



# OMWG D7.4: Mapping Discovery Requirements

DERI OMWG Working Draft 28/02/05

**This version:**

<http://www.omwg.org/TR/d7/d7.4/v0.1/20060228/>

**Latest version:**

<http://www.omwg.org/TR/d7/d7.4/v0.1/>

**Previous version:**

<http://www.omwg.org/TR/d7/d7.4/v0.1/20060228/>

**Authors:**

Fancois Scharffe

**Editors:**

Francois Scharffe, Atanas Kiryakov

Copyright © 2004 [DERI](#)®, All Rights Reserved. [DERI](#) liability, trademark, document use, and software licensing rules apply.

this document is available as a non normative [PDF](#) version

---

## Table of contents

### [1 Introduction](#)

### [2 Syntactic Mapping Discovery](#)

#### [2.1 Pre-processing](#)

#### [2.2 String Distance Techniques](#)

### [3 Semantic Mapping Discovery](#)

#### [3.1 Lexical Semantics](#)

##### [3.1.1 Using a Dictionary or Thesaurus](#)

##### [3.1.2 Using the Web](#)

##### [3.1.3 Using a Corpus of Documents](#)

#### [3.2 Structure Based Techniques](#)

#### [3.3 Hybrid Techniques](#)

### [4 Implementation Priority List](#)

### [5 Global Architecture](#)

## [6 Conclusion](#)

## [7 Acknowledgements](#)

## [References](#)

---

# 1 Introduction

Ontology Mapping Discovery helps the user in finding correspondences between ontologies using different techniques issued from Natural Language processing and artificial intelligence. The algorithms detailed in the present document have as a goal to discover similarities between the concepts, the relations or the instances of the two ontologies. The mapping tool then suggests the discovered similarities to the user that may validate them. We don't believe fully automatic ontology mapping is possible as the contextualized representations of a same domain might hardly be aligned, even for a human. We will in this document describe the different techniques required to maximize the efficiency of the mapping discovery process.

## 2 Syntactic Mapping Discovery

The more evident method to relating two entities is to look at the label describing them. This label generally consists of one term that has to be compared with the labels of the other ontology. Syntactical analysis of the terms might give some good results when they have been properly pre-processed.

### 2.1 Pre-processing

This phase takes place before comparing the strings corresponding to the labels of the entities to be aligned. Depending on the type of entities worked on, different methods are relevant.

Relations and attributes are often labeled by using more than one term generally having a verb ("is-a", "hasParent", "hasNrOfChildren", ...). Concepts are generally labeled using one noun ("car" for example) but can also be the concatenation of many nouns, this being the easy way of realizing specialization the term to reflect the concept-hierarchy (for example "ConvertibleSportCar"). Comparison of these labels require extraction of the substrings it contains. We can for this purpose use Stop-Characters lists and Case sensitive string splitting. This phase returns a set of strings for each label.

Once the list of terms corresponding to each label obtained, it might be useful to remove the morphological difference between similar terms (compare for example "computing" and "computation"). This task known as stemming will return the 'root' of each term to allow further processing ("comput" in our example). Different algorithms are doing the stemming of English words using a set of rules corresponding to the words termination [[Porter1980](#)]. A survey of

stemming algorithms is available in [\[Hull1996\]](#).

## 2.2 String Distance Techniques

String distance or string similarity algorithms take as an input two character strings and return a value indicating the distance or similarity between them. The notion of distance varies depending on which algorithm is used. Applied to ontology mediation, the string distance techniques are rather weak as they disregard the terms semantics. They are however a fast way to compare two labels without needing preprocessing (see above). Among the large number of algorithms available we may cite the Hamming Distance [\[Hamming1980\]](#), that measures the number of dissimilar bits encoding the string. Another popular techniques more particular to strings is the Levenstein Distance [\[Levenstein1966\]](#). It measures the number of operations (move, delete or insert a character) necessary to translate one string into the other. The Jaro-Winkler string distance technique, originally described in [\[Jaro1989\]](#) is based on the number and order of common characters in two strings. These methods are among others, an extensive survey of string distance reviewing the optimized versions of these methods is available in [\[Cohen2003\]](#).

## 3 Semantic Mapping Discovery

### 3.1 Lexical Semantics

The techniques reviewed in the precedent section are efficient but need to be completed. Two terms may be similar even if they are completely differently spelt. This is the example of synonyms. More generally, two terms having a **related sense** deserve to be somehow related. By related sense we mean that the two terms are either synonyms or near-synonyms, that one term is describing a more general/specific fact or object than the other does, we will in this case say that the general term is an hypernym of the more specific term, which is itself an hyponym of the more general one. Terms might also be related when one describe a part of the other. Like a wheel is part of a car the terms "wheel" and "car" are related. This relation is called meronymy.

#### 3.1.1 Using a Dictionary or Thesaurus

In order to be able to capture these relations between the terms, it is necessary to get their semantics. For this purpose different tools are available. The now classical tool is Wordnet, the reference electronic thesaurus [\[Fellbaum1998\]](#). This general thesarus gives relations between sets of synonyms for the English language. It contains around 150,000 words organized in over 115,000 synsets for a total of 203,000 word-sense pairs. As it is freely available it is widely used in Natural Language Processing research that needs to exploit relations between words. It has however some limitations we can already mention. It contains some errors. The definitions of the synsets are sometimes not completely true and the synset themeselves do sometimes contain terms which can be seen as not being synonyms in a particular context. Moreover,

specific domains of knowledge are too much or not enough covered. Most of these problems are due to the fact that wordnet being wide and general, is not taking into account domain specific information. This will restrict our use since we plan to deal with specific domains of knowledge.

Electronic dictionaries are also a well-studied resource of information contained in the definitions structure. Parsing definitions or applying information extraction techniques to the implicit network of words give good results to evaluate the terms relatedness (how related two words are). [\[Chodorow1985\]](#) and [\[Jannink1999\]](#) propose methods to build a thesaurus out of a dictionary. However, the dictionaries are often not freely available in an electronic version, especially were they are about a specific domain of knowledge.

### 3.1.2 Using the Web

The Internet as the biggest database on the earth definitely contains terminological information. A number of approaches have been proposed in order to extract the meaning of terms on the internet [\[Fuji2000\]](#), or more recently to find out the relatedness between two terms using Google queries [\[Vitanyi2005\]](#). This last method is particularly interesting as it allows to use the web to retrieve terms relatedness. It is based on the principle that a Google query returns the number of pages associated to the query string. When querying for two different terms one can consider the number of pages resulting from the three queries consisting of one term, the other, and both. Measuring the difference between the number of pages give a measure of terms relatedness. Given a particular domain, this method could easily be extended by adding contextual information to the queries in order to remove pages that are not related to a particular domain.

### 3.1.3 Using a Corpus of Documents

A corpus of domain related documents contains information about this domain. A number of methods have been proposed to extract term definitions and concept hierarchies from large corpora, see [\[Jiang1997\]](#), [\[Faure1998\]](#). This reduces our problem of measuring the similarity between two terms to the proble of the lenght of a path in a concept hierarchy (which is trivial). These methods are generally complex to establish and time consuming because a domain specific corpus has to be constituted.

From the different methods seen in this section we keep in mind that the time spent on executing the method shouldn't be longer than the time spent to manually specify the terms correspondences. Web queries, a general thesaurus or domain specific dictionaries or corpuses are presenting different aspects and that should be taken into account when choosing the mapping approach. We will see in the next section how the structure of the ontology may also assist us in automatically finding ontology mappings.

## 3.2 Structure Based Techniques

This section presents how the structure of the ontologies, ie the graph representing the concept hierarchy, helps in finding similarities between ontologies.

Graph matching is a classical research area in mathematics. A recent algorithms propose to match two graphs given measures of similarity between the nodes of these graphs [Melnik2002]. It takes as a heuristic that two nodes are similar if their neighbours are also similar. An algorithm propagate the measures of similarity trough the graph until a stable state is reached.

### 3.3 Hybrid Techniques

Hybrid techniques are combining linguistic and structure analysis in order to come with a more complete mapping. A performant ontology mapping tool must uses hybrid techniques to take advantage of both approaches. Different linguistic analysis might be used. It is for example possible to measure the relatedness of two terms using both a dictionary and web search queries. In this case, the results of the algorithms specifying entities similarity have to be combined in some way.

## 4 Implementation Priority List

For the implementation we have distinguished three phases:

- Version 1
- Version 2
- Version 3

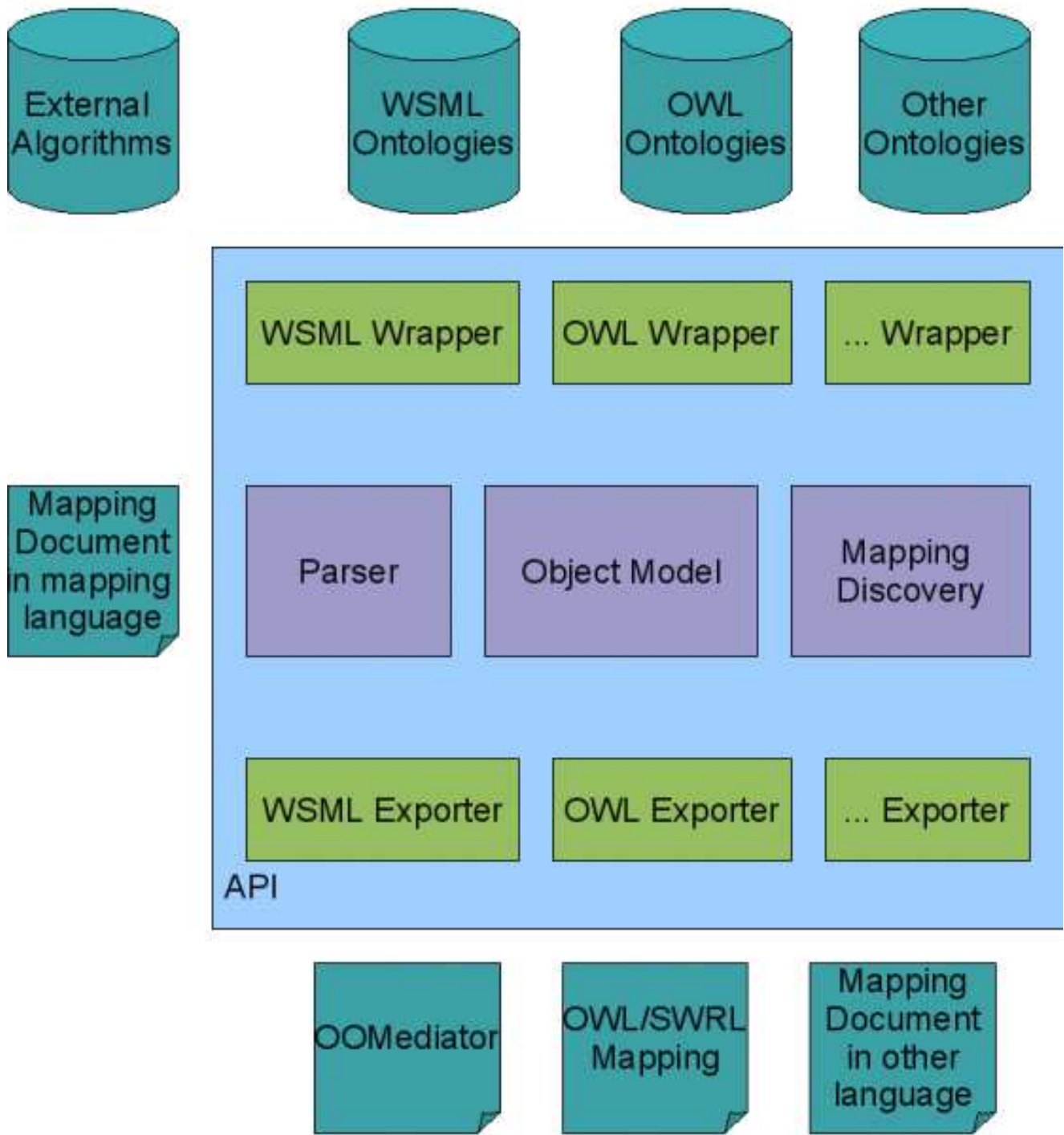
The Vs indicate in which phase which requirement is being initially tackled.

Req. ID.	Mapping tool Requirements	Version 1	Version 2	Version 3	Priority
V1	Interoperability/Compatibility	V			(affects all implementation)
V2	Genericity	V			(affects all implementation)
V3	String Edit Distance	V			HIGHEST
V4	Stemming		V		HIGH
V5	Web Search based Semantic Distance	V			HIGHEST
V6	Dictionary-Based Semantic Distance			V	LOW
V7	Corpus-Based Semantic Distance		V		HIGH

V8	Structure Matching technique		V		HIGH
V10	Hybrid Techniques			V	HIGH

## 5 Global architecture

Here is the updated architecture diagram of the OMWG mapping API, the Mapping discovery module is the one we deal with in this document.



## 6 Conclusion

We have seen different techniques used to find correspondences between ontologies. In the next deliverables D7.5 and D7.6, we will present into more details the choices we have made for our tool. We will also relate the mapping discovery module to the other module of the OMWG mapping API.

## 7 Acknowledgement

The work is funded by the European Commission under the projects DIP, Knowledge Web, Ontoweb, SEKT, TSC by Science Foundation Ireland under the DERI-Lion project; and by the Vienna city government under the CoOperate programme.

The authors would like to thank to all the [members of the OMWG working group](#) for their advices and inputs to this document.

---

## References

- [Porter1980]**Porter, M.F. (1980) An Algorithm for Suffix Stripping, Program, 14(3): 130-137
- [Hull1996]**Hull, D.A. & Grefenstette, G. (1996) A Detailed Analysis of English Stemming Algorithms, Xerox Technical Report
- [Hamming1980]**Hamming R.W., Coding and Information Theory, Englewood Cliffs, NJ, Prentice-Hall, (1980).
- [Jaro1989]**Jaro, M. A. 1989 "Advances in record linking methodology as applied to the 1985 census of Tampa Florida". Journal of the American Statistical Society 64:1183-1210
- [Cohen2003]**W. Cohen, P. Ravikumar, and S. Fienberg. A comparison of string metrics for matching names and records. Proceedings of KDD Data Cleaning Workshop, Aug 2003.
- [A1-Halami1998]**R. A1-Halami, R. Berwick, et. al. Christiane Fellbaum editor, "WordNet, an electronic lexical database"; Bradford Books. May 1998, ISBN 0-262-06197-X
- [Atkins1986]**Atkins, B. T., J. Kegl, et al. (1986). "Explicit and implicit information in dictionaries." Lexicon Project Working Papers 12.
- [Levenstein1966]**Levenstein A., Binary Codes Capable of Correcting Deletions, Insertions and Reversals Soviet Physics-Doklady, vol. 10, 1966.
- [Jannink1999]**Jan Jannink and Gio Wiederhold, "Thesaurus Entry Extraction from an On-line Dictionary", Proceedings of Fusion 1999, Sunnyvale CA, 1999.
- [Chodorow1985]**Martin Chodorow, Roy J. Byrd and George E. Heidorn, "Extracting Semantic Hierarchies from a Large On-Line Dictionary", ACL 1985, pages 299-304
- [Fujii2000]**Fujii, Atsushi and Tetsuya Ishikawa. 2000. Utilizing the world wide web as an encyclopedia: Extracting term descriptions from semi-structured text. In Proc. 38th Meeting of the ACL, pages 488-495, Hong Kong, October.

**[Vitanyi2005]**P.M.B. Vitanyi, Universal Similarity, Proc. ITW2005 - IEEE ITSOC Information Theory Workshop 2005 on Coding and Complexity, 29th Aug. - 1st Sept., 2005, Rotorua, New Zealand.

**[Jiang1997]**Jiang, J. and D. Conrath. "Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy". International Conference on Lexical Linguistics (ROCLING X), 1997. Taiwan.

**[Faure1998]**Faure, D. & Nedellec, C. (1998). A corpus-based conceptual clustering method for verb frames and ontology acquisition. In LREC workshop on adapting lexical and corpus resources to sublanguages and applications, Granada, Spain.

**[Melnik2002]**S Melnik, H Garcia-Molina, E Rahm. "Similarity flooding: a versatile graph matching algorithm and its application to schema matching". In Proceedings of the 18th ICDE Conference. 2002.

**[Rahm2001]**E. Rahm, P. Bernstein: A Survey of Approaches to Automatic Schema Matching The VLDB Journal, 2001.



\$Date: 2004/10/22 16:12:55 \$

[webmaster](#)