An Ontology-Driven Sociomedical Web 3.0 Framework

Jinan El-Hachem

A Thesis
In The Department of
Computer Science and Software Engineering

Presented in Partial Fulfillment of the Requirements
For the Degree of
Doctor of Philosophy in Computer Science
Concordia University
Montréal, Québec, Canada

April 2014

CONCORDIA UNIVERSITY
SCHOOL OF GRADUATE STUDIES

This is to certify that the thesis prepared

By : **Jinan El-Hachem**

Entitled : **An Ontology-Driven Web 3.0 Sociomedical Framework**

and submitted in partial fulfilment of the requirements for the degree of

Doctor of Philosophy in Computer Science

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee :

____________________________________ Chair
Dr. Otmane Ait Mohamed

____________________________________ External Examiner
Dr. Michel Gagnon

____________________________________ External to Program
Dr. Lingyu Wang

____________________________________ Examiner
Dr. Nematollaah Shiri

____________________________________ Examiner
Dr. Jürgen Rilling

____________________________________ Thesis Supervisor
Dr. Volker Haarslev

Approved by ____________________________________
Dr. Volker Haarslev Graduate Program Director

April 3rd, 2014 . ____________________________________
Dr. Christopher W. Trueman, Interim Dean
Faculty of Engineering and Computer Science

Abstract

An Ontology-Driven Web 3.0 Sociomedical Framework
Jinan El-Hachem, PhD.
Concordia University, 2014

Web 3.0, the web of social and semantic cooperation, calls for a methodological multidisciplinary architecture in order to reach its mainstream objectives. With the lack of such an architecture and the reliance of existing efforts on lightweight semantics and RDF graphs, this thesis proposes “Web3.OWL”, an ontology-driven framework towards a Web 3.0 knowledge architecture.

Meanwhile, the online social parenting data and their corresponding websites users known as “mommy bloggers” undergo one of the fastest online demographics growth, and the available literature reflects the very little attention this growth has so far been given and the various deficiencies the parenting domain suffers from; these deficiencies all fall under the umbrella of the scarcity of parenting sociomedical analysis and decision-support systems.

The Web3.OWL framework puts forward an approach that relies on the Meta-Object Facility for Semantics standard (SMOF) for the management of its modeled OWL (Web Ontology Language) expressive domain ontologies on the one hand, and the coordination of its various underlined Web 3.0 prerequisite disciplines on the other.

Setting off with a holistic portrayal of Web3.OWL’s components and workflow, the thesis progresses into a more analytic exploration of its main paradigms. Out of its different ontology-aware paradigms are notably highlighted both its methodology for expressiveness handling through modularization and projection techniques and algorithms, and its facilities for tagging inference, suggestion and processing.

Web3.OWL, albeit generic by conception, proves its efficiency in solving the deficiencies and meeting the requirements of the sociomedical domain of interest. Its conceived ontology for parenting analysis and surveillance, baptised “ParOnt”, strongly contributes to the backbone metamodel and the various constituents of this ontology-driven framework.

Accordingly, as the workflow revolves around Description Logics principles, OWL 2 profiles along with standard and beyond-standard reasoning techniques, conducted experiments and competency questions are illustrated, thus establishing the required Web 3.0 outcomes. The empirical results of the diverse preliminary decision-support and recommendation services targeting parenting public awareness, orientation and education do ascertain, in conclusion, the value and potentials of the proposed conceptual framework.

Dedication

To my son Ryan,
and to each and every person with whom,
through one way or another,
I shared this fascinating experience...

Acknowledgments

I would like to use this area in order to look back at the past few years of doctoral studies of my life, and think about "the contributions"...
For once in this document, the word "contributions" will not refer to those the thesis endeavors to establish (and yes *that* is a relief). Instead, it refers to those *I* have been granted by those same years. I thus translate them below into thoughts of heartfelt thankfulness and appreciation.

First and foremost, I will never have enough words to express my respect and gratitude to Dr. Haarslev, for all his support, guidance, and encouragement. It has been both an honor and a pleasure to have him as my supervisor. His confidence in my efforts yielded an invaluable motivation. His optimism and enthusiasm for research formed a major source of inspiration, most notably during the tough times of my PhD pursuit. I hope this thesis is nothing but the beginning of a journey full of achievements led by his vision and insightfulness.

I am very appreciative of my examining committee's time, comments and constructive feedback. I transmit my sincerest esteem to each of its members.

I had the good fortune to work and collaborate with very talented and committed people at the McGill Clinical and Health Informatics (MCHI), thanks to Dr. Arash Shaban-Nejad. I am aware of the extent to which that teamwork boosted the interdisciplinary aspects of my research and am looking forward to further renewed cooperation.

To all my friends, lab mates, colleagues and groups that became a part of my Concordia life (especially to Muna Al-Khayat, Christine Kehyayan, Jocelyn Faddoul, Laleh Roosta Pour, Mina Aslani, Ming Zuo, Kejia Wu, Mina Kazemi and Jelena Vlasenko), I was exceptionally lucky I got the chance of enjoying with them memorable times that shall forever be graved in my heart; I wish them all continued success.

I dedicate this thesis to my beloved mom and dad, my sister and soulmate Layal, my cherished brother Jad, and my treasured aunts and uncles who raised me and are to me as dear as my parents. I extend my best wishes, and hope the years to come will bring us all health, happiness and prosperity.

It is now the perfect occasion to put into words my affection and appreciation to my husband Walid; this work is chiefly the outcome of his genuine interest and endless devotion... In contrast, I candidly refrain from making a similar statement when it comes to my son Ryan, yet I solemnly declare that his birth, even though it took place in the midst of my PhD candidacy, was and will always be by far the most precious gift of my life.

My thoughts and considerations at this momentous time lead me to old friends, family members and important people in my life. I am really thankful, for instance, for the support of my Masters thesis supervisor Dr. Marie Khair, and that of my Mathematics professor, Dr. Ajaj Tarabay.

Looking back at my situation five years ago, I can clearly see how big and meaningful the “PhD dream” was to me. Yet, had it not been for the backing of significant people in my life, I doubt I would have had the necessary confidence and strength to drastically change paths, resign from my position and move to Montreal. In that regard, I seize the opportunity to convey my earnest gratefulness to this exquisite city that allowed me to realize personal and professional contributions. I am full of hope that the best is yet to come...

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivations

In parallel with the Semantic Web's extremely active research community, a continuous and exceptionally rising propagation of the Social Web is witnessed. A remarkable advancement can be made if a proper methodology for maximizing the cooperation between the two webs can be set. Such a methodology should highly encourage the first web to bring in its theories and formalisms to the second, in exchange for some of the latter's popularity and proliferation, yielding the new generation of the web, nowadays known as Web 3.0.

An amplified fusion between the Social and the Semantic Webs is indeed a strongly beneficial achievement to both disciplines. It shall solve the foremost problems undergone by each of them, producing Web 3.0, an outcome that by far surpasses the sum of its individual components. Among the strengths and characteristics of this web are endorsed automation, standardization and interoperability, with efficient information extraction, querying and aggregations. In Web 3.0 for instance, valuable large data sets from the abundant Social Networking (SN) Web 2.0 sites feed the Semantic Web applications. As a consequence, these sites benefit from the Semantic Web applications to generate semantically-rich data, and an overall network effect of the henceforth strongly formalized Social Web gets reflected on the Semantic Web, boosting its formerly limited usage.

Meanwhile, the fundamental principles behind the Model Driven Architecture (MDA) fall under the umbrella of elaborated platform-independent machine understandable domain models. These models are subject to transformations before reaching their target platform-specific applications. They hereby meet with the Semantic Web and its call for machine understandable methods for publishing information, with its conceptual models consisting of ontologies.

In view of these motivating factors on the one hand, and of the potentials that can be achieved by conceiving an ontology-driven approach that stems from the Model Driven Architecture on the other, this thesis looks into the state of the art of ontology-based Web 3.0 efforts.
It tackles the relevant implied issues from several angles, identifying their main limitations, and the potential areas that can be subject to improvement.

Summarized in Table 1.1 and later described in the coming subsections, the examined perspectives fall under three angles or categories:

- The Semantic Web and the need for ontologies and expressiveness in Web 3.0: how well and to what extent are Web 3.0 endeavors taking advantage of ontologies and their expressiveness?
- The Web 3.0 social parenting data management: what are the sociomedical Web 3.0 accomplishments and what is the role of an ontology within these accomplishments?
- The metamodeling standards and Web 3.0: what if the resolution was to adopt an ontology-driven Web 3.0 approach? Will the existing underlined standards be adequate and sufficient to consolidate such an adoption process?

The Need for Ontologies and Semantic Web Expressiveness in Web 3.0	
Great reliance on RDF & RDF Graphs	Promoting expressive Semantic Web for Web 3.0
Lack in efforts involving OWL and formal semantics	Overcoming RDF deficiencies with OWL-based ontologies and scenarios
High cost of expressiveness and performance degradation	Anticipating strategies to deal with high expressiveness performance costs
Lack of formal semantics and expressiveness exploitation in Semantic Tagging	Proposing OWL-based techniques for tagging management and suggestion
Limitations in expressive ontology-aware NLP	Reinforcing workarounds for ontology-aware NLP limitations through tagging
The Sociomedical Web 3.0	
Scarcity in decision-support systems & consolidated Knowledge Bases	Proposing ontology-driven decision-support sociomedical systems
Conflicting/Confusing conveyed information	Transforming weaknesses into windows of opportunities
Non-structured & non-homogeneously distributed sources	Setting the grounds for a strengthened knowledge base for sociomedical aspects
The Metamodeling Standards for Web 3.0	
No conformance for meta-language interoperability (RDF, OWL, etc.)	Proposing SMOF extensions for conformance with Meta-language interoperability
No conformance for logical sub-languages	Proposing SMOF extensions for logical sub-languages (OWL 2 DL, EL, QL, RL) interoperability
No means for ontology modularity, matching and alignment	Proposing SMOF extensions for ontology modularity, matching & alignment

Table 1.1: Summary of Problems (to the left) and Objectives (to the right)

1.1.1 Problem Statement

Despite huge efforts in the area of the Semantic Web to achieve standardization and reach an advanced level of machine-processable web data, a relatively limited number of applications that can take full advantage of these data and efforts has been developed. Below is an overview of the main observations on limitations and improvement-susceptible areas according to each of the above listed angles.

From the Web 3.0 Semantic Web Expressiveness perspective:

By delving into the Semantic Web's main achievements for Social Networking, this research notes the following:

- A great reliance on RDF (Resource Description Framework) graph data for the majority of Semantic Web realizations for SNs, RDF manifesting major constraints and limitations at the semantic and formal representation levels.
- A lack in efforts involving the Semantic Web's advanced findings and relatively complicated vocabularies and grammars (ontological syntax), particularly in the endeavors pertaining to OWL 2 (Web Ontology Language) novelties. In short, Web 3.0 uses a very small OWL subset.
- Tagging being a very fruitful activity bestowed by Web 2.0, the Semantic tagging Web 3.0 computation and recommendation endeavors still take limited advantage of expressiveness and standardization advances.
- Data mining raised concerns as to the applicability of "expressive" ontology-aware Natural Language Processing (NLP) for Web 3.0; major limitations and constraints related to the high complexity and low accuracy of NLP algorithms that are based on expressive ontology features are recognized.

With the predominance of RDF over most social semantic efforts, the different endeavors remain restrained by RDF's known loopholes and weaknesses. These can be summarized in limited expressivity, loose semantics, lack of standardization, and therefore of known supporting reasoners [2].

A more comprehensive review of the overall Semantic Web realizations for Social Networks reveals them to be mainly relying on RDF. While limited OWL 1 constructs usage is reported, no explicit and formalized OWL 2 vocabularies and enhancements are disclosed.

In [46], a highly cited reference for Web 3.0, it is explicitly stated that Web 3.0 uses only a very small OWL subset; [66] contains a selection of the major Web 3.0

papers that report advances in research and technology showing how “a basic level of formal semantics” is being applied to the infrastructure and pages on the web.

The thesis outlines expressivity and issues encountered with Social Networks (SN) data representation methods, and highlights constructs’ wrong usage and mistreatment.

The global tendency towards low expressivity by avoiding complex Description Logics (DL) constructs and highly expressive languages is justified by arguments claiming that low level semantics are amply adequate and furthermore favorable for a wider adoption of the Social Semantic Web. Since the vast majority of algorithms rely on very modest formal semantics for graph pattern detection, RDF, based on its graph-oriented nature, was deemed sufficient for expressing Social Web data semantics, and correspondingly SPARQL (SPARQL Protocol and RDF Query Language) for querying it.

In short, low expressivity and RDF-based aspects still stamp most recent achievements of renowned authors of the field, and of many others alike, [10, 37, 45] for instance.

Chapter 2, along with specifically relevant sections of following chapters will provide more extensive overviews in this regard.

From the Web 3.0 Social Parenting Data Management Perspective

Parenting websites known as “mommy blogging sites” are among the most outspread and openly accessible media for Social Networking. Their increasing volume and velocity reflect the extent to which social media affects parents and mommy bloggers. A massive blogosphere with thousands of such sites, millions of members, posts and replies denote how much common it is becoming for users to seek health information and parenting advice from the Internet.

While the challenging aspects of the parenting domain with its heterogeneous multidimensional components are indisputable, different studies and surveys report the drawbacks and deficiencies of its actual social media management.

On the one hand, as child rearing depends on the interaction between various factors including economic, intellectual, cultural, and social ones, information and parenting advice conveyed by social media often turns out to be conflicting and confusing. On the other hand, sources of reliable information are not easily accessible and tend to be difficult to evaluate, especially due to their chaotic unstructured nature.

In the same context, the available informational non-structured data is not homogeneously distributed, as parents with adolescents and practitioners working with these parents receive less support and information than those with younger children.

Meanwhile, the different approaches to manage parenting data and respond to sociomedical needs are short of a strengthened consolidated knowledge base (KB), one that comprises ontology-supported analysis of the domain, whether it was RDF, OWL or any of their sub-languages and derivatives featuring as the exploited ontological support languages.

The next chapter will comprise the references that support these claims, including an overview of the state of the art parenting and sociomedical projects.

From the Web 3.0 Metamodeling Standardization Perspective

By seeking standards and formalisms for machine understandable data publication, the Semantic Web meets with the Model Driven Architecture, itself an architecture that highly promotes and depends on standardized models for documenting and expressing domains.

While investigating the feasibility of proposing an ontology-driven Web 3.0 framework that arises from the Model Driven Architecture, this research is faced with shortcomings of the most relevant MDA adopted standards for metamodeling: the Meta-Object Facility (MOF).

Luckily, some of these shortcomings get covered with Support for Semantic Structures (SMOF), an extension to MOF. This extension's adoption process started in September 2010, its formalization however did not take place until recently [72].

Nevertheless, this extension still lacks some requirements in order for it to become compliant with the expectations of the proposed framework, and very freshly in August 2013, the Object Management Group (OMG, the group behind all these standardizations and formalizations), issued an RFP (Request for Proposal) inviting potential contributors to submit proposals for a framework that shall be called OntolOp, and shall consist of extensions to SMOF for the handling of specific needs overlooked in its current version [73].

Some of the expected extensions that are crucial for the immediate needs of the proposed framework are mentioned herein:

- Providing a meta language for interoperability among Semantic Web languages (OWL, RDF), with translations between different logical languages without changing the original formalisms and specifications of any of those languages.
- Establishing means of conformance for the logical languages: OWL 2 (along with its Existential, Rule and Query Language profiles EL, RL and QL), RDF, RIF (Rule Interchange Format), etc.
- Setting up the means for expressing modularity operations and relations, as well as for matching and alignment of different ontologies covering one domain.

The coming chapters will gradually provide detailed enlightments on the above metamodeling requirements. For the time being, the fact that the current version of the metamodel disregards them is undoubtedly a setback for an ontology-driven framework that relies on them.

1.1.2 Thesis Objectives

The aim of this thesis is to contribute to the efforts deployed for Web 3.0 by suggesting an ontology-driven conceptual framework that would solve, or in the worst case find workarounds to overcome the above mentioned problems and deficiencies.

This thesis shall convey its persuasion that resides in the fact that an ontology-driven framework is indeed the most appropriate framework for the different specifications and requirements of Web 3.0, and most particularly for the deficiencies of the parenting sociomedical domain.

More explicitly, if the different perspectives initiated in the previous subsection are to be considered, naturally, the underlined objectives are self-explanatory; nevertheless, below are further incentives and clarifications:

- **From the Web 3.0 Semantic Web need for ontologies and expressiveness perspective:**

  Given the introduced RDF limitations and the so far stated facts describing how current social data representations are still deficient in the ontological formalisms and robust expressiveness provided by OWL, enhancing applications by OWL 2 ontologies presents different advantages:

  - It promotes means for unambiguously specifying the meaning of data in the application.
  - It provides background knowledge and vocabularies necessary for the formulation of accurate queries by users (for instance SN users, and mommy bloggers in particular in the case of the proposed framework).
  - It offers entailment-based enriched query answers that unveil information not explicitly represented in the dataset.

  While it is true that a limited level of expressivity will be compensated for with efficient performance, the claim (in [15]) according to which restricted constructs are sufficiently adequate for SN applications is somehow unconvincing. Since there exists no restricted and final set of a priori defined Semantic Social Web applications and requirements, added expressivity surely goes hand in hand with added value and potentials. These potentials remain prevalent at different levels of Web 3.0 underlined activities, like tagging for example.

  To attain more powerful reasoning, meaningful and structured data represented expressively should be provided. Still, when faced with complexity constraints leading to high performance costs, expressiveness management techniques and workarounds are expected to be offered. Overall, promoting higher expressiveness through suggested strategies to deal with its costs is among the main objectives of the thesis. The underlined undertakings do not ignore the state of the art achievements and the so far available Web 3.0 standards. They rather build upon them through an approach that guarantees interoperability with the existing efforts, and these efforts' evolution as well.

- **From the Web 3.0 Social Parenting Data Management perspective:** Transforming the parenting domain of interest's significant weaknesses into windows of opportunity and effectively collecting knowledge to take advantage of the positive impact SN media should have on parents and parenting. This includes building an OWL expressive parenting ontology as part of a strong knowledge base. Further ontological useful knowledge can be imported and similarly managed in this knowledge base, through ontology modularization for instance. The different endeavors are to be carried out with an emphasis on knowledge consolidation while establishing consensus based on research and reliable key findings.

- **From the Web 3.0 Metamodeling Standardization perspective:** It is important to make it clear that proposing the OntolOp framework that responds to OMG's RFP is not among the objectives of the thesis. The objective on the other hand is the establishment of the emphasized SMOF extensions that arise as vital requirements for the present framework. These extensions will make it possible for the proposed framework to benefit from visual modeling of ontologies and from different OMG relevant interoperability standards. They allow the conception of a coherent framework that, all along its various workflow and architectural components, is driven by ontologies. That way, research convictions underpinning the fact that an ontology-driven framework is indeed a mostly convenient Web 3.0 framework are sustained.

1.2 Contributions

In its simplest definition, a framework is a basic conceptional structure. Led by the introduced motivational elements, this thesis proposes “Web3.OWL”, an ontology-driven Web 3.0 sociomedical framework. Web3.OWL surpasses traditional research design by exploring many disciplines and sectors based on standardized metamodel foundations reinforced with called upon extensions.

To achieve the desired Web 3.0 outcomes and decision-support services, the framework exploits advances in ontology domain design and language expressiveness, namely those related to OWL, and in standard and beyond standard DL and OWL reasoning. These facts justify the reason for which it was baptized “Web3.OWL”.

As Figure 1-1 lists the elements forming the proposed contribution, it should be emphasized that the framework, while being driven by ontologies and semantic web concepts, is multidisciplinary and relies on other fields’ well-established technologies. Nevertheless, exploitation of those fields and disciplines takes place through semantic ontology-aware strategies.

Figure 1-1 presents a high-level plot of the components and activities in the conceptual framework. While different workflow modes that apply the proper sequencing of the framework’s components illustrated in this figure will be provided later on, the highlighted darker blocks represent the areas where the contribution is mostly underlined. As for the rest of the blocks, they typically consist of a mere reuse of existing well-established frameworks and technologies.

The aim of Figure 1-1 is not to describe these components and activities and clarify their interactions, but to simply identify and list them. The illustration situates the knowledge base and ontological activities in the center of the framework, the rest of the components and activities pivoting around them. This is done on purpose to

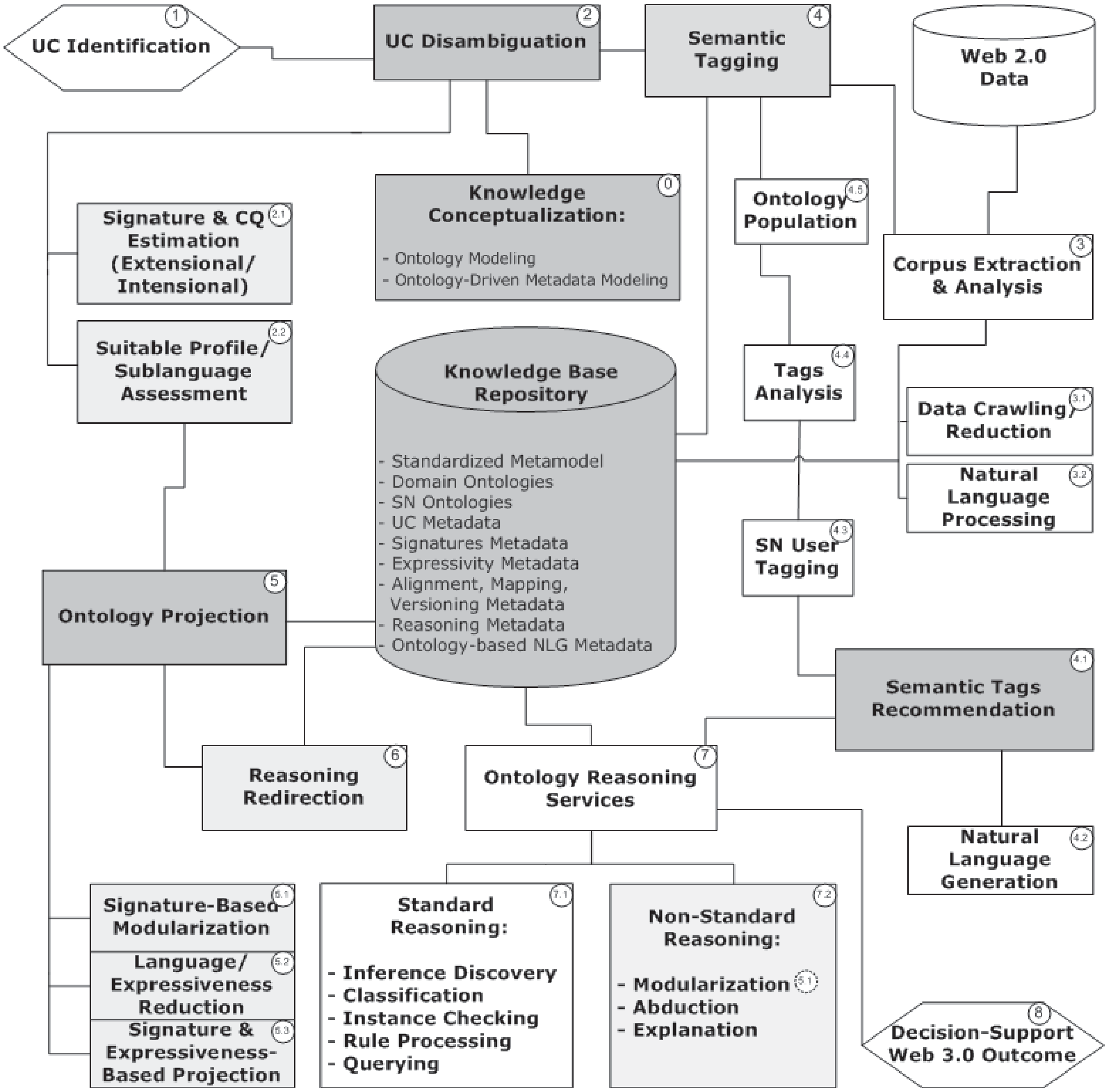


Figure 1-1: Proposed Ontology-Driven Framework -
Knowledge Base and Surrounding Web 3.0 Collaboration Model
Darker blocks emphasize analytic parts of the contribution;
UC = Use Case; CQ = Competency Question;
Rounded numbers to the right represent the conceptual hierarchy,
different process flows are possible;

point out the proposed ontology-driven architectural aspects.

To convey a basic level of their hierarchical and sequential occurrence, blocks have been assigned unique reference numbers.

The following items present an interpretation of this figure along with definitions of its introduced terminological concepts, when needed:

- Knowledge conceptualization (0) is an essential preliminary step consisting of the ontological modeling of concepts, properties, axioms, instances and rules, and ontology-driven metadata modeling to describe ontological knowledge in the knowledge base repository. The repository consists of a relational database and contains standard-compliant framework metadata.

- A typical framework scenario is triggered by a Use Case (UC) that is identified (1). This UC needs to be disambiguated (2). The disambiguation process has the purpose of estimating the UC signature and its competency question (CQ) (2.1), and determining the most suitable logical language and sub-language profile (2.2).

  A signature is the set of ontology concepts the UC refers to, and a CQ is a question or query that a knowledge base should be able to answer based on the modeled ontology. A use case usually comprises one or many competency questions. The estimated assessment in (2.1) determines the CQ nature and applicable corresponding reasoning (intensional: concerning concepts and properties, or extensional: related to intances as well), along with the most suitable profile (2.2).

- Ontology projection (5) takes place based on the recommended profile assessment in (2.2). Projection is defined as a technique to reduce ontology size and/or expressiveness through the application of modularization (5.1): extraction of modules according to a defined signature, or expressiveness-projection (5.2): algorithms for the depreciation of expressiveness, or both (5.3).

- When several profiles have been determined suitable (2.2) for the same CQ based on the results of (2), deprecated knowledge output obtained from (5.2) plays a role in the decision and the logical language with least deprecated knowledge is prioritized.

- Reasoning redirection (6) takes place according to this decision, and based on predefined metadata related to mostly recommended reasoners to be invoked per logical language.

- Unless the UC is performed by a framework expert user (a knowledge expert, engineer or modeler who masters the ontology and the KB repository), the outcomes expected of (2) can be strongly reinforced by the process of semantic tagging (4). In the context of the present framework, semantic tagging refers to the annotation of SN resources (blogs, forums questions and answers) with knowledge existing in the knowledge base ontology. Its purpose is to classify these resources with respect to logical knowledge, and/or populate the knowledge base ontology with information on SN users (writers and readers of these resources). Semantic tagging can be an automatic or a semi-automatic activity.

- Automatic semantic tagging is achieved following a corpus extraction and analysis phase (3): a corpus being a set of online textual data consisting of the UC question blog, and extracted data obtained through crawling the web (3.1), its analysis consists of ontology-aware natural language processing (3.2), i.e., applying NLP algorithms that take into consideration ontological vocabularies.

- Semi-automatic semantic tagging takes place when a SN user intervention is invoked after (3.2), with recommended semantic tags (4.1) consisting of logical expressions from the knowledge base, obtained through beyond-standard reasoning (7.2).

  Recommended tags are suggested after applying Natural Language Generation (NLG) techniques (4.2). Once the SN user tagging occurs (4.3), tags are subject to preliminary analysis (4.4) before being populated in the knowledge base

(4.5).

- (7) Ontology reasoning services (7) composed of standard (7.1) and beyond-standard (7.2) reasoning techniques are applied throughout the framework's different components. In (7.2), modularization is taken advantage of to suggest tags relevant to a certain vocabulary; abductive reasoning can be exploited for the retrieval of certain hypotheses that other scenarios will be able to prove/disprove.

  All decision-support outcomes (8) will be obtained based on standard reasoning techniques (7.1). For instance, through intensional CQs, the ontology is queried for information related to the populated individuals.

The contribution can thus be abridged holistically on the one hand, through the complete proposed framework and the ontology-driven approach its devising process is founded on, and analytically on the other, through the framework's inherent methodologies making up its particularities:

- Stemming from the Model Driven Architecture, Web3.OWL is an ontology-driven OWL-based framework dedicated for Web 3.0. This framework to begin with seeks to overcome previous efforts' limited usage of OWL expressivity, then to solve and minimize the drawbacks in SN media and applications for the parenting domain.

  The framework extends the cooperation between the Social and Semantic Webs, through the underlined use of advanced semantic web technologies. In particular, the different constructs, vocabularies and grammars of the OWL language will be exploited. The efforts are undertaken while maintaining a structured emphasis on the available profiles according to their corresponding application scenarios, and while stressing on the added expressivity and formal semantics of OWL 2, compared to the existing Social Networking efforts that use RDF and restricted OWL features.

- A knowledge base repository relying on an SMOF standardized metamodel to devise, model and arrange ontologies and their metadata. This includes an extension to the standard with additional structures and metadata elements known as “meta-semantics”. The extensions are necessary for the framework’s requirements, partially responding to the RFP (Request for Proposal) issued by the OMG task forces, in which they acknowledge and describe the limitations of its current version.

The knowledge base serves as a foundation for the domain ontologies and other SN standardized ontologies with mappings, signatures, categorization, modularization and alignment facilities. The main reference domain ontology that holds the most inclusive set of knowledge is identified as the “canonical domain ontology”, conforming to well-established patterns and rules of formal semantics.

- A sociomedical parenting ontology (“ParOnt”) built based on resources from knowledgeable scientists, domain experts, along with reliable references and data sources; this ontology strengthens the parenting KB, in particular by consolidating different sources’ knowledge and building consensus about key findings. It represents the framework’s groundwork that leads the rest of the process, supported by the “meta-semantics” arranged in the metamodel repository.

  ParOnt thus benefits from OWL 2’s expressiveness and accumulates the maximum possible knowledge of the field, even if in many application scenarios to follow, the complex constructs denoting high expressivity will not be brought into play. ParOnt thus forms the canonical domain ontology, as just introduced in the previous enumerated item.

- An expressiveness management approach that relies on ontology projection strategies including signature-based modularization and Description Logics fragments reduction. Algorithms in accordance with the OWL 2 profiles specifications are typically provided, and redirection mechanisms that lead to the most suitable reasoner are applied, with the purpose of retaining efficient results and satisfactory performance for complex tasks. This approach forms a workaround to bypass the serious performance obstacles experienced due to the high expressivity of the OWL language.

- A semantic tagging approach that relies on non-standard reasoning services supported by the metamodel contents on the one hand, and on a strategy for applying ontology-aware pattern-matching grammars for data mining and natural language processing (NLP) on the other.

  Mechanisms to suggest semantic tags following ontology modularization, disjunctive constructs, abduction and explanation reasoning techniques are advocated. Initiated on account of domain-specific semantic arrangements in the knowledge base repository, these tagging assignments are offered to taggers through a natural language generation (NLG) technique that hides all constructs and semantics' formalisms.

  This approach leads to ontology population with data on SN users, their characteristics and opinions; furthermore it can play an important role in validating the competency question and thus deducing the suitable profile for some of the application scenarios.

- Suggested steps and methodologies that illustrate several framework workflows, along with experimental scenarios based on the domain ontology ParOnt. These are presented with real-life examples demonstrating the usability and practicality of Web3.OWL in generating semantically engineered social data for efficient decision-support and recommender systems to deal with the drawbacks of the sociomedical parenting domain.

Web3.OWL, being an ontology-driven framework – where the key role of the ontology is highly acknowledged in the supporting knowledge base platform, as well as in each of its components – is not domain-specific by nature.

Nevertheless, several crucial elements lead to the fact that Web3.OWL, as it is currently proposed, is deemed to be a sociomedical framework.

They can be summarized in:

- The reference canonical ontology ParOnt that determines the scope; as a leading driver of the framework, substituting the ontology would open the door for the framework to be applied to different areas. Still, the time and effort to model and design a suitable ontology for a given domain are to be reckoned with, not to mention the availability of the appropriate level of know-how per domain to allow the right capturing of knowledge. Meanwhile, different parenting sources, professional expertise and collaborations formed a valuable opportunity that boosted the present work.

- The adequate data sources and their level of availability and accessibility; in the case of the current domain of interest, the numerous and abundant mommy blogging forums and websites constitute an exquisite advantage.

- The different identified domain deficiencies and limitations; OWL 2 recognized as the most suitable language for domain-specific ontologies modeling, a solid knowledge base is highly beneficial for the requirements of these domain aspects. This in addition was strongly suggested by the reviewed parenting reports and references.

- The required level of SN user participation and collaboration in the insinuated tagging activities; again, for the parenting sociomedical domain in question, the mommy bloggers happen to be ideal actors due to their collaboration willingness and to the type of “candid” and conscientious users factual studies report they are.

In terms of implementations, the thesis reports use cases, proofs of concept and real-life examples. These include experimental results highlighting the role of the conceived sociomedical ontology, the usability of other employed disciplines and ultimately the practicality and value of the proposed conceptual framework. These are all assessed by the role of Web3.OWL in contributing to the solution brought to different weaknesses experienced in online SN applications and resources for the parenting domain.

Here are possible use cases to translate scenarios in which the framework is able to assist at the practical domain level: when a SN user (mommy blogger) asks a simple question and receives tons of answers which are not all trustworthy, as some - often not a few - are full of fallacies, misjudgments, wrong conceptions or perceptions: are there means of assessment as to how much reliable blog answers are?
On the other hand, users' questions are sometimes too elaborated for a simple search or a weblog to accurately respond to them: "What should I do if my 3 year-old boy is suddenly stuttering and has gradually acquired an introverted attitude?", or "How can I teach discipline to my two children who have a defiant attitude and are in continuous rivalry?"

The aim of the framework, in contrast, is to avoid risky responses and numerous irrelevant search results, by providing direct and reliable answers inferred from the ParOnt sociomedical ontology.

Furthermore, profiling and segmenting the community based on particular criteria of interest is often constructive and contributes to pushing accurate recommendations, updates and follow-ups at the right timing or developmental stage. That way, it becomes more efficient for instance to make sure parents do not miss important milestones relevant to their child's profile on a timely basis.
Additionally, although not a primary goal of the framework, services linked to promotional and marketing suggestions fit to targeted audience are feasible based on the framework's outcomes.
Similarly, a typical use case could uncover the following competency questions: "what is the most common behavior witnessed in today's adolescents?" Of course this can be narrowed down to include a certain behavior type, a particular age group, sex, location, etc.; "What is/are the reason(s) today's youth is unhappy? What is a Montreal preschooler's favorite activity? (or of course the same query for different combinations of locations, age categories, etc.).

In conclusion, through the above mentioned contributions, this thesis seeks to maximize the exploitation of the huge efforts in the area of the Semantic Web. It thus takes hold of the opportunity of advancing the global adoption of these efforts, an adoption that lags far behind that related to the semantic web research achievements. The framework's reliance on standards and on other well-established disciplines driven by ontologies offers benefits in both the short and long term.

1.3 Thesis Overview

The thesis regroups, rewrites and extends previous work published in [26, 28, 27], in addition to potential imminent submissions currently underway.

That said, the thesis endeavors to present the contribution elements as clearly, concisely and efficiently as possible, abiding by the following organization:

Chapter 2 reviews the background literature that mostly influences the proposed framework.

Chapter 3 introduces holistically the Web3.OWL framework along with all its ontology-driven aspects, comprised layers and components.

Chapter 4 presents a complete overview of the expressiveness management strategy. It revisits previous work on modularization, explains how that work is embedded in the strategy, and lays out the proposed projection algorithm.

Chapter 5 details the proposed approach for semantic tagging.

Chapter 6 describes the efforts exerted to prove effectiveness and efficiency of the proposed framework and inclusive approaches. Different experiments have been conducted to cover a wide set of possibilities and features expedient for parenting sociomedical decision-support and recommender systems.

Finally, in Chapter 7, the different problems and objectives stated in the current chapter are revisited. Together with the contribution elements, they are matched and summarized as a channel of communication through which is asserted how well the thesis addresses them. This is followed by a wrapping up discussion along with observations and directions for open and future related work.

Chapter 2

Background Literature Review

This chapter presents the literature review of the most relevant work.

As proposing a holistic Web 3.0 framework then making particular emphasis on expressiveness management and tagging requires an inclusive literature review of the most critical pertinent subjects, this chapter introduces background information applicable to the work presented in this thesis. A particular emphasis is placed on the elements referred to in Chapter 1's problem statement section.

The present review cannot deeply cover all concepts and technologies tackled by the explored framework. The reader is expected to have a basic comprehension of knowledge bases and conceptual modeling. For more information on MDA, RDF and OWL, the reader is strongly encouraged to consult their corresponding cited references.

The chapter proceeds by presenting Web 3.0 in Section 2.1. Section 2.2 defines the Semantic Web fundamentals, laying a particular emphasis on Description Logics and OWL 2 particularities. Section 2.3 provides highlights on the standardization efforts and adopted models used for ontologies. The social semantic efforts and frameworks are described in Section 2.4, and an overview of the online sociomedical domain and characteristics is finally provided in Section 2.5.The chapter is closed in Section 2.6.

2.1 Web 3.0 in a Nutshell

Although it recognizes the fact that the vision of Web 3.0 that will be presented next is not adopted by the whole web community (namely because to some, the web is a unique web that cannot be categorized according to different generations), this thesis acknowledges this definition and builds on the way it has been envisioned.

Founded on the integration of Semantic Web technologies into Social Web applications, Web 3.0 is the newest generation of the Web. To elaborate more on its origin and emergence, we can go back to Tim Berners-Lee's initial vision on the web [8], which described it to be a read/write web. Despite that, the first version of the web turned out to be a read-only medium for the majority of users. This passiveness was gradually broken through a series of changes in usage patterns and technology. This opened the door to the socialization of the web with the appearance of blogs, wikis and other forms of web-based communication and collaboration sites and services, leading to Web 2.0.

On the other hand, with its famous aim of offering a much more intelligent web and superior knowledge management systems, the Semantic Web has witnessed a tremendous development, reshaping the whole AI research agenda by giving center stage to the fields of Knowledge Representation and Reasoning. The main idea behind the Semantic Web is to bridge the gap between humans and machines through advanced knowledge representation techniques, which entails the underlined idea consisting in providing knowledge in forms computers can readily process and reason with.

The marriage between Web 2.0, which sites are today's fastest growing segment of the Internet, and the Semantic Web, with its potential and dynamically growing research effort, yielded what is known as Web 3.0. Figure 2-1 shows how Web 3.0 extends Web 2.0 via Semantic Web technologies, where the Resource Description Framework (RDF) is the base of Web 3.0 applications, and the Web Ontology Language (OWL) is used to a very limited degree, as asserted in [46].

Figure 2-1: Web 3.0 Extending Web 2.0 via Semantic Web Technologies

2.2 Semantic Web Fundamentals, Description Logics and OWL 2

Since the Semantic Web concepts are the foundation on which the research presented in this thesis is based, this section presents the background information of its most impacting aspects.

A Semantic Web knowledge base represents knowledge in the form of ontologies. Originating from philosophy, the term "Ontology" has been "hijacked by computer science", and has consequently witnessed a reshaping process for it to fit to the context of the web and represent "a shared understanding of a domain" that allows terminology differences to be overcome. It is characterized by a list of terms or concepts of a domain and the relationships between them [2].

Gruber, on the other hand, was also known for the following definition of an Ontology: "an Ontology is an explicit specification of a conceptualization". He defined clarity, coherence, extendibility among the main ontology design principles that are to be respected while creating the kind of conceptual data model that is what an ontology at the end is [51, 41].

Several representation languages have been developed for ontologies and the Semantic Web, but the languages that are mostly efficient, standardized, and relevant to the present research are in fact the Resource Description Framework (RDF), and more predominantly the Web Ontology Language (OWL).

RDF is essentially a graph-based resource description language usually serialized as XML and built upon a statement formed of a block containing the triple "object-attribute-value". One important property of RDF is domain-independence. In other words, RDF is not restricted to a particular domain; instead, users are allowed to label and conceive their own terminology and vocabularies through the exploitation of a particular schema language known by RDF Schema (RDFS) [2].

That said, RDF and RDFS present major expressivity limitations [53], a fact that triggered the development of the Web Ontology Language (OWL), based on Description Logics (DLs).

Both OWL and DL main relevant aspects will be presented next.

2.2.1 Description Logics Languages and Reasoning

As a logic-based formalism for knowledge representation languages, description logics (DLs) [6] have been studied and developed for more than 25 years. They are decidable fragments of First-Order-Logic (FOL). DLs consist of a structured way for representing an application domain's terminological knowledge, with building blocks consisting of three kinds of entities: concepts (FOL unary predicates), roles (FOL binary predicates) and individual names (FOL constants). A DL ontology consists of statements called axioms formed based on the different types of entities and separated into three groups: the set of terminological axioms ($TBox$) for concepts, assertional axioms ($ABox$) for individuals, and relational axioms ($RBox$) for roles.

DL expressivity can reach a high, complex and yet decidable level. Various DL languages with different expressiveness levels do exist; of these we mention $\mathcal{SROIQ}$

[52]. $\mathcal{SROIQ}$ is a well-known expressive DL language; it is at the origin of the standardization efforts related to the Web Ontology Language OWL 2 [86]. OWL 2 main characteristics, profiles and fragments are to be overviewed in subsequent sections.

What stands behind the potentials and the prominence of DLs are the tractable reasoning methods that provide firm logic-based reasoning services. Accordingly, FaCT++ [85], Pellet [79], RACER [43], and HermiT [69] are considered as the most powerful and effective DL reasoning engines or reasoners built so far.

2.2.2 OWL 2 Fundamentals and Particularities

OWL is the most widely accepted and standardized ontology language of the Semantic Web, due to the fact that it fulfills the requirements that consist in "a well-defined syntax, efficient reasoning support, formal semantics, sufficient expressive power and convenience of expression".

OWL exploits RDF and RDF Schema; it can be expressed using RDF's XML-based syntax, along with other syntactic forms that have also been designed for OWL, like an easily readable XML syntax independent from RDF conventions [48], and an even more compact and readable abstract functional syntax [75].

The $\mathcal{SROIQ}(D)$ DL language that OWL 2 is expressed in has the following building blocks:

- $\mathcal{S}$ stands for the $\mathcal{ALC}$ family of DL, in addition to role transitivity axioms.
  As the smallest propositionally closed DL, $\mathcal{ALC}$ supports, in addition to atomic concepts (including $\top$ = Thing, the most general concept, and $\bot$ = Nothing, the inconsistent concept) and atomic roles:
  - Concept operators; the standard boolean operators: $\sqcap$ (conjunction), $\sqcup$ (disjunction), $\neg$ (negation), and restricted form of quantifiers: $\exists$ (existential restriction), $\forall$ (universal restriction).
  - Support for atomic roles only (no role operators or role axioms).

  - Concept axioms: $\equiv$ (equivalence), $\sqsubseteq$ (subsumption)

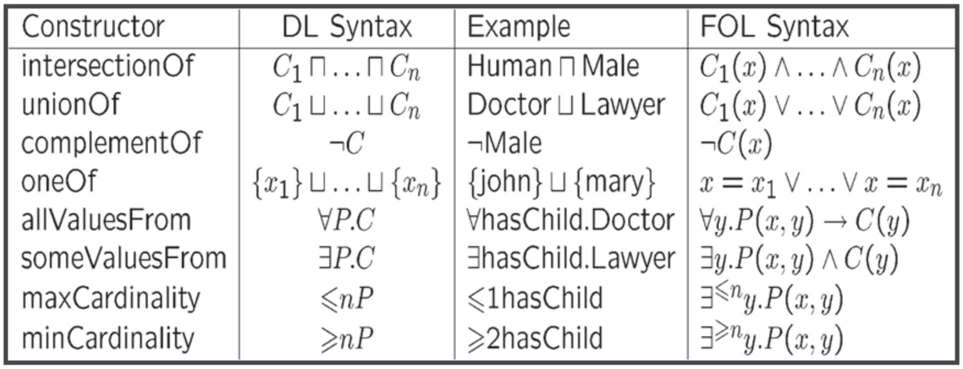
- $\mathcal{R}$ for property chains, property hierarchy, and property characteristics (symmetry, asymmetry, reflexivity, irreflexivity, functional and inverseFunctional, in addition to transitivity which is already included in $\mathcal{S}$)

- $\mathcal{O}$ for nominals (singleton classes).

- $\mathcal{I}$ for inverse properties.

- $\mathcal{Q}$ for qualified cardinality constraints.

- $\mathcal{D}$ for datatype properties (concrete properties).



Constructor	DL Syntax	Example	FOL Syntax
intersectionOf	$C_1 \sqcap \ldots \sqcap C_n$	Human ⊓ Male	$C_1(x) \wedge \ldots \wedge C_n(x)$
unionOf	$C_1 \sqcup \ldots \sqcup C_n$	Doctor ⊔ Lawyer	$C_1(x) \vee \ldots \vee C_n(x)$
complementOf	$\neg C$	¬Male	$\neg C(x)$
oneOf	$\{x_1\} \sqcup \ldots \sqcup \{x_n\}$	{john} ⊔ {mary}	$x = x_1 \vee \ldots \vee x = x_n$
allValuesFrom	$\forall P.C$	∀hasChild.Doctor	$\forall y.P(x,y) \rightarrow C(y)$
someValuesFrom	$\exists P.C$	∃hasChild.Lawyer	$\exists y.P(x,y) \wedge C(y)$
maxCardinality	$\leqslant nP$	⩽1hasChild	$\exists^{\leqslant n} y.P(x,y)$
minCardinality	$\geqslant nP$	⩾2hasChild	$\exists^{\geqslant n} y.P(x,y)$

Table 2.1: Examples of OWL Constructors

Table 2.1 contains examples of these constructs. It shows OWL constructors, their corresponding DL and FOL syntax along with sample expressions[1].

2.2.3 Profiles or Tractable Fragments

For the convenience of different communities with different needs, OWL (1) was divided into 3 upward-compatible sub-languages, based on expressivity factors: the first is OWL Lite, a sub-language of the second: OWL DL, which itself is a sub-language of OWL Full. A slightly more detailed description can be depicted next:

---

[1] Source: OWL Tutorials/Presentations - University of Oxford

- OWL Full: this is the full OWL language, with utmost expressiveness. It typically provides extreme syntactic freedom, to the point that it has grown to become an undecidable language.

  As a consequence, no complete or efficient reasoning support currently exists for OWL Full, and it is unlikely that any reasoning engine preserving all OWL Full features will be designed.

- OWL DL: this sub-language is a syntactic variant of the description logic $\mathcal{SHOIN}(D)$.

- OWL Lite: this is the least expressive OWL sub-language. It limits the OWL constructors to simply allow a restricted set that encompasses a classification hierarchy and non-sophisticated constraints.

  This sub-language, naturally, is the easiest to grasp, and presents a major weakness in its narrow expressivity. Nevertheless, despite its apparent simplicity, a few workarounds (implicit negations in axioms) enable OWL Lite to recapture most of OWL DL complexity, and become theoretically equivalent to the description logic $SHIF(\mathbf{D})$.

  Consequently, OWL Lite reveals an ExpTime worst-case complexity, which is slightly more favorable than OWL DL's NExptime.

Conclusively, OWL 1 sub-languages are all recognized as highly intractable. To deal with this problem, OWL 2 adopts a different structure of its sub-languages. First, it disregards OWL Lite, and copes with intractability through the design of profiles or tractable fragments under OWL 2 DL.
In what follows, the aspects of OWL 2 that are mostly relevant to this research will be described, specifically the OWL 2 profiles or tractable fragments, as well as a few other particularities.

OWL 2 Profiles or Tractable Fragments are "trimmed-down" versions of OWL 2 DL: they are the result of a simple trade between all-inclusive expressivity and efficient reasoning. The main profiles presented for OWL 2 are OWL 2 EL, OWL 2

QL, and OWL 2 RL [68]. Every fragment addresses a favorable application area, it is therefore essential to identify the target scenario in order to apply the accordingly most favorable OWL profile. In terms of reasoning engines, the regular OWL 2 reasoners are applicable; however, more capable ones specifically designed based on every fragment's constructs have been built.

The main OWL 2 tractable fragments are next overviewed [68]:

- OWL 2 EL

  Based on the EL++ family of description logics, the OWL 2 EL profile is a syntactic restriction on OWL DL, providing polynomial-time computation for ontologies with a huge number of classes [5].

  This profile offers OWL's expressive features required by large-scale ontologies, while eliminating unnecessary features and reducing common reasoning tasks' computational complexity. It is dedicated for ontologies in which the large number of collections of classes are taxonomically organized, and classification (the computation of the subclass relation between all the classes in an ontology) is the main reasoning service of interest.

  This profile offers OWL's expressive features required by large-scale ontologies such as the "Systemised Nomenclature of Medicine - Clinical Terms" (SNOMED-CT)[1] [80] renowned ontology.

  The profile's most important modeling features include:

  - Class Declarations
  - Subclass-of relationships
  - Class conjunction (Intersection-of set operations)
  - Some-values-from, has-value, and self-exists restrictions
  - Single individual enumerations

  To ensure tractability, disallowed class and property descriptions include:

  - All-values-from and Cardinality restrictions

---

[1] www.ihtsdo.org/snomed-ct

  - Negation and Disjunction
  - Inverse, irreflexive, functional, symmetric and asymmetric properties

  Despite its limitations, this profile often turns out to be appropriate for expressing interesting requirements. Note that it supports property domain and range restrictions, a fact that makes up for some of its restrictions.

- OWL 2 QL

  Based on the DL-Lite family of description logics, conceived specifically for reasoning with large amounts of data organized consistently with relatively simple schemata, the OWL 2 QL profile enables satisfiability of conjunctive queries in Logspace taking into consideration the queried knowledge base's number of assertions [17]. This fragment is similar to OWL EL with its polynomial-time computation when it comes to consistency determination and individual to class mapping.

  Upon attempting to compute all tuples of individuals that answer a conjunctive query q in a given ontology O, the idea is to start by rewriting a union of conjunctive queries uq of q that captures O's implicit information, and to then use conventional relational database techniques in order to answer the union of queries; in other words, the query is under an efficient expansion and translation process before being issued against its underlying storage mechanism, that being SQL, RDF or any other implementation. QL semantics are accomplished through a backwards chaining process, and offers capabilities that allow the capture of conceptual models, such as UML models, ER diagrams, and database schemas.

  This profile presents the following modeling features:

  - Class disjointness
  - Properties' domain and range
  - Participation constraints (e.g., every child has a mother)

On the other hand, subclass axioms are restricted to a limited set of language elements:

  - Explicitly defined classes
  - Some-values-from restriction
  - Complement-of and Intersection-of operations

  OWL 2 QL guarantees the ability to rewrite each ontology into a union of conjunctive queries by disallowing the use of disjunction, all-values-from restrictions, as well as property chains, transitive, reflexive, irreflexive, asymmetric, and inverse-functional property types which require recursive query evaluation.

- OWL 2 RL

  Based on a rule-based description logic fragment [40], and on parts of OWL Full rule-based implementations [82] that influenced its design, the OWL 2 RL profile is a forward-chaining rule processing system that allows the implementation of a set of rules while providing polynomial time complete reasoning. The profile's rule processing system is required to support conjunctive rules only; it also restricts individual reasoning on those individuals that are explicitly defined in the system, which preserves a deterministic reasoning.

  OWL 2 RL makes syntactic restrictions on axioms in primary OWL 2 constructs including:

  - Subclass expression classes are allowed to be described using only explicit class definition, one-of, some-values-from, and has-value restrictions, intersection-of and union-of operations
  - Superclass expression classes are allowed to be described using only explicit class definition, all-values-from and has-value restrictions, intersection-of operations, and zero or one value max-cardinality restrictions
  - Equivalent expression classes are allowed to be described using has-value restrictions and intersection-of operations

2.2.4 OWL 2 Other Effective Particularities

OWL 2 presents important evolutions and reengineered concepts; some of which will have a bigger impact on this proposal's exploration line of work, typically the ontology imports, versioning and XML syntax, and the Meta Object Facility (MOF).

- Ontology Imports, Versioning and XML Syntax [39]

  Through a very precise URI-based publishing on the web (URI: Uniform Resource Identifier), each ontology is uniquely identified and accessed. An ontology can directly locate its imported ontologies in OWL 2 "by name and location", with a very convenient redirection mechanism.

  In addition, every ontology keeps a version URI denoting its version; any particular version of an ontology is always accessible through this URI, again with a transparent redirection mechanism.

  OWL 2 introduces an XML syntax that presents several improvements for ontology web publishing. This syntax typically offers convenient and straightforward parsing and processing, equipped with XML's wide adoption and tools support. Moreover, it can be directly obtained through a plain translation from the MOF diagrams into XML schema. It has been used as a basis for the work and XML message exchange of the OWL API [49] and the OWL Link Protocol[1] [60].

  OWL Link offers an implementation-neutral mechanism to access DL reasoners, powered by the support of nearly all DL reasoners and a large number of ontology editors. The OWL API is a standard application programming interface offering the developers the needed functionalities and components to build the Semantic Web.

- Meta Object Facility (MOF)

  To overcome syntactic issues encountered in OWL's original version, the Object Management Group OMG designed a metalanguage that serves the objective of unambiguously defining an ontology's structural specification. MOF allows an ontology to be described independently of its serializing syntax. It consists

---

[1] www.owllink.org

of 22 UML class diagrams, providing a precise canonical structure for OWL 2. This structure forms a foundation for developers to rely upon in order for their different OWL 2 APIs to be interoperable and based on a well-known metamodel whose latest formal specification can be found in [71]. The metamodel extensions and other based standardized models are to be explored next due to their influence on the framework proposed in this thesis.

2.3 The Model Driven Architecture and the Meta Object Facility Extensions

The main principle behind the Model Driven Architecture (MDA), the architecture the proposed framework stems from, is to elaborate different platform-independent models (PIM) that rely on the domain in question's business model. Based on these PIMs, transformations are applied in order to reach the target platform-specific model (PSM) that leads to the concrete implementation of the system.

MDA is a software design approach to system development that maximizes the role of models. It is model-driven because it provides means through which models direct the course of "understanding, design, construction, deployment, operation, maintenance and modification" of a system [58].

The motivations behind MDA are thus the separation of the design from the architecture and implementation. By focusing on requirements through modeling instead of implementing, requirements' changes are drastically simplified. The challenges of MDA reside in the facts that it is not definite that all aspects of the required system can be expressed in the abstract model, and it is not guaranteed that the necessary specialized skills are available.

The term Architecture in MDA does not make reference to the modeled systems' architecture; it refers to the architecture of the different standards that form the basis for MDA. The above mentioned MOF (Meta Object Facility) metamodel, with its extensions and different supported models, are among the core MDA standards.

2.3.1 Meta Object Facility Support for Semantic Structures

As an extension to MOF that handles more dedicated requirements related to semantics and classification of objects, the Meta Object Facility Support for Semantic Structures (SMOF) supports requirements for ontology development, such as multiple inheritance and multiple classification. Proposed in September 2010, its formalization was underway, and it was not until very recently (in April 2013) that it was standardized and published [72].

2.3.2 Ontology Definition Metamodel

The Ontology Definition Metamodel ODM [70] is an MOF-compliant standardized language for the definition of ontologies; it enables using the UML notation in ontological engineering. Figure 2-2 shows the OWL class diagram as represented in ODM. ODM plays a crucial role by allowing a marriage between MDA and Semantic Web technologies and thus supporting Semantic Web services, ontology and policy-based communications and interoperability.

2.4 Social Semantic Efforts and Frameworks

2.4.1 Introduction

The next section will comprise an overview of how the Semantic Web has so far been used for online Social Networking. It consists of an exploitation of the main efforts deployed so far in this area, through a selection of the most impacting and critical findings and researches conducted by the experts in this interdisciplinary field.

As it will be demonstrated, the vast majority of researches and implementations of Semantic Web concepts for Social Networking mostly focus on RDF constructs, and at a second degree on OWL general expressions. There is no formal approach that explicitly considers OWL expressivity levels and specific constructs, OWL 2 characteristics and improvements.

The framework that will later on be described will either make use of, or build on

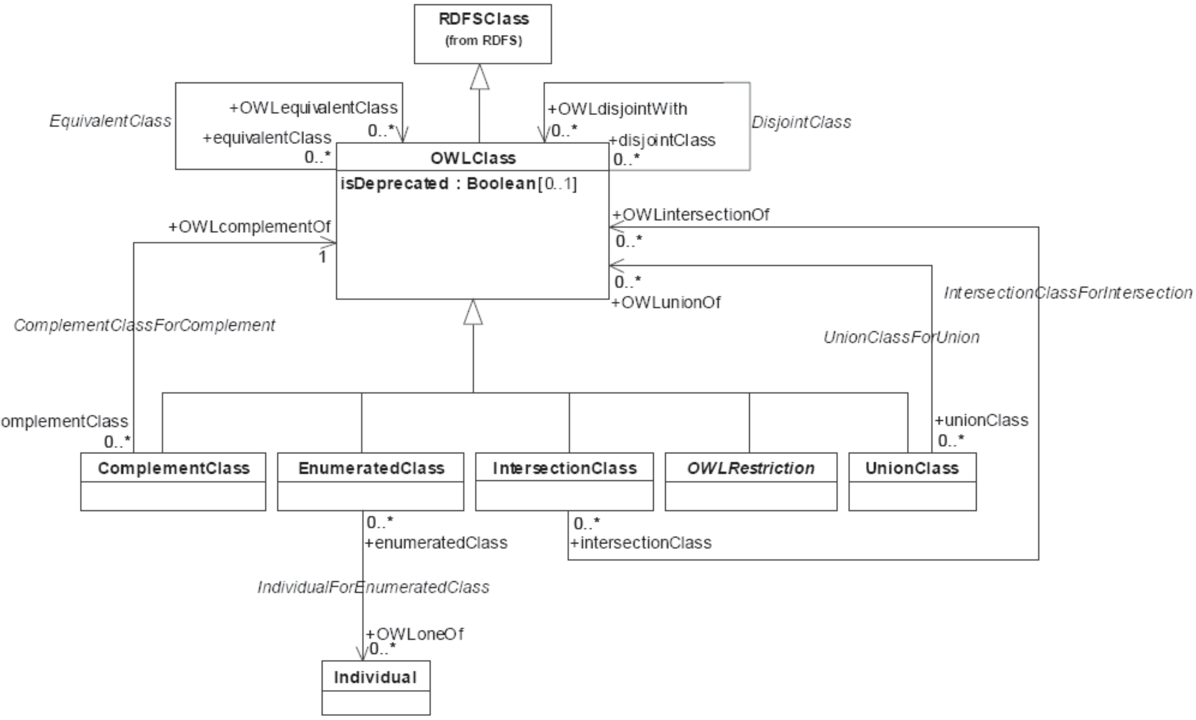


Figure 2-2: The OWL Class Descriptions Diagram
Source: [70]

most of these efforts. It can therefore be perceived, from one of its angles, as a sort of a formalization attempt tackling existing and future Semantic Web undertakings for Social Web purposes.

2.4.2 Review of Main Semantic Web Implementations and Efforts for Social Networks

When attempting to review the most significant efforts at the level of applying Semantic Web concepts to Social Networking, the endeavors that are mostly worth being mentioned are the ones summarized in the following subsections.

As a matter of fact, efforts conducted by a set of researchers, including Mika (jointly or individually) in [10, 66, 64, 63], Finin, Ding & al. in [24, 34], Gruber in [42], as well as several studies and researches led by Breslin, Bojars, Decker, Passant, jointly or individually with other collaborators in [11, 12, 56, 15, 74], all demonstrate the usage of the Semantic Web constructs for Social Networking. They lead primary and novel fundamental concepts strongly affecting the interdisciplinary field of Social Semantic Web, such as folksonomies and linked data, FOAF (Friend-of-a-Friend), SIOC (Semantically Interlinked Online Communities), SKOS (Simple Knowledge Organization System), Object-Centered Sociality and collective knowledge systems.

The main efforts undergone based on a cooperation between the Semantic and the Social Webs have yielded a vast number of interesting SN specifications, ontologies and projects. Some of the main contributions that the current framework is set to be compatible with, to reuse or extend, are summarized next:

- The Semantically Interlinked Online Community (SIOC[2]) initiative [14] presents an ontology for representing user activities in blogs and forums, thus increasing the integration of the information in online communities. SIOC is a description of online-community information. It offers means to represent "rich data" from the Social Web.

- The Friend-of-a-Friend (FOAF[3]) is an ontology for describing people along with their relationships. FOAF can be integrated with other Semantic Web vocabularies, like SIOC, SKOS, etc. and has been established as the most broadly used domain ontology on the Semantic Web [16]. That said, sections below will demonstrate the misuses and abuse of this ontology (conceived in OWL Full).

- The Meaning of a Tag (MOAT[4]) [74] framework allows the association of tags to semantics, via linking them to knowledge base URIs such as DBpedia[5] [3], GeoNames[6], etc.

These efforts primarily depict major RDF implementations, as well as some OWL usages, contributing to the rise of important Semantic Web and Social Networking dual concepts, projects and ontologies; no explicit clear descriptions and implementations specifically denoting OWL 2 and its vocabulary constructs, enhancements and profiles are witnessed.

---

[2] www. www.sioc-project.org

[3] www.foaf-project.org

[4] www. moat-project.org

[5] www.dbpedia.org

[6] www.geonames.org

2.4.3 Review of Expressivity in General for Online Social Networking Applications

When it comes to expressivity, many argue that the level provided by RDF is amply adequate for the needs of Social Networking, and that particularly due to the fact that it allows users to avoid complex constructs in more sophisticated languages (like the Web Ontology Language). The examples and citations are amply available to support this claim.

In [31] for instance, it is stated that current algorithms for Social Networking are essentially founded on graph pattern detection while employing very modest semantics. Correspondingly, the analyzed networks are described with very poor semantics, which entails the lack of semantic sociometric patterns identification. The different available community detection algorithms – that are graph-based and that do not take the relation types into consideration – can be used as an example, according to the authors.
Nevertheless, Collective Knowledge Systems in [42] propose a class of applications that expose the Social Web's collective intelligence to the reasoning power of the Semantic Web, highlighting the role of the latter in creating new value from the collected data. Underlining reasoning and computational power's dependence on the completeness and structuring level of the data, a tradeoff of cost with inference structure and depth is described, without delving into the details of the possibly suitable vocabularies and constructs for Social and Semantic Webs' synergy.

On the other hand, Mika in [64, 65] makes an account on expressivity, and on a number of issues encountered with the adopted methods for SN data representation on the web, especially upon considering RDF and microformat semantic annotations. While for instance he considers FOAF as a rich ontology for the characterization of individuals' online presence, he states several weaknesses especially when it comes to property description. He fortifies his claim by mentioning the fact that there is only one single *FOAF:knows* relationship that is overused with no ontological restrictions on its definition. Despite the original vocabulary makers' intention of providing maximum looseness for the purpose of targeting the widest scope possible, their expectations of others employing the RDF/OWL extensibility for more precise relationship notions and definitions were not met. Relationships' semantics ended up by becoming strongly diluted.

The same researcher in [66] reports the way the technologies of the Semantic Web have been "quietly maturing and spreading" until having reached a stage at which they are capable of providing a clear way to apply a "basic level of formal semantics" to the infrastructure and pages of the web.

Finally, the authors in [15] vividly promote their limited expressivity and attribute to it the success of their different endeavors, including SIOC, as well as the wide adoption of FOAF.

This is due, as they claim, to the mere fact that the masses will not collaborate unless the presented vocabulary is in its simplest form.

In the same reference, the research is concluded with an overall account on the issue of expressivity: a major obstructing factor to the area of Semantic Web and Social Networking is the inability to set an agreement on the set or sets of vocabularies to be used for particular situations.

The reason behind this, according to the authors, is the lack of community guidelines and tools and the surplus of users and new arrivals who wish to enrich their websites and deploy Semantic Web applications on them.

A possible alleviation to this problem, according to them, is an under-developed briefly described proposition, which they call the "vocabulary onion". The proposition includes the creation of a niche of vocabularies based on FOAF, SIOC and other known social ontologies at the core layer, as well as more specialized domain specific vocabularies at the outer layer.

2.4.4 Emphasis on the Primary RDF Role and Implementations

The vast majority of Social Semantic Web data is expressed by means of RDF, the Resource Description Framework, and queried through the RDF query language SPARQL. Numerous works demonstrate this fact, and in [9], a set of popular and core vocabularies recommended for linked data usage is published, with a strong stress on RDF and the RDF data models. Two basic principles are soundly stated to underline the use of:

- Firstly, the RDF data model for structured data web publishing
- Secondly, RDF links for different data sources' data interlinking

Moreover, and always in [9], a set of best practices and guidelines for employing and avoiding specific RDF constructs is made available.

RDF is present in virtually all Social Semantic Web applications; its graph-oriented nature smoothed the cooperation between Semantic Web RDF tripartite nodes and Social Network graphs.

In [78], an extensive description of implementing RDF and SPARQL for Social Networking is provided. According to this paper, when the objective is to obtain Social Network data exchange, interoperability, transformation, and querying, an RDF and SPARQL-based standard model is the key.

The authors describe an RDF data model for Social Network data – obtained through a map of the SN conceptual graph data model to RDF – along with a SPARQL query and transformation tool, functioning in conjunction with a few basic SQL queries. This combination of RDF and SPARQL is established as a perfect candidate for meeting the above stated objective.

In [31], an architecture for an RDF-based Social Network Analysis tool is described. The proposed tool delves into Social Network applications' users interactions and profiles annotated in RDF. It makes use of domain specific ontologies' conceptual vocabularies and of RDF graphs' path extraction according to certain criteria, to eventually provide extensions dedicated to Social Network Analysis (SNA), facilitating the exploration of RDF Social Network representations.

Researchers in [11] presented SIOC, the Semantically Interlinked Online Community, an ontology aiming at representing user activities in blogs and forums, thus increasing the integration of the information in online communities. SIOC is a description of online-community information. SIOC offers a means to represent "rich data" from the Social Web using limited OWL formal semantics. In [12], the research evolves and the SIOC Types module is introduced in order to play the role of a glue for various RDF representations of social data. After a description of the way Social Networks are created via object-centered sociality (connecting people via the common interests), RDF is stated as "a universal model" in which all kinds of information and real-world objects from Web 2.0 sites can be expressed. RDF allows the interpretation of semantic information in graphs whose constituting elements are the resources depicted as objects and connected by properties that describe their available attributes and relationships.

Thus, RDF is strongly present as the main foundation for Social Networking applications. In some of the reviewed efforts, the role of OWL comes subsequently with the introduction of a few OWL extensions to the RDF main foundation.
More details on these extensions and on the overall OWL role will be elaborated in the next section.

2.4.5 Overview of OWL Usages and Implementations

The literature reports relatively very little use of OWL for Social Semantic Web applications, compared to what exists for lightweight implementations and projects involving RDF and RDFS in the Social Semantic Web.

In [64], Mika states two reasons why most available web ontologies barely implement OWL: on the one hand they tend to have very little requirements and restricted needs, and on the other there is a difficulty in expressing general rule-based knowledge using OWL. He also notes the lack of a standard query language for OWL, but mentions [33] as a proposal for a query answering language for OWL. Later on, this research will dig deeper into OWL QL and study the feasibility and advantages of making use of it for Social Semantic Web needs.

Meanwhile, several references demonstrate how OWL ontologies having the highest level of expressivity are mostly at the service of expert systems designed for centralized, restricted environments, such as medicine and engineering. These domains are formal by nature and consequently benefit greatly from the increasing expressivity of OWL ontologies.

On the other hand, distributed and less controlled and restricted environments, of which the web is a perfect example, tend to seize lightweight ontologies described in RDF(S). This is particularly denoted in Mika's mentioned references, in which he further analyzes this overall preference and tendency towards lightweight ontologies in [63], where he brings into play the appearance of folksonomies. A folksonomy originates from the words "folk" and "taxonomy", to identify collaborative tagging and categorization freely completed by the masses on the web.

Breslin & al. elaborate more on folksonomies, by explaining and developing their semantics in [56], putting forward a semantic model for folksonomy known as SCOT ("Social Semantic Cloud of Tags ontology, its model is already introduced in [12]). SCOT uses basic RDF/OWL terms to explicitly illustrate tagging entities and their relationships.

The reference also presents a table in which the available tag ontologies are listed along with their representative features, demonstrating that only a few support basic preliminary OWL constructs. For instance, the SKOS, or the Simple Knowledge Organization System [7], offers a lightweight and intuitive language for knowledge organization systems. SKOS has its data model defined as an OWL Full ontology.

It takes advantage of the expressiveness in OWL Full, of its notion of inconsistency, and of its Open World Assumption, to store different kinds of data and information, such as thesauri, classification schemes, etc. SKOS data is defined as triples of RDF. This language may be used either independently, or combined with other formal representation languages such as the Web Ontology language (OWL).

In conclusion, a general consideration of the work of the researchers and authors in [15], reveals the fact that it is in its entirety based on the RDF Schema, with a few OWL extensions (most often the owl:sameAs and owl:differentFrom properties).

The Friend of a Friend ontology, FOAF [16], established as the most broadly used domain ontology on the Semantic Web, is handled by the different users in an open and extensible manner [24, 34]. In the same studies, the authors investigate FOAF's propagation and application, and analyze the Social Network's structure when formed through FOAF relations. In their analysis, they perceive the methods of using, and "abusing", the Semantic Web languages RDF and OWL for Social Networking Web usages and demonstrate their deficiencies.

An interesting research conducted by Mika in [62] involves the use of OWL's inverse functional properties for a task called 'smushing', or same object unification into one entity. This merge of equivalent instances can be applied to the different objects sharing the same value for their *owl:InverseFunctionalProperty*. For instance, the email address is an inverse functional property, as every email belongs to one person only. Mika elaborates in [64] on the limited capabilities of Description Logic OWL reasoners in the context of instance equality reasoning (which he defines as the inference of owl:sameAs and owl:differentFrom statements). He also reveals the fact that his practical application area, similarly to many others, experiences serious lacks in taking profit from the theoretical results of efficiency experienced in OWL.

2.4.6 Major Ontology-based projects and frameworks

In an attempt to identify previously conceived frameworks that could have similarities with the proposed one and identify the differences between them and the present one, the following paragraphs are reported.

“RapidOwl” [4] consists of an agile knowledge engineering methodology conceived to increase the collaboration between domain experts and knowledge engineers, inspired by the agile software engineering methodology. It thus ensues through iterative refinement and annotation to achieve its goal of knowledge base structuring. Web3.OWL however cannot be compared to RapidOwl because the latter is limited to knowledge base conception and design. It concentrates on the software engineering aspect, namely the agile methodology principles, to recommend iterative knowledge base refinement and annotation management.

On the other hand, other existing frameworks, namely ([19, 32]), process Web 2.0 documents in order to extract facts to enrich the knowledge base, following an ontology learning process.
The approach adopted in these frameworks is again different from the one in Web3.OWL. In this framework, the ontology is a sociomedical one, with trustful knowledge that forms the focal point driving the rest of the components. While an ontology population process of web users and their family information and beliefs takes place, the rules and medical information modeled remain intact in the knowledge base. Furthermore, these frameworks deal with lightweight ontologies, without providing a particular emphasis on expressiveness, DL vocabularies and advanced or specialized reasoning, the way Web3.OWL does.

No ontology-driven Web 3.0 frameworks stemming from MDA were perceived based on the conducted state of the art study.

2.5 Online Sociomedical Domain Overview - State of the Art and Limitations

While ontologies and the Semantic Web have proven their usability and effectiveness in the bio-medical domain, the case is not the same for the sociomedical domain. There is a lack of ontology-based efforts, particularly ones having the aim of parenting education, awareness and orientation.

2.5.1 Parenting Social Media - State of the Art and Limitations

Observations and research evidence and recommendations detailed in [38, 35, 61] showed that the past few decades witnessed a mass media explosion of information and advice about child-rearing, and that little attention was dedicated to the communicated messages' quality and impacts on parents and parenting. Additionally, reports relate undermining drawbacks that need to be addressed for the social media to bestow positive support, such as:

- The scarce and scattered reliable accessible information that combines different factors affecting parenting education, especially since such information is entrenched in a multiplicity of disciplines and communities (for example: social work and community development, psychology, law, early childhood/adult education, medicine, etc.).
- The frequently confusing and conflicting nature of the information and advice conveyed to parents. This is due to several factors: on the one hand, change has always been the only constant in child-rearing advice which is affected by economical, intellectual, cultural and social factors that frequently vary; on the other hand, many aspects are controversial between experts and throughout various generations. Furthermore, conveyed advice should take into consideration the particular case by case situation and facts in order for it to be reliable.

- The fact that social media does not dedicate equal attention to different stages of parenthood. There is a bigger focus on younger children on the account of adolescents and older children who often have critical developmental needs. Instead of getting appropriate support, adolescents are faced with a negative image set for them by social media and by media in general.

The suggested recommendations that should be adopted in order to deal and overcome these deficiencies, based on the above mentioned references, will be addressed later on as the framework is explained and depicted through various scenarios.

2.5.2 Sociomedical Frameworks and efforts - Summary from the State of the Art

In what follows, projects dealing with the sociomedical domain and subdomain aspects are shortly described.

There currently are a few research projects focusing on automatic surveillance of obesity and its associated diseases.

The European project EPODE [13] proposes a framework for analyzing community-based interventions for prevention of childhood obesity, and IBM's Smarter Planet Platform for Analysis and Simulation of Health (SPLASH) [20] in USA.

The SPLASH's goal is to create a platform that takes expert models of constituent real-world systems related to health, synthesize and integrate those models to lead to an interoperating complex composite system model with which policy-makers can try out alternatives in a low-cost, highly responsive way.

Most of the existing systems rely on databases and syntactic approaches, whereas the present approach enables researchers and public health practitioners to perform semantic integration and querying.

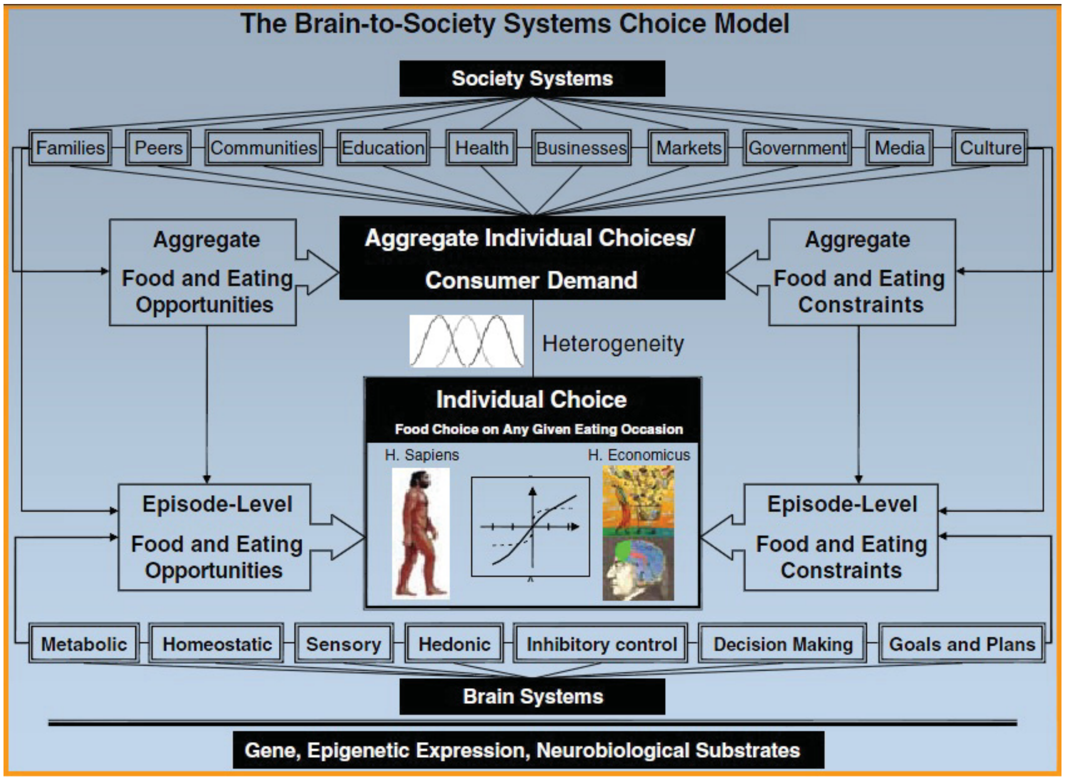


Figure 2-3: The Brain-To-Society Systems Choice Model - *Source:[25]*

2.5.3 The BTS - Brain-To-Society - Systems Choice Model

The BTS (Brain-To-Society) Systems Choice Model [25] is a transdisciplinary research project that aims at promoting a healthy lifestyle and preventing obesity, through individual, organizational and collective choice systems established on biologically-based models. These models take into consideration the environmental and societal factors as well.

As illustrated in Figure 2-3, the challenging undertakings aim at creating models that consider biological, environmental and societal factors that influence an individual's choice. Models thus take into account the implied theoretical, methodological and analytical developments, necessitating an integrative infrastructure. This infrastructure then deals with information about choice behavior drawn from many levels and many sectors.

The domain-centric efforts deployed within the scope of the proposed framework partly stem from the BTS project, most particularly the endeavors published in [28], as it will be shown later on.

2.6 Conclusion

This chapter covered an inclusive comprehensive overview of the background literature related to areas and projects that will have an impact on the rest of this thesis. The next chapters will be dedicated for the description of the proposed framework along with its comprised particularities. They will relate, when applicable, other relevant literature state of the art reviews.

Chapter 3

Introducing Web3.OWL: An Ontology-Driven Web 3.0 Framework

3.1 Introduction

With the broad and specific motivations and purposes of the thesis having been covered in the previous chapters, the present chapter addresses the Web3.OWL proposed framework abiding by a top-down approach to present its different comprised layers and most characterizing aspects.

In Section 3.2, a recap of some of the motivations behind its conception, as well as a high level depiction of its components are stated. Next, Section 3.3 explains the fundamentals behind its knowledge base (KB) repository and ontology design, and Section 3.4 provides an overview of the data analysis strategy. Section 3.5 runs through the framework infrastructure and workflow, providing typical processing scenarios. Section 3.6 presents the advantages and challenging aspects of the proposed framework; the chapter is wrapped up with Section 3.7.

The work presented in this chapter is partly based on [26] and [28].

3.2 Framework Overview and Main Components

To answer the research stated problems and its correspondingly exposed aims, an ontology-driven Web 3.0 framework that guarantees the cooperation between the Social and the Semantic Webs through the underlined use of OWL 2 with its different constructs, vocabularies and grammars, is proposed as a rewarding solution.

This framework was baptized Web3.OWL to emphasize that it fundamentally addresses Web 3.0, and is furthermore driven by ontologies. In particular, it exploits OWL 2 features as opposed to the existing efforts that consist in graph-based RDF. To emphasize the advantages of insisting on having the framework established on OWL 2 instead of RDF, the main limitations of RDF are recalled: limited expressivity, loose semantics, lack of standardization, and therefore of known supporting reasoners (the reader is referred to [2] for a more extensive survey of RDF limitations).

Following is a summary of the most important RDF restraining aspects which, and as they become beaten by the framework's OWL features, allow it to prove its value and significance:

- Expressivity deficiencies denoted by the lack of support for equivalence, disjointness, negation, or for certain property axioms (such as transitive, inverse, functional properties), or cardinality restrictions, to name a few.
- Lack of automated reasoning support, particularly beneficial for larger ontologies for which manual and on-demand development of consistency, classification and instance checking tools become more and more strenuous.

On the opposite side, by basing the framework on OWL, it gets to capitalize on all techniques and advancements achieved owing to the strong DL theoretical research:

- It benefits from the range and sophistication of tools and supporting infrastructure, like the different editors, the OWL API, mapping, integration and modularization facilities.

- It gets endorsed by the different reasoning services and facilities, whether standard reasoning (entailment-based, to be explored later on), or non-standard OWL reasoning (most particularly "modularization" and "abduction", more details will be provided in subsequent chapters on their usefulness and applicability). Reasoners are also fully-fledged (i.e. traditional reasoners that are not dedicated to a particular sub-language), and specialized (that process particular language fragments).
- It is supported by the standard metamodel ODM (based on SMOF), with all the potentials this benefit offers, as it will be demonstrated in the coming sections.

While Figure 3-1 provides a high-level depiction of its main flow, components and layers, the next subsections offer more details when it comes to those layers and components and to the essential characteristics that make up its particularities.

Looking at Web3.OWL from its widest angle, two main components are distinguished as follows:

- The infrastructural elements (the bottom horizontal layer in Figure 3-1) represented by the KB repository and the ontological data it comprises:
  - An SMOF-supported metamodel enclosing the structures necessary for the semantic and syntactic containment. These have been abstractly defined, and at the practical level, a relational database was defined to hold them along with their data.
  - The canonical domain ontology, i.e. the reference ontology conforming to well-established patterns and rules based on formal semantics, and holding the complete domain knowledge.
  - Projected domain sub-ontologies; i.e. reduced versions based on signature-projection (modularization) and/or expressivity depreciation.

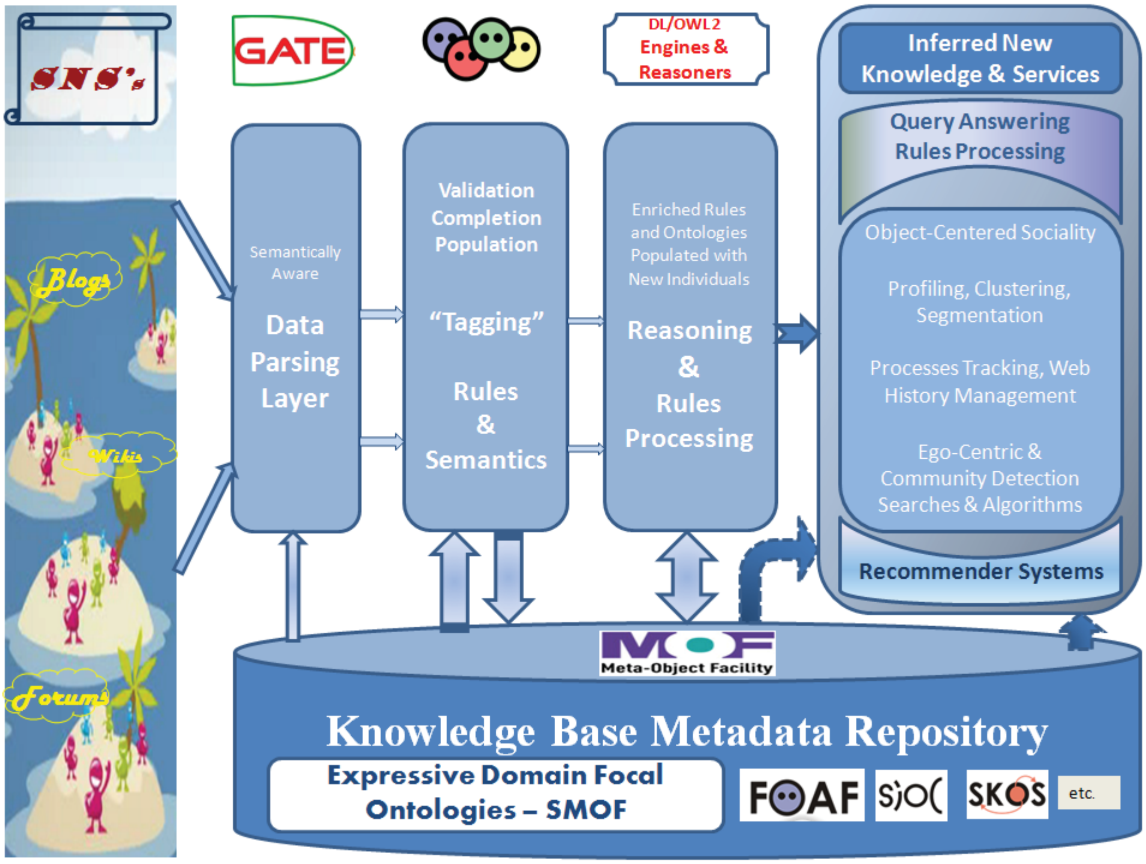


Figure 3-1: Web3.OWL High-Level Overview

  - Previously existing social semantic standardized ontologies (the framework benefit from their existing models and builds on them).

- The collaboration flow (the vertical towers in Figure 3-1 that follow the leftmost one showing SNS websites represented by islands in the sea) that paves the way for the social data processing towards the desired outcomes:
  - The semantically-aware data parsing layer: a data mining and ontology-based NLP-assisted approach to parse and detect semantics from the crawled SN Website data (mommy blogs).
  - Semantic tagging facilities that enhance the previous data mining layer and promote semi-automatic ontology population with annotated semantic information on resources and SN users (mommy bloggers).

- Domain ontologies (with or without automatically/semi-automatically population of instances) are subject to reasoning. Reasoning redirection takes place after expressiveness management through projection approaches to avoid complexity costs.
- Reasoning capabilities open the door to a variety of knowledge and information systems services that feed the decision-support and recommender systems. Particular types of services are mentioned as examples in the rightmost tower in Figure 3-1, more details on these services will be provided in Section 3.6.

3.3 Ontology Modeling and Knowledge Base Repository

The introduced Web3.OWL framework embracing the efforts described in this thesis is founded on a modeling platform whose backbone is a KB repository containing an expressive conceptualized domain ontology (considered as the repository's canonical ontology) that interoperates with other existing standardized ontologies (such as FOAF, SIOC) while being subject to projection and modularization techniques.

3.3.1 Knowledge Base Repository Overview - SMOF Extended

For the different requirements of a suggested ontology-driven framework, it is not surprising to resort to the fundamental standards behind the Model Driven Architecture. While being an extensible model driven integration framework for the definition, manipulation and integration of metadata and data in a platform-independent approach, MOF is considered as an OWL 2 incorporated novelty [39]. The Semantic Meta-Object-Facility, SMOF [72], a derivative of MOF, solves some particular constraints related to OWL and ontology modeling, and is used herein as the basis for the modeling of the framework's canonical domain ontology according to the ODM

(Ontology Definition Metamodel). Section 2.3 in Chapter 2 was developed to highlight these facts.

To clarify the principles of metamodels according to the OMG's metamodeling infrastructure and the way the framework capitalizes on them, the four metamodeling layers need to be explained with examples of what they represent in Web3.OWL's context:

1. L^3 is the most abstract metamodeling layer. It is named the meta-metamodel and defines the most abstract syntax of a modeling language. It conforms to itself.
   In the ontological semantic context, this layer is expressed in MOF and SMOF models. In these models for instance, it is defined that an ontology consists of classes (concepts) that can be linked through relations called properties; a property can have property characteristics, and more than one characteristic can be attributed to a property at the same time (SMOF).

2. L^2 is the metamodel, it instantiates L^3 and conforms to the abstract syntax defined in it.
   In the ontological semantic context, this layer is expressed in the Ontology Definition Model (ODM) that respects the constraints and syntaxes in MOF/SMOF and represents them through diagrams. In Web3.OWL, these models (and their proposed extensions) were created in a database relational schema that forms the knowledge base repository, special structures called "meta-semantics" were added to extend ODM with additional knowledge on the logical levels of the SMOF elements.
   Appendix B presents and describes this schema.

3. L^1 represents the model that instantiates the metamodel and conforms to the abstract syntax represented by that metamodel.
   This model in the ontological semantics context comprises the ontology *TBox* and *RBox* elements. At the framework level, it consists of the creation of the

ontology (in the OWL XML-based syntax) and its instantiation in the repository structures created in L^2 and including the meta-semantics.

4. L^0 represents the data layer that instantiates L^1.
   In the ontological semantics context, this layer represents the instances (*ABox* elements).
   The separation of L^1 and L^0 is not strict (similarly to the way data instantiates schema in database models for example), since axioms can be formed of constructs including concepts, relationships and instances at the same time (*TBox* and *RBox* with *ABox* elements).

The framework thus extends SMOF by adding particular layers that are essential to some of its core components.
In August 2013, OMG (the Object Management Group) issued an RFP (Request for Proposal) inviting potential contributors to submit extensions to SMOF that handle specific needs not covered in the actual version of SMOF [73]. Proposals are due by the end of 2014, and some of the expected extensions that happen to be crucial for the immediate needs of the proposed framework are mentioned next:

- Providing a meta language for interoperability among Semantic Web languages (OWL, RDF), with translations between different logical languages without changing the original formalisms and specifications of any of the languages

- Establishing means of conformance for the logical languages: OWL 2 (along with its profiles EL, RL, QL), RDF, RIF Core (Rule Interchange Format), etc.

- Setting up the means for expressing modularity operations and relations, as well as for the matching and alignment of different ontologies covering one domain.

- Giving a description of constraints and conformance criteria for additional concrete languages and translations between concrete languages.

It should be noted that integral extracts from this RFP describing the details of the requirements are provided in Appendix A).

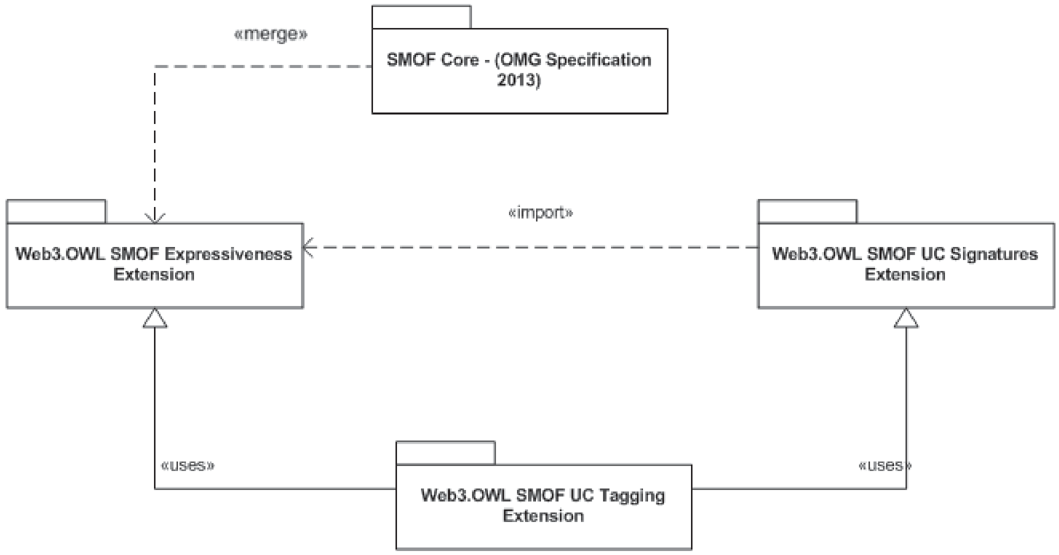


Figure 3-2: Web3.OWL Package Diagram

Relying on a standardized metamodel minimizes models' misinterpretation and misclassification by both humans and machines. It enables users to build models that have clear and precise semantics.
In an SMOF-based model, an element is appropriately defined in the context of an extent that distinguishes it unambiguously from other elements through a unique identifier, with links similarly detected between elements.

Figure 3-2 presents the package diagram that illustrates the way the framework's repository was conceptualized based on SMOF and how it extended it:

- The main package "SMOF Core" represents the standardized specification-based ODM metamodel that holds the ontological definitions, classes, properties and their relationships.
- This package merges into the one denoted by "Web3.OWL SMOF Expressiveness Extension" which reuses it and adds additional layers to group different

constructs according to their supported language and profile. More details will be provided in Chapter 4 and in Appendix B's list of "SMOF" and "ODM" structures.

- The rightmost package denoted by "Web3.OWL SMOF UC Signatures Extension" is dedicated for the modeling of arranged signatures computed according to the particular use case (UC) business area. This will be described more extensively in chapters 4 and 5 and in Appendix B's list of "ODM and SYS" structures.

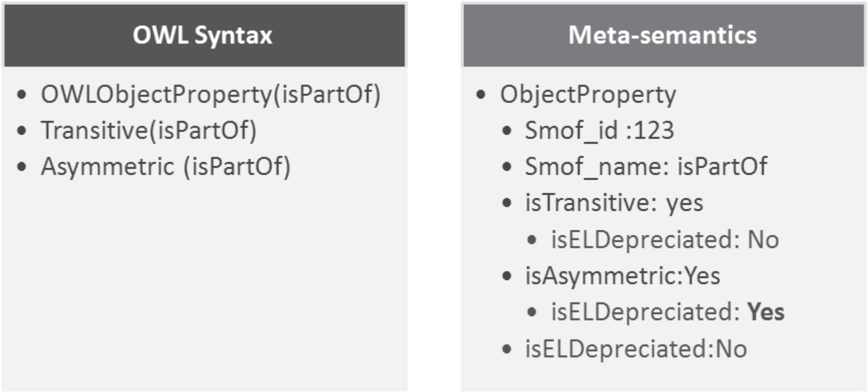
- Finally, the package "Web3.OWL SMOF UC Tagging Extension" describes the tagging and SN data particularities, by using elements defined in the upper package. Chapter 5 and Appendix B's list of "SYS" structures will reveal more details on the contents of this package.

The featured extensions are appropriate for dealing with the framework specifications. For instance, the repository holds dedicated structures known as "meta-semantics" structures; these structures play a crucial role in sorting and grouping the different axioms in the knowledge base, thus further "semanticizing" them, to later allow automatic up or down scaling between the OWL 2 sub-languages having varying levels of expressivity.

Algorithms and methods that allow this categorization procedure will be further explored in the next chapter, since this semanticizing process will have an important impact at the different levels of the platform.

It will later on be demonstrated how the Web3.OWL framework extends SMOF metamodels and adds to their usage particularly when the latter usage goes beyond the purpose of ontology modeling and reaches outcomes that make it possible to apply projection, reasoning redirection and services for decision-support systems.

Figure 3-3 illustrates, as an example of Web3.OWL meta-semantics, how an OWL



OWL Syntax

- OWLObjectProperty(isPartOf)
- Transitive(isPartOf)
- Asymmetric (isPartOf)

Meta-semantics

- ObjectProperty
  - Smof_id :123
  - Smof_name: isPartOf
  - isTransitive: yes
    - isELDepreciated: No
  - isAsymmetric:Yes
    - isELDepreciated: **Yes**
  - isELDepreciated:No

Figure 3-3: Web3.OWL Metamodeling Example

property is defined and then described in the L^1 repository structures. It presents to the left a defined transitive and asymmetric OWL Object Property (*isPartOf*), and to the right the way it was described in the metadata repository. Explicit elements (in this example, related to the OWL 2 EL profile) are shown to indicate the depreciation status of the property. For instance, in this illustration, Object Property is not depreciated in EL, neither is its transitivity characteristic; however, its asymmetric characteristic is depreciated in EL (since this profile does not support asymmetry).

While the extensions suggested within the scope of Web3.OWL are not intended to fully answer the mentioned RFP's complete requirements, they still address a number of its main requisites.
Moreover, the simple fact that the RFP was submitted proves that the framework's approach that opted for extending SMOF (more that 3 years before the actual date of the RFP submission) was indeed the right approach.

3.3.2 Ontology Modeling Overview - Introducing ParOnt

As the challenge of capturing the maximum set of domain knowledge that is commonly understandable is presented, the adopted approach to model the domain canonical ontology profits from the high level of expressiveness promoted by the framework. Parenting resources of information are obtained via adequate references and domain experts. Based on this reliable information, engineers, modeling and metamodeling experts capture and model the knowledge, notwithstanding awareness that a certain level of that knowledge is not for imminent use.

For the particular purposes of the domain of interest, the sociomedical parenting ontology ParOnt was established with the expressivity of the OWL 2 $\mathcal{SROIQ}(D)$ DL language. With OWL 2 being recognized as the most adequate and powerful ontology modeling language, $\mathcal{SROIQ}$ is the highly expressive, yet decidable language covering extensive sets of grammars and constructs that form the basis of OWL 2.

As part of the Brain-to-Society (BtS) research endeavors that call for a whole-of-society (WoS) transformation centered on the individual [25], the Childhood Obesity Prevention [Knowledge] Enterprise COPE had already been established and later on extended to OWL 2 for recommender systems [28]. Childhood obesity surveillance being a subdomain of the parenting education domain with health, nutrition and cross-sectional shared areas, it was natural to use COPE as a trustworthy ground serving as a base for the development of ParOnt.

While building ParOnt, several recommended steps for enhancing parenting education were taken into consideration; those included, first, consolidating findings and building consensus among researchers and practitioners involved in issues of parenting, and second, ensuring that the emerging knowledge is disseminated in careful, extensive, and effective ways.

ParOnt is subject to continuous and incremental development. Its subdomains cover pregnancy aspects, babies and children's development, routines, activities, behavior and discipline, health and nutrition, and include inter alia:

- Family-related aspects (only child, divorced parents, adopted child, sibling rivalry, etc.)

- Character traits (bad manners, materialistic, bad sports, insensitiveness, indecisiveness, etc.)
- Behavioral aspects (defiance, indecisiveness, swearing, yelling, lying, etc.)
- Emotional traits (angry, fearful, dependent, shy, perfectionist, etc.)
- Social scene behavioral aspects (gathers a particular set of behavioral aspects, related to behavior with friends, in the daycare or at school, etc.)
- Nutritional aspects (based on baby or child's age, controversial nutritional habits, recommendations, etc.)
- Health aspects (diseases and deficits, special needs, eating disorders, learning disabilities, etc.)
- Pregnancy Characteristics and Challenges (various stages characteristics, risks, symptoms, advice, etc.)

When deemed necessary and advantageous, the ontology conceptualization phase capitalizes on key research outcomes related to the modular reuse of ontologies [21], while applying the different services that guarantee safety and locality.

Definition 1 *Σ representing the vocabulary, i.e. the set of concepts that form the axioms of an ontology, a module $\mathcal{M}$ for an ontology $\mathcal{O}$ w.r.t. a signature Σ is an ontology $\mathcal{M} \subseteq \mathcal{O}$ such that $\mathcal{M}$ and $\mathcal{O}$ entail the same axioms over Σ [21].*

While modularization strategies to be explored in the coming chapters are used within Web3.OWL for various purposes, at this level, the framework benefits from signature-based modularization to obtain modules for knowledge reuse. As for the rest of the purposes, they will be explored in the subsequent two chapters.

Thus, a possibility of incorporating parts of reliable and well established ontologies that happen to be already developed for certain inter-related sub-domains is offered.

Based on defined signatures held appropriately and accordingly with the metamodel specifications (in the extension package related to alignment and modularization), modules are extracted and imported into the canonical main ontology.

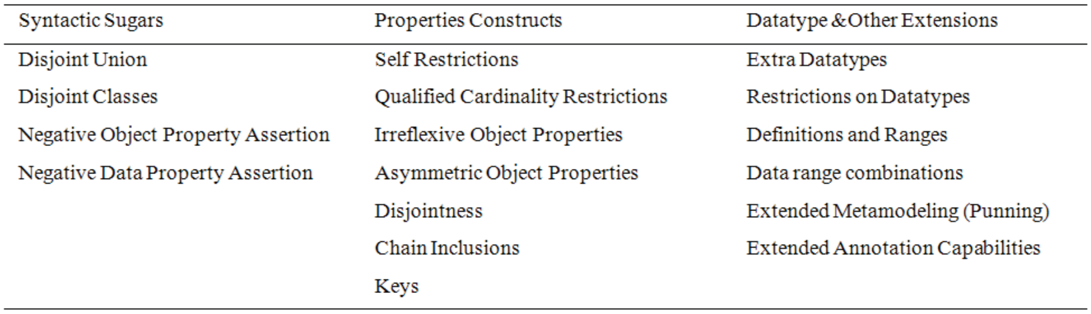
As examples of typical ontologies for knowledge reuse, SNOMED presents potentials for certain medical sub-domains, whereas FOAF offers the benefits of its widely adopted attributes and elements for persons, agents, documents, etc. (distributed between the FOAF core and the Social Web extensions).

In short, ontological activities and non-standard reasoning services that point toward strengthening the parenting knowledge base – by consolidating knowledge and building coherent consensus related to controversial issues – are put forward and promoted by the ontology engineering approach.

Hence, the ontological representation of mommy bloggers and of SN users in general, depicts them along with their children as individuals, grouped according to their rightful or wrongful beliefs and opinions, their traits and characteristics, their physical activities, locations, behaviors, needs and recommendations, among other criteria. This KB approach sets the stage for the Social Networks' engagement in initiatives for higher level of parenting education in which existing knowledge is exploited more effectively and coherently, and critical needs of children, parents, and families are addressed more consciously and comprehensively.

While Chapter 6 will enclose more details on ParOnt's contents and particular classes and axioms, Table 3.1 and the following subsections provide highlights on the main OWL 2 features incorporated into it, along with corresponding justifications and applicable examples.

OWL 2 features fall into the following categories: syntactic sugars, aiming at a simpler way of conveying information, new properties' constructs for maximized expressivity, extended datatypes support, and metamodeling and various other capabilities. OWL 2's extended annotation capabilities are used in favor of the verbalization of the ontology's axioms, an important step that is useful within the tagging process explained later on.



Syntactic Sugars	Properties Constructs	Datatype & Other Extensions
Disjoint Union	Self Restrictions	Extra Datatypes
Disjoint Classes	Qualified Cardinality Restrictions	Restrictions on Datatypes
Negative Object Property Assertion	Irreflexive Object Properties	Definitions and Ranges
Negative Data Property Assertion	Asymmetric Object Properties	Data range combinations
	Disjointness	Extended Metamodeling (Punning)
	Chain Inclusions	Extended Annotation Capabilities
	Keys	

Table 3.1: Summary of ParOnt OWL 2 Implemented Features per Category

Syntactic Sugars

For the sake of some patterns' simplicity in writing and expressing, new constructs known as "syntactic sugars" were introduced in OWL 2. These constructs do not affect the expressiveness level, they rather have an impact on the efficiency of processing. They also come in really practical for the actual domain's application needs, given the nature of information available in SN websites, and the writing style adopted in the blogosphere.
For instance:

- Using the *DisjointUnion* construct, diseases can be expressed as being exclusively the union of disjoint sets of disease types (cardiovascular, endocrine, gastrointestinal, musculoskeletal, etc.).
- The *DisjointClasses* construct correspondingly complies, for instance, with the definition of dysmorphic disorder, depression, as mutually disjoint subclasses of the "psychological disorder" class.

- *NegativeObjectPropertyAssertion* and *NegativeDataPropertyAssertion* are used for properties that are found not to hold for certain individuals, they come in handy for instance to express that a certain child does not perform any physical activity, or is not followed up by a nutritionist, etc.

New Properties Constructs

OWL 2 witnessed major developments in properties' constructs, overcoming previous weaknesses in OWL 1.
More particularly:

- The Property Qualified Cardinality Restrictions are proven to be very useful for the definition of classes similar to the class of children individuals who suffer from at least (*ObjectMinCardinality*), at most (*ObjectMaxCardinality*), or exactly (*ObjectExactCardinality*) a certain number of abnormal psychological behaviors.
  Equivalent constructs:
  (*DataMinCardinality*, *DataMaxCardinality*, and *DataExactCardinality*)
  are applied for data properties as well, for instance, with certain human and product specification concrete roles.

- The Self Restriction *ObjectHasSelf* allows the differentiation between individuals planning to follow a certain treatment by themselves, as opposed to those who are seeking recommendations for their friends and/or family members. It thus makes it possible to connect the individuals to themselves through their conscious planning or regulatory behavior properties.

- The OWL 2 properties *IrreflexiveObjectProperty* and *AsymmetricObjectProperty* are assigned for instance to the abstract role denoting a person being followed up by some nutritionist.

- Disjoint Properties are also crucial for the domain, particularly for roles involving a good vs. a bad effect or behavior for example.

- Property Chains Inclusions are also employed, for instance, to propagate certain family-related roles, and create useful rule-like axioms.

- With regards to OWL 2 keys, the *HasKey* property was applied for instance for the email specification of a Human entity, to enforce a unique identification. In other words, to the same email address, corresponds the same individual, and therefore any individuals detected with the same email address are considered the same. This example comes in handy for the Social Networking applications, since it allows the retrieval of the same individuals across different platforms and SN sites.

Extended Datatype Capabilities

OWL 2's novel extensions to datatype capabilities are deemed considerably profitable for the parenting domain:

- Extra supported data types (such as float, double, etc.) fulfill the needs of values applicable for instance to BMI (Body Mass Index), blood profiling exams, etc.

- Restrictions on datatypes offer additional constraining details, like minimum and maximum lengths (for strings), and values (for integers).

- Datatype definitions, data ranges and data range combinations provide extra flexibility and allow, for example, a smooth definition of age groups and ranges; of BMI values and applicable range levels for underweight, normal weight, overweight and obese children and adults; as well as of other medical information, like the definition of triglycerides, cholesterol, and blood sugar healthy and unhealthy level ranges.

- Data range combinations facilitate the representation of datatypes as complements, unions or intersections of other datatypes. Always in the scope of Humans and their characterizing criteria, adults can be defined as the complements of children, which in turn is the union of infants, toddlers, play age and school age humans.

Other Capabilities

Other various types of features were considered during the ontology's design; they are summarized next:

- Extended Metamodeling Capabilities in OWL 2, primarily introduced through "Punning", are revealed as convenient for the COPE ontology requirements. As an example, when modeling diseases along with their different categories (Cardiovascular, Endocrine, Musculoskeletal, etc.), certain categories turn out not to be as populated and complex as others, and do not stipulate pertinent required instances. For such occurrences, punning, represented by assigning the same identification for both the Class and the Individual, is very useful, specifically when these diseases are needed to be specified for Object Properties referring to them.
- Extended annotation capabilities, targeting different elements of the ontology (axioms, properties, etc.) and offering different benefits. Given the framework's collaborative aspect, these capabilities play a positive role. For instance, they provide extra logical information regarding who was responsible for asserting a certain axiom, and when that action took place.

The previous subsections highlighted OWL 2 enhancements that had a direct impact on the modeling of the domain ontology ParOnt. The fact that some other remaining improvements were not covered herein does not underrate their influence on the overall framework's approach and design.

3.4 Framework Data Analysis Overview

The ontology-driven data analysis approach described in this section highlights the role of data mining via Natural Language Processing (NLP). Furthermore, it summarizes the role of semantic tagging and ontology population activities that precede reasoning and querying for decision-making.

The aim of the framework's data analysis is to gather structured and semantic information on individuals based on the blogs and information they read and/or write. The gathered information is obtained based on textual data annotations, and used to populate the KB ontology. Extensional queries on individuals' profiling and opinion-mining profit from this information.

With textual blogs and forums being the main source of online information processable by the framework, the data mining methodology puts NLP in use. The framework being ontology-driven, the essential requirement is to analyze data according to ontological knowledge. The exploited NLP tool (to be shortly presented) is able to load an ontology as a language resource and process data according to its taxonomical hierarchy and lightweight properties. In other words, it has the ability to produce ontology-aware NLP outcomes (textual annotations). This textual analysis approach was thus deemed suitable for the framework's needs, appropriate and compatible with the rest of its flow components.

3.4.1 Highlights on Ontology-Aware Natural Language Processing

A lot of endeavors have been carried out for the purpose of extracting ontological (mostly RDF) knowledge from texts via Natural Language Processing. Recent works reported in [37] and [36] describe the process of extraction of RDF facts from unstructured textual data. In [45], the NIF/NLP2RDF project is aimed at the establishment, via ontologies, of a complete ecosystem of interoperable NLP tools and services (including the one used in the scope of this framework).

The next paragraphs describe the ontology-aware NLP approach that fulfilled the requirements of Web3.OWL, and contributed in the generation of the preliminary semantic tags used in the rest of the process.

The different Web 2.0 platforms (such as Twitter, Facebook, LinkedIn, etc.[1]) as

---

[1] twitter.com, facebook.com, linkedIn.com

well as conventional Web logs (blogs), wiki and forum websites all form adequate sources of online SN data to be exploited by the framework, with different levels of availability. The framework specifically relies on blog and forum posts due to their accessibility facilities, and predominantly on “mommy blogs” given the nature of their data (describing parents’ preoccupations, children problems, activities, behaviors, etc.) which is of a particular relevance to the parenting sociomedical domain.

The data parsing layer targeting semantic information extraction from the available SN data is based on GATE (the General Architecture for Text Engineering) [22], one of the most mature NLP platforms. The effectiveness of using GATE for ontology-aware language processing has already been demonstrated within several studies and projects, such as KIM[2] [76]. The latter is a platform for Information Extraction using GATE and targeting large-scale semantic annotation and ontology population based on the PROTON[3] lightweight ontology.

In a nutshell, GATE operates by running a processing pipeline that forms its Information Extraction components (known as Processing Resources or PRs, that are called according to a specific order), against a corpus, which represents a set of collected documents or web crawls. Ontological vocabularies used with JAPE[4] rules (Java Annotations Pattern Engine) form specific PRs that lead to textual annotations transformed into semantic tags.

These grammars and rules are supported by a multitude of IE algorithms, including Named Entity recognition: NE algorithms identify values and names related to people, dates, places, organizations, amounts, etc. They generally make use of sophisticated processes, as well as Gazetteers, which are predefined and customizable lists of appropriate values corresponding to the domain in question.

As a consequence of this NLP phase, a (semi) automatic creation of the semantic annotations that correspond to the available medical and social ontological knowledge is generated.

---

[2] www.ontotext.com/kim

[3] http://proton.semanticweb.org

[4] http://gate.ac.uk/sale/thakker-jape-tutorial/GATE%20JAPE%20manual.pdf

3.4.2 Semantic Tagging, Ontology Population and Reasoning Services

Ontology-based information extraction aims at retrieving and automatically populating ontologies with the valid instances from the texts being analyzed. Instances will accordingly appear in the ontology in one or more locations, given the multidimensional nature of an ontology.

In straightforward scenarios for which moderately expressive basic OWL-DL axioms were sufficient, the OwlExporter resource [88] was used; this particularly interesting GATE resource charts the way for ontology population through an easy mapping between NLP analysis pipelines and OWL ontologies.

For the semi-automatic tagging-supported scenarios, a less straightforward approach for ontology population is adopted by Web3.OWL.

Elucidated more in Chapter 5, this approach suggests the usage of the NLP outcomes consisting of recognized ontology concepts and roles for the following purpose: identifying the signature elements related to them and inferring from the KB additional expressive axioms that are logically and conceptually related to them.

The inferred expressions are then used as tags suggested to the SN users. Through the SN user collaboration, by enabling the validation or negation of the offered suggestions, the collected tags are analyzed to serve for ontology population. Upon reaching the target number of populated individuals, i.e., a number deemed reasonable for the decision-making required for the UC scenario in question, ontology reasoning takes place.

Typical reasoning services, known as standard reasoning services, are elucidated in [6], and include:

- Satisfiability (concept and role consistency checking)
- Classification and subsumption
- Instance retrieval and instance checking

- Inference discovery
- Query answering
- Rule validation and processing

That said, these reasoning services are characterized by being intensional or extensional, or both:

- Intensional reasoning concerns the algorithms that handle the *TBox* and *RBox* aspects of the ontology. For instance, *TBox* traditional intensional tasks are concept satisfiability and concept subsumption.
- Extensional reasoning concerns the algorithms that handle the *ABox* aspect of the ontology. Once an instance is implicated in the standard reasoning task, extensional reasoning is called upon in order to support the intensional reasoning (for example for query answering, instance checking and retrieval, consistency of the whole ontology).

Considering the UC in question, an essential step of the process consists in estimating the most appropriate fragment based on the reasoning activity implied.
Typically, OWL 2 QL is mostly suitable for extensional scenarios involving large ABoxes, and so is OWL 2 RL, particularly when scenarios require rules processing; alternatively, OWL 2 EL has proven its efficiency for intensional reasoning tasks of large TBoxes.

OWL DL on the other hand experiences severe limitations upon performing extensional reasoning tasks for largely populated ontologies. The reasons behind that are that first, its current algorithms are designed on purely intensional tasks, and no real optimization techniques were applied for their ABox services, and second, those same algorithms process their tasks in main memory, a fact that presents a bottleneck for ontologies with large instances.

Thus, when reasoning is applied to large and highly expressive ontologies, the trade-off with efficiency is to be taken into account. A special treatment is anticipated in order not to cross the border of decidability and thus ensure reasoning termination. This approach will be discussed more thoroughly in the next chapter, but very briefly, it seeks a convenient compromise that guarantees decidability while preserving a decent level of expressivity. The ParOnt ontology is projected according to the most appropriate level of expressiveness that suits to the scenario in question, often one of the OWL 2 tractable fragments.
The just introduced reasoning services are then applied to ParOnt derived ontologies; they form the means by which are attained the outcomes of the decision support systems.

Based on the projected languages and fragments for the scenario in question, redirection mechanisms allow the employment of appropriate mostly advanced and powerful DL-based reasoners and rule engines reported in the literature and available for exploration. For example, Pellet [79] can handle OWL 2 DL and RL; RacerPro [43, 44] can manage a subset OWL 2 DL and OWL 2 EL; ELK [55] is very efficient with OWL 2 EL ontologies; HermiT [69] and FaCT++ [85] can cope with OWL 2 DL. On the other hand, the Jena framework [54] and the database Oracle 11g enable the processing of OWL 2 RL rules, whereas Quill [83], a TrOWL [84] component, provides OWL 2 QL querying capabilities.
The already introduced OWLLink API assists in the access to the different reasoners. OWLLink [60] provides a flexible and extensible protocol that handles the communication between different OWL reasoning systems, facilitating the configuration of reasoners and the transmission of OWL 2 ontologies by client applications.

3.5 Putting it all Together - Framework Infrastructure and Workflow Overview

This section explores the way semantics and reasoning are incorporated into all essential stages of Web3.OWL's workflow. It thus presents, side by side, the metamodel and the essential elements it comprises, and the proposed strategy that relies on the ontological data and metadata, in order to achieve its goals.

Figure 3-4 provides a comprehensive overview of the platform infrastructure (left panel) and adopted workflow (right panel). After presenting its contents, this section explores different modes of scenarios applicable in Web3.OWL.

3.5.1 Knowledge Base Platform Infrastructure

The contents of this subsection are in accordance with Figure 3-4's left panel.

As a rule of thumb, the different Web 2.0 platforms such as the well-known previously mentioned SN sites, as well as conventional Web logs (blog), wiki and forum websites all form adequate sources of online SN data exploitable by Web3.OWL. Levels of availability naturally differ by type of source. Throughout the explored framework, blog and forum posts form the sources of data mostly relied upon, and that due to their wide accessibility facilities. In particular, sites known as "mommy blog websites" contain information extremely relevant to the parenting domain of interest, and thus present abundant and significant data disposed for thorough processing.

To semantically analyze this data, Web3.OWL sets up a workflow that ensures the coordination between its different components. The semantic layer managed in the conceived repository makes the collaboration process a possibility.

To cope with data mining requirements, metadata found in the KB repository has already been presented. It is used to describe, map and interrelate:

- Supported ontology modeling languages, such as RDF, OWL, OWL 2.
- Available sub-languages, fragments or profiles, such as OWL DL, EL, RL, QL,

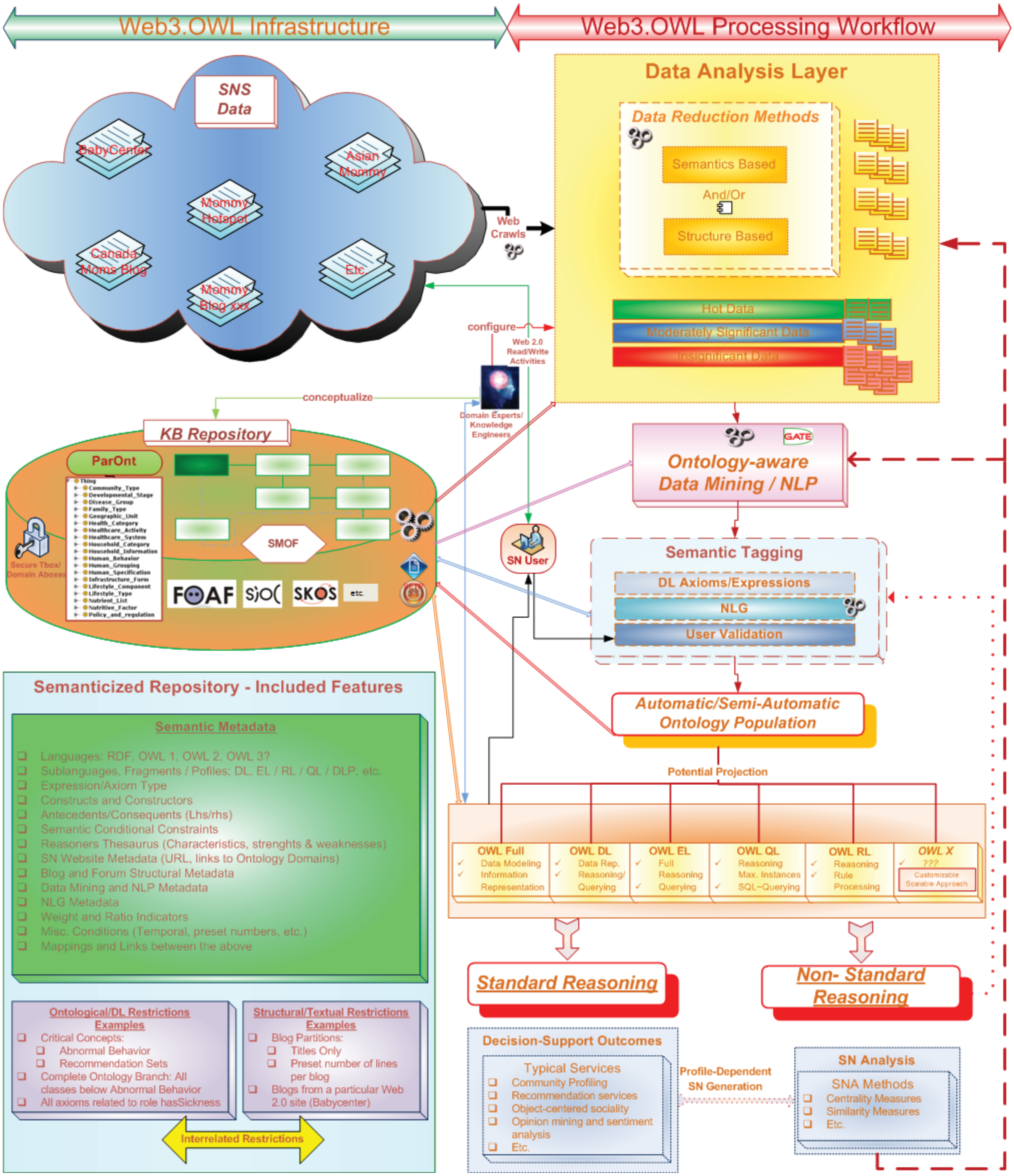


Figure 3-4: Platform Infrastructure & Processing Workflow

etc.

- Signature Elements retrieved from the canonical ontology.
- Use Case (UC) information, including type, Competency Question (CQ), recommended applicable profile, projected ontology, etc.
- Expression/Axiom Types, such as class equivalence, subsumptions, property, data and cardinality restrictions, etc.
- Constructs and Constructors types, denoting axioms constituents and their characteristics.
- Antecedents and Consequents, distinguishing between left hand side and right hand side constructs.
- Semantic Conditions forming sets of distinguishable restrictions having an effect on levels of expressivity and profiles.
- Reasoners thesaurus, including each engine's characteristics, strengths and weaknesses.
- Particular indicators and identifiers highlighting the weight and relevance of certain ontological elements (branches, concepts with/without children and related axioms, etc.), based on the context or scenario in question.
- SN Web 2.0 platforms metadata, covering a website's main characteristics, an estimation of the number of its adhered users, its URL (allowing automatic web crawling activities), and when applicable, its associated domains or subdomains.
- Data Mining and NLP metadata, properties of analysis and annotation methods.
- Blog and forum metadata, identifying textual partitions, characteristics and elements (titles, paragraphs, lines and sentences).

- Significance ratios for ontological and blog data.
- Miscellaneous useful conditional elements: for instance temporal expressions, preset criteria and numbers to associate to other metadata elements (number of tags, number of populated individuals, duration before timeout,...), etc.
- NLG (Natural Language Generation) metadata, including links to automatic tools, in addition to prepared translations of the ontological properties, concepts and expressions (particularly for complex axioms).
- Mappings and links between all of the above, concentrating on connections jointly with the canonical ontology elements on the one hand, and the SN ontologies that hold relevant information on the other.

These elements are part of the data repository whose structures are described in Appendix B.

They come on top of the repository's canonical ontology which, as conceived through a cooperation between domain experts and knowledge engineers, has to remain well-managed, maintained and secured. The repository moreover supports and interlinks projected and available standardized ontologies, including the widely known and populated social ontologies (FOAF, SIOC, etc.).

Appendix B contains the major KB data structures holding this information.

3.5.2 Social Data Engineering Workflow Interpretation

In line with Figure 3-4's right panel, this subsection aims at describing the framework's workflow and processing that manage the SN data and lead to the decision-support and recommender systems outcomes.

The workflow details highlighted herein are particularly about actions and scenarios performed by knowledge engineers who are aware of the logical and structural characteristics of the framework, and have "administrator" privileges. Scenarios performed on the other hand by regular SN users are less complicated and involve fewer steps (usually typical query-answering use cases).

Thus, a framework's administrator user, upon dealing with a particular UC, is capable of identifying, based on the existing metadata, whether its corresponding signature and ontological model is available in the KB. If not, an effort will be initiated to make the necessary arrangements in terms of ontology and metadata modeling and UC-specific configurations.

To deal with traditional constraints related to the size of the social data and to the level of complexity of the ontology's semantics, the framework has the possibility to build on the interlinking process and resource description and classification achieved through the social semantic ontologies SIOC, FOAF, SKOS.

By setting up combined semantic and structural criteria and conditions, the possibility of applying data reduction per scenario is brought into play. In other words, the massive exploitable social data is reduced since only a subset of the data is relevant to the UC scenario.

For instance, blogs instantiated and described in the above ontologies are sorted out based on combinations of criteria that depend on the case study in question:

1. Blogs tagged in SIOC according to the most significant ontology classes (signature elements for example).
2. Blogs tagged in SIOC according to less significant (but still relevant) ontological concepts.
3. The blogs and forums satisfying a certain chronological period.
4. The blogs and forums relevant to a particular SN site (or blogger) that is known to be mostly dedicated to (or expert in) the domain or sub-domain in question.

Depending on how the data ends up sorted out according to this input criteria, three main data subset groupings are considered:

- “Hot Significant data”, consisting of the mostly impacting subsets of the original data. This is the subset that is satisfied at least by the first criterium.

In general, the amount of resulting data falling in this category is the least among the three.

- “Moderately Significant data”, with an average impact factor on the rest of the process. This is the subset that is satisfied by at least one of the last three criteria (2, 3 and 4).
  In general, a considerable amount of data ends up under this category.
- “Insignificant data”, with an extremely low impact factor. This is the subset that is not satisfied by any of the criteria.
  In general, a huge amount of data ends up falling under this category.

Following this categorization, the knowledge engineer has the choice of applying a different data mining strategy to each subset, as deemed advantageous to his scenario purposes. The third subset of insignificant data is simply ignored; advanced semantically aware data mining and NLP algorithms are applied to the first set of “hot” data, and reasonably complex ones to the second set of moderately significant data. GATE resources and plugins are exploited for the information and annotation extraction involved.

Moving forward in the workflow, the semantic tagging phase is reached.

The provisional output resulting from NLP strategy consists mainly of constructed templates of preliminary non-validated sets of semantics. These include identity relations and rules that are made available in form of “suggested tags” in natural language; the SN user has the optional role of confirming, correcting and/or refuting them.

Although not mandatory, this semi-automatic tagging suggestion approach that requires a user intervention is deemed extremely advantageous. It allows to overcome the NLP technology’s severe restrictions upon dealing with complex and expressive vocabularies and ontologies. Moreover, it leads to a more reliable ontology-population phase in which the individuals are correctly assigned to their roles and concepts in the ontology.

Once this stage is reached, and the ontology is populated, standard reasoning services are applied according to the suitable profile. Results are then interpreted to form the outcomes of the decision support services in question.
As for the rightmost lower block mentioning Social Network Analysis (SNA) methods, it reflects the logical networks of SN users that are formed. Such networks are deduced based on the classified ontology, and can form incentives for the knowledge engineer to trigger the scenario for some other correlated UC. The application of the SNA methods and the relaunching of the process remain part of some future work and scenarios that are still under development.

3.5.3 Typical Supported Use Cases and Scenarios

The framework supports different workflow modes for the support of its use cases, mostly depending on the UC user, and on the UC context and objective. The two main workflow modes are shown here, and subsequent chapters will illustrate and exemplify others as well.

Basic Workflow

This is the standard process executed within the framework based on a predefined configuration, for example at a certain periodic basis (Figure 3-5). Such configurations are usually set by the knowledge engineer; the process mode can also take place upon request for some advanced scenarios executed by the knowledge engineer.

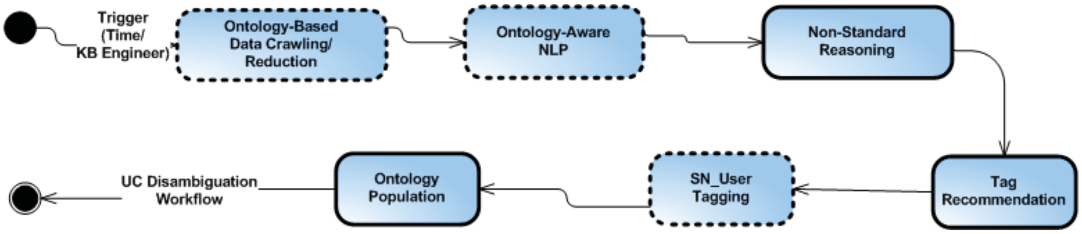


Figure 3-5: Competency Question Basic Workflow - Up to UC Disambiguation
Dashed blocks represent components that are not required at every occurrence of a scenario

1. **Data Crawling and Reduction:** triggered according to predefined conditions; search-based data crawls and ontology signature-supported reduction techniques.

2. **Ontology-Aware NLP:** like Named Entity recognition, Part-of-Speech tagging with ontology-supported grammars.

3. **Non-standard Reasoning:** essentially, the KB contents are referred to in order to deduce relevant axioms flagged as "potential tags" and matching the previous step's ontological outcomes. Also note that this can take place in light of the previous steps, or independently. For example, based on some signature provided by the metadata knowledge engineer, modularization and/or abduction techniques are conducted to identify interesting tags.

4. **Tag Recommendation:** identified semantic axioms are recommended as tags, through natural language generated expressions. Tags are recommended on priority basis (according to their relevancy with respect to the UC outcome), furthermore, tags have a suggestion weight, so that if actual tagging does not occur (after a certain predefined period), the knowledge engineer can enforce the population process relying on the NLP and non-standard reasoning outcomes.

5. **SN User Tagging:** validation, negation or correction of suggested tags by the SN user (mommy blogger).

6. **Ontology Population:** population of the ontology, after a tag analysis phase that takes into consideration tags priority, weight, and frequency, when the case applies. Otherwise, population relies solely on the results of non-standard reasoning.

As a continuation of this flow, the typical Competency Question mode to be presented next usually arises.

Competency Question (CQ) Regular Workflow

This is the standard CQ process executed by the framework on demand. This process mode can take place based on executed queries and requests either from the SN user, or the knowledge engineer.

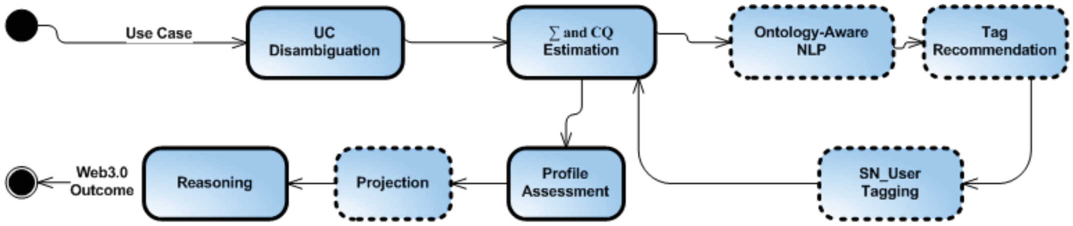


Figure 3-6: Competency Question Regular UC Disambiguation Workflow
Dashed blocks represent components that are not required at every occurrence of a scenario

1. **UC Disambiguation:** upon detecting the UC, its disambiguation process is carried out in order to determine the UC objective and applicable profile.
2. **Σ and CQ Estimaton:** an attempt to check and identify the UC's signature and Competency Question directly against the KB's existing contents.
3. **Ontology-Aware NLP:** if the KB existing contents are short of the UC signature and CQ, Named Entity recognition, Part-of-Speech tagging with ontology-supported grammars are applied.
4. **Tag Recommendation:** identified semantic axioms are recommended as tags, through natural language generated expressions. Tags are recommended on priority basis (according to their relevancy with respect to the UC outcome), furthermore, tags have a suggestion weight, so that if actual tagging does not occur (after a certain predefined period), the knowledge engineer can enforce the population process relying on the NLP and non-standard reasoning outcomes.
5. **SN User Tagging:** validation, negation or correction of suggested tags by the SN user (mommy blogger).

6. **Profile Assessment:** based on the identified signature and CQ, the objective of the UC and the underlined type of reasoning is determined (intensional vs. extensional). Consequently, the KB contents and the projection and deprecation-revealing algorithms point out the “theoretical” appropriate reasoning profile and corresponding reasoner.
7. **Projection:** in case the KB does not already contain a version of the projected ontology.
8. **Reasoning:** the typical standard reasoning services (instance checking, query answering, etc.) are invoked to meet the required Web 3.0 outcomes.

3.6 Framework Highlighted Challenges and Advantages

The framework’s most important challenge is to be able to coherently control the underlined technologies and manage the domain requirements. This however is unsurprising since Web 3.0 is by definition a web of collaborating technologies and disciplines. Any holistic framework proposed for Web 3.0 has to take notice of that fact and incorporate its key technologies and players.

Being ontology-driven, Web3.OWL’s major advantage in this regard is that it relies on standard grounds for representing terms and concepts, and classifying them according to their expressivity and features, thereby supporting easy transmission and interpretation of data for various applications.

Accordingly, a basic framework prerequisite is the ontology-modeling know-how and the maintenance of the modeled knowledge in the extended SMOF metamodel. Any deficiency at this level will lead to serious implications at later stages of the process.

Thus, having described Web3.OWL and examined its features, an evaluation of its challenging aspects reveals the following considerations:

- An evident challenge for an ontology-driven framework is to master the methods and best practices of ontology modeling and management. Having established the ontology as an interoperability-enabler across the multiple framework components and disciplines, it is essential to set up techniques to manage its complexity, size and constraints. The expressiveness management approach with projection and modularization based on rigorous logical formalisms plays an important role at this level.

- The framework's exploitation of other disciplines, including data mining, NLP and NLG techniques makes it apt to inherit these disciplines' accuracy and ambiguity constraints. The semantic tagging layer overcomes these limitations; this layer depends on the taggers' willingness to cooperate. The Web 2.0 community promotes high expectations, based on evidence, as to the gradual increase of this cooperation willingness.

- A very specialized level of expertise and know-how in ontology and metadata modeling is required; such expertise cannot be easily made available. This is a known drawback of MDA. While there exists an exhaustive reliance on modeling and correctness of critical metadata and configurations (for semantics and scenarios), errors and inconsistencies at their level have important effects on the whole flow.

  There is also a reliance on the expertise of the knowledge engineer to deal with and correct any potential inconsistencies in metadata and ontological information.

- Web3.OWL's full implementation brings in several complications, especially if all possible configurations and possibilities are to be made available. The incompleteness in the metamodeling requirements prevent the adaptation of transformations that should be made available through the standardized components. The project could also be strongly boosted by a certain level of industrial support, or by particular Web 2.0 sites pushing generated tag recommendations in a cloud computing environment. Chapter 6 reports the implemented parts of the framework.

- While Web3.OWL benefits from the enhancements and novelties of the Semantic Web and Description Logics, it is also constrained by limitations in the areas that are least subject to the efficient advances (for instance OWL Full support; querying facilities for certain complex roles and axioms; etc.).

  Again at this stage, the underlined projection facilities and algorithms by modularizing and/or trimming down the knowledge look for the appropriate least depreciated subsets, indirectly overcoming the aforementioned constraints. In parallel, the framework suggests methods to make use of extra expressiveness in other contextual scenarios.

  While the field is in constant progress with an evolutionary research effort, the continuously positive impacts will automatically be reflected in the framework's collaboration platform. This was the main aim behind its standard-based conception which included extended means and methods for interoperability.

  For example, if new or variations of the existing OWL profiles are proposed along with applicable reasoners that are specifically designed to efficiently process them, the existent metamodel will allow its integration in the framework. Similarly to how the current fragments are identified and projected, the data in the meta-semantics will allow the extraction of other potential fragments, to their associated most suitable reasoners, use cases and so on.

  As another example, if a new SN ontology proves its usefulness and proliferation on the net, the means to integrate, describe and map it to the existing ontologies in the repository are all available.

On the other hand, apart from the already underlined Web3.OWL characteristics, it is worth re-emphasizing its underlying principle:

A knowledge base for reliable sociomedical modeled facts enriched with semantically engineered social data. This data consists in populated individuals describing the community, and the categorization of resources according to specific ontological axioms.

As the KB Web 3.0 data is made readily accessible for further extensive reasoning and analysis, its reached outcomes surpass by far the sum of its social and semantic data components. The whole process typically leads to significant services, recommender and decision support systems.

Taking into consideration the applicable involved reasoning, the opportunity of identifying, creating and expanding social and semantic networks is presented.

Implemented algorithms allow opinion mining, detection of ties and similarities between people, leading to connections via shared interests or any possible common ground areas.

Social Networks can be deduced through the users' joint actions and interactions, their created, commented upon, linked to, or similarly annotated contents.

Many aspects of the conclusions and findings are thus related to the concept of "object-centered sociality", which connects people via the common interests associated with their occupations, hobbies, jobs, etc.

We can further highlight the following analogous potentials and benefits of the conceptual framework. They serve the purposes of recommender and decision-support systems:

- User profiling, clustering and segmentation based on certain traits and criteria, all of which are endeavors considered closely related to opinion mining and sentiment analysis undertakings

- Tracking processes to identify a user's Web history from different Web 2.0 platforms, outlining this user's general overall contributions to the Web and reporting their different activities, goals and problems

- Improved quality of the search process, with ego-centric algorithms and searches to identify a key user's associated or closely related nodes, as well as community detection algorithms to trace two or more key users' surrounding community

3.7 Conclusion

This chapter was dedicated for the holistic proposed framework, pointing out its global flow's details on the one hand, and its most impacting principles and components on the other.

Its role was to present the framework, i.e. the basic conceptional structure. It highlighted the big impact of the metamodel and the knowledge base, and the role of the ontology driving the rest of the components. It did not tackle implementation-related aspects and detailed use case examples; these will be left to Chapter 6.

The next two chapters will reveal a more analytic approach to present the aspects considered at the heart of the framework's originality.

Chapter 4

Expressiveness Management Approach

4.1 Introduction

When one of the main purposes of Web3.OWL is to promote the usage of high expressiveness and go beyond available RDF-based efforts, it is essential for it to adopt a particular approach to handle this expressiveness in such a way that it does not eventually become an obstructing factor of the framework, due to the tradeoff between performance and expressiveness.

This chapter makes a particular emphasis on the ontology-driven suggested method to model, arrange and manage the expressive knowledge. It starts with a recapitulation of the state of the art exploited and possibly exploitable efforts (Section 4.2), and in Section 4.3, it states the expressiveness drawbacks due to which the proposed framework calls for an approach to manage expressiveness. Following that, in Section 4.4, it revisits the already introduced SMOF-based extensions to develop the packages related to this chapter, and the way they handle particular expressiveness elements. Section 4.5 encapsulates the whole to then reveal and explain the algorithm that first processes the sublanguage projected ontology, and accordingly assesses the deprecated and depreciated knowledge. At the end, Sections 4.6 and 4.7 wrap up the chapter with discussions and conclusions.

4.2 State of the Art Exploited and Potentially Exploitable Efforts

Among the most promising efforts that demonstrate the feasibility and value of employing hybrid reasoning - by making appropriate usage of OWL fully-fledged reasoners jointly with OWL 2 RL dedicated reasoners - is what was recently presented in [90]. The proposed novel approach tackles the issue of query answering completeness on beyond OWL 2 RL ontologies, by delegating most of the computational workload to the RL reasoner, and leaving the intervention of the OWL traditional (fully-fledged) reasoner only for ensuring completeness when necessary, the challenge of the realization being in appropriately and efficiently determining when exactly the DL traditional reasoner should be called upon.

This novel approach, and due to time constraints, has not been exploited yet in the scope of the current framework. As part of some future work, it will be interesting to explore how well it could fit in the overall endeavors, and the customizations needed to particularly incorporate it.

That said, for the time being, the proposed approach to manage expressive large domain ontologies builds on the realizations achieved so far in the area of modularization as described in [21]. It mimics the way these modularization techniques are capable of extracting meaningful modules from ontologies. Given a set of terms, a method to capture all axioms relevant to the meaning of these terms and extract a "safe" module from the source ontology is made available, along with specific tools[1] and APIs (Application Programming Interfaces) to support it. More details on modularization will be explored in the coming sections.

---

[1] http://mowl-power.cs.man.ac.uk:8080/modularity/
http://www.cs.ox.ac.uk/isg/tools/ModuleExtractor/

4.3 The Need for an Expressiveness Management Approach

This section presents a recall of previously explicitly or implicitly mentioned obstacles that necessitate an approach to handle OWL expressiveness.

They can be summarized as follows:

- The intractability of OWL is a well-known issue. With an 2NExpTime worst-case complexity, scalability of size and expressiveness comes at serious performance expenses. In a typical decision-support scenario, a reasonable minimum level of efficiency in response time is expected.
  DL reasoning cannot guarantee this level for all queries. When DL reasoning is recommended for a particular scenario, a possible workaround is to first provide an initial query answer (within the accepted efficiency time), waiting for the DL reasoning (running in the background) to respond with the most complete answer.

- There is a multiplicity of reasoners, whether fully-fledged or specialized. Each reasoner's most suitable target application and expressiveness is neither straightforwardly nor automatically recognizable.

- The ability to move from one profile to another is not an easy task. The adopted strategy to model ontologies is to make an a priori assessment regarding the most suitable OWL fragment to adopt, and then build the ontology according to this fragment's syntax, to later on use the specialized reasoners mostly efficient for it. Currently, there are no means and tools to extract a fragment automatically from a more expressive developed ontology.

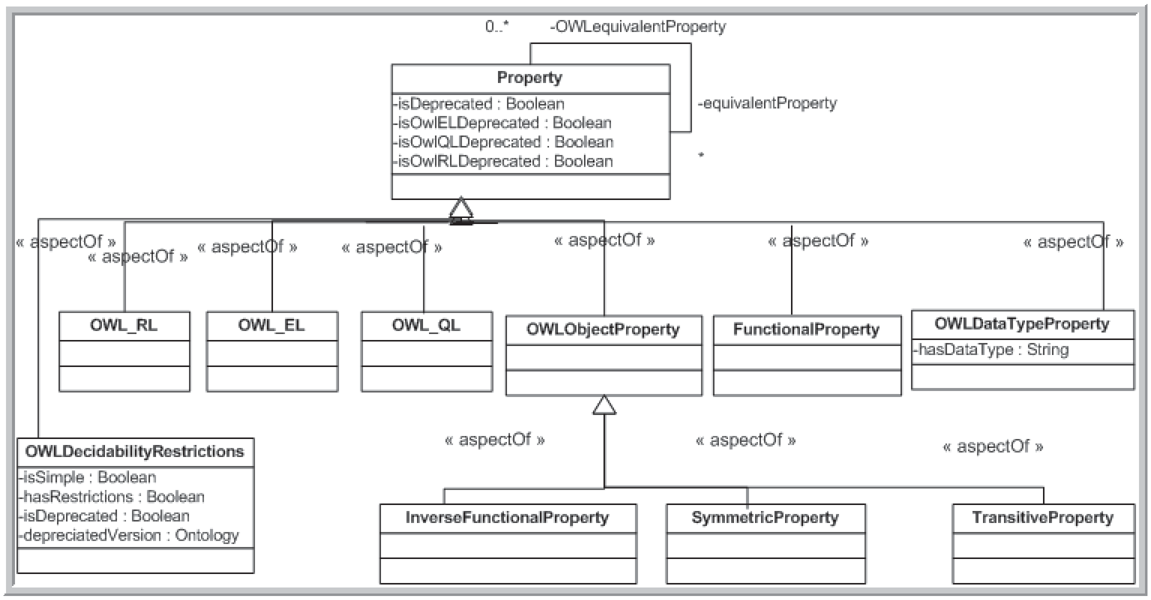


Figure 4-1: SMOF/ODM-Based Extension Extracts for Knowledge Classification

4.4 Knowledge Control and Classification via the Metamodel Repository

The key to the expressiveness management approach is embodied in the way knowledge is organized and “tagged” in the metamodel.

In order for the ontology to be able to drive the devised framework’s process, its constructs need to be identified and controlled to fulfill its requirements at different levels.

Figure 4-1 and Chapter 5’s Figure 5-2 provide an overview of the SMOF and ODM-based extended structures.
Appendix B’s lists the “SMOF” and “ODM” structures’ data dictionary.

These parts of the metamodel are extracted from the global set of extensions. Below is their brief description, including the particular aspects of the requirements that they address:

- Conformance means for tagging the knowledge and establishing conformance of the OWL 2 profiles. The OWL semantics specify that a property can be any of its illustrated subclasses, or a combination of these classes. A property can have more than one property characteristics (for example, a transitive and asymmetric property). Moreover, it can be expressed in more than one language and different semantics. For instance, a property type can be pertinent to EL, QL and RL specifications at the same time. The "AspectOf" generalization relationship type, as intended for OWL semantics in ODM using SMOF, allows this classification, and makes it possible for the properties to be managed and combined independently and in any order.

  This arrangement is also useful for instances in OWL as they can be classified by any number of classifiers.

- A list of concrete signatures, their association to ontologies and use cases. Accordingly, dedicated structures for signatures per ontology and use case definitions are referred to for the extraction of signature-based modules.

- A description of constraints and conformance criteria for additional concrete languages and translations between concrete languages. While these languages might not be explicitly supported (through reasoning for example), they nonetheless have equivalent uses that could be recognized and correctly processed by implementations. For instance, the extended metamodel supports natural language versions of the semantics, as well as mappings with RDF and RDFS, targeted at cross-language interoperability.

- Specific depreciation and deprecation conformance formalisms. For instance, some constructs might be obsolete in the context of a certain logical formalism. There are also restrictions on certain combinations of constructs. The "IncompatibleWith" relationship describes which combinations are invalid.

  On the other hand, restrictions can be resolved through particular depreciated versions of the knowledge. These are set at the definition level of the SMOF elements, and can be overriden in case they differ per use case scenario.

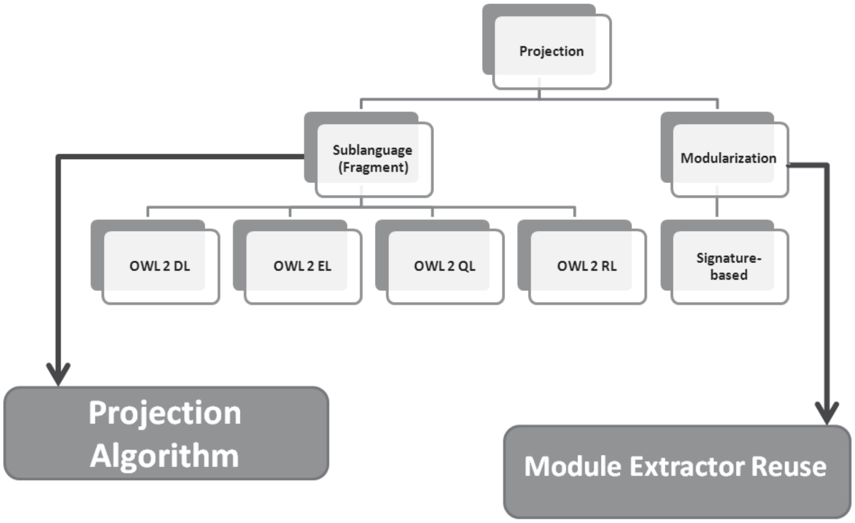


Figure 4-2: Global Projection Approach

4.5 Expressiveness Management Approach

This section tackles the different aspects of the framework's expressiveness management approach. It starts by giving an overview of the comprehensive approach. As a next stage, it emphasizes the role of modularization, addresses the projection based on expressiveness requirements, before moving on to illustrate and explain the semantics management algorithms.

4.5.1 Approach Global Overview

Since the approach privileges a maximized capture of knowledge, expressive data modeling is essential for describing and managing the domain. High expressiveness however comes at the expense of computational cost, and efficient automated reasoning remains a favored framework feature for most scenarios.

Web3.OWL proposes an approach that makes sure ontology complexity and reasoning constraints are bypassed through particular strategies for projection based on subsets of the language and/or signature elements (Figure 4-2).

In a nutshell, the expressiveness management approach acts as follows:

- Upon considering the scenario in question, signature elements are identified, either automatically (based on ontology-aware NLP) or according to feedback from the user (based on ontology-aware NLP and user validation through tagging); the purpose of the use case gets also disambiguated, along with a preliminary assessment of a potentially most suitable sub-language.
- An extraction of the signature's module takes place through signature-based modularization.
- When DL reasoning complexity issues are faced, due to the presence in the extracted module of complex OWL axioms (leading non-decidability, details will be explored in subsequent sections), the projection algorithms are invoked to reduce the expressiveness and trim the extracted module's knowledge expressivity according to a favorable sub-language.
- In cases where conflicts or ambiguities prevent the decision regarding the most favorable scenario, the approach resorts again to projection algorithms in order to assess the most favorable sub-language based on the inferred depreciation level.
- Once this stage is reached, the suitable reasoning engines are brought into play.

The approach thus relies on the SMOF metalanguage specification for the retrieval of signature-based axioms, their classification according to sub-language conformance criteria, and later on in the process for obtaining the corresponding favorable reasoner.

4.5.2 Emphasis on Modularization

Definition 1 in Chapter 3 briefly introduced the concept of modularization.

Consequently, signature-based modularization forms a non-standard ontology technique through which a module $\mathcal{M}$ extracted from an ontology $\mathcal{O}$ according to a signature Σ behaves exactly the same for all axioms α in Σ. In other words, the same

logical consequences are yielded for α in $\mathcal{M}$ and in $\mathcal{O}$.
Furthermore, if $\mathcal{M}$ is imported into $\mathcal{O}'$, the same logical consequences are yielded for α in $\mathcal{O}' \cup \mathcal{O}$ and $\mathcal{O}' \cup \mathcal{M}$, provided $\alpha \in \mathcal{M} \cap \mathcal{O}'$.

Web3.OWL capitalizes on the benefits of modularization for a variety of goals and utilities:

- Knowledge reuse, as already described within the ontology design phase of the previous chapter.
- Complexity management and efficient reasoning, the subject of the current section.
- Tags suggestion, tackled extensively in the next chapter.

Although Web3.OWL counts on advances in reasoning efficiency, it remains a fact that even with the most recent developments in reasoning techniques, performance decreases significantly with the increasing size of expressive ontologies.

The framework's strategy then is to refer to the meta structures on signature per use case. Modularization is considered advantageous and is carried out when the UC signature elements are identified as elements whose resulting modules will consist of limited ontological knowledge compared to the original ontology.

Alternatively, certain scenarios' reasoning tasks require full or most ontological knowledge. In such cases, the expressiveness approach invokes projection algorithms on the full ontology rather than on the modularized one.

4.5.3 The Role of Expressiveness Projection

The most characterizing aspect of the tackled management approach consists in the projection based on sub-languages and levels of expressiveness.

Definition 2 *Similarly to its definition in relational algebra, an ontological projection pertaining to Web3.OWL is a technique through which axioms that do not abide by the required expressiveness level are either completely discarded, or replaced by corresponding resolution axioms that respect the desired expressiveness level. Denoted by* Π*, the* $\Pi_{DL}(\mathcal{M})$ *represents the depreciated version of* $\mathcal{M}$ *according to the DL language decidability restrictions, as an example.*

Projection typically takes place on the modularized ontology, unless required otherwise for the particular scenario context.

The metalanguage structures provide facilities for automatically sorting out axioms according to their corresponding family of vocabularies. Appropriate vocabulary classification schemes are mainly obtained based on generic prototypes of the different constructs. Accordingly, a certain construct or axiom in a given ontology can be attributed to one or many families or fragments. When needed, particular language or profile axioms can automatically be projected and retrieved for appropriate processing and exploitation.

Figure 4-3 illustrates the decomposition and classification of sub-languages in OWL 1 vs. that of OWL 2.

It also hints at a potential OWL 3, indicating that although its structural decomposition is so far unknown, the evolution from OWL 2 to OWL 3, if it is meant to occur, will be similar to that from OWL 1 to OWL 2. The adopted metamodeling-based approach will be able to appropriately handle it.

As an example of the OWL 2 decomposition illustrated in Figure 4-3, any OWL 2 EL/QL/RL axiom is a valid OWL 2 DL axiom. Some of the EL axioms can simultaneously be QL or/and RL axioms and vice-versa. While the relationship between OWL 1 and OWL 2 is straightforward since OWL 1 is incorporated into OWL 2 (the latter simply supersedes the former), the case is not manifested similarly for the profiles. In fact, no containment relationship exists between any of the profiles; none of them can be considered as a fragment of the other.

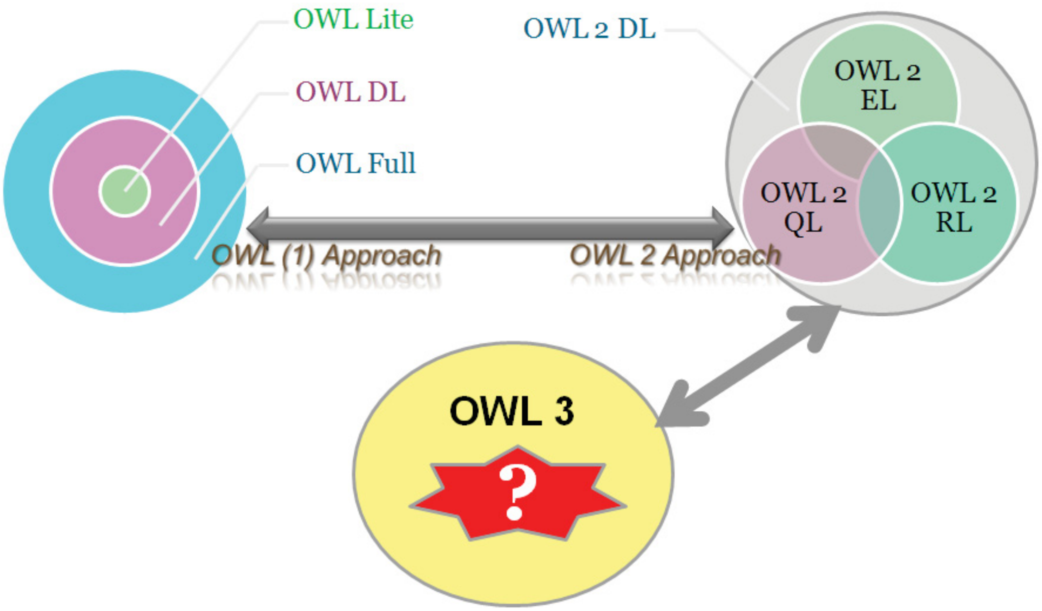


Figure 4-3: OWL 1 vs. OWL 2 Sublanguage Composition

With the available OWL 2 fragments, and the possibility of future prospects to be proposed (perhaps OWL 3 for example), the semantic categorization endeavor in the metamodel repository has been a particularly beneficial approach; it made possible the conceptualization of algorithms to facilitate the projection from one fragment to another, according to the framework stage in question.

Several factors thus contribute to attributing an axiom to a certain profile, or extracting all axioms that fall under a given profile. The nature of the feature, the included DL operation, the location of the constructs forming a certain axiom (at the left hand or the right hand side), and the availability or absence of certain elements which combination with already existing elements cross the border of expressiveness, are all among the most influencing aspects.
Accordingly, again in the set of “meta-semantics” entities, the groupings for fragments and languages applicable to a given axiom are identified. Additional pertinent conditions are pointed out as well, when applicable. These entities are designed in a way to be easily extensible, in order to encompass any future fragment or even language. One of the fundamental principles and properties of the described approach is

in fact to be predisposed for smooth potential upgrades and customizations, bearing in mind that OWL and the Semantic Web field are in a continuous evolution process.

The restrictions and specifications on decidability and efficiency, along with the tables to follow have been compiled after the analysis and reorganization of information based on several sources. They include, inter alia, the OWL 2 books and specifications [47, 81, 86, 68].

That said, with the aim of making the approach's description easier, five main levels of expressiveness have been considered:

- OWL 2 DL Full, which comprises all expressive axioms, including the syntactic restrictions that usually put decidability in jeopardy. Examples of such restrictions are:
  - Cyclic property chains (taking into consideration the simple vs. the composite aspects of roles).
  - The already introduced "punning" extended metamodeling capabilities.
  - The transitive properties in cardinality restrictions, and the transitivity and asymmetric features combined for a property.
- OWL 2 DL with resolved restrictions, yet containing non-deterministic axioms that cause guessing and backtracking during reasoning. These features are the main sources for intractability and undecidability, they are therefore forbidden in all the profiles:
  - Disjunction, or negation and conjunction combined in the same construct.
  - Maximum cardinality restrictions (for greater than 1 cardinalities)
  - Existential and universal restrictions combined in superclasses
  - Non-unary finite class expressions (aka nominals, again usually for more than one nominal) and certain particular datatype expressions.

- OWL 2 EL, QL and RL, abiding by the expressiveness rules and restrictions in each of these profiles.

Table 4.1 summarizes the way axioms are grouped in the different OWL 2 profiles.

For a more extensive overview including for instance particular supported datatypes and conditional specifications, the reader is referred to the references mentioned in the previous page.

OWL 2 Profiles Expressivity Summary					
Expression Type	Expression	Syntax	EL	QL	RL
Assertion Axioms	ClassAssertion	$a : C$	✓	✓	✓
	ObjectPropertyAssertion	$(a, b) : R$	✓	✓	✓
Class Expressions	Unary Predicate	A	✓	✓	✓
	Conjunction	$C_1 \sqcap C_2$	✓	(1)	
	Nominal (only 1)	$\{a\}$	✓		
	Object Exist. Quantif.	$\exists R.C$	✓		
	Instance Exist. Quantif.	$\exists R.\{a\}$	✓		
	Self Restriction	$\exists S.Self$	✓		
Class Axioms	Subsumption	$C_1 \sqsubseteq C_2$	✓	✓	✓
	Equivalence	$C_1 \equiv C_2$	✓	✓	✓
	Disjointness	$C_1 \sqcap C_2 \sqsubseteq \bot$	✓	✓	✓
Property Axioms	Role Inclusion	$R_1 \sqsubseteq R_2$	✓	✓	✓
	Role Equivalence	$R_1 \equiv R_2$	✓	✓	✓
	Role Disjointness	$disjoint(R_1, R_2)$	×	✓	✓
	Role Inverse	$R_1 \equiv R_2^-$	×	✓	✓
	Role Domain	$domain(R, C)$	✓	✓	✓
	Role Range	$range(R, C)$	✓	✓	✓
	Reflexivity	$reflexive(R)$	✓	✓	✓
	Irreflexivity	$irreflexive(R)$	×	×	✓
	Transitivity	$transitive(R)$	✓	×	✓
	Symmetry	$symmetric(R)$	×	✓	✓
	Asymmetry	$asymmetric(R)$	×	✓	✓
	Functional	$functional(R)$	×	×	✓
	InverseFunctional	$inversefunctional(R)$	×	×	✓

Table 4.1: Extracted OWL 2 Profiles Supported Constructs

In Table 4.1, R represents a role, S a simple role, A an Atom (a unary predicate), C is a class (R and C can have subscripts), and a an instance.

"Class Expressions" are the class restrictions that can be used in axioms.
The indicator (1) refers to the fact that for QL and RL, restrictions on class expressions depend on whether constructs are part of superclass vs. subclass axioms.

On the other hand, when indicated, Class Axioms for QL and RL are supported provided they abide by the superclass/subclass restrictive forms on predicates.

RL predicates are more elaborated than the contents of the described syntax, especially when used in order to interpret rules.

4.5.4 Semantics Expressiveness Management Algorithms

The semantic expressiveness management is based on the above section's modularization and projection techniques.

Algorithms of which extracts are elucidated here were devised based on the contents of the above section. They stress on the resolution aspect of expressive axioms (Algorithm 1) and then on projecting and building lists of deprecated axioms per projected expressiveness (Algorithm 2, itself based on Algorithm 1).

Definition 3 *Deprecated ontology elements are those rendered obsolete due to the projection π of an ontology $\mathcal{O}$ from a Language A (usually OWL DL) to a Language B (a DL fragment). Depreciated elements form the "resolved" versions of extra expressive elements once their expressive features are deprecated. A depreciated element can be equal to $\perp$ when no particular depreciated version exists for the deprecated element.*

Algorithm 1 Interpretation

SMOF structures contain elements that allow the detection of restriction axioms that have DL undecidability side effects, along with their corresponding depreciated elements. These elements can potentially be set per use case: the depreciation level that the algorithm privileges (lines 3 through 5). However, in general they are defined independently, per expressiveness level, i.e. the second depreciation level privileged by the algorithm (lines 7 through 8); or simply according to their type, the third depreciation level applied by the algorithm (in lines 11 through 13).

Finally, the restricted elements for which no depreciation is identified are deprecated

Algorithm 1 *resolve(α,ϵ)*: Resolve Axiom α accordingly with expressiveness ϵ

```
1: Input: the axiom to be resolved $\alpha$, the expressiveness level to abide by $\epsilon$
2: Output: the resolved axiom $\alpha$
3: $\alpha' \leftarrow extract\ UC.depreciated(\alpha)$;
4: if $\alpha'$ is not null then
5:     set $\alpha \leftarrow \alpha'$;
6: else
7:     $\alpha' \leftarrow extract\ smofElement.depreciated(\alpha)$ w.r.t. $\epsilon$;
8:     if $\alpha'$ is not null then
9:         set $\alpha \leftarrow \alpha'$;
10:    else
11:        $\alpha' \leftarrow extract\ smofElement.depreciated(\alpha)$;
12:        if $\alpha'$ is not null then
13:            set $\alpha \leftarrow \alpha'$;
14:        else
15:            $deprecate(\alpha)$; // $\alpha = \perp$
16:        end if
17:    end if
18: end if
19: return $\alpha$;
```

(line 15).

A deprecated axiom is rendered obsolete (set for elimination from the ontology to be projected).

Algorithm 2 Interpretation

This pseudocode represents extracts taken from the complete projection algorithm. These extracts highlight the basic elements behind its logic and conception. The algorithm can be called for particular levels of expressivity, and for specific deprecated lists to be returned for comparison purposes.

Its asymptotic complexity is $\mathcal{O}(n)$, where n is the number of input axioms to be traversed by the algorithm and projected according to their classified expressiveness in the metamodel.

Starting with a definition of empty SMOF ontology containers (line 3), and dictionaries of (key,value) pair lists to hold (deprecated, depreciated) elements (line 4), the pseudocode then considers the set comprising all axioms in the input ontology (line 5).

It visits these axioms one after the other. Lines 6 through 9 treat the DL restriction axioms (by calling the previous algorithm to resolve them appropriately).

Up to line 16 in the algorithm, a pass to determine which of the profiles an axiom

Algorithm 2 Extracts from Projection Algorithm:

```
Input: $O_{DL} \leftarrow$ {The set of all axioms forming the OWL DL Ontology} ;
Output: Projected ontologies $\pi_{DL}$, $\pi_{EL}$, $\pi_{QL}$, $\pi_{RL}$ along with corresponding
            Dictionary of (deprecated, depreciated) elements $DL^-$, $EL^-$, $QL^-$, $RL^-$
    ;
$\pi_{DL}, \pi_{EL}, \pi_{QL}, \pi_{RL} \leftarrow \{\}$ ;
$DL^-, EL^-, QL^-, RL^- \leftarrow$ empty (key,value) pair lists ;
for all $axiom\ \alpha \in O_{DL}$ do
    if $\alpha.type \in (RestrictionAxioms_{DL})$ then
        $(O_{DL}.swap(\alpha, resolve(\alpha, DL))$;
        $DL^-.add(\alpha, resolve(\alpha, DL))$ ;
    end if
    if $\alpha.type \in (Semantics_{EL}$,$Semantics_{QL}$,$Semantics_{RL})$ then
        $(\pi_{EL}.add(\alpha), \pi_{QL}.add(\alpha), \pi_{RL}.add(\alpha))$ respectively;
    else
        $(\pi_{EL}.swap(\alpha, resolve(\alpha, EL))$,
         $\pi_{QL}.swap(\alpha, resolve(\alpha, QL))$,
         $\pi_{RL}.swap(\alpha, (resolve(\alpha, RL)))$ respectively;
        $(EL^-.add(\alpha, resolve(\alpha, EL))$,
         $QL^-.add(\alpha, resolve(\alpha, QL))$,
         $RL^-.add(\alpha, resolve(\alpha, RL)))$ respectively;
    end if
end for
for all $\alpha_{rl} \in \pi_{RL}$ do
    if $\alpha_{rl}.type \in Direction\_Conditional\_Constructs$ then
        $construct_{lhs} \leftarrow lhs(\alpha_{rl})$
        $construct_{rhs} \leftarrow rhs(\alpha_{rl})$
        if $(construct_{lhs} \in disallowed_{lhs})$ or $(construct_{rhs} \in disallowed_{rhs})$ then
            $\pi_{RL}.add(resolve(\alpha_{rl}, RL))$ ;
            $RL^-.add(\alpha_{rl}, resolve(\alpha_{rl}, RL))$ ;
        end if
    end if
end for
for all $\alpha_{ql} \in \pi_{QL}$ do
    if $\alpha_{ql}.type \in QL\_Conditional\_Semantics$ then
        if $\exists clash\_condition \in \pi_{QL}$ then
            $\pi_{QL}.resolve(\alpha_{ql})$
            $QL^-.add(\alpha_{ql}, resolve(\alpha_{ql}, QL))$ ;
        end if
    end if
end for
$\pi_{DL} \leftarrow O_{DL}$
return $\pi_{DL}$, $\pi_{EL}$, $\pi_{QL}$, $\pi_{RL}$ and $EL^-$, $QL^-$, $RL^-$ correspondingly;
```

belongs to is made (note that very often one axiom belongs to more than one fragment); each axiom is accordingly treated, with projected ontologies and deprecation lists being built progressively.

Every profile's syntactic fragment metadata helps determining a first subset of DL axioms, which still need some more specific filtering to be applied.

From line 17 to line 26, some RL specific processing takes place. The preliminary set of RL axioms is reconsidered to determine whether its constituting antecedents and consequents happen to belong to a further restrictive group. In such a restrictive group, constructs are differently interpreted based on their belonging to the left or to the right hand side of the expression: such a detected condition causes the complete axiom to be excluded from the profile.

It should be pointed out that this reasoning is similarly repeated for the QL syntactic fragment: this part is not shown in the extracts, for the sake of simplicity and ease of presentation.

Lines 27 through 34 depict the analysis of the henceforth isolated set of QL axioms, by further checking whether their constituents can be affected if combined with other particular constructs, and whether these specific constructs happen to be present in the isolated set of axioms being investigated. Such collected axioms, if the test turns out positive, are resolved to get depreciated or excluded from the profile.

Again, it is noteworthy that this reasoning is similarly repeated for the EL and QL syntactic fragments: again, this part is not shown in the extracts, for the sake of simplicity and ease of representation.

The final statements are straightforward and refer to the returned resulting projected ontologies and lists of deprecated/depreciated elements.

Projected KB Example

The following example lists a subset of axioms in a KB (in which the DL notation is used). An illustration of how they will be processed by the projection algorithm will follow it.

$\text{Orphan} \equiv \text{Person} \sqcap \forall\text{hasParent.Dead}$
Transitive(hasSibling)
Asymmetric(hasSibling)

This KB contains an class equivalence axiom that includes a universal restriction at the right hand side, and an object property that is at the same time transitive and asymmetric.

The following tables show the resulting output of an EL, QL and RL projections of the above KB, along with the deprecated and depreciated corresponding elements:

Projected EL Ontology π_{EL}	**Deprecated** → **Depreciated List** EL^-
$\text{Orphan} \sqsubseteq \text{Person}$	$\text{Orphan} \equiv \text{Person} \sqcap \forall\text{hasParent.Dead}$ $\rightarrow \text{Orphan} \sqsubseteq \text{Person}$
Transitive(hasSibling)	Asymmetric(hasSibling) → hasSibling

Universal restrictions in EL are depreciated to subsumption axioms based on the arranged metamodel contents; the asymmetric property characteristic is not supported as well; it is thus deprecated (the object property is left without this characteristic).

Projected QL Ontology π_{QL}	**Deprecated** → **Depreciated List** QL^-
$\text{Orphan} \equiv \text{Person} \sqcap \forall\text{hasParent.Dead}$	
Asymmetric(hasSibling)	Transitive(hasSibling) → hasSibling

Universal restrictions are supported in QL right hand side axiom constructs. The transitive property characteristic is not supported and is thus deprecated (the object property is left without this characteristic).

Projected RL Ontology π_{RL}	**Deprecated → Depreciated List** RL^{-}
Orphan ≡ Person ⊓ ∀hasParent.Dead Asymmetric(hasSibling) Transitive(hasSibling)	$\perp$

This KB is fully supported in RL, therefore its projection leaves it intact, and returns an empty list of deprecated/depreciated elements.

4.6 Discussions

The decision whether the projection is signature or fragment-based or both depends mostly on the decision-support scenario in question. A matrix of recommended expressiveness projection and reasoners is arranged in the knowledge base repository. This matrix takes into consideration most of the impacting factors including:

- The signature of the use case.
- The size and expressiveness of the generated module.
- The type of the use case competency question (intensional, extensional, and their more detailed sublevels).
- The computed depreciation per fragment.

In Appendix B, the structure “SYS_DECISION_MATRIX” shows more details related to this matrix.

That said, experimentation revealed that a considerable number of the domain's straightforward use cases were satisfied by the DL performance; Chapter 6's Section 6.7 comes later in support of this claim.

For more elaborated scenarios, the use of OWL flavors was appropriate, based on each profile's target application. As intricately related classes and properties are subject to OWL 2 EL proficient reasoning, the large number of instance data takes advantage of OWL 2 QL efficient querying facilities. On the other hand, when deemed advantageous, OWL 2 RL projections make rule-based implementations a Web3.OWL possibility.

As such, the approach that incorporates modularization and expressiveness-based projection techniques adopted for this framework does not guarantee completeness. It guarantees efficiency through reasoning results in Polynomial time (instead of Exponential time DL complexity), and feasibility of a variety of semantically rich scenarios through solutions to interesting entailment problems. On another note, for the domain in question, and in almost all the extensional use cases, completeness is not critical.

With the continuous evolution of the DL and reasoning efforts and their considerable tangible impacts, it was essential for any Web 3.0 devised conceptual framework to be able to capitalize on these evolutive achievements.

Web3.OWL, given its extended metamodel standard, offers a smooth process to incorporate and adapt to support structural linguistic advances, such as further sublanguages, tractable fragments, rules, etc., all at the heart of the DL research community.

Recent developments show that not only established reasoners' underlined methods are continuously being pushed further, but new reasoners are regularly entering the field as well. Reasoning capabilities and performance time have been able to improve from exponential to polynomial complexity [77, 23, 90].

Waiting for similar advances in querying capabilities to become well-established and reliable for adoption and inclusion, Web3.OWL builds on the existing efforts and remains ready to proactively absorb such advancements.

4.7 Conclusion

The approach to organize and handle expressiveness described in this chapter is very essential for the framework, it allows indeed for overcoming computational reasoning and querying complexity for scenarios that are restrained by that complexity.

In the next chapter, an analytic view of the framework's semantic tagging feature is unveiled, revealing a different approach and utilities for addressing extra expressiveness.

Chapter 5

Semantic Tagging Approach

5.1 Introduction

Web 3.0 is an evolution of Web 2.0 that does not eliminate the latter's most characterizing aspect: the active role of the social network user. Formerly, before Web 2.0, the user was a simple passive reader. Upon receiving the information, there were no possibilities granted to the SN user allowing him to collaboratively get involved in the contents of the information. With Web 2.0, the user's role switched to that of a contributor. Different tagging facilities opened the door for him to interfere through highlighting, evaluating and describing the information.

Tagging activities have proved their significance in Web 2.0 and Web 3.0 endeavors. In the context of the present framework, semantic tagging refers to the annotation of SN resources (blogs, forums questions and answers) with ontological knowledge existing in the knowledge base ontology. The purpose of the endeavor is to classify these resources according to logical knowledge, and/or populating the knowledge base ontology with information on SN users (writers and readers of these resources). Semantic tagging is an automatic or a semi-automatic activity.

Web3.OWL hence attempts to magnify the role of tagging by acknowledging previous efforts and building upon them. Following an approach that goes along with its foundation concepts and established infrastructure, the framework benefits from its extra expressiveness and from advanced non-standard reasoning techniques to generate effective tag suggestions beneficial for different decision support scenarios.

This chapter, partly based on work already presented in [27], is dedicated to addressing the adopted semantic tagging approach. After introducing some state of the art relevant efforts in Section 5.2, it describes the complete approach in Section 5.3. Subsequently, Section 5.4 particularly underlines the semantic tagging suggestion process and provides example scenarios to highlight it before wrapping up with the conclusion in Section 5.5.

5.2 Semantic Tagging - Summary from Relevant State of the Art Efforts

Along with the continuous expansion of Web 2.0, tagging activities and their related research efforts are very active. Several projects and endeavors demonstrate their usefulness at different levels such as text classification, recommender systems and web search.

Recent studies presented in [67] and [89] provide reviews on social tagging systems and tag-aware recommenders, and provide methods (network, tensor and topic-based) and metrics for tag evaluation. Furthermore, based on an overview of limitations and possible extensions to the current generation of social tagging systems, the usage of folksonomies (taxonomies generated by the "folks", i.e. the SN users) is considered among the areas for future research and advancement of the state of the art.

On the other hand, in an effort deemed as the "closest" to the sociomedical domain, the issue of usefulness and feasibility of tagging in information websites for cancer patients and their relatives is explored in [1].

The study's approach was to analyze literature available on such patients' behavior, existing tags and tagging activities in several blog postings and cancer websites; terminologies and tags were then used for the development of a social tagging prototype in (cancer.dk). This prototype's usability was also evaluated in the same study.
The research's findings demonstrated the potential of tags for describing and facilitating information access and assessment, and also as a means to reconcile the differences between scientific viewpoints and terminology on the one hand, and everyday vocabulary on the other; they were positive as well with regards to the users' reaction towards tagging activities and tagging practicality. The whole effort in this study was completely free of any ontological or semantic web support.

5.3 Semantic Tagging Approach

Just as the Web3.OWL canonical ontology design follows the methodology of maximizing expressiveness and knowledge for optimal future uses and applications, the framework's tagging approach adopts the same principles and for the same practical purposes.

The rationale behind the semantic tagging approach is to associate tags to SMOF metamodel elements representing OWL ontology axioms (logical expressions).

Among the variety of accessible Web 2.0 sites, "Mommy blogs" are those in which Social Network users provide tons of information related to children, their problems and behaviors and to parenting in general; as for "Mommy bloggers", usually parents, they are perceived as extremely active and cooperative users who constantly access and manage their blogs.

A progressive role granted to the SN User allows him to get exposed to generated semantic suggestions. Consequently, the user is expected to explicitly authenticate and possibly extend them by communicating meaningful expressive corrections and validations.
This course of action is enthused by the different available SN tagging systems - for

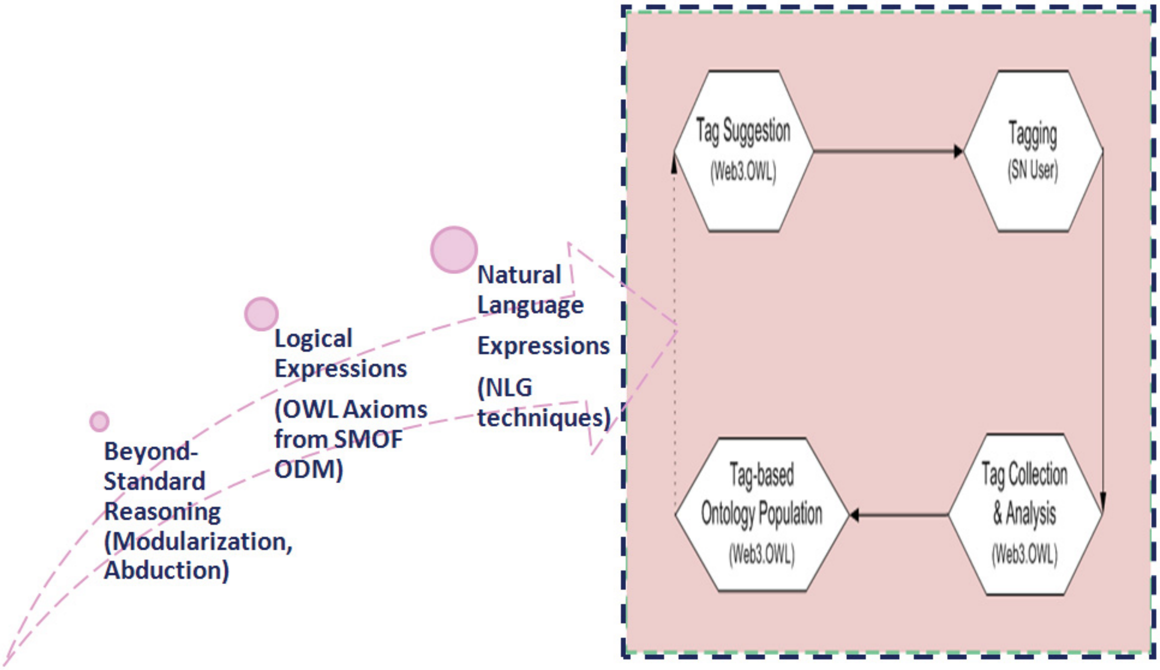


Figure 5-1: Semantic Tagging Process in Web3.OWL

instance Flickr[1] and Del.icio.us[2] - that make it possible for users to tag their photos, documents and webpages with simple descriptive taxonomies.

Thus, Web3.OWL confers a SN user collaboration novelty residing in promoting "semantic tagging assignments" that are initiated on account of domain-specific semantic arrangements in the knowledge base repository. These tagging assignments are offered to taggers through a natural language generation (NLG) technique that hides all constructs and semantics formalisms.

In Figure 5-1, a high-level representation of the semantic tagging process is depicted. The main purpose behind that is to allow the reader to differentiate between the tagging activities reflected in the framework.

---

[1] www.flickr.com

[2] www.delicious.com

Starting with the generation of tag suggestions, which principles and methodology are at the heart of the contributed approach, the actual tagging of blogs and SN users information takes place. Subsequently, tagging collection takes place. Once collected, tags are subject to a preliminary selective analysis prior to their population in the ontology.

Once populated with individuals (FOAF members for instance) pertaining to particular classes and axioms, the ontology is subject to reasoning algorithms. The applied reasoning techniques have the potential of inferring new knowledge, which in certain cases can lead to further recommended tags. Thus, a repetition of the semantic tagging process can occur.
Examples illustrating this scenario (in which a process repetition is induced) will be provided in Section 5.4 below.

Figure 5-2 presents the Knowledge Base Repository extensions specific for the tagging approach. Appendix B shows these structures' details.

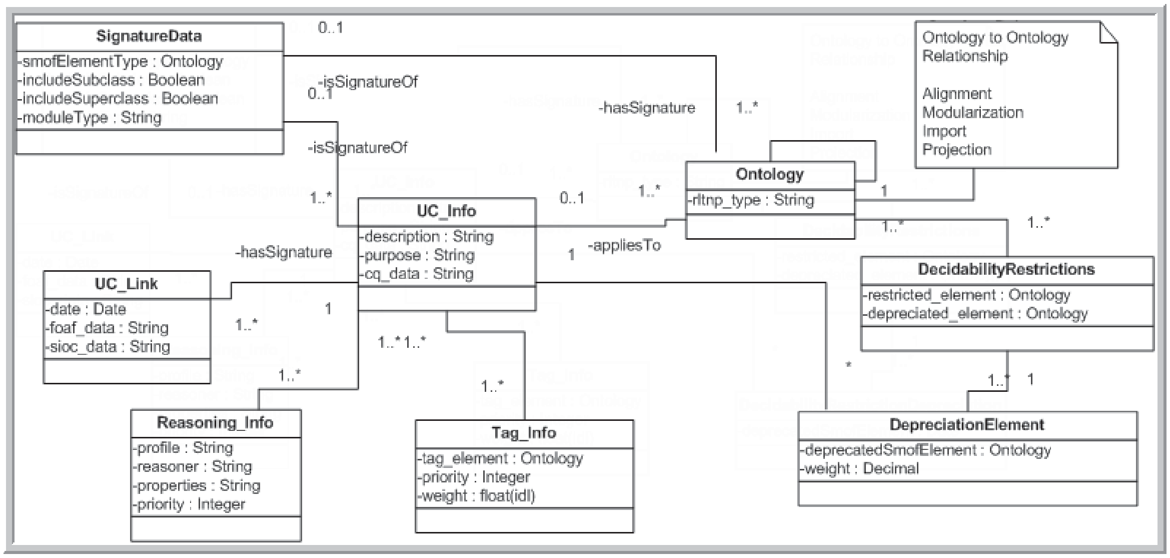


Figure 5-2: SMOF/ODM-Based Extensions for Tagging and Use Case Handling

The meta-semantics structures contain information related to the selection of the tags to push and process. According to the use case under analysis, different tags having distinct priorities are considered. They can be identified through their corresponding ontology axiom, from which they inherit all relevant properties.

For example, the axioms and expressions in Table 5.1 represent possible tags suggested for validation by the SN user.

DL Axioms (SMOF elements)	**Corresponding NLG Expressions**
John: Child $\sqcap$ hasAge(1 integer) $\sqcap$ $\exists$suffersFrom.Obesity	John is a 1 year-old child who suffers from obesity
$\exists$isParent(BlogAuthor, John)	The blog's author is John's parent

Table 5.1: Examples of SMOF logical elements and NLG expressions

5.3.1 Semantic Tagging Modes and Motivations

The aspects that characterize Web3.OWL while particularly promoting tagging suggestion facilities are the following:

- The profile of SN users who deal with the framework, i.e. the “candid” mommy bloggers. These SN actors are known by their strong willingness to contribute to the social parenting media. Given their honesty and enthusiasm regarding all parenting aspects, they usually are eager to collaborate and assist in making parenting media reach advanced levels and attain objectives that endorse knowledge and awareness.
- The nature of parenting information, its controversial facets for which surveys and tagging analysis are deemed very beneficial.
- The level of expressiveness of the metamodel that allows meaningful modeling of motivating opinion-mining cases (as an example).

- The fact that the framework is ontology-driven, and the availability of non-standard reasoning techniques that offer interesting potentials for tags suggestion.

On the other hand, the Web3.OWL tagging activities serve at least one of the following ends:

- Populating the ontology with SN users according to their profile and criteria (sex, age, parent, child, presents a certain symptom or behavior, etc.).
- Validating the competency question meaning and purpose against the ParOnt applicable signature, a step having a strong impact on identifying the most suitable OWL 2 profile, when needed.
- Populating opinion-mining ParOnt derivatives (extracted modules) within the scope of particular sentiment analysis, segmentation and profiling survey scenarios.
- Validating or canceling tag axioms that are pushed in order to "close the world" within a certain context. Such axioms are predefined with the purpose of overcoming the scenarios that can be blocked by the Open World Assumption (OWA), thus rendering querying with accurately returned results a possibility (instead of obtaining responses with incomplete results, or with no results at all).

That said, the framework supports different tagging modes, depending on the nature of the use case, as follows:

- On-going suggestion and collection of tags, for instance on a periodic basis. In general, these tags are generated based on ontology-aware NLP grammars that take into consideration the global signature of ParOnt, generating tags based on Named Entity Recognition.
- Use case-oriented suggestion and collection of tags. For instance, for those comprising user queries that once analyzed generate tags on the fly (query disambiguation and validation, collection of additional relevant information).

Finally, and in accordance with the above, SN users can perform their tagging activities independently from their undertaken SN occupation, in other words:

- As readers who access certain parenting blogs, mommy bloggers are offered the facility of contributing tags related to their own profile, to the contents of the blog they just accessed, or to certain parenting use cases underway.
- As writers, upon saving their work (post or question submission), mommy bloggers are given the chance to contribute tags related to their submitted post.

5.3.2 Tagging Approach Fundamental Principle

The primary formal definition of a tag was first provided in [63]. The definition was based on a tripartite model relying on an actor (a user), a concept (a tag or keyword), and an instance (annotated resource):

$$Tag \subseteq Actor \times Concept \times Instance \tag{5.1}$$

Later on in [74], the tripartite model definition was extended to a quadripartite one, after adding a local semantic meaning to each tag, obtained by a URI:

$$Tagging \subseteq User \times Resource \times Tag \times Meaning \tag{5.2}$$

In this thesis, the above definition is hence further extended, and a more granular element is assigned to the definition, using Description Logics (DL) to denote constructs, axioms and expressions.

The definition can thus be denoted by the following:

$$Tagging \subseteq User \times Resource \times Tag \times DL_{SEMANTICS} \tag{5.3}$$

where $\mathcal{DL}_{\mathcal{SEMANTICS}}$ are OWL 2 Constructs, Axioms and Expressions, in other words DL Building blocks forming OWL 2 fragments and languages, such as $\mathcal{ALC}$

and $\mathcal{SHOIN}(D)$ [6, 39].

Web3.OWL thus promotes a syntactic and more formalized approach, benefiting from its metamodel repository's already introduced meta-semantics structures. These allow the distinguishing between different constructs' formal semantics, for instance: existential restrictions, class conjunctions, disjunctions and negations, cardinality restrictions, ranges and datatypes, nominals, role properties (inverse, transitive, hierarchical, and so on).

Axioms in Table 5.1 represent examples of $\mathcal{DL}_{\mathcal{SEMANTICS}}$ along with their NLG expressions.

The motivation behind attaining this level of granularity is to overcome data mining and NLP limitations by reusing ontology definitions and rules tagging, enforced by a possible user cooperation, thus cutting down complex algorithms and compensating for them through reasoning.

Being aware of the fact that non-ontology experts will surely face difficulties trying to read and understand ontology formalized elements, natural language definitions of classes and axioms are made available based on ontology and metamodel prepared annotations on the one hand, and on NLG-based techniques (such as OntoVerbal[1] [29]) on the other. The SN user is thus faced with verbalized naturalistic versions of the formal semantics.

It is the availability or absence of the SN website's user collaboration that will determine whether the overall strategy towards ontology population is semi or fully automatic.

At this stage, the tag collection process takes place. Collected tags are analyzed

---

[1] http://swatproject.org/demos.asp

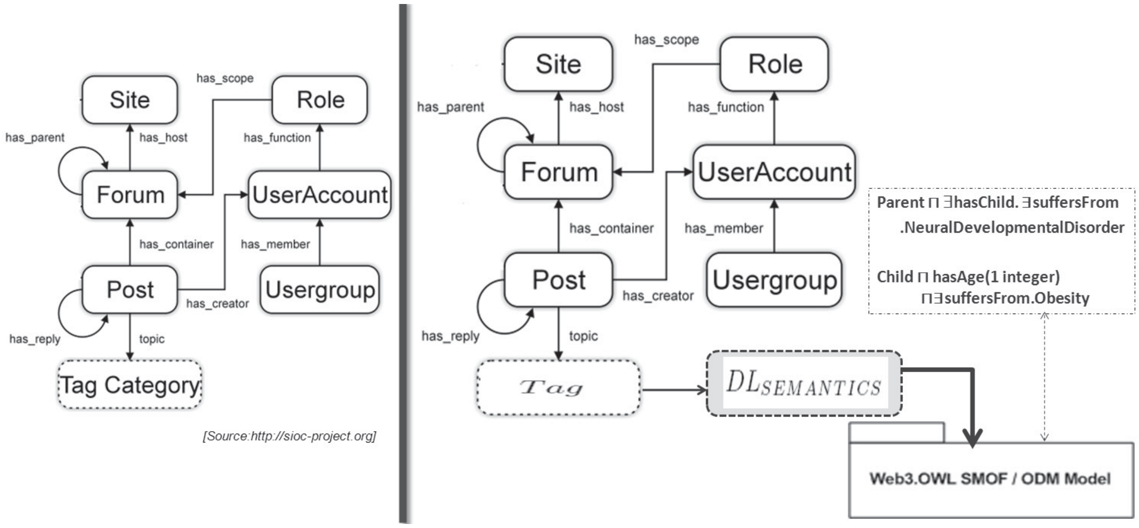
before being populated. Thus, the hereafter populated ontology can be subject to projection procedures. Depending on the application or requirements of interest, the suitable OWL profile can be projected for the applicable reasoning to be performed, leading to the pursued decision-support scenarios outcomes.

From simple information modeling and knowledge representation capabilities, to query and rule execution, the remaining phases of the tagging workflow are iterative and repetitive.

An additional interesting service that particularly relies on the relations and big instance numbers in fragmented ontologies, is the generation of social networks of individuals linked through semantic relations. The ontology can return individuals that are linked through specific semantic relations. Such relations among SN users are more sophisticated than the overused traditional social network links (for example the "knows" relationship). Such SN generation use cases go beyond the overused FOAF relationships and the SIOC lightweight properties. They open the door to SN analysis methods, the results of which can induce further interesting use cases, with different suitably customized configurations. Details of this process however, and as previously indicated, are beyond the scope of the thesis.

To give more insights on the integration of SIOC and related SN ontologies into the framework, Figure 5-3 illustrates the definition of a tag in the SIOC ontology (to the left) and shows how it is extended in Web3.OWL (to the right). A tag's definition can go beyond its URI categorization to get associated to more specific SMOF metamodel elements representing OWL ontology axioms (logical expressions).

As a result, relevant SIOC resources are well identified and grouped according to very specific logical elements.
For example, distinguishing blogs that refer to parents with more than one child



[Source:http://sioc-project.org]

Figure 5-3: SIOC Tag Definition Extended in Web3.OWL

having obesity problems, or blogs written by parents of children suffering from autism, etc., will be a possibility.

5.4 Ontology-Driven Semantic Tagging Suggestion

As it has just been stated and described in the previous section, the generation of ontology-based $\mathcal{DL}_{\mathcal{SEMANTICS}}$ is essential for effective tag suggestion to communicate to the SN users.

The availability and grouping of the maximum domain knowledge in the metamodel repository, together with the fact that Web3.OWL is an ontology-driven framework, boost the tagging suggestion methodologies.
The reinforcement is witnessed particularly through:

- The expressiveness level, and the richness in axioms that can be extracted from the repository. Those typically include, among others, the set of usually deprecated elements, that are not part of traditional reasoning (disjunction)

- The non-standard reasoning techniques that offer interesting tag recommendation potentials, particularly in the case of modularization and abduction tasks.

5.4.1 Non-Standard Reasoning for Tags Suggestion

The requirements for the current tagging approach overlap with cutting-edge research in Description Logic, which is the foundation for OWL 2, namely in what concerns non-standard reasoning services. In Chapter 3 (Section 3.4.2), the standard reasoning techniques were introduced. Thus, whenever algorithms over ontologies go beyond those traditional techniques, the literature point outs the "non-standard" reasoning techniques. For instance, modularization (already elucidated in Chapter 4), conjunctive querying (to be addressed in the next chapter), and abduction (the subject of the current section), are all examples of non-standard reasoning tasks, to name a few only.

Tags Suggestion Based on Abductive Reasoning

Abductive reasoning, in logic, is the task of reasoning from observed effects to possible causes. More formally, as used in [57]:

Definition 4 *Abduction is a form of hypothetical reasoning in which, given a certain observed conclusion, possible input facts that might lead to that conclusion are derived. Its simplest associated inference scheme is the following:*

$$\{\alpha \rightarrow \Gamma; \Gamma\} \rightarrow \alpha$$

In this scheme, $\alpha \rightarrow \Gamma$is background inference knowledge and$\Gamma$is a fact that needs to be explained, for which α is deduced as the conclusion explanation.

Abductive reasoning comes at a price: the inferred conclusion might not turn out to be the right conclusion. It is hence classified as a form of hypothetical or fallible reasoning.

For the current framework's purposes, recently established abductive reasoning techniques can be exploited.

A significant research on DL abductive reasoning [57] proposed sound and complete methods to compute abductive solutions with certain minimality over expressive DL.

Earlier in [30], a case study was presented to demonstrate the usefulness of abductive reasoning in the context of ontologies. In very recent studies, abductive reasoning is put in use to deduce negative query answering explanation services [18], another type of non-standard reasoning; and in [59], abduction serves as an effective means to discover missing particular "is-a" (subsumption) relationships.

However, and to the best of the author's knowledge, abduction has not been exploited for tag recommendation, and certainly not within the scope of the domain in question.

A Case for Tags Suggestion through Abductive Reasoning

In what follows, a case on children's aggressive behavior is presented to demonstrate a possible usage of abductive reasoning within Web3.OWL.
This example however has not been implemented, it remains a conceptual case to illustrate the usability of this type of reasoning within the framework.
Implemented use cases (on other scenarios of tagging activities) will be explored in Chapter 6).

In abductive reasoning, a number of facts collected from a multitude of sources are gathered and analyzed in order to make a certain assessment on the enclosed information. Knowing a given "effect" fact, the quest goes for some most likely hypothesis that would explain certain observations (the "cause" fact) is adopted as the starting point of research.

Consider the following knowledge base (KB), extracted from ParOnt at a certain point in time, subsequently to previous cycles of tags collection and population into the ontology. In other words, in the context of the framework, abductive reasoning builds upon observed phenomena of previous cycles and existing TBox and ABox content of ParOnt.

Representative KB extracts relevant to the illustrated example are:

$$
\begin{array}{lcll}
\hline
\text{Child} & \sqsubseteq & \text{Person} & (1) \\
\text{Preschooler} & \sqsubseteq & \text{Child} & (2) \\
\text{Cartoon} & \sqsubseteq & \text{Show} & (3) \\
\exists \text{watches}.\top & \sqsubseteq & \text{Person} & (4) \\
\top & \sqsubseteq & \forall \text{watches}.\text{Show} & (5) \\
\exists \text{hasContent}.\top & \sqsubseteq & \text{Show} & (6) \\
\top & \sqsubseteq & \forall \text{hasContent}.\text{ViolenceContent} & (7) \\
\exists \text{hasBehavior}.\top & \sqsubseteq & \text{Person} & (8) \\
\top & \sqsubseteq & \forall \text{hasBehavior}.\text{Behavior} & (9) \\
\text{XMen} & : & \text{Cartoon} \sqcap \exists \text{hasContent}.\text{ViolenceContent} & (10) \\
\text{Ryan} & : & \text{Preschooler} \sqcap \exists \text{hasBehavior}.\text{AggressiveBehavior} & (11) \\
\text{Troy} & : & \text{Preschooler} \sqcap \exists \text{watches}.\text{XMen} & (12) \\
\text{Ryan} & : & \exists \text{watches}.(\text{Cartoon} \sqcap \exists \text{hasContent}.\text{Violence}) & (13) \\
\text{AggressiveBehavior} & : & \text{Behavior} & (14) \\
\text{ViolenceContent} & : & \text{Content} & (15) \\
\hline
\end{array}
$$

In this KB, (1) to (9) represent the *TBox* and *RBox* subsumption, domain and range definition axioms. From (10) to (15) is a list of sample *ABox* axioms.

While querying the KB, the phenomenon of aggressive behavior is observed for a considerable number of preschoolers. Furthermore, a subset of these preschoolers reports watching cartoons known to have a certain level of violent content, and are thus classified accordingly in the ontology.

Upon searching for a possible cause behind the witnessed aggressive behavior, abductive reasoning leads to the proposition of the following hypothesis: Could the action of watching cartoons that contain violence be the reason behind preschooler's aggressive emotional behavior?

$$\text{Preschooler} \sqcap \exists\text{watches}.(\text{Cartoon} \sqcap \exists\text{hasContent}.\text{ViolenceContent})$$
$$\sqsubseteq$$
$$\exists\text{hasBehavior}.\text{AggressiveBehavior}$$

So far, the process of coming up with the hypothesis that could best explain the evidence was achieved. As this hypothesis is not necessarily correct since other factors may play a role or cause the conclusion, a good attempt at validating it is to rely on tagging outcomes.

Accordingly, a list of suggested tags deduced from the above-mentioned subsumption axioms is pushed to the SN user (in natural language). This by the way is the stage of the process depicted in Figure 5-1's dashed arrow that goes into the block "Tag Suggestion".

Thus, by confirming or negating the suggested tags, the SN user contributes in the establishment of a profiled population (a partitioning of the population according to specified traits, criteria or behaviors).

$$\exists\text{watches}.(\text{Cartoon} \sqcap \exists\text{hasContent}.\text{ViolenceContent})$$
$$\neg\exists\text{watches}.(\text{Cartoon} \sqcap \exists\text{hasContent}.\text{ViolenceContent})$$
$$\exists\text{hasBehavior}.\text{AggressiveBehavior}$$
$$\neg\exists\text{hasBehavior}.\text{AggressiveBehavior}$$

The above tags, once collected and analyzed, will permit querying the preschooler population to form partitions according to the following activities and behavior, described in the form of conjunctive queries (the notation of rules and conjunctive queries are defined in the next chapter's Section 3.3):

- Watches cartoons containing violence and has an aggressive behavior:

$$\text{Q(x)} = \text{Child(x)} \wedge \text{hasBehavior(x, AgressiveBehavior)} \wedge \text{Cartoon(y)} \wedge \\ \text{hasContent(y, ViolenceContent)} \wedge \text{watches(x, y)}$$

- Does not watch cartoons containing violence and has an aggressive behavior:

$$\text{Q(x)} = \text{Child(x)} \wedge \text{hasBehavior(x, AgressiveBehavior)} \wedge \text{Cartoon(y)} \wedge \\ \text{hasContent(y, ViolenceContent)} \wedge \neg \text{ watches(x, y)}$$

- Watches cartoons containing violence and does not have an aggressive behavior:

$$\text{Q(x)} = \text{Child(x)} \wedge \neg \text{ hasBehavior(x, AgressiveBehavior)} \wedge \text{Cartoon(y)} \wedge \\ \text{hasContent(y, ViolenceContent)} \wedge \text{watches(x, y)}$$

- Does not watch cartoons containing violence and does not have an aggressive behavior:

$$\text{Q(x)} = \text{Child(x)} \wedge \neg \text{ hasBehavior(x, AgressiveBehavior)} \wedge \text{Cartoon(y)} \wedge \\ \text{hasContent(y, ViolenceContent)} \wedge \neg \text{ watches(x, y)}$$

Given the explicit nature of the tags (the axiom and its negation), the Open World Assumption effects will be overcome, and named individuals with specific characteristics will be available in the KB.

Based on the total number of respondents, and on the percentage of responses according to each of the above conjunctive queries, the findings will establish or refute the existence of a correlation between the suggested cause and the hypothesized effect.

A Case for Modularization and Disjunction

As an interesting application of modularization for a purpose that is different from the ones described in the previous chapters, this section describes the use of modules for tags generation and recommendation.

Consider the following (again conceptual) scenario:

As part of some use case on the analysis and assessment of controversial aspects, there is a particular interest in studying the perception of slapping children among different societies.

For such a use case, it is typical for only "Slapping" specific ParOnt segments to be considered for the opinion mining services and sentiment analysis applications in question. Thus, it is essential to use the previously mentioned modularization mechanism in order to extract the required module.

Once extracted, there are big chances the module would contain extra expressiveness features; there is no need to apply projection techniques as is the case when standard reasoning services are invoked. Expressive axioms are rather exploited during the tag recommendation and collection process.

Thus, the following KB illustrates sample expressive axioms relevant for the considered use case:

$$\begin{array}{lcl}
\hline
\mathrm{SlappingControversy} & \equiv & \mathrm{SlapTolerant} \sqcup \mathrm{SlapForbidding} \\
\mathrm{SlapTolerant} & \sqcap & \mathrm{SlapForbidding} \sqsubseteq \bot \\
\hline
\end{array}$$

Beyond RDF formal semantics (OWL 2 syntactic sugar feature), disjoint union axioms prove useful for the detection and analysis of the population's attitude towards slapping children.

Accordingly, users are split based on their beliefs and experiences. For instance:

FoafId1	:	SlapTolerant
FoafId2	:	SlapForbidding

Moreover, the module contains useful opinion-mining axioms as to the attitude of users towards the possible effects of slapping children, for example whether they associate it to developmental disorders.

Collaborative tags collected over time, and analyzed via queries, will demonstrate the evolution of opinions, the changes in perspectives between a location and another, the differences between mentalities according to other related aspects.
The role and weight of tagging across different use cases will be further highlighted in Chapter 6.

5.5 Conclusions

There are clearly great potentials for tagging when based on well-defined expressive axioms, instead of when only relying on Wikipedia categories and on the simple least expressive taxonomy aspect of the ontology.

To deal with notorious domain aspects, the best way is to achieve a consensus through conquering large-scale analysis methods. Those methods identify controversies and try to investigate on their causes, factors and characteristics.

In the context of the parenting sociomedical domain, by simply presenting examples illustrating the advantages of collaborative tagging in an expressive semantic environment, SN users will be willing to cooperate in the underlined activities, and tagging will be perceived as a less cumbersome task. After all, every parent is always looking forward for accurate information, alerts, good practices and reliable recommendations.

That said, the framework, being ontology-driven, confers facilities and conveniences for the described ontology-based semantic tagging approach. Collaborative tagging based on methods supported by formal semantics is an important aspect of Web3.OWL. It promotes the level of tagging activities and outcomes, intensifying the role of tagging in advanced decision making services.

The next chapter will reveal more on the experimental results.

Chapter 6

Experiments and Proof of Concept Application Scenarios

6.1 Introduction

Previous chapters, particularly the preceding one, had already introduced conceptual use cases. This one is dedicated for the most critical so far carried out experimental scenarios, use cases and proofs of concept implementations.
It presents the major efforts conducted to prove the value and effectiveness of the framework.

The rest of the chapter is organized as follows: Section 6.2 outlines the implemented components of the framework; Section 6.3 presents the KB structures and the metamodeling implementations, and Section 6.4 provides an overview of the exploited data sources; Section 6.5 depicts the ParOnt ontology and its conceptualization. In Section 6.6, the data analysis and semantic tagging experiments are described. Section 6.7 describes the competency questions portfolio: experimental scenarios with some useful examples based on particular workflow arrangements, singled out from the set of possible use cases described in Chapter 3.
Section 6.8 concludes the chapter.

6.2 Outline of Implemented Components

Acknowledging the fact that the framework is difficult to be fully implemented in a way to make experiments on the variety of possible configurations, this chapter describes typical scenarios and use cases for proof of concept ends.
The thesis does not have the objective of carrying out a performance evaluation study, therefore it was irrelevant to state details and specifications related to the environment and hardware put in use.

To prove the effectiveness of Web3.OWL, the major accomplished implementation efforts consisted in:

1. The design and creation of the metamodel structures in the KB repository, based on the SMOF and ODM standards, and on the proposed extensions for interoperability and language projection.

2. The analysis and modeling of the domain ontology, first through expressive extensions to COPE, and later on through the addition of sub-domains extensions and modules to form ParOnt.

3. The load of data based on XSL (eXtensible Stylesheet Language) transformations from the RDF/XML ontology syntax to the KB data structures.

4. The conducted projection and modularization experiments, and the test over a variety of reasoners.
   This included a comparative study of the performance achieved per DL fragment.

5. The analysis of textual data via NLP through GATE, and the exploitation of parts of the processed data for conducted semantic tagging experiments.

6. The UC design and semantic querying experiments based on the estimated CQ(s).

In terms of decision-support and recommender system queries execution, the main path undertaken was to build a portfolio of competency questions, derived from a set of use cases, grouped hierarchically per parenting subdomain. Many of these competency questions were adapted from or directly built to match questions sent out by mommy bloggers; upon executing them against the ontology, the aim was to compare the obtained results with the ones received from other bloggers.

Other competency questions were conceived without having blogs that correspond to them, particularly to denote the advantage of the framework with respect to the SN sites that are incapable of providing adequate responses to such questions. Experimental cases on semantic tagging were carried out as well.
A subset of the undertaken use cases summarizes the assessed most suitable OWL 2 fragment based on the competency question and the type of information it seeks, highlighting the particularity of the approach in taking advantage of OWL 2 characteristics and findings.

In the below sections, more details will be provided on these listed endeavors.

6.3 Knowledge Base and Metamodeling Implementations

A reversely engineered diagram of the SMOF physical model (generated using the Sybase Power Designer modeling and architecture tool) is provided in Figure 6-1.
It shows the definition structure (SMOF_ELT_DEF) containing all elements along with their type, their fragment-independent depreciated version, and their natural language expression. This structure is linked to a set of other structures that are used to hold the following metadata:

- Element-to-element relationship (SMOF_ELT_RLTNP), with different possible relationship types (containment, alignment, import, etc.)

- Logical levels and sublevels association to the different elements (one element associated to many logical levels/sublevels)
- Data definition code-values (used all over the KB model)

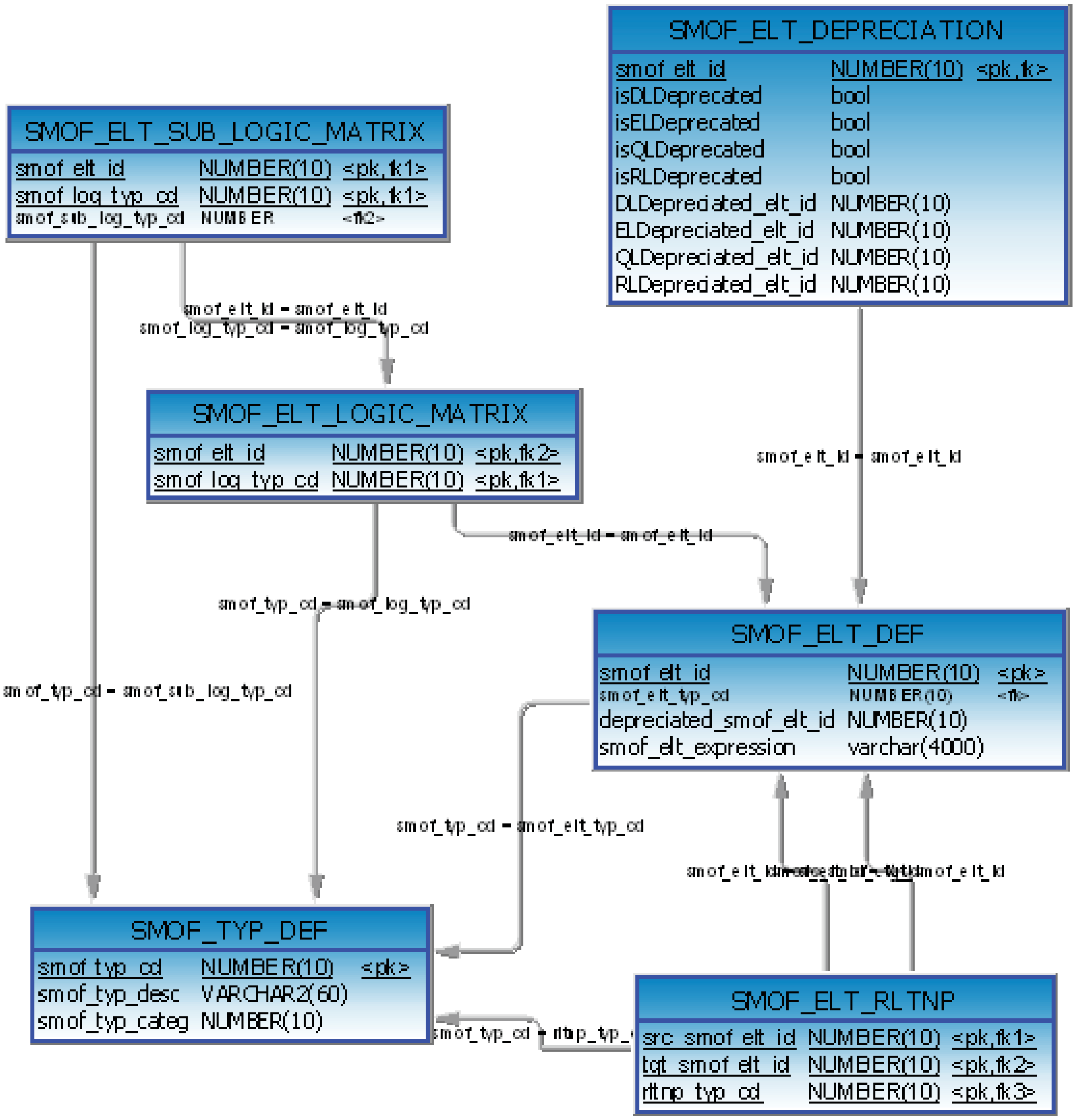


Figure 6-1: Main SMOF Structures Reversely Engineered

The data dictionary for the different structures is provided in Appendix B.

Three main schemas are defined, including the SMOF schema.

In summary, they represent:

- The SMOF schema holding a centralized definition of all elements, their interrelationships and associations to logical levels/sub-levels. Corresponding depreciated versions of elements are also defined.
- The ODM schema to hold the standard metamodel structures, and link them to the SMOF extended schema structures.
- The SYS schema in which system structures are created. Linkage between UCs, reasoning and SMOF elements (for signatures definition for instance) is also managed at this level.

To populate the metamodel with the needed ontological data, the created ontology's RDF/XML syntax file is processed through XSL.
This syntax is used since it can be easily interpreted via XSL transformations.
Scripts are automatically generated based on the XML node defining the ontology element's nature.

As an example, the "owl:Class" node will generate an insertion script in the SMOF definition data structure (with a unique SMOF definition ID), followed by a corresponding ODM insertion in the schema's Class data structure (linked to its parent SMOF structure via the unique identifier). The "owl:AllDisjointClasses" similarly generates an insertion of an SMOF element, and then inserts the element into the ODM Class Disjointness structure (with the link to the existing classes already created along with their SMOF ID/elements).
The matrices for expressiveness are filled according to the disjointness class expression support per profile.

The projection algorithm extracts the ontology elements from these structures and visits their definitions, associations, determining their depreciated versions, to end up projecting a sub-ontology according to the required profile and outputting the list of deprecated and depreciated elements.

6.4 Data Sources Description

Two types of data sources are dealt with for the requirements of the framework:

- Ontology conception data sources, consisting of domain reliable sources[1].

  ParOnt's data sources (mainly relevant to *TBox*, *RBox* and *ABox* ontological data) are described in Table 6.1. These sources required various degrees of analysis and transformations before being incorporated into the ontology, this effort however was carried out by the aforementioned scientists[2].

  Scientific social and medical references were also brought into play, in addition to renowned ontologies (FOAF, SNOMED) to which modularization was applied to be able to reuse a few of their well-established modules.

- Ontology population data sources, consisting of SN sites (blogs and forums) textual data.

  Our efforts being carried out under the scope of parenting awareness and orientation, useful data sources typically beneficial for our domain reside in "Mom Blogger websites".

  While these sites are extremely active and abundant, most of our data is extracted based on Babycenter[3] (which alone counts more than 20 million users), Circle of Moms[4] which connects over 6 million moms worldwide, Canada Moms Blog[5], Raising Children Network[6], among others.

  They yield ParOnt's assertional *ABox* data generated from the ontology population workflow. Particularly, it is important to note that these data sources do not play a role in ontology learning. In other words, there is no automatic extraction of ontology concepts, relations and constructs from this data.

---

[1] Most of these sources were made available thanks to the support of the MCHI (McGill Clinical and Health Informatics)

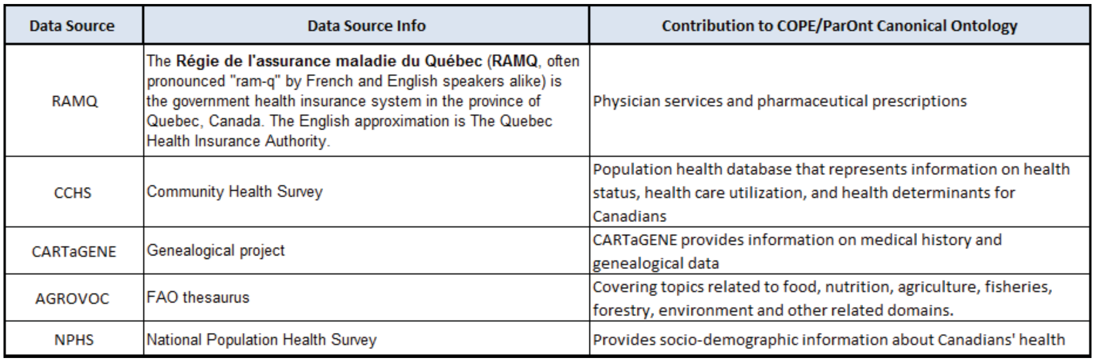
[2] MCHI scientists and researchers

[3] www.babycenter.com

[4] www.circleofmoms.com

[5] www.canadamomsblog.com

[6] www.raisingchildren.net.au



Data Source	Data Source Info	Contribution to COPE/ParOnt Canonical Ontology
RAMQ	The **Régie de l'assurance maladie du Québec** (**RAMQ**, often pronounced "ram-q" by French and English speakers alike) is the government health insurance system in the province of Quebec, Canada. The English approximation is The Quebec Health Insurance Authority.	Physician services and pharmaceutical prescriptions
CCHS	Community Health Survey	Population health database that represents information on health status, health care utilization, and health determinants for Canadians
CARTaGENE	Genealogical project	CARTaGENE provides information on medical history and genealogical data
AGROVOC	FAO thesaurus	Covering topics related to food, nutrition, agriculture, fisheries, forestry, environment and other related domains.
NPHS	National Population Health Survey	Provides socio-demographic information about Canadians' health

Table 6.1: Summary of Ontology Conception Data Sources

6.5 The Knowledge Base Canonical Ontology

ParOnt was conceived with the aim of allowing the analysis of the parenting domains and sub-domains, and consequently generating both generic and customized preventive recommendations.

The ontology was extended and enriched with OWL 2 constructs (already listed in Chapter 3's Table 3.1) that maximize its richness in terms of represented knowledge, and make possible the execution of scenarios that validate the expressiveness management approach.

While Figure 6-2 highlights its major subdomains, the rest of the chapter illustrates exemplified sets of axioms in different KB subsets per explored UC scenario.

Figure 6-3 depicts on the other hand a partial visualization of the ontology concepts and relationships. It highlights ontological interactions mostly modeled within the scope of the COPE-related efforts in [28].
These interactions are between concepts from different sub-domains that have their consequences on the sociomedical one.
They point out how the domain is strongly influenced by the multiple and interact-

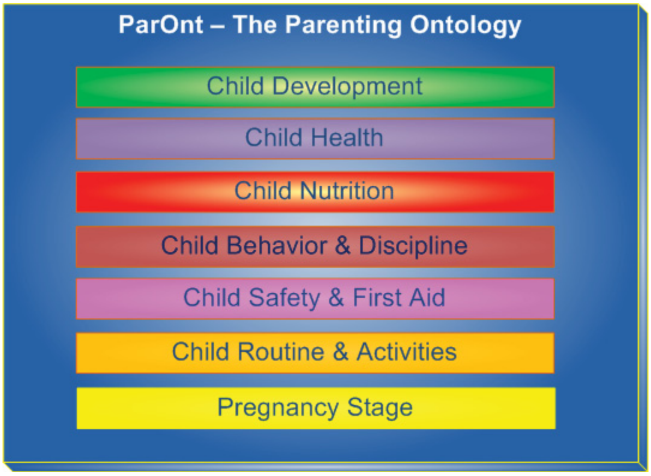


Figure 6-2: Main ParOnt Subdomains

ing bonds through which biological (i.e., gene, brain and physiology), and societal systems (e.g. education, health, agriculture, agri-business, media, and finance) collectively operate on a diversity of spatial and temporal scales.

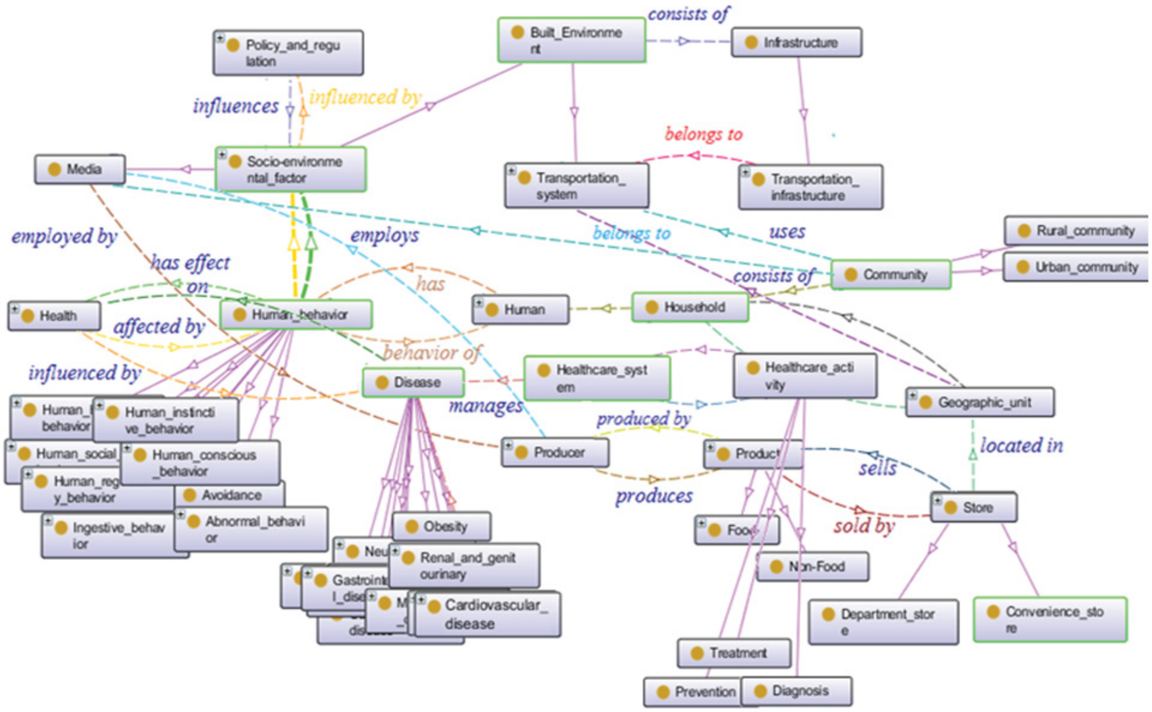
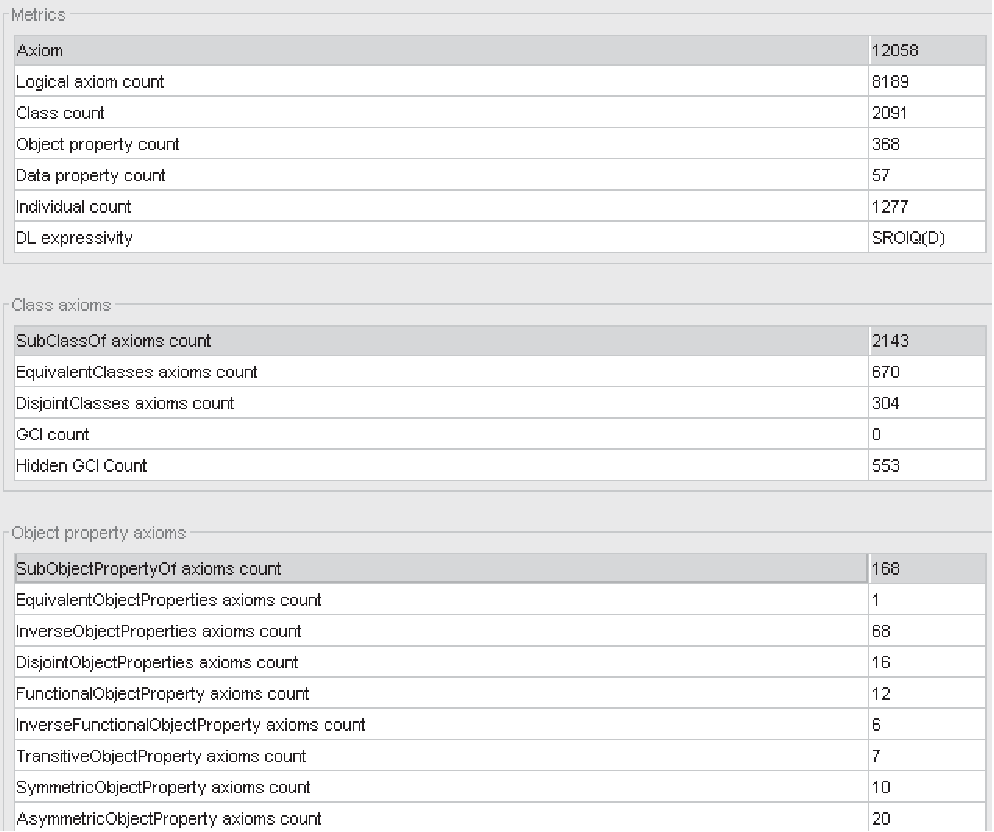


Figure 6-3: Concepts and Interactions Partial View

Finally, in Figure 6-4, the ontology's current metrics are presented.

While being regularly subject to incremental amendments and updates, at the time of writing, ParOnt consists of more than 2000 classes (around 70 of which are immediately below the Top concept), 400 properties (more that 350 object properties and 50 data properties), and 12000 axioms.



Metrics

Axiom	12058
Logical axiom count	8189
Class count	2091
Object property count	368
Data property count	57
Individual count	1277
DL expressivity	SROIQ(D)

Class axioms

SubClassOf axioms count	2143
EquivalentClasses axioms count	670
DisjointClasses axioms count	304
GCI count	0
Hidden GCI Count	553

Object property axioms

SubObjectPropertyOf axioms count	168
EquivalentObjectProperties axioms count	1
InverseObjectProperties axioms count	68
DisjointObjectProperties axioms count	16
FunctionalObjectProperty axioms count	12
InverseFunctionalObjectProperty axioms count	6
TransitiveObjectProperty axioms count	7
SymmetricObjectProperty axioms count	10
AsymmetricObjectProperty axioms count	20

Figure 6-4: Concepts and Interactions Partial View

Both Figures 6-3 and 6-4 have been generated via the Protégé Ontology Editor [50].

Projected ontology modules are also available in the KB, with a much higher number of populated individuals based on the different experimented scenarios. The collected individuals have not been populated to the main ontology yet due to other priorities. This remains part of some future work (that will most probably comprise the application of SNA methods on large numbers of individuals).

6.6 Experimental Senarios Setup

6.6.1 Data Analysis Experiments

"Mom blog" sites listed in Section 6.4 were used to build a data set consisting of more than 500 K blogs and replies, analyzed according to the described data analysis workflow, with the purpose of evaluating the effectiveness of the semantic framework. ParOnt axioms, with stress on particular subsets to be denoted later on throughout this chapter, served as the ontological source for the semantics-aware NLP grammars and Information Extraction algorithms, executed via GATE. Consecutive GATE Web crawls were applied on the variety of SN sites. The data sets were partitioned into multiple corpora (one per SN site) and analyzed by the GATE Processing Resources. For the sake of these proof of concept implementations, and due to time constraints, no sophisticated JAPE[1] (Java Annotations Pattern Engine) rules were coded. Instead, only straightforward algorithms were applied, with simple Named Entity recognition and ontology annotation based on the ParOnt language resource.

A first data analysis pass on a corpus consisting of "blog titles" exclusively helped to sort out the "hot data" relevant to a particular sub-domain of interest. The resulting reduced data set consisting of the positively annotated blogs along with all their corresponding replies were subject to full text processing; the objective of course was to come up with supporting grounds for the sub-domain examples to follow.

6.6.2 Semantic Tagging Experiments

The next step following data analysis consists of "semantic tagging", for which the help of a group of almost 80 persons, with varying degrees of commitment and responsiveness, was called upon. Different textual blogs along with possibly matching axioms represented in natural language were regularly forwarded to these "taggers". These particular sub-experiments were conducted on a total of 2000 posts. Results

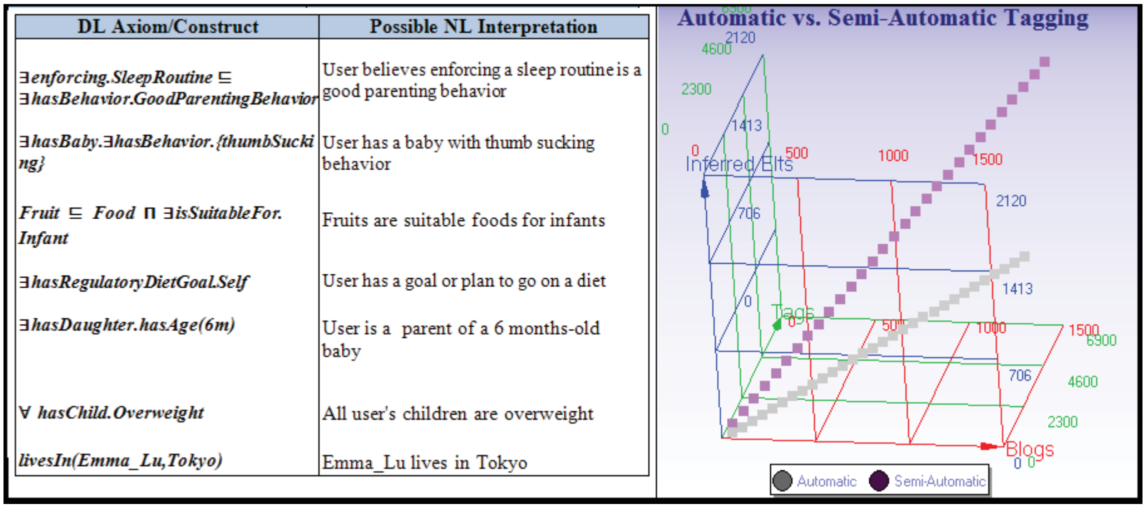
---

[1] http://gate.ac.uk/sale/thakker-jape-tutorial

were collected, analyzed and interpreted for the population of "ParOnt" with suitable *ABox* assertional data consisting of individuals (mainly persons) with certain traits and criteria, behaviors, convictions, etc.

Figure 6-5 below provides, to the left, sample generated semantics (represented in DL axioms/constructs) along with their contextual natural language verbalized interpretation. The graph to the right highlights throughout a straightforward comparison the advantage of incorporating "semantic tagging", translated in a considerable increase in the number of inferred elements (checked and validated instances, answered queries, processed rules, etc.). The upper plot illustrates the presence of tags whereas the lower one sketches their absence. The comparative experiment depicted in this figure was conducted under the scope of the work presented in [27].

It is noteworthy that the estimations provided for the comparison of tagging elements according to the automatic (lower plot line), vs. the semi-automatic (upper plot line) approaches are not static numbers; their exact value is not important in itself. The idea is to prove that for a given number of analyzed blogs (represented by the x blue axis), the number of tags (represented by the y green axis) easily doubles between an automatic and a semi-automatic approach, proportionally with the number of inferred new knowledge, of services, etc. (represented by the x blue axis).



DL Axiom/Construct	Possible NL Interpretation
$\exists$***enforcing.SleepRoutine*** $\sqsubseteq$ $\exists$***hasBehavior.GoodParentingBehavior***	User believes enforcing a sleep routine is a good parenting behavior
$\exists$***hasBaby.***$\exists$***hasBehavior.{thumbSucking}***	User has a baby with thumb sucking behavior
Fruit $\sqsubseteq$ ***Food*** $\sqcap$ $\exists$***isSuitableFor.Infant***	Fruits are suitable foods for infants
$\exists$***hasRegulatoryDietGoal.Self***	User has a goal or plan to go on a diet
$\exists$***hasDaughter.hasAge(6m)***	User is a parent of a 6 months-old baby
$\forall$ ***hasChild.Overweight***	All user's children are overweight
livesIn(Emma_Lu,Tokyo)	Emma_Lu lives in Tokyo

Figure 6-5: To the left, NLG semantic tags examples; to the right, highlighting the "Semantic Tagging" effect

Needless to mention, with an increased level of ontological complexity for data analysis, the weight of semantic tagging is expected to grow exponentially. In other words, when the ontology-based grammars go beyond the taxonomy and look for meaningful complex expressions in the text, the role of validating the tags suggestions becomes much more significant.

Having reached this stage, reasoning procedures that follow the already discussed mechanisms for redirection are applied. Interoperability is ensured through an established link between detected individuals and existing FOAF users within the communities formed through SIOC. Reasoning techniques are exploited to directly or indirectly offer the services for parenting awareness and orientation recommender systems.

6.7 Competency Questions Portfolio

This section depicts the major experimental scenarios carried out within the scope of the thesis. Starting with a holistic overview of the competency questions (CQs) evaluation metrics, it continues to highlight and delve into use cases singled out from the complete set of explored cases. Moreover, it presents an overview of the efforts undertaken for the childhood obesity surveillance subdomain.

6.7.1 Competency Questions Portfolio Holistic Evaluation

Based on the data described earlier in this chapter, a portfolio of different use cases and competency questions aiming at proving the model's usability and effectiveness was built. The tables to follow report its main characterizing aspects and findings.

Table 6.2 provides an overview of the competency questions' descriptive metrics. In summary, more than 200 CQs were built in accordance with questions translated from existing blogs, or prepared based on interesting domain use cases including community segmentation scenarios and recommendation case studies. Groups of relevant

CQ Metrics Element	Value
Number of CQs	200
Number of sub-ontologies	36
Number of analyzed blogs	150K
Defined Individuals	5.5K
Populated Individuals	20K
Tagging-supported scenarios (%)	60
DL-Decidable CQ(%)	84
Success in all fragments(%)	11

Table 6.2: Competency Questions Portfolio Descriptive Elements

CQ signature elements were used to form 36 ParOnt-based reduced sub-ontologies. A reduced ontology had its projected versions corresponding to each of the OWL 2 profiles prepared.

While to one particular CQ corresponds one mostly suitable fragment, the CQ portfolio queries were tested against the different fragments in an effort to evaluate feasibility and decidability.

Fragment	Favorable CQs (%)	Supported CQs (%)	DL Efficiency Ratio
EL	36	92	0.41
QL	43	94	0.22
RL	13	96	0.28
DL	8	100	1

Table 6.3: Competency Questions Portfolio Experimental Results Metrics

Table 6.3 summarizes the descriptive metrics against each of the OWL 2 fragments used, including OWL 2 DL. The first column provides an estimation of the percentage of CQs favorable to each of the fragments.

In the second column, the percentage of CQs with signature metadata that support their corresponding favorable fragment is represented.

The third column displays the computation of the efficiency ratio of the execution via the profile in question, compared to its execution against the DL version. This ratio is computed through the average response time per fragment, divided by the total number of CQs, as follows:

$$\frac{\sum \frac{T_{CQ_{Fragment}}}{T_{CQ_{DL}}}}{N_{CQ}} \tag{6.1}$$

where:

$T_{CQ_{Fragment}}$: *Fragment Response Time*; Response time of the CQ based on its fragment-projected ontology version

$T_{CQ_{DL}}$: *DL Response Time*; Response time of the CQ based on its DL ontology version

N_{CQ} : *Total CQ Number*; Total number of executed CQs

Thus, when this ratio is equal for example to 0.25, it reflects the fact that the fragment in question is 4 times more efficient than its corresponding DL version.

The displayed results demonstrate the usability of the different profiles, and the efficiency of profile-based reasoning over DL reasoning. At the same time, they serve to show that DL decidability issues are more theoretical than practical, as a vast majority of the implemented CQs can be applicable in DL, and response time is often not a restrictive constraint: the worst case complexity caveat has more theoretical than practical implications.

For the UC scenarios requiring the most efficient possible response time, the answers resulting from the fragment-based reasoning can be provided first, waiting for the DL response to be ready (after setting it to execute in parallel in the background).

6.7.2 Highlighted Use Case Scenarios and Competency Questions

The below reported cases provide more detailed CQ examples; they are presented along with their context, usability, and potential profile. When appropriate, the main signature elements exemplified via sample axioms are provided, along with applicable queries and rules.

It is noteworthy that the language used to illustrate the different CQs is considerably informal and simple, thus reflecting the nature of the language used during blogging and forum exchanges.

Conjunctive Queries and Rules Syntactic Definitions

Preliminary definition of exploited syntactical elements are provided herein.

The conjunctive queries and rules that illustrate the examples to follow are all based on parts of the knowledge base $KB = (T, A)$, where T is a restricted finite set of inclusion and functionality axioms, called $TBox$, and A is a finite set of membership axioms, called $ABox$. A conjunctive query is a conjunction of atoms or query conditions, where every atom is either a simple, logical operator-free description logic formula, or its negation.

A rule atom can take the following form:

$$\forall x_1 ... \forall x_m . (B_1 \wedge ... B_k \rightarrow H),$$

where $B_1 ... B_k$ are atoms, and $x_1 ... x_m$ are exactly the variables that occur within these atoms.

A conjunctive query takes the form:

$\exists x, y\ \varphi(x, y)$, where φ is a conjunction of atoms and x and y are variable or individual names.

A query atom can take one of the following forms:

$C(x)$ *or* $\neg C(x)$ where C is a class name, and x is a variable or an individual's name.

R(x,y) where R is a property name, and x and y are individual or variable names.

CQ Regular Workflow Illustration

Consider the following use case scenario, in which a SN user is "wondering whether "Sunflower seeds" are known to cause allergy for children."
Figure 3-5 in Chapter 3 can be referred to for more clarifications.

1. **UC Disambiguation:** UC information collection (such as UC originator, checking whether it has been previously executed (by referring to KB UC dictionary structures)).

2. **Signature and CQ Estimaton:** Depending on the previous step's results. For the time being, it is supposed that this is a first time executed UC; the workflow thus proceeds.

3. **Ontology-Aware NLP:** KB existing contents having been validated to be short of the UC corresponding signature and CQ, results of this step indicate the following:
   SunflowerSeeds: ABox instance;
   Allergy: TBox concept;
   canCause, alwaysCause, hasAllergyRisk: RBox roles;

4. **Tag Recommendation:** Represented herein is the output forwarded to the end user, DL axioms underline these outcomes; "Is this what you are looking for?
   SunflowerSeeds is a type of edible seed, a healthy food;
   Allergy is a hypersensitive immune response a type of disease disorder;
   What are causes of allergies?
   Can sunflower seeds cause allergies?
   What are the benefits of sunflower seeds?"

5. **SN User Tagging:** underlines the validation of the statement:
   "Can sunflower seeds cause allergies?".
   Signature = $\{AllergyCausingFood\}$.
6. **Profile Assessment:** Instance checking, extensional reasoning on an ABox element. DL reasoning
7. **Projection:** DL projection, rendering obsolete the DL restrictive axioms.
8. **Reasoning:** standard instance checking returning the query answer:
   Sunflower seeds can cause allergies.

Case Study 1: SN Users profiling

Use Case Objective: Linking parents having children with development disorders. The following axioms and assertions illustrate samples used for this case study:

$$\text{Autism} : \text{NeuralDevelopmentDisorder} \tag{1}$$

$$\text{Parent} \sqcap \exists\text{hasChild}.\exists\text{suffersFrom}.\text{NeuralDevelopmentDisorder} \sqsubseteq \exists\text{needPromoted}.\text{SNActivities} \tag{2}$$

The purpose of this use case is to exploit the framework's capabilities in localizing parents sharing the same experience of having a child with a certain neural disorder (autism in this example).

At the medical level, there are no particular recommendations generated. However, at the social level, it is known that encouraging these parents to get involved in SN activities can bring them considerable psychological support.

Simple mining grammars are enough to recognize such parents (defined in Axiom (2)'s antecedent). Reasoning associates these individuals to their corresponding classes, and deduces the promoted social recommendations (defined in Axiom (2)'s consequent). Instances of these SN recommendations can be: social events invitations forwarding, contact information exchange (after the consent of the SN user in question), etc.

The following conjunctive query can be used to build SNs of such users (the last clause is optional, it is added to highlight that geographic neighborhood is also relevant):

$$Q_1(x) =$$
$$\exists \text{x,y (Parent(x)} \land \text{hasChild(x,y)}$$
$$\land \text{ suffersFrom(y,autism)} \land \text{livesIn(x,montreal))}$$

This use case and similar ones are typical scenarios through which EL as well as QL reasoning and querying stimulate the creation of semantic social networks.

Case Study 2: Rule Implementations

The following includes a list of rules and OWL 2 RL axioms that were particularly useful for the "Safety and First Aid" subdomain, for orientation and awareness means:

$$\begin{aligned} &\mathit{experience}(\mathit{child}, \mathit{falling}) \land \mathit{hasSymptom}(\mathit{child}, \mathit{looseTooth}) \\ &\qquad \rightarrow \mathit{recommended}(\mathit{dentistVisit}, \mathit{child}) \end{aligned} \tag{1}$$

$$\begin{aligned} &\mathit{experience}(\mathit{child}, \mathit{falling}) \land \mathit{hasSymptom}(\mathit{child}, \mathit{brokenTooth}) \\ &\qquad \land \mathit{hasSymptom}(\mathit{child}, \mathit{breathingProblem}) \\ &\qquad \rightarrow \mathit{recommended}(\mathit{headingToER}, \mathit{child}) \end{aligned} \tag{2}$$

$$\begin{aligned} &\mathit{experience}(\mathit{child}, \mathit{hotDrinkPouring}) \land \mathit{hasSymptom}(\mathit{child}, \mathit{burns}) \\ &\qquad \land \mathit{affect}(\mathit{burns}, \mathit{criticalBodyPart}) \\ &\qquad \rightarrow \mathit{recommended}(\mathit{headingToER}, \mathit{child}) \end{aligned} \tag{3}$$

(1), (2) and (3) represent sample defined rule axioms.

$$CriticalBodyPart \equiv \{genitals, face, hands, feet\} \tag{4}$$

$$HomeRemedy \sqsubseteq \exists isPossibly.FallacyBurnRemedy \tag{5}$$

$$FallacyBurnRemedy(butter) \tag{6}$$

$$FallacyBurnRemedy(oil) \tag{7}$$

$$FallacyBurnRemedy(mayonnaise) \tag{8}$$

$$FallacyBurnRemedy(petroleumJelly) \tag{9}$$

Rule (1) states that it is recommended to take the child to the dentist (and not to ER like some blog replies suggested) in case they hit the ground and knock a tooth loose.

Rule (2) specifies different conditions to head to ER: the tooth was lost and the child exhibits certain breathing problems (examples: coughing, wheezing).

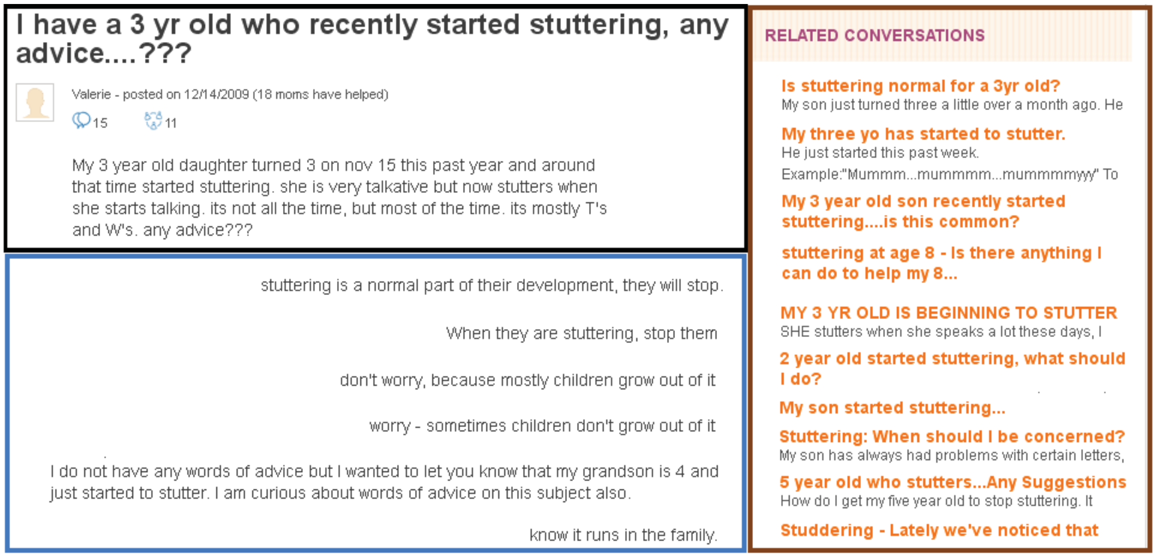
Rule (3) states what should be recommended following a child's pouring of a hot drink on some critical part of their body.

Definition (4) identifies the nominals forming the critical body parts.

The subsumption (5) expresses that home remedies are not recommended to cure burns (with annotated explanations: the reason is they create a barrier that holds the heat in), examples of such remedies are also explicitly mentioned in ABox instance definitions (6) to (9).

Case Study 3: Children Stuttering

As illustrated in Figure 6-6, which contents' further details can be easily searched and browsed, to the same question, there might be tons of different answers that do not take into consideration prerequisite constraints, are very subjective and based on bloggers' personal experience; sometimes the answers are ambiguous and contradictory (act/do not act, worry/do not worry, etc.).



I have a 3 yr old who recently started stuttering, any advice....???

Valerie - posted on 12/14/2009 (18 moms have helped)

15 11

My 3 year old daughter turned 3 on nov 15 this past year and around that time started stuttering. she is very talkative but now stutters when she starts talking. its not all the time, but most of the time. its mostly T's and W's. any advice???

stuttering is a normal part of their development, they will stop.

When they are stuttering, stop them

don't worry, because mostly children grow out of it

worry - sometimes children don't grow out of it

I do not have any words of advice but I wanted to let you know that my grandson is 4 and just started to stutter. I am curious about words of advice on this subject also.

know it runs in the family.

RELATED CONVERSATIONS

Is stuttering normal for a 3yr old?
My son just turned three a little over a month ago. He

My three yo has started to stutter.
He just started this past week.
Example:"Mummm...mummmm...mummmmyyy" To

My 3 year old son recently started stuttering....is this common?

stuttering at age 8 - Is there anything I can do to help my 8...

MY 3 YR OLD IS BEGINNING TO STUTTER
SHE stutters when she speaks a lot these days, I

2 year old started stuttering, what should I do?

My son started stuttering...

Stuttering: When should I be concerned?
My son has always had problems with certain letters,

5 year old who stutters...Any Suggestions
How do I get my five year old to stop stuttering. It

Studdering - Lately we've noticed that

Figure 6-6: Stuttering Posts Examples

Most Relevant KB axioms for this example's execution:

$$\mathit{GoodStutteringRecommendation} \sqcap \mathit{BadStutteringRecommendation} \sqsubseteq \bot \tag{1}$$

$$\mathit{GoodStutteringRecommendation} \sqsubseteq \mathit{GoodRecommendation} \tag{2}$$

$$\mathit{BadStutteringRecommendation} \sqsubseteq \mathit{BadRecommendation} \tag{3}$$

$$\mathit{Post} \sqcap \geq 1\ \mathit{contains}.\mathit{BadRecommendation} \sqsubseteq \mathit{MisleadingPost} \tag{4}$$

$$\mathit{Blogger} \sqcap \geq 2\ \mathit{recommends}.\mathit{BadRecommendation} \sqsubseteq \mathit{NonTrustworthyBlogger} \tag{5}$$

$$\mathit{Blogger} \sqcap \leq 1\ \mathit{recommends}.\mathit{BadRecommendation} \sqsubseteq \mathit{TrustworthyBlogger} \tag{6}$$

These axioms classify good and bad recommendations by establishing a disjointness between them (1); they also divide recommendations into sub-categories (stuttering, good and bad recommendations) in (2, 3). Qualified Cardinality Restriction expressions (QCRs) are then employed to build the definition of a misleading post (containing a bad recommendation), from which trustworthy vs. non trustworthy bloggers are inferred.

$$Transitive(follows) \quad (7)$$
$$Transitive(precedes) \quad (8)$$
$$follows \equiv precedes^{-} \quad (9)$$
$$\top \sqsubseteq \forall follows.DevelopmentalStage \quad (10)$$
$$\top \sqsubseteq \forall precedes.DevelopmentalStage \quad (11)$$
$$\exists precedes.\top \sqsubseteq DevelopmentalStage \quad (12)$$
$$\exists follows.\top \sqsubseteq DevelopmentalStage \quad (13)$$
$$\exists hasAge.4 \sqsubseteq preschoolStage \quad (14)$$

This list of axioms defines developmental stages and the transitive and inverse properties that link them. From the child's age (example in axiom (14)), his developmental stage is determined, and so are the appropriate recommendations.

In the below, a child who is stuttering while his parents have a reprimanding reaction is likely to acquire an introverted attitude.

$$\begin{gathered}\text{experience(Child, stuttering)} \land \text{hasParent(Child, Parent)} \\ \land \text{ hasAttitude(parent, reprimandingAttitude)} \\ \rightarrow \\ \text{acquire(Child,introvertedAttitude)}\end{gathered}$$

This use case validates the effective role of the framework in:

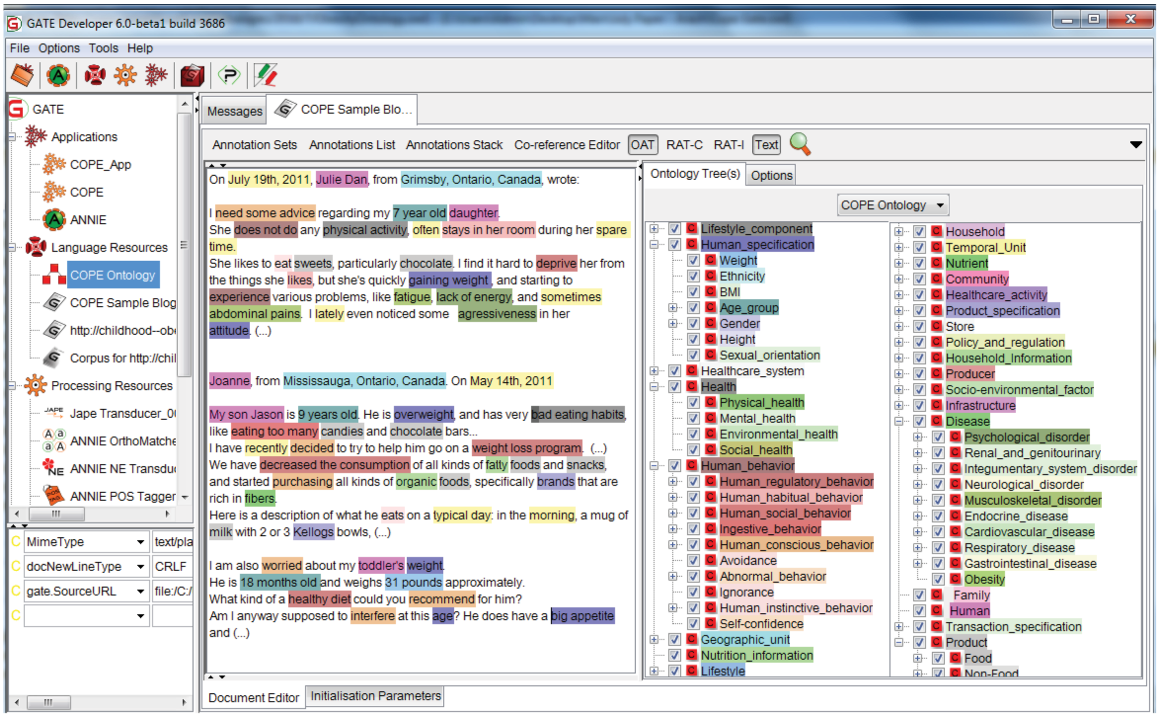
- Detecting posts about stuttering, parents' relevant interests and inquiries

- Sorting out the diversity of replies and opinions, and proving they include inaccuracies, and sometimes fallacies.
- Demonstrating the reasoning translated in recommended behaviors based on suitable preconditions

Childhood Obesity Surveillance Subdomain Experiments

Prior to the establishment of the ParOnt ontology, considerable efforts targeting the childhood obesity parenting subdomain were carried out in [28].
The following details aim at summarizing them.

First, illustrating the outcome of the SN data analysis strategy, Figure 6-7 depicts GATE sample extracts from blogs with meaningful ontology annotations .



On July 19th, 2011, Julie Dan, from Grimsby, Ontario, Canada, wrote:

I need some advice regarding my 7 year old daughter.
She does not do any physical activity, often stays in her room during her spare time.
She likes to eat sweets, particularly chocolate. I find it hard to deprive her from the things she likes, but she's quickly gaining weight , and starting to experience various problems, like fatigue, lack of energy, and sometimes abdominal pains. I lately even noticed some agressiveness in her attitude. (...)

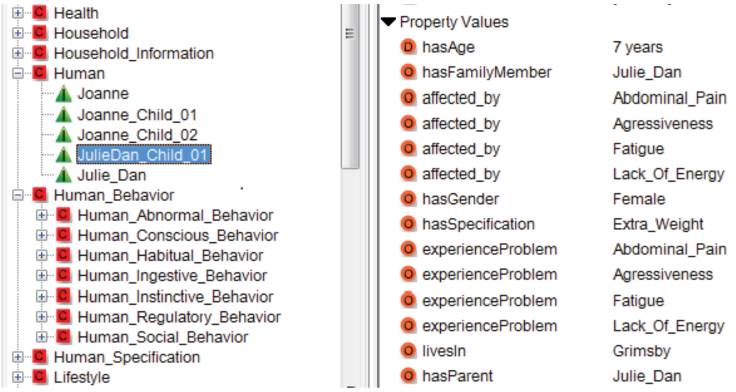
Joanne, from Mississauga, Ontario, Canada. On May 14th, 2011

My son Jason is 9 years old. He is overweight, and has very bad eating habits, like eating too many candies and chocolate bars...
I have recently decided to try to help him go on a weight loss program. (...)
We have decreased the consumption of all kinds of fatty foods and snacks, and started purchasing all kinds of organic foods, specifically brands that are rich in fibers.
Here is a description of what he eats on a typical day: in the morning, a mug of milk with 2 or 3 Kellogs bowls, (...)

I am also worried about my toddler's weight.
He is 18 months old and weighs 31 pounds approximately.
What kind of a healthy diet could you recommend for him?
Am I anyway supposed to interfere at this age? He does have a big appetite and (...)

Figure 6-7: Sample processed annotations using GATE.
(Note that the results have been shown in different color codes)

Original markups and default GATE PR annotations are merged with ontological elements from the COPE Ontology to detect geographical and temporal units, humans' names, household, health, and lifestyle data, behavioral information, products and food types, etc.



hasAge	7 years
hasFamilyMember	Julie_Dan
affected_by	Abdominal_Pain
affected_by	Agressiveness
affected_by	Fatigue
affected_by	Lack_Of_Energy
hasGender	Female
hasSpecification	Extra_Weight
experienceProblem	Abdominal_Pain
experienceProblem	Agressiveness
experienceProblem	Fatigue
experienceProblem	Lack_Of_Energy
livesIn	Grimsby
hasParent	Julie_Dan

Figure 6-8: Sample Individual Properties and Values

Figure 6-8 concentrates on identified properties along with their values, as assigned to a certain "Human" retrieved individual. Pertinent examples of ontological data processing within the scope of the COPE project can for instance consist of the identification of individuals that have children with diet-related problems, and/or who might be presenting certain abnormal behaviors or health symptoms, as well as the classification of individuals in the ontology according to several factors, among others.

On the other hand, the KB depicted below, followed by Table 6.4, contain a subset of sample axioms, queries and rules:

$\mathrm{hasAge(Sara\_Miller,\ 12)} \ \rightarrow Child(Sara\_Miller)$

$PsychologicalDisease(Obesity)$

$PhysicalDisease(Obesity)$

$affectedBy(Sara\_Miller, Obesity)$

$Child(x) \rightarrow Human(x)$

$Child(x) \land affectedBy(x, Obesity) \rightarrow currentlyRisk(x, PoorSelfEsteem)$

$Child(x) \land affectedBy(x, Obesity) \rightarrow currentlyRisk(x, NegativeSelfImage)$

$$Child(x) \wedge affectedBy(x, Obesity) \rightarrow currentlyRisk(x, SocialIsolation)$$

$$Child(x) \wedge affectedBy(x, Obesity) \rightarrow currentlyRisk(x, PeersTeasing)$$

$$Child(x) \wedge affectedBy(x, Obesity) \rightarrow currentlyRisk(x, SadnessFeelings)$$

$$Human(x) \wedge affectedBy(x, Obesity) \rightarrow hasFutureRisk(x, CVD)$$

$$Human(x) \wedge affectedBy(x, Obesity) \rightarrow hasFutureRisk(x, ElevatedBloodLipidLevel)$$

$$Human(x) \wedge affectedBy(x, Obesity) \rightarrow hasFutureRisk(x, HighBloodPressure)$$

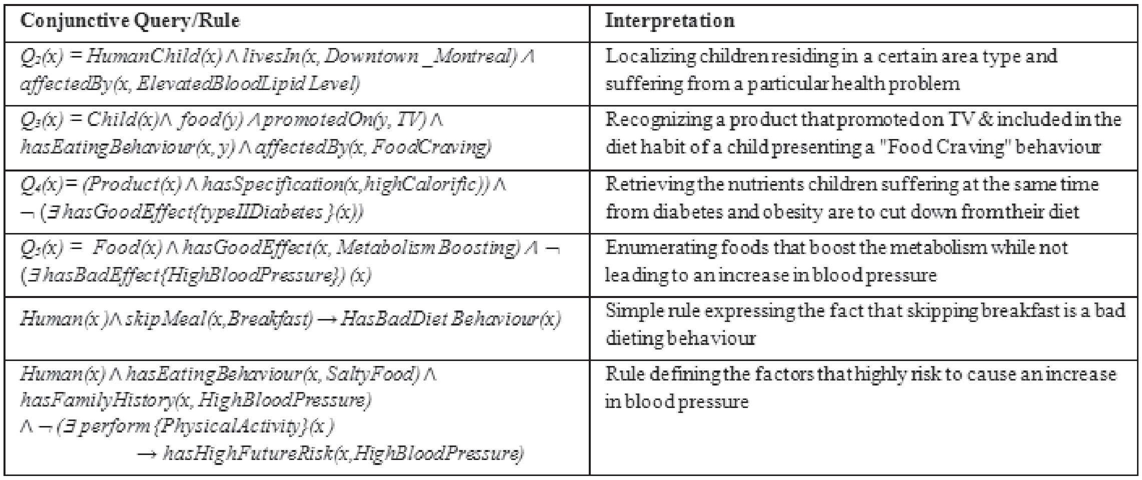
$$Human(x) \wedge affectedBy(x, Obesity) \rightarrow hasFutureRisk(x, TypeIIDiabetes)$$

The KB query allowing the retrieval of inferred risks based on the fact that Sara Miller is a child who suffers from obesity (the first 3 KB axioms), is the following:

$$Q_6(x) =$$

$$\exists \text{x} ((\text{Disease(x)} \wedge \text{currentlyRisk(Sara\_Miller, x)}) \vee$$

$$(\text{MedicalRisk(x)} \wedge \text{hasFutureRisk(Sara\_Miller, x)}))$$



Conjunctive Query/Rule	Interpretation
$Q_2(x) = HumanChild(x) \wedge livesIn(x, Downtown\_Montreal) \wedge affectedBy(x, ElevatedBloodLipid\ Level)$	Localizing children residing in a certain area type and suffering from a particular health problem
$Q_3(x) = Child(x) \wedge food(y) \wedge promotedOn(y, TV) \wedge hasEatingBehaviour(x, y) \wedge affectedBy(x, FoodCraving)$	Recognizing a product that promoted on TV & included in the diet habit of a child presenting a "Food Craving" behaviour
$Q_4(x) = (Product(x) \wedge hasSpecification(x, highCalorific)) \wedge \neg (\exists hasGoodEffect\{typeIIDiabetes\}(x))$	Retrieving the nutrients children suffering at the same time from diabetes and obesity are to cut down from their diet
$Q_5(x) = Food(x) \wedge hasGoodEffect(x, Metabolism\ Boosting) \wedge \neg (\exists hasBadEffect\{HighBloodPressure\})(x)$	Enumerating foods that boost the metabolism while not leading to an increase in blood pressure
$Human(x) \wedge skipMeal(x, Breakfast) \rightarrow HasBadDiet\ Behaviour(x)$	Simple rule expressing the fact that skipping breakfast is a bad dieting behaviour
$Human(x) \wedge hasEatingBehaviour(x, SaltyFood) \wedge hasFamilyHistory(x, HighBloodPressure) \wedge \neg (\exists perform\{PhysicalActivity\}(x) \rightarrow hasHighFutureRisk(x, HighBloodPressure)$	Rule defining the factors that highly risk to cause an increase in blood pressure

Table 6.4: Sample Queries and Rules along with their Interpretation

In the following, a list of queries is presented with KB query answer examples. Such queries can be predefined as useful frequently asked questions the users can regularly verify and similarly apply to other subdomains:

- A healthy lifestyle involves the whole family, with a set of activities and behaviors to be adopted by all family members, (and not simply by the child following the weight loss program for example).

  Outdoor and indoor family activities, commitment from the whole family, hiding bad habits from children, showing constant interest in healthy food, providing moral support, etc., are all examples from the resulting outcome of the following query:

  $$Q_7(x) =$$
  $$\exists \text{x} ((\text{HumanHabitualBehavior(x)} \wedge \text{involves (x,WholeFamily)}$$
  $$\wedge \text{ hasGoodEffect (x,HealthyLifestyle))} \vee$$
  $$(\text{HumanSocialBehavior(x)} \wedge \text{involves (x,WholeFamily)}$$
  $$\wedge \text{ hasGoodEffect (x,HealthyLifestyle))} \vee$$
  $$(\text{LifestyleComponent(x)} \wedge \text{involves (x,WholeFamily)}$$
  $$\wedge \text{ hasGoodEffect (x,HealthyLifestyle)))}$$

- Children forced to adhere to a weight loss diet often experience psychological problems.

  It is interesting to obtain a list of children on a diet and not being followed up by some professional (for example a psychiatrist or a nutritionist). The following query can be referred to for this purpose:

  $$Q_8(x) =$$
  $$\exists \text{x,y} ((\text{Child(x)} \wedge \text{follows(x, WeightLossDiet)}$$
  $$\wedge \text{ Therapist(y)} \wedge \neg(\exists \text{followedUpBy(x,y)))}$$

- Helpful and practical components and behaviors to be avoided for the purpose of a healthier lifestyle can be sorted out through the following query:

  $$\begin{gathered} Q_9(x) = \\ \exists \text{x } ((\text{HumanHabitualBehavior(x)} \wedge \text{hasBadEffect (x,HealthyLifestyle)}) \\ \vee (\text{HumanSocialBehavior(x)} \wedge \text{hasBadEffect (x,HealthyLifestyle)}) \\ \vee (\text{LifestyleComponent(x)} \wedge \text{hasBadEffect (x,HealthyLifestyle)})) \end{gathered}$$

  Examples of resulting components can be: less physical activity, greater reliance on cars, eating out more often, relying on sedentary entertainment (like TV, video and computer games), high-calorific food choices, bigger food portions, safety concerns related to outside free play, neighborhood designs (absence of sidewalks), children walking to school less often, etc.

- Identifying the set of parents who have at least 1 child suffering from a certain disorder is often useful for building a network of users sharing similar goals, interests, and consequently expecting similar appropriate recommendations; this can be achieved through the query:

  $$\begin{gathered} Q_{10}(x) = \\ \exists \text{x,y (Child(x)} \wedge \text{Parent (y)} \\ \wedge \text{hasChild(y,x)} \wedge \text{suffersFrom (x,Autism))} \end{gathered}$$

- Another typical blog question is to provide recommendations to deal with obesity :

  $$\begin{gathered} Q_{11}(x) = \\ \text{isRecommendedFor(x,y)} \wedge \text{Toddler(y)} \\ \wedge \text{suffersFrom(y,Obesity)} \end{gathered}$$

> While a typical Web 2.0 answer to this question would be a list of millions of web links containing the CQ keywords, with relevant and irrelevant responses, the add-on provided by the framework is a specific list of recommendations (for example: “OrganizeMeals, ReconsiderMilkBrand, PromoteOutings, SharingPlayActivities, CheckWithDoctor”).
>
> Had the additional specification BreastFedBaby(y) been part of the query, the resulting recommendation outcomes would have disregarded the “ReconsiderMilkBrand” recommended suggestion.
>
> Moreover, tagged documents and posts pertaining to the exact query conditions can be obtained.
>
> All the framework outcomes take place on top of the existing Web 2.0 results, without disregarding them.

Appendix C contains an additional set of conjunctive queries.

6.8 Conclusions

This chapter presented the experiments and empirical results carried out for the purpose of proving the potentials of the conceptual framework, its value and effectiveness.

The experiments were designed with an emphasis on illustrating the role of the ontology, of expressive modeling and expressiveness management, tagging, and of course reasoning.

Although the implementation was not a complete and integrated one, it was sufficient to validate the feasibility of the different components and suggested flows.

The experimental procedures and activities undertaken were necessary to set the grounds for the implementation of a comprehensive and integrated knowledge base for appropriate dissemination of parenting knowledge and information.

The KB role was also confirmed by the sample queries and reasoning services having the potential of contributing in overcoming the parenting domain deficiencies, and participating in transforming limitations into challenging opportunities to perform interesting tasks and rich scenarios.

Chapter 7

Conclusions and Future Work

7.1 Contributions

This thesis presented a Web 3.0 conceptual framework which, to the best of our knowledge, is unique in several ways and at more than one level. Stemming from the Model Driven Architecture's fundamental principles, the proposed framework consists in an ontology-driven integrated approach built on standards and collaborations between well-established organizations, technologies and disciplines.

In Chapter 1, which contents are revisited in this section, the research problems and objectives were classified to be viewed under three main angles:

1. Semantic Web Expressiveness and the Need for Ontologies in Web 3.0
2. Social Parenting Data Management in Web 3.0
3. Metamodeling Standards and Web 3.0

Similarly, and based on this same classification, the detailed aspects of the contribution will next be accentuated through a sequence of $\mathcal{M}$ followed by $\mathcal{C}$ elements denoting the following:

- $\mathcal{M}$: Motivation research factor, representing summarized elements from the problems statement and research objectives.

- $\mathcal{C}$: Contribution framework aspect, describing how the thesis addresses its preceding pointed out motivations, taking into consideration the fact that the aspect may bring a direct or an indirect solution to the stated problem.

1. **Semantic Web Expressiveness and the Need for Ontologies in Web 3.0:**

$\mathcal{M}$	:	A great reliance on RDF (Resource Description Framework) graph data for the majority of Semantic Web realizations for social networks. A lack in efforts involving the Semantic Web's advanced findings and relatively complicated vocabularies and grammars, particularly in the endeavors related to OWL 2 (Web Ontology Language) novelties.
$\mathcal{C}$	:	A coherent ontology-driven framework based on OWL 2 and its novelties (added expressiveness, tractable fragments, reasoning advances), and building on the existing efforts through alignments and cooperations with available well-established SN ontologies. An expressiveness management approach founded on projection strategies including signature-based modularization and Description Logics fragments reduction. Methodologies and algorithms in accordance with the OWL 2 profiles specifications are typically provided. Suggested methods and applications to benefit from the presence of extra expressiveness for a variety of use cases that do not encompass typical reasoning and querying services.

$\mathcal{M}$: Semantic tagging Web 3.0 computation and recommendation endeavors still take limited advantage of expressiveness and standardization advances.

$\mathcal{C}$: Ontology and reasoning-supported strategies for tag management and tags recommendations. Through these strategies, tags are identified as expressive axioms and constructs from the knowledge base.

Advocated mechanisms to suggest semantic tags following ontology modularization, disjunctive constructs, and abduction non-standard reasoning techniques.

$\mathcal{M}$: Data mining raised concerns as to the applicability of "expressive" ontology-aware Natural Language Processing for Web 3.0.

$\mathcal{C}$: A strategy for applying ontology-aware pattern-matching grammars for data mining through NLP that is reinforced with the semantic tagging approach. Thus, tagging is suggested as a workaround or an indirect solution to overcome the aforementioned concerns.

2. **Social Parenting Data Management in Web 3.0:**

$\mathcal{M}$: Information and parenting advice conveyed by social media often turns out to be conflicting and confusing.

Sources of reliable information are not easily accessible and tend to be difficult to evaluate, especially when it comes to the social information of a chaotic and controversial nature.

Available informational data is neither structured nor homogeneously distributed.

$\mathcal{C}$: A parenting sociomedical ontology, baptized ParOnt. This ontology contains expressive and of course structured information on the parenting domain.

A parenting knowledge base having its contents continuously strengthened by resources from knowledgeable scientists, domain experts and reliable data sources. The KB is scalable, it consolidates different sources' knowledge and builds consensus around key findings.

Means for transforming the deficiencies and areas of controversy into windows of opportunity, typically for reasoning and tagging-supported scenarios to identify controversial aspects, analyze their origins, reasons and criteria.

$\mathcal{M}$: The different approaches to manage parenting data and respond to sociomedical needs are short of a strengthened consolidated knowledge base, one that comprises ontology-supported analysis of the domain, whether it was RDF, OWL or any of their sub-languages and derivatives featuring as the exploited ontological support language.

$\mathcal{C}$: All the contribution elements described for the precedingly mentioned problematic aspects. Furthermore, the proposed ontology-driven framework comprises several factors setting off the fact that it is mostly adapted for the requirements of the sociomedical domain. These include: the knowledge base and the canonical ontology, the sociomedical expertise, the parenting online data sources, and the mommy blogger SN users expected to maintain maximum cooperation and involvement in the emphasized tagging activities.

Several presented framework possible workflows, along with experimental scenarios based on the domain ontology. These are illustrated by real-life examples demonstrating the usability and practicality of Web3.OWL.

---

3. **Metamodeling Standards and Web 3.0:**

$\mathcal{M}$: MDA standards (SMOF and ODM) lack requirements for the interoperability between Semantic Web languages, and the establishment of means of conformance for the logical languages such as OWL 2 (along with its Existential, Rule and Query Language profiles EL, RL and QL).

$\mathcal{C}$: An SMOF standardized metamodel in which ontologies and their metadata are modeled and arranged, this metamodel includes an extension to the standard with additional "meta-semantics" structures in order to be able to incorporate the framework's requirements.
A partial response to the OMG Task Forces RFP, through the above mentioned extensions, covering mappings, signatures, categorization, modularization and alignment facilities.

7.2 Future Work

In terms of future work, the plan is to keep on fostering the different efforts pertaining to this conceptual framework.
These include:

- Establishing intense scrutiny of profiles and rule languages, looking for the incorporation of maximized sets of rules and Description Logics-based fragments; enhancing the projection algorithms, the tagging recommendations strategy based on additional explorations of semantic checkers, of non-standard reasoners and reasoning techniques.

- Elaborating further implementations that target the integrated framework in a Big Data cloud environment within which knowledge acquisition systems run on Hadoop [87] and NoSQL clusters. Achieving such an endeavor has the potential of greatly boosting the framework's efficiency in dealing with the Web 2.0 sources of massive data.

- Assimilating and evaluating Social Network Analysis methods, typical SNA methods include centrality and similarity measures, clustering methods, etc. These methods had been on the initial framework roadmap since they can be exploited for various decision-support outcomes; other priorities and time constraints prevented their incorporation.

- Researching the feasibility of exploiting the framework similarly for the requirements of other domains. The sociomedical domain's online resources and SN users offered a particular opportunity, the quest will thus be for other domains that could present similar advantages. Alternatively, for domains with different characteristics and challenging aspects, exploring what possible component replacements or workarounds can be considered for these domains.

- Integrating the SMOF extensions once the OntolOp called upon framework will be formalized and issued; comparing and assessing the differences between a potential OMG recommendation and the framework's conceived extensions, their corresponding advantages and disadvantages.

This thesis proposed and presented Web3.OWL while targeting its specific critical aspects, most particularly those related to DL and the Semantic Web. However, while attempting to estimate possible future related endeavors, it is worth looking at the proposed framework from the broadest perspective.

On the one hand, and for the time being, Web3.OWL is intended for the sociomedical domain. This domain is a fusion between the social and medical domains along with their underlined aspects. On the other hand, it is a Web 3.0 framework, and Web 3.0 is nothing but a web built on the marriage between the social and the semantic webs. The framework also abides by the standards and specifications of two well-established standardization organizations: OMG and W3C. It combines and exploits the concepts and techniques of, jointly, ontologies and databases, as well as NLP, tagging and NLG. The fact that it is ontology-driven leaves room for continuous improvements in line with reasoning and Semantic Web research advancements. As a consequence, the potentials for future work are numerous and at many levels.

In the midst of the Big Data era's outbreak with its forthcoming non-traditional approaches to handle computing problems, this framework, in our opinion, is prone to be the bedrock of many future efforts to come.

7.3 Discussions

An indirect aim of the framework is to contribute to the adoption process of the expressive Semantic Web, an adoption that lags behind its corresponding research endeavors.

With the continuous evolution of the DL and reasoning research agenda,Web3.OWL, given its extended metamodel standard, offers a smooth process to incorporate and adapt to support structural linguistic and reasoning advances.

This thesis spared no effort to show that it is by setting the ground for an ontology-driven collaboration model that semantic web accomplishments address and unravel challenging areas, like those of the sociomedical domain.

The next subsections summarize the advantages and disadvantages of the framework.

7.3.1 Summary of Challenges and Disadvantages

In Chapter 3, the disadvantages and challenging aspects of the proposed framework were presented.
Here is a summary of the most important ones:

- Similarly to MDA, a very specialized level of expertise and know-how in ontology and metadata modeling is required; such expertise cannot be easily made available.
- There obviously exists an intensive reliance on modeling and correctness of critical metadata and configurations (for semantics and scenarios). Errors and inconsistencies at this level have important effects on the whole flow.
- A comprehensive integrated implementation of the framework is intricate, particularly when individually attempted. Furthermore, the incomplete metamodeling requirements prevent the adaptation of transformations that should be made available through the standardized components.
- For scenarios within which semantic tagging is essential, there is a dependency on the cooperation from the SN users (the mommy bloggers).

7.3.2 Summary of Advantages

The advantages of the proposed conceptual approach have already been explicitly and implicitly listed throughout its chapters and within the explored contributions in the present one.

They can be quickly summarized in the following:

- Benefiting from standard reasoning services for querying the KB and for the different decision-support and recommender systems that target parenting education and public awareness.

- Overcoming expressiveness performance costs while capitalizing on the continuous evolutions in OWL and DL reasoning (fully-fledged and specialized reasoners).

- Identifying, creating and expanding social and semantic networks, extending existing ones according to explicit and clear ontological vocabularies.

- User profiling, clustering, segmentation and detection of communities and subcommunities, again according to expressive unambiguous logical constructs. These include tagging-supported surveys and studies on the controversial social domain aspects.

- Tracking web processes and activities, managing web history, reaching a detailed level of categorizing resources according to expressive logical constructs (particularly beneficial for important and sensitive resources).

7.4 Closing Remarks

In conclusion, the conceptual framework that has been proposed, again to the best of our knowledge, is the first ontology-driven Web 3.0 framework that relies on the OMG MDA standards and extends them, it is also a unique parenting framework for which the sociomedical knowledge base is supported by ontologies.

Consisting of both reused as well as innovative Web 3.0 approaches, the framework's novel features form its pillars and can be summarized in:

- The metamodeling extensions for language and sub-language interoperability, ontology alignment and modularization, among others.

- The expressiveness management approach: a mixture of per signature and per syntactic fragment ontology projection.

- The semantic tagging strategy: association of tags to logical ontology elements, and suggestion of tags according to well-defined formal expressions, possibly based on beyond-standard reasoning.

- The knowledge base and parenting ontology: a groundwork established to centralize and model the different aspects of the domain, caused to undergo incremental growth and maintenance.

The proposed conceptual sociomedical framework Web3.OWL, to conclude, is an ongoing endeavor.

Although it forms a strong foundation with ascertained contributions and demonstrated benefits, it shall necessitate further substantial efforts before reaching the mature “beyond-conceptual” framework and architecture that we aspire to build.

References

[1] M.C. Adland and M. Lykke. Social tagging in support of Cancer patients' information interaction. *Social Information Research Library and Information Science*, 5:101–128, 2012.

[2] Grigoris Antoniou, Paul Groth, Frank van Harmelen, and Rinke Hoekstra. *Semantic Web Primer*. The MIT Press, 2012.

[3] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. Dbpedia: A nucleus for a web of open data. *The Semantic Web*, pages 722–735, 2007.

[4] S. Auer and H. Herre. RapidOWL An agile knowledge engineering methodology. *Perspectives of Systems Informatics*, pages 424–430, 2007.

[5] Franz Baader, Sebastian Brandt, and Carsten Lutz. Pushing the EL envelope. In *International Joint Conference on Artificial Intelligence*, volume 19, page 364. Lawrence Erlbaum Associates LTD, 2005.

[6] Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter Patel-Schneider. *The Description Logic handbook: Theory, implementation and applications*. Cambridge University Press, 2007.

[7] Sean Bechhofer and Alistair Miles. SKOS Simple Knowledge Organization System reference. *W3C Recommendation*, 2009.

[8] Tim Berners-Lee, Mark Fischetti, and Michael L Foreword By-Dertouzos. *Weaving the Web: The original design and ultimate destiny of the World Wide Web by its inventor*. HarperInformation, 2000.

[9] Christian Bizer, Tom Heath, Kingsley Idehen, and Tim Berners-Lee. Linked data on the web (LDOW). In *Proceedings of the 17th international conference on World Wide Web*, pages 1265–1266. ACM, 2008.

[10] Roi Blanco, Berkant Barla Cambazoglu, Peter Mika, and Nicolas Torzec. Entity recommendations in web search. In *The Semantic Web–ISWC 2013*, pages 33–48. Springer, 2013.

[11] U. Bojars, J.G. Breslin, V. Peristeras, G. Tummarello, and S. Decker. Interlinking the Social Web with semantics. *Intelligent Systems, IEEE*, 23(3):29–40, 2008.

[12] U Bojārs, John G Breslin, Aidan Finn, and Stefan Decker. Using the Semantic Web for linking and reusing data across web 2.0 communities. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(1):21–28, 2008.

[13] J-M Borys, Y Le Bodo, SA Jebb, JC Seidell, C Summerbell, D Richard, Stefaan De Henauw, LA Moreno, M Romon, TLS Visscher, et al. EPODE approach for childhood obesity prevention: methods, progress and international development. *Obesity Reviews*, 13(4):299–315, 2012.

[14] J. Breslin, A. Harth, U. Bojars, and S. Decker. Towards Semantically-Interlinked Online Communities. *The Semantic Web: Research and Applications*, pages 71–83, 2005.

[15] John G. Breslin, Passant Alexandre, and Decker Stefan. *The Social Semantic Web*. Springer, 2009.

[16] D. Brickley and L. Miller. FOAF vocabulary specification 0.99. *Namespace Document*, 9, 2014.

[17] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Tractable reasoning and efficient query answering in Description Logics: The DL-Lite family. *Journal of Automated reasoning*, 39(3):385–429, 2007.

[18] Diego Calvanese, Magdalena Ortiz, Mantas Šimkus, and Giorgio Stefanoni. Reasoning about explanations for negative query answers in DL-Lite. *Journal of Artificial Intelligence Research*, 48:635–669, 2013.

[19] Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R Hruschka Jr, and Tom M Mitchell. Toward an architecture for never-ending language learning. In *Proceedings of the Twenty-Fourth Conference on Artificial Intelligence (AAAI 2010)*, volume 2, pages 3–3, 2010.

[20] Melissa Cefkin, Susanne M Glissmann, Peter J Haas, Leila Jalali, Paul P Maglio, Patricia Selinger, and Wang Tan. SPLASH: A progress report on building a platform for a 360 degree view of health. In *Proceedings of the 5td INFORMS Workshop on Data Mining and Health Informatics*. IBM, 2010.

[21] Bernardo Cuenca Grau, Ian Horrocks, Yevgeny Kazakov, and Ulrike Sattler. Modular reuse of ontologies: Theory and practice. *J. Artif. Intell. Res.(JAIR)*, 31:273–318, 2008.

[22] Hamish Cunningham. GATE, a General Architecture for Text Engineering. *Computers and the Humanities*, 36(2):223–254, 2002.

[23] Kathrin Dentler, Ronald Cornet, Annette ten Teije, and Nicolette de Keizer. Comparison of reasoners for large ontologies in the OWL 2 EL profile. *Semantic Web*, 2(2):71–87, 2011.

[24] Li Ding, Lina Zhou, Tim Finin, and Anupam Joshi. How the Semantic Web is being used: An analysis of FOAF documents. In *System Sciences, 2005. HICSS'05. Proceedings of the 38th Annual Hawaii International Conference on*, pages 113c–113c. IEEE, 2005.

[25] Laurette Dubé, Antoine Bechara, Ulf Böckenholt, Asim Ansari, Alain Dagher, Mark Daniel, Wayne S. DeSarbo, Lesley K. Fellows, Ross A. Hammond, Terry T.K. Huang, et al. Towards a brain-to-society systems model of individual choice. *Marketing letters*, 19(3):323–336, 2008.

[26] Jinan El-Hachem and Volker Haarslev. A user and NLP-assisted strategic workflow for a Social Semantic OWL 2-based knowledge platform. *Semantic Analysis in Social Media @EACL 2012, Avignon, France*, pages 37–45, 2012.

[27] Jinan El-Hachem and Volker Haarslev. When Mommy Blogs are semantically tagged. *The Joint International Semantic Technology Conference JIST, Springer Berlin Heidelberg, Nara, Japan*, pages 313–318, 2013.

[28] Jinan El-Hachem, Arash Shaban-Nejad, Volker Haarslev, Laurette Dubé, and David L. Buckeridge. An OWL 2-based knowledge platform combining the Social and Semantic Webs for an ambient childhood obesity prevention system. *Procedia Computer Science*, 10:110–119, 2012.

[29] B. Ell, D. Vrandecic, and E. Simperl. SPARTIQULATION: Verbalizing SPARQL queries. *Interacting with Linked Data (ILD 2012)*, page 50, 2012.

[30] Corinna Elsenbroich, Oliver Kutz, Ulrike Sattler, et al. A case for abductive reasoning over ontologies. In *OWLED*, volume 216, 2006.

[31] Guillaume Erétéo, Michel Buffa, Fabien L Gandon, Patrick Grohan, Mylène Leitzelman, Peter Sander, et al. A state of the art on Social Network analysis and its applications on a Semantic Web. In *SDOW2008*, 2008.

[32] Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S Weld. Open information extraction from the web. *Communications of the ACM*, 51(12):68–74, 2008.

[33] Richard Fikes, Patrick Hayes, and Ian Horrocks. OWL-QL - a language for deductive query answering on the Semantic Web. *Web semantics: Science, services and agents on the World Wide Web*, 2(1):19–29, 2004.

[34] Tim Finin, Li Ding, Lina Zhou, and Anupam Joshi. Social Networking on the Semantic Web. *The Learning Organization*, 12(5):418–435, 2005.

[35] The Harvard Center for Health Communication. The Harvard Parenting Project, last accessed: 02/02/2014.

[36] Michel Gagnon, Caroline Barrière, and Eric Charton. Full syntactic parsing for enrichment of RDF dataset. In *Linked Data for Information Extraction (LD4IE), Workshop at The Semantic Web–ISWC 2013*. Springer, 2013.

[37] Daniel Gerber, Sebastian Hellmann, Lorenz Bühmann, Tommaso Soru, Ricardo Usbeck, and Axel-Cyrille Ngonga Ngomo. Real-time RDF extraction from unstructured data streams. In *The Semantic Web–ISWC 2013*, pages 135–150. Springer, 2013.

[38] Anthony Giddens, Mitchell Duneier, and Richard P Appelbaum. *Introduction to Sociology*. WW Norton & Company, 2012.

[39] B.C. Grau, I. Horrocks, B. Motik, B. Parsia, P. Patel-Schneider, and U. Sattler. OWL 2: The next step for OWL. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(4):309–322, 2008.

[40] Benjamin N. Grosof, Ian Horrocks, Raphael Volz, and Stefan Decker. Description Logic programs: Combining logic programs with Description Logic. In *Proceedings of the 12th international conference on World Wide Web*, pages 48–57. ACM, 2003.

[41] Thomas R Gruber. Toward principles for the design of ontologies used for knowledge sharing? *International journal of human-computer studies*, 43(5):907–928, 1995.

[42] Thomas R Gruber. Collective knowledge systems: Where the Social Web meets the Semantic Web. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(1):4–13, 2008.

[43] Volker Haarslev and Ralph Möller. RACER System Description. *Automated Reasoning*, pages 701–705, 2001.

[44] Volker Haarslev, Ralph Möller, and Sebastian Wandelt. The revival of structural subsumption in tableau-based Description Logic reasoners. In *Proc. of the 21st Int. Workshop on Description Logics (DL 2008)*, volume 353, 2008.

[45] Sebastian Hellmann, Jens Lehmann, Sören Auer, and Martin Brümmer. Integrating NLP using linked data. In *The Semantic Web–ISWC 2013*, pages 98–113. Springer, 2013.

[46] Jim Hendler. Web 3.0 emerging. *IEEE Computer*, 42(1):111–113, 2009.

[47] Pascal Hitzler, Markus Krötzsch, and Sebastian Rudolph. *Foundations of Semantic Web Technologies*. Chapman and Hall/CRC, 2011.

[48] Masahiro Hori, Jérôme Euzenat, and Peter F Patel-Schneider. OWL Web Ontology Language XML presentation syntax. *W3C Note*, 11, 2003.

[49] Matthew Horridge and Sean Bechhofer. The OWL API: A java API for OWL ontologies. *Semantic Web*, 2(1):11–21, 2011.

[50] Matthew Horridge, Holger Knublauch, Alan Rector, Robert Stevens, and Chris Wroe. A practical guide to building OWL ontologies using the Protégé-OWL plugin and CO-ODE tools - edition 1.0. *University of Manchester*, 2004.

[51] Ian Horrocks. Ontologies and the Semantic Web. *Communications of the ACM*, 51(12):58–67, 2008.

[52] Ian Horrocks, Oliver Kutz, and Ulrike Sattler. The even more irresistible SROIQ. *KR*, 6:57–67, 2006.

[53] Ian Horrocks, Peter F Patel-Schneider, and Frank Van Harmelen. From SHIQ and RDF to OWL: The making of a Web Ontology Language. *Web semantics: science, services and agents on the World Wide Web*, 1(1):7–26, 2003.

[54] A. Jena. Semantic Web framework for Java, 2007.

[55] Yevgeny Kazakov, Markus Krötzsch, and Frantisek Simančík. Concurrent classification of EL Ontologies. *The Semantic Web–ISWC 2011*, pages 305–320, 2011.

[56] Hak-Lae Kim, Stefan Decker, and John G Breslin. Representing and sharing folksonomies with semantics. *Journal of Information Science*, 36(1):57–72, 2010.

[57] Szymon Klarman, Ulle Endriss, and Stefan Schlobach. ABox abduction in the Description Logic ALC. *Journal of Automated Reasoning*, 46(1):43–80, 2011.

[58] Anneke G Kleppe, Jos B Warmer, and Wim Bast. *MDA explained, the Model Driven Architecture: Practice and promise*. Addison-Wesley Professional, 2003.

[59] Patrick Lambrix, Zlatan Dragisic, and Valentina Ivanova. Get my pizza right: Repairing missing is-a relations in ALC ontologies. In *Semantic Technology*, pages 17–32. Springer, 2013.

[60] Thorsten Liebig, Marko Luther, Olaf Noppens, and Michael Wessel. OWLLink. *Semantic Web – Interoperability, Usability, Applicability*, 2(1):23–32, 2011.

[61] Nicholas Long. E-Parenting. *Handbook of parenting: theory and research for practice*, pages 369–379, 2004.

[62] Peter Mika. Flink: Semantic Web technology for the extraction and analysis of Social Networks. *Web Semantics: Science, Services and Agents on the World Wide Web*, 3(2):211–223, 2005.

[63] Peter Mika. Ontologies are us: A unified model of Social Networks and semantics. *The Semantic Web–ISWC 2005*, pages 522–536, 2005.

[64] Peter Mika. *Social Networks and the Semantic Web. Semantic Web And Beyond: Computing for Human Experience*. Springer, 2007.

[65] Peter Mika. Current issues with Social Network representations, 2008.

[66] Peter Mika and Mark Greaves. Editorial: Semantic Web & web 2.0. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(1), 2012.

[67] Aleksandra Klasnja Milicevic, Alexandros Nanopoulos, and Mirjana Ivanovic. Social tagging in recommender systems: a survey of the state-of-the-art and possible extensions. *Artificial Intelligence Review*, 33(3):187–209, 2010.

[68] Boris Motik, Bernardo Cuenca Grau, Ian Horrocks, Zhe Wu, Achille Fokoue, and Carsten Lutz. OWL 2 Web Ontology Language: Profiles. *W3C Recommendation*, 27:61, 2009.

[69] Boris Motik, Rob Shearer, and Ian Horrocks. Optimized reasoning in Description Logics using hypertableaux. *Automated Deduction–CADE-21*, pages 67–83, 2007.

[70] Object Management Group OMG. Ontology Definition Metamodel (ODM) version 1.0, 2009.

[71] Object Management Group OMG. Meta Object Facility (MOF) 2.0 core specification, 2011.

[72] Object Management Group OMG. MOF Support for semantic structures (SMOF) version 1.0, 2013.

[73] Object Management Group OMG. Ontology, model and specification integration and interoperability (OntoIOp) request for proposal, 2013.

[74] A. Passant and P. Laublet. Meaning of a tag: A collaborative approach to bridge the gap between tagging and linked data. In *Proceedings of the WWW 2008 Workshop Linked Data on the Web (LDOW), Beijing, China*, page 48, 2008.

[75] Peter F Patel-Schneider, Patrick Hayes, Ian Horrocks, et al. OWL 2 Web Ontology Language structural specification and functional-style syntax (second edition). *W3C Recommendation*, 2012.

[76] Borislav Popov, Atanas Kiryakov, Angel Kirilov, Dimitar Manov, Damyan Ognyanoff, and Miroslav Goranov. KIM–semantic annotation platform. In *The Semantic Web-ISWC 2003*, pages 834–849. Springer, 2003.

[77] Ana Armas Romero, Bernardo Cuenca Grau, and Ian Horrocks. MORe: Modular combination of OWL reasoners for ontology classification. In *The Semantic Web–ISWC 2012*, pages 1–16. Springer, 2012.

[78] Mauro San Martín and Claudio Gutierrez. Representing, querying and transforming Social Networks with RDF/SPARQL. In *The Semantic Web: Research and Applications*, pages 293–307. Springer, 2009.

[79] Evren Sirin, Bijan Parsia, Bernardo Cuenca Grau, Aditya Kalyanpur, and Yarden Katz. Pellet: A practical OWL-DL reasoner. *Web Semantics: science, services and agents on the World Wide Web*, 5(2):51–53, 2007.

[80] M.Q. Stearns, C. Price, K.A. Spackman, and A.Y. Wang. SNOMED Clinical Terms: overview of the development process and project status. In *Proceedings of the AMIA Symposium*, page 662. American Medical Informatics Association, 2001.

[81] Umberto Straccia. *Foundations of Fuzzy Logic and Semantic Web Languages.* CRC Press, 2014.

[82] H.J. ter Horst. Completeness, decidability and complexity of entailment for RDF schema and a semantic extension involving the OWL vocabulary. *Web Semantics: Science, Services and Agents on the World Wide Web*, 3(2):79–115, 2005.

[83] Edward Thomas and Jeff Z. Pan. R-quill: Reasoning with a billion triples. In *8th International Semantic Web Conference–ISWC 2009*, 2009.

[84] Edward Thomas, Jeff Z. Pan, and Yuan Ren. TrOWL: Tractable OWL 2 reasoning infrastructure. *The Semantic Web: Research and Applications*, pages 431–435, 2010.

[85] Dimitri Tsarkov and Ian Horrocks. FaCT++ Description Logic reasoner: System description. *Automated Reasoning*, pages 292–297, 2006.

[86] World Wide Web Consortium W3C. OWL 2 Web Ontology Language Document Overview. 2009.

[87] Tom White. *Hadoop: The definitive guide.* " O'Reilly Media, Inc.", 2012.

[88] René Witte, Ninus Khamis, and Juergen Rilling. Flexible ontology population from text: The OwlExporter. In *LREC*, volume 2010, pages 3845–3850, 2010.

[89] Zi-Ke Zhang, Tao Zhou, and Yi-Cheng Zhang. Tag-aware recommender systems: a state-of-the-art survey. *Journal of Computer Science and Technology*, 26(5):767–777, 2012.

[90] Yujiao Zhou, Yavor Nenov, Bernardo Cuenca Grau, and Ian Horrocks. Complete query answering over horn ontologies using a triple store. In *The Semantic Web–ISWC 2013*, pages 720–736. Springer, 2013.

Appendix A

Ontolop

A.1 Extracts from the OMG Request of Proposal for the OntolOp Framework

This appendix includes extracts from the OMG recently issued RFP which requirements coincide with already anticipated extensions to the SMOF standard on which the ontology-driven framework is based.

The reader is invited to take a quick peek, most particularly at:

- The date of issuing of the RFP, along with the different anticipated submission deadlines.
- The objective of the RFP (on page 1).
- The problem statement and the headlines of the different suggested use cases.
- The clauses of Section 6.5 (Sublanguage Conformance Criteria, Metalogical Relationships, Metalanguage Specification).

Object Management Group

109 Highland Avenue
Needham, MA 02494
USA

Telephone: +1-781-444-0404
Facsimile: +1-781-444-0320
rfp@omg.org

Ontology, Model and Specification Integration and Interoperability (OntoIOp) Request For Proposal

OMG Document: ad/2013-08-02

Letters of Intent due: 24 February 2014
Submissions due: 18 December 2014

Objective of this RFP

This RFP solicits proposals for the following:

- A specification for an abstract meta language with an associated meta-model targeted at cross-language interoperability among a class of concrete languages used to record logical expressions found in ontologies, models and specifications.
- A list of concrete languages and translations to be recognized and correctly processed by implementations of this specification.
- A description of constraints and conformance criteria for additional concrete languages and translations between concrete languages that are not explicitly supported, but nonetheless have equivalent uses that could be recognized and correctly processed by implementations.

For further details see Section 6 of this document.

1 Introduction

1.1 [...]

2 Architectural Context

[...]

3 Adoption Process

3.1 [...]

4 Instructions for Submitters

4.1 [...]

5 General Requirements on Proposals

[...]

6 Specific Requirements on Proposals

6.1 *Problem Statement*

Logical languages are used in several fields of computing for the development of formal, machine-processable texts that carry a formal semantics. Among those fields are 1) ontologies formalizing domain knowledge, 2) formal models of systems, and 3) the formal specification of systems.

A great diversity of logical languages with model-theoretic semantics is in use for these purposes:

- the ontology languages OWL [OWL2], RDF [RDF, RDF-Semantics], RDFS [RDFS],
- the modeling language UML [UML] (fUML [FUML] equips part of UML with a formal semantics)
- general-purpose first-order languages: TPTP FOF, TPTP TFF [TPTP], F-logic [FLogic], Common Logic [CL]
- more specialized specification logics like modal logics, temporal logics, higher-order logics, TPTP THF [TPTP]
- more complex fully-fledged specification languages like VDM [VDM], B [B], Z [Z], CASL [CASL]
- the rule languages in the RIF [RIF] (Rule Interchange Format) and RuleML [RuleML] families of languages, as well as in OMG PRR (at least as far they are based on monotonic logics; for non-monotonic logics, see the non-mandatory requirements section)
- further languages listed in the discussion section

This great diversity of languages is partly justified by different application areas and by different technical properties of the languages. However, often the diversity makes interoperability of ontologies, models, specifications and systems more difficult. Moreover, it is not possible to find a single logical language into which all others can be mapped; rather, it is necessary to adopt a heterogeneous approach to interoperability.

Heterogeneity can be seen at both the level of the ontology languages as well as the ontologies themselves. For example, it is possible to specify the metalogical relationships among the ontology languages currently being used:

For complex logical theories, often several of these languages are used together:

Use case 1
It is common practice to informally annotate OWL ontologies with FOL axioms (e.g. Keet's mereo-topological ontology[1], Dolce Lite[2], BFO-OWL[3]). Moreover, the OMG Ontology Definition Metamodel (ODM) provides a variety of transformations between languages. OntoIOp will free the user from the necessity to ban such FOL axioms into informal annotations, and provide means to replace such informal annotation by formal axioms in a suitable ontology language.

Use case 2

The OMG Date-Time Vocabulary (DTV) has been formulated in different languages, each of which addresses different audiences:

- SBVR: business users
- UML: software implementers
- OWL: ontology developers and users
- Common Logic: (foundational) ontology developers and users

With OntoIOp, one can e.g.

- formally relate the different formalizations,
- specify the OWL version to be an approximation of the Common Logic version, and
- extract submodules covering specific aspects.

Use case 3
A UML model involving different diagram types shall be checked for semantic consistency. Once a formal semantics for the different diagram types has been chosen[4], it is possible to define what overall consistency means, and check this by suitable tools.

Use case 4
A temporal logic specification shall be checked against some process model, which is then refined into some finite automaton.

Use case 5
Refinement of some UML protocol state machine (possibly enriched with some UML sequence diagrams and OCL constraints) to a UML behaviour state machine.

Use case 6

The use of RDFS or OWL to specify a taxonomy of sorts for a more expressive logic with many-sorted semantics (like CASL).

Use case 7

---

[1] Keet, C. M., and Artale, A. Representing and reasoning over a taxonomy of part–whole relations. Applied Ontology 3, 1 (2008), 91–110.

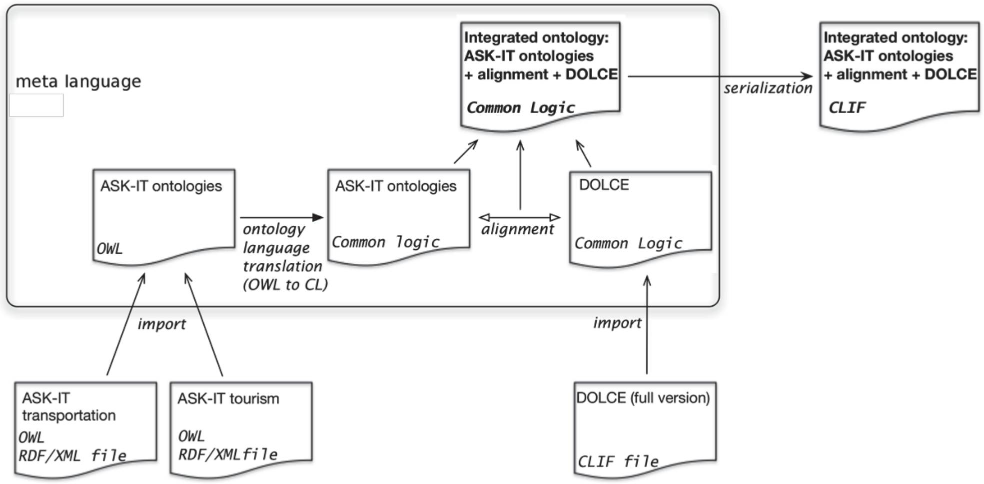
[2] http://www.loa.istc.cnr.it/DOLCE.html

[3] http://bfo.googlecode.com/svn/releases/2012-07-20-graz/owl-group/bfo.owl

[4] See e.g. María Victoria Cengarle, Alexander Knapp, Andrzej Tarlecki, Martin Wirsing: A Heterogeneous Approach to UML Semantics

The use of Common Logic to express metadata concerning modelling assumptions for simulation (e.g. climate change) datasets (e.g. in Datalog). The Datalog representation assumes a closed world on the observed dataset. The Common Logic theory is open-world and describes the physical laws of the object of observation.

Use Case 8
A domain ontology written in OWL shall be integrated with a foundational ontology written in Common Logic (see figure).



There are however no systematic approaches to supporting such use cases, nor is there any way to choose the involved language translations.

Another diversity is that of various operations and relations on logical theories that are in use:

- matching and alignment of different ontologies covering one domain. Note that the task of finding or construct matching and alignments is outside the scope of this RFP; proposals will only provide a meta language for writing these down.
- interpretation and refinement of logical theories
- module extraction - get relevant information out of large logical theories
- approximation – model in an expressive language, reason fast in a lightweight one
- querying
- ontology-based database access/data management
- bridges between different axiomatizations, e.g. distributed description logics, E-connections
- translations of logical theories to other languages
- combinations of logical theories

There are no standardised methods (languages, workflows, tools) for dealing with this variety. Again, this is an obstacle for (formal) interoperability.

Hence, a main target of this RFP is to provide a meta language for interoperability among logical theories, with a well-defined semantics and model theory.

6.2 *Scope of Proposals Sought*

Proposals shall face the diversity of languages, and not add to it by proposing yet another language that would subsume all the others. Instead, proposals shall accept the diverse reality and formulate means (on a sound and formal semantic basis) to compare and integrate logical theories (representing ontologies, models, specifications…) that are written in different formalisms.

Proposals shall specify a meta language which shall be able to handle logical theories formulated in specific languages (as listed in 6.1), as well as provide means for expressing modularity operations and relations between logical theories, even if these are formalized in different logical languages.
Thus, the meta language shall enable interoperability with a formal grounding and make heterogeneous logical theories based on them amenable to checking of coherence (e.g. consistency, conservativity, intended consequences, and compliance).

Within the OntoIOp framework, existing logical theories in conforming established languages shall remain as they are, acknowledging the wide tool support these languages enjoy. The proposed meta language will enhance their modularity facilities to a superset of the modularity and annotation facilities they provide themselves. The meta language's modularity constructs are semantically well-founded within a library of formal relationships between the logics underlying the different supported logical languages.
Proposals shall specify a meta language providing constructs for

a) heterogeneous logical theories (ontologies, models and specifications) that combine parts written in different languages
b) links between distributed and heterogeneous (possibly structured) logical theories associating globally unique identifiers [URI, IRI, URI-Fragment] to any symbol, sentence, ontology and ontology link to allow for reference and annotation by means other than the meta language itself
c) translations between different logical languages
d) a formal semantics of (a)–(d)
e) criteria for existing or future logical languages and translations to conform with OntoIOp

6.3 *Relationship to other OMG Specifications and activities*

6.3.1 Relationship to OMG specifications

- Ontology Definition Metamodel (ODM) provides a graph of ontology languages and translations. Note that it captures abstract syntax only (using MOF meta models), not model theory. Proposals shall build on and may extend this graph, and have to consider model theory.
- Model Driven Architecture (MDA)
- Meta Object Facility (MOF) – meta language that will be used for the specification of abstract syntaxes of languages
- MOF Support for Semantic Structures (SMOF) - extension of MOF for multiple classifications and instantiations
- XML Metadata Interchange (XMI) - standard for exchanging metadata information via Extensible Markup Language (XML)
- MOF 2 XMI Mapping - mapping allowing the storage of MOF models as XMI/XML data
- MOF Model to Text Transformation Language (Mof2Text) - useful for specifying the transformation from the MOF model of the abstract syntax to the concrete syntax
- Unified Modeling Language (UML) – one specific language whose conformance with OntoIOp shall be established. Submitters shall use UML 2.4.1 or later
- Production Rule Representation (PRR) – one specific language whose conformance with

OntoIOp may be established
- Semantics of Business Vocabulary and Business Rules (SBVR) – one specific language whose conformance with OntoIOp may be established (see discussion)
- Date-Time Vocabulary (DTV) – use case for OntoIOp, as it has been implemented in UML, OCL, SVBR, Common Logic and OWL

6.3.2 Relationship to other OMG Documents and work in progress

- Application Programming Interfaces to Knowledge Bases (API4KB) – API for heterogeneous knowledge bases, for which OntoIOp can provide a language and semantic basis. Vice versa, API4KB will be of importance when implementing OntoIOp, and developing OntoIOp-related APIs (see discussion).
- Semantic Information Modeling for Federation (SIMF) – RFP related to OntoIOp but with a different scope

6.4 *Related non-OMG Activities, Documents and Standards*

ISO
- WD (Working Draft) 17347 OntoIOp (ontoiop.org) developed within ISO TC 37/SC 3/WG 3 - initiative similar to the present one that has been cancelled in the meantime; the aim is to have a liaison with ISO
- Metadata Repository (ISO 19763, ISO 11179), Terminology, Metamodeling - standards for metadata. In particular, these standard's practices for allocating identifiers, and for associating downloadable human- and machine-readable encodings of descriptions of logical languages with such identifiers will be of interest of OntoIOp.
- Common Logic (ISO 24707) - family of languages that may be shown to be conformant with OntoIOp
- SQL - individual language that may be shown to be conformant with OntoIOp

W3C
- OWL, RDF, RDFS, RIF, SKOS - these are W3C standards defining individual languages that may (or, in the case of OWL and RDF; shall) be shown to be conformant with OntoIOp

Other
- Open Ontology Repository Initiative (OOR) - aims at ontology repositories covering multiple ontology languages
- NeOn project - defines a number of modularity operations
- Future Internet Enterprise Systems (FInES)
- Software Platform for Integration of Engineering and Things (SPRINT)
- schema.org - RDFS-like schema developed by big search engines with the goal of structuring meta data for web pages

6.5 *Mandatory Requirements*

6.5.1 Metalogical Relationships

Proposals shall provide a specification of a meta-language for the following relationships:

6.5.1.1 ***logically heterogeneous logical theories, particularly in the case in which the application T(Th) of a language translation T : L1 → L2 to a logical theory Th written in language L1,***

6.5.1.2 ***modular logical theories,***

6.5.1.3 ***relationships between logical theories and their extracted modules e.g. the whole theory is a conservative extension of the module***

6.5.1.4 ***approximations of logical theories in less expressive languages such that the approximation is logically implied by the original theory***

6.5.1.5 ***links such as imports, interpretations, equivalences, and alignments between logical theories/modules,***

6.5.1.6 ***combination of logical theories along links.***

.

6.5.2 Applicability to Multiple Logics

The constructs of the meta language shall be applicable to different logics.

6.5.2.1 ***The meta language shall neither be restricted to logical theories in a specific domain, nor to logical theories represented in a specific logical language.***

6.5.2.2 ***The meta language shall provide syntactic constructs for***

- The meta language shall not provide its own constructs for expressing sentences. Instead, it shall inherit the logical language aspects of conforming logical languages.
- structuring logical theories regardless of the logic in which their sentences are formalized;
- basic and structured logical theories and facilities to identify them in a globally unique way;

6.5.3 Specification of the Metalanguage

Proposals shall provide the following specifications for the meta language:

6.5.3.1 ***an abstract syntax specified as an SMOF compliant meta model;***

6.5.3.2 ***a human-readable lexical concrete syntax in EBNF [EBNF] and serialization in XML [XML];***

6.5.3.3 ***complete round-trip mappings from the human-readable concrete syntax to the abstract syntax and vice versa;***

6.5.3.4 ***a formal semantics for the abstract syntax, including the relationships in 6.5.1 and the constructs in 6.5.2.***

.

6.5.4 Scope of Conformant Logical Languages

Proposals shall be applicable to any logical language which either has a formal, logic-based semantics with notions of satisfiability and entailment, or which has a semantics defined by translation to another logical language with such a formal semantics.

6.5.4.1 ***Existing logical theories in existing serializations (e.g. the XML-based XCL serialization of Common Logic, or the text-based OWL Manchester Syntax) shall validate as logical theories the meta language with a minimum amount of syntactic adaptation.***

6.5.4.2 ***It shall be possible to refer to existing files/documents from a logical theory implemented in the meta language without the need for modification***

6.5.5 Conformance Criteria

Proposals shall specify formal criteria for establishing the conformance of a logical language and/or translation as required in 6.5.4.

6.5.5.1 ***Informative annexes shall establish the conformance of a number of relevant logical languages. An initial set of language translations may be part of an informative annex,***

6.5.5.2 ***Conformance of the following subset of logical languages (see 6.1 above) shall be established: OWL2 (with profiles EL, RL, QL, see [OWL2-Profiles]), CLIF [CL], RDF, RIF-Core.***

6.5.6 Registry of Logical Languages

Proposals shall specify the technology and the rules and procedures for maintaining a *registry* of all conforming logical languages.

Appendix B

SMOF Structures and Elements

B.1 Introduction

This appendix presents the most important structures in the knowledge base metadata repository.

It describes the Data Dictionary of the different structures, grouped under three schemas (SMOF for global SMOF structures, ODM for ODM-specific structures and SYS for System and UC-relevant data).

Following are guidelines on the documentation conventions employed in the data dictionary:

- The first column, entitled “STRUCTURE NAME-Field Name”, contains the structure name (when the description is in upper case) and element name (when the contents are in lower case).
- The second column, entitled “Datatype”, contains the element’s data type (a primitive form).
- The third column, entitled “Nul”, indicates, when set to “Y”, that the element is optional; and when set to “N”, it denotes the element is mandatory.

- The fourth column, entitled "Details", contains brief descriptions on the structures and elements.
- Primary key elements of a structure have their names highlighted in Bold characters.
- Parent tables are referenced in the "Details" column.

The structures having their names starting with "SMOF" are common to all logics and languages. They need to be populated (database population) according to their type right before the population of the element that will refer to them.

SMOF Extension Structures - Definition of all elements, associations to logical levels/sub-levels, element-to-element relationships			
STRUCTURE NAME/ Field Name	**Data Type**	**Nul**	**Details**
SMOF_TYP_DEF			Global System Dictionary Table. Defines and lists all type codes and values, per category.
smof_typ_cd	number	N	Internal Unique Code per type
smof_typ_desc	varchar2	N	Type Description. Examples are provided in children structures referring to smof_typ_cd
smof_typ_categ	varchar2	N	Category of Type Codes; Examples: Logic; Sublogic; Element Type; Restriction Type, etc.
SMOF_ELT_DEF			Global Definition Table. Defines all SMOF elements, regardless their type.
smof_elt_id	number	N	SMOF Element ID - Surrogate Key
smof_elt_typ_cd	number	N	Parent Structure: SMOF_TYP_DEF Examples: OWL Ontology; OWL Axiom, OWL Property, DataType, Class, etc.
depreciated_smof_elt_id	number	Y	Parent Structure: SMOF_ELT_DEF. (used for elements not having their own definition structure)
smof_elt_expression	varchar2	Y	Natural Language Expression corresponding to this element
SMOF_ELT_LOGIC_MATRIX			Meta-semantics Structure (Extension). Association of SMOF elements to logical levels. (Languages)
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF
smof_log_typ_cd	number	N	Parent Structure: SMOF_TYP_DEF Logic Type Code Examples: OWL, RDF, RDFS
SMOF_ELT_SUB_LOGIC_MATRIX		1-1	Meta-semantics Structure (Extension). Association of SMOF elements to logical levels. (sublanguages).
smof_elt_id	number	N	Parent Structure: SMOF_ELT_LOGIC_MATRIX
smof_log_typ_cd	number	N	Parent Structure: SMOF_ELT_LOGIC_MATRIX Logic Type Code Examples: OWL, RDF, RDFS
smof_sub_log_typ_cd	number	Y	Parent Structure: SMOF_TYP_DEF Sublevel logic Type Code Examples: OWL DL Restr., OWL DL, EL, QL, RL

STRUCTURE NAME/ Field Name	**Data Type**	**Nul**	**Details**
SMOF_ELT_DEPRECIATION			Meta-semantics Structure (Extension). Identification of SMOF elements depreciated version per logic sub-level
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF
isDLDeprecated	boolean	N	Flag determining whether SMOF element is deprecated in DL
isELDeprecated	boolean	N	Flag determining whether SMOF element is deprecated in EL
isQLDeprecated	boolean	N	Flag determining whether SMOF element is deprecated in QL
isRLDeprecated	boolean	N	Flag determining whether SMOF element is deprecated in RL
DLDepreciated_elt_id	number	Y	Parent Structure: SMOF_ELT_DEF Depreciated version corresponding to deprecated DL element
ELDepreciated_elt_id	number	Y	Parent Structure: SMOF_ELT_DEF Depreciated version corresponding to deprecated EL element
QLDepreciated_elt_id	number	Y	Parent Structure: SMOF_ELT_DEF Depreciated version corresponding to deprecated QL element
RLDepreciated_elt_id	number	Y	Parent Structure: SMOF_ELT_DEF Depreciated version corresponding to deprecated RL element
SMOF_ELT_RLTNP			SMOF element to element association. Identifying relationship between elements according to their types.
src_smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF Source element.
tgt_smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF Target element.
smof_rltnp_typ_cd	number	N	Parent Structure: SMOF_TYP_DEF Relationship Type Code. Examples: Containment, Importing, Imported

ODM Structures, with Linkage to SMOF elements

STRUCTURE NAME/ Field Name	Data Type	Nul	Details
ODM_RDF_TRIPLE_DEF			Resource Definition Table. Defines all SMOF elements, regardless their type.
rdf_triple_id	number	N	RDF Triple ID - Surrogate Key
smof_elt_id	number	Y	Parent Structure: SMOF_ELT_DEF SMOF Element ID
rdf_triple_content	xmltype	N	RDF Triple containment (subject, predicate, object)
ODM_RDFS_RESOURCE_DEF			Resource Definition Table. Defines all SMOF elements, regardless their type.
rdfs_rsrc_id	number	N	RDFS Resource ID - Surrogate Key
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
uri_ref	varchar2	Y	URI Reference
ODM_ELT_RDFS_RESOURCE			Linkage of SMOF elements to RDFS Resources/URIs
smof_elt_id	number	N	Parent Structure: SMOF_TYP_DEF
rdfs_rsrc_id	number	N	Parent Structure: ODM_RDFS_RESOURCE_DEF
ODM_OWL_ONTOLOGY_DEF			OWL Ontology Definition. Defines all OWL ontologies; An OWL ontology contains a sequence of annotations, axioms, and facts.
owl_ontol_id	number	N	OWL Ontology ID - Surrogate Key
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
ontol_name	varchar2	N	Descriptive URI/Name of the ontology; the name of an ontology in the abstract syntax is the URI where it can be found
ODM_OWL_ONTOLOGY_ANNOTATION			OWL Ontology Annotations. Annotations on OWL ontologies can be used to record authorship and other information associated with an ontology, including imports references to other ontologies.
ontol_annot_id	number	N	Ontology Annotation ID - Surrogate Key
owl_ontol_id	number	N	Parent Structure: ODM_OWL_ONTOLOGY_DEF
ontol_annot_typ_cd	number	N	Parent Structure: SMOF_ELT_DEF Ontology Annotation Type Code. Examples: Versioning Info, backwardCompatible info, importing info, etc.
ontol_annot_data	xmltype	N	Ontology Annotation Type Data, according to type code. Examples: Versioning Info Data (author, date, etc.); Imports Info Data (owl_ontol_id, backwardcompatiblewith, etc.)

STRUCTURE NAME/ Field Name	**Data Type**	**Nul**	**Details**
ODM_OWL_ONTOLOGY_CLASS			OWL Ontology Classes. Classes provide an abstraction mechanism for grouping resources with similar characteristics.
ontol_class_id	number	N	Ontology Class ID - Surrogate Key
owl_ontol_id	number	N	Parent Structure: ODM_OWL_ONTOLOGY_DEF
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
owl_ontol_name	varchar2	N	Describes a class through a class name (syntactically represented as a URI reference)
isDeprecated	boolean	N	Indicates whether this class is deprecated
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated class
ODM_OWL_ONTOLOGY_CLASS_EQUIV			OWL Ontology Class Equivalence.
ontol_class_id	number	N	Parent Structure: ODM_OWL_ONTOLOGY_CLASS
equiv_ontol_class_id	number	N	Parent Structure: ODM_OWL_ONTOLOGY_CLASS
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
owl_expression	varchar2	N	Describes the class expression in the owl syntax
isDeprecated	boolean	N	Indicates whether this class axiom is deprecated
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated class axiom
ODM_OWL_ONTOLOGY_CLASS_DISJNT			OWL Ontology Class Disjointness.
ontol_class_id	number	N	Parent Structure: ODM_OWL_ONTOLOGY_CLASS
disjoint_ontol_class_id	number	N	Parent Structure: ODM_OWL_ONTOLOGY_CLASS
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
owl_expression	varchar2	N	Describes the class expression in the owl syntax
isDeprecated	boolean	N	Indicates whether this class axiom is deprecated
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated class axiom

STRUCTURE NAME/ Field Name	**Data Type**	**Nul**	**Details**
ODM_OWL_ONTOLOGY_CLASS_COMPL			OWL Ontology Class ComplementOf. Describes a class for which the class extension contains exactly those individuals that do not belong to the class extension of the class description that is the object of the statement
axiom_smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF
complement_smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
owl_expression	varchar2	N	Describes the class expression in the owl syntax
isDeprecated	boolean	N	Indicates whether this class axiom is deprecated
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated class axiom
ODM_OWL_ONTOLOGY_CLASS_UNION			OWL Ontology Class UnionOf.
axiom_smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF
union_smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
owl_expression	varchar2	N	Describes the class expression in the owl syntax
isDeprecated	boolean	N	Indicates whether this class axiom is deprecated
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated class axiom
ODM_OWL_ONTOLOGY_CLASS_INTERS			OWL Ontology Class IntersectionOf.
axiom_smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF
intersect_smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
owl_expression	varchar2	N	Describes the class expression in the owl syntax
isDeprecated	boolean	N	Indicates whether this class axiom is deprecated
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated class axiom

STRUCTURE NAME/ Field Name	**Data Type**	**Nul**	**Details**
ODM_OWL_ONTOLOGY_CLASS_RESTR			OWL Ontology Class Restrictions OWL property restrictions describe special kinds of class descriptions. OWL distinguishes two kinds of property restrictions: value constraints and cardinality constraints.
axiom_smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF
restr_smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF
restriction_typ_cd	number	N	Parent Structure: SMOF_TYP_DEF Restriction Type: value constraints, cardinality constraints
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
owl_expression	varchar2	N	Describes the class expression in the owl syntax
isDeprecated	boolean	N	Indicates whether this class axiom is deprecated
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated class axiom
ODM_OWL_ONTOL_CLASS_RESTR_DET			OWL Ontology Class Restriction Details Details of OWL property restrictions; describe special kinds of class descriptions. OWL distinguishes two kinds of property restrictions: value constraints and cardinality constraints.
axiom_smof_elt_id	number	N	Parent Structure: ODM_OWL_ONTOLOGY_CLASS_RESTR
restr_smof_elt_id	number	N	Parent Structure: ODM_OWL_ONTOLOGY_CLASS_RESTR (Property, Property with qualifier)
restriction_typ_cd	number	N	Parent Structure: ODM_OWL_ONTOLOGY_CLASS_RESTR
restr_det_typ_cd	number	N	Parent Structure: SMOF_TYP_DEF Restriction Detail Type: HasValueRestriction, AllValuesFromRestriction, SomeValuesFromRestriction, CardinalityRestriction, MaxCardinalityRestriction, MinCardinalityRestriction
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
owl_expression	varchar2	N	Describes the class restriction expression in the owl syntax
isDeprecated	boolean	N	Indicates whether this class axiom is deprecated
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated class axiom

STRUCTURE NAME/ Field Name	Data Type	Nul	Details
ODM_OWL_ONTOL_CLASS_REST_DET_R			DataRange for OWL Ontology Class Restriction Details (for AllValuesFromRestriction and SomeValuesFromRestriction) OWL provides two constructs for defining a range of data values, namely (1) an enumerated datatype, which is an enumerated list of literals or (2) it identifies a specific datatype class from the RDF datatypes (e.g., xsd:integer) that a value in the data range must reflect.
axiom_smof_elt_id	number	N	Parent Structure: ODM_OWL_ONTOL_CLASS_RESTR_DET
restr_smof_elt_id	number	N	Parent Structure: ODM_OWL_ONTOL_CLASS_RESTR_DET
restriction_typ_cd	number	N	Parent Structure: ODM_OWL_ONTOL_CLASS_RESTR_DET
restr_det_typ_cd	number	N	Parent Structure: ODM_OWL_ONTOL_CLASS_RESTR_DET Restriction Detail Type: HasValueRestriction, AllValuesFromRestriction, SomeValuesFromRestriction, CardinalityRestriction, MaxCardinalityRestriction, MinCardinalityRestriction
datatype_smof_elt_id	number		Parent Structure: SMOF_ELT_DEF Element representing an enumerated datatype, which is an enumerated list of literals or (2) it identifies a specific datatype class from the RDF datatypes (e.g., xsd:integer) that a value in the data range must reflect
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
owl_expression	varchar2	N	Describes the class restriction expression in the owl syntax
isDeprecated	boolean	N	Indicates whether this class axiom is deprecated
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated class axiom
ODM_OWL_ONTOLOGY_CLASS_ENUM			OWL Ontology Class Enumerations (oneOf) A list of individuals that are the instances of the class.
axiom_smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF
enum_smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF (Individual)
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
restriction_typ_cd	number	N	Parent Structure: SMOF_TYP_DEF Restriction Type: value constraints, cardinality constraints
owl_expression	varchar2	N	Describes the class expression in the owl syntax
isDeprecated	boolean	N	Indicates whether this class axiom is deprecated
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated class axiom

STRUCTURE NAME/ Field Name	**Data Type**	**Nul**	**Details**
ODM_OWL_ONTOLOGY_INDIVIDUAL			OWL Ontology Individuals. Individuals are defined with individual axioms (also called “facts”). Two types of facts are supported in OWL: (1) Facts about class membership and property values of individuals, and (2) Facts about individual identity
individual_id	number	N	Individual ID - Surrogate Key
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
owl_ontol_id	number	N	Parent Structure: ODM_OWL_ONTOLOGY_DEF
owl_expression	varchar2	N	Describes the class expression in the owl syntax
isDeprecated	boolean	N	Indicates whether this class axiom is deprecated
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated class axiom
ODM_OWL_ONTOLOGY_INDIV_ASSOC			OWL Ontology Individuals Associations
individual_id	number	N	Parent Structure: ODM_OWL_ONTOLOGY_INDIVIDUAL
assoc_smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF (Other Individual or some Restriction)
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
assoc_typ_cd	number	N	Parent Structure: SMOF_TYP_DEF Associaion Type: enumeratedClass, owlSameAs, owlDifferentFrom, restrictionClass
owl_expression	varchar2	N	Describes the class expression in the owl syntax
isDeprecated	boolean	N	Indicates whether this class axiom is deprecated
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated class axiom
ODM_OWL_PROPERTY			All Properties Definition. OWL refines the notion of an RDF property to support two main categories of properties as well as annotation properties that may be useful for ontology documentation.
property_id	number	N	Ontology Class ID - Surrogate Key
owl_ontol_id	number	N	Parent Structure: ODM_OWL_ONTOLOGY_DEF
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
property_name	varchar2	N	Describes a property through a name (syntactically represented as a URI reference)
isDeprecated	boolean	N	Indicates whether this property is deprecated
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated property

STRUCTURE NAME/ Field Name	**Data Type**	**Nul**	**Details**
ODM_OWL_PROPERTY_ANNOTATION			Property 's OWLAnnotationProperty. (owl:versionInfo, rdfs:label, rdfs:comment, rdfs:seeAlso, rdfs:isDefinedBy)
property_id	number	N	Parent Structure: ODM_OWL_PROPERTY
property_annot_data	xmltype	N	Annotation Data for OWLAnnotationProperty. (also used for NLG entries)
ODM_OWL_PROPERTY_EQUIV			Property that is also a FunctionalProperty. A functional property is a property that can have only one (unique) value y for each instance x
property_id	number	N	Parent Structure: ODM_OWL_PROPERTY
equiv_property_id	number	N	Parent Structure: ODM_OWL_PROPERTY
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated property equivalence
ODM_OWL_PROPERTY_FUNCTIONAL			Property that is also a FunctionalProperty. A functional property is a property that can have only one (unique) value y for each instance x
property_id	number	N	Parent Structure: ODM_OWL_PROPERTY
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated property characteristic
ODM_OWL_PROPERTY_DATA			Property that is a DataTypeProperty. Datatype properties are used to link individuals to data values. A datatype property is defined as an instance of the built-in OWL class owl:DatatypeProperty.
property_id	number	N	Parent Structure: ODM_OWL_PROPERTY
datatype_smof_elt_id	number		Parent Structure: SMOF_ELT_DEF Element representing an enumerated datatype, which is an enumerated list of literals or (2) it identifies a specific datatype class from the RDF datatypes (e.g., xsd:integer) that a value in the data range must reflect
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated property characteristic
ODM_OWL_PROPERTY_OBJECT			Property that is a ObjectProperty. An object property relates an individual to other individuals. An object property is defined as an instance of the built-in OWL class owl:ObjectProperty
property_id	number	N	Parent Structure: ODM_OWL_PROPERTY
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated property characteristic

STRUCTURE NAME/ Field Name	**Data Type**	**Nul**	**Details**
ODM_OWL_PROPERTY_OBJECT_INV			Property that is an Inverse ObjectProperty (InverseProperty).
property_id	number	N	Parent Structure: ODM_OWL_PROPERTY_OBJECT
inv_property_id	number	N	Parent Structure: ODM_OWL_PROPERTY_OBJECT
smof_elt_id	number	Y	Parent Structure: SMOF_ELT_DEF SMOF Element ID
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated property characteristic
ODM_OWL_PROPERTY_OBJECT_INVFUNC			Object Property that is also an InverseFunctionalProperty. If a property is declared to be inverse-functional, then the object of a property statement uniquely determines the subject (some individual)
property_id	number	N	Parent Structure: ODM_OWL_PROPERTY_OBJECT
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated property characteristic
ODM_OWL_PROPERTY_OBJECT_TRANS			Object Property that is also an TransitiveProperty. When one defines a property P to be a transitive property, this means that if a pair (x, y) is an instance of P, and the pair (y, z) is also instance of P, then we can infer the pair (x, z) is also an instance of P.
property_id	number	N	Parent Structure: ODM_OWL_PROPERTY_OBJECT
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated property characteristic
ODM_OWL_PROPERTY_OBJECT_SYMM			Object Property that is also an SymmetricProperty. A symmetric property is a property for which holds that if the pair (x, y) is an instance of P, then the pair (y, x) is also an instance of P.
property_id	number	N	Parent Structure: ODM_OWL_PROPERTY_OBJECT
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated property characteristic

SYS System Structures, with Linkage to UCs, reasoning configs. and SMOF elements			
STRUCTURE NAME/ Field Name	**Data Type**	**Nul**	**Details**
ODM_OWL_PROPERTY_OBJECT_ASYMM			Object Property that is also an AsymmetricProperty. An asymmetric property is a property for which does not hold that if the pair (x, y) is an instance of P, then the pair (y, x) is also an instance of P.
property_id	number	N	Parent Structure: ODM_OWL_PROPERTY_OBJECT
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
depreciated_smof_elt_id	number	Y	Depreciated element corresponding to deprecated property characteristic
SYS_REASONING_DATA			Object Property that is also an AsymmetricProperty. An asymmetric property is a property for which does not hold that if the pair (x, y) is an instance of P, then the pair (y, x) is also an instance of P.
reasoner_id	number	N	UC Name - Surrogate Key
reasoner_name	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
reasoner_data	xmltype	N	Reasoner Properties (technical details, api calls, location, type: fully fledged/specialized etc.)
SYS_REASONING_LOG			Reasoner preferred logic (sub-language) per priority (1 = first priority)
reasoner_id	number	N	Parent Structure: SYS_REASONING_DATA
reasoner_sub_log_typ_cd	number	N	Parent Structure: SMOF_TYP_DEF logic sub-level Type Code Examples: OWL DL Restr., OWL DL, EL, QL, RL
priority_typ_cd	number	N	Parent Structure: SMOF_TYP_DEF Priority Type Code (1, 2, 3, etc.)
SYS_UC_DEFINITION			Object Property that is also an AsymmetricProperty. An asymmetric property is a property for which does not hold that if the pair (x, y) is an instance of P, then the pair (y, x) is also an instance of P.
uc_id	number	N	UC ID - Surrogate Key
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID
uc_objective_typ_cd	number	N	Parent Structure: SMOF_TYP_DEF UC Objective Type Code Examples: extensional, intensional querying
cq_data	xmltype	Y	Competency Question(s) related to the UC, along with their corresponding purpose(s). *(This element will be split into another one-to-many structure, at the time of writing, it has been implemented as such)*

STRUCTURE NAME/ Field Name	**Data Type**	**Nul**	**Details**
SYS_UC_SIGNATURE			Object Property that is also an AsymmetricProperty. An asymmetric property is a property for which does not hold that if the pair (x, y) is an instance of P, then the pair (y, x) is also an instance of P.
uc_id	number	N	Parent Structure: SYS_UC_DEFINITION
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID (signature element)
includeSuperClass	boolean	N	Flag indicating whether signature element superclass are to be included in the module
includeSubClass	boolean	N	Flag indicating whether signature element superclass are to be included in the module
SYS_UC_SMOF_LINK			Links a UC to all SMOF elements (ontology modules, ontology axioms, etc.) and provides when applicable the depreciated version of a deprecated element.
uc_id	number	N	Parent Structure: SYS_UC_DEFINITION
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID Examples: ontology (module), class, axiom, etc. (can bel from FOAF, SIOC or any interoperating ontology)
uc_link_typ_cd	number	N	Parent Structure: SMOF_TYP_DEF Relationship Type Code Examples: associated module, mapped element, contained element, etc.
depreciated_smof_elt_id	number	Y	Depreciated element per UC (priority 1 handling through projection algo, if existing)
SYS_UC_SMOF_TAG			Links a UC to all SMOF elements forming its associated tags
uc_id	number	N	Parent Structure: SYS_UC_DEFINITION
smof_elt_id	number	N	Parent Structure: SMOF_ELT_DEF SMOF Element ID Examples: ontology (module), class, axiom, etc. (can bel from FOAF, SIOC or any interoperating ontology)
tag_weight_typ_cd	decimal	Y	Parent Structure: SMOF_TYP_DEF Weight Type Code . Weight associated to a tag in a UC
tag_generation_typ_cd	boolean	Y	Parent Structure: SMOF_TYP_DEF Tag generation Type Code . (automatic, manual, semi-automatic)

STRUCTURE NAME/ Field Name	**Data Type**	**Nul**	**Details**
SYS_DECISION_MATRIX			Matrix defining the reasoning profile according to different criteria configured and deduced from the UC properties.
matrix_elt_cd	number	N	Internal Unique Identifier
uc_objective_typ_cd	number	N	Parent Structure: SMOF_TYP_DEF UC Objective Type Code Examples: extensional, intensional querying
uc_module_size			Determined based on the modularization resulting ontology per UC signature
el_depreciation_weight			Determined based on projection algorithm according to the EL profile: a factor of n, n being the number of axioms in projected ontology; if 20 elements are in the depreciation list, 20/n is the weight
ql_depreciation_weight			Determined based on projection algorithm according to the QL profile
rl_depreciation_weight			Determined based on projection algorithm according to the RL profile
sub_log_typ_cd	number	N	Parent Structure: SMOF_TYP_DEF logic sub-level Type Code Examples: OWL DL Restr., OWL DL, EL, QL, RL
priority_typ_cd	number	N	Parent Structure: SMOF_TYP_DEF Priority Type Code (1, 2, 3, etc.)

Appendix C

Semantic Querying

C.1 Introduction

This appendix presents extracts from the queries prepared within the scope of the different implementation phases of the thesis.

It presents a set of sample conjunctive queries that come on top of the ones already described in Chapter 6.

C.2 Conjunctive Queries

The following conjunctive queries are extracted samples that can be similarly designed for other concepts and criteria:

- Listing all the foods containing saturated fats while not containing proteins:

$$Q_1(x) =$$

$$\exists \text{x,y (Food(x)} \wedge \text{hasSpecification (x, SaturatedFat)}$$

$$\wedge \text{ Protein(y)} \wedge \neg(\exists \text{hasSpecification(x,y)))}$$

- Listing adolescents manifesting particular irregular behaviors and suffering from particular diseases:

$$Q_2(x) =$$
$$\exists \text{x,y (Adolescent(x)} \land \text{hasBehavior(x, Overeating)}$$
$$\land \text{ EndocrineDisease(y)} \land \text{affectedBy(x,y))}$$

- Listing products or lifestyle components having a constipation-reducing effect:

$$Q_3(x) =$$
$$\exists \text{x ((Product(x)} \land \text{hasGoodEffect (x,ReduceConstipation))}$$
$$\lor \text{ (LifestyleComponent(x)} \land \text{hasGoodEffect (x,ReduceConstipation)))}$$

- Listing all recommended activities for toddlers:

$$Q_4(x) =$$
$$\exists \text{x,y (Toddler(x)} \land \text{Activity(y)}$$
$$\land \text{ isSuitableFor(y,x))}$$

- Recognizing some product that had been promoted on TV and included in the diet habit of a child who presents a “Food Craving” abnormal behavior:

$$Q_5(x) =$$
$$\exists \text{x,y (Child(x)} \land \text{PromotedFood(y)}$$
$$\land \text{ hasIncludedinDiet(x,y)} \land \text{affectedBy(x, FoodCraving))}$$

- Recognizing food products that are classified as having high allergy risks:

$$Q_6(x) =$$
$$\exists \text{x (Food(x)} \land \text{hasAllergyRisk(x,1))}$$

- Recognizing development stages relevant to certain displayed behaviors:

$$Q_7(x) =$$

$$\exists \text{x,y (DevelopmentalStage(x)} \wedge \text{Child(y)}$$

$$\wedge \text{ isAtDevelopmentalStage(y,x)} \wedge \text{hasBehavior(y,z))}$$

---

z in this example should be replaced by the behavior the user is interested in.

- Classifying behaviors according to emotional symptoms:

$$Q_8(x) =$$

$$\exists \text{x (Behavior(x)} \wedge \text{causedByEmotion(x,Anger))}$$

Anger being an instance of the Emotion concept.

- Identifying the behavioral traits of the subsequent developmental stage (given a particular one):

$$Q_9(x) =$$

$$\exists \text{x,y (Behavior(x)} \wedge \text{DevelopmentalStage(y)}$$

$$\wedge \text{ hasNextDevelopmentalStage(ToddlerStage,y)}$$

$$\wedge \text{ hasDevelopmentalStageBehavior(y,x))}$$

- Detecting behavior(s) not displayed by a certain child who is at the developmental stage at which the behavior(s) detected is(are) supposed to be manifested:

$$Q_{10}(x) =$$

$$\exists \text{x,y (Behavior(x)} \wedge \text{DevelopmentalStage(y)}$$

$$\wedge \text{ occursAtStage(x,y)} \wedge \text{hasDevelopmentalStage(John, y)}$$

$$\wedge \neg \text{ hasBehavior(John,x))}$$

- Identifying causes behind certain behaviors:

$$Q_{11}(x) =$$
$$\exists \text{x,y (Behavior(x)} \wedge \text{BehavioralCause(y)}$$
$$\wedge \text{ isCausedBy(x,y))}$$
$$\wedge \text{ hasDevelopmentalStageBehavior(y,x))}$$

- Recognizing fruits that are not suitable for toddlers:

$$Q_{12}(x) =$$
$$\exists \text{x,y (Fruit(x)} \wedge \text{Toddler(y))}$$
$$\wedge\neg \text{ isSuitableFor(x,y))}$$

- Recognizing food products that are classified as having high allergy risks and are suitable for toddlers:

$$Q_{13}(x) =$$
$$\exists \text{x,y (Food(x)} \wedge \text{hasAllergyRisk(x,1))}$$
$$\wedge \text{ Toddler(y)} \wedge \text{isSuitableFor(x,y))}$$

Appendix D

List of Acronyms

Appendix - List of Acronyms

ABox	Assertional Box
AI	Artificial Intelligence
ANNIE	A Nearly New Information Extraction System
API	Application Programming Interface
BTS	Brain-To-Society
CQ	Competency Question
DL	Description Logic
EL	Existential Language
FOAF	Friend-of-a-Friend
GATE	General Architecture for Text Engineering
HaSIE	Health and Safety Information Extraction
IE	Information Extraction

JAPE	Java Annotation Patterns Engine
KB	Knowledge Base
MDA	Model Driven Architecture
MOF	Meta Object Facility
SMOF	Meta Object Facility for Semantics
NE	Named Entity
NLG	Natural Language Generation
NLP	Natural Language Processing
ODM	Ontology Definition Metamodel
OMG	Object Management Group
OWA	Open World Assumption
OWL	Web Ontology Language
PIM	Platform-Independent Model
PSM	Platform-Specific Model
QL	Query Language
RDF	Resource Description Framework
RFP	Request For Proposal
RIF	Rule Interchange Format
RL	Rule Language
RBox	Role Box
SCOT	Semantic Cloud of Tags

SIOC	Semantically Interlinked Online Communities
SKOS	Simple Knowledge Organization System
SN	Social Network(s)/Networking
SNA	Social Network Analysis
SPARQL	SPARQL Protocol and RDF Query Language
TBox	Terminological Box
UC	Use Case
UML	Unified Modeling Language
URI	Uniform Resource Identifier
W3C	World Wide Web Consortium
XMI	XML Metadata Interchange
XML	eXtensible Markup Language
XSL	eXtensible Stylesheet Language