having to have full advance knowledge of a dataset's existence and characteristics. In particular, metadata defines an information resource, increases its accessibility, and provides other useful resource information helpful for its management [180]. Since metadata is the information necessary to create order within the information, providing description, classification, and organization, and as it is a systematic method for describing resources, thereby improving their access, our MOMA Framework utilizes metadata within the determination process of suitable matchers. In this context, the MOMA Framework uses additional information about the ontologies, i.e. ontology metadata, as well as available matching approaches, i.e. matcher metadata, to determine which of the latter are appropriate in a given application context. Ontology metadata captures information about matching-relevant ontology features such as implementation language, size of the model, or language used for labeling ontological primitives. In turn, matching metadata describes the most important characteristics of the matching approaches, such as input and output parameters, applied heuristics, cardinality, etc. To ensure a rich and at the same time formal representation of the ambiguous semantics of the metadata and to enable its integration and exchange in and between Semantic Web applications, we have modelled this information in ontological form and implemented it using Semantic Web representation languages, i.e. metadata is stored declaratively in repositories and offers an ontological description of the most significant properties of the corresponding items (i.e. matching algorithms and their inputs). These MOMA-ontologies have been developed in accordance with established ontology engineering methodologies [61], while empirical findings acquired in case studies [16, 17, 135, 143] are the main input for the elaboration of requirements underlying metadata schemes.

7.3.1 Matcher Metadata Model

Since suitability of the given matching approaches is determined with careful consideration to a number of factors relevant to the selection process, the matcher metadata is intended to capture information about existing ontology matchers. In order to specify the contents of the target metadata model, we have analyzed matcher features that have been empirically proven to have an impact on the quality of matching tasks. However, in regard to the ontology building process, the ontology developer must consider whether to build the target ontology reusing existing sources or to build an ontology from scratch. Reuse of existing sources within ontology engineering is recommended by default in current methodologies and guidelines as a key factor for developing cost-effective and high quality ontologies [152, 193]. Furthermore, the benefits of source reuse go beyond the common cost saving and interoperability usually mentioned in related engineering disciplines. Since ontologies are understood as a means for shared knowledge conceptualization, reusing existing sources increases application interoperability both on the syntactic and semantic levels. Humans or software using the same (ontological) sources assumably hold the same view of the modelled universe of discourse and thus define and use domain concepts in the same way. As ontology reuse typically starts with the identification of knowledge sources useful for the application domain, we have returned to the Multilevel Characteristic for Matching approaches (MCMA) described in Chapter 5. Since MCMA was developed in a systematic process by the utilization of:

- the analysis of the requirements collected during the development of various Semantic Web-based applications;
- the analysis of the literature;
- an intensive and systematic collaboration with experts in the ontology matching domain, ontology and software engineers; and
- our findings gleaned from various ontology engineering case studies,

it was a matter of course to consider this characteristic as the basis for our matcher metadata model[3]. In order to collect data regarding existing matchers that will be considered within the selection process, we have developed an online questionnaire based on the MCMA described in Section 5.3. The survey allows us to analyze existing matchers, collect data relevant to the selection process of matching approaches and deliver the weighting of the matchers w.r.t. the particular features.

The matcher metadata model has been conceptualized in the form of an ontology and implemented in OWL, in which the MCMA dimensions and factors have been mapped into the target ontology as concepts and the attributes as ontological properties (cf. Fig. 7.3). `OntologyMatcher`, as the main concept in the matcher metadata, represents the matching approaches and is defined (`hasCharacteristic`) by matcher characteristic (`MatcherCharacteristic`). The (general) matcher characteristic is defined by the different characteristics – the MCMA dimensions – related to it in the form of a hierarchical structure with "sub-class" relationships (`Input_Characteristic`, `Output_Characteristic`, `Approach_Characteristic`, etc.). The further sub-classes of the matcher characteristics (e.g. `How_Well_Does_The_Approach_Support_Cardinality` in the `Output_Characteristic`) and their properties stem directly from the questionnaire mentioned earlier and map the

[3]The matcher metadata model is available at `http://moma.ag-nbi.de/matcher_metadata.owl` accessed on 10.08.2008

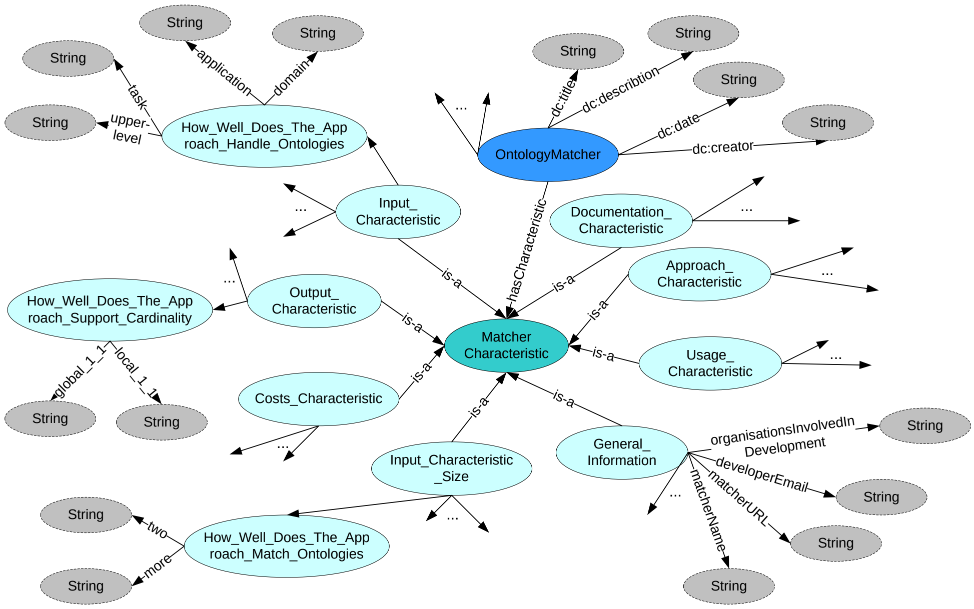


Figure 7.3: Matcher Metadata (fragment)

questions and possible answers. Apart from the concepts and properties that are relevant for the matcher selection process, the ontology also contains some general information regarding the approach itself (`matcherName`, `matcherURL`), its developers (`developerEmail`), and the institutions (`organisationsInvolvedInDevelopment`) involved in the development of the particular matcher.

7.3.2 Ontology Metadata Model

Not only the matching approaches but also the matcher inputs, i.e. ontologies to be matched, need to be described in the corresponding metadata (cf. Fig. 7.4). For this purpose, following the ontology reuse paradigm as a key factor for the developing of the cost effective ontologies, we have utilized for our purposes the information model described in [14, 15, 80], which is a contextual model for Semantic Web resources[4] incorporating intrinsic and extrinsic ontology properties, some of which are recognized as influencing the matcher selection.

[4]The ontology metadata model is available at `http://swpatho.ag-nbi.de/context/meta.owl`; accessed on 10.08.2008

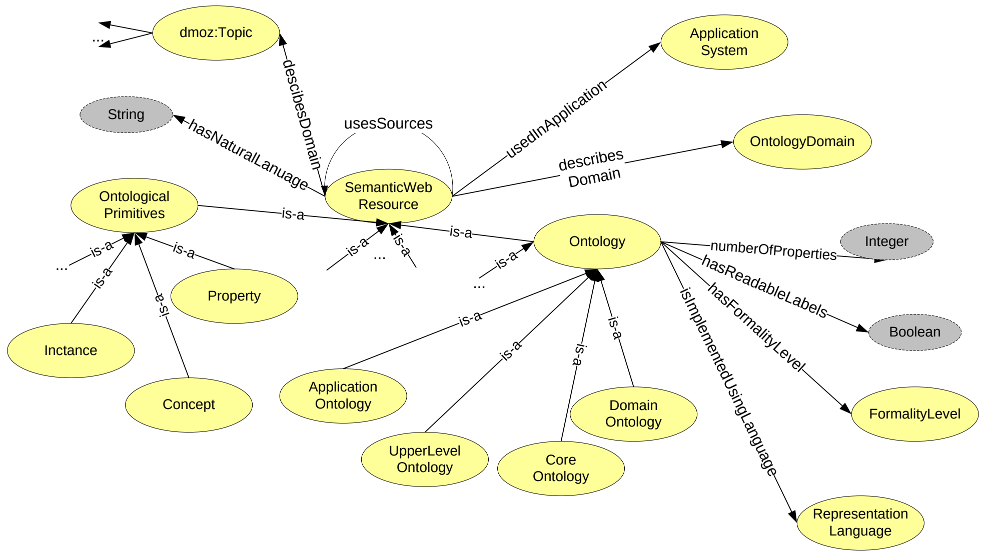


Figure 7.4: Ontology Metadata (fragment)

Accounting for the fact that matching algorithms cannot be applied with the same success expectations regardless of the dimensions of the ontology metadata model, and in order to identify the features relevant to the matching issue, we have sketched out in the following the individual feature categories, which are classified from a content-oriented perspective into syntactical, semantic, and pragmatic characteristics:

Syntactic features offer quantitative and qualitative information about the ontology and its underlying (graph) topology. The quantitative characteristics is composed of, for example, the depth of an inheritance tree, number of incoming properties, the number of concept instances, average path length, number of connected components, while the qualitative features contain representation language-dependent information like the representation language itself, the number of syntax constructs used, and syntactical correctness [14]. Within the model, the syntactic features are conceptualized as DatatypeProperties with an integer or a string range for the numerical and qualitative information, respectively [14, 148]. The quantitative syntactic features, such as the number of specific ontological primitives, have a crucial influence on the matcher selection process since they affect the matching execution performance and quality. However, this also includes the features concerning the representation and natural language of the ontological primitives, since the matchers can usually handle only specific natural or representation languages.

Semantic features are related, according to [14, 148], to the formal semantics of the representation language and the meaning of the ontological content: consistency (as measured by a reasoner), correctness (i.e. whether the asserted information is true), readability (i.e. the non-ambiguous interpretation of the meaning of the concept names w.r.t. a lexicon, the usage of human-readable concept names), formality level (e.g. highly informal, semi-informal, semi-formal, rigorously formal), model type (upper-level, domain ontology, thesaurus, etc.), ontology domain (e.g. human resource), and representation paradigm (i.e. the class of representation languages w.r.t. expressivity, such as a specific Description Logic). The semantic features are modeled as OWL classes or individuals. For interoperability purposes, every instance of `OntologyDomain` references a topic in the Open Directory taxonomy[5], which was translated to OWL for this purpose. The class `DomainType` is used to define the generality levels of a conceptualization as stated in [75, 201]. By means of the class `RepresentationParadigm` and its individuals (e.g. some Description Logic), one can define which ontological primitives are supported in the representation language of the matching input (e.g. supports existential constraints) and which are actually used in the respective model (e.g. the ontology is written in OWL DL, but it uses solely classes and sub-class relationships). General purpose `FormalityLevels` are defined as in [191, 201]. The properties of different kinds of ontologies (such as thesauri, taxonomies and Semantic Web ontologies) can be declared by means of OWL constraints. All these semantic features play a crucial role in the context of matcher selection, since they all affect the suitability of approaches and, in turn, matcher execution:

- As some matchers are incapable of treating instances, it is important to know whether the incoming sources include instances.
- Since some approaches support only specific representation or natural language, the features concerning representation and natural languages of the primitives are important for the selection process.
- Features such as formality level, model type, or the information whether it is a domain- or task-specific ontology restrict the number of potential applicable matching algorithms, since the matcher may be capable

[5]`http://www.dmoz.org` accessed on 15.06.2008

of handling only a particular formality level or have been specifically developed to deal with a certain domain/task.

Pragmatic features refer to authoring and historical data of an ontology, for example, when, by whom, and for what purpose it was developed, whether multiple versions are available, or about the engineering process from which the ontology originated [14, 148]. Such features are not crucial characteristics to matcher selection; however, in future development, we could foresee analyzing, for example, versions or engineering process issues.

7.3.3 Rule repository

For a given set (pair) of ontologies to be matched, the selection engine must decide which matching algorithms are to be applied to satisfy requirements and obtain desired outputs. The engine is aware of background information that details available matching approaches and properties of the input ontologies. However, in order to automatically infer which algorithms suit the concrete inputs, it needs explicit knowledge concerning the dependencies between these algorithms and the incoming sources on which they successfully operate. This knowledge, which has been gained during the analysis of different test cases and Semantic Web-based applications and the discussions with experts in ontology matching domain, has been formalized in terms of *dependency rule-statements* that determine which elements, i.e. which matchers, fulfill the rule-conditions. The set of rule statements has been divided into mandatory and selection rules:

mandatory rules decide whether ontologies are at all 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 to find suitable approaches for the ontological input given (some examples):

- **concept names:** Do not apply linguistic matchers to ontologies with incompatible concept names.
- **no instances:** If ontologies have no instance data, only matchers capable of dealing with scheme can be applied.
- **instances:** If ontologies consist of instances, only matchers able to deal with instance data 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 incoming sources are formalized in OWL, only matchers that can handle OWL sources may be applied).
- **natural language:** Consider only matchers that are able to deal with the natural language of the incoming sources (e.g. if the concepts of incoming sources are defined in English, only matchers that can handle English sources can be applied).
- **formality level** Apply only matchers that can deal with the formality level of the incoming sources (e.g. if we are considering formal ontologies, only matchers that can handle formal ontological sources can be applied).
- **input size - concepts:** Consider only matchers that are able to deal with the given number of concepts within the incoming sources (e.g. if the input sources have over 1.000 concepts each, suitable matchers are those that can match this (middle-size) number of concepts).
- **input size - properties:** Consider only matchers capable of dealing with the particular number of properties (e.g. if the input has over 100 properties each, we need an approach that can match at least this number of properties).
- **input size - instances:** Consider only matchers that can deal with the given number of instances (if the input has over 1.000 instances each, appropriate matching algorithms are those that can deal with a large number of instances).
- **input size - axioms:** If the incoming sources consist of axioms, only matchers capable of handling axioms may be applied.

The rules established are the result of collaborations with experts and an analysis of recent publications of ontology matching domain. As described in [13, 17, 134] they have been confirmed empirically by projects such as "KnowledgeNets" and "A Semantic Web for Pathology" (cf. Sec. 5.1.2), which required ontology matching techniques to merge and integrate existing ontologies into corresponding target ontologies used to build various Semantic Web applications.

For implementation purposes, both mandatory and selection rules have been implemented in SWRL[6] – a rule language for the Semantic Web (cf. Sec. 2.1) – that allows for the formalization of our rules in terms of the concepts defined in the two metadata models.

[6] `http://www.w3.org/Submission/SWRL/` accessed on 15.05.2008

7.4 Rule-based matcher selection process

After presenting the main idea of the rule-based selection approach and elaborating on the main components in the MOMA Framework, in this section we concentrate on metadata generation and functionality of the MOMA Selection Engine, which is the heart of the framework responsible for matcher evaluation and ranking process. For a set of ontologies to be matched, the Selection Engine must decide which matching algorithms are applicable to obtain the desired outputs. The engine is aware of the background information detailing the available matchers (in the form of matcher metadata) and the properties of the input ontologies (ontology metadata). In order to automatically infer which algorithms suit certain inputs, the engine needs explicit knowledge (rules) regarding the dependencies between these algorithms and the structures on which they operate.

7.4.1 Metadata extraction and generation

As mentioned before, since it is impossible (at this time) to automatically generate metadata for a particular matcher, the approaches to be considered by the MOMA Framework must be annotated manually using the MCMA-based online survey (cf. Sec. 5.3). Having the information regarding the matching approaches collected by the online survey, the data relevant for the selection process is converted into instance data of the matcher metadata model (cf. Sec 7.3.1). Thus, for each matching approach that has been annotated by the online survey, a corresponding matcher metadata has been generated according the semantic matcher metadata model (cf. "matcher metadata extraction" in Fig. 7.5).

Moreover, to be able to evaluate the matching approaches, the framework needs not only the matcher metadata but also additional information regarding ontological sources. Therefore the ontologies to be matched are loaded into the system, whereby the framework automatically generates the ontology metadata utilizing the OntoMeta approach. OntoMeta [31] offers methods and heuristics for automatic extraction and calculation of the pre-defined syntactic and semantic information for an input in the form of an ontology and delivers a metadata entry structured according to the ontology metadata model (cf. Sec. 7.3.2); i.e. OntoMeta generates metadata for incoming ontological sources in a form compatible with the ontology context model (cf. "ontology metadata generation" in Fig. 7.5). Since the syntactic metadata captures information about the extrinsic properties of a Semantic Web ontology, its underlying graph structure, and the syntax of the implementation language, the features of this category can be derived programmatically for every ontology represented in a (semi-)formal representation language on the basis of information included within the ontological sources. In contrast, the semantic metadata that provides descriptive information about the (formal) meaning of the contents of the ontology must be acquired by means of special heuristics, which exploit well-known classifications such as the Open Directory and the Web through common search engines such as Google [31, 148].

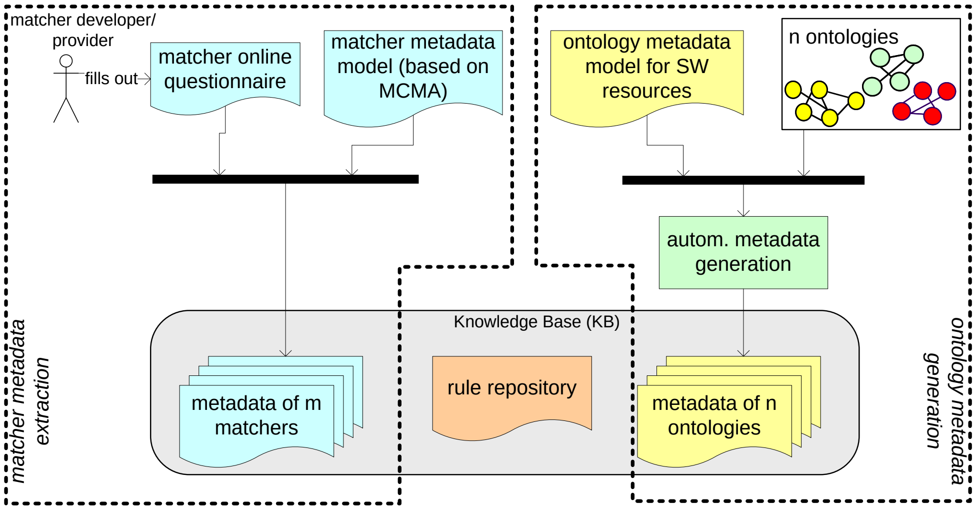


Figure 7.5: Metadata extraction and generation

7.4.2 Rule execution

After the extraction and generation of matcher and ontology metadata, respectively, as described in the latter section, the metadata is stored along with the SWRL selection rules in a repository (Sesame[7] repository). The MOMA Knowledge Base including metadata and rules is ready to be used in the matcher selection process and, in turn, the framework can create the queries (SeRQL[8] queries) on the repository. However, the usage of the rules in decision making processes requires a reasoning engine capable of operating on OWL ontologies and SWRL rules – an issue that is still the subject of research in the Semantic Web community. For this reason, we

[7]Sesame is an open source framework for storage, inferencing and querying of RDF data, `http://www.openrdf.org/`; accessed on 17.09.2008

[8]SeRQL query language, `http://www.openrdf.org/doc/sesame/users/ch06.html`; accessed on 18.09.2008

have integrated into the framework an engine supporting reasoning SWRL rules in RDF graphs – SWRL Engine[9]– and adopted it for our purposes by augmenting the OWL support. Finally, having both matcher and ontology metadata together with a rule set in Knowledge Base and the extended SWRL Engine, the entire framework is ready for the crucial step – rule execution (cf. Fig. 7.6) – which (when rule body = true) links two ontologies (in the case of mandatory rules) or an ontology with a matcher (in the case of selection rules).

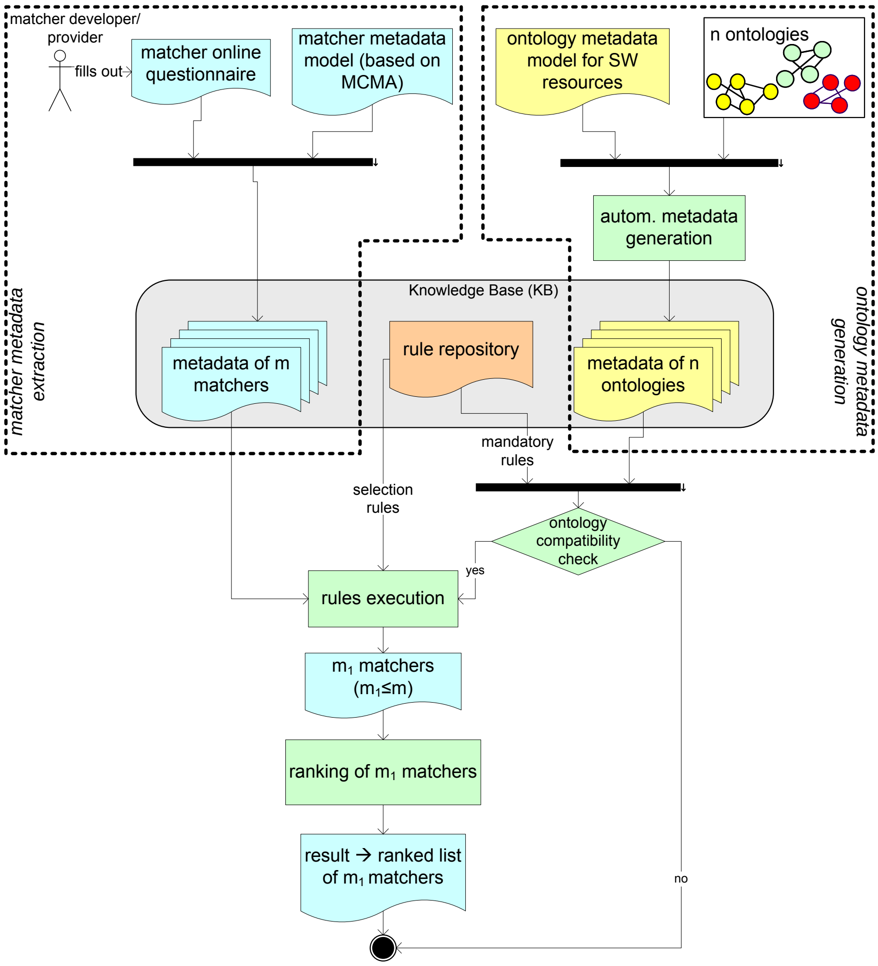


Figure 7.6: Rule-based matcher selection (flow chart)

[9]http://www.ag-nbi.de/research/swrlengine/ accessed on 15.05.2008

By inserting triples in the form:

```
matcher meta:isCompatibleAccordingSmallInstanceNumber ontology
```

we are able to create such links on the ontological level. The rule execution starts with the step in which each rule is converted into a SeRQL query to select all the variables defined in the rule's body and place the rule conditions into the `SELECT`'s `WHERE` part. Based on the query's result set, the framework inserts new triples, defined in the rule's head, into the Sesame repository.

As mentioned before, we distinguish between mandatory and selection rules(cf. Sec. 7.3.3), whereby the former rules decide whether ontologies are at all suitable for being matched:

Ex. 1: SWRL rule tests whether the ontologies to be matched use the same natural language:

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

while the latter rules evaluate the matcher's suitability; for ontologies that, according to our mandatory rules, are compatible for the matching process, the framework queries the Sesame repository for matchers that can handle these sources. In particular, since each property created by the SWRL engine between a matcher and the ontologies means that the matcher conforms to this particular condition, the framework allocates the weighting (from the survey data using a predefined scale from 0 to 8, with 8 being the most optimal) to the analyzed matcher, thus increasing the suitability level of the particular approach.

Ex. 2: SWRL rule tests whether a matcher can handle formal ontologies

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

In the course of the matcher selection process, the number of suitable matchers (m_1) may be reduced in comparison to the preliminary candidates number ($m_1 \leq m$). In the next step, the framework ranks the remaining m_1 matchers 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 (cf. Fig. 7.6).

7.5 Final remarks on the rule-based approach

In this chapter, we have described a (semi-) automatic MOMA Framework that treats some of the matcher limitations by being aware of the link between matching algorithms and the ontologies (or the types of ontologies) for which they were originally designed or to which they can be successfully applied; the framework uses semantic descriptions of both single matching algorithms and Web ontologies, which are then linked by means of rules to optimize matching results. We have presented the high-level architecture and elaborated the main components of the framework: Knowledge Base that contains matching metadata based on a matching metadata model defining the properties of matchers, the ontology metadata generated by the OntoMeta tool according to the ontology metadata model, and the rule repository with mandatory and selection rule linking ontology and matching properties, and the Selection Engine, which uses the reasoner to evaluate and rank the matchers regarding their suitability for particular ontological sources. The proposed (semi-)automatic MOMA Framework allows for a more flexible, (semi-)automatically triggered selection of various matching algorithms, depending on their suitability to the ontology management process. Due to its generic character, the approach can be applied in a service-oriented context to facilitate the discovery and operation of appropriate matching services required to deal with specific (previously unknown) ontologies.

As in the case of the manual, AHP-based MOMA Framework, we have evaluated the rule-based part of framework in the context of real world case studies defined by the Ontology Alignment Evaluation Initiative (OAEI) 2006 Campaign (cf. Sec. 3.1). We detail the evaluation method applied along with the results achieved in Sec. 8.5.

Chapter 8

Evaluation

Abstract To determine the worth and significance of something and to verify systematically whether something is achieving its goals, it needs to be evaluated. Thus, in the course of this chapter, we first give a brief overview of existing evaluation models before deciding on the method with which to evaluate the MOMA Framework (cf. Fig 8.1). The evaluation process starts with the expert-based evaluation of the Multilevel Characteristic for Matching Approaches (MCMA), which results in refinement of the preliminarily defined characteristic and ends in a revised MCMA, which has been used within both the AHP- and rule-based approaches for matcher selection. Because the next evaluation step has been dedicated to accuracy verification of the predictions of both proposed selection methodologies, we report on the case study evaluation of both methods in accordance with two test cases defined by the Ontology Alignment Evaluation Initiative (OAEI) 2006 Campaign.

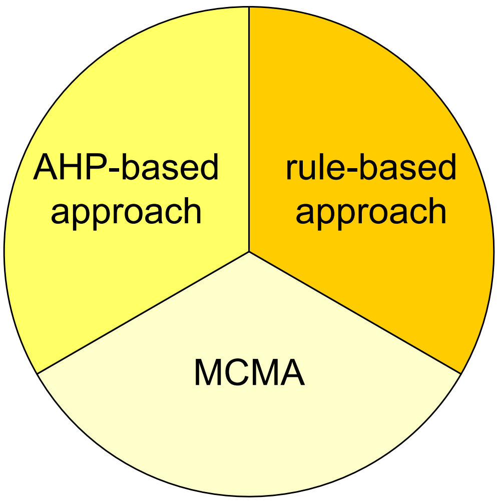


Figure 8.1: Parts of the MOMA Framework to be evaluated

8.1 Overview of evaluation methods

In order to systematically determine the value and significance of the research work w.r.t the attainment of the goals defined, the work needs to be evaluated. **Evaluation** is the analysis and comparison of the progress vs. original plans, and is oriented toward improving plans for future development. In other words, evaluation helps to document whether something is achieving its goals, and identifies the strengths and weaknesses of parts that need to be modified and improved. Since evaluations are used in a large number of fields and in many different contexts, and concern a very wide range of issues regarding when and how interventions or treatments work [154], there are also many different perspectives, approaches, forms, and models of evaluation, which are variously classified, depending on the respective criteria. Before the appropriate evaluation method can be chosen, we need to understand the underlying models [59]. **Evaluation models**[1] are an important part of the evaluation issue, since they provide a guide or heuristic for thinking about how an evaluation can be conducted [20]. There are two main classifications of evaluation models: The Stufflebeam and Webster taxonomy [175], which classifies approaches according to their orientation toward the role of values, and the House classification [93], which concentrates on freedom of choice, uniqueness of the individual, and empirical inquiry grounded in objectivity. In our research, we concentrate on the latter, which distinguishes between eight general model types (cf. Tab. 8.1):

- **systems analysis**, where the evaluator assumes quantitative output measures (e.g. test scores);
- **behavioral objectives**, in which the objectives of a system are expressed in terms of user performances/behavior;
- **decision making** that is structured by the decision to be made;
- **goal free**, which is an approach without goals (the evaluator makes no final decision);
- **art criticism**, wherein someone who is attuned from experience and training judges the important aspects based on the basis of his/her expertise;
- **professional interview** (accreditation), in which the reviewers (professionals/experts) are asked to judge or comment on the information collected;

[1]The evaluation model has been taken as the methodologies for conducting the actual evaluation, rather than as persuasions or frameworks in which more specific constructs and methods must be placed [20].

- **quasi-legal**(adversary) that takes the form of a trial by jury; and
- **case study** (transaction), which is an empirical inquiry that analyzes a phenomenon within its real-life context whereby the "stake" is the leading action.

Model	Major audience	Assumes consensus on	Methodology	Outcomes	Typical questions
system analysis	management, government, regulatory bodies	goals, known cause and effect, quantification, procedures and quantified variables	cost-benefit analysis	efficiency	Are the expected effects achieved? What are the most efficient programs?
decision making, operational research	decision makers, project management, administrators	criteria	general goals, surveys, interviews, questionnaires	effectiveness, quality control	Is it effective? Which parts are effective?
behavioral objectives	managers, psychologists	pre-specified objectives, quantified outcome, variables	(established) behavioral objectives, achievement tests	productivity, accountability	Is the system achieving the objectives? Is the system productive? Are people behaving appropriately for the system?
goal free	users, consumers	consequences, criteria	requires bias control, logical analysis, attention to modus operandi	consumer choice, social utility	What are the effects?
art criticism	connoisseurs, consumers	critics, standards	critical review	improved standards	Would a critic approve of this system?
professional review (accreditation)	professionals, public	criteria, panel procedures, panel composition	review by panel, self-study	professional acceptance	How would professionals rate this system?
quasi-legal (adversary)	jury	procedures, judges	quasi-legal procedures	resolution	What are the arguments for and against this system?
case study (transaction)	practitioners, users	negotiation activities	case studies, interviews, observations	understanding of diversity	What does the program look like to different people?

Table 8.1: House taxonomy of major evaluation models [93, 94]

The models mentioned above can be further subdivided into quantitative evaluation – the first three types of the House models – and into qualitative evaluation – the remaining models.

- **Qualitative evaluation** produces a description usually in non-numerical terms and uses qualitative and naturalistic methods, sometimes alone, but often in combination with quantitative data. Qualitative findings in evaluations provide people with a more "user-friendly" understanding of the circumstances than an analysis hidden behind numbers. Qualitative methods include three kinds of data collection:

- (in-depth) interviews - open-ended questions and probes yield in-depth responses about people's experiences, opinions, and knowledge; data consist of personal statements with sufficient context to be interpretable;
- direct observations - fieldwork descriptions of activities, behaviors, actions, conversations, interpersonal interactions, or any other aspect of observable human experience; data consist of richly detailed descriptions, including the context within which observations have been made;
- written documents - written material and other documents from organizational, clinical, or program records, memoranda and correspondence; official publications and reports; letters and written responses to open-ended surveys; data consist of excerpts from documents.

- **Quantitative evaluation** is linked to validity and threats to internal validity that determine efforts to assess the incremental outcomes of the system (program). It emphasizes hypotheses and questions that reflect a limited number of possible stakeholder perspectives [128], while the typical key evaluation question is whether the system has produced the outcomes observed, i.e. was the program effective? The more an evaluation emphasizes the control of antecedent and output conditions, the more it is concerned with conventional (typically quantitative) and less with qualitative evaluation approaches [149].

Both evaluation methods, quantitative and qualitative, have certain strengths and weaknesses (cf. Tab. 8.2); however, depending on the context and objects to be evaluated, the drawbacks of one approach can be outweighed by the advantages of another, and vice versa.

In the next sections, we initiate the decision regarding the selection of evaluation methods for each specific evaluation step and elaborate on their application to the main parts of the MOMA Framework.

8.2 Choosing a suitable evaluation

In order to identify the most appropriate method (or combination of methods) likely to yield fruitful answers, the evaluators need to define the aim and the role of the evaluation process, describe the expected results, and specify an important, thought

Qualitative evaluation is characterized by	**Quantitative evaluation is characterized by**
inductive approach to data gathering	research hypotheses and questions that are tested in the evaluation
holistic approach: finding gestalts to evaluate results	finding patterns that either corroborate or refute particular hypotheses and answer the evaluation questions
understanding the subjective live experiences for program stockholders (discovering their truths)	understanding how social reality, as observed by the evaluator, corroborates or refutes hypotheses and answers the evaluation questions
using natural language throughout the evaluation process	emphasis on measurement procedures that lend themselves to numerical representation of variables
in-depth, detailed data collection	representative samples of stockholder groups
use of case studies	use samples size with sufficient statistical power to detect expected outcomes
the evaluator as the primary measure	measuring instruments constructed with a view to making them reliable and valid
a naturalistic approach, one that does not explicitly manipulate the setting	control of evaluator and ability to manipulate the setting, which improves the internal validity, the statistical conclusion validity design

Table 8.2: Qualitative vs. quantitative evaluation [128]

provoking, and researchable issue to be resolved in the course of the evaluation. Since our work, as elaborated in Sec. 1.2.3, results in three different artifacts and as they have different needs, goals, and expectations regarding the evaluation (cf. Tab. 8.3), we must apply various evaluation methods. For this reason, we first initiate the utilization of a particular evaluation approach so as to elaborate the evaluation process in the following sections.

<table>
<tr><th>Artifact type</th><th>Our contributions</th><th>Evaluation goal</th></tr>
<tr><td>model</td><td>Multilevel Characteristic for Matching Approaches (MCMA) (cf. Chapter 5)</td><td>evaluation of the preliminary (a priori) characteristic in order to assure the quality and accuracy of the developed model</td></tr>
<tr><td rowspan="2">methods</td><td>AHP-based matcher selection (cf. Chapter 6)</td><td rowspan="4">the goal of the methods and instantiations evaluation, in our case, is directly related to their usage in real-world situations and to the accuracy of their predictions</td></tr>
<tr><td>rule-based matcher selection (cf. Chapter 7)</td></tr>
<tr><td rowspan="2">instantiations</td><td>AHP-based MOMA Framework (cf. Chapter 6)</td></tr>
<tr><td>rule-based MOMA Framework (cf. Chapter 7)</td></tr>
</table>

Table 8.3: Evaluation of our artifacts

8.2.1 Choosing the evaluation for MCMA

The main goal of developing our MCMA was the idea of having a matcher characteristic suitable for matcher comparison that, in turn, is to be applied to the selection of the appropriate (regarding given requirements) matching approaches. The preliminary version of the MCMA, which is a result of the analysis and grouping phases as described in Sec. 5.1 and 5.2, is called *a priori multilevel characteristic for matching* *approaches* and has been evaluated with the aim of building, at the end, a validated and revised *a posteriori* model used for the selection process (cf. Fig. 8.2).

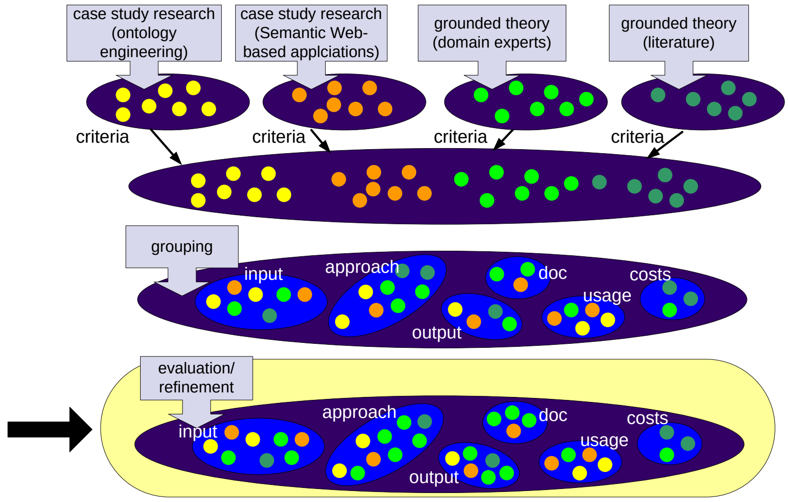


Figure 8.2: Evaluation of the matcher characteristic

The distinction between *a priori* and *a posteriori* has already been applied to a very wide range of objects, concepts, proposition, truths, models, and knowledge. The most common way of defining this distinction is to refer to Kant's claim that *a posteriori* knowledge, which is empirical and experience-based, is different from *a priori* knowledge in that the latter is non-empirical and independent of all experienced knowledge [37].

> *"...Since some authors find the negative characterization of a priori knowledge unsatisfactory... the common approach offering a positive characterization is to maintain that in the case of basic a priori propositions, understanding the proposition is sufficient to justify one in believing that it is true..."* [40]

Through the evaluation process, we validated the preliminary characteristic so as to assure the quality and accuracy of the model developed. The most important criterion in the MCMA evaluation is the reliability of the characteristic when applied to the selection issues. Keeping this goal in mind, the evaluation of the MCMA was dedicated to the overall relevance and accuracy of the *a priori* multilevel characteristic regarding the matcher issue [2].

[2] In this context, we mean the issue of selecting the appropriate matcher approaches regarding the given requirements.

Since asking the right questions to obtain the answers sought is crucial to the success of the evaluation process [200], we formulated the following evaluation questions:

- Is the MCMA accurate?
- Would a critic approve of this characteristic?
- How would professionals/experts within the matching domain rate the MCMA and its accuracy?

Taking into account our (most important) evaluation questions and the House taxonomy of the major evaluation models (cf. Tab.8.1), we attempted to strike a balance between the art criticism and accreditation (professional view) evaluation models. Considering the fact that both models are based on the tacit knowledge and expertise gained though experience, and that both can be classified as qualitative evaluations, we incorporated an expert-based methodology to evaluate the *a priori* MCMA. The data delivered in such an evaluation can be collected during (in-depth) interviews, from direct observations, or documents, and is based on the knowledge, experience, and expertise of specially chosen evaluators. Expert-based evaluation techniques refer to any form of evaluation involving an expert examination of a given system and assessment, i.e. the evaluation relies on the interpretation and judgement of the evaluator. *Expert judgment* as a part of the expert-based methods is an approach deployed to solicit informed opinions from individuals with specific expertise, while judgements are expressions, based on knowledge and experience, that experts make in response to technical problems [131]. This method may be used, on the one hand, to provide estimates on new, rare, complex, or poorly understood phenomena and, on the other hand, to integrate heterogeneous information and determine the state of knowledge in a problem. Since expert judgement is used in all technical fields, such as economics, (software) engineering, risk/safety assessment, knowledge acquisition, decision sciences, and environmental studies, it is also suitable for the MCMA evaluation. Typically, expert judgement is used in two ways: to structure the technical problem, i.e. experts may determine, for example, which data is relevant for analysis or which assumptions are valid, and to provide estimates, like failure or incidence rates. For our purposes, we have applied the expert judgment method in the former way, which is also the most common, statistically speaking, way of using this approach [131].

8.2.2 Choosing the evaluation for the MOMA selection processes

The evaluation of the MOMA Framework is performed in two phases: The first step of the evaluation, which is dedicated to the general relevance of the *a priori* Multilevel Characteristic for Matching Approaches (MCMA), is conducted, as mentioned in the previous section, using the expert judgment method. The additional aspects of the framework evaluation are directly related to its usage in real-world situations and to the accuracy of its predictions. Since the two remaining parts of the framework, AHP- and rule-based approaches, are immediately responsible for the matcher selection process, we have applied the same evaluation method for both approaches. In order to test the accuracy of the approaches, and taking into account the House classification of general evaluation models described earlier, the case study model, which is part of the qualitative evaluation methods, seems to be the most appropriate in this context.

> *"A case study is a method for learning about a complex instance, based on a comprehensive understanding of that instance obtained by extensive description and analysis of that instance taken as a whole and in its context."*[41]

The case study evaluation, like other evaluation methods, has both benefits and limitations, the former being the wealth of possibilities due to the detailed, qualitative information, the relatively straightforward use, and the opportunity to obtain and understand information at a sufficiently deep level. Its greatest deficit is the difficulty in identifying suitable cases. To tackle this weakness and to facilitate the evaluation of our approaches against real world matching case studies by examining the effects of both methodologies in the context of the same test cases, we have defined certain requirements that the cases appropriate for our purposes need to fulfill (the first five requirements must be met):

- different matchers are to be tested on the same input sources;
- matching results or the ranking of the matchers regarding their suitability for the specific sources must be available;
- matching results must be approved;
- sources to be matched must be specified;
- certain kinds of information regarding the desired matcher must be provided;
- information over whether additional sources, e.g. a dictionary, should be used;
- certain details concerning required matcher usage (such as the approach to be used for sources integration) should be available;
- information concerning matcher output should be specified, e.g. matcher cardinality;

- information concerning the matcher documentation should be at hand, e.g. the availability of examples; and

- specification regarding the significance of matcher costs, such as matcher licence costs, should be available.

In particular, the more information is available regarding the case studies, the better the prediction of our methodology can be put to the test.

Taking into account the requirements defined above, we have found such test cases in *Ontology Alignment Evaluation Initiative (OAEI)*, which, as stated in Sec. 3.1 & 5.3, aims to set up an evaluation campaign and benchmark tests to assess the strengths and weaknesses of the approaches available. During the campaigns, participants sample their matchers on the test cases provided by the contest organizers, who, at the end, rank matchers according to their suitability for a particular case. For the purpose of our evaluation, we have chosen two test cases :

- *anatomy* test case (expressive ontologies) that covers the domain of human anatomy and consists of two ontologies: the Foundational Model of Anatomy[3] and the OpenGalen Anatomy Model[4]; and

- *food* test case containing AGROVOC[5] vocabulary designed to cover the terminology in the subject fields of agriculture, forestry, fisheries, food, and related domains, and the NAL Agricultural Thesaurus[6], which is an online vocabulary reference tool for agricultural and biological terms.

Furthermore, as we collected information about the matching approaches presented in OAEI contests, which is essential for the evaluation of both matcher selection approaches, we asked the matcher developers involved to participate in our online questionnaire. The results from the questionnaire serve as a basis for the compilation of matcher data that is step #3 in the AHP method as well as for the matcher metadata within the rule-based approach (cf. Sec 6.3.3 & Sec. 7.3.1, respectively). After collecting preliminary information regarding the different approaches, the organizers of the OAEI campaign, as matcher experts, were asked to review them in order to ensure the objectivity of the data.

[3] `http://sig.biostr.washington.edu/projects/fm/AboutFM.html` accessed on 02.10.2008

[4] `http://www.opengalen.org/` accessed on 02.10.2008

[5] `http://www.fao.org/aims/agintro.htm` accessed on 02.10.2008

[6] `http://agclass.nal.usda.gov/agt/agt.shtml` accessed on 02.10.2008

8.2.3 Evaluation types in MOMA

As mentioned previously, we need to evaluate three main parts of the MOMA Framework by applying different evaluation approaches. After analyzing the evaluation goals for a particular situation and examining various evaluation methods, we decided to utilize the *expert judgment method to evaluate the MCMA* and the *case study approach to evaluate both the manual as well as the (semi-)automatic matcher selection processes* (cf. Fig 8.3). We thus decided to apply the qualitative approaches to evaluate the entire MOMA Framework, since these methods emphasize an interpretive approach that uses data to both pose and resolve research questions [105].

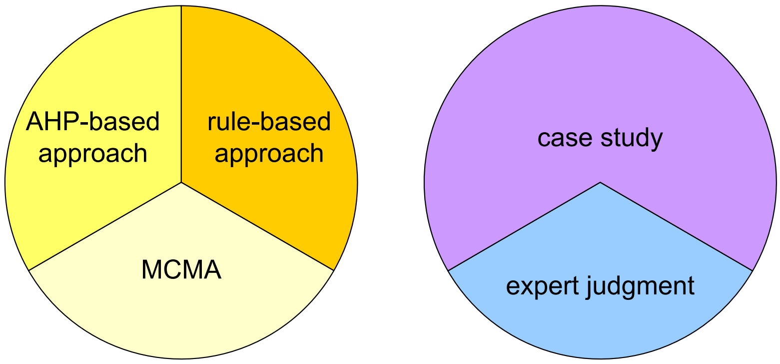


Figure 8.3: Evaluation of the MOMA Framework

In the following sections, we elaborate on the application of certain approaches for particular MOMA parts.

8.3 Evaluation of the MCMA

To evaluate the *a priori* Multilevel Characteristic for Matching Approaches, data collection during the comprehensive expert-based evaluation of the *a priori* MCMA was conducted in two steps: the first was carried out with reviews from anonymous matching experts, while the second was based on comments from specific experts. This information, completed by domain specialists, is used to evaluate and revise the initial *a priori* model, thus creating a validated *a posteriori* model. It is important to select competent judges, individuals who have background experience in the subject and are recognized as qualified to address the technical problems in question. As members of program committees of the various workshops and conferences are qualified to review the contributions submitted, we decided to enlist the help of this group of experts because of their ability to contribute to the MCMA evaluation process. In our particular case, through submission of the MCMA description as a part of our publication *'Applying an Analytic Method for Matching Approach Selection"* [133] to the International Workshop on Ontology Matching (OM-2006)[7], which was collocated with the 5th International Semantic Web Conference ISWC-2006[8], we were able to get the members of the Program Committee[9] on board. These experts suggested a slight restructuring of the preliminary taxonomy of the characteristic, suggested changes to the group names that, in their opinion, were more intuitive and easier to understand, and recommended the addition of new attributes. Following their suggestions, we have:

- split the factor *input parameter* into two new groups: factor *input type*, which contains attributes: scheme and instances, and factor *external sources*, which includes additional user input, previous matching decisions, training matches, domain constraints, and a list of valid domain values;
- added attributes: *dictionary*, *mismatch information*, *matching rules*, and *general schemas* to the factor *external sources*;
- taken into account the language independence of an algorithm by inserting the attributes *natural language (NL) independent* and *representation language (RL) independent* into the factors input natural language and input representation language, respectively;
- distinguished for greater precision between dependency on one or more natural/representation languages within the factors input natural language and input representation language;
- included the *DBscheme* as an attribute of the factor input category;
- renamed the factor matcher role into *application area*;
- renamed the factor usage parameter into *usage type*;
- renamed the dimension usability characteristic into *usage characteristic*; and

[7] `http://om2006.ontologymatching.org` accessed on 10.09.2008

[8] `http://iswc2006.semanticweb.org` accessed on 10.09.2008

[9] Some members of the Program Committee: R. Benjamins, Intelligent Software Components, Spain; P. Bouquet (University of Trento, Italy), F. Giunchiglia (University of Trento, Italy), A. Hess (Vrije Universiteit Amsterdam, Netherlands), R. Ichise (National Institute of Informatics, Japan), D. McGuinness (Stanford University, USA), N. Noy (Stanford University, USA), S. Staab (University of Koblenz, Germany), H. Stuckenschmidt (University of Mannheim, Germany), Y. Sure (University of Karlsruhe, Germany), M. Uschold (The Boeing Company, USA).

- defined more precisely the attributes related to the size of the ontology (input size, instance size, concepts size, etc.) by specifying "small", "middle" and "large" by numbers (small up to 100, middle 100-1,000, and large - over 1,000).

Since having a representative sample of experts to cover the full spectrum of opinions on an issue is a very important factor within the expert-based evaluation method, we conducted the second phase of the evaluation with the help of specific experts. This approach differs from the latter step because, in contrast to this step, in the earlier phase we did not know who exactly evaluated our characteristic, having only the list of potential reviewers who were members of the OM-2006 Workshop Program Committee. Since the development of our Multilevel Characteristic for Matching Approaches was supported by the Knowledge Web EU Network of Excellence, it enriched the Knowledge Web "Heterogeneity" work package (WP2.2), which contributed to the improvement of solving heterogeneity problems through matching and exploiting the resulting alignments, and was consequently a part of the Knowledge Web project deliverable (D1.2.2.2.1 [55]). Researchers involved in the work of the WP 2.2., and especially the main players responsible for the deliverable, who are accepted not only on the European (EU Commission) but also the international levels as experts within the ontology matching domain, evaluated our Multilevel Characteristic for Matching Approaches.[10] Furthermore, because each Knowledge Web deliverable had to go through quality control prior to submission, we were also able to obtain another expert opinion from the quality controller of the deliverable D1.2.2.2.1[11].

This step of the evaluation resulted in some changes within the taxonomy of the characteristic, the addition of a few new attributes, and the partitioning of the size-related factor. Strictly speaking, we:

- separated the attributes simultaneous execution and sequential execution from the *processing* factor, thus building a new factor *execution type*, since the experts suggested to

  > *"...separate the last two attributes"* (simultaneous execution and sequential execution) *"to another factor. The first...attributes"* (of the processing factor) *"deal with human involvement and the last two with parallelism"*; in this context we were also advised to *"...add things with respect to the name of the factor"*;

[10]The experts who worked in WP 2.2. and evaluated the characteristic were e.g. J. Euzenat (INRIA Rhône-Alpes, France), P. Shvaiko (University of Trento, Italy) and R. García-Castro (Facultad de Informática, Universidad Politécnica de Madrid, Spain).

[11]The quality controller for the D1.2.2.2.1 was Frank van Harmelen, Vrije Universiteit, Holland.

- renamed factor adaptation parameter to *adaptation ability* within the usage dimension;
- changed some attribute descriptions (e.g. matching experts recommended *"...using only the term entities"* instead of using "concepts", "elements" within the description of factor matching cardinality) in order to achieve greater precision and understandability, as the reviewer underscored the need to be more explicit in the description of attributes;
- changed the breakdown of the characteristics *size of ontology*, *number of instances*, *concepts* and *axioms* since the experts were not convinced that the preliminary categorization was appropriate for our goals;
- extended an attribute domain constraints with the addition of *domain specific resources*;
- introduced the attribute DAG structure into the factor input structure since *"...DAG is an important class before talking about arbitrary graphs..."*; and
- extended the attributes of the factor application area by the augmentation of merging, transformation, query answering, data mediation, query reformulation, and navigation for greater precision.

The most controversial area during the evaluation of the matcher characteristic was *input size*. In particular, the reviewers had different opinions regarding the breakdown of the characteristics *size of ontology*, *number of instances*, *concepts*, and *axioms* into the categories small, middle and large, which form the preliminary classification. One of the experts was of the opinion that:

> *"...many schemas ... that come from medium companies are about 40-60 entities. What is now under "small" and "medium" (note: small up to 100, middle between 100 and 1000) does not make a clear cut to me. I would rather prefer: up to 30 (small), up to 100 (medium), up to 1000 (large) and more than 1000 (extra large)".*

On the other hand, another reviewer asked why the values for sizing should be chosen as (0-30, 30-100, 100-1000, and over 1000), while another expert added that it would be

> *"...strange to have just one category for ontology with more than 1,000 concepts"*, while there would be *"...two (categories) for ontologies containing less than 100 concepts. The problem is even more important for the instance aspect; considering a database of more than 1000 individuals as extra large is wrong...".*

We took into account the comments of the judges and re-examined the literature and some Semantic Web-based use cases where matching approaches were utilized. We decided to modify the old classification following the experts' recommendations, which, after the corresponding revisions, resulted in final MCMA (cf. Tab. 5.1, 5.2, 5.3, 5.4, 5.5, and 5.6).

8.3.1 Lessons learned

The initial *a priori* characteristic has been qualitatively evaluated by applying the expert judgment methodology. The experts who participated in the evaluation process expressed their concerns about the matcher characteristic-related issues. They wrote down their experiences regarding the development and application of matching algorithms and proposed changes within the characteristic. Nevertheless, the matcher experts agreed that this characteristic for matching approaches is inventory and quite precise since it goes beyond the published characterizations of matching systems in terms of attributes [56, 98, 102, 157, 169]. The evaluation conducted resulted in revised *a posteriori* Multilevel Characteristic for Matching Approaches to be used for the selection of suitable matching approaches whose purpose is to establish more precisely the adequacy of a matcher with an application. The evaluation of the reliability of the MCMA when applied to the selection issues is described in the next sections.

8.4 Evaluation of the AHP-based approach

Since the main aspects of the MOMA Framework evaluation are directly related to its application in real world situations and to the accuracy of its predictions, in the following we describe how we applied our methodology to two case studies we selected from the OAEI 2006 Campaign. According to the AHP-based method (cf. Sec. 6.3), having the matcher data collected during the OAEI contest and the criteria hierarchy in the form of the MCMA, we now needed to weigh the criteria in a pairwise comparison according to the requirements of the OAEI 2006 Campaign case studies. Thus, for each level of criteria (dimension, factors, and attributes from the multilevel characteristic for matching approaches), pairwise comparisons between the sibling nodes were to be built. In the following sections, we elaborate on how, using the revised Multilevel Characteristic of Matching Approaches and the AHP scale, we specified the requirements for the suitable matchers. Furthermore, we discuss the corresponding matcher rankings delivered by the AHP-based approach and compare them with the results obtained during the OAEI 2006 Campaign.

8.4.1 Anatomy test case

As mentioned before (cf. Sec 8.2.2), one of the test cases during the OAEI 2006 Campaign was the anatomy real world case with the Foundational Model of Anatomy and the OpenGalen Anatomy Model. To define the requirements of the application that may be built on the basis of these sources, we analyzed the available information regarding these two ontologies along with the restrictions regarding the suitable matching approaches. In order to compare the application requirements and the characteristic of the incoming sources, we first need to "translate" each part of the requirement description into corresponding terms from the Multilevel Characteristic for Matching Approaches (MCMA)(cf. Sec.5). The overview of the "translation" of terms included from the description of the input anatomy sources and the matching processing, which reflect the requirements regarding the potential suitable matchers, is shown in Tab. 8.4, and the "translation" process is briefly explained in the following. We first analyze as exemplary a fragment of the general description of the anatomy test sources as well as the OAEI regulations regarding the matching approaches, and mark the crucial terms that are important for the matcher selection and that will be translated into the corresponding MCMA terms.

Excerpt of the general description of the anatomy test sources[12]:

> *"The task is placed in the **medical domain** as this is the domain where we find **large**, carefully designed **ontologies**. The specific characteristics of the ontologies are:*
>
> - ***Very large models**: prepared to handle **OWL models** of more than 50MB;*
> - ***Extensive Class Hierarchies**: then **thousands of classes** organized according to different views on the domain; and*
> - ***Complex Relationships**: Classes are connected by a **number of different relations**."*

Considering the above description:

- the term **OWL models** from the description of the anatomy sources has been annotated in terms of MCMA as:
  - `input characteristic:input category:ontology` and
  - `input characteristic:input formality level:formal ontology;`

[12] `http://oaei.ontologymatching.org/2006/anatomy/`, accessed on 10.03.2008

- the term carefully designed **ontologies** has been mapped to the MCMA term `input characteristic:input category:ontology`; and
- in the case of the term **"thousands of classes"**, we have translated it into MCMA as:
  - `input characteristic:input size:extra large`,
  - `input characteristic:input size:large`,
  - `input characteristic:input size:number of concepts:large`,
  - `input characteristic:input category:number of concepts: extra large`,

  i.e. the potential matcher must deal with *large* (up to 1000 primitives) and *extra large sources* (over 1000 primitives) containing *large* (up to 1000 concepts) and *extra large* numbers of concepts (over 1000 concepts).

From the following excerpt of the description regarding the matching processing rules within the OAEI Campain:

> *"The task is to find* ***alignment between classes in the two ontologies****. In order to find the alignment any information in the* ***two models*** *can be used. ... it is allowed to use* ***background knowledge****, that has not specifically been created for the alignment tasks (i.e. no hand-made mappings between parts of the ontologies). Admissible background knowledge are* ***other medical terminologies*** *such as UMLS as well as* ***medical dictionaries*** *and document sets*[13]*... "*

we have, for instance:

- considered the term **background knowledge**, which has been mapped to the MCMA term:
  `input characteristic:external sources:domain specific resources`
- while the term **medical dictionaries** has been translated to:
  `input characteristic:external sources:dictionary`

Description from text	**MCMA notation**
General information	
ontologies	input characteristic: input category: ontology
to be continued ...	

[13] `http://oaei.ontologymatching.org/2006/anatomy/`, accessed on 10.03.2008

Description from text	**MCMA notation**
large ontologies	input characteristic: input size: **large** input characteristic: input size: **extra large**
very large models	input characteristic: input size: **large** input characteristic: input size: **extra large**
owl models	input characteristic: input category: ontology input characteristic :input formality level: formal ontology
extensive class hierarchies	input characteristic: input structure: tree structure input characteristic: input size: number of concepts: **large** input characteristic: input size: number of concepts: **extra large**
thousands of classes	input characteristic: input category: input size: **extra large** input characteristic: input size: input size: **large** input characteristic: input size: number of concepts: **large** input characteristic: input size: number of concepts: **extra large**
different views of the domain	input characteristic: input model type: domain ontology
complex relationships	input characteristic: input structure: heterogeneous relations
number of different relations	input characteristic: input structure: heterogeneous relations input characteristic: input size: number of relations: small input characteristic: input size: number of relations: medium
classes . . . (**large ontologies**)	input characteristic: input category: number of concepts: **large** input characteristic: input size: number of concepts: **extra large**
relations	input characteristic: input size: number of relations: small input characteristic: input size: number of relation: medium
alignment between two ontologies	input characteristic: input size: number of ontologies: two
any information in the two models can be used	input characteristic: external sources: domain specific resources
it is allowed to use background knowledge	input characteristic: external sources: domain specific resources
limited use of axioms	input characteristic: input size: number of axioms: small
background knowledge, that has not been specifically created for the alignment tasks	(NO) input characteristic: external sources: previous matching decision (NO) input characteristic: external sources: training matches (NO) input characteristic: external sources: miss-match information (NO) input characteristic: external sources: matching rules
other medical terminologies	input characteristic: external sources: domain specific resources
medical dictionaries	input characteristic: external sources: dictionary
document sets	input characteristic: external sources: domain specific resources
Foundational Model of Anatomy	
owl ontology	input characteristic: input category: ontology input characteristic: input formality level: formal ontology
class hierarchy	input characteristic: input structure: tree structure
relations between classes	input characteristic: input size: number of relations: medium
free text documentation	input characteristic: input size: domain specific resources
synonyms	input characteristic: external sources: domain specific resources
names in different languages	input characteristic: input natural language (nl): many languages input characteristic: input natural language (nl): nl-independent
OpenGalen Anatomy Model	
owl ontology	input characteristic: input category: ontology input characteristic: input formality level: formal ontology
big concept hierarchy	input characteristic: input size: number of concepts: **large** input characteristic: input size: number of concepts: **extra large**
relations between concepts	input characteristic: input category: number of relations: small input characteristic: input size: number of relations: medium

Table 8.4: Translation of requirements into notation of the AHP-methodology

After we isolated the crucial terms and annotated them using the multilevel characteristic, we needed to specify the importance of each feature using the AHP scale (cf. Tab. 6.2) for pairwise comparisons between sibling nodes. Following requirements regarding the suitable matcher from multilevel notation of the anatomy case study, it becomes obvious that one of the essential features is the size (large or extra large) of the sources to be matched. In the context of the AHP-based MOMA Framework, it is expressed by a very high rating of the characteristics reflected by the size of incoming sources. This means that suitable matchers *must* be capable of dealing with inputs of such magnitude. Figure 8.4 shows the weighting of the attributes *small*, *medium*, *large*, and *extra large* within the factor *ontology size* to provide an insight into the way properties characterizing the anatomy case studies are ranked.

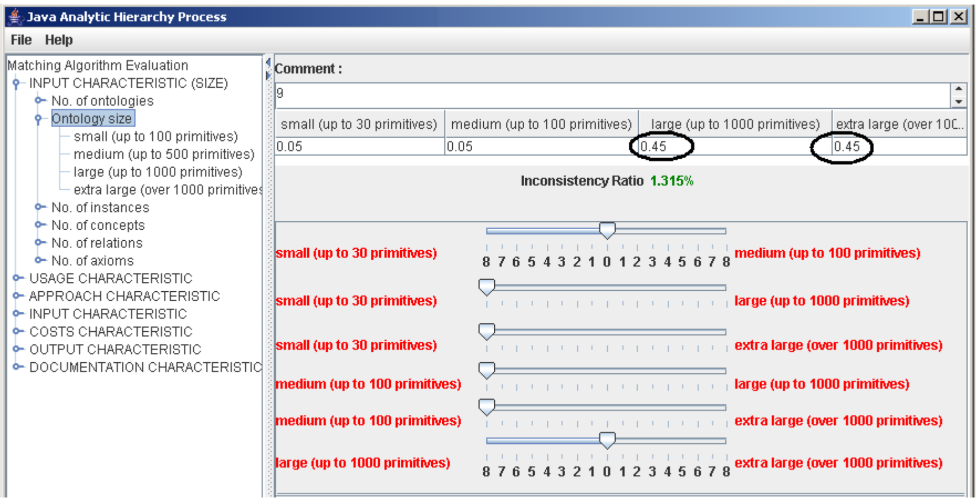
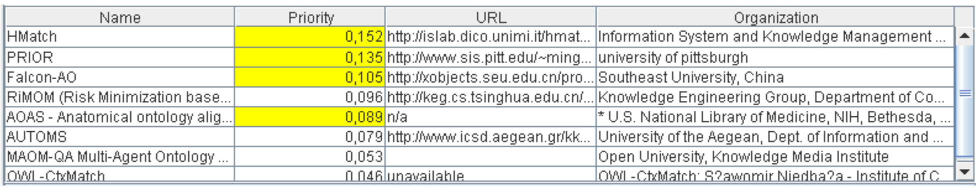


Figure 8.4: Weighting of the attributes within the factor “ontology size”

The last step of the AHP-based methodology for matcher selection is the calculation of the result. The AHP result is a list of matching algorithms ranked according to their suitability in terms of requirement set defined by the anatomy case study (cf. Fig. 8.5).



Name	Priority	URL	Organization
HMatch	0,152	http://islab.dico.unimi.it/hmat...	Information System and Knowledge Management ...
PRIOR	0,135	http://www.sis.pitt.edu/~ming...	university of pittsburgh
Falcon-AO	0,105	http://xobjects.seu.edu.cn/pro...	Southeast University, China
RiMOM (Risk Minimization base...	0,096	http://keg.cs.tsinghua.edu.cn/...	Knowledge Engineering Group, Department of Co...
AOAS - Anatomical ontology alig...	0,089	n/a	* U.S. National Library of Medicine, NIH, Bethesda, ...
AUTOMS	0,079	http://www.icsd.aegean.gr/kk...	University of the Aegean, Dept. of Information and ...
MAOM-QA Multi-Agent Ontology ...	0,053		Open University, Knowledge Media Institute
OWL-CtxMatch	0,046	unavailable	OWL-CtxMatch: S?awomir Niedba?a - Institute of C

Figure 8.5: Anatomy test case: results of the AHP-based MOMA (screenshot)

Comparison: AHP methodology vs. OAEI 2006 Campaign

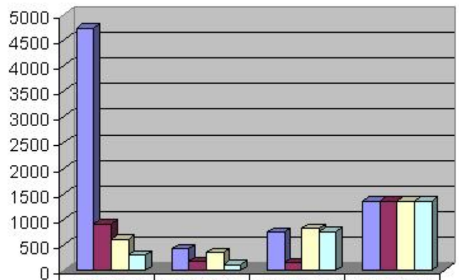
In the anatomy test case, the OAEI evaluation of the matching results delivered by the participating systems was conducted in a diluted form.[14] The organizers did not attempt at precision or recall of the systems by applying strict measurements of the different systems[15], as defined in Sec. 3.3, but instead analyzed the commonalities and differences of the results of the different systems as well as the coverage of the terminology in the ontologies. Instead of directly measuring precision and recall, they concentrated on indicators that point to the systems' probable precision and recall. Therefore, the coverage of the terminology is an indicator for the recall of the matching systems. The percentage of mappings that have also been found by other systems is an indicator of the precision, as mappings found by different systems are more likely to be correct than mappings only found by a single system. To analyze the coverage of mappings produced by the systems, the OAEI has analyzed the overlap in the mappings produced.

Considering the numbers in Fig. 8.6, Falcon is the system with the highest degree of overlap with other systems. NIH[16] has a large number of mappings that have been found by other systems as well, but it also has a significant number of mappings not located by any other system. For the PRIOR system, we have observed an even larger number of mappings not found by other systems. The OAEI assessed that the results confirm the observation that the **Falcon system seems to emphasize the correctness of matching results**, which is an indication of high **precision**. This is documented in a relatively small number of mappings, most of which have also been found by at least one other system. On the other hand, the **ISLab HMatch system seems to focus more on recall**. This is evidenced by the large number of mappings found as well as by the fact that the ISLab HMatch results contain a very large number of mappings that have also been found by at least one other system.

[14] OAEI 2006 results of the anatomy task: `http://oaei.ontologymatching.org/2006/results` accessed on 28.06.2008

[15] The matcher that participated in the OAEI Campaign 2006 has been briefly described in Sec. 3.4.

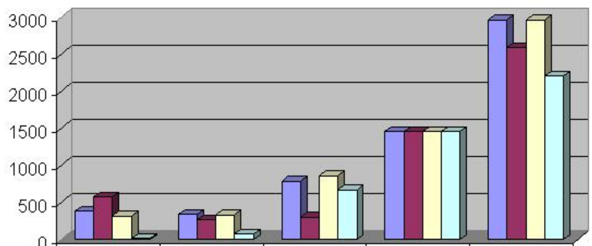
[16] The NIH system is also known as Anatomical Ontology Alignment System (AOAS), cf. Sec. 3.4.



	found by no other system	found by 1 other system	found by 2 other systems	found by 3 other systems
IsLab	4741	419	748	1351
PRIOR	908	181	149	1351
NIH	608	351	822	1351
Falcon	297	109	761	1351

Figure 8.6: Number of mappings

Furthermore, considering the number of terms, the OAEI confirmed that there is a significant overlap in the sets of terms that can be mapped by the different systems (cf. Fig. 8.7). This result had been expected by the OAEI organizers, as obvious matches should be found by all systems – around 1500 terms have been mapped by the ISLab HMatch, Falcon, NIH (AOAS), and PRIOR.



	this system only	found by 2 systems	found by 3 systems	found by 4 systems	total number of terms found
IsLab HMatch	384	340	784	1455	2963
PRIOR	574	262	299	1455	2590
NIH	318	331	862	1455	2966
Falcon	10	69	670	1455	2204

Figure 8.7: Number of terms

On the basis of the outcomes provided by OAEI Campaign 2006, and considering the above discussion on the evaluation results, we have evaluated the ranking results achieved by the AHP-based MOMA Framework. Analyzing the total number of terms provided by the particular OAEI systems (cf. Fig 8.7), we observe that the matchers ranking delivered by the AHP-based methodology is comparable to the results from the OAEI 2006, as shown in Tab. 8.5. The order of the ISLab HMatch, PRIOR, and Falcon approaches is the same in both the cases of the AHP-based processing and the real-time execution (regarding the terms mapped in the context of the OAEI Campaign 2006). The results of our manual MOMA Framework is even more convincing if we consider the recall aspects of the various matchers. With regard to the OAEI results as described previously, since the ISLab HMatch system focuses on recall and the Falcon is a system with quite high precision, our system delivers accurate results, placing the ISLab HMatch system in the first position ahead of the Falcon system.

Results / Ranking - Anatomy test case				
OAEI Campaign 2006			**AHP-based MOMA**	
Approach	**Terms mapped**	**Ranking**	**Approach**	**Ranking**
NIH (AOAS)	2966	1	-	-
ISLab HMatch	2963	2	ISLab HMatch	1
PRIOR	2590	3	PRIOR	2
Falcon	2204	4	Falcon	3
-	-	-	RiMOM	4
-	-	-	NIH (AOAS)	5

Table 8.5: Anatomy case: OAEI 2006 Campaign vs. AHP-based results

However, one deviance between the AHP-based results and outcomes achieved by the OAEI matcher execution can be identified: the NIH (AOAS) approach is placed first within the context of OAEI results, though it is behind the other approaches in the MOMA ranking. Although the NIH (AOAS) approach had the highest number of mappings during the contest, it is not classified as high in the ranking delivered by the AHP-based methodology. However, if we take into account that, during the OAEI 2006, the NIH approach had a significant number of mappings not found in any other system, the result of the manual MOMA Framework begins to appear more comprehensible.

8.4.2 Food test case

Another test case during the OAEI 2006 was the food task (cf. Sec 8.2.2) in which contestants **automatically** aligned **two** SKOS **thesauri** using relations from the SKOS **mapping vocabulary**. One of the sources was the AGROVOC **thesaurus**, a **multi-**

lingual structured and controlled vocabulary designed to cover the terminology of all the subject fields of agriculture, forestry, fisheries, food, and related **domains**. AGROVOC is available in **nine languages**, contains approximately **16,000 terms**, and is made up of terms consisting of one or more words representing a singular concept. For each term, a word block is displayed showing the **hierarchical relation** to other terms: BT (broader term), NT (narrower term), RT (related term), UF (non-descriptor). The second source, the NAL Agricultural Thesaurus, is an online vocabulary reference tool for **agricultural and biological terms**. The **thesaurus** is organized into 17 subject categories, indicated by the "Subject Category" designation in the thesaurus, and contains approximately **41,000 terms**. The NAL includes **hierarchical**, **equivalence**, and **associative relationships** among its concepts.

Although the description of the food test case was not as detailed as the description of the anatomy task, we managed to isolate certain issues relevant to the selection of the appropriate matching approach and which can be deployed within the AHP tool. In the passage above, we have marked in bold important information that has been translated into terms taken from our multilevel characteristic of matching approaches. For the food task, we conducted the same procedure as for the anatomy task (cf. Sec 8.4.1) using the MCMA (cf. Sec. 5.2) and the AHP sale (cf. Tab. 6.2) and have determined, for example, that:

- the paramount attribute within the highly important factor *input category* is the attribute *thesaurus*;
- the *large* and *extra large* attributes have *absolute importance* in comparison with other attributes of the *ontology size* factor; and
- the paramount attribute regarding the *absolute important* matcher *processing* factor (within the dimension *approach characteristic*) is the *black box paradigm*.

After weighting all the relevant (regarding the information gleaned from the case description) dimensions, factors, and attributes, the AHP tool calculated the results and delivered the ranking of the most suitable matching approaches (cf. Fig. 8.8).

Name	Priority	URL	Organization
HMatch	0,169	http://islab.dico.unimi.it/hm...	Information System and Knowledge Managem...
Falcon-AO	0,113	http://xobjects.seu.edu.cn/p...	Southeast University, China
PRIOR	0,101	http://www.sis.pitt.edu/~min...	university of pittsburgh
RiMOM (Risk Minimization bas...	0,1	http://keg.cs.tsinghua.edu.c...	Knowledge Engineering Group, Department of ...
AUTOMS	0,092	http://www.icsd.aegean.gr/k...	University of the Aegean, Dept. of Information a...
AOAS - Anatomical ontology ali...	0,068	n/a	* U.S. National Library of Medicine, NIH, Bethes...
MAOM-QA Multi-Agent Ontology...	0,059		Open University, Knowledge Media Institute
OWL-CtxMatch	0,041	unavailable	OWL-CtxMatch: S?awomir Niedba?a - Institute

Figure 8.8: Food test case: AHP-based MOMA Framework results (screenshot)

Comparison: AHP-methodology vs. OAEI 2006 Campaign

As in the medical task in the previous section, we have also compared the food test case ranking provided by the AHP tool with the results achieved by the OAEI 2006 Campaign. The evaluation of the results during the OAEI2006 was conducted differently from the evaluation of the medical test case and was based on the following steps[17]:

- participants submitted their mappings in the common format for alignments to the OAEI organizers;
- the mappings were distributed at random and anonymously to a group of domain experts (food safety researchers and food product development researchers);
- a (small) sample of the mappings were distributed at random and anonymously to other participants;
- the domain experts and participants were asked to assess the mappings; and
- the evaluation measurements of the participants' systems were calculated according to the list of reference alignments.

At the end, the organizers submitted the precision and recall of each participating system; however, as declared by the OAEI organizers, evaluating a sample large enough to lead to significant results for recall was not feasible. Nevertheless, we have compared the outcomes achieved by the AHP-based MOMA Framework in the food test case with both the precision and recall OAEI results. The precision in the food test case was performed in three categories: biological and chemical mappings, taxonomical mappings, and miscellaneous mappings (geography, legislation,

[17] OAEI 2006 results of the food task: `http://oaei.ontologymatching.org/2006/results/food` accessed 15.05.2008

food stuffs, etc.). In our analysis, however, we concentrated on the all-round precision, which was calculated on the basis of the single outcomes for each approach. As shown in Fig. 8.9, the **Falcon system has greater precision** than all other systems; however, the difference between the Falcon and the other systems is rather insignificant.

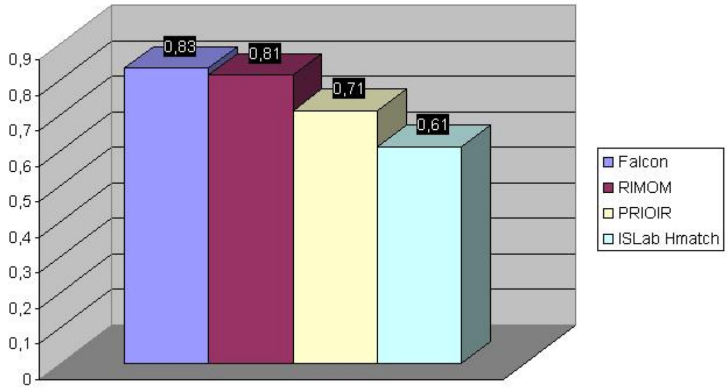


Figure 8.9: All-round precision in the OAEI 2006 food test case

Comparing the AHP selection process, the Falcon approach also achieved the first position, while the matchers PRIOR and RiMOM ranked third (and fourth), since the differences between the results obtained by these two approaches were minute (cf. Fig. 8.8). Even if the AHP-based ranking of the ISLab HMatch approach is slightly different from the OAEI precision results, both results are clearly comparable (cf. Tab. 8.6).

Results / Ranking - Food test case				
OAEI Campaign 2006			**AHP methodology**	
Approach	**"all-round" precision**	**Ranking**	**Approach**	**Ranking**
-	-		ISLab HMatch	1
Falcon	0.83	1	Falcon	2
RiMOM	0.81	2	RiMOM/PRIOR	3
PRIOR	0.71	3	RiMOM/PRIOR	4
ISLab HMatch	0.61	4	-	-

Table 8.6: Food case: OAEI 2006 Campaign vs. AHP-based results

As mentioned earlier, even if the evaluation of a sample large enough to lead to significant results for recall was not feasible, the OAEI delivered some preliminary results (indication for recall results) on 191 mappings in the two sets from miscellaneous and two sets from taxonomical domain where recall of only exactMatch mappings as well as recall of exact, broad, and narrowMatch were examined. In our MOMA evaluation, we concentrated on the recall of all types of mappings. Considering the indication for recall of the participating matching approaches, we note that their recall values are quite similar to each other, even when the RiMOM system seems to be slightly better than the others (cf. Fig. 8.10).

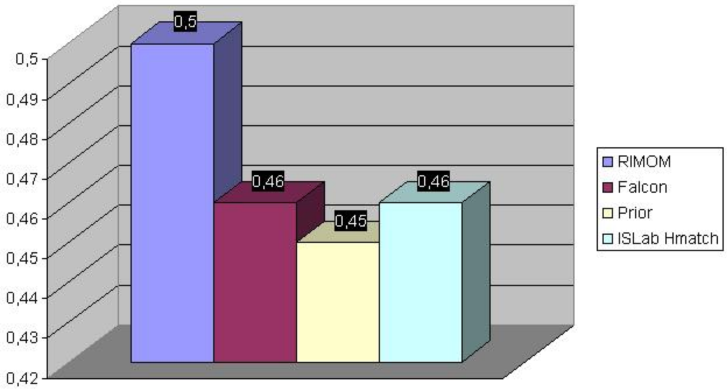


Figure 8.10: All-round (indication for) recall in the OAEI 2006 food test case

The **RiMOM system managed to discover more good results than the Falcon system on the small sample recall bases, though at the cost of some precision**. Comparing these results with the AHP-based outcomes, we conclude that our ranking of ISLab HMatch, Falcon, and PRIOR systems is the same as in OAEI (cf. Tab. 8.7).

Results / Ranking - Food test case				
OAEI Campaign 2006			**AHP-based MOMA**	
Approach	**recall**	**Ranking**	**Approach**	**Ranking**
RiMOM	0,50	1		-
ISLab HMatch /Falcon	0.46	2	ISLab HMatch	1
ISLab HMatch /Falcon	0.46	3	Falcon	2
PRIOR	0.45	3	RiMOM/PRIOR	3
-	-	-	RiMOM/PRIOR	4

Table 8.7: Food case: OAEI 2006 Campaign vs. AHP-based results (recall)

The only difference between the AHP-based and real execution-based outcomes can be observed regarding the RiMOM system. However, the AHP-based result is intended to classify in a similar way as results based on real testing and execution. This is especially apparent if we note that all these approaches[18] deployed in the food task, and therefore having proved their relevance and suitability for this test case, achieve the first four places within the ranking list conducted by the AHP-based methodology.

8.4.3 Lessons learned

Application of the AHP-based MOMA Framework in the two case studies shows that it produces relevant information for the detection of suitable matching approaches. This information can serve as a basis for the reuse of existing matchers in new ontology-based applications. At the same time, the additional costs of applying AHP to the matcher selection issue is relatively minor if we consider the long and time consuming process of developing new matcher approaches, or if we take into account the costs related to the selection of the appropriate matchers with neither methodological nor tool support. We assume that the adaptation of the AHP method to the matcher issue can optimize the entire process of semantic-based application development, and consequently contribute to the realization of the fully developed Semantic Web.

8.5 Evaluation of the rule-based approach

In order to evaluate the rule-based approach, we have to prepare the MOMA Framework Knowledge Base that includes a description of available matching algorithms (matcher metadata), which may be selected for application, the metadata of ontological sources (ontology metadata) to be matched, and a set of rule statements stored in the rule repository describing the dependencies between the matching approaches and the ontological sources. As the rules are predefined in the framework, all we had to do was to create the corresponding matcher and ontology metadata; we need only to extract the matcher metadata and automatically generate the ontology metadata for both sets of ontologies. The metadata for the matching approaches, which are available in the system (i.e. matchers involved in the OAEI Campaign 2006), has been created according to the matcher metadata model, which was adapted from the MCMA on the basis of data extracted from the online survey filled out by matcher developers and evaluated by the matcher domain experts (cf. Fig. 8.11).

Since the evaluation process is conducted according to the same matcher set, i.e. the matchers participating in the OAEI2006 contest, the matcher metadata extrac-

[18]Eight out of ten developer teams participating in the OAEI 2006 Campaign filled out our questionnaire; their matchers have been taken into account during the processing of our AHP-based methodology.

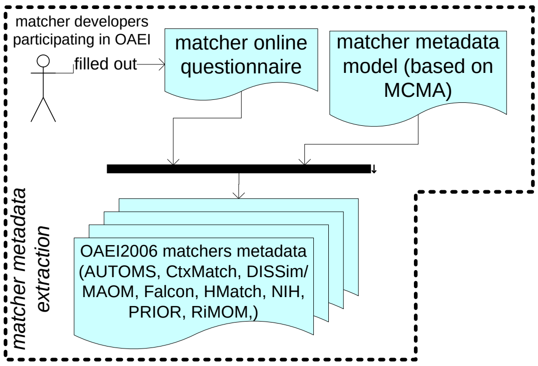


Figure 8.11: Extracting of the metadata for OAEI matchers

tion is the same for both test cases. However, since the next step, the generation of the ontology metadata, must occur separately for each ontological source, we first elaborate the evaluation of the anatomy test case, which is followed by that of the food.

8.5.1 Anatomy test case

For both incoming ontological sources of the anatomy test case, the MOMA Framework automatically generates the ontology metadata based on the ontology metadata model, as shown in Fig. 8.12, i.e. the rule-based MOMA Framework generates FMA metadata (cf. Fig. 8.13) and OpenGalen metadata (cf. Fig. 8.14).

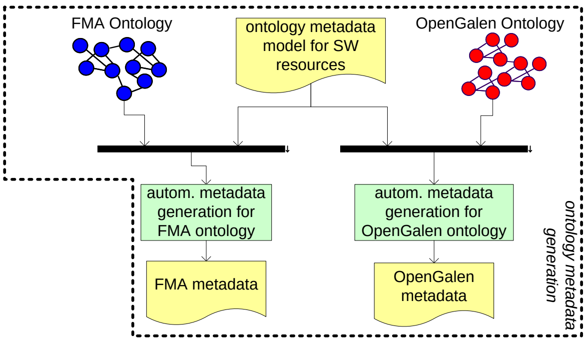


Figure 8.12: Generating of the metadata for the anatomy sources (FMA & OpenGalen)

file:/D:/diss/data/ekaw08%20ontologien/anatomie/FMA.owl

Property	Value
http://swpatho.ag-nbi.de/context/meta.owl#numberOfAxioms	0
http://swpatho.ag-nbi.de/context/meta.owl#hasType	domain ontology
http://moma.ag-nbi.de/meta_ext.owl#namespace	http://sig.biostr.washington.edu/projects/fm
http://swpatho.ag-nbi.de/context/meta.owl#hasDocumentation	http://www.fma.org/; http://en.wikipedia.org/wiki/Fullmetal_Alchemist; http://www.fmadirect.com/; htt...
http://swpatho.ag-nbi.de/context/meta.owl#hasNaturalLanguage	English
http://swpatho.ag-nbi.de/context/meta.owl#numberOfInstances	0
http://swpatho.ag-nbi.de/context/meta.owl#numberOfImportedOntologies	0
http://swpatho.ag-nbi.de/context/meta.owl#numberOfInheritanceLevels	21
http://swpatho.ag-nbi.de/context/meta.owl#numberOfClasses	72559
http://moma.ag-nbi.de/meta_ext.owl#isConsistentAccordingToPellet	true
http://moma.ag-nbi.de/meta_ext.owl#ontologyTypeAccordingToInternalStructure	Ontology
http://swpatho.ag-nbi.de/context/meta.owl#numberOfComments	0
http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://swpatho.ag-nbi.de/context/meta.owl#Ontology
http://swpatho.ag-nbi.de/context/meta.owl#numberOfProperties	100
http://swpatho.ag-nbi.de/context/meta.owl#hasFormalityLevel	http://swpatho.ag-nbi.de/context/meta.owl#formal

Figure 8.13: FMA metadata (screenshot)

file:/D:/diss/data/ekaw08%20ontologien/anatomie/OpenGALEN.owl

Property	Value
http://swpatho.ag-nbi.de/context/meta.owl#numberOfAxioms	0
http://swpatho.ag-nbi.de/context/meta.owl#hasType	domain ontology
http://moma.ag-nbi.de/meta_ext.owl#namespace	http://www.opengalen.org/open/crm/crm-anatomy.owl
http://swpatho.ag-nbi.de/context/meta.owl#hasDocumentation	http://www.opengalen.org/; http://www.opengalen.org/sources/sources.html; http://en.wikipedia.or...
http://swpatho.ag-nbi.de/context/meta.owl#hasNaturalLanguage	English
http://swpatho.ag-nbi.de/context/meta.owl#numberOfInstances	9495
http://swpatho.ag-nbi.de/context/meta.owl#numberOfImportedOntologies	0
http://swpatho.ag-nbi.de/context/meta.owl#numberOfInheritanceLevels	0
http://swpatho.ag-nbi.de/context/meta.owl#numberOfClasses	72626
http://moma.ag-nbi.de/meta_ext.owl#isConsistentAccordingToPellet	true
http://moma.ag-nbi.de/meta_ext.owl#ontologyTypeAccordingToInternalStructure	Ontology
http://swpatho.ag-nbi.de/context/meta.owl#numberOfComments	0
http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://swpatho.ag-nbi.de/context/meta.owl#Ontology
http://swpatho.ag-nbi.de/context/meta.owl#numberOfProperties	33
http://swpatho.ag-nbi.de/context/meta.owl#hasFormalityLevel	http://swpatho.ag-nbi.de/context/meta.owl#formal

Figure 8.14: OpenGalen metadata (screenshot)

With regard to the FMA & OpenGalen metadata generated, we can confirm the crucial characteristic of matching approaches capable of dealing with these two ontological sources. Thus, in terms of the semantic features that are related to the formal semantics of the representation language and the meaning of the ontology content (cf. Sec. 7.3.2), we need approaches that can handle *formal domain ontologies*, as both sources are *ontologies*, namely *domain ontologies*, and have a *formal* formality level. Furthermore, regarding the quantitative syntactic features, both ontologies have over 72,000 classes each and share a number of properties (FMA - 100 properties, OpenGalen - 33 properties); only OpenGalen has some instances while none of the sources have axioms. These numbers and attributes mean that the desired matchers need to deal with (extra) large ontological sources, taking into consideration the number of concepts and middle-sized sources regarding properties, though the matching approaches do not have to handle axioms[19]. Deploying the matcher

[19]For details regarding the feature classification, the reader is referred to the MCMA in Sec. 5.2.

metadata for OAEI matchers and the metadata for both ontological sources, the MOMA Framework has executed the predefined rules from the rule repository. As the mandatory rules were satisfied by the ontologies, since both sources describe the same domain (i.e. anatomy), and use the same natural language (i.e. English), the framework conducted the selection rules, and then ranked the matchers according to the requirements based on the ontologies and their characteristics (cf. Fig. 8.15).

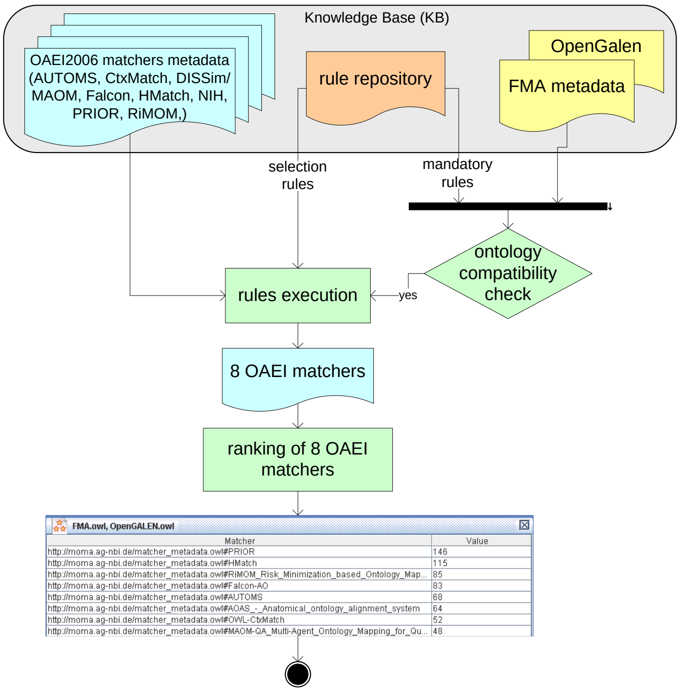


Figure 8.15: Anatomy test case: the rule-based matcher selection procedure

Comparison: Rule-based methodology vs. OAEI 2006 Campaign

The results of the selection process achieved by the rule-based MOMA Framework (cf. Fig 8.16) performed prior to the execution of a matching algorithm are relatively similar to the results achieved by the OAEI2006 Campaign (cf. Tab. 8.8), even if the exact ranking of the OAEI results has not been achieved. Analyzing the rule-based results of the MOMA, we conclude that the PRIOR system, which strives to find a balance between precision and recall, has achieved the first position ahead of the ISLab HMatch system, which seems to focus more on high recall, as mentioned in Sec. 8.4, and the Falcon, which mainly concentrates on precision (cf. Fig. 8.6 and 8.7).

The only major difference between the rule-based MOMA outcomes and the results based on the real execution and testing of the approaches occurs in the placing of the NIH (AOAS) approach. However, as we have mentioned in Sec. 8.4.1, the NIH approach had a significant number of mappings not found by any other system during the contest, thus this may be responsible for the different findings.

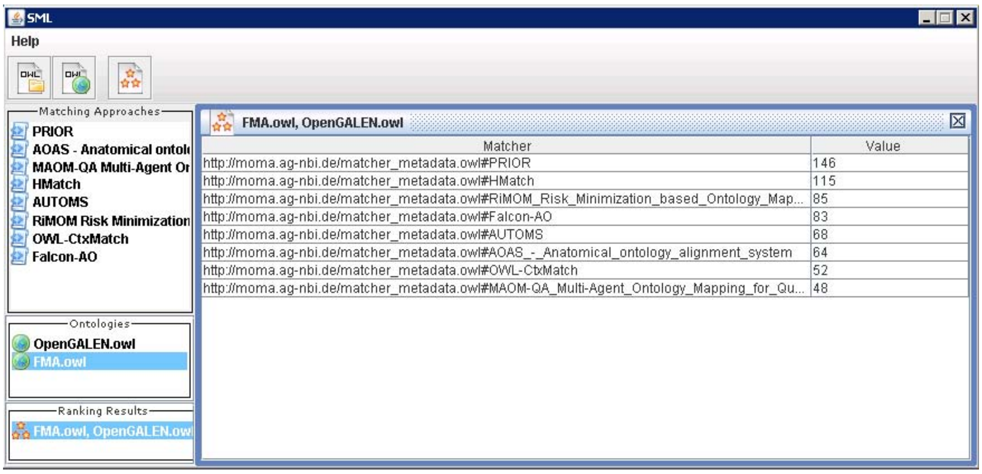


Figure 8.16: Anatomy test case: results of the rule-based MOMA Framework (screenshot)

Results - Anatomy test case				
OAEI Campaign 2006			**Rule-based MOMA**	
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/RiMOM	3
-	-	-	AUTOMS	4
-	-	-	NIH (AOAS)	5

Table 8.8: Anatomy task: OAEI 2006 Campaign vs. MOMA-based results

8.5.2 Food test case

The same procedure as for the anatomy case was conducted for the food test case: after the matcher metadata was extracted, the MOMA Framework automatically generated the ontology metadata for incoming sources, which resulted in an AGROVOC metadata and NAL Agricultural metadata. As the mandatory rules were satisfied, the framework fired off the pre-defined selection rules and allocated the weighting (from the survey data using a predefined scale) to the matcher analyzed, thus increasing the suitability level of the approach in question (cf. Fig. 8.17).

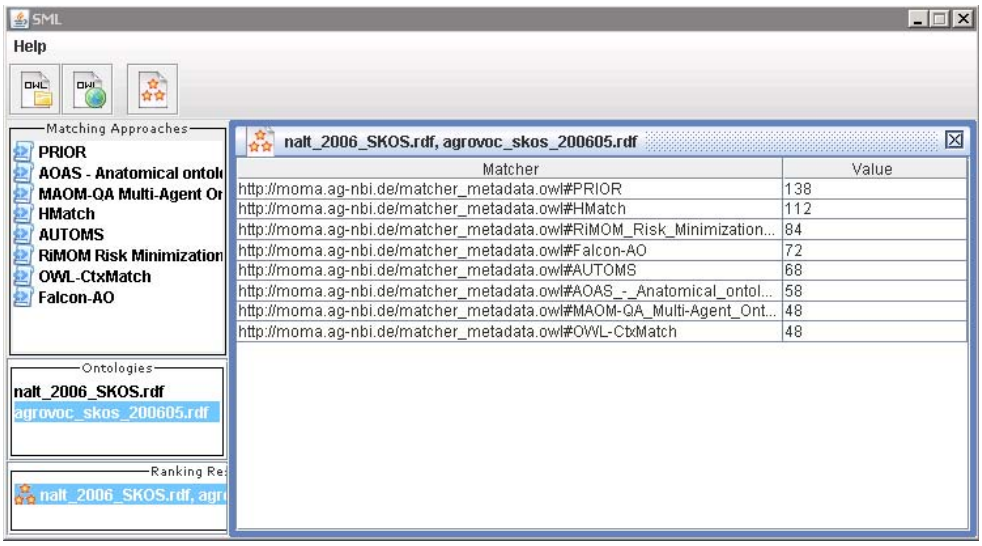


Figure 8.17: Food test case: results achieved by the rule-base MOMA Framework (screenshot)

Comparison: Rule-based methodology vs. OAEI 2006 Campaign

With the food test case findings (cf. Fig 8.17) in mind, we find that the outcomes of the MOMA evaluation are similar to results achieved by the OAEI2006 regarding precision. The order of PRIOR and ISLab HMatch are equal to that of the OAEI2006 order, while only the MOMA order of Falcon and RiMOM diverge from their OAEI ranking(cf. Tab. 8.9).

Even though the OAEI organizers declared that evaluating a sample large enough to lead to significant results for recall was not feasible, we compare our results with recall outcomes of the OAEI Campaign; when comparing the rule-based outcomes

Results - Food test case				
OAEI Campaign 2006			Rule-based MOMA	
Approach	**Precision**	**Ranking**	**Approach**	**Ranking**
Falcon	0.83	1	-	-
RiMOM	0.81	2	-	-
PRIOR	0.71	3	PRIOR	1
ISLab HMatch	0.61	4	ISLab HMatch	2
-	-	-	RiMOM/Falcon	3
-	-	-	RiMOM/Falcon	4
-	-	-	AUTOMS	5
-	-	-	NIH (AOAS)	6
-	-	-	DSSim (MAOM)	7
-	-	-	OWL CtxMatch	8

Table 8.9: Food task: OAEI 2006 Campaign vs. MOMA-based results (precision)

with the OAEI results regarding recall, our results seems to be even more similar (in comparison to the precision results of the OAEI2006), as shown in Tab. 8.10. When the Falcon, ISLab HMach, and PRIOR systems were run, the same recall values were arrived at; they were ranked 2, 3 and 4 at the same time. In our approach, the PRIOR achieved first place, ISLab HMatch second and Falcon system third in the ranking. The only major difference is reflected in the order of the RiMOM system, since when the rule-based approach was used, it ranked third (together with Falcon), but came out first in the OAEI contest.

Results - Food test case				
OAEI Campaign 2006			Rule-based MOMA	
Approach	**Recall**	**Ranking**	**Approach**	**Ranking**
RiMOM	0.50	1	-	-
Falcon/ISLab HMatch/PRIOR	0.46	2	PRIOR	1
Falcon/ISLab HMatch/PRIOR	0.46	3	ISLab HMatch	2
Falcon/ISLab HMatch/PRIOR	0.46	4	RiMOM/Falcon	3
-	-	-	RiMOM/Falcon	4
-	-	-	AUTOMS	5
-	-	-	NIH (AOAS)	6
-	-	-	DSSim (MAOM)	7
-	-	-	OWL CtxMatch	8

Table 8.10: Food task: OAEI 2006 Campaign vs. MOMA-based results (recall)

However, if we take into account that, with eight possible matching approaches in the MOMA system presented in OAEI Campaign 2006, the first four in our ranking are also the same four that achieved the best results in the campaign, in our opinion, the MOMA results can be evaluated as very promising.

8.5.3 Lessons learned

The application of the rule-based MOMA Framework to two case studies from the OAEI Campaign 2006 in the case study evaluation methodology attests to the fact that the matcher selection, based on background information detailing the available matching approaches and the properties of the input ontologies, together with explicit knowledge concerning the dependencies between these algorithms and the incoming sources on which they can operate, delivers very promising results. These results can serve as a basic module for the further examination of the algorithms associated with their execution.

8.6 Further Findings on MOMA and its Evaluation

In order to complete the evaluation of the MOMA Framework, we also compare the results achieved by the manual MOMA Framework with the outcomes provided by the rule-based MOMA mode. We discuss the different results achieved by the two MOMA modis and explain the reasons for the differences. Furthermore, as we wish to contribute to resolving the data integration and interoperability issue and support the dissemination of the Semantic Web vision within the real-world business context, we briefly analyze the impact of the usage of the MOMA Framework on the costs related to the ontology engineering process. Summarizing, we conclude with the entire evaluation and draw conclusions on the results achieved.

8.6.1 Manual vs. (semi-)automatic Approach

When we consider the findings from the manual and (semi-)automatic matcher selection approaches regarding the particular test case, as described in the latter sections, we notice that the outcomes of the two MOMA modis differ from one another. Furthermore, we have concluded that the manual results are very similar to the rankings of the OAEI Campaign, which were based on the real execution of the tested matchers, while the outcomes of the rule-based approach are indeed similar but not as accurate as the manual results (cf. Tab. 8.11, 8.12, and 8.13).

Results - Food test case		
OAEI 2006 (recall)	**AHP-based**	**Rule-based**
RiMOM		-
ISLab HMatch/Falcon/PRIOR	ISLab HMatch	PRIOR
ISLab HMatch/Falcon/PRIOR	Falcon	ISLab HMatch
ISLab HMatch/Falcon/PRIOR	RiMOM/PRIOR	RiMOM/Falcon
-	RiMOM/PRIOR	RiMOM/Falcon

Table 8.11: Food test case: Comparison of all results (recall)

Results - Food test case		
OAEI 2006 (precision)	**AHP-based**	**Rule-based**
	ISLab HMatch	
Faclon	Falcon	-
RiMOM	RiMOM/PRIOR	
PRIOR	RiMOM/PRIOR	PRIOR
ISLab HMatch		ISLab HMatch
-		RiMOM/Falcon
-		RMOM/Falcon

Table 8.12: Food test case: Comparison of all results (precision)

Results - Anatomy test case		
OAEI 2006 (terms mapped)	**AHP-based**	**Rule-based**
NIH (AOAS)		-
ISLab HMatch	ISLab HMatch	PRIOR
PRIOR	PRIOR	ISLab HMatch
Falcon	Falcon	Falcon/RiMOM
-	RiMOM	AUTOMS
-	NIH (AOAS)	NIH (AOAS)

Table 8.13: Anatomy test case: Comparison of all results

At this point, we must admit that we had expected diversity in the ranking results and would have been rather surprised if the results had been identical, since the manual approach is more detailed than its (semi-)automatic counterpart. With the manual approach, the developers specify the requirements of the given application and have an extensive pallet of information relevant to the matcher selection:

- They have thoroughly analyzed the situation in which the matching approach is to be applied.

- They know for which kind of application or task the matcher is be to utilized.
- They can specify in great detail (considering input category, model, type, structure, etc.) the ontological sources to be matched, which, in turn, means that they know the kind of input the matching algorithm is to serve.
- They know exactly what to expect from the suitable matcher (syntactic or semantic structure-based matcher, with or without manual pre- or postprocessing, etc.).
- They can specify the expected output.
- Through the analysis of the application, they have information regarding the matcher usage (usage goal and type, application area, etc.).
- They have specific requirements regarding the availability of the matcher documentation and costs.

In contrast, the (semi-)automatic MOMA approach possesses only data regarding the input sources (ontology metadata), which has been automatically generated using the information from the particular ontology and additional sources, and data regarding the characteristic of the particular approaches; the automatically generated data contains less information. This situation results in a less defined ontology metadata that cannot be compared with the more precise data provided by the developers after requirements analysis. Furthermore, the detailed context of the application in which the matcher is to be applied and the extensive requirements regarding such issues as matcher processing or execution parameters (e.g. application of additional resources) are missing in the rule-based approach. Since, in the rule-based approach, the framework is incapable of automatically generating the information regarding the desired output, this kind of data is also absent from this MOMA mode. Furthermore, in the food test evaluation, we noticed that the rule-based MOMA Framework seems to deliver results that are based on a mix between high precision and high recall, while the AHP-based MOMA Framework delivers ranking more oriented on the recall order of the considered approaches.

In conclusion, we can state that, in the manual MOMA approach, we obtain much more detailed information that is either not as extensive or is simply missing in the rule-based MOMA. This, in turn, means that, in the manual MOMA, we are able to determine suitable matchers more precisely, while in the (semi-)automatic MOMA, we are only able to provide an orientation or recommendation for further analysis.

8.6.2 MOMA Framework and Costs Issues

As stated in [39], integration and interoperability grow rapidly when organizations are forced to share information and knowledge; they devote much of their resources to deal with these issues. The information and knowledge are organized and used in various ways and should be integrated with the use of semantic technologies like common ontologies and ontology matching process, which, in turn, means the need for application of support tools and methodologies like MOMA. Considering the different stakeholder groups of the Semantic Web, we have determined that especially four of five groups could need or be potentially interested in ontology matching issues and MOMA Framework (cf. Fig. 8.18):

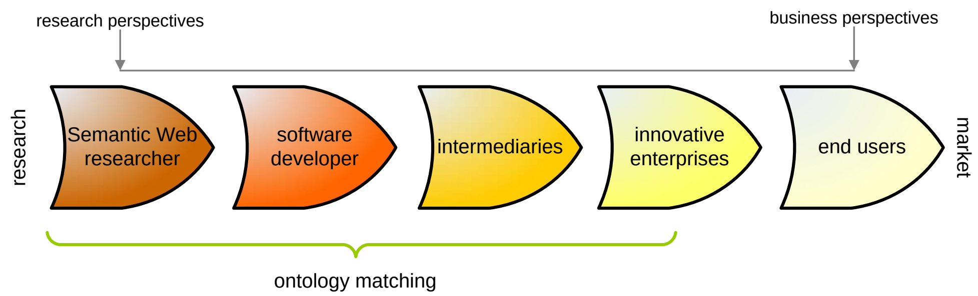


Figure 8.18: Stakeholders of Semantic Web [39] and the ontology matching interested among them

Semantic Web researchers are developing and innovating theories and methods for the Semantic Web. They need tools and methods that support the ontology matching process and, in turn, facilitate their research work on scalability, heterogeneity, and dynamics issues.

Software developers require aid in the development process of the Semantic Web applications, matching recommendations, and tools to facilitate their work.

Intermediaries interested in transferring technology and knowledge from researchers and developers to users (practitioners) can promote the Semantic Web vision mainly on the basis of existing and tested methods, approaches, and tools.

Innovative enterprises interested in grasping new opportunities from the Semantic Web and developing new business models need tool support since this would help them arrange and conduct their everyday business.

Even, as stated in [39], if some companies are starting to test the semantic technologies, they still consider Semantic Web technologies to be a highly challenging issue. Furthermore, current Semantic Web research often ignores some aspects of engineering processes that are fundamental in real-world business contexts, like cost issues, means to monitor the business value, and the impact of these technologies in terms of organization. In the context of the MOMA application, when we talk about costs we should consider our ONTOlogy COst Model (ONTOCOM) [170] — a parametric cost model for the estimation of the effort involved in building, reusing, and maintaining ontologies. ONTOCOM uses different types of cost drivers [171]:

- personnel-related cost drivers that reflect the role of team experience, ability, and continuity w.r.t. the effort invested in the engineering process;
- product-related cost drivers that account for the impact of the characteristics of the product to be engineered (i.e. the ontology) on the overall costs; and
- process(project)-related cost drivers that relate to overall characteristics of an ontology engineering process and their impact on the total costs.

The particular cost drivers within these groups have a rating level (from very low to very high and linked to a weight — effort multiplier) to express their impact on the development effort.

One of the personnel-related cost drivers that the matching approaches affect is the *tool experience (TEXP)*, which measures the level of experience of the project team w.r.t. the ontology management tools. The less tool experience (also experience with the matching approaches) a project team has, the higher the effort multiplier will be and, in turn, the greater the expenditure required for the ontology development. In this regard, since not every matcher can do the matching job in question, developers wishing to apply the existing approaches are confronted with the dilemma as to which matching algorithm to choose. Without the MOMA Framework, they would have to "test" the suitability of existing approaches by trial and error, which is extremely time consuming; without MOMA, developers would have to invest much more time to arrive at an overview and check the suitability of existing approaches. Conversely, with MOMA, we can lend support to researchers, software developers, intermediaries, and innovative enterprises (cf. Fig. 8.18), who posses little expertise in ontology engineering during the ontology engineering process as well as during the development of particular ontology-based applications. They can more quickly understand their requirements with regard to the matching algorithm, and find suitable candidates. Thus, we can positively affect the TEXP driver, lower its effort multiplier, and contribute to the reduction of development costs.

The other cost driver relevant for the matching issue instrumental in the ontology reuse process is the *ontology integration (OI)* driver; it accounts for the additional efforts required to integrate or merge multiple ontologies into a target ontology. Also in this case, the more difficult the integration and merging of source ontologies is, the higher the effort multiplier will be and, in turn, the greater the expenditure required for the ontology development. Since matching algorithms is the key issue in almost every phase of an ontology engineering and management process as well as in ontology merging, our framework facilitating the selection of suitable matchers contributes to the cost reduction regarding the OI cost driver. Furthermore, the systematic application of existing matchers provides the test cases for a particular domain, application, or task, which then revises the potential of a single approach and enhances the quality of an algorithm, and, in turn, (indirectly) positively impacts the OI driver.

Even if it is a non-trivial task to estimate how much time and effort would be saved by applying the MOMA Framework versus the time and effort, that need to be spent to deploy and execute the different algorithms, we can assume, considering the MOMA evaluation results, that the tool and methodological support for the matcher selection process can reflect positively on the overall development effort and costs needed for the implementation of Semantic Web-based applications.

8.6.3 Final remarks on evaluation

Since the aim of the MOMA Framework is to merely suggest appropriate and (potentially) suitable matching algorithms for the matching of a given set of sources, we do not expect to achieve exactly the same ranking results generated by the MOMA Framework as in the case of approach ranking based on real execution of matchers. We simply aim to deliver a set of recommendations for future analysis, to reduce the number of candidates that can be considered for application, and recommend matchers for the main focus of reuse strategy specifically for a given set of ontological sources. While the evaluation of the MOMA Framework within the context of the two test cases from the OAEI 2006 Campaign reveals that matcher candidates can be ranked according to their effective suitability, both methodologies can facilitate, for instance, researchers and software/Semantic Web-based application developers with little expertise in Ontological Engineering and the alignment domain to obtain an overview of existing and, in particular, the most suitable matchers w.r.t their applications requirements. Considering the test cases where both AHP- and rule-based modes have been applied, the advantage is that, in each case, we were able to restrict the number of potentially useful matchers and thus simplify and speed up the selection of the suitable algorithms. Through the shortening of the search process and selection of suitable matchers, we can reduce the costs of the application development, which is ultimately one of the most important aspects in the development of the Semantic Web. Finally, with our MOMA Framework, we aim to support the Semantic Web vision, since every effort that helps to simplify and reduce time and costs within the ontology engineering process and development of the Semantic Web-based applications is one step closer on the long road to the acceptance of the semantic technologies by wide industry.

Chapter 9

Conclusion

Abstract In this chapter, we summarize the results of our research work by highlighting the main contributions, discussing the results achieved, and sketching their impact on the Semantic Web community. We also outline some possible research directions for further development connected with our MOMA Framework. Finally, since our framework can be seen as a general metadata-based selection framework, we briefly discuss its possible application in the context of other selection issues and outline the corresponding required adaptations.

"It would be possible to describe everything scientifically, but it would make no sense; it would be without meaning, as if you described a Beethoven symphony as a variation of wave pressure."

Albert Einstein

9.1 Summary

"The important thing in science is not so much to obtain new facts as to discover new ways of thinking about them."

Sir William Bragg (1862 - 1942)

In this thesis, we have discussed the current situation in the ontology matching domain with particular emphasis on the significant role of matching approaches in the realization of the Semantic Web vision. We have analyzed the main open issues and considered what kind of methodological and tool support is needed to cover the gaps. As a possible solution to close the open issues, and to align to the global Semantic Web, we have presented the **M**etadata-based **O**ntology **MA**tching Framework – MOMA Framework – that evaluates and ranks matching approaches according to their suitability for a given set of ontological sources. MOMA takes into account the capabilities of existing matchers and suggests appropriate approaches for individual cases without (prior to) the matchers' execution. MOMA Framework is based on the **M**ultilevel **C**haracteristic for **M**tching **A**pproaches (MCMA) – a model developed to specify the matching approaches, their incoming ontological sources, and the application features in which the matcher is to be applied. The heart of the MOMA Framework is the selection engine which, depending on the mode, manual or (semi-)automatic, conducts the matcher determination process based on the AHP-approach or predefined rule set, respectively.

For evaluation of both MCMA and the MOMA selection modes, we applied the qualitative approaches since these methods emphasize an interpretive approach that uses data to both pose and resolve research questions. However, depending on the context and objects to be evaluated, we have utilized different qualitative evaluation methods. The evaluation starts with the expert-based evaluation of the MCMA, which resulted in refinement of the preliminarily defined characteristic and ended in a revised MCMA. The revised MCMA serves as a basis for both the AHP- and rule-based approaches for matcher selection. Since the main aspects of the MOMA Framework evaluation are directly related to its application in different situations and to the accuracy of its predictions, both MOMA selection modes have been evaluated in the course of the case study evaluation; we have applied our MOMA Framework to the case studies selected from the OAEI Campaign. While the application of the MOMA Framework in these case studies has shown that matcher candidates can be ranked according to their effective suitability prior to their execution, MOMA can facilitate, for instance, domain experts or Semantic Web-based application developers with little expertise in Ontological Engineering and the alignment domain to obtain an overview of existing and, in particular, the most suitable matchers w.r.t their applications requirements; the MOMA Framework reduces the number of candidates that can be considered for application and recommends matchers for the main focus of reuse strategy specifically for a given set of ontological sources.

Concluding, we can say that, with our MOMA Framework, we 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 Semantic Web technologies. Our framework contributes to data integration and interoperability by maintaining awareness of the link between matching algorithms and a wide variety of ontologies. It is the first step towards the reuse of existing ontology matching approaches that supports the more optimal utilization of ontology matching tasks as envisioned by the Semantic Web community, addresses the issues of matchers heterogeneity, exploits the valuable ideas embedded in current matching approaches, and supports developers by giving them recommendations regarding suitable matcher solutions. Furthermore, we would once more like to highlight that the MOMA Framework has a share in solving some of the main open issues within the ontology matching domain; it contributes to the "no overarching matching", "missing infrastructure" and "evil's diversity" issues and, to some measure, has addressed the problems related to "unused" reuse and "holes" in the approaches.

9.2 Further applications

As mentioned in Sec.4.2, the presented framework can be seen as a general selection framework; it is not limited to matcher detection but, after corresponding changes, it can also be applied for further selection issues. Our metadata-based selection framework could be applied in both simple and complex selection cases; however, utilization of the framework in the case of simple selection issues, e.g. when decision depends on just a few criteria and/or scales are consistent, could be an oversized issue for such a situation.

Staying in the Semantic Web world, an eligible example for the application of our metadata-based selection framework could be in obtaining a particular ontology network life cycle. According to [177], an ontology network life cycle is defined as the project-specific sequence of activities created by mapping the activities identified in the ontology network development process onto a selected ontology network life cycle model (ONLCM). ONLCM, in turn, is defined as the framework selected by each organization on which to map the activities identified in the ontology network development process to produce the ontology network life cycle. As stated in [178], there are five main steps to establish the ontology network life cycle:

- Step #1: Identify ontology network development requirements. Ontology developers identify the main needs of the ontology network development.
- Step #2: Select the ontology network life cycle model (ONLCM) to be used. The main question is: Which ontology network life cycle model should be chosen?
- Step #3: Select activities to be carried out. The main question is: Which activities are to be carried out?
- Step #4: Map the selected activities into the selected ontology network life cycle model.
- Step #5: Set the order of the activities: the result is the ontology network life cycle for the ontology network.

Since Steps #2 & #3 involve different selection activities, we have considered appling our selection framework for these issues (cf. Fig. 9.1).

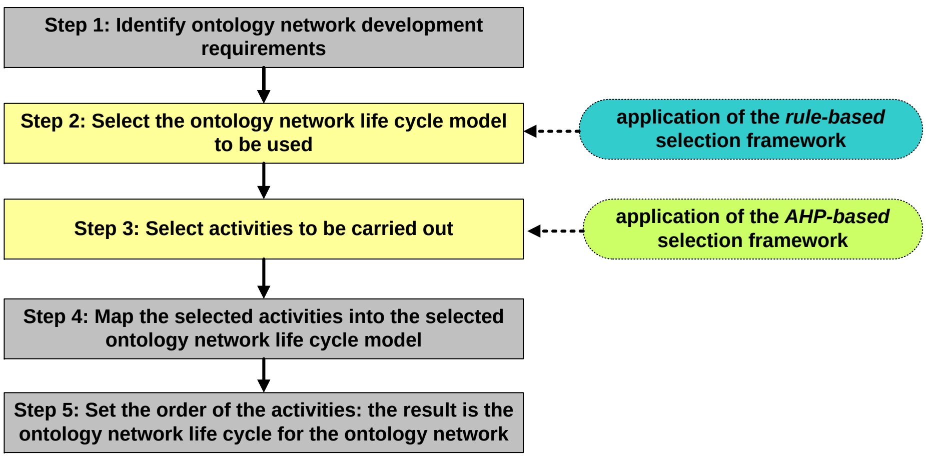


Figure 9.1: Steps for establishing the ontology network life cycle [178] and the application of the MOMA Framework

Step #2

Since, in Step #1, the ontology developers have identified the main needs of the ontology network development, this information will be ontologised and serves as a basis for the selection process. To be able to decide which ONLCM is appropriate for given ontology network development requirements, our metadata-based selection framework also needs information regarding the existing ONLCMs (ONLCMs metadata) which will be annotated in an ontological form. In [178], a decision regarding the suitability of ONLCMs is made by the application of an informal decision tree. Applying our framework, we would exchange the decision tree approach by the execution of rule statements whereby the required rules would be derived from the information already available in (questions defined for) the decision tree. In the end, as shown in Fig. 9.2, the "Ontologies & Matchers" part of the MOMA Framework (cf. Fig. 7.2) has been replaced by the "Requirements & Models", the "ontology and matcher metadata" in the Knowledge Base has been exchanged with the "ONCLMs metadata and ontologiesed requirements", the content of rule repository has been changed, and, in turn, instead of a ranked list of a matchers, the selection engine delivers a ranked list of ONCLMs. In general, we can state that, to apply the rule-based part of the framework in the context of a new problem, the "old" metadata and "old" rule repository must be exchanged considering the given situation (the appropriate metadata must be available and rule statements must be defined).

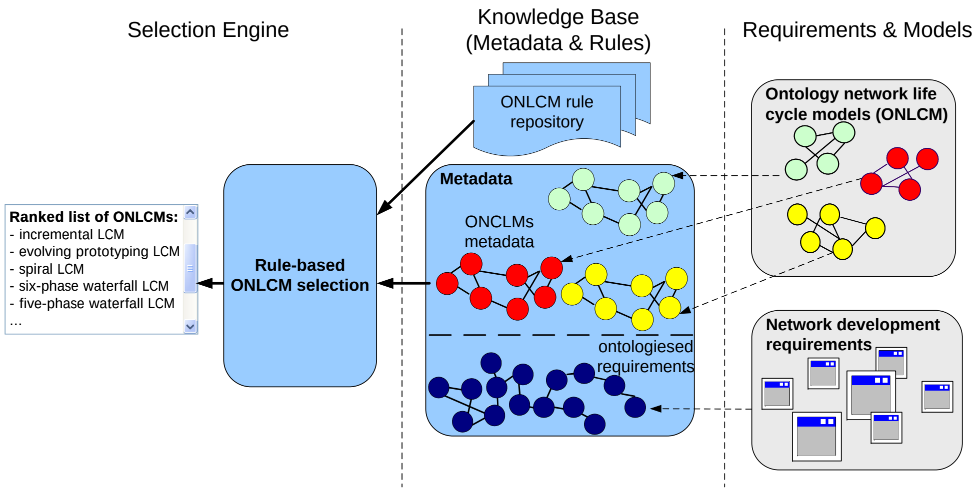


Figure 9.2: Rule-based framework in the context of the ONLCM selection

Step #3

In [178], this step is intended to be solved using the natural language questions. For this issue, we propose supporting the questions with the AHP-based part of our framework. The application of the AHP-based selection framework helps ontology developers to set priorities and to make the best decision regarding the possible activities. A general rule for the application of the AHP-based part of the selection framework is the building of an appropriate AHP hierarchy structure (cf. Fig. 6.4 and 6.5), which is specific for the objectives of a given problem and has to be adapted according the given criteria and the alternative solutions in question.

On the example of the selection of an ontology network life cycle suitable for a particular case, we have shown that our framework can be seen as a general selection framework able to serve different selection issues and/or different domains. Further application of the MOMA Framework could, for instance, consider the selection of the suitable MultiCriteria Decision Analysis approach, taking into account the predefined guidelines (cf. Sec. 6.1) or the selection of the appropriate evaluation methodology (cf. Sec. 8.2). Depending on the available information and requirements regarding the human involvement, the rule-based mode, AHP-based mode, or both parts of the framework may be utilized.

9.3 Future Work

In Sec. 1.2.3, we have outlined the main contributions of our research work: method, model and instantiation. In this section, we would like to come back to them and briefly discuss some possible directions that would continue the research conducted in this thesis:

method In the global Semantic Web, data and service providers should be capable of systematically publishing Web-wide their resources, be they ontologies or new matching algorithms. In so doing, they should also provide the descriptive metadata for the subscribed resources, as this will guarantee a higher visibility and usability of their products in terms of incoming inquiries. This would enhance the process of matcher selection based on the automatic gathering of information. In this context, a further step in the strategy that strives for an optimization of the matching process through the selection of existing matchers could be dedicated to another important topic that we have not yet addressed in the thesis - the matching execution. This step would utilize the results of our MOMA; it would consider the best selected matchers and apply them in the context of given application. This, in turn, could, for example, require an automatic composition of the candidate matching services to achieve the desired results.

model Regarding the model contribution, some pragmatic (e.g. authoring and historical data of an ontology) and further syntactic features (e.g. average path length) could be analyzed mainly in the context of the rule-based selection and its ontology metadata. Such features are not crucial characteristics to matcher selection; however, in future development, we could imagine analyzing, for instance, versions or engineering process issues.

instantiation From the scientific perspective, an automatic extraction of matcher metadata (instead of the manual annotation of matchers through the developed questionnaire) for the rule-based matcher selection derived, for instance, from the OAEI Campaign results could be a very interesting and promising approach. This would, on the one hand, speed up the further collection of matcher metadata and, on the other, it would allow for the easier growth of the MOMA Knowledge Base, which, in turn, would increase the usefulness and benefit of the MOMA Framework. Regarding the automatic ontology metadata generation, we could consider exchanging the OntoMeta approach utilized so far with the new Ontology MEtadata GenerAtion (OMEGA) approach that was presented during the 16th International Conference on Knowledge Engineering and Knowledge Management Knowledge Patterns (EKAW2008). As stated in [187], OMEGA generates metadata about arbitrary ontologies on the Web in an automatic manner, and since it has been evaluated in terms of coverage, precision, recall, and the overall user-perceived quality with positive results, it could improve the quality and amount of generated information, and in turn, the accuracy of the rule-based MOMA results. Furthermore, the extension of the rule repository, especially considering the in model integrated heuristic and syntactic features with specific performance and accuracy parameters, could have a positive impact on the MOMA ranking results and increase their accuracy. From the very pragmatic view, the future implementation work could also be dedicated to the development of a web service-based MOMA access to allow a broader access especially to the (semi-)automatic MOMA service.

Considering the future work, we cannot omit the time and effort issues and connected with it, the already mentioned cost factors. It would be interesting to analyze how much time and effort could be saved though the application of the MOMA Framework in contrast to the time and effort spent during the deployment and execution tests of the existing matching approaches.

Bibliography

[1] Rohit Aggarwal. Semantic Web Services Languages and Technologies: Comparison and Discussion. LSDIS Lab, University of Georgia, http://lsdis.cs.uga.edu/lib/download/aggarwal_essay_2004.pdf, accessed on 07.10.2008, 2004.

[2] David Aumüller, Hong-Hai Do, Sabine Maßmann, and Erhard Rahm. Schema and Ontology Matching with COMA++. In *Proceedings 24th International Conference on Management of Data (SIGMOD), Software Demonstration*, pages 906–908, Baltimore (MD US), 2005.

[3] Tim Berners-Lee. What the Semantic Web can represent. http://www.w3.org/DesignIssues/RDFnot.html, accessed on 17.08.200, September 1998.

[4] Tim Berners-Lee. Semantic Web – XML2000. http://www.w3.org/2000/Talks/1206-xml2k-tbl/Overview.html, accessed on 05.09.2008, 2000.

[5] Tim Berners-Lee. Semantic Web on XML. http://www.w3.org/2000/Talks/1206-xml2k-tbl/, accessed on 18.08.2008, 2000.

[6] Tim Berners-Lee. The Semantic Web. http://www.technologyreview.com/player/video/video_compact_leader.aspx?bcpid=79489195&bclid=60818931&bctid=708783761, accessed on 29.08.2007, 2007.

[7] Tim Berners-Lee and Mark Fischetti. *Weaving the Web : The Original Design and Ultimate Destiny of the World Wide Web by its Inventor*. Harper San Francisco, September 1999.

[8] Tim Berners-Lee, James Hendler, and Ora Lassila. The Semantic Web. *Scientific American*, 284(5):34–43, May 2001.

[9] Wayne L. Bethea, Clayton R. Fink, and John S. Beecher-Deighan. JHU/APL Onto-Mapology Results for OAEI 2006. In *Proceedings of the International Workshop on Ontology Matching (OM-2006) collocated with the 5th International Semantic Web Conference (ISWC-2006)*, volume 225 of *CEUR Workshop Proceedings*, pages 144 – 152, November 2006.

[10] Nigel Bevan. Quality in Use: Meeting User Needs for Quality. *Journal of System and Software*, 49(1):89–96, 1999.

[11] Navneet Bhushan and Kanwal Rai. *Strategic Decision Making: Applying the Analytic Hierarchy Process*. Springer, 2004.

[12] Yaser Bishr. Overcoming the semantic and other barriers to GIS interoperability. *International Journal of Geographical Information Science*, 12(4):299–314, June 1998.

[13] Christian Bizer, Ralf Heese, Malgorzata Mochol, Radoslaw Oldakowski, Robert Tolksdorf, and Rainer Eckstein. The Impact of Semantic Web Technologies on Job Recruitment Processes. In *Proceedings of the 7th Internationale Tagung Wirtschaftsinformatik 2005*, pages 1367–1383, 2005.

[14] Elena Paslaru Bontas. Context-enhanced Ontology Reuse. In *Doctoral Consortium at the 5th International and Interdisciplinary Conference on Modeling and Using Context (CONTEXT05)*, 2005.

[15] Elena Paslaru Bontas. Using Context Information to Improve Ontology Reuse. In *Proceedings of the Doctoral Consortium at the CAISE'05*, 2005.

[16] Elena Paslaru Bontas and Malgorzata Mochol. Towards a reuse-oriented methodology for ontology engineering. In *Proceedings of 7th International Conference on Terminology and Knowledge Engineering (TKE 2005)*, pages 175 – 188, 2005.

[17] Elena Paslaru Bontas, Malgorzata Mochol, and Robert Tolksdorf. Case Studies on Ontology Reuse. In *Proceedings of the 5th International Conference on Knowledge Management (I-Know'05)*, pages 345 – 353, June 2005.

[18] Elena Paslaru Bontas, Sebastian Tietz, Robert Tolksdorf, and Thomas Schrader. Generation and Management of a Medical Ontology in a Semantic Web Retrieval System. In *Proceedings of the CoopIS/DOA/ODBASE (1)*, volume 3290/2004, pages 637–653, 2004.

[19] Elena Paslaru Bontas, Robert Tolksdorf, and Thomas Schrader. Ontology-based Knowledge Organization in a Semantic Web for Pathology. In *Porc. of the 4th International Conference on Knowledge Management (I-Know'04)*, 2004.

[20] Gary D. Borich. Evaluation models: A Question of Purpose Not Terminology. *Educational Evaluation and Policy Analysis*, 5(1):61–63, 1983.

[21] Pieter W.G. Bots and Jose'e A.M. Hulshof. Designing multi-criteria decision analysis processes for priority setting in health policy. *Journal of Multi-Criteria Decision Analysis*, 9:56–75, 2000.

[22] Steve Bratt. Semantic Web, and Other Technologies to Watch. `http://www.w3.org/2007/Talks/0130-sb-W3CTechSemWeb`, accessed on 18.08.2008, January 2007.

[23] Britannica. Encyclopedia Britannica Online. `http://http://original.britannica.com/`, accessed on 25.09.2008, 2000.

[24] John Buchanan and Phil Sheppard. Ranking Projects Using the ELECTRE Method. In *Proceedings of the 33rd Annual Conference of the Operational Research Society of New Zealand*, `http://www.orsnz.org.nz/conf33/papers/p58.pdf`, accessed on 16.05.2008, 1998.

[25] Silvana Castano, Alfio Ferrara, and Gianpaolo Messa. Results of the HMatch Ontology Matchmaker in OAEI 2006. In *Proceedings of the International Workshop on Ontology Matching (OM-2006) collocated with the 5th International Semantic Web Conference (ISWC-2006)*, volume 225 of *CEUR Workshop Proceedings*, pages 134–143, November 2006.

[26] Silvana Castano, Alfio Ferrara, and Stefano Montanelli. Methods and Techniques for Ontology-based Semantic Interoperability in Networked Enterprise Contexts. In *Proceedings of the 1st CAiSE INTEROP Workshop On Enterprise Modelling and Ontologies for Interoperability (EMOI - INTEROP 2004)*, pages 261–264, June 2004.

[27] Silvana Castano, Alfio Ferrara, and Stefano Montanelli. Matching Ontologies in Open Networked Systems: Techniques and Applications. *Journal on Data Semantics (JoDS)*, V:25–63, 2006.

[28] Silvana Castano, Alfio Ferrara, Stefano Montanelli, and G Racca. Matching Techniques for Resource Discovery in Distributed Systems Using Heterogeneous Ontology Descriptions. In *Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'04)*, pages 360–366. IEEE Computer Society, 2004.

[29] B. Chandrasekaran, John R. Josephson, and V. Richard Benjamins. What Are Ontologies, and Why Do We Need Them? *IEEE Intelligent Systems and Their Applications. Special Issue on Ontologies*, 14(1):20–26, 1999.

[30] Kevin Chen-Chuan Chang, Bin He, and Zhen Zhang. Toward Large Scale Integration: Building a MetaQuerier over Databases on the Web. In *Proceedings 2nd Biennial Conference on Innovative Data Systems Research (CIDR)*, pages 44–55, September 2005.

[31] Yao Chen. Automatic Generation of Ontology Metadata. Diploma Thesis, Freie Universität Berlin, April 2006.

[32] Paul Clements, Felix Bachmann, Len Bass, David Garlan, James Ivers, Reed Little, Robert Nord, and Judith Staffor. *Documenting Software Architectures: Views and Beyond*. Addison-Wesley Professional, 2002.

[33] William W. Cohen. Integration of Heterogeneous Databases Without Common Domains Using Queries Based on Textual Similarity. In *Proceedings 17th International Conference on Management of Data (SIGMOD)*, pages 201–212, 1998.

[34] Royal Eugene Collins. *Mathematical Methods for Physicists and Engineers: Second Corrected Edition*. Dover Publications, 2 edition, January 1998.

[35] Andrew M. Colman. *A Dictionary of Psychology*. Oxford University Press, February 2007.

[36] World Wide Web Consortium. Semantic Web. `http://www.w3.org/2001/sw/`, accessed on 17.08.2008, 2004.

[37] Edward Craig. *The Shorter Routledge Encyclopedia of Philosophy*. Routledge, 2005.

[38] Isabel F. Cruz and Huiyong Xiao. Using a Layered Approach for Interoperability on the Semantic Web. In *Proceedings of the 4th International Conference on Web Information Systems Engineering (WISE 2003)*, pages 221–232, December 2003.

[39] Roberta Cuel, Alexandre Delteil, Vincent Louis, and Carlo Rizzi. Knowledge Web Technology Roadmap: The Technology Roadmap of the Semantic Web. Technical report, Knowledge Web EU Network of Excellence, Decemper 2007.

[40] Jonathan Dancy and Ernest Sosa, editors. *A Companion to Epistemology*. Blackwell Companions Philosophy. Blackwell Publishing Limited, 1994.

[41] Lois-Ellin Datta, Linda G. Morra, and Amy C. Friedlander. *Program Evaluation and Methodology Division. Case Study Evaluations*. United States General Accounting Office, November 1990.

[42] Stefan Decker, Sergey Melnik, Frank van Harmelen, Dieter Fensel, Michel C. A. Klein, Jeen Broekstra, Michael Erdmann, and Ian Horrocks. The Semantic Web: The Roles of XML and RDF. *IEEE Internet Computing*, 4(5):63–74, 2000.

[43] Ying Ding, Dieter Fensel, and Hans-Georg Stork. The Semantic Web: from Concept to Percept. *Austrian Artificial Intelligence Journal*, 21(4):34–43, 2003.

[44] Hong-Hai Do, Sergey Melnik, and Erhard Rahm. Comparison of Schema Matching Evaluations. In *Proceedings of the 2nd International Workshop on Web Databases (German Informatics Society)*, volume 2593 of *Lecture notes in computer science*, pages 221–237, `http://www.izbi.uni-leipzig.de/izbi/publikationen/publi_2002/do_showDoc.pdf`, accessed on 18.08.2008, 2002.

[45] Hong-Hai Do and Erhard Rahm. COMA – A System for Flexible Combination of Schema Matching Approaches. In *Proceedings of the 28th International Conference on Very Large Data Bases (VLDB)*, pages 610–621, 2002.

[46] An-Hai Doan, Pedro Domingos, and Alon Halevy. Reconciling Schemas of disparate Data sources: A Machine Learning Approach. In *Proceedings of the 20th International Conference on Management of Data (SIGMOD)*, pages 509–520, 2001.

[47] An-Hai Doan, Jayant Madhavan, Pedro Domingos, and Alon Halevy. Ontology Matching: A Machine Learning Approach. In Steffen Staab and Rudi Studer, editors, *Handbook on Ontologies*, pages 385–516. Springer Verlag, 2004.

[48] Asuman Dogac, Gokce Laleci, Yildiray Kabak, and Ibrahim Cingil. Exploiting Web Service Semantics: Taxonomies vs. Ontologies. *IEEE Data Engineering Bulletin*, 25(4), 2002.

[49] Merlin Dorfman and Richard H. Thayer. *Standards, Guidelines, and Examples on System and Software Requirements Engineering*. IEEE Computer Society Press, 1990.

[50] Dejing Dou, Drew McDermott, and Peishen Qi. Ontology Translation on the Semantic Web. In *Proceedings of the AInt'l Conf. on Ontologies, Databases and Applications of Semantics (ODBASE2003)*, pages 952–969, 2003.

[51] Edd Dumbill. Building the Smenatic Web. `http://www.xml.com/pub/a/2001/03/07/buildingsw.html`, accessed on 18.08.2008 (article adapted from the closing keynote delivered at Knowledge Technologies 2001), March 2001.

[52] Carlos A. Bana e Costa, Jean-Marie De Corte, and Jean-Claude Vansnik. The MACBETH Approach. Technical Report LSEOR 03.56, Operational Research Group, London School of Economics, 2003.

[53] Jody Eisen, Pat Gallagher, Graydon L. Karlson, Thomas L. Saaty, Marcia Schwab, Jaime Silva, and Wendy Webb. Forecasting the Future of The Soviet Union. *Models, Methods, Concepts & Aplications of the Analytic Hierarchy Porcess*, pages 219–232, 2000.

[54] Jérôme Euzenat, Thanh Le Bach, Jesús Barrasa, Paolo Bouquet, Jan De Bo, Rose Dieng, Marc Ehrig, Manfred Hauswirth, Mustafa Jarrar, Ruben Lara, Diana Maynard, Amedeo Napoli, Giorgos Stamou, Heiner Stuckenschmidt, Pavel Shvaiko, Sergio Tessaris, Sven Van Acker, and Ilya Zaihrayeu. State of the art on current alignment techniques. Technical Report Deliverable 2.2.3, Knowledge Web EU Network of Excellence, 2004.

[55] Jérôme Euzenat, Marc Ehrig, Anja Jentzsch, Malgorzata Mochol, and Pavel Shvaiko. Case-based recommendation of matching tools and techniques. Technical Report Deliverable 1.2.6, Knowledge Web EU Network of Excellence, 2006.

[56] Jérôme Euzenat and Pavel Shvaiko. *Ontology matching*. Springer, 2007.

[57] Jérôme Euzenat and Petko Valtchev. Similarity-based ontology alignment in OWL-lite. In *Proceedings 15th European Conference on Artificial Intelligence (ECAI)*, pages 333–337, 2004.

[58] Kateryna Falkovych, Marta Sabou, and Heiner Stuckenschmidt. UML for the Semantic Web: Transformation-Based Approaches. In *Knowledge Transformation for the Semantic Web*, pages 92–106. B. Omelayenko and M. Klein, 2003.

[59] Barbara Farbey and Anthony Finkelstein. Evaluation in software engineering: ROI, but more than ROI. In *Proceedings of the 3rd International Workshop on Economics-Driven Software Engineering Research (EDSER-3 2001)*, pages 12–19, May 2001.

[60] Dieter Fensel. *Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce*. Springer, Heidelberg (DE), 2nd edition, 2004.

[61] Mariano Fernández-Lopéz and Asunción Gómez-Pérez. Overview and analysis of methodologies for building ontologies. *Knowledge Engineering Review*, 17(2):129–156, 2002.

[62] José Figueira, Slavatore Greco, and Matthias Ehrgott. *Multiple criteria decsion analysis: State of the Art Surveys*. Springer, 2005.

[63] Ernest Forman and Mary Ann Selly. *Decision By Objectives: How to convince others that you are right*. World Scientific Press, 2001.

[64] Xavier Franch and Juan Pablo Carvallo. Using Quality Models in Software Package Selection. *IEEE Computer Society*, 20(1):34–41, 2003.

[65] Frédéric Fürst and Francky Trichet. Axiom-based Ontology Matching. In *Proceedings of the 3rd international conference on Knowledge capture (K-CAP'05)*, pages 195–196, New York, NY, USA, 2005. ACM Press.

[66] Jörg Garbles, Magnus Niemann, and Malgorzata Mochol. A Personalized Hotel Selection Engine. In *Proceedings of the Poster Session of 3rd European Sematnic Web Conference (ESWC) 2006*, 2006.

[67] C. J. Michael Geisterfer and Sudipto Ghosh. Software Component Specification: A Study in Perspective of Component Selection and Reuse. In *Proceedings of the 5th International Conference on COTS Based Software Systems (ICCBSS)*, pages 100–108. IEEE Computer Society Press, February 2006.

[68] Anne Gilliland-Swetland. Setting the Stage. In *Introduction to Metadata, Pathway to Digital Information*. Getty Information Institute, 2000.

[69] Fausto Giuchiglia and Pavel Shvaiko. Semantic Matching. *The Knowledge Engineering Review (KER) Journal*, 18(3):265–280, 2004.

[70] Asunción Gómez-Pérez, Mariano Fernández-Lopéz, and Oscar Corcho-Garcia. Technical Roadmap D.1.1.1. Technical report, OntoWeb, `http://www.ontoweb.org/About/Deliverables/Deliverable111.pdf`, accessed on 19.08.2007, November 2001.

[71] Asunción Gómez-Pérez, Mariano Fernández-Lopéz, and Oscar Corcho-Garcia. *Ontological Engineering: With Examples from the Areas of Knowledge Management, E-commerce and the Semantic Web (Advanced Information and Knowledge Processing)*. Springer Verlag, December 2004.

[72] Michael Grüninger and Mark S. Fox. Methodology for the Design and Evaluation of Ontologies. In *Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing (IJCAI'95)*, 1995.

[73] Tom R. Gruber. A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*, 5(2):199–220, 1993.

[74] Tom R. Gruber. Toward principles for the design of ontologies used for knowledge sharing. *Journal of Human Computer Studies*, 43(5-6):907–928, 1995.

[75] Nicola Guarino. Formal Ontology and Information Systems. In *Proceedings of the International Conference of Formal Ontology in Information Systems (FOIS1998)*, pages 3–15, 1998.

[76] Adel Guitouni and Jean-Marc Martel. Tentative guidelines to help choosing an appropriate MCDA method. *European Journal of Operational Research*, 109:501–521, 1998.

[77] Adel Guitouni, Jean-Marc Martel, and Philippe Vincke. A Framework to Choose a Discrete Multicriterion Aggregation Procedure, 2000.

[78] Tommi Gustafsson, Ahti Salo, and Ramakrishnan Ramanathan. Multicriteria Methods for Technology Foresight. *Journal of Forecasting*, 22:235–255, 2003.

[79] Eugene D. Hahn. Better decisions come from a results-based approach. *Marketing News*, 36(36):22–24, 2002.

[80] Jens Hartmann, Elena Paslaru Bontas, Raúl Palma, and Asunción Gómez-Pérez. DEMO - A Design Environment for Metadata About Ontologies. In York Sure and John Domingue, editors, *The Semantic Web: Research and Applications: 3rd European Semantic Web Conference (ESWC 2006)*, volume 4011/2006 of *Lecture Notes in Computer Science*, pages 427–441. Springer Verlag, 2006.

[81] Frederick Hayes-Roth. Rule-based systems. *Communications of the ACM*, 28(9):921–932, 1985.

[82] Hai He, Weiyi Meng, Clement Yu, and Zonghuan Wu. Automatic Integration of Web Search Interfaces with WISE-Integrator. *The VLDB Journal - The International Journal on Very Large Data Bases*, 13(3):256–273, 2004.

[83] Ralf Heese, Malgorzata Mochol, and Radoslaw Oldakowski. Semantic Web Technologies in the Recruitment Domain. In Miguel-Angel Sicilia, editor, *Competencies in Organizational E-Learning: Concepts and Tools*, pages 299–318. IGI Global, 2006.

[84] Jeff Heflin and James Hendler. Semantic Interoperability on the Web. In *Proceedings of Extreme Markup Languages 2000*, pages 111–120. Graphic Communications Association, 2000.

[85] James Hendler, Tim Berners-Lee, and Eric Miller. Integrating Applications on the Semantic Web. *Journal of the Institute of Electronic Engineers of Japan*, 122(10):676–680, October 2002.

[86] Anne De Piante Henriksen. A technology assessment primer for the management of technology. *International Journal of Technology Management*, 13:615–638, 1997.

[87] Martin Hepp. Ontologies: State of the Art, Business Potential, and Grand Challenges. In Martin Hepp, Pieter De Leenheer, Aldo de Moor, and York Sure, editors, *Ontology Management: Semantic Web, Semantic Web Services, and Business Applications*, pages 3–22. Springer Verlag, 2007.

[88] Alan R. Hevner, Sudha Ram, Salvatore T. March, and Jinsoo Park. Design Science in Information System Research. *MIS Quarterly*, 28(1):75–105, 2004.

[89] Raimo P. Hämäläainen. Facts or values - How do parliamentarians and experts see nuclear power? *Energy Policy*, 19:464–472, 1991.

[90] Laura Hollink, Suzanne Little, and Jane Hunter. Evaluating the Application of Semantic Inferencing Rules to Image Annotation. In *Proceedings of the 1st Asian Semantic Web Conference (ASWC'06)*. ACM Press, 2005.

[91] S. Hong and R. Nigam. Analytic Hierarchy Process Applied to Evaluation of Financial Modeling Software. In *Proceedings of the 1st International Conference on Decision Support Systems*, 1981.

[92] Ian Horrocks, Peter F. Patel-Schneider, Sean Bechhofer, and Dmitry Tsarkov. OWL Rules: A Proposal and Prototype Implementation. *Journal of Web Semantics*, 3(1):23–40, 2005.

[93] Ernest R. House. Assumptions Underlying Evaluation Models. *Educational Researcher*, 7(3):4–12, 1978.

[94] Ernest R. House. *Evaluating with validity*. Sage Publications, 1981.

[95] Wei Hu, Gong Cheng, Dongdong Zheng, Xinyu Zhong, and Yuzhong Qu. The Results of Falcon-AO in the OAEI 2006 Campaign. In *Proceedings of the International Workshop on Ontology Matching (OM-2006) collocated with the 5th International Semantic Web Conference (ISWC-2006)*, volume 225 of *CEUR Workshop Proceedings*, pages 124–133, November 2006.

[96] Wei Hu, Yuanyuan Zhao, and Yuzhong Qu. Partition-based block matching of large class hierarchies. In *Proceedings of the 1st Asian Semantic Web Conference (ASWC'06)*, pages 72–83, 2006.

[97] Watts S. Humphrey and James W. Over. *Introduction to the Team Software Process*. Addison-Wesley Professional, 1999.

[98] Mirela Huza, Mounira Harzallah, and Francky Trichet. OntoMas: a tutoring system dedicated to ontology matching. In *Proceedings ISWC workshop on ontology matching*, pages 228–323, Athens (GA US), 2006.

[99] Science in the Information Society. Workshop Science in the Information Society, `http://www.icsu.org/Gestion/img/ICSU_DOC_DOWNLOAD/31_DD_FILE_recommendations.pdf`, accessed on 17.08.2008, 2003.

[100] ISO. ISO/IEC Standard 9126-1 Software Engineering - Product Quality - Part 1: Quality Model, June 2001.

[101] E. Jacquet-Lagréze and J. Siskos. Assessing a set of additive utility functions for multicriteria decision making. *European Journal of Operational Research*, 10:151–164, 1982.

[102] Yannis Kalfoglou and Marco Schorlemmer. Ontology mapping: the state of the art. *The Knowledge Engineering Review (KER)*, 18(1):1–31, 2003.

[103] Annika Kangas, Jyrki Kangas, and Jouni Pykäläinen. Outranking Methods As Tools in Strategic Natural Resources Planning. *Silva Fennica*, 35:215–227, 2001.

[104] Jyrki Kangas, Annika Kangas, Pekka Leskinen, and Jouni Pykäläinen. MCDM Methods in Strategic Planning of Forestry on State-Owned Lands in Finland: Applications and Experiences. *Journal of Milti-Criteria Decision Analysis*, 10:257–271, 2001.

[105] Bonnie Kaplan and Dennis Duchon. Combining Qualitative and Quantitative Methods in Information Systems Research: A Case Study. *MIS Quarterly*, 12:571–586, December 1988.

[106] Barbara Kitchenham and Shari Lawrence Pfleeger. Software Quality: The Elusive Target. *IEEE Computer Society*, 13(1):12–21, 1996.

[107] Marja-Riitta Koivunen. W3C Semantic Web Activity. `http://www.w3.org/Talks/2001/1102-semweb-fin`, accessed on 18.08.2008, 2001.

[108] Jyrki Kontio. A Case Study in Applying a Systematic Method for COTS Selection. In *ICSE' 96: Proceedings of 18th International Conference on Software Engineering*, pages 201–209. IEEE Computer Society, 1996.

[109] Konstantinos Kotis, Alexandros Valarakos, and George Vouros. AUTOMS: Automated Ontology Mapping through Synthesis of Methods. In *Proceedings of the International Workshop on Ontology Matching (OM-2006) collocated with the 5th International Semantic Web Conference (ISWC-2006)*, volume 225 of *CEUR Workshop Proceedings*, pages 96–105, November 2006.

[110] Konstantinos Kotis and George A. Vouros. The HCONE Approach to Ontology Merging. In Christoph Bussler, John Davies, Dieter Fensel, and Rudi Studer, editors, *Proceedings of the First European Semantic Web Symposium (ESWS 2004)*, pages 137–151, 2004.

[111] Konstantinos Kotis, George A. Vouros, and Konstantinos Stergiou. Towards Automatic Merging of Domain Ontologies: The HCONE-merge approach. *Journal of Web Semantics (JWS)*, 4(1):60–79, 2006.

[112] Charles W. Krueger. Software reuse. *ACM Computing Surveys (CSUR)*, 24(2):131–183, 1992.

[113] Thomas S. Kuhn. *The Structure of Scientific Revolutions*. University of Chicago Press, 3 edition, December 1996.

[114] HP Labs. Introduction to Semantic Web Technologies. `http://www.hpl.hp.com/semweb/sw-technology2.htm?jumpid=reg_R1002_USEN`, accessed on 13.05.2008.

[115] Soren Lauesen. *Software Requirements: Styles and Techniques*. Addison-Wesley Professional, 2002.

[116] Alain Léger, Robert Meersman, Mustafa Jarrar, Jean Luc Tagliamonte, and David Hamon. Industry Board and Memorandum of Understanding. Technical Report Deliverable 1.1.1, Knowledge Web EU Network of Excellence, 2005.

[117] Mark A. Lemley and David W. O'Brien. Encouraging Software Reuse. *Stanford Law Review*, 49(2):255–304, 1997.

[118] Li Li, Baolin Wu, and Yun Yang. Agent-based Ontology Integration for Ontology-based Applications. In *Proceedings of the Australasian Ontology Workshop (AOW 2005)*, volume 58, pages 53–59, 2005.

[119] I. Linkov, F.K. Satterstrom, G. Kiker, C. Batchelor, T. Bridges, and E. Ferguson. From comparative risk assessment to multi-criteria decision analysis and adaptive management: Recent developments and applications. *Environment International*, 32(8):1072–1093, 2006.

[120] I. Linkov, A. Vardhese, S. Jamil, T.P. Seager, G. Kiker, and T. Bridges. Multi-criteria Decision Analysis: A Framework for Structuring Remedial Decisions at Contaminated Sites. *Comparative Risk Assessment and Environmental Decision Making (Nato Science Series: IV: Earth and Environmental Sciences)*, pages 15–54, 2004.

[121] Adolfo Lozano-Tello and Asunción Gómez-Pérez. ONTOMETRIC: A Method to Choose the Appropriate Ontology. *Journal of Database Management*, 15:1–18, 2004.

[122] Earl R. MacCormac. Forecasting Loads and Designing Rates for Electric Utilities. *Analytic Hierarchy Process. Applications and Studies*, pages 138–154, 1989.

[123] Jayant Madhavan, Philip A. Bernstein, Pedro Domingos, and Alon Y. Halevy. Representing and Reasoning about Mappings between Domain Models. In *Proceedings of the 18th National Conference on Artificial Intelligence and 4th Conference on Innovative Applications of Artificial Intelligence (AAAI/IAAI)*, pages 80–86, 2002.

[124] Jayant Madhavan, Philip A. Bernstein, and Erhard Rahm. Generic Schema Matching with Cupid. In *Proceedings of the 27th International Conference on Very Large Databases (VLDB)*, pages 48–58, 2001.

[125] Ming Mao and Yefei Peng. PRIOR System: Results for OAEI 2006. In *Proceedings of the International Workshop on Ontology Matching (OM-2006) collocated with the 5th International Semantic Web Conference (ISWC-2006)*, volume 225 of *CEUR Workshop Proceedings*, pages 173–180, November 2006.

[126] Salvatore T. March and Gerald F. Smi. Design and natural science research on information technology. *Decision Support Systems*, 15(4):7251–266, 1995.

[127] Sabine Massmann, Daniel Engmann, and Erhard Rahm. COMA++: Results for the Ontology Alignment Contest OAEI 2006. In *Proceedings of the International Workshop on Ontology Matching (OM-2006) collocated with the 5th International Semantic Web Conference (ISWC-2006)*, volume 225 of *CEUR Workshop Proceedings*, pages 107–114, November 2006.

[128] James C. McDavid and Laura R. L. Hawthorn. *Program Evaluation & Performance Measurement: An Introduction to Practice*. Sage Publications, 2005.

[129] Deborah L. McGuinness, Richard Fikes, James Rice, and Steve Wilder. The Chimaera ontology environment. In *Proceedings of the 17th National Conference on Artificial Intelligence (AAAI-2000)*, pages 1123–1124, 2000.

[130] Sergey Melnik, Hector Garcia-Molina, and Erhard Rahm. Similarity Flooding: A Versatile Graph Matching Algorithm and Its Application to Schema Matching. In *Proceedings of the 18th International Conference on Data Engineering (ICDE02)*, pages 117–128, 2002.

[131] Mary A. Meyer and Jane M. Booker. *Eliciting and Analyzing Expert Judgment: A Practical Guide*. ASA-SIAM Series on Statistics and Applied Probability. Society for Industrial & Applied Mathematics, 2001.

[132] Malgorzata Mochol and Elena Paslaru Bontas. A Metadata-Based Generic Matching Framework for Web Ontologies. Technical Report TR-B-05-03, FU Berlin, June 2005.

[133] Malgorzata Mochol, Anja Jenztsch, and Jérôme Euzenat. Applying an Analytic Method for Matching Approach Selection. In *Proceedings of the International Workshop on Ontology Matching (OM-2006) collocated with the 5th International Semantic Web Conference (ISWC-2006)*, volume 225 of *CEUR Workshop Proceedings*, pages 37–48, November 2006.

[134] Malgorzata Mochol, Radoslaw Oldakowski, and Ralf Heese. Ontology based Recruitment Process. In *Proceedings of the GI Jahrestagung (GI2004)*, pages 198–202, 2004.

[135] Malgorzata Mochol and Elena Paslaru B. Simperl. Practical Guidelines for Building Semantic eRecruitment Applications. In *Proceedings of the International Conference on Knowledge Management (iKnow'06), Special Track: Advanced Semantic Technologies*, 2006.

[136] Malgorzata Mochol and Elena Paslaru Bontas Simperl. A high-level Architecture of Metadata-based Ontology Matching (MOMA) Framework. In *Proceedings of the 5th International Workshop on Web Semantics (WebS '06) at DEXA'06*, pages 354–358, September 2006.

[137] Andrea De Montis, Pasquale De Toro, Bert Droste-Franke, Ines Omann, and Sigrid Stagl. Criteria for quality assessment of MCDA-methods. In *Proceedings of the 3rd Biennial Conference of the European Society for Ecological Economics (ESEE)*, 2000.

[138] Michael D. Myers. Investigating Information Systems with Ethnographic Research. *Communication of the AIS*, 2:1–20, 1999.

[139] Miklos Nagy, Maria Vargas-Vera, and Enrico Motta. DSSim-ontology Mapping with Uncertainty. In *Proceedings of the International Workshop on Ontology Matching (OM-2006) collocated with the 5th International Semantic Web Conference (ISWC-2006)*, volume 225 of *CEUR Workshop Proceedings*, pages 115–123, November 2006.

[140] Robert Neches, Richard Fikes, Tim Finin, Thomas Gruber, Ramesh Patil, Ted Senator, and William R. Swartout. Enabling technology for knowledge sharing. *AI Magazine*, 3(12):35–56, 1991.

[141] Sławomir Niedbala. OWL-CtxMatch: Algorytm semantycznego dopasowywania ontologii OWL. In *Proceedings of the Conference Bazy Danych: Aplikacje i Systemy (BDAS'06)*, June 2006.

[142] Sławomir Niedbala. OWL-CtxMatch in the OAEI 2006 alignment contest. In *Proceedings of the International Workshop on Ontology Matching (OM-2006) collocated with the 5th International Semantic Web Conference (ISWC-2006)*, volume 225 of *CEUR Workshop Proceedings*, pages 165–172, November 2006.

[143] Magnus Niemann, Malgorzata Mochol, and Robert Tolksdorf. Improving Online Hotel Search - What Do We Need Semantics For? In *Proceedings of the Semantics 2006 (Application Paper)*, 2006.

[144] Wei Hu Ningsheng Jian, Gong Cheng, and Yuzhong Qu. Falcon-AO: Aligning Ontologies with Falcon. In *Proceedings K-CAP Workshop on Integrating Ontologies*, pages 87–93, 2005.

[145] Lyndon J. B. Nixon and Malgorzata Mochol. Prototypical Business Use Cases. Technical Report Deliverable 1.1.2, Knowledge Web EU Network of Excellence, 2005.

[146] Natalya Noy and Mark Musen. PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment. In *Proceedings 17th National Conference of Artificial Intelligence (AAAI)*, pages 450–455, Austin (TX US), 2000.

[147] Natalya F. Noy. Semantic integration: a survey of ontology-based approaches. *ACM SIGMOD RecRecord*, 33(4):65–70, 2004.

[148] Elena Paslaru-Bontas. *A Contextual Approach to Ontology Reuse: Methodology, Methods and Tools for the Semantic Web.* PhD thesis, Freie Universität Berlin, February 2007.

[149] James A. Pearsol. The Nature of Explanation in Qualitative Evaluation. *Evaluation & the Health Professions*, 8(2):129–147, 1985.

[150] Brian J. Peterson, William A. Andersen, and Joshua Engel. Knowledge Bus: Generating Application-focused Databases from Large Ontologies. In *Proceedings of the KRDB'98*, volume 10, pages 2.1–2.10, 1998.

[151] Iris Pinkster, Bob van de Burgt, Dennis Janssen, and Erik van Veenendaal. *Successful Test Management: An Integral Approach*. Springer, 2 edition, June 2006.

[152] Helena Sofia Pinto and Joáo P Martins. A methodology for ontology integration. In *Proceedings of the International Conference on Knowledge Capture (K-CAP'01)*, pages 131–138. ACM Press, 2001.

[153] Jonathan Poole and J. A. Campbell. A Novel Algorithm for Matching Conceptual and Related Graphs. In *Proceedings of the 3rd International Conference on Conceptual Structures: Applications, Implementation and Theory*, volume 954, pages 293–307. Springer Berlin / Heidelberg, June 1995.

[154] Susan Purdon, Carli Lessof, Kandy Woodfield, and Caroline Bryson. Research Methods for Policy Evaluation. Technical Report 2, National Centre for Social Research, Department for Work and Pensions, 2001.

[155] Dennis Quan and David R. Karger. How to Make a Semantic Web Browser. In *Proceedings of the 13th International World Wide Web Conference (WWW'04)*, pages 255–265. ACM, 2004.

[156] Vincent Quint and Iréne Vatton. Structured Templates for Authoring Semantically Rich Documents. In *Proceedings of the International Workshop on Semantically Aware Document Processing And Indexing (SADPI'07)*, pages 41–48, May 2007.

[157] Erhard Rahm and Philip A. Bernstein. A survey of approaches to automatic schema matching. *Journal of Very Large Data Bases*, 10(4):334–350, 2001.

[158] Erhard Rahm and Philip A. Bernstein. On Matching Schemas Automatically. Technical Report MSR-TR-2001-17, Microsoft Research Technical Report, February 2001.

[159] Robert N. Rapoport. Three Dilemmas in Action Research: With Special Reference to the Tavistock Experience. *Human Relations*, 23(6):499–513, 1970.

[160] Stephen L. Reed and Douglas B. Lenat. Mapping Ontologies into Cyc. In *In AAAI 2002 Conference Workshop on Ontologies For The Semantic Web*, July 2002.

[161] Bernard Roy. *Multicriteria Methodology for Decision Aiding (Nonconvex Optimization and Its Applications)*. Kluwer Academic Publisher, 1996.

[162] Thomas Russ, Andre Valente, Robert MacGregor, and William Swartout. Practical Experiences in Trading Off Ontology Usability and Reusability. In *Proceedings of the Knowledge Acquisition Workshop (KAW99)*, 1999.

[163] Thomas L. Saaty. *The Analytic Hierarchy Process, Planning, Piority Setting, Resource Allocation*. McGraw-Hill, 1980.

[164] Thomas L. Saaty. How to Make a Decision: The Analytic Hierarchy Process. *European Journal of Operational Research*, 48(1):9–26, 1990.

[165] Thomas L. Saaty. How to Make a Decision. *Models, Methods, Concepts & Aplications of the Analytic Hierarchy Porcess*, pages 1–25, 2000.

[166] John Scott and Gordon Marshall. *A Dictionary of Sociology*. Oxford University Press, August 2005.

[167] Pavel Shvaiko. A Classification of Schema-Based Matching Approaches. In *Proceedings of the Meaning Coordination and Negotiation Workshop at the ISWC'04*, pages 119–130, 2004.

[168] Pavel Shvaiko. Iterative schema-based semantic matching. Technical Report DIT-04-020, University of Trento, June 2004.

[169] Pavel Shvaiko and Jérôme Euzenat. A Survey of Schema-Based Matching Approaches. *Journal on Data Semantics (JoDS)*, 4:146–171, 2005.

[170] Elena Paslaru Bontas Simperl and Malgorzata Mochol. Cost Estimation for Ontology Development. In *Proceedings of the 9th Business Information System (BIS 2006)*, June 2006.

[171] Elena Paslaru Bontas Simperl, Christoph Tempich, and Malgorzata Mochol. Cost Estimation for Ontology Development: Applying the ONTOCOM Model. In Witold Abramowicz and Heinrich C. Mayr, editors, *Technologies for Business Information Systems*, pages 327–339. Springer, 2007.

[172] Barry Smith. Ontology. In Luciano Floridi, editor, *Blackwell Guide to the Philosophy of Computing and Information*, pages 155–166. Oxford: Blackwell, 2003.

[173] Barry Smith. Ontology and Information Systems. In *Proceedings of the ECOOP2004 Workshop: Philosophy, Ontology, and Information Systems*, 2004.

[174] Rudi Studer, V. Richard Benjamins, and Dieter Fensel. Knowledge Engineering: Principles and Methods. *Data Knowledge Engineering*, 25(1-2):161–197, 1998.

[175] Daniel L. Stufflebeam and William J. Webster. An Analysis of Alternative Approaches to Evaluation. *Educational Evaluation and Policy Analysis*, 2(3):5–20, 1980.

[176] Gerd Stumme and Alexander Maedche. FCA-MERGE: Bottom-up merging of ontologies. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 225–230, 2001.

[177] Mari Carmen Suárez-Figueroa, Guadalupe Aguado de Cea, Carlos Buil, Caterina Caracciolo, Martin Dzbor, Asunción Gómez-Pérez, German Herrero, Holger Lewen, Elena Montiel, and Valentina Presutti. D5.3.1 NeOn Development Process and Ontology Life Cycle. NeOn Project Deliverable D5.3.1, Universidad Politecnica de Madrid, AUG 2007.

[178] Mari Carmen Suárez-Figueroa and Asunción Gómez-Pérez. Building Ontology Networks: How to Obtain a Particular Ontology Network Life Cycle? In *Proceedings of the International Conference on Semantic Systems (I-SEMANTICS2008)*, pages 142 – 149, 2008.

[179] York Sure, Steffen Staab, and Rudi Studer. On-To-Knowledge Methodology (OTKM). In Steffen Staab and Rudi Studer, editors, *Handbook on Ontologies*, pages 117–132. Springer Verlag, 2004.

[180] M. Carmen Suárez-Figueroa, Raúl García-Castro, Asunción Gómez-Pérez, Raúl Palma, Lyndon J.B. Nixon, Elena Paslaru, Jens Hartmann, and Mustafa Jarrar. Identification of standards on metadata for ontologies. Deliverable D1.3.2 KWEB/2004/D1.3.2/v1.0, EU-IST Network of Excellence (NoE) IST-2004-507482 KWEB, January 2005.

[181] Bill Swartout, Ramesh Patil, Kevin Knight, and Tom Russ. Toward Distributed Use of Large-Scale Ontologies. In *Proceedings of the 10th Knowledge Acquisition for Knowledge-Based Systems Workshop (KAW 10)*, pages 1–19, `http://ksi.cpsc.ucalgary.ca/KAW/KAW96/swartout/Banff_96_final_2.html`, accessed on 15.08.2008, November 1996.

[182] Said Tabet and Mike Dea. SWRL - Issues List and Language Extensions. http://www.w3.org/Submission/2004/SUBM-SWRL-20040521/issuesList.html, accessed on 04.09.2007, 2004.

[183] Hideaki Takeda, Paul Veerkamp, Tetsuo Tomiyama, and Hiroyuki Yoshikawa. Modeling Design Processes. *AI Magazine*, 11:37–48, Winter 1990.

[184] Jie Tang, Juanzi Li, Bangyong Liang, Xiaotong Huang, Yi Li, and Kehong Wang. Using Bayesian decision for ontology mapping. *Web Semantics: Science, Services and Agents on the World Wide Web*, 4(4):243–262, December 2006.

[185] Evangelos Triantaphyllou. Reduction of Pairwise Comparison in Decision Making via Duality Approach. *Journal of Multi-Criteria Decision Analysis*, 8:299–310, 1999.

[186] KeWei Tu and Yong Yu. CMC: Combining Multiple Schema-Matching Strategies based on Credibility Prediction. In *Proceedings of 10th International Conference on Database Systems for Advanced Applications (DASFAA 2005)*, volume 3453 of *Lecture notes in computer science*, pages 888–893, 2005.

[187] Rachanee Ungrangsi and Elena Simperl. OMEGA: an automatic ontology metadata generation algorithm. In *Proceedings of the 16th International Conference on Knowledge Engineering and Knowledge Management Knowledge Patterns (EKAW2008)*, pages 239 – 254, 2008.

[188] Michael Uschold. Where are the Semantics in the Semantic Web? http://www.starlab.vub.ac.be/WhereAreSemantics-AI-Mag-FinalSubmittedVersion2.pdf, accessed on 28.08.2008, 2002.

[189] Michael Uschold and Michael Gruninger. Ontologies and Semantics for Seamless Connectivity. *SIGMOD Rec.*, 33(4):58–64, 2004.

[190] Michael Uschold, Michael Healy, Keith Williamson, Peter Clark, and Steve Woods. Ontology Reuse and Application. In *Proceedings of the International Conference on Formal Ontology and Information Systems (FOIS1998)*, pages 179–192, 1998.

[191] Mike Uschold and Michael Grüninger. ONTOLOGIES: Principles, Methods and Applications. *Knowledge Engineering Review*, 11(2):93–155, 1996.

[192] Mike Uschold and Robert Jasper. A Framework for Understanding and Classifying Ontology Applications. In *Proceedings of the IJCAI99 Workshop on Ontologies and Problem-Solving Methods(KRR5)*, 1999.

[193] Mike Uschold and Martin King. Towards a Methodology for Building Ontologies. In *Proceedings of the IJCAI'95, Workshop on Basic Ontological Issues in Knowledge Sharing*, 1995.

[194] Vijay Vaishnavi and Bill Kuechler. Design Research in Information Systems. `http://www.isworld.org/Researchdesign/drisISworld.htm`, accessed on 02.10.2007, 2007.

[195] Alexandros G. Valarakos, Georgios Paliouras, Vangelis Karkaletsis, and George Vouros. A name-Matching Algorithm for supporting Ontology Enrichment. In George A. Vouros and Themistoklis Panayiotopoulos, editors, *Methods and Applications of Artificial Intelligence*, volume 3025/2004 of *Lecture Notes in Computer Science*, pages 381–389. Springer Verlag, April 2004.

[196] Cornelis Joost van Rijsbergen. *Information Retrieval*. Butterworths London, `http://www.dcs.gla.ac.uk/Keith/Preface.html`, accessed on 17.02.2008, 2nd edition, 1979.

[197] Luis G. Vargas. An Overview of the Analytic Hierarchy Process and its applications. *European Journal of Operational Research*, 48:2–8, 1990.

[198] Ubbo Visser. *Intelligent Information Integration for the Semntic Web*. Springer, 2004.

[199] John G. Vlahakis and Willian R. Partridge. Assessment of Security at Facilities that produce Nuclear Weapons. *Analytic Hierarchy Process. Applications and Studies*, pages 182–191, 1989.

[200] Yoland Wadsworth. *Everyday Evaluation on the Run*. Allen & Unwin Academic, 2 edition, 1997.

[201] Yair Wand and Ron Weber. Information Systems and Conceptual Modelling: A Research Agenda. *Information Systems Research*, 13(4):363–376, 2002.

[202] M. Williamson. Special Section: Software Reuse. *CIO Magazine*, 3 1997.

[203] Duo Zhang Yi Li, Juanzi Li and Jie Tang. Result of Ontology Alignment with RiMOM at OAEI'06. In *Proceedings of the International Workshop on Ontology Matching (OM-2006) collocated with the 5th International Semantic Web Conference (ISWC-2006)*, volume 225 of *CEUR Workshop Proceedings*, pages 181–190, November 2006.

[204] Jacek Zak. The Comparison of Multiobjective Ranking Methods Applied to solve the Mass Transit Systems' Decision Problems. In *Proceedings of the 16th*

Mini - EURO Conference Artificial Intelligence in Transportation and 10th Meeting of EWGT, September 2005.

[205] Songmao Zhang and Olivier Bodenreider. NLM Anatomical Ontology Alignment System Results of the 2006 Ontology Alignment Contest. In *Proceedings of the International Workshop on Ontology Matching (OM-2006) collocated with the 5th International Semantic Web Conference (ISWC-2006)*, volume 225 of *CEUR Workshop Proceedings*, pages 153–164, November 2006.

[206] Huimin Zhao. Semantic matching. Accros Heterogenous Data Sources. *Communication of the ACM*, 50(1):45–50, 2007.

Appendix A

Zusammenfassung

Interoperabilität gewinnt immer mehr an Bedeutung und spielt eine wichtige Rolle in der Semantic Web Community. Obwohl semantische Technologien eine eindeutige Identifikation von Konzepten, sowie die formale Beschreibung der Verbindungen zwischen den Konzepten und dadurch eine bedeutungsvolle und maschinenverständliche Darstellung von Daten erlauben, werden Entwickler heutzutage leider immer noch mit dem Problem semantischer Interoperabilität konfrontiert, das als wichtiger Baustein zum Erreichen des vollen Potentials des Web anzusehen ist. Um semantische Interoperabilität erreichen zu können, müssen verschiedene Systeme in der Lage sein, die Daten so auszutauschen, dass deren genaue Bedeutung leicht zugänglich ist und sie in das Format übersetzt werden können, welches das entsprechende System versteht. Demzufolge spiegeln die Schema oder Ontologie Matching-(Vergleichs-) und Mapping-(Abbildungs-)Verfahren das zentrale Problem bzgl. der Dateninteroperabilität und -integration im Semantic Web wider.

Angesichts dieser Situation, und weil die Entwicklung und Existenz von bereits getesteten und bewährten Ontologie-Matching-Algorithmen (Matcher) und Tools entscheidend für die zukünftige Weiterentwicklung und Ausbreitung von semantischen Technologien sein wird, wollen wir zu den Lösungen in diesem Bereich beisteuern. Wir haben ein *Metadata-based Ontology Matching Framework (MOMA Framework)* entwickelt, das zur Datenintegration und -interoperabilität beiträgt, indem es eine Verbindung zwischen Matchern und Ontologien schafft, um schließlich die passenden Verfahren für die betroffenen Ontologien in Abhängigkeit zu spezifischen Anwendungsanforderungen, in denen diese verwendet werden sollten, vorzuschlagen. Da es schwierig ist die existierenden Matchingverfahren auf einer theoretischen Basis miteinander zu vergleichen, wurde eine mehrstufige Charakteristik für Matchingverfahren (*MCMA – Multilevel Characteristic for Matching Approaches*) entwickelt. Sie beschreibt die verschiedenen Ansätze auf unterschiedlichen Detaillierungsniveaus und wird zur Matcherauswahl verwendet. Unter Berücksichtigung der Anforderungen für die erfolgreiche Anwendung der semantischen Technologien soll das MOMA Framework in der Lage sein, die Bedürfnisse und Ansprüche verschiedener Nutzergruppen – Menschen und Maschinen – zu bedienen. Für menschliche Benutzer kann der Auswahlprozess der passenden Ontologie-Matching-

Algorithmen manuell durchgeführt werden, während Maschinen zumindest eine semi-automatische Matcherauswahl erfordern. Da die Entscheidung über die Angemessenheit der Algorithmen von mehreren Kriterien abhängig ist und Vergleichsskalen nicht konsistent sind, wurde in dem manuellen Auswahlprozess ein systematisches Verfahren (analytischer Hierarchien-Prozess, AHP), das die Erwartung, die Intuition und die heuristische Entscheidungsfindung strukturiert, eingesetzt. Um die semi-automatische Auswahl der passenden Algorithmen zu ermöglichen benutzt das MOMA Framework zusätzliche Informationen – Metadaten – über Ontologien und vorhandene Matcher. Die Ontologie-Metadaten beinhalten Informationen über Ontologie-Eigenschaften, die eine entscheidende Rolle bei der Matcherauswahl spielen und die Matcher-Metadaten, die auf MCMA basieren, beschreiben die wichtigsten Eigenschaften der Matchingverfahren. Darüber hinaus, da explizites Wissen über die Abhängigkeiten zwischen den Matching Algorithmen und den Strukturen, auf denen die Matcher angewandt werden können, benötigt wird, haben wir das Wissen in Form von Abhängigkeitsregeln unter Betrachtung der Matchercharakteristik und den ontologischen Strukturen, die verglichen werden sollten, notiert und legten damit fest, welche Matcher für welche ontologische Quellen eingesetzt werden können.

Da die Evaluationsaspekte von MOMA Framework direkt mit dem Einsatz des Frameworks in realen Situationen zusammenhängen, wurde sowohl die Evaluation vom AHP-basierten als auch vom regel-basiertem Ansatz im Kontext von Anwendungsfällen aus dem Wettbewerb der Ontology Alignement Evaluation Initiative (OAEI) durchgeführt. Die Ergebnisse des Evaluationsprozesses bestätigen die Anwendbarkeit des MOMA Frameworks und zeigen die Richtigkeit seiner Auswahlprognosen.

Appendix B

Erklärung / Declaration

Hiermit versichere ich, Małgorzata Mochól, an Eides statt, dass ich die vorliegende Dissertationsschrift mit dem Titel "The Methodology for Finding Suitable Ontology Matching Approaches" selbständig verfasst habe. Als Hilfsmittel bei der Durchführung der Arbeit und Verfassung der Schrift dienten mir nur die darin angegebenen Literaturquellen.

I hereby affirm that I, Małgorzata Mochól, alone composed this thesis with the title"The Methodology for Finding Suitable Ontology Matching Approaches". My research was carried out and this thesis was composed solely with the aid of the literature referenced herein.

Małgorzata Mochól
Berlin
November 18, 2008

I also declare that some material in this thesis has been already published in the following papers:

- M. Mochol, A. Jentzsch: Towards a rule-based matcher selection. In Proceedings of the 6th International Conference on Knowledge Engineering and Knowledge Management Knowledge Patterns (EKAW2008), Springer, LNAI 5268, p. 109–120, Acitrezza, Catania, Italy, September 29 - October 03, 2008.
- M. Mochol: Interoperability Issues, Ontology Matching and MOMA. In Proceedings of the International Conference on Semantic Systems (I-SEMANTICS), Graz, Austria, September 03-05, 2008.

- M. Mochol: Detecting Suitable Matching Approaches for Semantic Hotel Search. In Proceedings of the 8th International Conference Information Management ICIM, Information Management/ Human-Computer Interaction (IM/HCI 2007), Poland, June, 2007.
- M. Mochol, A. Jentzsch, J. Euzenat: Towards a methodology for selection suitable matching approaches - A Case Study. Poster at the 4th European Semantic Web Conference (ESWC2007), Austria, June 2007
- M. Mochol, A. Jentzsch, J. Euzenat: Applying an Analytic Method for Matching Approach Selection. In Proceedings of the International Workshop on Ontology Matching (OM-2006) collocated with the 5th International Semantic Web Conference (ISWC-2006), volume 225 of CEUR Workshop Proceedings, p. 37–48, Athens, Georgia, USA, November, 2006.
- M. Mochol, E. Paslaru Bontas Simperl: A high-level Architecture of Metadata-based Ontology Matching (MOMA) Framework. In Proceedings of the 5th International Workshop on Web Semantics (WebS '06) at DEXA'06, p. 354–358, Poland, September 2006.
- M. Mochol: MOMA Framework - How to find a suitable matching approach. Poster at the KnowledgeWeb PhD Symposium 2006 (KWEPSY2006) co-located with the Knowledge Web general assembly meeting, Montenegro, June 2006.
- M. Mochol: Metadata-Based Matching Framework for Ontologies. In Proceedings of the 13th Doctoral Consortium at CAiSE'06, C. Rolland and P. F. Campos (Eds.), volume 263 of CEUR Workshop Proceedings, p. 1136–1143, Luxemburg, June 2006.

Appendix C

List of Publications

2008

- E. Simperl, M. Mochol: A Case Study in Building Semantic eRecruitment Applications. In Semantic Web for Business: Cases and Applications, R. Garcia (Eds.), IGI Global, ISBN 978-1-60566-066-0, 2008.

- A. Polleres, M. Mochol: Expertise bewerben und finden im Social Semantic Web. In Social Semantic Web: Die Konvergenz von Social Software, Web 2.0 und Semantic Web, Reihe: X.media.press, Blumauer, Andreas; Pellegrini, Tassilo (Eds.), p. 175–207, 2008.

- M. Niemann, M. Mochol, R. Tolksdorf: Enhancing Hotel Search with Semantic Web Technologies. Journal of Theoretical and Applied Electronic Commerce Research, ISSN 0718-1876 Electronic Version, Vol. 2, Issue 2, 2008.

- M. Mochol: Interoperability Issues, Ontology Matching and MOMA. In Proceedings of the International Conference on Semantic Systems (I-SEMANTICS), Graz, Austria, September 03-05, 2008.

- M. Mochol, A. Jentzsch: Towards a rule-based matcher selection. In Proceedings of the 6th International Conference on Knowledge Engineering and Knowledge Management Knowledge Patterns (EKAW2008), Springer, LNAI 5268, p. 109–120, Acitrezza, Catania, Italy, September 29 - October 03, 2008.

2007

- M. Mochol: Detecting Suitable Matching Approaches for Semantic Hotel Search. In Proceedings of the 8th International Conference Information Management ICIM, "Information Management /Human-Computer Interaction" (IM/HCI 2007), Jurata, Polnad, 12 - 14 June, 2007.

- M. Mochol, A. Jentzsch, H. Wache: Suitable employees wanted? Find them with semantic techniques. Workshop Making Semantics Work For Business, European Semantic Technology Conference 2007 (ESTC2007), Vienna, Austria, May 31 - June 1, 2007.
- B. Aleman-Meza, U. Bojars, H. Boley, J. G. Breslin, M. Mochol, L. J.B. Nixon, A. Polleres, A. V. Zhdanova: Combining RDF Vocabularies for Expert Finding. In Proceedings of the 4th European Semantic Web Conference (ESWC2007), Innsbruck, Austria, 3-7 June 2007.
- M. Mochol, A. Jentzsch, J. Euzenat: Towards a methodology for selection suitable matching approaches - A Case Study. Poster at the 4th European Semantic Web Conference (ESWC2007), Innsbruck, Austria, 3-7 June 2007.
- M. Mochol, Holger Wache, L. Nixon: Improving the accuracy of job search with semantic techniques. In Proceedings of the 10th International Conference on Business Information Systems (BIS2007), Poznan, Poland 25-27 April 2007.
- E. Paslaru Bontas Simperl, C. Tempich, and M. Mochol: Cost estimation for ontology development: applying the ONTOCOM model. In Technologies for Business Information Systems, W. Abramowicz, H. C. Mayr (Eds.), 432 p., Hardcover ISBN: 978-1-4020-5633-8, 2007.
- M. Mochol, Anne Cregan, Denny Vrandecic, Sean Bechhofer: Exploring OWL and Rules - A Simple Teaching Case. International Journal of Teaching and Case Studies (IJTCS), Special Issue: Teaching Semantic Web: Integration with CS/IS curriculum and Teaching Case Studies, 2007.
- R. Heese, M. Mochol, R. Oldakowski: Semantic Web Technologies in the Recruitment Domain. In Competencies in Organizational E-Learning: Concepts and Tools, M.-A. Sicilia (Eds.), Anfang 2007.

2006

- M. Niemann, M. Mochol, R. Tolksdorf: Improving Online Hotel Search - What Do We Need Semantics For?. In Proceedings of the Semantics 2006 (Application Paper), Vienna, Austria, November 2006.
- M. Mochol, A. Jentzsch, J. Euzenat: Applying an Analytic Method for Matching Approach Selection. In Proceedings of the International Workshop on Ontology Matching (OM-2006) collocated with the 5th International Semantic

Web Conference (ISWC-2006), volume 225 of CEUR Workshop Proceedings, pages 37–48, Athens, Georgia, USA, November, 2006.

- J. Euzenat, M. Mochol, P. Shvaiko, H. Stuckenschmidt, O. Šváb, V. Svátek, W. R. van Hage, and M. Yatskevich: Results of the Ontology Alignment Evaluation Initiative 2006. In Proceedings of the International Workshop on Ontology Matching (OM-2006) collocated with the 5th International Semantic Web Conference (ISWC-2006), volume 225 of CEUR Workshop Proceedings, pages 73–95, Athens, Georgia, USA, November, 2006.

- M. Mochol, H. Wache, L. Nixon: Improving the recruitment process through ontology-based querying. In Proceedings of the 1st International Workshop on Applications and Business Aspects of the Semantic Web (SEBIZ 2006), collocated with the 5th International Semantic Web Conference (ISWC-2006), Athens, Georgia, USA, November, 2006.

- M. Mochol, E. Paslaru Bontas Simperl: Practical Guidelines for Building Semantic eRecruitment Applications. In Proceedings of the International Conference on Knowledge Management (iKnow'06), Special Track: Advanced Semantic Technologies (AST'06), Graz, Austria, September 2006.

- M. Mochol, E. Paslaru Bontas Simperl: A high-level Architecture of Metadata-based Ontology Matching (MOMA) Framework. In Proceedings of the 5th International Workshop on Web Semantics (WebS '06) at DEXA'06, p. 354–358, Poland, September 2006.

- M. Mochol: MOMA Framework - How to find a suitable matching approach. Poster at the KnowledgeWeb PhD Symposium 2006 (KWEPSY2006), Montenegro, June 2006.

- J. Garbers, M. Niemann, M. Mochol: A Personalized Hotel Selection Engine. Poster at the 3rd European Semantic Web Conference 2006 (ESWC2006), Montenegro, June 2006.

- M. Mochol: Metadata-Based Matching Framework for Ontologies. In Proceedings of the 13th Doctoral Consortium at CAiSE'06, C. Rolland and P. F. Campos (Eds.), volume 263 of CEUR Workshop Proceedings, p. 1136–1143, Luxemburg, June 2006.

- E. Paslaru Bontas Simperl, M. Mochol: Cost Estimation for Ontology Development. In Proceedings of the Business Information System (BIS 2006), Klagenfurt, Austria, June 2006.

- T. Falk, R. Heese, C. Kaspar, M. Mochol, D. Pfeiffer, M. Thygs, R. Tolksdorf: Semantic-Web-Technologien in der Arbeitsplatzvermittlung. InformatiksSpektrum, Heft 2, 2006, ISSN: 0170-6012 (Paper), 1432–122X (Online), DOI: 10.1007/s00287-006-0061-4.

- R. Tolksdorf, M. Mochol, R. Heese, R. Eckstein, R. Oldakowski, C. Bizer: Semantic-Web-Technologien im Arbeitsvermittlungsprozess. Wirtschatfsinformatik: Internetökonomie, Vol.48, No.1, p. 17–26, January 2006.

2005

- A. Cregan, M. Mochol, D. Vrandecic, S. Bechhofer: Pushing the limits of OWL, Rules and Protégé. A simple example. In Proceedings of the Workshop - OWL: Experiences and Directions (OWLED-2005), Galway, Ireland, November 2005.

- E. Paslaru, M. Mochol: Towards a Cost Estimation Model for Ontology Engineering. In Proceedings of the Berliner XML Tage 2005, Berlin, Germany, September 2005.

- E. Paslaru Bontas, M. Mochol: Towards a reuse-oriented methodology for ontology engineering. In Proceedings of the 7th International Conference on Terminology and Knowledge Engineering (TKE 2005), Copenhagen, Denmark, August 2005.

- E. Paslaru Bontas, M. Mochol, R. Tolksdorf: Case Studies on Ontology Reuse. In Proceedings of the 5th International Conference on Knowledge Management (I-Know'05), Graz, Austria, June 2005.

- M. Mochol: Application of Semantic Web Technologies in the Human Resource Domain. In Proceedings of the 7th International Conference on Information Management (Interaction Human-Computer; HCI2005),Gdansk, Poland, June 2005, published in "Information Management", Eds.: B.F.Kubiak, A.Korowicki, pages 394-401

- C. Bizer, R. Heese, M. Mochol, R. Oldakowski, R. Tolksdorf, R. Eckstein: The Impact of Semantic Web Technologies on Job Recruitment Processes. In Proceedings of the 7. Internationale Tagung Wirtschaftsinformatik 2005, Bamberg, Germany, February 2005.

2004 and earlier:

- M. Mochol, R. Oldakowski, R. Heese: Ontology Based Recruitment Process. In Proceedings of the Workshop: Semantische Technologien für Informationsportale, at INFORMATIK 2004, pages 198–202, Ulm, Germany, September 2004.
- M. Mochol: Discourse Support Design Patterns. In Proceedings of the Workshop on Online Dispute Resolution (ODR) at the International Conference on Artificial Intelligence and Law (ICAIL 2003), Eds. Arno R. Lodder, et al. Edinburgh,pages 61-74, June 2003.
- M. Mochol: Discourse Support Design Patterns. In Proceedings of the 6th International Conference HUMAN-COMPUTER INTERACTION (HCI2003) in business process re-engineering and creating integrated information systems, Sopot, June 2003.