



HAL
open science

Context-based Semantic Similarity across ontologies

Patrick Hoffmann

► **To cite this version:**

Patrick Hoffmann. Context-based Semantic Similarity across ontologies. Computer Science [cs]. Université Claude Bernard - Lyon I, 2008. English. NNT: . tel-00363300v1

HAL Id: tel-00363300

<https://theses.hal.science/tel-00363300v1>

Submitted on 21 Feb 2009 (v1), last revised 27 Apr 2009 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

présentée

devant l'UNIVERSITÉ CLAUDE BERNARD – LYON 1

pour l'obtention

du **DIPLÔME DE DOCTORAT**
(Spécialité Informatique, CNU 27)

(arrêté du 7 août 2006 et arrêté du 6 janvier 2005)

présentée et soutenue publiquement le **16 décembre 2008**

par

M. Patrick HOFFMANN

TITRE :

Context-based Semantic Similarity across Ontologies

(Similarité sémantique inter-ontologies basée sur le contexte)

Directeurs de thèse : Parisa GHODOUS / Ram D. SRIRAM

JURY : M. Kokou YÉTONGNON, rapporteur
M. Aris M. OUKSEL, rapporteur
Mme Parisa GHODOUS, examinateur
M. Djamal BENSLIMANE, examinateur
M. Shaw C. FENG, examinateur

Acknowledgments

I would like to express my gratitude to my supervisor, the Pr. Parisa Ghodous, to have persisted in believing that I could achieve this PhD, supported and encouraged me; she has battled to give me the privilege to defend today. I acknowledge my debt to the Pr. Ram D. Sriram to have welcomed me at the National Institute of Standards and Technology (NIST) in the context of a co-supervision agreement. At NIST, I have notably benefited from the counsel and advice from Eswaran Subramahnan, Boonserm Kulvatunyou, Nenad Ivezic, and Evan Wallace. Without this rich experience the thesis would not have been the same.

I would like to thank particularly the Pr. Kokou Yétongnon from the University of Bourgogne and the Pr. Aris M. Ouksel from the University of Illinois at Chicago, who have accepted to review my dissertation. I am grateful for the positive report that they have made, and for their presence as jury members. Thanks go out also to the other jury members: Shaw C. Feng, from NIST, with whom I have enjoyed working for some time at NIST, and who was always encouraging and positive; and Djamel Benslimane, from the University Claude Bernard of Lyon 1.

Among those who have helped me during the writing of my thesis, I would like to address a very special “thank you” to Gaurav Ameta who taught me some helpful edition hints, and believed in this work; Catarina Ferreira Da Silva who provided some excellent comments and welcome encouragements; Edouard Morris who challenged me with new ideas and corrected my usage of English; my sister-in-law Ingrid Hoffmann, who reviewed a few chapters; Géraldine Husser who helped me with the bibliography; without them, the thesis would not have been exactly the same. Others have participated in one way or another, including Lionel Médini and Fabian Neuhaus. Others still have helped in practical matters, for example my sister Evelyne Hoffmann and my friend Steve Schmerbeck who have assisted me in the printing of the dissertation.

I have been blessed with numerous people motivating, encouraging and comforting me, particularly when the light at the end of the tunnel seemed very dim. Claudette Rosado-Reyes has given regular encouragement, as did Aaron Johnson, Benoîte Lefort, Vorapong Chaichanamongkol, Lucy Salah, Daniel Davies, my sister, my mother, etc.; others have also prayed ... they should rejoice, their prayers have been heard! Others have been a family, welcoming me as I travelled in France and in the USA; among them are Edouard and Christelle Thollot, and Steve and Gayla Schmerbeck. Still others have been my friends, sometimes despite the circumstances, and have been there when I needed it, and this was enough. I am deeply indebted to my parents for the support they provided through my entire life and in particular during these last months; thanks for imparting me the resolution to go further in the studies to their conclusion.

Finally, this research could not have been accomplished without the grant from the French government, the support of NIST during my stay in the USA, and the generous contribution of the Rhône-Alpes region for my supplementary expenses in the context of the co-supervision agreement.

Table of Contents

Acknowledgments	3
Table of Contents	5
List of Figures	9
List of tables	10
GENERAL INTRODUCTION	13
Approach and Contributions.....	13
Organization of the thesis.....	14
PART I – SEMANTIC INTEROPERABILITY	17
Chapter 1: Interoperability – Limits of traditional approaches	19
1.1 Interoperability by standards.....	19
1.2 Interoperability by integration of incompatible software	21
1.3 Conclusion.....	22
Chapter 2: Semantic Interoperability	24
2.1 Ontology definition	24
2.2 Formal ontology languages	26
2.3 Semantic interoperability in practice.....	26
2.4 Conclusion.....	27
Chapter 3: Ontology reconciliation	28
3.1 Heterogeneities that hinder accurate ontology reconciliation.....	28
3.2 Ontology reconciliation approaches	29
3.3 Conclusion.....	30
Chapter 4: Ontology Matching Methods and Tools	32
4.1 Methods for ontology matching	32
4.2 Ontology matching process and tools.....	35
4.3 Mapping formats.....	37
4.4 Conclusion.....	38
Chapter 5: Limits of Ontology Model and Development	40
5.1 Weaknesses of the ontology model.....	40
5.2 Lack of precise specifications for ontology modelling	44
5.3 Limitations of present development tools, methods, and practices	45
5.4 Complexity of the reality to model	46
5.5 Conclusion.....	47

Chapter 6: Problem Statement: Reconciliation of Independent Ontologies through Mappings is pragmatically Inconsistent	49
6.1 Limitations of ontology mappings for a real ontology reconciliation	49
6.2 Calls for context-based ontology reconciliation	50
6.3 Thesis statement	51
6.4 Research questions	51
6.5 Significance of the study	52
6.6 Research scope.....	52
PART II – STATE OF THE ART ON CONTEXT AND SIMILARITY	53
Chapter 7: Frameworks for improved ontology development and reconciliation.....	55
7.1 Frameworks for regulating ontology development	55
7.2 Frameworks for regulating ontology reconciliation	58
7.3 Conclusion	60
Chapter 8: Notion of Context in Computer Science	62
8.1 That which relates an entity with its surrounding environment.....	63
8.2 A formal representation of conditional statements.....	65
8.3 Conclusion	68
Chapter 9: Measure of Similarity and Relatedness between Concepts.....	70
9.1 Characteristics of similarity	70
9.2 Similarity measures intra-ontology	71
9.3 Semantic relatedness intra-ontology	73
9.4 Similarity measures across ontologies	74
9.5 Conclusion	74
Chapter 10: Context-based Similarity Measures.....	76
10.1 Context-based similarity measures between concepts.....	77
10.2 Context-based similarity measures between instances	78
10.3 Conclusion	79
PART III – CONTEXT-BASED EVALUATION OF CONCEPT PERTINENCE	81
Chapter 11: Methodology for an Effective Contextualisation	83
11.1 Stage 1: guidelines to contextualize the application.....	84
11.2 Stage 2: Connection between target context and reference	88
11.3 Stage 3: context-based architecture that fulfils the set objectives.....	91
11.4 Conclusion	92
Chapter 12: Guidelines to Contextualize Ontology Reconciliation	93
12.1 Objectives.....	93
12.2 Usages of context	94
12.3 Triples (target, reference, aim of the comparison)	95
12.4 Conclusion	97
Chapter 13: Connection between the Contexts of the Root Concept and of the Concept enquired.....	98
13.1 Relevant features to describe the context of concepts	98
13.2 Conditions to retrieve data	99
13.3 Model of perspectives.....	101
13.4 Model of the connection between perspectives.....	104
13.5 Evaluation of the data for the connection	105

13.6 Interpretation of the data to achieve the disambiguation	106
13.7 Conclusion	107
Chapter 14: Connection between the Contexts of the Applicant and of the Root Concept	108
14.1 Relevant features of the applicant's context	108
14.2 Conditions to retrieve domains and tasks	108
14.3 Model of domains and tasks	109
14.4 Model of the connection between applicant's context and concept context	110
14.5 Evaluation of the data for the connection	110
14.6 Interpretation of the data to achieve the personalization	111
14.7 Conclusion	111
Chapter 15: Connection between the Contexts of the Interoperability Need and of the Concept enquired.....	112
15.1 Relevant features	112
15.2 Conditions to retrieve data about the interoperability task	113
15.3 Model of the Interoperability task	114
15.4 Model of the connection between interoperability task and perspectives	115
15.5 Evaluation of the data for the connection	115
15.6 Interpretation of the data to evaluate the concept enquired	115
15.7 Conclusion	116
Chapter 16: A Context-based Architecture that Takes into Account Practical Implications for Better Ontology Reconciliation	117
16.1 Application architecture.....	117
16.2 Evaluation.....	120
16.3 Completion by a generic measure to better compare concepts.....	121
16.4 Conclusion	125
PART IV – IMPLEMENTATION AND APPLICATION.....	127
Chapter 17: Principles of Implementation of the System.....	129
17.1 Working of the context-based system	129
17.2 UML Use cases	129
17.3 Dependency diagram	130
17.4 Architecture of the system.....	132
Chapter 18: Interaction of an Agent with a Business Application that enquires our System	134
18.1 Interaction with a business application	134
18.2 A need for interoperability.....	135
18.3 The request	135
18.4 The business application returns contextualized results	136
18.5 Conclusion	137
Chapter 19: An Expert Models the Company Information into the System.....	138
19.1 The expert gathers the ontologies developed	138
19.2 The expert models perspectives	140
19.3 The expert models domains and tasks.....	143
19.4 The expert connects domains and tasks with perspectives.....	144
19.5 The expert models interoperability tasks	145
19.6 Conclusion.....	146
Chapter 20: An Expert Prepares the System for a new Collaboration	147
20.1 The expert manages ontology mappings	147

20.2 The expert relates the perspectives of both organizations.....	150
20.3 The expert relates interoperability tasks and perspectives	152
20.4 Conclusion	153
Chapter 21: Algorithms and Database schema	154
21.1 Disambiguation by the comparison of concept perspectives	154
21.2 Personalization by the comparison of domains/tasks and perspectives	157
21.3 Evaluation by the comparison of interoperability tasks and perspectives.....	159
21.4 Evaluation of the generic measure of semantic similarity	160
21.5 Conclusion	162
GENERAL CONCLUSION	163
Thesis summary and contributions	163
Discussion and limits	164
Future research	165
Perspectives	165
BIBLIOGRAPHY	167
APPENDICES	177
Appendix A Ontologies in practice: a basic example	179

List of Figures

FIGURE 1: USE OF A STANDARD AS NEUTRAL REPRESENTATION	20
FIGURE 2: EXAMPLE OF SCHEMA MATCHING [RAHM AND BERNSTEIN, 2001].....	22
FIGURE 3: SPECTRUM OF ARTEFACTS CALLED "ONTOLOGY"	25
FIGURE 4: USE OF AN ONTOLOGY TO ACHIEVE INTEROPERABILITY BETWEEN TWO APPLICATIONS.....	27
FIGURE 5: CLASSIFICATION OF ONTOLOGY MATCHING METHODS.....	32
FIGURE 6: CLASSIFICATION OF MATCHING APPROACHES AND METHODS [EUZENAT ET AL, 2004]	33
FIGURE 7: MODELLING WITH THE VIEW "STATE" OR "PROCESS"	41
FIGURE 8: VARIOUS CLASSIFICATIONS OF CONCEPTS [KLEIN, 2001]	42
FIGURE 9: INTRODUCTION OF A DISTINCTION WITH NEW CLASSES OR NEW ATTRIBUTES	43
FIGURE 10: MOST ABSTRACT CONCEPTS OF VARIOUS UPPER-LEVEL ONTOLOGIES	47
FIGURE 11: PARTIAL VIEW OF THE METADATA TAXONOMY [PASLARU-BONTAS, 2007]	60
FIGURE 12: EFFECT OF THE CONTEXT: THE TWO ELLIPSES SEEM TO HAVE A DIFFERENT COLOUR	76
FIGURE 13: PATHS RELATE INSTANCES THROUGH DISTINCT REGIONS [ALEMAN-MEZA ET AL, 2005]	79
FIGURE 14: SCHEMA TO ILLUSTRATE THE METHODOLOGY.....	83
FIGURE 15: FIRST STAGE OF THE METHODOLOGY.....	84
FIGURE 16: SECOND STAGE OF THE METHODOLOGY.....	88
FIGURE 17: THIRD AND LAST STAGE OF THE METHODOLOGY	91
FIGURE 18: APPLICATION OF THE METHODOLOGY (FIRST STAGE).....	93
FIGURE 19: THE DIFFERENT USAGES OF CONTEXT SELECTED.....	97
FIGURE 20: APPLICATION OF THE METHODOLOGY (SECOND STAGE, FIRST USAGE OF CONTEXT).....	98
FIGURE 21: CLASSIFICATION OF FRENCH ROADS.....	100
FIGURE 22: EXAMPLE OF ONTOLOGY OF PERSPECTIVE.....	103
FIGURE 23: CONNECTION BETWEEN ONTOLOGIES AND THEIR PERSPECTIVES.....	105
FIGURE 24: ILLUSTRATION OF THE DISAMBIGUATION PROCESS.....	107
FIGURE 25: APPLICATION OF THE METHODOLOGY (SECOND STAGE, SECOND USAGE OF CONTEXT) ...	108
FIGURE 26: APPLICATION OF THE METHODOLOGY (SECOND STAGE, THIRD USAGE OF CONTEXT).....	112
FIGURE 27: APPLICATION OF THE METHODOLOGY (THIRD STAGE)	117
FIGURE 28: ILLUSTRATION OF THE ARCHITECTURE WITH AN EXAMPLE	118
FIGURE 29: COMPARISON OF THE NORMS (-LOG(X) AND OUR NORM).....	123
FIGURE 30: ILLUSTRATION OF THE CONTEXT-BASED APPROACH.....	129
FIGURE 31: USE CASES FOR AGENT AND EXPERT, AND ADMINISTRATOR.....	130
FIGURE 32: DIAGRAM OF THE DEPENDENCIES	131
FIGURE 33: SIMPLIFIED EXAMPLE OF API ACCORDING TO THE FAÇADE PATTERN	133
FIGURE 34: WHEEL COMPOSITION	135
FIGURE 35: ONTOLOGIES DEVELOPED BY THE COMPANY IMANO	139
FIGURE 36: ONTOLOGIES ANNOTATED WITH PERSPECTIVES.....	142
FIGURE 37: MAPPINGS BETWEEN TWO ONTOLOGIES OF IMANO AND RICHELIN	148
FIGURE 38: RICHELIN'S PRODUCT ONTOLOGY ANNOTATED WITH PERSPECTIVES	150
FIGURE 39: CONNECTION BETWEEN THE PERSPECTIVES OF THE TWO ORGANIZATIONS	152
FIGURE 40: CLASS DIAGRAM TO REPRESENT CONTEXTUAL INFORMATION	154
FIGURE 41: DATABASE SCHEMA ABOUT ANNOTATION OF CONCEPTS WITH PERSPECTIVES	155
FIGURE 42: ASSOCIATION OF A PATH FOLLOWING [BIDAULT, 2002].....	156
FIGURE 43: ONTOLOGY WITH PRE-ORDERING NUMBERS.....	157
FIGURE 44: DATABASE SCHEMA ABOUT RELATIONS BETWEEN PERSPECTIVES.....	157
FIGURE 45: DATABASE SCHEMA ABOUT THE AGENT CONTEXT	158
FIGURE 46: DATABASE SCHEMA ABOUT INTEROPERABILITY TASKS AND PERSPECTIVES.....	159
FIGURE 47: CLASS DIAGRAM FOR THE COMPUTATION OF GENERIC SIMILARITY	160
FIGURE 48: DATABASE SCHEMA ABOUT SIMILARITY BETWEEN CONCEPTS.....	161

List of tables

TABLE 1: COMPARISON OF THE DIFFERENT APPROACHES FOR ONTOLOGY RECONCILIATION.....	31
TABLE 2: MAPPING FORMAT PROPOSED BY OAEI (LEVEL 0) [EUZENAT, 2006]	37
TABLE 3: KINDS OF PERSPECTIVES AND THEIR RESPECTIVE IMPACT ON ONTOLOGY DESIGN	40
TABLE 4: THE WORLD WE LIVE IN AND OBSTACLES TO ITS MODELLING WITH FORMAL LOGIC	47
TABLE 5: SEMANTIC RELATIONS SUGGESTED FOR STANDARDIZATION [KINGSTON, 2008]	55
TABLE 6: THE ZACHMAN FRAMEWORK, REPRESENTATION [ZACHMAN, 1987]	56
TABLE 7: SEMANTIC RELATIONS VALUABLE FOR MOST NOUN PHRASES [MOLDOVAN ET AL, 2004]	57
TABLE 8: THEMATIC RELATIONS SUGGESTED TO STRUCTURE ONTOLOGIES [SOWA, 1996]	58
TABLE 9: SYNTHESIS OF THE DIFFERENT PROPOSALS	61
TABLE 10: SYNTHESIS OF THE USE OF CONTEXT IN COMPUTER SCIENCE	69
TABLE 11: COMMON FUNCTIONS FOR THE MEASURES OF SIMILARITY	72
TABLE 12: FORMULAS OF THE SIMILARITY MEASURES CITED	73
TABLE 13: FUNCTIONS USED BY THE MEASURES OF SIMILARITY	77
TABLE 14: MAIN USAGES OF CONTEXTUAL INFORMATION.....	85
TABLE 15: USAGES OF CONTEXT AND THE CORRESPONDING TARGETS, REFERENCES, AIMS OF THE COMPARISON, AND TYPES OF CONTEXT.....	87
TABLE 16: LIST OF DEFINITIONS OF TERMS FOR OUR APPROACH	94
TABLE 17: EXAMPLE OF DIFFERENCE OF PRAGMATIC MEANING FOR A CONCEPT	95
TABLE 18: SIGNS TO RECOGNIZE VARIOUS KINDS OF PERSPECTIVES IN AN ONTOLOGY	100
TABLE 19: COMMON FUNCTIONS FOR THE MEASURES OF SIMILARITY	124
TABLE 20: ASSOCIATION OF RELEVANCY TO TRIPLES (DOMAIN, TASK, PERSPECTIVE).....	145

Abbreviations and acronyms

CAD	COMPUTER-AIDED DESIGN
CAE	COMPUTER-AIDED ENGINEERING
CAM	COMPUTER-AIDED MANUFACTURING
CE	CONCURRENT ENGINEERING
DBF	DESIGN BY FEATURE
DL	DESCRIPTION LOGICS
FBS	FUNCTION-BEHAVIOUR-STRUCTURE FRAMEWORK
FTP	FILE TRANSFER PROTOCOL
IRI	INTERNATIONAL RESOURCE IDENTIFIER
IDE	INTEGRATED DEVELOPMENT ENVIRONMENT
ISO	INTERNATIONAL ORGANIZATION FOR STANDARDIZATION
NIST	NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
NLP	NATURAL LANGUAGE PROCESSING
OAEI	ONTOLOGY ALIGNMENT EVALUATION INITIATIVE
OWL	WEB ONTOLOGY LANGUAGE
PLM	PRODUCT LIFECYCLE MANAGEMENT
RDF	RESOURCE DESCRIPTION FRAMEWORK
SAT	PROPOSITIONAL SATISFIABILITY
STEP	STANDARD FOR THE EXCHANGE OF PRODUCT MODEL DATA
HTTP	HYPERTEXT TRANSFER PROTOCOL
URI	UNIVERSAL RESOURCE IDENTIFIER
XHTML	EXTENDED HYPERTEXT MARKUP LANGUAGE
XML	EXTENSIBLE MARKUP LANGUAGE
WWW	WORLD WIDE WEB

Typographical notations

Concepts: *Thing* ,

Semantic relations: *is-a*, *part-of*, etc.

Attribute: price

General Introduction

With globalization and the increase of competition, companies need to collaborate more than ever with other organizations, in order to achieve better products for a reduced cost. In such cases, companies need to exchange data, and to use the data exchanged within their software applications. This is a pre-requisite so that the business applications from the different companies can inter-operate.

Our hypothesis for this thesis is that companies should model by ontology the concepts that describe the meaning of the data that their software applications manipulate, and that they provide an access to this ontology for the partner organizations.

Various tools exist to reconcile the ontologies of two organizations. Mostly based on terminological and structural methods, these tools can detect couples of concepts that may be equivalent, called mappings.

When a need for interoperability occurs, it is expressed under the form of a request. The success of the exchange is evaluated according to the relevancy of the information or service obtained for the need that triggered the request.

As companies have different views on the data manipulated, it is not sufficient to ensure that ontology concepts are mapped correctly. Indeed, the concept might be the right concept, without being a pertinent answer for the need expressed. It may for example be associated with data totally irrelevant for the usage intended by the first company.

Approach and Contributions

This research work has been done in the Lyon Research Center for Images and Intelligent Information Systems (LIRIS), in collaboration with the National Institute of Standards and Technology (NIST), so as to benefit from the insight of research fellows on interoperability of processes and product design in the business world.

The approach undertaken is to examine whether a context-based approach can improve the reconciliation of ontologies by mappings. Indeed, ontology matching methods and tools do not take into consideration any contextual information on how the concepts will actually be used. The assumption is that by considering the context, one should provide a much better evaluation of pertinence of a concept for a given interoperability request.

We have therefore reviewed the literature about context in Computer Science, to highlight the principles of context definition, how it is modeled and used. We have developed a methodology to determine what is contextual, how to collect, model and employ it. This is our first contribution.

We have applied the methodology to the reconciliation of ontologies for unanticipated collaborations between companies. We have found three different contexts:

- The context of the concepts is well characterized with the “perspectives” that were recognized to have guided the ontology development, or that show some variations concerning the data associated with the ontology. This is our second contribution.
- The context of the agent that originates the request is modeled by a selection of the domains and tasks of the agent’s company.
- The context of the interoperability need is made of the “interoperability task” for which the result of the request is intended to be used.

These contexts will be employed with the different usages:

- By comparing the concept perspectives, one *disambiguates* their pragmatic meaning
- By comparing agent’s domains and tasks with the concept perspectives one will *personalize* the previous disambiguation, through the recognition of the understanding that the agent has of the concept.
- By comparing the interoperability task with the concept perspectives, one *evaluates* the practical value of the concept.

Ontology mappings may not be in sufficient number to relate all the concepts defined in the ontologies of the agent’s company with the ontologies from other organizations. Indeed, the agent may send a request about any concept. We propose therefore a generic semantic similarity measure across ontologies, based on the edge-counting measure from [Leacock and Chodorow, 1998]. This is our third contribution.

Finally, we propose a context-based evaluation of concept pertinence that relies on the three models of context described earlier. A request is composed of four elements: the agent that originates it; the root concept defined in one of the agent’s company ontologies, that gives an approximation of the desired result; the interoperability task; the concept enquired, which is the concept to be evaluated, defined by an ontology from the partner organizations. This is our fourth and last contribution.

Organization of the thesis

This study is organized into four parts: part I describes the notion of semantic interoperability that holds out the prospect of flexible connection between heterogeneous software and data, but shows its limitations when more than one ontology is used. Part II presents a review of the literature about frameworks that could improve the validity of ontology reconciliation, and on context and similarity in Computer Science. Part III gives our methodology to determine, model and use contextual information, and apply it for the reconciliation of ontologies. Part IV exhibits the architecture of a system based on the approach, and an application example.

Part I – Semantic interoperability

Chapter 1 introduces the need for flexible inter-operation between heterogeneous software and data. Chapter 2 defines the notion of semantic interoperability and how it is to be implemented in practice. Chapter 3 presents the need for reconciliation of ontologies, the heterogeneities that hinder this reconciliation and the different architectures proposed. Chapter 4 reviews the methods and tools to match ontologies,

which is a pre-requisite for most ontology reconciliation approaches. Chapter 5 describes the limits of ontology model and development that result in a lack of reliability of mappings. Chapter 6 states the problem that will be studied in this thesis, that is, the reconciliation of ontologies is pragmatically inconsistent; it introduces the choice of a context-based approach to improve the reconciliation of ontologies.

Part II – State of the art on context and similarity

Chapter 7 examines frameworks proposed specifically for the regulation of ontology development or ontology comparison that could improve the validity of ontology reconciliation. Chapter 8 presents a panorama of context-based approaches in Computer Science, in order to observe some general principles in all these approaches. Chapter 9 reviews the measures proposed between concepts defined in the same ontology or in different ontologies. Chapter 10 examines the context-based measures between concepts or concept instances.

Part III – Context-based evaluation of concept pertinence

Chapter 11 introduces a methodology to establish context-based solution. Chapter 12 shows how the methodology is applied to the reconciliation of ontologies. Chapter 13 describes the application of the second stage of the methodology, to show how the context of concepts could be compared for disambiguation. Chapter 14 describes in the same way how the agent's context could be compared with the context of concepts for personalization. Chapter 15 describes how the context interoperability need could be compared to the context of concepts for evaluation. Chapter 16 contains the application of the last stage of the methodology, including the presentation of a generic similarity measure between ontology concepts across ontologies, and of a context-based semantic similarity measure across ontologies made of the assembling of the usages studied previously, that is, disambiguation, personalisation and evaluation.

Part IV – Implementation and application

Chapter 17 presents the architecture of a system that implements the approach, and outlines how the system may be used. Chapter 18 gives a scenario of utilisation, where an agent encounters an interoperability need while using a business application. Chapter 19 shows the preliminary work that must have been done by an expert, to prepare the contextual information specific to the company. Chapter 20 does the same with contextual information recorded into the system, for each new collaboration. Chapter 21 details the algorithms and the structure of the database.

PART I – Semantic Interoperability

In this part, we introduce the need for semantic interoperability, and the weakness of the current approaches for semantic interoperability. We will first introduce the need for interoperability, and the limits of the traditional approaches. After defining the notion of semantic interoperability and the ontological approach, we will display the heterogeneities that hinder semantic interoperability when resources refer to concepts defined in different ontologies, and the different architectures proposed to reconcile ontologies. We will then survey the methods and tools to match ontologies, as ontology matching is a pre-requisite for most ontology reconciliation approaches. Finally, after describing the limits of ontology model and development that result in a lack of reliability of mappings, we will state the problem that studied in this thesis, that is, the reconciliation of ontologies is pragmatically inconsistent, and propose to consider the context to improve the reconciliation of ontologies.

Chapter 1: Interoperability – Limits of traditional approaches

In this chapter, we will introduce the need for interoperability and its importance. We will present the traditional approaches of integration and standardisation of formats, and demonstrate that they are insufficient.

Organizations rely increasingly on software, especially business applications, to help them provide efficient services and products. In order to get the maximum efficiency gain expected from information technology, organizations have to manage their software configuration in such a way that the data computed or retrieved by some applications can be used as input for others. Interoperability is commonly understood as “*a property referring to the ability of diverse systems and organizations to work together (inter-operate)*”¹. By extension, it includes the ability to “*exchange information and to use the information that has been exchanged*”².

Various studies estimate the costs of lack of interoperability between software applications in millions of dollars in industries [Brunnermeier and Martin, 1999], [Gallaher et al., 2002]. A well-known example of such costs is found in [Mel and Bartholomew, 2007] and [Clark, 2007]. They describe how the production of the Airbus 380 experienced delays and changes that the company estimated as \$6 billions of lost profits, due to errors generated during the exchange of computer-aided designs between two software applications not fully compatible.

1.1 Interoperability by standards

When different software applications use a similar type of data, a way to facilitate interoperability is to rely on established models of data, or standards. Applications can implement standards and use them as their internal format; they can also import and export in this neutral format, usually with loss of information. Figure 1 shows how the use of a neutral representation leads to a reduction of the number of translators required for the inter-operation of n software applications. Yes indeed, standards reduce the need to build converters to export and import into the internal format to a few ones, instead of having to build converters for all possible proprietary formats, which may also lack reliable documentation.

“*Interoperability is made possible by the implementation of standards*”, states the IEEE glossary of standards. The most evident example of this is probably the World Wide Web, which connects different platforms on the same Internet network, based on

¹ From Wikipedia, see <http://en.wikipedia.org/wiki/Interoperability>

² Definition of interoperability from IEEE (Institute of Electrical and Electronics Engineers, Inc.), see <http://www.ieee.org>

standard protocols such as HTTP³ and FTP⁴, and a language to describe Web pages, (X)HTML^{5 6}.

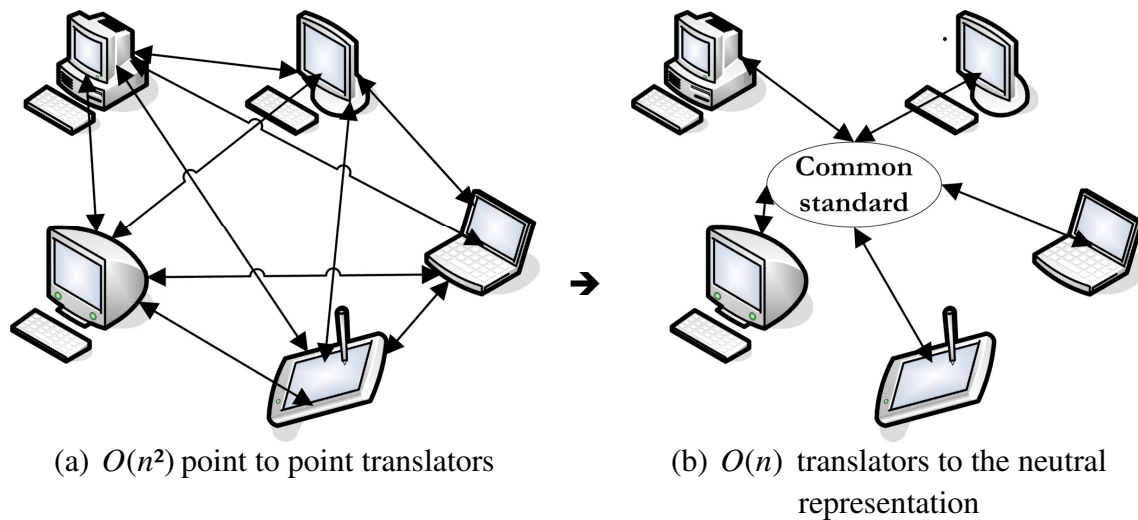


Figure 1: Use of a standard as neutral representation⁷

Until recently, standards were usually limited to the specification of a particular type of data. These standards were precisely defined, but sufficiently generalised to be used by a variety of applications. A format that is good for a kind of data is rarely adequate for another kind. For example, PNG is an image format that is text-friendly, but it does not support text editing as the ASCII TEXT format does. Other examples of standards include UML, PDF, CORBA/OMG, IDEF5, ebXML, etc.

There are sometimes different standards for a similar kind of data. For example, exchange of product geometrical information may be achieved using the standards IGES (“Initial Graphic Exchange Standard”, from the USA National Bureau of Standards), SET (“Standard d’Echange et de Transfert” from the France AFNOR), VDA-FS, STEP (STandard for the Exchange of Product Model Data, which is ISO 10303), etc.

The standard specification is not only concerned with the type of data, but also on its representation: when applications perform different kinds of operations on the data, they may need a representation that is adapted to these operations. For example, all types of images can look pretty much the same on the screen, but bitmap-based and vector-based images rely on completely different representations (e.g. JPEG and PNG for bitmaps, and SVG for vectorial drawing).

³ HyperText Transfer Protocol (HTTP), see <http://www.w3.org/Protocols/>

⁴ File Transfer Protocol (FTP), see http://www.w3.org/Protocols/rfc959/2_Overview.html

⁵ HyperText Markup Language (HTML) is a recommendation from the W3C, see <http://en.wikipedia.org/wiki/HTML> and <http://www.w3.org/TR/REC-html40/>

⁶ The extension of HTML, eXtensible HyperText Markup Language (XHTML), is also a recommendation from the W3C, and conforms to XML syntax, see <http://en.wikipedia.org/wiki/XHTML> and <http://www.w3.org/TR/xhtml1/>

⁷ From [Patil, 2005], with permission

In 1998, the W3C introduced XML⁸, which is “*a general-purpose specification for creating custom markup languages*”. XML is extensible, and can therefore represent almost any type of data. It allows the representation of any structures based on lists and trees. But XML is not only a standard to structure data in a mostly hierarchical way. It is also a standard to annotate data with metadata. Data is annotated with tags whose label states what the data represents.

For an application to implement a XML-based format, it can use one of the many generic converters available for XML serialization for various development languages. Indeed, the XML specification rules the way the data is serialized. What remains, to treat an XML document, is to decide how to structure the data in the internal development language, and to do appropriate operations according to the different types of data.

With XML-based formats, the serialization of data is made easier, but there is still much work left to define specific representations of data. XML does not furnish any means of comparing the data descriptors. Some standards have been developed to represent the meaning of these data descriptors, so as to be able to compare them: RDF (Resource Description Framework), OIL, DAML, OWL... we will describe the role of these standards in more detail in the next chapter.

Steven Ray, from the National Institute of Standards and Technology (NIST), points out that “*the number of communication standards is growing geometrically*” and that the standards have to represent more and more complex information structures. The current way of preparing standards is reaching its limits. Indeed, it takes a long time to have concerned organizations reach an agreement on which technology to adopt as a standard, and then to develop and test it. The increasing demand for standards is “*outpacing the ability to keep up*” [Ray, 2002].

Most organizations have legacy software, that they have developed internally, that relies on old computer technologies and formats, or is no longer supported by its vendor. This software may serve its purpose, and, over the years, has probably been cleared of most of its bugs. Rewriting such software would be not only costly but also error-prone, as new bugs can be introduced, so it is generally avoided. Therefore interfaces must be written between these different software applications, to enable them to cooperate.

1.2 Interoperability by integration of incompatible software

Interoperability is usually achieved by integrating all applications and resources together, that is by binding them tightly by configuration files and converters. Every time that the organization implements a new software technology, converters have to be developed manually to relate all applications that have been written previously. Because of this necessary update at every new technology generation, and because the bindings to be done are mainly specific to the organization, integration is a time-consuming and costly way of achieving interoperability. This is true all the more as the range of applications grows wider.

⁸ Extensible Markup Language (XML) is recommended by the World Wide Web Consortium (W3C), see <http://en.wikipedia.org/wiki/XML> and <http://www.w3.org/XML/>

A particular case of integration is when software applications rely on data stored in databases. To reuse the data stored in other databases, a usual approach consists in searching correspondences between the database schemas, or schema matching. For example, companies that participate in Amazon.com’s marketplace identify mappings between entries of their catalogues with entries of Amazon.com’s catalogue. As a result, buyers at Amazon may consult and buy their items almost as if they were Amazon.com’s items.

An example of schema matching is given in Figure 2. The matching has been effective, as all the elements of the schema S2 have been matched. We see that the hierarchy is not necessarily the same in the schema compared (here, “article” and “payee” have the same level of hierarchy in the schema S2, while they correspond to two different levels in the schema S). Also, there may be some elements in a schema that have no corresponding element, such as “Contact”, “Name”, “Address”.

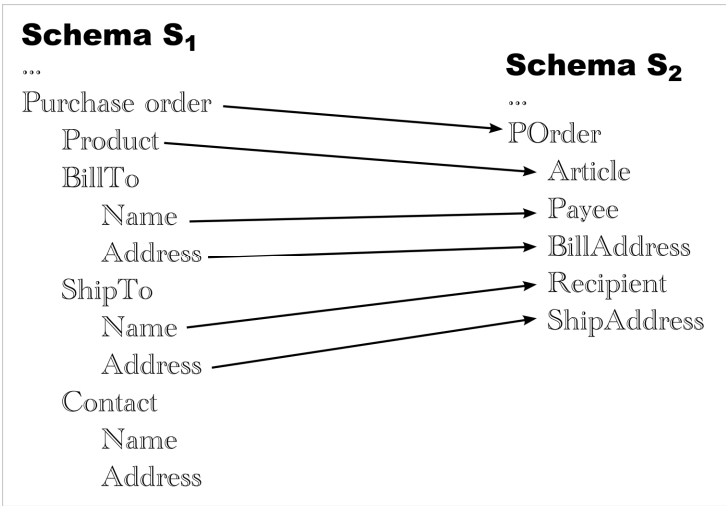


Figure 2: Example of schema matching [Rahm and Bernstein, 2001]

Integration by schema matching is hindered by the heterogeneity of the representations, revealing different levels of conflicts [Naiman and Ouksel, 1995]. Schema matching (or catalogue matching) remains for the main part a manual and error-prone task.

[Ouksel and Sheth, 1999] describe the shift that has occurred since the 1980s. At that time, the concern was to match objects “that were represented differently but were related conceptually”. With the information overflow that characterizes the last twenty years, a query may return thousands of results, and the concern is first of all to recognize which objects are related conceptually.

1.3 Conclusion

Integration is costly, time-consuming and error-prone, and the process of standardisation takes too much time compared to the increase of the needs, as the pace of technology fastens. Because of the limitations of interoperability by traditional standardisation and by integration, the scientific community is much concerned with providing means for systems to exchange information in a more flexible way.

XML has given a way to structure data in a uniform way, but there is still to manually declare the relation between every data element of the applications that need to communicate. In line with the general evolution of computer science, the perceived solution is to increase the level of abstraction: instead of describing XML data types with sole human-understandable labels, one would use concepts that associate labels with a determined meaning, or semantic.

Chapter 2: Semantic Interoperability

In this chapter we introduce the notion of semantic interoperability, and what needs to be done to make it work. We illustrate how this approach can be implemented in practice. We present the critical case when applications rely on distinct artefacts to define the meaning of the data that they manipulate.

The idea behind the noun phrase “semantic interoperability” is that one could specify the meaning, or “*semantics*”⁹, of the data, and achieve interoperability at the semantic level. By using tags with an established meaning in XML-based formats, one could more freely exchange data among applications: even applications that have never inter-operated with one another could communicate, if they can exploit some part of the data represented. Interoperability achieved this way is named semantic interoperability, “*the ability of information systems to exchange information on the basis of shared, pre-established and negotiated meanings of terms and expressions*” [Veltman, 2001].

An illustration of what semantic interoperability could involve is Tim Berners Lee’s¹⁰ vision of a “semantic Web”, sometimes called “Web 3.0”. To achieve this, content data should be annotated so as to describe their type – be it price, image, people’s profile – and meaning – what does this image represent, etc. The Web would become a “*Web of data*”, where the type and meaning of the data content of one website could be recognized, and the data used as input for another website or even desktop application [Berners Lee, 2000], [Shadbolt et al., 2006]. Tim Berners Lee describes a situation that should not exist anymore with the advent of the semantic Web: “*you are looking at a Web page, you find ... an event that you want to go to. The event has a place and has a time and it has some people associated with it. But you have to ... open your calendar to put the information on it. ... If you want the corporate details about people, you have to cut and paste the information... because your address book file and your original data files are not integrated together.*” [Moon, 2007].

For semantic interoperability to be possible, one needs to establish the meaning of tags used to annotate the data. This is done in what is called “*ontology*”.

2.1 Ontology definition

The artefact “ontology” is named after the philosophical study of being¹¹. The ambition is to have it describe the world, in such a way that computers could “understand” it by reasoning with its terms and rules. A generally accepted definition describes ontology as “*a specification of a conceptualization*” [Gruber, 1993]. It is a very broad definition, as (1) *conceptualization* could stand for “model of reality” as well as for “model of some virtual device” and (2) “*specification*” implies that the model must be precisely defined,

⁹ Semantics is “the meaning of a string in some language, as opposed to syntax which describes how symbols may be combined independent of their meaning.”, specialty definition for the domain of Computing, according to Webster's Online Dictionary, see <http://www.websters-online-dictionary.org/definition/semantics>

¹⁰ Tim Berners Lee, the author of the most known protocols of the WWW, namely HTTP and FTP

¹¹ The reader interested in the subject will find a resource guide on <http://www.formalontology.it/>

but does not state whether it is defined so as to be clear and precise for human beings or for computers. Also, neither “specification” nor “conceptualization” determines any degree of complexity, any format or language.

In practice, the term “ontology” represents “a spectrum of useful artefacts, from formal upper-level ontologies expressed in first order logic (...) to the simple lists of user-defined keywords”¹² [Bodenreider et al, 2007]. Figure 3 displays this spectrum, from simple data classifications (on the left) to ontologies defined using general logic (on the right). Ontologies are built for various purposes, including the modelling of reality or of information, the prescription of controlled terms, and the specification of constraints to ease the maintenance of a software application [Musen, 2007]. This range of purposes is surely a factor of the difficulty to agree on a common definition.

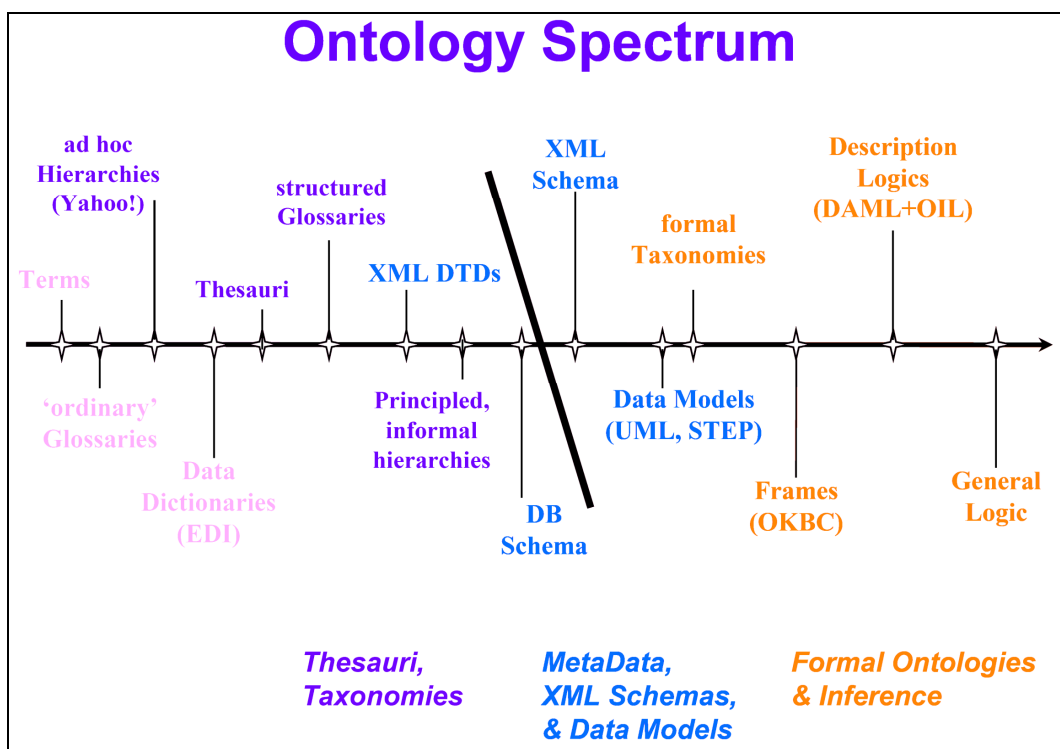


Figure 3: Spectrum of artefacts called "ontology"¹³

Another widely accepted definition defines an ontology as “an engineering artefact constituted by a specific vocabulary used to describe a certain reality, plus a set of explicit assumptions regarding the intended meaning of the vocabulary” [Guarino, 1998]. These descriptions should state, in a precise and computer-processable way, the “relationships between symbols and what they represent”. Appendix A introduces the basics of knowledge representation in ontologies.

¹² An upper-level ontology is an ontology that aims at defining abstract concepts that could thus be referred to by other ontologies; first order logic is a type of logic very expressive.

¹³ From the course “Interopérabilité et Intégration de Systèmes d’Information”, 2007-2008, by Parisa Ghodous and Nicolas Lumineau, with permission. Originally from AAAI 1999- Ontologies Panel by Gruninger, Lehmann, McGuinness, Uschold, Welty; – updated by McGuinness. Description in: www.ksl.stanford.edu/people/dlm/papers/ontologies-come-of-age-abstract.html

Guarino's definition does not stress that the model has to be machine-processable, and that all meanings should be expressed with precision and clarity. Also, it may be hard to discriminate what is a "*description of a certain reality*". For the purpose of this thesis we adopt the definition "*an engineering artefact describing a machine-processable model, that is constituted by a formal vocabulary plus a set of explicit assumptions defining precisely and with clarity the intended meaning of the vocabulary; the model describe classes (representing concepts), instances (representing individuals of concepts), attributes, and constrained relations*".

2.2 Formal ontology languages

Various representation languages have been proposed and used to describe ontologies, based on frames (such as F-Logic), graphs (such as conceptual graphs and RDF graphs) or any subclass of logic (such as KIF¹⁴ and CycL). As the meaning must be established globally, and be accessible, the advantage is given to languages developed especially for the World Wide Web, and based on the XML specification. Such languages include the possibility of declaring universal identifiers for any kind of resources (URI)¹⁵.

Because of this preference for languages devoted for the Web, the priority is given to languages recommended by the W3C: RDF, and since 2004, its extension as a "Web Ontology Language", OWL¹⁶ with its three variants OWL-Lite, OWL-DL and OWL-Full. Most ontologies nowadays are developed with OWL-DL. OWL-DL benefits from well defined semantics, known reasoning algorithms, and highly optimised implemented systems. It is a compromise between expressivity and computability. Its successor, OWL-DL 2.0, which is expected to be released in late 2009, will have more expressivity (e.g. remove qualified cardinality restrictions) with similar computational properties, and should be terser (but will not be fully compatible with RDF anymore).

2.3 Semantic interoperability in practice

When applications that need to inter-operate describe their data with semantic annotations, it becomes possible to find mappings between the data elements on which the applications rely. Then these mappings can serve to bind these applications so that they can work together. This is illustrated in Figure 4.

An illustration of this principle is given by [Hyvönen et al, 2005], who proceeded in four steps to resolve heterogeneities of data from many Finnish museums. To be able to present all items on a common portal, they created an RDF ontology, converted the content of their databases into XML format, and then related the vocabulary terms into the equivalent ontology concepts. This permitted to semi-automatically convert the information of the different artefacts into instances of the right ontology concept.

¹⁴ Knowledge Interchange Format (KIF), see <http://www.ksl.stanford.edu/knowledge-sharing/kif/>

¹⁵ "a compact string of characters used to identify or name a resource on the Internet", definition from Wikipedia, see <http://en.wikipedia.org/wiki/URI>

¹⁶ Web Ontology Language (OWL), See <http://www.w3.org/2004/OWL/>

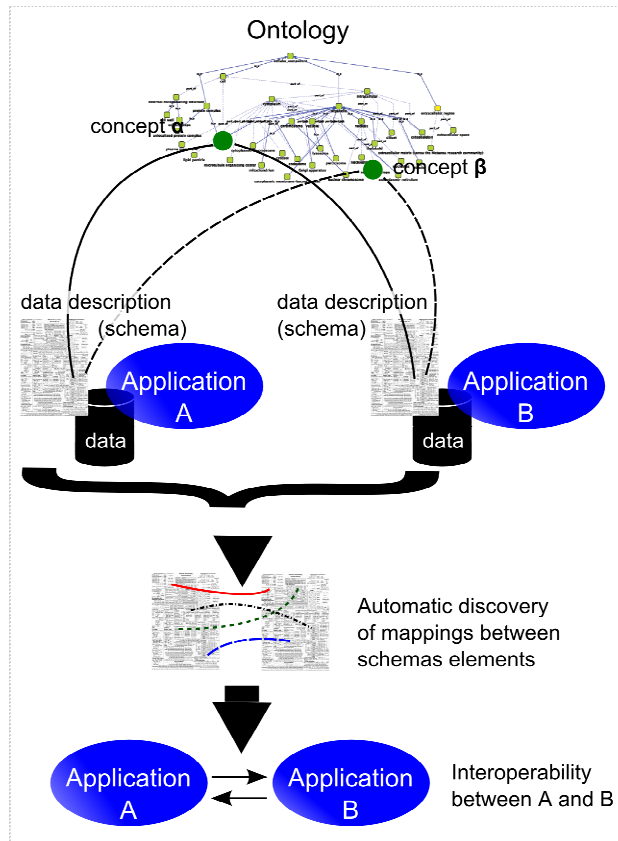


Figure 4: Use of an ontology to achieve interoperability between two applications

2.4 Conclusion

Semantic interoperability is expected to provide flexibility in the way applications interoperate. To achieve semantic interoperability, applications will rely not on a strict data format of exchange, but on a specification that permits to determine a custom structure for the data, and to annotate data with metadata (like XML). The meaning of this metadata can then be related to some established meaning, identified globally in artefacts named “ontologies”.

Semantic interoperability is much hindered when applications refer to identifiers of data meaning established in distinct ontologies. This is the case when different organizations need to collaborate for projects limited temporally: the collaboration is not known ahead of time, and the organizations cannot agree to use one shared ontology¹⁷ that should establish the meaning of all their data. There is therefore a need to be able to reconcile – that is, bring into agreement – ontologies that have been built in different contexts.

¹⁷ Some efforts have been done to build standard upper ontologies, but they have necessarily much information to represent, and thus are huge. This results in poor performance and in higher maintenance costs. See next chapter for more information about upper ontologies.

Chapter 3: Ontology reconciliation

In this chapter, we describe the different approaches that have been proposed to reconcile ontologies. We begin with a typology of the various heterogeneities between ontologies that these approaches have to solve.

3.1 Heterogeneities that hinder accurate ontology reconciliation

Ontologies are developed most of the time autonomously. The terminology used to describe a same domain will often differ from one ontology to the other. A same concept may be described by other relationships and attributes; a concept which is mainly the same may be understood somewhat differently in practice. These heterogeneities can be classified as terminological, modelling, and conceptual heterogeneities, depending of the (increasing) depths of disagreements [Chalupsky 2000], [Klein 2001]:

☞ *Terminological* heterogeneities are differences of notation, of terms chosen to describe the ontology entities. For example, the ontologies may define a same concept with a different label (synonymy), or define different concepts with the same label (homonymy); they can represent a same kind of data with different data types and units (currency, imperial system versus international system, etc.).

☞ *Modelling* heterogeneities are differences of modelling of an entity. For example, a same concept can be represented in different ways: a circle can be described by a point and a radius, or by a set of three points (difference of paradigm); the human gender can be represented by having two subclasses “man” and “woman” to “human being” or by adding an attribute “gender” with two possible values (difference of modelling convention). The ontology may also have been designed for a given purpose (e.g. to be used for an application), guiding how concepts would be represented. Terminological and modelling heterogeneities can proceed from the comparison between ontologies written in a different language¹⁸.

☞ *Conceptual* heterogeneities are differences of perception of what an entity that is considered the same imply in the reality. Ontologies may have a same concept, but have slightly different instances corresponding to the concept (difference of scope); for example, all administrations have a same understanding of the concept “employee”, but they associate with it distinctive rights and responsibilities. Ontologies may describe the same concepts with different levels of granularity; their coverage may intersect only partially; for example, we could find an ontology modelling red wine from the world, another covering all kind of French wine, and a last one focusing on Bordeaux wines, considering grape variety, vintage, chemical proprieties, etc.

¹⁸ Ontology languages differ not only in syntax, but also in expressivity (support to express negation, sets, default values, etc.), in how they allow to represent logical notions, etc. [Chalupsky, 2000]

3.2 Ontology reconciliation approaches

[Wache et al, 2001] determine three configurations to reconcile ontologies for achieving interoperability. The approaches described are in line with the standard for interoperability ISO 14258¹⁹:

☞ The first approach, “*integration*”, involves that ontologies should be “*integrated*” or “*merged*” into a single coherent ontology. This approach is interesting when there is a lasting need for interoperability²⁰, when the resources annotated by concepts of the reused ontologies can be retrieved and possibly re-annotated, and when there is no need to keep any relation with the ontologies “*reused*”. This approach is probably to be avoided where many actors are involved, as it becomes harder to find a consensus on design decisions, and when the domain described is large, because of the higher maintenance cost.

☞ Another approach, “*federation*” consists in relating ontologies by a process of “*ontology matching*”. Ontology matching is the process of finding correspondences between the entities of two ontologies. It consists in determining a set of similar ontology entities (concepts and roles), and relating them by mappings, to describe agreement between the ontology definitions. Ontology matching²¹ is also called “*ontology mapping*” and “*ontology alignment*”. This latter term may sometimes imply that the ontologies are modified after the process, in such a way that inconsistencies are removed. In practice, the two words “*mapping*” and “*alignment*” are interchangeable, particularly to name correspondences between the ontology entities. This approach is the most suitable approach to allow exchange among resources developed for independent purposes and which evolve independently. See [Giunchiglia et al, 2006] for an overview of the current matching technology. The subject is widely treated, notably in a book recently published [Euzenat and Shvaiko, 2007]²².

The *federation* approach has a variant, “*runtime ontology resolution*” or “*dynamic ontology matching*” in which the process of mapping search is centred on a few concepts instead of the whole ontology, and in which some restrictions apply, to limit the consumption of resources [Ding et al, 2007], [Giunchiglia et al, 2006]. Although the searching process might be different, the agreement between the two ontologies is done through mappings. This approach is fitted where the need for interoperability is limited in time, and the reliability is not the main concern. An example of this is for ontologies associated with agents that need to dynamically negotiate so as to find an agreement based on their respective ontologies. Some agents may modify their ontology following the collaboration, by machine-learning. [Giunchiglia et al, 2006] identify a few other cases of use of this approach, and examine the requirements for dynamic ontology matchers.

¹⁹ Standard ISO 14258, “Concepts and rules for enterprise modelling”, exposed in [Chen, Doumeingts, 2003].

²⁰ This approach is in common use, but mostly for means other than interoperability, as ontology engineers often try to reuse existing ontologies while designing a new ontology.

²¹ The term “matching” has come from the database domain, where it was associated with schema matching, i.e. establishment of n-ary correspondences between database schemas entities [Rahm and Bernstein, 2001].

²² A bibliography can be found on the book website <http://www.ontologymatching.org/publications.html>.

☞ A final approach, “*unification*” or “*mediated*”, consists in developing or modifying the ontologies according to some shared vocabulary or other rules defined in an “*upper ontology*”. This approach also requires ontology engineers to modify the ontologies in a consistent way. This approach is probably the most suitable one where there is a lasting interoperability need, and a restricted number of actors have to agree on the design decisions. This approach relies on ontology matching to relate the different ontologies, but there is already an agreement on some important concepts. [Chen, Doumeings, 2003] advice this approach for interoperability, for it limits the number of correspondences to find between resources. Also, the need of an expert to validate the mappings is not as important as in the previous case. [Mascardi et al, 2007] review various comparisons of the most visible upper ontologies in the research community²³.

Table 1 presents a synthesis of these approaches, and compares them on the basis of criteria of whether the result is consistent, whether a relation is kept with the ontologies sources, and how flexible the approaches are.

3.3 Conclusion

For semantic interoperability to be possible at a large scale, one must be able to compare different ontologies. As ontologies developed autonomously are heterogeneous at various levels, it is necessary to reconcile them accordingly. The approaches of integration, federation, and unification have been proposed, and each of them deals differently with heterogeneity: integration rewrites the ontologies, so as to remove the most disturbing heterogeneities. Federation lives with them, relying on the matching process to state what is comparable. Unification aims at avoiding heterogeneity where it is possible, by relying on design guidelines.

As the need is primarily for reconciling ontologies developed in an autonomous way, flexibility is the most important criteria, and federative approaches are the most relevant ones.

All federative approaches rely on ontology mappings²⁴. We believe that the need for increased reliability in the quality of the mappings produced will become a more important concern as ontologies are actually being developed and used. We will thus examine in the following chapter the process of ontology matching.

²³ See also [Schlenoff et al, 2000]’s study of upper ontologies to evaluate their interest for manufacturing, and [Semy et al, 2004 for US government and army domains.

²⁴ The integration approach relies also on ontology matching tools and methods, to assist ontology engineers when merging ontologies.

Table 1: Comparison of the different approaches for ontology reconciliation

	process	Consistency of the result	Relation with the sources	Most flexible approach	Preferred Use
Integration	Ontology merging	Yes	No	5 th	<i>Reuse existing ontologies to build a new ontology that legacy resources will refer to</i>
Federation	Ontology matching	No	Yes	2 nd	<i>Relate ontologies developed independently that will evolve independently</i>
	Ontology Alignment	limited	No	3 rd	<i>Relate ontologies developed independently</i>
	Dynamic ontology matching	No	Yes	1 st	<i>Negotiation among independent agents</i>
Unification	Development based on some conventions	Yes	Yes	4 th	<i>Develop many ontologies in a same (group of) organization(s)</i>

Chapter 4: Ontology Matching Methods and Tools

In this chapter, we present different methods proposed to recognize which entities defined in distinct ontologies are similar. Each method addresses some of the heterogeneities described in the previous chapter. We present also some of the semi-automatic tools that have been developed and that implement one or more of these methods.

4.1 Methods for ontology matching

We will now give a summary of the various methods used, mainly based on the review from the KnowledgeWeb²⁵ project [Euzenat et al, 2004].

Classification of ontology matchers and methods

A classification that provided a comprehensible overview of schema matchers by [Rahm and Bernstein, 2001] was adapted by [Euzenat et al, 2004] to classify ontology matchers and matching methods (Figure 6).

We modify the classification of methods so as to consider the source used; we also introduce a distinction between methods that work on the representation and those that work on the representation semantic. We add a class for representation-language-based methods following [Ferreira Da Silva, 2007], and for methods based on the ontology development context (Figure 5).

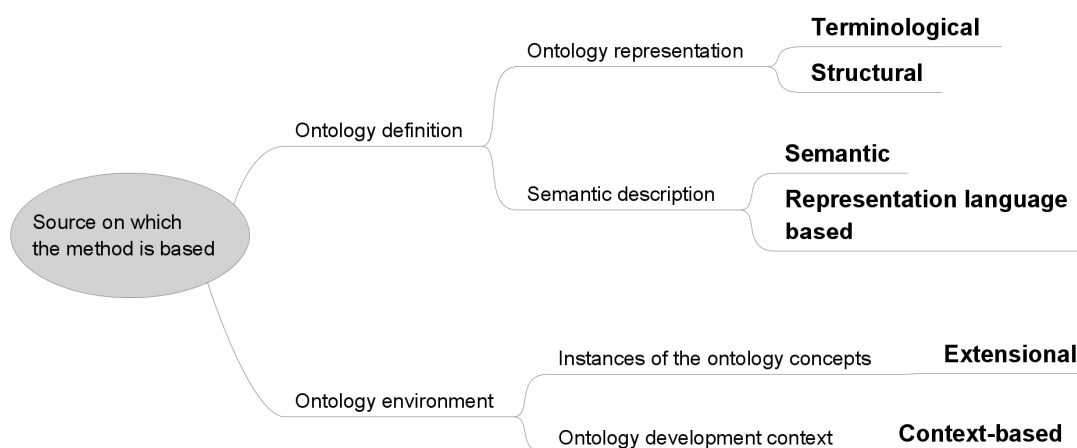


Figure 5: Classification of ontology matching methods

²⁵ The mission of the European project KnowledgeWeb is to help industry take up semantic web technologies. See <http://knowledgeweb.semanticweb.org/>

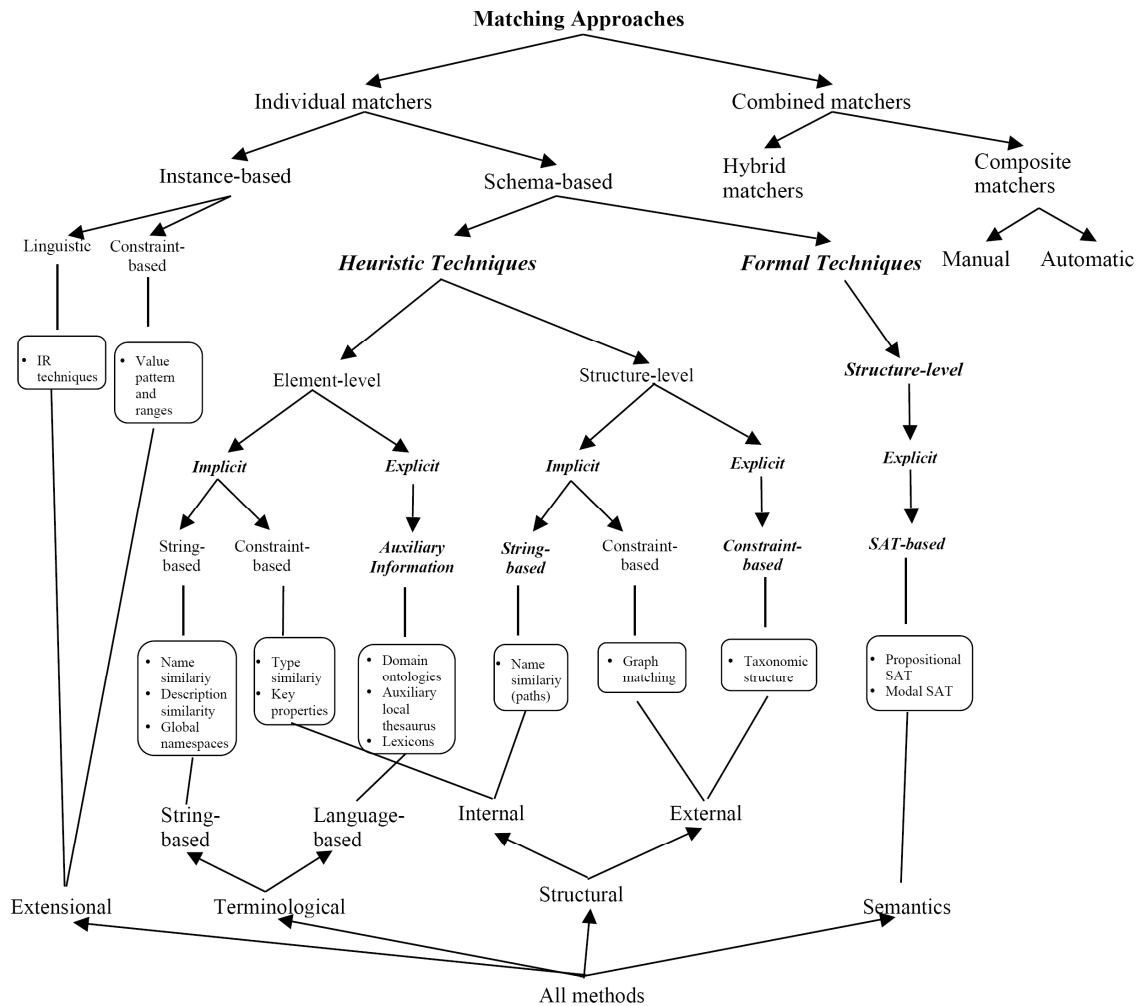


Figure 6: Classification of matching approaches and methods [Euzenat et al, 2004]²⁶

Methods based on ontology representation

Terminological methods compare the class descriptions based on the textual description of their name and label, and more rarely comments. Usually labels and class names will be normalized by removing accents, converting to lowercase, and so forth. Some further transformation can be done, such as concatenating the class name with all its super classes [Do et al., 2003]. A similarity is computed between strings based on a distance such as the Hamming distance (the number of different characters), the Levenshtein distance (the number of atomic operations to be performed to transform one string into another), and so forth. More and more tools make use of external linguistic resources, such as WordNet²⁷, to find for synonyms, homonyms, etc. Comments are commonly compared using token-based distances. The text is segmented in significant word (tokens), and is associated with a vector defining the number of times each token is found in the text. These vectors are then compared using a tf-idf²⁸ statistical measure, or

²⁶ Image reproduced with the first author's permission.

²⁷ Wordnet, "a lexical database for the english language", <http://wordnet.princeton.edu/>

²⁸ term frequency–inverse document frequency, is a statistical measure often used in information retrieval and text mining.

a similarity measure such as Cosine or Jaccard [Baeza-Yates and Ribeiro-Neto, 1999]. Texts can also be compared using Natural Language Processing (NLP) methods. Terminological methods are based on the assumption that classes are described in a human-understandable form, and are well documented. Most algorithms handle only one language, usually English. As people define ontologies using real world vocabulary, they say much more by the choice of these words than what they really define in the computer language. The drawback is that the human language is rich in nuances and context-dependent meanings. The comparison of ontology entities by terminological methods may therefore lead to erroneous interpretations and wrong conclusions.

Structural methods compare classes according to their “*internal*” (resp. “*external*”) structure, made of attributes (resp. relations with other classes). Attributes names are compared using string-based methods, and the data types are compared as well. The general purpose when comparing attributes is to create clusters of possibly similar classes, to limit the number of classes to compare. Algorithms are mostly based on the relation of subsumption (commonly called “*is-a*”). This is because it is a standard relation, widely used, and which by definition categorizes classes. The relation of mereology (or “*part-of*”) is less commonly used. Other relations are very seldom considered, and when they are, they are compared to one another mainly using a string based method. A correspondence between such relations can serve to prove equivalence between classes²⁹ (the reciprocal is also true) [Zhang and Bodenreider, 2004].

Methods based on the semantic content

Semantic methods compare the ontology entities according to what can be inferred from logic axioms. Different techniques are used, such as Propositional satisfiability (SAT), modal SAT, or description-logic-based techniques. These methods rely on a set of basic axioms between some entities of the different ontologies, given by experts or possibly furnished by one of the methods above. Most tools consider the sole axioms of subsumption, disjunction and equivalence. The tools employed traditionally return the axioms inferred (rather than traditional mappings). The relations discovered do not involve the notion of similarity, nor of confidence. Indeed, the axioms that serve as input for these methods are supposed to be always true, as well as the axioms defined in the ontologies. The inference engine finds logical conclusions which are supposed to be 100% true always. Employing fuzzy techniques, some return traditional mappings with a similarity value of 0 for an axiom of disjunction, 1 for an axiom of equivalence, and in between for an axiom of subsumption.

Representation-Language-based methods exploit the semantics of the representation language used. oMap [Straccia and Troncy, 2005] is the only tool based on such a method, for ontologies in OWL-DL. The tool compares complex concepts defined by operators (such as *union* or *intersection*) and/or by quantifiers (such as *for all*) and cardinality restrictions. The weight of the atomic concepts and relations used in the description are manually optimized for the relation between the operators or quantifiers

²⁹ If a class A (defined in O) is related to the class B (defined in O) through R, and the equivalent class A' (defined in O') is related to B' (defined in O') through R'; in such a case, if R is equivalent to R', then B and B' will be judged equivalent. The reciprocal (A equivalent to A' and B equivalent to B' imply R equivalent to R') is also true.

used. The tool requires mappings to be given as input. The method permits to compare complex concepts, which are seldom considered by other methods.

Methods based on the environment in which the ontology is constructed or used

Extensional methods compare instances related to the classes of the ontologies. Theoretically, if two classes are the same, then they should share the same instances. Practically, the corpus associated with the ontologies may be different, and thus also the instances for each class; additionally, the association of instances to the different classes is rarely exhaustive. When the corpus is different, instances are first compared (most commonly using terminological similarity methods). The sets of instances associated with each class can then be compared using dissimilarity functions rather than similarity functions, to determine how much dissimilar they are, and whether the possible equivalence between classes should be rejected or not.

Methods *based on the ontology development context* exploit the sources used for the ontology development to provide contextual information that could help the ontology matching. OntoBuilder is a system to assist the user to search a specific service (flight, car rental, etc) among different competitors on the Web [Gal et al, 2005]. Based on an initial ontology, the user can select the different Web sites he is interested in, and go through the website forms as if with a traditional browser. OntoBuilder then extract the forms' labels and types to build up an ontology, which it then compares to the original ontology. Further, it performs an ontology matching between the different ontologies, based notably on the order of the forms' presentation on the Web sites: the approach supposes that the importance and generality of concepts augments with their "*precedence*" in the business application.

We just discussed methods for ontology matching methods. Let us now consider the different possible stages of the matching process, and how the methods that we discussed are implemented in semi-automatic ontology matching tools.

4.2 Ontology matching process and tools

Most tools combine different matchers, in sequential or parallel composition [Euzenat and Shvaiko, 2007]. In a sequential composition, the first matcher output serves as input to the second. In a parallel composition, the results of the different matchers are compared using an aggregation operation such as triangular norms, multidimensional distances, and fuzzy aggregation to provide definitive mappings. The alignment process usually returns a similarity value superior to 0 (the concepts are distinct) and inferior or equal to 1 (the concepts are equivalents) for each possible mapping.

Mapping extraction consists in determining the ultimate alignment between the ontologies, from a set of mappings with a given similarity, given by any of the methods above. It involves checking the global consistency of the alignment by applying logical constraints to remove inconsistent mappings, such as when a class is asserted equivalent to two disjoint classes. It is often a manual or semi-automatic stage where a user or a community of users checks the results and judges what mappings are correct [Zhdanova and Shvaiko, 2006]. Recently some approaches proposed to combine matchers and extract mappings in a way that takes consideration of the possible weaknesses of the matchers combined, and of the uncertainty inherent to ontology matching.

Most ontology matching tools make use of terminological and structural methods. In [Euzenat et al, 2004]’s state-of-the art, 8 out of 20 tools rely on no other method. The ONION system [Mitra et al, 2002] and the Anchor-PROMPT tool [Noy and Musen, 2000] implement them both. [Ferreira Da Silva, 2007] presents a model based on the semantic method, and use fuzzy methods to return a similarity value between 0 and 1 when an axiom of subsumption is inferred between two ontology concepts. GLUE is an example of a tool based exclusively on an extensional method [Doan et al, 2003]. X-SOM is based on terminological and structural methods and uses a neural network to weight the mappings [Curino et al., 2007]. iMapper analyzes instances and enriches ontologies semantically, in order to help improve the precision of further ontology matching [Su and Gulla, 2004]. [Euzenat et al, 2007] give a good description of current tools, which they classify based on data from the contest for comparing ontology mapping tools³⁰, and on results from about 60 specific tests where specific alterations are made to the ontology to be mapped (example of alterations include labels replaced with synonyms, instances removed, class composition expanded or flattened, comments translated in another language or removed).

A good indicator on how the field is active, 9 tools have been added in the update [Euzenat et al, 2007] compared to the report [Euzenat et al, 2004] released three years before. Examples of these include H-Match [Castano et al., 2006], which aim at finding mappings between ontologies in peer-to-peer environments, and oMap [Straccia and Troncy, 2005]. [Sunna and Cruz, 2007] and X-SOM have been released since.

The mapping extraction stage is included by most alignment tools, traditionally with a threshold value to determine which mappings to keep and which not. [Meilicke and Stuckenschmidt, 2007] propose a comparison of three algorithms for mapping extraction. [Haeri et al., 2007], [Qasvinian et al, 2008] introduce a “coincidence-based” weighting mechanism to score the result of the matchers, and inject the weights obtained in a genetic algorithm system. [Nagy et al, 2007] and [Laamari and Ben Yahghlane, 2007] use a belief function to deal with the uncertainty inherent to the matching results it combines so as to improve the precision of the mappings. [Gracia et al, 2007] consider the properties and attributes of the anchor terms as well as the external resource Wordnet to validate “ambiguous” mappings.

Although mappings should also relate relation to relation, and instance to instance³¹, most ontology matching tools only produce mappings between concepts, at the exclusion of any other entity. The majority of tools return binary mappings (relate only two ontology entities) that express a symmetric similarity (the similarity of the first entity with the second is the same as the reciprocal).

Most alignment tools are based on the sole subsumption relation, and do not consider the possible expressivity of ontologies. The ontology couples to match proposed at the

³⁰ Ontology alignment Evaluation Initiative, campaign <year> <http://oei.ontologymatching.org/<year>/>

³¹ [Ghidini et al, 2007] introduced a mapping language that allows establishing semantic relations between heterogeneous components (concept to relation, for example).

Ontology Alignment Evaluation Initiative³² (OAEI) annual campaigns never included anything else but taxonomies and ontologies from the medical domain, which mainly consist of a huge hierarchy.

4.3 Mapping formats

Most matching methods return mappings expressing the similarity of concepts with a magnitude given by a float value between 0 (for incompatible) and 1 (for equivalent). Semantic-based matching methods usually return the axioms (of subsumption, equivalence, etc.) that they find, but some incorporate approaches to transform these axioms into traditional mappings using fuzzy techniques.

OAEI requires participants to use a same model for alignments [Euzenat, 2006], with 3 possible levels, from the straight-forward one-to-one mappings (level 0) to more complex relations established by rules in a language such as OWL, SQL, F-Logic (level 2). The Table 2 gives an example of alignment using this format, for the level 0.

Table 2: Mapping format proposed by OAEI (level 0) [Euzenat, 2006]

```
<?xml version='1.0' encoding='utf-8' standalone='no'?>
<!DOCTYPE rdf:RDF SYSTEM "align.dtd">
<rdf:RDF
xmlns='http://knowledgeweb.semanticweb.org/heterogeneity/alignment'
xmlns:rdf='http://www.w3.org/1999/02/22-rdf-syntax-ns#'
xmlns:xsd='http://www.w3.org/2001/XMLSchema#'>
<Alignment>
  <xml>yes</xml>
  <level>0</level>
  <type>**</type>
  <ontol>http://www.example.org/ontology1</ontol>
  <onto2>http://www.example.org/ontology2</onto2>
  <map><Cell>
    <entity1 rdf:resource=
      'http://www.example.org/ontology1#reviewedarticle' />
    <entity2 rdf:resource=
      'http://www.example.org/ontology2#article' />
    <measure
      rdf:datatype='&xsd;float'>0.6363636363636364</measure>
    <relation>=</relation>
  </Cell></map>
  <map><Cell>
    <entity1 rdf:resource=
      'http://www.example.org/ontology1#journalarticle' />
    <entity2 rdf:resource=
      'http://www.example.org/ontology2#journalarticle' />
    <measure rdf:datatype='&xsd;float'>1.0</measure>
    <relation>=</relation>
  </Cell></map>
</Alignment>
</rdf:RDF>
```

³² The Ontology Alignment Evaluation Initiative (OAEI) <http://oei.ontologymatching.org/> continues the series of workshop launched by the I³CON conference, <http://www.atl.external.lmco.com/projects/ontology/i3con.html>

In level 0, mappings are associated with a type of relation (such as equivalence, subsumption, and incompatibility) and a float value between 0 and 1. This value conveys the confidence held in the mapping. The format requires the matching tool to declare the arity of the mapping (it is not expressed though whether it is due to a limitation of the matching algorithm, a decision, or if it is due to the representation of the ontologies matched).

4.4 Conclusion

Ontology matching methods and tools have been flourishing these last years. Most tools are based on the ontology definition, exploiting the terminology used and the graph structure of the ontologies to detect similarities between them. A few tools consider the semantic expressed in the ontologies, and use inferences to detect new relations between ontology entities, based on a predefined set given by experts. A few tools exploit information external to the ontology definitions, such as concept instances or information from the sources that were used to build the ontologies.

Despite all the efforts invested in ontology matching methods and tools, the results are still disappointing. Consequently, some propose to use sophisticated methods such as neuronal networks to gather the respective strengths of all methods; some propose to remove subsiding ambiguity by letting users express their confidence in the mappings returned; some propose to evaluate matching methods more objectively, launching evaluation campaigns where couples of ontologies are to be aligned, and the result returned in a common mapping format and compared to a benchmark.

The main argument for ontologies is that they define the semantic of terms used in applications. When a need arises for some ontologies to be related to one another, while being kept independent, ontology matching methods and tools are used to map the ontology entities. Yet, contradictorily to what would be expected – ontologies being artefacts that define meaning with formal logic, ontology matching is a highly heuristic process. Terminological and structural methods rely on subjective choices made by the ontology designers. The semantic methods do actually consider the semantic described in the ontologies, but ontologies are most of the time inconsistent with one another, which reduce to void the gains of reliability and automation that were expected.

The meaning of mappings is immediately related to the evaluation of similarity between concepts. The ground for this evaluation fixes the real meaning of the mappings. If mappings have been generated by matching automatic tools based on terminological and structural methods alone, then concepts will be judged equivalent if they are named the same, or have similar relations. If the alignment is verified by experts, one can suppose that the concepts express mainly the same idea, although probably with different points of views, as ontologies are usually developed for different applications.

With the current representation of ontologies, there is no reliable way to match ontologies together. The methods and tools presented above can be used to alleviate the work of experts when integrating ontologies, or to match taxonomies automatically, where accuracy and exhaustiveness is not of concern. In all other cases, we argue that the approaches presented above rely on an assumption that the different ontologies

represent concepts and relations in a very similar way, and that the only things that really vary among ontologies is how concepts and relations are named and labelled. We will discuss this assumption in the next chapter, as well as the impact that the issue could have on the validity of ontology mapping and on the consideration of ontology matching as a means to achieve semantic interoperability.

Chapter 5: Limits of Ontology Model and Development

We present in this chapter many reasons why ontology matching lacks reliability, based on the shortcomings of the ontology model, on the specificity of ontological representation and on the absence of regulation for ontology development.

The way ontologies are designed, developed and evolve has a strong impact on their content, and how they represent it. This in turn will affect the way that they are mapped with other ontologies. Design decisions are made “*with respect to what aspects of the world are relevant*” [Stuckenschmidt, 2006]. The evaluation of this relevancy is left to knowledge engineers, who judge in view of the applications the ontologies have been developed for; but the rules that they will use and the knowledge that will guide them are influenced by their culture, organizational culture, working experience in the field, and so on. In this exercise, they are limited by the tools and their (lack of) experience in ontology development, and possibly the level of expressiveness of the ontology language; if the ontology development is done by ontology engineers, there can still be all sorts of communication problems between the ontology experts and the domain experts.

There is no bijection between the levels impacted and the type of the source of the impact. The Table 3 below shows my attempt to draw a relation between application, point of view, design rules (e.g. internal to an organization), etc. and the impact that they have on ontology design. I will therefore use the term “perspective” from now on to represent any particular background that is recognized as having an impact on the design of the ontology.

Table 3: Kinds of perspectives and their respective impact on ontology design

What drives changes	Design rules	Technical purpose	Application	Point of view
Level impacted				
Terminological	■■■■	■■■■	■■■■	■■■■
Modelling	■■	■■■■	■■■■	■■■■
Conceptual	■	■■	■■■■	■■■■

■■■■ strong, ■■■ medium, ■ weak impact

5.1 Weaknesses of the ontology model

Here we show how the traditional building bricks used to model ontologies, that is classes and relations including *is-a* and *part-of*, presents some characteristics of the ontology model that make it difficult to determine mappings between ontology entities.

☞ **The meaning of concepts is not described formally**

There is no way offered to describe the meaning of a concept or relation other than the human language. Yes indeed, concepts are related to each other in some particular way, but that is not enough to give any relevant information about a new concept. Even if all concepts and relations were precisely defined, it would not be enough to determine in all case what a concept newly defined relatively to the other concepts by some relations would represents exactly.

☞ **There are different ways to represent a same conceptualisation**

Ontologies represent a conceptualization with the help of concepts, attributes and relations. But it is not as if these building elements were completely separate. Instead, depending on the perception that the knowledge engineer has of things, they can almost be interchangeable. For example, an attribute *price* can be changed into a relation *hasPrice* and a concept *Price*. Figure 7 shows an example of automaton that could be represented in two ways: with a concept for each state and a relation for the result of each action on the state; or with classes for each action, using attributes to show the result of the action for each state.

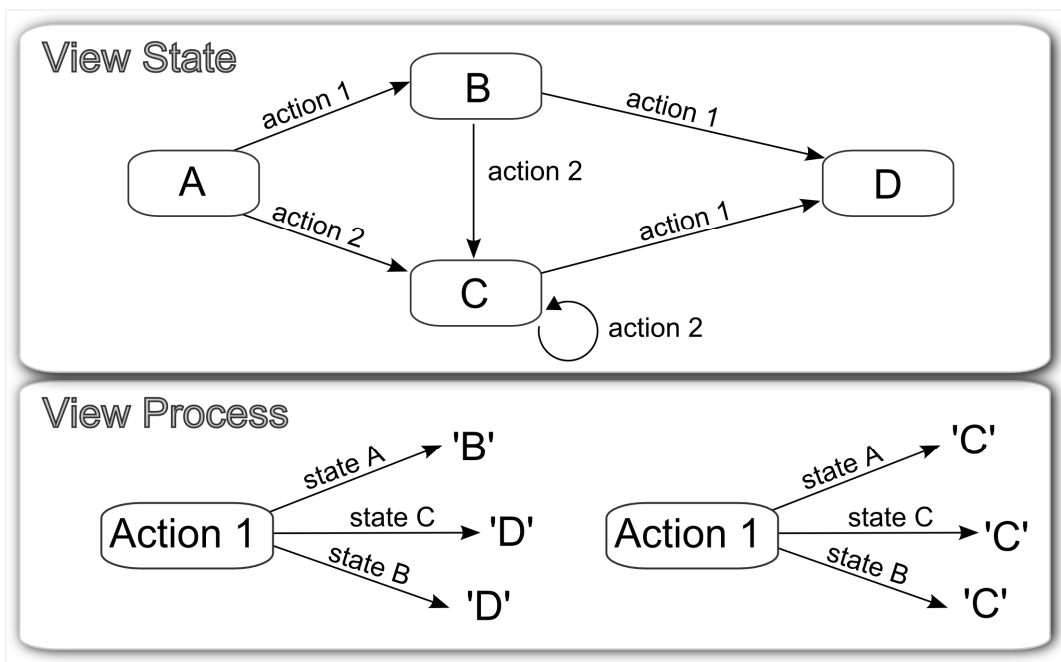


Figure 7: Modelling with the view "state" or "process"

☞ **The meaning of the subsumption relation is ambiguous**

The relation of subsumption (commonly called *is-a*) is “the inclusion or placement within something larger or more comprehensive: encompassment as a subordinate or component element <red, green, and yellow are subsumed under the term ‘color’>”³³. The notion of subsumption does not specify any means by which we could evaluate that a concept is actually more comprehensive than another. Rather, the term “placement” would seem to show that it is more a choice than an evaluation.

³³ Definition from the Merriam Webster Dictionary, <http://www.merriam-webster.com/>

A definition for logic gives subsumption as “*the minor premise of a syllogism*”³⁴. This second definition describes subsumption as an assumption (1) that states that one type could be described by another type more general, (2) that is considered as always true, and (3) that should serve to draw some conclusion. In the same way as for the first definition, this does not say anything about what could or could not actually be stated; instead, it will be highly dependent on the perspective that the ontology engineer has on the subject.

Declaring a subsumption between two concepts is very limited information: it says that one is more general than the other. It does not say how. There is thus no rule given whatsoever as to how to choose subsuming concepts. In the same way as users usually classify their documents in a multitude of different manners, there are different possible ways to choose subsuming concepts, as well as the order between them. The following example Figure 8 shows three possible classifications of the concept `Dissertation` [Klein, 2001]. In an ontology, the relation *is-a* can carry multiple meaning, what [Guarino and Giaretta, 1995] call “*is-a overloading*” [Kingston, 2008].

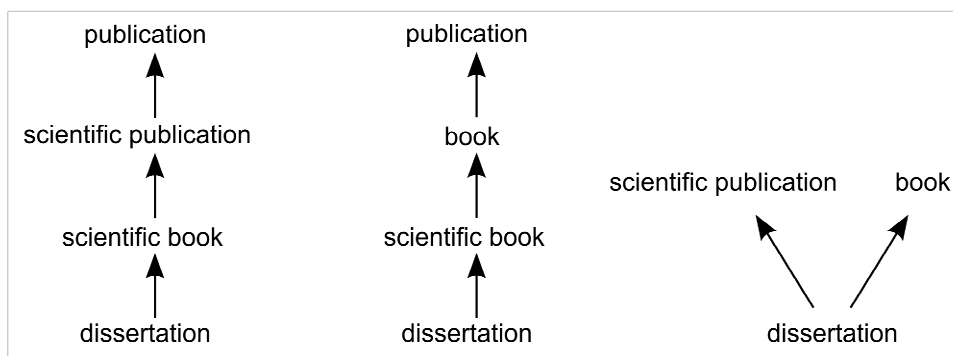


Figure 8: Various classifications of concepts [Klein, 2001]

Finally, the meaning expressed by the subsumption relation can sometimes be modeled differently, for example using an attribute. Some people are left-handed, others are right-handed; this can be modelled either with two classes `left-handed_person` and `right-handed_person` or an attribute `isLeftHanded`. One could model `Red`, `Green` and `Yellow` as three concepts subsumed by the concept `Color`. But one could consider also that color is best determined by its wavelength interval³⁵, and that `color_red`, `color_green` and `color_yellow` are each instances of `Color` with a different wavelength (Figure 9).

³⁴ The Free Dictionary <http://www.thefreedictionary.com/subsumption> . For a syllogism such as “*All dogs are animals, foxhounds are dogs, therefore foxhounds are animals*”, the definition gives subsumption as “*All dogs are animals*”.

³⁵ See Figure Color, wavelength, frequency and energy of light, on <http://en.wikipedia.org/wiki/Color> (consulted April 26th, 2008)

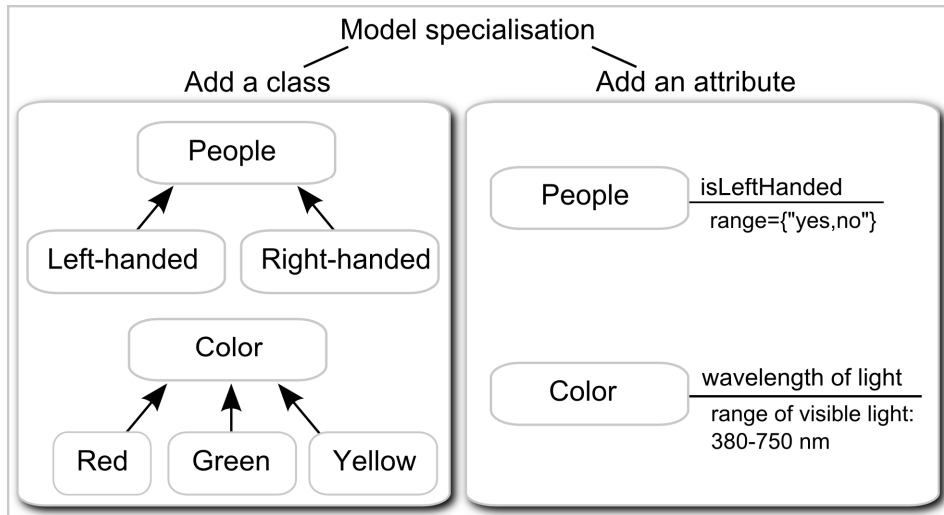


Figure 9: Introduction of a distinction with new classes or new attributes

The subsumption relation does not have atomic significance. Instead, it is highly dependent on the perspectives that drive the ontology development. Most ontology matchers rely on this relation, though. If there is a relation of equivalence between two concepts, and that one has sub-classes, and the other has an attribute, ontology matchers might be able to make the relation. But what if there are more than one attribute? The comparison of ontologies with different classifications, as in Figure 8, will be considered as inconsistent and may prevent the recognition of mappings between equivalent concepts.

☞ The granularity is not homogeneous

The standard relations *is-a* and *part-of* are transitive³⁶ and can therefore express a relation of subsumption between two concepts very close or very different. These relations do not carry any notion of granularity. For example, an ontology **A** could state *Swallow is-a Bird* and *Bird is-a Animal*, and an ontology **B** could state only *Swallow is-a Animal*, while being completely consistent with one another. These relations don't have the same granularity, making it harder to compare concepts from different ontologies. Indeed, if we compare the ontology **A** with an ontology of the same domain, where we have *x* (found to be equivalent to *Swallow*) *is-a y is-a z* (found to be equivalent to *Animal*), we cannot infer anything about *y* other than *Swallow is-a y is-a Animal*. We cannot state any relation between *x* and *y*, such as *y is same-as Bird*, *y is-a Bird*, or *Bird is-a y*.

Ontology engineers usually aim at having all *is-a* relations having as object a same concept express a similar granularity, so that siblings (concepts related to a same concept by the relation *is-a*) are really siblings. Non-expert naturally respect that rule most of the time. Yet, there is no evaluation possible of this. And even where this rule is respected, granularity varies along the *is-a* hierarchy. An evident example being the most abstract concepts under the standard most abstract concept **Thing** (**Top** in

³⁶ “a binary relation *R* over a set *X* is transitive if whenever an element *a* is related to an element *b*, and *b* is in turn related to an element *c*, then *a* is also related to *c*.”, see http://en.wikipedia.org/wiki/Transitive_relation

description logics) that includes everything: how to compare the abstraction level of the concepts **Process** and **Time**, for example? Also, from an ontology to another, the degree of specialization differs, making the granularity vary widely. One could imagine that this could be done by evaluating the degree of abstraction of the concepts directly under **Thing**. But a very specialized ontology can include some very abstract concepts.

The granularity varies also between among groups of concepts: ontology concepts can be divided into at least three categories of concepts: (1) the concepts of concern for the application, (2) the generally more abstract concepts that are usually necessary to describe the first concepts, and (3) the other concepts that are not useful in themselves, but presented for the sake of relative completeness (helping for example to understand the scope of some concept relevant for the application).

The concepts from the first category are described in some detail, as needed by the application. The second category concepts are not as relevant, and may be described at a more abstract level, or they may be references to concepts defined in other ontologies. The concepts from the third category may be described minimally, not at all, or be absent (so there is no consideration about granularity for them). Globally the granularity will be higher for the concepts of the first category than for the concepts of the second category. The granularity may also vary inside these categories depending on the focus of the application.

5.2 Lack of precise specifications for ontology modelling

The definitions of ontology that we saw in the first chapter describe ontology artefacts mainly by what they should represent. There is no general consensus on what ontologies should look like, technically. To begin, there is no simple way to “recognize” ontologies. Usually, the language used (such as RDF, OWL) can be a good indicator, but an RDF file could contain only instances. Heuristics can rely on no more than the presence of many relations *is-a* and *part-of*, or the presence of a comment “*This is an ontology*”.

☞ The terminology is not restricted

The knowledge engineers are free to specify the conceptualisation as they judge adequate:

- They select the ontology language of their choice.
- They relate concepts through relations of any granularity.
- They choose concepts of whatever degree of specialization, with whatever label and attributes.
- They give textual definitions with the language of their choice, and even not to give any comment or any human-understandable label.

Organizations usually have some rules and methodologies that reduce the freedom that ontology engineers have, giving good design principles. Though, they do not rule everything, and no methodology is generally accepted.

☞ There is no way to distinguish different “worlds”

Ontology concepts can represent concepts of various levels of abstraction and of virtuality, from philosophical concepts, like **life**, **happiness**, **suffering**, to categories

of objects in our world, such as `chair`, `table`. In a single ontology, nothing prevents to have an instance of the concept `book` related to an instance of the concept `author` and `story`, and to have also an instance of the main `actor` (in the `story`) and the `book` that she is reading. Thus one can have in the same ontology concepts or instances that refer to different “worlds”. No means is provided to differentiate these different “worlds”, there is no mean given to define, for example, that there are a “real” world and a “story” world and to fix (for example) the relations that allow to pass from one “world” to another.

☞ Lack of standard relations

There are very few standard relations between concepts: in OWL-DL, for example, they are limited to *is-a* – with the flaws that we just pointed out, *part-of*, and *same-as*. Ontologies will necessary introduce new concepts; if both concepts and relations are totally new, ontology mapping can rely only on error-prone heuristics.

Today, ontology matching tools do not use any relations between concepts when comparing ontologies other than *is-a* and *part-of*. The mapping of relations is indeed even more complicated as the mapping of concepts, partly because concepts are at least structured by the *is-a* relation, while relations are almost never defined relatively to one another. Relations are probably also more specific to the application for which the ontology is developed.

Ontology matching can therefore rely on the structure established by sole standard relations³⁷ between concepts to find relevant mappings. [Zachman, 1987] propose a set of candidates for such relations.

5.3 Limitations of present development tools, methods, and practices

Ontologies bring together the connectivity of the Web and knowledge modelling. Knowledge modelling has a long history, but ontology development is a relatively new field. This has an impact on the quality of the ontologies developed.

☞ Lack of “standard” ontologies

[Rahm and Bernstein, 2001] support that *“frequently used entities, such as address, customer, employee (...) should be defined (...) in a (...) library”*. Yet, apart from the medical domain, where some ontologies begin to emerge as, and the Dublin core, there is no ontology in my knowledge that is recognized a “standard” for its domain, and to whose entities other ontologies could refer [Bouquet et al, 2007].

☞ Ontologies are developed in isolation

This is probably one reason why ontologies usually include very few relations with other ontology entities but instead define almost every concept used. Most ontologies are defined with knowledge representation languages designed according to the open world assumption³⁸ but developed in isolation, as in a close world. *“[They] are rarely built to be shared and reused”* [Paslaru-Bontas, 2007]. The comparison of ontology

³⁷ as well as other relations defined relatively to these standard relations

³⁸ See http://en.wikipedia.org/wiki/Open_World_Assumption

entities relies on no common reference, and is therefore entirely based on heuristics. Does the power of the Web not lie in the links that relate the different websites?

☞ **Lack of a powerful development interface**

The more known and effective tool³⁹ for ontology, Protégé⁴⁰, has a flawed editing interface, navigation based only on *is-a* hierarchies, and very limited visualisation. There is no automatic search for concepts in a choice of ontologies on the Web, only one embedded ontology matching tool, no default ontology versioning, to cite a few. Since the connection with other ontologies is not assisted and encouraged, ontology engineers usually focus on consistency alone, which is already a time-consuming task.

5.4 Complexity of the reality to model

When the ontology is to model what exists (“descriptive”) rather than what should exist (“prescriptive”), the ontology engineers are faced with the general complexity of the world: it is merely far more complex than what we usually imagine. Nature seldom fits in our categories, but is very diverse, with continuous variations, that do not always fit a straight digitalisation. This is also true to a certain degree with human cultures, which evolve continually in space and time, and impact the way people think and act. Unless codified, the realisation of a product will probably be described by a different process in different countries even in a same organization.

Not only reality is complex, but also one perceives it according on ones culture. Every ontology engineer has another perception of the concepts to represent, their relative importance, and the possible connections with other concepts. Various upper-level ontologies have been developed, such as Cyc and SUMO, with the objective or representing the more generic concept. Figure 10 shows the most abstract concepts of these ontologies, which are all different.

Reality is complex, and cannot be represented exhaustively, but instead a selection of concepts and relations must be done. Most of the context that usually serves to disambiguate terms and situations is lost. Logic based representations are well adapted when defined assertions are supposed to rule how things are to be. These representations have not yet been proved to be satisfying for approximating a complex world that is not fully comprehended. Table 4 presents some oppositions between the world we live in and the representations in formal logic.

☞ **Ontologies are a compromise between a definition of entities in formal logic, and a knowledge representation understandable by human beings in a culture**

The ontologies should represent knowledge in a way that has to be computer-processable, but also comprehensible by other human beings who are most of the time immersed in a culture comparable to the authors’ one.

³⁹ The development of Swoop was stopped, GrOWL has a limited editing interface and its visualisation scheme, though appealing, is not very helpful.

⁴⁰ See <http://protege.stanford.edu/>

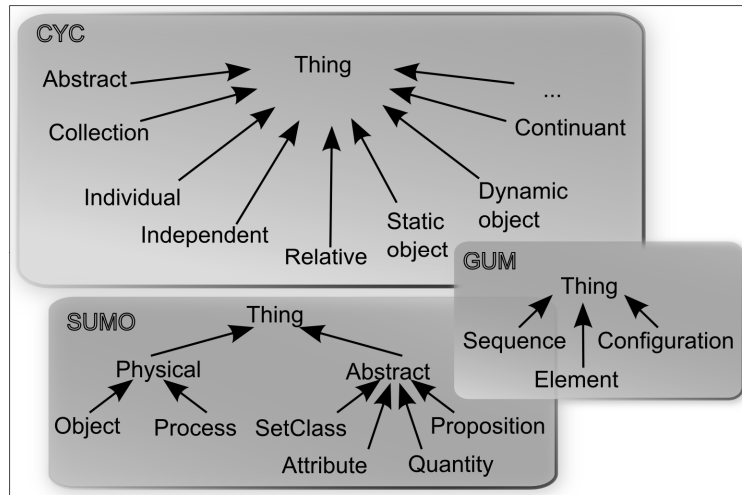


Figure 10: Most abstract concepts of various upper-level ontologies

Table 4: The world we live in and obstacles to its modelling with formal logic

This world	Formal logic	Matter
We always are somewhat wrong in what we think we know	Assumptions are supposed to be completely true.	Information reliability and consistency
We usually do not represent things we know in the right way		
Assumptions are mostly overgeneralizations	Assumptions are supposed to be always true (no exception)	Circumstantial validity of information
Complex, nothing is simply black or white. There is not necessarily a “right” way to describe things	Inconsistency is not allowed	Variety of organization of information

5.5 Conclusion

Ontology matching suffers from many shortcomings of the ontological model. This model is adapted for reasoning, but not much for comparing different models with one another. It is adapted to represent information that is under control, but not – as it is often the case – information that is complex and subject to different points of view. Finally, the tools to develop ontologies focus on ontology consistency, and not much on designing for sharing and reuse.

There is no formal paradigm that explains the general structure of the ontology, by which the ontologies could be compared. Ontologies are partial representations of a complex world. Depending on the task for which they are developed, the conceptualisation – selection of concepts and relations with their particular characteristics – will be organised differently. There exists to our knowledge no rigorous method to get a unique – and always valid – conceptualization of a domain. Instead, each choice will be subjective, and will follow a different paradigm, more adapted to the task. This will result in different representations. When comparing ontologies of intersecting domains, these different representations are compared. The paradigm used for the ontology development is not represented in the ontology, at least

not formally. The method used to convert the conceptualisation into a representation is not included either.

The meaning of entities is not defined formally, and therefore cannot be compared formally. Ontology languages permit the categorization of concepts in a logic way: entities are assigned a unique URI, by which an entity can be referred to, and recognized, even though used in different ontologies. But there is no way to compare ontology entities based on logic, unless the entities are actually identical, or are somewhat constructed with identical entities. There exists no standard library of concepts, and no rule of concept construction to define precisely and formally new concepts from existing ones.

Mappings of equivalence have a limited interest if the tasks for which the ontologies have been developed are not compatible. When entities are judged equivalent, they are most of the time equivalent in isolation from the rest of the ontology: the entities were probably included for distinct reasons, and thus the relations with neighbour entities are different. When using equivalence mappings, one expects to be able to replace one entity by the other, for a given task. This makes no sense if the tasks for which the ontologies were developed are totally incompatible.

Chapter 6: Problem Statement: Reconciliation of Independent Ontologies through Mappings is pragmatically Inconsistent

In this chapter, we present different shortcomings of reconciliation by ontology mappings. We demonstrate the need for the consideration of context in order for the ontology reconciliation to be more reliable. We then present the thesis statement, and give the scope of the research.

To achieve semantic interoperability between applications developed autonomously, one has to reconcile the ontologies that describe the meaning of the data manipulated by these applications. Among the configurations proposed to do this, integration and unification require the modification of ontologies, and a lot of expert work, for each reconciliation. Only the federative approach is adapted for the reconciliation of ontologies developed independently. In this research work, we will focus on interoperability among resources described by ontologies, following the federative approach, that is, by ontology matching.

Even within a single domain, ontologies are built with various concerns and points of view, and for different applications. These factors influence the meaning of relations and concepts, resulting in irrelevant results when comparing ontologies with mappings. Musen and Day gave a presentation for the BioPortal earlier this year, giving the example of a mapping between the concept “blood” of an ontology of zebra fish anatomy (popular aquarium fish) and the concept “blood” of an ontology of adult mouse anatomy. What does such a mapping mean? If some zebra fish blood was transferred into the adult mouse blood, would that not lead to complications?

6.1 Limitations of ontology mappings for a real ontology reconciliation

Ontologies are written according to different perspectives, which depend on the application considered, the point of view of the author, etc. They influence the scope of the ontology, its granularity, and so forth. Yet this information, which is necessary to understand how the ontology is built, is not included into the ontologies.

☞ **The significance of the mappings is limited by their format “one by one”**

Mappings display a connection between a few entities. In the same way as the translation of a sentence gives generally a poor result when words are translated one by one, the reconciliation of ontologies by ontology mappings is highly flawed, losing most contextual information. Yes indeed, extensional methods take in consideration the practical use of ontologies; but instances are not always available, and if it is the case, there is no evident way to infer from these whether a given mapping is of interest or not in some given circumstance.

☞ There is no standard rule to establish a degree of similarity (resp. confidence) between concepts

This degree of similarity of mappings summarizes the similarity of the concepts. A value of 1.0 is normally not ambiguous, as it means that theoretically the concepts are equal in all aspects. But a value between 0.0 and 1.0 may signify that the attributes and relations represented for both classes are only partially comparable, that the concepts have a different scope – the value giving the proportion of the scope intersection, and so forth, or a mix of these. There is no standard rule to decide which value should be attributed to any of these measures, or to normalize their results into a unique value.

The traditional mapping format associates a degree of confidence to the mappings, giving an estimation of the trust that one can have in the validity of the mapping. There is no standard method to state this confidence: it might be the confidence in the method used, an estimation from an expert of his/her assurance that the concepts are similar to a certain degree, a degree of acceptance proposed by [Paulheim et al, 2007], and so on, or a mix of these. Therefore there is low reliability in the precision of this degree of confidence. Moreover, this degree of confidence is often relatively small, such as 70%, being therefore a indication of mistrust than of trust. Who would use a service that gives 30% of wrong answers, with no way to know when the answer is wrong?⁴¹

The values of similarity and confidence give a summary of how the concepts are related. But too much is lost in the process. If two concepts are found to be partially similar, this could be used to relate the concepts in an appropriate way, especially if comparable attributes are known. But with a single value of similarity, you lose that information. Knowing that two concepts have a different scope may help to decide whether in a particular case the concepts are similar or not. This is not possible with a unique value of similarity.

6.2 Calls for context-based ontology reconciliation

Semantic interoperability is context-dependent. Yet, ontology reconciliation is done once for all, and ontology mapping is a heavy process which is done before the moment of the request.

☞ Interoperability is context- and task-dependent, but no contextual information is traditionally taken into consideration when creating, evaluating or using ontology mappings

[Ouksel and Sheth, 1999] state that semantic interoperability should support context-sensitive information requests that hide all levels of heterogeneity of information systems, and to limit information overload. Also, “*an important consequence of associating abstractions or mappings with context is that the same two objects can be related to each other differently in two different contexts*”.

⁴¹ And that is the best case. If one uses the information associated with the concept retrieved, such as attributes and semantic relations, the risk of getting something completely irrelevant is dramatically increased, as (in)compatibilities between the concepts’ attributes and relations are not recorded along with the mappings.

[Giunchiglia et al, 2006] argue that “*semantic interoperability is highly context- and task-dependent*”, and call for task-oriented dynamic ontology matching. Assuredly, all data is not useful for all task, and all data is not appropriate in all context. To be an answer for semantic interoperability, it is therefore essential that the context of development of the ontologies mapped, the context of the enquiry for interoperability, and the task that triggered the need of information [Byström and Hansen, 2002] should be considered.

When a need for interoperability occurs, it is in a particular context, and the result expected of the request is intended to be used for a particular task. Ontology reconciliation methods should therefore take this information into consideration, to compare it with the context of the data sources interrogated, so as to judge of the relevancy of the data.

Yet, the context of ontology development, including the task for which the ontology has been developed, is usually not explicitly stated in any document that could serve as reference. No matching method or tool uses this information to improve the relevancy of ontology mappings⁴². And contextual information of the interoperability need is not known yet when mappings are established.

☞ Similarity is context- and task-dependent, but the similarity of mappings is computed once and for all

Mappings are supposed to express a similarity between concepts. But according to [Rada et al, 1989], the result of similarity is different dependent on which properties of the objects have been considered for the computation of similarity. Yet, depending on the context, all properties are not equally relevant. So, the similarity between concepts should be determined according to the most relevant properties.

Apart from dynamic ontology matching⁴³, the similarity of mappings is asserted once and for all situations. As the task that generates the information need varies, the similarity between concepts should vary accordingly.

6.3 Thesis statement

The purpose of this study is to evaluate whether the consideration of context can lead to the improvement of reliability of ontology mappings for interoperability.

6.4 Research questions

- What is contextual when utilizing ontology mappings for interoperability?
- What improvements can be expected by the consideration of context?

⁴² [Paslaru-Bontas, 2007] is the only proposal that we know to take these contextual information into consideration; but her context-based approach is limited to the selection of relevant ontologies for reuse in the development of a new ontology.

⁴³ Dynamic ontology matching methods discover mappings after the information need is known, and when one knows how the information retrieved will be used. Yet, such methods are not reliable yet, and the mappings that they generate, being not verified by experts, cannot offer much guarantee of reliability.

6.5 Significance of the study

Some of the problems cited above will always exist: the problems related to ontology representation will not be solved before at least a few years; if one day they were solved, it would take a few more years for ontology matchers to be adapted to these new representations. During that time, traditional ontologies will still need to be mapped.

To our knowledge, this study is the first one that explores methodologically how contextual information can be used to improve the way ontologies are related to one another through ontology mappings. If this problem is not addressed, then unless someone discovers a way to describe the paradigm used to organize concepts in an ontology, and a method to compare ontology entities that take advantage of it, the use of ontology mappings will be limited to an approximate search of possibly relevant data, and will fail as an answer for semantic interoperability.

6.6 Research scope

Our research work is not concerned with the discovery of mappings between ontologies. We want to focus on relating ontology entities once the process of finding, validating and possibly evaluating mappings is complete. Also, we will restrict our study on one-to-one mappings that express a similarity between concepts, along with a confidence value.

The notion of context and similarity are object of many research in psychology, philosophy, sociology, etc. as ontologies are deeply connected with human perception and conceptualization. Yet, we will limit our review to the most evident aspects of the notion of context and similarity in human sciences that will suffice for our purpose.

To limit the problem to the smallest significant problem, I consider that ontologies are all written in the ontology language OWL, sublanguage OWL-DL, which is recommended by the W3C. It is to be expected that if any other language takes the advantage, there will be conversion tools available

PART II – State of the Art on Context and Similarity

In this part, we will review the literature on diverse subjects that could improve ontology reconciliation. First, we will examine frameworks proposed specifically with the aim of improving ontology reconciliation, or at regulating ontology development, which would also improve the validity of ontology reconciliation. Then, we will view a panorama of context-based approaches in Computer Science, in order to observe some general principles in all these approaches, and to determine whether a context-based approach could indeed improve ontology reconciliation. As ontology reconciliation implies a relationship between ontology entities, we will review the measures proposed between concepts defined in the same ontology or in different ontologies. Finally, we will review the approaches that consider contextual information when comparing concepts with one another.

Chapter 7: Frameworks for improved ontology development and reconciliation

In the following chapter we will see a panorama of diverse frameworks, architectures, and representations that have been proposed to improve in some way the interoperability between ontologies developed autonomously. These authors have proposed categories to describe more precisely the semantic implied by the concepts and semantic relations defined in the ontology; or to make use of contextual information to relate ontologies more efficiently. We will critique the different efforts and put them in perspective at the end of the chapter.

7.1 Frameworks for regulating ontology development

Different authors have proposed to structure ontologies by categorizing concepts or semantic relations according to their “role”, determined by the “journalistic” questions “what”, “how”, “where” ..., or by the thematic relations, notion adapted from linguistics. The authors notably aim at reaching a higher level of precision in the semantic description of concepts and relations: without these categories, the meaning of concepts and semantic relations relies mainly on text-based description.

[Zachman, 1987] proposed to categorize information according to the questions what, how, where, who, when, and why (Table 6). “How” represent processes, “what” a taxonomic or mereonomic hierarchy, “who the roles, “when” the temporal connections between concepts, and “why” some justifications or goal.

[Kingston, 2008] proposed to structure ontologies according to the Zachman framework. The questions provide interesting topics into which the properties could be separated, as different “viewpoints”. This framework should facilitate accurate communication between domain experts and ontology engineers, to improve the quality of ontologies developed. But it does not give any information on what actually guided the development of the ontology, and requires to develop ontologies a certain way, or to rewrite them. Inspired by the Zachman framework, the author proposed a list of relations that could be standardised for ontology development (Table 5).

Relations typically associated with particular perspectives

WHAT	<i>is-a</i> (taxonomy), <i>part-of</i> (mereonomy)
HOW	<i>achieves</i> (goal), <i>transforms</i> , <i>creates/destroys</i> , any term reflecting a specific action (<i>selects</i> , <i>matches</i> , etc.)
WHO	<i>plays-the-role-of</i> , <i>responsible-for</i> , <i>has-rights-to</i>
WHEN	<i>precedes/follows</i> , possible with time intervals specified
WHERE	<i>location</i> , <i>connected-to</i> , or terms reflecting geographical relationships (<i>close-to</i> , <i>south-of</i> , etc.)
WHY	<i>causes</i> , <i>justifies</i>

Table 5: Semantic relations suggested for standardization [Kingston, 2008]

		What	How	Where	Who	When	Why	
Row 1 – Scope External Requirements and Drivers Business Function Modeling	1	Contextual						Contextual
Row 2 – Enterprise Model Business Process Models	2	Conceptual						Conceptual
Row 3 – System Model Logical Models Requirements Definition	3	Logical						Logical
Row 4 – Technology Model Physical Models Solution Definition and Development	4	Physical						Physical
Row 5 – As Built As Built Deployment	5	As Built						As Built
Row 6 – Functioning Enterprise Evaluation	6	Functioning						Functioning
		What	How	Where	Who	When	Why	

Table 6: The Zachman framework, representation⁴⁴ [Zachman, 1987]

[Moldovan et al, 2004] established in Table 7 a list of semantic relations found in noun phrases that “covers a large majority of text semantics”. This list contains many semantic relations that could serve as standard relations between concepts.

[Sowa, 1996]⁴⁵ proposed to adapt the linguistic notion of thematic relation⁴⁶, which characterizes the role that a noun phrase plays in a sentence, to be an intermediate level of his formal ontology. Depending on the context, each entity plays a specific role. John Sowa proposed to arrange these roles in a matrix (Figure), with verb categories as rows and kind of participants as columns. Thematic roles (such as Agent or Location) are arranged within a hierarchy of participants depending on their position in the matrix (Table 8): “Source” (resp. “Product”) is the most general participant, and is specialized into “Initiator” and “Resource” (resp. “Goal” and “Essence”). “Essence” is specialized again in “Patient”, “Theme”, “Location”, etc. The list of possible roles can be extended, to match the domain in a more appropriate way.

Sowa suggested using these relations as an assistant to help determine how to specialize the ontology concepts in a way adapted to the domain and the application. He illustrates this by the choice of how to model an entity “taxi driver”. As the specificity of a taxi driver is that he/she possesses a licence to exercise, the concept should not be placed under “Driver”, because this latter concept represents a person who is actively driving a vehicle (Driver<Doer<Agent<Initiator<...).

This proposal is comparable to the previous one, as it aims at improving the ontology structure, adding some generic semantic constructors. The list of generic relations is larger than the list by [Kingston, 2008], and should be adaptable to any application. Thematic relations are linguistic tools to analyse sentences, placing noun phrases in

⁴⁴ Image from http://en.wikipedia.org/wiki/Zachman_framework

⁴⁵ Synthesized from [Janowicz, 2005]

⁴⁶ See http://en.wikipedia.org/wiki/Thematic_relation

relationship with one another. One could therefore imagine that implementing thematic relations in ontologies could strengthen the semantic relations between concepts; the comparison of thematic relations associated with concepts mapped would probably improve the comparison of these concepts use.

Table 7: Semantic relations valuable for most noun phrases [Moldovan et al, 2004]⁴⁷

Semantic Relation	Definition / Example
POSSESSION	an animate entity possesses (owns) another entity; (<i>family estate; the girl has a new car.</i>); (Vanderwende 1994)
KINSHIP	an animated entity related by blood, marriage, adoption or strong affinity to another animated entity; (<i>Mary's daughter; my sister</i>); (Levi 1979)
PROPERTY/ ATTRIBUTE-HOLDER	characteristic or quality of an entity/event/state; (<i>red rose; The thunderstorm was awful.</i>); (Levi 1979)
AGENT	the <i>doer</i> or instigator of the action denoted by the predicate; (<i>employee protest; parental approval; The king banished the general.</i>); (Baker, Fillmore, and Lowe 1998)
TEMPORAL	time associated with an event; (<i>5-o'clock tea; winter training; the store opens at 9 am</i>), includes DURATION (Navigli and Velardi 2003).
DEPICTION- DEPICTED	an event/action/entity depicting another event/action/entity; (<i>A picture of my niece.</i>).
PART-WHOLE (MERONYMY)	an entity/event/state is part of another entity/event/state (<i>door knob; door of the car</i>), (Levi 1979), (Dolan et al. 1993),
HYPERNYMY (IS-A)	an entity/event/state is a subclass of another; (<i>daisy flower; Virginia state; large company, such as Microsoft</i>) (Levi 1979), (Dolan et al. 1993)
ENTAIL	an event/state is a logical consequence of another; (<i>snoring entails sleeping</i>)
CAUSE	an event/state makes another event/state to take place; (<i>malaria mosquitoes; to die of hunger; The earthquake generated a Tsunami</i>), (Levi 1979)
MAKE/PRODUCE	an animated entity creates or manufactures another entity; (<i>honey bees; nuclear power plant; GM makes cars</i>) (Levi 1979)
INSTRUMENT	an entity used in an event/action as instrument; (<i>pump drainage; the hammer broke the box</i>) (Levi 1979)
LOCATION/SPACE	spatial relation between two entities or between an event and an entity; includes DIRECTION; (<i>field mouse; street show; I left the keys in the car</i>), (Levi 1979), (Dolan et al. 1993)
PURPOSE	a state/action intended to result from a another state/event; (<i>migraine drug; wine glass; rescue mission; He was quiet in order not to disturb her.</i>) (Navigli and Velardi 2003)
SOURCE/FROM	place where an entity comes from; (<i>olive oil; I got it from China</i>) (Levi 1979)
TOPIC	an object is a topic of another object; (<i>weather report; construction plan; article about terrorism</i>); (Rosario and Hearst 2001)
MANNER	a way in which an event is performed or takes place; (<i>hard-working immigrants; enjoy immensely; he died of cancer</i>); (Blaheta and Charniak 2000)
MEANS	the means by which an event is performed or takes place; (<i>bus service; I go to school by bus.</i>) (Quirk et al.1985)
ACCOMPANIMENT	one/more entities accompanying another entity involved in an event; (<i>meeting with friends; She came with us</i>) (Quirk et al.1985)
EXPERIENCER	an animated entity experiencing a state/feeling; (<i>Mary was in a state of panic.</i>); (Sowa 1994)
RECIPIENT	an animated entity for which an event is performed; (<i>The eggs are for you</i>) ; includes BENEFICIARY; (Sowa 1994)
FREQUENCY	number of occurrences of an event; (<i>bi-annual meeting; I take the bus every day</i>); (Sowa 1994)
INFLUENCE	an entity/event that affects other entity/event; (<i>drug-affected families; The war has an impact on the economy.</i>);
ASSOCIATED WITH	an entity/event/state that is in an (undefined) relation with another entity/event/state; (<i>Jazz-associated company</i>);
MEASURE	an entity expressing quantity of another entity/event; (<i>cup of sugar; 70-km distance; centennial rite; The jacket cost \$60.</i>)
SYNONYMY (NAME)	a word/concept that means the same or nearly the same as another word/concept; (<i>Marry is called Minnie</i>); (Sowa 1994)
ANTONYMY	a word/concept that is the opposite of another word/concept; (<i>empty is the opposite of full</i>); (Sowa 1994)
PROBABILITY OF EXISTENCE	the quality/state of being probable; likelihood (<i>There is little chance of rain tonight</i>); (Sowa 1994)
POSSIBILITY	the state/condition of being possible; (<i>I might go to Opera tonight</i>); (Sowa 1994)
CERTAINTY	the state/condition of being certain or without doubt; (<i>He definitely left the house this morning</i>);
THEME	an entity that is changed/involved by the action/event denoted by the predicate; (<i>music lover; John opened the door</i>); (Sowa 1994)
RESULT	the inanimate result of the action/event denoted by the predicate; includes EFFECT and PRODUCT. (<i>combustion gases; I finished the task completely</i>); (Sowa 1994)
STIMULUS	stimulus of the action or event denoted by the predicate (<i>We saw [the painting]. I sensed [the eagerness] in him. I can see [that you are feeling great].</i>) (Baker, Fillmore, and Lowe 1998)
EXTENT	the change of status on a scale (by a percentage or by a value) of some entity; (<i>The price of oil increased [ten percent]. Oil's price increased by [ten percent].</i>); (Blaheta and Charniak 2000)
PREDICATE	expresses the property associated with the subject or the object through the verb; (<i>He feels [sleepy]. They elected him [treasurer].</i>) (Blaheta and Charniak 2000)

Yet, thematic relations are not included in any current ontology language; if they were, one would need to precise it for each relation, as a concept may play as many roles as it has relations. It would be burdensome, would require rewriting existing ontologies. It would also probably require a supplementary training in linguistic, and add complexity to ontologies, which may hide the different perspectives with which they have been developed.

⁴⁷ Image reproduced with the first author's authorization.

	Source		Product	
	Initiator	Resource	Goal	Essence
Action	Actor, Effector	Instrument	Result, Recipient	Patient, Theme
Process	Agent, Origin	Matter	Result, Recipient	Patient, Theme
Transfer	Agent, Origin	Instrument, Medium	Experiencer, Recipient	Theme
Spatial	Origin	Path	Destination	Location
Temporal	Start	Duration	Completion	Point in Time
Ambient	Origin	Instrument, Matter	Result	Theme

Table 8: Thematic relations suggested to structure ontologies⁴⁸ [Sowa, 1996]

7.2 Frameworks for regulating ontology reconciliation

Some authors realize the problems posed when reconciling ontologies by mappings, and propose to make use of contextual information to relate ontologies more efficiently.

[Paslaru-Bontas, 2007] proposed a metadata model⁴⁹ to describe the characteristics of ontology development judged relevant for the guidance of the ontology reuse process; she developed the ontology in accordance with well-tried ontology engineering methods and empirical findings. She formalized the metadata model in the Web ontology language OWL DL to facilitate the integration and exchange of contextual data in Semantic Web applications.

The metadata information is categorized, following [Guenther and Radebaugh, 2004], into

- *structural* metadata – statistical information about the ontology structure;
- *descriptive* metadata – information about the content modelled, such as domain and topic classifications; and
- *administrative* metadata – information about ontologies as artefacts such as authorship, ownership, right management, etc.

⁴⁸ From [Janowicz, 2005], with permission.

⁴⁹ The metadata ontology implementation is available online at <http://swpatho.ag-nbi.de/context/meta.owl>. The first three levels of the is-a hierarchy are presented, and the more important classes commented, in [Paslaru-Bontas, 2007], pages 104-128.

The metadata judged relevant for ontology reuse is again classified into

- *syntactic* features – characteristics of the representation as medium for the conceptualisation, such as the number of concepts and the representation language;
- *semantic* features – characteristics of the semantic expressed in the ontology, such as level of correctness, domain modelled, and point of view; and
- *pragmatic* features – information related to the context of development and use, such as history, successive development purposes and input information sources.

Metadata features are qualified as *required*, *optional* or *extensional* (not presented in detail but that may be the object of extension modules).

Figure 11 shows a partial view of the metadata ontology, including the first two levels of the *is-a* hierarchy, and the third level for a few chosen classes. The Open Directory⁵⁰ topics and a classification of software application on Wikipedia⁵¹ are some of the resources imported in the ontology, as pre-defined vocabularies aimed at reducing the number of simple string properties.

This research work is good news for ontology interoperability, as it provides some contextual information that should help to search and evaluate ontologies. There is still some improvement possible, as the ontology is lacking semantic, hard to instantiate, and heavy. The latter point is due to the fact that OWL DL 1.0 requires concepts referred to in the ontology to be imported (e.g. more than 3000 concepts imported from the Open Directory classification of topics!).

The real drawback for the interoperability need presented in this study is that the granularity of the metadata is the ontology. Multiple Topics can be indicated, but there is no relationship between the topics and the related concepts. On the pragmatic point of view, very few ontologies include this metadata; having to instantiate the metadata ontologies for existing ontologies should not be as burdensome as other approaches presented in this chapter, and would not need to modify the ontologies; but it requires to do some research to discover the context of the ontologies.

[Ehrig et al, 2004] propose a framework to regulate the combination of diverse similarity measures in order to achieve a better comparison of concepts within an ontology or in different ontologies. They categorize the measures into three successive layers:

1. data layer – comparison of data types and of string values (edit distance, etc.),
2. ontology layer – comparison of concepts using the graph structure of the ontology (include intra-ontology methods such as edge-counting methods, but also ontology matching methods such as extensional ones)
3. context layer – comparison of instances in their particular context of use

⁵⁰ Open classification of topics, see <http://www.dmoz.org/>

⁵¹ See http://en.wikipedia.org/wiki/Software_application

The only interest of this framework is that it opens the way towards standard interfaces of similarity measures for various categories of input. This should simplify the evaluation of such measures and therefore act for their general improvement.



Figure 11: partial view of the metadata taxonomy [Paslaru-Bontas, 2007]

7.3 Conclusion

Different authors have proposed to add some information to ontologies, either to restrict the meaning of entities by linguistic/journalistic tools, or to facilitate their comparison with one another by considering their context. Table 9 synthesizes the various approaches proposed.

The first approaches require that ontology development should be regulated by the categorisations proposed. They show directions that the ontology development should take to allow for more reliable ontology reconciliation, but are of no help for ontology reconciliation between ontologies already developed.

The meta-model proposed by [Paslaru-Bontas, 2007] can be possibly added after the ontologies have been developed. But it is intended only for the right selection of ontologies to reconcile.

What is lacking is an understanding of how the context of ontologies could help to relate them with one another. We will therefore examine in the following chapter the literature review about the notion of context in computer science; later on, we will see how the notion of context applies to the measure of similarity between concepts, to finally propose a methodology for context-based improvement of an existing ontology reconciliation.

Table 9: Synthesis of the different proposals

Author	Proposal	Structural elements
[Kingston, 2008]	Categorize properties	What, How, Who, When, Where, Why (cf. Table 5)
[Moldovan et al, 2004]	(the semantic relations found could be used to categorize properties)	Most frequent semantic relations between noun phrases (Table 7)
[Sowa, 1996]	Categorize concepts	Thematic relations (Table 8)
[Paslaru-Bontas, 2007]	Determine the main characteristics of the context of ontology development for ontology reuse	Ontology meta-data (cf. Figure 11)
[Ehrig et al, 2004]	Categorize similarity measures	Data, ontology, and context layer

Chapter 8: Notion of Context in Computer Science

In this chapter, we will introduce the notion of context. We will propose a classification of context-based approaches that reflects the nuances in the understanding and interpretation of the notion. We will then present a panorama of those approaches according to this classification.

The word “context” has appeared in the English language in the 16th century, to mean “*the part of a discourse that surround a word or passage and can throw light on its meaning*”. Although this is still the main meaning today, the term has been progressively used in other circumstances with the more general meaning of “*the interrelated conditions in which something exists or occurs*”⁵².

The notion of context exists in various domains of Computer Science, such as artificial intelligence (AI), software development, databases, data integration, machine learning, knowledge representation, information retrieval (IR). The recognition of the notion of context as a major principle in Computer Science goes back to 1993 when McCarthy stressed the importance of the notion for AI, claiming that context should be a first class object in logic. I propose here a classification of the main meanings that the word “context” has in Computer Science:

1. That which relates the entity of interest to its surrounding environment.
 - Linguistic context** – The words surrounding the word (or group of word) of interest, that can help disambiguating its meaning;
 - Situational context** – Any information that can be used to characterize the situation of an entity, where an entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and the application themselves; [Dey, 2001]
 - Connectional context** – Any information that participates in characterizing an entity of interest involved in the event that triggers the need for context, including its interactions with other entities, where the distinctive features that compose the characterization are judged according to a given purpose.
Particular case: Context – Connectional context, with the purpose of explaining the appearance or some particular characteristic(s) of the entity of interest
2. A formal representation of a perception of reality, either
 - Background information** – A representation of background information considered inherent to a particular situation;
 - Viewpoint** – An autonomous, local model of a reality expressing the point of view of a community or an individual.

We will us now see a panorama of the notion of context in computer science, according to the classification. The works presented are not evaluated, as evaluation criteria

⁵² Definitions from the Merriam-Webster’s Online Dictionary <http://www.m-w.com/dictionary/context>

depend on the methodology to determine what is contextual, which we will see in the following chapter.

8.1 That which relates an entity with its surrounding environment

Most context-based approaches consider the context as what relates the entity into focus to its environment.

☞ **Linguistic context: the words surrounding a group of words**

Information seeking and retrieval is the main domain in Computer Science to consider primarily context as the immediate words that surround a word of interest. Context is often a means to disambiguate keyword-based search. For example, [Finkelstein et al, 2001] propose to disambiguate keyword-based search by generating an augmented query from the paragraph which contains the word selected by the user; IntelliZap returns first results more relevant than the traditional keyword-based search engine on which it is built, when used with 1-3 words.

[Budzick and Hammond, 2000] consider a wider context to disambiguate keyword-based search; they assume that the information need probably occurred as the user was working on some artefact; they use for context the textual content of artefacts such as Word documents opened at the time of the request.

☞ **Situational context: the description of a situation**

In the paradigm of ubiquitous (or pervasive) computing, “context awareness” is concerned with bringing portable electronic devices to be “aware” of the environment in which they are used. Context-aware systems rely on various sensors (GPS, clock, etc.) to provide a human-computer interaction adapted to the circumstances; context is understood as “environmental”, currently essentially “situational” information. [Dey, 2001] defines context as *“any information that can be used to characterize the situation of an entity, where an entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and the application themselves.”*

Context-aware systems⁵³ usually manage location, time, subscription to some source of events such as media, and relates them to a user profile to detect particular settings relevant to the user; when these conditions occur, the system may alter its functioning or alert the user so as to provide her/him the opportunity to engage a corresponding action. Most context-aware systems are sensitive only to spatial location. For example, a context-aware handheld device might display restaurants fitting to the user’s preferences, privileging the closest ones. Some systems are sensitive to other dimensions, and include sensors for noise, light, time, and include in the user profile information such as the user’s position in an organization. A system sensitive to all of these contexts could, for example, allow a user to ask for a message to be delivered to the first doctor that enters room number 108 after 8am [Baldauf et al, 2007].

⁵³ For further information on history of origin of context-awareness, see [Chalmers, 2004]. For a more general review including the design principles, and existent systems and frameworks, consult [Baldauf et al, 2007].

Research work about context-awareness is mainly concerned about sensors and how to use information retrieved from sensors to personalize the information. There is also the concern for adaptation of information format and content for portable devices to offer, for example, adapted internet navigation and interaction. Comparing ontologies should not be dependent in any way on the location of the user, the time of access, or any sensor related to the user. The information described in context models for context-awareness is not relevant for the comparison of concepts across ontologies.

☞ Connectional Context: that which characterize the involvement of an entity in an event, judged according to a given purpose

Many research efforts in the domain of IR consider context, as giving more understanding of the search process may help users to reformulate their queries, and obtain more appropriate results. SearchPad [Bharat, 2000] keeps tracks of queries made, during one or more sessions of Web search, along with the respective relevant results. The association of search queries and results should help the users have a better understanding of how they are related. With this handy information, users can pursue the search process when they need it, with queries construed more knowledgeably, and in average more pertinent results.

[Lawrence, 2000] presents a short review of the use of context in Web search; he recognizes different categories of origin of contextual information: in some cases, the user is prompted to provide the information need that triggered the search event; in others, the system searches automatically for contextual information in the documents that the user is editing, in his files, bookmarks, etc. . The information considered as contextual may be personal data about the users of the search engine (assuming that their search concerns their own information need), to personalize the results. It may be statistical information based of archived queries from a wide range of users, to estimate the goal of the search based on associations of keywords (this is contextual information about the user, based on the assumption that the user is affected by the surrounding culture); it may be the knowledge of the search engine utilized, or the users' IP address (that presumably gives a current location, which is supposedly the place where the information found will serve).

For problem solving, [Brézillon, 2007] represents the organized conditions to realize a task in a “contextual” graph, which is a finite state machine model. This model is aimed at providing a track record of the task realisation history. Armed with this information, one could know, when some particular circumstance occurs, what triggered it, and the appropriate action to take.

[Firat et al, 2007] model contexts as a list of units (such as currency), formats (for example date format), and background assumptions (such as services, taxes included or not in the price). Context models are instantiated to describe the characteristics of the information recorded in each data source represented, as well as the user preferences. The list of possible context modifiers is fixed, with the associated list of dimensions. Conversion functions transform the data at need from a context to another.

[Jouanot et al, 2003] propose an approach to facilitate the cooperation of Information Systems, based on mediation of schemas and contexts. They propose to describe

information stored in any data source by information objects (I-Objects). A context is modelled to precise the validity of these I-Objects, by a set of concepts. Contexts from the different information systems are compared using a semantic similarity measure between their concepts. The result of the comparison serves to discover which I-Objects classes correspond to one another. When a consumer using an information system encounters an information need that requires the information systems to cooperate, the data associated with the I-Objects is rendered in virtual I-Objects, using interpretation rules to avoid ambiguities of the data meaning.

[Paslaru-Bontas, 2007] provides a context-sensitive methodology to discriminate ontologies that should be reused for a particular ontology development. She proposes to evaluate the candidate ontologies for reuse by criteria such as the estimated relevance for the application domain, the quality of the modelling, the technical context, and the tasks for which the ontology was built (compared to the task for which the new ontology will be used). The need of the ontology has occurred in a particular context. The methodology relies on this context to refine and optimize the integration strategy accordingly. The context of ontology reuse is described by the ontology task and role, and the reuse level (vocabulary, vocabulary and semantics, or instance data).

8.2 A formal representation of conditional statements

Some authors name “context” representations or views that are designed to represent the same information in different ways, with distinct relations and rules between entities represented, etc.

☞ Background information: representation of background information considered inherent to a particular situation

Human beings, when they interact, take usually for granted some background assumptions (sometimes named implicit knowledge) considered to be generally true in a default setting. It seems that one main purpose of this behaviour is to reduce the amount of information to be transmitted. It could be compared to the classification of network layers, where within each layer, systems communicate with a restricted vocabulary, and solve only a restricted set of problems; if someone considers the highest layer in isolation, without knowing that there are other layers, he/she would certainly make wrong interpretations of what is actually happening.

This background information is capital to evaluate every new information bit correctly. When this knowledge is asserted qualitatively, the evaluation is to be done in the reference system determined by the background information. For example, one could ask “is the cheetah fast?”, and people would understand that the reference system is the animal kingdom ... so, the answer could be “yes, it is a fast runner” ... but if the reference system comprised also vehicles made by human beings, the answer would rather be “less than my motorcycle!”

John Mc-Carthy proposed in 1993 to introduce the notion of context in logic as a first class object [McCarthy, 1993]. His aim was to provide means to formalize background knowledge within different reference systems called “contexts”; a same assertion can be true in a context, and false in another. His vision was that rules and reasoning capabilities would allow to “transcend” contexts, that is, to transfer some information

from one context to another. The lack of such rules is still the Achilles' heel of these representations in formal logic.

His PhD student at the time, Ramanathan V. Guha, [Guha, 1992], used his research experience to partly formalized background as “micro theories” in Cyc⁵⁴. This was done to help reduce the contradictions of common sense sentences from various origins, and to limit the memory required to store the millions of assertions (1.5 millions in 1991). According to [Mascardi et al, 2007], Cyc knowledge base contains currently thousands of “micro theories”, more than 300 000 terms and nearly 3 000 000 assertions hand-entered or inferred by the system. A micro theory is associated with a topic, like “TV repair”, “what to look for when buying a cellular phone”, etc. It contains simplified assertions about the world that are not generally true, but true for that topic. The interest of such a huge base of “common sense” has not yet been demonstrated. [Friedland et al, 2004] describe the project HALO⁵⁵, where three companies were attempting to build a knowledge system that should encode knowledge from a chemistry textbook in order to answer questions on a freshman chemistry exam. Despite its large knowledge base, Cyc had the lowest score. With the micro-theories implemented in Cyc, the input got even more complex, done by hand at the cost of about \$10000 the page of textbook, as every information has to be re-entered in Cyc specific language. The open source project OpenCyc⁵⁶ is being developed as IEEE standard, and contains already hundreds of thousands of terms, and millions of assertions between them. Despite its large base, Cyc is not much likely to include the domain and the contexts one is interested in.

☞ **Viewpoint: autonomous, local model of a perception of reality**

Giunchiglia proposed to represent “context” in logic⁵⁷ at the same conference than [McCarthy, 1993], in Annecy, France, but more as a constructor to allow reasoning between various points of view. [Attardi and Simi, 1995] proposed a formalization of the notion of viewpoint in first order logic. [Bouquet et al, 2003] proposed C-OWL, an extension of OWL-DL, (which is based on a subset of the first-order logic), where C stands for context. They make a distinction between “ontologies”, for them the result of a consensus and a shared model that different organizations agree on to enable collaboration, and “contexts”, which they view as local models developed in an autonomous way and expressing the point of view of a community or an individual. They consider that a semantic resource is a “context” when its autonomy must be preserved, and when the correspondence with other resources has not to be perfect: for example, for access of distributed knowledge, document classification, catalogue integration, etc.

Approaches based on logic are mostly interested in guaranteeing the consistency of axioms among different co-existent local models of logic. After more than fifteen years, the central issue when modelling contexts is still “*that of lifting*” (passing from a

⁵⁴ Cycorp, inc. See <http://www.cyc.com> or the open source version at <http://www.cyc.com/cyc/opencyc>

⁵⁵ Long-term research and development initiative that aimed at developing an application capable of answering novel questions and solving advanced problems in a broad range of scientific disciplines, see <http://www.projecthalo.com/>

⁵⁶ See <http://www.opencyc.org/>

⁵⁷ A review of the representation of context in logic systems can be found in [de Paiva, 2003].

context to another) [Guha and McCarthy, 2003], which is exactly the main interest in context modelling! The state-of-the-art approach gives the possibility to put logical assumptions together, and renounce to some of it if there is any contradiction. But contradictions are guaranteed. Even in the case when contradictions would not appear, the meaning behind the logic probably differs, and thus the conclusions reached will be probably wrong.

The software engineering domain has also a notion of “viewpoints” that has recently been adapted to ontologies. They are generally a unique representation where ontology entities are associated with one or more viewpoints. [Rivière and Dieng-Kuntz, 2002] proposed to represent viewpoints in conceptual graphs. The ontology is partitioned into various viewpoints, which each are a "coherent and partial view of the knowledge base". Experts are associated with a few viewpoints, which can be shared among multiple experts. The viewpoints are inserted in the ontology, under the form of viewpoint-specific subsumption relations.

[Benslimane et al, 2003] propose to represent a few distinct points of view on a same subject in a single ontology. They define a “MUltiRepresentation Ontology » (MuRO), aiming at solving problems specific to viewpoint management, as data filtering, etc. The passage from one representation to another is realized through the knowledge of common concepts.[Benslimane et al, 2006] propose an extension of the OWL-DL language with modal logic, so as to represent many viewpoints in the same ontology; definitions are no longer general, but associated with a given viewpoint. The idea of representing different points of view in an ontology is interesting, as one could possibly use mappings only in the appropriate points of view. In the case where viewpoints are chosen to match the different purposes that the ontology has been developed for, restricting the use of concepts mapped to some selected viewpoints would probably improve the reliability of ontology interoperability. But, the development of multi-viewpoint ontology is demanding, and rare: most ontologies are developed with a single point of view, and this will not change in the near future. Besides, multi-viewpoint ontologies would not solve everything: viewpoints will differ from one ontology to another, so there would still remain a need to connect the points of view together.

[Stuckenschmidt, 2006] provides a method to extract a “viewpoint” from an existing OWL-DL ontology to reuse it for a specific application. Unnecessary differentiations are removed, to keep only the definitions that fit the requirements for the application. Approximate subsumption reasoning permits some “viewpoint”-based reasoning, while preserving the original ontology. The approach is currently limited to a single ontology, not adapted in the case the viewpoint is more than a simple subset of the first ontology, and not addressing the connection of multiple ontologies. This approach constrains experts to select exhaustive concepts from all ontologies. Yet, the approach objective to get a fully workable sub-ontology becomes a much more complex task when building the viewpoint from different ontologies, making it – in our opinion – close to the burden of ontology building.

[Poslad and Zuo, 2008] propose a framework to allow database users to access data in a way determined by their small ontology, which acts as a view. The ontology is related by mappings to the global ontology, and permits to access all the sources as if the global

ontology was used directly. This approach requires to develop a global ontology, which may be huge, to provide an global access to the data. This is a lot of work, which is hardly compatible with the flexibility needed for project-oriented interactions between organizations, as presented in the introduction.

8.3 Conclusion

The notion of context is perceived easily, but is hard to define. Indeed, the characteristics that compose the context for one approach will usually not be relevant for the others. Literature devoted to context abounds, but is hardly relevant unless related to the field, the object studied and the purpose pursued. Table 10 synthesises the approaches presented to help to see the variations in the use of context.

We will now review the literature about semantic similarity and relatedness, and how context has been used to modify measures of similarity.

Table 10: Synthesis of the use of context in Computer Science

Context as what relates the entity of interest to its surrounding environment

Domain	Authors	focus	Notion of context	composition of the context	Purpose
Information Retrieval	[Finkelstein et al, 2001]	Word queried	Linguistic	Surrounding text	Disambiguate among possible meanings of the term
Information Retrieval	[Budzick and Hammond, 2000]	User's task	Linguistic	Text in documents opened by the user	Disambiguate among possible meanings of the term
Information Retrieval	[Bharat, 2000]	Discovery of the document	Connectional	Queries done concurrently	Uncover the way the document was obtained
Ubiquitous computing	[Dey, 2001], [Baldauf et al, 2007], etc.	Interaction of the user with a device	Situational	Location, noise, light, time, and role	Adaptation of service to the situation (external)
Problem solving	[Brézillon, 2007]	Problem that occurs	Connectional	Different stages that lead to the problem, and the stages to solve it.	Discover the circumstances that were the origin of the problem
Ontology integration	[Paslaru-Bontas, 2007]	Ontology artifact development	Connectional	Methods used, topics, tasks the ontology is designed for, information sources, etc.	Evaluate the interest of the ontology for reuse

Context as a formal representation

Domain	Authors and tools	Representation type	Purpose
logic	[McCarthy, 1993], [Guha, 1992], [Mascardi et al, 2007], Cyc, OpenCyc	Background information	Managing inconsistency of general knowledge
logic	[Bouquet et al, 2003]	Viewpoint	Managing inconsistency of general knowledge
Description logic	[Benslimane et al, 2006], [Barth, 2006]	Viewpoint	Representing multiple roles
Description logic	[Stuckenschmidt, 2006]	Viewpoint	Reducing a part of an ontology
Description logic	[Poslad and Zuo, 2008]	Viewpoint	Personalizing the request interface

Chapter 9: Measure of Similarity and Relatedness between Concepts

Similarity measures have been proposed in the literature to compare various things, from geometric figures to ontology concepts. As human beings we are used to compare objects and concepts and evaluate their similarity. It is a human perception, and therefore the studies of similarity will appeal to psychology, social studies, and experiments to gauge heuristic measures of similarity with values of similarity asserted by the average people. Similarity is a simplification of a reality for purposes such as decision aiding: one numeric value⁵⁸ summarizes the information gathered.

Similarity between ontology concepts is usually named “semantic similarity”. According to [Resnik, 1999], semantic similarity is an evaluation of the connection between two concepts with the objective to get an estimate of the degree of proximity between the signification of the two concepts.

9.1 Characteristics of similarity

[Tversky, 1977] argued that the human-based notion of similarity does not fit with the mathematical notion of a metric. He gave examples where the human perception of similarity is asymmetric and where the triangular inequality is not respected. People commonly compare concepts using metaphors “A is like B”, which cannot usually be transformed into “B is like A” without change of meaning (think of “lawyers are like sharks”). Tversky demonstrates that the triangular inequality is not always respected, giving as example that “*although Jamaica is very similar to Cuba and Cuba is very similar to Russia, Jamaica is not at all similar to Russia.*”

[Rada et al, 1989] refute the considerations given. They remark that the concepts play different roles in the metaphor, the saying attributing some well-known properties of the concept B to the concept A. About the triangular inequality, they argue that Jamaica was compared to Cuba according to geographical characteristics and to Russia according to political ones; therefore, a similarity measure would consider at least both characteristics, and bear results compatible with the triangular inequality property.

The two parties still have supporters. Psychologists consider that similarity measures should predict what human beings will perceive as being similar. Computer scientists, on the other side, tend to prefer a measure that has good mathematical properties, is easily computable from data widely available, etc. [Lin, 1998]. But the tension remains. For example, [Rodríguez and Egenhofer, 1999] propose a measure of similarity based on [Tversky, 1977]’s proposal, designing the measure, notably, to be asymmetric and not to respect the triangular inequality.

Semantic similarity measures have been proposed to compare concepts in a taxonomy since the 1980s (“intra-ontology” similarity measures). More recently similarity

⁵⁸ Sometimes, one can have two values of similarity and confidence, but often they are merged into one.

measures have been designed to provide an evaluation of the similarity of concepts defined in different ontologies (“inter-ontology” similarity measures).

9.2 Similarity measures intra-ontology

Intra-ontologies similarity measures aim at comparing concepts to associate to them a value of similarity that is close to the similarity that people would find between them. Approaches to compute semantic similarity among concepts in an ontology have been categorized into *features* based, *edge counting* based, *information content* based, and *hybrid* measures.

Feature-based measures originate with [Tversky, 1977], who studied human perception of similarity from a psychologist point of view. He consequently proposed to measure objects similarity by comparing their common and distinctive “features”, either by difference – the “contrast” model, or by division – the “ratio” model. The intuition is that the more features the objects share and the less distinct features they have, the more they are similar. More recently, [Rodríguez et al, 1999] proposed a similarity measure based on this approach, adapted for the comparison of concepts (actually, “entity classes”⁵⁹) in an ontology, interpreting features as concept attributes.

Feature-based measures suffer from many limitations: as features are unary properties, the comparison lies on the presence or absence of the feature, it is not possible to state and compare colour=”green” with colour=”red”. One cannot have a feature “age”, but it has to be replaced with categories such as child, teenager, adult; which poses problem when categories are not independent, as when one wants to distinguish the case “senior”, as a senior is an adult. Also, a concept cannot have a same feature twice. [Janowicz, 2005] gives the example of a theatre height, which one may want to associate to the building as well as to the stage, but it is not possible. Finally, there is no way to state the feature order of importance in the computation of similarity.

Edge-counting measures were introduced by [Rada et al, 1989] who wanted to adapt Tversky’s similarity measure to the comparison of ontology concepts. Their hypothesis was that the hyponymy links “*is-a*” indicate that two concepts share some defining features. They suggest that these relations are sufficient to compute a semantic similarity measure. They found that the minimum number of *is-a* links that separate two concepts was a good estimate of their semantic relatedness. The measure proposed has the properties of a distance, such as symmetry and triangular inequality (Equation 1, see Table 11 for the definitions of the functions used). [Wu and Palmer, 1994] proposed a slightly modified formula which considers the level of the concepts in the hierarchy (Equation 2). [Leacock and Chodorow, 1998] divided the measure from [Rada et al, 1989] by twice the ontology depth, and took the negative logarithm of the result, to transform it into a function that behave more like what is expected from a similarity measure (Equation 3).

[Bidault, 2002] proposed to modify the representation of concepts in a taxonomy, with the objective to provide on-demand similarity measure for arbitrary concepts. Concepts

⁵⁹ Entity classes are classes designed to get easier access to chunks of data with the same characteristics (entities) from a database. See <http://apps.carleton.edu/opensource/reason/developers/docs/entity/>

are uniquely characterized by the paths (succession of relations of subsumption) that connect them to the Top concept of the ontology. Each such path is implemented as a concatenation of symbols, so that each new *is-a* relation from a subsuming concept is associated with a different symbol. As a concept may be related to the Top by different paths, there are also many different identifiers for a concept. The least common super concept of concepts to compare is obtained by string comparison between their identifiers rather than by search in a graph. This representation is adapted for the computation of most semantic similarity measures, as it includes information about all the *is-a* links. The author proposed a similarity measure that is approximately the average, when comparing all possible paths to the Top, of the number of *is-a* links from the least common super concept to the Top plus the difference of the depth of the second concept with the ontology depth, divided by the ontology depth (Equation 6).

Information-content-based measures depend on tagged corpora to derive information content values for concepts. The assumption in these approaches is that similarity values can take advantage of a corpus to discriminate depending on the concept specificity, along with the taxonomic arrangement. Information content is a notion that comes from the field of information theory, and its assumption is that concepts that are more specific (less present in the corpus) are more informative than others. [Resnik, 1995] is the first to adapt it to the measure of concept similarity (Equation 4), followed notably by [Jiang and Conrath, 1998], [Lin, 1998] (Equation 5), [Lord et al, 2003], and [Seco et al, 2004].

The weakness of measures based on information content, is that they require a corpus. Sometimes a linguistic resource such as WordNet is used, but this is limited as ontologies usually include specialized vocabulary, which is not defined in general resources.

Table 11: Common functions for the measures of similarity

c_1, c_2	concepts
$len(c_1, c_2)$	minimum of <i>is-a</i> edges separating the concepts
$depth(c)$	(minimum) number of <i>is-a</i> edges from the concept c to the ontology root
$D = \max_{c \in O} \{depth(c)\}$	Ontology depth
$S(c_1, c_2)$	Set of concepts that subsume both c_1 and c_2
$p(c)$	Probability of encountering an instance of concept c in the corpus associated with the taxonomy
$lcs(c_1, c_2)$	Least common super concept of c_1 and c_2
$nbPaths(c)$	Number of paths from the concept c to the ontology root
$depth_{path}(c)$	Depth of a concept according to the path (to the Top)

$$sim_{Rada}(c_1, c_2) = len(c_1, c_2) \quad (1)$$

$$sim_{WP}(c_1, c_2) = \frac{2len(lcs, Thing)}{len(c_1, lcs) + len(c_2, lcs) + 2len(lcs, Thing)} \quad (2)$$

$$sim_{LC}(c_1, c_2) = -\log \frac{len(c_1, c_2)}{2D} \quad (3)$$

$$sim_{Re_snik}(c_1, c_2) = \max_{c \in S(c_1, c_2)} (-\log(p(c))) \quad (4)$$

$$sim_{Lin}(c_1, c_2) = \frac{2\log(p(lcs))}{\log(p(c_1)) + \log(p(c_2))} \quad (5)$$

$$sim_{Bidault}(c_1, c_2) \approx \frac{1}{nbPaths(c_1) \cdot nbPaths(c_2) \cdot D} \sum_{paths} D + depth_{path}(lcs_{path}) - depth_{path}(c_2) \quad (6)$$

$$S_{rel}(c_1, c_2) = \frac{|\{synset(y) \cap synset(y'), rel(c_1, y), rel(c_2, y')\}|}{|\{synset(y) \cup synset(y'), rel(c_1, y), rel(c_2, y')\}|} \quad (7)$$

$$sim_{Petrakis}(c_1, c_2) = \max\{S_{part-of_neighbor}(c_1, c_2), S_{is-a_neighbor}(c_1, c_2), S_{term_description}(c_1, c_2)\} \quad (8)$$

$$sim_{HSIO}(c_1, c_2) = K - \min_{path} len(path(c_1, c_2)) - D \cdot \min_{path} nb(directionChange(path)) \quad (9)$$

Table 12: Formulas of the similarity measures cited

Hybrid methods

[Rodríguez and Egenhofer, 2004] adapt Tversky's approach to compare ontology instances, and consider the taxonomy links to get structural information adding some "context" to the comparison. We will examine this method in the next chapter.

[Bernstein et al, 2005] reviewed many semantic similarity measures, and conducted an experiment with 50 people, using instances from the MIT Process Handbook ontology. They suggested that the correlation between human judgment and similarity measures was more dependent on the quality of the ontology examined than on the choice of a particular measure.

9.3 Semantic relatedness intra-ontology

Similarity measures compare concepts to find those that are equivalent or have some similar characteristics. Some metrics are concerned with relatedness, and will try to find concepts that have a semantically related with one another, for example a relation of antonymy, or a (few) custom semantic relation(s). [Rada et al, 1989] were the first who tried to take semantic relations into account for their similarity measure, but according

to their experiments, the results were more comparable to human judgment of similarity when used with only *is-a* links.

[Hirst and StOnge, 1998] propose a measure to compare concepts that do not consider sole *is-a* relations. The authors' intuition is that the relatedness of concepts is higher when the shortest path that connects them is made of fewer relations. The strength of the relation (in most cases) is given by Equation 9, where C and K are constants.

9.4 Similarity measures across ontologies

In the same way as ontology matching methods, inter-ontology similarity measures rely on methods described in chapter II, but also on intra-ontology similarity measures. The main difference between the following measures and the ontology matching methods is that the latter methods focus in delivering a limited set of mappings between the concepts judged most similar, whereas semantic similarity measures can be applied to any couple of concepts, on demand

[Rodríguez and Egenhofer, 2003] propose a model for semantic similarity across different non-axiomatized ontologies. Their objective is that such a similarity measure should help to establish strong mappings across different ontologies⁶⁰. They compare classes (“entity class”) based on terminology (they suppose that a set of synonyms is associated with each class), structure (“schematic neighbourhood”), and features (they separate class properties into “parts”, “functions” and “attributes” to treat them separately, with the risk of mismatch when concept properties must be separated into these three categories, first; they compare features across ontologies with pure string matching). The model is based on [Rodríguez and Egenhofer, 1999], presented in the next chapter.

[Petrakis et al, 2006] propose a simplified version of this feature-based measure model. It is. As features, they consider the synsets and term descriptions of each class. Two terms with one common synonym term are 100% similar. Other terms obey the rule Equation 8. On an experiment to match terms between WordNet and the medical taxonomy MeSH, their method was more correlated to the human judgment than the previous one (Equations 7, 8).

Because the knowledge present in these sources has supposedly already served to compute the mappings, we are not interested in these measures.

9.5 Conclusion

The notion of semantic similarity is looked at with points of view that seem contradictory: psychologists view it as a measure that should manifest the properties that they observe in the population; scientists insist that the similarity should be based on a distance, with the corresponding mathematical properties.

The measure of semantic similarity is mainly evaluated within a taxonomy or an ontology, based on the *is-a* hierarchy that forms a classification of the concepts. Some

⁶⁰. [De Souza and Davis, 2004] use this similarity measure for their method for ontology alignment based on formal concept analysis (FCA), with qualified success.

measures of semantic similarity rely also on statistical information about a corpus related to the ontology. One speaks of semantic relatedness when other types of semantic relations than *is-a* are considered to relate the two concepts by a path.

When human beings evaluate two or more possible solutions to a problem, they usually consider the context to choose which criteria are more important. In the same way, concepts have various characteristics (such as attributes, semantic relations), that distinguish them one to another. There is usually no unique way of comparing concepts. The context is needed to determine what makes concepts more similar. It is needed to decide, among the various criteria for comparing concepts, which ones are the more important. Semantic relatedness measures are even more context-dependent than semantic similarity measures. Concepts can be related in numerous ways. Context can limit the types of relation that are appropriate.

Chapter 10: Context-based Similarity Measures

In this chapter, we will uncover the influence that the context has on similarity, to find out that the notion of similarity is context-dependent by nature. We will then review the literature, and present context-based semantic measures of similarity. As there are very few of them, we present not only context-based measures between concepts, but also context-based measures between concept instances, with the thought that, although unlike, such approaches might suggest interesting directions.

The human perception of similarity is largely influenced by the context [Goldstone et al, 1997]. The following figure is a good example of that principle. The two ellipses will be judged to be slightly dissimilar, having a different colour, whereas they have exactly the same colour. The other objects that surround them make us give another interpretation. [Tversky, 1977] give an illustration of the “*extension effect*”, which is a particular case of the influence of external conditions: a group had to evaluate the similarity of two countries in South America among a set of countries from the American continent; the average value given increased when a European country was added to the set of countries.

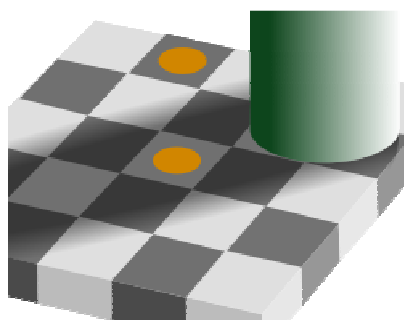


Figure 12: Effect of the context: the two ellipses seem to have a different colour⁶¹

[Goldstone et al, 1997] made experiments consisting in selecting a figure from a set of figures, to match with a standard figure (by similarity or dissimilarity). They showed that the variation of a non relevant figure in the set had an effect on the choice the participants made. This demonstrates that, according to [Tversky, 1977]’s contrast model, the weights that represent the salience of the features used to realize the comparison are not fixed until the very moment of the comparison: the selection is context dependent.

All these experiments plead for context-dependent similarity measures, either for dynamic ontology matching, or to adapt static ontology mappings to a particular context. We will therefore review the literature on proposals of context-dependent semantic similarity measures; as they are few, I include also research work on context-

⁶¹ Image from <http://en.wikipedia.org/wiki/Colour>

dependent similarity of ontology instances, with the objective of giving a broader perspective on what has been considered as context about ontologies.

10.1 Context-based similarity measures between concepts

We will now review principally the similarity measure of [Rodríguez & Egenhofer, 2004], that compares concepts defined in the same ontology, making use of contextual information constituted by the difference of depth of the concepts and the user's intended task.

[Rodríguez & Egenhofer, 1999], followed by [Rodríguez & Egenhofer, 2004], proposed a new model of semantic similarity which takes the user's task into account. Their "matching distance similarity measure" (MDSM) is mainly a feature-based similarity measure. The authors divide concept properties into "parts", "functions" and "attributes", and construct three differentiated measures of similarity after Tversky's ratio model (Equation 10, see the definition of the named functions in Table 13).

Table 13: Functions used by the measures of similarity

$C_f(c), C_p(c), C_a(c)$	Set of functions, parts and attributes of the concept c
$lcs(c_1, c_2)$	Least common super concept of c_1 and c_2
$\alpha(c_1, c_2) = \min_{c \in \{c_1, c_2\}} \frac{len(c, lcs(c_1, c_2))}{len(c_1, c_2)}$ with $0 \leq \alpha(c_1, c_2) \leq 1$	Normalized weight to account for the difference of depth between c_1 and c_2
$V_{x,AD} = 1 - \frac{1}{ \{c \in O\} \cdot \{f \in AD\} } \sum_{f \in AD} \{c \in O, f \in C_x(c)\} $	Variability for a given feature f of type x in an application domain AD , related to the whole ontology O
$\omega_{x,AD} = \frac{V_{x,AD}}{\sum_{x \in \{f,p,a\}} V_{x,AD}}$ with $\omega_{f,AD} + \omega_{p,AD} + \omega_{a,AD} = 1$	Normalized weights for the similarity measures for functions, parts and attributes, dependent of the application domain

$$S_x(c_1, c_2) = \frac{|C_x(c_1) \cap C_x(c_2)|}{|C_x(c_1) \cap C_x(c_2)| + \alpha(c_1, c_2) |C_x(c_1) - C_x(c_2)| + (1 - \alpha(c_1, c_2)) |C_x(c_2) - C_x(c_1)|} \quad (10)$$

$$Sim_{RE}(c_1, c_2) = \omega_{f,AD} \cdot S_f(c_1, c_2) + \omega_{p,AD} \cdot S_p(c_1, c_2) + \omega_{a,AD} \cdot S_a(c_1, c_2) \quad (11)$$

At request time, users select their intended action among the list of functions, and this choice is compared with the concepts functions to determine the "application domain" (set of possible concepts). The final semantic similarity measure is a weighted sum of the three measures (Equation 11). The normalized weights are determined by a "variability" of the type of property in the application domain, to give more importance to the type of property that appears the more relevant to characterize concepts locally.

The main interest of the similarity measure proposed by [Rodríguez & Egenhofer, 2004] is that it takes the “user intended operation” into account. It has, however, many drawbacks: it is based on features, with all the disadvantages described in the last chapter; this probably explains its general low score when similarity measures are compared; it relies on a distinction of concept properties into “tasks”, “functions” and “attributes”, which does not exist in ontologies.

[Janowicz, 2005] extends [Rodríguez and Egenhofer, 2004]’ similarity measure with thematic relations, so that the similarity measure would judge as more similar entity classes whose members “behave” in a similar way. He implements only the “participant” hierarchy, and do not consider the verb categories. His approach is limited in that it relies on the sole general interest of the user, and do not allow to contextualize according to a particular task. This approach has the same limitations as the previous one, and has no advantage as thematic relations are not implemented in ontologies.

Information content based measures are context-based measures, although it is not openly admitted. Indeed, they consider the frequency of use of the corresponding terms in a corpus to modify the measure done between the concepts. The consideration of context is limited to statistical information, though, and the approach is not well adaptable to the comparison of concepts across ontologies, unless the corpus used by the ontologies is the same.

10.2 Context-based similarity measures between instances

As the number of works that aims at contextualizing the measure of similarity between ontology concepts is limited, I include also a short survey of the context-dependent measure proposed by [Aleman-Meza et al, 2005]. This measure aims at evaluating the pertinence of all semantic paths that relate two instances, in a given context.

[Aleman-Meza et al, 2005] aim at discovering and ranking “*complex and meaningful relationship*” among metadata collected on the Web. Their hypothesis is that two entities on the Web are probably connected to one another by numerous paths, but the most direct ones may hide unexpected ones, which could be of higher interest for the observer. They use a commercial tool to extract data from the Web, formatted in RDF triples.

They propose to rank the different paths by which two instances can be connected, according to “regions” or sets of concepts that the user found interesting (Figure 13). The paths passing through regions with higher weight will be ranked first. They propose different parameters to privilege scarcity versus commonness, paths going through a “popular”⁶² concept, long versus short paths, etc.

⁶² having many relations with other instances

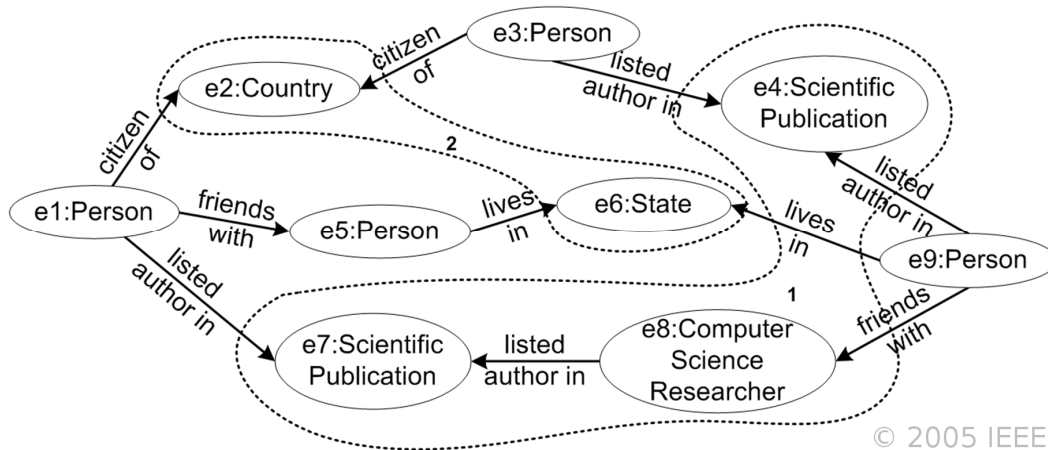


Figure 13: Paths relate instances through distinct regions [Aleman-Meza et al, 2005]

The particularity of [Aleman-Meza et al, 2005] is that they consider semantic relations, at the exception of *is-a* relations (as most edge-counting measures) and attributes (as features-based measures). It concerns instances and not concepts, and does not evaluate a couple of instances, but rather evaluate all the paths that connect the two instances; it is also not strictly a similarity measure, as instances given are probably different, and can be instances of totally different concepts. Their method is adapted when instances are connected with one another. But we need to deal with concepts, in order to retrieve relevant instances (data) that are not connected together.

10.3 Conclusion

What is considered contextual varies greatly for the measures considered here, as well as how the context is used to modify the value that these measures return. These measures could not be used “as is”, because, we have seen, context varies with the field and the application. It is therefore important to determine which information is actually contextual, if one desires to evaluate how concepts are related accurately. It is evident that the value returned by the measure should be altered in different ways, depending on the type of information that is contextual, and on their order of importance.

We will introduce the next part with our methodology to determine what information is contextual in our case, and how to use it in an appropriate way when comparing concepts.

PART III – Context-based Evaluation of Concept Pertinence

In this part, we introduce a methodology to establish a context-based solution. Then, we apply it to the reconciliation of ontologies in five chapters: in the chapter 12, we apply the first stage of the methodology; in the chapters 13-15, we apply the second stage of the methodology for the three usages of context that we chose to implement, and the chapter 16 contains the application of the last stage of the methodology.

Chapter 11: Methodology for an Effective Contextualisation

In this chapter we introduce a methodology to build an effective context-based solution. To this purpose, we observe the application and set objectives for the context-based solution. They will serve to prepare some guidelines that will help to characterize different contexts. The models and measures developed will then be combined in a global architecture that should be evaluated according to the objectives set in the first stage.

The methodology is made of three stages, each composed of a number of steps (Figure 14). The first stage consists in determining guidelines that will orient the application of the rest of the methodology. The second stage is done as many times as there are usages of context that will be implemented to achieve the set objectives: one will characterize the information that is contextual, available, and find how to model it and use it. The third and last stage assembles the different usages of context to form a global architecture that should satisfy the objectives set in the first stage.

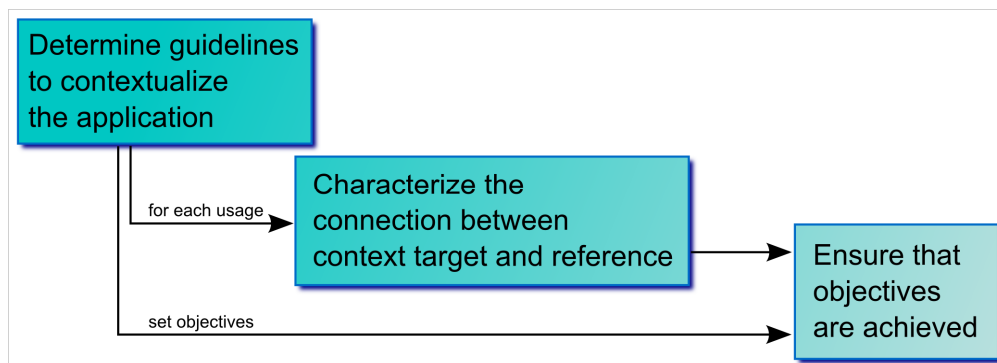


Figure 14: Schema to illustrate the methodology

To simplify the application of the methodology, the points to follow in each step will be itemized with the symbol ➤.

11.1 Stage 1: guidelines to contextualize the application

The first stage is composed of three steps (cf. Figure 15): set objectives for the context-based approach, select the usages of context that will be implemented to reach these objectives, and instantiate these usages to the current application with triples (target, reference, comparison aim).

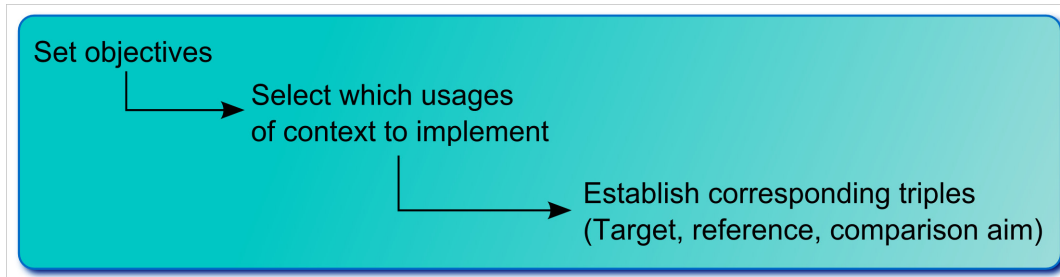


Figure 15: First stage of the methodology

① Set objectives

A context-based solution will be judged on its effect on the application, as any other type of solution. It is therefore essential not to lose the sight of the application. Contextualisation is expected to have a positive effect on the application: it should remove some of the application deficiencies or improve it in some way, while preserving most of its assets.

This step includes therefore the following points to follow. Determine:

- The application that will be contextualized
- The objectives set for the context-based solution:
 - The deficiencies that one wants to solve
 - The improvements that one expects to add
 - The assets that ought to be preserved

② Select which usages of context to implement

We propose in Table 14 a categorization of different usages of context, based on the literature review about context. Approaches that consider context actually collect contextual information, model it or employ it.

Either to solve a deficiency to solve or to add an improvement, distinct usages of contextual information might be successfully implemented. We propose to determine for each objective the usages of context to implement, so as to guide the further characterisation of context. This step includes therefore one point to follow. Determine:

- The usages of context to be implemented to reach the objectives

Table 14: Main usages of contextual information

What is done with context	Main usages of contextual information
Collect contextual information	<i>Transmission</i> to the user: History of operations can be traced during a determined period. When an object is found relevant, the operations that precede the discovery are supposed to characterize it. They are associated with the object as contextual information. This may be done to help the user to remember the circumstances of “discovery” of the object.
	<i>Provision</i> as specifications for other applications: Descriptive data is associated with the object, to be used by other applications that may judge them relevant (contextual).
Model contextual information.	<i>Reasoning</i> with inconsistent sets of rules about the same objects: Different typical contextual rules are described in a logic way, so that the rules used for reasoning can be context-dependent.
Employ contextual information (may collect and model contextual information for its purpose)	<i>Disambiguation</i> between different meanings: When the information contained in an object is not enough to determinate which further procedure to take about it, contextual information may help select the appropriate one.
	<i>Personalization</i> to one individual or group: Contextual information about that individual may filter out results that are supposed to be not relevant for that individual, or highlight in some way what is supposed to be more relevant.
	<i>Adaptation</i> to a particular technical device: Contextual information about that device may allow to furnish data to the device in a format more adapted to it
	<i>Evaluation</i> of the pertinence of an object for a given task: Contextual information about the object may allow to measure in what proportion the object is appropriate for the task.
	<i>Search</i> of an object through its connection with its environment: Information about an object may be searched by querying contextual information about it when information to characterize the object in itself is lacking or when there is no search engine to get the object with the type of information known. Example: the search of images is mostly done by meta-data about the image, such as modification date, name...)

③ Establish corresponding triples (target, reference, aim of the comparison)

In this step, we propose to determine the entities whose context is to be uncovered. This has to be done for each usage recognized in the previous step.

Contextual information describes the connection of an entity with its environment. It is peripheral information about an entity in focus. Contextual information is needed when the intrinsic knowledge of the entity into focus is not sufficient or not retrievable in a satisfying way. To determine what information is contextual, it is therefore necessary to first determine for which entities contextual information is needed.

When the goal is to *employ* contextual information, this information is never to be used alone, but it has to be compared with some particular information [Jouanot et al, 2003]. We will therefore define the following terms:

- “*target*”: the entity into focus whose context is to be uncovered.
- “*reference*”: the information to which the context of the target is compared.
- “*aim of the comparison*”: the direct purpose of placing the target in a referential, by comparing target context with reference.

For information to be contextual, it has to reveal something concerning the target that is relevant to compare the target with the reference, in such a way as to reach the ‘aim of the comparison’.

We propose in Table 15 a categorization of usages of context, with the possible target, reference, comparison aim, and types of context. This step includes therefore the following points to follow. For each usage of context, determine:

- The target
- The reference
- The aim of the comparison of target and reference
- The type of context to use

This methodology that we propose in this chapter is concerned with the determination of what is contextual in a particular setting, to *employ* it for a particular application. As the determination of the linguistic context is rather immediate, we will limit our methodology to the types of context “*connectional*”, “*situational*”, and “*context*”.

In this stage, we have showed how the application and objectives for the context-based solution were used to determine usages of context to implement. For each of these usages of context, there is a corresponding triple (target, reference, aim of the comparison). In the following stage, we will study how to characterize each of these triples (target, reference, aim of comparison).

Table 15: Usages of context and the corresponding targets, references, aims of the comparison, and types of context.

Usage	Event that triggers the need for context	Target	Reference	Aim of the Comparison	Purpose	Possible Types of context
<i>Disambiguate</i>	Situation where the meaning of an entity needs to be clarified	The entity to disambiguate	The possible variants	Select the correct variant	Use the entity according to its correct meaning	Linguistic, context
<i>Personalize</i>	The interaction of the user with an application	The user	The possible options	Select the more appropriate for the user	Personalize the interaction	Connectional, context
<i>Adapt</i>	The use of the device (at a given time and place)	The current environment, such as time and place (with sensors)	A list of situations described by a set of conditions	Select the situation(s) that correspond to the information sensed	Change the behaviour of the device when the situation occurs	Situational
	Interaction with the device for a particular service	The device	Set of configurations considered	Select the appropriate configuration	Use the device with the best possible configuration	Connectional
<i>Evaluate</i>	Situation of choice between different options	An option to evaluate	The criteria and requirements	Note the options according to their suitability with criteria and requirements	Act accordingly to the result, or transfer the information to the user	Context
<i>Find</i>	Information need, where only circumstances related to the entity are known	Entity to find	List of entities and the connections with their environment	Find entities that have are connected with the environment described	Act accordingly to the result, or transfer the information to the user	Connectional

11.2 Stage 2: Connection between target context and reference

The second stage is to be done for each distinct triple found in the first stage. It is made of six steps. It aims at characterizing the connection between the target context and the reference (Figure 16). They include the characterization of the target context, the model of the connection between target context and reference, and the technical details concerning the retrieving and measurement of data.

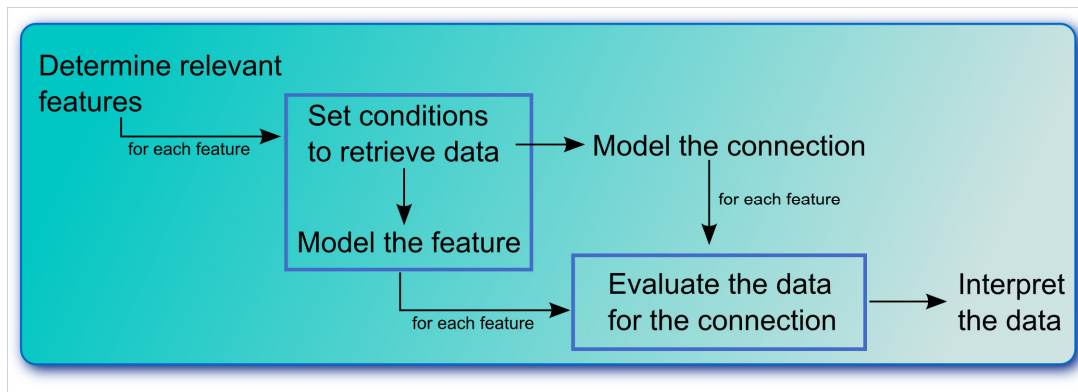


Figure 16: Second stage of the methodology

The steps are presented in a logical order, but the characterization of context will probably need backwards and forwards, comparably to the spiral model of development⁶³.

① Relevant features

The first thing to do is to search information about the target, by instantiating and answering the questions of the W5H⁶⁴. Usually only some of these questions are relevant to compare the entity to the reference information. The relevancy is judged according to the aim of the comparison that is awaited.

We define the following term:

- “*feature*”: any characteristic of the target that is judged relevant to the aim of the comparison between target context and the reference, and may participate in this comparison.

All features do not have the same importance to characterize the target context. It is necessary to make a selection of the most relevant features because the treatment of context is costly, including many phases, such as determination, retrieval, modelling and interpretation.

It is not necessarily evident to determine which features characterize the best the target context, as they are often inter-connected. The search of features is akin to the cause

⁶³See http://en.wikipedia.org/wiki/Spiral_model

⁶⁴Acronym for the questions Who What Where When Why How. For an example, see <http://www.able.state.pa.us/able/cwp/view.asp?a=15&q=127928>. These questions are notably used in the Zachman framework to categorize information [Zachman, 1987]

analysis so one could use tools such as the Pareto chart⁶⁵. This step includes therefore the following point to follow. Determine:

- The features the most relevant to compare the target with the reference.

② Conditions to retrieve data

Contextual data is usually not immediately available. To employ contextual information, it is therefore necessary to retrieve it beforehand. This can be done either automatically, or by manual input when data is either not easily available under a computerized form or too difficult to retrieve.

Depending on the sources, the quantity of data, but also their quality may change; all sources are not always available. All these criteria and others have to be used to determine the most relevant sources of data.

When retrieving data, it is necessary to adjust the precision of the retrieval method, so that the data should have an appropriate granularity. A measure more precise will result in a series of fluctuating values instead of a unique value. If these fluctuations have an impact on the relevant properties of the object observed, there are contextual. If not, the precision has to be reduced, or an average value has to be computed.

The data may not be always valid, depending on some particular conditions specific to the kind of data. For example, in conducting interviews, answers obtained with forms where some questions are suggesting a preferred answer may be rejected. Or some data measurement has to be done at some particular time of the day, to avoid an exceeding loss of precision.

This step includes therefore the following points to follow. Determine:

- The most relevant sources of available contextual data
- A pertinent precision and scale for the measurement
- The conditions that ensure the validity of the data retrieved.

③ Model of the feature

Contextual data may be used as is, but it may seem appropriate in most cases to structure it into a model, in such a way that the employment of contextual information should be made easier. This model plays the role of an interface with external software applications: it allows to modify the sources of contextual data, or to use the contextual data for other applications, with limited modification of the software already developed.

The structure of the model, the technology used to implement it, and so on, are dependent on the particular setting. We therefore propose to begin by specifying the

⁶⁵ See how to create a Pareto chart on <http://www.reliableplant.com/article.aspx?articleid=2251>

criteria and requirements for such a model. This step includes therefore the following points to follow. Determine:

- Criteria and requirements to represent the contextual properties in an appropriate and effective way.
- A model of context that conforms to these criteria and requirements.

④ Model of the connection

The connection between the target context and the reference should be known to compare them successfully. We therefore propose to model how the target context and the reference are connected.

This model should include a method to compare two kinds of information, so as to reach the comparison aim. This step includes therefore the following points to follow. Determine:

- A model of the connection between the context of the target and the reference.
- A method to compare the target features with the reference information.

⑤ Evaluation of the data for the connection

The data associated with a feature may have been retrieved from different sources, and be of various kinds. When comparing the target context with the reference, the data retrieved has to be evaluated in a way appropriate to each kind of data. These measures have to be compared with the corresponding information in the reference.

Especially where there is more than one feature, it seems desirable to compute one single measure for each feature. It is preferable that all data-specific measures should all be normalized, so that the measures could be more easily balanced (possibly dynamically), and normalized into a unique feature-specific measure.

The norm may be a weighted sum, a statistical summation of the measures, etc. The resulting value should constitute a meaningful indicator of the accordance of the feature-specific contextual data with the corresponding reference information. This step includes therefore the following points to follow. Determine:

- Normalized measures to evaluate each kind of contextual data by comparing it with the corresponding reference information.
- Weights to balance the different measures according to how their variation affects the feature.
- A norm to form a unique feature-specific measure from all the data-specific measures corresponding to a feature.

⑥ Interpretation of the data

When the aim of the comparison between target context and reference is reached, there is usually some slight transformation to do with the result to achieve the purpose of the corresponding usage of context. One should therefore develop a method appropriate to transform the result obtained so as to achieve the purpose associated with the usage of context.

This method has to “interpret” the evaluation of the match to transpose the result into an appropriate action (such as raise an emergency alarm depending on the contextual conditions), a value in an appropriate scale (for example out of 20), a qualitative appreciation, or other. This step includes therefore the following point to follow. Determine:

- A method to transform the result of all feature-specific measures to achieve the usage purpose.

We have described how to characterize the context of the target, model the connection between target and reference, and establish the measures to treat the corresponding data. In the coming stage, we will assemble the information determined for all triples (target, reference, aim of the comparison) to reach the objectives fixed for the context-based solution.

11.3 Stage 3: context-based architecture that fulfils the set objectives

The last stage puts together the models and measures determined in the second stage. It aims at finalizing and validating a context-based solution. It is made of three steps, which include to build the context-based architecture, evaluate it according to the objectives set in the first stage, and complete and adjust it with appropriate methods and models (Figure 17).

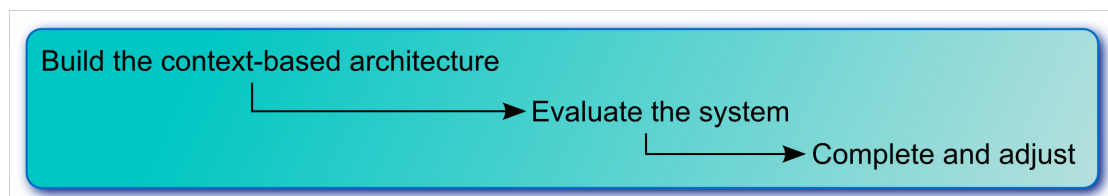


Figure 17: Third and last stage of the methodology

① Application Architecture

Now is the time to develop an architecture that combines all models and measures into a context-based system that achieves the objectives fixed. This step includes therefore the following point to follow. Determine:

- An architecture that combines all models corresponding to the different usages of context selected in the first stage.

② Evaluation

In the application of the first stage of the methodology, some objectives have been determined, that the context-based solution should follow. We will now evaluate the context-based solution according to these objectives. Determine:

- The evaluation of the context-based solution according to the objectives stated in the first stage.

③ Completion, adjustment

At this point, it may appear that the architecture does not fulfil all the objectives fixed during the application of the first stage of the methodology. This may come from objectives too ambitious, from unavailability of contextual data, or from omission of a usage of context.

But it may also come from the fact that the context-based solution is first of all a system. “Context-based” does not mean that only contextual information should be used to reach the objectives. Regular methods, models, and measures may be successfully added to the system. Determine:

- Methods, models and measures that may contribute to the success of the context-based solution in reaching the objectives

11.4 Conclusion

We have elaborated a methodology to develop a context-based solution, based on the analysis of the particular setting in which the application has to be contextualized. The application and the objectives for the context-based solution will help determine the different usages of context that are desirable for the context-based solution.

For each of these usages, one will need to characterize the context of the target, and to determine how to compare it with the reference information, in a way that permits to reach the aim of the comparison. Then one will have to interpret the measures so as to achieve the usage purpose.

Finally, the models and measures developed are assembled in a global architecture that is judged according to the objectives fixed earlier, and adjusted if needed.

In the five following chapters, we will apply this methodology to improve the reconciliation of ontologies, by taking into consideration practical implications. We will now follow the methodology elaborated in this chapter, to determine the relevant properties and the conditions to retrieve the associated data, to build a context model that should be appropriate for ontology reconciliation, and to determine appropriate measures for each feature.

Chapter 12: Guidelines to Contextualize Ontology Reconciliation

In this chapter, we apply the first stage of the methodology (Figure 18) that we just elaborated, so as to determine an appropriate context-based approach to improve the reconciliation of ontologies, based on existing mappings between ontology concepts.

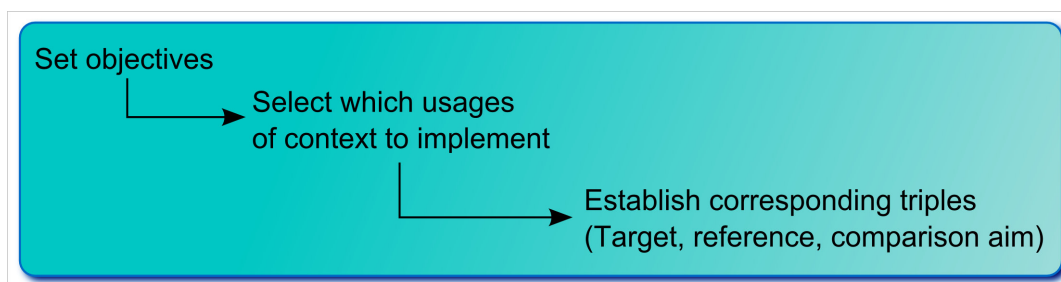


Figure 18: Application of the methodology (first stage)

12.1 Objectives

In this step, we first highlight our application, and then the objectives that we have for this application.

- Our application is the reconciliation of ontologies developed autonomously, and evolving independently, in the situation of collaborations limited in time between organizations.

Our objectives for this study are strongly related to the thesis introduction and particularly the problem statement. They express in what the consideration of context is expected to contribute to a better achievement of the application.

- The *deficiency* that we pointed out in the problem statement is that the reconciliation of independent ontologies by mappings does not relate ontology entities in a way that is relevant with practical implications. We hope to reduce this problem by relying on context to evaluate the connection between concepts. We believe that it is only through the consideration of contextual information that one can discover the practical meaning associated originally to the concepts [Porzel et al, 2006].
- The *improvements* that we want to add:
 - Expand the actual usability of the reconciliation. When ontologies are aligned, only a few concepts are related to one another by mappings. We want to propose an evaluation measure between custom concepts from two ontologies aligned.
 - Evaluate the reality of the reconciliation on the pragmatic level. Mappings are most of the time limited to signify a relative equivalence of two concepts from the ontologies aligned. Yet, as collaboration is based on the exchange of data, it is important to know the pertinence of a

concept to “replace” another concept. We want to propose an evaluation measure of the context-dependent pertinence for any concept defined by ontology of the collaborating organization

- The *asset* that we want to preserve is the flexibility of the federative approach. Indeed, the federative approach uses ontology mappings to reconcile ontologies, which does not alter the ontologies and is therefore adapted to the reconciliation of ontologies evolving independently. In the same way, we want our approach not to alter the ontologies mapped.

12.2 Usages of context

Different usages of context can serve our objectives for this study. We want to:

- *Disambiguate* between the pragmatic meanings of the concepts.
We want the measure to evaluate whether the two concepts compared may be associated with data that is comparable, in terms of practical use.
- *Personalize* the result.
Every agent has another understanding of the concepts of the company ontologies. Knowing which agent makes the interoperability request is therefore necessary to evaluate correctly the pertinence of concepts mapped from other ontologies.
- *Evaluate* the pertinence of a concept for the interoperability task.
When a request is made for an evaluation between two concepts, it is in the circumstance of some interoperability need. As interoperability is task-oriented, the measure of pertinence of a concept defined by an ontology of the collaborating organization should be dependent of the interoperability task that triggered the information need.

We define in Table 16 a list of terms that will simplify the description of our approach, and that we will use from now on.

Table 16: List of definitions of terms for our approach

Term	Definition
Applicant	The agent that makes the interoperability request.
Agent	User or software component that realizes some specified tasks in an autonomous way
Company	The organization of whom the applicant is member
Partner	The collaborating organization
Root concept	The concept selected in an ontology of the company that illustrates the kind of concept that the applicant seeks.
Concept enquired	The concept selected in an ontology of the partner ontologies that is the subject of the evaluation of pertinence.
Pragmatic meaning (for a concept)	Practical significance of a concept in terms of the actual data associated with it.

Table 17: Example of difference of pragmatic meaning for a concept

Concept	Book	Textbook	Book
Organization	Books on Demand www.bod.fr	Nathan Editions http://www.nathan.fr/	Amazon.com www.amazon.com
Data (for each book)	digital mock-up ISBN Barcode List of bookstores where it is sold ...	Title grade Mandatory/optional Year of publication Editors Authors Collection...	Title Authors Review List price Price In stock...
Point of view	Book edition	Specialised Book seller	Book seller

12.3 Triples (target, reference, aim of the comparison)

For each of the three usages distinguished, we have different triples (target, reference, aim of the comparison):

Disambiguate

We want to disambiguate the pragmatic meanings of the concept enquired by comparing them with the pragmatic meanings of the root concept⁶⁶. To do this, we need to compare the context of the concepts of the two organizations.

- Target: concept enquired
- Reference: context of the root concept
- Aim of the comparison: determine whether the concept enquired can be pertinent to “replace” the root concept.
- Type of context: for both the concept enquired and the root concept, the context considered is of the type “*context*”: the purpose is “*to explain the appearance or some particular characteristic(s) of the entity of interest*”.

Collaborations usually require that both organizations should have access to the data of one another. As the disambiguation process has to compare the possible pragmatic meanings of the two concepts, it seems appropriate not to make a distinction between concept enquired and root concept when determining the model of context, but instead, to make this latter symmetric.

Personalize

We want to evaluate whether the concept enquired is likely to answer the information need expressed by the applicant. This information need is described with the help of the

⁶⁶ See Table 17 for an example of different pragmatic meanings for a single concept. Although the concepts are almost identical, the information associated with the concept is either slightly different, as between TextBook and Book from Amazon.com, or completely different, as between the two concepts Book.

root concept, on which the applicant has a particular view. To have access to the pragmatic meaning that the applicant associates with the concept, we have to know the applicant's context. We therefore have to compare the context of the applicant with the context of the root concept:

- Target: root concept
- Reference: context of the applicant
- Aim of the comparison: select among the possible pragmatic meanings of the root concept, according to what pragmatic meanings the applicant may have intended to inquire.
- Type of context: the context of the applicant is the data which describes the applicant, according to a set of categories defined in accordance with a given purpose ("*connectional context*").

The context of the root concept is described here in the same way as the context of concepts for the usage of disambiguation described in the previous paragraph. To connect the different usages together, it seems appropriate to use the same context model for the root concept. As the context of the applicant has also to be determined, we will follow the methodology with inverted target and reference:

- Target: applicant
- Reference: context of the root concept

☞ *Evaluate*

We want to determine whether the concept enquired can be a satisfying answer for an interoperability need. We therefore have to compare the context of the concept enquired with the context of the interoperability need:

- Target: concept enquired
- Reference: context of the interoperability need
- Aim of the comparison: determine in what measure the data associated with the concept enquired may be relevant to answer the interoperability need that originated the request.
- Type of context: The context of the concept enquired has to present at least some correspondences with the context of the concept enquired described in the item "disambiguate"; it has also the type "*context*". The context of the request is of the type "*connectional context*".

For the same reason as in the previous paragraph, we will invert target and reference to follow the methodology:

- Target: interoperability need
- Reference: context of the concept enquired

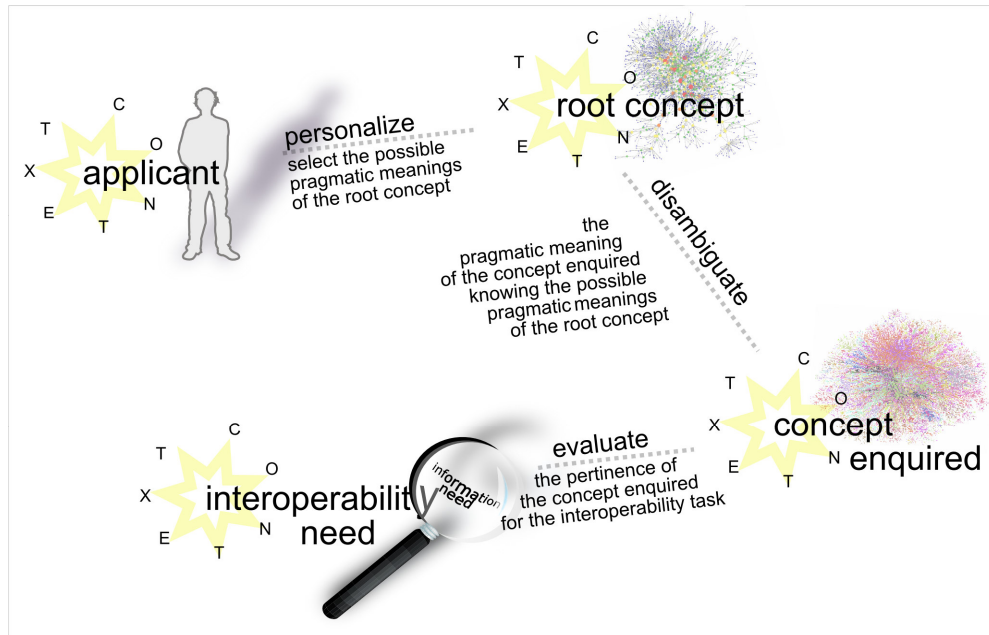


Figure 19: The different usages of context selected

12.4 Conclusion

We have determined in this chapter the application, the objectives of the approach, and the different usages of context that we will employ to reach our objectives. We want to improve the reconciliation of independent ontologies through mappings by proposing an evaluation of pertinence of a concept defined by an ontology of the partner organization.

To reach these objectives, we have determined for each of them the target, reference, aim of their comparison, and the type of context involved. We have three different contexts to consider: the context of the concepts, of the applicant, and of the interoperability need (Figure 19).

Now we have to determine what information about these three entities is contextual, available, and can be used for the comparisons between targets and references, in such a way as to reach the aims followed. We will therefore study, in the three following chapters, how to relate the context of the root concept with the context of the concept enquired for disambiguation, how to relate the context of the root concept with the context of the application for personalization, and how to relate the context of the concept enquired with the context of the interoperability need for evaluation.

Chapter 13: Connection between the Contexts of the Root Concept and of the Concept enquired

In this chapter we apply the second stage of the methodology described in the chapter 11, to characterize the connection between the context of the concept enquired and of the root concept for the usage of disambiguation (Figure 20). The purpose is to enhance the connection between ontologies reconciled by considering their practical implications. We will disambiguate the pragmatic meaning of the concept enquired by comparing it with the possible pragmatic meanings of the root concept.

As the root concept and of the concept enquired play symmetric roles in the disambiguation process, we determine the context of concepts without making a distinction between the two roles.

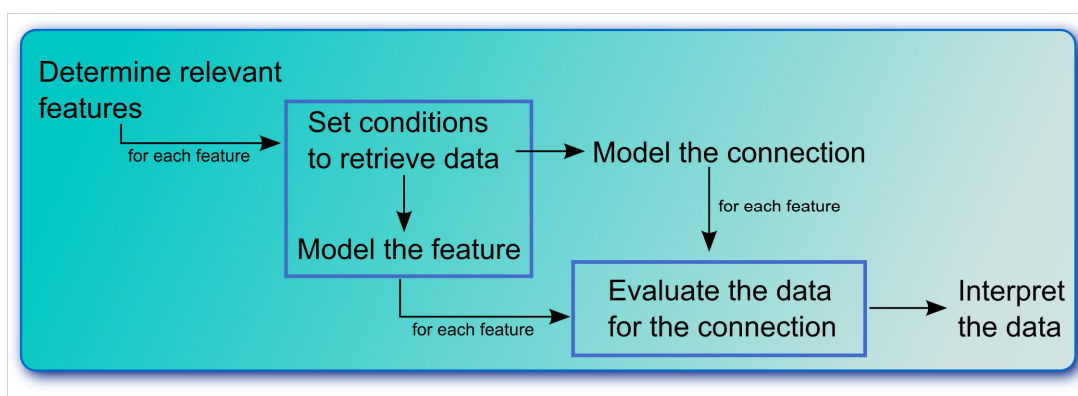


Figure 20: Application of the methodology (second stage, first usage of context)

13.1 Relevant features to describe the context of concepts

In this step, we determine the features that are the most relevant for ontology reconciliation.

[Ehrig et al, 2004] consider that the context of use of the ontology entities is the most pertinent context to evaluate the real meaning of concepts. The context of use could indeed be used to disambiguate the meaning of entities in ontologies, but it is also often confidential information, hardly available, and implemented in heterogeneous ways. It seems impossible to us to elaborate a method to compare ontologies contexts with one another, on the basis of such unpredictable information.

Ontology instances are strongly related to the context of ontology use. We have described extensional methods that compare instances of ontology concepts, (see Chapter 3, page 35). For this objective, instances usually serve to uncouple concepts that have no or few attributes associated with comparable data. In our setting, instances may be more available than in other settings, but are more likely to be confidential, and therefore not available. Also, to use instances with the aim of enhancing ontology reconciliation, mappings between concept attributes would be a prerequisite. These

mappings can hardly be provided by existing extensional method for alignment, as they are needed between all concepts (our evaluation measure is to compare custom concepts). We cannot rely on such information that is typically not provided, and is hard to obtain.

[Paslaru-Bontas, 2007] proposes to use the context of the ontology artefact development to evaluate whether it is pertinent to reuse a particular ontology for a given application. In our setting, the question is not to decide whether an ontology is to be kept or rejected; instead, the disambiguation is needed at the level of ontology entities. Because of this, we are not concerned by most of the criteria⁶⁷ that she established, but only with those that concern the meaning carried by the ontology.

We propose to follow [Paslaru-Bontas, 2007] and to keep as feature the context of development of the ontology. The ontology development may be characterized by a methodology, tools for edition, merging and alignment, the ontology engineers, and the perspectives with and for which the ontology was developed. Among these, we believe that these latter are the features that characterizes the ontology the best.

We therefore consider that the ontology context is composed of the perspectives with which experts have developed (and refined) the ontology. In our understanding of the notion of perspectives, we consider also the data associated with the concepts: some perspectives may mark differences in the kind of data associated with the concepts. We want to detect whether the root concept and the concept enquired have been developed⁶⁸ with or for perspectives that are compatible.

We keep for the term “*perspective*” the definition “*an individual way of regarding a situation, e.g. one influenced by personal experience or considerations*”⁶⁹.

The following point summarizes the answer to the corresponding point in the methodology:

- Feature: the perspectives with which the concept has been developed

13.2 Conditions to retrieve data

In this step, we evaluate how to get data related to the context of ontology development, the precision necessary, and the conditions to follow to be sure of the information validity.

The most relevant sources of data which characterize perspectives are certainly the applications, the databases and other data sources that the ontologies served to integrate, and the ontologies. It seems impossible to elaborate a method to retrieve any data from these resources, as their respective relevance and characteristics vary depending on the

⁶⁷ The criteria include the estimated relevance for the application domain, the quality of the modelling (user assessment by ontology engineers), the content (technical evaluation by domain experts), and the tasks for which the ontology was built.

⁶⁸ We do not make a distinction between the original ontology developments and the successive refinements.

⁶⁹ Definition from the Harrap’s 21st Century Dictionary

conditions of the ontologies development. The only available resource for which we can elaborate a method is the ontology itself.

The perspectives that influence the ontology development do not restrict the ontology engineers in such a way that there would be one unique structure and content possible. Rather, ontology building is subjective, and the impact of perspectives in the ontology is therefore not evident, but has to be analysed. Information associated with a perspective may be ubiquitous in the ontology, limited to a specific portion of the ontology, or be made of only one type of ontology entity.

It is not our objective to develop any automatic method to help experts determine the perspectives with which an ontology may have been developed, or entities affected by a given perspective. We nevertheless discuss ways to recognize perspectives manually: although knowing the type of a perspective does not permit to determine its effect on an ontology, we enumerate a few signs that may help to recognize the different types of perspectives in an ontology (Table 18).

Table 18: Signs to recognize various kinds of perspectives in an ontology

Type of perspective	The perspective may be recognized by:
Application domain	Higher granularity in a portion of the <i>is-a</i> hierarchy
Application or technical purpose	Presence of some entities (in particular, attributes) that have not much sense for the application domain, or are much more specific than the general granularity.
Viewpoint or role	Presence of <i>is-a</i> relations having different criteria of categorisation. This can be illustrated by an example from the partial classification of French roads (Figure 21): the concept “Route” is classified into several concepts that are associated with diverse perspectives, such as the environment in which the road is situated and the administrative responsibility. Such difference in the categorisation is though not enough to determine for sure different perspectives, as the different categorisations may come from historical background, but be used all for one single purpose.

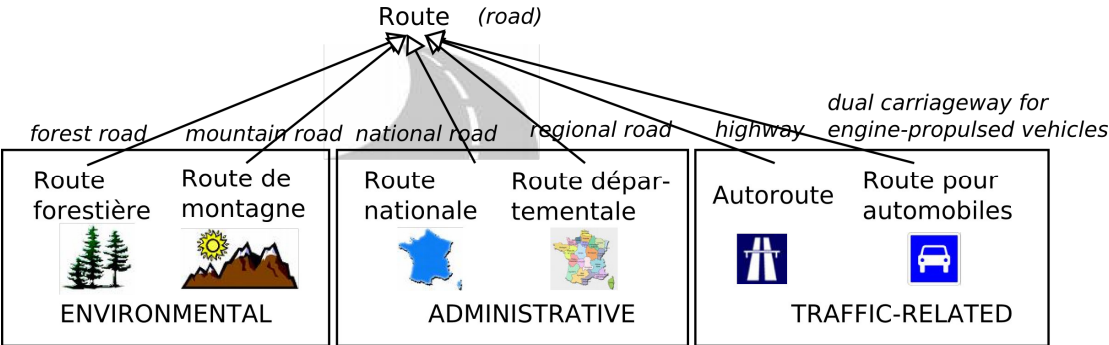


Figure 21: Classification of French roads

The ontologies have to be analysed well enough, so as to discover most of the perspectives that have guided its development, and that express the connection with the data. The success of the approach proposed here depends on this identification.

Ideally, the search of perspectives with which an ontology was developed should be done by a domain expert who participated to the ontology development, as he/she is in the right spot to know the rationale for the choice of each concept, its placement in the *is-a* classification, its name, attributes, and semantic relations. The following points summarize the answer to the methodology:

- Sources of available contextual data: the ontologies themselves
- Precision: as precise as possible.
- Conditions for relevant measurements: if possible, perspectives should be recorded by an expert who has participated to the ontologies development

13.3 Model of perspectives

In this step, we search what requirements the model of perspective has to meet, to fit our setting; then, we propose a model of representation of perspectives.

Requirements for the model

The following three requirements are essential to meet the needs for more reliable flexible ontology reconciliation. For companies to be interested in an approach to connect their data with those of their partners for temporary collaborations, the model of perspective should be:

- Simple, and use the standards, in order to limit the energy and time spent on it overall;
- Flexible, so as to allow to reuse what has been done for new collaborations;
- Secure, to limit access to the company's confidential data.

We will now discuss these requirements to determine corresponding guidelines and choices for the model:

☞ As the purpose of this approach is to improve ontology reconciliation, it seems appropriate to represent perspectives in an OWL-DL ontology as well. Doing this allows to use relations or mappings to relate perspectives together, and to take advantage of OWL namespace mechanism to ensure the uniqueness of perspectives names. Alternatively, a simple list of identifiers may be enough, with the risk of non uniqueness of naming, when two lists of perspectives from different organizations are compared.

☞ Our approach is based on ontology mappings, for reasons of flexibility, because ontology matching can be done between ontologies developed autonomously and evolving independently. The representation of perspectives should not require altering the ontologies either.

As the model is to help organizations to collaborate on projects during a short time, the model should permit reuse of most manual work required. The model should therefore keep separate what is specific to the organization, and what is specific to its collaborations. If perspectives are modelled by an ontology, this can be achieved by having organizations describe their own perspectives in an ontology of perspectives. When a need for collaboration occurs, the perspectives may be related by mapping these very ontologies.

☞ Temporal collaborations should not be holes in the organizations security watch. The data should be filtered before it is made available, so as not to provide any access to confidential data. Associating specific perspectives to confidential data, and managing perspective and data access from inside the organization should permit to restrict the access to confidential data⁷⁰. Alternatively, a reduced version of the organization ontologies might be made available for the collaboration, where all concepts and relations that seem confidential are withdrawn.

Definition of the model of perspectives

In line with the guidelines determined in the previous paragraphs, we propose to model perspectives in an “ontology of perspectives”. We model perspectives as classes, and relate them with object properties labelled according to the meaning of a corresponding switch of perspective. As switches of perspective have a completely different meaning when they are taken from one direction or the opposite, the object properties that model them are not symmetric.

It is possible also to use *is-a* relations to specialize perspectives, though there is no point in describing more perspectives than those that have really been recognized to impact the ontology development.

The Figure 22 shows an example of ontology of perspective, for the representation of product. Four perspectives are represented: Design, Geometry, Manufacturing and Machining. These perspectives are mainly related by object properties defined for the occasion. The perspective Machining is defined as a subclass of the perspective Manufacturing.

Technical concerns for the annotation of ontology entities with perspectives

To annotate the ontology entities with the appropriate perspectives, without modifying the ontologies, the idea that seems the more natural is to put that information in the ontology of perspectives instead. This may though oblige to share access with some more information about the ontologies, even information associated with perspectives judged confidential.

It has also the disadvantage to include many relations in the ontology of perspective (even though it is possible to limit the number of relations, for example by associating in one relation all the concepts subsumed by a concept), with the result of making it harder to understand, and heavier. Finally, OWL-DL does not permit to relate a concept

⁷⁰ It belongs to the organisation not to make appear any confidential information in the ontology of perspective itself.

to a semantic relation. It would therefore be impossible to annotate directly object properties and datatype properties⁷¹.

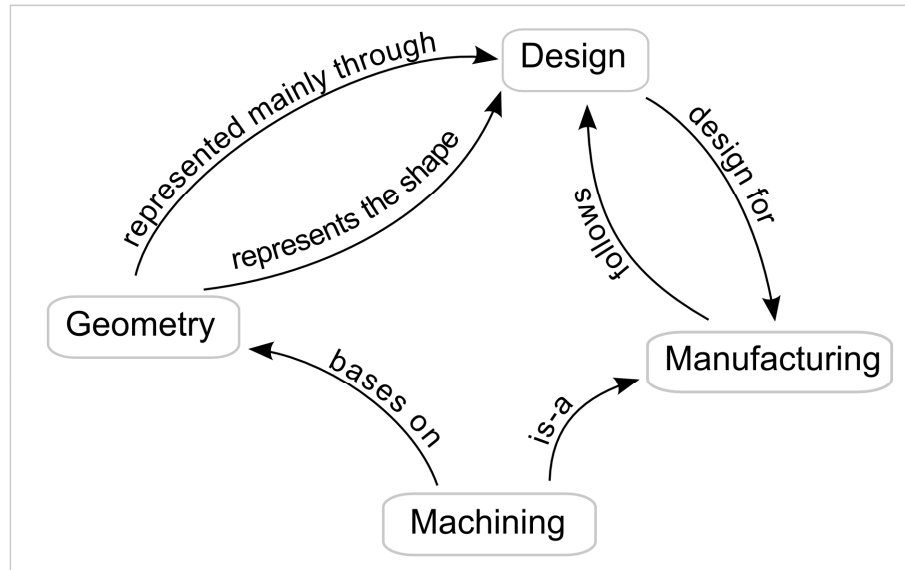


Figure 22: Example of ontology of perspective

We therefore propose to connect ontologies and data in another resource, for example, a relational database. This solution has the disadvantage of including another technology. Yet, every organization has a relational database, it is easy to install, and efficient to use. Many more technical solutions are available to relate relational databases to web forms (to allow non ontology experts to insert relations between perspectives and ontologies), and to use in applications. This solution is probably the best if efficiency is a concern.

Annotation of ontology entities with perspective

When we identify that a concept has been defined according to a given perspective, then its concepts that it subsumes may be viewed according to the same perspective (or by a perspective that specializes it). In such a case, the concept that subsumes the concepts associated with different perspectives possesses also these perspectives: in the example Figure 21, the concept “Route” has the three perspectives “environmental”, “administrative” and “traffic-related”, although it is not apparent.

In OWL, object properties and datatype properties are defined globally, although their domain and range can be restricted. We propose that if they are annotated with a perspective, then all the concepts in the domain are by default associated with the perspective as well; reciprocally, if a concept is associated with a perspective, then by default all properties in whose domain the concept is included are annotated with the perspective as well; unless it is explicitly removed from the perspective.

⁷¹ An alternative is to use attributes to annotate classes, attributes and object properties with perspectives, referencing them by their URI (String value).

We propose a basic set of operations that can serve of interface to annotate the entities of an ontology with a perspective:

- Add or remove a single concept
- Add or remove a concept with all the concept that it subsumes
- Add or remove an object property
- Add or remove an object property with all the object properties that it subsumes
- Add or remove a datatype property
- Remove all the datatype properties in whose domain a given concept is included

The following points summarize the answer to the methodology:

- Criteria and requirements: simple, flexible, secure
- Model of context: OWL-DL ontology, in which perspectives are represented as classes, and object properties represent “switches” of perspective. Alternatively, a simple list of identifiers may suffice.

13.4 Model of the connection between perspectives

In this step, we propose a model to compare perspectives defined in different organizations, and a method to associate perspectives to the ontologies that they have been recognized to influence.

Definition of the model

To compare the context of development of ontologies from different organizations requires establishing relations between their perspectives. This may be done by mapping the classes that represent the various perspectives. But in practice, the compatibility between perspectives may vary. To allow for further customisation, we therefore propose to relate ontologies of perspectives in a new ontology, in which we import the ontologies of perspectives from both organizations.

Figure 23 represents the connection between ontologies of perspectives. Ontologies of the organizations that collaborate are displayed at the bottom of the picture, and some of their concepts are mapped. Each organization has an ontology of perspectives (top of the picture), and the perspectives defined are used to annotate a selection of concepts in the organization’s ontologies. The ontologies of perspectives are connected with one another through asymmetric semantic relations.

In the case when perspectives are modelled by a simple list of identifiers, the relation between perspectives can be done, for example, in a relational database, and associated with the collaboration.

Method to compare perspectives from different ontologies

We want to compare perspectives in such a way as to determine which perspectives are compatible. We rely for this on object properties that relate perspectives defined in the ontologies of perspectives from the two organizations that collaborate.

We judge that the perspectives are compatible, in our setting, when they are related to one another, by an object property whose domain is in the organization ontology and whose range is in the partner ontology. If there is no such object property, then the perspectives are considered as not related.

There is no particular method to compare the context of concepts with one another, as this context is defined in the same way, by perspectives. The way to compare perspectives is therefore similar to ontology matching (or matching of elements in a list, if perspectives are represented by a list of identifiers).

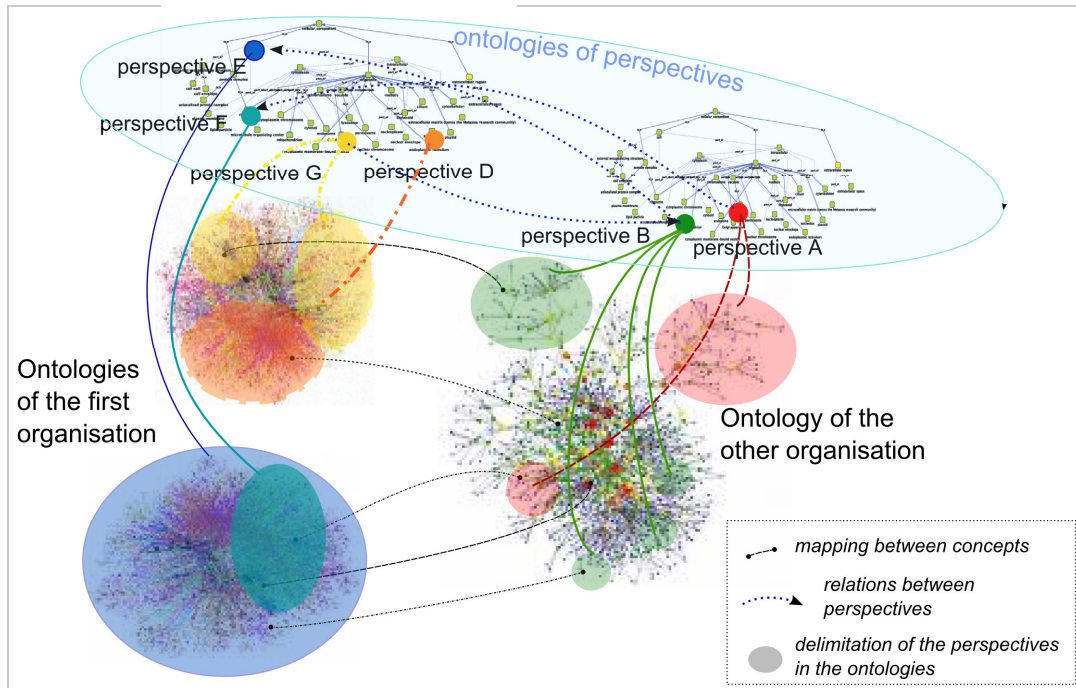


Figure 23: Connection between ontologies and their perspectives

The following points summarize the answer to the methodology:

- Model of the connection between perspectives: an ontology imports the ontologies of perspectives of both organizations, and relates perspectives with asymmetric object properties that represent the switches of perspectives.
- Method to compare perspectives: ontology matching

13.5 Evaluation of the data for the connection

It is hardly possible to be assisted by ontology matching tools to compare ontologies of perspectives. Yes indeed, terminological methods would not be efficient because perspectives are abstract concepts, whose terminology is harder to compare; structural methods would be limited by the reduced number of perspectives, and the few relations between them; semantic measures have no sense, as perspectives defined in an organization cannot be strictly equivalent with perspectives defined in another organization, and there are no rules that would allow to find more mappings by inference.

This limitation should not be an obstacle to the adoption of the approach, though, as the number of perspectives to match should not be high. Experts could take advantage of statistical information from the mappings found between the entities of the organization ontologies. If many concepts associated with a perspective are mapped with concepts defined by an ontology from the partner organization, and therefore associated with another perspective, then it is highly probable that these perspectives are related in some way. The following points summarize the answer to the methodology:

- Measure: there is no way to evaluate manually the connection between perspectives. Experts from both organizations should relate perspectives together.

13.6 Interpretation of the data to achieve the disambiguation

The root concept and the concept enquired are both associated with perspectives that have been recognized to influence the ontologies that define them. Perspectives from the company and its partner organization are related to one another by asymmetric object properties defined by experts at the time of the collaboration.

The disambiguation process is done by searching all couples of perspectives so that

- the first perspective is associated with the root concept,
- the second is associated with the concept enquired, and
- The two perspectives are related to one another by an object property whose domain includes the first perspective, and whose range includes the second.

Figure 24 illustrates the disambiguation process with an example. The root concept is annotated with the perspectives E, F and G. The concept enquired is annotated with the perspectives A and B, in the collaborating organization's ontology of perspectives. The perspective F is related to the perspective A, G with B, and E with B. But the only relations considered are those that relate the root concept's perspectives to the perspectives which annotate the concept enquired. Thus in this example, the two concepts are related through the couples of compatible perspectives (G, B) and (F, A).

The following point summarizes the answer to the methodology:

- Method to disambiguate the pragmatic meanings of the concepts compared: find all perspectives from the root concept that are related with a perspective to the concept enquired.

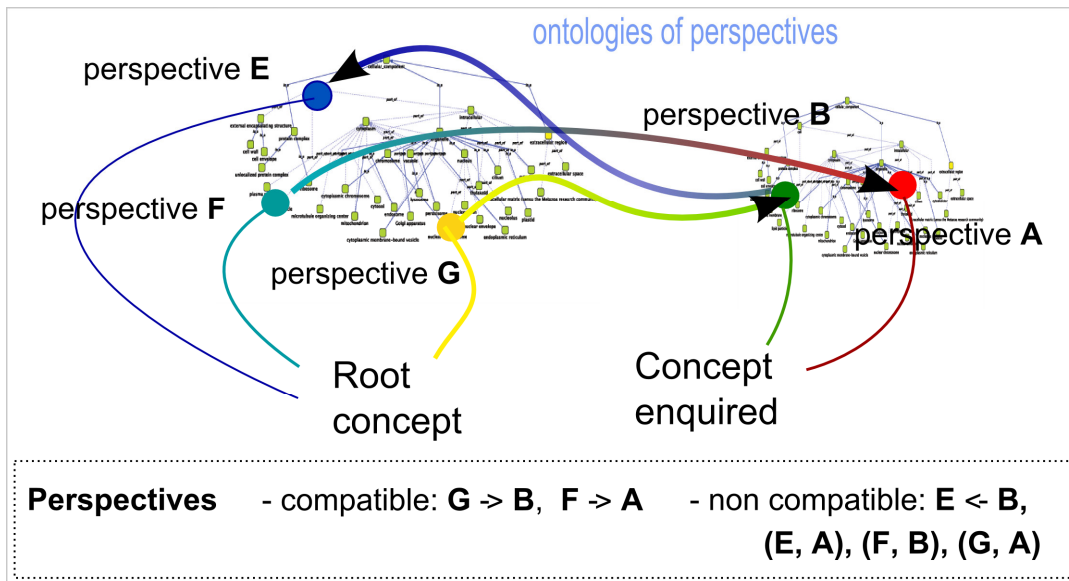


Figure 24: Illustration of the disambiguation process

13.7 Conclusion

We choose to represent ontology context by the means of the perspectives with which the ontologies have been initially developed and have evolved. Experts represent the perspectives of all the organization ontologies as classes in a new OWL-DL ontology (or alternatively, with a simple list of perspectives).

When a need for collaboration occurs, these perspectives are related with the partner perspectives, with asymmetric object properties, that might later be associated with task-dependent pertinence values. The perspectives of the root concept are compared with the perspectives of the concept enquired, so as to judge whether they are compatible.

When agents make a request, the perspectives associated with the root concept are compared with the perspectives associated with the concept enquired. The couples of perspectives that are compatible are returned.

But the perspectives defined for the company are not all relevant for every applicant. This may change dramatically the result if the evaluation of the connection between the concepts returned a high or low value of relevance based mainly on these perspectives. We will study the applicant's context in the next chapter, and see how it can be compared to the company perspectives, in order to find out with which perspective the root concept is to be associated, and get a more faithful evaluation of pertinence.

Chapter 14: Connection between the Contexts of the Applicant and of the Root Concept

In this chapter we apply the second stage of our methodology to determine the characteristics of the applicant's context and of its connection with the context of the root concept for the usage of personalization (Figure 25). The purpose is to improve the reconciliation of ontologies, by restricting the possible pragmatic meanings of the root concept to the ones that correspond to the applicant's perception of that concept.

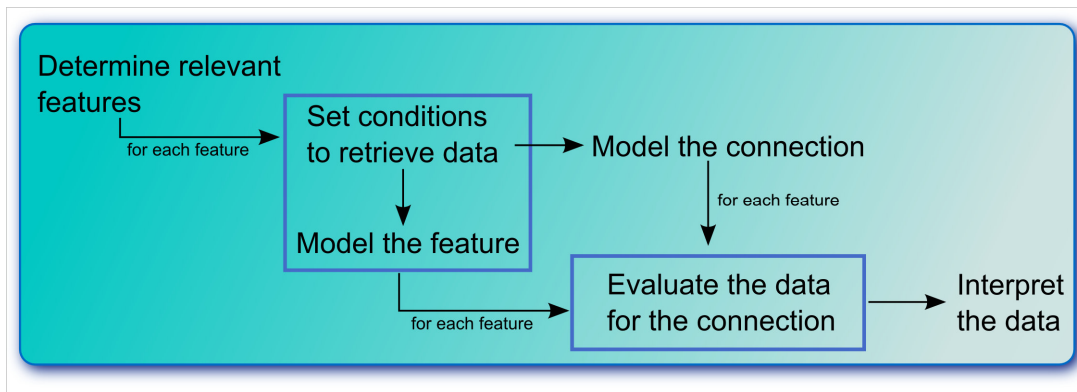


Figure 25: Application of the methodology (second stage, second usage of context)

14.1 Relevant features of the applicant's context

In this step, we determine the features that are the most relevant to characterize the applicant's context for ontology reconciliation.

The applicant's context in our setting is concerned only with information about the applicant that concerns the company, and with the request. We therefore describe the applicant's context by a portion of the company domain and tasks. As domains and tasks are complementary, we do not balance them with different weights; rather, we use them together, as one single feature.

The personalization is associated with the disambiguation, in that it restricts the number of possibilities for the pragmatic meanings of a concept. Therefore, the reference information has to be the same as in the previous chapter, that is, the model of perspectives: only some perspectives of the company ontologies are relevant for the applicant's domains and tasks. The following point summarizes the answer to the methodology:

- Feature: domains and tasks (complementary, to use as one feature) in which the applicant is involved.

14.2 Conditions to retrieve domains and tasks

In this step, we have to find out measures to determine the applicant's domains and tasks, and the precision required.

Our setting does not give any information about the agents and their use of some particular software application. We do not have any information to elaborate here any method to automatically retrieve which domains and tasks are relevant to them. Unless the application setting authorizes any automatic way, their context should be entered manually.

Users having different roles in the organization should obtain different results. The granularity of the data stored in the representation of domains and tasks has to be thin enough to distinguish applicants from one another, though not unnecessarily thinner. The following points summarize the answer to the methodology:

- Source: the agent has to furnish the information
- Precision and scale: domains and tasks have to be described precisely enough to make a difference between agents, and roughly enough to make the selection easy.
- Conditions to fulfil to ensure the relevancy of data: none.

14.3 Model of domains and tasks

In this step, we find criteria for a model of the context of the agent to be effective and propose such a model.

Criteria to represent domains and tasks in an appropriate and effective way

The criteria that we have found for the applicant's context are:

- Simple and effortless input
- Minimal frequency of updates necessary

Definition of the model

We therefore propose that the agents should describe their context by a simple selection of their organization domains and tasks in a list, corresponding to their potential information needs.

It is possible to improve this model, by associating a value of probability to the applicant's selection of domains and tasks. This value would be an estimation of the probability that the agent would request the system for an information need, for a particular domain or task. One could therefore present the most probable result or a list of results from the most probable to the less probable.

We will not retain this possible improvement in our model, as it makes the input of the agent more complicated and it is not evident to judge the probability of the information needs. It also makes the result more complex, because all different results should be returned, unless the domain and task corresponding to the request are indicated are indicated at the time of the request: who knows whether the information need judged less probable is not more important than the others? So, we cannot simply keep only the evaluation with the most probable domain or task.

We choose to represent domains and tasks as two simple lists of identifiers that can be recorded in a relational database. In the case where the domains or tasks are more numerous, one may add a level of more general domains and tasks. Unless a more complex classification of domains and tasks is required for other reasons, it seems more reasonable to privilege the simplicity of relational databases. Web-based forms can easily be generated, to update the information as well as to allow the agents to setup their contextual information. The following points summarize the answer to the methodology:

- Criteria: simple and effortless for users to fill their context, with minimum number of updates.
- The model of the applicant's context is made of a simple selection among the company domains and tasks, which are described as lists of identifiers.

14.4 Model of the connection between applicant's context and concept context

In this step, we model the connection between the applicant's context and the context of the root concept, and determine a method to compare these two contexts.

The company domains and tasks have to be compared with the company perspectives. We propose here to import the perspectives in a table in the relational database and to associate a normalized value of relevance between a perspective and a couple (domain, task).

To simplify the administration of such a model for the connection between domains/tasks and perspectives, we propose that default values could be introduced. For example a given perspective could be associated a default value, whatever the domain and task. The final value associated with a triplet (domain, task, perspective) is the maximal value among all the values possible. This permits to optimize the connection between domains/tasks and perspectives progressively, as needed. The following points summarize the answer to the methodology:

- Model of the connection between domains/tasks and perspectives: a value of relevance for each triplet (domain, task, perspective)
- Method to relate domains/tasks with perspectives: perspectives have to be matched with couples (domain, task).

14.5 Evaluation of the data for the connection

In this step, we search measures to evaluate domains and tasks, and establish the connection with perspectives.

Again, as for the users it is arduous to establish a profile directly against perspectives, the connection between domains or tasks and a perspective has to be determined by an expert from the company. It would preferably be someone who has participated to the ontology development, and is also involved in the company for a long time, so as to

know most of the company domains and tasks. The following point summarizes the answer to the methodology:

- Measure: there is no way to determine an algorithm to measure automatically the connection domain/task and perspectives; it has to be evaluated manually.

14.6 Interpretation of the data to achieve the personalization

The personalization process is done by associating a value of relevance for each perspective associated with the root concept:

- For each perspective, there may be different couples (domain, task) in the selection of the agent.
- These triplets (domain, task, perspective) are associated with a different value that indicates the relevance of the perspective for the couple (domain, task).
- The maximal value is kept as applicant-specific value of relevance for the perspective.

A value of relevance for each perspective can therefore be computed off-line from the applicant's context. The following point summarizes the answer to the methodology:

- Interpretation method for personalization: associate a applicant-specific value of relevance to each perspective associated with the root concept.

14.7 Conclusion

We propose that experts should elaborate an ontology of the company domains and tasks, so that users could easily define their context, made of a selection of their domains and tasks in the organization. The domains and tasks are related to perspectives, and experts can refine the evaluation of relevancy of perspectives for the couples (domain, task), progressively, as needed. Based on these weights, it is possible to compute an applicant-specific weight for each perspective.

We have exposed in this chapter how to determine which perspectives of the root concept are relevant, according to the applicant's domains and tasks. But when an applicant makes a request, it is possible that the data associated with the concept enquired should not be at all adapted to what the applicant wants to do with the data, even though some perspectives associated with the root concept and the concept enquired are compatible.

We will study the context of the interoperability need in the next chapter, and see how it can be compared to the company perspectives, in order to evaluate the pertinence of the concept enquired according to the interoperability need.

Chapter 15: Connection between the Contexts of the Interoperability Need and of the Concept enquired

In this chapter we apply the second stage of our methodology to determine the characteristics of the interoperability need and of its connection with the context of the concept enquired (Figure 26). The purpose is to improve the reconciliation of ontologies, by evaluating the pertinence of the data associated with the concept enquired for the interoperability need.

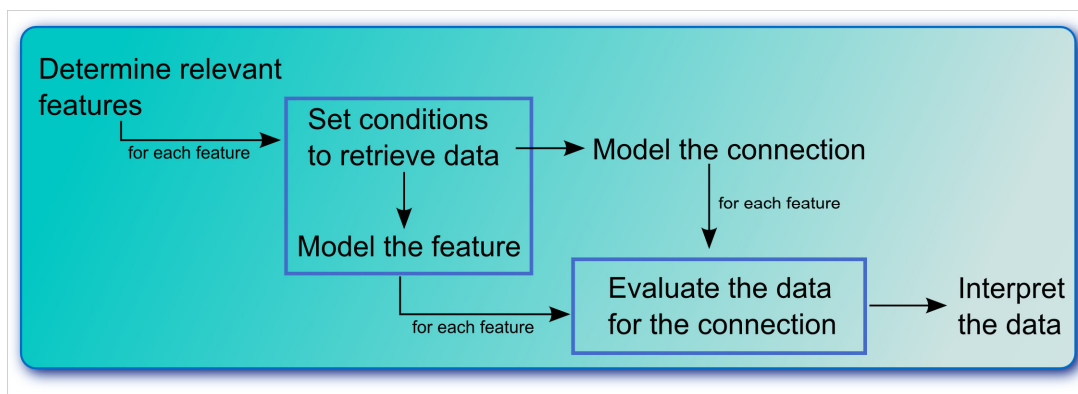


Figure 26: Application of the methodology (second stage, third usage of context)

15.1 Relevant features

In this step, we determine the features that are the most relevant to characterize the context of the interoperability need for ontology reconciliation.

Interoperability is task-oriented, which means that the request is made in a situation where the company is interested to use some data of the partner organization, with the objective of applying this data to a particular task. The evaluation of the data retrieved should therefore be done according to the task for which the data will be used [Byström and Hansen, 2002].

The evaluation of pertinence is associated with the disambiguation, in that it restricts the pragmatic meanings of the concept enquired, keeping only those that are pertinent for the interoperability need. Therefore, it is preferable that the reference information would be described in the same way as in the previous chapter, that is, the model of perspectives.

We define the term “*interoperability task*” for the data-dependent task for which the data retrieved from the request is intended. An example of interoperability task is the incorporation of designs of the partner products in designs elaborated by the company.

Other possible features, such as were the need has appeared, when and why, are not relevant to evaluate the concept enquired against the interoperability need. The following point summarizes the answer to the methodology:

- Feature: the interoperability task

15.2 Conditions to retrieve data about the interoperability task

In this step, we have to find out how to determine the interoperability task corresponding to an information need.

The applicant is the one that is the most knowledgeable about the interoperability task for which the request was initiated. Yet, the context of the interoperability need has to be compared with the reference, that is, with the context of the concept enquired. For that connection to be done beforehand, possible interoperability tasks have to be categorised. At the moment of the request, the applicant selects the interoperability task that typifies the best the intended use of the data retrieved.

The possible interoperability tasks depend on the specialty of the partner organization and on the data that this organization manipulates and makes available for the collaboration. This knowledge may therefore be recorded at the time of collaboration with the partner organization, or be reused from previous collaborations with similar organizations.

In the same way as for the company domains and tasks, a compromise has to be found for the precision of the categorisation of interoperability tasks. It is to be noted that the interoperability tasks can participate in the completion of a company task. Interoperability tasks are more specific as the company tasks, which are not directly associated with some particular data.

To categorise interoperability tasks, the experts who have related the perspectives from the company with the ones from the partner organization, and who have got a first access to the data of the partner, are in the best spot to identify the use that the company can make of this data. The following points summarize the answer to the methodology:

- Source of available contextual data: applicants are the ones that prepare the request, and that should select the category of interoperability task the closer to what they intend to do with the data.
- Precision: experts have to find a compromise for the precision of the categorisation of interoperability tasks, in the same way as for the definition of the company domains and tasks.
- Conditions for the measurements: the categorisation should be done by experts who have a good knowledge of the company business, and who have been involved in the connection between the perspectives for the collaboration.

15.3 Model of the Interoperability task

In this step, we find criteria for a model of the context of the interoperability need to be effective and propose such a model.

Criteria for the interoperability task model

The criteria for the model of the interoperability task flow from those defined for the model of perspective:

- Simple, and use the standards, to facilitate the adoption
- Reusable with different collaborations
- Secure, to limit access to the company's confidential data.

We will now discuss these requirements to determine corresponding guidelines and choices for the model:

☞ As interoperability tasks are to be selected by the applicant, and a simple categorisation is needed, we propose to model it with a list or a tree.

☞ When a new collaboration occurs, the interoperability tasks determined for previous collaborations can be used as pattern to build a categorisation of interoperability tasks for this particular collaboration. An alternative is to use a global categorisation of interoperability tasks, and to select for the collaboration the relevant interoperability tasks.

☞ There is no reason why the partner should know what interoperability tasks the company will do with the data that the partner organization makes available for the collaboration. The model should therefore keep separate the categorisation of tasks from the information shared with the partner.

Definition of the model

One may model the interoperability tasks by a simple list of identifiers in a relational database, or by a taxonomy.

One might desire to include in the model the data-level properties with which the data is described. The difficulty is that to compare the structure of data, properties, etc. with the partner data, the usual solution is to map properties from the sources of data of the company and of the partner, which is long and error-prone. We will therefore not include data-level properties to describe the interoperability tasks.

The interoperability tasks may be related to the company task. This might simplify the selection of the interoperability task at the request time, and restrict the number of company tasks considered for the computation of applicant-specific value of relevance. In that case, the computation of the applicant-specific value of relevancy associated with each perspective of the root concept should be delayed until the time of the request. Indeed, as the applicant-specific value of pertinence is the maximum relevancy value for each triple (domain, task, perspective), the result changes as the number of possible tasks is reduced.

The following points summarize the answer to the methodology:

- Criteria: simple, reusable, secure
- Model of interoperability task: a list of identifiers

15.4 Model of the connection between interoperability task and perspectives

In this step, we model the connection between the context of the interoperability need and the context of the concept enquired.

The pertinence of a perspective for an interoperability task does not change with the request. It may therefore be determined once for the collaboration, and be represented by a single value of pertinence. We therefore propose to model the connection between perspectives and interoperability task by associating a normalized value of pertinence to each couple (perspective, interoperability task). This may be implemented with a relational database.

As in the previous chapter, we propose to use default values so that experts may refine the evaluation of relevancy of perspectives for the interoperability tasks, progressively, as needed. While refining the connection between perspectives and interoperability tasks, experts may realize that the connection with the data is not well represented by the partner perspectives. They may ask the experts from the partner organization to incorporate more detailed perspectives that would better reflect the characteristics of the data associated with the partner ontologies. The following point summarizes the answer to the methodology:

- Model of the connection: a value of relevance is associated with each couple (perspective, interoperability task)
- Method to relate perspective and interoperability task: the interoperability tasks have to be matched with the perspectives that correspond to data pertinent to accomplish them.

15.5 Evaluation of the data for the connection

The connection between interoperability tasks and the partner perspectives has to be determined by an expert from the company. It would preferably be someone who was involved in the connection between perspectives of the two collaborating organizations, and therefore knows the partner perspectives. The following point summarizes the answer to the methodology:

- Measure: there is no way to automatically measure the connection between interoperability task and perspectives; it has to be evaluated manually.

15.6 Interpretation of the data to evaluate the concept enquired

The context of the interoperability need is made of the interoperability task for which the data that is to be retrieved from the request is intended. The evaluation process is

simply the retrieval of the values of pertinence that correspond to the perspectives associated with the concept enquired:

- For each perspective that is associated with the concept enquired, there is a unique value of pertinence for the interoperability task selected at the time of the request.
- If there is more than one value of pertinence, the maximal value is kept.

The following point summarizes the answer to the methodology:

- Interpretation method to evaluate the concept enquired for an interoperability task: it is the maximum value of pertinence for all couples (perspective, the interoperability task selected), where the perspective is associated with the concept enquired.

15.7 Conclusion

We propose that experts should elaborate a list of interoperability tasks that the company will be able to perform from the partner data. Experts evaluate the pertinence of the data related with the partner perspectives for the different interoperability tasks, and record the value of pertinence, for example in a relational database. Experts can refine the evaluation of relevancy of perspectives for the couples (perspective, interoperability task), progressively, as needed.

At the moment of the request, applicants select the interoperability task associated with the interoperability need that triggered the request. The pertinence of the concept enquired is evaluated against the interoperability task, by the maximum pertinence value associated with the interoperability task and any of the concept perspectives.

We have determined in this chapter how to evaluate the pertinence of the concept enquired, according to a given interoperability task. In the following chapter, we will consider the different models that we have built for the usages of personalization, disambiguation and evaluation, and design an architecture to use them all together.

Chapter 16: A Context-based Architecture that Takes into Account Practical Implications for Better Ontology Reconciliation

In this chapter we apply the last stage of our methodology to design an architecture that assembles all models developed, and ensure that it fulfils all the objectives that we have stated in the first stage (Figure 27). The purpose is to improve the reconciliation of ontologies by the means of a full context-based approach.

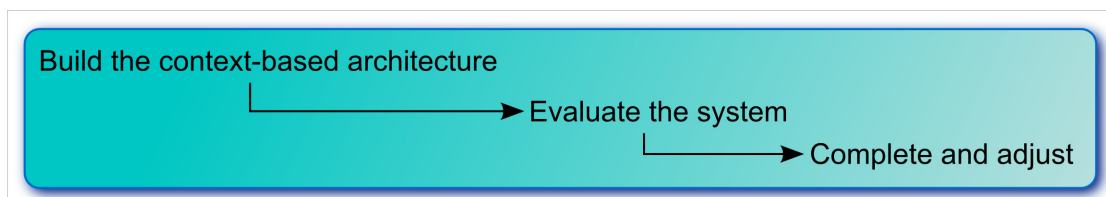


Figure 27: Application of the methodology (third stage)

Our objectives, as stated in the first stage of the methodology, were

- To *improve* the actual usability of the ontology reconciliation by providing an evaluation measure between custom concepts
- To *reduce* the lack of consideration of practical implications in ontology reconciliation by considering contextual information
- To *improve* the connection between concepts by considering the pertinence of the data associated with them.
- To *preserve* the flexibility imparted by ontology mappings for the ontology reconciliation.

16.1 Application architecture

In this step, we develop an architecture to use the models described in the three previous chapters, following the objectives described in the chapter 12.

The architecture is very simple: it is enough to put all usages one after another, in a linear way: personalization, disambiguation and evaluation (Figure 28, top of the figure). It is to be noted that the order of preparation of contextual data is almost the same: (1) Personalization (including the model of perspectives), then when a collaboration is planned, (2) the data for the disambiguation and (3) the evaluation is prepared or obtained (for the model of perspectives of the other organization). Finally, the applicant, root concept, concept enquired and interoperability task are given at the moment of the request.

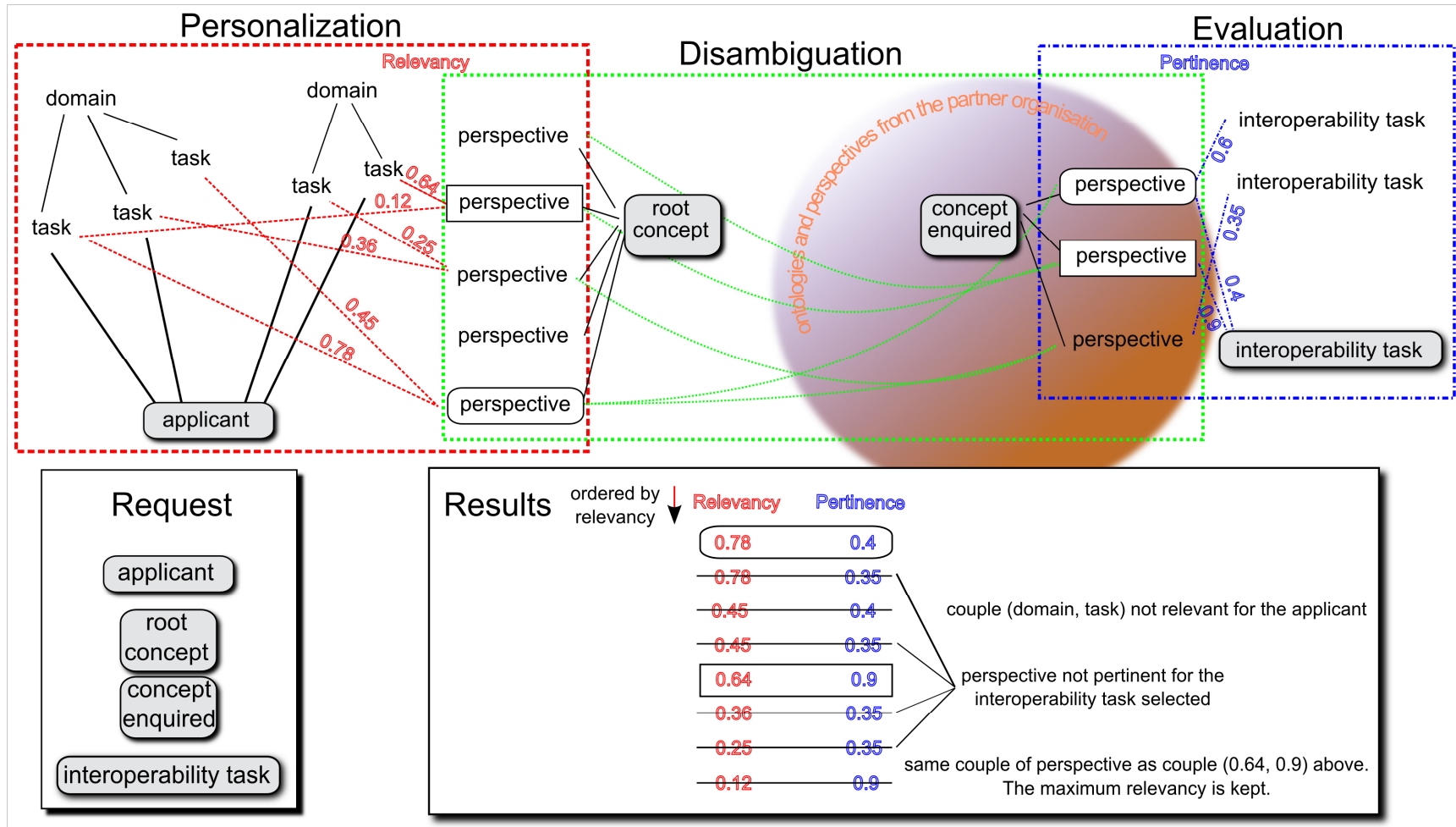


Figure 28: illustration of the architecture with an example

The request made by the applicant is made of the root concept, the concept enquired, and the interoperability task,

- The root concept describes the kind of data that the applicant searches. The applicant may select it in one of the company ontology.
- The concept enquired is the concept in one of the partner ontologies, which is to be evaluated.
- The interoperability task signals the intention that the applicant wants to do with the data retrieved. The applicant may select it in a list.

☞ The usage of personalization restricts the perspectives associated with the root concept, to keep only those that are relevant for the applicant. With each of these perspectives, a value of relevancy is associated that indicates the highest relevancy possible of the perspective for the applicant's domains and tasks.

☞ The usage of disambiguation limits the perspectives associated with the concept enquired to those that are compatible with the perspectives associated with the root concept, and that are relevant for the applicant.

☞ The usage of evaluation associates a value of pertinence to all perspectives associated with the concept enquired, according to the relevancy of the data from the partner for the interoperability task selected at the time of the request.

When a request is executed, and the computations are done, the result may be composed of the following information:

- A list (relevancy, pertinence) for each possible couple of perspective for the root concept and concept enquired (only couples where neither relevancy nor pertinence is null are shown). Each row of this list may be completed by a list of the company domains and tasks that are relevant for the corresponding perspective. The list may be ordered by decreasing value of relevancy.
- More detail can be given, if needed (in particular when there is no result, to find the reason, and know how to enlarge the request):
 - a. For each row, the couple of perspectives.
 - b. Proportion of the root concept perspectives that are relevant for the applicant's context.
 - c. Proportion of the concept enquired perspectives that are compatible with any of the root concept perspectives.
 - d. Proportion of concept enquired perspectives that are pertinent for the interoperability task.

The three usages put together permit the context-based evaluation of the concept enquired. The order for the computations that seems the most intuitive is the opposite as the order of preparation of contextual data: as the interoperability task is given, one can easily find the various possible couples of relevance and pertinence (see the results sheet at the bottom of Figure 28)

The following point summarizes the answer to the methodology:

- The architecture is made of a simple reordering of the usages into the sequence: personalization, disambiguation and evaluation. These usages form a chain that should relate the applicant to the interoperability task. The connection between domains/tasks and perspectives indicates the relevancy of the perspectives for the domains/tasks; the connection between compatible perspectives indicates the compatibility between some perspectives of the organisations; the connection between perspectives and interoperability task indicates the pertinence of the perspective for the task. The couples (relevancy, pertinence) found are the result of the global context-based evaluation of the concept enquired.

16.2 Evaluation

In this step, we examine the context-based solution that we just proposed at the light of the objectives that we recalled at the beginning of this chapter.

☞ Our solution improves the actual usability of the ontology reconciliation by providing an evaluation measure between custom concepts.

☞ Our solution is based on contextual information, including the applicant, the ontology development (including the data that the ontologies serve to integrate), and the interoperability need. This contextual information puts ontology reconciliation in the larger context of the collaboration between two organizations, and of the practical use of the data retrieved through the collaboration.

☞ Our solution considers the pertinence of the data associated with the concept enquired. It does not consider directly the data associated with the root concept, but the perspective associated with it should show the connection with the data. The evaluation of the connection between concepts is based on the comparison of the concepts perspectives, thus taking into consideration the data associated with them.

☞ Our solution does not require the modification of ontologies, nor of data sources, thus preserving the flexibility of federative approaches. More, if an ontology evolves and that an entity name is altered, for example, the solution will still work (not for the new entities or the entities altered, but in general).

The solution that we proposed answers almost completely the objectives posed at the beginning of the application of our methodology. There are though two points that need more discussion:

- The contextual information considered does not provide any help to reconcile the actual format of the data associated with the concepts.
- The evaluation of the connection between concepts is limited to the comparison of perspectives. It does not consider any of the information that is usually considered to define the meaning of concepts relatively to one another: mappings and semantic relations (including *is-a* relations).

Concerning the first point, that is, the absence of consideration of the format of the data that company and partner make available for the cooperation: the approach aims at improving the connection between ontologies, which describe the meaning of data and

not its format (higher level of abstraction). The reader interested in reconciling data format with context may read [Firat et al, 2007].

Concerning the second point, we will attempt in the next section to provide a generic measure based on the information mentioned. This generic measure should be combined with the previous evaluation measure to render a better evaluation of the connection between concepts.

The following point summarizes the answer to the methodology:

- The context-based solution that we proposed provides an answer for all the points of the objectives. There is though a serious limitation: the evaluation does not even consider the semantic relations and mappings that may relate the root concept and the concept enquired, which may therefore be concept very dissimilar. We propose therefore to include a generic semantic measure to our approach.

16.3 Completion by a generic measure to better compare concepts

In this step, we will seek to complete our context-based solution with an evaluation of the connection between concepts that should use the information described in ontologies and mappings.

Kind of measure needed

We have discussed in the literature review three types of measures between concepts. These measures determine either the similarity between concepts in an ontology, the relatedness between concepts in an ontology, or the similarity between concepts defined in different ontologies.

We stated earlier that the evaluation of pertinence of the concept enquired, and its comparison with the root concept should not be limited to a strict notion of similarity. Measures of relatedness seem therefore a proper alternative, as they consider different types of semantic relations between concepts. But ontologies define custom semantic relations⁷², whose meaning is not directly obtainable. In these conditions, it seems almost impossible to define in which conditions these semantic relations might be adequate to relate concepts. This is particularly true when combining two semantic relations or more: it is not evident that two concepts related through two different semantic relations would still be judged as semantically related.

Considering then the measures of similarity between concepts, it seems evident that the more appropriate ones would be the measures between concepts defined in different ontologies. But these measures actually rely on terminological, structural matching methods. These methods require a long computation time, while the measure that we propose has to be done between custom concepts, determined at the request time. Anyway, we want to improve ontology matching, not do it again. We want to rely on the work done, as [Ehrig et al, 2004] suggested to have layers for similarity measures.

⁷² As said in the literature review, there are very few standard semantic relations that could serve to categorize user-defined semantic relations.

That is why we want to rely on existing mappings, and if possible on existing measures between concepts.

The matter for this approach is not to improve the measurement of similarity, but to combine a measure of similarity with our context-based solution. I therefore propose to select a known intra-ontology similarity measure, and to adapt it so as to use mappings and compare concepts across ontologies.

Choice of an existing semantic measure of similarity

Feature-based measures usually consider attributes, sometimes semantic relations, as features, to compare concepts with one another. But when the concepts to compare are defined in different ontologies, their attributes and semantic relations will be completely different. Even in the case where some attributes and semantic relations may be related to their equivalent by mappings, it is probable that it would concern a minority of them, resulting in an ineffective measure.

Information-content based measures rely on a corpus of documents associated with the ontologies. When comparing a concept defined in an ontology with a concept defined in an ontology from another organization, it is highly probably that the corpora associated with the ontologies should be very different from one another. An option would be to define a common corpus at the time of the collaboration, but as the documents are certainly specialized, they would come from one organization or the other, and the result would not be balanced either, with much work involved. Similar arguments can be raised against extensional measures.

Edge-counting based measures are based on the *is-a* classification that is almost omnipresent in ontologies. The *is-a* relation renders the way concepts are categorized in the ontology. Although it has not exactly the same meaning everywhere, it gives means to evaluate whether two concepts have a similar degree of abstraction, and in what proportion they can be said to have a similar type. We are not interested solely in evaluating the strict similarity between the root concept and the concept enquired, but it is interesting to be able to compare the degree of abstraction of concepts, and whether they can be categorized in a similar way. We will therefore choose an edge-counting measure.

[Petrakis et al, 2006] gives a evaluation of 12 measures of the 4 types, for the ontologies WordNet and MeSH. The edge-counting measure from [Leacock and Chodorow, 1998] is a measure of similarity which is one of the simplest, yet is either first or second in their tests (Equation 12). In [Budanitsky and Hirst, 2001], this same measure gets results homogeneously good. Therefore, we will consider this measure as our basic similarity measure.

Adaptation of the measure to compare concepts from different ontologies

We want to adapt a measure that compares concepts defined by a single ontology, based on relations of subsumption, in such a way that it could be used to compare concepts from distinct ontologies, using mappings between concepts. The two ontologies to compare have probably a different depth, and therefore it is not possible to keep exactly the same formula.

Let a root concept c_1 defined by an ontology O_1 be compared with a concept c_2 defined by an ontology O_2 . Let us say that there exists a mapping of equivalence (for example, a mapping with a value of similarity between 90% and 100%) that relates the concept c_{m1} defined in O_1 with the concept c_{m2} defined in O_2 , so that the concepts c_1 and c_2 can be related to one another by a succession of *is-a* relations, and the mapping $c_{m1} - c_{m2}$.

The edge-counting-based distance on which the similarity of [Leacock and Chodorow, 1998] is based (Equation 3), $\text{len}(c_1, c_2)/2D$, is linear (Table 19). I therefore propose to replace the distance from c_1 to c_2 by the sum of the distances from c_1 to c_{m1} and from c_{m2} to c_2 . If the mapping that relates c_{m1} to c_{m2} is a mapping of subsumption, it is equivalent to adding a subsumption relation between c_{m1} and a temporary concept mapped as equivalent to c_{m2} ; therefore, in that case, we add 1 to the length computed.

We propose a modification of the formula (Equation 13) so that the result should be normalized. Also, by summing the distances $\text{len}(c_1, c_{m1})/D_1$ and $\text{len}(c_2, c_{m2})/D_2$ we increase the range of the distance to $[0, 2]$. The norm should therefore return values between 0 and 1, decreasing on the interval $[0, 2]$. We increase the range a little bit more, to take into account the possible consideration of a mapping of subsumption. Figure 29 shows the norm chosen (in green) compared to the logarithm chosen by [Leacock and Chodorow, 1998] to transform the distance into a value of similarity.

There may be different mappings possible to relate c_1 and c_2 . As there is no reason to select a mapping over another, the final similarity measure is the one that selects the mapping so as to maximize the similarity value (Equation 14).

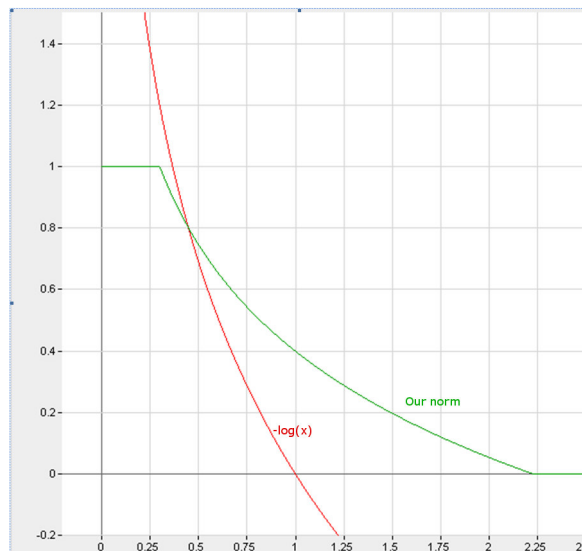


Figure 29: Comparison of the norms ($-\log(x)$ and our norm)

Table 19: Common functions for the measures of similarity

$c_1 \in O_1, c_2 \in O_2$	Concepts defined respectively in an ontology of the company, or in an ontology of the partner
$len(c_1, c_2)$	minimum number of <i>is-a</i> edges needed to relate the concepts
$\lambda_{subs}(c_{m1}, c_{m2})$	1 if the mapping determined by (c_{m1}, c_{m2}) is a mapping of subsumption, 0 otherwise
$p_1 \triangleleft c_1, p_2 \triangleleft c_2$	Perspectives p_1 and p_2 associated respectively with the concepts c_1 and c_2
$rel_{applicant}(p_1)$	Applicant specific relevancy
$pert_{ITask}(p_2)$	Interoperability-task specific pertinence

$$sim_{LC}(c_1, c_2) = -\log \frac{len(c_1, c_2)}{2D} \quad [12]$$

$$norm(x) = \min(\max(-\log(x^2) + 0.4, 1), 0) \quad [13]$$

$$sim(c_1, c_2) = \max_{\{m, m \equiv c_{m1} - c_{m2}\}} \left[norm \left(\frac{len(c_1, c_{m1})}{2D_{O_1}} + \frac{len(c_{m2}, c_2)}{2D_{O_2}} + \frac{\lambda_{subs}(c_{m1}, c_{m2})}{D_{O_1} + D_{O_2}} \right) \right] \quad [14]$$

$$sim_{applicant, ITask}(c_1, c_2) = \max_{p_1 \triangleleft c_1, p_2 \triangleleft c_2} \sqrt{rel_{applicant}(p_1) \cdot (\lambda \cdot sim(c_1, c_2) + (1 - \lambda) \cdot pert_{ITask}(p_2))} \quad [15]$$

with $0 \leq \lambda \leq 1$

Integration of the generic semantic measure with the approach

The generic similarity measure is aimed at completing the context-based evaluation measure that we have determined in the three previous chapters.

When observing the illustration of the architecture, Figure 28, we see that the similarity measure will relate the root concept and concept enquired, that were previously connected only indirectly by their perspectives.

The generic similarity measure has to be computed, at least partially, at the moment of the request: the global measure has to be possible for evaluating any couple of concepts. The computation of the measure though requires a long computation time, which cannot be done at the time of the request. We propose therefore to adapt the method proposed by [Bidault, 2002] to our configuration. We will therefore associate with each concept a string of alphanumeric characters unique for the ontology indexed. This string will allow to compare easily the concepts relatively to the paths made of *is-a* relations that relate them to the ontology root.

At the time of the request, the mappings that relate the ontologies in which the root concept and the concept enquired are defined are selected. For each mapping $c_{m1} - c_{m2}$, the distances $len(c_1, c_{m1})$ and $len(c_2, c_{m2})$, where c_1 is the root concept and c_2 the concept enquired, are computed. The required comparison of thousands of strings (twice the possible number of mappings) of length inferior to a hundred characters (a possible

average depth of the concept in the ontology) can be done in less than 100ms on a recent computer.

Earlier in the chapter, we have proposed that the comparison of the concepts should be returned essentially with a couple of values (relevancy, pertinence). There is one single value of similarity for many possible couples of (relevancy, similarity). It might be interesting to propose a unique value of pertinence of the concept enquired. This might help, for example, when comparing the pertinence of different concepts for a same interoperability need. We therefore propose to compute a global value of pertinence based on the product of the relevancy with the parameterized average of the values of similarity and pertinence (Equation 15).

The following point summarizes the answer to the methodology:

- We propose to integrate into the context-based architecture a generic measure based on an existing measure of semantic similarity, and on mappings: the measures of semantic relatedness rely on semantic relations, which are hardly comparable across ontologies, and existing measures of semantic similarity across ontologies use the same methods on which ontology matching tools are based.

We choose an edge-counting measure, as such measures rely on *is-a* relations, which, despite their possible ambiguity, are present in most ontologies. We select the measure from [Leacock and Chodorow, 1998] for its performance and simplicity. We adapt the measure for the comparison of concepts across ontologies, and for fast evaluation of custom concepts. We finally propose a global measure to evaluate the pertinence of a concept.

16.4 Conclusion

The models for our three usages of context form a context-based evaluation measure of pertinence. They are put together in the sequence: personalization, disambiguation, and evaluation. A context-based solution based on this architecture may return a couple of (relevancy, pertinence) for each possible couple of perspectives.

The architecture answers all of the objectives set in the application of the first stage of the methodology. Yet, there is a serious limitation, that is, the root concept and concept enquired are not at all compared with any of the semantic relations defined in ontologies, nor by mappings; they may therefore be very dissimilar, while the root concept is supposed to guide the evaluation of the concept enquired.

We therefore propose to include a generic measure of semantic similarity into the architecture. We adapt the semantic similarity measure from [Leacock and Chodorow, 1998] to the comparison of concepts across ontologies. We use the indexing method proposed by [Bidault, 2002] to provide a fast computation of similarity between the concepts at the time of the request.

We have now applied the methodology to the reconciliation of ontologies, and have designed an architecture composed of the context models that we have characterized. With this architecture, we propose an evaluation of pertinence of a concept defined by

an ontology from a partner organization, guided by a concept defined by an ontology of the company and by the interoperability task for which the data retrieved is to be used. The measure returns a global value of pertinence and possibly a value of generic similarity as well as a list of couples (relevance, pertinence) for each couple of perspectives ordered by relevance.

In the next part, we will display how a system may be implemented following the approach proposed. We will also illustrate with an example how such a system may be used, and give some technical details of the implementation.

Part IV – Implementation and Application

In this part, we will present an implementation and application example of the approach. We will first show the architecture of a system that implements the approach, and how it is supposed to be used. Then we will give a scenario of use, where an agent encounters an interoperability need while using a business application. The two chapters that follow will illustrate the preliminary work that must have been done by an expert, to prepare the contextual information that the system has used to evaluate the possible solutions. The last chapter details the algorithms and the structure of the database.

Chapter 17: Principles of Implementation of the System

In this chapter, we will display a few figures to give an overview of the context-based system. First, we will show how the context-based system may be utilized. Then, we will present use cases for the agent and the different roles of experts. Finally, we will display the dependencies between the different processes.

17.1 Working of the context-based system

The Figure 30 illustrates a possible working of the context-based system. While interacting with a user, the business application submits a request to the context-based system. This request is made of the root concept and concept enquired, of the user ID, and of the interoperability task selected. The context-based system relies on the user context, on ontologies and mappings, on perspectives and interoperability tasks to provide a response to the business application. The response is made of a value of global pertinence, a value of similarity, as well as couples of values of relevancy and pertinence corresponding to the possible couples of perspectives.

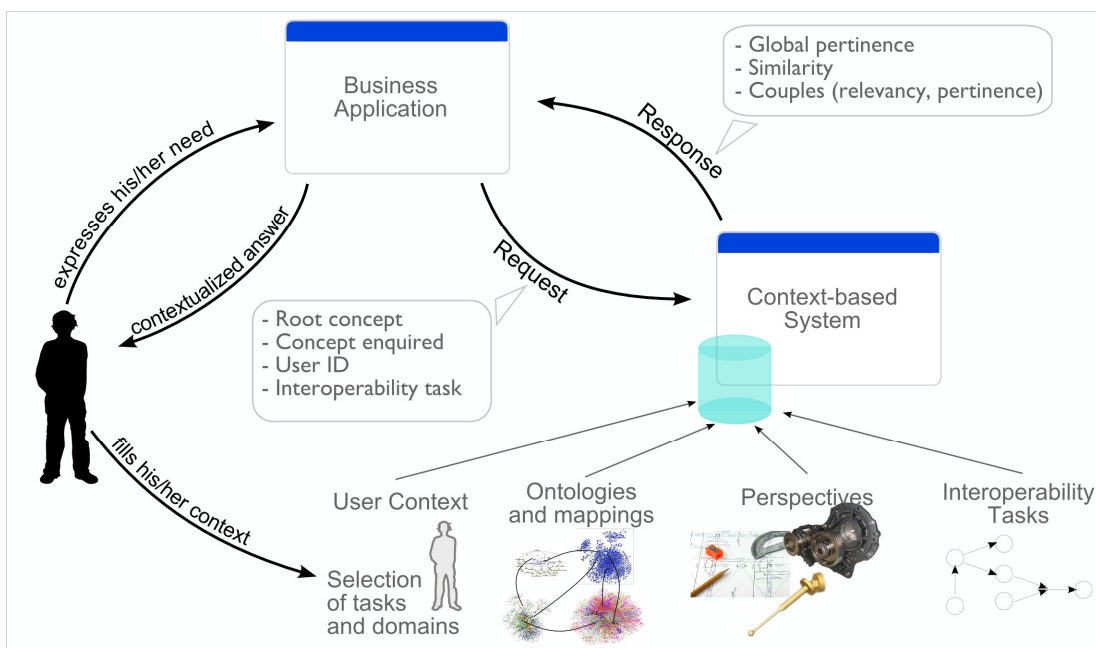


Figure 30: Illustration of the context-based approach

17.2 UML Use cases

The following UML use case diagram Figure 31 shows the various interactions that agents and experts have with the context-based system. We distinguish here two types of experts: experts that manage the data and contextual data in the company, and experts that work in collaboration with experts from the partner organization, to choose which mappings between ontology entities to import, and to relate the perspectives defined by the two organizations.

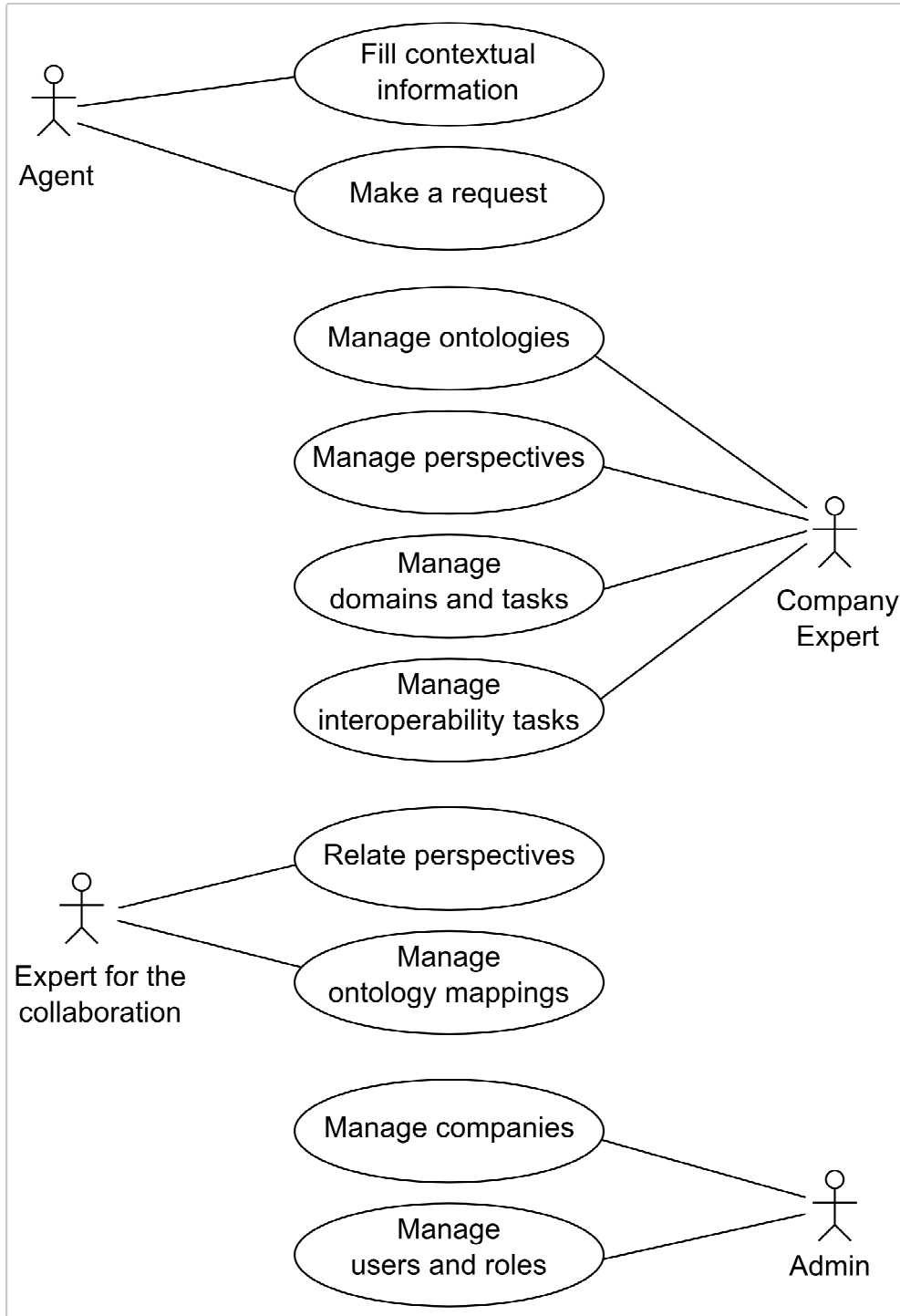


Figure 31: Use cases for agent and expert, and administrator

17.3 Dependency diagram

The following dependency diagram Figure 32 shows the dependencies between the various processes. The diagram depicts four main stages, from the left to the right: the first stage consists in the preparation of the data and contextual data specific to the company, including the import of ontologies, domains and tasks, and interoperability tasks into the system. Experts then have to prepare perspectives and relate them to the domains and tasks, associating them with a value of relevance.

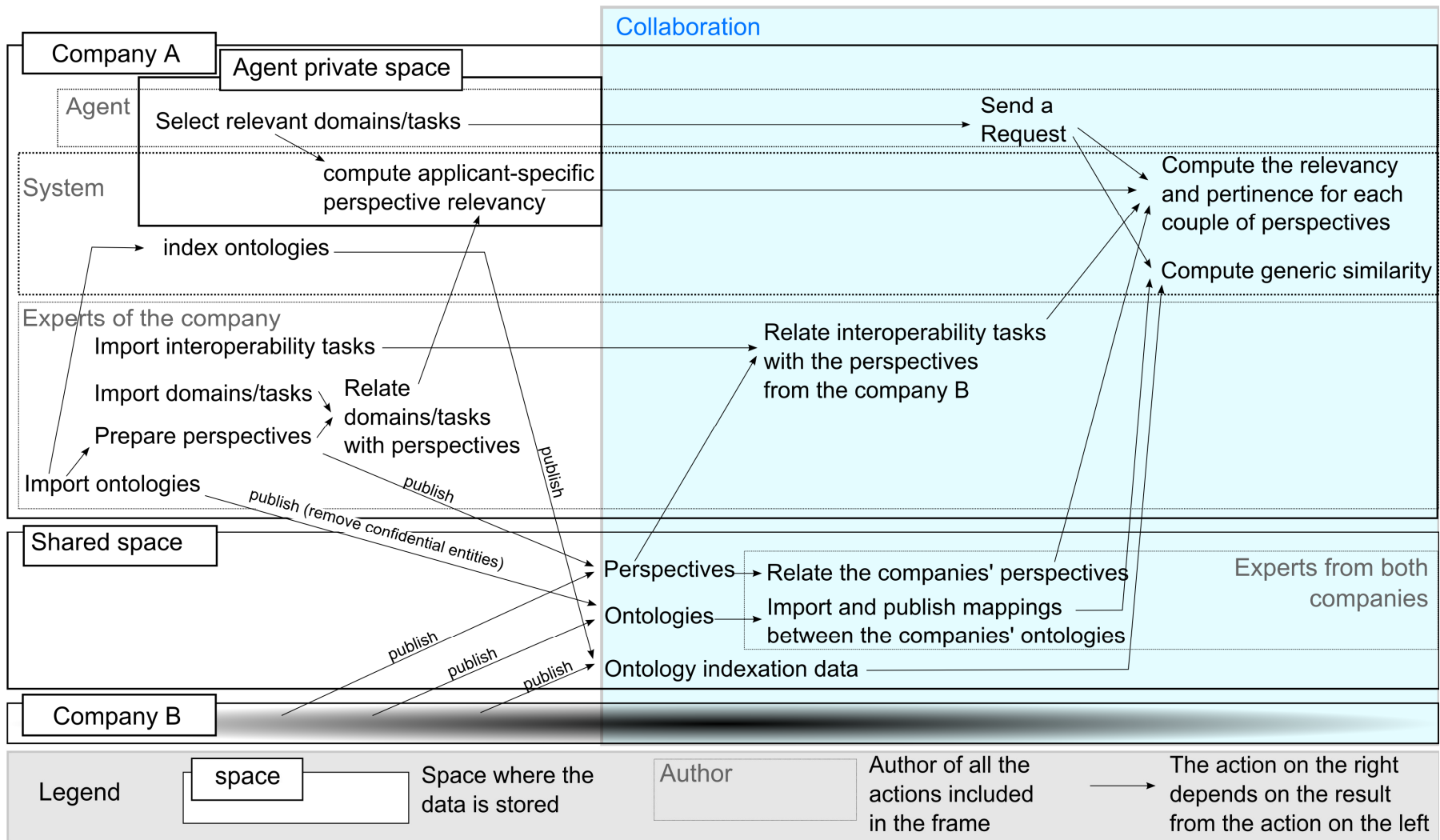


Figure 32: Diagram of the dependencies

The second stage consists in the agents' selection of their context among the domains and tasks imported by the experts. The system uses this information along with the relevancy associated with each triple (domain, task, perspective) to compute an applicant-specific value of relevancy for each perspective of the company.

The third stage consists in the preparation of a new collaboration with a new partner organization, and begins with the publishing in a shared space of the ontologies (with their indexing data computed by the system) and perspectives of the organizations. Then, experts from both organisations use external ontology matching tools to generate mappings between the organizations ontologies, verify these mappings, and import them into the system. They relate also the perspectives of both organizations together. Depending on the needs for collaboration, experts relate also the interoperability tasks of their own company to the perspective defined by the partner organization, and associated with them a value of pertinence.

The last stage, in the context of the collaboration, is the request by agents on two concepts, one from an ontology of the partner organisation and the other from an ontology defined by their own company, along with an interoperability task of their own company. The system then generates a generic value of similarity between the concepts, and as many couples of relevancy and pertinence as there are couples of perspectives corresponding to the concepts that are compatible with one another, whose first perspective is relevant according to the agent's domains and tasks, and whose second perspective is pertinent according to the interoperability tasks selected.

The diagram displays the different access rights to the data, and the authors of the process, among agent, expert, expert for the collaboration, and system.

17.4 Architecture of the system

The Figure 33 gives an example of programming interface (with simplifies typology). It assembles the various actions possible according to the author (agent, company expert, expert for the collaboration) and divides the computations done by the system into context-based computations and others.

Having overviewed the processes of the system, we will show in the next chapter an application example that will go on for the two following chapters.

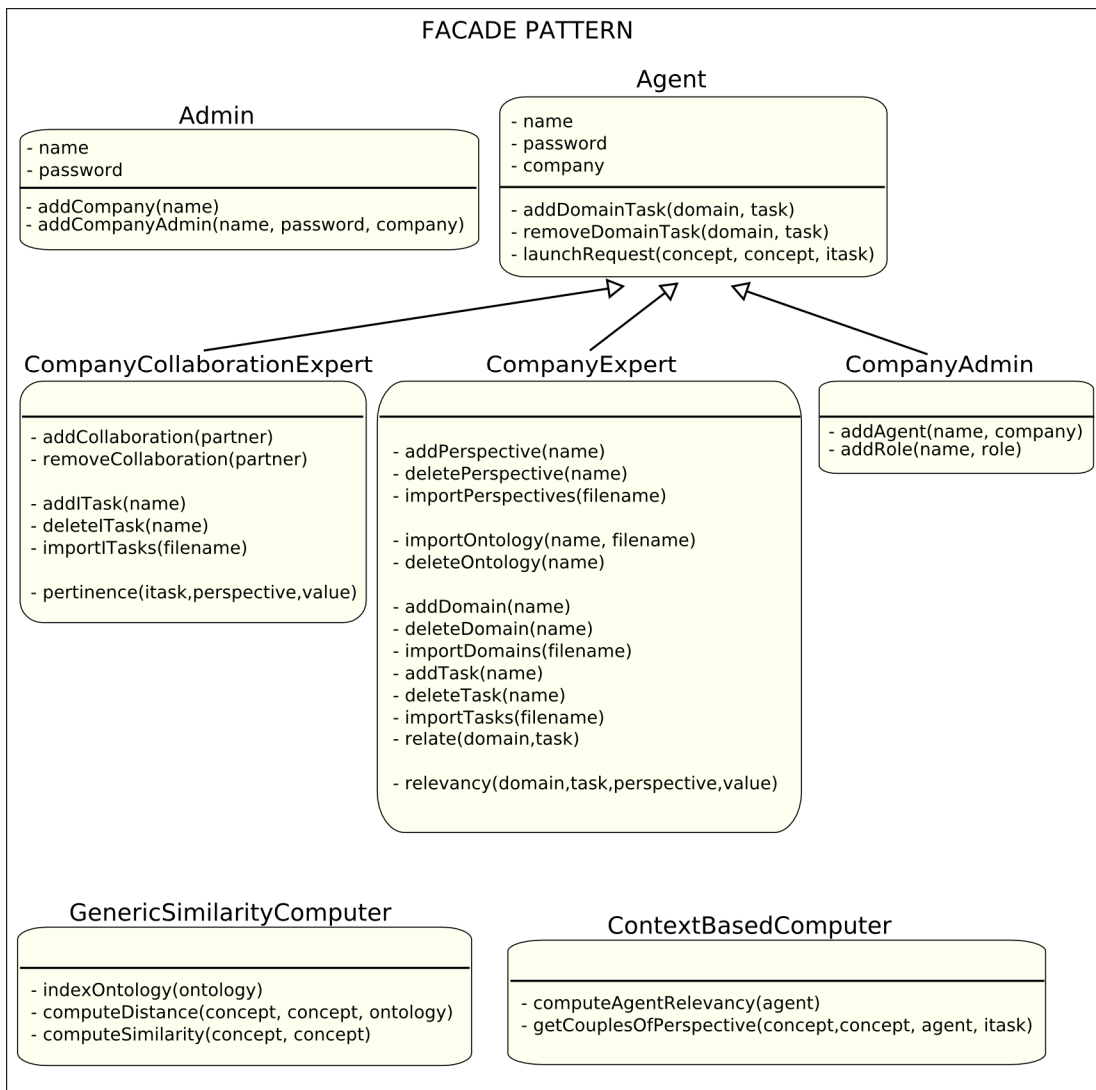


Figure 33: Simplified example of API according to the Façade pattern

Chapter 18: Interaction of an Agent with a Business Application that enquires our System

In this chapter we will describe an example of application to illustrate our approach. We will show how a user may interact with a context-based system based on our approach in an indirect way, through a business application that uses our system as a service for evaluating the pertinence of concepts.

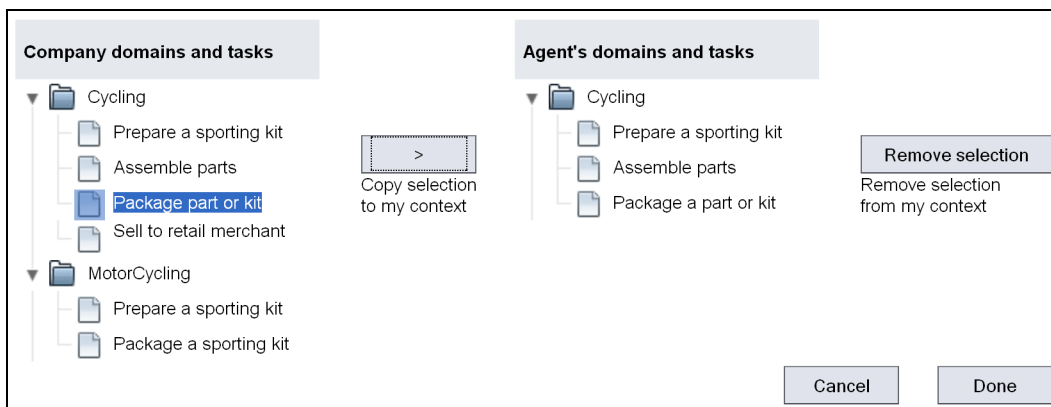
18.1 Interaction with a business application

Marie Dupont is an assembly engineer in the company Imano. The company Imano is specialised in the preparation of kits, their assembly, and packaging of sport equipment, for suppliers of sport equipment. We will consider the case of two of these suppliers, the retail merchants TenThlons and MySport. Marie manages a family of products for cycling, and prepares all information for the assembly or preparation of kits of bicycle parts.

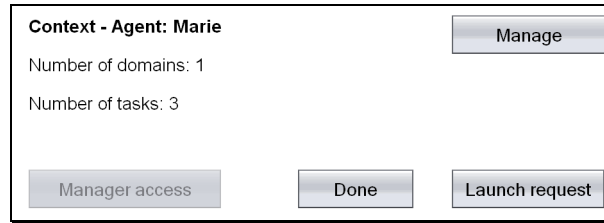
Description of the agent context

For this purpose, she interacts with an integrated design software application connected to our context-based system, which is based on a server online. Marie has never used the context-based system yet, she therefore needs to fill her profile with her contextual information, consisting of the company domains and tasks that she is involved in. This is necessary for the system to know the perspective that she has on the concepts of her company ontologies. She logs into the system with the login and password given to her by the administrator.

She fills her context by selecting the company domains and tasks that she is responsible for: the domain is “Cycling”, and the tasks “Prepare a sporting kit”, “Assemble parts”, and “Package a part or kit” (Screenshot 1). She then confirms and reaches her personal page that provides statistics about her personal context (Screenshot 2).



Screenshot 1: Marie selects the domains that she is involved in



Screenshot 2: Marie's summary page is updated

18.2 A need for interoperability

She is currently involved in the preparation of a new front wheel assembly for racing bikes that should be a performance/value leader, to answer a demand from the retailer TenThlons. A wheel is composed of a rim, a valve, a front hub (the central assembly of the front wheel, consisting of an axle, bearings and a hub shell), of spokes that relate the hub to the rim, of a tire, and so on (Figure 34). She has already selected the rim, the front hub and the spokes.

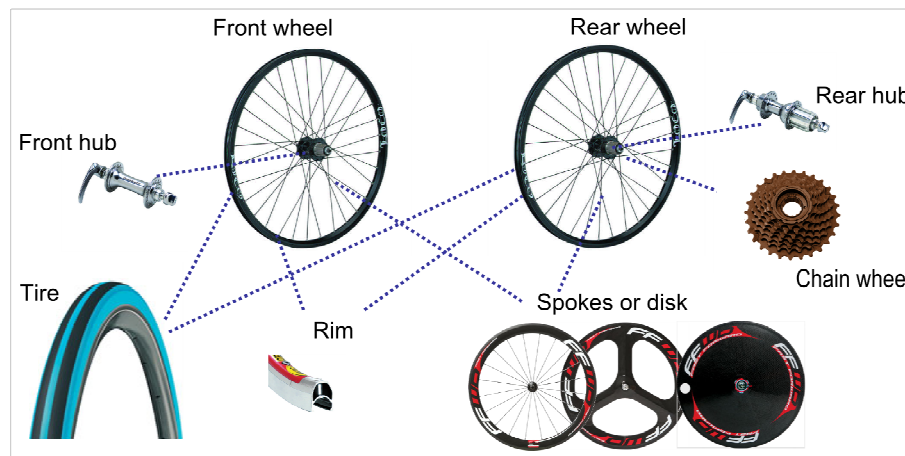


Figure 34: Wheel composition

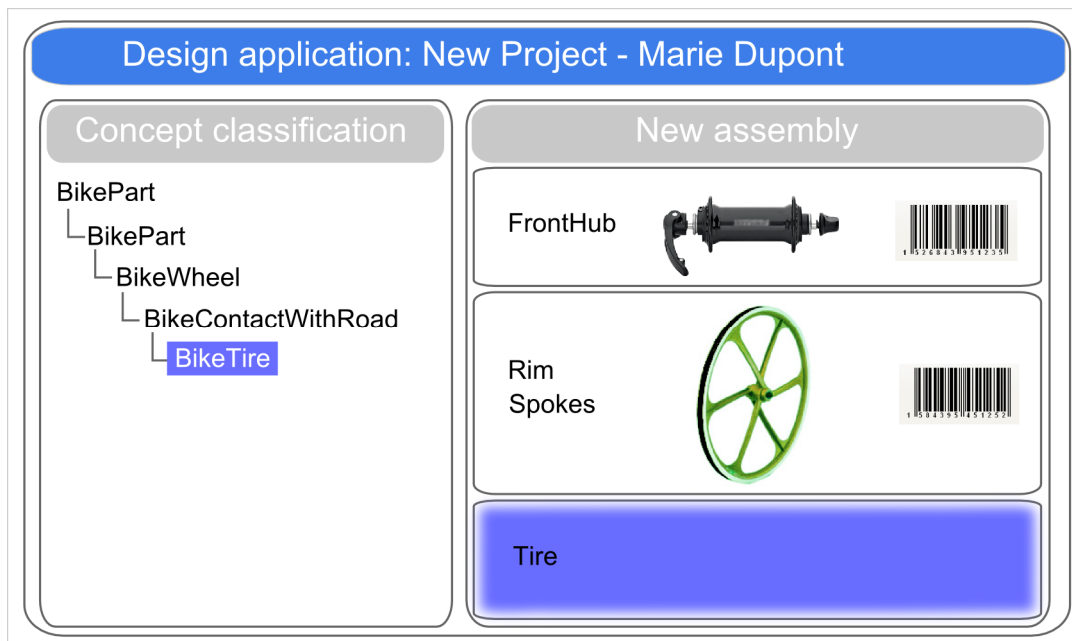
Now, she wants to select an appropriate tire to give to the wheel the qualities of a road bike apt for racing beginners, for an advantageous price. She selects the concept “tire” in the design application (Screenshot 3), and chooses in the menu “find appropriate part”.

18.3 The request

The design application, with information about Marie's context, prepares a suite of requests to the context-based system. The request is made of the four following elements: the agent on the account of whom the request is made, the root concept that describes the kind of part that is being searched for, the interoperability task that describes how the result of the search is to be used, and the concept enquired that is evaluated as a possible pertinent: possible concepts from other organizations, that is to be evaluated.

From the data already recorded with the new project, the design application retrieves the appropriate interoperability task: the new assembly is to be included in a range of products, which are defined according specific criteria of quality of assembly and of distribution. This assembly is designed for a quality production and a limited

distribution. Only parts that may be bought in limited amount should therefore be considered.



Screenshot 3: Marie prepares a new kit within the design application

The company Imano has existing collaborations with two competitors, for the supply of tires of cycles and motorcycles: FireStorm and RicheLin. The design application relies on mappings between the Imano ontology and the ontologies of the cited organizations to find concepts that are likely to correspond to the need expressed by the concept “BikeTire”. It is indeed necessary to make a first selection of possible concepts inquired, as these companies produce all kinds of tires: for trucks, cars, vans, motorcycles, bicycles, and so on. Additionally, Richelin produces also tires for construction site vehicles, bus and coach tires, as well as inner tubes.

The requests are made on the account of the user, with the following information:

Agent: **Marie**

Root concept: **BikeTire**

Interoperability task: **Quality Assembly for Medium distribution**

Concept enquired: **Richelin RacingBicycle_Tire**

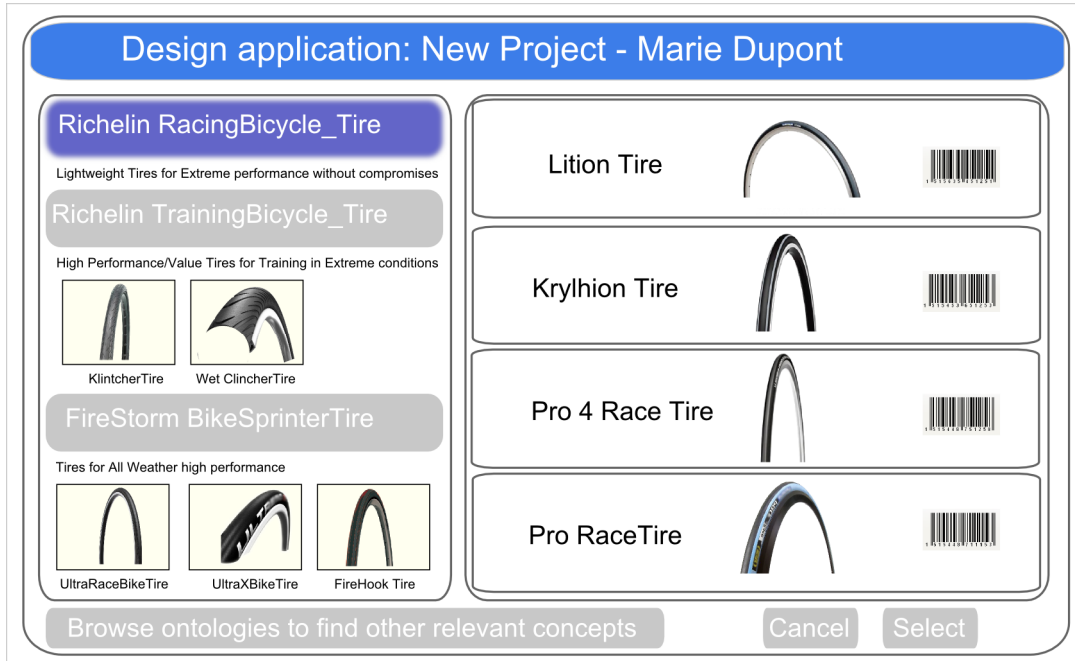
The business application repeats the request, changing only the concept enquired:

- **Richelin TouringBicycleTire**
- **Richelin TrainingBicycleTire**
- **FireStone RoadBicycleTire**
- **FireStone SprinterTire**

18.4 The business application returns contextualized results

The business application creates a screen form in which the various concepts are presented in order of relevancy (Screenshot 4). For each concept, the textual definition in the ontology is given. The data corresponding to the concepts is shown in small

pictures below the concepts, and detailed when the corresponding concept is selected. In the case where the concepts selected are not relevant, a possibility is given to browse the supplier ontologies to find directly relevant concepts. Marie can therefore select the tires that are the more appropriate for the wheel assembly that she is preparing.



Screenshot 4: The design application shows the data associated with the most relevant concepts

18.5 Conclusion

We have described an example of application of our approach, showing how a business application can make requests to the context-based system, in a transparent way for the user. The results returned by the context-based system provide the design application with an evaluation of the concepts pertinence that the application can use to rank the concepts in an order of decreasing pertinence.

We will describe in the two following chapters the preliminary preparation of data and contextual data that would have to be done ahead of time by the collaborating companies for this scenario to be possible.

Chapter 19: An Expert Models the Company Information into the System

In this chapter, we continue the example, showing preliminary work that must be done so that Marie could get the results presented in the previous chapter. We will present the input of the data and contextual data specific to the company before any collaboration occurs, and that may be reused for all collaborations.

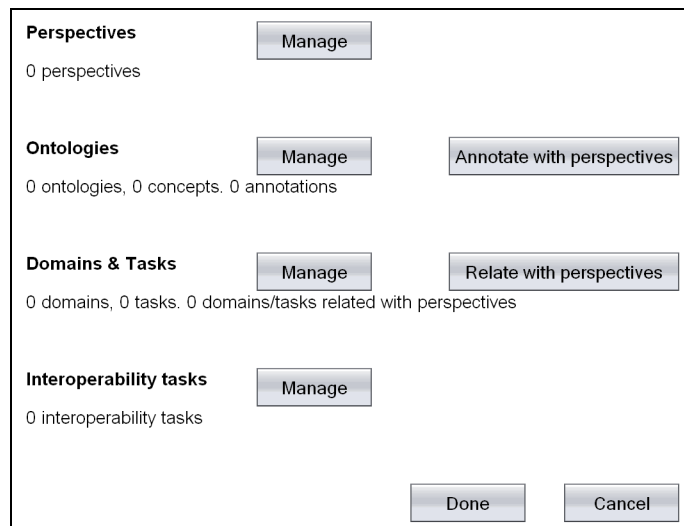
19.1 The expert gathers the ontologies developed

We will follow the operations of an expert working for the company, Pierre Schmidt. He logs into the system. As a company expert, he has access to the management of the system, from his user page (Screenshot 5).



Screenshot 5: Pierre Schmidt's user page, with manager access

Pierre Schmidt clicks the button “Manager access”, and gets the manager page, that summarizes the data and contextual data already stored for his company (Screenshot 6).



Screenshot 6: The manager access to a summary page for the company

The company has developed three ontologies, presented Figure 35: The “bike and motorbike parts”, that describe the parts of bicycles and motorcycles that the company is interested in, for assembly or for the preparation of kits. The ontology “RollerBlade parts” does the same for roller blades. These ontologies classify the types of parts

obtained from suppliers. For example, bicycle components are described according to their function: contact with the road (tire, rim), rotation centre (hub), and connection between them (spokes or disk).

The “Kits classification” describes the kits prepared by the company. This latter ontology defines kits by the means of semantic relations *part-of* that relate the kits to the components that constitute it, mainly constituted of parts defined in the two other ontologies. These links are not shown here, but only the *is-a* hierarchy.

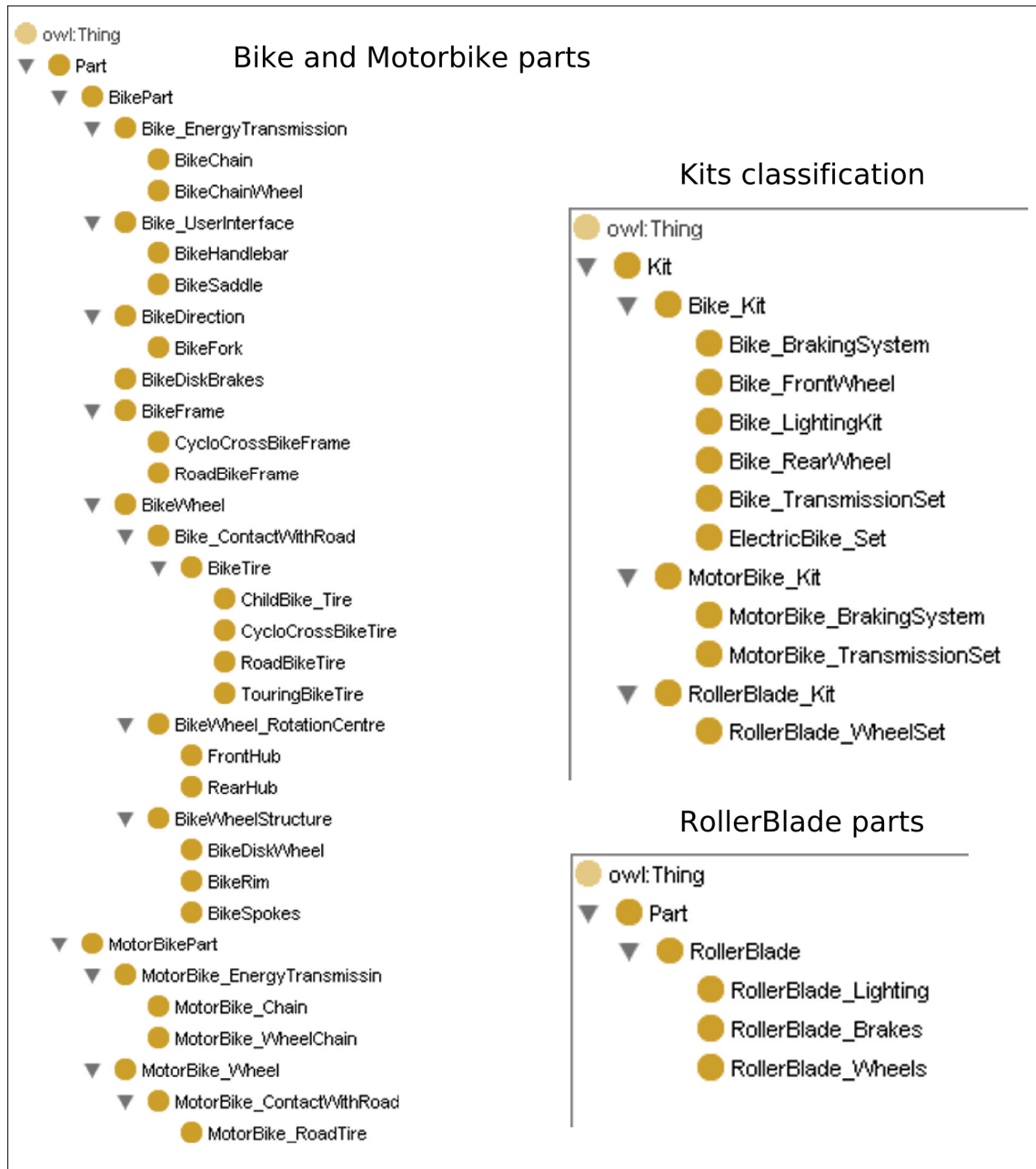
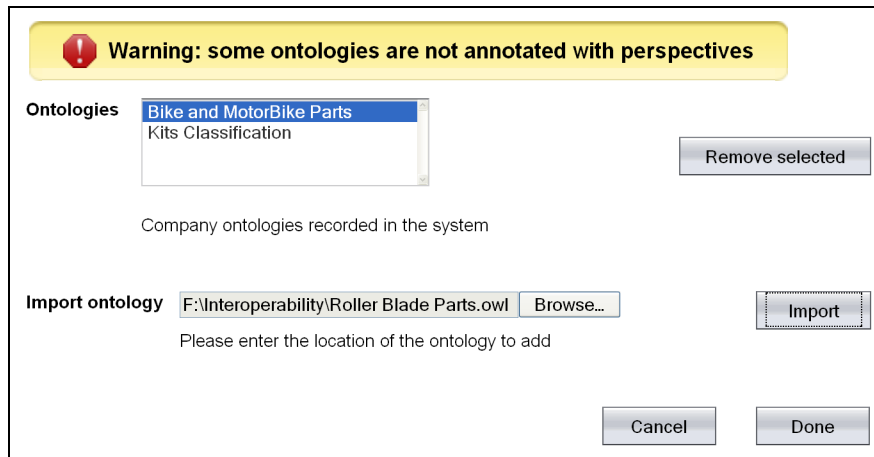


Figure 35: Ontologies developed by the company Imano

Pierre Schmidt imports the ontologies of the company into the system (Screenshot 7). The system reads the ontologies, records the concepts, and indexes them according to

their position in the *is-a* hierarchy. The system stores on the server a version of the ontologies from different companies, suppliers and others, with which the company Imano has collaborations.

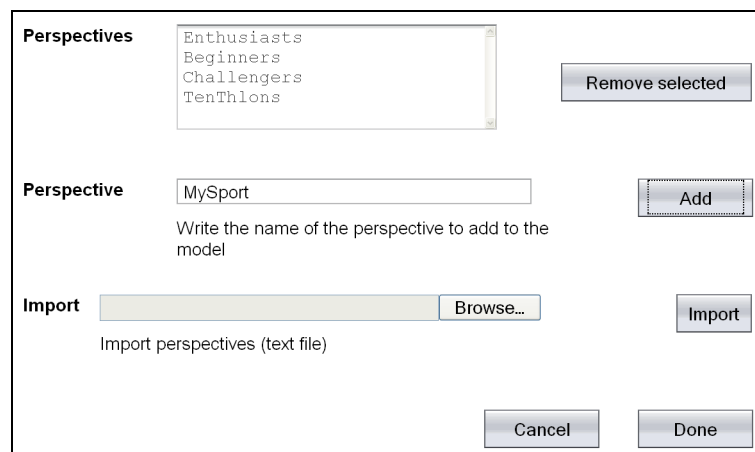


Screenshot 7: The expert imports ontologies into the system

19.2 The expert models perspectives

The company Imano follows marketing strategies that determine the position of the products in the market: some products assembled by the company are designed for enthusiasts, beginners; others are middle-range products that propose excellent performance for a performance/value quality that is though to beat. The following perspectives proceed from these strategies:

- Enthusiasts,
- Beginners,
- Challengers



Screenshot 8: The expert adds the appropriate perspective to the model

The company Imano distributes its products through two retail companies that are interested in different products. Therefore the experts have decided to represent these using perspectives:

- TenThlon,
- MySport

Pierre Schmidt records all these perspectives into the system (Screenshot 8).

The system includes a keyword-based search for the annotation of concepts (Screenshot 9). When a row is highlighted in the table, the list of the concepts subsumed by the concept is shown below, so as to see whether the concepts subsumed should be included along with the concept. There is also the possibility to include the concepts subsumed, but not the concept highlighted. Here we see that the concept “BikeTire” imported by the ontology “Kits classification” subsumes the four concepts “RoadBike Tire”, “CycloCrossBike Tire”, “TouringBike Tire”, and “ChildBike Tire”.

Concept

Search by keyword in all the ontologies of the company

Concepts that contain this keyword in the ontologies of the company

Ontology	Local name	baseURL	Select	include concepts subsumed
Bike and MotorBike Parts	BikeTire	http://www.imano.com/parts.owl#	<input type="checkbox"/>	<input type="checkbox"/>
Bike and MotorBike Parts	MotorBike RoadTire	http://www.imano.com/parts.owl#	<input type="checkbox"/>	<input type="checkbox"/>
Bike and MotorBike Parts	RoadBike Tire	http://www.imano.com/parts.owl#	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Bike and MotorBike Parts	CycloCrossBike Tire	http://www.imano.com/parts.owl#	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Kits Classification	BikeTire	http://www.imano.com/parts.owl#	<input type="checkbox"/>	<input type="checkbox"/>

Concepts subsumed by the concept highlighted

- RoadBike Tire
- CycloCrossBike Tire
- TouringBike Tire
- ChildBike Tire

Perspective

Perspective with which to annotate the concepts selected

Annotate the concepts selected

Screenshot 9: the expert annotates concepts with perspectives

Pierre Schmidt selects the concepts that are pertinent for the various perspectives. When he thinks he has finished the annotation of concepts with the perspectives present in the model, Pierre checks quickly the list of concepts annotated to make sure he has not forgotten anything (Screenshot 10). The ontologies annotated are presented in the Figure 36.

Concepts annotated with the various perspectives

Ontology	local name	Perspectives	Remove
Bike and MotorBike Parts	BikeFork	Enthusiasts, TenThlons	<input type="button" value="Remove"/>
Kits Classification	Bike TransmissionSet	Enthusiasts, TenThlons	<input type="button" value="Remove"/>
Bike and MotorBike Parts	Bike Saddle	Enthusiasts, TenThlons	<input type="button" value="Remove"/>
Kits Classification	Bike FrontWheel	Enthusiasts, Beginners, Challengers, TenThlons, MySport	<input type="button" value="Remove"/>
Kits Classification	Bike RearWheel	Enthusiasts, Challengers, TenThlons, MySport	<input type="button" value="Remove"/>

Screenshot 10: the expert checks that the concepts are annotated with the right perspectives

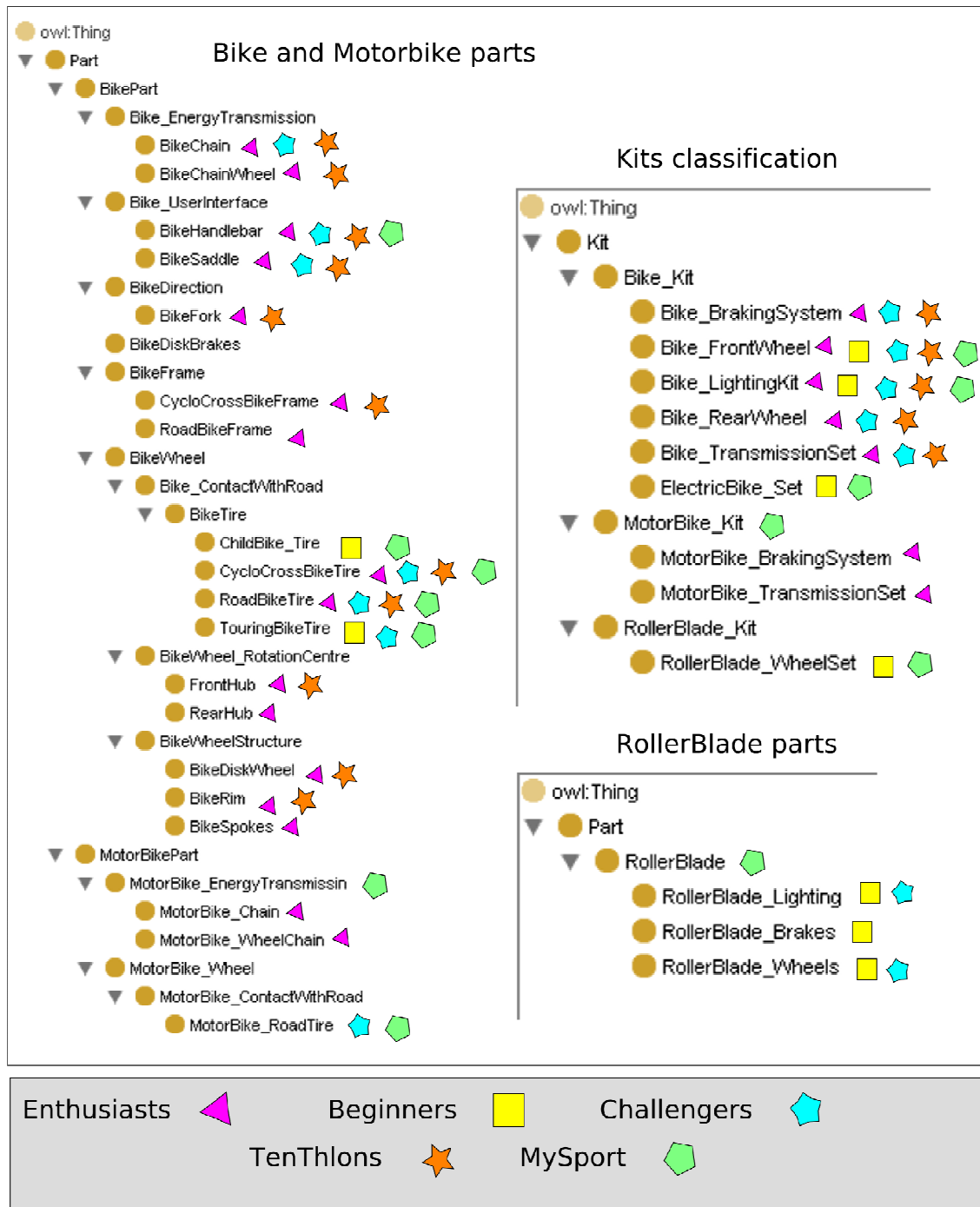


Figure 36: Ontologies annotated with perspectives

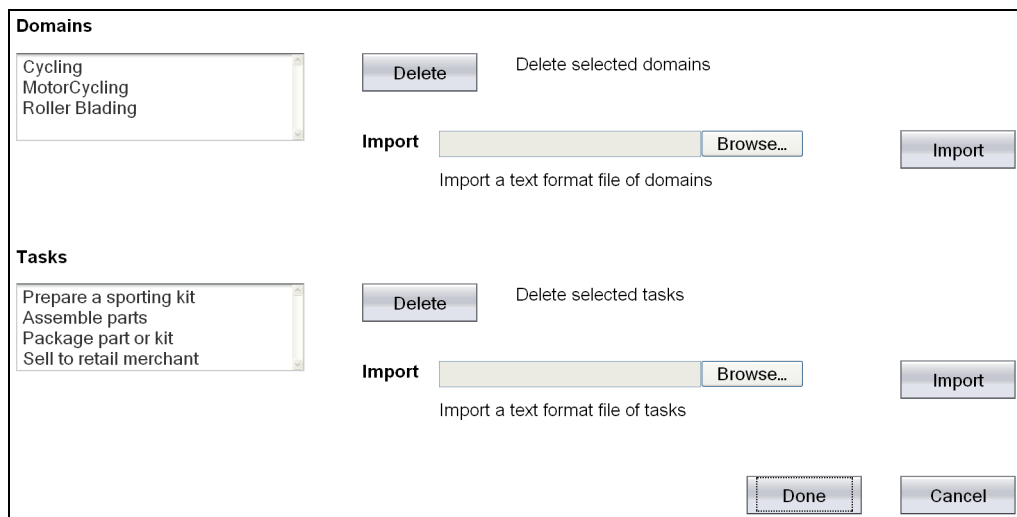
19.3 The expert models domains and tasks

Pierre Schmidt continues to prepare the data specific to the company by importing domains and tasks from existing taxonomies of domains and tasks (Screenshot 11). The company assembles parts and prepares kits for the following sports:

- roller blade,
- cycling,
- motor bike.

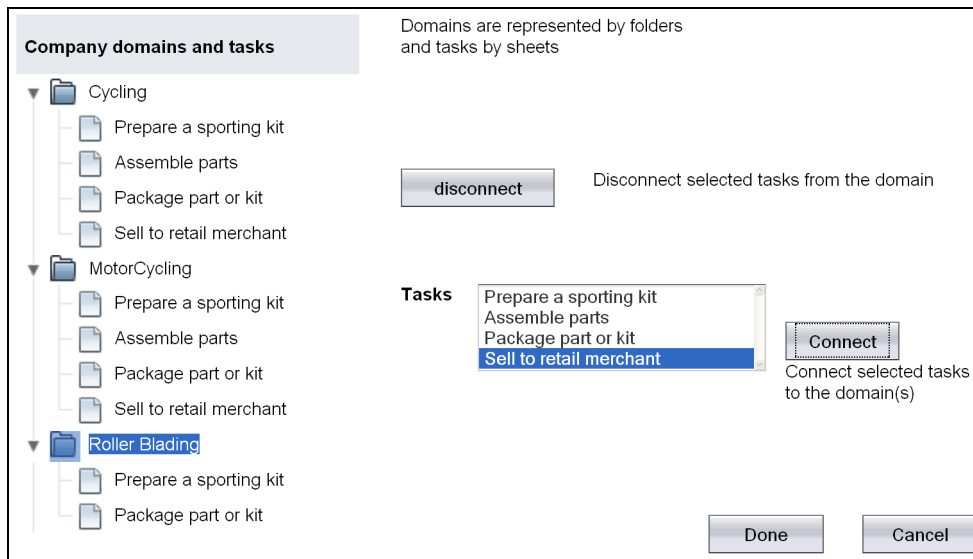
The company has enumerated the following tasks that:

- prepare a sporting kit,
- assemble parts together,
- package parts or kits,
- sell parts and kits to retail merchants



Screenshot 11: The expert imports domains and tasks into the system

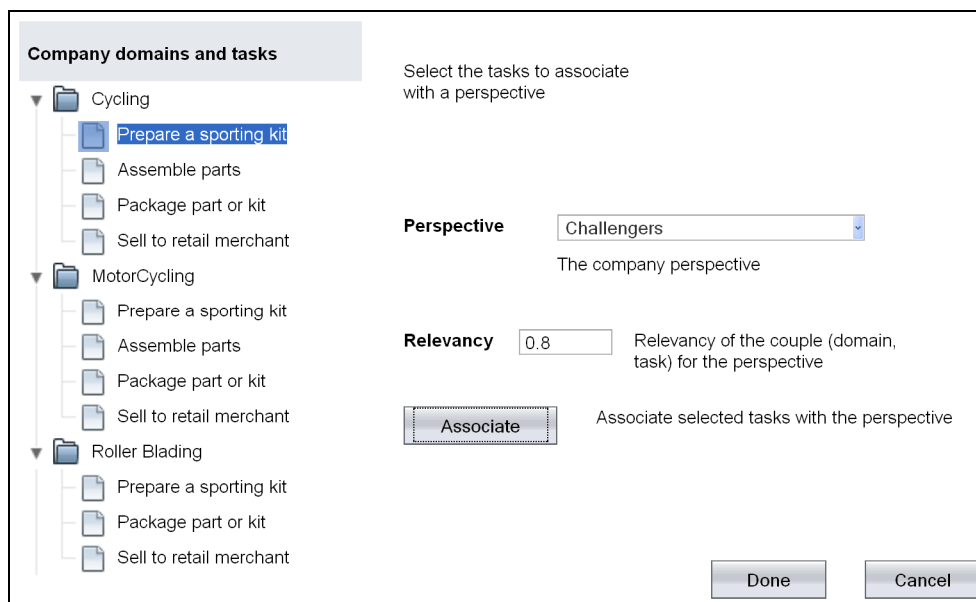
The domains and tasks imported, Pierre connects them together: a domain may be associated with various tasks, and a task to various domains (Screenshot 12).



Screenshot 12: the expert connects domains and tasks

19.4 The expert connects domains and tasks with perspectives

The interface for the connection between domains/tasks and perspectives resembles the previous one: Pierre has to select tasks in the tree on the left (Screenshot 13), a perspective, and to associate them with a value of relevancy. As the task selected in the tree corresponds to a particular domain (under which “folder” it is placed), it is indeed the couple (domain, task) that is related with the perspective.



Screenshot 13: the expert relates domains/tasks with perspectives

When this is done, Pierre checks that everything is fine (Screenshot 14). The Table 20 gives a possible set of annotation of triples (domain, task, perspective) with a value of relevancy. For example, the domain “Roller Blading” is associated with a relevancy of “0” for the retail merchant TenThlon, who does not sell roller blades. The task “preparation of kit” and the perspective “beginners” are associated with a low

relevancy, as beginners tend to prefer to buy complete bicycles or roller blades. The values associated with the task “selling for retail” indicate some statistics on the items sold according to the different marketing strategies, for the different domains.

Relation between domains, tasks and perspectives					
Domain	Task	Perspective	Relevancy		Remove
Cycling	Sell to retail merchant	Challengers	0.6	Change	Remove
Cycling	Sell to retail merchant	TenThlons	1	Change	Remove
Cycling	Sell to retail merchant	MySport	1	Change	Remove
MotorCycling	Prepare a sporting kit	Enthusiasts	0.35	Change	Remove
MotorCycling	Prepare a sporting kit	Beginners	0.1	Change	Remove

Screenshot 14: the expert checks that the couples (domain, task) are associated with a correct perspective

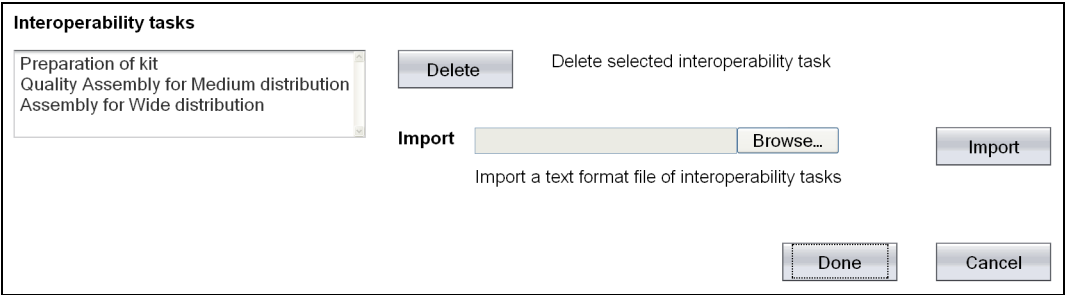
Table 20: Association of relevancy to triples (domain, task, perspective)

Domains	Tasks	Perspective Relevancy				
		Enthusiasts	beginners	challengers	TenThlon	MySport
Cycling	preparation of kit	0.6	0.2	0.8	0.8	0.2
	Assembly	0.4	0.9	0.6	0.6	0.4
	Packaging of part or kit	0.9	0.1	0.2	0.7	0.3
	Selling for retail	0.7	0.4	0.6	1	1
Motorcycling	preparation of kits	0.3	0.1	0.3	0.2	0.3
	Assembly	0.6	0.9	0.6	0.3	0.3
	Packaging of part or kit	0.1	0.1	0.2	0	0.1
	Selling for retail	0.2	0.6	0.8	0.3	0.4
Roller blading	preparation of kits	0.1	0.3	0.6	0	0.5
	Assembly	0.1	0.8	0.5	0	0.8
	Packaging of part	0	0.1	0.5	0	0.2
	Selling for retail	0	0.9	0.7	0	0.8

19.5 The expert models interoperability tasks

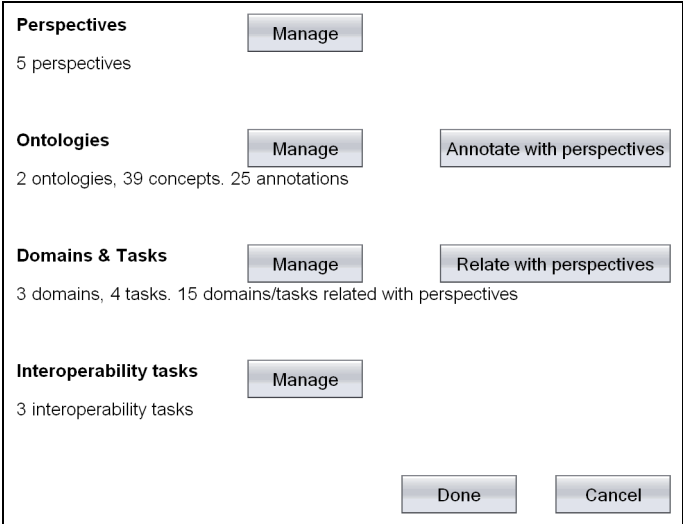
The last thing that Pierre Schmidt has to do is to import the interoperability tasks into the system (Screenshot 15). He has recognized three interoperability tasks: “Preparation of kit”, which is identical to the corresponding task. The company has two robotised assembly line: the first is completely robotised, and is used for assembly of parts for a large distribution, usually with cheaper parts; the other with possibly manual operations, and manual inspection, for high quality assembly. Pierre chooses to associate a

particular interoperability task to these chains, as notably the conditions of supply are distinct.



Screenshot 15: The expert imports the interoperability tasks

The manager page shows a summary of the operations (Screenshot 16).



Screenshot 16: Summary of the operations done

19.6 Conclusion

We have presented an example of the operations done by a company to present the contextual data, necessary for further collaborations. The first step consists in developing and importing ontologies in the system. The perspectives have then to be recognized, and the ontology concepts annotated with them. Finally, the company domains and tasks are imported into the system, and compared with the perspectives recorded earlier.

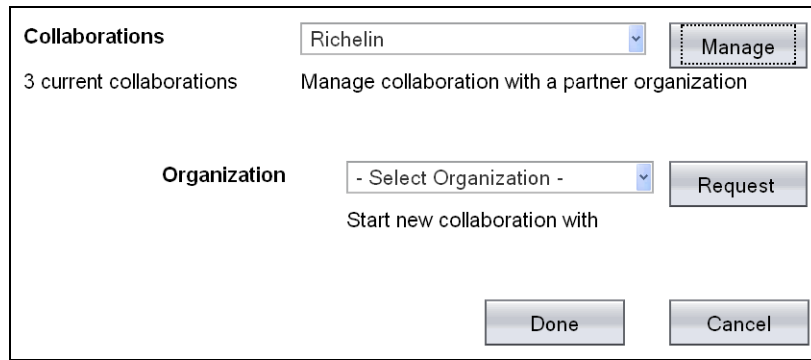
In the next chapter, we will follow the example, and describe how the collaboration with the organization Richelin had been prepared.

Chapter 20: An Expert Prepares the System for a new Collaboration

In this chapter, we will show all the preliminary phases that had to be done for the collaboration between the organizations Imano and Richelin to happen. We will see how the ontologies of the organizations have to be related by mappings, as well as the organizations' perspectives. Finally the interoperability tasks of the company Imano have to be related with the perspective of the Richelin organization.

20.1 The expert manages ontology mappings

Sylvie Durand has the role of expert for the collaboration in the company Imano. In the manager page, she requests a collaboration with the organization Richelin. After the approval of this choice organization, she can access to the page specific to this collaboration (Screenshot 17, Screenshot 18).



Screenshot 17: The expert chooses to manage the collaboration with the organization Richelin



Screenshot 18: The page for the collaboration

The organizations Richelin and FireStorm also have ontologies that describe their products. The product ontology defined by the organization Richelin is shown in the figure below. Sylvie utilises ontology matching tools to find possible mappings between

the ontologies of the company Imano and the product ontology of the organisation Richelin (Figure 37).

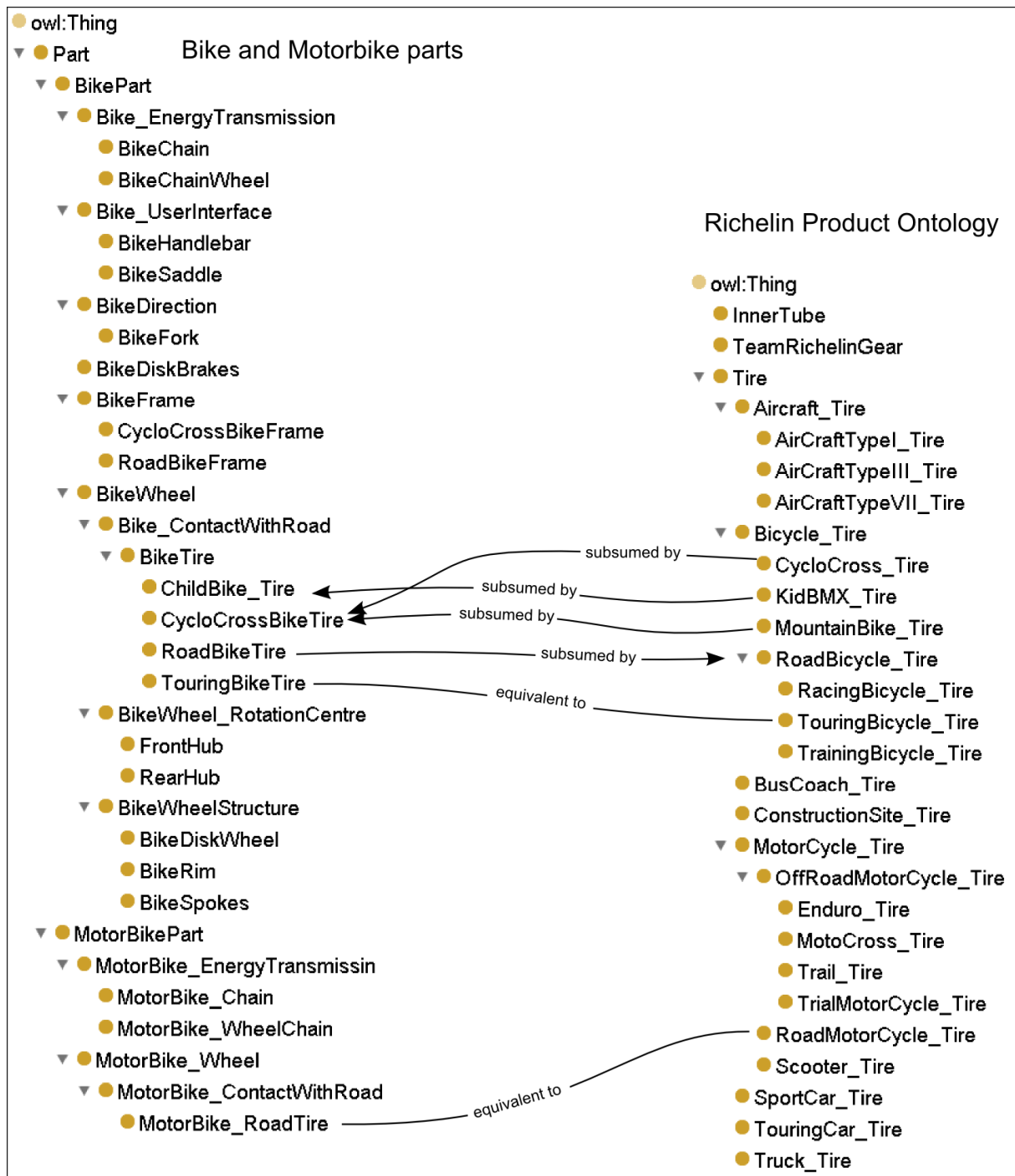
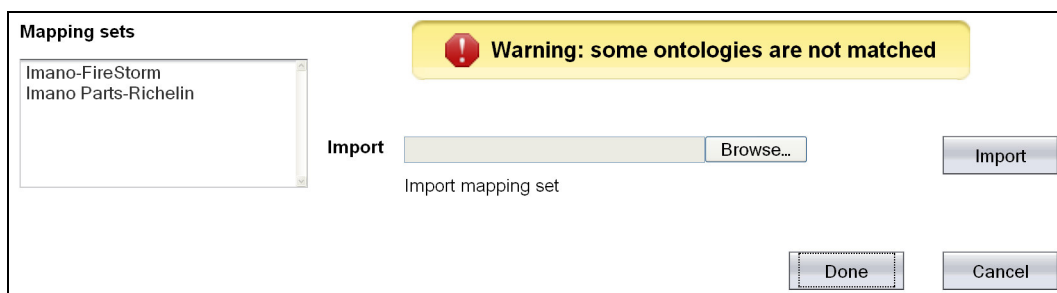


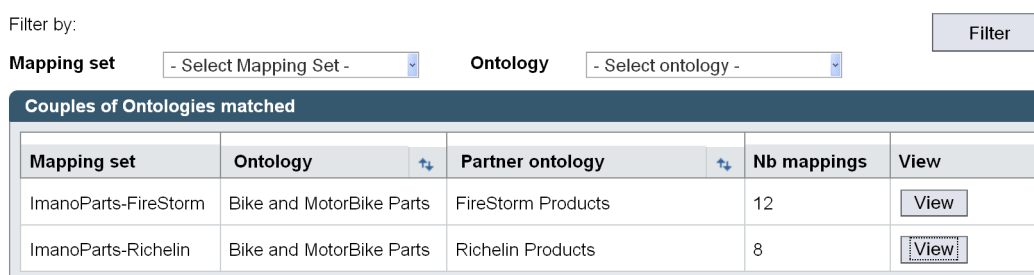
Figure 37: mappings between two ontologies of Imano and Richelin

After validating the mappings with Richelin experts, she imports them in the context-based system (Screenshot 19). When the mappings are imported into the system, bidirectional mappings of equivalence are converted into two directional mappings.



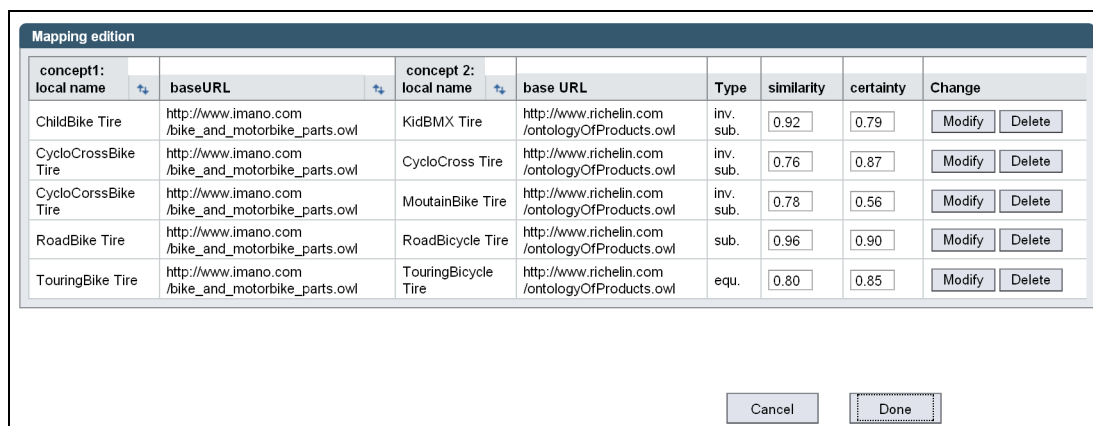
Screenshot 19: The expert imports mappings into the system

Once the mapping sets are imported into the system, Sylvie visualizes the statistics about the mappings between the ontologies of Imano and of Richelin (Screenshot 20). She selects the second row, corresponding to the mappings with the organization Richelin, and clicks the button “View”.



Screenshot 20: The expert visualizes the statistics about mappings

Sylvie Durand checks quickly the mappings (Screenshot 21).



Screenshot 21: The expert edits a few mappings

20.2 The expert relates the perspectives of both organizations

Tires have various characteristics: dimensions differ; some are tubeless while others require an inner tube; some are excellent in dry conditions but lose their grip when the weather gets wet; others have a light weight, are comfortable to ride, are particularly durable, and so forth.

The data associated with the different types of tires is different, for historical reasons, but also because the needs are divergent: aircraft tires require particularly high resistance to compression and extreme changes of temperature that are not required of car tires. But car tires have to endure much longer distance. Cross country bicycles and motorcycle tires have to grip rocky and muddy terrains, while road bikes grip bitumen.

The organization Richelin has perspectives that depend on various categories of vehicle that share similar characteristics: (1) cars, vans, pickups and SUVs, (2) bicycles and motorcycles, (3) trucks and bus. For each of these, they indeed have distinct strategies: for trucks and cars, where competition is strong, perspectives are durability and reduction of rolling resistance, grip under extreme weather circumstances. For heavy earthmovers, it may be the ability to support a heavy weight over a long distance.

The Figure 38 shows the annotation of the product ontology with perspectives that experts of the Richelin organization have applied. These annotations show that if some notions are relevant for all products, such as price, others are specific to a selection of concepts. Not all tires are adapted for all terrain and the style “urban” applies only to a few tires.

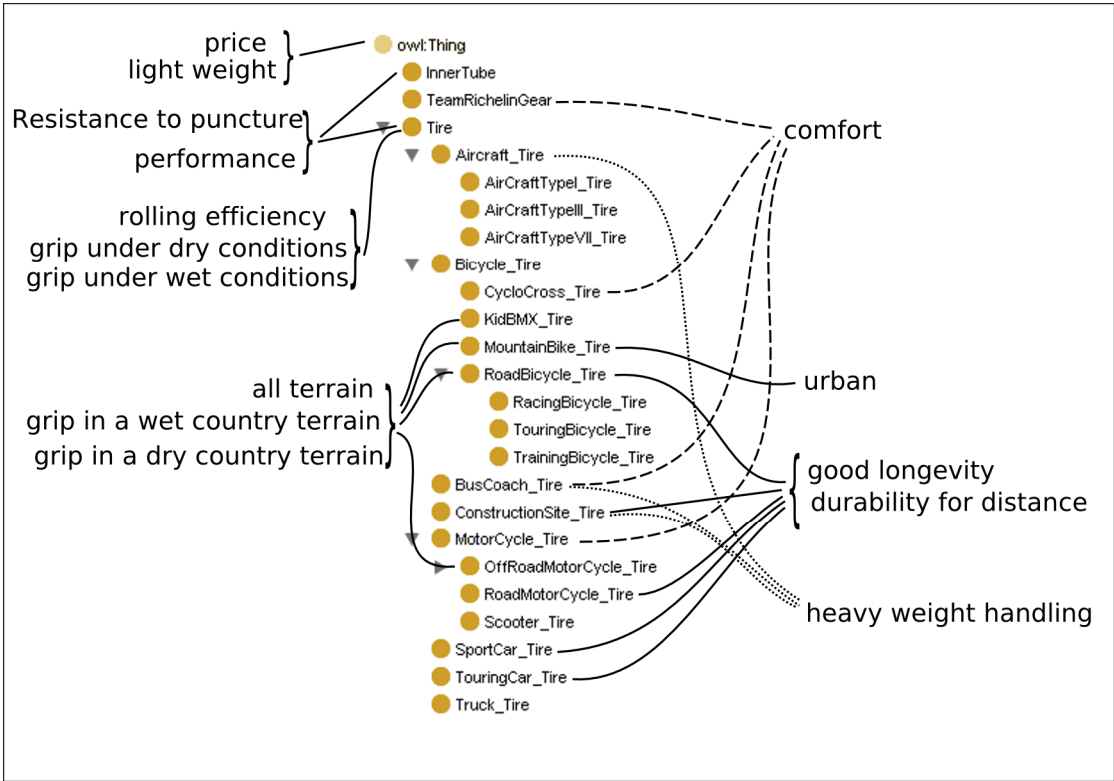


Figure 38: Richelin’s product ontology annotated with perspectives

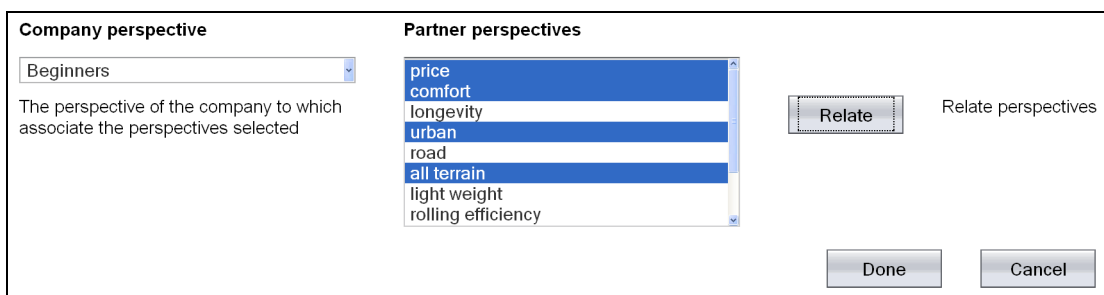
Perspectives include:

- Resistance to puncture,
- light weight,
- rolling efficiency,
- grip under dry conditions
- grip under wet conditions,
- performance,
- durability for distance,
- good longevity
- comfort,
- price
- all terrain
- urban
- road
- grip in a wet country terrain
- grip in a dry country terrain
- heavy weight handling
- ...

Also, the ontology presents all products of the organization. But Richelin manufactures tires in different factories. Some tires are produced in all factories, but some tires are produced only in one factory. Richelin has therefore decided to add perspectives to the name of its factories:

- Charlesruhe
- Clerront-Ferland
- Mehico City

Sylvie has to relate the perspectives from the company Imano with the perspectives of the organisation Richelin (Screenshot 22).



Screenshot 22: The expert relates the perspectives of the two organizations

She then verifies that the perspectives of the two organizations are accurately related (Screenshot 23). The Figure 39 presents the relations that she found between the perspectives. For each of the perspectives of the Imano company, the appropriate perspectives defined by Richelin are placed in a rounded box. The only exception is the last box, which shows Richelin perspectives that are not related at all to any perspective defined by the company Imano.

Relations between the perspectives of the company and the Partner perspectives		
Company Perspective	Partner Perspective	Remove
Beginners	grip under dry conditions	<input type="button" value="Remove"/>
Beginners	all terrain	<input type="button" value="Remove"/>
Beginners	urban	<input type="button" value="Remove"/>
Challengers	rolling efficiency	<input type="button" value="Remove"/>
Challengers	road	<input type="button" value="Remove"/>

Screenshot 23: The expert checks that the perspectives are correctly related

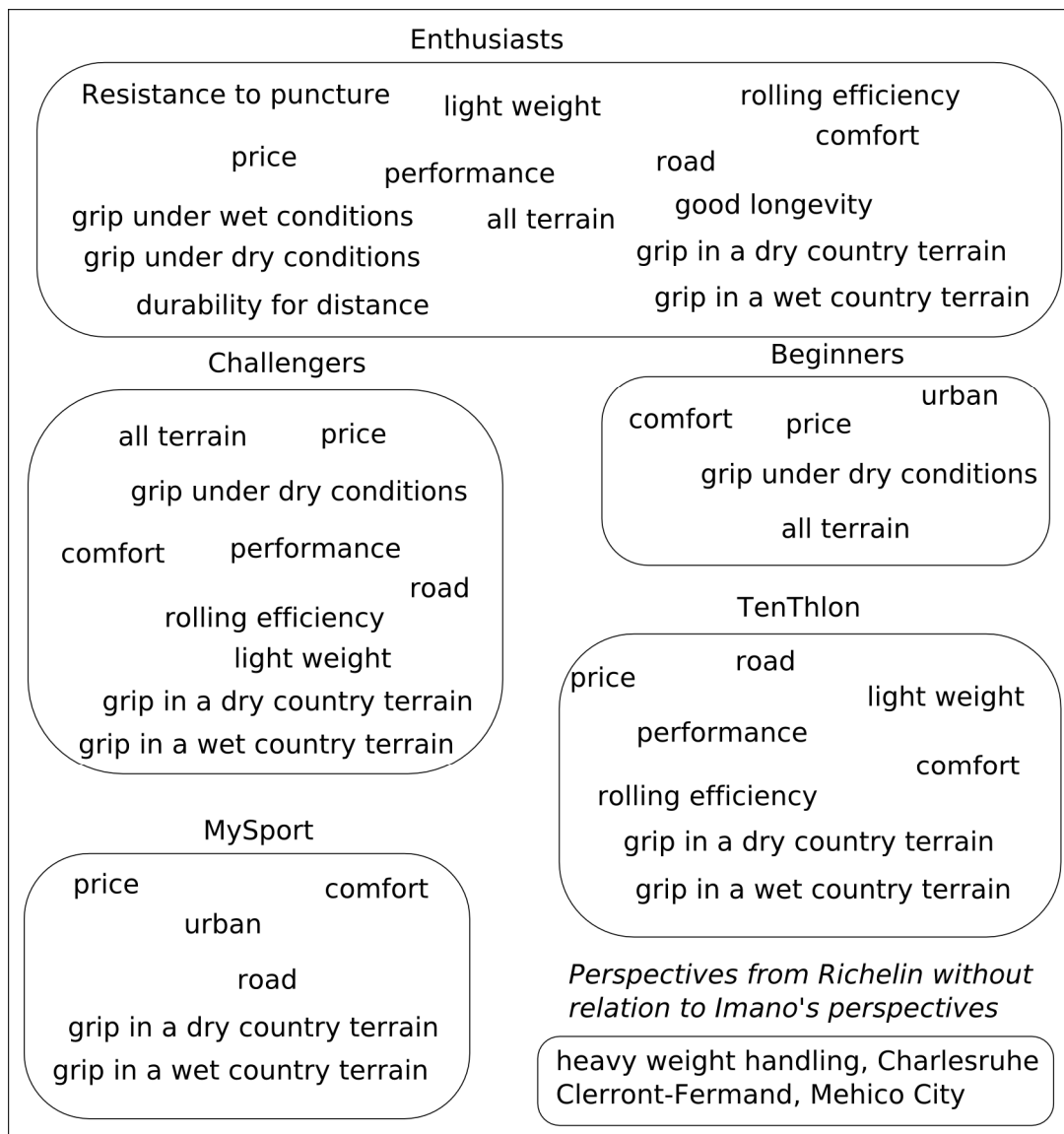


Figure 39: connection between the perspectives of the two organizations

20.3 The expert relates interoperability tasks and perspectives

The interoperability tasks may now be associated by perspectives from Richelin. The factory of Mehico City is far away, and is not adapted to the reactivity needed for

assembly in the case of medium distribution, and the cost of transport is not compensated by a large purchase order. Sylvie therefore associates a pertinence of 0.2 for this interoperability task and this perspective (Screenshot 24).

Screenshot 24: The expert associates interoperability tasks with perspectives

She does the same with the others, and checks the result (Screenshot 25). She associates a high value of pertinence of Clerrond-Ferland for a wide production, as the factory has a big capacity there. She may also reduce the value of pertinence for Mehico City in the case of a medium distribution to 0.1.

Interoperability tasks and perspectives of diverse partners			
Interoperability task	Perspective	Pertinence	Remove
Quality Assembly for medium distribution	Charlesruhe	0.8 <input type="button" value="Modify"/>	<input type="button" value="Remove"/>
Quality Assembly for medium distribution	Clerrond-Ferland	0.8 <input type="button" value="Modify"/>	<input type="button" value="Remove"/>
Quality Assembly for medium distribution	Mehico City	0.1 <input type="button" value="Modify"/>	<input type="button" value="Remove"/>
Assembly for wide distribution	Charlesruhe	0.6 <input type="button" value="Modify"/>	<input type="button" value="Remove"/>
Assembly for wide distribution	Clerrond-Ferland	0.8 <input type="button" value="Modify"/>	<input type="button" value="Remove"/>

Screenshot 25: The expert checks the perspectives associated with each interoperability task

20.4 Conclusion

We have presented in this chapter, the three types of data that had to be prepared for the collaboration. Firstly, mappings were imported, to relate the concepts from the ontologies defined by the organizations. Secondly, the experts related the perspectives. Thirdly, the interoperability task of the company Imano was related with the perspectives of the organization Richelin, to evaluate the practical conditions for interoperability.

In the next chapter, we will see the details of an implementation of the context-based system, including the design of a relational database, and algorithms for the various computations.

Chapter 21: Algorithms and Database schema

In this chapter, we will present the underlying structure of the system, consisting in the schema of the relational database, and the algorithms for the various computations. We will show the underlying structure corresponding to the usages of context, in the order in which we presented them in the third part of this dissertation; then we will do the same for the computation of the generic semantic similarity measure.

21.1 Disambiguation by the comparison of concept perspectives

In this section, we will describe the implementation of the usage of disambiguation by the connection with perspectives. For reasons of simplicity, we model perspectives by a list of identifiers, and we consider that ontologies are uploaded on the system as a list of concepts. The whole data can thus be implemented by a relational database, and the connection between the data can be clearly represented without unnecessary technical details.

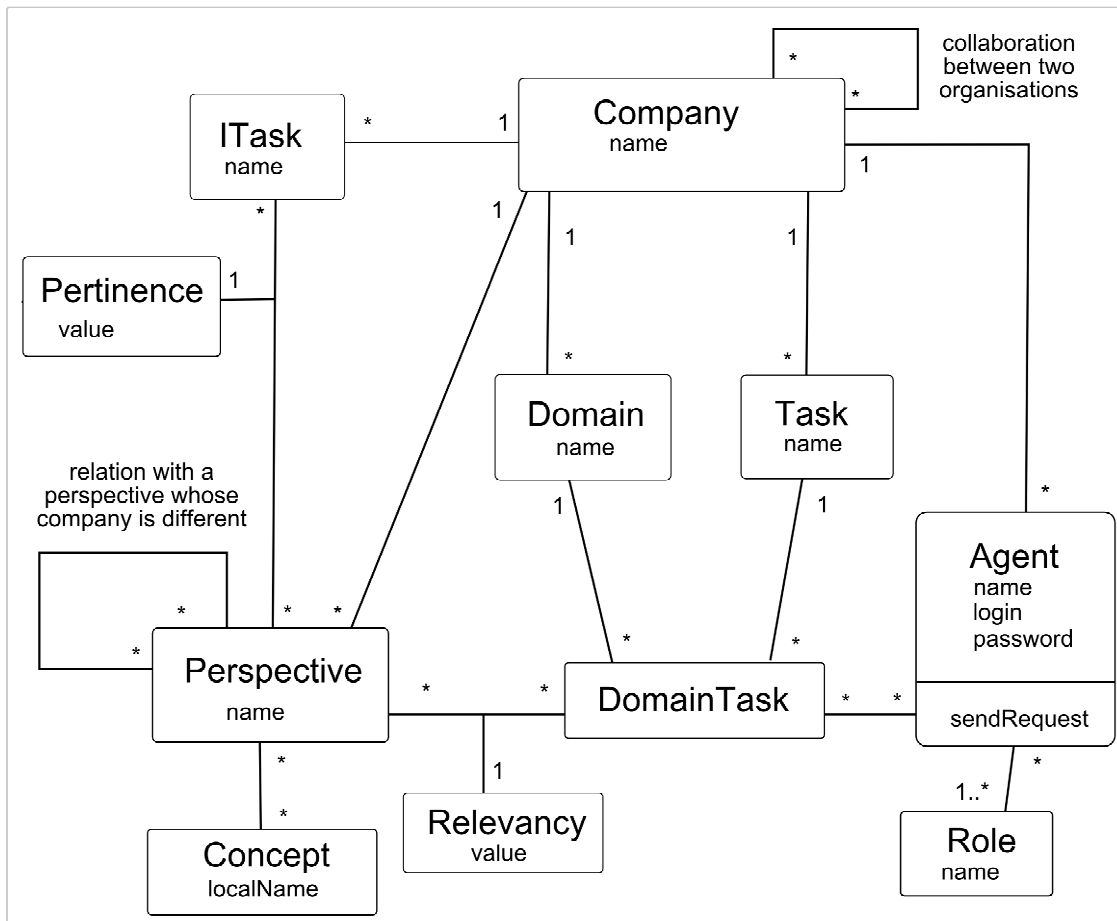


Figure 40: Class diagram to represent contextual information

For the same reasons, we choose to consider the case where the system is implemented as an external service provided to facilitate the collaboration between organizations.

Non-confidential data is uploaded by the various organizations. The data is kept private, unless there is an explicit collaboration between two organizations.

The Figure 40 displays the class diagram for the connections between contexts. The two main classes are Company and Perspective. The former permits to collect on the server all information concerning the different companies, and make it public only when there is a collaboration recorded between the companies, expressed by a relation between the 2 classes Company. The latter class is central to the model, being the common point between the classes ITask, DomainTask and Concept. For each usage of context implemented, we will see the resulting database schema.

The database schemas have been made using the MySQL workbench. The yellow symbol in form of a key indicates a primary key. The foreign keys are presented just after, symbolized with red diamonds, their name begins with “id” as well, and has usually the same name as the primary key it references. Last come the other attributes, displayed with a blue diamond. The arrow demonstrates the relation of dependency of one of the foreign key with a primary key of another table. For example, in the Figure 41, the table “Concept-Perspective” has two foreign keys. The first references the primary key “idConcept” in the table “Concept”. The second references the primary key “idPerspective” in the table “Perspective”.

21.1.1 Annotation of concepts with perspectives

The Figure 41 shows the database schema corresponding to the implementation of the usage of disambiguation. Perspectives are associated with a company, and are given a name. We associate directly perspectives to concepts. Concepts are identified by their URI, made of a base URL and a local name. For reasons of simplicity, we associate perspectives only to concepts, and not to other kinds of entities. The tables Concept and Ontology are notably related so as to be able to determine which concepts are represented in the ontologies of the same company.

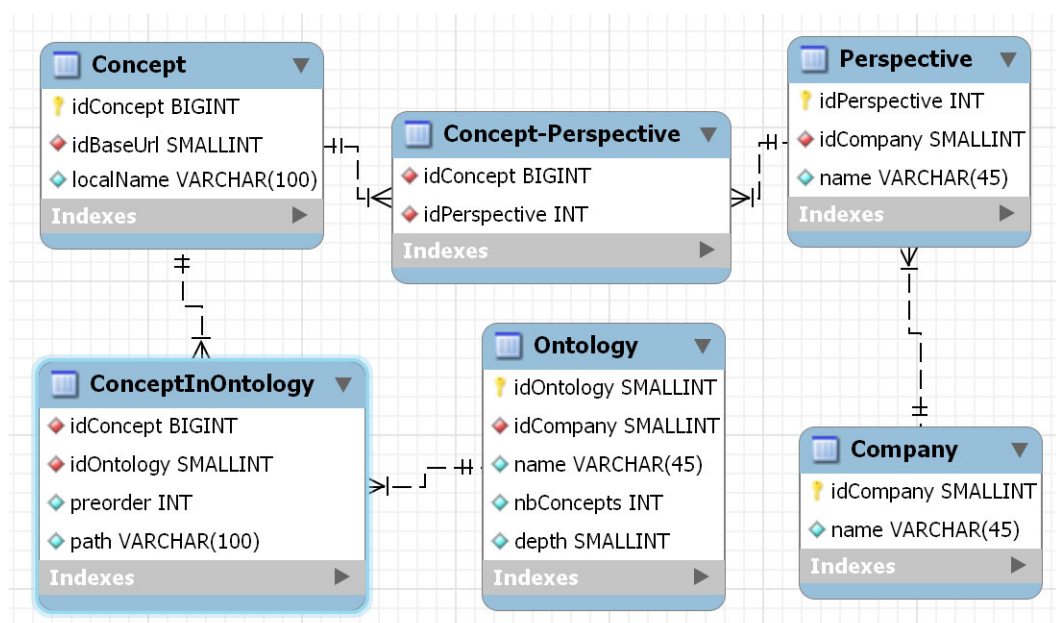


Figure 41: Database schema about annotation of concepts with perspectives

When annotating concepts with perspectives, it may be often useful to give the possibility to annotate in the same time all concepts subsumed. The Table ConceptInOntology shows the different *is-a* paths that relate the concept to the Top concept of the ontology. We associate with each of these paths a chain of characters, according to [Bidault, 2002]. The principle of generation of these paths is illustrated Figure 42 with the ontology in the Annex A. The path of the Top concept is empty. Each leaf is associated a distinct character of a given alphabet (such as the set of alpha-numeric characters), in the order where they are treated. Each leaf adds a new character, in the same way.

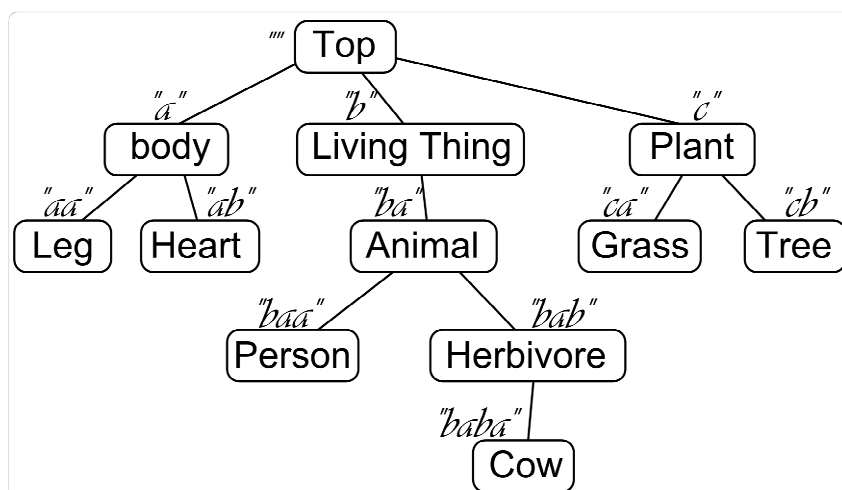


Figure 42: Association of a path following [Bidault, 2002]

Having these paths, one knows which concepts are subsumed by a concept, as they correspond to all the paths that begin with the same characters. For example, all the concepts subsumed by “Living Thing” are associated with a path that begins with “b”. And if a path begins by “b”, for the same ontology, then it specializes the concept “Living Thing”. Now, to facilitate the search of these concepts, we also include the number corresponding to the tree pre-ordering. The Figure 43 gives an example of pre-ordering. The Top concept is numbered with 1. Then the first leaf is treated. Its root “body” is given the next number, 2. The first leaf is treated, again, “Leg” is given the number 3. This concept has no leaves. The second leaf of “body” is treated, and given the next number 4. No more leaves, neither for “Heart” nor for “body”. We pass to the next leaf of the Top concept, with is given the number 5. The procedure is continued for the rest of the tree.

When ordering the concepts according to this number, the paths are ordered in the alpha-numeric order (in the case this alphabet was chosen). This simplifies the search of all concepts subsumed for a concept. There may be different paths for a concept, in an ontology. Any of these paths will give the same result. This repetition is done to take into consideration the possible multiple inheritances of the concepts, and the variation in the computation of the generic semantic measure, treated at the end of this chapter.

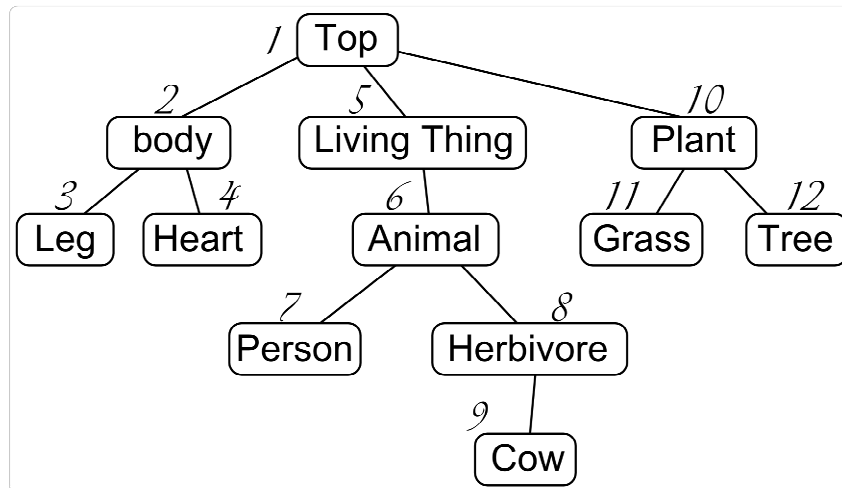


Figure 43: Ontology with pre-ordering numbers

21.1.2 Connection between perspectives

The Figure 44 shows how the connection between perspectives may be implemented. Each perspective is associated with a company, and perspectives from distinct organizations can be mapped with one another. The table “Collaboration” shows what relations are active, and may be seen or utilized for contextualization.

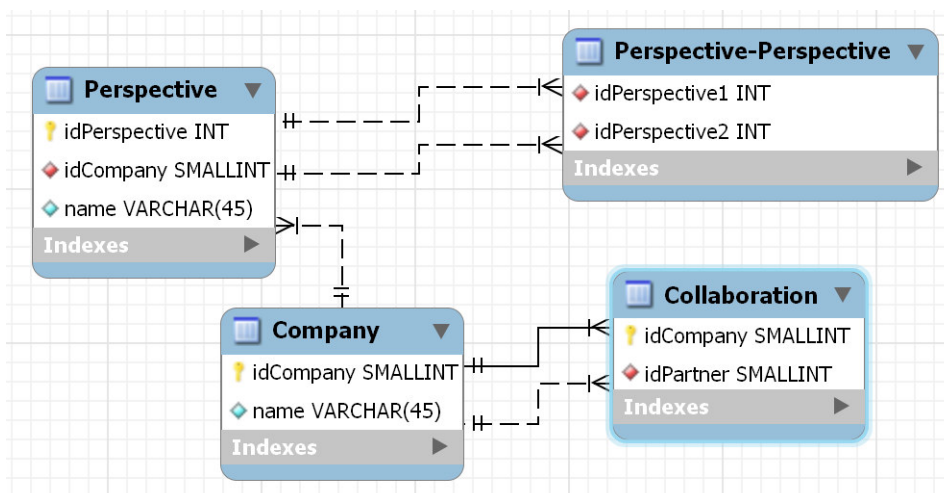


Figure 44: Database schema about relations between perspectives

21.2 Personalization by the comparison of domains/tasks and perspectives

In this section, we will describe the implementation of the usage of personalization by the connection of domains/tasks selected by an agent and the company perspectives.

The Figure 45 shows how this implementation can be done. Each couple (domain, task) that is relevant is identified in the table “Domain-Task”, and associated in the table DTP (named after “Domain-Task-Perspective”) with as many values of relevance as there are perspectives for which the couple is relevant. The agent context is described as a selection of these couples that the agent finds relevant.

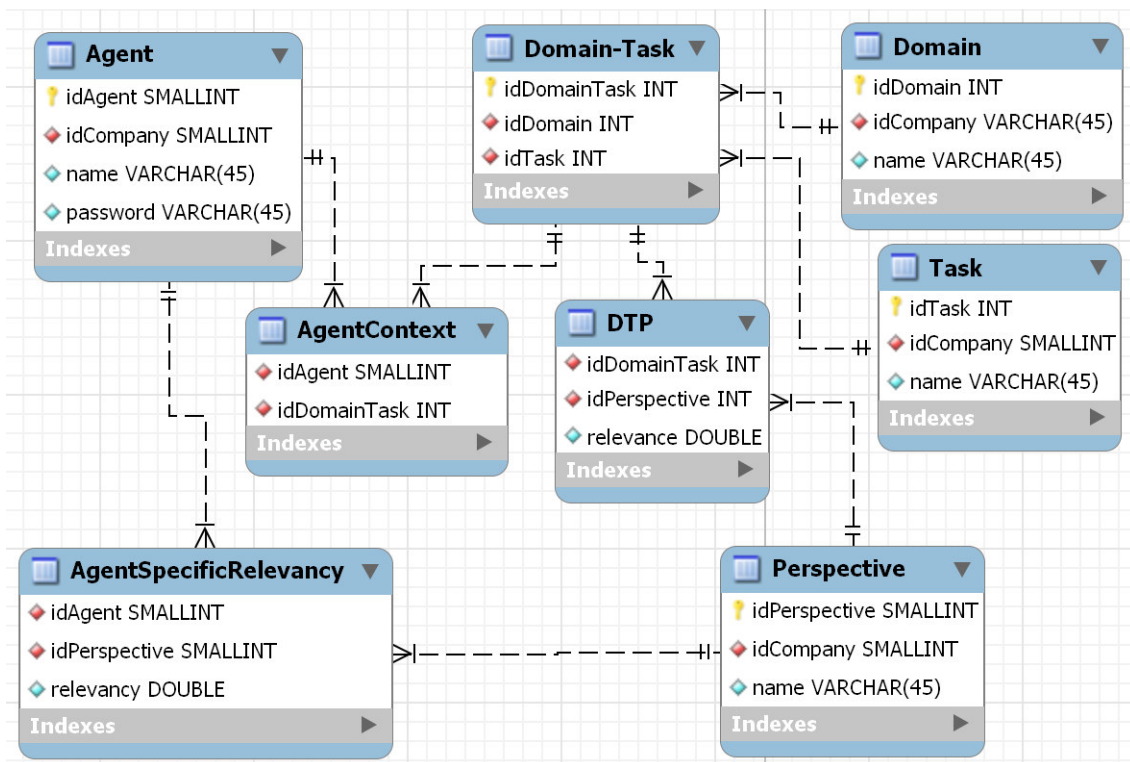


Figure 45: Database schema about the agent context

The computation of the agent-specific relevancy value for all perspectives may be stored in a table “AgentSpecificRelevancy”. This table stores the result of the pre-computation of the relevancy of perspectives for the agent. This is not needed if default values are not implemented: in that case, the agent-specific relevancy value for a perspective is the maximum value for the perspective in the table DTP among all couples (domain, task) associated with the agent in the table AgentContext. It can be computed quickly by a simple SQL query.

Implementation of default values of relevancy, for example with a perspective and a domain or with a perspective and a task, may be done by introducing a fictitious domain “DefaultDomain” and a fictitious task “DefaultTask”. Algorithm 1 presents a possible algorithm to retrieve applicant-specific relevancies in that case.

Algorithm 1: Algorithm to compute an agent applicant-specific relevancy for perspectives

```

Initiate the table AgentSpecificRelevancy with the perspectives
associated with at least one DomainTask relevant for the agent,
and the maximum value of relevancy associated in the table DTP
for this DomainTask.
For each perspective present in the table AgentSpecificRelevancy do
| Select the rows of the table “DTP” where the DomainTask
| includes a default Task or Domain, and the other is present
| in the selection of the agent context
| If the value of relevance is higher than the value recorded
| for that perspective in the table AgentSpecificRelevancy,
| then replace that value

```

21.3 Evaluation by the comparison of interoperability tasks and perspectives

In this section, we will describe the implementation of the usage of evaluation by the connection of the interoperability task selected at the moment of the request with the perspectives of the partner organization.

The interoperability tasks are specific to the company. They are related to perspectives which have been defined for a partner organization. Figure 46 shows a possible implementation of the connection between interoperability tasks and perspectives. The table ITask-Perspective associates a value of pertinence with appropriate couples of interoperability task and perspectives that have been defined by different companies. The table “Concept-Perspective” is represented here to demonstrate how the interoperability task is related to the concept enquired through the concept perspectives.

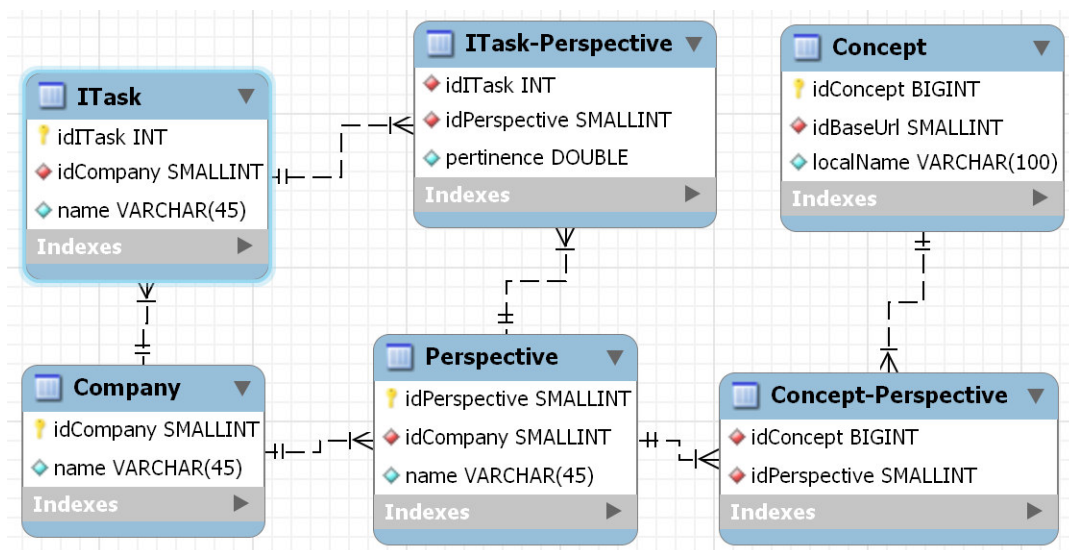


Figure 46: Database schema about interoperability tasks and perspectives

Algorithm 2 describes how to retrieve the couples of perspectives that are relevant for a request, based on the Figure 44, Figure 46, and Figure 41.

Algorithm 2: Search of couples of perspectives and the associated relevancy and pertinence.

```

Select all perspective in the table "ITask-Perspective" that is
related to the concept enquired in "Concept-Perspective", where
the interoperability task is the one selected (named "it").
For each of these perspective p2 do
| Select in the table "Perspective-Perspective" all perspectives
| p1 related to p2, whose company is the same as an ontology
| associated with the ontology of the concept enquired,
| and that are related to the agent indicated in the request
| in the table "AgentSpecificRelevancy".
| Retrieve the pertinence value of the couple (p2, it)
| Retrieve the relevancy value associated with p1 for the agent.

```


21.4 Evaluation of the generic measure of semantic similarity

In this section, we will describe the implementation of the generic measure of semantic similarity. Figure 47 represents the class diagram that displays the relations between classes for the computation of the generic measure of semantic similarity. Concepts may be added into the system either by a mapping set or by an ontology. A concept may be present in more than one ontology, and more than one mapping set, and is identified by its URI, made of base URL and local name. If a concept is present in an ontology, it can be associated with different paths in the table `ConceptInOntology`.

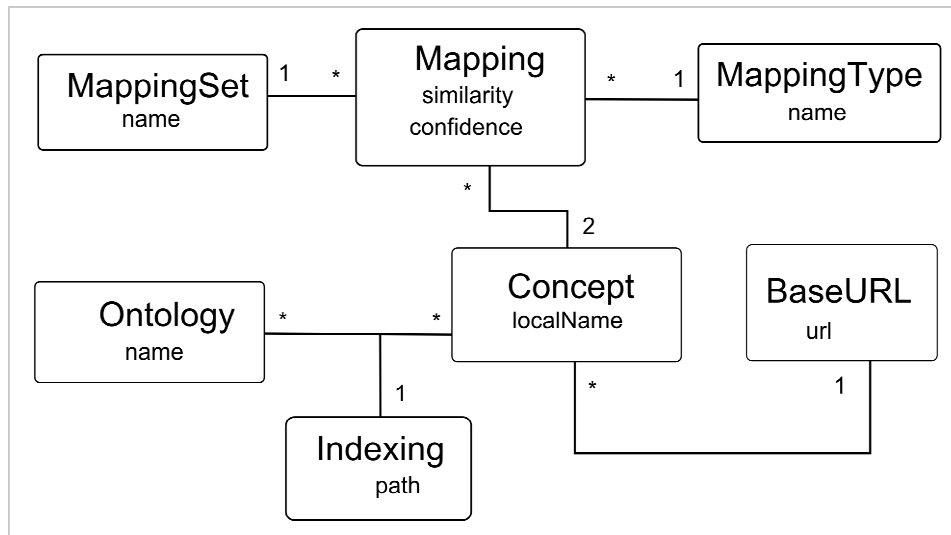


Figure 47: Class diagram for the computation of generic similarity

In the first section of this chapter, we described how a path and a pre-order number were associated with each concept in an ontology, according to [Bidault, 2002]. The number of such paths associated with a concept in an ontology is equal to the number of possibilities to relate that concept to the Top concept of the ontology. Algorithm 3 describes how to compute the edge-counting distance between concepts in an ontology using this implementation.

Algorithm 3: Computation of the measure between two concepts in an ontology

<p><i>Generic Distance Intra Ontology (GeDIO)</i> Returns: the minimal distance for all the paths in the ontology Input: Concept <code>cpt1</code>, Concept <code>cpt2</code>, Ontology <code>ont</code></p> <pre> minDist <- {highest possible value} For each path1: getPaths(cpt1, ont) do For each path2: getPaths(cpt2, ont) do String common <- commonPath(path1, path2) newDist <- len(path1) + len(path2) - 2x len(common) minDist <- min(minDist, newDist) return minDist </pre>
<p><i>getPaths</i> Returns: the list of paths that relate the concept to the top concept Input: Concept <code>cpt</code>, Ontology <code>ont</code></p> <p>Return all path from the table "ConceptInOntology" where <code>idConcept=cpt</code></p>

and idOntology=ont
<i>commonPath</i> Input: String path1, String path2
Return the String that is common to the paths, beginning with their first character

Figure 48 shows a possible implementation of the generic measure of semantic similarity. We aggregate mappings into mappings sets, as the mappings may be the result of distinct ontology matching methods and tools. The table MappingType serves to indicate a mapping of equivalence or subsumption. The values of similarity and certainty are not used, but might be used for further parameterisation of the algorithm. Algorithm 4 shows how to compute the generic measure of semantic similarity described in Chapter 16.

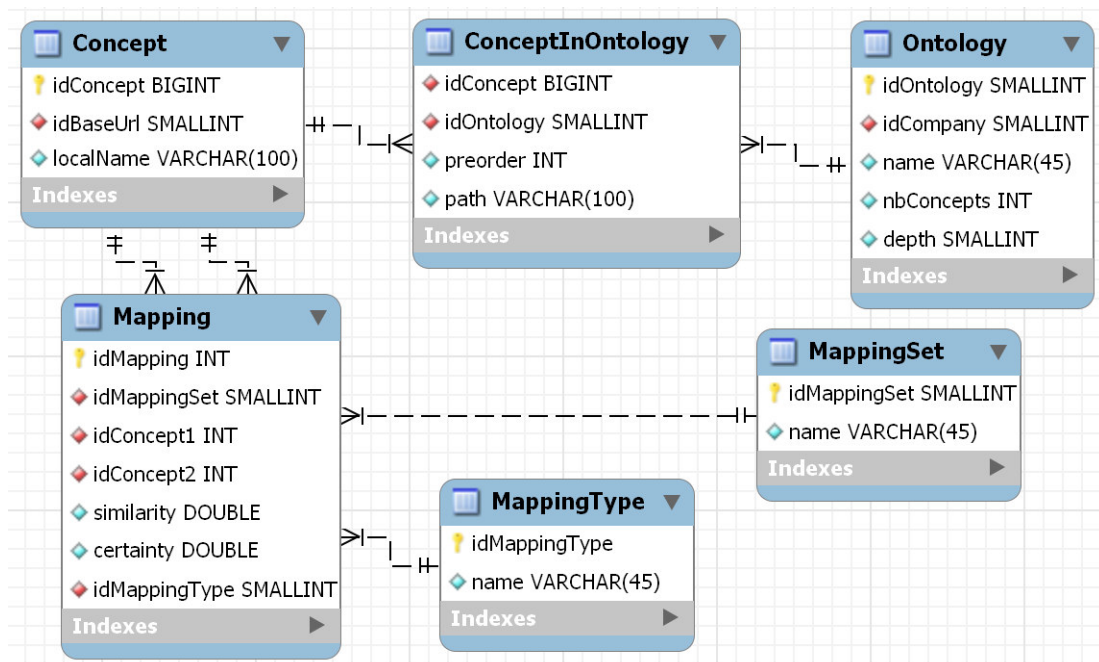


Figure 48: Database schema about similarity between concepts

Algorithm 4: Computation of the measure of semantic similarity

<i>getMappings</i> Returns: the list of mappings between two ontologies Input: Ontology ont1, Ontology ont2
Return all mappings from table "Mapping" for which there exist in table "ConceptInOntology" a row with their concept idConcept1 and the ontology ont1, and a row with their concept idConcept2 and the ontology ont2
<i>Similarity</i> Returns: similarity value Input: Distance dist1, Ontology ont1, Distance dist2, Ontology ont2, mapping depth1 <- depth(ont1)

```

depth2 <- depth(ont2)
lambda <- { 1 if type(mapping) = subsumption
           { 0 otherwise
sim <- dist1 x depth2 + dist2 x depth1 + lambda x depth1 x depth2
sim <- sim / (depth1 x depth2)
Return sim

Generic similarity measure across ontologies
Returns: similarity of concepts across ontologies
Input: Concept cpt1, Concept cpt2

maxSimilarity <- 0
For each ontology ont1 that contains cpt1 do
  | For each ontology ont2 that contains cpt2 do
    | mappings <- getMappings(ont1,ont2)
    | For each mapping(cptmapped1, cptmapped2): mappings do
      | dist1 <- GeDIO(cpt1,ctpmapped1,ont1)
      | dist2 <- GeDIO(cptmapped2,ctp2,ont2)
      | sim <- similarity(dist1, ont1, dist2, ont2, mapping)
      | maxSimilarity <- max(maxSimilarity, sim)
Return maxSimilarity

```

21.5 Conclusion

We have presented in this part how our approach could be implemented based on a relational database design. We have described an algorithm to determine the couples of perspectives to return for the evaluation of the concept with their respective values of relevancy and pertinence, an algorithm to compute an edge-counting distance between concepts in an ontology relying on previous indexation, and an algorithm to measure the generic semantic similarity between concepts across ontologies. We have presented an example of application of the approach adapted from the real world of collaborations between suppliers and clients, and accessible to the general public.

We will now conclude the study by highlighting our contributions, discussing the intrinsic limitations of the approach proposed, and suggesting research directions for further improvement of the approach as well as for more reliable ontology definition and reconciliation.

General Conclusion

Thesis summary and contributions

This thesis studies how a context-based approach could improve interoperability between heterogeneous ontologies developed and evolving autonomously. The hypothesis for this study is that organizations have ontologies that represent the concepts corresponding to the data they manipulate, and that they need to collaborate for concurrent projects. For these projects, data is exchanged, and so they have to reconcile their ontologies to facilitate this exchange.

As these ontologies have been developed independently and are evolving autonomously, ontology mapping is the appropriate approach to reconcile these ontologies. Tools exist to find possible mappings between ontology entities. Yet, as the ontologies have been developed in totally different settings, the entities mapped have different scope, are associated with different kinds of data, and so forth (Chapter 3, Section 1, page 28). Also, supposing that the need of an agent is well described by a concept defined by an ontology of his company; the concept defined in an ontology from a partner organization that corresponds the best to an agent's requirement may not be the concept that is in average the most similar to the first concept. It may instead be a concept that is more similar to it in the particular context of the interoperability need.

We propose in this study to examine the context surrounding the reconciliation of ontologies by mappings. Traditionally, ontologies are mapped without considering any contextual information on how the concepts will actually be used. Our approach is based on the intuition that by considering the context, one could provide a better evaluation of pertinence of a concept for a given interoperability request.

This brings us to our first contribution: we have reviewed the literature concerned with context in Computer Science, to expose the main definitions and the principles that rule the modeling and use of context. We proposed a general definition of context, named “connectional context”, as “*any information that participates in characterizing an entity of interest (...) where the distinctive features that compose the characterization are judged according to a given purpose*”. Following this definition, we have developed a methodology to determine what is contextual, how to model and use it.

We have applied this methodology to the reconciliation of ontologies, in the situation of concurrent collaborations between an organization and two or more collaborators, where flexibility is needed. We have found that a reasonable solution to disambiguate the pragmatic meaning of concepts was to annotate ontology concepts with perspectives. These perspectives were recognized to have guided the ontology development, or reveal different kinds of data associated with the ontology. This is our second contribution.

Ontology reconciliation done by ontology mappings is also limited in that only a few concepts are related to one another. We therefore propose a generic semantic similarity measure across ontologies to relate all the concepts defined in the ontologies from the agent's company with concepts defined in ontologies from other organizations. We base this measure on the intra-ontology edge-counting semantic similarity measure from [Leacock and Chodorow, 1998], and on existing ontology mappings. This is our third contribution.

Finally, we have proposed a context-based evaluation of pertinence of a "concept enquired", based on three kinds of contextual information. The evaluation requires the following information: the identity of the agent making the request, the description of the need by a "root concept", and the "interoperability task" for which the request is made. With the help of the methodology, we have indeed recognized and modeled three contexts:

1. The context of the ontology concepts made of perspectives, as described earlier. The perspectives of the root concept are compared with the perspectives of the concept enquired, to *disambiguate* the pragmatic meanings of concepts by keeping only the couples of perspectives that are compatible with one another.
2. The context of the agent that makes the request, which consists in a selection of domains and tasks of his company; this context is introduced to *personalize* the measure, by recognizing the understanding that the agent has of the root concept, and retaining sole, the perspectives that are relevant for him.
3. The context of the interoperability need, made of the interoperability task for which the entity searched for is intended to be used. This context is introduced so as to *evaluate* the practical value of the concept enquired. The perspectives of the concept enquired are evaluated against the interoperability task, to judge of their pertinence.

The context-based evaluation of pertinence is composed of a generic value of similarity, and of couples of values of pertinence and relevancy corresponding to the couples of perspectives compatible among the perspectives describing the pragmatic meaning of root concept and concept enquired. This context-based measure of pertinence across ontologies is the last of our four contributions.

Discussion and limits

The study has partially answered to the first research question: with limited requirements concerning the source of contextual information (the ontology itself, limited information given by experts, limited information given by users), one can have an evaluation of the practical pertinence of a concept for an interoperability need.

It does not provide a direct answer concerning the expectations that one can have for context-based approaches applied to ontology reconciliation. Our approach is based on data that we estimate to be available in most cases. It seems that in the state of the art of ontology development, most information that is contextual for ontology reconciliation cannot be retrieved automatically. To be automatically processable, contextual information would indeed have to be described completely by a list of meta-data properties established for each usage of context that is to be implemented. To generalize the contextualisation of ontology reconciliation, the annotation with this meta-data

would have to be standardized. Although the context-based approach could be taken further than that which we proposed in this study, it would therefore certainly require much more manual work to annotate the ontology entities with the appropriate meta-data, or to retrieve some other contextual information manually. The amount of effort to be made in preparing the contextual data has to be determined in relation to the gain expected.

Our approach requires that a certain amount of preliminary work should be done by some members of the organization who have good knowledge of the company domains and tasks, and are able to identify the perspectives with or for which the ontologies have been developed. This lays another burden on the shoulders of these experts: they have not only to verify the mappings returned by ontology matching tools, but also to determine perspectives, to compare them with the perspectives from the organizations with which the company needs to collaborate, and compare these latter with interoperability tasks of the company.

Is the preparatory work worth it? It certainly is, if as a result the mappings can actually be used, and the comparison between concepts trusted. However, the reliability of the approach is limited by the same problem that affects ontology modelling and development: the recognition of perspectives and the attribution of relevancy and pertinence values will vary from one expert to another.

Future research

Desirable improvements for the approach proposed include:

☞ Discover potential concepts enquired in other organizations' ontologies concepts, for example by searching all concepts that are similar to a given root concept, and are defined by other ontologies. Indeed, the approach presented in this thesis requires the request to be made of two concepts, while it is probable that in most cases, only the concept describing the need (root concept) might be known.

☞ Give the possibility to describe the request, not with a root concept only, but with two or more concepts. The system implementing such an approach could for example take advantage of standard semantic relations to retrieve the concepts that are semantically related to all of the concepts queried.

☞ Improve the generic semantic similarity measure by considering the level of confidence associated with mappings to present different values of similarity depending on the minimal confidence required by the applicant for an interoperability need.

Perspectives

We propose the following research directions for ontology reconciliation:

- ☞ Develop standard micro-ontologies, that would be limited to a very specific and limited domain;
 - Have tools for ontology development (such as a plug-in in Protégé) that get connected to these micro-ontologies to propose to refer existing concepts when

possible, for example by doing a keyword-based search among these micro-ontologies.

- The micro-ontologies should be task-oriented. For example there could be an ontology that defines the information included in postal addresses. By referring this ontology, one would know that the data associated with the concept may serve for operations such as mail sending.

☞ Standardize semantic relations between concepts, and oblige all semantic relations defined by the user to specialize a semantic relation. By doing this, both the comparison of ontologies⁷³ and the search in ontologies would be simplified, and made more reliable. This would open the door to the use of semantic relations to determine the interest of a concept for another. Examples of standard relations could include *medium* (inverse relation *content*), *content source*.

☞ Modify the conceptual model of ontologies, so that the *is-a* links are not fixed, but associated with a context defined explicitly: the categorization would depend on the view that one has on the concepts. Indeed, categorizations are only valid for a paradigm, which should be included in the representation.

⁷³ When properties are fixed, it gets easier to determine mappings between concepts (and reciprocally).

Bibliography

- Albertoni, Riccardo, et Monica De Martino. "Asymmetric and Context-Dependent Semantic Similarity among Ontology Instances." Dans *Journal on Data Semantics X*, 1-30, 2008. http://dx.doi.org/10.1007/978-3-540-77688-8_1.
- . "Semantic Similarity of Ontology Instances Tailored on the Application Context." Dans *On the Move to Meaningful Internet Systems 2006: CoopIS, DOA, GADA, and ODBASE*, 1020-1038, 2006. http://dx.doi.org/10.1007/11914853_66.
- Aleman-Meza, Boanerges. "Searching and Ranking Documents based on Semantic Relationships," 2006. [http://csdl2.computer.org/persagen/DLAbsToc.jsp?resourcePath=/dl/proceedings/s/&toc=comp/proceedings/icdew/2006/2571/00/2571toc.xml&DOI=10.1109/ICDEW.2006.131](http://csdl2.computer.org/persagen/DLAbsToc.jsp?resourcePath=/dl/proceedings/&toc=comp/proceedings/icdew/2006/2571/00/2571toc.xml&DOI=10.1109/ICDEW.2006.131).
- Aleman-Meza, Boanerges, Christian Halaschek-Wiener, I. Budak Arpinar, Cartic Ramakrishnan, et Amit P. Sheth. "Ranking Complex Relationships on the Semantic Web." *IEEE Internet Computing* 9, no. 3 (2005): 37-44. doi:<http://doi.ieeecomputersociety.org/10.1109/MIC.2005.63>.
- Analyti, Anastasia, Manos Theodorakis, Nicolas Spyrtatos, et Panos Constantopoulos. "Contextualization as an independent abstraction mechanism for conceptual modeling" (2007): 24-60.
- Asperti, A., et G. Longo. *Categories, types, and structures*. MIT Press Cambridge, Mass, 1991.
- Attardi, G., et M. Simi. "A Formalization of Viewpoints." *Fundamenta Informaticae* 23 (1995): 149-173.
- Baeza-Yates, Ricardo, et Berthier Ribeiro-Neto. *Modern Information Retrieval*. 1er éd. Addison Wesley, 1999.
- Baldauf, Matthias, Schahram Dustdar, et Florian Rosenberg. "A survey on context-aware systems." *IJAHUC* 2 (2007): 263-277.
- Bao, Jie, Doina Caragea, et Vasant Honavar. "Modular Ontologies - A Formal Investigation of Semantics and Expressivity.." Dans , édité par Riichiro Mizoguchi, Zhongzhi Shi, et Fausto Giunchiglia, 4185:616-631. Lecture Notes in Computer Science. Springer, 2006. <http://dblp.uni-trier.de/db/conf/aswc/aswc2006.html#BaoCH06>.
- Bartholomew, Doug Duvall. *PLM: Boeing's Dream, Airbus' nightmare*, 2007.
- Benslimane, Djamel, Ahmed Arara, Gilles Falquet, Zakaria Maamar, Philippe Thiran, et Faiez Gargouri. "Contextual Ontologies." Dans *Advances in Information Systems*, 168-176, 2006. http://dx.doi.org/10.1007/11890393_18.
- Benslimane, Djamel, Christelle Vangenot, Catherine Roussey, et Ahmed Arara. "Multirepresentation in Ontologies." Dans *Advances in Databases and Information Systems*, 4-15, 2003. <http://www.springerlink.com/content/5papg0dn59ycyj9r>.
- Berg, E. van den, R. Bidarra, et W. F. Bronsvort. "Web-based interaction on feature models." *From Geometric Modeling to Shape Modeling: Ifip Tc5 Wg5*. 2

- Seventh Workshop on Geometric Modeling: Fundamentals and Applications, October 2-4, 2000, Parma, Italy* (2001).
- Berners-lee, Tim. *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web*. 1er éd. Collins, 2000.
- Bernstein, Abraham, Esther Kaufmann, Christoph Bürki, et Mark Klein. "How Similar Is It? Towards Personalized Similarity Measures in Ontologies." Dans *Wirtschaftsinformatik 2005*, 1347-1366, 2005. http://dx.doi.org/10.1007/3-7908-1624-8_71.
- Bharat, Krishna. "SearchPad: explicit capture of search context to support Web search." *Computer Networks* 33 (2000): 493-501. doi:10.1016/S1389-1286(00)00047-5.
- Bidault, A. "Affinement de requêtes posées à un médiateur." Université Paris XI, Orsay, Paris, France, 2002.
- Blanchard, E., M. Harzallah, P. K. H. Briand, et P. Kuntz. "A typology of ontology-based semantic measures." Porto, Portugal, 2005.
- Blanchard, Emmanuel, Pascale Kuntz, Mounira Harzallah, et Henri Briand. "A Tree-Based Similarity for Evaluating Concept Proximities in an Ontology." Dans *Data Science and Classification*, 3-11, 2006. http://dx.doi.org/10.1007/3-540-34416-0_1.
- Bodenreider, Olivier, Tom Gruber, Nicola Guarino, Ivan Herman, Deborah McGuinness, Mark Musen, et coll. *Ontology, Taxonomy, Folksonomy: Understanding the distinctions*. NIST, Gaithersburg, Maryland, USA.: Ontology Summit 2007, 2007. http://ontolog.cim3.net/cgi-bin/wiki.pl?OntologySummit2007_Communique.
- Bouquet, Paolo, Heiko Stoermer, et Xin Liu. "Okkam4P: A Protégé Plugin for Supporting the Re-use of Globally Unique Identifiers for Individuals in OWL/RDF Knowledge Bases." Dans , 314: Bari, Italy, 2007.
- Bradley, N.A., et M.D. Dunlop. "Toward a multidisciplinary model of context to support context-aware computing," 2005. ScientificCommons. <http://eprints.cdlr.strath.ac.uk/2927/>.
- Brézillon, Patrick. "Context Modeling: Task Model and Practice Model." Dans , 122-135. Springer Berlin / Heidelberg, 2007. http://dx.doi.org/10.1007/978-3-540-74255-5_10.
- Brunnermeier, Smita, et Sheila A. Martin. *Interoperability Cost Analysis of the U.S. Automotive Supply Chain: Final Report*. DIANE Publishing, 1999.
- Budanitsky, A., et G. Hirst. "Semantic distance in WordNet: An experimental, application-oriented evaluation of five measures." Dans , 2:, 2001.
- Budzik, J., et K. J. Hammond. "User interactions with everyday applications as context for just-in-time information access." *Proceedings of the 5th international conference on Intelligent user interfaces* (2000): 44-51.
- Byström, K., et P. Hansen. "Work Tasks As Units for Analysis in Information Seeking and Retrieval Studies." *Emerging Frameworks and Methods: Proceedings of the Fourth International Conference on Conceptions of Library and Information Science (Colis 4)* (2002).
- Castano, Ferrara, et Montanelli. "Matching Ontologies in Open Networked Systems: Techniques and Applications." *Journal on Data Semantics V* 3870/2006. Lecture Notes in Computer Science (2006): 25-63.
- Chalmers, M. "A Historical View of Context." *Computer Supported Cooperative Work (CSCW)* 13 (2004): 223-247.

- Chalupsky, H. "OntoMorph: A Translation System for Symbolic Knowledge." Dans , 471-482. Colorado,USA, 2000.
- CHEN, David, et Guy DOUMEINGTS. "Basic Concepts and Approaches to Develop Interoperability of Enterprise Applications." Dans , 323-330. Lugano, Switzerland: Kluwer Academic Publisher, 2003.
- Clark, Nicola. "The Airbus saga: Hubris and haste snarled the A380." *International Herald Tribune* (2006). <http://www.iht.com/bin/print.php?id=3854623>.
- Costello, Roger L. *Enhancing Data Interoperability with Ontologies, Canonical Forms, and Include Files*, 2003. <http://www.xfront.com/interoperability/CanonicalForms.html>.
- Curino, Carlo, Giorgio Orsi, et Letizia Tanca. "X-SOM: A Flexible Ontology Mapper." Dans , 424-428. IEEE Computer Society, 2007.
- Dartigues, Christel, Parisa Ghodous, Michael Gruninger, Denis Pallez, et Sriram Ram. "CAD/CAPP Integration using Feature Ontology." *Concurrent Engineering* (2007): 237-249.
- Dey, Anind K. "Understanding and Using Context." *Personal and Ubiquitous Computing* 5, no. 1 (Février 20, 2001): 4-7. doi:10.1007/s007790170019.
- Ding, Kolari, Ding, et Avancha. "Using Ontologies in the Semantic Web: A Survey." Dans *Ontologies*, 14:79-113. Integrated Series in Information Systems. Springer US, 2007. http://dx.doi.org/10.1007/978-0-387-37022-4_4.
- Do, Melnik, et Rahm. *Comparison of Schema Matching Evaluations*, 2003. <http://www.springerlink.com/content/4wbxb1jbw8w4ek9v>.
- Doan, AnHai, Jayant Madhavan, Robin Dhamankar, Pedro Domingos, et Alon Halevy. "Learning to match ontologies on the Semantic Web." *The VLDB Journal* 12 (2003): 303-319.
- Ehrig, M., P. Haase, M. Hefke, et N. Stojanovic. "Similarity for Ontologies-a Comprehensive Framework." Dans , 13-24, 2004.
- Ekbia, Hamid R., et Ana Gabriela Maguitman. "Context and Relevance: A Pragmatic Approach." Dans , 156-169. Springer-Verlag, 2001.
- Euzenat, Jérôme, Thanh Le Bach, Jesús Barrasa, Paolo Bouquet, Jan De Bo, Rose Dieng-Kuntz, Marc Ehrig, et al. *State of the art on ontology alignment*. Deliverable. Knowledge web, 2004. <ftp://ftp.inrialpes.fr/pub/exmo/reports/kweb-223.pdf>.
- Euzenat, Jérôme. *An API for ontology alignment*, 2006. <http://gforge.inria.fr/docman/view.php/117/251/align.pdf>.
- Euzenat, Jérôme, Marc Ehrig, Anja Jentsch, Malgorzata Mochol, et Pavel Shvaiko. *Case-based recommendation of matching tools and techniques*. Knowledge Web, 2007. <ftp://ftp.inrialpes.fr/pub/exmo/reports/kweb-126.pdf>.
- Euzenat, Jérôme, et Pavel Shvaiko. *Ontology Matching*. Berlin Heidelberg (DE): Springer-Verlag, 2007.
- Finkelstein, L., E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, et coll. "Placing search in context: the concept revisited." *Proceedings of the 10th international conference on World Wide Web* (2001): 406-414.
- Firat, A., S. Madnick, et F. Manola. "Multi-dimensional Ontology Views via Contexts in the ECOIN Semantic Interoperability Framework" (2005).
- Firat, Aykut, Stuart Madnick, et Benjamin Grosf. "Contextual alignment of ontologies in the eCOIN semantic interoperability framework" (2007): 47-63.

- FRIEDLAND, NS, PG ALLEN, G. MATTHEWS, M. WITBROCK, D. BAXTER, J. CURTIS, et coll. "Project Halo towards a Digital aristotle." *The AI magazine* 25 (2004): 29-47.
- Gagnon, Michel. "Ontology-based integration of data sources." Dans , 1-8, 2007. doi:10.1109/ICIF.2007.4408086.
- Gal, Avigdor, Giovanni Modica, Hasan Jamil, et Ami Eyal. "Automatic ontology matching using application semantics" (2005): 21-31.
- Gallaher, Michael P., Alan C. O'Connor, et Thomas Phelps. *Economic Impact Assessment of the International Standard for the Exchange of Product Model Data (STEP) in Transportation Equipment Industries*. RTI International, 2002. www.nist.gov/director/prog-ofc/report02-5.pdf.
- Gärdenfors, P. *Conceptual Spaces: The Geometry of Thought*. Mit Pr, 2000.
- Geryville, H. M., A. Bouras, Y. Ouzrout, et N. S. Sapidis. "A solution for actors? viewpoints representation with collaborative product development." *Proceedings of Virtual Concept* (2006).
- Ghidini, et Serafini. "Reconciling Concepts and Relations in Heterogeneous Ontologies," 2006. http://dx.doi.org/10.1007/11762256_7.
- Ghidini, Serafini, et Tessaris. "On Relating Heterogeneous Elements from Different Ontologies," 2007. http://dx.doi.org/10.1007/978-3-540-74255-5_18.
- Ghidini, Chiara, et Fausto Giunchiglia. "Local Models Semantics, or contextual reasoning=locality+compatibility" (2001): 221-259.
- Giunchiglia, F., M. Schorlemmer, F. McNeill, A. Bundy, M. Marchese, M. Yatskevich, et coll. *OpenKnowledge Deliverable 3.1.: Dynamic Ontology Matching: a Survey*. Trento, Italy, 2006.
- Goldstone, R. L., D. L. Medin, et J. Halberstadt. "Similarity in context." *Memory and Cognition* 25, no. 2 (1997): 237-255.
- Gracia, Jorge, Vanesa Lopez, Mathieu D'Aquin, Marta Sabou, Enrico Motta, et Eduardo Mena. "Solving Semantic Ambiguity to Improve Semantic Web based Ontology Matching." Dans , édité par Pavel Shvaiko, Jérôme Euzenat, Fausto Giunchiglia, et Bin He, 2007.
- Gruber, Thomas R. "A translation approach to portable ontology specifications." *Knowl. Acquis.* 5 (1993): 199-220. doi:<http://dx.doi.org/10.1006/knac.1993.1008>.
- Guarino, N. *Formal Ontology in Information Systems: Proceedings of the 1st International Conference June 6-8, 1998, Trento, Italy*. IOS Press, 1998.
- Guarino, N., et Giaretta, P. "Ontologies and Knowledge Bases: Towards a Terminological Clarification." Dans *Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing*, 25-32. IOS Press, Amsterdam, 1995.
- Guenther, R., et J. Radebaugh. "Understanding Metadata." *National Information Standard Organization (NISO) Press, Bethesda, USA* (2004).
- Guha, Ramanathan. "Contexts: a formalization and some applications." Stanford University, 1992.
- Haeri, S. H., H. Abolhassani, V. Qazvinian, et B. B. Hariri. "Coincidence-Based Scoring of Mappings in Ontology Alignment." *Journal of Advanced Computational Intelligence Vol 11* (2007).
- Hirst, Graeme, et David St-Onge. "Lexical chains as representations of context for the detection and correction

- of malapropisms..” Dans *WordNet: an electronic lexical database*, 305–332. I NetLibrary, 1998.
- Horridge, M., S. Bechhofer, et O. Noppens. “Igniting the OWL 1.1 Touch Paper: The OWL API.” Innsbruck, Austria, 2007. http://www.webont.org/owled/2007/PapersPDF/submission_32.pdf.
- Hyvönen, Eero, Eetu Mäkelä, Mirva Salminen, Arttu Valo, Kim Viljanen, Samppa Saarela, et coll. “MuseumFinland--Finnish museums on the semantic web.” *Web Semantics: Science, Services and Agents on the World Wide Web 3* (2005): 224-241.
- IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries: 610*. Inst of Elect & Electronic, 1991.
- Jacobs, J., et A. Linden. *Semantic Web technologies take middleware to the next level*. Gartner Group, 2002. http://www.gartner.com/DisplayDocument?doc_cd=109295.
- Janowicz, Krzysztof. “Extending Semantic Similarity Measurement with Thematic Roles.” Dans *GeoSpatial Semantics*, 137-152, 2005. http://dx.doi.org/10.1007/11586180_10.
- , “Kinds of Contexts and their Impact on Semantic Similarity Measurement,” 2008. http://ifgi.uni-muenster.de/%7Ejanowicz/wp-content/uploads/2007/12/janowicz_comorea2007_kinds_of_contexts_and_their_impact_on_semantic_similarity_measurement.pdf.
- Jiang, J. J., et D. W. Conrath. “Semantic similarity based on corpus statistics and lexical taxonomy.” *Proceedings of International Conference on Research in Computational Linguistics 33* (1997).
- Jouanot, Fabrice, Nadine Cullot, et Kokou Yétongnon. “Context Comparison for Object Fusion.” Dans *Advanced Information Systems Engineering*, 1031, 2003. http://dx.doi.org/10.1007/3-540-45017-3_36.
- Kalfoglou, Yannis, et Marco Schorlemmer. “Ontology mapping: the state of the art.” *Knowl. Eng. Rev* 18 (2003): 1-31.
- Keßler, Carsten. “Similarity Measurement in Context.” Dans *Modeling and Using Context*, 277-290, 2007. http://dx.doi.org/10.1007/978-3-540-74255-5_21.
- Keßler, Carsten, Martin Raubal, et Krzysztof Janowicz. “The Effect of Context on Semantic Similarity Measurement.” Dans *On the Move to Meaningful Internet Systems 2007: OTM 2007 Workshops*, 1274-1284, 2007. http://dx.doi.org/10.1007/978-3-540-76890-6_55.
- Kim, Junhwan, Michael J. Pratt, Raj Iyer G, et Ram D. Sriram. “Standardized data exchange of CAD models with design intent.” *Computer-Aided Design* (2007).
- Kim, Kyoung-Yun, David G. Manley, et Hyungjeong Yang. “Ontology-based assembly design and information sharing for collaborative product development” (2006): 1233-1250.
- Kingston, John. “Multi-perspective ontologies: Resolving common ontology development problems” (2008): 541-550.
- Klein, Michel, Asuncion Gomez-Perez, Michael Gruninger, Heiner Stuckenschmidt, et Michael Uschold. “Combining and relating ontologies: an analysis of problems and solutions.” *Workshop on Ontologies and Information Sharing, IJCAI'01* (2001). <http://citeseer.ist.psu.edu/klein01combining.html>.
- Laamari, et Ben Yaghlane. “Uncertainty in Semantic Ontology Mapping: An Evidential Approach,” 2007. http://dx.doi.org/10.1007/978-3-540-75256-1_38.

- Lassila, Ora, et Deepali Khushraj. "Contextualizing Applications via Semantic Middleware." Dans , 183-191. IEEE Computer Society, 2005.
- Lawrence, S. "Context in Web Search." *IEEE Data Engineering Bulletin* 23 (2000): 25-32.
- Leacock, C., et M. Chodorow. "Combining local context and WordNet similarity for word sense identification." *WordNet: An Electronic Lexical Database* 49, no. 2 (1998): 265-283.
- Li, W. D., W. F. Lu, J. Y. H. Fuh, et Y. S. Wong. "Collaborative computer-aided design?research and development status." *Computer-Aided Design* 37 (2005): 931-940.
- Lin, D. "An information-theoretic definition of similarity." *Proceedings of the 15th International Conference on Machine Learning* (1998): 296–304.
- Lord, P. W., R. D. Stevens, A. Brass, et C. A. Goble. "Investigating semantic similarity measures across the Gene Ontology: the relationship between sequence and annotation." *Bioinformatics* 19, no. 10 (Juillet 1, 2003): 1275-1283. doi:10.1093/bioinformatics/btg153.
- Ma, Y. -S., et T. Tong. "Associative feature modeling for concurrent engineering integration." *Computers in Industry* 51 (2003): 51-71. doi:10.1016/S0166-3615(03)00025-3.
- Madhavan, Jayant, Philip A. Bernstein, et Erhard Rahm. "Generic Schema Matching with Cupid." Dans , 49-58. Morgan Kaufmann Publishers Inc, 2001.
- Maguitman, Ana, Filippo Menczer, Fulya Erdinc, Heather Roinestad, et Alessandro Vespignani. "Algorithmic Computation and Approximation of Semantic Similarity." *World Wide Web* 9, no. 4 (Décembre 23, 2006): 431-456. doi:10.1007/s11280-006-8562-2.
- Malgorzata Mochol, et Elena Paslaru-bontas. "A High-Level Architecture of a Metadata-based Ontology Matching Framework." Dans , 354-358, 2006. <http://csdl2.computer.org/persagen/DLAbsToc.jsp?resourcePath=/dl/proceedings/&toc=comp/proceedings/dexa/2006/2641/00/2641toc.xml&DOI=10.1109/DEXA.2006.9>.
- Mascardi, Viviana, Valentina Cordi, et Paolo Rosso. *A comparison of upper ontologies*, 2007.
- McCarthy, John L. "Notes on Formalizing Context.." Dans , 555-562, 1993.
- Meilicke, Christian, et Heiner Stuckenschmidt. "Analyzing Mapping Extraction Approaches." Dans , édité par Pavel Shvaiko, Jérôme Euzenat, Fausto Giunchiglia, et Bin He. Bexco, Busan, Korea, 2007.
- Mitra, P., et G. Wiederhold. "Resolving terminological heterogeneity in ontologies." Proc. ECAI-02 Workshop, CEUR-WS 64, Lyon, France, 2002.
- Mochol, Malgorzata, et Elena Paslaru-bontas. "A High-Level Architecture of a Metadata-based Ontology Matching Framework." Dans , 354-358, 2006. <http://csdl2.computer.org/persagen/DLAbsToc.jsp?resourcePath=/dl/proceedings/&toc=comp/proceedings/dexa/2006/2641/00/2641toc.xml&DOI=10.1109/DEXA.2006.9>.
- Moldovan, D., A. Badulescu, M. Tatu, D. Antohe, et R. Girju. "Models for the Semantic Classification of Noun Phrases." Boston, MA, 2004.
- Moon, Peter. *The future of the Web as seen by its creator*, 2007. <http://www.itworld.com/Tech/4535/070709future/>.

- Musen, Mark. *Ontology classification by purpose(?)*. National Institute of Technologies, Gaithersburg, Maryland, USA, 2007.
- Nagy, M., E. Motta, et M. Vargas-Vera. "Multi-Agent Ontology Mapping with Uncertainty on the Semantic Web." Dans , 49-56, 2007.
- Naiman, Channah F., et Aris M. Ouksel. "A classification of semantic conflicts in heterogeneous database systems." Dans , 167-193. Dallas, Texas, United States: Ablex Publishing Corp., 1995.
- Noy, Natalya Fridman, et Mark A. Musen. "PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment." Dans , 450-455. AAAI Press / The MIT Press, 2000.
- Ouksel, A. M., et A. Sheth. "Semantic interoperability in global information systems." *SIGMOD Rec.* 28, no. 1 (1999): 5-12.
- Paslaru-Bontas, Elena. "A contextual approach to ontology reuse." University of Berlin, 2007. <http://www.diss.fu-berlin.de/2007/230/>.
- Patil, Lalit M. "Interoperability of formal semantics of product data among product development systems." University of Michigan, 2005.
- Paulheim, H., M. Rebstock, et J. Fengel. "Context-Sensitive Referencing for Ontology Mapping Disambiguation." Dans , 49-58, 2007.
- Petrakis, E. G. M., G. Varelas, A. Hliaoutakis, et P. Raftopoulou. "X-Similarity: Computing Semantic Similarity between Concepts from Different Ontologies." *Journal of Digital Information Management (JDIM)* 4, no. 4 (2006): 233-238.
- Porzel, Robert, Hans-Peter Zorn, Berenike Loos, et Rainer Malaka. "Towards a separation of pragmatic knowledge and contextual information." Riva del Garda, Italy, 2006.
- Poslad, Stefan, et Landong Zuo. "An Adaptive Semantic Framework to Support Multiple User Viewpoints over Multiple Databases." Dans *Advances in Semantic Media Adaptation and Personalization*, 261-284, 2008. http://dx.doi.org/10.1007/978-3-540-76361_13.
- Pratt, M.J. "Extension of ISO 10303, the STEP standard, for the exchange of procedural shape models." Dans , 317-326, 2004.
- Pratt, Michael J. "Introduction to ISO 10303---the STEP Standard for Product Data Exchange." *Journal of Computing and Information Science in Engineering* 1 (2001): 102-103.
- Qazvinian, V., H. Abolhassani, et S. H. Haeri. " Evolutionary Coincidence-Based Ontology Mapping Extraction." *Journal of Knowledge Engineering Special Issue on Advances in Ontologies Expert Systems* (2008). http://www.macs.hw.ac.uk/~hossein/expert_sys07.pdf.
- Rada, R., H. Mili, E. Bicknell, et M. Blettner. "Development and application of a metric on semantic nets." *Systems, Man and Cybernetics, IEEE Transactions on* 19 (1989): 17-30.
- Rahm, Erhard, et Philip A. Bernstein. "A survey of approaches to automatic schema matching." *The VLDB Journal* 10 (2001): 334-350.
- Ray, Steven R. "Interoperability Standards in the Semantic Web." *Journal of Computing and Information Science in Engineering* 2 (2002): 65-69.
- Rector, Alan, Natasha Noy, Nick Drummond, et Mark Musen. *Ontology Design Patterns and Problems: Practical Ontology Engineering using Protege-OWL*. Galway, Ireland, 2005. <http://www.co-ode.org/resources/tutorials/iswc2005/ISWC-Tutorial-Best-practice-2005.ppt>.

- Resnik, P. "Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language." *Journal of Artificial Intelligence* 11 (1999): 95-130.
- , "Using information content to evaluate semantic similarity in a taxonomy." Dans , 1:448-453, 1995.
- Rodríguez, M., et Max Egenhofer. "Putting Similarity Assessments into Context: Matching Functions with the User's Intended Operations." Dans *Modeling and Using Context*, 833, 1999. http://dx.doi.org/10.1007/3-540-48315-2_24.
- Rodríguez, M. A., et M. J. Egenhofer. "Comparing geospatial entity classes: an asymmetric and context-dependent similarity measure." *International Journal of Geographical Information Science* 18 (2004): 229-256.
- Rodríguez, M. Andrea, et Max J. Egenhofer. "Determining Semantic Similarity among Entity Classes from Different Ontologies." *IEEE Trans. on Knowl. and Data Eng.* 15 (2003): 442-456.
- Rodríguez, M. Andrea, Max J. Egenhofer, et Robert D. Rugg. "Assessing Semantic Similarities among Geospatial Feature Class Definitions." Dans , 189-202. Springer-Verlag, 1999.
- Rosén, J. "FEDERATED THROUGH-LIFE SUPPORT, ENABLING ONLINE INTEGRATION OF SYSTEMS WITHIN THE PLM DOMAIN." *Proceedings of the 1st Nordic Conference on Product Lifecycle Management* (2006).
- Scanlan, J., A. Rao, C. Bru, P. Hale, et R. Marsh. "DATUM Project: Cost Estimating Environment for Support of Aerospace Design Decision Making." *Journal of Aircraft* 43 (2006): 1022-1028.
- Schlenoff, Craig, Peter Denno, Rob Ivester, Don Libes, et Simon Szykman. "An analysis and approach to using existing ontological systems for applications in manufacturing" (2000): 257-270.
- Schwering, Angela. "Hybrid Model for Semantic Similarity Measurement." Dans *On the Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE*, 1449-1465, 2005. http://dx.doi.org/10.1007/11575801_32.
- Seco, N., T. Veale, et J. Hayes. "An intrinsic information content metric for semantic similarity in WordNet." Dans , 1089--1090, 2004.
- Segev, et Gal. "Putting Things in Context: A Topological Approach to Mapping Contexts to Ontologies," 2007. http://dx.doi.org/10.1007/978-3-540-74987-5_4.
- Semy, Salim K., Mary K. Pulvermacher, et Leo Obrst. *Toward the Use of an Upper Ontology for U.S. Government and U.S. Military Domains: An Evaluation*. Bedford, Massachusetts: MITRE, 2004. <http://colab.cim3.net/file/work/SICoP/resources/UpperOntologyUseLong.doc>.
- Seydoux, Florian. "Exploitation de connaissances sémantiques externes dans les représentations vectorielles en recherche documentaire." Ecole Polytechnique Fédérale de Lausanne (EPFL), 2006. http://biblion.epfl.ch/EPFL/theses/2006/3654/EPFL_TH3564.pdf.
- Shadbolt, Nigel, Tim Berners-Lee, et Wendy Hall. "The Semantic Web Revisited" (2006). <http://eprints.ecs.soton.ac.uk/12614/>.
- Shvaiko, P., et J. Euzenat. "A Survey of Schema-Based Matching Approaches." *Journal on Data Semantics IV* 4 (2005): 146-171.
- SILVA, Catarina E. FERREIRA DA. "Découverte de correspondances sémantiques entre ressources hétérogènes dans un environnement coopératif." Université Claude Bernard Lyon I, 2007.

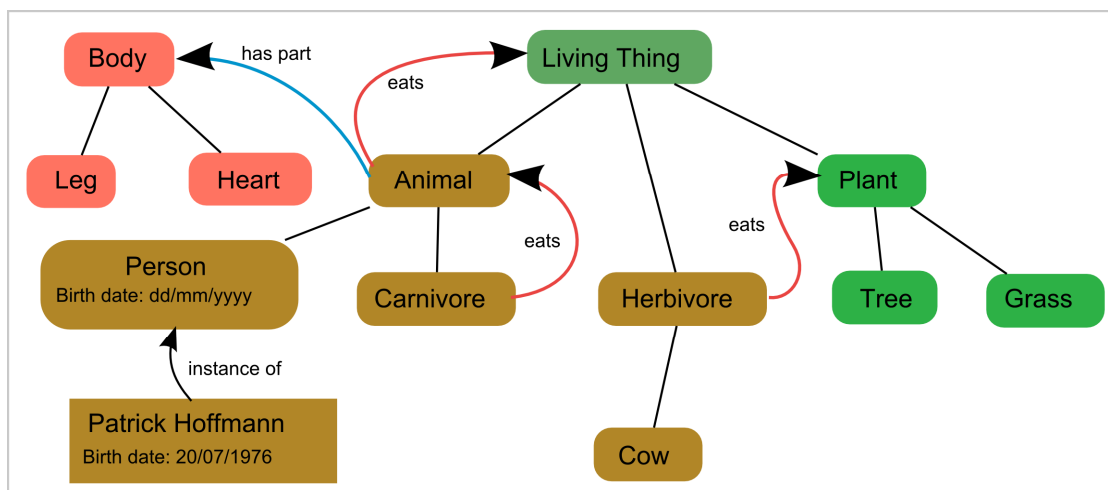
- de Souza, Kleber Xavier Sampaio, et Joseph Davis. "Aligning Ontologies and Evaluating Concept Similarities." Dans *On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE*, 1012-1029, 2004. <http://www.springerlink.com/content/lvxrln7qmvv38728>.
- Sowa, J. F. "The Challenge of Knowledge Soup." Homi Bhabha Centre, Mumbai: J. Ramadas and S. Chunawala, 2006. <http://www.jfsowa.com/pubs/challenge.pdf>.
- Sowa, John. "Processes and participants." Dans *Conceptual Structures: Knowledge Representation as Interlingua*, 1-22, 1996. http://dx.doi.org/10.1007/3-540-61534-2_1.
- Standardization, ISO - International Organization for. *STEP*, 2006. http://www.iso.org/iso/iso_cafe_step.htm.
- Straccia, Umberto, et Raphaël Troncy. "oMAP: Combining Classifiers for Aligning Automatically OWL Ontologies." Dans *Web Information Systems Engineering ? WISE 2005*, 3806/2005:133-147. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2005. http://dx.doi.org/10.1007/11581062_11.
- Strang, Thomas, et Claudia Linnhoff-Popien. *A Context Modeling Survey*, 2004. <http://elib.dlr.de/7444/>.
- Stuckenschmidt. "Toward Multi-viewpoint Reasoning with OWL Ontologies," 2006. http://dx.doi.org/10.1007/11762256_21.
- Su, Xiaomeng. "Semantic enrichment for ontology mapping." Norwegian University of Science and Technology, Trondheim, Norway, 2004.
- Su, Xiaomeng, et Jon Atle Gulla. "Semantic Enrichment for Ontology Mapping." Dans *Natural Language Processing and Information Systems*, 217-228, 2004. <http://www.springerlink.com/content/0yrgqvpv34lxbe21>.
- Su, Xiaomeng, Sari Hakkarainen, et Terje Brasethvik. "Semantic enrichment for improving systems interoperability." Dans , 1634-1641. Nicosia, Cyprus: ACM, 2004.
- Sunna, et Cruz. "Structure-Based Methods to Enhance Geospatial Ontology Alignment." Dans *GeoSpatial Semantics*, 2007. http://dx.doi.org/10.1007/978-3-540-76876-0_6.
- Tang, M. X., et J. Frazer. "A representation of context for computer supported collaborative design." *Automation in Construction* 10 (2001): 715-729.
- The role of standards in engineering and technology*. IEEE Computer Society, 2005. http://www.ieee.org/portal/cms_docs_iportals/iportals/education/setf/tutorials/baseline/glossary.html.
- Turney, P. D. "Measuring semantic similarity by latent relational analysis." *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI-05)* (2005): 1136-1141.
- "Similarity of Semantic Relations." *Computational Linguistics* 32, no. 3 (2006): 379-416.
- Tversky, A. "Features of similarity." *Psychological Review* 84, no. 4 (1977): 327-352.
- Veltman, Kim. H. "Syntactic and semantic interoperability: new approaches to knowledge and the semantic web." *NEW REVIEW OF INFORMATION NETWORKING* (2001): 159-184.
- Visser, P. R. S., D. M. Jones, T. J. M. Bench-Capon, et M. J. R. Shave. "Assessing heterogeneity by classifying ontology mismatches." *Formal Ontology in Information Systems* (1998): 148-162.

- Wache, H., T. Vogele, U. Visser, H. Stuckenschmidt, G. Schuster, H. Neumann, et coll. "Ontology-Based Integration of Information? A Survey of Existing Approaches." Dans , 108-117. Seattle, WA, 2001.
- Wu, Z., et M. Palmer. "Verb semantics and lexical selection." *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics* (1994): 133-138.
- Yétongnon, Kokou, Seksun Suwanmanee, Djamal Benslimane, et Pierre-Antoine Champin. "A web-centric semantic mediation approach for spatial information systems." *Journal of Visual Languages & Computing* 17, no. 1 (Février 2006): 1-24. doi:10.1016/j.jvlc.2005.04.002.
- ZACHMAN, JA. "A framework for information systems architecture." *IBM systems journal* 26 (1987): 276-292.
- Zha, X. F., et H. Du. "Knowledge-intensive collaborative design modeling and support Part I: Review, distributed models and framework." *Computers in Industry* 57 (2006): 39-55.
- Zhang, S., et O. Bodenreider. "Comparing associative relationships among equivalent concepts across ontologies." *Medinfo* 11, no. Pt 1 (2004): 459-466.
- Zhdanova, Anna, et Pavel Shvaiko. "Community-Driven Ontology Matching." Dans *The Semantic Web: Research and Applications*, 34-49, 2006. http://dx.doi.org/10.1007/11762256_6.
- Zhu, et Madnick. *Scalable Interoperability Through the Use of COIN Lightweight Ontology*, 2007. http://dx.doi.org/10.1007/978-3-540-75474-9_3.
- "Scalable Interoperability Through the Use of COIN Lightweight Ontology," 2007. http://dx.doi.org/10.1007/978-3-540-75474-9_3.

Appendices

Appendix A

Ontologies in practice: a basic example



In this graphical representation, *classes* are represented by their label, surrounded by a rounded rectangular box, and represent a *concept*. The relation of *subsumption*, sometimes named “*Is-a*”, is represented here by black lines, directed from the bottom to the top. It states that the class that is below is a member of the category described by (is *subsumed by*) the class above. The relation of subsumption is probably the most used in ontologies, as it allows to describe how concepts are categorized.

Along with classes, it is possible to define some specific relations between classes. In this example, we have the relation “*eats*” that allows the authors to state that an “*animal eats (some) living things*”, and an “*herbivore eats (some) plants*”. For all these classes, specific attributes can be defined, according to what the authors want to model. For the class “*person*”, you may want to define the attributes “*birth date*”, as three different values with some constraints (e.g. days between 1 and 31, month between 1 and 12, year between 0 and 2500...).

The parallelogram shape represents an individual, which must be unique, and is supposed to exist in the world represented by the conceptualization. The relation “*instance of*” points to the class of which it is an individual. In this schema, “*Patrick Hoffmann*” is considered as an individual of “*Person*”, with attribute value “*20/07/1976*”. We may suppose that this statement describes the author of this PhD thesis. But we cannot be sure until we state that clearly, e.g. by stating that this “*Patrick Hoffmann*” is the same that the one who described himself⁷⁴ with the URI <http://hoffmannp.free.fr/foaf.rdf#PH>. Also the name given to the instance does not necessarily correspond to the first name plus family name as is the case here. I could as well have named it “*hoffmannp*” or “*P. Hoffmann*”, etc.

⁷⁴ Berners Lee’s weblog, post “Give yourself a URI”, <http://dig.csail.mit.edu/breadcrumbs/node/71>

Different philosophical questions arise quickly, e.g. should Roosevelt also be represented as an individual, although he no longer exists as a person? Probably, in that case one might need to update the representation so that to add as person's attribute the date of death. Should Batman be represented? Probably not, for although the man in the street would talk of Batman as he would for a person, it does not fit in this representation, which describe the "living things".

This ontology will be described using an ontology language, in textual format. Most of current ontology languages are based on XML⁷⁵ in the likeness of the basic language of the Web, XHTML⁷⁶, so as to facilitate their exchange on the WWW.

Example adapted from the tutorial [Rector et al., 2005].

⁷⁵ Extensible Markup Language (XML), Official W3C specification for the markup language at <http://www.w3.org/XML/>

⁷⁶ Extensible HyperText Markup Language (XHTML), Official W3C specification for the markup language at <http://www.w3.org/TR/xhtml1/>

TITRE en français

Similarité sémantique inter-ontologies basée sur le contexte

RÉSUMÉ en français

Cette thèse étudie l'intérêt du contexte pour améliorer l'interopérabilité entre ontologies hétérogènes, d'une manière qui permette leur évolution indépendante. Lors de collaborations, les organisations échangent leurs données, qui sont décrites par des concepts définis dans des ontologies. L'objectif est d'obtenir un service d'évaluation de tels concepts, basé sur le contexte.

Nous proposons une méthodologie pour déterminer, modeler et utiliser le contexte. En l'appliquant, nous découvrons trois usages du contexte qui contribuent à améliorer la réconciliation d'ontologies : Nous proposons de *désambiguïser* les sens pragmatiques possibles des concepts en comparant les "perspectives" avec lesquelles les concepts ont été développés ; de *personnaliser* en considérant le contexte des agents, constitué d'une sélection pertinente parmi les domaines et tâches de l'organisation ; d'*évaluer* la pertinence des données associées au concept pour la tâche qui a suscité le besoin en interopérabilité.

RÉSUMÉ en anglais

This thesis studies how a context-based approach could improve interoperability between heterogeneous ontologies developed and evolving autonomously. The hypothesis is that during collaborations, organizations exchange their data whose meaning is described by concepts defined by ontologies. The objective is to propose a context-based evaluation of such concepts.

We propose a methodology to determine, model and use the context. Applying this methodology, we uncover three usages of context that may be used together to improve ontology reconciliation: We propose to *disambiguate* among the possible pragmatic meanings of the concepts by comparing the "perspectives" with or for which the concepts have been developed; to *personalize* this comparison by considering the agent's context, made of a relevant selection of the agent's company domains and tasks; to *evaluate* the pertinence of the data associated with the concept for the task that triggered the interoperability need.

DISCIPLINE

Informatique (CNU 27)

MOTS-CLÉS

Ontologies, ontology matching, mapping, contexte, point de vue, similarité sémantique

INTITULÉ ET ADRESSE DE L'U.F.R. OU DU LABORATOIRE :

UFR Informatique
Université Claude Bernard Lyon 1
43, Boulevard du 11 novembre 1918
69622 Villeurbanne Cedex